Citrix Session Remote Start

Contents

Introduction

October 10, 2025

For organizations using virtual apps and desktops, productivity starts after logging in. However, prolonged logon times, often lasting several minutes, disrupt workflows and disappoint the employees. Citrix Session Remote Start addresses this challenge by delivering seamless, efficient, and faster access to virtualized resources.

Session Remote Start provides APIs for trusted third-party services to enumerate, launch, and log off Citrix sessions. It enables unattended logons triggered by events like building badge scans, eliminating delays in time-intensive environments. Optional logon scripts can disconnect sessions post-logon, keeping them ready for users to reconnect when needed.

With seamless integration into existing Citrix components, Session Remote Start simplifies deployment, enhances user experiences, and redefines how businesses manage virtualized access boosting the overall productivity.

SRS Server System Requirements

The following table lists the minimum requirements for a Session Remote Start server.

Requirements	Details
OS	Windows Server OS - recommended 2022 and above
Processor	4 or more cores on a compatible 64-bit processor with 2 GHz or faster
RAM	Min 16GB
Storage	50

Session Remote Start - Prerequisites

The Session Remote Start requires the following components:

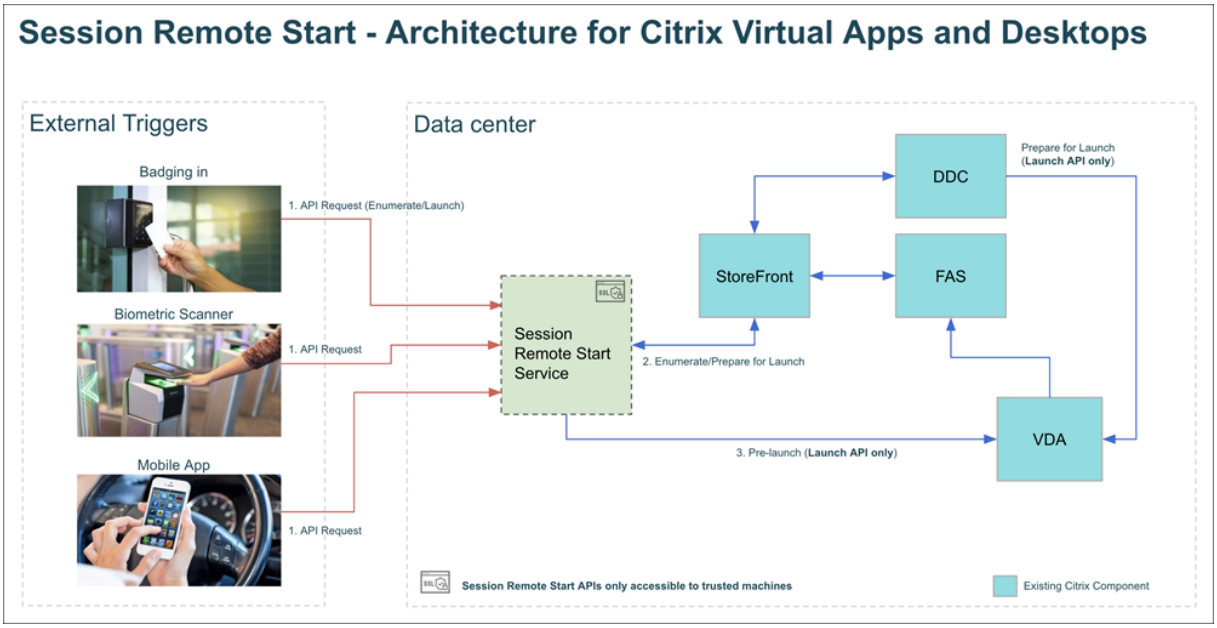
1. **Windows Active Directory (AD)** or Microsoft Entra hybrid-joined - Use an existing server or service.
2. Citrix StoreFront 2311 or later - Use an existing server.
3. Citrix FAS (Federated Authentication Service) - Use an existing server, or set up a new one if unavailable.

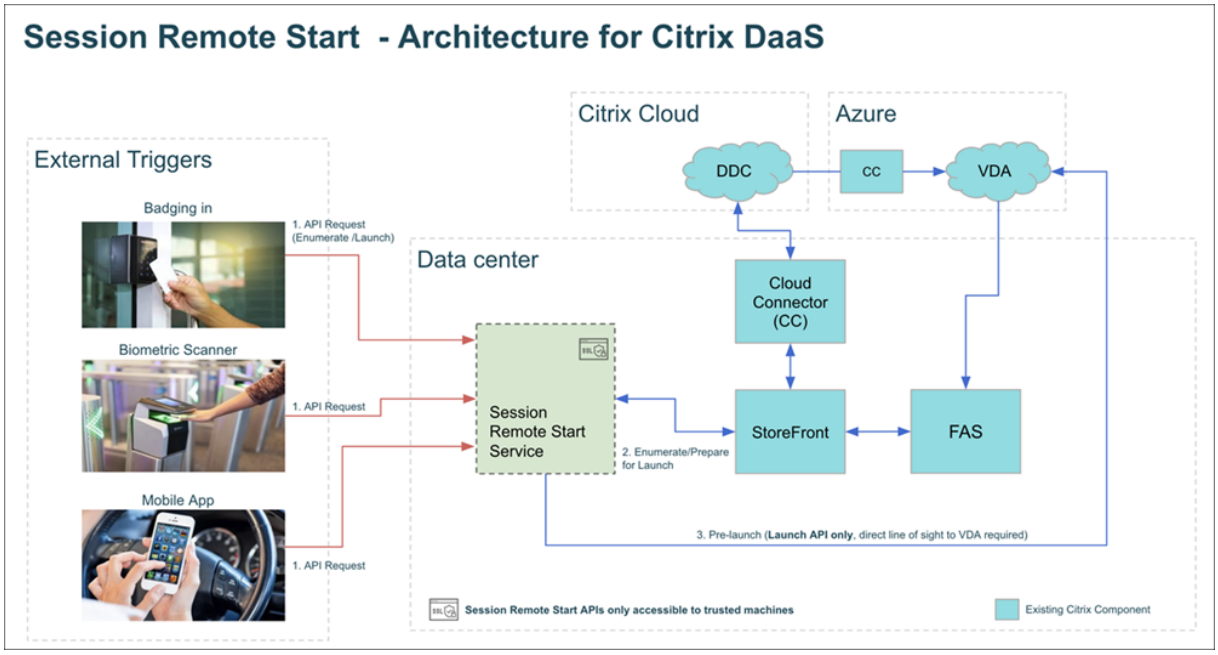
4. **StoreFront™ Store Configuration** - Create a dedicated store on the Citrix StoreFront server exclusively for Session Remote Start.
5. **IIS Requirement** - IIS is required on the Session Remote Start server. If it is not installed already, Session Remote Start installs it automatically (an internet connection is required).
6. **SSL Certificates** - install SSL certificates on both, the Session Remote Start server and the servers making API requests to ensure that only trusted services can issue requests.
7. **Connectivity requirements** - Session Remote Start must have a direct line-of-sight to Citrix StoreFront and the VDAs intended for pre-launch.

> **Important Note:**
>
> - Session Remote Start is compatible with Citrix Virtual Apps and Desktops™ and Citrix DaaS, but only when StoreFront is used.
> - Currently, Session Remote Start does not support Citrix Workspace™ (the cloud version of StoreFront).
> - Citrix FAS and StoreFront are a mandatory requirements for Session Remote Start.

Architecture: Citrix Virtual Apps and Desktops and Citrix DaaS

You can deploy Session Remote Start on both Citrix Virtual Apps and Desktops and Citrix DaaS™ environments as long as you use Citrix StoreFront in DaaS instead of Citrix Workspace. Here are the high-level architecture diagrams that explain the data flow, user action, and Session Remote Start workflow.





Installation of Session Remote Start

March 26, 2025

Installation Steps

- Download the latest Session Remote Start installation package from Citrix Downloads.
- The installation package has five folders, each containing the necessary files for deployment on four different servers respectively.
- Open the **SessionRemoteStart** folder, and double-click `SessionRemoteStartSetup.exe`.
- Ensure to install as an administrator and follow the on-screen instructions to complete the installation.

> **Note:**
>
> If the installation fails, the completion page displays the location of the failure log. Open the log file to review the installation failure details.

Certificates

September 7, 2025

Install SSL Certificate for Accessing StoreFront™

During the use of Session Remote Start, a large number of calls to StoreFront's API are required. Therefore, we need to install StoreFront's certificate on the Session Remote Start server to ensure that Session Remote Start can successfully access StoreFront's Web APIs.

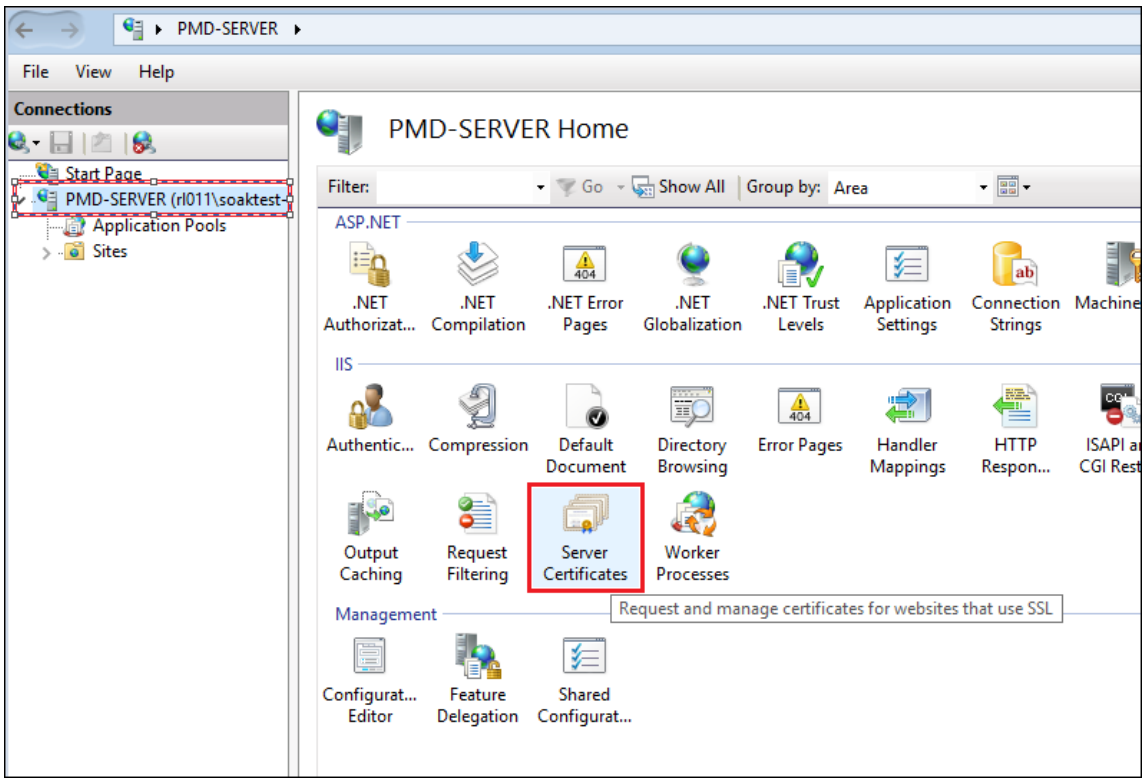
> **Note:**
>
> - Session Remote Start is hosted on **IIS** and runs under a different identity than the user installing the certificate. Ensure that the **Session Remote Start** service has the necessary permissions to load the certificate.
> - Citrix® recommends installing the certificate under **Local Machine** to allow access for all users.
> - The **IIS identity** under which Session Remote Start is running must be able to visit the StoreFront URL of **Receiver for Web Site** without warning. (For example, `https://storefront.rl011.local/Citrix/srsWeb`)

Import SSL certificate for Session Remote Start to IIS Manager

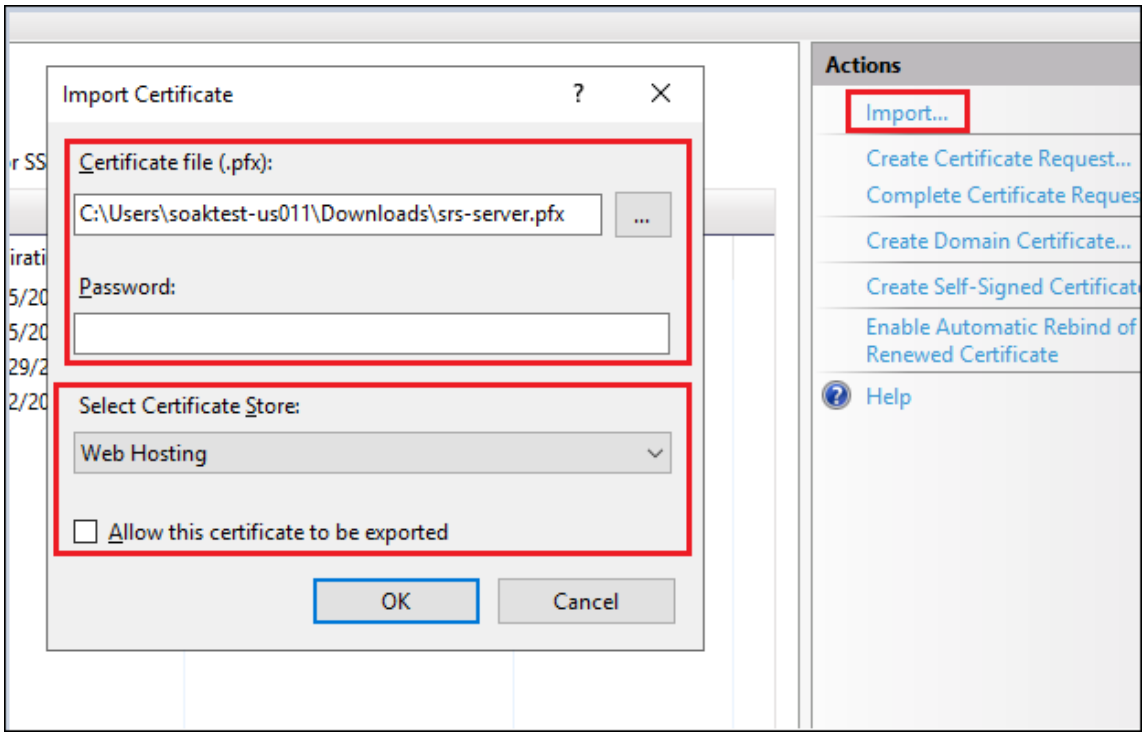
> **Note:**
>
> Ignore this step if your Default Web Site is already configured.

Securing access and encrypting traffic with SSL certificates is the preferred way of deploying Session Remote Start. Follow the steps below to achieve:

1. Open up IIS Manager, select the Session Remote Start Server name, and open the **Server Certificates**.



2. Click **Import...** in the **Actions** panel on the right.



The settings shown for **Select Certificate store** and **Allow this certificate to be exported** in the images are recommended for enhanced security. If **Web Hosting** is selected, ensure to import the full

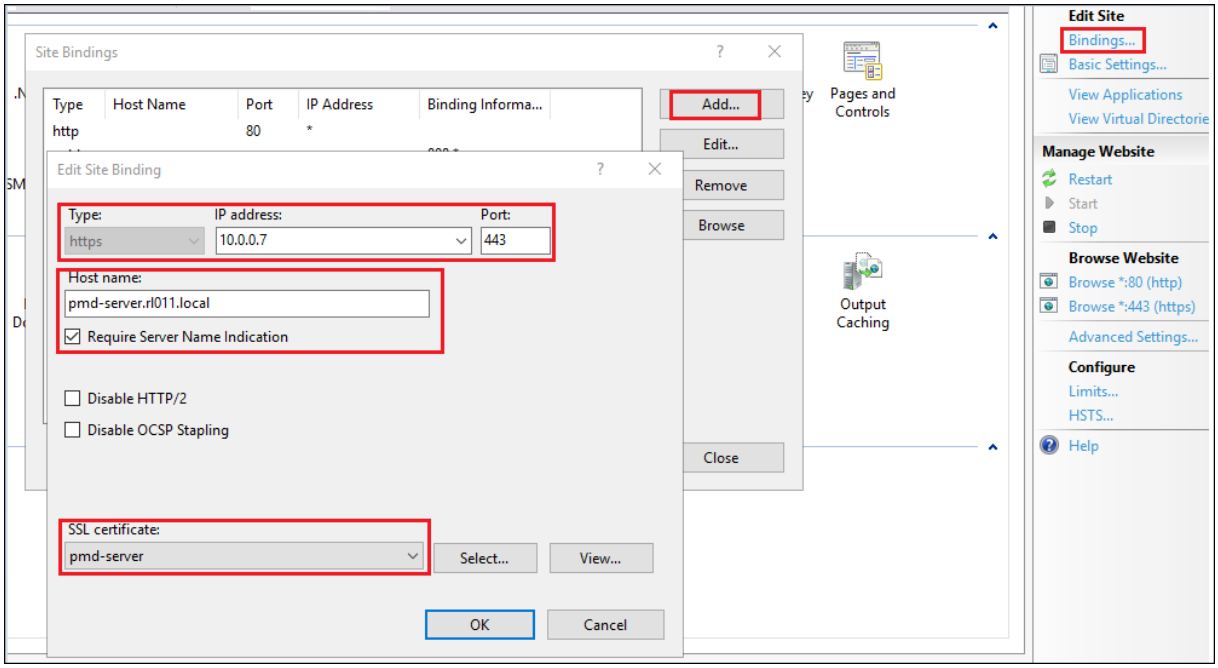
certificate chain.

Also, ensure that the Session Remote Start service IIS identity has the necessary permissions to load the certificate.

Create HTTPS Binding

> **Note:**
>
> Ignore this step if your Default Web Site is already configured.

Create HTTPS binding in IIS Manager.

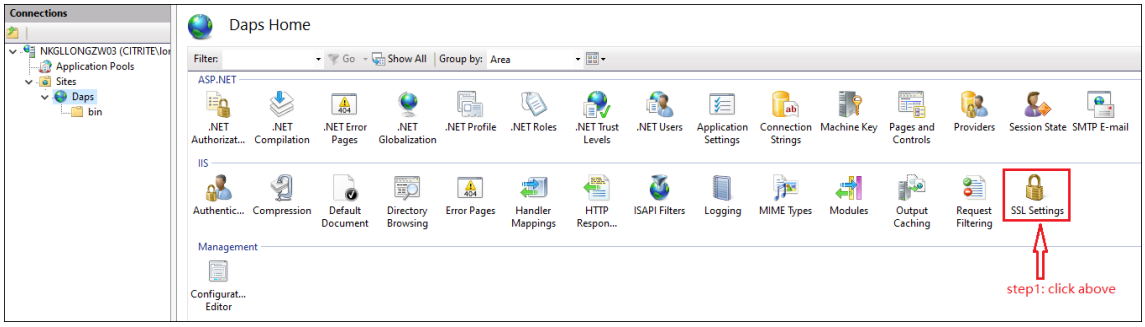


1. Open the IIS Manager, click **Default Web Site** under **Sites**.
2. On the right panel, under **Actions > Edit Site**, click **Bindings**.
3. Click **Add**.
4. Under the **Add Site Binding** screen.
    - Select the type as **https**.
    - Set https port to **443**.
    - Enter the IP address and host name of Session Remote Start server respectively.
    - Start server respectively in the respective fields.
    - Click **OK**.
    - Click **Edit** on the newly created Binding.
    - Select the **SSL certificate**.

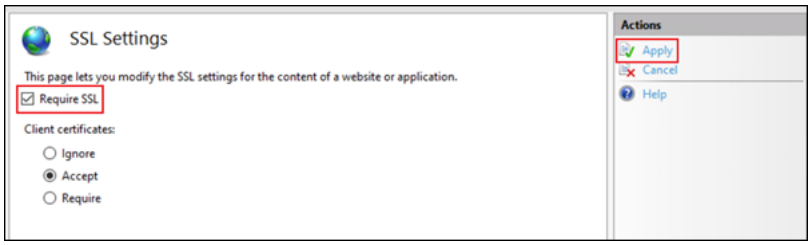
> **Note:**
>
> For the **Binding** settings, it is considered that the third-party authentication service and Store-Front use the same network connection for **Session Remote Start**. If they don't, then remove any IP address and hostname restrictions and clear the **Require Server Name Indication** check box to avoid connection issues.

Require SSL

1. In IIS Manager, select the **Session Remote Start Site**, and double-click **SSL Settings**.



2. On the **SSL Settings** page, select **Require SSL** check box and under **Client certificates**, select **Accept** and click **Apply**.



Configure Session Remote Start

December 9, 2025

Configuration file

The Session Remote Start configuration parameters are stored in the `Web.config file`, which is located in the **Session Remote Start** installation directory `C:\Program Files\Citrix\SRS\`. This file contains essential settings that control the behavior of Session Remote Start and should be configured carefully to ensure proper functionality.

Options

Parameter	Optional/Required	Description
StoreFront™		
StoreFrontServer	Required	The Store Web URL of the store explicitly created for Session Remote Start. E.g. `https://<baseURL>/Citrix/<storename>Web`
LocalFqdn	Required	FQDN of Session Remote Start server.
SessionIdCacheTtlSeconds	Optional	TTL of authentication session between Session Remote Start and StoreFront. The default value is 300. Strongly recommended to keep it unchanged.
UseLegacyStoreFront	Optional	"True"if StoreFront is an earlier version. The default value is "false".
REST API		
RestApiUrl	Optional	Is needed only for certain features, see [Configuration for Accessing Other API Services](/en-us/session-remote-start/optional-configurations#2-configuration-for-accessing-other-api-services).
RestAPICredentialName	Optional	[Is needed only for certain features, See Configuration for Accessing Other API Services.] Note: If the Monitor Service API and Rest API use the same credential, both options can refer to the same credential name. **Note:** DDC version 2308 or later is required.
SiteId	Optional	[Is needed only for certain features, See Configuration for Accessing Other API Services.] For CVAD on-Prem, fill with site ID. For DaaS, empty.

Parameter	Optional/Required	Description
CustomerId	Optional	[Is needed only for certain features, See Configuration for Accessing Other API Services.] For CVAD on-Prem, 'CitrixOnPremises'. For DaaS, customer ID.
DeliveryGroupsCacheTtlMinutes	Optional	TTL of `Delivery Group - Tag` mapping cache. The default value is 10. Strongly recommended to keep it unchanged.
DesktopsCacheTtlMinutes	Optional	TTL of Desktop name cache. The default value is 10. Strongly recommended to keep it unchanged.
ICA® Client		
IcaClientName	Required	A unique client name used for Session Remote Start initiated launches. Must be consistent with the value of '$HostName' in Logon Script DisconnectSession.ps1. It is recommended to keep the default 'srs-server' parameter. Make sure the same value is set in DisconnectSession.ps1.

Parameter	Optional/Required	Description
LaunchDebugMode	Optional	Enable launch debug mode. The default value is ‘False’. If set to ‘True’, the ICA file is saved to disk, and the launch is not executed. Note: For security reasons, the ICA file does not contain sensitive information. As a result, it can only be used for connection testing and cannot be used to actually launch a session.
IcaFileDirectory	Optional	The directory of ICA files (debug mode enabled). The default value is “%App-Data%\Citrix\SessionRemoteStart\IcaFiles” .
IcaLog	Optional	“True”if you save the ICA communication log. The default value is “False”. It is recommended to keep this setting as ‘False’to save space. Enabling it significantly impacts performance and should only be used for troubleshooting.
IcaLogFile	Optional	ICA log path. The default value is “%App-Data%\Citrix\SessionRemoteStart\IcaLogs” .
Session Remote Start		
MaxRequestConcurrency	Optional	The number of concurrent API requests. The default value is 100.
RequestTimeoutSeconds	Optional	The timeout seconds of HTTP request from a 3rd-party service. The default value is “300”.

Parameter	Optional/Required	Description
AutoRefreshConnections	Optional	“True”if Session Remote Start will auto refresh connections periodically. Strongly recommended to keep it unchanged.
Logging		
LogToConsole	Optional	Developer use only.
LogToConsoleLevel	Optional	Developer use only.
LogToDebug	Optional	Developer use only.
LogToDebugLevel	Optional	Developer use only.
OverwriteLogFile	Optional	Overwrite the log file on the next service start. The default value is “false”.
LogFileName	Optional	Log file location. The default value is “%App-Data%\Citrix\SessionRemoteStart\Logs\Ses . If empty, no logging will occur.
LogToFileLevel	Optional	Trace severity level. See here for details. The default value is 5.
MaxLogFileSizeMB	Optional	Defines the maximum size of a single log file (in MB). A new file is created when this limit is reached.
MaxBackupLogFiles	Optional	Specifies the maximum number of backup log files to keep. The oldest files are deleted when the limit is exceeded.

Parameter	Optional/Required	Description
LogToEventViewer	Optional	Enable event viewer logging. The default value is `False`. The log is under Windows Logs > Application (path: `%SystemRoot%\System32\Winevt\Logs\Application.evtx`) and the event source name is "SessionRemoteStart".
PreviousWebConfigDirectory	Required	A folder that stores previous configuration versions, allowing SRS to monitor changes to Web.config. The default value is "%App-Data%\Citrix\SessionRemoteStart\PreviousA .
Telemetry		
TelemetryCredentialName	Optional	[Is needed only for certain features, See Configuration for Accessing Other API Services.] Configure Citrix Monitor Service API Credential. **Note:** If the Monitor Service API and Rest API use the same credential, both options can refer to the same credential name.
TelemetryDataDirectory	Optional	The directory of telemetry data. The default value is `%AppData%\Citrix\SessionRemoteStart\TelemetryData`, even if no value is specified.
TelemetryRotationDays	Optional	Telemetry data is collected only if this field is specified, as it also acts as a feature toggle. The number of days to retain the data files in "%App-Data%\Citrix\SessionRemoteStart\Telemetry and "%App-Data%\Citrix\SessionRemoteStart\Telemetry directories.The default value is 90. DDC version 2305 or later is required.

Parameter	Optional/Required	Description
AutoTelemetryCollectingDays	Optional	The number of days of telemetry data for SRS service to collect automatically in the background and store in the telemetry data file.The default value is 30.**Note:** The SRS service needs about 30 minutes for every 30 days to collect if this feature is firstly used.
Mutual TLS		
mTLSEnabled	Optional	"True"if you want to enable mTLS support. The default value is "false".
mTLSClientCertificateThumbprints	Optional	The thumbprint list of client certificates, separated by comma. Required if mTLS is enabled.
Smart Access Tags		
SmartAccessFarmName	Optional	Farm name, for example: `_XD_192.168.1.19_443`
SmartAccessConditions	Optional	List of conditions, for example: `PL_WB_10.107.197.243, PL_WB_10.107.197.244`
Auto Logoff		
AutoLogoffEnabled	Optional	[Is needed only for certain features, see Configuration for Accessing Other API Services.] Enable automatic logoff for sessions pre-launched by Session Remote Start if never reconnected within the specified time. The default value is "False".
AutoLogoffIdleMinutes	Optional	The idle duration (in minutes) before automatically logging off. The default value is 30.

Parameter	Optional/Required	Description
AutoLogoffCheckIntervalMinutes	Optional	The interval (in minutes) at which Session Remote Start server checks whether a session needs to be auto logged off. The default value is 5. Strongly recommended to keep it unchanged.
Support for Resuming Hibernated VMs		
ResumeMachinesEnabled	Optional	[Is needed only for certain features, See Configuration for Accessing Other API Services.] Enable resuming hibernated machines during the pre-launch process. The default value is `False`.
ResumeMachinesCheckIntervalMinutes	Optional	The interval (in minutes) at which Session Remote Start server synchronizes hibernated machines. The minimum value is 5. The default value is 5.
BatchResumeMachinesDisabled	Optional	Disable batch processing of machine resume requests. While requests may arrive individually, they can be processed in batches to improve backend performance. The default value is `False`.
BatchResumeMachinesIntervalSeconds	Optional	The interval (in seconds) at which Session Remote Start server resumes hibernated machines in batches. The default value is 10.

