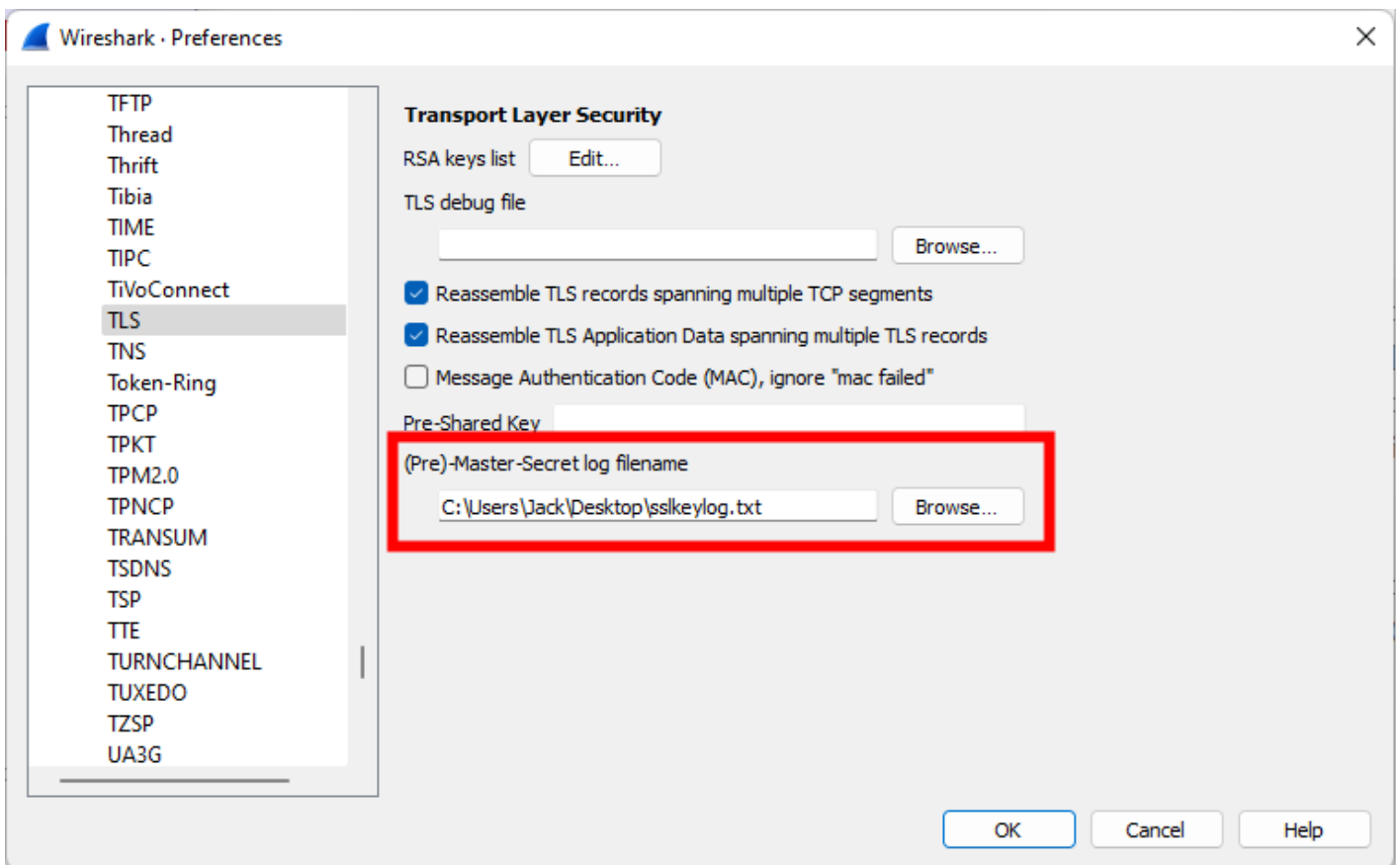


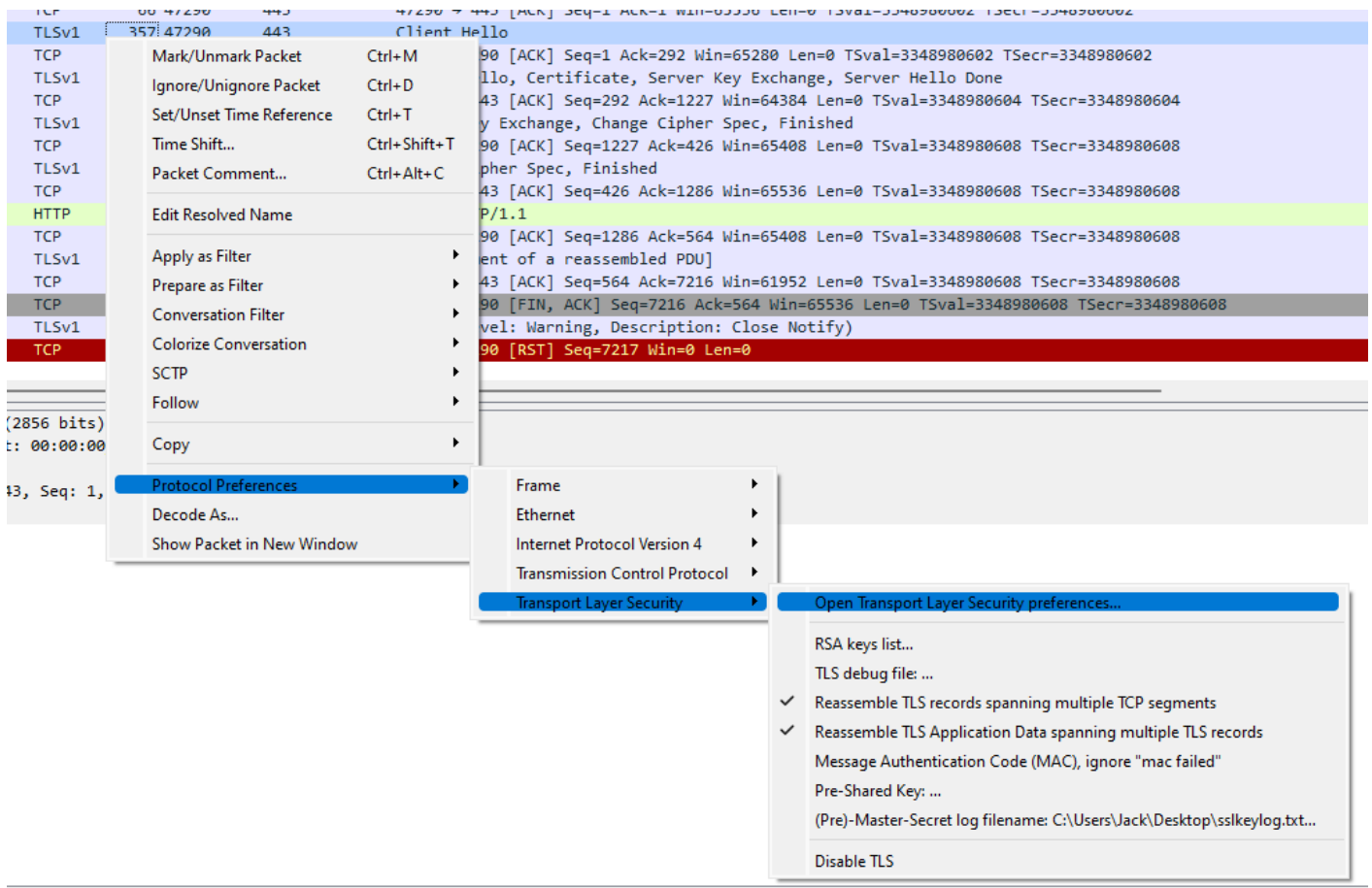
Frida SSL intercept

Overview

The output from these Frida scripts should be entered into Wireshark's '(Pre)-Master-Secret log filename' field under 'TLS' in 'Protocol Preferences'. The files should contain one set of keys per line. The window is shown below.



To quickly navigate to this window, right click on a TLS frame and follow the screenshot below.



