opentext™

# AD Bridge CE 24.3(v3.5)

## Administration Guide

**July 2024**

## Legal Notice

# Contents

# About This Guide

The *AD Bridge Administration Guide* provides information to help you understand, install, configure, and employ the AD Bridge product to help manage your enterprise environment.

## Audience

This guide is written for administrators and users who will use AD Bridge to more effectively manage Active Directory and group policies in a cross-platform environment.

## Feedback

We want to hear your comments and suggestions about this manual and the other documentation included with this product. Please use the User Comments feature at the bottom of each page of the online documentation.

## Additional Documentation

AD Bridge is supported by other documentation (in both PDF and HTML formats) that you can use to learn about and implement the product. For additional documentation, see the AD Bridge documentation website.

# 1 Getting Started

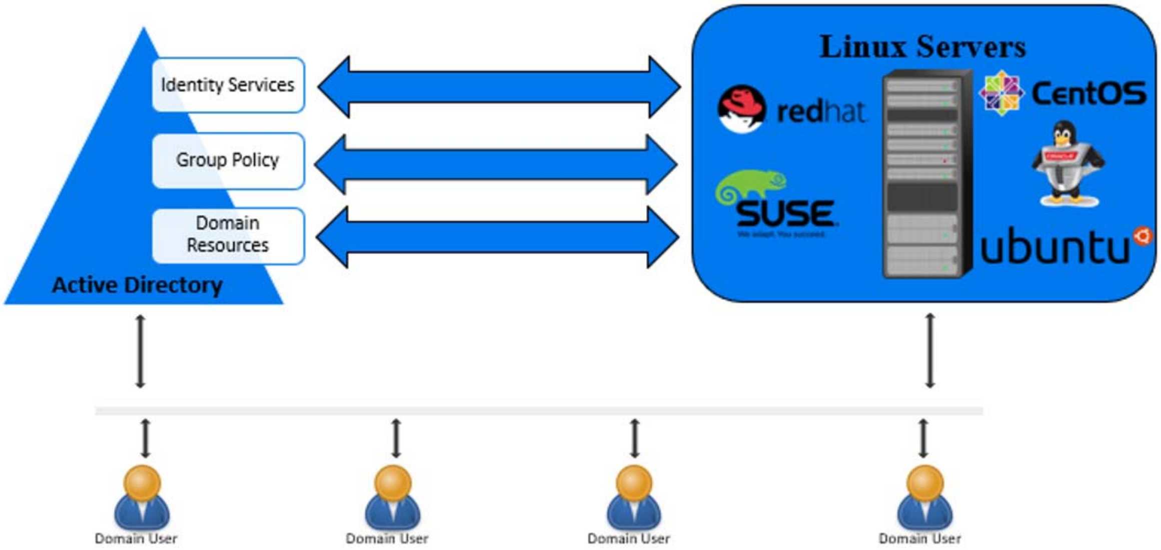
AD Bridge is a solution that extends Active Directory (AD) capabilities by enabling domain controllers to add on premises Linux servers and Linux virtual machines in the Cloud to the AD environment to interface with identity services, group policies, and domain resources. This is accomplished with the installation of an AD Bridge Linux Agent on Linux computers, AD Bridge and Cloud Gateways each and a GPMC snap-in tool “AD Bridge GPEdit Extension” on a workstation which is a domain member.

After the AD Bridge and Cloud Gateways, AD Bridge Linux Agent and GPEdit Extension are installed on their respective computers, you can configure built-in and custom group policies for Linux agents via the Group Policy Management Console on a workstation which is a domain member. and bridge Linux virtual machines (VMs) in the cloud with the AD Bridge Gateway and push universal policies created on the Cloud Gateway to cloud Linux VMs. Some of the capabilities include the following group policy options for agent computers:

- Configure Allow and Deny controls for Firewall settings
- Start, stop, or restart agent services
- Import and manage Open SSH, Sudoers and Custom Configuration files
- Modify and control agent application files
- Execute commands
- Configure Active Directory login controls

For more information about these settings, see the Linux Agent GPO Settings.

Reference the graphic below for a visual depiction of how AD Bridge will work with your Active Directory environment.



AD Bridge thus delivers unique capabilities, that modern organizations need to capitalize on their investments in the Active Directory and Group Policy space, increasing security while reducing risk.

# 2 Installing AD Bridge

This section contains information that will help you understand the following:

## Linux Requirements and Supported Platforms

Review the information in this section before installing the AD Bridge Agent on a Linux server.

### Linux Requirements

Complete the following requirements before you install the Linux Agent and join Active Directory:

- Install the Linux Agent with `root` (requires administrator password)
- DNS name servers on the Linux Agent must list the Active Directory DNS servers
- The Active Directory domain is listed as one of the default search domains
- Download and install prerequisite Linux packages from respective vendors during the Linux Agent installation.

  Otherwise, you must install the following Linux packages prior to running the Linux Agent installation:

| Linux Distribution | Required Linux Packages |
|---|---|
| All distributions | .NetCore system update to install .NetCore 6 and its prerequisites.<br>For more information, see *Linux distribution dependencies*. |

| Linux Distribution | Required Linux Packages |
|---|---|
| RHEL 7,RHEL 8, RHEL 9 | `bpftrace`<br>`kernel-headers`<br>`kernel-devel-$(uname -r)`<br>`aspnetcore-runtime-6.0`<br>`nss.x86_64`<br>`realmd`<br>`oddjob`<br>`oddjob-mkhomedir`<br>`sssd`<br>`adcli`<br>`openldap-clients`<br>`samba-common`<br>`samba-common-tools`<br>`samba-client`<br>`krb5-workstation`<br>`cifs-utils`<br>`python3-policycoreutils` |
|  | `kernel-headers`<br>`kernel-devel`<br>`kernel-devel-$(uname -r)`<br>`scl-utils`<br>`rh-dotnet60`<br>`realmd`<br>`oddjob`<br>`oddjob-mkhomedir`<br>`sssd`<br>`adcli`<br>`openldap-clients`<br>`samba-common`<br>`samba-common-tools`<br>`samba-client`<br>`krb5-workstation`<br>`cifs-utils` |
| SLES 12, SLES 15 | `aspnetcore-runtime-6.0`<br>`samba-client`<br>`sssd`<br>`realmd`<br>`krb5-client`<br>`openldap2-client`<br>`adcli`<br>`sssd-tools`<br>`sssd-ad`<br>`mozilla-nss`<br>`aspnetcore-runtime-6.0`<br>`mozilla-nss`<br>`realmd`<br>`adcli`<br>`sssd`<br>`sssd-tools`<br>`sssd-ad`<br>`samba-client`<br>`krb5-client`<br>`openldap2-client` |

| Linux Distribution | Required Linux Packages |
|---|---|
| Ubuntu 18.04 | `aspnetcore-runtime-6.0`<br>`samba-client`<br>`sssd`<br>`realmd`<br>`krb5-client`<br>`openldap2-client`<br>`adcli`<br>`sssd-tools`<br>`sssd-ad`<br>`mozilla-nss` |
| Ubuntu 20.04 | `apt-transport-https`<br>`aspnetcore-runtime-6.0`<br>`realmd`<br>`sssd`<br>`sssd-tools`<br>`libnss-sss`<br>`libpam-sss`<br>`adcli`<br>`samba-common-bin`<br>`oddjob`<br>`oddjob-mkhomedir`<br>`packagekit`<br>`smbclient`<br>`cifs-utils`<br>`keyutils`<br>`krb5-user` |
| Ubuntu 22.04 | `apt-transport-https`<br>`aspnetcore-runtime-6.0`<br>`realmd`<br>`sssd`<br>`sssd-tools`<br>`libnss-sss`<br>`libpam-sss`<br>`adcli`<br>`samba-common-bin`<br>`oddjob`<br>`oddjob-mkhomedir`<br>`packagekit`<br>`smbclient`<br>`cifs-utils`<br>`keyutils`<br>`krb5-user` |

**NOTE:** If a prerequisite package check or installation fails, the failure notice will identify any missing prerequisites.

## Supported Linux Platforms

AD Bridge 3.5 supports the following Linux platforms:

| Linux Distribution | Version |
|---|---|
| RHEL | ◆ RHEL 9.2<br>◆ RHEL 8.9<br>◆ RHEL 7.9 |
| SLES | ◆ SLES 15 SP4 (SUSE Linux Enterprise Server)<br>◆ SLES 12 SP5 |
| Ubuntu | ◆ Ubuntu 22.04<br>◆ Ubuntu 20.04<br>◆ Ubuntu 18.04 |

**NOTE:** If your cloud environment uses a GoDaddy SSL certificate on RHEL 7, RHEL 8, Ubuntu 18, Ubuntu 20, Ubuntu 22, SLES 12, and SLES 15 operating systems, you must copy it to agent machines and manually assign trust.

For more information, see "Adding a GoDaddy SSL Certificate" on page 16.

# Installing the Cloud Gateway in Microsoft Azure

The AD Bridge Cloud Gateway is used to bridge Linux virtual machines (VMs) in the cloud with the on premises AD Bridge Gateway and push universal policies created on the Cloud Gateway to cloud Linux VMs.

**To set up the AD Bridge Cloud Gateway:**

1. Install the gatekeeper on a Windows Server 2016 or later VM from your chosen cloud provider.
2. Install the gateway on an on-premises VM or a cloud VM with access to a managed Directory Services provider such as Amazon Directory Services or Microsoft Entra DS.
3. If the cloud VM has access to Directory Services, you can install the gateway on the same cloud VM as the gatekeeper.
4. The gatekeeper VM must have ports 443 and 30 exposed to the Internet.
5. You can use a managed database like AWS RDS or Azure SQL, or you can install SQL Server on a cloud VM yourself. Installing SQL Server on the same VM as the gatekeeper is acceptable.
6. Download the AD Bridge installation files from the Open text Downloads website onto a Windows device.
7. Extract the contents of the `ADBRIDGECLOUD_3.5.zip` file.
8. Open the `ServiceConfiguration.Release.cscfg` configuration file available in the extracted contents and modify the highlighted text as shown in the snippet below according to your environment:

```
<?xml version="1.0" encoding="utf-8"?>
<ServiceConfiguration serviceName="HAPI.Mvc.Gatekeeper.CloudService"
xmlns="http://schemas.microsoft.com/ServiceHosting/2008/10/
ServiceConfiguration" osFamily="6" osVersion="*" schemaVersion="2015-
04.2.6">
  <Role name="HAPI.Mvc.Gatekeeper.CloudHost">
    <Instances count="1" />
    <ConfigurationSettings>
      <Setting name="DatabaseConnection"
value="Server=tcp:myserver.database.windows.net,1433;Initial
Catalog=ADBridge;Persist Security Info=True;User
ID=myuser@myserver;Password=" />
      <Setting name="WildcardDomain" value="your domain name.com" />
      <Setting name="LogStorageConnectionString"
value="DefaultEndpointsProtocol=https;AccountName=mystorageaccount;Acc
ountKey=" />
      <Setting name="AzureLogShare" value="ADB" />
      <Setting name="AzureLogDirectory" value="Logs" />
      <Setting name="LoggingLevel" value="Error" />
    </ConfigurationSettings>
    <Certificates>
      <Certificate name="Certificate1" thumbprint="<thumbprint here>"
thumbprintAlgorithm="sha1" />
    </Certificates>
  </Role>
  <Role name="HAPI.Mvc.Gatekeeper.TraversalWorker">
    <Instances count="1" />
    <ConfigurationSettings>
      <Setting name="DatabaseConnection"
value="Server=tcp:myserver.database.windows.net,1433;Initial
Catalog=ADBridge;Persist Security Info=True;User
ID=myuser@myserver;Password=" />
      <Setting name="LogStorageConnectionString"
value="DefaultEndpointsProtocol=https;AccountName=somestorageaccount;A
ccountKey=" />
      <Setting name="AzureLogShare" value="ADB" />
      <Setting name="AzureLogDirectory" value="Logs" />
      <Setting name="LoggingLevel" value="Error" />
    </ConfigurationSettings>
    <Certificates>
      <Certificate name="Certificate1" thumbprint="<thumbprint here>"
thumbprintAlgorithm="sha1" />
    </Certificates>
  </Role>
  <NetworkConfiguration>
    <VirtualNetworkSite name="Group resource group virtual network" />
    <AddressAssignments>
      <InstanceAddress roleName="HAPI.Mvc.Gatekeeper.CloudHost">
        <Subnets>
          <Subnet name="subnet name" />
        </Subnets>
      </InstanceAddress>
      <InstanceAddress roleName="HAPI.Mvc.Gatekeeper.TraversalWorker">
        <Subnets>
          <Subnet name="subnet name" />
        </Subnets>
      </InstanceAddress>
    </AddressAssignments>
  </NetworkConfiguration>
</ServiceConfiguration>
```

**9** Configure your Linux VM:

**9a** Install NGINX.

```
# yum update

# reboot

# yum install epel-release

# yum install nginx
```

**9b** Install your SSL certificate on the NGINX server.

**9c** Copy the `cors.include` file from extracted contents to the`/etc/nginx` directory of the NGINX server.

**9d** Copy the `nginx.conf` file from extracted contents to the`/etc/nginx` directory of the NGINX server and replace the existing version of the file.

