DAW Práctica 2.4: Certificados Apache

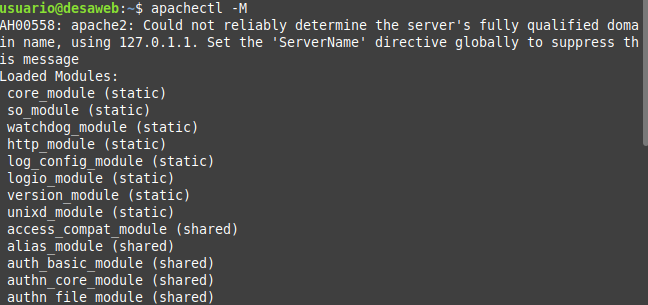
**Elabora un documento donde figuren todos los pasos realizados con las pantallas significativas, explicando cada uno de los pasos**.

**Procedimiento:**

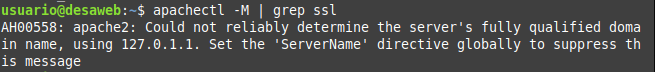
Apache

1. En apache utilizaremos el módulo SSL con su configuración por defecto (default-ssl). Para ver que módulos tenemos activos en apache utilizamos el siguiente comando:

apachectl –M

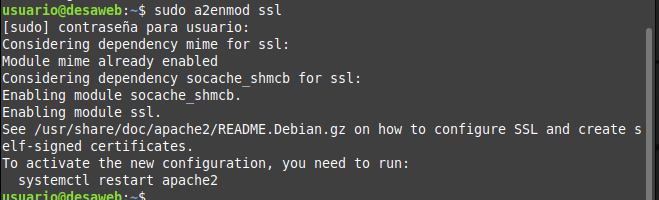


Si queremos filtrar los resultados obtenidos ya sabéis que podemos añadir | grep ssl.

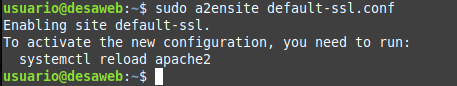


1. En caso de que no esté activo deberemos habilitarlo:

sudo a2enmod ssl



1. Habilitar la configuración por defecto:

sudo a2ensite default-ssl.conf

1. Reiniciar el servicio.



1. Creamos un directorio llamado ssl en /etc/apache2



1. Dentro del directorio recientemente creado generamos la clave privada con el cifrado des3 y la longitud 2048 bits. Pedirá introducir el nombre que vamos a generar como clave.

sudo openssl genrsa -des3 -out server.key 2048



1. Completa las funcionalidades del anterior comando

***genrsa***

***-des3***

***-out***

***Numbits(2048)***

Genrsa es un comando de openssl que nos permite crear claves privadas.

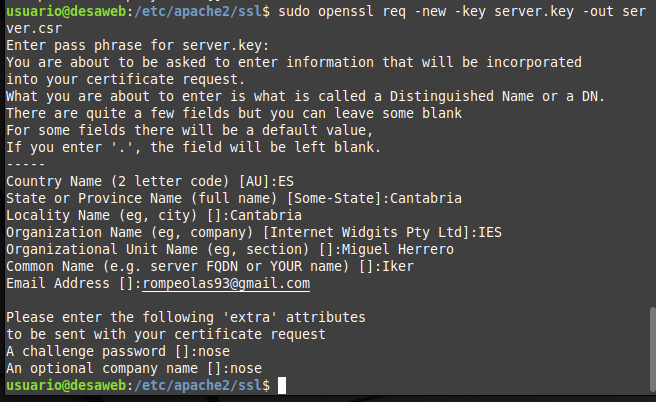
-des3 cifra la clave privada antes de mandarla al archivo

-out sirve para especificar el nombre donde se guardara la clave privada cifrada

Numbits(2048) indica la longitud en bits que tendra la clave, 2048 es el estandar minimo.

1. Crear la solicitud de certificado con la llave con el siguiente comando (contestar a las preguntas que nos hagan):

sudo openssl req -new -key server.key -out server.csr



1. Completa las funcionalidades del anterior comando

***req***

***-new***

***-key***

***-out***

Req se usa para generar y procesar solicitudes de firma de certificado

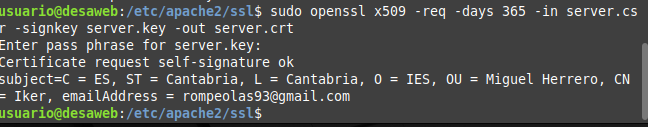
-new es para generar una nueva solicitud

-key especifica el nombre del archivo donde se encuentra la clave

-out se utiliza para especificar el archivo de salida donde se guardará el resultado de un comando, ya seaque se haya podido realizar correctamente o no.

