SafeNet Authentication Service
Integration Guide

SAS Using RADIUS Protocol with NetMotion Mobility XE
Document Information

<table>
<thead>
<tr>
<th>Document Part Number</th>
<th>007-012561-001, Rev. C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release Date</td>
<td>August 2015</td>
</tr>
</tbody>
</table>

Trademarks

All intellectual property is protected by copyright. All trademarks and product names used or referred to are the copyright of their respective owners. No part of this document may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, chemical, photocopy, recording, or otherwise, without the prior written permission of SafeNet, Inc.

Disclaimer

SafeNet makes no representations or warranties with respect to the contents of this document and specifically disclaims any implied warranties of merchantability or fitness for any particular purpose. Furthermore, SafeNet reserves the right to revise this publication and to make changes from time to time in the content hereof without the obligation upon SafeNet to notify any person or organization of any such revisions or changes.

We have attempted to make these documents complete, accurate, and useful, but we cannot guarantee them to be perfect. When we discover errors or omissions, or they are brought to our attention, we endeavor to correct them in succeeding releases of the product.

SafeNet invites constructive comments on the contents of this document. These comments, together with your personal and/or company details, should be sent to the address or email below.

<table>
<thead>
<tr>
<th>Contact Method</th>
<th>Contact Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mail</td>
<td>SafeNet, Inc.</td>
</tr>
<tr>
<td></td>
<td>4690 Millennium Drive</td>
</tr>
<tr>
<td></td>
<td>Belcamp, Maryland 21017, USA</td>
</tr>
<tr>
<td>Email</td>
<td><a href="mailto:TechPubs@safenet-inc.com">TechPubs@safenet-inc.com</a></td>
</tr>
</tbody>
</table>
# Contents

Third-Party Software Acknowledgement ................................................................. 4  
Description ............................................................................................................. 4  
Applicability ........................................................................................................... 4  
Environment .......................................................................................................... 4  
Audience ................................................................................................................. 5  
About PEAP ........................................................................................................... 5  
RADIUS-based Authentication using SAS Cloud .................................................... 5  
RADIUS-based Authentication using SAS-SPE and SAS-PCE ............................... 6  
RADIUS Authentication Flow using SAS .............................................................. 6  
RADIUS Prerequisites ......................................................................................... 7  
Single Sign-On Dataflow (SafeNet PEAP Support Package) .................................. 7  
  PEAP Checklist ....................................................................................................... 8  
  Configuring the SafeNet PEAP Package ............................................................... 8  
  SafeNet PEAP Support Package v8.0 Security Recommendations ..................... 13  
Single Sign-On Dataflow (FreeBSD) .................................................................... 14  
  FreeBSD Checklist ................................................................................................. 15  
  FreeBSD Installation ............................................................................................. 15  
  FreeBSD Configuration ......................................................................................... 20  
  FreeRADIUS Installation ...................................................................................... 22  
  FreeRADIUS Configuration .................................................................................. 23  
Joint Configuration for Both Solutions ................................................................ 25  
  Configuring Certificates ...................................................................................... 25  
Defining a RADIUS Server ................................................................................... 28  
Defining PEAP ....................................................................................................... 29  
Configuring the NetMotion Mobility Server ........................................................... 30  
Configuring the NetMotion Mobility Client ............................................................. 31  
Defining the SafeNet FreeRADIUS Agent ............................................................. 32  
Configuring SafeNet Authentication Service ........................................................ 33  
  Synchronizing Users Stores to SAS ....................................................................... 33  
  Assigning an Authenticator in SAS ....................................................................... 34  
  Adding FreeRADIUS Server as an Authentication Node in SAS ...................... 34  
Running the NetMotion Mobility XE Solution ....................................................... 35  
Appendix A ............................................................................................................ 37  
  FreeRADIUS Testing ............................................................................................ 37  
Appendix B ............................................................................................................. 38  
  Customizing a Field Name in the Mobility Logon Window .................................. 38  
  NMexport Utility ................................................................................................. 40  
Appendix C ............................................................................................................. 41  
  Configuring Failover Server in SafeNet PEAP Agent .......................................... 41  
Support Contacts .................................................................................................... 42
Third-Party Software Acknowledgement

This document is intended to help users of SafeNet products when working with third-party software, such as NetMotion Mobility XE.

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.

Description

SafeNet Authentication Service delivers a fully automated, versatile, and strong authentication-as-a-service solution. It provides smooth management processes and highly flexible security policies, token choice, and integration APIs.

The NetMotion mobility management solution provides you the connectivity, visibility, control, and diagnostics needed to solve many challenges that impede the success of a mobile deployment. NetMotion Mobility XE is the only mobile VPN software that is self-diagnosing and healing. Plus, it supports both per-app and traditional VPNs, giving you more control to get the most out of mobile investments.

This document describes how to:

- Deploy multi-factor authentication (MFA) options in NetMotion Mobility XE using SafeNet OTP authenticators managed by SafeNet Authentication Service.
- Configure NetMotion Mobility XE to work with SafeNet Authentication Service in RADIUS mode.

