



Panduan Administrator

Manajer Sesi Amazon DCV



Manajer Sesi Amazon DCV: Panduan Administrator

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

Merek dagang dan tampilan dagang Amazon tidak boleh digunakan sehubungan dengan produk atau layanan apa pun yang bukan milik Amazon, dengan cara apa pun yang dapat menyebabkan kebingungan di antara pelanggan, atau dengan cara apa pun yang merendahkan atau mendiskreditkan Amazon. Semua merek dagang lain yang tidak dimiliki oleh Amazon merupakan hak milik masing-masing pemiliknya, yang mungkin atau mungkin tidak terafiliasi, terkait dengan, atau disponsori oleh Amazon.

Table of Contents

Apa itu Manajer Sesi?	1
Bagaimana Manajer Sesi bekerja	1
Fitur	3
Batasan	4
Penetapan harga	4
Persyaratan	4
Persyaratan jaringan dan konektivitas	6
Siapkan Session Manager	8
Langkah 1: Siapkan server Amazon DCV	8
Langkah 2: Siapkan broker	9
Langkah 3: Siapkan agen	12
Langkah 4: Konfigurasi server Amazon DCV	19
Langkah 5: Verifikasi instalasi	21
Verifikasi agen	21
Verifikasi broker	23
Mengkonfigurasi Manajer Sesi	24
Manajer Sesi Penskalaan	24
Langkah 1: Buat profil instance	25
Langkah 2: Siapkan sertifikat SSL untuk penyeimbang beban	26
Langkah 3: Buat penyeimbang beban aplikasi Broker	27
Langkah 4: Luncurkan Broker	28
Langkah 5: Buat penyeimbang beban aplikasi Agen	29
Langkah 6: Luncurkan Agen	30
Menggunakan tag di server Amazon DCV	31
Mengkonfigurasi server otorisasi eksternal	33
Mengkonfigurasi ketekunan broker	38
Konfigurasi broker untuk bertahan di DynamoDB	39
Konfigurasi broker untuk bertahan di MariaDB/MySQL	40
Mengintegrasikan dengan Amazon DCV Connection Gateway	41
Siapkan Broker Manajer Sesi sebagai Resolver Sesi untuk Amazon DCV Connection Gateway	41
Opsional - Aktifkan otentikasi klien TLS	42
Server Amazon DCV - Referensi pemetaan DNS	44
Integrasi dengan Amazon CloudWatch	46

Memutakhirkan Manajer Sesi	48
Memutakhirkan agen Manajer Sesi Amazon DCV	48
Memutakhirkan broker Amazon DCV Session Manager	52
Referensi CLI broker	55
register-auth-server	56
Sintaksis	56
Opsi	56
Contoh	56
list-auth-servers	57
Sintaksis	56
Output	57
Contoh	56
unregister-auth-server	58
Sintaksis	56
Opsi	56
Output	57
Contoh	56
register-api-client	59
Sintaksis	56
Opsi	56
Output	57
Contoh	56
describe-api-clients	60
Sintaksis	56
Output	57
Contoh	56
unregister-api-client	62
Sintaksis	56
Opsi	56
Contoh	56
renew-auth-server-api-kunci	63
Sintaksis	56
Contoh	56
generate-software-statement	63
Sintaksis	56
Output	57

Contoh	56
describe-software-statements	65
Sintaksis	56
Output	57
Contoh	56
deactivate-software-statement	66
Sintaksis	56
Opsi	56
Contoh	56
describe-agent-clients	67
Sintaksis	56
Output	57
Contoh	56
unregister-agent-client	68
Sintaksis	56
Opsi	56
Contoh	56
register-server-dns-mappings	69
Sintaksis	56
Opsi	56
Contoh	56
describe-server-dns-mappings	70
Sintaksis	56
Output	57
Contoh	56
Referensi File Konfigurasi	72
File konfigurasi broker	72
File Konfigurasi Agen	90
Catatan Rilis dan Riwayat Dokumen	98
Catatan Rilis	98
2025.0-544— 2 Februari 2026	99
2025.0-544— 23 Desember 2025	99
2025.0-539— 12 November 2025	99
2025.0-539— 22 Oktober 2025	100
2024.0-531— 17 Juni 2025	100
2024.0-504— 31 Maret 2025	100

2024.0-493— 15 Januari 2025	101
2024.0-457— 1 Oktober 2024	101
2023.1-17652— 1 Agustus 2024	101
2023.1-16388— 26 Juni 2024	102
2023.1— November 9, 2023	102
2023.0-15065— 4 Mei 2023	102
2023.0-14852— 28 Maret 2023	103
2022.2-13907— November 11, 2022	103
2022.1-13067— 29 Juni 2022	103
2022.0-11952— 23 Februari 2022	103
2021.3-11591— Desember 20, 2021	104
2021.2-11445— November 18, 2021	104
2021.2-11190— 11 Oktober 2021	104
2021.2-11042— September 01, 2021	105
2021.1-10557— 31 Mei 2021	105
2021.0-10242— 12 April 2021	106
2020.2-9662— 04 Desember 2020	106
.....	107
Riwayat dokumen	107
.....	cxi

Apa itu Manajer Sesi Amazon DCV?

Note

Amazon DCV sebelumnya dikenal sebagai NICE DCV.

Amazon DCV Session Manager adalah kumpulan paket perangkat lunak yang dapat diinstal (Agen dan Broker) dan antarmuka pemrograman aplikasi (API) yang memudahkan pengembang dan vendor perangkat lunak independen (ISVs) untuk membangun aplikasi front-end yang secara terprogram membuat dan mengelola siklus hidup sesi Amazon DCV di seluruh armada server Amazon DCV.

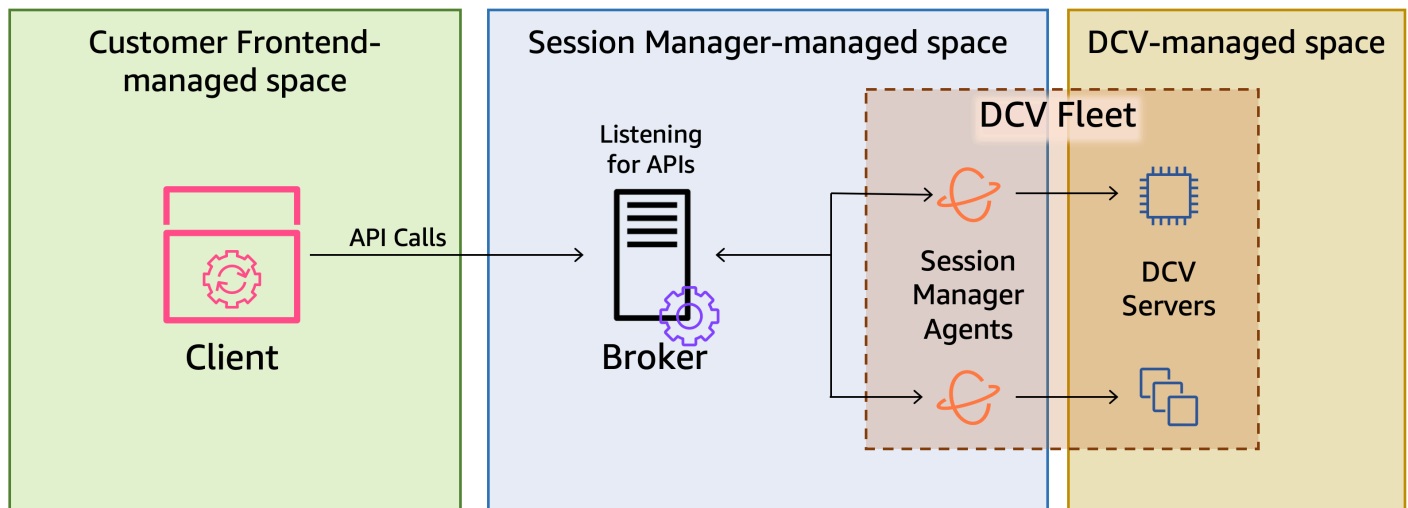
Panduan ini menjelaskan cara menginstal dan mengkonfigurasi Agen Manajer Sesi dan Broker. Untuk informasi selengkapnya tentang menggunakan Pengelola Sesi APIs, lihat Panduan Pengembang Manajer Sesi Amazon DCV.

Topik

- [Bagaimana Manajer Sesi bekerja](#)
- [Fitur](#)
- [Batasan](#)
- [Penetapan harga](#)
- [Persyaratan Manajer Sesi Amazon DCV](#)

Bagaimana Manajer Sesi bekerja

Diagram berikut menunjukkan komponen tingkat tinggi Manajer Sesi.



Pialang

Broker adalah server web yang menghosting dan mengekspos Manajer APIs Sesi. Ini menerima dan memproses permintaan API untuk mengelola sesi Amazon DCV dari klien, dan kemudian meneruskan instruksi ke Agen yang relevan. Broker harus diinstal pada host yang terpisah dari server Amazon DCV Anda, tetapi harus dapat diakses oleh klien, dan harus dapat mengakses Agen.

Agen

Agen diinstal pada setiap server Amazon DCV di armada. Agen menerima instruksi dari Broker dan menjalankannya di server Amazon DCV masing-masing. Agen juga memantau keadaan server Amazon DCV, dan mengirim pembaruan status berkala kembali ke Broker.

APIs

Session Manager mengekspos satu set antarmuka pemrograman aplikasi REST (APIs) yang dapat digunakan untuk mengelola sesi Amazon DCV pada armada server Amazon DCV. APIs ini di-host dan diekspos oleh Broker. Pengembang dapat membangun klien manajemen sesi khusus yang memanggil file APIs.

Klien

Klien adalah aplikasi front-end atau portal yang Anda kembangkan untuk memanggil Session Manager APIs yang diekspos oleh Broker. Pengguna akhir menggunakan klien untuk mengelola sesi yang dihosting di server Amazon DCV di armada.

Token akses

Untuk membuat permintaan API, Anda harus memberikan token akses. Token dapat diminta dari Broker, atau server otorisasi eksternal, oleh klien APIs terdaftar. Untuk meminta dan mengakses token, API klien harus memberikan kredensi yang valid.

API Klien

API klien dihasilkan dari file YAMM definisi API Session Manager, menggunakan Swagger Codegen. API klien digunakan untuk membuat permintaan API.

Sesi Amazon DCV

Sesi Amazon DCV adalah rentang waktu ketika server Amazon DCV dapat menerima koneksi dari klien. Sebelum klien Anda dapat terhubung ke sesi Amazon DCV, Anda harus membuat sesi Amazon DCV di server Amazon DCV. Amazon DCV mendukung sesi konsol dan virtual, dan setiap sesi memiliki pemilik dan serangkaian izin yang ditentukan. Anda menggunakan Pengelola Sesi APIs untuk mengelola siklus hidup sesi Amazon DCV. Sesi Amazon DCV dapat berada di salah satu negara bagian berikut:

- CREATING—Broker sedang dalam proses membuat sesi.
- READY—sesi siap menerima koneksi klien.
- DELETING—sesi sedang dihapus.
- DELETED—sesi telah dihapus.
- UNKNOWN—tidak dapat menentukan status sesi. Broker dan Agen mungkin tidak dapat berkomunikasi.

Fitur

Manajer Sesi DCV menawarkan fitur-fitur berikut:

- Menyediakan informasi sesi Amazon DCV —dapatkan informasi tentang sesi yang berjalan di beberapa server Amazon DCV.
- Kelola siklus hidup untuk beberapa sesi Amazon DCV —buat atau hapus beberapa sesi untuk beberapa pengguna di beberapa server Amazon DCV dengan satu permintaan API.
- Mendukung tag —gunakan tag khusus untuk menargetkan sekelompok server Amazon DCV saat membuat sesi.

- Mengelola izin untuk beberapa sesi Amazon DCV —ubah izin pengguna untuk beberapa sesi dengan satu permintaan API.
- Menyediakan informasi koneksi —mengambil informasi koneksi klien untuk sesi Amazon DCV.
- Mendukung cloud dan lokal —gunakan Pengelola Sesi di AWS, lokal, atau dengan server berbasis Internet alternatif.

Batasan

Session Manager tidak menyediakan kemampuan penyediaan sumber daya. Jika Anda menjalankan Amazon DCV di EC2 instans Amazon, Anda mungkin perlu menggunakan AWS layanan tambahan, seperti Amazon Auto EC2 Scaling untuk mengelola penskalaan infrastruktur Anda.

Penetapan harga

Session Manager tersedia tanpa biaya untuk AWS pelanggan yang menjalankan EC2 instans.

Pelanggan lokal memerlukan lisensi Amazon DCV Plus atau Amazon DCV Professional Plus. Untuk informasi tentang cara membeli lisensi Amazon DCV Plus atau Amazon DCV Professional Plus, lihat [Cara Membeli](#) di situs web Amazon DCV dan temukan distributor atau pengecer Amazon DCV di wilayah Anda. Agar semua pelanggan lokal dapat bereksperimen dengan Manajer Sesi Amazon DCV, persyaratan lisensi hanya akan diberlakukan mulai dari Amazon DCV versi 2021.0.

Untuk informasi selengkapnya, lihat [Melisensikan Server DCV Amazon di Panduan Administrator Amazon DCV](#).

Persyaratan Manajer Sesi Amazon DCV

Agen dan Broker Manajer Sesi Amazon DCV memiliki persyaratan berikut.

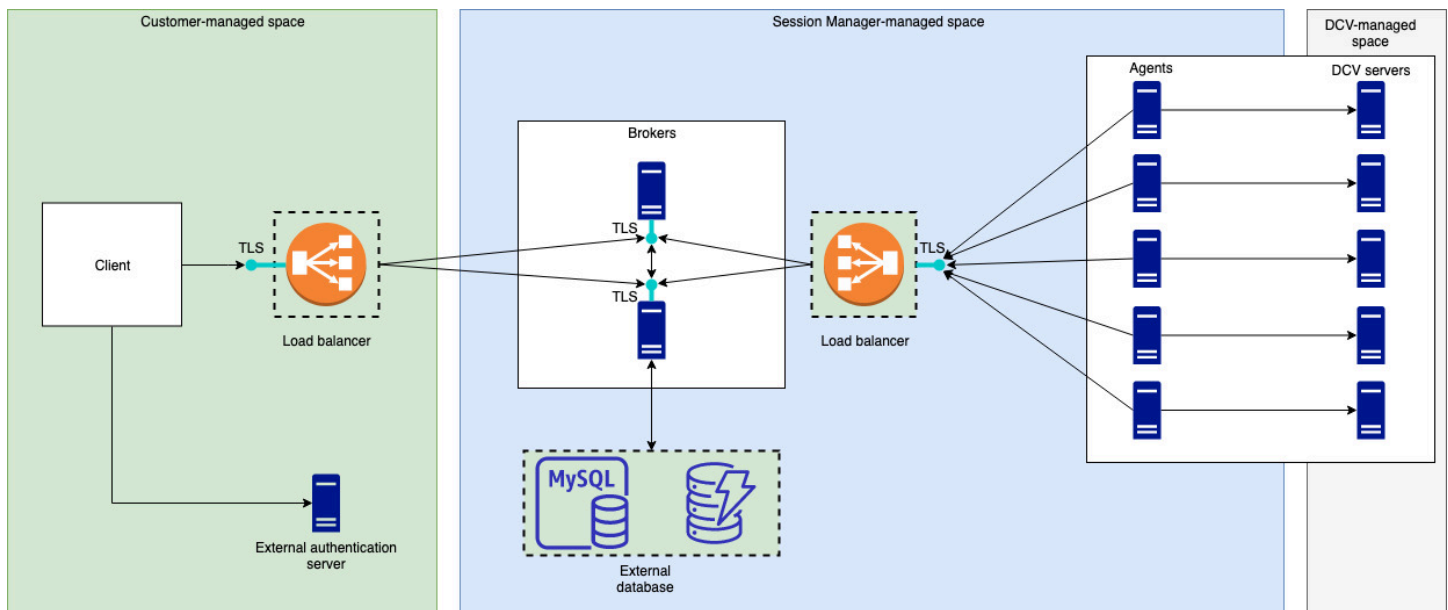
	Pialang	Agen
Sistem operasi	<ul style="list-style-type: none"> • Amazon Linux 2 • Amazon Linux 2023 • CentOS Aliran 9 • RHEL 8.x • RHEL 9.x 	<ul style="list-style-type: none"> • Windows • Windows 11 • Windows Server 2025 • Windows Server 2022

	Pialang	Agen
	<ul style="list-style-type: none"> • Rocky Linux 8.5 atau yang lebih baru • Linux berbatu 9.x • Ubuntu 22.04 • Ubuntu 24.04 	<ul style="list-style-type: none"> • Windows Server 2019 • Windows Server 2016 • Server Linux <ul style="list-style-type: none"> • Amazon Linux 2 • Amazon Linux 2023 • CentOS Aliran 9 • RHEL 8.x • RHEL 9.x • Rocky Linux 8.5 atau yang lebih baru • Linux berbatu 9.x • Ubuntu 22.04 • Ubuntu 24.04 • SUSE Linux Enterprise 15 dengan SP6 atau lebih baru • macOS <ul style="list-style-type: none"> • macOS 13 (Ventura) • macOS 14 (Sonoma) • macOS 15 (Sequoia)
Arsitektur	<ul style="list-style-type: none"> • 64-bit x86 • LENGAN 64-bit 	<ul style="list-style-type: none"> • 64-bit x86 • ARM 64-bit (Amazon Linux 2, Amazon Linux 2023, CentOS 9.x, RHEL 8.x/9.x, Rocky 8.x/9.x, Ubuntu 22.04/24.04, dan macOS 13/14/15)
Memori	8 GB	4 GB

	Pialang	Agen
Versi Amazon DCV	Amazon DCV 2020.2 dan yang lebih baru	Amazon DCV 2020.2 dan yang lebih baru
Persyaratan tambahan	Java 11	-

Persyaratan jaringan dan konektivitas

Diagram berikut memberikan gambaran tingkat tinggi dari jaringan Manajer Sesi dan persyaratan konektivitas.



Broker harus diinstal pada host terpisah, tetapi harus memiliki konektivitas jaringan dengan Agen di server Amazon DCV. Jika Anda memilih untuk memiliki beberapa Broker untuk meningkatkan ketersediaan, maka Anda harus menginstal dan mengkonfigurasi setiap broker pada host terpisah, dan menggunakan satu atau lebih penyeimbang beban untuk mengelola lalu lintas antara klien dan Broker, dan Broker dan Agen. Pialang juga harus dapat berkomunikasi satu sama lain untuk bertukar informasi tentang server dan sesi Amazon DCV. Broker dapat menyimpan kunci dan data status mereka pada database eksternal dan memiliki informasi ini tersedia setelah reboot atau penghentian. Ini membantu mengurangi risiko kehilangan informasi Broker penting dengan menyimpannya di database eksternal. Anda dapat mengambilnya nanti. Jika Anda memilih untuk memilikinya, maka

Anda harus mengatur database eksternal dan mengkonfigurasi broker. DynamoDB, MariaDB, dan MySQL didukung. Anda dapat menemukan parameter konfigurasi yang tercantum pada [File Konfigurasi Broker](#).

Agen harus dapat memulai HTTPs koneksi dua arah yang aman, persisten, dan dua arah dengan Broker.

Klien Anda, atau aplikasi frontend, harus dapat mengakses Broker untuk menelepon. APIs Klien juga harus dapat mengakses server otentikasi Anda.

