Network Security

Assignment 3, Friday, September 19, 2014, version 1.2

Handing in your answers: Submission via Blackboard (http://blackboard.ru.nl)

Deadline: Friday, September 26, 23:59:59 (midnight)

In this exercise you will be using the following tools:

- Nmap: http://nmap.org/
- The Python 3 network sniffer from the first exercise, if you have not handed in exercise 4 from assignment 2 yet (a reference implementation (sniffer.py) is available on the course page: http://cryptojedi.org/peter/teaching/network-security-2014.shtml).

Do not compile these programs from source. Rather, use your Linux distribution's package manager to install the package nmap (e.g. Ubuntu / Debian: `apt-get install nmap`, Arch: `pacman -S nmap`, Fedora: `yum install nmap`, Gentoo: `emerge nmap`).

This assignment consists of several theoretical questions and one practical exercise. Don't forget that exercise 4 from assignment 2 can also still be handed in this week. Refer to that assignment for the exercise details.

Please turn in all your work in plain text files (program source files are also plain text). If you prefer a document with formatting for whatever reason, like including images, use the PDF format to turn in your work (most editors allow you to export to PDF). Note that it's okay to include images separately and then refer to them from within the text files.

1. This question is about IP address spoofing. Write your answers to every subquestion to a file called `exercise1`, making sure to prefix each answer with the letter for that question.

   (a) Take a look at the RFC for the Internet Protocol, RFC 791 (https://www.ietf.org/rfc/rfc791.txt). Explain what IP address spoofing is, and what a host on the network must do to spoof its IP address.

   (b) Take a look at the RFC for the User Datagram Protocol, RFC 768 (http://www.ietf.org/rfc/rfc768.txt), and the RFC for the Transmission Control Protocol, RFC 793 (https://www.ietf.org/rfc/rfc793.txt). Explain why an attacker cannot just grab any existing IP packet carrying UDP or TCP, change only the IP addresses in there, and expect the target host to accept the packet. Especially for TCP, don't read the entire RFC but focus on the header (pages 15–19).

   (c) During the lecture we explained SYN flooding attacks. Review that now (slide 8 (page 27) in http://cryptojedi.org/peter/teaching/netsec2014/tcpip.pdf).

   If an attacker wants to do SYN flooding while IP spoofing, she faces a problem. Let's first consider the case where the attacker, Mallory, tries spoofing the IP of an existing host, Bob, to SYN flood her target, Alice. Using the TCP RFC, explain which packets get sent to whom when Mallory sends a SYN to Alice using Bob's address as source. Focus on the TCP three-way handshake and the Reset Generation in section 3.4, "Establishing a connection". Explain why this will cause the SYN flood attack to fail.

   (d) When Mallory tries to use the address of Ursula, who's currently not on the network, she does not face the problem you uncovered in the previous question. However, there is now another protocol in play which causes the attack to fail. Take a look at the RFC for the Internet Control Message Protocol (https://tools.ietf.org/rfc/rfc792.txt). Read the first five pages, and explain which packets get sent to whom when Mallory sends a SYN to Alice using Ursula's address as source. Explain why this will cause the SYN flood attack to fail.

(e) Using what you've learned in this course so far, describe a way to make Mallory's SYN flood attack succeed against Alice, while IP spoofing using either Bob's or Ursula's address. You may assume that Mallory is in the same network as Alice. If you make any more assumptions (e.g. Mallory is able to modify all traffic on the network) please state these.

2. This question is about port blocking. Write your answer to a file called `exercise2`.

   As a network security measure, some network administrators attempt to restrict what external services their users can access by blocking any outgoing connection made to all but a few well-known ports. This by itself is not a very effective measure, since it will inconvenience normal users, but it will do nothing to stop somebody in control of the external endpoint.

   Consider you are such a person. You know that soon you will be on a network that blocks every outgoing connection except on port 53 (DNS), 80 (HTTP) and 443 (HTTPS). What could you do to be able to access the SSH service on your external server once you are inside this network?

3. This exercise has you using nmap. Write your answers to separate files in a folder called `exercise3`. So for exercise 3a you should use `exercise3/exercise3a`, etc.

   You will be using the nmap manual page (`man nmap`) a lot, since there will be almost no hints on how to perform the tasks in this exercise. It has an "examples" near the end. To search the man page, press "/", then type the string you want to search for, then hit "Enter". To search for the next occurrence, press "n". To search for the previous occurrence, press "shift+n". Always use "shift+n" if it seems like the string you are searching for has not been found, since it might occur earlier in the document and by default searching only works forwards.

   Note that you will need root rights to execute many of the scan types nmap provides, since they use raw sockets.

   Connect to the homework network. You should have recovered the key in the previous exercise. If you have not, e-mail Pol for the key or take this opportunity to crack it (There's an encrypted reference capture you can use for this at `http://www.cs.ru.nl/~erikpoll/netsec/reference-capture.cap`). If you connect using a static IP, use an IP address in the range 192.168.84.100–149. The netmask is /24, or 255.255.255.0.

   The network will be present at the werkcollege, and at other times it will be reachable from the central common area on the first floor of the Mercator 1 building. (*Not* the ground floor, you American. We start counting at zero.)

   Note that some of you will be scanning each other if you're performing this exercise at the same time. This is fine. Portscans are not harmful[1]. However, do *not* attack each other.

   (a) Read the manual page section on host discovery. Your first task is to map the network. Discover all the active hosts, and write your results to `exercise3a`. Also explain how you discovered them.

   You should find at least 4 active hosts in the range 192.168.84.1–60

   Since you're scanning on a wireless network, the scans appear to be not as reliable as on a wired network. Furthermore, you're not supposed to spend hours waiting on scans to complete. Read the manual page section on timing and performance. Try to play with a higher timing template (-T), different max-retries or host-timeout, and other stuff, if hosts seem to intermittently drop from the network, or if scans take too long.

   (b) Read the manual page section on OS detection. Try to detect the Operating Systems running on every host you found, at least in the range 192.168.84.1–60. Write your results to `exercise3b`, and explain how you got them.

[1] If your network stack is sane. Some devices actually manage to be so crappy they crash when portscanned, but your laptops are not among them. However, if you ever find yourself using the skills you learn here to scan networks with critical devices, always check whether it is safe to scan them before proceeding.

(c) Read the sections on port scanning basics, port scanning techniques, and port specification and scan order. Scan for open ports on the hosts you found, at least in the range 192.168.84.1–60. Write your results to `exercise3c`, and explain your scanning techniques.

(d) Read the section on service and version detection. Try to detect what services are running on what ports for some interesting hosts in the range 192.168.84.1–60. List these results in `exercise3d`. Also mention which of these services are running on non-standard ports. If nmap is unable to determine what a service is, make a guess based on its port number.

A list of standard ports is often available on Linux machines in `/etc/services`, and can also be found at the Internet Assigned Numbers Authority (IANA) (http://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xhtml).

(e) (OPTIONAL) If you were an attacker, intent on gaining access to one of these machines, which service would you attack first, and why? Write your answer to `exercise3e`.

4. Don't forget exercise 4 from assignment 2! Use the folder `exercise4` for that.

5. Place the files and directories `exercise1`, `exercise2`, `exercise3`, and `exercise4`, and all their contents in a folder called `netsec-assignment3-STUDENTNUMBER1-STUDENTNUMBER2`. Replace `STUDENTNUMBER1` and `STUDENTNUMBER2` by your respective student numbers, and accomodate for extra / fewer student numbers. Make a `tar.gz` archive of the whole `netsec-assignment3-STUDENTNUMBER1-STUDENTNUMBER2` directory and submit this archive in Blackboard.