Procedure for Setting up United States Department of Defense Common Access Cards on a Linux System Using PC/SC and CoolKey

Kenneth L. Van Alstyne, Jr.
kenneth.vanalstyne@nrlssc.navy.mil

United States Naval Research Laboratory
Code 7321
Stennis Space Center, MS

Last Modified July 20, 2006

This page is intentionally left blank.

Introduction

- Acronyms explained:
  - ATR - Answer To Reset; string usually used to identify a smart card.
  - CAC - Common Access Card; a DoD issued smart card.
  - CVS - Concurrent Versions System; a version control system.
  - GCC - GNU Compiler Collection; collection of compilers produced by the GNU project.
  - MUSCLE - Movement for the Use of Smart Cards in a Linux Environment; a collection of projects that brings smart card functionality to Linux.

- This document covers the installation of the necessary packages to gain CAC functionality in Linux. It has been tested on (at least) Fedora Core 5 on i686, Fedora Core 5 on x86_64, and Slackware 10.2 on i686. This document should apply to all distributions.

- CoolKey is being used instead of the "traditional" MUSCLE
  - One of The most notable reasons is that no elevated privileges are needed each time a new card is inserted into a particular machine. With the MUSCLE project, the card's ATR must be added to the configuration file every time a new card is inserted into a system.
  - All of the code required for this project is GPL and LGPL compliant. The MUSCLE project requires a piece of Apple Public Source Licensed software to be functional.
  - No custom patches are necessary to gain functionality. Patches to libmusclepkcs11 are required to use the MUSCLE project.

- Acknowledgments:
  Roy Keene at the Naval Research Laboratory, Stennis Space Center, MS
  Phil Hopfner at the Naval Postgraduate School, Monterey, CA
  John Ferreira at the Naval Research Laboratory, Monterey, CA

- You will need a working GCC development environment for these procedures to work.

Procedure

1. Download the appropriate packages

(Noted are what versions successfully used and where to get them. " wget" is an excellent tool to handle the downloading. You may need to specify " --no-check-certificate " if you use wget to download from the https sites.)

libusb - Version Used (0.1.12) (Your system may include this already. If so, no need to download it again.)
Purpose: Userspace library to communicate with USB devices in Linux.
Project URL: " http://libusb.sourceforge.net/ "
Download URL: " http://superb-east.dl.sourceforge.net/sourceforge/libusb/libusb-0.1.12.tar.gz "

pcsc-lite - Version Used (1.3.1)
Purpose: Daemon and libraries to provide smart card support on UNIX.
Project URL: " http://pcsclite.alioth.debian.org/ "
Download URL: " https://alioth.debian.org/download.php/1565/pcsc-lite-1.3.1.tar.gz "

pcsc-tools - Version Used (1.4.5)
Purpose: Tools for managing and communicating with smart cards through PC/SC.
Project URL: " http://ludovic.rousseau.free.fr/softwares/pcsc-tools/ "
Download URL: " http://ludovic.rousseau.free.fr/softwares/pcsc-tools/pcsc-tools-1.4.5.tar.gz "

ccid - Version Used (1.0.1)
Purpose: Generic smart card device drivers for PC/SC.
Project URL: " http://pcsclite.alioth.debian.org/ccid.html "
Download URL: " https://alioth.debian.org/download.php/1563/ccid-1.0.1.tar.gz "

CoolKey - Version Used (CVS 1.0.2 as of July 18, 2006)
Purpose: Libraries for use by clients to communicate to the smart card.
- As of the time of this writing, the only method of retrieving the source code is via CVS.
- You can check out the latest method of retrieval at: " http://directory.fedora.redhat.com/wiki/CoolKey "
- As of writing this is the method of retrieval:

```
" CVSROOT=:pserver:anonymous@cvs.fedora.redhat.com:/cvs/dirsec ; export CVSROOT "
" cvs login "
" cvs checkout coolkey "
```

2. Uncompress all of the sources

```
tar xzfv libusb-0.1.12.tar.gz
tar xzfv pcsc-lite-1.3.1.tar.gz
tar xzfv pcsc-tools-1.4.5.tar.gz
tar xzfv ccid-1.0.1.tar.gz
```

(CoolKey does not require unpacking if retrieved from CVS, as it is downloaded into its own directory.)