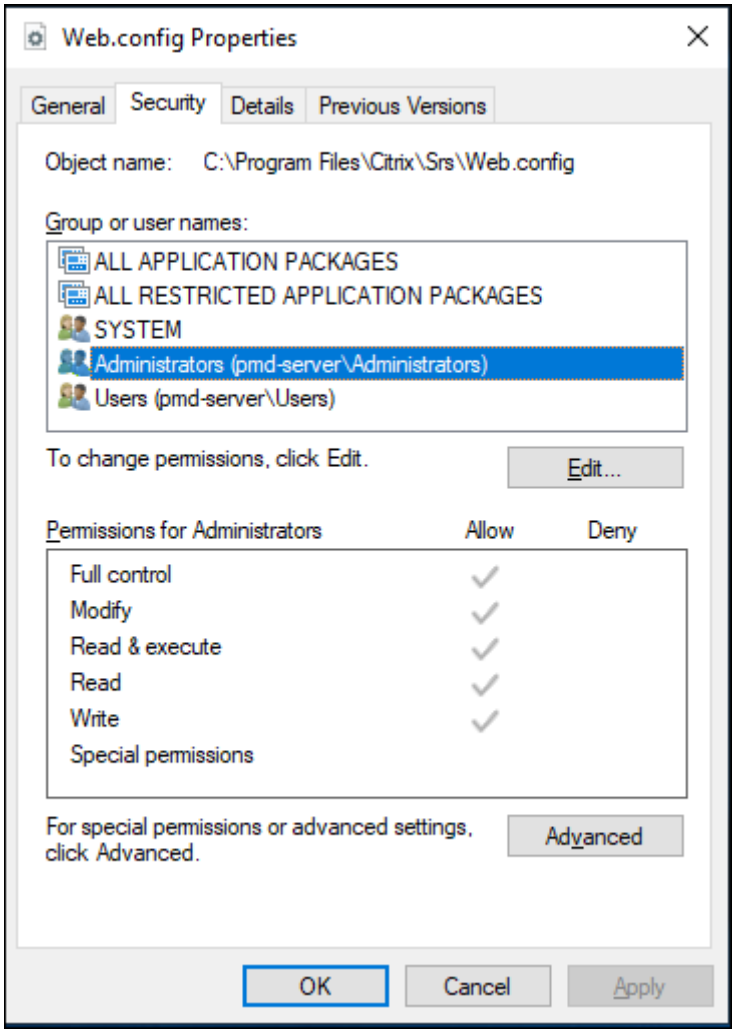
Parameter	Optional/Required	Description
ResumingByStoreFront	Optional	By default, the SRS resumes a hibernated VM through Orchestration. To enable resuming a hibernated VM through StoreFront, the value must be set to 'true'. The default value is "false".
ReconnectionTimeoutSecondsAfterResume	Optional	Sets a timeout in seconds for the temporary reconnection after a VM is resumed. Once this time limit is reached, the session is automatically disconnected. This ensures the session returns to a disconnected state. The default value is 60.

For log-related configuration, see Change Log File/Telemetry File Location section.

Configuration file permissions

Check the permissions of the `Web.config` file located at `C:\Program Files\Citrix\SRS\Web.config` (default location). Ensure that administrator accounts have full control over the file, allowing them to read, modify, and manage it. At the same time, other users or groups should not have modify permissions to prevent unauthorized changes.

This helps maintain security and prevents unintended modifications to the configuration.



StoreFront™ Configuration

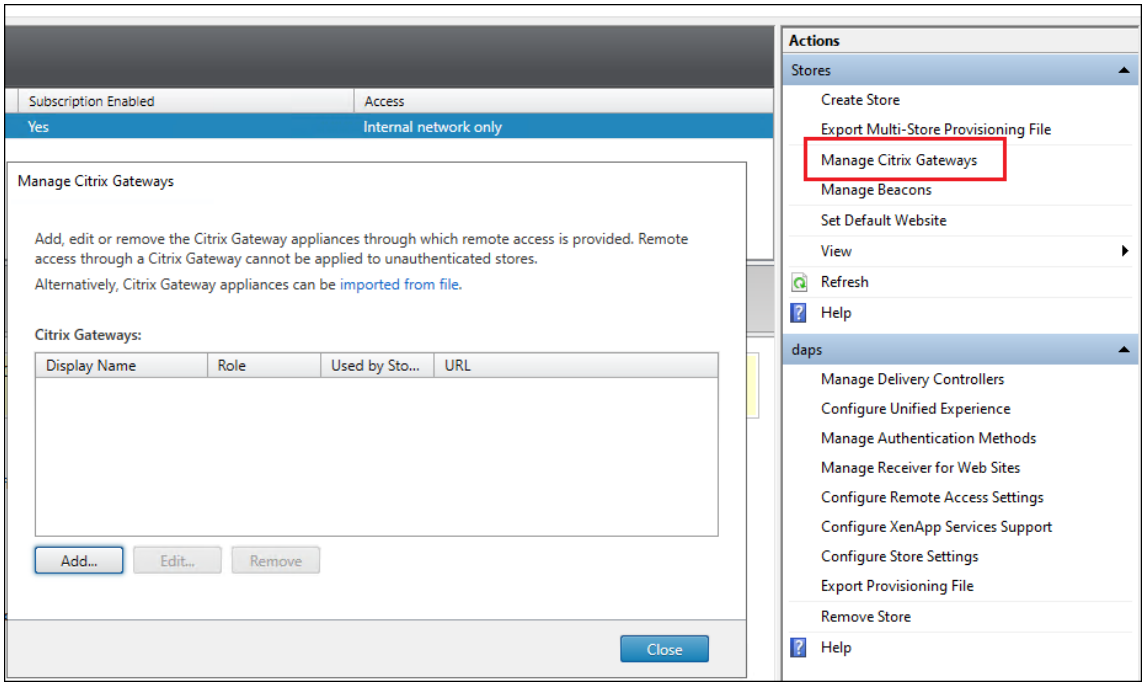
November 26, 2025

Install SSL certificate for accessing Session Remote Start

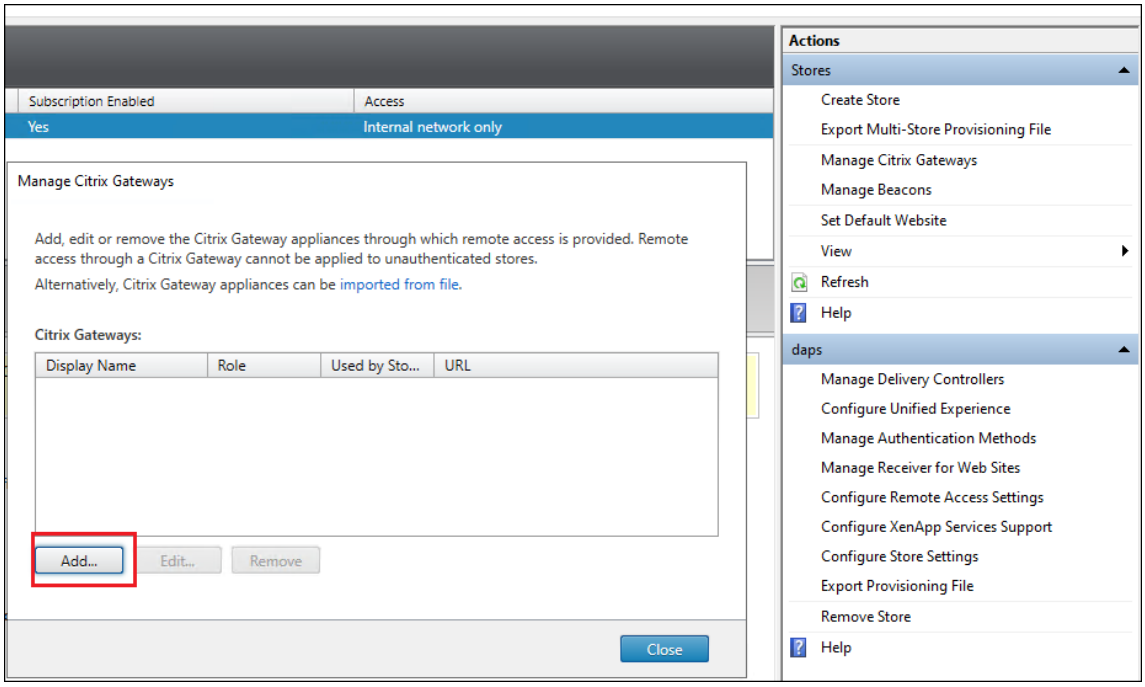
To simplify deployment, each Session Remote Start server is set up as a specialized Gateway with StoreFront. Therefore, StoreFront must be able to communicate with Session Remote Start using the specified Gateway callback URL: `https://<Session Remote Start FQDN>/SessionRemoteStart/CitrixAuthService/AuthService.asmx`.

Add Session Remote Start as a Gateway

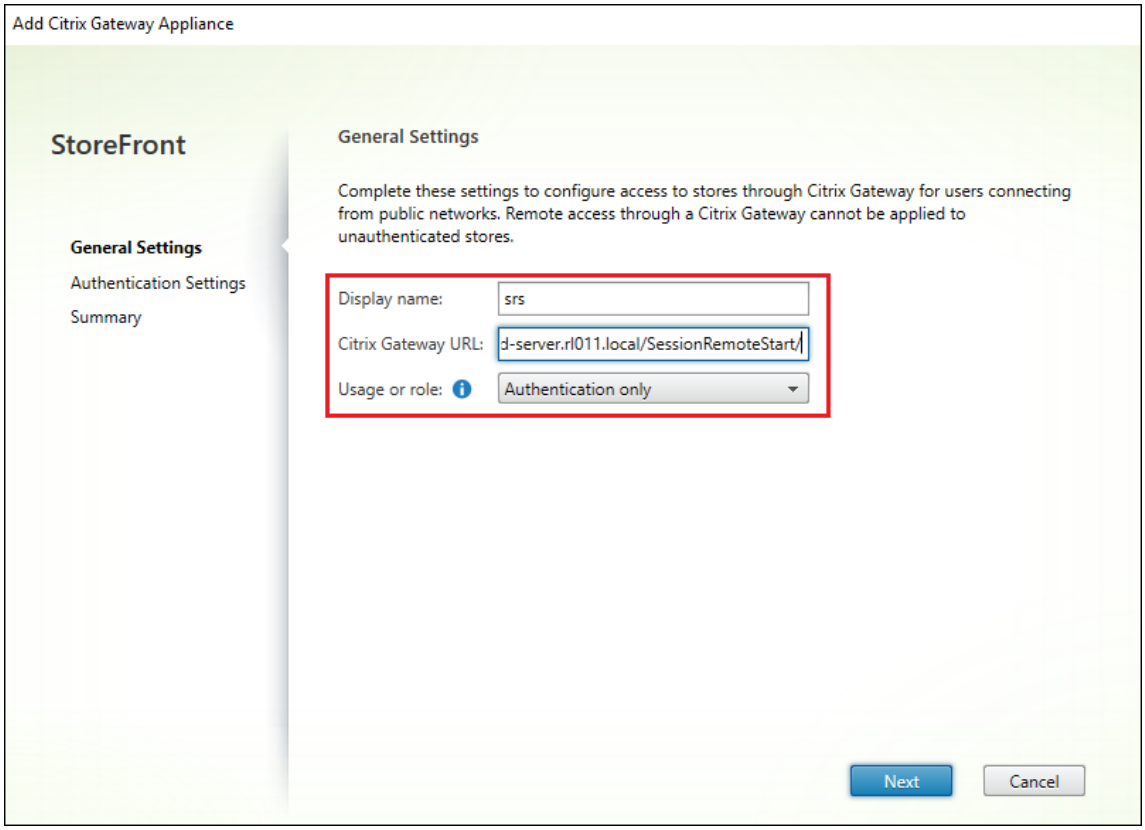
1. Click **Manage Citrix Gateways** in the **Stores** panel on the right.



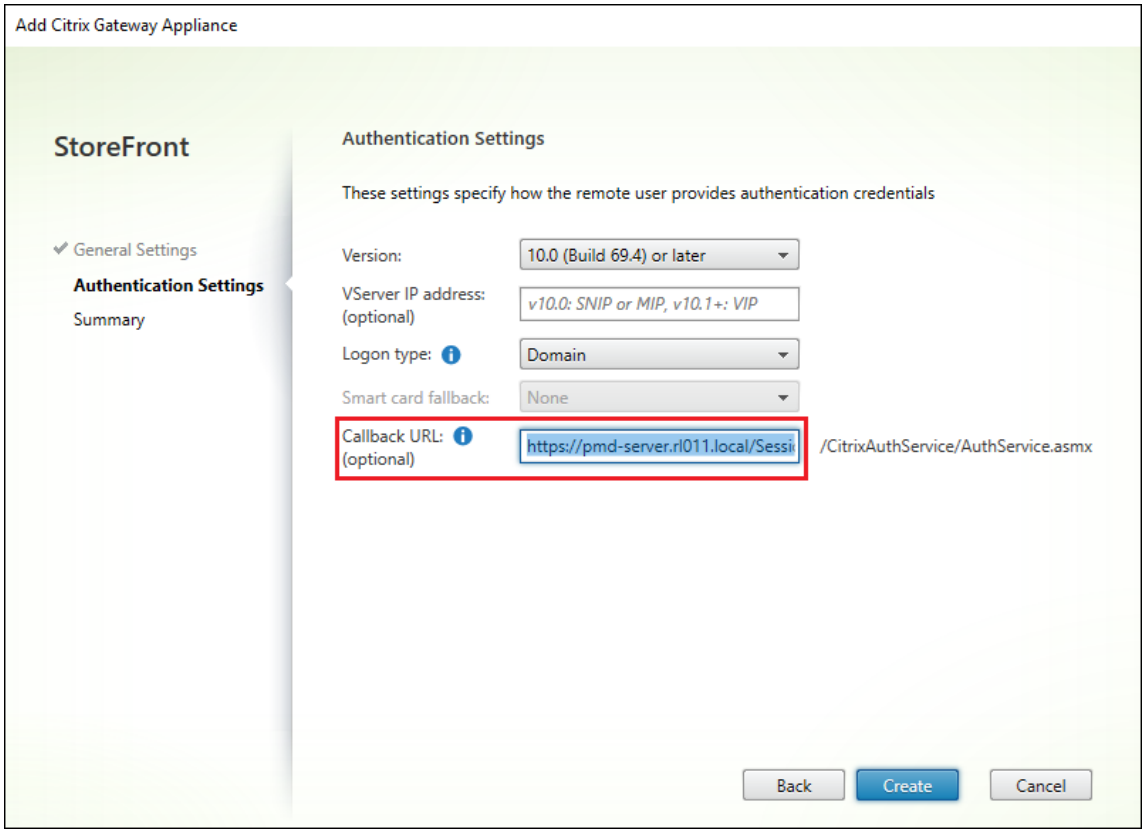
2. Add a new gateway.



3. Set Display name, set Citrix Gateway URL with `https://<Session Remote Start FQDN>/SessionRemoteStart/`, and change **Usage or role** to **Authentication only**.



4. Set Callback URL with `https://<Session Remote Start FQDN>/SessionRemoteStart`.

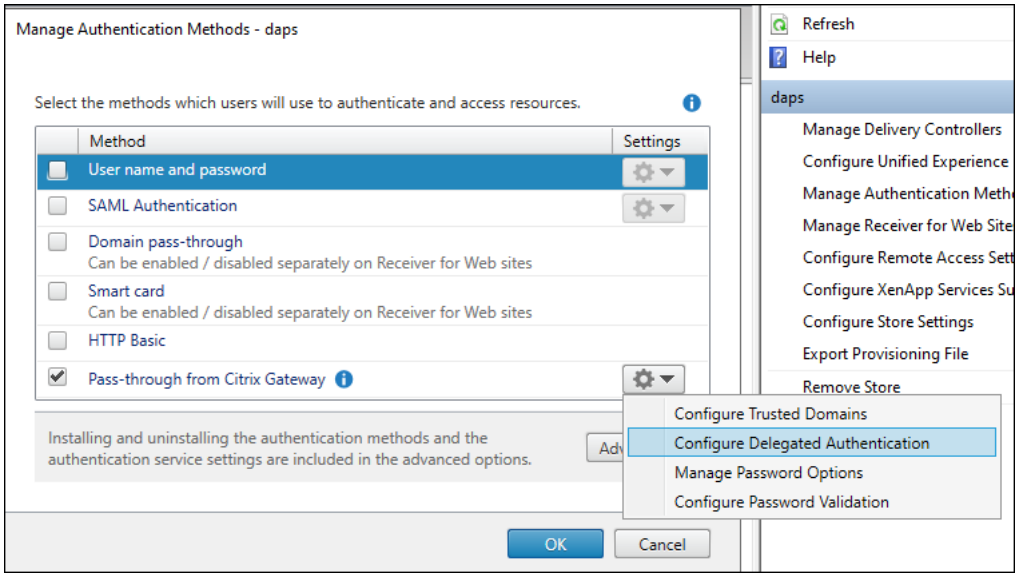


Create a New Store for Session Remote Start

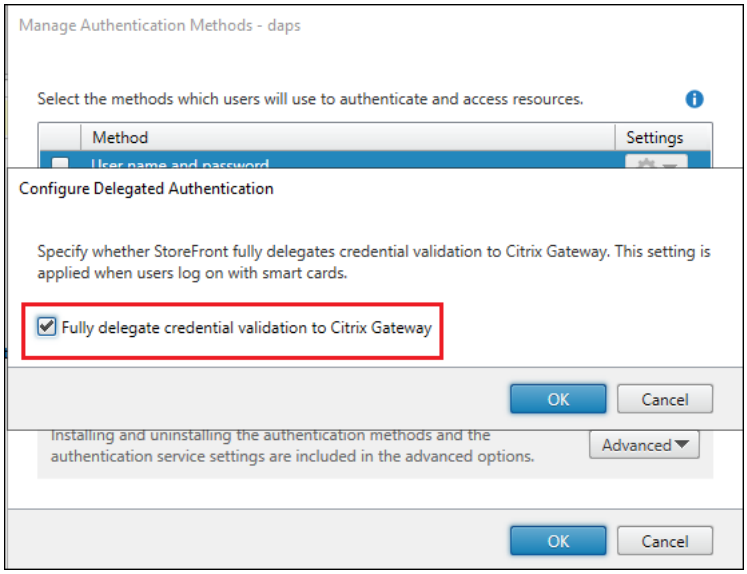
A new Store must be created exclusively for Session Remote Start to ensure proper functionality and integration.

Manage Authentication Methods

1. On the new Store, find the **Manage Authentication Methods** in the Store configuration panel on the right.
2. Ensure the **Pass-through from Citrix Gateway** check box is selected.
3. Expand settings, click **Configure Delegated Authentication**.

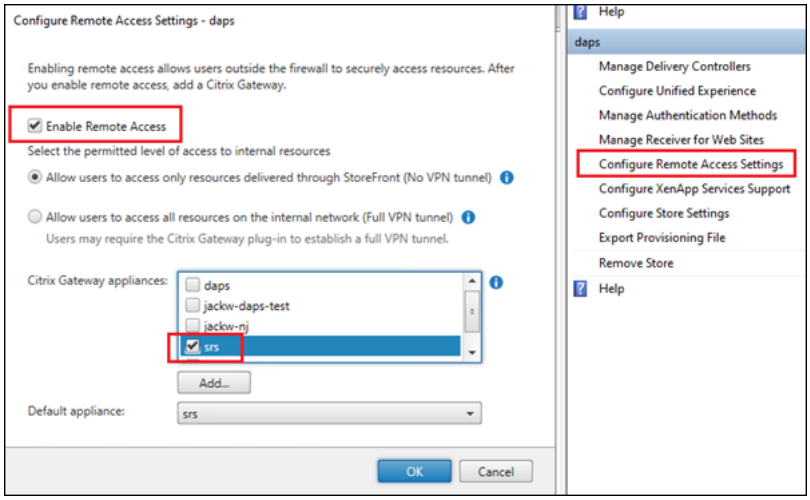


4. Enable the **Fully delegate credential validation to Citrix Gateway** option.



Configure Remote Access Settings

1. Click **Configure Remote Access Settings** in the Store configuration panel on the right.
2. Check the box **Enable Remote Access**.
3. Select the Gateway configured above.



Configure DDC

September 13, 2025

Enable TrustRequestsSentToTheXmlServicePort

Enable this setting on the Delivery Controller™ (DDC) to ensure that Session Remote Start requests coming through StoreFront are trusted.

For Citrix Virtual Apps and Desktops™

Run the following commandlets on the Delivery Controller:

```
asnp Citrix*
Set-BrokerSite -TrustRequestsSentToTheXmlServicePort $true
```

For Citrix DaaS™

Run the following commandlets:

1. On a machine with internet connection, install Citrix DaaS Remote PowerShell SDK.
   Other articles also contain related information:
2. Open PowerShell, and execute the command:
   `Get-XdAuthentication`
   An authentication dialog opens.

3. Execute the commandlets as On-Prem:
   ```
   asnp Citrix*

   Set-BrokerSite -TrustRequestsSentToTheXmlServicePort $true
   ```
4. Log off.
   ```
   Clear-XDCredentials
   ```

App Protection: If customers have an App protected delivery group, see App Protection.

HTTP Proxy: Customers must configure the necessary settings and follow the steps in HTTP Proxy Configuration if their setup includes HTTP proxies to ensure proper functionality.

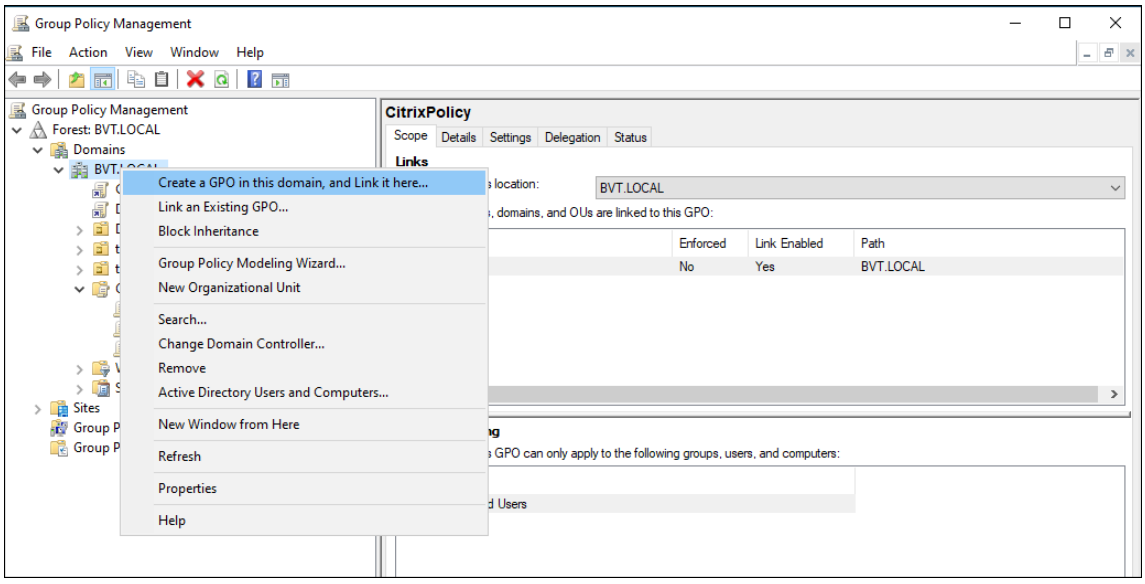
AD Server - Logon Script Configuration

March 12, 2025

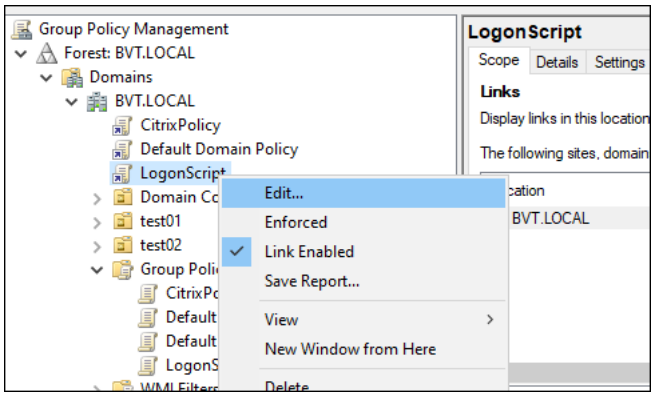
The Logon script is used by the Virtual Delivery Agent (VDA) to disconnect the pre-launch session after the Session Remote Start initiated logon completes. This script is applied to all VDAs and users using Group Policy to ensure that no resources remain active and unused. It allows users to reconnect later through their regular login method, ensuring efficient resource management and a seamless user experience.

Steps to create a GPO and link it to a Domain

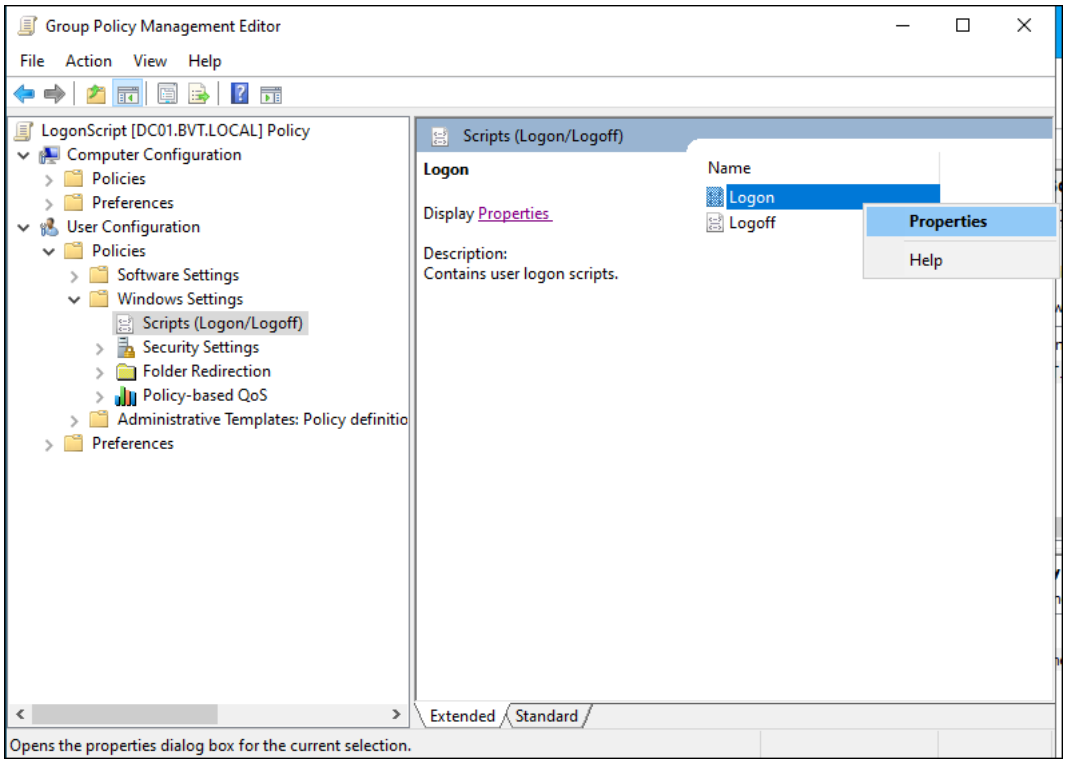
1. On the Windows domain controller, open **Group Policy Management** and create a GPO under the domain or on the Citrix VDAs OU.



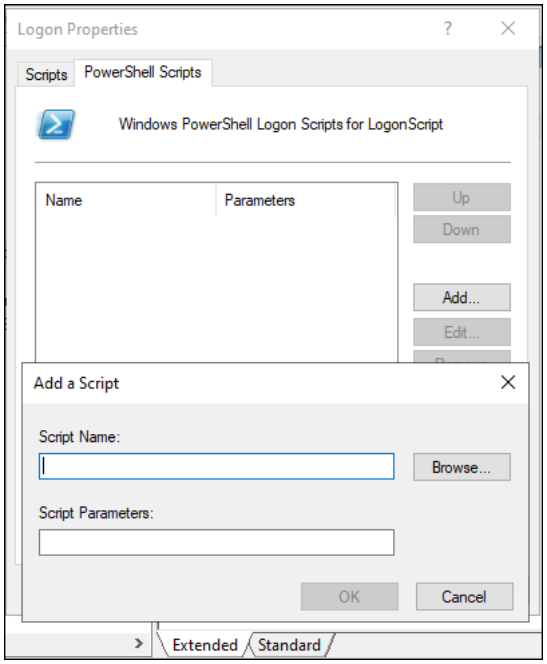
2. Right-click on the GPO, select **Edit** to open the Group Policy Management Editor.



