**PKI实验报告**

实验报告需求：按实验指导书按步执行，实验报告记录执行的命令，对执行结果截屏，并对指导书中提出的问题进行回答和截屏展示。

## 实验环境设置

实验域名（用姓拼音+年份来命名实验域名，如[www.yang2023.edu.cn](http://www.yang2023.edu.cn)）：www.fan2024.edu.cn

### 1.1 修改SEED环境中的/etc/hosts

|  |
| --- |
| 任务：/etc/hosts文件配置修改 |
| 命令：  Sudo vi /etc/hosts  添加如下内容 10.9.0.80 [www.fan2024.edu.cn](http://www.fan2024.edu.cn) |
|  |

### 1.2 启动Docker，并进入docker环境,准备实验

|  |
| --- |
| 任务：启动docker并进入实验环境 |
| 命令：  cd Labsetup  docker-compose up |
|  |
| dockps |
|  |
| docksh 551 |
|  |

## 实验步骤

### 2.1 Task 1: Becoming a Certificate Authority (CA)

本任务建立一个我们自己的CA。

|  |
| --- |
| 任务：复制并修改openssl.cnf文件，并建立需要的目录和文件，为后面的CA生成做准备  位置：SEED |
| 命令:1.复制命令：cp /usr/lib/ssl/openssl.cnf myCA\_openssl.cnf  2.进入修改：vi myCA\_openssl.cnf  将myCA\_openssl.cnf文件中的  # unique\_subject = no  # copy\_extensions = copy  取消注释 |
| 截图： |

|  |
| --- |
| 任务：执行命令建立ca  位置：SEED |
| 命令：① 在虚拟机VM的当前实验目录下创建CA的工作目录  mkdir -p demoCA && cd demoCA  mkdir -p certs crl newcerts  touch index.txt serial  echo 1000 > serial  cd ..  ② 创建CA的公私钥并自签证书（将使用默认配置文件/usr/lib/ssl/openssl.cnf）  openssl req -x509 -newkey rsa:4096 -sha256 -days 3650 \  -keyout ca.key -out ca.crt \  -subj "/CN=www.fan2024.edu.cn/O=Fan2024 Inc./C=US" \  -passout pass:dees  ③ 查看证书文件ca.crt和私钥文件ca.key  openssl x509 -in ca.crt -text -noout  openssl rsa -in ca.key -text -noout |
| 截图  ①    ②      ③ |

|  |
| --- |
| 任务：执行openssl x509和openssl rsa命令回答以下问题  位置：SEED |
| 问题：What part of the certificate indicates this is a CA’s certificate?  在证书中，有一个字段称为“基本约束”（Basic Constraints），该字段设置为 CA:TRUE 表示这是一个证书颁发机构的证书。 |
| 截图: |
| 问题：What part of the certificate indicates this is a self-signed certificate？  自签名证书的发行者（Issuer）和主体（Subject）字段是相同的。在证书中，如果这两个字段相同，则表明证书是自签名的。 |
| 截图 |
| 问题：In the RSA algorithm, we have a public exponent e, a private exponent d, a modulus n, and two secret numbers p and q, such that n = pq. Please identify the values for these elements in your certificate and key files. |
| 截图  公共指数 e：通常在 RSA 密钥中，e 是一个较小的整数，如 65537（即 0x10001）。    私有指数 d：这是私钥中的一部分，不应在公共文件中公开。它用于解密加密的数据或生成数字签名。      模数 n：这是公钥和私钥共有的部分，是 p 和 q 的乘积。n 的大小决定了 RSA 密钥的安全性。      秘密数 p 和 q：这些是大的质数，用于生成 RSA 密钥对。它们应保密. |

