CA CloudMinder™

Getting Started with the Advanced Authentication Service

1.51

CA Technologies Product References

This document references the following CA Technologies products:

- CA CloudMinder™ Identity Management
- CA CloudMinder™ Advanced Authentication
- CA CloudMinder™ Single Sign-On
- CA Directory
- CA IdentityMinder™
- CA AuthMinder™
- CA RiskMinder™
- CA SiteMinder®
- CA SiteMinder® for Secure Proxy Server
- CA Layer 7

Contact CA Technologies

Contact CA Support

For your convenience, CA Technologies provides one site where you can access the information that you need for your Home Office, Small Business, and Enterprise CA Technologies products. At http://ca.com/support, you can access the following resources:

- Online and telephone contact information for technical assistance and customer services
- Information about user communities and forums
- Product and documentation downloads
- CA Support policies and guidelines
- Other helpful resources appropriate for your product

Providing Feedback About Product Documentation

If you have comments or questions about CA Technologies product documentation, you can send a message to techpubs@ca.com.

To provide feedback about CA Technologies product documentation, complete our short customer survey which is available on the CA Support website at http://ca.com/docs.

Contents

Chapter 1: Advanced Authentication Service Overview

CA CloudMinder provides the Advanced Authentication service to protect resources in a tenant organization against unauthorized access. You can configure the Advanced Authentication service for tenants based on input that they provide. Before you learn about the Advanced Authentication service, it is important to understand the CA CloudMinder solution as a whole.

This section provides an overview of CA CloudMinder and the Advanced Authentication service, and it explains how to get started with the service.

This section contains the following topics:

CA CloudMinder Services

CA CloudMinder is a suite of cloud-based security services that enable organizations to manage user identities and authentication effectively. These services are delivered using the software-as-a-service (SaaS) model, and can be described as follows:

- **Identity Management**

  The *Identity Management* service offers user management and provisioning capabilities. This service enables administrators to control access to system resources such as applications and cloud-based services.

- **Advanced Authentication**

  The *Advanced Authentication* service offers protection against unauthorized access by using a combination of *strong authentication* (also known as *two-factor authentication*) and *risk evaluation*.

  The strong authentication feature verifies an end user's identity using a One-Time Password (OTP) or a Public Key Infrastructure (PKI) credential with a password. Consequently, this type of authentication is harder to compromise.

  The risk evaluation feature provides real-time protection against fraud in online transactions by examining a wide range of data that is collected from the end user and their device. A risk score and advice are generated based on this data and the end user is granted access, denied access, or requested to provide additional authentication.

  A tenant can choose to use two-factor authentication, risk evaluation, or both features.

- **Single Sign-On**

  The *Single Sign-On* service offers secure single sign-on across a network of trusted business partners. This service enables an organization to establish federated partnerships so that an end user who is logged in to one application or portal can access a trusted partner application simultaneously without having to log in again.

CA CloudMinder administrators are responsible for installing, hosting, and managing all the CA products that form the CA CloudMinder solution. An administrator at the tenant end, who is given the *tenant administrator* role, can perform the configurations and maintenance that their business requires, such as managing end users and their credentials.

Note: For more information, see *CA CloudMinder Overview*.

Advanced Authentication Features

The Advanced Authentication service provides the following features:

Strong Authentication

Strong authentication addresses the exponential increase in internet-based fraud over the last few years. The basic user name-password model for authentication is no longer sufficient.

Strong authentication uses two-factor authentication, where an end user is required to provide more than one form of identification. For example, in addition to the typical user name-password (something the user knows), the end user also has to provide an additional hardware or software credential (something the user has).

The Advanced Authentication service provides proprietary software credentials, which can be used as the possession factor (something the user has) for authentication. The end user's password or PIN is used as the knowledge factor (something the user knows). As a result, end users retain the familiar user name-password login process. They need to know only their user name and password or PIN, but are protected by a strong authentication solution that works in the background.

The following strong authentication credential types are available to protect resources in an organization:

- ArcotID PKI
- ArcotID OTP

These credential types are discussed in detail in later topics.

Risk Evaluation

When an end user tries to access a protected resource, the Advanced Authentication service first collects a wide range of data, such as details about the following:

- End-user device identification
- Location
- User and transaction

The service evaluates that data using risk evaluation rules.

A *risk evaluation rule* is a set of conditions against which the end user or device data is validated. The result of each rule is then evaluated in the order of priority that is set by an administrator. A score and advice are generated based on the first rule that matched (the higher the risk score, the greater the probability of a fraud). Based on this advice, the end user is granted access, denied access, or asked for additional authentication.

Risk evaluation rules are listed and explained in a later section.

Advanced Authentication Flows

The Advanced Authentication service provides a set of predefined flows that have been derived based on a combination of strong authentication and/or risk evaluation. A hosting administrator can configure these flows based on the organization's needs. Each predefined flow defines the steps that must be performed, in a specific order, to authenticate end users who have been given a specific type of credential.

The predefined advanced authentication flows are as follows:

- ArcotID PKI Only
- ArcotID PKI and Risk
- ArcotID OTP Only
- ArcotID OTP and Risk

These flows are described in detail in a later section.

How to Get Started

The Advanced Authentication service is set up with certain default configurations for strong authentication and risk evaluation. Of these configurations, some can be modified to suit the tenant's requirements, while others are fixed and cannot be modified. Some of the configurations that can be modified require input from the tenant.

To learn about the predefined configurations and the configurations that require tenant input, see Information Required to Configure the Advanced Authentication Service (see page 91).

For detailed information about the advanced authentication features, see the following topics:

Strong Authentication Mechanisms (see page 13)

Risk Evaluation and Fraud Detection (see page 21)

How Advanced Authentication Flows Work (see page 68)

Chapter 2: Strong Authentication Mechanisms

The Advanced Authentication service provides strong authentication by using ArcotID PKI and ArcotID OTP, which are based on the patented *Cryptographic Camouflage* key concealment technology. Tenants can request an authentication mechanism that best suits the security requirements of their organization. In Cryptographic Camouflage, the keys are encrypted such that only one key decrypts it correctly, but can produce many keys that look valid enough to fool an attacker. In this manner, the Cryptographic Camouflage technique protects an end user's private key against dictionary attacks and Man-in-the-Middle (MITM) attacks, as a smartcard does, but entirely in the software format.

Primary authentication

Primary authentication refers to the typical authentication flow in which an end user accessing a protected resource is prompted for the user name and password (or OTP, if the ArcotID OTP credential is used). ArcotID PKI and ArcotID OTP are the supported primary authentication mechanisms.

Secondary authentication

Secondary authentication refers to the additional authentication that is performed in the following cases:

- An end user has either forgotten or wants to reset the password or PIN.
- An end user's ArcotID PKI or ArcotID OTP credential has expired.
- A roaming end user is trying to authenticate from a device that is different from the one used to enroll with the system, or one that is already marked trusted during a previous roaming attempt.
- Risk evaluation is enabled, and it generates an advice to increase authentication for the transaction that the end user is trying to perform.

Question and answer pairs, and Security Code, which is similar to a one-time password, are the supported secondary authentication mechanisms.

A tenant can request a combination of these authentication mechanisms.

As secondary authentication is typically invoked when performing sensitive tasks, it is recommended that a combination of these authentication mechanisms be chained together for enhanced security. CloudMinder supports the enforcement of **two-step authentication** for a selected flow. When two-step authentication is enabled, an end user is authenticated consecutively using two different authentication methods.

The sections that follow describe the primary and secondary authentication mechanisms that the Advanced Authentication service provides.

This section contains the following topics:

ArcotID PKI (see page 15)
ArcotID OTP (see page 17)

Security Code (see page 19)
Question and Answer Pairs (see page 19)

ArcotID PKI

ArcotID PKI is a CA-proprietary secure software credential that provides strong authentication. This credential is used for primary authentication. It protects a user's credentials by using the patented Cryptographic Camouflage key concealment technology. ArcotID PKI can be used to authenticate to a website or other online resource, through a Web browser.

ArcotID PKI is a small data file that resides on the end user's desktop or mobile device. The credential is pushed to the user's device when an end user tries to access a protected resource the first time. During subsequent logins, the user authenticates by providing the user name and LDAP password. Behind the scenes, the Advanced Authentication service verifies the user's identity by accessing the ArcotID PKI credential and then provides access to the resource.

In addition to authenticating the end user, this solution also verifies the authenticity of the site that is requesting for the end user's credentials, ensuring that the user is not providing credentials to a spurious site. Each ArcotID PKI credential contains information about the web domain that issued the credential. This information is used to check whether the site requesting the credential is in fact the same site that issued it. If the site requesting the credential did not issue it, the transaction fails, preventing identity theft and fraud.

ArcotID PKI Features

The important features of the ArcotID PKI credential are as follows:

- An ArcotID PKI can be accessed only with the correct password.
- ArcotID PKI authentication uses a challenge-response authentication protocol. During authentication, a client application on the end user's device signs the challenge with the end user's private key. The signed challenge is then sent to the Advanced Authentication Server for verification.
- A plausible response is generated for every password that is entered, even if the password is incorrect.
- The validity period for the ArcotID PKI credential is configurable.

Support for Roaming Download of ArcotID PKI

The ArcotID PKI credential can be used with any device that the end user uses to access protected resources. This feature provides the end user instant access to critical data and services while keeping the data safe from unauthorized access.

ArcotID PKI Client

To enable end users to authenticate using an ArcotID PKI credential, CA CloudMinder provides the *ArcotID PKI Client* software, which must be installed on the end user's system. To support a wide variety of application environments, the ArcotID PKI Client is available as:

- **Native Client**, which must be installed manually. This client is available in the Downloads section of the CA Support site at https://support.ca.com. Tenants can download it and provide it to their end users.
- **JavaScript Client**, which is pushed to the end user's system during authentication.
- **Mobile client application**, which mobile users can download for free from App Store or Google Play. A single client application can be used with multiple accounts to access different applications and web portals.

The ArcotID PKI Client is used for the following operations:

- To download an ArcotID PKI credential to the end user's device.
- To collect the end user's password, sign the challenge using that password, and send it to the server for verification.

ArcotID OTP

ArcotID OTP is a secure software authentication mechanism that allows the use of mobile phones, iPads, and other PDAs as convenient authentication devices. The ArcotID OTP credential is used for primary authentication, and it supports the Open Authentication (OATH) standard. Similar to the ArcotID PKI credential, ArcotID OTP also uses CA Arcot's patented Cryptographic Camouflage technology to protect credentials from brute force attacks.

Authentication using ArcotID OTP involves the use of a passcode generator. For every session that an end user initiates, a unique OTP is generated, which is only valid for that session or for a very short period. Consequently, OTP authentication lowers the chances of relay attacks. The ArcotID OTP mechanism can be used for authentication on computers and mobile devices.

The passcode generator is the *ArcotID OTP application*, which must be installed on the end user's mobile device. At the time of enrollment, the end user is prompted to set a PIN and is also sent instructions to configure their device for ArcotID OTP generation. Once the device is configured, the ArcotID OTP credential is provisioned to the device. At runtime, the end user opens the ArcotID OTP application, authenticates to it using their PIN, generates an OTP, and uses that OTP to authenticate to a protected resource.

JavaScript ArcotID OTP

For users who do not want to manage the ArcotID OTP application on their device to generate OTPs, the Advanced Authentication service provides a JavaScript Client that invisibly runs in the end user's Web browser and generates an OTP each time it is invoked.

The JavaScript Client eliminates the need for users to read the OTP from a device and then type it into the login page. Typically, the JavaScript Client is invoked when an end user who has registered for ArcotID OTP authentication tries to access a protected resource, but does not have the phone that has the ArcotID OTP application. If the end user states that their phone is not available, secondary authentication is performed and the JavaScript Client is invoked in the background to generate an OTP. This OTP is sent to the Advanced Authentication service for verification.

Support for Roaming Download of ArcotID OTP

The Advanced Authentication service offers roaming capabilities to enable end users to download their ArcotID OTP securely and authenticate from any system when the need arises. Roaming users who do not have the ArcotID OTP application or JavaScript Client on their device can set up a different device to retrieve their ArcotID OTP from the Advanced Authentication service. The downloaded ArcotID OTP can then be used to authenticate to any protected resource in a browser.

To enable roaming, a secondary authentication mechanism must be configured for the user during enrollment. At runtime, if secondary authentication is successful, the ArcotID OTP credential is downloaded to the end user's device.

If Security Question is used for secondary authentication, during enrollment the end user is prompted to provide additional private information, which is composed of a series of user-defined question and answer pairs. Similarly, if Security Code is used for secondary authentication, during enrollment the end user is prompted to provide an email address or telephone number to which the security code must be sent. At runtime, an end user who tries to download the ArcotID OTP from a different device is first authenticated using the questions and answers or the security code that they received as an email message, SMS, or voice message.

ArcotID OTP Application

To enable end users to authenticate using an ArcotID OTP credential, CA CloudMinder provides the *ArcotID OTP* application, which must be installed on the end user's device. To support a wide variety of application environments, the ArcotID OTP application is available in the form of a desktop client and a mobile application:

- The desktop client is available in the Downloads section of the CA Support site at https://support.ca.com. Tenants can download it and provide it to their end users.
- Mobile users can download the ArcotID OTP application for free from App Store or Google Play. A single ArcotID OTP application can be used with multiple accounts to access different applications and web portals.

