Managing a Web Server

Lab 5: SSL & CMS

Introduction

This lab demonstrates the creation of self-signed SSL certificates, and how to install the open source CMS - WordPress. It is important to carefully proceed with each step as there are lots of places to make mistakes. Create a logbook showing the ***key steps*** to upload this to the Moodle once completed.

Before Starting

Labs 2, 3 and 4 probably did all sorts of damage to your virtual machine. It would be sensible to re-import the MAWS.ova to VMware, then install apache2, php and MariaDB again – follow the steps in the LAMP lab.

Part 0: Check OpenSSL Version

At the top of the logbook document, please specify the following

- 0.1 The OpenSSL version running on the server
- 0.2 The OpenSSL build date
- 0.3 The directory in which OpenSSL is stored on the server
- 0.4 The directory of the OpenSSL engine on the server

Part 1: Create an SSL certificate

1.1 Using OpenSSL generate an x509 certificate, a 365 day key with a size of 2048 and provide a path for the key and the certificate:

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

You will be prompted for PEM passphrase, enter: **edinburghcollege**
The rest of the questions about country code etc are not important.

1.2 Now make a copy of the default-ssl.conf in sites-available:

```
cp default-ssl.conf default-ssl.conf.BACKUP
```

There are lots and lots of comments in the default-ssl.conf file, clean it up to look exactly like figure 1 this note the paths to the certificate and key that were created in step 1.1

```
<IfModule mod_ssl.c>
        <VirtualHost _default_:443>
                DocumentRoot /var/www/html
                ErrorLog ${APACHE_LOG_DIR}/error.log
                CustomLog ${APACHE_LOG_DIR}/access.log combined
                SSLCertificateFile /etc/ssl/certs/apache-selfsigned.crt
                SSLCertificateKeyFile /etc/ssl/private/apache-selfsigned.key
        </VirtualHost>
</IfModule>
```

Figure 1: Example SSL File

1.3 Now enable the Apache ssl configuration file and the Apache ssl module, if Apache won't start you should run:

```
apache2ctl configtest

a2enmod rewrite ssl

a2ensite default-ssl.conf <- enter the PEM password from step 1.1 here

systemctl restart apache2
```

1.4 Now go to https://localhost in Firefox and accept the certificate, it isn't trusted because it is not signed, but this shows the certificate is working. You should see the https:// in the URL.

Part 2: Create a WordPress Database

2.1 Now create a database for WordPress, you'll need this password and username (shown in red) to get into WordPress so avoid any typos.

```
mariadb -u root -p

CREATE DATABASE wordpress;

GRANT ALL PRIVILEGES ON wordpress.* TO 'wordpress-user'@'localhost' IDENTIFIED
BY 'edinburghcollege';

FLUSH PRIVILEGES;

quit;
```

Part 3: Download latest WordPress

3.1 You use the **wget** program - if it is not installed install it using **apt**, change directory to the tmp folder to download and extract WordPress, notice they provide it in a gzip.

```
cd /tmp

wget http://wordpress.org/latest.tar.gz

tar -xvf latest.tar.gz
```

3.2 Copy the new WordPress site to the www directory and remove the default index file:

```
cp -rf wordpress/* /var/www/html
rm /var/www/html/index.html
```

3.3 Now set permissions and groups

```
chmod -R 775 /var/www/html
chgrp -R www-data /var/www/html
```

Run this command, you should see the WordPress files:

```
ls -la /var/www/html
```

3.4 Before loading WordPress ensure you have php installed (see lab 3). There are two steps, installing **`php libapache2-mod-php php-mysql`** and then changing Apache mods-enabled/dir.conf to put php as the first choice in the list. Now go to https://localhost and confirm the security exception, you should see something like figure 2.

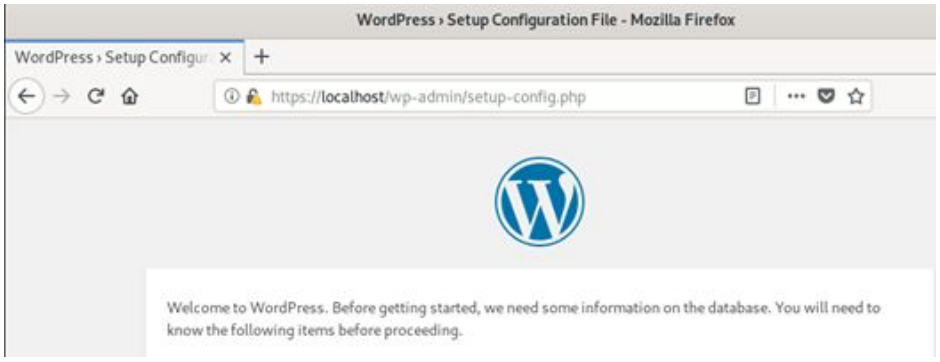

Figure 2: HTTPs working with padlock

Click 'Lets Go' at the bottom of the page and change only the database username and the database password, to the ones that you set with the GRANT command in Part 2. If you get an error about there being no wp-config.php file you will have to create that file.

In this case create the file in **`/var/www/html`** and then run the **`chmod`** and **`chgrp`** commands again from step 3.2. NOTE: You need to type :set paste into vi in normal mode, then go into insert mode then press CTRL+SHIFT+V to paste the config into the file.

3.5 Set the WordPress database options to reflect the database and user that you created in Part 2

Part 4: Upload Work

If you made it through error free, crop and annotate your screenshots and upload to the Moodle.