电子科技大学信息与软件工程学院

**实 验 报 告**

学 号 2017221302006

姓 名 周玉川

（实验）课程名称 网络安全协议

理论教师 罗绪成

实验教师 罗绪成

合作队友 蓝浩诚（2017221302004）

**电子科技大学教务处制表**

**电 子 科 技 大 学**

**实 验 报 告**

**学生姓名：周玉川/蓝浩诚 学号：2017221302006／2017221302004指导教师：罗绪成 实验地点：信软楼 实验时间：2019.10.24**

**一、实验名称：**TLS 配置与流量分析实验

**二、实验学时：**2学时

**三、实验目的：**

1）理解 TLS 协议原理；

2）掌握 apache 服务器的 HTTPS 配置方法；

3）掌握 TLS 流量分析方法。

**四、实验原理：**

TLS协议分层

TLS的密码学安全目标包括：保密性、完整性、身份认证

对于保密性来说，通常是通过对称加密组件实现。对称加密的前提是通信双方要有共享密钥，因此需要一个密钥协商组件。TLS的设计中，将上述功能分为：

（1）对称加密传输的记录协议，即： Record Protocol

（2）认证密钥协商的握手协议，即： Handshake protocol

另外，还有三个精助协议

（1） Change Cipher Spec协议

（2）Aert协议

（3） Application Data协议

因此，在设计上，TLS协议是由TLS记录协议（ Record protocol）和TLS握手协议（Handshake Protocol）两层协议构成。记录协议位于下层，握手协议位于上层，记录协议对上层数据包进行封装，然后利用TCP协议进行传输。

（1）TLS记录协议

位于TLS协议的下层，负责安全传输数据，也就是确保数据传输的保密性和完整性。保密性和完整性通过对称密码和消息验证码来完成，因为候选的加密算法和消息验证码算法很多，因此需要通信双方协商来确定一致性，而这个协商过程就是由握手协议

（2）TLS握手协议

TLS捏手协议又细分为：握手协议（ Handshake Protocol）、变更密码规格协议（ Change CipherSpec Protocol）、警告协议（ Alter Protocol）和应用数据协议（ Aplicationdata Protocol）。

TLS Record protocol

SSL记录协议为SSL连接提供两个服务

1. Confidentialit
2. Message Integrity

首先，消息被分制成多个较短的片段（ fragment），然后分别对每个片段进行压缩具体所采用的压缩算法通过前面的揠手协议确定。实际使用中，很少采用压维操作。

接下来，经过压的片段会被加上消息认证码，目的是为了保证消息完整性和数据源发认证。这里利用了MAC的功能。时，为了防止重放攻击，在计算消息认证码时，还加上了片段的編号。这一步操作所需的算法和密钥同样是在握手协议阶段协商获得包括：哈函缴算法、消息认证码所使用的共享密钥

然后，经过压缩的片段拼上消息认证码作为输入，通过对称加密算法进行加密如果加密使用CBC模式，所需的初始化向量（I）通过主密钥（ master secret）生成具体所采用的对称加密算法和共享密钥也是通过前面的捏手协议确定。

最后，上面所得的密文加上记录协议头，则构成了记录协议报文，交由下层的TCP协议发送。

在TCP协议栈中，每层协议都体现为封装上层协议，即：给上层协议报文添加个头部和尾部（根据具体协议而言）.相应地，TLS记录层协议也会给上层协议的报添加一个头部，即 TLS Record header。

TLS HandShake protocol

握手协议的工作原理:握手协议是TLS握手协议的一个子协议，用于在客户端和服务器之间协商产生用记录协议中所要使用的密码算法和共享密钥，基于证书的认证操作也在这个协议中完成。

握手协议实现如下功能：

（1）客户端和服务器端协商TLS协议版本号和一个密码套件

（2）认证对端身份（可选，htp3中一般只认证服务器的身份）

（3）使用密钥协商算法协商共享的 master secret

具体流程如下

（1）交换Hello消息，协商出箅法，交换 random值，检查 session resumption.