Menyiapkan Manajer Sesi Amazon DCV

Bagian berikut menjelaskan cara menginstal Session Manager dengan satu broker dan beberapa agen. Anda dapat menggunakan beberapa broker untuk meningkatkan skalabilitas dan kinerja. Untuk informasi selengkapnya, lihat [Manajer Sesi Penskalaan](#).

Untuk menyiapkan Amazon DCV Session Manager, lakukan hal berikut:

Langkah-langkah

- [Langkah 1: Siapkan server Amazon DCV](#)
- [Langkah 2: Siapkan broker Amazon DCV Session Manager](#)
- [Langkah 3: Siapkan agen Manajer Sesi Amazon DCV](#)
- [Langkah 4: Konfigurasi server Amazon DCV untuk menggunakan broker sebagai server otentikasi](#)
- [Langkah 5: Verifikasi instalasi](#)

Langkah 1: Siapkan server Amazon DCV

Anda harus memiliki armada server Amazon DCV yang ingin Anda gunakan Session Manager. Untuk informasi selengkapnya tentang menginstal server Amazon DCV, lihat [Menginstal server Amazon DCV di Panduan Administrator Amazon DCV](#).

Di server Linux dan macOS Amazon DCV, Session Manager menggunakan pengguna layanan lokal bernama `dcvsmagent`. Pengguna ini secara otomatis dibuat ketika agen Session Manager diinstal. Anda harus memberikan hak administrator pengguna layanan ini untuk Amazon DCV sehingga dapat melakukan tindakan atas nama pengguna lain. Untuk memberikan hak administrator pengguna layanan Session Manager, lakukan hal berikut:

Untuk menambahkan pengguna layanan lokal untuk server Linux dan macOS Amazon DCV

1. Buka `/etc/dcv/dcv.conf` menggunakan editor teks pilihan Anda.
2. Tambahkan `administrators` parameter ke `[security]` bagian dan tentukan pengguna Session Manager. Contoh:

```
[security]
```

```
administrators=["dcvsmagent"]
```

3. Simpan dan tutup file .
4. Hentikan dan mulai ulang server Amazon DCV.

Session Manager hanya dapat membuat sesi Amazon DCV atas nama pengguna yang sudah ada di server Amazon DCV. Jika permintaan dibuat untuk membuat sesi untuk pengguna yang tidak ada, permintaan gagal. Oleh karena itu, Anda harus memastikan bahwa setiap pengguna akhir yang dituju memiliki pengguna sistem yang valid di server Amazon DCV.

Tip

Jika Anda bermaksud menggunakan beberapa host broker atau server Amazon DCV dengan agen, kami sarankan Anda mengonfigurasi hanya satu broker dan satu server Amazon DCV dengan agen dengan melakukan langkah-langkah berikut, membuat Amazon Machine Images (AMI) dari host dengan konfigurasi lengkap, dan kemudian menggunakan AMIs untuk meluncurkan broker yang tersisa dan server Amazon DCV. Atau, Anda dapat menggunakan AWS Systems Manager untuk menjalankan perintah pada beberapa instance dari jarak jauh.

Langkah 2: Siapkan broker Amazon DCV Session Manager

Broker harus diinstal pada host Linux. Untuk informasi selengkapnya tentang distribusi Linux yang didukung, lihat [Persyaratan Manajer Sesi Amazon DCV](#). Instal broker pada host yang terpisah dari agen dan host server Amazon DCV. Host dapat diinstal pada jaringan pribadi yang berbeda, tetapi harus dapat terhubung dan berkomunikasi dengan agen.

Untuk menginstal dan memulai broker

1. Connect ke host tempat Anda ingin menginstal broker.
2. Paket-paket tersebut ditandatangani secara digital dengan tanda tangan GPG yang aman. Untuk mengizinkan pengelola paket memverifikasi tanda tangan paket, Anda harus mengimpor kunci Amazon DCV GPG. Jalankan perintah berikut untuk mengimpor kunci Amazon DCV GPG.
 - Amazon Linux 2, RHEL, CentOS, dan Rocky Linux

```
$ sudo rpm --import https://d1uj6qtbmh3dt5.cloudfront.net/NICE-GPG-KEY
```

- Ubuntu

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/NICE-GPG-KEY
```

```
$ gpg --import NICE-GPG-KEY
```

3. Unduh paket instalasi.

- Amazon Linux 2

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2025.0/SessionManagerBrokers/nice-dcv-session-manager-broker-2025.0.539-1.el7.noarch.rpm
```

- Amazon Linux 2023

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2025.0/SessionManagerBrokers/nice-dcv-session-manager-broker-2025.0.539-1.amzn2023.noarch.rpm
```

- RHEL 8.x, dan Rocky Linux 8.x

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2025.0/SessionManagerBrokers/nice-dcv-session-manager-broker-2025.0.539-1.el8.noarch.rpm
```

- CentOS 9.x, RHEL 9.x, dan Rocky Linux 9.x

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2025.0/SessionManagerBrokers/nice-dcv-session-manager-broker-2025.0.539-1.el9.noarch.rpm
```

- Ubuntu 20.04

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2025.0/SessionManagerBrokers/nice-dcv-session-manager-broker_2025.0.539-1_all.ubuntu2004.deb
```

- Ubuntu 22.04

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2025.0/SessionManagerBrokers/nice-dcv-session-manager-broker_2025.0.539-1_all.ubuntu2204.deb
```

- Ubuntu 24.04

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2025.0/SessionManagerBrokers/nice-dcv-session-manager-broker_2025.0.539-1_all.ubuntu2404.deb
```

4. Instal paket .

- Amazon Linux 2

```
$ sudo yum install -y ./nice-dcv-session-manager-broker-2025.0.539-1.el7.noarch.rpm
```

- Amazon Linux 2023

```
$ sudo yum install -y ./nice-dcv-session-manager-broker-2025.0.539-1.amzn2023.noarch.rpm
```

- RHEL 8.x dan Rocky Linux 8.x

```
$ sudo yum install -y ./nice-dcv-session-manager-broker-2025.0.539-1.el8.noarch.rpm
```

- CentOS 9.x, RHEL 9.x, dan Rocky Linux 9.x

```
$ sudo yum install -y ./nice-dcv-session-manager-broker-2025.0.539-1.el9.noarch.rpm
```

- Ubuntu 20.04

```
$ sudo apt install -y ./nice-dcv-session-manager-broker_2025.0.539-1_all.ubuntu2004.deb
```

- Ubuntu 22.04

```
$ sudo apt install -y ./nice-dcv-session-manager-broker_2025.0.539-1_all.ubuntu2204.deb
```

- Ubuntu 24.04

```
$ sudo apt install -y ./nice-dcv-session-manager-broker_2025.0.539-1_all.ubuntu2404.deb
```

5. Periksa apakah versi lingkungan Java default adalah 11

```
$ java -version
```

Jika tidak, Anda dapat secara eksplisit mengatur direktori home Java yang akan digunakan broker untuk menargetkan versi Java yang tepat. Ini dilakukan pengaturan parameter `broker-java-home` dalam file konfigurasi broker. Untuk informasi selengkapnya, lihat [File Konfigurasi broker](#).

6. Mulai layanan broker dan pastikan bahwa itu dimulai secara otomatis setiap kali instance dimulai.

```
$ sudo systemctl start dcv-session-manager-broker && sudo systemctl enable dcv-session-manager-broker
```

7. Tempatkan salinan sertifikat yang ditandatangani sendiri broker di direktori pengguna Anda. Anda akan membutuhkannya ketika Anda menginstal agen di langkah berikutnya.

```
sudo cp /var/lib/dcvsmbroker/security/dcvsmbroker_ca.pem $HOME
```

Langkah 3: Siapkan agen Manajer Sesi Amazon DCV

Agan harus diinstal pada semua host server Amazon DCV di armada. Agan dapat diinstal pada host Windows, Linux dan macOS. Untuk informasi selengkapnya tentang sistem operasi yang didukung, lihat [Persyaratan Manajer Sesi Amazon DCV](#).

Prasyarat

Server Amazon DCV harus diinstal pada host sebelum menginstal agen.

Linux host

Note

[Agen Session Manager tersedia untuk distribusi dan arsitektur Linux yang tercantum dalam Persyaratan:](#)

Instruksi berikut adalah untuk menginstal agen pada host 64-bit x86. Untuk menginstal agen pada host ARM 64-bit ganti `x86_64` dengan `arch64`. Untuk Ubuntu, ganti `amd64` dengan `arm64`.

Untuk menginstal agen pada host Linux

1. Paket-paket tersebut ditandatangani secara digital dengan tanda tangan GPG yang aman. Untuk mengizinkan pengelola paket memverifikasi tanda tangan paket, Anda harus mengimpor kunci Amazon DCV GPG. Jalankan perintah berikut untuk mengimpor kunci Amazon DCV GPG.

- Amazon Linux 2, RHEL, CentOS, dan SUSE Linux Enterprise

```
$ sudo rpm --import https://d1uj6qtbmh3dt5.cloudfront.net/NICE-GPG-KEY
```

- Ubuntu

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/NICE-GPG-KEY
```

```
$ gpg --import NICE-GPG-KEY
```

2. Unduh paket instalasi.

- Amazon Linux 2

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2025.0/SessionManagerAgents/nice-dcv-session-manager-agent-2025.0.801-1.el7.x86_64.rpm
```

- Amazon Linux 2023

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2025.0/SessionManagerAgents/nice-dcv-session-manager-agent-2025.0.888-1.amzn2023.x86_64.rpm
```

- RHEL 8.x dan Rocky Linux 8.x

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2025.0/SessionManagerAgents/nice-dcv-session-manager-agent-2025.0.888-1.el8.x86_64.rpm
```

- CentOS 9.x, RHEL 9.x, dan Rocky Linux 9.x

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2025.0/SessionManagerAgents/nice-dcv-session-manager-agent-2025.0.888-1.el9.x86_64.rpm
```

- Ubuntu 20.04

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2025.0/SessionManagerAgents/nice-dcv-session-manager-agent_2025.0.888-1_amd64.ubuntu2004.deb
```

- Ubuntu 22.04

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2025.0/SessionManagerAgents/nice-dcv-session-manager-agent_2025.0.888-1_amd64.ubuntu2204.deb
```

- Ubuntu 24.04

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2025.0/SessionManagerAgents/nice-dcv-session-manager-agent_2025.0.888-1_amd64.ubuntu2404.deb
```

- SUSE Linux Perusahaan 12

```
$ curl -O https://d1uj6qtbmh3dt5.cloudfront.net/2025.0/SessionManagerAgents/nice-dcv-session-manager-agent-2025.0.888-1.sles12.x86_64.rpm
```

- SUSE Linux Perusahaan 15

```
$ curl -O https://d1uj6qtbmh3dt5.cloudfront.net/2025.0/SessionManagerAgents/nice-dcv-session-manager-agent-2025.0.888-1.sles15.x86_64.rpm
```

3. Instal paket .

- Amazon Linux 2

```
$ sudo yum install -y ./nice-dcv-session-manager-agent-2025.0.888-1.el7.x86_64.rpm
```

- Amazon Linux 2023

```
$ sudo yum install -y ./nice-dcv-session-manager-agent-2025.0.888-1.amzn2023.x86_64.rpm
```

- RHEL 8.x dan Rocky Linux 8.x

```
$ sudo yum install -y ./nice-dcv-session-manager-agent-2025.0.888-1.el8.x86_64.rpm
```

- CentOS 9.x, RHEL 9.x, dan Rocky Linux 9.x

```
$ sudo yum install -y ./nice-dcv-session-manager-agent-2025.0.888-1.el9.x86_64.rpm
```

- Ubuntu 20.04

```
$ sudo apt install ./nice-dcv-session-manager-agent_2025.0.888-1_amd64.ubuntu2004.deb
```

- Ubuntu 22.04

```
$ sudo apt install ./nice-dcv-session-manager-agent_2025.0.888-1_amd64.ubuntu2204.deb
```

- Ubuntu 24.04

```
$ sudo apt install ./nice-dcv-session-manager-agent_2025.0.888-1_amd64.ubuntu2404.deb
```


- SUSE Linux Perusahaan 12

```
$ sudo zypper install ./nice-dcv-session-manager-agent-2025.0.888-1.sles12.x86_64.rpm
```

- SUSE Linux Perusahaan 15

```
$ sudo zypper install ./nice-dcv-session-manager-agent-2025.0.888-1.sles15.x86_64.rpm
```

4. Tempatkan salinan sertifikat yang ditandatangani sendiri broker (yang Anda salin pada langkah sebelumnya) di `/etc/dcv-session-manager-agent/` direktori pada agen.
5. Buka `/etc/dcv-session-manager-agent/agent.conf` menggunakan editor teks pilihan Anda dan lakukan hal berikut.
 - Untuk `broker_host`, tentukan nama DNS host tempat broker diinstal.

 Important

Jika broker berjalan pada EC2 instans Amazon, untuk `broker_host` Anda harus menentukan alamat Ipv4 pribadi instans.

- (Opsional) Untuk `broker_port`, tentukan port untuk berkomunikasi dengan broker. Secara default agen dan broker berkomunikasi melalui port 8445. Hanya ubah ini jika Anda perlu menggunakan port yang berbeda. Jika Anda mengubahnya, pastikan broker dikonfigurasi untuk menggunakan port yang sama.
- Untuk `ca_file`, tentukan jalur lengkap file sertifikat yang Anda salin di langkah sebelumnya. Contoh:

```
ca_file = '/etc/dcv-session-manager-agent/broker_cert.pem'
```

Atau, jika Anda ingin menonaktifkan verifikasi TLS, atur `tls_strict` ke `false`.

6. Simpan dan tutup file .
7. Jalankan perintah berikut untuk memulai agen.

```
$ sudo systemctl start dcv-session-manager-agent
```

Windows host

Untuk menginstal agen pada host Windows

1. Unduh [penginstal agen](#).
2. Jalankan pemasang. Pada layar Selamat Datang, pilih Berikutnya.
3. Pada layar EULA, baca perjanjian lisensi dengan cermat, dan jika Anda setuju, pilih Saya menerima persyaratan dan pilih Berikutnya.
4. Untuk memulai instalasi, pilih Instal.
5. Tempatkan salinan sertifikat yang ditandatangani sendiri broker (yang Anda salin pada langkah sebelumnya) di `C:\Program Files\NICE\DCVSessionManagerAgent\conf\` folder pada agen.
6. Buka `C:\Program Files\NICE\DCVSessionManagerAgent\conf\agent.conf` menggunakan editor teks pilihan Anda, lalu lakukan hal berikut:
 - Untuk `broker_host`, tentukan nama DNS host tempat broker diinstal.

⚠ Important

Jika broker berjalan pada EC2 instans Amazon, untuk `broker_host` Anda harus menentukan IPv4 alamat pribadi instans.

- (Opsional) Untuk `broker_port`, tentukan port untuk berkomunikasi dengan broker. Secara default agen dan broker berkomunikasi melalui port 8445. Hanya ubah ini jika Anda perlu menggunakan port yang berbeda. Jika Anda mengubahnya, pastikan broker dikonfigurasi untuk menggunakan port yang sama.
- Untuk `ca_file`, tentukan jalur lengkap file sertifikat yang Anda salin di langkah sebelumnya. Contoh:

```
ca_file = 'C:\Program Files\NICE\DCVSessionManagerAgent\conf\broker_cert.pem'
```

Atau, jika Anda ingin menonaktifkan verifikasi TLS, atur `tls_strict` ke `false`.

7. Simpan dan tutup file .
8. Hentikan dan mulai ulang layanan agen agar perubahan diterapkan. Jalankan perintah berikut di command prompt.

```
C:\> sc stop DcvSessionManagerAgentService
```

```
C:\> sc start DcvSessionManagerAgentService
```

macOS host

Untuk menginstal agen pada host macOS


1. Unduh [penginstal agen](#) atau gunakan baris perintah:

```
$ curl -O https://d1uj6qtbmh3dt5.cloudfront.net/2025.0/SessionManagerAgents/nice-dcv-session-manager-agent-2025.0.888-macos-arm64.pkg
```

2. Instal paket . Anda dapat menggunakan baris perintah atau melalui penginstal agen:
 - Baris perintah:

```
$ sudo installer -pkg ./nice-dcv-session-manager-agent-2025.0.888-macos-arm64.pkg -target /
```

- Installer agen: Klik dua kali .pkg file dan ikuti panduan instalasi.
3. Tempatkan salinan sertifikat yang ditandatangani sendiri broker (yang Anda salin pada langkah sebelumnya) di `/etc/dcv-session-manager-agent/` direktori pada agen.
 4. Buka `/etc/dcv-session-manager-agent/agent.conf` menggunakan editor teks pilihan Anda dan lakukan hal berikut.
 - Untuk `broker_host`, tentukan nama DNS host tempat broker diinstal.

 Important

Jika broker berjalan pada EC2 instans Amazon, untuk `broker_host` Anda harus menentukan IPv4 alamat pribadi instans.

- (Opsional) Untuk `broker_port`, tentukan port untuk berkomunikasi dengan broker. Secara default agen dan broker berkomunikasi melalui port 8445. Hanya ubah ini jika Anda perlu menggunakan port yang berbeda. Jika Anda mengubahnya, pastikan broker dikonfigurasi untuk menggunakan port yang sama.
- Untuk `ca_file`, tentukan jalur lengkap file sertifikat yang Anda salin di langkah sebelumnya. Contoh:

```
ca_file = '/usr/local/etc/dcv-session-manager-agent/broker_cert.pem'
```

Atau, jika Anda ingin menonaktifkan verifikasi TLS, atur `tls_strict` ke `false`.

5. Simpan dan tutup file .
6. Mulai layanan agen.

```
$ sudo launchctl load /Library/LaunchDaemons/com.amazon.dcv.session-manager.agent.plist
```

Langkah 4: Konfigurasi server Amazon DCV untuk menggunakan broker sebagai server otentikasi

Konfigurasi server Amazon DCV untuk menggunakan broker sebagai server otentikasi eksternal untuk memvalidasi token koneksi klien. Anda juga harus mengonfigurasi server Amazon DCV untuk mempercayai CA yang ditandatangani sendiri oleh broker.

Linux Amazon DCV server

Untuk menambahkan pengguna layanan lokal untuk server Linux Amazon DCV

1. Buka `/etc/dcv/dcv.conf` menggunakan editor teks pilihan Anda.
2. Tambahkan `auth-token-verifier` parameter `ca-file` dan ke `[security]` bagian.
 - Untuk `ca-file`, tentukan jalur ke CA yang ditandatangani sendiri oleh broker yang Anda salin ke host pada langkah sebelumnya.
 - Untuk `auth-token-verifier`, tentukan URL untuk verifikasi token pada broker dalam format berikut: `https://broker_ip_or_dns:port/agent/validate-authentication-token`. Tentukan port yang digunakan untuk komunikasi broker-agent, yaitu 8445 secara default. Jika Anda menjalankan broker pada EC2 instans Amazon, Anda harus menggunakan DNS pribadi atau alamat IP pribadi.

Sebagai contoh

```
[security]
ca-file="/etc/dcv-session-manager-agent/broker_cert.pem"
auth-token-verifier="https://my-sm-broker.com:8445/agent/validate-authentication-token"
```

3. Simpan dan tutup file .
4. Hentikan dan mulai ulang server Amazon DCV. Untuk informasi selengkapnya, lihat [Menghentikan Server DCV Amazon](#) dan [Memulai Server DCV Amazon di Panduan Administrator Amazon DCV](#).

