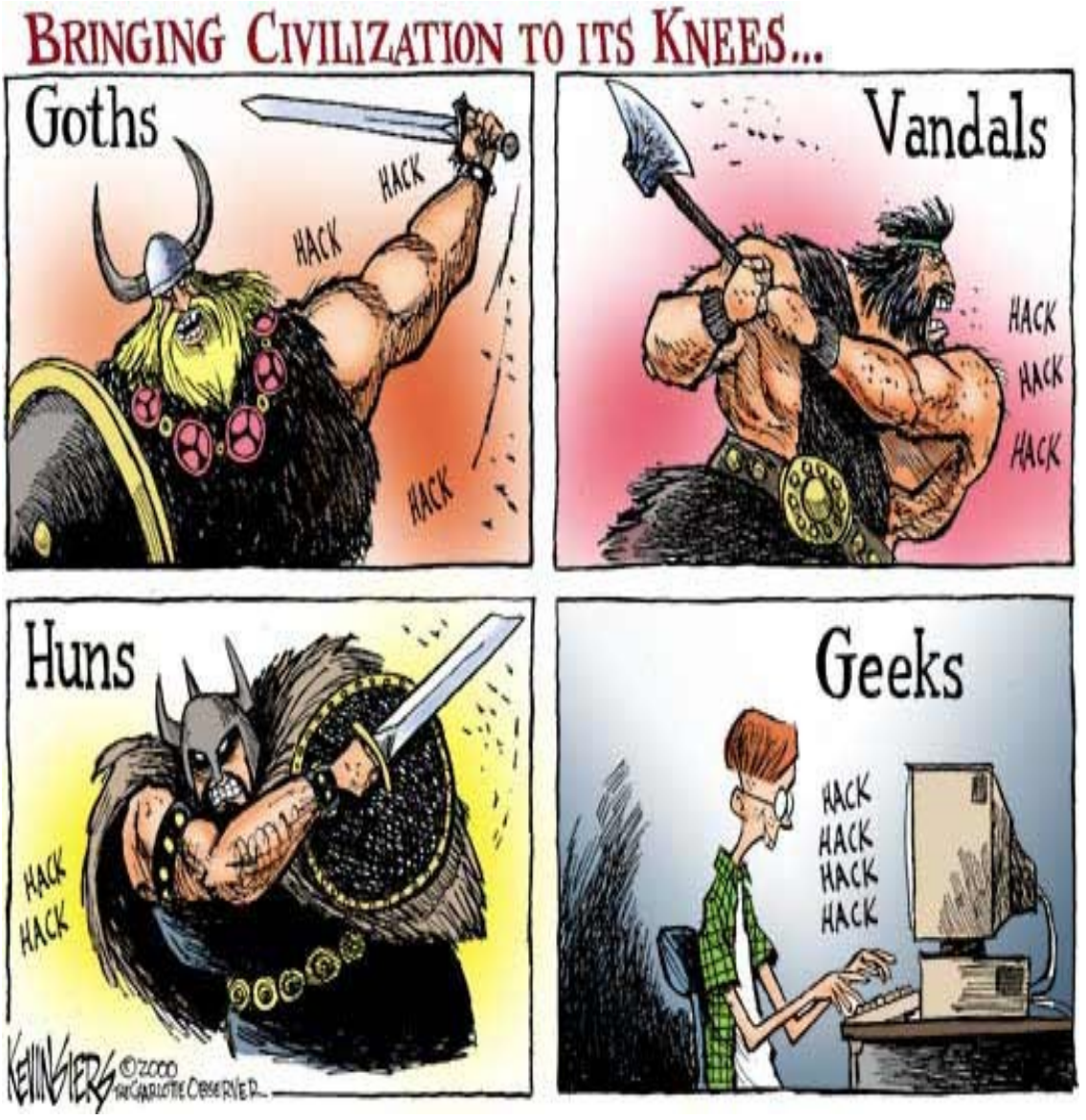
Security

Script Kiddies



Security

Network Security

Port Scans

Port scans are like a “brute force style of bulk mail” (Fyodor, The Art of Port Scanning).

Send an “advertisement” in hopes of receiving a response. A response indicates you have a live connection.

TCP port scans utilize the TCP protocol to send SYN packets to a port. If the scanner receives a SYN/ACK from the port, the port is live. The scanner sends a RST to shut down the port.

Log the “live ones” for use later.

Security

Network Security

Port Scans

First, you gotta find *suckers*

Either loop through the address space
129.186.0.1
129.186.0.2
.....

Or run a network monitoring tool like **iptraf** or **etherape** or **tcpdump** or ...

Then you have to see what entrances they've left open...

Security

Network Security

Port Scans

Many port scanners are available for use

The most popular is **nmap** and comes free with Red Hat

(http://www.insecure.org/nmap/)

Exercise:

Use **nmap** to find the open ports on your system:

/usr/bin/nmap -v -sS -O localhost

Security


Honey! I think
our network is
having another
Smurf attack!
Julie
©2000

Security

Network Security

DOS

Any incident which prevents your computer from executing its tasks is a **D**enial **O**f **S**ervice

A DOS is typically a network attack against one or more ports, rendering attempts to connect useless.