（2）交换必要的密码学参效，来允许 client和 server协商出 premaster secret

（3）交换证书和密码学参数，让 client和 server做认证，证明自己的身份

（4）从 premaster secret和交换的 random值，生成出 master secret

（5）把 Security Parameters提供给 record层。

（6）允许 client和 server确认对端得出了相同的 SecurityParameters。

**五、实验内容**

1）配置 TLS 协议分析环境；

2）配置 apache 的 HTTPS 协议；

3）对指定域名发起 HTTPS 请求，抓包分析 TLS 协议流程、提取其中的关键信息。

**六、实验器材（设备、元器件）：**

台式机一台，能够安装一台 ubuntu 虚拟机，能够访问互联网。

**七、实验步骤：**

这次实验由周玉川与蓝浩诚合作共同完成。周玉川负责Apache服务器的TLS配置，以及实验原理及实验步骤的书写。蓝浩诚负责Wireshark的对数据包的捕获及TLS流量分析，以及实验报告的排版整理。

步骤一，搭建环境。

配置两台主机，一台是装有wireshark的windows环境，一台是装有apache的Ubuntu环境，确保两台主机可以连通。

步骤二，Ubuntu服务器上配置apache的HTTPS

1. 创建证书密钥等文件
2. 配置apache服务器
3. 设置防火墙
4. 使其生效

步骤三，TLS流量分析

启动wireshark进行流量分析。

**八、实验结果与分析（含重要数据结果分析或核心代码流程分析）**

步骤一，搭建环境。

配置两台主机，一台是装有wireshark的windows环境，一台是装有apache的Ubuntu环境，确保两台主机可以连通。

先检查两台主机的连通性，如图1-1所示

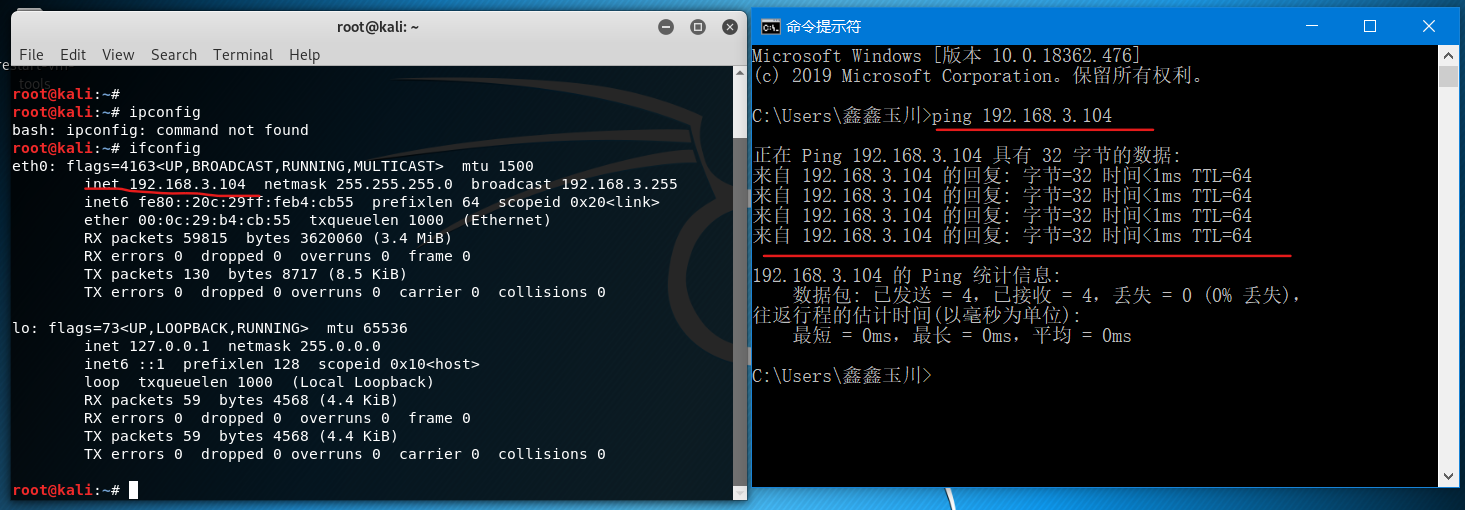


图1-1两台主机可以连接

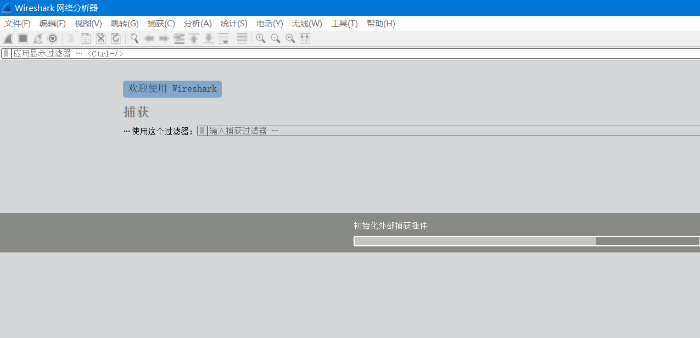
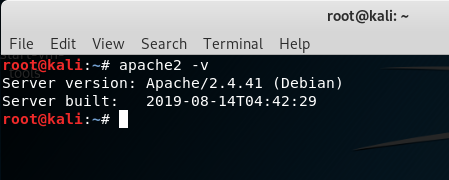
windows和ubuntu分别安装相关软件，如图1-2所示

图1-2安装结束

步骤二，Ubuntu服务器上配置apache的HTTPS

1. 创建证书密钥等文件

输入命令：sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/ssl/private/apache-selfsigned.key -out /etc/ssl/certs/apache-selfsigned.crt

如图2-1-1所示openssl req -x5ß9 -nodes -days 365 -newkey rsa:2048 -keyout /etC/Ss 
l/private/apache•selfsigned. key •out /etc/ssl/certs/apache•setfsigned.crt 
Generating a RSA private key 
writing new private key to '/etc/sst/private/apache•setfsigned.key• 
You are about to be asked to enter information that will be incorporated 
into your certificate request. 
What you are about to enter is What is called a Distinguished Name or a DN. 
There are quite a few fields but you can leave some blank 
For some fields there will be a default value, 
If you enter 
