SafeNet Authentication Service

INTEGRATION GUIDE: USING RADIUS PROTOCOL FOR LINUX PAM MODULES
Gemalto 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 Gemalto 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.

Gemalto 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 Gemalto be held liable for any third party actions and in particular in case of any successful attack against systems or equipment incorporating Gemalto products. Gemalto 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 ................................................................................................................................................. 6
Third-Party Software Acknowledgement .................................................................................................. 6
Description .................................................................................................................................................. 6
Applicability ............................................................................................................................................... 6
Environment ............................................................................................................................................... 7
RADIUS Prerequisites .............................................................................................................................. 7
Audience .................................................................................................................................................... 7
Support Contacts ....................................................................................................................................... 8
Customer Support Portal ........................................................................................................................ 8
Telephone Support .................................................................................................................................... 8
Email Support ............................................................................................................................................ 8

CHAPTER 1: Authentication Flow ............................................................................................................... 9

CHAPTER 2: SafeNet Authentication Service Setup .................................................................................... 10
Creating Users Stores ............................................................................................................................... 10
Assigning an Authenticator ....................................................................................................................... 11
Adding Linux Server as an Authentication Node ..................................................................................... 11

CHAPTER 3: Linux PAM Modules Setup .................................................................................................. 13
Compiling the PAM Module ....................................................................................................................... 13
Adding a RADIUS Server for the PAM Module ...................................................................................... 13
Configuring the Application-Specific Configuration Files ................................................................. 15
Configuring the sshd file for Single-Factor Authentication .................................................................. 15
Configuring the sshd file for Two-Factor Authentication ..................................................................... 16

CHAPTER 4: Running the Solution ........................................................................................................... 19
Using MobilePASS+ (or MobilePASS) .................................................................................................... 19
Using Push OTP ....................................................................................................................................... 21
Using SMS ............................................................................................................................................... 24
Third-Party Software Acknowledgement

This document is intended to help users of Gemalto products when working with third-party software, such as Linux PAM Modules.

Material from third-party software is being used solely for the purpose of making instructions clear. Screen images and content obtained from third-party software will be acknowledged as such.

This document contains the following chapters:

➢ "Authentication Flow" on page 9
➢ "SafeNet Authentication Service Setup" on page 10
➢ "Linux PAM Modules Setup" on page 13
➢ “Running the Solution” on page 19

Description

SafeNet Authentication Service (SAS) delivers a fully automated, versatile, and strong authentication-as-a-service solution.

With no infrastructure required, SafeNet Authentication Service provides smooth management processes and highly flexible security policies, token choice, and integration APIs.

Linux PAM Modules can be used in Linux environments to provide an additional level of security with a given service or application that is PAM aware. This document describes how to:

➢ Deploy multi-factor authentication (MFA) options in Linux PAM Modules using SafeNet one-time password (OTP) authenticators managed by SafeNet Authentication Service.
➢ Configure Linux PAM Modules to work with SafeNet Authentication Service in RADIUS mode.

It is assumed that the Linux PAM Modules environment is already configured and working with static passwords prior to implementing the multi-factor authentication using SafeNet Authentication Service.

Linux PAM Modules can be configured to support multi-factor authentication in several modes. The RADIUS protocol will be used for the purpose of working with SafeNet Authentication Service.

Applicability

The information in this document applies to:

➢ SafeNet Authentication Service (SAS)—SafeNet’s cloud-based authentication service
➢ SafeNet Authentication Service – Service Provider Edition (SAS-SPE)—A server version that is used by service providers to deploy instances of SafeNet Authentication Service
SafeNet Authentication Service – Private Cloud Edition (SAS-PCE)—A server version that is used to deploy the solution on-premises in the organization

Environment

The integration environment that is used in this document is based on the following software versions:

- SafeNet Authentication Service – Private Cloud Edition (SAS-PCE)—only when using this version. For Cloud not necessary to fill in version number.
- Linux PAM Modules—Version 1.4.0 on RHEL 7.4

RADIUS Prerequisites

To enable SafeNet Authentication Service (SAS) to receive RADIUS requests from Linux PAM Modules, ensure the following:

- End users can authenticate using the Linux PAM Modules with a static password before configuring the Linux PAM Modules to use RADIUS authentication.
- Ports 1812/1813 are open to and from the Linux PAM Modules.
- A shared secret key has been selected. A shared secret key provides an added layer of security by supplying an indirect reference to a shared secret key. It is used by a mutual agreement between the RADIUS server and RADIUS client for encryption, decryption, and digital signatures.
- If Linux PAM packages are not installed, run the following commands to install the Linux PAM packages:
  
  ```bash
  #yum install pam
  #yum install pam-devel
  #yum install gcc
  ```

- RADIUS PAM modules must be downloaded. Download the modules from the http://freeradius.org/pam_radius_auth/ URL.
  
  **Note:** The URL is a third-party website and can change at any time.

- Extract the contents of the `tar.gz` file (PAM modules you downloaded) to a directory. Ensure that you have read and write permission to the directory.
- The user name in Linux and the user ID in SAS must be identical.

Audience

This document is targeted to system administrators who are familiar with Linux PAM Modules, and are interested in adding multi-factor authentication capabilities using SafeNet Authentication Service.
Support Contacts

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

Gemalto 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 Gemalto 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.gemalto.com, is a 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 Support

If you have an urgent problem, or cannot access the Customer Support Portal, you can contact Gemalto Customer Support by telephone at +1 410-931-7520. Additional local telephone support numbers are listed on the support portal.

Email Support

You can also contact technical support by email at technical.support@gemalto.com.
CHAPTER 1: Authentication Flow

SafeNet Authentication Service communicates with a large number of VPN and access-gateway solutions using the RADIUS protocol.

The image below describes the data flow of a multi-factor authentication transaction for Linux PAM Modules.

1. A user attempts to log on to PAM-aware Linux service using an OTP authenticator.
2. Linux server sends a RADIUS request with the user’s credentials to SafeNet Authentication Service (SAS) for validation.
3. The SAS authentication reply is sent back to the Linux server.
4. The user is granted or denied access to the Linux server based on the OTP value calculation results from SAS.

For SafeNet Authentication Service (Cloud), a RADIUS agent is already configured and can be used without any additional agent installation or configuration requirements.

For SafeNet Authentication Service (SPE and PCE), a RADIUS agent (SAS Agent for Microsoft IAS or NPS, and FreeRADIUS) needs to be configured in the customer’s environment.

For more information on how to install and configure the SAS Agent for Microsoft IAS, Microsoft NPS, and FreeRADIUS, refer to the Agent Documentation.
CHAPTER 2: SafeNet Authentication Service Setup

The deployment of multi-factor authentication using SafeNet Authentication Service (SAS) with Linux PAM Modules using RADIUS protocol requires the following:

- Creating Users Stores, page 10
- Assigning an Authenticator, page 11
- Adding Linux Server as an Authentication Node, page 11

Creating Users Stores

Before SafeNet Authentication Service (SAS) can authenticate any user in your organization, you need to create a user store in SAS that reflects the users that would need to use multi-factor authentication. User records are created in the SAS user store using one of the following methods:

- Manually, one user at a time, using the Create User shortcut
- Manually, by importing one or more user records via a flat file
- Automatically, by synchronizing with your Active Directory / LDAP server using the SAS Synchronization Agent

For additional details on importing users to SafeNet Authentication Service, refer to “Creating Users” in the “SafeNet Authentication Service Subscriber Account Operator Guide” available here.

All SafeNet Authentication Service documentation can be found on the SafeNet Knowledge Base site.
Assigning an Authenticator

SafeNet Authentication Service (SAS) supports a number of authentication methods that can be used as a second authentication factor for users who are authenticating through Linux PAM Modules.

The following authenticators are supported:

- MobilePASS+ / MobilePASS
- SMS token
- eToken PASS
- RB-1 Keypad Token
- KT-4 Token
- SafeNet Gold

Authenticators can be assigned to users in two ways:

- **Manual provisioning**—Assign an authenticator to users one at a time.
- **Provisioning rules**—The administrator can set provisioning rules in SAS so that the rules will be triggered when group memberships and other user attributes change. An authenticator will be assigned automatically to the user.