Windows Amazon DCV server

Di server Windows Amazon DCV

1. Buka Windows Registry Editor dan arahkan ke tombol HKEY_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv/security/.
2. Buka parameter ca-file.
3. Untuk data Nilai, tentukan jalur ke CA yang ditandatangani sendiri oleh broker yang Anda salin ke host pada langkah sebelumnya.

Note

Jika parameter tidak ada, buat parameter string baru dan beri nama ca-file.

4. Buka auth-token-verifierparameter-nya.
5. Untuk data Nilai, tentukan URL untuk verifikasi token pada broker dalam format berikut: `https://broker_ip_or_dns:port/agent/validate-authentication-token`.
6. Tentukan port yang digunakan untuk komunikasi broker-agent, yaitu 8445 secara default. Jika Anda menjalankan broker pada EC2 instans Amazon, Anda harus menggunakan DNS pribadi atau alamat IP pribadi.

Note

Jika parameter tidak ada, buat parameter string baru dan beri nama auth-token-verifier.

7. Pilih OK dan tutup Windows Registry Editor.
8. Hentikan dan mulai ulang server Amazon DCV. Untuk informasi selengkapnya, lihat [Menghentikan Server DCV Amazon](#) dan [Memulai Server DCV Amazon di Panduan Administrator Amazon DCV](#).

macOS Amazon DCV server

Untuk menambahkan pengguna layanan lokal untuk server macOS Amazon DCV

1. Buka `/etc/dcv/dcv.conf` menggunakan editor teks pilihan Anda.

2. Tambahkan `auth-token-verifier` parameter `ca-file` dan ke `[security]` bagian.
 - Untuk `ca-file`, tentukan jalur ke CA yang ditandatangani sendiri oleh broker yang Anda salin ke host pada langkah sebelumnya.
 - Untuk `auth-token-verifier`, tentukan URL untuk verifikasi token pada broker dalam format berikut: `https://broker_ip_or_dns:port/agent/validate-authentication-token`. Tentukan port yang digunakan untuk komunikasi broker-agent, yaitu 8445 secara default. Jika Anda menjalankan broker pada EC2 instans Amazon, Anda harus menggunakan DNS pribadi atau alamat IP pribadi.

Sebagai contoh

```
[security]
ca-file="/usr/local/etc/dcv-session-manager-agent/broker_cert.pem"
auth-token-verifier="https://my-sm-broker.com:8445/agent/validate-authentication-token"
```

3. Simpan dan tutup file .
4. Hentikan dan mulai ulang server Amazon DCV. Untuk informasi selengkapnya, lihat [Menghentikan Server DCV Amazon](#) dan [Memulai Server DCV Amazon di Panduan Administrator Amazon DCV](#).

Langkah 5: Verifikasi instalasi

Setelah Anda mengatur agen, mengatur broker, dan mengonfigurasi keduanya di server Amazon DCV, Anda perlu memverifikasi bahwa instalasi berfungsi dengan baik.

Topik

- [Verifikasi agen](#)
- [Verifikasi broker](#)

Verifikasi agen

Setelah Anda menginstal broker dan agen, pastikan agen berjalan dan dapat terhubung ke broker.

Host agen Linux dan macOS

Perintah untuk menjalankan tergantung pada versinya.

- Sejak versi 2022.0

Dari host agen, jalankan perintah berikut:

```
$ grep 'sessionsUpdateResponse' /var/log/dcv-session-manager-agent/agent.log | tail -1 | grep -o success
```

- Versi sebelum 2022.0

Dari host agen, jalankan perintah berikut, dan tentukan tahun, bulan, dan hari saat ini.

```
$ grep 'sessionsUpdateResponse' /var/log/dcv-session-manager-agent/agent.log.yyyy-mm-dd | tail -1 | grep -o success
```

Sebagai contoh

```
$ grep 'sessionsUpdateResponse' /var/log/dcv-session-manager-agent/agent.log.2020-11-19 | tail -1 | grep -o success
```

Jika agen berjalan dan dapat terhubung ke broker, perintah harus kembali success.

Jika perintah mengembalikan output yang berbeda, periksa file log agen untuk informasi lebih lanjut. File log terletak di sini: `/var/log/dcv-session-manager-agent/`.

Host agen Windows

Buka file log agen, yang terletak di `C:\ProgramData\NICE\DCVSessionManagerAgent\log`.

Jika file log menyertakan baris yang mirip dengan yang di bawah ini, agen berjalan dan dapat terhubung ke broker.

```
2020-11-02 12:38:03,996919 INFO ThreadId(05) dcvsessionmanageragent::agent:Processing broker message "{\n  \"sessionsUpdateResponse\" : {\n    \"requestId\" : \"69c24a3f5f6d4f6f83ffbb9f7dc6a3f4\", \n    \"result\" : {\n      \"success\" : true\n    }\n  }\n}"
```

Jika file log Anda tidak memiliki baris yang sama, periksa file log untuk kesalahan.

Verifikasi broker

Setelah Anda menginstal broker dan agen, pastikan broker Anda berjalan dan dapat dijangkau dari pengguna dan aplikasi front-end Anda.

Dari komputer yang seharusnya bisa menjangkau broker, jalankan perintah berikut:

```
$ curl -X GET https://broker_host_ip:port/sessionConnectionData/aSession/aOwner --insecure
```

Jika verifikasi berhasil, broker mengembalikan yang berikut:

```
{  
  "error": "No authorization header"  
}
```

Mengkonfigurasi Manajer Sesi Amazon DCV

Untuk memberikan pengalaman yang mulus dan aman, penting untuk mengonfigurasi Manajer Sesi dengan benar sesuai dengan kebutuhan dan persyaratan organisasi Anda. Bagian ini memandu Anda melalui langkah-langkah kunci yang terlibat dalam menyiapkan dan mengonfigurasi Manajer Sesi, termasuk mengelola akses pengguna, mengonfigurasi pengaturan jaringan, dan menyesuaikan pengaturan sesi.

Topik

- [Manajer Sesi Penskalaan](#)
- [Menggunakan tag untuk menargetkan server Amazon DCV](#)
- [Mengkonfigurasi server otorisasi eksternal](#)
- [Mengkonfigurasi ketekunan broker](#)
- [Mengintegrasikan dengan Amazon DCV Connection Gateway](#)
- [Integrasi dengan Amazon CloudWatch](#)

Manajer Sesi Penskalaan

Untuk mengaktifkan ketersediaan tinggi dan meningkatkan kinerja, Anda dapat mengonfigurasi Session Manager untuk menggunakan beberapa Agen dan Broker. Jika Anda berniat menggunakan beberapa Agen dan Broker, kami sarankan Anda menginstal dan mengonfigurasi hanya satu host Agen dan Broker, membuat Gambar Mesin Amazon (AMI) dari host tersebut, dan kemudian meluncurkan host yang tersisa dari AMIs.

Secara default, Session Manager mendukung penggunaan beberapa Agen tanpa konfigurasi tambahan. Namun, jika Anda berniat menggunakan beberapa Broker, Anda harus menggunakan penyeimbang beban untuk menyeimbangkan lalu lintas antara klien frontend dan Broker, dan antara Broker dan Agen. Pengaturan dan konfigurasi penyeimbang beban sepenuhnya dimiliki dan dikelola oleh Anda.

Bagian berikut menjelaskan cara mengkonfigurasi Session Manager untuk menggunakan beberapa host dengan Application Load Balancer.

Langkah-langkah

- [Langkah 1: Buat profil instance](#)

- [Langkah 2: Siapkan sertifikat SSL untuk penyeimbang beban](#)
- [Langkah 3: Buat penyeimbang beban aplikasi Broker](#)
- [Langkah 4: Luncurkan Broker](#)
- [Langkah 5: Buat penyeimbang beban aplikasi Agen](#)
- [Langkah 6: Luncurkan Agen](#)

Langkah 1: Buat profil instance

Anda harus melampirkan profil instans ke host Broker dan Agen yang memberi mereka izin untuk menggunakan Elastic Load Balancing APIs. Untuk informasi selengkapnya, lihat [peran IAM untuk Amazon EC2](#) di Panduan Pengguna Amazon EC2.

Buat profil instans

1. Buat peran AWS Identity and Access Management (IAM) yang mendefinisikan izin untuk digunakan dalam profil instance. Gunakan kebijakan kepercayaan berikut:

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "ec2.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

Kemudian lampirkan kebijakan berikut:

JSON

```
{
```

```

"Version": "2012-10-17",
"Statement": [
  {
    "Action": [
      "ec2:DescribeInstances"
    ],
    "Effect": "Allow",
    "Resource": "*"
  },
  {
    "Action": [
      "elasticloadbalancing:DescribeTargetHealth"
    ],
    "Effect": "Allow",
    "Resource": "*"
  }
]
}

```

Untuk informasi selengkapnya, lihat [Membuat peran IAM](#) di Panduan Pengguna IAM.

2. Buat profil instance baru. Untuk informasi selengkapnya, lihat [create-instance-profile](#) dalam AWS CLI Referensi Perintah.
3. Tambahkan peran IAM ke profil instance. Untuk informasi selengkapnya, lihat [add-role-to-instance-profile](#) di Referensi AWS CLI Perintah.
4. Lampirkan profil instance ke host Broker. Untuk informasi selengkapnya, lihat [Melampirkan peran IAM ke instans](#) di Panduan Pengguna Amazon EC2.

Langkah 2: Siapkan sertifikat SSL untuk penyeimbang beban

Saat Anda menggunakan HTTPS untuk listener penyeimbang beban, Anda harus men-deploy sertifikat SSL pada penyeimbang beban. Penyeimbang beban menggunakan sertifikat untuk mengakhiri koneksi dan mendekripsi permintaan dari klien sebelum mengirimkannya mereka ke target.

Untuk menyiapkan sertifikat SSL

1. Buat otoritas sertifikat pribadi (CA) AWS Certificate Manager Private Certificate Authority (ACM PCA). Untuk informasi selengkapnya, lihat [Prosedur untuk Membuat CA](#) di Panduan Pengguna Otoritas AWS Certificate Manager Private Certificate Authority.

2. Instal CA. Untuk informasi selengkapnya, lihat [Menginstal Sertifikat Root CA](#) di Panduan Pengguna Otoritas AWS Sertifikat Pribadi Certificate Manager Certificate.
3. Minta sertifikat pribadi baru yang ditandatangani oleh CA. Untuk nama domain, gunakan *.*region*.elb.amazonaws.com dan tentukan Wilayah tempat Anda ingin membuat penyeimbang beban. Untuk informasi selengkapnya, lihat [Meminta Sertifikat Pribadi](#) di Panduan Pengguna Otoritas AWS Sertifikat Pribadi Certificate Manager Certificate.

Langkah 3: Buat penyeimbang beban aplikasi Broker

Buat penyeimbang beban aplikasi untuk menyeimbangkan lalu lintas antara klien front-end Anda dan Broker.

Untuk membuat penyeimbang beban

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>

Di panel navigasi, pilih Load Balancers dan kemudian pilih Create Load Balancer. Untuk tipe load balancer, pilih Application Load Balancer.

2. Untuk Langkah 1: Konfigurasi Penyeimbang Beban, lakukan hal berikut:
 - a. Untuk Nama, masukkan nama deskriptif untuk penyeimbang beban.
 - b. Untuk Skema, pilih yang menghadap ke internet.
 - c. Untuk Protokol Load Balancer, pilih HTTPS, dan untuk Port Load Balancer, masukkan 8443
 - d. Untuk VPC, pilih VPC yang akan digunakan dan kemudian pilih semua subnet di VPC itu.
 - e. Pilih Berikutnya.
3. Untuk Langkah 2: Konfigurasikan Pengaturan Keamanan, lakukan hal berikut:
 - a. Untuk jenis Sertifikat, pilih Pilih sertifikat dari ACM.
 - b. Untuk nama Sertifikat, pilih sertifikat pribadi yang Anda minta sebelumnya.
 - c. Pilih Berikutnya.
4. Untuk Langkah 3: Konfigurasikan Grup Keamanan, buat grup keamanan baru, atau pilih grup keamanan yang ada yang memungkinkan lalu lintas masuk dan keluar antara klien frontend Anda dan Broker melalui HTTPS dan port 8443.

Pilih Berikutnya.

5. Untuk Langkah 4: Konfigurasi Routing, lakukan hal berikut:
 - a. Untuk grup Target, pilih Grup target baru.
 - b. Untuk Name, masukkan nama untuk grup target.
 - c. Untuk tipe Target, pilih Instance.
 - d. Untuk Protokol, pilih HTTPS. Untuk Port, masukkan 8443. Untuk versi Protokol, pilih HTTP1.
 - e. Untuk pemeriksaan kesehatan Protokol, pilih HTTPS, dan untuk Path, masukkan /health.
 - f. Pilih Berikutnya.
6. Untuk Langkah 5: Daftarkan Target, pilih Berikutnya.
7. Pilih Buat.

Langkah 4: Luncurkan Broker

Buat Broker awal dan konfigurasi untuk menggunakan penyeimbang beban, buat AMI dari Broker, dan kemudian gunakan AMI untuk meluncurkan Broker yang tersisa. Ini memastikan bahwa semua Broker dikonfigurasi untuk menggunakan CA yang sama dan konfigurasi penyeimbang beban yang sama.

Untuk meluncurkan Broker

1. Luncurkan dan konfigurasi host Broker awal. Untuk informasi selengkapnya tentang menginstal dan mengonfigurasi Broker, lihat [Langkah 2: Siapkan broker Amazon DCV Session Manager](#).

Note

Sertifikat yang ditandatangani sendiri oleh broker tidak diperlukan karena kami menggunakan penyeimbang beban aplikasi.

2. Hubungkan ke Broker, buka `/etc/dcv-session-manager-broker/session-manager-broker.properties` menggunakan editor teks pilihan Anda, dan lakukan hal berikut:
 - a. Komentari `broker-to-broker-discovery-addresses` parameter dengan menempatkan hash (`#`) di awal baris.

- b. Untuk `broker-to-broker-discovery-aws-region`, masukkan Wilayah tempat Anda membuat penyeimbang beban aplikasi.
 - c. Untuk `broker-to-broker-discovery-aws-alb-target-group-arn`, masukkan ARN dari kelompok sasaran yang terkait dengan penyeimbang beban Broker.
 - d. Simpan dan tutup file .
3. Hentikan contoh Broker.
 4. Buat AMI dari instance Broker yang dihentikan. Untuk informasi selengkapnya, lihat [Membuat AMI Linux dari instans](#) di Panduan Pengguna Amazon EC2 untuk Instans Linux.
 5. Gunakan AMI untuk meluncurkan Broker yang tersisa.
 6. Tetapkan profil instans yang Anda buat untuk semua instance Broker.
 7. Tetapkan grup keamanan yang memungkinkan Broker to Broker dan Broker memuat lalu lintas jaringan penyeimbang ke semua instance Broker. Untuk informasi selengkapnya tentang port jaringan, lihat [File Konfigurasi Broker](#).
 8. Daftarkan semua instance Broker sebagai target untuk load balancer Broker. Untuk informasi selengkapnya, lihat [Mendaftarkan target dengan grup target Anda](#) di Panduan Pengguna untuk Penyeimbang Beban Aplikasi.

Langkah 5: Buat penyeimbang beban aplikasi Agen

Buat penyeimbang beban aplikasi untuk menyeimbangkan Agen dan Pialang.

Untuk membuat penyeimbang beban

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>

Di panel navigasi, pilih Load Balancers dan kemudian pilih Create Load Balancer. Untuk tipe load balancer, pilih Application Load Balancer.
2. Untuk Langkah 1: Konfigurasi Penyeimbang Beban, lakukan hal berikut:
 - a. Untuk Nama, masukkan nama deskriptif untuk penyeimbang beban.
 - b. Untuk Skema, pilih yang menghadap ke internet.
 - c. Untuk Protokol Load Balancer, pilih HTTPS, dan untuk Port Load Balancer, masukkan 8445
 - d. Untuk VPC, pilih VPC yang akan digunakan dan kemudian pilih semua subnet di VPC itu.
 - e. Pilih Berikutnya.

3. Untuk Langkah 2: Konfigurasi Pengaturan Keamanan, lakukan hal berikut:
 - a. Untuk jenis Sertifikat, pilih Pilih sertifikat dari ACM.
 - b. Untuk nama Sertifikat, pilih sertifikat pribadi yang Anda minta sebelumnya.
 - c. Pilih Berikutnya.
4. Untuk Langkah 3: Konfigurasi Grup Keamanan, buat grup keamanan baru, atau pilih grup keamanan yang ada yang memungkinkan lalu lintas masuk dan keluar Agen dan Broker melalui HTTPS dan port 8445.

Pilih Berikutnya.
5. Untuk Langkah 4: Konfigurasi Routing, lakukan hal berikut:
 - a. Untuk grup Target, pilih Grup target baru.
 - b. Untuk Name, masukkan nama untuk grup target.
 - c. Untuk tipe Target, pilih Instance.
 - d. Untuk Protokol, pilih HTTPS. Untuk Port, masukkan 8445. Untuk versi Protokol, pilih HTTP1.
 - e. Untuk pemeriksaan kesehatan Protokol, pilih HTTPS, dan untuk Path, masukkan/health.
 - f. Pilih Berikutnya.
6. Untuk Langkah 5: Daftarkan Target, pilih semua instans Broker dan pilih Tambahkan ke terdaftar. Pilih Berikutnya: Tinjauan.
7. Pilih Buat.

Langkah 6: Luncurkan Agen

Buat Agen awal dan konfigurasi untuk menggunakan penyeimbang beban, buat AMI dari Agen, lalu gunakan AMI untuk meluncurkan Agen yang tersisa. Ini memastikan bahwa semua Agen dikonfigurasi untuk menggunakan konfigurasi penyeimbang beban yang sama.

Untuk meluncurkan Agen

1. Siapkan server Amazon DCV. Untuk informasi selengkapnya, lihat [Langkah 1: Siapkan server Amazon DCV](#).
2. Tempatkan salinan kunci publik CA yang dibuat di [Langkah 2: Siapkan sertifikat SSL untuk penyeimbang beban](#). Pilih atau buat direktori yang dapat dibaca oleh pengguna mana pun. File kunci publik CA harus dapat dibaca oleh pengguna mana pun juga.

3. Instal dan konfigurasi Agen. Untuk informasi selengkapnya tentang menginstal dan mengonfigurasi Agen, lihat [Langkah 3: Siapkan agen Manajer Sesi Amazon DCV](#).

 Important

Saat memodifikasi file konfigurasi Agen:

- untuk `broker_host` parameter, masukkan DNS Agent load balancer
- untuk `ca_file` parameter, masukkan path ke file kunci publik CA yang dibuat pada langkah sebelumnya

4. Konfigurasi server Amazon DCV untuk menggunakan Broker sebagai server otentikasi. Untuk informasi selengkapnya, lihat [Langkah 4: Konfigurasi server Amazon DCV untuk menggunakan broker sebagai server otentikasi](#).