If an end user has enrolled for ArcotID OTP authentication, a typical authentication flow using ArcotID OTP is as follows:

1. An end user tries to access a protected resource, and is prompted for the user name and ArcotID OTP.
2. The end user opens the ArcotID OTP application on their device, authenticates to it using their PIN, and generates an OTP.
3. The end user then reads the OTP that is generated, switches back to the browser, and provides the user name, password, and OTP on the login page.

   Upon successful authentication, the end user is allowed to access the resource.

Security Code

Security Code is a password credential that is used for secondary authentication. This credential, which is similar to a one-time password, is a randomly generated numeric or alphanumeric string, which is valid for a configured duration. Consequently, this type of authentication lowers the chances of relay attacks.

Security Code can be delivered to an end user through an email message, voice message, or SMS. Tenants can decide on one method of delivery for all users or they can enable users to choose a method of their choice. The email ID and telephone numbers are collected from end users at the time of enrollment.

Security Code over Email

If this option is enabled, then at the time of authentication a security code is sent to the end user in an email.

The Advanced Authentication service supports SMTP protocol for sending emails.

Security Code over SMS

If this option is enabled, then at the time of authentication a security code is sent to the end user as an SMS.

A third-party provider of SMS messaging services, such as Clickatell, is used to send messages to individual end users.

Security Code over Voice

If this option is enabled, then at the time of authentication a security code is sent to the end user as a voice message.

A third-party provider of voice messaging services, such as Adeptra, is used to send voice messages to individual end users.

Question and Answer Pairs

User identity verification by using question and answer pairs is a challenge-response authentication mechanism. End users authenticate themselves by providing correct answers for the questions they are asked. Users set the questions and answers themselves at the time of enrollment.

Chapter 3: Risk Evaluation and Fraud Detection

The Advanced Authentication service provides real-time protection against fraud in online transactions. The following sections describe the risk evaluation features and process in detail:

Data Used for Risk Evaluation (see page 21)

Risk Score and Advice (see page 24)

Risk Evaluation Rules (see page 25)

User Device Association (see page 27)

How Risk Evaluation Works (see page 28)

Data Used for Risk Evaluation

The Advanced Authentication service analyzes the risk in a transaction by using the following data:

End-User Device Identification Data (see page 22)

Location Data (see page 24)

End-User Device Identification Data

The following sections describe the device identification and analytics techniques that the Advanced Authentication service uses.

Machine FingerPrint (MFP)

Machine FingerPrint is the data that is gathered from the end user's device. Machine FingerPrint is also referred to as Device fingerprint or PC fingerprint in industry terms. The device data that is collected is used to generate a risk profile of the device in real time. The data that is collected includes:

- Operating system name and version
- Browser information (such as name, UserAgent, major version, minor version, JavaScript version, and HTTP headers)
- Screen settings (such as height, width, and color depth)
- System information (such as time zone, language, and system locale)

When the end user tries to access a protected resource, the Advanced Authentication service matches the corresponding MFP stored in its database with the MFP calculated from the incoming data. If the match percentage is equal to or more than a preconfigured value (set at the time of service initialization), then the login attempt is considered to be coming from a known, and therefore safe, source.

Note: The MFP is not available during the end user's first transaction attempt. The Advanced Authentication service uses the MFP for risk evaluation only on subsequent transaction attempts from the same device.

Device ID

The *Device ID* is an identifier that the Advanced Authentication service generates and sets on the end user's system to identify and track the device used by the end user for subsequent logins and transactions.

The Device ID can be stored in one of the following formats:

- A Flash Shared Object (FSO), which is a file stored with a .sol or .ssl (if SSL is being used for communication) extension. It is available in the Flash Player directory of the user's profile. This type of identifier is common across most browsers.
- A browser cookie, which is an HTTP-based object whose extension and storage location depend on the browser used by the end user.

When an end user is evaluated for the first time, the Advanced Authentication service generates a unique Device ID and sets it on the user's system. During subsequent login attempts by the end user, the Advanced Authentication service checks whether the Device ID on the user's system matches the Device ID stored in the Advanced Authentication store. If the two Device IDs match, then the transaction attempt is considered to be coming from a known, and therefore safe, device.

Note: The Device ID is not available during the end user's first transaction attempt. The Advanced Authentication service uses the Device ID for risk evaluation only on subsequent transaction attempts from the same device.

DeviceDNA

DeviceDNA uses both MFP and Device ID for more accurate information analyses. To improve the accuracy of the risk evaluation process, more data is collected when the DeviceDNA technique is used than in the case of MFP. The following are examples of some of the data items collected:

- System information (such as platform, CPU, fonts, and MEP)
- Browser information (such as vendor, VendorSubID, and BuildID)
- Screen settings (such as buffer depth, pixel depth, DeviceXDPI, and DeviceYDPI)
- Plug-in information (such as QuickTime, Flash, Windows Media Player, ShockWave, and Internet Explorer plug-ins)
- Network information (such as IP address and connection type)

Location Data

Location data is derived from the end user's device IP address. This data includes geo-location information such as locale, ISP, time zone, and related geographical information. To obtain this data, the Advanced Authentication service is integrated with Quova®, which specializes in providing detailed geographic information based on IP addresses.

During a transaction attempt by an end user, the Advanced Authentication service matches the incoming IP address and the location data derived from this IP address with the corresponding data stored in the Advanced Authentication store. The location data information is also used as an input for some of the risk evaluation rules.

Risk Score and Advice

The Advanced Authentication service evaluates each rule, and returns the score corresponding to the highest-priority rule that returns a negative result. The risk score is then used to generate the corresponding advice, which is returned to the application.

The risk score is an integer from 0 through 100. A high score implies that the chances of fraud are higher. The following are the predefined risk scores and corresponding advices:

- **Score Value 0 – 30**

  Advice: ALLOW

  Default recommended action: Allow the transaction to proceed.

- **Score Value 31 – 50**

  Advice: ALERT

  Default recommended action: Take appropriate action. For example, if the user name is unknown, then ALERT advice is generated. In this scenario, the login attempt can be redirected to an administrator or the end user can be prompted to register with the system.

- **Score Value 51 – 70**

  Advice: INCREASEAUTH

  Default recommended action: Perform additional authentication before proceeding any further.

- **Score Value 71 – 100**

  Advice: DENY

  Default recommended action: Deny the transaction.

Risk Evaluation Rules

The Advanced Authentication service provides the following risk evaluation rules:

- **Exception User Check**

  An organization may choose to exclude an end user from risk evaluation during a certain time interval. For example, suppose an end user travels to a country that is configured as negative in the Advanced Authentication service. For the duration of the user's stay in that country, the user can be designated as an exception user. All transactions that originate from that user during that specified period are allowed to proceed. None of the other risk evaluation rules are applied when these exception users log in to perform a transaction. In other words, these exception users are allowed to proceed with their transaction even if their transaction did not clear some of the other rules.

  Transactions originating from exception users receive a low risk score and the advice is typically ALLOW.

- **Untrusted IP Check**

  The Untrusted IP Check rule uses a list of IP addresses that originate from anonymizer proxies or have been the origin of known fraudulent transactions in the past.

  Transactions originating from these negative IP addresses receive a high score and the advice is DENY.

- **Negative Country Check**

  The Negative Country Check rule uses a list of countries from which a significant number of fraudulent transaction attempts have been made in the past. During a transaction attempt, the country information is derived from the device IP address.

  Transactions that originate from configured negative countries receive a high score and the advice is DENY.

- **Trusted IP/Aggregator Check**

  The Trusted IP/Aggregator Check rule uses a list of IP addresses and aggregators that are considered to be trusted sources by the organization. Many organizations use the services of account and data aggregation service providers to expand their online reach. The originating IP address when an end user logs in from a protected portal is different from the IP address used when the end user comes in through such an aggregator. An organization may choose to exclude a transaction attempt from risk evaluation if the originating IP address shows a trusted source.

  Transactions originating from IP addresses and aggregators that are trusted by the organization receive a low score, and the advice is ALLOW.

- **User Known**

  The User Known rule uses a list of end users who are already registered with the Advanced Authentication service.

  If the end user is unknown to the Advanced Authentication service, then an ALERT advice is returned. An administrator can then choose to prompt the end user to register with the system.

- **DeviceID Known**

  The DeviceID Known rule uses a list of Device IDs that have been generated and assigned to end users by the Advanced Authentication service.

  Transactions originating from known devices receive a low risk score and the advice is ALLOW.

- **User Associated with DeviceID**

  The User Associated with DeviceID rule uses a list of user-device associations that were generated during earlier transactions.

  Transactions originating from a known device and known user receive a low score, and the advice is ALLOW.

  Transactions originating from a known device that is not associated with a known user receive a medium score, and the advice is INCREASEAUTH.

- **Device MFP Match**

  The Device MFP Match rule uses a list of known devices and their associated DeviceDNAs.

  Transactions originating from a known device whose DeviceDNA does not match receive a medium score, and the advice is INCREASEAUTH.

  Transactions originating from an unknown device that is not associated with a known user receive a high score, and the advice is DENY.

- **User Velocity Check**

  The User Velocity Check rule checks for the frequency with which an end user is trying to perform transactions. Frequent use of the same user ID could be an indication of risky behavior. For example, a fraudster might use the same user ID and password from different devices to watch a specific activity in a targeted account.

  Too many transactions originating from the same user within a short interval receive a high score and the advice is DENY.

- **Device Velocity Check**

  The Device Velocity Check rule checks for the frequency with which a device is used for transactions. Frequent use of the same device could also be an indication of risky behavior. For example, a fraudster might use the same device to test multiple combinations of user IDs and passwords.

Too many transactions originating from the same user device within a short interval receive a high score and the advice is DENY.

- **Zone Hopping Check**

  In the case of consecutive logins from locations in different time zones, the Zone Hopping Check rule checks for the time interval between login attempts. If an end user logs in from two long-distance locations within a short time span by using the same user ID, this might be a strong indication of fraudulent activity.

  However, there may be cases where a User ID is shared, in which case, the Advanced Authentication service understands that the two people sharing the same User ID can be in geographically different locations and responds with an appropriate response.

  Transactions originating from the same user from locations that are far apart from each other within a short interval receive a high score and the advice is DENY.

Consider an example scenario where four rules are configured in the following order:

1. Negative IP, with a score of 85
2. User Velocity, with a score of 70
3. Device Velocity, with a score of 65
4. DeviceID Known, with a score of 30

If the Advanced Authentication service determines that a transaction is coming from a negative IP address, then it returns a score of 85 (DENY), based on the first configured rule that matched. Another transaction exceeding the configured Device Velocity gets a score of 65, which results in a request for increased authentication.

User Device Association

After authenticating an end user for the first time, the Advanced Authentication service creates and stores associations of the end user and the device used for that transaction. This is referred to as device binding in risk evaluation terminology. Users who are not bound to a device are more likely to receive the INCREASEAUTH advice at the time of authentication. The Advanced Authentication service also allows users to be bound to more than one device. For example, an end user can use a work and a home computer to access an application. Similarly, a single device can be bound to more than one user. For example, members of a family can use the same computer to access an application.

How Risk Evaluation Works

The Advanced Authentication service uses rules to evaluate risk in a transaction. By default, each rule is assigned a priority and is evaluated in the specific order of its priority level. Risk assessment can be performed either before the user logs in or after the user has logged in.

A typical risk assessment flow is as follows:

1. An end user accesses a protected application.
2. The application calls the Advanced Authentication service to analyze the risk associated with the transaction.
3. The Advanced Authentication service evaluates the risk by using the incoming IP address of the user and the configured rules. It uses the data discussed in the section, Location Information, and does the following:
   a. Executes all the applicable rules, in the order of execution priority.
   b. This execution priority is internal, and is defined by the Advanced Authentication service.
   c. Generates an individual risk score and advice for each rule that it executes.
   d. Uses the result for each rule and parses the rules based on the scoring priority.
   e. Stops the scoring activity at the first matched rule.
   f. Returns the score and advice of the rule that matched as final.

   Note: If the first rule itself matched, then steps c onwards are not performed.
4. Based on the result of rules that were executed and whether the assessed information matched, the Advanced Authentication service generates a risk score and advice.
5. The end user is validated as follows:
   - If the risk is low, the user is allowed to access the application.
   - If the risk is high, the user is denied access to the application.
   - If the transaction is tagged as suspicious, then the application challenges the user for additional (secondary) authentication to prove their identity.

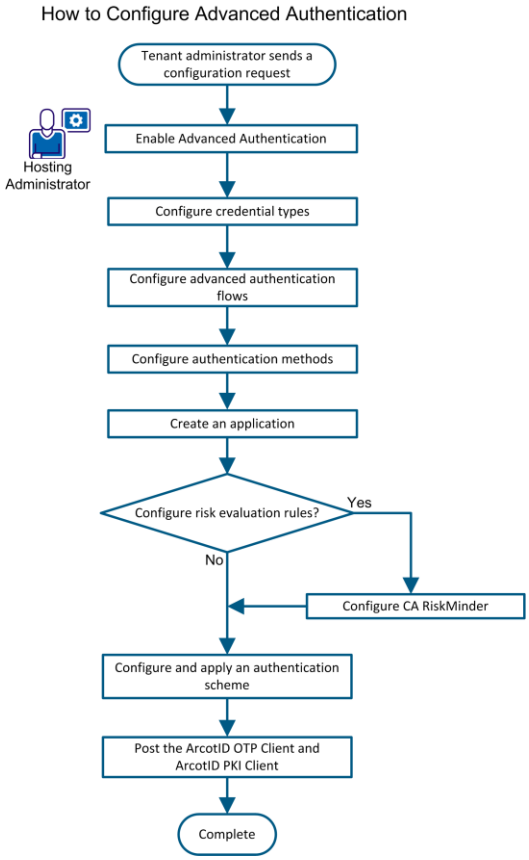
Chapter 4: Configuring Advanced Authentication

This section contains the following topics:

How to Configure Advanced Authentication

The tenant administrator decides on the credential types and the corresponding advanced authentication flows that must be used to protect access to the tenant's resources. The tenant administrator sends this information as a configuration request to the hosting administrator. As the hosting administrator, you configure Advanced Authentication according to the requirements specified in the configuration request.