Refer to “Provisioning Rules” in the “SafeNet Authentication Service Subscriber Account Operator Guide” (available [here](#)) to learn how to provision the different authentication methods to the users in the SAS user store.

Adding Linux Server as an Authentication Node

Add a RADIUS entry in the SafeNet Authentication Service (SAS) **Auth Nodes** module to prepare it to receive RADIUS authentication requests from the Linux server. You will need the IP address of the Linux server and the shared secret to be used by both SAS and Linux server.

1. Log in to the SAS console with an Operator account, click the **COMMS** tab and then select **Auth Nodes**.
2. In the **Auth Nodes** module, click the **Auth Nodes** link.

![Auth Nodes Module](image)

**NOTE:** Before adding SafeNet Authentication Service (SAS) as a RADIUS server in Linux PAM Modules, check its IP address (Primary RADIUS Server IP). The IP address will then be added to Linux PAM Modules as a RADIUS server at a later stage.

3. Under **Auth Nodes**, click **Add**.
4. Under **Add Auth Nodes**, complete the following fields, and then click **Save**:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Auth Node Name</td>
<td>Enter a name for the Auth node.</td>
</tr>
<tr>
<td>Host Name</td>
<td>Enter the name of the host that will authenticate with SAS.</td>
</tr>
<tr>
<td>Low IP Address In Range</td>
<td>Enter the IP address of the host or the lowest IP address in a range of addresses that will authenticate with SAS (in this case, a range of IP addresses is being used).</td>
</tr>
<tr>
<td>High IP Address In Range</td>
<td>Enter the highest IP address in a range of IP addresses that will authenticate with SAS (in this case, a range of IP addresses is being used).</td>
</tr>
<tr>
<td>Configure FreeRADIUS Synchronization</td>
<td>Select this option.</td>
</tr>
<tr>
<td>Shared Secret</td>
<td>Enter the shared secret key.</td>
</tr>
<tr>
<td>Confirm Shared Secret</td>
<td>Re-enter the shared secret key.</td>
</tr>
</tbody>
</table>

![Add Auth Node](image)

The authentication node is added to the system.

![Auth Nodes](image)

5. Note down the **Primary RADIUS Server IP** given under **Auth Nodes**. You will need the IP address while entering RADIUS server details on the Linux server.
CHAPTER 3: Linux PAM Modules Setup

Configuring the Linux PAM Modules to use RADIUS authentication for PAM-aware applications requires:

- Compiling the PAM Module, page 13
- Adding a RADIUS Server for the PAM Module, page 13
- Configuring the Application-Specific Configuration Files, page 15

Compiling the PAM Module

1. Log in to your Linux machine as a root user.
2. Go to the directory where the PAM module package is extracted.
   
   cd /<directory>

3. Run the following commands to compile the PAM module:
   
   make clean
   
   ./configure
   
   make

4. Run the following command to copy a PAM module (pam_radius_auth.so) to /lib64/security:
   
   cp pam_radius_auth.so /lib64/security/pam_radius_auth.so

Adding a RADIUS Server for the PAM Module

The FreeRADIUS PAM module searches for the server file in the /etc/raddb directory. The server file contains the location of the RADIUS servers, the shared secret, and the order in which each the RADIUS server is checked.

Perform the following steps to add a RADIUS server for the PAM module:

1. Verify that the /etc/raddb directory exists. If the directory does not exist, run the following command to create it:
   
   #mkdir /etc/raddb

2. A generic server configuration file, pam_radius_auth.conf exists in the directory where the PAM module package is extracted. Run the following command to move the pam_radius_auth.conf file to the /etc/raddb directory, and rename it to server:
   
   #mv pam_radius_auth.conf /etc/raddb/server
3. Run the following command to open the server file:

```
# vi /etc/raddb/server
```

# server[:port] secret [timeout]

# the port name or number is optional. The default port name is
# "radius", and is looked up from /etc/services. The timeout field is optional. The default timeout is 3 seconds.

# If multiple RADIUS server lines exist, they are tried in order. The first server to return success or failure causes the module to return success or failure. Only if a server fails to respond is it skipped, and the next server in turn is used.

