THALES

SafeNet Authentication Client

LINUX ADMINISTRATOR GUIDE

Document Information

Product Version	10.8 (GA)
Document Number	007-013842-002
Release Date	June 2021

Revision History

Revision	Date	Reason
A	June 2021	Updated for 10.8 (GA) release

Trademarks, Copyrights, and Third-Party Software

Disclaimer

All information herein is either public information or is the property of and owned solely by Thales DIS France S.A. and any of its subsidiaries who shall have and keep the sole right to file patent applications or any other kind of intellectual property protection in connection with such information.

Nothing herein shall be construed as implying or granting to you any rights, by license, grant or otherwise, under any intellectual and/or industrial property rights of or concerning any information of Thales DIS France S.A. and any of its subsidiaries and affiliates (collectively referred to herein after as “Thales”).

This document can be used for informational, non-commercial, internal, and personal use only provided that:

- The copyright notice, the confidentiality and proprietary legend and this full warning notice appear in all copies.
- This document shall not be posted on any publicly accessible network computer or broadcast in any media, and no modification of any part of this document shall be made.

Use for any other purpose is expressly prohibited and may result in severe civil and criminal liabilities.

The information contained in this document is provided “AS IS” without any warranty of any kind. Unless otherwise expressly agreed in writing, Thales makes no warranty as to the value or accuracy of information contained herein.

The document could include technical inaccuracies or typographical errors. Changes are periodically added to the information herein. Furthermore, Thales reserves the right to make any change or improvement in the specifications data, information, and the like described herein, at any time.

Thales hereby disclaims all warranties and conditions with regard to the information contained herein, including all implied warranties of merchantability, fitness for a particular purpose, title and non-infringement. In no event shall Thales be liable, whether in contract, tort or otherwise, for any indirect, special or consequential damages or any damages whatsoever including but not limited to damages resulting from loss of use, data, profits, revenues, or customers, arising out of or in connection with the use or performance of information contained in this document.

Thales does not and shall not warrant that this product will be resistant to all possible attacks and shall not incur, and disclaims, any liability in this respect. Even if each product is compliant with current security standards in force on the date of their design, security mechanisms' resistance necessarily evolves according to the state of the art in security and notably under the emergence of new attacks. Under no circumstances, shall Thales be held liable for any third party actions and in particular in case of any successful attack against systems or equipment incorporating Thales products. Thales disclaims any liability with respect to security for direct, indirect, incidental or consequential damages that result from any use of its products. It is further stressed that independent testing and verification by the person using the product is particularly encouraged, especially in any application in which defective, incorrect or insecure functioning could result in damage to persons or property, denial of service, or loss of privacy.

CONTENTS

PREFACE: About this Document

This document describes the operational and administrative tasks you can perform to maintain the functionality and efficiency of your SafeNet Authentication Client.

This section also identifies the audience, explains how to best use the written material, and discusses the documentation conventions used. They are:

> "Audience" below
> "Document Conventions" below
> "Support Contacts" on page 8

For information regarding the document status and revision history, see "Document Information" on page 2.

Audience

This document is intended for personnel responsible for maintaining your organization's security infrastructure. This includes SafeNet Authentication Client users and administrators.

All products manufactured and distributed by Thales Group are designed to be installed, operated, and maintained by personnel who have the knowledge, training, and qualifications required to safely perform the tasks assigned to them. The information, processes, and procedures contained in this document are intended for use by trained and qualified personnel only.

It is assumed that the users of this document are proficient with security concepts.

Document Conventions

This section describes the conventions used in this document.

Command Syntax and Typeface Conventions

This document uses the following conventions for command syntax descriptions, and to highlight elements of the user interface.

Format	Convention
bold	The bold attribute is used to indicate the following: > Command-line commands and options that you enter verbatim (Type **dir /p**.) > Button names (Click **Save As**.) > Check box and radio button names (Select the **Print Duplex** check box.) > Dialog box titles (On the **Protect Document** dialog box, click **Yes**.) > Field names (**User Name**: Enter the name of the user.) > Menu names (On the **File** menu, click **Save**.) (Click **Menu** > **Go To** > **Folders**.) > User input (In the **Date** box, type **April 1**.)
italics	In type, the italic attribute is used for emphasis or to indicate a related document. (See the *Installation Guide* for more information.)
<variable>	In command descriptions, angle brackets represent variables. You must substitute a value for command line arguments that are enclosed in angle brackets.
[**optional**] [<optional>]	Represent optional **keywords** or <variables> in a command line description. Optionally enter the keyword or <variable> that is enclosed in square brackets, if it is necessary or desirable to complete the task.

{**a**|**b**|**c**}
{<a>|<b>|<c>}
	Represent required alternate **keywords** or <variables> in a command line description. You must choose one command line argument enclosed within the braces. Choices are separated by vertical (OR) bars.

[**a**|**b**|**c**] [<a>|<b>|<c>]	Represent optional alternate keywords or variables in a command line description. Choose one command line argument enclosed within the braces, if desired. Choices are separated by vertical (OR) bars.

Notifications and Alerts

Notifications and alerts are used to highlight important information or alert you to the potential for data loss or personal injury.

Tips

Tips are used to highlight information that helps to complete a task more efficiently.

TIP This is some information that will allow you to complete your task more efficiently.

Notes

Notes are used to highlight important or helpful information.

NOTE Take note. Contains important or helpful information.

Cautions

Cautions are used to alert you to important information that may help prevent unexpected results or data loss.

> **CAUTION!** Exercise caution. Contains important information that may help prevent unexpected results or data loss.

Warnings

Warnings are used to alert you to the potential for catastrophic data loss or personal injury.

> ****WARNING**** **Be extremely careful and obey all safety and security measures. In this situation you might do something that could result in catastrophic data loss or personal injury.**

Support Contacts

If you encounter a problem while installing, registering, or operating this product, please refer to the documentation before contacting support. If you cannot resolve the issue, contact your supplier or Thales Customer Support.

Thales Customer Support operates 24 hours a day, 7 days a week. Your level of access to this service is governed by the support plan arrangements made between Thales and your organization. Please consult this support plan for further information about your entitlements, including the hours when telephone support is available to you.

Customer Support Portal

The Customer Support Portal, at https://supportportal.thalesgroup.com, is where you can find solutions for most common problems. The Customer Support Portal is a comprehensive, fully searchable database of support resources, including software and firmware downloads, release notes listing known problems and workarounds, a knowledge base, FAQs, product documentation, technical notes, and more. You can also use the portal to create and manage support cases.

> **NOTE** You require an account to access the Customer Support Portal. To create a new account, go to the portal and click on the **REGISTER** link.

Telephone

The support portal also lists telephone numbers for voice contact (Contact Us).

CHAPTER 1: Introduction

SafeNet Authentication Client (SAC) is a middleware client that manages Thales's extensive SafeNet portfolio of certificate-based authenticators, including eToken, IDPrime smart cards, USB and software based devices.

With full backward compatibility and incorporating features from previous middleware versions, SafeNet Authentication Client ensures complete support for all currently deployed eToken devices, as well as IDPrime and .NET smart cards.

Overview

SAC is a Public Key Infrastructure (PKI) middleware that provides a secure method for exchanging information based on public key cryptography, enabling trusted third-party verification of user identities. It utilizes a system of digital certificates, Certificate Authorities, and other registration authorities that verify and authenticate the validity of each party involved in an internet transaction.

The SafeNet Authentication Client Tools application and the SafeNet Authentication Client tray icon application are installed with SafeNet Authentication Client, providing easy-to-use configuration tools for users and administrators.

> **NOTE** The term *Token* is used throughout the document and is applicable to both Smart Cards and USB Tokens.

For SAC system requirement details and compatibility information, see *SafeNet Authentication Client Release Notes*.

SAC Password Quality Information

SafeNet Authentication Client supports password quality settings for Administrator passwords (also known as Security Officer (SO) passwords) and Initialization keys that are implemented by SafeNet Authentication Client software. The setting is the same for all devices and cannot be modified. Though, it can be switched off for backward compatibility.

Additionally, IDPrime supports the insertion of the Administrator Key directly (without derivation), in which case the password policy is not validated. The Administrator Key derivation method is proprietary and may vary depending on the device.

The Administrator password quality and Initialization Key quality must include three out of the following four rules:

1. English uppercase letters (ASCII 0x41...0x5A)
2. English lowercase letters (ASCII 0x61...0x7A)
3. Numeric (ASCII 0x30...0x39)
4. Special characters (ASCII 0x20...0x2F + 0x3A...0x40 + 0x5B...0x60 + 0x7B...0x7F)

For backward compatibility, the Administrator Password quality check can be switched off via the SAC `pqAdminPQ` property.

Initialization key password quality check cannot be switched off.

> **NOTE** The Password quality is in use only when the Administrator Password and Initialization keys are used in a ‘Friendly’ (textual) format. For more information, see the 'Friendly Admin Password' section in the *SafeNet Authentication Client User Guide*.
>
> eToken 5110 FIPS and eToken 5110 devices support only *Friendly Admin* passwords.
>
> If a customer does not want to be compliant with these PIN Quality policies, use hexadecimal keys (also via SAC UI and SAC API). Friendly Admin PIN length can be 24 binary or 48 hexadecimal. The Initialization Key length can be 32 binary or 64 hexadecimal. In this case, the keys are used as-is (without derivation) and PIN Quality is not checked.

SAC supports password quality settings for the User PIN. The implementation of these settings may differ on various devices. User PIN policies are created or modified during a device's initialization process or during the device's life cycle after Administrator (SO) authentication.

Depending on the device model (for example: IDPrime or eToken devices) and initialization mode that is set (for example: the device is initialized without password policies), password quality policies are enforced by the device or by the middleware software (SAC).

