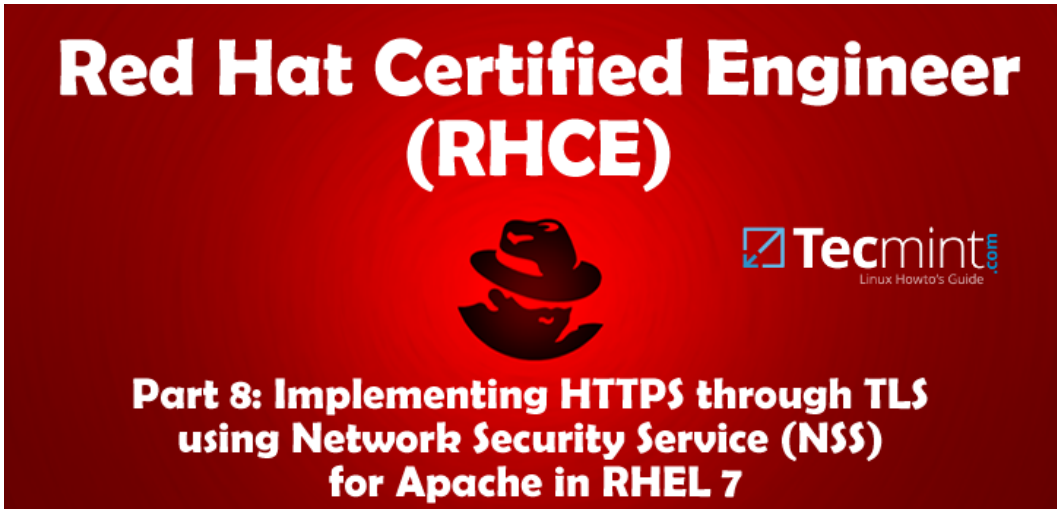
RHCE 系列（八）：在 Apache 上使用网络安全服务（NSS）实现 HTTPS

2015-12-6 09:00 评论: 1

参考原文：http://www.tecmint.com/create-... 作者： Gabriel Cánepa
编译文章：LCTT https://linux.cn/article-6696-1.html 译者： ictlyh

如果你是一个负责维护和确保 web 服务器安全的系统管理员，你需要花费最大的精力确保服务器中处理和通过的数据任何时候都受到保护。



RHCE 系列：第八部分 - 使用网络安全服务（NSS）为 Apache 通过 TLS 实现 HTTPS

为了在客户端和服务器之间提供更安全的连接，作为 HTTP 和 SSL（安全套接层 Secure Sockets Layer）或者最近称为 TLS（传输层安全 Transport Layer Security）的组合，产生了 HTTPS 协议。

由于一些严重的安全漏洞，SSL 已经被更健壮的 TLS 替代。由于这个原因，在这篇文章中我们会解析如何通过 TLS 实现你 web 服务器和客户端之间的安全连接。

这里假设你已经安装并配置好了 Apache web 服务器。如果还没有，在进入下一步之前请阅读下面站点中的文章。

- 在 RHEL/CentOS 7 上安装 LAMP（Linux，MySQL/MariaDB，Apache 和 PHP）<https://linux.cn/article-5789-1.html>

安装 OpenSSL 和一些工具包

首先，确保正在运行 Apache 并且允许 http 和 https 通过防火墙：

```
# systemctl start http
# systemctl enable http
# firewall-cmd --permanent –-add-service=http
# firewall-cmd --permanent –-add-service=https
```

然后安装一些必需的软件包：

```
# yum update && yum install openssl mod_nss crypto-utils
```

重要：请注意如果你想使用 OpenSSL 库而不是 NSS（ Network Security Service ）实现 TLS，你可以在上面的命令中用 mod_ssl 替换 mod_nss（使用哪一个取决于你，但在这篇文章中我们会使用 NSS，因为它更加安全，比如说，它支持最新的加密标准，比如 PKCS #11）。

如果你使用 mod_nss，首先要卸载 mod_ssl，反之如此。

```
# yum remove mod_ssl
```

配置 NSS（网络安全服务）

安装完 mod_nss 之后，会创建默认的配置文件 /etc/httpd/conf.d/nss.conf 。你应该确保所有 Listen 和 VirualHost 指令都指向 443 号端口（HTTPS 默认端口）：

nss.conf – 配置文件

```
Listen 443
VirtualHost _default_:443
```

然后重启 Apache 并检查是否加载了 mod_nss 模块：

```
# apachectl restart
# httpd -M | grep nss
```

