OpenVAS-4

Join the Discussion
Connect

By Russ McRee – ISSA member, Puget Sound (Seattle), USA Chapter

Prerequisites

VirtualBox 3.x, VMWare for the OpenVAS-4 virtual appliance

Pick your platform, over 20 options available[1]

Introduction

Way back when, Internet eons ago (2005), Tenable closed source code for Nessus, the venerable but now proprietary, comprehensive vulnerability scanner. The inevitable fork, initially called GNessUs, became the Open Vulnerability Assessment System, or OpenVAS. First released as version 1.0 in August 2008, the March 17 release of OpenVAS-4 struck me as an ideal opportunity to introduce you to the other "framework of several services and tools offering a comprehensive and powerful vulnerability scanning and vulnerability management solution."

When I reached out to Dr. Jan-Oliver Wagner, OpenVAS project lead and Greenbone CEO (Greenbone is the commercial venture that is the driving force behind OpenVAS), he indicated that much what we need to know about OpenVAS-4 is included in the latest release announcement.[2]

Some highlights include:

- OpenVAS-4 includes the following OpenVAS modules[3]:
  - Libraries 4.0 (aggregated shared functionality)
  - Scanner 3.2 (executes the actual Network Vulnerability Tests (NVTs))
  - Manager 2.0 (central service that consolidates vulnerability scanning solution)
  - Administrator 1.1 (command line tool or as a full-service daemon offering the OpenVAS Administration Protocol (OAP))
  - GSA 2.0 (lean web service offering a user interface for web browsers)
  - GSD 1.1 (Qt-based desktop client)
  - CLI 1.1 (command line tool which allows batch process creation to drive OpenVAS Manager)
- The most significant new features:
  - Report Format Plugin Framework
  - Master-Slave mode
  - Improved Scanner.
- The extended OpenVAS Management Protocol (OMP) 2.0 of OpenVAS Manager makes several new features consistently available to all of its clients (Web, Desktop, CLI).

A related structure diagram is exhibited in Figure 1.

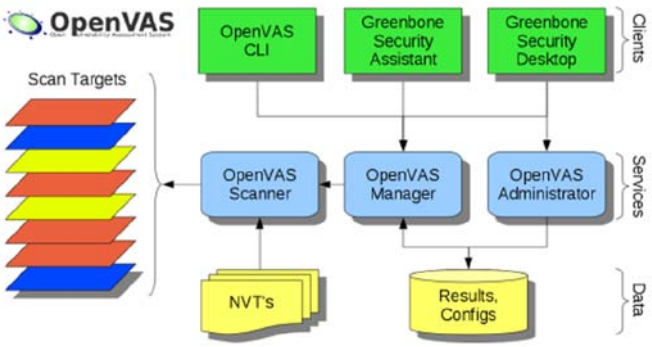

Figure 1 – OpenVAS structure

Jan-Oliver also pointed out that the OpenVAS Developer Conference is taking place in July, in Osnabrück, Germany. Greenbone Networks as mentioned above is the OpenVAS developer and offers commercial solutions based on this open source tool. Additionally, Greenbone is very proud to not run an "Open Core"[4] business or have dual licensing.

Configuring and using OpenVAS-4

I used the OpenVAS-4 Virtual Appliance Community Edition virtual appliance[5] as offered for demonstration purposes. That said, this latest release of OpenVAS is the first to include installation packages for over 20 platforms, several installation quick guides, a tool to check proper setup and, the above mentioned virtual appliance.

1 http://www.openvas.org/install-packages.html.
2 http://www.openvas.org/news_archive.html#openvas4.
3 http://www.openvas.org/about-software.html.
4 http://carlodaffara.conecta.it/?p=104.
5 http://www.openvas.org/vm.html.

Once you've installed the VM (really easy with VMware Workstation, just boot to the ISO and accept defaults), and navigate via your browser to the VM's IP address over HTTPS and port 9392. In my case, https://192.168.122.141:9392 dropped me right to the Greenbone Security Assistant (GSA). Gotta love the Greenbone mascot. ;-)

Figure 2 – Greenbone Security Assistant

Default user is *openvas*; you'll note the necessary users and credentials as the VM completes installation and first boot. As a first step I suggest changing the openvas user password to something you prefer via *Administration => Users*.