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

NetMotion Mobility XE 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 used in this document is based on the following software versions:

- SafeNet Authentication Service—Cloud-based authentication service
- NetMotion Mobility XE—Version 10.52
Audience

This document is targeted to system administrators who are familiar with NetMotion Mobility XE, and are interested in adding multi-factor authentication capabilities using SafeNet Authentication Service.

About PEAP

The Protected Extensible Authentication Protocol (PEAP) is a common authentication protocol for communication between a VPN server and mobile devices. PEAP encapsulates the Extensible Authentication Protocol (EAP) within an encrypted and authenticated TLS tunnel.

FreeRADIUS has a built-in feature to locally terminate the Transport Layer Security (TLS) outer tunnel, decrypt the PEAP tunnel, and then extract the inner identity to proxy the MS-CHAPv2 authentication mechanism to another RADIUS server.

PEAP creates two concentric tunnels:
- An encrypted and authenticated TLS outer tunnel
- An inner tunnel that uses an EAP method (such as EAP-MS-CHAPv2) for authentication, and is protected by the TLS outer tunnel

RADIUS-based Authentication using SAS Cloud

SAS Cloud provides two RADIUS mode topologies:
- **SAS cloud hosted RADIUS service**—A RADIUS service that is already implemented in the SAS cloud environment and can be used without any installation or configuration requirements.

- **Local RADIUS hosted on-premises**—A RADIUS agent that is implemented in the existing customer’s RADIUS environment. The agent forwards the RADIUS authentication requests to the SAS cloud environment. The RADIUS agent can be implemented on a FreeRADIUS server only.
This document demonstrates the solution using the SAS cloud hosted RADIUS service.
For more details on how to install and configure FreeRADIUS, refer to the *SafeNet Authentication Service FreeRADIUS Agent Configuration Guide*.

**RADIUS-based Authentication using SAS-SPE and SAS-PCE**

In addition to the pure cloud-based offering, SafeNet Authentication Service comes with two on-premises versions:

- **SafeNet Authentication Service – Service Provider Edition (SPE)** — An on-premises version of SafeNet Authentication Service targeted at Service Providers interested in hosting SAS in their data center.
- **SafeNet Authentication Service – Private Cloud Edition (PCE)** — An on-premises version of SafeNet Authentication Service targeted at organizations interested in hosting SAS in their private cloud environment.

For both on-premises versions, SAS can be integrated with the following solution that serves as a local RADIUS server:

- **FreeRADIUS** — The SAS FreeRADIUS Agent is a strong authentication agent that is able to communicate with SAS through the RADIUS protocol.

For more information on how to install and configure the SAS FreeRADIUS Agent, refer to the SafeNet Support Portal.

**RADIUS Authentication Flow using SAS**

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

The dataflow of a multi-factor authentication transaction for NetMotion Mobility XE is as follows:

1. A user attempts to log on to NetMotion Mobility XE using an OTP authenticator.
2. NetMotion Mobility XE sends a RADIUS request with the user’s credentials to SafeNet Authentication Service for validation.
3. The SAS authentication reply is sent back to NetMotion Mobility XE.
4. The user is granted or denied access to NetMotion Mobility XE based on the OTP value calculation results from SAS.
RADIUS Prerequisites

To enable SafeNet Authentication Service to receive RADIUS requests from NetMotion Mobility XE, ensure the following:

- End users can authenticate in the NetMotion Mobility XE environment with a static password before configuring NetMotion Mobility XE to use RADIUS authentication.
- Ports 1812/1813 are open to and from the NetMotion Mobility XE Policy server.
- 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.

Single Sign-On Dataflow (SafeNet PEAP Support Package)

The SafeNet PEAP Support Package delivers a customized FreeRADIUS (version 2.1.10) application running on a Windows platform preconfigured to terminate Phase 1 outer identity and to proxy an MS-CHAPv2 authentication mechanism to another RADIUS server.

1. Bob, a user, wants to log on to his organization using the NetMotion solution. Bob leverages the single sign-on capabilities embedded in the organization’s SafeNet Authentication Service solution.
2. The Mobility XE client collects Bob’s credentials and passes them to the NetMotion Mobility XE server.
3. The NetMotion Mobility XE server forwards the request to the FreeRADIUS server (including the SafeNet PEAP Support Package) through RADIUS protocol. The FreeRADIUS server is EAP compatible.
4. The SafeNet PEAP Support Package unwraps EAP and forwards MSCHAPv2 to the SAS FreeRADIUS agent. The SAS FreeRADIUS agent generates an API call to SAS over an HTTPS tunnel.
5. SafeNet Authentication Service (SAS) evaluates Bob’s credentials and returns an accept or reject response to the FreeRADIUS agent.
7. The SafeNet PEAP Support Package sends a RADIUS Access accept or reject response to the Mobility XE server.
8. If the OTP value is valid and the user meets the remote access policies, the user is authorized and a RADIUS Access accept response is returned. If the OTP value is not valid, or the user does not meet the remote access policies, the user is not authorized and a RADIUS Access reject response is returned.

