SafeNet Authentication Service PCE/SPE and SafeNet Trusted Access (STA)

INTEGRATION GUIDE: USING RADIUS PROTOCOL FOR AIX PAM_RADIUS

Document Information

Product Version	1.0
Document Part Number	007-013638-001
Release Date	September 2019

Revision History

Revision	Date	Reason
C	September 2019	Updating Changes

Trademarks, Copyrights, and Third-Party Software

Disclaimer

All information herein is either public information or is the property of and owned solely by Gemalto NV. and/or 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 of Gemalto's information.

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

> The copyright notice below, the confidentiality and proprietary legend and this full warning notice appear in all copies.

> This document shall not be posted on any 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, Gemalto 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, Gemalto reserves the right to make any change or improvement in the specifications data, information, and the like described herein, at any time.

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

Third-Party Software Acknowledgement

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

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:

Description

SafeNet Authentication Service (SAS (PCE/SPE)) and SafeNet Trusted Access (STA delivers a fully automated, versatile, and strong authentication-as-a-service solution.

With no infrastructure required, SAS (PCE/SPE) and STA provides smooth management processes and highly flexible security policies, token choice, and integration APIs.

AIX pam_radius is the PAM to RADIUS authentication module. It allows machine to become a RADIUS client for authentication and password change requests.

This document describes how to:

> Deploy multi-factor authentication (MFA) options in AIX pam_radius using SafeNet one-time password (OTP) authenticators managed by SAS (PCE/SPE) and STA.

> Configure AIX pam_radius to work with SAS (PCE/SPE) and STA in RADIUS mode.

It is assumed that the AIX pam_radius environment is already configured and working with static passwords prior to implementing the multi-factor authentication using SAS (PCE/SPE) and STA.

AIX pam_radius can be configured to support multi-factor authentication in several modes. The RADIUS protocol will be used for the purpose of working with SAS (PCE/SPE) and STA.

Applicability

The information in this document applies to:

> **SafeNet Trusted Access (STA)**—SafeNet’s cloud-based authentication and access management 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)—**Version 3.4.316.26039
- **AIX pam_radius**—Version 1.40 on AIX 6.1

RADIUS Prerequisites

To enable SAS (PCE/SPE) and STA to receive RADIUS requests from AIX pam_radius, ensure the following:

- End users can authenticate from the AIX pam_radius with a static password before configuring the AIX pam_radius to use RADIUS authentication.
- Ports 1812/1813 are open to and from the AIX pam_radius.

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.

Pam-auth Prerequisites

To enable radius authentication we need to install and configure pam_radius on AIX. Following are the dependencies for pam_radius:

- Gcc
- gcc-c++
- gcc-cpp
- gettext
- gmp
- gmp-devel
- info
- libcommon
- libcommon-devel
- libgcc
- libmpc

- libisgsegv
- libsigsegv-devel
- libstdc++
- libstdc++-devel
- lzlib
- lzlib-devel
- m4
- mpfr
- mpfr-devel
- zlib
- zlib-devel

Download the above prerequisites in /tmp directory then go to /tmp directory using “cd /tmp” command and by rpm command, install above packages(we will be needing root user permissions for this). Following is example:

$ cd /tmp

$ rpm -Uvh gcc-4.8.3-1.aix7.1.ppc.rpm gcc-c++-4.8.3-1.aix7.1.ppc.rpm gettext-0.10.40-8.aix5.2.ppc.rpm gmp-6.0.0a-1.aix5.1.ppc.rpm gmp-devel-6.0.0a-1.aix5.1.ppc.rpm info-5.1-2.aix5.1.ppc.rpm libcommon-0.97.3-1.aix5.1.ppc.rpm libcommon-devel-0.97.3-1.aix5.1.ppc.rpm libgcc-4.8.3-1.aix7.1.ppc.rpm libmpc-1.0.3-1.aix5.1.ppc.rpm libstdc++-4.8.3-1.aix7.1.ppc.rpm libstdc++-devel-4.8.3-1.aix7.1.ppc.rpm m4-1.4.17-1.aix5.1.ppc.rpm gcc-cpp-4.8.3-1.aix7.1.ppc.rpm libsigsegv-2.10-1.aix5.2.ppc.rpm libsigsegv-devel-2.10-1.aix5.2.ppc.rpm lzlib-1.6-1.aix5.1.ppc.rpm lzlib-devel-1.6-1.aix5.1.ppc.rpm mpfr-3.1.3-1.aix5.1.ppc.rpm mpfr-devel-3.1.3-1.aix5.1.ppc.rpm zlib-1.2.4-2.aix5.1.ppc.rpm zlib-devel-1.2.4-2.aix5.1.ppc.rpm

Audience

This document is targeted to system administrators who are familiar with AIX pam_radius, and are interested in adding multi-factor authentication capabilities using SAS (PCE/SPE) and STA.