**9e** Configure the Azure firewall to allow HTTPS (port 443) traffic to the NGINX server.

**10** Open the `/etc/nginx/nginx.conf` file and modify the highlighted text as shown in the snippet below according to your environment:

```
user nginx;
worker_processes auto;
error_log /var/log/nginx/error.log;
pid /run/nginx.pid;

# Load dynamic modules. See /usr/share/nginx/README.dynamic.
include /usr/share/nginx/modules/*.conf;

events {
    worker_connections 1024;
}

http {
    log_format  main  '$remote_addr - $remote_user [$time_local]
"$request" '
                      '$status $body_bytes_sent "$http_referer" '
                      '"$http_user_agent" "$http_x_forwarded_for"';

    access_log  /var/log/nginx/access.log  main;

    sendfile            on;
    tcp_nopush          on;
    tcp_nodelay         on;
    keepalive_timeout   65;
    types_hash_max_size 2048;

    include             /etc/nginx/mime.types;
    default_type        application/octet-stream;

    server {
        listen       80 default_server;
        listen       [::]:80 default_server;

        server_name _;
        return 301 https://$host$request_uri;
    }

    server {
        listen       443 ssl http2 default_server;
        listen       [::]:443 ssl http2 default_server;
        server_name  ~$(?<subdomain>\.)?(?<domain>.+)$;
        root         /usr/share/nginx/html;

        #replace with your certificate in the next two lines
        ssl_certificate "/etc/pki/nginx/cert-here.crt";
        ssl_certificate_key "/etc/pki/nginx/cert-here.pem";
ssl_protocols TLSv1.2;
        location ~*
"^/(api|portal|content|scripts|images|swagger)" {
            gzip on;
            gzip_proxied any;
            gzip_types text/html application/json
application/javascript text/xml;
            proxy_redirect off;
            proxy_set_header host $host;
            proxy_set_header X-forward-for $proxy_add_x_forwarded_for;
            proxy_set_header X-real-ip $remote_addr;
            include cors.include;
            rewrite ^/(.*) /$1 break;
            proxy_connect_timeout 300;
            proxy_send_timeout 300;
            proxy_read_timeout 300;
            send_timeout 300;
            proxy_pass http://10.1.0.4;#replace with IP of your Cloud
Host role

}

   location /ws {
            proxy_http_version 1.1;
            proxy_set_header Upgrade $http_upgrade;
            proxy_set_header Connection "Upgrade";
            proxy_redirect off;
            proxy_set_header host $host;
            proxy_set_header X-forward-for $proxy_add_x_forwarded_for;
            proxy_set_header X-real-ip $remote_addr;
            include cors.include;
            proxy_pass https://10.1.0.5/ws; #replace with IP of your
TraversalWorker role

        }

        location /route {
            proxy_connect_timeout 300;
            proxy_send_timeout 300;
            proxy_read_timeout 300;
            send_timeout 300;
            gzip on;
            gzip_proxied any;
          gzip_types text/html application/json application/javascript
text/xml;
            proxy_redirect off;
            proxy_set_header host $host;
            proxy_set_header X-forward-for $proxy_add_x_forwarded_for;
            proxy_set_header X-real-ip $remote_addr;
            proxy_set_header If-Modified-Since "";
            add_header 'Cache-Control' 'no-store, no-cache, must-
revalidate, proxy-revalidate, max-age=0';
            expires off;
            include cors.include;

           proxy_pass https://10.1.0.5/route; #replace with IP of your
TraversalWorker role

        }
    }
}
```

**11** Copy compressed files of the web console to the Linux VM and run the following commands:

**11a** Remove the old HTML files # `rm -rf /usr/share/nginx/html/*`

**11b** Extract the web console files # `tar -jxvf WebConsole_22472.tar.bz2 -C /usr/share/nginx/html/`.

**11c** Restart the nginx service# `systemctl restart nginx`.

## Adding a GoDaddy SSL Certificate

To add a GoDaddy SSL certificate, you must download the certificate, copy to the necessary agent machine and manually assign trust to the certificate:

### Prerequisite

Download the gdig2.crt.pem certificate from the GoDaddy Repository.

**For RHEL 7:**

**1** Copy the gdig2.crt.file and ca-certificates.crt to `/etc/pki/tls/certs`.

2. Type `ln -s /etc/pki/tls/certs/gdig2.crt.pem /etc/pki/tls/certs/27eb7704.0` and press Enter.
3. Type `certutil -d sql:/etc/pki/nssdb -A -t "C,C,C" -n "Go Daddy Secure Certificate Authority - G2" -i /etc/pki/tls/certs/gdig2.crt.pem` and press Enter.

**For RHEL 8 and RHEL 9:**

1. Copy the `Go Daddy Secure Certificate Authority - G2.crt and ca-certificates` files to `/usr/share/pki/ca-trust-source/anchors`.
2. Type `update-ca-trust` and press Enter.

**For SLES 12 and SLES 15:**

1. Copy the Go Daddy Secure Certificate Authority - G2.crt and ca-certificates files to `/etc/pki/trust/anchors/`.
2. Type `update-ca-certificates` and press Enter.
3. Restart the agent.

**For Ubuntu 22, 20 and 18:**

1. Copy the `certificate.crt` and ca-certificates files to `/usr/local/share/ca-certificates/certificate.crt`.
2. Type `dpkg-reconfigure ca-certificates` and press Enter.

# Installing the AD Bridge Gatekeeper and Gateway

The AD Bridge Gateway is used to push policies from Active Directory to the Cloud Gateway.

Complete the following prerequisites before you install the AD Bridge Gateway:

- Microsoft Server 2016 or later installed
- Domain Administrator account access

The AD Bridge Gateway installer also installs: Microsoft .Net Framework 4.8.

**To install the Universal Policy Administrator On Premises Gatekeeper and Gateway:**

**NOTE:** When you execute the AD `Bridge Gateway.exe` file, it installs both Gatekeeper and Gateway On premises.

1. Download the Universal Policy Administrator On Premises Gateway installer file AD `Bridge Gateway.exe` from the Opentext Downloads website.
2. Execute the downloaded AD Bridge Gateway.exe file.
3. When the installation wizard opens, select both Install Gatekeeper and Install Gateway options and click **Install**.

   If .NET Framework 4.8.x is not already installed on the server, it is installed as part of the prerequisite check before the Universal Policy Administrator On Premises Gateway installation starts.

4 Click **Next** when the AD Bridge Gatekeeper Setup wizard opens.

5 Read and Accept the License Agreement, and click **Next**.

6 Browse your system to select a certificate in the .pfx file format, specify the password, and click **Next**.

7 Specify the connection string in the Gatekeeper Configuration wizard, and click **Next**.

8 Select the destination folder for the installation files and click **Next**.

9 Click **Install** to copy the Gatekeeper files.

10 Click **Finish** to complete the Gatekeeper setup.

11 **NOTE:** The Gateway installation automatically starts.

Click **Next** when the Universal Policy Administrator On Premises Gateway setup wizard opens.

12 Read and Accept the License Agreement, and click **Next**.

13 Select an installation option. The available options are:

- NAT Traversal
- DMZ or Port Forward

**NOTE:** In most cases, select **NAT Traversal**.

14 Click **Next**.

15 Enter domain administrator credentials and click **Next**.

16 Enter the Cloud Gateway URL and Universal Policy Administrator On Premises Gateway owner account credentials, and click **Next**.

**NOTE:** Click **Register** and create a new account if one does not exist.

17 Retain or change the default location for program installation, and then click **Next**.

18 Click **Install** to copy the Gateway installer files.

19 Click **Finish** on the last screen of the wizard to complete the installation.

## Configuring the AD Bridge Syslog Provider

You can configure AD Bridge 3.5 to forward events and syslog messages to one or more SIEM solutions.

**To configure the AD Bridge Syslog Provider:**

1 Open the `C:\Program Files\OpenText\AD Bridge\Gateway\WebApp\Web.Config` file.

2 Modify the highlighted text as shown in the snippet below according to your environment:

```
<syslogSettings CEFVendor="Opentext" CEFProduct="AD Bridge"
CEFVersion="3.5">
    <Forwarders>
      <add host="localhost" port="514" senderType="UDP"
rfcType="Rfc5242" filterType="None" />
    </Forwarders>
  </syslogSettings>
```

The available options for each of these attributes are:

- **senderType:** The default value is UDP.
  - TCP
  - UDP
- **rfcType:** The default value is Rfc5242.
  - `Rfc5242`
  - `Rfc3164`
- **filterType:** The default value is None.
  - `SyslogOnly`
  - `AuditOnly`
  - None

  **NOTE:** AD Bridge 3.5 only supports the filterType attribute value, `AuditOnly`.

3 Set `CEFVendor`, `CEFProduct`, and `CEFVersion` to values of your choice.

**NOTE:** You can specify multiple forwarders in the same `Web.Config` file.

# Installing the AD Bridge GPEdit Extension

The native AD Group Policy Management Console (GPMC) manages Group policies for AD Bridge Linux Agents joined to Active Directory, with the AD Bridge GPEdit Extension snap-in installed on the domain workstation.

The snap-in adds extensions for Linux-based settings that you can configure within the structure of a Group Policy Object (GPO).

Complete the following prerequisites before installing the GPEdit Extension snap-in:

- Install GPMC on the domain workstation.
- Access to the domain administrator account
- Install Microsoft .NET Framework 4.8.x (*Can be installed by the snap-in installer*)

**To install the AD Bridge GPEdit Extension:**

1. Log in to a domain-joined member server with GPMC installed, as an administrator.
2. Execute the downloaded `ADB_GPEdit_Extension_3_5.exe` file.
3. When the installation wizard opens, click **Change** to change the default copy location for installation files; otherwise, click **Install**.

If .NET Framework 4.8.x is not already installed on your Domain workstation, it is installed as part of the prerequisite check before the snap-in installation starts.

4. Click **Next**.
5. Read and Accept the License Agreement, and click **Next**.
6. Retain or change the default location for program installation, and then click **Next**.
7. Click **Install** to copy the GPEdit extension files.
8. Click **Finish** on the last screen of the wizard to complete the installation.

You can now configure GPOs for your Linux endpoints that have the AD Bridge Linux Agent installed. For information about configuring GPOs, see Managing Linux GPO Settings.

# Installing the AD Bridge Linux Agent

When you download the AD Bridge Linux Agent installer, you will need to unpack the installer for your specific Linux distribution. An example of the files included with the final distribution installer is shown below:

- `adb-agent-rh7-2.0.rpm`
- `install.sh`
- `uninstall.sh`

**NOTE:** The AD Bridge Linux Agent installer also installs .Net Core 6, which is necessary during uninstallation.

**To install the AD Bridge Linux Agent on a Linux machine:**

1. Copy the Linux Agent installer file applicable to your distribution onto the Linux machine.

| Installer file | Linux distribution |
|---|---|
| `ADB_3_5_LinuxAgents.tar.gz` | ◆ RHEL 7,8 and 9<br>◆ SLES 12 and 15 |
| `ADB_3_5_UbuntuAgents.tar.gz` | ◆ Ubuntu 22<br>◆ Ubuntu 20<br>◆ Ubuntu 18 |

2. On the command line, log in as the `root` user and type the following command to unpack the applicable installation package from the table above: `tar xvzf <file name>`.
3. For all distributions except Ubuntu, type the command again using the file name specific to your platform from the table below.