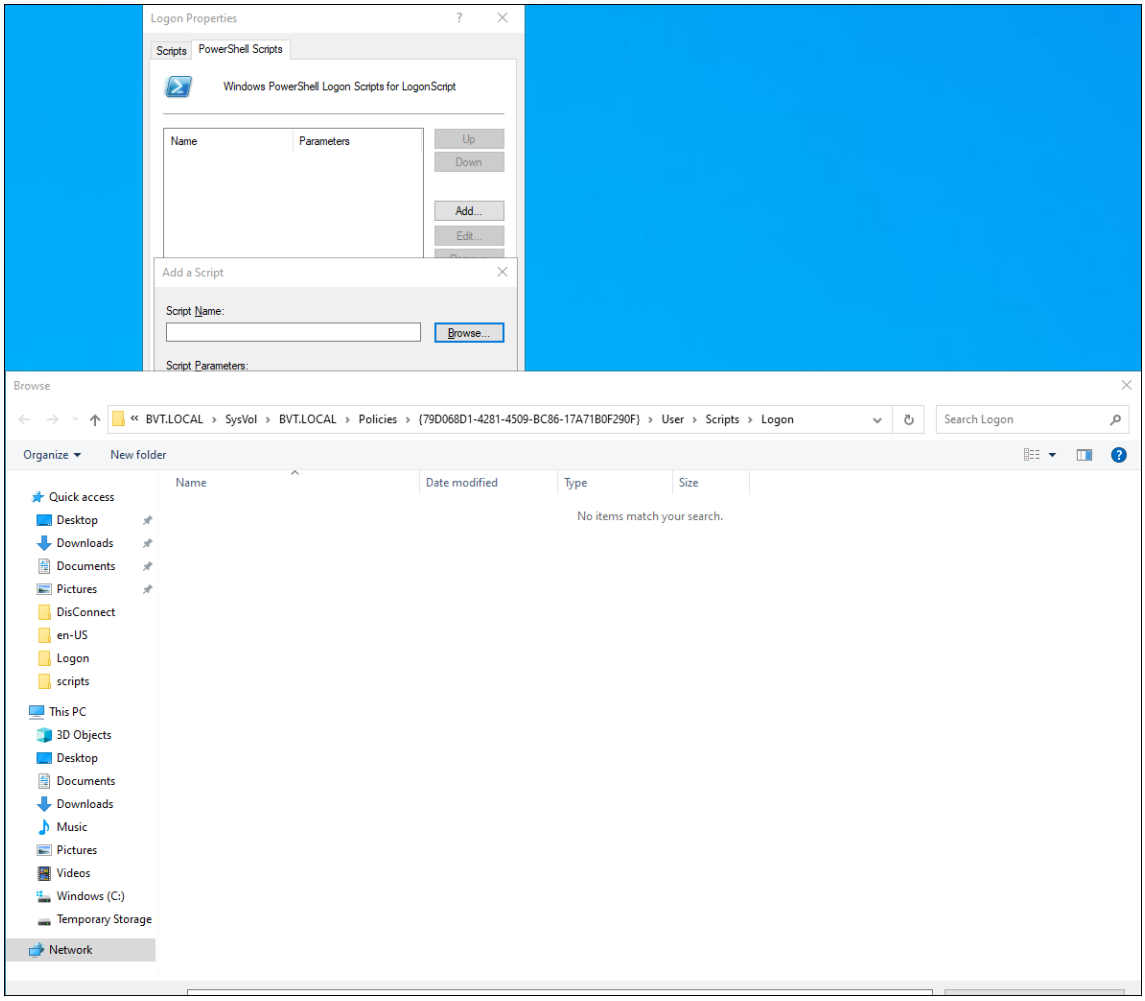
3. In the **Group Policy Management Editor**, expand **User Configuration > Policies > Windows Settings > Scripts (Logon/Logoff)**. Right-click **Logon on** in the right panel and select **Properties**.



4. Switch to the **PowerShell Scripts** tab, and click **Add**….



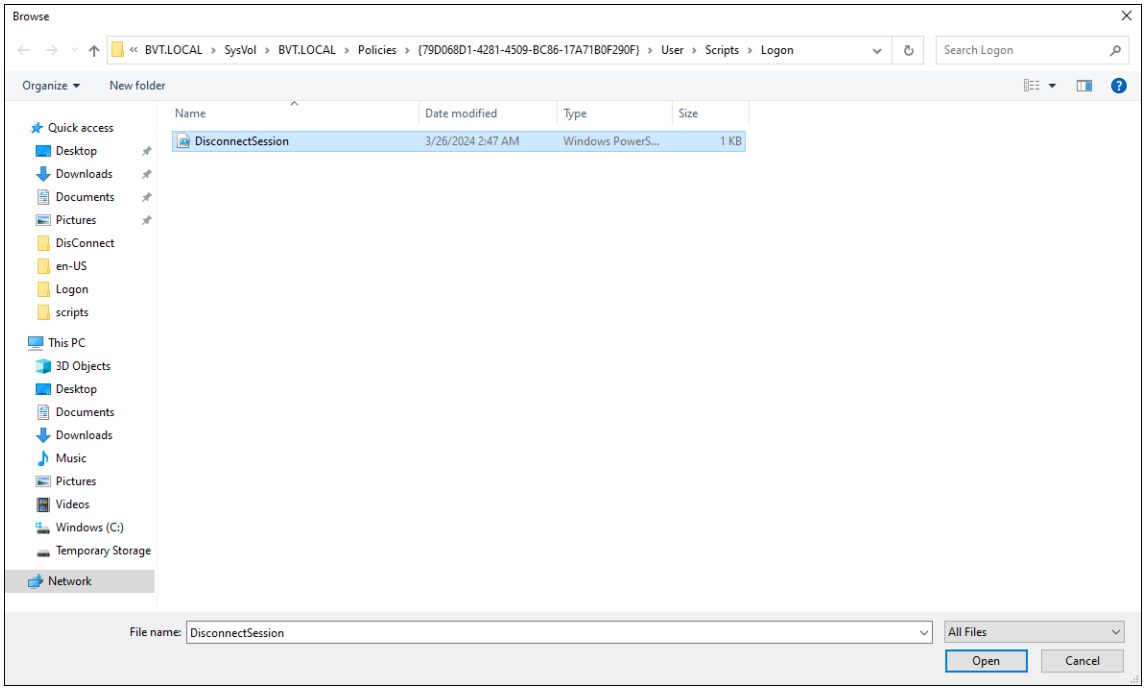
5. Click **Browse...** on the right of the **Script Name** field, a file browser pops up, located in the **NetLogon** folder by default.



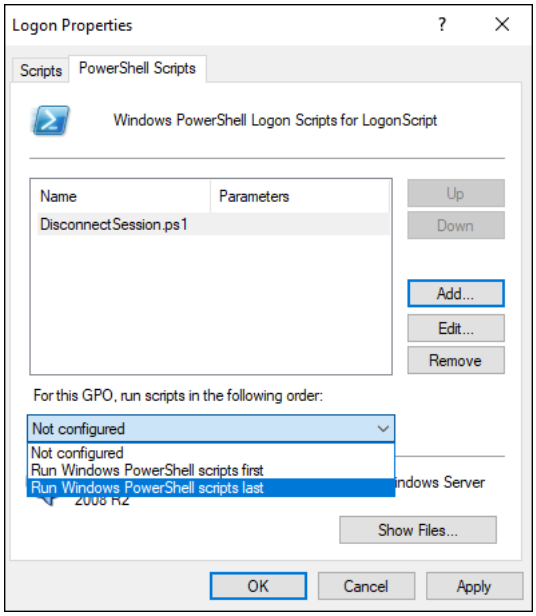
The Logon folder is a shared folder which typically has read-only and execute permissions by machines and users. To avoid permission issues, it is recommended not to use any other folder.

Create **DisconnectSession.ps1** under this folder.

> **Note:**
>
> Use the one provided by Citrix as part of the installation file.



6. Select **Run Windows PowerShell scripts last** and click **OK** or **Apply**.



Configure Citrix FAS

September 7, 2025

Steps to Install and configure FAS

To install and configure FAS in your setup, see Install and configure.

FAS Do's and Dont's

1. It is mandatory to Enable the FAS plug-in for the SRS Store only (using the PowerShell script) as we are using Citrix Virtual Apps and Desktops StoreFront.
2. Check the rule on your FAS server, to ensure your STF is allowed in the **Manage StoreFront™ Permissions** page.
    a) Be aware that by default, there is a DENY entry for Domain Computers. On windows DENY always "wins".
    b) Verify if the permissions from Storefront servers for groups such as domain computers is not set to Deny.
    c) Verify if **FAS rules > access control > domain computers** is set to **Allow**.
3. Ensure the domain users are added in the FAS console.
    a) **FAS console > Rule Default > restrictions > Manage user permissions** check if you have the domain user here.
    b) Set the Domain users for FAS user authentication in the FAS console to **allowed**.
4. Storefront servers should be authorized to use FAS by default, if not, do it explicitly.
    a) On the FAS server, open **Citrix Federated Authentication Service** console.
    b) Go to the **Rules** tab, select the appropriate policy and click the pencil icon to edit.
    c) On the left menu select **Access control** and click the **Manage StoreFront access permissions** link.
    d) In the **Permission for StoreFront Servers** page, add your StoreFront servers and give them the **Assert Identity** permission and click **OK**.

Installation checklist

September 13, 2025

Session Remote Start Server

1. Open <`https://<baseURL>/Citrix/<storename>Web>` in a browser and verify that it loads without any warnings.

> **Note:**
>
> Ignore the **No logon methods are available on this platform** message.

2. If **LogFileName** is configured:
    - Check if the log file exists.
    - Review the log for any errors that may indicate issues with the setup.

3rd-party Auth Service Server

1. Open `https://<Session Remote Start FQDN>/SessionRemoteStart/index.html` and verify if it opens without a warning.

StoreFront™ Server

1. Open `<https://<Session Remote Start FQDN>/SessionRemoteStart/CitrixAuthService/AuthService.asmx>` and verify if it opens without a warning.
2. Double check the configurations, including:
    a) Gateway, especially double check the gateway URL and callback URL must be Session Remote Start server(`https://<Session Remote Start FQDN>/SessionRemoteStart`).
    b) Authentication methods, especially **Delegated Authentication**.
    c) Remote Access, specifically applies to the gateway configured above.
3. Federated Authentication Service integration should be enabled on a StoreFront Store using the PowerShell script. For more information, see, Enable the FAS plug-in on StoreFront stores.

DDC

Check if **TrustRequestsSentToTheXmlServicePort** is enabled. Run the following commandlets:

```
asnp Citrix*
Get-BrokerSite
```

For more information about the commandlets, see DDC Configuration.

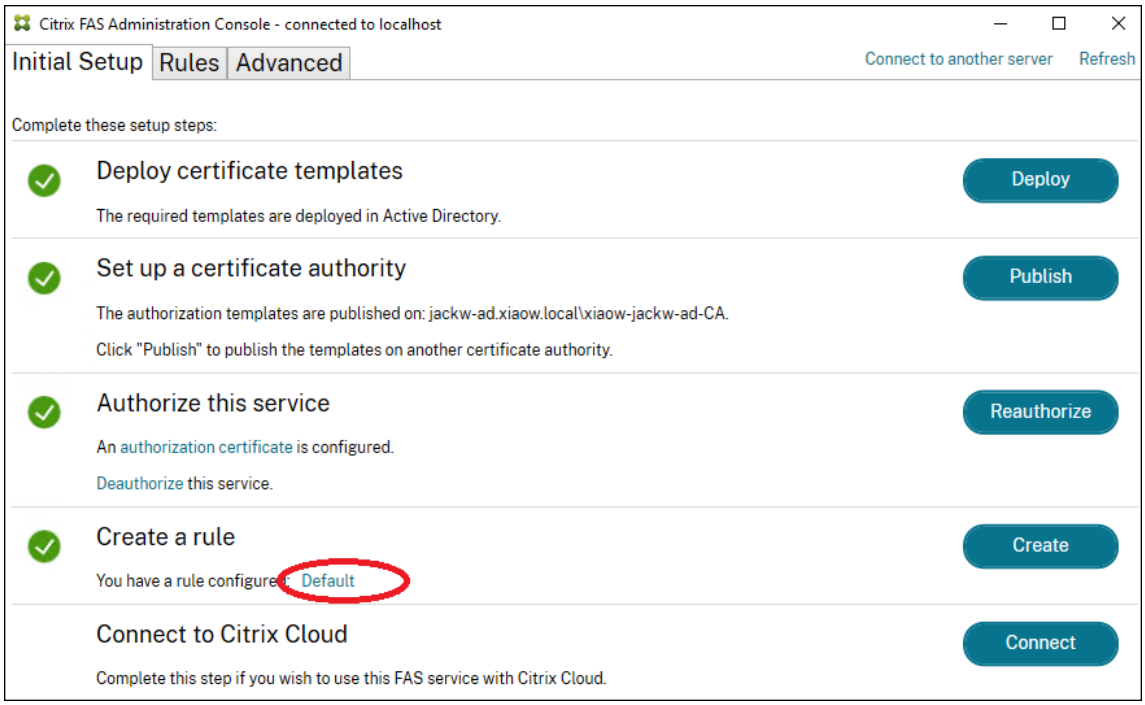
AD

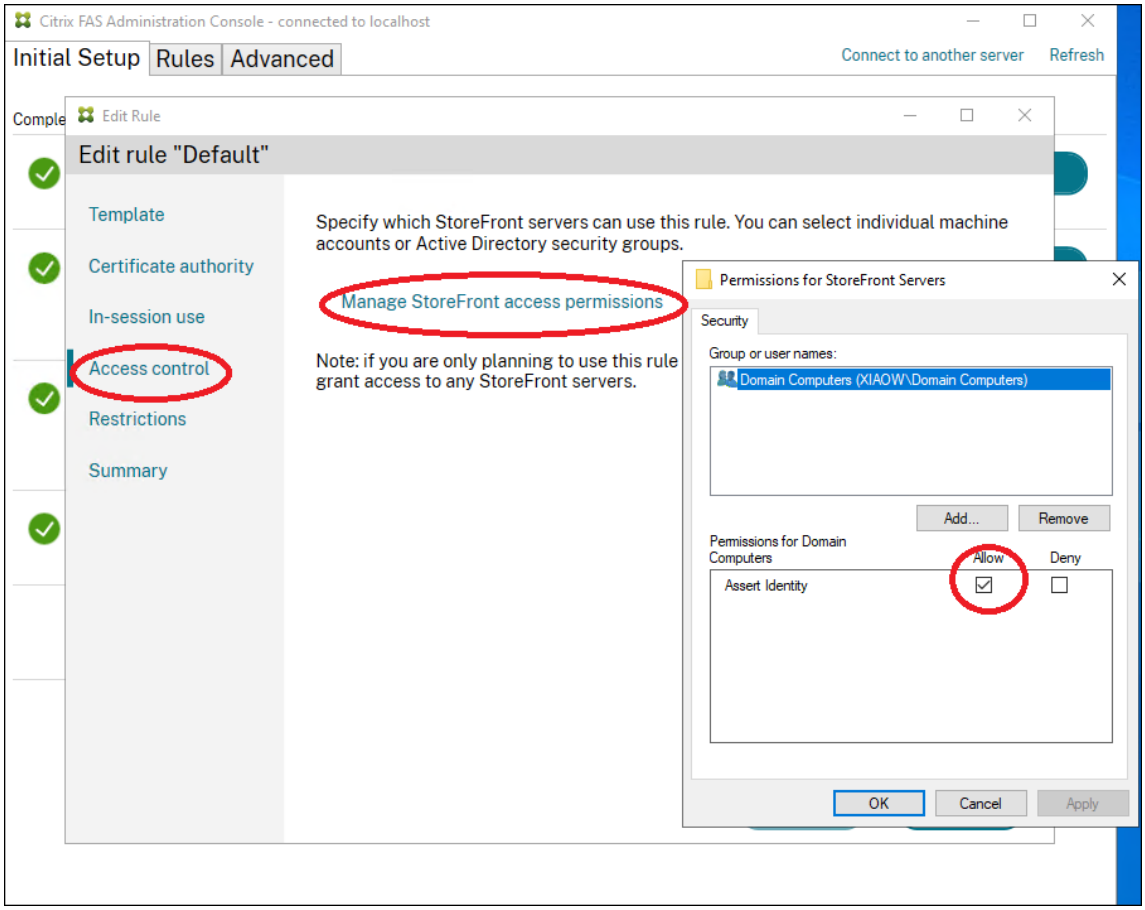
Verify if the logon script is configured.

FAS

If FAS has never been installed before, and it is specifically for the installation of Session Remote Start, please ensure that FAS is correctly configured and functional.

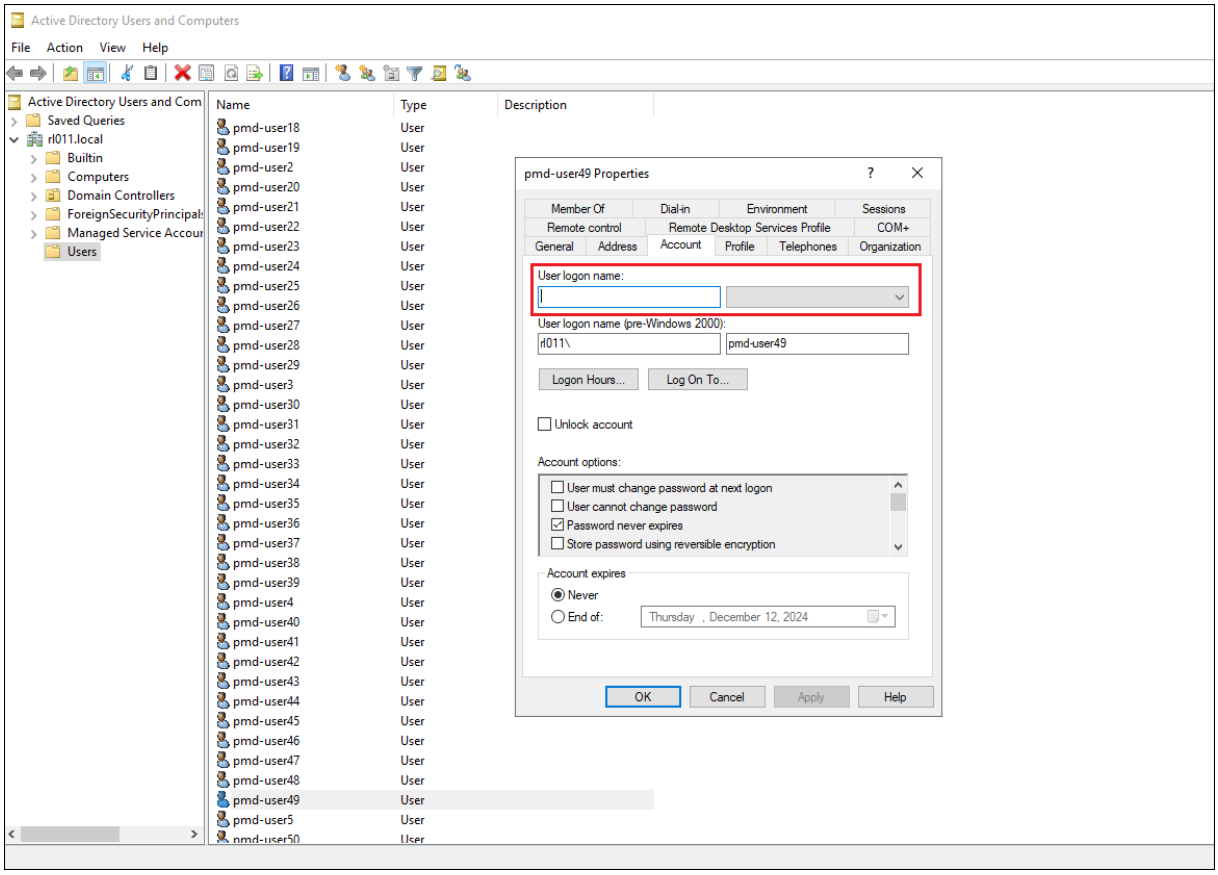
1. Reconfirm that FAS plug-in is enabled on StoreFront for SRS store, as mentioned in StoreFront checklist using the PowerShell script.
2. Reconfirm that DDC trust the StoreFront servers requests, as mentioned in DDC checklist (**TrustRequestsSentToTheXmlServicePort** enabled).
3. Confirm that Group Policy is correctly configured, especially FAS FQDN is configured. Confirm Group Policy is applied to all necessary machines (StoreFront/VDA/DDC).
4. Confirm the user rules. If the default rule is used for quick FAS configuration, please do confirm that Domain Computers is NOT denied.





User Logon Name

For every end user, the **User logon name** must be configured.



Session Remote Start local testing

March 12, 2025

> **Note:**
>
> Some features may require additional configurations. For more information, see Optional Configuration.

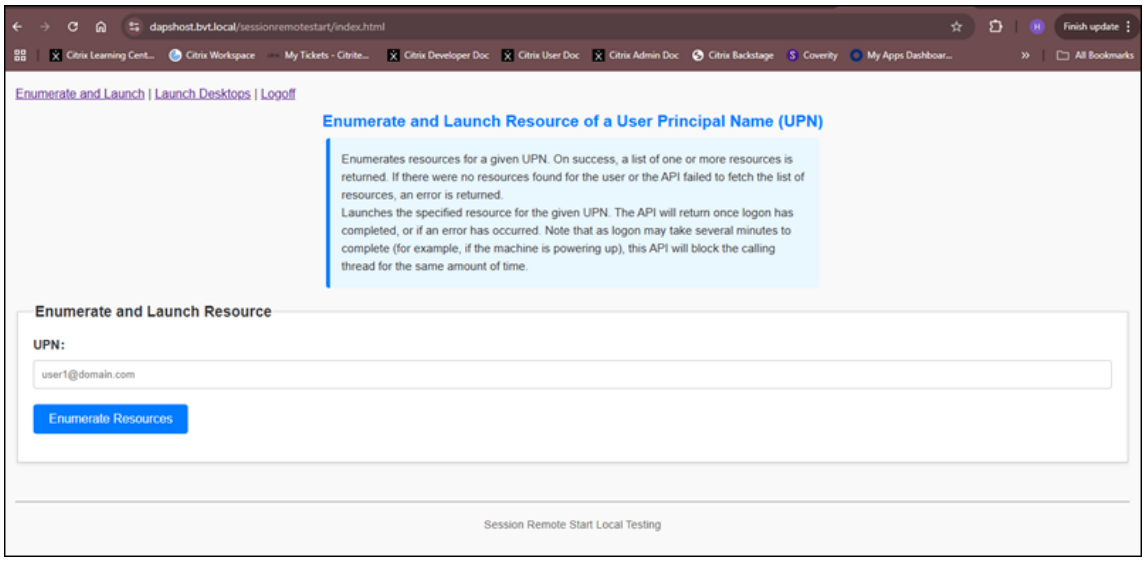
Open Session Remote Start Web Local Testing

Open `https://<Session Remote Start FQDN>/SessionRemoteStart/index.html`, in a web browser and check if it loads successfully without any warnings.

There are three tabs on the top left: **Enumerate and Launch**, **Launch Desktops**, and **Logoff**.

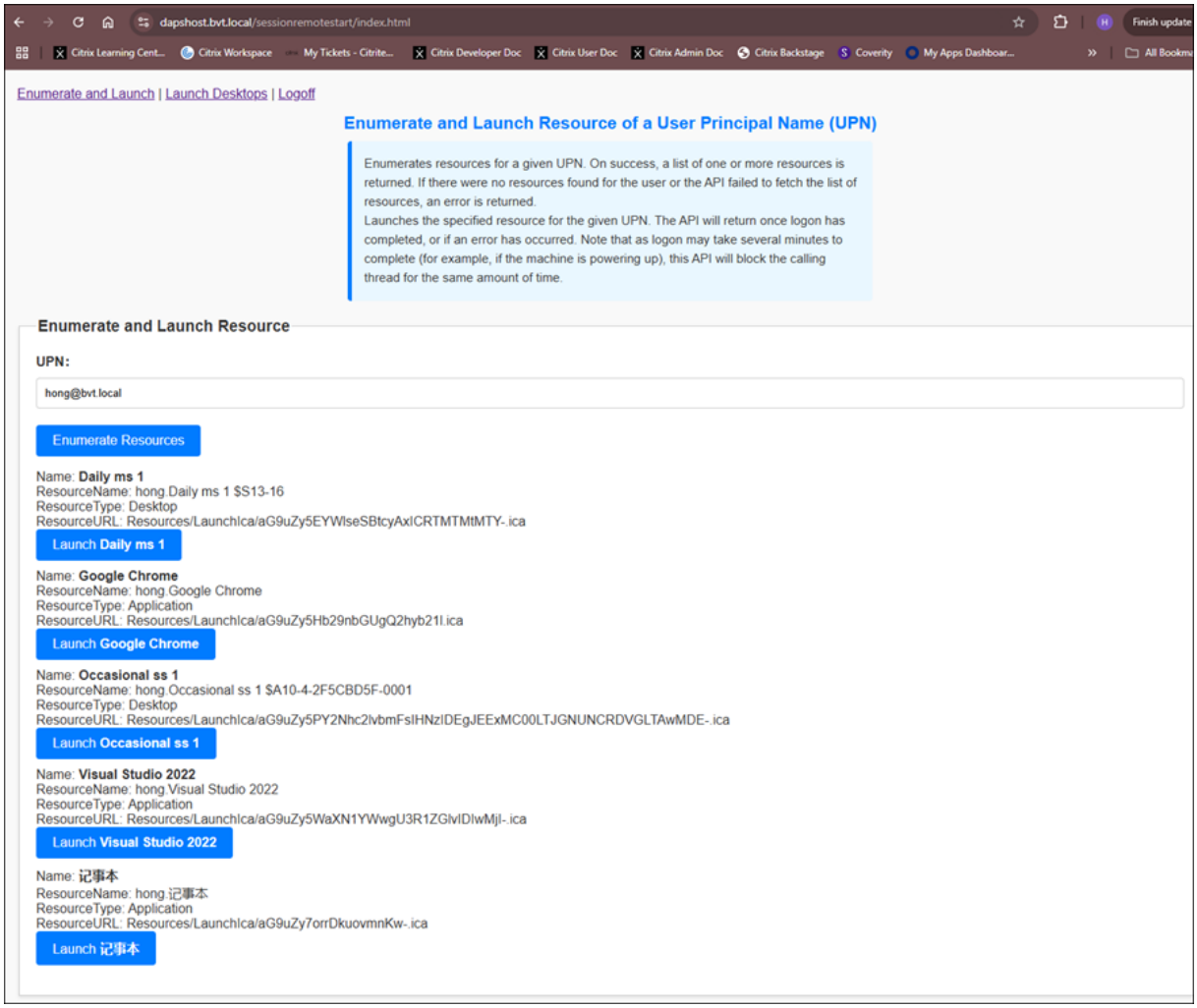
1. **Enumerate and Launch**: Enumerate available resources for a **User Principal Name (UPN)** and allows launching a selected resource for the UPN.

2. **Launch Desktops**: Session Remote Start supports launching all desktops using three methods.
    - Launch All Desktops of a User Principal Name (UPN)
    - Launch All Desktops by Tags of a User Principal Name (UPN)
    - Launch All Desktops assigned to the UPNs in the specified AD groups
3. **Logoff**: Logs off all sessions for a given UPN and device name.