The next suggested step is to navigate to *NVT Feed* (Network Vulnerability Tests) under *Administration* and synchronize with the OpenVAS NVT feed[6] as seen in Figure 3.

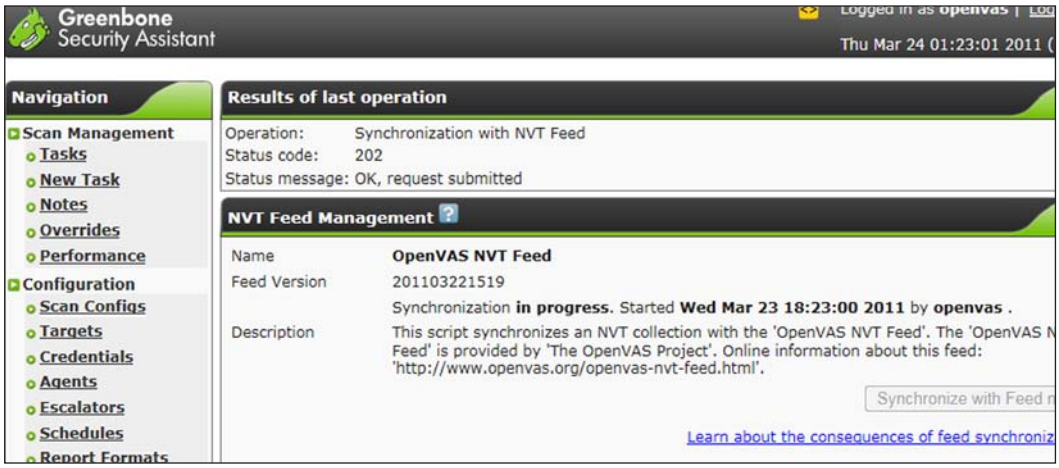

Figure 3 – NVT feed synchronization

As of February 2011, there are more than 20,000 NVTs; commercial users also benefit from daily updates, as well as compliance rulesets and enhanced service level agreements and features.

Once your NVT set is current, move to *Scan Configs* under *Configuration*, and create some specific scans. Here's where I vent my spleen. Nothing makes me more nuts when logs I'm reviewing are full of vulnerability checks that have absolutely no good reason to have been run against the host in question.

Examples: Windows vulnerability tests against a Linux host? Oracle checks against MS-SQL database servers? Solaris vulnerability checks against Windows hosts? PHP checks against IIS 7.5 running ASP.net apps? Yep, I've seen it all. Aaargh! Typically, do not run big, loud, default scans with all checks enabled. You're fragging bandwidth and wasting cycles needlessly.

OpenVAS detects the operating system and some other aspects. Then, automatically, only those tests will run that do make sense. If Port 80 is served by IIS, Apache tests will not be launched, etc. The OpenVAS team believes this is better than burden the user with first thinking about which hosts have to be scanned with which tests and then do all the configurations. "Just start the scan!" they say.

Thus, you can use first scans to map the network and establish baselines for host IP ranges or logical groupings. If you're an all-Windows shop, trim unrelated checks and vice versa if you're a Linux shop. Hybrid shops are tougher, but you should still manage assets well enough to create asset groupings that allow you to scan appropriately. There are some OpenVAS expert settings that include brute-force capabilities, but these are off by default; use them when and where appropriate.

A truly specific OpenVAS scan configuration makes sense to save time, especially when scanning, for example, 1000 Windows hosts for a very specific aspect (e.g. Conficker).

Ok, I've vented a bit, and countered with OpenVAS capabilities; on to some suggested solutions.

Under Scan Configs you'll note preconfigured scans for Full and Fast, Full and Fast Ultimate, Full and Very Deep, Full and Very Deep Ultimate (Figure 4). These imply the use of (or the lack of trust of) previously collected information for future scans, but that doesn't preclude my rant above. Keep the noise down even during a first scan (information collection). Name your new scan config (I used **Win2k3 Pwnzor Edition** as this is the name and OS for the target VM), add comments if you wish, and choose *empty*, *static* and *fast*, then click *Create Scan Config*. You can also import XML-based scan configurations if you are managing multiple platforms. You'll then see your newly created scan config under (you guessed it) the Scan Configs column.

