PingDirectory™

Release 2.0

Delegated Admin Application Guide

Notice

PingDirectory™ Product Documentation

Trademarks

Disclaimer

Support

https://support.pingidentity.com/

Contents

Chapter 1

Introduction to the Delegated Admin application

Topics:

Simple account administration can be performed by a group of administrators through the Delegated Admin application. This application manages identity attributes stored in the Directory Server.

Delegated Admin overview

Many organizations need to responsibly delegate some of the responsibility of managing identities in the Directory Server to reduce some of the burden of recurring tasks, such as password resets or updating account data. These simple administrative tasks can be delegated to a subset of administrators.

The Directory Server can be installed with the Delegated Admin application, which enables delegated administration of identities for scenarios such as help desk or customer service representatives (CSR) initiating a password reset and unlock; an HR administrator updating an employee profile; or an application administrator updating identity attributes.

Prerequisites

The Delegated Admin application manages user identity data stored in one or more Directory Servers. After a Directory Server instance is installed, the Delegated Admin application can be installed. You will need the HTTPS port that was configured during Directory Server setup.

A PingFederate server is used as the identity provider for authentication and authorization. Follow the PingFederate product documentation for installation and configuration instructions. Additional information is provided here for configuration with the Directory Server and the Delegated Admin application.

Chapter 2

Delegated Admin installation and configuration

Topics:

The Delegated Admin application relies on a previously installed PingDirectory Server and PingFederate Server. This section includes the configuration needed to support the Delegated Admin application.

Configuration overview

The Delegated Admin application must have a PingDirectory Server, and PingFederate Server installed. See the documentation for each product for installation instructions.

Configuration to support the Delegated Admin application on the PingDirectory Server includes:

- Configure administrators as Delegated Admin users.
- Configure attributes and attribute searching.

Configuration on the PingFederate Server includes:

- Configure PingFederate as the identity provider for the Delegated Admin application.
- Configure PingFederate as the OAuth server for the Delegated Admin application.
- Register the Delegated Admin application as a client.
- Register the PingDirectory Server as an OAuth token validator.

Authentication

The delegated administrator logs in to the Delegated Admin application through the PingFederate Server, which is configured as the authentication server and OpenID Connect (OIDC) provider. PingFederate validates the user credentials against the PingDirectory Server, and then issues an access token. The application obtains an access token from the PingFederate server, which encapsulates information `"claims"` about the user identity. The Delegated Admin application then presents this token to the Directory Server in the HTTP Authorization request header.

Interaction with the PingDirectory Server

The PingDirectory Server is configured to accept access tokens using Access Token Validators. The values that the PingFederate Server sets for access token `subject` claims must be mappable to a DN in the PingDirectory Server. Setting up an access token validator for use with the Delegated Admin application requires some coordination with the server configuration. In the suggested default configuration, the access token contains the entryUUID of the administrator user entry in the `sub` field. This value is mapped back to a PingDirectory Server entry using an Exact Match Identity Mapper.

Authorization by PingDirectory Server

Once validated, the PingDirectory Server checks the Delegated Admin configuration for authorization of the delegated administrator. Users or groups of users are authorized as delegated administrators in the Directory Server Administration Console, or with the `dsconfig` tool.

Client configuration

The Delegated Admin application uses PingFederate to authenticate users within the PingDirectory Server. Before installing and configuring the Delegated Admin application, you must configure two OAuth clients within PingFederate. One client is the Delegated Admin application, which will obtain an OIDC token that describes the authenticated user. The second client is the PingDirectory Server itself, which will call PingFederate to validate the OIDC token passed from the Delegated Admin application.

Specifically, the Delegated Admin OAuth client must be configured so that:

- The client id is "dadmin" and it requires no client secret key.
- The redirect URL is `https://SERVER:PORT/delegator/*`, where `SERVER:PORT` are the public host and port to access the Delegated Admin installation.
- The grant type is Implicit.
- The OIDC policy uses JWT tokens, where the entryUUID of the user is passed through the `subject` claim of the OIDC token.

The PingDirectory Server OAuth client must be configured so that:

- The client ID is "pingdirectory," and it does require a secret key.
- The grant type is Access Token Validation.

A detailed example of how PingFederate can be configured is available at the end of this guide.