   For example: `tar xvf <file name>`

| Installer file | Linux distribution |
|---|---|
| `RHEL9.tar` | ◆ RHEL 9 |
| `RHEL8.tar` | ◆ RHEL 8 |
| `RHEL7.tar` | ◆ RHEL 7 |
| `Ubuntu22.tar` | Ubuntu 22 |
| `Ubuntu20.tar` | Ubuntu 20 |
| `Ubuntu18.tar` | Ubuntu 18 |
| `SLES15.tar` | SLES 15 |
| `SLES12.tar` | SLES 12 |

4. Copy the Linux Agent installer file applicable to your distribution onto the Linux machine.
5. On the command line, log in as the `root` user and type the following command to unpack the applicable installation package from the table above: `tar xvzf <file name>`.
6. Verify the installer files are on the machine with a list command: `# ls`.
7. Run the `install.sh` script file as `root` to set up the Linux Agent. For example:
   - `# ./install.sh`
   - `#bash install.sh`

   Available agent configuration types are:

   ```
   (a) - Join the Agent to Active Directory
   (g) - Join the agent to the Cloud Gateway Only
   (h) - Join the agent to the Cloud Gateway, and create an AD object for
     this computer (Hybrid Mode)
   (n) - Don't join the agent to anything
   ```

   Installation time varies depending on your environment and prerequisites that need installation. Warning messages during the installation are informational and do not necessarily require action unless you experience an installation failure.

**IMPORTANT:** For SUSE installations, you may receive a confirmation prompt `y/n` before the installation starts. For SUSE 15 installations, the `dotnet-runtime-6` installation displays a problem dependency for `libicu52-1`.

Enter 2 to ignore the dependency and enter y when prompted to install “NEW packages.”

**IMPORTANT:** When you encounter a realm error when attempting to install or uninstall the adjunct agent on Ubuntu 22 server, install or uninstall the adjunct agent, use the respective script with sudo privileges: 'sudo ./install.sh' for installation and 'sudo ./uninstall.sh' for uninstallation.

**IMPORTANT:** When the install agent with AD Join for SLES OS fails, insert the respective SLES ISO image and run **zypper  install realmd adcli sssd sssd-tools sssd-ad samba-client krb5-client openldap2-client** and **zypper refresh && zypper update** commands. Once the packages are installed then run the SLES Agent Installer.

8 (Optional) Enter a, g, h, or n when prompted to join Active Directory.

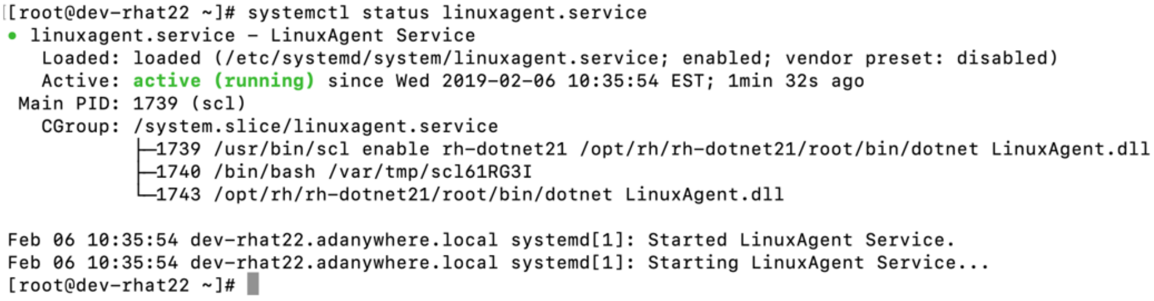
**NOTE:** This step and the following step are optional if you want to join agent configuration type at a later time. For information about joining Agent Configuration Type after installation, see Joining Agent Configuration Type Post Installation.

9 (Optional) When prompted, provide the full domain name, the AD account with rights to join a domain, and AD account password. For example:

```
myCompany.local
administrator
<password>
```

**NOTE:** A fully qualified domain name (FQDN) is only required to join the agent to Active Directory.

During the installation, the Linux Agent is added by default to the Computers OU in Active Directory. After the installation is complete, the Linux Agent service runs on the Linux system, as demonstrated in the example below of an installation on a Red Hat distribution.



```
[root@dev-rhat22 ~]# systemctl status linuxagent.service
● linuxagent.service - LinuxAgent Service
   Loaded: loaded (/etc/systemd/system/linuxagent.service; enabled; vendor preset: disabled)
   Active: active (running) since Wed 2019-02-06 10:35:54 EST; 1min 32s ago
 Main PID: 1739 (scl)
   CGroup: /system.slice/linuxagent.service
           ├─1739 /usr/bin/scl enable rh-dotnet21 /opt/rh/rh-dotnet21/root/bin/dotnet LinuxAgent.dll
           ├─1740 /bin/bash /var/tmp/scl61RG3I
           └─1743 /opt/rh/rh-dotnet21/root/bin/dotnet LinuxAgent.dll

Feb 06 10:35:54 dev-rhat22.adanywhere.local systemd[1]: Started LinuxAgent Service.
Feb 06 10:35:54 dev-rhat22.adanywhere.local systemd[1]: Starting LinuxAgent Service...
[root@dev-rhat22 ~]# █
```

**NOTE:** For information about how to start the Linux Agent Service or verify it is running, see Linux Agent Commands and Lookups.

# Licensing the Linux Agent

The AD Bridge Linux Agent installation comes with a built-in 30-day evaluation period. To continue using AD Bridge after 30 days, purchase the product and install the license before 30 days elapse. For more information, contact Opentext Sales.

To download this product, go to the Opentext Downloads or Customer Center website.

**To activate the AD Bridge license subscription:**

1 Copy the license XML file into the Linux Agent directory `/opt/adb-agent.`

2 Restart the Linux Agent Service.

For more information, see Start the Linux Agent Service.

Restarting the service replaces the temporary license with the new active license and enables all AD Bridge functionality.

# Installing the AD Bridge Windows Agent

The AD Bridge Windows Agent allows you to manage non domain join Windows servers with group policy settings. You configure these settings in the WebUI, when installed in Gatekeeper mode or native group policy tools in Hybrid mode.

Complete the following prerequisites before you install the AD Bridge Windows Agent:

- Microsoft Windows Server 2016 or later, installed and running.
- Domain Administrator Account.

The AD Bridge Windows Agent installer also installs: Microsoft .NET Framework 4.8

**To install the AD Bridge Windows Agent:**

1. Log in to a non-domain joined Microsoft Windows Server 2016 or later as a local administrator.
2. Download the AD Bridge Windows Agent installer file `HAPI Agent.exe` from the Opentext Downloads website and copy onto the non-domain joined Windows Server.
3. Execute the downloaded `HAPI Agent.exe` file.
4. (Optional) Click **Options** to change the location of the installation other than default.
5. Click **Next** when the AD Bridge Windows Agent setup wizard opens.
6. Read and Accept the License Agreement, and click **Next**.
7. Enter domain administrator credentials and click **Next**.

   **NOTE:** The Cloud Gateway URL is pre-populated.

8. Retain or change the default location for program installation, and then click **Next**.
9. Click **Install** to begin copying files.

   **NOTE:** If .NET Framework 4.8 is not already installed on your Domain Controller, it is installed as a part of the prerequisite check, before the AD Bridge Windows Agent installation starts.

10. Click **Finish** on the last screen of the wizard to complete the installation.

# Installing the AD Bridge MAC Agent

The AD Bridge MAC Agent allows you to manage non-domain joined MAC computers with group policies configured in the WebUI and installed in Gatekeeper mode or native group policy tools in Hybrid mode.

Complete the following prerequisites before you install the AD Bridge MAC Agent:

- macOS x or later, installed and running.
- Domain Administrator Account.

The AD Bridge Windows Agent installer also installs: Microsoft .NET Framework 4.8.x.

**To install the AD Bridge MAC Agent:**

1. Log in to a non-domain joined Mac computer as a local administrator.
2. Download the AD Bridge MAC Agent installer file from the Opentext Downloads website and copy onto the non-domain joined Mac computer.
3. Execute the downloaded installer file.
4. (Optional) Click **Change Install Location** to copy the install files to a location other than default.
5. Click **Continue** when the AD Bridge MAC Agent setup wizard opens.
6. Click **Install** to begin copying files.
7. Enter the local macOS password if prompted to start `Terminal` and proceed with installation.

   **NOTE:** Microsoft .NET Framework 4.8.x is installed as part of the prerequisite check, if not installed already and before the AD Bridge MAC Agent installation starts.

8. Choose an agent configuration type. The available options are:

```
(a) - Join the Agent to Active Directory
(g) - Join the agent to the Cloud Gateway Only
(h) - Join the agent to the Cloud Gateway, and create an AD object for
  this computer (Hybrid Mode)
(n) - Don't join the agent to anything
```

   **NOTE:** Installation time varies depending on your environment and prerequisites that need installation. Warning messages during the installation are informational and do not necessarily require action unless you experience an installation failure.

9. (Optional) Enter a, g, h, or n when prompted to join Active Directory.

   **NOTE:** This step and the following step are optional if you want to join agent configuration type at a later time. For information about joining Agent Configuration Type after installation, see Joining Agent Configuration Type Post Installation.

# Joining Agent Configuration Type Post Installation

If you did not join your Linux computer to Active Directory or Cloud Gateway in Gateway Only or Hybrid mode when installing the AD Bridge Linux Agent, follow these instructions on the Linux Agent at a later time:

1. Open the Linux Terminal and locate the `agent` directory. For example:

   `cd /opt/adb-agent.`

2. Type respective commands for given agent configuration types:
   - **Active Directory:** `dotnet LinuxJoinAD.dll <full domain name> <AD Admin account name> [distinguished name of the computer OU]`

     For example: `dotnet LinuxJoinAD.dll myCompany.com administrator`.

**NOTE:** The Linux server is on a corporate network and you choose to join Active Directory for management with native AD tools and GPOs.

- **Cloud Gateway:** `dotnet CloudLinuxJoin.dll <gatekeeperServer[:port]> <traversalServer[:port]> <domainUser>`

  **NOTE:** The Linux server is in the cloud (outside the corporate network) and does not have a computer object in Active Directory. You can manage this Linux server only from the AD Bridge 3.5 web console using Universal Policies.

- **Hybrid Mode:** `dotnet CloudLinuxJoin.dll <gatekeeperServer[:port]> <traversalServer[:port]> <domainUser> [-create-ad-object]`

  **NOTE:** The Linux server is in the cloud (outside the corporate network) and will have a computer object in Active Directory linked to the AD Bridge 3.5 Secure Gateway. Choose this option to manage your cloud Linux server with native Active Directory tools and GPOs.

3 Enter the AD account password when prompted.

**NOTE:** You can also choose to join a specified OU of Active Directory.

## Configuring AD Bridge to use a Third-Party Identity Provider

## Install the AD Bridge Gatekeeper and Gateway

Install the ADB Gatekeeper and Gateway. For more information, see Chapter 2, "Installing AD Bridge," on page 9.

## Configure the AD Bridge application in the identity provider

1. Create the Application for AD Bridge in the identity provider's console.
2. If the identity provider requires it, assign or grant users and groups permission to use the application.
3. Configure the authentication settings in the identity provider application.

### SAML Authentication Settings

1. If the identity provider allows for importing SAML metadata, import the AD Bridge SAML metadata into the identity provider Application or Integration.
2. The AD Bridge SAML metadata is available at (https://<gatekeeper>/Portal/SSO/GetSPMetadata) or by clicking the **Get SAML Metadata** link in the SSO page of the AD Bridge Owner Portal (https://<gatekeeper>/Portal/Account).
3. If the identity provider does not provide an option to import a metadata XML file, use the following values:
   - Entity ID: https://<gatekeeper>
   - Single Signon (SSO) URL: (https://<gatekeeper>/Portal/SSO/SamlACS)
   - Name ID Format: EmailAddress (recommended)
   - Single Logout (SLO) URL: (https://<gatekeeper>/Portal/SSO/SLO)

### Configuring Relay State

Choose a provider name for the SAML connection. Provider name is used when configuring the SAML connection in AD Bridge. The connection name should consist of only alphanumeric characters. Set the SAML Relay State parameter to the provider name.

