



Whitepaper : Using Unsniff Network Analyzer to analyze SSL / TLS

A number of applications today use SSL and TLS as a security layer. Unsniff allows authorized users to analyze these applications by decrypting the SSL/TLS streams in real time. This is done without interrupting the SSL streams in any way. Unsniff can also strip out the SSL/TLS layer completely and analyze the application protocols as if the security layer never existed. For example: If you are working with a secure web server, you can analyze the HTTPS protocol, including the ability to reconstruct complete web pages.



[References : RFC2246 \(TLS 1.0\), RFC2459 \(X.509v3\), PKCS Standards \(RSA Website\)](#)

[Feature overview](#)

[Working with PDUs and Streams](#)

[Decrypting SSL/TLS](#)

[Analyzing upper layer protocols](#)

[Howto : Analyzing a secure Microsoft IIS web server](#)

[Howto : Analyzing a secure Apache web server](#)

[Howto : Analyzing protocols tunneled via stunnel](#)

[FAQ](#)

Feature Overview

SSL records	Unsniff shows SSL/TLS records as separate entities in the PDU sheet irrespective of how they were carried at the link layer.
Stream analysis	Monitor entire SSL / TLS sessions in real time via the Streams sheet.
Decryption	Provided you have the servers private key material you can decrypt SSL / TLS sessions in real time. Most of the popular ciphers are supported.
Advanced Decryption	Unsniff supports SSL / TLS features such as session reuse and cipher renegotiation.
Upper layer decryption	SSL/TLS only acts as a transport layer for higher layer protocols. Ultimately we are interested in the analysis of higher layer protocols such as HTTP, LDAP etc. Unsniff does not just stop at showing you the decrypted text, it actually strips off the security layer and performs full analysis of the upper layer protocol. This allows you to do things like view complete web pages transferred via HTTPS.
Scripting	Like all other protocols, SSL/TLS is completely scriptable. You can write tools in Ruby or VBScript to scan through a capture file looking for weak ciphers or untrusted certificates.

Working with PDUs and Streams

PDU Analysis

SSL/TLS is a record oriented protocol that runs on top of TCP. These SSL/TLS records (or PDUs) can be upto 16K bytes in length. They also do not respect link layer packet boundaries. You can have multiple SSL/TLS records per ethernet packet or a single record spread over multiple packets. Unlike older protocol analyzers, Unsniff analyzes PDUs (protocol data units) as first class entities. This means that you do not have to dig through ethernet (or other link layer) frames while trying to locate SSL / TLS records.

To view entire SSL/TLS records switch to the PDU sheet

S...	Stream	Receiver	Type	Size	Seek	Description
45	64.124.178.51	59.92.38.48	TLS	3208	1134	Handshake: Application Data
46	59.92.38.48	64.124.178.51	TLS	102	0	Handshake: Client Hello TLS 1.0, resume session E889D6A2...
47	59.92.38.48	64.124.178.51	TLS	102	0	Handshake: Client Hello TLS 1.0, resume session E889D6A2...
46	64.124.178.51	59.92.38.48	TLS	79	0	Handshake: Server Hello TLS 1.0, select cipher TLS_RSA_WITH...
46	64.124.178.51	59.92.38.48	TLS	6	79	Change Cipher Spec
46	64.124.178.51	59.92.38.48	TLS	37	85	Handshake: Encrypted Message
46	59.92.38.48	64.124.178.51	TLS	6	102	Change Cipher Spec
46	59.92.38.48	64.124.178.51	TLS	37	108	Handshake: Encrypted Message
46	59.92.38.48	64.124.178.51	TLS	374	145	Handshake: Application Data
47	64.124.178.51	59.92.38.48	TLS	79	0	Handshake: Server Hello TLS 1.0, select cipher TLS_RSA_WITH...
47	64.124.178.51	59.92.38.48	TLS	6	79	Change Cipher Spec
47	64.124.178.51	59.92.38.48	TLS	37	85	Handshake: Encrypted Message
47	59.92.38.48	64.124.178.51	TLS	6	102	Change Cipher Spec

Stream Analysis

Unsniff also allows you to monitor entire SSL / TLS streams as first class entities. You can see entire SSL/TLS sessions being established and completed in real time in the Streams sheet. The main advantage of stream monitoring is you can instantly zone in on the exact stream you want. You can even copy-paste entire streams into another capture file or iterate through streams via the scripting interface.

