CSE 127
Discussion 5

PA4: Network Attacks

- Scavenger hunt! You need to find Deepak’s “password reset token”
- You should be receiving a tar.gz file in your email
    - This will be the starting point
    - Subject: [CSE 127] PA4 Flash Drive Dump
    - From: ta_admin@bungle.sysnet.ucsd.edu
- Please make a piazza post if you did not receive a tar.gz file in your email
    - Could also email me if I don’t get back to you in time

- START EARLY! You could be stuck for a while if you don’t know what to do, and it can be hard to estimate how much further you still have to go. If you’re stuck, office hours are always a great place to go for a nudge in the right direction.

PA4: Network Attacks

This is a puzzle hunt - it’s not as fun or useful as an exercise if you spoil the later stages for others.

Please use *private* Piazza posts for this PA.

Logistics

- Deadline: Thursday, 2/26 at 11:59pm

- Submit to the Autograded Gradescope assignment:
  - Part A: Mystery
    - What to submit will be revealed in the middle of the PA
  - Part B: Token
    - Submit a single file named "token"
    - If you are in a group, you can submit any group member’s token
    - If the autograder says that you’ve submitted someone else’s token:
      - **Don’t panic!** We won’t assume you’ve cheated - there are legitimate ways that students can end up with others’ tokens.
      - Go to office hours or make a private post on Piazza, describe what you’ve done so far, and ask for advice
      - You will not receive credit for completing the scavenger hunt if you only submit the incorrect token

- Submit to the PDF Gradescope assignment:
  - Part C: Writeup
    - TEXT File describing your steps and thought process towards the mystery item and the token
    - Writing guidelines: Imagine your boss has instructed you to figure out how to complete this scavenger hunt, and, once you’ve succeeded, to write instructions for your coworkers so they can replicate your steps

Tools for PA4

- nc - Allows you to make connections locally
- nmap - Scan ports/IPs (locally and externally)
- ssh - Connect to servers over shell
- tcpdump - View network traffic on machine
- wget/curl - Download files from the internet