Device Type	Where the policy is stored:	Policy is enforced by:
eToken 5110 GA eToken 5110 FIPS	Depends on how the device is formatted: On board SAC configuration	Middleware
IDPrime MD 840/3840 SafeNet IDPrime 940/3940 eToken 5110 CC	On board	Middleware (except for the PIN length, which is validated on board)
IDPrime MD 830/3811 SafeNet IDPrime 930/3930 eToken 5300	On board	On board

> **NOTE** Each device (IDPrime / eToken) has a different policy setting. For more information, see the Token Settings chapter in *SafeNet Authentication Client User Guide*.

The SAC Client Settings policy is currently used only on eToken 5110 GA and 5110 FIPS. This policy is used in the following cases:

- The device is initialized without on board policies
- The default values used during the device initialization flow

PIN Retry Counter

Setting the Administrator/User PIN Retry Counter may vary depending on your device type:

Administrator PIN Retry Counter

- **Gemalto IDPrime MD 840** - The Administrator PIN Retry Counter cannot be modified on this device.
- **SafeNet IDPrime 940/3940** - The Administrator PIN Retry Counter is supported. The parameter is configured during factory settings and therefore, cannot be modified.
- **Gemalto IDPrime MD 830 B / SafeNet IDPrime 930/3930** - The Administrator PIN Retry Counter is supported. The parameter can be modified using SAC.

User PIN Retry Counter

SafeNet eToken 5110 FIPS or SafeNet eToken 5110 - Due to an eToken applet limitation, the User Retry Counter cannot be set on these smart cards, unless they are initialized.

Collecting SAC Logs

Collecting SAC logs allows administrators and technical-support personnel to diagnose the source of many problems that may have occurred while working with SafeNet Authentication Client. This information is used for debugging purposes.

SAC logs are collected by the following method:

- SAC GUI (SAC Tools)

Perform the following steps to enable SAC logs through SAC GUI (SAC Tools):

1. Open **SAC Tools** > **Advanced View** > **Client Settings**, and click the **Advanced** tab.
2. Click **Enable Logging**.

   The button will change to: Disable Logging. (For more information, see 'Enable Logging' in *SafeNet Authentication Client User Guide*.
3. Restart the application that requires the debug logs to be created.

> **NOTE** SAC Log files are created in the following directory `/temp/eToken.log`.

CHAPTER 2: Common Criteria

IDPrime Common Criteria Profile

The IDPrime Applet 4.0 is Common Criteria certified on Common Criteria based smart cards and tokens. See the *SafeNet Authentication Client Release Notes* for a list of supported smart cards and tokens. These devices can have certain parameters customized in the factory with values that differ from the default profile.

> **NOTE** The IDPrime MD 840/ 3840 cards or eToken 5110 CC do not support modifying the retry counter on the Admin Key. The recommended workaround is to set the profiles with a PUK instead of the Admin Key.
> To ensure maximum security, when using friendly mode, set the password with at least 16 random printable characters.

The following parameters can be customized:

- Number and type of key containers
- Support of RSA 4,096-bit key containers (import operation only).

> **NOTE** The card needs to be configured by the SAC supported key length.

- PINs (#1, #3 and #4 only)
- Try Limit
- Unblock PIN (PIN#1 only)
- PIN validity period
- Secure messaging in contactless mode

Number and Type of Key Containers

The list below are the default settings. For other options consult your Thales representative.

By default, the IDPrime Applet 4.0 is pre-personalized with:

- 2 X 2,048-bit CC Sign Only RSA Keys
- 2 X 1,024-bit Standard Sign and Decrypt RSA Keys
- 8 X 2,048-bit Standard Sign and Decrypt RSA Keys
- 2 X 256-bit Standard Sign and Decrypt EC Keys

By default, the IDPrime Applet 4.4.2 is pre-personalized with:

- 2 X 2048-bit CC Sign Only RSA Keys
- 2 X 4096-bit CC Sign Only RSA Keys

- 2 X 256-bit CC Sign Only ECC Keys
- 8 X 2048-bit CC Sign and Decrypt RSA Keys
- 2 X 1024-bit CC Sign and Decrypt RSA Keys
- 2 X 4096-bit CC Sign and Decrypt RSA Keys
- 2 X 1024-bit CC Sign and Decrypt ECC Keys

> **NOTE** The Key Generation method for CC key containers is either OBKG or Key import.

Common Criteria API Adjustments

Below table provides a high-level description of the adjustments that are made to the Standard and Extended PKCS#11 API to work with IDPrime CC devices. For more detailed information, see the code samples.

Standard PKCS#11 API	Extended PKCS#11 API
The `C_InitToken` function must receive the current Security Officer (SO) Password	The `C_InitToken` function must receive the current Security Officer (SO) Password
> When the `C_InitToken` function is called, you can enable link mode on the IDPrime CC device. See "Configuration Properties" on page 26. > To revert a device back to unlinked mode after it was initialized in linked mode, use the PKCS#11 Extended API, or by using SAC Tools initialization process.	> To initialize the IDPrime CC device, the `ETCKA_CC` attribute must be set to `CK_TRUE`. > To initialize a device in linked mode, set the `ETCKA_IDP_CC_LINK` attribute to `1`. > To pass the current Digital Signature PUK value, use the `ETCKA_IDP_CURRENT_PUK` attribute. > To revert a device back to unlinked mode after it was initialized in linked mode, set the `ETCKA_IDP_CC_LINK` attribute to 0 and use the `ETCKA_PUK` attribute to set the new Digital Signature PUK value.
If a device is not configured to use linked mode, the `C_InitToken` function ignores the Digital Signature PUK and Digital Signature PIN.	If a device is not configured to use linked mode, use the `ETCKA_PUK`attribute to set the new Digital Signature PUK value.

Standard PKCS#11 API	Extended PKCS#11 API
> After the device has been initialized in linked mode, the `C_InitPIN` function initializes the Digital Signature PIN and the User PIN. Both PIN's are set to the same value. > The `C_SetPIN` function used with the `CKU_SO` flag changes both the Administrator PIN and Digital Signature PUK to a new value. For details on Friendly Admin Password, see *SafeNet Authentication Client User Guide*. > The `C_InitPIN` function used with the `CKU_USER` flag changes both the User PIN and Digital Signature PIN to a new value.	If the device is initialized to use linked mode, the `C_InitPIN` function and `C_SetPIN` function behaves the same as described in the *Standard PKCS#11* section.

SafeNet eToken Devices vs Thales IDPrime Devices

Below table displays the differences between SafeNet eToken devices and Thales IDPrime devices.

Feature	eToken 5110, eToken 5110 FIPS (and all other eToken based devices)	IDPrime , IDPrime .Net, eToken 5110 CC
Initialization	3 Roles (Initialization key, Admin PIN, User PIN)	2 Roles (Admin PIN and User PIN)
	Device erased by using the Initialization key	Device is cleared by using the Admin PIN (no changes are made to the scheme)
	Initialization key is used only for initializing the device	If the Admin PIN is locked, the device cannot be cleared
Profile	Dynamic profile that allows an unlimited number of keys depending on the devices memory capacity	FIPS based devices - Dynamic profile limited to 15 key containers
		CC based devices - Static profile defined by perso
Password Policy	Off-Board (saved on token)	On-Board
	Full UTF-8 character encoding capabilities supported	Only ASCII character codes supported
Enhanced Security Mode	Support Propriety RSM mode	Support Secure Key Injection (through IDGo800 Minidriver)

Feature	eToken 5110, eToken 5110 FIPS (and all other eToken based devices)	IDPrime , IDPrime .Net, eToken 5110 CC
On Board RSAPadding (PSS/OAEP)	Not supported	Supported
Common Criteria	Deprecated	4 Roles (Admin PIN, User PIN, Digital Signature PIN, Digital Signature PUK).
	Digital Signature PIN is derived from the User PIN and the Digital Signature PUK is derived from the Administrator PIN	**Linked mode** - User PIN and Digital Signature PIN are identical and Digital Signature PUK is derived from Admin PIN. **Unlinked mode** - Each role has a different value.
	Appropriate Athena CC certified Applet for CC keys	Thales CC certified Applet
Symmetric Key operations	Support 3DES and AES	Not supported
Protocol for Contact	Support T1	Support T1, T0 and CTL

CHAPTER 3: Installation

This chapter provides the installation procedures for SafeNet Authentication Client (SAC) 10.8 (GA) Linux. Local administrator rights are required to install or uninstall it.

Installation Files

The software package provided includes files for installing or upgrading to SAC 10.8 (GA) Linux . The following Linux installation and documentation files are provided:

File	Description
Installation Files	
`GPG-KEY-SafenetAuthenticationClient.txt`	> This file is the public key (GnuPG. > The signature confirms that the package is signed by an authorized party and also confirms the integrity and origin of your file. > Use this file to verify the signature before installing them to ensure that they are not altered from the original source of the packages.
`SafenetAuthenticationClient-10.8.xx-1.el[x].x86_64.rpm`	> Installs SafeNet Authentication Client core on 64-bit platform. > Installs eToken core library and IFD Handler.
`safenetauthenticationclient_10.8.xx_amd64.deb`	> Installs SafeNet Authentication Client core on 64-bit platform. > Installs eToken core library and IFD Handler.
Documentation Files	

File	Description
`007-013841-003_SafeNet Authentication Client_10.8_Linux_GA_Release_Notes`	SafeNet Authentication Client Release Notes. Read before installation for last minute updates that may affect installation; contains important information such as resolved and known issues and troubleshooting for Linux.
`007-013843-002_SafeNet Authentication Client_10.8_Linux_GA_User_Guide`	SafeNet Authentication Client User Guide. Provides detailed information for the user and system administrator regarding the use of SafeNet Authentication Client for Linux.
`007-013842-002_SafeNet Authentication Client_10.8_Linux_GA_Administrator_Guide`	SafeNet Authentication Client Administrator Guide (this document). Provides detailed information for the system administrator regarding the installation, configuration, maintenance, and management of SafeNet Authentication Client for Linux.

