ONE IDENTITY™

One Identity Safeguard for Privileged Sessions 5.8

YubiKey Multi-Factor Authentication - Tutorial

SPS YubiKey Multi-Factor Authentication - Tutorial
Updated - August 2018
Version - 5.8

Contents

Introduction

This document describes how you can use the services of Yubico (YubiKey) to authenticate the sessions of your privileged users with One Identity Safeguard for Privileged Sessions (SPS).

One Identity Safeguard for Privileged Sessions:

One Identity Safeguard for Privileged Sessions (SPS) controls privileged access to remote IT systems, records activities in searchable, movie-like audit trails, and prevents malicious actions. SPS is a quickly deployable enterprise device, completely independent from clients and servers — integrating seamlessly into existing networks. It captures the activity data necessary for user profiling and enables full user session drill down for forensic investigations.

SPS acts as a central authentication gateway, enforcing strong authentication before users access sensitive IT assets. SPS can integrate with remote user directories to resolve the group memberships of users who access nonpublic information. Credentials for accessing information systems can be retrieved transparently from SPS's local credential store or a third-party password management system. This method protects the confidentiality of passwords as users can never access them. When used together with YubiKey (or another multi-factor authentication provider), SPS directs all connections to the authentication tool, and upon successful authentication, it permits the user to access the information system.

Integrating YubiKey with SPS:

SPS can interact with your YubiKey account and can automatically request strong multi-factor authentication for your privileged users who are accessing the servers and services protected by PSM. When used together with YubiKey, SPS directs all connections to the YubiKey tool, and upon successful authentication, it permits the user to access the information system.

The integration adds an additional security layer to the gateway authentication performed on SPS. YubiKey 4, YubiKey 4 Nano, and YubiKey NEO devices are pre-configured with the Yubico one-time password (OTP) (all other YubiKeys, except for the FIDO U2F Security Key by Yubico, also support Yubico OTP). The OTP will be used for authentication to the One Identity platform. This way, the device turns into a two-factor authentication token for the user. The one-time password is changed after every authentication and is generated using dynamic keys.

Meet compliance requirements

ISO 27001, ISO 27018, SOC 2, and other regulations and industry standards include authentication-related requirements, for example, multi-factor authentication (MFA) for accessing production systems, and the logging of all administrative sessions. In addition to other requirements, using SPS and YubiKey helps you comply with the following requirements:

- PCI DSS 8.3: Secure all individual non-console administrative access and all remote access to the cardholder data environment (CDE) using multi-factor authentication.

- PART 500.12 Multi-Factor Authentication: Covered entities are required to apply multi-factor authentication for:

    - Each individual accessing the covered entity's internal systems.

    - Authorized access to database servers that allow access to nonpublic information.

    - Third parties accessing nonpublic information.

- NIST 800-53 IA-2, Identification and Authentication, network access to privileged accounts: The information system implements multi-factor authentication for network access to privileged accounts.

- The General Data Protection Regulation (GDPR) goes into effect on May 25, 2018, and is applicable to organizations keeping Personally identifiable information (PII) and offering goods or services to individuals based in the EU.

    YubiKey provides strong authentication to secure access to PII and comply with GDPR.

- The Defense FAR Supplement (DFARS) clause went into effect on December 31, 2017, and is applicable to US Department of Defense (DoD) contractors to protect unclassified DoD information and minimize loss of information.

    The multi-protocol YubiKey meets DFARS requirements for strong authentication, and is the only hardware authentication solution to meet DoD contractor security requirements.

- The revised Directive on Payment Services (PSD2) provides recommendations on standardized access to customer data and banking infrastructure, including draft regulatory technical standards specifying the requirements of strong customer authentication (SCA).

    Yubico and FIDO are playing active roles in the PSD2 framework with proven technology.

