Citrix Always On Tracing 2507 LTSR CU1

Contents

Introduction

July 25, 2026

Always On Tracing (AOT) is a modern diagnostic framework from Citrix designed to continuously capture relevant log data across the Citrix Virtual Apps and Desktops (CVAD) environment. AOT is a subset of Citrix Diagnostic Facility (CDF) traces, with additional logs from other components, that can be helpful when troubleshooting common failures (for example, VDA registrations and application/desktop launches etc..).

Unlike traditional tools that require manual intervention, AOT runs in the background by default, ensuring that critical information is available when issues occur, even if the problem cannot be reproduced.

AOT is built to simplify troubleshooting for IT administrators and Citrix support teams to improve the mean-time-to-resolution by:

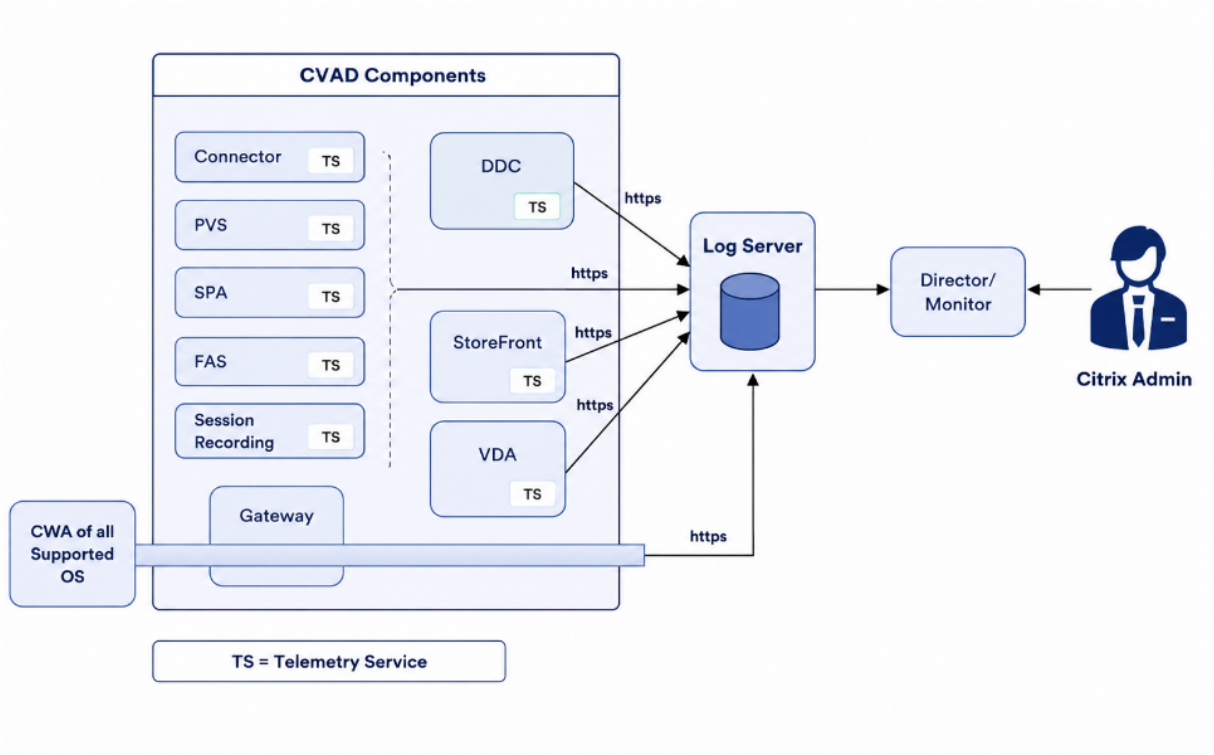
- Automatically capturing log data for user sessions and system events.
- Eliminating the need to reproduce issues.
- Making logs easily accessible and readable without engineering-level expertise.
- Enabling self-service log data to reduce dependency on Citrix Support.

Key Benefits

- Faster MTTR (Mean Time to Resolution)
- Always-on, no manual trace initiation needed
- Self-diagnosis for common issues
- Streamlined escalation to Citrix Support with full context
- Secure, compliant log handling with sensitive data redaction

Core Concepts

July 25, 2026



The Always on Tracing (AOT) architecture is designed to provide continuous, real-time log collection across the Citrix Virtual Apps and Desktops (CVAD) and DaaS environments. It enables IT administrators and Citrix Support to troubleshoot user and infrastructure issues without requiring manual log initiation or issue reproduction.

The AOT system is built on several integrated components working in sequence to capture, transfer, and store logs from multiple Citrix infrastructure layers into a centralized repository.

Key Components and Workflow

1. **AOT Log Generators (Citrix Core Components)**: Citrix components such as the Virtual Delivery Agent (VDA), Delivery Controller (DDC), StoreFront, and others act as log generators. These components are equipped with the Telemetry Service or a similar service, which uses the AOT API to capture trace data, record pre-defined step traces, and log error events relevant to user sessions and transactions.
2. **Enable Log Collection**: When an administrator initiates a request to collect logs (typically through the web studio settings, instructions are saved in the site database and sent to the broker, which orchestrates the log collection process across the relevant components, including broker, storefront and VDA. The administrator initiates the request to collect logs through a PowerShell command from other components separately like Session Recording, PVS, FAS etc…

3. **Telemetry Service**: After configuring the Log Server address in the Citrix Web Studio, and when the settings are propagated, the Telemetry Service on each component then activates the new real-time AOT listener, which collects logs based on predefined events, failures, or trigger points.
4. **Log Transfer to Centralized Log Server**: Once the logs are collected, the Telemetry Service or an independent service transfers them securely to a centralized log server directly. In environments where endpoints are connected from external networks through a Citrix Gateway, the transfer typically occurs over a SOCKS tunnel to ensure secure and seamless transfer.
5. **Log Storage and Organization**: The Centralized Log Server receives, parses, and stores the AOT logs in a structured and searchable format using an indexed database backend (OpenSearch by default). Logs are tagged by session, component, and timestamp, enabling easy access and efficient troubleshooting.

Logs in the Citrix AOT system are stored in a structured and searchable format, tagged with various fields to enable easy access and efficient troubleshooting. These tags include:

- **MachineName**: The name of the machine where the log originated.
- **MachineIP**: The IP address of the machine.
- **Role**: The role of the Citrix component (e.g., VDA, DDC, StoreFront).
- **TimeStamp**: The UTC timestamp when the log event occurred.
- **Message**: The actual log message content. Could be searched/filtered by words.
- **Level**: The severity level of the log (e.g., info, warning, error).
- **Module**: The specific software module that generated the log.
- **ProcessName**: The name of the process that generated the log.
- **ProcessId**: The ID of the process.
- **Thread**: The thread ID within the process.
- **Cpu**: Information related to CPU id at the time of the log.
- **SessionId**: The ID of the user session associated with the log.
- **Class**: The class or component within the module.

These detailed tags allow administrators to quickly filter, search, and analyze logs based on specific criteria, facilitating the identification and resolution of issues.

In summary, the AOT workflow begins with Citrix components generating logs. A centralized request is initiated to collect these logs, which the Telemetry Service gathers from the relevant components. The collected logs are then securely transferred to a centralized log server for indexing, storage, and later analysis.

How AOT Works

Citrix Always on Tracing (AOT) continuously captures log data from key components in your environment. When triggered, logs are collected automatically, securely transferred to a centralized log

server, and retained for analysis. This eliminates the need for manual tracing and simplifies troubleshooting.

Tracing Triggers and Log Collection: Citrix components such as VDA, DDC, StoreFront, and other components are equipped with the AOT API to track critical steps and errors. These components serve as AOT log generators. When an administrator initiates a log collection request from the Delivery Controller:

- The DDC sends instructions to relevant Citrix components. Each component forwards the request to its local Telemetry Service.
- The Telemetry Service starts the real-time AOT listening program to collect AOT logs and forwards the logs to the Centralized Log Server.

Log Storage and Retention: Once the AOT logs are received, the Centralized Log Server formats and indexes them into a structured, searchable database. Logs are retained based on a defined retention policy to optimize storage usage. By default, AOT logs are retained for 7 days, after which they are automatically purged to conserve space.

AOT vs Traditional Logging

Aspect	AOT	Traditional CDF Tracing
Activation	Automatic, continuous	Manual, issue reproduction required
Complexity	User-friendly, readable logs	Engineering-level parsing
Diagnostic Speed	Immediate logs available	Delayed by reproduction effort
Resource Usage	Low overhead with optimized buffers	Moderate to high if used incorrectly

Traditional tracing tools require manual effort and only work if you catch the problem while it's happening. With AOT, logging runs continuously in the background, so issues are captured as they happen —even if you're not watching. This saves time and makes troubleshooting much easier.

AOT Highlights

- **Always On**: Tracing runs all the time, no need to turn it on manually.
- **Auto Capture**: Logs are collected always when issues occur.
- **Central Log Server**: All logs are stored in one place for easy access.
- **Easier Troubleshooting**: Helps quickly pinpoint common failures.
- **No Reproduction Needed**: No need to recreate the issue; it's already captured.

Minimum supported versions for Always on Tracing (AOT)

July 25, 2026

This table outlines the minimum supported versions of Citrix components required to enable Always On Tracing (AOT) log collection. Version requirements may vary depending on how users access their resources either through **direct access** (for example, directly to StoreFront) or through **Gateway-based access** (Citrix Gateway or Cloud Gateway Services).

Component / Platform	Type	Minimum Version (Direct Access)	Minimum Version (Access via Gateway)
Citrix Workspace App Windows	Client	2511.10 and 2507 CU1 or later	2603 and 2507 CU2 or later
Citrix Workspace App Mac	Client	2603 or later	2603 or later
Citrix Workspace App Linux	Client	Coming soon	Coming soon
Citrix Workspace App HTML5	Client	2511 or later	2603 or later
Citrix Workspace App ChromeOS	Client	2511 or later	2603 or later
Citrix Workspace App Android	Client	2603 or later	2603 or later
Citrix Workspace App iOS	Client	2603 or later	2603 or later
Citrix Virtual Apps and Desktops (Windows VDA, DDC, PVS, FAS, Session Recording, STF, Linux VDA, Mac VDA, WEM, UPM)	Platform	2402 CU4, 2507 CU1, 2511 or later	2402 CU4, 2507 CU1, 2511 or later
Citrix DaaS *	Service	Post February 2026	Post February 2026
Citrix NetScaler	Infrastructure	14.1.60.57	14.1.60.57
Citrix License Server		540000 or later	54000 or later

*For Citrix DaaS deployments, AOT log collection follows the same process. You must configure the DaaS settings to forward logs to the AOT Log Server. All components deployed within the resource location can forward logs to the Log Server once configured. Citrix DaaS control plane components

do not send logs to the customer-managed Log Server; instead, they continue to forward logs to Citrix-managed cloud telemetry (Splunk). This behaviour applies unless those cloud control plane components are replaced or supplemented with customer-managed CVAD components, such as using StoreFront instead of Workspace Platform or Citrix Gateway instead of Cloud Gateway Services, where applicable.
For more details on log availability across different deployment models, see the **AOT Logs Availability Across Deployment Scenarios** section.

> **Note:**
>
> **Direct Access** refers to scenarios without Gateway involvement (for example, direct StoreFront access).
>
> **Access via Gateway** includes Citrix Gateway and Cloud Gateway Services (CGS).
>
> "Coming soon"indicates planned support in upcoming releases.

Centralized AOT Log Server

July 25, 2026

The AOT Log Server acts as a centralized destination for collecting and storing Always on Tracing (AOT) logs from various Citrix components. It is containerized using Docker technology and is preferably deployed on Linux, though it also supports Windows environments via Docker Desktop, and on the Citrix Connector appliance.

Key Features of the AOT Log Server

- **Centralized Log Storage**: Aggregates logs from multiple Citrix components into a unified and searchable location.
- **Retention Policy**: Logs are retained based on a configurable policy (default: 7 days) to optimize storage usage and performance.
- **Deployment**: The Log Server is deployed using packaged installer scripts and Docker containers. The setup requires compatible hardware and a Docker runtime environment or the Cloud Connector appliance (if selected).
- **Secure Log Access**: Logs can be retrieved using CLI tools included with the Log Server. All access requires proper authentication to ensure security and compliance.
- **Integration with Citrix Core Components**: The AOT Telemetry Service is embedded within core Citrix components, enabling them to act as log generators. Telemetry Service ensures

seamless and automated log capture across these components, enabling consistent logs visibility throughout the Citrix infrastructure.

AOT Log Server Deployment and Configuration

AOT Prerequisites

The AOT Log Server relies on Docker container technology to function. While Linux is the recommended host operating system, installation on Windows is also possible with Docker Desktop. Alternatively, using a Citrix Connector Appliance provides a packaged solution removing the need to manage hosts and Docker infrastructure.

A single Log Server can support **up to 128,000 active component connections** at any moment, and it can process **up to 10,000 log events per second**. Any sizing or deployment recommendation should be based on these two factors:

- **Connection capacity** determines how many components can be connected simultaneously.
- **Log throughput** determines how many logs the system can ingest and index without performance impact.

A single Log Server is suitable as long as your environment stays within these limits.

Citrix Connector Appliance

- Ensure you are running the latest version of Citrix Connector Appliance.
- If you don't have it, download the latest version from Citrix Cloud and install it.

Linux

- **Supported OS**: Ubuntu 24.04 and RHEL 8.10, 9.4
- **Hardware requirements**: 4 cores of CPU, 16 GB RAM, and 500 GB free disk space.
- **Software Requirements**: Install Docker Engine for Linux or Docker Desktop for Linux.
- **User permissions**: Ensure the user can start Docker without requiring root.

Windows

- **Supported OS**: Windows 11, Windows 10, Windows Server 2022 and Windows Server 2025. For Azure VMs see recommendations here.
- **Hardware requirements**: 8 cores of CPU, 24 GB RAM, and 500 GB free disk space.
- **Software requirements**: Install Docker Desktop (subscription may be required) for Windows.
    - **WSL version 2.1.5** or later is required.

- Enable **Hyper-V** and **Containers** Windows features.
- Set the memory limit >= 12 GB in docker desktop settings

AOT Logs availability across deployment scenarios

July 25, 2026

This table shows how AOT (Always On Tracing) log collection happens across different deployment scenarios. It highlights which components can send logs to the AOT Log Server based on the architecture used, including **on-prem, Gateway-based, cloud, and hybrid deployments**.