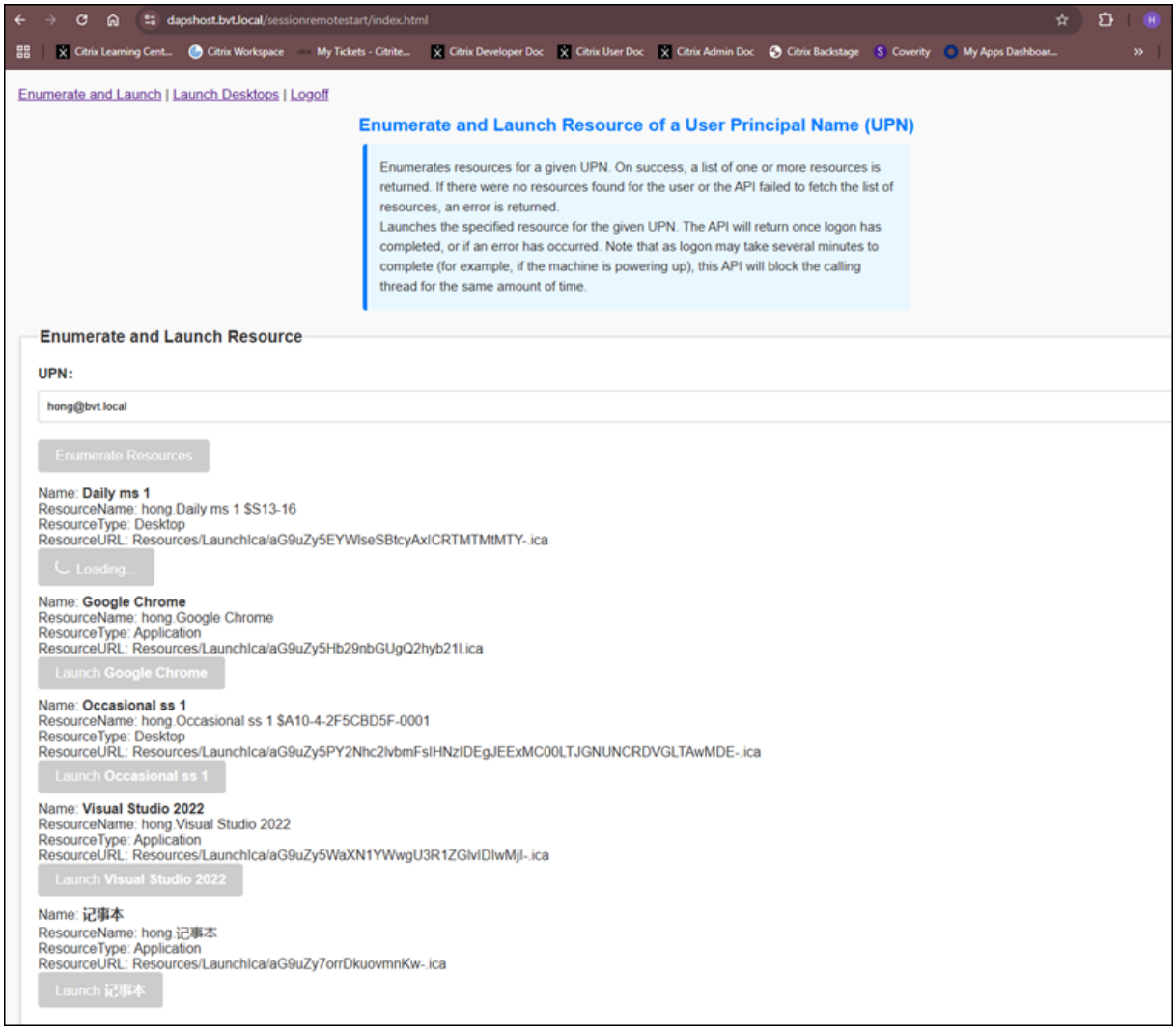


Enumerate and Launch Resource of a User Principal Name (UPN)

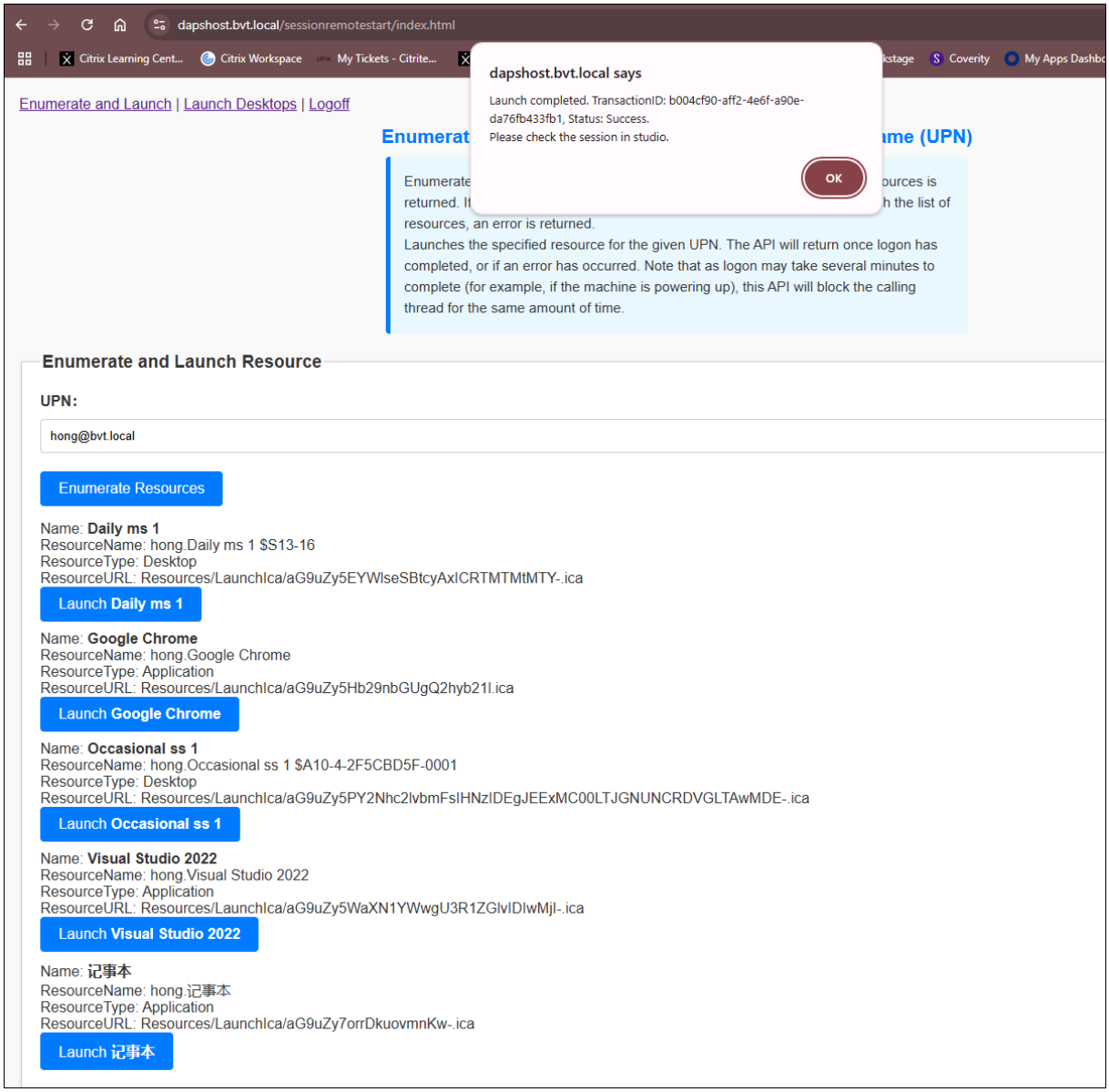
Input the User Principal Name (UPN), then click **Enumerate Resources**.

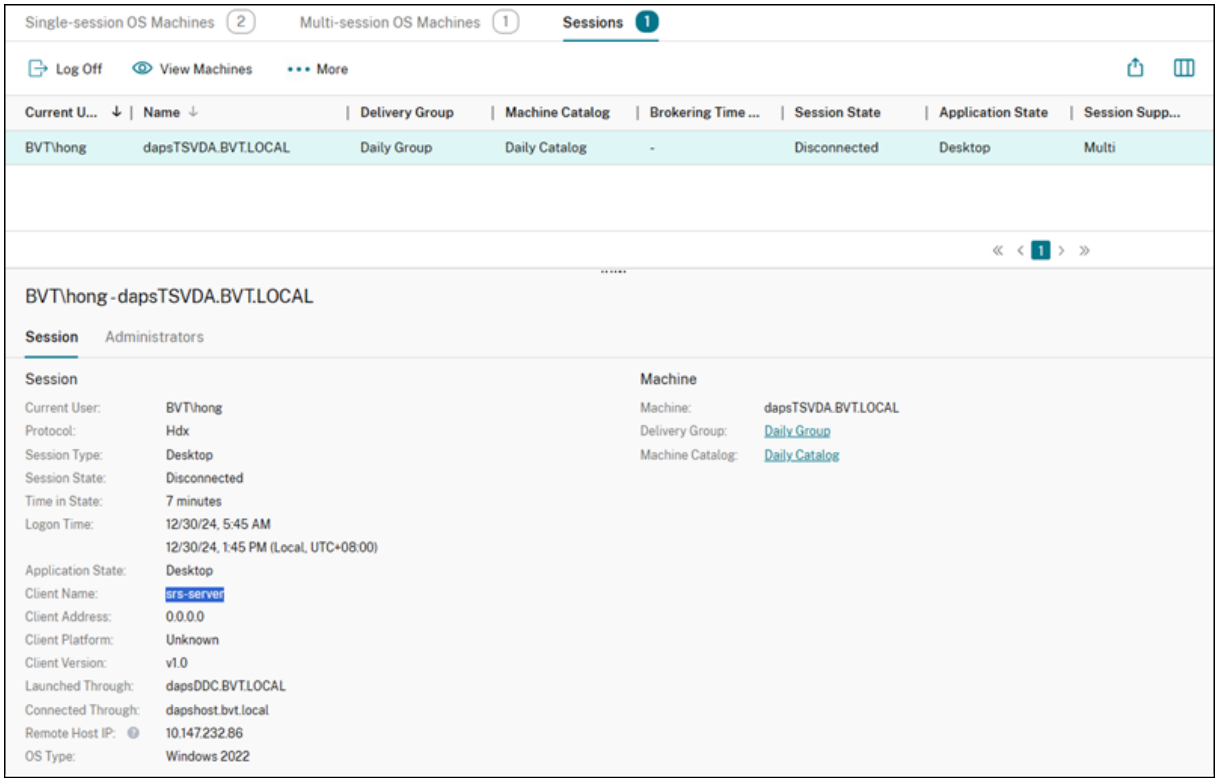


Then choose a resource, for example: `Daily ms 1` and click `Launch Daily ms 1`. The API waits for a logon notification from the VDA before returning to the caller.



A prompt appears with a notification if the resource is launched successfully. Then, check the session in the studio.

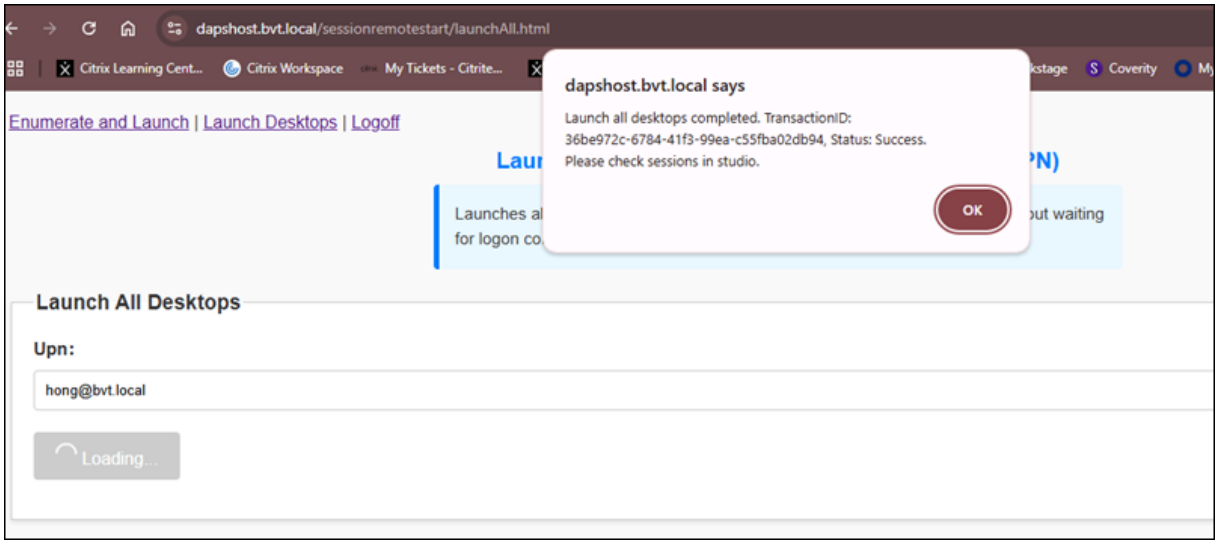


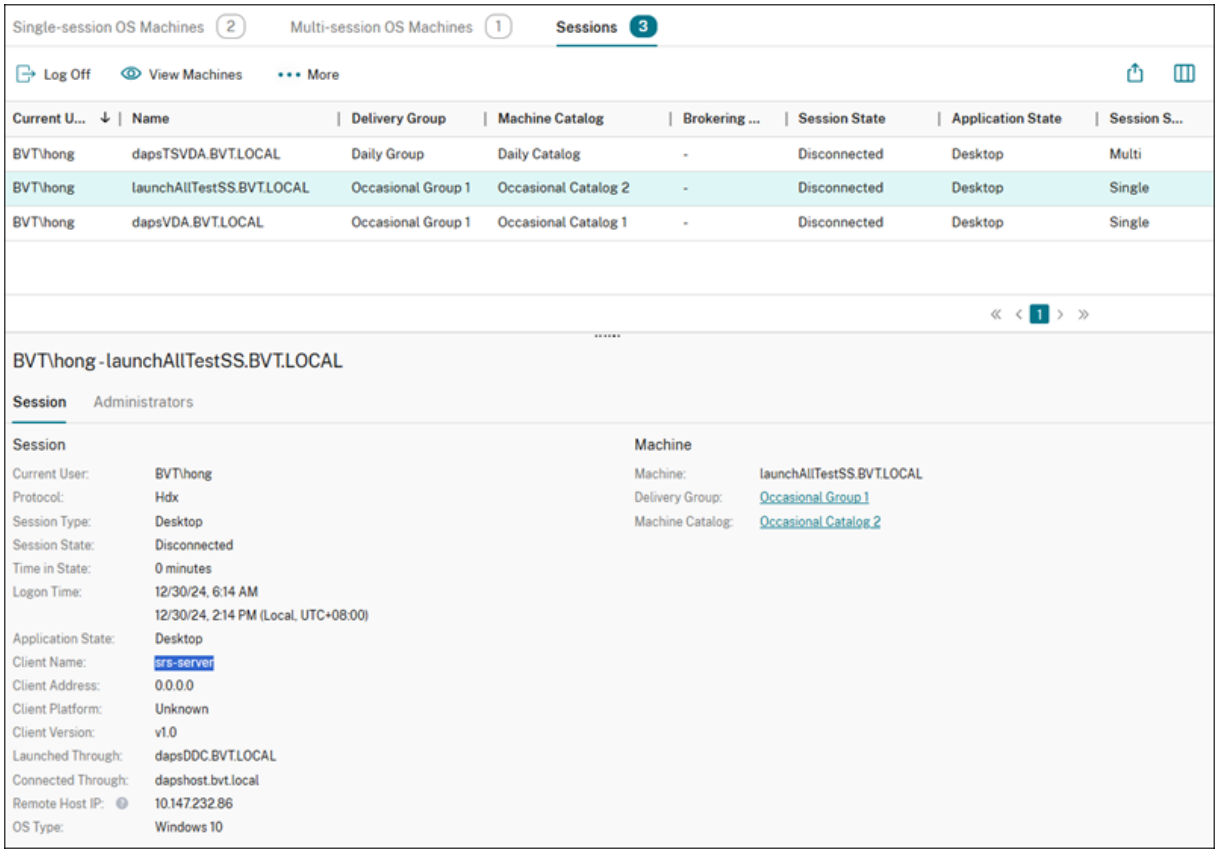


Launch Desktops

Launch All Desktops of a User Principal Name (UPN)

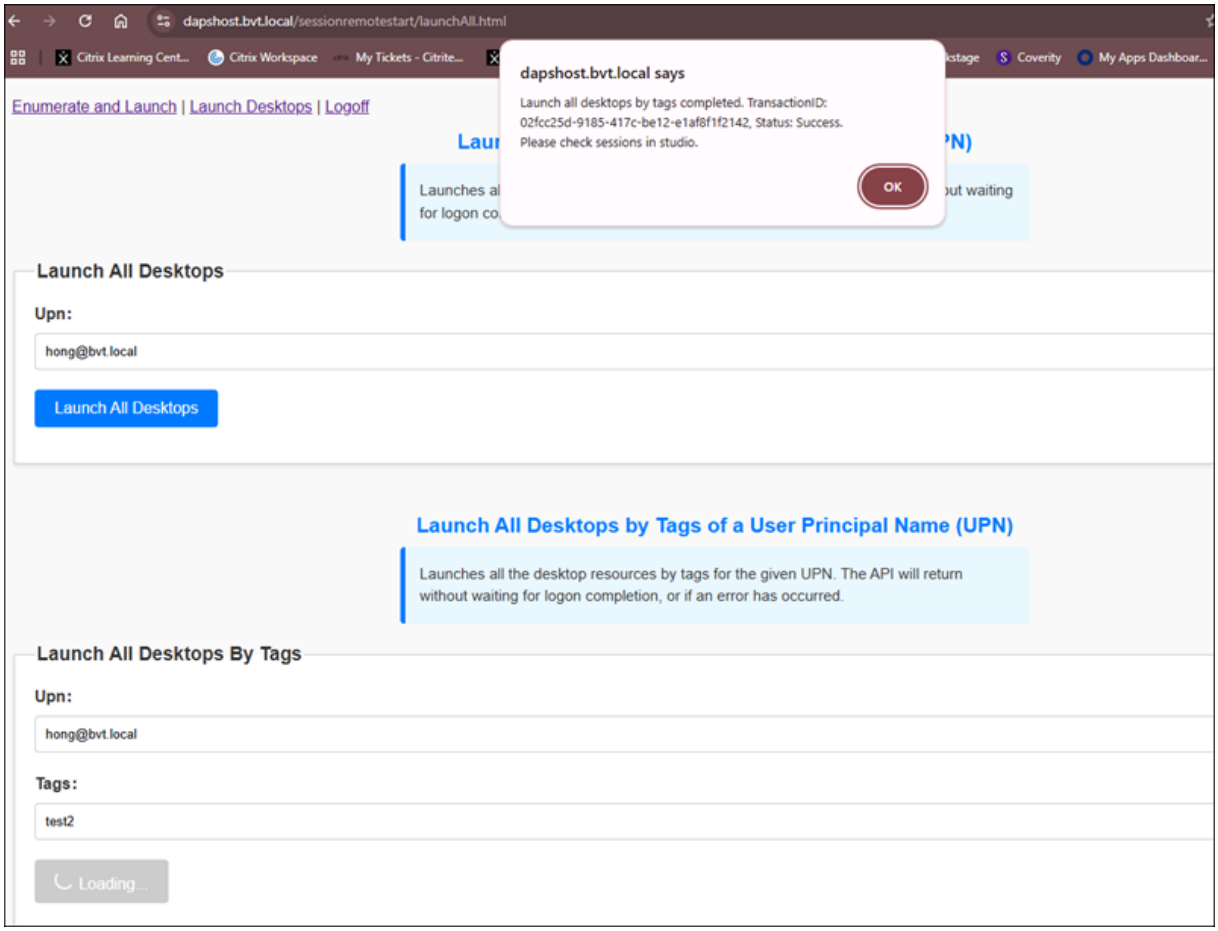
Enter the User Principal Name (UPN) and click **Launch All Desktops**. A prompt will confirm whether the resources launched successfully. Then, verify the sessions in **Citrix Studio**.

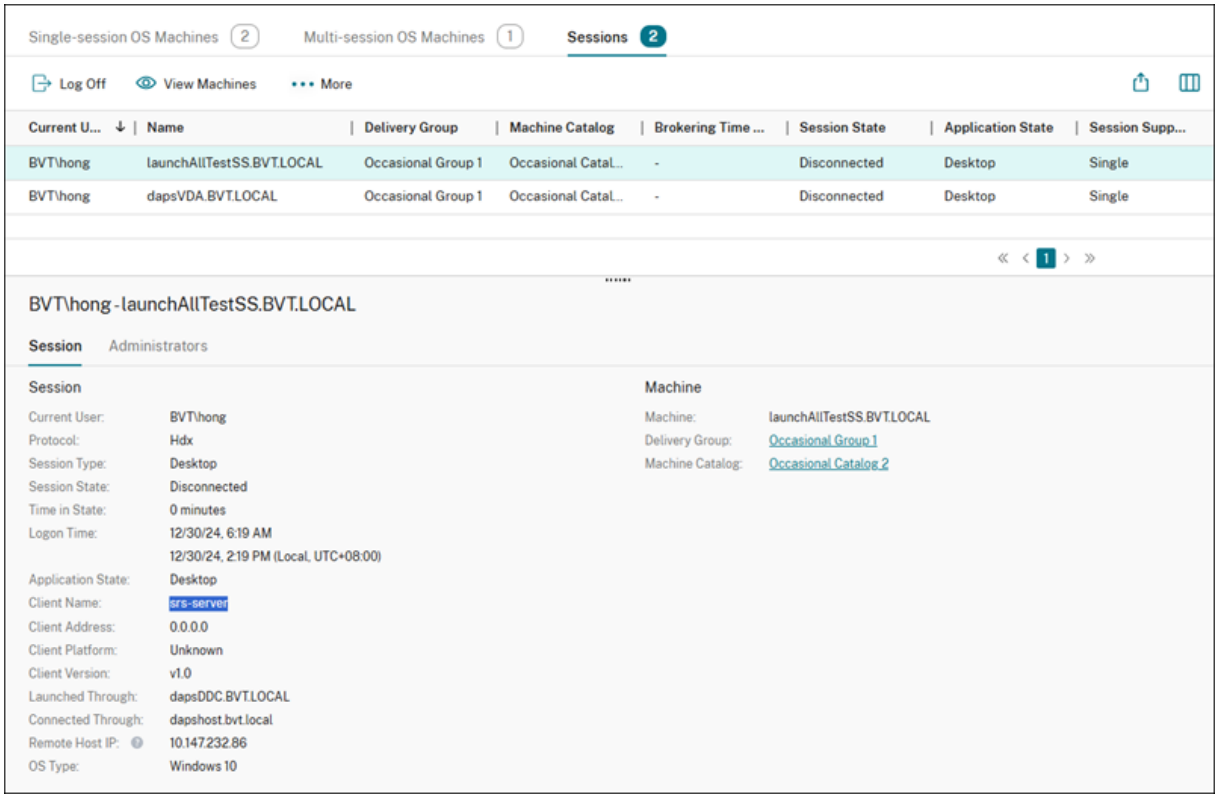




Launch All Desktops by Tags of a User Principal Name (UPN)

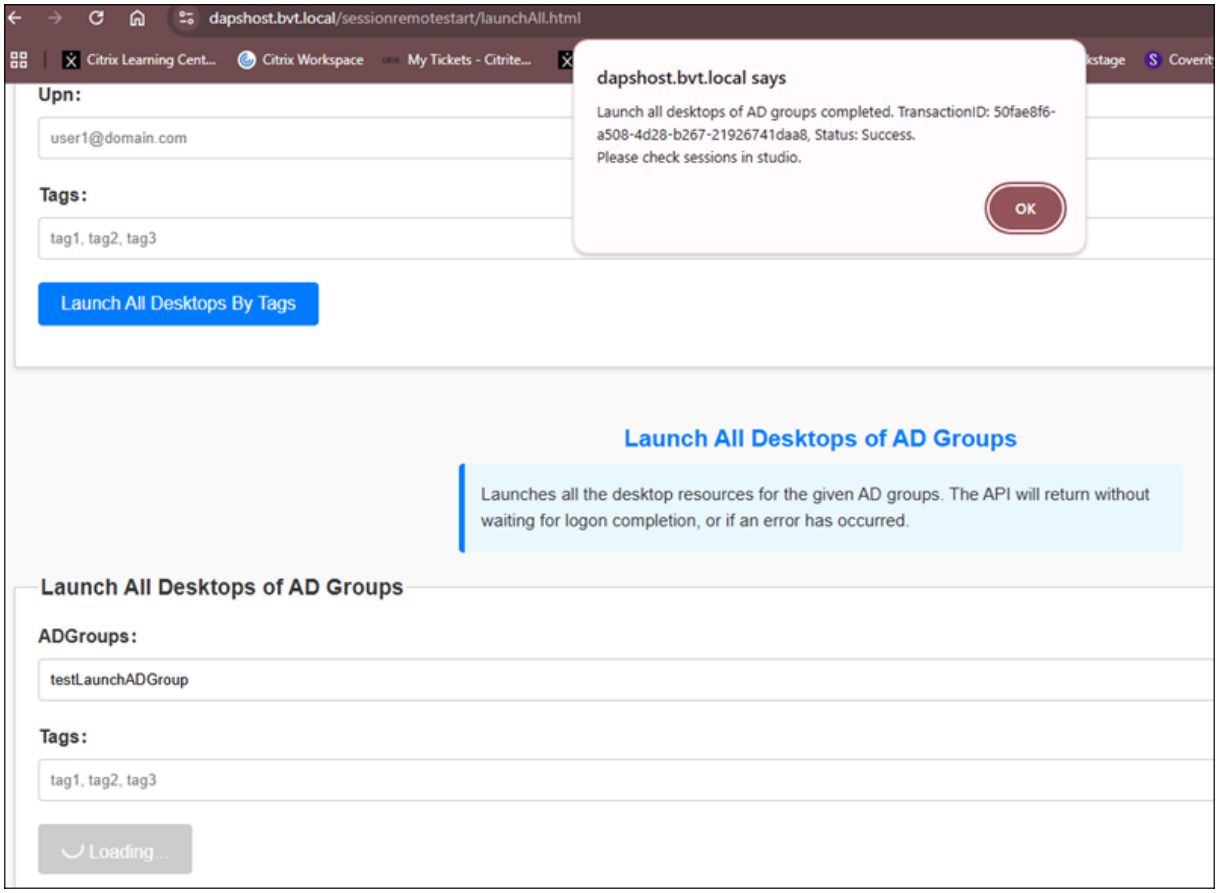
Enter the **User Principal Name (UPN)** and **tags**, scroll down and then click **Launch All Desktops By Tags**. A prompt will confirm whether the resource is successfully triggered. Then, verify the sessions in **Citrix Studio**.

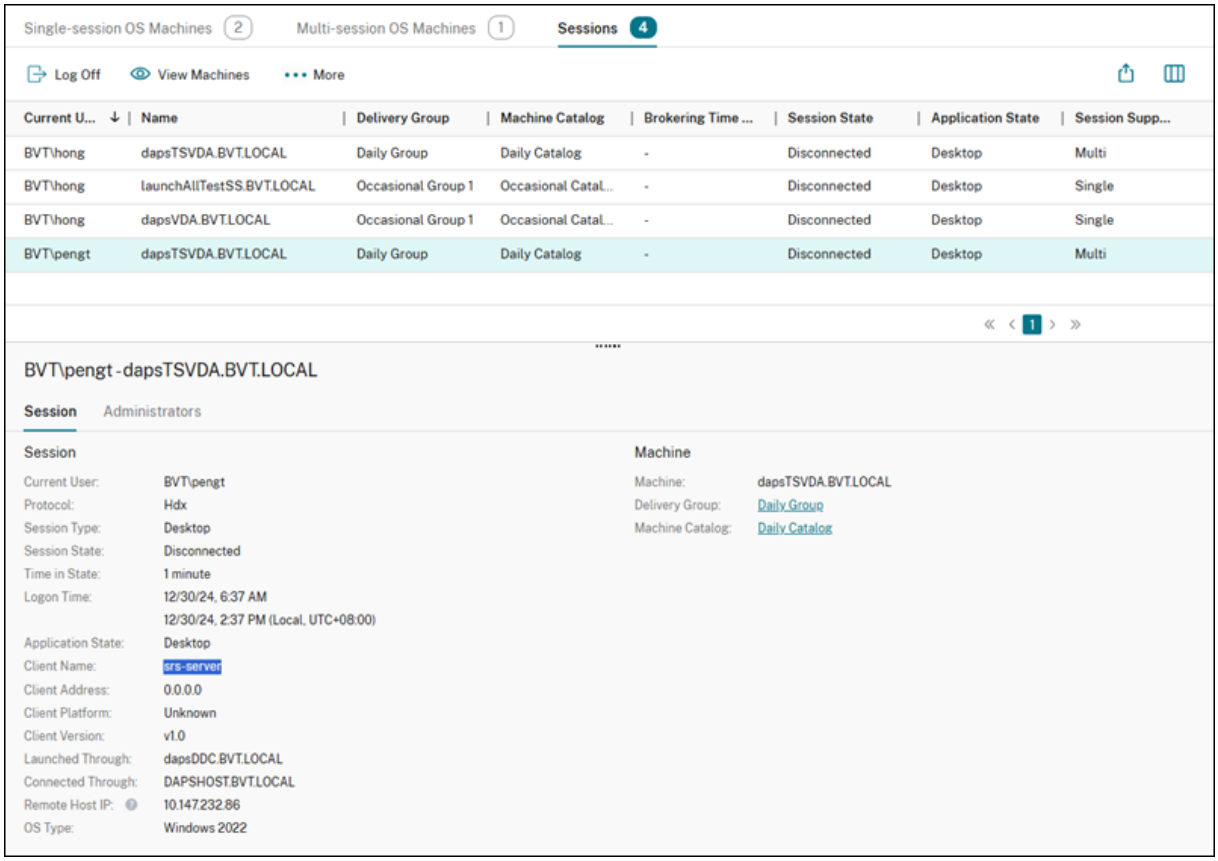




Launch All Desktops of AD Groups

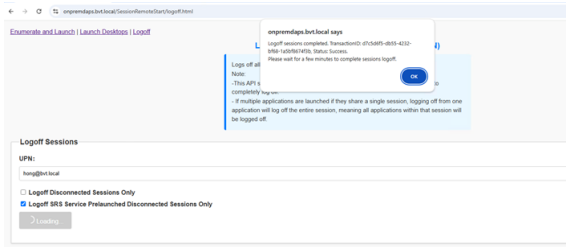
Enter the **AD Groups**, scroll down and then click **Launch All Desktops of AD Groups**. A prompt will confirm if the launch resources successfully triggered. Then, verify the sessions in **Citrix Studio**.