How SPS and YubiKey MFA work together

Figure 1: How SPS and YubiKey work together
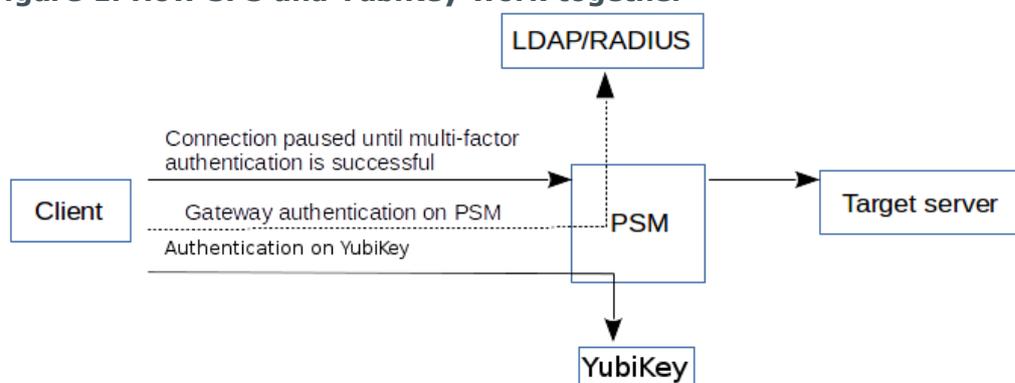
1. A user attempts to log in to a protected server.

2. **Gateway authentication on SPS**

   SPS receives the connection request and authenticates the user. SPS can authenticate the user to a number of external user directories, for example, LDAP, Microsoft Active Directory, or RADIUS. This authentication is the first factor.

3. **Outband authentication on YubiKey**

   If gateway authentication is successful, SPS connects the YubiKey server to check which authentication factors are available for the user. Then SPS requests the second authentication factor from the user. SPS supports authentication factors that are based on keyboard interaction such as Yubico-OTP, OATH-OTP, OATH-HOTP, and OATH-TOTP. For details on these authentication factors, see What is OATH?.

   For OTP-like authentication factors, SPS requests the one-time password (OTP) from the user, and sends it to the YubiKey Validation Service for verification either running on premise or using the YubiCloud Validation Service.

4. If multi-factor authentication is successful, the user can start working, while SPS

records the user's activities. (Optionally, SPS can retrieve credentials from a local or external credential store or password vault, and perform authentication on the server with credentials that are not known to the user.)

Technical requirements

In order to successfully connect SPS with YubiKey, you need the following components.

In YubiKey:

- The users must have a YubiKey device and a means to map usernames to YubiKey Public IDs. For details, see "[users]" in the YubiKey Multi-Factor Authentication - Tutorial and "[ldap]" in the YubiKey Multi-Factor Authentication - Tutorial.

- The YubiKey Client ID and API Key.

  For details on generating your Client ID and API Key, see How do I get an API key for YubiKey development?.

  To generate your Client ID and API Key, authenticate yourself using a Yubikey One-Time Password and provide your e-mail address as a reference at Yubico get API key.

  A Yubico OTP is a 44-character, one-use, secure, 128-bit encrypted Public ID and Password. The OTP is comprised of two major parts: the first 12 characters remain constant and represent the Public ID of the YubiKey device itself. The remaining 32 characters make up a unique passcode for each OTP generated.

  For example, in the following Yubico OTP, the characters **cccjgjgkhcbb** are the Public ID, and the remaining characters are the passcode.

  *cccjgjgkhcbb*irdrfdnlnghhfgrtnnlgedjlftrbdeut

- YubiKey does not require network connectivity or access to a mobile phone device. Just touch or tap the YubiKey device to authenticate.

In SPS:

- A One Identity Safeguard for Privileged Sessions appliance (virtual or physical), at least version 5 F1.

- A copy of the SPS YubiKey plugin. This plugin is an Authentication and Authorization (AA) plugin customized to work with the YubiKey multi-factor authentication service.

- SPS must be able to access the validation service.

  The connection also requires the Client ID and API Key.

- SPS supports Authentication and Authorization plugins in the RDP, SSH, and Telnet protocols.

- In RDP, using an **AA plugin** together with Network Level Authentication in a Connection Policy has the same limitations as using Network Level Authentication without domain membership. For details, see "Network Level Authentication without domain membership" in the Administration Guide.