### 2.2 Task 2: Generating a Certificate Request for Your Web Server

本任务为我们自己的网站域名发起一个证书请求

|  |
| --- |
| 任务：为自己的域名生成一个证书请求，注意这里需要同时为多个域名发起请求，除了前面你生成的域名，增加两个其他域名:[www. fan2024A.edu.cn,www.fan2024B.edu.cn](http://www.yang2023A.edu.cn,www.yang2023B.edu.cn)  位置：SEED |
| 命令: 创建www.fan2024.edu.cn的公私钥和CSR，并添加别名:[www. fan2024A.edu.cn,www.fan2024B.edu.cn](http://www.yang2023A.edu.cn,www.yang2023B.edu.cn)  openssl req -newkey rsa:2048 -sha256 \  -keyout server.key -out server.csr \  -subj "/CN=www.fan2024edu.cn/O=Fan2024 Inc./C=US" \  -passout pass:dees \  -addext "subjectAltName = DNS:www.fan2024.edu.cn, \  DNS:www.fan2024A.edu.cn, \  DNS:www.fan2024B.edu.cn" |
| 截图 |

|  |
| --- |
| 任务：用openssl req 命令查看生成csr文件，需要展示csr文件中Subject Alternative Name部分  位置：SEED |
| 命令: openssl req -in server.csr -text -noout |
| 截图: |

### 2.3 Task 3: Generating a Certificate for your server

本任务为我们自己的网站域名签发证书

|  |
| --- |
| 任务：为自己的域名签发证书  位置：SEED |
| 命令  openssl ca -config myCA\_openssl.cnf -policy policy\_anything \  -md sha256 -days 3650 \  -in server.csr -out server.crt -batch \  -cert ca.crt -keyfile ca.key |
| 截图 |

|  |
| --- |
| 任务：用openssl x509命令查看生成crt文件，需要展示crt文件中Subject Alternative Name部分，是否将所有域名都包含进来  位置：SEED |
| 命令: openssl x509 -in server.crt -text -noout |
| 截图 |

### 2.4 Task 4: Task 4: Deploying Certificate in an Apache-Based HTTPS Website

将证书配置到Apache服务器中，启动网站，并通过浏览器测试网站证书的效果

|  |
| --- |
| 任务：将生成好的证书和私钥拷贝docker容器中/certs目录下  位置：docker-www SEED |
| 命令  将证书文件拷贝到共享目录，在虚拟机上执行  cp server.key Labsetup/volumes/  cp server.crt Labsetup/volumes/  将证书文件转移到证书目录并重命名，在容器上执行  cp /volumes/server.key /certs/fan2024.key  cp /volumes/server.crt /certs/fan2024.crt |
| 截图 |

|  |
| --- |
| 任务：进入docker，找到并修改bank32\_apache\_ssl.conf文件中的域名和证书、私钥信息，并重启apache服务  位置：docker-www |
| 命令:  1.在image\_www中创建fan2024\_apache\_ssl.conf，内容如下:  <VirtualHost \*:443>  DocumentRoot /var/www/fan2024  ServerName www.fan2024.com  ServerAlias www.fan2024A.com  ServerAlias www.fan2024B.com  DirectoryIndex index.html  SSLEngine On  SSLCertificateFile /certs/server.crt  SSLCertificateKeyFile /certs/server.key  </VirtualHost>  2.将server.crt和server.key复制进certs中  3. 修改dockerfile  FROM handsonsecurity/seed-server:apache-php  ARG WWWDIR=/var/www/fan2024  COPY ./index.html ./index\_red.html $WWWDIR/  COPY ./fan2024\_apache\_ssl.conf /etc/apache2/sites-available  COPY ./certs/server.crt ./certs/server.key /certs/  RUN chmod 400 /certs/server.key \  && chmod 644 $WWWDIR/index.html \  && chmod 644 $WWWDIR/index\_red.html \  && a2enmod ssl \  && a2ensite fan2024\_apache\_ssl \    CMD tail -f /dev/null  4.重启apache：dockps  dosksh551  service apache2 start |
| 截图 |

|  |
| --- |
| 任务：通过SEED中的Firefox浏览器访问网站，显示结果并说明原因  位置：SEED |
| 截图: |
| 原因分析：由于没有添加证书，证书不受信任，网站访问有风险。 |

|  |
| --- |
| 任务：对SEED中的Firefox浏览器访导入我们自己的CA证书  位置：SEED  方法：在网站about：preferences#privacy中添加。 |
| 截图： |

|  |
| --- |
| 任务：通过SEED中的Firefox浏览器再次访问网站，显示结果并说明原因  位置：SEED |
| 截图： |
| 原因分析：添加证书，证书收到信任，访问网站时受到信任，不再有warning。新证书的有效性、与域名的匹配性、证书链的完整性、内容的安全性、浏览器的兼容性以及系统时间的正确性都得到了保障。 |