Logoff Sessions of a User Principal Name (UPN)

Enter the **User Principal Name (UPN)** and select whether to log off **disconnected sessions only** or **Session Remote Start service-launched and disconnected sessions only**. Then, click **Logoff Sessions**. A prompt will confirm if the logoff sessions request is successfully triggered. Finally, verify the sessions in **Citrix Studio**.



Verify Session Remote Start API Calls

March 12, 2025

Enumeration and Launch

A sample script is provided in the Citrix download package to verify the installation, configuration, and API calls. Script Name: `enum_launch_example.ps1`

Recommendation:

- Run the script on the third-party Authentication Service host for proper verification.
- The script retrieves and launches resources by calling Session Remote Start APIs.

Execution in PowerShell:

Run the script with two mandatory parameters:

1. Fqdn –The FQDN of the Session Remote Start server
2. UPNs –An array of UPNs and Resource Names, formatted as:
    - `UPN:ResourceName` (example: `user@example.com:Desktop1`)
    - If the resource name is omitted, the first available resource will be launched.

Example:

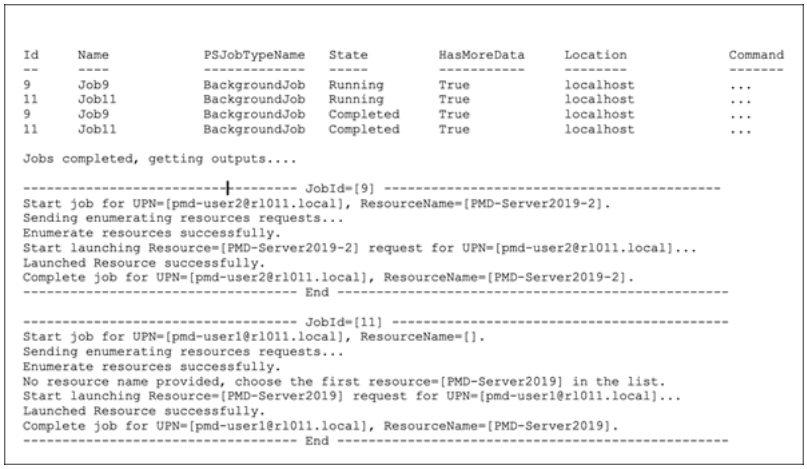
```
.\enum_launch_example.ps1 -Fqdn "srs-server.domain.com"-Upns "SRSuser1@domain
.com:SRS-Server2019","SRSuser2@domain.com"
```

Explanation:

- `SRSuser1@domain.com` is assigned to launch the desktop “SRS-Server2019”(the name as it appears in their workspace).
- `SRSuser2@domain.com` has no specific resource assigned. In this case, the script will automatically launch the first available resource from the user’s resource list.

This ensures that Session Remote Start correctly retrieves and launches desktops for the specified users.

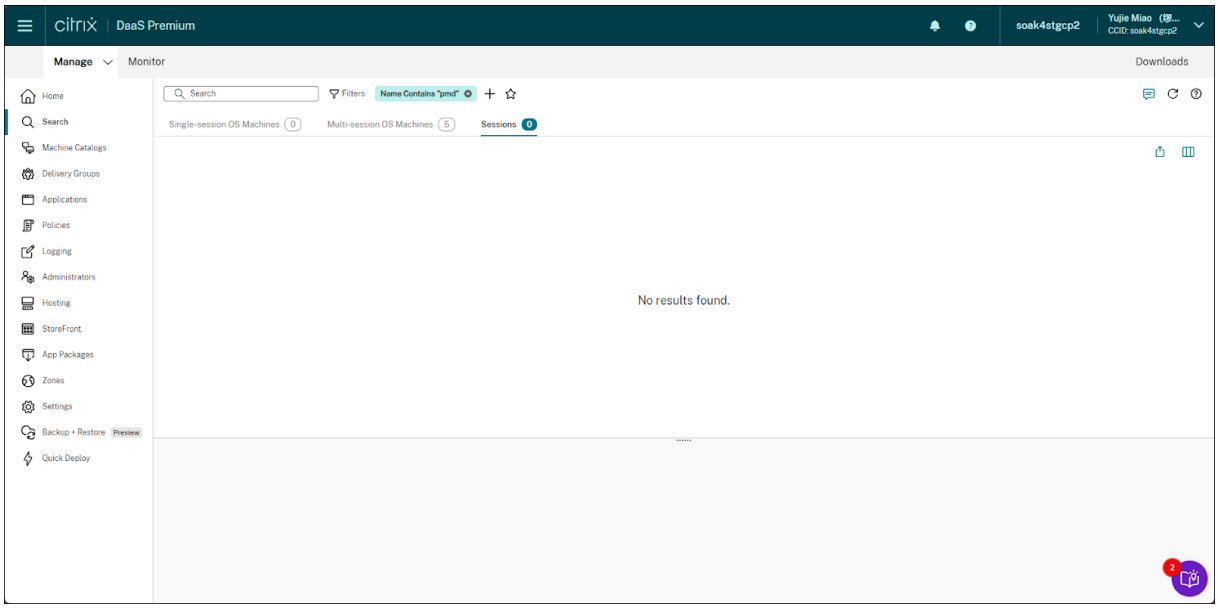
Output:



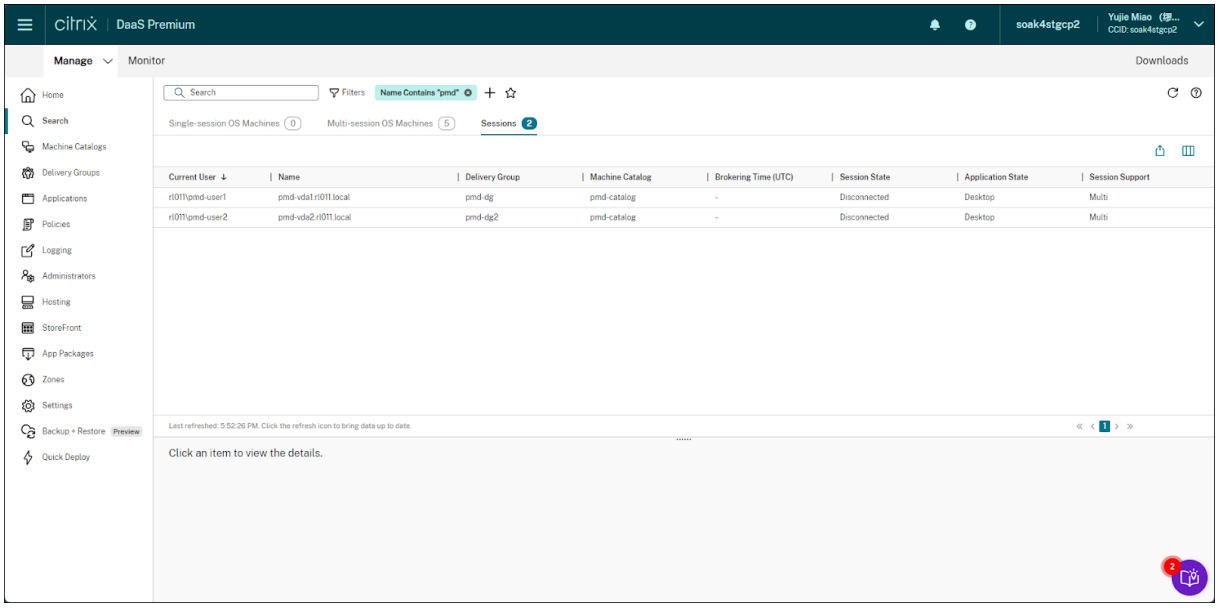
```
Id     Name            PSJobTypeName   State         HasMoreData     Location             Command
--     ----            -------------   -----         -----------     --------             -------
9      Job9            BackgroundJob   Running       True            localhost            ...
11     Job11           BackgroundJob   Running       True            localhost            ...
9      Job9            BackgroundJob   Completed     True            localhost            ...
11     Job11           BackgroundJob   Completed     True            localhost            ...

Jobs completed, getting outputs....

-------------------------+-------- JobId=[9] ----------------------------------------
Start job for UPN=[pmd-user2@rl011.local], ResourceName=[PMD-Server2019-2].
Sending enumerating resources requests...
Enumerate resources successfully.
Start launching Resource=[PMD-Server2019-2] request for UPN=[pmd-user2@rl011.local]...
Launched Resource successfully.
Complete job for UPN=[pmd-user2@rl011.local], ResourceName=[PMD-Server2019-2].
---------------------------------- End ----------------------------------------------

---------------------------------- JobId=[11] ---------------------------------------
Start job for UPN=[pmd-user1@rl011.local], ResourceName=[].
Sending enumerating resources requests...
Enumerate resources successfully.
No resource name provided, choose the first resource=[PMD-Server2019] in the list.
Start launching Resource=[PMD-Server2019] request for UPN=[pmd-user1@rl011.local]...
Launched Resource successfully.
Complete job for UPN=[pmd-user1@rl011.local], ResourceName=[PMD-Server2019].
---------------------------------- End ----------------------------------------------
```


Review the script execution output to ensure resources are pre-launched successfully.

Alternatively, verify in Citrix Studio. Before running the script, no active sessions must be visible for the target machines.



After running the script: The sessions should appear in a disconnected state, confirming that the resources are pre-launched and ready for user reconnection.



Logoff

This script logs off user sessions by calling Session Remote Start APIs. A sample script is provided in the Citrix download package to verify. The Script Name: `logoff_example.ps1`

Running the Logoff Script in PowerShell with two mandatory parameters:

1. Fqdn –The FQDN of the Session Remote Start server.
2. Upns –An array of UPNs (User Principal Names). By default, the script logs off all sessions, regardless of their state.
3. AllDisconnected: It will only log off the disconnected sessions if set to true.
4. SRSLaunchedDisconnectedOnly: Set to **`true`** to log off only sessions launched by the Session Remote Start Service that are currently disconnected.

Example:

```
.\logoff_example.ps1 -Fqdn "srs-server.domain.com"-Upns "SRSuser1@domain.com","SRSuser2@domain.com"-SRSLaunchedDisconnectedOnly $true
```

In this example, the script will logoff all the resources launched by Session Remote Start service and currently in disconnected state for the users `SRSuser1@domain.com` and `SRSuser2@domain.com`.

Output:

```
Id     Name            PSJobTypeName   State         HasMoreData     Location             Command
--     ----            -------------   -----         -----------     --------             -------
9      Job9            BackgroundJob   Running       True            localhost            ...
11     Job11           BackgroundJob   Running       True            localhost            ...
9      Job9            BackgroundJob   Completed     True            localhost            ...
11     Job11           BackgroundJob   Completed     True            localhost            ...

Jobs completed, getting outputs....

---------------------------------- JobId=[9] ----------------------------------------
Start job for UPN=[pmd-user2@rl011.local].
Sending logoff resources request...
Logoff resources successfully.
Complete job for UPN=[pmd-user2@rl011.local].
---------------------------------- End ----------------------------------------------

---------------------------------- JobId=[11] ----------------------------------------
Start job for UPN=[pmd-user1@rl011.local].
Sending logoff resources request...
Logoff resources successfully.
Complete job for UPN=[pmd-user1@rl011.local].
---------------------------------- End ----------------------------------------------
```

> Note:
>
> This API only sends the logoff request and does not wait for the sessions to fully log off.

NetScaler® for Load Balancing Multiple Session Remote Start Servers

November 26, 2025

Session Remote Start supports multiple servers to load balance the requests effectively. We recommend using NetScaler as a Load Balancer.

NetScaler Configuration

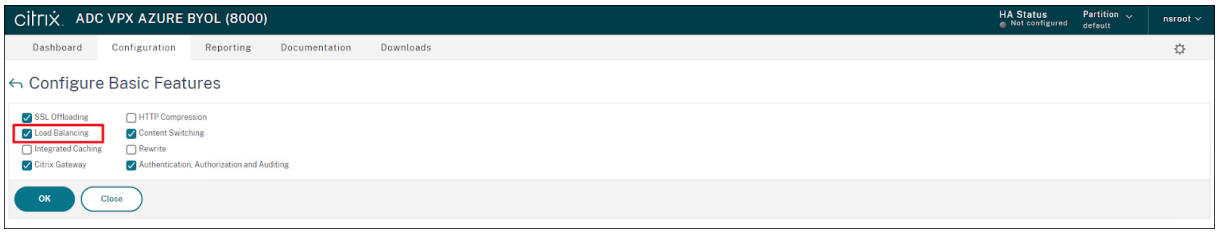
System Requirements for NetScaler as a Load Balancer

The following table lists the minimum requirements.

OS	NetScalerVPX 13.1
Processor	4 or more cores on a compatible 64-bit processor
RAM	Min 8GB
Storage	50 GB

Enable load balancing

Navigate to **System > Settings**, and in **Configure Basic Features**, select **Load Balancing**.

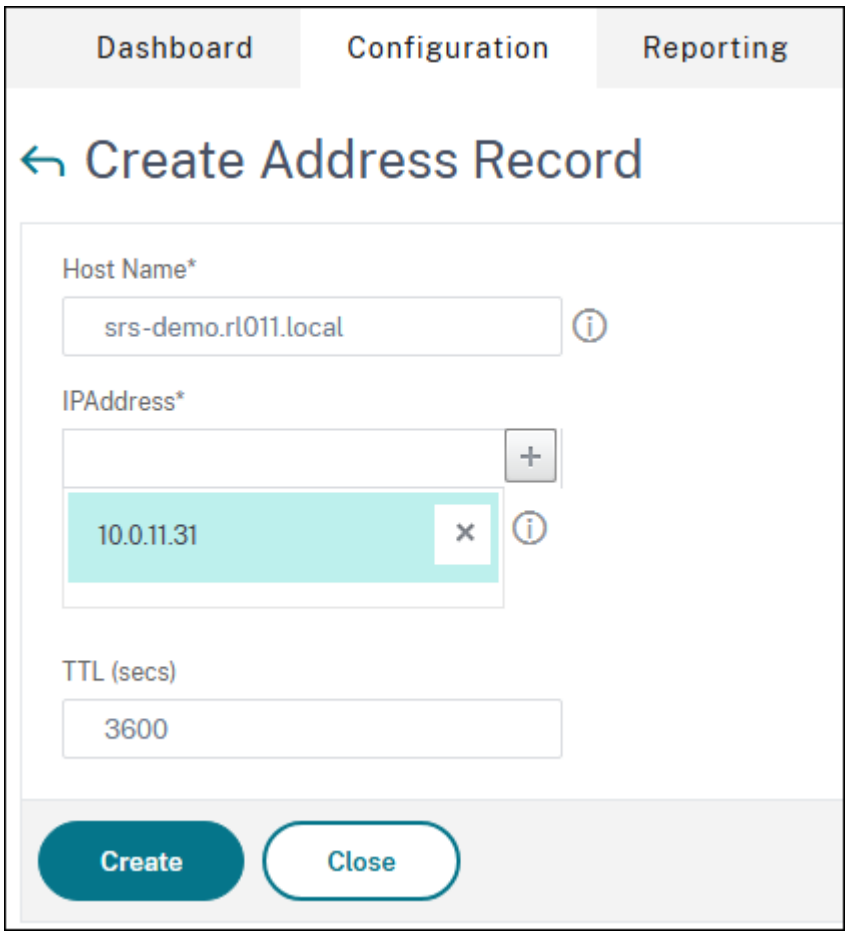


Configure server objects

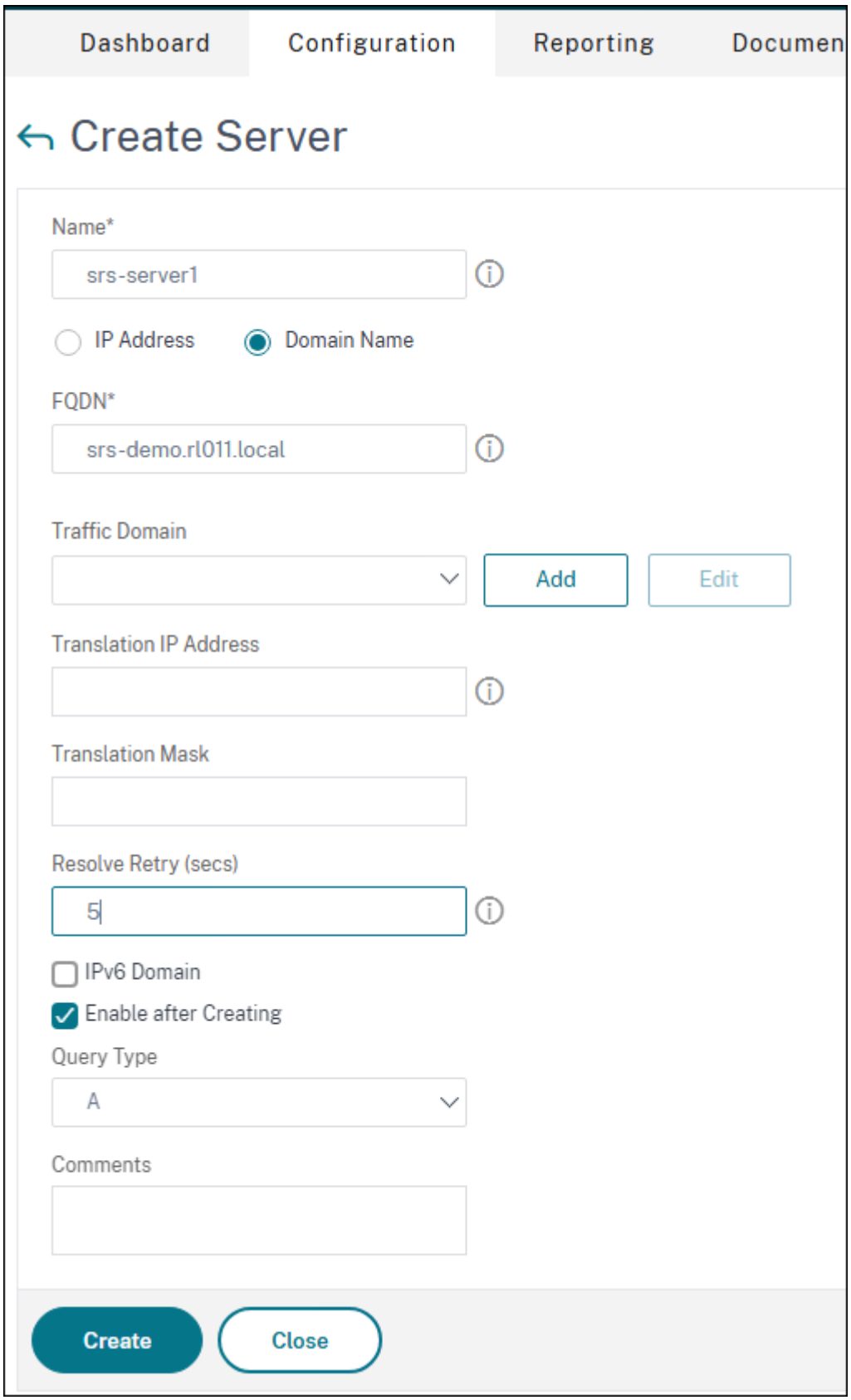
Create an entry for your server on the NetScaler appliance. The NetScaler appliance supports IP address based servers and domain-based servers. If you create an IP address based server, you can specify the name of the server instead of its IP address when you create a service.

1. If you want to specify the name of the server instead of its IP address, add an address record first. Otherwise, skip this step.

   Navigate to **Traffic Management > DNS > Records > Address Records**, and add an address record.



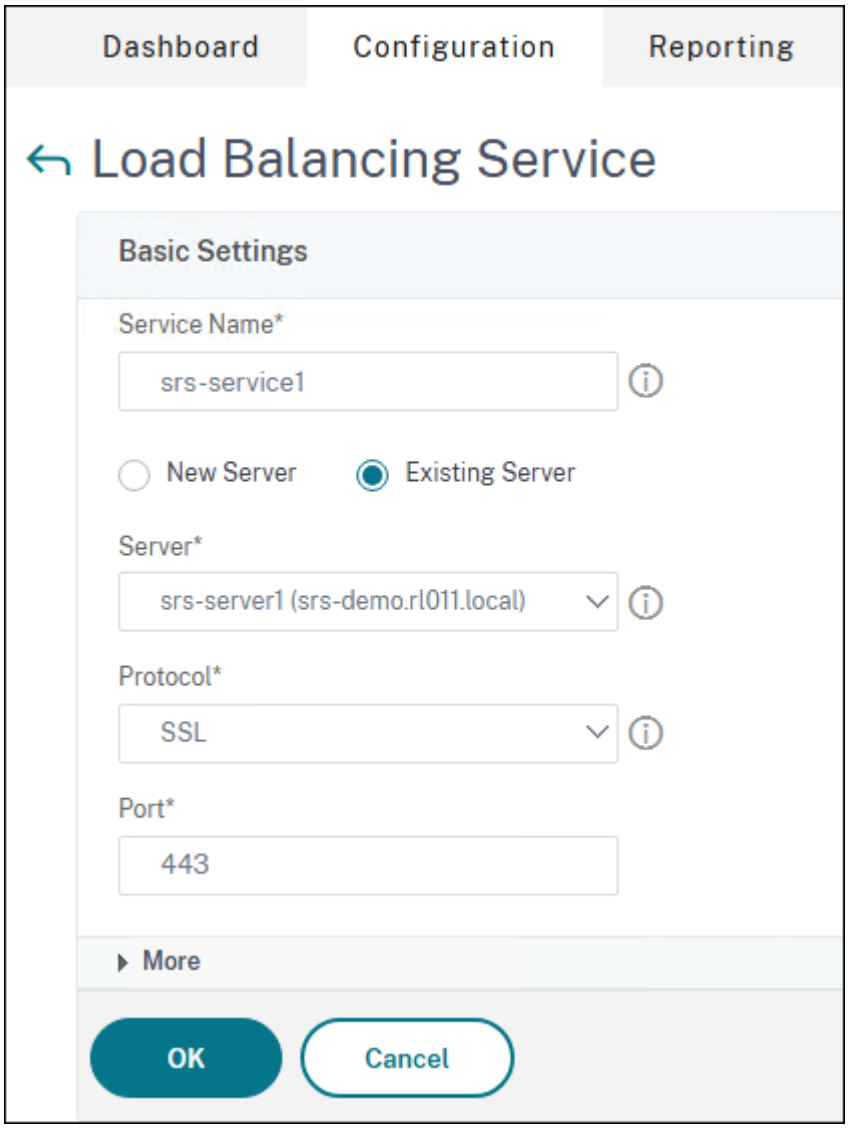
2. Navigate to **Traffic Management > Load Balancing > Servers**, and add a server object.



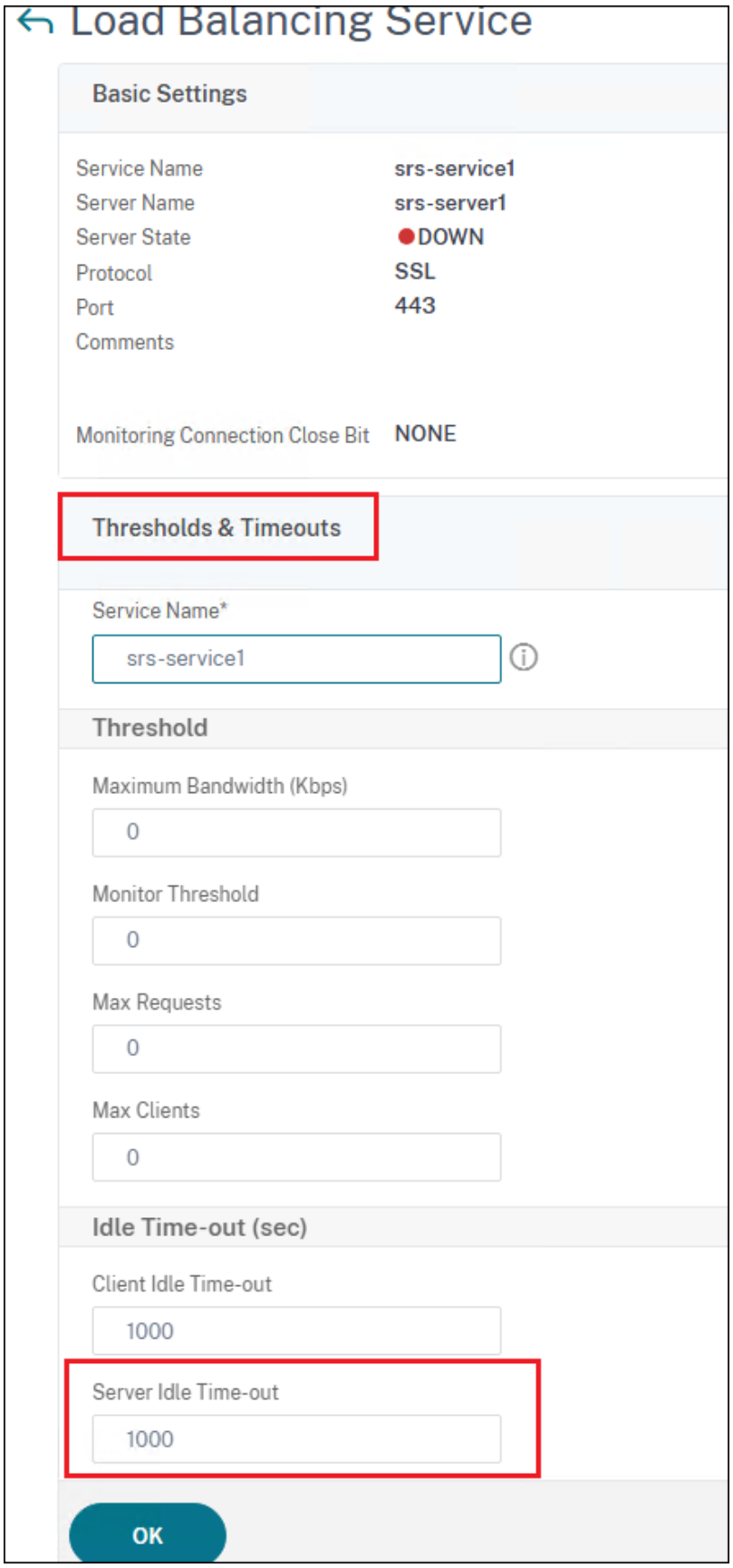
Repeat the two steps if you have multiple servers.

