

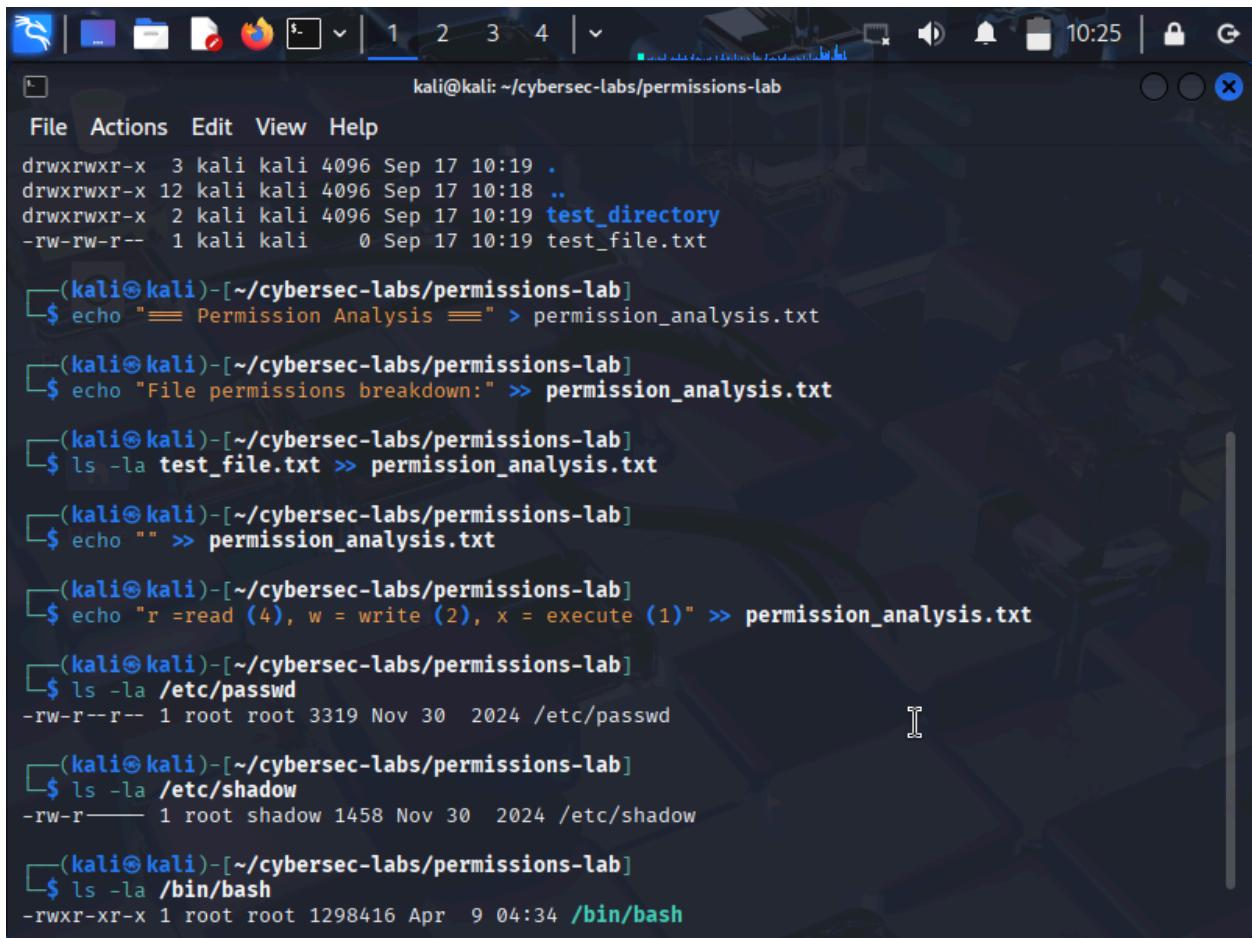
Lab 4: Linux Basics Part 4 - Permissions and File Discovery

Matthew Cox, COMP325-01, term (Fall 2025)

Introduction:

This lab will focus on Linux file permissions, analyzing SUID/SGID files, and locating files and directories. I will gain a greater understanding of how hackers might exploit misconfigured files. At the same time, I will be learning more in-depth about how Security Analysts need these skills to investigate incidents.

