Legal Notice


Copyright © 2019 NetIQ Corporation, a Micro Focus company. All Rights Reserved.
## Contents

About NetIQ Corporation .......................................................... 9
About this Book ........................................................................ 11

1 Advanced Authentication Overview ........................................ 13
   How Is Advanced Authentication Better Than Other Solutions .... 13
   Key Features ......................................................................... 13
   Advanced Authentication Server Components .......................... 14
      Administration Portal ......................................................... 14
      Self-Service Portal ............................................................. 15
      Helpdesk Portal ................................................................ 15
      Reporting Portal ................................................................ 15
   Architecture ......................................................................... 16
      Basic Architecture .............................................................. 16
      Enterprise Level Architecture ............................................. 17
      Enterprise Architecture With A Load Balancer ..................... 19
   Terminologies ...................................................................... 20
      Authentication Method ....................................................... 20
      Authentication Chain ......................................................... 20
      Authentication Event ........................................................ 20
      Endpoint ........................................................................... 20

Part I Configuring Advanced Authentication ................................. 21

2 Logging In to the Advanced Authentication Administration Portal 23

3 Configuring Advanced Authentication Server Appliance ............. 25
   Managing Dashboard ............................................................. 25
   Adding Widgets .................................................................... 26
   Customizing Dashboard ......................................................... 28
   Updating Dashboard to View Real Time or Historical Data .......... 28
   Customizing the Default Widgets ......................................... 29
   Exporting Widgets ................................................................ 32
   Adding a Repository ............................................................ 32
      Adding an LDAP Repository ............................................... 33
      Adding an SQL Database .................................................... 41
      Adding an External Repository ............................................ 42
      Local Repository ............................................................... 43
   Configuring Methods ........................................................... 43
      BankID ............................................................................ 45
      Bluetooth ......................................................................... 46
      Card ................................................................................ 46
      Email OTP ........................................................................ 47
      Emergency Password ......................................................... 47
      Facial Recognition ............................................................. 48
      FIDO 2.0 .......................................................................... 49
      Fingerprint ....................................................................... 50
      LDAP Password ............................................................... 52
      OATH OTP ...................................................................... 53
<table>
<thead>
<tr>
<th>Configuring Events</th>
<th>85</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuring an Existing Event</td>
<td>85</td>
</tr>
<tr>
<td>Creating a Customized Event</td>
<td>90</td>
</tr>
<tr>
<td>Managing Endpoints</td>
<td>94</td>
</tr>
<tr>
<td>Configuring Policies</td>
<td>95</td>
</tr>
<tr>
<td>Authenticator Management Options Policy</td>
<td>96</td>
</tr>
<tr>
<td>Cache Options Policy</td>
<td>97</td>
</tr>
<tr>
<td>Custom Messages</td>
<td>98</td>
</tr>
<tr>
<td>Custom CSS</td>
<td>102</td>
</tr>
<tr>
<td>Delete Me Options</td>
<td>103</td>
</tr>
<tr>
<td>Endpoint Management Options</td>
<td>103</td>
</tr>
<tr>
<td>Event Categories</td>
<td>104</td>
</tr>
<tr>
<td>Geo Fencing Options</td>
<td>104</td>
</tr>
<tr>
<td>Google reCAPTCHA Options</td>
<td>105</td>
</tr>
<tr>
<td>Helpdesk Options</td>
<td>106</td>
</tr>
<tr>
<td>Locked Chains</td>
<td>107</td>
</tr>
<tr>
<td>Lockout Options</td>
<td>107</td>
</tr>
<tr>
<td>Login Options</td>
<td>108</td>
</tr>
<tr>
<td>Logon Filter for Active Directory</td>
<td>108</td>
</tr>
<tr>
<td>Mail Sender</td>
<td>109</td>
</tr>
<tr>
<td>Password Filter for Active Directory</td>
<td>110</td>
</tr>
<tr>
<td>Reporting Options</td>
<td>111</td>
</tr>
<tr>
<td>SMS Sender</td>
<td>111</td>
</tr>
<tr>
<td>Services Director Options</td>
<td>114</td>
</tr>
<tr>
<td>Voice Sender</td>
<td>114</td>
</tr>
<tr>
<td>Web Authentication</td>
<td>116</td>
</tr>
<tr>
<td>Adding a License</td>
<td>122</td>
</tr>
<tr>
<td>Adding a Report</td>
<td>123</td>
</tr>
</tbody>
</table>

4 Enrolling the Authentication Methods 131

Part II Configuring Integrations 133

5 OAuth 2.0 135

Building Blocks of OAuth 2.0 135
OAuth 2.0 Roles 135
OAuth 2.0 Grants 135
Sample OAuth 2.0 Application Integrated with Advanced Authentication 138
Running the Sample Web Application 143
OAuth 2.0 Attributes 143
Non Standard Endpoints 144
6 RADIUS Server
   Customizing Prompt Messages For RADIUS Event ........................................... 148
   Challenge-Response Authentication ................................................................. 149

7 SAML 2.0
   Integrating Advanced Authentication with SAML 2.0 ........................................... 151
   Requesting Advanced Authentication Methods and Chains Through a SAML AuthnRequest ... 152

8 Examples of Integrations
   Configuring Integration with Barracuda ......................................................... 155
   Configuring the Advanced Authentication RADIUS server .................................. 156
   Configuring the Barracuda SSL VPN Appliance ............................................... 156
   Authenticating on Barracuda SSL VPN Using Advanced Authentication ................. 157
   Configuring Integration with Citrix NetScaler ............................................... 157
   Configuring the Advanced Authentication RADIUS Server .................................. 157
   Configuring the Citrix NetScaler Appliance ..................................................... 158
   Authenticating on the Citrix NetScaler Using Advanced Authentication .................. 158
   Configuring Integration with Dell SonicWall SRA EX-Virtual Appliance ................. 159
   Configuring the Advanced Authentication RADIUS Server .................................. 159
   Configuring the Dell SonicWall SRA Appliance ................................................ 160
   Authenticating on Dell SonicWall SRA Workspace Using Advanced Authentication ..... 160
   Configuring Integration with FortiGate ............................................................ 160
   Configuring the Advanced Authentication RADIUS Server .................................. 161
   Configuring the FortiGate Appliance ............................................................... 161
   Authenticating on FortiGate Using Advanced Authentication ............................... 161
   Configuring Integration with OpenVPN ............................................................ 162
   Configuring the Advanced Authentication RADIUS Server .................................. 162
   Configuring the OpenVPN Appliance .............................................................. 163
   Configuring Integration with Palo Alto GlobalProtect Gateway ............................ 163
   Adding the RADIUS Server ............................................................................. 164
   Adding an Authentication Profile ...................................................................... 164
   Configuring GlobalProtect Gateway .................................................................. 164
   Configuring Integration with Salesforce .......................................................... 164
   Configuring the Salesforce Domain Name ......................................................... 165
   Configuring the SAML Provider ........................................................................ 165
   Configuring the Advanced Authentication SAML 2.0 Event ................................. 166
   Configuring to Authenticate on Salesforce with SAML 2.0 ................................. 167
   Configuring Integration with ADFS ..................................................................... 167
   Configuring the Advanced Authentication SAML 2.0 Event ................................. 168
   Making the Corresponding Changes in ADFS ..................................................... 169
   Configuring Integration with Google G Suite ..................................................... 170
   Configuring Google G Suite .............................................................................. 170
   Configuring the Advanced Authentication Event ................................................. 172
   Configuring to Authenticate on Google G-Suite with SAML 2.0 ............................ 172
   Configuring Integration with Office 365 ........................................................... 172
   Configuring Advanced Authentication SAML 2.0 Event ..................................... 173
   Making the Corresponding Changes in ADFS ..................................................... 174
   Authenticating on Office 365 .............................................................................. 174
   Configuring Integration with Sentinel .................................................................. 175
   Configuring the CEF Log Forward Policy on Advanced Authentication ................. 175
   Searching the Events on Sentinel ..................................................................... 175
Part III  Maintaining Advanced Authentication  177

9  Logging  179
   Syslog ................................................................. 180
   RADIUS Logs ......................................................... 194
   Async Logs ........................................................... 194
   Fingerprint Logs ...................................................... 194

10 Reporting  195

11 Managing Tokens  197
   CSV File Format To Import OATH Compliant Tokens ..................... 198

12 Searching a Card Holder’s Information  199

13 Troubleshooting  201
   The ON/OFF Switch Is Broken If the Screen Resolution Is 110% ............. 201
   Users Can Login Using the Old Password ..................................... 201
   Error is Displayed in the User Report Section of the Helpdesk Portal ......... 201
About NetIQ Corporation

We are a global, enterprise software company, with a focus on the three persistent challenges in your environment: Change, complexity and risk—and how we can help you control them.

Our Viewpoint

Adapting to change and managing complexity and risk are nothing new

In fact, of all the challenges you face, these are perhaps the most prominent variables that deny you the control you need to securely measure, monitor, and manage your physical, virtual, and cloud computing environments.

Enabling critical business services, better and faster

We believe that providing as much control as possible to IT organizations is the only way to enable timelier and cost effective delivery of services. Persistent pressures like change and complexity will only continue to increase as organizations continue to change and the technologies needed to manage them become inherently more complex.

Our Philosophy

Selling intelligent solutions, not just software

In order to provide reliable control, we first make sure we understand the real-world scenarios in which IT organizations like yours operate—day in and day out. That's the only way we can develop practical, intelligent IT solutions that successfully yield proven, measurable results. And that's so much more rewarding than simply selling software.

Driving your success is our passion

We place your success at the heart of how we do business. From product inception to deployment, we understand that you need IT solutions that work well and integrate seamlessly with your existing investments; you need ongoing support and training post-deployment; and you need someone that is truly easy to work with—for a change. Ultimately, when you succeed, we all succeed.

Our Solutions

- Identity & Access Governance
- Access Management
- Security Management
- Systems & Application Management
- Workload Management
- Service Management
Contacting Sales Support

For questions about products, pricing, and capabilities, contact your local partner. If you cannot contact your partner, contact our Sales Support team.

Worldwide:  www.netiq.com/about_netiq/officelocations.asp
United States and Canada:  1-888-323-6768
Email:  info@netiq.com
Web Site:  www.netiq.com

Contacting Technical Support

For specific product issues, contact our Technical Support team.

Worldwide:  www.netiq.com/support/contactinfo.asp
North and South America:  1-713-418-5555
Europe, Middle East, and Africa:  +353 (0) 91-782 677
Email:  support@netiq.com
Web Site:  www.netiq.com/support

Contacting Documentation Support

Our goal is to provide documentation that meets your needs. The documentation for this product is available on the NetIQ Web site in HTML and PDF formats on a page that does not require you to log in. If you have suggestions for documentation improvements, click Add Comment at the bottom of any page in the HTML version of the documentation posted at www.netiq.com/documentation. You can also email Documentation-Feedback@netiq.com. We value your input and look forward to hearing from you.

Contacting the Online User Community

NetIQ Communities, the NetIQ online community, is a collaborative network connecting you to your peers and NetIQ experts. By providing more immediate information, useful links to helpful resources, and access to NetIQ experts, NetIQ Communities helps ensure you are mastering the knowledge you need to realize the full potential of IT investments upon which you rely. For more information, visit community.netiq.com.
About this Book

This Administration Guide is intended for system administrators and describes the procedure of Advanced Authentication Server appliance configuration.

Intended Audience

This book provides information for individuals responsible for understanding administration concepts and implementing a secure, distributed administration model.
Advanced Authentication Overview

Advanced Authentication™ is a multi-factor authentication solution that enables you to protect your sensitive data by using a more advanced way of authentication on top of the typical username and password authentication. With Advanced Authentication, you can authenticate on diverse platforms by using different types of authenticators such as Fingerprint, Card, and OTP. Advanced Authentication provides a single authentication framework that ensures secure access to all your devices with minimal administration.

Authentication comprises of the following three factors:

- Something that you know such as password, PIN, and security questions.
- Something that you have such as smartcard, token, and mobile phone.
- Something that you are such as biometrics (fingerprint or iris).

You can achieve multi-factor or strong authentication by using any two factors out of this list. For example, multi-factor authentication can include combination of a password and a token or a smartcard and a fingerprint.

This section contains the following topics:

- “How Is Advanced Authentication Better Than Other Solutions” on page 13
- “Key Features” on page 13
- “Advanced Authentication Server Components” on page 14
- “Architecture” on page 16
- “Terminologies” on page 20

How Is Advanced Authentication Better Than Other Solutions

Advanced Authentication leverages the needs of users to authenticate on different platforms with different needs. The following points explain how Advanced Authentication is different from other solutions:

- Works on multiple platforms such as Windows, Mac OS X, Linux and so on.
- Supports multi-site configurations that helps organizations to distribute the authentication globally.

Key Features

- **Multi-factor Authentication**: The solution provides a flexibility of combining more than twenty authentication methods to create authentication chains. You can assign these chains to different events to use the specific authentication chains for different kinds of endpoints.
- **Supports Multiple Repositories**: Advanced Authentication supports Active Directory, Active Directory Lightweight Domain Services, NetIQ eDirectory, and other RFC 2037 and RFC 2037bis compliant LDAP repositories.
- **Supports Distributed Environments**: Advanced Authentication works on geographically distributed environments containing high loads.
- **Multitenancy**: A single Advanced Authentication solution can support multiple tenants to serve multiple customers with different environments.
- **Supports Multiple Platforms**: Advanced Authentication works on various platforms such as Windows, Linux, and Mac OS.
- **Helpdesk**: Advanced Authentication provides a separate role of Helpdesk or Security officer. A user with Helpdesk or Security Officer role can manage authenticators for the end users through the Helpdesk portal.
- **Supports the RADIUS Server**: Advanced Authentication Server contains a built-in RADIUS server to provide strong authentication for third-party RADIUS clients. Also, it can act as a RADIUS client for the third-party RADIUS servers.
- **Supports ADFS 3 and 4, OAuth 2.0, and SAML 2.0**: Advanced Authentication integrates with Active Directory Federation Services, OAuth 2.0, and SAML 2.0. This enables you to perform strong authentication for the users who need to access the third-party consumer applications.
- **Reporting**: Advanced Authentication provides the Reporting portal that enables you to access different security reports. You can also create customized reports based on your requirement.
- **Syslog support**: Advanced Authentication provides the central logging server that can be used for log forwarding. You can configure the solution to forward logs to an external Syslog server.
- **Supports Localization**: Advanced Authentication supports several languages such as Arabic, Chinese, Dutch, and Danish.

### Advanced Authentication Server Components

Advanced Authentication server comprises of the following components:

- **Administration Portal**
  
  For more information, see "Administration Portal" on page 14

- **Self-Service Portal**

  For more information, see "Self-Service Portal" on page 15

- **Helpdesk Portal**

  For more information, see "Helpdesk Portal" on page 15

- **Reporting Portal**

  For more information, see "Reporting Portal" on page 15

### Administration Portal

Administration Portal is a centralized portal that helps you to configure and manage various authentication settings such as methods, events, and so on. You can also configure various policies that are required for authentication. You can perform the following tasks:

- **Add repositories**: A repository is a database that stores users information. For example: An organization, Digital Airlines contains an Active Directory that stores all of the user’s information such as username, telephone, address, and so on. Administrator can add this Active Directory to
Advanced Authentication solution to help different departments in the organization such as the IT, finance, HR, and Engineering departments to authenticate based on their requirements. For more information about how to add repositories, see “Adding a Repository”.

- **Configure methods**: A method or an authenticator helps to confirm the identification of a user (or in some cases, a machine) that is trying to log on or access resources. You can configure the required settings for the appropriate methods depending on the requirement by each department. For more information about how to configure methods, see “Configuring Methods”.

- **Create chains**: A chain is a combination of methods. Users must authenticate with all the methods in a chain. For example, a chain with Fingerprint and Card method can be applicable for the IT department and a chain with Smartphone, LDAP Password, and HOTP is applicable for the Engineering department. For more information about how to create chains, see “Creating a Chain”.

- **Configure events**: An event is triggered by an external device or application that needs to perform authentication such as a Windows machine, a RADIUS client, a third party client and so on. After creating the chain, Administrator maps the chain to an appropriate event. For more information about how to configure events, see “Configuring Events”.

- **Map endpoints**: An endpoint is a device on which you can authenticate. Endpoints can be computers, Laptops, tablets, and so on. For more information about how to configure endpoints, see “Managing Endpoints”.

- **Configure policies**: An administrator can manage policies that are specific to users, devices, or locations to control a user’s authentication. In Advanced Authentication, you can manage the policies in a centralized policy editor. For more information about how to configure policies, see “Configuring Policies”.

**Self-Service Portal**

The Self-Service Portal allows users to manage the available authentication methods. This portal consists of Enrolled authenticators and Add authenticator. The Enrolled authenticators section displays all the methods that users have enrolled. The Add authenticator section displays additional methods available for enrollment. You must configure and enable the Authenticators Management event to enable users to access the Self-Service portal. For more information on Self-Service portal, see Advanced Authentication- User guide.

**Helpdesk Portal**

The Helpdesk Portal allows the helpdesk administrators to enroll and manage the authentication methods for users. Helpdesk administrators can also link authenticators of a user to help authenticate to another user’s account. For more information on Helpdesk portal, see the Advanced Authentication- Helpdesk Administrator guide.

**Reporting Portal**

The Reporting Portal allows you to create or customize security reports that provide information about user authentication. It also helps you understand the processor and memory loads. For more information on Reporting portal, see “Reporting”.

Advanced Authentication Overview 15
Advanced Authentication architecture is based on the following three levels of architecture:

- **Basic Architecture**
  
  For more information, see “Basic Architecture” on page 16

- **Enterprise Level Architecture**
  
  For more information, see “Enterprise Level Architecture” on page 17

- **Enterprise Architecture With A Load Balancer**
  
  For more information, see “Enterprise Architecture With A Load Balancer” on page 19

## Basic Architecture

The basic architecture of Advanced Authentication is a simple configuration that requires only one Advanced Authentication server.

An Advanced Authentication server is connected to a directory such as Active Directory Domain Services, NetIQ eDirectory, Active Directory Lightweight Directory Service or other compliant LDAP directories. An Event Endpoint can be Windows, Linux or Mac OS X machine, NetIQ Access Manager, NetIQ CloudAccess, or RADIUS Client to authenticate through the RADIUS Server that is built-in the Advanced Authentication Server. For a complete list of supported events, see Configuring Events.
Enterprise Level Architecture

In the enterprise level architecture of Advanced Authentication, you can create several sites for different geographical locations.

For example, the Figure 1-1 on page 17 displays two Advanced Authentication sites, Site A and Site B.

Figure 1-1  Enterprise Level Architecture

- **Site A**: The first site that is created for headquarters in New York. The first Advanced Authentication server of site A contains the **Global Master** and **Registrar** roles. This server contains a master database and it can be used to register new sites and servers.
- **Site B**: Another site created for the office in London. The structure of site B is similar to site A. The Global Master in another site has the DB Master role. DB servers interact with the DB Master.

**DB Server** provides a database that is used for backup and fail-over. You can create a maximum of two DB servers per site. When the Global Master is unavailable, the DB server responds to the database requests. When the Global Master becomes available again, the DB server synchronizes with the Global Master and the Global Master becomes the primary point of contact for database requests again.

Endpoints interact with Global Master or DB Master servers. When these servers are not available, they interact with DB servers.

**NOTE:** DB servers connect to each other directly. If the Global Master is down, the DB servers will replicate.

A Global Master must have a connection to each of the LDAP servers. Hence in a data center with Global Master, you must have LDAP servers for all the used domains.
Master servers do not initiate a connection to the DB servers. Master servers initiate connection to Master servers only. DB servers initiate connection to the DB Master of the same site and Registrar only.

**IMPORTANT:** Ensure to take regular snapshots or to clone the primary site to protect from any hardware issues or any other accidental failures. It is recommended to do it each time after you change the configuration of repositories, methods, chains, events and policies, or add or remove servers in the cluster.

You can convert DB server of primary site to Global Master. This requires corresponding DNS changes. Nothing can be done if Global Master and all slaves are lost.
Enterprise Architecture With A Load Balancer

The enterprise architecture with a load balancer contains web servers and load balancers along with the components in Enterprise Level Architecture. Figure 1-2 on page 19 illustrates the Enterprise architecture with a load balancer.

**Figure 1-2  Enterprise Architecture with Load Balancer**

- **Web Servers**: Web server does not contain a database. It responds to the authentication requests and connects to Global Master. It is not recommended to deploy more than 5-6 web servers per site.
- **Load Balancer**: A load balancer provides an ability to serve authentication requests from External Endpoints. A load balancer is a third-party component. It must be configured to interact with Web servers.
**WARNING:** Do not place the Advanced Authentication server in Demilitarized Zone (DMZ). It is recommended to use Load Balancer to process authentication requests from the external endpoints.

---

## Terminologies

- "Authentication Method" on page 20
- "Authentication Chain" on page 20
- "Authentication Event" on page 20
- "Endpoint" on page 20

### Authentication Method

An authentication method verifies the identity of an individual who wants to access data, resources, or applications. Validating that identity establishes a trust relationship for further interactions.

### Authentication Chain

An authentication chain is a combination of authentication methods. A user must pass all methods in the chain to be successfully authenticated. For example, if you create a chain with LDAP Password and SMS, a user must first specify the LDAP Password. If the password is correct, the system sends an SMS with a One-Time-Password (OTP) to the user’s mobile. The user must specify the correct OTP to be authenticated.

You can create chains with multiple methods that are applicable for highly secure environments. You can create authentication chains for specific group of users in the repositories.

### Authentication Event

An authentication event is triggered by an external device or application that needs to perform authentication. It can be triggered by a RADIUS Client (Citrix Netscaler, Cisco VPN, Juniper VPN and so on) or an API request. Each event can be configured with one or more authentication chains that enables a user to authenticate.

### Endpoint

An endpoint is a device on which you can authenticate. Endpoints can be computers, Laptops, tablets, Smartphones, and so on.
Advanced Authentication Server Appliance is intended for processing requests for authentication coming from the Advanced Authentication system users.

This chapter contains the following sections:

- Chapter 2, "Logging In to the Advanced Authentication Administration Portal," on page 23
- Chapter 3, "Configuring Advanced Authentication Server Appliance," on page 25
- Chapter 4, "Enrolling the Authentication Methods," on page 131
Logging In to the Advanced Authentication Administration Portal

After you set up an applicable server mode, the Advanced Authentication Administration portal is displayed.

To log in to the Advanced Authentication Administration portal, perform the following steps:

1. Specify the administrator's credentials in the format: repository\user (local\admin by default).
2. Click Next.
3. The Admin Password chain is selected by default as the only available chain. Specify the password that you specified while setting up the DB Master server mode.
4. Click Next.
   The Dashboard page is displayed.
5. You can change the language from the list on the upper-right corner of the Administration portal.
   The languages supported are: Arabic, Canadian French, Chinese Simplified, Chinese Traditional, Danish, Dutch, English, French, German, Italian, Japanese, Polish, Portuguese (Brazilian), Russian, Spanish, Hebrew, and Swedish.

**IMPORTANT:** Password of local\admin account expires by default. For uninterrupted access to the Administration portal, it is strongly recommended to add authorized users or group of users from a configured repository to the FULL ADMINS role. Then you must assign chains, which contain methods that are enrolled for users, to the AdminUI event (at a minimum with an LDAP Password).

**NOTE:** It is not recommended to access the Advanced Authentication Administration portal through a load balancer, as the replicated data may not be displayed.
Logging In to the Advanced Authentication Administration Portal
In the Administration portal, you can configure and manage various authentication settings such as methods, events, and so on. You can also configure various policies that are required for authentication.

Advanced Authentication Administration portal contains the Help option that guides you on how to configure all settings for your authentication framework. The Help section provides you with information on the specific section you are working on.

This chapter contains the following sections:

- “Managing Dashboard” on page 25
- “Adding a Repository” on page 32
- “Configuring Methods” on page 43
- “Creating a Chain” on page 83
- “Configuring Events” on page 85
- “Managing Endpoints” on page 94
- “Configuring Policies” on page 95
- “Adding a License” on page 122
- “Adding a Report” on page 123

**Managing Dashboard**

After you login into the Advanced Authentication Administration console, the Dashboard is displayed. Dashboard contains widgets that you can add or customize to view a graphical representation of data. The information in the Dashboard helps administrators to track memory utilization, tenant information, successful or failed logins, and so forth.

You can view the Dashboard for all the tenants or specific tenants.
You can perform the following to manage the Dashboard:

- Add widgets
- Customize Dashboard
- Update Dashboard
- Customize the Default Widgets
- Export Widgets

### Adding Widgets

To add widgets, perform the following steps:

1. Click Add widget in the top-right corner of the Dashboard screen.
2. Select the widget from the list that you want to add to the dashboard.
3. Specify the appropriate details for the widget in the Add Widget screen.
4. Click OK.

You can add the following types of widgets:

- Pie chart
- Stacked chart
- Activity stream
- Enroll activity stream
- Users
- Authenticators
- Licenses
- Events count line chart
- Events count line chart grouped by field
- Distinct Events Count Line Chart
- Distinct Events Count Line Chart Grouped by Field

**Pie Chart**

This widget displays the information collected on a specific parameter and represents information in the Pie chart format. You can also sort the parameter is ascending and descending order.

**Stacked Chart**

This widget displays a stacked bar chart that classifies and compares different categories of Field 1 and Field 2 parameters to track the maximum and minimum number of logons. X-axis represents categories of the Field 2 parameter. Y-axis represents logon count. Segments in each vertical bar represents categories of Field 1 parameter. Different colors are used to depict different categories and label for each category is displayed in upper-right corner of the widget.

**Activity Stream**

This widget displays information about user, tenant, chain, method used for authentication, and the result.

**Enroll Activity Stream**

This widget displays information about enrolled users: last log on time, tenant, user, method used for authentication, and event type.

**Users**

This widget displays information about the enrolled users: tenant name, user name, enrollment status and last log on time.

**Authenticators**

This widget displays information about the enrolled authenticators: tenant name, user name, event category, method, comment and owner of the account.

**Licenses**

This widget displays information about the license id, license validity dates (such as From and To dates), license expire status and license warnings (regarding license expiry, exceed in user count)
Event Count Line Chart

This widget tracks and displays logon count of all events in the appliance. The X-axis (horizontal) represents time and Y-axis (vertical) represents logon count. Each data point on the chart represents numbers of user logged on at a specific time. All the data points are plotted and connected with a line to track the maximum and minimum number of logons.

Events Count Line Chart Grouped by Field

This widget tracks and displays logon count of specific parameter. The X-axis (horizontal) represents time and Y-axis (vertical) represents logon count. Data points of different colors represent specific category of the selected parameter. The label for each category is displayed in upper-right corner of the widget. All the data points are plotted and connected with a line to track the maximum and minimum number of logons.

Distinct Events Count Line Chart

This widget tracks and displays distinct count of all categories in the selected parameter (Distinct values by field). X-axis (horizontal) represents time and Y-axis (vertical) represents distinct logon count. Each data point on the chart represents unique logon count at a specific time. All the data points are plotted and connected with a line to track the maximum and minimum number of distinct logons.

For example: If the Distinct events count line chart widget is customized as follows:

- Interval set to 1 hour.
- Distinct values by field is set to User name.

The widget displays number of unique users logged in to all events for the time duration of 1 hour.

Distinct Events Count Line Chart Grouped by Field

This widget displays and classifies distinct logon count of each event. The X-axis (horizontal) represents time and Y-axis (vertical) represents distinct logon count. Each data point on the chart represents unique logon count of particular event at a specific time. All the data points are plotted and connected with a line to track the maximum and minimum number of distinct logons to particular event.

Customizing Dashboard

You can customize the Dashboard by moving the widgets or deleting the unused widgets.

To move the widgets, click on the widget and the drag icon  appears. You can then drag and drop the widget to the desired location of the Dashboard.

To delete unused widgets, click Delete on the top of each widget.

After customizing the dashboard, click Save Dashboard on the upper-right corner of the Dashboard screen.

Updating Dashboard to View Real Time or Historical Data

You can update Dashboard to view the data based on the time interval or historical data.
Viewing Dashboard based on Time Interval

To view records based on real time interval, perform the following steps:

1. Set Relative time interval to ON in the Dashboard section.
2. Select the time interval from Relative interval. By default, time interval is set to Last 15 minutes.
3. Click Update.

Viewing Dashboard for Previous Records

To view previous records, perform the following steps:

1. Set Relative time interval to OFF in the Dashboard section.
2. Select the Date range.
3. Click Update.

Customizing the Default Widgets

To customize the widget, click Edit and select the appropriate filters. You can edit the widget title and customize the display based on the following filter factors:

- **Event type**: Select preferred event type. Options available are All logon events, Failed logon events and Successful logon events.
- **Interval**: Select Time interval.
- **Size**: Select number of records.
- **Sort**: Select sorting order. Options available are ascending or descending order.
- **Field**: Select the parameter based on which the data must be collected to display on the widget. Options available are Event Name, Chain Name, Method Name, Endpoint Name and so on.
- **Users**: Select specific user.
- **Events**: Select specific event.
- **Chains**: Select specific chain.

Following are the default widgets when you login. You can edit these widgets according to your need:

- “Server Metrics” on page 30
- “CPU and Memory Usage Per Server” on page 30
- “Tenants” on page 30
- “Authentications” on page 30
- “Logons Per Result” on page 30
- “Total Users” on page 30
- “Total Users Per Event” on page 31
- “Activity Stream” on page 31
- “Successful/Failed Logons” on page 31
- “Top Events With Successful Logon Per Chain” on page 31
- “Top Events With Failed Logon Per Method” on page 31
- “Top 10 Events” on page 31
- “Top 10 chains With Successful Result” on page 31
Server Metrics

This widget displays statistics about user’s login, popularity and so on. The following section defines each server metric:

- **All Logons**: Total number of logins.
- **Failed Login**: Total number of failed logins by the users.
- **Succeeded Login**: Total number of successful logins by the users.
- **Active Users Count**: The number of active users.
- **Most Popular User**: The user that has used the console most.
- **Most Popular Event**: The event that users have used the most.
- **Most Popular Repo**: The repository that users have used the most.
- **CPU Usage Stats**: The average percentage of CPU usage.

CPU and Memory Usage Per Server

These widgets display information about percentage of CPU and memory usage of server for the set time interval. These widgets display average CPU and memory usage.

Tenants

This widget displays information about the tenants and their login.

Authentications

This widget displays the total logon count for time interval.

Logons Per Result

This widget displays two lines: one for successful logons and one for failed logons.

Total Users

This widget displays the total number of logged in users for time interval.
**Total Users Per Event**
This widget displays the total number of logged in users for each event.

**Activity Stream**
This widget displays information about user, tenant, chain, method used for authentication, and the result.

**Successful/Failed Logons**
This widget displays information about the successful or failed users login.

**Top Events With Successful Logon Per Chain**
This widget displays the top events based on the successful logon for each chain.

**Top Events With Failed Logon Per Method**
This widget displays the top events based on the failed logon for each chain.

**Top 10 Events**
This widget displays the top ten events the user has performed.

**Top 10 chains With Successful Result**
This widget displays the top ten chains the user has successfully authenticated with.

**Top 10 Servers**
This widget displays the top ten servers the user has used to authenticate.

**Top 10 Tenants**
This widget displays the top ten tenants.

**Top 10 Repositories**
This widget displays the top ten repositories.

**Top 5 Events for Logons**
This widget displays the top five events for login.

**Top 5 Users for Logons**
This widget displays the top five users for login.
Top 10 Users With Failed Logon
This widget displays the top ten users who have failed in the login attempt.

Top 10 Users
This widget displays the top ten users.

Top 10 Events
This widget displays the top ten events.

Top 10 Methods With Failed Result
This widget displays the top ten methods with failed authentication results.

Exporting Widgets
When you export a widget, Advanced Authentication creates a copy of the selected widget in the Reports section. You must navigate to Reports page to download the exported file on your local drive.

To export a widget, perform the following steps:

1. Select the preferred widget on the Dashboard page.
2. Click Export and select preferred format. Formats available are:
   - .csv
   - .json
3. Click Reports.
4. Click the exported file name in the Exported reports section, to download on the local drive.

Adding a Repository
A repository is a central location where the user’s data is stored. Advanced Authentication uses the repository only to retrieve the user information and configurations in Advanced Authentication do not affect the repository. The authentication templates are stored inside the appliance and are fully encrypted.

Advanced Authentication supports any LDAP compliant directory such as Active Directory Domain Services, NetIQ eDirectory, Active Directory Lightweight Directory Services, OpenLDAP, and OpenDJ. Advanced Authentication also supports the SQL database.

When you add a new repository, you can match the users in the repository to the authentication chains. You require only the read permission to access a repository.

You can add the following repositories:

- Any LDAP repository
- SQL database
- External repository
Adding an LDAP Repository

To add a repository, perform the following steps:

1. Click Repositories > Add.
2. Select an applicable repository type from the LDAP type list. The options are:
   - AD for Active Directory Domain Services
   - AD LDS for Active Directory Lightweight Domain Services
   - eDirectory for NetIQ eDirectory
   - Other for OpenLDAP, OpenDJ and other types
   For AD, a repository name is automatically set to the NetBIOS name of the domain. For other LDAP repository types, you need to specify the name in Name.
3. Specify a container for the users in Base DN. When you select the Subtree option, Advanced Authentication performs a search for the users in all the child nodes. You can change the search scope by selecting the Search one level only option.
4. Specify a user account in User and specify the password of the user in Password. Ensure that the user’s password has no expiry.
5. You can specify a container for the groups in Group DN (optional). When you select the Subtree option, Advanced Authentication performs a search for the groups in all the child nodes. You can change the search scope by selecting the Search one level only option.
6. If you have selected AD as the LDAP type, you can perform the DNS discovery either automatically or manually.
   - Automatic DNS discovery
   - Manual DNS Discovery

Automatically Performing the DNS Discovery

1. Select DNS discovery in the LDAP servers option.
2. Specify the DNS zone.
3. Specify the Site name (optional).
4. The Use SSL option is set to OFF by default. This indicates that the DNS discovery is done on a non-SSL mode for the port 389. An _ldap SRV record is retrieved from the DNS server when this option is disabled. For example, _ldap._tcp.test2.local2.
   To use SSL for DNS discovery on port 636, turn Use SSL to ON. An _ldaps SRV record is retrieved from the DNS server. For example, _ldaps._tcp.test2.local2. However, administrators must create the SRV record on the DNS server before using the SSL option.
5. Click Perform DNS Discovery.
   When the DNS discovery is done, the DNS servers list is updated every three hours.

Manually Performing the DNS Discovery

1. Select the Manual setting option in the LDAP servers option to add LDAP servers manually.
2. Click Add server. You can add the different servers in your network. The list is used as a pool of servers. Each time the connection is open, a random server is selected in the pool and unavailable servers are discarded.
3. Specify an LDAP server’s Address and Port.
4. Turn SSL to ON to use SSL (if applicable).
NOTE: If you specify an RODC (Read Only Domain Controller) in the LDAP server, the server uses this DC for read requests (get groups, get user info) and for logon requests (LDAP Password method and bind requests for Advanced Authentication LDAP user). These requests are redirected to a writable DC because RODC is installed in untrusted locations and does not have copies of the user’s passwords. Therefore, if a writable DC is not available, Advanced Authentication will not be able to bind to the LDAP repository.

To solve this issue, you must enable the password replication of a user account specified in Step 4. To do this, you must add the account to the Allowed RODC Password Replication Group.

However, even when you enable such replication, users cannot use the LDAP Password method because user’s passwords are not replicated. It is recommended not to replicate passwords of all the users. For more information, see the article Understanding “Read Only Domain Controller” authentication.

NOTE: A Global Master must have connection to each of the LDAP servers. Therefore, in a data center with Global Master, you must have LDAP servers for all the used domains. In the secondary sites, ensure that the LDAP servers list contains only local LDAP servers to prevent an Advanced Authentication server to communicate to an LDAP Server that is located remotely. This is because communication to servers that are located far may result in delays.

5. Click the save icon next to server’s credentials.

Add additional servers (if applicable).

7 (Conditional) To configure custom attributes, expand Advanced Settings. The Advanced Settings are required for OpenDJ, OpenLDAP, and in some cases for NetIQ eDirectory.

8 Click Save.

NOTE: If you use NetIQ eDirectory with the option Require TLS for Simple Bind with Password enabled, you may get the error: Can't bind to LDAP: confidentialityRequired. To fix the error, you must either disable the option or do the following:

1. Click LDAP > LDAP Options > Connections in the NetIQ eDirectory Administration portal.
2. Set Client Certificate to Not Requested.
3. Set a correct port number and select SSL in the Repository settings.
4. Click Sync now with the added repository.

9 You can change the search scope and the Group DN (optional) functionality. In Advanced Authentication 5.2, you had to specify a common Base DN for users and groups.

10 To verify the synchronization of a repository, click Edit and you can view the information in Last sync.

11 Click Full synchronization to perform a complete synchronization of the repository.

NOTE: Full synchronization can be initiated only on the Global Master server.

Advanced Authentication performs automatic synchronization of modified objects (fast synchronization) on an hourly basis for AD repositories only. The fast synchronization is used to remove the users who are no longer a member of the groups that are assigned to the authentication chain of Advanced Authentication. It allows to free up their license.

The complete synchronization (Full synchronization) is performed on a weekly basis for all types of repositories.
NOTE: If an LDAP server is unavailable for 2.5 seconds, Advanced Authentication excludes it from the LDAP requests for a period of 3 minutes.

Advanced Settings

Advanced Settings allow you to customize attributes that Advanced Authentication reads from a repository. Click + to expand the Advanced Settings. The following list describes the different attributes in Advanced Settings:

- “User Lookup Attributes” on page 35
- “User Name Attributes” on page 35
- “User Mail Attributes” on page 35
- “User Cell Phone Attributes” on page 35
- “Group Lookup Attributes” on page 36
- “Group Name Attributes” on page 36
- “Verify SSL Certificate” on page 37
- “Enable Paged Search” on page 37
- “Enable Nested Groups Support” on page 37
- “Framed IPv4 Address Attribute” on page 37
- “Used Attributes” on page 38

User Lookup Attributes

Advanced Authentication validates the specified attributes for an entered user name.

