Apica ProxySniffer

Using PKCS#11 Security Devices for Recording Web Surfing Sessions and for Executing Web Load Tests

English Edition

1. Abstract

Wikipedia: In cryptography, PKCS#11 is one of the family of standards called Public-Key Cryptography Standards (PKCS), published by RSA Laboratories. It defines a platform-independent API to cryptographic tokens, such as Hardware Security Modules (HSM) and smart cards.

Proxy Sniffer version 4.4 supports executing web load tests and web stress tests using PKCS#11 Security Devices. PKCS#11 Security Devices are hardware cryptographic devices with an integrated cryptographic processor, and are capable of storing embedded X509 certificates which can be used for HTTPS/SSL client authentication and digitally signing documents. Some "special" PKCS#11 Security Devices may also be implemented as a pure software device.

The use of PKCS#11 Security Devices by Proxy Sniffer is supported for the following operating systems:

- Windows (32 Bit and 64 Bit versions).
- Linux (32 Bit and 64 Bit versions)
- Solaris (SPARC processors only)

The following web site contains a listing of all supported hardware devices:
http://jce.iaik.tugraz.at/sic/Products/Core-Crypto-Toolkits/PKCS-11-Wrapper/tested-products

New Java Paths, starting from Proxy Sniffer version 4.4:

- In the case where you do not use the Proxy Sniffer Console (that is, when you start Proxy Sniffer manually from a terminal command line), ensure that the Java CLASSPATH contains the files **prxsniff.jar**, **iaik_jce_full.jar**, **iaikPkcs11Provider.jar**, and **the Proxy Sniffer installation directory**, and **the default directory** (.).
- The java option **-Xbootclasspath/p:** **must NOT** be used any longer (Proxy Sniffer version 4.4 or later versions will not work if this option is set).

Example:

```
set CLASSPATH=
.;C:\Users\miller\ProxySniffer\prxsniff.jar;C:\Users\miller\ProxySniffer;C:\Users\miller\ProxySniffer\iaik_jce_full.jar;C:\Users\miller\ProxySniffer\iaikPkcs11Provider.jar
```

License: The usage of the cryptographic libraries from IAIK, as well as the usage of the Proxy Sniffer web load test tool (which already includes sub-license from IAIK - usable only together with Proxy Sniffer), requires a commercial license.

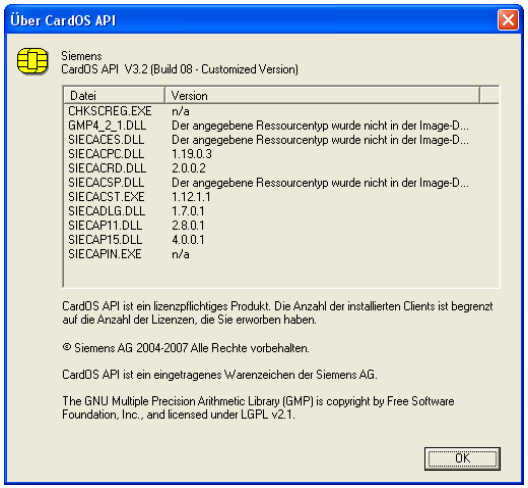
2. Configuring and Using the PKCS#11 Device

1) Plug the PKCS#11 Security Device into your computer.

2) Copy the operating system-specific 32- or 64-Bit file named **pkcs11wrapper.dll** or **libpkcs11wrapper.so** into **the Proxy Sniffer installation directory** (where prxsniff.jar is located). You will find the corresponding file inside the **PKCS#11 subfolder**, which is located in the Proxy Sniffer installation directory.

