



Práctica de Implementación de TLS

En esta practica, se analizará el tráfico TLS de un servidor web Apache en un entorno configurado en la practica de PKI. Luego, se implementará la autenticación de dos vías TLS, permitiendo a los usuarios acceder al servidor web con certificados de cliente. Además, se mostrará cómo controlar el acceso utilizando campos del certificado, como la afiliación a un sector o grupo de la organización, fortaleciendo así la seguridad y el control de acceso.

Indice

Analisis de Trafico TLS.....	2
Configurar Autenticacion de 2 Vias.....	4
Prerequisito Apache.....	4
Autenticacion de dos Vias.....	5
Creacion de Certificado de usuario.....	5
Exportacion certificado de la CA.....	6
Configuracion de apache para soportar autenticacion cliente.....	6
Control de acceso basado en atributos del usuario.....	7
Opcionales.....	8
Revocación de Certificado y CRL.....	8
SubCA de Personas.....	8
Referencias.....	8



Analisis de Trafico TLS

Arrancar el equipo kali e iniciar el apache2

service apache2 restart

Iniciar el Wireshark y capturar el trafico

navegar a <https://apache.empresa.com>

1. Ver el paquete de Client Hello. ¿que cifradores ofrece el cliente? ¿que contiene la extension de SSL server name? ¿para que sirve?
2. Ver el paquete de ServerHello. Cual es el Cifrador que selecciono el Servidor?

Limpiar el trafico en el wireshark y volver a escuchar

En una linea de comando ejecutar:

```
export SSLKEYLOGFILE=ssl.log
```

en la misma consola ejecutar el comando curl para bajar la pagina web

```
curl --cacert SubCA.crt https://apache.empresa.com
```

El comando muestra la pagina y genera el archivo ssl.log conteniendo las claves generadas durante el intercambio TLS.

```
$ cat ssl.log
SERVER_HANDSHAKE_TRAFFIC_SECRET 0953b02f956a41a3edf13d0a6cbb8d37f0f782eba275e13bdaec6915c0d367a8
ed1f757e2a132ec5d611326e40267fdab8965c330fbd6ca03e5132e325a4cf6f880daa923bee6e8c13774be2fcc5457d
EXPORTER_SECRET 0953b02f956a41a3edf13d0a6cbb8d37f0f782eba275e13bdaec6915c0d367a8
caf9723996caba82577e091c4885bb0117487cef9b4287421fa5de006392264fb0177e061e2254b0575e0df5e88dddf2
SERVER_TRAFFIC_SECRET_0 0953b02f956a41a3edf13d0a6cbb8d37f0f782eba275e13bdaec6915c0d367a8
078290ba369a9ed62de4c2f2c4a2aae312736f0e8a4ee05e5cfd4a2f51260eb301bb735a1491420b0e283b8fb704ca2
CLIENT_HANDSHAKE_TRAFFIC_SECRET 0953b02f956a41a3edf13d0a6cbb8d37f0f782eba275e13bdaec6915c0d367a8
d32f9ff41e552dea9adedf8c06dad5a6f2449fa3af3f34dd4c6f11d579a34c2036e5659d28f19ffd0027a3f26b8bb094
CLIENT_TRAFFIC_SECRET_0 0953b02f956a41a3edf13d0a6cbb8d37f0f782eba275e13bdaec6915c0d367a8
9f169843a02c22564f470203c734d0bab1fff77629262466ecebbec58f9013e6befe37d10f0d3a4985bd824a4c02e5
```

