

Smart card OMNIKEY® 6121 Mobile USB Reader integration with Linux

Tested with Ubuntu 10.04 and Linux Mint 9 Isadora 32/64-bit

A. Overview. HID Global's OMNIKEY product brand, one of the world's leading manufacturers of innovative smart card readers, has developed a portable USB smart card reader. The OMNIKEY® 6121 is a fully functional smart card reader for SIM-sized smart cards and is especially well-suited for use with mobile devices. Applications for this reader include the download of GSM applications from the internet to the SIM card, W-LAN authentication, secure PC log-on, PKI for mobile users, digital signature, secure mobile banking and online transactions, loyalty programs, healthcare solutions and many more applications in the future.

More info at : http://www.hidglobal.com/prod_detail.php?prod_id=182

B. Installation of drivers and software

Execute each checkpoint condition until you get valid feedback from the executed command. Sensitive output is replaced with 'X' and '*'. Order is arguably important. Install in the following order:

1. Checkpoint: **pcsc_scan** (test package availability)

on fail:

1.1. **sudo apt-get install pcsc-tools**

1.2. **sudo apt-get install pcscd**

ON SUCCESS:

```
$ pcsc_scan
PC/SC device scanner
V 1.4.16 (c) 2001-2009, Ludovic Rousseau <ludovic.rousseau@free.fr>
Compiled with PC/SC lite version: 1.5.3
Scanning present readers...
Waiting for the first reader...
```

after insert of the dongle to a USB slot:

```
Waiting for the first reader...found one
Scanning present readers...
0: OmniKey CardMan 6121 00 00
```

```
Mon Jul 26 11:10:15 2010
Reader 0: OmniKey CardMan 6121 00 00
Card state: Card inserted,
ATR: 3B F2 18 00 02 C1 0A 31 FE 58 C8 08 74
```

```
ATR: 3B F2 18 00 02 C1 0A 31 FE 58 C8 08 74
+ TS = 3B --> Direct Convention
+ T0 = F2, Y(1): 1111, K: 2 (historical bytes)
TA(1) = 18 --> Fi=372, Di=12, 31 cycles/ETU
          129032 bits/s at 4 MHz, fMax for Fi = 5 MHz => 161290 bits/s
TB(1) = 00 --> VPP is not electrically connected
TC(1) = 02 --> Extra guard time: 2
TD(1) = C1 --> Y(i+1) = 1100, Protocol T = 1
-----
TC(2) = 0A --> Work waiting time: 960 x 10 x (Fi/F)
TD(2) = 31 --> Y(i+1) = 0011, Protocol T = 1
-----
TA(3) = FE --> IFSC: 254
TB(3) = 58 --> Block Waiting Integer: 5 - Character Waiting Integer: 8
+ Historical bytes: C8 08
Category indicator byte: C8 (proprietary format)
+ TCK = 74 (correct checksum)
```

```
Possibly identified card (using /usr/share/pcsc/smartcard_list.txt):
3B F2 18 00 02 C1 0A 31 FE 58 C8 08 74
```

2. Checkpoint: **opensc-tool -i** (test package availability)

on fail:

2.1. **sudo apt-get install opensc**

2.2. **sudo vi /etc/opensc/opensc.conf**

Find and the string 'reader_drivers', remove the '#' comment and change it to
reader_drivers = pcsc;

2.3 **sudo vi /etc/opensc/opensc.conf** (fine-tune optional)

Navigate to app opensc-pkcs11{...} section, remove the '#' comments and change the next values to:

plug_and_play = false;

max_virtual_slots = 4;

slots_per_card = 1;

2.4. **sudo service pcscd restart**

3. Checkpoint: **pkcs11-tool -lt** (test package availability, note the capital i)

on fail:

3.1. **sudo apt-get install pkcs11-tools**

ON SUCCESS:

```
$ pkcs11-tool -lt
```

```
Cryptoki version 2.20
```

```
Manufacturer      OpenSC (www.opensc-project.org)
```

```
Library    smart card PKCS#11 API (ver 0.0)
```

```
C_SeedRandom() and C_GenerateRandom():
```

```
  seeding (C_SeedRandom) not supported
```

```
  seems to be OK
```

```
Digests:
```

```
  all 4 digest functions seem to work
```

```
  MD5: OK
```

```
  SHA-1: OK
```

```
  RIPEMD160: OK
```

```
Signatures: not logged in, skipping signature tests
```

```
Verify: not logged in, skipping verify tests
```

```
Key unwrap: not logged in, skipping key unwrap tests
```

```
Decryption: not logged in, skipping decryption tests
```

```
Testing card detection
```

```
Please press return to continue, x to exit:
```

```
Available slots:
```

```
Slot 0    OmniKey CardMan 6121 00 00
```

```
token label: ***** (PIN)
```

```
token manuf: Siemens AG (C)
```

```
token model: PKCS#15
```

```
token flags: login required, PIN initialized, token initialized
```

```
serial num :
```

```
Slot 1    (empty)
```

```
Slot 2    (empty)
```

```
Slot 3    (empty)
```

```
Please press return to continue, x to exit: x
```

```
No errors
```