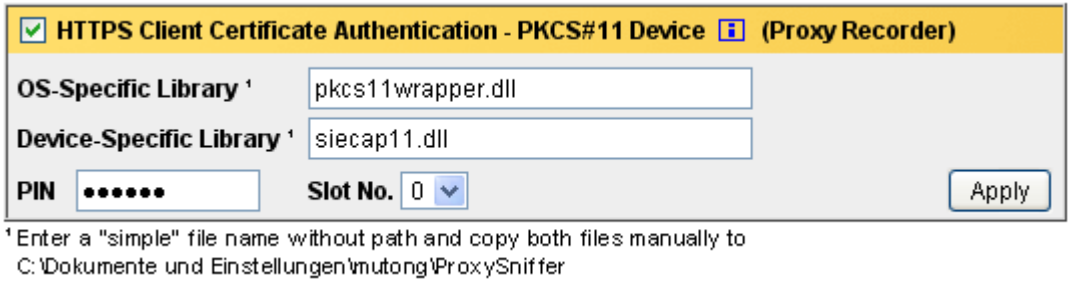
3) Copy the **PKCS#11 device-specific file** (*.dll or *.so) into **the Proxy Sniffer installation directory**; for example, iidp11.dll or siecap11.dll). Appendix A at the end of this document contains a list of some well-known PKCS#11 device-specific files.

   Example of how to find out which library is used by the PKCS#11 device:

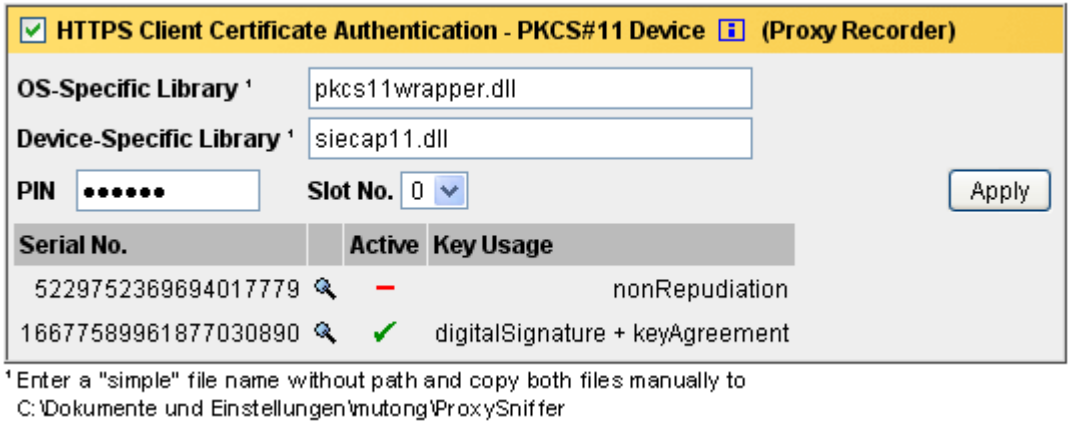


4) Start, or restart, the Proxy Sniffer Console.

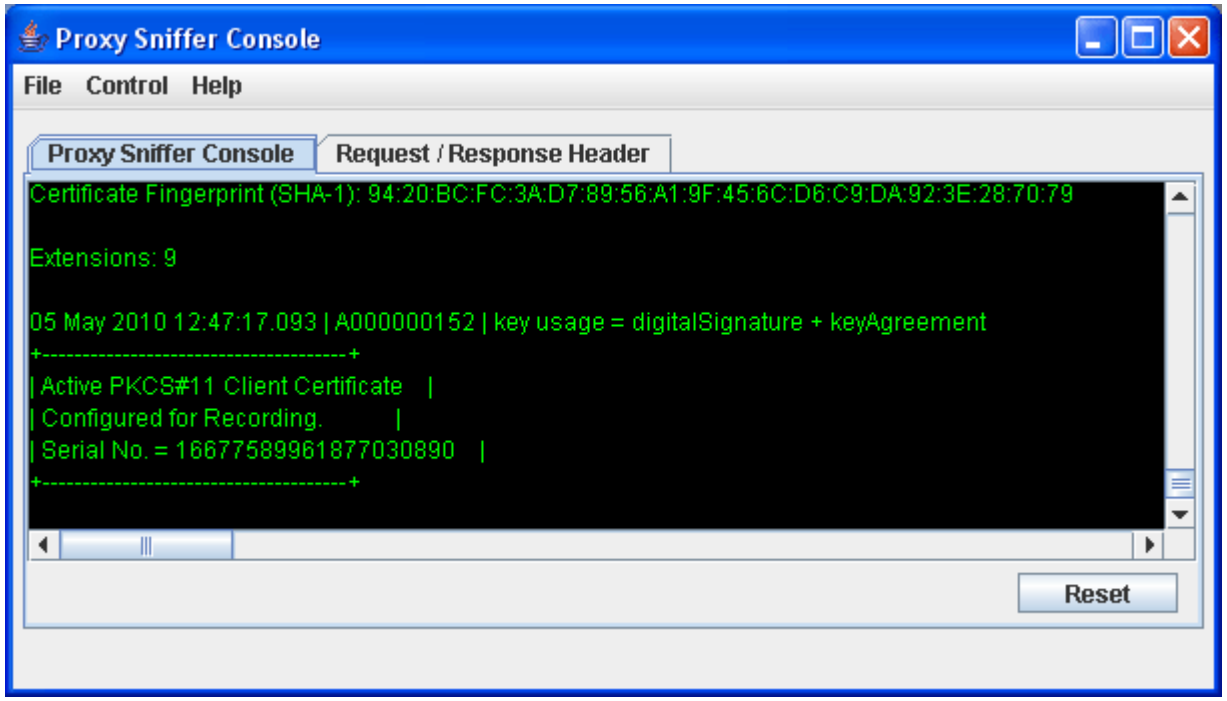
5) Configure the PKCS#11 Security Device inside the Proxy Sniffer **Personal Settings** menu:



6) Select the corresponding X509 Client Certificate by clicking inside the red bar on the certificate, which then turns this bar to a green check mark:

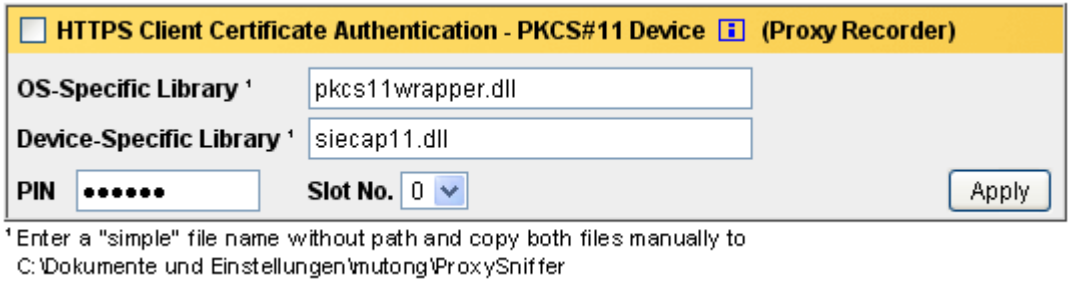


7) You will see the following output in the Proxy Sniffer Console:

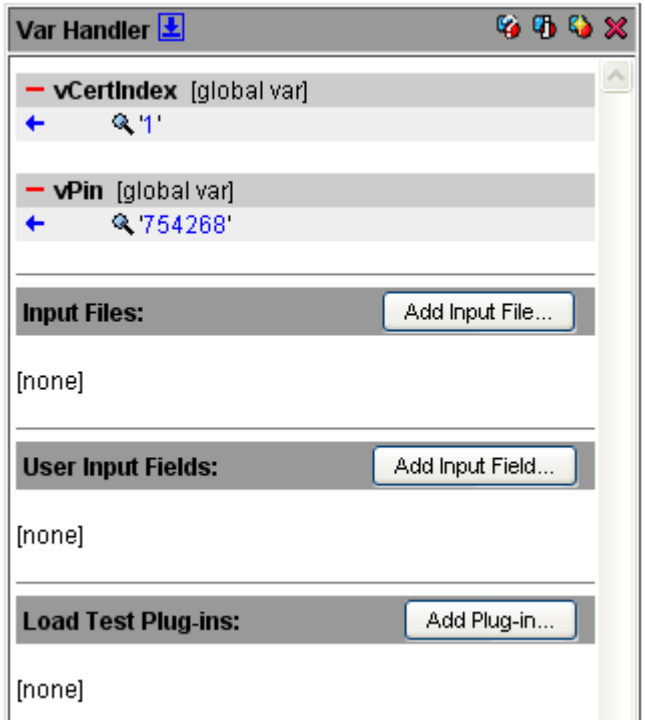


8) Record your Web Surfing Session as usual.