- In RDP, using an **AA plugin** requires TLS-encrypted RDP connections. For details, see "Enabling TLS-encryption for RDP connections" in the Administration Guide.

Availability and support of the plugin

The SPS YubiKey plugin is available as-is, free of charge to every SPS customer from the Plugin Page. In case you need any customizations or additional features, contact professionalservices@balabit.com.

You can use the plugin on SPS 5 F5 and later. If you need to use the plugin on SPS 5 LTS, contact professionalservices@balabit.com.

How SPS and YubiKey work together in detail

1. A user attempts to log in to a protected server.

2. **Gateway authentication on SPS**

   SPS receives the connection request and authenticates the user. SPS can authenticate the user to a number of external user directories, for example, LDAP, Microsoft Active Directory, or RADIUS. This authentication is the first factor.

3. **Check if the user is exempt from multi-factor authentication**

   You can configure SPS using whitelists and blacklists to selectively require multi-factor authentication for your users, for example, to create break-glass access for specific users.

   - If multi-factor authentication is not required, the user can start working, while SPS records the user's activities. The procedure ends here.
   - If multi-factor authentication is required, SPS continues the procedure with the next step.

   For details on creating exemption lists, see SPS YubiKey plugin parameter reference.

4. **Determining the YubiKey username**

   If the gateway usernames are different from the YubiKey usernames, you must configure the SPS YubiKey plugin to map the gateway usernames to the YubiKey usernames. The mapping can be as simple as appending a domain name to the gateway username, or you can query an LDAP or Microsoft Active Directory server. For details, see Mapping SPS usernames to YubiKey identities.

5. **Outband authentication on YubiKey**

   If gateway authentication is successful, SPS connects the YubiKey server to check which authentication factors are available for the user. Then SPS requests the second authentication factor from the user.

For OTP-like authentication factors, SPS requests the OTP from the user, and sends it to the YubiKey server for verification.

6. If multi-factor authentication is successful, the user can start working, while SPS records the user's activities. (Optionally, SPS can retrieve credentials from a local or external credential store or password vault, and perform authentication on the server with credentials that are not known to the user.)

7. If the user opens a new session within a short period, they can do so without having to perform multi-factor authentication again. After this configurable grace period expires, the user must perform multi-factor authentication to open the next session. For details, see SPS YubiKey plugin parameter reference.

Mapping SPS usernames to YubiKey identities

By default, SPS assumes that the YubiKey username of the user is the same as the gateway username (that is, the username the user used to authenticate on SPS during the gateway authentication). To identify the users, SPS uses the username (login) field in YubiKey, which is an email address.

If the gateway usernames are different from the YubiKey usernames, you must configure the SPS YubiKey plugin to map the gateway usernames to the YubiKey usernames. You can use the following methods:

- To simply append a string to the gateway username, configure the append_domain parameter. In this case, SPS automatically appends the @ character and the value of this option to the username from the session, and uses the resulting username on the YubiKey server to authenticate the user. For example, if the domain is set as append_ domain: example.com and the username is Example.User, the SPS plugin will look for the user Example.User@example.com on the YubiKey server.

- To look up the YubiKey username of the user from an LDAP/Active Directory database, configure the [ldap] section of the SPS YubiKey plugin. Typically, the SPS plugin queries the email address corresponding to the username from your LDAP or Active Directory database. For details on LDAP parameters, see SPS YubiKey plugin parameter reference.

- If you configure both the append_domain parameter and the [ldap] section of the SPS YubiKey plugin, SPS appends the @ character and the value of the append_domain parameter to the value retrieved from the LDAP database.

- If you have configured neither the Domain parameter nor the [ldap] section, SPS assumes that the YubiKey username of the user is the same as the gateway username.

Bypassing YubiKey authentication

Having to perform multi-factor authentication to a remote server every time the user opens a session can be tedious and inconvenient for the users, and can impact their productivity. SPS offers the following methods to solve this problem:

- In SPS, the Connection policy determines the type of authentication required to access a server. If you do not need multi-factor authentication for accessing specific servers, configure your Connection policies accordingly.

