Active Roles 8.2

Best Practices Guide

Legend

WARNING: A WARNING icon highlights a potential risk of bodily injury or property damage, for which industry-standard safety precautions are advised. This icon is often associated with electrical hazards related to hardware.

CAUTION: A CAUTION icon indicates potential damage to hardware or loss of data if instructions are not followed.

Active Roles Best Practices Guide
Updated - 01 October 2024, 16:58

For the most recent documents and product information, see Online product documentation.

Contents

Introduction

The Active Roles Best Practices Guide describes best practices for core Active Roles product functionalities. This document is intended for general use, because Active Roles environments have different requirements and their functionality can extend beyond the scope of the best practices listed in this document.

Installing and upgrading Active Roles

IMPORTANT: Before installing Active Roles, see *System requirements* in the *Active Roles Release Notes* to ensure that all hardware and software components meet the minimum requirements. Failure to do so can result in an unsupported configuration.

The minimum requirements for Active Roles core components only list the necessary requirements for basic Active Roles functionality. When installing Active Roles, also consider the operating system and any other software installed on the system as well.

For example, an operating system with IIS installed requires **4** GB of RAM. Active Roles with all components installed requires an additional **10** GB of RAM. That total minimum is **14** GB.

Third-party add-ons or custom scripting also require additional memory.

Supported upgrade paths

CAUTION: Upgrading from an unsupported version can result in a loss of functionality.

The supported upgrade paths can change with each version of Active Roles. For more information, see *Upgrade and installation instructions* in the *Active Roles Release Notes*.

Upgrading Active Roles

In addition to the Active Roles system requirements, check the following:

- Make sure that any third-party components or add-ons, for example Safeguard Authentication Services, are compatible with the version of Active Roles that you are upgrading to. Refer to the documentation of those components and add-ons before proceeding with an upgrade.
- Custom solutions provided by One Identity Professional Services might be version specific. Refer to any provided documentation or consult One Identity Professional Services before proceeding with an upgrade.
- Test any other customizations in a lab or staging environment before upgrading.

TIP: One Identity recommends to test upgrading in a staging environment before upgrading in production.

You can upgrade Active Roles in two ways:

- By performing an in-place upgrade. In this case, you install the new Active Roles version on top of an existing installation, providing a quick turnaround for service availability.
- By performing a clean installation, then importing the configuration data. This allows you to fall back to the previous version if you experience any problems with the new Active Roles version.

Whichever option you choose, make sure to do the following:

- Perform the upgrade as the existing Active Roles service account.
- Follow the upgrade steps described in the *Active Roles Upgrade Guide*.

NOTE: Active Roles cannot communicate with older versions of SQL. Therefore, if the source Configuration database or Management History database resides on an older version of SQL, first copy the database to a supported version.

Active Roles topology

Active Roles can either run as a single instance or as multiple instances of the Active Roles service. When running multiple instances, you can configure the Active Roles servers to use the same SQL databases or take advantage of distributed SQL technologies.

SQL database options

Active Roles supports the following SQL Server topologies.

1. Standalone
    - One or more Active Roles services using one SQL Server.
2. Multiple SQL Servers
    - Merge replication
    - Mirroring
    - Clustering
    - Always On availability groups
    - Distributed Always On availability groups
    - Transaction log shipping

TIP: One Identity recommends implementing a fault tolerant configuration such as:

- SQL mirroring
- Two Active Roles services using the same database sources

Active Roles Web Interface topologies

You can install Active Roles Web Interface either on the machine where Active Roles Administration Service is running, or on a standalone server. However, when Active Roles Web Interface is installed on a standalone server, the following limitations apply:

- Federated authentication is not supported in a standalone configuration.
- You must configure constrained delegation.

Active Roles client components

CAUTION: Hazard of data loss!

Active Roles client components are version-specific. Using mismatched or older versions can lead to a loss of functionality or data corruption.