Use Case	Deployment Scenario	CWA	StoreFront (On-Prem)	Citrix Workspace (Cloud)	Gateway (On-Prem)	Citrix Gateway service	DDC (On-Prem)	Cloud DDC	Cloud Connector	VDA	FAS	PVS	Other Components
1	On-Prem Direct Access (StoreFront + on-prem DDC)	Yes	Yes	N/A	N/A	N/A	Yes	N/A	N/A	Yes	Yes	Yes	Yes

Use Case	Deployment Scenario	CWA	StoreFront (On-Prem)	Citrix Workspace (Cloud)	Gateway (On-Prem)	Citrix Gateway service	DDC (On-Prem)	Cloud DDC	Cloud Connector	VDA	FAS	PVS	Other Components
2	On-Prem via Gateway (StoreFront + on-prem DDC + Gateway)	Yes	Yes	N/A	Yes	N/A	Yes	N/A	N/A	Yes	Yes	Yes	Yes
3	Fully Cloud (Citrix Workspace + Cloud DDC + Citrix Gateway service)	Coming Soon	N/A	Yes*	N/A	Yes*	N/A	Yes*	Yes	Yes	Yes	Yes	Yes

Use Case	Deployment Scenario	CWA	StoreFront (On-Prem)	Citrix Workspace (Cloud)	Gateway (On-Prem)	Citrix Gateway service	DDC (On-Prem)	Cloud DDC	Cloud Connector	VDA	FAS	PVS	Other Components
4	Hybrid Cloud (Citrix Workspace + Cloud DDC + Gateway)	Coming Soon	N/A	Yes*	Yes	N/A	N/A	Yes*	Yes	Yes	Yes	Yes	Yes
5	Hybrid Cloud (StoreFront + Cloud DDC + Gateway)	Coming Soon	Yes (Manual Configuration Required)	N/A	Yes	N/A	N/A	Yes*	Yes	Yes	Yes	Yes	Yes

Note:

- **Yes** → Component supports uploading logs to the AOT Log Server
- **Yes*** → Logs are available, but they are handled and stored through Citrix-managed service instead of the customer-managed AOT Log Server
- **Manual Configuration Required** → Manual setup is required for log upload to the Log Server
- **Coming Soon (CWA)** → Indicates current gap in direct log upload from Citrix Workspace app (work in progress)

Installing and Configuring Log Server

July 25, 2026

The Log Server can be set up on individual Linux or Windows servers or hosted on the Citrix connector appliance and leverage evergreen updates. Refer to the relevant section for installation and configuration steps for each option.

> **Note for Installation**
>
> - For enhanced security, HTTPS is recommended in the deployment.
> - Ensure that the selected port is not already in use.
> - Avoid using privileged ports (0–1023) as they require administrator or system-level permissions.
> - Verify that firewall rules allow traffic on the chosen port.
> - Use a port number within the valid range (0–65535), but avoid ports commonly used by system services to prevent conflicts.
> - If you are using the Citrix Connector Appliance to deploy the Log Server, ensure that only port 443 is configured.
> - The port numbers used in the examples (8080 for HTTP and 8443 for HTTPS) are for reference only. You are not required to use these exact ports. Select appropriate port numbers based on your environment and follow the installation guidelines outlined above when configuring the AOT Log Server.

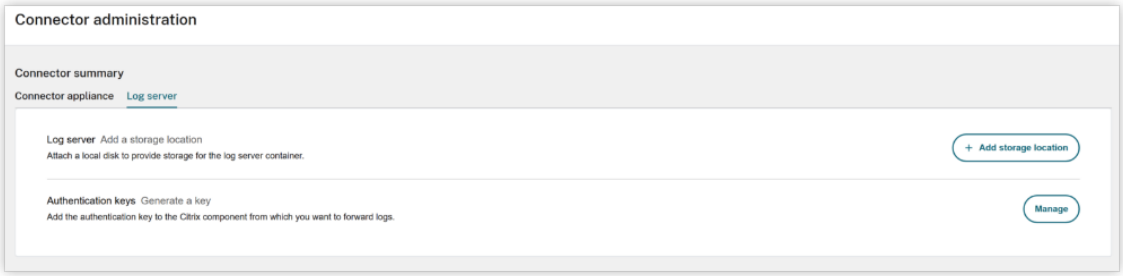
Installing Log Server via Citrix Connector Appliance

The Log Server can be deployed within the Citrix Connector Appliance. This approach eliminates the need to deploy and manage a host VM and manually download images or run container commands. The Log Server is automatically onboarded during Connector Appliance upgrade and remains up to date through ongoing connector updates, ensuring you always have the latest version. For more information, see Connector appliance updates

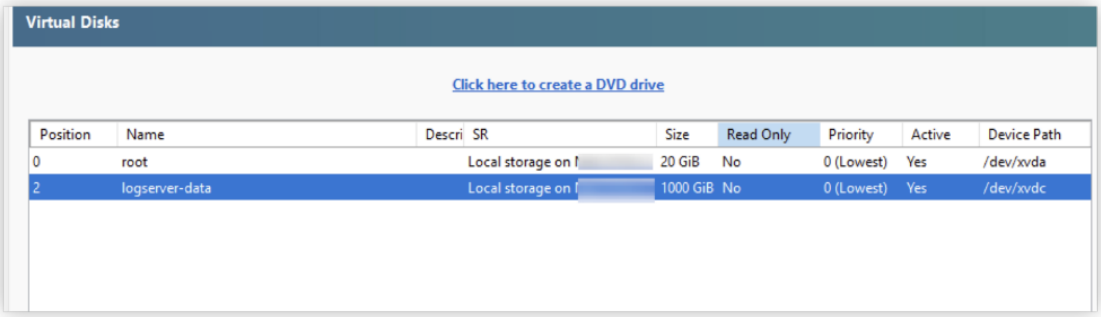
Steps to Deploy Log Server via Citrix Connector Appliance

1. If your environment does not already have a Connector Appliance, deploy the appliance to your hypervisor or from your public cloud marketplace. The minimum required Citrix Connector Appliance is 11.4.1.444. After importing, register the appliance with Citrix Cloud. For more information, see Obtain the connector appliance

2. By default, Connector Appliance has 2 vCPUs and 4GB memory; increase resources to at least 4 vCPUs and 16GB memory to handle log server requests.
3. Create a DNS "A"record for the Connector Appliance FQDN so that supported components can resolve the Log Server hostname. Joining the Connector Appliance to an Active Directory domain is optional.
4. Cloud Monitor retrieves AOT logs through the **Monitor Connector service running on a supported Windows Cloud Connector** in the Resource Location. If the Log Server is deployed on the Citrix Connector Appliance, a supported Windows Cloud Connector must also be present in the same Resource Location. The Connector Appliance alone is not sufficient for Cloud Monitor to retrieve AOT traces. The Cloud Connector must be running version **6.141.0.13739** (or **4.420.0.13739**) or later. Earlier versions will cause the `GetAotTraces` API call to fail, resulting in an HTTP 500 error in Monitor.
5. The Connector Appliance provides a self-signed certificate that is served to a browser that connects to the Connector Appliance administration page. To be able to connect to logserver via HTTPs, you must replace this self-signed certificate with one of your own that is signed by your organization or generated by using your organization's chain of trust, for more details refer to Managing Certificates or Replace Server Certificate.
6. Once the upgrade is complete, log in to the Connector Appliance UI at `https://<connector-appliance-FQDN>/?enable=logserver`, ensure to have the "?enable=logserver"to be able to see the logserver UI. The new Log Server tab now shows storage and authentication key management options.

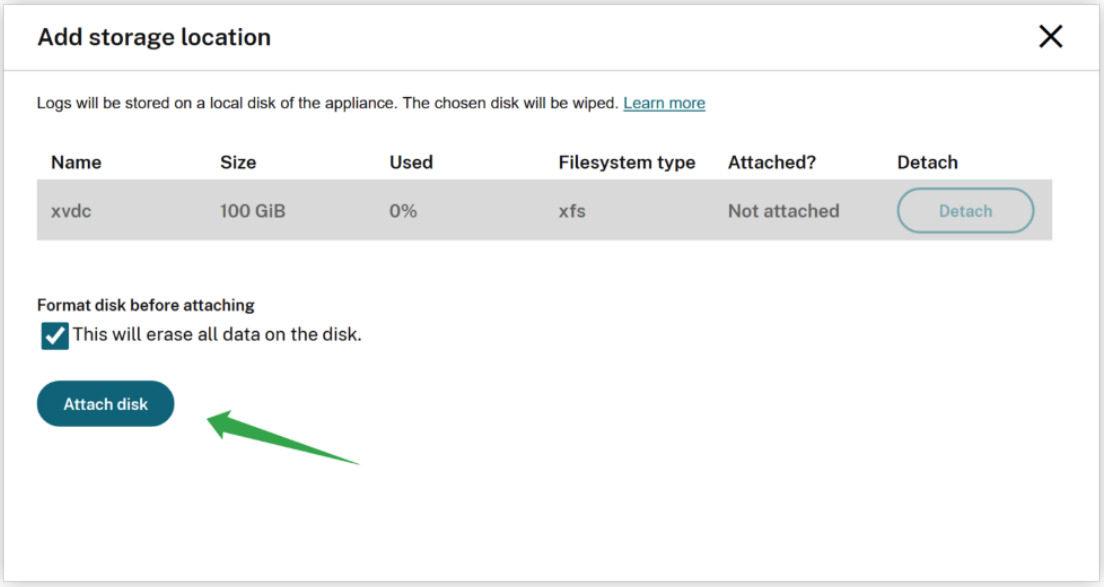


7. The connector appliance boot disk has a capacity of 20GB. To support efficient log storage on the logserver, you should add another virtual disk to the connector appliance using your hypervisor or cloud management platform. This additional disk should have enough space to meet your log storage needs (as explained in the previous section).The XenServer screenshot below shows an example of a connector appliance with an extra disk configured.

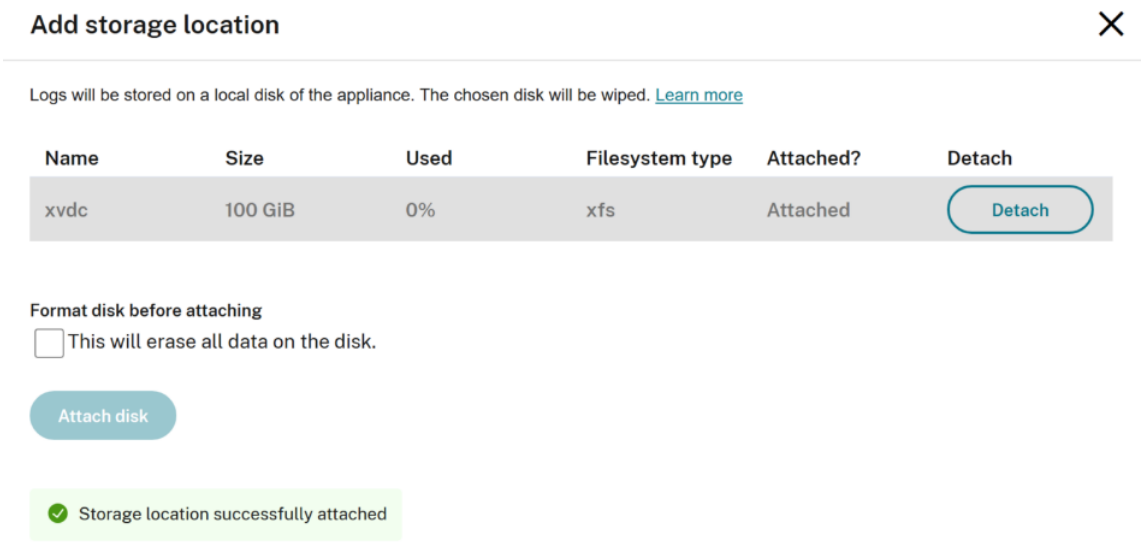


> **Note:**
>
> **A known requirement for VMware ESXi:** Users must attach the additional data disk to a different SCSI controller than the one used by the root disk.

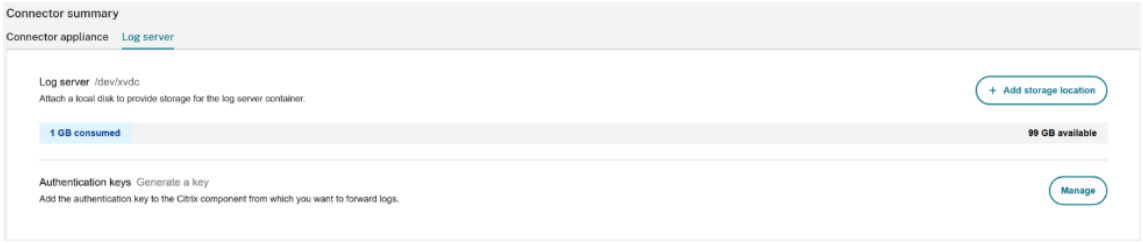
8. After adding the key disk, the logserver UI will automatically detect it, allowing you to format and mount it to the logserver container on the connector appliance.



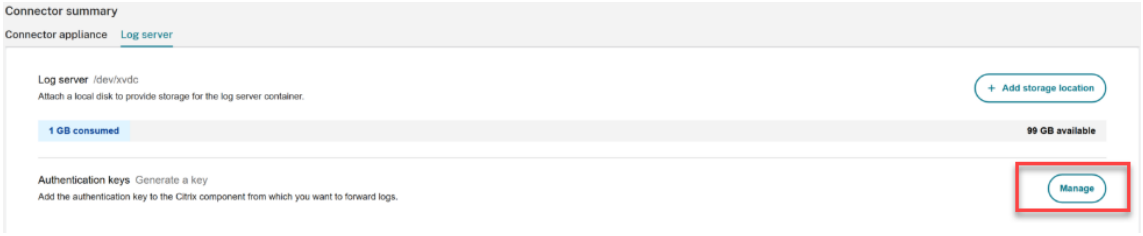
9. Once you click the “Attach disk”button, the disk will be mounted to the logserver container.