4. Checkpoint: **pkcs11-tool -lt** (test all certificates, note the small L)

failure may occur with RSA-2048 type certificate or more than 1 certificate count at the smart card

```
$ pkcs11-tool -lt
```

```
Please enter User PIN:
```

```
C_SeedRandom() and C_GenerateRandom():
```

```
  seeding (C_SeedRandom) not supported
```

```
  seems to be OK
```

```
Digests:
```

```
  all 4 digest functions seem to work
```

```
  MD5: OK
```

```
  SHA-1: OK
```

```
  RIPEMD160: OK
```

```
Signatures (currently only RSA signatures)
```

```
testing key 0 (XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX)
all 4 signature functions seem to work
testing signature mechanisms:
  RSA-X-509: OK
  RSA-PKCS: OK
  SHA1-RSA-PKCS: OK
  MD5-RSA-PKCS: OK
  RIPEMD160-RSA-PKCS: OK
testing key 1 (2048 bits, label=XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX) with 1 signature mechanism
[opencsc-pkcs11] reader-pcsc.c:239:pcsc_transmit: unable to transmit
[opencsc-pkcs11] apdu.c:394:do_single_transmit: unable to transmit APDU
[opencsc-pkcs11] iso7816.c:832:iso7816_decipher: APDU transmit failed: Transmit failed
[opencsc-pkcs11] sec.c:39:sc_decipher: returning with: Transmit failed
[opencsc-pkcs11] pkcs15-sec.c:125:sc_pkcs15_decipher: sc_decipher() failed: Transmit failed
error: PKCS11 function C_Sign failed: rv = CKR_GENERAL_ERROR (0x5)
```

Aborting.

MD5-RSA-PKCS:

4.1 on fail:

```
sudo apt-get purge libccid
sudo apt-get install pcsc-omnikey
```

ON SUCCESS:

```
$ pkcs11-tool -lt
Please enter User PIN: .....
C_SeedRandom() and C_GenerateRandom():
  seeding (C_SeedRandom) not supported
  seems to be OK
Digests:
  all 4 digest functions seem to work
  MD5: OK
  SHA-1: OK
  RIPEMD160: OK
Signatures (currently only RSA signatures)
testing key 0 (XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX)
all 4 signature functions seem to work
testing signature mechanisms:
  RSA-X-509: OK
  RSA-PKCS: OK
  SHA1-RSA-PKCS: OK
  MD5-RSA-PKCS: OK
  RIPEMD160-RSA-PKCS: OK
testing key 1 (2048 bits, label=XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX) with 1 signature mechanism
  MD5-RSA-PKCS: OK
Verify (currently only for RSA):
testing key 0 (XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX)
  RSA-X-509: OK
  RSA-PKCS: OK
  SHA1-RSA-PKCS: OK
  MD5-RSA-PKCS: OK
  RIPEMD160-RSA-PKCS: OK
testing key 1 (XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX) with 1 mechanism
  RSA-X-509: OK
Key unwrap (RSA)
testing key 0 (XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX)
  DES-CBC: OK
  DES-EDE3-CBC: OK
  BF-CBC: OK
  CAST5-CFB: OK
testing key 1 (XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX)
  DES-CBC: OK
  DES-EDE3-CBC: OK
  BF-CBC: OK
  CAST5-CFB: OK
Decryption (RSA)
testing key 0 (XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX)
  RSA-X-509: OK
  RSA-PKCS: OK
testing key 1 (XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX)
  RSA-X-509: OK
  RSA-PKCS: OK
Testing card detection
Please press return to continue, x to exit: x
```

Testing card detection using C_WaitForSlotEvent
Please press return to continue, x to exit: x
No errors

C. Integration with Firefox 3.X browser (for online banking)

1. Import the core bank certificate chain to:

Preferences --> Advanced --> View Certificates --> Your Certificates --> Import

2. Add a new module at 'Security Devices'

Preferences --> Advanced --> Security Devices --> Load

Name the module as liked, navigate and add the module `/usr/lib/onepin-opensc-pkcs11.so`.