SecureSet Prep
Introduction to Network Security 3

Glenn Webb, CISSP, GSEC

Table of Contents

Netcat

Netcat (often abbreviated to nc) is a computer networking utility for reading from and writing to network connections using TCP or UDP. Netcat is designed to be a dependable back-end that can be used directly or easily driven by other programs and scripts. At the same time, it is a feature-rich network debugging and investigation tool, since it can produce almost any kind of connection its user could need and has a number of built-in capabilities.

Simple Chat Application

1. on your original VM start wireshark
2. on your original VM run:
    a. nc -l 5000
3. on your clone VM run:
    a. nc *IP_Addr_of_original* 5000

Simple Web Client

1. nc www.ebay.com 80
2. GET / HTTP/1.1
3. Host: www.ebay.com
4. press Enter

Simple Scanner

1. nc -zv -w1 insecure.org 22
2. repeat the above command using 80, 443, 31337

Simple Web Server

1. Bring up Firefox in your main VM
2. In the URL bar type in the following but don't hit Enter yet
    a. *http://IP_address_of_Ubuntu_clone:8080*
3. In your Clone VM bring up a terminal window and type the following
    a. `{ printf 'HTTP/1.0 200 OK\r\nContent-Length: %d\r\n\r\n' "$(wc -c < /usr/share/wireshark/rawshark.html)"; cat /usr/share/wireshark/rawshark.html; } | nc -l 8080`
    b. press `Enter`
4. In your main VM, hit Enter for the URL you entered
    a. You should see a man page which was served by your temporary netcat webserver

ARPANET, TCP/IP, and DNS

The **Advanced Research Projects Agency Network** (**ARPANET**) was an early packet switching network and the first network to implement the protocol suite TCP/IP. Both technologies became the technical foundation of the Internet.

As the project progressed, protocols for internetworking were developed by which multiple separate networks could be joined into a network of networks. Access to the ARPANET was expanded in 1981 when the National Science Foundation (NSF) funded the Computer Science Network (CSNET). In 1982, the Internet protocol suite (TCP/IP) was introduced as the standard networking protocol on the ARPANET. In the early 1980s the NSF funded the establishment for national supercomputing centers at several universities, and provided interconnectivity in 1986 with the NSFNET project, which also created network access to the supercomputer sites in the United States from research and education organizations. The ARPANET was decommissioned in 1990.

https://en.wikipedia.org/wiki/ARPANET

In the good old days, the mapping between hostname and addresses was kept in a single text file that was managed centrally and distributed to all the hosts on the ARPANET. Hostnames were not hierarchical and the procedure for naming a computer included verify that no one else in the world had taken the name you wanted. Updates consumed a large portion of the ARPANET's bandwidth, and the file was constantly out of date. It soon became clear that although a static host table was reasonable for a small networks it was inadequate for the large

and growing ARPANET. DNS solves the problems of a static table by using two key concepts: hierarchical hostnames and distributed responsibility. In 1985, Kevin Dunlap produced the first version of BIND (Berkeley Internet Name Domain system).

The DNS namespace is a tree of domains. Each domain represents a distinct chunk of the namespace and is loosely managed by a single administrative entity. The root of the tree is called ". " or dot, and beneath it are the top-level (or root-level) domains. The top-level domains are relatively fixed.

One branch of the naming tree maps hostnames to IP addresses, and a second branch maps IP addresses back to hostnames. The former branch is called the "forward mapping" and the BIND data files associated with it are called "forward zone files". The address-to-hostname branch is the "reverse mapping" and its data files are called "reverse zone files".

Within the DNS system, fully qualified names are terminated by a dot, for example, "boulder.colorado.edu**.**". The lack of a final dot indicates a relative address. The final dot convention is generally hidden from everyday users of DNS. In fact, some systems (such as mail) will break if you supply the dot yourself. A domain can be subdivided into subdomains, for example, along departmental lines, i.e. anchor.cs.colorado.edu. The creation of subdomains must be coordinated with the administrators of the domain above to guarantee uniqueness.