For Active Directory (AD), the default attributes are `sAMAccountName` and `userPrincipalName`. For other repositories, `cn` is the default attribute.

User Name Attributes

Advanced Authentication shows a name from the first, non-empty specified field for an entered user name.

For AD, the default attributes are `sAMAccountName` and `userPrincipalName`. For other repositories, `cn` is the default attribute.

User Mail Attributes

Advanced Authentication validates the specified attributes to retrieve a user's email address.

Default attributes are `mail` and `otherMailbox`.

User Cell Phone Attributes

Advanced Authentication validates the specified attributes to retrieve a user's phone number. These attributes are used for methods such as SMS OTP, Voice, and Voice OTP. Previously, the first attribute of User cell phone attributes was used as a default attribute for authenticating with SMS OTP, Voice, and Voice OTP methods. Now, users can use different phone numbers for these methods. For example, Bob wants to authenticate with SMS OTP, Voice, and Voice OTP methods. He
has a cell phone number, a home phone number, and an IP phone number and wants to use these numbers for each of these methods. He can define these phone numbers in the respective settings of these methods.

Default attributes: `mobile, otherMobile`.

---

**NOTE:** If you have multiple repositories, you must use the same configuration of User cell phone attributes for all the repositories.

---

**Group Lookup Attributes**

Advanced Authentication validates the specified attributes for an entered group name.

For Active Directory, the default attribute is `sAMAccountName`. For other repositories, `cn` is the default attribute.

**Group Name Attributes**

Advanced Authentication shows a name from the first, non-empty specified field for an entered group name.

For Active Directory, the default attribute is `sAMAccountName`. For other repositories, `cn` is the default attribute.

Advanced Authentication supports the RFC 2037 and RFC 2037 bis. RFC 2037 determines a standard LDAP schema and contains a `memberUid` attribute (POSIX style). RFC 2037 bis determines an updated LDAP schema and contains a member attribute. Active Directory, LDS, and eDir support RFC 2037 bis. OpenLDAP contains `posixAccount` and `posixGroup` that follows RFC 2037.

Advanced Authentication supports the following attributes for the Group Name attributes:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Default Value</th>
<th>Value for the Repository</th>
</tr>
</thead>
<tbody>
<tr>
<td>User Object Class</td>
<td>user</td>
<td>OpenDJ and OpenLDAP: <code>person</code></td>
</tr>
<tr>
<td>Group Object Class</td>
<td>group</td>
<td>OpenDJ: <code>groupOfNames</code></td>
</tr>
<tr>
<td></td>
<td></td>
<td>OpenLDAP: <code>posixGroup</code></td>
</tr>
<tr>
<td>Group Member Attribute</td>
<td>member</td>
<td>OpenDJ: <code>member</code></td>
</tr>
<tr>
<td></td>
<td></td>
<td>OpenLDAP: <code>memberUid</code></td>
</tr>
<tr>
<td></td>
<td></td>
<td>If a required group contains <code>groupOfNames</code> class, disable <strong>POSIX style groups</strong>. If the group contains <code>posixGroup</code>, enable <strong>POSIX style groups</strong>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• User UID attribute</td>
</tr>
<tr>
<td></td>
<td></td>
<td>This attribute is available only when <strong>POSIX style groups</strong> is <strong>ON</strong>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Default value: <code>uid</code></td>
</tr>
</tbody>
</table>

Object ID Attribute     | entryUUID     |
This attribute is available only for other LDAP type only.
NOTE: For information about the Logon filter settings (Legacy logon tag and MFA logon tag), see Configuring Logon Filter.

Verify SSL Certificate

Enable Verify SSL Certificate to ensure that the LDAP connection to appliance is secured with a valid self-signed SSL certificate. This helps to prevent any attacks on the LDAP connection and ensures safe authentication. Click Browse to browse the self-signed certificate.

Enable Paged Search

The Enable paged search option allows LDAP repositories to support paged search in which the repositories can retrieve a result of a query set in small portions. By default, this option is set to ON. For openLDAP (with file-based backend), the option must be set to OFF.

NOTE: You must not disable the option for Active Directory repositories. It can also affect the performance on other supported repositories such as NetIQ eDirectory.

Enable Nested Groups Support

This option allows you to enable or disable nested groups support. By default, the Enable nested groups support option is set to ON.

If Enable nested groups support option is set to ON, then Advanced Authentication will authenticate all the users of the group and its nested groups assigned to a chain. If Enable nested groups support option is set to OFF, then Advanced Authentication will authenticate only the members of the group assigned to the chain. The members of the nested groups cannot access the chain.

Consider there is a group by name All Users assigned to SMS Authentication chain and the All Users group has subgroups Contractors and Suppliers. When Enable nested groups support option is set to ON, then Advanced Authentication will authenticate All Users group and the nested groups Contractors and Suppliers for SMS Authentication chain. When the option is set to OFF, then Advanced Authentication will authenticate only the members of All Users group and the nested group members will not have access to SMS Authentication chain. This improves the login performance of the appliance.

Framed IPv4 Address Attribute

This attribute is applicable for the RADIUS Server event.

For Active Directory, when the Framed IPv4 Address is blank, the Advanced Authentication RADIUS server returns value of the msRADIUSFramedIPAddress attribute as Framed-IP-Address after you log in with the RADIUS event. When you specify any other attribute in Framed IPv4 Address attribute, then the value of the specified attribute is returned as the Framed-IP-Address instead of the msRADIUSFramedIPAddress attribute value. You can configure the Framed-IP-Address in Active Directory Users and Computers > Dial-in > Assign Static IP Addresses and click Static IP Addresses. It supports only IPv4.

For the other repositories, when the Framed IPv4 Address is blank, the Advanced Authentication RADIUS server returns value of the radiusFramedIFAddress attribute as Framed-IP-Address after you log in with the RADIUS event. When you specify any other attribute in Framed IPv4 Address attribute, then the value of the specified attribute is returned as the Framed-IP-Address instead of the radiusFramedIFAddress attribute value.
# Used Attributes

The following table describes the attributes that the appliance uses in the supported directories.

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>LDAP Name</th>
<th>Description</th>
<th>Type</th>
<th>Supported in Active Directory</th>
<th>Supported in LDS</th>
<th>Supported in eDirectory</th>
</tr>
</thead>
<tbody>
<tr>
<td>CN (Common Name)</td>
<td>CN</td>
<td>An identifier of an object</td>
<td>String</td>
<td>?</td>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>Mobile</td>
<td>Mobile</td>
<td>A phone number of an object’s cellular or mobile phone</td>
<td>Phone number</td>
<td>?</td>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>Email Address</td>
<td>mail</td>
<td>An email address of a user</td>
<td>Email address</td>
<td>?</td>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>User-Principal-Name (UPN)</td>
<td>userPrincipalName</td>
<td>An Internet based format login name for a user</td>
<td>String</td>
<td>?</td>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>SAM-Account-Name</td>
<td>sAMAccountName</td>
<td>The login name used to support clients and servers running earlier versions of operating systems such as Windows NT 4.0</td>
<td>String</td>
<td>?</td>
<td>×</td>
<td>×</td>
</tr>
<tr>
<td>GUID</td>
<td>GUID</td>
<td>An assured unique value for any object</td>
<td>Octet String</td>
<td>×</td>
<td>×</td>
<td>?</td>
</tr>
<tr>
<td>Object Class</td>
<td>Object Class</td>
<td>An unordered list of object classes</td>
<td>String</td>
<td>?</td>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>Member</td>
<td>Member</td>
<td>A list that indicates the objects associated with a group or list</td>
<td>String</td>
<td>?</td>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>User-Account-Control</td>
<td>userAccountControl</td>
<td>Flags that control the behavior of a user account</td>
<td>Enumeration</td>
<td>?</td>
<td>×</td>
<td>×</td>
</tr>
<tr>
<td>ms-DS-User-Account-Control-Computed</td>
<td>msDS-UserAccountControl-Computed</td>
<td>Flags that are similar to userAccountControl, but the attribute’s value can contain additional bits that are not persisted</td>
<td>Enumeration</td>
<td>?</td>
<td>?</td>
<td>×</td>
</tr>
<tr>
<td>Primary-Group-ID</td>
<td>primaryGroupId</td>
<td>A relative identifier (RID) for the primary group of a user</td>
<td>Enumeration</td>
<td>?</td>
<td>×</td>
<td>×</td>
</tr>
<tr>
<td>Object-Guid</td>
<td>objectGUID</td>
<td>A unique identifier for an object</td>
<td>Octet String</td>
<td>?</td>
<td>?</td>
<td>×</td>
</tr>
<tr>
<td>object-Sid</td>
<td>objectSid</td>
<td>A Binary value that specifies the security identifier (SID) of the user</td>
<td>Octet String</td>
<td>?</td>
<td>?</td>
<td>×</td>
</tr>
</tbody>
</table>
### LDAP Queries for Repository Sync

**Active Directory DS and AD LDS Queries**

1. **Search users**

   `{(&(usnChanged>=217368)(&(objectClass=user)(|(cn=*)(sAMAccountName=*)(userPrincipalName=*))))}`

   **Requested attributes:**

   ```
   ['objectSID', 'sAMAccountName', 'objectClass', 'logonHours', 'primaryGroupId', 'otherMobile', 'mobile', 'userAccountControl', 'cn', 'usnChanged', 'userPrincipalName', 'msDS-User-Account-Control-Computed', 'objectGUID', 'mail', 'otherMailbox', 'GUID']
   ```

2. **Search groups**

   `{(&(usnChanged>=217368)(&(objectClass=group)(|(cn=*)(sAMAccountName=*))))}

   **Requested attributes:**

   ```
   ['objectSID', 'sAMAccountName', 'objectClass', 'logonHours', 'primaryGroupId', 'userAccountControl', 'cn', 'usnChanged', 'msDS-User-Account-Control-Computed', 'objectGUID', 'GUID']
   ```

**eDirectory Queries**

The queries are the same as for Active Directory DS and Active Directory LDS, except for 'usnChanged' (this filter is not used).

1. **Search users**

   `{(objectClass=user)(|(cn=*)(sAMAccountName=*)(userPrincipalName=*))}

   **Requested attributes:**

   ```
   ['objectSID', 'sAMAccountName', 'objectClass', 'logonHours', 'primaryGroupId', 'otherMobile', 'mobile', 'userAccountControl', 'cn', 'userPrincipalName', 'msDS-User-Account-Control-Computed', 'objectGUID', 'mail', 'otherMailbox', 'GUID']
   ```

2. **Search groups**

   `{(objectClass=group)(|(cn=*)(sAMAccountName=*))}

   **Requested attributes:**

   ```
   ['objectSID', 'sAMAccountName', 'objectClass', 'logonHours', 'primaryGroupId', 'userAccountControl', 'cn', 'userPrincipalName', 'msDS-User-Account-Control-Computed', 'objectGUID', 'mail', 'otherMailbox', 'GUID']
   ```

### Logon-Hours

*Hours that the user is allowed to logon to the domain*

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>LDAP Name</th>
<th>Description</th>
<th>Type</th>
<th>Supported in Active Directory</th>
<th>Supported in LDS</th>
<th>Supported in eDirectory</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logon-Hours</td>
<td>logonHours</td>
<td>Hours that the user is allowed to logon to the domain</td>
<td>Octet String</td>
<td>?</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>USN-Changed</td>
<td>uSNChanged</td>
<td>An update sequence number (USN) assigned by the local directory for the latest change including creation</td>
<td>Interval</td>
<td>?</td>
<td>?</td>
<td>x</td>
</tr>
</tbody>
</table>

**NOTE:** The `sAMAccountName` and `userPrincipalName` attributes are supported only for AD DS repository. The Active Directory LDS and eDirectory repositories do not support the attributes.
Requested attributes:

['objectSID', 'sAMAccountName', 'objectClass', 'logonHours', 'primaryGroupId', 'userAccountControl', 'cn', 'msDS-User-Account-Control-Computed', 'objectGUID', 'GUID']

LDAP Queries During Logon

For Active Directory LDS queries, the attributes are same as Active Directory DS except for the objectSid (the filter is not used in queries on membership in groups).

In the examples below, the username is pjones, base_dn is DC=company,DC=com

Active Directory DS and Active Directory LDS queries

1. Basic user information

(&{objectClass=user} {|(cn=pjones) (sAMAccountName=pjones) (userPrincipalName=pjones)})

Requested attributes:

{objectGUID=\0f\d1\14\49\bc\cc\04\44\bf\19\06\15\c6\82\55})

Requested attributes:

['otherMobile', 'GUID', 'userAccountControl', 'msDS-User-Account-Control-Computed', 'mobile', 'primaryGroupId', 'cn', 'objectGUID', 'userPrincipalName', 'objectSID', 'mail', 'sAMAccountName', 'objectClass', 'logonHours', 'otherMailbox']

2. Group membership information for user

Active Directory specific query using objectSid filter:

{((member=CN=pjones,CN=Users,DC=company,DC=com) (objectSid=S-1-5-21-3303523795-41305529-2892985274-513))}

Requested attributes:

['GUID', 'userAccountControl', 'msDS-User-Account-Control-Computed', 'primaryGroupId', 'objectGUID', 'cn', 'objectSID', 'objectClass', 'sAMAccountName', 'logonHours']

3. Iteratively query about each group received from above query

((member=CN=Performance Monitor Users,CN=Builtin,DC=company,DC=com)

Requested attributes:

['GUID', 'userAccountControl', 'msDS-User-Account-Control-Computed', 'primaryGroupId', 'objectGUID', 'cn', 'objectSID', 'objectClass', 'sAMAccountName', 'logonHours']

eDirectory Queries

Basic user information

(&{objectClass=user} {|(cn=pjones) (sAMAccountName=pjones) (userPrincipalName=pjones)})

Requested attributes:
Adding an SQL Database

You can add an MSSQL database to be consumed as a repository by Advanced Authentication. The following version of SQL servers are supported:

- Microsoft SQL Server 2016
- PostgreSQL 11
- MySQL 5.5

To add an SQL database, perform the following steps:

1. Click **Repositories > Add SQL repo**.
2. Specify the following details of the SQL database:
   - **Name**: Name of the repository.
   - **Database type**: Select one of the following options:
     - **MSSQL** (Select the option for Microsoft SQL Server)
     - **MYSQL**
     - **POSTGRESQL**
   - **DB host**: IP address of the database host.
   - **DB name**: Name of the database.
   - **DB user**: Name of the database user.
   - **Password**: Password of the database.
- **Table or view name**: Name of the table or view in the database.
- **User's id column** and **User's id type**: User's id column and id type in the database.
- **User's name column** and **User's name type**: The username column and the type in which the name is specified.
- **User's phone column**, and **User's email column**: The phone and email column in the database.

**IMPORTANT**

- The LDAP Password method is not applicable for the users in SQL repository. The Password method for the users is not enrolled automatically and can be enrolled manually by the Helpdesk administrator only.
- You must disable the *Ask credentials of management user* in the Helpdesk Options policy for the SQL repository. This enables the helpdesk administrator to set an authenticator for a user, without getting authenticated with the user's password on the *User to Manage* page of the Helpdesk portal.
- The SQL repository supports auto enrollment of Email OTP, SMS OTP, and Voice OTP methods. If you use only these methods, you can create a chain with one or some of these methods. You do not need the Helpdesk administrator’s assistance for the enrollment of these methods. It is not recommended to use a single factor chain with only one of these methods as it is not secure.

### Adding an External Repository

You can add an external repository that will act as a Repo Agent. This agent will act as an intermediate between the LDAP repository and Advanced Authentication. This agent will take care of all the synchronizations of the repositories even when the Advanced Authentication is hosted on cloud.

To add a Repo Agent, perform the following steps:

1. Click **Repositories > Add External repo**.
2. Specify the following details of the external repository:
   - **Name**: Name of the repository.
   - **Username**: Name of the user using the repository.
   - **Password**: Password of the repository.

   **NOTE**: Name of the repository must be the same as what is defined in the Repo Agent. The name of the repository must not contain spaces.

   **Username**: Name of the user using the repository.
   - **Password**: Password of the repository.

   **NOTE**: The Username and Password are defined in the secret.json file of the Repo Agent. For information about the secret.json file, see *Setting Up the Config Folder of Repo Agent*.

3. Add external server configurations:
   - **Add external server configurations**:
     3a. Click **Add Server**.
     3b. Specify the IP address of the Repo Agent in **Address**.
3c Specify the port number of the server in Port. For example, 9443.

3d Click the save icon next to the server credentials.

4 Click Choose File to upload the CA certificate for the agent.

For more information about uploading the CA certificates, see “Setting Up the Repo Agent for Certificates and Services” in the Advanced Authentication - Repo Agent guide.

5 Click Save.

Local Repository

The Local repository contains the Advanced Authentication server data. You can manage users and set roles for users in the local repository.

To edit a local repository, perform the following steps:

1 Click Edit in the LOCAL section of Repositories.

2 In the Global Roles tab, you can manage the Helpdesk administrators as ENROLL ADMINS, Advanced Authentication administrators as FULL ADMINS, and an additional privilege to share the authenticators to the Helpdesk administrators as SHAREAUTH ADMINS.

By default, there are no ENROLL ADMINS and the account LOCAL\ADMIN is specified as FULL ADMIN. You can change this by adding the user names from local or the repositories in Members.

NOTE: By default the helpdesk administrator cannot share the authenticators. Only when the helpdesk administrator is added in Members in the SHAREAUTH ADMINS, the helpdesk administrator will be able to share the authenticators. However, the Enable sharing of authenticators in “Authenticator Management Options Policy” policy must be enabled to share authenticators.

3 Click Save.

4 In the Users tab, you can manage the local users.

To add the new local account, click Add and specify the required information of the user.

5 In the Settings tab, you can edit the name of the Local repository.

Configuring Methods

A method is a way of authenticating the identity of an individual who attempts to access an endpoint. Advanced Authentication provides several such methods.

To configure an authentication method for Advanced Authentication, perform the following steps:

1 Click Methods.

2 Click the Edit icon next to the authentication method.

3 Make the required changes.

4 Click Save.
Customizing Method Names

You can translate the method name to a preferred language in the Custom names section. The translated method name will appear in the following portals, clients, and events:

- Portals: Administration, Helpdesk, Self-Service, and Reporting
- Clients: Windows, Linux PAM, and Mac OS X
- Events: OSP, RADIUS, and custom events.

To customize and translate the method name to a specific language, perform the following steps:

1. Open the method for which you want to localize the method name.
2. Specify the method name in a specific language field in the Custom names section.
3. Click Save.

Tenancy Settings

After configuring the authentication methods, you must create an authentication chain and map the configured methods to the chain. You can also create a chain with a single method. For example, you can create different authentication chains for an organization that has two departments, IT and Finance. For the IT department, you can create a chain with Password and Smartphone methods. For the Finance department, a chain with only the Fingerprint method can be created. For more information about creating chains, see “Creating a Chain”.

The methods do not appear in the Self-Service portal until you include them in a chain, and link that chain to an event.

You can configure the following methods in Advanced Authentication:

- BankID
- Bluetooth
- Card
- Email OTP
- Emergency Password
- Facial Recognition
- FIDO 2.0
- Fingerprint
- LDAP Password
- OATH OTP
- Password
- PKI
- RADIUS Client
- Security Questions
- Smartphone
- SMS OTP
- Swisscom Mobile ID
- FIDO U2F
- Voice
- Voice OTP
• Web Authentication Method
• Windows Hello

NOTE: Configurations that have been set by a top administrator for a particular method are grayed out. The configurations are not displayed, if the configurations are hidden by the top administrator.

BankID

Advanced Authentication provides the BankID method that facilitates users to authenticate with their personal identification number. Advanced Authentication supports both the desktop and the mobile versions of BankID. In this method, the user must configure the BankID app with the personal identification number, activation, and security code. The security code is mapped with the personal identification number.

NOTE: The user must ensure to set the security code with six digits in non-sequential format (for example: 221144) in the BankID app.

While enrolling the user, the specified identification number is saved as a template in the Advanced Authentication database. This method allows the users to get authenticated by specifying their secret code configured on the BankID app.

When a user wants to authenticate on an endpoint such as a laptop or a website with the BankID method. In this scenario, the authentication flow is as follows:

1. When the authentication request is initiated, the endpoint contacts the Advanced Authentication server.
2. The Advanced Authentication server validates the user’s credentials.
3. After validating the credentials, the Advanced Authentication server sends a request to the BankID app.
   - Click Identify on the Mobile app.
   - Click Verify my identity on the Desktop app.
5. The Security code is sent to the BankID server to validate.
6. The BankID server validates the authentication and the endpoint gets authenticated.

To configure the BankID method, perform the following steps:

NOTE: Ensure that you have the BankID client SSL certificate as a pre-requisite.

1. Click Browse then select the client SSL certificate from the local drive.
   - The certificate must be in PKCS12 format.
2. Specify Private key password.
3. Set Enable Test Mode to ON, to allow the user to test the authenticator with valid test BankID.
   - If you set this option to OFF, users must use valid production BankID to enroll the authenticator.
4. Click Save.
Bluetooth

In the Bluetooth method, you can enroll your smartphone or a mobile device. For example, Bob wants to be authenticated through the Bluetooth method. He enrolls the Bluetooth method on the Advanced Authentication Self-Service portal. He can get authenticated with the Bluetooth method only when his smartphone is in the range.

By default, the Enable reaction on device removal option is enabled. When this option is enabled and a user tries to logs in to Windows using Bluetooth, Windows gets locked automatically in the following scenario:

- When the Bluetooth device is disabled
- When the Bluetooth device is out of range

**NOTE:** It is recommended to combine the Bluetooth method with another authentication method in a chain to enhance the security.

Card

The Card authentication happens when a user places a contactless card on a card reader.

Advanced Authentication supports the Microsoft policy Interactive logon: Smart card removal behavior that allows you to specify an action on the card event. You can configure the policy to perform a force log off or lock a user session when a user places a card on the reader. Only Microsoft Windows supports this policy.

By default, the Enable Tap&Go option is disabled. When this option is disabled, a card must be placed on the reader when a user logs in. When the user removes the card from the reader, the Windows Client runs an action that is specified in the Interactive logon: Smart card removal behavior policy. When you set this option to ON, users can tap a card to perform the following actions (depending on the Interactive logon: Smart card removal behavior policy) without keeping their cards on the reader:

- To log in
- To lock a session
- To log off

**NOTE:** The policy is supported for Microsoft Windows only and it is not supported for the PKI authenticators.

Email OTP

In the Email OTP authentication method, the server sends an email with a one-time password (OTP) to the user's e-mail address. The user must specify the OTP on the device where the user needs to get authenticated. It is a best practice to use the Email OTP authentication method with other methods such as Password or LDAP Password to achieve multi-factor authentication and to prohibit malicious users from sending SPAM mails to a user’s email box with authentication requests.

To configure the Email OTP method, specify the following details:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OTP period</td>
<td>Lifetime of an OTP token in seconds. The default OTP period is 120 seconds. Maximum value for the OTP period is 86400 seconds.</td>
</tr>
<tr>
<td>OTP format</td>
<td>Length of an OTP token. The default value is 6 digits.</td>
</tr>
<tr>
<td>Subject</td>
<td>Subject of the mail.</td>
</tr>
<tr>
<td>Format</td>
<td>Format of an email message. The default format is Plain Text. The HTML format allows to use embedded images. You can specify an HTML format of the message in HTML.</td>
</tr>
<tr>
<td>Body</td>
<td>For the Plain Text format, you can specify the following variables:</td>
</tr>
<tr>
<td></td>
<td>• {user}: Username.</td>
</tr>
<tr>
<td></td>
<td>• {endpoint}: Device that a user authenticates to.</td>
</tr>
<tr>
<td></td>
<td>• {event}: Name of the event where the user is trying to authenticate to.</td>
</tr>
<tr>
<td></td>
<td>• {otp}: One-Time-Password to be sent to the user.</td>
</tr>
</tbody>
</table>

Allow to override email address

Option that allows to prevent users from providing an email address that is not registered in the LDAP repository. The option is set to ON by default. Set to OFF to prevent users to specify a different email address during the enrollment.

Allow user enrollment without e-mail

Option to configure settings for the user to enroll the Email OTP authenticator without an email in the repository.

Set this option to OFF to ensure that a user does not enroll the Email OTP authenticator without an email. The user gets an error message that you can specify in Error message.

Set this option to ON to allow the user to enroll the Email OTP authenticator without an email.

Emergency Password

The Emergency Password method facilitates the use of a temporary password for users if they lose a smartcard or forget their smartphone. Only a helpdesk administrator can enroll the Emergency Password method for users.

**WARNING:** An administrator can misuse this method by trying to access other user's account. Full administrator must be vigilant to select the right helpdesk administrators.
To configure the Emergency Password method, specify the following details:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum password length</td>
<td>The length of the password must be at least five characters long.</td>
</tr>
<tr>
<td>Password age (days)</td>
<td>The validity period of a password. The default value is 3 days.</td>
</tr>
<tr>
<td>Max logins</td>
<td>The maximum number of login attempts that a user can perform before the password gets expired. The default value is 10.</td>
</tr>
<tr>
<td>Complexity requirements</td>
<td>Set to ON to enforce users creating a complex password. Password must meet the following requirements:</td>
</tr>
<tr>
<td></td>
<td> Contains at least one uppercase character</td>
</tr>
<tr>
<td></td>
<td> Contains at least one lowercase character</td>
</tr>
<tr>
<td></td>
<td> Contains at least one digit</td>
</tr>
<tr>
<td></td>
<td> Contains at least one special character</td>
</tr>
<tr>
<td>Allow change options during enrollment</td>
<td>When set to ON, this option allows a helpdesk administrator to set Start date, End date, and Maximum logons manually in the Helpdesk portal. This manual configuration overrides the settings in the Emergency Password method.</td>
</tr>
</tbody>
</table>

**Facial Recognition**

Advanced Authentication provides advanced biometric authentication with the Facial Recognition method. This method allows users to get automatically authenticated by presenting their face. The image of the face is captured by an integrated or external camera and recorded by the Microsoft API server, when the user enrolls the method. When the user tries to authenticate on an application, the recorded image is compared with the actual image. If the images match, the user is authenticated.

**IMPORTANT:** It is recommended to combine the Facial recognition method with another method in a chain to enhance security.

You can configure the following settings for the Facial recognition method:

- “Generating Access Key and Endpoint URL” on page 49
- “Configuring Facial Recognition Method” on page 49

**WARNING:** To use the Facial recognition method for OAuth 2.0 and SAML 2.0 integrations, you must have the Advanced Authentication Device Service installed.
Generating Access Key and Endpoint URL

Before you configure the Facial Recognition method, you must generate the Access Key and Endpoint URL from the Microsoft Cognitive Services (https://azure.microsoft.com/en-in/services/cognitive-services/).

To generate the Access Key and Endpoint URL, perform the following steps:

1. Click Get API against Face API.
2. Agree to the license agreement.
3. Login with the preferred credentials.
4. Capture the Access Key and Endpoint URL for the Face API.

While generating the access key for the Face API, two keys are displayed. You can use anyone of the two keys.

Configuring Facial Recognition Method

To configure the Facial Recognition method, perform the following steps:

1. Click Methods > Facial Recognition.
2. Specify the Access Key that you have generated in the Microsoft Cognitive Services. This key is used while authenticating the user.
3. Specify the Endpoint URL. This URL is location based.

NOTE

- For a better quality of recognition, you must use cameras with a high definition of 720p and above.
- During enrollment, the captured images are placed on Microsoft servers and Microsoft Cognitive Services returns only the Face ID to Advanced Authentication. The Advanced Authentication stores this Face ID as enrolled authenticator. Therefore, when you change to another Access Key, the related enrollments are lost.
- This method is not supported for cache of Windows Client, Mac OS X Client, and Linux PAM Client.

FIDO 2.0

The FIDO 2.0 method facilitates users to use the devices that comply with FIDO standards for authenticating to any web-based environment. The devices can be built-into the platform or external devices connected through USB. The FIDO 2.0 method uses the Web Authentication (WebAuthn) API, and Client to Authenticator Protocol (CTAP). The WebAuthn enables strong authentication with public key cryptography and allows password-less authentication.

NOTE: Advanced Authentication FIDO 2.0 method supports the following:

- Firefox and Google Chrome browsers with the U2F device
- Microsoft Edge browser with Windows Hello authentication
While you use Google Chrome browser, it is required to set a valid domain name for your Advanced Authentication server rather than an IP address.

If users have enrolled the FIDO 2.0 method using the Windows Hello in Microsoft Edge 17 or earlier supported browser versions then they must authenticate using the same browser. After upgrading to the latest version of Edge that supports the FIDO 2.0 standards, users must re-enroll the FIDO 2.0 method.


An Example of Authenticating with the FIDO 2.0 Method

Thomas, an end user, has enrolled the FIDO 2.0 method in the Advanced Authentication Self-Service portal by using the FIDO compliant U2F token. He wants to authenticate to the mycompany.com website. When he opens the browser and follows the prompts to access the website. Then, he is required to touch the token when there is a flash. Thomas is validated with the device and gets authenticated to mycompany.com.

Fingerprint

The Fingerprint method is one of the strongest biometric authentication methods of Advanced Authentication. Users can authenticate with methods such as Password (something they know) and Fingerprint (something they are) for multi-factor authentication. Users need to place their finger on a fingerprint scanner to enroll and authenticate.

To configure the Fingerprint method, perform the following steps:

1. Set the Similarity score threshold by moving the slider to the desired score.

   **NOTE:** Default and recommended value for Similarity score threshold is 50. Reducing the score may result in different fingerprints getting validated.

2. Select the number of fingers that a user must enroll.

   It is recommended to specify a number that is more than 1 because if a finger is injured, the user can use the other enrolled finger.

   **NOTE:** If you want to allow the use of multi-finger reader for enrollment, ensure to select the number of fingers to be enrolled as 4, 6, 8, or 10.

3. Select the number of scans required for enrollee’s each finger.

   **NOTE:** To improve the quality of the fingerprint enrollment, it is recommended to have multiple captures. The total number of captures including all the enrolled fingers must not exceed 25.

4. Set Enable multi-finger reader to enroll to ON, to allow users to enroll the Fingerprint method using the Green Bit DactyScan84c multi-finger reader. Users can set Use multi-finger reader for enrollment to ON and enroll with the multi-finger reader on the Self-Service portal. The Green Bit DactyScan84c device can scan one of the following fingers combination at a time:
   - Four fingers of the right hand
   - Four fingers of the left hand
   - Two thumbs
To enforce the users to scan fingers using the Green Bit DactyScan84c reader, set Force to use multi-finger reader to ON.

5 Set Specify fingers during enrollment to ON, if you want to enforce selected fingers for a user to enroll.

6 Select the preferred fingers to enroll from the Selected fingers list.

7 Set Enable Duress finger configuration to ON, to allow users to assign one of the enrolled fingers as duress. In case of emergency or under a threat, user can authenticate with the duress finger. Authentication with the duress finger triggers an alert notification to the configured email address and phone number.

In the Alert Configuration section, specify the following details to configure the alert notification that is to be sent to the preferred email address and phone number:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Email Alert Settings</td>
<td></td>
</tr>
<tr>
<td>Email Recipient</td>
<td>The email address of recipient to whom you want to send the email alert.</td>
</tr>
<tr>
<td>Email Alert Subject</td>
<td>Subject of the email alert.</td>
</tr>
<tr>
<td>Format</td>
<td>Format of email alert. Plain Text is the default format. Other available option is HTML.</td>
</tr>
<tr>
<td></td>
<td>If you select HTML format, specify the message in HTML.</td>
</tr>
<tr>
<td>Email Alert Body</td>
<td>Body of email alert. You can specify the following variables:</td>
</tr>
<tr>
<td></td>
<td>• {user}: Username.</td>
</tr>
<tr>
<td></td>
<td>• {endpoint}: Device that a user authenticates to.</td>
</tr>
<tr>
<td></td>
<td>• {event}: Name of the event where the user is trying to authenticate to.</td>
</tr>
<tr>
<td>SMS Alert Settings</td>
<td></td>
</tr>
<tr>
<td>SMS Recipient</td>
<td>Phone number of recipient to whom you want to send the SMS alert.</td>
</tr>
<tr>
<td>SMS Alert Body</td>
<td>Text in the SMS that is sent to the recipient. You can specify the following variables:</td>
</tr>
<tr>
<td></td>
<td>• {user}: Username.</td>
</tr>
<tr>
<td></td>
<td>• {endpoint}: Device that a user authenticates to.</td>
</tr>
<tr>
<td></td>
<td>• {event}: Name of the event where the user is trying to authenticate to.</td>
</tr>
</tbody>
</table>

8 Click Save.

NOTE: Ensure that you configure the Mail Sender and SMS Sender policies with the sender details that are required to send an alert.

Example 1: Enrolling Multiple Fingers and Authenticating with One of the Enrolled Fingers

Consider Thomas, an administrator has performed the following steps to enforce users to enroll the Fingerprint method using the Greenbit DactyScan84c device. Users can authenticate to Linux workstation with the Fingerprint method.

1. Set Force to use multi-finger reader to ON in the Fingerprint method.
2. Created a chain with the Fingerprint method and added another preferred method such as LDAP password or Password.
3. Mapped the chain to the Linux Logon event.

Paul, an end user, logs in to the Self Service portal and clicks on the Fingerprint icon. He selects the four fingers of Right hand and enrolls using the Green Bit DactyScan device. After enrollment, Paul authenticates to his Linux workstation with the Nitgen device using one of the enrolled fingers. He gets authenticated successfully.

Example 2: Authenticating with a Duress Finger During an Emergency Situation

Consider Thomas, an administrator has performed the following steps to assign an enrolled finger as duress:

1. Set Enable Duress finger configuration to ON in the Fingerprint method.
2. Configured Alert Configuration with the alert notification text, mail address and phone number of a network security officer to send email and SMS.
3. Created a chain with the Fingerprint method along with preferred methods such as LDAP password and Password. Assigned the chain to Networks group.
4. Mapped the chain to the Linux logon event. Mail server is hosted on the Linux workstation.

Paul, a network staff, logs in to the Self Service portal and clicks on the Fingerprint icon. He enrolls the middle, index, ring and little fingers of the left hand. Later, he selects Left index from Assign Duress Finger drop down.

Assume, on an unfortunate day, a miscreant forcibly enters the organization and threatens Paul to authenticate to the Linux workstation. In this situation, Paul can use the duress finger (Left index finger) for authentication which triggers an alert notification to configured security personnel, who will take the necessary action.

LDAP Password

In the LDAP Password method, the Advanced Authentication client retrieves password that is stored in the user repository from the Advanced Authentication server.

To configure LDAP Password method, perform the following steps:

1. Set Save LDAP password to ON, the prompt for LDAP password synchronization is displayed only for the first time until the password is changed or reset.
   If you set this option to OFF:
   • If the LDAP Password method is included in a chain, users will be prompted for synchronization each time.
   • If the LDAP Password method is not included in a chain, users will not be prompted for synchronization.
2. Set Enable SSPR integration to ON if you want to enable the Self Service Password Reset integration for Advanced Authentication web portals.
   • Specify the SSPR link text. This link is displayed on the login page where user specifies the LDAP Password.
   • Specify the SSPR URL. This URL points to the Self Service Password Reset portal.
3. Set Enable cached logon to ON to validate user specified password with password stored (cached) in the Advanced Authentication server during authentication.
When the Enable cached logon option is set to **OFF** (default behavior), the Advanced Authentication server always contacts the LDAP server to validate the user password. It may cause performance issues.

If the user password does not match with the stored password or password is not stored on the Advanced Authentication server, then cached value gets reset and Advanced Authentication server contacts the LDAP server to validate the user password.

If the user specified password matches the cached password, the Advanced Authentication server validates user password with LDAP server in the background. If the validation failed, the password stored on Advanced Authentication Server gets reset, so next login will be without cache.

**NOTE:** The Enable cached logon option works only if any one of the following setting is set to **ON**:

- Save LDAP password in the LDAP Password method.
- Enable local caching in the Cache Options policy.

4 Click **Save**.

LDAP password is stored on the Advanced Authentication server at the following two places:

1. User data: It is used for OS logon (Windows Client, Mac OS X Client, and Linux PAM Client) and is stored when Save LDAP password option in LDAP Password method is set to **ON**.
2. LDAP password authenticator: It is used while using cached logon. The password is stored when the Enable local caching option is set to **ON** in the Cache Options Policy.

**OATH OTP**

OATH (Initiative for Open Authentication) is an industry-wide collaboration to develop an open reference architecture using open standards to promote the adoption of strong authentication using OTP.

Advanced Authentication supports the following two different types of OATH OTP:

- **HOTP**
- **TOTP**

You can configure the following settings for the OATH methods:

- Importing PSKC or CSV Files
- CSV File Format To Import OATH Compliant Tokens

**HOTP**

HOTP is a counter based one time password. To configure the HOTP authenticator, you can specify the following parameters:

- **OTP format:** The number of digits in the OTP token. The default value is 6 digits. The value must be the same as of the tokens you are using.
- **OTP window:** The size of OTP window defines number of valid OTP for authentication. When the counters are out of sync, this parameter determines the difference between the counter on the token and the server. Based on the difference, the server can recalculate the next OTP value to validate with the OTP received from the token. The server stores the last counter value (C) for
which the user has provided a valid password. While verifying a new OTP from the token, the server validates C+1, C+2... until one of the OTP is identical, or till C+w, where w represents the OTP window.

You can use the HOTP token such as Yubikey token to access not only Advanced Authentication, but also some websites or third-party services. After each use or when users press the token button accidentally, the HOTP counter on the token is increased by 1. Therefore, the counter will be out of sync between the token and Advanced Authentication server.

For example, if the OTP window is set to 10 (by default), and the current counter value of the server is 100, then any OTP generated from the token with a counter value from 100 to 110 are valid for authentication.

**WARNING:** Do not increase the HOTP window value to more than 100 as it may decrease the security by causing false matches.

During enrollment or HOTP counter synchronization in the Self-Service portal, **Enrollment HOTP window** that has a value of 100,000 is used. This is helps in the following:

- HOTP tokens may be used for a long period before the enrollment in Advanced Authentication and the value is unknown and can be equal to some thousands.
- Secure because users must provide 3 consequent HOTPs.