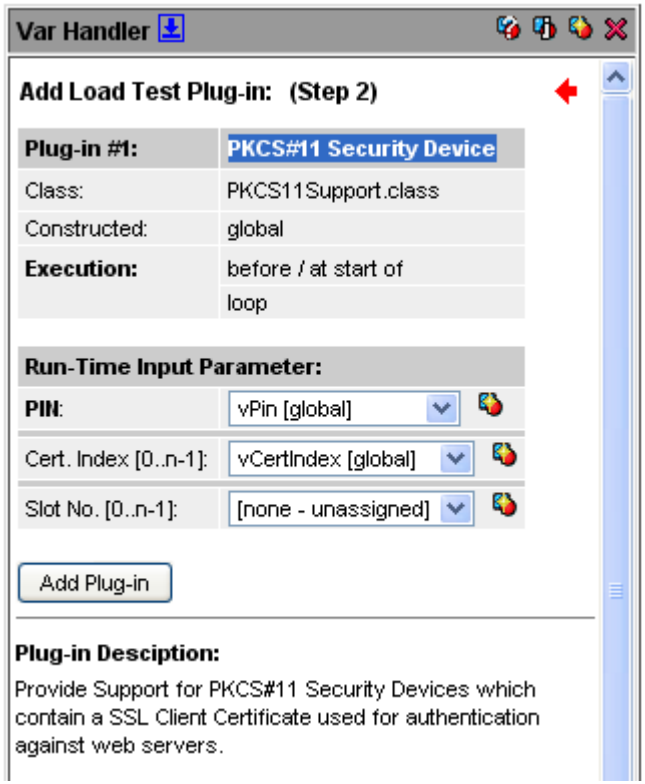
9) After recording the Web Surfing Session, de-activate the configuration for the PKCS#11 Security Device by disabling the checkbox inside the yellow bar and then clicking on the Apply button:



10) Create a global variable for the **PIN** and – if several certificates are stored inside the PKCS#11 Device – also for **the index of the certificate** (0..n-1):