Make sure that all Active Roles client components are up-to-date. The supported client components include:

- Web Interface
- SPML
- ADSI
- PowerShell or custom code via ADSI or EDMS calls

To check the version of Active Roles client components

1. Open the Active Roles Console.
2. To check the list of all clients currently connected to the Active Roles service, navigate to **Server Configuration** > **Client Sessions**.
3. In the **Client Version** column, take note of any older clients, and upgrade them immediately.

Multi-factor authentication and two-factor authentication support

The Active Roles Web Interface supports the following federated authentication options:

- Windows authentication
- WS-Federation
- SAML 2.0 Authentication (as of Active Roles 8.2).

WS-Federation can be used with Microsoft Entra ID or Active Directory Federation Services (ADFS). SAML 2.0 Authentication can utilize any SAML provider, like OneLogin by One Identity.

You can configure Active Roles Web Interface to use SAML 2.0 Authentication with a number of common Identity Providers. The provider can then request both primary and secondary authentication. For more information, see the *Active Roles Administration Guide*.

Additional third-party providers can be configured using the Redistributable Secure Token Server (RSTS).

For more information of federated authentication, see *Configuring federated authentication* in the *Active Roles Administration Guide*.

Integration with other One Identity and Quest products

The supported One Identity and Quest products include the following:

- Change Auditor
- Defender
- Enterprise Reporter
- Identity Manager
- Recovery Manager for Active Directory
- Safeguard
- Safeguard Authentication Services
- Starling

For more information on these products, see *Active Roles integration with other One Identity and Quest products* in the *Active Roles Administration Guide*.

Customization support

You can extend Active Roles via customization, using either out-of-the-box capabilities or other options, such as scripting. However, One Identity Support cannot provide assistance for every type of customization.

One Identity Support can provide assistance for the following customization types:

- SDK sample scripts.
- Any interactive graphical user interface (GUI) elements, such as workflows, policies, and Access Templates.
- Web Interface items, including logos and options that have a **Click here to customize** setting, such as adding to or removing attributes from pages.

However, One Identity Support cannot provide assistance for the following customization types:

- Custom scripts that do not use Active Roles SDK sample code.
- Undocumented modifications of the Web Interface configuration files.
- Third-party add-ons not provided by One Identity or Quest.

General best practices

Consider the following general best practices when using (or planning to use) Active Roles.

Minimum permissions

Active Roles requires permissions to perform its functions and tasks. These required permissions can change with each version of Active Roles. For more information on the minimum permissions, see *Minimum permissions for the Active Roles service account* in the *Active Roles Installation Guide*.

IMPORTANT: Failure to use the documented permissions will result in limited functionality with Active Roles and an unsupported configuration.

Active Roles service recovery

By default, the Active Roles service recovery option is not set. As with any Windows service, you can configure recovery options, such as restarting the computer or running a program.

One Identity recommends setting an option that is appropriate for your specific environment. For example, your can use the **Run program** option to create an email alert with the `Send-MailMessage`PowerShell cmdlet, so that you can notify the appropriate people in the event of a service failure. For more information on the `Send-MailMessage` cmdlet, see *Microsoft Send-MailMessage* in the *Microsoft PowerShell documentation*.

NOTE: `Send-MailMessage` is a Microsoft cmdlet and is not supported by One Identity.

Establishing a baseline or benchmark

One Identity recommends creating a performance baseline to avoid potential performance issues later. Active Roles contains performance monitoring (perfmon) counters that can be useful in both avoiding performance issues and troubleshooting existing ones.

After configuring and installing Active Roles, to determine if additional resources are required, One Identity recommends to monitor the overall resource usage. In some environments, the time required to determine this might take a week, while in others, it might take a month.

Once your system resources match the base usage requirements, One Identity recommends to create a baseline to establish typical resource usage in your environment.

Once you have established this typical baseline, it will be much easier to identify any abnormal resource usage scenarios.

For more information on Active Roles perfmon counters and their usage, see *Monitoring performance* in the *Active Roles Administration Guide*.

