Security Testing your Apache Configuration with Nikto

By xenlab
Created 2006-08-11 07:44

Security Testing your Apache Configuration with Nikto

Introduction

By now you've got the perfect setup for your new Ubuntu 6.0.6 (Dapper Drake) box. You may have even followed the excellent Intrusion Detection and Prevention with BASE and Snort tutorial. And as an added precaution you installed DenyHosts to prevent hack attempts via ssh. But now that you've got your new LAMP server on the internet, how can you tell that your new web server is secure? You test it, of course!

This tutorial, inspired by one of the chapters in Hardening Apache by Tony Mobily (APress), will show you how to set up the free web server security scanner tool, Nikto. This tool will probe your Apache set-up for vulnerabilities, so you can get an idea of what holes may exist in your configuration. This tutorial will only get you so far as installing the tool, and running your first scan. A google search or the afore mentioned book will give you plenty of information on actually securing your Apache server.

Remember, only scan servers you own or that you have permission to scan, or you could easily risk legal action and jail time.

Let's get started.

1.1 Installing Net_SSLeay

Net_SSLeay is a Perl Module that adds the ability to connect over SSL connections. The latest version is 1.30 (as of this writing), and can be downloaded from the CPAN repository. This will be required by Nikto if you plan on testing SSL enabled servers.

I generally create a `/src` directory to download all my source files into, and will be doing that first.

```
mkdir /src
cd /src
```

Now we can download the Net_SSLeay Perl Module source:

```
wget http://search.cpan.org/CPAN/authors/id/F/FL/FLORA/Net_SSLeay.pm-1.30.tar.gz
```

Once it finishes downloading, let's extract it and enter the unarchived folder:

```
tar -xzvf Net_SSLeay.pm-1.30.tar.gz
cd ./Net_SSLeay.pm-1.30
```

Now, Let's install this module with a few simple commands:

```
perl Makefile.PL
make
make install
```

1.2 Installing Nikto

First we download the latest version of Nikto. This can be retrieved from the web site of the security experts that wrote the software at CIRT.net.

Go back to the `/src` directory:

```
cd /src
```

And now get the Nikto software (current version 1.35, but the link below should always download the latest stable release), unarchive it:

```
wget http://www.cirt.net/nikto/nikto-current.tar.gz
tar -xzvf nikto-current.tar.gz
```

Nikto is built on top of rfp's LibWhisker (for all of it's base network functionality). It's included with Nikto, but let's go ahead and update it to the latest version (of the 1.x branch).

```
wget http://www.wiretrip.net/rfp/libwhisker/LW.pm
cp LW.pm ./nikto-1.35/LW.pm
```

Since Nikto is just a perl script, it doesn't need to be installed, but we should go ahead and move it to a more permanent location such as `/usr/local`

```
mv nikto-1.35/ /usr/local/nikto
```

Now, let's change into this directory so we can update Nikto's database.

```
cd /usr/local/nikto
perl nikto.pl -update
```

1.3 Using Nikto

Now that we're all up to date, let's take it out for a test drive.

The standard test (assuming you've installed Nikto directly on your :

```
perl nikto.pl -h localhost
```

When running this test on a standard installation based on the Perfect Set-Up how-to, I found 5 errors. Nothing too critical, 3 out of date notices (Apache, PHP, OpenSSL) and 2 Apache configuration errors (Manual and Icon directories still accessible, letting potential malicious hackers know that you haven't done much to reconfigure Apache).

If you want to give Snort a run for it's money, you can add the -evasion flag, and have it try to sidestep your IDS systems, like so:

```
perl nikto.pl -h example.com -evasion 1
```

Substitute example.com in the example above with the URL or IP address of your web server. There are 9 different options for the -evasion flag. 1 is for Random URI encoding (non-UTF8). This scan is decidedly slower, so you may want to go make a sandwich. For more information on the available options that Nikto has to offer, study the README file (located in the ... /nikto/docs/nikto_usage.html, or online).

Conclusion

Security is a state of being, not a state to be achieved. By testing your configurations, you can find holes that you may have missed. However, no tool is a path to a secure system, but only a guide. It is highly recommended that you keep educating yourself and subscribe to security alerts from a respected authority on the subject. Only then will you hope to stay ahead of the baddies, and keep you and your server from being compromised.

Happy Scanning!

Source URL:
http://www.howtoforge.com/apache_security_testing_with_nikto