- If the user opens a new session within a short period, they can do so without having to perform multi-factor authentication. After this configurable grace period expires, the user must perform multi-factor authentication to open the next session. For details, see SPS YubiKey plugin parameter reference.

- You can configure SPS using whitelists and blacklists to selectively require multi-factor authentication for your users, for example, to create break-glass access for specific users. For details on creating exemption lists, see SPS YubiKey plugin parameter reference.

Configure your YubiKey account for SPS

Prerequisites:

The users must have a YubiKey device and a means to map usernames to YubiKey Public IDs. For details, see SPS YubiKey plugin parameter reference and SPS YubiKey plugin parameter reference.

Steps:

1. Generate the YubiKey Client ID and API Key.

   For details on generating your Client ID and API Key, see How do I get an API key for YubiKey development?.

   To generate your Client ID and API Key, authenticate yourself using a Yubikey One-Time Password and provide your e-mail address as a reference at Yubico get API key.

   A Yubico OTP is a 44-character, one-use, secure, 128-bit encrypted Public ID and Password. The OTP is comprised of two major parts: the first 12 characters remain constant and represent the Public ID of the YubiKey device itself. The remaining 32 characters make up a unique passcode for each OTP generated.

   For example, in the following Yubico OTP, the characters **cccjgjgkhcbb** are the Public ID, and the remaining characters are the passcode.

   *cccjgjgkhcbb*irdrfdnlnghhfgrtnnlgedjlftrbdeut

2. YubiKey does not require network connectivity or access to a mobile phone device. Just touch or tap the YubiKey device to authenticate.

Configure SPS to use YubiKey multi-factor authentication

Prerequisites:

- Your YubiKey Client ID and API Key.

  > ⚠ **CAUTION:**
  >
  > **According to the current YubiKey policies, your API token expires if it is not used for 30 days. Make sure that you use it regularly, because SPS will reject your sessions if the API token is expired.**

- Administrator access to SPS.
- Make sure that you have all the required components listed in Technical requirements.

Steps:

1. **Download the SPS YubiKey plugin**

   SPS customers can download the plugin from Plugin Page.

2. **Upload the plugin to SPS**

   Upload the plugin to SPS. For details, see Administration Guide.

3. **Configure the plugin on SPS**

   The plugin includes a default configuration file, which is an ini-style configuration file with sections and name=value pairs. You can edit it on the **Policies > AA Plugin Configurations** page of the SPS web interface.

   a. Configure the usermapping settings if needed. SPS must find out which YubiKey user belongs to the username of the authenticated connection. For that, it can query your LDAP/Microsoft Active Directory server. For details, see Mapping SPS usernames to YubiKey identities.

b. Configure other parameters of your plugin as needed for your environment. For details, see SPS YubiKey plugin parameter reference.

4. **Configure a Connection policy and test it**

Configure a Connection policy on SPS. In the **AA plugin** field of the Connection policy, select the SPS YubiKey plugin you configured in the previous step, then start a session to test it. For details on how a user can perform multi-factor authentication, see Perform multi-factor authentication with the SPS YubiKey plugin in terminal connections and Perform multi-factor authentication with the SPS YubiKey plugin in Remote Desktop connections.

⚠ **CAUTION:**

According to the current YubiKey policies, your API token expires if it is not used for 30 days. Make sure that you use it regularly, because SPS will reject your sessions if the API token is expired.

SPS YubiKey plugin parameter reference

This section describes the available options of the SPS YubiKey plugin.

The plugin uses an ini-style configuration file with sections and name=value pairs. This format consists of sections, led by a [section] header and followed by name=value entries. Note that the leading whitespace is removed from values. The values can contain format strings, which refer to other values in the same section. For example, the following section would resolve the %(dir)s value to the value of the dir entry (/var in this case).

```
[section  name]
 dirname=%(dir)s/mydirectory
 dir=/var
```

All reference expansions are done on demand. Lines beginning with # or ; are ignored and may be used to provide comments.

