

Universidad Nacional Autónoma de México

Facultad de Ciencias

Redes de Computadoras

Práctica 2

Protocolo HTTP e instalación de servidor web Apache

Profesor: Paulo Contreras Flores

Ayudante Lab: José Daniel Campuzano Barajas

Objetivo

Que el alumno instale y configure un servidor Web usando el protocolo HTTPS para el envío de información cifrada.

Pondrá en práctica el concepto de Protocolos por niveles visto en clase.

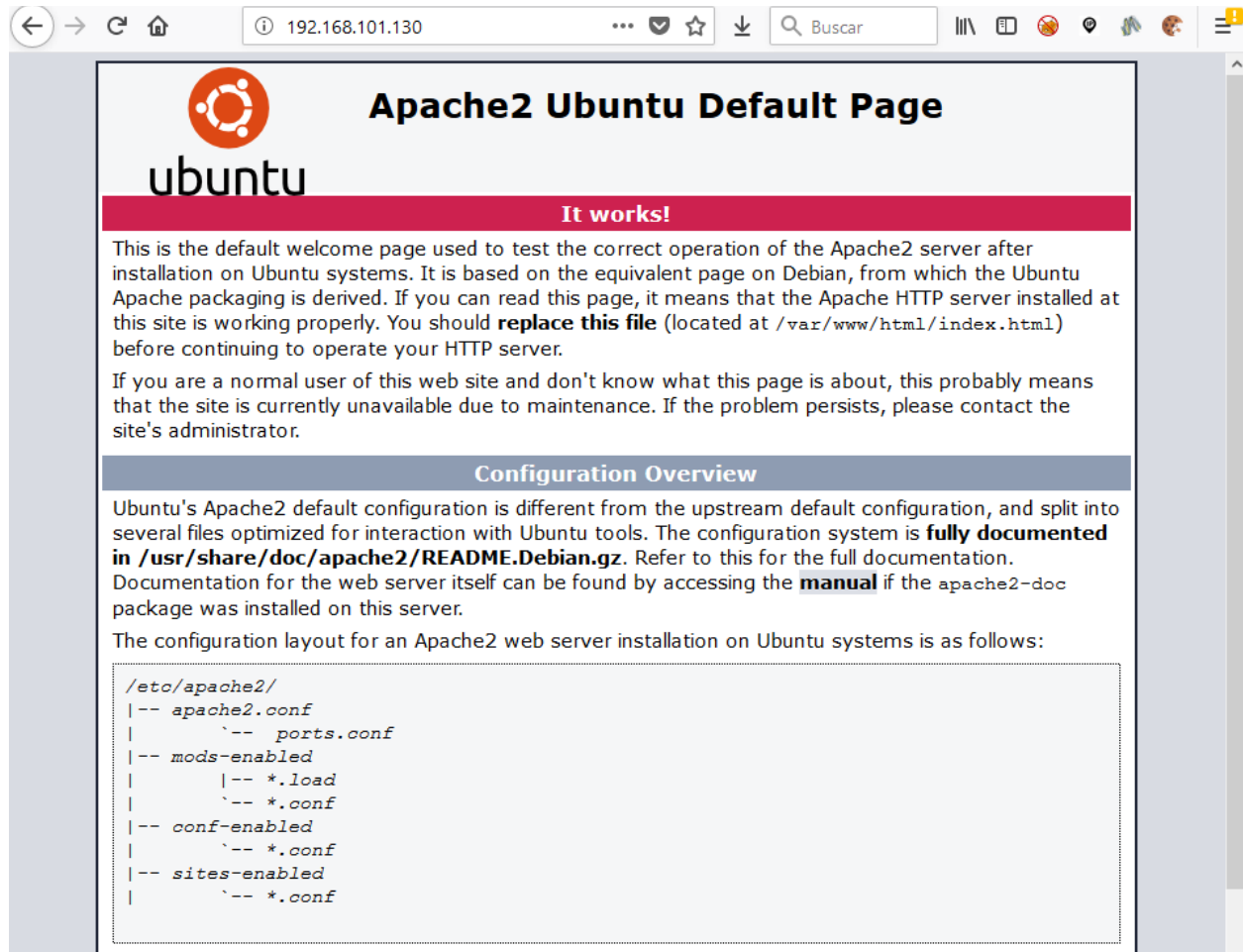
Conocerá el protocolo de la Capa de aplicación HTTP