### Federation Metadata

Download the federation metadata from the identity provider. You will need this metadata to configure AD Bridge in the next step.

### OIDC Authentication Settings

1. Set the Redirect URI to: (https://<gatekeeper>/Portal/SSO/OIDC)
2. Set the logout URI to: (https://<gatekeeper>/Portal/SSO/Logout)
3. Make a note of the Client ID 'OpenID Connect metadata document URL'
4. Set claim type to token.

# Configure AD Bridge to use SAML or OIDC Authentication

- Sign in to the Owner portal (https://<gatekeeper>/Portal/Account) using the Owner account created during the Gatekeeper installation.
- Click the **SSO** button.

## AD Bridge SAML Authentication Settings

1. Click the **Add SAML Provider** button.
2. Specify the provider name (the same name used in the Relay State)
3. Set the Tenancy ID to 1.

**IsDefault:** Use this provider as the default identity provider. If IsDefault is checked, the ADB web console will use this provider for logins. If IsDefault is not checked, to log in to the ADB web console using this provider, you will need to use this URL: https://<gatekeeper>/Portal/SSO/SamlLogin?provider=<ProviderName>.

**NameIdFormat:** The format of the SAML NameID. This value should match the value configured on the identity provider. In most cases, EmailAddress is the recommended value.

**SignatureAlgorithm:** The encryption algorithm used to sign SAML requests and responses. This setting should match the configuration of the identity provider. The recommended setting is SHA_256.

## Provisioning Mode

The provisioning mode determines how users and groups are imported or provisioned into AD Bridge.

- **Automatic provisioning:** The identity provider's provisioning service makes calls to the AD Bridge SCIM endpoint to provision users or groups.
- **SCIM connector:** AD Bridge queries the identity provider's SCIM endpoint to retrieve user or group information.
- **Match to AD account:** In scenarios where there is a local Active Directory with user accounts synchronized with the identity provider, the SAML-authenticated user will be matched to an existing Active Directory user. In this scenario, AD Bridge permissions can be delegated to the Active Directory users and groups.
- **Just In Time provisioning:** In this model, the customer adds a custom attribute to the user accounts, specifying the name of the AD Bridge role assignment to which the user should be added. This value is then sent as a claim during login. When the user logs on, the user account is created in AD Bridge and added to the specified role assignment.
- **Manual provisioning:** If the identity provider does not support automatic provisioning, the customer can use a PowerShell script to create the user and group accounts.

## SAML Claims Mapping

Specify the names of the SAML claims that correspond to the user properties:

- **Require signed requests:** This setting causes all SAML requests, including logout requests, to be signed.
- **Logout URL:** If the identity provider provides a URL for single sign-out, specify it here. This setting overrides the Single Sign-out (SSO) endpoint specified in the SAML metadata.

## AD Bridge OIDC Authentication Settings

To use OIDC authentication:

1. Click the **“Add OIDC Provider”** button
2. Specify the provider name.
3. Set the Tenancy ID to 1.

**IsDefault:** Use this provider as the default identity provider. If IsDefault is checked, the ADB web console will use this provider for logins. If IsDefault is not checked, to log in to the ADB web console using this provider, use the following URL: https://<gatekeeper>/Portal/SSO/OIDCLogin?provider=<ProviderName>.

**Config URL:** The OpenID Connect metadata URL provided by the Identity Provider.

**Identity Claim:** The name of the OpenID Connect claim that contains the identity of the user. (Refer to the identity provider’s documentation for details).

## Additional Parameters

If the identity provider requires additional information to be sent with the request (such as a tenancy id, you can add it in the Additional Parameters.

# Configure Provisioning

Before external users can log in to the ADB console, they must be provisioned or imported into the ADB database. ADB provides two methods for provisioning users.

## Automatic Provisioning

Automatic Provisioning requires the identity provider to support SCIM provisioning. In this scenario, the identity provider’s provisioning service makes SCIM calls to the ADB SCIM service to provision users and groups.

To configure automatic provisioning, you will need to configure the following settings in the identity provider’s provisioning settings:

- **Scim Endpoint:** https://<gatekeeper>/api/scim
- **Authentication or Secret Token:**
  - Navigate to the SSO page in the ADB owner portal.
  - Click **Edit** for the identity provider.

- Click **Configure Provisioning**
- On the Configure Provisioning page, click Get SCIM Token.

### Scim Connector

If the identity provider does not provide a SCIM provisioning service but exposes a SCIM endpoint, you can use the ADB SCIM Connector to import users and groups. The ADB SCIM connector queries the identity provider’s SCIM endpoint to provision users and groups.

Configure the ADB SCIM Connector with the following settings:

- **Server URL:** The server name portion of the Identity Provider’s SCIM endpoint (e.g., https://server.domain.com)
- **Base URL:** The relative URL to the SCIM endpoint on the identity provider (e.g., "/scim/v2").
- **AuthToken:** The authentication token (client secret) provided by the identity provider for SCIM access.
- **Import Users:** Indicates whether user information should be imported.
- I**mport Groups:** Indicates whether group information should be imported.
- **Refresh Interval:** The interval, in minutes, at which the ADB SCIM Connector should query the identity provider for changes to users and groups.

## Assigning the ADB global Administrator Role to a User

- Allow time for the initial provisioning cycle to complete. (If using automatic provisioning, you can check the provisioning status in the Identity Provider’s portal)
- Once the initial provisioning cycle is complete, go to the ADB Owner Portal
- Navigate to the SSO page
- Select the identity provider, and click **List Users**
- Examine the list of users to verify the imported data
- Select a user from the dropdown list and click **Set User as Global Admin**.

This user will now be able to log in to ADB at https://<gatekeeper>. On the Administration tab of the ADB web portal, this user can delegate ADB permissions to other users and groups as desired.

## Configuring AD Bridge Agent Login to use SAML/OIDC Login

For cloud/hybrid Windows or Linux agents, this feature enables users to log in to the device using their AD credentials. If SAML/OIDC login is configured, they can also use their SAML/OIDC credentials.

Since SAML/OIDC authentication requires users to authenticate directly with the identity provider, the login interface will display a URL: https://<gatekeeper>/Portal/SSO/OOB?id=<requestId>.

Users must visit this URL, which will redirect them to the identity provider to complete the login process. Afterward, a Passcode will be displayed. Users must then enter this passcode in the login interface on the client machine to complete the login.

## Windows Agent

To log in using the AD Bridge Agent Login feature, select 'AD Bridge Login' on the login screen. By default, the AD Bridge Agent Login feature will use the default identity provider for the gatekeeper (so if SAML/OIDC is selected as default, SAML/OIDC will be used). To allow login using a non-default provider, set AllowMultipleProviders=1.

The Windows Agent Login feature includes two components: HAPIAUTH, a custom LSA authentication package, which performs the login operations, and HAPICredentialProvider, a custom credential provider, which provides the UI displayed for the AD Bridge Login. The settings for both these components are stored under the registry key HKLM\Software\OpenText\HAPIAUTH.

The following settings can be configured:

- GatekeeperUrl (REG_SZ): The URL of the HAPI gatekeeper.
- LogPath (REG_SZ): The path for the HAPIAuth log file (C:\ProgramData\OpenText\Logs\HapiAuth.log).
- LogLevel (REG_DWORD) (1=Debug, 2=Info, 3=Warning, 4=Error, 5=Critical): Determines the minimum severity of events to write to the log file.
- EventLogLevel (REG_DWORD) (1=Debug, 2=Info, 3=Warning, 4=Error, 5=Critical): Determines the minimum severity of events to write to the event log.
- AllowMultipleProviders (REG_DWORD) (0=Disabled, 1=Enabled): If enabled, AD Bridge Login will display a dropdown list of identity providers (including SAML/OIDC and AD), and the user can select which provider to use for login.
- ShowQRCode (REG_DWORD) (0=Disabled, 1=Enabled): If enabled, AD Bridge will display a link that will open a window containing a QR code. The QR code represents the login URL that the user must visit to complete the login.

**NOTE:** Only administrators can read the HAPIAuth.log. To view the HAPIAuth.log, use "Run as Administrator.

## Linux Agent

During installation, the settings are configured to use the default identity provider. To switch to a different identity provider, update the DomainName and DomainSid properties in /etc/nss_hapi.conf.

**The following settings can be configured:**

1. **GatekeeperUrl:** The URL of the gatekeeper.
2. **GenerateUids:** (yes/no) Generates UID numbers for external users. If set to no, only user accounts that have a value specified in the uidNumber property are allowed to log in.
3. **UidBase:** (default=10000) - The starting number for Generated UIDs.
4. **DomainName:** The name of the Active Directory domain or SAML/OIDC provider to use for authentication.
5. **DomainSid:** For AD domains, the Domain SID. For SAML/OIDC providers, this should be set to TenancyId_ProviderId (in the HAPI Owner portal, select the identity provider, and click List Users – the DomainSid will be the first half of the unique ID for each user).

The Linux Agent Login feature includes a PAM module and NSS module. The settings for these modules are defined in /etc/nss_hapi.conf and can be configured via Universal Policy using the Linux/AD Logins/Cloud/Custom settings.

## Configuring SAML Authentication with Microsoft ENTRA

1. Install the ADB Gatekeeper and Gateway.
2. Create and configure an ENTRA Enterprise Application for ADB:
    - In the ENTRA console, navigate to Enterprise Applications, and select **Create a new Application**.
        - Give the application a name, and select “Integrate any other application you don't find in the gallery (Non-gallery).
        - Click Create.
    - In the ENTRA console, assign Users and Groups to the Enterprise Application.
    - Configure the Enterprise Application to use SAML authentication.
    - In the ENTRA Enterprise Application Settings, go to Single Sign On, and select **SAML**.
    - Download the HAPI SAML metadata:
        - In the ADB owner portal (https://<gatekeeper>/portal/account) (https://<gatekeeper>/portal/account), select **SSO**.
        - Click **Get SAML Metadata**.
        - Save the metadata to a file.
    - In the ENTRA Application SAML settings,
        - Select **Upload metadata file** and upload the **HAPI SAML metadata**.
        - Under “Relay State” specify a domain name for ADB to use for the ENTRA users and groups (for example “ENTRA” or “MYDOMAIN”).
        - (Optional) Set Sign on URL to (https://<gatekeeper>/Portal/SSO/SamlLogin?provider=<ProviderName>) where ProviderName is the name of the SAML provider in ADB – the name you specified for RelayState.
        - Select **Download Federation Metadata** XML and save the metadata to a file.
3. Configuring ADB to use SAML authentication:
    - In the ADB Owner Portal, navigate to SSO settings and click **Add SAML Provider**.
    - Set the provider name to the same value used for Relay State.
    - Check the IsDefault checkbox. This will set the ADB web console to use this SAML provider as the default identity provider for logins. To login with a non-default SAML provider, go to https://<gatekeeper>/Portal/SSO/SamlLogin?provider=<ProviderName>.
    - Set NameIdFormat to Email Address.
    - Set SignatureAlgorithm to SHA_256.
    - Set provisioning mode to AutomaticProvisioning.
    - Set the following values for SAML claims:

      DisplayName: (http://schemas.xmlsoap.org/ws/2005/05/identity/claims/displayname)

      Email: (http://schemas.xmlsoap.org/ws/2005/05/identity/claims/name)

  Unique ID: (http://schemas.xmlsoap.org/ws/2005/05/identity/claims/name)

  - Save the changes.