### Configuring Yubikey for Advanced Authentication Server

1. Download and install the Yubikey Personalization Tool from Yubico.

   To download the Yubikey Personalization Tool, see the Yubico website (https://www.yubico.com/products/services-software/download/yubikey-personalization-tools/).

2. Insert the Yubikey token.

   Ensure that the token is recognized. The recognition is indicated by a message *Yubikey is inserted* at the top-right corner of the Personalization tool.

3. Select **OATH-HOTP mode**.

4. Select **Configuration Slot 1**, generate the **OATH Token Identifier** and **Secret Key**.

5. In **Logging Settings**, select **Log configuration output**.

6. Select **Traditional format** or **Yubico format**.

7. Click **Write Configuration** and save the CSV file.

For information about how to enroll the HOTP method, see “HOTP ” in the *Advanced Authentication-User* guide.

### TOTP

TOTP is a time based one time password. To configure the TOTP authenticator, you can specify the following parameters:

- **OTP period (sec)**: The value to specify how often a new OTP is generated. The default value is 30 seconds. The maximum value for the OTP period is 360 seconds.

- **OTP format**: The number of digits in the OTP token. The default value is 6 digits. The value must be the same as the tokens you are using.

- **OTP window**: The value to specify the periods used by Advanced Authentication server for TOTP generation. For example, if you have a period of 30 and a window of 4, then the token is valid for 2*30 seconds before current time and 2*30 seconds after current time, which is ±2
minutes. These configurations are used because time can be out-of-sync between the token and the server and may impact the authentication. The maximum value for the OTP window is 64 periods.

**IMPORTANT:** It is not recommended to use an OTP window equal to 32 and higher for 4-digit OTP because it reduces security.

- **Google Authenticator format of QR code (Key URI):** Option to display the QR code for the TOTP enrollment of the software token in a format that is compatible with the Google Authenticator, Microsoft Authenticator, or the NetIQ Auth apps. When you disable the option, the displayed QR code can be scanned only with the NetIQ Auth smartphone app. Enable the option to allow enrollment with the Google Authenticator or Microsoft Authenticator apps. The QR code of Google Authenticator format can also be scanned with the NetIQ Auth app (supported by the last iOS and Android apps).

**IMPORTANT:** OTP format must be set to 6 digits when you use the Google Authenticator format of QR code.

- **Allow manual enrollment:** When you enable the option, the Specify the TOTP secret manually section is displayed on the TOTP enrollment page of the Self-Service portal with the following parameters: Secret, Period, and Google Authenticator format of secret (Base32). By default, the option is disabled and the settings are hidden. Enabling the option may result in security risks.

You must perform the following tasks to allow the users to enroll TOTP method using the Desktop OTP tool:

- Generating an Enrollment Link
- Sending an Enrollment Link Through Email

### Generating an Enrollment Link

Users can click the enrollment link to enroll the TOTP authenticator automatically on the Desktop OTP tool and following the further steps as described in Desktop OTP Tool. To generate an enrollment link, you can encode the server URL, tenant ID, and category name to the Base64 format using any online tool. The generated link is then sent to the users through the email to access the Desktop OTP tool and enroll the TOTP authenticator. The users can create an account on the tool to enroll the TOTP authenticator in the Self-Service portal.

**To generate the enrollment link in the Base64 format, perform the following steps:**

1. To encode use the details such as server URL, tenant ID and category name in the following format:

   ```json
   {"server_url":"<domain-name>","tenant_name":"<tenant-name>","category_name": "HOME"}
   ```

   For example, `{"server_url": "aafserver.company.com", "tenant_name": "netiq", "category_name": "HOME"}`

   You can specify the preferred category name for `category_name` parameter if you have added categories in the Event Categories policy. You can remove the parameter `category_name`, if you have not added any category.

2. Encode the value including `}` to Base64 (charset: UTF-8) format.

   For example, the encoded link is displayed as:
Configuring Advanced Authentication Server Appliance

**Sending an Enrollment Link Through Email**

1. Compose an email with the subject and body.
   
   For example, specify TOTP Enrollment Link in the Subject and body as follows:
   
   Hi Users, Click here to enroll for the TOTP authenticator using the Desktop OTP tool.

2. Right click on the preferred text and select **Hyperlink**.

3. Specify the encoded link and prefix **aaf-otp** in **Address**.
   
   For example, aaf-otp:eyJzZXJ2ZXJfdXJsIjogImFhZnNlcnZlc5jb21wYW55LmNvbSIslCIJ0ZW5hbWVfbmFtZSI6Im5ldGlwIiwicGFja19fcmVzc2VjIjoic2NoYXJ0clwiLCJ0TW9kZW1fbmFtZSI6IiJ9

4. Specify the email address of the preferred users in **To** then click **Send**.
   
   User can click the hyperlink to open the Desktop OTP automatically.

**Importing PSKC or CSV Files**

You can import the **PSKC** or **CSV** files. These token files contain token information. To import these files, perform the following steps:

1. Click the **OATH Token** tab.

2. Click **Add**.

3. Click **Browse** and select a **PSKC** or **CSV** file.

4. Choose a **File type**. The options are:
   
   - **OATH compliant PSKC**: This file type must be compliant with OAuth. For example, HID OATH TOTP compliant tokens.
   
   - **OATH csv**: This file type must contain the format as described in CSV File Format To Import OATH Compliant Tokens. You cannot use the YubiKey CSV files.
   
   - **Yubico csv**: In this file type, you must use one of the supported Log configuration output (see YubiKey Personalization Tool > Settings tab > Logging Settings) formats with comma as a delimiter.
     
     - **Traditional format**: In this file type, OATH Token Identifier must be enabled.
     
     - **Yubico format**: This file type is supported only for HOTP Length set to 6 Digits and OATH Token Identifier set to All numeric.

   **IMPORTANT**: Moving Factor Seed must not exceed 100000.

5. Add the encrypted **PSKC** files. For this, select **Password** or **Pre-shared key** in **PSKC file encryption type** and provide the information. You can select **Not encrypted**, if the **PSKC** file is not encrypted with either the password or key.

6. Click **Upload** to import tokens from the file.

**NOTE**: Advanced Authentication receives an **OTP format** from the imported tokens file and stores the information in the enrolled authenticator. Therefore, you need not change the default value of **OTP format** on the **Edit Method** tab.
When the tokens are imported, you can see the list and you must assign the tokens to users. This can be done in the following two ways:

- Click **Edit** next to the token and select **Owner** and click **Save**.
- A user can self-enroll a token in the Self-Service portal. Administrator must let the user know an appropriate value from the **Serial** column for the self-enrollment.

**NOTE:** Tenancy settings are not supported for the OATH tokens. Therefore, the configurations in the **OATH Tokens** tab cannot be enforced on tenant administrators.

### CSV File Format To Import OATH Compliant Tokens

A **CSV** file, which is imported as **OATH csv** file in the **Administration portal > Methods > OATH OTP > OATH Tokens** tab, must contain fields with the following parameters:

- Token’s serial number
- Token’s seed
- (Optional) Type of the token: TOTP or HOTP (by default HOTP)
- (Optional) OTP length (default value is 6 digits)
- (Optional) Time step (default value is 30 seconds)

Comma is a delimiter.

The following is an example of a CSV file:

```
Token001, 15d2fa517d3c6b791bd4cc2044c241429307001f
Token002, 8c557fc050721037fd31e1d3345b5d3263263e0f, totp, 8
Token003, 658208efea5ac49d5331ba781e66f2c808ccc8e, hotp, 6
Token004, 89f0dfe1c90379da6a11aaca2fc1070f606efe36, totp, 6, 60
```

**IMPORTANT:** For the YubiKey tokens, you must use the traditional format of the CSV (check **YubiKey Personalization Tool > Settings tab > Logging Settings**) with comma as a delimiter. Use Yubico csv file type (**Advanced Authentication Administration portal > Methods > OATH OTP > OATH Tokens**).

### Password

In the **Password** authentication method, you can configure security options for passwords that are stored in the appliance. For example, the **local/admin** user who does not have an LDAP Password can use this option.

**NOTE:** Do not use the **Password** method in chains that contain only one factor. You must always combine the **Password** method with other factors.

You can configure the following options for the **Password** method:

- **Minimum password length**: The maximum length of the password.
- **Maximum password age**: The validity period of the password. The default value is 42 days. If you set the value to 0, the password never expires.
Complexity requirements: Option to enable users to create a complex and not easily detectable password. Set to ON to enable this option. Password must meet the following requirements:

- Contains at least one uppercase character
- Contains at least one lowercase character
- Contains at least one digit
- Contains at least one special character

IMPORTANT: Advanced Authentication does not generate notifications about the password expiry. After the password expires, the local administrator cannot sign-in to the Administration portal and users using this method cannot get authenticated. However, an administrator and a user can change their passwords in the Self-Service portal.

PKI

The Public Key Infrastructure (PKI) creates, stores, and distributes digital certificates. These certificates are used to verify whether a particular public key belongs to a specific entity.

Advanced Authentication supports the following two forms of PKI authentication:

- PKI Device
- Virtual Smartcard

PKI Device

PKI device stores the digital certificates and private keys securely. It uses the PKI infrastructure to store personal details of user such as private key, PIN, and digital certificate.

You can configure the following settings for the PKI method:

- “Adding the Trusted Root Certificates” on page 58
- “Disabling the Key-Pair Option” on page 60

Adding the Trusted Root Certificates

You must upload the trusted root certificates for the PKI method. These certificates must meet the following requirements:

- Root CA certificate is in the .pem format.
- All certificates in the certification path (except Root CA) contain AIA and CDP http link to check revocation status.
- The certificate for PKI device contains a key pair: public and private key in the x509 format. The certificates that do not comply with the requirements are ignored and hidden during enrollment.

For more information, see Single Tier PKI Hierarchy Deployment and Two Tier PKI Hierarchy Deployment.

To upload a new trusted root certificate, perform the following steps:

1. Click Add in the Edit Method page of PKI.
2. Click Browse.
Choose a .pem certificate file and click Upload.

Click Save.

NOTE: You must upload only the Root CA on appliance.

You can configure the PKI method (with certificates) in one of the following ways:

- Standalone Root CA
- Subordinate CA

NOTE: Advanced Authentication supports the p7b format of parent certificates. These p7b format files can contain certificates and chain certificates, but not the private key. They are Base64 encoded ASCII files with extensions .p7b or .p7c.

Configuring the Environment for a Standalone Root CA

1. Install Web Server (IIS) Role.
2. Create the CertEnroll Folder and grant Share & NTFS permissions to the Cert Publishers group.
3. Create CertEnroll Virtual Directory in IIS.
4. Enable Double Escaping on IIS Server.
5. Install Enterprise Root CA using Server Manager.
6. Enable Object Access Auditing on CA.
7. Configure the AIA and CDP.
8. Publish the Root CA Certificate to AIA.
9. Export Root CA in .der format and convert the format to .pem.
10. Export personal certificate (that was signed by Root CA) with private key and place it on a PKI device.

Configuring the Environment for a Subordinate CA

1. Install Web Server (IIS) Role.
2. Create the CertEnroll Folder and grant Share & NTFS permissions to Cert Publishers group.
3. Create CertEnroll Virtual Directory in IIS.
4. Enable Double Escaping on IIS Server.
5. Install the Standalone Offline Root CA.
6. Create a CAPolicy.inf for the standalone offline root CA.
7. Installing the Standalone Offline Root CA.
8. Enable Auditing on the Root CA.
9. Configure the AIA and CDP.
10. Install Enterprise Issuing CA.
11. Create CAPolicy.inf for Enterprise Root CA.
12. Publish the Root CA Certificate and CRL.
13. Install Subordinate Issuing CA.
14. Submit the Request and Issue subordinate Issuing CA Certificate.
15 Install the subordinate Issuing CA Certificate.
17 Enable Auditing on the Issuing CA.
18 Configure the AIA and CDP.
19 Install and configure the Online Responder Role Service.
20 Add the OCSP URL to the subordinate Issuing CA.
21 Configure and publish the OCSP Response Signing Certificate on the subordinate Issuing CA.
22 Configure Revocation Configuration on the Online Responder.
23 Configure Group Policy to provide the OCSP URL for the subordinate Issuing CA.
24 Export Root CA in .der format and convert the format to .pem.
25 Export personal certificate (that was signed by subordinate CA) with private key and place it on a PKI device.

**Disabling the Key-Pair Option**

The Allow key-pair option is enabled by default. This indicates that the enrollment of the PKI method can be done with either the CA certificates or through the key-pair generation. However, you can disable the key-pair based enrollment of the PKI device and enforce PKI enrollment only using a user certificate issued by the CA. To disable this option, set Allow key-pair to OFF.

**Virtual Smartcard**

Virtual Smartcard is an extension of PKI method. Advanced Authentication allows users to enroll the PKI method using a virtual smartcard that is imported to the browser on the user’s system and used for authentication. Virtual smartcard is a certificate that contains information such as digital signature, expiration date, name of user, name of CA (Certificate Authority), and can be used in client SSL certificate. Typically, the certificate is available in .pfx format. The information available in the virtual smartcard is used to authenticate the user to any web environment.

**NOTE:** The virtual smartcard supports authentication to the OAuth 2.0 and SAML 2.0 events. The virtual smartcard does not support authentication to Advanced Authentication portals, such as Administration, Helpdesk, Self-Service, and Reporting.

To configure the virtual smartcard, perform the following steps:

**NOTE:** Before you configure the virtual smartcard support for the SAML 2.0 events, ensure to specify the Identity Provider’s URL in format https://webauth.domain_name in the Web Authentication policy. Later, save the settings before downloading the SAML 2.0 metadata file.

**NOTE:** Before you configure virtual smartcard support for the PKI method, ensure to perform the following tasks:

- Resolve the IP address of Advanced Authentication server with the following host names on the DNS server:
  - `<aaserver_ip_address> <aaserver_hostname>`
  - `<aaserver_ip_address> <webauth.aaserver_hostname>`
Define the following attributes in the third-party application that you want to integrate with Advanced Authentication server:

- `authorization_endpoint = https://webauth.aaserver_hostname/osp/a/TOP/auth/oauth2/grant`
- `token_endpoint = https://webauth.aaserver_hostname/osp/a/TOP/auth/oauth2/getattributes`

1. Configure the following settings in the HTTPS Options policy:
   - Set Enable Client SSL for Webauth Service to ON and upload Root CA certificate in the .pem format that is used by the Web server.
   - Set Enable auto enrollment based on certificate to ON. This enables you to allow users to auto-enroll the PKI method using virtual smartcard for the OAuth 2.0 and SAML 2.0 events.

   **NOTE:** The manual enrollment of the PKI method using the virtual smartcard is not supported. Therefore, it is required to set Enable auto enrollment based on certificate to ON in the HTTPS Options. With this configuration, the users can auto-enroll PKI method using virtual smartcard when they access OAuth 2.0 event for the first time and select a valid certificate. This auto-enrollment happens irrespective of enrollment status of other method(s) that are available with the PKI method in the same authentication chain.

   To allow a user to login to the OAuth 2.0 and SAML 2.0 events before auto-enrolling the PKI method, ensure to add at least one more chain to the event (for example, a chain with only the LDAP Password method) below the PKI chain. The user must enroll all method(s) of new chain. During the first login attempt, the PKI method using the virtual smartcard gets enrolled automatically. For the subsequent log ins, the top chain in the list (which is PKI) is selected and user is authenticated automatically.

3. Import the client SSL certificate to the users browser.

   **NOTE:** The procedure to import the client SSL certificate varies on each browser.

   For more information about how to import the client SSL certificate to the Chrome browser, see Importing Client SSL Certificate to a Certificate Store.

### An Example of Auto-enrolling PKI Method with the Virtual Smartcard

Consider the administrator has performed the following steps to allow auto-enrollment of the PKI method using the virtual smartcard:

- Created a chain with the PKI method and another chain with preferred methods such as LDAP password and Password.
- Mapped the chain to the OAuth 2 event.
- Configure the following settings in the HTTPS options policy:
  - Set Enable SSL Client Certificate to ON and uploaded a valid CA certificate.
  - Set Enable Auto Enrollment based on certificate to ON.
- Imported the client certificate to the user’s browser in the .pfx format containing details, such as digital signature, expiration date, name of user, name of CA and so on.

Mark, an end user, wants to auto-enroll the PKI method using the virtual smartcard. When he tries to access the somecompany.com website, the user name stored in the certificate gets filled in the user name field in the login form automatically. Mark is required to select the preferred certificate to
validate his identity in the User Identification Request dialog box. Then, Mark must specify LDAP details for additional validation. If the specified details are valid, Mark gets auto-enrolled to the PKI method using the virtual smartcard without physical PKI token.

During subsequent logins, Mark may experience one of the following scenario:

- If there is a chain with only PKI method associated to the web authentication event, then Mark gets authenticated automatically.
- If there are more than one chain associated to the web authentication event, then Mark is prompted with the list of chains that contains PKI in addition to other available chains. In this case, he can select the chain with only PKI method to authenticate automatically or select preferred chain and provide corresponding details to authenticate successfully.

### Importing Client SSL Certificate to a Certificate Store

To enable and achieve the virtual smartcard authentication to the web environment, it is required to import the Client SSL certificate to the browser.

**NOTE:** The procedure to import the client SSL certificate varies on each browser.

To import the client SSL certificate to Google Chrome browser, perform the following steps:

1. Navigate to Settings > Manage Settings.
   - The Certificates wizard is displayed.
2. Click Import and select the client SSL certificate.
   - Ensure that the certificate is in .pfx format.
3. Click Next and Finish.
   - A message Certificate has been imported successfully is displayed.

### RADIUS Client

In the RADIUS Client method, Advanced Authentication forwards the authentication request to a third-party RADIUS server. This can be any RADIUS server. For example, you can use RADIUS Client as an authentication method when you have a token solution such as RSA or Vasco. You want to migrate users to Advanced Authentication with the flexibility that users can use the old tokens while the new users can use any of the other supported authentication methods.

You can configure the following options for the RADIUS Client method:

- **Send the repository name:** Option for a repository name to be used automatically with a username. For example, company\pjones. Set to ON to enable the option.
- **NAS Identifier:** An attribute that contains a string identifying the NAS originating the Access-Request. It is only used in Access-Request packets. Either NAS-IP-Address or NAS-Identifier must be present in an Access-Request packet.
- **Timeout:** Specify the number of seconds till when the RADIUS client waits for the RADIUS server to reply before prompting an error Connection time out. The default value is 5 seconds.
- **Retries count:** Specify the number of times, the RADIUS client tries to connect to the RADIUS server. If a connection is not established during the retry attempts, a message Failed to connect to the server is displayed. The default value is set to 3. If set to 0, the RADIUS client does not try to connect after the first unsuccessful attempt.
Specify servers per site: Option to configure the third-party RADIUS servers that are specific to a site. When set to ON, the sites available in the cluster are populated and you can add more than one servers to the preferred site.

When this option is set to OFF, you can add single third-party RADIUS server details that are applicable for all sites in the cluster by specifying the following details:

- **Server**: The Hostname or IP address of the third-party RADIUS server.
- **Secret**: The shared secret between the RADIUS server and Advanced Authentication.
- **Port**: The port to where the RADIUS authentication request is sent. The default port is 1812.

Security Questions

In Security Questions authentication method, an administrator can set up a series of predefined questions. A user must answer these questions to get authenticated. Security Questions are used when users forget their passwords.

Security questions are often easy to guess and can often bypass passwords. Therefore, Security Questions do not prove to be secure.

You must follow few guidelines to use this method. You must use Good security questions that meet five criteria. Ensure that the answers to a good security question are:

1. **Safe**: Cannot be guessed or researched.
2. **Stable**: Does not change over time.
3. **Memorable**: Can be remembered.
4. **Simple**: Precise, easy, and consistent.
5. **Many**: Has many possible answers.

Some examples of good, fair, and poor security questions according to goodsecurityquestions.com are as follows. For a full list of examples, see the website [http://goodsecurityquestions.com/](http://goodsecurityquestions.com/).

**GOOD**

- What is the first name of the person you first kissed?
- What is the last name of the teacher who gave you your first failing grade?
- What is the name of the place your wedding reception was held?
- In what city or town did you meet your spouse/partner?
- What was the make and model of your first car?

**FAIR**

- What was the name of your elementary / primary school?
- In what city or town does your nearest sibling live?
- What was the name of your first stuffed animal, doll, or action figure?
- What time of the day were you born? (hh:mm)
- What was your favorite place to visit as a child?

**POOR**

- What is your pet's name?
• In what year was your father born?
• In what county were you born?
• What is the color of your eyes?
• What is your favorite _____?

Configure the following options for the **Security Questions** method:

• **Minimum answer length**: The minimum number of characters an answer must contain.
• **Correct answers for logon**: The number of answers a user must answer correctly to get access.
• **Total questions for logon**: The number of questions that are presented to the user while authenticating.

For example, if the **Correct answers for logon** is set to 3 and the **Total questions for logon** is set to 5, the user needs to specify only 3 correct answers out of a set of 5 questions.

### Adding Questions

You can add questions based on your requirement. These questions can be translated in languages that are supported by the Advanced Authentication portals. For example, you set a security questions as **What is your pet name?**. While enrolling and authenticating, this question will be displayed in the language that the user selects in the portal.

To add questions, perform the following:

1. Click **Add** to add a question in the **Question** window.
2. Specify the question in **Question**.
3. You can specify the question to be translated in the required language.
   - This translated question is displayed in the portals and Clients based on the selected language.
4. Click the save icon to save the question related settings.

You can add more questions depending on the requirement.

Click **Save** to save the configuration settings for the Security questions method.

### Smartphone

Advanced Authentication provides the **Smartphone** method that facilitates users to authenticate through their Smartphone. The authentication happens through the NetIQ smartphone app to perform the out-of-band authentication. The out-of-band authentication is typically a two-factor authentication that requires a secondary verification through a separate communication channel along with the ID and password.

The authentication flow for the Smartphone method in Advanced Authentication is described in the following image.
A user wants to authenticate on an endpoint such as a laptop or a website with the Smartphone method. The following steps describe the authentication flow:

1. When the authentication request is initiated, the endpoint contacts the Advanced Authentication server.
2. The Advanced Authentication server validates the user’s credentials.
3. After validating the credentials, the Advanced Authentication server sends a push message to proxy.authasas.com.
4. Depending on the platform of the Smartphone, the server selects an appropriate push service and then forwards the push message to the Smartphone.
5. The push message is then delivered to the user’s Smartphone to inform that an authentication request has been initiated.
6. When the user opens the Smartphone app, the app reaches the Advanced Authentication server to validate if there is an authentication needed. The authentication is indicated by the Accept and Reject options. The user’s selection is then sent to the server.
7. Finally, the server validates the authentication and the endpoint gets authenticated.

HTTPS protocol is used for the communication.

This authentication method is recommended to use in combination with another method such as Password or LDAP Password to achieve multi-factor authentication and protect a user from getting SPAM push messages.
Access Configurations

The following are the configurations required for the Smartphone method.

- Advanced Authentication server must be accessible by the specified Server URL address from smartphones (HTTPS, outbound).
- Advanced Authentication server must have a permitted outbound connection to proxy.authasas.com (HTTPS).

Scenario for Authenticating with the Smartphone Method

Bob wants to authenticate on the myexample.com website. When he logs in to the website, the Smartphone authentication method sends a push message to Bob’s mobile phone. When he opens the Smartphone app installed on his phone, he sees Accept and Reject options. If he selects the Accept option, the authentication request is sent over the mobile network (secure) back to the Authentication framework. Without specifying an OTP code, Bob has been authenticated to myexample.com.

When your smartphone does not have a network connection, you can use a backup OTP as offline authentication.

Configuring Enrollment Link

Users can enroll the Smartphone method either by a QR code or through a link sent to their email or SMS. You as an administrator must configure the link and send it to all the users whom you want to enroll the authenticator. You can use one of the following links as per the requirement:

- \https://<public_external_url>/smartphone/enroll\n- \https://<public_external_url>/smartphone/enroll?category=cat1\n- \https://<public_external_url>/smartphone/enroll?tenant=t1\n- \https://<public_external_url>/smartphone/enroll?category=cat2&tenant=t1\n
Default category is default. Default tenant is TOP.

Configuring Smartphone Method

To configure the Smartphone method, specify the following details:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Push salt TTL</td>
<td>The lifetime of an authentication request sent to the smartphone.</td>
</tr>
<tr>
<td>Learn timeout</td>
<td>The time that is valid for the user to scan the QR code for enrollment.</td>
</tr>
<tr>
<td>Authentication salt TTL</td>
<td>The lifetime in which the out-of-band authentication needs to be accepted before authentication fails.</td>
</tr>
<tr>
<td>TOTP Length</td>
<td>The length of OTP token used for backup authentication.</td>
</tr>
<tr>
<td>TOTP step</td>
<td>The time a TOTP is displayed on a screen before the next OTP is generated.</td>
</tr>
<tr>
<td>TOTP time window</td>
<td>The time in seconds in which the TOTP entered is accepted.</td>
</tr>
<tr>
<td>Server URL</td>
<td>The URL of Advanced Authentication server to where the smartphone app connects for authentication.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Require PIN</td>
<td>Set to <strong>ON</strong> to enforce the Enable PIN setting for the Smartphone application. A user will not be able to edit the settings on the Smartphone.</td>
</tr>
<tr>
<td><strong>NOTE:</strong> If the PIN is not set, then the user is prompted to set the PIN during authentication.</td>
<td></td>
</tr>
<tr>
<td>Minimum PIN length if the PIN is required</td>
<td>The minimum length of the PIN. The available options are 4, 5, and 6.</td>
</tr>
<tr>
<td>Require biometrics</td>
<td>Set to <strong>ON</strong> to enforce the fingerprint setting for the Smartphone application. A user will not be able to edit the settings on the Smartphone.</td>
</tr>
<tr>
<td>Use image on mobile devices</td>
<td>Select the option to use a customized image on your Smartphone app. Browse the image. This image is displayed in the About screen of your Smartphone app.</td>
</tr>
<tr>
<td><strong>NOTE:</strong> The Require PIN, Require biometrics, and Use image on mobile devices policies are automatically applied on the smartphone if a user has an enrolled authenticator in the smartphone app and the app is open on one of the screens: Authentication Requests, Enrolled Authenticators, or Requests History. It takes 2 to 30 seconds to display the authentication request.</td>
<td></td>
</tr>
<tr>
<td>• If a user has configured a 4-digit PIN but a 6-digit PIN has been enforced by the administrator, then the user will be able to use the 4-digit PIN until the user decides to change the PIN.</td>
<td></td>
</tr>
<tr>
<td>• If <strong>Require biometrics</strong> is set in the policies, but a user’s device does not support fingerprint, the policy will not be applied for the device.</td>
<td></td>
</tr>
<tr>
<td>• If a user has authenticators enrolled for two different Advanced Authentication servers with different policies, then the policies are combined for the device and the most secure policies are applied for the app.</td>
<td></td>
</tr>
<tr>
<td>Disable offline authentication</td>
<td>Select this option to disable users from authenticating using the Smartphone TOTP. By default this option is disabled and users can login using Smartphone even when Smartphone is not connected to a network. Enabling this option will disallow users to use the One-Time Password of the Smartphone method to login to the offline mode.</td>
</tr>
<tr>
<td>Advanced Settings</td>
<td>These settings are optional.</td>
</tr>
<tr>
<td>• Vendor</td>
<td>If you have an approved vendor whose certificate is uploaded to proxy.authasas.com, you can specify the Vendor ID of your iOS app or specify the Google Project ID for your Android app. The push notifications will be sent only to the app whose Vendor name or Google Project ID matches with the app.</td>
</tr>
<tr>
<td>• Google project ID</td>
<td>By default Advanced Authentication works with the NetIQ Auth apps.</td>
</tr>
</tbody>
</table>
NOTE: To use geo-fencing, ensure that access to the location is enabled for the NetIQ Advanced Authentication app on the smartphone.

NOTE: You can customize the authentication request message that is displayed on the NetIQ Auth app using the Custom Messages policy.

For more information about customizing the authentication request message, see Customizing Authentication Request Message For Smartphone Method.

### SMS OTP

In the **SMS OTP** authentication method, a one time password (OTP) is sent with the SMS text to the user’s phone. The user receives the OTP and enters it on the device where the authentication is happening. The OTP must be used within a specific time frame. The OTPs delivered through text messages prevent phishing and malicious attacks. SMS OTP is recommended to be used with other methods, such as Password or LDAP Password.

**NOTE:** In the User’s settings of a repository, ensure that a phone number without extension is used. An SMS is not sent to the user’s mobile where the phone number contains an extension.
To configure the SMS OTP method, specify the following details:

- **OTP Period**: The lifetime of an OTP in seconds. The default value is 120 seconds. The maximum value for the OTP period is 360 seconds.
- **OTP format**: The number of digits in the OTP. The default value is 6.
- **Body**: The text in the SMS that is sent to the user. The following structure describes the text in the OTP:
  - `{user}`: Name of the user.
  - `{endpoint}`: Device the user is authenticating to.
  - `{event}`: Name of the event where the user is trying to authenticate to.
  - `{otp}`: One-Time Password.
- **User cell phone attribute**: The cell phone number of a user on which the OTP is sent through SMS. You can use custom attributes such as `mobile`, `homePhone`, `ipPhone`, and other attributes of a repository. You must define the attribute in “User Cell Phone Attributes” of the Repositories section.

**NOTE**: If you do not configure the attribute in the method settings, then the first attribute defined in the “User Cell Phone Attributes” section of Repository configuration is used when the user tries to authenticate. For example, if you define `mobile` as the first attribute in **User cell phone attribute** and do not configure the attribute in method settings of **SMS OTP**, then while authenticating, the first attribute, which is the `mobile` attribute, is used for the **SMS OTP** method authentication.

- **Allow overriding phone number**: Option that allows to prevent users from providing a phone number that is not registered in the LDAP repository. The option is set to **ON** by default. Set to **OFF** to prevent users to specify a different phone number during the enrollment.
- **Allow user enrollment without a phone**: Option to configure settings for the user to enroll the SMS OTP authenticator without a phone number in the repository.

  Set this option to **OFF** to ensure that a user does not enroll the SMS OTP authenticator without a phone. The user gets an error message that you can specify in **Error message**.

  Set this option to **ON** to allow the user to enroll the SMS OTP authenticator without a phone.

### Swisscom Mobile ID

In the **Swisscom Mobile ID** authentication method, a PKI-based mobile signature secure encryption technology is stored on a user’s SIM card. When the user tries to authenticate, the Swisscom Mobile ID is validated against the user’s mobile phone attribute in the repository. If the number is validated, the user gets authenticated.

To configure the Swisscom Mobile ID method, specify the following details:

- **Application Provider ID**: Identifier of the application provider.
- **Application Provider password**: Password of the application provider.
- **Swisscom Mobile ID service URL**: Interface of the Swisscom Mobile ID.
- **Notification message prefix**: Message that is displayed on the user’s mobile as a notification.

In addition, you can upload the Swisscom client certificates as follows:

1. Browse **Client SSL certificate**. The required certificate must be in a `.pem` format and self-signed with a private key.
2. Specify **Private key password** for the certificate.
3. Click **Save**.

**NOTE:** Users must activate the Mobile ID service for the **Swisscom SIM card**.

For more information about the Swisscom Mobile ID method, see the **Mobile ID Reference guide**.

## FIDO U2F

With the **FIDO U2F** authentication method, users can authenticate with the touch of a finger on the U2F device.

Advanced Authentication supports the Microsoft policy **Interactive logon: Smart card removal behavior** that allows you to specify an action on the U2F. You can configure the policy to perform a force log off or lock a session when a user removes the U2F device from a computer. This policy is supported for Windows only. When the user removes the U2F device from the computer, the Windows Client runs an action that is specified in the **Interactive logon: Smart card removal behavior policy**.

**IMPORTANT:** To use the FIDO U2F authentication for Access Manager in the **OAuth 2.0** event, you must configure an external web service to perform enrollment and authentication for one domain name. For more information, see Configuring a Web Server to Use the FIDO U2F Authentication.

The YubiKey tokens may flash with a delay when the token is initialized in a combination mode. For example, when authentication uses OTP and U2F methods. This may cause the users to wait for the token to flash before enrollment or authentication. Therefore, it is recommended to flash the tokens only in the U2F mode if the other modes are not needed.

You can configure the following settings for this method:

- "**Configuring the Certificate Settings**" on page 71
- "**Configuring Facets**" on page 71
- "**Configuring Yubikey for Advanced Authentication Server**" on page 72
- "**Configuring a Web Server to Use the FIDO U2F Authentication**" on page 72
Configuring the Certificate Settings

You can configure certificate settings for the FIDO U2F authentication method. By default, Advanced Authentication does not require the attestation certificate for authentication by the FIDO U2F compliant token. Ensure that you have a valid attestation certificate added for your FIDO U2F compliant token, when you configure this method. The Yubico and Feitian attestation certificates are pre-configured in the Advanced Authentication appliance.

To validate the attestation certificate for the FIDO U2F authentication, perform the following steps:

1. Set **Require attestation certificate** to **ON** to enable validation of attestation certificate.
2. Select the attestation certificate:
   1a. To use a default certificate, click **Add Default**.
   1b. To use a custom certificate instead of predefined device manufacturer certificate, perform the following steps:
      1b1. Click **Remove** next to the default attestation certificate to remove the certificate.
      1b2. Click **Add** to add a custom certificate.
      1b3. Click **Browse** then select the custom certificate and click **Upload**.
      The certificate must be in the **PEM** format.
   To restore the deleted attestation certificate, click **Add Default**.

Configuring Facets

You can add a list of facets for the FIDO U2F tokens to work on multiple sub-domains of a single domain.

Previously, the U2F RFC standards allowed authentication only on the domain name on which the enrollment was done. But with the FIDO U2F standards update (https://fidoalliance.org/specs/fido-u2f-v1.2-ps-20170411/fido-appid-and-facets-v1.2-ps-20170411.html), the FIDO alliance introduces facets that allows users to authenticate even on domains on which the enrollment is not done.

For example, if a user enrolls a token on https://.mytest.com and wants to get authenticated on https://u2f.mytest.com, you as an administrator can do this by adding https://u2f.mytest.com as a facet of the primary domain https://.mytest.com.

**WARNING:** Even if you are not using the facets, ensure to configure **Facets** to enable users to authenticate with the FIDO U2F method. If the **Facets** is not configured, then while authenticating with FIDO U2F, the user is prompted with a message **The visited URL doesn't match the application ID or it is not in use.**

To add facets, perform the following steps:

1. Expand **Facets settings**.
2. Specify the facet in **Facets**. For example, you can specify https://u2f.mytest.com.
3. Click **Add** to add more facets.
4. Specify the main URL in **App ID**. This ID is used to identify applications. For example, https://.mytest.com.
   If the **App ID** is left blank, the first facet is used as the App ID.
From the above example, if a user logs in to https://lu2f.mytest.com with the U2F token enrolled on https://.mytest.com, the browser sends a plain GET request to the https://URL/<tenant-ID/app-id.json URL and waits for the list of allowed facets (sub-domains). If the list is returned, browser allows the user to use token on the URLs specified in the Facets list.

5 Click Save.

NOTE: Facets are supported only on Google Chrome. The support for sub-domains is not stabilized in Chrome, therefore you might get an error message The visited URL doesn't match the application ID or it is not in use during enrollment and authentication.

### Configuring Yubikey for Advanced Authentication Server

1. Download and install the Yubikey Personalization Tool from Yubico.
   
   To download the Yubikey Personalization Tool, see the Yubico website (https://www.yubico.com/products/services-software/download/yubikey-personalization-tools/).

2. Insert the Yubikey token.
   
   Ensure that the token is recognized. The recognition is indicated by a message Yubikey is inserted at the top-right corner of the Personalization tool.

3. Select Yubico OTP mode.

4. Select Configuration Slot 1, generate the Public Identity, Private Identity, and Secret Key.

5. Click Write Configuration and specify the configurations.

6. Open the Advanced Authentication Self-Service portal and select U2F method.

7. Click Save to complete the enrollment.

### Configuring a Web Server to Use the FIDO U2F Authentication

This section is applicable for Debian 8 Jessie. The procedure may differ for other distributives.

This sections explains how to configure web server to use the FIDO U2F authentication in NetIQ Access Manager for the OAuth 2.0 event.

According to the FIDO U2F specification, both enrollment and authentication must be performed for one domain name. As NetIQ Access Manager and Advanced Authentication appliance are located on different servers, you must configure web server to enable performing the following actions:

- Port forwarding to Advanced Authentication appliance for the FIDO U2F method enrollment
- Port forwarding to NetIQ Access Manager for further authentication using FIDO U2F tokens

Perform the following actions to configure a web server to use the FIDO U2F authentication.

### Installing Nginx Web Server

You must install the Nginx web server for URL forwarding.

To install Nginx, add the following two lines to the `/etc/apt/sources.list` file:

```bash
deb http://packages.dotdeb.org jessie all
deb-src http://packages.dotdeb.org jessie all
```

### Preparing SSL Certificate

Run the following commands:
mkdir -p /etc/nginx/ssl
openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/nginx/ssl/proxy.key -out /etc/nginx/ssl/proxy.crt

Preparing Nginx Proxy Configuration

Add the following to the /etc/nginx/sites-available/proxy file:

```
server {
    listen 443 ssl;
    error_log /var/log/nginx/proxy.error.log info;
    server_name nam.company.local;
    ssl_certificate /etc/nginx/ssl/proxy.crt;
    ssl_certificate_key /etc/nginx/ssl/proxy.key;
    location ~ ^/account {
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-Server $host;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header Host $host;
        proxy_pass https://<appliance_IP>$uri?$args;
    }
    location ~ ^/static {
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-Server $host;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header Host $host;
        proxy_pass https://<appliance_IP>$uri?$args;
    }
    location ~ ^/admin {
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-Server $host;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header Host $host;
        proxy_pass https://<appliance_IP>$uri?$args;
    }
    location / {
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-Server $host;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header Host $host;
        proxy_read_timeout 300;
        proxy_pass https://<NAM_IP>;
    }
}
```

Create a link and restart the nginx service running the following commands:

```
ln -s /etc/nginx/sites-available/proxy /etc/nginx/sites-enabled/proxy
service nginx reload
```

Adding DNS Entries

Ensure that the NetIQ Access Manager name server corresponds to the IP address of web server.
Enrolling U2F FIDO

To enroll U2F, open the link `https://<NAM_FQDN>/account`. The Self-Service portal of Advanced Authentication server appliance is displayed.

