Defense Information Systems Agency

Passive Vulnerability Scanner

Background and Theory

October 30, 2012

V1

## Introduction

Passive vulnerability scanning is the process of monitoring network traffic at the packet layer to determine topology, clients, applications and related security issues. Tenable has expanded the functionality of the Passive Vulnerable Scanner (PVS) to include traffic profiling and system compromise detection. The PVS can not only keep track of the vulnerabilities for more than 25,000 systems at a time, it can also:

* Detect when they are compromised based on application intrusion detection
* Highlight all interactive and encrypted network sessions
* Detect when new hosts are added to a network
* Track exactly which systems communicate with other systems and on what ports
* Detect what ports are served and what ports are browsed by each system
* Detect how many hops away each monitored host is

This document includes a description of the technology of passive vulnerability scanning, traffic profiling and system compromise detection and also provides directions for deploying, configuring and operating the PVS.

## PVS Feature Overview

### Not a Network Intrusion Detection System

Although it performs some similar functions, the PVS is not a Network Intrusion Detection System (NIDS). It does not run large signature sets of known network attack patterns or probe activity. Instead, as the PVS learns about a network’s applications, it looks for compromise events and vulnerability signatures in traffic originating from just those systems. Traditional NIDS are focused on the type of attacks targeting your network. PVS is focused on determining what is actually running on your network.

For example, if the PVS learns that there is an IIS Web Server running on a specific system and port, it will begin to evaluate the traffic leaving just from that system for evidence of successful Windows buffer overflow attacks, successful downloading of .ASP source code and other attacks that are specific to IIS. If the system were port scanning or had attacks launched against it that did not affect IIS, the PVS would not report anything.

In some cases, the PVS will detect the vulnerabilities left behind by a successful worm, “bot” or “zombie” network or attack. These vulnerabilities are a form of an “intrusion detection” that the PVS can report in real-time.

Another distinction from a NIDS is that the PVS does not record all network traffic. The true value of the PVS is the real-time monitoring of vulnerabilities on the network. The PVS is focused on new activity only, not historical data. Once it detects a particular vulnerability on a system, it does not report it again until the report lifetime threshold is met (seven days, by default) or the PVS is restarted. This enables the PVS to provide the SecurityCenter with concise information about the current condition of the systems on the network it monitors and to prevent multiple instances of the same exact vulnerability from showing up in SecurityCenter.

### Why Passive Vulnerability Scanning?

The PVS is a packet sniffer that looks for vulnerabilities in the client and server applications it observes. It also looks for ports and hosts that may indicate changes on a network. What distinguishes PVS from a NIDS is that it looks for vulnerabilities on the network endpoints continuously in real-time. This provides a much more accurate picture of current conditions. The PVS behaves like a security motion detector on the network, mapping new hosts and services as they appear on the network and monitoring for vulnerabilities as they appear.

The PVS has several advantages over traditional active vulnerability scanners, such as the Nessus vulnerability scanner (<http://www.nessus.org/>) or NMAP (<http://www.insecure.org/>).

First, an active scanner’s results become stale over time. Even if someone can launch an active scan once a day, there still may be new hosts added or removed during that day. Most organizations only scan once a week or month, and the results of their active scan become less valuable over time.

|  |  |
| --- | --- |
| 11769225_Caution_HiRes.png | Often these daily scans are actually “minimal” scans and will miss new services, such as the backdoor installed by the Sasser worm. |

Second, an active scan may inadvertently disrupt a system it is testing. Even if there is no disruption, it will cause a firewall or tested server to generate many log files. A passive scanner such as the PVS has no impact on the systems it is monitoring.

### HTTP and FTP Monitoring

PVS offers extensive web and FTP activity monitoring. By passively monitoring any HTTP or FTP transaction, the PVS can determine and report useful information about each host on your network such as:

* all client and server web-based vulnerabilities and applications
* a list of all web-agents used on each host
* passive enumeration of all files shared via FTP
* real-time logging of every web GET, POST or file download
* real-time logging of every FTP file GET or PUT
* real-time logging of every DNS query

The data is useful for analyzing insider activity, employee activity and any type of malware or “Advanced Persistent Threat”. Many of these logs can be sent to the Log Correlation Engine for further analysis, correlation search and long term storage.

### Windows Domain and Directory Monitoring

PVS offers advanced protocol analysis of the Microsoft SMB protocol. If the PVS is deployed on the interior of a network where it can see Active Directory network traffic, it can automatically learn:

* each system’s hostname and workgroup name
* a list of all files shared on any folder
* logins and file downloads from a network share in real-time

The ability to passively determine this information in real-time has tremendous forensic and situational awareness value. For large networks, the ability to passively determine all shared folder contents can greatly facilitate identification of potentially sensitive data. Sending a record of each file that was shared over the network to the Log Correlation Engine enables forensic analysis of employees and malware activity.

### SQL Database Logging and Monitoring

The PVS also can look at network traffic and identify SQL devices, determine vulnerabilities associated with them and log this activity in real-time.