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.

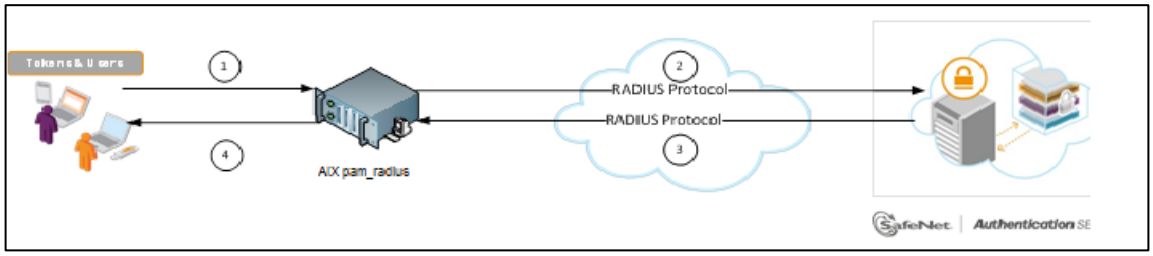
Email Support

You can also contact technical support by email at technical.support@gemalto.com.

CHAPTER 1: Authentication Flow

SAS (PCE/SPE) and STA 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 AIX pam_radius.



1. A user attempts to log on to AIX pam_radius using an OTP authenticator.
2. AIX pam_radius sends a RADIUS request with the user's credentials to SAS (PCE/SPE) or STA for validation.
3. The SAS (PCE/SPE) or STA authentication reply is sent back to the AIX pam_radius.
4. The user is granted or denied access to the AIX pam_radius based on the OTP value calculation results from SAS (PCE/SPE) or STA.

For SafeNet Trusted Access (STA), a RADIUS agent is already configured and can be used without any additional agent installation or configuration requirements.

For SafeNet Authentication Service (PCE/SPE), a RADIUS agent (SafeNet 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 SafeNet Agent for Microsoft IAS, Microsoft NPS, and FreeRADIUS, refer to the Agent Documentation.

CHAPTER 2: SAS/STA Setup

The deployment of multi-factor authentication using SAS (PCE/SPE) and STA with AIX pam_radius using RADIUS protocol requires the following:

> "Creating Users Stores", page 11

> "Assigning an Authenticator", page 12

> "Adding AIX pam_radius as an Authentication Node", page 12

Creating Users Stores

Before SAS (PCE/SPE) and STA can authenticate any user in your organization, you need to create a user store in SAS (PCE/SPE) and STA that reflects the users that would need to use multi-factor authentication. User records are created in the SAS (PCE/SPE) and STA 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 SafeNet Synchronization Agent

For additional details on importing users to SAS (PCE/SPE) and STA, refer to "Creating Users" in the "*SafeNet Authentication Service Subscriber Account Operator Guide"* available here.

All SAS (PCE/SPE) and STA documentation can be found on the SafeNet Knowledge Base site.

Assigning an Authenticator

SAS (PCE/SPE) and STA supports a number of authentication methods that can be used as a second authentication factor for users who are authenticating through AIX pam_radius.

The following authenticators are supported:

- MobilePASS
- eToken PASS
- SMS Token
- 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 (PCE/SPE) and STA 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 (PCE/SPE) and STA user store.

