



Guía del usuario de

Amazon Linux 2023



Amazon Linux 2023: Guía del usuario de

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Las marcas comerciales y la imagen comercial de Amazon no se pueden utilizar en relación con ningún producto o servicio que no sea de Amazon, de ninguna manera que pueda causar confusión entre los clientes y que menosprecie o desacredite a Amazon. Todas las demás marcas registradas que no son propiedad de Amazon son propiedad de sus respectivos propietarios, que pueden o no estar afiliados, conectados o patrocinados por Amazon.

Table of Contents

¿Qué es Amazon Linux 2023?	1
Liberar cadencia	1
Versiones principales y secundarias	2
Consumir nuevas versiones	2
Política de soporte a largo plazo	2
Denominación y control de versiones	3
Optimizaciones operativas y de rendimiento	4
Relación con Fedora	6
Personalizar cloud-init	6
Actualizaciones y características de seguridad	8
Actualizaciones administradas	8
Seguridad en la nube	8
modos SELinux	8
Programas de conformidad	9
Servidor SSH predeterminado	9
Características principales de OpenSSL 3	9
Servicio de redes	9
Paquetes principales de cadenas de herramientas glibc, gcc, binutils	10
Herramienta de administración de paquetes	11
Configuración predeterminada del servidor SSH	12
Funcionalidad obsoleta	14
Paquetes de compat-	14
Funcionalidad obsoleta discontinuada en AL1 y eliminada en AL2	14
AMI x86 (i686) de 32 bits	15
aws-apitools-* se ha sustituido por AWS CLI	15
systemd reemplaza a upstart en AL2	16
Funcionalidad obsoleta AL2 y eliminada en AL2023	16
Paquetes x86 (i686) de 32 bits	17
aws-apitools-*reemplazado por AWS CLI	18
amazon-cloudwatch-agentreemplaza awslogs	18
Sistema de control de revisiones bzip	18
cgroup v1	19
Hotpatch de log4j (log4j-cve-2021-44228-hotpatch)	19
lsb_release y el paquete system-lsb-core	19

mcrypt	20
OpenJDK 7 (java-1.7.0-openjdk)	20
Python 2.7	20
rsyslog-opensslreemplaza rsyslog-gnutls	21
Servicio de información de red (NIS)/yp	21
Varios nombres de dominio en Amazon VPC create-dhcp-options	21
Sun RPC en glibc	22
Huella digital clave de OpenSSH en el registro audit	22
Vinculador de ld.gold	22
ping6	22
Paquete ftp	23
Obsoleta en AL2023	24
Compatibilidad con tiempo de ejecución de 32 bits x86 (i686)	25
aspell	25
Berkeley DB (libdb)	25
cron	26
IMDSv1	26
pcre versión 1	27
System V init (sysvinit)	27
Los paquetes EOL están obsoletos	27
¿Comparando AL2 y 023 AL2	29
Paquetes añadidos, actualizados y eliminados	30
Soporte para cada versión	30
Cambios de nombres y control de versiones	30
Optimizaciones	31
Proviene de múltiples fuentes	31
Servicio del sistema de redes	31
Administrador de paquetes	31
Uso de cloud-init	31
Soporte de escritorio gráfico	32
Triplete del compilador	32
Paquetes x86 (i686) de 32 bits	32
lsb_release y el paquete system-lsb-core	33
EPEL	33
axel- HTTP/FTP cliente	35
brotli y libbrotli: compresión	36

collectd: daemon de recopilación de estadísticas	36
cpulimit	36
exim: agente de transferencia de correo	37
fuse3: sistema de archivos en el espacio de usuario (FUSE) v3	37
ganglia: sistema de monitoreo distribuido	37
git-lfs: control de versiones de archivos grandes con Git	37
haveged: una fuente de entropía que utiliza el algoritmo HAVEGE	38
inotify-tools: herramientas de línea de comando de inotify	38
iperf- Punto de referencia TCP/UDP de rendimiento	38
jemalloc: implementación alternativa de malloc	39
libbsd: biblioteca de funciones compatible con BSD	39
libserf: biblioteca de cliente de HTTP	39
libzstd: biblioteca de compresión zstd	39
Servidor web lighttpd	40
lshell: un intérprete de comandos restringido	40
monit: monitor de procesos, archivos, directorios y dispositivos	40
nodejs	41
perl-Config-General	41
python2-lockfile: bloqueo de archivos	41
python2-rsa: Python RSA puro	42
python2-simplejson: rutinas JSON para Python 2	42
rkhunter: Rootkit Hunter	42
rssh: un intérprete de comandos restringido para usar con OpenSSH	43
sscg: generador de certificados SSL autofirmado	43
stress: prueba de esfuerzo	43
stress-ng: prueba de esfuerzo	43
tmpwatch: elimina los archivos en función de la hora del último acceso	44
xmlstarlet: utilidades XML de línea de comandos	44
Python 2.7 ha sido reemplazado por Python 3	44
Actualizaciones de seguridad	44
SELinux	45
OpenSSL 3	45
IMDSv2	45
Eliminación del hotpatch log4j (log4j-cve-2021-44228-hotpatch)	46
Actualizaciones deterministas para mayor estabilidad	47
gp3 como tipo de volumen de Amazon EBS predeterminado	47

Jerarquía de grupos de control unificados (cgroup v2)	47
Los temporizadores de systemd reemplazan a cron	48
Cadena de herramientas mejorada: gcc, binutils y glibc	48
El diario systemd reemplaza a rsyslog	49
Dependencias del paquete minimizadas	50
Cambios en los paquetes para curl y libcurl	50
GNU Privacy Guard (GNUPG)	50
Amazon Corretto como JVM predeterminada	51
AWS CLIV2	51
UEFI Preferred y arranque seguro	51
Cambios en la configuración predeterminada del servidor SSH	51
El núcleo cambia AL2023 de AL2	52
IPv4 TTL	52
Cambios en la configuración del kernel centrados en la seguridad	52
Otros cambios en la configuración del kernel	57
Compatibilidad con sistema de archivos del kernel	59
Cambios de /tmp	64
Cambios en la AMI y en la imagen del contenedor	64
Comparación de la AMI de Amazon Linux 2 y AL2023.	65
Comparación de la AMI de Amazon Linux 2 y AL2023	98
Comparación del contenedor de Amazon Linux 2 y AL2023	119
Comparación de AL1 y AL2023	127
Soporte para cada versión	127
systemd reemplaza upstart como sistema init	128
Python 2.6 y 2.7 ha sido reemplazado por Python 3	128
OpenJDK 8 como el JDK más antiguo	128
El núcleo cambia AL2023 de AL1	128
Kernel Live Patching	128
Compatibilidad con sistema de archivos del kernel	128
Cambios en la configuración del kernel centrados en la seguridad	131
Otros cambios en la configuración del kernel	134
Comparación entre la AMI de AL1 y AL2023	135
Comparación de AMI mínima de AL1 y AL2023	169
Comparación del contenedor AL1 y AL2023	189
Requisitos del sistema	198
Requisitos de CPU para ejecutar AL2023	198

Requisitos de CPU ARM para AL2023	198
Requisitos de CPU x86-64 para AL2023	199
Requisitos de memoria (RAM) para ejecutar AL2023	200
El núcleo de Amazon Linux	201
ciclo de vida del núcleo	201
Actualizaciones del kernel	202
Las versiones del núcleo se superponen entre distribuciones	202
Manejo de CVE	203
¿Qué debe hacer	203
Escritorio gráfico	204
Temas relacionados	204
Paquetes complementarios para Amazon Linux	205
¿Qué son los paquetes suplementarios para Amazon Linux (o SPAL)?	205
Ventajas	206
Support de paquetes SPAL	206
Informar de problemas relacionados con los paquetes	206
Temas relacionados	207
Configure SPAL en el 023 AL2	207
Requisitos previos	208
Comprobación de los requisitos previos	208
Instalar SPAL en su sistema	208
Instalación de paquetes SPAL	210
Desinstalar el repositorio SPAL del sistema	211
Temas relacionados	211
Ejecución de aplicaciones	212
Control de recursos con systemd	212
Control de recursos con systemd-run para ejecutar comandos únicos	212
Control de recursos en un servicio systemd	215
Uso de las utilidades de cgroups	220
Uso AL2023 en AWS	222
Empezando con AWS	222
Inscríbase en una Cuenta de AWS	222
Creación de un usuario con acceso administrativo	223
Concesión de acceso mediante programación	224
AL2023 en Amazon EC2	226
Lanzamiento de AL2023 con la consola de Amazon EC2	227

Lanzamiento AL2023 mediante el parámetro SSM y AWS CLI	228
Lanzamiento de la AL2023 AMI más reciente mediante CloudFormation	229
Lanzamiento AL2023 mediante un ID de AMI específico	231
AL2023 Obsolescencia y ciclo de vida de la AMI	231
Conexión a instancias AL2023	232
Comparación entre AL2023 estándar (predeterminado) y mínimo AMIs	232
AL2023 en contenedores	260
AL2023 Imagen del contenedor base	261
AL2023 Imagen de contenedor mínima	263
Creación de imágenes de contenedores básicas AL2023	265
AL2023 Comparación de listas de paquetes de imágenes de contenedores	269
AL2023 AMI mínima en comparación con las imágenes de contenedores	275
AL2023 en Elastic Beanstalk	292
CloudShell en AL2023	293
AL2023 para hosts de contenedores de Amazon ECS	293
Cambios relevantes en Amazon ECS desde AL2	293
AMI personalizada optimizada para Amazon ECS	294
Amazon EFS en AL2023	295
amazon-efs-utils	295
Montaje de un sistema de archivos de Amazon EFS	296
Amazon EMR en AL2023	296
AL2023 versiones basadas en Amazon EMR	296
AL2023 basado en Amazon EMR en EKS	296
AL2023 en AWS Lambda	296
Tiempo de ejecución de provided.al2023 Lambda	297
AL2023 tiempos de ejecución basados	297
Tutoriales	298
Instale LAMP en AL2023	298
Paso 1: Preparar el servidor LAMP	299
Paso 2: Probar el servidor LAMP	304
Paso 3: Proteger el servidor de base de datos	306
Paso 4: Instalación (opcional) phpMyAdmin	307
Solución de problemas	310
Temas relacionados	311
Configurar SSL/TLS en AL2023	312
Requisitos previos	313

Paso 1: Habilitar TLS en el servidor	314
Paso 2: Obtener un certificado firmado por una CA	317
Paso 3: Probar y reforzar la configuración de seguridad	325
Solución de problemas	329
Aloja un WordPress blog en AL2023	330
Requisitos previos	331
Instala WordPress	332
Siguientes pasos	342
Ayuda Ha cambiado el nombre DNS público y ahora el blog se ha roto	343
Transición de Redis 6 a Valkey en 023 AL2	345
Cronología de soporte para Redis 6	345
Introducción a Valkey	345
Plan de migración y calendario	345
Opciones y pasos de migración	345
Temas relacionados	349
Instale GNOME en AL2023	349
Requisitos previos	349
Instalación	350
Temas relacionados	350
Configure VNC en AL2023	350
Requisitos previos	351
Paso 1: instalación	351
Paso 2: configuración	351
Paso 3: conectarse mediante un cliente de VNC	353
(Opcional) Inicie el servicio al arrancar	354
(Opcional) Desactive la pantalla de bloqueo inactiva	354
Temas relacionados	355
Uso de LRU multigeneración (MGLRU) en los núcleos AL2023	355
Configuración y ajuste	355
Instalar LibreOffice en AL2023	357
Requisitos previos	357
Paso 1: Determine la arquitectura del sistema	358
Paso 2: Descargar el paquete LibreOffice RPM	358
Paso 3: Extrae el archivo	359
Paso 4: Instale los paquetes RPM	359
Paso 5: Verificar la instalación	359

Lanzamiento LibreOffice	360
Resolución de problemas	360
AL2023 fuera de Amazon EC2	362
Descargue las imágenes de la máquina virtual de AL2023	362
Configuraciones admitidas	362
Requisitos de KVM	363
Requisitos de VMware	365
Requisitos de Hyper-V	367
Configuración de una máquina virtual AL2023	370
Configuración basada en NoCloud <code>seed.iso</code>	371
Configuración basada en VMware <code>guestinfo</code>	374
Comparación de la lista de paquetes de AL2023 para la imagen KVM y la AMI estándar	376
Comparación de la lista de paquetes de AL2023 para la imagen AMI estándar y la imagen OVA de VMware	401
Comparación de la lista de paquetes de AL2023 para la imagen Hyper-V y la AMI estándar	427
Identificación de versiones de Amazon Linux	453
<code>/etc/os-release</code>	453
Diferencias clave	454
Tipos de campos:	454
<code>/etc/os-release</code> Ejemplos de	456
Comparación con otras distribuciones	457
Archivos específicos de Amazon Linux	459
<code>/etc/system-release</code>	460
<code>/etc/image-id</code>	460
Ejemplo específicos de Amazon Linux	461
Código de ejemplo	463
Diseño del sistema de archivos	476
<code>/</code>	476
<code>/boot</code>	477
<code>/boot/efi</code>	477
<code>/etc</code>	477
<code>/home</code>	477
<code>/root</code>	478
<code>/srv</code>	478
<code>/tmp</code>	479
<code>/run</code>	480

/usr	480
/usr/bin	481
/usr/include	481
/usr/lib y /usr/lib64	481
/usr/local	481
/usr/share	482
/var	482
/var/cache	482
/var/lib	482
/var/log	482
/var/spool	483
/var/tmp	483
Actualización AL2023	485
Prácticas recomendadas para implementar actualizaciones de forma segura	485
Preparación para actualizaciones menores	489
Preparación para actualizaciones importantes	489
Recibir notificaciones sobre nuevas actualizaciones	490
Actualizaciones deterministas mediante repositorios versionados	491
Control de las actualizaciones recibidas de las versiones principales y secundarias	491
Diferencias entre las actualizaciones de versión principales y secundarias	492
Saber cuándo están disponibles las actualizaciones	492
Controle las actualizaciones de paquetes disponibles en los AL2023 repositorios	493
Reemplazo de instancias	493
Actualizaciones deterministas locales	494
Administración de actualizaciones	502
Comprobación de las actualizaciones de los paquetes disponibles	503
Aplicación de actualizaciones de seguridad mediante DNF y las versiones de repositorio	508
Reinicio automático del servicio tras las actualizaciones (de seguridad)	521
¿Cuándo es necesario reiniciar para aplicar las actualizaciones de seguridad?	522
Lanzamiento de una instancia con la última versión del repositorio habilitada	523
Obtener información de soporte del paquete	523
dnf check-release-update	524
Inclusión, habilitación y deshabilitación de nuevos repositorios	528
Inclusión de repositorios con cloud-init	531
Kernel Live Patching	532
Limitaciones	533

Configuraciones admitidas y requisitos previos	533
Usar Kernel Live Patching	534
Actualizaciones del kernel	539
Versiones del núcleo de Linux en AL2023	539
AL2023 Actualización a una versión más reciente del núcleo	540
AL2023 kernels: preguntas frecuentes	545
Lenguajes de programación y tiempos de ejecución	547
C/C++ y Fortran	547
GCC 14	548
Comparación de versiones de lenguajes estándar	549
.NET	551
Migración a versiones nuevas de .NET	551
Instalación de .NET	551
Go	552
AL2023 Función Lambda: Go	552
Java	553
Node.js	41
Perl	555
Módulos Perl	555
PHP	555
Migración a versiones nuevas de PHP	555
Migración desde PHP 7.x	556
Módulos PHP	556
Python	556
Migración a versiones nuevas de Python	557
Módulos Python	557
Ruby	557
Rust	559
AL2023 Función Lambda: Rust	560
Swift	560
AL2023 función: Swift	561
TypeScript	561
Usuarios y grupos reservados de AL2023	566
Lista de usuarios reservados de AL2023	566
Lista de grupos reservados de AL2023	574
Códecs disponibles en 023 AL2	587

Seguridad y conformidad	590
Avisos de seguridad	591
Anuncios en ALAS	591
POR DESGRACIA FAQs	592
Avisos de ALAS	592
Avisos y repositorios de RPM	592
Asesoramiento IDs	593
Fecha de publicación del aviso y fecha de actualización del aviso	593
Tipos de avisos	594
Gravedad de los avisos	594
Avisos y paquetes	595
Avisos y CVEs	596
Texto de aviso	596
Avisos de parches activos del kernel	597
Esquema updateinfo.xml	598
Lista de avisos aplicables	599
Actualizaciones in situ	602
Aplicación de las actualizaciones mencionadas en un aviso	603
Configuración de los modos de SELinux para AL2023	606
Estado y modos predeterminados de SELinux para AL2023	607
Cambiar al modo enforcing	608
Opción para deshabilitar SELinux	609
Habilitar el modo FIPS en AL2023	611
Habilitación del modo FIPS en un contenedor de AL2023	612
Intercambio de los proveedores de FIPS de OpenSSL en AL2023	614
Fortalecimiento del kernel	616
Opciones de fortalecimiento del kernel (independientes de la arquitectura).	616
Opciones de fortalecimiento del kernel específicas para x86-64	634
Opciones de refuerzo del kernel específicas de aarch64	637
Firma de metadatos del repositorio	639
Cómo funciona la firma de metadatos del repositorio	640
Diferencia entre gpgcheck y repo_gpgcheck	640
Habilitar la verificación de los metadatos del repositorio	641
Comprobar que la firma de metadatos del repositorio funciona	641
Claves públicas GPG para repositorios AL2023	642
UEFI Secure Boot está activado AL2023	643

Active el arranque seguro de UEFI AL2023 643

Inscripción de una instancia existente 644

Registrar imagen de una instantánea 644

Actualizaciones de revocación 645

Cómo funciona UEFI Secure Boot en AL2023 645

Inscribir sus propias claves 646

..... dcxlvii

¿Qué es Amazon Linux 2023?

Amazon Linux 2023 (AL2023) es la próxima generación de Amazon Linux de Amazon Web Services (AWS). Con AL2023, puede desarrollar y ejecutar aplicaciones empresariales en la nube en un entorno de tiempo de ejecución seguro, estable y de alto rendimiento. Además, obtiene un entorno de aplicaciones que ofrece soporte a largo plazo con acceso a las últimas innovaciones en Linux. AL2023 se ofrece sin cargo adicional.

AL2023 es el sucesor de Amazon Linux 2 (AL2). Para obtener información sobre las diferencias entre AL2023 y AL2, consulte [Comparando AL2 y AL2 023](#) y [Cambios del paquete en AL2023](#).

Temas

- [Liberar cadencia](#)
- [Denominación y control de versiones](#)
- [Optimizaciones operativas y de rendimiento](#)
- [Relación con Fedora](#)
- [Personalizar cloud-init](#)
- [Actualizaciones y características de seguridad](#)
- [Servicio de redes](#)
- [Paquetes principales de cadenas de herramientas glibc, gcc, binutils](#)
- [Herramienta de administración de paquetes](#)
- [Configuración predeterminada del servidor SSH](#)

Liberar cadencia

Amazon Linux 2023 (AL2023) se lanzó en marzo de 2023 y tendrá soporte hasta el 30 de junio de 2029. El soporte consta de dos fases:

- Soporte estándar: durante esta fase, la versión recibe actualizaciones menores trimestrales. La fase de soporte estándar finaliza el 30 de junio de 2027.
- Mantenimiento: durante esta fase, la versión solo recibe actualizaciones de seguridad y correcciones de errores críticos. Estas actualizaciones se publican en cuanto están disponibles. La fase de mantenimiento finaliza el 30 de junio de 2029.

Versiones principales y secundarias

Con cada versión de Amazon Linux (versión principal, versión secundaria o versión de seguridad), publicamos una nueva Imagen de máquina de Amazon (AMI) para Linux.

- **Lanzamiento de la versión principal:** incluye nuevas características y mejoras en la seguridad y el rendimiento en todas las versiones. Las mejoras pueden incluir cambios importantes en kernel, la cadena de herramientas, Glib C, OpenSSL y cualquier otra biblioteca y utilidad del sistema. Las principales versiones de Amazon Linux se basan en parte en la versión actual de la distribución de upstream de Fedora Linux. AWS podría añadir o reemplazar paquetes específicos de otras versiones anteriores ajenas a Fedora.
- **Lanzamiento de la versión secundaria:** actualización trimestral que incluye actualizaciones de seguridad, correcciones de errores, además de características y paquetes nuevos. Cada versión secundaria es una lista acumulativa de actualizaciones que incluye correcciones de errores y de seguridad, además de nuevas características y paquetes. Estas versiones pueden incluir los últimos tiempos de ejecución de los lenguajes, como PHP. También pueden incluir otros paquetes de software populares, como Ansible y Docker.

Consumir nuevas versiones

Las actualizaciones se proporcionan mediante una combinación de versiones nuevas de Imagen de máquina de Amazon (AMI) y los correspondientes repositorios nuevos. De forma predeterminada, una AMI nueva y el repositorio al que apunta están acoplados. Sin embargo, puede apuntar las instancias de Amazon EC2 en ejecución a versiones de repositorio más recientes a lo largo del tiempo para aplicar actualizaciones a las instancias en ejecución. También puede actualizar lanzando nuevas instancias de las AMI más recientes.

Política de soporte a largo plazo

Amazon Linux proporciona actualizaciones para todos sus paquetes y mantiene la compatibilidad dentro de una versión principal para las aplicaciones creadas en Amazon Linux. Los paquetes principales, como la biblioteca glibc, OpenSSL, OpenSSH y el administrador de paquetes DNF, reciben soporte durante la vigencia de la versión principal de AL2023. Los paquetes que no forman parte de los paquetes principales se admiten en función de sus orígenes de upstream específicos. Para ver el estado y las fechas de soporte específicos de los paquetes individuales, ejecute el siguiente comando.


```
$ sudo dnf supportinfo --pkg packagename
```

Para obtener información sobre todos los paquetes instalados actualmente, ejecute el siguiente comando.

```
$ sudo dnf supportinfo --show installed
```

La lista completa de los paquetes principales se completa durante la vista previa. Si desea ver más paquetes incluidos como paquetes principales, póngase en contacto con nosotros. Evaluamos a medida que recopilamos comentarios. Puede enviar comentarios sobre AL2023 a través de su representante designado de AWS o archivando un problema en el repositorio de [amazon-linux-2023](https://github.com/amazon-linux-2023) en GitHub.

Denominación y control de versiones

AL2023 ofrece una versión secundaria cada tres meses durante los dos años de soporte estándar. Cada versión se identifica mediante un incremento de 0 a N. El 0 se refiere a la versión principal original de esa iteración. Todas las versiones se denominarán Amazon Linux 2023. Cuando se lance la próxima versión de Amazon Linux, AL2023 contará con soporte ampliado y recibirá actualizaciones de seguridad y correcciones de errores críticos.

Por ejemplo, las versiones secundarias de AL2023 tienen el siguiente formato:

- 2023.0.20230301
- 2023.1.20230601
- 2023.2.20230901

Las AMI de AL2023 correspondientes tienen el siguiente formato:

- al2023-ami-2023.0.20230301.0-kernel-6.1-x86_64
- al2023-ami-2023.1.20230601.0-kernel-6.1-x86_64
- al2023-ami-2023.2.20230901.0-kernel-6.1-x86_64

Dentro de una versión secundaria específica, las versiones normales de la AMI se lanzan con una marca de tiempo de la fecha del lanzamiento de la AMI.

- al2023-ami-2023.0.**20230301**.0-kernel-6.1-x86_64

- `al2023-ami-2023.0.20230410.0-kernel-6.1-x86_64`
- `al2023-ami-2023.0.20230520.0-kernel-6.1-x86_64`

El método recomendado para identificar una instancia de AL2 o AL2023 comienza con la lectura de la cadena de enumeración de plataforma común (CPE) de `/etc/system-release-cpe`. A continuación, divida la cadena en sus campos. Por último, lea los valores de la plataforma y la versión.

AL2023 también presenta nuevos archivos para la identificación de la plataforma:

- `/etc/amazon-linux-release` enlaces simbólicos a `/etc/system-release`
- `/etc/amazon-linux-release-cpe` enlaces simbólicos a `/etc/system-release-cpe`

Estos dos archivos indican que una instancia es Amazon Linux. No es necesario leer un archivo ni dividir la cadena en campos, a menos que desee conocer los valores específicos de la plataforma y la versión.

Optimizaciones operativas y de rendimiento

Kernel 6.1 de Amazon Linux

- AL2023 utiliza los controladores más recientes para los dispositivos Elastic Network Adapter (ENA) y Elastic Fabric Adapter (EFA). AL2023 se centra en los backports de rendimiento y funcionalidad para el hardware de la infraestructura de Amazon EC2.
- Kernel Live Patching está disponible para los tipos de instancia `x86_64` y `aarch64`. Esto reduce la necesidad de reiniciarlos con frecuencia.
- Todas las configuraciones en tiempo de ejecución y compilación del kernel incluyen muchas de las mismas optimizaciones operativas y de rendimiento que AL2.

Selección de la cadena de herramientas básica y marcadores de compilación predeterminados

- Los paquetes de AL2023 se crean con las optimizaciones del compilador (`-O2`) habilitadas de forma predeterminada
- Los paquetes de AL2023 que se crean requieren `x86-64v2` para sistemas `x86-64` (`-march=x86-64-v2`) y Graviton 2 o superior para `aarch64` (`-march=armv8.2-a+crypto -mtune=neoverse-n1`).

- Los paquetes de AL2023 se crean con la vectorización automática habilitada (`-ftree-vectorize`).
- Los paquetes de AL2023 se crean con la optimización del tiempo de enlace (LTO) habilitada.
- AL2023 usa las versiones actualizadas de Rust, Clang/LLVM y Go.

Selección de paquetes y control de versiones

- Los backports seleccionados para los principales componentes del sistema incluyen varias mejoras de rendimiento para ejecutarse en la infraestructura de Amazon EC2, especialmente en instancias de Graviton.
- AL2023 está integrado con varias características y Servicios de AWS. Esto incluye la AWS CLI, SSM Agent, Amazon Kinesis Agent y CloudFormation.
- AL2023 utiliza Amazon Corretto como kit de desarrollo de Java (JDK).
- AL2023 proporciona actualizaciones en tiempo de ejecución de los motores de bases de datos y los lenguajes de programación para las versiones más recientes a medida que se van lanzando en los proyectos upstream. Los tiempos de ejecución de los lenguajes de programación con nuevas versiones se añaden cuando se realiza el lanzamiento.

Implementación en un entorno de nube

- La AMI base de AL2023 y las imágenes del contenedor se actualizan con frecuencia para permitir el reemplazo de instancias de revisiones.
- Las actualizaciones del kernel se incluyen en las actualizaciones de la AMI de AL2023. Esto significa que no necesita utilizar comandos como `yum update` y `reboot` para actualizar el kernel.
- Además de la AMI estándar de AL2023, también está disponible una AMI mínima y una imagen de contenedor. Elija la AMI mínima para ejecutar un entorno con la cantidad mínima de paquetes necesarios para ejecutar el servicio.
- De forma predeterminada, las AMI y los contenedores de AL2023 se encuentran bloqueados en una versión específica de los repositorios de paquetes. No hay actualización automática cuando se lanzan. Esto significa que usted siempre tiene el control de cuándo incorpora cualquier actualización de paquete. Siempre puede realizar las pruebas en un entorno beta / gamma antes de lanzarlas a producción. Si hay algún problema, puede utilizar la ruta de restauración validada previamente.

Relación con Fedora

AL2023 mantiene sus propios ciclos de vida de lanzamiento y soporte independientes de Fedora. AL2023 proporciona versiones actualizadas de software de código abierto, una mayor variedad de paquetes y lanzamientos frecuentes. Esto preserva los conocidos sistemas operativos basados en RPM.

La versión generalmente disponible (GA) de AL2023 no se puede comparar directamente con ninguna versión específica de Fedora. La versión GA de AL2023 incluye componentes de Fedora 34, 35 y 36. Algunos de los componentes son los mismos que los componentes de Fedora y otros están modificados. Otros componentes se parecen más a los componentes de CentOS Stream 9 o se desarrollaron de forma independiente. El kernel de Amazon Linux proviene de las opciones de soporte a largo plazo que se encuentran en kernel.org, elegidas independientemente de Fedora.

Personalizar cloud-init

El paquete cloud-init es una aplicación de código abierto que arranca imágenes de Linux en un entorno de computación en la nube. Para obtener más información, consulte la [documentación de cloud-init](#).

AL2023 contiene una versión personalizada de cloud-init. Con cloud-init, puede especificar lo que ocurre en la instancia en el momento del arranque.

Al lanzar una instancia, puede utilizar campos de datos de usuario a los que pasar acciones a cloud-init. De este modo, puede utilizar las imágenes de máquina de Amazon (AMI) habituales para numerosos casos de uso y configurarlas dinámicamente cuando inicia una instancia. AL2023 también utiliza cloud-init para configurar la cuenta `ec2-user`.

AL2023 usa las acciones cloud-init en `/etc/cloud/cloud.cfg.d` y `/etc/cloud/cloud.cfg`. Puede crear sus propios archivos de acción de cloud-init en el directorio `/etc/cloud/cloud.cfg.d`. Cloud-init lee todos los archivos de este directorio en orden lexicográfico. Los archivos recientes sobrescriben los valores en los archivos anteriores. Cuando cloud-init lanza una instancia, el paquete cloud-init realiza las siguientes tareas de configuración:

- Ajusta la configuración local predeterminada
- Ajusta el nombre de host
- Analiza y gestiona los datos de usuario
- Genera claves de SSH privadas de host

- Agrega claves SSH públicas del usuario a `.ssh/authorized_keys` para facilitar el inicio de sesión y la administración
- Prepara los repositorios para la administración de paquetes
- Controla acciones de paquetes definidas en los datos de usuario
- Ejecuta scripts de usuario que están en los datos de usuario
- Monta volúmenes de almacén de instancias si es preciso
 - De forma predeterminada, si el volumen de almacén de instancias `ephemeral0` está presente y contiene un sistema de archivos válido, el volumen del almacén de instancias se monta en `/media/ephemeral0`. De lo contrario, no está montado.
 - De forma predeterminada, para los tipos de instancias `m1.small` y `c1.medium`, cualquier volumen de intercambio asociado a la instancia se monta.
 - Puede anular el montaje del volumen de almacén de instancias predeterminado con la siguiente directiva de cloud-init:

```
#cloud-config
mounts:
- [ ephemeral0 ]
```

Para obtener más control sobre los montajes, consulte [Mounts](#) en la documentación de cloud-init.

- Cuando se lanza una instancia, los volúmenes del almacén de instancias que admiten TRIM no se formatean. Antes de poder montar los volúmenes del almacén de instancias, debe particionar y formatear los volúmenes del almacén de instancias.

Para obtener más información, consulte [Soporte TRIM del volumen de almacén de instancias](#) en la Guía del usuario de Amazon EC2.

- Al lanzar las instancias, puede utilizar el módulo `disk_setup` para dividir y formatear los volúmenes de almacén de instancias.

Para obtener más información, consulte [Disk Setup](#) en la documentación de cloud-init.

Para obtener más información sobre el uso de cloud-init con SELinux, consulte [Use cloud-init para habilitar el modo enforcing](#).

Para obtener información sobre formatos cloud-init de datos de usuario, consulte [User-Data Formats](#) en la documentación de cloud-init.

Actualizaciones y características de seguridad

AL2023 ofrece numerosas actualizaciones y soluciones de seguridad.

Temas

- [Actualizaciones administradas](#)
- [Seguridad en la nube](#)
- [modos SELinux](#)
- [Programas de conformidad](#)
- [Servidor SSH predeterminado](#)
- [Características principales de OpenSSL 3](#)

Actualizaciones administradas

Aplicar actualizaciones de seguridad mediante DNF y las versiones de repositorio. Para obtener más información, consulte [Gestione las actualizaciones de paquetes y sistemas operativos en AL2023](#).

Seguridad en la nube

La seguridad es una responsabilidad compartida entre AWS y usted. En el [modelo de responsabilidad compartida](#), se habla de seguridad de la nube y seguridad en la nube: Para obtener más información, consulte [Seguridad y conformidad en Amazon Linux 2023](#).

modos SELinux

De forma predeterminada, SELinux está activado y configurado en modo permisivo en AL2023. En el modo permisivo, las denegaciones de permisos se registran pero no se aplican.

Las políticas de SELinux definen los permisos para los usuarios, los procesos, los programas, los archivos y los dispositivos. Con SELinux, puede elegir una de dos políticas. Las políticas son de seguridad segmentada o multinivel (MLS).

Para obtener más información sobre los modos y la política de SELinux, consulte [Configuración de los modos de SELinux para AL2023](#) y [la wiki del proyecto SELinux](#).

Programas de conformidad

Audidores independientes evalúan la seguridad y la conformidad de AL2023 junto con muchos programas de conformidad de AWS.

Servidor SSH predeterminado

AL2023 incluye OpenSSH 8.7. OpenSSH 8.7 deshabilita de forma predeterminada el algoritmo de intercambio de claves `ssh-rsa`. Para obtener más información, consulte [Configuración predeterminada del servidor SSH](#).

Características principales de OpenSSL 3

- El protocolo de administración de certificados (CMP, RFC 4210) incluye tanto el CRMF (RFC 4211) como la transferencia HTTP (RFC 6712).
- Un cliente HTTP o HTTPS de libcrypto admite las acciones GET y POST, redirección, contenido simple y codificado en ASN.1, proxies y tiempos de espera.
- EVP_KDF trabaja con funciones de derivación clave.
- EVP_MAC API trabaja con MACs.
- Soporte TLS del kernel de Linux.

Para obtener más información, consulte la [guía de migración de OpenSSL](#).

Servicio de redes

El proyecto de código abierto `systemd-networkd` está ampliamente disponible en las distribuciones modernas de Linux. El proyecto utiliza un lenguaje de configuración declarativo similar al resto del marco `systemd`. Sus tipos de archivos de configuración principales son los archivos `.network` y `.link`.

El paquete `amazon-ec2-net-utils` genera configuraciones específicas de la interfaz en el directorio `/run/systemd/network`. Estas configuraciones habilitan redes IPv4 e IPv6 en las interfaces cuando están conectadas a una instancia. Estas configuraciones también instalan reglas de enrutamiento de políticas que ayudan a garantizar que el tráfico de origen local se enrute a la red a través de la interfaz de red de la instancia correspondiente. Estas reglas garantizan que el

tráfico correcto se enrute a través de la interfaz de red elástica (ENI) desde las direcciones o prefijos asociados. Para obtener más información sobre el uso de la ENI, consulte [Uso de ENI](#) en la Guía del usuario de Amazon EC2.

Puede personalizar este comportamiento de red mediante la ubicación de un archivo de configuración personalizado en el directorio `/etc/systemd/network` para anular los ajustes de configuración predeterminados que contiene `/run/systemd/network`.

La documentación de [systemd.network](#) describe cómo el servicio `systemd-networkd` determina la configuración que se aplica a una interfaz específica. También genera nombres alternativos, conocidos como altnames, para las interfaces respaldadas por la ENI a fin de reflejar las propiedades de varios recursos de AWS. Estas propiedades de la interfaz respaldadas por la ENI son el campo `ENI ID` y el campo `DeviceIndex` del adjunto de la ENI. Puede hacer referencia a estas interfaces mediante sus propiedades cuando utilice diversas herramientas, como el comando `ip`.

Los nombres de las interfaces de las instancias de AL2023 se generan mediante el esquema de nombres de ranuras `systemd`. Para obtener más información, consulte [systemd.net naming scheme](#).

Además, AL2023 utiliza de forma predeterminada el algoritmo de programación de transmisiones de la red de gestión de cola activa `fq_codel`. Para obtener más información, consulte [Información general de CoDel](#).

Paquetes principales de cadenas de herramientas glibc, gcc, binutils

Un subconjunto de paquetes de Amazon Linux se designa como paquetes principales de la cadena de herramientas. Como parte principal AL2023, los paquetes principales reciben cinco años de soporte. Es posible que cambiemos la versión de un paquete, pero el soporte a largo plazo se aplica al paquete incluido en la versión de Amazon Linux.

Estos tres paquetes principales proporcionan una cadena de herramientas del sistema que se utiliza para crear la mayoría del software de distribución de Amazon Linux.

Package	Definición	Finalidad
glibc 2.34	Biblioteca del sistema C	Utilizada por la mayoría de los programas binarios que proporcionan funciones

Package	Definición	Finalidad
		estándar y por la interfaz entre los programas y el kernel.
gcc 11.2	Conjunto de compiladores gcc	Compila C, C++, Fortran.
binutils 2.35	Ensamblador y vinculador, además de otras herramientas binarias	Manipula o inspecciona programas binarios.

Recomendamos que las actualizaciones de cualquier biblioteca glibc van seguidas por un reinicio. Para actualizar paquetes que controlan servicios, puede ser suficiente con reiniciar los servicios y elegir las actualizaciones. Sin embargo, un reinicio del sistema garantiza que se hayan completado todas las actualizaciones anteriores de los paquetes y bibliotecas.

Herramienta de administración de paquetes

La herramienta de administración de paquetes de software predeterminada en AL2023 es DNF. DNF es la sucesora de YUM, la herramienta de administración de paquetes de AL2.

DNF es similar YUM en su uso. Muchos comandos de DNF son iguales y tienen las mismas opciones que los comandos de YUM. En una interfaz de la línea de comandos (CLI), en la mayoría de los casos `dnf` reemplaza a `yum`.

Por ejemplo, para los siguientes comandos `yum` de AL2:

```
$ sudo yum install packagename
$ sudo yum search packagename
$ sudo yum remove packagename
```

En AL2023, se convierten en los siguientes comandos:

```
$ sudo dnf install packagename
$ sudo dnf search packagename
$ sudo dnf remove packagename
```

En AL2023, el comando `yum` sigue disponible, pero como un puntero del comando `dnf`. Por lo tanto, cuando el comando `yum` se usa en el intérprete de comandos o en un script, todos los comandos y

opciones son los mismos que los de DNF CLI. Para obtener más información sobre las diferencias entre YUM CLI y DNF CLI, consulte [Cambios en DNF CLI en comparación con YUM](#).

Para obtener una referencia completa de los comandos y las opciones del comando `dnf`, consulte la página `man dnf` del manual. Para obtener más información, consulte [Referencia de los comandos DNF](#)

Configuración predeterminada del servidor SSH

Si tiene clientes SSH de hace varios años, es posible que vea un error al conectarse a una instancia. Si el error indica que no se ha encontrado un tipo de clave de host coincidente, actualice su clave de host SSH para solucionar este problema.

Desactivación predeterminada de las firmas **ssh-rsa**

AL2023 incluye una configuración predeterminada que deshabilita el algoritmo de clave de host heredado `ssh-rsa` y genera un conjunto reducido de claves de host. Los clientes deben admitir el algoritmo de clave de host `ssh-ed25519` o el algoritmo de clave de host `ecdsa-sha2-nistp256`.

La configuración predeterminada acepta cualquiera de estos algoritmos de intercambio de claves:

- `curve25519-sha256`
- `curve25519-sha256@libssh.org`
- `ecdh-sha2-nistp256`
- `ecdh-sha2-nistp384`
- `ecdh-sha2-nistp521`
- `diffie-hellman-group-exchange-sha256`
- `diffie-hellman-group14-sha256`
- `diffie-hellman-group16-sha512`
- `diffie-hellman-group18-sha512`

De forma predeterminada, AL2023 genera las claves de host `ed25519` y `ECDSA`. Los clientes admiten el algoritmo de clave de host `ssh-ed25519` o el algoritmo de clave de host `ecdsa-sha2-nistp256`. Cuando se conecta mediante SSH a una instancia, debe usar un cliente que admita un algoritmo compatible, como `ssh-ed25519` o `ecdsa-sha2-nistp256`. Si necesita usar otros tipos de claves, anule la lista de claves generadas con un fragmento `cloud-config` en los datos de usuario.

En el siguiente ejemplo, `cloud-config` genera una clave de host `rsa` con las claves `ecdsa` y `ed25519`.

```
#cloud-config
ssh_genkeytypes:
- ed25519
- ecdsa
- rsa
```

Si utiliza un par de claves RSA para la autenticación de clave pública, su cliente SSH debe admitir una firma `rsa-sha2-256` o `rsa-sha2-512`. Si utiliza un cliente incompatible y no puede actualizarlo, vuelva a habilitar la compatibilidad `ssh-rsa` en su instancia. Para volver a habilitar el soporte de `ssh-rsa`, active la política de cifrado LEGACY del sistema mediante los siguientes comandos.

```
$ sudo dnf install crypto-policies-scripts
$ sudo update-crypto-policies --set LEGACY
```

Para obtener más información sobre la administración de las claves de host, consulte las [claves de host de Amazon Linux](#)

Funcionalidad obsoleta en AL2023

La funcionalidad obsoleta en AL2 y no presente en AL2023 se documenta aquí. Se trata de funciones como las características y los paquetes que están presentes en AL2, pero no en AL2023 y no se añadirán a AL2023. Para obtener más información sobre durante cuánto tiempo se admite la funcionalidad en AL2, consulte [Funcionalidad obsoleta en AL2](#).

En AL2023 también hay una característica obsoleta y se eliminará en la próxima versión. En este capítulo se describe en qué consiste esa funcionalidad, cuándo dejará de ser compatible y cuándo se eliminará de Amazon Linux. Comprender la funcionalidad obsoleta le ayudará a implementar AL2023 y a prepararse para la próxima versión principal de Amazon Linux.

Temas

- [Paquetes de compat-](#)
- [Funcionalidad obsoleta discontinuada en AL1 y eliminada en AL2](#)
- [Funcionalidad obsoleta AL2 y eliminada en AL2023](#)
- [Obsoleta en AL2023](#)

Paquetes de **compat-**

Todos los paquetes de AL2 con el prefijo de `compat-` se proporcionan por motivos de compatibilidad binaria con binarios antiguos que aún no se han reconstruido para las versiones modernas del paquete. Cada nueva versión principal de Amazon Linux no transferirá ningún paquete `compat-` de versiones anteriores.

Todos los paquetes `compat-` de una versión de Amazon Linux (por ejemplo, AL2) están obsoletos y no están presentes en la versión posterior (por ejemplo, AL2023). Recomendamos encarecidamente que el software se reconstruya con las versiones actualizadas de las bibliotecas.

Funcionalidad obsoleta discontinuada en AL1 y eliminada en AL2

En esta sección se describen las funciones que están disponibles en AL1 y que ya no están disponibles en AL2.

 Note

Como parte de la fase de soporte de mantenimiento de AL1, algunos paquetes tenían una fecha de fin de vida útil (EOL) anterior a la EOL de AL1. Para obtener más información, consulte las [declaraciones de soporte de paquetes de AL1](#).

 Note

Algunas funciones de AL1 se suspendieron en versiones anteriores. Para obtener información, consulte las [notas de la versión de AL1](#).

Temas

- [AMI x86 \(i686\) de 32 bits](#)
- [aws-apitools-* se ha sustituido por AWS CLI](#)
- [systemd reemplaza a upstart en AL2](#)

AMI x86 (i686) de 32 bits

Como parte de la [versión 2014.09 de AL1](#), se anunció que sería la última versión en producir las AMI de 32 bits. Por lo tanto, a partir de la [versión 2015.03 de AL1](#), Amazon Linux dejó de admitir la ejecución del sistema en modo de 32 bits. AL2 ofrece soporte de tiempo de ejecución limitado para binarios de 32 bits en hosts x86-64 y no proporciona paquetes de desarrollo que permitan crear nuevos binarios de 32 bits. AL2023 ya no incluye ningún paquete de espacio de usuario de 32 bits. Recomendamos a los usuarios que completen su transición al código de 64 bits antes de migrar a AL2023.

Si necesita ejecutar binarios de 32 bits en AL2023, es posible utilizar el espacio de usuario de 32 bits de AL2 dentro de un contenedor de AL2 que se ejecute sobre AL2023.

aws-apitools-* se ha sustituido por AWS CLI

Antes del lanzamiento de AWS CLI en septiembre de 2013, AWS creó un conjunto de utilidades de línea de comandos, implementadas en Java, que permitían a los usuarios realizar llamadas a la API de Amazon EC2. Estas herramientas se suspendieron en 2015 y AWS CLI se convirtió en la forma

preferida de interactuar con las API de Amazon EC2 desde la línea de comandos. El conjunto de utilidades de línea de comandos incluye los siguientes paquetes `aws-apitools-*`.

- `aws-apitools-as`
- `aws-apitools-cfn`
- `aws-apitools-common`
- `aws-apitools-ec2`
- `aws-apitools-elb`
- `aws-apitools-mon`

El soporte inicial para los paquetes `aws-apitools-*` finalizó en marzo de 2017. A pesar de la falta de soporte previo, Amazon Linux siguió ofreciendo algunas de estas utilidades de línea de comandos, por ejemplo `aws-apitools-ec2`, para proporcionar compatibilidad con versiones anteriores a los usuarios. AWS CLI es una herramienta más sólida y completa que los paquetes `aws-apitools-*`, ya que se mantiene activamente y proporciona un medio para utilizar todas las API de AWS.

Los paquetes `aws-apitools-*` quedaron obsoletos en marzo de 2017 y no recibirán más actualizaciones. Todos los usuarios de cualquiera de estos paquetes deberían migrar a AWS CLI lo antes posible. Estos paquetes no están presentes en AL2023.

AL1 también proporcionó los paquetes `aws-apitools-iam` y `aws-apitools-rds`, que quedaron obsoletos en AL1 y no están presentes en Amazon Linux a partir de AL2.

systemd reemplaza a upstart en AL2

AL2 fue la primera versión de Amazon Linux en utilizar el sistema init `systemd`, sustituyendo a `upstart` en AL1. Cualquier configuración específica de `upstart` debe cambiarse como parte de la migración de AL1 a una versión más reciente de Amazon Linux. No se puede usar `systemd` en AL1, por lo que pasar de `upstart` a `systemd` solo se puede hacer como parte del cambio a una versión principal más reciente de Amazon Linux, como AL2 o AL2023.

Funcionalidad obsoleta AL2 y eliminada en AL2023

En esta sección se describen las funciones que están disponibles y que ya no están disponibles en AL2. AL2023

Temas

- [Paquetes x86 \(i686\) de 32 bits](#)
- [aws-apitools-*reemplazado por AWS CLI](#)
- [awslogsobsoleto en favor del agente unificado de Amazon CloudWatch Logs](#)
- [Sistema de control de revisiones bzd](#)
- [cgroup v1](#)
- [Hotpatch de log4j \(log4j-cve-2021-44228-hotpatch\)](#)
- [lsb_release y el paquete system-lsb-core](#)
- [mccrypt](#)
- [OpenJDK 7 \(java-1.7.0-openjdk\)](#)
- [Python 2.7](#)
- [rsyslog-opensslreemplaza rsyslog-gnutls](#)
- [Servicio de información de red \(NIS\)/yp](#)
- [Varios nombres de dominio en Amazon VPC create-dhcp-options](#)
- [Sun RPC en glibc](#)
- [Huella digital clave de OpenSSH en el registro audit](#)
- [Vinculador de ld.gold](#)
- [ping6](#)
- [Paquete ftp](#)

Paquetes x86 (i686) de 32 bits

Como parte de la [versión 2014.09 de AL1](#), anunciamos que sería la última versión en producir 32 bits. AMIs Por lo tanto, a partir de la [versión 2015.03 de](#), AL1 Amazon Linux ya no admite la ejecución del sistema en modo de 32 bits. AL2 proporciona soporte de tiempo de ejecución limitado para binarios de 32 bits en hosts x86-64 y no proporciona paquetes de desarrollo que permitan crear nuevos binarios de 32 bits. AL2023 ya no incluye ningún paquete de espacio de usuario de 32 bits. Recomendamos a los clientes que lleven a cabo la transición al código de 64 bits.

Si necesita ejecutar archivos binarios de 32 bits AL2023, puede utilizar el espacio de usuario de 32 bits desde el AL2 interior de un AL2 contenedor que se ejecute sobre él. AL2023

aws-apitools-*reemplazado por AWS CLI

Antes de su lanzamiento AWS CLI en septiembre de 2013, puso a AWS disposición un conjunto de utilidades de línea de comandos, implementadas en Java, que permitían a los clientes realizar llamadas a la API de Amazon EC2. Estas herramientas quedaron obsoletas en 2015 y AWS CLI se convirtieron en la forma preferida de interactuar con Amazon APIs EC2 desde la línea de comandos. Esto incluye los siguientes paquetes `aws-apitools-*`.

- `aws-apitools-as`
- `aws-apitools-cfn`
- `aws-apitools-common`
- `aws-apitools-ec2`
- `aws-apitools-elb`
- `aws-apitools-mon`

El soporte inicial para los paquetes `aws-apitools-*` finalizó en marzo de 2017. A pesar de la falta de soporte previo, Amazon Linux siguió ofreciendo algunas de estas utilidades de línea de comandos, por ejemplo `aws-apitools-ec2`, para proporcionar compatibilidad con versiones anteriores a los clientes. AWS CLI Es una herramienta más sólida y completa que los `aws-apitools-*` paquetes, ya que se mantiene activamente y proporciona una forma de utilizarla toda. AWS APIs

Los paquetes `aws-apitools-*` quedaron obsoletos en marzo de 2017 y no recibirán más actualizaciones. Todos los usuarios de cualquiera de estos paquetes deberían migrar a ellos lo antes AWS CLI posible. Estos paquetes no están presentes en AL2023.

awslogsobsoleto en favor del agente unificado de Amazon CloudWatch Logs

El [awslogs](#) paquete está obsoleto en AL2 y ya no está presente en AL2023. Se sustituye por el [agente CloudWatch Logs unificado](#), disponible en el `amazon-cloudwatch-agent` paquete. Para obtener más información, consulta la [Guía del usuario CloudWatch de Amazon Logs](#).

Sistema de control de revisiones **bzr**

El sistema de control de revisiones [GNU Bazaar](#) (`bzr`) está descontinuado AL2 y ya no está presente en AL2023.

Se recomienda a los usuarios de `bzr` migrar sus repositorios a `git`.

cgroup v1

AL2023 pasa a la jerarquía del Grupo de Control Unificado (cgroup v2), mientras que AL2 usa cgroup v1. Como AL2 no es compatible con cgroup v2, esta migración debe completarse como parte del traslado a. AL2023

Hotpatch de log4j (**log4j-cve-2021-44228-hotpatch**)

Note

El `log4j-cve-2021-44228-hotpatch` paquete está en desuso AL2 y se eliminó en. AL2023

En respuesta al error [CVE-2021-44228](#), Amazon Linux lanzó una versión empaquetada en RPM del [Hotpatch para Apache Log4j](#) para y. AL1 AL2 En el [anuncio de la adición del hotpatch a Amazon Linux](#), señalamos que “la instalación del hotpatch no reemplaza la actualización a una versión de log4j que mitigue los errores CVE-2021-44228 o CVE-2021-45046”.

El hotpatch era una medida de mitigación para dar tiempo a aplicar el parche log4j. La primera versión de disponibilidad general AL2023 se publicó 15 meses después de la versión [CVE-2021-44228](#), por lo que no se incluye con el hotpatch (si está activado o no). AL2023

Se recomienda a los clientes que utilicen sus propias versiones de log4j en Amazon Linux que se aseguren de actualizarlas a versiones que no estén afectadas por los códigos [CVE-2021-44228](#) o [CVE-2021-45046](#).

lsb_release y el paquete system-lsb-core

Históricamente, algunos programas utilizaban el `lsb_release` comando (incluido en AL2 el `system-lsb-core` paquete) para obtener información sobre la distribución de Linux en la que se estaba ejecutando. La base de estándares de Linux, Linux Standards Base (LSB), introdujo este comando y las distribuciones de Linux lo adoptaron. Las distribuciones de Linux han evolucionado para utilizar el estándar más simple para almacenar esta información en `/etc/os-release` y otros archivos relacionados.

El estándar de `os-release` proviene de `systemd`. Para obtener más información, consulte la [documentación de systemd os-release](#).

AL2023 no viene con el `lsb_release` comando y no incluye el `system-lsb-core` paquete. El software debe completar la transición al estándar de `os-release` para mantener la compatibilidad con Amazon Linux y otras distribuciones principales de Linux.

mcrypt

La `mcrypt` biblioteca y la PHP extensión asociada quedaron obsoletas en AL2 y ya no están presentes en AL2023.

La versión inicial de PHP [dejó de utilizar la extensión mcrypt en PHP 7.1](#), que se publicó por primera vez en diciembre de 2016 y su versión final en octubre de 2019.

La [última vez que se publicó la mcrypt biblioteca original fue en 2007](#), y no ha realizado la migración desde el control de cvs revisiones [SourceForge necesaria para las nuevas confirmaciones en 2017](#). La más reciente (y solo la correspondiente a tres años antes) data de 2011, y no se menciona que el proyecto tenga un responsable.

Se recomienda a los demás usuarios de que `mcrypt` transfieran su código a OpenSSL, ya que no `mcrypt` se añadirá más. AL2023

OpenJDK 7 (**java-1.7.0-openjdk**)

Note

AL2023 proporciona varias versiones de [Amazon Corretto para admitir cargas](#) de trabajo Java basadas. Los paquetes de OpenJDK 7 están en desuso y ya no están presentes AL2 en. AL2023 El JDK más antiguo disponible AL2023 es el de Corretto 8.

Para obtener más información sobre Java en Amazon Linux, consulte [Java en AL2023](#).

Python 2.7

Note

AL2023 eliminó Python 2.7, por lo que cualquier componente del sistema operativo que requiera Python está escrito para funcionar con Python 3. Para seguir utilizando una versión de Python proporcionada y compatible con Amazon Linux, convierta el código de Python 2 a Python 3.

Para obtener más información sobre Python en Amazon Linux, consulte [Pythonen AL2023](#).

rsyslog-opensslreemplaza rsyslog-gnutls

El `rsyslog-gnutls` paquete está obsoleto en AL2 y ya no está presente en AL2023. El paquete `rsyslog-openssl` debe ser un sustituto directo para cualquier uso del paquete `rsyslog-gnutls`.

Servicio de información de red (NIS)/yp

El Servicio de Información de Red (NIS), originalmente denominado Páginas Amarillas o YP está en desuso en AL2 y ya no está presente en AL2023. Esto incluye los siguientes paquetes: `ybind`, `ypserv` y `yp-tools`. A otros paquetes que se integran con esta funcionalidad se NIS ha eliminado. AL2023

Varios nombres de dominio en Amazon VPC `create-dhcp-options`

En Amazon Linux 2, era posible pasar varios nombres de dominio en el parámetro `domain-name` a [create-dhcp-options](#), lo que daba como resultado que `/etc/resolv.conf` contuviera algo parecido a `search foo.example.com bar.example.com`. El servidor DHCP de Amazon VPC envía la lista de nombres de dominio proporcionados mediante la opción DHCP 15, que solo admite un nombre de dominio único (consulte la [sección 3.17 de RFC 2132](#)). Dado que se AL2023 utiliza `systemd-networkd` para la configuración de redRFC, que sigue a la, esta función accidental no AL2 está presente en AL2023

La [AWS CLI](#) y la [documentación de Amazon VPC](#) dice lo siguiente: “Algunos sistemas operativos Linux aceptan varios nombres de dominio separados por espacios. Sin embargo, Windows y otros sistemas operativos de Linux tratan el valor como un dominio único, lo que da lugar a un comportamiento inesperado. Si el conjunto de opciones de DHCP está asociado a una VPC que tiene instancias en las que se ejecutan los sistemas operativos que tratan el valor como un dominio único, especifique solo un nombre de dominio”.

En estos sistemas, por ejemplo AL2023, al especificar dos dominios mediante la DHCP opción 15 (que solo permite uno), y dado que el [carácter de espacio no es válido en los nombres de dominio](#), esto provocará que el carácter de espacio se codifique como `032`, por lo `/etc/resolv.conf` que contendrá `search foo.exmple.com032bar.example.com`.

Para admitir varios nombres de dominio, un servidor DHCP debe usar la opción DHCP 119 (consulte la [sección 2 de RFC 3397](#)). Consulte la [Guía del usuario de Amazon VPC](#) para saber si el servidor DHCP de Amazon VPC lo admite.

Sun RPC en **glibc**

La implementación de Sun RPC in `glibc` está en desuso y se eliminó en. AL2 AL2023 Se recomienda a los clientes que pasen a utilizar la `libtirpc` biblioteca (disponible en AL2 y AL2023) si se requiere alguna Sun RPC funcionalidad. La adopción de `libtirpc` también permite que las aplicaciones admitan IPv6.

Este cambio refleja la adopción por parte de la comunidad en general de la eliminación de esta funcionalidad en `glibc` inicial, por ejemplo, la [eliminación de las interfaces Sun RPC de `glibc` en Fedora](#) y un [cambio similar en Gentoo](#).

Huella digital clave de OpenSSH en el registro **audit**

Más adelante, en el ciclo de vida de AL2, se agregó un parche al paquete OpenSSH para emitir la huella digital de la clave utilizada para autenticar. Esta funcionalidad no está presente en. AL2023

Vinculador de **ld.gold**

El `ld.gold` enlazador está disponible en AL2 y se elimina en AL2023. Los clientes que creen software que haga referencia explícita al vinculador de gold deben migrar al vinculador normal (`ld.bfd`).

Las [notas de la versión 2.44](#) de [GNU Binutils](#) (publicadas en febrero de 2025) documentan la eliminación de `ld.gold`: “A diferencia de lo que veníamos haciendo hasta ahora, en esta versión el archivo `tar binutils-2.44.tar` no contiene el origen del vinculador gold. Esto se debe a que el vinculador gold ahora está obsoleto y finalmente se eliminará, a menos que se presenten voluntarios que se ofrezcan a continuar con su desarrollo y mantenimiento”.

ping6

En AL2023, la `ping` utilidad normal es compatible IPv6 de forma nativa y la separada ya no `/bin/ping6` es necesaria. En AL2023, `/usr/sbin/ping6` es un enlace simbólico al `/usr/bin/ping` ejecutable.

Este cambio se produce tras la adopción por parte de la comunidad en general de `iputils` versiones más recientes que proporcionan esta funcionalidad, por ejemplo, el [IPv6 cambio de ping en Fedora](#).

Paquete **ftp**

El paquete `ftp` en AL2 ya no está disponible en Amazon Linux a partir de AL2023. Esta decisión se tomó como parte de nuestro compromiso continuo con la seguridad, la facilidad de mantenimiento y las prácticas modernas de desarrollo de software. Como parte (o antes) de la migración a AL2023, recomendamos migrar cualquier uso del paquete `ftp` heredado a una de sus alternativas.

Introducción

El paquete `ftp` heredado no se ha mantenido de forma activa desde hace muchos años. La última actualización importante del código fuente se produjo a principios de la década de 2000 y el repositorio fuente original ya no está disponible. Si bien algunas distribuciones de Linux incluyen parches para solucionar vulnerabilidades de seguridad, la base de código permanece prácticamente sin mantenimiento.

Alternativas recomendadas

AL2023 ofrece varias alternativas modernas y de mantenimiento activo para la funcionalidad de FTP:

`lftp` (disponible en AL2 y AL2023)

Un sofisticado programa de transferencia de archivos compatible con FTP, HTTP, SFTP y otros protocolos. Ofrece más funciones que el cliente `ftp` tradicional y se mantiene activamente.

Instalar con: `dnf install lftp`

`curl` (disponible en AL2 y AL2023)

Una herramienta de línea de comandos versátil para transferir datos con direcciones URL, compatible con FTP, FTPS, HTTP, HTTPS y muchos otros protocolos.

Disponible de forma predeterminada en AL2023 a través del paquete `curl-minimal`. Para obtener un soporte de protocolo más amplio, puede actualizarlo opcionalmente a `curl-full` con `dnf swap curl-minimal curl-full`.

`wget` (disponible en AL2 y AL2023)

Utilidad de línea de comandos no interactiva para descargar archivos de la web, compatible con los protocolos HTTP, HTTPS y FTP.

Instalar con: `dnf install wget` (no se instala de forma predeterminada en todas las imágenes de AL2023)

sftp (disponible en AL2 y AL2023)

Un protocolo seguro de transferencia de archivos que funciona a través de SSH y proporciona transferencias de archivos cifradas.

Disponible de forma predeterminada como parte del paquete OpenSSH.

Consideraciones sobre la migración

Si sus aplicaciones o scripts dependen del cliente ftp anterior, considere los siguientes enfoques de migración:

1. Actualice los scripts para usar alternativas modernas: modifique los scripts para usar `lftp`, `curl`, `wget` o `sftp` en lugar del cliente ftp heredado.
2. Revise las dependencias de los paquetes: algunas aplicaciones pueden incluir el paquete ftp como una dependencia en los metadatos de sus paquetes, aunque hace tiempo que migraron al uso interno de protocolos modernos. En estos casos, la aplicación puede funcionar correctamente en AL2023 a pesar de que `/usr/bin/ftp` no esté incluida en el paquete ftp. Revise los requisitos reales de su solicitud en lugar de basarse únicamente en las dependencias establecidas.
3. Actualice las dependencias de las aplicaciones: en el caso de las aplicaciones que mantiene y que aún declaran una dependencia del paquete ftp pero que en realidad no la utilizan, actualice los metadatos del paquete para eliminar esta dependencia innecesaria.

Consideraciones de seguridad

El protocolo FTP transmite los datos, incluidas las credenciales de autenticación, en texto plano. Para las aplicaciones sensibles a la seguridad, recomendamos encarecidamente utilizar alternativas cifradas, como SFTP o HTTPS, que son compatibles con las herramientas alternativas recomendadas.

Obsoleta en AL2023

En esta sección se describe la funcionalidad que existe en AL2023 y que es probable que se elimine en una futura versión de Amazon Linux. En cada sección se describe cuál es la funcionalidad y cuándo se espera que se elimine de Amazon Linux.

 Note

Esta sección se irá actualizando a medida que el ecosistema Linux evolucione y en el futuro se acerquen las versiones principales de Amazon Linux.

Temas

- [Compatibilidad con tiempo de ejecución de 32 bits x86 \(i686\)](#)
- [aspell](#)
- [Berkeley DB \(libdb\)](#)
- [cron](#)
- [IMDSv1](#)
- [pcre versión 1](#)
- [System V init \(sysvinit\)](#)
- [Los paquetes EOL están obsoletos](#)

Compatibilidad con tiempo de ejecución de 32 bits x86 (i686)

AL2023 conserva la capacidad de ejecutar binarios x86 (i686) de 32 bits. Es probable que la próxima versión principal de Amazon Linux ya no admita la ejecución de binarios de espacio de usuario de 32 bits.

aspell

Si bien AL2023 viene con el paquete `aspell`, está obsoleto y se eliminará en la próxima versión principal de Amazon Linux. Se recomienda a los clientes migrar a reemplazos modernos, como `hunspell` o `enchant2`.

La retirada de `aspell` en AL2023 sigue el cambio general de la comunidad, por ejemplo, la retirada de [aspell en Fedora](#).

Berkeley DB (**libdb**)

AL2023 viene con la versión 5.3.28 de la biblioteca Berkeley DB (`libdb`). Esta es la última versión de Berkeley DB antes de que la licencia cambiara a la licencia GNU Affero GPLv3 (AGPL), en lugar de la licencia Sleepycat, menos restrictiva.

Hay pocos paquetes en AL2023 que sigan dependiendo de Berkeley DB (`libdb`), y la biblioteca se eliminará en la próxima versión principal de Amazon Linux.

 Note

El administrador de paquetes `dnf` de AL2023 conserva el soporte de solo lectura para una base de datos `rpm` en formato Berkeley DB (BDB). Este soporte se eliminará en la próxima versión principal de Amazon Linux.

La retirada de `libdb` sigue el cambio general de la comunidad, por ejemplo, la retirada de [libdb en Fedora](#).

cron

El paquete `cronie` se instaló de forma predeterminada en la AMI de AL2, lo que proporciona soporte para la forma `crontab` tradicional de programar tareas periódicas. En AL2023, `cronie` no se incluye de forma predeterminada. Por lo tanto, ya no se ofrece soporte para `crontab` de forma predeterminada.

En AL2023, puede instalar el paquete `cronie` para utilizar los trabajos clásicos `cron`. Se recomienda migrar a temporizadores `systemd` debido a la funcionalidad adicional que proporcionan mediante `systemd`.

Es posible que una futura versión de Amazon Linux, posiblemente la próxima versión principal, deje de incluir soporte para trabajos clásicos `cron` y complete la transición a temporizadores `systemd`. Se recomienda que deje de usar `cron`.

IMDSv1

De forma predeterminada, las AMI de AL2023 están configuradas para iniciarse solo en modo IMDSv2, lo que deshabilita el uso de IMDSv1. Todavía existe la opción de usar AL2023 con iMDSv1 habilitado. Es probable que una futura versión de Amazon Linux solo aplique IMDSv2.

Para obtener más información sobre la configuración del IMDS para las AMI, consulte [Configuración de las AMI](#) en la Guía del usuario de Amazon EC2.

pcre versión 1

El paquete `pcre` heredado está obsoleto y se eliminará en la próxima versión principal de Amazon Linux. El paquete `pcre2` es el sucesor. Si bien las primeras versiones de AL2023 se enviaron con un número limitado de paquetes adicionales `pcre`, estos paquetes se migrarán a `pcre2` en AL2023. La biblioteca `pcre` obsoleta seguirá disponible en AL2023.

Note

La versión obsoleta de `pcre` no recibirá actualizaciones de seguridad durante toda la vida útil de AL2023. Para obtener más información sobre el ciclo de vida del soporte de `pcre` y el tiempo durante el que el paquete recibirá las actualizaciones de seguridad, consulte las [declaraciones de soporte del paquete pcre](#).

La retirada de `pcre` en favor de `pcre2` sigue el cambio general de la comunidad en esta dirección, por ejemplo, la [retirada de pcre en Fedora](#).

System V init (**sysvinit**)

Aunque AL2023 mantiene la compatibilidad con los scripts de servicio (`init`) de System V, el proyecto inicial `systemd`, como parte de su [versión v254](#), anunció que [dejará de admitir los scripts de servicio de System V](#) e indicó que dicha compatibilidad se eliminará en una futura versión de `systemd`. Para obtener más información, consulte [systemd](#).

AL2023 mantendrá la compatibilidad con versiones anteriores de los scripts de servicio (`init`) de System V, pero se recomienda a los usuarios que opten por utilizar archivos de unidades `systemd` nativos a fin de estar preparados para cuando se elimine la compatibilidad con los scripts de servicio (`init`) de System V de Amazon Linux, probablemente en la próxima versión principal.

Los paquetes EOL están obsoletos

Cada paquete disponible en AL2023 tiene una [declaración de soporte](#) asociada que cubre información específica de Amazon Linux. Estas declaraciones cubren el núcleo del sistema operativo y su vida útil, así como paquetes como [the section called “PHP”](#) y [the section called “Python”](#), en los que AL2023 incluye varias versiones y cada una de ellas es compatible durante el tiempo que dura el proyecto original de código abierto.

En AL2023, puede obtener información de soporte de paquetes utilizando el administrador de paquetes `dnf`. Para obtener más información, consulte [Obtener información de soporte del paquete](#).

Si un paquete deja de ser compatible antes del final de la versión principal de Amazon Linux, se debe suponer que este paquete está obsoleto y no estará presente en la próxima versión principal de Amazon Linux.

En el caso de paquetes como [the section called “PHP”](#) y [the section called “Python”](#), en los que cada versión principal de Amazon Linux incluye varias versiones, cada una con un ciclo de vida de soporte diferente, es probable que sigan presentes en las nuevas versiones principales de Amazon Linux, aunque con poca o ninguna superposición de las versiones principales de los paquetes. Se recomienda tener en cuenta los plazos de soporte de los paquetes de Amazon Linux al seleccionar las dependencias.

Comparando AL2 y AL2 023

En los temas siguientes se describen las principales diferencias entre la 023 AL2 y la AL2 023.

Para obtener más información sobre las funciones obsoletas en AL1, y AL2 023 AL2, consulte.

[Funcionalidad obsoleta en AL2023](#)

Temas

- [Paquetes añadidos, actualizados y eliminados](#)
- [Soporte para cada versión](#)
- [Cambios de nombres y control de versiones](#)
- [Optimizaciones](#)
- [Proviene de múltiples fuentes](#)
- [Servicio del sistema de redes](#)
- [Administrador de paquetes](#)
- [Uso de cloud-init](#)
- [Soporte de escritorio gráfico](#)
- [Triplete del compilador](#)
- [Paquetes x86 \(i686\) de 32 bits](#)
- [lsb_release y el paquete system-lsb-core](#)
- [Extra Packages for Enterprise Linux \(EPEL\)](#)
- [Python 2.7 ha sido reemplazado por Python 3](#)
- [Actualizaciones de seguridad](#)
- [Actualizaciones deterministas para mayor estabilidad](#)
- [gp3 como tipo de volumen de Amazon EBS predeterminado](#)
- [Jerarquía de grupos de control unificados \(cgroup v2\)](#)
- [Los temporizadores de systemd reemplazan a cron](#)
- [Cadena de herramientas mejorada: gcc, binutils y glibc](#)
- [El diario systemd reemplaza a rsyslog](#)
- [Dependencias del paquete minimizadas](#)

- [Amazon Corretto como JVM predeterminada](#)
- [AWS CLIv2](#)
- [UEFI Preferred y arranque seguro](#)
- [Cambios en la configuración predeterminada del servidor SSH](#)
- [AL2023 el kernel cambia de AL2](#)
- [/tmp ahora es tmpfs](#)
- [Cambios en la AMI y en la imagen del contenedor](#)
- [Comparación de paquetes instalados en las AMI de Amazon Linux 2 y Amazon Linux 2023.](#)
- [Comparación de paquetes instalados en las AMI de Amazon Linux 2 y Amazon Linux 2023](#)
- [Comparación de paquetes instalados en imágenes de contenedor base de Amazon Linux 2 y Amazon Linux 2023](#)

Paquetes añadidos, actualizados y eliminados

AL2EI 023 contiene miles de paquetes de software disponibles para su uso. Para obtener una lista completa de todos los paquetes añadidos, actualizados o eliminados en la versión AL2 023 en comparación con las versiones anteriores de Amazon Linux, consulte [Cambios de paquetes en la versión AL2 023](#).

Para solicitar que se añada o modifique un paquete en la versión AL2 023, publique un problema en el repositorio de [amazon-linux-2023](#) en GitHub.

Soporte para cada versión

Para AL2 023, ofrecemos cinco años de soporte.

Para obtener más información, consulte [Liberar cadencia](#).

Cambios de nombres y control de versiones

AL2EI 023 admite los mismos mecanismos que AL2 admite la identificación de plataformas. AL2EI 023 también presenta nuevos archivos para la identificación de plataformas.

Para obtener más información, consulte [Denominación y control de versiones](#).

Optimizaciones

AL2023 optimiza el tiempo de arranque para reducir el tiempo que transcurre desde el lanzamiento de la instancia hasta la ejecución de la carga de trabajo del cliente. Estas optimizaciones abarcan la configuración, las `cloud-init` configuraciones y las funciones del kernel de la EC2 instancia de Amazon integradas en los paquetes del sistema operativo, como `kmod` y `systemd`.

Para obtener más información sobre optimizaciones, consulte [Optimizaciones operativas y de rendimiento](#).

Proviene de múltiples fuentes

AL2023 está basado en RPM e incluye componentes procedentes de múltiples versiones de Fedora y otras distribuciones, como CentOS 9 Stream. El kernel de Amazon Linux proviene de las versiones de soporte a largo plazo (LTS) directamente de `kernel.org`, elegidas independientemente de otras distribuciones.

Para obtener más información, consulte [Relación con Fedora](#).

Servicio del sistema de redes

El servicio del `systemd-networkd` sistema administra las interfaces de red en 023. AL2 Se trata de un cambio con respecto a AL2, que utiliza ISC `dhclient` `odhcpd`.

Para obtener más información, consulte [Servicio de redes](#).

Administrador de paquetes

La herramienta de administración de paquetes de software predeterminada en el AL2 023 es DNF. DNF es la sucesora de YUM la herramienta de gestión de paquetes incluida en AL2.

Para obtener más información, consulte [Herramienta de administración de paquetes](#).

Uso de cloud-init

En AL2 023, `cloud-init` administra el repositorio de paquetes. De forma predeterminada, en las versiones anteriores de Amazon Linux, `cloud-init` instalaba actualizaciones de seguridad. Este no es el valor predeterminado de AL2 023. Las nuevas funciones de actualización deterministas para la actualización `releasever` en el momento del lanzamiento describen la forma AL2 023

de habilitar las actualizaciones de paquetes en el momento del lanzamiento. Para obtener más información, consulte [Gestione las actualizaciones de paquetes y sistemas operativos en AL2023](#) y [Actualizaciones deterministas para mayor estabilidad](#).

Con el AL2 023, se puede utilizar con. cloud-init SELinux Para obtener más información, consulte [Use cloud-init para habilitar el modo enforcing](#).

Cloud-init carga el contenido de la configuración con cloud-init desde ubicaciones remotas mediante HTTP(S). En versiones anteriores, Amazon Linux no notificaba si los recursos remotos no estaban disponibles. En el AL2 023, los recursos remotos no disponibles crean un error grave y no se puede ejecutar. cloud-init Este cambio en el comportamiento de AL2, proporciona un comportamiento predeterminado más seguro de «cierre por error».

Para obtener más información, consulte [Personalizar cloud-init](#) y [Documentación de cloud-init](#).

Soporte de escritorio gráfico

AL2La versión 023 incluye un entorno gráfico de escritorio basado en GNOME a partir de la versión 2023.7, que sustituye al escritorio MATE utilizado en él. AL2 Esta versión ofrece a los usuarios una experiencia de escritorio diferente y, al mismo tiempo, mantiene AL2 el rendimiento optimizado para la nube de la 023. El entorno de escritorio GNOME ofrece varias opciones de personalización, funciones de integración del sistema y un diseño de interfaz de usuario distinto, lo que proporciona a los usuarios una alternativa al entorno de escritorio MATE anterior. Para obtener más información, consulte la página [AL2Escritorio gráfico 023](#).

Triplete del compilador

AL2023 establece el triplete del compilador para indicar que es el GCC proveedorLLVM. amazon

Por lo tanto, pasa a ser 023 AL2 aarch64-redhat-linux-gcc. aarch64-amazon-linux-gcc
AL2

Esto debería ser completamente transparente para la mayoría de los usuarios y solo podría afectar a aquellos que estén creando compiladores en AL2 023.

Paquetes x86 (i686) de 32 bits

Como parte de la [versión 2014-09 AL1](#), se anunció que sería la última versión en producir 32 bits. AMIs Por lo tanto, a partir de la [versión 2015.03 de](#), AL1 Amazon Linux dejó de admitir la ejecución

del sistema en modo de 32 bits. AL2 ofrecía un soporte de tiempo de ejecución limitado para binarios de 32 bits en hosts x86-64 y no proporcionaba paquetes de desarrollo que permitieran crear nuevos binarios de 32 bits. AL2023 ya no incluye ningún paquete de espacio de usuario de 32 bits. Le recomendamos que lleve a cabo la transición al código de 64 bits.

Si necesita ejecutar binarios de 32 bits en el AL2 023, es posible utilizar el espacio de usuario de 32 bits desde el AL2 interior de un contenedor que se ejecute sobre el 023. AL2 AL2

lsb_release y el paquete system-lsb-core

Históricamente, algunos programas utilizaban el `lsb_release` comando (incluido en AL2 el `system-lsb-core` paquete) para obtener información sobre la distribución de Linux en la que se estaba ejecutando. La base de estándares de Linux, Linux Standards Base (LSB), introdujo este comando y las distribuciones de Linux lo adoptaron. Las distribuciones de Linux han evolucionado para utilizar el estándar más simple para almacenar esta información en `/etc/os-release` y otros archivos relacionados.

El estándar de `os-release` proviene de `systemd`. Para obtener más información, consulte la [documentación de systemd os-release](#).

AL2023 no viene con el `lsb_release` comando y no incluye el `system-lsb-core` paquete. El software debe completar la transición al estándar de `os-release` para mantener la compatibilidad con Amazon Linux y otras distribuciones principales de Linux.

Extra Packages for Enterprise Linux (EPEL)

Warning

El AL2 `epel` Extra habilitó el repositorio de terceros EPEL7. A partir del 30 de junio de 2024, el repositorio de terceros EPEL7 ya no se mantendrá.

Este repositorio de terceros no tendrá actualizaciones en el futuro. Esto significa que no habrá correcciones de seguridad para los paquetes del repositorio de EPEL.

En esta sección se tratarán las opciones de la versión AL2 023 para los paquetes que se encuentran en EPEL.

Extra Packages for Enterprise Linux (EPEL) es un proyecto de la comunidad Fedora cuyo objetivo es crear una amplia gama de paquetes para sistemas operativos Linux de nivel empresarial. El

proyecto ha producido RHEL y CentOS empaquetado principalmente. AL2 presenta un alto nivel de compatibilidad con CentOS 7. Como resultado, muchos EPEL7 paquetes funcionan correctamente en AL2.

No hay EPEL versiones que sean compatibles en formato binario con el AL2 023. Sin embargo, los clientes que deseen utilizar sus EPEL7 paquetes en el AL2 023 tienen varias opciones. Algunos EPEL paquetes tienen alternativas en el AL2 023, mientras que otros se proporcionan como parte de él. [Paquetes complementarios para Amazon Linux](#)

Warning

Agregue únicamente los repositorios diseñados para usarse con AL2 023.

Si bien los repositorios diseñados para otras distribuciones pueden funcionar hoy en día, no hay garantía de que continúen haciéndolo con cualquier actualización de paquete en AL2 023 o con el repositorio que no esté diseñado para usarse con 023. AL2

Esta página proporciona información sobre los EPEL7 paquetes que utilizan los clientes de la 023 AL2 y sus homólogas. AL2

Para el resto de los paquetes, es posible que los clientes puedan utilizar los paquetes suplementarios para Amazon Linux (SPAL). SPAL ofrece miles de EPEL9 paquetes, creados específicamente para Amazon Linux 2023, pero estos paquetes no están cubiertos por AWS Support Plans. Esto significa que CVEs los paquetes de SPAL no están siendo rastreados y los parches solo se proporcionan cuando están disponibles desde el principio.

Important

Consulte la documentación [Paquetes complementarios para Amazon Linux](#) antes de usarlo.

Temas

- [axel- HTTP/FTP cliente](#)
- [brotli y libbrotli: compresión](#)
- [collectd: daemon de recopilación de estadísticas](#)
- [cpulimit: limitador de uso de la CPU](#)
- [exim: agente de transferencia de correo](#)

- [fuse3: sistema de archivos en el espacio de usuario \(FUSE\) v3](#)
- [ganglia: sistema de monitoreo distribuido](#)
- [git-lfs: control de versiones de archivos grandes con Git](#)
- [haveged: una fuente de entropía que utiliza el algoritmo HAVEGE](#)
- [inotify-tools: herramientas de línea de comando de inotify](#)
- [iperf- Punto de referencia TCP/UDP de rendimiento](#)
- [jemalloc: implementación alternativa de malloc](#)
- [libbsd: biblioteca de funciones compatible con BSD](#)
- [libserf: biblioteca de cliente de HTTP](#)
- [libzstd: biblioteca de compresión zstd](#)
- [Servidor web lighttpd](#)
- [lshell: un intérprete de comandos restringido](#)
- [monit: monitor de procesos, archivos, directorios y dispositivos](#)
- [nodejs](#)
- [perl-Config-General](#)
- [python2-lockfile: bloqueo de archivos](#)
- [python2-rsa: Python RSA puro](#)
- [python2-simplejson: rutinas JSON para Python 2](#)
- [rkhunter: Rootkit Hunter](#)
- [rssh: un intérprete de comandos restringido para usar con OpenSSH](#)
- [sscg: generador de certificados SSL autofirmado](#)
- [stress: prueba de esfuerzo](#)
- [stress-ng: prueba de esfuerzo](#)
- [tmpwatch: elimina los archivos en función de la hora del último acceso](#)
- [xmlstarlet: utilidades XML de línea de comandos](#)

axel- HTTP/FTP cliente

El paquete `axel` estaba en EPEL7 y nunca se envió como parte de Amazon Linux. Las alternativas disponibles en AL2 023 son `curl` y `wget`.

⚠ Warning

La opción `-S` para `axel` usa una conexión `http` no cifrada para detectar réplicas de un archivo.

Se recomienda encarecidamente migrar cualquier uso de `axel` a `curl` o `wget`.

brotli y libbrotli: compresión

Los `libbrotli` paquetes `brotli` y `libbrotli` estaban disponibles en EPEL7, mientras que solo el `brotli` paquete estaba disponible en el AL2 núcleo.

`brotli` Tanto el paquete como el `libbrotli` paquete están incluidos en el AL2 023.

El `brotli` paquete se puede instalar en el AL2 023 con el siguiente comando:

```
[ec2-user ~]$ sudo dnf install brotli
```

El `libbrotli` paquete se puede instalar en el AL2 023 con el siguiente comando:

```
[ec2-user ~]$ sudo dnf install libbrotli
```

collectd: daemon de recopilación de estadísticas

El `collect` paquete estaba disponible y también estaba disponible en los `collectd-python3` AL2 Extras `collectd` y Extras. EPEL7

El `collectd` paquete está incluido en AL2 023 y se puede instalar ejecutando el siguiente comando:

```
[ec2-user ~]$ sudo dnf install collectd
```

cpulimit: limitador de uso de la CPU

En Amazon Linux 2023, `systemd` proporciona una funcionalidad para limitar el uso de la CPU de los procesos o grupos de procesos. Esta funcionalidad también es fácil de usar para cualquier servicio de `systemd`.

systemd proporciona potentes funciones de control de recursos que pueden utilizarse para garantizar que cualquier tarea o grupo de tareas tenga un límite en cuanto a los recursos que puede consumir. Para obtener más información, consulte la documentación sobre [systemd.resource-control](#) de la versión inicial, junto con [Limite el uso de recursos de procesos en AL2023 mediante systemd](#).

exim: agente de transferencia de correo

El exim paquete estaba incluido EPEL7 y anteriormente estaba disponible en AL1. Amazon Linux 2023 proporciona tanto los agentes de transferencia de postfix sendmail correo como los agentes de transferencia de correo (MTAs).

fuse3: sistema de archivos en el espacio de usuario (FUSE) v3

El paquete fuse3 (incluidos fuse3-libs y fuse3-devel) estaba disponible en EPEL7. Estos paquetes forman parte de AL2 023 y cada uno de ellos se puede instalar ejecutando el siguiente comando correspondiente:

```
[ec2-user ~]$ sudo dnf install fuse3
```

```
[ec2-user ~]$ sudo dnf install fuse3-libs
```

```
[ec2-user ~]$ sudo dnf install fuse3-devel
```

ganglia: sistema de monitoreo distribuido

El ganglia paquete estaba y anteriormente estaba disponible en AL1. EPEL7 No se envió con AL2.

El proyecto inicial tuvo un período de inactividad en el que no se CVEs abordaron algunas cuestiones pendientes. Si bien recientemente se han registrado actividades en el proyecto upstream, no está previsto ampliarlas al 023. ganglia AL2

git-lfs: control de versiones de archivos grandes con Git

El paquete git-lfs estaba disponible en EPEL7. En Amazon Linux 2023, el paquete git-lfs se incluye en el repositorio principal. En el AL2 023, se git-lfs puede instalar ejecutando el siguiente comando:

```
[ec2-user ~]$ sudo dnf install git-lfs
```

haveged: una fuente de entropía que utiliza el algoritmo HAVEGE

El paquete haveged estaba disponible en EPEL7. Amazon Linux 2023 viene preconfigurado con fuentes de entropía, por lo que no requiere el uso de haveged.

inotify-tools: herramientas de línea de comando de inotify

El inotify-tools paquete estaba en el EPEL7 AL2 023 y está incluido en él.

Note

En AL2 023, systemd admite la activación basada en rutas, que se puede utilizar para tomar medidas en caso de eventos, como cuando una ruta existe o cambia.

Gran parte de los usos de inotify-tools ahora se pueden llevar a cabo de manera más fiable utilizando la activación de rutas de systemd. Para obtener más información, consulte [systemd.path](#).

El inotify-tools paquete está incluido en AL2 023 y se puede instalar ejecutando el siguiente comando:

```
[ec2-user ~]$ sudo dnf install inotify-tools
```

iperf- Punto de referencia TCP/UDP de rendimiento

El paquete de la iperf versión 2 estaba disponible y también estaba disponible en la testing AL2 Extra. Y también estaba disponible en EPEL7 AL1

Note

El paquete iperf3 también está disponible, y ofrece la versión 3 de iperf.

El iperf paquete está incluido en la AL2 versión 023 y se puede instalar ejecutando el siguiente comando:

```
[ec2-user ~]$ sudo dnf install iperf
```

jemalloc: implementación alternativa de malloc

El jemalloc paquete estaba incluido EPEL7 y estaba disponible en lamp-mariadb10.2-php7.2 y mariadb10.5 AL2 Extras.

El jemalloc paquete está incluido en AL2 023 y se puede instalar ejecutando el siguiente comando:

```
[ec2-user ~]$ sudo dnf install jemalloc
```

libbsd: biblioteca de funciones compatible con BSD

El libbsd paquete estaba incluido EPEL7 y también estaba disponible en el testing AL2 Extra.

El libbsd paquete está incluido en la AL2 versión 023 y se puede instalar ejecutando el siguiente comando:

```
[ec2-user ~]$ sudo dnf install libbsd
```

Los archivos de desarrollo de libbsd se pueden instalar ejecutando el siguiente comando.

```
[ec2-user ~]$ sudo dnf install libbsd-devel
```

libserf: biblioteca de cliente de HTTP

El paquete libserf estaba disponible en EPEL7. El paquete libserf se proporciona en Amazon Linux 2023. Se puede instalar ejecutando el siguiente comando:

```
[ec2-user ~]$ sudo dnf install libserf
```

libzstd: biblioteca de compresión zstd

El libzstd paquete estaba tanto en el AL2 núcleo como en EPEL7. El libzstd paquete también forma parte del AL2 023.

```
[ec2-user ~]$ sudo dnf install libzstd
```

Servidor web **lighttpd**

El `lighttpd` paquete estaba disponible y anteriormente estaba disponible en AL1. EPEL7 Amazon Linux 2023 proporciona servidores web `httpd` y `nginx` de Apache.

lshell: un intérprete de comandos restringido

El paquete `lshell` nunca se envió como parte de Amazon Linux. Estaba disponible en EPEL6. El [repositorio de paquetes de Fedora para lshell](#) explica [por qué no se incluyó](#) en EPEL7 ni en Fedora 30. También se [eliminó de Debian](#).

[El lshell proyecto original ya no se mantiene de forma activa y contiene componentes críticos conocidos sin parches CVEs: los CVE-2016-6902 y CVE-2016-6903.](#)

La alternativa sugerida en el error de Debian [rssh](#) tampoco se mantiene en el origen, y el autor cita problemas de seguridad imposibles de solucionar como motivo.

Por estas razones, no está previsto añadir el número 023. `lshell` AL2

monit: monitor de procesos, archivos, directorios y dispositivos

En Amazon Linux 2023, `systemd` ofrece una amplia gama de funciones para supervisar, iniciar, detener y reiniciar los servicios. Esto incluye limitar la velocidad de los reinicios, esperar entre intentos de reinicio e iniciar otro servicio en caso de error. Para obtener más información, consulte la documentación sobre [systemd.service](#).

En AL2 023, `systemd` también es compatible con la activación basada en rutas, que se puede utilizar para tomar medidas en caso de eventos, por ejemplo, cuando una ruta existe o cambia. Para obtener más información, consulte [systemd.path](#).

Existen opciones de configuración comunes para las unidades `systemd` que permiten especificar dependencias, condiciones y acciones a realizar en caso de éxito o fracaso. Para obtener más información, consulte la documentación sobre [systemd.unit](#).

`systemd` proporciona potentes funciones de control de recursos que pueden utilizarse para garantizar que ninguna tarea de supervisión utilice una cantidad excesiva de CPU o memoria. Para obtener más información, consulte [systemd.resource-control](#).

nodejs

El paquete de la nodejs versión 16 estaba disponible EPEL7 y ahora nodejs está incluido en la AL2 023. En el momento de escribir este artículo, tanto la nodejs versión 18 como la 20 estaban disponibles en la versión AL2 023. Puede instalar nodejs 18 en la AL2 023 con el siguiente comando:

```
[ec2-user ~]$ sudo dnf install nodejs
```

Puede instalar nodejs 20 en el AL2 023 con el siguiente comando:

```
[ec2-user ~]$ sudo dnf install nodejs20
```

perl-Config-General

El perl-Config-General paquete estaba incluido y ahora está incluido en el AL2 023. EPEL7 Puede instalar el perl-Config-General paquete en AL2 023 con el siguiente comando:

```
[ec2-user ~]$ sudo dnf install perl-Config-General
```

Los módulos de Perl también se pueden instalar solicitando a DNF que instale el paquete que proporciona un módulo de Perl en particular. Con este método, puede utilizar el nombre del módulo de Perl más familiar en lugar del nombre del paquete del sistema operativo.

```
[ec2-user ~]$ sudo dnf install 'perl(Config:General)'
```

python2-lockfile: bloqueo de archivos

El python2-lockfile paquete estaba aquí EPEL7 e AL2 incluía un python-lockfile paquete. En AL2 023 [Python 2.7 ha sido reemplazado por Python 3](#), por lo que una variante de Python 2 de este paquete no se agregará a AL2 023.

La versión Python 3 de este paquete está incluida en AL2 023. Puede instalar el python3-lockfile paquete en AL2 023 con uno de los siguientes comandos:

```
[ec2-user ~]$ sudo dnf install python3-lockfile
```

Los módulos de Python también se pueden instalar solicitando a DNF que instale el paquete que proporciona un módulo de Python en particular.

```
[ec2-user ~]$ sudo dnf install 'python3dist(lockfile)'
```

python2-rsa: Python RSA puro

El `python2-rsa` paquete estaba aquí EPEL7 e AL2 incluía un `python2-rsa` paquete. En AL2 023 [Python 2.7 ha sido reemplazado por Python 3](#), por lo que una variante de Python 2 de este paquete no se agregará a AL2 023.

La versión Python 3 de este paquete está incluida en AL2 023. Puede instalar el `python3-rsa` paquete en AL2 023 con uno de los siguientes comandos:

```
[ec2-user ~]$ sudo dnf install python3-rsa
```

Los módulos de Python también se pueden instalar solicitando a DNF que instale el paquete que proporciona un módulo de Python en particular.

```
[ec2-user ~]$ sudo dnf install 'python3dist(rsa)'
```

python2-simplejson: rutinas JSON para Python 2

El paquete `python2-simplejson` estaba disponible en EPEL7. En AL2 023 [Python 2.7 ha sido reemplazado por Python 3](#), por lo que una variante de Python 2 de este paquete no se agregará a AL2 023.

La versión Python 3 de este paquete está incluida en AL2 023. Puede instalar el `python3-simplejson` paquete en AL2 023 con el siguiente comando:

```
[ec2-user ~]$ sudo dnf install python3-simplejson
```

Los módulos de Python también se pueden instalar solicitando a DNF que instale el paquete que proporciona un módulo de Python en particular.

```
[ec2-user ~]$ sudo dnf install 'python3dist(simplejson)'
```

rkhunter: Rootkit Hunter

El `rkhunter` paquete viene incluido en el AL2 023 junto con. `chkrootkit`


```
[ec2-user ~]$ sudo dnf install rkhunter
```

```
[ec2-user ~]$ sudo dnf install chkrootkit
```

rsch: un intérprete de comandos restringido para usar con OpenSSH

El paquete `rsch` estaba disponible en EPEL7. El paquete [rsch](#) inicial no se mantiene, y el autor cita problemas de seguridad irreparables como motivo.

Dado que el autor cita problemas de seguridad imposibles de solucionar, no está previsto `rsch` añadir el AL2 023.

sscg: generador de certificados SSL autofirmado

El `sscg` paquete estaba en el AL2 núcleo, así como en. EPEL7 El `sscg` paquete también forma parte del AL2 023.

```
[ec2-user ~]$ sudo dnf install sscg
```

stress: prueba de esfuerzo

El `stress` paquete estaba disponible EPEL7 y también estaba disponible en AL1

El `stress` paquete está incluido en AL2 023 y se puede instalar ejecutando el siguiente comando:

```
[ec2-user ~]$ sudo dnf install stress
```

stress-ng: prueba de esfuerzo

El `stress-ng` paquete estaba incluido EPEL7 y también estaba disponible en el testing AL2 Extra.

El `stress-ng` paquete está incluido en la AL2 versión 023 y se puede instalar ejecutando el siguiente comando:

```
[ec2-user ~]$ sudo dnf install stress-ng
```

tmpwatch: elimina los archivos en función de la hora del último acceso

En Amazon Linux 2023, esta funcionalidad la proporciona [systemd-tmpfiles](#).

xmlstarlet: utilidades XML de línea de comandos

El `xmlstarlet` paquete estaba en el EPEL7 AL2 023 y no está disponible en él.

El paquete original no se ha modificado desde hace más de 9 años (se modificó por última vez en agosto de 2014). Durante los cuatro años anteriores (al menos desde julio de 2010), una solicitud para un nuevo mantenedor ha quedado sin respuesta. Es por esta razón que no está previsto añadir `xmlstarlet` al AL2 023.

Python 2.7 ha sido reemplazado por Python 3

AL2 proporciona soporte y parches de seguridad para Python 2.7 hasta junio de 2025, como parte de nuestro compromiso de soporte a largo plazo (LTS) para los paquetes AL2 principales. Este soporte se extiende más allá de la declaración de Python 2.7 de enero de 2020 de la comunidad end-of-life original de Python.

AL2 usa el administrador de `yum` paquetes, que depende en gran medida de Python 2.7. En AL2 023, el administrador de `dnf` paquetes migró a Python 3 y ya no requiere Python 2.7. AL2023 se ha trasladado por completo a Python 3.

Note

AL2023 eliminó Python 2.7, por lo que cualquier componente del sistema operativo que requiera Python está escrito para funcionar con Python 3. Para seguir utilizando una versión de Python proporcionada y compatible con Amazon Linux, convierta el código de Python 2 a Python 3.

Para obtener más información sobre Python en Amazon Linux, consulte [Python en AL2023](#).

Actualizaciones de seguridad

Amazon Linux 2023 mejora el endurecimiento presente en AL2. Para obtener más información, consulte [Seguridad y conformidad en Amazon Linux 2023](#). Para obtener más información sobre

los cambios en el endurecimiento del núcleo AL2, consulte. [Cambios en la configuración del kernel centrados en la seguridad](#)

Temas

- [SELinux](#)
- [OpenSSL 3](#)
- [IMDSv2](#)
- [Eliminación del hotpatch log4j \(log4j-cve-2021-44228-hotpatch\)](#)

SELinux

Por defecto, Security Enhanced Linux (SELinux) para AL2 023 es `enabled` y está configurado en modo `permissive`. En el modo `permissive`, las denegaciones de permisos se registran pero no se aplican.

SELinux es una función de seguridad del núcleo de Amazon Linux, que estaba `disabled` en AL2. SELinux es un conjunto de funciones y utilidades del núcleo que proporciona una arquitectura de control de acceso (MAC) obligatoria a los principales subsistemas del núcleo.

Para obtener más información, consulte [Configuración de los modos de SELinux para AL2023](#).

[Para obtener más información sobre SELinux los repositorios, las herramientas y las políticas, consulte SELinux Bloc de notas, Tipos de SELinux políticas y SELinux Proyecto.](#)

OpenSSL 3

AL2023 incluye el kit de herramientas de Open Secure Sockets Layer version 3 (OpenSSL 3) criptografía. AL2023 soporta TLS 1.3 y protocolos de red. TLS 1.2

Por defecto, AL2 viene con OpenSSL 1.0.2. Puede crear aplicaciones con OpenSSL 1.1.1.

Para obtener más información sobre OpenSSL, consulte la [guía de migración de OpenSSL](#).

Para obtener más información acerca de la seguridad, consulte [Actualizaciones y características de seguridad](#).

IMDSv2

De forma predeterminada, cualquier instancia lanzada con la AMI AL2 023 IMDSv2 solo requiere -y el límite de saltos predeterminado se establecerá en 2 para permitir el soporte de cargas de

trabajo en contenedores. Para ello, defina el parámetro `imds-support` en `v2.0`. Para obtener más información, consulte [Configurar la AMI](#) en la Guía del EC2 usuario de Amazon.

Note

El tiempo de validez del token de sesión puede oscilar entre 1 segundo y 6 horas. Las direcciones para dirigir las solicitudes API de las consultas IMDSv2 son las siguientes:

- IPv4: 169.254.169.254
- IPv6: fd00:ec2::254

Puede anular manualmente estas configuraciones y habilitar IMDSv1 mediante las propiedades de inicio de la opción de metadatos de instancia. También puede seguir utilizando los controles de IAM para aplicar configuraciones IMDS diferentes. Para obtener más información sobre la configuración y el uso del servicio de metadatos de instancias, consulte [Uso IMDSv2](#), [configuración de opciones de metadatos de instancia para instancias nuevas](#) y [Modificación de opciones de metadatos de instancia para instancias existentes](#), en la Guía del EC2 usuario de Amazon.

Eliminación del hotpatch log4j (**log4j-cve-2021-44228-hotpatch**)

Note

AL2EI 023 no se envía con el `log4j-cve-2021-44228-hotpatch` paquete.

En respuesta al error [CVE-2021-44228](#), Amazon Linux lanzó una versión empaquetada en RPM del [Hotpatch para Apache Log4j](#) para y. AL1 AL2 En el [anuncio de la adición del hotpatch a Amazon Linux](#), señalamos que “la instalación del hotpatch no reemplaza la actualización a una versión de log4j que mitigue los errores CVE-2021-44228 o CVE-2021-45046”.

El hotpatch era una medida de mitigación para dar tiempo a aplicar el parche log4j. La primera versión de disponibilidad general (GA) de la versión AL2 023 se publicó 15 meses después de la [CVE-2021-44228](#), por lo que la 023 no se incluye con el hotpatch (si está activado o no). AL2

Los usuarios que utilicen sus propias versiones de log4j en Amazon Linux deben asegurarse de actualizarlas a versiones que no estén afectadas por los códigos [CVE-2021-44228](#) o [CVE-2021-45046](#).

AL2La 023 proporciona orientación sobre cómo mantenerse al día con los [Actualización AL2023](#) parches de seguridad. Los avisos de seguridad se publican en el [Centro de seguridad de Amazon Linux](#).

Actualizaciones deterministas para mayor estabilidad

Con la función determinista de actualizaciones a través de repositorios versionados, cada AL2 AMI 023 está bloqueada de forma predeterminada en una versión de repositorio específica. Puede utilizar actualizaciones deterministas para lograr una mayor coherencia entre las versiones y actualizaciones de los paquetes. Cada versión, principal o secundaria, incluye una versión de repositorio específica.

Como novedad en la versión AL2 023, la actualización determinista está habilitada de forma predeterminada. Se trata de una mejora con respecto al método de bloqueo manual e incremental que se utilizaba en las versiones anteriores AL2 y en otras versiones anteriores.

Para obtener más información, consulte [Actualizaciones deterministas mediante repositorios versionados en AL2023](#).

gp3 como tipo de volumen de Amazon EBS predeterminado

La AMI AL2 023 y AL2 ambas utilizan el sistema de XFS archivos del sistema de archivos raíz. En el caso de AL2 023, las `mkfs` opciones del sistema de archivos del dispositivo raíz están optimizadas aún más para Amazon EC2. AL2El 023 también es compatible con otros sistemas de archivos que puede utilizar en otros volúmenes para cumplir con sus requisitos específicos.

AL2023 AMIs usa volúmenes de Amazon EBS de forma predeterminada, mientras que usa gp3 volúmenes de AL2 AMIs Amazon EBS gp2 de manera predeterminada. Puede cambiar el tipo de volumen cuando inicia una instancia.

Para obtener más información sobre los tipos de volúmenes de Amazon EBS, consulte [Volúmenes de Amazon EBS de uso general](#).

Para obtener más información sobre el lanzamiento de una EC2 instancia de Amazon, consulta [Lanzar una instancia](#) en la Guía del EC2 usuario de Amazon.

Jerarquía de grupos de control unificados (cgroup v2)

Un grupo de control (cgroup) es una característica del kernel de Linux que permite organizar jerárquicamente los procesos y distribuir los recursos del sistema entre ellos. Los grupos de control

se utilizan ampliamente para implementar un tiempo de ejecución de contenedores, y mediante `systemd`.

AL2 soportescgroupv1, y AL2 023 soportescgroupv2. Esto es destacable si se ejecutan cargas de trabajo en contenedores, como en [Uso de AMI de Amazon ECS basadas en AL2023 para alojar cargas de trabajo en contenedores](#).

Aunque la AL2 023 sigue incluyendo código que permite que el sistema funcione con cgroupv1 ella, no es una configuración recomendada ni admitida, y se eliminará por completo en una futura versión importante de Amazon Linux.

Existe una amplia documentación sobre las [interfaces del kernel de Linux de bajo nivel](#), así como [documentación de delegación de systemd cgroup](#).

Un caso de uso común fuera de los contenedores es la creación de unidades `systemd` que limiten los recursos del sistema que pueden utilizar. Para obtener más información, consulte [systemd.resource-control](#).

Los temporizadores de **systemd** reemplazan a **cron**

El `cronie` paquete se instaló de forma predeterminada en la AL2 AMI, lo que proporciona soporte para la `crontab` forma tradicional de programar tareas periódicas. En AL2 023, no `cronie` se incluye de forma predeterminada. Por lo tanto, ya no se ofrece soporte para `crontab` de forma predeterminada.

Si lo desea, puede instalar el paquete `cronie` para utilizar los trabajos clásicos `cron`. Se recomienda migrar a temporizadores `systemd` debido a la funcionalidad adicional que proporcionan mediante `systemd`.

Cadena de herramientas mejorada: gcc, binutils y glibc

AL2023 incluye muchos de los mismos paquetes principales que. AL2

Hemos actualizado los siguientes tres paquetes principales de cadenas de herramientas para AL2 el 023.

Nombre del paquete	AL2	AL2023
glibc	2.26	2.34

Nombre del paquete	AL2	AL2023
gcc	7.3	11.3
binutils	2.29	2.39

Para obtener más información, consulte [Paquetes principales de cadenas de herramientas glibc, gcc, binutils](#).

Para obtener más información sobre C, C++ y los tiempos de ejecución del lenguaje Fortran, incluidos los estándares predeterminados actualizados del lenguaje, consulte [C, C++ y Fortran en AL2023](#).

Para obtener más información sobre optimizaciones, consulte [Optimizaciones operativas y de rendimiento](#).

El diario **systemd** reemplaza a **rsyslog**

En AL2 023, el paquete del sistema de registro ha cambiado de. AL2 AL2EI 023 no se instala de forma predeterminada, **rsyslog** por lo que los archivos de registro basados en texto, como los `/var/log/messages` que estaban disponibles en, AL2 no están disponibles de forma predeterminada. La configuración por defecto de AL2 023 es **systemd-journal**, que puede examinarse utilizando. **journalctl** Aunque **rsyslog** es un paquete opcional en AL2 023, recomendamos la nueva **journalctl** interfaz **systemd** basada y los paquetes relacionados. Para obtener más información, consulte la página del manual [journalctl](#).

En la siguiente tabla se describe el equivalente de **systemd journal** a algunos comandos de **syslog** de uso común.

AL2 syslog command	AL2EI equivalente a 023 systemd journal
<code>[ec2-user ~]\$ cat /var/log/messages</code>	<code>[ec2-user ~]\$ journalctl</code>
<code>[ec2-user ~]\$ tail -f /var/log/messages</code>	<code>[ec2-user ~]\$ journalctl -f</code>

AL2 syslog command	AL2EI equivalente a 023 systemd journal
<pre>[ec2-user ~]\$ grep foo /var/log/messages</pre>	<pre>[ec2-user ~]\$ journalctl grep foo</pre>

Dependencias del paquete minimizadas

Amazon Linux 2023 minimiza el gráfico de dependencias de muchos paquetes para reducir el tamaño de las aplicaciones. Entre los cambios más destacables se AL2 incluyen los `gnupg-minimal` paquetes `curl-minimal` y, que reducen significativamente la cantidad de paquetes necesarios y, al mismo tiempo, conservan las funciones de uso común.

Temas

- [Cambios en los paquetes para curl y libcurl](#)
- [GNU Privacy Guard \(GNUPG\)](#)

Cambios en los paquetes para **curl** y **libcurl**

AL2023 separa los protocolos y la funcionalidad comunes de los paquetes y los divide en `curl` y `libcurl`. `curl-minimal` `libcurl-minimal` Esto reduce el espacio de disco, memoria y dependencia para la mayoría de los usuarios y es el paquete predeterminado para el AL2 023 AMIs y los contenedores.

Si se requiere la funcionalidad completa de `curl`, por ejemplo, para el soporte `gopher://`, ejecute los siguientes comandos para instalar los paquetes `curl-full` y `libcurl-full`.

```
$ dnf swap libcurl-minimal libcurl-full
```

```
$ dnf swap curl-minimal curl-full
```

GNU Privacy Guard (GNUPG)

AL2EI 023 separa la funcionalidad mínima de la completa del `gnupg2` paquete en paquetes `gnupg2-minimal` y `gnupg2-full` paquetes. De forma predeterminada, solo está instalado el

paquete `gnupg2-minimal`. Esto proporciona la funcionalidad mínima necesaria para verificar las firmas digitales de los paquetes `rpm`.

Para obtener más funcionalidad de `gnupg2`, como la posibilidad de descargar claves de un servidor de claves, asegúrese de que el paquete `gnupg2-full` esté instalado. Ejecute el siguiente comando para intercambiar `gnupg2-minimal` por `gnupg2-full`.

```
$ dnf swap gnupg2-minimal gnupg2-full
```

Amazon Corretto como JVM predeterminada

AL2023 incluye [Amazon Corretto](#) como kit de desarrollo de Java (JDK) predeterminado (y único). Todos los paquetes Java basados en AL2 023 están contruidos con. Amazon Corretto 17

Si está migrando desde AL2, puede realizar la transición sin problemas de la OpenJDK versión equivalente a la siguiente. AL2 Amazon Corretto

AWS CLIV2

AL2EI 023 se envía con AWS CLI la versión 2, mientras que se AL2 envía con la versión 1 delAWS CLI.

UEFI Preferred y arranque seguro

De forma predeterminada, todas las instancias lanzadas con la AMI AL2 023 en tipos de instancias que admitan el firmware UEFI se iniciarán en modo UEFI. Para ello, configure el parámetro AMI del modo de arranque en `uefi-preferred`. Para obtener más información, consulta [Modos de arranque](#) en la Guía del EC2 usuario de Amazon.

En los tipos de EC2 instancias de Amazon que admiten el arranque seguro de UEFI, es posible habilitar el arranque seguro en Amazon Linux 2023. Para obtener más información, consulte [UEFI Secure Boot está activado AL2023](#).

Cambios en la configuración predeterminada del servidor SSH

Para la AMI AL2 023, cambiamos los tipos de claves de `sshd` host que generamos con la versión. También eliminamos algunos tipos de claves antiguas para evitar generarlos en el momento del

lanzamiento. Los clientes deben admitir los protocolos `rsa-sha2-256` y `rsa-sha2-512` o `ssh-ed25519` con el uso de una clave `ed25519`. De forma predeterminada, las firmas `ssh-rsa` están desactivadas.

Además, las AL2 023 opciones de configuración del `sshd_config` archivo predeterminado contienen: `UseDNS=no` Con esta nueva configuración, es menos probable que las deficiencias de DNS bloqueen la capacidad de establecer sesiones `ssh` con las instancias. La desventaja es que las entradas de línea `from=hostname.domain,hostname.domain` de los archivos `authorized_keys` no se resolverán. Como `sshd` ya no intenta resolver los nombres de DNS, cada valor `hostname.domain` separado por comas debe traducirse a la correspondiente IP address.

Para obtener más información, consulte [Configuración predeterminada del servidor SSH](#).

AL2023 el kernel cambia de AL2

AL2023 incluye el kernel 6.1, así como muchos cambios de configuración para optimizar aún más Amazon Linux para la nube. Para la mayoría de los usuarios, estos cambios deberían ser muy claros.

IPv4 TTL

El TTL para IPv4 se configura mediante `sysctl`, con los valores predeterminados presentes en. `/etc/sysctl.d/00-defaults.conf` Este valor se puede personalizar mediante los métodos `sysctl` habituales. Para obtener más información, consulte la página man de .

AL2 establece el `net.ipv4.ip_default_ttl` valor en 255, mientras que lo AL2023 establece en 127. Esto hace que los valores predeterminados de Amazon Linux estén en línea con los de otras distribuciones principales de Linux. No se recomienda cambiar este valor predeterminado sin que se demuestre que es necesario.