To view SSL/TLS streams switch to the "Sessions" sheet

Num	State	Start	Source IP	S.P...	Dest IP	D....
1	TCP Packet Flags = (ACK=H	05-24-2006 15:4...	192.168.0.100	3584	192.168.0.101	443
2	[Synth/Decrypted] Handshake: Finished	05-24-2006 15:4...	192.168.0.100	3584	192.168.0.101	443
3	[Synth/Decrypted] TCP Layer: ACK, RST	05-24-2006 15:4...	192.168.0.100	3584	192.168.0.100	3584

Decrypting SSLv3.0 / TLS 1.0

How it works ?



Only legitimate users who have access to the servers private key such as system administrators can use this feature. Unsniff has no mechanisms to decrypt SSL / TLS sessions without legitimate server private key material.

The servers digital certificate plays a pivotal role in the authentication and encryption of data. Upon initiation of a SSL session, the SSL client and server quickly agree on a shared secret (the master secret) using public-key cryptography. When provided with the servers private key, Unsniff can decrypt the data transferred.

When the right keying material is available

- **For every encrypted record (PDU)**, Unsniff shows a corresponding clear text record. This helps you analyze encrypted SSL records such as alerts.
- **For every encrypted SSL stream**, Unsniff shows a corresponding clear text stream.
- **For every SSL stream**, Unsniff shows a corresponding application layer stream with the SSL / TLS layer completely stripped off. This “*Analyze Upper Layers*” option must be enabled for this feature.

Supported Cipher Suites Unsniff supports the following cipher suites. If you want support for a cipher not listed below, please post a request in our online forum.

- RC4_128_WITH_MD5
- RC4_128_EXPORT40_WITH_MD5
- RSA_WITH_AES_256_CBC_SHA
- RSA_EXPORT1024_WITH_RC4_56_SHA
- RSA_EXPORT1024_WITH_DES_CBC_SHA
- RSA_WITH_RC4_128_MD5
- RSA_WITH_RC4_128_SHA
- RSA-WITH-3DES-EDE-MD5

Note : *Ephemeral Diffie Hellman is not supported nor are export ciphers less than 1024 bits in length.*

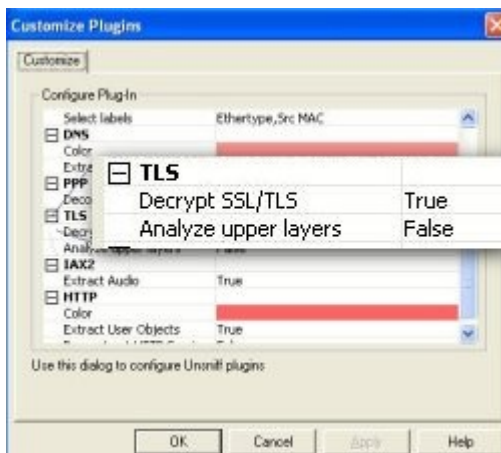
Step by step instructions

This section contains detailed instructions for setting up Unsniff to decrypt SSL / TLS.

- Enable the SSL/TLS decryption feature
- Obtain the server private key
- Enter the key information into Unsniff
- Start capturing and analyzing SSL / TLS

Enable the SSL/TLS decryption feature

You must first ensure that you have enabled SSL/TLS decryption. This option is enabled by default. Click on “Plugins->Customize” or the Customize Plugins toolbar icon. Ensure that the “Decrypt SSL/TLS” option is set to “True”



Obtaining the server private key

If you are developing or testing protocols you can request the private key file from the system administrator. However, you must immediately change the key after you are done with capturing network data. Once you have the private key file, you must convert it to unencrypted PKCS #8 format. This is the format understood by Unsniff.

Private Key Formats

There are three main key formats :

- **OpenSSL Traditional** : (this is a format that has been in use in OpenSSL and its predecessor SSLeay)

- **PKCS#8** : This is a standard format for storing private keys (Unsniff requires keys to be in this format)
- **PKCS#12** : This is a combined file format for keys and certificates (used by web browsers)

(Public Key Cryptography Standards. These standards are issued by RSA and can be found on the web at <http://www.rsasecurity.com/rsalabs/pkcs>)

Using OpenSSL to convert between private key formats

To convert the private key into PKCS#8 format you need to use the OpenSSL library. This library includes the openssl command line tool which will be used to convert between various format.

