# AP Cybersecurity

AP Cybersecurity site

- Cybersec progress 9.6.26
- class notes
- 9.8.26 notes

# Cybersec progress 9.6.26

======Review so far:  ==========
We learned about physical links:

- ethernet cables
- 8 wires in pairs
- on pair is transmit (TX) one is receive (RX)
- full duplex (talking and listening at the same time)

We used terminal to learn about our computer, and to reach out to other computers:

- whoami, last, history, top
- ssh (secure shell), ping, traceroute
- nslookup, dig (names to numbers)
- commands and flags (shutdown -r now)
- privileges: root and sudo
- ctrl-c and ctrl-x (stop and halt)

We learned how to scan a network:

- IP net monitor
- arp-scan
- nmap

We learned about network architecture:

- subnets
- IP addresses
- 192.168.1.x and 10.14.x.y networks
- gateways
- domain name service (DNS)

We need to learn to navigate around the guts of another computer, locally or remotely: why?

- pwd tells you where you are (print working directory) like on a elevator, which floor you are on
- ls tells me what is on that "floor"
- ls -la tells me about even the hidden things on the floor
- ls -lha does all of this in human form
- cd means change directory (floor) on the elevator

Try this:
sudo nano <name> is how we create a file

- in terminal, navigate to your desktop folder
- create a new document called secret: sudo nano secret

- type some secret stuff there
- type ctrl-o (overwrite) and ctrl-x (exit)
- look around your folders for the secret file
- Ok, now repeat the process, creating a file called .hidden (the period makes it hidden)
- type stuff and save
- now look for the file
- ask your friends to find your files by ssh into your machine (don't tell them what you wrote)

to create a new folder (directory) we use mkdir (make a directory)

- try mkdir mystuff (no spaces, computers freak out with spaces)
- cd into that folder: cd mystuff
- make a document there
- ask your friends to find it
- look on your desktop for the folder
- create a hidden file there, see if they can find it

I have hidden a file in another computer, with a reference to yet another computer.
Your challenge is to find the files.
**Next**:

1. secure copy: scp <file> user@<ipaddress:/location>
2. networks advanced: routers

# class notes

Cybersecurity HW-quiz

9.1.26:

enable ssh on your mac

ssh into mac from your pi

ls, pwd, cd, cd ..

sudo nano

ctrl-o, ctrl-x

mkdir

wildcards *

less, more, cat

ctrl-c (cancel)

ctrl-x (halt)

before———————

ifconfig

arp-a

nmap -sn 192.168.8.0/24

whoami

last

history

hw: 9.1.26

network chuck episode #2

**challenge:**

1. find the machine named victim on the network
2. ssh in
3. list last users and history
4. show who you are
5. create a folder on the desktop in your name
6. create two files: one hidden one visible in that folder with some creative (clean) message
7. find the folder named herobrine
8. read the message and leave a message of your own in a new document

next: 9.1.26 bring headphones

linux for hackers ep02

whoami

cp

mv

rm

cat, less, more

dir structure: bin, sbin, usr->bin, sbin, boot, var, tmp, lib, dev (devices)

/home/User

root user separate in /

adduser or useradd

which (shows which one you are using)

sudo nano /etc/interfaces

gpt: “make me a python game”

————ep03

man pages

what is a terminal? vt100? bit bucket? tty?

OS->shell->terminal

bash, zsh, csh (joke)

ps

powershell=windows shell

$ = user

# = root

who vs whami

netstat

ps: process

lsof: open files

man = manual pages, space down

ctrl-c

uname -h  for help

apropos usb = keyword search

create python program using ai

solve ssh known_hosts issue

**9.3.26**

top

htop f9

kill -9

nc -l 1234

scp

sudo apt install

(dpkg next)

pip install xyz --break-system-packages

fastfetch

python3 --version

quiz:

Make sure your laptop is on the elab network for this quiz

You will use the pi for this quiz, and open a text document called "networks" on the desktop with your numbered answers, which you will save on the desktop for me to grade.