3. Configure, compile, and install the various packages

(Note, the path used in this document to install these items is /usr/cac. The directions reflect this. You may install in any other location simply by replacing any instance of /usr/cac with the alternate location.)

libusb: cd libusb-0.1.12

- ./configure --prefix=/usr/cac
- make
- make install

pcsc-lite: cd pcsc-lite-1.3.1

- export PKG_CONFIG_PATH=/usr/cac/lib/pkgconfig:$PKG_CONFIG_PATH
- ./configure --prefix=/usr/cac
- make
- make install

pcsc-tools: cd pcsc-tools-1.4.5

- Edit the " Makefile " in the top level directory and change the " DESTDIR " variable to where you want the package installed.
- make
- make install

ccid: cd ccid-1.0.1

- ./configure --prefix=/usr/cac
- make
- make install

coolkey: cd coolkey

- ./configure --prefix=/usr/cac
- make
- make install

4. Post-installation configuration

If you have an ActivCard Gold USB 2.0 reader, to provide support for this reader in PC/SC, you must edit the following file: " /usr/cac/pcsc/drivers/ifd-ccid.bundle/Contents/Info.plist "

Change the section that reads:

" <key>ifdDriverOptions</key> "
" <string>0x0000</string> " **to**
" <key>ifdDriverOptions</key> "
" <string>0x0004</string> "

(It starts on line 38.)

5. Start the PC/SC daemon and make sure it starts at startup.

(This is the last step that requires elevated privileges. The rest of the steps will need to be done by each user and requires no administrative privileges.)

- You can start the daemon by typing " /usr/cac/sbin/pcscd " as the super user.
- You will probably want to add some sort of startup script to start the daemon at boot time. On Slackware this was done by simply adding " /usr/cac/sbin/pcscd " to the end of the /etc/rc.local file. This will vary by distribution.
- Make sure the reader is plugged in at this point.
- If a card is not detected in the card reader later on in this document and the reader was plugged in before the daemon was started, try unplugging the reader and plugging it back in. This is a bug in PC/SC.

6. Install the certificate authorities in your client.

(Importing the certificates is required.. If these are not installed correctly, Public Key Infrastructure will not function correctly.)

- Download the certificates from " **https://crl.chamb.disa.mil/** " and save them in a convenient place. Be sure to download them all.
- Import your certificates into your client:

Firefox:

- Select "Preferences" from the "Edit" menu.
- Click on the "Advanced" button.
- Click on the "Security" tab.
- Click "View Certificates".
- Click on the "Authorities" tab.
- Click the "Import" button.

- Browse to the location of your downloaded certificates and import them. You must import them one at a time, unfortunately.

Thunderbird:

- Select "Account Settings" from the "Edit" menu.
- Click on the word "Security" in the left pane of the window under the account you wish to associate your CAC with.
- Click the "View Certificates" button.
- Click on the "Authorities" tab.
- Click the "Import" button.
- Browse to the location of your downloaded certificates and import them. You must import them one at a time, unfortunately.

7. Install the "Security Device" into your clients.

(Make sure your Common Access Card is inserted at this point and enter your PIN any time the client prompts you for the "Master Password.")

Firefox:

- Select "Preferences" from the "Edit" menu.
- Click on the "Advanced" button.
- Click on the "Security" tab.
- Click the "Security Devices" button.
- Click the "Load" button when that window comes up.
- Under "Module Name" type " CAC Module ".
- Under "Module filename" type " /usr/cac/lib/pkcs11/libcoolkeypk11.so ".
- Click "OK".
- When it prompts you for confirmation, click "OK". At this point, a dialog box stating "A new security module has been installed." or something similar should appear.
- Dismiss this window by clicking "OK".
- Click "OK" to leave the "Device Manager" window.
- Click "Close" to dismiss your Preferences window.
- You can test your client by visiting " https://infosec.navy.mil/cgi-bin/testmypki.cgi ".

Thunderbird:

- Select "Account Settings" from the "Edit" menu.
- Click on the word "Security" in the left pane of the window under the account you wish to associate your CAC with.
- Click the "Security Devices" button.
- Click the "Load" button when that window comes up.
- Under "Module Name" type " CAC Module ".
- Under "Module filename" type " /usr/cac/lib/pkcs11/libcoolkeypk11.so ".
- Click "OK".
- When it prompts you for confirmation, click "OK". At this point, a dialog box stating "A new security module has been installed." or something similar should appear.

- Dismiss the this window by clicking "OK".
- Click "OK" to leave the "Device Manager" window.
- You should still be in the "Security" section in your preferences. Click the "Select" button in the "Digital Signing" section of the window. Select your "Email Signing" certificate.
- Click "Cancel" if a dialog box pops up asking if you would like to use the same certificate for both Encryption and signing.
- Next, click the "Select" button in the "Encryption" and select your "Email Encryption Certificate".
- Click "Cancel" if a dialog box pops up asking if you would like to use the same certificate for both Encryption and signing.
- Click "OK" to dismiss your Account Settings window.
- You can test your client by sending yourself a digitally signed email.

You should now have working PKI support in your Firefox and Thunderbird clients on Linux. The configuration information for each user is saved in their home directory under one or more of the following directories: " $HOME/.thunderbird ", " $HOME/.mozilla ", and " $HOME/.firefox ".