The following diagram outlines the steps that are involved in configuring Advanced Authentication:



To configure Advanced Authentication, complete the following steps:

1. Enable Advanced Authentication.
2. Configure credential types (see page 31).
3. Configure advanced authentication flows (see page 36).
4. Configure authentication methods (see page 38).
5. Create an application (see page 42).
6. Configure CA RiskMinder (see page 45).
7. Configure and apply an authentication scheme (see page 48).
8. Post the ArcotID OTP Client and ArcotID PKI Client (see page 52).

Configure Credential Types

The credential types form the basis of the advanced authentication flows. To configure a credential type, you enable the credential type and then set values for the attributes of the credential type. The values that you set are specified in the configuration request that is sent by the tenant administrator.

Note: For information about managing credentials after they are assigned to end users, see the CA Arcot Administration Console documentation.

Follow these steps:

1. Log in to the User Console.
2. Select Advanced Authentication, Configure Credential Types.

   The Configure Credential Types: Credential Type screen opens.
3. Use the arrow icons to move the required credential types to the Enabled list.
4. Click Next.

   The Configure Credential Types: Configure Enabled Credentials screen opens.
5. For each credential type to be configured, click the pencil icon, enter values to configure the credential type, and then click Submit.

   You can configure any combination of the following credential types:

   - **ArcotID OTP:** Specify a value for the following fields of the ArcotID OTP credential:

     PIN length

     Specifies the minimum length of the PIN.

     Select end date

     Specifies the expiry date for the credential.

In addition, specify values for the fields of the activation OTP profile:

Type

Indicates whether the activation OTP is numeric or alphanumeric.

OTP length

Specifies the length of the activation OTP.

Validity period

Specifies the period for which the activation OTP is valid.

Usage times

Specifies the number of times for which an activation OTP can be used.

- **ArcotID PKI:** Specify values for the following fields of the ArcotID PKI credential:

  ArcotID key length (in Bits)

  Specifies the length (in bits) of the credential.

  Use mobile client

  Indicates that authentication using the ArcotID PKI mobile client is enabled.

  Note: If you select this option, when configuring Advanced Authentication flows, you *must* configure at least one secondary authentication mechanism each for the expiry and roaming scenarios that use the ArcotID PKI mobile client. For more information, see Configure Advanced Authentication Flows (see page 36).

  Select end date

  Specifies the expiry date for the credential.

  ArcotID client preference

  Specifies the order of preference for ArcotID PKI clients. During enrollment, the Advanced Authentication server uses the first client in the order of preference to deliver and interact with the credential on the end user's device. If the attempt to use the first client fails, then the server uses the second client, and so on. During authentication, the client that is available (in the order of preference) on the end user's device is used to sign the challenge in the challenge-response transaction between the browser and Advanced Authentication server.

- **Risk Evaluation:** The following fields for configuring the Risk Evaluation credential type are not self-explanatory:

  Risk cookie type

  Specifies whether the cookie that stores risk-related information on the end user's device must be a Flash-based cookie or an HTTP-based cookie.

  HTTP Cookie Max Age (days)

  Defines the number of days after which the HTTP cookie must automatically expire. This field is displayed only after you select HTTP cookie from the Risk cookie type list.

- **Security Code:** The following fields for configuring the Security Code credential type are not self-explanatory:

  Note: Security Code is used during the secondary authentication process.

  Type

  Specifies whether the security code is only numeric or alphanumeric.

  Security Code length

  Specifies the length (in characters) of the credential.

  Validity Period

  Specifies the period for which the security code is valid.

  Lockout credential after

  Specifies the number of failed attempts after which the credential is locked.

  - If you want to configure the Security Code over Email option for this credential type, set values for the following fields:

    Enable

    Indicates that the Security Code over Email credential type must be enabled.

    From

    Defines the sender’s email address to be shown in the email that is sent to end users. If the email sent to the end user does not reach the end user for any reason, email notification is automatically sent to this email address.

Subject

Defines the subject line to be shown in the email that is sent to end users.

Email template

Defines the message to be shown in the email that is sent to end users. The security code is inserted in this message. The following is a sample message:

```
User [[USERNAME]], your Security Code is [[SECURITYCODE]].
```

In this example, [[USERNAME]] and [[SECURITYCODE]] are variables that are replaced at run time by actual values.

- If you want to configure the Security Code over SMS credential type, set values for the following fields:

  Enable

  Indicates that the Security Code over SMS credential type must be enabled.

  SMS Template

  Defines the template of the SMS message that is sent to end users. The security code is included in this message. The following is a sample message:

  ```
  User [[USERNAME]], your Security Code is [[SECURITYCODE]].
  ```

  In this example, [[USERNAME]] and [[SECURITYCODE]] are variables that are replaced at run time by actual values.

  SMS Provider Post URL

  Defines the URL from where the SMS service can be accessed.

  Username

  Defines the user name of the SMS account that has been created for the tenant.

  Password

  Defines the password of the SMS account that has been created for the tenant.

  App ID

  Specifies the unique identifier of the SMS API that handles the SMS request sent from CA CloudMinder.

  From

  Defines the string or number that must be displayed in the From field of the SMS message that is sent to end users.

- If you want to configure the Security Code over Voice credential type, set values for the following fields:

  Enable

  Indicates that the Security Code over Voice credential type must be enabled.

  Client ID

  Specifies the client ID that has been assigned to the tenant.

  Country ID

  Specifies the ISO country ID of the tenant's organization.

6. Click Submit.

   The Confirmation: Task Completed message appears after you click Submit. In addition, the current date and time are displayed in the Last Configured Date column for each credential type that you configure.

7. Click Finish after you configure all the credential types that you have enabled.

8. Refresh the cache in CA AuthMinder by performing the following steps:

   a. Log in to CA Arcot Administration Console as a Global Administrator.

   b. Select Services and Server Configurations, Administration Console, Refresh Cache in the System Configuration.

      The Refresh Cache screen opens.

   c. Select any one or both of the following options:

      - Refresh System Configuration
      - Refresh Organization Configuration

   d. Click OK.

      A message stating that the request was submitted successfully appears.

9. View the status of the cache refresh request by performing the following steps:

   a. Select Services and Server Configurations, Administration Console, Check Cache Refresh Status.

      The Search Cache Refresh Request screen opens.

   b. Select the request ID of the refresh request, and then click Search.

      The status of the refresh request is displayed. The SUCCESS message in the Status column indicates that the configuration changes made in the credential types are now effective.

More information:

Troubleshooting Advanced Authentication Errors (see page 53)

Configure Advanced Authentication Flows

The advanced authentication flows that you can enable and configure are based on the credential types that you have enabled. You configure the advanced authentication flows that are requested by the tenant administrator.

Follow these steps:

1. Log in to the User Console.
2. Select Advanced Authentication, Configure Advanced Authentication Flow.

   The Select Flow Types screen opens.
3. Use the arrow icons to move advanced authentication flow types to the Enabled list.
4. Click Next.

   The Enabled Flow Types screen opens.

5. Perform the following steps for each advanced authentication flow type that you have enabled:

   a. Click the pencil icon next to the advanced authentication flow type.

      The Flow Configuration screen displays a list of the different scenarios in which the end user is prompted for secondary authentication.

   b. Select the secondary authentication methods that must be enabled for each scenario.

      Note: An end user forgetting the password is an example of a use case in which the end user is prompted for secondary authentication. For information about all such use cases, see the *Getting Started Guide* for Advanced Authentication.

      Depending on the advanced authentication flow type that you are configuring, you can select any one or a combination of the following secondary authentication methods:

      - Security Question
      - Security Code over Email
      - Security Code over SMS
      - Security Code over Voice

      Note: Consider the following when selecting these mechanisms:

      - If you are configuring the ArcotID OTP with Risk flow or the ArcotID PKI with Risk flow, select at least two secondary authentication methods.
      - If you selected the Use mobile client option when configuring the ArcotID PKI credential type (as described in Configure Credential Types (see page 31)), then you must select at least one secondary authentication mechanism each for the Expiry from Mobile PKI Client and Roaming from Mobile PKI Client scenarios. If no authentication mechanism is selected, the end user cannot log in at run time.

   c. Select the Two Steps option to enforce two-step secondary authentication for a particular scenario.

      As secondary authentication is invoked when performing sensitive tasks, such as resetting passwords or authenticating roaming users, it is recommended that a combination of authentication mechanisms be chained together. Chaining of secondary authentication mechanisms provides a higher level of security.

      Note: Consider the following when selecting this option:

      - The Two Steps option is enabled only if you select 2 or more authentication mechanisms.
      - You can chain Security Question and any of the Security Code types, but you cannot chain two types of Security Code together.

- Two-step authentication is not applicable for scenarios that use the ArcotID PKI mobile client, and therefore, this option is disabled. If multiple authentication mechanisms are selected for the mobile scenarios, all the mechanisms are invoked one by one. The end user is not presented a choice.

d. Click Submit.

The current date and time is displayed in the Last Configured Date column.

6. Click Finish after you configure the required advanced authentication flows.

The configured advanced authentication flows are now available for use in authentication schemes that can be configured for the tenant's resources.

Configure Authentication Methods

An authentication method represents how an application is protected. After you configure an authentication method, you assign it to the application you want to protect. Multiple applications can use the same authentication method. A single application can reference multiple authentication methods.

Configure an authentication method that satisfies the protection requirements for an application.

Note: The system creates authentication methods corresponding to each of the advanced authentication flows. If you are configuring Advanced Authentication for the tenant, do not create an authentication method. Modify the existing authentication method as described in this procedure.

Follow these steps:

1. Log in to the User Console.
2. Navigate to Applications, Authentication Methods, Create an Authentication method.
3. In the top section of the Create Authentication method screen, complete the following fields:

Name

Enter a string that identifies the authentication method you are configuring.

Description

Enter a description for the authentication method. The login page displays this description as a label.

Enabled

Select this check box to make the authentication method immediately available.

4. In the Configure Authentication Method section, select one of the following options and enter the authentication URL for that option.

   When the authentication method is associated with an application, the authentication service appends the redirect URL for the application.

   Note the following variables in the URLs:

   cloud_host is the CA CloudMinder system.

   local_entity_ID is the name of the local entity that is specified in the IdP-to-SP partnership, which is configured at the CSP console.

   remote_entity_ID, consumer_entity_ID or resource_partner_ID is the name of the remote entity that is specified in the configuration of the asserting-to-relying party partnership. The partnership is configured at the CSP console.

   Basic

   Represents a form-based authentication scheme that uses the basic credentials of a user name and a password. The basic authentication method corresponds to the HTML Forms authentication scheme in the CSP console.

   Enter the authentication URL of the following format:

   http://*cloud_host*:*port*/chs/redirectservlet/*tenant_tag*/forms

   tenant_tag is a unique identifier for a tenant. You specify the tag when deploying a tenant environment in the CSP console. To view a list of tags, select the Tenants tab.

   External IDP—Google or Facebook

   Represents a third-party identity provider (IdP) that authenticates users. Social media sites, such as Google or Facebook can serve as external IdPs. Other federated partners that support the SAML and WS-Federation protocols can also serve as external IdPs.

   If Google or Facebook is acting as the third-party IdP, specify the OpenID or OAuth authentication method. Each site supports both protocols.

   Enter the relevant URL for the protocol, as shown:

   OpenID

   http://*cloud_host*:*port*/affwebservices/*tenant_tag*/*duplicate_openid_file*.jsp

   When configuring the OpenID authentication scheme at the CSP console, the default openid.jsp file is copied and given a unique name, such as openid-google.jsp. Having a unique jsp file is necessary to distinguish OpenID configurations.

   The default JSP file is located in the directory /opt/CA/secure-proxy/Tomcat/webapps/affwebservices/redirectjsp.

OAuth

http://*cloud_host*:*port/*affwebservices/*tenant_tag*/*duplicate_oauth_file*.jsp

When configuring the OAuth authentication scheme in the CSP console, the default oauth.jsp file is copied and given a unique name, such as oauth-google.jsp. Having a unique jsp file is necessary to distinguish OAuth configurations.

The default JSP file is located in the directory /opt/CA/secure-proxy/Tomcat/webapps/affwebservices/redirectjsp.

tenant_tag is a unique identifier for a tenant. You specify the tag when deploying a tenant environment in the CSP console. To view a list of tags, select the Tenants tab.

External IDP—Other

Select Other when a SAML or WS-Federation-compliant partner is the IdP. The federation profiles SAML 1.1, SAML 2.0, and WS-Federation 1.2 are all supported.

Enter the relevant URL for the protocol, as shown.

For SAML 1.1 transactions
http://*cloud_host.domain:port*/affwebservices/public/intersitetransfer?CONSUMERID=*consumer_entity_ID*&TARGET=http://*consumer_site*/*target_url*

For SAML 2.0 SP-initiated transactions

http://*cloud_host.domain:port/*affwebservices/public/saml2authnrequest?ProviderID=*local_entity_ID*&RelayState=http://*sp_site*/*target_url*

For SAML 2.0 IdP-initiated transactions

http://*cloud_host.domain:port/*affwebservices/public/saml2authnrequest?SPID=*remote_entity_ID*&RelayState=http://sp_site/*target_url*

For WS-Federation IP-initiated transaction

http://*cloud_host.domain:port*/affwebservices/public/wsfeddispatcher?wa=wsignin1.0&wtrealm=*resource_partner_ID*&wctx=*target_url*

Advanced Authentication

Represents one of the authentication protocols that the CA CloudMinder Advanced Authentication Service provides.