Enroll the U2F method in the Self-Service portal. For information about enrolling, see “Enrolling the Authentication Methods”.

Voice

In the Voice authentication method, a user receives a call with a PIN request, after which the user must specify the PIN on his or her phone.

The following workflow describes the Voice authentication method in Advanced Authentication:

1. A user tries to authenticate with the Voice method.
2. The user receives a call on the phone with a PIN request.
3. User must specify the PIN that has been enrolled in the Self-Service portal during the enrollment.
4. After the user specifies the PIN followed by a hash (#) symbol, user is authenticated with the Voice method.

**IMPORTANT:** Phone number with extensions are supported for this method.

Special characters “,” and “x” are used to indicate wait time and can be used as separators between phone number and extension.

For example, if +123456789 is the phone number and 123 is the extension, then it can be specified as +123456789,,,,123.

In the above example, “,” is specified 4 times and this multiplied by 0.5 (default value in Twilio) indicates the wait time, which is 2 (4*0.5) seconds. First, call is sent to the number 123456789 and after a wait period of 2 seconds, the extension 123 is dialed.

To configure the Voice method, specify the following details:

- **Minimum PIN length:** The length of the PIN must be at least three characters long.

- **Maximum PIN age:** The validity period of a PIN. The default value is 42 days. If you set the age to 0, the PIN will not expire.

- **User cell phone attribute:** The cell phone number of a user that is used to call the user for voice authentication. You can use custom attributes such as `mobile`, `homePhone`, `ipPhone`, and other attributes of a repository. You must define the attribute in “User Cell Phone Attributes” of the Repositories section.

**NOTE:** If you do not configure the attribute in the method settings, then the first attribute defined in the “User Cell Phone Attributes” section of Repository configuration is used when the user tries to authenticate. For example, if you define `mobile` as the first attribute in User cell phone attribute and do not configure the attribute in method settings of Voice, then while authenticating, the first attribute, which is the `mobile` attribute, is used for the Voice method authentication.

- **Allow overriding phone number:** Option that allows to prevent users from providing a phone number that is not registered in the LDAP repository. The option is set to **ON** by default. Set to **OFF** to prevent users to specify a different phone number during the enrollment.
- **Allow user enrollment without a phone**: Option to configure settings for the user to enroll the Voice authenticator without a phone number in the repository.
  
  Set this option to **OFF** to ensure that a user does not enroll the Voice authenticator without a phone. The user gets an error message that you can specify in **Error message**.
  
  Set this option to **ON** to allow the user to enroll the Voice authenticator without a phone.

**IMPORTANT**: Advanced Authentication does not notify a user about the expiry of a PIN.

### Voice OTP

In the **Voice OTP** authentication method, a user receives an OTP over a call. The user must specify this OTP on the device where the authentication is happening. The OTP must be used within a specific time frame. Voice OTP is recommended to use with other methods, such as Password or LDAP Password.

To configure the Voice OTP method, specify the following details:

- **OTP period**: The time period for which the Voice OTP is valid. Default time is 120 seconds. The maximum value for the Voice OTP period is 360 seconds.
- **OTP format**: The length of the Voice OTP token. Default length is 4.
- **Body**: The text or number in the Voice OTP that is sent to the user. Here, you can specify the `{otp}` variable, which is the actual one-time password. To repeat the one-time password during the call you can specify: Use the OTP for authentication: `{otp}`. OTP: `{otp}`.
- **User cell phone attribute**: Cell phone number of a user that is used to send the OTP through a call. You can use custom attributes such as `mobile`, `homePhone`, `ipPhone`, and other attributes of a repository. You must define the attribute in "**User Cell Phone Attributes**" of the **Repositories** section.

**NOTE**: If you do not configure the attribute in the method settings, then the first attribute defined in the "**User Cell Phone Attributes**" section of Repository configuration is used when the user tries to authenticate. For example, if you define `mobile` as the first attribute in **User cell phone attribute** and do not configure the attribute in method settings of **Voice OTP**, then while authenticating, the first attribute, which is the `mobile` attribute, is used for the **Voice OTP** method authentication.

- **Allow overriding phone number**: Option that allows to prevent users from providing a phone number that is not registered in the LDAP repository. The option is set to **ON** by default. Set to **OFF** to prevent users to specify a different phone number during the enrollment.
- **Allow user enrollment without a phone**: Option to configure settings for the user to enroll the Voice OTP authenticator without a phone number in the repository.
  
  Set this option to **OFF** to ensure that a user does not enroll the Voice OTP authenticator without a phone. The user gets an error message that you can specify in **Error message**.
  
  Set this option to **ON** to allow the user to enroll the Voice OTP authenticator without a phone.
Web Authentication Method

Advanced Authentication facilitates you to authenticate with different Identity Providers such as OAuth 2.0, OpenID Connect, and SAML 2.0 with the Web Authentication method. The Web Authentication method uses browser and http based authentication protocols and can be used in web environment or hybrid applications.

Before you configure the Web Authentication method, ensure that you that provisions Advanced Authentication to the users.

**NOTE:** Ensure that you use a valid certificate for the Advanced Authentication server. Users may face enrollment issues on the Internet Explorer and Microsoft Edge browsers, if the certificates are not valid.

To configure the Web Authentication method for Advanced Authentication, perform the following steps:

1. Click **Methods > Web Authentication**.
2. Click **Add** in **Identity providers**.
3. Select the **Authentication type**.
4. Click the arrow icon.

You can configure the Web Authentication method to use the following Identity Providers:

- SAML
- OpenID Connect
- OAuth 2.0

**SAML for Advanced Authentication**

To add the SAML Identity Provider, perform the following steps:

1. Specify the identity provider name in **Identity Provider**.
2. Select the **Available presets for Name ID Format**.
   - The **Name ID Format** is automatically populated.
   - or
   - Specify manually in **Name ID Format**.
3. Click **Browse** to upload the **Identity Provider Metadata file**.

**WARNING:** Ensure that you choose the Identity Provider Metadata file that is exported from a used Identity Provider. Do not use the metadata file exported from the **Administrative Portal > Policies > Web Authentication**.

4. Click the save icon.
5. In the **Upload SAML Service Provider signature certificate** section, you must upload a certificate file in the **PEM** format with a private key. This certificate is used by the Web Authentication method to sign a SAML **AuthnRequest** token.
   - If the private key is protected by a password, specify the password in **Private key password**.
6. Click **Save**.
An Example Configuration with ADFS

Perform the following steps to add ADFS as an Identity Provider for the Web Authentication method.

1. Specify `myexample-adfs` as the IdP provider name.


   The selected Name ID Format will be extracted from the SAML AuthnResponse token and saved as an authentication data (unique data which will be associated with the user).

3. Click Browse to upload the IdP Metadata file from the ADFS server.

4. Click the save icon.

5. In the Upload SAML Service Provider signature certificate section, upload a certificate file in the PEM format with a private key.

   If the private key is protected by a password, specify the password in Private key password.

6. Click Save.

Configuring the ADFS Identity Provider

1. Save the Service Provider metadata from Advanced Authentication to a file. Use the URL mentioned below to obtain the Service Provider metadata:

   https://AAF_SERVER/webauth/TENANT/metadata

   **NOTE:** The default TENANT is TOP. Use TOP as TENANT if you are not using multi-tenancy.

A sample Service Provider metadata is mentioned below:

```
<md:EntityDescriptor xmlns:md="urn:oasis:names:tc:SAML:2.0:metadata"
ID="_7a8608ad1cfbc149" entityID="https://www.d18r14.tk/webauth">
<md:SPSSODescriptor
protocolSupportEnumeration="urn:oasis:names:tc:SAML:2.0:protocol
<md:KeyDescriptor>
<ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
<ds:KeyName>https://www.d18r14.tk/webauth</ds:KeyName>
<ds:X509Data>
<ds:X509Certificate>
MIIEOzCCAYgAwIBAgIJAJcsrIQZzcT0MA0GCSqGSIb3DQEBCwUAMIGYMQswCQYD
VQQGEwJDTEEDMBQGA1UECxMVU29iZSBXaW1nbywgQ2FsbGVzIENvbXBvcnQgQ0Ex
MREwDQYJKoZIhvcNAQEFBQcMCgYIKoZIhvcNAQELBzIBIjANBgkqhkiG9w0BAQUw
AQMGA1UEBhMCVVRIZDIwMDAweG1sMQswCQYDVQQGEwJDSDEcMBoGA1UECAwTR3Jl
YXRlciBadXJpY2ggQXJlYTEPMA0GA1UEBwwGQUlvdXJlckFVc2V0czAvBgNVBAMM
Dk1pY3JvYXMxFzAVBgNVBAMMDk1pY3JvYXMwMwYDVQQDDBVSRU5VTUdFbHRlcmwF
MEQ5MRkGA1UEBhMCVVRIZDIwMDAweG1sMQswCQYDVQQGEwJDSDEcMBoGA1UECAwT
R3JlYXRlciBadXJpY2ggQXJlYTEPMA0GA1UEBwwGQUlvdXJlckFVc2V0czAvBgNVB
AMMDk1pY3JvYXMwMwYDVQQDDBVSRU5VTUdFbHRlcmwFMEQ5MRkGA1UEBhMCVVRIZ
DiwMDAweG1sMQswCQYDVQQGEwJDSDEcMBoGA1UECAwTR3JlYXRlciBadXJpY2ggQX
JlYTEPMA0GA1UEBwwGQUlvdXJlckFVc2V0czAvBgNVBAMMDk1pY3JvYXMwMwYDVQQ
HDBVSRU5VTUdFbHRlcmwFMEQ5MRkGA1UEBhMCVVRIZDiwMDAweG1sMQswCQYDVQQE
GDVSRU5VTUdFbHRlcmwFMEQ5MRkGA1UEBhMCVVRIZDiwMDAweG1sMQswCQYDVQQG
EE1qLmRvYXNzZXR0b24wIwYDVQQDDBVSRU5VTUdFbHRlcmwFMEQ5MRkGA1UEBhMC
VVRIZDiwMDAweG1sMQswCQYDVQQGEwJDSDEcMBoGA1UECAwTR3JlYXRlciBadXJp
Y2ggQXJlYTEPMA0GA1UEBwwGQUlvdXJlckFVc2V0czAvBgNVBAMMDk1pY3JvYXMw
MwYDVQQDDBVSRU5VTUdFbHRlcmwFMEQ5MRkGA1UEBhMCVVRIZDiwMDAweG1sMQsw
CQYDVQQGEwJDSDEcMBoGA1UECAwTR3JlYXRlciBadXJpY2ggQXJlYTEPMA0GA1UEB
wwGQUlvdXJlckFVc2V0czAvBgNVBAMMDk1pY3JvYXMwMwYDVQQDDBVSRU5VTUdFbHR
lcmwFMEQ5MRkGA1UEBhMCVVRIZDiwMDAweG1sMQswCQYDVQQGEwJDSDEcMBoGA1UE
CAwTR3JlYXRlciBadXJpY2ggQXJlYTEPMA0GA1UEBwwGQUlvdXJlckFVc2V0czAvB
```
In the ADFS Management console, click Relying Party Trusts > Add relying party trust.

In the Add Relying Party Trust wizard, click Start.

Select Import data about the relying party from a file.

Click Browse to upload the Advanced Authentication’s metadata file that you created in Step 1.

Click Next.

Specify the Display name.

Click Next.

Ensure that Open the Edit Claim Rules dialog for this relying party trust when the wizard closes is selected.

Click Close.

The Edit Claim Rules wizard is displayed.

Click Add Rule.

Select Transform an Incoming Claim from Claim rule template.

Click Next.

Specify the Claim rule name.

Set Incoming claim type to Windows account name.

Set Outgoing claim type to Name ID and Outgoing name ID format to Windows Qualified Domain Name.

Ensure that Pass through all claim values is selected.

Click Finish.

Click OK.

In the ADFS Management console, click Relying Party Trusts and select the relying party trust you added.

Right-click on the relying party trust and select Properties from the menu.

In Properties, click the Encryption tab and remove the certificate by clicking Remove.

Click OK.

NOTE: Web authentication method does not support the encrypted tokens.
OpenID Connect for Advanced Authentication

To add the Open ID Connect Identity Provider, perform the following steps:

1. Specify the name of the provider in **Provider name**.
2. Select the **Available presets**.
   - The **Issuer**, **Scope**, and **Key field** are automatically populated.
3. Specify the **Client ID** and **Client secret**.
   - The **Client ID** and **Client secret** can be obtained by registering with the respective Identity Provider that you select, for more information see “Integrating Third Party Applications with Advanced Authentication Using OpenID Connect” on page 79.

**NOTE:** Set the Callback URL at the respective Identity Provider. For example, `https://<aahostname>/webauth/callback`.

4. Turn **Send Client secret as an URL parameter** to **ON** to send the Client secret as a URL. By default, the option is set to **OFF**.
5. Click the save icon.
6. Click **Save** to save the method configuration.

Integrating Third Party Applications with Advanced Authentication Using OpenID Connect

The following sample configurations explains how to configure third party applications with Advanced Authentication using OpenID Connect.

- “Integrating Advanced Authentication with Facebook” on page 79
- “Integrating Advanced Authentication with Google” on page 80
- “Integrating Advanced Authentication with Yahoo” on page 81
- “Integrating Advanced Authentication with Microsoft Azure” on page 81

**Integrating Advanced Authentication with Facebook**

Perform the following steps to integrate Advanced Authentication with Facebook using OpenID Connect:

1. Login to facebook for developers (https://developer.facebook.com).
2. Click **My Apps**.
3. In the left pane, click **Settings > Basic**.
4. Make a note of **App ID** and **App Secret**. These are the Client ID and Client Secret for Advanced Authentication.
5. In **Display Name**, specify **Advanced Authentication**. This is the name for this OpenID Connect configuration.
6. In **App Domains**, specify the domain name of the Advanced Authentication Server. For example `aafapp.demo.live`.
8. Scroll through the page until you find the **Website** section. If you cannot find the **Website** section, click **Add Platform > Website**.
In the **Website** section, specify the web address of the Advanced Authentication Server. For example aafapp.demo.live.

Click **Save Changes**.

In the left pane, click **Settings > Advanced**.

Scroll through the page until you find the **Domain Manager** tab.

Click **Add a Domain**.

In the Add a Domain window, specify the URL of the Advanced Authentication Server in **Site URL**. For example aafapp.demo.live.

Click **Apply**.

Click **Save Changes**.

In the left pane, click **App Review**.

Make your application public by clicking the toggle switch in the **Make Advanced Authentication public?** section.

In the left pane, below the **Products** tab, click Settings.


Click **Save Changes**.

Specify the Client ID and Client Secret generated in Step 4 on page 79 in the **Client ID** and **Client Secret** fields of Advanced Authentication Administrative Portal.

**Integrating Advanced Authentication with Google**

Perform the following steps to integrate Advanced Authentication with Google using OpenID connect:

1. Login to Google APIs (https://console.developers.google.com/apis/credentials).
2. Click **Credentials > Create**.
3. Specify a **Project Name** and a **Location**.
4. Click **Create**.
5. Click Create credentials > **OAuth client ID**.
6. Click **Configure a consent screen**.
7. Specify a name in the **Application name** field. For example Advanced Authentication.
8. In **Authorised domains**, specify the domain name of the Advanced Authentication Server. For example aafapp.demo.live.
11. In **Application type**, select **Web application**.
12. In **Application Terms of Service link**, specify the web address of the Advanced Authentication Server. For example https://aafapp.demo.live.
13. In **Name**, specify a name for the OpenID Connect configuration.
14. In **Authorized JavaScript origins**, specify the Advanced Authentication server address. Ensure that you specify the complete server address including https. For example https://aafapp.demo.live.
In Authorized redirect URIs, specify https://<Advanced Authentication Server>/webauth/callback. Ensure that you specify the valid Advanced Authentication server name inside <>.

Click Save.

Make a note of the client ID and client secret specified in the OAuth client window. Click OK.

Specify the Client ID and Client Secret generated in Step 17 on page 81 in the Client ID and Client Secret fields of Advanced Authentication Administrative Portal.

**Integrating Advanced Authentication with Yahoo**

Perform the following steps to integrate Advanced Authentication with Yahoo using OpenID connect:

1. Login to Yahoo Developer Network (https://developer.yahoo.com/apps/).
2. Click Create an app.
3. In Application Name, specify a name for the OpenID Connect configuration.
4. In Application Type, select Web Application.
5. In Callback Domain, specify the domain name of the Advanced Authentication Server. For example aafapp.demo.live.
6. Click Create.
7. Make a note of the client ID and client secret. Click Update.

**Integrating Advanced Authentication with Microsoft Azure**

Perform the following steps to integrate Advanced Authentication with Microsoft Azure using OpenID connect:

1. Login to Microsoft Azure (https://portal.azure.com/).
2. In the left pane, click Azure Active Directory.
3. In the Manage section, click App registrations.
4. Click New application registration.
5. In Name, specify a name for the OpenID Connect configuration.
6. In Application Type, select Web app / API.
8. Click Create.
9. Make a note of Application ID. It is the Client ID for Advanced Authentication.
10. Click Settings > Keys.
11. In the Passwords section, specify key description and key duration.
12. Click Save.
13. Make a note of the text generated in the VALUE field. It is the Client Secret for Advanced Authentication.
14. In the left pane, click Azure Active Directory.
15. Click Properties.
17 Specify the text generated in Step 16 on page 81 in the Issuer field of Advanced Authentication Administrative Portal.


OAuth 2.0 for Advanced Authentication

To add the OAuth 2.0 Identity Provider, perform the following steps:

1 Specify the name of the provider in Provider name.

2 Select the Available presets.
   The Authorization endpoint, Token endpoint, Attributes endpoint, Scope, and Key field are automatically populated.

3 Specify the Client ID and Client secret.
   The Client ID and Client secret can be obtained by registering with the respective Identity Provider that you select.

   **NOTE:** Set the Callback URL at the respective Identity Provider. For example, https://<aahostname>/webauth/callback.

4 Turn Send Client secret as an URL parameter to ON to send the Client secret as a URL. By default, the option is set to OFF.

5 Select the format of the access token from Access token is returned in body encoded as.

6 Set Send access token in "Authorization: Bearer" header to ON to send the access token as a header. By default, the option is set to OFF.

7 Click the save icon.

8 Click Save to save the method configuration.

Windows Hello

Windows Hello authentication allows the users to use the Windows Hello Fingerprint and Facial Recognition authentication to log in to Windows 10. Advanced Authentication supports the Windows Hello fingerprint and facial recognition authentication.

To configure Windows Hello method in Advanced Authentication, perform the following steps:

1 Click Methods > Windows Hello.

2 (Optional) Set Allow to specify Username (for AD Users only) to ON if you want the Active Directory users to specify their account name while enrolling. By default, the option is disabled.
   This is applicable for Active Directory users only. This option does not affect local and other repository users and they must specify their account name while enrolling.

3 Click Save.
Creating a Chain

A chain is a combination of authentication methods. A user must pass all methods in the chain to be successfully authenticated. For example, if you create a chain with LDAP Password and SMS OTP, a user must first specify the LDAP Password. If the LDAP password is correct, the system sends an SMS with a One-Time-Password (OTP) to the user’s mobile. The user must specify the correct OTP to be authenticated.

Advanced Authentication contains the following chains that are created by default:

1. **LDAP Password Only**: Any user from a repository can use this chain to get authenticated with the LDAP Password (single-factor) method.
2. **Password Only**: Any user who has a Password method enrolled can use this chain to get authenticated with the Password (single-factor) method.

You can create any number of chains with multiple authentication methods. To achieve better security, you can include multiple methods in a chain.

Authentication comprises of the following three factors:

- **Something that you know** such as password, PIN, and security questions.
- **Something that you have** such as smartcard, token, and mobile phone.
- **Something that you are** such as biometrics (fingerprint or iris).

You can achieve multi-factor or strong authentication by using any two factors out of this list. For example, multi-factor authentication can include a combination of password and a token or a smartcard and a fingerprint.

After you create a chain, you can use the chain on specific user groups in your repository. The chain is then mapped to an event.

To create a new chain or edit an existing chain, perform the following steps:

1. Click **Chains**.
2. Click **Add** to create a chain. You can also click the edit icon against the chain that you want to edit.
3. Specify a name of the chain in **Name**.
4. Specify a **Short name**. The short name is used by a user to move to a chain. For example, if you name a chain containing the LDAP Password and SMS methods as SMS, then a user can specify <username> sms and the user is forced to use SMS as the chain. This is helpful in scenarios when the primary chain is not available.

**NOTE**: This is applicable only for the RADIUS Server event.

5. Set **Is enabled** to **ON** to enable the chain.
6. Select the methods that you want to add to the chain from the **Methods** section. You can prioritize the methods in the list. For example, if you create a chain with LDAP Password and HOTP methods, then the user will be prompted for the LDAP Password method first and then the OTP.
7. Specify the groups that will use the authentication chain in **Roles and Groups**.

You can specify the following roles and groups based on your requirement:

- **ALL USERS**: To use all the users and groups of all the added repositories.
- `<REPO|Group>`: To use a specific group from the repository. For example to specify users of an IT staff group, specify FOCUS\IT staff.
- `<REPO Users>`: To use all the users of a specific repository. For example to use all users in the repository FOCUS, specify FOCUS Users.

**IMPORTANT**: It is recommended not to use those groups from which you will not be able to exclude users because you will not be able to free up a user’s license. For example, you use a Repo Users group or ALL USERS group. If an employee from these groups leaves the company and you do not delete the user’s domain account but just disable it, the license will not be freed.

8 Expand Advanced Settings by clicking +.
9 Set Apply if used by endpoint owner to ON if an Endpoint owner must use the chain.

**NOTE**: The Endpoint owner feature is supported for Windows Client, Mac OS Client, and Linux PAM Client only.

10 Specify the MFA tags. When a user logs in to Windows on a workstation with Advanced Authentication Windows Client installed, the user’s account is moved to the group specified in the MFA tags.

**NOTE**: This functionality is available when you set the Enable filter to ON in the Logon Filter for AD policy and have configured the Logon Filter.

For example if you specify a Card users group from Active Directory in MFA tags, then the user will be moved from the legacy group (specified in the Advanced Settings of Active Directory repository) to the Card users group.

**NOTE**: If the user credentials are saved with Remember my credentials, the MFA tag does not work while connecting to the Remote Desktop.

11 Set Required chain to Nothing, if this is a required (high-security) chain. To configure a linked chain within a specific time period after successful authentication with a required chain, choose an appropriate required chain. You also need to specify a Grace period (mins). Within this time period, the linked chain can be used instead of the required chain. The maximum value for grace period is 44640 minutes (31 days).

**NOTE**: You must assign both a required and a linked chain to an Event. The linked chain must be of higher order than the corresponding required chain. The option is available when the Linked Chains policy is set to ON.

For example, LDAP Password+Card is a required chain and Card is a linked chain. The users must use LDAP Password+Card chain once in every 8 hours and within this period, they can provide only card without the LDAP Password to authenticate.

12 (Conditional) In Custom names, you can specify the chain name in a specific language. To do this click + to expand the settings and specify the chain name.

13 Click Save.

**IMPORTANT**: If you have configured more than one chain using one method (for example, LDAP Password, LDAP Password+Smartphone) and assigned it to the same group of users and the same event, then the top chain is always used if the user has enrolled all the methods in the chain. An exception is the use of a high-security chain and its appropriate simple chain, where the simple chain must be higher than its high-security chain.
Configuring Events

Advanced Authentication provides authentication events for the supported applications or devices. You can configure an event to leverage the Advanced Authentication functionalities for the respective application or device. The application or device triggers the respective authentication event when a user tries to access it.

You can create customized events for the following:

- Third-party integrations.
- To use Windows Client, Linux PAM Client or Mac OS X Client on both the domain joined and non-domain workstations and it requires to have a separate event to use the non-domain mode.
- Integrations using SAML 2.0 and OAuth 2.0.
- To create more than one RADIUS Server event.

This section contains the following:

- “Configuring an Existing Event” on page 85
- “Creating a Customized Event” on page 90

Configuring an Existing Event

1. Click Events.
2. Click the edit icon against the event that you want to edit.
3. Ensure that Is enabled is set to ON if you want to use the event.
4. Select the Event type.
   For most of the predefined events, you cannot change the Event type. For events such as Windows logon, Linux logon, and Mac OS logon, you can change the Event type from OS Logon (domain) to OS Logon (local) if the workstations are not joined to the domain.
   - Select OS Logon (domain) to allow only the domain joined users to login to the event.
   - Select OS Logon (local) to allow any Advanced Authentication user from any repository to access the event. However, users must map themselves to a local user account during their first login by providing the credentials.
5. Enable the reCAPTCHA option to ON if you want the Google reCAPTCHA option to be displayed in the login page for the particular event.
   The reCAPTCHA option is displayed only when you enable the Google reCAPTCHA Options policy.

   **NOTE:** The reCAPTCHA option is supported only for the Admin UI event, Authenticators Management event, Helpdesk event, Helpdesk user event, Report logon event, Tokens Management event, and the Search Card event.

6. Select the Authenticator category. The Authenticator category option is displayed only if you have added categories in the “Event Categories” policy.
7. Select the chains that you want to assign to the current event.
   In an event, you can configure a prioritized list of chains that can be used to get access to that specific event.
8 If you want to restrict access of some endpoints to the event, add all the endpoints that must have access to the **Endpoint whitelist**. The remaining endpoints are blacklisted automatically. If you leave the **Endpoints whitelist** blank, all the endpoints will be considered for authentication.

9 Set **Geo-fencing** to **ON** to enable geo-fencing. Move the permitted zones from **Available** to **Used**. For more information about configuring geo-fencing, see the **Smartphone** method.

**IMPORTANT:** You must enable the **Geo Fencing Options** policy to use the geo fencing functionality.

10 Select **Allow Kerberos SSO** if you want to enable single sign-on (SSO) to the Advanced Authentication portals. Kerberos SSO is supported for AdminUI, Authenticators Management, Helpdesk, and Report logon events.

11 Set **Bypass user lockout in repository** to **ON**, if you want to allow users who are locked on repository to authenticate on the Advanced Authentication. By default, **Bypass user lockout in repository** is set to **OFF** and users who are locked on repository are not allowed to authenticate.

12 Select the **Allow to logon to this event by shared template** option to allow users to login using shared authenticators. By default this option is disabled for the Authenticators Management, Helpdesk, Helpdesk User, AdminUI, Search Card, Token Management, and Report Logon events and enabled for all the other events.

13 Click **Save**.

14 If you want to revert the changes to the default configuration, click **Initialize default chains**.

**NOTE:** If you have configured more than one chain using one method (for example, LDAP Password, LDAP Password+Smartphone) and assigned it to the same group of users and to the same event, the top chain is always used if the user has enrolled all the methods in the chain. An exception is the use of a high-security chain and its appropriate simple chain, where the simple chain must be higher than its high-security chain.

**TIP:** It is recommended to have a single chain with the **Emergency Password** method at the top of the chains list in the Authenticators Management event and other events, which are used by users. The chain will be ignored if the user does not have the **Emergency Password** enrolled. The user can use the Emergency Password immediately after the helpdesk administrator enrolls the user with the Emergency Password authenticator.

**NOTE:** Configurations that have been set by a top administrator for a particular event are grayed out. The configurations are not displayed, if the configurations are hidden by the top administrator.

By default, Advanced Authentication contains the following events.

- ADFS Event
- AdminUI Event
- Authentication Agent Event
- Authenticators Management Event
- Helpdesk Event
- Helpdesk User Event
- Linux Logon Event
- Mac OS Logon Event
- Mainframe Logon Event
• NAM Event
• NCA Event
• RADIUS Server Event
• Report Logon Event
• Tokens Management Event
• Windows Logon Event

**ADFS Event**

This event is used to integrate Advanced Authentication with ADFS using the previous ADFS plug-in for Advanced Authentication 5.x.

For 6.0, you can use the new ADFS MFA plug-in. For more information see the *Configuring the Advanced Authentication Server for ADFS Plug-in* guide.

**AdminUI Event**

Use this event to access the Administration portal. You can configure the chains that can be used to get access to the `/admin` URL.

**IMPORTANT:** You must be careful when changing the default chains that are assigned to this event. You may block the access to the Administration portal.

**NOTE:** You can promote users or group of users from a repository to the **FULL ADMINS** role in **Repositories > Local**. After this, you must assign chains in which the methods are enrolled for users with the AdminUI event (at a minimum with an LDAP Password).

**WARNING:** If you have enabled the Google reCAPTCHA policy for the Admin UI event, you must consider the following guidelines. Otherwise, a deadlock scenario can happen and you will not be able to access the Administration portal without the cluster re-installation:

• If the site key or secret key gets deleted at the Google server, you will not be able to get the same site key or secret key. The site key and secret key used on the Administration portal are no more valid and there is no way to bypass the reCaptcha on the Administration portal.

• If you have registered the reCAPTCHA for one domain name and you change the domain name or migrate the Advanced Authentication server to another domain name, the site key or secret key used on the Administration portal are no more valid.

**Authentication Agent Event**

Configure the settings of this event to enable a login to the Authentication Agent on Windows Client.
**Authenticators Management Event**

Use this event to access the Self-Service portal. In the Self-Service portal, users can enroll to any of the methods that are configured for any chain and they are a member of the group assigned to the chain.

Add an **LDAP Password** chain as the last chain in the list of chains to ensure secure access to the portal for users who have methods enrolled.

---

**IMPORTANT:** If the Administration portal uses a repository that does not have any user, you must enable a chain with **Password** only (Authenticators Management - Password) for this event. This action enables you accessing the Self-Service portal or changing the password in the Self-Service portal.

You can also perform basic authentication with Advanced Authentication. To achieve basic authentication, set the **Allow basic authentication** option to **ON** in the Event Edit screen for Authenticators Management.

---

**NOTE:** The basic authentication is supported only for the **Authentication Management** event and for the **Password**, **LDAP Password**, and **HOTP** methods.

---

You must specify `/basic` with the URL to login to the enrollment page. The Login page appears and the format of the Username you must provide is: `username:PASSWORD|LDAP_PASSWORD|HOTP:1`. For example: `admin:PASSWORD:1`.

When you log in to the Self Service portal, by default the chain with the highest priority is displayed. To display the other chains with the enrolled methods, set **Show chain selection** to **ON**.

---

**NOTE:** If you enable to show the chain selection, but a chain is not displayed in the list of available chains in the Self-Service portal, ensure that all the methods of the chain are enrolled by the user.

For more information, see “Managing Authenticators” in the Advanced Authentication- User guide.

**Helpdesk Event**

Configure the settings of this event to enable the Helpdesk administrator to access the Helpdesk portal. One of the roles of a Helpdesk administrator is to set an emergency password for users. An emergency password is a temporary password for users when they lose their smart card or smart phone. Some companies restrict self-enrollment and have the Helpdesk administrator who does the enrollment after hiring. You can promote the repository administrators or users as Helpdesk administrators in the Repositories > LOCAL > Edit > Global Roles > ENROLL ADMINS section.

You can manage the enrollment and re-enrollment of the authenticators in one of the following ways:

- Restrict the self-enrollment and force users to enroll through the Helpdesk.

  Or

- Restrict only the re-enrollment or deletion of authenticator from the Self-Service portal using the **Disable re-enrollment** option.

For more information, see “Authenticators Management” in the Advanced Authentication- Helpdesk Administrator guide.
Helpdesk User Event

Configure the settings of this event to enable the Helpdesk administrator to authenticate users in the Helpdesk portal. This event is applicable for the User to manage screen that appears on the Helpdesk portal.

You must enable the Ask credentials of management user option in the Helpdesk Options policy before using this event.

Linux Logon Event

Configure the settings of this event to enable login to the Linux Client. If you want to use Linux Client on non-domain joined workstations, change the Event type from OS Logon (domain) to OS Logon (local).

Mac OS Logon Event

Configure the settings of this event to enable login to the Mac OS Client. If you want to use Mac OS Client on non-domain joined workstations, change the Event type from OS Logon (domain) to OS Logon (local).

Mainframe Logon Event

Configure the settings of this event to enable login to the Mainframe system.

NAM Event

Configure the settings of this event to facilitate the integration of Advanced Authentication with NetIQ Access Manager (https://www.netiq.com/products/access-manager/).

NCA Event

Configure the settings of this event to facilitate the integration of Advanced Authentication with NetIQ CloudAccess (https://www.netiq.com/products/cloudaccess/). CloudAccess must be configured to use Advanced Authentication as an authentication card and user stores must be added for the repositories for the integration to work. For more information, see the Advanced Authentication CloudAccess documentation.

RADIUS Server Event

The Advanced Authentication server contains a built-in RADIUS server to authenticate any RADIUS client using one of the chains configured for the event. For more information about configuring the RADIUS Server event, see Chapter 6, “RADIUS Server,” on page 147.

Report Logon Event

Configure the settings of this event to log in to the Advanced Authentication Reporting portal. For more information about the Reporting portal, see Chapter 10, “Reporting,” on page 195.
Search Card Event

Configure the settings of this event to log in to the Advanced Authentication Search Card portal. The Search Card functionality helps you to get the card holder’s contact information by inserting the card in the card reader. For more information about searching a card holder’s information, see Chapter 12, “Searching a Card Holder’s Information,” on page 199.

Tokens Management Event

Configure the settings of this event to log in to the Advanced Authentication Tokens Management portal. The Tokens Management functionality allows you to assign each token to specific user. For more information about assigning a token to user, see Chapter 11, “Managing Tokens,” on page 197.

Windows Logon Event

Configure the settings of this event to log in to the Windows Client.

Creating a Customized Event

You can create customized events for the following.

- Third-party integrations.
- When you must use Windows Client or Linux PAM Client, or Mac OS X Client on both the domain joined and non-domain workstations and you must have a separate event to use the non-domain mode.
- For integrations using SAML 2.0 and OAUTH 2.0.
- To create more than one RADIUS Server event.

You can create the following types of customized events:

- Generic
- OS Logon (domain)
- OAuth2
- SAML2
- RADIUS

Creating a Generic Event

You can create a generic event for Windows Client, Mac OS X Client, and Linux PAM Client workstation when these clients are not joined or bound to a domain.

Perform the following steps to create a generic event:

1. Click Events > Add.
2. Specify a name for the event.
3. Set Is enabled to ON.
5. Select the Authenticator category. The Authenticator category option is displayed only if you have added categories in the “Event Categories” policy.
6. Select the chains that you want to assign to the current event.
7 If you want to restrict access of some endpoints to the event, add all the endpoints that must have access to the Endpoint whitelist. The remaining endpoints are blacklisted automatically. If you leave the Endpoints whitelist blank, all the endpoints will be considered for authentication.

8 Set Geo-fencing to ON to enable geo-fencing. Move the permitted zones from Available to Used. For more information about configuring geo-fencing, see the Smartphone method.

**IMPORTANT:** You must enable the Geo Fencing Options policy to use the geo fencing functionality.

9 Set Bypass user lockout in repository to ON, if you want to allow users who are locked on repository to authenticate on the Advanced Authentication. By default, Bypass user lockout in repository is set to OFF and users who are locked on repository are not allowed to authenticate.

10 Click Save.

**NOTE:** When you create a custom event, you must specify the custom event in the configuration file of the related endpoints. For more information, see the Advanced Authentication- Linux PAM Client, Advanced Authentication - Mac OS X Client, or Advanced Authentication - Windows Client guides related to the specific endpoint.

### Creating an OS Logon (Domain) Event

You can create this event when the third-party application needs to read password of a user after authentication. For example, when Windows Client, Mac OS X Client, or Linux PAM Client workstation is joined or bound to a domain, the third-party application must read the password of the user.

The steps to create an OS Logon (domain) event are similar to the Generic event.

### Creating an OAuth 2.0 Event

You can create this event for third-party integrations with OAuth 2.0.

To create an OAuth 2 event, perform the following steps:

1 Click Events > Add.
2 Specify a name for the event.
3 Set Is enabled to ON.
4 Select OAuth2 in the Event type.
5 Select the Authenticator category. The Authenticator category option is displayed only if you have added categories in the “Event Categories” policy.
6 Select the chains that you want to assign to the current event.
7 Specify the Redirect URIs. The Client ID and Client secret are generated automatically. The Client ID, Client secret, and Redirect URI are consumed by the consumer web application. After successful authentication, the redirect URI web page specified in the event is displayed.
8 In Advanced Settings, perform the following actions:
   - Set the Use for Owner Password Credentials option to ON, if the consumer web application provides authorization in the form of Resource Owner Password Credentials Grant.
   - Set the option to OFF, if the consumer web application provides authorization in the form of Authorization Code Grant or Implicit Grant.
NOTE: If option is set to ON, you can use only the LDAP Password only chain for this event. It is recommended to use separate events for Resource Owner Password Credentials Grant (Use for Owner Password Credentials > ON) and Authorization Code Grant / Implicit Grant (Use for Owner Password Credentials > OFF).

9 Set Bypass user lockout in repository to ON, if you want to allow users who are locked on repository to authenticate on the Advanced Authentication. By default, Bypass user lockout in repository is set to OFF and users who are locked on repository are not allowed to authenticate.

10 Click Save.

After you have created an OAuth 2 event, perform the following steps to access the consumer web application:

1 Specify the Client ID, Client secret, and redirect URIs in the consumer web application.
2 Specify the appliance end point (authorization end point) in the web application. For example, https://<Appliance IP>/osp/a/TOP/auth/oauth2/grant.
3 Authenticate with the required authentication method(s) to access the consumer web application.

NOTE: Authorization is provided in the form of Authorization Code Grant or Implicit Grant or Resource Owner Password Credentials Grant.

Creating a SAML 2.0 Event

You can create this event for third-party integrations with SAML 2.0.

1 Click Events > Add.
2 Specify a name for the event.
3 Set Is enabled to ON.
4 Select SAML 2 in the Event type.
5 Select the Authenticator category. The Authenticator category option is displayed only if you have added categories in the "Event Categories" policy.
6 Select the chains that you want to assign to the current event.
7 In SAML 2.0 settings, perform the following:

NOTE: You must configure the Web Authentication policy for the SAML 2.0 event to work appropriately.

