Penetration Testing in a Box

Lee Epling
Computer Science Department
Northern Kentucky University
Highland Heights, KY 41099
eplingl1@mymail.nku.edu

Brandon Hinkel
Computer Science Department
Northern Kentucky University
Highland Heights, KY 41099
hinkelb1@mymail.nku.edu

Yi Hu
Computer Science Department
Northern Kentucky University
Highland Heights, KY 41099
huy1@nku.edu

ABSTRACT
Network and application vulnerability assessments have a tendency to be difficult and costly; however, failing to have an assessment done and fixing security loopholes may result in a security breach by malicious attackers. A security breach can cost an organization time and money remediating the damage, such as lost confidential business information, which far exceeds the cost of a security assessment. Our solution for this problem is a semi-automated system that can help a penetration tester, security professional, or systems administrator, scan and report on the vulnerabilities of a network and services running on the network. It is also able to perform some simulated attacks. This system relies on a miniaturized computer to host the necessary components to conduct a security assessment. This system has been done in an open source manner to allow others to contribute and benefit from a community effort in the name of security.

Categories and Subject Descriptors
K.6.5 [**MANAGEMENT OF COMPUTING AND INFORMATION SYSTEMS**]: Security and Protection – *Insurance*, *Unauthorized Access.*

General Terms
Design, Experimentation, Security

Keywords
Penetration Testing, Vulnerability Assessment

1. INTRODUCTION
Penetration testing, also referred to as pentest or white hat hacking, is the process of a company hiring computer security professionals to try to break into their IT infrastructure with the intent to find where the greatest vulnerabilities lie. Basically, a company hires security professionals to evaluate and hack into their network, servers, and services before the malicious users can do the same thing. The penetration testers will deliver a report on network, service, and application vulnerabilities. The report may also include how penetration testers are able to penetrate IT infrastructures and applications to get access to certain accounts or systems. In addition, the report will provide recommendations on how to fix these vulnerabilities. This allows the company to secure their networks, services, and applications against a variety of avenues of attacks in the future.

Information Security Curriculum Development Conference 2015
October 2015, Kennesaw, GA, USA.

http://dx.doi.org/10.1145/2885990.2885996

Penetration testing, and security assessments in general, are critical for any company that relies on an IT infrastructure. However, there are a few issues with them. A penetration test procedure can take several weeks or months, depending on the size and complexity of the network being targeted and the level of the details that the client wants. One company may have twenty users and all they want is a basic vulnerability scan to ensure they don't have any major problems with their system's configurations. In that case the scan can be done in a day or so with little interaction beyond entering the target systems and starting the scan. In another scenario, for a company of several thousand users and with a great variance in their infrastructure, the penetration testing process could take several weeks or even months to complete due to the complexity of the network/application architectures and many different attack avenues available.

Since the nature of a penetration testing can vary to a great extent between individual engagements, the price will scale a lot as well for a complex IT infrastructure. Even a basic vulnerability scan or review of a subset of systems can be very expensive for a small company. If the company wants more from the hired firm performing penetration testing, the price begins to increase significantly due to the number of man-hours required from the highly skilled individuals conducting the penetration test.

To streamline the penetration testing procedure and lower the cost of vulnerability assessment, different options exist. There's a number of self-contained penetration testing devices commercially available. Products like the Pwn Plug [1] offer a full suite of penetration testing tools in a home router sized form factor. You connect them on a network and log in from the outside to conduct the penetration test. These are amazing little devices and do their advertised job very well, but have a few characteristics that may make them less than ideal depending on the use case or other considerations.

In this paper, we propose a penetration testing architecture using miniaturized computers such as Raspberry Pi [2] to simplify security assessments and penetration tests. This scheme is meant to allow companies that do not have the budget for hiring a security firm to still perform a full security assessment. Although the penetration testing procedure may take longer to finish than employing commercially available dedicated pentest hardware and software, our experiments showed that the scheme is a cost effective way for discovering security blind spots and protecting valuable IT assets.

2. BACKGROUND AND MOTIVATION
The key difference between penetration testing and what most people would consider black hacking is that penetration testers

will get written permissions before even starting their security assessment. When getting this permission, the penetration testers will also rigidly define the scope and goals of the engagement. Getting permission and defining scope is what distinguishes a criminal or aspiring hacker from a security professional.