SSLv3 and TLSv1.0

In order to decrypt SSLv3 and TLSv1.0, we need the 'client random' and 'master key'.

An SSL context object is passed to several functions, including `SSL_read(SSL *ssl, char *buf, int num)` and `SSL_write(SSL *ssl, char *buf, int num)`.

△ Coincidentally, the `char *buf` field in each of the two functions listed above contains the plaintext data before and after encryption/decryption. Use this if you can't be bothered to use Wireshark. △

This SSL object contains two further objects, which contain the required values. These are an `ssl3_state_st` object called `s3` (offset `(*ssl)+128`) and an `SSL_SESSION` object called `session` (offset `(*ssl)+304`), and are shown in the diagram below. It should be noted that these will likely change between OpenSSL versions, so you may need to check in Ghidra for new offsets.

Structure Editor [ssl_st, ssl3_state_st, ssl_session_st] [CodeBrowser(2): FMG:/test/libssl.so.1.0.0]

Edit Help

Structure Editor - ssl_st (libssl.so.1.0.0)

Offset	Length	Mnemonic	DataType	Name	Comment
120	8	ssl2_state_st *	ssl2_state_st *	s2	
128	8	ssl3_state_st *	ssl3_state_st *	s3	
136	8	dtls1_state_st *	dtls1_state_st *	d1	
144	4	int	int	read_ahead	
152	8	void _func_3150(int write...	_func_3150 *	msg_callback	
160	8	void *	void *	msg_callback_arg	
168	4	int	int	hit	
176	8	X509_VERIFY_PARAM *	X509_VERIFY_PARAM *	param	
184	8	stack_st_SSL_CIPHER *	stack_st_SSL_CIPHER *	cipher_list	
192	8	stack_st_SSL_CIPHER *	stack_st_SSL_CIPHER *	cipher_list_by_id	
200	4	int	int	mac_flags	
208	8	EVP_CIPHER_CTX *	EVP_CIPHER_CTX *	enc_read_ctx	
216	8	EVP_MD_CTX *	EVP_MD_CTX *	read_hash	
224	8	COMP_CTX *	COMP_CTX *	expand	
232	8	EVP_CIPHER_CTX *	EVP_CIPHER_CTX *	enc_write_ctx	
240	8	EVP_MD_CTX *	EVP_MD_CTX *	write_hash	
248	8	COMP_CTX *	COMP_CTX *	compress	
256	8	cert_st *	cert_st *	cert	
264	4	uint	uint	sid_ctx_length	
268	32	uchar[32]	uchar[32]	sid_ctx	
304	8	SSL_SESSION *	SSL_SESSION *	session	
312	8	GEN_SESSION_CB	GEN_SESSION_CB	generate_session_id	

Search:

Byte Offset: 10 9 8 7 6 5 4 3 2

Component Bits: method type version

Name: ssl_st

Description:

Category: libssl.so.1.0.0/openssl_typ.h

Size: 688 Alignment: 8

align (minimum): ☒ default ☐ ☐ machine: 2

☒ pack ☐ default

ssl_st x ssl3_state_st x ssl_session_st x

Inside of the `ssl3_state_st` object is the `client_random` at an offset of `(*s3)+196`.

Structure Editor [ssl_st, ssl3_state_st, ssl_session_st] [CodeBrowser(2): FMG:/test/libssl.so.1.0.0]

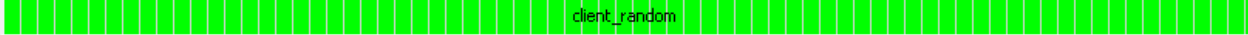
Edit Help

Structure Editor - ssl3_state_st (libssl.so.1.0.0)

