HOWTO

Configure Nginx for SSL with DoD CAC Authentication on CentOS 6.3

Joshua Penton
Geocent, LLC
joshua.penton@geocent.com

March 2013

Table of Contents

Overview

The expansion of web presence within the Department of Defense (DoD) is requiring more systems to provide a web-based interface to system information and resources. While many technologies comprise the stack from system resources to end user the one component that lies at the most exposed end point is that of the web server. Historically, this functionality has been delivered by monolithic applications such as Apache HTTP Server[1] or Microsoft Internet Information Services[2]. However, as the need for web servers shift to require software that is both easier to deploy and manage while still providing necessary levels of performance and configurability it behooves administrators to look towards emerging solutions.

Nginx[3] has emerged as a high performance solution that has gained wide adoption within the commercial sector. The open source project provides a web server and a reverse proxy with design principles centered on high concurrency and low memory usage coupled with a scalable module-based architecture. The end result is an easily configurable and cross-platform web server that has the ability to surpass the performance of traditional applications while both standardizing and easing configuration requirements.

Within the DoD a common requirement is for applications to challenge incoming requests for Common Access Card (CAC) credential for user identification and authorization. As a result any web server that seeks to provide functionality within the DoD infrastructure is required to support this functionality. This document provides the necessary steps to configure Nginx to enable request authorization based off of CAC credentials. While the target environment is that of CentOS 6.3 the instructions are applicable to additional platforms.

---

[1] "Welcome! - The Apache HTTP Server Project" - http://httpd.apache.org/

[2] "Home : The Official Microsoft IIS Site" - http://www.iis.net/

[3] "nginx news" - http://nginx.org/

Prerequisites

Install OpenSSL

If OpenSSL[4] is not already installed on the target system use `yum`[5] to install the necessary packages:

```
$ yum install openssl
```

Install Nginx

If Nginx is not already installed on the system use yum to install the necessary packages. First add the Nginx yum repository by creating a file named `/etc/yum.repos.d/nginx.repo` with the following contents:

```
[nginx]
name=nginx repo
baseurl=http://nginx.org/packages/centos/$releasever/$basearch/
gpgcheck=0
enabled=1
```

Save the contents of the file and then initiate the yum installation to install the Nginx packages:

```
$ yum install nginx
```

Generate SSL Certificates

If the target system does not already have the required SSL certificates generated and installed they can be generated at this time. First create the target directory to which the server key and certificate will be installed and set proper permissions:

```
$ mkdir /etc/nginx/ssl
$ chmod 700 /etc/nginx/ssl
```

Next generate the private key for the server:

```
$ cd /etc/nginx/ssl
$ openssl genrsa -des3 -out server.key.org 1024
$ openssl rsa -in server.key.org -out server.key
```

Next create a Certificate Signing Request (CSR):

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

If the server's certificate must be signed by a central signing authority submit the `server.csr` file to proper administrators and copy the returned certificate to

---

[4] "OpenSSL: The Open Source toolkit for SSL/TLS" - http://www.openssl.org/
[5] "Yum Package Manager - Trac" - http://yum.baseurl.org/

`/etc/nginx/ssl/server.crt`. Otherwise for development purposes a self-signed certificate can be generated and installed:

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

Finally ensure all of the installed files have proper permissions set:

```
$ chmod 600 /etc/nginx/ssl/*
```

Install DoD Root CA Certificates

The DoD currently hosts both its CA and ECA certificates at http://dodpki.c3pki.chamb.disa.mil/rootca.html and will need to be installed into the OpenSSL certificate store.

Download Root CA Certificates

The most recent certificates may be downloaded to the target server using `wget`:

```
$ wget http://dodpki.c3pki.chamb.disa.mil/rel3_dodroot_2048.p7b
$ wget http://dodpki.c3pki.chamb.disa.mil/dodeca.p7b
$ wget http://dodpki.c3pki.chamb.disa.mil/dodeca2.p7b
```

Convert DoD Root CA Certificates

The certificates downloaded from the DISA website are in PKCS#7 format but will need to be in a format recognizable by Nginx. OpenSSL can be used for converting the certificates to the Privacy Enhanced Mail (PEM) format usable by a wide variety of software including Nginx:

```
$ openssl pkcs7 -inform DER -outform PEM -in
rel3_dodroot_2048.p7b -out rel3_dodroot_2048.pem -print_certs
$ openssl pkcs7 -inform DER -outform PEM -in dodeca.p7b -out
dodeca.pem -print_certs
$ openssl pkcs7 -inform DER -outform PEM -in dodeca2.p7b -out
dodeca2.pem -print_certs
```

For deployment purposes with Nginx the root certificate file must be contained within a single certificate file. This is accomplished using the `cat` utility:

```
$ cat rel3_dodroot_2048.pem dodeca.pem dodeca2.pem > dod-root-
certs.pem
```

Install Converted Certificates

The converted certificates must be installed to the SSL module:

```
$ cp rel3_dodroot_2048.pem /etc/ssl/certs
$ cp dodeca.pem /etc/ssl/certs
$ cp dodeca2.pem /etc/ssl/certs
$ cp dod-root-certs.pem /etc/ssl/certs
```

Configure Nginx

Within the `/etc/nginx/conf.d` directory either create a new configuration file or modify a relevant file that is currently in use. To the configuration file add the following information:

```
listen 443;
ssl on;
ssl_certificate ssl/server.crt;
ssl_certificate_key ssl/server.key;
ssl_verify_client on;
ssl_verify_depth 2;
ssl_client_certificate /etc/ssl/certs/dod-root-certs.pem;
```

Restart the Nginx process and access your site.

```
$ service nginx restart
```

Upon connection Nginx will prompt the user for their DoD certificate.