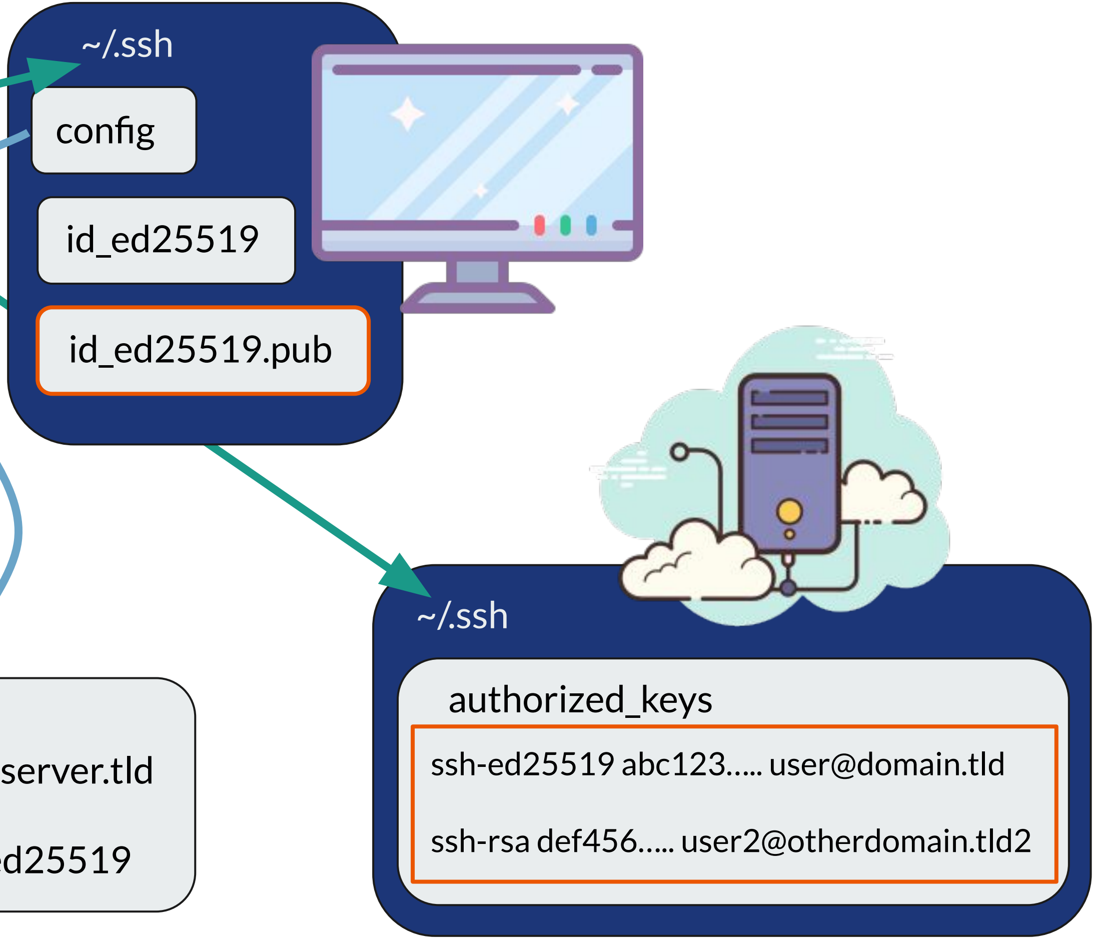
Check out all their "man" pages

Try to find the commands as well as the options that give you exactly what you need

- ~/ means “this user’s home directory”
- Files and directories that start with **.** are hidden by default

ssh

- “Secure shell” - allows you to securely open a terminal on another host
  - Not always a whole other computer! Remember ssh-ing into your PA1 and PA2 VMs?
- Uses public & private keys



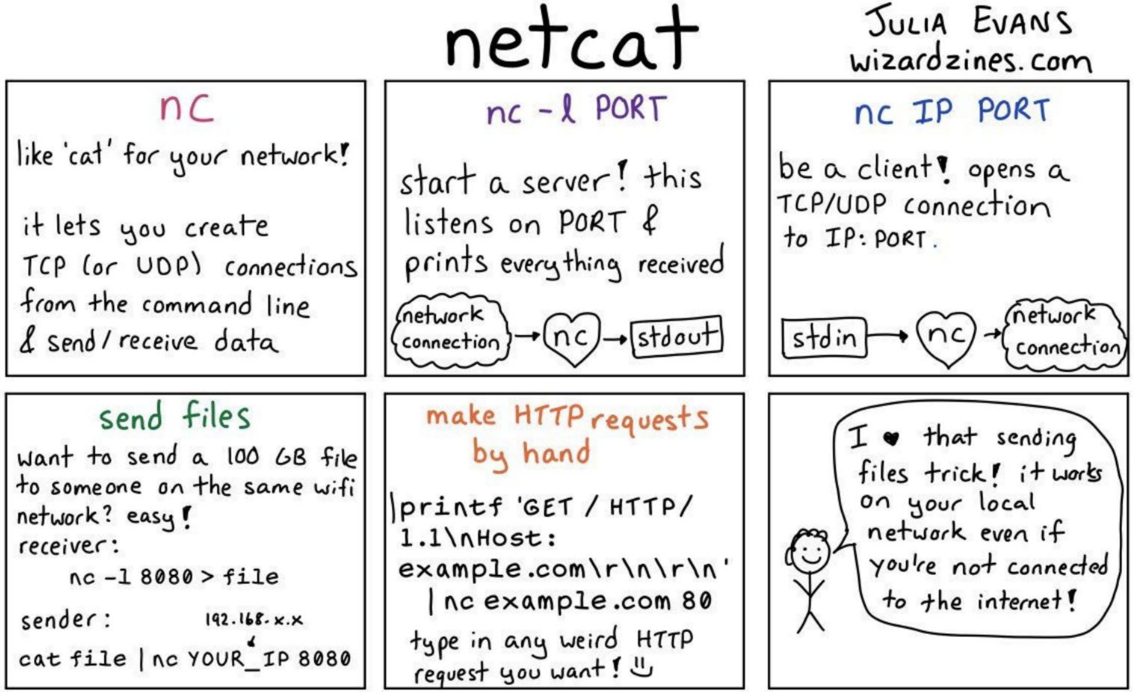
```
ssh-ed25519 abc123..... user@domain.tld

ssh-rsa def456..... user2@otherdomain.tld2
```

```
Host my-server
        HostName my.remote.server.tld
        User alice
        IdentityFile ~/.ssh/id_ed25519
```