, the field be left blank. 
Country Name (2 letter code) [AU] : CN 
State or province Name (futt name) [Some-Statel :Chengdu 
Locality Name (eg, city) [l :sichuan 
Organization Name leg, company) [Internet Widqits Pty Ltd) •UESTC 

图2-1-1

然后为了实现前向保密性，perfect forward secrecy，还需创建一个强的DH群。命令为

sudo openssl dhparam -out /etc/ssl/certs/dhparam.pem 2048

结果如图2-1-2Country Name (2 letter code) [Aul :cN 
State or Province Name (full name) (Some-Statel : Chengdu 
Locality Name (eg, city) :Sichuan 
Organization Name leg, company) [Internet Widgits Pty Ltd] :UESTC 
Organizational unit Name (eg, section) (l:Software 
common Name (e.g. server FQDN or YOUR name) [l: localhost 
Email Address 
opensst dhparam •out /etc/sst/certs/dhparam.pem 2048 
Generating DH parameters, 2048 bit long safe prime, generator 2 
This is going to take a tong time 

图2-1-2

1. 配置apache服务器

创建apache配置文件，进行强加密设置

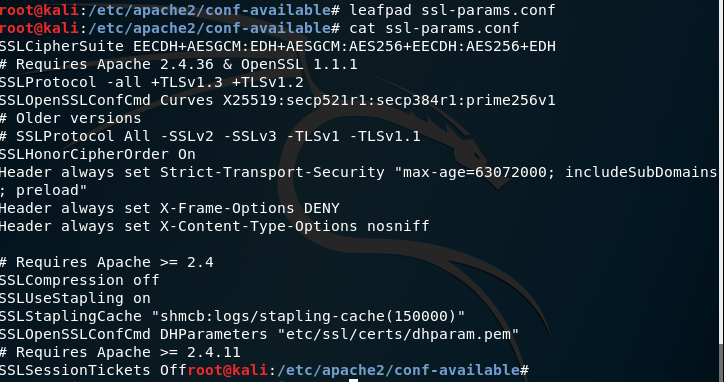
vi /etc/apache2/conf-available/ssl-params.conf

或者 cd /etc/apache2/conf-available vi ssl-params.conf

填入以下配置

|  |
| --- |
| SSLCipherSuite EECDH+AESGCM:EDH+AESGCM:AES256+EECDH:AES256+EDH  # Requires Apache 2.4.36 & OpenSSL 1.1.1  SSLProtocol -all +TLSv1.3 +TLSv1.2  SSLOpenSSLConfCmd Curves X25519:secp521r1:secp384r1:prime256v1  # Older versions  # SSLProtocol All -SSLv2 -SSLv3 -TLSv1 -TLSv1.1  SSLHonorCipherOrder On  Header always set Strict-Transport-Security "max-age=63072000; includeSubDomains; preload"  Header always set X-Frame-Options DENY  Header always set X-Content-Type-Options nosniff  # Requires Apache >= 2.4  SSLCompression off  SSLUseStapling on  SSLStaplingCache "shmcb:logs/stapling-cache(150000)"  SSLOpenSSLConfCmd DHParameters "/etc/ssl/certs/dhparam.pem"  # Requires Apache >= 2.4.11  SSLSessionTickets Off |