Active Roles deployment considerations

Latency between geographic locations can negatively impact user experience and data updates between regions. Consider implementing region-specific Active Roles and SQL Servers to help reduce latency.

For some environments, dedicated servers, such as job servers or Web Interface-only hosts, can help reduce the load on the main Active Roles servers.

Custom scripts

One Identity Support cannot provide assistance for custom scripts. If you need help with custom scripts, contact One Identity Professional Services.

For issues with code snippets or examples from the Active Roles SDK, contact One Identity Support.

In general, consider the following when using custom scripts.

Expensive script calls

Use the following triggers with caution, because they can significantly increase computational overhead:

- `onPreGet`
- `onPostGet`
- `onGetEffectivePolicy`

Scope

Adjust the scope of the scripts to the relevant object class. For example, if the script is intended for user objects, the first line of the script function must be `if($Request.class -ne 'user'){return}`.

Function usage

Use **`return`**, instead of **`exit`**.

Using the **`exit`** function causes the Active Roles runspace to stop and reload, resulting in a negative performance impact to Active Roles.

NOTE: Always use `return` in PowerShell.

Connect-QADService

For scripts within Active Roles, using `Connect-QADService` is not necessary, as all Active Roles cmdlets will function internally as expected.

Troubleshooting scripts

Active Roles saves a verbose log named `ds.log` that contains a system summary. This system summary also has all scripts within the Active Roles configuration exported in a plain text format. Use this export to quickly search all scripts at once to check if the `exit` function is used anywhere. This is especially useful for large environments that have many script modules.

You can enable script debugging individually for each script.

To enable script debugging

1. In the Active Roles Console, right-click the Script Module.
2. Navigate to **Properties** > **Debugging** > **Enable debugging**, then select the desired tracing level.
3. To see the debug trace, click **View Log**.

You can use performance monitoring (perfmon) to troubleshoot performance issues.

If perfmon logs indicate that the average run time of a Script Module is a potential performance bottleneck, it could also indicate poor performance when connecting to or working with Microsoft Exchange. When working with Microsoft Exchange, Remote PowerShell is used. Remote PowerShell calls are included in all Active Roles Script Module performance counters.

Management History and Report Pack use cases

Management History helps you in tracking the changes made to directory data, including the time and the initiator of the change. As such, Management History is not intended for auditing changes in data or exploring large volumes of data changes that occurred during a long period of time.

For this reason—in addition to the Management History feature—Active Roles also provides a suite of reports for change tracking and auditing, which is part of the Active Roles Report Pack.

Both the Management History and Report Pack options have their own advantages and limitations. Consider the following before choosing the one that best suits your needs.

Management History considerations

Use the Management History functionality to examine changes that were made to directory data via Active Roles. Management History is designed to help you answer the following typical questions:

- Who made the most recent changes to a given user or group object?
- Who modified a given user or group object during a set time period?
- What changes were made to a given user object in the specified time period?
- If any modifications have been planned for a given user or group object, were those modifications actually performed?
- What objects did a given delegated administrator modify during the specified time period?

Management History best practices

To investigate or troubleshoot a problem that results from the inappropriate modifications of directory data, use Management History.

Management History includes a dedicated repository to store information about data changes, referred to as the Change Tracking log, and a GUI to retrieve and display information from that repository. No additional actions, such as collecting or consolidating information, are required to build Management History results.

However, Management History also has some limitations. Because of this, before you start using Management History, consider the following.

Management History limitations

The main factor to consider is the size of the **Change Tracking** log. To ensure real-time update of the log on all Administration Service instances, the log is normally stored in the Active Roles Management History database. This imposes some limitations on the log size.

By default, the Change Tracking log is configured to store information about changes that occurred within the last 30 days.