En el Wireshark se ve el trafico encriptado

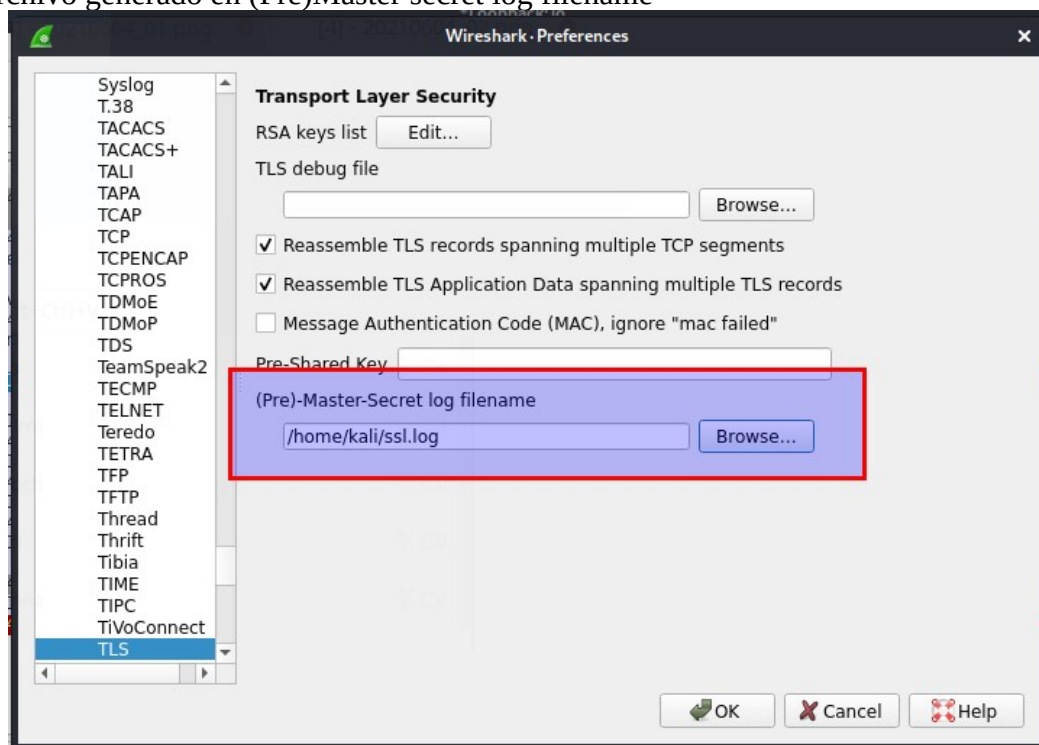


Práctica de Implementación de TLS

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	127.0.0.1	127.0.0.1	TCP	74	39548 → 443 [SYN] Seq=0 Win=65495 Len=0 MSS=65495 SACK_PERM=1 TSval=3341665667 TSecr=0 WS=128
2	0.000024418	127.0.0.1	127.0.0.1	TCP	74	443 → 39548 [SYN, ACK] Seq=0 Ack=1 Win=65483 Len=0 MSS=65495 SACK_PERM=1 TSval=3519109748 TSecr=3341665667 WS=128
3	0.000043579	127.0.0.1	127.0.0.1	TCP	66	39548 → 443 [ACK] Seq=1 Ack=1 Win=65536 Len=0 TSval=3341665667 TSecr=3519109748
4	0.001306729	127.0.0.1	127.0.0.1	TLSv1.3	583	Client Hello
5	0.001339667	127.0.0.1	127.0.0.1	TCP	66	443 → 39548 [ACK] Seq=1 Ack=518 Win=65024 Len=0 TSval=3519109749 TSecr=3341665668
6	0.003686841	127.0.0.1	127.0.0.1	TLSv1.3	1841	Server Hello, Change Cipher Spec, Application Data, Application Data, Application Data, Application Data
7	0.003751669	127.0.0.1	127.0.0.1	TCP	66	39548 → 443 [ACK] Seq=518 Ack=1776 Win=64128 Len=0 TSval=3341665671 TSecr=3519109752
8	0.003859039	127.0.0.1	127.0.0.1	TLSv1.3	140	Change Cipher Spec, Application Data
9	0.005622427	127.0.0.1	127.0.0.1	TCP	66	443 → 39548 [ACK] Seq=1776 Ack=598 Win=65536 Len=0 TSval=3519109753 TSecr=3341665672
10	0.005820003	127.0.0.1	127.0.0.1	TLSv1.3	170	Application Data
11	0.005826896	127.0.0.1	127.0.0.1	TCP	66	443 → 39548 [ACK] Seq=1776 Ack=702 Win=65536 Len=0 TSval=3519109754 TSecr=3341665673
12	0.005914291	127.0.0.1	127.0.0.1	TLSv1.3	369	Application Data
13	0.005920038	127.0.0.1	127.0.0.1	TCP	66	39548 → 443 [ACK] Seq=702 Ack=2079 Win=65280 Len=0 TSval=3341665673 TSecr=3519109754
14	0.006068998	127.0.0.1	127.0.0.1	TLSv1.3	369	Application Data
15	0.006093998	127.0.0.1	127.0.0.1	TCP	66	39548 → 443 [ACK] Seq=702 Ack=2382 Win=65024 Len=0 TSval=3341665673 TSecr=3519109754
16	0.006321542	127.0.0.1	127.0.0.1	TLSv1.3	11066	Application Data, Application Data
17	0.006358483	127.0.0.1	127.0.0.1	TCP	66	39548 → 443 [ACK] Seq=702 Ack=13382 Win=59520 Len=0 TSval=3341665673 TSecr=3519109754
18	0.008750249	127.0.0.1	127.0.0.1	TLSv1.3	90	Application Data
19	0.008786401	127.0.0.1	127.0.0.1	TCP	66	443 → 39548 [ACK] Seq=13382 Ack=726 Win=65536 Len=0 TSval=3519109757 TSecr=3341665676
20	0.008892329	127.0.0.1	127.0.0.1	TCP	66	39548 → 443 [FIN, ACK] Seq=726 Ack=13382 Win=65536 Len=0 TSval=3341665676 TSecr=3519109757
21	0.008906632	127.0.0.1	127.0.0.1	TLSv1.3	90	Application Data
22	0.008930592	127.0.0.1	127.0.0.1	TCP	54	39548 → 443 [RST] Seq=726 Win=0 Len=0