结果如图2-2-1所示

图2-2-1

修改默认的Apache SSL默认虚拟主机文件，在/etc/apache2/sites-available/default-ssl.conf。

修改之前先备份sudo cp /etc/apache2/sites-available/default-ssl.conf /etc/apache2/sites-available/default-ssl.conf.bak。

如图2-2-2所示

sudo cp / etc/ apache2/sites-avai labLe/ 
conf /etc/apache2/sites-avai conf. bak 
Is 
det a 
cha rset. conf 
j -common. conf 
Othe r- vhosts-access- log. conf SSI - pa rams. conf 
security. conf 
Localized-error-pages. conf serve- cgi -bin. conf 
cd /etc/apache2/sites-avaiIabIe 
300 - default. conf default-SSI. conf default-SSI . conf. bak 

图2-2-2

修改文件为，红色为修改部分，也可直接复制黏贴，注意文件位置要正确

|  |
| --- |
| <IfModule mod\_ssl.c>  <VirtualHost \_default\_:443>  ServerAdmin 1738733078@qq.com  ServerName 127.0.0.1  DocumentRoot /var/www/html  ErrorLog ${APACHE\_LOG\_DIR}/error.log  CustomLog ${APACHE\_LOG\_DIR}/access.log combined  SSLEngine on  SSLCertificateFile /etc/ssl/certs/apache-selfsigned.crt  SSLCertificateKeyFile /etc/ssl/private/apache-selfsigned.key  <FilesMatch "\.(cgi|shtml|phtml|php)$">  SSLOptions +StdEnvVars  </FilesMatch>  <Directory /usr/lib/cgi-bin>  SSLOptions +StdEnvVars  </Directory>  BrowserMatch "MSIE [2-6]" \  nokeepalive ssl-unclean-shutdown \  downgrade-1.0 force-response-1.0  </VirtualHost>  </IfModule>  # vim: syntax=apache ts=4 sw=4 sts=4 sr noet |

然后修改未加密的Virtual Host file 自动重定向到加密的Virtual Host

为了更好的安全性，通常设置自动重定向到https访问，通过修改配置文件/etc/apache2/sites-available/000-default.conf来完成。

在文件中添加以下命令

RewriteEngine on #开启rewrite引擎

RewriteCond %{HTTPS} !=on # 不是https时执行以下规则，等同于RewriteCond %{SERVER\_PORT} !^443$

RewriteRule ^.\*$ https://%{SERVER\_NAME}$1 [R,L] #^ 匹配行的开始 ，$1引用RewriteRule中的第一个正则(.\*)代表的字符， %{SERVER\_NAME}就是监听的网站域名。[L]：结尾标识。停止重写操作，并不再应用其他重写规则，防止本条规则被后续规则影响。R 强制外部重定向。

ServerAdmin webmaster@localhost 
DocumentRoot /var/wvv/html 
# Redirect 
RewriteEngine on 
Rewritecond !-on 
RewriteRute https://%{SERVER NAME}$I 
# Available loalevels: trace8. 
. tracel. 
debua . 
info. 
notice. 
warn. 

然后输入命令a2enmod rewrite

1. 设置防火墙

ufw enable(如果没安装sudo apt install ufw)

ufw allow 443

ufw allow 80

打开80，443端口

1. 使其生效

输入命令a2enmod ssl

a2enmod 
Considering dependency setenvif for SSI: 
Module setenvif al ready enabled 
Considering dependency mime for SSI: 
Module mime already enabled 
Considering dependency socache_shmcb for SSI: 
Enabling module socache_shmcb. 
Enabling module SSI. 
See /usr/share/doc/apache2/README.Debian.gz on how to configure SSL and create s 
elf-signed certificates. 
To activate the new configuration, you need to run: 
s sternctl restart a achc2 

再输入命令：a2enmod headers

Enabling module headers. 
To activate the new configuration, you need to run: 
systemctl restart apache2 