Port, ProxyJump, ForwardAgent...

tcpdump

- Used to display TCP/IP and other packets that are transported over a network the machine is in
- Reading the tcpdump of a machine can be very noisy
  - Use "tcpdump -D" to see what interfaces are available
  - Specify an interface with the "-i" option
- By default, tcpdump only looks at packet header information
  - If you wish to view the packet contents, you must use the "-X" or "-A" options



netcat

Julia Evans
wizardzines.com

nc

like 'cat' for your network!

it lets you create TCP (or UDP) connections from the command line & send / receive data

nc -l PORT

start a server! this listens on PORT & prints everything received

nc IP PORT

be a client! opens a TCP/UDP connection to IP: PORT.

send files

want to send a 100 GB file to someone on the same wifi network? easy!

receiver:

```
nc -l 8080 > file
```

sender:

192.168.x.x

```
cat file | nc YOUR_IP 8080
```

make HTTP requests by hand

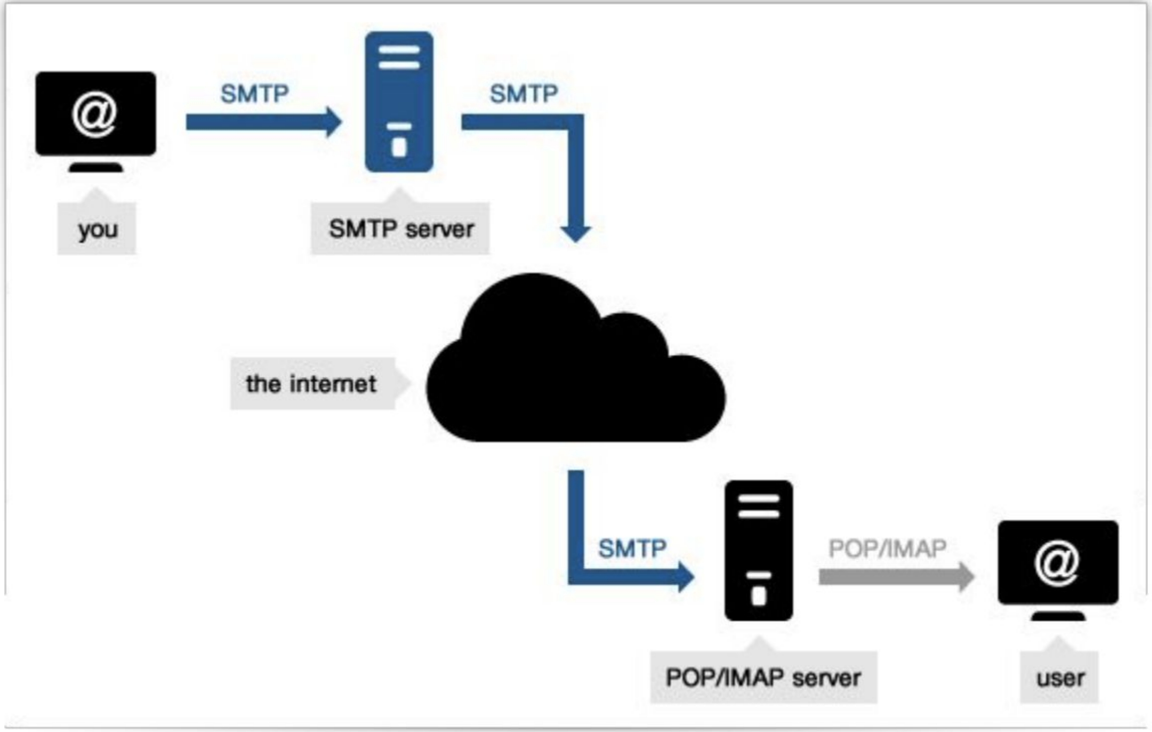
```
printf 'GET / HTTP/
1.1\nHost:
example.com\r\n\r\n'
  | nc example.com 80
```

type in any weird HTTP request you want! ☺

I ♥ that sending files trick! it works on your local network even if you're not connected to the internet!