Select one of the following options and the URL is entered automatically:

For ArcotID PKI Only

For environments created in CA CloudMinder 1.51 or later:

https://cl*oud_host*:p*ort*/chs/redirectservlet/*tenant_tag*/arcotid

For environments created before CA CloudMinder 1.51:

https://cl*oud_host*:p*ort*/affwebservices/<tenant-name>/arcotid.jsp

For ArcotID PKI with Risk

For environments created in CA CloudMinder 1.51 or later:

https://cl*oud_host*:p*ort*/chs/redirectservlet/*tenant_tag*/arcotidrisk

For environments created before CA CloudMinder 1.51:

https://cl*oud_host*:p*ort*/affwebservices/<tenant-name>/arcotidrisk.jsp

For ArcotID OTP Only

For environments created in CA CloudMinder 1.51 or later:

https://cl*oud_host*:p*ort*/chs/redirectservlet/*tenant_tag*/arcototp

For environments created before CA CloudMinder 1.51:

https://cl*oud_host*:p*ort*/affwebservices/<tenant-name>/arcototp.jsp

For ArcotID OTP with Risk

For environments created in CA CloudMinder 1.51 or later:

https://cl*oud_host*:p*ort*/chs/redirectservlet/*tenant_tag*/arcototprisk

For environments created before CA CloudMinder 1.51:

https://cl*oud_host*:p*ort*/affwebservices/<tenant-name>/arcototprisk.jsp

tenant_tag is a unique identifier for a tenant. You specify the tag when deploying a tenant environment in the CSP console. To view a list of tags, select the Tenants tab.

5. Click Submit.

The authentication method is available to protect an application.

Create an Application

In the User Console, an application represents the resource that the tenant administrator wants to protect. An application defines the type and level of security that end users encounter when they try to access the resource. You can apply any one or a combination of the authentication methods that you define to protect access to the application.

When a tenant is created in CA CloudMinder, the following applications are automatically created for the tenant:

- An application for the JCS Management Console
- An application for the User Console

You configure both applications according to the tenant's requirements. In addition, you can create applications to secure other resources of the tenant.

After an application is configured, the application icon is displayed on the home page of the User Console. Users can click the icon to access the application. As an administrator, you can also give end users access to the application by inserting a link to the application in any web page. For example, you can insert an icon on your corporate web portal that links to the application.

Note: This section describes the steps to modify an application. These are very similar to the steps to create an application. There are differences only in the first few steps of the procedure.

Follow these steps:

1. Log in to the User Console.
2. Select Applications, Applications, Modify Application.

   The Modify Application screen opens.
3. Use the search feature to display the list of applications for the tenant.

   The list of applications whose names meet the search criteria is displayed. If this is the first time you are performing this procedure, the search results display only the two preconfigured applications that are mentioned earlier in this section.

4. (Optional) Associate a group with the application.
5. Enter a launch URL for the application.

   A launch URL is the fully qualified domain name of the software resource you want to make available to users. Enter the fully qualified domain name of the software resource in the following format:

   https://resource-domain-name

   Example: https://forward-inc.com

   Note: Forward, Inc. is a fictitious company name that is used strictly for instructional purposes only and is not meant to reference an existing company.

   If you are creating an application for the User Console, enter a launch URL in the following format:

   `https://SPS-hostname/iam/im/tenant-name/`

   Example: https://forward-inc.com/iam/im/forward01/
6. Select a logo.

   This is the icon for the application that appears in the User Console home page. Users can click the icon to access the software resource.

   Note: You can also give users access to the application by inserting a link to the application on a web page.
7. Enter a welcome message.

   When users click any link you provide to the application, a login screen opens. The welcome message appears at the top of the login screen.

8. Select a self-registration task.

   With a self-registration task specified, end users who do not have an account can register themselves with the application. You can select one of the following self-registration tasks:

   Create New Account

   Presents a simple registration form. When this form is submitted, a user account is created.

   Create New Account with Workflow

   Presents a simple registration form. When this form is submitted, the request for creating a user account is forwarded to one or more approvers. The account is created on approval of the request.

   Create New Account with Domain Validation

   Presents a simple registration form. When this form is submitted, the user's email domain is compared with the tenant email domain. If the domains match, a confirmation email is sent to the user. The account is created upon user confirmation.

   Note: The tenant email domain is specified in the User Console, under Tenant Administration, Tenant Settings.

   Self-Registration with Attribute Exchange

   Do not select this self-registration task in the context of application access. This task is intended for a different purpose.

9. Click Add in the Authentication Methods area.

   The Select Authentication Methods screen displays a list of the authentication methods available in the tenant environment.

10. Select one or more authentication methods.

11. Click Select.

    The Create Application screen appears, updated with the list of authentication methods that you select.

12. (Optional) Select a default authentication method from the drop-down list.

    The application is created.

Configure CA RiskMinder

CA RiskMinder is one of the components of Advanced Authentication. When a tenant is created, an organization representing the tenant is automatically created in CA RiskMinder. The Risk Evaluation credential type is based on the predefined risk evaluation rules in CA RiskMinder. If the tenant administrator wants to use the Risk Evaluation credential, the tenant administrator sends the configuration settings for the risk evaluation rules as part of the configuration request. You can apply these configuration settings for the risk evaluation rules. See Configure risk evaluation rules (see page 46) for detailed information.

Note: For detailed information about the procedure to configure CA RiskMinder, see the *CA Arcot RiskMinder Administration Guide*.

Assign a Channel to the Organization

CA RiskMinder supports risk evaluation requests coming from multiple channels. By assigning channels to the organization that represents the tenant, you specify the type of risk evaluation requests that must be processed.

When a tenant is created, channels are automatically assigned to the tenant. In addition, a default channel is assigned to the tenant. Perform the procedure that is described in this section only if the tenant administrator requests any changes or additions to these default assignments.

Important! Configuring channels is expected to be a one-time configuration. You can add a channel to your existing deployment, but removing support for a channel and changing the default channel requires careful consideration. If you want to change these settings in a production environment, contact CA Support to understand the implications.

Follow these steps:

1. Log in to the CA RiskMinder Administration Console as the Global Administrator.
2. Select Organizations, Manage Organizations.
3. Use the search feature to search for and open the organization.
4. Click the RiskFort Configuration tab.
5. Click Assign Channels and Configure Default Account Types in the General RiskFort Configurations section.
6. Select the Select Channels to Associate check box for the channels that you want to associate with the organization.

7. Select one of the assigned channels as the default channel.
8. Click Save.

   Channels are assigned to the organization.

Configure Risk Evaluation Rules

Some of the predefined risk evaluation rules have default values. The tenant administrator can specify that these defaults values must be accepted. Alternatively, the tenant administrator can specify the values that they want to set. You set these values to configure the risk evaluation rules.

Follow these steps:

1. Log in to the CA Arcot Administration Console as the Global Administrator.
2. Create the ruleset as follows:
   a. Click Services and Server Configurations, RiskFort, Create Ruleset.

      The Create Ruleset screen opens.

   b. Enter a name for the ruleset in the Name field.

   c. (Optional) If you want to copy the rules configuration from an existing ruleset, select the Copy from an Existing Ruleset check box and then select the name of the ruleset whose configuration you want to copy.

   d. Click Create.

      The ruleset is created.

3. Configure the rules in the ruleset as follows:
   a. Select Services and Server Configurations, RiskFort, Rules and Scoring Management.

      The Rules and Scoring Configuration screen opens.

   b. Select the ruleset from the Select the Rulesets list.

      The Rules and Scoring Management screen opens.

   c. Perform the following steps in the Proposed column for each rule that you want to enable or modify:

      - Ensure that the Enable check box is selected.
      - Set the risk score and the priority.
      - Click the rule name in the Rule Name column.
      - (Optional) Specify values to configure the rule if you do not want to accept the default settings.

        Note: Some of the rules are not configurable.

   d. Set the default risk score for the ruleset in the table that is displayed below the list of rules, and then click Save.

4. Migrate the changes to production by performing the following steps:
   a. Select Services and Server Configurations, Migrate to Production, Migrate to Production.

      The Migrate to Production screen opens.

   b. Select the ruleset from the Select Rulesets list, and then click Migrate.

      The Migrate to Production screen opens.

   c. Click Confirm.

      The request to migrate the updated ruleset to production is sent to RiskMinder Server.

5. Refresh the server cache by performing the following steps:
   a. Select Services and Server Configurations, Administration Console, Refresh Cache.

      The Refresh Cache screen opens.

   b. Select Refresh System Configuration, and then click OK.

      A confirmation message appears.

   c. Click OK.

      A message displaying the request ID for the refresh request appears.

6. Verify that the cache refresh has been carried out by performing the following steps:

   a. Select Services and Server Configurations tab, Administration Console, Check Cache Refresh Status.

      The Search Cache Refresh Request screen opens.

   b. Enter the request ID, and then click Search.

      The View Cache Refresh Request screen opens. Use the information that is displayed on this screen to verify that the cache has been refreshed.

   The risk evaluation rules are configured.

Configure and Apply an Authentication Scheme

Authentication schemes corresponding to the advanced authentication flows are preconfigured in the CSP Console. These authentication schemes are as follows:

- For the ArcotID OTP Only flow: ARCOTOTP_MOBILE_ONLY_AUTH_SCHEME
- For the ArcotID OTP with Risk flow: RISK_AND_ARCOTOTPMOB_ AUTH_SCHEME
- For the ArcotID PKI Only flow: ARCOTID_ONLY_AUTH_SCHEME
- For the ArcotID PKI with Risk flow: RISK_AND_ARCOTID_ AUTH_SCHEME

You establish a one-to-one correspondence between an authentication method configured in the User Console and an authentication scheme in the CSP Console. The authentication method and authentication scheme work together to protect access to the specified application.

The authentication scheme protects the authentication URL that is specified for a given authentication method. To apply the authentication scheme, you assign the authentication scheme to a realm and then include the realm in a policy.

Important! You can also use these steps to apply an authentication scheme for protecting the User Console.

Follow these steps:

1. Configure a realm and a rule for the resource (see page 49).
2. Add rules to the tenant policy (see page 51).
3. Configure an authentication scheme for the User Console (see page 52).

Configure a Realm and a Rule for the Resource

A realm groups resources that have similar security requirements and share a common authentication scheme. In the tenant domain, create a realm for each authentication scheme that the tenant administrator wants to use.

Note: The following procedure assumes that you are creating an object. You can also copy the properties of an existing object to create an object.

Follow these steps:

1. Log in to the CSP console.
2. Select Policies, Domain, Realms.

   The Realms screen opens.
3. Click Create Realm.
4. Select the tenant domain that you want to modify, and then click Next.

   Note: The tenant domain name is in the *tenant-tag*Domain format.
5. Type a name and description for the realm.

   Specify a name that indicates that the realm is for an authentication URL.
6. Click Lookup Agent/Agent Group.

7. Select **cam-agent** from the list of agents, and then click OK.
8. Specify the resource filter for the authentication scheme. This scheme must tie in to the authentication method chosen in the User Console.

   ArcotID OTP

   For environments created in CA CloudMinder 1.51 or later:

   /chs/redirect/*tenant_tag*/arcototp

   For environments created before CA CloudMinder 1.51:

   /affwebservices/<tenant-name>/arcototp.jsp

   ArcotID OTP with Risk

   For environments created in CA CloudMinder 1.51 or later:

   /chs/redirect/*tenant_tag*/arcototprisk

   For environments created before CA CloudMinder 1.51:

   /affwebservices/<tenant-name>/arcototprisk.jsp

   ArcotID PKI

   For environments created in CA CloudMinder 1.51 or later:

   /chs/redirect/*tenant_tag*/arcotid

   For environments created before CA CloudMinder 1.51:

   /affwebservices/<tenant-name>/arcotid.jsp

   ArcotID PKI with Risk

   For environments created in CA CloudMinder 1.51 or later:

   /chs/redirect/*tenant_tag*/arcotidrisk

   For environments created before CA CloudMinder 1.51:

   /affwebservices/<tenant-name>/arcotidrisk.jsp

   tenant_tag is a unique identifier for a tenant. You specify the tag when deploying a tenant environment in the CSP console. To view a list of tags, select the Tenants tab.
9. Complete the remaining fields:

   Default Resource Protection

   Protected

   Authentication Scheme

   Select the authentication scheme that corresponds to the resource filter.

10. Create a rule as follows:
    a. Click Create in the Rules area.

       The Create Rule screen opens.

    b. Enter a name and description for the rule.

    c. Enter the asterisk (*) in the Resource field.

    d. Select Get and Post from the Action list.

    e. Accept the defaults for the remaining settings, and then click OK.

       The rule is created.

11. Specify the session properties.

    Note: Click Help for information about these properties.

12. Skip the other configuration options.

13. Click Finish.

    The realm is configured.

Add Rules to the Policy

Rules indicate which resources are part of a policy and whether to allow or deny access to the resources when the rule fires.

Note: Add at least one rule or rule group to a policy.

Follow these steps:

1. Select Policies, Domain, Domains.

   The Domains screen opens.