PEAP Checklist

Do the following to enable PEAP authentication:
1. Install and configure the SafeNet PEAP Support Package
2. Configure FreeRADIUS
3. Configure certificates
4. Define a RADIUS Server
5. Define PEAP
6. Define a Mobility XE Server and client settings

Configuring the SafeNet PEAP Package

The steps required for using the SafeNet PEAP Support Package are as follows:
1. Install the SafeNet Support PEAP Package
2. Convert certificate files using OpenSSL
3. Modify the eap.conf file
4. Modify the clients.conf file
5. Modify the proxy.conf file
6. Start the FreeRADIUS service

Installing the SafeNet PEAP Support Package

1. Run the SafeNetOTPAuthenticationPlugIns-SetupManager-8.2.exe file.
2. On the Welcome window, click Next.
3. On the Plug-In Selection window, select OTP Plug-In for PEAP, and then click Install.
4. On the Welcome window, click Next.

5. On the License Agreement window, select I accept the license agreement, and then click Next.

6. On the Destination Folder window, accept the default folder or use the Browse button to select a different folder. When ready, click Next to continue.
7. The installation process begins. When the **Completing SafeNet OTP Plug-in Package 8.2 Setup Wizard** window is displayed, click **Finish**.

### Configuring Windows FreeRADIUS

The SafeNet PEAP Support Package v8.0 installs FreeRADIUS version 2.1.10, preconfigured to handle EAP termination. PEAP requires a server-side certificate to identify the FreeRADIUS server during EAP negotiation.

**NOTE:** Information regarding configuring certificates can be found in “Configuring Certificates” on page 25.

FreeRADIUS requires the issuing Root CA certificate. These certificate files should be in `.pem` format and can be placed in the following location:

`C:\ProgramFiles\SafeNet\Authentication\SafeNetPEAP\Package\FreeRADIUS\etc\raddb\certs`

The SafeNet PEAP Support Package v8.0 installation delivers the necessary OpenSSL tools to convert your Microsoft CA Root certificate, as well as your FreeRADIUS certificate issued by the same Microsoft CA, into the `.pem` format.

1. Before running the OpenSSL commands to convert the files, place both the certificate files in the OpenSSL folder at: `C:\Program Files\SafeNet\Authentication\SafeNetPEAP\Package\OpenSSL`.

**NOTE:** Windows uses the `.pfx` extension by default. Rename the FreeRADIUS certificate file with a `.p12` extension.

2. Browse to the `C:\Program Files\SafeNet\Authentication\SafeNetPEAP\Package\OpenSSL` folder, and then launch `openssl.exe` on the command prompt. Ignore the warning message shown below.
Convert Root CA Certificate to .pem

Usage: x509 -inform der -in certificate.cer -out certificate.pem

Example: x509 -inform der -in myca.cer -out myca.pem

Convert FreeRADIUS Certificate to .pem


passin-pfx password
passout-pem new password

Modifying the eap.conf File

The eap.conf file contains all relevant certificate information required for FreeRADIUS to establish a successful EAP session.

From the default path, C:\Program Files\SafeNet\Authentication\SafeNetPEAPPackage\FreeRADIUS\etc\raddb\, open the eap.conf file:

1. In the eap section, locate this line:
   default_eap_type = md5
   Change the value to peap.

2. Look for the tls section and modify your FreeRADIUS and RootCA certificates that were prepared using OpenSSL commands.

3. In the PEAP section, locate this line:
   proxy_tunneled_request_as_eap = yes
   Change the value to no.

   NOTE: You must delete the “hash” mark before the line (highlighted in green below):

   ```
   # EAP-MSCHAP-V2 as normal MSCHAPv2:
   proxy_tunneled_request_as_eap = no
   ```
Modifying the Clients.conf File

The clients.conf file contains information regarding RADIUS clients. The default path for the clients.conf file is C:\Program Files\SafeNet\Authentication\SafeNetPEAPPackage\FreeRADIUS\etc\raddb. Open the clients.conf file and then specify the NetMotion XE server as the RADIUS client.

Example:

```
client 192.168.1.10/24
{
    secret     = 1ws3fh7a0#ghfiLGC564@#$=h0q
    shortname  = SAS Server
}
```

Modifying the Proxy.conf File

The proxy.conf file contains information on the RADIUS servers that the RADIUS requests are being proxied to. The default path for the proxy.conf is:

C:\Program Files\SafeNet\Authentication\SafeNetPEAPPackage\FreeRADIUS\etc\raddb

Go to the end of the file and paste in the following:

```
home_server SAS {
    type = auth
    ipaddr = 10.10.10.21
    port = 1812
    secret = 1234
}
realm Test {
    type = radius
    authpool = my_auth_failover
}
```

Go to the home_server_pool my_auth_failover section, and search for the following line:

```
home_server = localhost
```

Replace localhost with the name of the home_server defined above. (For example, SAS)

Modifying the Users File

Open the Users file present at the location C:\Program Files\SafeNet\Authentication\SafeNetPEAPPackage\FreeRADIUS\etc\raddb. Add the following line at the end of the file and save it.

```
DEFAULT FreeRADIUS-Proxyed-To == 127.0.0.1, Proxy-To-Realm := Test
```
Starting the FreeRADIUS Service

Once all FreeRADIUS configuration has been performed as described in the previous sections, proceed to start the FreeRADIUS service from the Services snap-in.