IMPORTANT: If you increase this setting, do it carefully. Otherwise, you might encounter the following problems:

- Increasing the log size excessively will significantly increase the time required to build and display Change History and User Activity results.
- As the log size grows, so does the size of the Management History database. This considerably increases the time required to back up and restore the database. When you join an additional Administration Service instance to Active Roles replication, this also causes high network traffic during the database replication process.
- The graphical user interface (GUI) is not suitable to represent large volumes of

Management History results. As there are no filtering or paging capabilities, it can be difficult to sort through the results.

Report Pack considerations

To address the limitations of Management History, Active Roles provides different means for change auditing and change-tracking reports, as a part of the Active Roles Report Pack. These reports are designed to help answer the following questions:

- What management tasks were performed on a given object within a certain period of time?
- What management tasks were performed on a given object during the object's entire existence?
- When was a certain attribute of a given object modified?

Change tracking report best practices

Change-tracking reports are based on data collected from event logs. A separate log is stored on each computer running the Administration Service, and each log only contains events generated by one Administration Service instance. Therefore, to use reports, the events from all event logs must be consolidated to form a complete audit trail.

The process of consolidating events, referred to as the data collection process, is performed by a separate Active Roles component, the Collector. With the Collector wizard, you can configure and run data collection jobs, and schedule them to run on a regular basis.

Change tracking report limitations

The main limitation of change-tracking reports is that the information must be collected and consolidated in a separate database before you can build the reports. However, the data collection process also has the following disadvantages:

- Collecting data might be a very lengthy operation and the database size might grow unacceptable when collecting all events that occurred within a long period of time in a large environment.
- Collecting data is impossible over slow WAN links. This limitation is inherent to the Active Roles component intended to collect data for reporting.

Job Server

In certain environments, a dedicated job server might help offset the load from the Active Roles server. Processes such as dynamic group processing and Synchronization Service tasks might use a lot of CPU and memory that would be otherwise used by other Active Roles components. In such cases, using an isolated and dedicated server for those processes can help take the burden off the main Active Roles server.

Example: Using a job server for Active Roles Synchronization Service

This example scenario describes how to set up an Active Roles Synchronization Service server that will perform hourly updates from an HR system (involving a total number of 80,000 users) to Active Roles.

To create a job server for Active Roles Synchronization Service

1. Create a new IP subnet.
2. Install or move a domain controller (DC) to this new subnet.
3. Install Active Roles and Synchronization Service on a new host in this new subnet. The database configuration can either be a new subscriber or it can use an existing database.
4. Prevent Active Roles from publishing its Service Connection Point to ensure no users connect to this instance. For more information, see *Knowledge Base Article 4216122* on the One Identity support portal.
5. Configure this Active Roles instance to only use the DC.
    a. Navigate to **Configuration** > **Server Configuration** > **Administration Services**, then select the server.
    b. Right-click **Properties**.
    c. Select **DirSync Servers** > **Change**.
    d. Select **Only specified Domain Controller** and choose the DC that you installed or moved to the subnet.
6. Configure Synchronization Service to use this Active Roles instance to perform all workflow steps as required.

Configuring Active Roles to handle real-time dynamic group updates

If you have an environment where all dynamic group membership rules are configured to use system-provided Active Directory attribute values, you can create a dedicated Active Roles configuration in the Active Roles Console specifically for handling real-time dynamic group updates.

This has two advantages:

- Dynamic group processing might be resource-intensive. Therefore, using a dedicated configuration for it can free up resources in the primary Active Roles Administration Service configuration for other operations, for example servicing client requests.
- Using a dedicated configuration also hides operation logging for dynamic group processing on the primary Active Roles configuration.

NOTE: Dynamic groups that are scoped to Managed Units (MUs) or other dynamic groups might negatively impact performance. Dynamic groups and MUs are technically search results, and they add additional overhead. For this reason, One Identity does not recommend using MUs or dynamic groups in membership rules.

