Hacking: Basic Tools and Attack Vectors

Time: 4-5:15pm
Date: Wednesday, May 5th
Where: IT Lab

Joseph Paul Cohen
Henry Z Lo
Veronica Carrillo Marquez

Overview

- Sockets/Ports
- NetCat (NC)
  - NCat
  - NMAP
  - HTTP

Sockets/Ports

Each Internet device has a series of **ports**.

Ports can communicate with each other, forming a **connection**.

There is a "listening" port, and a "connecting" port.

There are also types of ports which don't form connections.

Ports and connections are the basis of the entire Internet.

Example Ports

Computers that are servers listen on certain standard ports.

HTTP servers (i.e. the world wide web) listen on port 80.

HTTPS listens on port 443.

Secure Shell (SSH) listens on port 22.

Clients connect **from** random ports, but connect **to** predefined ones.

NetCat(nc)

NAME
  nc — arbitrary TCP and UDP connections and listens

SYNOPSIS
  nc [-46DdhklnrStUuvzC] [-i interval] [-P proxy_username]
     [-p source_port][-s source_ip_address] [-T ToS] [-w timeout]
     [-X proxy_protocol] [-x proxy_address[:port]] [hostname]
     [port[s]]

DESCRIPTION
  The nc (or netcat) utility is used for just about anything under the sun involving TCP or UDP. It can open TCP connections, send UDP packets, listen on arbitrary TCP and UDP ports, do port scanning, and deal with both IPv4 and IPv6. Unlike telnet(1), nc scripts nicely, and separates error messages onto standard error instead of sending them to standard output, as telnet(1) does with some.

NetCat(nc)

- Bind to ports (create sockets)
- Send data to and from a socket
  - Send Mode
    - nc **$host $port**
  - Receive Mode
    - nc -l **$port**
- Takes input from stdin
  - nc **$host $port** < input.txt
- Sends output to stdout
  - nc -l **$port** > output.txt

nc examples

```
$cat hello.http
HTTP/1.0 200 OK


<html>
  <body>
    <h1>Hello, world!</h1>
  </body>
</html>
```

```
$nc -l localhost 8080 < hello.http
```

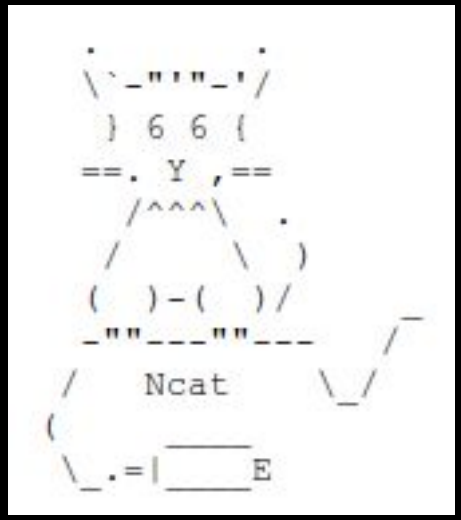
```
$nc localhost 8080
HTTP/1.0 200 OK


<html>
  <body>
    <h1>Hello, world!</h1>
  </body>
</html>
```

(ncat) Netcat for the 21st Century

Name
ncat — Concatenate and redirect sockets

Synopsis
ncat [*<OPTIONS>* ...] [*<hostname>*] [*<port>*]