Introducción

El protocolo HTTPS (HyperText Transfer Protocol Secure) es un protocolo de la Capa de aplicación que permite el envío de información cifrada usando los protocolos HTTP y SSL/TLS. El protocolo HTTP envía información en claro a través del medio, el protocolo SSL/TLS es el encargado de encapsular el protocolo HTTP para ser enviado de manera cifrada.

Desarrollo

1. Instalación del servidor HTTP.
 - a. Instalar el servidor web Apache con soporte para ejecutar programas en Python. Realizar la instalación por medio del gestor de paquetes.
 - b. Después de instalar el servidor web, abrir el navegador web e ingresar a la dirección `http://127.0.0.1`, si se instaló de manera correcta se mostrará una página parecida a la mostrada en la Figura 1.



- c. El gestor de paquetes tiene distintas rutas predeterminadas al momento de instalar el servidor web. Por defecto la carpeta para contenido web es `/var/www/html`. Ingresar al directorio y crear el archivo **form.html** tal y agregar el código mostrado en el Código 1.

```

<html>
<head>
    <title>Login</title>
</head>
<body>
    <form action="/cgi-bin/index.py" method="get">
        User: <input type="text" name="user">
        Pass: <input type="password" name="pass">
        <input type="submit" name="Enviar">
    </form>
</body>
</html>

```

- d. Crear el archivo **index.py** dentro del directorio `/usr/lib/cgi-bin` y agregar el Código 2. Posteriormente cambiar los permisos de ejecución del archivo a lectura y ejecución con el comando: **chmod +x index.py**.

```

#!/usr/bin/python
import cgi

print "Content-type: text/html"
print

print"""
Hola mundo
"""

form=cgi.FieldStorage()
print "<p>User:", form["user"].value
print "<p>Pass:", form["pass"].value

```

- e. Con ambos códigos creados en sus respectivos directorios, proceder a realizar los cambios en la configuración del servidor Apache. Por defecto los archivos de configuración se encuentran dentro del directorio `/etc/apache2/`. Editar el archivo **mime.conf** que se encuentra dentro del directorio `/etc/apache2/mods-enabled/` agregando las siguientes configuraciones en la línea 200:

```

AddHandler cgi-script .cgi
AddHandler cgi-script .py

```

- f. En este punto se han realizado las siguientes actividades:
- Instalación del servidor web Apache
 - Creación del archivo `form.html` dentro del directorio `/var/www/html/`
 - Creación del archivo `index.py` dentro del directorio `/usr/lib/cgi-bin/`
 - Modificación del archivo `mime.conf` añadiendo el Código 3 en la línea 200
- g. Ingresar desde el navegador web a la dirección `http://192.168.101.130/form.html` y validar que se muestre un formulario similar al mostrado en la Figura 2.

User: Pass:

- h. Instalar el analizador de protocolos Wireshark con el comando **apt-get install wireshark**. Lanzar la aplicación y colocar un filtro para el puerto 80 TCP.
- i. Llenar el formulario mostrado en la **sección g** con datos aleatorios y presionar el botón enviar.