Adding AIX pam_radius as an Authentication Node

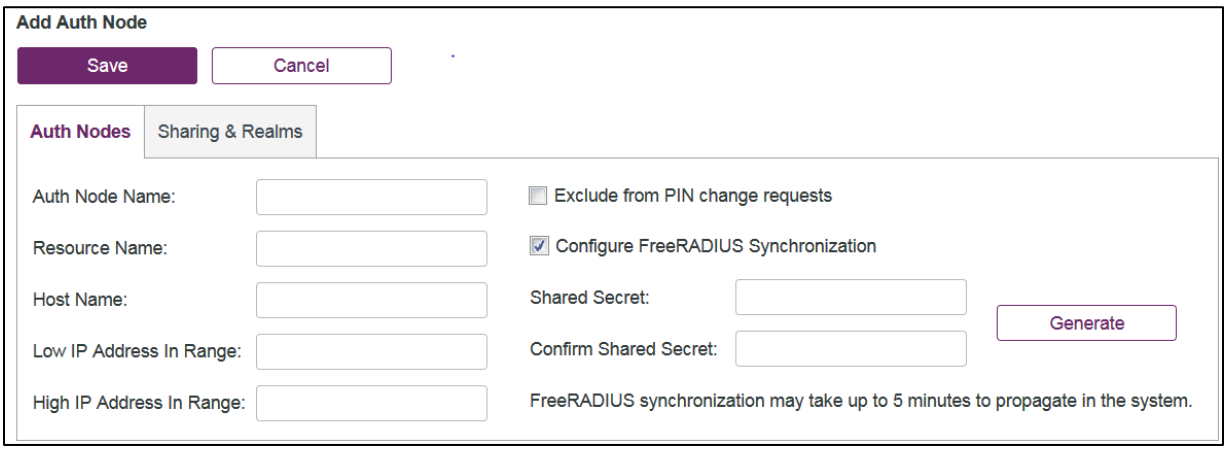
Add a RADIUS entry in the SAS (PCE/SPE) or STA **Auth Nodes** module to prepare it to receive RADIUS authentication requests from AIX pam_radius. You will need the IP address of AIX pam_radius and the shared secret to be used by both SAS/STA and AIX pam_radius.

1. Log in to the SAS (PCE/SPE) or STA 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.

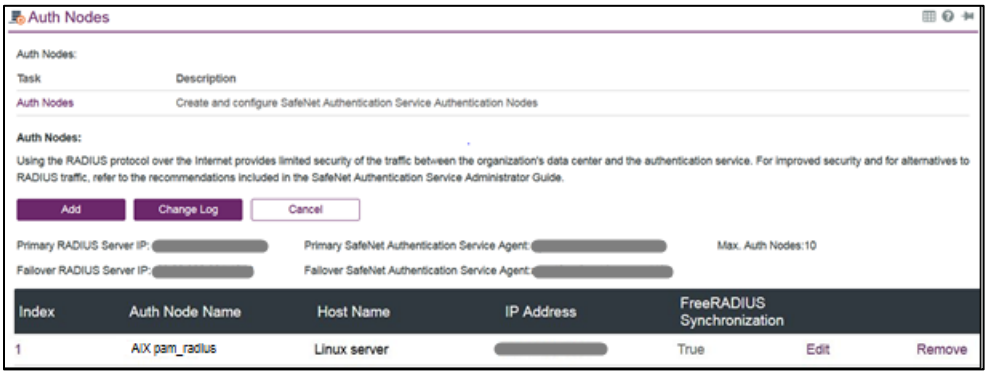
NOTE: Before adding SAS (PCE/SPE) or STA as a RADIUS server in AIX pam_radius, check its IP address (Primary RADIUS Server IP). The IP address will then be added to AIX pam_radius as a RADIUS server at a later stage.

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

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



The authentication node is added to the system.