Using your pi400 server:

1. there is an intruder on the cyber5 network, find the IP and MAC address for this intruder
2. what brand of computer is this?
3. what “doors” does this intruder have open?
4. how many computers are on this network: 192.168.5.0/24
5. what is the usual router for this network?
6. what would the netmask for this look like?

next:

whoami, who, last, history

DNS and DHCP

nslookup, dig

DHCP leases

basic linux commands:

ctrl-c, z x

pwd, ls, cd

mkdir, sudo nano

ls -la

veda: adev100

leon: 6leon200

tandino: waikii20

rashad: drones34

philip: 99affe

niklas: 99hamburg

ali: scriptkiddie4

8.26.26 homework

1. what does traceroute measure?
2. run traceroute to 8.8.8.8 and describe your results

run the following commands and email your results:

ifconfig

route -n get default

ipconfig getifaddr en0

sudo arp-scan --localnet

nmap -sn 192.168.50.0/24 (change to match the network)

scan a target: nmap 192.168.50.20. (change for target)

deeper scan: nmap -sV 192.168.50.20 (change for target)

sudo nmap -O 192.168.50.20

sudo nmap -sS -sV -O 192.168.50.20

Plan: Q1

- CIA triad
- ethernet cables (the “tubes”)
- connection: 8 wires
- challenge: connection speed-iperf
- compare with fiber, wifi: security, range, speed
- homebrew: linux on owner laptops
- IPNM: gui tool
- terminal: arp -a and nmap -sn 192.168.8.0/24
- challenge: find the intruder
- nmap solo: ports
- challenge: find the doors
- network topology: /24 and /16
- IP address, gateway, subnet, DNS (more on subnet and DNS next)
- MAC address: ID of machine
- challenge: ID intruder
- OUI lookup: who is on what machine
- IPNM: bonjour scan vs address scan (scope)
- back to addresses: test at dorm, home
- ethernet adapters: usb-c for laptops, home scans on wired network: compare
- pi server stations: wifi at first, cyberwifi network (192.168.8.1 with tailscale on glinet)
- challenge: ssh into other machines, then change password (share with me)
- pi scanning tools, gui?
- challenge: scan local wLAN
- navigation: network chuck videos (bring headphones)
- basic commands: ping, traceroute
- navigation: pwd, ls, cd, mkdir, touch, nano, mv, rm
- create hidden file
- ssh challenge: find the file
- navigation: wildcards, up, back, ctrl-c  ctrl-z
- extended commands:  nslookup, dig
- DNS and DHCP, test with IPNM

- python scripts: http server
- setup apache2
- setup network: router, inbound port mapping, AP
- security: MAC filtering
- security: client monitoring
- defense: MAC spoofing, arp cache poisoning
- defense: tcpdump
- defense: wireshark intro-packets
- defend your home project: http, ssh, ftp servers
- firewalls-intro static rules
- SPI firewalls-dynamic rules
- IP addresses → MAC addresses → ports → services → scanning → attack surface
- TCP/IP conversations
- capture, decoding PCAP
- locate, watch, attack
- simple: DDOS
- encryption of traffic
- VPN-truly safe?
- LAN VPN?
- secure web traffic: http vs. https
- content filtering: ad blocking (simple)
- linux tools: Kali linux
- passwords
- hashing
- parrot OS and Kali
- MIM wireless attacks-aircrack-ng
- malware-clam AV EICAR
- log analysis with parrot os
- privacy: VPNs and TOR
- social engineering: internal vs. external vulnerabilities
- social engineering a scammer: https://www.youtube.com/watch?v=ec57eDghdvM
- privacy
- SIM swapping: https://www.youtube.com/watch?v=64p_WkYc9d0

Case studies:

1. physical vulnerability: wifi, ethernet, KB, screenlocks, drives, backups
2. LAN security: firewalls, SPI, DPI, uPNP internal holes
3. physical hacking: stuxnet, SCADA
4. prevention: firewall, air gap, defense in depth
5. human: Snowden USB, stuxnet
6. personal: spoofed email, password entropy
7. OSINT: what is available to whom and why

8. LLL analysis of three levels: personal, small shop, enterprise
9. most common attacks: internal vs. external
10. attack detection: log analysis
11. cyberlaw: now and future TGC
12. coding and scripts
13. setup an ad blocker on rPI
14. deep fakes: PBS Hany Farid, C2PA certification
15. cryptocurrency-attacks
16. iot devices-vulnerabilities
17. AI enabled attacks

cybersecurity notes 2026 page

cybesecurity is about protecting your house in a dangerous neighborhood

network: cyber

wifi: cyber

pass: goat8grass*

(only works on project room east U6lite AP)

Download and install homebrew:

xcode-select --install

/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"

Download and install IPNetMonitor:

IPNetMonitorX_2.8c1.dmg

install code:

<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE plist SYSTEM "file://localhost/System/Library/DTDs/PropertyList.dtd">

<plist version="0.9">

<dict>

<key>IPNetMonitorX_key</key>

<string>2f12c691 ef2c1a46 ed36d1c2 1ad56cfd 27f87fcb 605a3d56</string>

<key>date</key>

<date>2021-09-03T00:00:00Z</date>

<key>licensedCopies</key>

<string>1</string>

<key>name</key>

<string>Bill Wiecking</string>

<key>notice</key>

<string>This file contains a licensed registration code and is Copyright 2002 by Sustainable Softworks Inc.  Unauthorized copying and distribution is expressly prohibited.</string>

<key>organization</key>

<string></string>

</dict>

</plist>

Your neighborhood:

IP address=house address on a street

subnet mask=zip code (segments part of the area geographically)

gateway/router=street to highway

DNS (domain name server)=address directory (inside version and outside version)

ports=doors into your house

Who is in your neighborhood?