Real-time logs for SQL querries can be sent to the Log Correlation Engine for search, storage and analysis of attacks such as SQL injection from web services. Full instrumentation of all SQL activity can be achieved by combining the PVS data with Nessus SQL database configuration and vulnerability auditing data, as well as log data gathered from a SQL database server with an LCE agent.

### Example Scenario

Consider the following situation – a security team has hired an outside firm to conduct a security audit of their perimeter network. They scan with NMAP, the Nessus vulnerability scanner and ISS Scanner (<http://www.iss.net/>) and have a good handle on their vulnerabilities. They feel they are prepared for the audit.

The night before the outside scan starts, someone in the company places an unpatched Windows 2000 server in the DMZ. Of course, this violates the company’s change control procedures, but the infraction is not caught until the outside firm starts their testing and compromises the server.

Using a vulnerability scanner “one more time” would have caught the new server, but in this case, the security group needs to get permission to launch a scan.

If a PVS were deployed, it would have provided an alert on the new Windows 2000 machine, identified its open ports and likely have detected its unpatched vulnerabilities.

Additionally, many IT organizations have an end of year IT freeze on network changes. If a PVS were deployed before the freeze, it could provide information about network activity at a time when a scheduled scan may be prohibited. Since the PVS does not generate network traffic, it would not violate the freeze policy.

### Network Deployment

The PVS is a software application that can be deployed on several different operating systems. To be effective, the PVS needs to see as many network sessions as possible for the network it is monitoring. Typically, these devices are deployed with sniffers or NIDS and sometimes on the same platforms. The PVS can be deployed on existing network IDS devices, firewalls, email servers, DHCP servers and many other places without affecting the underlying system’s operation. The PVS can also be deployed on a “standalone” device for dedicated monitoring.

The PVS can monitor Ethernet networks including 10 Mb, 100 Mb and Gigabit Ethernet systems. The PVS also has logic to automatically recognize 802.1q “spanned” port protocols used to share trunks between switches as well as traffic using the Generic Routing Encapsulation (GRE) protocol.

### Performance

The PVS is designed to operate in high-speed environments and has a variety of configuration options that can be tuned to increase performance. PVS does not do any session reconstruction. Since it examines banners, it generally only needs to see the first 1K of data (the headers) to determine vulnerable versions, protocol types, etc. Even though events on the network that would trigger the PVS plugins may appear repeatedly, PVS only reports the presence of the vulnerability once until PVS is restarted or when the report threshold is reached. PVS also does not have to deal with all of the evasion issues that NIDS have to handle.

A NIDS is focused on attacks and must capture every one. PVS is focused on vulnerabilities and does not need to capture attack data. This helps reduce a lot of the “noise” inherent in NIDS deployments. If a Code Red attack is launched against a Unix system, the NIDS does not know by default that the target system is not vulnerable to Windows attacks. PVS knows this by default since it is focused on your network systems and not just network activity.

A NIDS has to do TCP reconstruction on thousands of simultaneous network sessions because any one of them may be an attack. From the PVS’s point of view, those thousands of network sessions may be focusing only on a few servers. Once the PVS starts to analyze a server’s communication over a particular port for a particular vulnerability, there is no need to analyze it again if a different connection occurs. Even in extremely high-speed environments, the PVS can still lock on to servers and services it has not seen yet, and ignore traffic it has already evaluated.

Enterprise networks can vary in performance, capacity, protocols and overall activity. PVS deployments should consider raw network speed, the size of the network being monitored and the configuration of the PVS application when determining resource requirements.

The following chart outlines some basic hardware requirements for operating PVS:

|  |  |
| --- | --- |
| Scenario | Recommended Hardware |
| Passive Vulnerability Scanner managing up to 20,000 hosts | CPU: 1 single-core 2 GHz CPU  Memory: 1 GB RAM (2 GB RAM recommended)  HDD: 72 GB at 7,200 rpm (72 GB at 10,000 rpm recommended) |
| Passive Vulnerability Scanner managing in excess of 20,000 hosts | CPU: 1 dual-core 3 GHz CPU (2 dual-core recommended)  Memory: 2 GB RAM (4 GB RAM recommended)  HDD: 72 GB at 10,000 rpm (72 GB at 15,000 rpm recommended) |

|  |  |
| --- | --- |
| 11769225_Caution_HiRes.png | Please research your VM software vendor for recommendations as VMs typically see a 12 to 30% loss in efficiency compared with dedicated servers. |

Processor requirements will increase with greater throughput and number of network interfaces. Memory requirements will increase for networks with more hosts. The requirements for both of these components are affected by options such as a long report-lifetime and enabling some or all of the PVS optional services in the configuration file.

The PVS only keeps a list of discovered hosts from one process to the next. Closing and restarting PVS will make it forget any data such as open ports, browsed ports or vulnerabilities. This does not relate to any data that has been logged or reports generated.