BIND software

The BIND system has three components:

- a daemon called **named** that answers queries
- library routines that resolve host queries by contacting the servers of the DNS distributed database
- command-line interfaces to DNS: **nslookup**, **dig**, and **host**

named

named answers queries about hostnames and IP addresses. If named doesn't know the answer to a query, it asks other servers and caches their responses. named also performs "zone transfers" to copy data among the servers of a domain. Nameservers deal with zones and a zone is a domain minus its subdomains.

Types of named servers

Type of server	Description
authoritative	An official representative of a zone
master	The primary repository of a zone's data; gets data from a disk file
slave	Copies its data from the master
nonauthoritative	Answers a query from cache; doesn't know if the data is still valid
caching	Caches data from previous queries; usually has no local zones
forwarder	Performs queries on behalf of many clients; builds a large cache
recursive	Queries on your behalf until it returns either an answer or an error
nonrecursive	Refers you to another server if it can't answer a query

Master, slave, and caching-only servers

Master, slave, and caching-only servers are distinguished by two characteristics: where the data comes from and whether the server is authoritative for the domain. Each zone has one master name server. The master keeps the official copy of the zone's data on disk.

A slave server gets its data from the master server through a "zone transfer" operation. A zone can have several slave name servers and ***must*** have at least one. It is fine for the same machine to be both a master server for your zones and a slave server for other zones.

A caching-only name server loads the addresses of the servers for the root domain from a startup file and accumulates the rest of its data by caching answers to the queries it resolves. A caching-only name server has no data of its own and is not authoritative for any zone.

An authoritative answer from a name server is guaranteed to be accurate; a non-authoritative answer might be out of date (but not typically). Master and slave servers are authoritative for their own zones but not for information they have cached about other domains.

Although they are not authoritative, caching-only servers can reduce the latency seen by your users and the amount of DNS traffic on your internal networks. Consider putting a caching-only server on each subnet for quickly answering user's queries.

Recursive and nonrecursive servers

Name servers are either recursive or nonrecursive. If a **nonrecursive** server has the answer to a query cached from a previous transaction or is authoritative for the domain to which the query pertains, it provides an appropriate response. Otherwise, instead of returning a real answer, it returns a referral to the authoritative servers of another domain that are more likely to know the answer. Root servers and top-level domain servers are all nonrecursive because of the number of queries they receive.

A **recursive** server returns only real answers or error messages. It follows referrals itself, relieving the client of this responsibility.

DNS Query Process

Suppose we want to look up the address for vangogh.cs.berkeley.edu from lair.cs.colorado.edu. We assume that none of the required information has been cached before the query except for the names and IP addresses of the root servers.

1. The host lair asks its local name server ns.cs.colorado.edu for the answer.
2. The local name server ns.cs.colorado.edu (which is a recursive server) doesn't know anything about cs.berkeley.edu, berkeley.edu, or even edu so it queries a root server about vangogh.cs.berkeley.edu.
3. The root server refers us to the servers for the edu domain
4. ns.cs.colorado.edu asks the edu server about vangogh.cs.berkeley.edu

5. The edu servers refer us to the berkeley.edu servers
6. ns.cs.colorado.edu asks the berkeley.edu servers about vangogh.cs.berkeley.edu
7. The berkeley.edu servers refer us to the cs.berkeley.edu servers
8. ns.cs.colorado.edu asks the cs.berkeley.edu servers about vangogh.cs.berkeley.edu
9. The cs.berkeley.edu servers return us the IP address of vangogh.cs.berkeley.edu
10. ns.cs.colorado.edu returns the IP address of vangogh.cs.berkeley.edu to lair

DoS/DDoS and Network Security

Defense in Depth

Defenses are deployed in a layered fashion. What would defense in depth look like in a corporate setting or your school setting?

