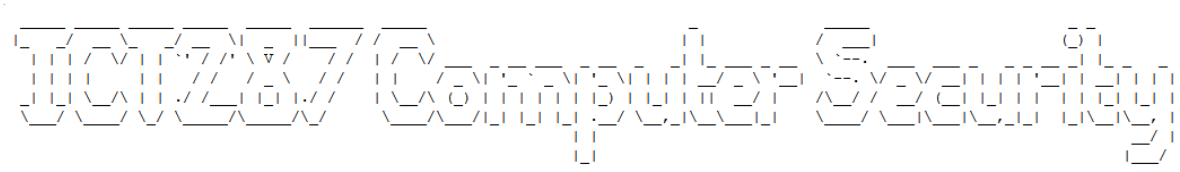


Tutorial Two: Network Security

In the first lab we already looked at reconnaissance-related techniques, such as whois and nslookup. In this lab we will look at scanning a remote target with the tool nmap and identifying vulnerabilities.

Finally, we will have a brief look at how to use the search engine Shodan to find vulnerable targets in the Internet. As a challenge you should also have look at how to use vulnerability scanners, e.g. OpenVAS or Nessus, for a more detailed scan.

Save the output of each task for the participation quiz.

Port Scanning

When you start doing active recon on a network, the possibilities are endless. A port scan will help you to identify and enumerate hosts, and to ascertain what services are running on them in some cases too.

Before we start, download and unzip the Windows 7 VM from LMS or if you are in the Perth lab from the link below. Add it to VMware by opening the .ova file. If an error pops up because of a lack of OVF spec conformance, click on Retry.

http://dnoc-isocache.murdoch.edu.au/VMs/ICT287/Windows7VM.7z

Confirm and/or set the network interface of the VMs to Host-only if you are in the campus lab using VMWare (VM->Settings). If you do this at home with VirtualBox then you need to set the network interfaces of both VMs to either "NAT network" or "Internal network" and first you may have to create a NAT network under File->Preferences->Network. Start the Kali and Windows 7 VMs. After they are booted up, confirm that there is a working network connection between Kali VM and Windows 7 VM.

Manual port scanning

We can do manual port scanning simply by connecting to a port of our choosing and viewing the server response. **netcat** can connect to a port of our choosing and show us the output. This can be used to determine what services are available.

Task 1: Use netcat to check if there are any SMTP or POP services running on the VM. Write down your findings.

Of course, this would take forever if we wanted to check out a whole range of ports so we would either script this task or use the right tool for the job: **nmap**

Task 2: One of this week's readings is an overview of nmap. Using the basic command reference as your guide to perform a nmap scan of the Windows 7 VM with default settings.

Task 3: Use nmap to discover full version information about any services running on the target machine as well as information about the OS on the target machine.

Task 4: Find out what ports nmap scans by default. How can you scan other ports that are not included in the default set?

Vulnerability assessment

Task 5: Use the list from Task 3 and search (Google is your friend) to find out if there are known exploits or vulnerabilities for the listed services. One possible option for the search is https://www.cvedetails.com/. There are only a few services so it's not a huge job, but don't spend more than 10 minutes on this.

You can see that if there were dozens of ports open and more machines, this manual searching task would become tiresome and infeasible. nmap actually has a cool scripting functionality to automate a bunch of things including scanning for vulnerabilities. You can find all the scripts under /usr/share/nmap/scripts/.

Task 6: Run nmap with --script against the target using the following two (sets of) scripts: (1) vulners, (2) smb-vuln-* and see what you can find. Make sure version and OS detection are enabled as for the previous scans in task 3.

However, nmap only supports a few scripts for some vulnerabilities and there are other vulnerability assessment tools better suited for the job. Two such tools are **OpenVAS/GVM** and **Nessus**. We won't be using them for now, but you should install and use one of them on your Kali machine for the challenge task.

Task 7: If you identify any vulnerability in task 6 (or your manual searches), check if there is a Metasploit module available using https://www.rapid7.com/db. Metasploit is a tool to automate exploits which we will be using in a later lab. Report back to your tutor on any vulnerability you find. Note that the goal here is not to exploit the vulnerability, we will do this in a later lab.

Vulnerable Computer Search Engine

Another way to identify and profile hosts is through search engines. This is a whole topic in itself, but for now we will use Shodan.

Shodan is a search engine that lets the user find specific types of computers (routers, servers, etc.) connected to the internet using a variety of filters. Some have also described it as a search engine of service banners, which are meta-data the server sends back to the client. This can be information about the server software, what options the service supports, a welcome message or anything else that the client can find out before interacting with the server.

Shodan can be found at https://www.shodan.io/

Another search engine is Censys: https://censys.io/

Task 8: Use Shodan to identify a vulnerable host somewhere on the Internet. Consider what you might be able to do with this host if you would gain access. Don't spend more than 10 minutes on this in the lab, you can continue this at home.