Body:



The screenshot shows a terminal window titled "kali@kali: ~/cybersec-labs/permissions-lab". The terminal displays a series of commands used for permission analysis. It starts by listing the contents of the current directory, which includes a directory named "test_directory" and a file named "test_file.txt". The user then creates a log file named "permission_analysis.txt" and adds several lines of text to it, including a header and descriptions of file permissions. Finally, the user runs commands to analyze the permissions of system files like "/etc/passwd" and "/etc/shadow", and a binary file like "/bin/bash".

```
drwxrwxr-x 3 kali kali 4096 Sep 17 10:19 .
drwxrwxr-x 12 kali kali 4096 Sep 17 10:18 ..
drwxrwxr-x 2 kali kali 4096 Sep 17 10:19 test_directory
-rw-rw-r-- 1 kali kali 0 Sep 17 10:19 test_file.txt

[(kali㉿kali)-~/cybersec-labs/permissions-lab]
$ echo "==== Permission Analysis ===" > permission_analysis.txt

[(kali㉿kali)-~/cybersec-labs/permissions-lab]
$ echo "File permissions breakdown:" >> permission_analysis.txt

[(kali㉿kali)-~/cybersec-labs/permissions-lab]
$ ls -la test_file.txt >> permission_analysis.txt

[(kali㉿kali)-~/cybersec-labs/permissions-lab]
$ echo "" >> permission_analysis.txt

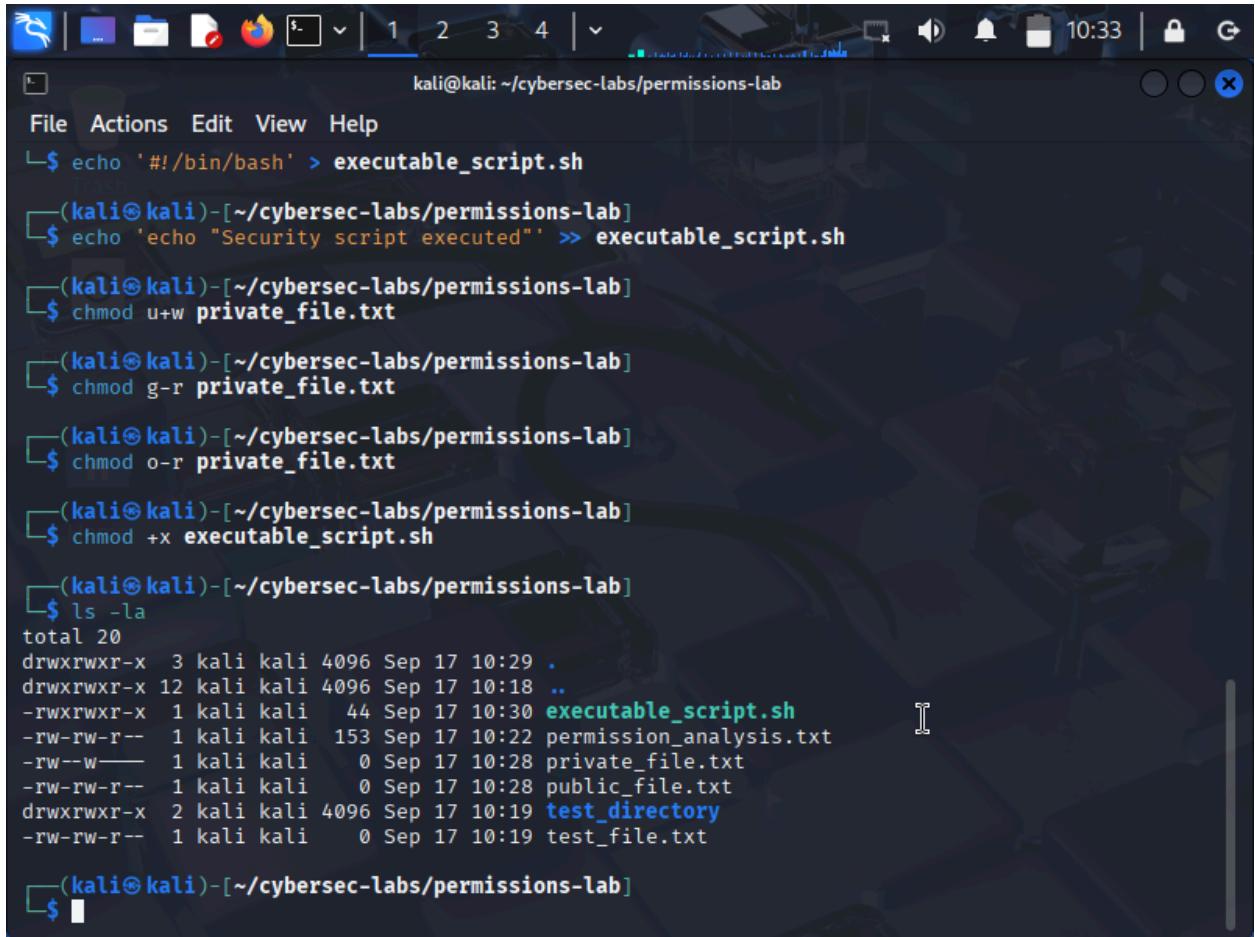
[(kali㉿kali)-~/cybersec-labs/permissions-lab]
$ echo "r =read (4), w = write (2), x = execute (1)" >> permission_analysis.txt

[(kali㉿kali)-~/cybersec-labs/permissions-lab]
$ ls -la /etc/passwd
-rw-r--r-- 1 root root 3319 Nov 30 2024 /etc/passwd

[(kali㉿kali)-~/cybersec-labs/permissions-lab]
$ ls -la /etc/shadow
-rw-r----- 1 root shadow 1458 Nov 30 2024 /etc/shadow

[(kali㉿kali)-~/cybersec-labs/permissions-lab]
$ ls -la /bin/bash
-rwxr-xr-x 1 root root 1298416 Apr 9 04:34 /bin/bash
```