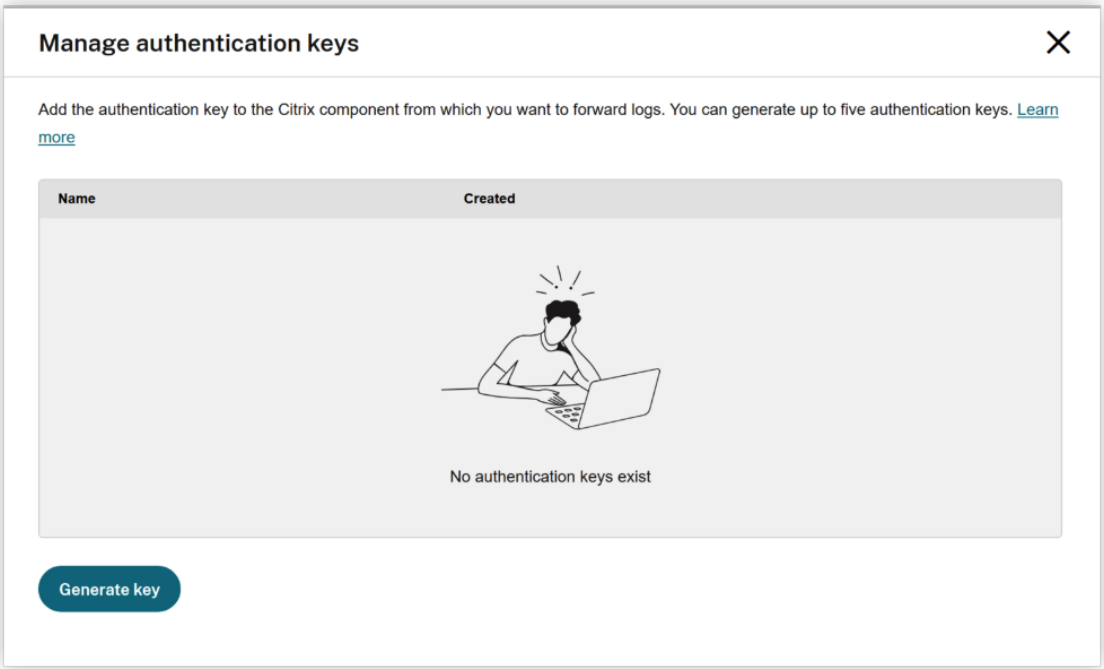


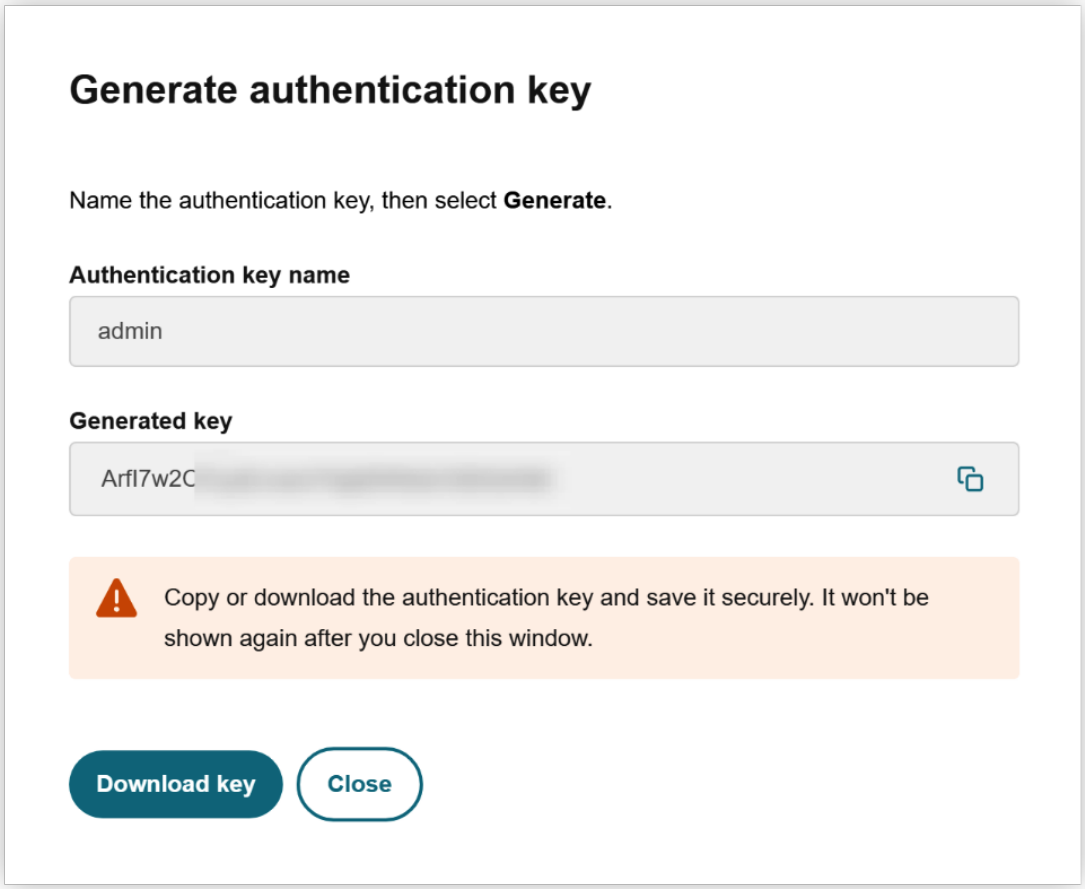
10. The main page will display the disk size along with information about how much space is used and how much remains.



11. After the disk is attached, verify that the Log Server is running by calling the ping endpoint: `https://<connector-appliance-FQDN>/ctxlogserver/Ping`. A pong response confirms that the Log Server has successfully started.
12. Click on Manage button, and click "Generate Key". Now, enter the role name, then copy or download the auth key. The key won't be shown again after you close the window.







Linux installation

1. Download the log server docker container image from Citrix downloads.
2. Place the downloaded files in the same directory.
3. Run the installer in the directory with terminal (Linux) or command prompt (Windows) and follow the instructions:

```
chmod +x ./InstallLogServer

#Install with https mode with port 8443 with default path
./InstallLogServer --https --cert </path/your_private_cert_key.pfx> --port 8443

#Install with http mode, with port 8080 with default path
./InstallLogServer --port 8080

#Command to change the config path and data path of your choice with https mode
./InstallLogServer --https --cert </path/your_private_cert_key.pfx> --port 8443 --config /Path/LogServer/Config --database /Path/LogServer/Data

#Command to change the config path and data path of your choice with http mode
./InstallLogServer --port 8080 --config /Path/LogServer/Config --database /Path/LogServer/Data

#To support CWA client uploading AOT logs, some additional parameters need to be added after the install log server command.
--sta-server http://STA_SERVER_FQDN:port --log-server LOG_SERVER_FQDN:PORT
```

Where,

- STA_SERVER_FQDN is the hostname or ip address of the STA server (In on-prem installation, the STA server usually installs along with DDC.)
- LOG_SERVER_FQDN and PORT are the hostname of the log server itself, and the port specified (8080, 8443 or the –port value in the install parameter).
- The log server certificate your_private_cert_key.pfx must be trusted by other Citrix components, where the AOT client will use TLS connection to upload logs.

The **STA_SERVER** address allows the Log Server to issue reconnection STA tickets to Citrix Workspace app (CWA) clients when StoreFront cannot provide them—such as after a session timeout. This enables CWA clients to re-establish the connection directly with the Log Server if a connection break occurs.

The **LOG_SERVER** address is used when the Log Server requests an STA ticket from the configured STA server. The STA server returns a ticket that authorizes connections specifically to the Log Server

endpoint.

These parameters are optional if CWA clients don't need reconnection STA tickets or connect to LogServer directly without a gateway.

Post Installation on Linux

After the installation, some useful script files are generated:

```
# In Linux, sh scripts will be generated
DownloadLogsByTime.sh
DownloadLogsByWords.sh
GetAuthKey.sh
ListMachines.sh
StartLogServer.sh
```

Use `./StartLogServer.sh to start the server. Check your configpath/weblogs.txt` to confirm LogServer has started successfully.

If the LogServer starts successfully, we can see the below message in the weblogs file. Port 5000 is used by LogServer internally in docker containers with selected http or https protocol.

```
Now listening on https://[::]:5000
```

If the log server was installed in HTTP mode, successful logs should show the below:

```
Now listening on http://[::]:5000
```

If LogServer uses HTTPS, ensure its certificate is trusted on all machines uploading AOT logs

> **Note:**
>
> - The port configured in the installation step (8080 or 8443 or any port specified) should be used when configuring LogServer url in DDC, Storefront, VDA, etc.
> - Usually, it takes 30s to 60s to start on Linux.

Windows Installation

1. Download the log server docker container image from Citrix downloads.
2. Place the downloaded files in the same directory.
3. Run the installer in the directory with terminal (Linux) or command prompt (Windows) and follow the instructions:

Step 1

Install Docker Desktop (subscription may be required) for Windows on the log server VM. Follow the steps below to ensure Docker Desktop installs and starts correctly on Windows systems that rely on WSL 2.

1. Set the memory limit >= 12 GB in docker desktop settings
2. Docker Desktop requires the following Windows features, ensure to verify these features are enabled.
    - Hyper-V
    - Virtual Machine Platform
    - Windows Subsystem for Linux (WSL)
3. If any features are missing, install them and restart the VM to take effect.
4. After the system restarts, open **PowerShell** (Run as Administrator) and update WSL by executing the command `wsl --update`
5. Docker Desktop requires WSL 2. Configure it as the default by executing the command `wsl --set-default-version 2`
6. Once WSL is updated and the required Windows features are enabled, the Docker Desktop Engine should start successfully.

Step 2

Run the commands below to continue the installation.

> **Note:**
>
> The default location of Config and Database (Data) folders are created under C:\Users<username>\LogServer.You can change them with the commands below

```
#Install with https mode with port 8443 with default path
InstallLogServer.exe --https --cert <c:\path\cert.pfx> --port 8443

#Install with http mode, with port 8080 with default path
InstallLogServer.exe --port 8080

#Command to change the config path and data path of your choice with https mode
InstallLogServer.exe --https --cert <c:\path\cert.pfx> --port 8443 --config C:\LogServer\Config --database C:\LogServer\Datacmd

#Install with specific config path and data path
```



```
InstallLogServer.exe --port 8080 --config C:\LogServer\Config --
    database C:\LogServer\Datacmd

#To support CWA client uploading AOT logs, some additional parameters
    need to be added after the install log server command.
--sta-server http://STA_SERVER_FQDN:port --log-server LOG_SERVER_FQDN:
    PORT
```

Where,

- STA_SERVER_FQDN is the hostname or ip address of the STA server (In on-prem installation, the STA server usually installs along with DDC.)
- LOG_SERVER_FQDN and PORT are the hostname of the log server itself, and the port specified (8080, 8443 or the –port value in the install parameter).
- The log server certificate your_private_cert_key.pfx must be trust by other Citrix components, where the AOT client will use TLS connection to upload logs.

The STA_SERVER address enables LogServer to supply reconnection STA tickets to CWA clients when StoreFront cannot provide them due to session timeout. As a result, CWA clients can obtain reconnection tickets directly from LogServer in the event of connection failures.

The LOG_SERVER address is utilised when the logserver requests a STA ticket from the STA_SERVER. The STA_SERVER issues a STA ticket that exclusively authorises connections to the LOG_SERVER address.

These parameters are optional if CWA clients don't need reconnection STA tickets or connect to LogServer directly without a gateway.

Post Installation on Windows

After the installation, a few useful script files are generated in the same directory where you saved the installer files.

> **Note:**
>
> You can move these files to a different location. However, remember the new location, as they will be required again when configuring the Log Server.

```
#In Windows, bat scripts will be generated in the same directory where
    you saved the installer files.
DownloadLogsByTime.bat
DownloadLogsByWords.bat
GetAuthKey.bat
ListMachines.bat
StartLogServer.bat
```


Use `StartLogServer.bat` to start the log server.

Check your `configpath\weblogs.txt` to confirm LogServer started successfully

When the logs show the below, it means to say the Log Server has started successfully. If the LogServer starts successfully you will see the below message in the weblogs file. Port 5000 is used by LogServer internally in docker containers with selected http or https protocol.

```
Now listening on: https://[::]:5000
```

If the log server was installed in HTTP mode, successful logs should show the below:

```
Now listening on: http://[::]:5000
```

If LogServer uses HTTPS, ensure its certificate is trusted on all machines uploading AOT logs.

> **Note:**
>
> - The port configured in the installation step (8080 or 8443 or any port specified) should be used when configuring logserver URL in DDC, Storefront, VDA, etc.
> - Usually, it takes 1~10 minutes depending on the hardware in Windows.

Mutual TLS authentication (optional)

Mutual TLS (mTLS) provides an additional layer of security between the Log Server and clients (VDA, DDC, StoreFront, CWA). When mTLS is enabled, both the client and the server authenticate each other using certificates issued by your enterprise PKI.

mTLS is useful in environments where:

- Network segments are untrusted or shared
- There is a requirement to authenticate not only the Log Server but also each AOT log client
- Customers want to prevent unauthorized systems from sending log data
- Regulatory or compliance policies require certificate-based authentication.

Although mTLS is optional, it enhances security by ensuring that only trusted Citrix components can communicate with the Log Server, and that the log server can verify every incoming connection before accepting telemetry data.

Certificate requirements

To configure mTLS, the following certificates must be generated:

- **aotclient.pfx** –Certificate used by AOT log clients (VDA, DDC, StoreFront, CWA)
- **logserver.pfx** –Certificate used by the Log Server

- **enterprise-ca.cer** –The root or intermediate certificate used to sign both .pfx files

> **Note**
>
> - Skip this Mutual TLS section if you use Citrix Connector Appliance, as it isn't supported.
> - The enterprise-ca.cer file must be imported into the Trusted Root Certification Authorities store on both the Log Server and telemetry clients.
> - The aotclient.pfx and logserver.pfx certificates should not be protected by password.
> - The subject of aotclient.pfx must be CitrixAOTClient, allowing the telemetry client to automatically locate the certificate during runtime.

To enable mTLS, include the –ca parameter in the Log Server installation command.
This parameter specifies the path to the enterprise-ca.cer certificate.

```
# with default path
./InstallLogServer --https --cert logserver.pfx --ca enterprise-ca.cer
    --port 8443

# with customized path
./InstallLogServer --config /YourPath/LogServer/Config --database /
    YourPath/LogServer/Data --cert /YourPath/logserver.pfx --ca /
    YourPath/enterprise-ca.cer --port 8443

# delete temp certificate logserver.pfx in current install directory
sudo rm -rf /YourPath/logserver.pfx

# keep logserver.pfx accessed only by the container process user '
    ubuntu'.
sudo chmod 400 LogServer/Config/logserver.pfx
sudo chown ubuntu:ubuntu LogServer/Config/logserver.pfx
```

If mutual TLS authentication is required, run the following PowerShell command in DDC, Storefront, VDA and other CVAD components with administrator privileges.

```
# import client cert at the machine aot client
Import-PfxCertificate -CertStoreLocation Cert:\LocalMachine\My\ -
    FilePath c:\aotclient.pfx

# Verify successful import
Get-ChildItem Cert:\LocalMachine\My | Where-Object {
  $_.Subject -like "*AOTclient*"  }


# delete temp certificate aotclient.pfx
Remove-Item -Path "C:\aotclient.pfx" -Force
# Ensure LogServer's certificate is trusted on all machines uploading
    AOT logs.
```


> **Note:**
>
> The telemetry service is running in "network service"account, so it is required to manually grant NETWORK SERVICE full control over the private key of the CitrixAOTClient certificate, using the certlm.msc graphical interface.
>
> 1. Press Win + R, type certlm.msc and press Enter to open the Certificates (Local Computer) console.
> 2. Expand Certificates (Local Computer) > Personal > Certificates.
> 3. In the right pane, locate the certificate issued to CitrixAOTClient.
> 4. Open "Manage Private Keys"
> 5. Right-click the certificate and select All Tasks > Manage Private Keys
> 6. In the permissions dialog, select Add, type NETWORK SERVICE, and click Check Names (it should resolve to NT AUTHORITY\NETWORK SERVICE).
> 7. Click OK to apply the permissions.

If customer supplies self-signed certificates, do as follows:

- On the logserver side, logserver.pfx and aotclient.cer are installed as previously described. The aotclient.cer serves the role of enterprise-ca.cer.
- On the client side, aotclient.pfx and logserver.cer are imported as previously described. The logserver.cer serves the role of enterprise-ca.cer.
- For more information, seeCreate a new certificate

Verify the Log Server

Open your browser on Log Server or VDA or DDC, visit http://YourLogServerFQDN:8080/Ping

A response string "Pong UTC:08/19/2025 01:03:29 Version: 2511.1.6 will show in the browser. The UTC time should be the LogServer UTC time. The version string contains the release name and builder number.

> **Note:**
>
> Change port 8080 to your configured port if not using the default port, and change http to https if installed with HTTPS mode.

If the log server verification fails, please check the following logs:

- Run docker logs logserver to check docker logs.
- For Linux - `$HOME/LogServer/Config/weblogs.txt` (change $HOME/LogServer to your real installed path if not using default one)

- For Windows - `C:\Users\YourUserName\LogServer\Config\weblogs.txt` (Change YourUserName to real username. Change C:\Users\YourUserName\LogServer to your real installed path if not using default one)

Advanced Configuration of Log Server

On Linux or Windows

Run `docker stop logserver` to stop the logserver

By default, the log server is configured with the below values. To make the changes, edit the `StartLogServer.sh` or `StartLogServer.bat` if installed in windows.

```
-e MAX_RESERVE_DAYS=7
-e MAX_DISK_USAGE_PERCENTAGE=85
-e LOCAL_DOWN_ONLY=true
-e OPENSEARCH_JAVA_OPTS="-Xms2G -Xmx2G"
```

LogServer Configuration Options	Default Value	Value Range	Desc
LOG_LEVEL	2	0-4	0=Trace, 1=Debug, 2=Info, 3=Warning, 4=Error
CORS_ORIGINS		“Url”or “url1;url2;url3”	Modify this value to allow CWA H5/Chrome client uploading AOT logs. Support multiple urls separated by “;”
MAX_RESERVE_DAYS	7	1~30	Log Server stores log entries max days based on the TimeStamp field. Logs that were inserted 7 days ago will be deleted. Check every 10 minutes.