Installing the Standard Package

Installing on Red Hat Enterprise, SUSE, CentOS, and Fedora

The installation package for SAC on Red Hat, SUSE, CentOS, and Fedora is the RPM Package. RPM is an installation file that can install, uninstall, and update software packages.

> **NOTE** For the PKCS#11, module to be installed automatically on a Firefox browser during the SAC installation, make sure that the *nss-tools* package is installed prior to installing SAC.

> On SUSE, Fedora, Centos, and Red Hat operating systems, in cases where the *nss-tools* package is not installed, install it as a privileged user by running the following command:

```
yum install nss-tools
```

> **NOTE** As the SAC tray icon is visible only in GNOME Classic(x11) desktops such as Red Hat Enterprise Linux v8 and CentOS v8, you need to perform some steps (below) to view it.

To view SAC tray icon for GNOME Classic(x11) desktops

Perform the following steps:

1. Install the following packages:
   a. `gnome-shell-extension-top-icons`
   b. `gnome-tweaks`
2. Run the **Tweaks** application to enable Top icons in the extensions.

Following is the SAC `.rpm` package name:

> `SafenetAuthenticationClient-10.8.xx-1.el[x].x86_64.rpm`

Where: `xx` is the build number

To install from the terminal

Perform the following steps:

1. On the terminal, log on as a root user.
2. Run the following command to import the public key:
   ```
   rpm --import GPG-KEY-SafenetAuthenticationClient.txt
   ```
3. Run the following command:
   ```
   rpm -Uvh SafenetAuthenticationClient-10.8.xx-1.el[x].x86_64.rpm
   ```
   Where: `xx` is the version number
4. Run the following command to verify the signature of RPM package:
   ```
   rpm --checksig --verbose SafenetAuthenticationClient-10.8.xx-1.el[x].x86_
   64.rpm
   ```

Installing on Ubuntu (.deb)

The installation packaging for SAC running on Ubuntu is the Debian software package (`.deb`).

Following is the SAC `.deb` package name:

> `safenetauthenticationclient_10.8.xx_amd64.deb`

Where: `xx` is the build number

To install from the package installer

Perform the following steps:

1. Double-click the relevant `.deb` file.

   The package installer is displayed.
2. Click **Install Package**.

   A password prompt appears.
3. Enter the Super User or root password.

   The installation process runs.

4. To run SafeNet Authentication Client Tools, do one of the following:
   - From the taskbar, select **Applications** > **SafeNet Authentication Client**.
   - Right-click the **SafeNet Authentication Client** tray icon, and select **Tools**.

   The **SafeNet Authentication Client Tools** window is displayed.

> **NOTE** Log out and log back in to enable the tray icon menu in the notification area.

To install from the terminal

Perform the following steps:

1. Enter the following command:

   ```
   sudo dpkg -i safenetauthenticationclient_10.8.xx_amd64.deb
   ```

   Where: `xx` is the build number

   A password prompt appears.
2. Enter the password.

   The installation process runs.
3. If the installation fails due to a lack of dependencies, enter the following command:

   ```
   sudo apt-get install -f
   ```

   The dependencies are installed and the installation continues.
4. To run SafeNet Authentication Client Tools, do one of the following:
   - From the taskbar, select **Applications** > **SafeNet Authentication Client**.
   - Right-click the **SafeNet Authentication Client** tray icon, and select **Tools**.

   The **SafeNet Authentication Client Tools** window is displayed.
5. Run the following command to import the public key:

   ```
   gpg --import GPG-KEY-SafenetAuthenticationClient.txt
   ```
6. Run the following command to verify the signature of `.deb` package:

   ```
   dpkg-sig --verify safenetauthenticationclient_10.8.xx_amd64.deb
   ```

> **NOTE** Ensure you log out and log back in to see the tray icon menu.

Installing the Core Package

Installing on Red Hat Enterprise, SUSE, CentOS and Fedora

The installation package for SAC running on RedHat and CentOS is the RPM Package Manager. RPM is a command line package management system that can install, uninstall, and update software packages.

Following is the SAC `.rpm` package name:

> `SafenetAuthenticationClient-core-10.8.xx-1.el[x].x86_64.rpm`

Where: `xx` is the build number

To install from the package installer

Perform the following steps:

1. Double-click the relevant `.rpm` file.

   The package installer is displayed.

2. Click **Install Package**.

   A password prompt appears.

3. Enter the Super User or root password.

   The installation process runs.

To install from the terminal

Perform the following steps:

1. On the terminal, log on as a root user.
2. Run the following command:

   ```
   rpm --import GPG-KEY-SafenetAuthenticationClient.txt
   ```

3. Run the following command:

   ```
   rpm -hi SafenetAuthenticationClient-core-10.8.xx-1.el[x].x86_64.rpm
   ```

   Where: `xx` is the build number

4. Run the following command to check the signature of RPM package:

   ```
   rpm --checksig --verbose SafenetAuthenticationClient-core-10.8.xx-1.el
   [x].x86_64.rpm
   ```

Installing on Ubuntu (.deb)

The installation packaging for SAC running on Ubuntu is the Debian software package (`.deb`).

> **NOTE**
> - When installing from the user interface with a user that is not an administrator, the following message is displayed:
> *The package is of bad quality*.
> Click **Ignore and Install**, and continue with the installation.
>
> - After installing SAC on Ubuntu, log off, and then log back on in order for the SAC monitor to run, and to display the tray icon.

Following is the SAC `.deb` package name:

> `safenetauthenticationclient-core_10.8.xx_amd64.deb`

Where: `xx` is the build number

To install from the package installer

Perform the following steps:

1. Double-click the relevant `.deb` file.

   The package installer is displayed.

2. Click **Install Package**.

   A password prompt appears.

3. Enter the Super User or root password.

   The installation process runs.

To install from the terminal

Perform the following steps:

1. Enter the following command:

   ```
   sudo dpkg -i safenetauthenticationclient-core_10.8.xx_amd64.deb
   ```

   Where: `n` is the version number

   A password prompt appears.

2. Enter the password.

   The installation process runs.

3. If the installation fails due to a lack of dependencies, enter the following:

   ```
   sudo apt-get install -f
   ```

   The dependencies are installed and the installation continues.

4. Run the following command to import the public key:

   ```
   gpg --import GPG-KEY-SafenetAuthenticationClient.txt
   ```

5. Run the following command to verify the signature of `.deb` package:

   ```
   dpkg-sig --verify safenetauthenticationclient-core_10.8.xx_amd64.deb
   ```

Linux External Dependencies

Red Hat Enterprise, SUSE, CentOS, Fedora and Ubuntu

- Prerequisite - SAC 10.8 (GA) requires OpenSSL 1.0 or higher to be installed
- PCSC (Smart Card Resource manager): `libpcsclite1,pcscd`
  - To install on Ubuntu- Run `sudo apt-get install libpcsclite1 pcscd`
  - To install on Red Hat/CentOS/Fedora- Run `yum install pcsc-lite`

Installing the Firefox Security Module

When SAC is installed, it does not install the security module in Firefox. This must be done manually.

Perform the following steps to install the security module in Firebox:

1. Open **Firefox Settings** > **Privacy & Security** > **Certificates**.
2. Click **Security Devices**.

The **Device Manager** window is displayed.

3. Click **Load**.

   The **Load PKCS#11 Device** window is displayed.

4. In the **Module Filename** field, enter the following string:
   - **On Ubuntu**: `/usr/lib/libeTPkcs11.so`
   - **On Red Hat, Fedora, CentOS**: `/usr/lib64/libeTPkcs11.so`

> **NOTE**
> - To work with CC devices in unlinked mode, enter the following string for Multi-Slot support:
> **For Ubuntu:** `/usr/lib/libIDPrimePKCS11.so`
> **For Red Hat, Fedora, CentOS:** `/usr/lib64/libIDPrimePKCS11.so`
>
> - For information on how to work with Multi-Slots, see the PKCS#11 Digital Signature PIN Authentication section of the *SafeNet Authentication Client User Guide*.

   The **Confirm** window is displayed.

3. Click **OK**.

   The new security module is installed.

Installing the Thunderbird Security Module

When SAC is installed, it does not install the security module in Thunderbird. This must be done manually.

Perform the following to install the security module in Thunderbird:

1. Select **Thunderbird** > **Preferences** > **Privacy & Security**.
2. On the **Certificate** tab, click **Security Devices**.

   The **Device Manager** window is displayed.

3. Click **Load**.

   The **Load PKCS#11 Device** window is displayed.

4. In the **Module Filename** field enter the following string:
   - **On Ubuntu**: `/usr/lib/libeTPkcs11.so`
   - **On Red Hat, Fedora, CentOS**: `/usr/lib64/libeTPkcs11.so`

> **NOTE**
> - To work with CC devices in unlinked mode, enter the following string for Multi-Slot support:
> **For Ubuntu:** `/usr/lib/libIDPrimePKCS11.so`
> **For Red Hat, Fedora, CentOS:** `/usr/lib64/libIDPrimePKCS11.so`
>
> - For information on how to work with Multi-Slots, see the PKCS#11 Digital Signature PIN Authentication section of the *SafeNet Authentication Client User Guide*.

The **Confirm** window is displayed.

3. Click **OK**.

   The new security module is installed.