4. Setting up SCIM provisioning:
   - Get a SCIM Authentication Token:
     - In the ADB Owner Portal, navigate to the SSO page
     - Select the SAML provider and click **Edit**
     - Click **Edit Provisioning**, then Get SCIM Token.
   - Configure Provisioning in the Entra Console:
     - Go to the Enterprise Application's Provisioning tab.
     - Select Provisioning Mode: Automatic.
     - Under Admin Credentials/Tenant URL, specify the SCIM endpoint: https://<gatekeeper>/api/scim.
     - For Admin Credentials/Secret Token, paste the SCIM Auth Token from the first step.
     - Click **Save Changes**.
     - Click **Start Provisioning**.
5. Assigning ADB Global Administrator role:
   - Wait for the initial provisioning:
     - Allow time for the initial provisioning cycle to complete. You can check the provisioning status in the Microsoft Entra portal.
   - Assign Global Admin Role:
     - Once the provisioning cycle is complete, go to the ADB Owner Portal, SSO page.
     - Select the List Users button for the SAML provider.
     - Confirm that the users or groups have been imported correctly.
     - Select the user to grant Global Administrator permissions to and click **Set User** as Global Admin.

## Configuring OIDC Authentication with Microsoft Entra

1. Install the ADB Gatekeeper/Gateway.
2. Create an ENTRA Enterprise Application for ADB:
   - Create an ENTRA Enterprise Application:
     - In the ENTRA console, navigate to Enterprise Applications
     - Select **Create a new Application**
     - Give the application a name and select Integrate any other application you don't find in the gallery (Non-gallery)
     - Click **Create.**
3. Assign Users and Groups:
   - In the ENTRA console, assign users and groups to the newly created Enterprise Application.
4. Configure Application Authentication:
   - In the ENTRA console, go to Applications/App Registrations

- Select the application
- Under Authentication, select Add a Platform and add the Web platform
- Set Redirect URI to https://<gatekeeper>/Portal/SSO/OIDC
- Set Front Channel Logout URL to https://<gatekeeper>/Portal/SSO/Logout
- Check the Identity Tokens checkbox.

5. Configure ADB to use OIDC authentication.
   - You will need the following information from the ENTRA App Registration:
     - Application (client) ID
     - Directory (tenant) ID
     - OpenID Connect metadata document URL (Click on Endpoints)
   - In the ADB owner portal, on the SSO page, click **Add OIDC Provider,** and configure the following settings:
     - **Provider Name:** Specify a name for this identity provider.
     - **Tenancy ID:** Must be 1.
     - **Is Default:** Set to checked.
     - **Provisioning Mode:** Automatic Provisioning
     - **Config URL:** The OpenID Connect metadata document URL.
     - **Client ID:** The Application (client) ID
     - **Claims Mapping:** Click Configure Claims Mapping and set the following values:
       - Match login claims to users using this property: EmailAddress
       - Unique ID: email
       - Username: email
       - Email Address: email
     - **ExtraPropertyName:** tenancyId
     - **ExtraPropertyValue1:** The Directory (tenant) ID
     - **ExtraPropertyName2:** scope
     - **ExtraPropertyValue2:** openid email
     - Click **Save Changes.**
6. Set up SCIM provisioning
   - In the ADB owner portal, get a SCIM authentication token:
     - On the SSO page, select the SAML provider, and click **Edit**.
     - Click **Edit Provisioning**, then click **Get SCIM Token**.
   - In the Entra console: go to the Enterprise Application’s Provisioning tab.
     - Select Provisioning Mode: Automatic.
     - Under Admin Credentials/Tenant URL, specify the SCIM endpoint: https://<gatekeeper>/api/scim
   - For the Admin Credentials or Secret Token:
     - Paste the Scim Auth Token from the first step

        - Click **Save Changes**
        - Click **Start Provisioning.**

7 Assign ADB Global Administrator role
    - Wait for the initial provisioning cycle to complete.
    - In the ADB owner portal:
        - Go to the SSO page, and select the **List Users** button for the OIDC provider.
    - Confirm that the users/groups have been imported correctly:
        - Select the user to grant Global Administrator permissions to
        - Click Set User as Global Admin.

## Configuring SAML Authentication with ENTRA

1 Install the ADB Gatekeeper and Gateway.

2 Create and configure an ENTRA Enterprise Application for ADB.
    - In the OKTA console, go to Applications, and click “Create App Integration”
        - Select “SAML 2.0” as the authentication type.
        - Give the application a name and click Next.
        - Configure the SAML settings:
            - Single Signon URL: https://<gatekeeper>/Portal/SSO/SamlACS
            - Use this for Recipient URL and Destination URL: Checked
            - Audience URI: https://<gatekeeper>
            - Default Relay State specify a domain name for ADB to use for the ENTRA users and groups (for example “ENTRA” or “MYDOMAIN”)
            - NameIDFormat: EmailAddress
            - Application Username: ENTRA Username
            - Update application username on: Create and Update
            - Under Advanced Options, upload the HAPI certificate: (On the Gatekeeper machine, the certificate can be found in C:\Program Files\OpenText\AD Bridge\Gatekeeper\nginx\conf\certificate.crt)
            - Check the “Allow application to initiate single logout” checkbox Single Logout URL: https://<gatekeeper>/Portal/SSO/SLO
        - In the ENTRA console, Assign Users and Groups to the Application.
        - Under “Relay State” specify a domain name for ADB to use for the ENTRA users and groups (for example “ENTRA” or “MYDOMAIN”)
        - Download the ENTRA SAML metadata and save the metadata to a file.

3 Configure ADB to use SAML authentication.
    - In the ADB Owner Portal, SSO settings, click “Add SAML Provider”.
        - Set the provider name to the same value used for Relay State.

   - Check the IsDefault checkbox. This causes the ADB web console to use this SAML provider as the default identity provider for logins. To login with a non-default SAML provider, go to https://<gatekeeper>/Portal/SSO/
     SamlLogin?provider=<ProviderName>
   - Set NameIdFormat to Email Address.
   - Set SignatureAlgorithm to SHA_256.
   - Set provisioning mode to AutomaticProvisioning.
   - Save Changes

4. Set up SCIM provisioning
   - In the ADB owner portal, get a SCIM authentication token:
      - On the SSO page, select the SAML provider, and click Edit.
      - Click “Edit Provisioning”, then “Get SCIM Token”.
   - In the OKTA console, go to the ADB Application, and check the “Enable SCIM Provisioning” checkbox.
   - In the Provisioning/Integration tab, set the following values:
      - Scim Connector Base URL: https://<gatekeeper>/api/scim
      - Unique Identifier field for Users: username
      - Push New Users
      - Push Profile Updates
      - Push Groups
      - Authentication Mode: Http Header Token: <the SCIM token from ADB
   - Under Provisioning/To App/Attribute Mappings, remove the following mappings:
      - Manager Value

        Employee Number

        Cost Center

        Organization

        Division

        Department

        Manager Display Name
   - Click “Save Changes.
   - Click “Force Sync”

5. Assign ADB Global Administrator role
   - Wait for the initial provisioning cycle to complete. Once the provisioning cycle has completed, go to the ADB owner portal, SSO page, and select the “List Users” button for the SAML provider.
   - Confirm that the users/groups have been imported correctly, then select the user to grant Global Administrator permissions to, and click “Set User as Global Admin”.

## Configuring OIDC Authentication with OKTA

1. Install the ADB Gatekeeper/Gateway.
2. Create and configure an OKTA Application for ADB
    - In the OKTA console, click “Create App Integration”
        - Select Sign-in Method: OIDC – Open ID Connect
        - Select Application Type: Web Application
        - Specify an application name
        - Select grant types “Authorization Code”, “Refresh Token” and “Implicit (hybrid)
        - Set Sign-in redirect URI: https://<gatekeeper>/Portal/SSO/OIDC
        - Set Sign-out redirect URI: https://<gatekeeper>/Portal/SSO/Logout
    - Set user/group assignments as desired.
3. Configure ADB to use OIDC authentication.
    - You will need the Client ID from the OKTA Application properties.
    - In the ADB owner portal, SSO page, click “Add OIDC Provider”, and configure the following settings:
        - Provider Name: Specify a name for this identity provider.
        - Tenancy ID: Must be 1.
        - Is Default: Set to checked.
        - Provisioning Mode: Automatic Provisioning
        - Config URL: https://${yourOktaDomain}/.well-known/openid-configuration (see https://developer.okta.com/docs/reference/api/oidc/#well-known-openid-configuration)
        - Client ID: The Application (client) ID
        - Configure Claims Mapping: Click "Configure Claims Mapping" and set the following values:
            - Match login claims to users using this property: EmailAddress
            - Unique ID: email
            - Username: email
            - Email Address: email
        - Click Save Changes.
4. Automatic Provisioning:
    - Configure SCIM provisioning
    - OKTA does not currently support SCIM provisioning for OIDC applications. In order to use OKTA provisioning, you must create a SAML Application in OKTA.
        - Create an additional Application in OKTA. Choose SAML 2.0.
        - Specify a name for the application. Specify the gatekeeper URL in the required URL fields (these values will not be used, because this App will only be used for provisioning, not authentication). Check “Enable SCIM Provisioioning” and “Do not display application icon to users”.

- In the Provisioning/Integration tab, set the following values:
    - Scim Connector Base URL: https://<gatekeeper>/api/scim
    - Unique Identifier field for Users: username
    - Push New Users
    - Push Profile Updates
    - Push Groups
    - Authentication Mode: Http Header
    - Token: <the SCIM token from ADB>
- Under Provisioning/To App/Attribute Mappings, remove the following mappings:
    - Manager Value

      Employee Number

      Cost Center

      Organization

      Division

      Department

      Manager Display Name

      Click “Save Changes”.

      Click “Force Sync”
- Alternatively, instead of Automatic Provisioning, you can use JustInTime provisioining to enable JustInTime provisioning:
    - In the OKTA console, in Directory/Profile Editor, create a custom attribute “ADBRole” (the name of the attribute doesn’t matter).
    - Add a mapping for the custom property to the  Application profile for the application.
    - Populate the ADBRole for each user with the name of a role assignment in ADB.

      **NOTE:** When the user attempts to log in to the ADB console, a SCIM user will be created for them, if one doesn’t already exist. If the SCIM user has not been assigned to any roles, it will be assigned to the role specified in the ADBRole property.

    - In the ADB owner portal, set the provider’s ProvisioiningMode to “JustInTime”.
    - For additional properties enter: scope “openid email profile”
    - Add the following Attribute Mappings:
        - DisplayName=”name”

          Email=”email”

          UserName=”email”

          RoleAssignment=”ADBRole”

**5** Assign ADB Global Administrator role

- Wait for the initial provisioning cycle to complete. Once the provisioning cycle has completed, go to the ADB owner portal, SSO page, and select the “List Users” button for the OIDC provider.
- Confirm that the users/groups have been imported correctly, then select the user to grant Global Administrator permissions to, and click “Set User as Global Admin”.

## Configuring SAML Authentication with Ping Identity

**1** Install the ADB Gatekeeper and Gateway.

**2** Create and configure a Ping Identity Application for ADB.

- In the Ping Identity console, Select Applications, and click the “+” button.
  - Give the application a name, and select “SAML Application”
  - Click Configure.
- Select “Import from URL”. Enter the following url: https://<gatekeeper>/Portal/SSO/GetSPMetadata and click Import, then Save.
- On the attribute mappings tab, specify the following mappings:
  - saml-subject: User ID

    email: Email Address

    username: Username
- On the configuration tab, click “Download Metadata”

**3** Configure ADB to use SAML authentication.

- In the ADB Owner Portal, SSO settings, click “Add SAML Provider”.
  - Specify a name for the identity provider.
  - Check the IsDefault checkbox. This causes the ADB web console to use this SAML provider as the default identity provider for logins. To login with a non-default SAML provider, go to https://<gatekeeper>/Portal/SSO/SamlLogin?provider=<ProviderName> .
  - Set NameIdFormat to Email Address.
  - Set SignatureAlgorithm to SHA_256
  - Set provisioning mode to AutomaticProvisioning.
  - Set the following values for SAML claims:
    - DisplayName: userName

      Email: email

      Unique ID: userName
  - Save Changes.

**4** Set up SCIM provisioning

- In the ADB owner portal, get a SCIM authentication token: On the SSO page, select the SAML provider, and click Edit. Click “Edit Provisioning”, then “Get SCIM Token”.
- In the PingIdentity console, go to Integrations/Provisioning/New Connection.
- Choose Connection Type: Identity Store, then choose “SCIM Outbound”.