To create a configuration for handling real-time dynamic group updates

1. To open the Active Roles Console, on the **Apps** page or **Start** menu—depending on the version of your Windows operating system—open **Active Roles 8.2 Console**.
2. In the **Console tree**, navigate to **Configuration** > **Policies** > **Administration** > **Builtin**.
3. In the details pane, right-click **Built-in Policy - Dynamic Groups**, then select **Policy Scope**.
4. To remove scope links, in the **Policy Scope** window, select all links, click **Remove**, then click **OK**.
5. In the details pane, right-click **Built-in Policy - Dynamic Groups**, then select **Properties**.
6. In the **Properties** window, select the **Policies** tab, then select the policy and click **View/Edit**.
7. In the **Policy Properties** window, select the **Policy Settings** tab, and clear the **Receive directory changes from DirSync control** check box. Then, in the pop-up dialog, click **OK**.
8. To apply turning off receiving directory changes from DirSync control, in both **Properties** windows, click **OK**.
9. On the dynamic group job server, do the following:
    a. In the **Console tree**, navigate to **Configuration** > **Server Configuration**.
    b. In the details pane, double-click **Change Tracking Log Configuration**.
    c. Under the **Log Settings** tab, set the value to a low number, such as 1 day.
    d. Configure membership rules for dynamic groups as preferred. For more information, see *Adding a membership rule to a dynamic group* in the *Active Roles Administration Guide*.

Performance bottlenecks

When using Active Roles, the following components and factors can cause performance bottlenecks:

- Scripts
- Internet Information System (IIS). In particular, do not install the WebDAV feature.

  NOTE: For general best practices on optimizing IIS performance, see *Optimizing IIS Performance* in the *Microsoft BizTalk Server documentation*.
- Incorrect virtual memory (pagefile) settings. The virtual memory must be managed by the system.
- Antivirus software. To add exclusions, follow the instructions of *Knowledge Base Article 4216244* in the One Identity support portal.
- Firewall ports. To add exclusions, follow the instructions of *Knowledge Base Article 4227036* in the One Identity support portal.

NOTE: Make sure that the Active Roles service account is not a member of the Active Roles administrator group.

Known performance issues and workarounds

When using Active Roles, consider the following known performance issues and their workarounds:

- In environments with many Active Roles Virtual Attributes, the Active Roles Administration Service host and the Microsoft SQL Server host might experience high CPU utilization and poor performance. This might be related to a known issue that was addressed but is not enabled by default. For more information on resolving this issue, see *Knowledge Base Article 4216183* on the One Identity support portal.
- In environments with many Microsoft Exchange mailboxes, retrieving accounts with Microsoft Exchange attributes might be noticeably slower than retrieving accounts without Microsoft Exchange attributes. For more information on resolving this issue, see *Knowledge Base Article 4336544* on the One Identity support portal.
- The Application Policy Script Module might be incorrectly configured after an upgrade. For more information on resolving this issue, see *Knowledge Base Article 4338971* on the One Identity support portal.
- If you experience performance issues only with the Active Roles Web Interface, then SignalR might be blocked. For more information on resolving this issue, see *Knowledge Base Article 4319280* on the One Identity support portal.

- If you experience performance issues specifically when listing groups or viewing the **memberOf** tab of an Active Directory object, then an expensive Managed Unit might be responsible for the performance impact. For more information on resolving this issue, see *Knowledge Base Article 4371881* on the One Identity support portal.
- In environments where the **Replicating Directory Changes** extended right was not granted to the domain management account, Active Roles Managed Unit rules might break. These broken rules might negatively impact the performance of all Active Roles clients. For more information on resolving this issue, see *Knowledge Base Article 4373208* on the One Identity support portal.

Active Roles Synchronization Service connector performance

If you experience poor connector performance when using Active Roles Synchronization Service, enable multithreading for that connector. For more information, see *Knowledge Base Article 4338969* in the One Identity support portal.

SQL Server considerations

One Identity recommends hosting the Active Roles databases on a dedicated SQL Server instance. Do not share SQL Server resources with SQL-intensive applications (such as Quest Change Auditor), as it will negatively impact the performance of the Active Roles database.