LogServer Configuration Options	Default Value	Value Range	Desc
MAX_DISK_USAGE_PERCENTAGE	85	10~90	Log Server monitors the data storage percentage. If the usage percentage is more than 90%, Log Server will delete old logs day by day until the usage percentage is less than 90%. Check every 10 minutes.
LOCAL_DOWN_ONLY	true	true/false	If true, only the machine installed Log Server can visit the /Download/*APIs. If false, then other machines with AuthKey could visit the /Download/* APIs.
OPENSEARCH_JAVA_OPTS	"-Xms2G -Xmx2G"	2G ~ MaxMem/2	Opensearch memory configurations. Provide more memories if there are a lot of machines that send logs to Log Server. Eg., 0~999 machines: 2GB 1000~1999 machines: 4GB 2000~9999 machines: 6GB

Run `./StartLogServer.sh` on Linux to verify if the changes are updated.

Run `StartLogServer.bat` on Windows to verify if the changes are updated.

On Citrix Connector Appliance

Below are the steps for to configure the advanced settings of log server on Citrix connector appliance via local API.

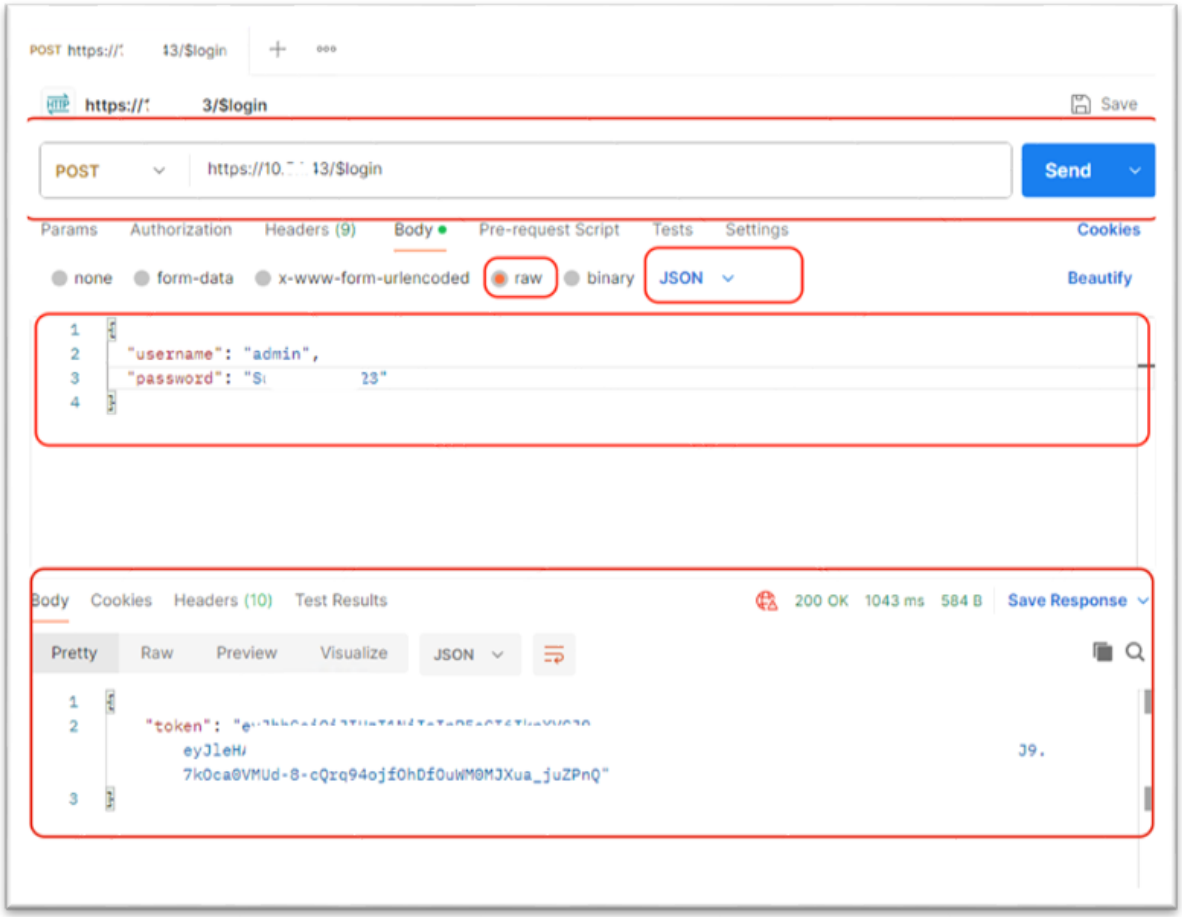
You can execute this on either Postman or Curl or PowerShell. Below is an example of instructions executed on Postman:

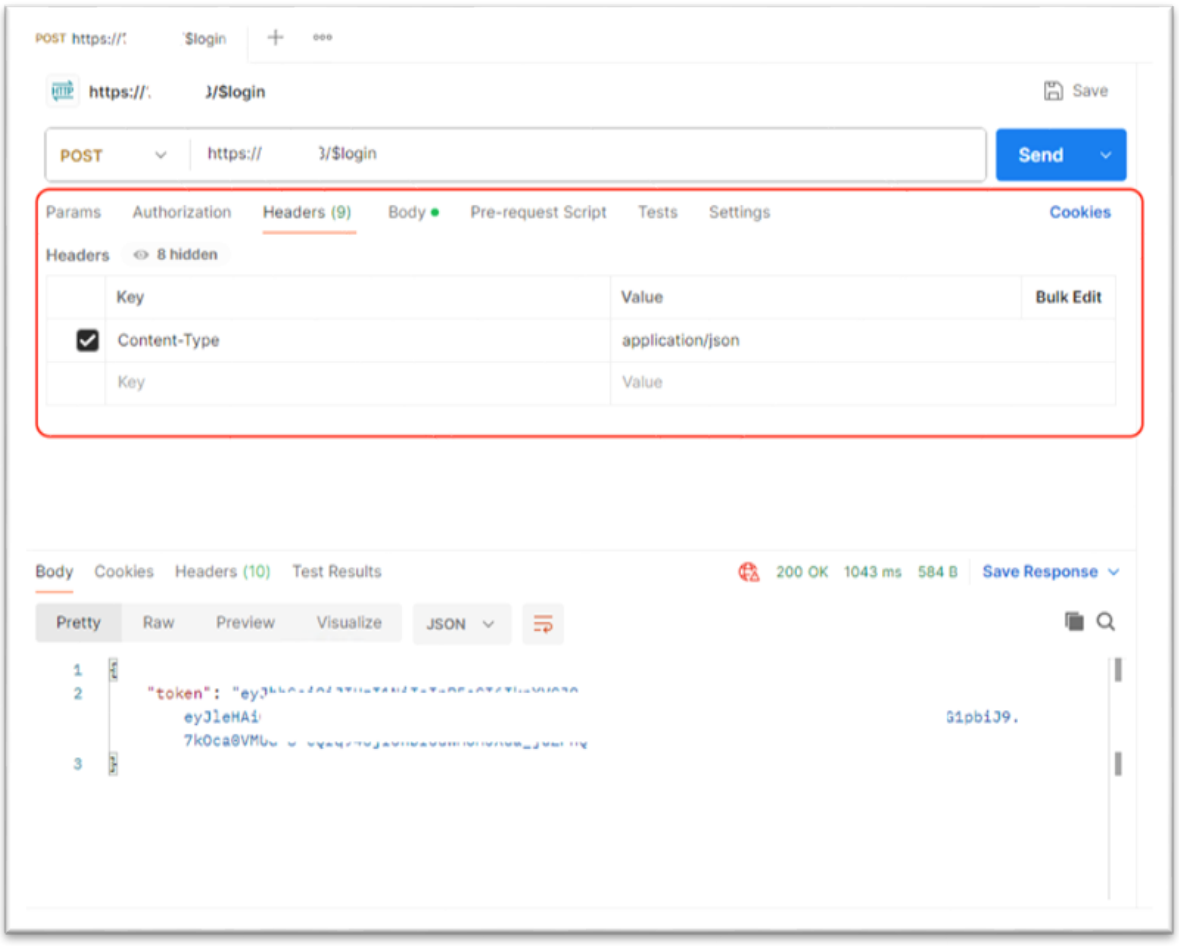
1. **Authentication: Generating a JWT**

All API calls must be authenticated using a JSON Web Token (JWT). You must first generate a token which is then included in the header of subsequent requests.

Step 1.1: Generate the Token To generate the token, run a POST request to the $login endpoint.

- Endpoint: POST https://[ip]/$login
- Body: You'll need to include the necessary JSON payload for authentication (e.g., username and password).
  If the credentials are correct, the API will return a token.

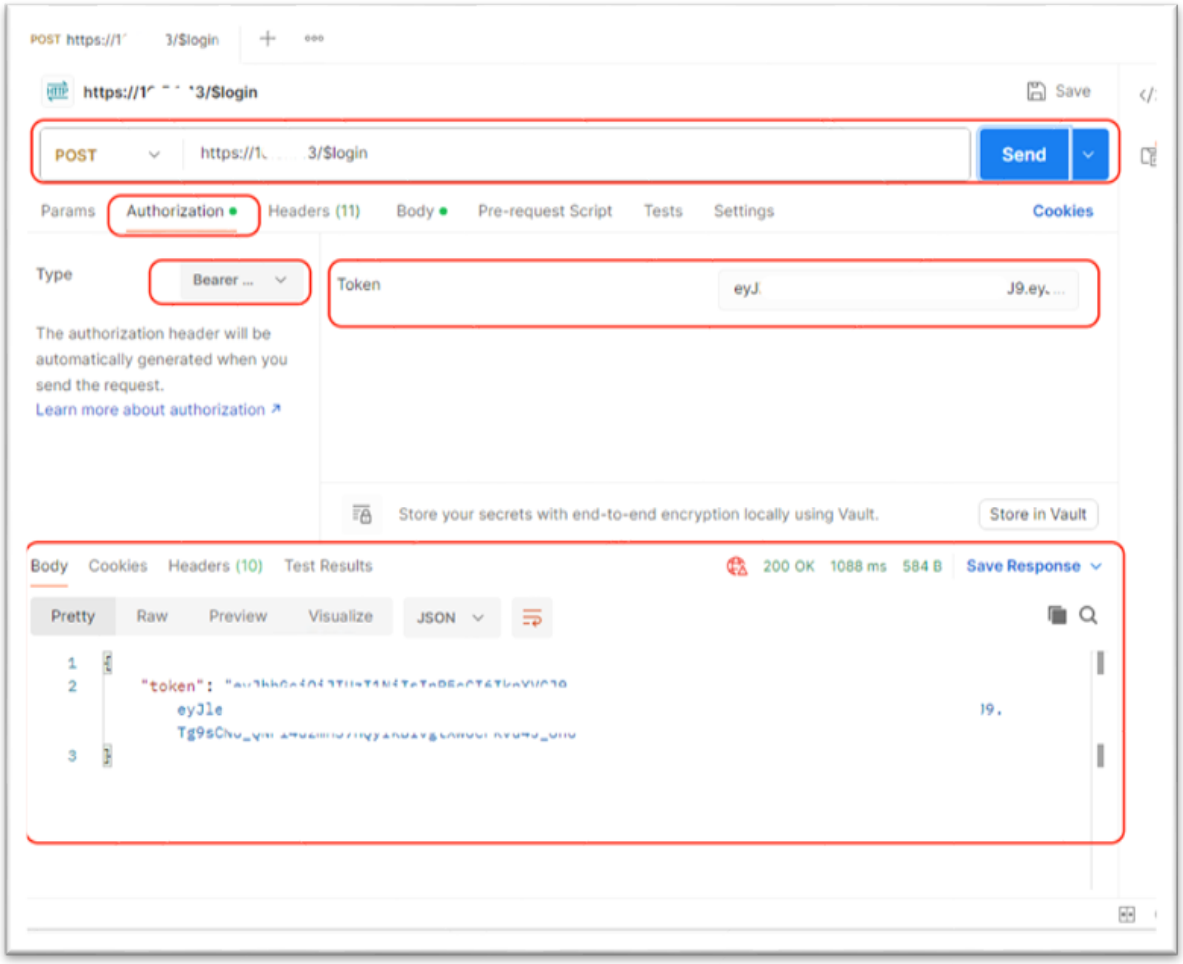




Copy this token value for use in the next steps.

Step 1.2: Authorize API Calls Include the generated token in the Authorization header for all subsequent API calls. The token must be prefixed with Bearer.

Authorization: **Bearer abCD.efGH.ijKL**



A valid token will allow the API call to proceed. An invalid or expired token will be rejected with an error message.

1. **Adding advanced settings**

Once authenticated, you can configure the settings for a target container like logserver **Configure the MAX_RESERVE_DAYS**

This action configures the MAX_RESERVE_DAYS for the logserver.

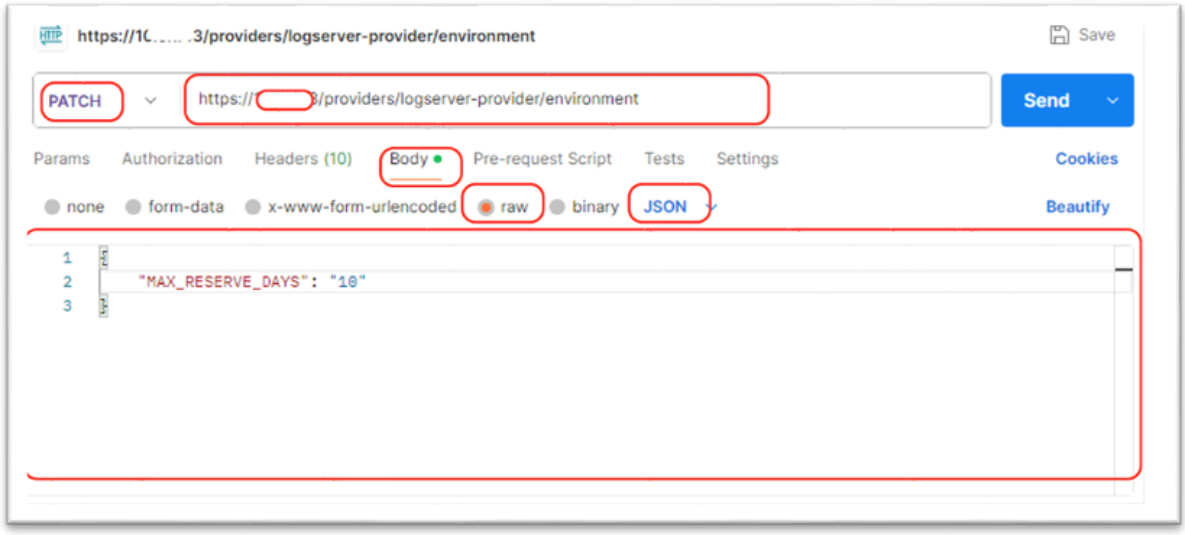
- Endpoint: https://[ip]/providers/logserver-provider/environment
- Method: PATCH