- Specify a name for the connection and click Next.
- Set the following properties on the Configure Authentication page:
  - Scim Base URL: https://<gatekeeper>/api/scim

    Users Resource: /Users

    SCIM Version: 2.0

    Authentication method: “OAuth 2 Bearer token”

    Auth type header: Bearer

    Oauth Access Token: <the SCIM token from ADB>
- Integrations/Provisioning/Rules/New Rule.
  - Select the SCIM connection you created in the previous step.

    Configure the user filter and attribute mappings, if desired.

    Enable the rule.

5 Assign ADB Global Administrator role:

- Wait for the initial provisioning cycle to complete. You can check the provisioning status in provisioning rule on the Ping Identity portal.
- Once the provisioning cycle has completed, go to the ADB owner portal, SSO page, and select the “List Users” button for the SAML provider.
- Confirm that the users/groups have been imported correctly, then select the user to grant Global Administrator permissions to, and click “Set User as Global Admin”.

## Configuring OIDC Authentication with Ping Identity

1 Install the ADB Gatekeeper and Gateway.

2 Create and configure a Ping Identity Application for ADB.

- In the Ping Identity console, Select Applications, and click the “+” button.
  - Give the application a name, and select “OIDC Web App”

    Click Save.
- Edit Configuration:
  - Response Type: Code, ID Token

    Grant Type : Authorization Code

    Redirect URIs: https://<gatekeeper>/Portal/SSO/OIDC https://<gatekeeper>

    Token endpoint authentication method: Client Secret Basic

    Initiate Login URI: https://<gatekeeper>/Portal/SSO/OIDCLogin?provider=<ADBProviderName>
- On the attribute mappings tab, specify the following mappings:
  - ub, UserID, openid
  - email, Email Address,openid
  - userName: Username, openid

3. Configure ADB to use OIDC authentication.

   - You will need the following information from the configuration tab of the PingIdentity Application:

     - Application ID

       OpenID Connect metadata document URL

   - In the ADB owner portal, SSO page, click “Add OIDC Provider”, and configure the following settings:

     - Provider Name: Specify a name for this identity provider.

       Tenancy ID: Must be 1.

       Is Default: Set to checked.

       Provisioning Mode: Automatic Provisioning

       Config URL: The OpenID Connect metadata document URL.

       Client ID: The Client ID

       Identity Claim: email

       Click Save Changes.

4. Set up SCIM provisioning

   - In the ADB owner portal, get a SCIM authentication token: On the SSO page, select the SAML provider, and click Edit.

     Click “Edit Provisioning”, then “Get SCIM Token”.

     In the PingIdentity console, go to Integrations/Provisioning/New Connection.

     Choose Connection Type: Identity Store, then choose “SCIM Outbound”.

     Specify a name for the connection and click Next.

     Set the following properties on the Configure Authentication page:

     Scim Base URL: https://<gatekeeper>/api/scim

     - Users Resource: /Users

       SCIM Version: 2.0

       Authentication method: “OAuth 2 Bearer token”

       Auth type header: Bearer

       Oauth Access Token: <the SCIM token from ADB>

   - Integrations/Provisioning/Rules/New Rule.

     - Select the SCIM connection you created in the previous step.

       Configure the user filter and attribute mappings, if desired.

       Enable the rule.

5. Assign ADB Global Administrator role:

   - Wait for the initial provisioning cycle to complete. You can check the provisioning status in provisioning rule on the Ping Identity portal.

Once the provisioning cycle has completed, go to the ADB owner portal, SSO page, and select the “List Users” button for the OIDC provider.

Confirm that the users/groups have been imported correctly, then select the user to grant Global Administrator permissions to, and click “Set User as Global Admin”.

## Configuring SAML Authentication with Amazon IAM

1. Install the ADB Gatekeeper and Gateway.
2. Create and configure an Amazon IAM Application for ADB.
    - In the Amazon IAM console, Select Applications, and click "Add Application”.
        - Select “I have an application I want to set up”

          Select Application type “SAML 2.0”, and click “Next”
    - Specify a name for the Application.
    - Click the link to download the IAM Identity Center SAML metadata file.
    - Specify the Application Start URL and Relay State:
        - Application Start URL: https://<gatekeeper>/Portal/SSO/SamlLogin?provider=<ProviderName>
        - Relay State: <ProviderName>

          Where ProviderName is the name you will give this SAML provider in the ADB owner portal.
    - Download the ADB SAML metadata from: https://<gatekeeper>/Portal/SSO/GetSPMetadata and save it to a file.
    - In the IAM Application Metadata section, select “Upload Application SAML Metadata file”, and select the downloaded ADB SAML metadata.
    - Assign users and groups to the application as desired.
3. Configure ADB to use SAML authentication.
    - Specify a name for the identity provider.

      Check the IsDefault checkbox. This causes the ADB web console to use this SAML provider as the default identity provider for logins. To login with a non-default SAML provider, go to https://<gatekeeper>/Portal/SSO/SamlLogin?provider=<ProviderName>

      Set NameIdFormat to Email Address.

      Set SignatureAlgorithm to SHA_256.

      Set provisioning mode to AutomaticProvisioning.
    - Set the following values for SAML claims:
        - DisplayName: name

          Email: email

          Unique ID: name
    - Save Changes.
4. Set up SCIM provisioning
    - In the ADB owner portal, get a SCIM authentication token: On the SSO page, select the SAML provider, and click Edit. Click “Edit Provisioning”, then “Get SCIM Token”.

In the PingIdentity console, go to Integrations/Provisioning/New Connection.

Choose Connection Type: Identity Store, then choose “SCIM Outbound”.

Specify a name for the connection and click Next.

- Set the following properties on the Configure Authentication page:
  - Scim Base URL: https://<gatekeeper>/api/scim

    Users Resource: /Users

    SCIM Version: 2.0

    Authentication method: “OAuth 2 Bearer token”

    Auth type header: Bearer

    Oauth Access Token: <the SCIM token from ADB>
- Integrations/Provisioning/Rules/New Rule.
  - Select the SCIM connection you created in the previous step.

    Configure the user filter and attribute mappings, if desired.

    Enable the rule.

5. Assign ADB Global Administrator role:
   - Wait for the initial provisioning cycle to complete. You can check the provisioning status in provisioning rule on the Ping Identity portal.
   - Once the provisioning cycle has completed, go to the ADB owner portal, SSO page, and select the “List Users” button for the SAML provider.
   - Confirm that the users/groups have been imported correctly, then select the user to grant Global Administrator permissions to, and click “Set User as Global Admin”.

## Configuring OIDC Authentication with Ping Identity

1. Install the ADB Gatekeeper and Gateway.
2. Create and configure a Ping Identity Application for ADB.
   - In the Ping Identity console, Select Applications, and click the “+” button.
     - Give the application a name, and select “OIDC Web App”

       Click Save.
   - Edit Configuration:
     - Response Type: Code, ID Token

       Grant Type : Authorization Code

       Redirect URIs: https://<gatekeeper>/Portal/SSO/OIDChttps://<gatekeeper>

       Token endpoint authentication method: Client Secret Basic Initiate Login URI: https://<gatekeeper>/Portal/SSO/OIDCLogin?provider=<ADBProviderName>
   - On the attribute mappings tab, specify the following mappings:
     - ub, UserID, openid

       email, Email Address,openid

       userName: Username, openid

**3** Configure ADB to use OIDC authentication.

- You will need the following information from the configuration tab of the PingIdentity Application:
  - Application ID

    OpenID Connect metadata document URL
- In the ADB owner portal, SSO page, click “Add OIDC Provider”, and configure the following settings:
  - Provider Name: Specify a name for this identity provider.

    Tenancy ID: Must be 1.

    Is Default: Set to checked.

    Provisioning Mode: Automatic Provisioning

    Config URL: The OpenID Connect metadata document URL.

    Client ID: The Client ID

    Identity Claim: email

    Click Save Changes.

**4** Set up SCIM provisioning

- In the ADB owner portal, get a SCIM authentication token:

  On the SSO page, select the SAML provider, and click Edit.

  Click “Edit Provisioning”, then “Get SCIM Token”.

  In the PingIdentity console, go to Integrations/Provisioning/New Connection.

  Choose Connection Type: Identity Store, then choose “SCIM Outbound”.

  Specify a name for the connection and click Next.
- Set the following properties on the Configure Authentication page:
  - Scim Base URL: https://<gatekeeper>/api/scim

    Users Resource: /Users

    SCIM Version: 2.0

    Authentication method: “OAuth 2 Bearer token”

    Auth type header: Bearer

    Oauth Access Token: <the SCIM token from ADB>
- Integrations/Provisioning/Rules/New Rule.
  - Select the SCIM connection you created in the previous step.

    Configure the user filter and attribute mappings, if desired.

    Enable the rule.

**5** Assign ADB Global Administrator role:

- Wait for the initial provisioning cycle to complete. You can check the provisioning status in provisioning rule on the Ping Identity portal.

Once the provisioning cycle has completed, go to the ADB owner portal, SSO page, and select the “List Users” button for the OIDC provider.

Confirm that the users/groups have been imported correctly, then select the user to grant Global Administrator permissions to, and click “Set User as Global Admin”.

# Silent Agent Installation

In AD Bridge, you can now silently install agents with an agent install token that can be tied to a specific OU for the agent to land in. This token can be retrieved from the Universal Policy Administrator Web Console.

**To Install an Agent with an Agent Install Token**

1. Login to the **Web Console** as an Administrator.
2. Click the **Administrator** icon at the top-right corner.
3. Click **Install Agents** to open the Install Agents window.
4. In the **Install Agents** window, select the Cloud OU where the agents should be placed during installation.
5. Copy the token displayed.
6. Run the `"AD Bridge Agent.exe" /i/quiet GatekeeperBaseUrl=https://vlab011745.dom012700.lab" LoginAuthToken="5t9DHaZkoi6I16OBJxnTOVAkVzPKBYmZ0T81nZR4Y-k1"` command in the Command Prompt (Change the Gatekeeper URL and the Agent Install Token as needed).

# Agent Policy Push

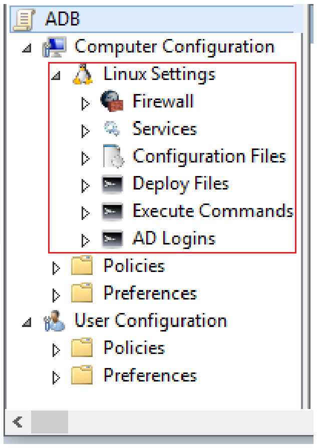
The Agent Policy Push feature in AD Bridge ensures that any universal policy created for AD Bridge in the Universal Policy Administrator console is instantly available on the agent machine.

**NOTE:** Agent Policy Push is designed to work specifically with cloud-only agents and applies to UNIX policies.

When you create a universal policy, link the policy to the appropriate cloud organizational unit (OU) and assign it to the specific agent machine. This ensures instant deployment, reducing time lag and enhancing administrative efficiency.

# 3 Managing Linux GPO Settings

If you have the AD Bridge GPEdit Extension for the Group Policy Management Console (GPMC) installed on your domain controller, you will see a new node, **Linux Settings**, under Computer Configuration when you open the Group Policy Object (GPO) editor on a GPO. This node has five child nodes, Firewall, Services, Configuration Files, Deploy Files, Execute Commands and AD Logins, which you can use to create, modify, or delete GPO settings for Linux Agent clients in the domain.



When you link a GPO that has rules configured in Linux Settings to an OU that has one or more AD Bridge Linux agents, those GPO settings are applied to the Linux computers in that OU (assuming the Linux Agent Service is running on those computers).

This section demonstrates how to create a new GPO and configure rules in the AD Bridge GPMC snap-in and apply them to your Linux Agent computers. When an AD Bridge Linux Agent is installed on a Linux computer, the computer is automatically added to the Active Directory's "Computers" OU. As a best practice, you should create a custom OU for linking GPOs to Linux Agent computers in your environment.

**IMPORTANT:** To minimize the risk of introducing harmful Group Policy errors into your production environment, you should thoroughly test and evaluate Linux Agent GPOs in a non-production environment before you implement them.