Cambios en la configuración del kernel centrados en la seguridad

Opción	AL2/4.1	AL2/4.1	AL2/5.1	AL2/5.1	AL2023	AL2023	AL2023	AL2023	AL2023	AL2023/6.
CONFIG	aarch64	x86_64	aarch64	x86_64	1/ arch64	1/ x86_64	12/ aarch64	12/ x86_64	18/ aarch64	18/ x86_64
CONFIG_G_ON_D	n	y	n	y	y	y	y	y	y	y

Opción	AL2/4.1	AL2/4.1	AL2/5.1	AL2/5.1	AL2023	AL2023	AL2023	AL2023	AL2023	AL2023/6.
CONFIG	aarch64	x86_64	aarch64	x86_64	1/ arch64	1/ x86_64	12/ aarch64	12/ x86_64	18/ aarch64	18/ x86_64
<u>__CORRUPTION</u>										
<u>CONFIG_FAULT_P_MIN_R</u>	4096	4096	4096	4096	65536	65536	65536	65536	65536	65536
<u>CONFIG_VMEM</u>	n	y	n	y	n	n	n	n	n	n
<u>CONFIG_VPORT</u>	n	y	n	y	n	n	n	n	n	n
<u>CONFIG_RTIFY_RCE</u>	n	y	n	y	y	y	y	y	y	y
<u>CONFIG_RDENED_ERCOPY_LLBACK</u>	N/A	N/A	y	y	N/A	N/A	N/A	N/A	N/A	N/A
<u>CONFIG_IT_ON_OC_DEFAULT_ON</u>	N/A	N/A	n	n	n	n	n	n	n	n
<u>CONFIG_IT_ON_E_DEFAULT_ON</u>	N/A	N/A	n	n	n	n	n	n	n	n

Opción	AL2/4.1	AL2/4.1	AL2/5.1	AL2/5.1	AL2023	AL2023	AL2023	AL2023	AL2023	AL2023/6.
CONFIG	aarch64	x86_64	aarch64	x86_64	1/ arch64	1/ x86_64	12/ aarch64	12/ x86_64	18/ aarch64	18/ x86_64
CONFIG_MMU_DELT_DMA_RICT	N/A	N/A	N/A	N/A	n	n	n	n	n	n
CONFIG_ISC_AUOAD	y	y	y	y	n	n	n	n	n	n
CONFIG_HED_CO	N/A	N/A	N/A	N/A	N/A	y	N/A	y	N/A	y
CONFIG_HED_ST_END_CK	n	y	n	y	y	y	y	y	y	y
CONFIG_CURITY_ESG_REICT	n	n	n	n	y	y	y	y	y	y
CONFIG_CURITY_LINUX_ABLE	y	y	y	y	n	n	N/A	N/A	N/A	N/A
CONFIG_UFFLE_E_ALLC_OR	N/A	N/A	y	y	y	y	y	y	y	y

Opción CONFIG	AL2/4.1 aarch64	AL2/4.1 x86_64	AL2/5.1 aarch64	AL2/5.1 x86_64	AL2023 1/ arch64	AL2023 1/ x86_64	AL2023 12/ aarch6 4	AL2023 12/ x86_64	AL2023 18/ aarch6 4	AL2023/6. 18/ x86_64
CONFIG_AB_FREEST_HEAP	n	y	y	y	y	y	y	y	y	y
CONFIG_AB_FREEST_RAN	n	n	y	y	y	y	y	y	y	y

Cambios específicos en la configuración del kernel centrados en la seguridad de x86-64

Opción CONFIG	AL2/4,14/ x86_64	AL2/5.10/ x86_64	AL2023/6,1/ x86_64	AL2023/6,12/ x86_64	AL2023/6,18/ x86_64
CONFIG_AMD_IOMMU	y	y	y	y	y
CONFIG_AMD_IOMMU_V2	m	m	y	N/A	N/A
CONFIG_RANDOMIZE_MEMORY	N/A	y	y	y	y

Cambios específicos en la configuración del kernel centrados en la seguridad de aarch64 (ARM/Graviton)

Opción CONFIG	AL2/4.14/ aarch64	AL2/5.10/ aarch64	AL2023/6.1/ aarch64	AL2023/6.12/ aarch64	AL2023/6.18/ aarch64
CONFIG_ARM64_PTR_AUTH	N/A	y	y	y	y
CONFIG_ARM64_PTR_AUTH_KERNEL	N/A	N/A	y	y	y
CONFIG_ARM64_SW_TTBR0_PAN	y	y	y	y	y

/dev/mem, /dev/kmem y /dev/port

Amazon Linux 2023 desactiva /dev/mem y /dev/port (CONFIG_DEVMEMyCONFIG_DEVPORT) por completo, se basa en las restricciones ya vigentes en AL2.

El /dev/kmem código se eliminó por completo de Linux en el kernel 5.13 y, aunque estaba deshabilitado en AL2, ahora no es aplicable a. AL2023

Esta opción es una de las [configuraciones recomendadas por el proyecto de autoprotección del kernel](#).

FORTIFY_SOURCE

AL2023 se habilita CONFIG_FORTIFY_SOURCE en todas las arquitecturas compatibles. Esta característica es una característica de refuerzo de la seguridad. Cuando el compilador puede determinar y validar los tamaños de los búferes, esta característica puede detectar los desbordamientos del búfer en funciones comunes de cadenas y memoria.

Esta opción es una de las [configuraciones recomendadas por el proyecto de autoprotección del kernel](#).

Carga automática de Line Discipline (**CONFIG_LDISC_AUTOLOAD**)

El AL2023 núcleo no carga automáticamente las disciplinas de línea, por ejemplo, si el software utiliza el `TIOCSETDioctl`, a menos que la solicitud provenga de un proceso con los `CAP_SYS_MODULE` permisos necesarios.

Esta opción es una de las [configuraciones recomendadas por el proyecto de autoprotección del kernel](#).

Acceso **dmesg** para usuarios sin privilegios (**CONFIG_SECURITY_DMESG_RESTRICT**)

De forma predeterminada, AL2023 no permite que los usuarios sin privilegios accedan `admesg`.

Esta opción es una de las [configuraciones recomendadas por el proyecto de autoprotección del kernel](#).

SELinux **selinuxfs** inhabilitar

AL2023 desactiva la opción obsoleta `CONFIG_SECURITY_SELINUX_DISABLE` del núcleo, que permitía deshabilitar la política mediante un método de ejecución SELinux antes de cargar la política.

Esta opción es una de las [configuraciones recomendadas por el proyecto de autoprotección del kernel](#).

Otros cambios en la configuración del kernel

Opción	AL2/4.1	AL2/4.1	AL2/5.1	AL2/5.1	AL2023	AL2023	AL2023	AL2023	AL2023	AL2023/6.
CONFIG	aarch64	x86_64	aarch64	x86_64	1/ arch64	1/ x86_64	12/ aarch64	12/ x86_64	18/ aarch64	18/ x86_64
CONFIG	100	250	100	250	100	100	100	100	100	100
CONFIG	4096	8192	4096	8192	4096	8192	4096	8192	4096	8192
CONFIG	y	n	y	n	y	y	y	y	y	y
NIC_ON										
PS										

Opción	AL2/4.1	AL2/4.1	AL2/5.1	AL2/5.1	AL2023	AL2023	AL2023	AL2023	AL2023	AL2023/6.
CONFIG	aarch64	x86_64	aarch64	x86_64	1/	1/	12/	12/	18/	18/
					arch64	x86_64	aarch64	x86_64	aarch64	x86_64
CONFIG_NIC_ONPS_VAL	1	0	1	0	1	1	1	1	N/A	N/A
CONFIG_P	m	m	m	m	m	m	m	m	m	m
CONFIG_IP	m	m	m	m	n	n	n	n	n	n
CONFIG_N_PV	N/A	y	N/A	n	N/A	n	N/A	n	N/A	n

CONFIG_HZ

AL2023 se establece en 100 en ambas CONFIG_HZ plataformas. x86-64 aarch64

CONFIG_NR_CPUS

AL2023 se establece CONFIG_NR_CPUS en un número más cercano al número máximo de núcleos de CPU que se encuentran en Amazon EC2.

Pánico en kernel

El AL2023 kernel entrará en pánico cuando se interrumpa. Esta característica equivale a arrancar oops=panic desde la línea de comandos del kernel.

Kernel oops se produce cuando kernel ha detectado un error interno que puede afectar a la fiabilidad del sistema.

Compatibilidad con PPP y SLIP

AL2023 no es compatible con el protocolo SLIP, pero puede admitir el protocolo PPP en los núcleos más recientes. AL2023

Compatibilidad con el huésped de Xen PV

AL2023 no admite la ejecución como huésped de Xen PV.

Compatibilidad con sistema de archivos del kernel

Se han producido varios cambios en los sistemas de archivos en los que el núcleo AL2 admitirá el montaje, además de cambios en los esquemas de particionamiento que analizará el núcleo.

Opción	AL2/4.1	AL2/4.1	AL2/5.1	AL2/5.1	AL2023	AL2023	AL2023	AL2023	AL2023	AL2023/6.
CONFIG	aarch64	x86_64	aarch64	x86_64	1/ arch64	1/ x86_64	12/ aarch64	12/ x86_64	18/ aarch64	18/ x86_64
CONFIG_S_FS	n	m	n	m	n	n	n	n	n	n
CONFIG_RXRPC	n	m	n	m	n	n	n	n	n	n
CONFIG_D_DISK_EL	y	y	y	y	n	n	n	n	n	n
CONFIG_AMFS	m	m	m	m	n	n	n	n	n	n
CONFIG_AMFS_BKDEV	N/A	N/A	y	n	N/A	N/A	N/A	N/A	N/A	N/A
CONFIG_CLONE	N/A	N/A	n	n	n	n	n	n	n	n
CONFIG_EA	m	n	m	n	n	n	n	n	n	n

Opción CONFIG	AL2/4.1 aarch64	AL2/4.1 x86_64	AL2/5.1 aarch64	AL2/5.1 x86_64	AL2023 1/ arch64	AL2023 1/ x86_64	AL2023 12/ aarch6 4	AL2023 12/ x86_64	AL2023 18/ aarch6 4	AL2023/6. 18/ x86_64
CONFIG_INTEGRITY	n	m	n	m	m	m	m	m	m	m
CONFIG_LOG_WARNINGS	n	n	m	m	m	m	m	m	m	m
CONFIG_SWIOTLB	m	n	m	n	n	n	n	n	n	n
CONFIG_VERIFY	m	n	m	n	m	m	m	m	m	m
CONFIG_RYPTFS	n	m	n	m	n	n	n	n	n	n
CONFIG_FAT_FS	N/A	N/A	m	m	m	m	m	m	m	m
CONFIG_T2_FS	n	m	n	m	n	n	n	n	n	n
CONFIG_T3_FS	n	m	n	m	n	n	n	n	N/A	N/A
CONFIG_S2_FS	m	m	m	m	n	n	n	n	n	n
CONFIG_SPLUS_FS	n	m	n	m	n	n	n	n	n	n
CONFIG_S_FS	n	m	n	m	n	n	n	n	n	n

Opción	AL2/4.1	AL2/4.1	AL2/5.1	AL2/5.1	AL2023	AL2023	AL2023	AL2023	AL2023	AL2023/6.
CONFIG	aarch64	x86_64	aarch64	x86_64	1/ arch64	1/ x86_64	12/ aarch64	12/ x86_64	18/ aarch64	18/ x86_64
CONFIG_S_FS	n	n	n	n	n	n	n	n	n	n
CONFIG_M_PARTON	n	y	n	y	n	n	n	n	n	n
CONFIG_C_PARTON	n	y	n	y	n	n	n	n	n	n
CONFIG_S_V2	n	m	n	m	n	n	n	n	n	n
CONFIG_FS_FS	n	m	n	n	n	n	n	n	n	n
CONFIG_MFS_FS	n	m	n	m	n	n	n	n	n	n
CONFIG_LARIS_PARTI_N	n	y	n	y	n	n	n	n	n	n
CONFIG_UASHFSTD	n	y	n	y	y	y	y	y	y	y
CONFIG_N_PARTON	n	y	n	y	n	n	n	n	n	n

Compatibilidad con el sistema de archivos Andrew (AFS)

El núcleo ya no está construido con soporte para el sistema de archivos. `afs` AL2 no se envió con soporte de espacio de usuario para `afs`.

Compatibilidad con `cramfs`

Kernel ya no se crea con soporte para el sistema de archivos `cramfs`. El sucesor AL2023 es el sistema de `squashfs` archivos.

Compatibilidad con las etiquetas de disco BSD

Kernel ya no se crea con soporte para etiquetas de disco BSD. Si es necesario leer volúmenes con etiquetas de disco BSD, se BSDs pueden lanzar varios.

Device Mapper cambia

Se han realizado varios cambios en los destinos de Device Mapper configurados en el AL2023 núcleo.

eCryptFs soporte

El sistema de archivos `ecryptfs` ha quedado obsoleto en Amazon Linux. Los componentes del espacio de usuario `ecryptfs` estaban presentes AL1, se eliminaron y ya AL2023 no compilan el núcleo con `ecryptfs` soporte. AL2

exFAT

Support para el sistema de `exFAT` archivos se agregó en el kernel 5.10 en AL2. No estaba presente en el momento del AL2 lanzamiento con un núcleo de la versión 4.14. AL2023 sigue siendo compatible con el sistema de `exFAT` archivos.

Sistemas de archivos `ext2`, `ext3` y `ext4`

AL2023 incluye la `CONFIG_EXT4_USE_FOR_EXT2` opción, lo que significa que el código del sistema de `ext4` archivos se utilizará para leer los sistemas de `ext2` archivos antiguos.

`CONFIG__FS` GFS2

El núcleo ya no está compilado con `CONFIG__FS`. GFS2

Compatibilidad con el sistema de archivos extendido HFS de Apple (HFS+)

En AL2, solo los x86-64 núcleos se crearon con el soporte del hfsplus sistema de archivos. El núcleo AL2 5.15 no incluye hfsplus soporte para ninguna arquitectura. En AL2023, completamos la obsolescencia del hfsplus soporte en Amazon Linux.

Compatibilidad con el sistema de archivos HFS

En AL2, solo los x86-64 núcleos se crearon con el soporte del sistema de hfs archivos. El núcleo AL2 5.15 no incluye hfs soporte para ninguna arquitectura. En AL2023, completamos la obsolescencia del hfs soporte en Amazon Linux.

Compatibilidad con sistema de archivos JFS

Los AL2 x86-64 núcleos más antiguos se crearon con soporte para sistemas de jfs archivos. El núcleo AL2 5.15 no incluye jfs soporte para ninguna arquitectura. Ninguno de los dos AL1 o AL2 se suministra con el espacio de usuario de JFS. En AL2023, completamos la obsolescencia del jfs soporte en Amazon Linux.

El kernel original de Linux está [considerando la posibilidad de eliminar JFS](#). Por lo tanto, si tiene datos en un sistema de archivos JFS, debe migrarlos a otro sistema de archivos. En 2024, JFS se eliminó de todos los kernels de Amazon Linux actuales.

Compatibilidad con el Administrador de discos lógicos (disco dinámico) de Windows (CONFIG_LDM_PARTITION)

AL2023 ya no admite Windows 2000 discos Windows Vista dinámicos o dinámicos con particiones de MS-DOS estilo. Windows XP Este código nunca fue compatible con los nuevos discos dinámicos basados en GPT presentados con Windows Vista.

Compatibilidad con los mapas de particiones de Macintosh

AL2023 ya no es compatible con el mapa de particiones clásico de Macintosh. Las versiones modernas de macOS crearán tablas de particiones GPT modernas de forma predeterminada sobre este tipo anterior.

NFSv2 soporte

AL2023 ya no admite NFSv2, pero sigue siendo compatible con NFSv3 NFSv4, NFSv4 .1 y NFSv4 .2. Se recomienda migrar a una versión NFSv3 anterior o posterior.

NTFS (**CONFIG_NTFS_FS**)

El `ntfs3` código se sustituyó `ntfs` para acceder a los sistemas de archivos NTFS en Amazon Linux a partir del kernel 5.10 en. AL2 AL2023 ya no incluye el `ntfs` código y se basa exclusivamente en él para acceder a los `ntfs3` sistemas de archivos NTFS.

sistema de archivos romfs

El sistema de `squashfs` archivos es el sucesor del sistema de `romfs` archivos de Amazon Linux y el AL2023 núcleo ya no se crea con soporte para `romfs`.

Formato de partición de disco duro x86 de Solaris

AL2023 ya no es compatible con el formato de partición de disco duro x86 de Solaris.

squashfszstdcompresión

AL2023 añade soporte para sistemas de `squashfs` archivos `zstd` comprimidos en todas las arquitecturas compatibles.

Compatibilidad con las tablas de particiones Sun

AL2023 ya no admite el formato de tabla de particiones Sun (`CONFIG_SUN_PARTITION`).

/tmp ahora es **tmpfs**

Amazon Linux 2023 introduce cambios en el comportamiento de `/tmp` en comparación con Amazon Linux 2. La configuración por defecto AL2 era que ambos `/tmp` `/var/tmp` estuvieran en el sistema de archivos raíz. Amazon Linux 2023 usa de forma predeterminada `tmpfs` para `/tmp` con un límite del 50 % de RAM y un máximo de un millón de inodes. Estos cambios hacen que Amazon Linux se adapte al comportamiento de otras distribuciones de Linux.

Para obtener todos los detalles del diseño del sistema de archivos de AL2 023, consulte [/tmp](#) y [/var/tmp](#) en la [Diseño del sistema de archivos](#) sección.

Cambios en la AML y en la imagen del contenedor

Se han realizado algunos cambios en los paquetes incluidos en AMLs los contenedores.

Amazon Linux 2023 presenta un [the section called “AL2023 Imagen de contenedor mínima”](#) y soporte para la compilación de [the section called “Creación de imágenes de contenedores básicas AL2023”](#). Para obtener más información, consulte [Uso AL2023 en contenedores](#).

Comparación de paquetes instalados en las AMI de Amazon Linux 2 y Amazon Linux 2023.

Comparación de las RPM presentes en las AMI estándar de Amazon Linux 2 y AL2023.

Paquete	AMI de AL2	AMI de AL2023
acl	2.2.51	2.3.1
acpid	2.0.19	2.0.32
alternatives		1.15
amazon-chrony-config		4.3
amazon-ec2-net-utils		2.5.1
amazon-linux-extras	2.0.3	
amazon-linux-extras-yum-plugin	2.0.3	
amazon-linux-repo-s3		2023.6.20241031
amazon-linux-sb-keys		2023.1
amazon-rpm-config		228
amazon-ssm-agent	3.3.987.0	3.3.987.0
amd-ucode-firmware	20200421 (noarch)	20210208 (noarch)
at	3.1.13	3.1.23
attr	2.4.46	2.5.1

Paquete	AMI de AL2	AMI de AL2023
audit	2.8.1	3.0.6
audit-libs	2.8.1	3.0.6
authconfig	6.2.8	
aws-cfn-bootstrap	2.0	2.0
awscli	1.18.147	
awscli-2		2.15.30
basesystem	10.0	11
bash	4.2.46	5.2.15
bash-completion	2.1	2.11
bc	1.06.95	1.07.1
bind-export-libs	9.11.4	
bind-libs	9.11.4	9.18.28
bind-libs-lite	9.11.4	
bind-license	9.11.4	9.18.28
bind-utils	9.11.4	9.18.28
binutils	2.29.1	2.39
blktrace	1.0.5	
boost-date-time	1.53.0 (x86_64)	
boost-filesystem		1.75.0
boost-system	1.53.0 (x86_64)	1.75.0

Paquete	AMI de AL2	AMI de AL2023
boost-thread	1.53.0 (x86_64)	1.75.0
bridge-utils	1.5	
bzip2	1.0.6	1.0.8
bzip2-libs	1.0.6	1.0.8
ca-certificates	2023.2.68	2023.2.68
c-ares		1.19.1
checkpolicy		3.4
chkconfig	1.7.4	1.15
chrony	4.2	4.3
cloud-init	19.3	22.2.2
cloud-init-cfg-ec2		22.2.2
cloud-utils-growpart	0.31	0.31
coreutils	8.22	8.32
coreutils-common		8.32
cpio	2.12	2.13
cracklib	2.9.0	2.9.6
cracklib-dicts	2.9.0	2.9.6
cronie	1.4.11	
cronie-anacron	1.4.11	
crontabs	1.11	1.11

Paquete	AMI de AL2	AMI de AL2023
crypto-policies		20220428
crypto-policies-scripts		20220428
cryptsetup	1.7.4	2.6.1
cryptsetup-libs	1.7.4	2.6.1
curl	8.3.0	
curl-minimal		8.5.0
cyrus-sasl-lib	2.1.26	2.1.27
cyrus-sasl-plain	2.1.26	2.1.27
dbus	1.10.24	1.12.28
dbus-broker		32
dbus-common		1.12.28
dbus-libs	1.10.24	1.12.28
device-mapper	1.02.170	1.02.185
device-mapper-event	1.02.170	
device-mapper-event-libs	1.02.170	
device-mapper-libs	1.02.170	1.02.185
device-mapper-persistent-data	0.7.3	
dhclient	4.2.5	

Paquete	AMI de AL2	AMI de AL2023
dhcp-common	4.2.5	
dhcp-libs	4.2.5	
diffutils	3.3	3.8
dmidecode	3.2	
dmraid	1.0.0.rc16	
dmraid-events	1.0.0.rc16	
dnf		4.14.0
dnf-data		4.14.0
dnf-plugin-release-notification		1.2
dnf-plugins-core		4.3.0
dnf-plugin-support-info		1.2
dnf-utils		4.3.0
dosfstools	3.0.20	4.2
dracut	033	055
dracut-config-ec2	2.0	3.0
dracut-config-generic	033	055
dwz		0.14
dyninst	9.3.1 (x86_64)	10.2.1

Paquete	AMI de AL2	AMI de AL2023
e2fsprogs	1.42.9	1.46.5
e2fsprogs-libs	1.42.9	1.46.5
ec2-hibinit-agent	1.0.8	1.0.8
ec2-instance-connect	1.1	1.1
ec2-instance-connect-selinux	1.1	1.1
ec2-net-utils	1.7.3	
ec2-utils	1.2	2.2.0
ed	1.9	1.14.2
efibootmgr	15 (aarch64)	
efi-filesystem		5
efi-srpm-macros		5
efivar		38
efivar-libs	31 (aarch64)	38
elfutils-debuginfod-client		0.188
elfutils-default-yama-scope	0.176	0.188
elfutils-libelf	0.176	0.188
elfutils-libs	0.176	0.188
ethtool	4.8	5.15

Paquete	AMI de AL2	AMI de AL2023
expat	2.1.0	2.5.0
file	5.11	5.39
file-libs	5.11	5.39
filesystem	3.2	3.14
findutils	4.5.11	4.8.0
fipscheck	1.4.1	
fipscheck-lib	1.4.1	
fonts-srpm-macros		2.0.5
freetype	2.8	
fstrm		0.6.1
fuse-libs	2.9.2	2.9.9
gawk	4.0.2	5.1.0
gdbm	1.13	
gdbm-libs		1.19
gdisk	0.8.10	1.0.8
generic-logos	18.0.0	
GeoIP	1.5.0	
gettext	0.19.8.1	0.21
gettext-libs	0.19.8.1	0.21
ghc-srpm-macros		1.5.0

Paquete	AMI de AL2	AMI de AL2023
glib2	2.56.1	2.74.7
glibc	2.26	2.34
glibc-all-langpacks	2.26	2.34
glibc-common	2.26	2.34
glibc-gconv-extra		2.34
glibc-locale-source	2.26	2.34
glibc-minimal-lang pack	2.26	
gmp	6.0.0	6.2.1
gnupg2	2.0.22	
gnupg2-minimal		2.3.7
gnutls		3.8.0
go-srpm-macros		3.2.0
gpgme	1.3.2	1.15.1
gpm-libs	1.20.7	1.20.7
grep	2.20	3.8
groff-base	1.22.2	1.22.4
grub2	2.06	
grub2-common	2.06	2.06
grub2-efi-aa64	2.06 (aarch64)	
grub2-efi-aa64-ec2	2.06 (aarch64)	2.06 (aarch64)

Paquete	AMI de AL2	AMI de AL2023
grub2-efi-aa64-modules	2.06 (noarch)	
grub2-efi-x64-ec2	2.06 (x86_64)	2.06 (x86_64)
grub2-pc	2.06 (x86_64)	
grub2-pc-modules	2.06 (noarch)	2.06
grub2-tools	2.06	2.06
grub2-tools-minimal	2.06	2.06
grubby	8.28	8.40
gssproxy	0.7.0	0.8.4
gzip	1.5	1.12
hardlink	1.3	
hibagent	1.1.0	
hostname	3.13	3.23
hunspell	1.3.2	1.7.0
hunspell-en	0.20121024	0.20140811.1
hunspell-en-GB	0.20121024	0.20140811.1
hunspell-en-US	0.20121024	0.20140811.1
hunspell-filesystem		1.7.0
hwdata	0.252	0.384
info	5.1	6.7
inih		49

Paquete	AMI de AL2	AMI de AL2023
initscripts	9.49.47	10.09
iproute	5.10.0	6.10.0
iptables	1.8.4	
iptables-lib	1.8.4	
iputils	20180629	20210202
irqbalance	1.7.0	1.9.0
jansson	(2.10)	2.14
jbigkit-lib	2.0	
jemalloc		5.2.1
jitterentropy		3.4.1
jq		1.7.1
json-c	0.11	0.14
kbd	1.15.5	2.4.0
kbd-legacy	1.15.5	
kbd-misc	1.15.5	2.4.0
kernel	5.10.228	6.1.112
kernel-libbpf		6.1.112
kernel-livepatch-r epo-s3		2023.6.20241031
kernel-srpm-macros		1.0
kernel-tools	5.10.228	6.1.112

Paquete	AMI de AL2	AMI de AL2023
keyutils	1.5.8	1.6.3
keyutils-libs	1.5.8	1.6.3
kmod	25	29
kmod-libs	25	29
kpartx	0.4.9	
kpatch-runtime	0.9.4	0.9.7
krb5-libs	1.15.1	1.21.3
langtable	0.0.31	
langtable-data	0.0.31	
langtable-python	0.0.31	
less	458	608
libacl	2.2.51	2.3.1
libaio	0.3.109	0.3.111
libarchive		3.7.4
libargon2		20171227
libassuan	2.1.0	2.5.5
libattr	2.4.46	2.5.1
libbasicobjects	0.1.1	0.1.1
libblkid	2.30.2	2.37.4
libcap	2.54	2.48

Paquete	AMI de AL2	AMI de AL2023
libcap-ng	0.7.5	0.8.2
libcbor		0.7.0
libcollection	0.7.0	0.7.0
libcom_err	1.42.9	1.46.5
libcomps		0.1.20
libconfig	1.4.9	1.7.2
libcroco	0.6.12	
libcrypt	2.26	
libcurl	8.3.0	
libcurl-minimal		8.5.0
libdaemon	0.14	
libdb	5.3.21	5.3.28
libdb-utils	5.3.21	
libdhash		0.5.0
libdnf		0.69.0
libdrm	2.4.97	
libdwarf	20130207 (x86_64)	
libeconf		0.4.0
libedit	3.0	3.1
libestr	0.1.9	

Paquete	AMI de AL2	AMI de AL2023
libev		4.33
libevent	2.0.21	2.1.12
libfastjson	0.99.4	
libfdisk	2.30.2	2.37.4
libffi	3.0.13	3.4.4
libfido2		1.10.0
libgcc	7.3.1	11.4.1
libgcrypt	1.5.3	1.10.2
libgomp	7.3.1	11.4.1
libgpg-error	1.12	1.42
libibverbs		48.0
libicu	50.2	
libidn	1.28	
libidn2	2.3.0	2.3.2
libini_config	1.3.1	1.3.1
libjpeg-turbo	2.0.90	
libkcapi		1.4.0
libkcapi-hmacalc		1.4.0
libldb		2.6.2
libmaxminddb		1.5.2

Paquete	AMI de AL2	AMI de AL2023
libmetalink	0.1.3	0.1.3
libmnl	1.0.3	1.0.4
libmodulemd		2.13.0
libmount	2.30.2	2.37.4
libnetfilter_connt rack	1.0.6	
libnfnetworklink	1.0.1	
libnfsidmap	0,25	2.5.4
libnhttp2	1.41.0	1.59.0
libnl3	3.2.28	3.5.0
libnl3-cli	3.2.28	
libpath_utils	0.2.1	0.2.1
libpcap	1.5.3	1.10.1
libpciaccess	0.14 (x86_64)	
libpipeline	1.2.3	1.5.3
libpkgconf		1.8.0
libpng	1.5.13	
libpsl	0.21.5	0.21.1
libpwquality	1.2.3	1.4.4
libref_array	0.1.5	0.1.5
librepo		1.14.5

Paquete	AMI de AL2	AMI de AL2023
libreport-filessystem		2.15.2
libseccomp	2.5.2	2.5.3
libselinux	2,5	3.4
libselinux-utils	2,5	3.4
libsemanage	2,5	3.4
libsepol	2,5	3.4
libsigsegv		2.13
libsmartcols	2.30.2	2.37.4
libsolv		0.7.22
libss	1.42.9	1.46.5
libssh2	1.4.3	
libsss_certmap		2.9.4
libsss_idmap	1.16.5	2.9.4
libsss_nss_idmap	1.16.5	2.9.4
libsss_sudo		2.9.4
libstdc++	7.3.1	11.4.1
libstoragemgmt	1.6.1	1.9.4
libstoragemgmt-python	1.6.1	
libstoragemgmt-python-clibs	1.6.1	

Paquete	AMI de AL2	AMI de AL2023
libsysfs	2.1.0	
libtalloc		2.3.4
libtasn1	4.10	4.19.0
libtdb		1.4.7
libteam	1.27	
libtevent		0.13.0
libtextstyle		0.21
libtiff	4.0.3	
libtirpc	0.2.4	1.3.3
libunistring	0.9.3	0.9.10
libuser	0.60	0.63
libutempter	1.1.6	1.2.1
libuuid	2.30.2	2.37.4
libuv		1.47.0
libverto	0.2.5	0.3.2
libverto-libev		0.3.2
libverto-libevent	0.2.5	
libwebp	0.3.0	
libxcrypt		4.4.33
libxml2	2.9.1	2.10.4

Paquete	AMI de AL2	AMI de AL2023
libxml2-python	2.9.1	
libyaml	0.1.4	0.2.5
libzstd		1.5.5
linux-firmware-whe nce		20210208 (noarch)
lm_sensors-libs	3.4.0	3.6.0
lmdb-libs		0.9.29
logrotate	3.8.6	3.20.1
lsof	4.87	4.94.0
lua	5.1.4	
lua-libs		5.4.4
lua-srpm-macros		1
lvm2	2.02.187	
lvm2-libs	2.02.187	
lz4	1.7.5	
lz4-libs		1.9.4
make	3.82	
man-db	2.6.3	2.9.3
man-pages	3.53	5.10
man-pages-overrides	7.5.2	
mariadb-libs	5.5.68	

Paquete	AMI de AL2	AMI de AL2023
mdadm	4.0	
microcode_ctl	2.1 (x86_64)	2.1 (x86_64)
mlocate	0.26	
mpfr		4.1.0
mtr	0.92	
nano	2.9.8	5.8
ncurses	6.0	6.2
ncurses-base	6.0	6.2
ncurses-libs	6.0	6.2
nettle	2.7.1	3.8
net-tools	2.0	2.0
newt	0.52.15	0.52.21
newt-python	0.52.15	
nfs-utils	1.3.0	2.5.4
npth		1.6
nspr	4.35.0	4.35.0
nss	3.90.0	3.90.0
nss-pem	1.0.3	
nss-softokn	3.90.0	3.90.0
nss-softokn-freebl	3.90.0	3.90.0

Paquete	AMI de AL2	AMI de AL2023
nss-sysinit	3.90.0	3.90.0
nss-tools	3.90.0	
nss-util	3.90.0	3.90.0
ntsysv	1.7.4	1.15
numactl-libs	2.0.9	2.0.14
ocaml-srpm-macros		6
oniguruma		6.9.7.1
openblas-srpm-macros		2
openldap	2.4.44	2.4.57
openssh	7.4p1	8.7p1
openssh-clients	7.4p1	8.7p1
openssh-server	7.4p1	8.7p1
openssl	1.0.2k	3.0.8
openssl-libs	1.0.2k	3.0.8
openssl-pkcs11		0.4.12
os-prober	1.58	1.77
p11-kit	0.23.22	0.24.1
p11-kit-trust	0.23.22	0.24.1
package-notes-srpm-macros		0.4
pam	1.1.8	1.5.1

Paquete	AMI de AL2	AMI de AL2023
parted	3.1	3.4
passwd	0.79	0.80
pciutils	3.5.1	3.7.0
pciutils-libs	3.5.1	3.7.0
pcre	8.32	
pcre2	10.23	10.40
pcre2-syntax		10.40
perl	5.16.3	
perl-Carp	1.26	1.50
perl-Class-Struct		0.66
perl-constant	1.27	1.33
perl-DynaLoader		1.47
perl-Encode	2.51	3.15
perl-Errno		1.30
perl-Exporter	5.68	5.74
perl-Fcntl		1.13
perl-File-Basename		2.85
perl-File-Path	2.09	2.18
perl-File-stat		1.09
perl-File-Temp	0.23.01	0.231.100

Paquete	AMI de AL2	AMI de AL2023
perl-Filter	1.49	
perl-Getopt-Long	2.40	2.52
perl-Getopt-Std		1.12
perl-HTTP-Tiny	0.033	0.078
perl-if		0.60.800
perl-interpreter		5.32.1
perl-IO		1.43
perl-IPC-Open3		1.21
perl-libs	5.16.3	5.32.1
perl-macros	5.16.3	
perl-MIME-Base64		3.16
perl-mro		1.23
perl-overload		1.31
perl-overloading		0.02
perl-parent	0.225	0.238
perl-PathTools	3.40	3.78
perl-Pod-Escapes	1.04	1.07
perl-podlators	2.5.1	4.14
perl-Pod-Perldoc	3.20	3.28.01
perl-Pod-Simple	3.28	3.42

Paquete	AMI de AL2	AMI de AL2023
perl-Pod-Usage	1.63	2.01
perl-POSIX		1.94
perl-Scalar-List-Utils	1.27	1.56
perl-SelectSaver		1.02
perl-Socket	2.010	2.032
perl-srpm-macros		1
perl-Storable	2.45	3.21
perl-subs		1.03
perl-Symbol		1.08
perl-Term-ANSIColor		5.01
perl-Term-Cap		1,17
perl-Text-ParseWords	3.29	3.30
perl-Text-Tabs+Wrap		2021.0726
perl-threads	1.87	
perl-threads-shared	1.43	
perl-Time-HiRes	1.9725	
perl-Time-Local	1.2300	1.300
perl-vars		1.05
pinentry	0.8.1	
pkgconf		1.8.0

Paquete	AMI de AL2	AMI de AL2023
pkgconfig	0.27.1	
pkgconf-m4		1.8.0
pkgconf-pkg-config		1.8.0
plymouth	0.8.9	
plymouth-core-libs	0.8.9	
plymouth-scripts	0.8.9	
pm-utils	1.4.1	
polycoreutils	2.5	3.4
polycoreutils-python-utils		3.4
popt	1.13	1.18
postfix	2.10.1	
procps-ng	3.3.10	3.3.17
protobuf-c		1.4.1
psacct	6.6.1	6.6.4
psmisc	22.20	23.4
pth	2.0.7	
publicsuffix-list-dafsa	20240208	20240212
pygpgme	0.3	
pyliblzma	0.5.3	

Paquete	AMI de AL2	AMI de AL2023
pystache	0.5.3	
python	2.7.18	
python2-botocore	1.18.6	
python2-colorama	0.3.9	
python2-cryptography	1.7.2	
python2-dateutil	2.6.1	
python2-futures	3.0.5	
python2-jmespath	0.9.3	
python2-jsonschema	2.5.1	
python2-oauthlib	2.0.1	
python2-pyasn1	0.1.9	
python2-rpm	4.11.3	
python2-rsa	3.4.1	
python2-s3transfer	0.3.3	
python2-setuptools	41.2.0	
python2-six	1.11.0	
python3	3.7.16	3.9.16
python3-attrs		20.3.0
python3-audit		3.0.6
python3-awscrt		0.19.19

Paquete	AMI de AL2	AMI de AL2023
python3-babel		2.9.1
python3-cffi		1.14.5
python3-chardet		4.0.0
python3-colorama		0.4.4
python3-configobj		5.0.6
python3-cryptography		36.0.1
python3-daemon	2.2.3	2.3.0
python3-dateutil		2.8.1
python3-dbus		1.2.18
python3-distro		1.5.0
python3-dnf		4.14.0
python3-dnf-plugins-core		4.3.0
python3-docutils	0.14	0.16
python3-gpg		1.15.1
python3-hawkey		0.69.0
python3-idna		(2.10)
python3-jinja2		2.11.3
python3-jmespath		0.10.0
python3-jsonpatch		1.21
python3-jsonpointer		2.0

Paquete	AMI de AL2	AMI de AL2023
python3-jsonschema		3.2.0
python3-libcomps		0.1.20
python3-libdnf		0.69.0
python3-libs	3.7.16	3.9.16
python3-libselinux		3.4
python3-libsemanage		3.4
python3-libstorage mgmt		1.9.4
python3-lockfile	0.11.0	0.12.2
python3-markupsafe		1.1.1
python3-netifaces		0.10.6
python3-oauthlib		3.0.2
python3-pip	20.2.2	
python3-pip-wheel		21.3.1
python3-ply		3.11
python3-policycore utils		3.4
python3-prettytable		0.7.2
python3-prompt-too lkit		3.0.24
python3-pycparser		2.20

Paquete	AMI de AL2	AMI de AL2023
python3-pyrsistent		0.17.3
python3-pyserial		3.4
python3-pysocks		1.7.1
python3-pystache	0.5.4	
python3-pytz		2022.7.1
python3-pyyaml		5.4.1
python3-requests		2.25.1
python3-rpm		4.16.1.3
python3-ruamel-yaml		0.16.6
python3-ruamel-yaml-clib		0.1.2
python3-setools		4.4.1
python3-setuptools	49.1.3	59.6.0
python3-setuptools-wheel		59.6.0
python3-simplejson	3.2.0	
python3-six		1.15.0
python3-systemd		235
python3-urllib3		1.25.10
python3-wcwidth		0.2.5
python-babel	0.9.6	

Paquete	AMI de AL2	AMI de AL2023
python-backports	1.0	
python-backports-s sl_match_hostname	3.5.0.1	
python-cffi	1.6.0	
python-chardet	2.2.1	
python-chevron		0.13.1
python-configobj	4.7.2	
python-daemon	1.6	
python-devel	2.7.18	
python-docutils	0.12	
python-enum34	1.0.4	
python-idna	2.4	
python-iniparse	0.4	
python-ipaddress	1.0.16	
python-jinja2	2.7.2	
python-jsonpatch	1.2	
python-jsonpointer	1.9	
python-jwcrypto	0.4.2	
python-kitchen	1.1.1	
python-libs	2.7.18	
python-lockfile	0.9.1	

Paquete	AMI de AL2	AMI de AL2023
python-markupsafe	0.11	
python-pillow	2.0.0	
python-ply	3.4	
python-pycparser	2.14	
python-pycurl	7.19.0	
python-repoze-lru	0.4	
python-requests	2.6.0	
python-simplejson	3.2.0	
python-srpm-macros		3.9
python-urlgrabber	3.10	
python-urllib3	1.25.9	
pyxattr	0.5.1	
PyYAML	3.10	
qrencode-libs	3.4.1	
quota	4.01	4.06
quota-nls	4.01	4.06
rdate	1.4	
readline	6.2	8.1
rng-tools	6.8	6.14
rootfiles	8.1	8.1

Paquete	AMI de AL2	AMI de AL2023
rpcbind	0.2.0	1.2.6
rpm	4.11.3	4.16.1.3
rpm-build-libs	4.11.3	4.16.1.3
rpm-libs	4.11.3	4.16.1.3
rpm-plugin-selinux		4.16.1.3
rpm-plugin-systemd-inhibit	4.11.3	4.16.1.3
rpm-sign-libs		4.16.1.3
rsync	3.1.2	3.2.6
rsyslog	8.24.0	
rust-srpm-macros		21
sbsigntools		0.9.4
scl-utils	20130529	
screen	4.1.0	4.8.0
sed	4.2.2	4.8
selinux-policy	3.13.1	38.1.45
selinux-policy-targeted	3.13.1	38.1.45
setserial	2.17	
setup	2.8.71	2.13.7
setuptools	1.19.11	

Paquete	AMI de AL2	AMI de AL2023
sgpio	1.2.0.10	
shadow-utils	4.1.5.1	4.9
shared-mime-info	1.8	
slang	2.2.4	2.3.2
sqlite	3.7.17	
sqlite-libs		3.40.0
sssd-client	1.16.5	2.9.4
sssd-common		2.9.4
sssd-kcm		2.9.4
sssd-nfs-idmap		2.9.4
strace	4.26	6.8
sudo	1.8.23	1.9.15
sysctl-defaults	1.0	1.0
sysstat	10.1.5	12.5.6
systemd	219	252.23
systemd-libs	219	252.23
systemd-networkd		252.23
systemd-pam		252.23
systemd-resolved		252.23
systemd-sysv	219	

Paquete	AMI de AL2	AMI de AL2023
systemd-udev		252.23
system-release	2	2023.6.20241031
systemtap-runtime	4.5	4.8
sysvinit-tools	2.88	
tar	1.26	1.34
tbb		2020.3
tcp_wrappers	7.6	
tcp_wrappers-libs	7.6	
tcpdump	4.9.2	4.99.1
tcsh	6.18.01	6.24.07
teamd	1.27	
time	1.7	1.9
traceroute	2.0.22	2.1.3
tzdata	2024a	2024a
unzip	6.0	6.0
update-motd	1.1.2	2.2
usermode	1.111	
userspace-rcu		0.12.1
ustr	1.0.4	
util-linux	2.30.2	2.37.4

Paquete	AMI de AL2	AMI de AL2023
util-linux-core		2.37.4
vim-common	9.0.2153	9.0.2153
vim-data	9.0.2153	9.0.2153
vim-enhanced	9.0.2153	9.0.2153
vim-filesystem	9.0.2153	9.0.2153
vim-minimal	9.0.2153	9.0.2153
virt-what	1.18	
wget	1.14	1.21.3
which	2.20	2.21
words	3.0	3.0
xfsdump	3.1.8	3.1.11
xfsprogs	5.0.0	5.18.0
xxd	9.0.2153	9.0.2153
xxhash-libs		0.8.0
xz	5.2.2	5.2.5
xz-libs	5.2.2	5.2.5
yajl	2.0.4	
yum	3.4.3	4.14.0
yum-langpacks	0.4.2	
yum-metadata-parser	1.1.4	

Paquete	AMI de AL2	AMI de AL2023
yum-plugin-priorities	1.1.31	
yum-utils	1.1.31	
zip	3.0	3.0
zlib	1.2.7	1.2.11
zram-generator		1.1.2
zram-generator-defaults		1.1.2
zstd		1.5.5

Comparación de paquetes instalados en las AMI de Amazon Linux 2 y Amazon Linux 2023

Comparación de las RPM presentes en las AMI mínimas de Amazon Linux 2 y AL2023.

Paquete	AL2 mínimo	AL2023 mínimo
acl	2.2.51	
alternatives		1.15
amazon-chrony-config		4.3
amazon-ec2-net-utils		2.5.1
amazon-linux-extras	2.0.3	
amazon-linux-repo-s3		2023.6.20241031
amazon-linux-sb-keys		2023.1

Paquete	AL2 mínimo	AL2023 mínimo
amd-ucode-firmware	20200421 (noarch)	20210208 (noarch)
audit	2.8.1	3.0.6
audit-libs	2.8.1	3.0.6
authconfig	6.2.8	
awscli-2		2.15.30
basesystem	10.0	11
bash	4.2.46	5.2.15
bind-export-libs	9.11.4	
bzip2-libs	1.0.6	1.0.8
ca-certificates	2023.2.68	2023.2.68
checkpolicy		3.4
chkconfig	1.7.4	
chrony	4.2	4.3
cloud-init	19.3	22.2.2
cloud-init-cfg-ec2		22.2.2
cloud-utils-growpart	0.31	0.31
coreutils	8.22	8.32
coreutils-common		8.32
cpio	2.12	2.13
cracklib	2.9.0	2.9.6

Paquete	AL2 mínimo	AL2023 mínimo
cracklib-dicts	2.9.0	2.9.6
cronie	1.4.11	
cronie-anacron	1.4.11	
crontabs	1.11	
crypto-policies		20220428
cryptsetup-libs	1.7.4	2.6.1
curl	8.3.0	
curl-minimal		8.5.0
cyrus-sasl-lib	2.1.26	2.1.27
dbus	1.10.24	1.12.28
dbus-broker		32
dbus-common		1.12.28
dbus-libs	1.10.24	1.12.28
device-mapper	1.02.170	1.02.185
device-mapper-libs	1.02.170	1.02.185
dhclient	4.2.5	
dhcp-common	4.2.5	
dhcp-libs	4.2.5	
diffutils	3.3	3.8
dnf		4.14.0

Paquete	AL2 mínimo	AL2023 mínimo
dnf-data		4.14.0
dnf-plugin-release-notification		1.2
dnf-plugins-core		4.3.0
dnf-plugin-support-info		1.2
dracut	033	055
dracut-config-ec2	2.0	3.0
dracut-config-generic	033	055
e2fsprogs	1.42.9	1.46.5
e2fsprogs-libs	1.42.9	1.46.5
ec2-utils	1.2	2.2.0
efibootmgr	15 (aarch64)	
efi-filesystem		5
efivar		38
efivar-libs	31 (aarch64)	38
elfutils-default-yama-scope	0.176	0.188
elfutils-libelf	0.176	0.188
elfutils-libs	0.176	0.188
expat	2.1.0	2.5.0

Paquete	AL2 mínimo	AL2023 mínimo
file	5.11	5.39
file-libs	5.11	5.39
filesystem	3.2	3.14
findutils	4.5.11	4.8.0
fipscheck	1.4.1	
fipscheck-lib	1.4.1	
freetype	2.8	
fuse-libs	2.9.2	2.9.9
gawk	4.0.2	5.1.0
gdbm	1.13	
gdbm-libs		1.19
gdisk	0.8.10	1.0.8
gettext	0.19.8.1	0.21
gettext-libs	0.19.8.1	0.21
glib2	2.56.1	2.74.7
glibc	2.26	2.34
glibc-all-langpacks	2.26	2.34
glibc-common	2.26	2.34
glibc-locale-source	2.26	2.34
glibc-minimal-lang pack	2.26	

Paquete	AL2 mínimo	AL2023 mínimo
gmp	6.0.0	6.2.1
gnupg2	2.0.22	
gnupg2-minimal		2.3.7
gnutls		3.8.0
gpgme	1.3.2	1.15.1
grep	2.20	3.8
groff-base	1.22.2	1.22.4
grub2	2.06	
grub2-common	2.06	2.06
grub2-efi-aa64	2.06 (aarch64)	
grub2-efi-aa64-ec2	2.06 (aarch64)	2.06 (aarch64)
grub2-efi-aa64-modules	2.06 (noarch)	
grub2-efi-x64-ec2	2.06 (x86_64)	2.06 (x86_64)
grub2-pc	2.06 (x86_64)	
grub2-pc-modules	2.06 (noarch)	2.06
grub2-tools	2.06	2.06
grub2-tools-minimal	2.06	2.06
grubby	8.28	8.40
gzip	1.5	1.12
hardlink	1.3	

Paquete	AL2 mínimo	AL2023 mínimo
hostname	3.13	3.23
hwdata		0.384
info	5.1	
inih		49
initscripts	9.49.47	10.09
iproute	5.10.0	6.10.0
iptables	1.8.4	
iptables-lib	1.8.4	
iputils	20180629	20210202
irqbalance	1.7.0	1.9.0
jansson		2.14
jitterentropy		3.4.1
jq		1.7.1
json-c		0.14
kbd		2.4.0
kbd-misc		2.4.0
kernel	4.14.355	6.1.112
kernel-libbpf		6.1.112
kernel-livepatch-r epo-s3		2023.6.20241031
keyutils-lib	1.5.8	1.6.3

Paquete	AL2 mínimo	AL2023 mínimo
kmod	25	29
kmod-libs	25	29
kpartx	0.4.9	
krb5-libs	1.15.1	1.21.3
less	458	608
libacl	2.2.51	2.3.1
libarchive		3.7.4
libargon2		20171227
libassuan	2.1.0	2.5.5
libattr	2.4.46	2.5.1
libblkid	2.30.2	2.37.4
libcap	2.54	2.48
libcap-ng	0.7.5	0.8.2
libcbor		0.7.0
libcom_err	1.42.9	1.46.5
libcomps		0.1.20
libcroco	0.6.12	
libcrypt	2.26	
libcurl	8.3.0	
libcurl-minimal		8.5.0

Paquete	AL2 mínimo	AL2023 mínimo
libdb	5.3.21	5.3.28
libdb-utils	5.3.21	
libdnf		0.69.0
libeconf		0.4.0
libedit	3.0	3.1
libestr	0.1.9	
libfastjson	0.99.4	
libfdisk	2.30.2	2.37.4
libffi	3.0.13	3.4.4
libfido2		1.10.0
libgcc	7.3.1	11.4.1
libgcrypt	1.5.3	1.10.2
libgomp	7.3.1	11.4.1
libgpg-error	1.12	1.42
libicu	50.2	
libidn	1.28	
libidn2	2.3.0	2.3.2
libkcapi		1.4.0
libkcapi-hmacalc		1.4.0
libmetalink	0.1.3	

Paquete	AL2 mínimo	AL2023 mínimo
libmnl	1.0.3	1.0.4
libmodulemd		2.13.0
libmount	2.30.2	2.37.4
libnetfilter_conntrack	1.0.6	
libnfnetlink	1.0.1	
libnhttp2	1.41.0	1.59.0
libpcap	1.5.3	
libpipeline	1.2.3	1.5.3
libpng	1.5.13	
libpsl	0.21.5	0.21.1
libpwquality	1.2.3	1.4.4
librepo		1.14.5
libreport-filesystem		2.15.2
libseccomp	2.5.2	2.5.3
libselinux	2,5	3.4
libselinux-utils	2,5	3.4
libsemanage	2,5	3.4
libsepol	2,5	3.4
libsigsegv		2.13
libsmartcols	2.30.2	2.37.4

Paquete	AL2 mínimo	AL2023 mínimo
libsolv		0.7.22
libss	1.42.9	1.46.5
libssh2	1.4.3	
libstdc++	7.3.1	11.4.1
libsysfs	2.1.0	
libtasn1	4.10	4.19.0
libtextstyle		0.21
libunistring	0.9.3	0.9.10
libuser	0.60	0.63
libutempter	1.1.6	1.2.1
libuuid	2.30.2	2.37.4
libverto	0.2.5	0.3.2
libxcrypt		4.4.33
libxml2	2.9.1	2.10.4
libyaml	0.1.4	0.2.5
libzstd		1.5.5
linux-firmware-whe nce		20210208 (noarch)
logrotate	3.8.6	3.20.1
lua	5.1.4	
lua-libs		5.4.4

Paquete	AL2 mínimo	AL2023 mínimo
lz4	1.7.5	
lz4-libs		1.9.4
make	3.82	
man-db	2.6.3	2.9.3
mariadb-libs	5.5.68	
microcode_ctl	2.1 (x86_64)	2.1 (x86_64)
mpfr		4.1.0
ncurses	6.0	6.2
ncurses-base	6.0	6.2
ncurses-libs	6.0	6.2
nettle	2.7.1	3.8
net-tools	2.0	2.0
newt	0.52.15	
newt-python	0.52.15	
npth		1.6
nspr	4.35.0	
nss	3.90.0	
nss-pem	1.0.3	
nss-softokn	3.90.0	
nss-softokn-freebl	3.90.0	

Paquete	AL2 mínimo	AL2023 mínimo
nss-sysinit	3.90.0	
nss-tools	3.90.0	
nss-util	3.90.0	
numactl-libs	2.0.9	2.0.14
oniguruma		6.9.7.1
openldap	2.4.44	2.4.57
openssh	7.4p1	8.7p1
openssh-clients	7.4p1	8.7p1
openssh-server	7.4p1	8.7p1
openssl	1.0.2k	3.0.8
openssl-libs	1.0.2k	3.0.8
openssl-pkcs11		0.4.12
os-prober	1.58	1.77
p11-kit	0.23.22	0.24.1
p11-kit-trust	0.23.22	0.24.1
pam	1.1.8	1.5.1
passwd	0.79	0.80
pciutils		3.7.0
pciutils-libs		3.7.0
pcre	8.32	

Paquete	AL2 mínimo	AL2023 mínimo
pcr2	10.23	10.40
pcr2-syntax		10.40
pinentry	0.8.1	
pkgconfig	0.27.1	
policycoreutils	2.5	3.4
popt	1.13	1.18
postfix	2.10.1	
procps-ng	3.3.10	3.3.17
psmisc	22.20	23.4
pth	2.0.7	
publicsuffix-list-dafsa	20240208	20240212
pygpgme	0.3	
pyliblzma	0.5.3	
python	2.7.18	
python2-cryptography	1.7.2	
python2-jsonschema	2.5.1	
python2-oauthlib	2.0.1	
python2-pyasn1	0.1.9	
python2-rpm	4.11.3	
python2-setuptools	41.2.0	

Paquete	AL2 mínimo	AL2023 mínimo
python2-six	1.11.0	
python3		3.9.16
python3-attrs		20.3.0
python3-audit		3.0.6
python3-awscrt		0.19.19
python3-babel		2.9.1
python3-cffi		1.14.5
python3-chardet		4.0.0
python3-colorama		0.4.4
python3-configobj		5.0.6
python3-cryptography		36.0.1
python3-dateutil		2.8.1
python3-dbus		1.2.18
python3-distro		1.5.0
python3-dnf		4.14.0
python3-dnf-plugins-core		4.3.0
python3-docutils		0.16
python3-gpg		1.15.1
python3-hawkey		0.69.0
python3-idna		(2.10)

Paquete	AL2 mínimo	AL2023 mínimo
python3-jinja2		2.11.3
python3-jmespath		0.10.0
python3-jsonpatch		1.21
python3-jsonpointer		2.0
python3-jsonschema		3.2.0
python3-libcomps		0.1.20
python3-libdnf		0.69.0
python3-libs		3.9.16
python3-libselenium		3.4
python3-libsemanage		3.4
python3-markupsafe		1.1.1
python3-netifaces		0.10.6
python3-oauthlib		3.0.2
python3-pip-wheel		21.3.1
python3-ply		3.11
python3-policycore utils		3.4
python3-prettytable		0.7.2
python3-prompt-toolkit		3.0.24
python3-pycompiler		2.20

Paquete	AL2 mínimo	AL2023 mínimo
python3-pyrsistent		0.17.3
python3-pyserial		3.4
python3-pysocks		1.7.1
python3-pytz		2022.7.1
python3-pyyaml		5.4.1
python3-requests		2.25.1
python3-rpm		4.16.1.3
python3-ruamel-yaml		0.16.6
python3-ruamel-yaml-clib		0.1.2
python3-setools		4.4.1
python3-setuptools		59.6.0
python3-setuptools-wheel		59.6.0
python3-six		1.15.0
python3-systemd		235
python3-urllib3		1.25.10
python3-wcwidth		0.2.5
python-babel	0.9.6	
python-backports	1.0	

Paquete	AL2 mínimo	AL2023 mínimo
python-backports-s sl_match_hostname	3.5.0.1	
python-cffi	1.6.0	
python-chardet	2.2.1	
python-configobj	4.7.2	
python-devel	2.7.18	
python-enum34	1.0.4	
python-idna	2.4	
python-iniparse	0.4	
python-ipaddress	1.0.16	
python-jinja2	2.7.2	
python-jsonpatch	1.2	
python-jsonpointer	1.9	
python-jwcrypto	0.4.2	
python-libs	2.7.18	
python-markupsafe	0.11	
python-ply	3.4	
python-pycparser	2.14	
python-pycurl	7.19.0	
python-repoze-lru	0.4	
python-requests	2.6.0	

Paquete	AL2 mínimo	AL2023 mínimo
python-urlgrabber	3.10	
python-urllib3	1.25.9	
pyxattr	0.5.1	
PyYAML	3.10	
qrencode-libs	3.4.1	
readline	6.2	8.1
rng-tools	6.8	6.14
rootfiles	8.1	8.1
rpm	4.11.3	4.16.1.3
rpm-build-libs	4.11.3	4.16.1.3
rpm-libs	4.11.3	4.16.1.3
rpm-plugin-selinux		4.16.1.3
rpm-plugin-systemd-inhibit	4.11.3	4.16.1.3
rpm-sign-libs		4.16.1.3
rsyslog	8.24.0	
sbsigntools		0.9.4
sed	4.2.2	4.8
selinux-policy	3.13.1	38.1.45
selinux-policy-targeted	3.13.1	38.1.45

Paquete	AL2 mínimo	AL2023 mínimo
setup	2.8.71	2.13.7
shadow-utils	4.1.5.1	4.9
shared-mime-info	1.8	
slang	2.2.4	
sqlite	3.7.17	
sqlite-libs		3.40.0
sudo	1.8.23	1.9.15
sysctl-defaults	1.0	1.0
systemd	219	252.23
systemd-libs	219	252.23
systemd-networkd		252.23
systemd-pam		252.23
systemd-resolved		252.23
systemd-sysv	219	
systemd-udev		252.23
system-release	2	2023.6.20241031
sysvinit-tools	2.88	
tar	1.26	1.34
tcp_wrappers-libs	7.6	
tzdata	2024a	2024a

Paquete	AL2 mínimo	AL2023 mínimo
update-motd	1.1.2	2.2
userspace-rcu		0.12.1
ustr	1.0.4	
util-linux	2.30.2	2.37.4
util-linux-core		2.37.4
vim-data	9.0.2153	9.0.2153
vim-minimal	9.0.2153	9.0.2153
which	2.20	2.21
xfspgrog	5.0.0	5.18.0
xz	5.2.2	5.2.5
xz-libs	5.2.2	5.2.5
yum	3.4.3	4.14.0
yum-metadata-parser	1.1.4	
yum-plugin-priorities	1.1.31	
zlib	1.2.7	1.2.11
zram-generator		1.1.2
zram-generator-defaults		1.1.2
zstd		1.5.5

Comparación de paquetes instalados en imágenes de contenedor base de Amazon Linux 2 y Amazon Linux 2023

Comparación de los RPM presentes en las imágenes de contenedor base de Amazon Linux 2 y AL2023.

Paquete	Contenedor de AL2	Contenedor de AL2023
alternatives		1.15
amazon-linux-extras	2.0.3	
amazon-linux-repo-cdn		2023.6.20241031
audit-libs		3.0.6
basesystem	10.0	11
bash	4.2.46	5.2.15
bzip2-libs	1.0.6	1.0.8
ca-certificates	2023.2.68	2023.2.68
chkconfig	1.7.4	
coreutils	8.22	
coreutils-single		8.32
cpio	2.12	
crypto-policies		20220428
curl	8.3.0	
curl-minimal		8.5.0
cyrus-sasl-lib	2.1.26	

Paquete	Contenedor de AL2	Contenedor de AL2023
diffutils	3.3	
dnf		4.14.0
dnf-data		4.14.0
elfutils-default-yama-scope		0.188
elfutils-libelf	0.176	0.188
elfutils-libs		0.188
expat	2.1.0	2.5.0
file-libs	5.11	5.39
filesystem	3.2	3.14
findutils	4.5.11	
gawk	4.0.2	5.1.0
gdbm	1.13	
gdbm-libs		1.19
glib2	2.56.1	2.74.7
glibc	2.26	2.34
glibc-common	2.26	2.34
glibc-langpack-en	2.26	
glibc-minimal-langpack	2.26	2.34
gmp	6.0.0	6.2.1

Paquete	Contenedor de AL2	Contenedor de AL2023
gnupg2	2.0.22	
gnupg2-minimal		2.3.7
gpgme	1.3.2	1.15.1
grep	2.20	3.8
info	5.1	
json-c		0.14
keyutils-libs	1.5.8	1.6.3
krb5-libs	1.15.1	1.21.3
libacl	2.2.51	2.3.1
libarchive		3.7.4
libassuan	2.1.0	2.5.5
libattr	2.4.46	2.5.1
libblkid	2.30.2	2.37.4
libcap	2.54	2.48
libcap-ng		0.8.2
libcom_err	1.42.9	1.46.5
libcomps		0.1.20
libcrypt	2.26	
libcurl	8.3.0	
libcurl-minimal		8.5.0

Paquete	Contenedor de AL2	Contenedor de AL2023
libdb	5.3.21	
libdb-utils	5.3.21	
libdnf		0.69.0
libffi	3.0.13	3.4.4
libgcc	7.3.1	11.4.1
libgcrypt	1.5.3	1.10.2
libgomp		11.4.1
libgpg-error	1.12	1.42
libidn2	2.3.0	2.3.2
libmetalink	0.1.3	
libmodulemd		2.13.0
libmount	2.30.2	2.37.4
libnghttp2	1.41.0	1.59.0
libpsl	0.21.5	0.21.1
librepo		1.14.5
libreport-filessystem		2.15.2
libselinux	2,5	3.4
libsepol	2,5	3.4
libsigsegv		2.13
libsmartcols		2.37.4

Paquete	Contenedor de AL2	Contenedor de AL2023
libsolv		0.7.22
libssh2	1.4.3	
libstdc++	7.3.1	11.4.1
libtasn1	4.10	4.19.0
libunistring	0.9.3	0.9.10
libuuid	2.30.2	2.37.4
libverto	0.2.5	0.3.2
libxcrypt		4.4.33
libxml2	2.9.1	2.10.4
libyaml		0.2.5
libzstd		1.5.5
lua	5.1.4	
lua-libs		5.4.4
lz4-libs		1.9.4
mpfr		4.1.0
ncurses	6.0	
ncurses-base	6.0	6.2
ncurses-libs	6.0	6.2
npth		1.6
nspr	4.35.0	

Paquete	Contenedor de AL2	Contenedor de AL2023
nss	3.90.0	
nss-pem	1.0.3	
nss-softokn	3.90.0	
nss-softokn-freebl	3.90.0	
nss-sysinit	3.90.0	
nss-tools	3.90.0	
nss-util	3.90.0	
openldap	2.4.44	
openssl-lib	1.0.2k	3.0.8
p11-kit	0.23.22	0.24.1
p11-kit-trust	0.23.22	0.24.1
pcre	8.32	
pcre2		10.40
pcre2-syntax		10.40
pinentry	0.8.1	
popt	1.13	1.18
pth	2.0.7	
publicsuffix-list-dafsa	20240208	20240212
pygpgme	0.3	
pyliblzma	0.5.3	

Paquete	Contenedor de AL2	Contenedor de AL2023
python	2.7.18	
python2-rpm	4.11.3	
python3		3.9.16
python3-dnf		4.14.0
python3-gpg		1.15.1
python3-hawkey		0.69.0
python3-libcomps		0.1.20
python3-libdnf		0.69.0
python3-libs		3.9.16
python3-pip-wheel		21.3.1
python3-rpm		4.16.1.3
python3-setuptools-wheel		59.6.0
python-iniparse	0.4	
python-libs	2.7.18	
python-pycurl	7.19.0	
python-urlgrabber	3.10	
pyxattr	0.5.1	
readline	6.2	8.1
rpm	4.11.3	4.16.1.3
rpm-build-libs	4.11.3	4.16.1.3

Paquete	Contenedor de AL2	Contenedor de AL2023
rpm-libs	4.11.3	4.16.1.3
rpm-sign-libs		4.16.1.3
sed	4.2.2	4.8
setup	2.8.71	2.13.7
shared-mime-info	1.8	
sqlite	3.7.17	
sqlite-libs		3.40.0
system-release	2	2023.6.20241031
tzdata	2024a	2024a
vim-data	9.0.2153	
vim-minimal	9.0.2153	
xz-libs	5.2.2	5.2.5
yum	3.4.3	4.14.0
yum-metadata-parser	1.1.4	
yum-plugin-ovl	1.1.31	
yum-plugin-priorities	1.1.31	
zlib	1.2.7	1.2.11

Comparación de AL1 y AL2023

En los siguientes temas se describen las diferencias clave entre AL1 y AL2023 que aún no se han abordado en la [comparación con AL2](#).

Note

AL1 llegó al final de su vida útil el 31 de diciembre de 2023 y no recibirá actualizaciones de seguridad ni correcciones de errores a partir del 1 de enero de 2024. Para obtener más información sobre el EOL de AL1 y el soporte de mantenimiento, consulte la publicación de blog [Actualización sobre el fin de vida útil de la AMI de Amazon Linux](#). Le recomendamos que actualice las aplicaciones a la versión de AL2023, que incluye soporte a largo plazo hasta 2028.

Temas

- [Soporte para cada versión](#)
- [systemd reemplaza upstart como sistema init](#)
- [Python 2.6 y 2.7 ha sido reemplazado por Python 3](#)
- [OpenJDK 8 como el JDK más antiguo](#)
- [AL2023 cambios en el kernel de Amazon Linux \(1AL1\)](#)
- [Comparación de paquetes instalados en las AMI de Amazon Linux 1 \(AL1\) y Amazon Linux 2023](#)
- [Comparación de paquetes instalados en las AMI mínimas de Amazon Linux 1 \(AL1\) y Amazon Linux 2023](#)
- [Comparación de paquetes instalados en imágenes de contenedor base de Amazon Linux 1 \(AL1\) y Amazon Linux 2023](#)

Soporte para cada versión

Para AL2023, ofrecemos cinco años de soporte desde la fecha de lanzamiento. AL1 finalizó el soporte estándar el 31 de diciembre de 2020 y finalizó el soporte de mantenimiento el 31 de diciembre de 2023.

Para obtener más información, consulte [Liberar cadencia](#).

systemd reemplaza **upstart** como sistema **init**

En AL2, **upstart** se reemplazó por **systemd** como sistema **init**. AL2023 también utiliza **systemd** como sistema **init**, adoptando además nuevas características y funcionalidades de **systemd**.

Python 2.6 y 2.7 ha sido reemplazado por Python 3

Si bien AL1 marcó Python 2.6 como EOL en la versión 2018.03, los paquetes aún estaban disponibles en los repositorios para su instalación. AL2 se suministró con Python 2.7 como la primera versión más antigua de Python compatible, y AL2023 completa la transición a Python 3. No se incluye ninguna versión de Python 2.x en los repositorios de AL2023.

Para obtener más información sobre Python en Amazon Linux, consulte [Python en AL2023](#).

OpenJDK 8 como el JDK más antiguo

AL2023 incluye [Amazon Corretto](#) como el kit de desarrollo de Java (JDK) predeterminado (y único). Todos los paquetes basados en Java de AL2023 están diseñados con Amazon Corretto 17.

En AL1, OpenJDK 1.6.0 (`java-1.6.0-openjdk`) finalizó su servicio con la primera versión 2018.03, y OpenJDK 1.7.0 (`java-1.7.0-openjdk`) finalizó su servicio a mediados de 2020, aunque ambas versiones estaban disponibles en los repositorios de AL1. La primera versión de OpenJDK disponible en AL2023 es OpenJDK 8, proporcionada por Amazon Corretto 8.

AL2023 cambios en el kernel de Amazon Linux (1AL1)

Kernel Live Patching

Ambos AL2023 y AL2 añaden soporte para la funcionalidad de parches en vivo del kernel. Esto le permite corregir vulnerabilidades de seguridad críticas e importantes en kernel de Linux sin necesidad de reiniciar el sistema ni perder tiempo de inactividad. Para obtener más información, consulte [Aplicación de parches activos del kernel en AL2023](#).

Compatibilidad con sistema de archivos del kernel

Se han producido varios cambios en los sistemas de archivos en los que el núcleo admite el montaje, además de cambios en los esquemas de particionamiento que AL1 analizará el núcleo.

Opción CONFIG	AL1/4.14/ x86_64	AL2023/6. 1/ aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/ aarch64	AL2023/6. 12/ x86_64	AL2023/6. 18/ aarch64	AL2023/6. 18/ x86_64
CONFIG_AVSFS	m	n	n	n	n	n	n
CONFIG_AVSFS_RRPC	m	n	n	n	n	n	n
CONFIG_BLOCK_DEVICE_LAI EL	y	n	n	n	n	n	n
CONFIG_CMAFS	m	n	n	n	n	n	n
CONFIG_CMAFS_BLOCK KDEV	N/A	N/A	N/A	N/A	N/A	N/A	N/A
CONFIG_DISK_CLONE	N/A	n	n	n	n	n	n
CONFIG_DISK_ERA	n	n	n	n	n	n	n
CONFIG_DISK_INTEGRITY	m	m	m	m	m	m	m
CONFIG_DISK_LOG_WRITES	n	m	m	m	m	m	m
CONFIG_DISK_SWITCH	n	n	n	n	n	n	n

Opción CONFIG	AL1/4.14/ x86_64	AL2023/6. 1/ aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/ aarch64	AL2023/6. 12/ x86_64	AL2023/6. 18/ aarch64	AL2023/6. 18/ x86_64
<u>CONFIG_DEBUG_KERNEL</u>	n	m	m	m	m	m	m
<u>CONFIG_CRYPTO</u>	m	n	n	n	n	n	n
<u>CONFIG_CRYPTO_FAT</u>	N/A	m	m	m	m	m	m
<u>CONFIG_CRYPTO_T2</u>	m	n	n	n	n	n	n
<u>CONFIG_CRYPTO_T3</u>	m	n	n	n	n	N/A	N/A
<u>CONFIG_GFS2</u>	n	n	n	n	n	n	n
<u>CONFIG_HFSPLUS</u>	m	n	n	n	n	n	n
<u>CONFIG_HFS</u>	m	n	n	n	n	n	n
<u>CONFIG_JFS</u>	n	n	n	n	n	n	n
<u>CONFIG_LVM2</u>	y	n	n	n	n	n	n
<u>CONFIG_MMC</u>	y	n	n	n	n	n	n

Opción CONFIG	AL1/4.14/ x86_64	AL2023/6. 1/ aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/ aarch64	AL2023/6. 12/ x86_64	AL2023/6. 18/ aarch64	AL2023/6. 18/ x86_64
CONFIG_NAMES_V2	m	n	n	n	n	n	n
CONFIG_NAMES_FS_FS	m	n	n	n	n	n	n
CONFIG_NAMES_MFS_FS	m	n	n	n	n	n	n
CONFIG_SECURITY_LARIS_X86_PARTITION	y	n	n	n	n	n	n
CONFIG_SECURITY_UASHFS_ZSTD	y	y	y	y	y	y	y
CONFIG_SECURITY_PARTITIONON	y	n	n	n	n	n	n

Cambios en la configuración del kernel centrados en la seguridad

Opción CONFIG	AL1/4,14/ x86_64	AL2023/6. 1/ aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/ aarch64	AL2023/6. 12/ x86_64	AL2023/6. 18/ aarch64	AL2023/6. 18/ x86_64
CONFIG_BUG_ON_DATA_CORRUPTION	y	y	y	y	y	y	y

Opción CONFIG	AL1/4,14/ x86_64	AL2023/6. 1/ aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/ aarch64	AL2023/6. 12/ x86_64	AL2023/6. 18/ aarch64	AL2023/6. 18/ x86_64
CONFIG_DEBUG_FAULT_MMIO P_MIN_ADDRESS	4096	65536	65536	65536	65536	65536	65536
CONFIG_DEBUG_VMEM	y	n	n	n	n	n	n
CONFIG_DEBUG_VPORT	y	n	n	n	n	n	n
CONFIG_FILTERIFY_SOURCE	y	y	y	y	y	y	y
CONFIG_HARDENED_USERCOPY_FALLBACK	N/A	N/A	N/A	N/A	N/A	N/A	N/A
CONFIG_INIT_ON_ALLOC_DEFAULT_ON	N/A	n	n	n	n	n	n
CONFIG_INIT_ON_FREE_DEFAULT_ON	N/A	n	n	n	n	n	n
CONFIG_INIT_MMU_DEFAULT_DMA_SUPPORT	N/A	n	n	n	n	n	n

Opción CONFIG	AL1/4,14/ x86_64	AL2023/6. 1/ aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/ aarch64	AL2023/6. 12/ x86_64	AL2023/6. 18/ aarch64	AL2023/6. 18/ x86_64
CONFIG_LSC_AUTOLOAD	y	n	n	n	n	n	n
CONFIG_SCHED_CORE	N/A	N/A	y	N/A	y	N/A	y
CONFIG_SCHED_STACK_END_CHECK	y	y	y	y	y	y	y
CONFIG_SECURITY_DEFAULT_RESTRICT	n	y	y	y	y	y	y
CONFIG_SECURITY_SILENT_LINUX_DISABLED	y	n	n	N/A	N/A	N/A	N/A
CONFIG_SWAPFILE_PAGE_ALLOCATION	N/A	y	y	y	y	y	y
CONFIG_SYSAB_FREEZE_HARDED	y	y	y	y	y	y	y

Opción CONFIG	AL1/4,14/ x86_64	AL2023/6. 1/ aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/ aarch64	AL2023/6. 12/ x86_64	AL2023/6. 18/ aarch64	AL2023/6. 18/ x86_64
CONFIG_SIBAB_FREELEST_RANDOM	n	y	y	y	y	y	y

Otros cambios en la configuración del kernel

Opción CONFIG	AL1/4,14/ x86_64	AL2023/6. 1/ aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/ aarch64	AL2023/6. 12/ x86_64	AL2023/6. 18/ aarch64	AL2023/6. 18/ x86_64
CONFIG_HZ	250	100	100	100	100	100	100
CONFIG_NR_CPUS	8192	4096	8192	4096	8192	4096	8192
CONFIG_PNIC_ON_OPS	n	y	y	y	y	y	y
CONFIG_PNIC_ON_OPS_VALUE	0	1	1	1	1	N/A	N/A
CONFIG_PIP	m	m	m	m	m	m	m
CONFIG_SIP	m	n	n	n	n	n	n
CONFIG_XIN_PV	y	N/A	n	N/A	n	N/A	n

Comparación de paquetes instalados en las AMI de Amazon Linux 1 (AL1) y Amazon Linux 2023

Comparación de las RPM presentes en las AMI estándar AL1 y AL2023.

Paquete	AMI de AL1	AMI de AL2023
acl	2.2.49	2.3.1
acpid	2.0.19	2.0.32
alsa-lib	1.0.22	
alternatives		1.15
amazon-chrony-config		4.3
amazon-ec2-net-utils		2.5.1
amazon-linux-repo-s3		2023.6.20241031
amazon-linux-sb-keys		2023.1
amazon-rpm-config		228
amazon-ssm-agent	3.2.2222.0	3.3.987.0
amd-ucode-firmware		20210208
at	3.1.10	3.1.23
attr	2.4.46	2.5.1
audit	2.6.5	3.0.6
audit-libs	2.6.5	3.0.6
authconfig	6.2.8	
aws-amitools-ec2	1.5.13	

Paquete	AMI de AL1	AMI de AL2023
aws-cfn-bootstrap	1.4	2.0
aws-cli	1.18.107	
awscli-2		2.15.30
basesystem	10.0	11
bash	4.2.46	5.2.15
bash-completion		2.11
bc	1.06.95	1.07.1
bind-libs	9.8.2	9.18.28
bind-license		9.18.28
bind-utils	9.8.2	9.18.28
binutils	2.27	2.39
boost-filesystem		1.75.0
boost-system		1.75.0
boost-thread		1.75.0
bzip2	1.0.6	1.0.8
bzip2-libs	1.0.6	1.0.8
ca-certificates	2023.2.62	2023.2.68
c-ares		1.19.1
checkpolicy	2.1.10	3.4
chkconfig	1.3.49.3	1.15

Paquete	AMI de AL1	AMI de AL2023
chrony		4.3
cloud-disk-utils	0.27	
cloud-init	0.7.6	22.2.2
cloud-init-cfg-ec2		22.2.2
cloud-utils-growpart		0.31
copy-jdk-configs	3.3	
coreutils	8.22	8.32
coreutils-common		8.32
cpio	(2.10)	2.13
cracklib	2.8.16	2.9.6
cracklib-dicts	2.8.16	2.9.6
cronie	1.4.4	
cronie-anacron	1.4.4	
crontabs	1.10	1.11
crypto-policies		20220428
crypto-policies-scripts		20220428
cryptsetup	1.6.7	2.6.1
cryptsetup-libs	1.6.7	2.6.1
curl	7.61.1	
curl-minimal		8.5.0

Paquete	AMI de AL1	AMI de AL2023
cyrus-sasl	2.1.23	
cyrus-sasl-lib	2.1.23	2.1.27
cyrus-sasl-plain	2.1.23	2.1.27
dash	0.5.5.1	
db4	4.7.25	
db4-utils	4.7.25	
dbus	1.6.12	1.12.28
dbus-broker		32
dbus-common		1.12.28
dbus-libs	1.6.12	1.12.28
dejavu-fonts-common	2,33	
dejavu-sans-fonts	2,33	
dejavu-serif-fonts	2,33	
device-mapper	1.02.135	1.02.185
device-mapper-event	1.02.135	
device-mapper-event-libs	1.02.135	
device-mapper-libs	1.02.135	1.02.185
device-mapper-persistent-data	0.6.3	
dhclient	4.1.1	

Paquete	AMI de AL1	AMI de AL2023
dhcp-common	4.1.1	
diffutils	3.3	3.8
dmraid	1.0.0.rc16	
dmraid-events	1.0.0.rc16	
dnf		4.14.0
dnf-data		4.14.0
dnf-plugin-release-notification		1.2
dnf-plugins-core		4.3.0
dnf-plugin-support-info		1.2
dnf-utils		4.3.0
dosfstools		4.2
dracut	004	055
dracut-config-ec2		3.0
dracut-config-generic		055
dracut-modules-growroot	0.20	
dump	0.4	
dwz		0.14
dyninst		10.2.1

Paquete	AMI de AL1	AMI de AL2023
e2fsprogs	1.43.5	1.46.5
e2fsprogs-libs	1.43.5	1.46.5
ec2-hibinit-agent	1.0.0	1.0.8
ec2-instance-connect		1.1
ec2-instance-connect-selinux		1.1
ec2-net-utils	0.7	
ec2-utils	0.7	2.2.0
ed	1.1	1.14.2
efi-filesystem		5
efi-srpm-macros		5
efivar		38
efivar-libs		38
elfutils-debuginfod-client		0.188
elfutils-default-yama-scope		0.188
elfutils-libelf	0.168	0.188
elfutils-libs		0.188
epel-release	6	
ethtool	3.15	5.15

Paquete	AMI de AL1	AMI de AL2023
expat	2.1.0	2.5.0
file	5.37	5.39
file-libs	5.37	5.39
filesystem	2.4.30	3.14
findutils	4.4.2	4.8.0
fipscheck	1.3.1	
fipscheck-lib	1.3.1	
fontconfig	2.8.0	
fontpackages-files ystem	1.41	
fonts-srpm-macros		2.0.5
freetype	2.3.11	
fstrm		0.6.1
fuse-libs	2.9.4	2.9.9
gawk	3.1.7	5.1.0
gdbm	1.8.0	
gdbm-libs		1.19
gdisk	0.8.10	1.0.8
generic-logos	17.0.0	
get_reference_source	1.2	
gettext		0.21

Paquete	AMI de AL1	AMI de AL2023
gettext-libs		0.21
ghc-srpm-macros		1.5.0
giflib	4.1.6	
glib2	2.36.3	2.74.7
glibc	2.17	2.34
glibc-all-langpacks		2.34
glibc-common	2.17	2.34
glibc-gconv-extra		2.34
glibc-locale-source		2.34
gmp	6.0.0	6.2.1
gnupg2	2.0.28	
gnupg2-minimal		2.3.7
gnutls		3.8.0
go-srpm-macros		3.2.0
gpgme	1.4.3	1.15.1
gpm-libs	1.20.6	1.20.7
grep	2.20	3.8
groff	1.22.2	
groff-base	1.22.2	1.22.4
grub	0.97	

Paquete	AMI de AL1	AMI de AL2023
grub2-common		2.06
grub2-efi-x64-ec2		2.06
grub2-pc-modules		2.06
grub2-tools		2.06
grub2-tools-minimal		2.06
grubby	7.0.15	8.40
gssproxy		0.8.4
gzip	1.5	1.12
hesiod	3.1.0	
hibagent	1.0.0	
hmaccalc	0.9.12	
hostname		3.23
hunspell		1.7.0
hunspell-en		0.20140811.1
hunspell-en-GB		0.20140811.1
hunspell-en-US		0.20140811.1
hunspell-filesystem		1.7.0
hwdata	0.233	0.384
info	5.1	6.7
inih		49

Paquete	AMI de AL1	AMI de AL2023
initscripts	9.03.58	10.09
iproute	4.4.0	6.10.0
iptables	1.4.21	
iputils	20121221	20210202
irqbalance	1.5.0	1.9.0
jansson		2.14
java-1.7.0-openjdk	1.7.0.321	
javapackages-tools	0.9.1	
jemalloc		5.2.1
jitterentropy		3.4.1
jpackage-utils	1.7.5	
jq		1.7.1
json-c		0.14
kbd	1.15	2.4.0
kbd-misc	1.15	2.4.0
kernel	4.14.336	6.1.112
kernel-libbpf		6.1.112
kernel-livepatch-r epo-s3		2023.6.20241031
kernel-srpm-macros		1.0
kernel-tools	4.14.336	6.1.112

Paquete	AMI de AL1	AMI de AL2023
keyutils	1.5.8	1.6.3
keyutils-libs	1.5.8	1.6.3
kmod	14	29
kmod-libs	14	29
kpartx	0.4.9	
kpatch-runtime		0.9.7
krb5-libs	1.15.1	1.21.3
lcms2	2.6	
less	436	608
libacl	2.2.49	2.3.1
libaio	0.3.109	0.3.111
libarchive		3.7.4
libargon2		20171227
libassuan	2.0.3	2.5.5
libattr	2.4.46	2.5.1
libbasicobjects		0.1.1
libblkid	2.23.2	2.37.4
libcap	2.16	2.48
libcap54	2.54	
libcap-ng	0.7.5	0.8.2

Paquete	AMI de AL1	AMI de AL2023
libcbor		0.7.0
libcgroup	0.40.rc1	
libcollection		0.7.0
libcom_err	1.43.5	1.46.5
libcomps		0.1.20
libconfig		1.7.2
libcurl	7.61.1	
libcurl-minimal		8.5.0
libdb		5.3.28
libdhash		0.5.0
libdnf		0.69.0
libeconf		0.4.0
libedit	2.11	3.1
libev		4.33
libevent	2.0.21	2.1.12
libfdisk		2.37.4
libffi	3.0.13	3.4.4
libfido2		1.10.0
libfontenc	1.0.5	
libgcc		11.4.1

Paquete	AMI de AL1	AMI de AL2023
libgcc72	7.2.1	
libgcrypt	1.5.3	1.10.2
libgomp		11.4.1
libgpg-error	1.11	1.42
libgssglue	0.1	
libibverbs		48.0
libICE	1.0.6	
libicu	50.2	
libidn	1.18	
libidn2	2.3.0	2.3.2
libini_config		1.3.1
libjpeg-turbo	1.2.90	
libkcap1		1.4.0
libkcap1-hmacalc		1.4.0
libldb		2.6.2
libmaxminddb		1.5.2
libmetalink		0.1.3
libmnl	1.0.3	1.0.4
libmodulemd		2.13.0
libmount	2.23.2	2.37.4

Paquete	AMI de AL1	AMI de AL2023
libnetfilter_contrack	1.0.4	
libnfnetlink	1.0.1	
libnfsidmap	0,25	2.5.4
libnhttp2	1.33.0	1.59.0
libnih	1.0.1	
libnl	1.1.4	
libnl3		3.5.0
libpath_utils		0.2.1
libpcap		1.10.1
libpipeline	1.2.3	1.5.3
libpkgconf		1.8.0
libpng	1.2.49	
libpsl	0.6.2	0.21.1
libpwquality	1.2.3	1.4.4
libref_array		0.1.5
librepo		1.14.5
libreport-filestore		2.15.2
libseccomp		2.5.3
libselinux	2.1.10	3.4
libselinux-utils	2.1.10	3.4

Paquete	AMI de AL1	AMI de AL2023
libsemanage	2.1.6	3.4
libsepol	2.1.7	3.4
libsigsegv		2.13
libSM	1.2.1	
libsmartcols	2.23.2	2.37.4
libsolv		0.7.22
libss	1.43.5	1.46.5
libssh2	1.4.2	
libsss_certmap		2.9.4
libsss_idmap		2.9.4
libsss_nss_idmap		2.9.4
libsss_sudo		2.9.4
libstdc++		11.4.1
libstdc++72	7.2.1	
libstoragegmt		1.9.4
libsysfs	2.1.0	
libtalloc		2.3.4
libtasn1	2.3	4.19.0
libtdb		1.4.7
libtevent		0.13.0

Paquete	AMI de AL1	AMI de AL2023
libtextstyle		0.21
libtirpc	0.2.4	1.3.3
libudev	173	
libunistring	0.9.3	0.9.10
libuser	0.60	0.63
libutempter	1.1.5	1.2.1
libuuid	2.23.2	2.37.4
libuv		1.47.0
libverto	0.2.5	0.3.2
libverto-libev		0.3.2
libX11	1.6.0	
libX11-common	1.6.0	
libXau	1.0.6	
libxcb	1.11	
libXcomposite	0.4.3	
libxcrypt		4.4.33
libXext	1.3.2	
libXfont	1.4.5	
libXi	1.7.2	
libxml2	2.9.1	2.10.4

Paquete	AMI de AL1	AMI de AL2023
libxml2-python27	2.9.1	
libXrender	0.9.8	
libxslt	1.1.28	
libXtst	1.2.2	
libyaml	0.1.6	0.2.5
libzstd		1.5.5
linux-firmware-wheel		20210208
lm_sensors-libs		3.6.0
lmdb-libs		0.9.29
log4j-cve-2021-44228-hotpatch	1.3	
logrotate	3.7.8	3.20.1
lsof	4.82	4.94.0
lua	5.1.4	
lua-libs		5.4.4
lua-srpm-macros		1
lvm2	2.02.166	
lvm2-libs	2.02.166	
lz4-libs		1.9.4
mailcap	2.1.31	

Paquete	AMI de AL1	AMI de AL2023
make	3.82	
man-db	2.6.3	2.9.3
man-pages	4.10	5.10
mdadm	3.2.6	
microcode_ctl	2.1	2.1
mingetty	1.08	
mpfr		4.1.0
nano	2.5.3	5.8
nc	1.84	
ncurses	5.7	6.2
ncurses-base	5.7	6.2
ncurses-libs	5.7	6.2
nettle		3.8
net-tools	1.60	2.0
newt	0.52.11	0.52.21
newt-python27	0.52.11	
nfs-utils	1.3.0	2.5.4
npth		1.6
nspr	4.25.0	4.35.0
nss	3.53.1	3.90.0

Paquete	AMI de AL1	AMI de AL2023
nss-pem	1.0.3	
nss-softokn	3.53.1	3.90.0
nss-softokn-freebl	3.53.1	3.90.0
nss-sysinit	3.53.1	3.90.0
nss-tools	3.53.1	
nss-util	3.53.1	3.90.0
ntp	4.2.8p15	
ntpd	4.2.8p15	
ntsysv	1.3.49.3	1.15
numactl	2.0.7	
numactl-libs		2.0.14
ocaml-srpm-macros		6
oniguruma		6.9.7.1
openblas-srpm-macros		2
openldap	2.4.40	2.4.57
openssh	7.4p1	8.7p1
openssh-clients	7.4p1	8.7p1
openssh-server	7.4p1	8.7p1
openssl	1.0.2k	3.0.8
openssl-libs		3.0.8

Paquete	AMI de AL1	AMI de AL2023
openssl-pkcs11		0.4.12
os-prober		1.77
p11-kit	0.18.5	0.24.1
p11-kit-trust	0.18.5	0.24.1
package-notes-srpm-macros		0.4
pam	1.1.8	1.5.1
pam_ccreds	10	
pam_krb5	2.3.11	
pam_passwdqc	1.0.5	
parted	2.1	3.4
passwd	0.79	0.80
pciutils	3.1.10	3.7.0
pciutils-libs	3.1.10	3.7.0
pcre	8.21	
pcre2		10.40
pcre2-syntax		10.40
perl	5.16.3	
perl-Carp	1.26	1.50
perl-Class-Struct		0.66
perl-constant	1.27	1.33

Paquete	AMI de AL1	AMI de AL2023
perl-Digest	1,17	
perl-Digest-HMAC	1.03	
perl-Digest-MD5	2.52	
perl-Digest-SHA	5.85	
perl-DynaLoader		1.47
perl-Encode	2.51	3.15
perl-Errno		1.30
perl-Exporter	5.68	5.74
perl-Fcntl		1.13
perl-File-Basename		2.85
perl-File-Path	2.09	2.18
perl-File-stat		1.09
perl-File-Temp	0.23.01	0.231.100
perl-Filter	1.49	
perl-Getopt-Long	2.40	2.52
perl-Getopt-Std		1.12
perl-HTTP-Tiny	0.033	0.078
perl-if		0.60.800
perl-interpreter		5.32.1
perl-IO		1.43

Paquete	AMI de AL1	AMI de AL2023
perl-IPC-Open3		1.21
perl-libs	5.16.3	5.32.1
perl-macros	5.16.3	
perl-MIME-Base64		3.16
perl-mro		1.23
perl-overload		1.31
perl-overloading		0.02
perl-parent	0.225	0.238
perl-PathTools	3.40	3.78
perl-Pod-Escapes	1.04	1.07
perl-podlators	2.5.1	4.14
perl-Pod-Perldoc	3.20	3.28.01
perl-Pod-Simple	3.28	3.42
perl-Pod-Usage	1.63	2.01
perl-POSIX		1.94
perl-Scalar-List-Utils	1.27	1.56
perl-SelectSaver		1.02
perl-Socket	2.010	2.032
perl-srpm-macros		1
perl-Storable	2.45	3.21

Paquete	AMI de AL1	AMI de AL2023
perl-subvars		1.03
perl-Symbol		1.08
perl-Term-ANSIColor		5.01
perl-Term-Cap		1,17
perl-Text-ParseWords	3.29	3.30
perl-Text-Tabs+Wrap		2021.0726
perl-threads	1.87	
perl-threads-shared	1.43	
perl-Time-HiRes	1.9725	
perl-Time-Local	1.2300	1.300
perl-vars		1.05
pinentry	0.7.6	
pkgconf		1.8.0
pkgconfig	0.27.1	
pkgconf-m4		1.8.0
pkgconf-pkg-config		1.8.0
pm-utils	1.4.1	
policycoreutils	2.1.12	3.4
policycoreutils-python-utils		3.4
popt	1.13	1.18

Paquete	AMI de AL1	AMI de AL2023
procmail	3.22	
procps	3.2.8	
procps-ng		3.3.17
protobuf-c		1.4.1
psacct	6.3.2	6.6.4
psmisc	22.20	23.4
pth	2.0.7	
publicsuffix-list-dafsa		20240212
python27	2.7.18	
python27-babel	0.9.4	
python27-backports	1.0	
python27-backports-ssl_match_hostname	3.4.0.2	
python27-boto	2.48.0	
python27-botocore	1.17.31	
python27-chardet	2.0.1	
python27-colorama	0.4.1	
python27-configobj	4.7.2	
python27-crypto	2.6.1	
python27-daemon	1.5.2	

Paquete	AMI de AL1	AMI de AL2023
python27-dateutil	2.1	
python27-devel	2.7.18	
python27-docutils	0.11	
python27-ecdsa	0.11	
python27-futures	3.0.3	
python27-imaging	1.1.6	
python27-iniparse	0.3.1	
python27-jinja2	2.7.2	
python27-jmespath	0.9.2	
python27-jsonpatch	1.2	
python27-jsonpointer	1.0	
python27-kitchen	1.1.1	
python27-libs	2.7.18	
python27-lockfile	0.8	
python27-markupsafe	0.11	
python27-paramiko	1.15.1	
python27-pip	9.0.3	
python27-ply	3.4	
python27-pyasn1	0.1.7	
python27-pycurl	7.19.0	

Paquete	AMI de AL1	AMI de AL2023
python27-pygpme	0.3	
python27-pyliblzma	0.5.3	
python27-pystache	0.5.3	
python27-pyxattr	0.5.0	
python27-PyYAML	3.10	
python27-requests	1.2.3	
python27-rsa	3.4.1	
python27-setuptools	36.2.7	
python27-simplejson	3.6.5	
python27-six	1.8.0	
python27-urlgrabber	3.10	
python27-urllib3	1.24.3	
python27-virtualenv	15.1.0	
python3		3.9.16
python3-attrs		20.3.0
python3-audit		3.0.6
python3-awscrt		0.19.19
python3-babel		2.9.1
python3-cffi		1.14.5
python3-chardet		4.0.0

Paquete	AMI de AL1	AMI de AL2023
python3-colorama		0.4.4
python3-configobj		5.0.6
python3-cryptography		36.0.1
python3-daemon		2.3.0
python3-dateutil		2.8.1
python3-dbus		1.2.18
python3-distro		1.5.0
python3-dnf		4.14.0
python3-dnf-plugins-core		4.3.0
python3-docutils		0.16
python3-gpg		1.15.1
python3-hawkey		0.69.0
python3-idna		(2.10)
python3-jinja2		2.11.3
python3-jmespath		0.10.0
python3-jsonpatch		1.21
python3-jsonpointer		2.0
python3-jsonschemata		3.2.0
python3-libcomps		0.1.20
python3-libdnf		0.69.0

Paquete	AMI de AL1	AMI de AL2023
python3-libs		3.9.16
python3-libselinux		3.4
python3-libsemanage		3.4
python3-libstorage mgmt		1.9.4
python3-lockfile		0.12.2
python3-markupsafe		1.1.1
python3-netifaces		0.10.6
python3-oauthlib		3.0.2
python3-pip-wheel		21.3.1
python3-ply		3.11
python3-policycore utils		3.4
python3-prettytable		0.7.2
python3-prompt-too lkit		3.0.24
python3-pycparser		2.20
python3-pyrsistent		0.17.3
python3-pyserial		3.4
python3-pysocks		1.7.1
python3-pytz		2022.7.1

Paquete	AMI de AL1	AMI de AL2023
python3-pyyaml		5.4.1
python3-requests		2.25.1
python3-rpm		4.16.1.3
python3-ruamel-yaml		0.16.6
python3-ruamel-yaml-clib		0.1.2
python3-setools		4.4.1
python3-setuptools		59.6.0
python3-setuptools-wheel		59.6.0
python3-six		1.15.0
python3-systemd		235
python3-urllib3		1.25.10
python3-wcwidth		0.2.5
python-chevron		0.13.1
python-srpm-macros		3.9
quota	4.00	4.06
quota-nls	4.00	4.06
readline	6.2	8.1
rmt	0.4	
rng-tools	5	6.14

Paquete	AMI de AL1	AMI de AL2023
rootfiles	8.1	8.1
rpcbind	0.2.0	1.2.6
rpm	4.11.3	4.16.1.3
rpm-build-libs	4.11.3	4.16.1.3
rpm-libs	4.11.3	4.16.1.3
rpm-plugin-selinux		4.16.1.3
rpm-plugin-systemd-inhibit		4.16.1.3
rpm-python27	4.11.3	
rpm-sign-libs		4.16.1.3
rsync	3.0.6	3.2.6
rsyslog	5.8.10	
ruby	2.0	
ruby20	2.0.0.648	
ruby20-irb	2.0.0.648	
ruby20-libs	2.0.0.648	
rubygem20-bigdecimal	1.2.0	
rubygem20-json	1.8.3	
rubygem20-psych	2.0.0	
rubygem20-rdoc	4.2.2	
rubygems20	2.0.14.1	

Paquete	AMI de AL1	AMI de AL2023
rust-srpm-macros		21
sbsigntools		0.9.4
screen	4.0.3	4.8.0
sed	4.2.1	4.8
selinux-policy		38.1.45
selinux-policy-targeted		38.1.45
sendmail	8.14.4	
setserial	2.17	
setup	2.8.14	2.13.7
sgpio	1.2.0.10	
shadow-utils	4.1.4.2	4.9
shared-mime-info	1.1	
slang	2.2.1	2.3.2
sqlite	3.7.17	
sqlite-libs		3.40.0
sssd-client		2.9.4
sssd-common		2.9.4
sssd-kcm		2.9.4
sssd-nfs-idmap		2.9.4
strace		6.8

Paquete	AMI de AL1	AMI de AL2023
sudo	1.8.23	1.9.15
sysctl-defaults	1.0	1.0
sysfsutils	2.1.0	
sysstat		12.5.6
systemd		252.23
systemd-libs		252.23
systemd-networkd		252.23
systemd-pam		252.23
systemd-resolved		252.23
systemd-udev		252.23
system-release	2018.03	2023.6.20241031
systemtap-runtime		4.8
sysvinit	2.87	
tar	1.26	1.34
tbb		2020.3
tcp_wrappers	7.6	
tcp_wrappers-libs	7.6	
tcpdump		4.99.1
tcsh		6.24.07
time	1.7	1.9

Paquete	AMI de AL1	AMI de AL2023
tmpwatch	2.9.16	
traceroute	2.0.14	2.1.3
ttmkfdir	3.0.9	
tzdata	2023c	2024a
tzdata-java	2023c	
udev	173	
unzip	6.0	6.0
update-motd	1.0.1	2.2
upstart	0.6.5	
userspace-rcu		0.12.1
ustr	1.0.4	
util-linux	2.23.2	2.37.4
util-linux-core		2.37.4
vim-common	9.0.2120	9.0.2153
vim-data	9.0.2120	9.0.2153
vim-enhanced	9.0.2120	9.0.2153
vim-filesystem	9.0.2120	9.0.2153
vim-minimal	9.0.2120	9.0.2153
wget	1.18	1.21.3
which	2.19	2.21

Paquete	AMI de AL1	AMI de AL2023
words	3.0	3.0
xfsdump		3.1.11
xfsprogs		5.18.0
xorg-x11-fonts-Type1	7.2	
xorg-x11-font-utils	7.2	
xxd	9.0.2120	9.0.2153
xxhash-libs		0.8.0
xz	5.2.2	5.2.5
xz-libs	5.2.2	5.2.5
yum	3.4.3	4.14.0
yum-metadata-parser	1.1.4	
yum-plugin-priorities	1.1.31	
yum-plugin-upgrade-helper	1.1.31	
yum-utils	1.1.31	
zip	3.0	3.0
zlib	1.2.8	1.2.11
zram-generator		1.1.2
zram-generator-defaults		1.1.2

Paquete	AMI de AL1	AMI de AL2023
zstd		1.5.5

Comparación de paquetes instalados en las AMI mínimas de Amazon Linux 1 (AL1) y Amazon Linux 2023

Comparación de las RPM presentes en las AMI mínimas AL1 y AL2023.

Paquete	AL1 mínimo	AL2023 mínimo
acpid	2.0.19	
alternatives		1.15
amazon-chrony-config		4.3
amazon-ec2-net-utils		2.5.1
amazon-linux-repo-s3		2023.6.20241031
amazon-linux-sb-keys		2023.1
amd-ucode-firmware		20210208
audit	2.6.5	3.0.6
audit-libs	2.6.5	3.0.6
authconfig	6.2.8	
awscli-2		2.15.30
basesystem	10.0	11
bash	4.2.46	5.2.15
binutils	2.27	

Paquete	AL1 mínimo	AL2023 mínimo
bzip2	1.0.6	
bzip2-libs	1.0.6	1.0.8
ca-certificates	2023.2.62	2023.2.68
checkpolicy	2.1.10	3.4
chkconfig	1.3.49.3	
chrony		4.3
cloud-disk-utils	0.27	
cloud-init	0.7.6	22.2.2
cloud-init-cfg-ec2		22.2.2
cloud-utils-growpart		0.31
coreutils	8.22	8.32
coreutils-common		8.32
cpio	(2.10)	2.13
cracklib	2.8.16	2.9.6
cracklib-dicts	2.8.16	2.9.6
cronie	1.4.4	
cronie-anacron	1.4.4	
crontabs	1.10	
crypto-policies		20220428
cryptsetup-libs		2.6.1

Paquete	AL1 mínimo	AL2023 mínimo
<u>curl</u>	7.61.1	
<u>curl-minimal</u>		8.5.0
cyrus-sasl	2.1.23	
cyrus-sasl-lib	2.1.23	2.1.27
dash	0.5.5.1	
db4	4.7.25	
db4-utils	4.7.25	
dbus		1.12.28
dbus-broker		32
dbus-common		1.12.28
dbus-libs	1.6.12	1.12.28
device-mapper		1.02.185
device-mapper-libs		1.02.185
dhclient	4.1.1	
dhcp-common	4.1.1	
diffutils	3.3	3.8
dnf		4.14.0
dnf-data		4.14.0
dnf-plugin-release-notification		1.2
dnf-plugins-core		4.3.0

Paquete	AL1 mínimo	AL2023 mínimo
dnf-plugin-support-info		1.2
dracut	004	055
dracut-config-ec2		3.0
dracut-config-generic		055
dracut-modules-growroot	0.20	
e2fsprogs	1.43.5	1.46.5
e2fsprogs-libs	1.43.5	1.46.5
ec2-utils	0.7	2.2.0
ed	1.1	
efi-filesystem		5
efivar		38
efivar-libs		38
elfutils-default-yama-scope		0.188
elfutils-libelf	0.168	0.188
elfutils-libs		0.188
ethtool	3.15	
expat	2.1.0	2.5.0
file	5.37	5.39

Paquete	AL1 mínimo	AL2023 mínimo
file-libs	5.37	5.39
filesystem	2.4.30	3.14
findutils	4.4.2	4.8.0
fipscheck	1.3.1	
fipscheck-lib	1.3.1	
fuse-libs	2.9.4	2.9.9
gawk	3.1.7	5.1.0
gdbm	1.8.0	
gdbm-libs		1.19
gdisk	0.8.10	1.0.8
generic-logos	17.0.0	
get_reference_source	1.2	
gettext		0.21
gettext-libs		0.21
glib2	2.36.3	2.74.7
glibc	2.17	2.34
glibc-all-langpacks		2.34
glibc-common	2.17	2.34
glibc-locale-source		2.34
gmp	6.0.0	6.2.1

Paquete	AL1 mínimo	AL2023 mínimo
gnupg2	2.0.28	
gnupg2-minimal		2.3.7
gnutls		3.8.0
gpgme	1.4.3	1.15.1
grep	2.20	3.8
groff	1.22.2	
groff-base	1.22.2	1.22.4
grub	0.97	
grub2-common		2.06
grub2-efi-x64-ec2		2.06
grub2-pc-modules		2.06
grub2-tools		2.06
grub2-tools-minimal		2.06
grubby	7.0.15	8.40
gzip	1.5	1.12
hesiod	3.1.0	
hmaccalc	0.9.12	
hostname		3.23
hwdata	0.233	0.384
info	5.1	

Paquete	AL1 mínimo	AL2023 mínimo
inih		49
initscripts	9.03.58	10.09
iproute	4.4.0	6.10.0
iptables	1.4.21	
iputils	20121221	20210202
irqbalance		1.9.0
jansson		2.14
jitterentropy		3.4.1
jq		1.7.1
json-c		0.14
kbd	1.15	2.4.0
kbd-misc	1.15	2.4.0
kernel	4.14.336	6.1.112
kernel-libbpf		6.1.112
kernel-livepatch-r epo-s3		2023.6.20241031
keyutils-libs	1.5.8	1.6.3
kmod	14	29
kmod-libs	14	29
krb5-libs	1.15.1	1.21.3
less	436	608

Paquete	AL1 mínimo	AL2023 mínimo
libacl	2.2.49	2.3.1
libarchive		3.7.4
libargon2		20171227
libassuan	2.0.3	2.5.5
libattr	2.4.46	2.5.1
libblkid	2.23.2	2.37.4
libcap	2.16	2.48
libcap54	2.54	
libcap-ng	0.7.5	0.8.2
libcbor		0.7.0
libcgroup	0.40.rc1	
libcom_err	1.43.5	1.46.5
libcomps		0.1.20
libcurl	7.61.1	
libcurl-minimal		8.5.0
libdb		5.3.28
libdnf		0.69.0
libeconf		0.4.0
libedit	2.11	3.1
libfdisk		2.37.4

Paquete	AL1 mínimo	AL2023 mínimo
libffi	3.0.13	3.4.4
libfido2		1.10.0
libgcc		11.4.1
libgcc72	7.2.1	
libgcrypt	1.5.3	1.10.2
libgomp		11.4.1
libgpg-error	1.11	1.42
libicu	50.2	
libidn	1.18	
libidn2	2.3.0	2.3.2
libkcapi		1.4.0
libkcapi-hmaccalc		1.4.0
libmnl	1.0.3	1.0.4
libmodulemd		2.13.0
libmount	2.23.2	2.37.4
libnetfilter_connt rack	1.0.4	
libnfnetworking	1.0.1	
libnghttp2	1.33.0	1.59.0
libnih	1.0.1	
libpipeline		1.5.3

Paquete	AL1 mínimo	AL2023 mínimo
libpsl	0.6.2	0.21.1
libpwquality	1.2.3	1.4.4
librepo		1.14.5
libreport-filesystem		2.15.2
libseccomp		2.5.3
libselinux	2.1.10	3.4
libselinux-utils	2.1.10	3.4
libsemanage	2.1.6	3.4
libsepol	2.1.7	3.4
libsigsegv		2.13
libsmartcols	2.23.2	2.37.4
libsolv		0.7.22
libss	1.43.5	1.46.5
libssh2	1.4.2	
libstdc++		11.4.1
libstdc++72	7.2.1	
libsysfs	2.1.0	
libtasn1	2.3	4.19.0
libtextstyle		0.21
libudev	173	

Paquete	AL1 mínimo	AL2023 mínimo
libunistring	0.9.3	0.9.10
libuser	0.60	0.63
libutempter	1.1.5	1.2.1
libuuid	2.23.2	2.37.4
libverto	0.2.5	0.3.2
libxcrypt		4.4.33
libxml2	2.9.1	2.10.4
libyaml	0.1.6	0.2.5
libzstd		1.5.5
linux-firmware-whe nce		20210208
logrotate	3.7.8	3.20.1
lua	5.1.4	
lua-libs		5.4.4
lz4-libs		1.9.4
make	3.82	
man-db		2.9.3
microcode_ctl	2.1	2.1
mingetty	1.08	
mpfr		4.1.0
ncurses	5.7	6.2

Paquete	AL1 mínimo	AL2023 mínimo
ncurses-base	5.7	6.2
ncurses-libs	5.7	6.2
nettle		3.8
net-tools	1.60	2.0
newt	0.52.11	
newt-python27	0.52.11	
npth		1.6
nspr	4.25.0	
nss	3.53.1	
nss-pem	1.0.3	
nss-softokn	3.53.1	
nss-softokn-freebl	3.53.1	
nss-sysinit	3.53.1	
nss-tools	3.53.1	
nss-util	3.53.1	
ntp	4.2.8p15	
ntpd	4.2.8p15	
numactl-libs		2.0.14
oniguruma		6.9.7.1
openldap	2.4.40	2.4.57

Paquete	AL1 mínimo	AL2023 mínimo
openssh	7.4p1	8.7p1
openssh-clients		8.7p1
openssh-server	7.4p1	8.7p1
openssl	1.0.2k	3.0.8
openssl-lib		3.0.8
openssl-pkcs11		0.4.12
os-prober		1.77
p11-kit	0.18.5	0.24.1
p11-kit-trust	0.18.5	0.24.1
pam	1.1.8	1.5.1
passwd	0.79	0.80
pciutils	3.1.10	3.7.0
pciutils-lib	3.1.10	3.7.0
pcre	8.21	
pcre2		10.40
pcre2-syntax		10.40
pinentry	0.7.6	
pkgconfig	0.27.1	
policycoreutils	2.1.12	3.4
popt	1.13	1.18

Paquete	AL1 mínimo	AL2023 mínimo
procmail	3.22	
procps	3.2.8	
procps-ng		3.3.17
psmisc	22.20	23.4
pth	2.0.7	
publicsuffix-list-dafsa		20240212
python27	2.7.18	
python27-babel	0.9.4	
python27-backports	1.0	
python27-backports-ssl_match_hostname	3.4.0.2	
python27-chardet	2.0.1	
python27-configobj	4.7.2	
python27-iniparse	0.3.1	
python27-jinja2	2.7.2	
python27-jsonpatch	1.2	
python27-jsonpointer	1.0	
python27-libs	2.7.18	
python27-markupsafe	0.11	
python27-pycurl	7.19.0	

Paquete	AL1 mínimo	AL2023 mínimo
python27-pygpme	0.3	
python27-pyliblzma	0.5.3	
python27-pyattr	0.5.0	
python27-PyYAML	3.10	
python27-requests	1.2.3	
python27-setuptools	36.2.7	
python27-six	1.8.0	
python27-urlgrabber	3.10	
python27-urllib3	1.24.3	
python3		3.9.16
python3-attrs		20.3.0
python3-audit		3.0.6
python3-awscrt		0.19.19
python3-babel		2.9.1
python3-cffi		1.14.5
python3-chardet		4.0.0
python3-colorama		0.4.4
python3-configobj		5.0.6
python3-cryptography		36.0.1
python3-dateutil		2.8.1

Paquete	AL1 mínimo	AL2023 mínimo
python3-dbus		1.2.18
python3-distro		1.5.0
python3-dnf		4.14.0
python3-dnf-plugins-core		4.3.0
python3-docutils		0.16
python3-gpg		1.15.1
python3-hawkey		0.69.0
python3-idna		(2.10)
python3-jinja2		2.11.3
python3-jmespath		0.10.0
python3-jsonpatch		1.21
python3-jsonpointer		2.0
python3-jjsonschema		3.2.0
python3-libcomps		0.1.20
python3-libdnf		0.69.0
python3-libs		3.9.16
python3-libselenium		3.4
python3-libsemanage		3.4
python3-markupsafe		1.1.1
python3-netifaces		0.10.6

Paquete	AL1 mínimo	AL2023 mínimo
python3-oauthlib		3.0.2
python3-pip-wheel		21.3.1
python3-ply		3.11
python3-policycore utils		3.4
python3-prettytable		0.7.2
python3-prompt-too lkit		3.0.24
python3-pycparser		2.20
python3-pyrsistent		0.17.3
python3-pyserial		3.4
python3-pysocks		1.7.1
python3-pytz		2022.7.1
python3-pyyaml		5.4.1
python3-requests		2.25.1
python3-rpm		4.16.1.3
python3-ruamel-yaml		0.16.6
python3-ruamel-yaml- clib		0.1.2
python3-setuptools		4.4.1
python3-setuptools		59.6.0

Paquete	AL1 mínimo	AL2023 mínimo
python3-setuptools-wheel		59.6.0
python3-six		1.15.0
python3-systemd		235
python3-urllib3		1.25.10
python3-wcwidth		0.2.5
readline	6.2	8.1
rng-tools		6.14
rootfiles	8.1	8.1
rpm	4.11.3	4.16.1.3
rpm-build-libs	4.11.3	4.16.1.3
rpm-libs	4.11.3	4.16.1.3
rpm-plugin-selinux		4.16.1.3
rpm-plugin-systemd-inhibit		4.16.1.3
rpm-python27	4.11.3	
rpm-sign-libs		4.16.1.3
rsyslog	5.8.10	
sbsigntools		0.9.4
sed	4.2.1	4.8
selinux-policy		38.1.45

Paquete	AL1 mínimo	AL2023 mínimo
selinux-policy-targeted		38.1.45
sendmail	8.14.4	
setserial	2.17	
setup	2.8.14	2.13.7
shadow-utils	4.1.4.2	4.9
shared-mime-info	1.1	
slang	2.2.1	
sqlite	3.7.17	
sqlite-libs		3.40.0
sudo	1.8.23	1.9.15
sysctl-defaults	1.0	1.0
sysfsutils	2.1.0	
systemd		252.23
systemd-libs		252.23
systemd-networkd		252.23
systemd-pam		252.23
systemd-resolved		252.23
systemd-udev		252.23
system-release	2018.03	2023.6.20241031
sysvinit	2.87	

Paquete	AL1 mínimo	AL2023 mínimo
tar	1.26	1.34
tcp_wrappers-libs	7.6	
tzdata	2023c	2024a
udev	173	
update-motd	1.0.1	2.2
upstart	0.6.5	
userspace-rcu		0.12.1
ustr	1.0.4	
util-linux	2.23.2	2.37.4
util-linux-core		2.37.4
vim-data	9.0.2120	9.0.2153
vim-minimal	9.0.2120	9.0.2153
which	2.19	2.21
xfsprogs		5.18.0
xz	5.2.2	5.2.5
xz-libs	5.2.2	5.2.5
yum	3.4.3	4.14.0
yum-metadata-parser	1.1.4	
yum-plugin-priorities	1.1.31	

Paquete	AL1 mínimo	AL2023 mínimo
yum-plugin-upgrade-helper	1.1.31	
zlib	1.2.8	1.2.11
zram-generator		1.1.2
zram-generator-defaults		1.1.2
zstd		1.5.5

Comparación de paquetes instalados en imágenes de contenedor base de Amazon Linux 1 (AL1) y Amazon Linux 2023

Comparación de los RPM presentes en las imágenes de contenedor base de AL1 y AL2023.