Click the wrench icon and you'll find yourself ready to edit network vulnerability test families as seen in Figure 5. As my VM, aptly named Win2k3 Pwnzor Edition, is an unpatched, poorly configured Windows Server 2003 image, you can follow the developer's logic and run a full scan discovery scan via the latter two of the four preconfigured options. Even if I don't run full sets, I'm more forgiving of choosing whole families, as long as they're even partly relevant. I chose to Select all NVTs for the Windows and Windows: Microsoft Bulletins. I am running IIS and MS-SQL on this image as well,

6 http://www.openvas.org/openvas-nvt-feed.html.

Figure 4 – Win2k3 Pwnzor Edition scan config

so I drilled (click the wrench icon gain for the related family to edit) into the related families but was more selective here. Under *Web Servers* and *FTP* I chose all tests related to IIS then clicked the *Save Config* button. I discovered that there are no MS-SQL tests (or at least none available via the community NVT feed) and consider that an obvious shortcoming. That said, I really like that you can leverage specific Nmap scripting engine elements under the Nmap NSE family. I enabled the IIS WebDAV and SMB related test for giggles.

By default scans with test families configured this way are defined as *Static* where new NVTs will not be added or considered when you synchronize. Be aware that you'll be adding NVTs at some point should you choose *Dynamic*.

Enough scan configuration, let's define a target, found under *Administration => Targets*. Under *New Target*, provide the target name and host(s), then click *Create Target*.

With targets defined, it's time to scan, an activity coordinated via *Scan Management => New Tasks*. Same routine; give the task a name and select your recently created Scan Config and Scan Targets. Some optional additional features to note here: *Schedule* should be obvious, *Escalator* refers to if and how you'd like to be notified of scan-related activity, and *Slave* allows you to take advantage of the fact that any OpenVAS Manager can use one or many other OpenVAS Managers as slaves to run scans. Imagine an OpenVAS farm distributed across multiple sites or datacenters for larger environments. Wrap the New Task process up with *Create Task*. As seen in Figure 6, you're ready to go. Click the green arrow icon under *Actions* and you're off to the races.

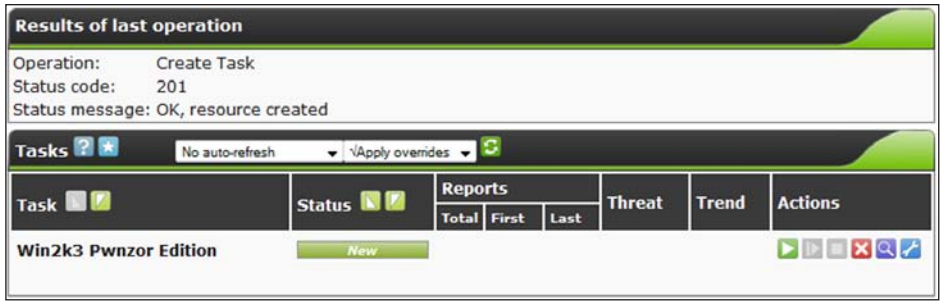

Figure 6 – Task ready to scan

And what's one of the most important expectations of any good vulnerability assessment mechanism? The report, right? You can manage *Report Formats* under *Configurations* where you'll find many options.

When your scan wraps up, click the purple magnifying glass icon for detail as seen in Figure 7 (next page), then choose your report option of preference. That's all there is to it.

If, when it's all said and done, you wish to update OpenVAS daemon settings, view them via Administration, then Settings, but edit them on the VM or host via `/etc/openvas/openvassd.conf`.

You can also create agents to be installed on target machines, and manage NVT overrides where you modify behavior specific to an NVT per threat, port, task, result, etc. You can also pull performance reports for your OpenVAS manager as well as slaves.

In conclusion

I'll admit, when not using a commercial vulnerability scanner, I've always used the free version of Nessus for personal use. I was