![Services (local) screenshot](image)

SafeNet PEAP Support Package v8.0 Security Recommendations

OS Hardening

Security of the RADIUS service depends on having a secured operating system as a foundation. SafeNet recommends hardening the operating system using your organization’s system administration requirements.

Ownership of raddb

The RADIUS client.conf file contains cleartext shared secrets to authenticate client systems and devices, and therefore must have minimal ownership and access. The /etc/raddb directory (or the equivalent directory containing FreeRADIUS configuration files) must be owned by a domain admin, as well as all files contained in the directory.

To enhance security, it is possible to use SafeNet ProtectDrive for encrypting files on your system.

Permissions for raddb

Allowing write access to the RADIUS DB directory would allow deletion and replacement of the configuration files, which would undermine the entire security of the server configuration.

The /etc/raddb directory, and the configuration files in the directory, must be set to read-only access for all users except the domain admin.
Strong Shared Secrets

The security of RADIUS depends heavily on the strength of the shared secret. The shared secret string for each client must be at least 22 characters, but not more than 31 characters. The secret must not use dictionary words, or guessable patterns or variations of words. It should include a variety of special, numeric, and alphabetic characters.

Restricted Network Access

Restricting network access with IP filtering to just those systems or networks that require access reduces the risk of the most common attacks. Restrict network access using host-based IP filtering to the minimum number of networks or systems requiring access.

NOTE: Refer to the SAS FreeRADIUS Agent Configuration Guide for more information on the FreeRADIUS Agent:
http://bel1web002:9876/Files/35e0f930d3a447bfa27ff701cf8d6bab

Single Sign-On Dataflow (FreeBSD)

The FreeBSD unwraps the PEAP layer and forwards authentication to the SAS Server.

1. Bob, a user, wants to log on to his organization using the NetMotion solution. Bob leverages the single sign-on capabilities embedded in the organization’s SafeNet Authentication Service solution.
2. The Mobility XE client collects Bob's credentials and passes them to the NetMotion Mobility XE server.
3. The NetMotion Mobility XE server forwards the request to the FreeBSD server through the RADIUS protocol.
4. The FreeBSD server unwraps EAP and forwards MSCHAPv2 to the SAS server.
5. SafeNet Authentication Service (SAS) evaluates Bob's credentials and returns an accept or reject response to the FreeBSD server.
6. The FreeBSD server sends a RADIUS Access accept or reject response to the Mobility XE server.
7. If the OTP value is valid and the user meets the remote access policies, the user is authorized and a RADIUS Access accept or reject response is returned. If the OTP value is not valid, or the user does not meet the remote access policies, the user is not authorized and a RADIUS Access reject response is returned.

FreeBSD Checklist

Do the following to configure the FreeBSD solution:

1. Install and configure FreeBSD
2. Install and configure FreeRADIUS
3. Configure certificates
4. Define a RADIUS Server
5. Define PEAP
6. Define a Mobility XE Server and client settings
7. Define a SafeNet FreeRADIUS agent

FreeBSD Installation

1. Download the FreeBSD application from the following location: http://www.freebsd.org/
2. After downloading the file, launch the installer.
3. On the first FreeBSD Installer window, select Install, and then press Enter.
4. On the Keymap Selection window, select No, and then press Enter.
5. On the **Set Hostname** window, enter the hostname for this machine, and then press **Enter**.

![Set Hostname window]

6. On the **Distribution Select** window, ensure that the **Games** option is not selected, and then press **Enter**.

![Distribution Select window]

7. Complete the following steps for partitioning:
   a. On the **Partitioning** window, select **Guided**, and then press **Enter**.

![Partitioning window]

   b. On the **Partition** window, select **Entire Disk**, and then press **Enter**.

![Partition window]

   c. On the **Partition Editor** window, press the **Tab** key to highlight **Finish**, and then press **Enter**.

![Partition Editor window]

8. On the **Confirmation** window, press **Enter** to commit your changes.

![Confirmation window]
The **Archive Extraction** window appears with an extraction progress bar. This process may take a while.

![Archive Extraction](image)

9. When extraction is finished, you are returned to the FreeBSD Installer window. At the **New Password** prompt, enter a new password for the **root** account.

![FreeBSD Installer](image)

10. Complete the following steps for network configuration:

   a. Select the network interface card and then press **Enter**.

   ![Network Configuration 1](image)

   b. Select **Yes** and then press **Enter**.