Setup in a replicated PingDirectory Server environment

Running the Delegated Admin setup script requires special consideration in an environment that includes replicated PingDirectory Servers. If possible, setup the application after replication is enabled for the PingDirectory Servers. See the *PingDirectory Server Administration Guide* for details about server replication.

Set up Delegated Admin in an existing replicated topology

Complete the following steps if replication is already enabled for PingDirectory Servers.

1. If needed, configure the PingDirectory Servers to use a configuration group called "all-servers." This will ensure that configuration changes are applied to all servers in a topology.

```
$ bin/dsconfig set-global-configuration-prop \
  --set configuration-server-group:all-servers
```

2. Run the Delegated Admin setup script. See *Install the Delegated Admin application*.

Add an additional server to a replicated topology

To add an additional server to a topology containing a server already configured with the Delegated Admin application, perform the following the steps. In this example, "DS1" is the original PingDirectory Server, and "DS2" is the second server that will be added as a replica.

1. Run the `config-diff` command without arguments on DS1 to produce a batch file that contains configuration changes that will be applied to DS2.

```
$ bin/config-diff > config-changes.dsconfig
```

2. Apply the `config-changes.dsconfig` file to DS2.

```
$ bin/dsconfig --no-prompt \
  --batch-file config-changes.dsconfig \
  --applyChangeTo single-server
```

3. Restart DS2.
4. Enable replication between the two servers.

Install the Delegated Admin application

After the PingDirectory Server and the PingFederate Server have been configured, install the Delegated Admin application.

For systems such as Microsoft Windows that do not support bash scripts, copy the `delegated-admin-template.dsconfig` file and replace variables (`${variable}`) with actual values.

Note: If installing the Delegated Admin application on a PingDirectory Server that had "install with sample data" chosen as an installation option, the following ACI must be removed from the PingDirectory Server base DN:

```
(targetattr!="userPassword")(version 3.0; acl "Allow anonymous
```

```
read access for anyone"; allow (read,search,compare) userdn="ldap:///
anyone"
```

1. Download and unzip the Delegated Admin application package to a location on the PingDirectory Server.
2. Copy the Delegated Admin delegator folder and its contents into the PingDirectory Server's `webapps` directory.
3. In the Delegated Admin application directory, copy or rename the `example.config.js` file to `config.js`.
4. In an editor, open the `config.js` and change the values for the variables according to the local setup. The file contains comments and placeholders for necessary information. Save the changes.

   The client ID needed in this file is the one defined for the PingFederate configuration. In this file, it is the client intended for token issuance, such as `dadmin`.
5. In the PingDirectory Server root directory, run the following script to generate a `dsconfig` batch file:

   ```
   $ sh /webapps/delegator/set-up-delegator.sh
   ```

   The client ID and secret required for this script is the client configured for token verification on the PingFederate server, such as `pingdirectory`.
6. Apply the commands in the generated batch file to the PingDirectory Server. If installing in a replicated topology, add the `--applyChangeTo server-group` argument.

   ```
   $ ./bin/dsconfig \
     --bindDN "cn=Directory Manager" \
     --no-prompt \
     --batch-file webapps/delegator/delegated-admin.dsconfig -w
    directoryPassword
   ```
7. Go to `https://directoryHost:httpPort/delegator` to log in to the application.

Upgrade Delegated Admin application

If upgrading the Delegated Admin application from a previous version, unzip the new zip file and copy the delegator folder into the PingDirectory Server `webapps` folder, merging with the previous files and removing any files no longer in the new zip file. Make sure that the `config.js` file (which was customized during the first installation) is still present.

This version of the Delegated Admin application requires PingDirectory Server, version 7.0.1.

Install on PingDirectoryProxy Server

The Delegated Admin application can be installed on PingDirectoryProxy Server instances using the same PingDirectory Server installation instructions and setup script. Although it is not necessary to run the Delegated Admin application on the PingDirectory Server instances behind PingDirectoryProxy Server, all of the PingDirectory Server instances must have the following configuration elements exactly as they are configured on PingDirectoryProxy Server:

- The "Delegated Admin Privilege" virtual-attribute.
- The `delegated-admin-resource-type`, including all referenced `delegated-administrator` and `delegated-group-administrator` definitions.
- The global ACI "Authenticated access to the multi-update extended request for the Delegated Admin API."
- The global ACI "Authenticated access to the no-op request control for the Delegated Admin API."