检查 Apache 是否加载 mod_nss 模块

下一步，在 /etc/httpd/conf.d/nss.conf 配置文件中做以下更改：

1、 指定 NSS 数据库目录。你可以使用默认的目录或者新建一个。本文中我们使用默认的：

```
NSSCertificateDatabase /etc/httpd/alias
```

2、 通过保存密码到数据库目录中的 /etc/httpd/nss-db-password.conf 文件来避免每次系统启动时要手动输入密码：

```
NSSPassPhraseDialog file:/etc/httpd/nss-db-password.conf
```

其中 /etc/httpd/nss-db-password.conf 只包含以下一行，其中 mypassword 是后面你为 NSS 数据库设置的密码：

```
internal:mypassword
```

另外，要设置该文件的权限和属主为 0640 和 root:apache：

```
# chmod 640 /etc/httpd/nss-db-password.conf
# chgrp apache /etc/httpd/nss-db-password.conf
```

3、 由于 POODLE SSLv3 漏洞，红帽建议停用 SSL 和 TLSv1.0 之前所有版本的 TLS（更多信息可以查看这里 <https://access.redhat.com/articles/1232123> ）。

确保 NSSProtocol 指令的每个实例都类似下面一样（如果你没有托管其它虚拟主机，很可能只有一条）：

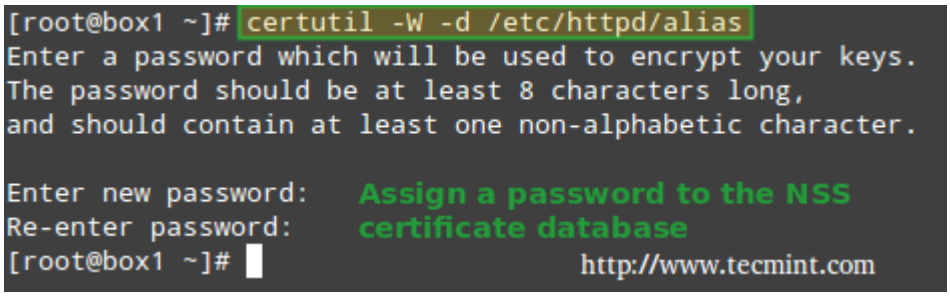
```
NSSProtocol TLSv1.0,TLSv1.1
```

4、 由于这是一个自签名证书，Apache 会拒绝重启，并不会识别为有效发行人。由于这个原因，对于这种特殊情况我们还需要添加：

```
NSSEnforceValidCerts off
```