Configure services

1. Navigate to **Traffic Management> Load Balancing > Services**, and add a service.



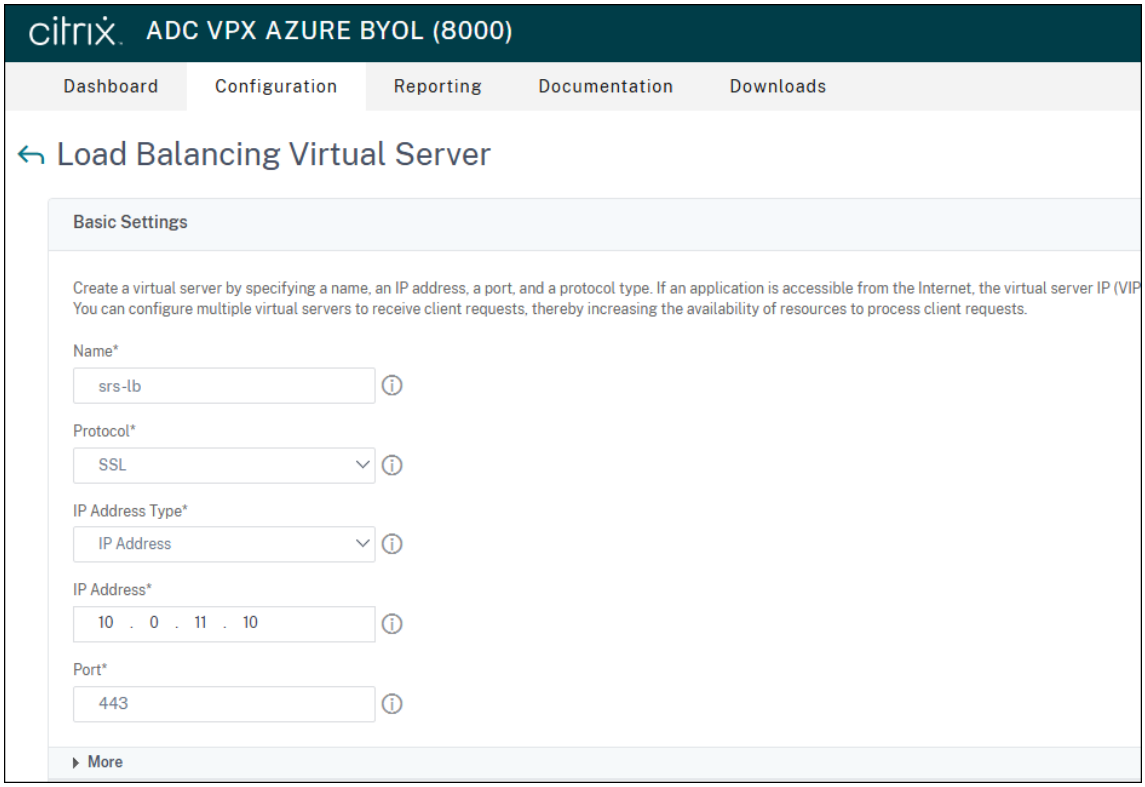
2. In **Thresholds & Timeouts**, modify the value of **Server Idle Time-out** according to the average launching processing seconds in your environment.



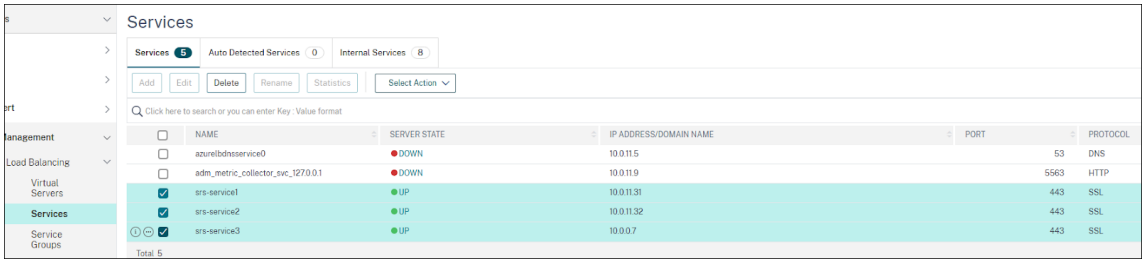
Repeat the operation if you have multiple servers.

Create a virtual server

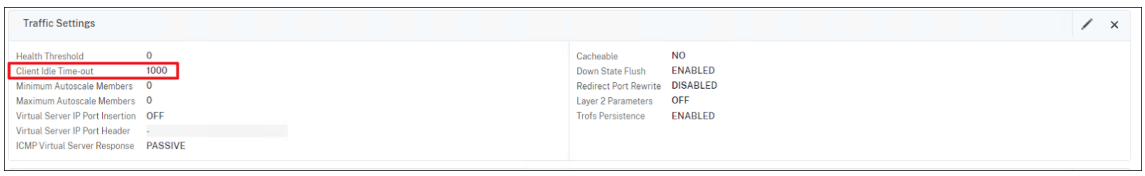
1. Navigate to **Traffic Management > Load Balancing > Virtual Servers**, and then create a virtual server.



2. Bind services to the virtual server.

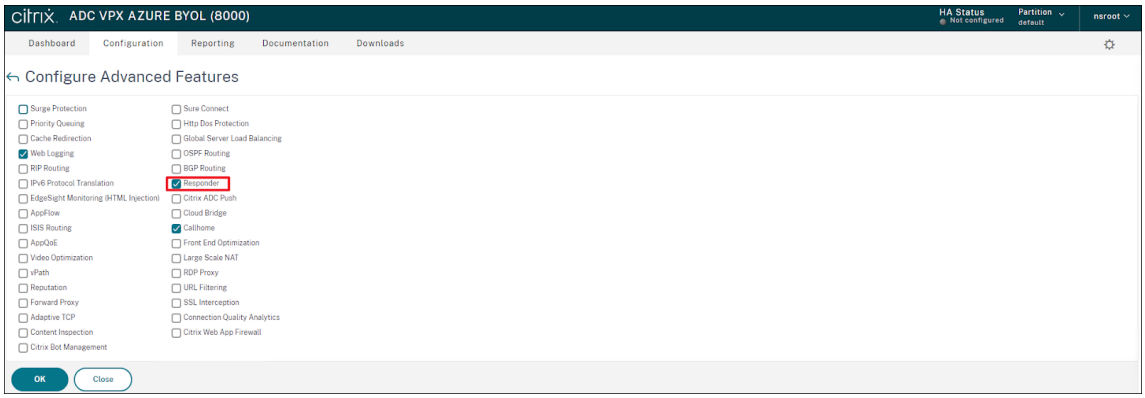


3. In **Traffic Settings**, modify the value of **Client Idle Time-out** according to the average launching processing seconds in your environment.

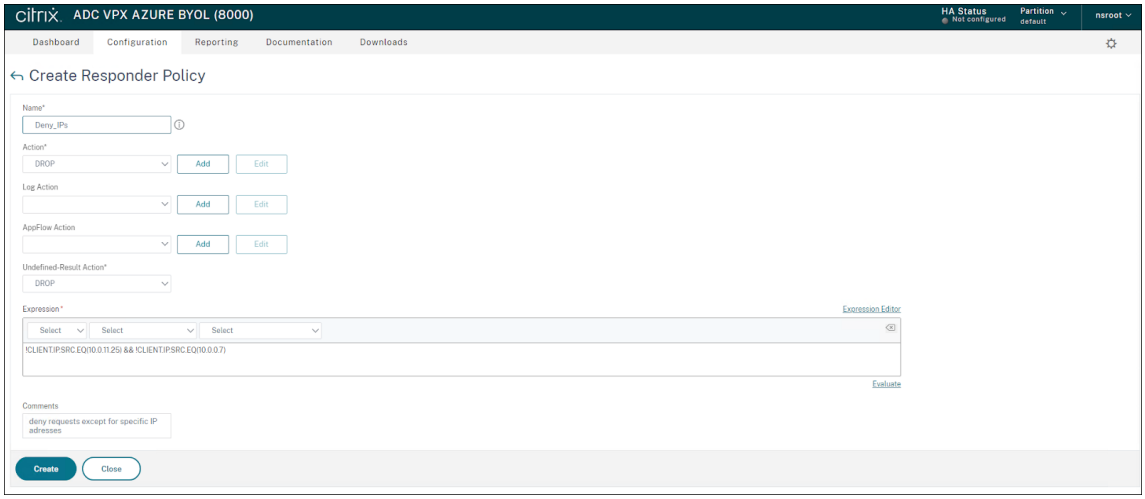


Configure a Responder Policy to drop untrusted IP addresses (Optional)

1. Enable the responder feature. Navigate to **System > Settings**, and in **Configure Advanced Features**, select **Responder**.

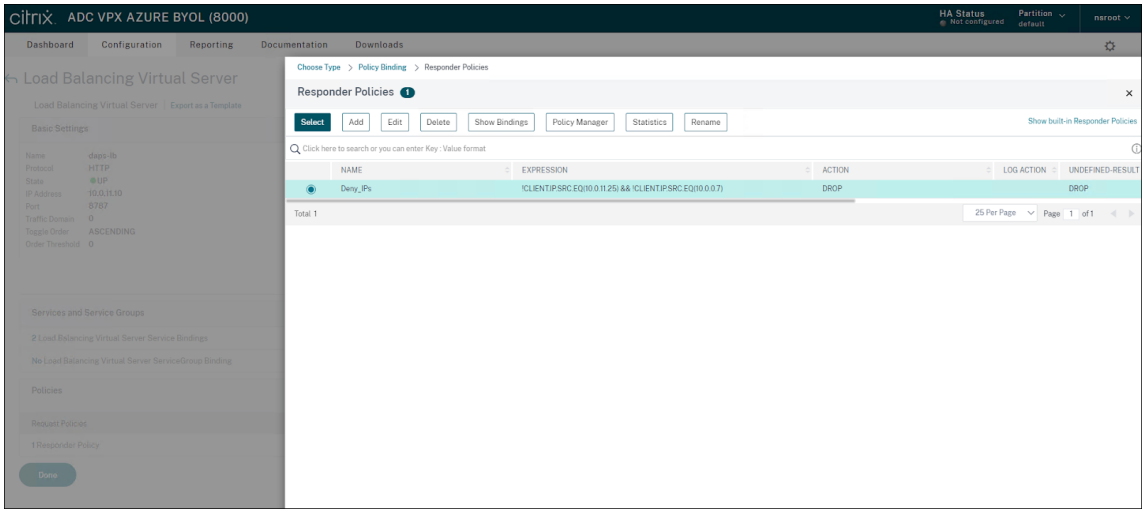


2. Configure a responder policy. Navigate to **AppExpert > Responder > Policies**, and add a policy.



   In this example, the policy drops any request except for the ones from the specific IP addresses - 10.0.11.25/10.0.0.7.

3. Bind policy to the virtual server. Navigate to **Traffic Management > Load Balancing > Virtual Servers**. On the **Load Balancing Virtual Servers** page, select the virtual server to which you want to bind the responder policy, and then click **Open**.



StoreFront™ Configuration

Gateway Configuration

1. Add each Session Remote Start as a gateway following ‘Add Session Remote Start as a Gateway’ section.
2. Apply each gateway to the Session Remote Start store following the ‘Configure Remote Access Settings’section.

StoreFront Plugin Configuration

Add each Session Remote Start URL to ‘srs_server_urls’following the ‘Configure Session Remote Start plugin’section.

Supplementary features

November 26, 2025

1. Telemetry

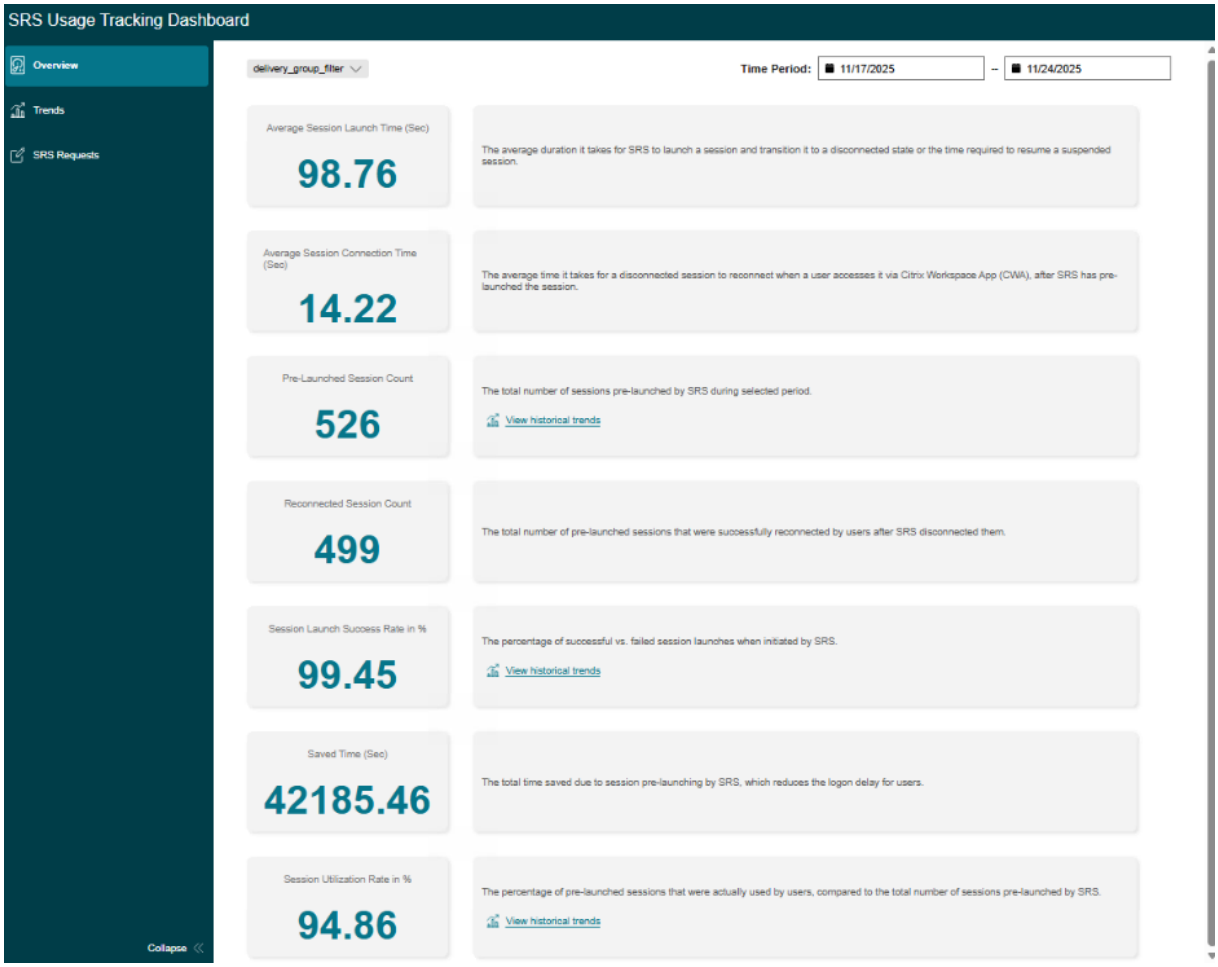
Telemetry is used to **collect and analyze** the usage of **Session Remote Start**. It tracks key metrics, including **detailed request logs, usage trends** (such as request frequency), **success and failure counts** and **Session launch success rate**. This data helps monitor performance, identify issues, and optimize system efficiency.

How to use

Open `https://<SessionRemoteStartFQDN>/SessionRemoteStart/telemetry/` in a web browser, and the overview page will be displayed.

Overview page

The **Overview** page displays data analysis based on seven key metrics: **average session launch time, average session connection time, pre-launched session count, reconnected session count, session launch success rate, total saved time, and session utilization rate**. Additionally, the **Telemetry UI** includes a **time range picker and a delivery group filter**,allowing users to filter data by a selected time period or a selected delivery group.



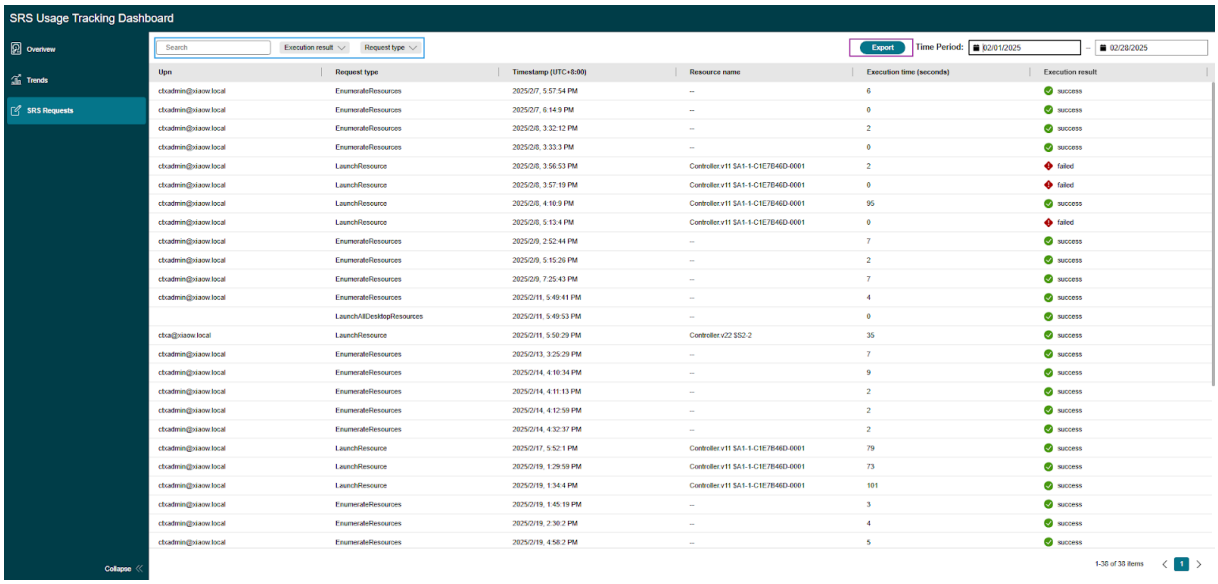
Trends page

Click **Trends** in the left navigation panel to open the **Trends** page. This page tracks SRS usage trends based on three key metrics: **session launch success rate**, **pre-launched session count**, and **session**

utilization rate. Users can also select different time periods or the delivery group to analyze usage patterns.

SRS Requests page:

Click **SRS Requests** in the left navigation panel to open the **SRS Requests** page. This page displays all **SRS operation requests**, allowing users to **search, filter by execution result and request type, and select a time period** for viewing requests. Additionally, users can **export operation requests** based on the applied search and filters.



Configurations

SRS Usage Insights collects the data through the three primary methods:

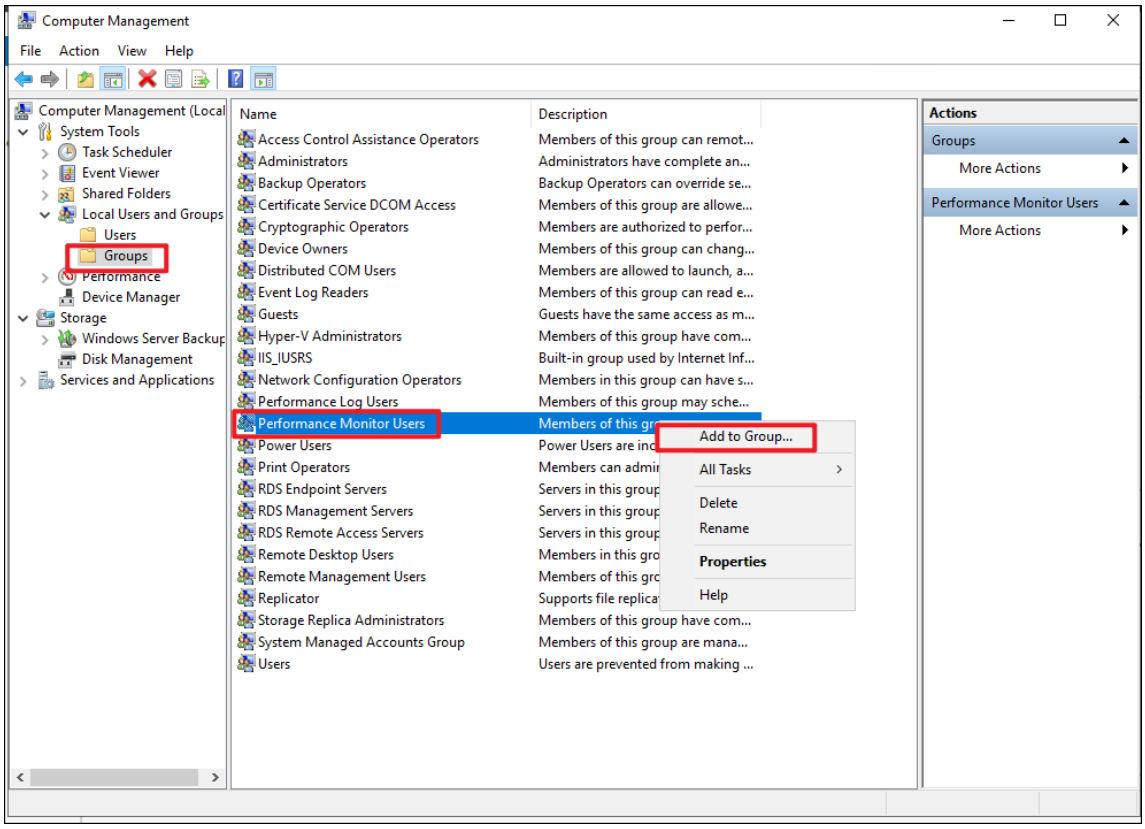
- Actively gathering data while Session Remote Start processes requests and storing it in data files.
- Retrieving data from the Citrix Monitor Service API to supplement relevant information.
- Automatically gathering data for the last N days (N is pre-defined in "AutoTelemetryCollecting-Days", the default value is 30) from local data files and Citrix Monitor Service API. It is executed by a background task every 24 hours.

Local Data Collection and Storage

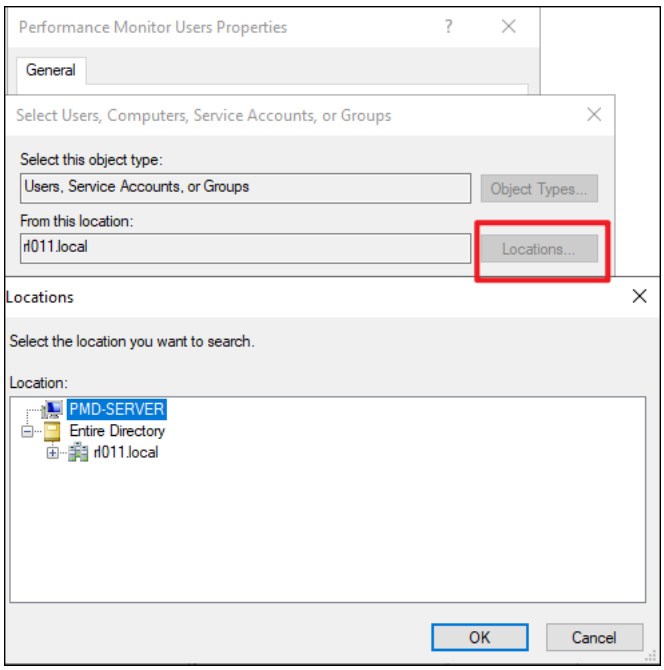
Telemetry server data collection is enabled if "TelemetryCredentialName"is configured. By default, telemetry data is stored in %AppData%\Citrix\SessionRemoteStart\TelemetryData.

Grant Permission to collect Performance Monitor Data

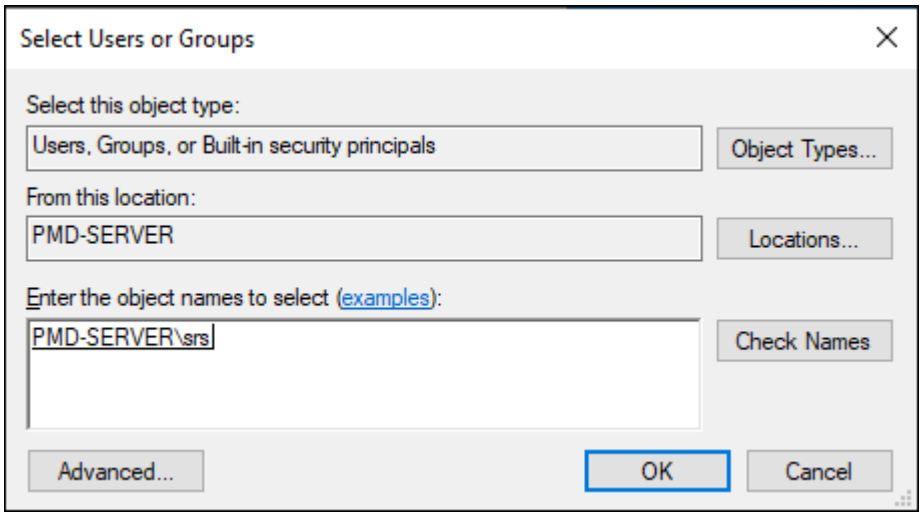
1. On the Session Remote Start server, open Computer Management, select **System Tools > Local Users and Groups > Groups**. Right-click **Performance Monitor Users** on the right panel, and select **Add to Group...**.



2. Click **Add..** and then click **Locations...** and select your local computer. If a domain user is specified as the application pool identity, select the domain instead. For more information, see Configuration for Accessing Other API Services.



3. Input the Session Remote Start application pool identity. For more information, see Configuration for Accessing Other API Services. If the default identity is used, please input `IIS AppPool\SrsAppPool` instead.



4. Open PowerShell, and run the following commandlet:

```
iisreset
```

Retrieving data from the Citrix Monitor Service API

Configure the Citrix Monitor Service API URL and Credential.

1. Test Connectivity and configure the credential. For more information, see Configure the Application Pool Identity for Session Remote Start and Configuration for Accessing Other API Services.
2. Edit `web.config` to set `TelemetryCredentialName` to the credential created in step 1. For more information, see Configuration file.

> **Note:**
>
> For security considerations, it is mandatory to enable TLS for Monitor OData API access. For more information, see Securing On-Premises Monitor OData API Access.

2. Auto Logoff of Idle Sessions Launched by SRS

To conserve resources, Session Remote Start (SRS) can be configured for automatic logoff of pre-launched sessions. If a session is not reconnected within the specified time, the logoff timer can be adjusted to automatically terminate it.

This feature requires:

1. Configure the Application Pool Identity for Session Remote Start and REST API access.
2. Configure `web.config` to enable this feature, and customize the time period. For more information, see Configuration file.

```
<add key="SmartAccessConditions" value=""/>
<add key="AutoLogoffEnabled" value="true"/>
<add key="AutoLogoffIdleMinutes" value="30"/>
<add key="AutoLogoffCheckIntervalMinutes" value="5"/>
<add key="PreviousWebConfigDirectory" value="%AppData%\C
```

3. Session Remote Start plugins for StoreFront™

Overview

During the Session Remote Start pre-launch process, if an end-user attempts to launch the same resource via Citrix Workspace™ App (CWA) or their normal login method, StoreFront may display an error message, which can be unexpected or confusing.

To prevent this, a plugin should be applied to the store that users access via CWA (not the Session Remote Start store). When applied, the plugin will:

1. Add a **Preparing** indicator to the resource name.

2. Hold the launch request if the user clicks the resource and resumes it once pre-launch is complete.

Additionally, the plugins support simultaneous execution with existing plugins without conflicts, meaning that this plugin can work alongside other StoreFront plugins without causing issues or interfering with their functionality.

Steps to add the StoreFront Plugins

Backup Existing Plugins

1. Navigate to the StoreFront Server: Go to the following directory: `C:\inetpub\wwwroot\Citrix\\%StoreName%\bin\`. Ensure this is the store that users access via CWA, not the Session Remote Start store.
2. Backup the existing plugin files: Copy the following files:
   - `StoreCustomization_Input.dll`
   - `StoreCustomization_Enumeration.dll` and paste them into a backup directory, for example: `C:\stf_original_plugin\`. This ensures that you have a backup of the original plugins before making any modifications.
3. Copy and paste the following three DLLs provided by the Citrix team (StoreFrontPlugin folder):
   - `StoreCustomization_Input.dll`
   - `StoreCustomization_Enumeration.dll`
   - `SrsStoreFrontPluginCommon.dll`
4. Set up system environment variables:
   a) Open **System Settings >** open **Control Panel >** click **System >** Click **Advanced system settings** > Click **Environment Variables** >Under the **System Variables** section, click **New ....**
   b) In the **New System Variable** window, enter the required **environment variables** as shown in the table below.
   c) Click **OK**. Close all remaining windows by clicking **OK**.

This ensures that the StoreFront plugin functions correctly with the Session Remote Start integration.

Variable Name	Required/Optional	Description	Example
srs_server_urls	Required	Session Remote Start server URL	`https://srs-server.domain.com/SessionRemoteStart`
stf_original_plugins_path	Optional	Original plugins directory	`C:\stf_original_plugin`
launching_suffix	Optional	Resource title suffix while preparing by Session Remote Start. If not configured, `- Preparing` by default.	`- Preparing`

1. Grant access to `stf_original_plugins_path`. Similar to Session Remote Start file permission configuration, grant access to StoreFront application pool identity.
2. Restart IIS service - `iisreset`.

FAQ

November 26, 2025

Response 404 - UPNNotFound

It may not really be caused by UPNNotFound, but most caused by StoreFront™ authentication for various reasons:

1. Communication error between Session Remote Start and StoreFront:
    a) Open `Web.config`, double confirm the **StoreFrontServer** option. A recommended practice is to set **StoreFrontServer** by copy from **Receiver from Web Sites** in StoreFront UI. Please note that the StoreFront store name is case-sensitive.
    b) Try to open **StoreFrontServer** by browser, check if there is a connection or certificate error. (Do not need to login, just check if the web page can be visited normally.)

c) Open CDF trace tool or enable Session Remote Start log file, check if there is any error detail.