En el Wireshark ir a Edit → Preferences → Protocols-TLS

poner el archivo generado en (Pre)Master secret log filename



Ver el trafico descifrado



Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help				
Apply a display filter ... <Ctrl-F>				
Time	Source	Destination	Protocol	Length Info
1 0.000000000	127.0.0.1	127.0.1.1	TCP	74 39548 - 443 [SYN] Seq=0 Win=65495 Len=0 MSS=65495 SACK_PERM=1 TSval=3341665667 TSecr=0 WS=128
2 0.000024419	127.0.1.1	127.0.0.1	TCP	74 443 - 39548 [SYN, ACK] Seq=0 Ack=1 Win=65483 Len=0 MSS=65495 SACK_PERM=1 TSval=3519109748 TSecr=3341665667 WS=128
3 0.000043579	127.0.0.1	127.0.1.1	TCP	66 39548 - 443 [ACK] Seq=1 Ack=1 Win=65536 Len=0 TSval=3341665667 TSecr=3519109748
4 0.001306729	127.0.0.1	127.0.1.1	TLSv1.3	583 Client Hello
5 0.001339667	127.0.1.1	127.0.0.1	TCP	66 443 - 39548 [ACK] Seq=1 Ack=518 Win=65024 Len=0 TSval=3519109749 TSecr=3341665668
6 0.003608841	127.0.1.1	127.0.0.1	TLSv1.3	1841 Server Hello, Change Cipher Spec, Encrypted Extensions, Certificate, Certificate Verify, Finished
7 0.003751669	127.0.0.1	127.0.1.1	TCP	66 39548 - 443 [ACK] Seq=518 Ack=1776 Win=64128 Len=0 TSval=3341665671 TSecr=3519109752
8 0.005559682	127.0.0.1	127.0.1.1	TLSv1.3	146 Change Cipher Spec, Finished
9 0.005622427	127.0.1.1	127.0.0.1	TCP	66 443 - 39548 [ACK] Seq=1776 Ack=598 Win=65536 Len=0 TSval=3519109753 TSecr=3341665672
10 0.005829003	127.0.0.1	127.0.1.1	HTTP	170 GET / HTTP/1.1
11 0.005926896	127.0.1.1	127.0.0.1	TCP	66 443 - 39548 [ACK] Seq=1776 Ack=702 Win=65536 Len=0 TSval=3519109754 TSecr=3341665673
12 0.005914291	127.0.1.1	127.0.0.1	TLSv1.3	369 New Session Ticket
13 0.005929038	127.0.0.1	127.0.1.1	TCP	66 39548 - 443 [ACK] Seq=702 Ack=2079 Win=65280 Len=0 TSval=3341665673 TSecr=3519109754
14 0.006086996	127.0.0.1	127.0.1.1	TLSv1.3	369 New Session Ticket
15 0.006093998	127.0.0.1	127.0.1.1	TCP	66 39548 - 443 [ACK] Seq=702 Ack=2382 Win=65024 Len=0 TSval=3341665673 TSecr=3519109754
16 0.006321542	127.0.1.1	127.0.0.1	HTTP	11066 HTTP/1.1 200 OK (text/html)
17 0.006358483	127.0.0.1	127.0.1.1	TCP	66 39548 - 443 [ACK] Seq=702 Ack=13382 Win=65536 Len=0 TSval=3341665673 TSecr=3519109754
18 0.008759249	127.0.0.1	127.0.1.1	TLSv1.3	90 Alert (Level: Warning, Description: Close Notify)
19 0.008786401	127.0.1.1	127.0.0.1	TCP	66 443 - 39548 [ACK] Seq=13382 Ack=726 Win=65536 Len=0 TSval=3519109757 TSecr=3341665676
20 0.008893239	127.0.0.1	127.0.1.1	TCP	66 39548 - 443 [FIN, ACK] Seq=726 Ack=13382 Win=65536 Len=0 TSval=3341665676 TSecr=3519109757
21 0.008906032	127.0.1.1	127.0.0.1	TLSv1.3	90 Alert (Level: Warning, Description: Close Notify)
22 0.008930592	127.0.0.1	127.0.1.1	TCP	54 39548 - 443 [RST] Seq=726 Win=0 Len=0