### 2.5 Task 5: Launching a Man-In-The-Middle Attack

对一个知名网站进行一次中间人攻击，这个知名网站可以自选，比如东大主页，百度主页等

|  |
| --- |
| 任务：修改apache的bank32\_apache\_ssl.conf配置文件，将被攻击网站域名加入virtualhost配置，并修改SEED中的/etc/hosts文件  位置：docker-www SEED |
| 命令：尝试攻击东大网站  1. 进入image\_www中创建seu\_apache\_ssl.conf，内容如下：  <VirtualHost \*:443>  DocumentRoot /var/www/fan2024  ServerName www.seu.edu.cn  DirectoryIndex index.html  SSLEngine On  SSLCertificateFile /certs/university.crt  SSLCertificateKeyFile /certs/university.key  </VirtualHost>  2.sudo vi /etc/hosts  添加 10.9.0.80 www.seu.edu.cn |
| 截图： |

|  |
| --- |
| 任务：通过SEED中的Firefox浏览器访问被攻击的知名网站，显示结果并说明原因  位置：SEED |
| 命令：在Firefox中输入<https://www.seu.edu.cn>网址，访问网站。 |
| 截图：  原因：证书不受信任，存在warning，无法安全访问网站。 |

### 2.6 Task 6: Launching a Man-In-The-Middle Attack with a Compromised CA

在2.5攻击的基础上，假设我们能够为被攻击网站生成一个合法的证书，再次观察攻击效果

|  |
| --- |
| 任务：用我们的CA为被攻击网站生成一个证书和私钥，并配置到docker的apache服务器中  位置：docker-www SEED |
| 命令:   1. openssl req -newkey rsa:2048 -sha256 -keyout university.key -out university.csr -subj "/CN=www.seu.edu.cn/O=Southeast University/C=US" -passout pass:dees 2. openssl ca -config myCA\_openssl.cnf -policy policy\_anything -md sha256 -days 3650 -in university.csr -out university.crt -batch -cert ca.crt -keyfile ca.key 3. 将university.crt和university.key复制进certs中 4. 修改dockerfile   FROM handsonsecurity/seed-server:apache-php  ARG WWWDIR=/var/www/fan2024  COPY ./index.html ./index\_red.html $WWWDIR/  COPY ./fan2024\_apache\_ssl.conf /etc/apache2/sites-available  COPY ./seu\_apache\_ssl.conf /etc/apache2/sites-available  COPY ./certs/server.crt ./certs/server.key /certs/  COPY ./certs/university.crt ./certs/university.key /certs/  RUN chmod 400 /certs/server.key \  && chmod 400 /certs/university.key \  && chmod 644 $WWWDIR/index.html \  && chmod 644 $WWWDIR/index\_red.html \  && a2enmod ssl \  && a2ensite fan2024\_apache\_ssl \  && a2ensite seu\_apache\_ssl  CMD tail -f /dev/null   1. 在Labsetup中重启apache：   dockps  dosksh551  service apache2 start |
| 截图： |

|  |
| --- |
| 任务：通过SEED中的Firefox浏览器访问被攻击的知名网站，显示结果并说明原因  位置：SEED |
| 命令：在Firefox中输入：<https://www.seu.edu.cn>访问该网站。 |
| 截图  原因：因为为CA生成了一个虚假的证书和私钥，所有可以作为中间人去访问攻击[www.seu.edu.cn](http://www.seu.edu.cn),不会呗拦截或warning。 |