2. StoreFront issue:
   Open **Event Viewer > Applications and Services Logs > Citrix Delivery Services** to check StoreFront log
3. FAS issue:
   For step 2, the error may be related to FAS, please refer to the FAS section of the checklist above.

Request Timeout

- Session Remote Start has a **RequestTimeoutSeconds** option in `Web.config`. When request duration reaches the timeout, Session Remote Start will respond with status code 408 and body content.
- Session Remote Start will never terminate http connection without response. Please check if there is any timeout related setting in the caller.
- For the **LaunchResource** API, there is an optional parameter `WaitForLogonNotification` recognized as **`true`** by default. For the VDAs really take a long time for launching, `WaitForLogonNotification` can be set to false. API will respond immediately and execute the launching process in the backend.

Remain in 'Connected' State

If a session remains in the **Connected** state, there may be an authentication issue.

- Please review the FAS configuration and check the Session Remote Start logs.
- In the DaaS environment, please check if the Cloud Connector is working properly. Pay special attention to whether the firewall is configured correctly and does not block traffic from StoreFront.

Remain in 'Active' State

If a session remains in the **Active** state, the logon script configured in the Group Policy on AD may have failed to trigger the disconnect operation. Verify the Group Policy configuration, especially whether it has been successfully applied to the VDA.

Double Hop Sessions

SRS functions as expected in a double hop scenario without requiring any additional configuration. However, if domain pass-through is enabled in the double hop setup, features like HTTP Basic Authen-

tication should be avoided on the second hop, as they can cause unexpected behavior.

Known issues

November 26, 2025

1. SRS does not support Linux or MAC VDA.
2. The **Launch All Desktops by Tags** feature has a limitation when configured with any of the following two settings:

   a) When a tag is applied to a **Static Delivery Group**.
   b) If the desktops within a delivery group are assigned to an Active Directory (AD) group instead of individual users.

   The feature will not be able to retrieve and launch those desktops.
3. If a session is resumed from a hibernated state by Session Remote Start (SRS), the auto logoff feature cannot currently identify it as a pre-launched session. As a result, the session will not be logged off automatically if configured to logoff.
4. SRS cannot log off resumed sessions if they are not reconnected, regardless of how they were resumed.
5. If the SRS app uses the default application pool SrsAppPool, the configuration of this pool will be reset.
6. When load balancing StoreFront with NetScaler ADC, the persistence type only supports `SOURCEIP`. `COOKIEINSERT` is not supported.
7. Due to a limitation in StoreFront versions prior to 2507 (i.e., 2503 and earlier), in a hybrid scenario, if StoreFront has Site Aggregation enabled, when launching a session, the SRS would incorrectly trigger an unnecessary launch operation because it fails to query/find the already-launched session when checking for its existence.
8. When upgrading StoreFront, if the SRS plugin has been previously applied, **it is required** to first restore the plugin, complete the StoreFront upgrade, and subsequently re-apply the plugin.

Upgrade Instructions

November 26, 2025

When updating SRS to the latest version, follow these instructions on both the SRS and StoreFront servers.

SRS server

1. Backup `Web.config`.
2. Backup Application Pool.
3. Execute `%windir%\system32\inetsrv\appcmd list apppool /config /xml > AppPoolsBackup.xml` or other commands that can record the app pool configuration.
4. Run `SessionRemoteStartSetup.exe` to perform an in-place upgrade.
5. Recover Application Pool.
6. Double-check the `Web.config` file to ensure the configuration is properly merged.

StoreFront

1. Replace the StoreFront Plugin: `StoreCustomization_Enumeration.dll`.

1.0.7 to 1.0.8

SRS Server

1. Backup `Web.config`.
2. Backup Application Pool.
   Execute `%windir%\system32\inetsrv\appcmd list apppool /config /xml > AppPoolsBackup.xml` or other commands that can record the app pool configuration.
3. Run `SessionRemoteStartSetup.exe` to perform an in-place upgrade.
4. Double-check the `Web.config` file to ensure the configuration is properly merged.
5. Double-check the Application Pool to ensure the configuration is not changed.

1.0.8 to 1.0.9

SRS Server

1. Backup `Web.config`.
2. Backup Application Pool.
   Execute `%windir%\system32\inetsrv\appcmd list apppool /config /xml > AppPoolsBackup.xml` or other commands that can record the app pool configuration.
3. Run `SessionRemoteStartSetup.exe` to perform an in-place upgrade.
4. Double-check the `Web.config` file to ensure the configuration is properly merged.
5. Double-check the Application Pool to ensure the configuration is not changed.

Optional Configurations

November 26, 2025

1. Configure the Application Pool Identity for SRS

Overview

By default, Session Remote Start (SRS) runs under the SrsAppPool application pool, with its identity set to `ApplicationPoolIdentity`. However, if specific features require elevated permissions, the identity must be configured as a domain user or local user to ensure proper functionality.

Prepare the Identity

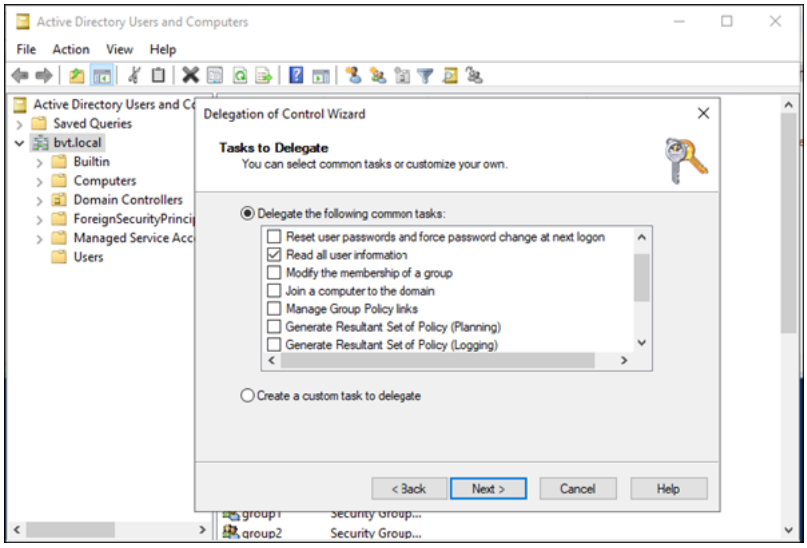
Domain User:

A domain user is required for the **Launch All Desktops by AD Group** feature. For security purposes, it is recommended to create a dedicated domain user with minimal access rights.

On the Domain Controller, create a user or use an existing Domain Service Account.

Open the Active Directory Users and Computer.

1. In the left pane, right-click your domain and select **Delegate Control** from the context menu to open the **Delegation of Control Wizard**.
2. Add/Select the domain user created above and grant the **Read all user information** permission.



Local user:

A local user account on SRS server is required for the following features to work.

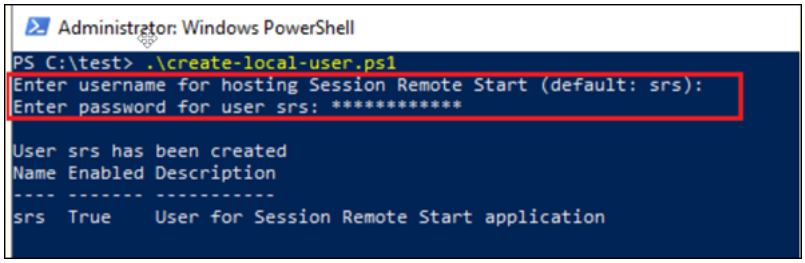
1. Launch All Desktops by Tags
2. Log off VDAs by a specified UPN (if they were pre-launched by SRS and not reconnected).
3. Support pre-launch from a hibernated VM.
4. Automatically log off VDAs pre-launched by SRS if they remain unconnected within the specified time.
5. Telemetry data collection and analysis.

To enable the above features, a local user must be created on SRS server and configured with the necessary permissions.

Two options are available to create the local user:

- Reuse an existing standard local user.
- Create a new one by running `create-local-user.ps1` on the SRS server from the installation package downloaded from Citrix Downloads, as an administrator.

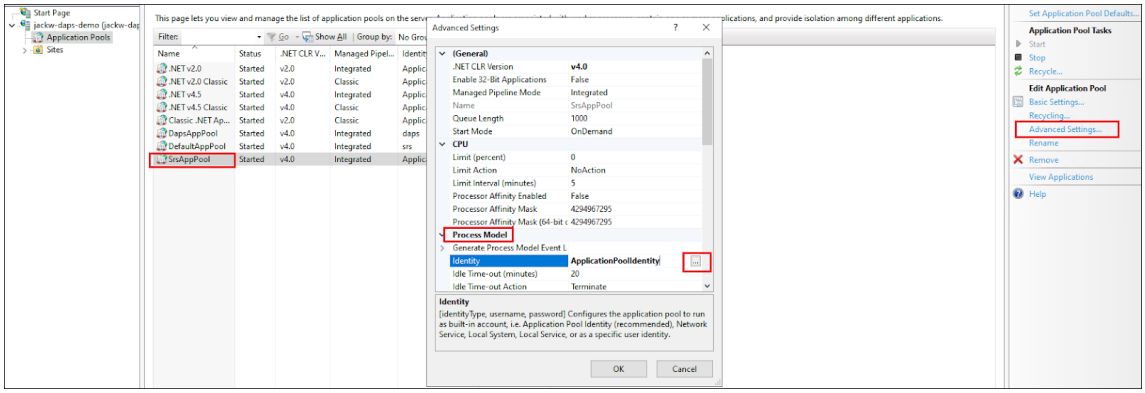
The script will create a standard local user with general access.



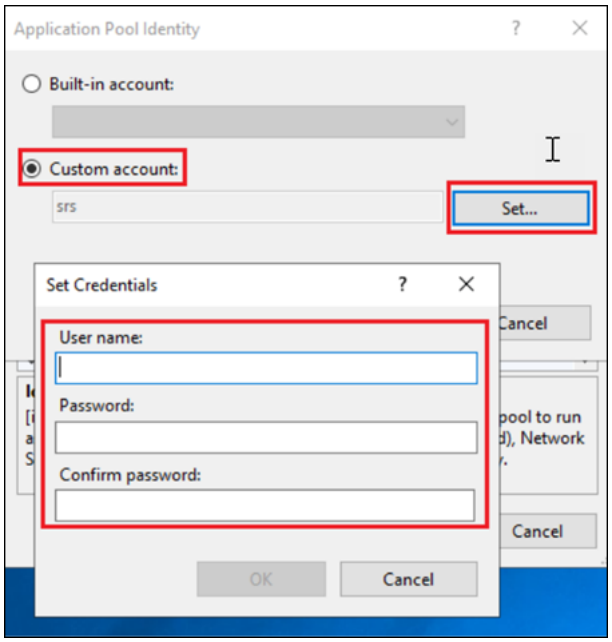
If none of the features listed above are required, you can skip the following steps in this section.

Configure Application Pool Identity

1. On the **SRS server**, open **IIS Manager**, from the **Application Pools**, highlight **SrsAppPool** and on the right side, select **Advanced Settings** under the **Edit Application Pool**. Scroll down to **Process Model > Identity** click on the three dots.



2. Select **Custom account**, click **Set**, and input the username and password of the user created above (domain or local user) to host Session Remote Start.



3. Ensure the Application Pool's `setProfileEnvironment` attribute is enabled.

   a) Navigate to the `%windir%/system32/inetsrv/config` folder.
   b) Open the `applicationHost.config` file.
   c) Locate the `<system.applicationHost><applicationPools><SrsAppPool><processModel>` element.
   d) Confirm that the `setProfileEnvironment` attribute is not present, which defaults the value to true, or explicitly set the attribute's value to true.

2. Configuration for Accessing Other API Services

Overview

The following Session Remote Start (SRS) features require CVAD 2308 or later and Citrix DaaS™ REST APIs to be enabled and supported:

- Launch All Desktops by Tags.
- Log off VDAs by a specified UPN (if pre-launched by SRS and not reconnected).
- Support pre-launch from a hibernated VM.
- Automatically log off VDAs pre-launched by SRS if they remain unconnected within the specified time.

Additionally, Citrix Monitor Service API requires CVAD 2305 or later for the following feature:

- Telemetry

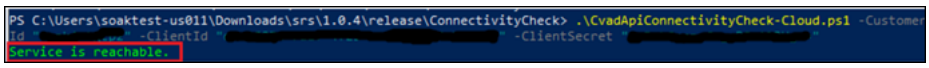
If none of the features listed above are required, you can skip the following steps in this section.

Connectivity test

Firstly, verify whether Session Remote Start (SRS) can communicate with the API services. These services are typically hosted on the Delivery Controller™ (DDC).

Execute the below test scripts in the installation package:

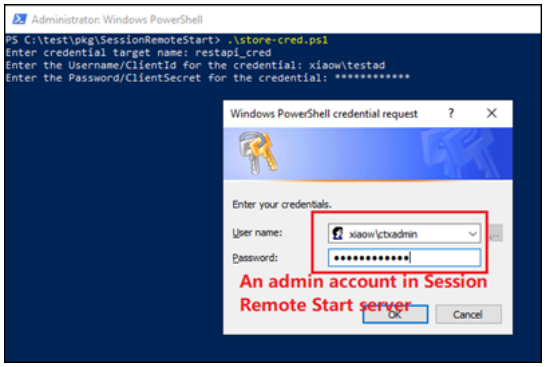
- For CVAD REST APIs (On-Prem), execute the `CvadApiConnectivityCheck-OnPrem.ps1`.
- For Citrix DaaS REST APIs, execute the `CvadApiConnectivityCheck-Cloud.ps1`.
- For Citrix Monitor Service API (On-Prem), execute the `MonitorServiceApiConnectivityCheck-OnPrem.ps1`. TLS for Monitor OData API access should be enabled. See Securing On-Premises Monitor OData API Access.
- For Citrix Monitor Service API (DaaS), execute the `MonitorServiceApiConnectivityCheck-Cloud.ps1`.



Configure the credentials

Session Remote Start (SRS) requires credentials to authenticate and issue API requests to external API services. These credentials are securely stored under a user account in the Windows Credential Manager to ensure secure access and authorization.

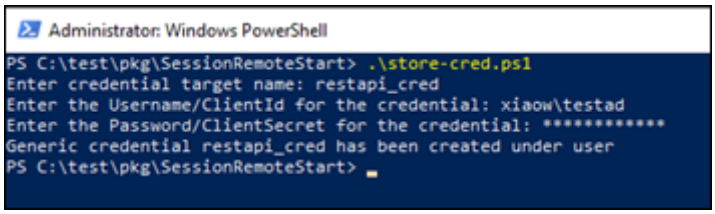
In the installation package, navigate to the **SessionRemoteStart** folder. Locate the PowerShell script `store-cred.ps1` and execute it as an administrator to configure the required credentials for Session Remote Start (SRS). This will securely store the credentials in **Windows Credential Manager** for API authentication.



For On-Prem environment, provide the admin credential. For more information, see here.
For the DaaS environment, provide the client identity. For more information, see here.

> **Note:**
>
> Different credential configurations can share the **same actual credential**. If multiple configurations refer to the same credential, only one entry needs to be created in **Windows Credential Manager**, and all configurations can reference it. This eliminates the need for duplicate credential entries.

A success message will be displayed after creation is complete.



3. Log configuration

Log Level Configuration

Currently, Session Remote Start logging supports CDF and File tracing. CDF logging is always enabled by default. File tracing availability depends on the configuration file settings.

Trace severity level

- 0 - Critical
- 1 - Urgent
- 2 - Significant

- 3 - Important
- 4 - ImportantDetailed
- 5 - Informational
- 6 - InformationalDetailed
- 7 - Notable
- 8 - NotableDetailed
- 9 - Insignificant

Log to File

Logging to file is enabled by default. To disable, set **LogFileName** in **Web.config** to empty.

The default log path is shown below, where **{Session Remote Start user}** should be replaced with the user hosting the SRS service. If the identity has been changed to another identity, replace **Session Remote Start user** with the corresponding user directory.

`"C:\Users\\{ Session Remote Start user } \AppData\Roaming\Citrix\SessionRemoteStart\Logs\SessionRemoteStart.log"`.

Customers can change the log file location and permissions as per their requirements, please see the steps required to change the Log file location and permissions.

4. Log Rotation Configuration

New Log Rotation Settings in `web.config`

Two new configuration parameters have been added to the web.config file to support log rotation: **MaxLogFileSizeMB** and **MaxBackupLogFiles**.

- **MaxLogFileSizeMB**: Sets the maximum size (in megabytes) for a log file. When this limit is reached (default: **100 MB**), the application automatically compresses and moves the log file to the `Archive` subfolder within the log directory. A new, empty log file is then created to continue logging.
- **MaxBackupLogFiles**: Specifies the maximum number of archived log files to retain. When the number of backups exceeds this limit (default: **10**), the oldest archived files are automatically deleted to make space for new ones.

5. Log to Event Viewer

Session Remote Start supports logging key operations to the Windows Event Log.

- Log Path: `Windows Logs > Application`
  (File location: `%SystemRoot%\System32\Winevt\Logs\Application.evtx`)
- Event Source Name: `"SessionRemoteStart"`

This allows administrators to monitor and troubleshoot Session Remote Start events through the **Windows Event Viewer**.

To enable this feature, set `LogToEventViewer` to **`true`** in `web.config`. (See Configuration file).

Event types:

Event ID	Description
1	Requests for enum, launch, and logoff operations.
2	Responses to enum, launch, and logoff requests.
3	Authentication requests from StoreFront™ to SRS.
4	Session launches via ICA® file.
5	Requests sent to StoreFront.
6	Requests sent to REST API.
7	Changes to Web.config.
8	Queries for groups/users under an AD group.
9	Reads from Windows Credential Manager.

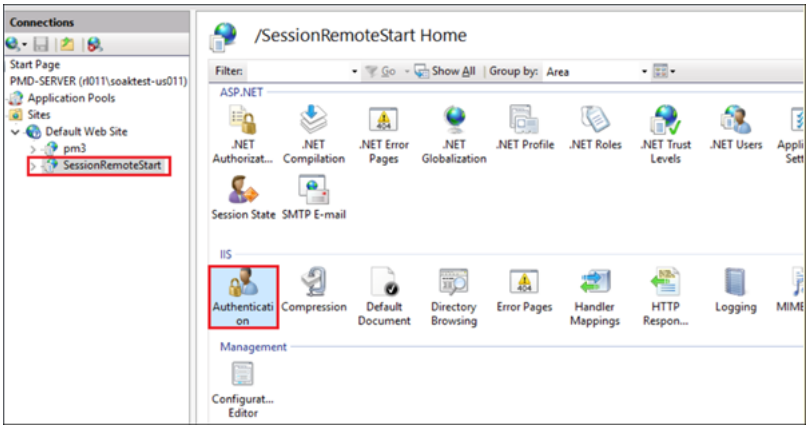
6. Change Log File/Telemetry File Location

If the Log File location is changed, make sure that the Session Remote Start service has the necessary permissions to modify the log file.

Log file permissions

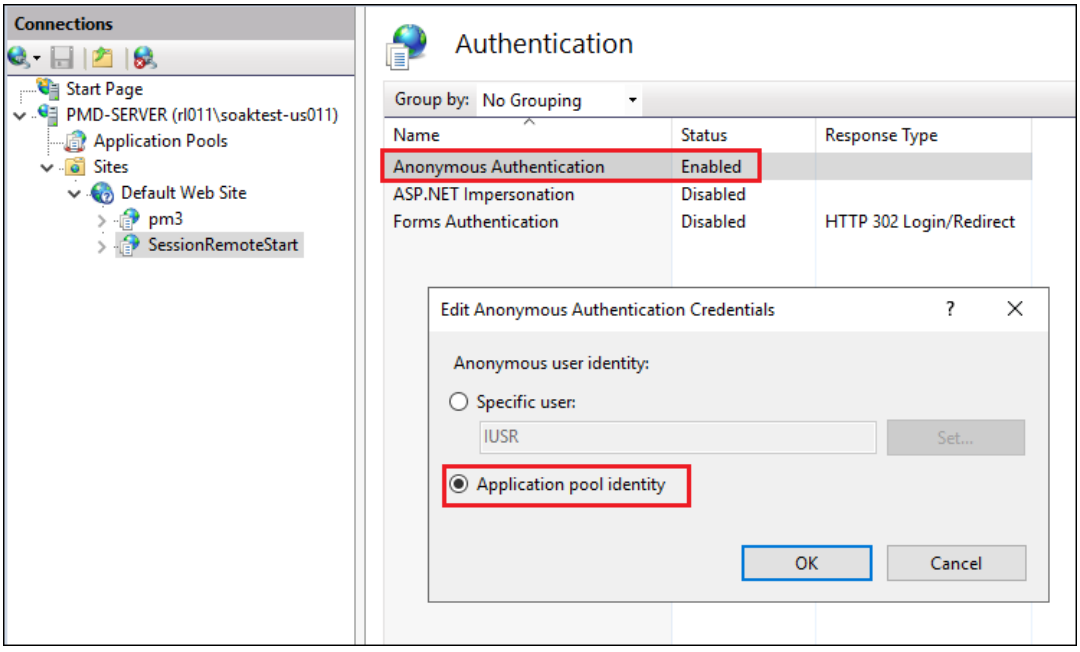
1. Set an anonymous user identity to Application Pool Identity.

   a) Open **IIS Manager > Sites > Default Web Site > SessionRemoteStart** and then open **Authentication** under **SessionRemoteStart Home** page.



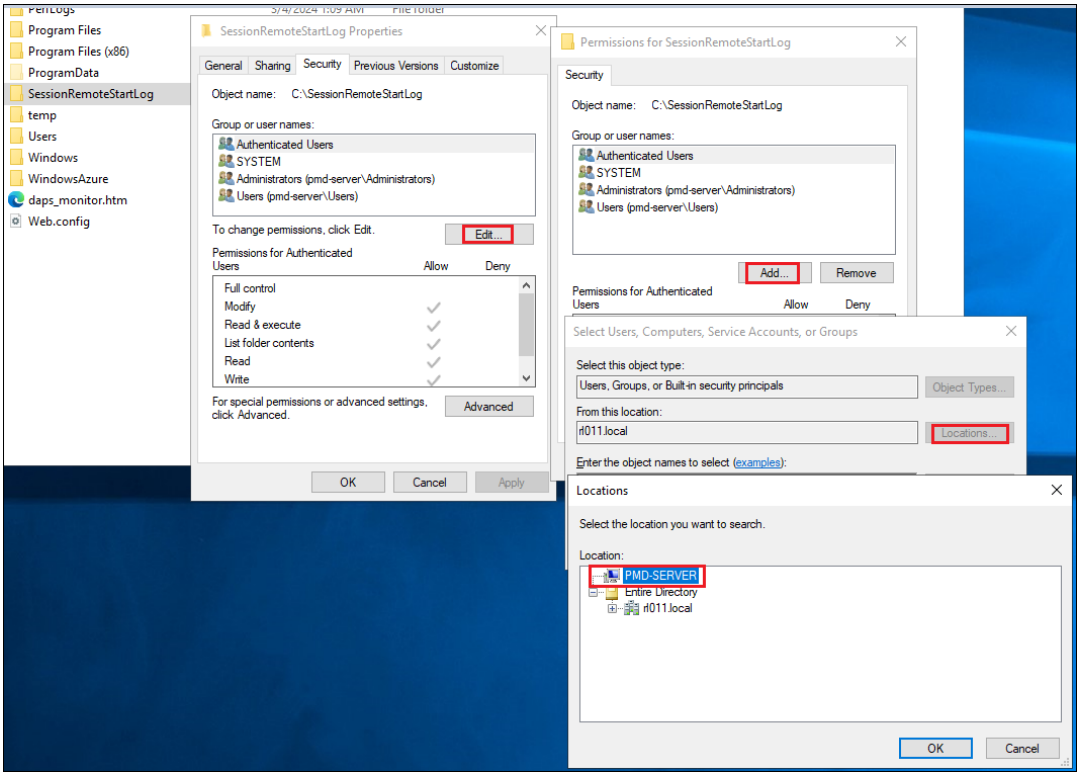
b) Right-click **Anonymous Authentication** and click **Enable**.

c) Again, right click on select **Anonymous Authentication >** choose **Edit >** select the identity consistent with Configuration for Accessing Other API Services.

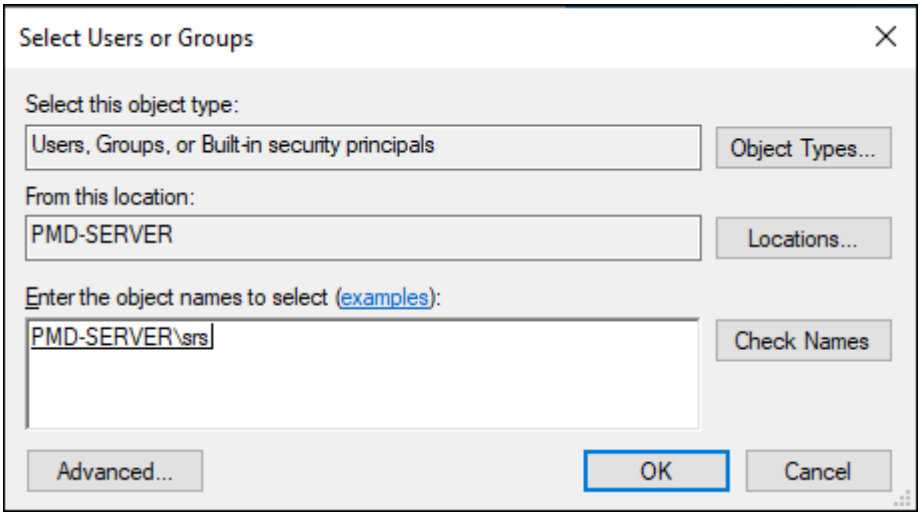


2. Now you can create a folder at your preferred location, and grant permissions to log to the new folder.

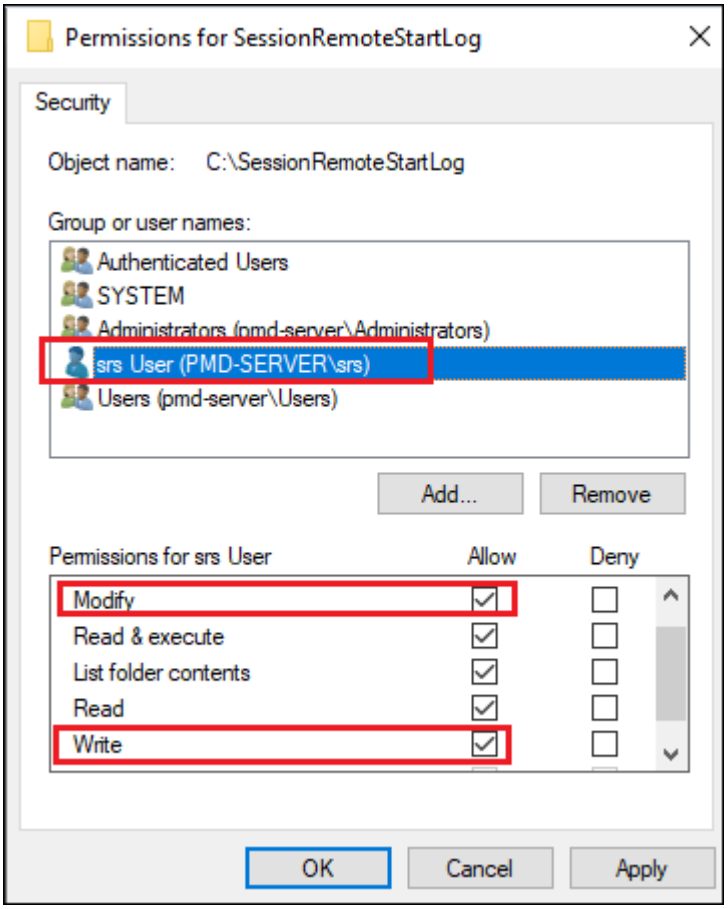
a) For example, after the folder creation under `C:\SessionRemoteStartLog`, right-click the folder and select **Properties**. In the **Security** tab, click **Edit** under **Group or user names** and then select **Add** to change the location to a local computer.



b) Input the Session Remote Start user created in the previous section. (If the default identity is used, input `IIS AppPool\SrsAppPool` instead).



c) Grant **Modify** and **Write** permissions. (If the default identity is used, grant access to **SrsAppPool**)

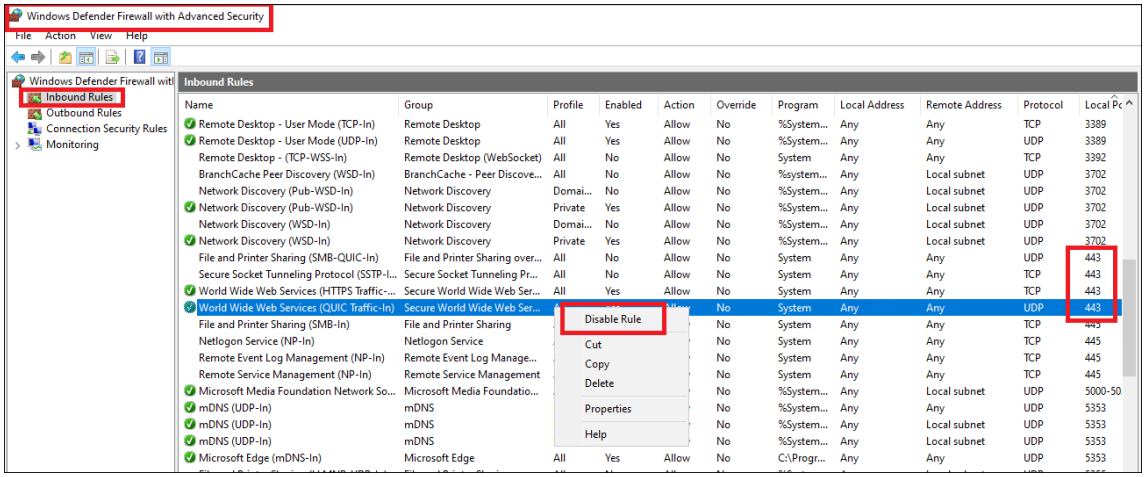


7. Configure Inbound Firewall Rules

Specifying IP addresses and host names of the trusted services and StoreFront ensures that only these sources can communicate with Session Remote Start and helps to prevent DoS or other opportunistic attacks against the Session Remote Start server.