1. Crear el certificado digital auto-firmado usando la clave privada:

sudo openssl x509 -req -days 365 -in server.csr -signkey server.key -out server.crt



1. Competa las funcionalidades del anterior comando

***x503***

***-req***

***-days***

***-in***

***-signkey***

***-out***

X509 indica que el comando trabajo con un certificado en formato X509 que es un estandar de certificados digitales que define su estructura y contenido.

-req especifica que elcomando procesara una solicitud de firma de certificado.

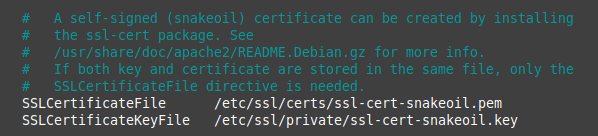
-days 365 define la validez del certificado en días.

-in especifica el archivo de entrada que contiene la entidad que solicita el certificado

-signkey especifica que el certificado sera firmado con una clave privada, específicamente la que se le indique.

-out define el archivo de salida donde se guarda el resultado de la solicitud.

1. Incluir el certificado en Apache modificando el fichero /etc/apache2/sites- available/default-ssl.conf, reemplazando la dirección del certificado y la clave que se ha creado previamente.



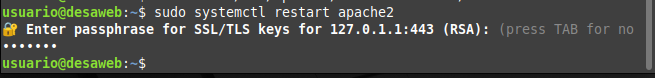




1. Reiniciar el servicio.

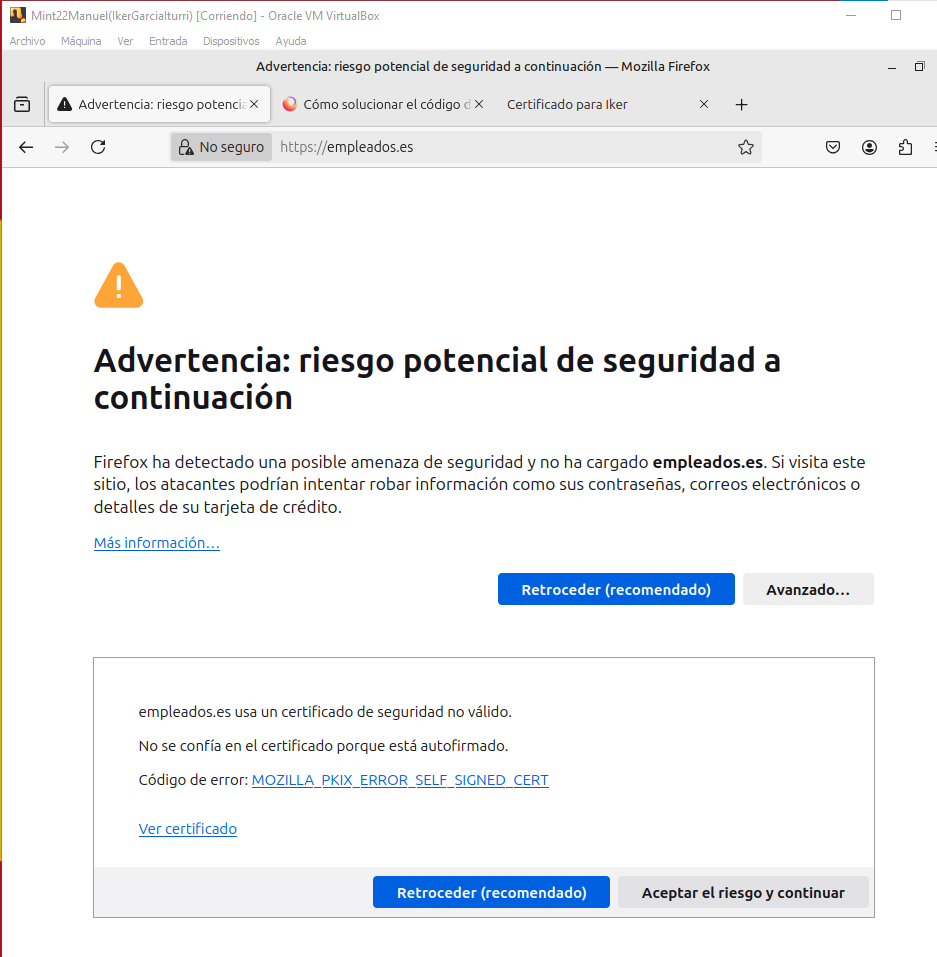


1. ¿Qué aparece al intentar acceder con https? ¿Por qué sucede?