Hola mundo

User: admin

Pass: sup3rp4ssw0rd

- j. Verificar los cambios en el analizador de protocolos y buscar el nombre de usuario y la contraseña como se muestra en la Figura 3.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.101.1	192.168.101.130	TCP	66	52668 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
2	0.000600	192.168.101.130	192.168.101.1	TCP	66	80 → 52668 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460 SACK_PERM=1
3	0.000760	192.168.101.1	192.168.101.130	TCP	54	52668 → 80 [ACK] Seq=1 Ack=1 Win=65536 Len=0
4	0.001328	192.168.101.1	192.168.101.130	HTTP	558	GET /cgi-bin/index.py?user=admin&pass=sup3rp4ssw0rd&Enviar=Enviar+consulta HTTP/1.1
5	0.001933	192.168.101.130	192.168.101.1	TCP	60	80 → 52668 [ACK] Seq=1 Ack=505 Win=30272 Len=0
6	0.004095	192.168.101.130	192.168.101.1	HTTP	294	HTTP/1.1 200 OK (text/html)
7	0.143228	192.168.101.1	192.168.101.130	TCP	54	52668 → 80 [ACK] Seq=505 Ack=241 Win=65280 Len=0
12	5.098657	192.168.101.1	192.168.101.130	TCP	54	52668 → 80 [FIN, ACK] Seq=505 Ack=241 Win=65280 Len=0
13	5.099699	192.168.101.130	192.168.101.1	TCP	60	80 → 52668 [FIN, ACK] Seq=241 Ack=506 Win=30272 Len=0
14	5.099763	192.168.101.1	192.168.101.130	TCP	54	52668 → 80 [ACK] Seq=506 Ack=242 Win=65280 Len=0

> Frame 4: 558 bytes on wire (4464 bits), 558 bytes captured (4464 bits) on interface 0

> Ethernet II, Src: Vmware_c0:00:08 (00:50:56:c0:00:08), Dst: Vmware_dd:d9:5a (00:0c:29:dd:d9:5a)

> Internet Protocol Version 4, Src: 192.168.101.1, Dst: 192.168.101.130

> Transmission Control Protocol, Src Port: 52668, Dst Port: 80, Seq: 1, Ack: 1, Len: 504

▼ Hypertext Transfer Protocol

> GET /cgi-bin/index.py?user=admin&pass=sup3rp4ssw0rd&Enviar=Enviar+consulta HTTP/1.1

Host: 192.168.101.130

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:61.0) Gecko/20100101 Firefox/61.0

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8

Accept-Language: es-MX,en;q=0.8,en-US;q=0.5,en;q=0.3

Accept-Encoding: gzip, deflate

Referer: http://192.168.101.130/form.html

Cookie: PHPSESSID=m1r8c461mqdho1amqa9nqm8b4

Connection: keep-alive

Upgrade-Insecure-Requests: 1

[Full request URI: http://192.168.101.130/cgi-bin/index.py?user=admin&pass=sup3rp4ssw0rd&Enviar=Enviar+consulta]

[HTTP request 1/1]

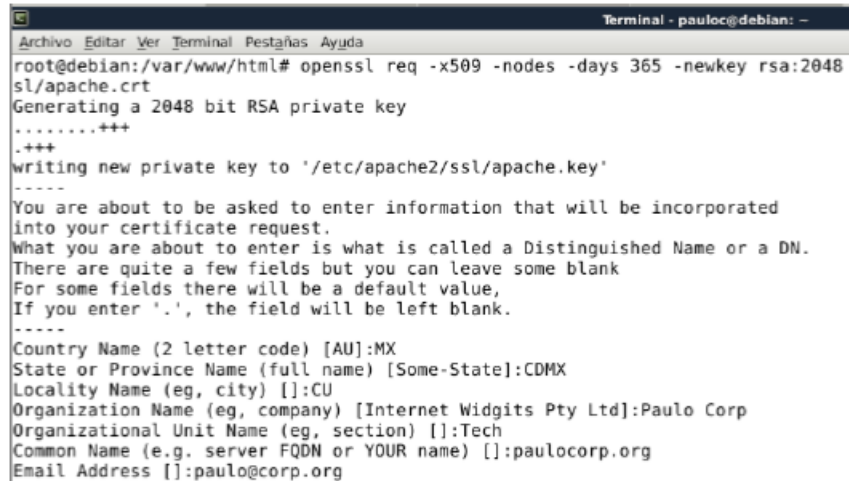
[Response in frame: 6]

2. Configuración HTTPS Apache

- a. continuación, se enlista a manera de resumen lo pasos para habilitar el protocolo i. i. i. HTTPS dentro de Apache:
 - ii. Habilitar el módulo SSL de apache
 - iii. Crear certificados SSL auto firmados
 - iv. Configurar Apache para usar SSL
 - v. Habilitar el servidor con la nueva configuración
- b. Para habilitar el módulo SSL es necesario ingresar al directorio `/etc/apache2/`. El comando **a2enmod** facilita la tarea de habilitar módulos, en este caso particular habilitar el módulo SSL de la siguiente manera: **a2enmod ssl**.