CHAPTER 4: Uninstall

After SafeNet Authentication Client (SAC) 10.8 (GA) Linux is installed, you can uninstall it. Local administrator rights are required to uninstall SAC.

When SAC is uninstalled, user configuration and policy files may be deleted.

Uninstalling the Standard Package

Before uninstalling SAC 10.8 (GA) Linux, make sure that SafeNet Authentication Client Tools is closed.

Uninstalling on Red Hat Enterprise, SUSE, CentOS, and Fedora

Perform the following step:

1. In the console, enter the following:

   ```
   rpm -e SafenetAuthenticationClient-10.8.xx-0.x86_64.rpm
   ```

   Where: `-e` is the parameter for uninstalling.

Uninstalling on Ubuntu (.deb)

Perform the following step:

1. In the console, enter the following:

   ```
   sudo dpkg --purge safenetauthenticationclient
   ```

   Where: `--purge` is the parameter for uninstalling.

Uninstalling the Core Package

Uninstalling on Red Hat Enterprise, SUSE, CentOS and Fedora

Perform the following step:

1. In the console, enter the following:

   ```
   rpm -e SafenetAuthenticationClient-core-10.8.xx-0.x86_64.rpm
   ```

   Where: `-e` is the parameter for uninstalling.

Uninstalling on Ubuntu (.deb)

Perform the following step:

1. In the console, enter the following:

   ```
   sudo dpkg --purge safenetauthenticationclient-core
   ```

Where: `--purge` is the parameter for uninstalling.

CHAPTER 5: Configuration Properties

SafeNet Authentication Client (SAC) properties are stored on the computer as `ini` files, which can be added and changed to determine SAC behavior. Depending on where an `ini` value is written, it applies globally, or limited to a specific user/application.

> **NOTE** All properties are set and edited manually.

General Settings

The following settings are written to the **General** section in the file `/etc/eToken.conf`.

> **NOTE** On a Linux machine, *PcscSlots* and *SoftwareSlots* configuration keys determine the number of slots. The *Reader Settings* window in SafeNet Authentication Client Tools, displays the configured slots but does not allow the user to change the settings.

Description	Value
Unlock Authorization Activates authorization protection for SAC Tools Unlock feature.	**Value Name:** UnlockAuthorization **Value:** > 0 - Do not activate authorization protection > 1 - Activate authorization protection **Default:** 0
Read Only Mode Prevents deletion of certificates from the Token. **NOTE** When a user deletes certificates on a Firefox browser and this property is set to **Selected**, Firefox displays these certificates as deleted when in fact they are not.	**Value Name:** ReadOnlyMode **Values:** > 0 (Disabled) - Any user with the right permission can delete the certificates and their associated keys. > 1 (Enabled) - Certificates and their associated keys cannot be deleted. **Default:** 0

Description	Value

Multi-Slot Support
> Determines if SAC is backward compatible with PKCS#11 Common Criteria devices (IDPrime MD 840, IDPrime MD 3840 and eToken 5110 CC).
> The Multi-Slot feature affects only SAC customized in compatible mode through `libIDPrimePKCS11.so`.
Following are the two ways to work with IDPrime MD 840/940 or CC Cards, where a login is required for the Digital Signature Role:
1. Use `libIDPrimePKCS11.so`, where the user has two smart cards:
a. Physical Smart Card
b. Virtual Smart Card (where Digital Signature Role is exposed as ROLE1 in the virtual smart card
2. To enable the prompt login through a flag in the `/etc/eToken.conf` file, add the following line to Section [GENERAL]:
`[GENERAL]`
`EnablePrompt=1`
This allows C_Login with Null or when a ROLE is not Logged in, a prompt is shown to enter the PIN/Password to complete the operation, such as C_SIGN / C_Encrypt/C_Decrypt ....
For more information on Multi-Slots, see the PKCS#11 Digital Signature PIN Authentication section of the *SafeNet Authentication Client User Guide*.
NOTE Linked Mode is not compatible with the Multi-Slot feature.
	Value Name: MultiSlotSupport
	Values:
	> Selected - Activates this feature
	> Not Selected - Normal operation
	Default: Not Selected

Touch Sense Notify Determines if the Touch Sense notification is displayed as balloon or in a window.	**Value Name:** TSNotify **Values:** > 0 - Show window > 1 - Show balloon > 2 - No notification **Default:** 1 (Show balloon)

Description	Value
PIN Pad Notify Determines if the Pin Pad notification is displayed as balloon or in a window.	**Value Name:** PinPadNotify **Values:** > 0 - Show window > 1 - Show balloon > 2 - No notification **Default:** 0 (Show window)
Full SM Mode > Enables/disables the full Security Messaging (SM) mode for IDPrime FIPS L2 devices. **NOTE** SAC cache must be reset after changing the *FullSMMode* property. > This configuration is for applet 4.3.5 L2 cards only.	**Value Name:** FullSMMode **Values:** > 0 (False) - Disabled > 1 (True) - FIPS L2 only **Default:** 0 (Disabled)
No Pin Pad Determines whether or not the PIN Pad reader is used as a regular smart card reader. SAC UI will require entering user credentials.	**Value Name:** NoPinPad **Values:** > 0 - Disabled > 1 - Enabled **Default:** 0 (Disabled)
ITI Certification Mode Enables ITI Certification, which requires the following: > Administrator and User Passwords must be changed at first logon. > If initialization is performed without changing the Administrator and User Passwords at first logon, the Administrator Password is required for the initialization process. **NOTE** When the *ITI Certification Mode* property is enabled, the *Enable Administrator Password Quality Check* property will be disabled.	**Value Name:** MustChangeAdmin **Values:** > 0- None > 1 - ITI certification mode > 2 - Special administrator PIN policy **Default:** 0

Description	Value
Software Slots Defines the number of virtual readers for SafeNet Virtual Tokens. **NOTE** Can be modified in *Reader Settings* in SafeNet Authentication Client Tools also.	**Value Name:** SoftwareSlots **Values:** >=0 (0 = SafeNet Virtual Token is disabled; only physical tokens are enabled) **Default:** 2
PCSC Slots Defines the total number of PC/SC slots for all USB tokens and smart cards. Included in this total: > The number of allocated readers for third-party providers. > The number of allocated readers for other SafeNet physical tokens, which can be modified in *Reader Settings* in SafeNet Authentication Client Tools.	**Value Name:** PcscSlots **Values:** >=0 (0 = Physical tokens are disabled) **Default:** 8
Legacy Manufacturer Name > Determines if *Aladdin Knowledge Systems Ltd.* is written as the manufacturer name in token and token slot descriptions. > Use for legacy compatibility only.	**Value Name:** LegacyManufacturerName **Values:** > 1 - The legacy manufacturer name is written > 0 - The new manufacturer name is written **Default:** 0
Enable Private Cache > Determines if SAC allows the token's private data to be cached. > Applies only to tokens that are initialized with the private data cache setting. > The private data is cached in per process memory. **NOTE** Can be set in SafeNet Authentication Client Tools	**Value Name:** EnablePrvCache **Values:** > 1 (True) - Private data caching is enabled > 0 (False) - Private data caching is disabled **Default:** 1 (True)

Description	Value
Tolerate Finalize Determines if C_Finalize can be called by DllMain. **NOTE** Define this property per process. Select this setting when using Novell Modular Authentication Service (NMAS) applications only.	**Value Name:** TolerantFinalize **Values:** > 1 (True) - C_Finalize can be called by DllMain > 0 (False) - C_Finalize cannot be called by DllMain **Default:** 0 (False)
Tolerate X509 Attributes Determines if CKA_SERIAL_NUMBER, CKA_SUBJECT, and CKA_ISSUER attributes can differ from those in CKA_VALUE during certificate creation. **NOTE** Enable TolerantX509Attributes when using certificates created in a non-DER encoded binary x.509 format. In some versions of PKI Client, this setting is not selected by default.	**Value Name:** TolerantX509Attributes **Values:** > 1 (True) - The attributes can differ > 0 (False) - Check that the values match **Default:** 0 (False)
Tolerate Find Templates Determines if PKCS#11 tolerates a *Find* function with an invalid template, returning an empty list instead of an error.	**Value Name:** TolerantFindObjects **Values:** > 1 (True) - A Find function with an invalid template is tolerated and returns an empty list > 0 (False) - A Find function with an invalid template is not tolerated and returns an error **Default:** 0 (False)

Description	Value
Disconnect SafeNet Virtual Token on Logoff Determines if SafeNet Virtual Tokens are disconnected when the user logs off.	**Value Name:** EtvLogoffUnplug **Values:** > 1 (True) - Disconnect SafeNet Virtual Token when logging off > 0 (False) - Do not disconnect SafeNet Virtual Token when logging off **Default:** 0 (False)
Protect Symmetric Keys Determines if symmetric keys are protected. **NOTE** If selected, even non-sensitive symmetric keys cannot be extracted.	**Value Name:** SensitiveSecret **Values:** > 1 - Symmetric keys cannot be extracted > 0 - Symmetric keys can be extracted **Default:** 0
Cache Marker Timeout Determines if SAC Service periodically inspects the cache markers of connected tokens for an indication that token content has changed. A common usage of this property is while using remote sessions or crossing between the machines.	**Value Name:** CacheMarkerTimeout **Values:** > 1 - Connected tokens' cache markers are periodically inspected > 0 - Connected tokens' cache markers are never inspected **Default:** 0
Override Non-Repudiation OIDs > Overrides SAC's list of standard certificate OIDs that require a high level of security. **NOTE** Users must log on to their tokens whenever signing with a certificate defined as non-repudiation. > Avoid authenticating every time when a cryptographic operation is required for certificates containing *Entrust certificate OID* details, remove the default registration key value.	**Value Name:** NonRepudiationOID **Value:** Empty **Default:** No override

Description	Value
Ignore Silent Mode Determines if the *Token Logon* window is displayed even when the application calls the CSP/KSP in silent mode.	**Value Name:** IgnoreSilentMode **Values:** > 1 (True) - Display the Token Logon window even in silent mode > 0 (False) - Respect silent mode **NOTE** Set to True when the SafeNet RSA KSP must use SHA-2 to enroll a CA private key to a token **Default:** 0 (False)

Initialization Settings

> **NOTE** The following new settings are applicable to IDPrime Cards only:
> - In **Init** section: `ForceInitExternalPinPolicy` and `ForceDefaultInitKey`
> - In **InitApp** section: `HideInitCreateAdmin` and `HideInitPinPolicy`

The following settings are written to the **Init** section in the file `/etc/eToken.conf`.