Paquete	Contenedor de AL1	Contenedor de AL2023
alternatives		1.15
amazon-linux-repo-cdn		2023.6.20241031
audit-libs		3.0.6
basesystem	10.0	11
bash	4.2.46	5.2.15
bzip2-libs	1.0.6	1.0.8
ca-certificates	2023.2.62	2023.2.68
chkconfig	1.3.49.3	
coreutils	8.22	

Paquete	Contenedor de AL1	Contenedor de AL2023
coreutils-single		8.32
crypto-policies		20220428
curl	7.61.1	
curl-minimal		8.5.0
cyrus-sasl-lib	2.1.23	
db4	4.7.25	
db4-utils	4.7.25	
dnf		4.14.0
dnf-data		4.14.0
elfutils-default-yama-scope		0.188
elfutils-libelf	0.168	0.188
elfutils-libs		0.188
expat	2.1.0	2.5.0
file-libs	5.37	5.39
filesystem	2.4.30	3.14
gawk	3.1.7	5.1.0
gdbm	1.8.0	
gdbm-libs		1.19
glib2	2.36.3	2.74.7
glibc	2.17	2.34

Paquete	Contenedor de AL1	Contenedor de AL2023
<code>glibc-common</code>	2.17	2.34
<code>glibc-minimal-langpack</code>		2.34
<code>gmp</code>	6.0.0	6.2.1
<code>gnupg2</code>	2.0.28	
<code>gnupg2-minimal</code>		2.3.7
<code>gpgme</code>	1.4.3	1.15.1
<code>grep</code>	2.20	3.8
<code>gzip</code>	1.5	
<code>info</code>	5.1	
<code>json-c</code>		0.14
<code>keyutils-libs</code>	1.5.8	1.6.3
<code>krb5-libs</code>	1.15.1	1.21.3
<code>libacl</code>	2.2.49	2.3.1
<code>libarchive</code>		3.7.4
<code>libassuan</code>	2.0.3	2.5.5
<code>libattr</code>	2.4.46	2.5.1
<code>libblkid</code>		2.37.4
<code>libcap</code>	2.16	2.48
<code>libcap-ng</code>		0.8.2
<code>libcom_err</code>	1.43.5	1.46.5

Paquete	Contenedor de AL1	Contenedor de AL2023
libcomps		0.1.20
libcurl	7.61.1	
libcurl-minimal		8.5.0
libdnf		0.69.0
libffi	3.0.13	3.4.4
libgcc		11.4.1
libgcc72	7.2.1	
libgcrypt	1.5.3	1.10.2
libgomp		11.4.1
libgpg-error	1.11	1.42
libicu	50.2	
libidn2	2.3.0	2.3.2
libmodulemd		2.13.0
libmount		2.37.4
libnghttp2	1.33.0	1.59.0
libpsl	0.6.2	0.21.1
librepo		1.14.5
libreport-filessystem		2.15.2
libselinux	2.1.10	3.4
libsepol	2.1.7	3.4

Paquete	Contenedor de AL1	Contenedor de AL2023
libsigsegv		2.13
libsmartcols		2.37.4
libsolv		0.7.22
libssh2	1.4.2	
libstdc++		11.4.1
libstdc++72	7.2.1	
libtasn1	2.3	4.19.0
libunistring	0.9.3	0.9.10
libuuid		2.37.4
libverto	0.2.5	0.3.2
libxcrypt		4.4.33
libxml2	2.9.1	2.10.4
libxml2-python27	2.9.1	
libyaml		0.2.5
libzstd		1.5.5
lua	5.1.4	
lua-libs		5.4.4
lz4-libs		1.9.4
make	3.82	
mpfr		4.1.0

Paquete	Contenedor de AL1	Contenedor de AL2023
ncurses	5.7	
ncurses-base	5.7	6.2
ncurses-libs	5.7	6.2
npth		1.6
nspr	4.25.0	
nss	3.53.1	
nss-pem	1.0.3	
nss-softokn	3.53.1	
nss-softokn-freebl	3.53.1	
nss-sysinit	3.53.1	
nss-tools	3.53.1	
nss-util	3.53.1	
openldap	2.4.40	
openssl	1.0.2k	
openssl-libs		3.0.8
p11-kit	0.18.5	0.24.1
p11-kit-trust	0.18.5	0.24.1
pcre	8.21	
pcre2		10.40
pcre2-syntax		10.40

Paquete	Contenedor de AL1	Contenedor de AL2023
pinentry	0.7.6	
pkgconfig	0.27.1	
popt	1.13	1.18
pth	2.0.7	
publicsuffix-list-dafsa		20240212
python27	2.7.18	
python27-chardet	2.0.1	
python27-iniparse	0.3.1	
python27-kitchen	1.1.1	
python27-libs	2.7.18	
python27-pycurl	7.19.0	
python27-pygpme	0.3	
python27-pyliblzma	0.5.3	
python27-pyattr	0.5.0	
python27-urlgrabber	3.10	
python3		3.9.16
python3-dnf		4.14.0
python3-gpg		1.15.1
python3-hawkey		0.69.0
python3-libcomps		0.1.20

Paquete	Contenedor de AL1	Contenedor de AL2023
python3-libdnf		0.69.0
python3-libs		3.9.16
python3-pip-wheel		21.3.1
python3-rpm		4.16.1.3
python3-setuptools-wheel		59.6.0
readline	6.2	8.1
rpm	4.11.3	4.16.1.3
rpm-build-libs	4.11.3	4.16.1.3
rpm-libs	4.11.3	4.16.1.3
rpm-python27	4.11.3	
rpm-sign-libs		4.16.1.3
sed	4.2.1	4.8
setup	2.8.14	2.13.7
shared-mime-info	1.1	
sqlite	3.7.17	
sqlite-libs		3.40.0
sysctl-defaults	1.0	
system-release	2018.03	2023.6.20241031
tar	1.26	
tzdata	2023c	2024a

Paquete	Contenedor de AL1	Contenedor de AL2023
xz-libs	5.2.2	5.2.5
yum	3.4.3	4.14.0
yum-metadata-parser	1.1.4	
yum-plugin-ovl	1.1.31	
yum-plugin-priorities	1.1.31	
yum-utils	1.1.31	
zlib	1.2.8	1.2.11

Requisitos del sistema de AL2023

En esta sección se describen los requisitos del sistema para usar AL2023.

Temas

- [Requisitos de CPU para ejecutar AL2023](#)
- [Requisitos de memoria \(RAM\) para ejecutar AL2023](#)

Requisitos de CPU para ejecutar AL2023

Para ejecutar cualquier código de AL2023, el procesador utilizado debe cumplir ciertos requisitos mínimos. Los intentos de ejecutar AL2023 en las CPU que no cumplan estos requisitos pueden provocar errores de instrucción ilegal en una fase muy temprana de la ejecución del código.

Estos requisitos mínimos se aplican a [AL2023 en Amazon EC2](#), [AL2023 en contenedores](#) y [AL2023 fuera de Amazon EC2](#).

Requisitos de CPU ARM para AL2023

Todos los binarios de AL2023 `aarch64` (ARM) están diseñados para 64 bits. No hay binarios ARM de 32 bits disponibles, por lo que se requiere una CPU ARM de 64 bits.

Note

Para las instancias basadas en ARM, AL2023 solo admite tipos de instancias que utilizan procesadores Graviton2 o posteriores. AL2023 no admite instancias A1.

AL2023 requiere un procesador compatible con ARMv8.2 con la extensión de criptografía (ARMv8.2+crypto). Todos los paquetes de AL2023 para `aarch64` están diseñados con el indicador del compilador `-march=armv8.2-a+crypto`. Aunque intentamos mostrar mensajes de error claros cuando se intenta ejecutar el código AL2023 en procesadores ARM más antiguos, es posible que el primer mensaje de error sea un error de instrucción ilegal.

 Note

Debido a los requisitos básicos de CPU de aarch64 de AL2023, todos los sistemas Raspberry Pi anteriores a Raspberry Pi 5 no cumplen los requisitos mínimos de CPU.

Requisitos de CPU x86-64 para AL2023

Todos los archivos binarios x86-64 de AL2023 están diseñados para la revisión x86-64v2 de la arquitectura x86-64 mediante su transferencia al compilador `-march=x86-64-v2`.

La revisión x86-64v2 de la arquitectura añade las siguientes características de la CPU a la arquitectura básica:

- CMPXCHG16B
- LAHF-SAHF
- POPCNT
- SSE3
- SSE4_1
- SSE4_2
- SSSE3

Esto se corresponde aproximadamente con los procesadores x86-64 lanzados en 2009 o después. Los ejemplos incluyen Intel Nehalem, AMD Jaguar y Atom Silvermont, junto con las microarquitecturas VIA Nano y Eden C .

En Amazon EC2, todos los tipos de instancias x86-64 admiten x86-64v2, incluidas las familias de instancias M1, C1 y M2.

No se han compilado binarios de AL2023 de 32 bits x86 (i686). Aunque AL2023 sigue siendo compatible con la ejecución de binarios de espacio de usuario de 32 bits, esta funcionalidad está en desuso y podría eliminarse en una futura versión principal de Amazon Linux. Para obtener más información, consulte [Paquetes x86 \(i686\) de 32 bits](#).

Requisitos de memoria (RAM) para ejecutar AL2023

La familia `.nano` de tipos de instancias de Amazon EC2 (`t2.nano`, `t3.nano`, `t3a.nano` y `t4g.nano`) tiene 512 MB de RAM, y este es el requisito mínimo para AL2023.

Note

Aunque el requisito mínimo es 512 MB, estos tipos de instancias tienen limitaciones de memoria y es posible que la funcionalidad y el rendimiento estén limitados.

Las imágenes de AL2023 no se han probado en sistemas con menos de 512 MB de RAM. La ejecución de imágenes de contenedores basadas en AL2023 en menos de 512 MB de RAM dependerá de cuál sea la carga de trabajo en contenedores.

Algunas cargas de trabajo, como `dnf upgrade` que se producen entre algunas versiones de AL2023, pueden requerir más de 512 MB de RAM. Por este motivo, la versión de [AL2023.3](#) introdujo la activación `zram` de forma predeterminada para las instancias con menos de 800 MB de RAM. En el caso de las cargas de trabajo en contenedores, esto significa que algunas cargas de trabajo pueden funcionar bien en instancias AL2023 con esta cantidad de memoria, pero fallan cuando se ejecutan en un contenedor restringido a esta cantidad de uso de memoria.

Para los tipos de instancias con menos de 800 MB de RAM, AL2023 (a partir de [AL2023.3](#) o posterior) ahora habilita el intercambio basado en `zram` de forma predeterminada. Algunos ejemplos de tipos de instancias de Amazon EC2 con menos de 800 MB de memoria son `t4g.nano`, `t3a.nano`, `t3.nano`, `t2.nano` y `t1.micro`. Esto se traduce en menos escenarios de falta de memoria para estos tipos de instancias, ya que AL2023 comprimirá y descomprimirá las páginas de memoria según se requiera. Esto permite cargas de trabajo que, de otro modo, requerirían un tipo de instancia con más memoria, a costa del uso de la CPU necesario para realizar la compresión.

El núcleo de Amazon Linux

Amazon Linux 2023 (AL2023) publica un nuevo kernel de Long-Term Support (LTS) en el primer trimestre de cada año, basado en el kernel LTS anual de la comunidad Linux upstream. Cada nuevo núcleo LTS incorpora las últimas correcciones de seguridad, mejoras de rendimiento, soporte de hardware y funciones del núcleo original de Linux.

ciclo de vida del núcleo

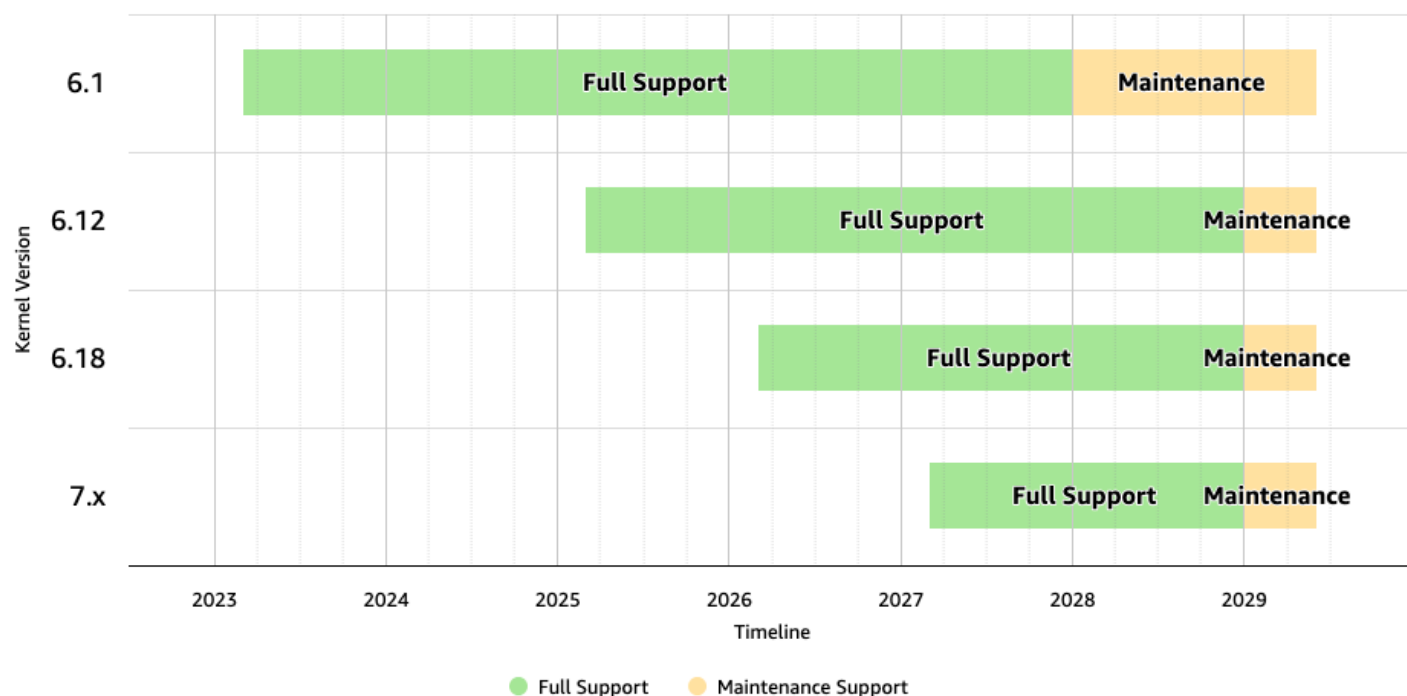
Cada núcleo AL2023 LTS recibe soporte durante cuatro años en dos fases:

1. Soporte completo (1 a 2 años): el núcleo recibe correcciones para todos los problemas de gravedad de la CVE mediante rebases periódicos en el núcleo LTS original. Esta fase se alinea con el período de soporte para el LTS de la comunidad originaria de Linux. Si upstream amplía la ventana de soporte de LTS, Amazon Linux hará lo mismo.
2. Soporte de mantenimiento (3 a 4 años): una vez finalizado el período de soporte inicial de LTS, el equipo de Amazon Linux sigue respaldando principalmente las correcciones de las vulnerabilidades críticas e importantes CVEs (puntuación CVSS de 7.0 o superior), así como las vulnerabilidades explotadas conocidas. Durante esta fase, no se admiten casos de gravedad CVEs baja o media.

Al cabo de cuatro años, el núcleo llega al final de su vida útil y ya no recibe actualizaciones de seguridad. Los núcleos más antiguos permanecen disponibles en los repositorios y puede seguir utilizándolos. No debe esperar más parches o correcciones. Recomendamos actualizar a un núcleo compatible antes de esta fecha. Los núcleos específicos ya no se AMIs actualizarán después de la fecha de finalización del soporte de un núcleo.

En el siguiente gráfico se muestra el cronograma de soporte para los actuales kernels de Amazon Linux LTS:

Amazon Linux 2023 Kernel Release Timeline



Actualizaciones del kernel

A partir de junio de 2026, AL2023 actualizará el núcleo predeterminado una vez al año. El [a12023-ami-kernel-default](#) conjunto de se AMIs actualizará a la versión más reciente del núcleo LTS, de modo que las instancias recién lanzadas incorporarán la nueva versión del núcleo.

Las instancias en ejecución no se actualizan automáticamente a un nuevo núcleo. Para actualizar el núcleo en una instancia existente, debe instalar de forma explícita el nuevo paquete del núcleo y reiniciarlo. Para obtener más información, consulte [Actualización del núcleo de Linux en AL2023](#).

Recomendamos encarecidamente adoptar nuevos núcleos cuanto antes para beneficiarse de las últimas mejoras de seguridad y rendimiento. Los núcleos más antiguos reciben menos correcciones de seguridad y son más lentas con el paso del tiempo. Incorpore las actualizaciones del núcleo a sus procesos de pruebas e implementación existentes para validar la compatibilidad antes de lanzarlas a producción.

Las versiones del núcleo se superponen entre distribuciones

Para simplificar las migraciones entre las distribuciones de Amazon Linux, estará disponible al menos una versión del núcleo tanto en la distribución actual como en la siguiente de Amazon Linux. Esto

le permitirá actualizar primero a un nuevo núcleo en su distribución actual, validar sus cargas de trabajo y, a continuación, migrar a la nueva distribución con la confianza de que allí estará disponible la misma versión del núcleo.

Manejo de CVE

AL2023 aborda el núcleo de CVEs la siguiente manera:

- Las vulnerabilidades críticas e importantes CVEs (CVSS 7.0 y versiones posteriores) y las que se explotan conocidas se transfieren a todos los núcleos compatibles.
- Los valores bajos y medios CVEs (CVSS por debajo de 7.0) se abordan mediante rebases periódicos de LTS ascendentes durante la fase de soporte completo. Durante la fase de soporte de mantenimiento, no se respaldan CVEs. Si un CVE bajo o medio no se resuelve mediante una rebase de LTS ascendente en un plazo de 60 días, se marcará como «No se ha planificado ninguna solución» en el Centro de seguridad de [Amazon Linux](#).

La mejor manera de obtener las actualizaciones más recientes de seguridad, rendimiento y funcionalidad es ejecutar el kernel más reciente disponible.

¿Qué debe hacer

1. Implemente la integración continua (CI) para sus aplicaciones: antes de realizar cualquier cambio en la configuración de producción, asegúrese de que esté debidamente validada en la fase de prueba. Incluya las actualizaciones del núcleo en su validación.
2. Utilice el núcleo más reciente: adopte cada nuevo núcleo LTS anual tan pronto como esté disponible. Los núcleos más nuevos reciben las correcciones de seguridad con mayor rapidez. Consume la [al2023-ami-kernel-default](#) serie de AMIs para estar automáticamente en la versión del kernel recomendada por el equipo de Amazon Linux.
3. Automatice las actualizaciones: utilice herramientas como [AWS Systems Manager](#) para gestionar las actualizaciones del núcleo en toda su flota.
4. Planifique los reinicios: todas las actualizaciones del núcleo requieren un reinicio. Incluya ventanas de reinicio en sus programas de mantenimiento.

AL2Escritorio gráfico 023

Amazon Linux 2023 ofrece una interfaz gráfica opcional, ligera y optimizada para la nube basada en GNOME a partir de la versión 2023.7. Este moderno entorno de escritorio ofrece características de productividad mejoradas con herramientas integradas como Firefox para una navegación segura, al tiempo que mantiene la compatibilidad con Amazon DCV y VCN para el acceso remoto.

Note

AL2La 023 sigue de cerca la versión original de Firefox Extended Support Release (ESR) y se actualiza a la siguiente ESR en cuanto esté disponible. Para obtener más información, consulta el [calendario de versiones de Firefox ESR](#) y las notas de [lanzamiento de Firefox](#).

Temas relacionados

Para obtener más información sobre la instalación del entorno de escritorio gráfico, consulte la siguiente documentación:

- [Tutorial: Instalar el entorno de escritorio GNOME en AL2023](#)

Paquetes complementarios para Amazon Linux

Esta sección presenta los paquetes suplementarios para Amazon Linux (SPAL) y describe sus ventajas y limitaciones, junto con las directrices para informar de problemas relacionados con los paquetes.

Temas

- [¿Qué son los paquetes suplementarios para Amazon Linux \(o SPAL\)?](#)
- [Ventajas](#)
- [Support de paquetes SPAL](#)
- [Informar de problemas relacionados con los paquetes](#)
- [Temas relacionados](#)
- [Tutorial: Configurar el repositorio SPAL en 023 AL2](#)

¿Qué son los paquetes suplementarios para Amazon Linux (o SPAL)?

Supplementary Packages for Amazon Linux (SPAL) es un repositorio de paquetes dedicado que proporciona acceso a miles de paquetes adicionales derivados de [Extra Packages for Enterprise Linux 9 \(EPEL9\)](#). Estos paquetes complementan el software existente disponible en el núcleo de Amazon Linux 2023.

SPAL simplifica la implementación del software al proporcionar paquetes prediseñados que son compatibles con AL2 023, lo que elimina la necesidad de que los clientes creen ellos mismos los paquetes a partir del código fuente. Esto ahorra tiempo y esfuerzo en el proceso de instalación del software.

Note

SPAL está disponible en todas las regiones AWS comerciales, incluidas las regiones AWS GovCloud (EE. UU.) y China, para AL2 023 instancias con una versión de lanzamiento igual 2023.9.20251117 o posterior.

Ventajas

SPAL ofrece varias ventajas clave a los usuarios de Amazon Linux 2023:

- Ampliación de los casos de uso de AL2 023: el acceso a paquetes adicionales populares que van más allá del repositorio principal de AL2 023, por ejemplo `pandocdrbd-utils`, `GDAL` permite a los clientes satisfacer múltiples necesidades empresariales y de desarrollo.
- Simplificación de la gestión de paquetes del AL2 023: al proporcionar paquetes prediseñados para el AL2 023, se elimina el proceso de creación de paquetes adicionales a partir del código fuente, lo que ahorra tiempo y reduce el riesgo de errores de compilación.
- Optimización de la migración AL2 a la versión AL2 023: el repositorio SPAL permite migrar sin problemas las cargas de trabajo de la versión 023 a la versión AL2 023, incluidas las cargas de trabajo AL2 que dependen de los paquetes. EPEL7

Support de paquetes SPAL

Los paquetes SPAL no cuentan con el mismo nivel de soporte que los paquetes AL2 023 principales, que reciben soporte durante toda la vida útil de Amazon Linux 2023.

Important

Antes de usar SPAL, los clientes deben evaluar detenidamente las siguientes consideraciones:

- Los paquetes SPAL NO están cubiertos por AWS Support Plans.
- Los paquetes SPAL se proporcionan «tal cual» desde el principio. EPEL9
- Los paquetes SPAL NO recibirán AWS un seguimiento de seguridad CVE.
- Los paquetes SPAL reciben parches de seguridad y correcciones de errores exclusivamente de EPEL9 upstream cuando están disponibles.

Informar de problemas relacionados con los paquetes

Si encuentra un problema con un paquete SPAL, le recomendamos que compruebe primero si se produce el mismo problema con el paquete correspondiente en el repositorio original. EPEL9 Para ello, consulta la [lista de problemas del repositorio](#) original de EPEL.

Si el problema no está presente EPEL9, cree uno en el [GitHubrepositorio de Amazon Linux 2023](#), ya que esto indica que el problema es específico de la compilación o configuración del paquete SPAL.

Este enfoque garantiza que los responsables del mantenimiento adecuados aborden los problemas y contribuye a la calidad general de los paquetes SPAL y EPEL originales.

 Note

Los problemas denunciados se gestionarán haciendo todo lo posible.

Temas relacionados

Para obtener información sobre la configuración de SPAL en su sistema, consulte la siguiente página de documentación:

- [Tutorial: Configurar el repositorio SPAL en 023 AL2](#)

Tutorial: Configurar el repositorio SPAL en 023 AL2

Supplementary Packages for Amazon Linux (SPAL) es un repositorio de paquetes adicional para AL2 023 que proporciona a los clientes acceso a miles de paquetes de código abierto.

El siguiente tutorial le ayuda a configurar el repositorio SPAL en su instancia 023. AL2 Al instalar el repositorio, obtendrá acceso a todos los paquetes RPM disponibles en SPAL. Una vez instalado, puede usar su administrador de paquetes para instalar y usar estos paquetes en su sistema.

Contenido

- [Requisitos previos](#)
- [Comprobación de los requisitos previos](#)
- [Instalar SPAL en su sistema](#)
- [Instalación de paquetes SPAL](#)
- [Desinstalar el repositorio SPAL del sistema](#)
- [Temas relacionados](#)

Requisitos previos

En este tutorial se asume que ya ha lanzado una instancia con la versión AL2 023 2023.9.20251117 o posterior. Para obtener más información, consulte las páginas [AL2023 en Amazon EC2](#) y [Actualización AL2023](#).

Comprobación de los requisitos previos

- Para comprobar que la instancia cumple los requisitos previos, puede comprobar la versión `system-release` instalada en el sistema.

Para comprobar la versión del paquete, puedes usar el siguiente comando.

```
[ec2-user ~]$ rpm -qi system-release
```

El comando mostrará información sobre el paquete, incluida la versión principal.

```
Name       : system-release
Version    : 2023.9.20251117
...
```

Note

Asegúrese de tener `system-release` instalada la última versión de. Puede correr `sudo dnf upgrade` para actualizar a la última versión.

Instalar SPAL en su sistema

1. Instale el `spal-release` paquete en su sistema. Esto añade el archivo `.repo` de configuración y las claves GPG a su sistema.

```
[ec2-user ~]$ sudo dnf install spal-release
```

 Note

Durante la instalación, se mostrará la declaración de soporte. La declaración explica el alcance del soporte y las limitaciones del SPAL. Tómese un tiempo para revisar esta información detenidamente.

2. Compruebe que la configuración del repositorio SPAL se haya agregado correctamente a su sistema.

```
[ec2-user ~]$ cat /etc/yum.repos.d/amazonlinux-spal.repo
```

Debería ver los dos repositorios configurados en su sistema: `amazonlinux-spal` y `amazonlinux-spal-source`

También puede comprobar la lista de repositorios configurados ejecutándolo. `dnf repolist`

```
[ec2-user ~]$ dnf repolist --all
```

 Note

La `--all` marca es obligatoria para ver los repositorios activados y desactivados.

Ambos repositorios SPAL deberían estar disponibles. Tenga en cuenta que el repositorio SPAL de Amazon Linux 2023: repositorio de paquetes fuente está deshabilitado de forma predeterminada.

repo id	repo name
status	
amazonlinux-spal	Amazon Linux 2023 SPAL repository
enabled	
amazonlinux-spal-source	Amazon Linux 2023 SPAL repository - Source packages
disabled	

3. (Opcional) Habilite el repositorio de origen.

 Note

Los repositorios fuente RPM (SRPM) suelen estar deshabilitados de forma predeterminada porque los utilizan principalmente los desarrolladores para crear paquetes, no los usuarios finales para la instalación del software. DNF habilita automáticamente los repositorios de código fuente cuando se utilizan comandos que requieren paquetes de código fuente, por ejemplo. `dnf download --source package`

No es necesario activar manualmente el repositorio de fuentes para realizar operaciones puntuales con los paquetes fuente. Siga este paso solo si quiere volver a crear su SRPMs sistema desde SPAL.

Para habilitar permanentemente el repositorio SPAL de Amazon Linux 2023 (repositorio de paquetes fuente) en su sistema, ejecute el siguiente comando:

```
[ec2-user ~]$ sudo dnf config-manager --enable amazonlinux-spal-source
```

Instalación de paquetes SPAL

- Instale los paquetes SPAL en su sistema ejecutando `dnf install` un comando.

```
[ec2-user ~]$ sudo dnf install package
```

 Note

Puede utilizarla `dnf list` para ver una lista completa de los paquetes SPAL.

```
[ec2-user ~]$ dnf list --repo=amazonlinux-spal
```

 Note

SPAL es un repositorio versionado. Asegúrese de tener `system-release` instalada la última versión de para ver la lista de paquetes más reciente.

Para obtener más información sobre las actualizaciones deterministas, puede consultar [Actualizaciones deterministas mediante repositorios versionados en AL2023](#)

Desinstalar el repositorio SPAL del sistema

1. Elimine la configuración del repositorio SPAL mediante un comando. `dnf remove`

```
[ec2-user ~]$ sudo dnf remove spal-release
```

2. Compruebe que el repositorio se haya eliminado ejecutando el `dnf repolist` comando.

```
[ec2-user ~]$ dnf repolist
```

Important

Al eliminar la configuración del repositorio de SPAL del sistema, no se elimina ningún paquete de SPAL instalado en el sistema.

Temas relacionados

Para obtener más información sobre el repositorio Supplementary Packages for Amazon Linux, consulte la siguiente documentación:

- [Paquetes complementarios para Amazon Linux](#)

Ejecución de aplicaciones en AL2023

En esta sección se describen los métodos para ejecutar aplicaciones en Amazon Linux 2023 (AL2023), incluida la administración del momento en que se inician (y se reinician) y el control del uso de los recursos.

Temas

- [Limite el uso de recursos de procesos en AL2023 mediante systemd](#)
- [Limitación del uso de recursos de procesos en AL2023 mediante cgroups](#)

Limite el uso de recursos de procesos en AL2023 mediante systemd

En Amazon Linux 2023 (AL2023), recomendamos usar `systemd` para controlar qué recursos pueden usar los procesos o grupos de procesos. El uso de `systemd` es una alternativa potente y fácil de usar tanto para manipular `cgroups` manualmente como para utilizar utilidades como [cpulimit](#), que anteriormente solo estaba disponible para Amazon Linux en el repositorio [EPEL](#) de terceros.

Para obtener información completa, consulte la documentación original de `systemd` para [systemd.resource-control](#) o la página sobre `man` para `systemd.resource-control` en una instancia de AL2023.

En los ejemplos siguientes se utilizará la prueba de stress de la CPU `stress-ng` (incluida en el paquete `stress-ng`) para simular una aplicación con un uso intensivo de la CPU y `memcached` con un uso intensivo de memoria.

En los siguientes ejemplos, se describe cómo establecer un límite de CPU para un comando único y un límite de memoria para un servicio. La mayoría de las restricciones de recursos que ofrece `systemd` se pueden utilizar en cualquier lugar en el que `systemd` ejecute un proceso, y se pueden utilizar varias al mismo tiempo. Los ejemplos siguientes se limitan a una sola restricción con fines ilustrativos.

Control de recursos con **systemd-run** para ejecutar comandos únicos

Si bien suele asociarse a los servicios del sistema, también pueden utilizar `systemd` los usuarios que no son `root` para ejecutar servicios, programar temporizadores o ejecutar procesos únicos. En el siguiente ejemplo, vamos a utilizar `stress-ng` como aplicación de ejemplo. En el primer ejemplo,

la ejecutaremos con `systemd-run` en la cuenta `ec2-user` predeterminada y, en el segundo, pondremos límites al uso de la CPU.

Example Use **`systemd-run`** en la línea de comandos para ejecutar un proceso, no para limitar el uso de recursos

1. Asegúrese de que el paquete `stress-ng` esté instalado, ya que lo usaremos en nuestro ejemplo.

```
[ec2-user ~]$ sudo dnf install -y stress-ng
```

2. Utilice `systemd-run` para ejecutar una prueba de estrés de la CPU de 10 segundos sin limitar la cantidad de CPU que puede utilizar.

```
[ec2-user ~]$ systemd-run --user --tty --wait --property=CPUAccounting=1 stress-ng
--cpu 1 --timeout 10
Running as unit: run-u6.service
Press ^] three times within 1s to disconnect TTY.
stress-ng: info:  [339368] setting to a 10 second run per stressor
stress-ng: info:  [339368] dispatching hogs: 1 cpu
stress-ng: info:  [339368] successful run completed in 10.00s
Finished with result: success
Main processes terminated with: code=exited/status=0
Service runtime: 10.068s
CPU time consumed: 9.060s
```

La opción `--user` le dice a `systemd-run` que ejecute el comando como el usuario con el que estamos conectados, la opción `--tty` significa que hay un TTY conectado, `--wait` significa esperar hasta que el servicio haya terminado, y la opción `--property=CPUAccounting=1` le indica a `systemd-run` que registre cuánto tiempo de CPU se utiliza para ejecutar el proceso. La opción de línea de comandos `--property` se puede utilizar para pasar la configuración de `systemd-run` que se podría configurar en un archivo de configuración `systemd.unit`.

Cuando se le indica que coloque carga en la CPU, el programa `stress-ng` utilizará todo el tiempo de CPU disponible para realizar su prueba durante el tiempo que usted le pida que se ejecute. Para una aplicación en el mundo real, puede ser conveniente establecer un límite en el tiempo total de ejecución de un proceso. En el siguiente ejemplo, le pediremos a `stress-ng` que se ejecute durante más tiempo que la restricción de duración máxima que le imponemos utilizando `systemd-run`.

Example Use **systemd-run** en la línea de comandos para ejecutar un proceso, limitando el uso de la CPU a 1 segundo

1. Asegúrese de que **stress-ng** esté instalado para ejecutar este ejemplo.
2. La propiedad **LimitCPU** es el equivalente a **ulimit -t**, lo que limitará la cantidad máxima de tiempo en la CPU que este proceso podrá utilizar. En este caso, dado que estamos pidiendo una ejecución de **stress** de 10 segundos y estamos limitando el uso de la CPU a 1 segundo, el comando recibirá una señal **SIGXCPU** y fallará.

```
[ec2-user ~]$ systemd-run --user --tty --wait --property=CPUAccounting=1 --  
property=LimitCPU=1 stress-ng --cpu 1 --timeout 10  
Running as unit: run-u12.service  
Press ^] three times within 1s to disconnect TTY.  
stress-ng: info:  [340349] setting to a 10 second run per stressor  
stress-ng: info:  [340349] dispatching hogs: 1 cpu  
stress-ng: fail:  [340349] cpu instance 0 corrupted bogo-ops counter, 1370 vs 0  
stress-ng: fail:  [340349] cpu instance 0 hash error in bogo-ops counter and run  
flag, 3250129726 vs 0  
stress-ng: fail:  [340349] metrics-check: stressor metrics corrupted, data is  
compromised  
stress-ng: info:  [340349] unsuccessful run completed in 1.14s  
Finished with result: exit-code  
Main processes terminated with: code=exited/status=2  
Service runtime: 1.201s  
CPU time consumed: 1.008s
```

Lo más habitual es restringir el porcentaje de tiempo de CPU que puede consumir un proceso concreto. En el siguiente ejemplo, restringiremos el porcentaje de tiempo de CPU que **stress-ng** puede consumir. En el caso de un servicio real, puede ser conveniente limitar el porcentaje máximo de tiempo de CPU que puede consumir un proceso en segundo plano a fin de dejar recursos libres para el proceso que atiende las solicitudes de los usuarios.

Example Use **systemd-run** para limitar un proceso al 10 % del tiempo de CPU en una CPU

1. Asegúrese de que **stress-ng** esté instalado para ejecutar este ejemplo.
2. Vamos a usar la propiedad **CPUQuota** para indicar a **systemd-run** que restrinja el uso de la CPU para el comando que vamos a ejecutar. No estamos limitando la cantidad de tiempo que puede ejecutarse el proceso, solo la cantidad de CPU que puede usar.

```
[ec2-user ~]$ systemd-run --user --tty --wait --property=CPUAccounting=1 --  
property=CPUQuota=10% stress-ng --cpu 1 --timeout 10  
Running as unit: run-u13.service  
Press ^] three times within 1s to disconnect TTY.  
stress-ng: info: [340664] setting to a 10 second run per stressor  
stress-ng: info: [340664] dispatching hogs: 1 cpu  
stress-ng: info: [340664] successful run completed in 10.08s  
Finished with result: success  
Main processes terminated with: code=exited/status=0  
Service runtime: 10.140s  
CPU time consumed: 1.014s
```

Observe cómo la contabilidad de la CPU nos indica que, aunque el servicio se ejecutó durante 10 segundos, solo consumió 1 segundo de tiempo real de CPU.

Hay muchas formas de configurar `systemd` para limitar el uso de recursos de la CPU, la memoria, las redes y las E/S. Para obtener información completa, consulte la documentación original de `systemd` para [systemd.resource-control](#) o la página sobre man para `systemd.resource-control` en una instancia de AL2023.

Mientras tanto, `systemd` usa características del kernel de Linux, como `cgroups`, para poner en práctica estos límites sin que tengas que configurarlos a mano. La [documentación del kernel de Linux para cgroup-v2](#) tiene un montón de detalles sobre cómo funcionan los `cgroups`.

Control de recursos en un servicio **systemd**

Hay varios parámetros que se pueden agregar a la sección `[Service]` de los servicios `systemd` para controlar el uso de los recursos del sistema. Estos incluyen límites estrictos y flexibles. Para conocer el comportamiento exacto de cada opción, consulta la documentación original de `systemd` para [systemd.resource-control](#) o la página sobre man para `systemd.resource-control` en una instancia de AL2023.

Los límites más utilizados son `MemoryHigh`, para especificar un límite de restricción en el uso de la memoria, `MemoryMax`, para establecer un límite máximo estricto (que, una vez alcanzado, se invoca el OOM Killer), y `CPUQuota` (como se ilustra en la sección anterior). También es posible configurar ponderaciones y prioridades en lugar de números fijos.

Example Uso de **systemd** para establecer límites de uso de memoria en los servicios

En este ejemplo, estableceremos un límite estricto de uso de memoria para memcached, una caché simple de clave-valor, y mostraremos cómo se invoca el OOM Killer para ese servicio en lugar de para todo el sistema.

1. En primer lugar, necesitamos instalar los paquetes necesarios para este ejemplo.

```
[ec2-user ~]$ sudo dnf install -y memcached libmemcached-awesome-tools
```

2. Habilite `memcached.service` y, a continuación, inicie el servicio para que se ejecute `memcached`.

```
[ec2-user ~]$ sudo systemctl enable memcached.service
Created symlink /etc/systemd/system/multi-user.target.wants/memcached.service # /usr/lib/systemd/system/memcached.service.
[ec2-user ~]$ sudo systemctl start memcached.service
```

3. Compruebe que `memcached.service` se esté ejecutando.

```
[ec2-user ~]$ sudo systemctl status memcached.service
# memcached.service - memcached daemon
   Loaded: loaded (/usr/lib/systemd/system/memcached.service; enabled; preset: disabled)
   Active: active (running) since Fri 2025-01-31 22:36:42 UTC; 1s ago
     Main PID: 356294 (memcached)
        Tasks: 10 (limit: 18907)
       Memory: 1.8M
          CPU: 20ms
      CGroup: /system.slice/memcached.service
             ##356294 /usr/bin/memcached -p 11211 -u memcached -m 64 -c 1024 -l 127.0.0.1,::1

Jan 31 22:35:36 ip-1-2-3-4.us-west-2.compute.internal systemd[1]: Started memcached.service - memcached daemon.
```

4. Ahora que `memcached` está instalado y ejecutándose, podemos observar que funciona insertando algunos datos aleatorios en la memoria caché

En `/etc/sysconfig/memcached` la variable `CACHESIZE` se establece en 64 de forma predeterminada, es decir, 64 megabytes. Al insertar más datos en la caché que el tamaño máximo de la misma, podemos ver que la llenamos y que algunos elementos se eliminan

utilizando `memcached-tool`, y que `memcached.service` está utilizando alrededor de 64 MB de memoria.

```
[ec2-user ~]$ for i in $(seq 1 150); do dd if=/dev/random of=$i bs=512k count=1;
memcp -s localhost $i; done
[ec2-user ~]$ memcached-tool localhost display
# Item_Size Max_age Pages Count Full? Evicted Evict_Time OOM
2 120B 0s 1 0 no 0 0 0
39 512.0K 4s 63 126 yes 24 2 0
[ec2-user ~]$ sudo systemctl status memcached.service
# memcached.service - memcached daemon
Loaded: loaded (/usr/lib/systemd/system/memcached.service; enabled; preset:
disabled)
Active: active (running) since Fri 2025-01-31 22:36:42 UTC; 7min ago
Main PID: 356294 (memcached)
Tasks: 10 (limit: 18907)
Memory: 66.7M
CPU: 203ms
CGroup: /system.slice/memcached.service
        ##356294 /usr/bin/memcached -p 11211 -u memcached -m 64 -c 1024 -l
127.0.0.1,::1

Jan 31 22:36:42 ip-1-2-3-4.us-west-2.compute.internal systemd[1]: Started
memcached.service - memcached daemon.
```

5. Utilice la propiedad `MemoryMax` para establecer un límite estricto para `memcached.service`, que, si se alcanza, activará OOM Killer. Se pueden configurar opciones adicionales para el servicio añadiéndolas a un archivo de anulación. Esto se puede hacer editando directamente el archivo `/etc/systemd/system/memcached.service.d/override.conf` o de forma interactiva utilizando el comando `edit` de `systemctl`.

```
[ec2-user ~]$ sudo systemctl edit memcached.service
```

Añada lo siguiente a la modificación para establecer un límite máximo de 32 MB de memoria para el servicio.

```
[Service]
MemoryMax=32M
```

6. Indique a systemd que vuelva a cargar su configuración

```
[ec2-user ~]$ sudo systemctl daemon-reload
```

7. Observe que ahora se está ejecutando memcached.service con un límite de memoria de 32 MB.

```
[ec2-user ~]$ sudo systemctl status memcached.service
# memcached.service - memcached daemon
   Loaded: loaded (/usr/lib/systemd/system/memcached.service; enabled; preset:
disabled)
   Drop-In: /etc/systemd/system/memcached.service.d
            ##override.conf
   Active: active (running) since Fri 2025-01-31 23:09:13 UTC; 49s ago
 Main PID: 358423 (memcached)
    Tasks: 10 (limit: 18907)
   Memory: 1.8M (max: 32.0M available: 30.1M)
      CPU: 25ms
   CGroup: /system.slice/memcached.service
           ##358423 /usr/bin/memcached -p 11211 -u memcached -m 64 -c 1024 -l
127.0.0.1,::1

Jan 31 23:09:13 ip-1-2-3-4.us-west-2.compute.internal systemd[1]: Started
memcached.service - memcached daemon.
```

8. El servicio funcionará normalmente si utiliza menos de 32 MB de memoria, lo que podemos comprobar cargando menos de 32 MB de datos aleatorios en la memoria caché y comprobando después el estado del servicio.

```
[ec2-user ~]$ for i in $(seq 1 30); do dd if=/dev/random of=$i bs=512k count=1;
memcp -s localhost $i; done
```

```
[ec2-user ~]$ sudo systemctl status memcached.service
# memcached.service - memcached daemon
   Loaded: loaded (/usr/lib/systemd/system/memcached.service; enabled; preset:
disabled)
   Drop-In: /etc/systemd/system/memcached.service.d
            ##override.conf
   Active: active (running) since Fri 2025-01-31 23:14:48 UTC; 3s ago
 Main PID: 359492 (memcached)
    Tasks: 10 (limit: 18907)
   Memory: 18.2M (max: 32.0M available: 13.7M)
```



```
CPU: 42ms
CGroup: /system.slice/memcached.service
        ##359492 /usr/bin/memcached -p 11211 -u memcached -m 64 -c 1024 -l
127.0.0.1,::1
```

```
Jan 31 23:14:48 ip-1-2-3-4.us-west-2.compute.internal systemd[1]: Started
memcached.service - memcached daemon.
```

9. Ahora podemos hacer que memcached utilice más de 32 MB de memoria intentando utilizar los 64 MB completos de caché que tiene la configuración predeterminada de memcached.

```
[ec2-user ~]$ for i in $(seq 1 150); do dd if=/dev/random of=$i bs=512k count=1;
memcp -s localhost $i; done
```

Observará que en algún momento del comando anterior se producen errores de conexión con el servidor memcached. Esto se debe a que OOM Killer ha cancelado el proceso debido a la restricción que le pusimos. El resto del sistema funcionará con normalidad y el OOM Killer no tendrá en cuenta ningún otro proceso, ya que solo hemos restringido memcached.service.

```
[ec2-user ~]$ sudo systemctl status memcached.service
# memcached.service - memcached daemon
   Loaded: loaded (/usr/lib/systemd/system/memcached.service; enabled; preset:
disabled)
   Drop-In: /etc/systemd/system/memcached.service.d
            ##override.conf
   Active: failed (Result: oom-kill) since Fri 2025-01-31 23:20:28 UTC; 2s ago
 Duration: 2.901s
   Process: 360130 ExecStart=/usr/bin/memcached -p ${PORT} -u ${USER} -m
${CACHESIZE} -c ${MAXCONN} $OPTIONS (code=killed, signal=KILL)
  Main PID: 360130 (code=killed, signal=KILL)
     CPU: 94ms
```

```
Jan 31 23:20:25 ip-1-2-3-4.us-west-2.compute.internal systemd[1]: Started
memcached.service - memcached daemon.
Jan 31 23:20:28 ip-1-2-3-4.us-west-2.compute.internal systemd[1]:
memcached.service: A process of this unit has been killed by the OOM killer.
Jan 31 23:20:28 ip-1-2-3-4.us-west-2.compute.internal systemd[1]:
memcached.service: Main process exited, code=killed, status=9/KILL
Jan 31 23:20:28 ip-1-2-3-4.us-west-2.compute.internal systemd[1]:
memcached.service: Failed with result 'oom-kill'.
```

Limitación del uso de recursos de procesos en AL2023 mediante cgroups

Si bien se recomienda usar [Control de recursos con systemd](#), en esta sección se describe el uso básico de las utilidades básicas `libcgroup-tools` para limitar el uso de la CPU y la memoria de los procesos. Ambos métodos son alternativas al uso de la utilidad `cpulimit`, que anteriormente se encontraba en [EPEL](#).

El siguiente ejemplo describe la ejecución de la prueba de `stress-ng` estrés (desde el paquete `stress-ng`) y, al mismo tiempo, limitar el uso de la CPU y la memoria mediante las utilidades del paquete `libcgroup-tools` y los ajustes incluidos `sysfs`.

Use **`libcgroup-tools`** en la línea de comandos para limitar el uso de recursos

1. Instale el paquete `libcgroup-tools`.

```
[ec2-user ~]$ sudo dnf install libcgroup-tools
```

2. Cree un cgroup con los controladores `memory` y `cpu`, y asígnele un nombre (`our-example-limits`). Utilice las opciones `-a` y `-t` para permitir al usuario `ec2-user` controlar los ajustes de cgroup

```
[ec2-user ~]$ sudo cgcreate -a ec2-user -t ec2-user -g memory,cpu:our-example-limits
```

Ahora hay un directorio `/sys/fs/cgroup/our-example-limits/` que contiene archivos que se pueden usar para controlar cada uno de los parámetros ajustables.

Note

Amazon Linux 2 usa `cgroup-v1` en lugar de `cgroup-v2` que se usa en AL2023. En AL2, las rutas `sysfs` son diferentes, y habrá directorios `/sys/fs/cgroup/memory/our-example-limits` y `/sys/fs/cgroup/cpu/our-example-limits` propiedad de `ec2-user` que contienen archivos que se pueden usar para controlar los límites de cgroup.

3. Limite el uso de memoria de todos los procesos de nuestro cgroup a 100 millones de bytes.

```
[ec2-user ~]$ echo 100000000 > /sys/fs/cgroup/our-example-limits/memory.max
```

Note

Amazon Linux 2 usa cgroup-v1 en lugar de cgroup-v2 que usa Amazon Linux 2023. Esto significa que algunos parámetros ajustables son diferentes. Para limitar el uso de memoria en AL2, se utiliza en su lugar el siguiente parámetro ajustable.

```
[ec2-user ~]$ echo 10000000 > /sys/fs/cgroup/memory/our-example-limits/  
memory.limit_in_bytes
```

4. Limite el uso de la CPU de todos los procesos de nuestro cgroup al 10 %. El formato del archivo `cpu.max` es `$MAX $PERIOD`, lo que limita al grupo a consumir `$MAX` por cada `$PERIOD`.

```
[ec2-user ~]$ echo 10000 100000 > /sys/fs/cgroup/our-example-limits/cpu.max
```

Amazon Linux 2 usa cgroup-v1 en lugar de cgroup-v2 que usa Amazon Linux 2023. Esto significa que algunos parámetros ajustables son diferentes, incluida la forma de limitar el uso de la CPU.

5. El siguiente ejemplo ejecuta `stress-ng` (que se puede instalar ejecutando `dnf install -y stress-ng`) en el cgroup de `our-example-limits`. Mientras se ejecuta el comando `stress-ng`, se puede observar con `top` que está limitado al 10 % del tiempo de CPU.

```
[ec2-user ~]$ sudo cgexec -g memory,cpu:our-example-limits stress-ng --cpu 1
```

6. Haga una limpieza eliminando el cgroup

```
[ec2-user ~]$ sudo cgdelete -g memory,cpu:our-example-limits
```

La [documentación del kernel de Linux para cgroup-v2](#) contiene amplios detalles sobre su funcionamiento. La documentación de los controladores de [cpu](#) y [memory](#) describe los detalles de cómo usar cada opción ajustable.

Uso AL2023 en AWS

Puede configurarlo AL2023 para usarlo con otros Servicios de AWS. Por ejemplo, puede elegir una AL2023 AMI al lanzar una instancia de [Amazon Elastic Compute Cloud](#) (Amazon EC2).

Para estos procedimientos de configuración, utiliza el servicio AWS Identity and Access Management (IAM). Para obtener información completa sobre IAM, consulte los siguientes materiales de referencia:

- [AWS Identity and Access Management \(IAM\)](#)
- [Guía del usuario de IAM](#)

Temas

- [Empezando con AWS](#)
- [AL2023 en Amazon EC2](#)
- [Uso AL2023 en contenedores](#)
- [AL2023 en AWS Elastic Beanstalk](#)
- [Uso de AL2023 en AWS CloudShell](#)
- [Uso de AMI de Amazon ECS basadas en AL2023 para alojar cargas de trabajo en contenedores](#)
- [Uso de Amazon Elastic File System en AL2023](#)
- [Uso de Amazon EMR integrado AL2023](#)
- [Uso AL2023 en AWS Lambda](#)

Empezando con AWS

Inscríbase en una Cuenta de AWS

Si no tiene una Cuenta de AWS, complete los siguientes pasos para crearlo.

Para suscribirse a una Cuenta de AWS

1. Abrir <https://portal.aws.amazon.com/billing/registro>.
2. Siga las instrucciones que se le indiquen.

Parte del procedimiento de registro consiste en recibir una llamada telefónica o mensaje de texto e indicar un código de verificación en el teclado del teléfono.

Cuando te registras en un Cuenta de AWS, Usuario raíz de la cuenta de AWS se crea un. El usuario raíz tendrá acceso a todos los Servicios de AWS y recursos de esa cuenta. Como práctica recomendada de seguridad, asigne acceso administrativo a un usuario y utilice únicamente el usuario raíz para realizar [Tareas que requieren acceso de usuario raíz](#).

AWS te envía un correo electrónico de confirmación una vez finalizado el proceso de registro. En cualquier momento, puede ver la actividad de su cuenta actual y administrarla accediendo a <https://aws.amazon.com/> y seleccionando Mi cuenta.

Creación de un usuario con acceso administrativo

Después de crear un usuario administrativo Cuenta de AWS, asegúrelo Usuario raíz de la cuenta de AWS AWS IAM Identity Center, habilite y cree un usuario administrativo para no usar el usuario root en las tareas diarias.

Proteja su Usuario raíz de la cuenta de AWS

1. Inicie sesión [Consola de administración de AWS](#) como propietario de la cuenta seleccionando el usuario root e introduciendo su dirección de Cuenta de AWS correo electrónico. En la siguiente página, escriba su contraseña.

Para obtener ayuda para iniciar sesión con el usuario raíz, consulte [Iniciar sesión como usuario raíz](#) en la Guía del usuario de AWS Sign-In .

2. Active la autenticación multifactor (MFA) para el usuario raíz.

Para obtener instrucciones, consulte [Habilitar un dispositivo MFA virtual para el usuario Cuenta de AWS raíz \(consola\)](#) en la Guía del usuario de IAM.

Creación de un usuario con acceso administrativo

1. Activar IAM Identity Center.

Consulte las instrucciones en [Activar AWS IAM Identity Center](#) en la Guía del usuario de AWS IAM Identity Center .

2. En IAM Identity Center, conceda acceso administrativo a un usuario.

Para ver un tutorial sobre su uso Directorio de IAM Identity Center como fuente de identidad, consulte [Configurar el acceso de los usuarios con la configuración predeterminada Directorio de IAM Identity Center en la](#) Guía del AWS IAM Identity Center usuario.

Inicio de sesión como usuario con acceso de administrador

- Para iniciar sesión con el usuario de IAM Identity Center, use la URL de inicio de sesión que se envió a la dirección de correo electrónico cuando creó el usuario de IAM Identity Center.

Para obtener ayuda para iniciar sesión con un usuario del Centro de identidades de IAM, consulte [Iniciar sesión en el portal de AWS acceso](#) en la Guía del AWS Sign-In usuario.

Concesión de acceso a usuarios adicionales

1. En IAM Identity Center, cree un conjunto de permisos que siga la práctica recomendada de aplicar permisos de privilegios mínimos.

Para conocer las instrucciones, consulte [Create a permission set](#) en la Guía del usuario de AWS IAM Identity Center .

2. Asigne usuarios a un grupo y, a continuación, asigne el acceso de inicio de sesión único al grupo.

Para conocer las instrucciones, consulte [Add groups](#) en la Guía del usuario de AWS IAM Identity Center .

Concesión de acceso mediante programación

Los usuarios necesitan acceso programático si quieren interactuar con personas AWS ajenas a. Consola de administración de AWS La forma de conceder el acceso programático depende del tipo de usuario que acceda. AWS

Para conceder acceso programático a los usuarios, elija una de las siguientes opciones.

¿Qué usuario necesita acceso programático?	Para	Mediante
IAM	(Recomendado) Utilice las credenciales de la consola como credenciales temporales para firmar las solicitudes programáticas dirigidas al AWS CLI AWS SDKs, o. AWS APIs	Siga las instrucciones de la interfaz que desea utilizar: • Para ello AWS CLI, consulte Iniciar sesión para el desarrollo AWS local en la Guía del AWS Command Line Interface usuario. • Para ello AWS SDKs, consulte Iniciar sesión para el desarrollo AWS local en la Guía de referencia de AWS SDKs and Tools.
Identidad del personal (Usuarios administrados en el IAM Identity Center)	Utilice credenciales temporales para firmar las solicitudes programáticas dirigidas al AWS CLI, AWS SDKs, o AWS APIs.	Siga las instrucciones de la interfaz que desea utilizar: • Para ello AWS CLI, consulte Configuración del AWS CLI uso AWS IAM Identity Center en la Guía del AWS Command Line Interface usuario. • Para AWS SDKs ver las herramientas y AWS APIs, consulte la autenticación del Centro de Identidad de IAM en la Guía de referencia de herramientas AWS SDKs y herramientas.
IAM	Utilice credenciales temporales para firmar las solicitudes programáticas dirigidas al	Siga las instrucciones de Uso de credenciales temporales

¿Qué usuario necesita acceso programático?	Para	Mediante
	AWS CLI AWS SDKs, o. AWS APIs	con AWS recursos de la Guía del usuario de IAM.
IAM	(No recomendado) Utilice credenciales de larga duración para firmar las solicitudes programáticas dirigidas al AWS CLI, AWS SDKs, o. AWS APIs	Siga las instrucciones de la interfaz que desea utilizar: • Para ello AWS CLI, consulte Autenticación con credenciales de usuario de IAM en la Guía del AWS Command Line Interface usuario. • Para obtener AWS SDKs información sobre las herramientas, consulte Autenticarse con credenciales de larga duración en la Guía de referencia de herramientas AWS SDKs y herramientas. • Para ello AWS APIs, consulte Administrar las claves de acceso para los usuarios de IAM en la Guía del usuario de IAM.

AL2023 en Amazon EC2

Utilice uno de los siguientes procedimientos para lanzar una instancia de Amazon EC2 con una AMI AL2023 . Puede elegir entre la AMI estándar o la AMI mínima. Para obtener más información acerca de las diferencias entre la AMI estándar y la AMI mínima, consulte [Comparación entre AL2023 estándar \(predeterminado\) y mínimo AMIs](#).

Temas

- [Lanzamiento de AL2023 con la consola de Amazon EC2](#)
- [Inicio AL2023 mediante el parámetro SSM y AWS CLI](#)
- [Lanzamiento de la AL2023 AMI más reciente mediante CloudFormation](#)
- [Lanzamiento AL2023 mediante un ID de AMI específico](#)
- [AL2023 Obsolescencia y ciclo de vida de la AMI](#)
- [Conexión a instancias AL2023](#)
- [Comparación entre AL2023 estándar y mínimo AMIs](#)

Lanzamiento de AL2023 con la consola de Amazon EC2

Utilice la consola Amazon EC2 para lanzar una AMI AL2023 .

Note

En el caso de las instancias basadas en ARM, AL2023 solo admite los tipos de instancias que utilizan procesadores Graviton2 o posteriores. AL2023 no admite instancias A1.

Lleve a cabo los pasos que se indican a continuación para lanzar una instancia de Amazon EC2 con una AMI de AL2023 desde la consola de Amazon EC2.

Para lanzar una instancia EC2 con una AMI AL2023

1. Abra la consola de Amazon EC2 en <https://console.aws.amazon.com/ec2/>.
2. En el panel de navegación, elija AMIs.
3. Elija Imágenes públicas en el menú desplegable.
4. En el campo de búsqueda, escriba **al2023-ami**.

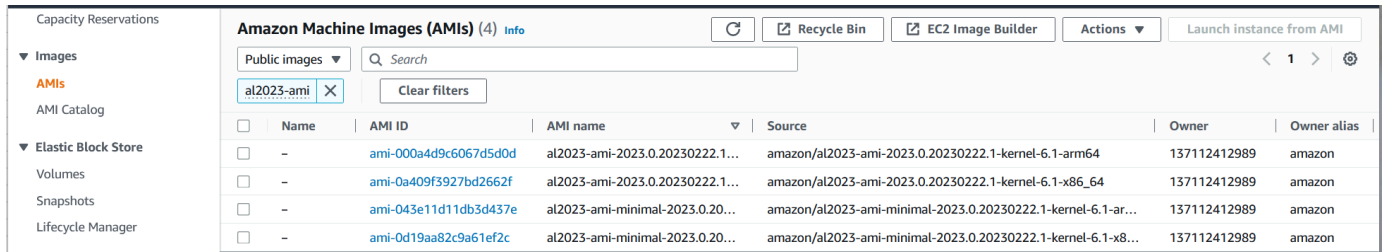
Note

Asegúrese de que Amazon aparezca en la columna Alias del propietario.

5. Seleccione una imagen en la lista. En Origen, puede determinar si la AMI es estándar o mínima. El nombre de una AL2023 AMI se puede interpretar con este formato:

```
'al2023-[ami || ami-minimal]-2023.0.[release build date].[build number]-kernel-[version number]-[arm64 || x86_64]'
```

6. La siguiente imagen muestra una lista parcial de AL2023 AMIs.



The screenshot shows the Amazon Machine Images (AMIs) console. On the left, there is a navigation menu with options like Capacity Reservations, Images, AMIs, AMI Catalog, Elastic Block Store, Volumes, Snapshots, and Lifecycle Manager. The main panel displays 'Amazon Machine Images (AMIs) (4)' with a search bar and a filter dropdown set to 'Public images'. A table lists several AMIs, with the first four being AL2023 AMIs. The table has columns for Name, AMI ID, AMI name, Source, Owner, and Owner alias.

Name	AMI ID	AMI name	Source	Owner	Owner alias
-	ami-000a4d9c6067d5d0d	al2023-ami-2023.0.20230222.1...	amazon/al2023-ami-2023.0.20230222.1-kernel-6.1-arm64	137112412989	amazon
-	ami-0a409f3927bd2662f	al2023-ami-2023.0.20230222.1...	amazon/al2023-ami-2023.0.20230222.1-kernel-6.1-x86_64	137112412989	amazon
-	ami-043e11d11db3d437e	al2023-ami-minimal-2023.0.20...	amazon/al2023-ami-minimal-2023.0.20230222.1-kernel-6.1-ar...	137112412989	amazon
-	ami-0d19aa82c9a61ef2c	al2023-ami-minimal-2023.0.20...	amazon/al2023-ami-minimal-2023.0.20230222.1-kernel-6.1-x8...	137112412989	amazon

Para obtener más información sobre cómo lanzar instancias de Amazon EC2, consulte [Introducción a las instancias Linux de Amazon EC2](#) en la Guía de usuario de Amazon EC2.

Inicio AL2023 mediante el parámetro SSM y AWS CLI

En el AWS CLI, puede utilizar el valor del parámetro SSM de una AMI para lanzar una nueva instancia de. AL2023 Más específicamente, utilice uno de los valores de los parámetros de SSM dinámicos de la siguiente lista y agregue `/aws/service/ami-amazon-linux-latest/` antes del valor/ del parámetro de SSM. Puede utilizar eso para lanzar una instancia en la AWS CLI.

- `al2023-ami-kernel-default-arm64` para la arquitectura arm64
- `al2023-ami-minimal-kernel-default-arm64` para la arquitectura arm64 (AMI mínima)
- `al2023-ami-kernel-default-x86_64` para la arquitectura x86_64
- `al2023-ami-minimal-kernel-default-x86_64` para la arquitectura x86_64 (AMI mínima)

Note

Cada uno de los *italic* elementos es un parámetro de ejemplo. Reemplácelos con su propia información.

```
$ aws ec2 run-instances \
  --image-id \
    resolve:ssm:/aws/service/ami-amazon-linux-latest/al2023-ami-kernel-default-x86_64 \
  --instance-type m5.xlarge \
  --region us-east-1 \
  --key-name aws-key-us-east-1 \
  --security-group-ids sg-004a7650
```

El indicador `--image-id` especifica el valor del parámetro SSM.

El indicador `--instance-type` especifica el tipo y el tamaño de la instancia. Este indicador debe ser compatible con el tipo de AMI que haya seleccionado.

La `--region` marca específica el Región de AWS lugar donde se crea la instancia.

El `--key-name` indicador especifica Región de AWS la clave que se usa para conectarse a la instancia. Si no proporciona una clave que exista en la región en la que creó la instancia, no podrá conectarse a la instancia mediante SSH.

El indicador `--security-group-ids` especifica el grupo de seguridad que determina los permisos de acceso para el tráfico de red entrante y saliente.

Important

AWS CLI Requiere que especifiques un grupo de seguridad existente que permita el acceso a la instancia desde tu máquina remota a través del puerto TCP:22. Sin un grupo de seguridad específico, la nueva instancia se ubica en un grupo de seguridad predeterminado. En un grupo de seguridad predeterminado, la instancia solo se puede conectar con las demás instancias de la VPC.

Para obtener más información, consulte [Lanzamiento, enumeración y finalización de instancias de Amazon EC2](#) en la Guía del usuario de AWS Command Line Interface .

Lanzamiento de la AL2023 AMI más reciente mediante CloudFormation

Para lanzar una AL2023 AMI mediante CloudFormation, utilice una de las siguientes plantillas.

Note

Las x86_64 y Arm64 AMIs cada una de ellas requieren tipos de instancias diferentes. Para obtener más información, consulte [Tipos de instancia de Amazon EC2](#)

Plantilla de JSON:

```
{  
  "Parameters": {
```

```

    "LatestAmiId": {
      "Type": "AWS::SSM::Parameter::Value<AWS::EC2::Image::Id>",
      "Default": "/aws/service/ami-amazon-linux-latest/al2023-ami-minimal-kernel-
default-x86_64"
    }
  },
  "Resources": {
    "MyEC2Instance": {
      "Type": "AWS::EC2::Instance",
      "Properties": {
        "InstanceType": "t2.large",
        "ImageId": {
          "Ref": "LatestAmiId"
        }
      }
    }
  }
}

```

Plantilla de YAML:

```

Parameters:
  LatestAmiId:
    Type: 'AWS::SSM::Parameter::Value<AWS::EC2::Image::Id>'
    Default: '/aws/service/ami-amazon-linux-latest/al2023-ami-minimal-kernel-default-
x86_64'

Resources:
  Instance:
    Type: 'AWS::EC2::Instance'
    Properties:
      InstanceType: 't2.large'
      ImageId: !Ref LatestAmiId

```

Asegúrese de reemplazar el parámetro AMI al final de la sección “Predeterminado”, si es necesario. Los valores siguientes están disponibles con:

- al2023-ami-kernel-6.1-arm64 para la arquitectura arm64
- al2023-ami-minimal-kernel-6.1-arm64 para la arquitectura arm64 (AMI mínima)
- al2023-ami-kernel-6.1-x86_64 para la arquitectura x86_64
- al2023-ami-minimal-kernel-6.1-x86_64 para la arquitectura x86_64 (AMI mínima)

A continuación se muestran las especificaciones dinámicas del kernel. La versión predeterminada del kernel cambia automáticamente con cada actualización de la versión principal del kernel.

- `al2023-ami-kernel-default-arm64` para la arquitectura `arm64`
- `al2023-ami-minimal-kernel-default-arm64` para la arquitectura `arm64` (AMI mínima)
- `al2023-ami-kernel-default-x86_64` para la arquitectura `x86_64`
- `al2023-ami-minimal-kernel-default-x86_64` para la arquitectura `x86_64` (AMI mínima)

Lanzamiento AL2023 mediante un ID de AMI específico

Puede lanzar una AMI AL2023 específica mediante el ID de la AMI. Puede determinar qué ID de AMI de AL2023 es necesario consultando la lista de AMI de la consola Amazon EC2. O bien, puede usar AWS Systems Manager. Si utiliza Systems Manager, asegúrese de seleccionar el alias de AMI de entre los que se muestran en la sección anterior. Para obtener más información, consulte [Buscar la AMI más reciente de Amazon Linux IDs mediante AWS Systems Manager Parameter Store](#).

AL2023 Obsolescencia y ciclo de vida de la AMI

Cada nueva AL2023 versión incluye una AMI nueva. Cuando se registra la AMI, se marca con una fecha de caducidad. La fecha de caducidad de cada AL2023 AMI es de 90 días a partir del momento en que se publicó para coincidir con el período de tiempo que [Aplicación de parches activos del kernel en AL2023](#) se ofrece para cada versión individual del núcleo.

Note

La fecha de caducidad de 90 días se refiere a una AMI individual y no al AL2023 [Liberar cadencia](#) período de soporte del producto.

Para obtener más información sobre la obsolescencia de las AMI, consulte [Desaprobar una AMI](#) en la Guía del usuario de Amazon EC2.

El uso regular de una AMI actualizada para lanzar una instancia garantiza que la instancia comience con las actualizaciones de seguridad más recientes, incluido un código kernel actualizado. Si lanza una versión anterior de una AMI y la actualiza, hay un período de tiempo en el que la instancia no tiene las actualizaciones de seguridad más recientes. Para asegurarse de que está utilizando la AMI más reciente, le recomendamos que utilice los parámetros de SSM.

Para obtener más información sobre el uso de los parámetros de SSM para lanzar una instancia, consulte:

- [Inicio AL2023 mediante el parámetro SSM y AWS CLI](#)
- [Lanzamiento de la AL2023 AMI más reciente mediante CloudFormation](#)

Conexión a instancias AL2023

Usa SSH o AWS Systems Manager para conectarte a tu instancia. AL2023

Conexión a su instancia mediante SSH

Para obtener instrucciones sobre cómo utilizar SSH para conectarse a una instancia, consulte [Conexión a la instancia de Linux mediante SSH](#) en la Guía del usuario de Amazon EC2.

Conéctese a su instancia mediante AWS Systems Manager

Para obtener instrucciones sobre cómo conectarse AWS Systems Manager a una AL2023 instancia, consulte [Conectarse a su instancia de Linux mediante el administrador de sesiones](#) en la Guía del usuario de Amazon EC2.

Uso de Amazon EC2 Instance Connect

La AMI de AL2023 (no la AMI mínima) viene con el agente EC2 Instance Connect instalado de forma predeterminada. Para utilizar EC2 Instance Connect con una AL2023 instancia lanzada desde la AMI mínima, debe instalar el `ec2-instance-connect` paquete. Para obtener instrucciones sobre cómo utilizar EC2 Instance Connect, consulte [Conexión a la instancia de Linux mediante EC2 Instance Connect](#) en la Guía del usuario de Amazon EC2.

Comparación entre AL2023 estándar y mínimo AMIs

Puede lanzar una instancia de Amazon EC2 con una AMI de AL2023 estándar (predeterminada) o mínima. Para obtener instrucciones sobre cómo lanzar una instancia de Amazon EC2 con el tipo AMI estándar o mínimo, consulte [AL2023 en Amazon EC2](#).

La AL2023 AMI estándar incluye todas las aplicaciones y herramientas más utilizadas.

Recomendamos la AMI estándar si desea empezar rápidamente y no está interesado en personalizar la AMI.

La AL2023 AMI mínima es la versión básica y simplificada que contiene solo las herramientas y utilidades más básicas necesarias para ejecutar el sistema operativo (SO). Recomendamos la AMI

mínima si desea tener el menor espacio posible en el sistema operativo. La AMI mínima ofrece una utilización ligeramente reducida del espacio en disco y una mejor rentabilidad a largo plazo. La AMI mínima es adecuada si desea un sistema operativo más pequeño y no le importa instalar herramientas y aplicaciones manualmente.

La imagen del contenedor se acerca más a la AMI AL2023 mínima del conjunto de paquetes.

Comparación de paquetes de imágenes instalados en Amazon Linux 2023

Una comparación del RPMs presente en las imágenes AL2023 AMI, Minimal AMI y Container.

Package	AMI	AMI mínima	Container
acl	2.3.1		
acpid	2.0.32		
alternatives	1.15	1.15	1.15
amazon-chrony-config	4.3	4.3	
amazon-ec2-net-utils	2.5.1	2.5.1	
amazon-linux-repo-cdn			2023,6 20241031
amazon-linux-repo-s3	2023,6 20241031	2023,6 20241031	
amazon-linux-sb-keys	2023,1	2023.1	
amazon-rpm-config	228		
amazon-ssm-agent	3.3.987,0		

Package	AMI	AMI mínima	Container
amd-ucode-firmware	20210208 (noarch)	20210208 (noarch)	
at	3.1,23		
attr	2.5.1		
audit	3.0.6	3.0.6	
audit-libs	3.0.6	3.0.6	3.0.6
aws-cfn-bootstrap	2.0		
awscli-2	2.15.30	2.15.30	
basesystem	11	11	11
bash	5.2.15	5.2.15	5.2.15
bash-completion	2.11		
bc	1.07.1		
bind-libs	9,18,28		
bind-license	9,18,28		
bind-utils	9,18,28		
binutils	2.39		
boost-fil esystem	1.75,0		
boost-system	1,75,0		
boost-thread	1,75,0		

Package	AMI	AMI mínima	Container
bzip2	1.0.8		
bzip2-libs	1.0.8	1.0.8	1.0.8
ca-certificates	2023,2,68	2023,2,68	2023,2,68
c-ares	1.19.1		
checkpolicy	3.4	3.4	
chkconfig	1.15		
chrony	4.3	4.3	
cloud-init	22.2.2	22.2.2	
cloud-init-cfg-ec2	22.2.2	22.2.2	
cloud-utils-growpart	0.31	0,31	
coreutils	8.32	8.32	
coreutils-common	8.32	8.32	
coreutils-single			8.32
cpio	2.13	2.13	
cracklib	2.9.6	2.9.6	
cracklib-dicts	2.9.6	2.9.6	
crontabs	1.11		
crypto-policies	20220428	20220428	20220428

Package	AMI	AMI mínima	Container
crypto-policies-scripts	20220428		
cryptsetup	2.6.1		
cryptsetup-libs	2.6.1	2.6.1	
curl-minimal	8.5.0	8.5.0	8.5.0
cyrus-sasl-lib	2.1.27	2.1.27	
cyrus-sasl-plain	2.1.27		
dbus	1.12.28	1.12.28	
dbus-broker	32	32	
dbus-common	1.12.28	1.12.28	
dbus-libs	1.12.28	1.12.28	
device-mapper	1.02.185	1.02.185	
device-mapper-libs	1.02.185	1.02.185	
diffutils	3.8	3.8	
dnf	4.14.0	4.14.0	4.14.0
dnf-data	4.14.0	4.14.0	4.14.0
dnf-plugin-release-notification	1.2	1.2	
dnf-plugins-core	4.3.0	4.3.0	

Package	AMI	AMI mínima	Container
dnf-plugin-support-info	1.2	1.2	
dnf-utils	4.3.0		
dosfstools	4.2		
dracut	055	055	
dracut-config-ec2	3.0	3.0	
dracut-config-generic	055	055	
dwz	0.14		
dyninst	10.2.1		
e2fsprogs	1.46,5	1,46.5	
e2fsprogs-libs	1,46.5	1,46.5	
ec2-hibinit-agent	1.0.8		
ec2-instance-connect	1.1		
ec2-instance-connect-selinux	1.1		
ec2-utils	2.2.0	2.2.0	
ed	1.14.2		
efi-filesystem	5	5	

Package	AMI	AMI mínima	Container
efi-srpm-macros	5		
efivar	38	38	
efivar-libs	38	38	
elfutils- debuginfod- client	0.188		
elfutils- default-yama- scope	0.188	0.188	0.188
elfutils-libelf	0.188	0.188	0.188
elfutils-libs	0.188	0.188	0.188
ethtool	5.15		
expat	2.5.0	2.5.0	2.5.0
file	5.39	5.39	
file-libs	5.39	5.39	5.39
filesystem	3.14	3.14	3.14
findutils	4.8.0	4.8.0	
fonts-srpm- macros	2.0.5		
fstrm	0.6.1		
fuse-libs	2.9.9	2.9.9	
gawk	5.1.0	5.1.0	5.1.0

Package	AMI	AMI mínima	Container
gdbm-libs	1.19	1.19	1.19
gdisk	1.0.8	1.0.8	
gettext	0.21	0,21	
gettext-libs	0,21	0,21	
ghc-srpm-macros	1.5.0		
glib2	2.74,7	2.74,7	2.74,7
glibc	2.34	2.34	2.34
glibc-all-langpacks	2.34	2.34	
glibc-common	2.34	2.34	2.34
glibc-gconv-extra	2.34		
glibc-locale-source	2.34	2.34	
glibc-minimal-langpack			2.34
gmp	6.2.1	6.2.1	6.2.1
gnupg2-minimal	2.3.7	2.3.7	2.3.7
gnutls	3.8.0	3.8.0	
go-srpm-macros	3.2.0		
gpgme	1.15.1	1.15.1	1.15.1
gpm-libs	1.20,7		

Package	AMI	AMI mínima	Container
grep	3.8	3.8	3.8
groff-base	1.22.4	1.22.4	
grub2-common	2.06	2.06	
grub2-efi-aa64-ec2	2.06 (aarch64)	2.06 (aarch64)	
grub2-efi-x64-ec2	2.06 (x86_64)	2.06 (x86_64)	
grub2-pc-modules	2.06	2.06	
grub2-tools	2.06	2.06	
grub2-tools-minimal	2.06	2.06	
grubby	8.40	8.40	
gssproxy	0.8.4		
gzip	1.12	1.12	
hostname	3.23	3.23	
hunspell	1.7.0		
hunspell-en	0,20140811,1		
hunspell-en-GB	0,20140811,1		
hunspell-en-US	0,20140811,1		
hunspell-filesystem	1.7.0		

Package	AMI	AMI mínima	Container
hwdata	0,384	0,384	
info	6.7		
inih	49	49	
initscripts	10,09	10,09	
iproute	6.10.0	6.10.0	
iputils	20210202	20210202	
irqbalance	1.9.0	1.9.0	
jansson	2.14	2.14	
jemalloc	5.2.1		
jitterentropy	3.4.1	3.4.1	
jq	1.7.1	1.7.1	
json-c	0,14	0.14	0.14
kbd	2.4.0	2.4.0	
kbd-misc	2.4.0	2.4.0	
kernel	6.1.112	6.1.112	
kernel-libbpf	6.1.112	6.1.112	
kernel- livepatch-repo- s3	2023,620241031	2023,6 20241031	
kernel-srpm- macros	1.0		

Package	AMI	AMI mínima	Container
kernel-tools	6.1.112		
keyutils	1.6.3		
keyutils-libs	1.6.3	1.6.3	1.6.3
kmod	29	29	
kmod-libs	29	29	
kpatch-runtime	0.9.7		
krb5-libs	1.21.3	1.21.3	1.21.3
less	608	608	
libacl	2.3.1	2.3.1	2.3.1
libaio	0.3.111		
libarchive	3.7.4	3.7.4	3.7.4
libargon2	20171227	20171227	
libassuan	2.5.5	2.5.5	2.5.5
libattr	2.5.1	2.5.1	2.5.1
libbasicobjects	0.1.1		
libblkid	2.37,4	2.37,4	2.37,4
libcap	2.48	2.48	2.48
libcap-ng	0.8.2	0.8.2	0.8.2
libcbor	0.7.0	0.7.0	
libcollection	0.7.0		

Package	AMI	AMI mínima	Container
libcom_err	1.46,5	1,46.5	1,46.5
libcomps	0.1,20	0.1,20	0.1,20
libconfig	1.7.2		
libcurl-minimal	8.5.0	8.5.0	8.5.0
libdb	5.3.28	5.3.28	
libdhash	0.5.0		
libdnf	0.69,0	0.69,0	0.69,0
libeconf	0.4.0	0.4.0	
libedit	3.1	3.1	
libev	4.33		
libevent	2.1.12		
libfdisk	2.37,4	2.37,4	
libffi	3.4.4	3.4.4	3.4.4
libfido2	1.10.0	1.10.0	
libgcc	11.4.1	11.4.1	11.4.1
libgcrypt	1.10.2	1.10.2	1.10.2
libgomp	11.4.1	11.4.1	11.4.1
libgpg-error	1.42	1,42	1,42
libibverbs	48,0		
libidn2	2.3.2	2.3.2	2.3.2

Package	AMI	AMI mínima	Container
libini_config	1.3.1		
libkcap	1.4.0	1.4.0	
libkcap-hmacalc	1.4.0	1.4.0	
libldb	2.6.2		
libmaxminddb	1.5.2		
libmetalink	0.1.3		
libmnl	1.0.4	1.0.4	
libmodulemd	2.13.0	2.13.0	2.13.0
libmount	2.37,4	2.37,4	2.37,4
libnfsidmap	2.5.4		
libnghttp2	1.59,0	1,59,0	1,59,0
libnl3	3.5.0		
libpath_utils	0.2.1		
libpcap	1.10.1		
libpipeline	1.5.3	1.5.3	
libpkgconf	1.8.0		
libpsl	0.21.1	0.21.1	0.21.1
libpwquality	1.4.4	1.4.4	
libref_array	0.1.5		
librepo	1.14.5	1.14.5	1.14.5

Package	AMI	AMI mínima	Container
libreport-filesystem	2.15.2	2.15.2	2.15.2
libseccomp	2.5.3	2.5.3	
libselinux	3.4	3.4	3.4
libselinux-utils	3.4	3.4	
libsemanage	3.4	3.4	
libsepol	3.4	3.4	3.4
libsigsegv	2.13	2.13	2.13
libsmartcols	2.37,4	2.37,4	2.37,4
libsolv	0.7.22	0.7.22	0.7.22
libss	1.46,5	1,46.5	
libsss_certmap	2.9.4		
libsss_idmap	2.9.4		
libsss_nss_idmap	2.9.4		
libsss_sudo	2.9.4		
libstdc++	11.4.1	11.4.1	11.4.1
libstoragemgmt	1.9.4		
libtalloc	2.3.4		
libtasn1	4.19,0	4.19,0	4.19,0

Package	AMI	AMI mínima	Container
libtdb	1.4.7		
libtevent	0.13.0		
libtextstyle	0,21	0,21	
libtirpc	1.3.3		
libunistring	0,9,10	0,9,10	0,9,10
libuser	0,63	0.63	
libutempter	1.2.1	1.2.1	
libuuid	2.37,4	2.37,4	2.37,4
libuv	1.47.0		
libverto	0.3.2	0.3.2	0.3.2
libverto-libev	0.3.2		
libxcrypt	4.4.33	4.4.33	4.4.33
libxml2	2.10.4	2.10.4	2.10.4
libyaml	0.2.5	0.2.5	0.2.5
libzstd	1.5.5	1.5.5	1.5.5
linux-firmware-whence	20210208 (noarch)	20210208 (noarch)	
lm_sensors-libs	3.6.0		
lmdb-libs	0.9.29		
logrotate	3.20.1	3.20.1	
lsof	4,94,0		

Package	AMI	AMI mínima	Container
lua-libs	5.4.4	5.4.4	5.4.4
lua-srpm-macros	1		
lz4-libs	1.9.4	1.9.4	1.9.4
man-db	2.9.3	2.9.3	
man-pages	5.10		
microcode_ctl	2.1 (x86_64)	2.1 (x86_64)	
mpfr	4.1.0	4.1.0	4.1.0
nano	5.8		
ncurses	6.2	6.2	
ncurses-base	6.2	6.2	6.2
ncurses-libs	6.2	6.2	6.2
nettle	3.8	3.8	
net-tools	2.0	2.0	
newt	0,52,21		
nfs-utils	2.5.4		
npth	1.6	1.6	1.6
nspr	4,35,0		
nss	3.90,0		
nss-softokn	3.90,0		
nss-softokn-freebl	3.90,0		

Package	AMI	AMI mínima	Container
nss-sysinit	3.90,0		
nss-util	3.90,0		
ntsysv	1.15		
numactl-libs	2.0.14	2.0.14	
ocaml-srpm-macros	6		
oniguruma	6.9.7.1	6.9.7.1	
openblas-srpm-macros	2		
openldap	2.4.57	2.4.57	
openssh	8.7p1	8.7p1	
openssh-clients	8.7p1	8.7p1	
openssh-server	8.7p1	8.7p1	
openssl	3.0.8	3.0.8	
openssl-libs	3.0.8	3.0.8	3.0.8
openssl-pkcs11	0.4.12	0.4.12	
os-prober	1.77	1.77	
p11-kit	0.24.1	0,24.1	0,24.1
p11-kit-trust	0,24.1	0,24.1	0,24.1
package-notes-srpm-macros	0.4		

Package	AMI	AMI mínima	Container
pam	1.5.1	1.5.1	
parted	3.4		
passwd	0,80	0,80	
pciutils	3.7.0	3.7.0	
pciutils-libs	3.7.0	3.7.0	
pcre2	10,40	10,40	10,40
pcre2-syntax	10,40	10,40	10,40
perl-Carp	1,50		
perl-Class-Struct	0.66		
perl-constant	1,33		
perl-DynaLoader	1,47		
perl-Encode	3.15		
perl-Errno	1,30		
perl-Exporter	5.74		
perl-Fcntl	1.13		
perl-File-Basename	2.85		
perl-File-Path	2.18		
perl-File-stat	1.09		
perl-File-Temp	0,231,100		

Package	AMI	AMI mínima	Container
perl-Getopt-Long	2.52		
perl-Getopt-Std	1.12		
perl-HTTP-Tiny	0,078		
perl-if	0,60.800		
perl-interpreter	5,32.1		
perl-IO	1,43		
perl-IPC-Open3	1.21		
perl-libs	5.32,1		
perl-MIME-Base64	3.16		
perl-mro	1.23		
perl-overload	1.31		
perl-overloading	0,02		
perl-parent	0,238		
perl-PathTools	3.78		
perl-Pod-Escapes	1,07		
perl-podlators	4.14		
perl-Pod-Perldoc	3.28,01		

Package	AMI	AMI mínima	Container
perl-Pod-Simple	3.42		
perl-Pod-Usage	2.01		
perl-POSIX	1,94		
perl-Scalar-List-Utils	1.56		
perl-SelectSaver	1.02		
perl-Socket	2,032		
perl-srpm-macros	1		
perl-Storable	3.21		
perl-subs	1.03		
perl-Symbol	1.08		
perl-Term-ANSIColor	5.01		
perl-Term-Cap	1,17		
perl-Text-ParseWords	3,30		
perl-Text-Tabs+Wrap	2021.0726		
perl-Time-Local	1.300		
perl-vars	1.05		
pkgconf	1.8.0		

Package	AMI	AMI mínima	Container
pkgconf-m4	1.8.0		
pkgconf-pkg-config	1.8.0		
polycoreutils	3.4	3.4	
polycoreutils-python-utils	3.4		
popt	1.18	1.18	1.18
procps-ng	3.3.17	3.3.17	
protobuf-c	1.4.1		
psacct	6.6.4		
psmisc	23.4	23.4	
publicsuffix-list-dafsa	20240212	20240212	20240212
python3	3.9.16	3.9.16	3.9.16
python3-attrs	203.0	203.0	
python3-audit	3.0.6	3.0.6	
python3-awscli	0.19.19	0.19.19	
python3-babel	2.9.1	2.9.1	
python3-cffi	1.14.5	1.14.5	
python3-chardet	4.0.0	4.0.0	

Package	AMI	AMI mínima	Container
python3-colorama	0.4.4	0.4.4	
python3-configobj	5.0.6	5.0.6	
python3-cryptography	36,01	36,01	
python3-daemon	2.3.0		
python3-dateutil	2.8.1	2.8.1	
python3-dbus	1.2.18	1.2.18	
python3-distro	1.5.0	1.5.0	
python3-dnf	4.14.0	4.14.0	4.14.0
python3-dnf-plugins-core	4.3.0	4.3.0	
python3-docutils	0,16	0,16	
python3-gpg	1.15.1	1.15.1	1.15.1
python3-hawkey	0.69,0	0.69,0	0.69,0
python3-idna	(2.10)	(2.10)	
python3-jinja2	2.11.3	2.11.3	
python3-jmespath	0.10.0	0.10.0	

Package	AMI	AMI mínima	Container
python3-j sonpatch	1.21	1.21	
python3-j sonpointer	2.0	2.0	
python3-j sonschema	3.2.0	3.2.0	
python3-l ibcomps	0.1.20	0.1,20	0.1,20
python3-libdnf	0.69,0	0.69,0	0.69,0
python3-libs	3,9,16	3,9,16	3,9,16
python3-l ibselinux	3.4	3.4	
python3-l ibsemanage	3.4	3.4	
python3-l ibstoragemgmt	1.9.4		
python3-l ockfile	0.12.2		
python3-m arkupsafe	1.1.1	1.1.1	
python3-n etifaces	0.10.6	0.10.6	
python3-o authlib	3.0.2	3.0.2	

Package	AMI	AMI mínima	Container
python3-pip-wheel	21.3.1	21.3.1	21.3.1
python3-ply	3.11	3.11	
python3-policycoreutils	3.4	3.4	
python3-prettytable	0.7.2	0.7.2	
python3-prompt-toolkit	3.0.24	3,0,24	
python3-pyparser	2.20	2.20	
python3-pyrsistent	0,17.3	0,17.3	
python3-pyserial	3.4	3.4	
python3-pysocks	1.7.1	1.7.1	
python3-pytz	2022.7.1	2022.7.1	
python3-pyyaml	5.4.1	5.4.1	
python3-requests	2.25.1	2.25.1	
python3-rpm	4.16.1.3	4.16.1.3	4.16.1.3
python3-ruamel-yaml	0,16,6	0,16,6	

Package	AMI	AMI mínima	Container
python3-ruamel-yaml-clib	0.1.2	0.1.2	
python3-setools	4.4.1	4.4.1	
python3-setuptools	59,6,0	59,6,0	
python3-setuptools-wheel	59,6,0	59,6,0	59,6,0
python3-six	1.15.0	1.15.0	
python3-systemd	235	235	
python3-urllib3	1,25,10	1,25,10	
python3-wcwidth	0.2.5	0.2.5	
python-chevron	0.13.1		
python-srpm-macros	3.9		
quota	4.06		
quota-nls	4.06		
readline	8.1	8.1	8.1
rng-tools	6.14	6.14	
rootfiles	8.1	8.1	
rpcbind	1.2.6		
rpm	4.16.1.3	4.16.1.3	4.16.1.3

Package	AMI	AMI mínima	Container
rpm-build-libs	4.16.1.3	4.16.1.3	4.16.1.3
rpm-libs	4.16.1.3	4.16.1.3	4.16.1.3
rpm-plugin-selinux	4.16.1.3	4.16.1.3	
rpm-plugin-systemd-inhibit	4.16.1.3	4.16.1.3	
rpm-sign-libs	4.16.1.3	4.16.1.3	4.16.1.3
rsync	3.2.6		
rust-srpm-macros	21		
sbsigntools	0.9.4	0.9.4	
screen	4.8.0		
sed	4.8	4.8	4.8
selinux-policy	381,45	381,45	
selinux-policy-targeted	381,45	381,45	
setup	2.13.7	2.13.7	2.13.7
shadow-utils	4.9	4.9	
slang	2.3.2		
sqlite-libs	3.40,0	3.40,0	3.40,0
sssd-client	2.9.4		
sssd-common	2.9.4		

Package	AMI	AMI mínima	Container
sssd-kcm	2.9.4		
sssd-nfs-idmap	2.9.4		
strace	6.8		
sudo	1.9.15	1.9.15	
sysctl-defaults	1.0	1.0	
sysstat	12.5.6		
systemd	252,23	252,23	
systemd-libs	252,23	252,23	
systemd-networkd	252,23	252,23	
systemd-pam	252,23	252,23	
systemd-resolved	252,23	252,23	
systemd-udev	252,23	252,23	
system-release	2023,6 20241031	2023,6 20241031	2023,6 20241031
systemtap-runtime	4.8		
tar	1,34	1,34	
tbb	2020,3		
tcpdump	4,99,1		
tcsh	6.24,07		

Package	AMI	AMI mínima	Container
time	1.9		
traceroute	2.1.3		
tzdata	2024a	2024a	2024a
unzip	6.0		
update-motd	2.2	2.2	
userspace-rcu	0.12.1	0.12.1	
util-linux	2.37,4	2.37,4	
util-linux-core	2.37,4	2.37,4	
vim-common	9,0,2153		
vim-data	9,0,2153	9,0,2153	
vim-enhanced	9,0,2153		
vim-filesystem	9,0,2153		
vim-minimal	9,0,2153	9,0,2153	
wget	1.21.3		
which	2.21	2.21	
words	3.0		
xfsdump	3.1.11		
xfspgrog	5.18,0	5.18,0	
xxd	9,0,2153		
xxhash-libs	0.8.0		

Package	AMI	AMI mínima	Container
xz	5.2.5	5.2.5	
xz-libs	5.2.5	5.2.5	5.2.5
yum	4.14.0	4.14.0	4.14.0
zip	3.0		
zlib	1.2.11	1.2.11	1.2.11
zram-generator	1.1.2	1.1.2	
zram-generator-defaults	1.1.2	1.1.2	
zstd	1.5.5	1.5.5	

Uso AL2023 en contenedores

Note

Para obtener más información sobre cómo alojar cargas AL2023 de trabajo en contenedores en Amazon ECS, consulte. [AL2023 para hosts de contenedores de Amazon ECS](#)

Existen varias formas de utilizarlas AL2023 dentro de los contenedores, según el caso de uso.

[AL2023 Imagen del contenedor base](#) Es muy similar a una imagen de contenedor de Amazon Linux 2 y a la AMI AL2023 mínima.

Para los usuarios avanzados, ofrecemos una imagen de contenedor mínima, presentada en la versión AL2023 .2, junto con documentación que describe cómo crear [contenedores básicos](#).

AL2023 también se puede utilizar para alojar cargas de trabajo en contenedores, ya sean imágenes de contenedores o contenedores AL2023 basados en otras distribuciones de Linux. Puede usar [AL2023 para hosts de contenedores de Amazon ECS](#) o usar directamente los paquetes de tiempo de ejecución de contenedores que se proporcionan. Los `nerdctl` paquetes `dockercontainerd`, y están disponibles para su instalación y uso. AL2023

Temas

- [Uso de la imagen del contenedor AL2023 base](#)
- [AL2023 Imagen de contenedor mínima](#)
- [Creación de imágenes de contenedores básicas AL2023](#)
- [Comparación de paquetes instalados en imágenes de contenedores de Amazon Linux 2023](#)
- [Comparación de paquetes instalados en una AMI mínima y en imágenes de contenedores en Amazon Linux 2023](#)

Uso de la imagen del contenedor AL2023 base

La imagen del AL2023 contenedor se crea a partir de los mismos componentes de software que se incluyen en la AL2023 AMI. Está disponible para su uso en cualquier entorno como imagen base para las cargas de trabajo de Docker. Si ya usa la AMI de Amazon Linux para las aplicaciones de [Amazon Elastic Compute Cloud](#) (Amazon EC2), puede incluir sus aplicaciones en contenedores con la imagen de contenedor de Amazon Linux.

Utilice la imagen del contenedor de Amazon Linux en su entorno de desarrollo local y, a continuación, inserte su aplicación para que AWS utilice [Amazon Elastic Container Service](#) (Amazon ECS). Para obtener más información, consulte [Utilización de imágenes de Amazon ECR con Amazon ECS](#) en la Guía del usuario de Amazon Elastic Container Registry.

La imagen de contenedor de Amazon Linux está disponible en Amazon ECR Public. Puede enviarnos sus comentarios AL2023 a través de su AWS representante designado o archivando un problema en el repositorio de [amazon-linux-2023](#) en GitHub.

Para extraer la imagen de contenedor de Amazon Linux desde Amazon ECR Public

1. Autentique el cliente Docker en su registro de Amazon Linux Public. Los tokens de autenticación son válidos durante 12 horas. Para obtener más información, consulte [Autenticación de registros privados](#) en la Guía del usuario de Amazon Elastic Container Registry.

Note

El `get-login-password` comando es compatible con la última versión de la versión 2. AWS CLI. Para obtener más información, consulte [Installing the AWS Command Line Interface](#) en la Guía del usuario de AWS Command Line Interface .

```
$ aws ecr-public get-login-password --region us-east-1 | docker login --username  
AWS --password-stdin public.ecr.aws
```

El resultado es el siguiente.

```
Login succeeded
```

2. Extraiga la imagen de contenedor de Amazon Linux ejecutando el comando `docker pull`. Para ver la imagen de contenedor de Amazon Linux en la galería pública de Amazon ECR, consulte [Galería pública de Amazon ECR: amazonlinux](#).

 Note

Al extraer la imagen del AL2023 Docker contenedor, puede usar las etiquetas en uno de los siguientes formatos:

- Para obtener la versión más reciente de la imagen del AL2023 contenedor, usa la `:2023` etiqueta.
- Para obtener una versión específica de AL2023, puedes usar el siguiente formato:
 - `:2023.[0-7 release quarter].[release date].[build number]`

En los ejemplos siguientes se utiliza la etiqueta `:2023` y se extrae la imagen de contenedor más reciente disponible de AL2023.

```
$ docker pull public.ecr.aws/amazonlinux/amazonlinux:2023
```

3. (Opcional) Ejecute el contenedor localmente.

```
$ docker run -it --security-opt seccomp=unconfined public.ecr.aws/amazonlinux/  
amazonlinux:2023 /bin/bash
```

Para extraer la imagen del AL2023 contenedor de Docker Hub

1. Extraiga la imagen del AL2023 contenedor con el `docker pull` comando.

```
$ docker pull amazonlinux:2023
```

2. (Opcional) Ejecute el contenedor localmente.

```
$ docker run -it amazonlinux:2023 /bin/bash
```

Note

La imagen del contenedor de AL2023 usa solo el administrador de `dnf` paquetes para instalar paquetes de software. Esto significa que no hay ningún comando `amazon-linux-extras` o un comando equivalente para usar con software adicional.

AL2023 Imagen de contenedor mínima

Note

Las imágenes de AL2023 contenedor estándar son adecuadas para la mayoría de los casos de uso, y es probable que la adaptación a la imagen de contenedor mínima requiera más trabajo que la adaptación a la imagen de contenedor AL2023 base.

La imagen del contenedor AL2023 mínimo, presentada en la sección AL2023 .2, difiere de la imagen del contenedor base porque contiene solo los paquetes mínimos necesarios para instalar otros paquetes. La imagen de contenedor mínimo está diseñada para ser un conjunto mínimo de paquetes, no un conjunto conveniente de paquetes.

La imagen AL2023 mínima del contenedor se crea a partir de componentes de software ya disponibles en AL2023. La diferencia clave en la imagen de contenedor mínimo es el uso de `microdnf` para proporcionar el administrador de paquetes `dnf` en lugar del `dnf` completo basado en Python. Esto permite reducir el tamaño de la imagen mínima del contenedor, con la desventaja de no disponer del conjunto completo de funciones del gestor de `dnf` paquetes que se incluye en la imagen del contenedor base AL2023 AMIs y en la imagen del contenedor.

La imagen AL2023 mínima del contenedor constituye la base del entorno de ejecución de `provided.al2023` AWS Lambda.

Para obtener una lista detallada de los paquetes incluidos en la imagen de contenedor mínimo, consulte [Comparación de paquetes instalados en imágenes de contenedores de Amazon Linux 2023](#).

Tamaño de imagen de contenedor mínimo

Como la imagen del contenedor AL2023 mínimo contiene menos paquetes que la imagen del contenedor AL2023 base, también es significativamente más pequeña. En la siguiente tabla se comparan las opciones de imagen de contenedor de las versiones actuales y anteriores de Amazon Linux.

Note

El tamaño de la imagen es el que se muestra en la [Galería pública de Amazon ECR en Amazon Linux](#).

Image	Versión	Tamaño de la imagen	Nota
Amazon Linux (1AL1)	2018.03.0.20230918.0	62,3 MB	Solo x86-64
Amazon Linux 2	2.0.20230926,0	64,2 MB	aarch64 es 1,6 MB mayor que x86-64
Imagen de contenedor base de Amazon Linux 2023	20232,20231002.0	52,4 MB	
Imagen de contenedor mínimo de Amazon Linux 2023	2023.2.20231002.0 - mínimo	35,2 MB	

Uso de la imagen Minimal Container AL2023

La imagen de contenedor AL2023 mínimo está disponible en ECR y la 2023-minimal etiqueta siempre apuntará a la imagen de contenedor mínima AL2023 basada en la última versión, mientras

que la `minimal` etiqueta puede estar actualizada a una versión de Amazon Linux más reciente que AL2023.

Puede extraer estas etiquetas con `docker`. Para ello, siga el ejemplo que se indica a continuación:

```
$ docker pull public.ecr.aws/amazonlinux/amazonlinux:minimal
```

```
$ docker pull public.ecr.aws/amazonlinux/amazonlinux:2023-minimal
```

A continuación se muestra un ejemplo de `Dockerfile` que toma la imagen de contenedor mínima e instala GCC sobre ella:

```
FROM public.ecr.aws/amazonlinux/amazonlinux:2023-minimal
RUN dnf install -y gcc && dnf clean all
```

Creación de imágenes de contenedores básicas AL2023

La imagen de contenedor de AL2023 se construye desde los mismos componentes de software que se incluyen en la AMI de AL2023. Incluye un software que permite que la capa base del contenedor se comporte de manera similar a como lo haría en una instancia de Amazon EC2, como el `dnf` del administrador de paquetes. En esta sección se explica cómo se puede crear un contenedor desde cero que incluya solo las dependencias mínimas necesarias para una aplicación.

Note

Las imágenes de AL2023 contenedores estándar son adecuadas para la mayoría de los casos de uso. El uso de la imagen de contenedor estándar facilita la creación sobre la imagen. Una imagen de contenedor básica dificulta la creación de imágenes basadas en ella.

Creación de un contenedor con las dependencias básicas mínimas para una aplicación

1. Determine las dependencias del tiempo de ejecución. Esto variará en función de la aplicación.
2. Cree un `Dockerfile` / `Containerfile` que se genere `FROM scratch`. El siguiente ejemplo de un `Dockerfile` se puede utilizar para crear un contenedor que contenga solo el intérprete de comandos `bash` y sus dependencias.

```
FROM public.ecr.aws/amazonlinux/amazonlinux:2023 as build
RUN mkdir /sysroot
RUN dnf --releasever=$(rpm -q system-release --qf '%{VERSION}') \
  --installroot /sysroot \
  -y \
  --setopt=install_weak_deps=False \
  install bash

FROM scratch
COPY --from=build /sysroot /
WORKDIR /
ENTRYPOINT ["/bin/bash"]
```

- Este Dockerfile funciona de la siguiente manera:
 1. Iniciar un AL2023 contenedor llamado build. Este contenedor se utilizará para arrancar el contenedor básico. Este contenedor no se implementa por sí mismo, sino que genera el contenedor que se va a implementar.
 2. Creando el directorio /sysroot. En este directorio, el contenedor build instalará las dependencias necesarias para el contenedor básico. En un paso posterior, la ruta /sysroot se empaquetará para que sea el directorio raíz de nuestra imagen básica.

Usando la `--installroot` opción de esta `dnf` manera es como creamos las otras AL2023 imágenes. Es una característica de `dnf` que permite que funcionen los instaladores y las herramientas de creación de imágenes.

3. Invocación a `dnf` para instalar paquetes en /sysroot.

El comando `rpm -q system-release --qf '%{VERSION}'` consulta (`-q`) el paquete `system-release` y establece el formato de consulta (`--qf`) para imprimir la versión del paquete que se está consultando (la variable `%{VERSION}` es la variable `rpm` de la versión de RPM).

Al establecer el argumento `--releasever` de `dnf` en la versión del `system-release` en el contenedor build, este Dockerfile se puede utilizar para reconstruir el contenedor básico siempre que se publique una imagen base de contenedores actualizada de Amazon Linux.

Es posible configurarlo en cualquier versión de `--releasever` Amazon Linux 2023, como la 2023.10.20260325. De este modo, el build contenedor se ejecutaría como la AL2023 versión más reciente, pero se compilaría el contenedor básico a partir de la 2023.10.20260325, independientemente de cuál fuera la versión actual. AL2023

La opción de configuración `--setopt=install_weak_deps=False` indica a `dnf` que solo hay que instalar las dependencias que sean necesarias y no las recomendadas o sugeridas.

4. Copia del sistema instalado en la raíz de un contenedor vacío (FROM scratch).
 5. Definición de ENTRYPOINT como el binario deseado, en este caso, `/bin/bash`.
3. Cree un directorio vacío y agregue el contenido del ejemplo del paso 2 a un archivo denominado `Dockerfile`.

```
$ mkdir al2023-barebones-bash-example
$ cd al2023-barebones-bash-example
$ cat > Dockerfile <<EOF
FROM public.ecr.aws/amazonlinux/amazonlinux:2023 as build
RUN mkdir /sysroot
RUN dnf --releasever=$(rpm -q system-release --qf '%{VERSION}') \
  --installroot /sysroot \
  -y \
  --setopt=install_weak_deps=False \
  install bash && dnf --installroot /sysroot clean all

FROM scratch
COPY --from=build /sysroot /
WORKDIR /
ENTRYPOINT ["/bin/bash"]
EOF
```

4. Cree el contenedor ejecutando el siguiente comando.

```
$ docker build -t al2023-barebones-bash-example
```

5. Ejecute el contenedor con el siguiente comando para ver qué tan mínimo es un contenedor de `bash` exclusivo.

```
$ docker run -it --rm al2023-barebones-bash-example
```

```

bash-5.2# rpm
bash: rpm: command not found
bash-5.2# du -sh /usr/
bash: du: command not found
bash-5.2# ls
bash: ls: command not found
bash-5.2# echo /bin/*
/bin/alias /bin/bash /bin/bashbug /bin/bashbug-64 /bin/bg /bin/catchsegv /bin/cd /
bin/command /bin/fc /bin/fg /bin/gencat /bin/getconf /bin/getent /bin/getopts /
bin/hash /bin/iconv /bin/jobs /bin/ld.so /bin/ldd /bin/locale /bin/localedef /
bin/pldd /bin/read /bin/sh /bin/sotruss /bin/sprof /bin/type /bin/tzselect /bin/
ulimit /bin/umask /bin/unalias /bin/wait /bin/zdump

```

Para obtener un ejemplo más práctico, el siguiente procedimiento crea un contenedor para una aplicación en C que muestra Hello World!.

1. Cree un directorio vacío y agregue el código fuente C y Dockerfile.

```

$ mkdir al2023-barebones-c-hello-world-example
$ cd al2023-barebones-c-hello-world-example
$ cat > hello-world.c <<EOF
#include <stdio.h>
int main(void)
{
    printf("Hello World!\n");
    return 0;
}
EOF

$ cat > Dockerfile <<EOF
FROM public.ecr.aws/amazonlinux/amazonlinux:2023 as build
COPY hello-world.c /
RUN dnf -y install gcc
RUN gcc -o hello-world hello-world.c
RUN mkdir /sysroot
RUN mv hello-world /sysroot/
RUN dnf --releasever=$(rpm -q system-release --qf '%{VERSION}') \
    --installroot /sysroot \
    -y \
    --setopt=install_weak_deps=False \
    install glibc && dnf --installroot /sysroot clean all

```

```
FROM scratch
COPY --from=build /sysroot /
WORKDIR /
ENTRYPOINT ["/hello-world"]
EOF
```

2. Cree el contenedor con el siguiente comando.

```
$ docker build -t al2023-barebones-c-hello-world-example .
```

3. Ejecute el contenedor con el siguiente comando.

```
$ docker run -it --rm al2023-barebones-c-hello-world-example
Hello World!
```

Comparación de paquetes instalados en imágenes de contenedores de Amazon Linux 2023

Una comparación del RPMs presente en la imagen del contenedor AL2023 base con el RPMs presente en la imagen del contenedor AL2023 mínimo.

Package	Container	Contenedor mínimo
alternatives	1.15	1.15
amazon-linux-repo-cdn	2023.6.20241031	2023,6 20241031
audit-libs	3.0.6	3.0.6
basesystem	11	11
bash	5.2.15	5.2.15
bzip2-libs	1.0.8	1.0.8
ca-certificates	2023,2,68	2023,2,68

Package	Container	Contenedor mínimo
coreutils-single	8.32	8.32
crypto-policies	20220428	20220428
curl-minimal	8.5.0	8.5.0
dnf	4.14.0	
dnf-data	4.14.0	4.14.0
elfutils-default-y ama-scope	0.188	
elfutils-libelf	0.188	
elfutils-libs	0.188	
expat	2.5.0	
file-libs	5.39	5.39
filesystem	3.14	3.14
gawk	5.1.0	5.1.0
gdbm-libs	1.19	
glib2	2.74,7	2.74,7
glibc	2.34	2.34
glibc-common	2.34	2.34
glibc-minimal-lang pack	2.34	2.34
gmp	6.2.1	6.2.1
gnupg2-minimal	2.3.7	2.3.7

Package	Container	Contenedor mínimo
gobject-introspection		1,73,0
gpgme	1.15.1	1.15.1
grep	3.8	3.8
json-c	0,14	0.14
keyutils-libs	1.6.3	1.6.3
krb5-libs	1.21.3	1.21.3
libacl	2.3.1	2.3.1
libarchive	3.7.4	3.7.4
libassuan	2.5.5	2.5.5
libattr	2.5.1	2.5.1
libblkid	2.37,4	2.37,4
libcap	2.48	2.48
libcap-ng	0.8.2	0.8.2
libcom_err	1.46,5	1,46.5
libcomps	0.1,20	
libcurl-minimal	8.5.0	8.5.0
libdnf	0.69,0	0.69,0
libffi	3.4.4	3.4.4
libgcc	11.4.1	11.4.1
libgcrypt	1.10.2	1.10.2

Package	Container	Contenedor mínimo
libgomp	11.4.1	
libgpg-error	1.42	1,42
libidn2	2.3.2	2.3.2
libmodulemd	2.13.0	2.13.0
libmount	2.37,4	2.37,4
libnghttp2	1,59,0	1,59,0
libpeas		1.32.0
libpsl	0,21.1	0.21.1
librepo	1.14.5	1.14.5
libreport-filessystem	2.15.2	2.15.2
libselinux	3.4	3.4
libsepol	3.4	3.4
libsigsegv	2.13	2.13
libsmartcols	2.37,4	2.37,4
libsolv	0.7.22	0.7.22
libstdc++	11.4.1	11.4.1
libtasn1	4.19,0	4.19,0
libunistring	0,9,10	0,9,10
libuuid	2.37,4	2.37,4
libverto	0.3.2	0.3.2

Package	Container	Contenedor mínimo
libxcrypt	4.4.33	
libxml2	2.10.4	2.10.4
libyaml	0.2.5	0.2.5
libzstd	1.5.5	1.5.5
lua-libs	5.4.4	5.4.4
lz4-libs	1.9.4	1.9.4
microdnf		3.10.0
microdnf-dnf		3.10.0
mpfr	4.1.0	4.1.0
ncurses-base	6.2	6.2
ncurses-libs	6.2	6.2
npth	1.6	1.6
openssl-libs	3.0.8	3.0.8
p11-kit	0.24.1	0,24.1
p11-kit-trust	0,24.1	0,24.1
pcre2	10,40	10,40
pcre2-syntax	10,40	10,40
popt	1,18	1.18
publicsuffix-list-dafsa	20240212	20240212
python3	3,9,16	

Package	Container	Contenedor mínimo
python3-dnf	4.14.0	
python3-gpg	1.15.1	
python3-hawkey	0.69,0	
python3-libcomps	0.1,20	
python3-libdnf	0.69,0	
python3-libs	3,9,16	
python3-pip-wheel	21.3.1	
python3-rpm	4.16.1.3	
python3-setuptools-wheel	59,6,0	
readline	8.1	8.1
rpm	4.16.1.3	4.16.1.3
rpm-build-libs	4.16.1.3	
rpm-libs	4.16.1.3	4.16.1.3
rpm-sign-libs	4.16.1.3	
sed	4.8	4.8
setup	2.13.7	2.13.7
sqlite-libs	3.40,0	3.40,0
system-release	2023,6 20241031	2023,6 20241031
tzdata	2024a	
xz-libs	5.2.5	5.2.5

Package	Container	Contenedor mínimo
yum	4.14.0	
zlib	1.2.11	1.2.11

Comparación de paquetes instalados en una AMI mínima y en imágenes de contenedores en Amazon Linux 2023

Una comparación del RPMs presente en la AMI AL2023 mínima con las imágenes RPMs actuales de la AL2023 base y del contenedor mínimo.