命令：a2ensite default-ssl

a2ensite default-sst 
Enabling site default-SSI. 
To activate the new configuration, you need to run: 
systemctl reload apache2 

输入命令：a2enconf ssl-params

root@kati : / etc /apache2/sites available* 
Enabling conf ssl•params. 
TO activate the new configuration, you need to run: 
systemctl ret oad apache2 

apache2ctl configtest 
AHß0558: apache2: Could not reliably determine the server's fully qualified doma 
in name, using 127.0.1.1. Set the 'Server Name • directive globally to suppress th 
is message 
Syntax OK 输入命令：apache2ctl configtest。如果没得问题会输出下图

最后重启服务器

systemctl restart apache2。

步骤三，TLS流量分析

启动wireshark进行流量分析。

首先打开wireshark准备流量分析，如图3-1

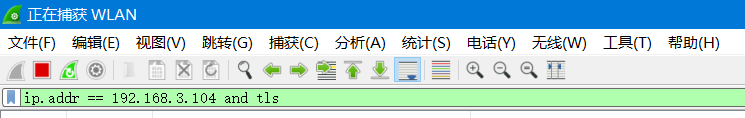


图3-1

然后通过浏览器访问192.168.3.104，如图3-2，3-3所示

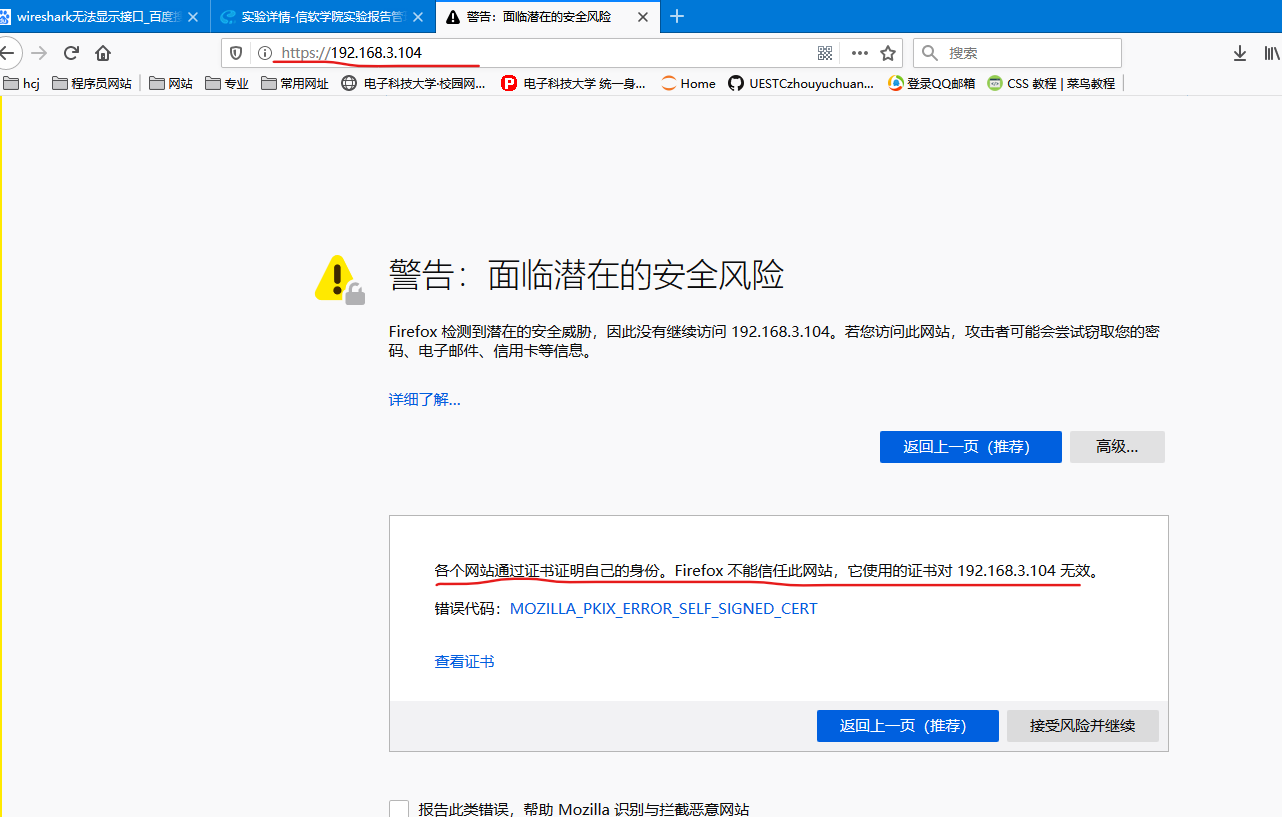
****

图3-2

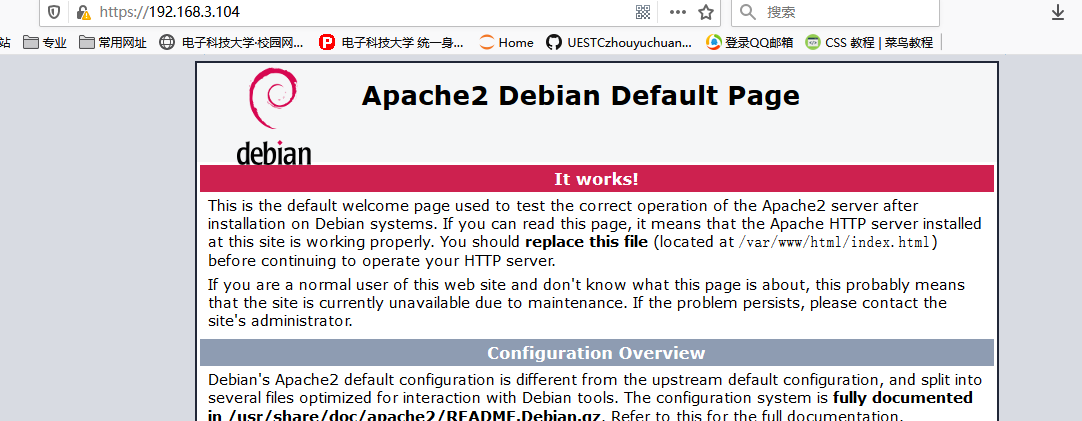
****

图3-3

wireshark结果图如图3-4

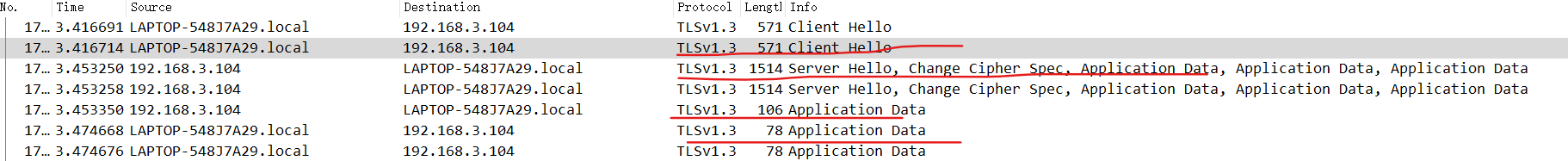
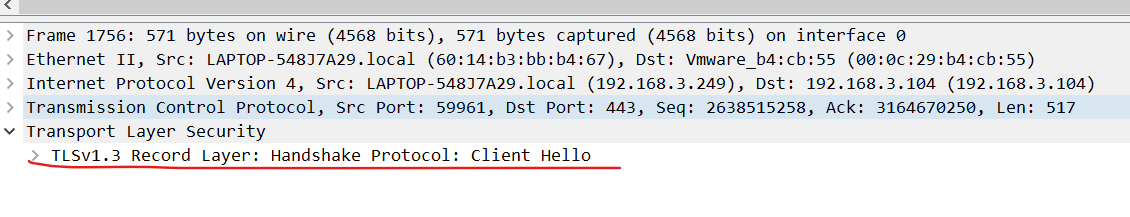