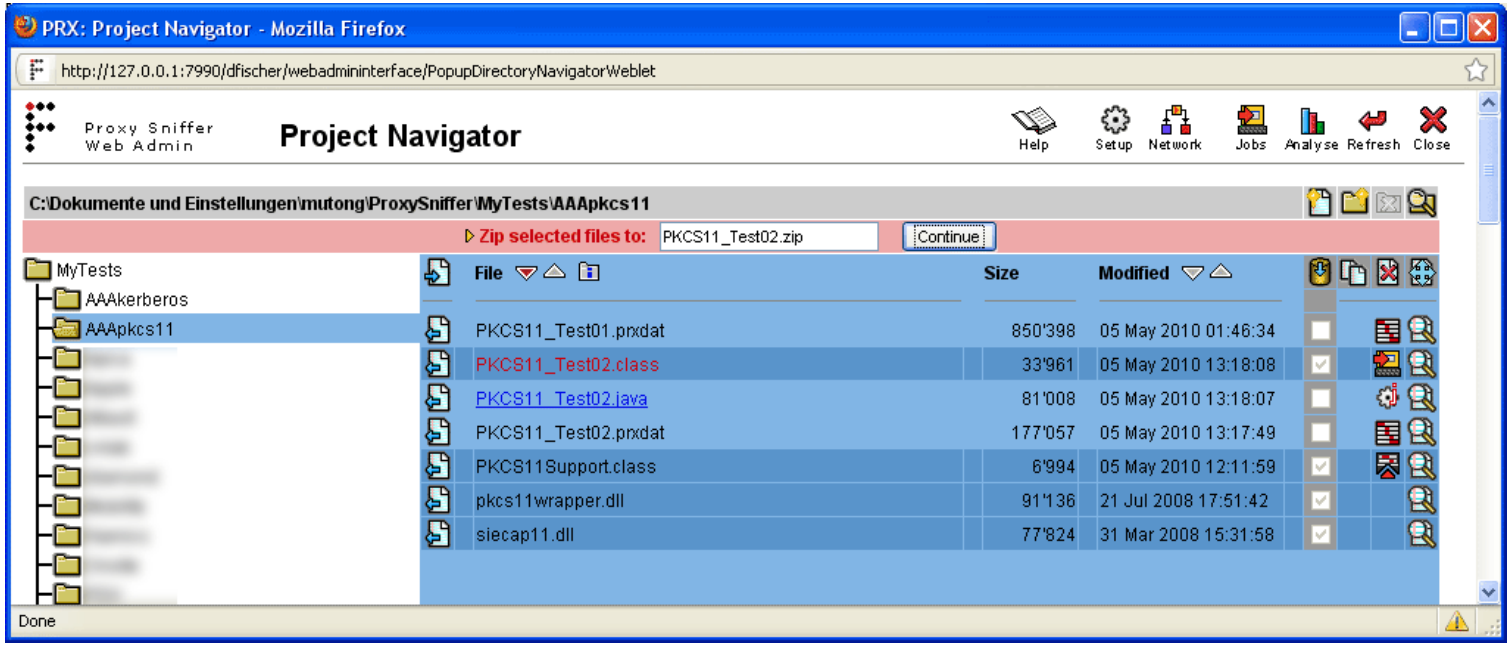


11) Add the plug-in named "**PKCS#11 Security Device**" to your recorded session:



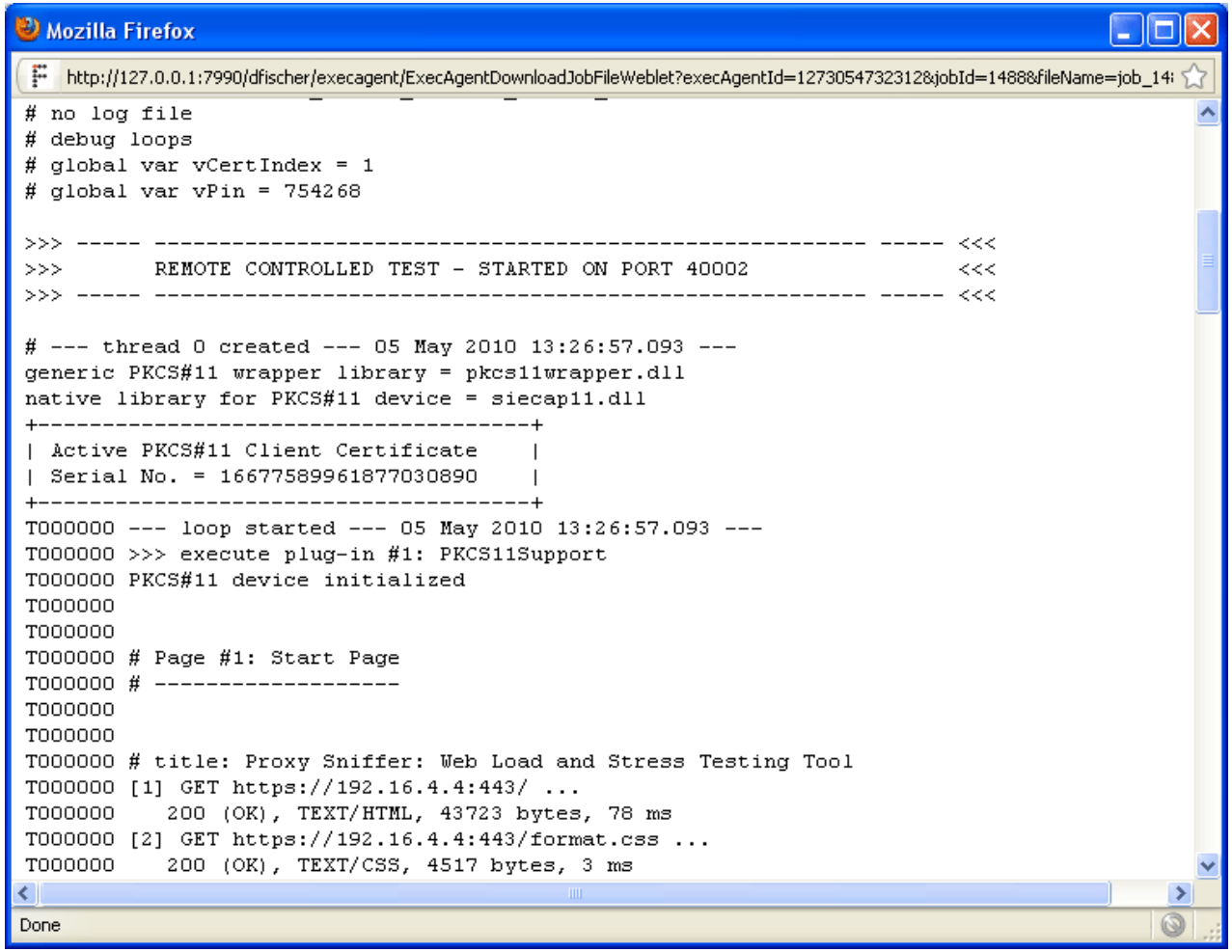
12) Generate the load test program as usual.