7a You can either insert your Service Provider's SAML 2.0 metadata in SP SAML 2.0 metadata or click Browse and select a Service Provider's SAML 2.0 metadata XML file to upload it.
7b Set the Send E-Mail as NameID (suitable for G-Suite) option to ON for integrating with the G-suite.
7c Set the Send SAMAccount as NameID option to ON to send SAMAccountName in the NameID attribute as a SAML response from the Advanced Authentication server. This option must be enabled for the integration with CyberArk.
WARNING: You can set Send SAMAccount as NameID to ON only when the Send E-Mail as NameID (suitable for G-Suite) option is turned OFF.

7d Set Bypass user lockout in repository to ON, if you want to allow users who are locked on repository to authenticate on the Advanced Authentication. By default, Bypass user lockout in repository is set to OFF and users who are locked on repository are not allowed to authenticate.

8 Click Save.

Creating a RADIUS Event

When you want to add multiple RADIUS clients, you can add them to the predefined RADIUS Server event. But all the RADIUS clients will use the same authentication chain(s). If you want to configure specific authentication chain(s) for different RADIUS clients, then you must create a custom RADIUS event. To add a custom RADIUS event, perform the following steps:

1 Click Events > Add.
2 Specify a name for the event.
3 Ensure that Is enabled is set to ON.
4 Select RADIUS from Event Type.
5 Select the chains that you want to assign to the event.
6 Select RADIUS from Endpoint whitelist.
7 Click Add to add and assign a RADIUS Client to the event:
   7a Specify the IP address of the RADIUS Client in IP Address.
   7b Specify the RADIUS Client name in Name.
   7c Specify the RADIUS Client secret and confirm the secret.
   7d Ensure that the RADIUS Client is set to ON.

7e Click to save the RADIUS Client.

7f Add more RADIUS Clients if required.

8 Specify NAS ID for the RADIUS event and use the same NAS ID on the configured RADIUS clients to associate them with the custom RADIUS event.

   NAS ID is a unique identifier to map RADIUS clients to the custom RADIUS event.

   NOTE: While configuring the predefined RADIUS Server event, NAS ID is optional. But while adding a custom RADIUS event, it is required to specify NAS ID that is used to map RADIUS clients with the custom RADIUS event.

9 Set Bypass user lockout in repository to ON, if you want to allow users who are locked on repository to authenticate on the Advanced Authentication. By default, Bypass user lockout in repository is set to OFF and users who are locked on repository are not allowed to authenticate.

10 Click Save.
Managing Endpoints

Endpoints are devices where the Advanced Authentication server authenticates. An endpoint can be a Windows workstation for Windows Client endpoint, or Advanced Authentication Access Manager appliance for the NAM endpoint and so on.

The endpoints are automatically added when you install a plug-in such as NAM or install Windows Client. The RADIUS endpoint, an OSP endpoint that is used for WebAuth authentication, and Endpoint41 and Endpoint42 are the predefined endpoints.

NOTE: Endpoint41 and Endpoint42 are created for the integration with legacy NAM and NCA plug-ins, which are used in NAM 4.2 and earlier versions with Advanced Authentication 5.1.

The NAM and NCA plug-ins work with the hard coded endpoint ID and secret. In Advanced Authentication 5.2 and later, you must register the endpoints. This breaks the backward compatibility with old plug-ins. These two legacy endpoints allow to keep the old plug-ins working.

To configure an endpoint for Advanced Authentication, perform the following steps:

1. In the **Endpoints** section, click **Edit** against the endpoint you want to edit.
2. You can rename the endpoint, change its description or endpoint type.
3. Set **Is enabled** to ON to enable the endpoint.
4. Set **Is trusted** to ON if the endpoint is trusted. In some integrations such as Migration Tool, Password Filter, NAM, and NCA you must enable the **Is trusted** option for their endpoints.
5. Specify an **Endpoint Owner** if you have configured a specific chain to be used by the Endpoint owner only. This is a user account that must be able to use a different chain than the other users for authentication.
   
   The Endpoint Owner feature is supported for Windows Client, Mac OS Client, and Linux PAM Client only.

**NOTE**: Additional information such as Operating System, Software version, Last session time and Device information are displayed. Also in Advanced properties, RAM information is displayed.

Advanced Authentication Windows Client 5.6 or newer, Advanced Authentication Linux PAM Client 6.0 or newer, Advanced Authentication Mac OS X Client 6.0 or newer must be installed on the endpoint.

6. Click **Save**.

You can create an endpoint manually. This endpoint can be used for the third-party applications that do not create endpoints.

To create an endpoint manually, perform the following steps:

1. In the **Endpoints** section, click **Add**.
2. On the **Add endpoint** page, specify a **Name** of the endpoint and its **Description**.
3. Set the **Type** to **Other**.
4. Set **Is enabled** to ON.
5. Set **Is trusted** to ON if the endpoint is trusted.
6. Leave **Endpoint Owner** blank.
7. Click **Save**. The **New Endpoint secret** window is displayed.
8 Take down the values specified in Endpoint ID and Endpoint Secret and place them in a secure place in your application.

NOTE: You will not be able to get the Endpoint ID and Endpoint Secret later on the appliance.

9 Click OK.

NOTE: Tenancy settings are not supported for Endpoints.

IMPORTANT: You must ensure not to remove an endpoint that has at least one component running on it such as Windows Client, Logon Filter, RD Gateway plug-in, or ADFS plug-in. Endpoint is removed automatically when you uninstall Windows Client. However you must remove the endpoint manually when you uninstall Logon Filter, RD Gateway plug-in or ADFS plug-in.

If you remove an endpoint accidentally, ensure to remove the records with prefix endpoint* from the %ProgramData%\NetIQ\Windows Client\config.properties file and re-start the machine. This recreates the endpoint.

Configuring Policies

Policies contain configuration settings for the Advanced Authentication methods, events, and so on. For example, to use the Email OTP method, you must configure the server and port settings in the Mail sender policy and to use the Multitenancy mode, you must enable the Multitenancy options policy.

Advanced Authentication provides the following policies:

- Authenticator management options
- Cache options
- Custom Messages
- Custom CSS
- Delete me options
- Endpoint management options
- Event categories
- Geo fencing options
- Google reCAPTCHA options
- Helpdesk Options
- Linked Chains
- Lockout Options
- Login Options
- Logon Filter for AD
- Mail sender
- Password Filter for AD
- Reporting Options
- SMS sender
- Services Director Options
To configure a policy, perform the following steps:

1. Click Policies in the Administration portal.
2. Click the Edit icon against the policy you want to configure.
   You can also double-click on the policy to edit the configuration.
3. Make the required changes for a specific policy.
4. Click Save.

**IMPORTANT:** The configured policies are applied for all the Advanced Authentication servers.

### Authenticator Management Options Policy

This policy allows you to configure the following two settings:

- “Enabling Sharing of Authenticators for the Helpdesk Administrators” on page 96
- “Disabling Re-Enrollment of the Authenticators in the Self-Service Portal” on page 96

#### Enabling Sharing of Authenticators for the Helpdesk Administrators

This setting allows a user to authenticate with his or her authenticator to another user’s account. The helpdesk administrator can share an authenticator of one user with another user.

To enable sharing authenticators, set **Enable sharing of authenticators** to **ON**.

The account of an helpdesk administrator must be added to the **SHAREAUTH ADMINS** group to grant privilege to share the authenticators. For more information about how to allow the helpdesk administrators to share authenticators, see “Local Repository”.

**NOTE:** Shared authenticators work only in the online mode. Cached login does not work for the shared authenticators. The supported methods for sharing authenticators are TOTP, HOTP, Password, Fingerprint, Card, and FIDO U2F.

For more information, see “Sharing Authenticators” in the Advanced Authentication - Helpdesk Administrator guide.

#### Disabling Re-Enrollment of the Authenticators in the Self-Service Portal

This setting allows you to restrict users from re-enrolling, editing, and deleting the enrolled authenticators in the Self-Service portal.

**NOTE:** This setting disables re-enrollment and removal of the authenticators only in the Self-Service portal. The setting has no effect on the Helpdesk portal.

To disable re-enrollment or removal of authenticators, set **Disable re-enrollment** to **ON**.
WARNING: If you access the Administration portal with a local user credentials such as `local\admin`, you might get into a lockout situation. This can happen when the administrator's password expires and it is not possible to change the password through the Self-Service portal. Therefore, to use the Disable re-enrollment option, you must configure the access of a repository account to the Administration portal. To do this:

- Add authorized users or a group of users from a repository to the FULL ADMINS role.
- Assign chains, which contain methods that are enrolled for users, to the AdminUI event (at a minimum with an LDAP Password method).

### Cache Options Policy

In this policy, you can disable the local caching of authenticators. The policy is supported for Windows Client, Mac OS X Client, and Linux PAM Client for chains that use the methods: LDAP Password, Password, HOTP, TOTP, Smartphone (offline mode), Card, FIDO U2F, Fingerprint, and PKI.

This policy allows you to configure the following settings:

- By default, the Enable local caching option is enabled. To disable the caching, set the option to OFF and click Save.

  The caching functionality enables the storing of credentials on the Client for offline authentication, when the Advanced Authentication server is not available. Therefore, a user who has successfully logged in once to the server with the authentication, can now login with the offline authentication.

- By default, the Cache expire time is set to 0, to indicate that the cache never expires. Use the Cache expire time option to set the duration (in hours) to store user authenticators in Client cache. The maximum expiry time that you can set is 24 * 366 (8784 hours). This setting is applicable for the Advanced Authentication Clients.

  When a user logs in with cached authenticators, Advanced Authentication compares the last online login time with the current offline authentication time. If the time duration is less than or equal to the specified duration in Cache expire time, the user is authenticated to Clients.

  For example, consider the Cache expire time is set to 2 hours. The last online log in time of the user to Client is 1:00 PM. When the user tries to log in to Windows Client using cached authenticator credentials at 2:30 PM, the authentication is successful and the user is logged in to Windows Client. But, if the user tries to log in with cached authenticator credentials at 4:00 PM, the offline authentication fails and displays the following message as the cache has expired.

  Authenticators of `<user name>` were not cached. Press OK and try again to log in as local user or cached user

NOTE: You can use the enforced cached logon instead of the default online logon, to improve the logon and unlock speed on Clients. For more information, refer to the following topics:

- For Linux, see “Configuring the Enforced Cached Login” in the “Advanced Authentication - Linux PAM Client” guide.
- For mac OS, see “Configuring the Enforced Cached Logon” in the “Advanced Authentication - Mac OS X Client” guide.
- For Windows, see “Configuring the Enforced Cached Login” in the “Advanced Authentication - Windows Client” guide.
Custom Messages

In this policy, you can customize the error messages, method message and prompt message of a specific language.

For example, you can customize the default logon error message in English to *Your login failed.* In the Self-Service portal, when the user specifies wrong user name, the customized error message is displayed.

To customize the messages, perform the following tasks:

- Customizing Messages in the Custom Localization File
- Customizing a Specific Message on the Portal

**NOTE:** The customized messages are cached in the Advanced Authentication server. The refresh interval for custom messages is one hour. Therefore, when you customize a message or upload a custom localization file, the respective message is displayed on the corresponding Advanced Authentication portals after an hour.

You can also perform the following tasks in the Custom Messages section:

- Customize the authentication request message displayed on the app. For more information, see Customizing Authentication Request Message For Smartphone Method.
- Customize the prompt messages of authentication methods for RADIUS event. For more information, see Customizing Prompt Messages of the Authentication Methods for RADIUS Event.
- Customize message on the clients. For more information, see Customizing the Message for Clients.

Customizing Messages in the Custom Localization File

To customize preferred messages using the Custom localization file, perform the following steps:

1. Click Custom Messages.
2. Perform one of the following action to download the custom localization file on your local drive:
   - Click **Download original** to save the `custom_messages.tar.gz` file that contains the default messages.
   - If you have customized the messages, click **Download current messages** to save the `current_custom_messages.tar.gz` file that contains the latest messages.
3. Extract the files from the `custom_messages.tar.gz` file.
4. Navigate to the preferred language folder.
   - To customize English messages, use the `custom_messages.pot` file and for other languages use the `custom_messages.po` file.
5. Open the `custom_messages.pot` file in the text format.
6. Specify the message in the `msgstr ""`. 
7 Save the changes.
8 Compress the custom_messages folder to .tar.gz or .zip format.
9 Click Browse and select the compressed custom_messages file from the local drive.
10 Click Upload.

Customizing a Specific Message on the Portal

To customize a specific message on the portal, perform the following steps:

1 Click Custom Messages.
2 Use the Message filter to search for a specific message or you can find the preferred message manually.
3 Use the Message Group to search a specific message by group. Options available are All, Method messages, Error messages, and Other messages.
4 Click the Edit 🆕 icon next to the preferred message. You can also double-click on the message to edit the content.
5 Specify the message in the preferred language.
6 Click Save.
Customizing Authentication Request Message For Smartphone Method

You can customize the authentication request message that is displayed on the NetIQ Auth app when user initiates Smartphone authentication. The authentication can be either to the endpoint or to the Advanced Authentication portals.

To customize the message for smartphone method, perform the following steps:

1. Click Custom Messages.
2. Search for one of the following keys:
   - `method.smartphone.authentication_hint` to edit the request message specific to endpoint authentication.
   - `method.smartphone.authentication_hint_no_endpoint` to edit the request message for any authentication that does not use endpoint such as Advanced Authentication portals login.
3. Click for the preferred key.
4. Specify any of the following parameters in the preferred language message as per your requirement:
   - `{user}` to fetch the user name.
   - `{client_ip}` to fetch the client IP address.
   - `{event}` to fetch the event name.
   - `{tenant}` to fetch the tenant name.
   - `{endpoint}` to fetch the endpoint name.
5. Click Save.

**NOTE:** The customized authentication request message reflects on the NetIQ smartphone app after an approximate delay of one hour.

For example, to customize the endpoint specific authentication message for the smartphone method you must search the key `method.smartphone.authentication_hint` and specify the message `{user} requested for authentication request from the client `{client_ip}` for the `{event}` to access the `{endpoint}` in the field corresponding to English language. When the user tries to authenticate to Windows Client using the smartphone method then the customized message is displayed on the NetIQ smartphone app as:

Bob requested for authentication request from the client 10.3.10.5 for the Windows logon to access the Windows-machine-589.
Customizing Prompt Messages of the Authentication Methods for RADIUS Event

You can customize prompt messages of the authentication methods that are configured for the RADIUS event. The customized prompt messages are displayed when a user initiates authentication to the RADIUS event using the configured methods.

To customize prompt message, perform the following steps:

1. Click Custom Messages.
2. Use the Message filter to search for a specific prompt message or you can find the preferred message manually.
   For example, specify radius.totp.prompt to search the prompt message displayed on RADIUS client for the TOTP method.
3. Click the Edit icon or double-click on the preferred message to edit the content.
5. Click Save.

For example, consider Thomas, an administrator, wants to customize the default prompt message of the Voice OTP method that is configured for the RADIUS event. Thomas must first search the key radius.voice_otp.prompt and modify the message to Specify the OTP that you heard from the voice call in the text box corresponding to English.

When Mark, an end user tries to authenticate to RADIUS event using the Voice OTP method, the customized prompt message is displayed.

Customizing the Message for Clients

You can customize the error messages, method message and prompt message specific to any authentication method that is displayed on endpoints such as Windows, Linux PAM, and Mac OS Clients.

To customize the message for clients, perform the following steps:

1. Copy the aucore_custom.zip custom localization file from one of the following path based on the Client:
   - Windows: C:\Program Files\NetIQ\Windows Client\locale\n   - Linux PAM: /opt/pam_aucore/locale/
   - Mac OS X: Library/Security/SecurityAgentPlugins/aucore_login.bundle/Contents/Resources/aucore/locale
2. Navigate to Policies > Custom Messages in the Administration portal.
3. Click Choose file and select the custom localization file.
4. Click Upload.

   **NOTE:** You can find the messages specific to the Clients with the prefix client. in the Key.

5. Search a specific message using the Message filter or find the preferred message manually.
   For example, specify client.method.smartcard.waiting_for_card to search the message displayed for the Card method on all clients.
6 Click **Edit** next to the preferred message. You can also double-click on the message to edit the same.

7 Specify the message in the preferred language.

8 Click **Save**.

**NOTE:** The customized messages reflect on the respective Clients after an approximate delay of one hour. However, after the first online log in to the Client, users can view the customized messages.

For example, consider Thomas, an administrator wants to customize the default method message (**Enter one-time password**) of the TOTP method that is displayed for all clients. In the key `client.method.totp.password`, Thomas can modify the default message to **Specify the OTP** that is displayed on Token or App in the text box corresponding to English language.

When Mark, an end user tries to authenticate to any client using the TOTP method, the customized method message is displayed.

**Custom CSS**

This policy allows you to use a customized css for all the Advanced Authentication portals.

To use a customized css, perform the following steps:

1 Place the css file in **Content**.
   
   For example, you can place the following sample css file.

```css
body {
  color: #000000;
  background-image: url("http://cgcreative.com/videos/poster/MicroFocus_2017_Brand_Cutdown_AMC_01.jpg") !important;
}

.skin-ias .main-header {
  background: linear-gradient(90deg,#0ecce4,#5c1bd7);
  color: #ffffff;
}

table.table-hover tr:hover td {
  background-color: #808080;
}

.skin-ias .sidebar-menu li a:hover {
  background-color: #808080;
}

.skin-ias .sidebar-menu li.active.open {
  background-color: #D3D3D3;
}

.content-wrapper {
  color: #000000;
  background: transparent !important;
}

.well {
  background: transparent !important;
  border: 0px;
  border-radius: 0px;
}
2 Click Save.

To revert the changes, remove the custom code from Content and click Save.

**Delete Me Options**

In this policy, you can configure settings that enable deleting all the user data from the server, including the enrolled methods.

When you set **Enable the Delete me policy** to ON, the users can view the **Delete me** option in a drop-down by clicking on the user name on the top-right corner of the Self-Service portal.

**NOTE:** To comply with General Data Protection Regulation (GDPR), you must set the **Enable the Delete me policy** option to ON.

**Endpoint Management Options**

In this policy, you can configure the following settings for managing an endpoint:

- **Require the administrator password to register an endpoint or workstation:** Set this option to **ON** for registering an untrusted endpoint from any IP address. Typically, this option is configured along with **Whitelist IP address**.

  You must disable the option when installing any components from the Advanced Authentication distributives package that uses endpoints (Advanced Authentication Windows Client, Mac OS X Client, Linux PAM Client, Logon Filter, and RDG plug-in). Otherwise, the endpoints are not created. You must use the option for third-party integrations only.

- **Whitelist IP Address:** Add the preferred IP addresses to the **Whitelist IP Address** to register either a trusted or an untrusted endpoint from these IP addresses. You can add a single IP address, multiple IP addresses, or a range of IP addresses to the whitelist. The IP address must be in IPv4 or IPv6 format.
The following conditions summarizes the use of endpoint management options:

- **Whitelist IP Address** is empty and **Require the administrator password to register an endpoint or workstation** is **OFF**: Untrusted endpoints can be registered from any IP address without the administrator’s credentials.

Regardless of the status of **Require the administrator password to register an endpoint or workstation** and **Whitelist IP Address** options, the administrator’s credentials are required to perform the following actions:

- To delete and update any endpoint.
- To register a trusted endpoint.

Endpoint registration is restricted only from those IPs that are specified in **Whitelist IP Address**.

- **Whitelist IP Address** is empty and **Require the administrator password to register an endpoint or workstation** is **ON**: The administrator’s credentials are required to register an untrusted endpoint from any IP address.

- IP addresses are specified in **Whitelist IP Address** and **Require the administrator password to register an endpoint or workstation** is **ON**: The administrator’s credentials are required to register untrusted endpoints only from the IP addresses specified in the whitelist.

The endpoint registration request from any other IP address that is not specified in the whitelist is blocked automatically.

### Event Categories

In this policy you can add categories, which can be used in an event to support multiple enrollments for a method. For each event, you can specify one category.

To add a category, perform the following steps:

1. Click **Event categories**.
2. Click **Add**.
3. Specify a name and description for the category.
4. Click **Save**.
5. Click **Events** and edit the required event to specify the category.

Ensure that users or helpdesk administrators enroll authenticators for the new category.

**NOTE:**

- You can enroll only one authenticator of one type for each category.
- The **Authenticator category** option in **Events** is not displayed when no category is created.
- The LDAP Password method is an exception. There is one LDAP password authenticator always, it can be used with any category.

### Geo Fencing Options

In this policy, you can create authentication zones by drawing boundaries for a geographical location. When you enable the geo-fencing policy, users can authenticate with their Smartphones only from the allowed geographical locations.

To enable geo-fencing, set **Enable Geo-fencing** to **ON**. For more information about how to configure the geo-zones, see the **Smartphone** method.
NOTE: When you enable the Geo-fencing options policy, the functioning of the TOTP mode of the Smartphone method, which is used in the offline mode, is affected. An error message TOTP login is disabled is displayed to the users when they try to authenticate with this method.

Google reCAPTCHA Options

The Google reCAPTCHA Options policy helps to prevent the Advanced Authentication web portals login page from bots and to confirm that the user is a human and not a robot. This policy adds an additional layer of security before users go through multi-factor authentication. A series of images are displayed and the users must select the images for the specified condition to login.

To configure the Google reCAPTCHA for Advanced Authentication, you must perform the following configuration tasks:

- “Registering the Google reCAPTCHA Account” on page 105
- “Configuring Google reCAPTCHA for Advanced Authentication” on page 106
- “Enabling the Google reCAPTCHA Options Policy for Events” on page 106

Registering the Google reCAPTCHA Account

Before you configure Google reCAPTCHA in Advanced Authentication, you must have a Google reCAPTCHA account.

To register for the Google reCAPTCHA account, perform the following steps:

1. Log in to the Google reCAPTCHA website with your Google account.
2. Click Get reCAPTCHA.
3. Specify a Label, select reCAPTCHA V2 from Choose the type of reCAPTCHA.
4. Specify the IP address or the domain name of the Advanced Authentication server in Domain.
5. Accept the terms of Google reCAPTCHA.
6. Click Register.
7. Copy the Site key and Secret key to configure reCAPTCHA in Advanced Authentication. For more information, see Configuring Google reCAPTCHA for Advanced Authentication.

NOTE: If you forget the generated secret key, you can retrieve it from your Google account.

WARNING: If you have enabled the Google reCAPTCHA policy for the Admin UI event, you must consider the following guidelines. Otherwise, a deadlock scenario can happen and you will not be able to access the Administration portal without the cluster re-installation:

- If the site key or secret key gets deleted at the Google server, you will not be able to get the same site key or secret key. The site key and secret key used on the Administration portal are no more valid and there is no way to bypass the reCaptcha on the Administration portal.
- If you have registered the reCAPTCHA for one domain name and you change the domain name or migrate the Advanced Authentication server to another domain name, the site key or secret key used on the Administration portal are no more valid.
Configuring Google reCAPTCHA for Advanced Authentication

To configure Google reCAPTCHA for Advanced Authentication, perform the following steps:

1. Log in to the Administration portal.
2. Click Policies > Google reCAPTCHA Options.
3. Specify the Site Key and Secret Key that you received when you registered for a Google reCAPTCHA account.
   For more information about how to register the Google reCAPTCHA account, see “Registering the Google reCAPTCHA Account”.
4. Click Test to test the policy after the configuration.
5. Click Save.

Enabling the Google reCAPTCHA Options Policy for Events

After you configure the Google reCAPTCHA policy, you must enable the policy for the respective events.

To enable the policy for events, perform the following steps:

1. Click Events.

   **NOTE:** You can enable the Google reCAPTCHA policy only for the Admin UI event, Authenticators Management event, Helpdesk event, Helpdesk User event, Report logon event, Tokens Management event, and Web authentication events such as OAuth and SAML 2.0 events.

2. Set Enable Google reCAPTCHA to ON.
3. Click Save.

Helpdesk Options

In this policy, you can configure the following settings for the Helpdesk portal:

- **Ask for the credentials of the managed user:** Set this to ON to prompt the helpdesk administrator to provide the credentials of the managed user in the Helpdesk portal. This enhances security, however reduces convenience of the operations.

  When this setting is enabled, the helpdesk administrator must know the users’ credentials to manage their authenticators. Ensure that you have specified a chain (with all the methods of the chain enrolled for the users) for the Helpdesk User event. When you set the option to OFF, the user management becomes faster, but less secure.

- **Allow to unlock user accounts:** Set to ON to allow a helpdesk administrator to unlock users who are locked in the Advanced Authentication server local repository. Users are locked when the Lockout options policy is enabled. The helpdesk administrator can view and unlock the users in the Helpdesk portal under the Locked Users tab.

- **Allow to manage endpoints:** Set Allow to manage endpoints to ON to allow a helpdesk administrator to manage the endpoints of the Advanced Authentication server. When the helpdesk administrator logs in to the Helpdesk portal, an Endpoints tab is displayed where all the endpoints are listed. The helpdesk administrator can remove the endpoints. This option is disabled by default. For more information, see Managing Endpoints.
Linked Chains

This policy allows you to perform the following settings:

- **Enable linked chains**: This policy allows users to use a simple chain within a few hours of authentication done with a high-security chain. You must enable this policy for the Require chain option while creating a chain.

**NOTE**: This policy has replaced the Last Logon Tracking Options policy.

For example, if a user authenticates with the LDAP Password+Card chain once in a day, the user can further use a linked chain with only the Card method without the LDAP Password method, or if a user authenticates with the Fingerprint+Smartphone chain once in every four hours, the user can authenticate once with this chain and next authentication he can use only the linked Smartphone chain. The duration for which he can use the linked chain depends on the grace period that you specify in the Require chain option.

- **Hide required chain**: After using the required chain within the grace period, a user will see both the required and linked chain on Windows Client, Mac Client, and Linux PAM Client. This policy allows to hide the required (high-security) chain after you authenticate once. Therefore, instead of displaying both the chains, after authenticating with the required chain, only the linked chain will be displayed. By default, this policy is disabled. Enable the policy to hide the high security chain.

Lockout Options

In this policy, you can configure settings to lock a user’s account when the user reaches the maximum failure attempts of login. This enhances security by preventing the guessing of passwords and one-time passwords (OTPs).

You can configure the following options in this policy:

- **Enable**: An option to enable the lockout settings.
- **Attempts failed**: The limit of failure attempts of authentication, after which the user’s account is locked. The default value is 3.
- **Lockout period**: The period within which the user’s account is locked and the user cannot authenticate. The default value is 300 seconds.
- **Lock in repository**: The option to lock the user account in repository. You cannot use Lockout period if you enable this option. Only the system administrator must unlock the user in the repository.

**IMPORTANT**: You must configure the appropriate settings in your repository for the options to function appropriately. For Active Directory Domain Services, you must enable the Account lockout threshold policy on Domain Controllers.

For NetIQ eDirectory, you must configure the Intruder Detection properly.

After a user’s account is locked (not in the repository), you can unlock the user account. To do this, click Repositories > Edit > Locked Users and click Remove against the user’s account name.

The Helpdesk administrator can also unlock the locked users, if the Allow to unlock user accounts is enabled in the Helpdesk Options policy.
Login Options

In this policy, you can configure the settings to add default repository and ensure not to disclose valid username for malicious attack.

This policy allows you to configure the following settings:

- **Default repository**: You can add repositories that are used as default repositories. Therefore while logging in, you need not prefix the repository name before the username for authentication.

  For example, if pjones is a member of the company repository, then while logging in, instead of specifying `company\pjones`, you can specify only `pjones`.

  To add a repository as default, move the repository from Available to Default and click Save.

- **Username disclosure**: This option is set to OFF by default. It is recommended to keep default setting to prevent security vulnerabilities and to make it difficult for hackers to predict the valid username.

  If you set Username disclosure to ON and a user specifies an invalid username on the Advanced Authentication login page, an error message User not found is displayed. When the user specifies a valid username, the associated chain details are prompted to confirm the specified username and disclosing valid username. This can cause security vulnerability making it easy for attackers to guess the valid username.

  When this option is set to OFF, chain details are displayed instead of error message even though a user specifies an invalid username on the login page. A user can select a preferred authentication method. If the input data specific to the selected method is incorrect, a generic message Invalid credentials is displayed. This does not disclose whether username or first-factor authentication is incorrect.

  For example, a user specifies an invalid username, selects the SMS OTP method from the authentication chain. In this case, the SMS with OTP is not sent to the user. If the user specifies some random 6 digit as OTP, the server prompts an error message Incorrect OTP password.

  This helps the user to determine that specified username is valid though it is invalid.

- **LDAP caching**: This option allows you to enable or disable the caching of a user's information on the Advanced Authentication server. This information can be the lockout status of users, whether users have been disabled, or about the expiry of a user's password.

  By default, the option is set to OFF. This indicates that the Advanced Authentication server communicates with the LDAP server each time to check a user's information. You can enable the option to allow the caching of a user's information. Enabling the option increases the performance. However, it may also lead to security vulnerabilities. Therefore, it is recommended to set the option to OFF.

Logon Filter for Active Directory

In this policy you can configure settings to enable the use of Logon Filter that you must install on all the Domain Controllers in the domain and configure it. Logon Filter allows you to automatically update group membership if you login with the Advanced Authentication Windows Client.

To enable the policy, set Enable filter to ON and click Save.

**NOTE:** Before enabling the policy, you must ensure the Advanced Authentication Logon Filter is installed on all the Domain Controllers in the domain. Else, you might face problems with password validation during password synchronization on workstations that have the Windows Client installed.

For information about how to configure Logon Filter, see Configuring Logon Filter.
Mail Sender

In the Mail sender policy, you can configure settings for the Email OTP method to facilitate sending email messages with one-time passwords to users.

To configure the Mail sender settings, perform the following steps:

1. Specify the following details:
   1. **Host**: The outgoing mail server name. For example, smtp.company.com.
   2. **Port**: The port number. For example, 465.
   3. **Username**: The username of an account that is used to send the authentication email messages. For example, noreply or noreply@company.com.
   4. **Password**: The password for the specified account.
   5. **Sender email**: The email address of the sender.
   6. **Recipient Mask**: Specify the masked value that you want to display for the email.
      The email address of the users value is masked when users authenticate with the email method.

   **NOTE**: The default value is set and if you do not change the Recipient Mask value, the default value is considered for masking of the email address.

   7. **TLS and SSL**: The cryptographic protocol used by the mail server.

2. You can test the configurations for the Mail sender policy in the Test section.
   2a. Specify the email address in E-mail to which you want to send the Email OTP.
   2b. Specify a message to be sent to the phone in Message.
   2c. Click Send test message!.

3. Click Save.
   Real messaging uses async sender. Ensure that you have configured a chain with the Email OTP method and assigned it to an event. Login to the Self-Service portal and test the Email authenticator. If it does not work, click async log.

Authentication Flow

The authentication flow for the Mail sender is described in the following image.
A user wants to authenticate on an endpoint such as a laptop or a website with the Email OTP method. The following steps describe the authentication flow:

1. When the authentication request is initiated, the endpoint contacts the Advanced Authentication server.
2. The Advanced Authentication server validates the user’s credentials and gets an email address of the user from a repository.
3. Advanced Authentication server sends the request to a configured mail server to send an email message with the content that includes a one-time password (OTP) for authentication.
4. Mail server sends the message to the user’s email address.
5. Mail server sends the sent signal to the Advanced Authentication server.
6. Advanced Authentication server sends a request to the user to specify an OTP on the endpoint.
7. The user specifies the OTP from the email message. The Advanced Authentication server gets the OTP.
8. Advanced Authentication server validates the authentication. The authentication is done or denied.

HTTPS protocol is used for the internal communication.

Access configuration

Advanced Authentication server - Mail Server (SMTP, outbound).

Password Filter for Active Directory

In this policy, you can configure settings to synchronize the password update between the appliance and Active Directory through the Password Filter. The Password Filter automatically updates the LDAP Password stored in Advanced Authentication, whenever the password is changed or reset in the Active Directory. This helps you to authenticate without getting any prompt to synchronize the password after it is changed or reset.

You can perform the following settings in this policy:

- Set Update password on change to ON to update the LDAP password automatically in Advanced Authentication when it is changed in the Active Directory. This helps you to authenticate without getting a prompt to synchronize the password after it is changed.

  Set Update password on change to OFF to prompt the user to synchronize the LDAP password while logging in to Windows when the password is changed in the Active Directory.

- Set Update password on reset to ON to update the LDAP password automatically in Advanced Authentication when it is reset in the Active Directory. This helps users to authenticate without getting a prompt to synchronize the password if it is reset.

  Set Update password on reset to OFF to prompt the user to synchronize the LDAP password while logging in to Windows when the user’s password has been reset in the Active Directory.
NOTE: If Enable local caching is set to ON in the Cache Options policy and when the password is changed or reset in the Active Directory. Then, a user is prompted to synchronize the password while logging in to Windows irrespective of the status of the following Password Filter for AD settings:

- Update password on change
- Update password on reset

If Enable local caching is set to OFF, the Password Filter works according to the settings configured in this policy.

NOTE: Endpoint for the Password Filter must be trusted. To do this, perform the following steps:

1. Click Endpoints in the Advanced Authentication Administration portal.
2. Edit an endpoint of the Password Filter.
3. Set Is trusted to ON and add a description.
4. Save the changes.

**Reporting Options**

In this policy, you can configure settings to delete the history about the login information of users that is recorded in the reports.

Specify a value in History max age(days). The default value is 30 (days). This indicates that the history about the login information of users will be recorded from the current date to the previous 30 days. Any data before that will be deleted.

**SMS Sender**

In this policy, you can configure the settings for the SMS OTP method. The SMS OTP method sends SMS messages with one-time passwords to the users. Advanced Authentication contains predefined settings for Twilio and MessageBird services.

The Sender Service consists of the following three options:

- Generic
- Twilio
- MessageBird

To configure SMS sender manually perform the following steps:

1. Select Generic in Sender service.
2. Recipient Mask: Specify the masked value that you want to display for the SMS.
   The SMS OTP of the users is masked when users authenticate with the SMS OTP method.

   NOTE: The default value is set and if you do not change the Recipient Mask value, the default value is considered for masking of the SMS OTP.

3. Specify a Service URL value. For example, Clickatell http://api.clickatell.com/http/sendmsg?.
4. Leave HTTP Basic Authentication Username and HTTP Basic Authentication Password blank.
5 Select POST from HTTP request method.

6 Click Add and create the following parameters in HTTP request body.
   - name: user
     value: name of your account
   - name: to
     value: {phone}
   - name: text
     value: {message}
   - name: api_id, this is a parameter that is issued after addition of an HTTP sub-product to your Clickatell account. A single account may have multiple API IDs associated with it.
   - name: from
     value: sender’s phone number

7 Click Add secure and create the following parameter in HTTP request body.
   - Name: password
     Value: current password that is set on the account

For more information about the additional parameters for Clickatell, see the Clickatell documentation.

**NOTE:** The parameters may differ for different SMS service providers. But the {phone} and {message} variables are mandatory.

To configure SMS sender settings for Twilio service, perform the following steps:

1 Select Twilio in Sender service.

2 Recipient Mask: Specify the masked value that you want to display for the SMS.
   The SMS OTP of the users is masked when users authenticate with the SMS OTP method.

**NOTE:** The default value is set and if you do not change the Recipient Mask value, the default value is considered for masking of the SMS OTP.

3 Specify the following details:
   - **Account sid** and **Authentication token:** In Twilio, the Account SID acts as a username and the Authentication Token acts as a password.
   - **Use Copilot:** The copilot option is used to send SMS from a Twilio’s phone number of your location. This is helpful when SMS messages have to be sent across the geographical locations. For example, with copilot, SMS will be sent from Indian phone number to the Indian users. Without copilot, SMS will be sent from US phone number to the Indian users.

For more information on Copilot option and its features, see https://www.twilio.com/copilot#phone-number-intelligence and https://www.twilio.com/docs/api/rest/sending-messages-copilot#features.
   - **Messaging Service SID:** Service SID.
   - **Sender phone:** Sender’s phone number.

For more information, see the Twilio website.
To configure SMS sender settings for MessageBird service, perform the following steps:

1. Select MessageBird in Sender service.
2. **Recipient Mask:** Specify the masked value that you want to display for the SMS. The SMS OTP of the users is masked when users authenticate with the SMS OTP method.

   **NOTE:** The default value is set and if you do not change the Recipient Mask value, the default value is considered for masking of the SMS OTP.

3. Specify the **Username**, **Password**, and **Sender name**.

   For more information, see the MessageBird website.

**IMPORTANT:** MessageBird API v2 is not supported. To activate MessageBird API v1, perform the following steps:

1. Go to the MessageBird account.
2. Click **Developers** in the left navigation bar and open the API access tab.
3. Click **Do you want to use one of our old API's (MessageBird V1, Mollie or Lumata)? Click here**.

You can test the configurations for the SMS sender policy in the **Test** section.

1. Specify the phone number in **Phone** to which you want to send the SMS OTP.
2. Specify a message to be sent to the phone in **Message**.
3. Click **Send test message!**.
4. Click **Save**.

Real messaging uses async sender. Ensure that you have configured a chain with the **SMS method** and assigned it to an event. Then sign-in to the Self-Service portal and test the SMS authenticator. If it does not work, see the **async logs**.

**Authentication Flow**

The authentication flow for the SMS sender in Advanced Authentication is described in the following image.
A user wants to authenticate on an endpoint such as a laptop or a website with the SMS method. The following steps describe the authentication flow:

1. When the authentication request is initiated, the endpoint contacts the Advanced Authentication server.
2. The Advanced Authentication server validates the user’s credentials and gets a phone number of the user from a Repository.
3. Advanced Authentication server sends the request to a configured SMS Service Provider to send an SMS message with the content that includes a one-time password (OTP) for authentication.
4. SMS Service Provider sends the SMS message to the user's phone.
5. SMS Service Provider sends the 'sent' signal to the Advanced Authentication server.
6. Advanced Authentication server sends a request to the user to specify an OTP on the endpoint.
7. The user specifies the OTP from the SMS message. The Advanced Authentication server gets the OTP.
8. Advanced Authentication server then validates the authentication. The authentication is done or denied.

HTTP/HTTPS protocol is used for the communication.

**Access configuration**

Advanced Authentication server - SMS Service Provider (HTTP/HTTPS, outbound).

**Services Director Options**

In this policy, you can configure settings required to integrate with the Services Director.

Perform the following steps to configure this policy:

1. Set Enable integration to **ON** to enable the integration of Advanced Authentication with Services Director.
2. Specify the Public DNS name of Advanced Authentication, Services Director DNS Name, Tenant administrator name, and Tenant administrator password of Services Director to integrate it with Advanced Authentication.

**Voice Sender**

In this policy, you can configure the settings for the Voice and Voice OTP methods. Advanced Authentication supports the Twilio service for the Voice methods.

To configure Voice Sender settings for Twilio service, perform the following steps.

1. **Recipient Mask**: Specify the masked value that you want to display for the Voice OTP.
   - The Voice OTP of the users is masked when users authenticate with the Voice OTP method.

   **NOTE**: The default value is set and if you do not change the Recipient Mask value, the default value is considered for masking of the Voice OTP.

2. Specify the following details in the Voice sender policy:
   - **Account sid** and **Authentication token**: In Twilio, the Account SID acts as a username, and the Authentication Token acts as a password.
- **Sender phone**: The phone number of the sender.
- **Server url**: The public URL to which the Twilio service connects for authentication. You can use http protocol for testing purpose, but for production environment you must use https protocol. You must have a valid certificate when you use https.

3 You can test the configurations for the Voice sender policy in the **Test** section.

3a Specify the phone number in **Phone** to which you want to send the Voice OTP.

3b Specify a message to be sent to the phone in **Message**.

3c Click **Send test message!**.

4 Click **Save**.

Real messaging uses async sender. Ensure that you have configured a chain with the **Voice OTP** method and assigned it to an event. Then sign-in to the Self-Service portal and test the Voice authenticator. If it does not work, see the **async logs**.

**IMPORTANT:** The users may receive calls with the **voice Application error**. This happens because of incorrect settings or invalid certificates. Ensure that the certificate is valid and is not expired. Invalid certificates cannot be applied by Twilio.

### Authentication Flow

The authentication flow for the Voice sender in Advanced Authentication is described in the following image.

A user wants to authenticate on an endpoint such as a laptop or a website with the **Voice Call** method. The following steps describe the authentication flow:

1 When the authentication request is initiated, the endpoint contacts the Advanced Authentication server.

2 The Advanced Authentication server validates the user’s credentials and gets a phone number of the user from a repository.

3 Advanced Authentication server sends the request to a configured voice call service provider (Twilio) to call the user.

4 The voice call service provider calls the user.

5 The user picks up the phone, listens to the call, and specifies the PIN followed by the hash (#) sign.
Voice call provider sends the specified PIN to the Advanced Authentication server.

Advanced Authentication server then validates the authentication. The authentication is done or denied.

HTTP/HTTPS protocol is used for the communication.

**Access configuration**

Advanced Authentication server - Voice Call Service Provider (HTTP/HTTPS, inbound/outbound).

**Web Authentication**

This policy replaces the SAML 2.0 options policy. The Web Authentication policy allows you to configure the following settings:

- Configuring Settings for the SAML 2.0 Events
- Customizing the Login Page of Web Authentication Events
- Customizing Messages and Authentication Method Names for the Web Authentication Events

**Configuring Settings for the SAML 2.0 Events**

You can configure the settings to specify the Identity Provider’s URL to download the SAML 2.0 metadata file. The downloaded SAML 2.0 metadata file is used to configure the service provider.

For more information about configuring this policy, see “SAML 2.0”.

**NOTE:** From Advanced Authentication 6.1 onwards, the web authentication services such as SAML 2.0, OAuth 2.0 are available only on the 8443 port. Any OAuth 2.0 or SAML 2.0 requests to the 443 port (default SSL) are redirected with a 308 status to the 8443 port. The third-party solutions that integrates with Advanced Authentication using the OAuth 2.0 or SAML 2.0 services must manage the URL redirection or append the port number 8443 with the IP address or domain name in the following format:

https://<ip address>:8443/osp... or https://<dns name>:8443/osp...

**Customizing the Login Page of Web Authentication Events**

You can customize the login page of the OAuth 2.0, SAML 2.0, or Open ID Connect events. To do this, perform the following steps:

1. Set Custom Branding to ON.
2. Click Download Template.
3. Save the osp-custom-resources.jar file.
4. Unzip the osp-custom-resources.jar file and in the resources folder open the file that you want to customize.

For example, to edit the custom branding in the English language, customize the oidp_enduser_custom_resources_en_US.properties file.

**NOTE:** Ensure that you edit the attributes in the Login page properties section of the oidp_enduser_custom_resources_en_US.properties file for the custom branding of the login pages in the English language.
5. After you edit the specific file in the `resources` folder, zip the file `osp-custom-resources.jar`.
6. Click `Browse` to upload the `osp-custom-resources.jar` file in the Web Authentication policy.
7. Click `Save`.

**NOTE:** When you upload the custom branding changes for the first time, you must restart the Advanced Authentication server to reflect the changes on the login pages of the web authentication events. This is applicable per tenant.

You can also add your customized `.css` file in the `css` folder of the `osp-custom-resources.jar` file.

The following section describes an example of the customization that you can achieve for the Web authentication.

### Example of Customizing a Login Page

To achieve the customized login page in the Figure 3-1 for Acme Group of company, you can perform the following:

- “Adding a Customized CSS for the Login Page” on page 118
- “Customizing the Logo of an Enterprise” on page 120
- “Customizing the Copyrights” on page 121
- “Customizing the Branding Text” on page 121
- “Adding Links on the Login Page” on page 122

*Figure 3-1 Customized Page for Acme Group*
Adding a Customized CSS for the Login Page

You can add a customized css file to reflect changes for the login pages.

The following sample.css file has been customized for achieving the customized login page in Figure 3-1 for the Acme Group of company.

/* general styles ------------------------ */
body {
  margin:0;
  padding:0;
  background:#fff url("/osp/TOP/images/login_bg.jpg") no-repeat center center fixed;
  -webkit-background-size: cover;
  -moz-background-size: cover;
  -o-background-size: cover;
  background-size: cover;
  font-family:Arial, Helvetica, sans-serif;
}
img {
  border:none;
  max-width: 100%;
}
/* login box ------------------------------- */
div.page-container {
  position:absolute;
  top: 50%;
  left: 0px;
  width:100%;
  margin:-265px auto 0 auto;
}
div.dialog {
  border: 12px solid rgba(255, 255, 255, 0.3);
  border-radius: 2px;
  width: 318px;
  max-width:100%;
  margin:0 auto;
  background-color: transparent;
}
div.dialog-content {
  height:525px;
  padding:0 15px;
  background:url(/osp/TOP/images/acme.png);
  background-color:#414749 ;
  background-position:180px 20px;
  background-repeat:no-repeat;
  font-family: Arial, Helvetica, sans-serif;
  text-align: left;
}
.dialog-header {
  margin:0;
  padding: 150px 0 40px 0;
  color:#48c6e7;
  font-size:22px;
  font-weight:100;
  background: none;
}
div.dialog-header-content {
  display:block;
  color:#fff;
Perform the following steps to add the sample.css file to the osp-custom-resources.jar file.

1. Open the osp-custom-resources.jar file.
2. Upload your .css file to the css folder.
3. Open the resources folder.
4. Open the oidp_enduser_custom_resources_en_US.properties file to edit the custom branding of the login pages in the English language.
5. Uncomment the line

   OIDPENDUSER.LoginCss=reset.css,uistyles.css,uistyles_loginselect.css

   by removing the # sign.

   You can add your .css file here. For example, OIDPENDUSER.LoginCss=sample.css.

### Customizing the Logo of an Enterprise

You can edit the logo displayed on the login page of web authentication event using the parameter

OIDPENDUSER.LoginProductImage available in the Login page properties.

For example, to edit the logo of the login page of an OAuth 2.0 event in the English language, perform the following:

1. Open the oidp_enduser_custom_resources_en_US.properties file and edit the following attribute:
OIDPENDUSER.LoginProductImage=company_img.png.

You can also edit the .css file. The following code has been added to the sample.css file to display the logo in the Figure 3-1:

```css
div.dialog-content {
    height: 525px;
    padding: 0 15px;
    background:url(/osp/TOP/images/company_img.png);
    background-color:#414749;
    background-position:180px 20px;
    background-repeat:no-repeat;
    font-family: Arial, Helvetica, sans-serif;
    text-align: left;
}
```

2. Ensure that you add the image that you want as a logo to the images folder with the name that matches with the attribute value in OIDPENDUSER.LoginProductImage.

By default the images folder contains the image company_img.

**Customizing the Copyrights**

You can edit the copyright text displayed on the login page of web authentication event using the parameter OIDPENDUSER.50004 available under the JSP Strings.

For example, to remove the copyright note that is displayed on the login page of an OAuth 2.0 event in the English language:

1. Open the oidp_enduser_custom_resources_en_US.properties file and search the following parameter:

   ```
   #OIDPENDUSER.50004=Copyright [copy] [year] NetIQ[nbsp]Corporation, a
   Micro[nbsp]Focus company. All rights reserved
   ```

2. Uncomment the following parameter as follows:

   ```
   OIDPENDUSER.50004=
   ```

   This removes the copyright note from the web authentication event - login page.

**Customizing the Branding Text**

You can edit the branding text displayed on the login page of web authentication event using the parameter OIDPENDUSER>LoginProductName available in the Login page properties section of the oidp_enduser_custom_resources_en_US.properties file.

For example, to edit the branding of the company to Acme Group, perform the following:

1. Open the oidp_enduser_custom_resources_en_US.properties file and search the following parameter:

   ```
   #OIDPENDUSER.LoginProductName=Company[nbsp]Name[reg]
   ```

2. Edit the following parameter as follows:

   ```
   OIDPENDUSER.LoginProductName=Acme[nbsp]Group[reg]
   ```

If you want to remove the branding text Acme Group, perform the following:

1. Open the oidp_enduser_custom_resources_en_US.properties file and search the following parameter:

   ```
   #OIDPENDUSER.LoginProductName=Company[nbsp]Name[reg]
   ```
2 Uncomment the following parameter as follows:

OIDPENDUSER.LoginProductName=

This removes the branding text, Acme Group, from the web authentication event - login page.

**Adding Links on the Login Page**

You can add links for the login page of the web authentication event.

For example, if you want to add the link *Forgotten Password* that is displayed on the login page in the English language, add the following:

1 Open the `oidp_enduser_custom_resources_en_US.properties` file.
2 Add the following:

```plaintext
#OIDPENDUSER.70000=null
OIDPENDUSER.70001=https://intra.sample.net/ForgottenPassword <link where the users gets redirected to>
OIDPENDUSER.70002=Forgotten Password? <label of the link>
OIDPENDUSER.70004=_top <name of the tenant>
OIDPENDUSER.70005=LOGIN_PAGE <attribute>
```

**Customizing Messages and Authentication Method Names for the Web Authentication Events**

You can customize the messages and authentication methods name for the Web Authentication events in the Custom Messages policy. Set *Use Custom Messages* to ON to enable using the custom messages for the OAuth, SAML 2.0, or Open ID Connect events. You must customize the messages in the “Custom Messages” policy.

**Adding a License**

To add a license for Advanced Authentication, perform the following steps:

1 Click **Licenses**.
2 Click **Add**.
3 Click **Browse** and select the valid license.
4 Click **Upload** to upload the license.

A user license is consumed when a user enrolls at least one authenticator through an automatic enrollment, enrollment by a Helpdesk administrator, or self-enrollment. This is an exception for the LDAP password, as a license is not consumed for it. An automatic enrollment is done only when a user performs a first authentication.

After you add the license, following details of the license are displayed:

- License ID
- Expiry date
- Restrictions: License type and applicable restrictions.
- Usage: List of applicable authentication methods with the total and usage count of each method.

Your license might be limited to some specific authentication methods. Other methods will be unavailable in the **Methods** section.
IMPORTANT: If the multi-tenancy mode is enabled, you must add licenses for each tenant.

TIP: To free up a user's license, perform the following steps:

1. Exclude the user from a group that is assigned to chains.
2. Click Repositories and edit a repository.
3. Click Full sync to perform a full synchronization of the repository.
   The existing user's authenticators are removed.

Adding a Report

Report provides you pictorial representation of collected data. You can examine data in different combinations, display report in easy-to-understand graphs, track data at different time intervals and export the report in JSON and CSV formats to share the result with others. With reports, you can track all logins (failed or successful), users' enrollment status, authentication methods used for specific event, license information, number of active users and so on.

You can add a report with specific report type as described in Table 3-1.

Table 3-2 Report Types

<table>
<thead>
<tr>
<th>Report Type</th>
<th>Description</th>
<th>Available Attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pie chart</td>
<td>This report displays the information collected on a specific parameter and represents information in the Pie chart format. You can sort the parameter in ascending and descending order.</td>
<td>- Name: Title of the report.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Relative Time Interval: Set this option to ON to select a specific time interval from the Relative Interval. Set this option to OFF to select preferred From and To dates from the Date range.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Event Type: Types of events to display in the report. Options available are All logon events, Failed logon events and Successful logon events.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Size: Number of records to filter in the report.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Order: Sorting order of selected parameter in the Field. Options available are Ascending and Descending.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Field: The parameter on which the data is collected to display in the report. Options available are Event Name, Chain Name, Method Name, Endpoint Name and so on.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Users: To filter records of specific user from directory.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Events: To filter records of specific event.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Chains: To filter records of specific chain.</td>
</tr>
<tr>
<td>Report Type</td>
<td>Description</td>
<td>Available Attributes</td>
</tr>
<tr>
<td>-------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------------------------------------------</td>
</tr>
</tbody>
</table>
| Stacked chart     | This report displays a stacked bar chart that classifies and compares different categories of Field 1 and 2 parameters to track the maximum and minimum number of logons. X-axis represents categories of the Field 2 parameter. Y-axis represents logon count. Segments in each vertical bar represents categories of Field 1 parameter. Different colors are used to depict different categories and label for each category is displayed in upper-right corner of the report. | - Name: Title of the report.  
- Relative Time Interval: Set this option to ON to select a specific time interval from the Relative Interval. Set this option to OFF to select preferred From and To dates from the Date range.  
- Event Type: Types of events to display in the report. Options available are All logon events, Failed logon events and Successful logon events.  
- Field 1: The parameter to represent on X-axis of the report. Options available are Event Name, Chain Name, Method Name, Endpoint Name and so on.  
- Size 1: Number of records to display on the X-axis.  
- Order 1: To sort the parameter selected in the Field 1. Options available are Ascending and Descending.  
- Field 2: The parameter to represent on Y-axis of the report. Options available are Event Name, Chain Name, Method Name, Endpoint Name and so on.  
- Size 2: Number of records to display on the Y-axis.  
- Order 2: To sort the parameter selected in the Field 2. Options available are Ascending and Descending.  
- Users: To filter records of specific user from directory.  
- Events: To filter records of specific event.  
- Chains: To filter records of specific chain. |
| Activity stream   | This report displays information about user, tenant, chain, method used for authentication, and the result.                                                                 | - Name: Title of the report.  
- Relative Time Interval: Set this option to ON, select a specific time interval from the Relative Interval. Set this option to OFF, select preferred From and To dates from the Date range.  
- Event Type: Types of events to display in the report. Options available are All logon events, Failed logon events and Successful logon events.  
- Users: To filter records of specific user from directory.  
- Events: To filter records of specific event.  
- Chains: To filter records of specific chain. |
<table>
<thead>
<tr>
<th>Report Type</th>
<th>Description</th>
<th>Available Attributes</th>
</tr>
</thead>
</table>
| Enroll activity stream   | This report displays information about enrolled users: last log on time, tenant, user, method used for authentication, and event type. | ✷ Name: Title of the report.  
 ✷ Relative Time Interval: Set this option to ON to select a specific time interval from the Relative Interval. Set this option to OFF to select preferred From and To dates from the Date range.  
 ✷ Users: To filter records of specific user from directory. |
| Users                    | This report displays information about the enrolled users: tenant name, user name, enrollment status and last log on time. | ✷ Name: Title of the report.  
 ✷ Relative Time Interval: Set this option to ON to select a specific time interval from the Relative Interval. Set this option to OFF to select preferred From and To dates from the Date range. |
| Authenticators           | This report displays information about the enrolled authenticators: tenant name, user name, event category, method, comment and owner of the account. | ✷ Name: Title of the report.  
 ✷ Relative Time Interval: Set this option to ON to select a specific time interval from the Relative Interval. Set this option to OFF to select preferred From and To dates from the Date range. |
| Licenses                 | This report displays information about the license id, license validity dates (such as From and To dates), license expiry status and license warnings (regarding license expiry, exceed in user count) | ✷ Name: Title of the report.  
 ✷ Relative Time Interval: Set this option to ON to select a specific time interval from the Relative Interval. Set this option to OFF to select preferred From and To dates from the Date range. |
| Event count line chart   | This report tracks and displays logon count of all events in the appliance. X-axis represents time and Y-axis represents logon count. Each data point on the chart represents numbers of user logged on at a specific time. All the data points are plotted and connected with a line to track the maximum and minimum number of logons. | ✷ Name: Title of the report.  
 ✷ Relative Time Interval: Set this option to ON to select a specific time interval from the Relative Interval. Set this option to OFF to select preferred From and To dates from the Date range.  
 ✷ Event Type: Types of events to display in the report. Options available are All logon events, Failed logon events and Successful logon events.  
 ✷ Interval: Regular interval to track the data point on the chart.  
 ✷ Users: To filter records of specific user from directory.  
 ✷ Events: To filter records of specific event.  
 ✷ Chains: To filter records of specific chain. |
Event count line chart group by field

This report tracks and displays logon count of specific parameter. X-axis represents time and Y-axis represents logon count. Data points of different colors represent specific category of the selected parameter. The label for each category is displayed in upper-right corner of the widget. All the data points are plotted and connected with a line to track the maximum and minimum number of logons.

- **Name**: Title of the report.
- **Relative Time Interval**: Set this option to ON to select a specific time interval from the **Relative Interval**. Set this option to OFF to select preferred From and To dates from the **Date range**.
- **Event Type**: Types of events to display in the report. Options available are **All logon events**, **Failed logon events** and **Successful logon events**.
- **Interval**: Regular interval to track the data point on the chart.
- **Size**: Number of records to filter in the report.
- **Order**: Sorting order of the parameter selected in the **Field**. Options available are **Ascending** and **Descending**.
- **Field**: The parameter on which the data is collected to display in the report. Options available are **Event Name**, **Chain Name**, **Method Name**, **Endpoint Name** and so on.
- **Users**: To filter records of specific user from directory.
- **Events**: To filter records of specific event.
- **Chains**: To filter records of specific chain.

<table>
<thead>
<tr>
<th>Report Type</th>
<th>Description</th>
<th>Available Attributes</th>
</tr>
</thead>
</table>
| Event count line chart group by field | This report tracks and displays logon count of specific parameter. X-axis represents time and Y-axis represents logon count. Data points of different colors represent specific category of the selected parameter. The label for each category is displayed in upper-right corner of the widget. All the data points are plotted and connected with a line to track the maximum and minimum number of logons. | ✷ Name: Title of the report.  
✦ Relative Time Interval: Set this option to ON to select a specific time interval from the **Relative Interval**. Set this option to OFF to select preferred From and To dates from the **Date range**.  
✦ Event Type: Types of events to display in the report. Options available are **All logon events**, **Failed logon events** and **Successful logon events**.  
✦ Interval: Regular interval to track the data point on the chart.  
✦ Size: Number of records to filter in the report.  
✦ Order: Sorting order of the parameter selected in the **Field**. Options available are **Ascending** and **Descending**.  
✦ Field: The parameter on which the data is collected to display in the report. Options available are **Event Name**, **Chain Name**, **Method Name**, **Endpoint Name** and so on.  
✦ Users: To filter records of specific user from directory.  
✦ Events: To filter records of specific event.  
✦ Chains: To filter records of specific chain. |
Distinct events count line chart

This report tracks and displays distinct count of all categories in the selected parameter (Distinct values by field). X-axis represents time and Y-axis represents distinct logon count. Each data point on the chart represents unique logon count at a specific time. All the data points are plotted and connected with a line to track the maximum and minimum number of distinct logons.

- **Name**: Title of the report.
- **Relative Time Interval**: Set this option to ON, select a specific time interval from the Relative Interval. Set this option to OFF, select preferred From and To dates from the Date range.
- **Event Type**: Events to display in the report. Options available are All logon events, Failed logon events and Successful logon events.
- **Interval**: Regular interval to track the data point on the chart.
- **Distinct values by field**: The parameter on which the data is collected to display in the report. Options available are Event Name, Chain Name, Method Name, Endpoint Name and so on.
- **Size**: Number of records to filter in the report.
- **Order**: Sorting order of the parameter selected in the Field. Options available are Ascending and Descending.
- **Users**: To filter records of specific user from directory.
- **Events**: To filter records of specific event.
- **Chains**: To filter records of specific chain.
Following are the generic steps to add a custom report:

1. Click Reports in the Administration portal.
2. Click Add.
3. Specify the report title in the Name.
4. Select the preferred Report type. Options available are:
   - Pie chart
   - Stacked chart
   - Activity stream
   - Enroll activity stream
   - Users
   - Authenticators
   - Licenses
   - Servers
   - Events count line chart
   - Events count line chart grouped by field

### Report Type Description Available Attributes

<table>
<thead>
<tr>
<th>Report Type</th>
<th>Description</th>
<th>Available Attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Distinct events count line chart</td>
<td>This report displays and classifies distinct logon count of each event. X-axis represents time and Y-axis represents distinct logon count. Each data point on the chart represents unique logon count of particular event. at a specific time. All the data points are plotted and connected with a line to track the maximum and minimum number of distinct logons to particular event.</td>
<td>Name: Title of the report. Relative Time Interval: Set this option to ON, select a specific time interval from the Relative Interval. Set this option to OFF, select preferred From and To dates from the Date range. Event Type: Events to display in the report. Options available are All logon events, Failed logon events and Successful logon events. Interval: Regular interval to track the data point on the chart. Size: Number of records to filter in the report. Order: Sorting order of the parameter selected in the Field. Options available are Ascending and Descending. Field: The parameter on which the data is collected to display in the report. Options available are Event Name, Chain Name, Method Name, Endpoint Name and so on. Distinct values by field: The parameter on which the data is collected to display in the report. Options available are Event Name, Chain Name, Method Name, Endpoint Name and so on. Users: To filter records of specific user from directory. Events: To filter records of specific event. Chains: To filter records of specific chain.</td>
</tr>
</tbody>
</table>
• Distinct events count line chart
• Distinct events count line chart grouped by field

5 When the Relative time interval is set to ON, the Relative Interval is displayed to select a specific time interval. When set to OFF, the date range is displayed to select preferred From and To dates.

6 Select the preferred Event type. Options available are All logon events, Failed logon events, and Successful logon events.

7 Select number of records from the Size to display in the report.

8 Select sorting order from the Order. Options available are Ascending or Descending.

9 Select the preferred parameter from the Field. Based on the selected parameter, the data is collected to display on the report. Options available are Event Name, Chain Name, Method Name, Endpoint Name and so on.

10 Specify and select the preferred domain joined user from the Users to filter records in the report.

11 Specify and select the preferred event from the Events to filter records in the report.

12 Specify and select the preferred chain from the Chains to filter records in the report.

13 Click Save.

14 Click Reload to generate and display the report based on the selected values.
Enrolling the Authentication Methods

Advanced Authentication server supports the following ways to enroll the authentication methods:

- **Automatic enrollment:** This type of enrollment is used for the SMS, Email, RADIUS, LDAP Password, and Swisscom Mobile ID methods.

  The methods are enrolled automatically if the chains containing them are assigned to any event.

- **Enrollment by Administrator:** This type of enrollment is used for the OATH Tokens.

  An administrator can import tokens from the PSKC or CSV files in the Administration portal > Methods > OATH OTP > OATH Tokens tab. You can assign tokens to the specific users.

- **Enrollment by Helpdesk administrator:** This type of enrollment is used by the Helpdesk administrator.

  A Helpdesk administrator can access the Helpdesk portal with the address: https://<NetIQ Server>/helpdesk. In the Helpdesk portal, the Helpdesk administrator can enroll the authentication methods for users. A Helpdesk administrator must be a member of the Enroll Admins group (Repositories > Local > Edit > Global Roles) to manage users’ authenticators.

- **Enrollment by User:** This method is applicable for the users. A user can access the Self-Service portal with the address: https://<NetIQ Server>/account, where the users can enroll any of the authentication methods.
Advanced Authentication facilitates clients to integrate with the third-party solutions using the following interface.

- OAuth 2.0
- RADIUS Server
- SAML 2.0
- REST API

The information about configuring Advanced Authentication with some of the third-party solutions is as follows:

- Configuring Integration with Barracuda
- Configuring Integration with Citrix NetScaler
- Configuring Integration with Dell SonicWall SRA EX-Virtual Appliance
- Configuring Integration with FortiGate
- Configuring Integration with OpenVPN
- Configuring Integration with Salesforce
- Configuring Integration with ADFS
In OAuth 2.0 authorization, the third-party client requests access to the resources that are controlled by the resource owner. Instead of using the resource owner’s credentials to access the protected resources, the third-party client obtains an access token. The third-party clients can be web applications, mobile phones, handheld devices, and desktop applications.

This section contains the following topics:

- “Building Blocks of OAuth 2.0” on page 135
- “Sample OAuth 2.0 Application Integrated with Advanced Authentication” on page 138
- “OAuth 2.0 Attributes” on page 143
- “Non Standard Endpoints” on page 144

### Building Blocks of OAuth 2.0

The following are the building blocks of OAuth 2.0.

- OAuth 2.0 Roles
- OAuth 2.0 Grants

#### OAuth 2.0 Roles

OAuth 2.0 consists of the following four roles:

- **Resource Owner**: Entity that grants access to a protected resource. It can be a system or a person (end-user) owning the resources.
- **Resource Server**: Server that hosts the protected resources. It accepts and responds to the protected resource requests using the access tokens.
- **Client**: Application that requests and get authorization on behalf of the resource owner to access a protected resource.
- **Authorization Server**: Server that issues access tokens to the client after the successful authentication of the resource owner and obtaining authorization.

#### OAuth 2.0 Grants

By default, Advanced Authentication supports the following OAuth 2.0 grant types. However, if you require to use the Resource owner password credential grant, you have to enable it using Advanced Authentication settings. For more information on OAuth 2.0 grant types, see the link (https://tools.ietf.org/html/rfc6749).

- “Authorization Code” on page 136
- “Implicit Grant” on page 137
Authorization Code

In authorization code, an authorization server acts as an intermediary between the client and the resource owner. Instead of requesting authorization directly from the resource owner, the client directs the resource owner to an authorization server, which in turn directs the resource owner back to the client with the authorization code.

The authorization grant type depends on the method used by the application to request authorization, and the grant types supported by the API.

The following diagram describes the workflow of authorization code grant.

```
1. The OAuth client initiates the flow when it directs the user agent of the resource owner to the authorization endpoint. The OAuth client includes its client identifier, requested scope, local state, and a redirection URI.
2. The authorization server authenticates the resource owner through the user agent and recognizes whether the resource owner grants or denies the access request.
3. If the resource owner grants access, the OAuth client uses the redirection URI provided earlier to redirect the user agent back to the OAuth client. The redirection URI includes an authorization code and any local state previously provided by the OAuth client.
4. The OAuth client requests an access token from the authorization server through the token endpoint. The OAuth client authenticates with its client credentials and includes the authorization code received in the previous step. The OAuth client also includes the redirection URI used to obtain the authorization code for verification.
5. The authorization server validates the client credentials and the authorization code. The server also ensures that the redirection URI received matches the URI used to redirect the client in Step 3. If valid, the authorization server responds back with an access token.
```
Implicit Grant

The implicit grant is similar to the authorization code grant with two distinct differences:

- It is used for user-agent-based clients. For example, single page web apps that cannot keep a client secret because all the application code and storage is easily accessible.
- Secondly, instead of the authorization server returning an authorization code which is exchanged for an access token, the authorization server returns an access token.

The following diagram describes the workflow of Implicit grant.

The workflow for implicit grant includes the following steps:

1. The OAuth client initiates the flow by directing the user agent of the resource owner to the authorization endpoint. The OAuth client includes its client identifier, requested scope, local state, and a redirection URI. The authorization server sends the user agent back to the redirection URI after access is granted or denied.

2. The authorization server authenticates the resource owner through the user agent and verifies whether the resource owner grants or denies the access request.

3. If the resource owner grants access, the authorization server redirects the user agent back to the client using the redirection URI provided earlier. The redirection URI includes the access token in the URI fragment.

4. The user agent follows the redirection instructions by making a request to the web server without the fragment. The user agent retains the fragment information locally.
5. The web server returns a web page, which is typically an HTML document with an embedded script. The web page accesses the full redirection URI including the fragment retained by the user agent. It can also extract the access token and other parameters contained in the fragment.

6. The user agent runs the script provided by the web server locally, which extracts the access token and passes it to the client.

Sample OAuth 2.0 Application Integrated with Advanced Authentication

To create a sample web application, you need Python v3 (the sample script prepared on v3.4.3).

The following web application describes the functionalities supported when Advanced Authentication is integrated with OAuth 2.0. OAuth 2.0 server is an authorization and resource server. As an Authorization Server, the OAuth server can prompt the users to go through authentication chains and as a resource server, the OAuth server can prompt the users to provide user details.

You must create the following five files:

1. Sample script (oauth2_test.py)

```python
from bottle import Bottle, request, run, redirect, SimpleTemplate, template
from urllib.parse import urlencode, quote
import urllib.request
import base64
import ssl
import json

app = Bottle()

client_id = 'id-rSCzuBLoqXCATfkXZ4fssedAo8sPsWxSs'
client_secret = 'secret-9IDpzWFD26RriURR7KJ1pFxFx7V9QeDm'
redirect_uri = 'http://localhost:8088/'  # this app callback URI
authorization_endpoint = 'https://192.168.0.151/osp/a/TOP/auth/oauth2/grant'
attributes_endpoint = 'https://192.168.0.151/osp/a/TOP/auth/oauth2/getattributes'
state = {}

@app.get('/getattr')
def get_attributes():
    params = urlencode({
        'attributes': 'client username userRepository user_dn user_cn mail sid upn netbiosName',
        'access_token': state['access_token']
    })
    url = attributes_endpoint + '?' + params
    print('getattr url: {}
'.format(url))
    req = urllib.request.Request(url)
    gcontext = ssl.SSLContext(ssl.PROTOCOL_TLSv1_2)  # avoid cert checking
    with urllib.request.urlopen(req, context=gcontext) as response:
        rsp = response.read()
        attributes = json.loads(rsp.decode('utf-8'))
        return template('attributes.html', items=attributes.items(),
                        refresh_token=urllib.parse.quote(state['refresh_token']))

@app.get('/')
```
def do_get():
    code = request.query.get('code')
    if code:
        # got code from OAuth 2 authentication server
        token = get_token_code(code)
        state.update(token)
        return template('token.html', items=token.items(),
                        refresh_token=urllib.parse.quote(token['refresh_token']))
    else:
        return template('main.html')

@app.get('/logon')
def do_logon():
    pr=list(urlparse(authorization_endpoint))
    # set query
    pr[4]=urlencode({'
        'response_type': 'code',
        'client_id': client_id,
        'redirect_uri': redirect_uri
    })
    # perform redirection to OAuth 2 authentication server
    redirect(urlunparse(pr))

@app.get('/logon-implicit')
def do_logon_implicit():
    # parse authorization_endpoint URL
    pr = list(urlparse(authorization_endpoint))
    # set query
    pr[4] = urlencode({'
        'response_type': 'token',
        'client_id': client_id,
    })
    # perform redirection to OAuth 2 authentication server
    redirect(urlunparse(pr))

@app.get('/logon-creds')
def do_logon_creds():
    return template('logonform.html')

@app.post('/logon-creds')
def do_logon_creds_post():
    username = request.forms.get('username')
    password = request.forms.get('password')
    token = get_token_password(username, password)
    state.update(token)
    return template('token.html', items=token.items(),
                    refresh_token=urllib.parse.quote(token['refresh_token']))

def get_token_password(username, password):
    # prepare POST parameters - encode them to urlencoded
    data = urlencode({
        'grant_type': 'password',
        'username': username,
        'password': password
    })
    data = data.encode('ascii')  # data should be bytes
    resp_text = post_data(data, prepare_headers())
    print(resp_text)
    return json.loads(resp_text)
@app.get('/refresh')
def do_refresh():
    token = refresh_access_token(request.query.get('refresh_token'))
    state.update(token)
    return template('token.html', items=token.items(),
                    refresh_token=state.get('refresh_token', ''))

def get_token_code(code):
    # prepare POST parameters - encode them to urlencoded
    data = urlencode({
        'grant_type': 'authorization_code',
        'code': code,
        'redirect_uri': redirect_uri
    })
    data = data.encode('ascii')  # data should be bytes
    resp_text = post_data(data, prepare_headers())
    print(resp_text)
    return json.loads(resp_text)

def refresh_access_token(refresh_token):
    print('refresh_token: {}'.format(refresh_token))
    # prepare POST parameters - encode them to urlencoded
    data = urlencode({
        'grant_type': 'refresh_token',
        'refresh_token': refresh_token,
    })
    data = data.encode('ascii')  # data should be bytes
    resp_text = post_data(data, prepare_headers())
    print(resp_text)
    return json.loads(resp_text)

def prepare_headers(use_content_type_hdr = True):
    hdrs = {
        'Authorization': 'Basic {}'.format(base64.b64encode(
            '{}:{}'.format(quote(client_id, safe=''), quote(client_secret, safe='')).encode('ascii')).decode('ascii')),
    }
    if use_content_type_hdr:
        hdrs.update({'Content-type': 'application/x-www-form-urlencoded'})
    return hdrs

def post_data(data, headers):
    print('post_data
headers:
{}
data:
{}'.format(headers, data))
    req = urllib.request.Request(authorization_endpoint, data, headers)
    gcontext = ssl.SSLContext(ssl.PROTOCOL_TLSv1_2)  # avoid cert checking
    with urllib.request.urlopen(req, context=gcontext) as response:  # perform
        POST request and read response
        rsp = response.read()
    return rsp.decode('utf-8')

run(app, host='0.0.0.0', port=8088)

**NOTE:** In the script, you must change the values for `client_id`, `client_secret`, and Advanced Authentication server address in `authorization_endpoint` and `attributes_endpoint` (lines 10-14).
2. Main menu (main.html)

```html
<!DOCTYPE html>
<html>
<head lang="en">
  <meta charset="UTF-8">
  <title></title>
  <script type="text/javascript">
    function getHashParam(name) {
      var hash = window.location.hash;
      if (hash) {
        if (name = (new RegExp('#&' + encodeURIComponent(name) + '=' + '[^&]*')).exec(hash))
          return decodeURIComponent(name[1]);
      }
    }
    function showResult() {
      if (window.location.hash) {
        document.getElementById('result').innerHTML = '<table border="1">
          <tr><td>access_token</td><td>' + getHashParam('access_token') + '</td></tr>
          <tr><td>token_type</td><td>' + getHashParam('token_type') + '</td></tr>
          <tr><td>expires_in</td><td>' + getHashParam('expires_in') + '</td></tr>
        </table>);
      } else {
        document.getElementById('result').innerHTML = 'Implicit granted token is not found';
      }
    }
  </script>
</head>
<body onload="showResult();">
<div id="result">result</div><br/>
Click <a href="/logon">here</a> to obtain an authentication token through Authorization Code Grant<br/>
Click <a href="/logon-implicit">here</a> to obtain an authentication token through Implicit Grant (the token will be received in hash part of THIS page)<br/>
Click <a href="/logon-creds">here</a> to obtain an authentication token through Resource Owner Password Credentials Grant<br/>
</body>
</html>
```

3. Token information (token.html)
Token<br/>
<table border="1">
  % for k, v in items:
    <tr>
      <td>{{k}}</td>
      <td>{{v}}</td>
    </tr>
  % end
</table>
<br/>
<a href="/getattr">Get attributes</a><br/>
<a href="/refresh?refresh_token={{refresh_token}}">Refresh token</a>

4. Attributes information (attributes.html)

Attributes<br/>
<table border="1">
  % for k, v in items:
    <tr>
      <td>{{k}}</td>
      <td>{{v}}</td>
    </tr>
  % end
</table>
<br/>
<a href="/refresh?refresh_token={{refresh_token}}">Refresh token</a>

5. Logon form for Resource Owner Password Credentials Grant mode (logonform.html)

User name: <input type="text" name="username"><br/>
Password: <input type="password" name="password"><br/>
<input type="submit">


Running the Sample Web Application

Perform the following steps to run the sample web application.

1. Run the script `python oauth2_test.py`.
   A message is displayed with the following modes:

   Authorization Code Grant
   Implicit Grant (the token will be received in hash part of THIS page)
   Resource Owner Password Credentials Grant (is not supported by default but it can be activated in AAF)

3. Select the grant based on your requirement.
   
   - **Authorization Code Grant**
     1. Ensure that Use for Owner Password Credentials is set to OFF in the Advanced settings section for the OAuth 2.0 event.
     2. Click the first link.
        The NetIQ Access page is displayed with the user name request.
     3. Specify the Username.
     4. Click Next.
     5. Authenticate using all required methods of the chain.
        The result page shows the access_token, token_type and expires_in.
        - Click Get attributes to look at the attributes.
        - Click Refresh token to refresh token. The access_token value is updated.

   - **Implicit Grant**
     1. Ensure that Use for Owner Password Credentials is set to OFF in the Advanced settings section for the OAuth 2.0 event.
     2. Click the first link.
        The NetIQ Access page is displayed with the user name request.
     3. Specify the Username.
     4. Click Next.
     5. Authenticate using all the required methods of the chain.
        The result page shows the access_token, token_type and expires_in.

   - **Resource Owner Password Credentials Grant**
     1. Open Advanced settings for the OAuth 2.0 event.
     2. Set Use for Owner Password Credentials to ON.
     3. Click the third link.
        A request for Username and Password is displayed.
     4. Specify the username and password, then click Submit.
        The result page displays the access_token, token_type, and expires_in.

OAuth 2.0 Attributes

The following table displays the OAuth 2.0 attributes for a test user from the Active Directory.
The following table displays the OAuth 2.0 attributes for a local user.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>user_name</td>
<td>pjones</td>
</tr>
<tr>
<td>repository_name</td>
<td>TESTCOMPANY</td>
</tr>
<tr>
<td>naafUserSID</td>
<td>S-1-5-21-3320677580-2179873152-1514081409-1103</td>
</tr>
<tr>
<td>naafUserDN</td>
<td>CN=Paul Jones, CN=Users, DC=testcompany, DC=local</td>
</tr>
<tr>
<td>naafUserCN</td>
<td>Paul Jones</td>
</tr>
<tr>
<td>naafUserUPN</td>
<td><a href="mailto:pjones@testcompany.local">pjones@testcompany.local</a></td>
</tr>
<tr>
<td>naafUsernameNetBIOS</td>
<td>TESTCOMPANY\pjones</td>
</tr>
<tr>
<td>client</td>
<td>id-0TRljvJe3qKwJiXvy3LbjcixfiY1Q</td>
</tr>
<tr>
<td>naafUserEmail</td>
<td><a href="mailto:pjones@testcompany.com">pjones@testcompany.com</a></td>
</tr>
</tbody>
</table>

The client attribute is a Client ID specified in the OAuth 2.0 settings.

Non Standard Endpoints

OSP provides a non-standard OAuth 2.0 endpoint for signing additional data that can be passed during the grant request. The URL of the sign endpoint is: https://<serverip>/osp/a/TOP/auth/oauth2/sign.

The sign endpoint helps to create a signed and encrypted data packet that can be used to supply data to other endpoints. For more information, see the Sign class documentation.

The only endpoint with which the signed data is currently used is the grant endpoint when it is used with the authorization code grant and implicit grant types.

The signed data can be used to supply one or both of the following:

- **Username**: Supplying the username for a client application is useful when you already know the username. For example, Advanced Authentication uses OSP for authentication after Advanced Authentication has obtained the username.

- **Advanced Authentication chain**: An Advanced Authentication server (5.6 or later) can be used to supply one or more additional authentication factors by authenticating with Advanced Authentication OAuth 2.0 for a user who is already authenticated. The username and name of the desired authentication chain containing the factor(s) is supplied.

You must be able to resolve username in an Advanced Authentication repository and you must configure the chain in the Advanced Authentication event for the OAuth 2.0 client used.
Submitting the Data

The sign endpoint is used by submitting a string value to the endpoint. The output is returned in a JSON structure. The output can be used with the grant endpoint with the parameters attribute.

You can accomplish OAuth 2.0 client authentication with HTTP Basic or Bearer authorization header value.

Request parameters

- **data** (required): The data to be signed and encrypted.

  The following JSON request object code is an example to sign an endpoint.