   ![Network Configuration 2](image)

   c. Select **No** and then press **Enter**.

   ![Network Configuration 3](image)

   d. Enter the static IP information for the server, and then press **Enter**.

   ![Network Configuration 4](image)
e. Select **No** and then press **Enter**.

![Network Configuration](image)

f. Enter your domain and DNS information, and then press **Enter**.

![Resolver Configuration](image)

11. Complete the following steps for clock configuration:

   a. On the **Select local or UTC clock** window, select **No**, and then press **Enter**.

   ![Select local or UTC clock](image)

   b. Make the following settings as shown in the screens below, pressing **Enter** to save each screen:

   ![Time Zone Selector](image)
12. On the **System Configuration** window, select **sshd** (required) and **ntpd** (if desired), and then press **Enter**.

![System Configuration window](image1)

13. On the **Dumpdev Configuration** window, select **No**, and then press **Enter**.

![Dumpdev Configuration window](image2)

14. On the **Add User Accounts** window, select **Yes**, and then press **Enter**.

![Add User Accounts window](image3)

15. On the **Add Users** window, follow the prompts as shown:
   (For this example, the username **mobilityxe** is used.)

![Add Users window](image4)

16. On the **Final Configuration** window, select **Exit** and press **Enter**.

![Final Configuration window](image5)
17. On the **Manual Configuration** window, select **No**, and then press **Enter**.

18. On the **Complete** window, select **Reboot**, and then press **Enter**.

19. After reboot is complete, log in as the **root** user.
   a. Enter the following command to give the **mobilityxe** user root privileges:
      
      ```bash
      pw groupmod wheel -m mobilityxe
      ```
   b. Press **Enter**.

**FreeBSD Configuration**

**PuTTY Installation**

Download the PuTTY SSH client from the following location:

http://the.earth.li/~sgtatham/putty/latest/x86/putty-0.62-installer.exe

After downloading the file, launch the installer on your Windows PC.

**Server Configuration**

1. Launch PuTTY on your Windows PC.

2. On the **PuTTY Configuration** window, enter the IP/Host Name of your FreeBSD server, and then click **Open**.
3. A **Security Alert** message is displayed regarding the server’s host key. Click **Yes** to continue.

4. Enter the username **mobilityxe** and the password.

5. Type **su** and then enter the root password.

6. Enter the following commands:

   - `portsnap fetch update`
   - `portsnap fetch extract update`

**Installing the Nano Text Editor**

Install the Nano text editor using the commands below. If you prefer to use another editor, you may skip this section.

```
cd /usr/ports/editors/nano
make install clean
ln -s /usr/local/bin/nano /usr/bin/nano
```
Bash Prompt

The `bash` prompt will display the path you are currently in instead of the generic prompt that FreeBSD provides by default:

```
cd /usr/ports/shells/bash
make install clean
chsh -s /usr/local/bin/bash
```

**NOTE:** If you don’t see the prompt after logging out and back in, it may have only applied to the **root** account. Simply run the `chsh` command again if this is the case.

FreeRADIUS Installation

Use the following commands to start the FreeRADIUS installation:

```
cd /usr/ports/net/freeradius2
make install
```

This installation will take a good amount of time and will install a number of prerequisites. You should accept the default settings unless otherwise instructed.
FreeRADIUS Configuration

1. Run the following command to edit the users file:
   
   cd /usr/local/etc/
   su <enter your password>
   cd raddb
   nano users
2. Add the following to the top of the file:

```
NULL Proxy-To-Realm := LOCAL
DEFAULT FreeRADIUS-Proxied-To == 127.0.0.1, Proxy-To-Realm := Safeword
```

3. When prompted, press Ctrl+X and select Yes to save.

4. Do the following to edit the `proxy.conf` file:
   a. Run this command: `nano proxy.conf`
   b. Page down to the end of the file and paste in the following:

```
realm LOCAL {
  type = radius
  authhost = LOCAL
  accthost = LOCAL
}

realm Safeword {
  type = radius
  authhost = 172.16.0.15:1812
  accthost = 172.16.0.15:1813
  secret = 1234
}
```

   c. When prompted, press Ctrl+X and select Yes to save the file.

5. Do the following to edit the `clients.conf` file:
   a. Run this command: `nano clients.conf`
   b. Page down to the end of the file and right-click the mouse to paste in the following:

```
client 172.16.0.5 {
  secret = testing123
  shortname = nms1
}
```

   c. When prompted, press Ctrl+X and select Yes to save the file.

   **NOTE:** You will need to change the `authhost`, `accthost`, and `secret` to match your installation of SafeWord. Simply use the arrow keys and replace the text in `nano`.

   **Important!** Do not change anything in the LOCAL realm.
6. Do the following to edit the eap.conf file:
   a. Run this command: nano eap.conf
   b. In the eap section, locate this line:
      
      ```
      default_eap_type = md5
      ```
      Change the value to `peap`.
   c. In the PEAP section, locate this line:
      ```
      proxy_tunneled_request_as_eap = yes
      ```
      Change the value to `no`.