SMTP

- Simple Mail Transfer Protocol
- A protocol for sending mail
- SMTP servers commonly use TCP on port 25
- SMTPS (S for secure) is often on port 465 as well
- The remote machine for this PA has a smtp server setup


SMTP
SMTP
you
SMTP server
the internet
SMTP
POP/IMAP
POP/IMAP server
user

SMTP Commands

- MAIL FROM
- RCPT TO
- DATA
  - From
  - To
  - Reply-To
  - Date
  - Subject
  - …
  - The Message

SMTP Fields

- **FROM**: this is the field that indicates where the mail is from. This is our traditional notion of who the mail's sender is
- **REPLY-TO**: This is added by the sender to indicate where human replies should be addressed to. When you press the "Reply" button on, say, your Gmail client, the email in this field will show up as you compose your reply

SMTP Fields: MAIL FROM vs. FROM

- **MAIL FROM** and **RCPT TO** are both fields in the “envelope” of the email address whereas FROM and other fields are in the “letter” of the email
- **MAIL FROM** is the one used by SMTP servers to transport the mail
- But when it shows up in the client, typically the envelope is discarded and only the FROM is shown

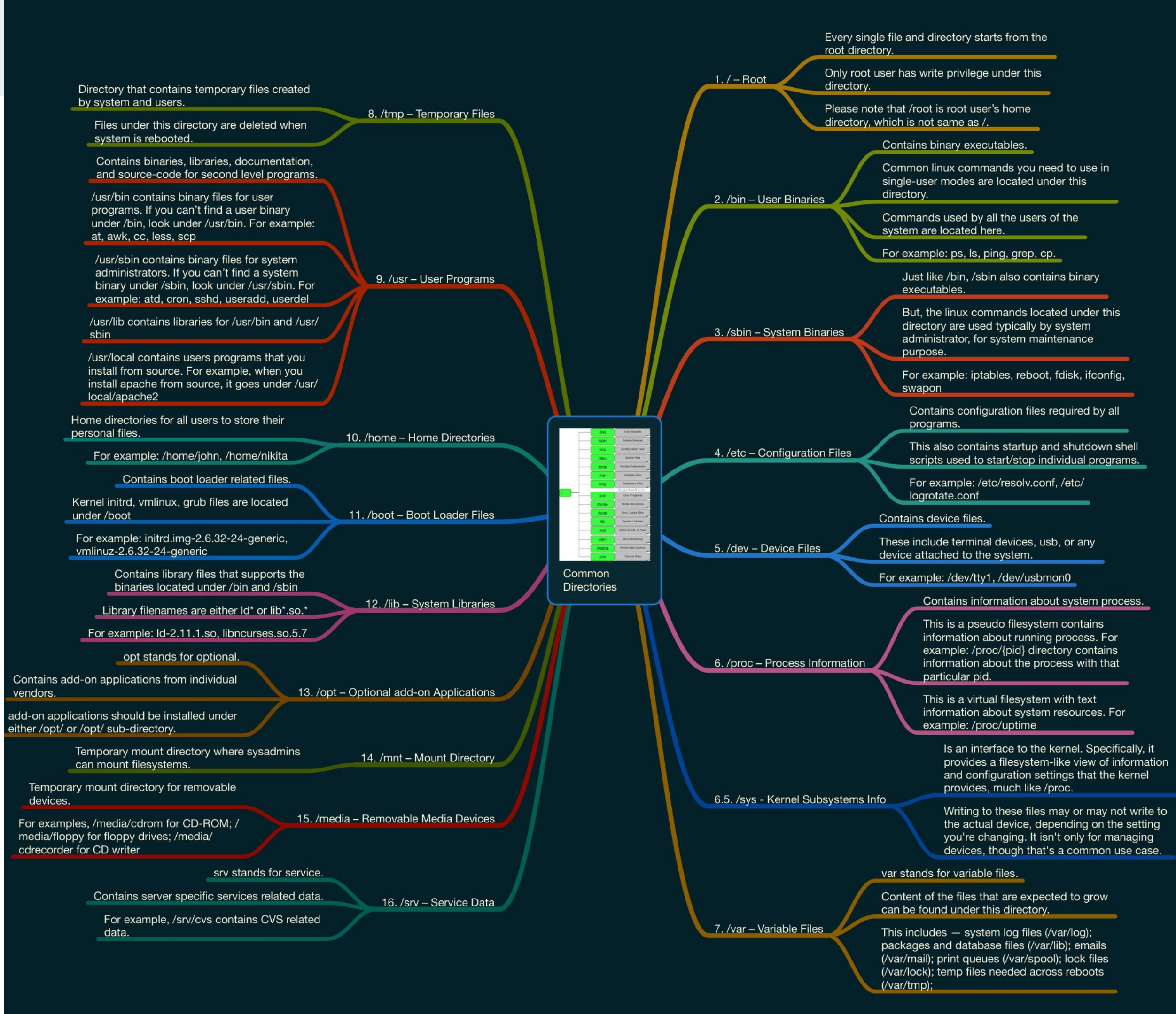
SMTP Server

“Relay access denied”: you are using the wrong SMTP server or entered the wrong domain

But how do I figure out what the SMTP server is?

I've given you a useful free clue: the SMTP server is part of the PA infrastructure, and is relatively isolated from the internet.

Another free hint: check out https://github.com/dformoso/linux-admin



Common Directories

1. / – Root
   - Every single file and directory starts from the root directory.
   - Only root user has write privilege under this directory.
   - Please note that /root is root user's home directory, which is not same as /.
2. /bin – User Binaries
   - Contains binary executables.
   - Common linux commands you need to use in single-user modes are located under this directory.
   - Commands used by all the users of the system are located here.
   - For example: ps, ls, ping, grep, cp.
3. /sbin – System Binaries
   - Just like /bin, /sbin also contains binary executables.
   - But, the linux commands located under this directory are used typically by system administrator, for system maintenance purpose.
   - For example: iptables, reboot, fdisk, ifconfig, swapon
4. /etc – Configuration Files
   - Contains configuration files required by all programs.
   - This also contains startup and shutdown shell scripts used to start/stop individual programs.
   - For example: /etc/resolv.conf, /etc/logrotate.conf
5. /dev – Device Files
   - Contains device files.
   - These include terminal devices, usb, or any device attached to the system.
   - For example: /dev/tty1, /dev/usbmon0
6. /proc – Process Information
   - Contains information about system process.
   - This is a pseudo filesystem contains information about running process. For example: /proc/{pid} directory contains information about the process with that particular pid.