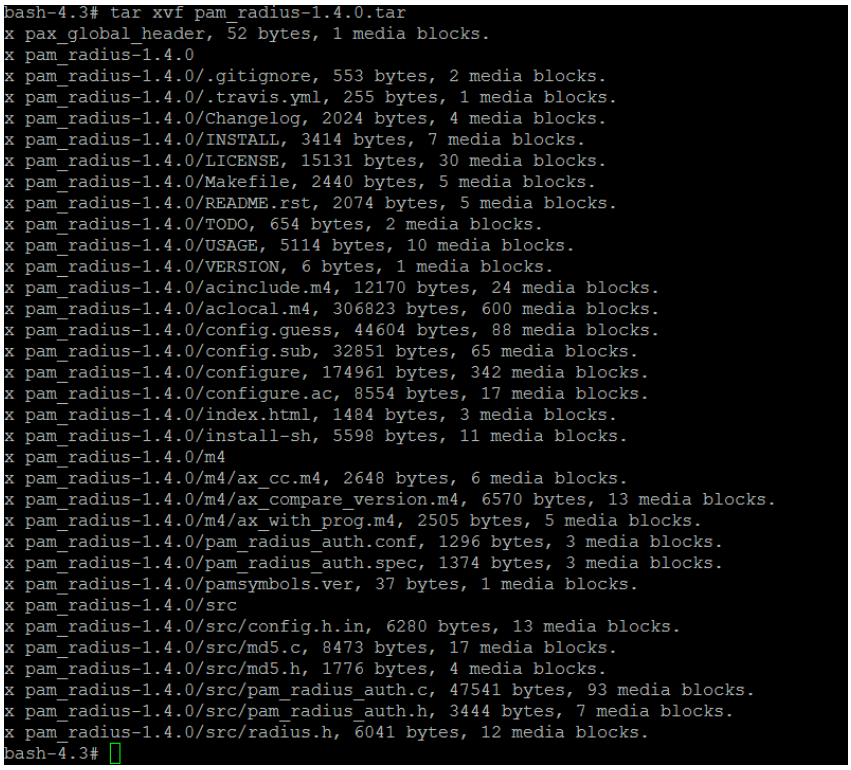
CHAPTER 3: AIX pam_radius Setup

1. Perform the following steps to download AIX pam_radius:
   a. In a web browser, open the following link to download Pamradius 1.4.0 rpm:

      http://ftp.cc.uoc.gr/mirrors/ftp.freeradius.org/

   b. Select **pam_radius-1.4.0.tar.gz** from the given list of files. The file size is 175K.
   c. Download and transfer the software from your windows machine to the AIX machine using winscp. If you use any other suitable software, it is recommended that you transfer your software to **/tmp** or **/root** path.
   d. Log in as the root user before you proceed for configuration.
2. Perform the following steps to compile **pam_radius 1.4.0**:
   a. Go to the folder where you have downloaded **pam_radius-1.4.0.tar.gz**.
   b. Run the following commands to untar the file.

      gunzip pam_radius-1.4.0.tar.gz
      tar xvf pam_radius-1.4.0.tar



```
bash-4.3# tar xvf pam_radius-1.4.0.tar
x pax_global_header, 52 bytes, 1 media blocks.
x pam_radius-1.4.0
x pam_radius-1.4.0/.gitignore, 553 bytes, 2 media blocks.
x pam_radius-1.4.0/.travis.yml, 255 bytes, 1 media blocks.
x pam_radius-1.4.0/Changelog, 2024 bytes, 4 media blocks.
x pam_radius-1.4.0/INSTALL, 3414 bytes, 7 media blocks.
x pam_radius-1.4.0/LICENSE, 15131 bytes, 30 media blocks.
x pam_radius-1.4.0/Makefile, 2440 bytes, 5 media blocks.
x pam_radius-1.4.0/README.rst, 2074 bytes, 5 media blocks.
x pam_radius-1.4.0/TODO, 654 bytes, 2 media blocks.
x pam_radius-1.4.0/USAGE, 5114 bytes, 10 media blocks.
x pam_radius-1.4.0/VERSION, 6 bytes, 1 media blocks.
x pam_radius-1.4.0/acinclude.m4, 12170 bytes, 24 media blocks.
x pam_radius-1.4.0/aclocal.m4, 306823 bytes, 600 media blocks.
x pam_radius-1.4.0/config.guess, 44604 bytes, 88 media blocks.
x pam_radius-1.4.0/config.sub, 32851 bytes, 65 media blocks.
x pam_radius-1.4.0/configure, 174961 bytes, 342 media blocks.
x pam_radius-1.4.0/configure.ac, 8554 bytes, 17 media blocks.
x pam_radius-1.4.0/index.html, 1484 bytes, 3 media blocks.
x pam_radius-1.4.0/install-sh, 5598 bytes, 11 media blocks.
x pam_radius-1.4.0/m4
x pam_radius-1.4.0/m4/ax_cc.m4, 2648 bytes, 6 media blocks.
x pam_radius-1.4.0/m4/ax_compare_version.m4, 6570 bytes, 13 media blocks.
x pam_radius-1.4.0/m4/ax_with_prog.m4, 2505 bytes, 5 media blocks.
x pam_radius-1.4.0/pam_radius_auth.conf, 1296 bytes, 3 media blocks.
x pam_radius-1.4.0/pam_radius_auth.spec, 1374 bytes, 3 media blocks.
x pam_radius-1.4.0/pamsymbols.ver, 37 bytes, 1 media blocks.
x pam_radius-1.4.0/src
x pam_radius-1.4.0/src/config.h.in, 6280 bytes, 13 media blocks.
x pam_radius-1.4.0/src/md5.c, 8473 bytes, 17 media blocks.
x pam_radius-1.4.0/src/md5.h, 1776 bytes, 4 media blocks.
x pam_radius-1.4.0/src/pam_radius_auth.c, 47541 bytes, 93 media blocks.
x pam_radius-1.4.0/src/pam_radius_auth.h, 3444 bytes, 7 media blocks.
x pam_radius-1.4.0/src/radius.h, 6041 bytes, 12 media blocks.
bash-4.3#
```