 Important

Saat memodifikasi file konfigurasi server Amazon DCV:

- untuk `ca-file` parameter, masukkan jalur yang sama ke file kunci publik CA yang digunakan pada langkah sebelumnya
- untuk `auth-token-verifier` parameter, gunakan DNS penyeimbang beban Agen untuk `broker_ip_or_dns`

5. Hentikan contoh Agen.
6. Buat AMI dari instans Agen yang dihentikan. Untuk informasi selengkapnya, lihat [Membuat AMI Linux dari instans](#) di Panduan Pengguna Amazon EC2 untuk Instans Linux.
7. Gunakan AMI untuk meluncurkan Agen yang tersisa dan tetapkan profil instans yang Anda buat untuk semuanya.
8. Tetapkan grup keamanan yang memungkinkan Agen memuat lalu lintas jaringan penyeimbang ke semua instance Agen. Untuk informasi selengkapnya tentang port jaringan, lihat [File Konfigurasi Agen](#).

Menggunakan tag untuk menargetkan server Amazon DCV

Anda dapat menetapkan tag kustom ke Agen Pengelola Sesi untuk membantu mengidentifikasi dan mengkategorikannya serta server Amazon DCV yang terkait dengannya. Saat membuat sesi Amazon

DCV baru, Anda dapat menargetkan grup server Amazon DCV berdasarkan tag yang ditetapkan ke Agen masing-masing. Untuk informasi selengkapnya tentang cara menargetkan server Amazon DCV berdasarkan tag Agen, lihat [CreateSessionRequests](#) di Panduan Pengembang Manajer Sesi.

Tag terdiri dari kunci tag dan pasangan nilai, dan Anda dapat menggunakan pasangan informasi apa pun yang masuk akal untuk kasus penggunaan atau lingkungan Anda. Anda dapat memilih untuk menandai Agen berdasarkan konfigurasi perangkat keras host mereka. Misalnya, Anda dapat menandai semua Agen dengan host yang memiliki memori 4 GB `ram=4GB`. Atau Anda dapat menandai Agen berdasarkan tujuan. Misalnya, Anda dapat menandai semua Agen yang berjalan di host produksi dengan `purpose=production`.

Untuk menetapkan tag ke Agen

1. Menggunakan editor teks pilihan Anda, buat file baru dan berikan nama deskriptif, misalnya `agent_tags.toml`. Jenis file harus `.toml`, dan isi file harus ditentukan dalam format file TOML.
2. Dalam file, tambahkan setiap kunci tag baru dan pasangan nilai pada baris baru menggunakan `key=value` format. Contoh:

```
tag1="abc"  
tag2="xyz"
```

3. Buka file konfigurasi Agen (`/etc/dcv-session-manager-agent/agent.conf` untuk Linux dan macOS atau `C:\Program Files\NICE\DCVSessionManagerAgent\conf\agent.conf` untuk Windows). Untuk `tags_folder`, dan tentukan jalur ke direktori tempat file tag berada.

Jika direktori berisi beberapa file tag, semua tag yang ditentukan di seluruh file menerapkan Agen. File dibaca dalam urutan abjad. Jika beberapa file berisi tag dengan kunci yang sama, nilainya ditimpa dengan nilai dari file baca terakhir.

4. Simpan dan tutup file .
5. Berhenti dan mulai ulang Agen.

- Windows

```
C:\> sc stop DcvSessionManagerAgentService
```

```
C:\> sc start DcvSessionManagerAgentService
```

- Linux

```
$ sudo systemctl stop dcv-session-manager-agent
```

```
$ sudo systemctl start dcv-session-manager-agent
```

- macOS

```
$ sudo launchctl unload /Library/LaunchDaemons/com.amazon.dcv.session-  
manager.agent.plist
```

```
$ sudo launchctl load /Library/LaunchDaemons/com.amazon.dcv.session-  
manager.agent.plist
```

Mengkonfigurasi server otorisasi eksternal

Server otorisasi adalah server yang bertanggung jawab untuk mengautentikasi dan mengotorisasi klien SDKs dan Agen.

Secara default, Session Manager menggunakan Broker sebagai server otorisasi untuk menghasilkan token akses OAuth 2.0 untuk pernyataan klien SDKs dan perangkat lunak untuk Agen. Jika Anda menggunakan Broker sebagai server otorisasi, tidak diperlukan konfigurasi tambahan.

Anda dapat mengonfigurasi Pengelola Sesi untuk menggunakan Amazon Cognito sebagai server otorisasi eksternal, bukan Broker. Untuk informasi selengkapnya, tentang Amazon Cognito, lihat Panduan Pengembang [Amazon Cognito](#).

Untuk menggunakan Amazon Cognito sebagai server otorisasi

1. Buat kumpulan pengguna Amazon Cognito baru. Untuk informasi selengkapnya tentang kumpulan pengguna, lihat [Fitur Amazon Cognito](#) di Panduan Pengembang Amazon Cognito.

Gunakan [create-user-pool](#) perintah, dan tentukan nama pool dan Region untuk membuatnya.

Dalam contoh ini, kami memberi nama kolam `dcv-session-manager-client-app` dan kami membuatnya `us-east-1`.

```
$ aws cognito-idp create-user-pool --pool-name dcv-session-manager-client-app --  
region us-east-1
```

Contoh Output

```
{  
  "UserPoolClient": {  
    "UserPoolId": "us-east-1_QLEXAMPLE",  
    "ClientName": "dcv-session-manager-client-app",  
    "ClientId": "15hhd8jjj74hf32f24uEXAMPLE",  
    "LastModifiedDate": 1602510048.054,  
    "CreationDate": 1602510048.054,  
    "RefreshTokenValidity": 30,  
    "AllowedOAuthFlowsUserPoolClient": false  
  }  
}
```

Catata `userPoolId`, Anda akan membutuhkannya di langkah berikutnya.

2. Buat domain baru untuk kumpulan pengguna Anda. Gunakan [create-user-pool-domain](#) perintah, dan tentukan nama domain dan kumpulan pengguna yang Anda buat pada langkah sebelumnya. `userPoolId`

Dalam contoh ini, nama domain adalah `mydomain-544fa30f-c0e5-4a02-8d2a-a3761EXAMPLE` dan kami membuatnya di `us-east-1`.

```
$ aws cognito-idp create-user-pool-domain --domain mydomain-544fa30f-  
c0e5-4a02-8d2a-a3761EXAMPLE --user-pool-id us-east-1_QLEXAMPLE --region us-east-1
```

Contoh Output

```
{  
  "DomainDescription": {  
    "UserPoolId": "us-east-1_QLEXAMPLE",  
    "AWSAccountId": "123456789012",  
    "Domain": "mydomain-544fa30f-c0e5-4a02-8d2a-a3761EXAMPLE",  
    "S3Bucket": "aws-cognito-prod-pdx-assets",  
    "CloudFrontDistribution": "dpp0gtexample.cloudfront.net",  
    "Version": "20201012133715",  
    "Status": "ACTIVE",  
  }  
}
```

```

    "CustomDomainConfig": {}
  }
}

```

Format domain kumpulan pengguna adalah sebagai berikut: `https://domain_name.auth.region.amazoncognito.com`. Dalam contoh ini, domain pool pengguna adalah `https://mydomain-544fa30f-c0e5-4a02-8d2a-a3761EXAMPLE.auth.us-east-1.amazoncognito.com`.

3. Membuat klien kolam pengguna. Gunakan [create-user-pool-client](#) perintah dan tentukan `userPoolId` kumpulan pengguna yang Anda buat, nama untuk klien, dan Wilayah tempat membuatnya. Juga, sertakan `--generate-secret` opsi untuk menentukan bahwa Anda ingin menghasilkan rahasia untuk klien kumpulan pengguna yang sedang dibuat.

Dalam hal ini, nama klien adalah `dcv-session-manager-client-app` dan kami membuatnya di `us-east-1` Wilayah.

```

$ aws cognito-idp create-user-pool-client --user-pool-id us-east-1_QLEXAMPLE --
client-name dcv-session-manager-client-app --generate-secret --region us-east-1

```

Contoh Output

```

{
  "UserPoolClient": {
    "UserPoolId": "us-east-1_QLEXAMPLE",
    "ClientName": "dcv-session-manager-client-app",
    "ClientId": "219273hp6k2ut5cugg9EXAMPLE",
    "ClientSecret": "1vp5e8nec7cbf4m9me55mbmht91u61hlh0a78rq1qki11EXAMPLE",
    "LastModifiedDate": 1602510291.498,
    "CreationDate": 1602510291.498,
    "RefreshTokenValidity": 30,
    "AllowedOAuthFlowsUserPoolClient": false
  }
}

```

Note

Buat catatan tentang `ClientId` dan `ClientSecret`. Anda harus memberikan informasi ini kepada pengembang ketika mereka meminta token akses untuk permintaan API.

4. Buat server sumber OAuth2 daya.0 baru untuk kumpulan pengguna. Sebuah server sumber daya adalah server untuk sumber daya yang dilindungi akses. Ini menangani permintaan otentikasi untuk token akses.

Gunakan [create-resource-server](#) perintah dan tentukan kumpulan pengguna, pengenal unik dan nama untuk server sumber daya, ruang lingkup, dan Wilayah tempat membuatnya. `userPoolId`

Dalam contoh ini, kita gunakan `dcv-session-manager` sebagai pengenal dan nama, dan kita gunakan `sm_scope` sebagai nama lingkup dan deskripsi.

```
$ aws cognito-idp create-resource-server --user-pool-id us-east-1_QLEXAMPLE
--identifier dcv-session-manager --name dcv-session-manager --scopes
ScopeName=sm_scope,ScopeDescription=sm_scope --region us-east-1
```

Contoh Output

```
{
  "ResourceServer": {
    "UserPoolId": "us-east-1_QLEXAMPLE",
    "Identifier": "dcv-session-manager",
    "Name": "dcv-session-manager",
    "Scopes": [
      {
        "ScopeName": "sm_scope",
        "ScopeDescription": "sm_scope"
      }
    ]
  }
}
```

5. Perbarui klien kumpulan pengguna.

Gunakan perintah [update-user-pool-client](#). Tentukan `userPoolId` kumpulan pengguna, klien kumpulan pengguna, dan Wilayah. `ClientId` Untuk `--allowed-o-auth-flows`, tentukan `client_credentials` untuk menunjukkan bahwa klien harus mendapatkan token akses dari titik akhir token dengan menggunakan kombinasi ID klien dan rahasia klien. Untuk `--allowed-o-auth-scopes`, tentukan pengenal server sumber daya dan nama lingkup sebagai berikut: `resource_server_identifier/scope_name`. Sertakan `--allowed-o-auth-flows-user-pool-client` untuk menunjukkan bahwa klien diizinkan untuk mengikuti OAuth protokol saat berinteraksi dengan kumpulan pengguna Cognito.

```
$ aws cognito-idp update-user-pool-client --user-pool-id us-east-1_QLEXAMPLE --
client-id 219273hp6k2ut5cugg9EXAMPLE --allowed-o-auth-flows client_credentials --
allowed-o-auth-scopes dcv-session-manager/sm_scope --allowed-o-auth-flows-user-
pool-client --region us-east-1
```

Contoh Output

```
{
  "UserPoolClient": {
    "UserPoolId": "us-east-1_QLEXAMPLE",
    "ClientName": "dcv-session-manager-client-app",
    "ClientId": "219273hp6k2ut5cugg9EXAMPLE",
    "ClientSecret": "1vp5e8nec7cbf4m9me55mbmht91u61h1h0a78rq1qki11EXAMPLE",
    "LastModifiedDate": 1602512103.099,
    "CreationDate": 1602510291.498,
    "RefreshTokenValidity": 30,
    "AllowedOAuthFlows": [
      "client_credentials"
    ],
    "AllowedOAuthScopes": [
      "dcv-session-manager/sm_scope"
    ],
    "AllowedOAuthFlowsUserPoolClient": true
  }
}
```

Note

Kumpulan pengguna sekarang siap untuk menyediakan dan mengautentikasi token akses. Dalam contoh ini, URL untuk server otorisasi adalah `https://cognito-idp.us-east-1.amazonaws.com/us-east-1_QLEXAMPLE/.well-known/jwks.json`.

6. Uji konfigurasi.

```
$ curl -H "Authorization: Basic `echo -
n 219273hp6k2ut5cugg9EXAMPLE:1vp5e8nec7cbf4m9me55mbmht91u61h1h0a78rq1qki11EXAMPLE
| base64`" -H "Content-Type: application/x-www-form-urlencoded" -X
POST "https://mydomain-544fa30f-c0e5-4a02-8d2a-a3761EXAMPLE.auth.us-
```

```
east-1.amazonaws.com/oauth2/token?grant_type=client_credentials&scope=dcv-session-manager/sm_scope"
```

Contoh Output

```
{
  "access_token": "eyJraWQiOiJGQ0VaRFpJUUpT3NSaW41MmtqaDdEbTZYb0RnSTQ5b2VUT0cxUU1Q2VJPSIsImF1Z2VudD06Zm9udjTXXkzHlZGScr6R0dZtId5dThkpEziSx0YwiiWe9crAlqoazlDcCsUJHIXDtgKW64pSj3-uQQGg1Jv_tyVjhrA4JbD0k67WS2V9NW-uZ7t4zwwaUm0i3KzpBmi54fpVgPaewiVlUm_aS4LUFcWT6hVJjiZF7om7984qb2g0a14iZxpXPBJTZX_gtG9EtvnS9u",
  "expires_in": 3600,
  "token_type": "Bearer"
}
```

7. Daftarkan server otorisasi eksternal untuk digunakan dengan broker dengan menggunakan [register-auth-server](#) perintah.

```
$ sudo -u root dcv-session-manager-broker register-auth-server --url https://cognito-idp.us-east-1.amazonaws.com/us-east-1_QLEXAMPLE/.well-known/jwks.json
```

Pengembang sekarang dapat menggunakan server untuk meminta token akses. Saat meminta token akses, berikan ID klien, rahasia klien, dan URL server yang dihasilkan di sini. Untuk informasi selengkapnya tentang meminta token akses, lihat [Membuat token akses dan membuat permintaan API](#) di Panduan Pengembang Manajer Sesi Amazon DCV.

Mengkonfigurasi ketekunan broker


Broker Session Manager mendukung integrasi dengan database eksternal. Database eksternal memungkinkan Session Manager untuk mempertahankan data status dan kunci sehingga mereka tersedia setelahnya. Bahkan data broker didistribusikan melalui cluster, membuatnya rentan terhadap kehilangan data jika host perlu reboot atau cluster dihentikan. Dengan fitur ini diaktifkan, Anda dapat menambah dan menghapus node broker. Selain itu, Anda dapat menghentikan cluster dan memulai ulang, tanpa perlu membuat ulang kunci atau kehilangan informasi tentang Amazon DCV Server mana yang terbuka atau ditutup.

Jenis informasi berikut dapat diatur untuk bertahan:

- Kunci untuk mengatur sesi untuk menjalin koneksi dengan klien
- Data sesi dalam penerbangan

- Status server Amazon DCV

Amazon DCV Session Manager mendukung database DynamoDB, MariaDB, dan MySQL. Anda harus mengatur dan mengelola salah satu database ini untuk menggunakan fitur ini. Jika mesin broker Anda di-host di Amazon EC2, kami sarankan menggunakan DynamoDB sebagai database eksternal, karena tidak memerlukan pengaturan tambahan.

 Note

Anda mungkin dikenakan biaya tambahan saat menjalankan database eksternal. Untuk melihat informasi tentang harga DynamoDB, [lihat Harga](#) untuk Kapasitas yang Disediakan.

Konfigurasi broker untuk bertahan di DynamoDB

Konfigurasi broker untuk mulai menyimpan data mereka di DynamoDB:

1. Buka `/etc/dcv-session-manager-broker/session-manager-broker.properties` menggunakan editor teks pilihan Anda dan lakukan pengeditan berikut:
 - Set `enable-persistence = true`
 - Set `persistence-db = dynamodb`
 - Untuk `dynamodb-region` tentukan `&aws;` Wilayah tempat Anda ingin menyimpan tabel yang berisi data broker. Untuk daftar Wilayah yang didukung, lihat titik akhir layanan [DynamoDB](#).
 - Untuk `dynamodb-table-rcu` tentukan jumlah Unit Kapasitas Baca (RCU) yang didukung setiap tabel. Untuk informasi selengkapnya tentang RCU, lihat Kapasitas yang disediakan [DynamoDB](#).
 - Untuk `dynamodb-table-wcu` tentukan jumlah Unit Kapasitas Tulis (WCU) yang didukung setiap tabel. Untuk info selengkapnya tentang WCU, lihat Kapasitas yang disediakan [DynamoDB](#).
 - Untuk `dynamodb-table-name-prefix` menentukan awalan yang ditambahkan ke setiap tabel DynamoDB (berguna untuk membedakan beberapa cluster broker menggunakan akun yang sama). Hanya karakter alfanumerik, titik, tanda hubung, dan garis bawah yang diizinkan.
2. Hentikan semua broker di cluster. Untuk setiap broker, jalankan perintah berikut:

```
sudo systemctl stop dcv-session-manager-broker
```

3. Pastikan semua broker di cluster dihentikan dan kemudian restart semuanya. Mulai setiap broker dengan menjalankan perintah berikut:

```
sudo systemctl start dcv-session-manager-broker
```

Host broker harus memiliki izin untuk memanggil DynamoDB APIs. Pada EC2 instans Amazon, kredensial diambil secara otomatis menggunakan layanan metadata Amazon. EC2 Jika Anda perlu menentukan kredensial yang berbeda, Anda dapat mengaturnya menggunakan salah satu teknik pengambilan kredensial yang didukung (seperti properti sistem Java atau variabel lingkungan). Untuk informasi selengkapnya, lihat [Menyediakan dan Mengambil &aws; Credentials](#).

Konfigurasi broker untuk bertahan di MariaDB/MySQL

Note

`/etc/dcv-session-manager-broker/session-manager-broker.properties` berisi data sensitif. Secara default, akses tulisnya dibatasi untuk root dan akses bacanya dibatasi untuk root dan pengguna yang menjalankan Broker. Secara default, ini adalah `dcvsmbroker` pengguna. Broker memeriksa saat startup bahwa file tersebut memiliki izin yang diharapkan.

Konfigurasi broker untuk mulai mempertahankan data mereka di MariaDB/MySQL:

1. Buka `/etc/dcv-session-manager-broker/session-manager-broker.properties` dengan editor teks pilihan Anda dan lakukan pengeditan berikut:

- Set `enable-persistence = true`
- Set `persistence-db = mysql`
- Set `jdbc-connection-url = jdbc:mysql://<db_endpoint>:<db_port>/<db_name>?createDatabaseIfNotExist=true`

Dalam konfigurasi ini, `<db_endpoint>` adalah endpoint database, `<db_port>` adalah port database, dan `<db_name>` merupakan nama database.

- Untuk `jdbc-user` tentukan nama pengguna yang memiliki akses ke database.
- Untuk `jdbc-password` tentukan kata sandi pengguna yang memiliki akses ke database.

2. Hentikan semua broker di cluster. Untuk setiap broker, jalankan perintah berikut:

```
sudo systemctl stop dcv-session-manager-broker
```

3. Pastikan semua broker di cluster dihentikan, dan kemudian restart semuanya. Untuk setiap broker, jalankan perintah berikut:

```
sudo systemctl start dcv-session-manager-broker
```

Mengintegrasikan dengan Amazon DCV Connection Gateway

[Amazon DCV Connection Gateway](#) adalah paket perangkat lunak yang dapat diinstal yang memungkinkan pengguna mengakses armada server Amazon DCV melalui satu titik akses ke LAN atau VPC.