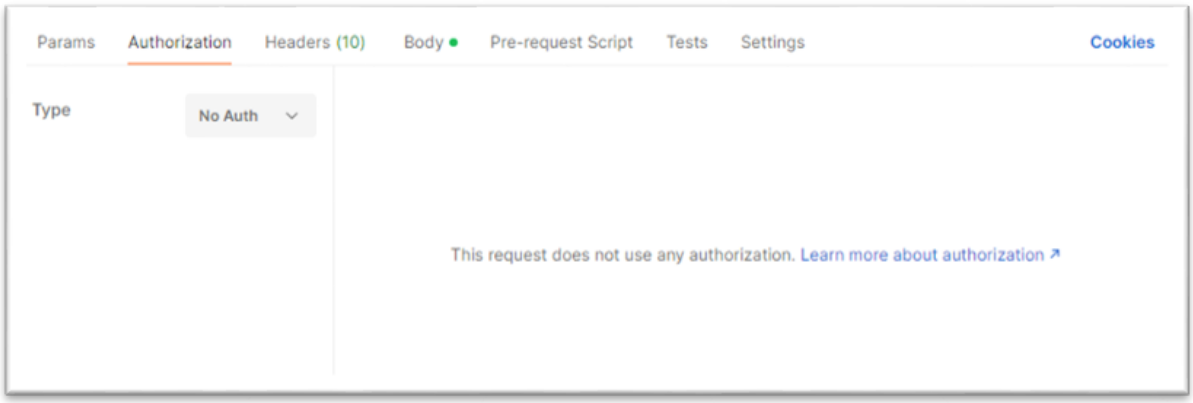
Request Body {
“MAX_RESERVE_DAYS”: “7”
}

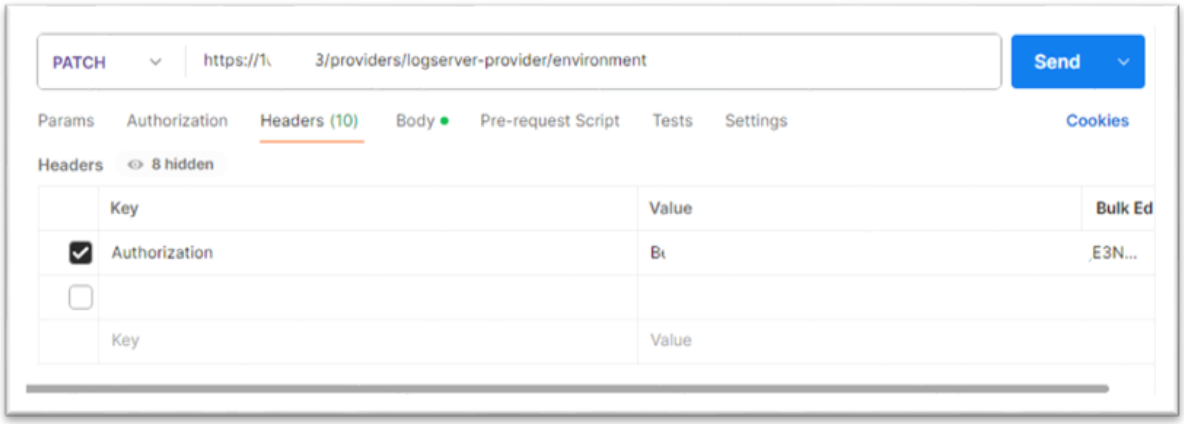
Fields

- MAX_RESERVE_DAYS (string, required): Default value is 7 days. Log Server stores log entries max days based on the TimeStamp field. Logs that were inserted 7 days ago will be deleted. Check every 10 minutes.

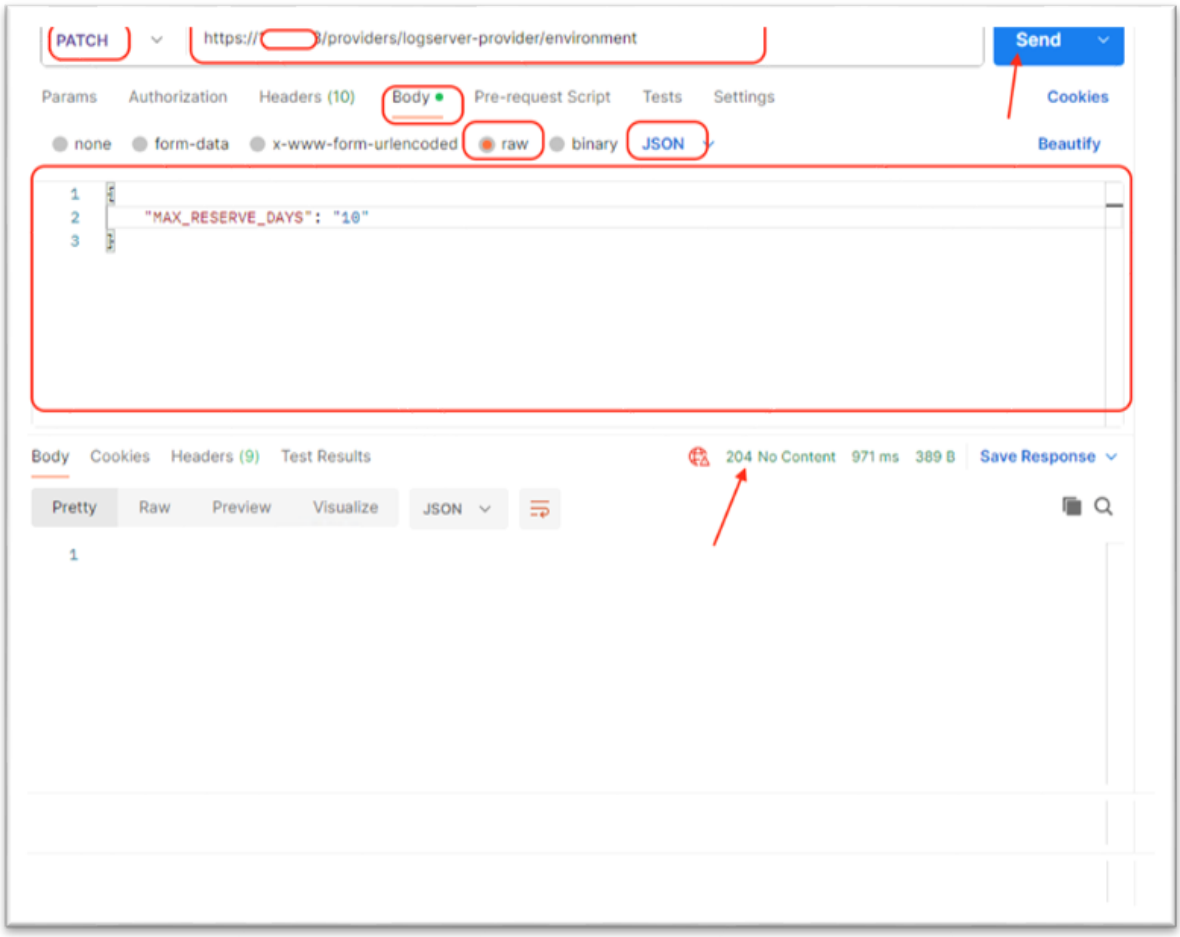
In this example, we are changing it to 10 days.



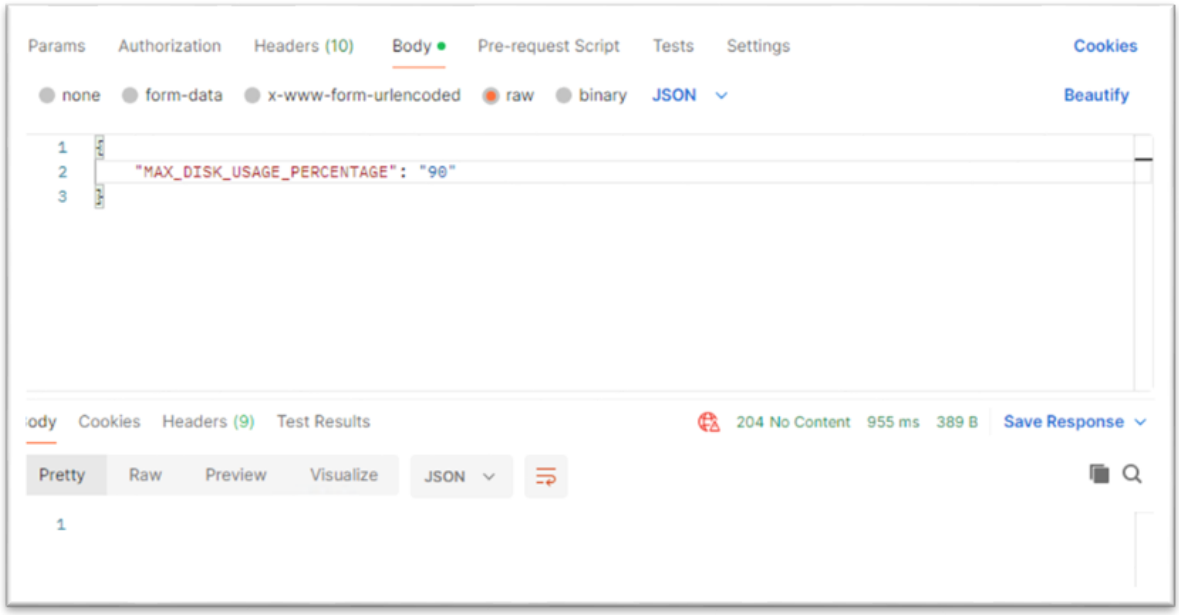




Responses ☒ Success (204 OK)
Indicates setting is properly configured



Similarly, we can also adjust the Max Disk Usage Percentage from its default of 85. We are changing it to 90 here. When disk usage hits 90%, old logs are deleted to make space for new ones.



Response ☒ Success (204 OK)
Indicates setting is properly configured

Log Server Recommendation

July 25, 2026

Daily Log Volume (Storage)

Item	Number of Machines	Sessions per Day/Machine	Avg. Log Size per Session	Total Logs per Day
Persistent VDAs	1000	2	2 MB	4,000 MB (≈4 GB)
Pooled VDAs	1000	50	2 MB	100,000 MB (≈100 GB)
CWA Endpoints	8000	4	1-1.5 MB	48,000 MB (≈48 GB)
Delivery Controllers	4	—	300 MB per DDC	1,200 MB (≈1.2 GB)

Item	Number of Machines	Sessions per Day/Machine	Avg. Log Size per Session	Total Logs per Day
StoreFront Servers	4	—	300 MB per server	1,200 MB (≈1.2 GB)
NetScaler Gateway /PVS/Session Recording/ FSA and other servers	4	—	300 MB per server	1,200 MB (≈1.2 GB)
Daily Total Log Size (Post Compression)	—	—	—	**Before compression is ≈156 GB/day** **Post compression is 80 GB to 125 GB/day**

Sizing Assumptions

- This example represents a reference deployment of approximately 10,000 managed endpoints and infrastructure components.
- For sizing purposes, Delivery Controllers, StoreFront servers, Gateway, PVS, FAS, Session Recording, and other infrastructure components are estimated at 300 MB per server per day.
- The 300 MB per server value is a conservative estimate used for storage planning and capacity calculations.

> **Note:**
>
> 1. These values are approximate and can differ widely between deployments. Log volume depends on site size, user activity, component mix, and retention needs. Use these numbers only as a baseline to begin planning and adjust the storage allocation based on your environment's actual usage, growth trends, and performance testing.
> 2. Log server uses the LZ4 compress algorithm by default. The compression percentage could be between 20% to 50%; it means LogServer would take half of the disk space when it does the maximum compression or 80% disk space when it does the minimum compression.
> 3. A single AOT Log Server can support up to **128,000 active component connections** and process **up to 10,000 log events per second**. The Log Server is designed to continuously

collect and store AOT logs around the clock, provided the environment remains within these connection and throughput thresholds.

4. These recommendations are driven not by the size of the virtual machine, but by the capacity of the underlying **search and indexing engine** that ingests, processes, and stores incoming AOT logs. This search layer ultimately determines maximum scale because it handles indexing, querying, and retention operations. As environments grow, this layer must scale horizontally to maintain performance.
5. Today, the Log Server ships as a single combined node that includes both the Log Server service and the search/index engine. This simplifies deployment but limits the ability to scale the search tier independently.
6. In upcoming releases, Citrix will transition to a **separated architecture**, where ingestion and storage/indexing run on independent nodes. This will allow customers to scale the search tier as their environment grows, improve performance across large or multi-site deployments, and remove the scalability ceiling imposed by the current single-node design.
7. Deploy the AOT Log Server on SSD-backed storage. SSDs are strongly recommended to ensure reliable log ingestion, fast search performance, and efficient correlation, especially in high-volume environments.

AOT Log Server Storage Sizing

Sizing storage correctly is one of the most important steps when planning the AOT Log Server. The total storage required depends on how many machines are sending AOT logs, how much data is generated per day, and how long logs must be retained.

Because each Citrix environment behaves differently, customers should begin by estimating their daily log volume using the log generation patterns of their VDAs, Citrix Workspace App endpoints, Delivery Controllers, and StoreFront servers. Once the daily volume is known, the total storage can be calculated using the formulas below.

Storage Calculation Formula
Total Storage (GB) = Daily Log Volume (GB) × Retention Days

AOT Log Server Storage Sizing Guidance

Use the following table as a reference for estimating daily log volume and storage requirements based on deployment size. These estimates are intended for capacity planning and may vary based on user activity, session density, enabled components, and deployment architecture.

Daily Log Volume Estimate

Deployment Size	Daily Log Volume Before Compression	Daily Log Volume After Compression (LZ4)	Recommended Category
5,000 machines	~78 GB/day	~40–63 GB/day	Small
10,000 machines	~156 GB/day	~80–125 GB/day	Medium
50,000 machines	~780 GB/day	~400–625 GB/day	Large
100,000 machines	~1.56 TB/day	~800 GB–1.25 TB/day	Enterprise

Retention-Based Storage Estimate

Deployment Size	7-Day Retention	10-Day Retention
5,000 machines	~300–450 GB	~400–630 GB
10,000 machines	~600 GB–1 TB	~800 GB–1.25 TB
50,000 machines	~3–5 TB	~4–6.3 TB
100,000 machines	~6–10 TB	~8–12.5 TB

> **Note:**
>
> - These values are approximate estimates intended for capacity planning. Actual storage requirements depend on deployment architecture, user activity, enabled components, retention period, and compression efficiency.
> - Estimates are based on the 10,000-machine reference deployment model and scaled linearly.

Uploading the AOT logs from CVAD Core Components

July 25, 2026

Beginning with CVAD 2507 CU1 and CVAD 2511, AOT logs are gathered from all applicable components. The supported components include DDC, StoreFront, Citrix Director, Windows VDA, Linux VDA, Mac VDA, License Server, FAS, Cloud Connector, PVS, WEM, Session Recording, Citrix Workspace UI, and Citrix Gateway.

Centralized log forwarding configuration via Citrix Web Studio for CVAD on-prem

Citrix Web Studio provides a centralized location for configuring Always-on Tracing (AOT) log forwarding across the site. This setting defines where AOT logs are sent and automatically propagates the configuration to relevant components in the environment.