(The screen image above is from F5 Networks® software. Trademarks are the property of their respective owners.)

   c. Perform the following steps to modify the **pam_radius-1.4.0/src/pam_radius_auth.h** file.

i. Run the following commands:

cd pam_radius-1.4.0

vi src/pam_radius_auth.h

ii. On line 80, add **"# define __sun"**, just before **#ifndef CONST**
There are two underscores before **sun** and there is a space between **define** and underscores.

After modification, the code will be changed to:

/*

* Platform specific defines

***/**

#define __sun
#ifndef CONST
 # if defined(__sun) || defined(__linux__) || defined(__FreeBSD__) || defined(__APPLE__)
/*

iii. Save the file.

d. Run the following commands to configure and compile.

i. **bash-4.3#./configure**

```
checking pam/pam_appl.h presence... no
checking for pam/pam_appl.h... no
checking security/pam_modules.h usability... yes
checking security/pam_modules.h presence... yes
checking for security/pam_modules.h... yes
checking pam/pam_modules.h usability... no
checking pam/pam_modules.h presence... no
checking for pam/pam_modules.h... no
checking for net/if.h... yes
checking for off_t... yes
checking for pid_t... yes
checking for size_t... yes
checking for uid_t in sys/types.h... yes
checking for socklen_t... yes
checking for uint8_t... yes
checking for uint16_t... yes
checking for uint32_t... yes
checking for uint64_t... yes
checking for snprintf... yes
checking for inet_aton... yes
checking for inet_pton... yes
checking for inet_ntop... yes
checking for strlcat... no
checking for strlcpy... no
checking for struct in6_addr... yes
checking whether byte ordering is bigendian... (cached) yes
checking for an ANSI C-conforming const... yes
checking for the compiler flag "-Wdocumentation"... no
checking if building with -DNDEBUG... no
configure: creating ./config.status
config.status: creating src/config.h
bash-4.3#
```

(The screen image above is from F5 Networks® software. Trademarks are the property of their respective owners.)

ii. **bash-4.3#make**

```
bash-4.3# make
        cc -Wall -fPIC -c src/pam_radius_auth.c -o pam_radius_auth.o
/opt/IBM/xlc/13.1.0/bin/.orig/cc: 1501-289 (W) Option -Wall was incorrectly specified. The option will be ignored.
"src/pam_radius_auth.h", line 84.2: 1506-218 (E) Unknown preprocessing directive #def.
"src/pam_radius_auth.c", line 1061.16: 1506-343 (S) Redeclaration of pam_sm_authenticate differs from previous declaration on line 36 of "/usr/include/security/pam_modules.h".
"src/pam_radius_auth.c", line 1061.16: 1506-377 (I) The type "char**" of parameter 4 differs from the previous type "const char**".
"src/pam_radius_auth.c", line 1284.16: 1506-343 (S) Redeclaration of pam_sm_setcred differs from previous declaration on line 43 of "/usr/include/security/pam_modules.h".
"src/pam_radius_auth.c", line 1284.16: 1506-377 (I) The type "char**" of parameter 4 differs from the previous type "const char**".
"src/pam_radius_auth.c", line 1379.16: 1506-343 (S) Redeclaration of pam_sm_open_session differs from previous declaration on line 57 of "/usr/include/security/pam_modules.h".
"src/pam_radius_auth.c", line 1379.16: 1506-377 (I) The type "char**" of parameter 4 differs from the previous type "const char**".
"src/pam_radius_auth.c", line 1384.16: 1506-343 (S) Redeclaration of pam_sm_close_session differs from previous declaration on line 64 of "/usr/include/security/pam_modules.h".
"src/pam_radius_auth.c", line 1384.16: 1506-377 (I) The type "char**" of parameter 4 differs from the previous type "const char**".
"src/pam_radius_auth.c", line 1393.16: 1506-343 (S) Redeclaration of pam_sm_chauthtok differs from previous declaration on line 88 of "/usr/include/security/pam_modules.h".
"src/pam_radius_auth.c", line 1393.16: 1506-377 (I) The type "char**" of parameter 4 differs from the previous type "const char**".
"src/pam_radius_auth.c", line 1619.16: 1506-343 (S) Redeclaration of pam_sm_acct_mgmt differs from previous declaration on line 50 of "/usr/include/security/pam_modules.h".
"src/pam_radius_auth.c", line 1619.16: 1506-377 (I) The type "char**" of parameter 4 differs from the previous type "const char**".
make: 1254-004 The error code from the last command is 1.
```