Having a security assessment done on company networks and services is not only important, it can be mandatory. Companies often hire security firms to perform comprehensive assessments in order to comply with various regulations and practices. Regulations like PCI and HIPAA, for payment card information and private health data respectively, have very strict requirements for security and can be grounds for fines if they are not met. Certification like the ISO 27000 series, which are for controls in an information security program, may require penetration tests before a company gets the certification. A company may also want to protect their trade secrets, or consumer data it holds, since those are critical to its continued functioning. There's also the fact that no company wants their name in the headlines concerning a security breach.

There are some existing devices utilizing miniaturized computers for pentesting such as Pwn Plug [1], Pwn Pi [3], and MiniPwner [4]. They have different capabilities and features on performing penetration testing.

The original Pwn Plug was released in 2012 as a low profile penetration testing device. It came with a 1.2GHZ ARM processor and 512MB of RAM. It came installed with a release of Debian 6 that was optimized for the hardware. The installed programs were open source security tools or optimized versions of open source tools. It included a simple web interface for administration called "Pwnix UI." It was designed to be stealthily connected to a wired network and maintain connections to an outside user. Newer models of the Pwn Plug came with more powerful hardware and built-in wireless functionality comparable to a home wireless router.

The Pwn Pi is a custom Linux distribution for the Raspberry Pi. It includes a long list of popular penetration testing tools. It's based on the Debian image and aims to be used as part of a Raspberry Pi drop box. It has not been updated since 2012 and may not support the latest hardware and software.

The MiniPwner is a penetration testing dropbox made using a portable wireless router, giving it built-in wireless networking capabilities. The battery operated wireless router is designed to be a network entry point, with very little work being done on the physical device itself. It's designed to have more complex or resource intensive tools run through a VPN connection. The battery operated feature also allows the device to be used for wireless war-walking or similar penetration testing scenarios of wireless networks.

Although commercially available small form pentesting devices mentioned above can achieve certain vulnerability scanning goals, our purpose is to build an extremely low cost (around $100) pentest platform using a miniaturized computer and open source tools.

3. MODEL OF THE PROPOSED PENETRATION TEST ARCHITECTURE USING MINIATURIZED COMPUTER

In our proposed model, a security professional can monitor and assess vulnerabilities on a target system in a corporate network by using a computer connected to the Internet, a custom designed penetration testing web interface, and a miniaturized computer (we call it Pentest box in figures and descriptions of this section). The proposed penetration test architecture is illustrated in Figure 1. It depicts a scenario where a Pentest box like a Raspberry Pi is plugged in a corporate network and run vulnerability scanning and penetration testing software. Since it runs behind the corporate firewall, in order to access vulnerability data generated by it a Virtual Private Server (VPS) box on the Internet is employing for running software collecting vulnerability data using SSH callback for creating a SSH tunnel with the Pentest box. Details for creating the connection and sending over the data from Pentest box to the VPS is illustrated in Figure 2, Figure 3, and Figure 4.

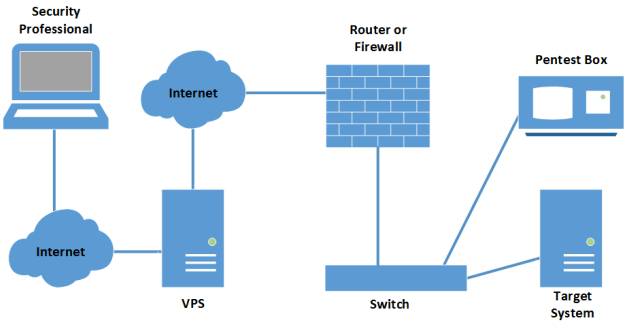

Figure 1. The proposed penetration architecture

Since every corporate network uses perimeter devices such as routers and firewalls to protect its intranet from malicious Internet traffic, it is difficult to initiate a direct connection to the Pentest Box using security professional's computer located outside of the corporate network. In order to connect to it, we can do it indirectly by using SSH callback technique from the Pentest Box. In another word, the Pentest Box will initiate the connection directly to the VPS on the Internet. The connection scheme is shown in Figure 2. You may think the VPS as a stepping stone for helping connect the Pentest Box with the security professional's computer.

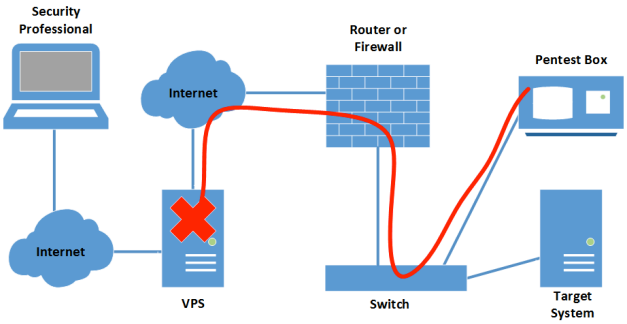

Figure 2. Callback connection from pentest box

After the VPS has an active connection with the Pentest box inside the corporate network, the box controlled by the security professional can connect to the VPS to check the penetration testing status through VPS. Figure 3 shows this connection step initiated from the computer of security professional to the VPS.