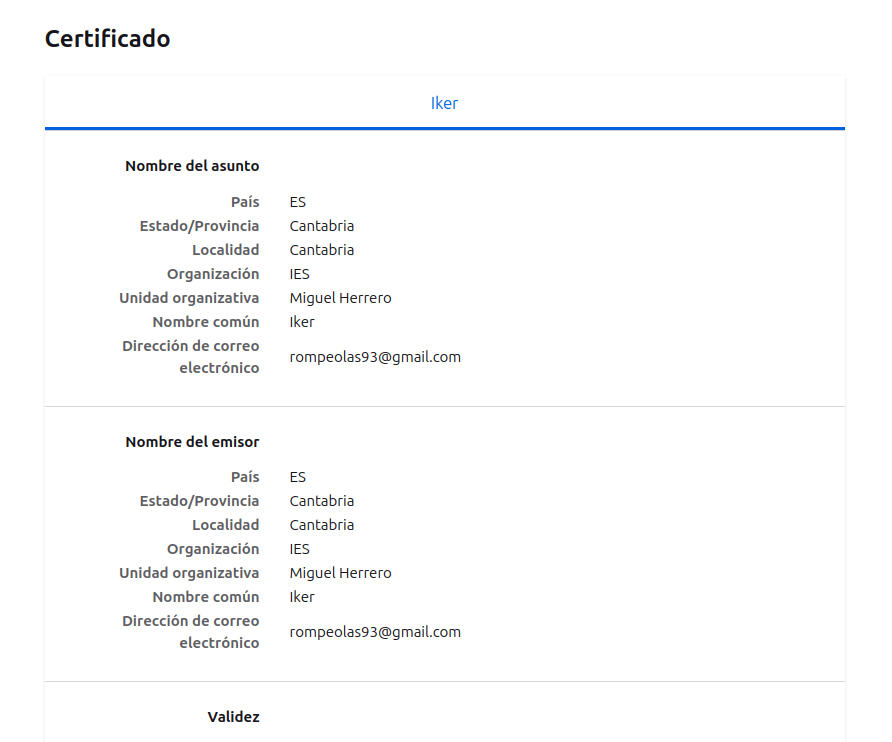


Me dice que hay un riesgo de seguridad y si le doy

avanzado puedo ver el certificado de seguridad, sucede porque este certificado lo hemos creado nosotros y no es válido como tal.