When you configure the log server in Web Studio:

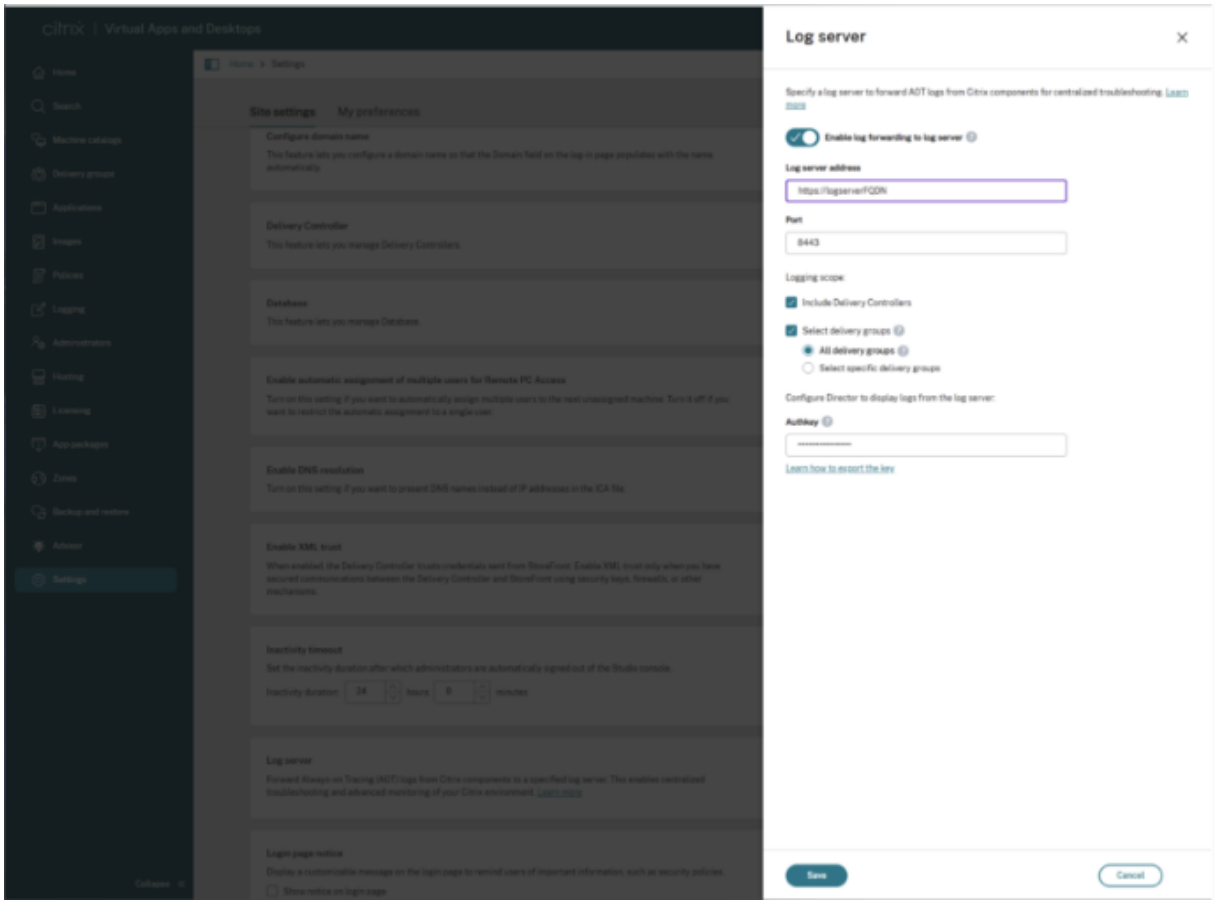
- The configuration is written in the **Site database**.
- **Delivery Controllers** retrieve the updated settings and distribute them to the VDAs in the selected delivery groups.
- **VDAs** use the received configuration to forward their AOT logs to the specified log server.
- **StoreFront** that participates in AOT logging receives the necessary details through the same centralized mechanism.
- **Director** uses the provided AuthKey and log server details to query and display AOT logs for monitoring and troubleshooting.

This centralized approach removes the need to configure each component individually. You set up the log server details once in Web Studio, and Citrix automatically propagates the configuration across the site.

Below is the list of components that will get log server details once it is set on Web Studio:

- Citrix Director/Monitor
- Citrix Delivery Controller
- Windows VDA
- Linux VDA
- Mac VDA
- StoreFront
- Session Recording Agent
- CWAs

To configure the log server settings for on-prem environment



1. Sign in to **Web Studio** and select **Settings** in the left pane.
2. In the **Log server** tile, select **Add** if no log server is configured.
3. On the **Log server** page, turn on **Enable log forwarding to log server**.
4. Enter the **Log server address** and **Port**. VDAs and Delivery Controllers use this information to forward AOT logs.

   > **Note:**
   >
   > a) If you are using the Citrix Connector Appliance to deploy the Log Server, ensure that only port 443 is configured.
   >
   > b) Updating the Log Server address and port in Citrix Studio works initially, but if you revisit the settings and modify the Log Server details without re-entering the authentication key, the authentication key placeholder becomes empty after saving, even though the operation appears successful. This results in missing or invalid configura-

> tion in Director and Monitor. As a workaround, always re-enter and validate the authentication key whenever modifying Log Server settings until this issue is resolved.

5. Choose which components forward logs:
   a) **Include Delivery Controllers**: Sends AOT logs from all Delivery Controllers.
   b) **Select delivery groups**: Sends logs from VDAs in selected delivery groups.
      - **All delivery groups**: Sends logs from all VDAs.
      - **Select specific delivery groups**: Choose individual delivery groups to include.
6. Enter the AuthKey generated on the log server. The dDirector uses this key to authenticate and retrieve logs.
7. Select **Save** to apply the configuration.

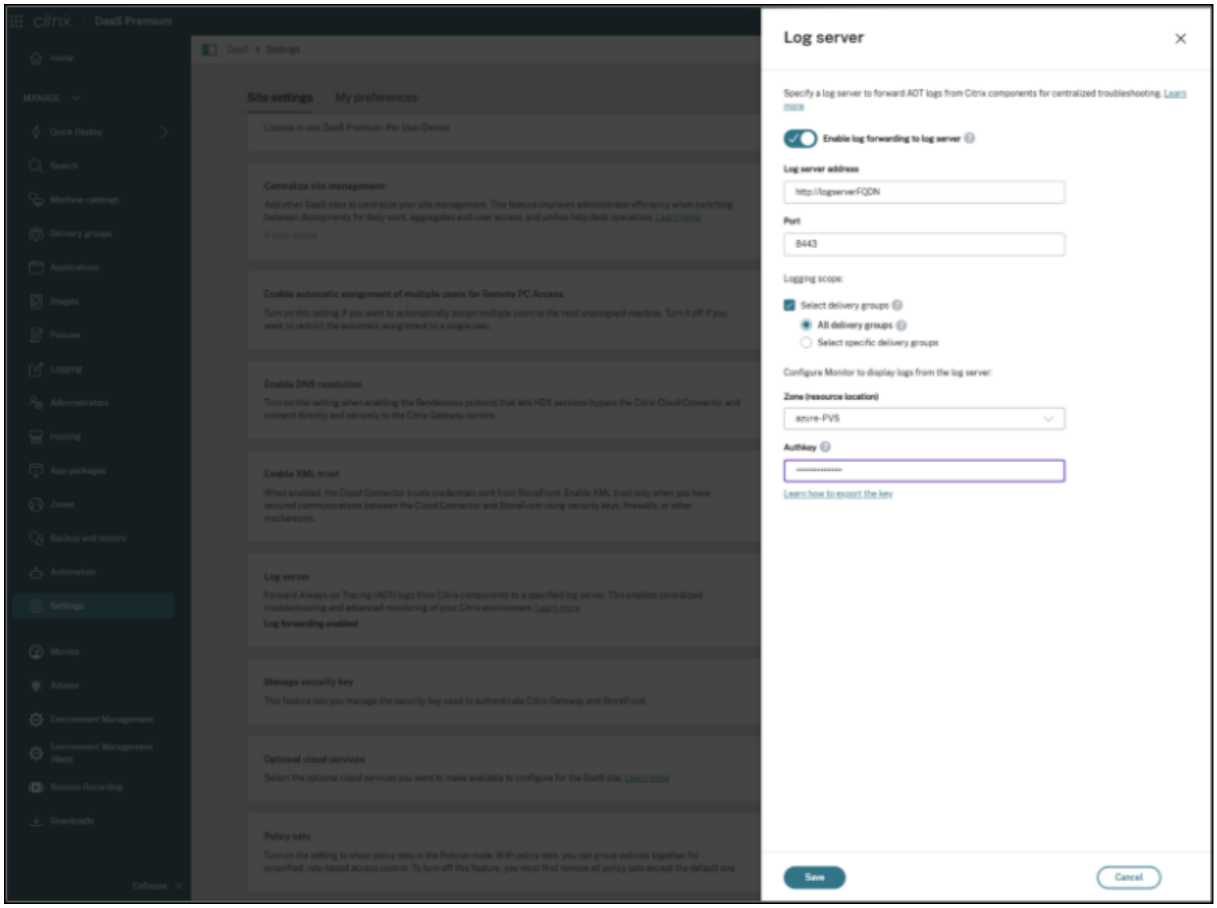
To edit an existing configuration

To update the log server or modify the logging scope, go to **Settings > Log server** and select **Edit**.

Configure the log server settings for Citrix DaaS/Cloud environment via DaaS settings

Always On Tracing (AOT) is supported for both on-premises (CVAD) and Citrix DaaS (cloud) deployments. However, log collection behaviour differs based on where the components are hosted.

After installing and configuring the AOT Log Server (on Linux, Windows, or Connector Appliance), complete the following to enable log forwarding to the Log Server in Citrix DaaS settings for Cloud deployments:



Citrix Web Studio for DaaS (DaaS settings) provides a centralized location to configure Always-on Tracing (AOT) log forwarding for cloud-managed sites. This configuration determines where AOT logs are sent and controls which VDAs participate in log forwarding. Monitor uses these settings to retrieve and display logs for troubleshooting.

1. Sign in to **Web Studio** for DaaS (DaaS Settings) and select **Settings** in the left pane.
2. In the **Log server** tile, select **Add** if no log server is currently configured.
3. On the **Log server** page, enter the **Log server address** and **Port**. VDAs use this information to forward AOT logs to the server.
4. Define which VDAs forward logs:
    a) Select **Select delivery groups** to control the scope.
    b) **All delivery groups** (default) forwards logs from all VDAs in the site.
    c) To target specific VDAs, select **Select specific delivery groups**, and choose the delivery groups you want to include.

> **Note:**
>
> If you are using the Citrix Connector Appliance to deploy the Log Server, ensure that only

> port 443 is configured.

5. Select the Zone (resource location) where the log server is deployed.
6. Enter the Authkey generated by the log server. Monitor uses this key to authenticate and retrieve logs for display.
7. Select Save to apply the configuration.

Log forwarding configuration for other components

Configure AOT log upload for StoreFront in Citrix Cloud deployments

For Citrix Cloud (hybrid) deployments, when on-prem storefront is used (instead of Citrix Workspace) with Cloud DDC, StoreFront does not automatically receive the AOT Log Server configuration from the Cloud Connector. Run the following PowerShell command on the StoreFront server to manually configure the AOT Log Server endpoint.

```
# Enable - Run the following command in StoreFront PowerShell

Enable-CitrixAOTUpload -AoTDataStoreEndpoint https://<FQDNofLogServer
    >:<server_port> -Role StoreFront

# Disable - Run the following command in StoreFront PowerShell
Disable-CitrixAOTUpload
```

> **Note:**
>
> Execute this command only for Citrix Cloud (hybrid) deployments that use Cloud DDCs. Do not run this command for on-premises CVAD deployments where StoreFront is paired with on-premises Delivery Controllers, as the Log Server configuration is automatically synchronized and no manual configuration is required.

Configure AOT logs upload for Session Recording Server

You can configure the Session Recording Server to upload AOT logs to a central log server by running the following PowerShell command on the Session Recording Server machine:

```
#Enable - run following command in Session Recording Server PowerShell
Enable-CitrixAOTUpload -AotDataStoreEndpoint <https://FQDNofServer>:<
    server port> -Role SessionRecording

#Disable - run following command in Session Recording Server PowerShell
Disable-CitrixAOTUpload
```


Configure AOT logs upload for FAS

You can configure FAS to upload AOT logs to a central log server by running the following PowerShell command on the FAS Server:

```
#Enable - run following command in FAS PowerShell
Enable-CitrixAOTUpload -AotDataStoreEndpoint <https://FQDNofServer>:<
    server port> -Role FAS

#Disable - run following command in FAS PowerShell
Disable-CitrixAOTUpload
```

> **Note:**
>
> Always-on tracing is not available if you install or upgrade using the FAS installer MSI file - FederatedAuthenticationService_x64.msi.
>
> Always install FAS from the CVAD MetaInstaller or XenDesktopFasSetup.exe or FasSetup_xxxx.exe

Configure AOT logs upload for PVS

You can configure PVS to upload AOT logs to a central log server by running the following PowerShell command on the PVS Server:

```
#Enable - run following command in PVS PowerShell
Enable-CitrixAOTUpload -AotDataStoreEndpoint <https://FQDNofServer>:<
    server port> -Role PVS

#Disable - run following command in PVS PowerShell
Disable-CitrixAOTUpload
```

> **Note:**
>
> If you have multiple PVS servers, run the command to enable or disable AOT log uploads on each one.

Uploading AOT logs for License Server

- The Logserver URL needs to be updated inside SLS config file, which can be find at the location "C:\Program Files (x86)\Citrix\Licensing\WebServicesForLicensing\SimpleLicenseServiceConfig.xml" with XML tag "AotServerURL"

Example:

```
<AotServerURL>https://<FQDNofLogServer:443>//ctxlogserver/
    UploadCitrixLogStream</AotServerURL>
```


> **Note:**
>
> **When are logs generated in License Server?**
>
> License Server AOT logs are generated only during specific events such as: License Server installation or upgrade, WSL (Citrix Web Services for Licensing) service restart, License activation failure, Dark site activation failures and Clock-related licensing issues
>
> If none of these events occurred recently, the absence of logs does not necessarily indicate a problem.

Quick Checks

1. Confirm whether a qualifying event (restart, upgrade, activation issue, etc.) occurred recently.
2. Restart the Citrix Web Services for Licensing service and allow up to one hour for upload processing.
3. Verify the AOT Log Server URL is configured correctly.
4. Confirm the Citrix Web Services for Licensing service is running.
5. Check: `C:\Program Files (x86)\Citrix\Licensing\LS\resource\aot`
   If the file contains data, logs may be waiting for upload.

> **Note:**
>
> License Server AOT logging is event-driven, not continuous. It is normal to see no logs when no qualifying events have occurred.

- The License Server periodically uploads the logs to AOT server automatically.

Configure AOT logs upload for Citrix Connector

To collect Cloud Connector logs in the AOT Log Server, the following requirements apply:

- Cloud Connector must be on **version 6.169.0.22492 or later**.
- In **Cloud DDC deployments**, Log Server configuration details are automatically synchronized to the registry with **May release**, and no manual configuration is required
- In **hybrid deployments (on-prem DDC)**, manual registry configuration is required to enable log collection. Create the following registry entries on the Cloud Connector machine:

Path:
`HKLM\Software\Citrix\ConfigSyncService`

Entries:

- `LogServerEnabled (REG_DWORD)= 1`
- `LogServerEndpoint (REG_SZ)= <https://<logserver-fqdn>:<port>>`

Configure AOT logs upload for Citrix Workspace apps

Citrix empowers Citrix Workspace app clients to automatically upload diagnostic logs to the centralized log server. This capability enhances troubleshooting efficiency by providing administrators with immediate access to client-side diagnostic information without requiring manual log collection. The client side AOTs will be automatically uploaded, and it triggers when AOT feature is enabled in the whole CVAD environment, and when user launches a resource.