Package	AMI mínima	Container	Contenedor mínimo
alternatives	1.15	1.15	1.15
amazon-chrony-config	4.3		
amazon-ec2-net-utils	2.5.1		
amazon-linux-repo-cdn		2023.6.20241031	2023,6 20241031
amazon-linux-repo-s3	2023,6 20241031		
amazon-linux-sb-keys	2023,1		
amd-ucode-firmware	20210208 (noarch)		
audit	3.0.6		
audit-libs	3.0.6	3.0.6	3.0.6

Package	AMI mínima	Container	Contenedor mínimo
awscli-2	2.15.30		
basesystem	11	11	11
bash	5.2.15	5.2.15	5.2.15
bzip2-libs	1.0.8	1.0.8	1.0.8
ca-certificates	2023,2,68	2023,2,68	2023,2,68
checkpolicy	3.4		
chrony	4.3		
cloud-init	22.2.2		
cloud-init-cfg-ec2	22.2.2		
cloud-utils-growpart	0.31		
coreutils	8.32		
coreutils-common	8.32		
coreutils-single		8.32	8.32
cpio	2.13		
cracklib	2.9.6		
cracklib-dicts	2.9.6		
crypto-policies	20220428	20220428	20220428
cryptsetup-libs	2.6.1		

Package	AMI mínima	Container	Contenedor mínimo
<u>curl-minimal</u>	8.5.0	8.5.0	8.5.0
cyrus-sasl-lib	2.1.27		
dbus	1.12.28		
dbus-broker	32		
dbus-common	1.12.28		
dbus-libs	1.12.28		
device-mapper	1.02.185		
device-mapper-libs	1.02.185		
diffutils	3.8		
dnf	4.14.0	4.14.0	
dnf-data	4.14.0	4.14.0	4.14.0
dnf-plugin-release-notification	1.2		
dnf-plugins-core	4.3.0		
dnf-plugin-support-info	1.2		
dracut	055		
dracut-config-ec2	3.0		

Package	AMI mínima	Container	Contenedor mínimo
dracut-config-generic	055		
e2fsprogs	1.46,5		
e2fsprogs-libs	1.46,5		
ec2-utils	2.2.0		
efi-filesystem	5		
efivar	38		
efivar-libs	38		
elfutils-default-yama-scope	0.188	0.188	
elfutils-libelf	0.188	0.188	
elfutils-libs	0.188	0.188	
expat	2.5.0	2.5.0	
file	5.39		
file-libs	5.39	5.39	5.39
filesystem	3.14	3.14	3.14
findutils	4.8.0		
fuse-libs	2.9.9		
gawk	5.1.0	5.1.0	5.1.0
gdbm-libs	1.19	1.19	

Package	AMI mínima	Container	Contenedor mínimo
gdisk	1.0.8		
gettext	0.21		
gettext-libs	0,21		
glib2	2.74,7	2.74,7	2.74,7
glibc	2.34	2.34	2.34
glibc-all-langpacks	2.34		
glibc-common	2.34	2.34	2.34
glibc-locale-source	2.34		
glibc-minimal-langpack		2.34	2.34
gmp	6.2.1	6.2.1	6.2.1
gnupg2-minimal	2.3.7	2.3.7	2.3.7
gnutls	3.8.0		
gobject-introspection			1,73,0
gpgme	1.15.1	1.15.1	1.15.1
grep	3.8	3.8	3.8
groff-base	1,22.4		
grub2-common	2.06		

Package	AMI mínima	Container	Contenedor mínimo
grub2-efi-aa64-ec2	2.06 (aarch64)		
grub2-efi-x64-ec2	2.06 (x86_64)		
grub2-pc-modules	2.06		
grub2-tools	2.06		
grub2-tools-minimal	2.06		
grubby	8.40		
gzip	1.12		
hostname	3.23		
hwdata	0,384		
inih	49		
initscripts	10,09		
iproute	6.10.0		
iputils	20210202		
irqbalance	1.9.0		
jansson	2.14		
jitterentropy	3.4.1		
jq	1.7.1		
json-c	0,14	0.14	0.14

Package	AMI mínima	Container	Contenedor mínimo
kbd	2.4.0		
kbd-misc	2.4.0		
kernel	6.1.112		
kernel-libbpf	6.1.112		
kernel-li vepatch-repo- s3	2023,620241031		
keyutils-libs	1.6.3	1.6.3	1.6.3
kmod	29		
kmod-libs	29		
krb5-libs	1,21.3	1.21.3	1.21.3
less	608		
libacl	2.3.1	2.3.1	2.3.1
libarchive	3.7.4	3.7.4	3.7.4
libargon2	20171227		
libassuan	2.5.5	2.5.5	2.5.5
libattr	2.5.1	2.5.1	2.5.1
libblkid	2.37,4	2.37,4	2.37,4
libcap	2.48	2.48	2.48
libcap-ng	0.8.2	0.8.2	0.8.2
libcbor	0.7.0		

Package	AMI mínima	Container	Contenedor mínimo
libcom_err	1.46,5	1.46,5	1.46,5
libcomps	0.1.20	0.1,20	
libcurl-minimal	8.5.0	8.5.0	8.5.0
libdb	5.3.28		
libdnf	0.69,0	0.69,0	0.69,0
libeconf	0.4.0		
libedit	3.1		
libfdisk	2.37,4		
libffi	3.4.4	3.4.4	3.4.4
libfido2	1.10.0		
libgcc	11.4.1	11.4.1	11.4.1
libgcrypt	1.10.2	1.10.2	1.10.2
libgomp	11.4.1	11.4.1	
libgpg-error	1.42	1,42	1,42
libidn2	2.3.2	2.3.2	2.3.2
libkcapi	1.4.0		
libkcapi-hmaccalc	1.4.0		
libmnl	1.0.4		
libmodulemd	2.13.0	2.13.0	2.13.0
libmount	2.37,4	2.37,4	2.37,4

Package	AMI mínima	Container	Contenedor mínimo
libnghttp2	1,59,0	1,59,0	1,59,0
libpeas			1.32.0
libpipeline	1.5.3		
libpsl	0,21.1	0.21.1	0.21.1
libpwquality	1.4.4		
librepo	1.14.5	1.14.5	1.14.5
libreport-filesystem	2.15.2	2.15.2	2.15.2
libseccomp	2.5.3		
libselinux	3.4	3.4	3.4
libselinux-utils	3.4		
libsemanage	3.4		
libsepol	3.4	3.4	3.4
libsigsegv	2.13	2.13	2.13
libsmartcols	2.37,4	2.37,4	2.37,4
libsolv	0.7.22	0.7.22	0.7.22
libss	1.46,5		
libstdc++	11.4.1	11.4.1	11.4.1
libtasn1	4.19,0	4.19,0	4.19,0
libtextstyle	0,21		

Package	AMI mínima	Container	Contenedor mínimo
libunistring	0,9,10	0,9,10	0,9,10
libuser	0,63		
libutempter	1.2.1		
libuuid	2.37,4	2.37,4	2.37,4
libverto	0.3.2	0.3.2	0.3.2
libxcrypt	4.4.33	4.4.33	
libxml2	2.10.4	2.10.4	2.10.4
libyaml	0.2.5	0.2.5	0.2.5
libzstd	1.5.5	1.5.5	1.5.5
linux-firmware- whence	20210208 (noarch)		
logrotate	3.20.1		
lua-libs	5.4.4	5.4.4	5.4.4
lz4-libs	1.9.4	1.9.4	1.9.4
man-db	2.9.3		
microcode_ctl	2.1 (x86_64)		
microdnf			3.10.0
microdnf-dnf			3.10.0
mpfr	4.1.0	4.1.0	4.1.0
ncurses	6.2		
ncurses-base	6.2	6.2	6.2

Package	AMI mínima	Container	Contenedor mínimo
ncurses-libs	6.2	6.2	6.2
nettle	3.8		
net-tools	2.0		
npth	1.6	1.6	1.6
numactl-libs	2.0.14		
oniguruma	6.9.7.1		
openldap	2.4.57		
openssh	8.7p1		
openssh-clients	8.7p1		
openssh-server	8.7p1		
openssl	3.0.8		
openssl-libs	3.0.8	3.0.8	3.0.8
openssl-pkcs11	0.4.12		
os-prober	1.77		
p11-kit	0.24.1	0,24.1	0,24.1
p11-kit-trust	0,24.1	0,24.1	0,24.1
pam	1.5.1		
passwd	0,80		
pciutils	3.7.0		
pciutils-libs	3.7.0		

Package	AMI mínima	Container	Contenedor mínimo
pcre2	10,40	10,40	10,40
pcre2-syntax	10,40	10,40	10,40
policycoreutils	3.4		
popt	1,18	1.18	1.18
procps-ng	3.3.17		
psmisc	23.4		
publicsuffix-list-dafsa	20240212	20240212	20240212
python3	3,9,16	3,9,16	
python3-attrs	203,0		
python3-audit	3.0.6		
python3-awscrt	0,19,19		
python3-babel	2.9.1		
python3-cffi	1.14.5		
python3-chardet	4.0.0		
python3-colorama	0.4.4		
python3-configobj	5.0.6		
python3-cryptography	36,01		

Package	AMI mínima	Container	Contenedor mínimo
python3-d ateutil	2.8.1		
python3-dbus	1.2.18		
python3-distro	1.5.0		
python3-dnf	4.14.0	4.14.0	
python3-dnf- plugins-core	4.3.0		
python3-d ocutils	0,16		
python3-gpg	1.15.1	1.15.1	
python3-hawkey	0.69,0	0.69,0	
python3-idna	(2.10)		
python3-jinja2	2.11.3		
python3-j mespath	0.10.0		
python3-j sonpatch	1.21		
python3-j sonpointer	2.0		
python3-j sonschema	3.2.0		
python3-l ibcomps	0.1.20	0.1,20	

Package	AMI mínima	Container	Contenedor mínimo
python3-libdnf	0.69,0	0.69,0	
python3-libs	3,9,16	3,9,16	
python3-l ibselinux	3.4		
python3-l ibsemanage	3.4		
python3-m arkupsafe	1.1.1		
python3-n etifaces	0.10.6		
python3-o authlib	3.0.2		
python3-pip- wheel	21.3.1	21.3.1	
python3-ply	3.11		
python3-p olicycoreutils	3.4		
python3-p rettytable	0.7.2		
python3-prompt- toolkit	3.0.24		
python3-p ycparser	2.20		
python3-p yrsistent	0,17.3		

Package	AMI mínima	Container	Contenedor mínimo
python3-pyserial	3.4		
python3-pysocks	1.7.1		
python3-pytz	2022.7.1		
python3-pyyaml	5.4.1		
python3-requests	2.25.1		
python3-rpm	4.16.1.3	4.16.1.3	
python3-ruamel-yaml	0,16,6		
python3-ruamel-yaml-clib	0.1.2		
python3-setuptools	4.4.1		
python3-setuptools	59,6,0		
python3-setuptools-wheel	59,6,0	59,6,0	
python3-six	1.15.0		
python3-systemd	235		
python3-urllib3	1,25,10		
python3-wcwidth	0.2.5		
readline	8.1	8.1	8.1

Package	AMI mínima	Container	Contenedor mínimo
rng-tools	6.14		
rootfiles	8.1		
rpm	4.16.1.3	4.16.1.3	4.16.1.3
rpm-build-libs	4.16.1.3	4.16.1.3	
rpm-libs	4.16.1.3	4.16.1.3	4.16.1.3
rpm-plugin-selinux	4.16.1.3		
rpm-plugin-systemd-inhibit	4.16.1.3		
rpm-sign-libs	4.16.1.3	4.16.1.3	
sbsigntools	0.9.4		
sed	4.8	4.8	4.8
selinux-policy	381,45		
selinux-policy-targeted	381,45		
setup	2.13.7	2.13.7	2.13.7
shadow-utils	4.9		
sqlite-libs	3.40,0	3.40,0	3.40,0
sudo	1.9.15		
sysctl-defaults	1.0		
systemd	252,23		

Package	AMI mínima	Container	Contenedor mínimo
systemd-libs	252,23		
systemd-networkd	252,23		
systemd-pam	252,23		
systemd-resolved	252,23		
systemd-udev	252,23		
system-release	2023,6 20241031	2023,6 20241031	2023,6 20241031
tar	1,34		
tzdata	2024a	2024a	
update-motd	2.2		
userspace-rcu	0.12.1		
util-linux	2.37,4		
util-linux-core	2.37,4		
vim-data	9,0,2153		
vim-minimal	9,0,2153		
which	2.21		
xfsprogs	5.18.0		
xz	5.2.5		
xz-libs	5.2.5	5.2.5	5.2.5
yum	4.14.0	4.14.0	

Package	AMI mínima	Container	Contenedor mínimo
zlib	1.2.11	1.2.11	1.2.11
zram-generator	1.1.2		
zram-generator-defaults	1.1.2		
zstd	1.5.5		

AL2023 en AWS Elastic Beanstalk

AWS Elastic Beanstalk es un servicio para implementar y escalar aplicaciones y servicios web. Puede cargar simplemente su código y Elastic Beanstalk se encarga automáticamente de la implementación, desde el aprovisionamiento de capacidad y el equilibrio de carga hasta el escalado automático y la supervisión del estado de las aplicaciones. Para obtener más información, consulte [AWS Elastic Beanstalk](#).

Para utilizar Elastic Beanstalk, debe crear una aplicación, cargar una versión de la aplicación como un paquete de código fuente (por ejemplo, un archivo Java.war) en Elastic Beanstalk y proporcionar cierta información sobre la aplicación. Elastic Beanstalk lanza automáticamente un entorno y crea y configura los recursos de AWS necesarios para ejecutar el código. Para obtener más información, consulte la [Guía para desarrolladores de AWS Elastic Beanstalk](#).

Las plataformas Linux de Elastic Beanstalk utilizan instancias de Amazon EC2 y estas instancias ejecutan Amazon Linux. A partir del 4 de agosto de 2023, Elastic Beanstalk ofrece las siguientes ramificaciones de la plataforma basadas en Amazon Linux 2023: Docker, Tomcat, Java SE, Node.js, PHP, y Python. Elastic Beanstalk está trabajando para ofrecer soporte para AL2023 en más plataformas de Elastic Beanstalk.

La lista completa de compatibilidad de la plataforma Elastic Beanstalk y de las plataformas actuales creadas a partir de AL2023 se encuentra en la sección [Plataformas Linux de Elastic Beanstalk](#) de la [Guía para desarrolladores de Elastic Beanstalk](#).

Puede encontrar las notas de la versión de las nuevas plataformas Elastic Beanstalk y las versiones de las plataformas existentes en [Notas de la versión de Elastic Beanstalk](#).

Uso de AL2023 en AWS CloudShell

AWS CloudShell es un intérprete de comandos previamente autenticado y basado en navegador que se puede lanzar directamente desde la Consola de administración de AWS. Puede navegar a CloudShell de varias maneras diferentes desde la Consola de administración de AWS. Para obtener más información, consulte [¿Cómo empezar a usar AWS CloudShell?](#)

AWS CloudShell, que actualmente se basa en Amazon Linux 2, migrará a AL2023. La migración a AL2023 empezará a implementarse en todas las Regiones de AWS partir del 4 de diciembre de 2023. Para obtener más información sobre la migración de CloudShell a AL2023, consulte [Migración de AWS CloudShell desde Amazon Linux 2 a Amazon Linux 2023](#).

Uso de AMI de Amazon ECS basadas en AL2023 para alojar cargas de trabajo en contenedores

Note

Para obtener más información sobre cómo usar AL2023 dentro de un contenedor, consulte [AL2023 en contenedores](#).

Amazon Elastic Container Service (Amazon ECS) es un servicio de orquestación de contenedores completamente administrado que facilita la implementación, la administración y el escalado de aplicaciones en contenedores. Como servicio totalmente administrado, Amazon ECS incluye configuración y prácticas recomendadas operativas integradas de AWS. Se integra con AWS y con herramientas de terceros, como Amazon Elastic Container Registry (Amazon ECR) y Docker. Esta integración facilita a los equipos centrarse en crear las aplicaciones, no en el entorno. Puede ejecutar y escalar las cargas de trabajo de contenedores en todas las regiones de AWS en la nube, sin la complejidad de administrar un plano de control.

Puede alojar fácilmente cargas de trabajo en contenedores en AL2023 mediante la AMI optimizada de Amazon ECS basada en AL2023. Para obtener más información, consulte la [AMI optimizada para Amazon ECS](#).

Cambios en AL2023 para Amazon ECS en comparación con AL2

Al igual que AL2, AL2023 proporciona los paquetes base necesarios para ejecutarse como una instancia Linux de Amazon ECS. En AL2, los paquetes `containerd`, `docker` y `ecs-init` ya

estaban disponibles a través de `amazon-linux-extras`, mientras que en AL2023 estos paquetes se incluyen en los repositorios principales.

Con la característica determinista de actualizaciones a través de repositorios versionados, cada AMI de AL2023 está bloqueada de forma predeterminada en una versión de repositorio específica. Esto también es válido para la AMI optimizada de AL2023 de Amazon ECS. Todas las actualizaciones de su entorno se pueden administrar y probar cuidadosamente antes de la implementación, además de proporcionar una forma fácil de volver al contenido de una AMI anterior en caso de que surja un problema. Para obtener más información acerca de esta característica de AL2023, consulte [Actualizaciones deterministas mediante repositorios versionados en AL2023](#).

AL2023 cambia a `cgroup v2` a través de la interfaz `cgroup v1` compatible con AL2. Para obtener más información, consulte [Jerarquía de grupos de control unificados \(cgroup v2\)](#).

Note

Las versiones de AL2023 anteriores a la [2023.2.20230920](#) (la primera versión de AL2023.2) contenían un error en `systemd` relacionado con la gestión de la falta de memoria (OOM) dentro de un `cgroup`. Siempre se eliminaban todos los procesos del `cgroup`, en lugar de que el asesino de OOM eligiera un proceso a la vez, que era el comportamiento previsto. Se trataba de una regresión en comparación con el comportamiento de AL2 y se corrigió a partir de la versión 2023.2.20230920 de AL2023.

El código para crear la AMI optimizada para Amazon ECS está disponible en el [proyecto amazon-ecs-ami de GitHub](#). Las [notas de la versión](#) describen qué versión de AL2023 se asigna a qué versión de la AMI de Amazon ECS.

Personalización de la AMI optimizada para Amazon ECS y basada en AL2023

Important

Le recomendamos que utilice la AMI de AL2023 optimizada para Amazon ECS. Para obtener más información, consulte la [AMI optimizada para Amazon ECS](#) en la Guía para desarrolladores de Amazon Elastic Container Service.

Puede utilizar los mismos scripts de compilación que utiliza Amazon ECS para crear las AMI personalizadas. Para obtener más información, consulte [Script de compilación de AMI de Linux optimizadas para Amazon ECS](#).

Uso de Amazon Elastic File System en AL2023

Amazon Elastic File System (Amazon EFS) proporciona un almacenamiento de archivos totalmente elástico y sin servidor para que pueda compartir datos de archivos sin aprovisionar ni administrar la capacidad de almacenamiento ni el rendimiento. Amazon EFS está diseñado para ampliarse a petabytes según la demanda sin interrumpir las aplicaciones, es decir que aumenta y disminuye automáticamente a medida que se agregan o eliminan archivos. Amazon EFS dispone de una sencilla interfaz de servicios web que le permite crear y configurar sistemas de archivos rápida y fácilmente. El servicio se encarga de administrar toda la infraestructura de almacenamiento de archivos, por lo que puede evitar la complejidad de implementación, aplicación de parches y mantenimiento de configuraciones complejas de sistemas de archivos.

Amazon EFS es compatible con la versión 4 (NFSv4.1 y NFSv4.0) del protocolo Network File System, para que las aplicaciones y herramientas que utiliza actualmente funcionen sin problemas con Amazon EFS. Varias instancias informáticas, incluidas Amazon EC2, Amazon ECS y AWS Lambda, pueden acceder a un sistema de archivos de Amazon EFS al mismo tiempo. Por tanto, un sistema de archivos de EFS puede proporcionar un origen de datos común para cargas de trabajo y aplicaciones que se ejecutan en más de una instancia o servidor.

Instalación de **amazon-efs-utils** en AL2023

El paquete **amazon-efs-utils** está disponible en los repositorios de AL2023 para que se pueda instalar y utilizar para acceder a los sistemas de archivos de Amazon EFS.

Instalación del paquete **amazon-efs-utils** en AL2023

- Instale **amazon-efs-utils** con el siguiente comando:

```
$ dnf -y install amazon-efs-utils
```

Montaje de un sistema de archivos de Amazon EFS en AL2023

Una vez instalado `amazon-efs-utils`, puede montar un sistema de archivos de Amazon EFS en la instancia de AL2023.

Montaje de un sistema de archivos de Amazon EFS en AL2023

- Para hacer el montaje con el ID del sistema de archivos, utilice el siguiente comando.

```
sudo mount -t efs file-system-id efs-mount-point/
```

También puede montar el sistema de archivos de forma que los datos en tránsito se cifren mediante TLS, o bien con el nombre DNS o la IP de destino del montaje en lugar del identificador del sistema de archivos. Para obtener más información, consulte [Montaje en instancias de Linux de Amazon mediante el asistente de montaje EFS](#).

Uso de Amazon EMR integrado AL2023

Amazon EMR es un servicio web que facilita el procesamiento rápido de enormes volúmenes de datos mediante Apache Hadoop y los servicios que ofrece AWS.

AL2023 versiones basadas en Amazon EMR

La versión 7.0.0 de Amazon EMR fue la primera versión basada en AL2023. Con esta versión, AL2023 es el sistema operativo base de Amazon EMR, que aporta todas las ventajas de AL2023 Amazon EMR. Para obtener más información, consulte las [notas de la versión de Amazon Linux 7.0.0](#).

AL2023 basado en Amazon EMR en EKS

Amazon EMR en EKS 6.13 fue la primera versión que se presentó AL2023 como opción. Con esta versión, puede lanzar Spark AL2023 como sistema operativo, junto con el motor de ejecución Java 17. Para obtener más información, consulte las [notas de la versión de Amazon EMR en EKS 6.13](#) y todas las [notas de la versión de Amazon EMR en EKS](#).

Uso AL2023 en AWS Lambda

Con AWS Lambda, puede ejecutar código sin aprovisionar ni administrar servidores. Solo paga el tiempo de computación que consume, sin ningún cargo mientras su código no se ejecuta. Puede

ejecutar código para prácticamente cualquier tipo de aplicación o servicio back-end, sin necesidad de administración. Solo tiene que cargar su código y Lambda se ocupará de todo lo necesario para ejecutarlo y escalarlo con alta disponibilidad.

AL2023 **provided.al2023** tiempo de ejecución gestionado e imagen de contenedor

El tiempo de ejecución `provided.al2023` base se basa en la [imagen AL2023 mínima del contenedor](#) y proporciona un tiempo de ejecución gestionado por Lambda y una imagen [base del contenedor AL2023](#) basados en Lambda. Como el `provided.al2023` tiempo de ejecución se basa en la imagen AL2023 mínima del contenedor, es considerablemente más pequeño, con menos de 40 MB, que el `provided.al2` tiempo de ejecución, con unos 109 MB.

Para obtener más información, consulte [Tiempos de ejecución de Lambda](#) y [Uso de imágenes de contenedor de Lambda](#).

AL2023 tiempos de ejecución basados en Lambda

Las versiones futuras de los tiempos de ejecución en lenguajes gestionados, como la Node.js 20, la Python 3.12, la Java 21 y la .NET 8, se basan en la imagen base AL2023 y se utilizarán como imagen base, `provided.al2023` tal como se describe en el [anuncio de los tiempos de AL2023 ejecución basados](#).

AL2023 funciones Lambda basadas en

- [AL2023 Funciones Lambda escritas en Go](#)
- [AL2023 Funciones Lambda escritas en Rust](#)

Para obtener más información, consulta [Tiempos de ejecución de Lambda](#) en la Guía para desarrolladores de AWS Lambda .

Tutoriales

Los siguientes tutoriales muestran cómo realizar tareas comunes con instancias de Amazon EC2 que ejecutan Amazon Linux 2023 (AL2023). Para obtener tutoriales en video, consulte [Vídeos y laboratorios instructivos de AWS](#).

Para AL2 obtener instrucciones, consulte [los tutoriales sobre instancias de Amazon EC2 que ejecutan Linux](#) en la Guía del usuario de Amazon EC2.

Tutoriales

- [Tutorial: Instalar un servidor LAMP en AL2023](#)
- [Tutorial: Configurar SSL/TLS en AL2023](#)
- [Tutorial: alojar un WordPress blog en AL2023](#)
- [Tutorial: Transición de Redis 6 a Valkey en 2023 AL2](#)
- [Tutorial: Instalar el entorno de escritorio GNOME en AL2023](#)
- [Tutorial: Configurar el servidor TigerVNC en AL2023](#)
- [Uso de LRU multigeneración \(MGLRU\) en los núcleos AL2023](#)
- [Tutorial: Instalar LibreOffice en AL2023](#)

Tutorial: Instalar un servidor LAMP en AL2023

Los siguientes procedimientos le ayudan a instalar un servidor web Apache compatible con PHP y [MariaDB](#) (una bifurcación de MySQL desarrollada por la comunidad) AL2023 en su instancia (a veces denominada servidor web LAMP o pila LAMP). Puede utilizar este servidor para alojar un sitio web estático o implementar una aplicación PHP dinámica que lea y escriba información en una base de datos.

Important

Estos procedimientos están diseñados para usarse con AL2023. Si está intentando configurar un servidor web LAMP en una distribución diferente, como Ubuntu o Red Hat Enterprise Linux, este tutorial no funcionará. Para Ubuntu, consulte la siguiente documentación de la comunidad de Ubuntu: [ApacheMySQLPHP](#). Para otras distribuciones, consulte su documentación específica.

Tareas

- [Paso 1: Preparar el servidor LAMP](#)
- [Paso 2: Probar el servidor LAMP](#)
- [Paso 3: Proteger el servidor de base de datos](#)
- [Paso 4: Instalación \(opcional\) phpMyAdmin](#)
- [Solución de problemas](#)
- [Temas relacionados](#)

Paso 1: Preparar el servidor LAMP

Requisitos previos

- En este tutorial se da por sentado que ya ha lanzado una nueva instancia con AL2023 un nombre de DNS público al que se pueda acceder desde Internet. Para obtener más información, consulte [AL2023 en Amazon EC2](#). También debe haber configurado el grupo de seguridad para que permita las conexiones SSH (puerto 22), HTTP (puerto 80) y HTTPS (puerto 443). Para obtener más información sobre estos requisitos previos, consulte [Autorización del tráfico entrante para sus instancias de Linux](#) en la Guía del usuario de Amazon EC2.
- El siguiente procedimiento instala la última versión de PHP disponible en la versión 8.1 AL2023, actualmente. Si tiene previsto utilizar otras aplicaciones de PHP diferentes a las que se indican en este tutorial, debe verificar su compatibilidad con 8.1.

Para preparar el servidor LAMP

1. Conéctese a la instancia. Para obtener más información, consulte [Conexión a instancias AL2023](#).
2. Para asegurarse de que todos los paquetes de software están actualizados, realice una actualización rápida del software en la instancia. Este proceso puede durar unos minutos, pero es importante realizarlo para asegurarse de que tiene las actualizaciones de seguridad y las correcciones de errores más recientes.

La opción `-y` instala las actualizaciones sin necesidad de confirmación. Si le gustaría examinar las actualizaciones antes de la instalación, puede omitir esta opción.

```
[ec2-user ~]$ sudo dnf upgrade -y
```

3. Instale las versiones más recientes del servidor web Apache y de los paquetes PHP para AL2023.

```
[ec2-user ~]$ sudo dnf install -y httpd wget php-fpm php-mysql cli php-json php php-devel
```

4. Instale los paquetes de software de MariaDB. Utilice el comando `dnf install` para instalar varios paquetes de software y todas las dependencias relacionadas al mismo tiempo.

```
[ec2-user ~]$ sudo dnf install mariadb105-server
```

Puede ver las versiones actuales de estos paquetes mediante el comando siguiente:

```
[ec2-user ~]$ sudo dnf info package_name
```

Ejemplo:

```
[root@ip-172-31-25-170 ec2-user]# dnf info mariadb105
Last metadata expiration check: 0:00:16 ago on Tue Feb 14 21:35:13 2023.
Installed Packages
Name           : mariadb105
Epoch         : 3
Version        : 10.5.16
Release        : 1.amzn2023.0.6
Architecture   : x86_64
Size           : 18 M
Source         : mariadb105-10.5.16-1.amzn2023.0.6.src.rpm
Repository     : @System
From repo      : amazonlinux
Summary        : A very fast and robust SQL database server
URL            : http://mariadb.org
License        : GPLv2 and LGPLv2
Description    : MariaDB is a community developed fork from MySQL - a multi-user,
                  multi-threaded
                  : SQL database server. It is a client/server implementation consisting
of
                  : a server daemon (mariabdb) and many different client programs and
libraries.
                  : The base package contains the standard MariaDB/MySQL client programs
and
                  : utilities.
```

5. Inicie el servidor web Apache.

```
[ec2-user ~]$ sudo systemctl start httpd
```

6. Utilice el comando systemctl para configurar el servidor web Apache de forma que se inicie cada vez que arranque el sistema.

```
[ec2-user ~]$ sudo systemctl enable httpd
```

Puede verificar que httpd está activo ejecutando el siguiente comando:

```
[ec2-user ~]$ sudo systemctl is-enabled httpd
```

7. Si aún no lo ha hecho, añada una regla de seguridad para permitir las conexiones HTTP entrantes (puerto 80) con la instancia. De forma predeterminada, se creó un grupo de seguridad launch-wizard para la instancia durante el lanzamiento. Si no agregó reglas de grupo de seguridad adicionales, este grupo contiene solo una regla para permitir las conexiones SSH.

- Abra la consola de Amazon EC2 en <https://console.aws.amazon.com/ec2/>.
- En el panel de navegación izquierdo, elija Instances (Instancias) y seleccione la instancia.
- En la pestaña Security (Seguridad), consulte las reglas de entrada. Debería ver la siguiente regla:

Port range	Protocol	Source
22	tcp	0.0.0.0/0

Warning

El uso 0.0.0.0/0 permite que todas IPv4 las direcciones accedan a la instancia mediante SSH. Esto es aceptable para un periodo de tiempo corto en un entorno de prueba, pero no es seguro en entornos de producción. En entornos de producción, solo se autoriza el acceso a la instancia a una dirección IP o a un rango de direcciones IP específicas.

- Si no hay una regla de entrada para permitir las conexiones HTTP (puerto 80), debe agregar la regla ahora. Elija el vínculo para el grupo de seguridad. Siga los procedimientos que se indican en [Autorización del tráfico entrante para sus instancias de Linux](#) para añadir una nueva regla de seguridad de entrada con los valores siguientes:

- Tipo: HTTP
- Protocolo: TCP
- Rango de puertos: 80
- Origen: personalizado

8. Pruebe el servidor web. En un navegador web, escriba la dirección DNS pública (o la dirección IP pública) de la instancia. Si no hay contenido en `/var/www/html`, debería aparecer la página de prueba de Apache, que mostrará el mensaje “It works!” (“¡Funciona!”).

Puede obtener el DNS público de su instancia mediante la consola Amazon EC2 (consulte la columna IPv4 DNS público; si esta columna está oculta, elija Preferencias (el icono con forma de engranaje) y active el DNS público). IPv4

Compruebe que el grupo de seguridad de la instancia contenga una regla para permitir el tráfico HTTP en el puerto 80. Para obtener más información, consulte [Agregar reglas al grupo de seguridad](#).

 Important

Si no utiliza Amazon Linux, es posible que también tenga que configurar el firewall en su instancia para permitir estas conexiones. Para obtener más información acerca de cómo configurar el firewall, consulte la documentación de su distribución específica.

Apache httpd sirve archivos que se guardan en un directorio denominado raíz de documentos de Apache. La raíz de documentos de Apache de Amazon Linux es `/var/www/html`, que es propiedad del directorio raíz de manera predeterminada.

Para permitir que la cuenta `ec2-user` manipule archivos de este directorio, debe modificar la propiedad y los permisos del directorio. Existen muchas formas de realizar esta tarea. En este tutorial se agrega el usuario `ec2-user` al grupo `apache`, se otorga al grupo `apache` la propiedad del directorio `/var/www` y se asignan permisos de escritura al grupo.

Para establecer permisos de archivo

1. Añada el usuario (en este caso, el usuario `ec2-user`) al grupo `apache`.

```
[ec2-user ~]$ sudo usermod -a -G apache ec2-user
```

2. Cierre sesión, luego vuelva a iniciarla para elegir el nuevo grupo y, a continuación, verifique si pertenece a este.

- a. Cierre sesión (utilice el comando `exit` o cierre la ventana del terminal):

```
[ec2-user ~]$ exit
```

- b. Para verificar si pertenece al grupo `apache`, vuelva a conectarse a la instancia y, entonces, ejecute el siguiente comando:

```
[ec2-user ~]$ groups  
ec2-user adm wheel apache systemd-journal
```

3. Cambie la propiedad de grupo de `/var/www` y su contenido al grupo `apache`.

```
[ec2-user ~]$ sudo chown -R ec2-user:apache /var/www
```

4. Para agregar permisos de escritura de grupo y establecer el ID de grupo en futuros subdirectorios, cambie los permisos del directorio `/var/www` y sus subdirectorios.

```
[ec2-user ~]$ sudo chmod 2775 /var/www && find /var/www -type d -exec sudo chmod 2775 {} \;
```

5. Para agregar permisos de escritura de grupo, cambie recursivamente los permisos de archivo de `/var/www` y sus subdirectorios:

```
[ec2-user ~]$ find /var/www -type f -exec sudo chmod 0664 {} \;
```

Ahora el usuario `ec2-user` (y cualquier futuro miembro del grupo `apache`) puede añadir, eliminar y editar archivos en la raíz de documentos de Apache, por lo que podrá añadir contenido, como un sitio web estático o una aplicación PHP.

Para proteger el servidor web (opcional)

Un servidor web que ejecuta el protocolo HTTP no proporciona seguridad de transporte de los datos que envía o recibe. Cuando se conecta a un servidor HTTP mediante un navegador web, lo URLs que visita, el contenido de las páginas web que recibe y el contenido (incluidas las contraseñas) de cualquier formulario HTML que envíe son visibles para los intrusos en cualquier parte de la ruta de

red. La práctica recomendada para proteger el servidor web es instalar soporte para HTTPS (HTTP seguro), que protege los datos con cifrado SSL/TLS.

Para obtener información sobre la habilitación de HTTPS en su servidor, consulte [Tutorial: Configurar SSL/TLS en AL2023](#).

Paso 2: Probar el servidor LAMP

Si el servidor está instalado y en funcionamiento, y tiene establecidos correctamente los permisos de archivos, la cuenta `ec2-user` debería poder crear un archivo PHP en el directorio `/var/www/html`, que está disponible en Internet.

Para probar el servidor LAMP

1. Cree un archivo PHP en la raíz de documentos de Apache.

```
[ec2-user ~]$ echo "<?php phpinfo(); ?>" > /var/www/html/phpinfo.php
```

Si recibe el error "Permiso denegado" al intentar ejecutar este comando, intente cerrar sesión e iniciarla de nuevo para actualizar los permisos de grupo correctos que ha configurado en [Para establecer permisos de archivo](#).

2. En un navegador web, escriba la URL del archivo que acaba de crear. Esta URL es la dirección DNS pública de la instancia seguida de una barra diagonal y el nombre del archivo. Por ejemplo:

```
http://my.public.dns.amazonaws.com/phpinfo.php
```

Debería ver la página de información PHP.:

PHP Version 8.1.7



System	Linux ip-172-31-16-77.ec2.internal 5.15.57-28.127.amzn2022.aarch64 #1 SMP Thu Aug 4 17:06:57 UTC 2022 aarch64
Build Date	Jun 7 2022 18:21:38
Build System	Linux
Build Provider	Amazon Linux
Compiler	gcc (GCC) 11.3.1 20220421 (Red Hat 11.3.1-2)
Architecture	aarch64
Server API	FPM/FastCGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc
Loaded Configuration File	/etc/php.ini
Scan this dir for additional .ini files	/etc/php.d
Additional .ini files parsed	/etc/php.d/10-opcache.ini, /etc/php.d/20-bz2.ini, /etc/php.d/20-calendar.ini, /etc/php.d/20-ctype.ini, /etc/php.d/20-curl.ini, /etc/php.d/20-dom.ini, /etc/php.d/20-exif.ini, /etc/php.d/20-fileinfo.ini, /etc/php.d/20-ftp.ini, /etc/php.d/20-gd.ini, /etc/php.d/20-gettext.ini, /etc/php.d/20-iconv.ini, /etc/php.d/20-mbstring.ini, /etc/php.d/20-mysqli.ini, /etc/php.d/20-pdo.ini, /etc/php.d/20-phar.ini, /etc/php.d/20-simplexml.ini, /etc/php.d/20-sockets.ini, /etc/php.d/20-sqlite3.ini, /etc/php.d/20-tokenizer.ini, /etc/php.d/20-xml.ini, /etc/php.d/20-xmlwriter.ini, /etc/php.d/20-xsl.ini, /etc/php.d/30-mysqli.ini, /etc/php.d/30-pdo_mysql.ini, /etc/php.d/30-pdo_sqlite.ini, /etc/php.d/30-xmlreader.ini
PHP API	20210902
PHP Extension	20210902
Zend Extension	420210902
Zend Extension Build	API420210902,NTS
PHP Extension Build	API20210902,NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	enabled
Zend Memory Manager	enabled
Zend Multibyte Support	provided by mbstring
IPv6 Support	enabled
DTrace Support	available, disabled
Registered PHP Streams	https, ftps, compress.zlib, php, file, glob, data, http, ftp, compress.bzip2, phar
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, tls, tlsv1.0, tlsv1.1, tlsv1.2, tlsv1.3
Registered Stream Filters	zlib.*, string.rot13, string.toupper, string.tolower, convert.*, consumed, dechunk, bzip2.*, convert.iconv.*

This program makes use of the Zend Scripting Language Engine:
 Zend Engine v4.1.7, Copyright (c) Zend Technologies
 with Zend OPcache v8.1.7, Copyright (c), by Zend Technologies

Si no ve esta página, compruebe que el archivo `/var/www/html/phpinfo.php` se ha creado correctamente en el paso anterior. También puede verificar que todos los paquetes necesarios se han instalado con el comando siguiente:

```
[ec2-user ~]$ sudo dnf list installed httpd mariadb-server php-mysqldb
```

Si alguno de los paquetes requeridos no aparece en la salida, instálelo con el comando `sudo yum install package`.

3. Elimine el archivo `phpinfo.php`. Aunque esta información puede resultar útil, no se debe difundir por Internet por motivos de seguridad.

```
[ec2-user ~]$ rm /var/www/html/phpinfo.php
```

Ahora ya tiene un servidor web LAMP totalmente funcional. Si añade contenido a la raíz del documento de Apache en `/var/www/html`, debería poder ver ese contenido en la dirección DNS pública de su instancia.

Paso 3: Proteger el servidor de base de datos

La instalación predeterminada del servidor MariaDB posee varias características que son perfectas para las pruebas y el desarrollo, pero que se deben deshabilitar o eliminar para los servidores de producción. El comando `mysql_secure_installation` le guía a través del proceso de configuración de una contraseña raíz y de eliminación de las características que no son seguras de la instalación. Aunque no tenga pensado utilizar el servidor MariaDB, recomendamos realizar este procedimiento.

Para proteger el servidor MariaDB

1. Inicie el servidor MariaDB.

```
[ec2-user ~]$ sudo systemctl start mariadb
```

2. Ejecute `mysql_secure_installation`.

```
[ec2-user ~]$ sudo mysql_secure_installation
```

- a. Cuando se le solicite, escriba una contraseña para la cuenta raíz.
 - i. Escriba la contraseña raíz actual. De manera predeterminada, la cuenta raíz no tiene configurada ninguna contraseña. Pulse Intro.
 - ii. Escriba **Y** para configurar una contraseña y escriba una contraseña segura dos veces. Para obtener más información sobre cómo crear una contraseña segura, consulte. <https://identitysafe.norton.com/password-generator/> Procure guardar esta contraseña en un lugar seguro.

La configuración de una contraseña raíz para MariaDB es solo la medida más básica para proteger la base de datos. Cuando se crea o se instala una aplicación basada

en base de datos, normalmente se crea un usuario del servicio de la base de datos para esa aplicación y se evita usar la cuenta raíz para cualquier cosa que no sea la administración de la base de datos.

- b. Escriba **Y** para eliminar las cuentas de usuarios anónimos.
 - c. Escriba **Y** para deshabilitar el inicio de sesión raíz remoto.
 - d. Escriba **Y** para eliminar la base de datos de prueba.
 - e. Escriba **Y** para volver a cargar las tablas de privilegios y guardar los cambios.
3. (Opcional) Si no tiene pensado utilizar el servidor MariaDB de inmediato, deténgalo. Puede reiniciarlo cuando lo necesite.

```
[ec2-user ~]$ sudo systemctl stop mariadb
```

4. (Opcional) Si desea que el servidor MariaDB se inicie cada vez que arranque el sistema, escriba el siguiente comando.

```
[ec2-user ~]$ sudo systemctl enable mariadb
```

Paso 4: Instalación (opcional) phpMyAdmin

[phpMyAdmin](#) es una herramienta de administración de bases de datos basada en la web que puede utilizar para ver y editar las bases de datos MySQL de su instancia EC2. Siga estos pasos para instalar y configurar phpMyAdmin en la instancia de Amazon Linux.

Important

No recomendamos utilizarla phpMyAdmin para acceder a un servidor LAMP a menos que lo tenga activado SSL/TLS en Apache; de lo contrario, la contraseña de administrador de la base de datos y otros datos se transmitirán de forma insegura a través de Internet. Para ver las recomendaciones de seguridad de los desarrolladores, consulte [Proteger la phpMyAdmin instalación](#). Para obtener información general sobre asegurar un servidor web en una instancia de EC2, consulte [Tutorial: Configurar SSL/TLS en AL2023](#).

Para instalar phpMyAdmin

1. Instale las dependencias requeridas.

```
[ec2-user ~]$ sudo dnf install php-mbstring php-xml -y
```

2. Reinicie Apache.

```
[ec2-user ~]$ sudo systemctl restart httpd
```

3. Reinicie php-fpm.

```
[ec2-user ~]$ sudo systemctl restart php-fpm
```

4. Navegue a la raíz de documentos de Apache: /var/www/html.

```
[ec2-user ~]$ cd /var/www/html
```

5. Seleccione un paquete fuente para la phpMyAdmin versión más reciente en <https://www.phpmyadmin.net/downloads>. Para descargar el archivo directamente a la instancia, copie el link y péguelo en un comando wget como el de este ejemplo:

```
[ec2-user html]$ wget https://www.phpmyadmin.net/downloads/phpMyAdmin-latest-all-languages.tar.gz
```

6. Cree la carpeta phpMyAdmin y extraiga aquí el paquete con el comando siguiente:

```
[ec2-user html]$ mkdir phpMyAdmin && tar -xvzf phpMyAdmin-latest-all-languages.tar.gz -C phpMyAdmin --strip-components 1
```

7. Elimine el *phpMyAdmin-latest-all-languages.tar.gz* archivo tar.

```
[ec2-user html]$ rm phpMyAdmin-latest-all-languages.tar.gz
```

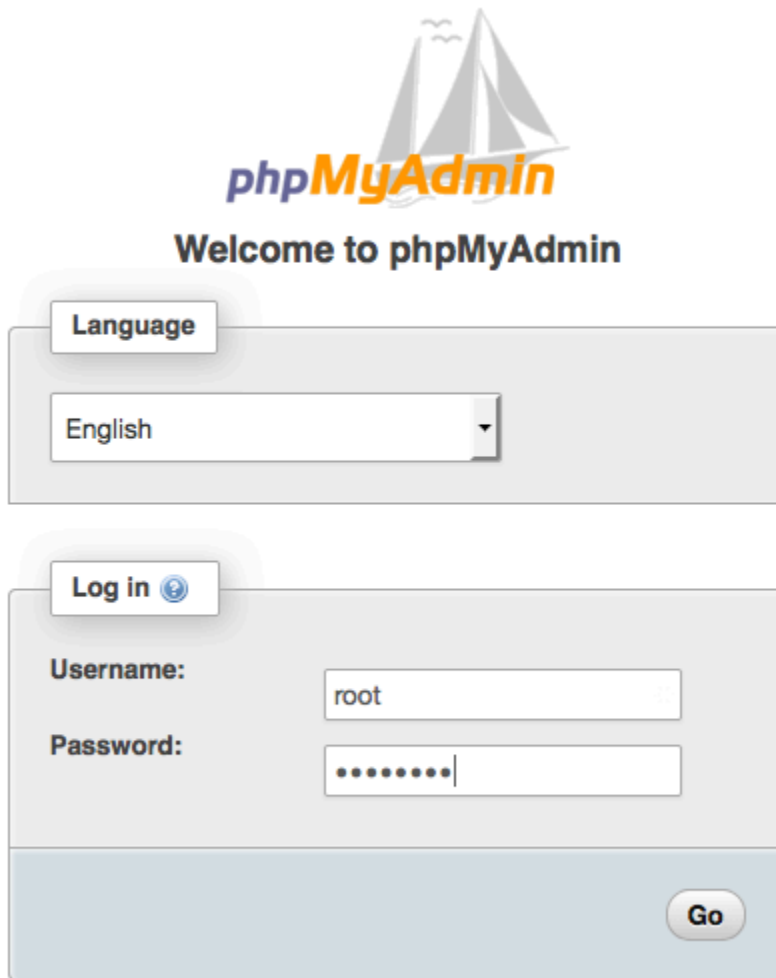
8. (Opcional) Si el servidor MySQL no está en ejecución, inícielo ahora.

```
[ec2-user ~]$ sudo systemctl start mariadb
```

9. En un navegador web, escriba la URL de la phpMyAdmin instalación. Esta URL es la dirección DNS pública (o la dirección IP pública) de la instancia, seguida de una barra inclinada y el nombre del directorio de instalación. Por ejemplo:

```
http://my.public.dns.amazonaws.com/phpMyAdmin
```

Debería ver la página de phpMyAdmin inicio de sesión:



phpMyAdmin

Welcome to phpMyAdmin

Language

English

Log in ?

Username: root

Password:

Go

10. Inicie sesión en la phpMyAdmin instalación con el nombre `root` de usuario y la contraseña `root` de MySQL que creó anteriormente.

Antes de poner en servicio la instalación, debe configurarla. Le sugerimos que comience con la creación manual del archivo de configuración de la siguiente manera:

- a. Para comenzar con un archivo de configuración mínimo, utilice su editor de texto favorito para crear un archivo nuevo y luego, copie el contenido de `config.sample.inc.php` en él.
- b. Guarde el archivo tal y como `config.inc.php` está en el phpMyAdmin directorio que lo contiene `index.php`.

- c. Consulte las instrucciones posteriores a la creación del archivo en la sección [Uso del script de configuración](#) de las instrucciones de phpMyAdmin instalación para cualquier configuración adicional.

Para obtener información sobre su uso phpMyAdmin, consulte la [Guía del phpMyAdmin usuario](#).

Solución de problemas

En esta sección, se ofrecen sugerencias para resolver los problemas comunes que puede encontrarse al configurar un servidor LAMP nuevo.

No puedo conectarme a mi servidor mediante un navegador web.

Realice las siguientes verificaciones para ver si el servidor web Apache funciona y se puede obtener acceso a él.

- ¿El servidor web funciona?

Puede verificar que httpd está activo ejecutando el siguiente comando:

```
[ec2-user ~]$ sudo systemctl is-enabled httpd
```

Si el proceso httpd no se está ejecutando, repita los pasos que se describen en [Para preparar el servidor LAMP](#).

- ¿El firewall está configurado correctamente?

Compruebe que el grupo de seguridad de la instancia contenga una regla para permitir el tráfico HTTP en el puerto 80. Para obtener más información, consulte [Agregar reglas al grupo de seguridad](#).

No puedo conectarme a mi servidor mediante HTTPS.

Realice las siguientes verificaciones para verificar si su servidor web Apache está configurado para admitir HTTPS.

- ¿El servidor web está configurado correctamente?

Después de instalar Apache, el servidor se configura para el tráfico HTTP. Para admitir HTTPS, habilite TLS en el servidor e instale un certificado SSL. Para obtener información, consulte [Tutorial: Configurar SSL/TLS en AL2023](#).

- ¿El firewall está configurado correctamente?

Verifique que el grupo de seguridad de la instancia contiene una regla para permitir el tráfico HTTPS en el puerto 443. Para obtener más información, consulte [Autorización del tráfico entrante para sus instancias de Linux](#).

Temas relacionados

Para obtener más información sobre la transferencia de archivos a la instancia o la instalación de un WordPress blog en el servidor web, consulte la siguiente documentación:

- [Transferencia de archivos a la instancia de Linux mediante WinSCP](#) en la Guía del usuario de Amazon EC2.
- [Transferencia de archivos a instancias de Linux mediante un cliente SCP](#) en la Guía del usuario de Amazon EC2.
- [Tutorial: alojar un WordPress blog en AL2023](#)

Para obtener más información acerca de los comandos y el software que se utilizan en este tutorial, visite las siguientes páginas web:

- Servidor web Apache: <http://httpd.apache.org/>
- Servidor de bases de datos MariaDB: <https://mariadb.org/>
- Lenguaje de programación PHP: <http://php.net/>

Para obtener más información acerca del registro de un nombre de dominio para el servidor web o la transferencia de un nombre de dominio existente a este host, consulte el tema [Creación y migración de dominios y subdominios a Amazon Route 53](#) en la Guía para desarrolladores de Amazon Route 53.

Tutorial: Configurar SSL/TLS en AL2023

Layer/Transport Secure Sockets Layer Security (SSL/TLS) creates an encrypted channel between a web server and web client that protects data in transit from being eavesdropped on. This tutorial explains how to add support manually for SSL/TLS en una instancia EC2 con AL2023 un servidor web Apache). En este tutorial, se presupone que no está utilizando un balanceador de carga. Si utiliza Elastic Load Balancing, puede elegir configurar la descarga SSL en el balanceador de carga, mediante un certificado de [AWS Certificate Manager](#) en su lugar.

Por motivos históricos, el cifrado web se suele denominar simplemente SSL. Aunque los navegadores web siguen admitiendo SSL, el protocolo que lo sustituye, TLS, es menos vulnerable a los ataques. AL2023 deshabilita el soporte del lado del servidor para todas las versiones de SSL de forma predeterminada. [Organismos de estándares de seguridad](#) consideran que TLS 1.0 no es seguro. TLS 1.0 y TLS 1.1 han quedado formalmente [obsoletos](#) en marzo de 2021. Este tutorial contiene asesoramiento basado exclusivamente en la habilitación de TLS 1.2. TLS 1.3 se finalizó en 2018 y estará disponible AL2 siempre que se admita y habilite la biblioteca TLS subyacente (OpenSSL en este tutorial). [Los clientes deben ser compatibles con TLS 1.2 o una versión posterior antes del 28 de junio de 2023](#). Para obtener más información sobre el estándar de cifrado actualizado, consulte [RFC 7568](#) y [RFC 8446](#).

Este tutorial se refiere a un cifrado web moderno tan simple como TLS.

Important

Estos procedimientos son para usar con AL2023. Si intenta configurar una instancia de EC2 que ejecute una distribución diferente o una instancia que ejecute una versión anterior de Amazon Linux, es posible que algunos procedimientos de este tutorial no funcionen. Para Ubuntu, consulte la siguiente documentación de la comunidad Ubuntu: [Open SSL on Ubuntu](#) (Abrir SSL en Ubuntu). Para Red Hat Enterprise Linux, consulte lo siguiente: [Configuración del servidor web HTTP Apache](#). Para otras distribuciones, consulte su documentación específica.

Note

Como alternativa, puede usar AWS Certificate Manager (ACM) para los enclaves de AWS Nitro, que es una aplicación de enclave que le permite usar SSL/TLS certificados públicos y privados con sus aplicaciones web y servidores que se ejecutan en instancias de Amazon

EC2 con Nitro Enclaves. AWS Nitro Enclaves es una capacidad de Amazon EC2 que permite la creación de entornos informáticos aislados para proteger y procesar de forma segura datos altamente confidenciales, SSL/TLS como certificados y claves privadas.

ACM para Nitro Enclaves funciona con nginx ejecutándose en su instancia de Amazon EC2 Linux para crear claves privadas, distribuir certificados y claves privadas y para administrar las renovaciones de certificados.

Para utilizar ACM para Nitro Enclaves, debe utilizar una instancia Linux habilitada para enclave.

Para obtener más información, consulte [¿Qué es AWS Nitro Enclaves?](#) y [AWS Certificate Manager para ver Nitro Enclaves](#) en la Guía del usuario de AWS Nitro Enclaves.

Contenido

- [Requisitos previos](#)
- [Paso 1: Habilitar TLS en el servidor](#)
- [Paso 2: Obtener un certificado firmado por una CA](#)
- [Paso 3: Probar y reforzar la configuración de seguridad](#)
- [Solución de problemas](#)

Requisitos previos

Siga estos pasos antes de comenzar este tutorial:

- Lance una instancia respaldada por EBS. AL2023 Para obtener más información, consulte [AL2023 en Amazon EC2](#).
- Configure los grupos de seguridad para que la instancia acepte conexiones en los siguientes puertos TCP:
 - SSH (puerto 22)
 - HTTP (puerto 80)
 - HTTPS (puerto 443)

Para obtener más información, consulte [Autorización del tráfico entrante para sus instancias de Linux](#) en la Guía del usuario de Amazon EC2.

- Instale el servidor web Apache. Para obtener step-by-step instrucciones, consulte. [Tutorial: Instalar un servidor LAMP en AL2023](#) Solo es necesario el paquete httpd y sus dependencias; por lo que puede omitir las instrucciones relacionadas con PHP y MariaDB.
- Para identificar y autenticar sitios web, la infraestructura de clave pública (PKI) de TLS se basa en el sistema de nombres de dominio (DNS). Si tiene pensado utilizar la instancia de EC2 para alojar un sitio web público, tiene que registrar un nombre de dominio para el servidor web o transferir un nombre de dominio existente al host de Amazon EC2. Para hacer esto, hay disponibles numerosos servicios de registro de dominios y alojamiento de DNS de terceros o bien puede utilizar [Amazon Route 53](#).

Paso 1: Habilitar TLS en el servidor

Este procedimiento le guiará por el proceso de configurar el TLS AL2023 con un certificado digital autofirmado.

Note

Se puede utilizar un certificado autofirmado para las pruebas, pero no para la producción. Si expone a Internet su certificado autofirmado, los visitantes del sitio recibirán advertencias de seguridad.

Para habilitar TLS en un servidor

1. Conéctese a su instancia y confirme que Apache se está ejecutando. Para obtener más información, consulte [Conexión a instancias AL2023](#).

```
[ec2-user ~]$ sudo systemctl is-enabled httpd
```

Si el valor devuelto no es "enabled", inicie Apache y configúrelo para que se inicie cada vez que arranque el sistema.

```
[ec2-user ~]$ sudo systemctl start httpd && sudo systemctl enable httpd
```

2. Para asegurarse de que todos los paquetes de software están actualizados, realice una actualización rápida del software en la instancia. Este proceso puede durar unos minutos, pero es importante realizarlo para asegurarse de que tiene las actualizaciones de seguridad y las correcciones de errores más recientes.

 Note

La opción `-y` instala las actualizaciones sin necesidad de confirmación. Si le gustaría examinar las actualizaciones antes de la instalación, puede omitir esta opción.

```
[ec2-user ~]$ sudo dnf install openssl mod_ssl
```

3. Después de ingresar el siguiente comando, se lo dirigirá a un mensaje donde podrá ingresar información sobre su sitio.

```
[ec2-user ~]$ sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/pki/tls/private/apache-selfsigned.key -out /etc/pki/tls/certs/apache-selfsigned.crt
```

Esto genera dos archivos:

- `/etc/pki/tls/private/apache-selfsigned.key`(la clave privada)
- `/etc/pki/tls/certs/apache-selfsigned.crt`(el certificado autofirmado)

El nombre del archivo del certificado coincide con el predeterminado asignado en la `SSLCertificateFile` directiva en `/etc/httpd/conf.d/ssl.conf`.

La instancia tiene ahora los archivos siguientes que utiliza para configurar el servidor seguro y crear un certificado para pruebas:

- `/etc/httpd/conf.d/ssl.conf`

El archivo de configuración para `mod_ssl`. Contiene directivas que le indican a Apache donde encontrar claves de cifrado y certificados, las versiones de protocolo TLS que se permiten y los cifrados que se aceptan. Este será el archivo de certificado local:

- `/etc/pki/tls/certs/apache-selfsigned.crt`
- `/etc/pki/tls/private/apache-selfsigned.key`

El `.crt` archivo contiene el certificado autofirmado y la `.key` clave privada. Apache requiere que el certificado y la clave estén en formato PEM que consta de caracteres ASCII codificados

en Base64 contenidos entre las líneas "BEGIN" y "END", como en el siguiente ejemplo abreviado.

```
-----BEGIN PRIVATE KEY-----
MIIEvgIBADANBgkqhkiG9w0BAQEFAASCBAgEAAoIBAQD2KKx/8Zk94m1q
3gQMZF9ZN66Ls19+3tHAgQ5Fpo9KJDhzLj00CI8u1PTcGmAah5kEitCEc0wzmNeo
BC10wYR6G0rGaKtK9Dn7CuIjvubtUysVyQoMVPQ971deakHWeRMiEJFXg6kZZ0vr
GvwnKoMh3D1K44D9dX7IDua2P1Yx5+eroA+1Lqf32ZSaA00bBIMIYTHigwbHMZoT
...
56tE7THvH7v0Ef4/iU0sIrEzaMaJ0mqkmY1A70qQGQKBgBF3H1qNRNHuyMcP0DFs
27hDzPDinrquSEvoZIggkDM1h2irTiipJ/GhkvtPoQlv0fK/VXw8vSgeaBuhwJvS
LXU9HvYq0U604FgD3nAyB9hI0BE13r1HjUvbjT7moH+RhNz6eqqdsccCS09VtRAo
4QQvAq0a8UheYeoXLdWcHaLP
-----END PRIVATE KEY-----

-----BEGIN CERTIFICATE-----
MIIEAzCCA10gAwIBAgICWxQwDQYJKoZIhvcNAQELBQAwbGExCzAJBgNVBAYTAi0t
MRIwEAYDVQQIDAlTb21lU3RhdGUxETAPBgNVBACMFNvbWVkaXR5MRkwFwYDVQQK
DBBTb21lT3JnYW5pemF0aW9uMR8wHQYDVQQLDBZTb21lT3JnYW5pemF0aW9uYXV
bm10MRkwFwYDVQQDDDBBpcC0xNzItMzEtMjAtMjMMSQwIgYJKoZIhvcNAQkBFhVy
...
z5rRUE/XzxRLBZ0oWZpNWTXJkQ3uFYH6s/sBwtHpKKZMz0vDedREjNKAvk4ws6F0
CuIjvubtUysVyQoMVPQ971deakHWeRMiEJFXg6kZZ0vrGvwnKoMh3D1K44D9d1U3
WanXWehT6FiSZvB4sTEXXJN2jdw8g+sHGnZ8zC0sc1knYhHrCVD2vnB1ZJKSZvak
3ZazhBxtQSukFM0nWPP2a0DMMFGYUHOd0BQE8sBJxg==
-----END CERTIFICATE-----
```

Los nombres y las extensiones de los archivos se utilizan simplemente por comodidad y no afectan al funcionamiento. Por ejemplo, puede llamar a un certificado `cert.crt`, `cert.pem`, o cualquier otro nombre de archivo, siempre que la directiva relacionada en el archivo `ssl.conf` utilice el mismo nombre.

Note

Cuando sustituya los archivos TLS predeterminados por sus propios archivos personalizados, asegúrese de que estén en formato PEM.

4. Reinicie Apache.

```
[ec2-user ~]$ sudo systemctl restart httpd
```

 Note

Asegúrese de que el puerto 443 de TCP sea accesible en la instancia de EC2, tal y como se ha explicado anteriormente.

5. Ahora el servidor web de Apache debería admitir HTTPS (HTTP seguro) en el puerto 443. Pruébalo escribiendo la dirección IP o el nombre completo de dominio de la instancia de EC2 en la barra URL del navegador con el prefijo **https://**.

Dado que se va a conectar a un sitio con un certificado de host autofirmado que no es de confianza, es posible que el navegador envíe una serie de advertencias de seguridad. Omita las advertencias y vaya al sitio.

Si se abre la página de prueba de Apache predeterminada, eso significa que ha configurado correctamente TLS en el servidor. Ahora todos los datos que pasan entre el navegador y el servidor se cifran.

 Note

Para evitar que los visitantes del sitio reciban pantallas de advertencia, tiene que obtener un certificado de confianza firmado por una CA que no solo cifre, sino que también le autentique públicamente como propietario del sitio.

Paso 2: Obtener un certificado firmado por una CA

Puede utilizar el siguiente proceso para obtener un certificado firmado por una CA:

- Genere una solicitud de firma de certificado (CSR) de una clave privada
- Envíe el CSR a una autoridad de certificación (CA)
- Obtenga un certificado de host firmado
- Configure Apache para utilizar el certificado

Un certificado de host TLS X.509 autofirmado es idéntico, desde el punto de vista criptográfico, a un certificado firmado por una CA. La diferencia es social, no matemática. Una CA promete, como mínimo, validar la propiedad de un dominio antes de emitir un certificado para el solicitante. Cada

navegador web contiene una lista de las personas CAs en las que el proveedor del navegador confía para ello. Un certificado X.509 consta principalmente de una clave pública, que se corresponde con la clave del servidor privado, y una firma de la CA que está vinculada criptográficamente a la clave pública. Cuando un navegador se conecta a un servidor web a través de HTTPS, el servidor presenta un certificado para que el navegador lo compare con su lista de usuarios de confianza CAs. Si el signatario está en la lista o se puede obtener acceso a él a través de una cadena de confianza compuesta por otros signatarios de confianza, el navegador negocia un canal de datos cifrados rápido con el servidor y carga la página.

Generalmente, los certificados cuestan dinero por el trabajo que supone la validación de las solicitudes, por lo que vale la pena comparar precios. Algunos CAs ofrecen certificados de nivel básico de forma gratuita. El más notable de ellos CAs es el proyecto [Let's Encrypt](#), que también apoya la automatización del proceso de creación y renovación de certificados. Para obtener más información acerca del uso del certificado Let's Encrypt, consulte [Obtenga Certbot](#).

Si planea ofrecer servicios de calidad comercial, [AWS Certificate Manager](#) es una buena opción.

La clave es hacer que el certificado del host esté subyacente. Desde 2019, grupos del [gobierno](#) y el [sector](#) recomiendan utilizar un tamaño mínimo de clave (módulo) de 2048 bits para claves RSA destinadas a proteger documentos hasta 2030. El tamaño de módulo predeterminado generado por OpenSSL es de 2048 bits, lo que es adecuado para su uso AL2023 en un certificado firmado por una CA. En el siguiente procedimiento, se proporcionó un paso opcional para aquellos que desearan una clave personalizada, por ejemplo, una con un módulo mayor o que utilice un algoritmo de cifrado diferente.

 Important

Estas instrucciones para adquirir un certificado de host firmado por una CA no funcionan a menos que posea un dominio DNS alojado y registrado.

Para obtener un certificado firmado por una CA

1. Conéctese a su instancia y vaya a `/etc/pki/tls/private/`. Este es el directorio donde almacena la clave privada del servidor para TLS. Si prefiere utilizar una clave de host existente para generar el CSR, vaya al paso 3. Para obtener más información sobre cómo conectarse a tu instancia, consulta [Conexión a instancias AL2023](#)

2. (Opcional) Genere una nueva clave privada. Estas son algunas ejemplos de configuraciones clave. Cualquiera de las claves resultantes funciona con el servidor web, pero son diferentes en el grado y el tipo de seguridad que implementan.

- Ejemplo 1: cree una clave de host RSA predeterminada. El archivo resultante, **custom.key**, es una clave privada RSA de 2048 bits.

```
[ec2-user ~]$ sudo openssl genrsa -out custom.key
```

- Ejemplo 2: cree una clave RSA más segura con un módulo mayor. El archivo resultante, **custom.key**, es una clave privada RSA de 4096 bits.

```
[ec2-user ~]$ sudo openssl genrsa -out custom.key 4096
```

- Ejemplo 3: cree una clave RSA cifrada de 4096 bits con protección con contraseña. El archivo resultante, **custom.key**, es una clave privada RSA de 4096 bits cifrada con AES-128.

 Important

El cifrado de la clave ofrece mayor seguridad, pero dado que una clave cifrada necesita una contraseña, los servicios que dependen de él no se pueden iniciar automáticamente. Cada vez que utilice esta clave, tiene que proporcionar la contraseña (en el ejemplo anterior "abcde12345") en una conexión SSH.

```
[ec2-user ~]$ sudo openssl genrsa -aes128 -passout pass:abcde12345 -out custom.key 4096
```

- Ejemplo 4: cree una clave con un cifrado que no sea RSA. La criptografía RSA puede ser relativamente lenta debido al tamaño de sus claves públicas, que se basan en el producto de dos números primos grandes. Sin embargo, es posible crear claves para TLS que utilicen cifrados que no sean RSA. Las claves que están basadas en el cálculo matemático de curvas elípticas son más pequeñas y más rápidas desde el punto de vista informático a la hora de proporcionar un nivel de seguridad equivalente.

```
[ec2-user ~]$ sudo openssl ecparam -name prime256v1 -out custom.key -genkey
```

El resultado es una clave privada de curva elíptica de 256 bits que utiliza prime256v1, una "curva con nombre" que admite OpenSSL. Su seguridad criptográfica es ligeramente mayor que la de una clave RSA de 2048 bits, [de acuerdo con NIST](#).

 Note

No todas CAs ofrecen el mismo nivel de compatibilidad con las elliptic-curve-based claves que con las claves RSA.

Asegúrese de que la nueva clave privada tenga una propiedad y unos permisos muy restrictivos (owner=root, group=root, solo para el propietario). read/write Los comandos serán como se muestra en el siguiente ejemplo.

```
[ec2-user ~]$ sudo chown root:root custom.key
[ec2-user ~]$ sudo chmod 600 custom.key
[ec2-user ~]$ ls -al custom.key
```

Los comandos anteriores devuelven el siguiente resultado.

```
-rw----- root root custom.key
```

Una vez que haya creado y configurado una clave satisfactoria, puede crear una CSR.

3. Para crear una CSR, utilice su clave preferida. El siguiente ejemplo utiliza **custom.key**.

```
[ec2-user ~]$ sudo openssl req -new -key custom.key -out csr.pem
```

OpenSSL abre un cuadro de diálogo y le pide la información que se muestra en la siguiente tabla. Todos los campos, excepto Common Name (Nombre común), son opcionales para un certificado de host de dominio validado básico.

Nombre	Descripción	Ejemplo
Country Name	La abreviatura ISO de dos letras del país.	US (= Estados Unidos)

Nombre	Descripción	Ejemplo
State or Province Name	Nombre del país o de la región donde se ubica su organización. Este nombre no se puede abreviar.	Washington
Locality Name	La ubicación de su organización, como una ciudad.	Seattle
Organization Name	Nombre legal completo de su organización. No abrevie el nombre de la organización.	Empresa de ejemplo
Organizational Unit Name	Información adicional de la organización, si existe.	Departamento de ejemplo
Common Name	Este valor debe ser exactamente igual que la dirección web que espera que introduzcan los usuarios en un navegador. Normalmente, tiene que ser un nombre de dominio precedido por un nombre de host o alias con el formato www.example.com . En las pruebas con un certificado autofirmado y sin resolución de DNS, el nombre común puede consistir únicamente en el nombre de host. CAs también ofrecen certificados más caros que aceptan nombres comodín, como *.example.com .	www.example.com
Email Address	Dirección de correo electrónico del administrador del servidor.	alguien@ejemplo.com

Por último, OpenSSL le solicita una contraseña de comprobación opcional. Esta contraseña solo se aplica a la CSR y a las transacciones entre usted y la CA, así que siga las recomendaciones de la CA a este respecto y el otro campo opcional, es decir, el nombre de empresa opcional. La contraseña de comprobación de CSR no afecta al funcionamiento del servidor.

El archivo resultante, **csr.pem** contiene la clave pública, la firma digital de la clave pública y los metadatos que ha especificado.

- Envíe la CSR a una CA. Este proceso suele consistir en abrir el archivo CSR en un editor de texto y copiar el contenido en un formulario web. En este momento, es posible que se le pida que proporcione uno o más nombres alternativos de sujeto (SANs) para incluirlos en el certificado. Si el nombre común es **www.example.com**, **example.com** sería un buen SAN y viceversa. Un visitante de su sitio que introdujese cualquiera de estos nombres no recibiría ningún error de conexión. Si su formulario web de CA lo permite, incluya el nombre común en la lista de SANs. Algunos lo CAs incluyen automáticamente.

Una vez aprobada su solicitud, recibe un nuevo certificado de host firmado por la CA. Es posible que también tenga que descargar un archivo de certificado intermedio que contiene los certificados adicionales necesarios para completar la cadena de confianza de la CA.

Note

La CA puede enviarle archivos en diversos formatos destinados a diversos fines. Para este tutorial, solo debe utilizar un archivo de certificado en formato PEM, que normalmente (aunque no siempre) está marcado con la extensión de archivo `.pem` o `.crt`. Si no tiene claro qué archivo debe utilizar, abra los archivos con un editor de texto y busque el que contiene uno o varios bloques que comienzan con la siguiente línea.

```
- - - - -BEGIN CERTIFICATE - - - - -
```

El archivo debería terminar también con la siguiente línea.

```
- - - - -END CERTIFICATE - - - - -
```

También puede probar el archivo en la línea de comandos, tal y como se muestra a continuación.

```
[ec2-user certs]$ openssl x509 -in certificate.crt -text
```

Verifique que estas líneas aparecen en el archivo. No utilice archivos que terminen con `.p7b`, `.p7c` o extensiones de archivos similares.

5. Coloque el nuevo certificado firmado por la CA y cualquier certificado intermedio en el directorio `/etc/pki/tls/certs`.

 Note

Hay varias formas de cargar el nuevo certificado en la instancia de EC2, pero la más sencilla e informativa consiste en abrir un editor de texto (por ejemplo, vi, nano o notepad) en el equipo local y en la instancia y, a continuación, copiar y pegar el contenido del archivo de uno al otro. Para realizar estas operaciones en la instancia de EC2, necesita permisos de raíz [sudo]. De esta forma, puede ver inmediatamente si hay algún problema con los permisos o las rutas. Procure, no obstante, no añadir más líneas al copiar el contenido o cambiarlo de ninguna forma.

Desde el `/etc/pki/tls/certs` directorio, compruebe que la configuración de propiedad, grupo y permisos del archivo coincide con los AL2023 valores predeterminados altamente restrictivos (owner=root, group=root, solo para propietario). read/write En el siguiente ejemplo, se muestran los comandos que se utilizarán.

```
[ec2-user certs]$ sudo chown root:root custom.crt
[ec2-user certs]$ sudo chmod 600 custom.crt
[ec2-user certs]$ ls -al custom.crt
```

Estos comandos deberían devolver el siguiente resultado.

```
-rw----- root root custom.crt
```

Los permisos del archivo de certificado intermedio son menos estrictos (propietario=raíz, grupo=raíz, propietario puede escribir, grupo puede leer, mundo puede leer). En el siguiente ejemplo, se muestran los comandos que se utilizarán.

```
[ec2-user certs]$ sudo chown root:root intermediate.crt
[ec2-user certs]$ sudo chmod 644 intermediate.crt
[ec2-user certs]$ ls -al intermediate.crt
```

Estos comandos deberían devolver el siguiente resultado.

```
-rw-r--r-- root root intermediate.crt
```

6. Coloque la clave privada que utilizó para crear la CSR en el directorio `/etc/pki/tls/private/`.

 Note

Hay varias formas de cargar la clave personalizada en la instancia de EC2, pero la más sencilla e informativa consiste en abrir un editor de texto (por ejemplo, vi, nano o notepad) en el equipo local y en la instancia y, a continuación, copiar y pegar el contenido del archivo de uno al otro. Para realizar estas operaciones en la instancia de EC2, necesita permisos de raíz [sudo]. De esta forma, puede ver inmediatamente si hay algún problema con los permisos o las rutas. Procure, no obstante, no añadir más líneas al copiar el contenido o cambiarlo de ninguna forma.

Desde el `/etc/pki/tls/private` directorio, utilice los siguientes comandos para comprobar que la configuración de propiedad, grupo y permisos del archivo coincide con los AL2023 valores predeterminados altamente restrictivos (owner=root, group=root, solo para propietario). read/write

```
[ec2-user private]$ sudo chown root:root custom.key
[ec2-user private]$ sudo chmod 600 custom.key
[ec2-user private]$ ls -al custom.key
```

Estos comandos deberían devolver el siguiente resultado.

```
-rw----- root root custom.key
```

7. Edite `/etc/httpd/conf.d/ssl.conf` para reflejar el nuevo certificado y los archivos de claves.

 Note

Asegúrese de que esté configurada en su bloque `SSLEngine` on. `VirtualHost`

- a. Proporcione la ruta y el nombre de archivo del certificado de host firmado por la CA en la directiva `SSLCertificateFile` de Apache:

```
SSLCertificateFile /etc/pki/tls/certs/custom.crt
```

- b. Si ha recibido un archivo de certificado intermedio (`intermediate.crt` en este ejemplo), proporcione su ruta y nombre de archivo utilizando la directiva `SSLCACertificateFile` de Apache:

```
SSLCACertificateFile /etc/pki/tls/certs/intermediate.crt
```

 Note

Algunos CAs combinan el certificado del anfitrión y los certificados intermedios en un solo archivo, lo que hace innecesaria la `SSLCACertificateFile` directiva. Consulte las instrucciones que le ha proporcionado su CA.

- c. Proporcione la ruta y el nombre de archivo de la clave privada (`custom.key` en este ejemplo) en la directiva `SSLCertificateKeyFile` de Apache:

```
SSLCertificateKeyFile /etc/pki/tls/private/custom.key
```

8. Guarde `/etc/httpd/conf.d/ssl.conf` y reinicie Apache.

```
[ec2-user ~]$ sudo systemctl restart httpd
```

9. Pruebe el servidor introduciendo su nombre de dominio en una barra de direcciones URL del navegador con el prefijo `https://`. El navegador debería cargar la página de prueba sobre HTTPS sin generar errores.

Paso 3: Probar y reforzar la configuración de seguridad

Cuando su TLS esté en funcionamiento y expuesto al público, debería probar su seguridad. Esta operación es muy sencilla con servicios online como [Qualys SSL Labs](#), que realiza un análisis gratuito y exhaustivo de su configuración de seguridad. En función de los resultados, puede decidir si debe reforzar la configuración de seguridad predeterminada controlando los protocolos que acepta,

qué cifrados prefiere y cuáles deben excluirse. Para obtener más información, consulte [cómo Qualys formula sus puntuaciones](#).

Important

Las pruebas reales son cruciales para la seguridad del servidor. Cualquier pequeño error de configuración puede provocar graves infracciones de seguridad y la pérdida de datos. Dado que las prácticas de seguridad recomendadas cambian continuamente en respuesta a las investigaciones y a las continuas amenazas, es esencial realizar auditorías de seguridad periódicas para mantener una buena administración del servidor.

En el sitio de [Qualys SSL Labs](#), introduzca el nombre de dominio completo de su servidor, con el formato **www.example.com**. Dos minutos después recibirá una puntuación (de A a F) de su sitio y un desglose detallado de los descubrimientos. En la siguiente tabla se resume el informe de un dominio con una configuración idéntica a la configuración predeterminada de Apache y con un certificado Certbot predeterminado. AL2023

Calificación global	B
Certificate	100%
Compatibilidad del protocolo	95%
Intercambio de clave	70 %
Seguridad del cifrado	90%

Aunque la información general muestra que la configuración es correcta, el informe detallado indica algunos posibles problemas, indicados aquí en orden de gravedad:

x Algunos RC4 navegadores antiguos admiten el uso del cifrado. Un sistema de cifrado es el núcleo matemático de un algoritmo de cifrado. RC4 [Se sabe que, un sistema de cifrado rápido que se utiliza para cifrar los flujos de datos de TLS, tiene varios puntos débiles graves](#). A menos que tenga muy buenas razones para admitir navegadores heredados, debería deshabilitar esta opción.

x Se admiten las versiones antiguas de TLS. La configuración admite TLS 1.0 (ya obsoleto) y TLS 1.1 (en vías de ser declarado obsoleto). Desde 2018 solo se ha recomendado TLS 1.2.

✗ La confidencialidad directa no se admite por completo. La [confidencialidad directa](#) es una característica de los algoritmos que cifra mediante claves de sesión temporales (efímeras) que se obtienen de la clave privada. En la práctica, esto significa que los atacantes no pueden descifrar los datos HTTPS aunque posean una clave privada a largo plazo del servidor web.

Para corregir y preparar para el futuro la configuración de TLS

1. Abra el archivo de configuración `/etc/httpd/conf.d/ssl.conf` en un editor de texto y comente la línea siguiente introduciendo `"#"` al principio de la línea.

```
#SSLProtocol all -SSLv3
```

2. Añada la siguiente directiva:

```
#SSLProtocol all -SSLv3  
SSLProtocol -SSLv2 -SSLv3 -TLSv1 -TLSv1.1 +TLSv1.2
```

Esta directiva deshabilita explícitamente las versiones 2 y 3 de SSL, además de las versiones 1.0 y 1.1 de TLS. Ahora, el servidor se niega a aceptar conexiones cifradas con clientes que no utilicen versiones compatibles de TLS 1.2. El texto de la directiva transmite con más claridad a un lector humano las acciones que se han configurado que haga el servidor.

Note

Al deshabilitar las versiones 1.0 y 1.1 de TLS de esta manera, bloquea un pequeño porcentaje de navegadores web desactualizados y evita que obtengan acceso a su sitio.

Para modificar la lista de cifrados permitidos

1. En el archivo de configuración `/etc/httpd/conf.d/ssl.conf`, encuentre la sección con la directiva **SSLCipherSuite** y comente la línea existente al introducir `"#"` al principio de la línea.

```
#SSLCipherSuite HIGH:MEDIUM:!aNULL:!MD5
```

2. Especifique conjuntos de cifrado explícitos y un orden de cifrado que dé prioridad a la confidencialidad directa y evite los cifrados inseguros. La directiva `SSLCipherSuite` que se utiliza aquí se basa en el resultado del [Mozilla SSL Configuration Generator](#), que personaliza

una configuración de TLS al software específico que se ejecuta en su servidor. En primer lugar, determine sus versiones de Apache y OpenSSL utilizando la salida de los comandos siguientes.

```
[ec2-user ~]$ yum list installed | grep httpd
```

```
[ec2-user ~]$ yum list installed | grep openssl
```

Por ejemplo, si la información devuelta es Apache 2.4.34 y OpenSSL 1.0.2, lo introducimos en el generador. Después elegimos el modelo de compatibilidad «moderno», esto crea una directiva `SSLCipherSuite` que aplica seguridad de forma agresiva pero sigue funcionando para la mayoría de navegadores. Si el navegador no admite la configuración moderna, puede actualizar el software o elegir la configuración «intermedia» en su lugar.

```
SSLCipherSuite ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-  
ECDSA-CHACHA20-POLY1305:  
ECDHE-RSA-CHACHA20-POLY1305:ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-GCM-  
SHA256:  
ECDHE-ECDSA-AES256-SHA384:ECDHE-RSA-AES256-SHA384:ECDHE-ECDSA-AES128-SHA256:ECDHE-  
RSA-AES128-SHA256
```

Los cifrados con la calificación más alta tienen ECDHE en su nombre, una abreviatura de Elliptic Curve Diffie-Hellman Ephemeral. El término ephemeral indica confidencialidad directa. Como subproducto, estos cifrados no son compatibles. RC4

Le recomendamos utilizar una lista de cifrados explícita en lugar de utilizar valores predeterminados o directivas escuetas cuyo contenido no es visible.

Copie la directiva generada e `/etc/httpd/conf.d/ssl.conf`.

Note

Aunque aquí se muestran en varias líneas para que la lectura sea más sencilla, la directiva debe estar en una sola línea cuando se copia a `/etc/httpd/conf.d/ssl.conf` con un solo símbolo de dos puntos (sin espacios) entre los nombres de los cifrados.

3. Por último, cancele el comentario de la siguiente línea eliminando el `#` al principio de la línea.

```
#SSLHonorCipherOrder on
```

Esta directiva obliga al servidor a preferir cifrados con una calificación alta, incluidos (en este caso) los que admiten la confidencialidad directa. Con esta directiva activada, el servidor intenta establecer una conexión muy segura antes de recurrir a los cifrados permitidos con menor seguridad.

Después de completar ambos procedimientos, guarde los cambios en `/etc/httpd/conf.d/ssl.conf` y reinicie Apache.

Si vuelves a probar el dominio en los [laboratorios SSL de Qualys](#), verás que la RC4 vulnerabilidad y otras advertencias han desaparecido y el resumen será similar al siguiente.

Calificación global	A
Certificate	100%
Compatibilidad del protocolo	100%
Intercambio de clave	90%
Seguridad del cifrado	90%

En cada actualización de OpenSSL, se introducen nuevos cifrados y se elimina la compatibilidad con los antiguos. Conserve su AL2023 instancia EC2 up-to-date, esté atento a los anuncios de seguridad de [OpenSSL](#) y manténgase alerta a los informes de nuevos ataques de seguridad en la prensa técnica.

Solución de problemas

- Mi servidor web Apache no se inicia a menos que especifique una contraseña

Es el comportamiento esperado si ha instalado una clave de servidor privado cifrada y protegida mediante contraseña.

Puede eliminar el cifrado y el requisito de contraseña de la clave. Supongamos que tiene una clave RSA cifrada privada que se denomina `custom.key` en el directorio predeterminado y que su

contraseña es **abcde12345**. Ejecute los siguientes comandos en la instancia de EC2 para generar una versión no cifrada de la clave.

```
[ec2-user ~]$ cd /etc/pki/tls/private/
[ec2-user private]$ sudo cp custom.key custom.key.bak
[ec2-user private]$ sudo openssl rsa -in custom.key -passin pass:abcde12345 -out
custom.key.nocrypt
[ec2-user private]$ sudo mv custom.key.nocrypt custom.key
[ec2-user private]$ sudo chown root:root custom.key
[ec2-user private]$ sudo chmod 600 custom.key
[ec2-user private]$ sudo systemctl restart httpd
```

Ahora, Apache debería iniciarse sin solicitarle una contraseña.

- Recibo errores cuando ejecuto `sudo dnf install -y mod_ssl`.

Al instalar los paquetes necesarios para SSL, puede que aparezcan errores similares a los siguientes:

```
Error: httpd24-tools conflicts with httpd-tools-2.2.34-1.16.amzn1.x86_64
Error: httpd24 conflicts with httpd-2.2.34-1.16.amzn1.x86_64
```

Por lo general, esto significa que la instancia EC2 no se está ejecutando. AL2023 Este tutorial solo admite instancias recién creadas a partir de una AMI de AL2023 oficial.

Tutorial: alojar un WordPress blog en AL2023

Los siguientes procedimientos le ayudarán a instalar, configurar y proteger un WordPress blog en su instancia AL2023. Este tutorial es una buena introducción al uso de Amazon EC2, ya que tiene el control total sobre el servidor web que aloja su WordPress blog, lo que no es habitual en un servicio de alojamiento tradicional.

Es responsable de actualizar los paquetes de software y de mantener los parches de seguridad del servidor. Para una WordPress instalación más automatizada que no requiera una interacción directa con la configuración del servidor web, el CloudFormation servicio proporciona una WordPress plantilla que también puede ayudarle a empezar rápidamente. Para obtener más información, consulte [Introducción](#) en la Guía del usuario de AWS CloudFormation . Si necesita una solución de alta disponibilidad con una base de datos disociada, consulte [Implementación de un WordPress sitio web de alta disponibilidad](#) en la AWS Elastic Beanstalk Guía para desarrolladores.

Important

Estos procedimientos están diseñados para usarse con. AL2023 Para obtener información acerca de otras distribuciones, consulte la documentación específica. Muchos de los pasos de este tutorial no funcionan en instancias de Ubuntu. Para obtener ayuda WordPress sobre la instalación en una instancia de Ubuntu, consulte [WordPress](#) la documentación de Ubuntu. También se puede utilizar [CodeDeploy](#) para realizar esta tarea en sistemas Amazon Linux, macOS o Unix.

Temas

- [Requisitos previos](#)
- [Instala WordPress](#)
- [Siguiendo pasos](#)
- [Ayuda Ha cambiado el nombre DNS público y ahora el blog se ha roto](#)

Requisitos previos

Le recomendamos encarecidamente que asocie una dirección IP elástica (EIP) a la instancia que utilice para alojar un WordPress blog. Esto impide que la dirección DNS pública de la instancia cambie e interrumpa la instalación. Si posee un nombre de dominio y quiere usarlo para el blog, puede actualizar el registro DNS del nombre de dominio para que apunte a la dirección EIP (para obtener ayuda al respecto, póngase en contacto con el registrador de nombres de dominio). Puede tener una dirección EIP asociada con una instancia en ejecución sin costo alguno. Para obtener más información, consulte [Direcciones IP elásticas](#) en la Guía del usuario de Amazon EC2. En el tutorial [Tutorial: Instalar un servidor LAMP en AL2023](#) también se incluyen pasos para configurar un grupo de seguridad que permita el tráfico HTTP y HTTPS, así como varios pasos para asegurar que los permisos de archivo estén establecidos de forma correcta en el servidor web. Para obtener más información sobre la adición de reglas a su grupo de seguridad, consulte [Agregar reglas a un grupo de seguridad](#).

Si todavía no tiene un nombre de dominio para el blog, puede registrar uno con Route 53 y asociarlo con la dirección EIP de la instancia. Para obtener más información, consulte [Registrar nombres de dominio mediante Amazon Route 53](#) en la Guía para desarrolladores de Amazon Route 53.

Instala WordPress

Conéctese a la instancia y descargue el paquete WordPress de instalación. Para obtener más información sobre cómo conectarse a la instancia, consulte [Conexión a instancias AL2023](#).

1. Descargue e instale estos paquetes con el siguiente comando.

```
dnf install wget php-mysqlnd httpd php-fpm php-mysqli mariadb105-server php-json  
php php-devel -y
```

2. Es posible que aparezca una advertencia con un texto similar en la salida (las versiones pueden variar con el tiempo):

```
WARNING:  
  A newer release of "Amazon Linux" is available.  
  
  Available Versions:  
  
dnf upgrade --releasever=2023.0.20230202  
  
  Release notes:  
    https://aws.amazon.com  
  
Version 2023.0.20230204:  
  Run the following command to update to 2023.0.20230204:  
  
    dnf upgrade --releasever=2023.0.20230204 ... etc
```

Como práctica recomendada, te recomendamos que mantengas el sistema operativo lo más up-to-date posible, pero quizás quieras ir iterando cada versión para asegurarte de que no haya conflictos en tu entorno. Si se produce un error en la instalación de los paquetes anteriores indicados en el paso 1, es posible que deba actualizarse a una de las versiones más recientes enumeradas y volver a intentarlo.

3. Descargue el paquete de WordPress instalación más reciente con el wget comando. El comando siguiente debería descargar siempre la última versión.

```
[ec2-user ~]$ wget https://wordpress.org/latest.tar.gz
```

4. Descomprima y desarchive el paquete de instalación. La carpeta de instalación se descomprime en una carpeta llamada wordpress.

```
[ec2-user ~]$ tar -xzf latest.tar.gz
```

Para crear una base de datos, un usuario y una base de datos para WordPress la instalación

WordPress La instalación debe almacenar información, como las entradas de blog y los comentarios de los usuarios, en una base de datos. Este procedimiento ayuda a crear una base de datos para el blog y un usuario que esté autorizado a leer y guardar información en ella.

1. Inicie la base de datos y el servidor web.

```
[ec2-user ~]$ sudo systemctl start mariadb httpd
```

2. Inicie sesión en el servidor de base de datos como el usuario `root`. Escriba la contraseña `root` de la base de datos cuando se lo pidan. Esta contraseña puede ser diferente de la contraseña `root` del sistema o incluso podría estar en blanco si no se ha protegido el servidor de bases de datos.

Si todavía no ha protegido el servidor de base de datos, es importante que lo haga. Para obtener más información, consulte [Paso 3: Proteger el servidor de base de datos](#) (AL2023).

```
[ec2-user ~]$ mysql -u root -p
```

3. Cree un usuario y una contraseña para la base de datos MySQL. WordPressLa instalación utiliza estos valores para comunicarse con la base de datos MySQL. Escriba el comando siguiente sustituyendo un nombre de usuario y contraseña únicos.

```
CREATE USER 'wordpress-user'@'localhost' IDENTIFIED BY 'your_strong_password';
```

Asegúrese de que crea una contraseña fuerte para el usuario. No utilice la comilla simple (') en la contraseña porque interrumpirá el comando anterior. No utilice ninguna contraseña existente y asegúrese de que la guarda en un lugar seguro.

4. Cree la base de datos. Póngale un nombre descriptivo y significativo, por ejemplo `wordpress-db`.

 Note

Los signos de puntuación que rodean el nombre de la base de datos en el comando siguiente son acentos graves. La tecla (`) se ubica por lo general sobre la tecla Tab en un teclado estándar. Los acentos graves no siempre son obligatorios pero permiten usar caracteres no válidos, por ejemplo, guiones, en los nombres de las bases de datos.

```
CREATE DATABASE `wordpress-db`;
```

5. Otorgue todos los privilegios de su base de datos al WordPress usuario que creó anteriormente.

```
GRANT ALL PRIVILEGES ON `wordpress-db`.* TO "wordpress-user"@"localhost";
```

6. Vacíe los privilegios de base de datos para recoger todos los cambios.

```
FLUSH PRIVILEGES;
```

7. Salga del cliente de mysql.

```
exit
```

Para crear y modificar el archivo wp-config.php

La carpeta WordPress de instalación contiene un ejemplo de archivo de configuración llamado wp-config-sample.php. En este procedimiento, va a copiar este archivo y a modificarlo para ajustarlo a su configuración específica.

1. Copie el archivo wp-config-sample.php en un archivo llamado wp-config.php. Esto crea un archivo de configuración nuevo y mantiene el archivo original de muestra intacto como copia de seguridad.

```
[ec2-user ~]$ cp wordpress/wp-config-sample.php wordpress/wp-config.php
```

2. Modifique el archivo wp-config.php con el editor de texto favorito (por ejemplo nano o vim) y escriba los valores de la instalación. Si no tiene un editor favorito, nano es más adecuado para principiantes.

```
[ec2-user ~]$ nano wordpress/wp-config.php
```

- a. Busque la línea que define DB_NAME y cambie database_name_here por el nombre de la base de datos que ha creado en [Step 4](#) de [Para crear una base de datos, un usuario y una base de datos para WordPress la instalación](#).

```
define('DB_NAME', 'wordpress-db');
```

- b. Busque la línea que define DB_USER y cambie username_here por el usuario de la base de datos que ha creado en [Step 3](#) de [Para crear una base de datos, un usuario y una base de datos para WordPress la instalación](#).

```
define('DB_USER', 'wordpress-user');
```

- c. Busque la línea que define DB_PASSWORD y cambie password_here por la contraseña fuerte que ha creado en [Step 3](#) de [Para crear una base de datos, un usuario y una base de datos para WordPress la instalación](#).

```
define('DB_PASSWORD', 'your_strong_password');
```

- d. Busque la sección denominada Authentication Unique Keys and Salts. Estos SALT valores KEY y otros proporcionan una capa de cifrado a las cookies del navegador que WordPress los usuarios almacenan en sus máquinas locales. Básicamente, agregar valores largos aleatorios hace que el sitio sea más seguro. Visite <https://api.wordpress.org/secret-key/1.1/salt/> para generar aleatoriamente un conjunto de valores clave que puede copiar y pegar en el wp-config.php archivo. Para pegar texto en un terminal PuTTY, coloque el cursor en el punto que quiere pegar el texto y haga clic con el botón derecho del ratón dentro del terminal PuTTY.

Para obtener más información sobre las claves de seguridad, visita <https://wordpress.org/support/article/editing-wp-config-php/#security-keys>.

Note

Los valores siguientes tienen una finalidad de ejemplo únicamente; no los use en la instalación.

```
define('AUTH_KEY',          ' #U$${RXN8:b^-L 0(WU_+ c+WFkI~c]o]-bHw+)/
Aj[wTwSiZ<Qb[mghEXcRh-');
define('SECURE_AUTH_KEY',   'Zsz._P=l/|y.Lq)XjlkwS1y5NJ76E6EJ.AV0pCKZZB,*~*r ?
60P$eJT@;+(ndLg');
define('LOGGED_IN_KEY',     'ju}qwre3V*+8f_z0Wf?{LlGsQ]Ye@2Jh^,8x>)Y |;(^[Iw]Pi
+LG#A4R?7N`YB3');
define('NONCE_KEY',         'P(g62HeZxEes|LnI^i=H,[XwK9I&[2s|:~0N}VJM%?;v2v]v+;
+^9eXUahg@::Cj');
define('AUTH_SALT',         'C$DpB4Hj[JK:~{q1`sRVa:{:7yShy(9A@5wg+`JJVb1fk%-
Bx*M4(qc[Qg%JT!h');
define('SECURE_AUTH_SALT',  'd!uRu#}+q#{f$Z?Z9uFPG.${+S{n~1M&%@~gL>U>NV<zpD-@2-
Es7Q10-bp28EKv');
define('LOGGED_IN_SALT',    ';j{00P*owZf)kVD+FVLn-~ >.|Y%Ug4#I^*LVd9QeZ^&XmK|
e(76miC+&W&+^0P/');
define('NONCE_SALT',       '-97r*V/cgxLmp?Zy4zUU4r99QQ_rGs2LTd%P;|
_e1tS)8_B/,.6[=UK<J_y9?JWG');
```

- e. Guarde el archivo y salga del editor de texto.

Para instalar sus WordPress archivos en la raíz de documentos de Apache

- Ahora que ha descomprimido la carpeta de instalación, ha creado una base de datos y un usuario MySQL y ha personalizado el archivo de WordPress configuración, puede copiar los archivos de instalación en la raíz de documentos del servidor web para poder ejecutar el script de instalación que complete la instalación. La ubicación de estos archivos depende de si quieres que tu WordPress blog esté disponible en la raíz real del servidor web (por ejemplo my.public.dns.amazonaws.com) o en un subdirectorio o una carpeta situada debajo de la raíz (por ejemplo, my.public.dns.amazonaws.com/blog)
- Si quieres WordPress ejecutarlo desde la raíz de tus documentos, copia el contenido del directorio de instalación de WordPress (pero no el directorio en sí) de la siguiente manera:

```
[ec2-user ~]$ cp -r wordpress/* /var/www/html/
```

- Si quieres WordPress ejecutarlo en un directorio alternativo bajo la raíz del documento, primero crea ese directorio y, a continuación, copia los archivos en él. En este ejemplo, WordPress se ejecutará desde el directorio `blog`:

```
[ec2-user ~]$ mkdir /var/www/html/blog
```

```
[ec2-user ~]$ cp -r wordpress/* /var/www/html/blog/
```

Important

Por razones de seguridad, si no pasa al siguiente procedimiento de inmediato, pare el servidor web Apache (httpd) ahora. Tras mover la instalación a la raíz de documentos de Apache, el script de WordPress instalación queda desprotegido y un atacante podría acceder a su blog si el servidor web Apache estuviera en funcionamiento. Para detener el servidor web Apache, escriba el comando `sudo service httpd stop`. Si pasa al siguiente procedimiento, no es preciso que pare el servidor web Apache.

Para permitir el uso WordPress de enlaces permanentes

WordPress Los enlaces permanentes necesitan usar `.htaccess` archivos de Apache para funcionar correctamente, pero esto no está habilitado de forma predeterminada en Amazon Linux. Use este procedimiento para permitir todas las anulaciones en la raíz de documentos de Apache.

1. Abra el archivo `httpd.conf` con el editor de textos que prefiera (como nano o vim). Si no tiene un editor favorito, nano es más adecuado para principiantes.

```
[ec2-user ~]$ sudo vim /etc/httpd/conf/httpd.conf
```

2. Busque la sección que comienza por `<Directory "/var/www/html">`.

```
<Directory "/var/www/html">
#
# Possible values for the Options directive are "None", "All",
# or any combination of:
#   Indexes Includes FollowSymLinks SymLinksifOwnerMatch ExecCGI MultiViews
#
# Note that "MultiViews" must be named *explicitly* --- "Options All"
# doesn't give it to you.
#
# The Options directive is both complicated and important. Please see
# http://httpd.apache.org/docs/2.4/mod/core.html#options
# for more information.
#
Options Indexes FollowSymLinks
```

```
#
# AllowOverride controls what directives may be placed in .htaccess files.
# It can be "All", "None", or any combination of the keywords:
#   Options FileInfo AuthConfig Limit
#
AllowOverride None

#
# Controls who can get stuff from this server.
#
Require all granted
</Directory>
```

3. Cambie la línea `AllowOverride None` de la sección anterior por `AllowOverride All`.

 Note

Hay múltiples líneas `AllowOverride` en este archivo; asegúrese de que cambia la línea de la sección `<Directory "/var/www/html">`.

```
AllowOverride All
```

4. Guarde el archivo y salga del editor de texto.

Para instalar la biblioteca de dibujos gráficos de PHP en AL2023

La biblioteca GD para PHP le permite modificar imágenes. Instale esta biblioteca si tiene que recortar la imagen de encabezado para su blog. La versión phpMyAdmin que instale puede requerir una versión mínima específica de esta biblioteca (por ejemplo, la versión 8.1).

Utilice el siguiente comando para instalar la biblioteca de dibujos gráficos de PHP AL2023. Por ejemplo, si instaló php8.1 desde el origen como parte de la instalación de la pila LAMP, este comando instala la versión 8.1 de la biblioteca de dibujo de gráficos de PHP.

```
[ec2-user ~]$ sudo dnf install php-gd
```

Para verificar la versión instalada, utilice el siguiente comando:

```
[ec2-user ~]$ sudo dnf list installed | grep php-gd
```


A continuación, se muestra un ejemplo del resultado:

```
php-gd.x86_64                8.1.30-1.amzn2                @amazonlinux
```

Para instalar la biblioteca de dibujo de gráficos de PHP en la Amazon Linux AMI

La biblioteca GD para PHP le permite modificar imágenes. Instale esta biblioteca si tiene que recortar la imagen de encabezado para su blog. La versión phpMyAdmin que instale puede requerir una versión mínima específica de esta biblioteca (por ejemplo, la versión 8.1).

Para comprobar qué versiones están disponibles, utilice el siguiente comando:

```
[ec2-user ~]$ dnf list | grep php
```

A continuación, se incluyen líneas de ejemplo de la salida de la biblioteca de dibujo de gráficos de PHP (versión 8.1):

```
php8.1.aarch64                8.1.7-1.amzn2023.0.1
                                @amazonlinux
php8.1-cli.aarch64            8.1.7-1.amzn2023.0.1
                                @amazonlinux
php8.1-common.aarch64         8.1.7-1.amzn2023.0.1
                                @amazonlinux
php8.1-devel.aarch64          8.1.7-1.amzn2023.0.1
                                @amazonlinux
php8.1-fpm.aarch64            8.1.7-1.amzn2023.0.1
                                @amazonlinux
php8.1-gd.aarch64             8.1.7-1.amzn2023.0.1
                                @amazonlinux
```

Utilice el siguiente comando para instalar una versión específica de la biblioteca de dibujo de gráficos de PHP (por ejemplo, la versión php8.1) en la AMI de Amazon Linux:

```
[ec2-user ~]$ sudo dnf install -y php8.1-gd
```

Para ajustar los permisos de archivo para el servidor web Apache

Algunas de las funciones disponibles WordPress requieren acceso de escritura a la raíz del documento de Apache (por ejemplo, cargar contenido multimedia a través de las pantallas de administración). Si todavía no lo ha hecho, aplique los siguientes permisos y suscripciones de grupo (como se describe detalladamente en el [tutorial del servidor web LAMP](#)).

1. Otorgue la propiedad de archivos de `/var/www` y su contenido al usuario `apache`.

```
[ec2-user ~]$ sudo chown -R apache /var/www
```

2. Otorgue la propiedad de grupo de `/var/www` y su contenido al grupo `apache`.

```
[ec2-user ~]$ sudo chgrp -R apache /var/www
```

3. Cambie los permisos del directorio `/var/www` y sus subdirectorios para agregar permisos de escritura de grupo y establecer el ID de grupo en futuros subdirectorios.

```
[ec2-user ~]$ sudo chmod 2775 /var/www  
[ec2-user ~]$ find /var/www -type d -exec sudo chmod 2775 {} \;
```

4. Cambie de forma recursiva los permisos de archivo de `/var/www` y sus subdirectorios.

```
[ec2-user ~]$ find /var/www -type f -exec sudo chmod 0644 {} \;
```

Note

Si pretendes utilizarlo también WordPress como servidor FTP, aquí necesitarás una configuración de grupo más permisiva. Para ello, revise los [pasos recomendados y la configuración WordPress de seguridad](#).

5. Reinicie el servidor web Apache para recoger el grupo y los permisos nuevos.

```
[ec2-user ~]$ sudo systemctl restart httpd
```

Para ejecutar el script WordPress de instalación con AL2023

Ya está listo para la instalación WordPress. Los comandos que utilice dependen del sistema operativo. Los comandos de este procedimiento se utilizan con AL2023. Utilice el procedimiento que sigue a este con AL2023 AMI.

1. Use el comando `systemctl` para asegurarse de que se inician los servicios `httpd` y de base de datos cada vez que se arranca el sistema.

```
[ec2-user ~]$ sudo systemctl enable httpd && sudo systemctl enable mariadb
```

2. Verifique que el servidor de base de datos se está ejecutando.

```
[ec2-user ~]$ sudo systemctl status mariadb
```

Si el servicio de base de datos no se está ejecutando, inícielo.

```
[ec2-user ~]$ sudo systemctl start mariadb
```

3. Verifique que el servidor web Apache (httpd) se está ejecutando.

```
[ec2-user ~]$ sudo systemctl status httpd
```

Si el servicio httpd no se está ejecutando, inícielo.

```
[ec2-user ~]$ sudo systemctl start httpd
```

4. En un navegador web, escriba la URL de su WordPress blog (la dirección DNS pública de la instancia o la dirección seguida de la blog carpeta). Deberías ver el script WordPress de instalación. Proporcione la información requerida para la WordPress instalación. Seleccione Instalar WordPress para completar la instalación. Para obtener más información, consulte el [paso 5: ejecutar el script de instalación](#) en el WordPress sitio web.

Para ejecutar el script WordPress de instalación con AL2023 AMI

1. Use el comando chkconfig para asegurarse de que se inician los servicios httpd y de base de datos cada vez que se arranca el sistema.

```
[ec2-user ~]$ sudo chkconfig httpd on && sudo chkconfig mariadb on
```

2. Verifique que el servidor de base de datos se está ejecutando.

```
[ec2-user ~]$ sudo service mariadb status
```

Si el servicio de base de datos no se está ejecutando, inícielo.

```
[ec2-user ~]$ sudo service mariadb start
```

3. Verifique que el servidor web Apache (httpd) se está ejecutando.

```
[ec2-user ~]$ sudo service httpd status
```

Si el servicio `httpd` no se está ejecutando, inícielo.

```
[ec2-user ~]$ sudo service httpd start
```

4. En un navegador web, escriba la URL de su WordPress blog (la dirección DNS pública de la instancia o la dirección seguida de la `blog` carpeta). Deberías ver el script WordPress de instalación. Proporcione la información requerida para la WordPress instalación. Seleccione Instalar WordPress para completar la instalación. Para obtener más información, consulte el [paso 5: ejecutar el script de instalación](#) en el WordPress sitio web.

Siguientes pasos

Una vez que haya probado su WordPress blog, considere la posibilidad de actualizar su configuración.

Uso de un nombre de dominio personalizado

Si tiene un nombre de dominio asociado con la dirección EIP de la instancia de EC2, puede configurar el blog para que use ese nombre en lugar de la dirección DNS pública de EC2. Para obtener más información, consulte [Cambiar la URL del sitio](#) en el WordPress sitio web.

Configuración del blog

Puede configurar el blog para usar distintos [temas](#) y [complementos](#) y ofrecer una experiencia más personalizada a los lectores. Sin embargo, en ocasiones el proceso de instalación puede producir un efecto indeseado y provocar la pérdida del blog completo. Recomendamos encarecidamente que cree una copia de seguridad de Amazon Machine Image (AMI) de la instancia antes de instalar temas o complementos, de forma que pueda restaurar el blog si algo sale mal durante la instalación. Para obtener más información, consulte [Creación de una AMI](#) en la Guía del usuario de Amazon EC2.

Aumentar la capacidad

Si tu WordPress blog se hace popular y necesitas más capacidad de procesamiento o almacenamiento, considera los siguientes pasos:

- Ampliar el espacio de almacenamiento de la instancia. Para obtener más información, consulte [Volúmenes elásticos de Amazon EBS](#).
- Mover la base de datos MySQL a [Amazon RDS](#) para aprovechar la capacidad de fácil escala del servicio.

Mejore el rendimiento de la red de su tráfico de Internet

Si espera que su blog impulse el tráfico a partir de los usuarios ubicados en todo el mundo, considere el uso de [AWS Global Accelerator](#). Global Accelerator le ayuda a reducir la latencia al mejorar el rendimiento del tráfico de Internet entre los dispositivos cliente de sus usuarios y la WordPress aplicación en la que se ejecuta AWS. Global Accelerator utiliza la [red AWS global](#) para dirigir el tráfico a un punto final de aplicación en buen estado en la AWS región más cercana al cliente.

Obtenga más información sobre WordPress

Los siguientes enlaces contienen más información sobre WordPress.

- Para obtener más información al respecto WordPress, consulte la documentación de ayuda del WordPress Codex en [Codex](#).
- Para obtener más información acerca de la solución de problemas de la instalación, consulte [Problemas de instalación comunes](#).
- [Para obtener información sobre cómo hacer que su WordPress blog sea más seguro, vaya a Hardening. WordPress](#)
- Para obtener información sobre cómo mantener tu WordPress blog up-to-date, consulta [Actualización WordPress](#).

Ayuda Ha cambiado el nombre DNS público y ahora el blog se ha roto

WordPress La instalación se configura automáticamente con la dirección DNS pública de la instancia EC2. Si detiene y reinicia la instancia, la dirección DNS pública cambia (salvo que está asociada a una dirección IP elástica) y el blog no funcionará porque hace referencia a los recursos de una dirección que ya no existe (o que se ha asignado a otra instancia de EC2). En <https://wordpress.org/support/article/changing-the-site-url/> encontrará una descripción más detallada del problema y varias posibles soluciones.

Si esto le ha ocurrido a su WordPress instalación, es posible que pueda recuperar su blog con el siguiente procedimiento, que utiliza la interfaz de línea de wp-cli comandos para WordPress.

Para cambiar la URL de su WordPress sitio por la wp-cli

1. Conéctese a la instancia de EC2 con SSH.
2. Anote la URL del sitio anterior y la URL del sitio nuevo para la instancia. Es probable que la antigua URL del sitio sea el nombre de DNS público de la instancia de EC2 cuando la instaló WordPress. La URL del sitio nuevo es el nombre DNS público actual de la instancia de EC2. Si no está seguro de la URL del sitio anterior, puede usar curl para buscarla con el comando siguiente.

```
[ec2-user ~]$ curl localhost | grep wp-content
```

Debería ver referencias al nombre DNS público anterior en el resultado, que tendrá un aspecto similar a lo siguiente (la URL del sitio anterior en rojo):

```
<script type='text/javascript' src='http://ec2-52-8-139-223.us-west-1.compute.amazonaws.com/wp-content/themes/twentyfifteen/js/functions.js?ver=20150330'></script>
```

3. Descargue el wp-cli con el comando siguiente.

```
[ec2-user ~]$ curl -O https://raw.githubusercontent.com/wp-cli/builds/gh-pages/phar/wp-cli.phar
```

4. Busque y sustituya la antigua URL del sitio en la WordPress instalación con el siguiente comando. Sustituya la dirección URL antigua y la nueva del sitio por la instancia de EC2 y la ruta de acceso a la WordPress instalación (normalmente, /var/www/html o /var/www/html/blog).

```
[ec2-user ~]$ php wp-cli.phar search-replace 'old_site_url' 'new_site_url' --path=/path/to/wordpress/installation --skip-columns=guid
```

5. En un navegador web, introduzca la nueva URL del sitio de su WordPress blog para comprobar que el sitio vuelve a funcionar correctamente. Si no es así, consulte [Cambiar la URL del sitio](#) y [Problemas comunes de instalación](#) para obtener más información.

Tutorial: Transición de Redis 6 a Valkey en 2023 AL2

La siguiente documentación describe los aspectos clave de la transición de Redis 6 a Valkey on 023. AL2

Cronología de soporte para Redis 6

Redis 6 llega al final de su vida útil (EOL) el 31 de enero de 2027. Después de esta fecha, Redis 6 ya no recibirá actualizaciones ni parches de seguridad del proyecto Redis. Recomendamos encarecidamente a los usuarios que migren a Valkey antes de enero de 2027 para garantizar el soporte y las actualizaciones de seguridad continuos.

[Para obtener más información sobre los plazos de soporte de las versiones de Redis, consulte la documentación de Redis Schedule. End-Of-Life](#)

Introducción a Valkey

Valkey es una versión de código abierto de Redis 7, mantenida por The Linux Foundation. Es totalmente compatible con las versiones 2.x a 7.2.x del software de código abierto (OSS) de Redis. Valkey mantiene la API y la funcionalidad familiares de Redis, al tiempo que ofrece varias mejoras:

- Mejora de rendimiento mediante subprocesos múltiples.
- Mejora de la eficiencia de la memoria, especialmente en modo clúster.
- Replicación de doble canal para una mejor coherencia de los datos.