      **NOTE:** You must delete the “hash” mark before the line (highlighted in green below):

      ```
      # default_eap_type = md5
      ```
   d. When prompted, press **Ctrl+X** and select **Yes** to save the file.

### Joint Configuration for Both Solutions

Protected EAP (PEAP) adds a TLS layer on top of EAP. PEAP uses TLS to authenticate the server to the client. To achieve this, the FreeRADIUS server is required to have a server certificate. In this guide, as an example, a Microsoft CA is used, but any other CA can be designated to provide a server certificate.

### Configuring Certificates

#### Configure a Certificate Template from the CA for FreeRADIUS

1. Open the CA snap-in.
2. In the left pane, right-click **Certificate Templates**, and then select **Manage**.
3. In the certificates list, right-click the **Web Server** template, and then select **Duplicate Template**.

![Certificate Templates Console]

3. **NOTE:** You can choose any other certificate template that has a Server Authentication EKU.

4. Select **Windows Server 2003 Enterprise**, and then click **OK**.

![Certificate Templates Console 2]
5. On the **Properties of New Template** window, enter a name in the **Template display name** field, and then click the **Request Handling** tab.

![Properties of New Template](image)

6. Select the option **Allow private key to be exported**. Click **Apply** and then click **OK**.

![Properties of New Template](image)
Defining a RADIUS Server

1. Log in to the Mobility XE Server console.

![Mobility XE Server Console]

(The screen image above is from NetMotion® Wireless software. Trademarks are the property of their respective owners.)

2. On the Server Settings window, click RADIUS Server List. Then, click Add.

![Server Settings Window]

(The screen image above is from NetMotion® Wireless software. Trademarks are the property of their respective owners.)
3. Define the following entries for each RADIUS server:
   - IP Address
   - Port
   - NAS ID
   - Shared Secret

   In the **RADIUS Server Entry** logon window, enter your logon credentials, and then click **OK**.

   (The screen image above is from NetMotion® Wireless software. Trademarks are the property of their respective owners.)

**Defining PEAP**

Define a RADIUS-EAP (PEAP and EAP-TLS) authentication scheme.

1. Log in to the Mobility XE Server console.

   (The screen image above is from NetMotion® Wireless software. Trademarks are the property of their respective owners.)
2. On the **Server Settings** window, click **Authentication: User – Protocol > Protocol**, select **RADIUS EAP (PEAP and EAP-TLS)**.

### Configuring the NetMotion Mobility Server

1. Log in to the Mobility XE Server console.
2. Click **Configure > Client Settings**.
3. Click **Logon - Prompt for User Credentials at Every Reconnect**.
4. Select **Prompt at every reconnect**, and then click **Apply**.

   All tokens are enabled for one-time passwords (OTPs), which are only valid for a single use. As such, the Mobility server should never use an existing token code; the user should always be prompted.

5. Click **Configure > Client Settings**.
6. Click **Logon - Try Windows Credentials**.
7. Clear the **Automatically try** check box, and then click **Apply**.

Since the username and password that the user logs in to Windows with will never match the username and token code of the user, you do not want to waste resources trying to authenticate a user password when it will always fail.

8. Click **Configure > Server Settings**.
9. Click **Authentication: User – Protocol**.
10. Select **RADIUS – EAP (PEAP and EAP-TLS)**, and then click **Apply**.

---

**Configuring the NetMotion Mobility Client**

Install the NetMotion Mobility Client on your client. Once the user provides valid credentials, normal Mobility Client operation does not require further access to the Mobility Client user interface, and such access may even be hidden from the end user.

Mobility Client is largely transparent to the end user. Each client must have the “trusted root certificate authority” pre-installed in the client’s certificate store in order to validate the server certificate during the PEAP process.

1. Open the Mobility XE client.
2. Select **Configuration**.
3. Click the **Server Certificates** tab and clear the **Validate server certificate** check box.
4. Click OK.

(The screen image above is from NetMotion® Wireless software. Trademarks are the property of their respective owners.)

Defining the SafeNet FreeRADIUS Agent

The SafeNet RADIUS Agent is a strong authentication agent that allows RADIUS clients to communicate with SAS via the RADIUS protocol.

This agent uses an encrypted key file to communicate with the authentication server. This ensures that all authentication attempts made against the server are from valid, recognized agents.

A key file must be loaded and registered with SafeNet ID agents, and a matching key must be registered with the server.

To set up this configuration, the main steps are:

- Load the specific encrypted key file to communicate with the SafeNet ID Server.
- Configure FreeRADIUS to use the agent.

SAS FreeRADIUS Agent integrates FreeRADIUS server with the SAS authentication server. SAS developers have written an rlm plug-in for the FreeRADIUS server that has defined a reprogrammable, publishable interface. This plug-in is free and is released as a GPL program similar to FreeRADIUS.