System requirements

- Citrix Virtual Apps and Desktops 2511 or CVAD 2507 CU1 and later versions
- Citrix Workspace app 2511 or CWA 2507.1 CU1 and later versions
- Ensure the required version of the selected Citrix Workspace app (CWA) is installed. For more information, see Supported versions

Prerequisite

Citrix Workspace app web launches **For Citrix Workspace app web** launches (including WebHelper, direct ICA file download, and web browser extension launch methods), StoreFront must be configured to use the modern User Interface (UI Experience).
For instructions on enabling the modern UI experience, see the StoreFront documentation.

> **Note:**
>
> This prerequisite applies only to web-based launches and is not required for native Citrix Workspace app launches.

Configure AOT logs upload for NetScaler

- SSH into NetScaler CLI and then go into the NetScaler shell by running “shell”command.
- Go into /var/analytics_conf directory

```
cd /var/analytics_conf
```

- Generate the schema file by executing an interactive python script “python {auditlog_schema_generator_for_hec_export.py}”. The schema file defines the HTTP headers (Auth-token, User-Agent etc) and Payload attributes (Role, Machine IP etc).

Snippet of the output generated by this script

```
root@Raju-adc2# python auditlog_schema_generator_for_hec_export.py
Schema Builder: Follow the prompts to define your schema.
Include 'Auth-Token'? (Y/n):
```



```
Enter the display name (alias) for 'Auth-Token' (press Enter to keep as is): Authkey
'Auth-Token' can only go in 'header'. Automatically selected.
Include 'HostName'? (Y/n): n
Include 'Content-type'? (Y/n): y
Enter the display name (alias) for 'Content-type' (press Enter to keep as is):
'Content-type' can only go in 'header'. Automatically selected.
Default value for 'Content-type' is 'application/json'.
Do you want to change it? (y/N):
Include 'User-agent'? (Y/n):
Enter the display name (alias) for 'User-agent' (press Enter to keep as is):
'User-agent' can only go in 'header'. Automatically selected.
Default value for 'User-agent' is 'AUDITLOGS/1.0'.
Do you want to change it? (y/N):
Include 'Role'? (Y/n): y
Enter the display name (alias) for 'Role' (press Enter to keep as is):
Where should 'Role' be included? (HEADER/payload): HEADER
Default value for 'Role' is 'GW'.
Do you want to change it? (y/N): y
Enter new default value for 'Role': Gateway
```

Sample schema file generated

```
{

     "Header": {

         "Auth-Token": {

             "name": "AuthKey"
          }
 ,
         "Content-type": {

             "name": "Content-type",
             "value": "application/json"
          }
 ,
         "User-agent": {

             "name": "User-agent",
             "value": "AUDITLOGS/1.0"
          }
 ,
         "Role": {

             "name": "Role",
             "value": "Gateway"
          }
 ,
         "MachineIP": {

```



```
            "name": "MachineIP"
         }
,
        "MachineName": {

            "name": "MachineName"
         }

      }
,
    "Payload": {

        "LogLevel": "Level",
        "MessageContent": "Message",
        "ModuleName": "Module",
        "Time": "TimeStamp"
      }
,
    "Format": {

        "delimiter": ""
      }

 }
```

- Exit the shell and come back to the NetScaler CLI
- Create a syslog Action which defines the Syslog Server, Ports and the schema file (created in the previous step) etc

```
add audit syslogAction {
 Action Name }
  {
 AOT Server IP/FQDN }
  -serverPort {
 Destination Port }
  -logLevel {
 Log Level }
  -transport HTTP -httpAuthToken {
 Authentication Token generated on the AOT sever }
  -httpEndpointUrl {
 AOT Server URL where logs will be sent }
  -httpSchemaFile {
 Schema File created in Previous Step }
```

Example:

```
add audit syslogAction act1 10.102.154.196 -serverPort 8088 -logLevel
    ALL -transport HTTP -httpAuthToken
    fbb94d5b234d86a6cb155e3f808dd33c9698a0d1a866e814573b1909e4c9e56896f433da1071e3
     -httpEndpointUrl "/services/collector/event" -httpSchemaFile sample
    .json
```


- Bind the syslogAction to the Audit Syslog Policy

```
add audit syslogPolicy {
 PolicyName }
  True {
 Name of the Syslog Action }
```

Example:

```
add audit syslogPolicy pol1 true act1
```

Viewing and Using AOT Logs

July 25, 2026

Accessing Logs via Director UI

Step1

Edit the `StartLogServer.bat` file

The `StartLogServer.bat` file contains all configuration parameters used to start the AOT Log Server container, including several environment variables that control log access and behavior. One of the key settings is **LOCAL_DOWN_ONLY**.

- When **LOCAL_DOWN_ONLY=false**, the Log Server accepts remote log-viewing requests. This allows **Citrix Director/Monitor** to connect to the Log Server and display AOT logs directly in the Monitor UI.
- When **LOCAL_DOWN_ONLY=true**, the Log Server restricts log access to local-only connections. In this mode, you must connect directly to the Log Server machine to view logs, and **Citrix Director Monitor will not be able to retrieve or display logs**.

File Edit View

```
@echo off
REM Auto-generated startup script for LogServer
docker rm -f logserver 2>nul
docker run -d --restart unless-stopped ^
  -e USE_HTTPS=false ^
  -e STA_SERVER="" ^
  -e LOG_SERVER="Win11-AOTLogSer:8088" ^
  -e LOG_LEVEL="2" ^
  -e MAX_RESERVE_DAYS="7" ^
  -e MAX_DISK_USAGE_PERCENTAGE="85" ^
  -e CORS_ORIGINS="" ^
  -e LOCAL_DOWN_ONLY=false ^
  -e OPENSEARCH_JAVA_OPTS="-Xms2G -Xmx2G" ^
  --name logserver ^
  -p 8088:5000 ^
  -v "C:\Users\mgali\LogServer\Config":/app/Config ^
  -v "C:\Users\mgali\LogServer\Data":/app/indexdb ^
  logserver

if %ERRORLEVEL% NEQ 0 (
  echo Failed to start LogServer container.
  exit /b %ERRORLEVEL%
)
echo LogServer container started successfully.
```

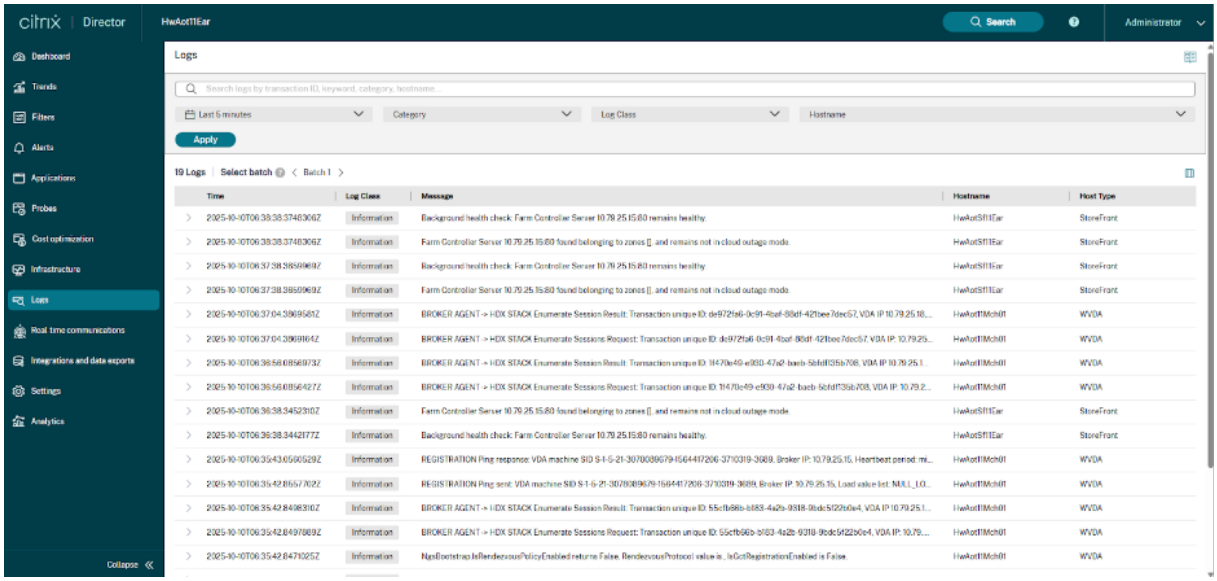
Step 2

Web Studio provides a dedicated setting that allows **Citrix Director** to connect to the AOT Log Server and display logs in the Monitor console. Once the Web Studio settings are saved, the Delivery Controllers update the site configuration, and Director receives the log server details, including the log server address, port, and authentication key. The director uses these settings to establish a secure connection to the Log Server.

Step 3

After configuration:

1. Open **Citrix Director**.
2. A new **Logs** option appears in the left navigation panel.
3. Select **Logs** and you will see the Get Started page, click on **Close** to skip it, as we have completed these settings earlier.
4. Now, you will see the Director retrieving and displaying AOT logs directly from the Log Server.



The **Logs** page in Director provides a unified view of AOT events from Delivery Controllers, VDAs, StoreFront servers, and other Citrix components. You can search, filter, and inspect logs in real time to troubleshoot issues quickly.

Free-text search: At the top of the Logs page, you'll find a free-text search bar. You can type any keyword to begin narrowing down the results. The search updates the results instantly once you apply filters. Examples include:

- Usernames
- Machine names
- Event keywords (for example, registration, authentication, STA)
- Transaction IDs
- Error messages or partial strings

Time filter: Use the time selector (for example, Last 5 minutes, Last 1 hour, Last 24 hours) to view logs generated within a specific duration. This helps you quickly isolate events around the exact time an issue occurred.

Category filter: The Category field lets you filter logs based on the type of event or subsystem. This helps you focus on logs related to Application launch, Registration, VDA configuration, Graphics, HDX Direct, ICA connection, etc.

Log Class filter: The Log Class filter groups logs by severity or event type. This lets you focus only on errors or important events when troubleshooting. Common classes include:

- Information
- Warning
- Error
- Failure

Hostname filter: The Hostname filter lets you select a specific machine —such as a VDA, Delivery Controller, CWA, or StoreFront server. This is useful when you want to drill into logs for a single endpoint or VM instead of searching across the entire deployment.

This centralized access streamlines troubleshooting by allowing for quick log retrieval and analysis from a single console. Learn more about the prerequisites and getting started in Director - Logs.

Accessing Logs via Log Server

The following content is Generate the AuthKey. To enable the access to logs, user needs

1. Modify the `StartLogServer.bat` script **LOCAL_DOWN_ONLY=false**
2. Generate the AuthKey

To keep your logs secure, you'll need an **AuthKey** before downloading them. Here's what to do:

1. **Get your AuthKey** –Use your own role-name/name to generate it.
2. **Run locally** –Scripts can only run on the machine where the Docker container is installed.
3. **Windows users** –Use the `GetAuthKey.bat` scripts instead of shell scripts.
4. **Download logs** –Once authenticated, you can safely retrieve your logs.

Generating and Managing authentication keys for AOT Log Server

You can use the provided .sh (Linux) or .bat (Windows) scripts to quickly generate a new authentication key for a specific role or user. These scripts simplify the process by returning the generated key along with the associated role.

For Linux

```
./GetAuthKey.sh role-name
```

For Windows

```
GetAuthKey.bat role-name
```

The AOT Log Server provides commands to manage authentication keys associated with different roles or users. These commands allow administrators to **add, list, validate**, and **delete** auth keys directly from the container. Use the available docker exec commands directly on the Log Server container.

- **Add a new auth key**: Creates a new auth key for a specified role or user:

```
docker exec logserver /app/authkey add <RoleName>
```

- **List all auth keys**: Displays all existing roles/users for which auth keys are created:


```
docker exec logserver /app/authkey list
```

- **Validate an auth key**: Verifies whether a given auth key is valid and identifies the associated role:

```
docker exec logserver /app/authkey validate <key>
```

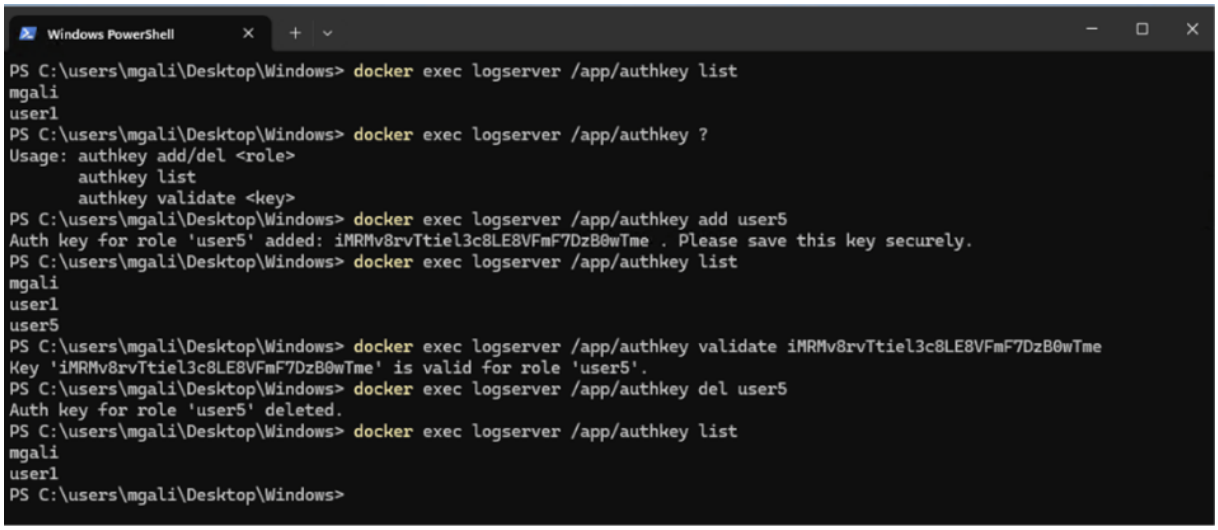
- **Delete an auth key**: Removes an auth key associated with a specific role:

```
docker exec logserver /app/authkey del <RoleName>
```

- **View command usage**: Displays available options for auth key management:

```
docker exec logserver /app/authkey ??
```

The following example demonstrates how to manage auth keys:



List machine names that have already sent their AOT logs to the logserver

The parameter is the key get from GetAuthKey.sh

For Linux

```
./ListMachines.sh ebac9b7726cb4be597c92c6769134d25
{
 "machines":["MachineName"] }
```

Empty machines means that there has no log

For Windows

```
ListMachines.bat ebac9b7726cb4be597c92c6769134d25
{
 "machines":["MachineName"] }
```


For windows powershell when LogServer starts with "-e LOCAL_DOWN_ONLY=false":

```
Invoke-WebRequest -Uri "https://logserver_fqdn:8443/ctxlogserver/
    Download/ListMachine" -Headers @{
  AuthKey = "ebac9b7726cb4be597c92c6769134d25"  }
```

Replace logserver_fqdn with logserver real FQDN, and 8443 with logserver real port. Replace https with http when installed in http mode.

Empty machines means that there has no log

Download logs by Machine name and Time range.