The following configuration elements are not needed on the PingDirectory Server instances behind PingDirectoryProxy Server:

- The "Delegator" web-application-extension.
- The "PingFederateValidator" access-token-validator.

These sections must be removed from the script before the script is run on the backend PingDirectory Server instances.

Configure delegated administrators on the PingDirectory Server

To use the Delegated Admin application, an administrator must have more than valid credentials and an access token successfully validated by the PingDirectory Server. The administrator must be designated through the Directory Server configuration. To delegate users or groups as administrators, use the PingDirectory Server Administrative Console (Delegated Admin Resource Types), or the `dsconfig create-delegated-administrator` command.

The following sample commands illustrate configuration options for delegated administration, and are performed on the PingDirectory Server.

- The following command restricts an administrator to manage specified subtrees:

```
$ bin/dsconfig create-delegated-administrator \
  --type-name users \
  --administrator-name admin1 \
  --set "admin-user-dn:uid=admin1,ou=people,dc=example,dc=com" \
  --set admin-scope:manages-specific-subtrees \
  --set "managed-subtree:ou=org1,dc=example,dc=com" \
  --set enabled:true
```

- An administrator could be restricted to manage the member users of one or more specified groups. The following example assumes the existence of a static or dynamic group entry whose members include the users to be managed.

```
$ bin/dsconfig create-delegated-administrator \
  --type-name users \
  --administrator-name admin1 \
  --set "admin-user-dn:uid=admin1,ou=people,dc=example,dc=com" \
  --set admin-scope:manages-users-in-specific-groups \
  --set "managed-users-in-group:cn=User Group,dc=example,dc=com" \
  --set enabled:true
```

- Rather than delegate a single user as an administrator, it may be more convenient to delegate an entire group of users as administrators:

```
$ bin/dsconfig create-delegated-administrator \
  --type-name users \
  --administrator-name admin-group1 \
  --set "admin-group-dn:cn=Admin Group,ou=people,dc=example,dc=com" \
  --set admin-scope:manages-all-entries \
  --set enabled:true
```

  In this example, groups can be configured to use manage specific subtrees or groups with the `manages-specific-subtrees` or `manages-users-in-specific-groups` settings for the `admin-scope`. For more information about PingDirectory Server administrators and configuring dynamic and static groups, see the *PingDirectory Server Administration Guide*.

Configure attributes and attribute search on PingDirectory Server

The file used to install the Delegated Admin application specifies the object class of user entries through `structural-ldap-objectclass:inetOrgPerson`, and specifies a number of user attributes to be exposed by the application.

1. If needed, the attribute designated as the primary display attribute can be changed.

```
$ bin/dsconfig set-delegated-admin-resource-type-prop \
```

```
  --type-name users \
  --set primary-display-attribute-type:mail
```

2. Configure any additional user attributes to display in the Delegated Admin application by specifying the LDAP attribute type to be exposed and providing a display name for it.

```
$ bin/dsconfig create-delegated-admin-attribute \
  --type-name users \
  --attribute-type customAttr
  --set "display-name:My custom attribute"
```

3. When search text is entered in the application, the attributes to be searched in the PingDirectory Server are specified by the `search-filter-pattern` property. The PingDirectory Server must have appropriate attribute indices defined to satisfy the query. Use the following command to set the search filter.

```
$ bin/dsconfig set-delegated-admin-resource-type-prop \
  --type-name users \
  --set 'search-filter-pattern:(|(cn=*%%*)(mail=%%*)(uid=%%*))'
```

Manage groups

Group administrators can be configured to manage the membership of static groups that exist in the PingDirectory Server. The Delegated Admin application cannot create or delete these groups.

Use the following command to delegate a user as a group administrator:

```
$ bin/dsconfig create-delegated-group-administrator \
 --type-name users \
 --administrator-name group-admin1
 --set enabled:true \
 --set "admin-user-dn:uid=admin1,ou=people,dc=example,dc=com" \
 --set user-scope:users-in-specific-subtrees \
 --set "user-subtree:ou=org1,dc=example,dc=com"  \
 --set "manage-membership-of-group:cn=Employees,dc=example,dc=com" \
 --set "manage-membership-of-group:cn=Contractors,dc=example,dc=com"
```

The user scope determines which users are visible to this group administrator. In this example, all users in the subtree `ou=org1,dc=example,dc=com` are visable. An administrator can be configured as both a delegated administrator able to edit users, and also a delegated group administrator to manage group membership and view users.