Offset	Length	Mnemonic	DataType	Name	Comment
0	8	long	long	flags	
8	4	int	int	delay_buf_pop_ret	
12	8	uchar[8]	uchar[8]	read_sequence	
20	4	int	int	read_mac_secret_size	
24	64	uchar[64]	uchar[64]	read_mac_secret	
88	8	uchar[8]	uchar[8]	write_sequence	
96	4	int	int	write_mac_secret_size	
100	64	uchar[64]	uchar[64]	write_mac_secret	
164	32	uchar[32]	uchar[32]	server_random	
196	32	uchar[32]	uchar[32]	client_random	
228	4	int	int	need_empty_fragments	
232	4	int	int	empty_fragment_done	
236	4	int	int	init_extra	
240	24	SSL3_BUFFER	SSL3_BUFFER	rbuf	
264	24	SSL3_BUFFER	SSL3_BUFFER	wbuf	
288	56	SSL3_RECORD	SSL3_RECORD	rrec	
344	56	SSL3_RECORD	SSL3_RECORD	wrec	
400	2	uchar[2]	uchar[2]	alert_fragment	
404	4	uint	uint	alert_fragment_len	

Search:

Byte Offset: 227 226 225 224 223 222 221 220 219

Component Bits: 

Name: ssl3_state_st

Description:

Category: libssl.so.1.0.0/ssl.h

Size: 1200 Alignment: 8

align (minimum): ☒ default ☐ ☐ machine: 2

☒ pack ☐ default

ssl_st x ssl3_state_st x ssl_session_st x

Inside of the `SSL_SESSION` object is the `master_key` at an offset of `(*session)+20`.

Structure Editor [ssl_st, ssl3_state_st, ssl_session_st] [CodeBrowser(2): FMG:/test/libssl.so.1.0.0]


Edit Help

Structure Editor - ssl_session_st (libssl.so.1.0.0)

Offset	Length	Mnemonic	DataType	Name	Comment
0	4	int	int	ssl_version	
4	4	uint	uint	key_arg_length	
8	8	uchar[8]	uchar[8]	key_arg	
16	4	int	int	master_key_length	
20	48	uchar[48]	uchar[48]	master_key	
68	4	uint	uint	session_id_length	
72	32	uchar[32]	uchar[32]	session_id	
104	4	uint	uint	sid_ctx_length	
108	32	uchar[32]	uchar[32]	sid_ctx	
140	4	uint	uint	krb5_client_princ_len	
144	256	uchar[256]	uchar[256]	krb5_client_princ	
400	8	char *	char *	psk_identity_hint	
408	8	char *	char *	psk_identity	
416	4	int	int	not_resumable	
424	8	sess_cert_st *	sess_cert_st *	sess_cert	
432	8	X509 *	X509 *	peer	
440	8	long	long	verify_result	
448	4	int	int	references	
456	8	long	long	timeout	

Search:

Byte Offset: 67 66 65 64 63 62 61 60 59

Component Bits:  master_key

Name: ssl_session_st

Description:

Category: libssl.so.1.0.0/ssl.h

Size: 600 Alignment: 8

align (minimum) ☒ default ☐ machine: 2

pack ☒ ☐

ssl_st x ssl3_state_st x ssl_session_st x

Once these offsets have been determined, the following script can be updated and used to extract the keys in the correct format.

```
const S3_OFFSET = 128
const CLIENT_RANDOM_OFFSET = 196

const SESSION_OFFSET = 304
const MASTER_KEY_OFFSET = 20

function buf2hex(buffer) { // buffer is an ArrayBuffer
  return [...new Uint8Array(buffer)]
    .map(x => x.toString(16).padStart(2, '0'))
    .join('');
}

const cb = {
  onEnter(args) {
    const ssl = ptr(args[0])
```

```
const s3 = ssl.add(S3_OFFSET).readPointer()
const client_random = s3.add(CLIENT_RANDOM_OFFSET).readByteArray(32)

const session = ssl.add(SESSION_OFFSET).readPointer()
const master_key = session.add(MASTER_KEY_OFFSET).readByteArray(48)

console.log(`CLIENT_RANDOM ${buf2hex(client_random)} ${buf2hex(master_key)}`)
}
}

Interceptor.attach(Module.findExportByName('libssl.so.1.0.0', 'SSL_read'), cb)
Interceptor.attach(Module.findExportByName('libssl.so.1.0.0', 'SSL_write'), cb)
```

Revision #2

Created 3 July 2022 22:39:08 by Jack

Updated 17 August 2022 23:37:24 by Jack