13) Copy the **operating system-specific 32- or 64-Bit file,** and the **device-specific file,** into the directory where your load test program is stored, and ZIP them together with the plug-in and the load test program:



Note: you must use the operating system-specific 32- or 64-Bit file and the device-specific file **of the operating system on which the Exec Agent (Load Generator) is running**!

14) Execute the load test program by clicking on the ZIP archive. Select for a first test a single user and one loop. The log output of the load test program will show a corresponding message about the PKCS#11 device:



Appendix A: List of some well-known PKCS#11 Device-Specific Files

(Source: Stiftung Secure Information and Communication Technologies SIC)

- aetpkss1.dll (for G&D StarCos and Rainbow iKey 3000)
- cs2_pkcs11.dll (for Utimaco CryptoServer LAN)
- CccSigIT.dll (for IBM MFC)
- pk2priv.dll (for GemSAFE, old version)
- gclib.dll (for GemSAFE, new version)
- dspkcs.dll (for Dallas iButton)
- slbck.dll (for Schlumberger Cryptoflex and Cyberflex Access)
- SetTokI.dll (for SeTec)
- acpkcs.dll (for ActivCard)
- psepkcs11.dll (for A-Sign Premium)
- id2cbox.dll (for ID2 PKCS#11)
- smartp11.dll (for SmartTrust PKCS#11)
- pkcs201n.dll (for Utimaco Cryptoki for SafeGuard)
- dkck201.dll (for DataKey and Rainbow iKey 2000 series)
- cryptoki.dll (for Eracom CSA)
- AuCryptoki2-0.dll (for Oberthur AuthentIC)
- eTpkcs11.dll (for Aladdin eToken, and some Siemens Card OS cards)
- cknfast.dll (for nCipher nFast or nShield)
- cryst201.dll (for Chrysalis LUNA)
- cryptoki.dll (for IBM 4758)
- softokn3.dll (for the Mozilla or Netscape crypto module)
- iveacryptoki.dll (for Rainbow CryptoSwift HSM)
- sadaptor.dll (for Eutron CryptoIdentity or Algorithmic Research MiniKey)
- pkcs11.dll (for TeleSec)
- siecap11.dll (for Siemens HiPath SIcurity Card API)
- asepkcs.dll (for Athena Smartcard System ASE Card)
- /opt/SUNWconn/cryptov2/lib/libvpkcs11.so (for SUN Crypto Accelerator 4000, 32-bit libraries)
- /opt/SUNWconn/cryptov2/lib/sparcv9/libvpkcs11.so (for SUN Crypto Accelerator 4000, 64-bit libraries)

- /opt/SUNWconn/crypto/lib/libpkcs11.so (for SUN Crypto Accelerator 1000, 32-bit libraries)
- /opt/SUNWconn/crypto/lib/sparcv9/libpkcs11.so (for SUN Crypto Accelerator 1000, 64-bit libraries)