   - This is a virtual filesystem with text information about system resources. For example: /proc/uptime

6.5. /sys - Kernel Subsystems Info
   - Is an interface to the kernel. Specifically, it provides a filesystem-like view of information and configuration settings that the kernel provides, much like /proc.
   - Writing to these files may or may not write to the actual device, depending on the setting you're changing. It isn't only for managing devices, though that's a common use case.

7. /var – Variable Files
   - var stands for variable files.
   - Content of the files that are expected to grow can be found under this directory.
   - This includes — system log files (/var/log); packages and database files (/var/lib); emails (/var/mail); print queues (/var/spool); lock files (/var/lock); temp files needed across reboots (/var/tmp);
8. /tmp – Temporary Files
   - Directory that contains temporary files created by system and users.
   - Files under this directory are deleted when system is rebooted.
9. /usr – User Programs
   - Contains binaries, libraries, documentation, and source-code for second level programs.
   - /usr/bin contains binary files for user programs. If you can't find a user binary under /bin, look under /usr/bin. For example: at, awk, cc, less, scp
   - /usr/sbin contains binary files for system administrators. If you can't find a system binary under /sbin, look under /usr/sbin. For example: atd, cron, sshd, useradd, userdel
   - /usr/lib contains libraries for /usr/bin and /usr/sbin
   - /usr/local contains users programs that you install from source. For example, when you install apache from source, it goes under /usr/local/apache2
10. /home – Home Directories
    - Home directories for all users to store their personal files.
    - For example: /home/john, /home/nikita
11. /boot – Boot Loader Files
    - Contains boot loader related files.
    - Kernel initrd, vmlinux, grub files are located under /boot
    - For example: initrd.img-2.6.32-24-generic, vmlinuz-2.6.32-24-generic
12. /lib – System Libraries
    - Contains library files that supports the binaries located under /bin and /sbin
    - Library filenames are either ld* or lib*.so.*
    - For example: ld-2.11.1.so, libncurses.so.5.7
13. /opt – Optional add-on Applications
    - opt stands for optional.
    - Contains add-on applications from individual vendors.
    - add-on applications should be installed under either /opt/ or /opt/ sub-directory.
14. /mnt – Mount Directory
    - Temporary mount directory where sysadmins can mount filesystems.
15. /media – Removable Media Devices
    - Temporary mount directory for removable devices.
    - For examples, /media/cdrom for CD-ROM; /media/floppy for floppy drives; /media/cdrecorder for CD writer
16. /srv – Service Data
    - srv stands for service.
    - Contains server specific services related data.
    - For example, /srv/cvs contains CVS related data.