2. Click the pencil icon for the tenant domain.

3. Click the Policies tab.

4. Click the pencil icon for the *tenant_tag*_chsauthmethods_policy_es policy.

5. Click the Rules tab.

6. Perform the following steps for each rule that you want to add:
   a. Click Add Rule.

      The Available Rules pane opens.

   b. Select the rule that you created for the authentication URL resource, and then click OK.

      The rule is added to the tenant policy.

Configure an Authentication Scheme for the User Console

The *tenant-tag*_ims_realm realm represents the User Console. To secure access to the User Console, one of the steps that you perform is to apply the required authentication scheme to this realm. The remaining steps are performed in the User Console itself.

Note: Perform this procedure only for the User Console. You need not perform this procedure for any other application.

Follow these steps:

1. Log in to the CSP Console.
2. Select Policies, Domain, Realms.

   The Realms screen opens.
3. Use the search feature to search for and open the *tenant-tag*_ims_realm realm for modification.
4. Select the *tenant-tag*_idm_chs_auth authentication scheme from the Authentication Scheme list.
5. Do not change the value of any other field on this screen.
6. Click Submit.

   The authentication scheme is applied for securing access to the User Console.

Post the ArcotID OTP Client and ArcotID PKI Client

The native clients for ArcotID OTP and ArcotID PKI are available in the Support section of the CA Technologies website. If the tenant administrator wants to enable their end users to use these clients, inform the tenant administrator about the location from where they can download these clients. The tenant administrator can then post these clients on their website and make the clients available to their end users for download and installation.

Chapter 5: Troubleshooting Advanced Authentication Errors

This section provides solutions for problems that may occur while configuring and administering Advanced Authentication.

Note: For information about other errors that you may encounter while configuring and administering Advanced Authentication, see the *CA Arcot AuthMinder Administration Guide* and the *CA Arcot RiskMinder Administration Guide*.

CA AuthMinder Server Is Not Reachable (see page 54)

Database Is Not Reachable (see page 55)

CA AuthMinder Server Is Not Reachable

Symptom:

While I'm enabling, configuring, or disabling a credential type or an advanced authentication flow, the following message appears:

```
Unable to connect to AuthMinder
```

Solution:

The host computer for the CA AuthMinder Server may not be reachable from the CA AuthMinder administration console. Alternatively, the CA AuthMinder Server itself may not be running.

To troubleshoot this problem:

1. Verify that the host computers of the CA AuthMinder administration console and CA AuthMinder Server are reachable to each other. If they are not reachable, fix the connection issue between the two computers so that each computer can be pinged from the other computer.
2. Start the CA AuthMinder Server by performing the following steps:
   a. Navigate to the following directory in a command window on the CA AuthMinder Server host computer:
      installation_location/arcot/bin/
   b. Run the following command:
      ```
      ./webfortserver start
      ```
      The message that appears depends on whether the CA AuthMinder Server is already running:
      - If the CA AuthMinder Server is already running, the following message appears:
        ```
        Operation start being performed on Server WebFort Server
        WebFort Server already running.
        ```
      - If the CA AuthMinder Server is not running, it is started and the following message appears:
        ```
        Operation status being performed on Server WebFort Server
        All environment variables are set
        ```
3. Resume the operation that was interrupted when you encountered the error. For example, if you were configuring a credential type when you encountered the error, try configuring the credential type again.
4. Restart the CA AuthMinder Server by performing the following steps, if you encounter the same error again:
   a. Navigate to the following directory in a command window on the CA AuthMinder Server host computer:
      installation_location/arcot/bin/
   b. Run the following command:

```
./webfortserver stop
./webfortserver start
```

You can now resume the operation that was interrupted when you encountered the error. For example, if you were configuring a credential type when you encountered the error, try configuring the credential type again.

Database Is Not Reachable

Symptom:

While I'm enabling, configuring, or disabling a credential type or an advanced authentication flow, the following message appears:

```
Unable to connect to AuthMinder
```

Solution:

The host computer for the database may not be reachable from the CA AuthMinder administration console or from the CA AuthMinder Server. Alternatively, the database service itself may not be running.

To troubleshoot this problem:

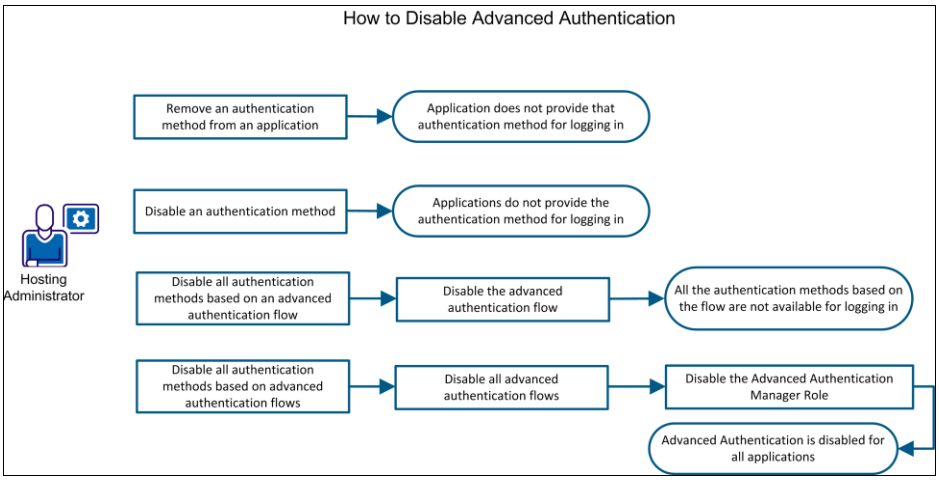
1. Verify that the host computers of the CA AuthMinder administration console, CA AuthMinder Server, and the database are reachable to each other. If they are not reachable, fix the connection issue between the computers so that each computer can be pinged from the other two computers.
2. Connect to the CA AuthMinder database server host computer, and check whether the related CA AuthMinder database service is running. If the database service is not running, then start it.

   You can now resume the operation that was interrupted when you encountered the error. For example, if you were configuring a credential type when you encountered the error, try configuring the credential type again.

How to Disable Advanced Authentication

Advanced authentication flows, authentication methods, and applications are components of Advanced Authentication in the User Console. When a tenant administrator sends a request to disable any one or a combination of these components, the effect of the procedure that you perform as the hosting administrator depends on the components that you disable.

The following diagram shows the various options for disabling all or individual Advanced Authentication components:



Depending on the configuration request from the tenant administrator, perform the required combination of the following tasks:

- Remove an Authentication Method from an Application (see page 56)
- Disable an Authentication Method (see page 57)
- Disable an Advanced Authentication Flow (see page 57)
- Disable the Advanced Authentication Manager Role (see page 58)

Remove an Authentication Method from an Application

Remove an authentication method from an application if you do not want to provide end users the option of using that authentication method to log in to the application. Other applications in which the same authentication method has been added would continue to provide the option of using that authentication method to log in.

Follow these steps:

1. Log in to the User Console.
2. Select Applications, Applications, Modify Application.
3. Enter a search string for the Application in which the authentication method has been added, and then click Search.

   The search results appear.
4. Select the Application, and then click Select.

   The Modify Application screen is displayed.
5. Click the icon (-) in the last column for removing the authentication method.

6. Click Submit.

   The authentication method is removed from the application.

Disable an Authentication Method

Disable an authentication method if you do not want any application to provide end users the option of using that authentication method for logging in.

Note: If you plan to disable an advanced authentication flow, first disable each authentication method that is based on that flow.

Follow these steps:

1. Log in to the User Console.
2. Select Applications, Authentication Methods, Modify Authentication Method.
3. Enter a search string for the authentication method that you want to disable, and then click Search.

   The search results appear.
4. Select the authentication method, and then click Select.

   The Modify Authentication Method screen opens.
5. Clear the Enabled check box, and click Submit.

   The Authentication Method is disabled.

More information:

Troubleshooting Advanced Authentication Errors (see page 53)

Disable an Advanced Authentication Flow

You can disable an advanced authentication flow. Before you disable an advanced authentication flow, you must disable all the authentication methods that are based on that flow.

Note: While disabling an advanced authentication flow, it is optional to disable the underlying credential types.

Follow these steps:

1. Log in to the User Console.
2. Select Advanced Authentication, Configure Advanced Authentication Flow.

   The Select Flow Types screen opens.
3. Use the left arrow icon to move the advanced authentication flow to the Disabled list.
4. Click Next.

   The Enabled Flow Types screen opens.
5. Click Finish.

   The advanced authentication flow is disabled.

Disable the Advanced Authentication Manager Role

The Advanced Authentication Manager role is one of the roles that control Advanced Authentication. To disable Advanced Authentication, disable all authentication methods that are based on Advanced Authentication, disable all advanced authentication flows, and then disable the Advanced Authentication Manager role. The outcome is that Advanced Authentication is not available to any application. No further configuration changes can be made to Advanced Authentication until you re-enable the role.

Follow these steps:

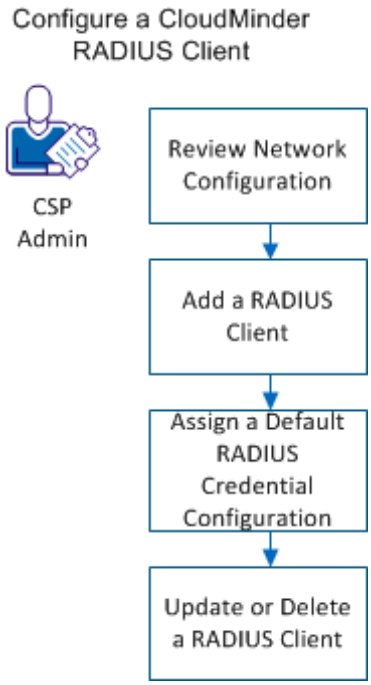
1. Log in to the User Console.
2. Select Admin Roles, Enable/Disable Admin Role.

   The Enable/Disable Admin Role screen opens.
3. Clear the Advanced Authentication Manager check box, and click Select.
4. Click Yes on the next screen that opens.

   The Advanced Authentication service is disabled.

How to Configure CA CloudMinder RADIUS Clients

CloudMinder 1.5 supports RADIUS. RADIUS offers two-factor authentication for VPN systems protected by CloudMinder. RADIUS is enabled by default, and you configure RADIUS clients as outlined in the following diagram:



As a prerequisite, configure the ArcotID OTP application to use the ArcotID OTP authentication type. To configure a CA CloudMinder RADIUS Client, complete the following tasks:

1. Review Network Configuration (see page 60)
2. Add a RADIUS Client (see page 61)
3. Assign a Default RADIUS Credential Type Resolution Configuration (see page 62)
4. Update or Delete a RADIUS Client (see page 63)

Review Network Configuration

Review this section before adding RADIUS clients and configuring a firewall and load balancer.

Network

Authentication Manager is not exposed outside the network. A proxy server runs on the web server, which forwards authentication requests. All requests must go through the proxy.

Ports

AuthMinder is on the app tier and listens on port 1812 for UDP traffic. The web-tier proxy server listens to client requests on 1812, and listens to AuthMinder responses on 1814. This information is important when configuring your firewall and load balancer.

Source NAT

If SNAT is enabled on the web-tier load balancer, each external IP of the VPN servers that sends requests to CA CloudMinder should be mapped to a unique, static, internal IP. The same internal IP should be used when you add RADIUS clients.

Add RADIUS Clients

You can add a RADIUS client for an organization from the Arcot Administration Console.

Follow these steps:

1. Log in to the Arcot Administration Console http://*<server name>*:9090/arcotadmin/adminlogin.htm as a global admin.
2. Click the Organizations tab, and search for the organization.
3. Select the organization, and click the Webfort Configuration tab.
4. From the left pane, click RADIUS Client.
5. From the main window, click Add.
6. Complete the following:

   RADIUS Client IP Address

   Specifies the IP Address of the RADIUS client through which users authenticate to AuthMinder Server.

   Shared Secret Key

   Specifies the secret key shared between the RADIUS client and the AuthMinder Server.

   Note: Keys must be between 1 and 512 characters.

   Description

   Specifies a short description of the RADIUS client. If you configure multiple clients, the description of each client helps distinguish between clients.

   Authentication Type

   Select In-Band Password.

7. In the RADIUS Retry Handling section, specify the following:
   - Select the Enable Retry option if you want the RADIUS client to retry sending the request to AuthMinder Server if it does not receive a response.
   - In the Retry Window field, enter the duration in seconds within which the client can retry connecting to the AuthMinder Server if it does not receive a response. After this period, the retry is considered invalid. Ensure that the retry window period is greater than the client timeout period.
8. In the Additional RADIUS Response Attributes section, specify the attributes that you want the AuthMinder Server to include in the response sent to the RADIUS client after successful authentication:

   Attribute ID

   Specify 224.

   Attribute Value

Specifies the value corresponding to the attribute ID. You can pass static values, such as user attributes or a combination of static values and variables. For example ,for the user JSmith, you can include the full name in RADIUS response as:

```
Name=$$LNAME$$,$$FNAME$$
```

to return:

```
224= [Name=Smith, John]
```

Note: The mapped attributes FNAME, LNAME, TELEPHONENUMBER, and EMAILADDR can be returned.

9. In the RADIUS Packet Drop Options section, select the event or events when AuthMinder Server must drop RADIUS packets.
10. Click Add.

    The RADIUS client is added.

Assign a Default RADIUS Credential Configuration

This section shows you how to assign a default RADIUS credential type resolution configuration.