5、 虽然并不是严格要求，为 NSS 数据库设置一个密码同样很重要：

```
# certutil -W -d /etc/httpd/alias
```



为 NSS 数据库设置密码

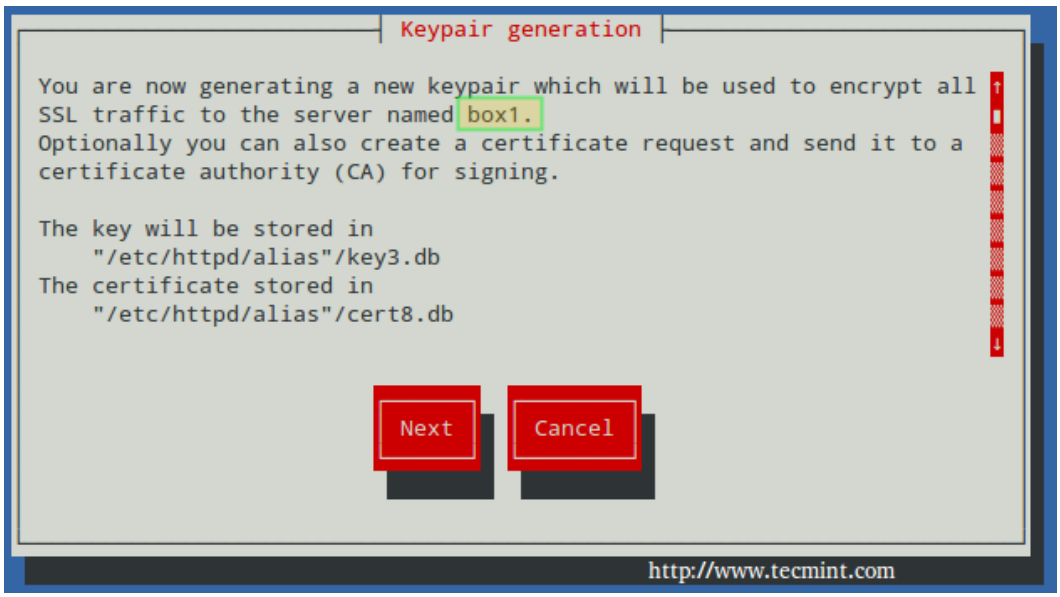
创建一个 Apache SSL 自签名证书

下一步，我们会创建一个自签名证书来让我们的客户机可以识别服务器（请注意这个方法对于生产环境并不是最好的选择；对于生产环境你应该考虑购买第三方可信证书机构验证的证书，例如 DigiCert）。