Configurar Autenticacion de 2 Vias

En esta sección se configurara el servidor apache para autenticar a los usuarios con certificado

Prerequisito Apache

crear el archivo de bienvenida `/var/www/html/index.php`

```
<?php
echo "SERVER=". $_SERVER['SERVER_NAME'];
echo "<br>";
echo "USUARIO DEL CERTIFICADO=". $_SERVER['SSL_CLIENT_S_DN_CN'];
?>
```

crear el archivo `/var/www/html/seguro.php`

```
<?php
echo "<h1>zona segura </h1>";
echo "SERVER=". $_SERVER['SERVER_NAME'] . "<br>";
echo "USUARIO DEL CERTIFICADO=" .
$_SERVER['SSL_CLIENT_S_DN_CN'] . "<br>";
echo "Organizacion=". $_SERVER['SSL_CLIENT_S_DN_O'] . "<br>";

echo "<h2> variables</h2>";
foreach ($_SERVER as $k=>$v)
echo $k . "=>" . $v . "<br>";

?>
```

Verificar que el archivo de configuracion de apache este de la siguiente manera: se esta

```
<VirtualHost *:443>
    ServerName apache.empresa.com
    SSLEngine on
```



Práctica de Implementación de TLS

```
SSLCertificateFile "/etc/apache2/apache.empresa.com.crt"  
SSLCertificateKeyFile "/etc/apache2/apache.key"
```

```
DirectoryIndex index.php index.html
```

```
ErrorLog ${APACHE_LOG_DIR}/SSLerror.log  
CustomLog ${APACHE_LOG_DIR}/SSLaccess.log combined  
</VirtualHost>
```

Navegar a <https://apache.empresa.com> y verificar que la pagina default cambio

Autenticacion de dos Vias

Creacion de Certificado de usuario

Creacion de dos certificados “Usuario 1” y “Usuario 2” el Usuario 2 estara en la organización contable_org el usuario 1 en la organización seguridad

El certificado debe ser firmado por la subCA y se debe aplicar el template TLS_client

Create x509 Certificate

Source Subject Extensions Key usage Netscape Advanced Comment

Signing request

☐ Sign this Certificate signing request apache.empresa.com

☒ Copy extensions from the request Show request

☐ Modify subject of the request

Signing

☐ Create a self signed certificate

☒ Use this Certificate for signing SubCA

Signature algorithm SHA 256

Template for the new certificate

[default] TLS_client

Apply extensions Apply subject Apply all

OK Cancel



Create x509 Certificate

Source Subject Extensions Key usage Netscape Advanced Comment

Internal Name

Distinguished name

countryName	AR	organizationalUnitName	seguridad	Distinguished name
stateOrProvinceName	CABA	commonName	Usuario 1	
localityName	CABA	emailAddress	usuario1@mail.com	
organizationName	seguridad			

Type	Content	Add	Delete
------	---------	-----	--------

Private key

Usuario 1 (RSA:2048 bit) ☐ Used keys too [Generate a new key](#)

OK Cancel

Exportacion certificado de la CA

Poner el certificado de la CA en formato PEM en [/etc/apache2/SubCA.pem](#)

Nota: es importante que el archivo contenga la cadena completa de certificación (chain).