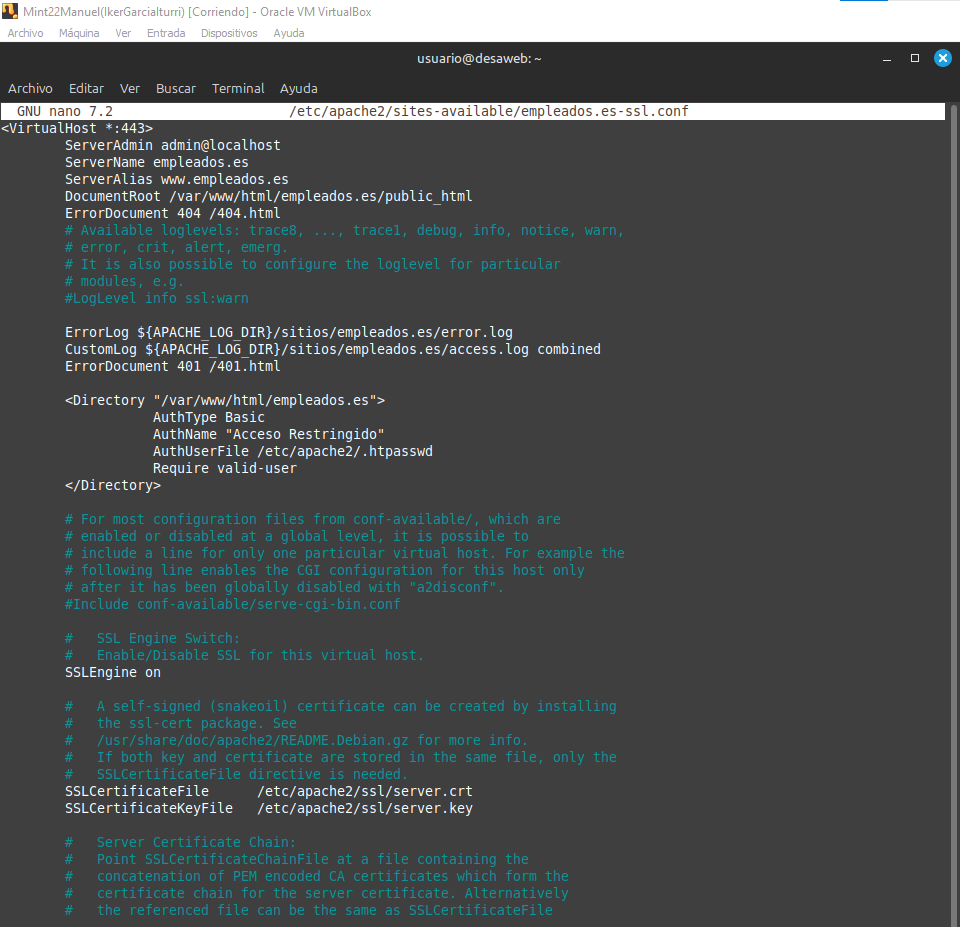


1. Los navegadores muestran información sobre los certificados que están utilizando, seleccionando el candado que aparece a la izquierda de la URL. Muestra la información de tu certificado.

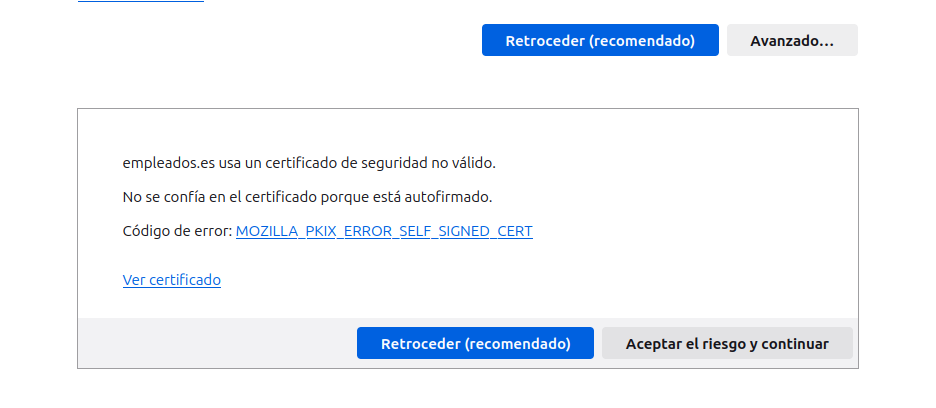


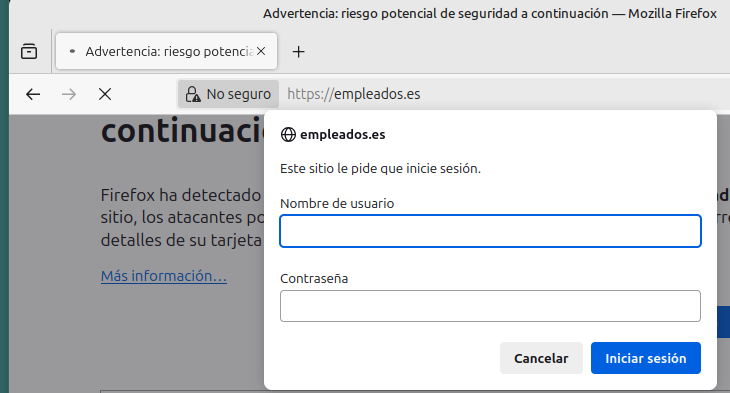
1. ¿Se podría implementar para un host virtual? ¿Qué deberías hacer?

Implementamos el host



Ahora vamos a la página, nos saldrá lo mismo pero esta vez si le damos a Avanzado->Aceptar el riesgo y continuar



Nos mandara a la pagina de empleados.es y tras logearnos podremos acceder a ella

Y ya estaría.