(The screen image above is from F5 Networks® software. Trademarks are the property of their respective owners.)

e. Run the following command:

gcc -fPIC -c src/pam_radius_auth.c -o pam_radius_auth.o

```
bash-4.3# gcc -fPIC -c src/pam_radius_auth.c -o pam_radius_auth.o
In file included from src/pam_radius_auth.c:62:0:
src/pam_radius_auth.h:84:2: error: invalid preprocessing directive #def
 #def __sun
  ^
src/pam_radius_auth.c:1061:16: error: conflicting types for 'pam_sm_authenticate'
 PAM_EXTERN int pam_sm_authenticate(pam_handle_t *pamh,int flags,int argc,CONST char **argv)
                ^
In file included from src/pam_radius_auth.h:31:0,
                 from src/pam_radius_auth.c:62:
/usr/include/security/pam_modules.h:36:1: note: previous declaration of 'pam_sm_authenticate' was here
 pam_sm_authenticate(
 ^
src/pam_radius_auth.c:1284:16: error: conflicting types for 'pam_sm_setcred'
 PAM_EXTERN int pam_sm_setcred(pam_handle_t *pamh,int flags,int argc,CONST char **argv)
                ^
In file included from src/pam_radius_auth.h:31:0,
                 from src/pam_radius_auth.c:62:
/usr/include/security/pam_modules.h:43:1: note: previous declaration of 'pam_sm_setcred' was here
 pam_sm_setcred(
 ^
src/pam_radius_auth.c:1379:16: error: conflicting types for 'pam_sm_open_session'
 PAM_EXTERN int pam_sm_open_session(pam_handle_t *pamh, int flags, int argc, CONST char **argv)
                ^
In file included from src/pam_radius_auth.h:31:0,
                 from src/pam_radius_auth.c:62:
/usr/include/security/pam_modules.h:57:1: note: previous declaration of 'pam_sm_open_session' was here
 pam_sm_open_session(
 ^
src/pam_radius_auth.c:1384:16: error: conflicting types for 'pam_sm_close_session'
 PAM_EXTERN int pam_sm_close_session(pam_handle_t *pamh, int flags, int argc, CONST char **argv)
                ^
In file included from src/pam_radius_auth.h:31:0,
                 from src/pam_radius_auth.c:62:
/usr/include/security/pam_modules.h:64:1: note: previous declaration of 'pam_sm_close_session' was here
 pam_sm_close_session(
 ^
src/pam_radius_auth.c:1393:16: error: conflicting types for 'pam_sm_chauthtok'
 PAM_EXTERN int pam_sm_chauthtok(pam_handle_t *pamh, int flags, int argc, CONST char **argv)
                ^
In file included from src/pam_radius_auth.h:31:0,
                 from src/pam_radius_auth.c:62:
/usr/include/security/pam_modules.h:88:1: note: previous declaration of 'pam_sm_chauthtok' was here
 pam_sm_chauthtok(
 ^
src/pam_radius_auth.c:1619:16: error: conflicting types for 'pam_sm_acct_mgmt'
```

(The screen image above is from F5 Networks® software. Trademarks are the property of their respective owners.)