我们用 genkey 命令为 box1 创建有效期为 365 天的 NSS 兼容证书。完成这一步后：

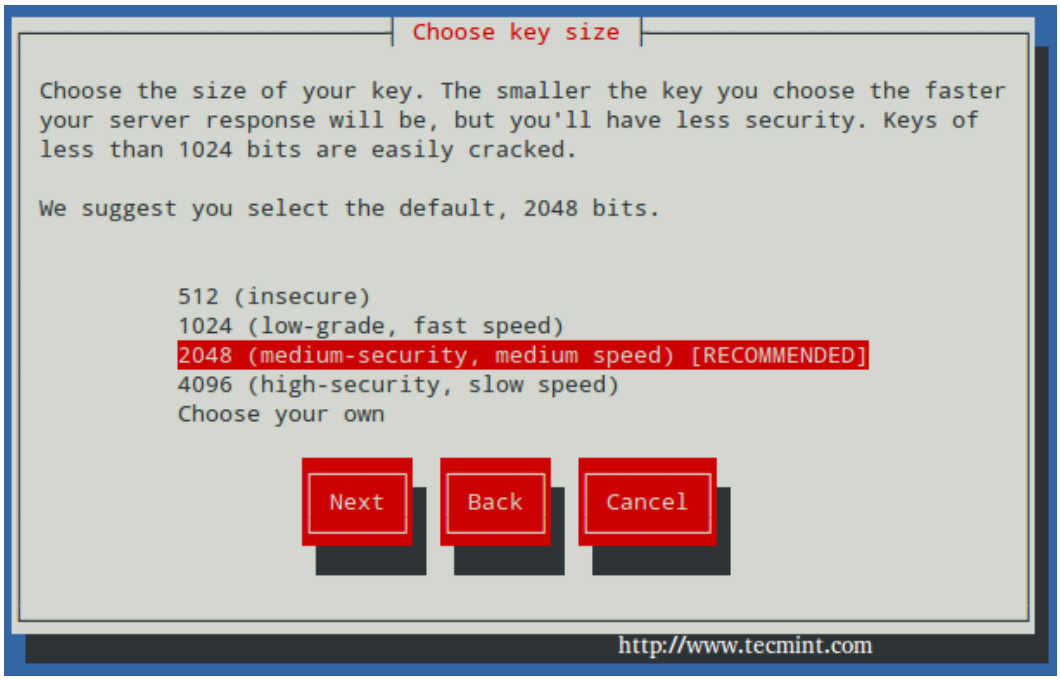
```
# genkey --nss --days 365 box1
```

选择 Next：



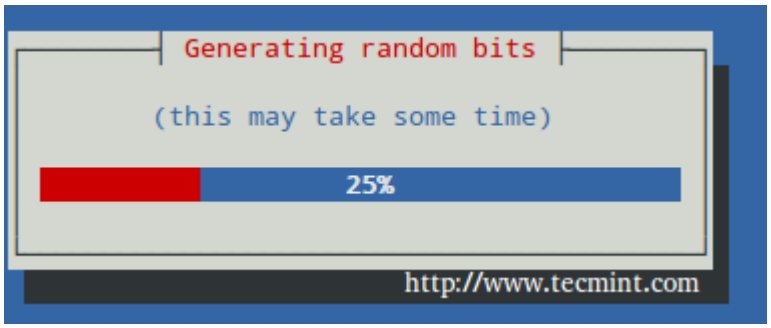
创建 Apache SSL 密钥

你可以使用默认的密钥大小（2048），然后再次选择 Next：



选择 Apache SSL 密钥大小

等待系统生成随机比特：



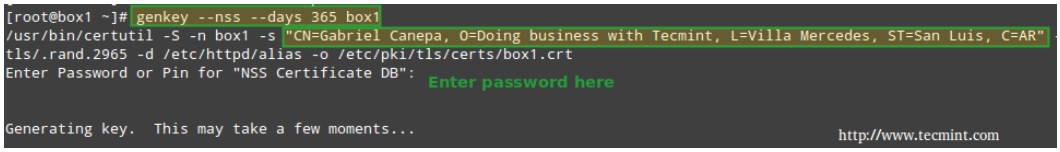
生成随机密钥比特

为了加快速度，会提示你在控制台输入随机字符，正如下面的截图所示。请注意当没有从键盘接收到输入时进度条是如何停止的。然后，会让你选择：

1. 是否发送验证签名请求（CSR）到一个验证机构（CA）：选择 No，因为这是一个自签名证书。
2. 为证书输入信息。

最后，会提示你输入之前给 NSS 证书设置的密码：

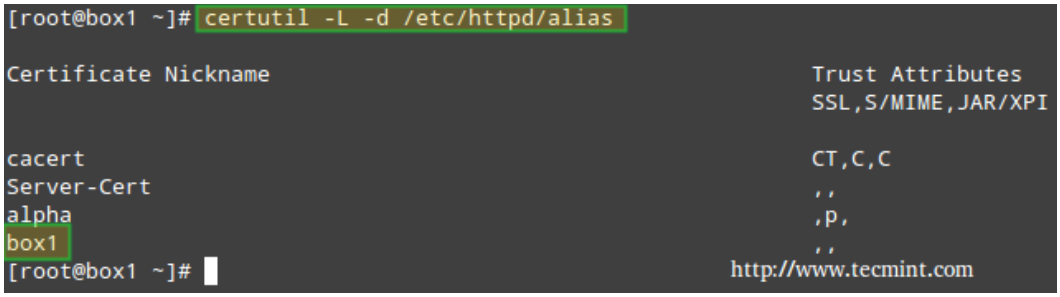
```
# genkey --nss --days 365 box1
```



Apache NSS 证书密码

需要的话，你可以用以下命令列出现有的证书：

```
# certutil –L –d /etc/httpd/alias
```



列出 Apache NSS 证书

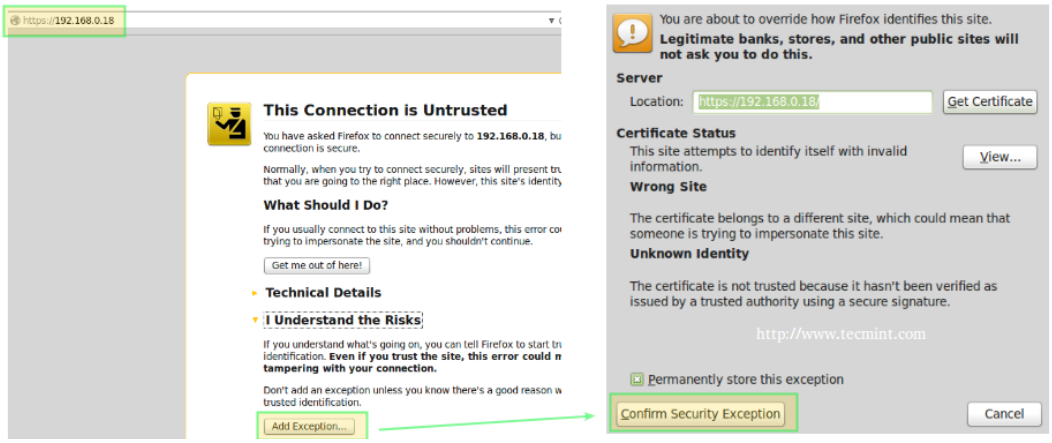
然后通过名字删除（如果你真的需要删除的，用你自己的证书名称替换 box1）：

```
# certutil -d /etc/httpd/alias -D -n "box1"
```