- c. Crear certificados SSL auto firmados haciendo uso de la herramienta **openssl**. Es necesario crear un par de llaves para poder hacer uso de HTTPS. En concreto el comando para la generación de llaves es el siguiente:
openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/apache2/apache.key -out /etc/apache2/apache.crt

Se solicitarán los datos mostrados en la Figura 4, para este ejemplo se utilizaron datos de ficticios. Utilice sus propios datos para la generación del certificado.



```
Terminal - pauloc@debian: -
Archivo Editar Ver Terminal Pestañas Ayuda
root@debian:/var/www/html# openssl req -x509 -nodes -days 365 -newkey rsa:2048
ssl/apache.crt
Generating a 2048 bit RSA private key
.....+++
.+++
writing new private key to '/etc/apache2/ssl/apache.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 will be left blank.
-----
Country Name (2 letter code) [AU]:MX
State or Province Name (full name) [Some-State]:CDMX
Locality Name (eg, city) []:CU
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Paulo Corp
Organizational Unit Name (eg, section) []:Tech
Common Name (e.g. server FQDN or YOUR name) []:paulocorp.org
Email Address []:paulo@corp.org
```

- d. Una vez que se tienen los certificados creados, es necesario especificar que se hará uso de certificados y donde se encuentran. Estas especificaciones se realizan en el archivo de configuración **default-ssl** que se encuentra en el directorio **/etc/apache/sites-available/**. Editar la línea 42 y 43 (aproximadamente) para cambiar la ruta actual por la ruta de los certificados recién creados.
- e. Después de realizar las configuraciones, se requiere cargar el archivo de configuraciones. La herramienta **a2ensite** permite habilitar la nueva configuración: **a2ensite default-ssl**. Reiniciar el servidor apache y verificar que se haya realizado correctamente la información accediendo desde el navegador web al recurso **https://192.168.101.130/form.html**.
- f. Repetir el procedimiento de la **sección i** y **sección j**.

Cuestionario

1. ¿De qué manera influyo la implementación de SSL/TLS a los códigos form.html e index.py? ¿Fue necesario realizar cambios en el código? Justifique su respuesta relacionándola con el concepto de Encapsulado de protocolos y Pila de protocolos de internet.
2. ¿Qué son y para qué sirven los certificados digitales?
3. ¿Cuáles sitios conoce que utilicen HTTPS y cuál podría ser una posible razón de su implementación?
4. En el código **form.html** cambie el método *get* por *post* y realice una nueva captura de Wireshark. ¿Cuál es la diferencia que se aprecia en la captura entre el método *get* y el método

post usando HTTP? ¿Cuál es la diferencia que se nota en el navegador web cuando se usa cada uno de los métodos?

5. Investigue a que hace referencia la dirección IP 127.0.0.1 y lo relacionado a ella.

Notas adicionales

- El reporte se entrega de manera individual.
- Registrar en el reporte los pasos que sean considerados necesarios para explicar cómo se realizó la práctica, incluir capturas de pantalla que justifiquen los resultados obtenidos.
- Incluir las respuestas del Cuestionario en el reporte.
- Se pueden agregar posibles errores, complicaciones, opiniones, críticas de la práctica o del laboratorio, o cualquier comentario relativo a la práctica.
- Subir los archivos relacionados con la práctica al Moodle o entregar el reporte impreso en papel al inicio del laboratorio.
- Fecha de entrega jueves 30 de agosto de 2018 antes de las 23:59 hrs.