You can edit the configuration file from the SPS web interface. The following code snippet is a sample configuration file.

```
[yubikey]
client_id=<YubiKey-Client-ID>
# Do NOT use api_key in production
; api_key=<YubiKey-API-key>
api_url=<API-URL1,API-URL2>
timeout=10

[users]
<exampleuser1>=abcdefghijkl
<exampleuser2>=mnopqrstuvwy

[plugin]
config_version=1
log_level=info
cred_store=<name-of-credstore-storing-sensitive-data>

[auth]
prompt=Enter your YubiKey one-time-password:
whitelist=name-of-a-userlist
```

```
[username_transform]
append_domain=""

[ldap]
ldap_server_config=<SPS-LDAP-server-policy-name>
filter=(&(samAccountName={})(objectClass=user))
user_attribute=mail

[cache]
soft_timeout=15
hard_timeout=90
conn_limit=5

[question_1]
key=<name-of-name-value-pair>
prompt=<the-question-itself-in-text>
disable_echo=No

[question_2]...
```

[yubikey]

This section contains the options related to your YubiKey account.

```
[yubikey]
client_id=<API-integration-key>
# Do NOT use api_key in production
; api_key=<API-security-key>
api_url=<API-URL>
timeout=10
```

client_id

Type:	string
Required:	yes
Default:	N/A

Description: Your YubiKey Client ID (also known as AuthID or API ID). For details on generating your Client ID and API Key, see How do I get an API key for YubiKey development?.

To generate your Client ID and API Key, authenticate yourself using a Yubikey One-Time Password and provide your e-mail address as a reference at Yubico get API key.

api_key

Type:	string
Required:	no
Default:	N/A

> ⚠ **CAUTION:**
>
> **This parameter contains sensitive data. Make sure to store this data in your local credential store. Never use it in production.**
>
> **For details, see "Store sensitive plugin data securely".**
>
> **Only use this parameter in the configuration for testing purposes in a secure, non-production environment.**

Description: Your YubiKey API key. SPS uses this to communicate with the YubiKey server. For details on using a local Credential Store to host this data, read Store sensitive plugin data securely.

For details on generating your Client ID and API Key, see How do I get an API key for YubiKey development?.

To generate your Client ID and API Key, authenticate yourself using a Yubikey One-Time Password and provide your e-mail address as a reference at Yubico get API key.

> ⚠ **CAUTION:**
>
> **According to the current YubiKey policies, your API token expires if it is not used for 30 days. Make sure that you use it regularly, because SPS will reject your sessions if the API token is expired.**

api_url

Type:	string
Required:	yes
Default:	N/A

Description: The default configuration uses Yubico's Cloud validation servers. You can also host your own YubiKey validation server. For details on hosting your own server, see YubiKey OTP Validation Server. If you decide to host your own, use this option to enter a comma-separated list of URLs that point to a YubiKey validation service. Multiple URLs will be attempted in parallel, the first response wins.

timeout

Type:	integer [seconds]
Required:	no
Default:	10

Description: How long the YubiKey server waits to respond.

[users]

This section contains user-YubiKey pairs.

```
[users]
<exampleuser1>=abcdefghijkl
<exampleuser2>=mnopqrstuvwy
```

<exampleuser>

Type:	integer [seconds]
Required:	no
Default:	10

Description: To pair YubiKeys with users, you have three options:

- Retrieve it from userattribute through LDAP/AD.
- Define a `[users]` section in the configuration file using the `user=deviceid` format.
- Store the the user/device mapping in a credential store with the usual syntax: `host=users, user=exampleuser, password=deviceid`.

Use the second (`[users]` section) option only if there are not too many users, or for testing purposes. If there are too many users, it can cause performance issues.