如果你需要继续进行的话，请继续阅读。

测试 Apache SSL HTTPS 连接

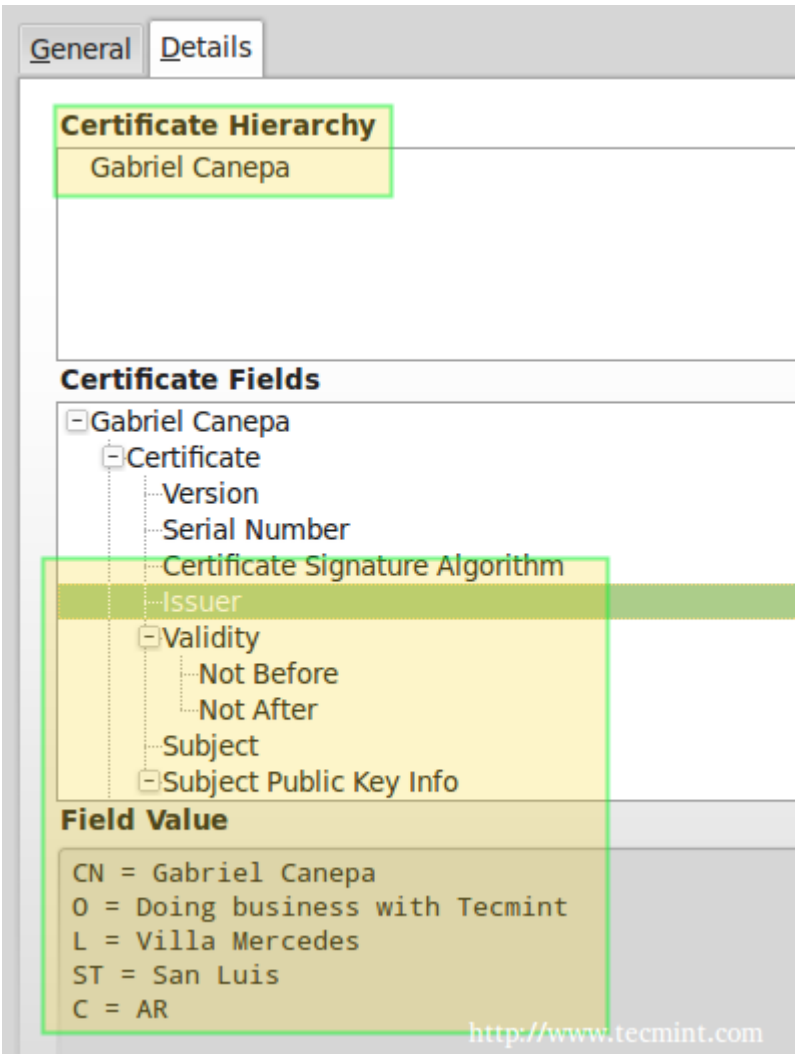
最后，是时候测试到我们服务器的安全连接了。当你用浏览器打开 https://<web 服务器 IP 或主机名>，你会看到著名的信息 “This connection is untrusted”：



检查 Apache SSL 连接

在上面的情况中，你可以点击 *Add* 添加例外 *Exception* 然后 *Confirm Security Exception* 确认安全例外 - 但先不要这么做。让我们首先来看看证书看它的信息是否和我们之前输入的相符（如截图所示）。

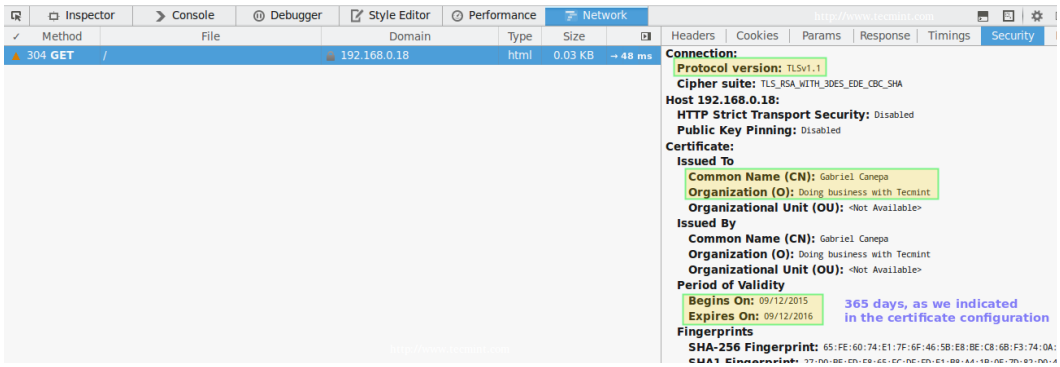
要做到这点，点击上面的 *View...* 视图 -> *Details* 详情 选项卡，当你从列表中选择发行人你应该看到这个：