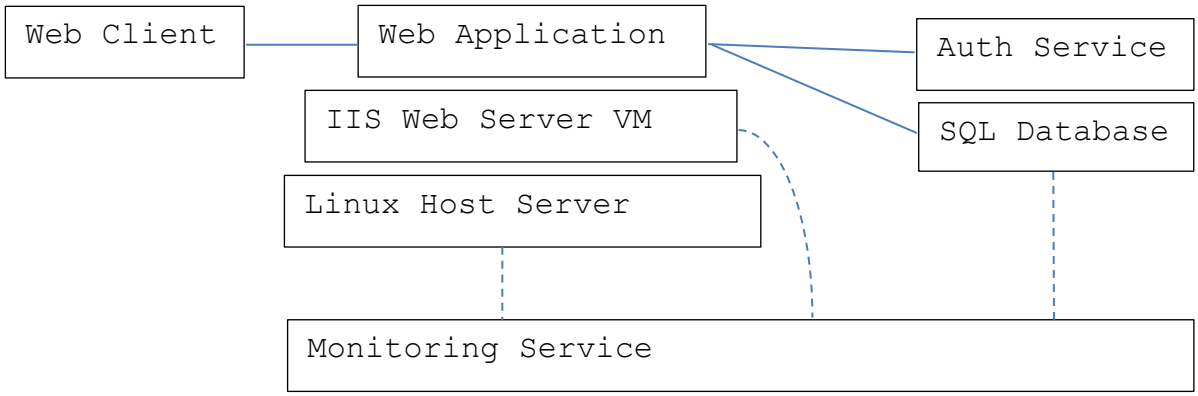
Threat Identification

Tools like nmap, OpenVAS/GVM and Nessus are great to identify vulnerabilities for networked application and services. However, these tools are limited to existing vulnerabilities, and they can't be used to assess things like physical security etc. There are many threats against systems and organisations that these tools can't identify. This is why threat identification and modelling are important.

Task 9: Based on the following description of an organisation use the STRIDE model discussed in the lecture to identify TWO technical threats (one threat related to physical assets and one threat related to the web service offered). Of course, as in the real-world descriptions always lack details and, in this case, we can't contact anybody from the organisation, so you need to fill in any blanks with reasonable assumptions (or educated guesses). Discuss the identified threats with other students in your group.

ACME Inc. is a start-up IT company that has created the new ACME web service. The company's offices are in an old residential building rather than a purpose-build office building. There are no locks inside the office space (including for rooms that house the internal servers), any IT equipment is not secured in any way and there is no CCTV. An alarm system has been installed and for convenience all employees have been told the code to enable/disable the system. Due to bad management the company is struggling to retain employees and staff turnover is very high. Desks in the offices are assigned in a chaotic way so that staff from different apartments are mixed. To save costs no training of any kind is offered to employees. The IT support is generally under-staffed which means they can only focus on the bare necessities. To keep things simple the company policy is to throw any old/damaged IT equipment in the rubbish bin and after a recent accident, the only document shredder has been disabled.

The following picture shows a simplified diagram of the company's web service architecture.



SSLv3 is used to secure traffic between clients the web application running on IIS servers. To allow for easy management of the web servers telnet is enabled on all servers with a username and password that is known by all administrators and developers. For convenience there are several hidden directories on the web servers that contain files that should not be accessed by users. The IIS web servers run in virtual machines on a Linux host.

For authenticating users, the 3rd part service DAuth is used. To make it convenient for users DAuth only used passwords and users can choose any passwords they like and there is no lockout policy. The DAuth server stores passwords hashed with MD5. Resetting passwords is done via a web page after the users answers a security question correctly that they selected when creating their account. DAuth uses a home-grown encryption algorithm for the communication with the web application that is extremely fast.

To track and maintain user sessions simple web cookies are used with cleartext data and predictable sessions IDs. For cost reasons no load balancers or firewalls are used in front of the web servers. To save disk space on the web servers logging is generally disabled. The backend SQL database is hosted by a cloud provider. Hardcoded credentials are used by the web application to access the database.

A monitoring service hosted in the cloud is used to monitor the Linux host, the web server VMs and the database. The service uses SSH to connect to the different systems with credentials hardcoded in the monitoring software. The information is made available via SNMPv1. Recently, an employee added some functions to start and stop web servers and enable/disable web server logging via a simple HTTP web interface which is secure via IP-address based ACLs.

Challenge Task

The challenge task is to install the vulnerability scanning tool OpenVAS/GVM or Nessus on your Kali machine and use it to scan the Windows 7 VM we previously scanned with nmap.

You can install OpenVAS/GVM on Kali with few simple steps https://www.agix.com.au/installing-openvas-on-kali-in-2020/. Start and access it on the local host with http://localhost.

You can download Nessus from https://www.tenable.com/products/nessus/nessus-essentials. It is free for personal use, but you need to register. Download, install and run Nessus on your Kali machine. You can access Nessus from within your Kali machine by opening a browser and navigating to https://localhost:8834.

To proof that you have done the challenge you need to submit a brief description of the steps you took to install the tool you used and carry out the scan. You also need to provide a screenshot of the top of the vulnerabilities list.