Time is in UTC format:YYYY-mm-ddTHH:MM:SSZ.

```
#Usage: ./DownloadLogsByTime.sh [AuthKey] [MachineName] [StartTime|YYYY
    -mm-ddTHH:MM:SSZ] [EndTime|YYYY-mm-ddTHH:MM:SSZ] [OutputFile]
#Example:
```

For Linux

```
./DownloadLogsByTime.sh ebac9b7726cb4be597c92c6769134d25 MachineName
    2025-01-01T00:00:00Z 2025-01-02T00:00:00Z logs.csv
```

For Windows

```
DownloadLogsByTime.bat ebac9b7726cb4be597c92c6769134d25 MachineName
    2025-01-01T00:00:00Z 2025-01-02T00:00:00Z logs.csv
```

For windows powershell when LogServer starts with "-e LOCAL_DOWN_ONLY=false":

```
Invoke-WebRequest -Uri "https://logserver_fqdn:8443/ctxlogserver/
    Download/TimeRange?start=2025-01-01T00:00:00Z&end=2025-01-02T00
    :00:00Z" -Headers @{
  AuthKey = "ebac9b7726cb4be597c92c6769134d25"  }
  -OutFile logs.csv
```

Download logs by keyword/s filter. Time is in UTC format

> **Note:**
>
> - A keyword can be a single word or a combination of words.
> - Keywords can match anywhere within the log message.
> - A **Transaction ID** can also be used as a keyword.

```
#Usage: ./DownloadLogsByWords.sh [AuthKey] [StartTime|YYYY-mm-ddTHH:MM:
    SSZ] [EndTime|YYYY-mm-ddTHH:MM:SSZ] [SearchWords] [OutputFile]
#Example:
```


For Linux

```
./DownloadLogsByWords.sh authkey 2025-01-01T00:00:00.000Z 2025-12-31T23
    :59:59.999Z "session launch" logs.csv
```

For Windows

```
DownloadLogsByWords.bat authkey 2025-01-01T00:00:00.000Z 2025-12-31T23
    :59:59.999Z "failed vda" logs.csv
```

For windows powershell when LogServer starts with "-e LOCAL_DOWN_ONLY=false":

```
Invoke-WebRequest -Uri "https://logserver_fqdn:8443/ctxlogserver/
    Download/SearchLog?start=2025-01-01T00:00:00Z&end=2025-01-02T00
    :00:00Z&words=failed vda" -Headers @{
  AuthKey = "ebac9b7726cb4be597c92c6769134d25"  }
  -OutFile logs.csv
```

Words are separated by space

Upgrade Log Server

July 25, 2026

For a successful upgrade, it is advisable to remove the existing logserver container beforehand by executing `docker rm –f logserver`.

- To upgrade Log Server, download the installation images and new executable log server file, and run the InstallLogServer command with the same arguments.
- The InstallLogServer will prompt whether to delete the current logserver container and images. It is safe to delete the container and images as the log data and configurations are saved separately.

- If you need to keep the old container, you should rename and stop the container (docker stop logserver) before the upgrade process.
- Don't start the old container and new container at the same time with the same –data path, otherwise the log data will be corrupted.
- After upgrading, the newly generated scripts should be used to start the Log Server.

> **Note:**
>
> This step is not required for Connector Appliance, as updates are managed automatically by Citrix.

Uninstall Log Server

July 25, 2026

1. Uninstall Log Server in Linux

Run the below commands in terminal:

```
docker rm -f logserver
docker rmi logserver

# List containers and images to make sure

docker ps -a
docker images

# Delete Log Server config and data files if not needed any more, change $HOME/LogServer to real installed path if not installed with the default one

rm -r $HOME/LogServer
```

2. Uninstall Log Server in Windows

Run the below commands in powershell:

```
docker rm -f logserver
docker rmi logserver

# List containers and images to make sure

docker ps -a
docker images

```



```
# Delete Log Server config and data files if not needed any more, change ~/LogServer to real installed path if not installed with the default one

Remove-Item -Recurse -Force ~/LogServer
```

Troubleshooting with AOT and Log Server

July 25, 2026

Top challenges addressed by AOT

1. Availability of AOT logs from a centralized log server for DDC, StoreFront™ and VDA components.
2. Improve log collection time and reduce repetitive log iterations.
3. Increase customer self-troubleshooting capabilities.
4. Accelerate issue resolution time (MTTR)
5. Quick search for logs based on various parameters/keywords.

FAQ

July 25, 2026

1. **How can I check the running log server configurations, like MAX_RESERVE_DAYS?**

   You can check the container environment values using either of the following commands:

   ```
   docker inspect logserver |findstr MAX_RESERVE_DAYS
   ```

   Or by checking the container environment:

   ```
   docker exec -it logserver env |grep MAX_RESERVE_DAYS
   ```

   If nothing is returned, it means the log server is using the default value:
   MAX_RESERVE_DAYS=7

2. **Do I need to purchase Docker Desktop licenses when installing the log server container image on Windows?**

   Yes. A valid Docker Desktop license is required.

3. **Should I use a new server for log server installation?**

   Yes. It is recommended to use a dedicated server for the log server installation to ensure performance and isolation.

4. **What is the sustained ingest capacity of a single AOT Log Server? What are the maximum and safe Events-Per-Second (EPS) limits?**

   A single AOT Log Server supports a sustained ingest capacity of **10,000 events per second (EPS)**. This value represents both the **maximum** and the **safe operating threshold** for continuous ingestion.

5. **How many components can a single AOT Log Server handle? Is there a maximum limit?**

   A single Log Server can connect up to 128,000 components, but it can only process about 10,000 logs per second. So the number of machines is rarely the issue —the real sizing factor is how many logs your users generate during peak activity.

6. **What is the burst tolerance of the AOT Log Server? How much sudden log spike can it handle without loss or backlog breach?**

   The AOT Log Server can handle short-term bursts of up to **2× the normal event rate** without losing logs. If log volume spikes beyond this (for example, 5×), the system will not be able to persist events reliably, and **OpenSearch will start dropping logs** because it cannot index them fast enough.

7. **Can the AOT Log Server compress logs? What compression ratio should we expect?**

   Yes. The AOT Log Server uses the default **LZ4 compression algorithm** for storing logs in OpenSearch. The typical compression ratio is approximately **2:1**, meaning log data is reduced to less than half its original size while maintaining fast read/write performance.

8. **For each infrastructure type (single-site on-prem, multi-site on-prem, single-region cloud, multi-region cloud, hybrid, MSP/tenant), where should the AOT Log Server be deployed and why?**

   The AOT Log Server should always be deployed **in the same line of sight as the VDAs**. This ensures stable connectivity and helps maintain low latency between the components generating logs and the log server that ingests them. As long as every component (VDAs, DDCs, StoreFront, Gateway, etc.) can reliably reach the log server, the environment will function correctly.

9. **Do we need one log server per region, or can everything be centralized? What about latency and egress?**

   You can centralize the log server, but customers should evaluate the **latency** and **egress cost** implications for their environment. The **latency between regions may affect log ingestion**, especially during high-volume periods. If the round-trip latency is high, spikes or bursts of logs may cause delays, backlog build-up, or potential loss during peak loads. Egress charges may apply when logs cross regional or cloud boundaries.

10. **Does AOT consume significant network bandwidth or system resources?**

No. AOT is designed to have minimal impact on endpoint, VDA or any component, and network performance.

AOT collects logs continuously and uploads them to the AOT Log Server in near real-time over HTTPS. Individual log records are typically very small in size (often only a few kilobytes), which helps minimize network overhead during normal operation.

The overall bandwidth consumed depends on the number of active sessions, enabled components, and environment activity. In most deployments, AOT log traffic represents a very small percentage of overall network utilization.

AOT also uses compression to reduce storage requirements and optimize data transfer to the Log Server.

11. **What are the minimum hardware specifications for supporting 1,000 machines with the AOT Log Server?**

    For environments with up to **1,000 machines**, you need: **1 node** (Log Server + OpenSearch combined), **4 vCPUs, 8 GB RAM, 2,000 IOPS** minimum (SSD or NVMe recommended), **1 Gbps NIC**. This setup is suitable for small or single-site deployments with moderate log volume.

12. **How can I troubleshoot if there is a problem with the Log Server?**

    LogServer is running as a docker container, so all docker commands could be used to find the issues:

    ```
    docker logs logserver
    docker inspect logserver
    ```

    And also, user could attach into the running container and view logs of logserver self:

    ```
    docker exec  – it logserver bash
    ```

    In the logserver docker container bash shell, user can verify the health of logserver and opensearch:

    ```
    curl http://localhost:5000/Ping
    curl http://localhost:9200/_cluster/health?pretty
    ```

    And check logs inside the container:

    ```
    tail Config/applogs.txt
    tail Config/weblogs.txt
    ```

    If need more logs, user could modify the LOG_LEVEL=0 in StartLogServer.sh/StartLogServer.bat and restart the logserver by these script files. Then the detailed logs will include TRACE, DEBUG, INFO, WARN, ERROR all levels.

13. **Recover from an improperly detached Log Server storage disk**

If you remove or detach the Log Server storage disk directly from the **hypervisor** or **cloud provider** without first detaching it from the Citrix Connector Appliance administration UI, the Connector Appliance retains the storage configuration and assumes that the disk is still attached. As a result, the previously attached disk remains configured in the Connector Appliance, attaching a new disk fails, and you may see an error similar to: **A local disk is already mounted to the provider**.

Option 1 (Recommended): If the original disk is still available:

a) Reattach the original disk to the Connector Appliance VM from the hypervisor or cloud provider.

b) Reboot the Connector Appliance.

c) After the appliance starts successfully, detach the disk using the Connector Appliance administration UI.

d) You can now attach a new disk if required.

Option 2: If the original disk is no longer available, manually remove the stale storage configuration by executing the following API request. Retrieve an Authorization token and run the appropriate command:

Linux:

```
curl -X POST "https://<connector-fqdn>/storage/$detach" \
-H "Content-Type: application/json" \
-H "Authorization: Bearer <token>" \
-d '{
 "targetProvider":"logserver-provider","storageType":"local" }
 '
```

Windows:

```
curl -X POST "https://<connector-fqdn>/storage/$detach" ^
-H "Content-Type: application/json" ^
-H "Authorization: Bearer <token>" ^
-d "{
 \"targetProvider\":\"logserver-provider\",\"storageType\":\"local
    \" }
 "
```

> **Note:**
>
> The API may return an error even though the stale storage configuration is successfully removed. Verify the storage configuration before retrying the disk attachment.

As a best practice, always detach the Log Server storage disk from the Connector Appliance administration UI before removing or detaching the disk from the hypervisor or cloud provider.

This ensures that the Connector Appliance cleans up its storage configuration and prevents stale storage references.

General Response Code for Log Server API

July 25, 2026

This table outlines the standard HTTP response codes used by the Log Server's API to communicate the result of client requests. These codes help clients understand whether their requests were successful or if errors occurred, and what kind of errors were encountered. Understanding these response codes is crucial for diagnosing issues, debugging API calls, and ensuring smooth integration with the Log Server.

Error Code	Status	Description
200	OK	The request was successful.
400	Bad request.	The server could not process the request due to missing parameters, malformed syntax, or absence of required HTTP headers. For example, if fields like MachineName or Role are missing in the request headers, this response is triggered.
401	Not Authorized.	The request was denied because the client is not authorized. This usually occurs when the IP address or the authentication key (AuthKey) used is not permitted to access the Log Server.
404	Not found.	The API path requested does not exist. This response is returned when the endpoint is invalid or mistyped.

Error Code	Status	Description
500	Internal error.	Indicates that the Log Server encountered an unexpected condition preventing it from fulfilling the request. This is commonly caused by server-side issues such as resource exhaustion (e.g., memory or disk space).

Known Issues and Limitations

July 25, 2026

- **Only One Log Server Is Supported Per Environment**

  The current architecture supports only a single AOT Log Server per environment, and both Web Studio and Director/Monitor are designed to work with only one Log Server configuration. All components must point to this same Log Server; configuring or distributing workloads across multiple Log Servers is not supported today. This single Log Server can handle up to 128,000 active component connections and process up to 10,000 log events per second, which is sufficient for most deployments. Since these limits are based on connection capacity and log-throughput, not on the number of components, even very large environments can operate with one Log Server as long as they remain within throughput thresholds.
  However, customers with extremely high log volumes during peak activity may reach ingestion or indexing limits due to the absence of horizontal scaling. Support for multiple Log Servers and broader scaling options will be introduced in future releases as part of the evolving architecture.

- Running the AOT Log Server on a **Windows VM hosted on XenServer** is **not supported.**
  XenServer does not support nested virtualization, which prevents Docker from running inside the guest VM. As a result, the Log Server container cannot be deployed in this configuration.

- **Docker Desktop on Windows requires an active user session for AOT Log Server**

  When the AOT Log Server is deployed on a Windows VM using Docker Desktop, the container runs within the context of the user session that started Docker Desktop. Docker Desktop is designed as a user-session-based application and does not run as a system-level service on Windows. As a result, if the user logs out, Docker Desktop stops, which causes the AOT Log Server container to stop. Because of this behaviour: The AOT Log Server stops when the user logs out

as the deployment depends on a specific user session. Other administrators cannot access or manage the running containers.

Workaround: Keep the user session active or in a disconnected state (do not log out) to ensure that Docker Desktop and the AOT Log Server container continue running.

Recommendation For production deployments, use one of the following:

- Deploy the AOT Log Server on a Linux VM using Docker Engine
- Use the Citrix Connector Appliance (CCA)-based deployment where supported

These options allow the Log Server to run independently of user sessions and provide a more reliable, always-on deployment model.

> **Note:**
>
> This approach may not be reliable in environments with session timeouts or strict security policies.
>
> This behaviour is a known limitation of Docker Desktop on Windows and is not specific to the AOT Log Server.

- **License Server logs appear with incorrect Host Type in Director/Monitor**

  When viewing AOT logs from the License Server in Director, the **Host Type** column may incorrectly display the value as **DDC** instead of **License Server**, even though the hostname is correct.

  This issue is limited to how the Host Type is represented in Director and does not impact log collection or functionality. This issue has been resolved in the latest License Server release and is fixed starting from **License Server version: 55000 or later**.

- **Citrix Cloud Connector is required to view AOT logs in Monitor**

  Customers using **Citrix Monitor** to view AOT logs must have at least one **Citrix Cloud Connector** deployed in their environment. This requirement applies regardless of where the AOT Log Server is hosted, including Linux VM or Windows VM or Citrix Connector Appliance

  Citrix Monitor relies on the **Monitor service running on the Cloud Connector** to retrieve logs from the configured Log Server and display them in the UI.

  Without a Windows Cloud Connector, AOT logs cannot be fetched or displayed in Monitor.

- **ADC Gateway logs display incorrectly in Citrix Director**

  Symptoms: When viewing Always On Tracing (AOT) logs for Citrix ADC Gateway in Citrix Director, you might observe:

  - Log Level displayed as **Error** instead of **Debug** or **Info**.
  - The **Message** column shows the entire raw log entry.

- The displayed timestamp differs from the original event timestamp.
- Only malformed JSON log records are affected. Valid log records display correctly.

Affected versions: Citrix ADC earlier than **14.1-73.x** and Citrix ADC earlier than **15.1-8.x**.

Cause: Citrix ADC Gateway uploads malformed JSON log records that cannot be parsed by the AOT Log Server. As a result, the Log Server stores the raw log entry and classifies it as an error.

Resolution: Upgrade Citrix ADC to one of the following versions or later:

- 14.1-73.x
- 15.1-8.x

Fixed in: Citrix ADC **14.1-73.x** and Citrix ADC **15.1-8.x**.

- The AOT Log Server does not currently support **FIPS-compliant deployments**. Customers with mandatory FIPS requirements should defer production deployment until FIPS support becomes available.

  Customers who do not require FIPS compliance may continue to deploy and evaluate the AOT Log Server in supported environments.

Third party notice

July 25, 2026

This release of Citrix AOT Log Server may include third-party software licensed under the terms defined in the following documents:

- Citrix Centralized Log Server Third Party Notices