For best practice information about configuring GPOs in AD Bridge, see GPO Best Practices.

# Accessing or Creating Group Policy Objects

In order to modify, create, or delete group policies for Linux Agent computers in the Active Directory domain, you either need work with existing GPOs or create new GPOs. These GPOs must be linked to any applicable Linux agents in an OU for the group policies to be effective.

When you locate a GPO or create a new one, you open the Group Policy Management Editor, expand the Linux Agent Settings node, and use the AD Bridge GPEdit Extension snap-in to make policy changes in the editor.

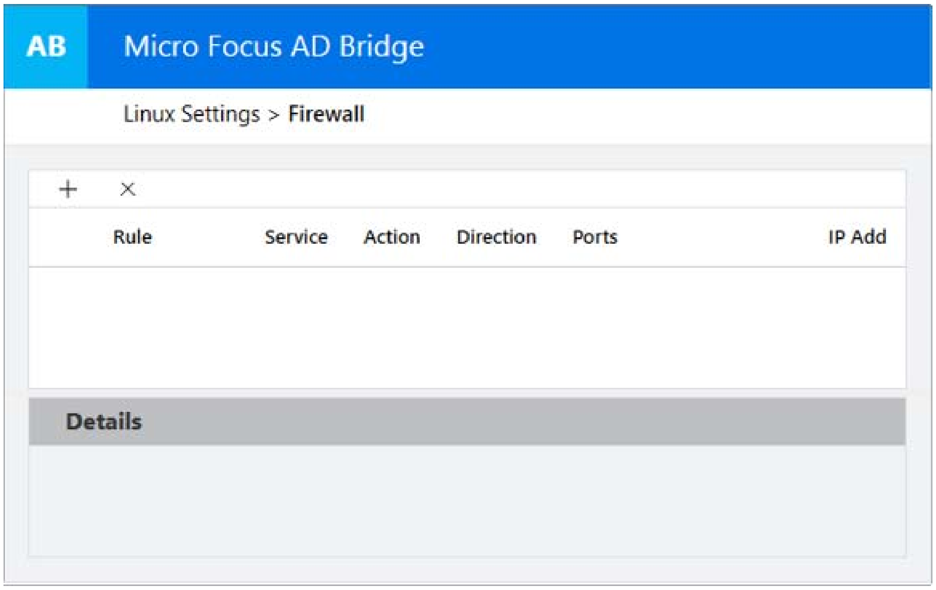
In general, you will use the plus symbol + to add or access settings or rules, if they are not selectable in the snap-in pane, and you will use the delete symbol x to remove them.

**To open the GPEdit Extension snap-in on a GPO:**

1. Open the GPMC on the domain controller where the GPEdit Extension was installed or from a server in the same domain.
2. Expand the domain tree and OU that contains the Linux Agent.
3. Right-click the applicable GPO, and select **Edit** to open the GPO editor.

   If needed, you can create a new GPO that is linked to the OU and then open the editor from the new object.
4. Expand **Linux Settings** in the GPO editor to access the setting nodes and the GPEdit Extension.

   The GPEdit Extension snap-in is shown below as it appears on a new GPO when first opening the Firewall policy settings.



# Configuring Linux GPO Settings

The procedures below provide an example of setting a new Firewall rule in a GPO and applying it to a Linux Agent. For this example we already have a custom OU that contains our Linux Agent.

For information about opening the GPEdit Extension snap-in on a GPO, see Accessing or Creating Group Policy Objects.

**To create a Linux Agent Firewall GPO in the AD Bridge GPMC snap-in:**

1. Right-click **Group Policy Objects** in the domain tree, and select **New** to create a new GPO.
2. Right-click the new GPO and select **Edit** to open the GPO editor.
3. Expand **Linux Settings** in the GPO editor, and click the **Firewall** node.
4. Click **+** to open the Rule drop-down menu in the AD Bridge snap-in, select **Allow HTTP** from the Rule list, and click **Save** to enable the new rule.

You can also create custom Firewall rules to block or allow Inbound or Outbound traffic based on port, protocol, or IP address.

**To add a custom Linux Agent Firewall rule in GPMC:**

1. Expand **Linux Settings** in the GPO editor, and click the **Firewall** node.
2. Click **+** to open the Rule drop-down menu in the AD Bridge snap-in, and select **Custom Rule** from the Rule list.
3. Use the Firewall Rule dialog box to name and configure the Action, Direction, Port, Protocol, and IP Address for the custom rule.
4. Add and save your changes.

**To apply a new or modified GPO to one or more Linux agents:**

1. Select the GPO with the new or modified Linux setting and drag it onto the OU that contains the Linux Agent(s).
2. Click **OK** to link the GPO to the OU and apply the policy to any applicable Linux Agent computers.

**NOTE:** In order for the policy to be applied to Linux Agent computers, the Linux Agent Service must be running on those devices. If the service is not running, use one of the commands below, applicable to the platform, to start the service:

- `systemctl start adb-agent.service`
- `service adb-agent start`

For information about all the GPO settings available in the GPEdit Extension snap-in, see Linux Agent GPO Settings.

# Managing Linux Agent Services with GPOs

You can monitor, start, stop, and restart services on Linux Agent computers from a GPO in the GPEdit Extension snap-in. You can use an existing GPO or create a new one, but the GPO needs to be linked with the OU that has the Linux Agent or agents where you want to perform the action.

**NOTE:** You can use the command-line interface to refresh Linux Agent policies.

Linux agents deliver flexible installation and configuration capability to work across enterprise and cloud. Supported Linux Agent Install Modes include agent machines joined to:

- **On Premises AD:** Manage with native tools
- **Cloud AD:** Manage with GPEdit Extension

- **Cloud Non AD:** Manage with web console or third party AD tools

This allows you to monitor files in real time and for persistence of local Linux Configuration files, outside of GPOs and the Sysvol check cycle.

In addition, The Linux cloud agent helps extend on premises AD management capabilities to cloud based Linux resources. This permits you to leverage on premises AD authorization and authentication to improve security and reduce the number of unmanaged identities.

For information about accessing or creating GPOs in Active Directory, see Accessing or Creating Group Policy Objects.

**To start, stop, or restart a service on a Linux Agent computer:**

1. Right-click the applicable GPO or GPO link in an OU, and select **Edit** to open the GPO editor.

   If needed, you can create a new GPO that is linked to the OU and then open the editor from the new object.
2. Expand **Linux Settings** in the GPO editor, and click the **Services** node.
3. Click the plus icon **+**, and enter the service name. This must be the actual service name as opposed to the friendly name of the service.
4. Select the desired option (**Start**, **Stop**, or **Restart**), and click **Add**.

**NOTE:** For information about checking service status on a Linux Agent, see Verify the Linux Agent Service is running.

# Importing Custom Configuration File Settings

From the Configuration Files node, you can import custom settings for Configuration Files into your Linux Agent. This enables you to create GPOs to manage the configuration of custom or legacy applications. When you import Configuration File settings, you can do the following:

- Add new settings without removing existing settings
- Change existing settings
- Overwrite existing settings
- Create a new configuration file

For information about accessing or creating GPOs in Active Directory, see Accessing or Creating Group Policy Objects.

**To import custom Configuration File settings:**

1. Right-click the applicable GPO or GPO link in the OU and select **Edit** to open the GPO editor.

   If needed, you can create a new GPO that is linked to the OU and then open the editor from the new object.
2. In the GPO editor, expand **Linux Settings**, right-click **Configuration Files**, and select **Add Custom Configuration File**.

   **NOTE:** While adding a custom configuration file using Add Custom Configuration File, you can use a hyphen(-) in the file name.

3 Provide the path and file name for the file you want to import, and click **Add**.

   If the specified configuration file does not exist, you have the option to create a new file from which you can create new custom Configuration File settings.

**To modify Configuration File settings:**

1 Click **OpenSSH** or **Sudoers** or a file in the Configuration Files node that you imported as demonstrated above.

2 Click the **+** icon and do one of the following:

   - Select an existing rule and modify the attributes as desired.
   - Add a custom rule and specify the attributes as desired.

3 If you want to delete an existing rule, select the rule, and click the **x** icon to delete it.

**To overwrite the existing settings:**

1 Click **OpenSSH** or **Sudoers** or a file in the Configuration Files node.

2 Select **Overwrite** check box to modify an existing rule. The existing values of the **Setting** and **Rule** get overwritten with the new values.

# Managing Linux Applications with GPOs

You can deploy application files on Linux Agent computers using GPOs to harden, manage, and persist application settings on these computers. With these GPOs in place, any attempts to modify an application configuration from the Linux Agent computer will be overwritten by the GPO configuration.

This is done from the Deploy Files node by importing existing application files into one or more GPOs and assigning the GPOs to the Linux Agent OU. All changes going forward for these applications can then be managed from the GPOs in Active Directory.

For example, if you have a Web Service in your enterprise environment that manages user access on the Internet or Intranet by restricting communication based on IP addresses, you can modify these settings in the GPO.

Before you can manage a Linux Agent application using a GPO, the following prerequisites need to be met:

- The GPO must be linked to applicable Linux agents
- You need to know the relative path for deploying the configuration file on the agent
- You need to know the location of the application file you will use to configure the group policy

**To begin managing Linux applications using GPOs:**

1 Expand the domain tree and OU that contains the applicable Linux Agent(s).

2 Right-click the applicable GPO, and select **Edit** to open the GPO editor.

3 Expand **Linux Settings** in the GPO editor, and click the **Deploy Files** node.

4 Click the plus symbol **+** in the GPEdit Extension and do the following:

   **4a** Name the new rule.

   **4b** Click the browse button and locate the application file.

4c Enter the relative path on the Linux Agent(s) where you will deploy the GPO configuration file.

4d Click **Add**.

5 Once you have the application file added to the GPO, make any required configuration changes from the GPEdit Extension options and save your changes to apply the group policy to the Linux Agent computers.

**NOTE:** You can add and deploy more than one application configuration to a GPO.

# Executing Commands with GPOs

You can create GPOs to execute commands or run shell scripts on your local computer, once or every hour.

**To execute a command in GPMC:**

1. Expand **Linux Settings** in the GPO editor, and click the **Execute Command** node.
2. Click **+** to add a command once if you choose to.
3. Add and save your changes.

# Managing User Logins with GPOs

Using group policies, you can control which users and groups are allowed or denied to log in on Linux Agent computers in your Active Directory domain. This is accomplished by creating or modifying one or more GPOs and setting the login privileges for specified users or groups.

**NOTE:** For cloud AD logins, users or groups must be part of the MFPolicy-Users group.

For information about accessing or creating GPOs in Active Directory, see Accessing or Creating Group Policy Objects.

**To configure and apply GPO login settings on Linux agents:**

1. Right-click the applicable GPO or GPO link in an OU, and select **Edit** to open the GPO editor.

   If needed, you can create a new GPO that is linked to the OU and then open the editor from the new object.
2. Expand **Linux Settings** in the GPO editor, and click the **AD Login** node.
3. Click the plus icon **+** and select **AD login provider mode**. Then select a mode in the pull-down menu.

   For example, select **Simple allow/deny list**.
4. Click the plus icon **+** again, and select the desired rule.

   For example, select **Prevent these AD users from logging in**.

**IMPORTANT:** When you configure a GPO to prevent users or groups from logging in, this in effect an exclusionary list for Active Directory objects. However, when you configure to “Allow AD users or groups” those objects will be the only AD users or groups that will be able to login on the Linux agents that have the GPO applied. You cannot have both Allow and Deny logins in the policy at the same time.

5 Click the browse button, and use the **Select Users** dialog box to (a) define if the rule is for users or groups, (b) choose the applicable domain, and (c) locate required users and or groups that are applicable to the policy.