Jika infrastruktur Anda menyertakan server Amazon DCV yang dapat diakses melalui Amazon DCV Connection Gateway, Anda dapat mengonfigurasi Session Manager untuk mengintegrasikan Amazon DCV Connection Gateway. Dengan mengikuti langkah-langkah yang diuraikan di bagian berikut, broker akan bertindak sebagai [Resolver Sesi](#) untuk Connection Gateway. Dengan kata lain, broker akan mengekspos titik akhir HTTP tambahan. Connection Gateway akan melakukan panggilan API ke titik akhir untuk mengambil informasi yang diperlukan untuk merutekan koneksi Amazon DCV ke host yang dipilih oleh broker.

Topik

- [Siapkan Broker Manajer Sesi sebagai Resolver Sesi untuk Amazon DCV Connection Gateway](#)
- [Opsional - Aktifkan otentikasi klien TLS](#)
- [Manajer Sesi Amazon DCV Server Amazon DCV - Referensi pemetaan DNS](#)

Siapkan Broker Manajer Sesi sebagai Resolver Sesi untuk Amazon DCV Connection Gateway

Sisi Broker Manajer Sesi

1. Buka `/etc/dcv-session-manager-broker/session-manager-broker.properties` menggunakan editor teks pilihan Anda dan terapkan perubahan berikut:

- Set `enable-gateway = true`
 - Setel `gateway-to-broker-connector-https-port` ke port TCP gratis (defaultnya adalah 8447)
 - Setel `gateway-to-broker-connector-bind-host` ke alamat IP host tempat Broker mengikat koneksi Amazon DCV Connection Gateway (defaultnya adalah 0.0.0.0)
2. Kemudian jalankan perintah berikut untuk menghentikan dan memulai ulang Broker:

```
sudo systemctl stop dcv-session-manager-broker
```

```
sudo systemctl start dcv-session-manager-broker
```

3. Ambil salinan sertifikat yang ditandatangani sendiri Broker dan letakkan di direktori pengguna Anda.

```
sudo cp /var/lib/dcvsmbroker/security/dcvsmbroker_ca.pem $HOME
```

Anda akan membutuhkannya saat menginstal Amazon DCV Connection Gateway di langkah berikutnya.

Sisi Gerbang Koneksi Amazon DCV

- Silakan ikuti [bagian](#) dalam dokumentasi Amazon DCV Connection Gateway.

Karena Amazon DCV Connection Gateway melakukan panggilan HTTP API ke broker, jika broker menggunakan sertifikat yang ditandatangani sendiri, Anda harus menyalin sertifikat broker ke host Amazon DCV Connection Gateway (diambil pada langkah sebelumnya) dan mengatur `ca-file` parameter di bagian konfigurasi Amazon DCV Connection Gateway. `[resolver]`

Opsional - Aktifkan otentikasi klien TLS

Setelah Anda menyelesaikan langkah sebelumnya, Session Manager dan Connection Gateway dapat berkomunikasi melalui saluran aman, di mana Connection Gateway dapat memverifikasi identitas Broker Manajer Sesi. Jika Anda juga mengharuskan Broker Manajer Sesi memvalidasi identitas Connection Gateway sebelum membuat saluran aman, Anda harus mengaktifkan fitur otentikasi klien TLS, mengikuti langkah-langkah di bagian berikutnya.

Note

Jika Manajer Sesi berada di belakang penyeimbang beban, otentikasi klien TLS tidak dapat diaktifkan dengan penyeimbang beban yang memiliki penghentian koneksi TLS, seperti Application Load Balancers () atau Gateway Load Balancers ()ALBs. GLBs Hanya penyeimbang beban tanpa penghentian TLS yang dapat didukung, seperti Network Load Balancers (). NLBs Jika Anda menggunakan ALBs atau GLBs, Anda dapat menegaskan bahwa hanya grup keamanan tertentu yang dapat menghubungi penyeimbang beban, memastikan tingkat keamanan tambahan; info selengkapnya tentang grup keamanan di sini: Grup keamanan untuk [VPC](#) Anda

Sisi Broker Manajer Sesi

1. Untuk mengaktifkan otentikasi klien TLS untuk komunikasi antara Broker Manajer Sesi dan Amazon DCV Connection Gateway, ikuti langkah-langkah berikut:
2. Hasilkan kunci dan sertifikat yang diperlukan dengan menjalankan: Output dari perintah akan memberi tahu Anda folder tempat kredensial dibuat dan kata sandi yang digunakan untuk membuat file. TrustStore

```
sudo /usr/share/dcv-session-manager-broker/bin/gen-gateway-certificates.sh
```

3. Tempatkan salinan kunci pribadi Amazon DCV Connection Gateway dan sertifikat yang ditandatangani sendiri di direktori pengguna Anda. Anda akan membutuhkannya saat mengaktifkan otentikasi klien TLS di Amazon DCV Connection Gateway pada langkah berikutnya.

```
sudo cp /etc/dcv-session-manager-broker/resolver-creds/dcv_gateway_key.pem $HOME
```

```
sudo cp /etc/dcv-session-manager-broker/resolver-creds/dcv_gateway_cert.pem $HOME
```

4. Kemudian buka `/etc/dcv-session-manager-broker/session-manager-broker.properties` menggunakan editor teks pilihan Anda dan lakukan hal berikut:
 - Setel `enable-tls-client-auth-gateway` ke `true`
 - Atur `gateway-to-broker-connector-trust-store-file` ke jalur TrustStore file yang dibuat pada langkah sebelumnya

- Setel `gateway-to-broker-connector-trust-store-pass` ke kata sandi yang digunakan untuk membuat TrustStore file di langkah sebelumnya
5. Kemudian jalankan perintah berikut untuk menghentikan dan memulai ulang Broker:

```
sudo systemctl stop dcv-session-manager-broker
```

```
sudo systemctl start dcv-session-manager-broker
```

Sisi Gerbang Koneksi Amazon DCV

- Silakan ikuti [bagian](#) dalam dokumentasi Amazon DCV Connection Gateway.
 - gunakan jalur lengkap file sertifikat yang Anda salin pada langkah sebelumnya saat mengatur `cert-file` parameter di bagian `[resolver]`
 - gunakan jalur lengkap dari file kunci yang Anda salin pada langkah sebelumnya saat mengatur `cert-key-file` parameter di bagian `[resolver]`

Manajer Sesi Amazon DCV Server Amazon DCV - Referensi pemetaan DNS

Amazon DCV Connection Gateway memerlukan nama DNS server Amazon DCV untuk terhubung ke instance server DCV. Bagian ini menggambarkan bagaimana Anda dapat menentukan file JSON yang berisi pemetaan antara setiap Server DCV dan nama DNS yang terkait.

Struktur file

Pemetaan terdiri dari daftar objek JSON dengan bidang berikut:

```
[
  {
    "ServerIdType": "Ip",
    "ServerId": "192.168.0.1",
    "DnsNames":
    {
      "InternalDnsName": "internal"
    }
  },
]
```

```
...  
]
```

Di mana:

ServerIdType:

Mengidentifikasi jenis id yang mengacu pada nilai; saat ini nilai yang tersedia adalah IPAdress,, dan agentServerId InstanceId:

Ip:

Tersedia untuk infrastruktur Amazon EC2 dan di lokasi; dapat dengan cepat diambil oleh administrator sistem dengan perintah ifconfig (Linux/macOS) atau ipconfig (Windows). Info ini juga tersedia dalam respons DescribeServers API.

Id:

Tersedia untuk infrastruktur Amazon EC2 dan di lokasi; Agen Manajer Sesi membuat UUID baru setiap kali nama host atau alamat ip berubah. Info ini tersedia dalam respons DescribeServers API.

Host.Aws.Ec2InstanceId:

Hanya tersedia untuk EC2 instans Amazon, ini secara unik mengidentifikasi mesin; itu tidak berubah setelah instance reboot. Dapat diambil di host dengan menghubungi <http://169.254.169.254/latest/meta-data/instance-id>. Info ini juga tersedia dalam respons DescribeServers API.

ServerId:

Id dari jenis tertentu yang secara unik mengidentifikasi setiap server Amazon DCV dalam jaringan.

DnsNames:

Objek yang berisi nama DNS yang terkait dengan server Amazon DCV objek ini akan berisi:

InternalDnsNames:

Nama DNS yang digunakan oleh Amazon DCV Connection Gateway untuk terhubung ke instans.

Silakan gunakan perintah CLI Broker Manajer Sesi `register-server-dns-mapping` untuk memuat pemetaan dari file (referensi halaman perintah: [register-server-dns-mapping](#)) dan

`describe-server-dns-mappings` untuk membuat daftar pemetaan yang saat ini dimuat di Broker Manajer Sesi (referensi halaman perintah:). [describe-server-dns-mappings](#)

Tetap

Kami sangat menyarankan agar Anda mengaktifkan fitur persistensi dari Session Manager Broker, untuk melindungi terhadap kerugian pemetaan ketika beberapa broker atau seluruh cluster turun. Untuk informasi selengkapnya tentang mengaktifkan persistensi data, lihat [Mengonfigurasi Persistensi Broker](#)

Integrasi dengan Amazon CloudWatch

Session Manager mendukung integrasi dengan Amazon CloudWatch for Brokers yang berjalan di EC2 instans Amazon, dan juga Broker yang berjalan di host lokal.

Amazon CloudWatch memantau sumber daya Amazon Web Services (AWS) Anda dan aplikasi yang Anda jalankan AWS secara real time. Anda dapat menggunakan CloudWatch untuk mengumpulkan dan melacak metrik, yang merupakan variabel yang dapat Anda ukur untuk sumber daya dan aplikasi Anda. Untuk informasi selengkapnya, lihat [Panduan CloudWatch Pengguna Amazon](#).


Anda dapat mengonfigurasi Broker Manajer Sesi untuk mengirim data metrik berikut ke Amazon CloudWatch:

- `Number of DCV servers`—Jumlah server DCV yang dikelola oleh Broker.
- `Number of ready DCV servers`—Jumlah server DCV yang berada di READY negara bagian yang dikelola oleh Broker.
- `Number of DCV sessions`—Jumlah sesi DCV yang dikelola oleh Broker.
- `Number of DCV console sessions`—Jumlah sesi konsol DCV yang dikelola oleh Broker.
- `Number of DCV virtual sessions`—Jumlah sesi virtual DCV yang dikelola oleh Broker.
- `Heap memory used`—Jumlah memori heap yang digunakan oleh Broker.
- `Off-heap memory used`—Jumlah memori off-heap yang digunakan oleh Broker.
- `Describe sessions request time`—Jumlah waktu yang dibutuhkan untuk menyelesaikan permintaan `DescribeSessions` API.
- `Delete sessions request time`—Jumlah waktu yang dibutuhkan untuk menyelesaikan permintaan `DeleteSessions` API.
- `Create sessions request time`—Jumlah waktu yang dibutuhkan untuk menyelesaikan permintaan `CreateSessions` API.

- `Get session connection data request time`—Jumlah waktu yang dibutuhkan untuk menyelesaikan permintaan `GetSessionConnectionData` API.
- `Update session permissions request time`—Jumlah waktu yang dibutuhkan untuk menyelesaikan permintaan `UpdateSessionPermissions` API.

Untuk mengonfigurasi Broker untuk mengirim data metrik ke Amazon CloudWatch

1. Buka `/etc/dcv-session-manager-broker/session-manager-broker.properties` menggunakan editor teks pilihan Anda dan lakukan hal berikut:
 - Setel `enable-cloud-watch-metrics` ke `true`
 - Untuk `cloud-watch-region`, tentukan Wilayah tempat mengumpulkan data metrik.

 Note

Jika Broker Anda berjalan pada EC2 instans Amazon, parameter ini opsional. Region secara otomatis diambil dari Layanan Metadata Instans (IMDS). Jika Anda menjalankan Broker di host lokal, parameter ini wajib.

2. Berhenti dan mulai ulang Broker.

```
$ sudo systemctl stop dcv-session-manager-broker
```

```
$ sudo systemctl start dcv-session-manager-broker
```

Host Broker juga harus memiliki izin untuk memanggil `cloudwatch:PutMetricData` API. AWS kredensial dapat diambil menggunakan salah satu teknik pengambilan kredensial yang didukung. Untuk informasi selengkapnya, lihat [Menyediakan dan Mengambil AWS Kredensi](#).

Memutakhirkan Manajer Sesi Amazon DCV

Seiring pertumbuhan sistem Amazon DCV dalam skala dan kompleksitas, penting untuk memastikan Manajer Sesi tetap up-to-date dan mampu menangani permintaan yang meningkat. Baik paket agen dan broker akan membutuhkan peningkatan dari waktu ke waktu. Bagian ini menguraikan proses untuk memutakhirkan Manajer Sesi Amazon DCV, yang mencakup prosedur pemutakhiran dan rekomendasi untuk memelihara sistem Anda.

Topik berikut menjelaskan cara meng-upgrade Session Manager.

Note

Kami sangat menyarankan agar Anda meningkatkan semua agen Session Manager sebelum meningkatkan broker Session Manager untuk menghindari masalah ketidakcocokan saat fitur baru diperkenalkan.

Topik

- [Memutakhirkan agen Manajer Sesi Amazon DCV](#)
- [Memutakhirkan broker Amazon DCV Session Manager](#)

Memutakhirkan agen Manajer Sesi Amazon DCV

Agan Manajer Sesi Amazon DCV menerima instruksi dari broker dan menjalankannya di server Amazon DCV masing-masing. Sebagai bagian dari pemeliharaan rutin, agen perlu ditingkatkan untuk memenuhi standar dan persyaratan baru. Bagian ini memandu Anda melalui proses upgrade agen Session Manager Anda.

Linux host

Note

Instruksi berikut adalah untuk menginstal agen pada host 64-bit x86. Untuk menginstal agen pada host ARM 64-bit, untuk Amazon Linux, RHEL, dan Centos, ganti `x86_64` dengan `aarch64`, dan untuk Ubuntu, ganti `amd64` dengan `arm64`.

Untuk memperbarui agen pada host Linux

1. Jalankan perintah berikut untuk menghentikan agen.

```
$ sudo systemctl stop dcv-session-manager-agent
```

2. Unduh paket instalasi.

- Amazon Linux 2 dan RHEL 7.x

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2025.0/SessionManagerAgents/nice-dcv-session-manager-agent-2025.0.888-1.el7.x86_64.rpm
```

- RHEL 8.x dan Rocky Linux 8.x

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2025.0/SessionManagerAgents/nice-dcv-session-manager-agent-2025.0.888-1.el8.x86_64.rpm
```

- Ubuntu 20.04

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2025.0/SessionManagerAgents/nice-dcv-session-manager-agent_2025.0.888-1_amd64.ubuntu2004.deb
```

- Ubuntu 22.04

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2025.0/SessionManagerAgents/nice-dcv-session-manager-agent_2025.0.888-1_amd64.ubuntu2204.deb
```

- Ubuntu 24.04

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2025.0/SessionManagerAgents/nice-dcv-session-manager-agent_2025.0.888-1_amd64.ubuntu2404.deb
```

- SUSE Linux Perusahaan 12

```
$ curl -O https://d1uj6qtbmh3dt5.cloudfront.net/2025.0/SessionManagerAgents/nice-dcv-session-manager-agent-2025.0.888-1.sles12.x86_64.rpm
```

- SUSE Linux Perusahaan 15

```
$ curl -O https://d1uj6qtbmh3dt5.cloudfront.net/2025.0/SessionManagerAgents/nice-dcv-session-manager-agent-2025.0.888-1.sles15.x86_64.rpm
```

3. Instal paket .

- Amazon Linux 2 dan RHEL 7.x

```
$ sudo yum install -y nice-dcv-session-manager-agent-2025.0.888-1.el7.x86_64.rpm
```

- RHEL 8.x dan Rocky Linux 8.x

```
$ sudo yum install -y nice-dcv-session-manager-agent-2025.0.888-1.el8.x86_64.rpm
```

- Ubuntu 20.04

```
$ sudo apt install ./nice-dcv-session-manager-agent_2025.0.888-1_amd64.ubuntu2004.deb
```

- Ubuntu 22.04

```
$ sudo apt install ./nice-dcv-session-manager-agent_2025.0.888-1_amd64.ubuntu2204.deb
```

- Ubuntu 24.04

```
$ sudo apt install ./nice-dcv-session-manager-agent_2025.0.888-1_amd64.ubuntu2404.deb
```

- SUSE Linux Perusahaan 12

```
$ sudo zypper install nice-dcv-session-manager-agent-2025.0.888-1.sles12.x86_64.rpm
```

- SUSE Linux Perusahaan 15

```
$ sudo zypper install nice-dcv-session-manager-agent-2025.0.888-1.sles15.x86_64.rpm
```

4. Jalankan perintah berikut untuk memulai agen.

```
$ sudo systemctl start dcv-session-manager-agent
```

Windows host

Untuk memperbarui agen pada host Windows

1. Hentikan layanan agen. Jalankan perintah berikut di command prompt.

```
C:\> sc stop DcvSessionManagerAgentService
```

2. Unduh [penginstal agen](#).
3. Jalankan pemasang. Pada layar Selamat Datang, pilih Berikutnya.
4. Pada layar EULA, baca perjanjian lisensi dengan cermat, dan jika Anda setuju, pilih Saya menerima persyaratan dan pilih Berikutnya.
5. Untuk memulai instalasi, pilih Instal.
6. Mulai ulang layanan agen. Jalankan perintah berikut di command prompt.

```
C:\> sc start DcvSessionManagerAgentService
```

macOS host

Untuk memperbarui agen pada host macOS

1. Jalankan perintah berikut untuk menghentikan agen.

```
$ sudo launchctl unload /Library/LaunchDaemons/com.amazon.dcv.session-manager.agent.plist
```

2. Unduh [penginstal agen](#) atau gunakan baris perintah:

```
$ curl -O https://d1uj6qtbmh3dt5.cloudfront.net/2025.0/SessionManagerAgents/nice-dcv-session-manager-agent-2025.0.888-macos-arm64.pkg
```

3. Instal paket . Anda dapat menggunakan baris perintah atau melalui penginstal agen:

- Baris perintah:

```
$ sudo installer -pkg ./nice-dcv-session-manager-agent-2025.0.888-macos-arm64.pkg -target /
```

- Grafis: Klik dua kali .pkg file dan ikuti panduan instalasi.

4. Jalankan perintah berikut untuk memulai agen.

```
$ sudo launchctl load /Library/LaunchDaemons/com.amazon.dcv.session-manager.agent.plist
```

Memutakhirkan broker Amazon DCV Session Manager

Broker Amazon DCV Session Manager meneruskan permintaan API ke agen terkait mereka. Mereka diinstal pada host yang terpisah dari server Amazon DCV. Sebagai bagian dari pemeliharaan rutin, broker perlu ditingkatkan untuk memenuhi standar dan persyaratan baru. Bagian ini memandu Anda melalui proses upgrade broker Session Manager Anda.

Untuk meng-upgrade broker

1. Connect ke host tempat Anda ingin meng-upgrade broker.
2. Hentikan layanan broker.