> **NOTE** None of the settings in this section are relevant to IDPrime MD cards, except for the *LinkMode* and *UserMaxRetry* settings.
>
> Properties relevant to end of life tokens and cards can be found in previous versions of the Administrator Guide.

Description	Value
Always Use Default Initialization Key Defines the use of default initialization key during token initialization. **NOTE** If Selected, the following windows on the SAC Tools UI are skipped while Initializing IDPrime FIPS Devices (with initialization key): -Administrator Logon -Initializing Key Settings	**Value Name:** ForceDefaultInitKey **Values:** > 1: Token is initialized with the default initialization key > 0: Token is initialized with the initialization key entered by the user **Default:** > 0

Description	Value
Use PIN Quality Parameters From Policy Defines if the PIN Quality parameters in the SAC Client Settings are used during initialization. **NOTE** If Selected, user cannot modify the Pin Policy of the card manually through *Initialize Token* setting. Also, all the fields in the *PIN Quality* and *Advanced* tabs on the SAC Tools are disabled.	**Value Name:** ForceInitExternalPinPolicy **Values:** > 1: Token is initialized with PIN Quality parameters stored in SAC Client Settings > 0: Token is initialized with PIN Quality parameters stored on the card or entered by the user **Default:** > 0
Maximum Token Password Retries Defines the default number of consecutive failed logon attempts that lock the token.	**Value Name:** UserMaxRetry **Values:** 1-15 **Default:** 15
Maximum Administrator Password Retries Defines the default number of consecutive failed administrator logon attempts that lock the token.	**Value Name:** AdminMaxRetry **Values:** 1-15 **Default:** 15
Force SO object on Token	**Value Name:** ForceAdmin **Values:** > 1(True) - Token is initialized with SO object > 0 (False) - Token is initialized without SO object **Default:** 1 (True)
Force User object on Token	**Value Name:** ForceUser **Values:** > 1(True) - Token is initialized with User object > 0(False) - Token is initialized without User object **Default:** 1(True)

Description	Value
Legacy Format Version Defines the default token format.	**Value Name:** Legacy-Format-Version **Values:** > 0 - Tokens are formatted as backwardly compatible to eToken PKI Client 3.65 (CardOS tokens only) > 4 - Tokens are not formatted as backwardly compatible, and password quality settings can be saved on the token (CardOS tokens only) > 5 - Format includes new RSA behavior that is not controlled by key size; each key is created in a separate directory (CardOS 4.20B FIPS or Java Card-based tokens only) **Default:** > 4 - For CardOS tokens > 5 - For 4.20B FIPS and Java Card - based tokens
Default Token Name Defines the default Token Name written to tokens during initialization.	**Value Name:** DefaultLabel **Value:** String **Default:** My Token
API: Keep Token Settings When initializing the token using the SDK, determines if the token is automatically re-initialized with its current settings. **NOTE** If selected, this setting overrides all other initialization settings.	**Value Name:** KeepTokenInit **Values:** > 1 (True) - Use current token settings > 0 (False) - Override current token settings **Default:** 0 (False)

Description	Value
Automatic Certification When initializing the token using the SDK. If the token has FIPS or Common Criteria certification, the token is automatically initialized with the original certification.	**Value Name:** Certification **Values:** > 1(True) - initialize the token with the original certification > 0 (False) - initialize the token without the certification **Default:** 1 (True) **NOTE** - Previous to SAC 8.2, the default setting was 0 (False). As CardOS 4.2 does not support both FIPS and RSA-2048, failure to take this into account may lead to token initialization failure when using PKCS#11. - To prevent this, ensure that the default is set to False, or else ensure that the application provides both the required FIPS and RSA-2048 settings.
API: Private Data Caching If using an independent API for initialization, and if *Enable Private Cache* is selected, determines the token's private data cache default behavior.	**Value Name:** PrvCachingMode **Values:** > 0 - Always > 1 - While user is logged on > 2 - Never **Default:** 0 (Always)
Enable Private Data Caching Modification Determines if the token's *Private Data Caching* mode is modified after initialization.	**Value Name:** PrvCachingModify **Values:** > 1 (True) - Can be modified > 0 (False) - Cannot be modified **Default:** 0 (False)

Description	Value
Private Data Caching Mode If *Enable Private Data Caching Modification* is selected, determines who has rights to modify the token's *Private Data Caching* mode.	**Value Name:** PrvCachingOwner **Values:** > 0 - Admin > 1 - User **Default:** 0 (Admin)
API: RSA Secondary Authentication Mode If using an independent API for initialization, determines the default behavior for protecting RSA private keys on the token.	**Value Name:** 2ndAuthMode **Values:** > 0 - Never > 1 - Prompt on application request > 2 - Always prompt user > 3- Always > 4 - Token authentication on application request **Default:** 0 -(Never)
Enable RSA Secondary Authentication Modified Determines if the token's RSA secondary authentication is modified after initialization.	**Value Name:** 2ndAuthModify **Values:** > 1 (True) - Can modify > 0 (False) - Cannot modify **Default:** 0 (False)
Use the same token and administrator passwords for digital signature operations.	**Value Name:** LinkMode **Values:** > 1 (True) - Linked > 0 (False) - Unlinked **Default:** 0 (False)

SafeNet Authentication Client Tools UI Initialization Settings

The following settings are written to the **AccessControl** section in the file `/etc/eToken.conf`.

Description	Value
Enable Advanced View Button Determines if the *Advanced View* icon is enabled in SAC Tools.	**Value Name:** AdvancedView **Values:** > 1 - Selected > 0 - Not selected **Default:** 1

The following settings are written to the **InitApp** section in the file `/etc/eToken.conf`.

Description	Value
Hide Create Administrator Password Fields Defines if Create Administrator Password fields are hidden/ visible in the Password Settings window.	**Value Name:** HideInitCreateAdmin **Values:** > 0: Create Administrator Password fields are visible > 1: Create Administrator Password fields are hidden **Default:** > 0
Hide PinPolicy Button Defines if the Pin Policy button is hidden/ visible in the Password Settings window.	**Value Name:** HideInitPinPolicy **Values:** > 0: PIN Policy button is visible > 1: PIN Policy button is hidden **Default:** > 0
Default Token Password Defines the default Token Password.	**Value Name:** DefaultUserPassword **Values:** String **Default:** 1234567890

Description	Value
Enable Change Password on First Logon Determines if the *Token Password must be changed on first logon* option can be changed by the user in the *Token Initialization* window. **NOTE** This option is selected by default. If the option is deselected, it can be selected again.	**Value Name:** MustChangePasswordEnabled **Values:** > 1 - Selected > 0 - Not selected **Default:** 1
Change Password on First Logon Determines if the *Token Password must be changed on first logon* option is selected by default in the *Token Initialization* window.	**Value Name:** MustChangePassword **Value:** > 1 - Selected > 0 - Not selected **Default:** 1
Private Data Caching If *Enable Private Cache* is selected, determines the token's private data cache default behavior. **NOTE** Can be set in SafeNet Authentication Client Tools. This option is not supported by IDPrime cards.	**Value Name:** PrivateDataCaching **Values:** > 0 (Fastest) - Private data is cached when used by an application while the user is logged on to the token, and erased only when the token is disconnected > 1 - Private data is cached when used by an application while the user is logged on to the token, and erased when the user logs off or the token is disconnected > 2 - Private data is not cached **Default:** 0
RSA Secondary Authentication Mode Defines the default behavior for protecting RSA private keys on the token. **NOTE** Can be set in SafeNet Authentication Client Tools. This option is not supported by IDPrime cards.	**Value Name:** RSASecondaryAuthenticationMode **Values:** > 0 - Never > 1 - Prompt user on application reques > 2 - Always prompt user > 3 - Always > 4 - Token authentication on application request **Default:** 0

Description	Value
Reuse Current Token Name Determines if the token’s current Token Name is displayed as the default Token Name when the token is re-initialized.	**Value Name:** ReadLabelFromToken **Values:** > 1 -The current Token Name is displayed > 0 -The current Token Name is ignored **Default:** 1

SafeNet Authentication Client Tools UI Settings

The following settings are written to the **UI** section in the file `/etc/eToken.conf`.