Security

Network Security

DOS

Exercise: SYN Flood

Open a second terminal (window 2)

In window 2:

```
# /bin/netstat -t -c
```

In window 1:

```
# cd /opt/exercises/Security
# ./dos localhost 22
```

Security

Network Security

DOS

Exercise: SYN Flood

Open a second terminal (window 2)

In window 2:

Kill **netstat** with CTRL-C then try logging in

/usr/bin/ssh localhost

Security

DOS
(Denial Of Service)

Other DOSes include defacing websites, spam, viruses and worms

The first worm did not attack a Microsoft or an Apple OS. It attacked UNIX computers world-wide

More dangerous are Distributed DOSes (DDOSes). One machine instructs several machines to attack one or more sites. First appeared in 2000 against Yahoo and CNN amongst others

Security


Sigh... YAHOO, ebay, AMAZON.COM, E*TRADE, BUY.COM, ZDNET, CNN... ALL SHUT DOWN BY HACKERS. WHAT AM I GOING TO DO NOW?
AH... SPEND TIME WITH YOUR FAMILY?

Security

Exploits and rootkits

A DOS is manageable.

What do you do when your system is compromised?

Rebuild and learn more about security?

Some programs are poorly written and are subject to *exploits*

Most common exploit:

buffer overrun

Security

Exploits and rootkits

Buffer overrun allows someone to cause a harmless program to start a shell by manipulating the memory

Functions, when executed, have their instructions placed on the *stack*

A piece of a stack looks like this:

buffer *sfp* *return*

[BBBBBBBBBB][xxxx][xxxx]

Security

Exploits and rootkits

buffer *sfp* *return*

[BBBBBBBBBB][xxxx][xxxx]

return is the important item here – it contains the *return address*. The return address points to the next code to be executed

Security

Exploits and rootkits

The goal of a buffer overrun is to replace the contents of the buffer with instructions to be executed and to replace the contents of the return address with the address in the buffer where our exploit code is.

So, we just take advantage of poor code that permits an overwrite of the buffer.

Poor code copies more data into the buffer than is needed. Our special return code is part of that extra data

strcpy vs **strncpy**

Security

Exploits and rootkits

If our exploit code contains the following:

```
char *name[2];
name[0] = "/bin/sh";
name[1] = 0x0;
execve(name[0], name, 0x0);
exit(0);
```

we've spawned a shell.

If the program is a daemon running as **root**, the user now has a shell **AS ROOT!**

Security

Exploits and rootkits

Once a shell has started, the user can do anything they want.

First thing to do – hide your tracks with a *rootkit*

rootkits replace programs such as **ps** and **ls** with modified binaries.

A sysadmin would normally use these to look for files and processes that shouldn't be there. The modified versions mask these files and processes

Security

Exploits and rootkits

Exercise: lrk5

cd /opt/exercises/Security/lrk5

more README

Security

Sniffing

One oldie-but-goody method is to hijack someone's password

On insecure network segments, just record or *sniff* the traffic on the segment

Save the data to a file and mine it for passwords

Two good packages for sniffing network traffic:

tcpdump ethereal (wireshark)

Security

Sniffing

With switched segments, this is becoming much harder to do

The switch monitors connections and makes sure that two connected sockets only talk to each other

But this can be compromised by *man-in-the-middle* attacks, where one machine impersonates another and intercepts and returns its packets

Security

Sniffing

Exercise: Telnet, ssh and Wireshark

Make your telnet service insecure:

nano /etc/xinetd.d/telnet

(Change disable=yes to disable=no)

service xinetd restart

Security

Sniffing

Exercise: Telnet and Wireshark

Capture network packets:

/usr/bin/wireshark &

Choose **Capture->Options** from menu

Choose interface l0

Click on **Start** button

Security

Sniffing

Exercise: Telnet, ssh and Wireshark

Capture network packets:

telnet localhost

*Login as **root***

ls /

CTRL-D

ssh root@localhost

Enter root password

ls /

CTRL-D

Choose **Capture -> STOP** in menu

Security

Network Security

Sniffing

Exercise: Telnet and Wireshark

Analyze the packets:

Click on **Protocol** column heading to sort

Click on a line that has **TELNET** as its protocol

Click on **Analyze->Follow TCP Stream**

VOILA!

Security

Network Security

Sniffing

Exercise: ssh and Wireshark

Analyze the packets:

Click on **Clear** to clear the packet filtering

Click on a line that has **ssh** as its protocol

Click on **Analyze->Follow TCP Stream**

Notice the difference?

Security

Network Security

Sniffing

Exercise: Telnet and Wireshark

Make your service secure again:

nano /etc/xinetd.d/telnet

(Change disable=no to disable=yes)

service xinetd restart