```
$ sudo systemctl stop dcv-session-manager-broker
```

3. Unduh paket instalasi.

- Amazon Linux 2 dan RHEL 7.x

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2025.0/SessionManagerBrokers/nice-dcv-session-manager-broker-2025.0.539-1.el7.noarch.rpm
```

- RHEL 8.x dan Rocky Linux 8.x

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2025.0/SessionManagerBrokers/nice-dcv-session-manager-broker-2025.0.539-1.el8.noarch.rpm
```

- Ubuntu 20.04

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2025.0/SessionManagerBrokers/nice-dcv-session-manager-broker-2025.0.539-1_all.ubuntu2004.deb
```

- Ubuntu 22.04

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2025.0/SessionManagerBrokers/nice-dcv-session-manager-broker-2025.0.539-1_all.ubuntu2204.deb
```

- Ubuntu 24.04

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2025.0/SessionManagerBrokers/nice-dcv-session-manager-broker-2025.0.539-1_all.ubuntu2404.deb
```

4. Instal paket .

- Amazon Linux 2 dan RHEL 7.x

```
$ sudo yum install -y nice-dcv-session-manager-broker-2025.0.539-1.el7.noarch.rpm
```

- RHEL 8.x dan Rocky Linux 8.x

```
$ sudo yum install -y nice-dcv-session-manager-broker-2025.0.539-1.el8.noarch.rpm
```

- Ubuntu 20.04

```
$ sudo apt install -y nice-dcv-session-manager-broker-2025.0.539-1_all.ubuntu2004.deb
```

- Ubuntu 22.04

```
$ sudo apt install -y nice-dcv-session-manager-broker-2025.0.539-1_all.ubuntu2204.deb
```

- Ubuntu 24.04

```
$ sudo apt install -y nice-dcv-session-manager-broker-2025.0.539-1_all.ubuntu2404.deb
```

5. Mulai layanan broker dan pastikan bahwa itu dimulai secara otomatis setiap kali instance dimulai.

```
$ sudo systemctl start dcv-session-manager-broker && sudo systemctl enable dcv-session-manager-broker
```

Referensi CLI broker

Broker Amazon DCV Session Manager adalah alat antarmuka baris perintah (CLI) yang menyediakan kontrol administratif atas Manajer Sesi. Referensi ini mencakup set lengkap perintah CLI yang tersedia untuk mengelola sesi, pengguna, sumber daya, dan aspek lain dari Manajer Sesi. Administrator dapat mengotomatiskan tugas manajemen rutin, memecahkan masalah, dan mengoptimalkan kinerja infrastruktur Amazon DCV mereka.

Gunakan perintah berikut jika Anda menggunakan server otentikasi eksternal untuk menghasilkan token akses OAuth 2.0:

- [register-auth-server](#)
- [list-auth-servers](#)
- [unregister-auth-server](#)

Gunakan perintah berikut jika Anda menggunakan broker Session Manager sebagai server otentikasi OAuth 2.0.

- [register-api-client](#)
- [describe-api-clients](#)
- [unregister-api-client](#)
- [renew-auth-server-api-kunci](#)

Gunakan perintah berikut untuk mengelola agen Session Manager.

- [generate-software-statement](#)
- [describe-software-statements](#)
- [deactivate-software-statement](#)
- [describe-agent-clients](#)
- [unregister-agent-client](#)

Gunakan perintah berikut untuk mengelola server DCV - file pemetaan nama DNS.

- [register-server-dns-mappings](#)
- [describe-server-dns-mappings](#)

register-auth-server

Mendaftarkan server otentikasi eksternal untuk digunakan dengan broker.

Secara default, Session Manager menggunakan broker sebagai server otentikasi untuk menghasilkan token akses OAuth 2.0. Jika Anda menggunakan broker sebagai server otentikasi, tidak diperlukan konfigurasi tambahan.

Namun, jika Anda memilih untuk menggunakan server otentikasi eksternal, seperti Active Directory atau Amazon Cognito, Anda harus menggunakan perintah ini untuk mendaftarkan server otentikasi eksternal.

Topik

- [Sintaksis](#)
- [Ops](#)
- [Contoh](#)

Sintaksis

```
sudo -u root dcv-session-manager-broker register-auth-server --url server_url.well-known/jwks.json
```

Ops

--url

URL server otentikasi eksternal yang akan digunakan. Anda harus menambahkan `.well-known/jwks.json` ke URL server otentikasi.

Tipe: String

Diperlukan: Ya

Contoh

Contoh berikut mendaftarkan server otentikasi eksternal dengan URL dari `https://my-auth-server.com/`

Perintah

```
sudo -u root dcv-session-manager-broker register-auth-server --url https://my-auth-server.com/.well-known/jwks.json
```

Keluaran

```
Jwk url registered.
```

list-auth-servers

Daftar server otentikasi eksternal yang telah terdaftar.

Topik

- [Sintaksis](#)
- [Output](#)
- [Contoh](#)

Sintaksis

```
sudo -u root dcv-session-manager-broker list-auth-servers
```

Output

Ur1s

Server otentikasi eksternal terdaftar. URLs

Contoh

Contoh berikut mencantumkan semua server otentikasi eksternal yang telah terdaftar.

Perintah

```
sudo -u root dcv-session-manager-broker list-auth-servers
```

Keluaran

```
Urls: [ "https://my-auth-server.com/.well-known/jwks.json" ]
```

unregister-auth-server

Membatalkan registrasi server otentikasi eksternal. Setelah Anda membatalkan pendaftaran server otentikasi eksternal, itu tidak dapat lagi digunakan untuk menghasilkan token akses OAuth 2.0.

Topik

- [Sintaksis](#)
- [Ops](#)
- [Output](#)
- [Contoh](#)

Sintaksis

```
sudo -u root dcv-session-manager-broker unregister-auth-server --url server_url.well-known/jwks.json
```

Ops

--url

URL server otentikasi eksternal untuk membatalkan pendaftaran. Anda harus menambahkan `.well-known/jwks.json` ke URL server otentikasi.

Tipe: String

Diperlukan: Ya

Output

Url

URL server otentikasi eksternal yang tidak terdaftar.

Contoh

Contoh berikut mendaftarkan server otentikasi eksternal dengan URL dari `https://my-auth-server.com/`

Perintah

```
sudo -u root dcv-session-manager-broker unregister-auth-server --url https://my-auth-server.com/.well-known/jwks.json
```

Keluaran

```
Jwk urlhttps://my-auth-server.com/.well-known/jwks.json unregistered
```

register-api-client

Mendaftarkan klien Session Manager dengan broker dan menghasilkan kredensi klien yang dapat digunakan oleh klien untuk mengambil token akses OAuth 2.0, yang diperlukan untuk membuat permintaan API.

Important

Pastikan Anda menyimpan kredensialnya di tempat yang aman. Mereka tidak dapat dipulihkan nanti.

Perintah ini hanya digunakan jika broker digunakan sebagai server otentikasi OAuth 2.0.

Topik

- [Sintaksis](#)
- [Ops](#)
- [Output](#)
- [Contoh](#)

Sintaksis

```
sudo -u root dcv-session-manager-broker register-api-client --client-name client_name
```

Opsi

--name

Nama unik yang digunakan untuk mengidentifikasi klien Session Manager.

Tipe: String

Diperlukan: Ya

Output

client-id

ID klien unik yang akan digunakan oleh klien Session Manager untuk mengambil token akses OAuth 2.0.

client-password

Kata sandi yang akan digunakan oleh klien Session Manager untuk mengambil token akses OAuth 2.0.

Contoh

Contoh berikut mendaftarkan klien bernama `my-sm-client`.

Perintah

```
sudo -u root dcv-session-manager-broker register-api-client --client-name my-sm-client
```

Keluaran

```
client-id: 21cfe9cf-61d7-4c53-b1b6-cf248EXAMPLE  
client-password: NjVmZDRlN2ItNjNmYS00M2QxLWF1ZmMtZmNmMDNkMEXAMPLE
```

describe-api-clients

Daftar klien Session Manager yang telah terdaftar di broker.

Topik

- [Sintaksis](#)
- [Output](#)
- [Contoh](#)

Sintaksis

```
sudo -u root dcv-session-manager-broker describe-api-clients
```

Output

name

Nama unik klien Session Manager.

id

ID unik klien Session Manager.

active

Menunjukkan status klien Session Manager. Jika klien aktif, nilainya adalah `true`; jika tidak, itu `false`.

Contoh

Contoh berikut mencantumkan klien Session Manager terdaftar.

Perintah

```
sudo -u root dcv-session-manager-broker describe-api-clients
```

Keluaran

```
Api clients
[ {
  "name" : "client-abc",
  "id" : "f855b54b-40d4-4769-b792-b727bEXAMPLE",
  "active" : false
}, {
  "name" : "client-xyz",
```

```
"id" : "21cfe9cf-61d7-4c53-b1b6-cf248EXAMPLE",  
"active" : true  
}]
```

unregister-api-client

Menonaktifkan klien Session Manager terdaftar. Klien Session Manager yang dinonaktifkan tidak dapat lagi menggunakan kredensialnya untuk mengambil token akses 2.0 OAuth .

Topik

- [Sintaksis](#)
- [Ops](#)
- [Contoh](#)

Sintaksis

```
sudo -u root dcv-session-manager-broker unregister-api-client --client-id client_id
```

Ops

--client -id

ID klien klien Session Manager untuk dinonaktifkan.

Tipe: String

Diperlukan: Ya

Contoh

Contoh berikut menonaktifkan klien Session Manager dengan ID klien dari. f855b54b-40d4-4769-b792-b727bEXAMPLE

Perintah

```
sudo -u root dcv-session-manager-broker unregister-api-client --client-id  
f855b54b-40d4-4769-b792-b727bEXAMPLE
```

Keluaran

```
Client f855b54b-40d4-4769-b792-b727bEXAMPLE unregistered.
```

renew-auth-server-api-kunci

Memperbarui kunci publik dan pribadi yang digunakan oleh broker untuk menandatangani token akses OAuth 2.0 yang dijual ke klien Session Manager. Jika Anda memperbarui kunci, maka Anda harus memberikan kunci pribadi baru kepada pengembang, karena diperlukan untuk membuat permintaan API.

Topik

- [Sintaksis](#)
- [Contoh](#)

Sintaksis

```
sudo -u root dcv-session-manager-broker renew-auth-server-api-key
```

Contoh

Contoh berikut memperbarui kunci publik dan pribadi.

Perintah

```
sudo -u root dcv-session-manager-broker renew-auth-server-api-key
```

Keluaran

```
Keys renewed.
```

generate-software-statement

Menghasilkan pernyataan perangkat lunak.

Agen harus terdaftar dengan broker untuk memungkinkan komunikasi. Agen membutuhkan pernyataan perangkat lunak untuk mendaftar dengan broker. Setelah agen memiliki pernyataan

perangkat lunak, ia dapat secara otomatis mendaftarkan dirinya dengan broker dengan menggunakan [Protokol Registrasi Klien Dinamis OAuth 2.0](#). Setelah agen terdaftar dengan broker, ia menerima ID klien dan rahasia klien yang digunakannya untuk mengotentikasi dengan broker.

Pialang dan agen menerima dan menggunakan pernyataan perangkat lunak default saat pertama kali diinstal. Anda dapat terus menggunakan pernyataan perangkat lunak default, atau Anda dapat memilih untuk menghasilkan yang baru. Jika Anda membuat pernyataan perangkat lunak baru, Anda harus menempatkan pernyataan perangkat lunak ke dalam file baru pada agen, dan kemudian menambahkan path file ke `agent.software_statement_path` parameter dalam `agent.conf` file. Setelah Anda melakukan ini, hentikan dan restart agen sehingga dapat menggunakan pernyataan perangkat lunak baru untuk mendaftar ke broker.

Topik

- [Sintaksis](#)
- [Output](#)
- [Contoh](#)

Sintaksis

```
sudo -u root dcv-session-manager-broker generate-software-statement
```

Output

software-statement

Pernyataan perangkat lunak.

Contoh

Contoh berikut menghasilkan pernyataan perangkat lunak.

Perintah

```
sudo -u root dcv-session-manager-broker generate-software-statement
```

Keluaran

```
software-statement:
```

```
ewogICJpZCIgOiAiYjc1NTVhN2QtNWI0MC00OTJhLWJjOTUtNmUzOWNhYzkyMDcxIiwKICAiYWN0aXZlIiA6IHRydWUsCi
```

describe-software-statements

Menjelaskan pernyataan perangkat lunak yang ada.

Topik

- [Sintaksis](#)
- [Output](#)
- [Contoh](#)

Sintaksis

```
sudo -u root dcv-session-manager-broker describe-software-statements
```

Output

software-statement

Pernyataan perangkat lunak.

issued-at

Tanggal dan waktu perangkat lunak dihasilkan.

is-active

Keadaan saat ini dari pernyataan perangkat lunak. `true` jika pernyataan perangkat lunak aktif; jika tidak, itu `false`.

Contoh

Contoh berikut menghasilkan pernyataan perangkat lunak.

Perintah

```
sudo -u root dcv-session-manager-broker describe-software-statements
```


Contoh

Contoh berikut menonaktifkan pernyataan perangkat lunak.

Perintah

```
sudo -u root dcv-session-manager-broker deactivate-software-statement --software-statement EXAMPLEpZCIg0iAiYjc1NTVhN2QtNWI0MC00TJhLWJjOTUtNmUzOWNhYzkxMDcxIiwKICAiaXNEXAMPLEQiIDogMTU5Nj
```

Keluaran

```
Software statement  
EXAMPLEpZCIg0iAiYjc1NTVhN2QtNWI0MC00TJhLWJjOTUtNmUzOWNhYzkxMDcxIiwKICAiaXNEXAMPLEQiIDogMTU5Nj  
deactivated
```

describe-agent-clients

Menjelaskan agen yang terdaftar di broker.

Topik

- [Sintaksis](#)
- [Output](#)
- [Contoh](#)

Sintaksis

```
sudo -u root dcv-session-manager-broker describe-agent-clients
```

Output

name

Nama Agen.

id

ID unik agen.

active

Keadaan agen. `true` jika agen aktif; jika tidak, itu `false`.

Contoh

Contoh berikut menjelaskan agen.

Perintah

```
sudo -u root dcv-session-manager-broker describe-agent-clients
```

Keluaran

```
Session manager agent clients
[ {
  "name" : "test",
  "id" : "6bc05632-70cb-4410-9e54-eaf9bEXAMPLE",
  "active" : true
}, {
  "name" : "test",
  "id" : "27131cc2-4c71-4157-a4ca-bde38EXAMPLE",
  "active" : true
}, {
  "name" : "test",
  "id" : "308dd275-2b66-443f-95af-33f63EXAMPLE",
  "active" : false
}, {
  "name" : "test",
  "id" : "ce412d1b-d75c-4510-a11b-9d9a3EXAMPLE",
  "active" : true
} ]
```

unregister-agent-client

Batalkan pendaftaran agen dari broker.

Topik

- [Sintaksis](#)
- [Ops](#)

- [Contoh](#)

Sintaksis

```
sudo -u root dcv-session-manager-broker unregister-agent-client --client-id client_id
```

Opsi

--client-id

ID agen untuk membatalkan pendaftaran.

Tipe: String

Diperlukan: Ya

Contoh

Contoh berikut membatalkan pendaftaran agen.

Perintah

```
sudo -u root dcv-session-manager-broker unregister-agent-client --client-id  
3b0d7b1d-78c7-4e79-b2e1-b976dEXAMPLE
```

Keluaran

```
agent client 3b0d7b1d-78c7-4e79-b2e1-b976dEXAMPLE unregistered
```

register-server-dns-mappings

Daftarkan Server DCV - Pemetaan nama DNS yang berasal dari file JSON.

Sintaksis

```
sudo -u root dcv-session-manager-broker register-server-dns-mappings --file-  
path file_path
```

Opsi

--file-path

Jalur file yang berisi Server DCV - pemetaan nama DNS.

Tipe: String

Diperlukan: Ya

Contoh

Contoh berikut mendaftarkan Server DCV - pemetaan nama DNS dari file `/tmp/mappings.json`.

Perintah

```
sudo -u root dcv-session-manager-broker register-server-dns-mappings --file-path /tmp/mappings.json
```

Keluaran

```
Successfully loaded 2 server id - dns name mappings from file /tmp/mappings.json
```

describe-server-dns-mappings

Jelaskan Server DCV yang tersedia saat ini - pemetaan nama DNS.

Sintaksis

```
sudo -u root dcv-session-manager-broker describe-server-dns-mappings
```

Output

serverIdType

Jenis ID server.

serverId

ID unik dari Server.

dnsNames

Nama DNS internal dan eksternal

internalDnsNames

Nama dns internal

externalDnsNames

Nama DNS eksternal

Contoh

Contoh berikut mencantumkan Server DCV terdaftar - pemetaan nama DNS.

Perintah

```
sudo -u root dcv-session-manager-broker describe-server-dns-mappings
```

Keluaran

```
[
{
  "serverIdType" : "Id",
  "serverId" : "192.168.0.1",
  "dnsNames" : {
    "internalDnsName" : "internal1",
    "externalDnsName" : "external1"
  }
},
{
  "serverIdType" : "Host.Aws.Ec2InstanceId",
  "serverId" : "i-0648aee30bc78bdff",
  "dnsNames" : {
    "internalDnsName" : "internal2",
    "externalDnsName" : "external2"
  }
}
]
```

Referensi File Konfigurasi

Bagian referensi ini memberikan ikhtisar opsi konfigurasi yang tersedia untuk Pengelola Sesi. Konfigurasi mencakup perubahan pada file agen dan broker. Setiap konfigurasi mencakup penjelasan tujuan, nilai yang diterima, dan dampak pada perilaku sistem secara keseluruhan. Amazon DCV Session Manager dapat disesuaikan untuk memenuhi persyaratan unik sistem Amazon DCV.

Topik

- [File konfigurasi broker](#)
- [File Konfigurasi Agen](#)

File konfigurasi broker

File konfigurasi broker (`/etc/dcv-session-manager-broker/session-manager-broker.properties`) mencakup parameter yang dapat dikonfigurasi untuk menyesuaikan fungsionalitas Manajer Sesi. Anda dapat mengedit file konfigurasi menggunakan editor teks pilihan Anda.

Note

`/etc/dcv-session-manager-broker/session-manager-broker.properties` berisi data sensitif. Secara default, akses tulisnya dibatasi untuk root dan akses bacanya dibatasi untuk root dan pengguna yang menjalankan broker. Secara default, ini adalah `dcvsmbroker` pengguna. Pialang memeriksa saat startup bahwa file tersebut memiliki izin yang diharapkan.

Tabel berikut mencantumkan parameter dalam file konfigurasi broker.

Nama parameter	Diperlukan	Nilai default	Deskripsi
<code>broker.java</code>	Tidak		Menentukan jalur ke direktori home Java yang akan digunakan broker

Nama parameter	Diperlukan	Nilai default	Deskripsi
va-home			<p>alih-alih default sistem. Jika diatur, broker akan menggunakan <code><broker-java-home>/bin/java</code> saat startup.</p> <p>Tip: broker memerlukan Java Runtime Environment 11 dan diinstal jika hilang sebagai ketergantungan pada instalasi yang berhasil. Jika versi 11 tidak ditetapkan sebagai lingkungan Java default, direktori home dapat diambil menggunakan perintah berikut:</p> <pre data-bbox="1117 1077 1507 1199">\$ sudo alternatives --display java</pre>
session-screenshot-max-width	Tidak	160	Menentukan lebar maksimum, dalam piksel, screenshot sesi yang diambil menggunakan API. <code>GetSessionScreenshots</code>

Nama parameter	Diperlukan	Nilai default	Deskripsi
<code>session-screenshots-max-height</code>	Tidak	100	Menentukan tinggi maksimum, dalam piksel, screenshot sesi yang diambil menggunakan API. <code>GetSessionScreenshots</code>
<code>session-screenshots-format</code>	Tidak	png	Format file gambar screenshot sesi yang diambil menggunakan <code>GetSessionScreenshotsAPI</code> .
<code>create-sessions-queue-max-size</code>	Tidak	1000	Jumlah maksimum permintaan <code>CreateSessionsAPI</code> yang tidak terpenuhi yang dapat diantrian. Ketika antrian penuh, permintaan baru yang tidak terpenuhi ditolak.
<code>create-sessions-queue-max-time-seconds</code>	Tidak	1800	Waktu maksimum, dalam hitungan detik, permintaan <code>CreateSessionsAPI</code> yang tidak terpenuhi dapat tetap berada dalam antrian. Jika permintaan tidak dapat dipenuhi dalam jumlah waktu yang ditentukan, itu gagal.