确认 Apache SSL 证书详情

现在你可以继续，确认例外（限于此次或永久），然后会通过 https 把你带到你 web 服务器的 DocumentRoot 目录，在这里你可以使用你浏览器自带的开发者工具检查连接详情：

在火狐浏览器中，你可以通过在屏幕中右击，然后从上下文菜单中选择 *Inspect Element* 检查元素 启动开发者工具，尤其要看“ *Network* 网络 ”选项卡：



检查 Apache HTTPS 连接

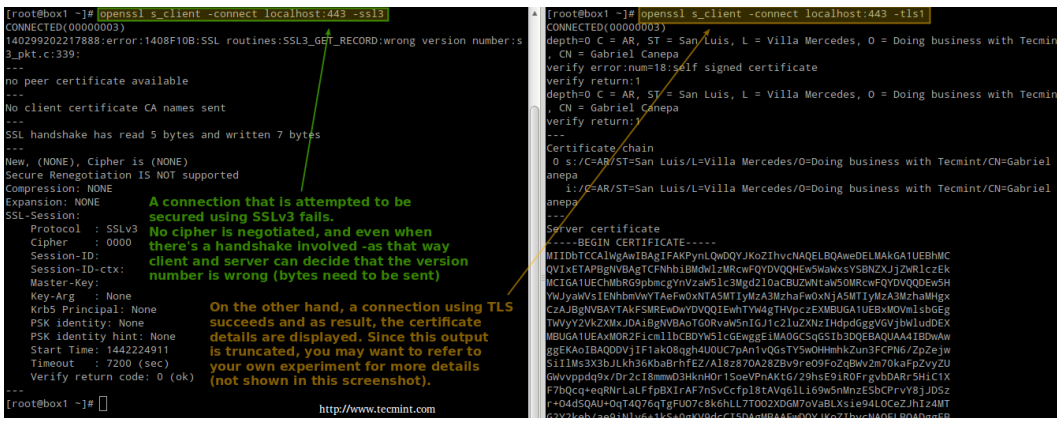
请注意这和之前显示的在验证过程中输入的信息一致。还有一种方式通过使用命令行工具测试连接：

左图（测试 SSLv3）：

```
# openssl s_client -connect localhost:443 -ssl3
```

右图（测试 TLS）：

```
# openssl s_client -connect localhost:443 -tls1
```



测试 Apache SSL 和 TLS 连接

参考上面的截图了解更详细信息。

总结

我想你已经知道，使用 HTTPS 会增加会在你站点中输入个人信息的访客的信任（从用户名和密码到任何商业/银行账户信息）。

在那种情况下，你会希望获得由可信验证机构签名的证书，正如我们之前解释的（步骤和设置需要启用例外的证书的步骤相同，发送 CSR 到 CA 然后获得返回的签名证书）；否则，就像我们的例子中一样使用自签名证书即可。

要获取更多关于使用 NSS 的详情，可以参考关于 mod-nss <https://git.fedorahosted.org/cgit/mod_nss.git/plain/docs/mod_nss.html> 的在线帮助。如果你有任何疑问或评论，请告诉我们。

---

via: http://www.tecmint.com/create-apache-https-self-signed-certificate-using-nss/ <http://www.tecmint.com/create-apache-https-self-signed-certificate-using-nss/>

作者：Gabriel Cánepa <http://www.tecmint.com/author/gacanepa/> 译者：ictlyh <http://www.mutouxiaogui.cn/blog/> 校对：wxy <https://github.com/wxy>

本文由 LCTT <https://github.com/LCTT/TranslateProject> 原创编译，Linux中国 <https://linux.cn/> 荣誉推出