Added Features

- SSL
- Command Execution
- Much More

http://nmap.org/ncat/

nc examples

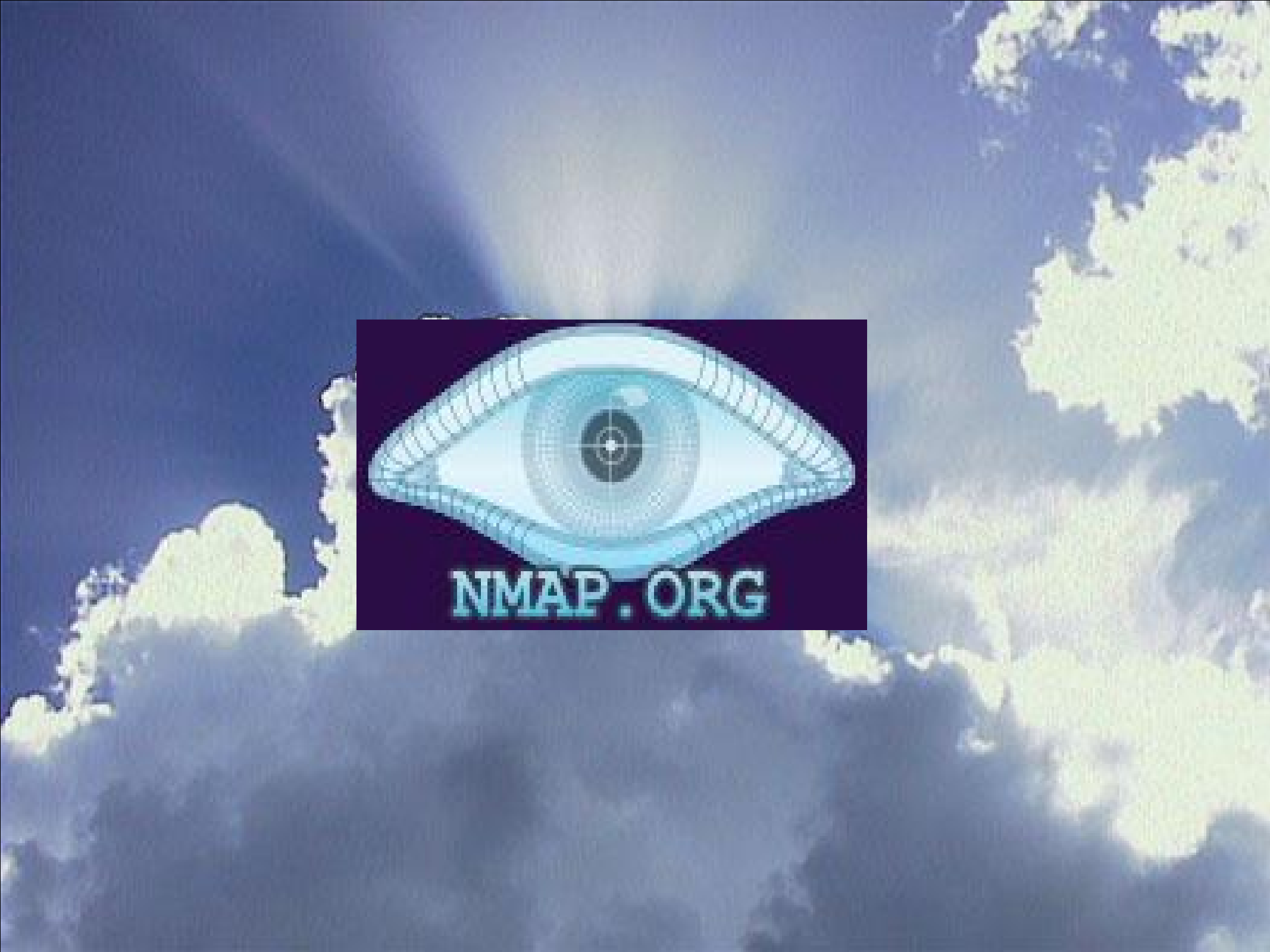
```
$ nc -v umb.edu 80
Connection to umb.edu 80 port [tcp/www] succeeded!
```

```
$ nc -v 127.0.0.2 80
nc: connect to 127.0.0.2 port 80 (tcp) failed: Connection refused
```

```
$ nc -v 192.168.0.2 80
nc: connect to 192.168.0.2 port 80 (tcp) failed: No route to host
```

```
$ nc -v 10.0.0.1 80
nc: connect to 10.0.0.1 port 80 (tcp) failed: Connection timed out
```

Wish you could try every port on every host on your subnet?


NMAP.ORG

NMap

```
nmap [Scan Type(s)] [Options] {target specification}
```

- HOST DISCOVERY
- DIFFERENT SCAN TECHNIQUES
- SERVICE/VERSION DETECTION
- OS DETECTION
- TIMING AND PERFORMANCE
- FIREWALL/IDS EVASION AND SPOOFING

Scan standard ports:

```
nmap umb.edu
```

Scan port 80 only:

```
nmap -p80 umb.edu
```

Scan all ports:

```
nmap -p- umb.edu
```

Scan with OS detection:

```
nmap -A umb.edu
```

HTTP
(Hypertext Transfer Protocol)

VERBS

- GET
- POST
- HEAD
- OPTIONS
- PUT
- DELETE
- TRACE
- CONNECT

HTTP GET

```
$nc www.cs.umb.edu 80
GET / HTTP/1.1
Host: www.cs.umb.edu
```

```
$nc google.com 80
GET /?q="gtfo"
```

```
$nc nsa.gov 80
GET / HTTP/1.1
Host: nsa.gov
```

HTTP POST

```
$nc kdl.cs.umb.edu 80
POST /w/wp-login.php HTTP/1.1
Content-Type: application/x-www-form-urlencoded
Content-Length: 120

log=joecohen&pwd=test&wp-
submit=Log+In&redirect_to=http%3A%2F%2Fkdl.cs.
umb.edu%2Fw%2Fwp-admin%2F
```