  ```
  {
      "username": "< username >",
      "LoginParameters": {
          "internal.osp.oidp.aa.chain-name": "< chain name >"
      }
  }
  ```

  where username is name of the user trying to authenticate and chain name is name of the chain configured in the Advanced Authentication server.

- **ttl** (optional): The time-to-live period of the result data in milliseconds. If no value is supplied, then the default value of 30 seconds is used.

HTTP status codes

The following table describes the HTTP status codes.

<table>
<thead>
<tr>
<th>HTTP Status Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>The operation was successful.</td>
</tr>
<tr>
<td>400</td>
<td>The operation was unsuccessful. Additional error information may be found in the response content.</td>
</tr>
<tr>
<td>401</td>
<td>Client authentication missing or invalid.</td>
</tr>
<tr>
<td>500</td>
<td>A server error occurred.</td>
</tr>
</tbody>
</table>

The cause of the error can be determined from the additional error information found in the response content.

Response content

The response to a successful request is a serialized JSON object (XML is not currently supported).

The **data** field is the signed and encrypted data to be used with another endpoint. The **exp** field is the expiration time of the data as defined by RFC 7519. For more information, see An example (https://tools.ietf.org/html/rfc7519#section-4.1.4).

The following sample code in javascript is an example of the response content.

```javascript
{
  "data": "_TXNCmy8ocXUg3Hg7u1TmRRJ3-2JQHcv3XggLbzhX216TcM-11sfY1VatE6K1hP1.e11JXX3Gj5ULFpoo03ig-4vczT2UtrAzbV4poyN592s~",
  "exp": 1488210079
}
```
NOTE: The web authentication does not query the LDAP directly for users. Web authentication routes the request to the Advanced Authentication server internally. Therefore, if the Advanced Authentication server can match the inbound username with an appropriate attribute in the LDAP server, it would be same as what Advanced Authentication provides.
The Advanced Authentication server provides a built-in RADIUS server that can authenticate any RADIUS client using one of the chains configured for the event.

**IMPORTANT**

- The built-in RADIUS server supports only the PAP method.
- The RADIUS server supports the following authentication methods: Email OTP, Emergency Password, LDAP Password, OATH OTP, Password, RADIUS Client, Security Questions, Smartphone, SMS OTP, Voice OTP, and Voice methods.
- By design, Advanced Authentication does not support the single-factor authentication with a Smartphone, Email OTP, SMS OTP, Security Questions, Voice OTP, and Voice method for RADIUS. These methods cannot be the first or single method in a chain. It is recommended to use it in a two-factor chain with the LDAP Password method.

To configure pre-defined RADIUS Server event, perform the following steps:

1. Click Events.
2. Click Edit next to the RADIUS Server event.
3. Ensure that Is enabled is set to ON.
4. Select the chains that you want to assign to the event.
5. Select RADIUS from Endpoint whitelist.
6. Click Add to add and assign a RADIUS Client to the event:
   - 6a Specify the IP address of the RADIUS Client in IP Address.
   - 6b Specify the RADIUS Client name in Name.
   - 6c Specify the RADIUS Client secret and confirm the secret.
   - 6d Ensure that the RADIUS Client is set to ON.
   - 6e Click next to the RADIUS Client.
   - 6f Add more RADIUS Clients if required.
7. Set Return user groups to ON to enable the RADIUS server to return all the groups of a user in the filter-id attribute in an authentication response to the RADIUS Client.
   - By default the option is set to OFF and the RADIUS server does not return the filter-id attribute in the authentication response.
   - 7a Specify the preferred user groups in User groups white list to allow the RADIUS server to return only the specified groups of a user in the filter-id attribute to the RADIUS Client.
   - If you set the Return user groups to ON and the User groups white list is empty, all the groups of a user are returned in the filter-id attribute.
   - 7b You can specify any attribute you want to return instead of the Filter-ID attribute in Groups attribute. For example, you can specify the class attribute in Groups attribute and the class attribute will be returned instead of the Filter-ID attribute. By default, the Filter-ID attribute is returned in an authentication response to the RADIUS Client.
NOTE: It is recommended to enable the Return user groups option and specify the preferred user groups because in large environments a user can be part of many groups and as a result, the list of all groups that are returned by the RADIUS server can be large. The size of RADIUS response exceeds the maximum size of RADIUS packet.

8 (Optional) Specify NAS ID while adding custom RADIUS server event. You must use the same NAS ID on the configured RADIUS clients to associate them with the custom RADIUS server event.

9 Set Bypass user lockout in repository to ON, if you want to allow repository locked-out users to be authenticated on the Advanced Authentication. By default, Bypass user lockout in repository is set to OFF and users locked on repository is not allowed to authenticate.

10 Click Save.

IMPORTANT: If you use more than one chain with the RADIUS server, follow one of the following ways:

1. Each chain assigned to the RADIUS event may be assigned to a different LDAP group. For example, LDAP Password+Smartphone chain is assigned to a Smartphone users group, LDAP Password+HOTP chain is assigned to a HOTP users group. If a RADIUS user is a member of both groups, the top group is used.

2. By default, the top chain specified in the RADIUS Server event in which all the methods are enrolled is used. But, you can authenticate with the RADIUS authentication using another chain from the list when specifying <username>&<chain shortname> in username. For example, pjones&sms. Ensure that you have specified the short names for chains. Some RADIUS clients such as FortiGate do not support this option.

NOTE: If you use the LDAP Password+Smartphone chain, you can use an offline authentication by specifying the password in the format <LDAP Password>&<Smartphone OTP>. For example, Q1w2e3r4&512385. This option is supported for LDAP Password+OATH TOTP, Password+Smartphone, Password+OATH TOTP, Password+OATH HOTP.

When you want to add multiple RADIUS clients, you can add them to the predefined RADIUS Server event. But all the RADIUS clients will use the same authentication chain(s). If you want to configure specific authentication chain(s) for different RADIUS clients, then you must create a custom RADIUS event. While adding the custom RADIUS event ensure to specify NAS ID that is essential to associate clients with the custom RADIUS event.

For more information about the custom RADIUS event, see Creating a RADIUS Event.

NOTE: If the RADIUS log files are overflowed of records with the error Discarding duplicate request from client, you can increase the timeout on the RADIUS Client. The optimal timeout value needs to be determined by experimenting. It must not exceed 60 seconds.

Customizing Prompt Messages For RADIUS Event

You can customize prompt messages of the authentication methods that are configured for the RADIUS event. The customized prompt messages are displayed when a user initiates authentication to RADIUS event using the configured methods.

For more information about customizing prompt message for RADIUS event, see Customizing Prompt Messages of the Authentication Methods for RADIUS Event.
Challenge-Response Authentication

If you have configured a multi-factor chain such as LDAP Password & SMS OTP or any other combination chain, some users (during the authentication) might not be able to specify the `<Password>&<OTP>` in a single line (because of the Password length limit in RADIUS). In this case, you can configure the existing RADIUS Client by performing the following steps:

1. Specify an LDAP password in **Password** and send the authentication request.

   Advanced Authentication server returns the access-challenge response with **State=<some value>** (example: State=WKKNNLTTBxP6QYfiZIipvscyt7RYrYsGag4h8s0Rh8R) and **Reply-Message=SMS OTP**. You will receive an SMS with a one-time password on the registered mobile.

2. Specify the OTP in **Password** and add an additional RADIUS attribute with **State=<value>** where, value is the value that is obtained in step 1.

3. Send the authentication request.

Using RADIUS in Multitenancy Mode

The following are the examples of integration with a RADIUS Server:

- Configuring Integration with Barracuda
- Configuring Integration with Citrix NetScaler
- Configuring Integration with Dell SonicWall SRA EX-Virtual Appliance
- Configuring Integration with FortiGate
- Configuring Integration with OpenVPN
SAML 2.0

SAML 2.0 is an XML-based protocol that uses security tokens containing assertions. The assertions are used for sending the information about a subject (an entity that is often a human user) from a SAML authority (Identity Provider) to a SAML consumer (Service Provider).

This chapter contains the following section:

- “Integrating Advanced Authentication with SAML 2.0” on page 151

Integrating Advanced Authentication with SAML 2.0

To integrate Advanced Authentication with the third-party solutions using SAML 2.0, perform the following steps:

1. Click Events > Add.
2. Specify a name for the new event.
3. Change the Event type to SAML2.
4. Select the required chains for the event.
5. Copy and paste your Service Provider’s SAML 2.0 metadata to SP SAML 2.0 metadata. OR
   - Click Browse and select a Service Provider’s SAML 2.0 metadata XML file to upload it.
6. Click Policies > Web Authentication.
7. (Conditional) Specify the Identity Provider’s URL in Identity provider URL.

   **NOTE:** To use multiple Advanced Authentication servers with SAML 2.0, you must do the following:
   1. Configure an external.
   2. Specify the address in Identity provider URL instead of specifying an address of a single Advanced Authentication server.

8. Click Download IdP SAML 2.0 Metadata to open a metadata. The metadata opens in a new browser page.
9. Save the metadata (XML text) from the browser.
10. (Conditional) Use the downloaded metadata file in your Service Provider.
11. (Conditional) Use the Identity Provider certificate in your Service Provider.
12 Change used hash to SHA-1 in your Service Provider, if the option is presented.

13 Set the **Send E-Mail as NameID (suitable for G-Suite)** option to **ON** for integrating with the G-suite.

14 Set the **Send SAMAccount as NameID** option to **ON** to send SAMAccountName in the NameID attribute as a SAML response from the Advanced Authentication server.

**WARNING:** You can set **Send SAMAccount as NameID** to **ON** only when the **Send E-Mail as NameID (suitable for G-Suite)** option is turned **OFF**.

15 Set **Bypass user lockout in repository** to **ON**, if you want to allow users who are locked on repository to authenticate on the Advanced Authentication. By default, Bypass user lockout in repository is set to **OFF** and users who are locked on repository are not allowed to authenticate.

The following are the examples of integration with SAML 2.0.

- Configuring Integration with Salesforce
- Configuring Integration with ADFS

**Requesting Advanced Authentication Methods and Chains Through a SAML AuthnRequest**

SAML 2.0 provides a mechanism to request an authentication class reference. For more information, see the SAML 2.0 Core specification (https://docs.oasis-open.org/security/saml/v2.0/saml-core-2.0-os.pdf) in section 3.3.2.2.1.

The Service Provider sends the following code in the `<AuthnRequest>`:

```xml
<samlp:RequestedAuthnContext>
</samlp:RequestedAuthnContext>
```

SAML 2.0 defines a bunch of URNs that corresponds to authentication classes. For more information, see SAML 2.0 Authentication Context (http://docs.oasis-open.org/security/saml/v2.0/saml-authn-context-2.0-os.pdf).
Some of the authentication class types of Advanced Authentication match the SAML 2.0 references. The Advanced Authentication auth class types are defined in an enum named AuthClassType.

In this XML example, the SAML class reference URN maps to the Advanced Authentication’s AuthClassType.MOBILE_ONE_FACTOR_CONTRACT. The Advanced Authentication value is mapped to NaafAuthMethod.SMARTPHONE (or NaafAuthMethod.SWISSCOM).

The code in NaafEventContractExecutable.filterChains selects from the available chains any chain that contains one of its methods (in this example) SMARTPHONE or SWISSCOM. (The map from Advanced Authentication methods to OSP auth class type is NaafContractExecutable.METHOD_TO_TYPE_MAP.)

In this example, after the user is identified, if there is a chain available with the Smartphone or Swisscom methods, then the authentication proceeds. If not, the authentication fails and Advanced Authentication returns a no requested authentication context status to the Service Provider.

An optional Comparison attribute can be set on the <RequestedAuthnContext>. This attribute is defined in the SAML 2.0 Core specification (https://docs.oasis-open.org/security/saml/v2.0/saml-core-2.0-os.pdf) in section 3.3.2.2.1.

In addition to requesting the Advanced Authentication methods using the SAML 2.0-defined URNs, Advanced Authentication also has a special contract parameters class reference URN. The URN is: urn:uuid:519a6c73-f092-43d3-ab11-8d789ebc2f79.

The contract parameters are added through the URN q-component. The URN syntax is defined at RFC 8141 (https://tools.ietf.org/html/rfc8141).

The <NaafEvent> contract executable contains attributes named allowClientChainSelection and allowClientEventSelection. These attributes allow the authentication chain and the authentication event to be selected through a contract parameter from the client, which in this example, is the SAML Service Provider. In the Advanced Authentication authcfg.xml, the default value of allowClientEventSelection is false and allowClientChainSelection is true.

For example, ISM is an event name with the following chains: LDAP+Smartphone, LDAP+SMS_OTP, LDAP+TOTP, LDAP+SecQuest, LDAP+U2F, and LDAP+Voice.

If the <NaafEvent> contract executable is configured with the ISM event, then the following code will request the LDAP+SMS_OTP chain.

```xml
<samlp:RequestedAuthnContext>
  <saml:AuthnContextClassRef>urn:uuid:519a6c73-f092-43d3-ab11-8d789ebc2f79?=internal.osp.oidp.aa.chain-name=LDAP%2BSMS_OTP</saml:AuthnContextClassRef>
</samlp:RequestedAuthnContext>
```

The plus sign '+' is encoded as '%2B'. Advanced Authentication considers that the q-component, which starts with ‘?’, is in the x-www-form-urlencoded format and ‘+’ is a reserved character for this syntax.

The two contract parameters that are defined in the Advanced Authentication class CFGNaafEvent are:

- internal.osp.oidp.aa.chain-name
- internal.osp.oidp.aa.event-name
Examples of Integrations

This chapter contains the following examples of third-party integrations.

- “Configuring Integration with Barracuda” on page 155
- “Configuring Integration with Citrix NetScaler” on page 157
- “Configuring Integration with Dell SonicWall SRA EX-Virtual Appliance” on page 159
- “Configuring Integration with FortiGate” on page 160
- “Configuring Integration with OpenVPN” on page 162
- “Configuring Integration with Palo Alto GlobalProtect Gateway” on page 163
- “Configuring Integration with Salesforce” on page 164
- “Configuring Integration with ADFS” on page 167
- “Configuring Integration with Google G Suite” on page 170
- “Configuring Integration with Office 365” on page 172
- “Configuring Integration with Sentinel” on page 175

Configuring Integration with Barracuda

This section provides the configuration information on integrating Advanced Authentication with Barracuda SSL VPN virtual appliance. This integration secures the Barracuda SSL VPN connection.

The following diagram represents integration of Advanced Authentication with Barracuda SSL VPN.
To configure the Advanced Authentication integration with Barracuda SSL VPN, perform the following configuration tasks:

- “Configuring the Advanced Authentication RADIUS server:" on page 156
- “Configuring the Barracuda SSL VPN Appliance:" on page 156
- “Authenticating on Barracuda SSL VPN Using Advanced Authentication” on page 157

**Configuring the Advanced Authentication RADIUS server:**

1. Open the Advanced Authentication Administration portal.
2. Click **Events > RADIUS Server**.
3. Set **Is enabled** to ON.
4. Move one or more chains from **Available** to **Used** list. Ensure that the chains are assigned to the appropriate group of users in **Roles & Groups** of the **Chains** section.
5. Click **Client > Add**.
6. Specify an **IP address** of the Barracuda SSL VPN appliance.
7. Specify a secret and confirm it.
8. Set **Enabled** to ON.
9. Click **Save in Client**.
10. Click **Save** in **Events**.

**Configuring the Barracuda SSL VPN Appliance:**

1. Sign-in to the Barracuda SSL VPN Configuration portal as **ssladmin**.
2. Click **Access Control > Configuration**.
3. Scroll down to **RADIUS**.
4. Specify an Advanced Authentication appliance IP address in **RADIUS Server**.
5. Specify a shared secret in **Shared Secret**.
6. Set **Authentication Method** to **PAP**.
7. Set **Reject Challenge** to **No** to allow challenge response.
8. Click **Save Changes**.
9. Click **Access Control > User Databases**.
10. Create a user database using the same storage as you are using for Advanced Authentication.
11. Click **Access Control > Authentication Schemes**.
12. Click **Edit** for the **Password** scheme for the user database.
13. Move **RADIUS** from **Available modules** to **Selected modules**.
14. Remove the **Password** module from the **Selected modules**.
15. Apply the changes.
Authenticating on Barracuda SSL VPN Using Advanced Authentication

1. Specify the user’s credentials.
2. Click More and select the configured user database (if the database is not selected by default).
3. Click Log In and approve the authentication on the user’s smartphone.

**NOTE:** Advanced Authentication can be configured with the other authentication chains.

---

Configuring Integration with Citrix NetScaler

This section provides the configuration information on integrating Advanced Authentication with Citrix NetScaler VPX. This integration secures the Citrix NetScaler VPX connection.

The following diagram represents Advanced Authentication in Citrix NetScaler.

To configure the Advanced Authentication integration with Citrix NetScaler VPX, perform the following configuration tasks:

- "Configuring the Advanced Authentication RADIUS Server" on page 157
- "Configuring the Citrix NetScaler Appliance" on page 158
- "Authenticating on the Citrix NetScaler Using Advanced Authentication" on page 158

Ensure that the following requirements are met:

- Citrix NetScaler VPX (version NS11.0 has been used to prepare these instructions) is installed.
- Advanced Authentication 5 appliance is installed.

### Configuring the Advanced Authentication RADIUS Server

1. Open the Advanced Authentication Administration portal.
2. Click Events > RADIUS Server.
3 Set Is enabled to ON.
4 Move one or more chains from Available to Used list. Ensure that the chains are assigned to the appropriate group of users in Roles & Groups of the Chains section.
5 Click Client > Add.
6 Specify an IP address of the Citrix NetScaler appliance.
7 Specify a secret and confirm it.
8 Set Enabled to ON.
9 Click Save in Client.
10 Click Save in Events.

Configuring the Citrix NetScaler Appliance

1 Sign-in to the Citrix NetScaler configuration portal as nsroot.
2 Click Configuration > Authentication > Dashboard.
3 Click Add.
4 Select RADIUS for Choose Server Type.
5 Specify Name of the Advanced Authentication server, IP Address, Secret Key, and Confirm Secret Key.
6 Change Time-out (seconds) to 120-180 seconds if you are using the Smartphone, SMS, Email or Voice methods.
7 Click More and ensure that PAP is selected in Password Encoding.
8 Click Create.
   If the connection to the RADIUS server is valid, the Up status is displayed.
9 Click Configuration > System > Authentication > RADIUS > Policy.
10 Click Add.
11 Specify Name of the Authentication RADIUS Policy.
12 Select the created RADIUS server from Server and select ns_true from the Saved Policy Expressions list.
13 Click Create.
14 Select the created policy and click Global Bindings.
15 Click Select Policy.
16 Select the created policy.
17 Click Bind.
18 Click Done.
   A check mark is displayed in the Globally Bound column.

Authenticating on the Citrix NetScaler Using Advanced Authentication

1 Specify the user’s credentials then click Login.
2 Accept the authentication on your smartphone.
NOTE: Advanced Authentication can be configured with other authentication chains.

Configuring Integration with Dell SonicWall SRA EX-Virtual Appliance

This section provides the configuration information on integrating Advanced Authentication with Dell SonicWall SRA EX-virtual appliance. This integration secures the Dell SonicWall SRA connection.

The following diagram represents Advanced Authentication in Dell SonicWall.

To configure the Advanced Authentication integration with Dell SonicWall SRA, perform the following configuration tasks:

- “Configuring the Advanced Authentication RADIUS Server” on page 159
- “Configuring the Dell SonicWall SRA Appliance” on page 160
- “Authenticating on Dell SonicWall Workspace Using Advanced Authentication” on page 160

Ensure that the following requirements are met:

- Dell SonicWall SRA EX-Virtual appliance v11.2.0-258 is installed.
- Advanced Authentication v5 appliance is installed.

Configuring the Advanced Authentication RADIUS Server

1. Open the Advanced Authentication Administration portal.
2. Click Events > RADIUS Server.
3. Set Is enabled to ON.
4. Move one or more chains from Available to Used list. Ensure that the chains are assigned to the appropriate group of users in Roles & Groups of the Chains section.
5. Click Client > Add.
7. Specify a secret and confirm it.
8 Set Enabled to ON.
9 Click Save in Client.
10 Click Save in Events.

**Configuring the Dell SonicWall SRA Appliance**

1. Sign-in to the Dell SonicWall SRA Management console as admin.
2. Click User Access > Realms.
3. Click New realm.
4. Create a New Authentication Server and set the RADIUS authentication directory.
5. Set RADIUS Server and Shared key.
6. Save and apply the configuration.
7. Click User Access > Realms.
   - Review the realm diagram.

**Authenticating on Dell SonicWall Workspace Using Advanced Authentication**

1. Open a browser and navigate to the workplace.
2. Specify your username and LDAP password.
3. Specify the SMS OTP and click OK.

**Configuring Integration with FortiGate**

This section provides the configuration information on integrating Advanced Authentication with FortiGate. This integration secures the FortiGate connection.

The following diagram represents Advanced Authentication in FortiGate.
To configure the Advanced Authentication integration with FortiGate perform the following configuration tasks:

- "Configuring the Advanced Authentication RADIUS Server" on page 161
- "Configuring the FortiGate Appliance" on page 161
- "Authenticating on FortiGate Using Advanced Authentication" on page 161

Ensure that the following requirements are met:

- Fortinet virtual appliance v5 (Firmware version 5.2.5, build 8542 has been used to prepare these instructions) is installed.
- Advanced Authentication v5 appliance is installed.

**Configuring the Advanced Authentication RADIUS Server**

1. Open the Advanced Authentication Administration portal.
2. Click Events > RADIUS Server.
3. Set Is enabled to ON.
4. Move one or more chains from Available to Used list. Ensure that the chains are assigned to the appropriate group of users in Roles & Groups of the Chains section.
5. Click Client > Add.
6. Specify an IP address of the FortiGate appliance.
7. Specify a secret and confirm it.
8. Set Enabled to ON.
9. Click Save in Client.
10. Click Save in Events.

**Configuring the FortiGate Appliance**

1. Sign-in to FortiGate configuration portal as admin.
2. Check which Virtual Domain is bound to the network interface.
3. Open the RADIUS Server configuration for an appropriate Virtual Domain and setup the required settings.
4. Click Test Connectivity and specify the credentials of Advanced Authentication administrator to test the connection.
5. Create a user group and bind it to a remote authentication server.
6. Create user and place in the created group.

**Authenticating on FortiGate Using Advanced Authentication**

1. Specify the user’s credentials and click Login.
2. Specify the OTP and click Login.

**NOTE:** The Token Code field has a limitation of 16 digits. Therefore, you may face issues when using the YubiKey tokens with 18-20 digits code.
Configuring Integration with OpenVPN

This section provides the configuration information on integrating Advanced Authentication with OpenVPN virtual appliance. This integration secures the OpenVPN connection.

The following diagram represents Advanced Authentication in OpenVPN.

To configure the Advanced Authentication integration with OpenVPN perform the following configuration tasks:

- "Configuring the Advanced Authentication RADIUS Server" on page 162
- "Configuring the OpenVPN Appliance" on page 163

Ensure that the following requirements are met:

- OpenVPN v2 appliance (version 2.0.10 was used to prepare these instructions) is installed.
- Advanced Authentication v5 appliance with a configured repository is installed.

Configuring the Advanced Authentication RADIUS Server

1. Open the Advanced Authentication Administration portal.
2. Click Events > RADIUS Server.
3. Set Is enabled to ON.
4. Move one or more chains from Available to Used list. Ensure that the chains are assigned to the appropriate group of users in Roles & Groups of the Chains section.
5. Click Client > Add.
6. Specify an IP address of the OpenVPN appliance.
7. Specify a secret and confirm it.
8. Set Enabled to ON.
9. Click Save in Client.
10. Click Save in Events.
Configuring the OpenVPN Appliance

1. Open the OpenVPN Access Server site.
2. Click Authentication > RADIUS.
3. Enable the RADIUS authentication.
4. Select PAP authentication method.
5. Add an IP address of the Advanced Authentication v5 appliance and specify the secret.

You must specify the <repository name>\<username> or only <username>, if you have set the following configurations:

- You have selected a chain from the Used section in the RADIUS Server settings for connecting to OpenVPN.
- You have set the default repository name in Policies > Login options of the Advanced Authentication v5 appliance.

You must specify a Short name of the chain in the username after the <username> and space (you can specify the Short name in the Chains section of the Advanced Authentication v5 appliance), if you have set the following configurations:

- You have selected multiple chains from the Used section for connecting to OpenVPN.

**NOTE:** For some authentication methods, the correct time must be configured on the OpenVPN appliance. You can sync the time of the OpenVPN appliance using the following commands:

```
/etc/init.d/ntp stop
/usr/sbin/ntpdate pool.ntp.org
```

User Account Locks After Three Successful authentications with SMS AP to OpenVPN

**Issue:** While authenticating with the SMS method to connect to OpenVPN, after three successful authentications the user account is locked by OpenVPN.

**Workaround:** OpenVPN assumes each attempt of the challenge response (request of additional data in chain) as an error.

To resolve the issue, you must change the number of failures that can be accepted. For more information, see Authentication failure lockout policy.

Configuring Integration with Palo Alto GlobalProtect Gateway

This section provides the configuration information on integrating Advanced Authentication with Palo Alto GlobalProtect Gateway. This integration secures the Palo Alto GlobalProtect Gateway connection.

**NOTE:** This configuration has been tested with PAN-OS 6.1.5 to 7.1.x and GlobalProtect 2.1x.
To configure the Advanced Authentication integration with Palo Alto GlobalProtect Gateway, perform the following configuration tasks:

- “Adding the RADIUS Server” on page 164
- “Adding an Authentication Profile” on page 164
- “Configuring GlobalProtect Gateway” on page 164

### Adding the RADIUS Server

1. Log in to the Palo Alto administrative interface.
2. Click **Device > Server Profiles > RADIUS**.
3. Click **Add** to add a new RADIUS server profile.
4. Specify **NetIQ RADIUS** in **Name**.
5. Specify 30 in **Timeout**.
6. In the **Servers** section, click **Add** to add a RADIUS server and specify the following information:
   - **Profile Name**
   - Set **Timeout and Retries in Server Settings**
   - Details in the **Servers** section
7. Click **Add** and configure a connection to the RADIUS server built-in to the Advanced Authentication server.
8. Click **OK**.

### Adding an Authentication Profile

1. Click **Device > Authentication Profile**.
2. Click **New** to add a new authentication profile.
3. Specify the Authentication Profile details such as the server type and user domain.

### Configuring GlobalProtect Gateway

1. Click **Network > GlobalProtect > Gateways**.
2. Click on your configured GlobalProtect Gateway to open the properties window.
3. In the **Authentication** section of the **GlobalProtect Gateway General properties** tab, select the **NetIQ authentication profile** created in **Add an Authentication Profile** from the list.
4. Click **OK** to save the GlobalProtect Gateway settings.

### Configuring Integration with Salesforce

This section provides the configuration information on integrating Advanced Authentication with Salesforce. This integration secures the Salesforce connection.

The following diagram represents Advanced Authentication in Salesforce.
To configure the Advanced Authentication integration with Salesforce, perform the following configuration tasks:

- “Configuring the Salesforce Domain Name” on page 165
- “Configuring the SAML Provider” on page 165
- “Configuring the Advanced Authentication SAML 2.0 Event” on page 166
- “Configuring to Authenticate with Salesforce with SAML 2.0” on page 167

### Configuring the Salesforce Domain Name

1. Login to your Salesforce account.
2. Create a domain. If the domain is not created, then perform the following tasks:
   1. Click Gear and select Setup Home in the Lightning Experience interface.
   2. Scroll down the setup toolbar and navigate to Company Settings.
   3. Click My Domain.
   4. Specify your domain name and click Save.

The domain is activated. Use your domain name to open Salesforce. For example, https://CompanyName.my.salesforce.com/. SAML provider requires the domain name.

### Configuring the SAML Provider

2. Create a text file and add the following Identity Provider certificate to the file.