SAS developers have also added their proprietary suite of protocols, which handle authentication with the SAS authentication server.
The following is an example of the `radiusd debug` command, resulting in successful authentication:

```
user@safenet:opt/freeRADIUS/freeRADIUS-server-2.1.11/sbin $ user@safe...bin
```

### Configuring SafeNet Authentication Service

The deployment of multi-factor authentication using SAS with NetMotion Mobility XE using the RADIUS protocol requires:

- Synchronizing Users Stores to SAS
- Assigning an Authenticator in SAS
- Adding FreeRADIUS Server as an Authentication Node in SAS

#### Synchronizing Users Stores to SAS

Before 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 further details on importing users to SafeNet Authentication Service, refer to the *SafeNet Authentication Service Subscriber Account Operator Guide*:
Assigning an Authenticator in SAS

SAS supports a number of authentication methods that can be used as a second authentication factor for users who are authenticating through NetMotion Mobility XE.

The following authenticators are supported:

- eToken PASS
- RB-1 Keypad Token
- KT-4 Token
- SMS Token
- MP-1 Software Token
- MobilePASS

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 to learn how to provision the different authentication methods to the users in the SAS user store.


Adding FreeRADIUS Server as an Authentication Node in SAS

To allow authentication requests to the SAS server, an authentication node must be created for the FreeRADIUS server on which the SafeNet FreeRADIUS agent is installed.

1. In your SAS console, select **Virtual Server > COMMS > Auth Nodes**.
2. Click **Add** and then enter at least the following:
   - A descriptive name
   - The IP address of the FreeRADIUS server
   - The FreeRADIUS shared secret (this must be identical in both SAS and on the FreeRADIUS server)
The following is an example of **Auth Nodes** settings:

![Auth Nodes screenshot](image.png)

### Running the NetMotion Mobility XE Solution

Now that NetMotion is configured to forward authentication requests to the SafeNet PEAP Support Package, which then forwards the request to SAS, users can log in to their organization using NetMotion and SAS authentication.

**NOTE:** The time on the NetMotion server, the FreeRADIUS server, and the SAS server should be NTP synchronized, as a difference in time of more than two (2) seconds will cause a failure.

1. Open the **NetMotion Mobility XE Client** window, and then select the **Status** tab.

![NetMotion Mobility XE Client screenshot](image.png)

*(The screen image above is from NetMotion® Wireless software. Trademarks are the property of their respective owners.)*
2. Click **Connect**.

3. On the **Mobility XE Logon** window, enter your enrolled OTP token to generate an OTP passcode, and then do the following:
   
   - In the **User name** field, enter your user name.
   - In the **Password** field, enter the generated OTP passcode.
   - In the **Domain** field, enter the domain in which these credentials are valid.
   - Click **OK**.

   Upon successful authentication, you are connected.

   (The screen image above is from NetMotion® Wireless software. Trademarks are the property of their respective owners.)

   Your authentication activity can be viewed in the SAS console.

   (The image shows a logon window with fields for user name, password, domain, and options for OK, Cancel, Bypass, and Help.)
Appendix A

FreeRADIUS Testing

In order to test the FreeRADIUS server, start the server in debug mode. This does two things:

- Starting the server in debug mode executes a script that will create the test certificates that contain these attributes. The server certificate must have special OID's or else the process will fail.
- Allows you to test the configuration.

Enter `radius -X` at the bash prompt. If all goes well, the last line will read as follows:

*Ready to process requests.*

If this result is not displayed, go back and check your work; you most likely entered something incorrectly.

The following is an example of the `radiusd debug` command, resulting in successful authentication:

```
[peap] Tunneled authentication was successful.
[peap] SUCCESS
*LEEP* returns handled
Sending Access-Challenge of id 231 to 10.10.10.11 port 50001
EAP-Message: 0x0b1a4a2f19b017035106622b93ceba89f188a0a28ab5f77963906a574b7e6d87d8b73716b78e8c76e2797717
Message-Authenticator = 0x00000000000000000000000000000000
State = [0a7a8a39022c3f81420b9458e4]
Finished request 49.
Going to the next request
Cleaning up request 40 ID 222 with timestamp 1663
Cleaning request 41 ID 222 with timestamp 1663
Waking up in 8.6 seconds.
rad_recv: Access-Request packet from host 10.10.10.11 port 50001, id=232, length 182

MS-Identifier = "NETMOTION"
User-Name = "SAFEINDEX\free"
Calling-Station-Id = "000000B0C0563"
EAP-Message = 0x0b1a4a2f19b017035106622b93ceba89f188a0a28ab5f77963906a574b7e6d87d8b73716b78e8c76e2797717
Message-Authenticator = 0x00000000000000000000000000000000
State = [0a7a8a39022c3f81420b9458e4]
# Executing section authorize from file etc/raddb/sites-enabled/default
+[-processes] returns ok
+[-empty] returns moop
[peap] EAP packet type response id 10 length 80
[peap] Continuing tunnel setup.
+[-easy] returns ok
Found multi-Type = EAP
# Executing group from file etc/raddb/sites-enabled/default
+[-processes] returns ok
[peap] processing type peap
[peap] processing EAP-TLS
[peap] eaptls_verify returned 7
[peap] Done initial handshake
[peap] eap_tls_process returned 7
[peap] EAP-TLS_OK
[peap] Session established. Decoding tunneled attributes.
[peap] EAP state send Auth success
[peap] Received EAP-TLS response.
[peap] Success
[peap] Padding handler
+[-easy] returns ok
Login OK: [SAFEINDEX\free] via Auth-Type = EAP
\{ from client MobilityXE port 0 cl1 000000B0C0563\}
# Executing section post-auth from file etc/raddb/sites-enabled/default
+[-processes] returns ok
```
Appendix B