Description	Value
Use Default Password Determines if the *Change Password on First Logon* process assumes the current Token Password is the default (defined in the Default Token Password), and does not prompt the user to supply it.	**Value Name:** UseDefaultPassword **Values:** > 1 (True) - The default Token Password is automatically entered in the password field > 0 (False) -The default Token Password is not automatically entered in the password field **Default:** 0 (False)
Password Term Defines the term used for the token's user password. **NOTE** If a language other than English is used, ensure that the Password Terms are translated.	**Value Name:** PasswordTerm **Values** (String): > Password > PIN > Passcode > Passphrase **Default:** Password

Description	Value
Decimal Serial Number Determines if the Token Information window displays the token serial number in hexadecimal or in decimal format.	**Value Name:** ShowDecimalSerial **Values:** > 1 (True) -Displays the serial number in decimal format > 0 (False) -Displays the serial number in hexadecimal format **Default:** 0
Enable Tray Icon Determines if the application tray icon is displayed when SafeNet Authentication Client is started.	**Value Name:** ShowInTray **Values:** > 0 - Never Show > 1 - Always Show **Default:** Always show
Enable Connection Notification Determines if a notification balloon is displayed when a token is connected or disconnected.	**Value Name:** ShowBalloonEvents **Values:** > 0 - Not Displayed > 1 - Displayed **Default:** 0
Enable Logging Control Determines if the *Enable Logging /Disable Logging* button is enabled in the **Client Settings** > **Advanced** tab.	**Value Name:** AllowLogsControl **Values:** > 1 - Enabled > 0 - Disabled **Default:** 1
Home URL Overwrites the SafeNet home URL in SafeNet Authentication Client Tools.	**Value Name:** HomeUrl **Values** (String): Valid URL **Default:** SafeNet's home URL

Description	Value
eToken Anywhere Determines if eToken Anywhere features are supported.	**Value Name:** AnywhereExtendedMode **Values:** > 1 - Supported > 0 - Not supported **Default:** 1
Enable Certificate Expiration Warning Determines if a warning message is displayed when certificates on the token are about to expire.	**Value Name:** CertificateExpiryAlert **Values:** > 1 (True) - Notify the user > 0 (False) - Do not notify the user **Default:** 1 (True)
Ignore Archived Certificates Determines if archived certificates are ignored, and no warning message is displayed for certificates that are about to expire.	**Value Name:** IgnoreArchivedCertificates **Values:** > 1 - Archived certificates are ignored > 0 - A warning message is displayed if the token contains archived certificates. **Default:** 1

Description	Value
Ignore Expired Certificates Determines if expired certificates are ignored, and no warning message is displayed for expired certificates.	**Value Name:** IgnoreExpiredCertificates **Values:** > 1 - Expired certificates are ignored > 0 - A warning message is displayed if the token contains expired certificates **Default:** 0
Certificate Expiration Verification Frequency Defines the minimum interval, in days, between certificate expiration date verifications.	**Value Name:** UpdateAlertMinInterval **Values:** > 0 **Default:** 14 days
Certificate Expiration Warning Period Defines the number of days before a certificate's expiration date during which a warning message is displayed.	**Value Name:** ExpiryAlertPeriodStart **Values:** > =0 (0 = No warning) **Default:** 30 days
Warning Message Title Defines the title to display in certificate expiration warning messages.	**Value Name:** AlertTitle **Values:** String **Default:** SafeNet Authentication Client
Certificate Will Expire Warning Message Defines the warning message to display in a balloon during a certificate's *Certificate Expiration Warning Period*.	**Value Name:** FutureAlertMessage **Values:** String **Default:** A certificate on your token expires in $EXPIRE_IN_DAYS days.

Description	Value
Expiry Date Format Defines the format of the certificate’s expiry date ($EXPIRY_DATE) displayed in a balloon.	**Value Name:** EXPIRY_DATE_FORMAT **Values:** Set the year/month/day in the required order using the format: %Y/%m/%d **Default:** %Y/%m/%d
Certificate Expired Warning Message Defines the warning message to display in a balloon if a certificate's expiration date has passed.	**Value Name:** PastAlertMessage **Values:** String **Default:** Update your token now.
Warning Message Click Action Defines what happens when the user clicks the message balloon.	**Value Name:** AlertMessageClickAction **Values:** > 0 - No action > 1 - Show detailed message > 2 - Open website **Default:** 0
Detailed Message If *Show detailed message* is selected in **Warning Message** > **Click Action** setting, defines the detailed message to display.	**Value Name:** ActionDetailedMessage **Values:** String **No default**
Website URL If *Open website* is selected in the **Warning Message** > **Click Action** setting, defines the URL to display.	**Value Name:** ActionWebSiteURL **Values** (string): Website address **No default**

Description	Value
Enable Password Expiration Notification Determines if a pop-up message is displayed in the system when the Token Password is about to expire.	**Value Name:** NotifyPasswordExpiration **Values:** > 1 (True) - A message is displayed > 0 (False) - A message is not displayed **Default:** 1 (True)
Display Virtual Keyboard Determines if SafeNet's keystroke-secure Virtual Keyboard replaces standard keyboard entry of password fields in the following windows: > Token Logon > Change Password **NOTE** The virtual keyboard supports English characters only.	**Value Name:** VirtualKeyboardOn **Values:** > 1 (True) - Virtual keyboard on > 0 (False) - Virtual keyboard off **Default:** 0 (False)
Password Policy Instructions If not empty, defines a string that replaces the default password policy description displayed in the *Unlock and Change Password* windows.	**Value Name:** PasswordPolicyInstructions **Values:** String **No default**
Define Initialization Mode Select this option if you want the *Initialization Options* window (first window displayed when initializing a device) to be ignored.	**Value Name:** DefInitMode **Values:** > 0 - Display the *Initialization Options* window > 1 - Set Preserve Mode > 2 - Set Configure Mode **Default:** 0

Description	Value
Import Certificate Chain Determines if the certificate chain is imported to the token.	**Value Name:** ImportCertChain **Values:** > 0 - Do not import certificate chain > 1 - Import certificate chain > 2- User selects import behavior **Default:** 0
Prevent Must Change Password dialog popup Determines if the tray icon will display a popup message to prompt the user to change the user password for tokens that are not initialized.	**Value Name:** DenyMustChangePopup **Values:** > 0 - Must Change Password pop-up message will not be displayed > 1 - Must Change Password pop-up message will be displayed **Default:** 0

Token Password Quality Settings

The following settings are written to the `PQ` section in the file `/etc/eToken.conf`.

> **NOTE** These settings are not relevant to IDPrime cards and eToken 5110 CC, as the *Password Quality* settings reside on the card itself.

Description	Value
Password - Include Non ASCII Characters Determines if the password can be included for non-ASCII characters. **Note:** Applicable for IDPrime cards only.	**Value Name:** pqNonAscii **Values:** > 0: Permitted > 1: Forbidden > 2: Mandatory **Default:** 0
Password - Number Of Different Repeating Characters Determines the number of different characters that can be repeated at least once.	**Value Name:•** pqNumDiffCharRepeat **Values:** >= 0 (0 = No check) **Default:** 0
Password - Maximum Number A Character Can Appear Determines the maximum number a character can appear.	**Value Name:•** pqMaxNumCharAppear **Values:** >= 0 (0 = No check) **Default:** 0
Password - Maximum Number Of Characters In A Sequence Determines the maximum number of characters in a sequence. For example: If the value is set to 4, the sequence 1,2,3,4,a,5 is allowed but 1,2,3,4,5,a is not allowed.	**Key Name:•** pqMaxNumCharSequence: **Values:** >= 0 (0 = No check) **Default:** 0
Password - Maximum Adjacent Repetitions Of A Character Determines the maximum number a character can be repeated in adjacent positions. **NOTE** If pqMaxNumCharRepeatPos = 0, then the value of pqMaxRepeated is applicable.	**Value Name:•** pqMaxNumCharRepeatPos **Values:** >= 0 (0 = No check) **Default:** 0

Description	Value
Password - Minimum Length Defines the minimum password length. **NOTE** Can be set in SafeNet Authentication Client Tools. For more information on how to configure the ‘Password Minimum Length’ property as permanent. See "Changing the Password Minimum Length Permanently" on page 1.	**Value Name:** pqMinLen **Values:** >=4 **Default:** 6
Password - Maximum Length Defines the maximum password length. **NOTE** Can be set in SafeNet Authentication Client Tools. > Devices that have an eToken applet (such as: eToken 5110, 5110 FIPS or IDCore 830B) the max pin length property is not saved on the device. This property has only a UI meaning (i.e.no security meaning). > The value of the proprietary PKCS#11 attribute `ETCKA_PIN_MAX_LEN` on these devices is always read from SAC's `pqMaxLen` property. > If the `pqMaxLen` property is not explicitly defined, it receives the default value (20). > In addition, SAC Tools has it's own limitation for the `ETCKA_PIN_MAX_LEN` attribute with a max of 16 characters. Even though the `pqMaxLen` value has a value that is greater than 16.	**Value Name:** pqMaxLen **Values:** Cannot be less than the Password Minimum Length **Default:** 16
Password - Maximum Usage Period Defines the maximum number of days a password is valid. **NOTE** Can be set in SafeNet Authentication Client Tools. **NOTE** This parameter is *Day Sensitive* that is the system counts the days and not the hour in which the user made the change.	**Value Name:** pqMaxAge **Values:**>=0 (0 =No expiration) **Default:** 0
Password - Minimum Usage Period Defines the minimum number of days between password changes. **NOTE** Can be set in SafeNet Authentication Client Tools.	**Value Name:** pqMinAge **Values:** >=0 (0 = No minimum) **Default:** 0

Description	Value
Password - Expiration Warning Period Defines the number of days before expiration during which a warning is displayed. **NOTE** Can be set in SafeNet Authentication Client Tools.	**Value Name:** pqWarnPeriod **Values:** >=0 (0 = No warning) **Default:** 0
Password - History Size Defines the number of recent passwords that must not be repeated. **NOTE** Can be set in SafeNet Authentication Client Tools. Maximum value of History size for IDPrime devices is 10.	**Value Name:** pqHistorySize **Values:** >= 0 (0 = No minimum) **Default:** 10
Password - Maximum Consecutive Repetitions Defines the maximum number of consecutive times a character can be used in a password. **NOTE** Can be set in SafeNet Authentication Client Tools. If pqMaxNumCharRepeatPos = 0, then the value of pqMaxRepeated is applicable.	**Value Name:** pqMaxRepeated **Values:** 0 - 16 (0 = No maximum) **Default:** 3
Password - Complexity > Determines if there is a minimum number of character types that must be included in a new Token Password. > The character types are upper-case letters, lower-case letters, numerals, and special characters. **NOTE** Can be set in SafeNet Authentication Client Tools.	**Value Name:** pqMixChars **Values:** > 1 - A minimum of 2 or 3 types must be included, as defined in the *Password- Minimum Mixed Character Types* setting > 0 -The rule for each character type is defined in the character type's *Include* setting **Default:** 1