# The timeout field controls how many seconds the module waits before deciding that the server has failed to respond.

# server[:port] shared_secret timeout (s)

127.0.0.1 secret 1
other-server other-secret 3
109.73.120.148:1812 1111 5

# having localhost in your radius configuration is a Good Thing.

See the INSTALL file for pam.conf hints.

4. In the server file, below `server [:port] shared_secret timeout (s)`, enter the RADIUS server details in the following format:

```
server:port shared_secret timeout
```

Where,

- **server** is the Primary RADIUS Server IP that you noted in step 5 of “Adding Linux Server as an Authentication Node”.
- **port** is the RADIUS port number. The default port number is **1812** or **1645**.
- **shared_secret** is the shared secret key that you entered earlier in step 4 of “Adding Linux Server as an Authentication Node”, on page 11.
- **timeout** is the module’s wait time before which the server fails to respond. This is optional.

**Note:** RADIUS timeout must be minimum 60 seconds for the MobilePASS+ token.

**Note:** It is recommended to configure a secondary (failover) IP address for your RADIUS equipment to cover for authentication disruptions if the service provider supports it.
5. Run the following commands to secure the server file:

```bash
#chown root /etc/raddb/
#chmod -rwx /etc/raddb
#chmod -rwx /etc/raddb/server
```

### Configuring the Application-Specific Configuration Files

Configure the PAM-aware application that you need to implement. SafeNet Authentication Service only supports the PAM-aware applications listed in the `/etc/pam.d` directory or in the `pam.conf` file. The image below lists all the applications in the `/etc/pam.d` directory. Here, the `sshd` PAM-aware file is configured.

```
[root@ip-172-31-13-98 ]# cd /etc/pam.d/
[root@ip-172-31-13-98 pam.d]# ls
chfn       password-auth      smartcard-auth     sudo-i
chsh       password-auth-ac   smartcard-auth-ac  su-l
config-util polkit-1          smtp
cron       postlogin          smtp.postfix
fingerprint-auth postlogin-ac  ssd
fingerprint-auth-ac remote    ssd Orig
login      rhn_register       system-auth
other      runuser            systemctl-user
passwd     runuser-1          systemd-user

[root@ip-172-31-13-98 pam.d]#  
```

**NOTE:** To avoid loss of any configuration, you must take back up of the existing PAM-aware file (for example, `sshd`) before deleting its’ content to test SAS authentication.

### Configuring the sshd file for Single-Factor Authentication

Perform the following steps to configure Single-Factor authentication (SFA):

1. Run the following command to open the `sshd` file located in the `/etc/pam.d` directory:

   ```bash
   #vi /etc/pam.d/sshd
   ```
2. Add the following line at top of the file after #%PAM-1.0

```
auth sufficient /lib64/security/pam_radius_auth.so
```

3. Save the file.

4. Run the following command to open the `sshd_config` file located in the `etc/ssh/` directory:

```
# vi /etc/ssh/sshd_config
```

5. Ensure that the values of the following parameters available in the `sshd_config` file are set as shown below:

- `PasswordAuthentication` yes
- `PermitEmptyPasswords` no
- `ChallengeResponseAuthentication` no
- `UsePrivilegeSeparation` no
- `UsePAM` yes

6. Run the following command to restart the SSHD services:

```
#Service sshd restart
```

### Configuring the sshd file for Two-Factor Authentication

PAM-aware applications can also be configured for two-factor authentication (2FA). Two-factor authentication can be configured in the following two ways:

- The user first enters the Linux password and then the OTP.
- The user first enters the OTP and then the Linux password.

### The user first enters the Linux password and then the OTP

1. Run the following command to open the `sshd` file located in the `etc/pam.d` directory:

```
#vi /etc/pam.d/sshd
```
2. Add the following lines at top of the file after `#PAM-1.0`

```bash
auth required /lib64/security/pam_unix.so not_set_pass
auth sufficient /lib64/security/pam_radius_auth.so
```

3. Save the file.

4. Run the following command to open the `sshd_config` file located in the `etc/ssh` directory:

   ```bash
   # vi /etc/ssh/sshd_config
   ``

5. Ensure that the values of the following parameters available in the `sshd_config` file are set as shown below:

   - `PasswordAuthentication` no
   - `PermitEmptyPasswords` no
   - `ChallengeResponseAuthentication` yes
   - `UsePrivilegeSeparation` no
   - `UsePAM` yes

6. Run the following command to restart the SSHD services:

   ```bash
   #Service sshd restart
   ```

The user first enters the OTP and then the Linux password

1. Run the following command to open the `sshd` file located in the `etc/pam.d` directory:

   ```bash
   #vi /etc/pam.d/sshd
   ```
2. Add the following line at top of the file after `#%PAM-1.0`.