6 Save the changes to apply the policy to applicable Linux agents.

**NOTE:** In order for the policy to be applied to Linux Agent computers, the Linux Agent Service must be running on those devices. If the service is not running, use one of the commands below, applicable to the platform, to start the service:

- `systemctl start adb-agent.service`
- `service adb-agent start`

## Managing User and Group IDs in Linux

You can manage AD objects with Active Directory Users and Computers (ADUC). An ADUC extension based tab **AD Bridge** allows you to manage User ID (UID) and Group ID (GID) for Linux users. The options available to manage are:

- Override ID Mapping
- UID
- GID

You must follow the steps below to enable this tab:

1 Change directory to `/etc/sssd/sssd.conf`.

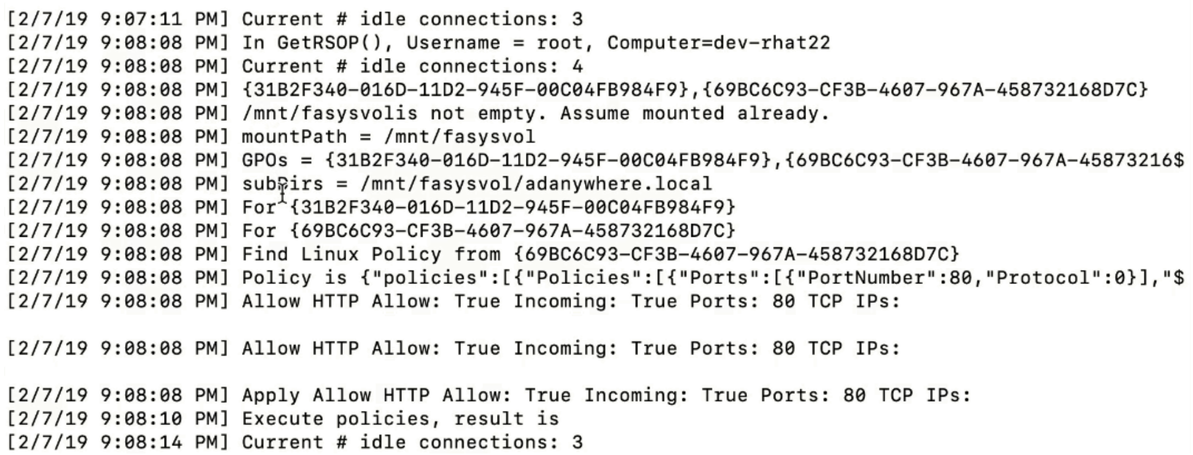
2 Edit `sssd.conf` and modify the value of the parameter `ldap_id_mapping` to `False`.

3 Restart the service with either of the commands, `systemctl restart sssd` or `service sssd restart`.

**NOTE:** You may need to wait for about 15 minutes to see changes take effect.

# Viewing Policy Injection on a Linux Agent

When a GPO rule is applied to the OU with one or more AD Bridge Linux agents, that setting is executed on the Linux Agent(s) and can be viewed with a tail of the log at `/var/log/adb-agent.log`, as shown in the example below:



```
[2/7/19 9:07:11 PM] Current # idle connections: 3
[2/7/19 9:08:08 PM] In GetRSOP(), Username = root, Computer=dev-rhat22
[2/7/19 9:08:08 PM] Current # idle connections: 4
[2/7/19 9:08:08 PM] {31B2F340-016D-11D2-945F-00C04FB984F9},{69BC6C93-CF3B-4607-967A-458732168D7C}
[2/7/19 9:08:08 PM] /mnt/fasysvolis not empty. Assume mounted already.
[2/7/19 9:08:08 PM] mountPath = /mnt/fasysvol
[2/7/19 9:08:08 PM] GPOs = {31B2F340-016D-11D2-945F-00C04FB984F9},{69BC6C93-CF3B-4607-967A-45873216$
[2/7/19 9:08:08 PM] subDirs = /mnt/fasysvol/adanywhere.local
[2/7/19 9:08:08 PM] For {31B2F340-016D-11D2-945F-00C04FB984F9}
[2/7/19 9:08:08 PM] For {69BC6C93-CF3B-4607-967A-458732168D7C}
[2/7/19 9:08:08 PM] Find Linux Policy from {69BC6C93-CF3B-4607-967A-458732168D7C}
[2/7/19 9:08:08 PM] Policy is {"policies":[{"Policies":[{"Ports":[{"PortNumber":80,"Protocol":0}],"$
[2/7/19 9:08:08 PM] Allow HTTP Allow: True Incoming: True Ports: 80 TCP IPs:

[2/7/19 9:08:08 PM] Allow HTTP Allow: True Incoming: True Ports: 80 TCP IPs:

[2/7/19 9:08:08 PM] Apply Allow HTTP Allow: True Incoming: True Ports: 80 TCP IPs:
[2/7/19 9:08:10 PM] Execute policies, result is
[2/7/19 9:08:14 PM] Current # idle connections: 3
```

You can also manage policies from the web console including, creation, modification, version control, approval, and deletion of policies for on premises and cloud based resources.

# 4 Using the Web Console

AD Bridge extends Active Directory (AD) capabilities further by adding a web console to oversee and manage policies and agents. The web console also displays information in figures and charts. This simplifies management and delivers analytics that demonstrate the effectiveness and reach of AD Bridge.

You can identify and manage domain joined Linux devices (both on premises and cloud), on a browser to improve security and provide better visibility into the AD Bridge infrastructure from any supported device and location. Thus, this single dashboard centralizes device and policy management beyond your organization as well.

To add a web console, you must first set up your web server in Microsoft Azure. The web console displays the following category types with charts:

- **Devices:** You can view and manage environment, agents versions and connection types across the AD Bridge infrastructure and devices.

  **NOTE:** You can link an available Universal Policy to a selected device.

- **Policies:** You can view and manage Linux and Windows policies and also create Universal Policies or import existing policies from a GPO in Active Directory and save them as universal policies. You can also:
  - Modify, approve, deploy (to agent machines) and export (to AD as GPOs) existing policies
  - Delete policies
  - Refresh policies

## Creating Universal Policies

**To create a Universal Policy from the web console:**

1. Click **+** to open the **New Universal Policy** dialog box in the web console.
2. Enter a name for the New Universal Policy.
3. (Optional) Select **Import policies from a GPO in Active Directory** and choose policies to import.
4. Click **Create**.
5. Select the created policy and click **+** to add additional settings.

## Exporting Universal Policies

**To export a Universal Policy from the web console:**

1. Select a Universal Policy and click **Export to Active Directory**.
2. Click **+** to open the **Export Universal Policy** dialog box.

3 Click **+** to open the **Add GPO Deployment Targets** dialog box.

4 Select a deployment target and click **Add**.

# 5 Troubleshooting

Log files help Open Text Technical Support to investigate and isolate the cause of an issue. You can adjust and collect various types of logs that include the following:

- Adjust Global Settings
- Adjust Customer Settings
- Adjust and Collect Gateway Logs
- Adjust and Collect Cloud Gateway Logs
- Adjust and Collect HTTP Call Logs

For detailed contact information, see the Support Contact Information website.

# A Appendix

Use this appendix to view the options you have for modifying built-in GPO settings, to view commands and lookups specific to the AD Bridge Linux Agent on Linux devices, and to understand review some best practices for AD Bridge.

## Linux Agent GPO Settings

Linux Agent GPO settings include rules for Firewall, Services, OpenSSH, Custom Configuration Files, and managing Active Directory logins. The Firewall settings include default Allow and Deny helper rules that you can configure, but you can also define custom Firewall rules.

Before deploying any configuration changes in your production environment, we strongly recommend that you first deploy GPOs in a Linux test environment to minimize the risk of introducing harmful Group Policy errors.

For examples of how to configure Linux Agent Settings in the GPO editor, see Managing Linux GPO Settings.

| Setting Type | Setting Name | Setting Data Type | Input Value (if any) |
|---|---|---|---|
| **Firewall** | All TCP | Allow/Deny | |
| | All UDP | Allow/Deny | |
| | SSH | Allow/Deny | |
| | HTTP | Allow/Deny | |
| | HTTPS | Allow/Deny | |
| | Samba | Allow/Deny | |
| | SMTP | Allow/Deny | |
| | MySQL | Allow/Deny | |
| | FTP | Allow/Deny | |
| **Services** | Start | String | |
| | Stop | String | |
| | Restart | String | |
| **Configuration Files**<br>◆ SSH<br>◆ Sudoers | | | |

| Setting Type | Setting Name | Setting Data Type | Input Value (if any) |
|---|---|---|---|
| **SSH** | Log Level | String Enum | QUIET, FATAL, ERROR, INFO, VERBOSE, DEBUG, DEBUG1, DEBUG2, DEBUG3 |
| | Set Login Grace Time | Integer | |
| | Set Client Alive Count Max | Integer | |
| | Use GSSAPI Authentication | String Enum | yes, no |
| | Use GSSAPI KeyExchange | String Enum | yes, no |
| | Use GSSAPI Cleanup Credentials | String Enum | yes, no |
| | Use Challenge Response Authentication | String Enum | yes, no |
| | Use PAM | String Enum | yes, no |
| | Use Password Authentication | String Enum | yes, no |
| | Allow Users | String | |
| | Deny Users | String | |
| | Deny Groups | String | |
| **Sudoers** | Number | Integer | |
| | Text | String | |
| | True /False | String Enum | |
| | Yes /No | String Enum | |
| **Deploy Files** | Source File | String | |
| **Execute Commands** | Command | String | |
| **AD Logins**<br>◆ On Premises<br>◆ Cloud | | | |
| **On Premises** | AD login provider mode | String Enum | add, simple, deny |
| | Allow these AD users to log in | String | |
| | Allow these AD groups to log in | String | |
| | Prevent these AD users from logging in | String | |
| | Prevent these AD groups from logging in | String | |

| Setting Type | Setting Name | Setting Data Type | Input Value (if any) |
|---|---|---|---|
| **Cloud** | Allow these AD groups to log in | String | |
| | Allow users matching this LDAP filter to log in | String | |

# Linux Agent Commands and Lookups

The items in this section contain useful Linux commands and lookups pertaining to the AD Bridge Linux Agent.

### Start the Linux Agent Service

If the Linux Agent Service is not running, use one of the following commands, applicable to your platform, to start the service:

- `systemctl start adb-agent.service`
- `service adb-agent start`

### Verify the Linux Agent Service is running

If you want to verify that the Linux Agent Service is running, use one of the following commands, applicable to your platform, to check the status:

- `systemctl status adb-agent.service`
- `service adb-agent status`

### Check for the Linux Agent version

If you need to know what version of the Linux Agent is installed on a given Linux device, access `/opt/adb-agent` and type a **tail** command for the `version` file to show the agent version.

For example:

1. `/opt/adb-agent`
2. `tail version`

### View the GPO Update Schedule

Installed Linux Agents are configured by default to run a pull from Active Directory every 60 minutes to check for any changes to Group Policy objects. This configuration is set in the `appsettings.json` file at `/opt/adb-agent` on the Linux Agent using the “`PullIntervalInMins`” element.

While this configuration can be changed, modifying this file is not recommended and may involve some risk.

# GPO Best Practices

Review the best practices in this section when working with Linux Agent GPOs in AD Bridge.

### Using the DenyUsers rule in SSH Configuration File settings

SSH DenyUsers for Active Directory user accounts should use ‘?’ in place of ‘@’. The ‘?’ in sshd is seen as a 1 character to 1 character wildcard. In some Linux platform’s the ‘@’ is used only as a Host identifier. Here is an example of the recommended DenyUser SSH Configuration File rule:

```
DenyUser user?domain.local
```

For more information about SSH and the sshd_conf, see https://en.wikibooks.org/wiki/OpenSSH.

### Removing GPO SSH Configuration File assignments from Linux agents

Due to the native behavior of working with configuration files in Linux platforms, you should remove the GPO from applicable Linux agents when removing a rule that you previously configured in the GPO. After removing the GPO, you can assign a new GPO if there are additional SSH settings that you require.

For example. If you have a **DenyUsers** SSH rule applied to one or more Linux agents using a Linux Agent GPO and you no longer require this exclusion, you will need to remove the GPO from applicable agents to clear the setting in the SSH configuration file. You can then apply a new GPO that does not have this rule configured.

### Understanding the AllowUsers rule in SSH Configuration File settings

When you add the **AllowUsers** rule in SSH Configuration File settings and apply it to one or more Linux agents, you should be aware that those users will be the only Active Directory users that will be able to login using SSH where the Linux agents have the GPO applied.