Microsoft SQL is a resource-intensive application, and Microsoft SQL will use all available memory on its host server by default. To avoid Microsoft SQL using up all the available memory and negatively impacting the performance of the host server, limit the amount of RAM available to Microsoft SQL. However, also make sure that a minimum of **4** GB of RAM is always available for it. For more information, see *How to: Set a Fixed Amount of Memory (SQL Server Management Studio)* in the *Microsoft SQL Server documentation*.

For example, if the Windows host has **16** GB of RAM allocated, then a dedicated SQL instance on a dedicated SQL Server host must be limited to **12** GB.

IMPORTANT: When authenticating to Microsoft SQL Server via Windows Authentication, ensure that the necessary Service Principal Names (SPNs) are configured and delegated.

To ensure that there are no errors with Windows Authentication, enable Kerberos logging on the Active Roles Administration Service and Active Roles Web Interface hosts. For more information, see *How to enable Kerberos event logging* in the *Microsoft Windows Server documentation*.

To configure the necessary SQL Server SPNs, see *Manual SPN registration* in the *Microsoft SQL Server documentation*. After you configured the SPNs, add the delegations in the **Delegation** tab of the **Active Directory Users and Computers** application.

Performance issues that are affecting only Active Roles Virtual Attributes

If you experience performance differences among multiple Active Roles clients when working with Active Roles Virtual Attributes—compared to similar operations that only reference system-provided Active Directory attributes—performing maintenance on SQL indexes might help improving performance. For more information, see *Knowledge Base Article 4334101* in the One Identity support portal.

Other Windows host configuration considerations

Consider the following Windows Server settings when using Active Roles:

- By default, Windows Server uses the **Balanced** power plan that balances power consumption and performance. If performance is your primary focus, change the power plan to **High Performance** under **Control Panel** > **All Control Panel Items** > **Power Options**.
- If set too low, the Windows page file settings on the host server might negatively impact performance. Set the minimum page file size to the amount of installed RAM in the host server, and set the maximum size not more than three times the amount of installed RAM. For more information, see *How to determine the appropriate page file size for 64-bit versions of Windows* in the *Microsoft Windows documentation*.

Latency to the environment

Active Roles is heavily impacted by latency on communications to the SQL Server and to the associated global catalog server. Ideally, these resources must be deployed on the same site, with a latency of less than **1** millisecond.

Clearing obsolete cached information

Active Roles caches environmental information in a few different locations. If the environment changes significantly, this cache could become obsolete that can negatively impact performance. This can occur, for example, after:

- Active Directory schema or functional level updates.
- Major SQL Server or Microsoft Exchange version updates.

For more information on clearing all of these caches, see *Knowledge Base Article 4322345* in the One Identity support portal.

Handling obsolete and incompatible external components

External applications, such as Change Auditor, Defender, Recovery Manager, and Safeguard Authentication Services all deploy Active Roles components. If either Active Roles or these external applications are upgraded, older versions of their deployed components could remain in the Active Roles configuration. In such cases, make sure to disable or remove these obsolete components.

Preventing high CPU usage when running the Dynamic Group Updater scheduled task

In environments with a large number of Active Roles dynamic groups, the nightly run of the **Dynamic Group Updater** scheduled task can take several hours, and can also cause high CPU utilization on the target Active Directory domain controller (DC).

To solve this issue, add indexes in Active Directory on the attributes that are being queried by the dynamic group membership rule or rules.

Common SQL bottlenecks

When using Active Roles, consider the following possible SQL bottlenecks:

- Database locks. Too many Active Roles services using the same shared database might cause deadlocks on tables, which will impact performance. In such cases, users could see delays in the Active Roles clients.
- Geographic latency.
- Insufficient resources, such as memory or hard disk.
- Replication. Use SQL Replication Monitor to view and resolve any errors with SQL replication. For assistance with SQL replication, contact Microsoft.