图3-4

首先客户端向服务端发送Client Hello如图3-5

图3-5

服务端向客户端发送Server Hello，通知Change Cipher Spec如图3-6

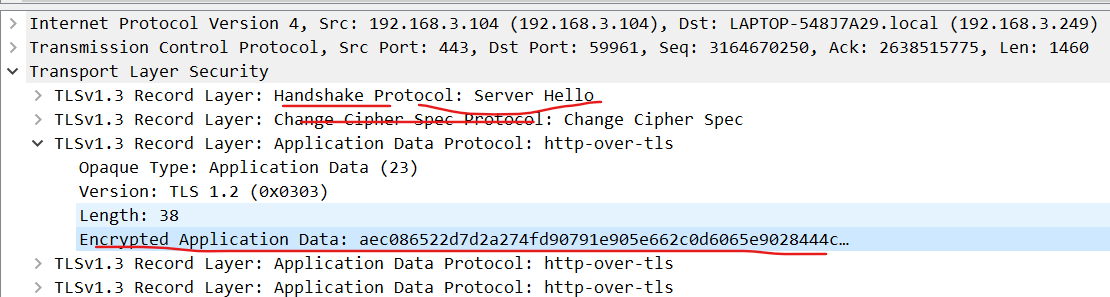


图3-6

然后客户端与服务端用约定好的加密方法传输密文数据，如图3-7所示

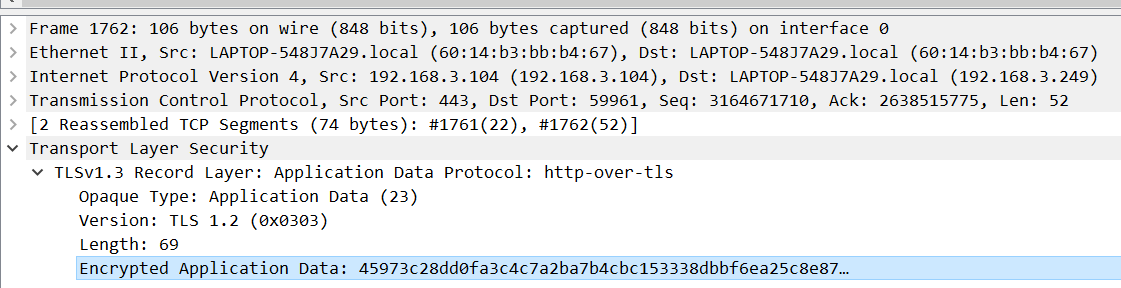


图3-7

中途可能会更换密钥和加密协议如图3-8所示

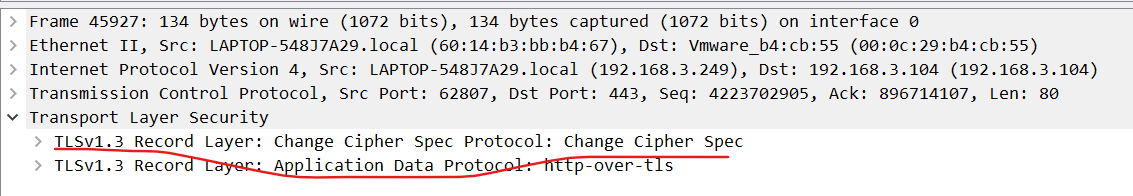


图3-8

**九、总结及心得体会：**

Apache配置HTTPS

在配置apache服务器重定向到https的时候，老师给的方法思路是直接显示的重定向到服务器的ip，但是服务端由于是配置的虚拟机IP地址可能会发生变化（更换局域网时会变），就思考有没有一种通用的方法能够自动重定向到当前ip，经过查阅资料，发现果然有这种通用的方法。方法的思路通过正则匹配，获取本机ip然后再重定向。

通过修改配置文件/etc/apache2/sites-available/000-default.conf，在文件中添加以下命令

RewriteEngine on #开启rewrite引擎

RewriteCond %{HTTPS} !=on # 不是https时执行以下规则，等同于RewriteCond %{SERVER\_PORT} !^443$

RewriteRule ^.\*$ https://%{SERVER\_NAME}$1 [R,L] #^ 匹配行的开始 ，$1引用RewriteRule中的第一个正则(.\*)代表的字符， %{SERVER\_NAME}就是监听的网站域名。[L]：结尾标识。停止重写操作，并不再应用其他重写规则，防止本条规则被后续规则影响。R 强制外部重定向。

TLS流量分析实验结果

通过对数据包进行TLS流量分析，加深了我对TLS协议的认知。TLS协议可以分为两层，一共有5个子协议，分别是上层的Handshake Protocol、ChangeCipherSpec Protocol、Alert Protocol、Application Data Protocol和下层的Record Protocol。在查看数据包以及进行分析的过程中，能够将理论与实际相结合，感受到了课本知识的实际应用。在分析数据包后，加深了自己的印象，对STL有更加深入的理解。

**十、对本实验过程及方法、手段的改进建议：**

无。

**报告评分：**

**指导教师签字：**