Nama parameter	Diperlukan	Nilai default	Deskripsi
session-manager-working-path	Ya	/tmp	Menentukan jalur ke direktori tempat broker menulis file yang diperlukan untuk beroperasi. Direktori ini harus dapat diakses hanya oleh broker.
enable-authorization-server	Ya	true	Menentukan apakah broker adalah server otentikasi yang digunakan untuk menghasilkan token akses OAuth 2.0 untuk klien APIs
enable-authorization	Ya	true	Mengaktifkan atau menonaktifkan otorisasi klien. jika Anda mengaktifkan otorisasi klien, API klien harus menyediakan token akses saat membuat permintaan API. Jika Anda menonaktifkan otorisasi klien, klien APIs dapat membuat permintaan tanpa token akses.
enable-agent-authorization	Ya	true	Mengaktifkan atau menonaktifkan otorisasi agen. Jika Anda mengaktifkan otorisasi agen, agen harus memberikan token akses saat berkomunikasi dengan broker.

Nama parameter	Diperlukan	Nilai default	Deskripsi
<code>delete-session-duration-hours</code>	Tidak	1	Menentukan jumlah jam setelah sesi yang dihapus menjadi tidak terlihat dan tidak lagi dikembalikan oleh panggilan <code>DescribeSession</code> API. Usang: <code>delete-session-duration-hours</code> ubah ke <code>delete-session-duration-seconds</code> — Tersedia sejak versi 2024.0-493.
<code>delete-session-duration-seconds</code>	Tidak	3600	Menentukan jumlah detik setelah sesi yang dihapus menjadi tidak terlihat dan tidak lagi dikembalikan oleh panggilan <code>DescribeSession</code> API. Parameter ini menggantikan <code>delete-session-duration-hours</code> parameter usang — Tersedia sejak versi 2024.0-493.
<code>connect-session-token-duration-minutes</code>	Tidak	60	Menentukan jumlah menit dimana <code>ConnectSession</code> token tetap valid.

Nama parameter	Diperlukan	Nilai default	Deskripsi
client-to-broker-connect-https-port	Ya	8443	Menentukan port HTTPS tempat broker mendengarkan koneksi klien.
client-to-broker-connect-bind-host	Tidak	0.0.0.0	Menentukan alamat IP host tempat broker mengikat koneksi klien.
client-to-broker-connect-key-store-file	Ya		Menentukan toko kunci yang digunakan untuk koneksi klien TLS.

Nama parameter	Diperlukan	Nilai default	Deskripsi
client-to-broker-connect-key-store-pass	Ya		Menentukan pass toko kunci.
agent-to-broker-connect-https-port	Ya	8445	Menentukan port HTTPS tempat broker mendengarkan koneksi agen.
agent-to-broker-connect-bind-host	Tidak	0.0.0.0	Menentukan alamat IP host tempat broker mengikat koneksi agen.

Nama parameter	Diperlukan	Nilai default	Deskripsi
agent-to-broker-connectokey-store-file	Ya		Menentukan toko kunci yang digunakan untuk koneksi agen TLS.
agent-to-broker-connectokey-store-pass	Ya		Menentukan pass toko kunci.
broker-to-broker-port	Ya	47100	Menentukan port yang digunakan untuk broker-to-broker koneksi.
broker-to-broker-bind-host	Tidak	0.0.0.0	Menentukan alamat IP host tempat broker mengikat koneksi. broker-to-broker

Nama parameter	Diperlukan	Nilai default	Deskripsi
broker-to-broker-discovery-port	Ya	47500	Menentukan port yang digunakan oleh broker untuk menemukan satu sama lain.
broker-to-broker-discovery-address	Tidak		Menentukan alamat IP dan port dari broker lain dalam armada dalam <i>port</i> format <i>ip_addresses</i> :. Jika ada beberapa broker, pisahkan nilainya dengan koma. Jika Anda menentukan <code>broker-to-broker-discovery-multicast-group</code> , <code>broker-to-broker-discovery-multicast-port</code> , <code>broker-to-broker-discovery-AWS-region</code> , <code>broker-to-broker-discovery-AWS-alb-target-group-arn</code> , hilangkan parameter ini.

Nama parameter	Diperlukan	Nilai default	Deskripsi
broker-to-broker-discovery-multicast-group	Tidak		Menentukan grup multicast untuk penemuan. broker-to-broker-discovery-multicast-group Jika Anda menentukan broker-to-broker-discovery-addresses ,broker-to-broker-discovery-aws-region ,ataubroker-to-broker-discovery-AWS-alb-target-group-arn , hilangkan parameter ini.
broker-to-broker-discovery-multicast-port	Tidak		Menentukan port multicast untuk penemuan. broker-to-broker-discovery-multicast-port Jika Anda menentukan broker-to-broker-discovery-addresses ,broker-to-broker-discovery-aws-region ,ataubroker-to-broker-discovery-AWS-alb-target-group-arn , hilangkan parameter ini.

Nama parameter	Diperlukan	Nilai default	Deskripsi
broker-to-broker-discovery-AWS-region	Tidak		Menentukan AWS Wilayah penyeimbang beban aplikasi yang digunakan untuk penemuan broker ke broker. Jika Anda menentukan broker-to-broker-discovery-multicast-group, broker-to-broker-discovery-multicast-port , atau broker-to-broker-discovery-addresses , hilangkan parameter ini.
broker-to-broker-discovery-AWS-alb-target-group-up-arn	Tidak		ARN dari pengguna grup target penyeimbang beban aplikasi untuk penemuan. broker-to-broker Jika Anda menentuka n broker-to-broker-discovery-multicast-group , broker-to-broker-discovery-multicast-port , atau broker-to-broker-discovery-addresses , hilangkan parameter ini.

Nama parameter	Diperlukan	Nilai default	Deskripsi
broker-to-broker-distributed-memory-max-size-mb	Tidak	4096	Menentukan jumlah maksimum memori off-heap yang akan digunakan oleh satu broker untuk menyimpan data sesi Amazon DCV.
broker-to-broker-key-store-file	Ya		Menentukan toko kunci yang digunakan untuk koneksi broker TLS.
broker-to-broker-key-store-pass	Ya		Menentukan pass toko kunci.
enable-cloud-watch-metrics	Tidak	false	Mengaktifkan atau menonaktifkan CloudWatch metrik Amazon. Jika Anda mengaktifkan CloudWatch Metrik, Anda mungkin perlu menentukan nilai untuk <code>cloud-watch-region</code> .

Nama parameter	Diperlukan	Nilai default	Deskripsi
<code>cloud-watch-region</code>	Tidak	Hanya diperlukan jika <code>enable-cloud-watch-metrics</code> disetel ke <code>true</code> . Jika broker diinstal pada instans Amazon EC2, wilayah tersebut diambil dari IMDS.	AWS Wilayah tempat CloudWatch metrik diposting.
<code>max-api-requests-per-second</code>	Tidak	1000	Menentukan jumlah maksimum permintaan yang dapat diproses oleh api broker setiap detik sebelum dibatasi.
<code>enable-throttling-forward-for-header</code>	Tidak	false	Jika diatur <code>true</code> ke throttling mengambil ip pemanggil dari header jika ada X-Forwarded-For.
<code>create-sessions-number-of-retries-on-failure</code>	Tidak	2	Menentukan jumlah maksimum percobaan ulang yang akan dilakukan setelah permintaan sesi pembuatan gagal pada host server Amazon DCV. Setel ke 0 untuk tidak pernah melakukan percobaan ulang pada kegagalan.

Nama parameter	Diperlukan	Nilai default	Deskripsi
<code>autorun-file-arguments-max-size</code>	Tidak	50	Menentukan jumlah maksimum argumen yang dapat diteruskan ke file <code>autorun</code> .
<code>autorun-file-arguments-max-argument-length</code>	Tidak	150	Menentukan panjang maksimum dalam karakter dari setiap argumen file <code>autorun</code> .
<code>enable-persistence</code>	Ya	false	Jika disetel ke <code>true</code> , data status broker disimpan pada database eksternal.
<code>persistence-db</code>	Tidak	Hanya diperlukan jika <code>enable-persistence</code> disetel ke <code>true</code> .	Menentukan database yang digunakan untuk persistensi. Satu-satunya nilai yang didukung adalah: <code>dynamodb</code> dan <code>mysql</code> .
<code>dynamodb-region</code>	Tidak	Hanya diperlukan jika <code>enable-persistence</code> disetel ke <code>true</code> dan <code>persistence-db</code> disetel ke <code>dynamodb</code> .	Menentukan wilayah di mana tabel DynamoDB dibuat dan diakses.

Nama parameter	Diperlukan	Nilai default	Deskripsi
<code>dynamoc-table-rcu</code>	Tidak	Hanya diperlukan jika <code>enable-persistence</code> disetel ke <code>true</code> dan <code>persistence-db</code> disetel ke <code>dynamodb</code> .	Menentukan Unit Kapasitas Baca (RCU) untuk setiap tabel DynamoDB Untuk informasi selengkapnya tentang RCU, lihat Harga untuk Kapasitas yang Disediakan.
<code>dynamoc-table-wcu</code>	Tidak	Hanya diperlukan jika <code>enable-persistence</code> disetel ke <code>true</code> dan <code>persistence-db</code> disetel ke <code>dynamodb</code> .	Menentukan Write Capacity Units (WCU) untuk setiap tabel DynamoDB. Untuk informasi selengkapnya tentang WCU, lihat Harga untuk Kapasitas yang Disediakan .
<code>dynamoc-table-name-prefix</code>	Tidak	Hanya diperlukan jika <code>enable-persistence</code> disetel ke <code>true</code> dan <code>persistence-db</code> disetel ke <code>dynamodb</code> .	Menentukan awalan yang ditambahkan ke setiap tabel DynamoDB (berguna untuk membedakan beberapa cluster broker menggunakan akun yang sama). AWS Hanya karakter alfanumerik, titik, tanda hubung, dan garis bawah yang diizinkan.

Nama parameter	Diperlukan	Nilai default	Deskripsi
<code>jdbc-connection-url</code>	Tidak	Hanya diperlukan jika <code>enable-persistence</code> disetel ke <code>true</code> dan <code>persistence-db</code> disetel ke <code>mysql</code> .	Menentukan URL koneksi ke MariaDB/MySQL database; itu berisi endpoint dan nama database. Url harus memiliki format ini: <div style="border: 1px solid #ccc; border-radius: 10px; padding: 10px; margin: 10px 0;"> <pre>jdbc:mysql://<db_endpoint>:<db_port>/<db_name>?createDatabaseIfNotExist=true</pre> </div> <p><db_endpoint> Dimana endpoint MariaDB/MySQL database, <db_port> adalah port database dan <db_name> merupakan nama database.</p>
<code>jdbc-user</code>	Tidak	Hanya diperlukan jika <code>enable-persistence</code> disetel ke <code>true</code> dan <code>persistence-db</code> disetel ke <code>mysql</code> .	Menentukan nama pengguna yang memiliki akses ke MariaDB/MySQL database.
<code>jdbc-password</code>	Tidak	Hanya diperlukan jika <code>enable-persistence</code> disetel ke <code>true</code> dan <code>persistence-db</code> disetel ke <code>mysql</code> .	Menentukan password dari pengguna yang memiliki akses ke MariaDB/MySQL database.

Nama parameter	Diperlukan	Nilai default	Deskripsi
seconds before deleting unreachable dcv-server	Tidak	1800	Menentukan jumlah detik setelah server Amazon DCV yang tidak terjangkau dihapus dari sistem.
seconds before deleting sessions-unreachable server	Tidak		Menentukan jumlah detik setelah sesi di server Amazon DCV yang tidak terjangkau dihapus dari sistem. Penghapusan sesi dari server yang tidak dapat dijangkau dinonaktifkan secara default. Untuk mengaktifkan penghapusan sesi dari server yang tidak dapat dijangkau, berikan nilai yang valid.

Nama parameter	Diperlukan	Nilai default	Deskripsi
session-screenshot-max-width	Tidak	160	Menentukan lebar maksimum, dalam piksel, screenshot sesi yang diambil menggunakan API. GetSessionScreenshots Jika <code>session-screenshot-max-width</code> diatur dalam file konfigurasi Klien Web , itu akan diutamakan dan mengesampingkan nilai default ini. Perhatikan bahwa ini adalah lebar maksimum, sehingga resolusi tangkapan layar sebenarnya mungkin lebih rendah.

Nama parameter	Diperlukan	Nilai default	Deskripsi
<code>session-screenshot-max-height</code>	Tidak	100	Menentukan tinggi maksimum, dalam piksel, screenshot sesi yang diambil menggunakan API. GetSessionScreenshots Jika <code>session-screenshot-max-height</code> diatur dalam file konfigurasi Klien Web , itu akan diutamakan dan mengesampingkan nilai default ini. Perhatikan bahwa ini adalah ketinggian maksimum, sehingga resolusi tangkapan layar sebenarnya mungkin lebih rendah.

File Konfigurasi Agen

File konfigurasi agen (`/etc/dcv-session-manager-agent/agent.conf` untuk Linux dan macOS, dan `C:\Program Files\NICE\DCVSessionManagerAgent\conf\agent.conf` untuk Windows) menyertakan parameter yang dapat dikonfigurasi untuk menyesuaikan fungsionalitas Session Manager. Anda dapat mengedit file konfigurasi menggunakan editor teks pilihan Anda.

Tabel berikut mencantumkan parameter dalam file konfigurasi agen.

Nama parameter	Diperlukan	Nilai default	Deskripsi
<code>agent.dns_server_host</code>	Ya		Menentukan nama DNS dari host broker.

Nama parameter	Diperlukan	Nilai default	Deskripsi
<code>agent.ker_port</code>	Ya	8445	Menentukan port di mana untuk berkomunikasi dengan broker.
<code>agent.config_file</code>	Tidak		Hanya diperlukan jika <code>tls_strict</code> diatur ke <code>true</code> . Menentukan jalur ke file sertifikat (.pem) yang diperlukan untuk memvalidasi sertifikat TLS. Salin sertifikat yang ditandatangani sendiri dari broker ke agen.
<code>agent.config_init_folder</code>	Tidak	<ul style="list-style-type: none"> <code>/var/lib/dcv-session-manager-agent/init</code> (Linux) 	Menentukan jalur ke folder di server host yang digunakan untuk menyimpan skrip kustom yang diizinkan untuk menginisialisasi sesi server Amazon DCV saat dibuat. Anda harus menentukan jalur absolut. Folder harus dapat diakses dan file harus dapat dieksekusi oleh pengguna yang menggunakan parameter <code>InitFilepermintaan API. CreateSessions</code>
<code>agent.config_tls_strict</code>	Tidak	<code>true</code>	Menunjukkan apakah validasi TLS yang ketat harus digunakan.

Nama parameter	Diperlukan	Nilai default	Deskripsi
agent_software_statement_path	Tidak		Hanya diperlukan jika pernyataan perangkat lunak default tidak digunakan. Menentukan path ke file pernyataan perangkat lunak. Untuk informasi selengkapnya, lihat generate-software-statement .
agent_sessions_folder	Tidak	<ul style="list-style-type: none"> /etc/dcv-session-manager-agent (Linux/macOS) C:\Program Files\NICE\DCVSessionManagerAgent\conf\tags (Jendela) 	Menentukan path ke folder di mana file tag berada. Untuk informasi selengkapnya, lihat Menggunakan tag untuk menargetkan server Amazon DCV .
agent_sessions_folder	Tidak	<ul style="list-style-type: none"> /var/lib/dcv-session-manager-agent/autorun (Linux) C:\ProgramData\NICE\DcvSessionManagerAgent\autorun (Jendela) 	Menentukan jalur ke folder di server host yang digunakan untuk menyimpan skrip dan aplikasi yang diizinkan untuk dijalankan secara otomatis saat sesi dimulai. Anda harus menentukan jalur absolut. Folder harus dapat diakses dan file harus dapat dieksekusi oleh pengguna yang menggunakan parameter AutorunFilePermintaan API. CreateSessions

Nama parameter	Diperlukan	Nilai default	Deskripsi
<code>agent.number_of_virtual_sessions</code>	Tidak	-1 (tanpa batas)	Jumlah maksimum sesi virtual yang dapat dibuat di server Amazon DCV menggunakan Amazon DCV Session Manager.
<code>agent.number_of_concurrent_sessions_per_user</code>	Tidak	1	Jumlah maksimum sesi virtual yang dapat dibuat di server Amazon DCV oleh satu pengguna menggunakan Amazon DCV Session Manager.

Nama parameter	Diperlukan	Nilai default	Deskripsi
agent.ker_upc e_inte l	Tidak	30	Menentukan jumlah detik untuk menunggu sebelum mengirim data yang diperbarui ke broker. Data terkirim mencakup server Amazon DCV dan status host, serta informasi sesi yang diperbarui. Nilai yang lebih rendah membuat Session Manager lebih sadar akan perubahan yang terjadi pada sistem tempat agen berjalan, tetapi meningkatkan beban sistem dan lalu lintas jaringan. Nilai yang lebih tinggi menurunkan beban sistem dan jaringan, tetapi Manajer Sesi menjadi kurang responsif terhadap perubahan sistem, sehingga nilai lebih tinggi daripada 120 yang tidak disarankan.
agent.e ble_que _logge n_users	Tidak	true	Aktifkan mendapatkan pengguna yang masuk di Windows. Ketika benar, sistem kueri untuk pengguna login aktual. Ketika salah, mengembalikan "tidak diketahui"

Nama parameter	Diperlukan	Nilai default	Deskripsi
agent.preferred_network_interface	Tidak		Menentukan nama antarmuka jaringan yang akan digunakan untuk deteksi IP host. Contoh: eth0 (Linux), Ethernet 4 (Windows). Linux/macOS nama peka huruf besar/kecil, nama Windows tidak peka huruf besar/kecil. Jika tidak ditentukan atau jika antarmuka jaringan yang ditentukan tidak ditemukan, antarmuka non-loopback pertama dipilih.

Nama parameter	Diperlukan	Nilai default	Deskripsi
log.level	Tidak	info	<p>Menentukan tingkat verbositas file log. Tingkat verbositas berikut tersedia:</p> <ul style="list-style-type: none"> • <code>error</code>—Memberikan detail paling sedikit. Termasuk kesalahan saja. • <code>warning</code>—Termasuk kesalahan dan peringatan. • <code>info</code>—Tingkat verbositas default. Termasuk kesalahan, peringatan, dan pesan informasi. • <code>debug</code>—Memberikan detail paling banyak. Memberikan informasi rinci yang berguna untuk masalah debugging.
log.directory	Tidak	<ul style="list-style-type: none"> • <code>/var/log/dcv-session-manager-agent/(Linux/macOS)</code> • <code>C:\ProgramData\NICE\DCVSessionManagerAgent\log</code> (Jendela) 	Menentukan direktori di mana untuk membuat file log.