Common memory bottlenecks

When using Active Roles, consider that the following features and components might cause increased memory usage:

- Custom scripting, such as:
  - Expensive scripts
  - Incorrect scopes
- Add-ons, such as:
  - Safeguard Authentication Services
  - Change Auditor
- Managed Units
- Synchronizing Active Directory permissions
- Workflows
- Dynamic groups, such as:
  - Complex rules
  - Large numbers of dynamic groups
- Group families
- Azure integration
- Custom scripting
- Third-party integrations

Hard drive space bottlenecks

Consider the following settings, because they can occupy a large amount of hard drive space on the Active Roles hosts:

- Limited page file size

  One Identity recommends using the **System managed size** setting regardless of the amount of memory installed on the host.

  To change the settings of the page file size

  1. To open the **Run** command window, press **Windows+R**.
  2. In the text box, type `SystemPropertiesAdvanced.exe` and press **Enter**.
  3. Under **Performance**, click **Settings**.

4. In the **Performance Options** window, select the **Advanced** tab.
5. Under **Virtual Memory**, click **Change**.

- Logging

  Enabling Active Roles **Verbose** logging might result in a very large log file. One Identity recommends enabling **Verbose** logging only for troubleshooting purposes. For more information on logging, see *Enabling or disabling diagnostic logs* in the *Active Roles Administration Guide*.

CPU, network, third-party integration and Azure integration bottlenecks

When using Active Roles, consider the following processor, network, and integration-related bottlenecks.

- CPU-intensive tasks, load balancers, proxies and network connections can cause performance bottlenecks across all Active Roles clients.
- For third-party add-ons and integrations, ensure that the installed version is valid for the Active Roles version that you use, especially after upgrading Active Roles.
- The number of Azure objects that you manage can impact overall performance in the Web Interface. Connectivity to Azure might be negatively impacted by load balancers, proxies and the general quality of the network connection.

Troubleshooting

If you encounter any problems, use the following options and key areas for troubleshooting Active Roles.

Troubleshooting in logging

- Windows Event Log: Active Roles Admin Service log.
- Active Roles Configuration Center logging:
  - Administration Service: Click **Log Viewer** > **Open**.
  - ADSI Provider: Use it to log connections such as SPML, remote PowerShell, or remote Active Roles clients.
  - Active Roles Web Interface: Use it on any Active Roles server hosting the Web Interface.
  - Active Roles Console (MMC Interface): Use it to log local Console activity.

Troubleshooting in the Web Interface

- In case of slow performance:
  - Check for connectivity issues between the Web Interface and the Active Roles server if they are on different hosts.
  - Check for defect 395064. For more information, see *Knowledge Base Article 4369863* in the One Identity support portal.
- Check Kerberos constrained delegation. For more information, see *Knowledge Base Article 4336757* in the One Identity support portal.
- In case of potential load balancer issues, consider that load balancers are not in the scope of One Identity Support. To confirm if this issue still persists, bypass the load balancer.

Troubleshooting domain connections

- Check for any latency between the Active Roles host and the domain controller (DC) that Active Roles uses.
- Change the currently active DC to another DC.
- Check the DC Event Logs for any potential issues.

About us

One Identity solutions eliminate the complexities and time-consuming processes often required to govern identities, manage privileged accounts and control access. Our solutions enhance business agility while addressing your IAM challenges with on-premises, cloud and hybrid environments.

Contacting us

For sales and other inquiries, such as licensing, support, and renewals, visit https://www.oneidentity.com/company/contact-us.aspx.

Technical support resources

Technical support is available to One Identity customers with a valid maintenance contract and customers who have trial versions. You can access the Support Portal at https://support.oneidentity.com/.

The Support Portal provides self-help tools you can use to solve problems quickly and independently, 24 hours a day, 365 days a year. The Support Portal enables you to:

- Submit and manage a Service Request
- View Knowledge Base articles
- Sign up for product notifications
- Download software and technical documentation
- View how-to videos at www.YouTube.com/OneIdentity
- Engage in community discussions
- Chat with support engineers online
- View services to assist you with your product