[plugin]

This section contains general plugin-related settings.

```
[plugin]
config_version=1
log_level=20
cred_store=<name-of-credstore-hosting-sensitive-data>
```

config_version

Type:	integer
Required:	yes
Default:	1

Description: The version number of the configuration format. This is used to enable potentially incompatible changes in the future. If provided, the configuration will not be upgraded automatically. If not provided, the configuration will be upgraded automatically.

cred_store

Type:	string
Required:	no
Default:	N/A

Description: The name of a local credential store policy configured on SPS. You can use this credential store to store sensitive information of the plugin in a secure way, for example, the ikey/skey values in the [yubikey] section. For details, see Store sensitive plugin data securely.

log_level

Type:	integer or string
Required:	no
Default:	info

Description: The logging verbosity of the plugin. The plugin sends the generated log messages to the SPS syslog system. You can check the log messages in the **Basic settings > Troubleshooting > View log files** section of the SPS web interface. Filter on the plugin: string to show only the messages generated by the plugins.

The possible values are:

- debug or 10
- info or 20
- warning or 30
- error or 40
- critical or 50

For details, see Python logging API's log levels: Logging Levels.

[auth]

This section contains the options related to authentication.

```
[auth]
prompt=Enter your YubiKey one-time password:
whitelist=name-of-a-userlist
```

prompt

Type:	string
Required:	no
Default:	Hit Enter to send push notification or provide the OTP:

Description: SPS displays this text to the user in a terminal connection to request an OTP interactively. The text is displayed only if the user uses an OTP-like factor, and does not send the OTP in the connection request.

```
prompt="Enter your YubiKey one-time password:"
```

whitelist

Type:	string
Required:	no
Default:	N/A

Description: The name of a user list configured on SPS (**Policies > User Lists**). You can use this option to selectively require multi-factor authentication for your users, for example, to create break-glass access for specific users.

- If you set the **Default Policy** of the user list to Reject, then the list is a whitelist, so the plugin will not request YubiKey authentication from the users on the list.

- If you set the **Default Policy** of the user list to Accept, then the list is a blacklist, so the plugin will request YubiKey authentication only from the users on the list.

For details on creating user lists, see "Creating and editing user lists" in the Administration Guide.

[cache]

This section contains the settings that determine how soon after performing a YubiKey authentication must the user repeat the authentication when opening a new session.

After the first YubiKey authentication of the user, SPS will not request a new YubiKey authentication from the user as long as the new authentications would happen within `soft_timeout` seconds from each other. After the `hard_timeout` expires (measured from the first YubiKey login of the user), SPS will request a new YubiKey authentication.

In other words, after opening the first session and authenticating on YubiKey, the user can keep opening other sessions without having to authenticate again on YubiKey as long as the time between opening any two sessions is less than `soft_timeout`, but must authenticate on YubiKey if `hard_timeout` expires.
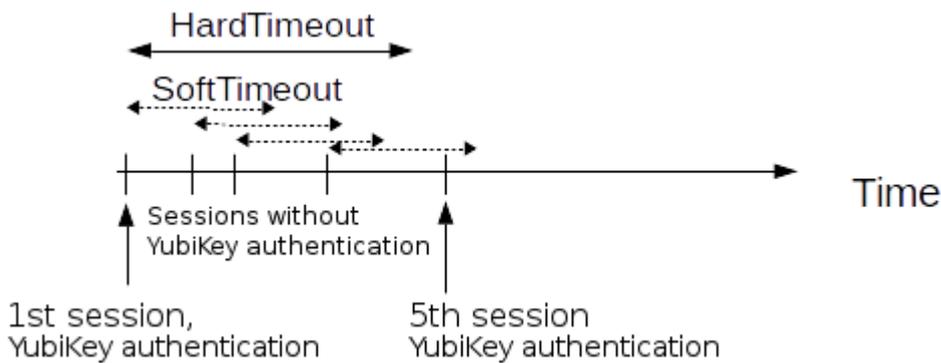

```
[cache]
soft_timeout=15
hard_timeout=90
conn_limit=5
```

soft_timeout

Type:	integer [seconds]
Required:	yes, if you want caching
Default:	N/A

Description: The time in seconds after which the SPS plugin requires a new YubiKey authentication for the next new session of the user, unless the user successfully authenticates another session within this period.

hard_timeout

Type:	integer [seconds]
Required:	yes, if you want caching
Default:	N/A

Description: The time in seconds after which the SPS plugin requires a new YubiKey authentication for the next new session of the user. The time is measured from the last YubiKey authentication of the user.

conn_limit

Type: integer [number of]

Description: The cache can be used `conn_limit` times without multi-factor authentication. If the number of logins exceeds this number, the plugin will request multi-factor authentication again. If this parameter is not set, the number of logins from cache are unlimited.