Plan de migración y calendario

Se recomienda encarecidamente a los usuarios que migren de Redis 6 a Valkey antes del 31 de enero de 2027, cuando Redis 6 llegue al final de su vida útil (EOL). Esta migración requiere una intervención manual y no es automática.

Amazon Linux recomienda esta migración para garantizar la funcionalidad, el soporte y las actualizaciones de seguridad continuos para las aplicaciones que dependen de Redis.

Opciones y pasos de migración

Proponemos tres rutas de migración a Valkey en función de sus requisitos de despliegue y sus necesidades operativas.

Opción 1: instalación de una nueva instancia

Para nuevas implementaciones o cuando no sea necesaria la migración de datos:

1. Instale Valkey:

```
[ec2-user ~]$ sudo dnf install valkey
```

2. Inicie Valkey:

```
[ec2-user ~]$ sudo systemctl start valkey
```

3. (Opcional) Habilite Valkey al arrancar:

```
[ec2-user ~]$ sudo systemctl enable valkey
```

4. Verificar la instalación:

```
[ec2-user ~]$ valkey-cli info server  
[ec2-user ~]$ valkey-cli ping
```

Opción 2: reemplazo in situ

Para las instancias existentes en las que no se requiere la persistencia de los datos:

1. Detenga Redis 6:

```
[ec2-user ~]$ sudo systemctl stop redis6
```

2. Instale Valkey:

```
[ec2-user ~]$ sudo dnf install valkey
```

3. (Opcional) Utilice la configuración de Redis 6 en Valkey:

```
[ec2-user ~]$ sudo cp /etc/redis6/redis6.conf /etc/valkey/valkey.conf  
[ec2-user ~]$ sudo cp /etc/valkey/valkey.conf /etc/valkey/valkey.conf.backup  
[ec2-user ~]$ sudo chown valkey:root /etc/valkey/valkey.conf  
[ec2-user ~]$ sudo sed -i 's|^dir\s.*|dir /var/lib/valkey|g' /etc/valkey/  
valkey.conf
```


4. (Opcional) Utilice el archivo de configuración Sentinel de Redis 6 en Valkey:

```
[ec2-user ~]$ sudo cp /etc/redis6/sentinel.conf /etc/valkey/sentinel.conf
[ec2-user ~]$ sudo chown valkey:root /etc/valkey/sentinel.conf
```

5. Inicie Valkey:

```
[ec2-user ~]$ sudo systemctl start valkey
```

6. (Opcional) Habilite Valkey al arrancar:

```
[ec2-user ~]$ sudo systemctl enable valkey
```

7. Verifique la instalación de Valkey:

```
[ec2-user ~]$ valkey-cli info server
[ec2-user ~]$ valkey-cli ping
```

8. Elimine Redis 6:

```
[ec2-user ~]$ sudo dnf remove redis6
```

Opción 3: migración de datos

Esta opción le permite ejecutar Redis 6 y Valkey simultáneamente.

1. Instale Valkey sin eliminar Redis 6:

```
[ec2-user ~]$ sudo dnf install valkey
```

2. (Opcional) Utilice la configuración de Redis 6 en Valkey:

```
[ec2-user ~]$ sudo cp /etc/redis6/redis6.conf /etc/valkey/valkey.conf
[ec2-user ~]$ sudo cp /etc/valkey/valkey.conf /etc/valkey/valkey.conf.backup
[ec2-user ~]$ sudo chown valkey:root /etc/valkey/valkey.conf
[ec2-user ~]$ sudo sed -i 's|^dir\s.*|dir /var/lib/valkey|g' /etc/valkey/
valkey.conf
```

3. (Opcional) Utilice el archivo de configuración Sentinel de Redis 6 en Valkey:

```
[ec2-user ~]$ sudo cp /etc/redis6/sentinel.conf /etc/valkey/sentinel.conf
```

```
[ec2-user ~]$ sudo chown valkey:root /etc/valkey/sentinel.conf
```

4. Modifique la configuración de Valkey:

Edite `/etc/valkey/valkey.conf` y establezca la directiva “port” en un valor diferente (por ejemplo, 6380) para evitar conflictos con Redis 6.

5. Inicie Valkey:

```
[ec2-user ~]$ sudo systemctl start valkey
```

6. (Opcional) Habilite Valkey al arrancar:

```
[ec2-user ~]$ sudo systemctl enable valkey
```

7. Verifique la instalación de Valkey:

```
[ec2-user ~]$ valkey-cli -p port info server  
[ec2-user ~]$ valkey-cli -p port ping
```

 Note

Sustitúyalo por el número de **port** puerto configurado.

8. Migre los datos:

Ahora puede migrar datos de Redis 6 a Valkey mediante métodos de replicación o transferencia manual de datos.

9. Actualice las configuraciones de aplicaciones existentes

Actualice gradualmente sus aplicaciones para usar el puerto de Valkey.

10. Elimine Redis 6:

Una vez que se hayan migrado todos los datos y las aplicaciones, podrá detener y eliminar Redis 6.

```
[ec2-user ~]$ sudo systemctl stop redis6  
[ec2-user ~]$ sudo dnf remove redis6
```

 Note

Se recomienda encarecidamente validar el proceso de migración en un entorno de prueba antes de implementar cambios en los sistemas de producción.

Temas relacionados

Para obtener más información sobre Valkey:

- Valkey: <https://valkey.io/>
- Migración de Valkey: <https://valkey.io/topics/migration/>

Tutorial: Instalar el entorno de escritorio GNOME en AL2023

El [entorno de escritorio GNOME](#) está disponible como interfaz gráfica de usuario opcional a partir de la versión 2023.7 o posterior. AL2023

Los siguientes procedimientos le ayudan a instalar el entorno de escritorio GNOME en su instancia. AL2023 Puede utilizar esta interfaz gráfica para interactuar con su sistema Linux mediante un entorno de escritorio conocido en lugar de utilizar únicamente la interfaz de línea de comandos.

Contenido

- [Requisitos previos](#)
- [Instalación](#)
- [Temas relacionados](#)

Requisitos previos

- El entorno de escritorio requiere al menos 2,4 GB de memoria. Por lo tanto, se recomienda una instancia del tipo `t2.medium` o superior para garantizar un rendimiento adecuado. Algunos ejemplos de tipos de instancias con memoria insuficiente son `t2.nano`, `t2.micro` y `t2.small`. Esta restricción también se aplica a `t3` y a las `t4` instancias de este tamaño, así como a cualquier otro tipo de instancia que no cumpla con los requisitos de memoria.

- En este tutorial se supone que ya ha lanzado una instancia con la versión 2023.7 o posterior en AL2023 ejecución. Para obtener más información, consulte las páginas [AL2023 en Amazon EC2](#) y [Actualización AL2023](#).

Instalación

- Instale el entorno de escritorio GNOME y los paquetes relacionados.

```
[ec2-user ~]$ sudo dnf groupinstall "Desktop" -y
```

Note

Para acceder al entorno gráfico de escritorio, necesitará instalar y configurar software adicional, como Amazon DCV o VNC. Estas herramientas le permiten conectarse a la interfaz gráfica de usuario e interactuar con ella a través de la red.

Temas relacionados

Para obtener más información sobre el entorno de escritorio gráfico, consulte la siguiente documentación:

- [¿Qué es Amazon DCV?](#) en la Guía del administrador de Amazon DCV
- [Tutorial: Configurar el servidor TigerVNC en AL2023](#)

Tutorial: Configurar el servidor TigerVNC en AL2023

Los siguientes procedimientos le ayudan a configurar el servidor VNC en su AL2023 instancia. VNC le permite acceder de forma remota al entorno gráfico de escritorio e interactuar con él a través de una conexión de red segura.

Contenido

- [Requisitos previos](#)
- [Paso 1: instalación](#)
- [Paso 2: configuración](#)
- [Paso 3: conectarse mediante un cliente de VNC](#)

- [\(Opcional\) Inicie el servicio al arrancar](#)
- [\(Opcional\) Desactive la pantalla de bloqueo inactiva](#)
- [Temas relacionados](#)

Requisitos previos

- En este tutorial se da por sentado que ya ha instalado el entorno de escritorio GNOME en la instancia AL2023 . Para obtener más información, consulte la página [Tutorial: Instalar el entorno de escritorio GNOME en AL2023](#).
- Este tutorial utiliza el reenvío de puertos SSH para acceder al servidor VNC. Para obtener más información sobre cómo configurar su par de claves, consulte [Conexión a la instancia de Linux mediante un cliente SSH](#) en la Guía del usuario de Amazon EC2.
- El siguiente procedimiento no lo guía a lo largo del proceso de instalación de un cliente de VNC. Debe tener un cliente de VNC instalado en su máquina local para poder conectarse e interactuar con el entorno de escritorio.

Paso 1: instalación

1. Conecte con la instancia . Para obtener más información, consulte [Conexión a instancias AL2023](#).
2. Instale el paquete de servidor TigerVNC para. AL2023

La opción `-y` instala el paquete sin necesidad de confirmación. Si le gustaría examinar el paquete antes de la instalación, puede omitir esta opción.

```
[ec2-user ~]$ sudo dnf install -y tigervnc-server
```

Paso 2: configuración

1. Asegúrese de que el usuario haya configurado una contraseña de VNC.

```
[ec2-user ~]$ vncpasswd
```

2. Asigne un número de pantalla al usuario.

```
[ec2-user ~]$ sudo vi /etc/tigervnc/vncserver.users
```

Agregue la siguiente configuración:

```
:1=ec2-user
```

 Note

Puede asignar cualquier número de pantalla al usuario. Para este ejemplo, utilizaremos la pantalla :1.

3. Edite el archivo de configuración del servidor VNC.

```
[ec2-user ~]$ sudo vi /etc/tigervnc/vncserver-config-defaults
```

Agregue la siguiente configuración:

```
session=gnome
securitytypes=vncauth,tlsvnc
geometry=1920x1080
localhost
alwaysshared
```

 Note

Puede cambiar la resolución de la pantalla mediante el parámetro `geometry`. Para este ejemplo, utilizaremos 1920x1080.

4. Inicie el servidor VNC. Este proceso debe repetirse cada vez que reinicie la instancia. Si desea automatizar el proceso de inicio de este servicio, consulte la sección opcional que aparece a continuación.

```
[ec2-user ~]$ sudo systemctl start vncserver@:1
```

 Important

Al iniciar el servicio `vncserver`, la parte posterior a `@` debe coincidir con el número de pantalla establecido para el usuario en el archivo `/etc/tigervnc/vncserver.users`.

Tras realizar este paso, puede crear el túnel SSH desde su máquina local y conectarse mediante su cliente VNC.

Paso 3: conectarse mediante un cliente de VNC

El servidor VNC expone un conector TCP para las conexiones de los clientes. Si bien puede exponer el puerto VNC directamente a través de su grupo de seguridad, en este tutorial se muestra cómo utilizar los túneles SSH como un enfoque más seguro al cifrar la conexión entre su máquina local y la instancia de EC2. Una vez conectado a través del túnel, se autenticará en el servidor VNC con la contraseña que configuró en el paso anterior. Para obtener más información sobre los grupos de seguridad, consulte [Grupos de seguridad de Amazon EC2 para su instancia de EC2](#) en la Guía del usuario de Amazon EC2.

1. Cree un túnel de SSH desde su máquina local.

```
$ ssh -i <keypair> -L 5901:localhost:5901 ec2-user@<address>
```

 Note

Sustituya `<keypair>` por la ruta a su clave SSH y `<address>` por la IP pública o el nombre DNS de la instancia. El puerto cambia en función del número de pantalla que se utilizó para iniciar `vncserver`. Por ejemplo, la pantalla `:1` usa el puerto 5901, la pantalla `:2` usa el puerto 5902, etc.

2. Utilice su cliente de VNC para conectarse a `localhost:5901` o `127.0.0.1:5901` con la contraseña de VNC previamente establecida.

 Important

Mantenga abierto el túnel SSH mientras usa VNC. Si el túnel SSH no está abierto, no podrá usar su cliente VNC para ver e interactuar con el entorno de escritorio.

(Opcional) Inicie el servicio al arrancar

Si planea usar VNC con regularidad, puede configurar el servidor VNC para que se inicie automáticamente cuando se inicie la instancia. Esto elimina la necesidad de iniciar manualmente el servidor VNC cada vez que reinicie la instancia. Esta configuración garantiza que el entorno gráfico de escritorio esté listo y accesible en cuanto la instancia complete su proceso de inicio.

- Configure el servicio para que se inicie al arrancar.

```
[ec2-user ~]$ sudo systemctl enable vncserver@:1
```

 Important

Al habilitar el servicio `vncserver`, la parte posterior a `@` debe coincidir con el número de pantalla establecido para el usuario en el archivo `/etc/tigervnc/vncserver.users`. Además, puede pasar el argumento `--now` después de `enable` para iniciar el servicio inmediatamente.

Tras realizar este paso, ya no tendrá que iniciar `vncserver` cada vez que reinicie la instancia.

(Opcional) Desactive la pantalla de bloqueo inactiva

- Establezca el retardo de inactividad en cero para deshabilitar la pantalla de bloqueo cuando el usuario haya estado inactivo durante un período de tiempo prolongado.

```
[ec2-user ~]$ gsettings set org.gnome.desktop.session idle-delay 0
```


Temas relacionados

Para obtener más información sobre el entorno de escritorio gráfico, consulte la siguiente documentación:

- [Tutorial: Instalar el entorno de escritorio GNOME en AL2023](#)
- [¿Qué es Amazon DCV?](#) en la Guía del administrador de Amazon DCV

Uso de LRU multigeneración (MGLRU) en los núcleos AL2023

[Multi-Gen LRU](#) es un algoritmo moderno de recuperación de páginas del kernel de Linux, diseñado para mejorar el rendimiento de la administración de la memoria en situaciones de presión de memoria. Sustituye al mecanismo tradicional LRU (utilizado menos recientemente) que se utiliza para determinar qué páginas de memoria recuperar cuando el sistema se queda sin memoria.

El mecanismo LRU tradicional utiliza un modelo de dos listas (activa e inactiva) para realizar un seguimiento del uso de las páginas, lo que puede resultar ineficiente en las cargas de trabajo modernas con conjuntos de trabajo grandes. MGLRU lo sustituye por varias “generaciones” de páginas, lo que permite al núcleo tomar decisiones más inteligentes basándose en información detallada sobre la antigüedad.

Los beneficios de MGLRU incluyen:

- Mejores decisiones de recuperación: identificación más precisa de las páginas frías (no utilizadas).
- Menor latencia y mayor rendimiento: especialmente para cargas de trabajo con grandes espacios de direcciones o muchos procesos simultáneos.
- Mejor retención de la memoria caché: las páginas que se utilizan recientemente tienen menos probabilidades de ser desalojadas prematuramente.
- Diseño escalable y eficiente en cuanto al bloqueo: funciona mejor en máquinas con muchos dispositivos. CPUs

Configuración y ajuste

La configuración del núcleo `CONFIG_LRU_GEN` está habilitada en AL2023 los núcleos. Esto se compila en MGLRU pero no lo habilita de forma predeterminada.

MGLRU se puede activar y ajustar mediante el archivo `/sys/kernel/mm/lru_gen/enabled`. El valor es una máscara de bits. Se recomienda activar todos los componentes, a menos que algunos de ellos tengan efectos secundarios no deseados.

Bit	Componentes
0	El interruptor principal de Multi-gen LRU.
1	Borrar el bit al que se ha accedido en las entradas de la tabla de hojas en lotes grandes, cuando la MMU lo configura (por ejemplo, en x86). En teoría, este comportamiento puede empeorar la contención de bloqueos (<code>mmap_lock</code>). Si está desactivado, Multi-gen LRU sufrirá una ligera degradación del rendimiento para las cargas de trabajo que asignan de forma contigua páginas activas, cuyos bits accedidos pueden borrarse con menos lotes más grandes.
2	Borrar también el bit de acceso en las entradas de la tabla de páginas que no son hojas, cuando la MMU lo establece (por ejemplo, en x86). Este comportamiento no se verificó en modelos x86 distintos de Intel y AMD. Si está desactivado, Multi-gen LRU sufrirá una degradación de rendimiento insignificante.
[yYnN]	Enable/disable todos los componentes anteriores.

Un ejemplo de cómo se puede activar MGLRU:

```
[ec2-user ~]$ echo y >/sys/kernel/mm/lru_gen/enabled
```

Esto habilita todos los componentes:

```
[ec2-user ~]$ cat /sys/kernel/mm/lru_gen/enabled  
0x0007
```

Tutorial: Instalar LibreOffice en AL2023

LibreOffice es una suite de productividad ofimática gratuita y de código abierto que incluye aplicaciones para procesamiento de textos, hojas de cálculo, presentaciones, diagramas, bases de datos y edición de fórmulas. Puede instalarlas LibreOffice en instancias de Amazon Linux 2023 (AL2023) para ofrecer a los usuarios una suite ofimática de escritorio con todas las funciones.

LibreOffice está oficialmente probado y respaldado AL2023 por The Document Foundation. Para obtener más información, consulte el [progreso de los LibreOffice 64 bits y la compatibilidad con Amazon Linux 2023](#).

Contenido

- [Requisitos previos](#)
- [Paso 1: Determine la arquitectura del sistema](#)
- [Paso 2: Descargar el paquete LibreOffice RPM](#)
- [Paso 3: Extrae el archivo](#)
- [Paso 4: Instale los paquetes RPM](#)
- [Paso 5: Verificar la instalación](#)
- [Lanzamiento LibreOffice](#)
- [Resolución de problemas](#)

Requisitos previos

- Una instancia de Amazon EC2 que se ejecuta AL2023 con un entorno de escritorio gráfico GNOME configurado y accesible a través de un protocolo de escritorio remoto (por ejemplo, Amazon DCV). Para obtener más información, consulte los siguientes temas:
 - [Tutorial: Instalar el entorno de escritorio GNOME en AL2023](#)
 - [Configuración del servidor Amazon DCV](#)
- sudo acceso root a la instancia.
- Al menos 1,5 GB de espacio libre en disco para la instalación.

- Una conexión a Internet activa para descargar el LibreOffice paquete o el paquete predescargado y transferido a la instancia.

Paso 1: Determine la arquitectura del sistema

Ejecute el siguiente comando para identificar la arquitectura de su procesador. Esto determina qué paquete RPM se debe descargar.

```
uname -p
```

El resultado es uno de los siguientes:

Output	Arquitectura	Tipo de paquete RPM
x86_64	Intel/AMD de 64 bits	Linux_x86-64_rpm
aarch64	ARM de 64 bits (Graviton)	Linux_aarch64_rpm

Paso 2: Descargar el paquete LibreOffice RPM

Para descargar el paquete LibreOffice RPM

1. Abra la [página de LibreOffice descarga](#).
2. Seleccione Linux (64 bits) (rpm) como sistema operativo.
3. Elija la versión adecuada para su arquitectura (x86_64 o aarch64).
4. Descarga el archivo a tu instancia .tar.gz. Puede usarlo wget o curl descargarlo directamente desde el terminal.

```
cd ~/Downloads
wget https://download.documentfoundation.org/libreoffice/stable/VERSION/
rpm/ARCHITECTURE/LibreOffice_VERSION_Linux_ARCHITECTURE.rpm.tar.gz
```

Sustituya *VERSION* y *ARCHITECTURE* por el número de versión y la arquitectura que desee instalar (por ejemplo, 25.2.5 y x86_64).

 Note

[Para encontrar el número de versión estable más reciente, visita la página de descargas. LibreOffice](#)

Paso 3: Extrae el archivo

Navegue hasta el directorio donde descargó el archivo y extraígallo.

```
cd ~/Downloads
tar zxvf LibreOffice_VERSION_Linux_ARCHITECTURE_rpm.tar.gz
```

Esto crea un directorio con un nombre similar
a LibreOffice_**VERSION**_Linux_**ARCHITECTURE**_rpm/.

Paso 4: Instale los paquetes RPM

Para instalar los paquetes RPM

1. Cambie al RPMS/ subdirectorio dentro de la carpeta extraída.

```
cd LibreOffice_VERSION_Linux_ARCHITECTURE_rpm/RPMS/
```

2. Instale todos los paquetes RPM utilizando `sudo rpm`.

```
sudo rpm -i *.rpm
```

Como alternativa, puede utilizar `dnf` para la resolución de dependencias:

```
sudo dnf install -y *.rpm
```

Paso 5: Verificar la instalación

Confirme que LibreOffice se instaló correctamente comprobando la versión.

```
libreoffice --version
```

Resultado previsto:

```
LibreOffice VERSION ...
```

Lanzamiento LibreOffice

Desde la terminal

Use el siguiente comando:

```
libreoffice
```

Si el `libreoffice` comando no está disponible, es posible que el binario esté versionado. En su lugar, puede lanzarlo con el nombre versionado:

```
libreofficeMAJOR_VERSION
```

Por ejemplo, `libreoffice25.2`.

Para lanzar una aplicación específica directamente:

```
libreoffice --writer      # Word processor
libreoffice --calc        # Spreadsheet
libreoffice --impress     # Presentations
libreoffice --draw        # Diagrams
libreoffice --base        # Database
libreoffice --math        # Formula editor
```

Desde el escritorio de GNOME

Tras la instalación, LibreOffice las aplicaciones aparecen en el menú de actividades de GNOME. Elija Actividades en la esquina superior izquierda, busque "LibreOffice«y seleccione la aplicación deseada.

Resolución de problemas

La siguiente sección puede ayudarle a solucionar problemas comunes al instalarla o LibreOffice ejecutarla. AL2023

Advertencia del entorno de ejecución de Java

Algunas LibreOffice funciones (como Base y determinadas extensiones) requieren un entorno de ejecución de Java (JRE). Si aparece una advertencia sobre la falta de Java, instálelo:

```
sudo dnf install -y java-17-amazon-corretto-headless
```

A continuación LibreOffice, vaya a Herramientas > Opciones LibreOffice> Avanzadas y compruebe que se ha detectado el JRE.

LibreOffice comando no encontrado

Si el `libreoffice` comando no se encuentra después de la instalación, es posible que el binario esté versionado. Intente ejecutar:

```
libreofficeMAJOR_VERSION
```

Por ejemplo, `libreoffice25.2`. También puedes localizar el binario:

```
find /opt -name "soffice" 2>/dev/null
```

Recursos adicionales

Si encuentra otros problemas, consulte la documentación oficial LibreOffice de instalación de Linux:

- [LibreOfficeGuía de instalación de Linux](#)
- [LibreOfficePágina de descarga](#)

Uso de Amazon Linux 2023 fuera de Amazon EC2

Las imágenes del contenedor de Amazon Linux 2023 se pueden ejecutar en entornos en tiempo de ejecución de contenedores compatibles. Para obtener más información sobre cómo usar Amazon Linux 2023 dentro de un contenedor, consulte [AL2023 en contenedores](#).

Amazon Linux 2023 (AL2023) también se puede ejecutar como un invitado virtualizado, además de ejecutarse directamente en Amazon EC2. Actualmente hay imágenes KVM (qcow2), VMware (OVA) y Hyper-V (vhdx) disponibles.

Note

La configuración de las imágenes de Amazon Linux 2023 es diferente a la de Amazon Linux 2.

Si viene de [ejecutar Amazon Linux 2 como máquina virtual en las instalaciones](#), necesitará adaptar la configuración para que sea compatible con AL2023.

Descarga de las imágenes de Amazon Linux 2023 para utilizarlas con KVM, VMware y Hyper-V

Las imágenes de disco de Amazon Linux 2023 para su uso con KVM, VMware y Hyper-V se pueden descargar desde cdn.amazonlinux.com.

Configuraciones compatibles de Amazon Linux 2023 para su uso en entornos virtualizados que no son de Amazon EC2

En esta sección se describen los requisitos para ejecutar Amazon Linux 2023 en entornos virtualizados que no son de Amazon EC2, como KVM, VMware o Hyper-V.

Los [Requisitos del sistema de AL2023](#) básicos se aplican a todos los entornos virtualizados que no son de Amazon EC2. En los temas siguientes se detalla una lista de los modelos de dispositivos compatibles para cada entorno de hipervisor.

Tanto KVM como VMware y Hyper-V ofrecen muchas opciones de configuración, por lo que es necesario tener cuidado al configurarlas de acuerdo con sus necesidades de seguridad, rendimiento y fiabilidad. Para obtener más información, consulte la documentación de su hipervisor.

Temas

- [Requisitos para funcionar AL2023 en KVM](#)
- [Requisitos para ejecutar AL2023 en VMware](#)
- [Requisitos para ejecutar Amazon Linux 2023 en Hyper-V](#)

Requisitos para funcionar AL2023 en KVM

En esta sección se describen los requisitos para funcionar AL2023 en KVM. Las imágenes KVM de AL2023 están disponibles para ambas `aarch64` arquitecturas `x86-64`. Estos requisitos son adicionales a los [Requisitos del sistema de AL2023](#) básicos para imágenes KVM.

Temas

- [Requisitos de host KVM para funcionar en KVM AL2023](#)
- [Soporte de dispositivo para un KVM AL2023](#)
- [Modo de arranque \(UEFI/BIOS\) compatibilidad con un AL2023 KVM](#)
- [Limitaciones de la ejecución en KVM AL2023](#)

Requisitos de host KVM para funcionar en KVM AL2023

Actualmente, las imágenes KVM están calificadas en un host que ejecuta Ubuntu 22.04.3 LTS con la versión 6.2+dfsg-2ubuntu6.15 de `qemu` que proporciona esta versión de Ubuntu, mediante un tipo de máquina `q35` para `x86-64` y un tipo de máquina `virt` para `aarch64`.

Soporte de dispositivo para un KVM AL2023

Los modelos de **qemu** dispositivos probados para su uso con imágenes AL2023 KVM (ambos **aarch64** y **x86-64**) son:

- `virtio-blk` (dispositivo de bloques `virtio`)
- `virtio-scsi` (controlador con dispositivo de disco `virtio` SCSI)
- `virtio-net` (dispositivo de red `virtio`)
- `ahci` (para usar con la unidad de CD-ROM virtual)
- `usb-storage` (a través de `xhci`)

Otros modelos de **qemu** dispositivos que permiten la calificación de imágenes AL2023 KVM, pero que no son muy exigentes, son:

- VGA (qemu VGA) solo en x86-64
- `virtio-rng` (generador virtual de números aleatorios)
- dispositivos de teclado AT y ratón PS/2 heredados
- dispositivo serie heredado

Modo de arranque (UEFIyBIOS) compatibilidad con un AL2023 KVM

La imagen x86-64 se ha probado tanto en el modo heredado BIOS como en el de arranque UEFI. Las imágenes aarch64 se prueban en el modo de arranque UEFI.

Note

De forma predeterminada, cuando se utiliza el modo de arranque UEFI, algunos administradores de máquinas virtuales aprovisionan la máquina virtual con claves de arranque seguro de Microsoft (y, por lo tanto, habilitan el arranque seguro). Esta configuración no AL2023 arrancará.

Como el gestor de AL2023 arranque no está firmado por Microsoft, la máquina virtual debe aprovisionarse sin claves UEFI o con las AL2023 claves de arranque seguro.

Important

Aún no se ha validado la compatibilidad con arranque seguro para imágenes KVM.

Limitaciones de la ejecución en KVM AL2023

Existen algunas limitaciones conocidas a la hora de funcionar con AL2023 KVM.

Note

El código que implementa algunas de las funciones no compatibles enumeradas puede existir AL2023 y funcionar correctamente. La lista de funciones no compatibles existe para

que puedas tomar decisiones informadas sobre qué funciones están disponibles actualmente y cuáles serán compatibles en futuras actualizaciones según el equipo de Amazon Linux.

Limitaciones conocidas de la ejecución AL2023 en KVM

- Actualmente, el agente invitado KVM no está empaquetado ni es compatible.
- No se admite la conexión y desconexión en caliente de la CPU, la memoria o cualquier otro tipo de dispositivo.
- No se admite la hibernación de máquinas virtuales.
- No se admite la migración de máquinas virtuales.
- No se admite la transferencia de ningún dispositivo, por ejemplo, mediante PCI Passthrough o USB Passthrough.

Requisitos para ejecutar AL2023 en VMware

En esta sección se describen los requisitos para ejecutar AL2023 en VMware. Las imágenes VMware de AL2023 están disponibles solo para la arquitectura x86-64. Las imágenes VMware para aarch64 no están disponibles o no son compatibles. Estos requisitos son adicionales a los [Requisitos del sistema de AL2023](#) básicos para imágenes VMware.

Temas

- [Requisitos de host de VMware para ejecutar AL2023 en VMware](#)
- [Compatibilidad de dispositivos para AL2023 en VMware](#)
- [Compatibilidad con el modo de arranque \(UEFI y BIOS\) para AL2023 en VMware](#)
- [Limitaciones al ejecutar AL2023 en VMware](#)

Requisitos de host de VMware para ejecutar AL2023 en VMware

Las imágenes OVA de VMware en AL2023 cumplen actualmente lo siguiente:

- VMware Workstation 17.5.0 que se ejecuta en hosts con un procesador Intel(R) Xeon(R) Platinum 8124M
- VMware vSphere 8.0 con un procesador Intel(R) Xeon(R) Platinum 8275CL

Las imágenes OVA de VMware en AL2023 especifican una versión 13 de hardware de la máquina.

La versión 13 de hardware de la máquina de VMware es compatible con:

- ESXi 6.5 o posterior
- VMware Workstation 14 o posterior

Compatibilidad de dispositivos para AL2023 en VMware

Los siguientes modelos de dispositivos VMware se probaron para su uso con imágenes OVA de VMware en AL2023 (solo **x86-64**)

- `vmw_pvscsi` (controlador VMware paravirtualizado de SCSI)
- `vmxnet3` (dispositivo de red paravirtualizado de VMware)
- `ata_piix` (IDE heredado para usar solo con la unidad de CD-ROM virtual)

Otros modelos de dispositivos de VMware que están habilitados para la calificación de imágenes VMware de AL2023, pero que no se han utilizado demasiado:

- `vmw_vmci` e interfaz de `vsock` relacionada (transporte de socket virtual para el agente invitado de VMware)
- dispositivo de memoria de globo `vmw_balloon`
- VMware controlador de SVGA
- dispositivos de teclado AT y ratón PS/2 heredados

El paquete del agente invitado de VMware (`open-vm-tools`) está disponible e instalado de forma predeterminada en las imágenes OVA de VMware en AL2023.

Compatibilidad con el modo de arranque (UEFI y BIOS) para AL2023 en VMware

A partir de la versión 2023.3.20231211, la imagen OVA de VMware de AL2023 se validó tanto en el modo heredado BIOS como en el modo de arranque UEFI. La configuración predeterminada de OVA sigue siendo la BIOS heredada, pero el usuario puede cambiarla.

 Important

La compatibilidad con arranque seguro requiere UEFI, que no ha sido validado para AL2023 ejecutándose en VMware.

Limitaciones al ejecutar AL2023 en VMware

Existen algunas limitaciones conocidas al ejecutar AL2023 en VMware.

 Note

Es posible que en AL2023 haya código que implemente algunas de las funcionalidades no compatibles enumeradas y que funcione correctamente. La lista de funciones no compatibles existe para que los clientes puedan tomar decisiones informadas sobre en qué pueden confiar para trabajar en la actualidad y en qué aspectos calificará el equipo de Amazon Linux como parte de futuras actualizaciones.

Limitaciones conocidas al ejecutar AL2023 en VMware

- El arranque seguro de UEFI no está validado actualmente con AL2023 en VMware.
- No se admite la conexión y desconexión en caliente de la CPU, la memoria o cualquier otro tipo de dispositivo.
- No se admite la hibernación de máquinas virtuales.
- No se admite la migración de máquinas virtuales.
- No se admite la transferencia de ningún dispositivo, por ejemplo, mediante PCI Passthrough o USB Passthrough.

Requisitos para ejecutar Amazon Linux 2023 en Hyper-V

En esta sección se explican los requisitos necesarios para ejecutar Amazon Linux 2023 en Hyper-V. Las imágenes Hyper-V de AL2023 están disponibles solo para la arquitectura x86-64. Las imágenes de Hyper-V para aarch64 no están disponibles ni son compatibles en este momento.

En esta sección se explican los requisitos adicionales además de los requisitos básicos de [Requisitos del sistema de AL2023](#) para las imágenes Hyper-V.

Temas

- [Requisitos del host de Hyper-V para ejecutar Amazon Linux 2023 en Hyper-V](#)
- [Compatibilidad de dispositivos para Amazon Linux 2023 en Hyper-V](#)
- [Limitaciones al ejecutar Amazon Linux 2023 en Hyper-V](#)

Requisitos del host de Hyper-V para ejecutar Amazon Linux 2023 en Hyper-V

La principal calificación de Amazon Linux 2023 en Hyper-V ocurre en Windows Server 2022 que se ejecuta en una instancia `c5.metal` de EC2.

Compatibilidad de dispositivos para Amazon Linux 2023 en Hyper-V

Amazon Linux 2023 se ha probado en máquinas virtuales Hyper-V de primera y segunda generación con el siguiente conjunto de hardware virtualizado:

- Máquina virtual de primera generación (arranque de BIOS heredado)
- Máquina virtual de segunda generación (arranque UEFI, arranque no seguro)
- Los siguientes modelos de dispositivos se han probado para su uso con imágenes de Hyper-V de AL2023:
 - Almacenamiento virtual Hyper-V `hv_storvsc` para el disco raíz y la unidad de CD-ROM emulada en las máquinas virtuales de segunda generación
 - IDE PIIX emulado para la unidad de CD-ROM virtual en máquinas virtuales `ata_piix` de primera generación
 - Ethernet virtual `hv_netvsc` de Hyper-V
- Los siguientes modelos de dispositivos están habilitados, pero se han probado ligeramente:
 - Modo de texto VGA tradicional en máquinas virtuales de primera generación
 - Framebuffer `simplifiedmfb` basado en el firmware UEFI en las máquinas virtuales de segunda generación
 - Hyper-V Balloon `hv_balloon`
 - Hyper-V HID/ratón `hid_hyperv`
- Los siguientes modos de dispositivo no están habilitados en AL2023 en este momento:
 - Transferencia PCI Hyper-V
 - Gráficos Hyper-V DRM

 Important

Para las máquinas virtuales de segunda generación, no se admite el arranque seguro y debe deshabilitarse antes de lanzar la máquina virtual para que Amazon Linux 2023 arranque correctamente. Actualmente, Hyper-V solo admite el arranque seguro con componentes de software firmados por las propias claves de Microsoft, mientras que el gestor de arranque de Amazon Linux está firmado por una clave privada de Amazon. En este momento, Hyper-V no admite la importación de claves de terceros.

Limitaciones al ejecutar Amazon Linux 2023 en Hyper-V

A continuación se indican algunas limitaciones conocidas de la ejecución de Amazon Linux 2023 en Hyper-V:

 Note

Es posible que en AL2023 haya código que implemente algunas de las funcionalidades no compatibles enumeradas y que funcione correctamente. La lista de funciones no compatibles existe para que los clientes puedan tomar decisiones informadas sobre en qué pueden confiar para trabajar en la actualidad y en qué aspectos calificará el equipo de Amazon Linux como parte de futuras actualizaciones.

Limitaciones conocidas al ejecutar AL2023 en Hyper-V

- El modo de arranque seguro UEFI actualmente no es compatible ni funciona con AL2023 en Hyper-V
- No se admite la conexión y desconexión en caliente de la CPU, la memoria o cualquier otro tipo de dispositivo.
- No se admite la hibernación de máquinas virtuales (VM).
- No se admite la migración de máquinas virtuales (VM).
- No se admite la transferencia de ningún dispositivo, por ejemplo, mediante PCI Passthrough o USB Passthrough.

Instalación y configuración de **cloud-init** de Amazon Linux 2023 cuando se utiliza fuera de Amazon EC2

En esta sección, se explica cómo instalar y configurar una máquina virtual Amazon Linux 2023 cuando no se ejecuta directamente en Amazon EC2, como en KVM, VMware o Hyper-V.

De forma predeterminada, las imágenes de una máquina virtual Amazon Linux 2023 no vienen provistas de ninguna contraseña de usuario o clave ssh y obtienen su configuración de red a través de DHCP de la primera interfaz de red descubierta. Esto significa que, de forma predeterminada, sin una configuración adicional, no hay forma de conectarse a la máquina virtual resultante.

Por lo tanto, es necesario proporcionar algún tipo de configuración a la máquina virtual. El mecanismo estándar para hacerlo en Amazon Linux es a través de orígenes de datos `cloud-init`.

Amazon Linux 2023 ha sido cualificado con los siguientes orígenes de datos:

NoCloud

Este es el método tradicional de configurar imágenes en las instalaciones mediante un CD-ROM virtual que contiene una imagen ISO9660 inicial con archivos de configuración `cloud-init`.

VMware

Amazon Linux 2023 también admite la configuración de imágenes de VMware que se ejecutan en vSphere mediante el origen de datos específico de VMware mediante `guestinfo.userdata` y `guestinfo.metadata` de VMware.

Note

La configuración de los orígenes de datos puede diferir de la de Amazon Linux 2. Más específicamente, Amazon Linux 2023 utiliza `systemd-networkd` para su configuración y requiere el uso de “Networking Config Version 2” de `cloud-init`, tal como se documenta en la [documentación de configuración de red de cloud-init](#).

La documentación completa sobre los mecanismos de configuración de `cloud-init` para la versión empaquetada de `cloud-init` en Amazon Linux 2023 se encuentra en la [documentación upstream de cloud-init](#).

Configuración de NoCloud (**seed.iso**) **cloud-init** para Amazon Linux 2023 en KVM y VMware

En esta sección se explica cómo crear y usar una imagen `seed.iso` para configurar Amazon Linux 2023 en KVM o VMware. Como los entornos KVM y VMware no tienen el [Servicio de metadatos de instancias de Amazon EC2 \(IMDS\)](#), se requiere un método alternativo para configurar Amazon Linux 2023, y uno de esos métodos es proporcionar una imagen `seed.iso`.

La imagen de arranque `seed.iso` contiene la información de configuración inicial necesaria para arrancar y configurar la nueva máquina virtual, como, por ejemplo, la configuración de red, el nombre de host y los datos de usuario.

Note

La imagen `seed.iso` solo contiene la información de configuración necesaria para arrancar la máquina virtual. No contiene los archivos de sistema operativo Amazon Linux 2023.

Para generar la imagen `seed.iso`, necesita al menos dos archivos de configuración, a veces tres:

meta-data

Este archivo normalmente incluye el nombre de host de la máquina virtual.

user-data

Este archivo normalmente configura las cuentas de usuario, sus contraseñas, pares de claves ssh o mecanismos de acceso. De forma predeterminada, las imágenes de VMware y KVM en Amazon Linux 2023 crean una cuenta de usuario `ec2-user`. Usted puede utilizar el archivo de configuración `user-data` para establecer la contraseña o las claves ssh de esta cuenta de usuario predeterminada.

network-config (opcional)

Por lo general, este archivo proporciona una configuración de red para la máquina virtual que anulará la configuración predeterminada. La configuración predeterminada utiliza DHCP en la primera interfaz de red disponible.

Creación de la imagen **seed.iso** del disco

1. En un equipo Linux o macOS, cree una nueva carpeta llamada `seedconfig` y acceda a ella.

 Note

Es posible usar Windows u otro sistema operativo para completar estos pasos, pero tendrá que buscar la herramienta equivalente a `mkisofs` para completar la creación de la imagen `seed.iso`.

2. Cree el archivo de configuración `meta-data`.
 - a. Cree un nuevo archivo llamado `meta-data`.
 - b. Abra el archivo `meta-data` con su editor preferido y agregue lo siguiente, reemplazando `vm-hostname` con el nombre de host de la máquina virtual:

```
#cloud-config
local-hostname: vm-hostname
```

- c. Guarde y cierre el archivo de configuración `meta-data`.
3. Cree el archivo de configuración `user-data`.
 - a. Cree un nuevo archivo llamado `user-data`.
 - b. Abra el archivo `user-data` con su editor preferido y agregue lo siguiente, realizando sustituciones según sea necesario:

```
#cloud-config
#vim:syntax=yaml
users:
# A user by the name 'ec2-user' is created in the image by default.
- default
  - name: ec2-user
ssh_authorized_keys:
  - ssh-rsa ssh-key
# In the above line, replace ssh key with the content of your ssh public key.
```

- c. Si lo desea, puede añadir más cuentas de usuario al archivo de configuración `user-data`.

También puede crear cuentas de usuario adicionales y especificar sus mecanismos de acceso, contraseñas y pares de claves. Para obtener más información sobre las directivas admitidas, [consulte la documentación de upstream de cloud-init](#).

- d. Guarde y cierre el archivo de configuración `user-data`.

4. (Opcional) Cree el archivo de configuración `network-config`.
 - a. Cree un nuevo archivo llamado `network-config`.
 - b. Abra el archivo `network-config` con su editor preferido y agregue lo siguiente, reemplazando las distintas direcciones IP con las apropiadas para su configuración.

```
#cloud-config
version: 2
ethernets:
  enp1s0:
    addresses:
      - 192.168.122.161/24
    gateway4: 192.168.122.1
    nameservers:
      addresses: 192.168.122.1
```

 Note

La configuración de red `cloud-init` proporciona mecanismos para hacer coincidir la dirección de la interfaz MAC en lugar de especificar el nombre de la interfaz, que puede cambiar en función de la configuración de la máquina virtual. Esta (y más) características `cloud-init` para la configuración de la red se describen con más detalle en la [documentación upstream de Network Config versión 2 cloud-init](#).

- c. Guarde y cierre el archivo de configuración `network-config`.
5. Cree la imagen de disco `seed.iso` con los archivos de configuración `meta-data`, `user-data` y `network-config` opcional creados en los pasos anteriores.

Lleve a cabo una de las siguientes acciones, dependiendo del sistema operativo en el que esté creando la imagen de disco `seed.iso`.

- En los sistemas Linux, utilice una herramienta como **mkisofs** o **genisoimage** para crear el archivo completo `seed.iso`. Vaya a la carpeta `seedconfig` y ejecute el comando siguiente:

```
$ mkisofs -output seed.iso -volid cidata -joliet -rock user-data meta-data
```

- Si usa un `network-config`, inclúyalo en la invocación de **mkisofs**:

```
$ mkisofs -output seed.iso -volid cidata -joliet -rock user-data meta-data
network-config
```

- En los sistemas macOS, puede utilizar una herramienta como **hdiutil** para generar el archivo `seed.iso` terminado. Como **hdiutil** toma un nombre de ruta en lugar de una lista de archivos, se puede usar la misma invocación independientemente de si se ha creado o no un archivo de configuración `network-config`.

```
$ hdiutil makehybrid -o seed.iso -hfs -joliet -iso -default-volume-name cidata
seedconfig/
```

6. El archivo `seed.iso` resultante ahora se puede adjuntar a su nueva máquina virtual Amazon Linux 2023 a través de una unidad de CD-ROM virtual para que `cloud-init` los encuentre en el primer arranque y aplique la configuración al sistema.

VMwarecloud-initconfiguración guestinfo para on AL2023 VMware

VMwarelos entornos no tienen el [Amazon EC2 Instance Metadata Service \(IMDS\)](#), por lo que AL2023 se requiere un método de configuración alternativo. En esta sección se explica cómo utilizar un mecanismo de configuración alternativo a la unidad de CD-ROM virtual `seed.iso` que se encuentra disponible en VMware vSphere.

Este método de configuración utiliza el mecanismo `extraconfig` de VMware para proporcionar datos de configuración a `cloud-init`. Para cada una de las siguientes claves, se debe proporcionar la propiedad **keyname.encoding** correspondiente.

Se pueden proporcionar las siguientes claves al mecanismo `extraconfig` de VMware.

guestinfo.metadata

JSON o YAML que contenga metadatos `cloud-init`

guestinfo userdata

documento YAML que contiene datos de usuario `cloud-init` en formato `cloud-config`.

guestinfo.vendordata (opcional)

YAML que contiene datos del proveedor `cloud-init`

Las propiedades de codificación correspondientes (`guestinfo.metadata.encoding`, `guestinfo.userdata.encoding` y `guestinfo.vendordata.encoding`) pueden contener:

base64

El contenido de la propiedad está codificado en base base64.

gzip+base64

El contenido de la propiedad se comprime con gzip después de la codificación base64.

Note

El método `seed.iso` admite un archivo de configuración `network-config` independiente (opcional). VMware de `guestinfo` difiere en la forma en que se proporciona la configuración de red. Las secciones siguientes facilitarán información adicional.

Si se desea una configuración de red explícita, debe estar integrada en metadata en forma de dos propiedades YAML o JSON:

network

Contiene la configuración de red codificada en formato JSON o YAML.

network.encoding

Contiene la codificación de los datos de configuración de red anteriores. Las codificaciones `cloud-init` admitidas son las mismas que para los datos `guestinfo`: `base64` y `gzip+base64`.

Example Uso de la herramienta CLI **govc** de VMware vSphere para pasar la configuración con **guestinfo**

1. Prepare los archivos de configuración meta-data y user-data y el archivo de configuración opcional `network-config` tal y como se describe en [Configuración de NoCloud \(seed.iso\) cloud-init para Amazon Linux 2023 en KVM y VMware](#).
2. Convierta los archivos de configuración a formatos utilizables por `guestinfo` de VMware.

```
# 'meta-data', `user-data` and `network-config` are the configuration
```

```
# files in the same format that would be used by a NoCloud (seed.iso)
# data source, read-them and convert them to VMware guestinfo
#
# The VM_NAME variable is assumed to be set to the name of the VM
# It is assumed that the necessary govc environment (credentials etc...) are
  already set

metadata=$(cat "meta-data")
userdata=$(cat "user-data")
if [ -e "network-config" ] ; then
    # We need to embed the network config inside the meta-data
    netconf=$(base64 -w0 "network-config")
    metadata=$(printf "%s\nnetwork: %s\nnetwork.encoding: base64" "$metadata"
"$netconf")
fi
metadata=$(base64 -w0 <<< "$metadata")
govc vm.change -vm "$VM_NAME" \
    -e guestinfo.metadata="$metadata" \
    -e guestinfo.metadata.encoding="base64"
userdata=$(base64 -w0 <<< "$userdata")
govc vm.change -vm "$VM_NAME" \
    -e guestinfo.userdata="$userdata" \
    -e guestinfo.userdata.encoding="base64"
```

Comparación de los paquetes instalados en la AMI estándar de Amazon Linux 2023 con la imagen KVM de AL2023

Comparación entre los RPM presentes en el AMI estándar AL2023 y los RPM presentes en la imagen KVM de AL2023

Paquete	AMI	KVM
acl	2.3.1	2.3.1
acpid	2.0.32	
alternatives	1.15	1.15
amazon-chrony-config	4.3	

Paquete	AMI	KVM
amazon-ec2-net-utils	2.5.1	
amazon-linux-onprem		1.2
amazon-linux-repo-cdn		2023.6.20241031
amazon-linux-repo-s3	2023.6.20241031	
amazon-linux-sb-keys	2023.1	2023.1
amazon-onprem-netw ork		1.2
amazon-rpm-config	228	228
amazon-ssm-agent	3.3.987.0	3.3.987.0
amd-ucode-firmware	20210208 (noarch)	20210208 (noarch)
at	3.1.23	3.1.23
attr	2.5.1	2.5.1
audit	3.0.6	3.0.6
audit-libs	3.0.6	3.0.6
aws-cfn-bootstrap	2.0	
awscli-2	2.15.30	2.15.30
basesystem	11	11
bash	5.2.15	5.2.15
bash-completion	2.11	2.11
bc	1.07.1	1.07.1

Paquete	AMI	KVM
bind-libs	9.18.28	9.18.28
bind-license	9.18.28	9.18.28
bind-utils	9.18.28	9.18.28
binutils	2.39	2.39
boost-filesystem	1.75.0	1.75.0
boost-system	1.75.0	1.75.0
boost-thread	1.75.0	1.75.0
bzip2	1.0.8	1.0.8
bzip2-libs	1.0.8	1.0.8
ca-certificates	2023.2.68	2023.2.68
c-ares	1.19.1	
checkpolicy	3.4	3.4
chkconfig	1.15	1.15
chrony	4.3	4.3
cloud-init	22.2.2	22.2.2
cloud-init-cfg-ec2	22.2.2	
cloud-init-cfg-onpre		22.2.2
cloud-utils-growpart	0.31	0.31
coreutils	8.32	8.32
coreutils-common	8.32	8.32

Paquete	AMI	KVM
cpio	2.13	2.13
cracklib	2.9.6	2.9.6
cracklib-dicts	2.9.6	2.9.6
crontabs	1.11	1.11
crypto-policies	20220428	20220428
crypto-policies-scripts	20220428	20220428
cryptsetup	2.6.1	2.6.1
cryptsetup-libs	2.6.1	2.6.1
curl-minimal	8.5.0	8.5.0
cyrus-sasl-lib	2.1.27	2.1.27
cyrus-sasl-plain	2.1.27	2.1.27
dbus	1.12.28	1.12.28
dbus-broker	32	32
dbus-common	1.12.28	1.12.28
dbus-libs	1.12.28	1.12.28
device-mapper	1.02.185	1.02.185
device-mapper-libs	1.02.185	1.02.185
diffutils	3.8	3.8
dnf	4.14.0	4.14.0
dnf-data	4.14.0	4.14.0

Paquete	AMI	KVM
dnf-plugin-release-notification	1.2	1.2
dnf-plugins-core	4.3.0	4.3.0
dnf-plugin-support-info	1.2	1.2
dnf-utils	4.3.0	4.3.0
dosfstools	4.2	4.2
dracut	055	055
dracut-config-ec2	3.0	
dracut-config-generic	055	055
dwz	0.14	0.14
dyninst	10.2.1	10.2.1
e2fsprogs	1.46.5	1.46.5
e2fsprogs-libs	1.46.5	1.46.5
ec2-hibinit-agent	1.0.8	
ec2-instance-connect	1.1	
ec2-instance-connect-selinux	1.1	
ec2-utils	2.2.0	
ed	1.14.2	1.14.2
efi-filesystem	5	5

Paquete	AMI	KVM
efi-srpm-macros	5	5
efivar	38	38
efivar-libs	38	38
elfutils-debuginfod-client	0.188	0.188
elfutils-default-yama-scope	0.188	0.188
elfutils-libelf	0.188	0.188
elfutils-libs	0.188	0.188
ethtool	5.15	5.15
expat	2.5.0	2.5.0
file	5.39	5.39
file-libs	5.39	5.39
filesystem	3.14	3.14
findutils	4.8.0	4.8.0
fonts-srpm-macros	2.0.5	2.0.5
fstrm	0.6.1	0.6.1
fuse-libs	2.9.9	2.9.9
gawk	5.1.0	5.1.0
gdbm-libs	1.19	1.19
gdisk	1.0.8	1.0.8

Paquete	AMI	KVM
gettext	0.21	0.21
gettext-libs	0.21	0.21
ghc-srpm-macros	1.5.0	1.5.0
glib2	2.74.7	2.74.7
glibc	2.34	2.34
glibc-all-langpacks	2.34	2.34
glibc-common	2.34	2.34
glibc-gconv-extra	2.34	2.34
glibc-locale-source	2.34	2.34
gmp	6.2.1	6.2.1
gnupg2-minimal	2.3.7	2.3.7
gnutls	3.8.0	3.8.0
go-srpm-macros	3.2.0	3.2.0
gpgme	1.15.1	1.15.1
gpm-libs	1.20.7	1.20.7
grep	3.8	3.8
groff-base	1.22.4	1.22.4
grub2-common	2.06	2.06
grub2-efi-aa64-ec2	2.06 (aarch64)	2.06 (aarch64)
grub2-efi-x64-ec2	2.06 (x86_64)	2.06 (x86_64)

Paquete	AMI	KVM
grub2-pc		2.06 (x86_64)
grub2-pc-modules	2.06	2.06 (noarch)
grub2-tools	2.06	2.06
grub2-tools-minimal	2.06	2.06
grubby	8.40	8.40
gssproxy	0.8.4	0.8.4
gzip	1.12	1.12
hostname	3.23	3.23
hunspell	1.7.0	1.7.0
hunspell-en	0.20140811.1	0.20140811.1
hunspell-en-GB	0.20140811.1	0.20140811.1
hunspell-en-US	0.20140811.1	0.20140811.1
hunspell-filesystem	1.7.0	1.7.0
hwdata	0.384	0.384
info	6.7	6.7
inih	49	49
initscripts	10.09	10.09
iproute	6.10.0	6.10.0
iputils	20210202	20210202
irqbalance	1.9.0	1.9.0

Paquete	AMI	KVM
jansson	2.14	2.14
jemalloc	5.2.1	5.2.1
jitterentropy	3.4.1	3.4.1
jq	1.7.1	
json-c	0.14	0.14
kbd	2.4.0	2.4.0
kbd-misc	2.4.0	2.4.0
kernel	6.1.112	6.1.112
kernel-libbpf	6.1.112	6.1.112
kernel-livepatch-r epo-cdn		2023.6.20241031
kernel-livepatch-r epo-s3	2023.6.20241031	
kernel-modules-extra		6.1.112
kernel-modules-ext ra-common		6.1.112
kernel-srpm-macros	1.0	1.0
kernel-tools	6.1.112	6.1.112
keyutils	1.6.3	1.6.3
keyutils-libs	1.6.3	1.6.3
kmod	29	29

Paquete	AMI	KVM
kmod-libs	29	29
kpatch-runtime	0.9.7	0.9.7
krb5-libs	1.21.3	1.21.3
less	608	608
libacl	2.3.1	2.3.1
libaio	0.3.111	0.3.111
libarchive	3.7.4	3.7.4
libargon2	20171227	20171227
libassuan	2.5.5	2.5.5
libattr	2.5.1	2.5.1
libbasicobjects	0.1.1	0.1.1
libblkid	2.37.4	2.37.4
libcap	2.48	2.48
libcap-ng	0.8.2	0.8.2
libcbor	0.7.0	0.7.0
libcollection	0.7.0	0.7.0
libcom_err	1.46.5	1.46.5
libcomps	0.1.20	0.1.20
libconfig	1.7.2	1.7.2
libcurl-minimal	8.5.0	8.5.0

Paquete	AMI	KVM
libdb	5.3.28	5.3.28
libdhash	0.5.0	
libdnf	0.69.0	0.69.0
libeconf	0.4.0	0.4.0
libedit	3.1	3.1
libev	4.33	4.33
libevent	2.1.12	2.1.12
libfdisk	2.37.4	2.37.4
libffi	3.4.4	3.4.4
libfido2	1.10.0	1.10.0
libgcc	11.4.1	11.4.1
libgcrypt	1.10.2	1.10.2
libgomp	11.4.1	11.4.1
libgpg-error	1.42	1.42
libibverbs	48.0	48.0
libidn2	2.3.2	2.3.2
libini_config	1.3.1	1.3.1
libkcapi	1.4.0	1.4.0
libkcapi-hmacalc	1.4.0	1.4.0
libldb	2.6.2	

Paquete	AMI	KVM
libmaxminddb	1.5.2	1.5.2
libmetalink	0.1.3	0.1.3
libmnl	1.0.4	1.0.4
libmodulemd	2.13.0	2.13.0
libmount	2.37.4	2.37.4
libnfsidmap	2.5.4	2.5.4
libnghttp2	1.59.0	1.59.0
libnl3	3.5.0	3.5.0
libpath_utils	0.2.1	0.2.1
libpcap	1.10.1	1.10.1
libpipeline	1.5.3	1.5.3
libpkgconf	1.8.0	1.8.0
libpsl	0.21.1	0.21.1
libpwquality	1.4.4	1.4.4
libref_array	0.1.5	0.1.5
librepo	1.14.5	1.14.5
libreport-filessystem	2.15.2	2.15.2
libseccomp	2.5.3	2.5.3
libselinux	3.4	3.4
libselinux-utils	3.4	3.4

Paquete	AMI	KVM
libsemanage	3.4	3.4
libsepol	3.4	3.4
libsigsegv	2.13	2.13
libsmartcols	2.37.4	2.37.4
libsolv	0.7.22	0.7.22
libss	1.46.5	1.46.5
libsss_certmap	2.9.4	
libsss_idmap	2.9.4	2.9.4
libsss_nss_idmap	2.9.4	2.9.4
libsss_sudo	2.9.4	
libstdc++	11.4.1	11.4.1
libstoragemgmt	1.9.4	1.9.4
libtalloc	2.3.4	
libtasn1	4.19.0	4.19.0
libtdb	1.4.7	
libtevent	0.13.0	
libtextstyle	0.21	0.21
libtirpc	1.3.3	1.3.3
libunistring	0.9.10	0.9.10
libuser	0.63	0.63

Paquete	AMI	KVM
libutempter	1.2.1	1.2.1
libuuid	2.37.4	2.37.4
libuv	1.47.0	1.47.0
libverto	0.3.2	0.3.2
libverto-libev	0.3.2	0.3.2
libxcrypt	4.4.33	4.4.33
libxml2	2.10.4	2.10.4
libyaml	0.2.5	0.2.5
libzstd	1.5.5	1.5.5
linux-firmware-whe nce	20210208 (noarch)	20210208 (noarch)
lm_sensors-libs	3.6.0	3.6.0
lmdb-libs	0.9.29	0.9.29
logrotate	3.20.1	3.20.1
lsof	4.94.0	4.94.0
lua-libs	5.4.4	5.4.4
lua-srpm-macros	1	1
lz4-libs	1.9.4	1.9.4
man-db	2.9.3	2.9.3
man-pages	5.10	5.10
microcode_ctl	2.1 (x86_64)	2.1 (x86_64)

Paquete	AMI	KVM
mpfr	4.1.0	4.1.0
nano	5.8	5.8
ncurses	6.2	6.2
ncurses-base	6.2	6.2
ncurses-libs	6.2	6.2
nettle	3.8	3.8
net-tools	2.0	2.0
newt	0.52.21	0.52.21
nfs-utils	2.5.4	2.5.4
npth	1.6	1.6
nspr	4.35.0	4.35.0
nss	3.90.0	3.90.0
nss-softoken	3.90.0	3.90.0
nss-softoken-freebl	3.90.0	3.90.0
nss-sysinit	3.90.0	3.90.0
nss-util	3.90.0	3.90.0
ntsysv	1.15	1.15
numactl-libs	2.0.14	2.0.14
ocaml-srpm-macros	6	6
oniguruma	6.9.7.1	

Paquete	AMI	KVM
openblas-srpm-macros	2	2
openldap	2.4.57	2.4.57
openssh	8.7p1	8.7p1
openssh-clients	8.7p1	8.7p1
openssh-server	8.7p1	8.7p1
openssl	3.0.8	3.0.8
openssl-lib	3.0.8	3.0.8
openssl-pkcs11	0.4.12	0.4.12
os-prober	1.77	1.77
p11-kit	0.24.1	0.24.1
p11-kit-trust	0.24.1	0.24.1
package-notes-srpm-macros	0.4	0.4
pam	1.5.1	1.5.1
parted	3.4	3.4
passwd	0.80	0.80
pciutils	3.7.0	3.7.0
pciutils-lib	3.7.0	3.7.0
pcre2	10.40	10.40
pcre2-syntax	10.40	10.40
perl-Carp	1.50	1.50

Paquete	AMI	KVM
perl-Class-Struct	0.66	0.66
perl-constant	1.33	1.33
perl-DynaLoader	1.47	1.47
perl-Encode	3.15	3.15
perl-Errno	1.30	1.30
perl-Exporter	5.74	5.74
perl-Fcntl	1.13	1.13
perl-File-Basename	2.85	2.85
perl-File-Path	2.18	2.18
perl-File-stat	1.09	1.09
perl-File-Temp	0.231.100	0.231.100
perl-Getopt-Long	2.52	2.52
perl-Getopt-Std	1.12	1.12
perl-HTTP-Tiny	0.078	0.078
perl-if	0.60.800	0.60.800
perl-interpretter	5.32.1	5.32.1
perl-IO	1.43	1.43
perl-IPC-Open3	1.21	1.21
perl-libs	5.32.1	5.32.1
perl-MIME-Base64	3.16	3.16

Paquete	AMI	KVM
perl-mro	1.23	1.23
perl-overload	1.31	1.31
perl-overloading	0.02	0.02
perl-parent	0.238	0.238
perl-PathTools	3.78	3.78
perl-Pod-Escapes	1.07	1.07
perl-podlators	4.14	4.14
perl-Pod-Perldoc	3.28.01	3.28.01
perl-Pod-Simple	3.42	3.42
perl-Pod-Usage	2.01	2.01
perl-POSIX	1.94	1.94
perl-Scalar-List-Utils	1.56	1.56
perl-SelectSaver	1.02	1.02
perl-Socket	2.032	2.032
perl-srpm-macros	1	1
perl-Storable	3.21	3.21
perl-subs	1.03	1.03
perl-Symbol	1.08	1.08
perl-Term-ANSIColor	5.01	5.01
perl-Term-Cap	1,17	1,17

Paquete	AMI	KVM
perl-Text-ParseWords	3.30	3.30
perl-Text-Tabs+Wrap	2021.0726	2021.0726
perl-Time-Local	1.300	1.300
perl-vars	1.05	1.05
pkgconf	1.8.0	1.8.0
pkgconf-m4	1.8.0	1.8.0
pkgconf-pkg-config	1.8.0	1.8.0
polycoreutils	3.4	3.4
polycoreutils-python-utils	3.4	
popt	1.18	1.18
procps-ng	3.3.17	3.3.17
protobuf-c	1.4.1	1.4.1
psacct	6.6.4	6.6.4
psmisc	23.4	23.4
publicsuffix-list-dafsa	20240212	20240212
python3	3.9.16	3.9.16
python3-attrs	20.3.0	20.3.0
python3-audit	3.0.6	3.0.6
python3-awscrt	0.19.19	0.19.19

Paquete	AMI	KVM
python3-babel	2.9.1	2.9.1
python3-cffi	1.14.5	1.14.5
python3-chardet	4.0.0	4.0.0
python3-colorama	0.4.4	0.4.4
python3-configobj	5.0.6	5.0.6
python3-cryptography	36.0.1	36.0.1
python3-daemon	2.3.0	
python3-dateutil	2.8.1	2.8.1
python3-dbus	1.2.18	1.2.18
python3-distro	1.5.0	1.5.0
python3-dnf	4.14.0	4.14.0
python3-dnf-plugins-core	4.3.0	4.3.0
python3-docutils	0.16	0.16
python3-gpg	1.15.1	1.15.1
python3-hawkey	0.69.0	0.69.0
python3-idna	(2.10)	(2.10)
python3-jinja2	2.11.3	2.11.3
python3-jmespath	0.10.0	0.10.0
python3-jsonpatch	1.21	1.21
python3-jsonpointer	2.0	2.0

Paquete	AMI	KVM
python3-jonschema	3.2.0	3.2.0
python3-libcomps	0.1.20	0.1.20
python3-libdnf	0.69.0	0.69.0
python3-libs	3.9.16	3.9.16
python3-libselinux	3.4	3.4
python3-libsemanage	3.4	3.4
python3-libstorage mgmt	1.9.4	1.9.4
python3-lockfile	0.12.2	
python3-markupsafe	1.1.1	1.1.1
python3-netifaces	0.10.6	0.10.6
python3-oauthlib	3.0.2	3.0.2
python3-pip-wheel	21.3.1	21.3.1
python3-ply	3.11	3.11
python3-policycore utils	3.4	3.4
python3-prettytable	0.7.2	0.7.2
python3-prompt-too lkit	3.0.24	3.0.24
python3-pycparser	2.20	2.20
python3-pyrsistent	0.17.3	0.17.3

Paquete	AMI	KVM
python3-pyserial	3.4	3.4
python3-pysocks	1.7.1	1.7.1
python3-pytz	2022.7.1	2022.7.1
python3-pyyaml	5.4.1	5.4.1
python3-requests	2.25.1	2.25.1
python3-rpm	4.16.1.3	4.16.1.3
python3-ruamel-yaml	0.16.6	0.16.6
python3-ruamel-yaml-clib	0.1.2	0.1.2
python3-setuptools	4.4.1	4.4.1
python3-setuptools	59.6.0	59.6.0
python3-setuptools-wheel	59.6.0	59.6.0
python3-six	1.15.0	1.15.0
python3-systemd	235	235
python3-urllib3	1.25.10	1.25.10
python3-wcwidth	0.2.5	0.2.5
python-chevron	0.13.1	
python-srpm-macros	3.9	3.9
quota	4.06	4.06
quota-nls	4.06	4.06

Paquete	AMI	KVM
readline	8.1	8.1
rng-tools	6.14	6.14
rootfiles	8.1	8.1
rpcbind	1.2.6	1.2.6
rpm	4.16.1.3	4.16.1.3
rpm-build-libs	4.16.1.3	4.16.1.3
rpm-libs	4.16.1.3	4.16.1.3
rpm-plugin-selinux	4.16.1.3	4.16.1.3
rpm-plugin-systemd-inhibit	4.16.1.3	4.16.1.3
rpm-sign-libs	4.16.1.3	4.16.1.3
rsync	3.2.6	3.2.6
rust-srpm-macros	21	21
sbsigntools	0.9.4	0.9.4
screen	4.8.0	4.8.0
sed	4.8	4.8
selinux-policy	38.1.45	38.1.45
selinux-policy-targeted	38.1.45	38.1.45
setup	2.13.7	2.13.7
shadow-utils	4.9	4.9

Paquete	AMI	KVM
slang	2.3.2	2.3.2
sqlite-libs	3.40.0	3.40.0
sssd-client	2.9.4	2.9.4
sssd-common	2.9.4	
sssd-kcm	2.9.4	
sssd-nfs-idmap	2.9.4	
strace	6.8	6.8
sudo	1.9.15	1.9.15
sysctl-defaults	1.0	1.0
sysstat	12.5.6	12.5.6
systemd	252.23	252.23
systemd-libs	252.23	252.23
systemd-networkd	252.23	252.23
systemd-pam	252.23	252.23
systemd-resolved	252.23	252.23
systemd-udev	252.23	252.23
system-release	2023.6.20241031	2023.6.20241031
systemtap-runtime	4.8	4.8
tar	1.34	1.34
tbb	2020.3	2020.3

Paquete	AMI	KVM
tcpdump	4.99.1	4.99.1
tcsh	6.24.07	6.24.07
time	1.9	1.9
traceroute	2.1.3	2.1.3
tzdata	2024a	2024a
unzip	6.0	6.0
update-motd	2.2	2.2
userspace-rcu	0.12.1	0.12.1
util-linux	2.37.4	2.37.4
util-linux-core	2.37.4	2.37.4
vim-common	9.0.2153	9.0.2153
vim-data	9.0.2153	9.0.2153
vim-enhanced	9.0.2153	9.0.2153
vim-filesystem	9.0.2153	9.0.2153
vim-minimal	9.0.2153	9.0.2153
wget	1.21.3	1.21.3
which	2.21	2.21
words	3.0	3.0
xfsdump	3.1.11	3.1.11
xfstools	5.18.0	5.18.0

Paquete	AMI	KVM
xxd	9.0.2153	9.0.2153
xxhash-libs	0.8.0	0.8.0
xz	5.2.5	5.2.5
xz-libs	5.2.5	5.2.5
yum	4.14.0	4.14.0
zip	3.0	3.0
zlib	1.2.11	1.2.11
zram-generator	1.1.2	
zram-generator-def aults	1.1.2	
zstd	1.5.5	1.5.5

Comparación de los paquetes instalados en la AMI estándar de Amazon Linux 2023 con la imagen OVA de VMware AL2023

Comparación entre los RPM presentes en el AMI estándar AL2023 y los RPM presentes en la imagen OVA de VMware AL2023

Paquete	AMI	VMware OVA
acl	2.3.1	2.3.1
acpid	2.0.32	
alternatives	1.15	1.15
amazon-chrony-config	4.3	

Paquete	AMI	VMware OVA
amazon-ec2-net-utils	2.5.1	
amazon-linux-onprem		1.2
amazon-linux-repo-cdn		2023.6.20241031
amazon-linux-repo-s3	2023.6.20241031	
amazon-linux-sb-keys	2023.1	2023.1
amazon-onprem-netw ork		1.2
amazon-rpm-config	228	228
amazon-ssm-agent	3.3.987.0	3.3.987.0
amd-ucode-firmware	20210208	20210208
at	3.1.23	3.1.23
attr	2.5.1	2.5.1
audit	3.0.6	3.0.6
audit-libs	3.0.6	3.0.6
aws-cfn-bootstrap	2.0	
awscli-2	2.15.30	2.15.30
basesystem	11	11
bash	5.2.15	5.2.15
bash-completion	2.11	2.11
bc	1.07.1	1.07.1

Paquete	AMI	VMware OVA
bind-libs	9.18.28	9.18.28
bind-license	9.18.28	9.18.28
bind-utils	9.18.28	9.18.28
binutils	2.39	2.39
boost-filesystem	1.75.0	1.75.0
boost-system	1.75.0	1.75.0
boost-thread	1.75.0	1.75.0
bzip2	1.0.8	1.0.8
bzip2-libs	1.0.8	1.0.8
ca-certificates	2023.2.68	2023.2.68
c-ares	1.19.1	
checkpolicy	3.4	3.4
chkconfig	1.15	1.15
chrony	4.3	4.3
cloud-init	22.2.2	22.2.2
cloud-init-cfg-ec2	22.2.2	
cloud-init-cfg-onpre		22.2.2
cloud-utils-growpart	0.31	0.31
coreutils	8.32	8.32
coreutils-common	8.32	8.32

Paquete	AMI	VMware OVA
cpio	2.13	2.13
cracklib	2.9.6	2.9.6
cracklib-dicts	2.9.6	2.9.6
crontabs	1.11	1.11
crypto-policies	20220428	20220428
crypto-policies-scripts	20220428	20220428
cryptsetup	2.6.1	2.6.1
cryptsetup-libs	2.6.1	2.6.1
curl-minimal	8.5.0	8.5.0
cyrus-sasl-lib	2.1.27	2.1.27
cyrus-sasl-plain	2.1.27	2.1.27
dbus	1.12.28	1.12.28
dbus-broker	32	32
dbus-common	1.12.28	1.12.28
dbus-libs	1.12.28	1.12.28
device-mapper	1.02.185	1.02.185
device-mapper-libs	1.02.185	1.02.185
diffutils	3.8	3.8
dnf	4.14.0	4.14.0
dnf-data	4.14.0	4.14.0

Paquete	AMI	VMware OVA
dnf-plugin-release-notification	1.2	1.2
dnf-plugins-core	4.3.0	4.3.0
dnf-plugin-support-info	1.2	1.2
dnf-utils	4.3.0	4.3.0
dosfstools	4.2	4.2
dracut	055	055
dracut-config-ec2	3.0	
dracut-config-generic	055	055
dwz	0.14	0.14
dyninst	10.2.1	10.2.1
e2fsprogs	1.46.5	1.46.5
e2fsprogs-libs	1.46.5	1.46.5
ec2-hibinit-agent	1.0.8	
ec2-instance-connect	1.1	
ec2-instance-connect-selinux	1.1	
ec2-utils	2.2.0	
ed	1.14.2	1.14.2
efi-filesystem	5	5

Paquete	AMI	VMware OVA
efi-srpm-macros	5	5
efivar	38	38
efivar-libs	38	38
elfutils-debuginfod-client	0.188	0.188
elfutils-default-yama-scope	0.188	0.188
elfutils-libelf	0.188	0.188
elfutils-libs	0.188	0.188
ethtool	5.15	5.15
expat	2.5.0	2.5.0
file	5.39	5.39
file-libs	5.39	5.39
filesystem	3.14	3.14
findutils	4.8.0	4.8.0
fonts-srpm-macros	2.0.5	2.0.5
fstrm	0.6.1	0.6.1
fuse3		3.10.4
fuse3-libs		3.10.4
fuse-common		3.10.4
fuse-libs	2.9.9	2.9.9

Paquete	AMI	VMware OVA
gawk	5.1.0	5.1.0
gdbm-libs	1.19	1.19
gdisk	1.0.8	1.0.8
gettext	0.21	0.21
gettext-libs	0.21	0.21
ghc-srpm-macros	1.5.0	1.5.0
glib2	2.74.7	2.74.7
glibc	2.34	2.34
glibc-all-langpacks	2.34	2.34
glibc-common	2.34	2.34
glibc-gconv-extra	2.34	2.34
glibc-locale-source	2.34	2.34
gmp	6.2.1	6.2.1
gnupg2-minimal	2.3.7	2.3.7
gnutls	3.8.0	3.8.0
go-srpm-macros	3.2.0	3.2.0
gpgme	1.15.1	1.15.1
gpm-libs	1.20.7	1.20.7
grep	3.8	3.8
groff-base	1.22.4	1.22.4

Paquete	AMI	VMware OVA
grub2-common	2.06	2.06
grub2-efi-x64-ec2	2.06	2.06
grub2-pc		2.06
grub2-pc-modules	2.06	2.06
grub2-tools	2.06	2.06
grub2-tools-minimal	2.06	2.06
grubby	8.40	8.40
gssproxy	0.8.4	0.8.4
gzip	1.12	1.12
hostname	3.23	3.23
hunspell	1.7.0	1.7.0
hunspell-en	0.20140811.1	0.20140811.1
hunspell-en-GB	0.20140811.1	0.20140811.1
hunspell-en-US	0.20140811.1	0.20140811.1
hunspell-filesystem	1.7.0	1.7.0
hwdata	0.384	0.384
info	6.7	6.7
inih	49	49
initscripts	10.09	10.09
iproute	6.10.0	6.10.0

Paquete	AMI	VMware OVA
iputils	20210202	20210202
irqbalance	1.9.0	1.9.0
jansson	2.14	2.14
jemalloc	5.2.1	5.2.1
jitterentropy	3.4.1	3.4.1
jq	1.7.1	
json-c	0.14	0.14
kbd	2.4.0	2.4.0
kbd-misc	2.4.0	2.4.0
kernel	6.1.112	6.1.112
kernel-libbpf	6.1.112	6.1.112
kernel-livepatch-r epo-cdn		2023.6.20241031
kernel-livepatch-r epo-s3	2023.6.20241031	
kernel-modules-extra		6.1.112
kernel-modules-ext ra-common		6.1.112
kernel-srpm-macros	1.0	1.0
kernel-tools	6.1.112	6.1.112
keyutils	1.6.3	1.6.3

Paquete	AMI	VMware OVA
keyutils-libs	1.6.3	1.6.3
kmod	29	29
kmod-libs	29	29
kpatch-runtime	0.9.7	0.9.7
krb5-libs	1.21.3	1.21.3
less	608	608
libacl	2.3.1	2.3.1
libaio	0.3.111	0.3.111
libarchive	3.7.4	3.7.4
libargon2	20171227	20171227
libassuan	2.5.5	2.5.5
libattr	2.5.1	2.5.1
libbasicobjects	0.1.1	0.1.1
libblkid	2.37.4	2.37.4
libcap	2.48	2.48
libcap-ng	0.8.2	0.8.2
libcbor	0.7.0	0.7.0
libcollection	0.7.0	0.7.0
libcom_err	1.46.5	1.46.5
libcomps	0.1.20	0.1.20

Paquete	AMI	VMware OVA
libconfig	1.7.2	1.7.2
libcurl-minimal	8.5.0	8.5.0
libdb	5.3.28	5.3.28
libdhash	0.5.0	
libdnf	0.69.0	0.69.0
libeconf	0.4.0	0.4.0
libedit	3.1	3.1
libev	4.33	4.33
libevent	2.1.12	2.1.12
libfdisk	2.37.4	2.37.4
libffi	3.4.4	3.4.4
libfido2	1.10.0	1.10.0
libgcc	11.4.1	11.4.1
libgcrypt	1.10.2	1.10.2
libgomp	11.4.1	11.4.1
libgpg-error	1.42	1.42
libibverbs	48.0	48.0
libidn2	2.3.2	2.3.2
libini_config	1.3.1	1.3.1
libkcap	1.4.0	1.4.0

Paquete	AMI	VMware OVA
libkcapi-hmaccalc	1.4.0	1.4.0
libldb	2.6.2	
libmaxminddb	1.5.2	1.5.2
libmetalink	0.1.3	0.1.3
libmnl	1.0.4	1.0.4
libmodulemd	2.13.0	2.13.0
libmount	2.37.4	2.37.4
libmspack		0.10.1
libnfsidmap	2.5.4	2.5.4
libnghttp2	1.59.0	1.59.0
libnl3	3.5.0	3.5.0
libpath_utils	0.2.1	0.2.1
libpcap	1.10.1	1.10.1
libpipeline	1.5.3	1.5.3
libpkgconf	1.8.0	1.8.0
libpsl	0.21.1	0.21.1
libpwquality	1.4.4	1.4.4
libref_array	0.1.5	0.1.5
librepo	1.14.5	1.14.5
libreport-fsfilesystem	2.15.2	2.15.2

Paquete	AMI	VMware OVA
libseccomp	2.5.3	2.5.3
libselinux	3.4	3.4
libselinux-utils	3.4	3.4
libsemanage	3.4	3.4
libsepol	3.4	3.4
libsigsegv	2.13	2.13
libsmartcols	2.37.4	2.37.4
libsolv	0.7.22	0.7.22
libss	1.46.5	1.46.5
libsss_certmap	2.9.4	
libsss_idmap	2.9.4	2.9.4
libsss_nss_idmap	2.9.4	2.9.4
libsss_sudo	2.9.4	
libstdc++	11.4.1	11.4.1
libstoragemgmt	1.9.4	1.9.4
libtalloc	2.3.4	
libtasn1	4.19.0	4.19.0
libtdb	1.4.7	
libtevent	0.13.0	
libtextstyle	0.21	0.21

Paquete	AMI	VMware OVA
libtirpc	1.3.3	1.3.3
libtool-ltdl		2.4.7
libunistring	0.9.10	0.9.10
libuser	0.63	0.63
libutempter	1.2.1	1.2.1
libuuid	2.37.4	2.37.4
libuv	1.47.0	1.47.0
libverto	0.3.2	0.3.2
libverto-libev	0.3.2	0.3.2
libxcrypt	4.4.33	4.4.33
libxml2	2.10.4	2.10.4
libxslt		1.1.34
libyaml	0.2.5	0.2.5
libzstd	1.5.5	1.5.5
linux-firmware-whe nce	20210208	20210208
lm_sensors-libs	3.6.0	3.6.0
lmdb-libs	0.9.29	0.9.29
logrotate	3.20.1	3.20.1
lsof	4.94.0	4.94.0
lua-libs	5.4.4	5.4.4

Paquete	AMI	VMware OVA
lua-srpm-macros	1	1
lz4-libs	1.9.4	1.9.4
man-db	2.9.3	2.9.3
man-pages	5.10	5.10
microcode_ctl	2.1	2.1
mpfr	4.1.0	4.1.0
nano	5.8	5.8
ncurses	6.2	6.2
ncurses-base	6.2	6.2
ncurses-libs	6.2	6.2
nettle	3.8	3.8
net-tools	2.0	2.0
newt	0.52.21	0.52.21
nfs-utils	2.5.4	2.5.4
npth	1.6	1.6
nspr	4.35.0	4.35.0
nss	3.90.0	3.90.0
nss-softokn	3.90.0	3.90.0
nss-softokn-freebl	3.90.0	3.90.0
nss-sysinit	3.90.0	3.90.0

Paquete	AMI	VMware OVA
nss-util	3.90.0	3.90.0
ntsysv	1.15	1.15
numactl-libs	2.0.14	2.0.14
ocaml-srpm-macros	6	6
oniguruma	6.9.7.1	
openblas-srpm-macros	2	2
openldap	2.4.57	2.4.57
openssh	8.7p1	8.7p1
openssh-clients	8.7p1	8.7p1
openssh-server	8.7p1	8.7p1
openssl	3.0.8	3.0.8
openssl-libs	3.0.8	3.0.8
openssl-pkcs11	0.4.12	0.4.12
open-vm-tools		12.3.0
os-prober	1.77	1.77
p11-kit	0.24.1	0.24.1
p11-kit-trust	0.24.1	0.24.1
package-notes-srpm-macros	0.4	0.4
pam	1.5.1	1.5.1
parted	3.4	3.4

Paquete	AMI	VMware OVA
passwd	0.80	0.80
pciutils	3.7.0	3.7.0
pciutils-libs	3.7.0	3.7.0
pcre2	10.40	10.40
pcre2-syntax	10.40	10.40
perl-Carp	1.50	1.50
perl-Class-Struct	0.66	0.66
perl-constant	1.33	1.33
perl-DynaLoader	1.47	1.47
perl-Encode	3.15	3.15
perl-Errno	1.30	1.30
perl-Exporter	5.74	5.74
perl-Fcntl	1.13	1.13
perl-File-Basename	2.85	2.85
perl-File-Path	2.18	2.18
perl-File-stat	1.09	1.09
perl-File-Temp	0.231.100	0.231.100
perl-Getopt-Long	2.52	2.52
perl-Getopt-Std	1.12	1.12
perl-HTTP-Tiny	0.078	0.078

Paquete	AMI	VMware OVA
perl-if	0.60.800	0.60.800
perl-interpreter	5.32.1	5.32.1
perl-IO	1.43	1.43
perl-IPC-Open3	1.21	1.21
perl-libs	5.32.1	5.32.1
perl-MIME-Base64	3.16	3.16
perl-mro	1.23	1.23
perl-overload	1.31	1.31
perl-overloading	0.02	0.02
perl-parent	0.238	0.238
perl-PathTools	3.78	3.78
perl-Pod-Escapes	1.07	1.07
perl-podlators	4.14	4.14
perl-Pod-Perldoc	3.28.01	3.28.01
perl-Pod-Simple	3.42	3.42
perl-Pod-Usage	2.01	2.01
perl-POSIX	1.94	1.94
perl-Scalar-List-Utils	1.56	1.56
perl-SelectSaver	1.02	1.02
perl-Socket	2.032	2.032

Paquete	AMI	VMware OVA
perl-srpm-macros	1	1
perl-Storable	3.21	3.21
perl-subs	1.03	1.03
perl-Symbol	1.08	1.08
perl-Term-ANSIColor	5.01	5.01
perl-Term-Cap	1,17	1,17
perl-Text-ParseWords	3.30	3.30
perl-Text-Tabs+Wrap	2021.0726	2021.0726
perl-Time-Local	1.300	1.300
perl-vars	1.05	1.05
pkgconf	1.8.0	1.8.0
pkgconf-m4	1.8.0	1.8.0
pkgconf-pkg-config	1.8.0	1.8.0
policycoreutils	3.4	3.4
policycoreutils-python-utils	3.4	
popt	1.18	1.18
procps-ng	3.3.17	3.3.17
protobuf-c	1.4.1	1.4.1
psacct	6.6.4	6.6.4
psmisc	23.4	23.4

Paquete	AMI	VMware OVA
publicsuffix-list-dafsa	20240212	20240212
python3	3.9.16	3.9.16
python3-attrs	20.3.0	20.3.0
python3-audit	3.0.6	3.0.6
python3-awscli	0.19.19	0.19.19
python3-babel	2.9.1	2.9.1
python3-cffi	1.14.5	1.14.5
python3-chardet	4.0.0	4.0.0
python3-colorama	0.4.4	0.4.4
python3-configobj	5.0.6	5.0.6
python3-cryptography	36.0.1	36.0.1
python3-daemon	2.3.0	
python3-dateutil	2.8.1	2.8.1
python3-dbus	1.2.18	1.2.18
python3-distro	1.5.0	1.5.0
python3-dnf	4.14.0	4.14.0
python3-dnf-plugins-core	4.3.0	4.3.0
python3-docutils	0.16	0.16
python3-gpg	1.15.1	1.15.1

Paquete	AMI	VMware OVA
python3-hawkey	0.69.0	0.69.0
python3-idna	(2.10)	(2.10)
python3-jinja2	2.11.3	2.11.3
python3-jmespath	0.10.0	0.10.0
python3-jsonpatch	1.21	1.21
python3-jsonpointer	2.0	2.0
python3-jsonschema	3.2.0	3.2.0
python3-libcomps	0.1.20	0.1.20
python3-libdnf	0.69.0	0.69.0
python3-libs	3.9.16	3.9.16
python3-libselinux	3.4	3.4
python3-libsemanage	3.4	3.4
python3-libstorage mgmt	1.9.4	1.9.4
python3-lockfile	0.12.2	
python3-markupsafe	1.1.1	1.1.1
python3-netifaces	0.10.6	0.10.6
python3-oauthlib	3.0.2	3.0.2
python3-pip-wheel	21.3.1	21.3.1
python3-ply	3.11	3.11

Paquete	AMI	VMware OVA
python3-policycore utils	3.4	3.4
python3-prettytable	0.7.2	0.7.2
python3-prompt-too lkit	3.0.24	3.0.24
python3-pycparser	2.20	2.20
python3-pyrsistent	0.17.3	0.17.3
python3-pyserial	3.4	3.4
python3-pysocks	1.7.1	1.7.1
python3-pytz	2022.7.1	2022.7.1
python3-pyyaml	5.4.1	5.4.1
python3-requests	2.25.1	2.25.1
python3-rpm	4.16.1.3	4.16.1.3
python3-ruamel-yaml	0.16.6	0.16.6
python3-ruamel-yaml- clib	0.1.2	0.1.2
python3-setools	4.4.1	4.4.1
python3-setuptools	59.6.0	59.6.0
python3-setuptools- wheel	59.6.0	59.6.0
python3-six	1.15.0	1.15.0
python3-systemd	235	235

Paquete	AMI	VMware OVA
python3-urllib3	1.25.10	1.25.10
python3-wcwidth	0.2.5	0.2.5
python-chevron	0.13.1	
python-srpm-macros	3.9	3.9
quota	4.06	4.06
quota-nls	4.06	4.06
readline	8.1	8.1
rng-tools	6.14	6.14
rootfiles	8.1	8.1
rpcbind	1.2.6	1.2.6
rpm	4.16.1.3	4.16.1.3
rpm-build-libs	4.16.1.3	4.16.1.3
rpm-libs	4.16.1.3	4.16.1.3
rpm-plugin-selinux	4.16.1.3	4.16.1.3
rpm-plugin-systemd-inhibit	4.16.1.3	4.16.1.3
rpm-sign-libs	4.16.1.3	4.16.1.3
rsync	3.2.6	3.2.6
rust-srpm-macros	21	21
sbsigntools	0.9.4	0.9.4
screen	4.8.0	4.8.0

Paquete	AMI	VMware OVA
sed	4.8	4.8
selinux-policy	38.1.45	38.1.45
selinux-policy-targeted	38.1.45	38.1.45
setup	2.13.7	2.13.7
shadow-utils	4.9	4.9
slang	2.3.2	2.3.2
sqlite-libs	3.40.0	3.40.0
sssd-client	2.9.4	2.9.4
sssd-common	2.9.4	
sssd-kcm	2.9.4	
sssd-nfs-idmap	2.9.4	
strace	6.8	6.8
sudo	1.9.15	1.9.15
sysctl-defaults	1.0	1.0
sysstat	12.5.6	12.5.6
systemd	252.23	252.23
systemd-libs	252.23	252.23
systemd-networkd	252.23	252.23
systemd-pam	252.23	252.23
systemd-resolved	252.23	252.23

Paquete	AMI	VMware OVA
systemd-udev	252.23	252.23
system-release	2023.6.20241031	2023.6.20241031
systemtap-runtime	4.8	4.8
tar	1.34	1.34
tbb	2020.3	2020.3
tcpdump	4.99.1	4.99.1
tcsh	6.24.07	6.24.07
time	1.9	1.9
traceroute	2.1.3	2.1.3
tzdata	2024a	2024a
unzip	6.0	6.0
update-motd	2.2	2.2
userspace-rcu	0.12.1	0.12.1
util-linux	2.37.4	2.37.4
util-linux-core	2.37.4	2.37.4
vim-common	9.0.2153	9.0.2153
vim-data	9.0.2153	9.0.2153
vim-enhanced	9.0.2153	9.0.2153
vim-filesystem	9.0.2153	9.0.2153
vim-minimal	9.0.2153	9.0.2153

Paquete	AMI	VMware OVA
wget	1.21.3	1.21.3
which	2.21	2.21
words	3.0	3.0
xfsdump	3.1.11	3.1.11
xfspgrog	5.18.0	5.18.0
xmlsec1		1.2.33
xmlsec1-openssl		1.2.33
xxd	9.0.2153	9.0.2153
xxhash-libs	0.8.0	0.8.0
xz	5.2.5	5.2.5
xz-libs	5.2.5	5.2.5
yum	4.14.0	4.14.0
zip	3.0	3.0
zlib	1.2.11	1.2.11
zram-generator	1.1.2	
zram-generator-def aults	1.1.2	
zstd	1.5.5	1.5.5

Comparación de los paquetes instalados en la AMI estándar de Amazon Linux 2023 con la imagen Hyper-V de AL2023

Comparación entre los RPM presentes en el AMI estándar AL2023 y los RPM presentes en la imagen Hyper-V de AL2023

Paquete	AMI	VHDX de Hyper-V
acl	2.3.1	2.3.1
acpid	2.0.32	
alternatives	1.15	1.15
amazon-chrony-config	4.3	
amazon-ec2-net-utils	2.4.1	
amazon-linux-onprem		1.2
amazon-linux-repo-cdn		2023.4.20240319
amazon-linux-repo-s3	2023.4.20240319	
amazon-linux-sb-keys	2023.1	2023.1
amazon-onprem-network		1.2
amazon-rpm-config	228	228
amazon-ssm-agent	3.2.2303.0	3.2.2303.0
at	3.1.23	3.1.23
attr	2.5.1	2.5.1
audit	3.0.6	3.0.6

Paquete	AMI	VHDX de Hyper-V
audit-libs	3.0.6	3.0.6
aws-cfn-bootstrap	2.0	
awscli-2	2.14.5	2.14.5
basesystem	11	11
bash	5.2.15	5.2.15
bash-completion	2.11	2.11
bc	1.07.1	1.07.1
bind-libs	9.16.48	9.16.48
bind-license	9.16.48	9.16.48
bind-utils	9.16.48	9.16.48
binutils	2.39	2.39
boost-filesystem	1.75.0	1.75.0
boost-system	1.75.0	1.75.0
boost-thread	1.75.0	1.75.0
bzip2	1.0.8	1.0.8
bzip2-libs	1.0.8	1.0.8
ca-certificates	2023.2.64	2023.2.64
c-ares	1.19.0	
checkpolicy	3.4	3.4
chkconfig	1.15	1.15

Paquete	AMI	VHDX de Hyper-V
chrony	4.3	4.3
cloud-init	22.2.2	22.2.2
cloud-init-cfg-ec2	22.2.2	
cloud-init-cfg-onpre		22.2.2
cloud-utils-growpart	0.31	0.31
coreutils	8.32	8.32
coreutils-common	8.32	8.32
cpio	2.13	2.13
cracklib	2.9.6	2.9.6
cracklib-dicts	2.9.6	2.9.6
crontabs	1.11	1.11
crypto-policies	20220428	20220428
crypto-policies-scripts	20220428	20220428
cryptsetup	2.6.1	2.6.1
cryptsetup-libs	2.6.1	2.6.1
curl-minimal	8.5.0	8.5.0
cyrus-sasl-lib	2.1.27	2.1.27
cyrus-sasl-plain	2.1.27	2.1.27
dbus	1.12.28	1.12.28

Paquete	AMI	VHDX de Hyper-V
dbus-broker	32	32
dbus-common	1.12.28	1.12.28
dbus-libs	1.12.28	1.12.28
device-mapper	1.02.185	1.02.185
device-mapper-libs	1.02.185	1.02.185
diffutils	3.8	3.8
dnf	4.14.0	4.14.0
dnf-data	4.14.0	4.14.0
dnf-plugin-release-notification	1.2	1.2
dnf-plugins-core	4.3.0	4.3.0
dnf-plugin-support-info	1.2	1.2
dnf-utils	4.3.0	4.3.0
dosfstools	4.2	4.2
dracut	055	055
dracut-config-ec2	3.0	
dracut-config-generic	055	055
dwz	0.14	0.14
dyninst	10.2.1	10.2.1

Paquete	AMI	VHDX de Hyper-V
e2fsprogs	1.46.5	1.46.5
e2fsprogs-libs	1.46.5	1.46.5
ec2-hibinit-agent	1.0.8	
ec2-instance-connect	1.1	
ec2-instance-connect-selinux	1.1	
ec2-utils	2.2.0	
ed	1.14.2	1.14.2
efi-filesystem	5	5
efi-srpm-macros	5	5
efivar	38	38
efivar-libs	38	38
elfutils-debuginfod-client	0.188	0.188
elfutils-default-yama-scope	0.188	0.188
elfutils-libelf	0.188	0.188
elfutils-libs	0.188	0.188
ethtool	5.15	5.15
expat	2.5.0	2.5.0
file	5.39	5.39

Paquete	AMI	VHDX de Hyper-V
file-libs	5.39	5.39
filesystem	3.14	3.14
findutils	4.8.0	4.8.0
fonts-srpm-macros	2.0.5	2.0.5
fstrm	0.6.1	0.6.1
fuse-libs	2.9.9	2.9.9
gawk	5.1.0	5.1.0
gdbm-libs	1.19	1.19
gdisk	1.0.8	1.0.8
gettext	0.21	0.21
gettext-libs	0.21	0.21
ghc-srpm-macros	1.5.0	1.5.0
glib2	2.74.7	2.74.7
glibc	2.34	2.34
glibc-all-langpacks	2.34	2.34
glibc-common	2.34	2.34
glibc-gconv-extra	2.34	2.34
glibc-locale-source	2.34	2.34
gmp	6.2.1	6.2.1
gnupg2-minimal	2.3.7	2.3.7

Paquete	AMI	VHDX de Hyper-V
gnutls	3.8.0	3.8.0
go-srpm-macros	3.2.0	3.2.0
gpgme	1.15.1	1.15.1
gpm-libs	1.20.7	1.20.7
grep	3.8	3.8
groff-base	1.22.4	1.22.4
grub2-common	2.06	2.06
grub2-efi-x64-ec2	2.06	2.06
grub2-pc		2.06
grub2-pc-modules	2.06	2.06
grub2-tools	2.06	2.06
grub2-tools-minimal	2.06	2.06
grubby	8.40	8.40
gssproxy	0.8.4	0.8.4
gzip	1.12	1.12
hostname	3.23	3.23
hunspell	1.7.0	1.7.0
hunspell-en	0.20140811.1	0.20140811.1
hunspell-en-GB	0.20140811.1	0.20140811.1
hunspell-en-US	0.20140811.1	0.20140811.1

Paquete	AMI	VHDX de Hyper-V
hunspell-filesystem	1.7.0	1.7.0
hwdata	0.353	0.353
hyperv-daemons		0
hyperv-daemons-lic ense		0
hypervfcopyd		0
hypervkvpd		0
hyperv-tools		0
hypervvssd		0
info	6.7	6.7
inih	49	49
initscripts	10.09	10.09
iproute	5.10.0	5.10.0
iputils	20210202	20210202
irqbalance	1.9.0	1.9.0
jansson	2.14	2.14
jitterentropy	3.4.1	3.4.1
jq	1.7.1	1.7.1
json-c	0.14	0.14
kbd	2.4.0	2.4.0
kbd-misc	2.4.0	2.4.0

Paquete	AMI	VHDX de Hyper-V
kernel	6.1.79	6.1.79
kernel-livepatch-r epo-cdn		2023.4.20240319
kernel-livepatch-r epo-s3	2023.4.20240319	
kernel-modules-extra		6.1.79
kernel-modules-ext ra-common		6.1.79
kernel-srpm-macros	1.0	1.0
kernel-tools	6.1.79	6.1.79
keyutils	1.6.3	1.6.3
keyutils-libs	1.6.3	1.6.3
kmod	29	29
kmod-libs	29	29
kpatch-runtime	0.9.7	0.9.7
krb5-libs	1.21	1.21
less	608	608
libacl	2.3.1	2.3.1
libaio	0.3.111	0.3.111
libarchive	3.5.3	3.5.3
libargon2	20171227	20171227

Paquete	AMI	VHDX de Hyper-V
libassuan	2.5.5	2.5.5
libattr	2.5.1	2.5.1
libbasicobjects	0.1.1	0.1.1
libblkid	2.37.4	2.37.4
libcap	2.48	2.48
libcap-ng	0.8.2	0.8.2
libcbor	0.7.0	0.7.0
libcollection	0.7.0	0.7.0
libcom_err	1.46.5	1.46.5
libcomps	0.1.20	0.1.20
libconfig	1.7.2	1.7.2
libcurl-minimal	8.5.0	8.5.0
libdb	5.3.28	5.3.28
libdhash	0.5.0	
libdnf	0.69.0	0.69.0
libeconf	0.4.0	0.4.0
libedit	3.1	3.1
libev	4.33	4.33
libevent	2.1.12	2.1.12
libfdisk	2.37.4	2.37.4

Paquete	AMI	VHDX de Hyper-V
libffi	3.4.4	3.4.4
libfido2	1.10.0	1.10.0
libgcc	11.4.1	11.4.1
libgcrypt	1.10.2	1.10.2
libgomp	11.4.1	11.4.1
libgpg-error	1.42	1.42
libibverbs	48.0	48.0
libidn2	2.3.2	2.3.2
libini_config	1.3.1	1.3.1
libkcapi	1.4.0	1.4.0
libkcapi-hmacalc	1.4.0	1.4.0
libldb	2.6.2	
libmaxminddb	1.5.2	1.5.2
libmetalink	0.1.3	0.1.3
libmnl	1.0.4	1.0.4
libmodulemd	2.13.0	2.13.0
libmount	2.37.4	2.37.4
libnfsidmap	2.5.4	2.5.4
libnghttp2	1.57.0	1.57.0
libnl3	3.5.0	3.5.0

Paquete	AMI	VHDX de Hyper-V
libpath_utils	0.2.1	0.2.1
libpcap	1.10.1	1.10.1
libpipeline	1.5.3	1.5.3
libpkgconf	1.8.0	1.8.0
libpsl	0.21.1	0.21.1
libpwquality	1.4.4	1.4.4
libref_array	0.1.5	0.1.5
librepo	1.14.5	1.14.5
libreport-filessystem	2.15.2	2.15.2
libseccomp	2.5.3	2.5.3
libselinux	3.4	3.4
libselinux-utils	3.4	3.4
libsemanage	3.4	3.4
libsepol	3.4	3.4
libsigsegv	2.13	2.13
libsmartcols	2.37.4	2.37.4
libsolv	0.7.22	0.7.22
libss	1.46.5	1.46.5
libsss_certmap	2.9.4	
libsss_idmap	2.9.4	2.9.4

Paquete	AMI	VHDX de Hyper-V
libsss_nss_idmap	2.9.4	2.9.4
libsss_sudo	2.9.4	
libstdc++	11.4.1	11.4.1
libstoragemgmt	1.9.4	1.9.4
libtalloc	2.3.4	
libtasn1	4.19.0	4.19.0
libtdb	1.4.7	
libtevent	0.13.0	
libtextstyle	0.21	0.21
libtirpc	1.3.3	1.3.3
libunistring	0.9.10	0.9.10
libuser	0.63	0.63
libutempter	1.2.1	1.2.1
libuuid	2.37.4	2.37.4
libuv	1.47.0	1.47.0
libverto	0.3.2	0.3.2
libverto-libev	0.3.2	0.3.2
libxcrypt	4.4.33	4.4.33
libxml2	2.10.4	2.10.4
libyaml	0.2.5	0.2.5

Paquete	AMI	VHDX de Hyper-V
libzstd	1.5.5	1.5.5
lm_sensors-libs	3.6.0	3.6.0
lmdb-libs	0.9.29	0.9.29
logrotate	3.20.1	3.20.1
lsof	4.94.0	4.94.0
lua-libs	5.4.4	5.4.4
lua-srpm-macros	1	1
lz4-libs	1.9.4	1.9.4
man-db	2.9.3	2.9.3
man-pages	5.10	5.10
microcode_ctl	2.1	2.1
mpfr	4.1.0	4.1.0
nano	5.8	5.8
ncurses	6.2	6.2
ncurses-base	6.2	6.2
ncurses-libs	6.2	6.2
nettle	3.8	3.8
net-tools	2.0	2.0
newt	0.52.21	0.52.21
nfs-utils	2.5.4	2.5.4

Paquete	AMI	VHDX de Hyper-V
npth	1.6	1.6
nspr	4.35.0	4.35.0
nss	3.90.0	3.90.0
nss-softokn	3.90.0	3.90.0
nss-softokn-freebl	3.90.0	3.90.0
nss-sysinit	3.90.0	3.90.0
nss-util	3.90.0	3.90.0
ntsysv	1.15	1.15
numactl-libs	2.0.14	2.0.14
ocaml-srpm-macros	6	6
oniguruma	6.9.7.1	6.9.7.1
openblas-srpm-macros	2	2
openldap	2.4.57	2.4.57
openssh	8.7p1	8.7p1
openssh-clients	8.7p1	8.7p1
openssh-server	8.7p1	8.7p1
openssl	3.0.8	3.0.8
openssl-libs	3.0.8	3.0.8
openssl-pkcs11	0.4.12	0.4.12
os-prober	1.77	1.77

Paquete	AMI	VHDX de Hyper-V
p11-kit	0.24.1	0.24.1
p11-kit-trust	0.24.1	0.24.1
package-notes-srpm-macros	0.4	0.4
pam	1.5.1	1.5.1
parted	3.4	3.4
passwd	0.80	0.80
pciutils	3.7.0	3.7.0
pciutils-libs	3.7.0	3.7.0
pcre2	10.40	10.40
pcre2-syntax	10.40	10.40
perl-Carp	1.50	1.50
perl-Class-Struct	0.66	0.66
perl-constant	1.33	1.33
perl-DynaLoader	1.47	1.47
perl-Encode	3.15	3.15
perl-Errno	1.30	1.30
perl-Exporter	5.74	5.74
perl-Fcntl	1.13	1.13
perl-File-Basename	2.85	2.85
perl-File-Path	2.18	2.18

Paquete	AMI	VHDX de Hyper-V
perl-File-stat	1.09	1.09
perl-File-Temp	0.231.100	0.231.100
perl-Getopt-Long	2.52	2.52
perl-Getopt-Std	1.12	1.12
perl-HTTP-Tiny	0.078	0.078
perl-if	0.60.800	0.60.800
perl-interpreter	5.32.1	5.32.1
perl-IO	1.43	1.43
perl-IPC-Open3	1.21	1.21
perl-libs	5.32.1	5.32.1
perl-MIME-Base64	3.16	3.16
perl-mro	1.23	1.23
perl-overload	1.31	1.31
perl-overloading	0.02	0.02
perl-parent	0.238	0.238
perl-PathTools	3.78	3.78
perl-Pod-Escapes	1.07	1.07
perl-podlators	4.14	4.14
perl-Pod-Perldoc	3.28.01	3.28.01
perl-Pod-Simple	3.42	3.42

Paquete	AMI	VHDX de Hyper-V
perl-Pod-Usage	2.01	2.01
perl-POSIX	1.94	1.94
perl-Scalar-List-Utils	1.56	1.56
perl-SelectSaver	1.02	1.02
perl-Socket	2.032	2.032
perl-srpm-macros	1	1
perl-Storable	3.21	3.21
perl-subsv	1.03	1.03
perl-Symbol	1.08	1.08
perl-Term-ANSIColor	5.01	5.01
perl-Term-Cap	1,17	1,17
perl-Text-ParseWords	3.30	3.30
perl-Text-Tabs+Wrap	2021.0726	2021.0726
perl-Time-Local	1.300	1.300
perl-vars	1.05	1.05
pkgconf	1.8.0	1.8.0
pkgconf-m4	1.8.0	1.8.0
pkgconf-pkg-config	1.8.0	1.8.0
policycoreutils	3.4	3.4

Paquete	AMI	VHDX de Hyper-V
polycoreutils-python-utils	3.4	
popt	1.18	1.18
procps-ng	3.3.17	3.3.17
protobuf-c	1.4.1	1.4.1
psacct	6.6.4	6.6.4
psmisc	23.4	23.4
publicsuffix-list-dafsa	20240212	20240212
python3	3.9.16	3.9.16
python3-attrs	20.3.0	20.3.0
python3-audit	3.0.6	3.0.6
python3-awscrt	0.19.19	0.19.19
python3-babel	2.9.1	2.9.1
python3-cffi	1.14.5	1.14.5
python3-chardet	4.0.0	4.0.0
python3-colorama	0.4.4	0.4.4
python3-configobj	5.0.6	5.0.6
python3-cryptography	36.0.1	36.0.1
python3-daemon	2.3.0	
python3-dateutil	2.8.1	2.8.1

Paquete	AMI	VHDX de Hyper-V
python3-dbus	1.2.18	1.2.18
python3-distro	1.5.0	1.5.0
python3-dnf	4.14.0	4.14.0
python3-dnf-plugins-core	4.3.0	4.3.0
python3-docutils	0.16	0.16
python3-gpg	1.15.1	1.15.1
python3-hawkey	0.69.0	0.69.0
python3-idna	(2.10)	(2.10)
python3-jinja2	2.11.3	2.11.3
python3-jmespath	0.10.0	0.10.0
python3-jsonpatch	1.21	1.21
python3-jsonpointer	2.0	2.0
python3-jjsonschema	3.2.0	3.2.0
python3-libcomps	0.1.20	0.1.20
python3-libdnf	0.69.0	0.69.0
python3-libs	3.9.16	3.9.16
python3-libselinux	3.4	3.4
python3-libsemanage	3.4	3.4
python3-libstorage mgmt	1.9.4	1.9.4

Paquete	AMI	VHDX de Hyper-V
python3-lockfile	0.12.2	
python3-markupsafe	1.1.1	1.1.1
python3-netifaces	0.10.6	0.10.6
python3-oauthlib	3.0.2	3.0.2
python3-pip-wheel	21.3.1	21.3.1
python3-ply	3.11	3.11
python3-policycore utils	3.4	3.4
python3-prettytable	0.7.2	0.7.2
python3-prompt-too lkit	3.0.24	3.0.24
python3-pycparser	2.20	2.20
python3-pyrsistent	0.17.3	0.17.3
python3-pyserial	3.4	3.4
python3-pysocks	1.7.1	1.7.1
python3-pytz	2022.7.1	2022.7.1
python3-pyyaml	5.4.1	5.4.1
python3-requests	2.25.1	2.25.1
python3-rpm	4.16.1.3	4.16.1.3
python3-ruamel-yaml	0.16.6	0.16.6

Paquete	AMI	VHDX de Hyper-V
python3-ruamel-yaml-clib	0.1.2	0.1.2
python3-setuptools	4.4.1	4.4.1
python3-setuptools	59.6.0	59.6.0
python3-setuptools-wheel	59.6.0	59.6.0
python3-six	1.15.0	1.15.0
python3-systemd	235	235
python3-urllib3	1.25.10	1.25.10
python3-wcwidth	0.2.5	0.2.5
python-chevron	0.13.1	
python-srpm-macros	3.9	3.9
quota	4.06	4.06
quota-nls	4.06	4.06
readline	8.1	8.1
rng-tools	6.14	6.14
rootfiles	8.1	8.1
rpcbind	1.2.6	1.2.6
rpm	4.16.1.3	4.16.1.3
rpm-build-libs	4.16.1.3	4.16.1.3
rpm-libs	4.16.1.3	4.16.1.3

Paquete	AMI	VHDX de Hyper-V
rpm-plugin-selinux	4.16.1.3	4.16.1.3
rpm-plugin-systemd-inhibit	4.16.1.3	4.16.1.3
rpm-sign-libs	4.16.1.3	4.16.1.3
rsync	3.2.6	3.2.6
rust-srpm-macros	21	21
sbsigntools	0.9.4	0.9.4
screen	4.8.0	4.8.0
sed	4.8	4.8
selinux-policy	37.22	37.22
selinux-policy-targeted	37.22	37.22
setup	2.13.7	2.13.7
shadow-utils	4.9	4.9
slang	2.3.2	2.3.2
sqlite-libs	3.40.0	3.40.0
sssd-client	2.9.4	2.9.4
sssd-common	2.9.4	
sssd-kcm	2.9.4	
sssd-nfs-idmap	2.9.4	
strace	5.16	5.16

Paquete	AMI	VHDX de Hyper-V
sudo	1.9.14	1.9.14
sysctl-defaults	1.0	1.0
sysstat	12.5.6	12.5.6
systemd	252.16	252.16
systemd-libs	252.16	252.16
systemd-networkd	252.16	252.16
systemd-pam	252.16	252.16
systemd-resolved	252.16	252.16
systemd-udev	252.16	252.16
system-release	2023.4.20240319	2023.4.20240319
systemtap-runtime	4.8	4.8
tar	1.34	1.34
tbb	2020.3	2020.3
tcpdump	4.99.1	4.99.1
tcsh	6.24.07	6.24.07
time	1.9	1.9
traceroute	2.1.3	2.1.3
tzdata	2024a	2024a
unzip	6.0	6.0
update-motd	2.2	2.2

Paquete	AMI	VHDX de Hyper-V
userspace-rcu	0.12.1	0.12.1
util-linux	2.37.4	2.37.4
util-linux-core	2.37.4	2.37.4
vim-common	9.0.2153	9.0.2153
vim-data	9.0.2153	9.0.2153
vim-enhanced	9.0.2153	9.0.2153
vim-filesystem	9.0.2153	9.0.2153
vim-minimal	9.0.2153	9.0.2153
wget	1.21.3	1.21.3
which	2.21	2.21
words	3.0	3.0
xfsdump	3.1.11	3.1.11
xfsprogs	5.18.0	5.18.0
xxd	9.0.2153	9.0.2153
xxhash-libs	0.8.0	0.8.0
xz	5.2.5	5.2.5
xz-libs	5.2.5	5.2.5
yum	4.14.0	4.14.0
zip	3.0	3.0
zlib	1.2.11	1.2.11

Paquete	AMI	VHDX de Hyper-V
zram-generator	1.1.2	
zram-generator-def aults	1.1.2	
zstd	1.5.5	1.5.5

Identificación de instancias y versiones de Amazon Linux

Determinar qué distribución de Linux y qué versión de esa distribución tiene una imagen o instancia del sistema operativo puede ser importante. Amazon Linux ofrece mecanismos para identificar Amazon Linux a diferencia de otras distribuciones de Linux, así como para identificar qué versión de Amazon Linux es la imagen.

En esta sección se tratarán los diferentes métodos que se pueden utilizar y sus limitaciones, y se analizarán algunos ejemplos de uso.

Temas

- [Utilización del estándar os-release](#)
- [Archivos específicos de Amazon Linux](#)
- [Código de ejemplo para la detección del sistema operativo](#)

Utilización del estándar **os-release**

Amazon Linux cumple con el [estándar os-release](#) de identificación de distribuciones de Linux. Este archivo proporciona información legible por máquinas sobre la identificación del sistema operativo y la versión.