Here is an example of defense in depth

https://www.pinterest.com/pin/966163544895660 17/

Discuss each layer of defense.

CIA can be affected in ways not related to hacking. What are some ways?

Attacks

Land attack - nmap a victim server to find a listening port. Then send a SYN to the target with the source:port and destination:port set to the victim (spoofed). This would cause the server to enter into a loop and eventually lock up. Describe the packet mechanics.
Ping of death - Early implementations of TCP/IP were written to only expect ICMP ping messages to be 32 bytes in length. A large ping (ICMP) packet would be created by the hacker and then sent to the target system. If the target system couldn't handle a large packet it would cause a buffer overflow condition in the kernel and thus crash the system.

What is a buffer overflow?

What does it mean for a system to crash?

Infected USB - what would the average person do if they found a really nice new USB drive? What if it was strategically place in the path of a person who had a system we wanted to target?

Smurf attack - this is a network distributed denial of service (DDOS) attack. An attacker gets the IP address of the system he wants to DDOS. He then sends an ICMP Echo Request (a ping) to the broadcast address on the LAN segment. (Why only on the LAN segment?) with the victim's IP address in the source field. What happens next?

Contents of a ping packet

https://www.slideshare.net/null0x00/dos-threats-and-countermeasures-12990644

Note: Smurf attack is no longer effective because implementers of TCP/IP figured out that it doesn't make sense to reply to an ICMP which goes to the broadcast address.

SYN flood - an attacker uses a bot-net to initiate a bunch of SYN messages. The flood of SYN messages will fill up the connections buffer on the victim machine and it won't accept anymore connections, thus accomplishing a DDOS attack. Each bot-node will send a bunch of SYN requests but each request will have a different destination port.

What is a bot-net?

Cross Site Request Forgery - requirements

1. The victim host has a current session cookie to a high value webserver
2. The victim (person) can be enticed to click on a link which points to a webserver with a page that has an image tag <IMG> in the HTML which points to the high-value website and when the victim's browser tries to load what it thinks is an image, the bad code gets executed on the high-value website using the cached (still active) credentials.

What would a XSRF attack look like?

Defenses

HIDS, NIDS, host-NIDS, HIPS

Vulnerability Scanners

Qualys, OpenVAS. Authenticated and unauthenticated scans. Do an OpenVAS scan using kali.
www.sectools.org

Honeypots

Systems which are deliberately designed to attract malicious traffic. They offer fake services and fake data. The attack attempts can be recorded and analysed by the white-hats to learn more about how the black-hats are attacking them.

Server hardening

Hardening is usually the process of securing a system by reducing its surface of vulnerability, which is larger when a system performs more functions; in principle a single-function system is more secure than a multipurpose one. Reducing available ways of attack typically includes changing default passwords, the removal of unnecessary software, unnecessary usernames or logins, and the disabling or removal of unnecessary services

https://en.wikipedia.org/wiki/Hardening_(computing)

Keeping up with New Vulnerabilities

Join a mailing list or RSS feed for new CVE's. Find security alerts for your Operating System at https://lwn.net/Alerts/

Recent High Profile Vulnerabilities

Robot, GoAhead, Spectre, Meltdown, Heartbleed, ShellShock. Find their CVE numbers and look them up at *CVE database* - https://cve.mitre.org/ or *NIST database* https://nvd.nist.gov

Network Mapping

Mapping your network on a regular basis will keep you informed of what is on your network and help to spot anomalies. Explore the different types of scans offered by nmap.
https://nmap.org/book/man.html

References

https://en.wikipedia.org/wiki/Netcat
https://docs.oracle.com/cd/E64076_01/E64078/E64078.pdf
https://www.ostechnix.com/install-and-configure-dns-server-ubuntu-16-04-lts/
https://www.sans.org/security-resources/sec560/netcat_cheat_sheet_v1.pdf