```bash
auth required /lib64/security/pam_radius_auth.so
auth sufficient /lib64/security/pam_unix.so
```

```
#%PAM-1.0
auth required lib64/security/pam_radius_auth.so
auth sufficient lib64/security/pam_unix.so
auth required pam_sespermit.so
auth sufficient pam_unix.so
auth include postlogins
# Used with polkit to reauthorize users in remote sessions
-auth optional pam_reauthorize.so prepare
account required pam_nologin.so
account include password-auth
password include password-auth
# pam_selinux.so close should be the first session rule
session required pam_selinux.so close
session required pam_loginuid.so
# pam_selinux.so open should only be followed by sessions to be executed in the
user context
session required pam_selinux.so open env params
session required pam_namespace.so
session optional pam_keyinit.so force revoke
session include password-auth
session include postlogins
# Used with polkit to reauthorize users in remote sessions
+session optional pam_reauthorize.so prepare
-- INSERT --
```

3. Save the file.

4. Run the following command to open the `sshd_config` file located in the `etc/ssh/` directory:

```bash
# vi /etc/ssh/sshd_config
```

5. Ensure that the values of the following parameters available in the `sshd_config` file are set as shown below:

```
PasswordAuthentication no
PermitEmptyPasswords no
ChallengeResponseAuthentication yes
UsePrivilegeSeparation no
UsePAM yes
```

6. Run the following command to restart the SSHD services:

```
#Service sshd restart
```
For this integration, the following tokens are configured for authentication with SafeNet Authentication Service (SAS):

> MobilePASS+ (or MobilePASS)
> Push OTP
> SMS

### Using MobilePASS+ (or MobilePASS)

1. Open PuTTY.
2. On the **PuTTY Configuration** window, perform the following steps:
   a. In the **Host Name** field, enter the IP address of the Linux server.
   b. Under **Connection type**, select the **SSH** option.
   c. Click **Open**.

3. On the SSH login window, enter the user name for which the MobilePASS+ (or MobilePASS) token is provisioned.
4. Enter the Linux password and press **Enter**.
5. On the registered mobile device, disable **Autosend Passcode**.

6. On the **PIN Code** screen, enter the token PIN, and tap **OK** to generate the **OTP**.

OTP will be generated.
7. On the SSH login window, enter the OTP and press Enter.

After successful authentication, the user is logged in to the machine.

Using Push OTP

1. Open PuTTY.

2. On the PuTTY Configuration window, perform the following steps:
   a. In the Host Name field, enter the IP address of the Linux server.
   b. Under Connection type, select SSH.
   c. Click Open.

3. On the SSH login window, enter the user name for which the MobilePASS+(or MobilePASS) token is provisioned.
4. Enter the Linux password and press **Enter**.

5. On the registered mobile device, enable **Autosend Passcode**.

6. On the registered mobile device, tap **Approve** to accept the OTP request.
7. On the **PIN Code** screen, enter the token PIN, and tap **OK** to send the approval with OTP to SAS.

A success message is displayed on the mobile device.

After successful authentication, the user is logged in to the machine.
Using SMS

1. Open PuTTY.
2. On the PuTTY Configuration window, perform the following steps:
   a. In the Host Name field, enter the IP address of the Linux server.
   b. Under Connection type, select SSH.
   c. Click Open.
3. On the SSH login window, enter the user name for which the SMS token is provisioned.
4. Enter the Linux password and press Enter.
5. Press Enter to submit a challenge request to receive an OTP/TokenCode.
6. On the registered mobile number, you will receive a TokenCode.
7. Enter the TokenCode and press Enter.

After successful authentication, the user is logged in to the machine.