```
-----BEGIN CERTIFICATE-----
MIIDkzCCAnugAwIBAgIESsmdMzANBgkqhkiG9w0BAQsFADB6MRAwDgYDVQQGEwJQYXNzGDgs
MDQwHhcNMTI2MDEwNTIyMTE3WjEwMBQxMDowMDAyMRswGQYDVQQDEwJQYXNzMS0wIwYDVQQD
EwJQYXNzMB4XDTMxMjIyOTc5My1JMTQ3MDEyMDAxNFoXDTMxMjIyOTc5My1JMTQ3MDEyMDAx
NjMxMjAxMTEyNjIzMFoX方才m87lNyAO8CtN5jlLe3CupLAAbUWRNY6av7lPortableDocumentFormat
-----END CERTIFICATE-----
```
In **Single Sign-On Settings**, click **New** and specify the following details:

1. **Name**: Advanced Authentication.
2. **API Name**: AAF.
3. **Issuer**: https://AdvancedAuthenticationServerAddress/osp/a/TOF/auth/saml2/metadata, where you must replace AdvancedAuthenticationServerAddress with the domain name or IP address of your Advanced Authentication server.
5. Click **Browse** to open the Identity Provider certificate.
6. **SAML Identity Type**: Select Assertion contains the Federation ID from the User object.
7. **SAML Identity Location**: Select Identity is in an Attribute element.
8. **Attribute Name**: upn.
9. **Service Provider Initiated Request Binding**: Select HTTP Redirect.
11. Select **User Provisioning Enabled**.
12. Click **Save**.

4 Click **Edit** for Federated Single Sign-On Using SAML.

5 Select **SAML Enabled**.

6 Click **Save**.

7 Click **Settings > Users**.

8 Click **Edit** for the required Salesforce users by adding **Federation ID** for the user accounts. The Federation ID corresponds to userPrincipalName attribute in Active Directory. For example, pjones@company.com.

---

**NOTE:** The name that you specify in **Federation ID** is case sensitive. The following error may occur, if you ignore the case:

We can't log you in. Check for an invalid assertion in the SAML Assertion Validator (available in Single-Sign On Settings) or check the login history for failed logins.

9 Click your profile icon and click **Switch to Salesforce Classic**.

This mode is required to tune the domain options.

10 Click **Setup Administrator > Domain Management > My Domain > Edit** to access the Authentication Configuration screen.

11 Select **Login Page** and osp options.

12 Click **Save**.

---

**Configuring the Advanced Authentication SAML 2.0 Event**

1 Click **username > Switch to Lightning Experience**.

2 Click **Gear** and select **Setup Home**.

3 Navigate to **Identity > Single Sign-On Settings**.

4 Click the created configuration (not for Edit).

5 Click **Download Metadata**.
6 Open the Advanced Authentication Administration portal.
7 Click Events > Add to add a new event.
8 Create an event with the following parameters.
   - Name: Salesforce
   - Chains: select the required chains.
   - Click Browse to Upload SP SAML 2.0 metadata file. Open the Salesforce metadata file and click Save.

**Configuring to Authenticate on Salesforce with SAML 2.0**

1 Click Policies > Web Authentication.
2 Set Identity provider URL to https://AdvancedAuthenticationServerAddress/ and replace AdvancedAuthenticationServerAddress with domain name or IP address of your Advanced Authentication server.

**NOTE:** To use multiple Advanced Authentication servers with SAML 2.0, you must do the following:
1. Configure an external .
2. Specify the address with port number in Identity provider URL instead of specifying an address of a single Advanced Authentication server.

**IMPORTANT:** You must use the server name or IP address specified in the Issuer field of Salesforce.
3 Open the URL https://CompanyName.my.salesforce.com/ and click Advanced Authentication to check the SAML 2.0 authentication.

**Configuring Integration with ADFS**

This section provides the configuration information on integrating Advanced Authentication with ADFS (Active Directory Federation Services). This integration secures the ADFS connection.

The following diagram represents Advanced Authentication and ADFS integration using SAML.
To configure the Advanced Authentication integration with ADFS using SAML 2.0 perform the following configuration tasks:

**NOTE:** These instructions are valid only for ADFS 3 and 4.

- “Configuring the Advanced Authentication SAML 2.0 Event” on page 168
- “Making the Corresponding Changes in ADFS” on page 169

### Configuring the Advanced Authentication SAML 2.0 Event

1. Open the Advanced Authentication Administration portal.
2. Click **Events > Add** to add a new event.
3. Create an event with the following parameters:
   - **Name:** ADFS_SAML.
   - **Event Type:** SAML 2.
   - **Chains:** Select the required chains.
   - **Paste the content of the file** `https://<adfs_hostname>/FederationMetadata/2007-06/FederationMetadata.xml` **to SP SAML 2.0 meta data.**
     
   Or
   - Click **Browse** and upload the saved XML file.
   - Click **Save.**
NOTE: Verify that you can access the file in your browser. If the file is not displayed, you have an issue on ADFS that you must resolve.

4 Click Policies > Web Authentication.

5 Set Identity provider URL to https://AdvancedAuthenticationServerAddress/ and replace AdvancedAuthenticationServerAddress with domain name or IP address of your Advanced Authentication server.

NOTE: To use multiple Advanced Authentication servers with SAML 2.0, you must do the following:
   1. Configure an external
   2. Specify the address in Identity provider URL instead of specifying an address of a single Advanced Authentication server.

6 Click Download IdP SAML 2.0 Metadata.
   You must open the file as an XML file.

NOTE: If {"Fault":{"...` is displayed, you must verify the configuration.

Making the Corresponding Changes in ADFS

1 Open the ADFS management console.
2 Expand Trust Relationships.
3 Click Add Claims Provider trust.
   It may not work for self-signed certificate. You can copy metadata from OSP URL to an XML file and provide the file name.
5 Specify the Display name.
6 Select Open the Edit Claim Rules dialog for this claims provider when the wizard closes.
7 In Edit Claims Rules, click Add Rule.
8 Select Send Claims Using a Custom Rule.
9 Click Next.
10 Specify Claim rule name.
11 Paste Custom rule and click Finish.
   c:[Type == "upn"]
 => issue(Type = "http://schemas.xmlsoap.org/ws/2005/05/identity/claims/upn",
  Issuer = c.Issuer, OriginalIssuer = c.OriginalIssuer, Value = c.Value,
  ValueType = c.ValueType);
12 In ADFS snap-in, double click on the provider name.
13 Click Advanced.
14 Move the hash algorithm from SHA-256 to SHA1.
15 Click OK.
Configuring Integration with Google G Suite

This section provides the configuration information on integrating Advanced Authentication with Google G Suite. This integration secures the connection.

The following diagram represents Advanced Authentication in Google G Suite.

To configure the Advanced Authentication integration with Google G Suite using SAML 2.0, perform the following configuration tasks:

- “Configuring Google G Suite” on page 170
- “Configuring the Advanced Authentication Event” on page 172
- “Configuring to Authenticate on Google G-Suite with SAML 2.0” on page 172

**NOTE:** As a prerequisite, ensure that you finalize the setup of G Suite by accepting the agreement and clicking Finalize setup.

### Configuring Google G Suite

1. Login to the Google’s Administration console.
2. Open the Security section.
3. Expand Set up single sign-on (SSO).
4. Enable Setup SSO with third party identity provider.
5. Specify the following parameters:
   - **Sign-in page URL:** `https://<AdvancedAuthenticationServerAddress>/osp/a/TOP/auth/saml2/sso`. Replace `AdvancedAuthenticationServerAddress` with the domain name or IP address of your Advanced Authentication server.
   - **Sign-out page URL:** `https://<AdvancedAuthenticationServerAddress>/osp/a/TOP/auth/app/logout`.
   - **Change password URL:** `https://<AdvancedAuthenticationServerAddress>` or Self-Service Password Reset URL.
   - **Create a text file and add the Identity Provider Certificate to it.**
-----BEGIN CERTIFICATE-----
MIIDkzCCAnugAwIBAgIIESsmdMzANBgkqhkiG9w0BAQsFADB6MRAwDgYDVQGEw9vbmhtub3duMRAw
DgYDVQQIEw9vbmhtub3duMRAwDgYDVQEEwhDxRoYXNhczESMBAGA1UECwMJQXV0aGFzYXMu
Nj0WhcNMjYwNDA0MDUzNjI0WjB6MRAwDgYDVQEEwhDxRoYXNhczESMBAGA1UECwMJQXV0aGFzYX
NhMRswGQYDVQQDEkJvc3AuYXV0aGFzYXMuGbG9jYWw8HhcNMTYwH0E3MDIzOTIyNjg3WhcNMjYw
BhE3MDIzOTIyNjg3WjB6
AoIBAQcw3YlZ03ghSZPXjc/WS+cZ2/E5ooglZKeJ3p4RR6USoarjmvnPq+maRfverxiwQrJRDgS
OFRb58cert/
misqzHBVmqfMnwcFVzuukjDebWFp9vLlqRkDzI1pCy13eNmBWuXM49Z6mm8XSfIw1AoYdNp5DK00
0Yrk6FNO10nOrnI5kHGVD0bd5SpDtvXSF1Wt5c5YT9UBUpfZnEsvVPWSkbeBXF84hYW8BtdzcTEyj
dso9Ra7uLtULW0U3LTgn9zS97nLkmhetmD1I3mEAE9SAmqTRyH1FNXZ/Zofi/
BJIf4+sz86f6pBbwYM2KTxvaAbgzeSpzp1PqrZKAFgMBAAoJITAfMBGALUdDqOWBBT5L8Pa
+e6YkB1k4YELZ7+AbfdA6DANBgkqBkikG9w0BAQwFAAOCAQEAm871NyAO8CtnN5j1e3cupLAAb
WRNY6av7LPa11JRIw+uvddMyOzlvoS1IwpDDncPtxGXsa21CkNgNPbLvSxeFVUXnFfGUCtuz+bT
ucUtiQbkiDwWLmAS6KeA+EIBFOeqBiudefkAZT78DF9KV6Wdz7J7BwLZ2YPbH/FRM8SLoyAd
RbphF215we3rvefWbwX70UGNyBUTb3zUsAmB3sHbcZ1XjZj3pJyGDaN9rs60szyG1ZLEYluV
RLT2FPEfEca1Eij0R1A31Z5h3J3d1XoCeNYLoMg4522QYeKTwvQeWkeYeJBEcxdL7VF6F91zm
fz bm1A4FY5zw==
-----END CERTIFICATE-----

5e  Upload the Identity Provider Certificate.

6  Clear Use a domain specific issuer if you have one domain in G Suite or select the option if you have more than one domain in G Suite.

Ensure that you have a user account in a repository that corresponds to a user account in Google. An email address specified in the Contact information for the Google account must be the same as an address from email attribute for the corresponding account of your repository.

NOTE: You cannot use the Google administrator account with SAML.

7  Create a new text file and add the Service Provider metadata to it:

    <EntityDescriptor entityID="google.com"
    xmlns="urn:oasis:names:tc:SAML:2.0:metadata">
    <SPSSODescriptor
        protocolSupportEnumeration="urn:oasis:names:tc:SAML:2.0:protocol">
        <NameIDFormat>urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress</NameIDFormat>
        <AssertionConsumerService index="1"
            Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
            Location="https://www.google.com/a/mycompany.com" />
    </SPSSODescriptor>
</EntityDescriptor>
Replace mycompany.com in the Location URL to your primary domain from the Domains settings in Google.

**NOTE:** You must use the Service Provider metadata when one domain exists in the G Suite. If you have more than one domain in G Suite, then every Service Provider metadata for each domain must have google.com as an entityID replaced with google.com/mycompany.com, where mycompany.com is your domain name.

8 Save the text file with a .xml extension.

### Configuring the Advanced Authentication Event

1. Open the Advanced Authentication Administration portal.
2. Click Events > Add to add a new event with the following options:
   2a. Name: Google
   2b. Chains: select the required chains.
   2c. Click Browse to upload the XML file.
   2d. Set Send E-Mail as NameID (suitable for G-Suite) to ON. This is applicable for the G-Suite.
   2e. Click Save.

### Configuring to Authenticate on Google G-Suite with SAML 2.0

1. In Policies > Web Authentication, set Identity provider URL to https://AdvancedAuthenticationServerAddress/ and replace AdvancedAuthenticationServerAddress with domain name or IP address of your Advanced Authentication server.

**NOTE:** To use multiple Advanced Authentication servers with SAML 2.0, you must do the following:

1. Configure an external.
2. Specify the address in Identity provider URL instead of specifying an address of a single Advanced Authentication server.

2. Open the Google Sign in page and specify an email address of the user from Basic information of the Google account (email address of Google account).

Google redirects to the Advanced Authentication server, where the user must authenticate. After successful authentication, the Advanced Authentication server redirects the user back to Google.

### Configuring Integration with Office 365

This section provides the configuration information on integrating Advanced Authentication with Office 365. This integration secures the connection.

The following diagram represents integration of Advanced Authentication with Office 365.
To configure the integration of Advanced Authentication with Office 365, perform the following tasks:

- “Configuring Advanced Authentication SAML 2.0 Event” on page 173
- “Making the Corresponding Changes in ADFS” on page 174
- “Authenticating on Office 365” on page 174

Ensure that the following requirements are met:

- ADFS v4.0, Domain Controller, and other components must be configured to work with Microsoft Office 365.

## Configuring Advanced Authentication SAML 2.0 Event

1. Open the Advanced Authentication Administration portal.
2. Click Events > Add to add a new event.
3. Create an event with the following parameters:
   - Name: Office 365
   - Event Type: SAML 2.
   - Chains: Select the required chains.
   - Paste the content of the file https://<adfs_hostname>/FederationMetadata/2007-06/FederationMetadata.xml to SP SAML 2.0 metadata.
     Or
   - Click Browse and upload the saved XML file.
   - Click Save.

   **NOTE:** Verify that you can access the file in your browser. If the file is not displayed, you have an issue on ADFS that you must resolve.

4. Click Policies > Web Authentication.
5. Set the **External URL** to https://AdvancedAuthenticationServerAddress/ and replace AdvancedAuthenticationServerAddress with domain name or IP address of your Advanced Authentication server.

   **NOTE:** To use multiple Advanced Authentication servers with SAML 2.0, you must do the following:
   1. Configure an external .
   2. Specify the address in **External URL** instead of specifying an address of a single Advanced Authentication server.

6. Click **Download IdP SAML 2.0 Metadata**.
   You must open the file as an XML file.
NOTE: If "Fault": {...` is displayed, you must verify the configuration.

174

Click Save.

Making the Corresponding Changes in ADFS

1 Open the ADFS management console.
2 Click Claims Provider Trusts > Add Claims Provider trust.
3 Click Start in the Add Claims Provider Trust Wizard.
4 Click Import data about the claims provider from a file in the Select Data Source tab.
5 Browse the Federation metadata file.
   You can download the Federation metadata from the Advanced Authentication metadata URL:
6 Click Next.
7 Specify the Display name.
8 Click Next.
9 Select Open the Edit Claim Rules dialog for this claims provider when the wizard closes.
10 Click Close.
11 Right-click the Display name and click Edit Claim Rules.
12 Click Add Rule.
13 Select Send Claims Using a Custom Rule from Claim rule template in the Add Transform Claim Rule Wizard.
14 Click Next.
15 Specify the Claim rule name.
16 Paste the following in Custom rule:
   
   c:[Type == "netbiosName"]
   => issue(Type = "http://schemas.microsoft.com/ws/2008/06/identity/claims/windowsaccountname", Issuer = c.Issuer, OriginalIssuer = c.OriginalIssuer, Value = c.Value, ValueType = c.ValueType);
17 Click OK.

Authenticating on Office 365

1 Launch http://office.com/.
2 Login with your credentials.
3 Select Advanced Authentication to go through the multi-factor authentication.
4 You will be redirected to the OAuth or SAML Login page.
5 You must go through the specified chains for authentication.
Configuring Integration with Sentinel

This section provides the configuration information about integrating Advanced Authentication with Sentinel for managing logs. With this integration the syslog files are gathered and transmitted from Advanced Authentication to Sentinel sever, where an administrator can search the events to analyze, monitor, and generate a report.

To configure the integration of Advanced Authentication with Sentinel, perform the following tasks:

- “Configuring the CEF Log Forward Policy on Advanced Authentication” on page 175
- “Searching the Events on Sentinel” on page 175

Configuring the CEF Log Forward Policy on Advanced Authentication

To forward the syslog details to Sentinel, you must configure the CEF log Forward policy by performing the following steps:

1. Open the Advanced Authentication Administration portal.
2. Click Policy > CEF Log Forward.
3. Specify the Sentinel server IP address in Syslog server.
4. Specify the port number in Port.
   For example, you can specify 1443.
5. Select the transport layer details in Transport.
   For example, you can select TCP with TLS.
6. Click Save.
7. Restart the Advanced Authentication server to apply the changes.

Searching the Events on Sentinel

1. Open the Sentinel console.
2. Specify the query ((sev:[0 TO 5])) AND (sp:"CEF") in the Search bar, then click Search.
   The events with severity 0 to 5 are displayed. You can download the events in the csv format.
Maintaining Advanced Authentication

This chapter contains the following sections:

- Chapter 9, “Logging,” on page 179
- Chapter 10, “Reporting,” on page 195
- Chapter 11, “Managing Tokens,” on page 197
- Chapter 12, “Searching a Card Holder’s Information,” on page 199
- Chapter 13, “Troubleshooting,” on page 201
Advanced Authentication provides the logging functionality. All the administrative and user actions and events are logged in this section.

Logs help to debug a problem based on the event or action performed.

There is a hard coded log rotation based on the file size. The maximum size of a log file is 20 MB and for WebAuth logs it is 10 MB. Advanced Authentication stores the last ten log files of each type.

Advanced Authentication supports the following types of logs:

- “Syslog” on page 180
- “RADIUS Logs” on page 194
- “Async Logs” on page 194
- “Fingerprint Logs” on page 194

You can change a time zone in the upper-right section that displays your local time zone. The changes are applied for only the logs displayed and are not applied for the exported logs. Advanced Authentication resets the time zone when you switch from the Logs section or close the Administration portal.

Exporting the Logs

To export logs, perform the following steps:

1. Click Logs.
2. Select the log you want to export.
3. Click Export.
4. Specify a Start date and End date to determine the required logging period.
5. Click Export.
   The exported log files are displayed in the File Name section.
6. Click the exported log file package that is exported in the format aucore-logs_<logging_period>.tar.gz to download it.

Clearing the Logs

You can clear all the logs on the server that you are currently logged on. To clear the logs, perform the following steps:

1. In the Logs page, click Clear.
   A message appears to confirm that you want to continue clearing the logs.

   **NOTE:** It is a good practice to export logs to save as a backup before you delete them.

2. Click OK to clear the logs.
These logs contain information about the system events and actions. The log message is displayed in the format `<date> <host>`.

CEF:0|AAA|Core|<version>|<code>|<message>|<severity>|<endpoint>|<event>|<authentication method name>|<template owner>|<tenant name>|<username>|<uwsgi process id>.

The Syslogs are classified as follows:

- 0 - 99: Maintenance
- 100 - 199: Access
- 200 - 299: App data
- 300 - 399: Endpoints
- 400 - 499: Repositories
- 500 - 599: Local users
- 600 - 699: Repository users
- 700 - 799: User templates
- 800 - 999: Policies
- 900 - 1099: Licenses
- 1000 - 1100: Settings
- 1100 - 1200: Password filter
- 1201 - 1300: Background logon
- 1301 - 1400: Events
- 1401 - 1500: Chains

<table>
<thead>
<tr>
<th>Code</th>
<th>Name</th>
<th>Class</th>
<th>Severity</th>
<th>Optional Parameters</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>New Request</td>
<td>Operational</td>
<td>1</td>
<td>None</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>2</td>
<td>Request failed</td>
<td>Operational</td>
<td>1</td>
<td>None</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>10</td>
<td>Server started</td>
<td>Operational</td>
<td>4</td>
<td>None</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>12</td>
<td>Server stopped</td>
<td>Operational</td>
<td>7</td>
<td>None</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>13</td>
<td>Server unexpectedly stopped</td>
<td>Operational</td>
<td>10</td>
<td>None</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>50</td>
<td>Server Message</td>
<td>Operational</td>
<td>5</td>
<td>Message</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>Code</td>
<td>Name</td>
<td>Class</td>
<td>Severity</td>
<td>Optional Parameters</td>
<td>Example</td>
</tr>
<tr>
<td>------</td>
<td>-------------------------------</td>
<td>----------</td>
<td>----------</td>
<td>--------------------------------------------------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>100</td>
<td>User logon started</td>
<td>Security</td>
<td>4</td>
<td>Username Ep Ep_addr Sid Unit_id Session_id Event Tenant_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>101</td>
<td>User was successfully logged on</td>
<td>Security</td>
<td>7</td>
<td>Username Ep Ep_addr Sid Session_id method_name method_comment method_info Event Tenant_name Template_owner</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>102</td>
<td>User was failed to authenticate</td>
<td>Security</td>
<td>9</td>
<td>Username Ep Ep_addr Sid Session_id Method_name Tenant_name Template_owner</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>103</td>
<td>User was switched to different method</td>
<td>Security</td>
<td>2</td>
<td>Username Ep Ep_addr Sid Session_id New_method_name Tenant_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>Code</td>
<td>Name</td>
<td>Class</td>
<td>Severity</td>
<td>Optional Parameters</td>
<td>Example</td>
</tr>
<tr>
<td>-------</td>
<td>-------------------------------------------</td>
<td>-------------</td>
<td>----------</td>
<td>--------------------------------------------------------------------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>104</td>
<td>User logon session was ended</td>
<td>Security</td>
<td>2</td>
<td>Username Ep Ep_addr Sid Session_id Tenant_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>105</td>
<td>User logon unwanted</td>
<td>Security</td>
<td>9</td>
<td>Username Ep Ep_addr Method_name Tenant_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>106</td>
<td>User was failed to authenticate method in the middle of a chain</td>
<td>Security</td>
<td>2</td>
<td>Username Ep Ep_addr Method_name Tenant_name</td>
<td>June 10 20:10:11 (UTC+0530) host CEF:0</td>
</tr>
<tr>
<td>200</td>
<td>User read app data</td>
<td>Security</td>
<td>3</td>
<td>Username Ep Ep_addr Sid Session_id Data_id Record_id Tenant_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>201</td>
<td>User write app data</td>
<td>Security</td>
<td>4</td>
<td>Username Ep Ep_addr Sid Session_id Data_id Record_id Tenant_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>Code</td>
<td>Name</td>
<td>Class</td>
<td>Severity</td>
<td>Optional Parameters</td>
<td>Example</td>
</tr>
<tr>
<td>--------</td>
<td>-----------------------------</td>
<td>-----------</td>
<td>----------</td>
<td>-----------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>300</td>
<td>Endpoint joined</td>
<td>Security</td>
<td>4</td>
<td>Ep_name Ep_addr Ep_id Username Tenant_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>301</td>
<td>No rights to join endpoint</td>
<td>Security</td>
<td>7</td>
<td>Ep_name Ep_addr Ep_id Username Tenant_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>302</td>
<td>Failed to join endpoint</td>
<td>Operational</td>
<td>7</td>
<td>Ep_name Ep_addr Ep_id Username Reason Tenant_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>303</td>
<td>Endpoint remove</td>
<td>Security</td>
<td>4</td>
<td>Ep_name Ep_addr Ep_id Username Tenant_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td></td>
<td>Failed to change endpoint</td>
<td>Operational</td>
<td>7</td>
<td>Ep_name Ep_addr Ep_id Username Tenant_name reason</td>
<td></td>
</tr>
<tr>
<td>304</td>
<td>No rights to remove endpoint</td>
<td>Security</td>
<td>7</td>
<td>Ep_name Ep_addr Ep_id Username Tenant_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>Code</td>
<td>Name</td>
<td>Class</td>
<td>Severity</td>
<td>Optional Parameters</td>
<td>Example</td>
</tr>
<tr>
<td>-------</td>
<td>------------------------------------------------</td>
<td>-------------------</td>
<td>----------</td>
<td>-------------------------------------------------------------------------------------</td>
<td>--------------------------------------------------------------------------</td>
</tr>
<tr>
<td>305</td>
<td>Failed to remove endpoint</td>
<td>Operational</td>
<td>7</td>
<td>Ep_name Ep_addr Ep_id Username Reason Tenant_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>306</td>
<td>Endpoint session started</td>
<td>Operational</td>
<td>2</td>
<td>Ep_name Ep_id Tenant_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>307</td>
<td>Endpoint session ended</td>
<td>Operational</td>
<td>2</td>
<td>Ep_name Ep_id Tenant_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>308</td>
<td>Invalid endpoint secret</td>
<td>Security</td>
<td>7</td>
<td>Ep_name Ep_id Tenant_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>309</td>
<td>Failed to create endpoint session</td>
<td>Operational</td>
<td>7</td>
<td>Ep_name Ep_id Reason Tenant_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>310</td>
<td>Failed to end endpoint session</td>
<td>Operational</td>
<td>7</td>
<td>Ep_name Ep_id Reason Tenant_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>Code</td>
<td>Name</td>
<td>Class</td>
<td>Severity</td>
<td>Optional Parameters</td>
<td>Example</td>
</tr>
<tr>
<td>------</td>
<td>-----------------------------</td>
<td>---------</td>
<td>----------</td>
<td>---------------------</td>
<td>--------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>401</td>
<td>New repository was added</td>
<td>Operational</td>
<td>4</td>
<td>repo_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>repo_type</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>session_id</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>tenant_name</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>user_name</td>
<td></td>
</tr>
<tr>
<td>402</td>
<td>Failed to add repository</td>
<td>Operational</td>
<td>7</td>
<td>repo_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>repo_type</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>session_id</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>tenant_name</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>user_name</td>
<td></td>
</tr>
<tr>
<td>403</td>
<td>Repository was removed</td>
<td>Operational</td>
<td>4</td>
<td>repo_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>repo_type</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>session_id</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>tenant_name</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>user_name</td>
<td></td>
</tr>
<tr>
<td>404</td>
<td>Failed to remove repository</td>
<td>Operational</td>
<td>7</td>
<td>repo_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>repo_type</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>session_id</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>tenant_name</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>user_name</td>
<td></td>
</tr>
<tr>
<td>405</td>
<td>Repository configuration was changed</td>
<td>Operational</td>
<td>4</td>
<td>repo_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>repo_type</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>session_id</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>tenant_name</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>user_name</td>
<td></td>
</tr>
<tr>
<td>501</td>
<td>Local user was created</td>
<td>Operational</td>
<td>4</td>
<td>user_name</td>
<td>2018-08-29T12:46:21.485790 (UTC+0530)+00:00 linux CEF:0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>session_id</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>tenant_name</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>target_user_name</td>
<td></td>
</tr>
<tr>
<td>Code</td>
<td>Name</td>
<td>Class</td>
<td>Severity</td>
<td>Optional Parameters</td>
<td>Example</td>
</tr>
<tr>
<td>-------</td>
<td>-------------------------------------------</td>
<td>-----------------</td>
<td>----------</td>
<td>---------------------</td>
<td>-------------------------------------------------------------------------</td>
</tr>
<tr>
<td>502</td>
<td>Local user was removed</td>
<td>Operational</td>
<td>5</td>
<td>user_name</td>
<td>2018-08-29T12:45:46.701541 (UTC+0530)+00:00 linux CEF:0</td>
</tr>
<tr>
<td>503</td>
<td>Failed to create local user</td>
<td>Operational</td>
<td>4</td>
<td>user_name</td>
<td>2018-08-29T12:45:25:343315 (UTC+0530)+00:00 linux CEF:0</td>
</tr>
<tr>
<td>504</td>
<td>No rights to remove local user</td>
<td>Security</td>
<td>7</td>
<td>user_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>505</td>
<td>Failed to remove local user</td>
<td>Operational</td>
<td>5</td>
<td>user_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>506</td>
<td>No rights to create local user</td>
<td>Security</td>
<td>7</td>
<td>user_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>601</td>
<td>User was created</td>
<td>Operational</td>
<td>4</td>
<td>user_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>Code</td>
<td>Name</td>
<td>Class</td>
<td>Severity</td>
<td>Optional Parameters</td>
<td>Example</td>
</tr>
<tr>
<td>-------</td>
<td>-----------------------------</td>
<td>---------------</td>
<td>----------</td>
<td>------------------------------------------------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>602</td>
<td>No rights to create user</td>
<td>Security</td>
<td>7</td>
<td>user_name session_id repo_name tenant_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>603</td>
<td>Failed to create user</td>
<td>Operational</td>
<td>4</td>
<td>user_name session_id repo_name reason tenant_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>604</td>
<td>User was removed</td>
<td>Operational</td>
<td>5</td>
<td>user_name session_id repo_name tenant_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>605</td>
<td>No rights to remove user</td>
<td>Security</td>
<td>7</td>
<td>user_name session_id repo_name tenant_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>606</td>
<td>Failed to remove user</td>
<td>Operational</td>
<td>5</td>
<td>user_name session_id repo_name reason tenant_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>607</td>
<td>Role was granted to user</td>
<td>Operational</td>
<td>7</td>
<td>user_name session_id tenant_name target_user_name role_name</td>
<td>2018-08-29T12:46:31.839284 (UTC+0530)+00:00 linux CEF:0</td>
</tr>
<tr>
<td>Code</td>
<td>Name</td>
<td>Class</td>
<td>Severity</td>
<td>Optional Parameters</td>
<td>Example</td>
</tr>
<tr>
<td>------</td>
<td>--------------------------------------------</td>
<td>-------------</td>
<td>----------</td>
<td>-------------------------------------------------------------------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>608</td>
<td>Failed to grant role to user</td>
<td>Operational</td>
<td>8</td>
<td>user_name session_id tenant_name target_user_name role_name</td>
<td>2018-08-29T12:46:31.839284 (UTC+0530)+00:00 linux CEF:0</td>
</tr>
<tr>
<td>609</td>
<td>Role was revoked from user</td>
<td>Operational</td>
<td>7</td>
<td>user_name session_id tenant_name target_user_name role_name</td>
<td>2018-08-29T12:46:35.776761 (UTC+0530)+00:00 linux CEF:0</td>
</tr>
<tr>
<td>610</td>
<td>Failed to revoke role from user</td>
<td>Operational</td>
<td>8</td>
<td>user_name session_id tenant_name target_user_name role_name</td>
<td>2018-08-29T12:46:35.776761 (UTC+0530)+00:00 linux CEF:0</td>
</tr>
<tr>
<td>701</td>
<td>Template was assigned to the user</td>
<td>Security</td>
<td>7</td>
<td>user_name session_id ap_name comment tenant_name template_owner</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>702</td>
<td>Template was enrolled for the user</td>
<td>Security</td>
<td>7</td>
<td>user_name session_id ap_name comment tenant_name template_owner</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>Code</td>
<td>Name</td>
<td>Class</td>
<td>Severity</td>
<td>Optional Parameters</td>
<td>Example</td>
</tr>
<tr>
<td>-------</td>
<td>-------------------------------------------</td>
<td>-----------</td>
<td>----------</td>
<td>-------------------------------------------------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>703</td>
<td>User enroll the assigned template</td>
<td>Security</td>
<td>7</td>
<td>user_name session_id ap_name comment tenant_name template_owner</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>704</td>
<td>Template is linked</td>
<td>Security</td>
<td>8</td>
<td>user_name target_user_name session_id ap_name comment tenant_name template_owner</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>705</td>
<td>Failed to assign template to the user</td>
<td>Security</td>
<td>7</td>
<td>user_name session_id ap_name comment reason tenant_name template_owner</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>706</td>
<td>Failed to enroll template for the user</td>
<td>Security</td>
<td>7</td>
<td>user_name session_id ap_name comment reason tenant_name template_owner</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>707</td>
<td>User can't enroll the assigned template</td>
<td>Security</td>
<td>7</td>
<td>user_name session_id ap_name comment reason tenant_name template_owner</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>Code</td>
<td>Name</td>
<td>Class</td>
<td>Severity</td>
<td>Optional Parameters</td>
<td>Example</td>
</tr>
<tr>
<td>-------</td>
<td>-------------------------------------------</td>
<td>---------------</td>
<td>----------</td>
<td>-------------------------------------------------------------------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>709</td>
<td>Failed to link template</td>
<td>Security</td>
<td>8</td>
<td>user_name target_user_name session_id ap_name comment reason tenant_name template_owner</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>709</td>
<td>Template link was removed</td>
<td>Security</td>
<td>6</td>
<td>user_name target_user_name session_id ap_name comment reason tenant_name template_owner</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>710</td>
<td>Failed to remove template link</td>
<td>Security</td>
<td>6</td>
<td>user_name target_user_name session_id ap_name comment reason tenant_name template_owner</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>711</td>
<td>Template was removed</td>
<td>Security</td>
<td>6</td>
<td>user_name ap_name comment session_id tenant_name template_owner</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>712</td>
<td>Failed to remove template</td>
<td>Security</td>
<td>6</td>
<td>user_name ap_name comment session_id reason tenant_name template_owner</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>Code</td>
<td>Name</td>
<td>Class</td>
<td>Severity</td>
<td>Optional Parameters</td>
<td>Example</td>
</tr>
<tr>
<td>------</td>
<td>-----------------------------------------</td>
<td>----------------</td>
<td>----------</td>
<td>---------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>713</td>
<td>Template was changed</td>
<td>Security</td>
<td>7</td>
<td>user_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>ap_name</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>comment</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>session_id</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>tenant_id</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>template_owner</td>
<td></td>
</tr>
<tr>
<td>714</td>
<td>Failed to change template</td>
<td>Security</td>
<td>6</td>
<td>user_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>ap_name</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>comment</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>session_id</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>tenant_id</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>template_owner</td>
<td></td>
</tr>
<tr>
<td>715</td>
<td>Template was changed during logon</td>
<td>Security</td>
<td>5</td>
<td>user_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>ap_name</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>comment</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>session_id</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>tenant_id</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>template_owner</td>
<td></td>
</tr>
<tr>
<td>801</td>
<td>Policy was changed</td>
<td>Security</td>
<td>7</td>
<td>session_id scope</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>component_id</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>policy_name</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>user_name</td>
<td></td>
</tr>
<tr>
<td>802</td>
<td>No rights to change policy</td>
<td>Security</td>
<td>8</td>
<td>session_id scope</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>component_id</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>policy_name</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>user_name</td>
<td></td>
</tr>
<tr>
<td>803</td>
<td>Failed to change policy</td>
<td>Operational</td>
<td>7</td>
<td>session_id scope</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>component_id</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>policy_name</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>reason user_name</td>
<td></td>
</tr>
<tr>
<td>Code</td>
<td>Name</td>
<td>Class</td>
<td>Severity</td>
<td>Optional Parameters</td>
<td>Example</td>
</tr>
<tr>
<td>------</td>
<td>------</td>
<td>-------------</td>
<td>----------</td>
<td>---------------------</td>
<td>-------------------------------------------------------------------------</td>
</tr>
<tr>
<td>901</td>
<td>New license was added</td>
<td>Operational</td>
<td>3</td>
<td>session_id license_id users_count enabled_features expire_date</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>902</td>
<td>Failed to add license</td>
<td>Operational</td>
<td>8</td>
<td>session_id license_id users_count enabled_features expire_date reason</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>1001</td>
<td>Global setting was changed</td>
<td>Security</td>
<td>9</td>
<td>session_id setting_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>1002</td>
<td>No rights to change global setting</td>
<td>Security</td>
<td>9</td>
<td>session_id setting_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>1003</td>
<td>Failed to change global setting</td>
<td>Operational</td>
<td>9</td>
<td>session_id setting_name reason</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>1101</td>
<td>Password was changed</td>
<td>Security</td>
<td>5</td>
<td>user_name ep ep_addr tenant_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>1102</td>
<td>Password was reset</td>
<td>Security</td>
<td>8</td>
<td>user_name ep ep_addr tenant_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>Code</td>
<td>Name</td>
<td>Class</td>
<td>Severity</td>
<td>Optional Parameters</td>
<td>Example</td>
</tr>
<tr>
<td>------</td>
<td>-----------------------------------</td>
<td>-------------</td>
<td>----------</td>
<td>-----------------------------------------------------</td>
<td>------------------------------------------------------------------------</td>
</tr>
<tr>
<td>1201</td>
<td>User successfully logged on using local cache</td>
<td>Security</td>
<td>8</td>
<td>user_name ep_addr event chain_name logon_time tenant_name</td>
<td>June 10 20:10:11 host CEF:0</td>
</tr>
<tr>
<td>1301</td>
<td>Event was created successfully</td>
<td>Security</td>
<td>4</td>
<td>event tenant_name session_id user_name</td>
<td>Jan 03 17:04:10 host CEF:0</td>
</tr>
<tr>
<td>1302</td>
<td>Failed to create event</td>
<td>Operational</td>
<td>7</td>
<td>event tenant_name reason session_id user_name</td>
<td></td>
</tr>
<tr>
<td>1303</td>
<td>Event was changed successfully</td>
<td>Security</td>
<td>4</td>
<td>event tenant_name session_id user_name</td>
<td>Jan 03 17:05:21 host CEF:0</td>
</tr>
<tr>
<td>1304</td>
<td>Failed to change event</td>
<td>Operational</td>
<td>7</td>
<td>event tenant_name reason session_id user_name</td>
<td></td>
</tr>
<tr>
<td>1305</td>
<td>Event was removed successfully</td>
<td>Security</td>
<td>4</td>
<td>event tenant_name session_id user_name</td>
<td>Jan 03 17:06:40 host CEF:0</td>
</tr>
<tr>
<td>1306</td>
<td>Failed to remove event</td>
<td>Operational</td>
<td>7</td>
<td>event tenant_name reason session_id user_name</td>
<td></td>
</tr>
<tr>
<td>1401</td>
<td>Chain was created successfully</td>
<td>Security</td>
<td>4</td>
<td>chain_name tenant_name session_id user_name</td>
<td>Jan 03 16:54:09 host CEF:0</td>
</tr>
<tr>
<td>1402</td>
<td>Failed to create chain</td>
<td>Operational</td>
<td>7</td>
<td>chain_name tenant_name reason session_id user_name</td>
<td></td>
</tr>
</tbody>
</table>
Logging

RADIUS Logs

These logs contain information about the logs that are recorded for the RADIUS server.

On the server, the `radius.log` file is stored in the `/var/lib/docker/volumes/aaf_radiusd-logs/_data/` directory.

After you export the RADIUS logs, you can find the `radius.log` file in the `/var/log/freeradius/` directory.

Async Logs

These logs contain information about the asynchronized delivery of OTP messages for the SMS, Email, and Voice methods.

On the server, the `async_commander.log` and `async_commander.*.log` files are stored in the `/var/lib/docker/volumes/aaf_aucore-logs/_data/` directory.

After you export the Async logs, you can find the `async_commander.log` and `async_commander.*.log` files in the `/opt/AuCore/logs/` directory.

Fingerprint Logs

These logs contain all information from a Fingerprint service.

On the server, the `nbisd.log` file is stored in the `/var/lib/docker/volumes/aaf_afisd-logs/_data/` directory.

<table>
<thead>
<tr>
<th>Code</th>
<th>Name</th>
<th>Class</th>
<th>Severity</th>
<th>Optional Parameters</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>1403</td>
<td>Chain was changed successfully</td>
<td>Security</td>
<td>4</td>
<td>chain_name tenant_name session_id user_name</td>
<td>Jan 03 16:59:45 host CEF:0</td>
</tr>
<tr>
<td>1404</td>
<td>Failed to change chain</td>
<td>Operational</td>
<td>7</td>
<td>chain_name tenant_name reason session_id user_name</td>
<td></td>
</tr>
<tr>
<td>1405</td>
<td>Chain was removed successfully</td>
<td>Security</td>
<td>4</td>
<td>chain_name tenant_name session_id user_name</td>
<td>Jan 03 16:56:16 host CEF:0</td>
</tr>
<tr>
<td>1406</td>
<td>Failed to remove chain</td>
<td>Operational</td>
<td>7</td>
<td>chain_name tenant_name reason session_id user_name</td>
<td></td>
</tr>
</tbody>
</table>
Advanced Authentication facilitates you to add and view reports according to your requirement. You can view information about the memory utilization, tenant information, successful or failed logins, licenses, and so forth in a graphical representation. You can also export these reports to JSON and CSV formats.

To log in to the Advanced Authentication Reporting portal, launch the URL: https://<NetIQServer>/report and log in with your credentials.

For more information, see “Adding a Report” section.
Managing Tokens

Managing Tokens functionality helps you to import a file that contains information about multiple tokens and you can assign the tokens to specific users such that the user can pass through the OATH authentication method.

To access Tokens Management portal, you must assign chains to the Tokens Management event in the Events section.

To import token files, perform the following steps:

2. Click Add.
3. Click Browse and add a PSKC or CSV file.
4. Select the File type. The options available are:
   - OATH compliant PSKC: This file type must be compliant with OATH. For example, HID OATH TOTP compliant tokens.
   - OATH csv: This file type must contain the format as described in CSV File Format To Import OATH Compliant Tokens. You cannot use the YubiKey CSV files.
   - Yubico csv: In this file type, you must use one of the supported Log configuration output (see YubiKey Personalization Tool > Settings tab > Logging Settings) formats with comma as a delimiter.
   - Traditional format: In this file type, OATH Token Identifier must be enabled.
   - Yubico format: This file type is supported only for HOTP Length set to 6 Digits and OATH Token Identifier set to All numeric.

**IMPORTANT:** Moving Factor Seed must not exceed 100000.

5. Add the encrypted PSKC files. Select Password or Pre-shared key in PSKC file encryption type and provide the information.
6. Click Upload to import tokens from the file.

**NOTE:** Advanced Authentication receives an OTP format from the imported tokens file and stores the information in the enrolled authenticator. Therefore, Advanced Authentication administrator need not change the default value of OTP format on the Method Settings Edit tab. For more information on the OTP format, see OATH OTP.

When the tokens are imported, you can see the list of tokens on the Tokens Management Portal. You must assign these tokens to the users. The tokens can be assigned either by an administrator or by user in the following ways:

- As an administrator, you can do the following:
  1. Click Edit next to the token.
2. Select Owner.
3. Click Save.

- A user can self-enroll a token in the Self-Service portal. Administrator must let the user know an appropriate value from the Serial column for the self-enrollment.

**CSV File Format To Import OATH Compliant Tokens**

A CSV file, which is imported as OATH csv file in the Administration portal > Methods > OATH OTP > OATH Tokens tab, must contain fields with the following parameters:

- Token’s serial number
- Token’s seed
- (Optional) Type of the token: TOTP or HOTP (by default HOTP)
- (Optional) OTP length (default value is 6 digits)
- (Optional) Time step (default value is 30 seconds)

Comma is a delimiter.

The following is an example of a CSV file:

```
Token001, 15d2fa517d3c6b791bd4cc2044c241429307001f
Token002, 8c557fc050721037fd31e1d3345b5d3263263e0f, totp, 8
Token003, 658208e6ea5ac49d5331ba781e66f2c808ccc8e, hotp, 6
Token004, 89f0dfe1c90379da6a1aaca2fc1070f606efe36, totp, 6, 60
```

**IMPORTANT:** For the YubiKey tokens, you must use the traditional format of the CSV (check YubiKey Personalization Tool > Settings tab > Logging Settings) with comma as a delimiter. Use Yubico csv file type (Advanced Authentication Administration portal > Methods > OATH OTP > OATH Tokens).
With the Search Card portal, you can get a card holder’s contact information by tapping the card on the card reader. Information such as name of the card holder, repository information, email address, and mobile number of the user can be obtained.

You must assign chains to the Search card event in the Events section.

**IMPORTANT:** To use this feature, you must have the Device Service installed on the computer.

To get the user information from the card, perform the following steps:

2. Tap a card on the card reader. The card holder’s user name, repository information, email address, and mobile number are displayed.

**NOTE:** If the card was not enrolled before, a message No user was found for this card is displayed.
NOTE: This chapter contains solutions for known issues. If you encounter any problems that are not mentioned here, contact the support service.

This chapter contains the following topics:

- “The ON/OFF Switch Is Broken If the Screen Resolution Is 110%” on page 201
- “Users Can Login Using the Old Password” on page 201
- “Error is Displayed in the User Report Section of the Helpdesk Portal” on page 201

The ON/OFF Switch Is Broken If the Screen Resolution Is 110%

While trying to edit the Lockout options policy, the ON/OFF switch is broken when the screen resolution is 110%.

As a solution, change the screen resolution to 100%.

Users Can Login Using the Old Password

Issue: When users use the LDAP Password only chain for authentication and change their LDAP password, they are still able to log in with their old LDAP password.

Workaround: You must disable the cache logon on Domain Controllers. To disable the cache logon, you must make the following registry changes:

1. Open the registry key: HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa\.
2. Create a DWORD parameter OldPasswordAllowedPeriod and set the parameter's value to 0.

Error is Displayed in the User Report Section of the Helpdesk Portal

Issue: The following error is displayed when navigating to the User Report tab of the Helpdesk portal on a web server:

```
ConnectionError

HTTPSCONNECTIONPOOLO(host='<hostname>', port=443): Max retries exceeded with url:
/admin/api/reports/multisite/table Caused by ProxyError('Cannot connect to proxy.', OSError('Tunnel connection failed: 503 Service Unavailable',))) (Unknown Error)
```

Solution: Perform the following steps:

1. Use yast to set the NO_PROXY settings:
2 Add the internal company’s domain (for example, .sample.com) that exists under No Proxy Domains.

3 Restart the configuration:
   sudo systemctl restart proxyenv aauth