Note

El estándar establece que primero se intenta analizar `/etc/os-release`, seguido de `/usr/lib/os-release`. Debe procurarse seguir el estándar en cuanto a nombres de archivos y rutas.

Temas

- [Principales diferencias de identificación](#)
- [Tipos de campos: legibles por máquinas o legibles por humanos](#)
- [/etc/os-releaseEjemplos de](#)
- [Comparación con otras distribuciones](#)

Principales diferencias de identificación

`os-release` se encuentra en `/etc/os-release`, y si no está presente, en `/usr/lib/os-release`. Consulte el [estándar os-release](#) para obtener información completa.

La forma más fiable de determinar si una instancia ejecuta Amazon Linux es marcar el campo `ID` en `os-release`.

La forma más fiable de distinguir entre versiones es marcar el campo `VERSION_ID` en `os-release`:

- AMI de Amazon Linux: `VERSION_ID` contiene una versión basada en fechas (por ejemplo, `2018.03`)
- AL2: `VERSION_ID="2"`
- AL2023: `VERSION_ID="2023"`

Note

Recuerde que `VERSION_ID` es un campo legible por máquinas diseñado para uso programático, pero `PRETTY_NAME` está diseñado para mostrarse a los usuarios. Para obtener más información sobre los tipos de campos, consulte [the section called “Tipos de campos:”](#).

Tipos de campos: legibles por máquinas o legibles por humanos

El archivo `/etc/os-release` (o `/usr/lib/os-release` si `/etc/os-release` no existe) contiene dos tipos de campos: campos legibles por máquinas destinados al uso programático y campos legibles por humanos para mostrarse a los usuarios.

Campos legibles por máquinas

Estos campos utilizan formatos estandarizados y están diseñados para ser procesados por scripts, administradores de paquetes y otras herramientas automatizadas. Solo contienen minúsculas, números y signos de puntuación limitados (puntos, guiones bajos y guiones).

- `ID`: identificador del sistema operativo. Amazon Linux usa `amzn` en todas las versiones, lo que lo distingue de otras distribuciones como Debian (`debian`), Ubuntu (`ubuntu`) o Fedora (`fedora`)

- `VERSION_ID`: versión del sistema operativo para uso programático (por ejemplo, `2023`)
- `ID_LIKE`: lista separada por espacios de distribuciones relacionadas (por ejemplo, `fedora`)
- `VERSION_CODENAME`: nombre de código de lanzamiento para scripts (por ejemplo, `karoo`)
- `VARIANT_ID`: identificador de variantes para decisiones programáticas
- `BUILD_ID`: identificador de compilación para imágenes del sistema
- `IMAGE_ID`: identificador de imagen para entornos en contenedores
- `PLATFORM_ID`: identificador de la plataforma (por ejemplo, `platform:al2023`)

Campos legibles por humanos

Estos campos están destinados a mostrarse a los usuarios y pueden contener espacios, mayúsculas y minúsculas, y texto descriptivo. Deben utilizarse al presentar información sobre el sistema operativo en interfaces de usuario.

- `NAME`: nombre del sistema operativo que se mostrará (por ejemplo, `Amazon Linux`)
- `PRETTY_NAME`: nombre completo del sistema operativo con la versión que se mostrará (por ejemplo, `Amazon Linux 2023.8.20250721`)
- `VERSION`: información sobre la versión adecuada para la presentación al usuario
- `VARIANT`: nombre de variante o edición que se mostrará (por ejemplo, `Server Edition`)

Otros campos de información

Estos campos proporcionan metadatos adicionales sobre el sistema operativo:

- `HOME_URL`: URL de la página de inicio del proyecto
- `DOCUMENTATION_URL`: URL de la documentación
- `SUPPORT_URL`: URL de información de soporte
- `BUG_REPORT_URL`: URL de notificación de errores
- `VENDOR_NAME`: nombre del proveedor
- `VENDOR_URL`: URL del proveedor
- `SUPPORT_END`: fecha de fin de soporte en el formato `AAAA-MM-DD`
- `CPE_NAME`: identificador de enumeración de plataforma común

- `ANSI_COLOR`: código de color ANSI para la pantalla del terminal

Al escribir scripts o aplicaciones que necesiten identificar Amazon Linux mediante programación, utilice campos legibles por máquinas como `ID` y `VERSION_ID`. Al mostrar la información del sistema operativo a los usuarios, utilice campos legibles por humanos, como `PRETTY_NAME`.

`/etc/os-release` Ejemplos de

El contenido del archivo `/etc/os-release` varía entre las versiones de Amazon Linux:

AL2023

```
[ec2-user ~]$ cat /etc/os-release
```

```
NAME="Amazon Linux"
VERSION="2023"
ID="amzn"
ID_LIKE="fedora"
VERSION_ID="2023"
PLATFORM_ID="platform:al2023"
PRETTY_NAME="Amazon Linux 2023.8.20250721"
ANSI_COLOR="0;33"
CPE_NAME="cpe:2.3:o:amazon:amazon_linux:2023"
HOME_URL="https://aws.amazon.com/linux/amazon-linux-2023/"
DOCUMENTATION_URL="https://docs.aws.amazon.com/linux/"
SUPPORT_URL="https://aws.amazon.com/premiumsupport/"
BUG_REPORT_URL="https://github.com/amazonlinux/amazon-linux-2023"
VENDOR_NAME="AWS"
VENDOR_URL="https://aws.amazon.com/"
SUPPORT_END="2029-06-30"
```

AL2

```
[ec2-user ~]$ cat /etc/os-release
```

```
NAME="Amazon Linux"
VERSION="2"
ID="amzn"
ID_LIKE="centos rhel fedora"
VERSION_ID="2"
```

```
PRETTY_NAME="Amazon Linux 2"  
ANSI_COLOR="0;33"  
CPE_NAME="cpe:2.3:o:amazon:amazon_linux:2"  
HOME_URL="https://amazonlinux.com/"  
SUPPORT_END="2026-06-30"
```

Amazon Linux AMI

```
[ec2-user ~]$ cat /etc/os-release
```

```
NAME="Amazon Linux AMI"  
VERSION="2018.03"  
ID="amzn"  
ID_LIKE="rhel fedora"  
VERSION_ID="2018.03"  
PRETTY_NAME="Amazon Linux AMI 2018.03"  
ANSI_COLOR="0;33"  
CPE_NAME="cpe:/o:amazon:linux:2018.03:ga"  
HOME_URL="http://aws.amazon.com/amazon-linux-ami/"
```

Comparación con otras distribuciones

Para entender cómo Amazon Linux encaja en el ecosistema Linux más amplio, compare su formato `/etc/os-release` con el de otras distribuciones principales:

Fedora

```
[ec2-user ~]$ cat /etc/os-release
```

```
NAME="Fedora Linux"  
VERSION="42 (Container Image)"  
RELEASE_TYPE=stable  
ID=fedora  
VERSION_ID=42  
VERSION_CODENAME=""  
PLATFORM_ID="platform:f42"  
PRETTY_NAME="Fedora Linux 42 (Container Image)"  
ANSI_COLOR="0;38;2;60;110;180"  
LOGO=fedora-logo-icon  
CPE_NAME="cpe:/o:fedoraproject:fedora:42"
```

```
DEFAULT_HOSTNAME="fedora"
HOME_URL="https://fedoraproject.org/"
DOCUMENTATION_URL="https://docs.fedoraproject.org/en-US/fedora/f42/system-
administrators-guide/"
SUPPORT_URL="https://ask.fedoraproject.org/"
BUG_REPORT_URL="https://bugzilla.redhat.com/"
REDHAT_BUGZILLA_PRODUCT="Fedora"
REDHAT_BUGZILLA_PRODUCT_VERSION=42
REDHAT_SUPPORT_PRODUCT="Fedora"
REDHAT_SUPPORT_PRODUCT_VERSION=42
SUPPORT_END=2026-05-13
VARIANT="Container Image"
VARIANT_ID=container
```

Debian

```
[ec2-user ~]$ cat /etc/os-release
```

```
PRETTY_NAME="Debian GNU/Linux 12 (bookworm)"
NAME="Debian GNU/Linux"
VERSION_ID="12"
VERSION="12 (bookworm)"
VERSION_CODENAME=bookworm
ID=debian
HOME_URL="https://www.debian.org/"
SUPPORT_URL="https://www.debian.org/support"
BUG_REPORT_URL="https://bugs.debian.org/"
```

Ubuntu

```
[ec2-user ~]$ cat /etc/os-release
```

```
PRETTY_NAME="Ubuntu 24.04.2 LTS"
NAME="Ubuntu"
VERSION_ID="24.04"
VERSION="24.04.2 LTS (Noble Numbat)"
VERSION_CODENAME=noble
ID=ubuntu
ID_LIKE=debian
HOME_URL="https://www.ubuntu.com/"
SUPPORT_URL="https://help.ubuntu.com/"
BUG_REPORT_URL="https://bugs.launchpad.net/ubuntu/"
```



```
PRIVACY_POLICY_URL="https://www.ubuntu.com/legal/terms-and-policies/privacy-policy"
UBUNTU_CODENAME=noble
LOGO=ubuntu-logo
```

Observe cómo los campos legibles por máquinas proporcionan una identificación coherente en todas las distribuciones:

- ID: identifica de forma exclusiva el sistema operativo: `amzn` para Amazon Linux, `fedora` para Fedora, `debian` para Debian, `ubuntu` para Ubuntu
- ID_LIKE: muestra las relaciones de distribución: Amazon Linux usa `fedora` (AL2023) o `centos` `rhel` `fedora` (AL2), mientras que Ubuntu usa `debian` para mostrar su herencia de Debian
- VERSION_ID: proporciona información sobre la versión analizable por máquinas: `2023` para AL2023, `42` para Fedora, `12` para Debian, `24.04` para Ubuntu

Por el contrario, los campos legibles por humanos están diseñados para mostrarlos a los usuarios:

- NAME: nombre del sistema operativo fácil de recordar: `Amazon Linux`, `Fedora Linux`, `Debian GNU/Linux`, `Ubuntu`
- PRETTY_NAME: nombre completo que se mostrará con la versión: `Amazon Linux 2023.8.20250721`, `Fedora Linux 42 (Container Image)`, `Debian GNU/Linux 12 (bookworm)`, `Ubuntu 24.04.2 LTS`
- VERSION: versión legible por humanos con contexto adicional, como nombres de código o tipos de versiones

Al escribir scripts multiplataforma, utilice siempre los campos legibles por máquinas (ID, VERSION_ID, ID_LIKE) para la lógica y las decisiones, y utilice los campos legibles por humanos (PRETTY_NAME, NAME) solo para mostrar información a los usuarios.

Archivos específicos de Amazon Linux

Hay algunos archivos específicos de Amazon Linux que se pueden utilizar para identificar Amazon Linux y su versión. El nuevo código debe usar el estándar [/etc/os-release](#) para que sea compatible con la distribución cruzada. No se recomienda el uso de ningún archivo específico de Amazon Linux.

Temas

- [El archivo `/etc/system-release`](#)
- [Archivo de identificación de imagen](#)
- [Ejemplos de archivos específicos de Amazon Linux](#)

El archivo `/etc/system-release`

Amazon Linux contiene un archivo `/etc/system-release` que especifica la versión actual instalada. Este archivo se actualiza mediante gestores de paquetes y, en Amazon Linux, forma parte del paquete `system-release`. Si bien otras distribuciones como Fedora también tienen este archivo, no está presente en las distribuciones basadas en Debian, como Ubuntu.

Note

El archivo `/etc/system-release` contiene una cadena legible por humanos y no debe usarse mediante programación para identificar un sistema operativo o una versión. En su lugar, utilice los campos legibles por máquinas que aparecen en `/etc/os-release` (o `/usr/lib/os-release` si `/etc/os-release` no existe).

Amazon Linux también contiene una versión legible por máquinas de `/etc/system-release` que sigue la especificación de Common Platform Enumeration (CPE) del archivo `/etc/system-release-cpe`.

Archivo de identificación de imagen

Cada imagen de Amazon Linux contiene un archivo `/etc/image-id` único que proporciona información adicional sobre la imagen original generada por el equipo de Amazon Linux. Este archivo es específico de Amazon Linux y no se encuentra en otras distribuciones de Linux como Debian, Ubuntu o Fedora. Este archivo contiene la siguiente información acerca de la imagen:

- `image_name`, `image_version`, `image_arch`: valores de la receta de creación que se utilizaron para construir la imagen.
- `image_stamp`: valor hexadecimal aleatorio exclusivo que se genera durante la creación de la imagen.
- `image_date`: la hora UTC de creación de la imagen, con el formato `AAAAMMDDhhmmss`
- `recipe_name`, `recipe_id`: nombre e ID de la receta de creación que se utilizaron para construir la imagen.

Ejemplos de archivos específicos de Amazon Linux

En las siguientes secciones se proporcionan ejemplos de los archivos de identificación específicos de Amazon Linux para cada versión principal de Amazon Linux.

Note

En cualquier código real, `/usr/lib/os-release` debe usarse si el archivo `/etc/os-release` no existe.

AL2023

Los siguientes ejemplos muestran los archivos de identificación de AL2023.

Ejemplo de `/etc/image-id` para AL2023:

```
[ec2-user ~]$ cat /etc/image-id
```

```
image_name="al2023-container"  
image_version="2023"  
image_arch="x86_64"  
image_file="al2023-container-2023.8.20250721.2-x86_64"  
image_stamp="822b-1a9e"  
image_date="20250719211531"  
recipe_name="al2023 container"  
recipe_id="89b25f7b-be82-2215-a8eb-6e63-0830-94ea-658d41c4"
```

Ejemplo de `/etc/system-release` para AL2023:

```
[ec2-user ~]$ cat /etc/system-release
```

```
Amazon Linux release 2023.8.20250721 (Amazon Linux)
```

AL2

Los siguientes ejemplos muestran los archivos de identificación de AL2.

Ejemplo de `/etc/image-id` para AL2:

```
[ec2-user ~]$ cat /etc/image-id
```

```
image_name="amzn2-container-raw"  
image_version="2"  
image_arch="x86_64"  
image_file="amzn2-container-raw-2.0.20250721.2-x86_64"  
image_stamp="4126-16ad"  
image_date="20250721225801"  
recipe_name="amzn2 container"  
recipe_id="948422df-a4e6-5fc8-ba89-ef2e-0e1f-e1bb-16f84087"
```

Ejemplo de `/etc/system-release` para AL2:

```
[ec2-user ~]$ cat /etc/system-release
```

```
Amazon Linux release 2 (Karoo)
```

AMI de Amazon Linux

Los siguientes ejemplos muestran los archivos de identificación de la AMI de Amazon Linux.

Ejemplo de `/etc/image-id` para la AMI de Amazon Linux:

```
[ec2-user ~]$ cat /etc/image-id
```

```
image_name="amzn-container-minimal"  
image_version="2018.03"  
image_arch="x86_64"  
image_file="amzn-container-minimal-2018.03.0.20231218.0-x86_64"  
image_stamp="407d-5ef3"  
image_date="20231218203210"  
recipe_name="amzn container"  
recipe_id="b1e7635e-14e3-dd57-b1ab-7351-edd0-d9e0-ca6852ea"
```

Ejemplo de `/etc/system-release` para la AMI de Amazon Linux:

```
[ec2-user ~]$ cat /etc/system-release
```

Código de ejemplo para la detección del sistema operativo

Los siguientes ejemplos muestran cómo detectar mediante programación el sistema operativo y la versión mediante el archivo `/etc/os-release` (o `/usr/lib/os-release` si `/etc/os-release` no existe). Estos ejemplos muestran cómo distinguir entre Amazon Linux y otras distribuciones, así como cómo utilizar el campo `ID_LIKE` para determinar las familias de distribución.

El siguiente script está implementado en varios lenguajes de programación diferentes y cada implementación producirá el mismo resultado.

Shell

```
#!/bin/bash

# Function to get a specific field from os-release file
get_os_release_field() {
    local field="$1"
    local os_release_file

    # Find the os-release file
    if [ -f /etc/os-release ]; then
        os_release_file='/etc/os-release'
    elif [ -f /usr/lib/os-release ]; then
        os_release_file='/usr/lib/os-release'
    else
        echo "Error: os-release file not found" >&2
        return 1
    fi

    # Source the file in a subshell and return the requested field.
    #
    # A subshell means that variables from os-release are only available
    # within the subshell, and the main script environment remains clean.
    (
        . "$os_release_file"
        eval "echo \"\${$field}\""
    )
}
```

```
is_amazon_linux() {
    [ "$(get_os_release_field ID)" = "amzn" ]
}

is_fedora() {
    [ "$(get_os_release_field ID)" = "fedora" ]
}

is_ubuntu() {
    [ "$(get_os_release_field ID)" = "ubuntu" ]
}

is_debian() {
    [ "$(get_os_release_field ID)" = "debian" ]
}

# Function to check if this is like Fedora (includes Amazon Linux, CentOS, RHEL,
# etc.)
is_like_fedora() {
    local id="$(get_os_release_field ID)"
    local id_like="$(get_os_release_field ID_LIKE)"
    [ "$id" = "fedora" ] || [[ "$id_like" == *"fedora"* ]]
}

# Function to check if this is like Debian (includes Ubuntu and derivatives)
is_like_debian() {
    local id="$(get_os_release_field ID)"
    local id_like="$(get_os_release_field ID_LIKE)"
    [ "$id" = "debian" ] || [[ "$id_like" == *"debian"* ]]
}

# Get the main fields we'll use multiple times
ID="$(get_os_release_field ID)"
VERSION_ID="$(get_os_release_field VERSION_ID)"
PRETTY_NAME="$(get_os_release_field PRETTY_NAME)"
ID_LIKE="$(get_os_release_field ID_LIKE)"

echo "Operating System Detection Results:"
echo "====="
echo "Is Amazon Linux: $(is_amazon_linux && echo YES || echo NO)"
echo "Is Fedora: $(is_fedora && echo YES || echo NO)"
echo "Is Ubuntu: $(is_ubuntu && echo YES || echo NO)"
echo "Is Debian: $(is_debian && echo YES || echo NO)"
echo "Is like Fedora: $(is_like_fedora && echo YES || echo NO)"
```

```

echo "Is like Debian: $(is_like_debian && echo YES || echo NO)"
echo
echo "Detailed OS Information:"
echo "======"
echo "ID: $ID"
echo "VERSION_ID: $VERSION_ID"
echo "PRETTY_NAME: $PRETTY_NAME"
[ -n "$ID_LIKE" ] && echo "ID_LIKE: $ID_LIKE"

# Amazon Linux specific information
if is_amazon_linux; then
    echo ""
    echo "Amazon Linux Version Details:"
    echo "======"
    case "$VERSION_ID" in
        2018.03)
            echo "Amazon Linux AMI (version 1)"
            ;;
        2)
            echo "Amazon Linux 2"
            ;;
        2023)
            echo "Amazon Linux 2023"
            ;;
        *)
            echo "Unknown Amazon Linux version: $VERSION_ID"
            ;;
    esac

    # Check for Amazon Linux specific files
    [ -f /etc/image-id ] && echo "Amazon Linux image-id file present"
fi

```

Python 3.7-3.9

```

#!/usr/bin/env python3

import os
import sys

def parse_os_release():
    """Parse the os-release file and return a dictionary of key-value pairs."""
    os_release_data = {}

```

```
# Try /etc/os-release first, then /usr/lib/os-release
for path in ['/etc/os-release', '/usr/lib/os-release']:
    if os.path.exists(path):
        try:
            with open(path, 'r') as f:
                for line in f:
                    line = line.strip()
                    if line and not line.startswith('#') and '=' in line:
                        key, value = line.split('=', 1)
                        # Remove quotes if present
                        value = value.strip('"\'')
                        os_release_data[key] = value
            return os_release_data
        except IOError:
            continue

print("Error: os-release file not found")
sys.exit(1)

def is_amazon_linux(os_data):
    """Check if this is Amazon Linux."""
    return os_data.get('ID') == 'amzn'

def is_fedora(os_data):
    """Check if this is Fedora."""
    return os_data.get('ID') == 'fedora'

def is_ubuntu(os_data):
    """Check if this is Ubuntu."""
    return os_data.get('ID') == 'ubuntu'

def is_debian(os_data):
    """Check if this is Debian."""
    return os_data.get('ID') == 'debian'

def is_like_fedora(os_data):
    """Check if this is like Fedora (includes Amazon Linux, CentOS, RHEL, etc.)."""
    if os_data.get('ID') == 'fedora':
        return True
    id_like = os_data.get('ID_LIKE', '')
    return 'fedora' in id_like

def is_like_debian(os_data):
```



```

    """Check if this is like Debian (includes Ubuntu and derivatives)."""
    if os_data.get('ID') == 'debian':
        return True
    id_like = os_data.get('ID_LIKE', '')
    return 'debian' in id_like

def main():
    # Parse os-release file
    os_data = parse_os_release()

    # Display results
    print("Operating System Detection Results:")
    print("=====")
    print(f"Is Amazon Linux: {'YES' if is_amazon_linux(os_data) else 'NO'}")
    print(f"Is Fedora: {'YES' if is_fedora(os_data) else 'NO'}")
    print(f"Is Ubuntu: {'YES' if is_ubuntu(os_data) else 'NO'}")
    print(f"Is Debian: {'YES' if is_debian(os_data) else 'NO'}")
    print(f"Is like Fedora: {'YES' if is_like_fedora(os_data) else 'NO'}")
    print(f"Is like Debian: {'YES' if is_like_debian(os_data) else 'NO'}")

    # Additional information
    print()
    print("Detailed OS Information:")
    print("=====")
    print(f"ID: {os_data.get('ID', '')}")
    print(f"VERSION_ID: {os_data.get('VERSION_ID', '')}")
    print(f"PRETTY_NAME: {os_data.get('PRETTY_NAME', '')}")
    if os_data.get('ID_LIKE'):
        print(f"ID_LIKE: {os_data.get('ID_LIKE')}")

    # Amazon Linux specific information
    if is_amazon_linux(os_data):
        print()
        print("Amazon Linux Version Details:")
        print("=====")
        version_id = os_data.get('VERSION_ID', '')
        if version_id == '2018.03':
            print("Amazon Linux AMI (version 1)")
        elif version_id == '2':
            print("Amazon Linux 2")
        elif version_id == '2023':
            print("Amazon Linux 2023")
        else:
            print(f"Unknown Amazon Linux version: {version_id}")

```

```
# Check for Amazon Linux specific files
if os.path.exists('/etc/image-id'):
    print("Amazon Linux image-id file present")

if __name__ == '__main__':
    main()
```

Python 3.10+

```
#!/usr/bin/env python3

import os
import sys
import platform

def is_amazon_linux(os_data):
    """Check if this is Amazon Linux."""
    return os_data.get('ID') == 'amzn'

def is_fedora(os_data):
    """Check if this is Fedora."""
    return os_data.get('ID') == 'fedora'

def is_ubuntu(os_data):
    """Check if this is Ubuntu."""
    return os_data.get('ID') == 'ubuntu'

def is_debian(os_data):
    """Check if this is Debian."""
    return os_data.get('ID') == 'debian'

def is_like_fedora(os_data):
    """Check if this is like Fedora (includes Amazon Linux, CentOS, RHEL, etc.)."""
    if os_data.get('ID') == 'fedora':
        return True
    id_like = os_data.get('ID_LIKE', '')
    return 'fedora' in id_like

def is_like_debian(os_data):
    """Check if this is like Debian (includes Ubuntu and derivatives)."""
    if os_data.get('ID') == 'debian':
        return True
```

```

    id_like = os_data.get('ID_LIKE', '')
    return 'debian' in id_like

def main():
    # Parse os-release file using the standard library function (Python 3.10+)
    try:
        os_data = platform.freedesktop_os_release()
    except OSError:
        print("Error: os-release file not found")
        sys.exit(1)

    # Display results
    print("Operating System Detection Results:")
    print("=====")
    print(f"Is Amazon Linux: {'YES' if is_amazon_linux(os_data) else 'NO'}")
    print(f"Is Fedora: {'YES' if is_fedora(os_data) else 'NO'}")
    print(f"Is Ubuntu: {'YES' if is_ubuntu(os_data) else 'NO'}")
    print(f"Is Debian: {'YES' if is_debian(os_data) else 'NO'}")
    print(f"Is like Fedora: {'YES' if is_like_fedora(os_data) else 'NO'}")
    print(f"Is like Debian: {'YES' if is_like_debian(os_data) else 'NO'}")

    # Additional information
    print()
    print("Detailed OS Information:")
    print("=====")
    print(f"ID: {os_data.get('ID', '')}")
    print(f"VERSION_ID: {os_data.get('VERSION_ID', '')}")
    print(f"PRETTY_NAME: {os_data.get('PRETTY_NAME', '')}")
    if os_data.get('ID_LIKE'):
        print(f"ID_LIKE: {os_data.get('ID_LIKE')}")

    # Amazon Linux specific information
    if is_amazon_linux(os_data):
        print()
        print("Amazon Linux Version Details:")
        print("=====")
        version_id = os_data.get('VERSION_ID', '')
        if version_id == '2018.03':
            print("Amazon Linux AMI (version 1)")
        elif version_id == '2':
            print("Amazon Linux 2")
        elif version_id == '2023':
            print("Amazon Linux 2023")
        else:

```

```

        print(f"Unknown Amazon Linux version: {version_id}")

    # Check for Amazon Linux specific files
    if os.path.exists('/etc/image-id'):
        print("Amazon Linux image-id file present")

if __name__ == '__main__':
    main()

```

Perl

```

#!/usr/bin/env perl

use strict;
use warnings;

# Function to parse the os-release file and return a hash of key-value pairs
sub parse_os_release {
    my %os_release_data;

    # Try /etc/os-release first, then /usr/lib/os-release
    my @paths = ('/etc/os-release', '/usr/lib/os-release');

    for my $path (@paths) {
        if (-f $path) {
            if (open(my $fh, '<', $path)) {
                while (my $line = <$fh>) {
                    chomp $line;
                    next if $line =~ /\s*$/ || $line =~ /\s*#/;

                    if ($line =~ /^(([^\s]+)=(.*)$)/) {
                        my ($key, $value) = ($1, $2);
                        # Remove quotes if present
                        $value =~ s/^[\'"]|[\']$//g;
                        $os_release_data{$key} = $value;
                    }
                }
                close($fh);
                return %os_release_data;
            }
        }
    }
}

```

```
    die "Error: os-release file not found\n";
}

# Function to check if this is Amazon Linux
sub is_amazon_linux {
    my %os_data = @_;
    return ($os_data{ID} // '') eq 'amzn';
}

# Function to check if this is Fedora
sub is_fedora {
    my %os_data = @_;
    return ($os_data{ID} // '') eq 'fedora';
}

# Function to check if this is Ubuntu
sub is_ubuntu {
    my %os_data = @_;
    return ($os_data{ID} // '') eq 'ubuntu';
}

# Function to check if this is Debian
sub is_debian {
    my %os_data = @_;
    return ($os_data{ID} // '') eq 'debian';
}

# Function to check if this is like Fedora (includes Amazon Linux, CentOS, RHEL,
etc.)
sub is_like_fedora {
    my %os_data = @_;
    return 1 if ($os_data{ID} // '') eq 'fedora';
    my $id_like = $os_data{ID_LIKE} // '';
    return $id_like =~ /fedora/;
}

# Function to check if this is like Debian (includes Ubuntu and derivatives)
sub is_like_debian {
    my %os_data = @_;
    return 1 if ($os_data{ID} // '') eq 'debian';
    my $id_like = $os_data{ID_LIKE} // '';
    return $id_like =~ /debian/;
}
```

```

# Main execution
my %os_data = parse_os_release();

# Display results
print "Operating System Detection Results:\n";
print "=====\n";
print "Is Amazon Linux: " . (is_amazon_linux(%os_data) ? "YES" : "NO") . "\n";
print "Is Fedora: " . (is_fedora(%os_data) ? "YES" : "NO") . "\n";
print "Is Ubuntu: " . (is_ubuntu(%os_data) ? "YES" : "NO") . "\n";
print "Is Debian: " . (is_debian(%os_data) ? "YES" : "NO") . "\n";
print "Is like Fedora: " . (is_like_fedora(%os_data) ? "YES" : "NO") . "\n";
print "Is like Debian: " . (is_like_debian(%os_data) ? "YES" : "NO") . "\n";
print "\n";

# Additional information
print "Detailed OS Information:\n";
print "=====\n";
print "ID: " . ($os_data{ID} // '') . "\n";
print "VERSION_ID: " . ($os_data{VERSION_ID} // '') . "\n";
print "PRETTY_NAME: " . ($os_data{PRETTY_NAME} // '') . "\n";
print "ID_LIKE: " . ($os_data{ID_LIKE} // '') . "\n" if $os_data{ID_LIKE};

# Amazon Linux specific information
if (is_amazon_linux(%os_data)) {
    print "\n";
    print "Amazon Linux Version Details:\n";
    print "=====\n";
    my $version_id = $os_data{VERSION_ID} // '';

    if ($version_id eq '2018.03') {
        print "Amazon Linux AMI (version 1)\n";
    } elsif ($version_id eq '2') {
        print "Amazon Linux 2\n";
    } elsif ($version_id eq '2023') {
        print "Amazon Linux 2023\n";
    } else {
        print "Unknown Amazon Linux version: $version_id\n";
    }

    # Check for Amazon Linux specific files
    if (-f '/etc/image-id') {
        print "Amazon Linux image-id file present\n";
    }
}

```

```
}
```

Cuando se ejecuta en sistemas diferentes, el script producirá el siguiente resultado:

AL2023

```
Operating System Detection Results:
=====
Is Amazon Linux: YES
Is Fedora: NO
Is Ubuntu: NO
Is Debian: NO
Is like Fedora: YES
Is like Debian: NO

Detailed OS Information:
=====
ID: amzn
VERSION_ID: 2023
PRETTY_NAME: Amazon Linux 2023.8.20250721
ID_LIKE: fedora

Amazon Linux Version Details:
=====
Amazon Linux 2023
Amazon Linux image-id file present
```

AL2

```
Operating System Detection Results:
=====
Is Amazon Linux: YES
Is Fedora: NO
Is Ubuntu: NO
Is Debian: NO
Is like Fedora: YES
Is like Debian: NO

Detailed OS Information:
=====
ID: amzn
VERSION_ID: 2
```

```
PRETTY_NAME: Amazon Linux 2
ID_LIKE: centos rhel fedora

Amazon Linux Version Details:
=====
Amazon Linux 2
Amazon Linux image-id file present
```

Amazon Linux AMI

```
Operating System Detection Results:
=====
Is Amazon Linux: YES
Is Fedora: NO
Is Ubuntu: NO
Is Debian: NO
Is like Fedora: YES
Is like Debian: NO

Detailed OS Information:
=====
ID: amzn
VERSION_ID: 2018.03
PRETTY_NAME: Amazon Linux AMI 2018.03
ID_LIKE: rhel fedora

Amazon Linux Version Details:
=====
Amazon Linux AMI (version 1)
Amazon Linux image-id file present
```

Ubuntu

```
Operating System Detection Results:
=====
Is Amazon Linux: NO
Is Fedora: NO
Is Ubuntu: YES
Is Debian: NO
Is like Fedora: NO
Is like Debian: YES

Detailed OS Information:
```



```
=====
ID: ubuntu
VERSION_ID: 24.04
PRETTY_NAME: Ubuntu 24.04.2 LTS
ID_LIKE: debian
```

Debian

```
Operating System Detection Results:
=====
Is Amazon Linux: NO
Is Fedora: NO
Is Ubuntu: NO
Is Debian: YES
Is like Fedora: NO
Is like Debian: YES

Detailed OS Information:
=====
ID: debian
VERSION_ID: 12
PRETTY_NAME: Debian GNU/Linux 12 (bookworm)
```

Fedora

```
Operating System Detection Results:
=====
Is Amazon Linux: NO
Is Fedora: YES
Is Ubuntu: NO
Is Debian: NO
Is like Fedora: YES
Is like Debian: NO

Detailed OS Information:
=====
ID: fedora
VERSION_ID: 42
PRETTY_NAME: Fedora Linux 42 (Container Image)
```

Diseño del sistema de archivos

En esta sección se describe el diseño del sistema de archivos de un sistema de AL2023, incluidos los detalles que pueden ser específicos de las instancias o los contenedores basados en AL2023. Para obtener más información, consulte la página man de `file-hierarchy(7)`.

Temas

- [/ \(el directorio raíz\)](#)
- [/boot \(Kernel, initramfs, etc.\)](#)
- [/etc \(Configuración del sistema\)](#)
- [/home \(Directorios de inicio del usuario\)](#)
- [/root \(directorio de inicio del usuario root\)](#)
- [/srv \(carga útil del servidor\)](#)
- [/tmp \(archivos temporales pequeños\)](#)
- [/run \(datos de tiempo de ejecución\)](#)
- [/usr \(recursos del sistema\)](#)
- [/var \(datos de sistema variables persistentes\)](#)

/ (el directorio raíz)

De forma predeterminada, las imágenes de AL2023 están configuradas con un elemento `/` grabable, lo que permite a los usuarios privilegiados crear nuevos archivos y directorios.

Es posible configurar los servicios de `systemd` para que utilicen una ruta o imagen diferente para que aparezca como `/` para ese servicio, así como establecer restricciones de acceso en cualquier ruta.

Note

Se recomienda configurar los servicios de `systemd` para restringir a qué tiene acceso el servicio. Esto puede incluir el uso de la directiva `ReadOnlyPaths=/` que hace que `/` sea de solo lectura.

Para obtener más información sobre cómo usar `systemd` para restringir el acceso de un servicio al sistema, consulte la página man de `systemd.exec(5)`.

/boot (Kernel, initramfs, etc.)

De forma predeterminada, las imágenes de arranque de AL2023 están configuradas con `/boot` en el sistema de archivos root. La ruta `/boot` solo es relevante para las imágenes de arranque, por lo que no se utiliza en las imágenes del contenedor de AL2023.

Este directorio alberga los archivos necesarios para el arranque de AL2023, como el núcleo de Linux y initramfs. El contenido de este directorio solo debe manipularse con las herramientas que se proporcionan con el sistema operativo.

/boot/efi (Partición del sistema EFI)

De forma predeterminada, las imágenes de arranque de AL2023 están configuradas con la partición EFI System montada en `/boot/efi`. Este sistema de archivos está gestionado por el sistema operativo y contiene el código y la configuración fundamentales para el arranque del sistema.

Esta ruta no es relevante para las imágenes del contenedor.

/etc (Configuración del sistema)

El directorio `/etc` en AL2023 contiene la configuración específica del sistema. De forma predeterminada, las imágenes de AL2023 vienen con `/etc` en el sistema de archivos root y los usuarios con privilegios pueden escribir en ellas.

Note

Es común que las aplicaciones (como `systemd`) mantengan la configuración predeterminada de [/usr \(recursos del sistema\)](#) que se puede anular introduciendo la configuración en [/etc \(Configuración del sistema\)](#).

En el caso de estas aplicaciones, al cambiar los archivos de [/usr \(recursos del sistema\)](#) en lugar de anular la configuración predeterminada de `/etc`, es probable que los cambios se anulen cuando se actualice el paquete.

/home (Directorios de inicio del usuario)

Los usuarios normales tienen sus directorios principales en `/home`, pero el software siempre debe buscar la variable de entorno `$HOME` por usuario en lugar de basarse en un patrón como `/home/$USER`.

De forma predeterminada, las imágenes de AL2023 tienen `/home` en el sistema de archivos `root`, pero el software no debe confiar en esto. Es perfectamente válido configurar el sistema operativo como un sistema de archivos independiente, que se monta más adelante durante el arranque o solo después de que el usuario se autentique en el sistema. `/home`

El directorio de inicio del usuario `root` no está en `/home`, sino que [/root \(directorio de inicio del usuario root\)](#) está disponible por si no se puede montar el sistema de archivos `/home`.

Note

Se recomienda configurar los servicios `systemd` que no necesitan acceso de escritura a `/home` con la directiva `ProtectHome=read-only`. Con esta opción, `/home`, `/root` y `/run/user` se convierten en solo lectura para el servicio.

También se recomienda que los servicios que no necesitan acceso a `/home` se configuren con la directiva `ProtectHome=tmpfs`, que ejecutará el servicio en un entorno de pruebas en el que `/home`, `/root` y `/run/user` son sistemas de archivos `tmpfs` vacíos de solo lectura.

Para obtener más información sobre cómo usar `systemd` para restringir el acceso de un servicio al sistema, consulte la página `man de systemd.exec(5)`.

`/root` (directorio de inicio del usuario `root`)

El directorio principal del usuario `root` es el directorio `/root`, separado a propósito de [/home \(Directorios de inicio del usuario\)](#) para que esté presente en caso de que [/home \(Directorios de inicio del usuario\)](#) se encuentre en un sistema de archivos que no esté disponible.

La práctica recomendada para configurar los servicios de `systemd` es la misma para `/root` que para [/home \(Directorios de inicio del usuario\)](#).

`/srv` (carga útil del servidor)

El administrador del sistema administra el directorio `/srv` y Amazon Linux 2023 no impone restricciones a la organización de este directorio.

Es posible configurar el directorio `/srv` para que esté en un sistema de archivos independiente, por lo que solo estará disponible más adelante durante el arranque.

/tmp (archivos temporales pequeños)

Note

Amazon Linux 2023 es diferente a Amazon Linux 2, ya que de forma predeterminada /tmp es ahora tmpfs y no una ruta en el sistema de archivos root.

Note

Cuando se ejecuta en un contenedor, normalmente será la configuración del tiempo de ejecución del contenedor la que determine si /tmp es tmpfs, o una ruta en el disco, y si hay un proceso de limpieza en ejecución o no.

El directorio /tmp es para archivos temporales pequeños y con un tamaño limitado. De forma predeterminada, AL2023 lo configura para que sea un sistema de archivos tmpfs con un límite de tamaño del 50 % de la RAM y un máximo de un millón de inodes.

Las aplicaciones deberían preferir la ruta de la variable de entorno \$TMPDIR antes que la ruta /tmp. A continuación, los usuarios pueden configurar la variable de entorno \$TMPDIR para anular la ruta que debe utilizar una aplicación /tmp

En el caso de archivos temporales más grandes, [/var/tmp](#) debería usarse en su lugar.

Warning

Como /tmp es compartido, es importante utilizar métodos seguros para crear archivos temporales. Para obtener más información, consulte la documentación de systemd sobre el [uso seguro de /tmp y /var/tmp](#).

Note

Se recomienda que los servicios de systemd se configuren con la directiva PrivateTmp= establecida yes o que disconnected ejecute el servicio en un entorno de pruebas, donde /tmp y [/var/tmp](#) no se comparten con el host ni con otros servicios.

Para obtener más información, incluida la forma de configurar dos servicios para que compartan los mismos directorios temporales privados, consulte la página `man de systemd.exec(5)`.

El contenido de `/tmp` se suele limpiar en el momento del arranque y los archivos no utilizados se limpian periódicamente. De forma predeterminada, el proceso de limpieza se ejecuta poco después del arranque y, a continuación, todos los días. Para obtener información sobre cómo configurar la limpieza de los archivos temporales, consulte `tmpfiles.d(5)` y las páginas del manual `man de systemd-tmpfiles(8)`.

Las rutas `/var/tmp` y `/tmp` están estrechamente relacionadas y existen para distintos propósitos.

/run (datos de tiempo de ejecución)

Los paquetes del sistema utilizan el directorio `/run` para almacenar pequeñas cantidades de datos en tiempo de ejecución (como archivos de socket). Es un sistema de archivos `tmpfs` y solo los programas privilegiados pueden escribir en él.

Los componentes del sistema pueden utilizar el directorio `/run/log` para almacenar los registros, ya sea antes de que se escriban en `/var/log` o antes de que el sistema de archivos `/var/log` esté disponible.

La ruta `/run/user/` contiene los directorios de tiempo de ejecución por usuario. De forma predeterminada, se trata de sistemas de archivos `tmpfs` individuales montados por `systemd` cuando el usuario inicia sesión y se borran cuando el usuario deja de iniciar sesión. Según la [especificación del directorio base de XDG](#), no se debe hacer referencia a estas rutas directamente, sino a través de la variable de entorno `$XDG_RUNTIME_DIR`.

/usr (recursos del sistema)

La jerarquía `/usr` corresponde a los recursos del sistema operativo proporcionados por el proveedor. A excepción de la jerarquía `/usr/local`, nada debería modificar nada dentro de `/usr` excepto el administrador de paquetes del sistema operativo.

Las aplicaciones de software deben asumir que `/usr` puede ser de solo lectura. La jerarquía `/usr` no debe usarse para datos inútiles. Con la excepción de `/usr/local`, la jerarquía `/usr` no debe

usarse para ningún dato que se añada o modifique fuera de la instalación o eliminación del paquete, como lo hace el administrador de paquetes del sistema operativo. El administrador de paquetes del sistema operativo puede suponer que toda la jerarquía `/usr` (excepto [`/usr/local`](#)) es el mismo punto de montaje.

El software que se instale fuera del administrador de paquetes del sistema operativo no debe almacenar datos en `/usr`, ya que esto podría impedir cualquier invocación futura del administrador de paquetes del sistema operativo. La jerarquía [`/usr/local`](#) es la excepción y está reservada para el software ajeno al administrador de paquetes del sistema operativo.

`/usr/bin` (ejecutables)

Archivos ejecutables que deben aparecer en la búsqueda estándar `$PATH` y que son útiles para invocar desde un intérprete de comandos. Los daemons y ejecutables que no son útiles para invocar desde un intérprete de comandos se encuentran en `/usr/lib` o `/usr/libexec`.

`/usr/include` (encabezados C/C++)

El directorio `/usr/include` contiene archivos de encabezado C y C++, que normalmente se encuentran en paquetes con el sufijo `-devel`.

`/usr/lib` y `/usr/lib64` (bibliotecas compartidas)

En Amazon Linux 2023, la ruta `/usr/lib64` se usa para bibliotecas compartidas de 64 bits y datos de paquetes que dependen de la arquitectura. Dado que AL2023 no incluye ningún soporte de espacio de usuario de 32 bits, solo hay bibliotecas compartidas de 64 bits disponibles.

La ruta `/usr/lib` es para los datos estáticos de los paquetes del sistema operativo, que son compatibles con todas las arquitecturas. Esto puede incluir ejecutables que normalmente no se invocan desde un intérprete de comandos, que también se pueden encontrar en `/usr/libexec`. Las bibliotecas compartidas se encuentran en `/usr/lib64` en lugar de en `/usr/lib`.

`/usr/local` (software instalado por el administrador del sistema)

En Amazon Linux 2023, la ruta `/usr/local` está disponible para que el administrador del sistema instale software que no sea propiedad del sistema operativo y que no vaya a ser afectado por el sistema operativo. La jerarquía `/usr/local` predeterminada refleja la jerarquía `/`.

/usr/share (recursos compartidos)

Los recursos compartidos, como la documentación, los orígenes y los datos de zona horaria, se encuentran en `/usr/share`. Es habitual que varias especificaciones determinen exactamente dónde y en qué formato se almacenan los datos en este directorio.

/usr/share/doc (documentación)

La documentación que viene con los paquetes se almacenará en `/usr/share/doc`.

/var (datos de sistema variables persistentes)

/var/cache (caché)

A diferencia de [/var/lib](#), borrar datos en `/var/cache` no provocará la pérdida de datos, ya que las aplicaciones deben poder reconstruir sus datos `/var/cache` a partir de otros orígenes.

/var/lib (datos de sistema persistentes)

El directorio `/var/lib` se utiliza para los datos persistentes del sistema. Varios componentes del sistema colocarán aquí los datos que son privados para ese componente. A diferencia de [/var/cache](#), borrar datos en `/var/lib` provocará la pérdida de datos.

Por ejemplo, el servidor de bases de datos PostgreSQL almacenará los datos de la base de datos de forma predeterminada en `/var/lib/pgsql`. El diseño y los formatos de archivo de estos datos son privados para PostgreSQL y son datos persistentes, ya que si se borran, el usuario sufre una pérdida de datos.

/var/log (registros persistentes)

Este directorio se usa para almacenar registros persistentes. Se recomienda que el software utilice las llamadas a la API `syslog(3)` o `sd_journal_print(3)` en lugar de almacenar directamente los archivos de registro en `/var/log`.

Note

En AL2023 [El diario systemd reemplaza a rsyslog](#), lo que supone una diferencia notable con respecto a la configuración predeterminada de Amazon Linux 2.

Para obtener más información sobre cómo leer los registros con `journalctl`, consulte la página del manual de [journalctl](#).

Muchas aplicaciones utilizan sus propios mecanismos para escribir y, a veces, rotar los archivos de registro que se encuentran en `/var/log`. Consulte la documentación de estas aplicaciones para saber cómo configurar sus archivos de registro.

`/var/spool` (colas de correo e impresoras)

Este directorio se usa para datos persistentes, como colas de correo o impresoras.

`/var/tmp` (archivos temporales más grandes)

Para archivos temporales pequeños y de tamaño limitado, posiblemente [/tmp](#) debería usarse en su lugar.

Si bien [/tmp](#) está configurado de forma predeterminada como un volumen `tmpfs`, por defecto `/var/tmp` está configurado como una ruta en el sistema de archivos raíz y, por lo tanto, es el lugar para archivos temporales más grandes y persistentes. De forma predeterminada, hay un trabajo de limpieza que se ejecuta de forma regular y que elimina los archivos a los que no se ha accedido recientemente.

Para obtener información sobre cómo configurar la limpieza de los archivos temporales, consulte `tmpfiles.d(5)` y las páginas del manual `man` de `systemd-tmpfiles(8)`.

Al igual que con [/tmp](#), las aplicaciones deben preferir la ruta especificada en la variable de entorno `$TMPDIR` sobre `/var/tmp`. Los usuarios pueden entonces establecer la variable de entorno `$TMPDIR` para anular la ruta que una aplicación debe utilizar para `/var/tmp`.

Warning

Dado que `/var/tmp` es compartido (al igual que [/tmp](#)), es importante utilizar métodos seguros para crear archivos temporales. Para obtener más información, consulte la documentación de `systemd` sobre el [uso seguro de /tmp y /var/tmp](#).

 Note

Se recomienda que los servicios de `systemd` se configuren con la directiva `PrivateTmp=` establecida `yes` o que `disconnected` ejecute el servicio en un entorno de pruebas, donde [/tmp](#) y [/var/tmp](#) no se comparten con el host ni con otros servicios.

Para obtener más información, incluida la forma de configurar dos servicios para que compartan los mismos directorios temporales privados, consulte la página man de `systemd.exec(5)`.

Las rutas [/var/tmp](#) y [/tmp](#) están estrechamente relacionadas y existen para distintos propósitos.

Actualización AL2023

Es importante mantenerse al día con las AL2023 versiones para poder beneficiarse de las actualizaciones de seguridad y las nuevas funciones. De este AL2023 modo, puede garantizar la coherencia entre las versiones de los paquetes y las actualizaciones en todo su entorno [Actualizaciones deterministas mediante repositorios versionados en AL2023](#).

Warning

La ejecución de `dnf --releasever=latest update` no es la mejor práctica y es probable que dé lugar a que una actualización del sistema operativo se pruebe por primera vez en producción.

En lugar de `latest` utilizarla, utilice una versión AL2023 de lanzamiento específica. Esto garantiza que esté implementando los mismos cambios en todas las instancias de producción que los que probó anteriormente. Por ejemplo, siempre `dnf --releasever=2023.10.20260325 update` se actualizará a la versión 2023.10.20260325.

[Para obtener más información, consulte la sección de actualizaciones de la Guía del usuario AL2023. AL2023](#)

Temas

- [Prácticas recomendadas para implementar actualizaciones de forma segura](#)
- [Recibir notificaciones sobre nuevas actualizaciones](#)
- [Actualizaciones deterministas mediante repositorios versionados en AL2023](#)
- [Gestione las actualizaciones de paquetes y sistemas operativos en AL2023](#)
- [Aplicación de parches activos del kernel en AL2023](#)
- [Actualización del núcleo de Linux en AL2023](#)

Prácticas recomendadas para implementar actualizaciones de forma segura

Amazon Linux 2023 (AL2023) tiene varias funciones diseñadas para ayudar a implementar de forma segura las actualizaciones del sistema operativo y poder saber qué ha cambiado entre las

actualizaciones y, si es necesario, volver fácilmente a la versión anterior. En esta sección se analizan las lecciones AWS aprendidas durante más de una década de uso interno y externo de Amazon Linux.

 Warning

La ejecución de `dnf --releasever=latest update` no es la mejor práctica y es probable que dé lugar a que una actualización del sistema operativo se pruebe por primera vez en producción.

En lugar de usarlo `latest`, utilice una versión AL2023 de lanzamiento específica.

Esto garantiza que esté implementando los mismos cambios en todas las instancias de producción que los que probó anteriormente. Por ejemplo, siempre `dnf --releasever=2023.10.20260325 update` se actualizará a la versión 2023.10.20260325.

[Para obtener más información, consulte la sección de actualizaciones de la Guía del usuario AL2023. AL2023](#)

Si no se planifica la seguridad de la implementación de las actualizaciones del sistema operativo, el impacto de una interacción negativa e inesperada entre una actualización del sistema operativo `application/service` y una actualización del sistema operativo puede ser considerablemente mayor, e incluso llegar a producirse una interrupción total. Como ocurre con cualquier problema de software, cuanto antes se detecte el problema, menor será el impacto que tendrá en los usuarios finales.

Es importante no caer en la trampa de creer dos cosas que son fundamentalmente falsas:

1. El proveedor del sistema operativo nunca se equivocará al actualizar el sistema operativo.
2. El comportamiento específico o la interfaz del sistema operativo en el que se basa coinciden con el comportamiento y las interfaces en las que el proveedor del sistema operativo consideraría fiables.

Es decir, tanto el proveedor del sistema operativo como usted estarán de acuerdo en que ha habido un problema con la actualización.

No confíe en las buenas intenciones, instale sistemas que garanticen que la seguridad de la implementación incluya cualquier actualización del sistema operativo.

No se recomienda probar las nuevas actualizaciones del sistema operativo implementándolas en entornos de producción. Se recomienda considerar el sistema operativo como una parte más de su

implementación y pensar en aplicar los mismos mecanismos de seguridad de implementación que considere adecuados para cualquier otro cambio en un entorno de producción.

Se recomienda probar todas y cada una de las actualizaciones del sistema operativo antes de implementarlas en los sistemas de producción. Al realizar la implementación, se recomienda una implementación gradual combinada con una buena supervisión. Las implementaciones por etapas permiten garantizar que, si se produce un problema, aunque no sea inmediato, el impacto se limite a un subconjunto de la flota, y se puede detener el despliegue de la actualización mientras se investigan y mitigan más a fondo.

La prioridad principal suele ser mitigar cualquier impacto negativo derivado de la actualización del sistema operativo, seguida de resolver el problema, esté donde esté. Cuando la introducción de una actualización del sistema operativo se correlaciona con un impacto negativo, la posibilidad de volver a la versión anterior del sistema operativo que se sepa que es correcta es una herramienta muy útil.

Amazon Linux 2023 presenta [Actualizaciones deterministas mediante repositorios versionados](#) una nueva y potente característica para garantizar que cualquier cambio en la versión del sistema operativo (o en los paquetes individuales) sea repetible. Por lo tanto, si se produce un problema al pasar de una versión del sistema operativo a la siguiente, existen mecanismos fáciles de usar para ceñirse a la versión del sistema operativo que se sabe que funciona y, al mismo tiempo, encontrar la forma de resolver el problema.

De AL2023 hecho, cada vez que publicamos nuevos paquetes de actualizaciones, hay una nueva versión que bloquea y otra nueva AMIs que bloquea esa versión. Las [notas AL2023 de la versión](#) incluyen los cambios en cada versión y los problemas [Avisos de seguridad de Amazon Linux para AL2023](#) de seguridad abordados en las actualizaciones de los paquetes.

[Por ejemplo, si te ha afectado el problema presente en la versión 2023.6.20241028, puedes volver a utilizar inmediatamente las imágenes AMIs y el contenedor de la versión anterior, la 2023.6.20241010.](#) En este caso, había un error en un paquete que se corrigió en la siguiente versión [2023.6.20241031](#), pero con [Actualizaciones deterministas mediante repositorios versionados](#), cualquier persona afectada podía tomar inmediatamente medidas sencillas para mitigarlo: basta con utilizar las imágenes anteriores.

[Actualizaciones deterministas mediante repositorios versionados](#) también garantiza que cualquier implementación en curso de una actualización del sistema operativo, ya sea implementada o mediante el lanzamiento de imágenes nuevas o de contenedores, no se verá afectada por las actualizaciones del sistema operativo publicadas posteriormente. AMIs

En nuestro primer ejemplo, la flota A es una gran flota que se encuentra a mitad del proceso de implementación de la actualización desde la versión [2023.5.20241001](#) a la versión [2023.6.20241010](#) cuando salga la versión [2023.6.20241028](#). [Actualizaciones deterministas mediante repositorios versionados](#) significa que la implementación para la flota A continúa sin ningún cambio en las actualizaciones que se están aplicando.

El objetivo de las estrategias de despliegue basadas en oleadas o en fases, como el despliegue primero en el 1 % de la flota y, después, en el 5 %, el 10 %, el 20 % o el 40 % hasta alcanzar el 100 %, es poder probar un cambio de forma limitada antes de ampliarlo. Por lo general, este tipo de estrategia de despliegue se considera la mejor práctica para implementar cualquier cambio en la producción.

Con una estrategia de implementación basada en oleadas y la actualización de la flota A a la versión [2023.6.20241010](#) al encontrarse en una fase en la que se está implementando en muchos hosts a la vez, el hecho de que se haya lanzado la versión [2023.6.20241028](#) no tiene ningún impacto en la implementación en curso gracias al uso de [Actualizaciones deterministas mediante repositorios versionados](#).

Si la flota B ejecutaba una versión anterior, por ejemplo, la [2023.5.20240708](#), y hubiera empezado a implementar la actualización a la versión [2023.6.20241028](#), y la flota B se viera afectada por el problema en esa versión, esto se notaría al principio de la implementación. En ese momento, se puede decidir si se debe pausar alguna implementación hasta que se encuentre disponible una solución para ese problema o si, mientras tanto, se inicia una implementación de la misma versión que estaba ejecutando la flota A, la versión [2023.6.20241010](#), de modo que la flota B reciba todas las actualizaciones entre las versiones [2023.5.20240708](#) y [2023.6.20241010](#).

Es importante tener en cuenta que no actualizar el sistema operativo a tiempo puede provocar problemas. Es probable que las nuevas actualizaciones contengan correcciones de errores y de seguridad que pueden ser relevantes para su entorno. Para obtener más información, consulte [Seguridad y conformidad en Amazon Linux 2023](#) y [Gestione las actualizaciones de paquetes y sistemas operativos en AL2023](#).

Es importante configurar los sistemas de implementación para que puedan incorporar fácilmente las nuevas actualizaciones del sistema operativo, probarlas antes de ponerlas en producción y utilizar mecanismos como las implementaciones basadas en oleadas para minimizar cualquier impacto negativo. Para poder mitigar cualquier impacto negativo de una actualización del sistema operativo, es importante saber cómo hacer que los sistemas de implementación apunten a una versión anterior del sistema operativo comprobada y, una vez solucionado el problema, dejen de

estar bloqueados a la versión anterior de funcionalidad comprobada, sino que pasen a una nueva versión de funcionalidad comprobada.

Preparación para actualizaciones menores

La preparación para actualizaciones más pequeñas del sistema operativo, como una nueva versión puntual de, AL2023 está pensada para que no suponga ningún esfuerzo. Asegúrese de leer las [notas de la AL2023 versión](#) para conocer los próximos cambios.

El [período de soporte de un paquete](#) que llega a su fin puede implicar el cambio una versión más reciente del tiempo de ejecución del lenguaje (por ejemplo, con [PHPen AL2 023](#)). Lo mejor es prepararse con antelación para ello, pasando a las nuevas versiones del tiempo de ejecución del lenguaje con suficiente antelación al final del período de soporte.

En el caso de paquetes como [pcre versión 1](#), también existe la posibilidad de planificar con antelación y migrar cualquier parte del código a su sustituto, que en este caso es la versión 2 de pcre. Lo mejor es hacerlo lo antes posible, para tener tiempo suficiente en caso de que surjan contratiempos.

Cuando no exista un sustituto directo, como en el caso de [Berkeley DB \(libdb\)](#), es posible que tengas que tomar una decisión en función de tu caso de uso.

Preparación para actualizaciones importantes

Casi todos consideran que la actualización a una nueva versión principal de un sistema operativo es algo que requiere planificación, trabajo para adaptarse a una funcionalidad modificada o obsoleta, y también pruebas antes de la implementación. No es raro poder prepararse para la próxima versión principal de Amazon Linux 2023 de forma gradual, por ejemplo, abordando cualquier uso de funciones obsoletas o eliminadas antes de pasar a la siguiente versión principal.

Por ejemplo, al pasar de AL2 a AL2023, leer la [Funcionalidad obsoleta AL2 y eliminada en AL2023](#) sección puede resultar en una serie de pasos pequeños y seguros que se pueden dar mientras se sigue utilizando AL2 para prepararse AL2023. Por ejemplo, cualquier uso de [Python 2.7 ha sido reemplazado por Python 3](#) (fuera del uso del sistema operativo, como en el administrador de paquetes yum) se puede migrar a Python 3 para prepararlo para usar [Pythonen AL2023](#). Si usa [PHP](#), tanto AL2 (a través del PHP 8.2 [AL2 Extra](#)) como PHP AL2023 8.2, es decir, la migración de la versión PHP y la migración del sistema operativo no tienen que realizarse simultáneamente.

Mientras se usa AL2023, también es posible prepararse para la próxima versión principal de Amazon Linux 2023 hoy, mientras se usa AL2023. La [Obsoleta en AL2023](#) sección cubre las funciones y los paquetes que están en desuso AL2023 y que se van a eliminar.

Por ejemplo, migrar cualquier uso restante de [System V init \(sysvinit\)](#) como los scripts de `init` a su equivalente en `systemd` le preparará para el futuro y le permitirá utilizar todas las características de `systemd` para supervisar el servicio, cómo y cuándo reiniciarlo, qué otros servicios necesita y si se deben aplicar restricciones de recursos o permisos.

En el caso de características como la compatibilidad con 32 bits, la obsolescencia puede afectar a varias versiones principales del sistema operativo. Para 32 bits, Amazon Linux 1 (AL1) está en desuso, [AMI x86 \(i686\) de 32 bits](#) Amazon Linux 2 está en desuso [Paquetes x86 \(i686\) de 32 bits](#) y Amazon Linux 2023 está en desuso. [Compatibilidad con tiempo de ejecución de 32 bits x86 \(i686\)](#) La transición desde [IMDSv1](#) también abarca varias versiones principales del sistema operativo. Para este tipo de cambios, se entiende que algunos clientes necesitan más tiempo para adaptarse, por lo que hay un amplio margen antes de que la funcionalidad deje de estar disponible en Amazon Linux 2023.

La lista de funciones obsoletas se actualiza a lo largo de la vida útil del sistema operativo y es recomendable mantenerse al día con los cambios que se introduzcan en la misma.

Recibir notificaciones sobre nuevas actualizaciones

Puede recibir notificaciones cada vez que se lance una nueva versión de AL2023. Las notificaciones se publican con [Amazon SNS](#) mediante el tema siguiente.

```
arn:aws:sns:us-east-1:137112412989:amazon-linux-2023-ami-updates
```

Los mensajes se publican aquí cuando se publica una nueva AMI de AL2023. La versión de la AMI se incluirá en el mensaje.

Estos mensajes se pueden recibir mediante varios métodos diferentes. Le recomendamos que utilice el siguiente:

1. Abra la [consola de Amazon SNS](#).
2. En la barra de navegación, cambie Región de AWS a Este de EE. UU. (Norte de Virginia), si es necesario. Debe seleccionar la región donde la notificación de SNS a la que se va a suscribir se ha creado.

3. En el panel de navegación, elija Subscriptions, Create subscription.
4. En el cuadro de diálogo Crear suscripción, haga lo siguiente:
 - a. En ARN del tema, copie y pegue el siguiente nombre de recurso de Amazon (ARN):
arn:aws:sns:us-east-1:137112412989:amazon-linux-2023-ami-updates.
 - b. En Protocolo, elija Correo electrónico.
 - c. En Punto de conexión, escriba una dirección de correo electrónico que pueda utilizar para recibir notificaciones.
 - d. Seleccione Crear suscripción.
5. Recibirá un correo electrónico de confirmación con el asunto "Notificación de AWS: confirmación de suscripción". Abra el correo electrónico y elija Confirmar suscripción para completar la suscripción.

Actualizaciones deterministas mediante repositorios versionados en AL2023

Note

De forma predeterminada, la AL2023 instancia no recibe automáticamente actualizaciones de seguridad críticas e importantes adicionales en el momento del lanzamiento. La instancia contiene inicialmente las actualizaciones que estaban disponibles en la versión de la AMI elegida AL2023 y en la misma.

Control de las actualizaciones recibidas de las versiones principales y secundarias

De este AL2023 modo, puede garantizar la coherencia entre las versiones de los paquetes y las actualizaciones en todo su entorno. También puede garantizar la coherencia de varias instancias de la misma Imagen de máquina de Amazon (AMI). Con la característica de actualizaciones determinista a través de repositorios versionados, que está activada de forma predeterminada, puede aplicar las actualizaciones según un cronograma que se adapte a sus necesidades específicas.

Cada vez que publicamos nuevas actualizaciones de paquetes, hay una nueva versión que bloquear y una nueva AMIs que se bloquea para esa versión.

AL2023 se bloquea a una versión específica de tu repositorio. Esto es compatible con las versiones principales y secundarias. La AL2023 AMI, expuesta a través de nuestros parámetros SSM, es siempre la última versión. Tiene la mayoría de los up-to-date paquetes y actualizaciones, incluidas las actualizaciones de seguridad críticas e importantes.

Si lanza una instancia desde una AMI existente, las actualizaciones no se aplican automáticamente. Todos los paquetes adicionales que se instalen como parte del aprovisionamiento se asignan a la versión de repositorio de la AMI existente.

Con esta característica, usted se encarga de garantizar la coherencia entre las versiones y actualizaciones de los paquetes en todo su entorno. Esto es especialmente cierto si lanza varias instancias desde la misma AMI. Puede aplicar las actualizaciones según un cronograma adecuado a sus necesidades. También puede aplicar un conjunto específico de actualizaciones en el momento del lanzamiento, ya que también se pueden bloquear en una versión de repositorio específica.

Diferencias entre las actualizaciones de versión principales y secundarias

Las versiones principales de AL2023 incluyen actualizaciones a gran escala y pueden añadir, eliminar o actualizar paquetes. Para garantizar la compatibilidad, actualice la instancia a una nueva versión principal solo después de probar la aplicación en esa versión.

Las versiones secundarias de AL2023 incluyen actualizaciones de funciones y seguridad, pero no incluyen cambios en los paquetes. Esto garantiza que las características de Linux y la API de la biblioteca del sistema permanezcan disponibles en las nuevas versiones. No es necesario probar la aplicación antes de actualizarla.

Saber cuándo están disponibles las actualizaciones

Para aplicar una actualización, debe saber si hay una disponible y, a continuación, saber cómo implementarla.

Para compilaciones derivadas AMIs cuando se AL2023 AMIs lanzan nuevas versiones, [EC2 Image Builder](#) puede compilar, parchear y AMIs probar automáticamente. Puedes activar tu propia AMI construyendo canalizaciones o usar la base AMIs. [Recibir notificaciones sobre nuevas actualizaciones](#)

Para aplicar los parches in situ, puede utilizar herramientas como el [administrador de parches AWS Systems Manager](#) para organizar la aplicación de las actualizaciones en toda la flota.

Para otros públicos AMIs AL2023, sus proveedores AMIs pueden tener su propio calendario de lanzamiento y métodos de notificación. Cuando utilice imágenes derivadas AMIs o de contenedores, consulte la documentación del editor para saber cuándo se publican las actualizaciones.

Los cambios de cada versión están documentados en las [notas de la AL2023 versión](#). Las actualizaciones de seguridad se publican en el [Centro de seguridad de Amazon Linux \(ALAS\)](#).

Controle las actualizaciones de paquetes disponibles en los AL2023 repositorios

Cuando publicamos una nueva versión de los AL2023 repositorios, todas las versiones anteriores siguen estando disponibles. De forma predeterminada, el complemento para administrar las versiones del repositorio se bloquea en la misma versión que se utilizó para crear la AMI. Si desea controlar las actualizaciones de paquetes, siga estos pasos.

1. Descubra las versiones disponibles del repositorio ejecutando el siguiente comando.

```
$ sudo dnf check-release-update
```

2. Seleccione una versión ejecutando el siguiente comando.

```
$ sudo dnf upgrade --releasever=version
```

Este comando inicia una actualización dnf desde la versión de lanzamiento actual de Amazon Linux hasta la versión de lanzamiento especificada en la línea de comandos. dnf presenta una lista de las actualizaciones del paquete. Antes de procesar la actualización, debe confirmarla. Una vez completada la actualización, la nueva versión de lanzamiento pasa a ser la versión de lanzamiento predeterminada que utilizará dnf para todas las actividades futuras.

Para obtener más información, consulte [Gestione las actualizaciones de paquetes y sistemas operativos en AL2023](#).

Actualizaciones deterministas mediante el reemplazo de instancias

La función [Actualizaciones deterministas mediante repositorios versionados en AL2023](#) de Amazon Linux 2023 hace que el reemplazo de instancias sea una forma fácil de implementar versiones actualizadas de AL2023 de forma determinista y segura. Las actualizaciones deterministas significan

que, a medida que se lanza una nueva versión de forma progresiva, si se encuentra algún problema, es fácil volver a la AMI anterior y determinar la causa del problema.

El reemplazo de instancias en lugar de la aplicación de parches in situ hace que las actualizaciones sean más deterministas y predecibles, ya que el lanzamiento de una nueva capacidad puede consistir en una ruta de código bien comprobada con estados A y B claros. Cada uno de los estados anterior y posterior puede probarse adecuadamente en un sistema de CI/CD antes de que comience la implementación.

Al aplicar los parches in situ, hay muchos estados intermedios entre el momento anterior y el posterior a la aplicación de las actualizaciones, lo que resulta más difícil de probar en todas las combinaciones de estados.

Una estrategia de actualización del sistema operativo que consiste en sustituir las instancias por actualizaciones deterministas se adapta perfectamente a los modelos de despliegue azul/verde, por oleadas y por fases.

Uso de actualizaciones deterministas a través de repositorios versionados

Temas

- [Uso de un sistema determinista mejorado](#)
- [Actualización selectiva de un sistema determinista actualizado](#)
- [Uso de la anulación persistente con una actualización determinista](#)

Uso de un sistema determinista mejorado

Note

El comportamiento predeterminado del administrador de paquetes ha cambiado de AL2.

Las actualizaciones deterministas son una forma eficaz de garantizar que todos los cambios en los entornos de producción puedan probarse exhaustivamente antes de una implementación generalizada. Cada AL2023 AMI nueva está bloqueada a una versión concreta de AL2023. Esto proporciona un comportamiento determinista de las versiones de los paquetes del sistema operativo que se instalan al lanzar la AMI específica. Las actualizaciones locales pueden ser para una versión de lanzamiento específica, lo que garantiza un comportamiento determinista en toda la flota. A medida que vaya pasando a una versión nueva AMIs o a una versión actualizada, podrá probar cada

una de ellas en CI/CD proceso y detectar cualquier posible problema antes de implementarlas en entornos de producción.

Puede utilizar herramientas como el [administrador de parches AWS Systems Manager](#) para organizar la aplicación de las actualizaciones en toda la flota. Para compilaciones derivadas AMIs cuando AL2023 AMIs se lanzan nuevas, [EC2 Image Builder](#) puede compilar, parchear y AMIs probar automáticamente, o [Recibir notificaciones sobre nuevas actualizaciones](#) puede saber cuándo hay nuevas AMIs bases disponibles o activar sus propias canalizaciones de creación de AMI.

Para obtener información sobre cómo restringir las actualizaciones a las de un aviso concreto, consulte [Aplicación de actualizaciones de seguridad in situ](#)

Para aplicar los parches in situ, puede utilizar el administrador de paquetes dnf. Al ejecutar el comando `dnf upgrade`, el sistema comprueba si hay actualizaciones en el repositorio que especifica la variable `releasever`. Una versión válida `releasever` es una versión con fecha *latest* o una versión con fecha, por ejemplo. *2023.10.20260325*

Puede cambiar el valor de `releasever` mediante uno de los métodos siguientes. Estos métodos se muestran en orden descendente de prioridad del sistema. Esto significa que el método 1 anula los métodos 2 y 3, y el método 2 anula el método 3.

1. El valor de la marca de línea de comandos, `--releasever=latest`, si se usa.
2. El valor que se especifica en el archivo de variables de anulación `/etc/dnf/vars/releasever`, si está establecido.
3. La versión del paquete `system-release` actualmente instalado.

En el siguiente ejemplo, la versión es: *2023.0.20230210*

```
$ rpm -q system-release
system-release-2023.0.20230210-0.amzn2023.noarch
```

En un sistema recién instalado, la variable de anulación no está presente. No hay actualizaciones disponibles porque el sistema está bloqueado en la versión instalada de `system-release`.

```
$ cat /etc/dnf/vars/releasever
cat: /etc/dnf/vars/releasever: No such file or directory
```

```
$ sudo dnf upgrade
Last metadata expiration check: 0:00:02 ago on Wed 15 Feb 2023 06:14:12 PM UTC.
```

```
Dependencies resolved.
Nothing to do.
Complete!
```

Puede obtener paquetes de una versión específica con la marca `releasever` para proporcionar la versión que desee.

```
$ rpm -q system-release
system-release-2023.0.20230222-0.amzn2023.noarch
```

```
$ sudo dnf upgrade --releasever=2023.0.20230329
Amazon Linux 2023 repository                26 MB/s | 12 MB      00:00
Dependencies resolved.
=====
Package                                Arch    Version                                Repository    Size
=====
Installing:
kernel                                aarch64 6.1.21-1.45.amzn2023                amazonlinux   26 M
Upgrading:
amazon-linux-repo-s3                  noarch  2023.0.20230329-0.amzn2023          amazonlinux   18 k
ca-certificates                       noarch  2023.2.60-1.0.amzn2023.0.1         amazonlinux   828 k
cloud-init                            noarch  22.2.2-1.amzn2023.1.7              amazonlinux   1.1 M

... [ list edited for clarity ]

system-release                        noarch  2023.0.20230329-0.amzn2023          amazonlinux   29 k

... [ list edited for clarity ]

vim-data                              noarch  2:9.0.1403-1.amzn2023.0.1          amazonlinux   25 k
vim-minimal                           aarch64 2:9.0.1403-1.amzn2023.0.1          amazonlinux   753 k

Transaction Summary
=====
Install    1 Package
Upgrade    42 Packages

Total download size: 56 M
```

Como la opción `--releasever` anula ambas, `system-release` y `/etc/dnf/vars/releasever`, el resultado de esta actualización es el siguiente:

1. La actualización reemplaza todos los paquetes instalados que cambiaron entre la versión anterior y la nueva.
2. La actualización bloquea el sistema en el repositorio de la nueva versión de `system-release`.

Al especificar siempre a qué AL2023 versión `releasever` (es decir, a qué versión) se debe actualizar, se obtiene un conjunto determinista de cambios en la flota. Lanzó la versión **A**, la actualizó **yB**, a continuación, la actualizó a **C**.

Actualización selectiva de un sistema determinista actualizado

Note

Se recomienda instalar todas las actualizaciones de una nueva versión en lugar de seleccionar actualizaciones específicas. Aplicar solo una parte de una actualización al sistema operativo debería ser una excepción a la práctica habitual de realizar la actualización completa.

Es posible que desee instalar algunos paquetes de una versión reciente y, al mismo tiempo, dejar el sistema bloqueado en la versión original.

Puede utilizar `dnf check-update` para identificar los paquetes que desee actualizar.

```
$ sudo dnf check-update --releasever=latest --security
Amazon Linux 2023 repository          13 MB/s | 10 MB      00:00
Last metadata expiration check: 0:00:02 ago on Wed 15 Feb 2023 02:52:21 AM UTC.

bind-libs.aarch64                     32:9.16.27-1.amzn2023.0.1      amazonlinux
bind-license.noarch                   32:9.16.27-1.amzn2023.0.1      amazonlinux
bind-utils.aarch64                    32:9.16.27-1.amzn2023.0.1      amazonlinux
cryptsetup.aarch64                    2.4.3-2.amzn2023.0.1           amazonlinux
cryptsetup-libs.aarch64                2.4.3-2.amzn2023.0.1           amazonlinux
curl-minimal.aarch64                  7.85.0-1.amzn2023.0.1          amazonlinux
glibc.aarch64                         2.34-40.amzn2023.0.2           amazonlinux
glibc-all-langpacks.aarch64            2.34-40.amzn2023.0.2           amazonlinux
glibc-common.aarch64                  2.34-40.amzn2023.0.2           amazonlinux
glibc-locale-source.aarch64            2.34-40.amzn2023.0.2           amazonlinux
gmp.aarch64                           1:6.2.1-2.amzn2023.0.1         amazonlinux
gnupg2-minimal.aarch64                 2.3.7-1.amzn2023.0.2           amazonlinux
gzip.aarch64                           1.10-5.amzn2023.0.1           amazonlinux
```

kernel.aarch64	6.1.12-17.42.amzn2023	amazonlinux
kernel-tools.aarch64	6.1.12-17.42.amzn2023	amazonlinux
libarchive.aarch64	3.5.3-2.amzn2023.0.1	amazonlinux
libcurl-minimal.aarch64	7.85.0-1.amzn2023.0.1	amazonlinux
libsepol.aarch64	3.4-3.amzn2023.0.2	amazonlinux
libsolv.aarch64	0.7.22-1.amzn2023.0.1	amazonlinux
libxml2.aarch64	2.9.14-1.amzn2023.0.1	amazonlinux
logrotate.aarch64	3.20.1-2.amzn2023.0.2	amazonlinux
lua-libs.aarch64	5.4.4-3.amzn2023.0.1	amazonlinux
lz4-libs.aarch64	1.9.4-1.amzn2023.0.1	amazonlinux
openssl.aarch64	1:3.0.5-1.amzn2023.0.3	amazonlinux
openssl-libs.aarch64	1:3.0.5-1.amzn2023.0.3	amazonlinux
pcr2.aarch64	10.40-1.amzn2023.0.1	amazonlinux
pcr2-syntax.noarch	10.40-1.amzn2023.0.1	amazonlinux
rsync.aarch64	3.2.6-1.amzn2023.0.2	amazonlinux
vim-common.aarch64	2:9.0.475-1.amzn2023.0.1	amazonlinux
vim-data.noarch	2:9.0.475-1.amzn2023.0.1	amazonlinux
vim-enhanced.aarch64	2:9.0.475-1.amzn2023.0.1	amazonlinux
vim-filesystem.noarch	2:9.0.475-1.amzn2023.0.1	amazonlinux
vim-minimal.aarch64	2:9.0.475-1.amzn2023.0.1	amazonlinux
xz.aarch64	5.2.5-9.amzn2023.0.1	amazonlinux
xz-libs.aarch64	5.2.5-9.amzn2023.0.1	amazonlinux
zlib.aarch64	1.2.11-32.amzn2023.0.3	amazonlinux

Instale los paquetes que desee actualizar. Utilice `sudo dnf upgrade --releasever=latest` y los nombres de los paquetes para asegurarse de que el paquete `system-release` permanezca inalterado.

```
$ sudo dnf upgrade --releasever=latest openssl openssl-libs
```

```
Last metadata expiration check: 0:01:28 ago on Wed 15 Feb 2023 02:52:21 AM UTC.
```

```
Dependencies resolved.
```

```
=====
Package           Arch      Version                               Repository      Size
=====
Upgrading:
openssl           aarch64   1:3.0.5-1.amzn2023.0.3               amazonlinux     1.1 M
openssl-libs      aarch64   1:3.0.5-1.amzn2023.0.3               amazonlinux     2.1 M
```

```
Transaction Summary
```

```
=====
Upgrade  2 Packages
```


Total download size: 3.2 M

Note

El uso de `sudo dnf upgrade --releasever=latest` actualiza todos los paquetes, incluido `system-release`. Después, la versión permanece bloqueada con el nuevo `system-release` a menos que establezca la anulación persistente.

Uso de la anulación persistente con una actualización determinista

Note

Con las actualizaciones deterministas, puede integrar los cambios del sistema operativo en su CI/CD proceso. Al deshabilitar las actualizaciones deterministas, se elimina la posibilidad de realizar pruebas antes de implementarlas.

En lugar de añadir `--releasever=latest`, puedes usar la anulación persistente para desbloquear el sistema estableciendo el valor de la variable en `latest`. Si se usa siempre `latest`, se revierte el comportamiento del AL2023 modelo de AL2 actualización, en el que cualquier llamada al administrador de paquetes siempre busca la versión más reciente y no está limitada a ninguna versión específica del sistema operativo.

Warning

Al desbloquear el administrador de paquetes mediante una anulación persistente de las actualizaciones deterministas, se corre el riesgo de descubrir cualquier posible incompatibilidad entre la aplicación y una actualización del sistema operativo en fase de producción.

Si bien las incompatibilidades son poco frecuentes, con una actualización del sistema operativo se integran nuevos cambios de código en el entorno, las pruebas de integración pueden impedir la implementación de cambios de código que tengan un impacto negativo en los entornos de producción.

```
$ echo latest | sudo tee /etc/dnf/vars/releasever
```

latest

\$ sudo dnf upgrade

Last metadata expiration check: 0:03:36 ago on Wed 15 Feb 2023 02:52:21 AM UTC.
 Dependencies resolved.

=====				
Package	Arch	Version	Repository	Size
=====				
Installing:				
kernel	aarch64	6.1.73-45.135.amzn2023	amazonlinux	24 M
Upgrading:				
acl	aarch64	2.3.1-2.amzn2023.0.1	amazonlinux	72 k
alternatives	aarch64	1.15-2.amzn2023.0.1	amazonlinux	36 k
amazon-ec2-net-utils	noarch	2.3.0-1.amzn2023.0.1	amazonlinux	16 k
at	aarch64	3.1.23-6.amzn2023.0.1	amazonlinux	60 k
attr	aarch64	2.5.1-3.amzn2023.0.1	amazonlinux	59 k
audit	aarch64	3.0.6-1.amzn2023.0.1	amazonlinux	249 k
audit-libs	aarch64	3.0.6-1.amzn2023.0.1	amazonlinux	116 k
aws-c-auth-libs	aarch64	0.6.5-6.amzn2023.0.2	amazonlinux	79 k
aws-c-cal-libs	aarch64	0.5.12-7.amzn2023.0.2	amazonlinux	34 k
aws-c-common-libs	aarch64	0.6.14-6.amzn2023.0.2	amazonlinux	119 k
aws-c-compression-libs	aarch64	0.2.14-5.amzn2023.0.2	amazonlinux	22 k
aws-c-event-stream-libs	aarch64	0.2.7-5.amzn2023.0.2	amazonlinux	47 k
aws-c-http-libs	aarch64	0.6.8-6.amzn2023.0.2	amazonlinux	147 k
aws-c-io-libs	aarch64	0.10.12-5.amzn2023.0.6	amazonlinux	109 k
aws-c-mqtt-libs	aarch64	0.7.8-7.amzn2023.0.2	amazonlinux	61 k
aws-c-s3-libs	aarch64	0.1.27-5.amzn2023.0.3	amazonlinux	54 k
aws-c-sdkutils-libs	aarch64	0.1.1-5.amzn2023.0.2	amazonlinux	26 k
aws-checksums-libs	aarch64	0.1.12-5.amzn2023.0.2	amazonlinux	50 k
awscli-2	noarch	2.7.8-1.amzn2023.0.4	amazonlinux	7.3 M
basesystem	noarch	11-11.amzn2023.0.1	amazonlinux	7.8 k
bash	aarch64	5.1.8-2.amzn2023.0.1	amazonlinux	1.6 M
bash-completion	noarch	1:2.11-2.amzn2023.0.1	amazonlinux	292 k
bc	aarch64	1.07.1-14.amzn2023.0.1	amazonlinux	120 k
bind-libs	aarch64	32:9.16.27-1.amzn2023.0.1	amazonlinux	1.2 M
bind-license	noarch	32:9.16.27-1.amzn2023.0.1	amazonlinux	14 k
bind-utils	aarch64	32:9.16.27-1.amzn2023.0.1	amazonlinux	206 k
binutils	aarch64	2.38-20.amzn2023.0.3	amazonlinux	4.6 M
boost-filesystem	aarch64	1.75.0-4.amzn2023.0.1	amazonlinux	55 k
boost-system	aarch64	1.75.0-4.amzn2023.0.1	amazonlinux	14 k
boost-thread	aarch64	1.75.0-4.amzn2023.0.1	amazonlinux	54 k
bzip2	aarch64	1.0.8-6.amzn2023.0.1	amazonlinux	53 k
bzip2-libs	aarch64	1.0.8-6.amzn2023.0.1	amazonlinux	44 k

```

c-ares                aarch64 1.17.2-1.amzn2023.0.1      amazonlinux 107 k
ca-certificates       noarch 2021.2.50-1.0.amzn2023.0.3    amazonlinux 343 k
checkpolicy           aarch64 3.4-3.amzn2023.0.1            amazonlinux 345 k
chkconfig             aarch64 1.15-2.amzn2023.0.1          amazonlinux 162 k
chrony                aarch64 4.2-7.amzn2023.0.4           amazonlinux 314 k
cloud-init            noarch 22.2.2-1.amzn2023.1.7        amazonlinux 1.1 M
cloud-utils-growpart  aarch64 0.31-8.amzn2023.0.2          amazonlinux 31 k
coreutils             aarch64 8.32-30.amzn2023.0.2         amazonlinux 1.1 M
coreutils-common      aarch64 8.32-30.amzn2023.0.2         amazonlinux 2.0 M
cpio                  aarch64 2.13-10.amzn2023.0.1         amazonlinux 269 k
cracklib               aarch64 2.9.6-27.amzn2023.0.1       amazonlinux 83 k
cracklib-dicts        aarch64 2.9.6-27.amzn2023.0.1       amazonlinux 3.6 M
crontabs              noarch 1.11-24.20190603git.amzn2023.0.1
                                                                amazonlinux 19 k

crypto-policies       noarch 20230128-1.gitdfb10ea.amzn2023.0.1
                                                                amazonlinux 61 k
crypto-policies-scripts noarch 20230128-1.gitdfb10ea.amzn2023.0.1
                                                                amazonlinux 81 k

...
Installing dependencies:
amazon-linux-repo-cdn noarch 2023.0.20230210-0.amzn2023    amazonlinux 16 k
xxhash-libs          aarch64 0.8.0-3.amzn2023.0.1            amazonlinux 32 k
Installing weak dependencies:
amazon-chrony-config noarch 4.2-7.amzn2023.0.4           amazonlinux 14 k
gawk-all-langpacks   aarch64 5.1.0-3.amzn2023.0.1          amazonlinux 207 k

Transaction Summary
=====
Install    5 Packages
Upgrade   413 Packages

Total download size: 199 M

```

Note

Si ha utilizado la variable de anulación `/etc/dnf/vars/releasever`, utilice el siguiente comando para restablecer el comportamiento de bloqueo predeterminado borrando el valor de anulación.

```
$ sudo rm /etc/dnf/vars/releasever
```

El uso de una modificación persistente para utilizar una versión `latest` en lugar de una específica es similar al comportamiento predeterminado de AL2. Hay servicios que se crean en AMIs función de los AL2, los cuales se deshabilita este comportamiento y se limitan a versiones de paquetes específicas, como las que se utilizan de forma predeterminada. AL2023

En lugar de deshabilitar las actualizaciones deterministas, recomendamos reemplazar las instancias por otras lanzadas desde una AMI nueva. Si el reemplazo de instancias no es una opción, le recomendamos que utilice herramientas como el [administrador de parches AWS Systems Manager](#) para organizar la aplicación de las actualizaciones en toda la flota. [EC2 Image Builder](#) también puede crear, aplicar parches y probar automáticamente sus AMIs propias imágenes AL2023 derivadas de las imágenes base. También puede usar [Recibir notificaciones sobre nuevas actualizaciones](#) para activar sus propias canalizaciones de creación de AMI.

El uso de `latest` en un entorno de preproducción y, posteriormente, su implementación en producción con `latest` no proporciona protección contra ningún problema entre una actualización del sistema operativo y la aplicación. Una nueva AL2023 versión puede publicarse en cualquier momento y, por lo tanto, todos los usos de la versión `latest` en producción conllevan riesgos.

Gestione las actualizaciones de paquetes y sistemas operativos en AL2023

A diferencia de las versiones anteriores de Amazon Linux, AL2023 AMIs están bloqueadas en una versión específica del repositorio de Amazon Linux. Para aplicar correcciones de seguridad y de errores a una AL2023 instancia, actualiza la DNF configuración a la última versión disponible. Como alternativa, lanza una AL2023 instancia más nueva.

En esta sección, se describe cómo administrar los paquetes DNF y los repositorios en una instancia en ejecución. También se describe cómo configurar DNF desde un script de datos de usuario para habilitar el último repositorio de Amazon Linux disponible en el momento del lanzamiento. Para obtener más información, consulte la [Referencia de los comandos DNF](#).

Se recomienda aplicar todas las actualizaciones disponibles en una AL2023 versión nueva. Elegir solo actualizaciones de seguridad o solo actualizaciones específicas debería ser la excepción y no la regla. Para ver una lista qué [Avisos de seguridad](#) son relevantes para una instancia concreta, consulte [Lista de avisos aplicables](#). Para obtener información sobre cómo instalar únicamente las actualizaciones relevantes para un [aviso](#) específico, consulte [Aplicación de actualizaciones de seguridad in situ](#).

Important

Si desea informar sobre una vulnerabilidad o tiene algún problema de seguridad relacionado con los servicios en la AWS nube o los proyectos de código abierto, póngase en contacto con el departamento de AWS seguridad a través de la [página de informes de vulnerabilidades](#)

Temas

- [Comprobación de las actualizaciones de los paquetes disponibles](#)
- [Aplicación de actualizaciones de seguridad mediante DNF y las versiones de repositorio](#)
- [Reinicio automático del servicio tras las actualizaciones \(de seguridad\)](#)
- [¿Cuándo es necesario reiniciar para aplicar las actualizaciones de seguridad?](#)
- [Lanzamiento de una instancia con la última versión del repositorio habilitada](#)
- [Obtener información de soporte del paquete](#)
- [Comprobación de las versiones más recientes del repositorio con `dnf check-release-update`](#)
- [Inclusión, habilitación y deshabilitación de nuevos repositorios](#)
- [Inclusión de repositorios con `cloud-init`](#)

Comprobación de las actualizaciones de los paquetes disponibles

Puede usar el comando `dnf check-update` para comprobar si hay actualizaciones para su sistema. Para AL2023 ello, le recomendamos que añada la `--releasever=version-number` opción al comando.

Al añadir esta opción, DNF también comprueba si hay actualizaciones para una versión posterior del repositorio. Por ejemplo, después de ejecutar el comando `dnf check-update`, utilice la última versión devuelta como valor para `version-number`.

Si la instancia se actualiza para usar la última versión del repositorio, en el resultado se incluye una lista de todos los paquetes que se van a actualizar.

Note

Si no especifica la versión de lanzamiento con el indicador opcional en el comando `dnf check-update`, sólo se marcará la versión del repositorio actualmente configurada. Esto significa que no se han comprobado los paquetes de la versión posterior del repositorio.

Updates in a specific version

En este ejemplo, analizaremos qué actualizaciones están disponibles en la versión [2023.1.20230628](#) si lanzamos un contenedor de la versión [2023.0.20230315](#).

Note

En este ejemplo se utilizan las versiones [2023.0.20230315](#) y [2023.1.20230628](#), y estas no son las versiones más recientes de Consulte las [notas de la versión para AL2023 ver las versiones más recientes, que contienen las actualizaciones de AL2023 seguridad](#) más recientes.

En este ejemplo, comenzaremos con una imagen de contenedor para la versión [2023.0.20230315](#).

En primer lugar, obtenemos esta imagen del contenedor del registro del contenedor. `.0` al final indica la versión de la imagen para una versión concreta; esta versión de imagen suele ser cero.

```
$ docker pull public.ecr.aws/amazonlinux/amazonlinux:2023.0.20230315.0
2023.0.20230315.0: Pulling from amazonlinux/amazonlinux
b76f3b09316a: Pull complete
Digest: sha256:94e7183b0739140dbd5b639fb7600f0a2299cec5df8780c26d9cb409da5315a9
Status: Downloaded newer image for public.ecr.aws/amazonlinux/
amazonlinux:2023.0.20230315.0
public.ecr.aws/amazonlinux/amazonlinux:2023.0.20230315.0
```

Ahora podemos generar un intérprete de comandos dentro del contenedor, desde el que comprobaremos si hay actualizaciones.

```
$ docker run -it public.ecr.aws/amazonlinux/amazonlinux:2023.0.20230315.0
bash-5.2#
```

El comando `dnf check-update` ahora se usa para comprobar las actualizaciones disponibles en la versión [2023.1.20230628](#).

Note

Aplicar actualizaciones de paquetes es una operación privilegiada. Aunque normalmente no es necesario aumentar los privilegios cuando se ejecuta en un contenedor, si se ejecuta en un entorno sin contenedores, como una instancia de Amazon EC2, puede comprobar si hay actualizaciones sin aumentar los privilegios.

```
$ dnf check-update --releasever=2023.1.20230628
```

```
Amazon Linux 2023 repository           60 MB/s | 15 MB      00:00
Last metadata expiration check: 0:00:02 ago on Mon Jul 22 17:25:34 2024.
```

amazon-linux-repo-cdn.noarch	2023.1.20230628-0.amzn2023	amazonlinux
ca-certificates.noarch	2023.2.60-1.0.amzn2023.0.2	amazonlinux
curl-minimal.x86_64	8.0.1-1.amzn2023	amazonlinux
glib2.x86_64	2.74.7-688.amzn2023.0.1	amazonlinux
glibc.x86_64	2.34-52.amzn2023.0.3	amazonlinux
glibc-common.x86_64	2.34-52.amzn2023.0.3	amazonlinux
glibc-minimal-langpack.x86_64	2.34-52.amzn2023.0.3	amazonlinux
gnupg2-minimal.x86_64	2.3.7-1.amzn2023.0.4	amazonlinux
keyutils-libs.x86_64	1.6.3-1.amzn2023	amazonlinux
libcap.x86_64	2.48-2.amzn2023.0.3	amazonlinux
libcurl-minimal.x86_64	8.0.1-1.amzn2023	amazonlinux
libgcc.x86_64	11.3.1-4.amzn2023.0.3	amazonlinux
libgomp.x86_64	11.3.1-4.amzn2023.0.3	amazonlinux
libstdc++.x86_64	11.3.1-4.amzn2023.0.3	amazonlinux
libxml2.x86_64	2.10.4-1.amzn2023.0.1	amazonlinux
ncurses-base.noarch	6.2-4.20200222.amzn2023.0.4	amazonlinux
ncurses-libs.x86_64	6.2-4.20200222.amzn2023.0.4	amazonlinux
openssl-libs.x86_64	1:3.0.8-1.amzn2023.0.3	amazonlinux
python3-rpm.x86_64	4.16.1.3-12.amzn2023.0.6	amazonlinux
rpm.x86_64	4.16.1.3-12.amzn2023.0.6	amazonlinux
rpm-build-libs.x86_64	4.16.1.3-12.amzn2023.0.6	amazonlinux
rpm-libs.x86_64	4.16.1.3-12.amzn2023.0.6	amazonlinux
rpm-sign-libs.x86_64	4.16.1.3-12.amzn2023.0.6	amazonlinux
system-release.noarch	2023.1.20230628-0.amzn2023	amazonlinux
tzdata.noarch	2023c-1.amzn2023.0.1	amazonlinux
bash-5.2#		

La versión del paquete `system-release` muestra la versión a la que se actualizaría un comando `dnf upgrade`, que es la versión [2023.1.20230628](#) que se solicitó en el comando `dnf check-update --releasever=2023.1.20230628`.

Updates in the latest version

[En este ejemplo, veremos qué actualizaciones están disponibles en la versión de si lanzamos un contenedor de la latest versión 2023.4.20240319. AL2023](#) En el momento de escribir este artículo, la versión latest es la [2023.5.20240708](#), por lo que las actualizaciones que se muestran en este ejemplo son las correspondientes a esa versión.

Note

En este ejemplo, se utilizan las versiones [2023.4.20240319](#) y [2023.5.20240708](#), siendo esta última la versión más reciente en el momento de redactar este artículo. [Para obtener más información sobre las versiones más recientes, consulta las notas de la versión. AL2023](#)

En este ejemplo, comenzaremos con una imagen de contenedor para la versión [2023.4.20240319](#).

En primer lugar, obtenemos esta imagen del contenedor del registro del contenedor. `.1` al final indica la versión de la imagen para una versión concreta. Aunque la versión de la imagen suele ser cero, este ejemplo utiliza una versión en la que la versión de la imagen es uno.

```
$ docker pull public.ecr.aws/amazonlinux/amazonlinux:2023.4.20240319.1
2023.4.20240319.1: Pulling from amazonlinux/amazonlinux
6de065fda9a2: Pull complete
Digest: sha256:b4838c4cc9211d966b6ea158dacc9eda7433a16ba94436508c2d9f01f7658b4e
Status: Downloaded newer image for public.ecr.aws/amazonlinux/
amazonlinux:2023.4.20240319.1
public.ecr.aws/amazonlinux/amazonlinux:2023.4.20240319.1
```

Ahora podemos generar un intérprete de comandos dentro del contenedor, desde el que comprobaremos si hay actualizaciones.

```
$ docker run -it public.ecr.aws/amazonlinux/amazonlinux:2023.4.20240319.1
bash-5.2#
```


El comando `dnf check-update` ahora se usa para comprobar las actualizaciones disponibles en la versión latest, que en el momento de escribir este artículo era la [2023.5.20240708](#).

Note

Aplicar actualizaciones de paquetes es una operación privilegiada. Aunque normalmente no es necesario aumentar los privilegios cuando se ejecuta en un contenedor, si se ejecuta en un entorno sin contenedores, como una instancia de Amazon EC2, puede comprobar si hay actualizaciones sin aumentar los privilegios.

```
$ dnf --releasever=latest check-update
```

```
Amazon Linux 2023 repository          78 MB/s | 25 MB      00:00
Last metadata expiration check: 0:00:04 ago on Mon Jul 22 17:39:13 2024.

amazon-linux-repo-cdn.noarch          2023.5.20240708-1.amzn2023      amazonlinux
curl-minimal.x86_64                  8.5.0-1.amzn2023.0.4            amazonlinux
dnf.noarch                            4.14.0-1.amzn2023.0.5           amazonlinux
dnf-data.noarch                      4.14.0-1.amzn2023.0.5           amazonlinux
expat.x86_64                         2.5.0-1.amzn2023.0.4            amazonlinux
glibc.x86_64                         2.34-52.amzn2023.0.10           amazonlinux
glibc-common.x86_64                 2.34-52.amzn2023.0.10           amazonlinux
glibc-minimal-langpack.x86_64        2.34-52.amzn2023.0.10           amazonlinux
krb5-libs.x86_64                    1.21-3.amzn2023.0.4             amazonlinux
libblkid.x86_64                     2.37.4-1.amzn2023.0.4           amazonlinux
libcurl-minimal.x86_64              8.5.0-1.amzn2023.0.4            amazonlinux
libmount.x86_64                     2.37.4-1.amzn2023.0.4           amazonlinux
libnghttp2.x86_64                   1.59.0-3.amzn2023.0.1           amazonlinux
libsmartcols.x86_64                 2.37.4-1.amzn2023.0.4           amazonlinux
libuuid.x86_64                      2.37.4-1.amzn2023.0.4           amazonlinux
openssl-libs.x86_64                 1:3.0.8-1.amzn2023.0.12         amazonlinux
python3.x86_64                      3.9.16-1.amzn2023.0.8           amazonlinux
python3-dnf.noarch                  4.14.0-1.amzn2023.0.5           amazonlinux
python3-libs.x86_64                 3.9.16-1.amzn2023.0.8           amazonlinux
system-release.noarch                2023.5.20240708-1.amzn2023      amazonlinux
yum.noarch                           4.14.0-1.amzn2023.0.5           amazonlinux
bash-5.2#
```

La versión del paquete `system-release` muestra la versión a la que se actualizaría un comando `dnf upgrade`.

Para este comando, si hay paquetes más recientes disponibles, el código de retorno es 100. Si no hay paquetes más recientes disponibles, el código de retorno es 0. Además, el resultado también muestra todos los paquetes que se van a actualizar.

Aplicación de actualizaciones de seguridad mediante DNF y las versiones de repositorio

Las nuevas actualizaciones de paquetes y de seguridad solo están disponibles para las nuevas versiones del repositorio. En el caso de las instancias que haya lanzado desde versiones anteriores de la AL2023 AMI, debe actualizar la versión del repositorio antes de poder instalar las actualizaciones de seguridad. El comando `dnf check-release-update` incluye un ejemplo de comando de actualización que actualiza todos los paquetes instalados en el sistema a versiones de un repositorio más reciente.

Note

Si no especifica la versión de lanzamiento con el indicador opcional en el comando `dnf check-update`, solo se marcará la versión del repositorio actualmente configurada. Esto significa que no se aplica ninguna actualización de los paquetes instalados que se encuentre en las versiones posteriores del repositorio.

En esta sección se describe la ruta de actualización recomendada, que consiste en aplicar todas las actualizaciones disponibles en lugar de seleccionar las actualizaciones individuales o solo las que estén marcadas como actualizaciones de seguridad. Al aplicar todas las actualizaciones, las instancias existentes se mueven al mismo conjunto de paquetes que al lanzar una AMI actualizada. Esta coherencia reduce la variación de las versiones de los paquetes en una flota. Para obtener más información, consulte [Aplicación de actualizaciones de seguridad in situ](#).

Applying updates in a specific version

En este ejemplo, vamos a aplicar las actualizaciones disponibles en la versión [2023.1.20230628](#) si lanzamos un contenedor de la versión [2023.0.20230315](#).

Note

En este ejemplo se utilizan las versiones [2023.0.20230315](#) y [2023.1.20230628](#), y no se trata de la versión más reciente de Consulte las [notas de la versión para AL2023 ver las](#)

[versiones más recientes, que contienen las actualizaciones de AL2023 seguridad](#) más recientes.

En este ejemplo, comenzaremos con una imagen de contenedor para la versión [2023.0.20230315](#).

En primer lugar, obtenemos esta imagen del contenedor del registro del contenedor. `.0` al final indica la versión de la imagen para una versión concreta; esta versión de imagen suele ser cero.

```
$ docker pull public.ecr.aws/amazonlinux/amazonlinux:2023.0.20230315.0
2023.0.20230315.0: Pulling from amazonlinux/amazonlinux
b76f3b09316a: Pull complete
Digest: sha256:94e7183b0739140dbd5b639fb7600f0a2299cec5df8780c26d9cb409da5315a9
Status: Downloaded newer image for public.ecr.aws/amazonlinux/
amazonlinux:2023.0.20230315.0
public.ecr.aws/amazonlinux/amazonlinux:2023.0.20230315.0
```

Ahora podemos generar un intérprete de comandos dentro del contenedor, desde el que aplicaremos las actualizaciones.

```
$ docker run -it public.ecr.aws/amazonlinux/amazonlinux:2023.0.20230315.0
bash-5.2#
```

El comando `dnf upgrade` ahora se usa para aplicar todas las actualizaciones presentes en la versión [2023.1.20230628](#).

Note

Aplicar actualizaciones de paquetes es una operación privilegiada. Aunque normalmente no es necesario elevar los privilegios cuando se ejecuta en un contenedor, si se ejecuta en un entorno sin contenedores, como una instancia de Amazon EC2, será necesario ejecutar el comando `dnf upgrade` como usuario `root`. Esto se puede hacer con los comandos `sudo` o `su`.

```
$ dnf upgrade --releasever=2023.1.20230628
Amazon Linux 2023 repository          38 MB/s | 15 MB    00:00
```

Last metadata expiration check: 0:00:02 ago on Mon Jul 22 17:49:08 2024.

Dependencies resolved.

Package	Arch	Version	Repository	Size
Upgrading:				
amazon-linux-repo-cdn	noarch	2023.1.20230628-0.amzn2023	amazonlinux	18 k
ca-certificates	noarch	2023.2.60-1.0.amzn2023.0.2	amazonlinux	829 k
curl-minimal	x86_64	8.0.1-1.amzn2023	amazonlinux	150 k
glib2	x86_64	2.74.7-688.amzn2023.0.1	amazonlinux	2.7 M
glibc	x86_64	2.34-52.amzn2023.0.3	amazonlinux	1.9 M
glibc-common	x86_64	2.34-52.amzn2023.0.3	amazonlinux	307 k
glibc-minimal-langpack	x86_64	2.34-52.amzn2023.0.3	amazonlinux	35 k
gnupg2-minimal	x86_64	2.3.7-1.amzn2023.0.4	amazonlinux	421 k
keyutils-libs	x86_64	1.6.3-1.amzn2023	amazonlinux	33 k
libcap	x86_64	2.48-2.amzn2023.0.3	amazonlinux	67 k
libcurl-minimal	x86_64	8.0.1-1.amzn2023	amazonlinux	249 k
libgcc	x86_64	11.3.1-4.amzn2023.0.3	amazonlinux	105 k
libgomp	x86_64	11.3.1-4.amzn2023.0.3	amazonlinux	280 k
libstdc++	x86_64	11.3.1-4.amzn2023.0.3	amazonlinux	744 k
libxml2	x86_64	2.10.4-1.amzn2023.0.1	amazonlinux	706 k
ncurses-base	noarch	6.2-4.20200222.amzn2023.0.4	amazonlinux	60 k
ncurses-libs	x86_64	6.2-4.20200222.amzn2023.0.4	amazonlinux	328 k
openssl-libs	x86_64	1:3.0.8-1.amzn2023.0.3	amazonlinux	2.2 M
python3-rpm	x86_64	4.16.1.3-12.amzn2023.0.6	amazonlinux	88 k
rpm	x86_64	4.16.1.3-12.amzn2023.0.6	amazonlinux	486 k
rpm-build-libs	x86_64	4.16.1.3-12.amzn2023.0.6	amazonlinux	90 k
rpm-libs	x86_64	4.16.1.3-12.amzn2023.0.6	amazonlinux	309 k
rpm-sign-libs	x86_64	4.16.1.3-12.amzn2023.0.6	amazonlinux	21 k
system-release	noarch	2023.1.20230628-0.amzn2023	amazonlinux	29 k
tzdata	noarch	2023c-1.amzn2023.0.1	amazonlinux	433 k

Transaction Summary

Upgrade 25 Packages

Total download size: 12 M

Is this ok [y/N]:

La versión del paquete `system-release` muestra la versión a la que se actualizaría un comando `dnf upgrade`, que es la versión [2023.1.20230628](#) que se solicitó en el comando `dnf upgrade --releasever=2023.1.20230628`.

De forma predeterminada, dnf le pedirá que confirme que desea aplicar las actualizaciones. Puede omitir este mensaje utilizando la marca `-y` para dnf. En este ejemplo, el comando `dnf upgrade -y --releasever=2023.1.20230628` no solicitaría confirmación antes de aplicar las actualizaciones. Esto resulta útil en scripts u otros entornos de automatización.