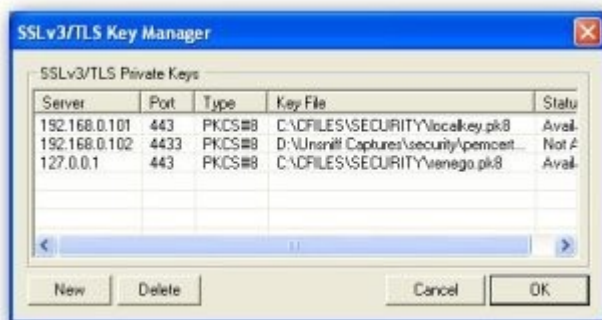
- You can download OpenSSL for free from Shining Light Productions at <http://www.slproweb.com/products/Win32OpenSSL.html>
- If you have access to a Linux machine, many distributions such as Fedora have OpenSSL preinstalled.

Enter the private key information for the server into Unsniff

Collect information about the IP address and TCP port on which your SSL / TLS enabled server is running. Now you are ready to enter this information into Unsniff.

Enter the server information into Unsniff

Select *Tools->TLS->Manage Private Keys* from the main Unsniff menu. This opens the “SSL/TLS Key Manager” Dialog.






Select *New* to open the “Enter New Key” dialog. Specify the IP Address, Port and the PKCS#8 private key file in this dialog and click OK.

Start capturing and analyzing SSL / TLS

Now you are ready to start analyzing SSL / TLS. Just click on the "Start Capture" button or import a capture file in tcpdump format. Unsniff will automatically decrypt and present you with clear text protocol data in real time.

Flags mark decrypted packets and PDUs

Decrypted packets and PDUs are flagged with special icons in the packet sheet. *Observe the "key" icons on the left side of the packets sheet.*

	27	09-29-2005 18:19:00-7...	240	TLS	192.168.0.101	192.168.0.102	Outgoing	Handshake: Client Key Exchange
	28	09-29-2005 18:19:00-7...	119	TLS	192.168.0.102	192.168.0.101	Sniffed	Handshake: Server Hello TLS 1.0, se...
	29	09-29-2005 18:19:00-7...	1028	TLS	192.168.0.102	192.168.0.101	Sniffed	Handshake: Server certificate
	30	09-29-2005 18:19:00-7...	49	TLS	192.168.0.102	192.168.0.101	Sniffed	Handshake: Server Hello Done
	31	09-29-2005 18:19:00-7...	60	TLS	192.168.0.102	192.168.0.101	Incoming	TCP Packet Flags = (ACK=1)

Analyzing upper layer protocols

An exciting new feature of Unsniff Network Analyzer is the ability to analyze upper layer protocols of SSL / TLS. This allows you to go beyond the clear text of the higher layer protocols and actually continue the analysis as if the security layer never existed. For example, when analyzing SSL/TLS web sessions, Unsniff can reconstruct the entire HTTPS session including the web pages as seen by the browser.



Enable the application analysis feature

You must first ensure that you have enabled the "SSL/TLS Application Analysis" feature. This feature is disabled by default due to the overheads involved with this feature.

- Click on *Plugins->Customize* or the Customize Plugins toolbar icon.
- Scroll down and locate the TLS group.
- Ensure that the "Application Analysis" option is set to "True"

Specify the SSL / TLS ports used by the application

Unsniff can analyze any application that uses SSL/TLS. If you want to perform application analysis, you must tell Unsniff what port number corresponds to your secure application. You have to use the Access Point Manager for that purpose. For example : HTTPS (secure HTTP based on SSL/TLS) runs on TCP port 443. You can specify addition ports or new protocols. If the upper layer protocol is not supported by Unsniff, you need not have an entry.

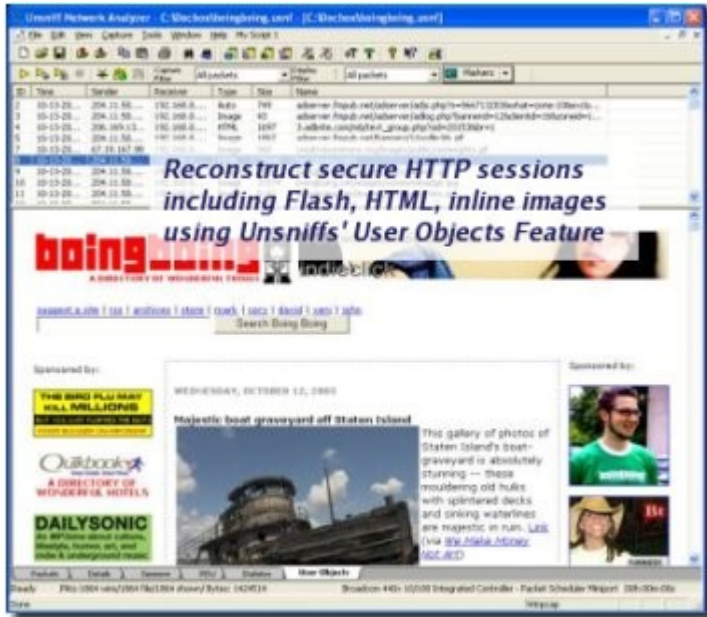
To customize SSL/TLS ports:

- Open the Access Point Manager via *Manage-> Access Points*
- Scroll down and locate the SSL/TLS group
- Click on the SSL/TLS entry
- Click on the New Access Point button at the top of the window
- Now you can create a new SSL/TLS to application layer port mapping

View application streams and user objects

Once everything is in place, Unsniff will automatically generate a new stream in the Streams Sheet that represents a real-time stripped version of the encrypted stream. You can expand the stream and view packets, or view the user objects in that stream and so forth.

Example: Capture a HTTPS session and view the reconstructed web page. Read more about Unsniff powerful HTTP analysis capabilities on our website <http://www.unleashnetworks.com>



Howto : Analyzing Microsoft IIS web server

The IIS server allows you to export the private and the server certificate in a PFX format. This tutorial explains how to export the key and convert it into unencrypted PKCS #8 format as expected by Unsniff.

Export the servers private key to PFX format

Microsoft Windows allows you to export your IIS private key and digital certificate in a format called PFX. It is roughly equivalent to the PKCS#12 format used by apache.

To export your servers private key follow the step-by-step instruction provided by Microsoft in this Technet article. Remember the password you used to export the private key. You will need it later !

[How to back up a server certificate in Internet Information Services 5.0](#)

Convert the PFX format private key to PKCS #8

Once you have the key material in PFX format, you must convert it into PKCS#8 format required by Unsniff.

First convert PFX to PEM

```
openssl pkcs12 -in MyCert.PFX -nocerts -nodes -out MyCert.PEM
```

* You will have to enter the password used to protect the server private key.

Next convert PEM to PKCS#8

```
openssl pkcs8 -in MyCert.PEM -topk8 -nocrypt -out MyCert.PK8
```

The file MyCert.PK8 is the key file you must use with Unsniff.

Howto : Analyzing a secure Apache web server

The apache web server stores its private key in a PEM format. You need to convert it to PKCS#8 format.

Locate the apache SSL certificate key (private key) file

- Open httpd.conf , this is usually located in /etc/httpd/conf
- The SSLCertificateKeyFile line tells you the name and location of the private key file
- By convention the private key file is placed in /etc/httpd/conf/ssl.key (You can then ask your administrator for help)

Convert the apache PEM format to PKCS#8

You need the secure webserver password to proceed. This is the password used to protect the server private key.

```
openssl pkcs8 -in myapachekey.pem -topk8 -out outkey.pk8 -nocrypt
```

Howto : Analyzing protocols tunneled via STUNNEL

[STUNNEL](#) is an excellent piece of software that allows you to secure arbitrary TCP connections inside a secure SSL tunnel. It is widely used to provide security to client-server systems that do not have it built in. Unsniff can look into these encrypted tunnels and provide complete visibility to the protocols that are carried inside them.

Locate the server private key

Stunnel can be started with the private key specified on the command line or reading the private key location from the stunnel.conf configuration file. The private key is in PEM format.

Convert the PEM format to PKCS#8

Use the following command to convert the private key to PKCS#8 format required by Unsniff.

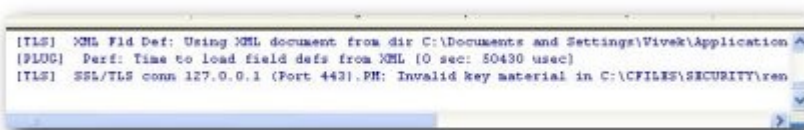
```
openssl pkcs8 -in stunnel.pem -topk8 -out outkey.pk8 -nocrypt
```

* You will be asked for the password if the stunnel.pem key is password protected

FAQ

How can I debug errors encountered during SSL/TLS analysis ?

Any error encountered during SSL/TLS analysis such as invalid key material or unsupported cipher is sent to the Unsniff Log Window. You can view the log window, via the "*View->Log Window*" menu. By default the log window only shows Major and Critical error messages. You can adjust the setting via "*Tools->Configure->Miscellaneous*", then select from the logging options.



```
[TLS] XML Fld Def: Using XML document from dir C:\Documents and Settings\Wivek\Application
[LOG] Perf: Time to load field defs from XML (0 sec: 50430 usec)
[TLS] SSL/TLS conn 127.0.0.1 (Port 443).PH: Invalid key material in C:\FILES\SECURITY\ren
```