Email Spoofing

- Helpful links:
  - https://en.wikipedia.org/wiki/Email_spoofing
  - https://www.linuxjournal.com/content/sending-email-netcat
- Pretend to be a legit sender
- Phishing and spam

```
FROM: Automated Helper <donotreply@service.com>

TO: Susie Queue <susie.queue@gmail.com>

REPLY-TO: <help@company.com>

Hi! Here’s a helpful automated notification about
your account at Company. Beep boop beep! This
notification was powered by Service :)
```

```
FROM: Bank of America <accounts@bankofamerica.com>

TO: Susie Queue <susie.queue@gmail.com>

REPLY-TO: <bad@hacker.evil>

We have frozen your account to investigate claims of
fraud.

Please reply with your Social Security Number to
verify your identity.
```

SMTP protocol does not require that these fields match, or even be from the same domain!

This is useful in some cases, and *exploitable* in other cases.

nmap

- When a machine is “listening”, such as a web server listening for new clients who might request a web page or other resource, that “listening” is associated with a specific port number, e.g. 22, 24, 80, 443
- Some specific port numbers are typically used for specific protocols (port 53 for DNS, port 22 for SSH, etc), but this is not strict
- nmap is a “port scanner” - it tells you which ports are listening for requests on a specific machine
- In the real world, ports must only be opened to the global internet *very* carefully, in *very* limited ways

wget

- Usage: wget [OPTIONS] [URL]
- http://[some ip address or domain]:[port number] or https://[some ip address or domain]:[port number]
    - http vs https is important for some server configurations!
- Try --no-check-certificate if you get a connection error
    - Connection error is different from connection refused

Misc

- SSH permission too open on WSL: try to ssh from Windows directly
- If you are stuck staring at some chat log, re-read what “y2shu” says
- Don’t send email to anyone’s real email (e.g. xxx@ucsd.edu). The email addresses used in this PA are local and the format is for you to figure out.
    - You will get points taken off for using external email addresses
- Do not use tcpdump to look for the reply to any email you might send. You will be given an in-terminal notification if you have mail. You can also use the “mail” command to check your inbox.

Being stuck

If you are stuck, please use PRIVATE piazza posts. If we think that they are general enough to be shared, we will make them public.

Clues

Please watch the last discussion video and look closely at the slides. They are especially relevant and many people coming to office hours and asking questions have not looked at them yet. They are probably VERY useful.

The clues are also in chronological order from the previous discussion

SSH

If you are unfamiliar with ssh-key generation, the idea is that you create a public and private key that correspond to each other. You send the public key to the server, and then later use the private key to gain access.

Look into using logging into SSH without using password and just using a key pair.

Transcript (possible spoilers!)

For the transcript (if you haven’t gotten here, it's fine this isn't really a spoiler): we want your user input and the response of the server. This is the FULL response of the server, from the initial 220 handshake response to the 221 Bye response.

No getting an email in response

If you are not getting an email, you have either

1) formatted or are missing parts of the SMTP request, or

2) (which is more common) not completed the request to a Bye/Quit stage.

3) content / subject of the email doesn’t have the required keywords

I know we have a habit of Ctrl+Z / Ctrl+C ing everything, but this can cause issues.

No personal emails

You should not be using any personal email addresses for this assignment