Follow these steps:

1. Log in to the Arcot Administration Console as global admin.
2. Complete the following steps:
   a. Click the Organizations tab.
   b. Search for the organization.
   c. Select the organization from the search results.
   d. Click the Webfort Configuration tab.
3. From the left pane, click Assign Default Configuration.
4. For the field ArcotOTP-OATH Profile, select MobileArcotOTPProfile_<TENANT GUID>.
5. From ArcotOTP-OATH Policy drop down list, select MobileArcotOTPPolicy_<TENANT GUID>.
6. For the field RADIUS Credential Type Resolution Configuration, select VerifyArcotOTP-OATH.
7. Click Save.

   The default RADIUS credential type resolution configuration is assigned.

Update or Delete a RADIUS Client

If a RADIUS client is configured, the RADIUS Configuration page displays the configured clients in the Configured RADIUS Clients table. You can use this table to update or delete the RADIUS client IP addresses.

Follow these steps:

1. From the Arcot Administration Console, click the Organizations tab, and search for the organization.
2. Select the organization, and click the Webfort Configuration tab.
3. From the left pane, click RADIUS Client.
4. Log in to the Administration Console.
5. From the Configured RADIUS Clients section, select the IP address of the machine that requires updates.
6. Edit the fields as needed, and click the Update or Delete button.

Chapter 6: Advanced Authentication Flows

This section describes the high-level steps involved in configuring and using the Advanced Authentication service. It also describes the advanced authentication flows. It includes the following topics:

Overview

The Advanced Authentication service of CA CloudMinder provides various advanced authentication flows that cater to a tenant's business requirements. Each flow is used to secure access to a tenant's resource and define the authentication steps that take place when end users try to access the resource.

The Advanced Authentication service offers ArcotID PKI, ArcotID OTP, Security Code, and Risk Evaluation as primary credential types that can be used to secure access to a resource. An advanced authentication flow is based on either a single credential type or a combination of these credential types.

The Advanced Authentication service offers the following advanced authentication flows for the supported credential types:

- ArcotID PKI Only
- ArcotID PKI with Risk
- ArcotID OTP Only
- ArcotID OTP with Risk

This section provides detailed information about how each advanced authentication flow works.

Advanced Authentication Service Configuration

A hosting administrator configures specific CA CloudMinder services after understanding a tenant's requirements.

To configure the Advanced Authentication service, the hosting administrator performs the following high-level steps:

1. Enables Advanced Authentication.
2. Configures credential types.
3. Configures advanced authentication flows.
4. Configures the risk evaluation rules.
5. Provides the download locations for the ArcotID OTP Client and the ArcotID PKI Client software.

   Tenants can decide about how to push these clients to their end users devices.

For information about the advanced authentication-specific details that you must provide, see Information Required to Configure the Advanced Authentication Service.

End User Enrollment for Advanced Authentication

If an application or resource is protected using advanced authentication credentials, an end user who accesses the resource for the first time is prompted to enroll for the credentials. However, only existing users, that is, registered CA CloudMinder users, are prompted for advanced authentication credential enrollment.

The advanced authentication credential enrollment process for an end user is as follows:

1. On the login page for the resource, the end user is prompted for their user name and LDAP password.
2. If authentication is successful, the end user is prompted for the following information:
   - Email ID and phone number to verify the end user's identity.
   - (For ArcotID PKI only) Whether to trust the end user's device. If this option is selected, then the ArcotID PKI credential is downloaded to the device.
   - (For ArcotID OTP only) A PIN to be used during ArcotID OTP authentication.

The end user can now access the protected resource by providing the user name and LDAP password (or OTP if ArcotID OTP is used for authentication). In addition to the user name and password or OTP, the Advanced Authentication service also verifies the ArcotID PKI or ArcotID OTP credential.

The detailed authentication flows are described in the sections that follow.

How Advanced Authentication Flows Work

This section provides information about the back-end operations that take place when an end user tries to access a protected resource. In this section, ArcotID PKI is used as an example of the Advanced Authentication credential that can be used by an end user.

Assumptions:

This flow is based on the following assumptions:

- You have enabled the ArcotID PKI credential in the tenant console and configured the ArcotID PKI Only flow.
- You have configured the Credential Handling Service to protect the resource realm with the CA SiteMinder authentication scheme corresponding to the ArcotID PKI Only flow.
- The browser used for transactions is capable of supporting Java Applet and Native Client.
- JavaScript is enabled in the browser.
- An ArcotID PKI credential has been downloaded to the end user's device.

The Flow:

1. In a browser window, the end user attempts to access a protected resource.
2. The end user is directed to the Credential Handling Service page.
3. The end user clicks the ArcotID PKI Only Flow button.
4. The SiteMinder Web Agent takes control of the request and performs the following operations:
    a. Checks for an existing Single Sign-On (SSO) session, if any. If an SSO session is available, it grants access to the resource.
    b. If no SSO session is available, then the Web Agent interacts with the Policy Server. The Policy Server configuration indicates that ArcotID PKI is configured to protect the resource.
    c. The SiteMinder policy determines that because ArcotID PKI is configured as the primary authentication mechanism, user authentication must be performed by Advanced Authentication service component called Shim and hence passes the authentication request to Shim.
5. Shim creates a shared token in the Advanced Authentication Data Service (AADS), which is a component of the Advanced Authentication service and resides in the Application Tier. It interacts with the database on behalf of the Advanced Authentication components..

    The shared token is used for communication between Shim and the Advanced Authentication application, and it contains information about the transaction state, tenantID, and the authentication scheme.

6. Shim returns to the Web Agent the Advanced Authentication URL to which the browser must be redirected.

   The specific URL to which the end user is redirected is specified when the resource is protected at SiteMinder. Depending on the tenant's business requirements, this URL corresponds to one of the advanced authentication flows.

   Note: Each advanced authentication flow is supported by a different URL within the Advanced Authentication application.

7. The Web Agent redirects the browser to the Advanced Authentication URL.
8. In the Advanced Authentication application, the shared token is accessed by invoking the AADS. Thus, tenant and end user information is now known to the Advanced Authentication application.
9. The Advanced Authentication application looks up tenant configuration information, creates a page containing the tenant's logo and style settings, and displays it to the end user.
10. The end user enters their user name and LDAP password on the page and clicks Submit.
11. If the end user enters an invalid password, an error page is displayed prompting the user to enter the correct password.
12. If the end user enters a valid password:
    a. The ArcotID PKI Client signs this challenge using the end user's private key.
    b. The Advanced Authentication application performs the following operations:
       a. Sends the signed challenge to the Advanced Authentication Server for verification. If the signature is verified, a success message is sent to the Advanced Authentication application.
       b. Updates the shared token in the database indicating the authentication status.
       c. Redirects the browser to the FCC LANDING URL providing the end user's user name, and tokenID as the password.

          FCC pages are static HTML pages used by Shim to collect user inputs during authentication.
    c. SiteMinder Web Agent receives the redirection request. The Policy Server invokes Shim and provides the user name and password (tokenID).
    d. Shim performs the following operations:
       a. Requests for the transaction state from State Manager.
       b. Verifies the LDAP password.
       c. Validates that the authentication was performed and forwards the authentication result to the Policy Server.
    e. The Policy Server generates a SiteMinder cookie.

f. The SiteMinder Web Agent adds the cookie to the HTTP header and redirects the browser to the protected resource.

Advanced Authentication Flows

The advanced authentication flows are based on strong authentication, or a combination of strong authentication and risk evaluation. These flows can be categorized as follows:

- ArcotID PKI-Based Flows (see page 70)
- ArcotID OTP-Based Flows (see page 76)
- Risk Evaluation-Based Flows (see page 82)

ArcotID PKI-Based Flows

The Advanced Authentication service uses ArcotID PKI as one of the credentials to protect end users from identity theft and fraud. ArcotID PKI acts as the second factor ("something you have") for multifactor authentication and works behind the scenes to protect and verify user identities. End users authenticate by using their user name and password.

This section describes the following ArcotID-based flows:

- ArcotID PKI Only Flow (see page 70)
- ArcotID Mobile PKI Client Flow (see page 71)
- ArcotID PKI Roaming Download Flow (see page 72)
- Forgot Password Flow (see page 75)

ArcotID PKI Only Flow

Defines the flow to authenticate end users with their ArcotID PKI credential only. Use the ArcotID PKI Only flow if you want to use only the ArcotID PKI credential to secure access to a resource.

This flow is the same as that described in the How Advanced Authentication Flows Work (see page 68) section.

ArcotID Mobile PKI Client Flow

Defines the flow of authentication with the ArcotID Mobile PKI client from the tenant administrator and the end user perspectives.

A tenant administrator configures ArcotID Mobile PKI as follows:

1. The administrator logs in to the User Console, selects Advanced Authentication Types, Configure Credential Types, Configure Enabled Credentials, and Modify ArcotID Profile.
2. The administrator selects the Use mobile client box and saves the changes.
3. The administrator navigates to Configure Advanced Authentication Flow, Enabled Flow Types, Configure Flows, and selects ArcotID PKI only. The administrator can also enable a secondary authentication mechanism for the Mobile PKI client.
4. The administrator creates and enrolls the user.
5. The administrator instructs the user to download the application and authenticate.

An end user authenticates with ArcotID Mobile PKI as follows:

1. The user opens the application store on their mobile device and searches for ArcotID PKI.
2. The user installs the mobile application.
3. From the mobile browser, the user accesses the protected resource and follows the on-screen authentication process.

ArcotID PKI Roaming Flow

For end users who do not have the ArcotID PKI credential present on the device from which they are trying to access a protected resource, the Advanced Authentication service offers roaming capabilities. With this feature, end users first download the ArcotID PKI after successfully completing secondary authentication and then use the ArcotID PKI to authenticate themselves and access the protected resource.

A roaming user can be authenticated using knowledge-based question and answer pairs, or security code through SMS, email, or voice message. Each security code is generated by the Advanced Authentication Server, and it does not require any credential-specific information.

This section describes the steps for the ArcotID PKI Roaming Download flow using security questions, security code, or both for secondary authentication.

Note: For detailed information about the back-end operations that take place when an end user tries to access a protected resource, see How Advanced Authentication Flows Work (see page 68).

Assumptions:

This flow is based on the following assumptions:

- You have enabled the ArcotID PKI credential in the tenant console and configured the ArcotID PKI Only flow.
- You have configured the Credential Handling Service to protect the resource realm with the CA SiteMinder authentication scheme corresponding to the ArcotID PKI Only flow.
- You have enabled roaming for ArcotID PKI and configured the flow to use security questions, security code, or a combination of the two as the secondary authentication mechanism.
- In the case of security code, you have enabled the preferred credential delivery channels in the User Console.
- The end user's record in the database contains a valid email address or phone number to which the credential can be delivered.
- The browser used for transactions is capable of supporting Java Applet and Native Client.
- JavaScript is enabled in the browser.
- The end user is enrolled with Advanced Authentication but the ArcotID PKI credential is not present on the end user's device.

The Flow:

1. In a browser window, the end user attempts to access a protected resource.

2. On the login page, the end user enters their user name and password, and then clicks Submit.
3. CA SiteMinder verifies the end user's login credentials.
4. The ArcotID PKI Client checks for an ArcotID PKI for the provided user name but does not find it on the end user's device.
5. The Advanced Authentication application invokes the Advanced Authentication Server to retrieve the end user's ArcotID PKI.
6. If the user name exists in the database but if their ArcotID PKI is not available on the device being used, the user is challenged for secondary authentication. Depending on the secondary authentication mechanism, one of the following sequence of steps takes place:
   - If security question has been set as the secondary authentication mechanism, then:
     a. The Advanced Authentication application invokes IdentityMinder to retrieve the security questions.
        The page with challenge questions is presented to the end user. On the same page, the end user can specify whether the ArcotID PKI must be stored for future sessions.
     b. The end user submits answers to the security questions.
     c. The Advanced Authentication application invokes IdentityMinder to verify the answers.
   - If security code has been set as the secondary authentication mechanism, then:
     a. The Advanced Authentication application invokes the Advanced Authentication Server to generate a security code and fetch the end user's email address and/or phone number.
     b. If more than one delivery channel has been configured, then the end user selects a preferred channel.
     c. The Advanced Authentication application invokes the delivery channel.
     d. The Advanced Authentication application presents a page challenging the end user for the security code.
        On the same page, the end user can specify whether the ArcotID PKI must be stored for future sessions.
     e. The end user submits the security code.
     f. The Advanced Authentication application invokes the Advanced Authentication Server to verify the security code.
7. If the verification is successful, depending on whether two-step authentication is enabled, either of the following steps take place:
   - If two-step authentication is not enabled:
     a. The ArcotID PKI credential is downloaded to the end user's device.

b. The ArcotID PKI Client loads the ArcotID PKI.

- If two-step authentication is enabled:

a. The end user is presented with the second form of authentication, and is authenticated as described in Step 6.

Note: If security question was used the first time, then security code is used in this step. Conversely, if security code was used the first time, then security question is used in this step.

b. If the verification is successful:

- The ArcotID PKI credential is downloaded to the end user's device.
- The ArcotID PKI Client loads the ArcotID PKI.

Note: Two-step authentication is not enabled for authentication using the ArcotID PKI mobile client. When a mobile client is used, all configured authentication methods are used one after the other.

8. Upon downloading the credential, the browser displays the login page with the user name and challenges the end user for the password.
9. The end user enters the password and completes the rest of the authentication process.
10. If the authentication is successful, then the browser is redirected to SiteMinder with a success message.

Forgot Password Flow

End users who forget their LDAP password can choose to reset their password by answering secret questions, which they set during enrollment. After changing the password, a new ArcotID is placed on the end user's device.