1. join the cyberwifi (goat8grass*)
2. arp -a (note the space and the “flag”)

http://localhost:1313/documents/cybersecurity-notes-2026/cybersecurity-notes-2026/

class: 8.19.26

quiz:

1. look at your notes, what was your IP, MAC address, subnet mask, dns server and gateway?
2. SCADA is both control and data-why?

finish cables-test

1. continuity pass/fail

2. speed

iperf test-speed

repeat to wifi clients-laptops why slower? duplex or simplex?

cyber neighborhood

1. macs with wifi: join cyber wifi (goat8grass*)
2. find your IP address (option wifi icon)
3. terminal
4. install xcode, homebrew
5. install IPNM
6. who is in my neighborhood? arp -a

7. MAC address and OUI lookup (wireshark page)
8. nmap -sn  192.168.4.0/24 (flags again, scope notation)
9. find a random machine, ID by MAC address (mac? pc? vendor?)
10. nmap 192.168.4.x
11. look for ports (doors) on the machine

Homework 8.19.26

Look up your IP address and use IP net monitor to run an address scan on your home network

example: IP address is 192.168.1.12

network scan would be 192.168.1.0/24

this scans 254 machines on your net

Repeat using nmap

example: my IP address is 192.168.1.12

nmap -sn 192.168.1./24

You may have to install homebrew on your mac

We will be using your data next week, so let me know if you have issues with this

Follow these up using arp -a and bonjour scan on IP net monitor

email all of your results before our next class

IPNM (phase one)

1. address scan of the wifi network
2. bonjour scan (like looking at your neighborhood from a drone)
3. DHCP test (leads to network setup)
4. DNS test (leads to nslookup and dig)

pick a pi: This is your server. Keep it alive.

every week we will learn a new attack

This is where you start:

username: pi password: pal82

SSH enabled

HTTP enabled

MQTT enabled

unnecessary service enabled

no firewall

poor permissions

no monitoring

challenge one: who is on my network? (macs: homebrew)

prep: wifi off, LAN on separate network

network scans arp -a  nmap -sN 10.14.254.0/16

find host, then nmap host, look for ports

id ports, is it a web server?

can you ssh in?

python script?

IP addresses → MAC addresses → ports → services → scanning → attack surface

hw: scan home wifi using nmap

challenge two: Id the suspect

next: getting out and around

ping, traceroute

navigation: pwd, ls, cd, mkdir, touch, nano, mv, rm

wildcards, up, back

ctrl-c  ctrl-z

create file: sudo nano, ctrl-o, ctrl-x (exit)

network chuck video?

parts: router-switch-LAN cable, wLAN AP, client

later: DNS-dig and nslookup

syslogd

dhcp

nat

inbound port mapping

Defense:

MAC spoofing

arp cache poisoning

head first networking-pdf

# 9.8.26 notes

**9.8.26**

**You should be comfortable with everything here**

- MC/FR quiz: exam in a box
- scp python code: laptop to pi
  - scp <filename> user@<ipaddress>:/<location>
- mv, cp, rm, cat, less, more
  - mv is move, also change name
  - cp is copy, from one place to another
  - rm is remove, rm -frd removes directories too
  - cat, less and more look inside files
- useradd
- **su root**
- **sudo nano /etc/interfaces**
- man pages
- **$ vs # prompt**
- **powershell=windows shell**
- **uname -h for help**
- **apropos usb = keyword search**
- apt install (see below)
- dpkg -i

**wget: installing stuff**

- wget http://ftp.us.debian.org/debian/pool/main/c/cmatrix/cmatrix_2.0-6+b1_arm64.deb
- ls -lh *.deb
- dpkg-deb --info cmatrix_*.deb
- sudo dpkg -i cmatrix_*.deb
- cmatrix

**using apt:**

- sudo apt update
- sudo apt install sl
- sl

**What's the difference?**

- wget → download something from the Internet

- .deb → Debian software package
- dpkg -i → install a local package
- cmatrix → run what we installed
- dpkg -L → see what files were installed

next: network structure review

- permissions
- htop
- firewalls-routers
- tcp/ip stack
- dns
- dhcp
- /etc/host file
- NAT
- inbound port mapping