Una vez que confirme que desea aplicar las actualizaciones, dnf las aplica.

```
Is this ok [y/N]:y
  Downloading Packages:
(1/25): libcap-2.48-2.amzn2023.0.3.x86_64.rpm    1.5 MB/s | 67 kB    00:00
(2/25): python3-rpm-4.16.1.3-12.amzn2023.0.6.x86 2.1 MB/s | 88 kB    00:00
(3/25): libcurl-minimal-8.0.1-1.amzn2023.x86_64. 2.6 MB/s | 249 kB    00:00
(4/25): glib2-2.74.7-688.amzn2023.0.1.x86_64.rpm 26 MB/s | 2.7 MB    00:00
(5/25): glibc-minimal-langpack-2.34-52.amzn2023. 1.3 MB/s | 35 kB    00:00
(6/25): rpm-build-libs-4.16.1.3-12.amzn2023.0.6. 2.8 MB/s | 90 kB    00:00
(7/25): rpm-libs-4.16.1.3-12.amzn2023.0.6.x86_64 6.6 MB/s | 309 kB    00:00
(8/25): libgcc-11.3.1-4.amzn2023.0.3.x86_64.rpm 3.9 MB/s | 105 kB    00:00
(9/25): glibc-common-2.34-52.amzn2023.0.3.x86_64 11 MB/s | 307 kB    00:00
(10/25): glibc-2.34-52.amzn2023.0.3.x86_64.rpm   31 MB/s | 1.9 MB    00:00
(11/25): rpm-sign-libs-4.16.1.3-12.amzn2023.0.6. 877 kB/s | 21 kB    00:00
(12/25): gnupg2-minimal-2.3.7-1.amzn2023.0.4.x86 15 MB/s | 421 kB    00:00
(13/25): openssl-libs-3.0.8-1.amzn2023.0.3.x86_6 35 MB/s | 2.2 MB    00:00
(14/25): libxml2-2.10.4-1.amzn2023.0.1.x86_64.rp 14 MB/s | 706 kB    00:00
(15/25): curl-minimal-8.0.1-1.amzn2023.x86_64.rp 4.2 MB/s | 150 kB    00:00
(16/25): rpm-4.16.1.3-12.amzn2023.0.6.x86_64.rpm 11 MB/s | 486 kB    00:00
(17/25): libgomp-11.3.1-4.amzn2023.0.3.x86_64.rp 7.0 MB/s | 280 kB    00:00
(18/25): libstdc++-11.3.1-4.amzn2023.0.3.x86_64. 14 MB/s | 744 kB    00:00
(19/25): keyutils-libs-1.6.3-1.amzn2023.x86_64.r 1.6 MB/s | 33 kB    00:00
(20/25): ncurses-libs-6.2-4.20200222.amzn2023.0. 10 MB/s | 328 kB    00:00
(21/25): tzdata-2023c-1.amzn2023.0.1.noarch.rpm  11 MB/s | 433 kB    00:00
(22/25): amazon-linux-repo-cdn-2023.1.20230628-0 781 kB/s | 18 kB    00:00
(23/25): ca-certificates-2023.2.60-1.0.amzn2023. 16 MB/s | 829 kB    00:00
(24/25): system-release-2023.1.20230628-0.amzn20 1.5 MB/s | 29 kB    00:00
(25/25): ncurses-base-6.2-4.20200222.amzn2023.0. 3.1 MB/s | 60 kB    00:00
-----
Total                                     28 MB/s | 12 MB    00:00
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing          :                               1/1
  Upgrading          : libgcc-11.3.1-4.amzn2023.0.3.x86_64 1/50
```

Running scriptlet:	libgcc-11.3.1-4.amzn2023.0.3.x86_64	1/50
Upgrading	: system-release-2023.1.20230628-0.amzn2023.noarch	2/50
Upgrading	: amazon-linux-repo-cdn-2023.1.20230628-0.amzn2023.no	3/50
Upgrading	: ncurses-base-6.2-4.20200222.amzn2023.0.4.noarch	4/50
Upgrading	: tzdata-2023c-1.amzn2023.0.1.noarch	5/50
Upgrading	: glibc-common-2.34-52.amzn2023.0.3.x86_64	6/50
Running scriptlet:	glibc-2.34-52.amzn2023.0.3.x86_64	7/50
Upgrading	: glibc-2.34-52.amzn2023.0.3.x86_64	7/50
Running scriptlet:	glibc-2.34-52.amzn2023.0.3.x86_64	7/50
Upgrading	: glibc-minimal-langpack-2.34-52.amzn2023.0.3.x86_64	8/50
Upgrading	: libcap-2.48-2.amzn2023.0.3.x86_64	9/50
Upgrading	: gnupg2-minimal-2.3.7-1.amzn2023.0.4.x86_64	10/50
Upgrading	: libgomp-11.3.1-4.amzn2023.0.3.x86_64	11/50
Running scriptlet:	ca-certificates-2023.2.60-1.0.amzn2023.0.2.noarch	12/50
Upgrading	: ca-certificates-2023.2.60-1.0.amzn2023.0.2.noarch	12/50
Running scriptlet:	ca-certificates-2023.2.60-1.0.amzn2023.0.2.noarch	12/50
Upgrading	: openssl-libs-1:3.0.8-1.amzn2023.0.3.x86_64	13/50
Upgrading	: libcurl-minimal-8.0.1-1.amzn2023.x86_64	14/50
Upgrading	: curl-minimal-8.0.1-1.amzn2023.x86_64	15/50
Upgrading	: rpm-libs-4.16.1.3-12.amzn2023.0.6.x86_64	16/50
Upgrading	: rpm-4.16.1.3-12.amzn2023.0.6.x86_64	17/50
Upgrading	: rpm-build-libs-4.16.1.3-12.amzn2023.0.6.x86_64	18/50
Upgrading	: rpm-sign-libs-4.16.1.3-12.amzn2023.0.6.x86_64	19/50
Upgrading	: python3-rpm-4.16.1.3-12.amzn2023.0.6.x86_64	20/50
Upgrading	: glib2-2.74.7-688.amzn2023.0.1.x86_64	21/50
Upgrading	: libxml2-2.10.4-1.amzn2023.0.1.x86_64	22/50
Upgrading	: libstdc++-11.3.1-4.amzn2023.0.3.x86_64	23/50
Upgrading	: keyutils-libs-1.6.3-1.amzn2023.x86_64	24/50
Upgrading	: ncurses-libs-6.2-4.20200222.amzn2023.0.4.x86_64	25/50
Cleanup	: glib2-2.73.2-680.amzn2023.0.3.x86_64	26/50
Cleanup	: libstdc++-11.3.1-4.amzn2023.0.2.x86_64	27/50
Cleanup	: libxml2-2.10.3-2.amzn2023.0.1.x86_64	28/50
Cleanup	: python3-rpm-4.16.1.3-12.amzn2023.0.5.x86_64	29/50
Cleanup	: rpm-build-libs-4.16.1.3-12.amzn2023.0.5.x86_64	30/50
Cleanup	: rpm-sign-libs-4.16.1.3-12.amzn2023.0.5.x86_64	31/50
Cleanup	: rpm-libs-4.16.1.3-12.amzn2023.0.5.x86_64	32/50
Cleanup	: libcap-2.48-2.amzn2023.0.2.x86_64	33/50
Cleanup	: gnupg2-minimal-2.3.7-1.amzn2023.0.3.x86_64	34/50
Cleanup	: ncurses-libs-6.2-4.20200222.amzn2023.0.3.x86_64	35/50
Cleanup	: libgomp-11.3.1-4.amzn2023.0.2.x86_64	36/50
Cleanup	: rpm-4.16.1.3-12.amzn2023.0.5.x86_64	37/50
Cleanup	: curl-minimal-7.88.1-1.amzn2023.0.1.x86_64	38/50
Cleanup	: libcurl-minimal-7.88.1-1.amzn2023.0.1.x86_64	39/50
Cleanup	: openssl-libs-1:3.0.8-1.amzn2023.0.1.x86_64	40/50

Cleanup	: keyutils-libs-1.6.1-2.amzn2023.0.2.x86_64	41/50
Cleanup	: amazon-linux-repo-cdn-2023.0.20230315-1.amzn2023.no	42/50
Cleanup	: system-release-2023.0.20230315-1.amzn2023.noarch	43/50
Cleanup	: ca-certificates-2023.2.60-1.0.amzn2023.0.1.noarch	44/50
Cleanup	: ncurses-base-6.2-4.20200222.amzn2023.0.3.noarch	45/50
Cleanup	: glibc-minimal-langpack-2.34-52.amzn2023.0.2.x86_64	46/50
Cleanup	: glibc-2.34-52.amzn2023.0.2.x86_64	47/50
Cleanup	: glibc-common-2.34-52.amzn2023.0.2.x86_64	48/50
Cleanup	: tzdata-2022g-1.amzn2023.0.1.noarch	49/50
Cleanup	: libgcc-11.3.1-4.amzn2023.0.2.x86_64	50/50
Running scriptlet:	libgcc-11.3.1-4.amzn2023.0.2.x86_64	50/50
Running scriptlet:	ca-certificates-2023.2.60-1.0.amzn2023.0.2.noarch	50/50
Running scriptlet:	rpm-4.16.1.3-12.amzn2023.0.6.x86_64	50/50
Running scriptlet:	libgcc-11.3.1-4.amzn2023.0.2.x86_64	50/50
Verifying	: libcurl-minimal-8.0.1-1.amzn2023.x86_64	1/50
Verifying	: libcurl-minimal-7.88.1-1.amzn2023.0.1.x86_64	2/50
Verifying	: libcap-2.48-2.amzn2023.0.3.x86_64	3/50
Verifying	: libcap-2.48-2.amzn2023.0.2.x86_64	4/50
Verifying	: glib2-2.74.7-688.amzn2023.0.1.x86_64	5/50
Verifying	: glib2-2.73.2-680.amzn2023.0.3.x86_64	6/50
Verifying	: python3-rpm-4.16.1.3-12.amzn2023.0.6.x86_64	7/50
Verifying	: python3-rpm-4.16.1.3-12.amzn2023.0.5.x86_64	8/50
Verifying	: glibc-minimal-langpack-2.34-52.amzn2023.0.3.x86_64	9/50
Verifying	: glibc-minimal-langpack-2.34-52.amzn2023.0.2.x86_64	10/50
Verifying	: rpm-libs-4.16.1.3-12.amzn2023.0.6.x86_64	11/50
Verifying	: rpm-libs-4.16.1.3-12.amzn2023.0.5.x86_64	12/50
Verifying	: rpm-build-libs-4.16.1.3-12.amzn2023.0.6.x86_64	13/50
Verifying	: rpm-build-libs-4.16.1.3-12.amzn2023.0.5.x86_64	14/50
Verifying	: glibc-2.34-52.amzn2023.0.3.x86_64	15/50
Verifying	: glibc-2.34-52.amzn2023.0.2.x86_64	16/50
Verifying	: libgcc-11.3.1-4.amzn2023.0.3.x86_64	17/50
Verifying	: libgcc-11.3.1-4.amzn2023.0.2.x86_64	18/50
Verifying	: glibc-common-2.34-52.amzn2023.0.3.x86_64	19/50
Verifying	: glibc-common-2.34-52.amzn2023.0.2.x86_64	20/50
Verifying	: rpm-sign-libs-4.16.1.3-12.amzn2023.0.6.x86_64	21/50
Verifying	: rpm-sign-libs-4.16.1.3-12.amzn2023.0.5.x86_64	22/50
Verifying	: openssl-libs-1:3.0.8-1.amzn2023.0.3.x86_64	23/50
Verifying	: openssl-libs-1:3.0.8-1.amzn2023.0.1.x86_64	24/50
Verifying	: gnupg2-minimal-2.3.7-1.amzn2023.0.4.x86_64	25/50
Verifying	: gnupg2-minimal-2.3.7-1.amzn2023.0.3.x86_64	26/50
Verifying	: libxml2-2.10.4-1.amzn2023.0.1.x86_64	27/50
Verifying	: libxml2-2.10.3-2.amzn2023.0.1.x86_64	28/50
Verifying	: curl-minimal-8.0.1-1.amzn2023.x86_64	29/50
Verifying	: curl-minimal-7.88.1-1.amzn2023.0.1.x86_64	30/50

Verifying	: rpm-4.16.1.3-12.amzn2023.0.6.x86_64	31/50
Verifying	: rpm-4.16.1.3-12.amzn2023.0.5.x86_64	32/50
Verifying	: libstdc++-11.3.1-4.amzn2023.0.3.x86_64	33/50
Verifying	: libstdc++-11.3.1-4.amzn2023.0.2.x86_64	34/50
Verifying	: libgomp-11.3.1-4.amzn2023.0.3.x86_64	35/50
Verifying	: libgomp-11.3.1-4.amzn2023.0.2.x86_64	36/50
Verifying	: keyutils-libs-1.6.3-1.amzn2023.x86_64	37/50
Verifying	: keyutils-libs-1.6.1-2.amzn2023.0.2.x86_64	38/50
Verifying	: ncurses-libs-6.2-4.20200222.amzn2023.0.4.x86_64	39/50
Verifying	: ncurses-libs-6.2-4.20200222.amzn2023.0.3.x86_64	40/50
Verifying	: ca-certificates-2023.2.60-1.0.amzn2023.0.2.noarch	41/50
Verifying	: ca-certificates-2023.2.60-1.0.amzn2023.0.1.noarch	42/50
Verifying	: tzdata-2023c-1.amzn2023.0.1.noarch	43/50
Verifying	: tzdata-2022g-1.amzn2023.0.1.noarch	44/50
Verifying	: amazon-linux-repo-cdn-2023.1.20230628-0.amzn2023.no	45/50
Verifying	: amazon-linux-repo-cdn-2023.0.20230315-1.amzn2023.no	46/50
Verifying	: system-release-2023.1.20230628-0.amzn2023.noarch	47/50
Verifying	: system-release-2023.0.20230315-1.amzn2023.noarch	48/50
Verifying	: ncurses-base-6.2-4.20200222.amzn2023.0.4.noarch	49/50
Verifying	: ncurses-base-6.2-4.20200222.amzn2023.0.3.noarch	50/50

Upgraded:

```
amazon-linux-repo-cdn-2023.1.20230628-0.amzn2023.noarch
ca-certificates-2023.2.60-1.0.amzn2023.0.2.noarch
curl-minimal-8.0.1-1.amzn2023.x86_64
glib2-2.74.7-688.amzn2023.0.1.x86_64
glibc-2.34-52.amzn2023.0.3.x86_64
glibc-common-2.34-52.amzn2023.0.3.x86_64
glibc-minimal-langpack-2.34-52.amzn2023.0.3.x86_64
gnupg2-minimal-2.3.7-1.amzn2023.0.4.x86_64
keyutils-libs-1.6.3-1.amzn2023.x86_64
libcap-2.48-2.amzn2023.0.3.x86_64
libcurl-minimal-8.0.1-1.amzn2023.x86_64
libgcc-11.3.1-4.amzn2023.0.3.x86_64
libgomp-11.3.1-4.amzn2023.0.3.x86_64
libstdc++-11.3.1-4.amzn2023.0.3.x86_64
libxml2-2.10.4-1.amzn2023.0.1.x86_64
ncurses-base-6.2-4.20200222.amzn2023.0.4.noarch
ncurses-libs-6.2-4.20200222.amzn2023.0.4.x86_64
openssl-libs-1:3.0.8-1.amzn2023.0.3.x86_64
python3-rpm-4.16.1.3-12.amzn2023.0.6.x86_64
rpm-4.16.1.3-12.amzn2023.0.6.x86_64
rpm-build-libs-4.16.1.3-12.amzn2023.0.6.x86_64
rpm-libs-4.16.1.3-12.amzn2023.0.6.x86_64
```



```
rpm-sign-libs-4.16.1.3-12.amzn2023.0.6.x86_64
system-release-2023.1.20230628-0.amzn2023.noarch
tzdata-2023c-1.amzn2023.0.1.noarch
```

```
Complete!
bash-5.2#
```

Updates in the latest version

En este ejemplo, vamos a aplicar las actualizaciones disponibles en la versión de Si lanzamos un contenedor de la latest versión 2023.4.20240319. AL2023 En el momento de escribir este artículo, la versión latest es la [2023.5.20240708](#), por lo que las actualizaciones que se muestran en este ejemplo son las correspondientes a esa versión.

Note

En este ejemplo, se utilizan las versiones [2023.4.20240319](#) y [2023.5.20240708](#), siendo esta última la versión más reciente en el momento de redactar este artículo. [Para obtener más información sobre las versiones más recientes, consulta las notas de la versión. AL2023](#)

En este ejemplo, comenzaremos con una imagen de contenedor para la versión [2023.4.20240319](#).

En primer lugar, obtenemos esta imagen del contenedor del registro del contenedor. `.1` al final indica la versión de la imagen para una versión concreta. Aunque la versión de la imagen suele ser cero, este ejemplo utiliza una versión en la que la versión de la imagen es uno.

```
$ docker pull public.ecr.aws/amazonlinux/amazonlinux:2023.4.20240319.1
2023.4.20240319.1: Pulling from amazonlinux/amazonlinux
6de065fda9a2: Pull complete
Digest: sha256:b4838c4cc9211d966b6ea158dacc9eda7433a16ba94436508c2d9f01f7658b4e
Status: Downloaded newer image for public.ecr.aws/amazonlinux/
amazonlinux:2023.4.20240319.1
public.ecr.aws/amazonlinux/amazonlinux:2023.4.20240319.1
```

Ahora podemos generar un intérprete de comandos dentro del contenedor, desde el que aplicaremos las actualizaciones.

```
$ docker run -it public.ecr.aws/amazonlinux/amazonlinux:2023.4.20240319.1
```

```
bash-5.2#
```

El comando `dnf upgrade` ahora se usa para aplicar las actualizaciones disponibles en la versión `latest`, que en el momento de escribir este artículo era la [2023.5.20240708](#).

 Note

Aplicar actualizaciones de paquetes es una operación privilegiada. Aunque normalmente no es necesario elevar los privilegios cuando se ejecuta en un contenedor, si se ejecuta en un entorno sin contenedores, como una instancia de Amazon EC2, será necesario ejecutar el comando `dnf upgrade` como usuario `root`. Esto se puede hacer con los comandos `sudo` o `su`.

De forma predeterminada, `dnf` le pedirá que confirme que desea aplicar las actualizaciones. En este ejemplo, omitimos este mensaje utilizando la marca `-y` para `dnf`.

```
$ dnf -y --releasever=latest update
Amazon Linux 2023 repository              75 MB/s | 25 MB      00:00
Last metadata expiration check: 0:00:04 ago on Mon Jul 22 18:00:10 2024.
Dependencies resolved.
=====
Package                                Arch    Version                                Repository    Size
=====
Upgrading:
amazon-linux-repo-cdn                  noarch  2023.5.20240708-1.amzn2023            amazonlinux   17 k
curl-minimal                           x86_64  8.5.0-1.amzn2023.0.4                  amazonlinux   160 k
dnf                                     noarch  4.14.0-1.amzn2023.0.5                  amazonlinux   460 k
dnf-data                               noarch  4.14.0-1.amzn2023.0.5                  amazonlinux    34 k
expat                                   x86_64  2.5.0-1.amzn2023.0.4                  amazonlinux   117 k
glibc                                   x86_64  2.34-52.amzn2023.0.10                 amazonlinux   1.9 M
glibc-common                           x86_64  2.34-52.amzn2023.0.10                 amazonlinux   295 k
glibc-minimal-langpack                 x86_64  2.34-52.amzn2023.0.10                 amazonlinux    23 k
krb5-libs                              x86_64  1.21-3.amzn2023.0.4                   amazonlinux   758 k
libblkid                               x86_64  2.37.4-1.amzn2023.0.4                 amazonlinux   105 k
libcurl-minimal                        x86_64  8.5.0-1.amzn2023.0.4                  amazonlinux   275 k
libmount                               x86_64  2.37.4-1.amzn2023.0.4                 amazonlinux   132 k
libnghttp2                             x86_64  1.59.0-3.amzn2023.0.1                 amazonlinux    79 k
libsmartcols                           x86_64  2.37.4-1.amzn2023.0.4                 amazonlinux    62 k
libuuid                                x86_64  2.37.4-1.amzn2023.0.4                 amazonlinux    26 k
openssl-libs                           x86_64  1:3.0.8-1.amzn2023.0.12               amazonlinux   2.2 M
```

```
python3                x86_64  3.9.16-1.amzn2023.0.8      amazonlinux  27 k
python3-dnf            noarch  4.14.0-1.amzn2023.0.5      amazonlinux  409 k
python3-libs           x86_64  3.9.16-1.amzn2023.0.8      amazonlinux  7.3 M
system-release         noarch  2023.5.20240708-1.amzn2023  amazonlinux  28 k
yum                    noarch  4.14.0-1.amzn2023.0.5      amazonlinux  32 k

Transaction Summary
=====
Upgrade  21 Packages

Total download size: 14 M
Downloading Packages:
(1/21): amazon-linux-repo-cdn-2023.5.20240708-1. 345 kB/s | 17 kB    00:00
(2/21): dnf-4.14.0-1.amzn2023.0.5.noarch.rpm    6.8 MB/s | 460 kB    00:00
(3/21): dnf-data-4.14.0-1.amzn2023.0.5.noarch.rp 1.6 MB/s | 34 kB     00:00
(4/21): expat-2.5.0-1.amzn2023.0.4.x86_64.rpm   4.6 MB/s | 117 kB    00:00
(5/21): glibc-2.34-52.amzn2023.0.10.x86_64.rpm  38 MB/s | 1.9 MB     00:00
(6/21): glibc-common-2.34-52.amzn2023.0.10.x86_6 8.8 MB/s | 295 kB     00:00
(7/21): glibc-minimal-langpack-2.34-52.amzn2023. 1.7 MB/s | 23 kB     00:00
(8/21): curl-minimal-8.5.0-1.amzn2023.0.4.x86_64 998 kB/s | 160 kB     00:00
(9/21): libblkid-2.37.4-1.amzn2023.0.4.x86_64.rp 4.1 MB/s | 105 kB     00:00
(10/21): krb5-libs-1.21-3.amzn2023.0.4.x86_64.rp 16 MB/s | 758 kB     00:00
(11/21): libmount-2.37.4-1.amzn2023.0.4.x86_64.r 7.9 MB/s | 132 kB     00:00
(12/21): libnghttp2-1.59.0-3.amzn2023.0.1.x86_64 5.6 MB/s | 79 kB      00:00
(13/21): libsmartcols-2.37.4-1.amzn2023.0.4.x86_ 4.4 MB/s | 62 kB      00:00
(14/21): libcurl-minimal-8.5.0-1.amzn2023.0.4.x8 7.1 MB/s | 275 kB     00:00
(15/21): libuuid-2.37.4-1.amzn2023.0.4.x86_64.rp 1.1 MB/s | 26 kB      00:00
(16/21): python3-3.9.16-1.amzn2023.0.8.x86_64.rp 1.5 MB/s | 27 kB      00:00
(17/21): python3-dnf-4.14.0-1.amzn2023.0.5.noarc 19 MB/s | 409 kB     00:00
(18/21): system-release-2023.5.20240708-1.amzn20 1.9 MB/s | 28 kB      00:00
(19/21): yum-4.14.0-1.amzn2023.0.5.noarch.rpm    1.6 MB/s | 32 kB      00:00
(20/21): openssl-libs-3.0.8-1.amzn2023.0.12.x86_ 26 MB/s | 2.2 MB     00:00
(21/21): python3-libs-3.9.16-1.amzn2023.0.8.x86_ 59 MB/s | 7.3 MB     00:00
-----
Total                                34 MB/s | 14 MB     00:00
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing      : 1/1
  Upgrading      : glibc-common-2.34-52.amzn2023.0.10.x86_64 1/42
  Upgrading      : glibc-minimal-langpack-2.34-52.amzn2023.0.10.x86_64 2/42
  Running scriptlet: glibc-2.34-52.amzn2023.0.10.x86_64 3/42
```

Upgrading	: glibc-2.34-52.amzn2023.0.10.x86_64	3/42
Running scriptlet:	glibc-2.34-52.amzn2023.0.10.x86_64	3/42
Upgrading	: libuuid-2.37.4-1.amzn2023.0.4.x86_64	4/42
Upgrading	: openssl-libs-1:3.0.8-1.amzn2023.0.12.x86_64	5/42
Upgrading	: krb5-libs-1.21-3.amzn2023.0.4.x86_64	6/42
Upgrading	: libblkid-2.37.4-1.amzn2023.0.4.x86_64	7/42
Running scriptlet:	libblkid-2.37.4-1.amzn2023.0.4.x86_64	7/42
Upgrading	: expat-2.5.0-1.amzn2023.0.4.x86_64	8/42
Upgrading	: python3-3.9.16-1.amzn2023.0.8.x86_64	9/42
Upgrading	: python3-libs-3.9.16-1.amzn2023.0.8.x86_64	10/42
Upgrading	: libnghttp2-1.59.0-3.amzn2023.0.1.x86_64	11/42
Upgrading	: libcurl-minimal-8.5.0-1.amzn2023.0.4.x86_64	12/42
Upgrading	: system-release-2023.5.20240708-1.amzn2023.noarch	13/42
Upgrading	: amazon-linux-repo-cdn-2023.5.20240708-1.amzn2023.no	14/42
Upgrading	: dnf-data-4.14.0-1.amzn2023.0.5.noarch	15/42
Upgrading	: python3-dnf-4.14.0-1.amzn2023.0.5.noarch	16/42
Upgrading	: dnf-4.14.0-1.amzn2023.0.5.noarch	17/42
Running scriptlet:	dnf-4.14.0-1.amzn2023.0.5.noarch	17/42
Upgrading	: yum-4.14.0-1.amzn2023.0.5.noarch	18/42
Upgrading	: curl-minimal-8.5.0-1.amzn2023.0.4.x86_64	19/42
Upgrading	: libmount-2.37.4-1.amzn2023.0.4.x86_64	20/42
Upgrading	: libsmartcols-2.37.4-1.amzn2023.0.4.x86_64	21/42
Cleanup	: yum-4.14.0-1.amzn2023.0.4.noarch	22/42
Running scriptlet:	dnf-4.14.0-1.amzn2023.0.4.noarch	23/42
Cleanup	: dnf-4.14.0-1.amzn2023.0.4.noarch	23/42
Running scriptlet:	dnf-4.14.0-1.amzn2023.0.4.noarch	23/42
Cleanup	: python3-dnf-4.14.0-1.amzn2023.0.4.noarch	24/42
Cleanup	: amazon-linux-repo-cdn-2023.4.20240319-1.amzn2023.no	25/42
Cleanup	: libmount-2.37.4-1.amzn2023.0.3.x86_64	26/42
Cleanup	: curl-minimal-8.5.0-1.amzn2023.0.2.x86_64	27/42
Cleanup	: libcurl-minimal-8.5.0-1.amzn2023.0.2.x86_64	28/42
Cleanup	: krb5-libs-1.21-3.amzn2023.0.3.x86_64	29/42
Cleanup	: libblkid-2.37.4-1.amzn2023.0.3.x86_64	30/42
Cleanup	: libnghttp2-1.57.0-1.amzn2023.0.1.x86_64	31/42
Cleanup	: libsmartcols-2.37.4-1.amzn2023.0.3.x86_64	32/42
Cleanup	: system-release-2023.4.20240319-1.amzn2023.noarch	33/42
Cleanup	: dnf-data-4.14.0-1.amzn2023.0.4.noarch	34/42
Cleanup	: python3-3.9.16-1.amzn2023.0.6.x86_64	35/42
Cleanup	: python3-libs-3.9.16-1.amzn2023.0.6.x86_64	36/42
Cleanup	: openssl-libs-1:3.0.8-1.amzn2023.0.11.x86_64	37/42
Cleanup	: libuuid-2.37.4-1.amzn2023.0.3.x86_64	38/42
Cleanup	: expat-2.5.0-1.amzn2023.0.3.x86_64	39/42
Cleanup	: glibc-2.34-52.amzn2023.0.8.x86_64	40/42
Cleanup	: glibc-minimal-langpack-2.34-52.amzn2023.0.8.x86_64	41/42

Cleanup	: glibc-common-2.34-52.amzn2023.0.8.x86_64	42/42
Running scriptlet:	glibc-common-2.34-52.amzn2023.0.8.x86_64	42/42
Verifying	: amazon-linux-repo-cdn-2023.5.20240708-1.amzn2023.no	1/42
Verifying	: amazon-linux-repo-cdn-2023.4.20240319-1.amzn2023.no	2/42
Verifying	: curl-minimal-8.5.0-1.amzn2023.0.4.x86_64	3/42
Verifying	: curl-minimal-8.5.0-1.amzn2023.0.2.x86_64	4/42
Verifying	: dnf-4.14.0-1.amzn2023.0.5.noarch	5/42
Verifying	: dnf-4.14.0-1.amzn2023.0.4.noarch	6/42
Verifying	: dnf-data-4.14.0-1.amzn2023.0.5.noarch	7/42
Verifying	: dnf-data-4.14.0-1.amzn2023.0.4.noarch	8/42
Verifying	: expat-2.5.0-1.amzn2023.0.4.x86_64	9/42
Verifying	: expat-2.5.0-1.amzn2023.0.3.x86_64	10/42
Verifying	: glibc-2.34-52.amzn2023.0.10.x86_64	11/42
Verifying	: glibc-2.34-52.amzn2023.0.8.x86_64	12/42
Verifying	: glibc-common-2.34-52.amzn2023.0.10.x86_64	13/42
Verifying	: glibc-common-2.34-52.amzn2023.0.8.x86_64	14/42
Verifying	: glibc-minimal-langpack-2.34-52.amzn2023.0.10.x86_64	15/42
Verifying	: glibc-minimal-langpack-2.34-52.amzn2023.0.8.x86_64	16/42
Verifying	: krb5-libs-1.21-3.amzn2023.0.4.x86_64	17/42
Verifying	: krb5-libs-1.21-3.amzn2023.0.3.x86_64	18/42
Verifying	: libblkid-2.37.4-1.amzn2023.0.4.x86_64	19/42
Verifying	: libblkid-2.37.4-1.amzn2023.0.3.x86_64	20/42
Verifying	: libcurl-minimal-8.5.0-1.amzn2023.0.4.x86_64	21/42
Verifying	: libcurl-minimal-8.5.0-1.amzn2023.0.2.x86_64	22/42
Verifying	: libmount-2.37.4-1.amzn2023.0.4.x86_64	23/42
Verifying	: libmount-2.37.4-1.amzn2023.0.3.x86_64	24/42
Verifying	: libnghttp2-1.59.0-3.amzn2023.0.1.x86_64	25/42
Verifying	: libnghttp2-1.57.0-1.amzn2023.0.1.x86_64	26/42
Verifying	: libsmartcols-2.37.4-1.amzn2023.0.4.x86_64	27/42
Verifying	: libsmartcols-2.37.4-1.amzn2023.0.3.x86_64	28/42
Verifying	: libuuid-2.37.4-1.amzn2023.0.4.x86_64	29/42
Verifying	: libuuid-2.37.4-1.amzn2023.0.3.x86_64	30/42
Verifying	: openssl-libs-1:3.0.8-1.amzn2023.0.12.x86_64	31/42
Verifying	: openssl-libs-1:3.0.8-1.amzn2023.0.11.x86_64	32/42
Verifying	: python3-3.9.16-1.amzn2023.0.8.x86_64	33/42
Verifying	: python3-3.9.16-1.amzn2023.0.6.x86_64	34/42
Verifying	: python3-dnf-4.14.0-1.amzn2023.0.5.noarch	35/42
Verifying	: python3-dnf-4.14.0-1.amzn2023.0.4.noarch	36/42
Verifying	: python3-libs-3.9.16-1.amzn2023.0.8.x86_64	37/42
Verifying	: python3-libs-3.9.16-1.amzn2023.0.6.x86_64	38/42
Verifying	: system-release-2023.5.20240708-1.amzn2023.noarch	39/42
Verifying	: system-release-2023.4.20240319-1.amzn2023.noarch	40/42
Verifying	: yum-4.14.0-1.amzn2023.0.5.noarch	41/42
Verifying	: yum-4.14.0-1.amzn2023.0.4.noarch	42/42

Upgraded:

```
amazon-linux-repo-cdn-2023.5.20240708-1.amzn2023.noarch
curl-minimal-8.5.0-1.amzn2023.0.4.x86_64
dnf-4.14.0-1.amzn2023.0.5.noarch
dnf-data-4.14.0-1.amzn2023.0.5.noarch
expat-2.5.0-1.amzn2023.0.4.x86_64
glibc-2.34-52.amzn2023.0.10.x86_64
glibc-common-2.34-52.amzn2023.0.10.x86_64
glibc-minimal-langpack-2.34-52.amzn2023.0.10.x86_64
krb5-libs-1.21-3.amzn2023.0.4.x86_64
libblkid-2.37.4-1.amzn2023.0.4.x86_64
libcurl-minimal-8.5.0-1.amzn2023.0.4.x86_64
libmount-2.37.4-1.amzn2023.0.4.x86_64
libnghttp2-1.59.0-3.amzn2023.0.1.x86_64
libsmartcols-2.37.4-1.amzn2023.0.4.x86_64
libuuid-2.37.4-1.amzn2023.0.4.x86_64
openssl-libs-1:3.0.8-1.amzn2023.0.12.x86_64
python3-3.9.16-1.amzn2023.0.8.x86_64
python3-dnf-4.14.0-1.amzn2023.0.5.noarch
python3-libs-3.9.16-1.amzn2023.0.8.x86_64
system-release-2023.5.20240708-1.amzn2023.noarch
yum-4.14.0-1.amzn2023.0.5.noarch
```

Complete!

bash-5.2#

Para descubrir AL2023 las actualizaciones, realice una o varias de las siguientes acciones:

- Ejecute el comando `dnf check-update`. Esto comprueba si hay actualizaciones no aplicadas en la versión de Amazon Linux a la que está bloqueado. Es posible que se muestren las actualizaciones si solo actualizó el paquete `system-release`, moviendo la versión de los repositorios a la que está bloqueada la instancia, pero sin aplicar ninguna de las actualizaciones disponibles en ella.
- Suscríbase al tema SNS sobre la actualización del repositorio de Amazon Linux (`arn:aws:sns:us-east-1:137112412989:amazon-linux-2023-ami-updates`). Para obtener más información, consulte [Suscripción a un tema de Amazon SNS](#) en la Guía para desarrolladores de Amazon Simple Notification Service.
- Consulte periódicamente las [notas de la AL2023 versión](#).

- Descubra las nuevas versiones mediante [Comprobación de las versiones más recientes del repositorio con `dnf check-release-update`](#).

 Important

Con frecuencia se publican nuevas versiones de las actualizaciones de seguridad AL2023 que contienen actualizaciones de seguridad. Asegúrese de mantenerse al día con los parches de seguridad pertinentes.

Reinicio automático del servicio tras las actualizaciones (de seguridad)

Amazon Linux ahora incluye el paquete [smart-restart](#). `Smart-restart` reinicia los servicios de `systemd` en las actualizaciones del sistema cada vez que se instala o elimina un paquete mediante el administrador de paquetes del sistema. Esto ocurre siempre que se ejecuta `dnf (update | upgrade | downgrade)`.

`Smart-restart` utiliza el paquete `needs-restarting` de `dnf-utils` y un mecanismo de registro de denegación personalizado para determinar qué servicios deben reiniciarse y si es aconsejable reiniciar el sistema. Si se recomienda reiniciar el sistema, se genera un archivo de sugerencias de reinicio (`/run/smart-restart/reboot-hint-marker`).

Para instalar **smart-restart**

Ejecute el siguiente comando DNF (al igual que con los demás paquetes).

```
$ sudo dnf install smart-restart
```

Tras la instalación, las transacciones subsiguientes activarán la lógica `smart-restart`.

Lista de denegados

`Smart-restart` puede recibir instrucciones para bloquear el reinicio de ciertos servicios. Los servicios bloqueados no contribuirán a decidir si es necesario reiniciar el sistema. Para bloquear servicios adicionales, añada un archivo con el sufijo `-denylist` en `/etc/smart-restart-conf.d/` como se muestra en el siguiente ejemplo.

```
$ cat /etc/smart-restart-conf.d/custom-denylist
```

```
# Some comments
myservice.service
```

Note

Todos los archivos `*-denylist` se leen y evalúan al decidir si es necesario reiniciar el equipo.

Enlaces personalizados

Además de la lista de denegados, `smart-restart` proporciona un mecanismo para ejecutar scripts personalizados antes y después de intentar reiniciar el servicio. Los scripts personalizados se pueden usar para realizar manualmente los pasos de preparación o para informar a otros componentes de un reinicio pendiente o completado.

Todos los scripts de `/etc/smart-restart-conf.d/` que llevan el sufijo `-pre-restart` o `-post-restart` se ejecutan. Si el orden es importante, añada un número a todos los scripts para garantizar el orden de ejecución, como se muestra en el siguiente ejemplo.

```
$ ls /etc/smart-restart-conf.d/*-pre-restart
001-my-script-pre-restart
002-some-other-script-pre-restart
```

¿Cuándo es necesario reiniciar para aplicar las actualizaciones de seguridad?

En algunas situaciones, Amazon Linux requiere un reinicio para aplicar las actualizaciones:

- Las actualizaciones del paquete del kernel de Linux requieren un reinicio para activar el nuevo kernel con las actualizaciones de seguridad más recientes. Los parches activos del kernel le pueden permitir posponer las actualizaciones de seguridad durante un periodo limitado. Para obtener más información, consulte [Aplicación de parches activos del kernel en AL2023](#).
- En las instancias EC2 Metal, Amazon Linux proporciona actualizaciones de microcódigos (a través del `microcode_ctl` paquete para Intel CPUs y el `amd-ucode-firmware` paquete para las CPU de AMD). Estas actualizaciones de microcódigos solo se activarán en los reinicios posteriores de la instancia. En el caso de las instancias de EC2 virtualizadas, el [sistema AWS Nitro](#) subyacente se encarga de las actualizaciones de microcódigos por usted.

- Algunos servicios de systemd en ejecución solo funcionarán correctamente tras un reinicio completo del sistema. El mecanismo `smart-restart` le informará sobre estas situaciones dejando sugerencias de reinicio. Consulte [Reinicio automático del servicio tras las actualizaciones \(de seguridad\)](#).

Lanzamiento de una instancia con la última versión del repositorio habilitada

Puede agregar comandos DNF a un script de datos de usuario para controlar qué paquetes RPM se instalan en una AMI de Amazon Linux cuando se lanza. En el siguiente ejemplo, se utiliza un script de datos de usuario para garantizar que cualquier instancia lanzada con el script de datos de usuario tenga instaladas las mismas actualizaciones de paquetes.

```
#!/bin/bash
dnf upgrade --releasever=2023.0.20230210
# Additional setup and install commands below
dnf install httpd php7.4 mysql80
```

Debe ejecutar este script como superusuario (raíz). Para ello, ejecute el siguiente comando.

```
$ sudo sh -c "bash nameofscript.sh"
```

Para obtener más información, consulte [Datos de usuario y scripts de intérprete de comandos](#) en la Guía del usuario de Amazon EC2.

Note

En lugar de utilizar un script de datos de usuario, lance la última AMI de Amazon Linux o una AMI personalizada basada en la AMI de Amazon Linux. La AMI más reciente de Amazon Linux tiene todas las actualizaciones necesarias instaladas y está configurada para apuntar a una versión de repositorio concreta.

Obtener información de soporte del paquete

AL2023 incorpora muchos proyectos diferentes de software de código abierto. Cada uno de estos proyectos se gestiona de forma independiente de Amazon Linux y tienen versiones y end-of-support cronogramas diferentes. Para proporcionarle información específica de Amazon Linux sobre estos

diferentes paquetes, el complemento DNF `supportinfo` proporciona metadatos sobre cada paquete. En el siguiente ejemplo, el **`dnf supportinfo`** comando devuelve los metadatos `glibc` del paquete.

```
$ sudo dnf supportinfo --pkg glibc
Last metadata expiration check: 0:07:56 ago on Wed Mar 1 23:21:49 2023.
Name                : glibc
Version             : 2.34-52.amzn2023.0.2
State               : installed
Support Status      : supported
Support Periods     : from 2023-03-15      : supported
                   : from 2028-03-15      : unsupported
Support Statement   : Amazon Linux 2023 End Of Life
Link               : https://aws.amazon.com/amazon-linux-ami/faqs/
Other Info          : This is the support statement for AL2023. The
                   ...: end of life of Amazon Linux 2023 would be March 2028.
                   ...: From this point, the Amazon Linux 2023 packages (listed
                   ...: below) will no longer, receive any updates from AWS.
```

La información de soporte de los paquetes también está disponible en la sección de [declaraciones de soporte](#) de las [notas de la AL2023 versión](#).

Comprobación de las versiones más recientes del repositorio con **`dnf check-release-update`**

En un AL2023 caso, puede utilizar la DNF utilidad para gestionar los repositorios y aplicar los RPM paquetes actualizados. Estos paquetes están disponibles en los repositorios de Amazon Linux. Puede usar el comando DNF `dnf check-release-update` para comprobar si hay nuevas versiones del repositorio DNF.

Note

AL2023 Las imágenes del contenedor no incluyen el `dnf check-release-update` comando de forma predeterminada.

```
$ dnf check-release-update
```

```
No such command: check-release-update. Please use /usr/bin/dnf --help
It could be a DNF plugin command, try: "dnf install 'dnf-command(check-release-update)'"
```

Cuando `dnf install 'dnf-command(check-release-update)'` se ejecute, `dnf` instalará el paquete que proporciona el comando `check-release-update`, que es el paquete `dnf-plugin-release-notification`. En el siguiente ejemplo, se da el argumento `-q` a `dnf` para que tenga una salida silenciosa.

```
$ dnf -y -q install 'dnf-command(check-release-update)'
Installed:
  dnf-plugin-release-notification-1.2-1.amzn2023.0.2.noarch
```

En entornos sin contenedores, como una instancia de Amazon EC2, el comando `check-release-update` se incluye de forma predeterminada.

```
$ sudo dnf check-release-update
WARNING:
  A newer release of "Amazon Linux" is available.

Available Versions:

Version 2023.0.20230210:
  Run the following command to update to 2023.0.20230210:

    dnf upgrade --releasever=2023.0.20230210

Release notes:
  https://docs.aws.amazon.com/linux/al2023/release-notes/relnotes.html
```

Esto devuelve una lista completa de todas las versiones más recientes de los repositorios DNF que están disponibles. Si no se devuelve nada, significa que DNF está configurado actualmente para usar la última versión disponible. La versión del paquete `system-release` actualmente instalado establece la variable `releasever` DNF. Para comprobar la versión actual del repositorio, ejecute el siguiente comando.

```
$ rpm -q system-release --qf "%{VERSION}\n"
```

Cuando ejecuta transacciones de paquetes DNF (como comandos de instalación, actualización o eliminación), un mensaje de advertencia avisa de cualquier nueva versión del repositorio. Por

ejemplo, si instalas el `httpd` paquete en una instancia que se lanzó desde una versión anterior de AL2023, se obtiene el siguiente resultado.

```
$ sudo dnf install httpd -y
Last metadata expiration check: 0:16:52 ago on Wed Mar 1 23:21:49 2023.
Dependencies resolved.
=====
Package                Arch    Version                               Repository    Size
=====
Installing:
httpd                   x86_64  2.4.54-3.amzn2023.0.4               amazonlinux   46 k
Installing dependencies:
apr                     x86_64  1.7.2-2.amzn2023.0.2               amazonlinux   129 k
apr-util                x86_64  1.6.3-1.amzn2023.0.1               amazonlinux   98 k
generic-logos-httpd
noarch                 18.0.0-12.amzn2023.0.3             amazonlinux   19 k
httpd-core              x86_64  2.4.54-3.amzn2023.0.4               amazonlinux   1.3 M
httpd-filesystem        noarch  2.4.54-3.amzn2023.0.4               amazonlinux   13 k
httpd-tools             x86_64  2.4.54-3.amzn2023.0.4               amazonlinux   80 k
libbrotli               x86_64  1.0.9-4.amzn2023.0.2               amazonlinux   315 k
mailcap                 noarch  2.1.49-3.amzn2023.0.3               amazonlinux   33 k
Installing weak dependencies:
apr-util-openssl        x86_64  1.6.3-1.amzn2023.0.1               amazonlinux   17 k
mod_http2               x86_64  1.15.24-1.amzn2023.0.3             amazonlinux   152 k
mod_lua                 x86_64  2.4.54-3.amzn2023.0.4               amazonlinux   60 k

Transaction Summary
=====
Install 12 Packages

Total download size: 2.3 M
Installed size: 6.8 M
Downloading Packages:
(1/12): apr-util-openssl-1.6.3-1.am 212 kB/s | 17 kB      00:00
(2/12): apr-1.7.2-2.amzn2023.0.2.x8 1.1 MB/s | 129 kB     00:00
(3/12): httpd-core-2.4.54-3.amzn202 8.9 MB/s | 1.3 MB     00:00
(4/12): mod_http2-1.15.24-1.amzn202 1.9 MB/s | 152 kB     00:00
(5/12): apr-util-1.6.3-1.amzn2023.0 1.7 MB/s | 98 kB      00:00
(6/12): mod_lua-2.4.54-3.amzn2023.0 1.4 MB/s | 60 kB      00:00
(7/12): httpd-2.4.54-3.amzn2023.0.4 1.5 MB/s | 46 kB      00:00
(8/12): libbrotli-1.0.9-4.amzn2023. 4.4 MB/s | 315 kB     00:00
(9/12): mailcap-2.1.49-3.amzn2023.0 753 kB/s | 33 kB      00:00
(10/12): httpd-tools-2.4.54-3.amzn2 978 kB/s | 80 kB      00:00
```

```

(11/12): httpd-filesystem-2.4.54-3. 210 kB/s | 13 kB      00:00
(12/12): generic-logos-httpd-18.0.0 439 kB/s | 19 kB      00:00
-----
Total                                6.6 MB/s | 2.3 MB      00:00
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing      :                                1/1
  Installing     : apr-1.7.2-2.amzn2023.0.2.x86_64 1/12
  Installing     : apr-util-openssl-1.6.3-1.amzn2023.0.1. 2/12
  Installing     : apr-util-1.6.3-1.amzn2023.0.1.x86_64 3/12
  Installing     : mailcap-2.1.49-3.amzn2023.0.3.noarch 4/12
  Installing     : httpd-tools-2.4.54-3.amzn2023.0.4.x86_ 5/12
  Installing     : generic-logos-httpd-18.0.0-12.amzn2023 6/12
Running scriptlet: httpd-filesystem-2.4.54-3.amzn2023.0.4 7/12
  Installing     : httpd-filesystem-2.4.54-3.amzn2023.0.4 7/12
  Installing     : httpd-core-2.4.54-3.amzn2023.0.4.x86_6 8/12
  Installing     : mod_http2-1.15.24-1.amzn2023.0.3.x86_6 9/12
  Installing     : libbrotli-1.0.9-4.amzn2023.0.2.x86_64 10/12
  Installing     : mod_lua-2.4.54-3.amzn2023.0.4.x86_64 11/12
  Installing     : httpd-2.4.54-3.amzn2023.0.4.x86_64 12/12
Running scriptlet: httpd-2.4.54-3.amzn2023.0.4.x86_64 12/12
  Verifying     : apr-1.7.2-2.amzn2023.0.2.x86_64 1/12
  Verifying     : apr-util-openssl-1.6.3-1.amzn2023.0.1. 2/12
  Verifying     : httpd-core-2.4.54-3.amzn2023.0.4.x86_6 3/12
  Verifying     : mod_http2-1.15.24-1.amzn2023.0.3.x86_6 4/12
  Verifying     : apr-util-1.6.3-1.amzn2023.0.1.x86_64 5/12
  Verifying     : mod_lua-2.4.54-3.amzn2023.0.4.x86_64 6/12
  Verifying     : libbrotli-1.0.9-4.amzn2023.0.2.x86_64 7/12
  Verifying     : httpd-2.4.54-3.amzn2023.0.4.x86_64 8/12
  Verifying     : httpd-tools-2.4.54-3.amzn2023.0.4.x86_ 9/12
  Verifying     : mailcap-2.1.49-3.amzn2023.0.3.noarch 10/12
  Verifying     : httpd-filesystem-2.4.54-3.amzn2023.0.4 11/12
  Verifying     : generic-logos-httpd-18.0.0-12.amzn2023 12/12

Installed:
  apr-1.7.2-2.amzn2023.0.2.x86_64
  apr-util-1.6.3-1.amzn2023.0.1.x86_64
  apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64
  generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch
  httpd-2.4.54-3.amzn2023.0.4.x86_64
  httpd-core-2.4.54-3.amzn2023.0.4.x86_64

```

```
httpd-filesystem-2.4.54-3.amzn2023.0.4.noarch
httpd-tools-2.4.54-3.amzn2023.0.4.x86_64
libbrotli-1.0.9-4.amzn2023.0.2.x86_64
mailcap-2.1.49-3.amzn2023.0.3.noarch
mod_http2-1.15.24-1.amzn2023.0.3.x86_64
mod_lua-2.4.54-3.amzn2023.0.4.x86_64
```

Complete!

Inclusión, habilitación y deshabilitación de nuevos repositorios

Warning

Agrega únicamente los repositorios diseñados para usarse con AL2023.

Si bien los repositorios diseñados para otras distribuciones pueden funcionar hoy en día, no hay garantía de que continúen haciéndolo con cualquier actualización de paquete AL2023 o con el repositorio no diseñado para su uso. AL2023

Para instalar un paquete desde un repositorio diferente al de los repositorios predeterminados de Amazon Linux, necesitará configurar el sistema de administración de paquetes DNF para saber dónde está el repositorio.

Para indicar a `dnf` información sobre un repositorio de paquetes, añada la información del repositorio a un archivo de configuración para ese repositorio en el directorio `/etc/yum.repos.d/`. Muchos repositorios de terceros proporcionan el contenido del archivo de configuración o un paquete instalable que incluye el archivo de configuración.

Note

Aunque los repositorios se pueden configurar directamente en el archivo `/etc/dnf/dnf.conf`, no es recomendable. Se recomienda configurar cada repositorio en su propio archivo `/etc/yum.repos.d/`.

Para saber qué repositorios están habilitados actualmente, puede ejecutar el siguiente comando:

```
$ dnf repolist all --verbose
```

```
Loaded plugins: builddep, changelog, config-manager, copr, debug, debuginfo-install,
download, generate_completion_cache, groups-manager, needs-restarting, playground,
```

```

release-notification, repoclosure, repodiff, repograph, repomanage, reposync,
supportinfo
DNF version: 4.12.0
cachedir: /var/cache/dnf
Last metadata expiration check: 0:00:02 ago on Wed Mar  1 23:40:15 2023.
Repo-id           : amazonlinux
Repo-name         : Amazon Linux 2023 repository
Repo-status      : enabled
Repo-revision     : 1677203368
Repo-updated      : Fri Feb 24 01:49:28 2023
Repo-pkgs        : 12632
Repo-available-pkgs: 12632
Repo-size        : 12 G
Repo-mirrors      : https://al2023-repos-us-west-2-de612dc2.s3.dualstack.us-
west-2.amazonaws.com/core/mirrors/2023.0.20230222/x86_64/mirror.list
Repo-baseurl      : https://al2023-repos-us-west-2-de612dc2.s3.dualstack.us-
west-2.amazonaws.com/core/guids/
cf9296325a6c46ff40c775a8e2d632c4c3fd9d9164014ce3304715d61b90ca8e/x86_64/
                  : (0 more)
Repo-expire       : 172800 second(s) (last: Wed Mar  1 23:40:15
                  : 2023)
Repo-filename     : /etc/yum.repos.d/amazonlinux.repo

Repo-id           : amazonlinux-debuginfo
Repo-name         : Amazon Linux 2023 repository - Debug
Repo-status      : disabled
Repo-mirrors      : https://al2023-repos-us-west-2-de612dc2.s3.dualstack.us-
west-2.amazonaws.com/core/mirrors/2023.0.20230222/debuginfo/x86_64/mirror.list
Repo-expire       : 21600 second(s) (last: unknown)
Repo-filename     : /etc/yum.repos.d/amazonlinux.repo

Repo-id           : amazonlinux-source
Repo-name         : Amazon Linux 2023 repository - Source packages
Repo-status      : disabled
Repo-mirrors      : https://al2023-repos-us-west-2-de612dc2.s3.dualstack.us-
west-2.amazonaws.com/core/mirrors/2023.0.20230222/SRPMS/mirror.list
Repo-expire       : 21600 second(s) (last: unknown)
Repo-filename     : /etc/yum.repos.d/amazonlinux.repo

Repo-id           : kernel-livepatch
Repo-name         : Amazon Linux 2023 Kernel Livepatch repository
Repo-status      : disabled
Repo-mirrors      : https://al2023-repos-us-west-2-de612dc2.s3.dualstack.us-
west-2.amazonaws.com/kernel-livepatch/mirrors/al2023/x86_64/mirror.list

```

```
Repo-expire      : 172800 second(s) (last: unknown)
Repo-filename    : /etc/yum.repos.d/kernel-livepatch.repo

Repo-id          : kernel-livepatch-source
Repo-name        : Amazon Linux 2023 Kernel Livepatch repository -
                  : Source packages
Repo-status      : disabled
Repo-mirrors     : https://al2023-repos-us-west-2-de612dc2.s3.dualstack.us-
west-2.amazonaws.com/kernel-livepatch/mirrors/al2023/SRPMS/mirror.list
Repo-expire      : 21600 second(s) (last: unknown)
Repo-filename    : /etc/yum.repos.d/kernel-livepatch.repo
Total packages: 12632
```

Note

Si no agregase el indicador de la opción `--verbose`, la salida solo incluirá la información `Repo-id`, `Repo-name` y `Repo-status`.

Para añadir un repositorio **yum** al directorio `/etc/yum.repos.d`:

1. Busque la ubicación del archivo `.repo`. En este ejemplo, el archivo `.repo` se encuentra en <https://www.example.com/repository.repo>.
2. Añada el repositorio con el comando `dnf config-manager`.

```
$ sudo dnf config-manager --add-repo https://www.example.com/repository.repo
Loaded plugins: priorities, update-motd, upgrade-helper
adding repo from: https://www.example.com/repository.repo
grabbing file https://www.example.com/repository.repo to /etc/
yum.repos.d/repository.repo
repository.repo | 4.0 kB      00:00
repo saved to /etc/yum.repos.d/repository.repo
```

Después de instalar un repositorio, debe habilitarlo como se describe en el procedimiento siguiente.

Para habilitar un yum repositorio/`/etc/yum.repos.d`, usa el `dnf config-manager` comando con la `--enable` marca y *repository* el nombre.

```
$ sudo dnf config-manager --enable repository
```


 Note

Para deshabilitar un repositorio, utilice la misma sintaxis de comando, pero sustituya `--enable` por `--disable` en el comando.

Inclusión de repositorios con cloud-init

Además de añadir un repositorio mediante el método anterior, también puede añadir un repositorio nuevo mediante el marco `cloud-init`.

Para añadir un repositorio de paquetes nuevo, se recomienda utilizar la siguiente plantilla. Considere la posibilidad de guardar este archivo localmente.

```
#cloud-config
yum_repos:
  repository.repo:
    baseurl: https://www.example.com/
    enabled: true
    gpgcheck: true
    gpgkey: file:///etc/pki/rpm-gpg/RPM-GPG-KEY-EXAMPLE
    name: Example Repository
```

 Note

Una de las ventajas de usar `cloud-init` es que puede añadir una sección `packages:` al archivo de configuración. En esta sección, puede incluir los nombres de los paquetes que desee instalar. Puede instalar los paquetes desde el repositorio predeterminado o desde el repositorio nuevo que agregó al archivo `cloud-config`.

Para obtener información más específica sobre la estructura del archivo YAML, consulte [Adding a YUM repository](#) en la documentación `cloud-init`.

Después de configurar el archivo de formato YAML, puede ejecutarlo en el marco `cloud-init` de la AWS CLI. Asegúrese de incluir la opción `--userdata` y el nombre del archivo `.yaml` para llamar las operaciones que desee.

```
$ aws ec2 run-instances \
  --image-id \
```

```
resolve:ssm:/aws/service/ami-amazon-linux-latest/al2023-ami-kernel-default-x86_64 \  
--instance-type m5.xlarge \  
--region us-east-1 \  
--key-name aws-key-us-east-1 \  
--security-group-ids sg-004a7650 \  
--user-data file://cloud-config.yml
```

Aplicación de parches activos del kernel en AL2023

Puede usar Kernel Live Patching para AL2023 para aplicar revisiones de vulnerabilidad de seguridad y de errores críticos específicos a un código kernel de Linux en ejecución, sin reinicios ni interrupciones en las aplicaciones en ejecución. Además, Kernel Live Patching permite mejorar la disponibilidad de aplicaciones, a la vez que aplica estas correcciones hasta que el sistema pueda reiniciarse.

AWS lanza dos tipos de parches activos del kernel para AL2023:

- Actualizaciones de seguridad: incluye actualizaciones para vulnerabilidades y exposiciones comunes de Linux (CVE). Normalmente, estas actualizaciones se califican como importantes o críticas mediante las clasificaciones de Amazon Linux Security Advisory. Por lo general, se asignan a una puntuación del sistema de clasificación de vulnerabilidades comunes (CVSS) de 7 o superior. En algunos casos, AWS puede proporcionar actualizaciones antes de asignar un CVE. En estos casos, los parches pueden aparecer como correcciones de errores.
- Correcciones de errores: incluye correcciones de errores críticos y problemas de estabilidad que no están asociados con CVE.

AWS proporciona parches en tiempo real del kernel para una versión del kernel de AL2023 durante un máximo de 3 meses después de su lanzamiento. Después de este período, debe actualizar a una versión posterior del kernel para continuar recibiendo parches activos del kernel.

Los parches activos del kernel de AL2023 están disponibles como paquetes RPM firmados en los repositorios AL2023 existentes. Los parches se pueden instalar en instancias individuales mediante flujos de trabajo del administrador de paquetes DNF existentes. O también, se pueden instalar en un grupo de instancias administradas mediante AWS Systems Manager.

Kernel Live Patching para AL2023 se proporciona sin costo adicional.

Temas

- [Limitaciones](#)
- [Configuraciones admitidas y requisitos previos](#)
- [Usar Kernel Live Patching](#)

Limitaciones

Al aplicar un parche activo del kernel, no se puede realizar la hibernación, utilizar herramientas avanzadas de depuración (como SystemTap, kprobes y herramientas basadas en eBPF) ni acceder a los archivos de salida de `fttrace` utilizados por la infraestructura de Kernel Live Patching.

Note

Debido a limitaciones técnicas, algunos problemas no se pueden solucionar con los parches activos. Por este motivo, estas correcciones no se incluirán en el paquete de parches activos del kernel, sino únicamente en la actualización del paquete nativo del kernel. Puede instalar el paquete nativo del kernel y [actualizar y reiniciar](#) el sistema para activar los parches como de costumbre.

Configuraciones admitidas y requisitos previos

Kernel Live Patching es compatible con instancias de Amazon EC2 y máquinas virtuales en las instalaciones que ejecutan AL2023.

Para usar Kernel Live Patching en AL2023, debe utilizar:

- Una arquitectura `x86_64` o `ARM64` de 64 bits
- Versión `6.1` o `6.12` del kernel

Requisitos de política

Para descargar paquetes de los repositorios de AL2023, Amazon EC2 necesita acceso a los buckets de Amazon S3 que son propiedad del servicio. Si utiliza un punto de conexión de Amazon Virtual Private Cloud (VPC) para Amazon S3 en su entorno, asegúrese de que su política de punto de conexión de VPC permita el acceso a esos buckets públicos. En la tabla, se describe cada uno de los buckets de Amazon S3 a los que Amazon EC2 puede necesitar acceder para Kernel Live Patching.

ARN del bucket de S3	Descripción
arn:aws:s3:::al2023-repos- <i>region</i> -de612dc2/*	Bucket de Amazon S3 que contiene repositorios de AL2023

Usar Kernel Live Patching

Puede habilitar y utilizar Kernel Live Patching en instancias individuales mediante la línea de comandos de la propia instancia. Como alternativa, puede habilitar y utilizar Kernel Live Patching en un grupo de instancias administradas mediante AWS Systems Manager.

En las siguientes secciones se explica cómo habilitar y usar Kernel Live Patching en instancias individuales mediante la línea de comandos.

Para obtener más información sobre cómo habilitar y utilizar Kernel Live Patching en un grupo de instancias administradas, consulte [Uso de Kernel Live Patching en instancias de AL2023](#) en la Guía del usuario de AWS Systems Manager.

Temas

- [Habilitar Kernel Live Patching](#)
- [Visualizar los parches activos disponibles del kernel](#)
- [Aplicar parches activos del kernel.](#)
- [Ver los parches activos del kernel aplicados](#)
- [Deshabilitar Kernel Live Patching](#)

Habilitar Kernel Live Patching

Kernel Live Patching está deshabilitado de forma predeterminada en AL2023. Para usar parches activos, debe instalar el complemento DNF de Kernel Live Patching y habilitar la funcionalidad de parches activos.

Para habilitar Kernel Live Patching

1. Los parches activos del kernel están disponibles para AL2023 con la versión del kernel 6.1 o posterior. Para verificar la versión del kernel, ejecute el siguiente comando.

```
$ sudo dnf list kernel
```

2. Instalar el complemento DNF de Kernel Live Patching.

```
$ sudo dnf install -y kpatch-dnf
```

3. Habilitar el complemento DNF de Kernel Live Patching.

```
$ sudo dnf kernel-livepatch -y auto
```

Este comando también instala la última versión del RPM del parche activo del kernel desde los repositorios configurados.

4. Para confirmar que el complemento DNF de Kernel Live Patching se ha instalado correctamente, ejecute el siguiente comando.

Cuando habilite Kernel Live Patching, se aplica automáticamente un RPM de parche activo del kernel vacío. Si Kernel Live Patching se habilitó correctamente, este comando devuelve una lista que incluye el RPM inicial vacío del parche activo del kernel (y otro RPM que configura el repositorio DNF que contiene los parches activos).

```
$ sudo rpm -qa | grep kernel-livepatch
kernel-livepatch-repo-s3-2023.7.20250428-0.amzn2023.noarch
kernel-livepatch-6.1.134-150.224-1.0-0.amzn2023.x86_64
```

5. Instale el paquete kpatch.

```
$ sudo dnf install -y kpatch-runtime
```

6. Actualice el servicio kpatch si se instaló previamente.

```
$ sudo dnf upgrade kpatch-runtime
```

7. Inicie el servicio kpatch. Este servicio carga todos los parches activos del kernel al inicializar o al arrancar.

```
$ sudo systemctl enable kpatch.service && sudo systemctl start kpatch.service
```

Visualizar los parches activos disponibles del kernel

Las alertas de seguridad de Amazon Linux se publican en el Centro de seguridad de Amazon Linux. Para obtener más información acerca de las alertas de seguridad de AL2023, que incluyen alertas para parches activos del kernel, consulte el [Centro de seguridad de Amazon Linux](#). Los parches activos del kernel tienen el prefijo ALASLIVEPATCH. Es posible que el Centro de seguridad de Amazon Linux no incluya revisiones activas del kernel que resuelven errores.

También puede descubrir los parches activos del kernel disponibles para avisos y CVE mediante la línea de comandos.

Para enumerar todos los parches activos del kernel disponibles para avisos.

Utilice el siguiente comando.

```
$ sudo dnf updateinfo list
Last metadata expiration check: 1:06:23 ago on Mon 13 Feb 2023 09:28:19 PM UTC.
ALAS2LIVEPATCH-2021-123    important/Sec. kernel-
livepatch-6.1.12-17.42-1.0-4.amzn2023.x86_64
ALAS2LIVEPATCH-2022-124    important/Sec. kernel-
livepatch-6.1.12-17.42-1.0-3.amzn2023.x86_64
```

Para enumerar todos los parches activos del kernel disponibles para los CVE.

Utilice el siguiente comando.

```
$ sudo dnf updateinfo list cves
Last metadata expiration check: 1:07:26 ago on Mon 13 Feb 2023 09:28:19 PM UTC.
CVE-2022-0123    important/Sec. kernel-livepatch-6.1.12-17.42-1.0-4.amzn2023.x86_64
CVE-2022-3210    important/Sec. kernel-livepatch-6.1.12-17.42-1.0-3.amzn2023.x86_64
```

Aplicar parches activos del kernel.

Aplicar parches activos del kernel con el administrador de paquetes DNF de la misma manera que aplicaría actualizaciones regulares. El complemento DNF para Kernel Live Patching administra los parches activos del kernel que se van a aplicar.

Tip

Le recomendamos que actualice el kernel regularmente con Kernel Live Patching para asegurarse de que recibe correcciones de seguridad específicas importantes y críticas

hasta que el sistema pueda reiniciarse. Compruebe también si hay disponibles correcciones adicionales para el paquete nativo del núcleo que no se puedan implementar como parches activos y, en esos casos, [actualice y reinicie](#) con la actualización del kernel.

Puede optar por aplicar un parche activo específico del kernel o aplicar cualquier parche activo del kernel disponible junto con las actualizaciones de seguridad regulares.

Para aplicar un parche activo del código kernel específico.

1. Obtenga la versión del parche activo del kernel con uno de los comandos descritos en [Visualizar los parches activos disponibles del kernel](#).
2. Aplique el parche activo del kernel para su kernel de AL2023.

```
$ sudo dnf install kernel-livepatch-kernel_version-package_version.amzn2023.x86_64
```

Por ejemplo, el siguiente comando aplica un parche activo del kernel para la versión 6.1.12-17.42 del kernel de AL2023.

```
$ sudo dnf install kernel-livepatch-6.1.12-17.42-1.0-4.amzn2023.x86_64
```

Para aplicar los parches activos del kernel disponibles junto con las actualizaciones de seguridad regulares

Utilice el siguiente comando.

```
$ sudo dnf upgrade --security
```

Omita la opción `--security` para incluir correcciones de errores.

Important

- La versión del kernel no se actualiza después de aplicar parches activos del kernel. La versión solo se actualiza a la nueva versión después de reiniciar la instancia.
- El kernel de AL2023 recibe parches activos del kernel durante un período de 3 meses. Una vez transcurrido este período, no se lanzan nuevos parches activos del kernel para esa versión del kernel.

- Para continuar recibiendo parches activos del kernel después de 3 meses, debe reiniciar la instancia para pasar a la nueva versión del kernel. La instancia seguirá recibiendo parches activos del kernel durante los 3 meses siguientes a la actualización.
- Para verificar la ventana de soporte para la versión del kernel, ejecute el siguiente comando:

```
$ sudo dnf kernel-livepatch support
```

```
The current version of the Linux kernel you are running will no longer receive  
live patches after 2025-07-22.
```

Ver los parches activos del kernel aplicados

Para ver los parches activos del kernel aplicados

Utilice el siguiente comando.

```
$ sudo kpatch list
```

```
Loaded patch modules:
```

```
livepatch_CVE_2022_36946 [enabled]
```

```
Installed patch modules:
```

```
livepatch_CVE_2022_36946 (6.1.57-29.131.amzn2023.x86_64)
```

```
livepatch_CVE_2022_36946 (6.1.57-30.131.amzn2023.x86_64)
```

El comando devuelve una lista de los parches activos del kernel de actualización de seguridad cargados e instalados. A continuación, se muestra un ejemplo del resultado.

Note

Un único parche activo del kernel puede incluir e instalar varios parches activos.

Deshabilitar Kernel Live Patching

Si ya no necesita utilizar Kernel Live Patching, puede desactivarlo en cualquier momento.

- Deshabilitar el uso de livepatches:

1. Deshabilitar el complemento:

```
$ sudo dnf kernel-livepatch manual
```

2. Deshabilitar el servicio kpatch:

```
$ sudo systemctl disable --now kpatch.service
```

- Quitar completamente las herramientas livepatch:

1. Eliminar el complemento:

```
$ sudo dnf remove kpatch-dnf
```

2. Eliminar kpatch-runtime:

```
$ sudo dnf remove kpatch-runtime
```

3. Eliminar todo lo que esté instalado en livepatches:

```
$ sudo dnf remove kernel-livepatch\*
```

Actualización del núcleo de Linux en AL2023

Temas

- [Versiones del núcleo de Linux en AL2023](#)
- [AL2023 Actualización a una versión más reciente del núcleo](#)
- [AL2023 kernels: preguntas frecuentes](#)

Versiones del núcleo de Linux en AL2023

AL2023 incluye regularmente nuevas versiones del núcleo basadas en las versiones de Long-Term Support (LTS) del núcleo de Linux.

AL2023 se lanzó originalmente en marzo de 2023 con el kernel 6.1.

En abril de 2025, AL2023 se agregó soporte para el kernel 6.12 de Linux. Este núcleo agregó nuevas funciones, como la programación del EEVDF, el I/O soporte de transferencia de FUSE, una nueva

API de Futex y mejoras en el eBPF. El kernel 6.12 también permite que un programa del espacio de usuario se proteja en tiempo de ejecución mediante el uso de columnas ocultas del espacio de usuario y el sellado de memoria.

En marzo de 2026, se AL2023 agregó soporte para el kernel 6.18 de Linux. El kernel 6.18 actualizado aporta mejoras adicionales en cuanto a la compatibilidad del procesador, la virtualización, la seguridad y el rendimiento. Entre sus características más destacables se incluyen las capacidades de IOMMU mejoradas en todas las arquitecturas y los controles vectoriales de ataque para gestionar las mitigaciones de vulnerabilidades de la CPU. Las mejoras en el rendimiento se obtienen mediante la optimización de la criptografía, con operaciones FSCRYPT más rápidas, las mejoras en la administración de la memoria y la introducción de Sheaves como una nueva capa de almacenamiento en caché opcional basada en matrices de CPU.

AL2023 Actualización a una versión más reciente del núcleo

A partir de junio de 2026, AL2023 actualizará el núcleo predeterminado una vez al año. El [al2023-ami-kernel-default](#) conjunto de AMIs actualizará a la versión más reciente del núcleo LTS, de modo que las instancias recién lanzadas incorporarán automáticamente la nueva versión del núcleo. Esta es la forma más sencilla de mantenerse al día con las últimas correcciones de seguridad y mejoras de rendimiento.

Si prefiere elegir una versión de núcleo específica, puede ejecutar AL2023 con el núcleo 6.12 o 6.18 seleccionando una AMI con el núcleo deseado preinstalado o actualizando una instancia EC2 de AL2023 existente.

Ejecución de una AL2023 AMI con una versión específica del núcleo

Puede optar por ejecutar una AL2023 AMI con un núcleo específico preinstalado a través de la consola de AWS o consultando SSM para obtener parámetros específicos. Las claves SSM que se van a consultar empiezan por `/aws/service/ami-amazon-linux-latest/`, seguido de una de estas opciones:

Para el kernel 6.12

- `al2023-ami-kernel-6.12-arm64` para la arquitectura arm64
- `al2023-ami-minimal-kernel-6.12-arm64` para la arquitectura arm64 (AMI mínima)
- `al2023-ami-kernel-6.12-x86_64` para la arquitectura x86_64
- `al2023-ami-minimal-kernel-6.12-x86_64` para la arquitectura x86_64 (AMI mínima)

Para el kernel 6.18

- `al2023-ami-kernel-6.18-arm64` para la arquitectura `arm64`
- `al2023-ami-minimal-kernel-6.18-arm64` para la arquitectura `arm64` (AMI mínima)
- `al2023-ami-kernel-6.18-x86_64` para la arquitectura `x86_64`
- `al2023-ami-minimal-kernel-6.18-x86_64` para la arquitectura `x86_64` (AMI mínima)

Consulte [Inicio AL2023 mediante el parámetro SSM y AWS CLI](#) para obtener más información sobre la selección AL2023 AMIs.

Actualización de una AL2023 instancia a un núcleo más reciente

Puedes actualizar in situ una AL2023 instancia en ejecución al núcleo 6.12 o 6.18 siguiendo estos pasos:

1. Detecta el núcleo actual y establece la versión de destino:

```
# Automatically detect current kernel version BEFORE upgrade
$ CURRENT_KERNEL=$(uname -r)
$ SOURCE_VERSION=""

$ if [[ $CURRENT_KERNEL == *"6.12"* ]]; then
    SOURCE_VERSION="6.12"
else
    SOURCE_VERSION=""
fi

# Save the source version to a persistent location for use after reboot
$ echo "${SOURCE_VERSION}" | sudo tee /var/lib/source_kernel_version > /dev/null
```

```
# Set your target version (change this to your desired kernel: 6.12 or 6.18)
$ TARGET_VERSION="6.12"
```

```
$ echo "Current kernel: ${SOURCE_VERSION:-6.1}"
$ echo "Upgrading to kernel ${TARGET_VERSION}"
```

2. Instale el paquete del núcleo de destino:

```
$ sudo dnf install -y kernel${TARGET_VERSION}
```

3. Obtenga la última versión del paquete del núcleo de destino:

```
$ version=$(rpm -q --qf '%{version}-%{release}.%{arch}\n' kernel${TARGET_VERSION} |
sort -V | tail -1)
```

4. Convierta el nuevo núcleo en su núcleo predeterminado:

```
$ sudo grubby --set-default "/boot/vmlinuz-$version"
```

5. Reinicie su sistema:

```
$ sudo reboot
```

6. Desinstale el núcleo anterior:

```
# Read the source kernel version from the saved file
$ SOURCE_VERSION=$(sudo cat /var/lib/source_kernel_version)

# Uninstall the source kernel
$ sudo dnf remove -y kernel${SOURCE_VERSION}
```

7. Sustituya los paquetes de núcleo adicionales por sus equivalentes de núcleo de destino:

```
# Set your target version (change this to your desired kernel: 6.12 or 6.18)
$ TARGET_VERSION="6.12"
```

```
$ declare -A pkgs
$ pkgs=(
[bpftool${SOURCE_VERSION}]=bpftool${TARGET_VERSION}
[kernel${SOURCE_VERSION}-debuginfo]=kernel${TARGET_VERSION}-debuginfo
[kernel${SOURCE_VERSION}-debuginfo-common]=kernel${TARGET_VERSION}-debuginfo-common
[kernel${SOURCE_VERSION}-headers]=kernel${TARGET_VERSION}-headers
[kernel${SOURCE_VERSION}-libbpf]=kernel${TARGET_VERSION}-libbpf
[kernel${SOURCE_VERSION}-libbpf-devel]=kernel${TARGET_VERSION}-libbpf-devel
[kernel${SOURCE_VERSION}-libbpf-static]=kernel${TARGET_VERSION}-libbpf-static
[kernel${SOURCE_VERSION}-modules-extra-common]=kernel${TARGET_VERSION}-modules-
extra-common
[kernel${SOURCE_VERSION}-tools]=kernel${TARGET_VERSION}-tools
[kernel${SOURCE_VERSION}-tools-devel]=kernel${TARGET_VERSION}-tools-devel
[perf${SOURCE_VERSION}]=perf${TARGET_VERSION}
[python3-perf${SOURCE_VERSION}]=python3-perf${TARGET_VERSION}
)
```

```
$ for pkg in "${!pkgs[@]"; do
  rpm -q $pkg && sudo dnf -y swap $pkg "${pkgs["$pkg"]}" ;
done
```

8. (Opcional) Desinstale kernel-devel para la versión anterior del núcleo:

```
$ rpm -q kernel${SOURCE_VERSION}-devel && sudo dnf remove -y kernel
${SOURCE_VERSION}-devel
```

Bajar a una versión anterior del núcleo

Si en algún momento necesita volver a una versión anterior del núcleo, siga los siguientes pasos:

1. Detecte el núcleo actual y establezca la versión de destino:

```
# Automatically detect current kernel version BEFORE downgrade
$ CURRENT_KERNEL=$(uname -r)
$ SOURCE_VERSION=""

$ if [[ $CURRENT_KERNEL == *"6.12"* ]]; then
  SOURCE_VERSION="6.12"
elif [[ $CURRENT_KERNEL == *"6.18"* ]]; then
  SOURCE_VERSION="6.18"
fi

# Save the source version to a persistent location for use after reboot
$ echo "${SOURCE_VERSION}" | sudo tee /var/lib/source_kernel_version > /dev/null
```

```
# Set your target version (change this to your desired kernel)
# Use "" for kernel 6.1, "6.12" for kernel 6.12
$ TARGET_VERSION=""
```

```
$ echo "Downgrading from kernel ${SOURCE_VERSION:-6.1} to kernel
${TARGET_VERSION:-6.1}"
```

2. Sustituya los paquetes adicionales del núcleo por sus equivalentes del núcleo de destino:

```
$ declare -A pkgs
$ pkgs=(
```

```

[bpftool${TARGET_VERSION}]=bpftool${SOURCE_VERSION}
[kernel${TARGET_VERSION}-debuginfo]=kernel${SOURCE_VERSION}-debuginfo
[kernel${TARGET_VERSION}-debuginfo-common]=kernel${SOURCE_VERSION}-debuginfo-common
[kernel${TARGET_VERSION}-headers]=kernel${SOURCE_VERSION}-headers
[kernel${TARGET_VERSION}-libbpf]=kernel${SOURCE_VERSION}-libbpf
[kernel${TARGET_VERSION}-libbpf-devel]=kernel${SOURCE_VERSION}-libbpf-devel
[kernel${TARGET_VERSION}-libbpf-static]=kernel${SOURCE_VERSION}-libbpf-static
[kernel${TARGET_VERSION}-modules-extra-common]=kernel${SOURCE_VERSION}-modules-extra-common
[kernel${TARGET_VERSION}-tools]=kernel${SOURCE_VERSION}-tools
[kernel${TARGET_VERSION}-tools-devel]=kernel${SOURCE_VERSION}-tools-devel
[perf${TARGET_VERSION}]=perf${SOURCE_VERSION}
[python3-perf${TARGET_VERSION}]=python3-perf${SOURCE_VERSION}
)
$ for pkg in "${!pkgs[@]}"; do
    rpm -q "${pkgs["$pkg"]}" && sudo dnf -y swap "${pkgs["$pkg"]}" $pkg ;
done

```

3. Instale el paquete del núcleo de destino:

```
$ sudo dnf install -y kernel${TARGET_VERSION}
```

4. Obtenga la última versión del paquete del núcleo de destino:

```
$ version=$(rpm -q --qf '%{version}-%{release}.%{arch}\n' kernel${TARGET_VERSION} |
sort -V | tail -1)
```

5. Convierta el núcleo de destino en el núcleo predeterminado:

```
$ sudo grubby --set-default "/boot/vmlinuz-$version"
```

6. Reinicie su sistema:

```
$ sudo reboot
```

7. Desinstale el núcleo de origen:

```

# Read the source kernel version from the saved file
$ SOURCE_VERSION=$(sudo cat /var/lib/source_kernel_version)

# Uninstall the source kernel

```

```
$ sudo dnf remove -y kernel${SOURCE_VERSION}
```

AL2023 kernels: preguntas frecuentes

1. ¿Debo reiniciar después de una actualización del kernel?

Cada cambio en el kernel en ejecución requiere un reinicio.

2. ¿Cómo puedo conservar los núcleos up-to-date en varias instancias?

Amazon Linux no ofrece funciones para administrar flotas de instancias. Le recomendamos que parchee grandes flotas con herramientas como [AWS Systems Manager](#).

3. ¿Cómo compruebo qué versión del kernel estoy ejecutando en este momento?

Ejecuta este comando en tu AL2023 instancia:

```
$ uname -r
```

4. ¿Qué núcleo me AL2023 recomienda usar?

Se recomienda actualizar a la última versión AL2023 del núcleo 6.18 mientras todos los demás AL2023 núcleos sigan siendo compatibles. Se recomienda a los clientes que prueben sus cargas de trabajo antes de realizar la actualización.

5. ¿Funcionarán mis aplicaciones actuales con cualquier AL2023 núcleo?

AL2023 admite un núcleo más reciente (6.12 o 6.18) de la misma manera que el kernel 6.1. Las aplicaciones funcionarán y las mejoras se están produciendo de forma clandestina. En cualquier caso, los clientes deberían probar sus cargas de trabajo específicas antes de cambiar a un núcleo más nuevo.

6. ¿Cómo instalo los encabezados del núcleo, los paquetes de desarrollo y los módulos adicionales para el núcleo 6.12 o 6.18?

Por favor, ejecute:

```
$ version=$(uname -r | grep -oP '^d+\\.d+')
```

```
$ sudo dnf install -y kernel${version}-modules-extra-$(uname -r) kernel${version}-headers-$(uname -r) kernel${version}-devel-$(uname -r)
```

7. ¿Durante cuánto tiempo se admitirá la compatibilidad con los núcleos 6.12 y 6.18?

Los núcleos 6.12 y 6.18 se admitirán hasta el final previsto de la vida útil de Amazon Linux en 2023, es decir, el 30 de junio de 2029.

Cómo empezar a programar tiempos de ejecución en AL2023

AL2023 proporciona diferentes versiones de los tiempos de ejecución de algunos idiomas. Trabajamos con proyectos originales que admiten varias versiones al mismo tiempo. Obtenga información sobre cómo instalar y administrar estos paquetes con versiones nominales mediante el comando `dnf` para buscar e instalar estos paquetes.

En los siguientes temas se describe cómo existe cada ecosistema lingüístico. AL2023

Temas

- [C, C++ y Fortran en AL2023](#)
- [.NET en AL2023](#)
- [Go en AL2023](#)
- [Java en AL2023](#)
- [Node.js en AL2023](#)
- [Perl en AL2023](#)
- [PHP en AL2023](#)
- [Python en AL2023](#)
- [Ruby en AL2023](#)
- [Rust en AL2023](#)
- [Swift en AL2023](#)
- [TypeScript en AL2023](#)

C, C++ y Fortran en AL2023

AL2023 incluye tanto la colección de compiladores de GNU (GCC) como la interfaz Clang de LLVM (máquina virtual de bajo nivel).

La versión principal de GCC permanecerá constante durante toda la vida útil de AL2023. Las versiones menores incluyen correcciones de errores y podrían incluirse en las versiones de AL2023. Es posible que otras correcciones de errores, rendimiento y seguridad estén incorporadas a la versión principal de GCC que se incluye en AL2023.

AL2023 incluye la versión 11 de GCC como compilador predeterminado con las interfaces C (gcc), C++ (g++) y Fortran (gfortran). Además, AL2023 proporciona la versión 14 de GCC como un compilador alternativo opcional que se puede instalar junto con la versión predeterminada.

AL2023 no habilita las interfaces Ada (gnat), Go (gcc-go), Objective-C ni Objective-C++.

Los indicadores de compilación predeterminados con los que se crean los RPM de AL2023 incluyen indicadores de optimización y endurecimiento. Para crear su propio código con GCC, le recomendamos que incluya indicadores de optimización y endurecimiento.

Note

Cuando se invoca `gcc --version`, se muestra una cadena de versión como la siguiente `gcc (GCC) 11.3.1 20221121 (Red Hat 11.3.1-4)`. Red Hat hace referencia a la [ramificación de proveedores de GCC](#) en la que se basa el paquete GCC de Amazon Linux. Según la URL del informe de errores que aparece en `gcc --help`, todos los informes de errores y las solicitudes de soporte deben dirigirse a Amazon Linux. Para obtener más información sobre algunos de los cambios a largo plazo en esta rama de proveedores, como la macro de `__GNUCC_RH_RELEASE__`, consulte las fuentes de los paquetes de [Fedora](#).

Para obtener más información sobre la cadena de herramientas principal, consulte [Paquetes principales de cadenas de herramientas glibc, gcc, binutils](#).

Para obtener información sobre AL2023 y su relación con otras distribuciones de Linux, consulte [Relación con Fedora](#).

Para obtener información adicional sobre el cambio del triplete del compilador en AL2023 en comparación con AL2, consulte [Triplete del compilador](#).

Temas

- [GCC 14](#)
- [Comparación de versiones de lenguajes estándar](#)

GCC 14

AL2023 proporciona GCC 14 como compilador opcional que se puede instalar junto con GCC 11 predeterminado. GCC 14 incluye las últimas funciones y optimizaciones del lenguaje, lo que lo hace

adecuado para proyectos que requieren compatibilidad con los estándares C, C++ o Fortran más recientes.

Para instalar GCC 14, utilice el siguiente comando:

```
sudo dnf install gcc14 gcc14-c++ gcc14-gfortran
```

Los compiladores de GCC 14 se instalan con nombres de comandos específicos de cada versión para evitar conflictos con GCC 11 predeterminado:

- `gcc14-gcc`: compilador de C
- `gcc14-g++`: compilador de C++
- `gcc14-gfortran`: compilador de Fortran

Ejemplo de uso:

```
gcc14-gcc -o myprogram myprogram.c
gcc14-g++ -o mycppprogram mycppprogram.cpp
gcc14-gfortran -o myfortranprogram myfortranprogram.f90
```

Puede comprobar la versión instalada ejecutando:

```
gcc14-gcc --version
```

Esto mostrará información de la versión similar a la siguiente: `gcc14-gcc (GCC) 14.2.1 20250110 (Red Hat 14.2.1-7)`

Note

Tanto GCC 11 como GCC 14 se pueden instalar simultáneamente en el mismo sistema. Los comandos `gcc`, `g++` y `gfortran` predeterminados seguirán utilizando GCC 11, mientras que el acceso a GCC 14 se realizará a través de los comandos específicos de la versión.

Comparación de versiones de lenguajes estándar

En la siguiente tabla se comparan las versiones estándar de lenguajes predeterminados entre las distintas versiones de Amazon Linux y versiones del compilador de GCC:

Versión de Amazon Linux	Estándar C (predeterminado)	Estándar C++ (predeterminado)	Estándar de Fortran
AL2 con GCC 7 (predeterminado)	C11 (201112L)	C++14 (201402L)	Fortran 2008
AL2 con GCC 10 (opcional)	C17/C18 (201710L)	C++14 (201402L)	Fortran 2008
AL2023 con GCC 11 (predeterminado)	C17/C18 (201710L)	C++17 (201703L)	Fortran 2008
AL2023 con GCC 14 (opcional)	C17/C18 (201710L)	C++17 (201703L)	Fortran 2008

Mejoras clave por versión de GCC:

- GCC 10 frente a GCC 7: se actualizó el estándar C predeterminado de C11 a C17/C18, se agregó compatibilidad con las funciones de C++20 y se mejoraron las capacidades de optimización.
- GCC 11 frente a GCC 10: se actualizó el estándar predeterminado de C++ de C++14 a C++17, se mejoró la compatibilidad con C++20 y se agregaron características experimentales de C++23.
- GCC 14 frente a GCC 11: se agregó compatibilidad total con el estándar C23, se mejoraron las características de C++23, se mejoró la optimización y se mejoró el cumplimiento de las normas.

Estándares de lenguajes compatibles:

- Estándares C: todas las versiones son compatibles con C90, C99, C11 y C17/C18. GCC 10+ es compatible con C2x (borrador de la C23), mientras que GCC 14 es totalmente compatible con la versión C23.
- Estándares C++: todas las versiones son compatibles con C++98, C++03, C++11, C++14, C++17 y C++20. GCC 11+ ofrece compatibilidad experimental con C++23 y GCC 14 ofrece características mejoradas de C++23.
- Estándares de Fortran: todas las versiones son compatibles principalmente con Fortran 2008, con diferentes niveles de funciones de Fortran 2018 según la versión GCC.

Note

Si bien los estándares predeterminados se mantienen consistentes entre la versión GCC 11 y la 14, GCC 14 proporciona una compatibilidad significativamente mejorada con las características del lenguaje, una mejor optimización, un diagnóstico mejorado y una implementación más completa de los estándares más nuevos cuando se solicita explícitamente mediante marcas `-std=`.

.NET en AL2023

AL2023 actualmente proporciona el [.NET](#) motor de ejecución y el SDK, en las versiones 8.0, 9.0 y 10.0. .NET La versión 6.0 llegó end-of-life en noviembre de 2024 y ya no es compatible. Cada versión es compatible durante el mismo período de tiempo que la versión anterior. .NET Para obtener más información, consulte las [declaraciones de soporte de paquetes](#).

Migración a versiones más recientes .NET

La .NET comunidad upstream proporciona documentación de migración para la migración:

- [a .NET 10](#)
- [a .NET 9](#)
- [a .NET 8](#)

Instalando .NET en AL2023

Se pueden instalar varias versiones del .NET motor de ejecución simultáneamente AL2023. Sin embargo, los `dotnet-host` paquetes `dotnet` y se comparten en todas las versiones del SDK, por lo que solo una versión del SDK puede ser propietaria del punto de entrada de la `/usr/bin/dotnet` CLI a la vez. Si se instala una versión diferente del SDK, el `dotnet` comando compartido se actualizará a esa versión.

Para instalar una versión específica del .NET SDK, usa el `dnf` comando:

```
sudo dnf install dotnet-sdk-10.0
```

10.0Sustitúyala por la versión deseada (8.0, 9.0 o 10.0).

Para instalar solo el motor de ejecución sin el SDK:

```
sudo dnf install dotnet-runtime-9.0
```

Go en AL2023

Es posible que desee crear su propio código escrito [Go](#) en Amazon Linux y utilizar una cadena de herramientas incluida. AL2023 Similar a AL2, AL2023 actualizará la Go cadena de herramientas a lo largo de la vida útil del sistema operativo. Esto puede ser en respuesta a cualquier CVE de la cadena de herramientas que enviamos o como parte de una publicación trimestral.

Go es un lenguaje que se mueve relativamente rápido. Puede darse una situación en la que las aplicaciones existentes escritas en Go tengan que adaptarse a las nuevas versiones de la cadena de herramientas de Go. Para obtener más información sobre Go, consulte [Go 1 y el futuro de los programas Go](#).

Aunque AL2023 incorporará nuevas versiones de la Go cadena de herramientas a lo largo de su vida útil, no estará en sintonía con las versiones anteriores. Go Por lo tanto, el uso de la Go cadena de herramientas que se proporciona en el documento AL2023 puede no ser adecuado si se quiere crear Go código con las funciones más avanzadas del Go lenguaje y de la biblioteca estándar.

Durante su vida útil AL2023, las versiones anteriores de los paquetes no se eliminan de los repositorios. Si se necesita una cadena de herramientas de Go anterior, los clientes pueden optar por no corregir los errores y las correcciones de seguridad de las cadenas de herramientas de Go más recientes, e instalar una versión anterior desde los repositorios con los mismos mecanismos disponibles para cualquier RPM.

Si quieres crear tu propio Go código AL2023, puedes utilizar la cadena de Go herramientas incluida AL2023 con la seguridad de que esta cadena de herramientas podría avanzar a lo largo de su vida útil. AL2023

AL2023 Funciones Lambda escritas en Go

Como Go compila en código nativo, Lambda trata a Go como un tiempo de ejecución personalizado. Puede usar el `provided.al2023` tiempo de ejecución para implementar Go funciones en AL2023 Lambda.

Para obtener más información, consulte [Creación de funciones de Lambda con Go](#) en la Guía para desarrolladores de AWS Lambda .

Java en AL2023

AL2023 ofrece varias versiones de [Amazon Corretto](#) para admitir cargas de trabajo basadas en Java. Todos los paquetes basados en Java incluidos en AL2023 están diseñados con Amazon Corretto 17.

Corretto es una versión del Open Java Development Kit (OpenJDK) con el apoyo de Amazon a largo plazo. Corretto está certificado mediante el kit de compatibilidad técnica de Java (TCK) para garantizar que cumple con el estándar Java SE y está disponible en Linux, Windows y macOS.

Hay un paquete [Amazon Corretto](#) disponible para cada uno de los Corretto 1.8.0, Corretto 11 y Corretto 17.

Cada versión de Corretto en AL2023 es compatible durante el mismo período de tiempo que la versión de Corretto, o hasta el fin del servicio de AL2023, lo que ocurra primero. Para obtener más información, consulte las [declaraciones de soporte del paquete Amazon Linux](#) y las [preguntas frecuentes de Amazon Corretto](#).

Node.js en AL2 023

[Node.js](#) en AL2 023 se representa mediante las versiones 20, 22 y 24. Amazon Linux sigue el [programa de soporte](#) previo y siempre se puede comprobar el estado del soporte de cualquier versión de Node.js en la [página de estado del soporte de Package](#). Todas las versiones compatibles de Node.js tienen espacios de nombres y se pueden instalar en el mismo sistema simultáneamente. El espacio de nombres garantiza que cada instalación de Node.js sea única dentro del sistema de archivos. Esto se consigue cambiando el nombre de los directorios y archivos clave en función de la versión en tiempo de ejecución. Los nombres reales de los ejecutables serán similares a `node-{MAJOR_VERSION}` o `npm-{MAJOR_VERSION}`. Solo puede haber una versión de Node.js activa a la vez. Esta versión activa proporciona los directorios y nombres de archivo predeterminados, como `node`, `npm` o `/usr/lib/node_modules`, y los dirige al entorno de ejecución actualmente activo.

Esto se logra utilizando las capacidades de la herramienta de alternativas. Es importante recordar que los nombres de los ejecutables predeterminados son virtuales y pueden cambiar en cualquier momento al apuntar a una versión de Node.js instalada diferente. Esta flexibilidad permite al software que utiliza `node` en el shebang seleccionar la versión deseada cuando se invoca. Sin embargo, cuando Node.js se requiere una versión específica de, la persistencia de la versión se puede lograr llamando al ejecutable con espacio de nombres (por ejemplo, `node-20` o `node-22`), que siempre usará la versión especificada del tiempo de ejecución. Además, los ejecutables con espacio de

nombres de la herramienta npm, como npm-20 o npm-22, siempre están asociados a la versión de Node.js correspondiente, independientemente del tiempo de ejecución activo actualmente.

Node.js se distribuye en varios paquetes con espacios de nombres que comienzan por "». `nodejs{MAJOR_VERSION}` Estos paquetes proporcionan node, una versión compatible de la herramienta npm, documentación, bibliotecas y mucho más. Por ejemplo, node y npm de los Node.js 22 los proporcionan los `nodejs22-npm` paquetes `nodejs22` y, respectivamente.

La herramienta de alternativas proporciona un único comando para cambiar de una versión de Node.js a otra. De forma predeterminada, la herramienta de alternativas está configurada para estar en modo automático, que usa prioridades para determinar la versión de Node.js actualmente activa. Sin embargo, puede activar cualquier versión instalada en cualquier momento. Actualmente, todas las versiones compatibles de Node.js tienen la misma prioridad, lo que significa que la primera versión instalada se activará automáticamente.

Algunos ejemplos útiles del uso de las alternativas

1. Compruebe para qué alternativas está configurada

```
alternatives --list
```

2. Compruebe la configuración actual del nodo

```
alternatives --display node
```

3. Cambie la versión de Node.js de forma interactiva

```
alternatives --config node
```

4. Cambie al modo manual y seleccione una versión específica

```
alternatives --set node /usr/bin/node-{MAJOR_VERSION}
```

5. Vuelva al modo de selección automática de versión

```
alternatives --auto node
```


Perl en AL2023

AL2023 proporciona la versión 5.32 del lenguaje de programación [Perl](#)

A pesar de que Perl ha ofrecido un alto grado de compatibilidad lingüística como parte de las versiones de Perl 5 en las últimas décadas, no se espera que Amazon Linux pase de Perl 5.32 en la versión de AL2023. Amazon Linux seguirá aplicando parches de seguridad a Perl durante toda la vigencia de AL2023, de acuerdo con nuestras [declaraciones de soporte de paquetes](#).

Módulos Perl en AL2023

En AL2023, varios módulos Perl están empaquetados como RPM. Si bien hay muchos módulos Perl disponibles como RPM, Amazon Linux no pretende empaquetar todos los módulos Perl posibles. Los módulos empaquetados como RPM pueden ser utilizados por otros paquetes RPM del sistema operativo, por lo que Amazon Linux dará prioridad a esos parches de seguridad frente a las actualizaciones de características puras.

AL2023 también incluye CPAN para que los desarrolladores de Perl puedan utilizar el administrador de paquetes idiomático para módulos Perl.

PHP en AL2023

AL2023 proporciona las versiones 8.1, 8.2, 8.3, 8.4 y 8.5 del lenguaje de [PHP](#) programación. Cada versión es compatible durante el mismo período de tiempo que el PHP inicial. Para obtener más información, consulte las [declaraciones de soporte de paquetes](#).

Migración desde versiones anteriores de PHP

La comunidad del PHP inicial recopiló una documentación de migración completa para pasar:

- [de PHP 8.4.x a 8.5.x PHP](#)
- [de PHP 8.3.x a PHP 8.4.x](#)
- [de PHP 8.2.x a PHP 8.3.x](#)
- [de PHP 8.1.x a PHP 8.2.x](#)
- [de PHP 8.0.x a PHP 8.1.x](#)

AL2 incluye PHP 8.0, 8.1 y 8.2 para `amazon-linux-extras` permitir una ruta de actualización sencilla a la versión 023. AL2

Migración desde las versiones PHP 7.x

Note

El proyecto [PHP](#) mantiene una lista y un cronograma de las [versiones compatibles](#), así como una lista de las [ramificaciones no compatibles](#).

Cuando se lanzó la AL2 023, todas las versiones 7.x y 5.x de la 023 no [PHP](#)contaban con el apoyo de la PHP comunidad y no estaban incluidas como opciones en la versión 023. AL2

La comunidad de PHP inicial recopiló una [documentación de migración completa para pasar a PHP 8.0 desde PHP 7.4](#). En combinación con la documentación a la que se hace referencia en la sección anterior sobre la migración a las versiones PHP 8.1 y PHP 8.2, puede elaborar todos los pasos necesarios para migrar su aplicación basada en PHP a la versión moderna de PHP.

Note

AL2 incluye PHP 7,1, 7,2, 7,3 y 7,4 pulgadas. `amazon-linux-extras` Es importante tener en cuenta que no se end-of-life garantiza que todos estos extras reciban más actualizaciones de seguridad.

PHPmódulos en AL2 023

AL2023 incluye muchos PHP módulos que se incluyen en PHP Core. AL2El objetivo de 023 no es incluir todos los paquetes de la [biblioteca comunitaria de PHP extensiones \(PECL\)](#).

Pythonen AL2023

AL2023 eliminó la Python versión 2.7 y todos los componentes Python necesarios ahora están escritos para funcionar con la Python versión 3.

AL2023 pone a disposición Python 3 `/usr/bin/python3` para mantener la compatibilidad con el código del cliente, así como con el código Python AL2023 incluido, que permanecerá como Python 3.9 durante toda su vida útil AL2023.

La versión de python a la que `/usr/bin/python3` apunta se considera el sistema Python y para AL2023 ello es la Python 3.9.

AL2023 actualmente proporciona versiones más recientes dePython, incluidas las versiones 3.11, 3.12, 3.13 y 3.14. Cada versión es compatible durante el mismo período de tiempo que la versión anterior. Python Para obtener más información, consulte las [declaraciones de soporte de paquetes](#).

Se Python pueden instalar varias versiones de simultáneamente. AL2023 Si bien `/usr/bin/python3` siempre será Python 3.9, cada versión de Python tiene un espacio de nombres y se puede encontrar por su número de versión. Por ejemplo, si `python3.11` está instalada, `/usr/bin/python3.11` existirá junto con `/usr/bin/python3.9` y el enlace simbólico `/usr/bin/python3` apunta a `/usr/bin/python3.9`.

Note

No cambies a qué apunta el `/usr/bin/python3` enlace simbólico, ya que esto podría interrumpir la funcionalidad principal de AL2023.

Migración a versiones más recientes Python

La Python comunidad upstream proporciona documentación de migración para la migración:

- [a Python 3.14](#)
- [a Python 3.13](#)
- [a Python 3.12](#)
- [a Python 3.11](#)

Pythonmódulos en AL2023

Varios Python módulos se empaquetan como RPMs en AL2023. Por lo general, RPMs Python los módulos se construirán pensando únicamente en la versión del sistema dePython.

Rubyen AL2023

[Ruby](#) en AL2023 está representado por las versiones 3.2 y 3.4. Amazon Linux sigue el programa de soporte previo y siempre se puede comprobar el estado del soporte de cualquier Ruby versión en el sitio [web de Ruby](#). Todas las versiones compatibles de Ruby tienen espacios de nombres y se pueden instalar en el mismo sistema simultáneamente. El espacio de nombres garantiza que

cada instalación de Ruby sea única dentro del sistema de archivos. Esto se consigue cambiando el nombre de los directorios y archivos clave en función de la versión en tiempo de ejecución. Los nombres reales de los ejecutables serán como `ruby {MAJOR.MINOR}` (p. ej., o). `ruby3.2` `ruby3.4` Ruby3.4 también proporciona el binario con espacio de nombres MRI (Matz's Ruby Interpreter) `ruby3.4-mri`, que hace referencia a la implementación de referencia estándar basada en C. Ruby Solo puede haber una versión de Ruby activa a la vez. Esta versión activa proporciona los directorios y nombres de archivo predeterminados, como `ruby`, `gem` o `bundle`, y los dirige al motor de ejecución actualmente activo.

Esto se logra utilizando las capacidades de la herramienta de alternativas. Es importante recordar que los nombres de los ejecutables predeterminados son virtuales y pueden cambiar en cualquier momento al apuntar a una versión de Ruby instalada diferente. Esta flexibilidad permite al software que utiliza `ruby` in the shebang seleccionar la versión deseada al invocarla. Sin embargo, cuando Ruby se requiere una versión específica de, la persistencia de la versión se puede lograr llamando al ejecutable con espacio de nombres (por ejemplo, `ruby3.2` o `ruby3.4`), que siempre utilizará la versión especificada del tiempo de ejecución. Además, los ejecutables con espacio de nombres de las herramientas gemelas y empaquetadoras, como, `oruby3.2-gem`, `ruby3.4-gem` `ruby3.2-bundler`, siempre están asociados a la Ruby versión correspondiente `ruby3.4-bundler`, independientemente del tiempo de ejecución activo en ese momento.

Ruby se distribuye en varios paquetes con espacios de nombres que comienzan por "». `ruby{MAJOR.MINOR}` Estos paquetes proporcionan versiones compatibles con Ruby de las herramientas Gem y Bundler, documentación, bibliotecas y mucho más. Por ejemplo, el `ruby3.2` paquete proporciona el tiempo de ejecución principal de Ruby 3.2, que incluye (proporciona la gema) y `ruby3.2-rubygems` `ruby3.2-rubygem-bundler` (proporciona el paquete y el paquete) como dependencias.

Tras instalar una Ruby versión, es posible que las entradas de las herramientas complementarias aparezcan como nulas en la configuración de las alternativas. Esto se puede comprobar ejecutando `alternatives --display ruby`. Si las entradas aparecen como nulas, deben registrarse manualmente mediante `alternatives --install`. Por ejemplo, para registrar todas las herramientas complementarias de la versión Ruby 3.4:

```
sudo alternatives --install /usr/bin/gem gem /usr/bin/ruby3.4-gem 34
sudo alternatives --install /usr/bin/bundle bundle /usr/bin/ruby3.4-bundle 34
sudo alternatives --install /usr/bin/bundler bundler /usr/bin/ruby3.4-bundler 34
sudo alternatives --install /usr/bin/erb erb /usr/bin/ruby3.4-erb 34
sudo alternatives --install /usr/bin/racc racc /usr/bin/ruby3.4-racc 34
sudo alternatives --install /usr/bin/rdoc rdoc /usr/bin/ruby3.4-rdoc 34
```

```
sudo alternatives --install /usr/bin/ri ri /usr/bin/ruby3.4-ri 34
```

El valor de prioridad (por ejemplo, 34 para Ruby 3.4, 32 para Ruby 3.2) debe coincidir con la prioridad utilizada en la entrada principal alternativa de Ruby. Una vez registradas, las herramientas complementarias se administrarán automáticamente junto con la alternativa a Ruby.

La herramienta de alternativas proporciona un único comando para cambiar de una versión de Ruby a otra. De forma predeterminada, la herramienta de alternativas está configurada para estar en modo automático, que usa prioridades para determinar la versión de Ruby actualmente activa. Sin embargo, puede activar cualquier versión instalada en cualquier momento. Actualmente, todas las versiones compatibles de Ruby tienen la misma prioridad, lo que significa que la primera versión instalada se activará automáticamente.

Algunos ejemplos útiles del uso de las alternativas

1. Compruebe para qué alternativas está configurada

```
alternatives --list
```

2. Comprueba la configuración actual de ruby

```
alternatives --display ruby
```

3. Cambie la versión de Ruby de forma interactiva

```
alternatives --config ruby
```

4. Cambie al modo manual y seleccione una versión específica

```
alternatives --set ruby /usr/bin/ruby{MAJOR.MINOR}
```

5. Vuelva al modo de selección automática de versión

```
alternatives --auto ruby
```

Rusten AL2023

Es posible que desee compilar código escrito [Rusten](#) Amazon Linux y utilizar una cadena de herramientas incluida. AL2023

Similar a AL2, AL2023 actualizará la Rust cadena de herramientas a lo largo de la vida útil del sistema operativo. Esto puede ser en respuesta a cualquier CVE de la cadena de herramientas que enviamos o como parte de una publicación trimestral.

[Rust](#) es un lenguaje que evoluciona con relativa rapidez, con nuevos lanzamientos con una cadencia aproximada de seis semanas. Estas versiones pueden añadir un nuevo lenguaje o características de biblioteca estándar. Aunque AL2023 incorporará nuevas versiones de la Rust cadena de herramientas a lo largo de su vida útil, no estará en sintonía con las versiones anteriores. Rust Por lo tanto, el uso de la Rust cadena de herramientas que se proporciona en el documento AL2023 puede no ser adecuado si desea crear Rust código con las funciones más avanzadas del lenguaje. Rust

Durante su vida útil AL2023, las versiones antiguas de los paquetes no se eliminan de los repositorios. Si se necesita una cadena de herramientas de Rust más antigua, los clientes pueden optar por no corregir los errores y las correcciones de seguridad de las cadenas de herramientas de Rust más recientes, e instalar una versión anterior desde los repositorios con los mismos mecanismos disponibles para cualquier RPM.

Si quieres crear tu propio Rust código AL2023, puedes utilizar la cadena de Rust herramientas incluida AL2023 con la seguridad de que esta cadena de herramientas podría avanzar a lo largo de su vida útil. AL2023

AL2023 Funciones Lambda escritas en Rust

Como Rust compila en código nativo, Lambda trata a Rust como un tiempo de ejecución personalizado. Puede usar el `provided.al2023` tiempo de ejecución para implementar Rust funciones en AL2023 Lambda.

Para obtener más información, consulte [Creación de funciones de Lambda con Rust](#) en la Guía para desarrolladores de AWS Lambda .

Swift en AL2023

AL2023 actualmente proporciona el [Swift](#) tiempo de ejecución y el SDK.

En AL2023, el Swift SDK y el tiempo de ejecución se dividen en paquetes separados. El `swiftlang-lib` paquete contiene las bibliotecas de tiempo de ejecución y el `swiftlang` paquete contiene el SDK completo. Esto permite a los usuarios ejecutar binarios compilados con el Swift SDK en los hosts de producción sin necesidad de instalar el SDK completo. En este momento, hay diferentes versiones de que no Swift son compatibles con ABI, por lo que Swift, cuando se actualicen, las aplicaciones deberán reconstruirse con la nueva versión antes de poder implementarlas.

 Note

Por ejemplo Rust, Swift las versiones seguirán a la última versión estable del Swift proyecto. Los paquetes no tendrán espacios de nombres y solo se admitirá una versión a la vez.

Algunas notas sobre el uso de Swift AL2023

- Interoperabilidad con C++: es `clang` posible que el administrador de paquetes del sistema no funcione con Swift. Para resolver este problema, añade `swift-clang` a su configuración `PATH` o utilice alternativas para `clang` configurarlas `swift-clang`.
- Depuración del código Swift: el sistema LLDB no puede depurar Swift las fuentes. En su lugar, utilice el que se LLDB incluye con el Swift paquete añadiendo el directorio de Swift instalación al suyo `PATH` o utilizando alternativas para `lldb` configurarlo. `swift-lldb`

TypeScript en AL2023

 Note

Este documento proporciona la información esencial sobre TypeScript y su entorno de ejecución basado en Node.js. También cubre un flujo de trabajo de desarrollo típico y explica cómo se empaqueta TypeScript en AL2023 para ofrecer un entorno de desarrollo coherente y reproducible.

[TypeScript](#) (TS) es un lenguaje de programación basado en JavaScript (JS) que ofrece todas las funciones de JS y, además, [lo amplía con un sistema de tipos](#). En un escenario típico, los programas escritos en TS se traducen primero al código JS y luego se ejecutan como cualquier otro programa JS normal Node.js. En la terminología específica de TS, este proceso de traducción se denomina [“compilación” y lo realiza un “compilador”](#), llamado `tsc`. El propio compilador `tsc` está escrito en JS, por lo que, para ejecutarse, también necesita un entorno de ejecución de JS, como Node.js. A diferencia de otros entornos de ejecución de JS, Node.js actualmente solo tiene soporte TS experimental y ligero. La compatibilidad total con TS, incluida la verificación de tipos, aún requiere el uso de paquetes de terceros, como [TypeScript](#). La forma esperada de obtener `tsc` (el compilador de TS) para el entorno de ejecución de Node.js es instalar el módulo de nodo `Typescript`. Esto se puede hacer usando uno de los administradores de paquetes, normalmente `npm`. Hay dos formas

de instalar el compilador de TS usando npm: globalmente y en un proyecto. [El método recomendado oficialmente es instalar](#) el compilador de TS por proyecto, lo que garantiza la coherencia y la reproducibilidad de los proyectos a largo plazo. Sin embargo, instalar el compilador de TS de forma global puede seguir siendo útil, ya que proporciona la misma versión para todo el host y su entorno de ejecución de JS y, por lo tanto, para los proyectos que no tienen un compilador de TS instalado localmente. Así es precisamente como los paquetes RPM disponibles en Amazon Linux, como `nodejs20-typescript` o `nodejs22-typescript`, instalan un compilador de TS: globalmente en el nivel del sistema y por separado para cada versión de Node.js compatible.

El `tsc` no depende directamente de ninguna versión de Node.js. El compilador espera un cierto nivel de características de tiempo de ejecución, que se definen en un archivo especial (`tsconfig.json`) a través de opciones tales como [target](#) y [lib](#). Los valores de estas opciones representan una versión del estándar [ECMAScript](#) (ES), que puede (o no) ser compatible con el entorno de ejecución JS. Las distintas versiones de Node.js admiten diferentes versiones de ES. Cuanto más reciente sea la versión de Node.js, más alta y completa será la compatible con la versión estándar de ES. Si `tsconfig.json` no existe en el directorio raíz de un proyecto, se utilizará el conjunto predeterminado de opciones de configuración. La tabla de compatibilidad con las distintas versiones de Node.js y las funciones compatibles de las distintas versiones estándar de ES está disponible en [node.green](#). El `tsc` tiene más de 100 opciones diferentes, que se pueden definir en `tsconfig.json`. También se admite el encadenamiento de configuraciones cuando algunas opciones de configuración se definen en otro archivo y luego se incluyen en el archivo principal. Este enfoque permite instalar una [configuración TS base](#) compatible con una versión determinada de y, a continuación, ampliarla con opciones específicas del proyecto. Afortunadamente, las configuraciones TS base de Node.js están disponibles como módulos de nodo que se pueden instalar en una carpeta de proyecto mediante npm. Este es el código fuente de la configuración para las versiones de Node.js [18](#), [20](#) y [22](#).

El diseño del tiempo de ejecución de Node.js basado tiene una cierta debilidad: solo admite una versión del tiempo de ejecución en un host y requiere la reproducibilidad y la coherencia de todas las dependencias en el nivel del proyecto. Esto llevó al siguiente enfoque común de usar TypeScript: el compilador de TS, la configuración base de TS para la versión actual de Node.js y todas las dependencias de software se instalan localmente, dentro de un proyecto. Aunque se espera que los módulos de nodo instalados globalmente sean solo herramientas CLI, como npm y `tsc`, que también es una herramienta CLI, rara vez se instala globalmente. Afortunadamente, las instalaciones globales (en todo el sistema) y locales (dentro de un proyecto) de `tsc` pueden coexistir sin problemas y también pueden ser versiones diferentes que se utilizan de forma independiente. Tenga en cuenta que un `tsc` instalado localmente debe ejecutarse con la herramienta `npx`, que se instala junto con npm. Por lo tanto, incluso con un compilador TS de sistema, los usuarios tienen la oportunidad de

elegir versiones de los componentes del motor de ejecución, por ejemplo Node.js (cambiando la versión activa por alternativas), un compilador de TS (instalándolo local o globalmente y también cambiando la versión activa a través de alternativas) y configurándolo según las necesidades específicas.

Amazon Linux empaqueta un compilador de TS de la misma forma que otros módulos de nodos instalados globalmente, como npm, en una versión básica por versión de Node.js. Los paquetes y los binarios tienen espacios de nombres y contienen la versión principal de Node.js como parte de sus nombres. El nombre ejecutable predeterminado del compilador, tsc, lo administra en tiempo de ejecución la herramienta de alternativas y apunta a la versión actualmente activa de Node.js para la que se instaló y con la que se ejecutará. Esta selección no depende de la versión de tiempo de ejecución actual de Node.js. Es posible tener un ejecutable de nodo que apunte a Node.js 20 y que tsc esté configurado para ser interpretado por Node.js 22. También es posible utilizar los nombres con espacios de nombres de un compilador de TS, por ejemplo, tsc- {MAJOR_VERSION} independientemente de para qué esté configurado el nombre de tsc predeterminado.

Algunos comandos útiles para administrar la versión activa de un compilador de TS

1. Compruebe para qué alternativas está configurada

```
alternatives --list
```

2. Compruebe la configuración actual de tsc

```
alternatives --display tsc
```

3. Cambie la versión de tsc de forma interactiva

```
alternatives --config tsc
```

4. Cambie al modo manual y seleccione una versión específica

```
alternatives --set tsc /usr/bin/tsc-{MAJOR_VERSION}
```

5. Vuelva al modo de selección automática de versión

```
alternatives --auto tsc
```

Un ejemplo de instalación y uso de varias versiones de Node y un compilador de TS en el mismo sistema:

```
# Check the AL2023 release
$ cat /etc/amazon-linux-release
Amazon Linux release 2023.9.20250929 (Amazon Linux)

# Install a TypeScript compiler for Node.js 20 and 22
# Node.js 20 and 22 will be installed automatically
$ sudo dnf install -qy nodejs20-typescript nodejs22-typescript

# Check what was installed
$ rpm -q nodejs20 nodejs20-typescript nodejs22 nodejs22-typescript
nodejs20-20.19.5-1.amzn2023.0.1.x86_64
nodejs20-typescript-5.9.2-1.amzn2023.0.1.noarch
nodejs22-22.19.0-1.amzn2023.0.1.x86_64
nodejs22-typescript-5.9.2-1.amzn2023.0.1.noarch

# Check the active version of Node.js - it is version 20
$ alternatives --display node
node - status is auto.
  link currently points to /usr/bin/node-20
/usr/bin/node-20 - priority 100
  slave npmrc: /usr/lib/nodejs20/lib/node_modules/npm/npmrc
  slave npm: /usr/bin/npm-20
  slave npx: /usr/bin/npx-20
  slave node_modules: /usr/lib/nodejs20/lib/node_modules
/usr/bin/node-22 - priority 100
  slave npmrc: /usr/lib/nodejs22/lib/node_modules/npm/npmrc
  slave npm: /usr/bin/npm-22
  slave npx: /usr/bin/npx-22
  slave node_modules: /usr/lib/nodejs22/lib/node_modules
Current 'best' version is /usr/bin/node-20.

# Check the active JS runtime version for TypeScript
# Currently, the tsc compiler will be executed by Node.js 22
$ alternatives --display tsc
tsc - status is auto.
  link currently points to /usr/bin/tsc-22
/usr/bin/tsc-22 - priority 100
  slave tsserver: /usr/bin/tsserver-22
/usr/bin/tsc-20 - priority 100
```

```
slave tsserver: /usr/bin/tsserver-20
Current 'best' version is /usr/bin/tsc-22.

# Check versions printed by executables
$ node -v
v20.19.5

$ tsc -v
Version 5.9.2

# while the node is 20, tsc is executed by node 22 anyway
$ head -1 /usr/bin/tsc
#!/usr/bin/node-22

# However, instead of default executable names, e.g. node or tsc,
# we can use namespaced names to target any installed version
$ node-20 -v
v20.19.5

$ node-22 -v
v22.19.0

$ tsc-20 -v
Version 5.9.2

$ tsc-22 -v
Version 5.9.2

$ head -1 /usr/bin/tsc-20
#!/usr/bin/node-20

$ head -1 /usr/bin/tsc-22
#!/usr/bin/node-22
```

Usuarios y grupos reservados de AL2023

AL2023 asigna previamente determinados usuarios y grupos tanto durante el aprovisionamiento de la imagen como durante la instalación de determinados paquetes. Los usuarios, los grupos y sus UID y GID asociados se enumeran aquí para evitar conflictos.

