Seguridad Apache con SSL

Primer Taller CEDIA

3 de Marzo, 2004

Presentado por Hervey Allen
Network Startup Resource Center

Compendio

- Apache corriendo con mod+ssl – que es?
- Certificados digitales con firma y sin firma.
- Como instalar apoyo por ssl en Apache:
  - Compilado desde fuente, o
  - por paquete de RPM
- Ventajas y desventajas de ambos.
- Configuramos un certificado local nuestro
  - Session de configuracion por referencia
- Resolviendo problemas:
  - iptables
  - /var/log/httpd/
  - /var/log/messages
- Resumen

Apache+mod_ssl – Que es?

Juntos Apache y mod_ssl da un sistema de seguridad con certificados digitales que te permite ofrecer conecciones a tu servidor de web en forma encodificado y seguro.

mod_ssl es un modulo de Apache que da soporte al “secure sockets layer” (ssl) y “transport layer security” (tls) entre un servidor de Web y clientes (Web browsers).

Certificados digitales y firmas

Si generas un certificado digital local se puede pagar para que un autoridad verifica tu certificado y te lo manda de vuelta con su firma.

Con la firma del autoridad tu certificado puede estar aceptado con los clientes de Web (Web browsers) sin mostrar el mensaje a los usuarios si deberian confiar en tu certificado o no.

El certificado digitalmente firmado impliqua confianza a los clientes que conecta a tu sitio que tu eres quien tu dices.

Instalando apoyo por SSL con Apache

Con Red Hat 9 ya esta listo el Apache con SSL si eliges de instalar Apache al principio.

Red Hat 9 usa un paquete de RPM por mod_ssl que se llama mod_ssl-2.0.xxxx.rpm.

El paquete generar y instala los siguiente:

- Certificados digitales locales en /etc/httpd/conf.
- El modulo por mod_ssl en /etc/httpd/modules.
- Archivo de configuracion /etc/httpd/conf.d/ssl.conf.

Instalando apoyo al SSL cont.

Otra forma de instalar y compilar Apache con mod_ssl juntos desde fuente.

Se bajaria el codigo desde:

- http://www.apache.org/
- http://www.modssl.org/

Y, puedes especificar un monton de opciones que el paquete de Red Hat no le permite (ya, han tomado las opciones para Ud.).

Ventajas y Desventajas

Paquete de RPM

- Facil, facil, facil.
- Configuracion (que puede ser complicado) ya esta hecho.
- Hacer una actualizacion en el futuro es mucho mas facil.
- Supuestamente la gente en Red Hat tiene mucha experiencia con SSL...?

Ventajas y desventajas cont.

Ventajas de Compilar desde Fuente:

- Puedes especificar *exactamente* como quieres instalado apoyo de SSL en Apache.
- Aprendes mas sobre este servicio.
- Que mas?

Configuramos un certificado local

- Haz una mirada en /etc/httpd/conf.d/ssl.conf.
- Vea la seccion de servidor virtual. Linea 90 en el archivo (en vi “:90”).
- Las primeras lineas y las lineas que punta a los archivos de certificado son mas interesante:
  - /etc/httpd/conf/ssl.crt/server.crt
  - /etc/httpd/conf/ssl.key/server.key

Configurando un certificado cont.

- Red Hat pone los componientes de un certificado en lugares un poco fuera el comun, pero este configuracion permite (mas facilmente) correr mas sitios virtuales con certificados.
- Ahora vamos a generar un certificado. Primero hacemos un respaldo del corriente configuracion de Apache por ser caso:
  - mkdir /tmp/apache
  - cp -r /etc/httpd/conf/* /tmp/apache/.

Configurando un certificado cont.

Haz los siguientes pasos:

- `mkdir /etc/httpd/conf/tmp`
- `cd /etc/httpd/conf/tmp`
- `openssl genrsa -des3 -out server.key 2048`
- `openssl rsa -in server.key -out server.pem`
- `openssl req -new -key server.key -out \`
  `server.csr`
- `openssl x509 -req -days 60 -in server.csr \`
  `-signkey server.key -out server.crt`

Configurando un certificado cont.

Explicacion

```
openssl genrsa -des3 -out server.key 2048
```

para generar una llave de RSA de 1024 bits usando las bibliotecas de OpenSSL. Este llave esta encodificado usando el algoritmo de des3 (triple des).

Este llave es privado.

Configurando un certificado cont.

```
openssl rsa -in server.key -out server.pem
```

Como se lo saca la contraseña de la llava privada.

Puede usar el archivo “server.pem” en vez de “server.key” en el futuro.

Vamos a mostrar esto un poco mas adelante.

Configurando un certificado cont.

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

Generar un csr para que se puede tener la llave firmado o para generar un certificado auto-firmado.

openssl x509 -req -days 60 -in server.csr -signkey server.key -out server.crt

Generar un certificado de corto plazo. Puede ser si era a pedir un certificado firmado desde un autoridad pero necesita algo por mientras.