NOTE: Ignore the warnings appearing on screen after you execute the command.

f. Run the following command:

gcc -shared pam_radius_auth.o md5.o -lpam -lc -o pam_radius_auth.sos

3. After the compilation is complete, copy the pam_radius_auth.so file to **/usr/lib/security/ cp pam_radius_auth.so /usr/lib/security/**

4. Run the following commands to configuring the RADIUS server in pam_radius:

mkdir /etc/raddb
cp pam_radius_auth.conf /etc/raddb/server
chown root /etc/raddb
chmod go-rwx /etc/raddb
chmod go-rwx /etc/raddb/server

5. Add the RADIUS server hostname or IP Address in **/etc/raddb/server** in following format:

   radius_server <secret code> <timemout>

6. Enable SSH for pam_radius authentication using PAM. Add the following lines at the end of **/etc/pam.conf** to enable ssh to use pam_radius:

   #SSHD
   sshd auth required /usr/lib/security/pam_radius_auth.so
   sshd account required /usr/lib/security/pam_aix
   sshd password required /usr/lib/security/pam_aix
   sshd session required /usr/lib/security/pam_aix

7. Modify the **/etc/security/login.cfg** file. Change **"auth_type = STD_AUTH"** to **"auth_type = PAM_AUTH"**.

8. Update the following parameter in **/etc/ssh/sshd_config**:

   PasswordAuthentication no
   PermitEmptyPasswords no
   UsePrivilegeSeparation no
   ChallengeResponseAuthentication yes
   UsePAM yes

9. Run the following command to restart the sshd service:

   stopsrc -s sshd ; startsrc -s sshd

CHAPTER 4: Running the Solution

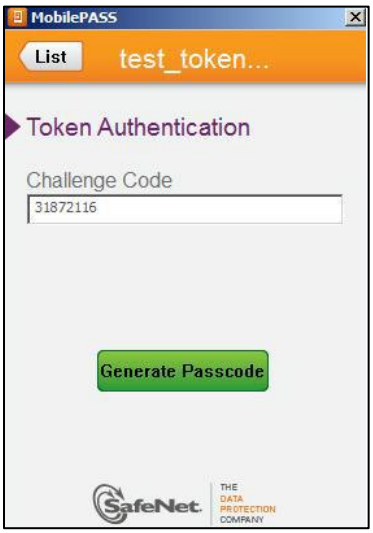
For this integration, the Mobile Pass token in the challenge response mode is configured for authentication with the SAS (PCE/SPE) and STA solution. Before running the solution, ensure that the SSH service is running on the client machine.

1. Login to the client machine, and then enter **ssh tom@127.0.0.1**.
2. In the **Password**, enter any one number, and then press **<Enter>**. You will receive a challenge code.

```
bash-4.3# ssh tom@127.0.0.1
Password:
Please respond to the challenge: 31872116
```

(The screen image above is from F5 Networks® software. Trademarks are the property of their respective owner

3. On the registered mobile device, on the **Token Authentication** screen, enter the challenge code, and then tap **Generate Passcode**. A passcode will be generated.

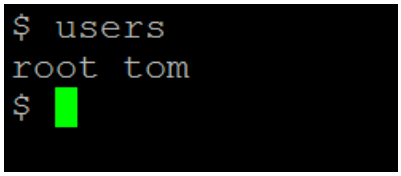


4. Enter the passcode after the challenge code on AIX. You will receive a welcome message as shown below.

```
Last unsuccessful login: Wed Sep  9 15:47:44 CDT 2015 on ssh
Last login: Wed Sep  9 15:52:37 CDT 2015 on /dev/pts/1
*******************************************************************************
*                                                                             *
*                                                                             *
*  Welcome to AIX Version 6.1!                                                *
*                                                                             *
*                                                                             *
*  Please see the README file in /usr/lpp/bos for information pertinent to    *
*  this release of the AIX Operating System.                                  *
*                                                                             *
*                                                                             *
*******************************************************************************
```

(The screen image above is from F5 Networks® software. Trademarks are the property of their respective owners.)

5. Run users command to check users. It shows the current users who are logged in to the system.



(The screen image above is from F5 Networks® software. Trademarks are the property of their respective owners.)