Customizing a Field Name in the Mobility Logon Window

NetMotion XE 10.52 includes a feature that allows customizing the Password prompt string in the Mobility Logon window (for NetMotion Client 10.51).

To customize the Password prompt string in the Mobility Logon window:

1. Open the NetMotion Mobility Console.

(The screen image above is from NetMotion® Wireless software. Trademarks are the property of their respective owners.)
2. On the top menu, click **Configure > Client Settings.**

   ![Client Settings Screen](image)

   *(The screen image above is from NetMotion® Wireless software. Trademarks are the property of their respective owners.)*

3. In the left pane, click **Global Client Settings.**

4. In the middle pane, expand **Logon** and then click **Customize Prompt.**

5. In the right pane, under **Logon – Customize Prompt,** in the **Global Setting** field, enter the new string, which will appear as the field name in the Mobility Logon window.

   ![Customize Prompt Screen](image)

   *(The screen image above is from NetMotion® Wireless software. Trademarks are the property of their respective owners.)*
6. On a client machine, using NetMotion Client 10.51 or above, connect to the Mobility XE server once in order to download the new field name (the changes will appear only after you have connected).

nmexport utility

In some cases, certain settings need to be configured on the client before connecting (for example, the **Password** prompt string in the Mobility Logon window). The NMexport utility was created for exporting the client configuration to a file so that it will be available for importing on other clients.

1. Configure a ‘master’ Mobility client with the settings you want (including the new **Mobility Logon** window field name).

   Export the Registry information from the master client and store it in an .inf file by running the following DOS prompt command (the default location for the utility is **C:\Program Files\NetMotion Client\**):

   ```
   nmexport.exe -f <filename>.inf
   ```

   - The **NMexport** utility does not export the client’s PID value (the unique identifier for each Mobility client).
   - Make sure you have write permissions on the **Netmotion Client** folder before executing the command.

2. Import the saved Registry settings to one or more Mobility client computers by copying the exported .inf file to your target Mobility client(s), and then right-click the .inf file and select **Install**.

   *(The screen image above is from NetMotion® Wireless software. Trademarks are the property of their respective owners.)*
Appendix C

Configuring Failover Server in SafeNet PEAP Agent

This section shows how a failover server can be added in the SafeNet PEAP Agent.

1. Open the `proxy.conf` file located at `C:\Program Files\SafeNet\Authentication\SafeNetPEAPPackage\FreeRADIUS\etc\raddb`.
2. Define a failover `home_server`. For example:
   ```
   home_server failover_1 {
       type = auth
       ipaddr = 10.10.10.18
       port = 1812
       secret = 1234
   }
   ```
   where `ipaddr` is the IP address of the failover RADIUS server and `secret` should be same as entered in the main `home_server` (for example, SAS).
3. Go to the `home_server_pool my_auth_failover` section and look for the following line:
   ```
   home_server = SAS
   ```
   Add the following line below it:
   ```
   home_server = failover_1
   ```
Support Contacts

If you encounter a problem while installing, registering, or operating this product, please make sure that you have read the documentation. If you cannot resolve the issue, contact your supplier or SafeNet Customer Support. SafeNet 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 SafeNet and your organization. Please consult this support plan for further information about your entitlements, including the hours when telephone support is available to you.

<table>
<thead>
<tr>
<th>Contact Method</th>
<th>Contact Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address</td>
<td>SafeNet, Inc.</td>
</tr>
<tr>
<td></td>
<td>4690 Millennium Drive</td>
</tr>
<tr>
<td></td>
<td>Belcamp, Maryland 21017 USA</td>
</tr>
<tr>
<td>Phone</td>
<td>United States 1-800-545-6608</td>
</tr>
<tr>
<td></td>
<td>International 1-410-931-7520</td>
</tr>
<tr>
<td>Technical Support</td>
<td><a href="https://serviceportal.safenet-inc.com">https://serviceportal.safenet-inc.com</a></td>
</tr>
<tr>
<td>Customer Portal</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Existing customers with a Technical Support</td>
</tr>
<tr>
<td></td>
<td>Customer Portal account can log in to manage</td>
</tr>
<tr>
<td></td>
<td>incidents, get the latest software upgrades, and</td>
</tr>
<tr>
<td></td>
<td>access the SafeNet Knowledge Base.</td>
</tr>
</tbody>
</table>