Description	Value
Password - Minimum Mixed Character Types > Defines the minimum number of character types that must be included in a new Token Password. > The character types are upper-case letters, lower-case letters, numerals, and special characters. **NOTE** - Applies only when the *Password - Complexity* setting is set to *Standard complexity*. - Can be set in SafeNet Authentication Client Tools.	**Value Name:** pqMixLevel **Values:** > 0 - At least 3 character types > 1 - At least 2 character types **Default:** 0
Password - Include Numerals Determines if the password can include numerals. **NOTE** - Applies only when the *Password - Complexity* setting is set to *Manual complexity*. - Can be set in SafeNet Authentication Client Tools.	**Value Name:** pqNumbers **Values:** > 0 - Permitted > 1 - Forbidden > 2 - Mandatory **Default:** 0
Password - Include Upper-Case Determines if the password can include upper-case letters. **NOTE** - Applies only when the *Password - Complexity* setting is set to *Manual complexity*. - Can be set in SafeNet Authentication Client Tools.	**Value Name:** pqUpperCase **Values:** > 0 - Permitted > 1 - Forbidden > 2 - Mandatory **Default:** 0
Password - Include Lower-Case Determines if the password can include lower-case letters. **NOTE** - Applies only when the *Password - Complexity* setting is set to *Manual complexity*. - Can be set in SafeNet Authentication Client Tools.	**Value Name:** pqLowerCase **Values:** > 0 - Permitted > 1 - Forbidden > 2 - Mandatory **Default:** 0

Description	Value
Password - Include Special Characters Determines if the password can include special characters, such as @,!, &. **NOTE** - Applies only when the *Password - Complexity* setting is set to *Manual complexity*. - Can be set in SafeNet Authentication Client Tools.	**Value Name:** pqSpecial **Values:** > 0 - Permitted > 1 - Forbidden > 2 - Mandatory **Default:** 0
Password Quality Check on Initialization Determines if the password quality settings are checked and enforced when a token is initialized. **NOTE** It is recommended that this policy must not be set when tokens are enrolled using SafeNet Authentication Manager.	**Value Name:** pqCheckInit **Values:** > 1 (True) -The password quality is enforced > 0 (False) - The password quality is not enforced **Default:** 0
Password Quality Owner Defines the owner of the password quality settings on a re-initialized token, and defines the default of the *Password Quality Modifiable* setting.	**Value Name:** pqOwner **Values:** > 0 - Administrator > 1 - User **Default:** > 0 - For tokens with an Administrator Password > 1 - For tokens without an Administrator Password

Description	Value
Enable Password Quality Modification Determines if the password quality settings on a newly initialized token can be modified by the owner. See the *Password Quality Owner* setting.	**Value Name:** pqModifiable **Values:** > 1 (True) - The password quality can be modified by the owner > 0 (False) - The password quality cannot be modified by the owner **Default:** > 1 (True) - For administrator owned tokens > 0 (False) - For user owned tokens
Enable Administrator Password Quality Check > Determines if the Administrator Password Quality Check is enabled. > When enabled, this property enforces an administrator (SO) password (on eToken and IDPrime devices) that has at least 3 different character types and a minimum length of 8 characters. The character types are: upper-case letters, lower-case letters, numerals, and special characters. **NOTE** For backward compatibility on IDPrime devices, the Administrator Key can be used with 48 hexadecimal characters via the UI and/or 24 binary bytes via the API call. > When disabled, the old behavior is as follows: • **eToken:** Minimum of 4 characters and no minimum character type enforcement • **IDPrime:** Minimum of 8 characters and no minimum character type enforcement, or the administrator key can be used. **NOTE** When the *ITI Certification mode* property is enabled, the *Enable Administrator Password Quality Check* property will be disabled.	**Value Name:** pqAdminPQ **Values:** > 1 (Enabled) - Administrator Password Quality is enforced > 0 (Disabled) - Administrator Password Quality is disabled **Default:** Enabled

SafeNet Authentication Client Tools UI Access Control List

Access Control Properties determine which features are enabled in the SafeNet Authentication Client Tools and Tray Menu.

The following settings are written to the **AccessControl** section in the file `/etc/eToken.conf`.

Access Control Feature	Value
All access control features are listed in below table	**Values:** > 1 (True) - The feature is enabled. > 0 (False) - The feature is disabled. **Default:** 1(True) - except where indicated in the table

> **NOTE** All access control features are enabled by default, except where indicated in the table.

Access Control Feature	Value Name	Description
Crypto Notification Timeout	CryptoNotificationTimeout	> Enables/Disables the notification: “The process may take a while....” > Enter the time in seconds after which the notification is displayed. For example, the value 30 means the notification is delayed by 30 seconds. **NOTE** By default, this feature is disabled.
Rename Token	RenameToken	Enables/Disables the *Rename Token* feature in SafeNet Authentication Client Tools.
Change Token Password	ChangePassword	Enables/Disables the *Change Token Password* feature in SafeNet Authentication Client Tools.
Unlock Token	UnlockEtoken	Enables/Disables the *Unlock Token* feature in SafeNet Authentication Client Tools.
Delete Token Content	ClearEToken	Enables/Disables the *Delete Token Content* feature in SafeNet Authentication Client Tools.
View Token Information	ViewTokenInfo	Enables/Disables the *View Token Information* feature in SafeNet Authentication Client Tools.

Access Control Feature	Value Name	Description
Disconnect SafeNet Virtual Token	DisconnectVirtual	Enables/Disables the *23* feature in SafeNet Authentication Client Tools.
Help	ShowHelp	Determines if the user can open the Help file in SafeNet Authentication Client Tools.
Advanced View	OpenAdvancedView	Determines if the user can open the *Advanced View* in SafeNet Authentication Client Tools.
Reader Settings	ManageReaders	Enables/Disables the *Reader Settings* feature in SafeNet Authentication Client Tools.
Connect SafeNet Virtual Token	AddeTokenVirtual	Enables/Disables the *Connect SafeNet Virtual Token* feature in SafeNet Authentication Client Tools.
Initialize Token	InitializeEToken	Enables/Disables the *Initialize Token* feature in SafeNet Authentication Client Tools.
Import Certificate	ImportCertificate	Enables/Disables the *Import Certificate* feature in SafeNet Authentication Client Tools.
Reset Default Certificate Selection	ClearDefaultCert	Enables/Disables the *Reset Default Certificate Selection* feature in SafeNet Authentication Client Tools.
Delete Certificate	DeleteCertificate	Enables/Disables the *Delete Certificate* feature in SafeNet Authentication ClientTools.
Export Certificate	ExportCertificate	Enables/Disables the *Export Certificate* feature in SafeNet Authentication Client Tools.
Copy Certificate Data to Clipboard	CopyCertificateData	Enables/Disables the *Copy Certificate Data to Clipboard* feature in SafeNet Authentication Client Tools.
Log On as Administrator	LoginAsAdministrator	Enables/Disables the *Log On as Administrator* feature in SafeNet Authentication Client Tools.

Access Control Feature	Value Name	Description
Change Administrator Password	ChangeAdministratorPassword	Enables/Disables the *Change Administrator Password* feature in SafeNet Authentication Client Tools.
Set Token Password	SetUserPassword	Enables/Disables the *Set Token Password* feature in SafeNet Authentication Client Tools.
Token Password Retries	AllowChangeUserMaxRetry	Enables/Disables the *Logon retries before token is locked* feature (for the Token Password) in SafeNet Authentication Client Tools.
Administrator Password Retries	AllowChangeAdminMaxRetry	Enables/Disables the *Logon retries before token is locked* feature (for the Administrator Password) in SafeNet Authentication Client Tools.
Advanced Initialization Settings	OpenAdvancedModeOfInitialize	Enables/Disables the *Advanced* button in the *Token Initialization* window in SafeNet Authentication Client Tools. **NOTE** If disabled, IDPrime CC card cannot be initialized.
Change Initialization Key during Initialization	ChangeInitializationKeyDuringInitialize	Enables/Disables the *Change Initialization key* button in the *Advanced Token Initialization Settings* window in SafeNet Authentication Client Tools.
Common Criteria Settings	CommonCriteriaPasswordSetting	Enables/Disables the *Common Criteria* option in the Certification combo box.
System Tray - Unlock Token	TrayIconUnlockEtoken	Enables/Disables the *Unlock Token* feature in the SafeNet Authentication Client Tray Menu.