Prerequisites:

This flow is based on the following configurations:

- The hosting administrator has enabled ArcotID PKI credential in the User Console and has configured the ArcotID PKI Only flow.
- The hosting administrator has configured the Credential Handling Service to protect the resource realm with the CA SiteMinder authentication scheme corresponding to the ArcotID PKI Only flow.
- The device used for transactions has ArcotID PKI native or mobile client installed or is capable of supporting Java Applet or JavaScript Client.
- An ArcotID PKI has been issued to the end user. The ArcotID PKI may or may not be present on the end user's device.

The Flow:

1. In a browser window, the end user attempts to access a protected resource.
2. On the login page, the end user specifies their user name and clicks the Forgot Password link.
3. The end user is prompted for secondary authentication, and the following steps take place:
    a. The Advanced Authentication application invokes IdentityMinder to retrieve the security questions.
    The page with challenge questions is presented to the end user. On the same page, the end user can specify whether the ArcotID PKI must be stored for future sessions.
    b. The end user answers the security questions.
    c. The Advanced Authentication application invokes IdentityMinder again to verify the answers.
4. The browser displays the login page with the user name and challenges the end user for the new password.

    The end user provides a new password.

Note: The behavior of this flow is also applicable in case a credential expires. The only difference is that the end user does not click on the "Forgot Password" link.

ArcotID OTP-Based Flows

The ArcotID OTP application is a secure software OTP generator which must be installed on the end user's device. To support a wide variety of application environments, the ArcotID OTP application is available in the form of a desktop client and a mobile application.

For users who do not want to manage the ArcotID OTP application on their device to generate OTPs, the Advanced Authentication service provides a JavaScript Client that invisibly runs in the end user's web browser and generates an OTP each time it is invoked.

This section describes the following ArcotID OTP-based flows:

- ArcotID OTP Only Flow (see page 77)
- ArcotID OTP Roaming Download Flow (see page 78)
- ArcotID OTP New Device Activation Flow (see page 81)
- Forgot My PIN Flow (see page 80)

ArcotID OTP Only Flow

This section lists the steps for ArcotID OTP authentication.

Note: For detailed information about the back-end operations that take place when an end user tries to access a protected resource, see How Advanced Authentication Flows Work (see page 68).

Prerequisites:

This flow is based on the following configurations:

- You have enabled the ArcotID OTP credential in the tenant console and configured the ArcotID OTP Only flow.
- You have configured the Credential Handling Service to protect the resource realm with the CA SiteMinder authentication scheme corresponding to the ArcotID OTP Only flow.
- The end user's smart phone or system has the ArcotID OTP application installed and the ArcotID OTP credential is provisioned to the phone or system.

The Flow:

1. In a browser window, the end user attempts to access a protected resource.
2. On the login page, the end user is prompted for their user name and OTP.
3. The end user accesses the ArcotID OTP application installed on their smart phone or system, authenticates to it with their PIN, and then generates an OTP.
4. The end user then returns to the login page in the browser, enters the user name and OTP, and clicks Submit.
5. The Advanced Authentication server verifies the OTP.
6. If OTP verification is successful, then the end user is granted access the resource.

ArcotID OTP Roaming Flow

This section lists the steps for ArcotID OTP roaming authentication.

Note: For detailed information about the back-end operations that take place when an end user tries to access a protected resource, see How Advanced Authentication Flows Work (see page 68).

Prerequisites:

This flow is based on the following configurations:

- You have enabled the ArcotID OTP credential in the tenant console and configured the ArcotID OTP Only flow.
- You have configured the Credential Handling Service to protect the resource realm with the CA SiteMinder authentication scheme corresponding to the ArcotID OTP Only flow.
- The end user's device does not have the ArcotID OTP application installed and the ArcotID OTP credential is not provisioned to the device.
- The end user's browser supports JavaScript Client.

The Flow:

1. In a browser window, the end user attempts to access a protected resource.
2. On the login page, the end user is prompted for their user name and OTP.
3. The end user clicks the Help icon next to the One Time Password field.

   The resulting help page provides three links to enroll for advanced authentication, reset PIN, and perform roaming authentication.
4. The end user clicks the My phone is unavailable link to perform roaming authentication.
5. On the resulting page, the end user is prompted for their user name.
6. If the user name is valid, the end user is prompted for secondary authentication using security question or security code.
7. If the authentication is successful, then depending on whether two-step authentication is enabled, either of the following steps take place:
   - If two-step authentication is not enabled, an ArcotID OTP credential associated with that end user is provisioned to the web browser store, and the end user is prompted for their PIN.
   - If two-step authentication is enabled:
     a. The end user is authenticated again using another form of secondary authentication.

Note: If security question was used the first time, then security code is used in this step. Conversely, if security code was used the first time, then security question is used in this step.

b. If the verification is successful, an ArcotID OTP credential associated with that end user is provisioned to the web browser store, and the end user is prompted for their PIN.

8. If the PIN is correct, a JavaScript client on the end user's device implicitly generates an OTP and sends it to the Advanced Authentication application.
9. The Advanced Authentication application invokes the Advanced Authentication Server to verify the OTP.
10. If the OTP verification is successful, then the browser is redirected to SiteMinder with a success message.

Forgot My PIN Flow

This section describes how end users who forget their PIN can reset it.

The flow described here is based on the following assumptions:

- An ArcotID PKI credential has been issued to the end user.
- The end user had set the PIN at the time of enrollment, but has forgotten it.

End users can reset their PIN as follows:

1. When trying to access a protected resource in a browser, the end user is prompted for their user name and OTP.
2. The end user, who has forgotten their PIN, specifies their user name and clicks the Help icon next to the One Time Password field.

   The resulting help page provides three links to enroll for advanced authentication, reset PIN, and perform roaming authentication.
3. The end user clicks the Forgot my PIN link.
4. On the resulting page, the end user is prompted for secondary authentication using the security question or security code mechanism.
5. The end user successfully completes the secondary authentication.
6. Depending on whether two-step authentication is enabled or not, either of the following steps take place:
   - If two-step authentication is not enabled, the end user is sent an activation email with a one-time password.
   - If two-step authentication is enabled:
     a. The end user is authenticated again using another form of secondary authentication.

     Note: If security question was used the first time, then security code is used in this step. Conversely, if security code was used the first time, then security question is used in this step.

     b. If the verification is successful, the end user is sent an activation email a one-time password.
7. The end user is prompted for this one-time password, after which they can set a new PIN and confirm the same.
8. On resetting their PIN, a new ArcotID OTP credential is placed on the end user's device.

   The end user will get mail with details to download the ArcotID OTP card in the ArcotID OTP client.
9. The end user must activate thier device again.
10. The end user is then taken back to the login page to proceed with authentication.

ArcotID OTP New Device Activation Flow

This section describes the flow to activate an end user's device for ArcotID OTP generation. The flow described here is based on the assumption that the device is a trusted device and the end user plans to use it for OTP generation.

An end user can activate a new device for ArcotID OTP generation in either of the following ways:

- Selecting CA ArcotID OTP Enrollment from the User Console.

  Note: This option is available to users that an administrator has given privileges of the role Advanced Authentication Self Manager.

- Directly on the Modify Security Settings page of the User Console.
- By raising a request with the tenant administrator. The tenant administrator then adds the device from the User Console.

An end user activates a new device for OTP generation as follows:

1. An end user logs in to the User Console and:
   - Selects CA ArcotID OTP Enrollment if avaialable, or
   - Accesses the Modify Security Settings page.
2. The user selects the Generate Activation Code check box.

   An email is sent to the end user with the activation code and instructions to configure their new device for OTP generation.
3. The end user follows the instructions and configures their new device.

A tenant administrator activates a new device for OTP generation as follows:

1. The tenant administrator:
   - Assigns users the priveleges of the Advanced Authentication Self Manager role and instructs users to select CA ArcotID OTP Enrollment in the User Console.
   - The administrator completes the following steps:

   a. Log in to the User Console and navigate to the Arcot OTP Mobile Activation page in the Advanced Authentication section.

   b. Search for the user who requested for activation of their new device, and clicks Select.

   c. On the resulting screen, the tenant administrator selects the Generate Activation Code check box.

      An email is sent to the end user with the activation code and instructions to configure their new device for OTP generation.
2. The end user follows the instructions and configures their new device.

Risk Evaluation-Based Flows

The Advanced Authentication service provides real-time protection against fraud in online transactions. When an end user tries to access a protected resource, the Advanced Authentication service can gather data about the end user and the device being used, evaluate the risk from the incoming request, generates a risk score, and provide the relevant authentication advice. If the advice suggests increased authentication, the end user's identity can be validated using security questions or OTP-based secondary authentication.

This section describes the following Risk evaluation-based flows:

- ArcotID PKI with Risk and ArcotID OTP with Risk (see page 83)
- ArcotID OTP Roaming with Risk Flow (see page 85)
- ArcotID PKI Roaming with Risk Flow (see page 88)

ArcotID PKI with Risk and ArcotID OTP with Risk

This section discusses the following flows:

- ArcotID PKI with Risk Flow
- ArcotID OTP with Risk Flow

In these flows, when an end user attempts to access a protected resource, they first authenticate themselves using the ArcotID PKI, or ArcotID OTP credential and are then assessed for potential risks.

Prerequisites:

This flow is based on the following configurations:

- The hosting administrator has enabled the Risk Evaluation credential type, and configured the ArcotID PKI with Risk flow, or the ArcotID OTP with Risk flow.
- The hosting administrator has configured multiple secondary authentication mechanisms.
- The hosting administrator has configured SiteMinder to protect the resource with one of the authentication schemes corresponding to the advanced authentication flow that was configured.

The Flow:

1. In a browser window, the end user attempts to access a protected resource.
2. On the login page, the end user is prompted for the following information:
   - If ArcotID PKI is used, then the user name and LDAP password.
   - If ArcotID OTP is used, then the user name and OTP.

     In this case, the end user generates an OTP using their ArcotID OTP application and uses that OTP for authentication.
3. The end user enters their user name and password or OTP and clicks Submit.
4. If the authentication is successful, then the Advanced Authentication application analyzes the risk associated with the login attempt as follows:
   a. The Advanced Authentication application looks up tenant flow configuration information and returns a page containing a DeviceDNA script with the tenant's preferences passed in.
   b. The script running in the browser collects the DeviceID information from the cookie, extracts the DeviceDNA data according to the tenant's configuration setting, and posts the results to the Advanced Authentication application.
   c. The Advanced Authentication application validates the DeviceID and DeviceDNA with the Advanced Authentication Server.
   d. If the Advanced Authentication Server returns a DENY advice, then:

- The Advanced Authentication application displays an error message indicating that the authentication failed.
- The Advanced Authentication application updates the token in AADS with the status indicating that the authentication failed, user message, risk score and other transaction state as required.

e. If the Advanced Authentication Server returns an ALLOW advice, then the Advanced Authentication application updates the token in AADS indicating successful authentication, risk score, and other transaction state as required.

   The user is allowed to access the protected resource.

f. If the Advanced Authentication Server returns an Increased Authentication advice, then secondary authentication is performed as described in ArcotID PKI Roaming Flow (see page 72) or ArcotID OTP Roaming Flow (see page 78).

g. If authentication is successful, then the Advanced Authentication application creates a token in AADS indicating successful authentication, risk score, and other transaction state as required.

   If the end user fails the secondary authentication challenge, then the Advanced Authentication application updates the token in AADS indicating failed authentication status, user message, risk score, and other transaction state as required.

ArcotID OTP Roaming with Risk Flow

This section describes the authentication and risk flow for an end user who is enrolled but is using a different device to which the ArcotID OTP credential has not been provisioned.

The end user is authenticated as follows:

1. When trying to access a protected resource in a browser, the end user is prompted for the user name and OTP.
2. The end user clicks the Help icon next to the One Time Password field.

   The resulting help page provides three links to enroll for advanced authentication, reset PIN, and perform roaming authentication.
3. The end user clicks the My phone is unavailable link to perform roaming authentication.
4. On the resulting page, the end user is prompted for their user name.
5. If the user name is valid, the end user is prompted for secondary authentication using security question or security code.
6. If the user is authenticated successfully, depending on whether two-step authentication is enabled, either of the following steps takes place:
   - If two-step authentication is not enabled, an ArcotID OTP credential associated with that end user is provisioned to the web browser store, and the end user is prompted for their PIN.
   - If two-step authentication is enabled:
     a. The end user is authenticated again using another form of secondary authentication.

     Note: If security question was used the first time, then security code is used in this step. Conversely, if security code was used the first time, then security question is used in this step.

     b. If the verification is successful, an ArcotID OTP credential associated with that end user is provisioned to the web browser store, and the end user is prompted for their PIN.
7. If the PIN is correct, a JavaScript client on the end user's device implicitly generates an OTP and sends it to the Advanced Authentication service.
8. The Advanced Authentication service verifies the details and authenticates the user.

9. If authentication is successful, the Advanced Authentication service performs a risk check as follows:

   a. A JavaScript that is running in the browser does the following:

      - Checks whether a DeviceID has been recorded on the device.
      - Extracts DeviceDNA from the device to identify the device.
      - Sends this information back to the Advanced Authentication service without requiring any user inputs.

   b. The Advanced Authentication service validates the DeviceID and DeviceDNA using the configured risk rules. It then generates a risk advice.

   c. Depending on the risk advice, one of the following happens:

      - If the Advanced Authentication service returns an ALLOW advice, then the end user is granted access to the resource.
      - If the Advanced Authentication service returns an INCREASEAUTH advice, the end user is prompted for secondary authentication. If secondary authentication (described in steps 5 and 6) is successful, the end user is granted access to the resource.
      - If the Advanced Authentication service returns a DENY advice, then an error message is displayed indicating that the authentication failed.