The group administrator is able to view, and add or remove any of the users within the `user-scope` to the membership of groups specified by `manage-membership-of-group`. Static groups can be nested. Users belonging indirectly to a group through nesting, are visible as group members but cannot be removed. Users can only be removed from the group in which they are a member. For example, an Employees group may include a Developers group as a nested member. A user in the Developers group is a direct member of that group, and is also an indirect member of Employees. This member can only be removed when viewing the Developers group, not when viewing the Employees group.

If `manage-membership-of-group` references a dynamic group or virtual static group, rather than a static group, then that group and its members will be visible, but the group membership cannot be modified.

Set group attributes

The default settings for group attributes specify `cn` and `description` as group attributes, with `cn` used for the group title in the Delegated Admin application. The equivalent commands to create the default settings are:

```
$ bin/dsconfig create-delegated-admin-group-attribute \
  --type-name users \
  --attribute-type cn \
```

```
  --set "display-name:Name"
```

```
$ bin/dsconfig dsconfig create-delegated-admin-group-attribute \
  --type-name users \
  --attribute-type description \
  --set "display-name:Description"
```

```
$ bin/dsconfig set-delegated-admin-resource-type-prop \
  --type-name users \
  --set group-title-attribute-type:cn
```

Set group search filter

When entering search text to search for groups, the attributes to be searched in the PingDirectory Server are specified by the `group-search-filter-pattern` property. The PingDirectory Server must have appropriate attribute indices defined to satisfy the query. The default setting searches the `cn` attribute for the search text (represented by % %). Use the following command to set the group search filter:

```
$ bin/dsconfig set-delegated-admin-resource-type-prop \
  --type-name users \
  --set 'group-search-filter-pattern:(cn=*%%*)'
```

Enable log tracing

Log tracing can be enabled for OAuth token processing, HTTP request and response actions, and API debugging.

To see OAuth token processing and full HTTP request/response tracing, enable the debug trace logger as follows:

```
$ bin/dsconfig set-log-publisher-prop \
  --publisher-name 'Debug Trace Logger' \
  --set enabled:true
```

To enable `≈dadmin` API debug logging, use the following commands:

```
$ bin/dsconfig create-debug-target \
  --publisher-name 'File-Based Debug Logger' \
  --target-name com.unboundid.directory.server.http \
  --set debug-level:VERBOSE
```

```
$ bin/dsconfig create-debug-target \
   --publisher-name 'File-Based Debug Logger' \
   --target-name com.unboundid.directory.server.extensions.dadmin \
   --set debug-level:VERBOSE
```

```
$ bin/dsconfig create-debug-target \
  --publisher-name 'File-Based Debug Logger' \
  --target-name com.unboundid.directory.broker.api \
  --set debug-level:VERBOSE
```

```
$ bin/dsconfig set-log-publisher-prop \
  --publisher-name 'File-Based Debug Logger' \
  --set enabled:true
```

Appendix A

Sample PingFederate configuration

Topics:

PingFederate offers many configuration options. This section provides a sample configuration that can be used to support the Delegated Admin application.

PingFederate sample configuration

The following is a sample PingFederate configuration. Minimal support for the Delegated Admin applications includes configuring PingFederate to use the HTML Form Adapter to authenticate users for the Delegated Admin application, and configuring PingFederate to identify an authenticated user through the user's entryUUID, which is mapped to the `subject` of the OIDC token.

Configure PingFederate as the Identity Provider

This procedure configures the PingFederate Server as the identity provider for PingDirectory Server.

Note: Before starting, download the LDAPS certificate from PingDirectory Server. All other steps are performed on the PingFederate server. See the *PingDirectory Server Administration Guide* for details.

