Fraunhofer

FKIE

FRAUNHOFER INSTITUTE FOR COMMUNICATION, INFORMATION PROCESSING AND ERGONOMICS FKIE

friTap - DECRYPTING TLS TRAFFIC ON THE FLY

Daniel Baier*, Francois Egner, Max J. Ufer

OVERVIEW

friTap enables forensic researchers to intercept the generation of encryption keys used by TLS for the purpose of decrypting the entire traffic an application sends while having full access to the device of interest.

The main features of friTap are:

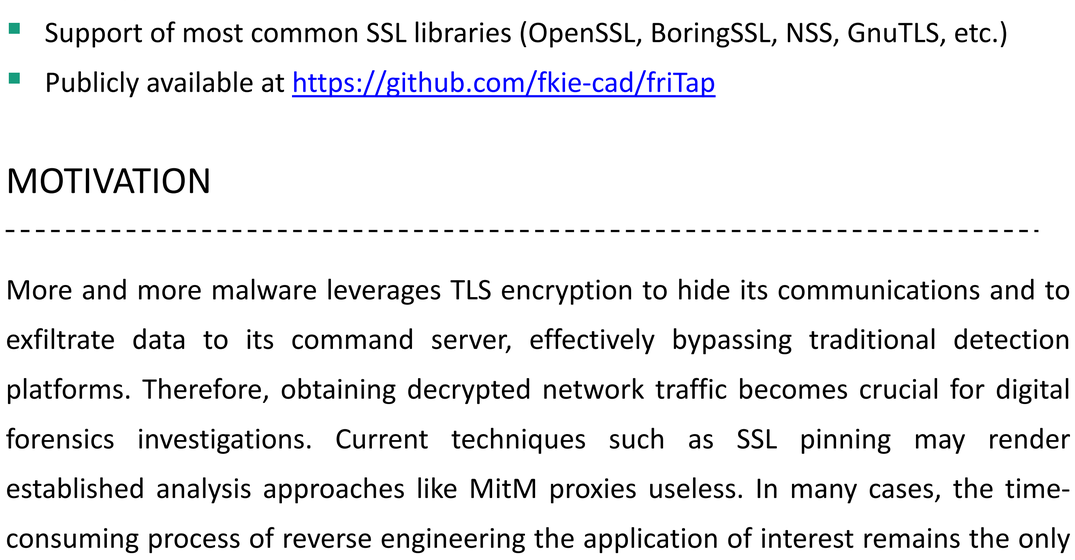
- Decryption of TLS payload as PCAP in real time
- TLS key extraction in real time
- Support of most common SSL libraries (OpenSSL, BoringSSL, NSS, GnuTLS, etc.)
- Publicly available at https://github.com/fkie-cad/friTap

MOTIVATION

More and more malware leverages TLS encryption to hide its communications and to exfiltrate data to its command server, effectively bypassing traditional detection platforms. Therefore, obtaining decrypted network traffic becomes crucial for digital forensics investigations. Current techniques such as SSL pinning may render established analysis approaches like MitM proxies useless. In many cases, the time-consuming process of reverse engineering the application of interest remains the only option to obtain the keys for decrypting the network traffic.

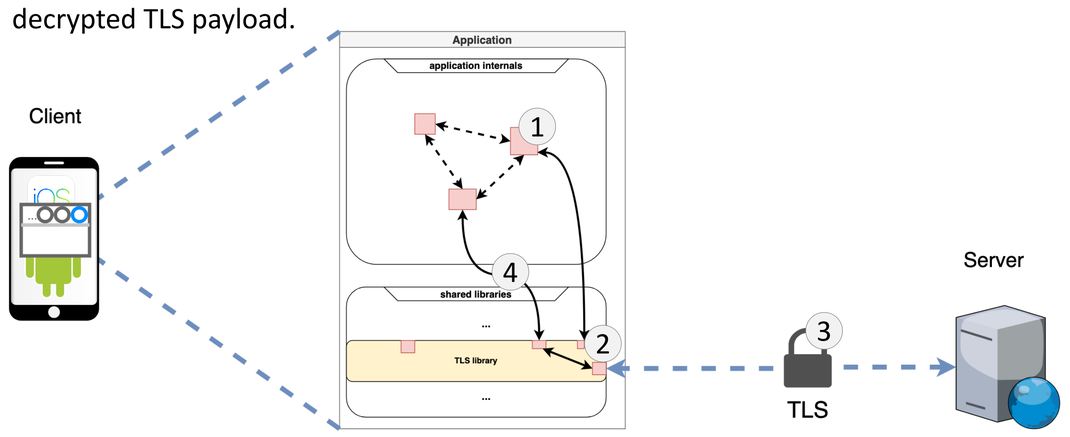
CONCEPT

friTap is a framework to solve these issues by intercepting the generation of encryption keys used by TLS for the purpose of decrypting the traffic an application sends.



Whenever an application decides to create a TLS connection (1) it usually utilizes its appropriate TLS library. This TLS library then creates the TLS socket (TLS handshake 2). When the TLS handshake is finished the TLS stream is established (3).

At this point the application uses the TLS write functions from the used TLS library to write its plaintext to the TLS stream where it gets encapsulated. In addition, the application utilizes the TLS read function from the used TLS library to process the decrypted TLS payload.



friTap identifies the TLS library used and creates the appropriate hooks (4) so that all plaintext is saved into a PCAP. Likewise, the plaintext can be output directly on the command line. Besides the possibility of saving the plaintext of TLS payload into a PCAP, friTap also enables the extraction of the TLS encryption keys.

WORKING WITH friTap

friTap provides two operation modes. One is to get the plaintext from the TLS payload as PCAP and the other is to get the used TLS keys. In order to get the decrypted TLS payload we need the `-p` parameter:

```
FKIE ~/DEF/research/friTap ./friTap.py -m -p decrypted_TLS.pcap <target_app>
…
[*] BoringSSL.so found & will be hooked on iOS!
[*] iOS dynamic loader hooked.
[*] Logging pcap to decrypted_TLS.pcap
```

The `-m` parameter indicates that we are analyzing a mobile application in the above example. Here, the implementations of the SSL libraries often differ from those of conventional desktop systems. For extracting the TLS keys from a target application we need the `-k` parameter:

```
FKIE ~/DEF/research/friTap ./friTap.py -m -k TLS_keys.log <target_app>
…
[*] NSS.so found & will be hooked on Android!
[*] Android dynamic loader hooked.
[*] Logging keylog file to TLS_keys.log
```

As a result friTap writes all TLS keys to the `TLS_keys.log` file using the NSS Key Log Format.

FUTURE PLANS

- Support for other SSL libraries
- API to support unknown SSL libraries
- Capability to alter the decrypted payload
- Support for statically linked libraries

Daniel Baier I +(49) 228 50212-427 I daniel.baier@fkie.fraunhofer.de

Max J. Ufer I +(49) 228 50212-526 I max.jens.ufer@fkie.fraunhofer.de

Francois Egner I +(49) 228 50212-621 I francois.egner@fkie.fraunhofer.de