[ldap]

This section contains the settings you configure when you need to use an LDAP query to map the usernames from your audited sessions to the usernames in YubiKey.

To look up the YubiKey username of the user from an LDAP/Active Directory database, configure the [ldap] section of the SPS YubiKey plugin. Typically, the SPS plugin queries the email address corresponding to the username from your LDAP or Active Directory database. For details on LDAP parameters, see SPS YubiKey plugin parameter reference.

If you configure both the append_domain parameter and the [ldap] section of the SPS YubiKey plugin, SPS appends the @ character and the value of the append_domain parameter to the value retrieved from the LDAP database.

For other methods of mapping gateway usernames to YubiKey usernames, see Mapping SPS usernames to YubiKey identities.

```
[ldap]
ldap_server_config=<SPS-LDAP-server-policy-name>
filter=(&(cn={})(objectClass=inetOrgPerson))
user_attribute=CN
```

ldap_server_config

Type:	string
Required:	no
Default:	N/A

Description: The name of a configured LDAP server policy in SPS. For details on configuring LDAP policies, see "Authenticating users to an LDAP server" in the Administration Guide.

filter

Type: string

Required:	no
Default:	(&(cn={})(objectClass=inetOrgPerson))

Description: The LDAP filter query that locates the user based on the gateway username. The plugin automatically replaces the {} characters with the gateway username from the session.

```
filter=(&(cn={})(objectClass=inetOrgPerson))
```

user_attribute

Type:	string
Required:	no
Default:	cn

Description: The name of the LDAP attribute that contains the YubiKey username.

[username_transform]

This section contains username transformation-related settings.

```
[username_transform]
append_domain=""
```

append_domain

Type:	string (nonrequired, no default)
Required:	no
Default:	N/A

Description: If the gateway usernames are different from the YubiKey usernames, you must configure the SPS YubiKey plugin to map the gateway usernames to the YubiKey usernames.

To simply append a string to the gateway username, configure the append_domain parameter. In this case, SPS automatically appends the @ character and the value of this option to the username from the session, and uses the resulting username on the YubiKey server to authenticate the user. For example, if the domain is set as append_domain: example.com and the username is Example.User, the SPS plugin will look for the user Example.User@example.com on the YubiKey server.

If you configure both the append_domain parameter and the [ldap] section of the SPS YubiKey plugin, SPS appends the @ character and the value of the append_domain parameter to the value retrieved from the LDAP database.

For other methods of mapping gateway usernames to YubiKey usernames, see Mapping SPS usernames to YubiKey identities.

[question_1]

Type:	integer [seconds]

Description: Used for communication between plugins. This is an interactive request/response right after authentication in order to supply data to credential store plugins. The question is transferred to the session cookie and all hooks of all plugins receive it.

For example, if you have an external authenticator app, you do not have to wait for the question to be prompted but can authenticate with a one-time password:

ssh otp=123456@root@scb

Name subsequent questions with the appropriate number, for example, [question_1], [question_2], and so on.

For details, see "Performing authentication with AA plugin in terminal connections" in the Administration Guide and "Performing authentication with AA plugin in Remote Desktop connections" in the Administration Guide.

key

Type:	string
Required:	yes
Default:	N/A

Description: The name of the name-value pair.

prompt

Type:	string
Required:	yes
Default:	N/A

Description: The question itself in text format.

disable_echo

Type:	boolean yes|no
Required:	no
Default:	no

Description: Whether the answer to the question is visible (yes), or replaced with asterisks (no).

Store sensitive plugin data securely

Purpose:

By default, the configuration of the plugin is stored on SPS in the configuration of SPS. Make sure that you store the sensitive parameters (for example, api_key) of the plugin in an encrypted way. To do this, complete the following procedure.