1. Add roles for OAuth and OIDC and Identity Provider under **Server Configuration > Server Settings > Roles and Protocols**.
2. Upload the PingDirectory Server LDAPS certificate in **Server Configuration > Trusted CAs**.
3. Add an LDAP data store in **Server Configuration > Data Stores**. Specify:
   a) The PingDirectory Server hostname and LDAPS port.
   b) Select **Use LDAPS**.
   c) Under **Advanced**, clear the **Verify LDAPS hostname** option.
4. Create the HTML form **IdP Adapter and Password Credential Validator** that is used to authenticate users against PingDirectory Server:
   a) Select **Identity Provider > Adapters > Create New Instance**.
   b) Select the HTML Form type.
   c) Scroll to bottom, and click **Manage Password Credential Validators**.
   d) Select **Create New Instance**.
   e) Select the LDAP Username Password Credential Validator.
   f) To use either email address or username to log in, enter the following search filter:

   ```
   (|(uid=${username})(mail=${username}))
   ```

   g) Extend the contract with "entryUUID" and "cn". These are used later.
   h) Click **Next** and **Save** until the Create Adapter Instance screen.
   i) Add a new row to Password Credential Validators, and choose the new LDAP Password Credential Validator.
   j) Extend the contract with "entryUUID" and "cn". These are used later.
   k) Check "entryUUID" for pseudonym, and **Save**.
5. Enable session tracking in **Identity Provider > Sessions**, and select the **Track adapter session for logout** and **Enable authentication sessions for all sources** options.
6. Click **Save**.

Configure the OAuth server

1. Click **IdP Adapter Mapping**, and add the new IdP adapter for creating OAuth grants. An additional attribute source is not needed. Fulfill the contract with the USER_KEY from Adapter "entryUUID" and USER_NAME from Adapter "cn", and click **Save**.
2. Click **Access Token Management > Create New Instance**. In this step, JSON Web Tokens are configured as an example:
   a) Select **JSON Web Tokens**.
   b) Choose one-way encryption for JWT, which only requires a symmetric key (not a certificate and private key). This requires the client to validate the token by hitting the validation endpoint on the server.
   c) Add a row to symmetric keys and use 32 bytes or 64 chars of hex.
   d) Choose the JWS Algorithm HMAC using SHA-256.
   e) Choose your symmetric key for Active Symmetric Key ID, and click **Next**.

f) Check all options.
g) List at least one attribute to be defined in the access token. Add `sub`, and click **Save**.

3. Click **Access Token Mapping** and map the access token attributes from the persistent grant.
   a) Choose **Default Context** and the new Access Token Manager.
   b) Map the `sub` claim from the USER_KEY of the persistent grant, which is mapped from the entryUUID.
4. Click **OpenID Connect Policy Management > Add Policy**.
   a) Choose the previously created Access Token Manager.
   b) Delete all of the extended contract attributes, except `sub`. Other scopes defined, if configured.
   c) Click **Next** to reach Contract Fulfillment.
   d) Fulfill the OIDC contract `sub` with the Access Token attribute `sub`.
   e) If a default OIDC policy is not already defined, set this new policy as the default, and click **Save**.
5. Add scopes for PingDirectory Server APIs.
   a) Click **Scope Management > Exclusive Scopes**.
   b) Add a value and description for `urn:pingidentity:directory-delegated-admin`.
   c) Click **Save**.

Configure the Delegated Admin application and PingDirectory Server as OAuth clients

The following steps configure PingDirectory Server as the token validator, and the Delegated Admin application as a new client.

1. Click **Create new client**, and set the client ID and name as PingDirectory Server.
   a) Use client secret for authentication and generate a new secret. Copy the key.
   b) Check **Access Token Validation** under Allowed Grant Types, and click **Save**.

      When creating a PingFederate Access Token Validator in PingDirectory Server, use the pingdirectory client ID and secret. An Identity Mapper in will also be used in the PingDirectory Server that will take the `sub` claim and match it against the entryUUID attribute.
2. Click **Create new client**, and set the client ID and name as dadmin.
   a) Do not configure authentication.
   b) Define the redirect URI as **"**`https://${directoryServer:port}/delegator/*`**"**, with the host and port of the PingDirectory Server.
   c) Check **Bypass Authorization Approval**, and choose the Implicit Grant type.
   d) Check **Allow Exclusive Scopes** and check `urn:pingidentity:directory-delegated-admin`.
   e) Check **Implicit** grant type.
   f) Choose the OIDC policy previously created, and click **Save**.
   g) Add **"**`https://${directoryServer:port}`**"** to the CORS origins under **OAuth Server > Authorization Server Settings > Cross-Origin Settings**.

Index