Once the connection between the Pentest box in the intranet and the computer controlled by the security professional outside of the corporate network is created by steps illustrated in Figure 2 and Figure 3, the security professional can access the vulnerability assessment results by visiting a pentest dashboard web application

designed by us. The Pentest dashboard application is hosted on the Pentest Box. Figure 4 shows the connection route that enables the penetration testing procedure.

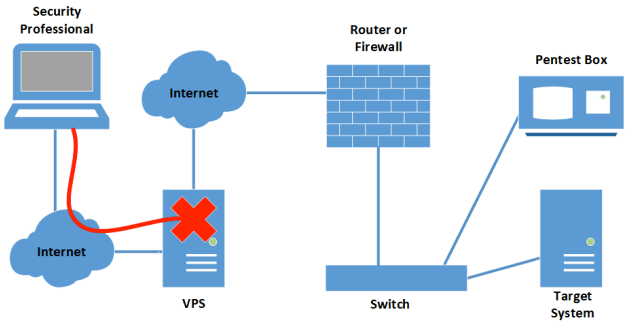

Figure 3. Connection from security professional's computer to the VPS

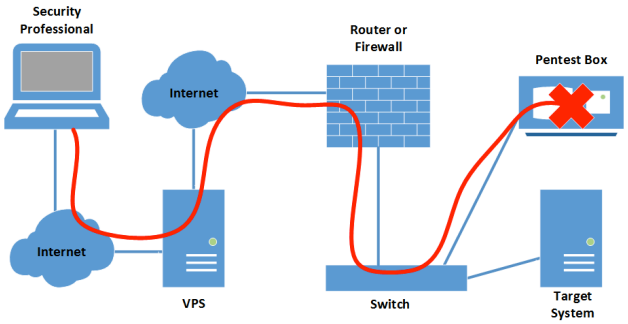

Figure 4. Connection from security professional's computer to the pentest box

4. EXPERIMENT AND ANALYSIS

The price range that you are willing to work with will determine the type of hardware that you select, and ultimately how much power that you have to run any software. Miniaturized computers range from the low cost Raspberry Pi to the more expensive, but more powerful, ODROID from Hardkernel [5]. If you can afford more powerful hardware, use that and save yourself some trouble for down the road. In developing our pentest box, we set ourselves a $100 limit to cover all our hardware purchases, including accessories in order to test how effective we can perform the penetration testing procedure on a very low budget.

Most small form factor devices are likely to have very limited resources, unless you have a large enough budget to afford something more powerful. When planning for minimal resources, expect to wait for some tasks to complete on the pentest box.

There are plenty of open source tools for security purposes freely available online. Don't look at tools if they require a license or have other restrictions on their usage if you plan to build a low cost open platform for penetration testing. Popular open source tools usually have a helpful user base behind them that can be contact through forums, mailing lists, and Github pages for the project. Making use of the existing community is helpful for solving any issues you may run into.

4.1 Experiment Planning Based on Goal Spectrum

The ability of a pentest box is only limited by what the user is able to use and configure, and what the hardware is able to support. There are a few possible goals for the pentest box:

Do you plan to do entire penetration tests?

Do you want to do vulnerability scanning?

How much automation would you need?

An entire penetration test would require constant and reliable connections, and the ability to run a diverse tool set depending on the targeted network. Vulnerability scanning is resource intensive and can be incredibly slow, so they will require a lot of patience. Automation efforts will vary dramatically depending on what is needed from the pentest box. The final configuration of the pentest box will vary depending on what its intended purpose is. It's okay to start making a pentest box without a specific end goal, but knowing what you want makes it easier to work toward a specific goal.

4.2 Hardware and Software Used

The hardware that we chose for the pentest box was the Raspberry Pi Model B+ and Pi 2. Pi 2 was the most recent version of the Raspberry Pi available. Any miniaturized computer should work, provided it can run a Linux distro like Kali. Accessories like storage media, a wireless adapter, and interface devices will be needed as well.

Most software for making a pentest box will come with Kali Linux for ARM. NMAP and Metasploit are included in Kali Linux and can be downloaded to most other operating systems if needed. Metasploit depends upon the Ruby programming language, so Ruby will need to be installed and working before Metasploit can be made useful. OpenVAS [8] will need to be installed, since it isn't always included with Kali. It has a more complex setup and will require multiple parts configured before everything works.