Una session de configuracion

Ahora mostramos cada paso con las respuesta que puede usar:

Primer paso

```
[root@localhost tmp]# openssl genrsa -des3 -out server.key 2048
Generating RSA private key, 1024 bit long modulus
.....++++++
.............++++++
e is 65537 (0x10001)
Enter pass phrase for server.key:                [contraseña va aqui]
Verifying - Enter pass phrase for server.key:    [contraseña va aqui]
[root@localhost tmp]#
```

Una session de configuracion cont.

Segundo paso:

```
[root@localhost tmp]# openssl rsa -in server.key -out server.pem
Enter pass phrase for server.key:        [contraseña de antes va aqui]
writing RSA key
[root@localhost tmp]#
```

Una session de configuracion cont.

Tercer paso:

[root@localhost tmp]# openssl req -new -key server.key -out server.csr
Enter pass phrase for server.key: [contraseña de antes va aqui]
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) [GB]:EC
State or Province Name (full name) [Berkshire]:Quito
Locality Name (eg, city) [Newbury]:Sangolqui
Organization Name (eg, company) [My Company Ltd]:CEDIA
Organizational Unit Name (eg, section) []:ESPE
Common Name (eg, your name or your server's hostname) []:Escuela del Ejercito
Email Address []:root@localhost

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []: [blanco]
An optional company name []: [blanco]
[root@localhost tmp]#

Una session de configuracion cont.

Cuarto paso:

```
[root@localhost tmp]# openssl x509 -req -days 60 -in server.csr -signkey server.key -out server.crt
Signature ok
subject=/C=EC/ST=Quito/L=Sangoquil/O=CEDIA/OU=ESPE/CN=Escuela de
Ejercito/emailAddress=root@localhost
Getting Private key
Enter pass phrase for server.key:          [contraseña de antes va aqui]
[root@localhost tmp]#
```

Instalar el certificado

Vaya a /etc/httpd/conf/ y haz lo siguiente:

- `cd /etc/httpd/conf`
- `cp tmp/server.crt ssl.crt/.`
- `cp tmp/server.key ssl.key/.`
- `cp tmp/server.csr ssl.csr/.`
- `service httpd stop`
- `service httpd start`

Instalar el certificado

Cuando el servidor de Apache pide una contraseña entra la contraseña que eligiste por tus llaves.

Ahora haz una mirada en /var/log/messages. Si tuviste una problema probablemente un mensaje acerca de ella estara en /var/log/messages.

Ahora, trata de abrir la pagina:

https://localhost/

Anota “https”. Que pasa? Examina el certficado.

Sacar la contraseña

Probablemente fijaste que ahora Apache pide una contraseña para inicializar. Desafortunadamente esto probablemente no va a funcionar en un ambiente de un servidor.

Para sacar la contraseña usa el archivo server.pem. Esto es igual a server.key pero no esta encodificado. Para hacer esto haz lo siguiente en /etc/httpd/conf:

```
cp tmp/server.pem ssl.key/server.key
```

Sacar la contraseña

Y, ahora reinicializar el servicio de Apache.

```
service httpd restart
```

Esta vez no deberias recibir ningun pedido por una contraseña para inicializar el servidor de Apache.

Vaya a https://localhost/ de nuveo y examinar el certificado. Repite el proceso si tienes que cambiar algo en el certificado.

Vaya a 192.188.58.nn de tu vecino y vea si puedes ver su pagina de principio y certificado.

Resolviendo problemas

Si no se puede conectar al servidor vea lo siguiente:

- Si iptables esta corriendo y bloqueando acceso al puerto 443.
- Si esta bien generado el certificado.
- Si la configuracion /etc/httpd/conf.d/ssl.conf esta bien hecho.
- Para ver errores de certificado y/o archivos de configuracion mira en: -->

Resolviendo problemas cont.

Ver errores en:

- /var/log/message (tail -f /var/log/messages)
- /var/log/httpd/error_log
- /var/log/httpd/ssl_error_log

Y, como siempre:

http://www.google.com/

o

http://www.google.com/linux

Mas Recursos

- http://www.modssl.org/
- http://www.apache.org/
- http://www.openssl.org/
- http://www.ws.afnog.org/
- http://www.oreilly.com/ y vea los libros que se trata con SSL.

En Resumen

La instalacion de mod_ssl con Apache te permite correr un servidor de Web seguro.

Si corres webmail esto es esencial a tu seguridad y la seguridad de tus clientes.

Apache con mod_ssl = https. Entonces, esto es una carga extra en tu servidor. Si tienes muchos clientes de Webmail planifica por esto.

Se puede conseguir un certificado firmado, hoy en dia, por no tanto dinero. Revisamos donde en tu Web Browser ahora.

Sin tener un certificado firmado hay un problema fundamental de confiar en tu(s) servidores.