      Notes:

      - For every time that secondary authentication is invoked in a flow, one or more secondary authentication mechanisms are exhausted. Therefore, for a flow that requires more than one round of secondary authentication, ensure that you enable as many secondary authentication mechanisms as possible. An error occurs if secondary authentication is invoked and no mechanism left.
      - If two-step authentication is enabled, and if one of the authentication methods overlaps for the roaming and risk flows, when the end user chooses that common method in the first flow and authenticates successfully, that authentication method is skipped in the next flow.

        For example, if security question or security code over email is enabled for roaming authentication, and security question or security code over SMS is enabled for risk authentication, and if the end user selects security question first and is authenticated successfully, they are not authenticated again during the risk flow. However, if the end user selects security code over email the first time and is authenticated successfully, then in the risk flow, the user is authenticated again using security question.

In another example where security question or security code over email is enabled for roaming authentication, and security question **and** security code over SMS are enabled for risk authentication, if the end user selects security question in the roaming flow and is authenticated successfully, then in the risk flow, the security code over SMS method is invoked. However, if the end user selects security code over email in the roaming flow, then both security question and security code over SMS are invoked in the risk flow.

A risk cookie is placed on the end user's device. During subsequent logins, the risk history is used to decide whether to grant access to the end user after authentication.

ArcotID PKI Roaming with Risk Flow

This section describes the authentication and risk flow for an end user who is enrolled but is using a different device to which the ArcotID PKI credential has not been provisioned.

The Flow:

1. In a browser window, the end user attempts to access a protected resource.
2. On the login page, the end user enters their user name and password, and then clicks Submit.
3. CA SiteMinder verifies the end user's login credentials.
4. The ArcotID PKI Client checks for an ArcotID PKI for the provided user name but does not find it on the end user's device.
5. The Advanced Authentication application invokes the Advanced Authentication Server to retrieve the end user's ArcotID PKI.
6. If the user name exists in the database but if their ArcotID PKI is on a different device, the user is challenged for secondary authentication. Depending on the secondary authentication mechanism, one of the following sequence of steps takes place:
   - If Security Question has been set as the secondary authentication mechanism, then:
     a. The Advanced Authentication application invokes IdentityMinder to retrieve the security questions.
        The page with challenge questions is presented to the end user. On the same page, the end user can specify whether the ArcotID PKI must be stored for future sessions.
     b. The end user submits answers to the security questions.
     c. The Advanced Authentication application invokes IdentityMinder to verify the answers.
   - If Security Code has been set as the secondary authentication mechanism, then:
     a. The Advanced Authentication application invokes the Advanced Authentication Server to generate a security code and fetch the end user's email address and/or phone number.
     b. If more than one delivery channel has been configured, then the end user selects a preferred channel.
     c. The Advanced Authentication application invokes the delivery channel.
     d. The Advanced Authentication application presents a page challenging the end user for the security code.
        On the same page, the end user can specify whether the ArcotID PKI must be stored for future sessions.

e. The end user submits the security code.

f. The Advanced Authentication application invokes the Advanced Authentication Server to verify the security code.

7. If the verification is successful, depending on whether two-step authentication is enabled, either of the following steps take place:

   - If two-step authentication is not enabled:
   - The ArcotID PKI credential is downloaded to the end user's device.

     a. The ArcotID PKI Client loads the ArcotID PKI.

   - If two-step authentication is enabled:

     a. The end user is presented the second form of authentication, and is authenticated as described in Step 6.

     Note: If security question was used the first time, then security code is used in this step. Conversely, if security code was used the first time, then security question is used in this step.

     b. If the verification is successful:

     - The ArcotID PKI credential is downloaded to the end user's device.
     - The ArcotID PKI Client loads the ArcotID PKI.

8. Upon downloading the credential, the browser displays the login page with the user name and challenges the end user for the password.

9. The end user enters the password and completes the rest of the authentication process.

10. If authentication is successful, the Advanced Authentication service performs a risk check as follows:

    a. A JavaScript that is running in the browser does the following:

    - Checks whether a DeviceID has been recorded on the device.
    - Extracts DeviceDNA from the device to identify the device.
    - Sends this information back to the Advanced Authentication service without requiring any user inputs.

    b. The Advanced Authentication service validates the DeviceID and DeviceDNA using the configured risk rules. It then generates a risk advice.

    c. Depending on the risk advice, one of the following happens:

    - If the Advanced Authentication service returns an ALLOW advice, then the end user is granted access to the resource.
    - If the Advanced Authentication service returns an INCREASEAUTH advice, the end user is prompted for secondary authentication. If secondary authentication (described in steps 6 and 7) is successful, the end user is granted access to the resource.

- If the Advanced Authentication service returns a DENY advice, then an error message is displayed indicating that the authentication failed.

Notes:

- For every time that secondary authentication is invoked in a flow, one or more secondary authentication mechanisms are exhausted. Therefore, for a flow that requires more than one round of secondary authentication, ensure that you enable as many secondary authentication mechanisms as possible. An error occurs if secondary authentication is invoked and no mechanism left.
- If two-step authentication is enabled, and if one of the authentication methods overlaps for the roaming and risk flows, when the end user chooses that common method in the first flow and authenticates successfully, that authentication method is skipped in the next flow.

  For example, if security question or security code over email is enabled for roaming authentication, and security question or security code over SMS is enabled for risk authentication, and if the end user selects security question first and is authenticated successfully, they are not authenticated again during the risk flow. However, if the end user selects security code over email the first time and is authenticated successfully, then in the risk flow, the user is authenticated again using security question.

  In another example where security question or security code over email is enabled for roaming authentication, and security question **and** security code over SMS are enabled for risk authentication, if the end user selects security question in the roaming flow and is authenticated successfully, then in the risk flow, the security code over SMS method is invoked. However, if the end user selects security code over email in the roaming flow, then both security question and security code over SMS are invoked in the risk flow.

A risk cookie is placed on the end user's device. During subsequent logins, the risk history is used to decide whether to grant access to the end user after authentication.

Chapter 7: Information Required to Configure the Advanced Authentication Service

To configure the Advanced Authentication service for a tenant organization, the tenant must provide the following information to the hosting administrator:

This section describes the information that is required from the tenant to configure the Advanced Authentication service. It also lists the default values used for certain configurations.

Selected Credential Types and Authentication Flows

As a first step, the tenant must decide on the types of credentials and flows to be enabled to protect different applications and resources.

A hosting administrator would require the following information from the tenant:

- Primary credential types to be enabled. The available options are as follows:
  - ArcotID PKI
  - ArcotID OTP
  - Risk Evaluation
  - Security Code
- Advanced Authentication flows to be enabled, and the secondary authentication mechanisms to be enabled for each flow. A tenant can use multiple secondary authentication mechanisms for a flow.

  The available flows are as follows:
  - ArcotID PKI Only
  - ArcotID PKI with Risk
  - ArcotID OTP Only
  - ArcotID OTP with Risk

  The available secondary authentication mechanisms are as follows:
  - Security Question
  - Security Code over Email
  - Security Code over SMS
  - Security Code over Voice

  In addition to selecting the mechanisms, the tenant can also specify whether two-step authentication must be enforced for a particular scenario.

  As secondary authentication is typically invoked when performing sensitive tasks, it is recommended that a combination of these authentication mechanisms be chained together for enhanced security.

  Note: For information about the Security Question mechanism (question and answer pairs), see CA IdentityMinder documentation.

ArcotID PKI-Specific Details

A tenant must provide the following information to configure the ArcotID PKI credential:

- The length (in bits) of the credential.

  Example: 1024

- Whether the use of an ArcotID PKI mobile client for authentication must be allowed.
- The expiry date for the ArcotID PKI credential.

  Example: 1 day

- ArcotID PKI client preference. This is the order of preference for the mechanism used to deliver the ArcotID PKI to the user's computer. The available options are JavaScript client and Native client.

  Example: JavaScript client.

ArcotID OTP-Specific Details

ArcotID OTP Credential

A tenant must provide the following information to configure the ArcotID OTP credential:

- The minimum length (in bits) of the OTP PIN. Available options are 4 to 32.

  Example: 5

- The expiry date for the ArcotID OTP credential.

  Example: 1 day

Activation OTP-Specific Details

During ArcotID OTP authentication, a roaming user who chooses to configure the ArcotID OTP application on their device is prompted for secondary authentication. If the end user selects Security Code over Email as the secondary authentication method, the end user is sent an email with a security code. The end user must specify this OTP on the login page to proceed with the ArcotID OTP application configuration.

A tenant must provide the following details to configure the activation OTP to be used with the ArcotID OTP credential:

- Whether the activation OTP must be of numeric or alphanumeric type.

  Example: Numeric

- The length (in bits) of the activation code.

  Example: 5

- The period for which the activation OTP is valid.

  Example: 1 day

- The number of times for which an activation OTP can be used.

  Example: 3

Security Code-Specific Details

A tenant must provide the following details to configure the profile of the Security Code credential type:

- Whether the credential must be only numeric or alphanumeric.

  Example: Numeric

- The length (in characters) of the credential.

  Example: 6

- The validity period (in seconds) of the credential. The credential automatically expires at the end of this validity period.

  Example: 30

- The number of failed attempts after which the credential is locked.

  Example: 3

Security Code over Email

To send the credential to end users by email, a tenant must provide the following details:

- Tenant's email address to be shown in the email that is sent to end users.
- The subject line to be shown in the email that is sent to end users.
- The message to be shown in the email that is sent to end users. The credential is included in this message.

Security Code over SMS

To send the credential to end users by SMS, a tenant must provide the following details:

- The template of the SMS message that is sent to end users. The credential is included in this message.
- The URL from where the SMS service can be accessed.
- The user name of the SMS account that has been created for the tenant.
- The password of the SMS account that has been created for the tenant.
- The text that must be displayed in the From field of the SMS message that is sent to end users.

Security Code over Voice

To send the credential to the end user as voicemail, a tenant must provide the following details:

- The client ID that has been assigned to the tenant.

- The country ID of the tenant's organization.

Selected Risk Evaluation Rules

A tenant who decides to use the Risk Evaluation credential type must also specify the risk evaluation rules that they want to enable. The available rules are:

- Exception User Check
- Untrusted IP Check
- Negative Country Check
- Trusted IP/Aggregator Check
- User Known Check
- DeviceID Known Check
- User Associated with DeviceID Check
- Device MFP Match Check
- Device Velocity Check
- User Velocity Check
- Zone Hopping Check

In addition to selecting risk evaluation rules, the tenant must provide information to configure the selected rules, as described in Risk Evaluation-Specific Details.

Risk Evaluation-Specific Details

To enable risk evaluation, the required risk evaluation rules must be configured. Some rules are configured with default values when the Advanced Authentication service is installed and set up. The remaining ones are configured by the CA CloudMinder administrator based on the information received from the tenant.

A tenant must provide the following details to configure risk evaluation:

- Whether the cookie that stores risk-related information on the end user's device must be a Flash-based cookie or an HTTP-based cookie.

  Example: HTTP-based cookie

- The maximum number of days after which the HTTP cookie must automatically expire.

  Example: 1 day

- Depending on the risk evaluation rules that a tenant has decided to use, some or all the following information must be provided:

  - **Untrusted IP Check**: A list or range of IP addresses that the tenant organization does no trust.

    These could be IP addresses that have been the origin of known anonymizer proxies or fraudulent and malicious transactions in the past.

  - **Negative Country Check**: A negative country list, which comprises all countries from which fraudulent or malicious transactions are known to have originated in the past.

  - **Trusted IP/Aggregator Check**:

    - A list or range of IP addresses that the tenant organization trusts.

      Transactions that originate from or are routed through these IP addresses are considered low risk. As a result, the Advanced Authentication service bypasses these transactions from risk evaluations and assigns them a low score and the ALLOW advice.

    - A list of trusted aggregator IP addresses.

      Transactions that originate from or are routed through aggregators “trusted” to the organization are considered low-risk. As a result, the Advanced Authentication service bypasses these transactions from risk evaluations and assigns them a low score and the ALLOW advice.

  - **User Velocity Check**: The number of transactions to evaluate per end user, and the time period inside which to perform risk evaluations.

    Example: 5 end user transactions evaluated in 60 minutes

  - **Device Velocity Check**: The number of transactions to evaluate per device, and the time period inside which to perform risk evaluations.

    Example: 10 transactions evaluated per device in 60 minutes

- **Zone Hopping Check**:
  - The maximum speed (S, in miles per hour) at which a user can physically travel using conventional transport, such as airplanes, cars, and trains.

    If the speed at which a user appears to have moved (in the time frame between two successive transactions) exceeds this pre-configured threshold speed (S), then the Advanced Authentication service considers it a case of zone hopping.

    Example: 500 miles
  - The maximum number of users who can share the same user name.

    This setting enables multiple users (for example, husband and wife) to use the same user name though they might be located in different zones without the fear of being considered risk.

    Example: 1
  - An uncertainty offset to accommodate the variation in the physical location of the IP address from which the transaction originated.

    This value is required as there may be a variation in the location of the IP address provided by ISPs. A user's physical location (geographic latitude and longitude) cannot be determined to a high level of precision by using their public IP address.

    Example: 50 miles
- **Device MFP Match Check**: A threshold value for the MFP match.

  The Advanced Authentication service checks whether the match percentage between the input device signature and the corresponding stored device signature is greater than or equal to this threshold value.

  Example: 50