4.3 Dashboard Site for Penetration Testing

In order to install the website on the Pi, you will need to install a web server, PHP, and the PHP PDO library for Postgresql. Thankfully, this only requires a single command using apt-get and a little bit of configuration editing for nginx [6], if you choose to use it. The command you will want to use to install everything that you need is ***apt-get*** *install nginx php5-fpm php5-pgsql*.

If you are unfamiliar with nginx or why you would want to use it, here is a brief introduction. nginx is a lightweight and scalable web server. Apache, a competitor to nginx, is easier to set up than nginx and would work for this project, but Apache is not meant to run on minimal resources. So you will be more likely to experience issues with resource utilization using Apache.

4.4 Automation of Penetration Testing Process

Automation is a very powerful and helpful technique for many common or repetitive tasks in information technology. It often takes a large amount of efforts at the start, but pays off later in the amount of efforts saved. The automation for the pentest box is focused on the actions occurring at startup. The automation script that we created conducts a reconnaissance scan of the local

network and populates the Metasploit database with the discovered hosts, ports, and vulnerabilities. A few other services may be started, but no actions beyond basic scanning occur.

Automation is also something to be very careful with. Launching certain actions has the potential to crash servers or take down networks. We attempted to automate vulnerability scans during our original development, but after a vulnerability scan unexpectedly blue-screened a Windows server, we determined it was best not to try automating a vulnerability scan at startup. When automating any task or series of tasks, make sure that they are unlikely to cause issues for any system involved.

4.5 Development of a Web Frontend

The web frontend is the interface that the security professional needs to interact with to do a full penetration test. The website is built on the Twitter Bootstrap framework for the overall design. This gives the website a clean and professional look with minimal custom designing needed. Time and efforts can be spent toward website functionality instead of design. Figure 5 shows the vulnerability scanning results on 3 servers, i.e., two Linux servers and one Windows 2008 server. Figure 6 shows the ports open on one Linux server. In addition, Figure 7 illustrates the vulnerabilities discovered and depicts the detailed nature of these vulnerabilities.

IP Address	Operating System	Services	Vulns	Extra
10.151.8.181	Linux	1	0	N/A
10.151.8.1	Linux	2	1	N/A
10.151.8.55	Windows 2008	2	1	N/A

Figure 5. Vulnerability scanning results

Port	Proto	State	Name	Info
21	tcp	open	ftp	vsftpd 2.3.4
22	tcp	open	ssh	OpenSSH 4.7p1 Debian 8ubuntu1 protocol 2.0
23	tcp	open	telnet	Linux telnetd

Figure 6. Ports opened on a server scanned

Name	Info
ProFTPD Multiple Remote Vulnerabilities	
NFS export	Here is the export list of 10.151.8.186 : / * Please check the permissions of this exports.
X Server	This X server does *not* allow any client to connect to it however it is recommended that you filter incoming connections to this port as attacker may send garbage data and slow down your X session or even kill the server. Here is the server version : 11.0 Here is the message we received : Client is not authorized Solution: filter incoming connections to ports 6000-6009

Figure 7. Vulnerabilities discovered on a server

5. CONCLUSIONS

Using a Raspberry Pi when conducting a penetration test can result in many different outcomes. There's no shortage of efforts going into making the most from the minimal resource to create stealth tools for use with computer security. Drawing on that existing knowledge gave us a great starting point for planning pentest box of our own. Miniaturized computer hardware can fit even the smallest budgets if you're willing to work with a bare minimum of resources. Choosing a software package that has all the necessary tools installed will streamline the rest of the setup process and help you hit the ground running. Automation is a time saving endeavor that should be tempered with a healthy caution concerning the effects of working fast. In the future, we plan to expand our architecture to make it a distributed penetration testing framework using a network of miniaturized computers. It is our hope that by open sourcing all we've done others will take on the concept and develop it into something even more useful.

6. ACKNOWLEDGMENTS

Funding for this project has been provided by the NKU Undergraduate Research Council Award.

7. REFERENCES

[1] Pwn Plug, https://www.pwnieexpress.com/

[2] Raspberry Pi, https://www.raspberrypi.org/products/model-b-plus/

[3] Pwn Pi, http://www.pwnpi.com/

[4] MiniPwner, http://www.minipwner.com/

[5] ODROID, http://www.hardkernel.com/main/main.php

[6] Nginx, http://nginx.org/

[7] Twitter Bootstrap, http://getbootstrap.com/

[8] OpenVAS, http://www.openvas.org/index.html