Access Control Feature	Value Name	Description
System Tray - Delete Token Content	TrayIconClearEToken	Enables/Disables the *Delete Token Content* feature in the SafeNet Authentication Client Tray Menu. **NOTE** By default, this feature is Disabled.
System Tray - Change Token Password	TrayIconChangePassword	Enables/Disables the *Change Token Password* feature in the SafeNet Authentication Client Tray Menu.
System Tray - Select Token	SwitcheToken	Enables/Disables the *Select Token* feature in the SafeNet Authentication Client Tray Menu.
System Tray - Synchronize Domain-Token Passwords	SyncDomainAndTokenPass	Enables/Disables the *Synchronize Domain Token Passwords* feature in the SafeNet Authentication Client Tray Menu.
System Tray - Tools	OpeneTokenProperties	Enables/Disables the *Tools* menu item (open SafeNet Authentication Client Tools) in the SafeNet Authentication Client Tray Menu.
System Tray - About	About	Enables/Disables the *About* menu item in the SafeNet Authentication Client Tray Menu.
Enable Change IdenTrust Identity	IdentrusChangePassword	Enables/Disables the *Change IdenTrust PIN* feature in SafeNet Authentication Client Tools.
Enable Unblock IdenTrust Passcode	IdentrusUnlock	Enables/Disables the *Unlock IdenTrust* feature in SafeNet Authentication Client Tools.
Delete Data Object	DeleteDataObject	Enables/Disables the *Delete Data Object* feature in SafeNet Authentication Client Tools.

Access Control Feature	Value Name	Description
Allow One Factor **NOTE** This property cannot be set in the **Access Control Properties** window.	AllowOneFactor	Enables/Disables the *Allow One Factor* feature in the *Advanced Token > Initialization Settings* window in SafeNet Authentication Client Tools.
PIN Type	PinType	Defines which GUI PIN Properties are enabled/disabled in SAC Tools *Advanced PIN Properties* tab and the *Initialization* window.
PIN Purpose	PinPurpose	
Cache Type	PinCacheType	
Cache Timeout	PinCacheInfo	
PIN Flags	PinFlags	
Ext. PIN Flags	PinFlagsEx	
Validity period (days)	PinValidity	
Expiration warning period (days)	PinWarning	

Access Control Feature	Value Name	Description
Minimum length (characters)	PinMinLen	Defines which GUI PIN Quality parameters are enabled/disabled in SAC Tools *Advanced* tab and the *Initialization* window.
Maximum length (characters)	PinMaxLen	
History size	PinHistory	
Number of different characters that can be repeated at least once	PinNumDiffCharRepeat	
Maximum number a characters can appear	PinMaxNumCharAppear	
Maximum number of characters in a sequence	PinMaxNumCharSequence	
Maximum number a character can be repeated in adjacent positions	PinMaxNumCharRepeatPos	
Numeric	PinNumber	
Alpha Upper	PinUpper	
Alpha Lower	PinLower	
Non alpha	PinSpecial	
Alpha	PinAlphabetic	
Non Ascii	PinNonAlphabetic	
Minimum usage period (days)	PinMinUse	
Maximum usage period (days)	PinMaxUse	
Must meet complexity requirements	PinComplexity	
Maximum consecutive repetitions	PinMaxRepeat	

Security Settings

The following settings are written to the **Crypto** section in the file `/etc/eToken.conf`.

Description	Value
Key Management > Defines key creation, export, unwrap, and off-board crypto policies. > SAC default behavior may be updated in future versions in order to comply with NIST requirements. > It is up to the customer to check that it will be compatible with third-party applications.	**Value Name:** Key-Management-Security **Values:** (String) > Compatible: • Enables the use of features that are deprecated in the Optimized and Strict configurations below. • This is the default value for SAC versions below 10.5. Setting this value causes SAC to be compatible with SAC 10.5 and below. • It is strongly recommended to read "Security Recommendations" on page 1 before applying legacy values. > Optimized: • Disable the generation or creation of exportable keys. • Disable the exporting of keys, regardless of how they are generated. • Disable any usage of symmetric keys off-board including unwrap. • Disable the unwrap-PKCS1.5 and unwrap-AES-CBC on hardware tokens (session enable). > Strict: • Disable the generation or creation of exportable keys. • Disable the exporting of keys, regardless of how they are generated. • Disable all the unwrap-PKCS1.5 and unwrap-AES-CBC operations. • Disable any usage of symmetric keys off-board including unwrap. **Default:** Optimized

Description	Value
Deprecated Cryptographic Algorithms and Features > The default list of deprecated cryptographic algorithms and features may be enhanced in order to comply with NIST requirements in future versions. > It is up to the customer to check that it will be compatible with third-party applications.	**Value Name:** Disable-Crypto **Values:** (String) > None - All SAC cryptographic algorithms and features are supported. • This was the default value for SAC versions below 10.5. Setting this value will cause SAC to be compatible with SAC 10.5 and below. • It is strongly recommended to read "Security Recommendations" on page 1 before applying legacy values. > Obsolete - A list of restricted and deprecated cryptographic algorithms and features. The following are deprecated: MD5, RC2, RC4, DES, 2DES, GenericSecret<112, RSA-RAW, RSA<2048, ECC<224, ECB, Sign-SHA1. > Manual - Create your own list of deprecated algorithms and features. (See the description below). **Default:** Obsolete

The following can be disabled:

- **Algorithms:** RSA, ECC, DES, 2DES, 3DES, AES, RC2, RC4, GenericSecret
- **Hash types:** MD5, SHA1, SHA2
- **Padding types:** RAW, PKCS1, OAEP, PSS
- **Cipher modes:** ECB, CBC, CTR, CCM
- **Mechanisms:** MAC, HMAC, ECDSA, ECDH
- **Operations:** Encrypt, Decrypt, Sign, Verify, Generate, Derive, Wrap, Unwrap, Digest, Create (keys only)
- **Weak key size:** RSA<2048
- **Object types:**
  - HWEF – Elementary file (EF) objects (used by eToken devices for storing exportable symmetric keys and symmetric keys without on-board implementation)
  - HWALL – All types of objects implemented on token (Base Security Object (BSO) and EF),
  - ETV – eToken Virtual

Example of a manual configuration: Encrypt-DES-ECB, Sign-3DES-MAC, DES-CTR, HMAC-MD5, HMAC-SHA1, HMAC-SHA2, DES-CBC, Unwrap-DES-ECB, RSA-PKCS1-MD5, Verify-RSAPSS-SHA2, AES-CTR, AES-MAC, Decrypt-RC2, Wrap-ECB.

Log Settings

The following settings are written to the **Log** section in the file `/etc/eToken.conf`.

Description	Value
Enabled Determines if the SafeNet Authentication Client Log feature is enabled.	**Value Name:** Enabled **Value:** > 1 - Enabled > 0 - Disabled **Default:** 0 (Disabled)
Days Defines the number of days log files will be saved from the time the log feature was enabled.	**Value Name:** Days **Value:** Enter the number of days (numerical). **Default:** 1 day
MaxFileSize Defines the maximum size of an individual log file. Once the maximum file size is reached, SAC removes older log records to allow saving newer log information.	**Value Name:** MaxFileSize **Value:** Enter a value in Bytes. **Default:** 2000000 (Bytes) (Approximately 2MB)
TotalMaxSizeMB Defines the total size of all the log files when in debug mode. (Megabytes).	**Value Name:** TotalMaxSizeMB **Value:** Enter a value in Megabytes. **Default:** 0 (Unlimited)

Description	Value
ManageTimeInterval Defines how often the TotalMaxSize parameter is checked to ensure that the total maximum size is not exceeded.	**Value Name:** ManageTimeInterval **Value:** Enter a value in minutes (numerical). **Default:** 60 minutes

CHAPTER 6: Security Recommendations

The information in this chapter helps you maintain a secured SAC environment and keep your information safe.

Enforcing Restrictive Cryptographic Policies

To allow organizations to enforce restrictive cryptographic policies when using SafeNet smart card and USB tokens, the following enhancements were introduced:

- Key Management Security Policy
- Disable Cryptographic Algorithm Policy

For more details, see "Security Settings" on page 58.

The motivation behind these enhancements:

- Legacy cryptographic schemes can cause organizations to fail current compliance requirements or expose cryptographic weakness associated with obsolete algorithms and mechanisms.

  The following enhancements were made to SafeNet Authentication Client to allow organizations to block the use of such schemes, according to organizational policies.

  - Enabling symmetric keys wrapping with other symmetric keys using GCM and CCM modes of operation.
  - Preventing legacy algorithms from being used by adding a key wrapping policy that enforces the usage of only GCM and CCM modes of operation for symmetric encryption, and PKCS#1 v2.1 padding for RSA encryption.
- SafeNet introduced a new mechanism that allows administrators to prevent the use of legacy or obsolete algorithms by third-party applications. These cryptographic algorithms conform to the National Institute of Standards and Technology (NIST), preventing third-party applications from using legacy or obsolete algorithms.

> **NOTE** Once a restrictive policy is set, the use of SafeNet Authentication Client with the above algorithms is blocked.
> - This might have implications on the way in which the third-party's applications currently work.
> - Administrators must make sure that the third-party applications used by the organization are configured accordingly, and do no use one of the algorithms listed above, as they will be blocked.

Create Symmetric Key Objects using PKCS#11

The following are performed as part of SafeNet Authentication Client security enhancement campaign:

- Protected memory is used when working with the private cache between PKCS#11 API calls. Private cache is unlocked to retrieve data and then locked immediately after retrieving the data to ensure that there is no sensitive data in the private cache. This ensures that the key cannot be revealed in plain text.
- Sensitive data is securely zeroed prior to freeing up the memory.
- AES and Generic symmetric key files were created with Secured Messaging (SM) protection, so that the Microsoft smart card transport layer does not contain any APDU data with plain symmetric key material.

  For SM to support the AES/3DES and Generic symmetric keys, the keys must be created on an eToken Java device that is initialized in FIPS/CC mode. Applying SM to symmetric keys changes the object format on the smart card, resulting in the keys not being backward compatible.

> **NOTE** Keys that are created with previous SAC versions or on eToken Java devices which are formatted in non-FIPS/CC mode are not protected by SM.
> AES/3DES keys that are created using the `CKA_SENSITIVE = TRUE` and `CKA_EXTRACTABLE = FALSE` attributes are backward compatible (BS Object keys).