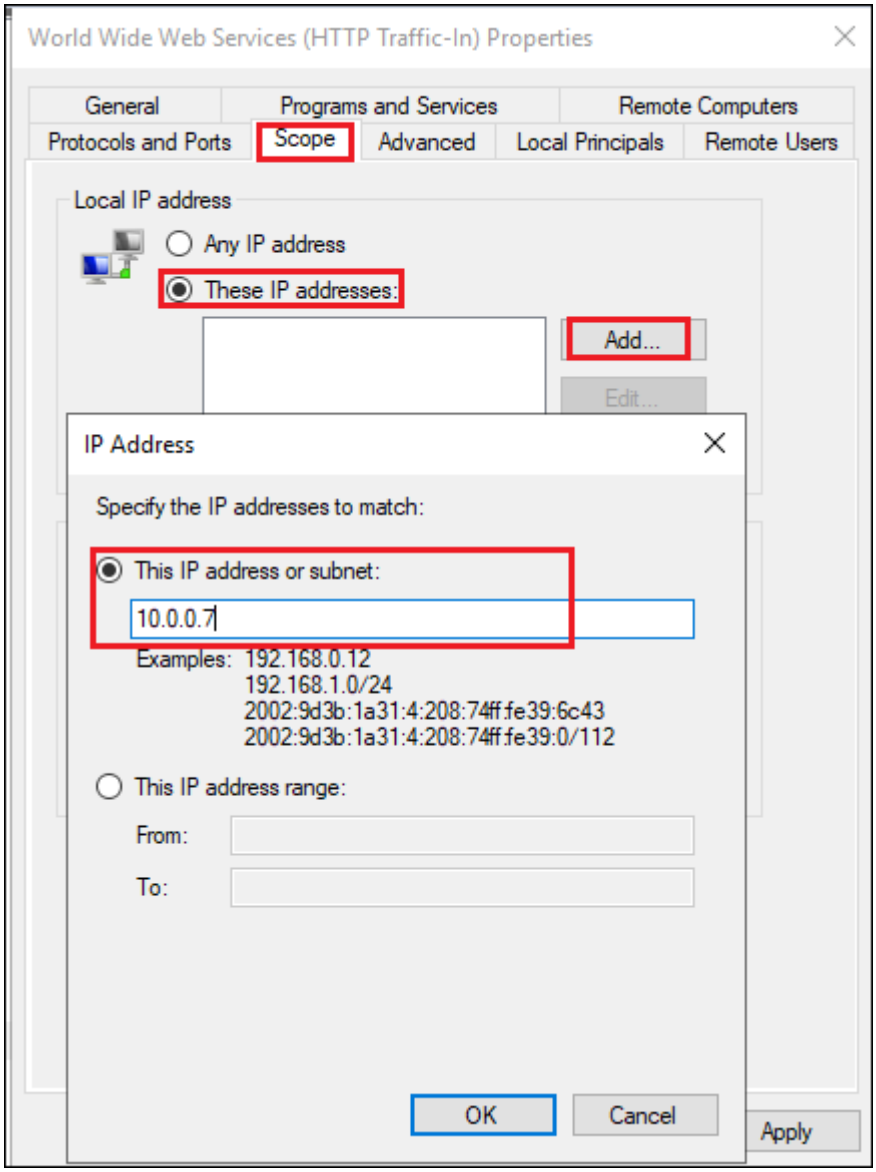
After creating the https binding on port 443, customers can configure inbound firewall rules by **Windows Defender Firewall with Advanced Security** UI to allow inbound TCP traffic.

1. Disable all 443 inbound rules except **World Wide Web Service (HTTPS Traffic-In)**.

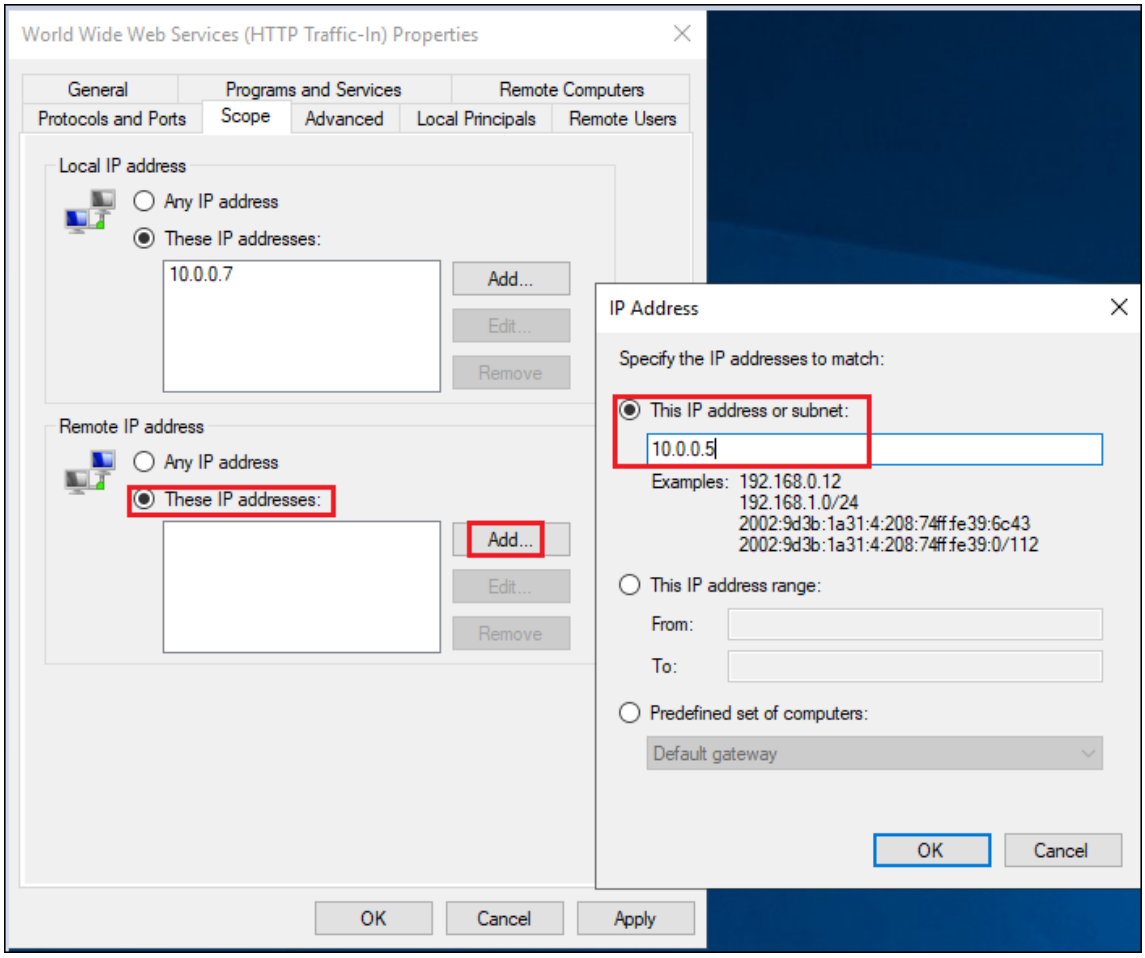


> **Note:**
>
> Remember to check if there are any enabled rules allowing all (all the limitations set to any) inbound traffic.

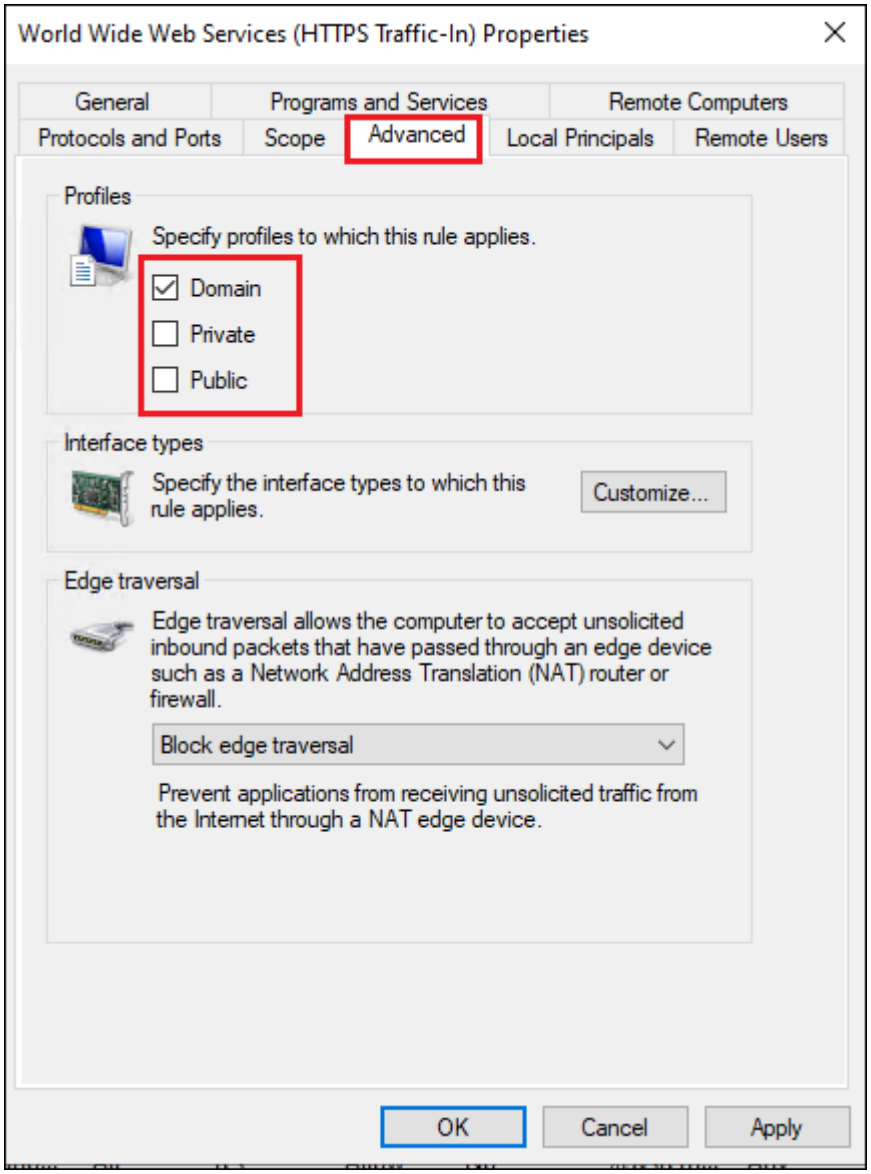
2. Double-click **World Wide Web Service (HTTPS Traffic-In)**, and open the **Properties** configuration. Switch to the **Scope** tab.
3. Add local IP address limitation. This is the local endpoint(s) (Network Interface) for 3rd-party Auth Service and StoreFront.



4. Add remote IP address limitation. Add IP addresses of 3rd-party Auth Service and StoreFront.



5. Switch to the **Advanced** tab and apply to related profile(s).



Third party firewall products will require configuring separately.

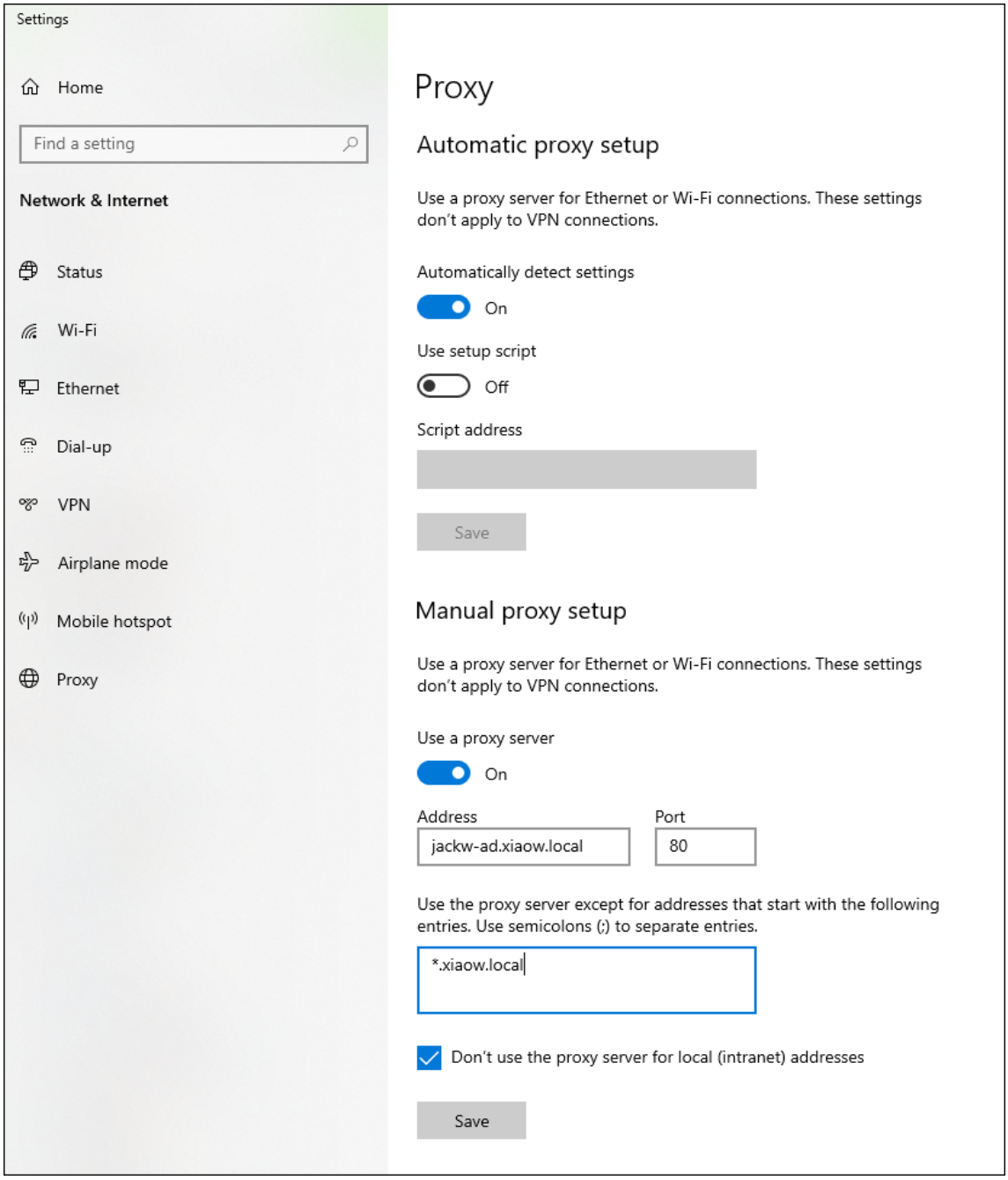
8. App Protection

If App Protection is enabled for a delivery group, the customization described in this Citrix documentation must be applied to the Session Remote Start store:

9. HTTP Proxy Configuration

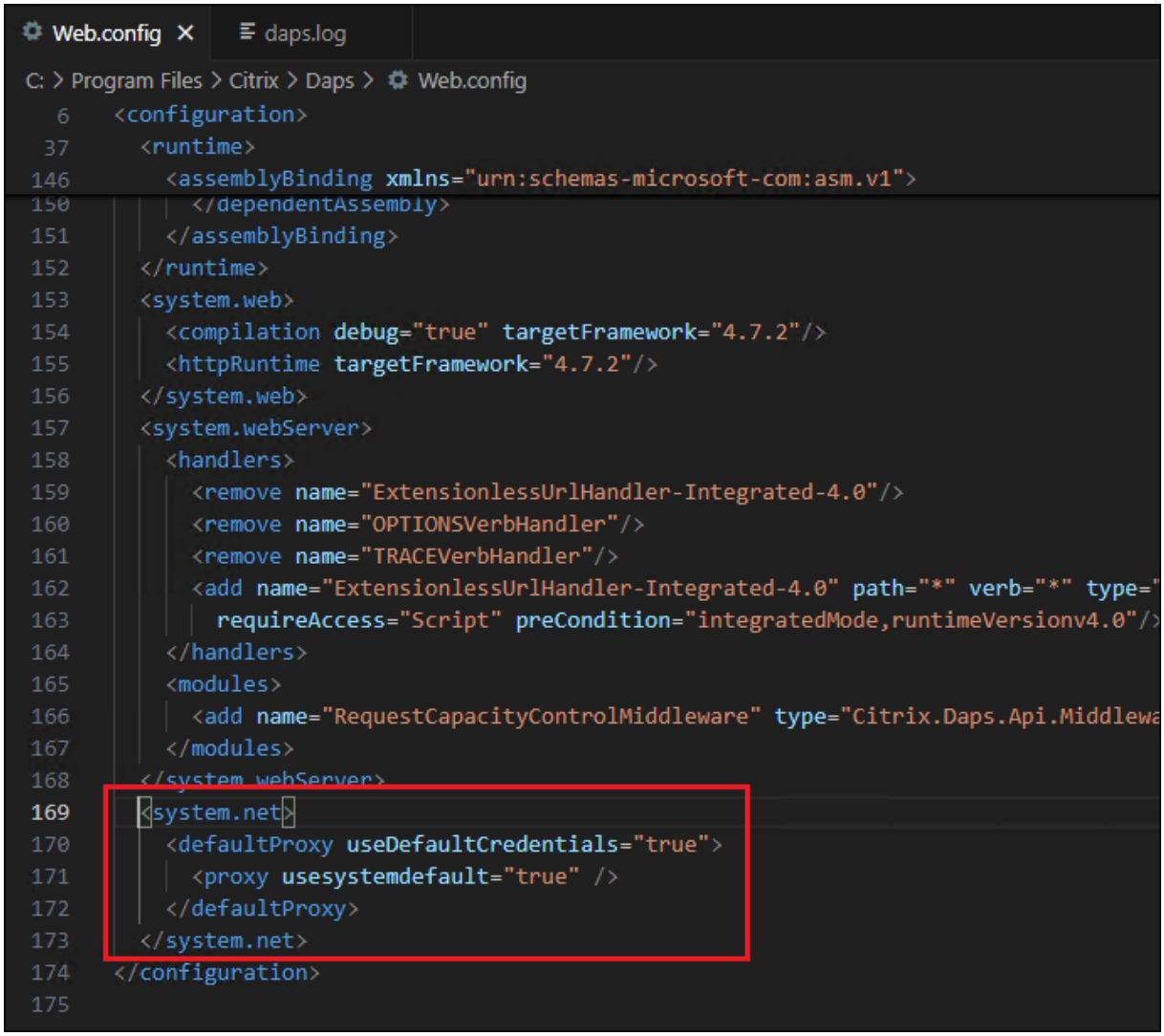
Session Remote Start supports only unauthenticated HTTP proxies.

1. Configure the WinINet HTTP proxy. e.g.



2. Append the following code to `web.config`.

```
<system.net>
  <defaultProxy useDefaultCredentials="true">
    <proxy usesystemdefault="true" />
  </defaultProxy>
</system.net>
```

10. mTLS configuration

Session Remote Start API does not require end-user authentication, unlike StoreFront. Therefore, it is essential to restrict access to trusted services only.

Enforcing Security with Mutual TLS (mTLS)

One way to ensure secure communication is by enforcing mutual TLS (mTLS) authentication between Session Remote Start and other trusted services.

Steps to Enable mTLS

1. Edit the `Web.config` file.

2. Locate the `mTLSEnabled` setting and set it to **`true`**.
3. Specify the `certificate thumbprint` to authenticate trusted services.

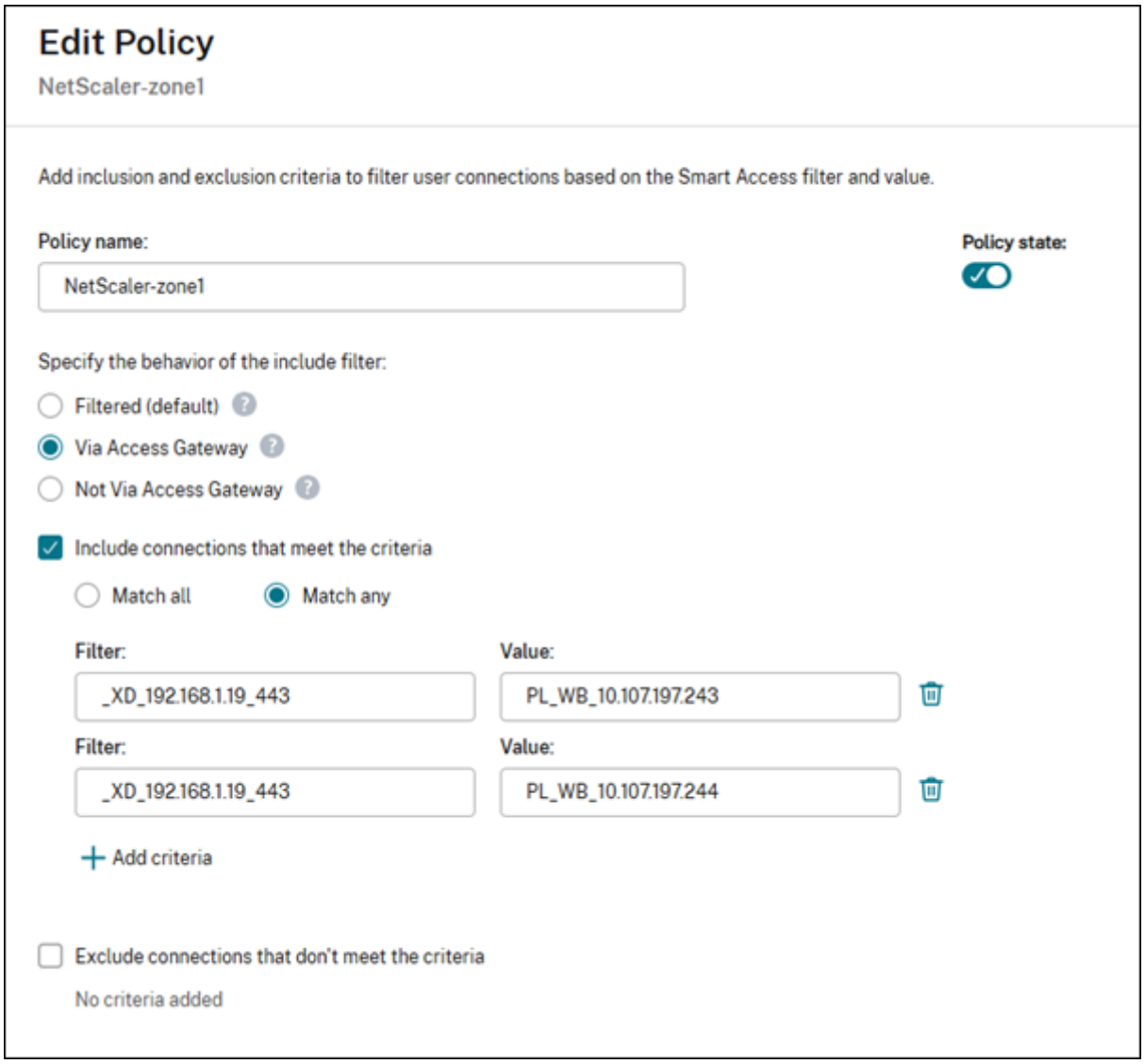
```
<add key="mTLSEnabled" value="false"/>
<add key="mTLSClientCertificateThumbprints" value="A1B2C3D4E5F67890123456789ABCDEF012345678"/>
```

This configuration ensures that only authorized services with a valid certificate can access Session Remote Start, enhancing security.

11. Filtering Enumerated Resource Results Using Smart Access

We support Access Gateway broker access rules to filter enumerated resources based on Smart Access policies.

For example, consider the below existing access policy.



Edit `Web.config` for Smart Access Filtering To enable Smart Access filtering for enumerated resource results, modify the Web.config file by adding or updating the following settings:

```
    <add key="mTLSEnabled" value="false"/>
    <add key="SmartAccessFarmName" value="_XD_192.168.1.19_443"/>
    <add key="SmartAccessConditions" value="PL_WB_10.107.197.243,PL_WB_10.107.197.243"/>
</appSettings>
```

`SmartAccessFarmName` - Specifies the Smart Access farm name (example: `_XD_192.168.1.19_443`).
`SmartAccessConditions` - Defines the conditions for filtering enumerated resources based on Smart Access rules.

12. Multi-Monitor Resolution Management

When customers use multiple monitors with different resolutions or scaling settings, they may experience display inconsistencies. The Launch User Resource API allows them to configure screen resolution settings for a seamless experience.

During session preparation, SRS simulates the user's screen layout based on the provided resolution settings, ensuring consistency across all monitors.

To ensure accurate simulation, customers must manually collect monitor layout and resolution details. Citrix provides the `collect_screen_resolution_example.ps1` script to simplify this process.

Step 1: Collect Screen Layout

Run the following command on the Virtual Desktop Agent (VDA) to retrieve the screen layout:

```
.\collect_screen_resolution_example.ps1 -type ScreenLayout
```

This will output the current screen layout configuration.

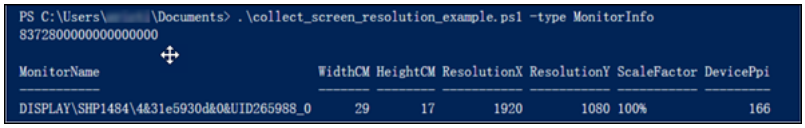
```
PS C:\Users\        .\collect_screen_resolution_example.ps1 -type ScreenLayout
<Screens>
    <Screen
        LeftPosition="0" TopPosition="0" Width="960" Height="1080"
    />
    <Screen
        LeftPosition="960" TopPosition="0" Width="960" Height="1080"
    />
</Screens>
```

Step 2: Collect Monitor Resolution and Scaling Information

If the client devices have different screen resolutions, run the script on the Citrix Workspace™ App (CWA) endpoint instead:

```
.\collect_screen_resolution_example.ps1 -type MonitorInfo
```


This will collect the details such as Device PPI and Scale Factor for the client device.



Then format this to former Screen XML attribute as:

```
<Screens>
    <Screen
        LeftPosition="0" TopPosition="0" Width="960" Height="1080"
            ScaleFactor=”100” DevicePpi=”166”
    />
    <Screen
        LeftPosition="960" TopPosition="0" Width="960" Height="1080"
            ScaleFactor=”100” DevicePpi=”166”
    />
</Screens>
```

And make this as the parameter for the **Launch User Resource API**.

13. Secure HDX Support

SRS supports secure HDX support without requiring additional configuration.