Configuracion de apache para soportar autenticacion cliente

Incorporar al Virtualhost del puerto 443 las siguientes directivas que configuran la autenticacion con certificados validando con la autoridad certificante

```
SSLCACertificateFile "/etc/apache2/SubCA.pem"

SSLVerifyClient require
SSLVerifyDepth 3
<Location / >
    SSLOptions +StdEnvVars -ExportCertData
    Options FollowSymLinks
    AllowOverride None
</Location>
```

Navegar al sitio <https://empresa.apache.com>, ahora dara acceso denegado

Configurar navegador para que tenga el certificado del usuario en la key store

Navegar al sitio <https://empresa.apache.com>, ahora pedira el certificado y la pagina principal reconoce el usuario.



Nota: si no arranca el apache se pueden ver los logs en el directorio */var/log/apache2*

Control de acceso basado en atributos del usuario

Configurar el virtualhost incorporando la protección del archivo seguro.php

```
<VirtualHost *:443>
    ServerName apache.empresa.com
    SSLEngine on
    SSLCertificateFile "/etc/apache2/apache.empresa.com.crt"
    SSLCertificateKeyFile "/etc/apache2/apache.key"
    SSLCACertificateFile "/etc/apache2/SubCA.pem"

    SSLVerifyClient require
    SSLVerifyDepth 3
    DirectoryIndex index.php index.html
    <Location / >
        SSLOptions +StdEnvVars -ExportCertData
        Options FollowSymLinks
        AllowOverride None
    </Location>

    <Location /seguro.php >
        SSLOptions +StdEnvVars -ExportCertData
        SSLRequire %{SSL_CLIENT_S_DN_O} in {"contable_org",
"desarrollo"}
        Options FollowSymLinks
        AllowOverride None
    </Location>

    ErrorLog ${APACHE_LOG_DIR}/SSLerror.log
    CustomLog ${APACHE_LOG_DIR}/SSLaccess.log combined
</VirtualHost>
```

La opción `SSLOptions` entrega los atributos del certificado en el contexto del PHP

La opción `SSLRequire` obliga a que el certificado tenga la organización “contable_org” para poder acceder al recurso.

1) Navegar al sitio presentar las credenciales del “Usuario 2” y ver que puede acceder a <https://apache.empresa.com/seguro.php>

2) Cerrar el navegador navegar nuevamente al sitio y presentar las credenciales del “Usuario 1” ver que no puede acceder a <https://apache.empresa.com/seguro.php>

ss



- 3) ¿que contiene la variable de PHP SSL_PROTOCOL?
- 4) ¿que contiene la variable de PHP SSL_CIPHER?
- 5) ¿que contiene la variable de PHP SSL_CLIENT_I_DN_CN?

Opcionales

Revocación de Certificado y CRL

1) Entrar a la xca y revocar el certificado de Usuario2, ver que aun puede entrar al apache.
¿por qué?

2) en la xca generar una CRL y exportarla como /home/kali/SubCAcrl.pem
luego poner en el archivo de configuracion de apache las siguientes directivas:

```
SSLCARevocationFile /home/kali/SubCAcrl.pem  
SSLCARevocationCheck leaf
```

SubCA de Personas

dar de baja los certificados de Usuario1 y Usuario2. Definir una SubCAPersonas emitir los certificados de Usuario1 y Usuario2 con esa subca y cambiar la configuracion para que permita todos los usuarios emitidos por dicha SubCAPersonas

```
SSLRequire %{SSL_CLIENT_I_DN_CN} == "SubCAPersonas"
```

Referencias

<https://blog.didierstevens.com/2020/12/14/decrypting-tls-streams-with-wireshark-part-1/>

https://httpd.apache.org/docs/2.4/ssl/ssl_howto.html

https://httpd.apache.org/docs/2.4/mod/mod_ssl.html#sslrequire

http://httpd.apache.org/docs/2.2/mod/mod_ssl.html#ssloptions

http://httpd.apache.org/docs/2.2/mod/mod_ssl.html#envvars

Using SSL Client Certificates with PHP <http://cweiske.de/tagebuch/ssl-client-certificates.htm>

<https://cwiki.apache.org/confluence/display/HTTPD/NameBasedSSLVHostsWithSNI>

<https://serverfault.com/questions/366412/requiring-client-certificate-issued-by-a-specific-intermediate-ca-in-apache>