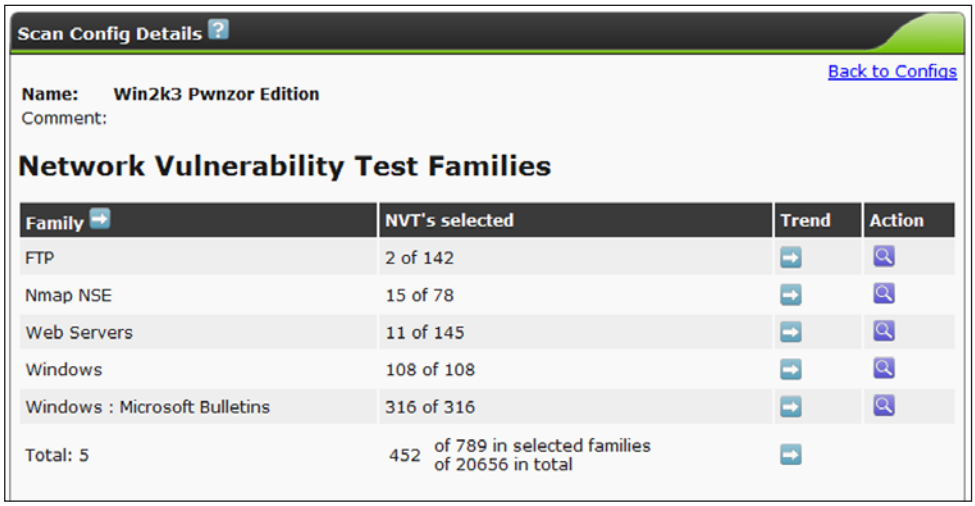

Figure 5 – Network Vulnerability Test Families

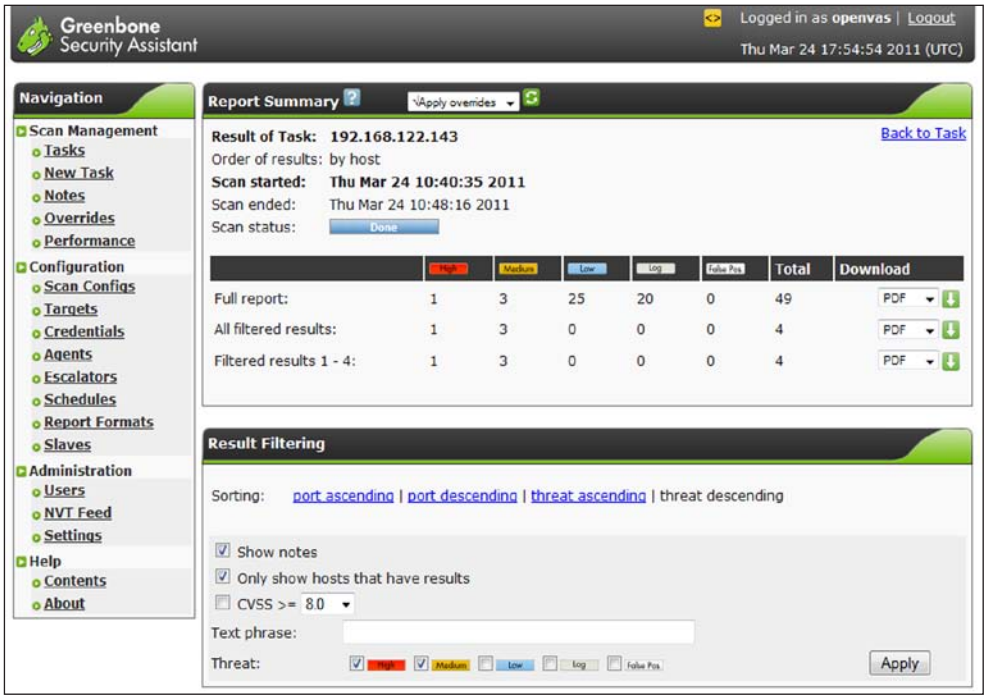

Figure 7 – Report Summary

intrigued by the OpenVAS project but hadn't paid it much attention until deciding to cover it for toolsmith. I'm really glad I did as it has already proven successful and useful on more than one engagement. I heartily recommend evaluating OpenVAS for your vulnerability scanning needs; I'd love to hear your feedback should you conduct a comparison study against Nessus or other commercial product.

Ping me via email if you have questions (russ at holisticinfosec dot org).

Cheers…until next month.

Acknowledgements

—Dr. Jan-Oliver Wagner, OpenVAS project lead and Greenbone CEO

About the Author

Russ McRee, GCIH, GCFA, GPEN, CISSP, is team leader and senior security analyst for Microsoft's Online Services Security Incident Management team. As an advocate of a holistic approach to information security, Russ' website is holisticinfosec.org. Contact him at russ at holisticinfosec dot org.