Nama parameter	Diperlukan	Nilai default	Deskripsi
log.rotation	Tidak	daily	Menentukan rotasi file log. Nilai yang valid adalah: <ul style="list-style-type: none"> hourly—File log diputar setiap jam. daily—File log diputar setiap hari.
log.maxfile-size	Tidak	10485760	Ketika ukuran file log mencapai ukuran yang ditentukan dalam byte, itu akan diputar. File log baru akan dibuat dan peristiwa log lebih lanjut akan ditempatkan di file baru.
log.rotate	Tidak	9	Jumlah maksimum file log yang diawetkan dalam rotasi. Setiap kali rotasi terjadi dan nomor ini tercapai, file log tertua akan dihapus.

Catatan rilis dan riwayat dokumen untuk Amazon DCV Session Manager

Halaman ini menyediakan catatan rilis dan riwayat dokumen untuk Amazon DCV Session Manager.

Topik

- [Catatan rilis Amazon DCV Session Manager](#)
- [Riwayat dokumen](#)

Catatan rilis Amazon DCV Session Manager

Bagian ini memberikan gambaran umum tentang pembaruan utama, rilis fitur, dan perbaikan bug untuk Amazon DCV Session Manager. Semua pembaruan diatur berdasarkan tanggal rilis. Kami sering memperbarui dokumentasi untuk mengatasi umpan balik yang Anda kirimkan kepada kami.

Topik

- [2025.0-544— 2 Februari 2026](#)
- [2025.0-544— 23 Desember 2025](#)
- [2025.0-539— 12 November 2025](#)
- [2025.0-539— 22 Oktober 2025](#)
- [2024.0-531— 17 Juni 2025](#)
- [2024.0-504— 31 Maret 2025](#)
- [2024.0-493— 15 Januari 2025](#)
- [2024.0-457— 1 Oktober 2024](#)
- [2023.1-17652— 1 Agustus 2024](#)
- [2023.1-16388— 26 Juni 2024](#)
- [2023.1— November 9, 2023](#)
- [2023.0-15065— 4 Mei 2023](#)
- [2023.0-14852— 28 Maret 2023](#)
- [2022.2-13907— November 11, 2022](#)
- [2022.1-13067— 29 Juni 2022](#)
- [2022.0-11952— 23 Februari 2022](#)

- [2021.3-11591— Desember 20, 2021](#)
- [2021.2-11445— November 18, 2021](#)
- [2021.2-11190— 11 Oktober 2021](#)
- [2021.2-11042— September 01, 2021](#)
- [2021.1-10557— 31 Mei 2021](#)
- [2021.0-10242— 12 April 2021](#)
- [2020.2-9662— 04 Desember 2020](#)
- [2020.2-9508— 11 November 2020](#)

2025.0-544— 2 Februari 2026

Membangun angka	Perubahan dan perbaikan bug
<ul style="list-style-type: none"> • Pialang: 544 • Agen: 902 • CLI: 159 	<ul style="list-style-type: none"> • Ditambahkan parameter <code>preferred_network_interface</code> konfigurasi untuk deteksi IP host.

2025.0-544— 23 Desember 2025

Membangun angka	Perubahan dan perbaikan bug
<ul style="list-style-type: none"> • Pialang: 544 • Agen: 893 • CLI: 159 	<ul style="list-style-type: none"> • Peningkatan batas ukuran WebSocket pesan untuk mengatasi kegagalan pengambilan tangkapan layar pada host macOS. • Lingkungan build Windows yang diperbarui ke Visual Studio 2022.

2025.0-539— 12 November 2025

Membangun angka	Perubahan dan perbaikan bug
<ul style="list-style-type: none"> • Pialang: 539 • Agen: 888 	<ul style="list-style-type: none"> • Pengenal bundel Agen macOS yang diganti merek dari Perangkat Lunak NICE ke Amazon.

Membangun angka	Perubahan dan perbaikan bug
<ul style="list-style-type: none"> • CLI: 159 	

2025.0-539— 22 Oktober 2025

Membangun angka	Perubahan dan perbaikan bug
<ul style="list-style-type: none"> • Pialang: 539 • Agen: 886 • CLI: 159 	<ul style="list-style-type: none"> • Menambahkan parameter konfigurasi <code>enable_query_logged_in_users</code> ke file konfigurasi Agen untuk menentukan perilaku kueri pengguna yang dicatat pada sistem Windows. • Mengganti PowerShell perintah dengan Windows asli APIs (WMI dan Windows Registry) untuk meningkatkan kinerja dan keandalan saat mengambil informasi sistem. • Memperbaiki resolusi nama DNS pada instans Windows Amazon EC2 dengan meningkatkan deteksi Amazon EC2 dengan fallback ke AWS layanan metadata saat deteksi berbasis UUID gagal. • Versi yang diperbarui ke 2025.

2024.0-531— 17 Juni 2025

Membangun angka	Perubahan dan perbaikan bug
<ul style="list-style-type: none"> • Pialang: 531 • Agen: 852 • CLI: 154 	<ul style="list-style-type: none"> • Menambahkan fitur untuk memperbarui sertifikat sebelum kedaluwarsa. • Berganti merek NICE DCV menjadi Amazon DCV. • Perbaikan bug.

2024.0-504— 31 Maret 2025

Membangun angka	Perubahan dan perbaikan bug
<ul style="list-style-type: none"> • Pialang: 504 	<ul style="list-style-type: none"> • Menambahkan dukungan untuk AL2023.

Membangun angka	Perubahan dan perbaikan bug
<ul style="list-style-type: none"> • Agen: 817 • CLI: 154 	<ul style="list-style-type: none"> • Perbaikan bug dan peningkatan performa.

2024.0-493— 15 Januari 2025

Membangun angka	Perubahan dan perbaikan bug
<ul style="list-style-type: none"> • Pialang: 493 • Agen: 801 • CLI: 152 	<ul style="list-style-type: none"> • Menambahkan parameter ke <code>GetSessionScreenshot</code> permintaan untuk menentukan tinggi dan lebar maksimum tangkapan layar. • Parameter yang ditambahkan ke file konfigurasi Broker yang menentukan jumlah detik setelah sesi pada server Amazon DCV yang tidak terjangkau dihapus dari sistem. • Memperbaiki masalah di mana <code>seconds-before-deleting-unreachable-dcv-server</code> parameter dalam file konfigurasi Broker tidak dihormati. • Perbaikan bug dan peningkatan performa.

2024.0-457— 1 Oktober 2024

Membangun angka	Perubahan dan perbaikan bug
<ul style="list-style-type: none"> • Pialang: 457 • Agen: 748 • CLI: 140 	<ul style="list-style-type: none"> • Berganti merek NICE DCV menjadi Amazon DCV. • Menambahkan dukungan untuk Ubuntu 24.04.

2023.1-17652— 1 Agustus 2024

Membangun angka	Perubahan dan perbaikan bug
<ul style="list-style-type: none"> • Pialang: 426 • Agen: 748 	<ul style="list-style-type: none"> • Perbaikan bug dan peningkatan performa.

Membangun angka	Perubahan dan perbaikan bug
<ul style="list-style-type: none">• CLI: 140	

2023.1-16388— 26 Juni 2024

Membangun angka	Perubahan dan perbaikan bug
<ul style="list-style-type: none">• Pialang: 417• Agen: 748• CLI: 140	<ul style="list-style-type: none">• Memperbaiki bug yang menunjukkan memori secara tidak benar sebagai TB, bukan GB.• Perbaiki bug dan peningkatan performa.

2023.1— November 9, 2023

Membangun angka	Perubahan dan perbaikan bug
<ul style="list-style-type: none">• Pialang: 410• Agen: 732• CLI: 140	<ul style="list-style-type: none">• Perbaiki bug dan peningkatan kinerja

2023.0-15065— 4 Mei 2023

Membangun angka	Perubahan dan perbaikan bug
<ul style="list-style-type: none">• Pialang: 392• Agen: 675• CLI: 132	<ul style="list-style-type: none">• Menambahkan dukungan untuk Red Hat Enterprise Linux 9, Rocky Linux 9, dan CentOS Stream 9 pada platform ARM.

2023.0-14852— 28 Maret 2023

Membangun angka	Perubahan dan perbaikan bug
<ul style="list-style-type: none">• Pialang: 392• Agen: 642• CLI: 132	<ul style="list-style-type: none">• Menambahkan dukungan untuk Red Hat Enterprise Linux 9, Rocky Linux 9, dan CentOS Stream 9.

2022.2-13907— November 11, 2022

Membangun angka	Perubahan dan perbaikan bug
<ul style="list-style-type: none">• Pialang: 382• Agen: 612• CLI: 123	<ul style="list-style-type: none">• Menambahkan Substate bidang DescribeSessions sebagai tanggapan.• Memperbaiki masalah yang dapat menyebabkan CLI gagal terhubung ke broker tergantung pada URL yang digunakan.

2022.1-13067— 29 Juni 2022

Membangun angka	Perubahan dan perbaikan bug
<ul style="list-style-type: none">• Pialang: 355• Agen: 592• CLI: 114	<ul style="list-style-type: none">• Menambahkan dukungan untuk menjalankan broker pada instance AWS Graviton.• Menambahkan dukungan agen dan broker untuk Ubuntu 22.04.

2022.0-11952— 23 Februari 2022

Membangun angka	Perubahan dan perbaikan bug
<ul style="list-style-type: none">• Pialang: 341• Agen: 520	<ul style="list-style-type: none">• Menambahkan kemampuan rotasi log ke Agen.

Membangun angka	Perubahan dan perbaikan bug
<ul style="list-style-type: none">• CLI: 112	<ul style="list-style-type: none">• Ditambahkan parameter konfigurasi untuk mengatur rumah Java di Broker.• Peningkatan pembilasan data dari cache ke disk di Broker.• Validasi URL tetap di CLI.

2021.3-11591— Desember 20, 2021

Membangun angka	Fitur baru
<ul style="list-style-type: none">• Pialang: 307• Agen: 453• CLI: 92	<ul style="list-style-type: none">• Menambahkan dukungan untuk mengintegrasikan dengan Amazon DCV Connection Gateway.• Menambahkan dukungan Broker untuk Ubuntu 18.04 dan Ubuntu 20.04.

2021.2-11445— November 18, 2021

Membangun angka	Perubahan dan perbaikan bug
<ul style="list-style-type: none">• Pialang: 288• Agen: 413• CLI: 54	<ul style="list-style-type: none">• Memperbaiki masalah dengan validasi nama login yang mencakup domain Windows.

2021.2-11190— 11 Oktober 2021

Membangun angka	Perubahan dan perbaikan bug
<ul style="list-style-type: none">• Pialang: 254• Agen: 413• CLI: 54	<ul style="list-style-type: none">• Memperbaiki masalah di antarmuka baris perintah yang mencegah peluncuran sesi Windows.

2021.2-11042— September 01, 2021

Membangun angka	Fitur baru	Perubahan dan perbaikan bug
<ul style="list-style-type: none"> • Pialang: 254 • Agen: 413 • CLI: 37 	<ul style="list-style-type: none"> • Amazon DCV Session Manager sekarang menawarkan dukungan antarmuka baris perintah (CLI). Anda dapat membuat dan mengelola sesi Amazon DCV di CLI, alih-alih menelepon. APIs • Amazon DCV Session Manager memperkenalkan ketekunan data Broker. Untuk ketersediaan yang lebih tinggi, broker dapat mempertahankan informasi status server pada penyimpanan data eksternal dan memulihkan data saat startup. 	<ul style="list-style-type: none"> • Saat mendaftarkan server otorisasi eksternal, Anda sekarang dapat menentukan algoritme yang digunakan server otorisasi untuk menandatangani Token Web berformat JSON. Dengan perubahan ini, Anda dapat menggunakan Azure AD sebagai server otorisasi eksternal.

2021.1-10557— 31 Mei 2021

Membangun angka	Fitur baru	Perubahan dan perbaikan bug
<ul style="list-style-type: none"> • Pialang: 214 • Agen: 365 	<ul style="list-style-type: none"> • Amazon DCV Session Manager menambahkan dukungan untuk parameter input yang diteruskan ke file autorun di Linux. • Properti server sekarang dapat diteruskan sebagai persyaratan ke CreateSessionsAPI. 	<ul style="list-style-type: none"> • Kami memperbaiki masalah dengan file autorun di Windows.

2021.0-10242— 12 April 2021

Membangun angka	Perubahan dan perbaikan bug
<ul style="list-style-type: none"> Pialang: 183 Agen: 318 	<ul style="list-style-type: none"> Amazon DCV Session Manager memperkenalkan yang baru APIs berikut ini: <ul style="list-style-type: none"> OpenServers CloseServers DescribeServers GetSessionScreenshots Ini juga memperkenalkan parameter konfigurasi baru berikut: <ul style="list-style-type: none"> Parameter broker: <code>session-screenshot-max-width</code> , <code>session-screenshot-max-height</code> , <code>session-screenshot-format</code> , <code>create-sessions-queue-max-size</code> , dan <code>create-sessions-queue-max-time-seconds</code> . Parameter agen: <code>agent.autorun_folder</code> , <code>max_virtual_sessions</code> , dan <code>max_concurrent_sessions_per_user</code> . Parameter agen: <code>agent.autorun_folder</code> , <code>max_virtual_sessions</code> , dan <code>max_concurrent_sessions_per_user</code> . Parameter agen: <code>agent.autorun_folder</code> , <code>max_virtual_sessions</code> , dan <code>max_concurrent_sessions_per_user</code> .

2020.2-9662— 04 Desember 2020

Membangun angka	Perubahan dan perbaikan bug
<ul style="list-style-type: none"> Pialang: 114 Agen: 211 	<ul style="list-style-type: none"> Kami memperbaiki masalah dengan sertifikat TLS yang dibuat secara otomatis yang mencegah Broker memulai.

2020.2-9508— 11 November 2020

Membangun angka	Perubahan dan perbaikan bug
<ul style="list-style-type: none"> Pialang: 78 Agen: 183 	<ul style="list-style-type: none"> Rilis awal Amazon DCV Session Manager.

Riwayat dokumen

Tabel berikut menjelaskan dokumentasi untuk rilis Amazon DCV Session Manager ini.

Ubah	Deskripsi	Date
Amazon DCV Versi 2025.0-544	Amazon DCV Session Manager telah diperbarui untuk Amazon DCV 2025.0-544. Untuk informasi selengkapnya, lihat 2025.0-544— 2 Februari 2026 .	Februari 2, 2026
Amazon DCV Versi 2025.0-544	Amazon DCV Session Manager telah diperbarui untuk Amazon DCV 2025.0-544. Untuk informasi selengkapnya, lihat 2025.0-544— 23 Desember 2025 .	Desember 23, 2025
Amazon DCV Versi 2025.0-539	Amazon DCV Session Manager telah diperbarui untuk Amazon DCV 2025.0-539. Untuk informasi selengkapnya, lihat 2025.0-539— 12 November 2025 .	November 12, 2025
Amazon DCV Versi 2025.0-539	Amazon DCV Session Manager telah diperbarui untuk Amazon DCV 2025.0-539. Untuk informasi selengkapnya, lihat 2025.0-539— 22 Oktober 2025 .	Oktober 22, 2025

Ubah	Deskripsi	Date
Amazon DCV Versi 2024.0-531	Amazon DCV Session Manager telah diperbarui untuk Amazon DCV 2024.0-531. Untuk informasi selengkapnya, lihat 2024.0-531— 17 Juni 2025 .	Juni 17, 2025
Amazon DCV Versi 2024.0-504	Amazon DCV Session Manager telah diperbarui untuk Amazon DCV 2024.0-504. Untuk informasi selengkapnya, lihat 2024.0-504— 31 Maret 2025 .	Maret 31, 2025
Amazon DCV Versi 2024.0-493	Amazon DCV Session Manager telah diperbarui untuk Amazon DCV 2024.0-493. Untuk informasi selengkapnya, lihat 2024.0-493— 15 Januari 2025 .	Januari 15, 2025
Amazon DCV Versi 2024.0-457	Amazon DCV Session Manager telah diperbarui untuk Amazon DCV 2024.0-457. Untuk informasi selengkapnya, lihat 2024.0-457— 1 Oktober 2024 .	September 30, 2024
Amazon DCV Versi 2023.1-17652	Amazon DCV Session Manager telah diperbarui untuk Amazon DCV 2023.1-17652. Untuk informasi selengkapnya, lihat 2023.1-17652— 1 Agustus 2024 .	Agustus 1, 2024
Amazon DCV Versi 2023.1-16388	Amazon DCV Session Manager telah diperbarui untuk Amazon DCV 2023.1-16388. Untuk informasi selengkapnya, lihat 2023.1-16388— 26 Juni 2024 .	Juni 26, 2024

Ubah	Deskripsi	Date
Amazon DCV Versi 2023.1	Amazon DCV Session Manager telah diperbarui untuk Amazon DCV 2023.1. Untuk informasi selengkapnya, lihat 2023.1— November 9, 2023 .	9 November 2023
Amazon DCV Versi 2023.0	Amazon DCV Session Manager telah diperbarui untuk Amazon DCV 2023.0. Untuk informasi selengkapnya, lihat 2023.0-14852— 28 Maret 2023 .	Maret 28, 2023
Amazon DCV Versi 2022.2	Amazon DCV Session Manager telah diperbarui untuk Amazon DCV 2022.2. Untuk informasi selengkapnya, lihat 2022.2-13907— November 11, 2022 .	11 November 2022
Amazon DCV Versi 2022.1	Amazon DCV Session Manager telah diperbarui untuk Amazon DCV 2022.1. Untuk informasi selengkapnya, lihat 2022.1-13067— 29 Juni 2022 .	Juni 29, 2022
Amazon DCV Versi 2022.0	Amazon DCV Session Manager telah diperbarui untuk Amazon DCV 2022.0. Untuk informasi selengkapnya, lihat 2022.0-11952— 23 Februari 2022 .	Februari 23, 2022
Amazon DCV Versi 2021.3	Amazon DCV Session Manager telah diperbarui untuk Amazon DCV 2021.3. Untuk informasi selengkapnya, lihat 2021.3-11591— Desember 20, 2021 .	Desember 20, 2021
Amazon DCV Versi 2021.2	Amazon DCV Session Manager telah diperbarui untuk Amazon DCV 2021.2. Untuk informasi selengkapnya, lihat 2021.2-11042— September 01, 2021 .	September 01, 2021

Ubah	Deskripsi	Date
Amazon DCV Versi 2021.1	Amazon DCV Session Manager telah diperbarui untuk Amazon DCV 2021.1. Untuk informasi selengkapnya, lihat 2021.1-10557— 31 Mei 2021 .	Mei 31, 2021
Amazon DCV Versi 2021.0	Amazon DCV Session Manager telah diperbarui untuk Amazon DCV 2021.0. Untuk informasi selengkapnya, lihat 2021.0-10242— 12 April 2021 .	12 April 2021
Rilis awal Amazon DCV Session Manager	Publikasi pertama dari konten ini.	11 November 2020

Terjemahan disediakan oleh mesin penerjemah. Jika konten terjemahan yang diberikan bertentangan dengan versi bahasa Inggris aslinya, utamakan versi bahasa Inggris.