Создание самоподписного SSL-сертификата

Опубликовано muff в Пт, 2009-12-25 02:16

Собственно в эту заметку буду собирать инфу о сертификатах. Покупать доверенные сертификаты на все хосты как-то не судьба (особенно для обслуживания небольшого офиса на 5-10 рабочих мест), поэтому генерировать их будем сами.

Все сертификаты на сервере будут размещаться в одной директории, соответственно необходимо создать ее:

```
# mkdir /etc/ssl/certs
```

ВАЖНО: При создании сертификатов обращаем внимание на поле **"Common Name"** - сюда необходимо вписать действительное DNS-имя сервера (**FQDN**), иначе сертификат не будет приниматься, в связи с недоверием к этому сертификату!

Exim

Почтовики, по определению, настраиваю с поддержкой **SSL**, соответственно необходим сертификат. В конфигурационном файле **exim**, например, за сертификат отвечают следующие строки:

```
tls_certificate = /etc/ssl/certs/mail.pem
tls_privatekey = /etc/ssl/certs/mail.pem
```

Перейдем в каталог сертификатов:

```
# cd /etc/ssl/certs
```

Сгенерируем сертификат. В процессе нужно будет ответить на несколько вопросов. Но ведь это нас не пугает?

```
# openssl req -new -x509 -days 3653 -nodes -out /etc/ssl/certs/mail.pem -keyout
/etc/ssl/certs/mail.pem

Generating a 1024 bit RSA private key
........++++++
.........................++++++
writing new private key to '/etc/ssl/certs/mail.pem'
-----
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]:UA
State or Province Name (full name) [Some-State]:Kiev Region
Locality Name (eg, city) []:Kiev
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Somebody Ltd.
Organizational Unit Name (eg, section) []:IT Department
```

```
Common Name (eg, YOUR name) []:mail.domain.com
Email Address []:username [at] domain [dot] com
```

Apache

Довольно часто необходимо на web-сервере организовать работу через **https**, а для этого соответственно нужно настроить поддержку **SSL** для **Apache**.

Перейдем в каталог сертификатов:

```
# cd /etc/ssl/certs
```

Генерируем сертификат (пароль вводим несложный, учитывая то, что скоро мы от него откажемся):

```
# openssl genrsa -out apache.key -rand randfile -des3 2048

0 semi-random bytes loaded
Generating RSA private key, 2048 bit long modulus
...........+++
.....................................................+++
e is 65537 (0x10001)
Enter pass phrase for apache.key:
Verifying - Enter pass phrase for apache.key:

# openssl req -new -x509 -key apache.key -out apache.crt -days 3653
Enter pass phrase for 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]:UA
State or Province Name (full name) [Some-State]:Kiev Region
Locality Name (eg, city) []:Kiev
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Local Network
Organizational Unit Name (eg, section) []:IT Department
Common Name (eg, YOUR name) []:web.domain.com
Email Address []:username [at] domain [dot] com
```

Для избежания проблем в дальнейшем, значение "**Common Name**" необходимо "привязать" к имени домена.

Избавляемся от пароля в сертификате:

```
# openssl rsa -in apache.key -out apache.key
Enter pass phrase for apache.key:
writing RSA key
```

Обезопасим систему:

```
# chmod 400 apache.key
```

На этом действия с сертификатом заканчиваются. Следующий шаг - добавить поддержку созданного сертификата в конфигурационный файл **Apache**. Достигается это внесением следующих строк в **httpd.conf**:

```
Listen 443

NameVirtualHost *:443



<VirtualHost *:443>
        ServerName domain.com
        ServerAdmin hostmasterhostmaster [at] domain [dot] com (@domain.com)

        DocumentRoot /usr/local/www/apache22/data

            ErrorLog /var/log/apache/domain.com-error.log
            TransferLog /var/log/apache/domain.com-access.log

SSLEngine on
SSLCipherSuite ALL:!ADH:!EXPORT56:RC4+RSA:+HIGH:+MEDIUM:+LOW:+SSLv2:+EXP:+eNULL
SSLCertificateFile "/etc/ssl/certs/apache.crt"
SSLCertificateKeyFile "/etc/ssl/certs/apache.key"

    <Directory "/usr/local/www/apache22/data">
            DirectoryIndex index.php
            AllowOverride All
            SSLRequireSSL
            Order Deny,Allow
            Allow from all
     </Directory>

</VirtualHost>
```

Проверяем конфигурацию Apache, и если все в порядке, перезапускаем службу:

```
# apachectl configtest
Syntax OK
# apachectl graceful
```

Теперь можно подключаться к серверу по протоколу **HTTPS**.

Источник (получено *2026-11-17 00:05*):
http://muff.kiev.ua/content/sozdanie-samopodpisnogo-ssl-sertifikata