Steps:

1. Log in to SPS and create a local Credential Store. For details, see "Configuring password-protected Credential Stores" in the Administration Guide.

   Instead of usernames and passwords, you will store the configuration parameters of the plugin in this Credential Store.

2. Add the plugin parameters you want to store in an encrypted way to the Credential Store. You can store any configuration parameter of the plugin in the Credential Store, but note that if an option appears in the Credential Store, the plugin will use it. If the same parameter appears in the configuration of the plugin, it will be ignored.

   - Enter the name of the configuration section without the brackets in the**HOST** field (for example, yubikey).
   - Enter the name of the plugin parameter in the**USERNAME** field (for example, api_key).
   - Enter the value of the plugin parameter in the**PASSWORD** field.

3. Commit your changes, and navigate to the configuration of the plugin on the **Policies > AA Plugin Configurations** page.

4. In the plugin configuration file, enter the name of the local Credential Store under the [plugin] section, in the cred_store parameter.

Perform multi-factor authentication with the SPS YubiKey plugin in terminal connections

Purpose:

To establish a terminal connection (SSH, TELNET, or TN3270) to a server, complete the following steps.

Steps:

1. Connect to the server.

   - If you can authenticate using an OTP or token, encode the OTP as part of the username. You can use the @ as a field separator. For example:

     `ssh otp=YOUR-ONE-TIME-PASSWORD@user@server`

     Replace `YOUR-ONE-TIME-PASSWORD` with your actual OTP. If needed, you can specify the type of OTP as a prefix to the OTP. For example, to specify the OTP of a YubiKey token:

     `ssh otp=y_YOUR-ONE-TIME-PASSWORD@user@server`

     - Google Authenticator: g
     - inWebo Authenticator: o
     - Symantec token: s
     - YubiKey: y
     - RSA token: r

   - If you need to authenticate using the YubiKey Verify push notification, approve the connection in your mobile app.

2. If SPS prompts you for further information, enter the requested information. If you need to authenticate with an OTP, but you have not supplied the OTP in your username, you will be prompted to enter the OTP.

3. Authenticate on the server.

4. If authentication is successful, you can access the server.

Perform multi-factor authentication with the SPS YubiKey plugin in Remote Desktop connections

Purpose:

To establish a Remote Desktop (RDP) connection to a server when the **AA plugin** is configured, complete the following steps.

Steps:

1. Open your Remote Desktop client application.

2. If you have to provide additional information to authenticate on the server, you must enter this information in your Remote Desktop client application in the *User name* field, before the regular content (for example, your username) of the field.

   If you can authenticate using an OTP or token, encode the OTP as part of the username. To encode additional data, you can use the following special characters:

   - % as a field separator
   - ~ as the equal sign
   - ^ as a colon (for example, to specify the port number or an IPv6 IP address)

   For example, use the following format:

   `domain\otp~YOUR-ONE-TIME-PASSWORD%Administrator`

   Replace `YOUR-ONE-TIME-PASSWORD` with your actual OTP. If needed, you can specify the type of OTP as a prefix to the OTP. For example, to specify the OTP of a YubiKey token: `domain\otp~y_YOUR-ONE-TIME-PASSWORD%Administrator`

   - Google Authenticator: `g`
   - inWebo Authenticator: `o`
   - Symantec token: `s`
   - YubiKey: `y`
   - RSA token: `r`

3. Connect to the server.

If you need to authenticate using the YubiKey Authenticator push notification, approve the connection in your mobile app.

4. Authenticate on the server.

5. If authentication is successful, you can access the server.

Learn more

To find out more about SPS, visit the One Identity page.

If you need help in connecting your YubiKey account with One Identity Safeguard for Privileged Sessions, contact our Sales Team or contact professionalservices@balabit.com.

About us

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

Contacting us

For sales or other inquiries, visit https://www.oneidentity.com/company/contact-us.aspx or call +1-800-306-9329.

Technical support resources

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

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

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