Temas

- [Lista de usuarios reservados de AL2023](#)
- [Lista de grupos reservados de AL2023](#)

Lista de usuarios reservados de AL2023

Nombre de usuario	UID
root	0
bin	1
daemon	2
adm	3
lp	4
sync (sincronizar)	5
shutdown	6
detener	7
correo	8
operador	11
games	12
ftp	14

Nombre de usuario	UID
squid	23
named	25
postgres	26
mysql	27
nscd	28
nscd	28
rpcuser	29
rpc	32
mailnull	47
Apache	48
smmsp	51
Tomcat	53
ldap	55
tss	59
nsld	65
avahi	70
tcpdump	72
sshd	74
radvd	75
dbus	81

Nombre de usuario	UID
postfix	89
dovecot	97
stapusr	156
stapsys	157
stapdev	158
avahi-autoipd	170
pulse	171
rtkit	172
sanlock	179
systemd-network	192
systemd-resolve	193
uidd	961
stap-server	962
systemd-journal-remote	963
redis6	970
pesign	971
smtpq	972
smtpd	973
nginx	974
munge	975

Nombre de usuario	UID
memcached	976
sphinx	977
haproxy	978
flatpak	979
debuginfod	980
dovnull	981
dnsmasq	982
unbound	983
clamscan	984
clamilt	985
clamupdate	986
colord	987
ods	988
aws-kinesis-agent-user	989
saslauth	990
cwagent	991
polkitd	992
ec2-instance-connect	993
chrony	994
systemd-timesync	995

Nombre de usuario	UID
systemd-coredump	996
libstoragemgmt	997
systemd-oom	999
ec2-user	1 000
nobody	65534

Listado por nombre

Nombre de usuario	UID
adm	3
Apache	48
avahi	70
avahi-autoipd	170
aws-kinesis-agent-user	989
bin	1
chrony	994
clamilt	985
clamscan	984
clamupdate	986
colord	987
cwagent	991

Nombre de usuario	UID
daemon	2
dbus	81
debuginfod	980
dnsmasq	982
dovecot	97
dovnull	981
ec2-instance-connect	993
ec2-user	1 000
flatpak	979
ftp	14
games	12
detener	7
haproxy	978
ldap	55
libstoragemgmt	997
lp	4
correo	8
mailnull	47
memcached	976
munge	975

Nombre de usuario	UID
mysql	27
named	25
nginx	974
nobody	65534
nscd	28
nscd	28
nsld	65
ods	988
operador	11
pesign	971
polkitd	992
postfix	89
postgres	26
pulse	171
radvd	75
redis6	970
root	0
rpc	32
rpcuser	29
rtkit	172

Nombre de usuario	UID
sanlock	179
saslauth	990
shutdown	6
smmsp	51
smtpd	973
smtpq	972
sphinx	977
squid	23
sshd	74
stap-server	962
stapdev	158
stapsys	157
stapusr	156
sync (sincronizar)	5
systemd-coredump	996
systemd-journal-remote	963
systemd-network	192
systemd-oom	999
systemd-resolve	193
systemd-timesync	995

Nombre de usuario	UID
tcpdump	72
Tomcat	53
tss	59
unbound	983
uidd	961

Lista de grupos reservados de AL2023

Nombre del grupo	GID
root	0
bin	1
daemon	2
sys	3
adm	4
tty	5
disco	6
disco	6
lp	7
mem	8
kmem	9
wheel	10

Nombre del grupo	GID
cdrom	11
correo	12
correo	12
man	15
dialout	18
floppy	19
games	20
slocate	21
utmp	22
squid	23
named	25
postgres	26
mysql	27
nscd	28
nscd	28
rpcuser	29
rpc	32
cinta	33
utempter	35
kvm	36

Nombre del grupo	GID
video	39
mailnull	47
Apache	48
ftp	50
smmsp	51
Tomcat	53
bloqueo	54
ldap	55
tss	59
audio	63
avahi	70
tcpdump	72
sshd	74
radvd	75
saslauth	76
dbus	81
screen	84
wbpriv	88
postfix	89
postdrop	90

Nombre del grupo	GID
dovecot	97
users	100
input	104
render	105
sgx	106
mock	135
stapusr	156
stapusr	156
stapsys	157
stapsys	157
stapdev	158
stapdev	158
avahi-autoipd	170
pulse	171
rtkit	172
sanlock	179
systemd-journal	190
systemd-network	192
systemd-resolve	193
usbmon	959

Nombre del grupo	GID
wireshark	960
uidd	961
stap-server	962
systemd-journal-remote	963
usershares	964
redis6	965
pesign	966
smtpq	967
smtpd	968
nginx	969
munge	970
memcached	971
sphinx	972
tracing	973
haproxy	974
flatpak	975
debuginfod	976
dovnull	977
dnsmasq	978
unbound	979

Nombre del grupo	GID
clamscan	980
clamilt	981
virusgroup	982
virusgroup	982
virusgroup	982
clamupdate	983
printadmin	984
colord	985
ods	986
docker	987
aws-kinesis-agent-user	988
cwagent	989
pulse-rt	990
pulse-access	991
ec2-instance-connect	993
chrony	994
systemd-timesync	995
systemd-coredump	996
libstoragemgmt	997
ssh_keys	998

Nombre del grupo	GID
systemd-oom	999
ec2-user	1000
ne	1001
polkitd	9920
nobody	65534

Listado por nombre

Nombre del grupo	GID
adm	4
Apache	48
audio	63
avahi	70
avahi-autoipd	170
aws-kinesis-agent-user	988
bin	1
cdrom	11
chrony	994
clamilt	981
clamscan	980
clamupdate	983

Nombre del grupo	GID
colord	985
cwagent	989
daemon	2
dbus	81
debuginfod	976
dialout	18
disco	6
disco	6
dnsmasq	978
docker	987
dovecot	97
dovnull	977
ec2-instance-connect	993
ec2-user	1 000
flatpak	975
floppy	19
ftp	50
games	20
haproxy	974
input	104

Nombre del grupo	GID
kmem	9
kvm	36
ldap	55
libstoragemgmt	997
bloqueo	54
lp	7
correo	12
correo	12
mailnull	47
man	15
mem	8
memcached	971
mock	135
munge	970
mysql	27
named	25
ne	1001
nginx	969
nobody	65534
nscd	28

Nombre del grupo	GID
nscd	28
ods	986
pesign	966
polkitd	9920
postdrop	90
postfix	89
postgres	26
printadmin	984
pulse	171
pulse-access	991
pulse-rt	990
radvd	75
redis6	965
render	105
root	0
rpc	32
rpcuser	29
rtkit	172
sanlock	179
saslauth	76

Nombre del grupo	GID
screen	84
sgx	106
slocate	21
smmsp	51
smtpd	968
smtpq	967
sphinx	972
squid	23
ssh_keys	998
sshd	74
stap-server	962
stapdev	158
stapdev	158
stapsys	157
stapsys	157
stapusr	156
stapusr	156
sys	3
systemd-coredump	996
systemd-journal	190

Nombre del grupo	GID
systemd-journal-remote	963
systemd-network	192
systemd-oom	999
systemd-resolve	193
systemd-timesync	995
cinta	33
tcpdump	72
Tomcat	53
tracing	973
tss	59
tty	5
unbound	979
usbmon	959
users	100
usershares	964
utempter	35
utmp	22
uudd	961
video	39
virusgroup	982

Nombre del grupo	GID
virusgroup	982
virusgroup	982
wbpriv	88
wheel	10
wireshark	960

Lista de códecs disponibles en 023 AL2

AL2023 proporciona una selección de códecs multimedia a través de sus repositorios estándar. En esta página se proporciona información general de los códecs y sus casos de uso típicos.

Important

El uso y la distribución de los códecs incluidos en Amazon Linux pueden requerir la obtención de derechos de licencia de terceros, incluidos los propietarios o licenciantes de determinados formatos de audio y vídeo de terceros. Usted es el único responsable de obtener estas licencias y de pagar las regalías o tasas necesarias.

Códec	Description (Descripción)
flac	Un códec de audio sin pérdida gratuito y de código abierto que comprime el audio sin perder datos ni calidad, utilizado habitualmente para almacenar audio de alta calidad
fdk-aac-free	Una implementación de código abierto del estándar AAC (Advanced Audio Codec), que proporciona compresión de audio de alta calidad para alternativas como la transmisión o el almacenamiento de archivos MP3
webrtc-audio-processing	Una biblioteca para el procesamiento de audio utilizada en WebRTC (comunicación web en tiempo real), que ofrece funciones como supresión de ruido, cancelación de eco y control de ganancia
opus	Un códec de audio altamente versátil y eficiente diseñado para la transmisión en tiempo real, que ofrece baja latencia y es compatible con una amplia gama de aplicacio

Códec	Description (Descripción)
	nes de audio, incluidas VoIP y transmisión de música
libsndfile	Biblioteca para leer y escribir archivos de audio en varios formatos (como WAV, AIFF y FLAC), que se utiliza habitualmente en las herramientas de procesamiento y manipulación de audio
openh264	Implementación de código abierto del códec de vídeo H.264. Proporciona soporte de codificación y decodificación para el vídeo H.264 utilizado en aplicaciones de streaming, conferencias y multimedia.
svt-av1	Implementación de código abierto y de alto rendimiento del códec de AV1 vídeo. Proporciona codificación y decodificación escalables para AV1 vídeo, optimizadas para el procesamiento moderno CPUs y paralelo.
dav1d	Una implementación de código abierto del decodificador de AV1 vídeo centrada en la velocidad, la eficiencia y la portabilidad. Proporciona una AV1 decodificación de alto rendimiento para una amplia gama de plataformas y aplicaciones.
libde265	Una implementación de código abierto del decodificador de vídeo H.265/HEVC. Proporciona una decodificación eficiente y portátil de flujos de vídeo HEVC para su uso en aplicaciones y marcos multimedia.

Códec	Description (Descripción)
libheif	Una biblioteca de código abierto para leer y escribir archivos de imagen HEIF y AVIF. Proporciona codificación, decodificación y conversión eficientes de imágenes y secuencias de imágenes utilizando formatos de compresión modernos como HEVC y AV1
mpg123	Un decodificador de alto rendimiento para transmisiones de audio MPEG, incluidos los archivos. MP3 Proporciona una decodificación de audio rápida y precisa para aplicaciones de reproducción, transmisión y procesamiento de audio.
x265	El objetivo principal del x265 es convertirse en el mejor codificador H.265/HEVC disponible en el mercado, ofreciendo la máxima eficiencia de compresión y el máximo rendimiento en una amplia variedad de plataformas de hardware. Este paquete contiene el codificador de línea de comandos.

Seguridad y conformidad en Amazon Linux 2023

Important

Si desea denunciar una vulnerabilidad o tiene algún problema de seguridad relacionado con los servicios en la AWS nube o los proyectos de código abierto, póngase en contacto con el departamento de AWS seguridad a través de la [página de informes de vulnerabilidades](#)

La seguridad en la nube AWS es la máxima prioridad. Como AWS cliente, usted se beneficia de una arquitectura de centro de datos y red diseñada para cumplir con los requisitos de las organizaciones más sensibles a la seguridad.

La seguridad es una responsabilidad compartida entre usted AWS y usted. El [modelo de responsabilidad compartida](#) la describe como seguridad de la nube y seguridad en la nube:

- Seguridad de la nube: AWS es responsable de proteger la infraestructura que ejecuta AWS los servicios en la AWS nube. AWS también le proporciona servicios que puede utilizar de forma segura. Los auditores externos prueban y verifican periódicamente la eficacia de nuestra seguridad como parte de los [AWS programas](#) de de . Para obtener más información sobre los programas de cumplimiento aplicables AL2023, consulte [AWS Servicios incluidos en el ámbito de aplicación por programa de conformidad y AWS servicios incluidos](#) .
- Seguridad en la nube: su responsabilidad viene determinada por el servicio de AWS que utilice. También es responsable de otros factores, incluida la confidencialidad de los datos, los requisitos de la empresa y la legislación y los reglamentos vigentes.

Temas

- [Avisos de seguridad de Amazon Linux para AL2023](#)
- [Lista de avisos aplicables](#)
- [Aplicación de actualizaciones de seguridad in situ](#)
- [Configuración de los modos de SELinux para AL2023](#)
- [Habilitar el modo FIPS en AL2023](#)
- [Habilitación del modo FIPS en un contenedor de AL2023](#)
- [Intercambio de los proveedores de FIPS de OpenSSL en AL2023](#)

- [AL2023 Endurecimiento del núcleo](#)
- [Inicio de sesión de metadatos del repositorio AL2023](#)
- [UEFI Secure Boot está activado AL2023](#)

Avisos de seguridad de Amazon Linux para AL2023

Aunque nos esforzamos por garantizar la seguridad de Amazon Linux, en ocasiones aparecerán problemas de seguridad que deberán solucionarse. Se emite un aviso cuando hay una solución disponible. El lugar principal donde publicamos los avisos es el Centro de seguridad de Amazon Linux (ALAS). Para obtener más información, consulte [Centro de seguridad de Amazon Linux](#).

Important

Si desea informar sobre una vulnerabilidad o tiene algún problema de seguridad relacionado con los servicios en la AWS nube o los proyectos de código abierto, póngase en contacto con el departamento de AWS seguridad a través de la página de informes de [vulnerabilidades](#)

El equipo de Amazon Linux publica información sobre AL2023 los problemas y las actualizaciones relevantes que los afectan en varios lugares. Es habitual que las herramientas de seguridad obtengan información de estos orígenes principales y le presenten los resultados. Por lo tanto, es posible que usted no interactúe directamente con los orígenes principales que publica Amazon Linux, sino con la interfaz proporcionada por su herramienta preferida, como [Amazon Inspector](#).

Anuncios en el Centro de seguridad de Amazon Linux

Los anuncios de Amazon Linux se refieren a elementos que no se pueden incluir en un aviso. Esta sección contiene anuncios sobre el propio ALAS, junto con información que no se puede incluir en un aviso. Para obtener más información, consulte [Anuncios del Centro de seguridad de Amazon Linux \(ALAS\)](#).

Por ejemplo, el [anuncio del hotpatch de Amazon Linux para Apache Log4j - 2021-001](#) se ajustaba más a un anuncio que a un aviso. En este anuncio, Amazon Linux añadió un paquete para ayudar a los clientes a mitigar un problema de seguridad en un software que no formaba parte de Amazon Linux.

El [Explorador de CVE del Centro de seguridad de Amazon Linux](#) también se anunció en los anuncios del ALAS. Para obtener más información, consulte el [nuevo sitio web de CVEs](#).

Preguntas frecuentes sobre el Centro de seguridad de Amazon Linux

Para obtener respuestas a algunas preguntas frecuentes sobre ALAS y sobre cómo evalúa Amazon Linux CVEs, consulte las [Preguntas frecuentes sobre Amazon Linux Security Center \(ALAS\) \(FAQs\)](#).

Avisos de ALAS

Un aviso de Amazon Linux contiene información importante relevante para los usuarios de Amazon Linux, normalmente información sobre actualizaciones de seguridad. El [Centro de seguridad de Amazon Linux](#) es el lugar donde se pueden ver los avisos en la web. La información de los avisos también forma parte de los metadatos del repositorio de paquetes RPM.

Avisos y repositorios de RPM

Un repositorio de paquetes de Amazon Linux 2023 puede contener metadatos que describan cero o más actualizaciones. El comando `dnf updateinfo` recibe su nombre del nombre del archivo de metadatos del repositorio que contiene esta información, `updateinfo.xml`. Aunque el comando se llama `updateinfo` y el archivo de metadatos hace referencia a un `update`, todos ellos se refieren a actualizaciones de paquetes que forman parte de un aviso.

Los avisos de Amazon Linux se publican en el sitio web del [Centro de seguridad de Amazon Linux](#), junto con la información que se encuentra en los metadatos del repositorio de RPM a los que hace referencia el administrador de paquetes `dnf`. Con el tiempo, los metadatos del sitio web y del repositorio son consistentes, y es posible que haya inconsistencias en la información del sitio web y en los metadatos del repositorio. Esto suele ocurrir cuando se AL2023 está publicando una nueva versión de, si se ha producido una actualización de un aviso después de la AL2023 publicación más reciente.

Aunque lo habitual es que se publique un nuevo aviso junto con la actualización del paquete que soluciona el problema, no siempre es así. Se puede crear un aviso para un nuevo problema que se aborda en paquetes ya publicados. Un aviso existente también puede actualizarse con otros nuevos CVEs que se aborden en la actualización existente.

La [Actualizaciones deterministas mediante repositorios versionados en AL2023](#) función de Amazon Linux 2023 significa que el repositorio RPM de una AL2023 versión concreta contiene una instantánea de los metadatos del repositorio RPM de esa versión. Esto incluye los metadatos que describen las actualizaciones de seguridad. El repositorio RPM de una AL2023 versión determinada no se actualiza después de la publicación. Los avisos de seguridad nuevos o actualizados no estarán visibles al consultar una versión anterior de los repositorios AL2023 RPM. Consulta la [Lista de](#)

[avisos aplicables](#) sección sobre cómo usar el administrador de `dnf` paquetes para ver la versión del `latest` repositorio o una versión específica AL2023 .

Asesoramiento IDs

Cada aviso tiene su propio id. Actualmente, es una peculiaridad de Amazon Linux, donde el sitio web del [Amazon Linux Security Center](#) incluirá un aviso como [ALAS-2024-581](#), mientras que el administrador de `dnf` paquetes [incluirá ese aviso con el identificador -2024-581. ALAS2023](#) Cuando ocurre [Aplicación de actualizaciones de seguridad in situ](#), es necesario usar el ID del administrador de paquetes si se refiere a un aviso específico.

En Amazon Linux, cada versión principal del sistema operativo tiene su propio espacio de nombres, Advisory. IDs No se deben hacer suposiciones en cuanto al formato del aviso de Amazon Linux IDs. Históricamente, Amazon Linux Advisory IDs ha seguido el patrón de `NAMESPACE-YEAR-NUMBER`. El rango completo de valores posibles para `NAMESPACE` no está definido, pero incluye `ALAS`, `ALASCORRETT08`, `ALAS2023`, `ALAS2`, `ALASPYTHON3.8` y `ALASUNBOUND-1.17`. `YEAR` ha sido el año en que se creó el aviso y `NUMBER` es un entero único dentro del espacio de nombres.

Si bien las IDs recomendaciones suelen ser secuenciales y se muestran en el orden en que se publican las actualizaciones, hay muchas razones por las que no puede ser así, por lo que no se debe dar por sentado.

Trate el ID de aviso como una cadena opaca que es única para cada versión principal de Amazon Linux.

En Amazon Linux 2, cada Extra estaba en un repositorio de RPM independiente y los metadatos de los avisos se encuentran únicamente en el repositorio para el que son relevantes. Un aviso de un repositorio no se aplica a otro repositorio. En el sitio web del [Centro de seguridad de Amazon Linux](#), actualmente hay una lista de avisos para cada versión principal de Amazon Linux y no está separada en listas por repositorio.

Como AL2023 no utiliza el mecanismo Extras para empaquetar versiones alternativas de paquetes, actualmente solo hay dos repositorios RPM, cada uno de los cuales tiene Advisories: el `core` repositorio y el repositorio. `livepatch` El repositorio `livepatch` es para [Aplicación de parches activos del kernel en AL2023](#).

Fecha de publicación del aviso y fecha de actualización del aviso

La fecha de publicación de los avisos de Amazon Linux indica cuándo se publicó por primera vez la actualización de seguridad en el repositorio de RPM. Los avisos se publican en el sitio web del

[Centro de seguridad de Amazon Linux](#) inmediatamente después de que las correcciones estén disponibles para su instalación a través del repositorio de RPM.

La fecha de actualización del aviso indica cuándo se ha añadido nueva información a un aviso después de su publicación previa.

No se debe hacer ninguna suposición entre el número de AL2023 versión (p. ej., 2023.6.20241031) y la fecha de publicación del aviso de publicación de los avisos publicados junto con esa versión.

Tipos de avisos

Los metadatos del repositorio de RPM admiten avisos de diferentes tipos. Aunque Amazon Linux casi siempre ha publicado únicamente avisos que son actualizaciones de seguridad, no se debe dar esto por sentado. Es posible que se publiquen avisos sobre eventos tales como correcciones de errores, mejoras y nuevos paquetes, y que el aviso se marque como contenedor de ese tipo de actualización.

Gravedad de los avisos

Cada aviso tiene su propia gravedad, ya que cada problema se evalúa por separado. Se CVEs pueden abordar varias en una sola advertencia y cada CVE puede tener una evaluación diferente, pero la propia advertencia tiene un nivel de gravedad único. Puede haber varios avisos que hagan referencia a una única actualización de paquete, por lo que puede haber varias gravedades para una actualización de paquete concreta (una por cada aviso).

Por orden decreciente de gravedad, Amazon Linux ha utilizado los niveles “Crítica”, “Importante”, “Moderada” y “Baja” para indicar la gravedad de un aviso. Es posible que los avisos de Amazon Linux tampoco tengan un nivel de gravedad, aunque esto es muy poco frecuente.

Amazon Linux es una de las distribuciones de Linux basadas en RPM que utiliza el término “Moderada”, mientras que otras distribuciones de Linux basadas en RPM utilizan el término equivalente “Media”. El administrador de paquetes de Amazon Linux trata ambos términos como equivalentes y los repositorios de paquetes de terceros pueden usar el término “Media”.

Los avisos de Amazon Linux pueden cambiar de gravedad con el tiempo a medida que se obtenga más información sobre los problemas relevantes abordados en el aviso.

Por lo general, la gravedad de un aviso registrará la puntuación CVSS más alta evaluada por Amazon Linux para la puntuación a la que CVEs hace referencia el aviso. Puede haber casos en los

que esto no sea así. Un ejemplo sería cuando se aborda un problema al que no se le ha asignado un CVE.

Consulte las [preguntas frecuentes de ALAS](#) para obtener más información sobre cómo Amazon Linux utiliza los índices de gravedad de los avisos.

Avisos y paquetes

Puede haber muchos avisos para un solo paquete, y no todos los paquetes tendrán un aviso publicado. Se puede hacer referencia a una versión de paquete concreta en varios avisos, cada uno con su propia gravedad y. CVEs

Es posible que se publiquen varios avisos para la misma actualización de paquete simultáneamente en una nueva AL2023 versión o en rápida sucesión.

Al igual que otras distribuciones de Linux, puede haber uno o varios paquetes binarios diferentes compilados a partir del mismo paquete de origen. Por ejemplo, [ALAS-2024-698](#) es un aviso que aparece en la sección [AL2023 del sitio web de Amazon Linux Security Center](#) como aplicable al paquete. `mariadb105` Este es el nombre del paquete de origen, y el propio aviso hace referencia a los paquetes binarios que aparecen junto con el paquete fuente. En este caso, se crean más de una docena de paquetes binarios a partir de un único paquete `mariadb105` de origen. Aunque es muy habitual que haya un paquete binario con el mismo nombre que el paquete de origen, esto no es universal.

Aunque las advertencias de Amazon Linux suelen incluir todos los paquetes binarios creados a partir del paquete de origen actualizado, no se debe dar esto por sentado. El formato de metadatos del administrador de paquetes y del repositorio de RPM permite incluir avisos que enumeran un subconjunto de los paquetes binarios actualizados.

Una advertencia concreta también puede aplicarse únicamente a una arquitectura de CPU concreta. Puede haber paquetes que no estén compilados para todas las arquitecturas, o problemas que no afecten a todas las arquitecturas. En el caso de que un paquete esté disponible en todas las arquitecturas, pero un problema solo afecte a una de ellas, Amazon Linux no suele publicar un aviso en el que solo se haga referencia a la arquitectura afectada, aunque esto no debe darse por sentado.

Debido a la naturaleza de las dependencias entre paquetes, es habitual que una notificación haga referencia a un paquete, pero la instalación de esa actualización requiera otras actualizaciones de paquetes, incluidos paquetes que no figuran en la notificación. El administrador de paquetes `dnf` se encargará de instalar las dependencias necesarias.

Avisos y CVEs

Un aviso puede abordar cero o más CVEs, y puede haber varios avisos que hagan referencia al mismo CVE.

Un ejemplo en el que un aviso puede hacer referencia a cero CVEs es cuando un CVE aún no está asignado (o nunca) a la emisión.

Un ejemplo en el que varios avisos pueden hacer referencia al mismo CVE cuando (por ejemplo) el CVE es aplicable a varios paquetes. Por ejemplo, [CVE-2024-21208](#) se aplica a Corretto 8, 11, 17 y 21. [Cada una de estas versiones de Corretto viene en un paquete independiente y hay un aviso para cada uno de estos paquetes: ALAS-2024-754 para el Corretto 8 AL2023, ALAS-2024-753 para el Corretto 11, ALAS-2024-752 para el Corretto 17 y ALAS-2024-752 para el Corretto 21.](#) Si bien todos estos lanzamientos de Corretto tienen la misma lista de CVEs, esto no debe asumirse.

Un CVE concreto puede evaluarse de forma diferente para distintos paquetes. Por ejemplo, si se hace referencia a un CVE concreto en un aviso con un nivel de gravedad “Importante”, es posible que se publique otro aviso que haga referencia al mismo CVE con un nivel de gravedad diferente.

Los metadatos del repositorio de RPM permiten obtener una lista de referencias para cada aviso. Si bien Amazon Linux normalmente solo hace referencia CVEs, el formato de metadatos permite otros tipos de referencia.

Solo se hará referencia a los metadatos del repositorio de paquetes RPM CVEs cuando haya una corrección disponible. La [sección Explore del sitio web de Amazon Linux Security Center](#) contiene información sobre CVEs lo que Amazon Linux ha evaluado. Esta evaluación puede dar como resultado una puntuación base CVSS, un nivel de gravedad y un estado para varias versiones y paquetes de Amazon Linux. El estado de un CVE para una versión o paquete concreto de Amazon Linux puede ser “No afectado”, “Pendiente de corrección” o “Sin corrección prevista”. El estado y la evaluación de CVEs pueden cambiar muchas veces y de cualquier manera antes de que se emita un aviso. Esto incluye la reevaluación de la aplicabilidad de un CVE a Amazon Linux.

La lista de CVEs referencias de una Asesoría puede cambiar después de la publicación inicial de esa Asesoría.

Texto de aviso

Un aviso también contendrá un texto que describa el problema o los problemas que motivaron su creación. Es habitual que este texto sea el texto del CVE sin modificaciones. Este texto puede hacer

referencia a los números de versión anteriores en los que hay una corrección disponible y que son diferentes de la versión del paquete a la que Amazon Linux ha aplicado una corrección. Es habitual que Amazon Linux transfiera las correcciones de las versiones anteriores más recientes. En el caso de que el texto del aviso mencione una versión anterior diferente a la versión incluida en una versión de Amazon Linux, las versiones del paquete de Amazon Linux del aviso serán las correctas para Amazon Linux.

Es posible que el texto de advertencia en los metadatos del repositorio RPM sea un texto de marcador de posición que simplemente remita al sitio web del [Centro de seguridad de Amazon Linux](#) para obtener más detalles.

Avisos de parches activos del kernel

Los avisos de los parches activos son únicos en el sentido de que se refieren a un paquete diferente (el kernel de Linux) del paquete en el que se basa el aviso (por ejemplo, `kernel-livepatch-6.1.15-28.43`).

Un aviso para un [parche activo del núcleo](#) hará referencia a los problemas (por ejemplo CVEs) que el paquete de parches activos concreto puede solucionar en relación con la versión específica del núcleo a la que se aplica el paquete de parches activos.

Cada parche activo es para una versión específica del kernel. Para aplicar un parche activo a un CVE, es necesario instalar el paquete de parches activos adecuado para la versión del kernel y aplicar el parche activo.

Por ejemplo, el [CVE-2023-6111](#) puede ser parcheado en tiempo real para AL2023 las versiones del núcleo, y. 6.1.56-82.125 6.1.59-84.139 6.1.61-85.141 También se ha lanzado una nueva versión del kernel con una corrección para este CVE, que cuenta con un [aviso independiente](#). Para corregir el error [CVE-2023-6111](#), se debe ejecutar una versión del núcleo igual o posterior a la especificada por [ALAS2023-2023-461](#), o AL2023 bien una de las versiones del núcleo con un parche activo para este CVE debe estar ejecutándose con el parche activo correspondiente.

Cuando hay nuevos parches activos disponibles para una versión específica del kernel que ya tiene un parche activo disponible, se publica una nueva versión del paquete `kernel-livepatch-KERNEL_VERSION`. Por ejemplo, el [ALASLIVEPATCH-2023-003](#) aviso se publicó con un `kernel-livepatch-6.1.15-28.43-1.0-1.amzn2023` paquete que contenía tres parches activos para el núcleo. 6.1.15-28.43 CVEs Más tarde, se publicó el [ALASLIVEPATCH-2023-009](#) aviso junto con el `kernel-livepatch-6.1.15-28.43-1.0-2.amzn2023` paquete, una actualización del anterior

paquete de parches activos para el 6.1.15-28.43 núcleo, que contenía parches activos para otros tres CVEs. También hubo otros problemas con los avisos de parches activos para otras versiones del kernel, con paquetes que contenían parches activos para esas versiones específicas del kernel.

Para obtener más información sobre la aplicación de parches activos, consulte [Aplicación de parches activos del kernel en AL2023](#).

Para cualquier persona que esté desarrollando herramientas en torno a los avisos de seguridad, también se recomienda consultar la sección [Esquema XML para avisos y updateinfo.xml](#) para obtener más información.

Esquema XML para avisos y **updateinfo.xml**

El archivo `updateinfo.xml` forma parte del formato del repositorio de paquetes. Son los metadatos que el administrador de paquetes `dnf` analiza para implementar funciones como [Lista de avisos aplicables](#) y [Aplicación de actualizaciones de seguridad in situ](#).

Recomendamos utilizar la API del administrador de paquetes `dnf` en lugar de escribir código personalizado para analizar los formatos de metadatos del repositorio. La versión de `dnf` in AL2023 puede analizar tanto el formato como el del AL2 repositorio AL2023 y, por lo tanto, la API se puede utilizar para examinar la información de asesoramiento de cualquier versión del sistema operativo.

El proyecto [RPM Software Management](#) documenta los formatos de metadatos RPM del repositorio [rpm-metadata](#) en. GitHub

Para aquellos que están desarrollando herramientas para analizar directamente los metadatos de `updateinfo.xml`, se recomienda encarecidamente prestar mucha atención a la [documentación sobre los metadatos de RPM](#). La documentación cubre lo que se ha observado en la práctica, lo que incluye muchas excepciones a lo que se podría interpretar razonablemente como una regla para el formato de metadatos.

También hay un conjunto cada vez mayor de ejemplos de `updateinfo.xml` archivos del mundo real en el repositorio [raw-historical-rpm-repository-examples](#) on. GitHub

En caso de que algo no quede claro en la documentación, puedes abrir una edición en el GitHub proyecto para que podamos responder a la pregunta y actualizar la documentación adecuadamente. Al tratarse de proyectos de código abierto, también se aceptan solicitudes de incorporación de cambios para actualizar la documentación.

Lista de avisos aplicables

El administrador de paquetes `dnf` tiene acceso a los metadatos que describen qué avisos están fijos en qué versiones del paquete. De este modo, puede enumerar qué avisos son aplicables a una imagen de instancia o contenedor.

Note

Herramientas como [AWS Systems Manager](#) pueden usar esta funcionalidad para mostrar qué actualizaciones son relevantes para una flota, en lugar de solo una instancia.

Al enumerar las actualizaciones, puedes indicar a `dnf` que consulte los metadatos de una versión de AL2023 concreta o los metadatos de la última versión.

Note

Una vez que se publica una versión de AL2023, no se modifica. Por lo tanto, los avisos nuevos o actualizados del [Centro de seguridad de Amazon Linux](#) solo se añaden a los metadatos de las nuevas versiones de AL2023.

A continuación, analizaremos algunos ejemplos de las recomendaciones que se aplican a algunas imágenes de contenedores de AL2023. Todos estos comandos funcionan en entornos no contenerizados, como las instancias de EC2.

Listing advisories in a specific version

En este ejemplo, analizaremos qué avisos están disponibles en la versión [2023.1.20230628](#) si lanzamos un contenedor de la versión [2023.0.20230315](#).

Note

En este ejemplo se utilizan las versiones [2023.0.20230315](#) y [2023.1.20230628](#), y estas no son las versiones más recientes de AL2023. Consulte las [notas de la versión de AL2023](#) para ver las versiones más recientes, que contienen las actualizaciones de seguridad más recientes.

En este ejemplo, comenzaremos con una imagen de contenedor para la versión [2023.0.20230315](#).

En primer lugar, obtenemos esta imagen del contenedor del registro del contenedor. `.0` al final indica la versión de la imagen para una versión concreta; esta versión de imagen suele ser cero.

```
$ docker pull public.ecr.aws/amazonlinux/amazonlinux:2023.0.20230315.0
2023.0.20230315.0: Pulling from amazonlinux/amazonlinux
b76f3b09316a: Pull complete
Digest: sha256:94e7183b0739140dbd5b639fb7600f0a2299cec5df8780c26d9cb409da5315a9
Status: Downloaded newer image for public.ecr.aws/amazonlinux/
amazonlinux:2023.0.20230315.0
public.ecr.aws/amazonlinux/amazonlinux:2023.0.20230315.0
```

Ahora podemos generar un intérprete de comandos dentro del contenedor, desde el que pediremos a `dnf` que indique qué avisos son relevantes para los paquetes instalados en el contenedor.

```
$ docker run -it public.ecr.aws/amazonlinux/amazonlinux:2023.0.20230315.0
bash-5.2#
```

El comando `dnf updateinfo` ahora se usa para mostrar un resumen de los avisos de la versión [2023.1.20230628](#) que son relevantes para nuestros paquetes instalados.

```
$ dnf updateinfo --releasever=2023.1.20230628
Amazon Linux 2023 repository                42 MB/s | 15 MB    00:00
Last metadata expiration check: 0:00:02 ago on Mon Jul 22 20:24:24 2024.
Updates Information Summary: available
  8 Security notice(s)
    1 Important Security notice(s)
    5 Medium Security notice(s)
    2 Low Security notice(s)
```

Para obtener una lista de los avisos, se puede utilizar la opción `--list` con `dnf updateinfo`.

```
$ dnf updateinfo --releasever=2023.1.20230628 --list
Last metadata expiration check: 0:01:22 ago on Mon Jul 22 20:24:24 2024.
ALAS2023-2023-193 Medium/Sec.    curl-minimal-8.0.1-1.amzn2023.x86_64
ALAS2023-2023-225 Medium/Sec.    glib2-2.74.7-688.amzn2023.0.1.x86_64
ALAS2023-2023-195 Low/Sec.       libcap-2.48-2.amzn2023.0.3.x86_64
ALAS2023-2023-193 Medium/Sec.    libcurl-minimal-8.0.1-1.amzn2023.x86_64
```

ALAS2023-2023-145	Low/Sec.	libgcc-11.3.1-4.amzn2023.0.3.x86_64
ALAS2023-2023-145	Low/Sec.	libgomp-11.3.1-4.amzn2023.0.3.x86_64
ALAS2023-2023-145	Low/Sec.	libstdc++-11.3.1-4.amzn2023.0.3.x86_64
ALAS2023-2023-163	Medium/Sec.	libxml2-2.10.4-1.amzn2023.0.1.x86_64
ALAS2023-2023-220	Important/Sec.	ncurses-base-6.2-4.20200222.amzn2023.0.4.noarch
ALAS2023-2023-220	Important/Sec.	ncurses-libs-6.2-4.20200222.amzn2023.0.4.x86_64
ALAS2023-2023-181	Medium/Sec.	openssl-libs-1:3.0.8-1.amzn2023.0.2.x86_64
ALAS2023-2023-222	Medium/Sec.	openssl-libs-1:3.0.8-1.amzn2023.0.3.x86_64

Listing advisories in the latest version

En este ejemplo vamos a ver qué actualizaciones están disponibles en la versión latest de AL2023 si lanzamos un contenedor de la versión [2023.4.20240319](#). En el momento de escribir este artículo, la versión latest es la [2023.5.20240708](#), por lo que las actualizaciones que se muestran en este ejemplo son las correspondientes a esa versión.

Note

En este ejemplo, se utilizan las versiones [2023.4.20240319](#) y [2023.5.20240708](#), siendo esta última la versión más reciente en el momento de redactar este artículo. Para obtener más información sobre las versiones más recientes, consulte las [notas de la versión de AL2023](#).

En este ejemplo, comenzaremos con una imagen de contenedor para la versión [2023.4.20240319](#).

En primer lugar, obtenemos esta imagen del contenedor del registro del contenedor. `.1` al final indica la versión de la imagen para una versión concreta. Aunque la versión de la imagen suele ser cero, este ejemplo utiliza una versión en la que la versión de la imagen es uno.

```
$ docker pull public.ecr.aws/amazonlinux/amazonlinux:2023.4.20240319.1
2023.4.20240319.1: Pulling from amazonlinux/amazonlinux
6de065fda9a2: Pull complete
Digest: sha256:b4838c4cc9211d966b6ea158dacc9eda7433a16ba94436508c2d9f01f7658b4e
Status: Downloaded newer image for public.ecr.aws/amazonlinux/
amazonlinux:2023.4.20240319.1
public.ecr.aws/amazonlinux/amazonlinux:2023.4.20240319.1
```

Ahora podemos generar un intérprete de comandos dentro del contenedor, desde el que comprobaremos si hay actualizaciones.

```
$ docker run -it public.ecr.aws/amazonlinux/amazonlinux:2023.4.20240319.1
bash-5.2#
```

El comando `dnf updateinfo` ahora se usa para mostrar un resumen de los avisos de la última versión que son relevantes para nuestros paquetes instalados. En el momento de escribir este artículo, la versión más reciente era la [2023.1.20230628](#).

```
$ dnf --releasever=latest updateinfo
Amazon Linux 2023 repository              76 MB/s | 25 MB      00:00
Last metadata expiration check: 0:00:04 ago on Mon Jul 22 20:59:54 2024.
Updates Information Summary: available
  9 Security notice(s)
    4 Important Security notice(s)
    4 Medium Security notice(s)
    1 Low Security notice(s)
```

Para obtener una lista de los avisos, se puede utilizar la opción `--list` con `dnf updateinfo`.

```
$ dnf updateinfo --releasever=latest --list
Last metadata expiration check: 0:00:58 ago on Mon Jul 22 20:59:54 2024.
ALAS2023-2024-581 Low/Sec.      curl-minimal-8.5.0-1.amzn2023.0.3.x86_64
ALAS2023-2024-596 Medium/Sec.  curl-minimal-8.5.0-1.amzn2023.0.4.x86_64
ALAS2023-2024-576 Important/Sec. expat-2.5.0-1.amzn2023.0.4.x86_64
ALAS2023-2024-589 Important/Sec. glibc-2.34-52.amzn2023.0.10.x86_64
ALAS2023-2024-589 Important/Sec. glibc-common-2.34-52.amzn2023.0.10.x86_64
ALAS2023-2024-589 Important/Sec. glibc-minimal-langpack-2.34-52.amzn2023.0.10.x86_64
ALAS2023-2024-586 Medium/Sec.  krb5-libs-1.21-3.amzn2023.0.4.x86_64
ALAS2023-2024-581 Low/Sec.      libcurl-minimal-8.5.0-1.amzn2023.0.3.x86_64
ALAS2023-2024-596 Medium/Sec.  libcurl-minimal-8.5.0-1.amzn2023.0.4.x86_64
ALAS2023-2024-592 Important/Sec. libnghttp2-1.59.0-3.amzn2023.0.1.x86_64
ALAS2023-2024-640 Medium/Sec.  openssl-libs-1:3.0.8-1.amzn2023.0.12.x86_64
ALAS2023-2024-605 Medium/Sec.  python3-3.9.16-1.amzn2023.0.7.x86_64
ALAS2023-2024-616 Important/Sec. python3-3.9.16-1.amzn2023.0.8.x86_64
ALAS2023-2024-605 Medium/Sec.  python3-libs-3.9.16-1.amzn2023.0.7.x86_64
ALAS2023-2024-616 Important/Sec. python3-libs-3.9.16-1.amzn2023.0.8.x86_64
```

Aplicación de actualizaciones de seguridad in situ

Para obtener información general sobre la aplicación de actualizaciones, consulte [Aplicación de actualizaciones de seguridad mediante DNF y las versiones de repositorio](#). La opción `--security`

para `dnf upgrade` restringirá las actualizaciones de los paquetes solo a aquellos que tengan un aviso. En el resto de esta sección se explicará cómo instalar únicamente actualizaciones de seguridad específicas.

 Note

Se recomienda aplicar todas las actualizaciones disponibles en una nueva versión de AL2023. Elegir solo actualizaciones de seguridad o solo actualizaciones específicas debería ser la excepción y no la regla.

Aplicación de las actualizaciones mencionadas en un aviso

Los identificadores de los avisos de la primera columna del resultado de `dnf upgradeinfo` se pueden utilizar para aplicar actualizaciones a los paquetes mencionados en el aviso. Se le puede dar instrucciones al administrador de paquetes `dnf` para que actualice los paquetes del aviso a la última versión disponible o solo a las versiones mencionadas en el aviso. Si las actualizaciones ya están instaladas, el comando de actualización no tiene ningún efecto.

Para aplicar las actualizaciones a los paquetes afectados únicamente hasta la versión mencionada en el aviso, utilice el comando `dnf upgrade-minimal` y utilice la opción `--advisory` para especificar el aviso. En el siguiente ejemplo se ejecuta `dnf upgrade-minimal` en un contenedor de la versión [2023.0.20230315](#) de AL2023.

```
$ dnf upgrade-minimal -y --releasever=2023.1.20230628 --advisory ALAS2023-2023-193
Amazon Linux 2023 repository                               46 MB/s | 15 MB      00:00
Last metadata expiration check: 0:00:03 ago on Mon Jul 22 20:36:13 2024.
Dependencies resolved.
=====
Package                Arch      Version                Repository            Size
=====
Upgrading:
curl-minimal            x86_64    8.0.1-1.amzn2023      amazonlinux           150 k
libcurl-minimal         x86_64    8.0.1-1.amzn2023      amazonlinux           249 k

Transaction Summary
=====
Upgrade 2 Packages

Total download size: 399 k
Downloading Packages:
```

```

(1/2): curl-minimal-8.0.1-1.amzn2023.x86_64.rpm 2.7 MB/s | 150 kB      00:00
(2/2): libcurl-minimal-8.0.1-1.amzn2023.x86_64. 3.8 MB/s | 249 kB      00:00
-----
Total                               2.5 MB/s | 399 kB      00:00
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing      :                                1/1
  Upgrading      : libcurl-minimal-8.0.1-1.amzn2023.x86_64 1/4
  Upgrading      : curl-minimal-8.0.1-1.amzn2023.x86_64    2/4
  Cleanup        : curl-minimal-7.88.1-1.amzn2023.0.1.x86_64 3/4
  Cleanup        : libcurl-minimal-7.88.1-1.amzn2023.0.1.x86_64 4/4
  Running scriptlet: libcurl-minimal-7.88.1-1.amzn2023.0.1.x86_64 4/4
  Verifying      : libcurl-minimal-8.0.1-1.amzn2023.x86_64 1/4
  Verifying      : libcurl-minimal-7.88.1-1.amzn2023.0.1.x86_64 2/4
  Verifying      : curl-minimal-8.0.1-1.amzn2023.x86_64    3/4
  Verifying      : curl-minimal-7.88.1-1.amzn2023.0.1.x86_64 4/4

Upgraded:
  curl-minimal-8.0.1-1.amzn2023.x86_64  libcurl-minimal-8.0.1-1.amzn2023.x86_64

Complete!

```

Las mismas versiones del paquete se actualizan incluso si se utiliza `--releasever=latest`, ya que la solicitud es que `dnf` realice la actualización mínima necesaria para abordar el aviso.

El uso del comando habitual `dnf upgrade` con la opción `--advisory` actualizará los paquetes pertinentes mencionados en el aviso a la última versión disponible, que puede ser más reciente que la versión mencionada en el aviso.

Note

A menos que se actualice el paquete `system-release`, la versión de los repositorios de AL2023 a la que está vinculado `dnf` no cambia.

Warning

Al instalar actualizaciones de una versión diferente de AL2023 sin cambiar la versión del repositorio a la que está vinculado `dnf`, se debe tener cuidado con cualquier operación

dnf de mutación posterior. Por ejemplo, al instalar o actualizar paquetes, dado que las dependencias de los paquetes pueden haber cambiado en la versión más reciente, es posible que la versión anterior en la que permanezca no pueda satisfacer estas nuevas dependencias.

El siguiente ejemplo se ejecuta en un contenedor de la versión [2023.0.20230315](#) de AL2023 en referencia a la última versión de AL2023, que en el momento de redactar este documento era [2023.5.20240708](#). Tenga en cuenta que tanto la versión de curl a la que se actualiza como la versión a la que se actualiza update-minimal son más recientes, pero que esta versión más reciente introduce nuevas dependencias.

```
$ dnf upgrade -y --releasever=latest --advisory ALAS2023-2023-193
Amazon Linux 2023 repository                80 MB/s | 25 MB      00:00
Last metadata expiration check: 0:00:04 ago on Mon Jul 22 20:48:38 2024.
Dependencies resolved.
=====
Package                                Arch      Version                                Repository      Size
=====
Upgrading:
curl-minimal                          x86_64    8.5.0-1.amzn2023.0.4                  amazonlinux     160 k
libcurl-minimal                       x86_64    8.5.0-1.amzn2023.0.4                  amazonlinux     275 k
libnghttp2                            x86_64    1.59.0-3.amzn2023.0.1                 amazonlinux     79 k
Installing dependencies:
libpsl                                x86_64    0.21.1-3.amzn2023.0.2                 amazonlinux     61 k
publicsuffix-list-dafsa               noarch    20240212-61.amzn2023                  amazonlinux     59 k

Transaction Summary
=====
Install  2 Packages
Upgrade  3 Packages

Total download size: 634 k
Downloading Packages:
(1/5): publicsuffix-list-dafsa-20240212-61.amzn 1.1 MB/s | 59 kB      00:00
(2/5): curl-minimal-8.5.0-1.amzn2023.0.4.x86_64 2.6 MB/s | 160 kB     00:00
(3/5): libpsl-0.21.1-3.amzn2023.0.2.x86_64.rpm  949 kB/s | 61 kB      00:00
(4/5): libnghttp2-1.59.0-3.amzn2023.0.1.x86_64. 3.7 MB/s | 79 kB      00:00
(5/5): libcurl-minimal-8.5.0-1.amzn2023.0.4.x86 6.7 MB/s | 275 kB     00:00
-----
Total                                3.5 MB/s | 634 kB     00:00
Running transaction check
```

```

Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing      :                                1/1
  Upgrading      : libnghttp2-1.59.0-3.amzn2023.0.1.x86_64 1/8
  Installing     : publicsuffix-list-dafsa-20240212-61.amzn2023.noarch 2/8
  Installing     : libpsl-0.21.1-3.amzn2023.0.2.x86_64 3/8
  Upgrading      : libcurl-minimal-8.5.0-1.amzn2023.0.4.x86_64 4/8
  Upgrading      : curl-minimal-8.5.0-1.amzn2023.0.4.x86_64 5/8
  Cleanup        : curl-minimal-7.88.1-1.amzn2023.0.1.x86_64 6/8
  Cleanup        : libcurl-minimal-7.88.1-1.amzn2023.0.1.x86_64 7/8
  Cleanup        : libnghttp2-1.51.0-1.amzn2023.x86_64 8/8
  Running scriptlet: libnghttp2-1.51.0-1.amzn2023.x86_64 8/8
  Verifying      : libpsl-0.21.1-3.amzn2023.0.2.x86_64 1/8
  Verifying      : publicsuffix-list-dafsa-20240212-61.amzn2023.noarch 2/8
  Verifying      : curl-minimal-8.5.0-1.amzn2023.0.4.x86_64 3/8
  Verifying      : curl-minimal-7.88.1-1.amzn2023.0.1.x86_64 4/8
  Verifying      : libcurl-minimal-8.5.0-1.amzn2023.0.4.x86_64 5/8
  Verifying      : libcurl-minimal-7.88.1-1.amzn2023.0.1.x86_64 6/8
  Verifying      : libnghttp2-1.59.0-3.amzn2023.0.1.x86_64 7/8
  Verifying      : libnghttp2-1.51.0-1.amzn2023.x86_64 8/8

Upgraded:
  curl-minimal-8.5.0-1.amzn2023.0.4.x86_64
  libcurl-minimal-8.5.0-1.amzn2023.0.4.x86_64
  libnghttp2-1.59.0-3.amzn2023.0.1.x86_64
Installed:
  libpsl-0.21.1-3.amzn2023.0.2.x86_64
  publicsuffix-list-dafsa-20240212-61.amzn2023.noarch

Complete!

```

Configuración de los modos de SELinux para AL2023

Para AL2023, Security Enhanced Linux (SELinux) de forma predeterminada está enabled y está configurado en modo permissive. En el modo permisivo, las denegaciones de permisos se registran pero no se aplican. SELinux es un conjunto de características y utilidades del kernel que proporcionan una arquitectura de control de acceso (MAC) sólida, flexible y obligatoria a los principales subsistemas del kernel.

SELinux proporciona un mecanismo mejorado para hacer cumplir la separación de la información en función de los requisitos de confidencialidad e integridad. Esta separación de la información reduce las amenazas de manipulación y elusión de los mecanismos de seguridad de las aplicaciones. También limita los daños que pueden causar las aplicaciones malintencionadas o defectuosas.

SELinux incluye un conjunto de ejemplos de archivos de configuración de políticas de seguridad diseñados para cumplir con los objetivos de seguridad diarios.

Para obtener más información sobre las características y funcionalidades de SELinux, consulte el [cuaderno de SELinux](#) y la [política de lenguajes](#).

Temas

- [Estado y modos predeterminados de SELinux para AL2023](#)
- [Cambiar al modo enforcing](#)
- [Opción para deshabilitar SELinux para AL2023](#)

Estado y modos predeterminados de SELinux para AL2023

Para AL2023, SELinux está configurado de forma predeterminada como `enabled` y en modo `permissive`. En el modo `permissive`, las denegaciones de permisos se registran pero no se aplican.

Los comandos **getenforce** o **sestatus** indican el estado, la política y el modo actuales de SELinux.

Con el estado predeterminado establecido en `enabled` y `permissive`, el comando **getenforce** vuelve a `permissive`.

El **sestatus** comando devuelve el estado de SELinux y la política de SELinux actual, como se muestra en el siguiente ejemplo:

```
$ sestatus
SELinux status:                enabled
SELinuxfs mount:              /sys/fs/selinux
SELinux root directory:       /etc/selinux
Loaded policy name:            targeted
Current mode:                  permissive
Mode from config file:         permissive
Policy MLS status:             enabled
```

```
Policy deny_unknown status:    allowed
Memory protection checking:    actual (secure)
Max kernel policy version:    33
```

Al ejecutar SELinux en modo `permissive`, los usuarios a veces pueden etiquetar los archivos de forma incorrecta. Cuando se ejecuta SELinux en ese estado `disabled`, los archivos no se etiquetan. Tanto los archivos incorrectos como los que no están etiquetados pueden causar problemas al cambiar al modo `enforcing`.

SELinux reetiqueta automáticamente los archivos para evitar este problema. SELinux evita problemas de etiquetado con el reetiquetado automático al cambiar el estado a `enabled`.

Cambiar al modo **enforcing**

Cuando se ejecuta SELinux en modo `enforcing`, la utilidad SELinux `enforcing` la política configurada. SELinux regula las capacidades de determinadas aplicaciones al permitir o denegar el acceso según las reglas de la política.

Para encontrar el modo SELinux actual, ejecute el comando `getenforce`.

```
getenforce
Permissive
```

Edite el archivo de configuración para habilitar el modo **enforcing**

Para cambiar el modo a `enforcing`, siga los pasos que se indican a continuación.

1. Edite el archivo `/etc/selinux/config` para cambiar al modo `enforcing`. El ajuste `SELINUX` debe ser similar a la siguiente especificación de ejemplo:

```
SELINUX=enforcing
```

2. Reinicie el sistema para completar el cambio al modo `enforcing`.

```
$ sudo reboot
```

En el siguiente arranque, SELinux vuelve a etiquetar todos los archivos y directorios del sistema. SELinux también añade el contexto de SELinux para los archivos y directorios que se crearon cuando SELinux estaba `disabled`.

Después de cambiar al modo **enforcing**, SELinux podría denegar algunas acciones debido a reglas de política de SELinux faltantes o incorrectas. Para ver las acciones que SELinux niega, use el comando siguiente.

```
$ sudo ausearch -m AVC,USER_AVC,SELINUX_ERR,USER_SELINUX_ERR -ts recent
```

Use cloud-init para habilitar el modo **enforcing**

Como alternativa, cuando lance la instancia, pasa el siguiente `cloud-config` como datos de usuario para habilitar el modo **enforcing**.

```
#cloud-config
selinux:
  mode: enforcing
```

De forma predeterminada, esta opción hace que la instancia se reinicie. Para una mayor estabilidad, le recomendamos que reinicie la instancia. No obstante, si lo prefiere, puede omitir el reinicio. Para ello, proporcione el siguiente `cloud-config`.

```
#cloud-config
selinux:
  mode: enforcing
  selinux_no_reboot: 1
```

Opción para deshabilitar SELinux para AL2023

Al deshabilitar SELinux, la política de SELinux no se carga ni se aplica y los mensajes AVC no se registran. Pierde todos los beneficios para ejecutar SELinux.

En lugar de deshabilitar SELinux, recomendamos usar el modo **permissive**. Ejecutarlo en modo **permissive** cuesta solo un poco más que deshabilitar SELinux por completo. La transición del modo **permissive** al modo **enforcing** requiere muchos menos ajustes de configuración que volver a un modo **enforcing** después de deshabilitar SELinux. Puede etiquetar los archivos y el sistema puede rastrear y registrar las acciones que la política activa podría haber denegado.

Cambiar del modo SELinux al modo **permissive**

Cuando se ejecuta SELinux en modo **permissive**, la política de SELinux no se aplica. En el modo **permissive**, SELinux registra los mensajes AVC pero no deniega las operaciones. Puede usar estos mensajes AVC para solucionar problemas, depurar y mejorar las políticas de SELinux.

Para cambiar SELinux al modo permisivo, siga estos pasos.

1. Edite el archivo `/etc/selinux/config` para cambiar al modo `permissive`. El valor SELINUX debería ser similar al siguiente ejemplo.

```
SELINUX=permissive
```

2. Reinicie el sistema para completar el cambio al modo `permissive`.

```
sudo reboot
```

Deshabilitar SELinux

Al deshabilitar SELinux, la política de SELinux no se carga ni se aplica y los mensajes AVC no se registran. Pierde todos los beneficios para ejecutar SELinux.

Para deshabilitar SELinux, siga estos pasos:

1. Asegúrese de que el paquete `grubby` esté instalado.

```
rpm -q grubby  
grubby-version
```

2. Configure su cargador de arranque para añadir `selinux=0` a la línea de comandos del kernel.

```
sudo grubby --update-kernel ALL --args selinux=0
```

3. Reinicie el sistema.

```
sudo reboot
```

4. Ejecute el comando `getenforce` para confirmar que SELinux está Disabled.

```
$ getenforce  
Disabled
```

Para obtener más información sobre SELinux, consulte [Cuaderno SELinux](#) y [Configuración de SELinux](#).

Habilitar el modo FIPS en AL2023

En esta sección se explica cómo habilitar los estándares federales de procesamiento de información (FIPS) en AL2023. Para obtener más información sobre FIPS, consulte:

- [Estándar Federal de Procesamiento de la Información \(FIPS\)](#)
- [Preguntas frecuentes sobre conformidad: Estándar Federal de Procesamiento de la Información](#)

Note

En esta sección se explica cómo activar el modo FIPS en AL2023, pero no se trata del estado de certificación de los módulos criptográficos de AL2023.

Requisitos previos

- Una instancia de Amazon EC2 AL2023 (AL2023.2 o superior) existente con acceso a Internet para descargar los paquetes necesarios. Para obtener más información sobre cómo lanzar una instancia de AL2023 en Amazon EC2, consulte [Lanzamiento de AL2023 con la consola de Amazon EC2](#).
- Debe conectarse a la instancia de Amazon EC2 mediante SSH o AWS Systems Manager. Para obtener más información, consulte [Conexión a instancias AL2023](#).

Important

Las claves de usuario SSH ED25519 no se admiten en el modo FIPS. Si lanzó su instancia de Amazon EC2 con un par de claves SSH ED25519, debe generar nuevas claves mediante otro algoritmo (como RSA) o podría perder el acceso a la instancia después de habilitar el modo FIPS. Para obtener más información, consulte [Creación de pares de clave](#) en la Guía del usuario de Amazon EC2.

Habilitar el modo FIPS

1. Conéctese a la instancia de AL2023 mediante SSH o AWS Systems Manager.

2. Asegúrese de que el sistema esté actualizado. Para obtener más información, consulte [Gestione las actualizaciones de paquetes y sistemas operativos en AL2023](#).
3. Asegúrese de que las utilidades `crypto-policies` estén instaladas y actualizadas.

```
sudo dnf -y install crypto-policies crypto-policies-scripts
```

4. Habilite el modo FIPS ejecutando el siguiente comando. Esto habilitará el modo FIPS en todo el sistema para los módulos enumerados en las [preguntas frecuentes de AL2023](#)

```
sudo fips-mode-setup --enable
```

5. Ejecute el siguiente comando para volver a arrancar la instancia.

```
sudo reboot
```

6. Para verificar que el modo FIPS está habilitado, vuelva a conectarse a la instancia y ejecute el siguiente comando.

```
sudo fips-mode-setup --check
```

En la siguiente salida de ejemplo se muestra que el modo FIPS está activado:

```
FIPS mode is enabled.
```

Habilitación del modo FIPS en un contenedor de AL2023

En esta sección se explica cómo habilitar los estándares federales de procesamiento de información (FIPS) en un contenedor de AL2023. Para obtener más información sobre FIPS, consulte:

- [Estándar Federal de Procesamiento de la Información \(FIPS\)](#)
- [Preguntas frecuentes sobre conformidad: Estándar Federal de Procesamiento de la Información](#)

Note

Esta sección documenta cómo habilitar el modo FIPS en un contenedor de AL2023. No cubre el estado de certificación de los módulos criptográficos de AL2023.

Requisitos previos

- Una instancia de Amazon EC2 AL2023 (AL2023.2 o superior) existente con acceso a Internet para descargar los paquetes necesarios. Para obtener más información sobre cómo lanzar una instancia de AL2023 en Amazon EC2, consulte [Lanzamiento de AL2023 con la consola de Amazon EC2](#).
- Debe conectarse a la instancia de Amazon EC2 mediante SSH o AWS Systems Manager. Para obtener más información, consulte [Conexión a instancias AL2023](#).

Important

El comando `fips-mode-setup` no funcionará correctamente desde el contenedor. Lea los pasos que se indican a continuación para configurar correctamente el modo FIPS en un contenedor de AL2023.

Habilitación del modo FIPS en un contenedor de AL2023

1. El modo FIPS debe habilitarse primero en el host del contenedor de AL2023. Siga las instrucciones que aparecen en [Habilitar el modo FIPS en AL2023](#) para habilitar el modo FIPS en el host.
2. Conéctese a su instancia de host del contenedor de AL2023 con SSH o AWS Systems Manager.
3. El modo FIPS se habilitará automáticamente en un contenedor de AL2023 si el host de AL2023 está en modo FIPS y se puede acceder a `/proc/sys/crypto/fips_enabled` desde dentro del contenedor. Si el contenido de `/proc/sys/crypto/fips_enabled` es `0`, FIPS no está habilitado y un valor de `1` indica que el modo FIPS está habilitado.

Puede verificar que FIPS está habilitado ejecutando el siguiente comando tanto en el host como en el contenedor de AL2023:

```
cat /proc/sys/crypto/fips_enabled
```

4. A continuación, habilite las crypto-políticas de FIPS en el contenedor. Existen varias formas de lograrlo, que se describen en las siguientes opciones. Seleccione la opción que sea mejor para su entorno.
 - a. Habilite las crypto-políticas de FIPS manualmente en el contenedor mediante el comando `update-crypto-policies`:

```
# Run these commands inside the container
dnf install -y crypto-policies-scripts
update-crypto-policies --set FIPS
```

- b. Cree montajes de bind dentro del contenedor de AL2023 (esto es similar a cómo funciona podman en otras distribuciones):

```
# Run these commands inside the container
mount --bind /usr/share/crypto-policies/back-ends/FIPS /etc/crypto-policies/back-ends
echo "FIPS" > /usr/share/crypto-policies/default-fips-config
mount --bind /usr/share/crypto-policies/default-fips-config /etc/crypto-policies/config
```

- c. También es posible crear un montaje de unión para que el contenedor de AL2023 coincida con las crypto-policies del anfitrión de AL2023. Puede ver un ejemplo en el siguiente comando: Esta configuración podría causar problemas si hay diferencias incompatibles en las crypto-policies y las versiones de los paquetes entre el contenedor y el host:

```
sudo docker pull amazonlinux:2023
sudo docker run --mount type=bind,readonly,src=/etc/crypto-policies,dst=/etc/crypto-policies -it amazonlinux:2023
```

5. Tras realizar los pasos anteriores, puede volver a comprobar que FIPS esté habilitado en el contenedor con los siguientes comandos:

```
$ cat /etc/crypto-policies/config
FIPS

$ cat /proc/sys/crypto/fips_enabled
1
```

Intercambio de los proveedores de FIPS de OpenSSL en AL2023

En esta sección se explica cómo cambiar entre los proveedores de FIPS de OpenSSL latest y certified en AL2023.

Para obtener más información sobre FIPS, consulte:

- [Estándar Federal de Procesamiento de la Información \(FIPS\)](#)
- [Preguntas frecuentes sobre conformidad: Estándar Federal de Procesamiento de la Información](#)
- [Política de FedRAMP para la selección y el uso de módulos criptográficos](#)

Important

En AL2023.7 y versiones posteriores, el proveedor de FIPS de OpenSSL predeterminado es el paquete `openssl-fips-provider-latest`, que recibe correcciones de errores y actualizaciones de seguridad periódicas.

Las instrucciones que aparecen a continuación son únicamente para los clientes que desean fijar un pin al paquete `openssl-fips-provider-certified`. Esta versión del proveedor de FIPS coincidirá con la suma de verificación del certificado NIST y es posible que no tenga las actualizaciones más recientes.

Consulte las [preguntas frecuentes de AL2023](#) para obtener más información sobre los módulos y las versiones de paquetes con certificación FIPS.

Requisitos previos

- Una instancia de Amazon EC2 AL2023 (AL2023.7 o superior) existente con acceso a Internet para descargar los paquetes necesarios. Para obtener más información sobre cómo lanzar una instancia de AL2023 en Amazon EC2, consulte [Lanzamiento de AL2023 con la consola de Amazon EC2](#).
- Debe conectarse a la instancia de Amazon EC2 mediante SSH o AWS Systems Manager. Para obtener más información, consulte [Conexión a instancias AL2023](#).
- Para activar el modo FIPS en AL2023, siga las instrucciones que se indican en [Habilitar el modo FIPS en AL2023](#).

Cambie entre **`openssl-fips-provider-latest`** y **`openssl-fips-provider-certified`**.

1. Se utiliza `dnf` para cambiar el proveedor de FIPS de OpenSSL:

```
sudo dnf -y swap openssl-fips-provider-latest openssl-fips-provider-certified
```

2. Compruebe que utiliza el proveedor de FIPS de OpenSSL certificado. Con AL2023 en modo FIPS, ejecute el siguiente comando:

```
openssl list -providers
```

Debería ver los siguientes datos de salida:

```
Providers:
  base
    name: OpenSSL Base Provider
    version: 3.2.2
    status: active
  default
    name: OpenSSL Default Provider
    version: 3.2.2
    status: active
  fips
    name: Amazon Linux 2023 - OpenSSL FIPS Provider
    version: 3.0.8-d694bfa693b76001
    status: active
```

AL2023 Endurecimiento del núcleo

El núcleo 6.1 de Linux AL2023 está configurado y construido con varias opciones y funciones de refuerzo.

Opciones de fortalecimiento del kernel (independientes de la arquitectura).

Opción CONFIG	AL2023/6.1/aarch64	AL2023/6.1/x86_64	AL2023/6.12/aarch64	AL2023/6.12/x86_64	AL2023/6.18/aarch64	AL2023/6.18/x86_64
CONFIG_ACPI_CUSTOM_METHOD	n	n	N/A	N/A	N/A	N/A
CONFIG_BINFMT_MISC	m	m	m	m	m	m

Opción CONFIG	AL2023/6. 1/aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/aarch64	AL2023/6. 12/x86_64	AL2023/6. 18/aarch64	AL2023/6. 18/x86_64
CONFIG_BUG	y	y	y	y	y	y
CONFIG_BUG_ON_DATA_CORRUPTION	y	y	y	y	y	y
CONFIG_CLANG_I	N/A	N/A	N/A	N/A	N/A	N/A
CONFIG_CLANG_I_PERMISSIVE	N/A	N/A	N/A	N/A	N/A	N/A
CONFIG_COMPAT	y	y	y	y	y	y
CONFIG_COMPAT_BRK	n	n	n	n	n	n
CONFIG_COMPAT_VDSO	N/A	n	N/A	n	N/A	n
CONFIG_DEBUG_CREDENTIALS	n	n	N/A	N/A	N/A	N/A
CONFIG_DEBUG_LIST	y	y	y	y	y	y
CONFIG_DEBUG_NOTIFIERS	n	n	n	n	n	n

Opción CONFIG	AL2023/6. 1/aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/aarch64	AL2023/6. 12/x86_64	AL2023/6. 18/aarch64	AL2023/6. 18/x86_64
<u>CONFIG_DE BUG_SG</u>	n	n	n	n	n	n
<u>CONFIG_DE BUG_VIRTU AL</u>	n	n	n	n	n	n
<u>CONFIG_DE BUG_WX</u>	n	n	n	n	n	n
<u>CONFIG_DE FAULT_MMA P_MIN_ADD R</u>	65536	65536	65536	65536	65536	65536
<u>CONFIG_DE VKMEM</u>	N/A	N/A	N/A	N/A	N/A	N/A
<u>CONFIG_DE VMEM</u>	n	n	n	n	n	n
<u>CONFIG_EF I_DISABLE _PCI_DMA</u>	n	n	n	n	n	n
<u>CONFIG_F0 RTIFY_SOU RCE</u>	y	y	y	y	y	y
<u>CONFIG_HA RDENED_US ERCOPY</u>	y	y	y	y	y	y

Opción CONFIG	AL2023/6. 1/aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/aarch64	AL2023/6. 12/x86_64	AL2023/6. 18/aarch64	AL2023/6. 18/x86_64
CONFIG_HARDENED_USERCOPY_FALLBACK	N/A	N/A	N/A	N/A	N/A	N/A
CONFIG_HARDENED_USERCOPY_PAGESPAN	N/A	N/A	N/A	N/A	N/A	N/A
CONFIG_HIBERNATION	y	y	y	y	y	y
CONFIG_HW_RANDOM_TPM	N/A	N/A	N/A	N/A	N/A	N/A
CONFIG_INET_DIAG	m	m	m	m	m	m
CONFIG_INET_ON_ALL_OSC_DEFAULT_T_ON	n	n	n	n	n	n
CONFIG_INET_ON_FREESTACK_DEFAULT_ON	n	n	n	n	n	n
CONFIG_INET_STACK_ALL_ZERO	N/A	N/A	N/A	N/A	N/A	N/A

Opción CONFIG	AL2023/6. 1/aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/aarch64	AL2023/6. 12/x86_64	AL2023/6. 18/aarch64	AL2023/6. 18/x86_64
<u>CONFIG_IO MMU_DEFAU LT_DMA_ST RICT</u>	n	n	n	n	n	n
<u>CONFIG_IO MMU_SUPPO RT</u>	y	y	y	y	y	y
<u>CONFIG_IO _STRICT_D EVMEM</u>	N/A	N/A	N/A	N/A	N/A	N/A
<u>CONFIG_KE XEC</u>	y	y	y	y	y	y
<u>CONFIG_KF ENCE</u>	n	n	n	n	n	n
<u>CONFIG_LD ISC_AUTOL OAD</u>	n	n	n	n	n	n
<u>CONFIG_LE GACY_PTYS</u>	n	n	n	n	n	n
<u>CONFIG_LO CK_DOWN_K ERNEL_FOR CE_CONFID ENTIALITY</u>	n	n	n	n	n	n

Opción CONFIG	AL2023/6. 1/aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/aarch64	AL2023/6. 12/x86_64	AL2023/6. 18/aarch64	AL2023/6. 18/x86_64
<u>CONFIG_MODULES</u>	y	y	y	y	y	y
<u>CONFIG_MODULE_SIG</u>	y	y	y	y	y	y
<u>CONFIG_MODULE_SIG_ALL</u>	y	y	y	y	y	y
<u>CONFIG_MODULE_SIG_FORCE</u>	n	n	n	n	n	n
<u>CONFIG_MODULE_SIG_HASH</u>	sha512	sha512	sha512	sha512	sha512	sha512
<u>CONFIG_MODULE_SIG_KEY</u>	certs/ sig ning_key. pem	certs/ sig ning_key. pem	certs/ sig ning_key. pem	certs/ sig ning_key. pem	certs/ sig ning_key. pem	certs/ sig ning_key. pem
<u>CONFIG_MODULE_SIG_SHA512</u>	y	y	y	y	y	y
<u>CONFIG_PACKAGE_POISONING</u>	n	n	n	n	n	n
<u>CONFIG_PACKAGE_POISONING_NO_SANITY</u>	N/A	N/A	N/A	N/A	N/A	N/A

Opción CONFIG	AL2023/6. 1/aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/aarch64	AL2023/6. 12/x86_64	AL2023/6. 18/aarch64	AL2023/6. 18/x86_64
CONFIG_PAGE_POISONING_ZERO	N/A	N/A	N/A	N/A	N/A	N/A
CONFIG_PANIC_ON_OOPS	y	y	y	y	y	y
CONFIG_PANIC_TIMEOUT	0	0	0	0	0	0
CONFIG_PARTITION_KCORE	y	y	y	y	y	y
CONFIG_RANDOMIZE_KERNEL_STACK_OFFSET_DEFAULT	n	n	n	n	n	n
CONFIG_RANDOMIZE_TRUST_BOOTLOADER	y	y	N/A	N/A	N/A	N/A
CONFIG_RANDOMIZE_TRUST_CPU	y	y	N/A	N/A	N/A	N/A
CONFIG_REFCOUNT_FULL	N/A	N/A	N/A	N/A	N/A	N/A

Opción CONFIG	AL2023/6. 1/aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/aarch64	AL2023/6. 12/x86_64	AL2023/6. 18/aarch64	AL2023/6. 18/x86_64
<u>CONFIG_SC HED_CORE</u>	N/A	y	N/A	y	N/A	y
<u>CONFIG_SC HED_STACK _END_CHEC K</u>	y	y	y	y	y	y
<u>CONFIG_SE CCOMP</u>	y	y	y	y	y	y
<u>CONFIG_SE CCOMP_FIL TER</u>	y	y	y	y	y	y
<u>CONFIG_SE CURITY</u>	y	y	y	y	y	y
<u>CONFIG_SE CURITY_DM ESG_RESTR ICT</u>	y	y	y	y	y	y
<u>CONFIG_SE CURITY_LA NDLOCK</u>	y	y	y	y	y	y
<u>CONFIG_SE CURITY_LO CKDOWN_LS M</u>	y	y	y	y	y	y

Opción CONFIG	AL2023/6. 1/aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/aarch64	AL2023/6. 12/x86_64	AL2023/6. 18/aarch64	AL2023/6. 18/x86_64
<u>CONFIG_SECURITY_LOCKDOWN_LSM_EARLY</u>	y	y	y	y	y	y
<u>CONFIG_SECURITY_SELINUX_BOOTPARAM</u>	y	y	y	y	y	y
<u>CONFIG_SECURITY_SELINUX_DEVELOP</u>	y	y	y	y	y	y
<u>CONFIG_SECURITY_SELINUX_DISABLE</u>	n	n	N/A	N/A	N/A	N/A
<u>CONFIG_SECURITY_SELINUX_HOLES</u>	N/A	N/A	N/A	N/A	N/A	N/A
<u>CONFIG_SECURITY_SELINUX_YAMA</u>	y	y	y	y	y	y
<u>CONFIG_SHUFFLE_PAGE_ALLOCATOR</u>	y	y	y	y	y	y

Opción CONFIG	AL2023/6. 1/aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/aarch64	AL2023/6. 12/x86_64	AL2023/6. 18/aarch64	AL2023/6. 18/x86_64
CONFIG_SLAB_FREEELIST_HARDENED	y	y	y	y	y	y
CONFIG_SLAB_FREEELIST_RANDOM	y	y	y	y	y	y
CONFIG_SLUB_DEBUG	y	y	y	y	y	y
CONFIG_STACKPROTECTOR	y	y	y	y	y	y
CONFIG_STACKPROTECTOR_STRONG	y	y	y	y	y	y
CONFIG_STATIC_USERMODEHELPER	n	n	n	n	n	n
CONFIG_STRICK_DEVMEM	n	n	n	n	n	n
CONFIG_STRICK_KERNEL_EL_RWX	y	y	y	y	y	y

Opción CONFIG	AL2023/6. 1/aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/aarch64	AL2023/6. 12/x86_64	AL2023/6. 18/aarch64	AL2023/6. 18/x86_64
CONFIG_STRICT_MODULE_RWX	y	y	y	y	y	y
CONFIG_SYSCALL_COOKIES	y	y	y	y	y	y
CONFIG_VMAP_STACK	y	y	y	y	y	y
CONFIG_WERROR	n	n	n	n	n	n
CONFIG_ZERO_CALL_UNSED_REGS	n	n	n	n	n	n

Permitir que los métodos ACPI estén en tiempo de ejecución (CONFIG_ACPI_CUSTOM_METHOD) inserted/replaced

Amazon Linux deshabilita esta opción porque permite a los usuarios root escribir en una memoria del kernel arbitraria.

Esta opción es una de las [configuraciones recomendadas por el proyecto de autoprotección del kernel](#).

Varios formatos binarios (**binfmt_misc**)

Aunque esta opción es una de las opciones recomendadas por el [Proyecto de autoprotección del núcleo \(KSPP\)](#), no establece esta opción de configuración con los valores recomendados por KSPP.

AL2023 En AL2023, esta función es opcional y se ha creado como un módulo del núcleo.

Compatibilidad con **BUG()**

Esta opción es una de las [configuraciones recomendadas por el proyecto de autoprotección del kernel](#).

BUG() si el kernel detecta daños en los datos al comprobar la validez de las estructuras de memoria del kernel

Algunas partes del kernel de Linux comprueban la coherencia interna de las estructuras de datos y pueden emitir **BUG()** cuando detecten daños en los datos.

Esta opción es una de las [configuraciones recomendadas por el proyecto de autoprotección del kernel](#).

COMPAT_BRK

Si esta opción está deshabilitada (que es la forma en que Amazon Linux configura el kernel), la configuración `randomize_va_space sysctl` tiene el valor predeterminado 2, lo que también permite la asignación al azar del montón además de la asignación al azar de la base `mmap`, la pila y las páginas del `VDSO`.

Esta opción existe en kernel para proporcionar compatibilidad con algunos binarios `libc.so.5` antiguos de 1996 y anteriores.

Esta opción es una de las [configuraciones recomendadas por el proyecto de autoprotección del kernel](#).

COMPAT_VDSO

Esta opción de configuración es relevante para `x86-64` y no para `aarch64`. Al configurar en `n`, el kernel de Amazon Linux no hace visible un objeto compartido dinámico virtual (`VDSO`) de 32 bits en una dirección predecible. La opción `glibc` más reciente de la que se tiene constancia al establecer esta opción en `n` es la opción `glibc 2.3.3`, del 2004.

Esta opción es una de las [configuraciones recomendadas por el proyecto de autoprotección del kernel](#).

CONFIG_DEBUG endurecimiento cerrado

Las opciones de configuración del núcleo de Linux controladas por `CONFIG_DEBUG` están diseñadas normalmente para su uso en núcleos creados para solucionar problemas de depuración, y aspectos

como el rendimiento no son prioritarios. AL2023 habilita la opción de CONFIG_DEBUG_LIST endurecimiento.

Deshabilite el DMA para los dispositivos PCI en el código auxiliar EFI antes de configurar el IOMMU

Si bien esta opción es una de las opciones recomendadas por el [Proyecto de autoprotección del núcleo \(KSPP\)](#), AL2023 no establece esta opción de configuración con los valores recomendados por KSPP.

Reforzamiento para copiar memoria entre el núcleo y el espacio de usuario

Cuando el kernel necesita copiar memoria hacia o desde el espacio de usuario, esta opción habilita varias comprobaciones que pueden evitar algunos tipos de problemas de desbordamiento de pila.

La opción CONFIG_HARDENED_USERCOPY_FALLBACK existía en los códigos kernel del 4.16 al 5.15 como ayuda para los desarrolladores del kernel para detectar cualquier entrada de la lista de permitidos que faltara a través de WARN(). Como AL2023 incluye un núcleo 6.1, esta opción ya no es relevante para. AL2023

La CONFIG_HARDENED_USERCOPY_PAGESPAN opción existía en los núcleos principalmente como una opción de depuración para los desarrolladores y ya no se aplica al núcleo 6.1 de. AL2023

Esta opción es una de las [configuraciones recomendadas por el proyecto de autoprotección del kernel](#).

Soporte para hibernación

Aunque esta opción es una de las opciones recomendadas por [el KSPP \(KSPP\)](#), AL2023 no establece esta opción de configuración con los valores recomendados por KSPP. Esta opción debe estar habilitada para permitir la [hibernación de la instancia bajo demanda](#) y para permitir la [hibernación de las instancias de spot interrumpidas](#)

Generación de números aleatorios

El kernel de AL2023 está configurado para garantizar que esté disponible la entropía adecuada para su uso en EC2.

CONFIG_INET_DIAG

Aunque esta opción es una de las configuraciones recomendadas por el [Proyecto de autoprotección del núcleo \(KSPP\)](#), [AL2023 no establece esta opción de configuración con los valores recomendados](#) por KSPP. En AL2023, esta función es opcional y se ha creado como un módulo del núcleo.

Ponga a cero todas las páginas del núcleo y la memoria del asignador de bloques durante la asignación y desasignación

Aunque esta opción es una de las configuraciones recomendadas por el [Proyecto de autoprotección del núcleo \(KSPP\)](#), [AL2023 no establece esta opción de configuración con los valores recomendados](#) por KSPP. Estas opciones están deshabilitadas AL2023 debido al posible impacto en el rendimiento que tendría la activación de esta funcionalidad de forma predeterminada. El comportamiento de CONFIG_INIT_ON_ALLOC_DEFAULT_ON se puede habilitar agregándolo `init_on_alloc=1` a la línea de comandos del kernel, y el comportamiento de CONFIG_INIT_ON_FREE_DEFAULT_ON se puede habilitar agregando `init_on_free=1`.

Inicializar todas las variables de la pila como cero (**CONFIG_INIT_STACK_ALL_ZERO**)

Si bien esta opción es una de las opciones recomendadas por el [Proyecto de autoprotección del núcleo \(KSPP\)](#), [AL2023 no establece esta opción de configuración con los valores recomendados](#) por KSPP. Esta opción requiere GCC 12 o más, mientras que se incluye con GCC 11 AL2023 .

Firma del módulo del núcleo

AL2023 firma y valida las firmas de los módulos del núcleo. La opción CONFIG_MODULE_SIG_FORCE, que requeriría que los módulos tuvieran una firma válida, no está habilitada para preservar la compatibilidad para los usuarios que crean módulos de terceros. Para los usuarios que deseen asegurarse de que todos los módulos del kernel estén firmados, [Módulo de seguridad Lockdown Linux \(LSM\)](#) pueden configurarlos para que así lo exijan.

kexec

Aunque esta opción es una de las opciones recomendadas por el [Proyecto de autoprotección del núcleo \(KSPP\)](#), [AL2023 no establece esta opción de configuración con los valores recomendados](#) por KSPP. Esta opción está habilitada para poder utilizar la funcionalidad `kdump`.

IOMMUSupport

AL2023 permite el soporte del IOMMU. La opción `CONFIG_IOMMU_DEFAULT_DMA_STRICT` no está habilitada de forma predeterminada, pero esta funcionalidad se puede configurar añadiéndola `iommu.passthrough=0 iommu.strict=1` a la línea de comandos del kernel.

kfence

Si bien esta opción es una de las opciones recomendadas por el [Proyecto de autoprotección del núcleo \(KSPP\)](#), AL2023 no establece esta opción de configuración con los valores recomendados por KSPP.

Soporte para **pty** heredado

AL2023 utiliza la interfaz moderna PTY (). `devpts`

Esta opción es una de las [configuraciones recomendadas por el proyecto de autoprotección del kernel](#).

Módulo de seguridad Lockdown Linux (LSM)

AL2023 crea el lockdown LSM, que bloqueará automáticamente el núcleo cuando se utilice Secure Boot.

La opción `CONFIG_LOCK_DOWN_KERNEL_FORCE_CONFIDENTIALITY` ahora no está habilitada. Si bien esta opción es una de las configuraciones recomendadas [por el KSPP \(KSPP\)](#), AL2023 no establece esta opción de configuración con los valores recomendados por KSPP. Cuando no se utiliza el arranque seguro, es posible activar el LSM de bloqueo y configurarlo como se desee.

Envenenamiento de páginas

Si bien esta opción es una de las configuraciones recomendadas por el [Kernel Self Protection Project \(KSPP\)](#), AL2023 no establece esta opción de configuración con los valores recomendados por KSPP. Del mismo modo [Ponga a cero todas las páginas del núcleo y la memoria del asignador de bloques durante la asignación y desasignación](#), está deshabilitada en el AL2023 núcleo debido a su posible impacto en el rendimiento.

Protector de pila

El AL2023 núcleo se ha creado con la función de protección contra pilas que GCC está habilitada con esta opción. `-fstack-protector-strong`

Esta opción es una de las [configuraciones recomendadas por el proyecto de autoprotección del kernel](#).

seccomp BPFABI

La característica seccomp de fortalecimiento se utiliza en programas como systemd y en los tiempos de ejecución de contenedores para reforzar las aplicaciones del espacio de usuario.

Esta opción es una de las [configuraciones recomendadas por el proyecto de autoprotección del kernel](#).

panic() timeout

El AL2023 núcleo está configurado con este valor establecido en 0, lo que significa que el núcleo no se reiniciará cuando entre en pánico. Aunque esta opción es una de las opciones recomendadas por el [Proyecto de autoprotección del núcleo \(KSPP\)](#), AL2023 no establece esta opción de configuración con los valores recomendados por KSPP. Esto se puede configurar mediante `sysctl`, `/proc/sys/kernel/panic` y en la línea de comandos del kernel.

Modelos de seguridad

AL2023 se activa SELinux en modo permisivo de forma predeterminada. Para obtener más información, consulte [Configuración de los modos de SELinux para AL2023](#).

Los módulos [Módulo de seguridad Lockdown Linux \(LSM\)](#) y yama también están habilitados.

/proc/kcore

Si bien esta opción es una de las opciones recomendadas por el [Kernel Self Protection Project \(KSPP\)](#), AL2023 no establece esta opción de configuración con los valores recomendados por KSPP.

La pila del kernel realiza una asignación al azar las entradas al `syscall`

Aunque esta opción es una de las configuraciones recomendadas por el [Proyecto de autoprotección del núcleo \(KSPP\)](#), AL2023 no establece esta opción de configuración con los valores recomendados por KSPP. Esto se puede habilitar configurando `randomize_kstack_offset=on` en la línea de comandos del kernel.

Comprobaciones de recuento de referencias (**CONFIG_REFCOUNT_FULL**)

Aunque esta opción es una de las configuraciones recomendadas por el [Proyecto de autoprotección del núcleo \(KSPP\)](#), AL2023 no establece esta opción de configuración con los valores recomendados por KSPP. Esta opción no está habilitada actualmente debido a su posible impacto en el rendimiento.

Conocimiento de los núcleos SMT por parte del programador (**CONFIG_SCHED_CORE**)

El AL2023 núcleo está construido con `CONFIG_SCHED_CORE`, lo que permite el uso de las aplicaciones del espacio de usuario. `prctl(PR_SCHED_CORE)` Esta opción es una de las [configuraciones recomendadas por el proyecto de autoprotección del kernel](#).

Comprobar si la pila está dañada en las llamadas a **schedule()** (**CONFIG_SCHED_STACK_END_CHECK**)

El AL2023 núcleo se ha creado con `CONFIG_SCHED_STACK_END_CHECK` enabled. Esta opción es una de las [configuraciones recomendadas por el proyecto de autoprotección del kernel](#).

Fortalecimiento del asignador de memoria

El AL2023 núcleo permite reforzar el asignador de memoria del núcleo con las opciones, y. `CONFIG_SHUFFLE_PAGE_ALLOCATOR` `CONFIG_SLAB_FREELIST_HARDENED` `CONFIG_SLAB_FREELIST_RANDOM` Esta opción es una de las [configuraciones recomendadas por el proyecto de autoprotección del kernel](#).

Soporte de depuración SLUB

El AL2023 núcleo `CONFIG_SLUB_DEBUG` lo activa, ya que esta opción habilita funciones de depuración opcionales para el asignador, que se pueden habilitar en la línea de comandos del núcleo. Esta opción es una de las [configuraciones recomendadas por el proyecto de autoprotección del kernel](#).

`CONFIG_STATIC_USERMODEHELPER`

Aunque esta opción es una de las opciones recomendadas por [el KSPP \(KSPP\)](#), AL2023 no establece esta opción de configuración con los valores recomendados por KSPP. Esto se debe a que `CONFIG_STATIC_USERMODEHELPER` requiere un soporte especial por parte de la distribución, que actualmente no está presente en Amazon Linux.

Texto del código en kernel de solo lectura y rodata (**CONFIG_STRICT_KERNEL_RWX** y **CONFIG_STRICT_MODULE_RWX**)

El AL2023 núcleo está configurado para marcar el texto y la memoria del núcleo y los módulos del núcleo como de solo lectura, y la rodata memoria no textual se marca como no ejecutable. Esta opción es una de las [configuraciones recomendadas por el proyecto de autoprotección del kernel](#).

Soporte de TCP syncookie (**CONFIG_SYN_COOKIES**)

El AL2023 núcleo está diseñado con soporte para cookies de sincronización TCP. Esta opción es una de las [configuraciones recomendadas por el proyecto de autoprotección del código kernel](#).

Pila virtualmente mapeada con páginas de protección (**CONFIG_VMAP_STACK**)

El AL2023 núcleo está construido con **CONFIG_VMAP_STACK**, lo que permite disponer de pilas de núcleos mapeadas virtualmente con páginas de protección. Esta opción es una de las [configuraciones recomendadas por el proyecto de autoprotección del código kernel](#).

Compilar con las advertencias del compilador como errores (**CONFIG_WERROR**)

Aunque esta opción es una de las configuraciones recomendadas por el [Proyecto de autoprotección del núcleo \(KSPP\)](#), AL2023 no establece esta opción de configuración con los valores recomendados por KSPP.

Registrar la puesta a cero en la función exit (**CONFIG_ZERO_CALL_USED_REGS**)

Aunque esta opción es una de las configuraciones recomendadas por el [Proyecto de autoprotección del núcleo \(KSPP\)](#), AL2023 no establece esta opción de configuración con los valores recomendados por KSPP.

Dirección mínima para la asignación del espacio de usuario

Esta opción de fortalecimiento puede ayudar a reducir el impacto de los errores del puntero NULL del kernel. Esta opción es una de las [configuraciones recomendadas por el proyecto de autoprotección del kernel](#).

clang opciones de endurecimiento específicas

El AL2023 núcleo está construido con y GCC no clang, por lo que la opción de **CONFIG_CFI_CLANG** endurecimiento no se puede habilitar, lo que también hace que **CONFIG_CFI_PERMISSIVE** no sea

aplicable. Aunque esta opción es una de las opciones recomendadas [por el KSPP \(KSPP\), AL2023 no establece esta opción de configuración con los valores recomendados](#) por KSPP.

Opciones de fortalecimiento del kernel específicas para x86-64

Opción CONFIG	AL2023/6. 1/aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/aarch64	AL2023/6. 12/x86_64	AL2023/6. 18/aarch64	AL2023/6. 18/x86_64
CONFIG_AMD_IOMMU	N/A	y	N/A	y	N/A	y
CONFIG_AMD_IOMMU_V2	N/A	y	N/A	N/A	N/A	N/A
CONFIG_IA32_EMULATION	N/A	y	N/A	y	N/A	y
CONFIG_INTEL_IOMMU	N/A	y	N/A	y	N/A	y
CONFIG_INTEL_IOMMU_DEFAULT_ON	N/A	n	N/A	n	N/A	n
CONFIG_INTEL_IOMMU_SVM	N/A	n	N/A	n	N/A	n
CONFIG_LEGACY_VSYSCALL_NONE	N/A	n	N/A	n	N/A	n

Opción CONFIG	AL2023/6. 1/aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/aarch64	AL2023/6. 12/x86_64	AL2023/6. 18/aarch64	AL2023/6. 18/x86_64
<u>CONFIG_MODIFY_LDT_SYSCALL</u>	N/A	n	N/A	n	N/A	n
<u>CONFIG_PAGE_TABLE_ISOLATION</u>	N/A	y	N/A	N/A	N/A	N/A
<u>CONFIG_RANDOMIZE_MEMORY</u>	N/A	y	N/A	y	N/A	y
<u>CONFIG_X86_64</u>	N/A	y	N/A	y	N/A	y
<u>CONFIG_X86_64_MSR</u>	N/A	y	N/A	y	N/A	y
<u>CONFIG_X86_64_VSYSCALL_EMULATION</u>	N/A	y	N/A	y	N/A	y
<u>CONFIG_X86_64_X32</u>	N/A	N/A	N/A	N/A	N/A	N/A
<u>CONFIG_X86_64_X32_ABI</u>	N/A	n	N/A	n	N/A	n

Compatibilidad con x86-64

La compatibilidad básica con x86-64 incluye la extensión de direcciones físicas (PAE) y la compatibilidad con bits sin ejecución (NX). Esta opción es una de las [configuraciones recomendadas por el proyecto de autoprotección del kernel](#).

Compatibilidad con AMD e Intel IOMMU

El AL2023 núcleo se construye con soporte para AMD e Intel. IOMMUs Esta opción es una de las [configuraciones recomendadas por el proyecto de autoprotección del kernel](#).

La opción `CONFIG_INTEL_IOMMU_DEFAULT_ON` no está configurada, pero se puede activar pasando `intel_iommu=on` a la línea de comandos del kernel. Si bien esta opción es una de las configuraciones recomendadas por el [Proyecto de autoprotección del núcleo \(KSPP\)](#), [AL2023 no establece esta opción de configuración con los valores recomendados](#) por el KSPP.

La `CONFIG_INTEL_IOMMU_SVM` opción no está habilitada actualmente en. AL2023 Si bien esta opción es una de las opciones recomendadas por el [Proyecto de autoprotección del núcleo \(KSPP\)](#), [AL2023 no establece esta opción de configuración con los valores recomendados](#) por KSPP.

Support para espacio de usuario de 32 bits

Important

La compatibilidad con el espacio de usuario x86 de 32 bits está en desuso y es posible que se elimine la compatibilidad con la ejecución de binarios del espacio de usuario de 32 bits en una futura versión principal de Amazon Linux.

Note

Si bien ya AL2023 no incluye ningún paquete de 32 bits, el núcleo seguirá admitiendo la ejecución de un espacio de usuario de 32 bits. Para obtener más información, consulte [Paquetes x86 \(i686\) de 32 bits](#).

Para permitir la ejecución de aplicaciones de espacio de usuario de 32 bits, AL2023 no habilita la `CONFIG_X86_VSYSCALL_EMULATION` opción y habilita las opciones, y. `CONFIG_IA32_EMULATION` `CONFIG_COMPAT` `CONFIG_X86_VSYSCALL_EMULATION` Si bien esta opción es una de las opciones

recomendadas por el [Proyecto de autoprotección del núcleo \(KSPP\)](#), AL2023 no establece esta opción de configuración con los valores recomendados por KSPP.

La ABI nativa de 32 bits x32 para procesadores de 64 bits no está habilitada (CONFIG_X86_X32 y CONFIG_X86_X32_ABI). Esta opción es una de las [configuraciones recomendadas por el proyecto de autoprotección del kernel](#).

Compatibilidad con registro específico de modelo x86 (MSR)

La opción CONFIG_X86_MSR está habilitada para admitir turbostat. Aunque esta opción es una de las configuraciones recomendadas por el [Proyecto de autoprotección del núcleo \(KSPP\)](#), AL2023 no establece esta opción de configuración con los valores recomendados por KSPP.

modify_ldt syscall

AL2023 no permite que los programas de usuario modifiquen la tabla de descriptores locales (LDT) x86 con la llamada al sistema. modify_ldt Esta llamada es necesaria para ejecutar código segmentado o de 16 bits, y su ausencia podría interrumpir el software, por ejemplo dosemu, si se ejecutan algunos programas en WINE y en bibliotecas de subprocessos muy antiguas. Esta opción es una de las [configuraciones recomendadas por el proyecto de autoprotección del kernel](#).

Eliminar el mapeo del kernel en modo de usuario

AL2023 configura el núcleo para que la mayoría de las direcciones del núcleo no se asignen al espacio de usuario. Esta opción es una de las [configuraciones recomendadas por el proyecto de autoprotección del kernel](#).

Asignación al azar de las secciones de memoria del kernel

AL2023 configura el núcleo para distribuir aleatoriamente las direcciones virtuales base de las secciones de memoria del núcleo. Esta opción es una de las [configuraciones recomendadas por el proyecto de autoprotección del kernel](#).

Opciones de refuerzo del kernel específicas de aarch64

Opción CONFIG	AL2023/6. 1/aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/aarch64	AL2023/6. 12/x86_64	AL2023/6. 18/aarch64	AL2023/6. 18/x86_64
CONFIG_ARM64_MTE	y	N/A	y	N/A	y	N/A
CONFIG_ARM64_MTE	y	N/A	y	N/A	y	N/A
CONFIG_ARM64_MTE	y	N/A	y	N/A	y	N/A

Opción CONFIG	AL2023/6. 1/aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/aarch64	AL2023/6. 12/x86_64	AL2023/6. 18/aarch64	AL2023/6. 18/x86_64
CONFIG_ARM64_BTI_KERNEL	N/A	N/A	N/A	N/A	N/A	N/A
CONFIG_ARM64_PTR_AUTH	y	N/A	y	N/A	y	N/A
CONFIG_ARM64_PTR_AUTH_KERNEL	y	N/A	y	N/A	y	N/A
CONFIG_ARM64_SW_TTBR0_PAN	y	N/A	y	N/A	y	N/A
CONFIG_UNMAP_KERNEL_AT_EL0	y	N/A	y	N/A	y	N/A

Identificación del objetivo de ramificaciones

El AL2023 núcleo permite la compatibilidad con Branch Target Identification (). `CONFIG_ARM64_BTI` Esta opción es una de las [configuraciones recomendadas por el proyecto de autoprotección del kernel](#).

La `CONFIG_ARM64_BTI_KERNEL` opción no está habilitada porque está construida con ellaGCC, y la compatibilidad para compilar el núcleo con esta opción está [actualmente deshabilitada en el núcleo original debido](#) a un error de `gcc`. AL2023 Aunque esta opción es una de las configuraciones recomendadas por [el KSPP \(KSPP\)](#), [AL2023 no establece esta opción de configuración con los valores recomendados](#) por KSPP.

Pointer Authentication (**CONFIG_ARM64_PTR_AUTH**)

El AL2023 núcleo se ha creado con soporte para la extensión Pointer Authentication (que forma parte de las extensiones ARMv8 .3), que se puede utilizar para mitigar las técnicas de programación orientada al retorno (ROP). El soporte de hardware necesario para Pointer Authentication en [Graviton](#) se introdujo con Graviton 3.

La opción CONFIG_ARM64_PTR_AUTH está habilitada y admite Pointer Authentication en el espacio de usuario. Como la CONFIG_ARM64_PTR_AUTH_KERNEL opción también está habilitada, el AL2023 núcleo puede utilizar la protección de direcciones de retorno por sí mismo.

Esta opción es una de las [configuraciones recomendadas por el proyecto de autoprotección del kernel](#).

Emular acceso privilegiado sin utilizar nunca la conmutación **TTBR0_EL1**

Esta opción impide que el kernel acceda directamente a la memoria del espacio de usuario, ya que las rutinas de acceso de los usuarios solo establecen temporalmente TTBR0_EL1 en un valor válido.

Esta opción es una de las [configuraciones recomendadas por el proyecto de autoprotección del kernel](#).

Anular la asignación del kernel cuando se ejecute en el espacio de usuario

El AL2023 núcleo está configurado para desmapearlo cuando se ejecuta en el espacio de usuario (CONFIG_UNMAP_KERNEL_AT_EL0). Esta opción es una de las [configuraciones recomendadas por el proyecto de autoprotección del kernel](#).

Inicio de sesión de metadatos del repositorio AL2023

A partir del lanzamiento 2023.11.20260406, AL2023 los repositorios incluyen firmas criptográficas para los metadatos del repositorio. Cada repomd.xml archivo del repositorio va acompañado de un archivo de firma GPG independiente (repomd.xml.asc) que puede utilizar para comprobar la autenticidad e integridad de los metadatos del repositorio antes de descargar los paquetes.

Esta firma se suma a la firma del paquete RPM existente (gpgcheck), que verifica los paquetes individuales. La firma de los metadatos del repositorio verifica los metadatos que describen el contenido del repositorio, como la lista de paquetes disponibles y sus sumas de verificación.

Cómo funciona la firma de metadatos del repositorio

Cuando se publican AL2023 los repositorios, los metadatos del repositorio (`repomd.xml`) se firman con una clave AWS KMS. La firma separada resultante (`repomd.xml.asc`) se coloca junto a los metadatos en el repositorio.

Cuando lo habilitas `repo_gpgcheck` en la configuración de tu repositorio, descarga y verifica DNF automáticamente la `repomd.xml.asc` firma con la clave pública GPG antes de usar los metadatos del repositorio. Si se produce un error en la verificación de la firma, DNF rechaza los metadatos del repositorio y no continúa con las operaciones de empaquetado desde ese repositorio. Para obtener más información al respecto `repo_gpgcheck`, consulte la [Referencia DNF de configuración](#).

Los siguientes AL2023 repositorios incluyen metadatos firmados:

- Repositorio principal () `amazonlinux`
- Repositorio Kernel Livepatch () `kernel-livepatch`
- Repositorio NVIDIA () `amazonlinux-nvidia`
- Paquetes complementarios para el repositorio de Amazon Linux (`amazonlinux-spl`)

Diferencia entre **gpgcheck** y **repo_gpgcheck**

Opción	Qué verifica	Predeterminado en AL2023
<code>gpgcheck=1</code>	Verifica la firma GPG de los paquetes RPM individuales antes de la instalación.	Habilitado
<code>repo_gpgcheck=1</code>	Verifica la firma GPG de los metadatos del repositorio (<code>repomd.xml</code>) antes de usarlo.	Desactivado (activado de forma predeterminada a partir de la versión 2023.12 trimestral)

Le recomendamos encarecidamente que active las dos `gpgcheck` opciones `repo_gpgcheck`. Esto garantiza que tanto los metadatos del repositorio como los paquetes individuales se verifiquen antes de usarlos.

Habilitar la verificación de los metadatos del repositorio

Puede habilitar la verificación de los metadatos de los repositorios individuales actualizando sus archivos de configuración.

Important

A partir de la versión 2023.12 trimestral, se `repo_gpgcheck=1` activará de forma predeterminada en los archivos de configuración del AL2023 repositorio.

Habilitar para un repositorio específico

Los archivos de configuración del AL2023 repositorio están `/etc/yum.repos.d/` `repo_gpgcheck=0` configurados de forma predeterminada. Para habilitar la verificación de los metadatos del repositorio, cambie este valor a 1 en la configuración del repositorio. Por ejemplo, para habilitarlo en el repositorio principal:

```
[amazonlinux]
name=Amazon Linux 2023 repository
...
gpgcheck=1
repo_gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-amazon-linux-2023
```

Comprobar que la firma de metadatos del repositorio funciona

Tras habilitar `repo_gpgcheck=1`, puedes comprobar que la verificación de los metadatos funciona borrando la DNF caché y actualizando los metadatos:

```
[ec2-user ~]$ sudo dnf clean metadata
[ec2-user ~]$ sudo dnf makecache
```

Si la verificación de los metadatos se realiza correctamente, DNF importa la clave GPG (si aún no se ha importado) y crea la caché de metadatos sin errores. Verá una salida similar a la siguiente:

```
Amazon Linux 2023 repository          1.7 MB/s | 1.8 kB      00:00
Importing GPG key 0xD832C631:
```

```
Userid      : "Amazon Linux <amazon-linux@amazon.com>"
Fingerprint: B21C 50FA 44A9 9720 EAA7 2F7F E951 904A D832 C631
From        : /etc/pki/rpm-gpg/RPM-GPG-KEY-amazon-linux-2023
Amazon Linux 2023 repository          18 MB/s |  55 MB    00:03
Metadata cache created.
```

Si la verificación de la firma no se realiza correctamente, DNF muestra un mensaje de error que indica un error en la verificación de la firma GPG y un error al crear la caché de metadatos.

Claves públicas GPG para repositorios AL2023

Las claves públicas GPG utilizadas para la verificación de los metadatos del repositorio se instalan mediante la configuración del repositorio correspondiente en. RPMs /etc/pki/rpm-gpg/ En la siguiente tabla se enumeran las claves públicas utilizadas por cada repositorio.

Repository	Clave de firma de paquetes	Clave de firma de Repodata	Distribuido en
Núcleo (amazonlinux)	RPM-GPG-KEY-amazon-linux-2023	RPM-GPG-KEY-amazon-linux-2023	system-release
Kernel Livepatch () kernel-livepatch	RPM-GPG-KEY-amazon-linux-2023	RPM-GPG-KEY-amazon-linux-2023	system-release
NVIDIA () amazonlinux-nvidia	RPM-GPG-KEY-NVIDIA-D42D0685	RPM-GPG-KEY-amazon-linux-2023-nvidia	nvidia-release
SPAL () amazonlinux-spal	RPM-GPG-KEY-amazonlinux-spal	RPM-GPG-KEY-amazonlinux-spal	spal-release

Estas claves se instalan automáticamente al instalar el RPM de configuración del repositorio correspondiente.

UEFI Secure Boot está activado AL2023

AL2023 admite el arranque seguro UEFI a partir de la versión 2023.1. Debe usar AL2023 con instancias de Amazon EC2 que admitan UEFI y Arranque seguro UEFI. Para obtener más información, consulte [Requisitos para lanzar una instancia de Amazon EC2 en modo de arranque UEFI en](#) la Guía del usuario de Amazon EC2.

AL2023 las instancias con el arranque seguro UEFI habilitado solo aceptan el código de nivel de núcleo, incluidos el núcleo de Linux y los módulos, que estén firmados por él, de Amazon modo que puede asegurarse de que su instancia solo ejecute códigos de nivel de núcleo firmados por. AWS

Para obtener más información sobre las instancias de Amazon EC2 y el arranque seguro UEFI, consulte [Arranque seguro UEFI](#) en la Guía del usuario de Amazon EC2.

Requisitos previos

- Debe utilizar una AMI con la AL2023 versión 2023.1 o superior.
- El tipo de instancia debe ser compatible con el modo Arranque seguro UEFI. Para obtener más información, consulte [Requisitos para lanzar una instancia de Amazon EC2 en modo de arranque UEFI en](#) la Guía del usuario de Amazon EC2.

Active el arranque seguro de UEFI AL2023

AL2023 AMIs Incorpora de forma estándar un gestor de arranque y un núcleo firmados con nuestras claves. Puede habilitar el arranque seguro de UEFI inscribiendo las instancias existentes o creándolas AMIs con el arranque seguro de UEFI prehabilitado registrando una imagen a partir de una instantánea. El arranque seguro UEFI no está activado de forma predeterminada en el estándar. AL2023 AMIs

El modo de arranque AL2023 AMIs está configurado para garantizar uefi-preferred que las instancias lanzadas con ellas AMIs utilicen el firmware UEFI, si el tipo de instancia es compatible con UEFI. Si el tipo de instancia no admite UEFI, la instancia se lanza con firmware de la BIOS anterior. Cuando una instancia se ejecuta en el modo BIOS anterior, no se aplica el Arranque seguro UEFI.

Para obtener más información sobre los modos de arranque de las AMI en instancias de Amazon EC2, consulte [Comportamiento de inicialización de instancias con modos de arranque de Amazon EC2](#) en la Guía del usuario de Amazon EC2.

Temas

- [Inscripción de una instancia existente](#)
- [Registrar imagen de una instantánea](#)
- [Actualizaciones de revocación](#)
- [Cómo funciona UEFI Secure Boot en AL2023](#)
- [Inscribir sus propias claves](#)

Inscripción de una instancia existente

Para inscribir una instancia existente, rellene las variables de firmware UEFI específicas con un conjunto de claves que permitan al firmware comprobar el cargador de arranque comprobar el kernel en el siguiente arranque.

1. Amazon Linux proporciona una herramienta para simplificar el proceso de inscripción. Ejecute el siguiente comando para aprovisionar la instancia con el conjunto de claves y certificados necesario.

```
sudo amazon-linux-sb enroll
```

2. Ejecute el siguiente comando para volver a arrancar la instancia . Una vez reiniciada la instancia, se habilitará Arranque seguro UEFI.

```
sudo reboot
```

Note

Amazon Linux AMIs actualmente no es compatible con Nitro Trusted Platform Module (NitroTPM). Si necesita NitroTPM además de Arranque seguro UEFI, utilice la información de la siguiente sección.

Registrar imagen de una instantánea

Al registrar una AMI a partir de una instantánea de un volumen raíz de Amazon EBS mediante la API `register-image` de Amazon EC2, puede aprovisionar la AMI con un blob binario que contenga el estado del almacén de variables UEFI. Si lo proporciona `AL2023 UefiData`, habilita el arranque seguro de UEFI y no necesita seguir los pasos de la sección anterior.

Para obtener más información sobre la creación y el uso de un blob binario, consulte [Crear un blob binario que contenga un almacén de variables precargado](#) en la Guía del usuario de Amazon EC2.

AL2023 proporciona un blob binario prediseñado que se puede utilizar directamente en las instancias de Amazon EC2. El blob binario se encuentra en `/usr/share/amazon-linux-sb-keys/uefi.vars` en una instancia en ejecución. Este blob lo proporciona el paquete `amazon-linux-sb-keys RPM`, que se instala de forma predeterminada a AL2023 AMIs partir de la versión 2023.1.

Note

Para asegurarse de que utiliza la versión más reciente de las claves y las revocaciones, utilice el blob de la misma versión AL2023 que utilizó para crear la AMI.

Al registrar una imagen, le recomendamos que utilice el parámetro `BootMode` de la API [RegisterImage](#) establecido en `uefi`. Esto le permite activar NitroTPM configurando el parámetro `TpmSupport` en `v2.0`. Además, al configurar `BootMode` en `uefi` garantiza que el Arranque seguro UEFI esté habilitado y no se pueda deshabilitar de forma accidental al cambiar a un tipo de instancia que no sea compatible con UEFI.

Para obtener más información sobre NitroTPM, consulte [Instancias de NitroTPM para Amazon EC2](#) en la Guía del usuario de Amazon EC2.

Actualizaciones de revocación

Puede ser necesario que Amazon Linux distribuya una nueva versión del cargador de arranque `grub2` o del kernel de Linux firmada con las claves actualizadas. En ese caso, es posible que sea necesario revocar la clave anterior para evitar que los errores explotables de versiones anteriores del cargador de arranque pasen por alto el proceso de verificación de Arranque seguro UEFI.

Las actualizaciones de paquetes a los paquetes `grub2` o `kernel` siempre actualizan automáticamente la lista de revocaciones en el almacén de variables UEFI de la instancia en ejecución. Esto significa que, con el Arranque seguro UEFI habilitado, ya no se puede ejecutar la versión anterior de un paquete después de instalar una actualización de seguridad para el paquete.

Cómo funciona UEFI Secure Boot en AL2023

A diferencia de otras distribuciones de Linux, Amazon Linux no proporciona un componente adicional, denominado shim, que sirva de cargador de arranque de primera fase. El shim

generalmente está firmado con claves de Microsoft. Por ejemplo, en las distribuciones de Linux con el shim, el shim carga el cargador de arranque grub2, que utiliza el propio código del shim para verificar el kernel de Linux. Además, el shim mantiene su propio conjunto de claves y revocaciones en la base de datos de claves del propietario de la máquina (MOK), ubicada en el almacén de variables UEFI y controlada con la herramienta `mokutil`.

Amazon Linux no proporciona un shim. Como el propietario de la AMI controla las variables UEFI, este paso intermedio no es necesario y afectaría negativamente a los tiempos de inicio y arranque. Además, optamos por no confiar en las claves de ningún proveedor de forma predeterminada, para reducir la posibilidad de que se puedan ejecutar archivos binarios no deseados. Como siempre, los clientes pueden incluir archivos binarios si así lo desean.

Con Amazon Linux, UEFI carga y verifica directamente nuestro cargador de arranque grub2. El cargador de arranque grub2 se modificó para usar UEFI para verificar el kernel de Linux después de cargarlo. Por lo tanto, el kernel de Linux se verifica con los mismos certificados almacenados en la variable db UEFI normal (base de datos de claves autorizadas) y se comprueba con la misma variable dbx (base de datos de revocaciones) que el cargador de arranque y otros archivos binarios de UEFI. Al proporcionar nuestras propias claves PK y KEK, que controlan el acceso a la base de datos db y a la base de datos dbx, podemos distribuir las actualizaciones y revocaciones firmadas según sea necesario sin un intermediario como el shim.

Para obtener más información sobre el arranque seguro UEFI, consulte [Funcionamiento del arranque seguro UEFI con instancias de Amazon EC2](#) en la Guía del usuario de Amazon EC2.

Inscribir sus propias claves

Como se ha documentado en la sección anterior, Amazon Linux no requiere un shim para el Arranque seguro UEFI en Amazon EC2. Cuando lea la documentación de otras distribuciones de Linux, puede encontrar documentación para administrar la base de datos de claves de propietario de la máquina (MOK)mokutil, que no está presente en ella. AL2023 Los entornos shim y MOK solucionan algunas limitaciones de la inscripción de claves en el firmware UEFI que no se aplican a la forma en que Amazon EC2 implementa el Arranque seguro UEFI. Con Amazon EC2, existen mecanismos para manipular directamente y con facilidad las claves del almacén de variables UEFI.

Si desea inscribir sus propias claves, puede hacerlo manipulando el almacén de variables dentro de una instancia existente (consulte [Agregar claves al almacén de variables desde la instancia](#)) o creando un blob binario precargado (consulte [Crear un blob binario que contenga un](#) almacén de variables precargado).

Las traducciones son generadas a través de traducción automática. En caso de conflicto entre la traducción y la version original de inglés, prevalecerá la version en inglés.