Here, I was just setting the start of this lab. I ran a couple of test files and created a log to get a grasp of permission notation. Then, I ran some commands to view the system file permissions.



The screenshot shows a terminal window titled "kali@kali: ~/cybersec-labs/permissions-lab". The terminal displays a series of commands demonstrating file permissions:

```
File Actions Edit View Help
└$ echo '#!/bin/bash' > executable_script.sh
└(kali㉿kali)-[~/cybersec-labs/permissions-lab]
└$ echo "Security script executed" >> executable_script.sh
└(kali㉿kali)-[~/cybersec-labs/permissions-lab]
└$ chmod u+w private_file.txt
└(kali㉿kali)-[~/cybersec-labs/permissions-lab]
└$ chmod g-r private_file.txt
└(kali㉿kali)-[~/cybersec-labs/permissions-lab]
└$ chmod o-r private_file.txt
└(kali㉿kali)-[~/cybersec-labs/permissions-lab]
└$ chmod +x executable_script.sh
└(kali㉿kali)-[~/cybersec-labs/permissions-lab]
└$ ls -la
total 20
drwxrwxr-x  3 kali kali 4096 Sep 17 10:29 .
drwxrwxr-x 12 kali kali 4096 Sep 17 10:18 ..
-rwxrwxr-x  1 kali kali   44 Sep 17 10:30 executable_script.sh
-rw-rw-r--  1 kali kali 153 Sep 17 10:22 permission_analysis.txt
-rw--w----
```

I created some permission examples. Then, I used symbolic notations with the command chmod.

```
kali@kali: ~/cybersec-labs/permissions-lab
File Actions Edit View Help

[(kali㉿kali)-~/cybersec-labs/permissions-lab]
$ cat public_file.txt

[(kali㉿kali)-~/cybersec-labs/permissions-lab]
$ echo "Secret data" > private_file.txt

[(kali㉿kali)-~/cybersec-labs/permissions-lab]
$ cat private_file.txt
Secret data

[(kali㉿kali)-~/cybersec-labs/permissions-lab]
$ echo "password=admin123" > config_file.txt

[(kali㉿kali)-~/cybersec-labs/permissions-lab]
$ chmod 644 config_file.txt

[(kali㉿kali)-~/cybersec-labs/permissions-lab]
$ echo "#!/bin/bash" > backup_script.sh
zsh: event not found: /bin/bash

[(kali㉿kali)-~/cybersec-labs/permissions-lab]
$ echo "cp /etc/passwd /tmp/backup/" >> backup_script.sh

[(kali㉿kali)-~/cybersec-labs/permissions-lab]
$ chmod 777 backup_script.sh

[(kali㉿kali)-~/cybersec-labs/permissions-lab]
$ ls -la config_file.txt backup_script.sh
-rwxrwxrwx 1 kali kali 28 Sep 17 10:38 backup_script.sh
-rw-r--r-- 1 kali kali 18 Sep 17 10:37 config_file.txt
```

I used numeric notations with chmod. Afterwards, I used some text permission effects. Then, I created some security-relevant scenarios. I'm going to document the permission analysis and examine some file ownerships.

The screenshot shows a terminal window titled "kali@kali: ~/cybersec-labs/permissions-lab". The terminal displays a series of commands and their outputs related to file permissions:

```
(kali㉿kali)-[~/cybersec-labs/permissions-lab]
$ sudo chown root:root config_file.txt

(kali㉿kali)-[~/cybersec-labs/permissions-lab]
$ ls -la config_file.txt
-rw-r--r-- 1 root root 18 Sep 17 10:37 config_file.txt

(kali㉿kali)-[~/cybersec-labs/permissions-lab]
$ sudo chgrp users public_file.txt

(kali㉿kali)-[~/cybersec-labs/permissions-lab]
$ ls -ls public_file.txt
0 -rw-r--r-- 1 kali users 0 Sep 17 10:28 public_file.txt

(kali㉿kali)-[~/cybersec-labs/permissions-lab]
$ echo "admin_password=secret123" > admin_config.txt

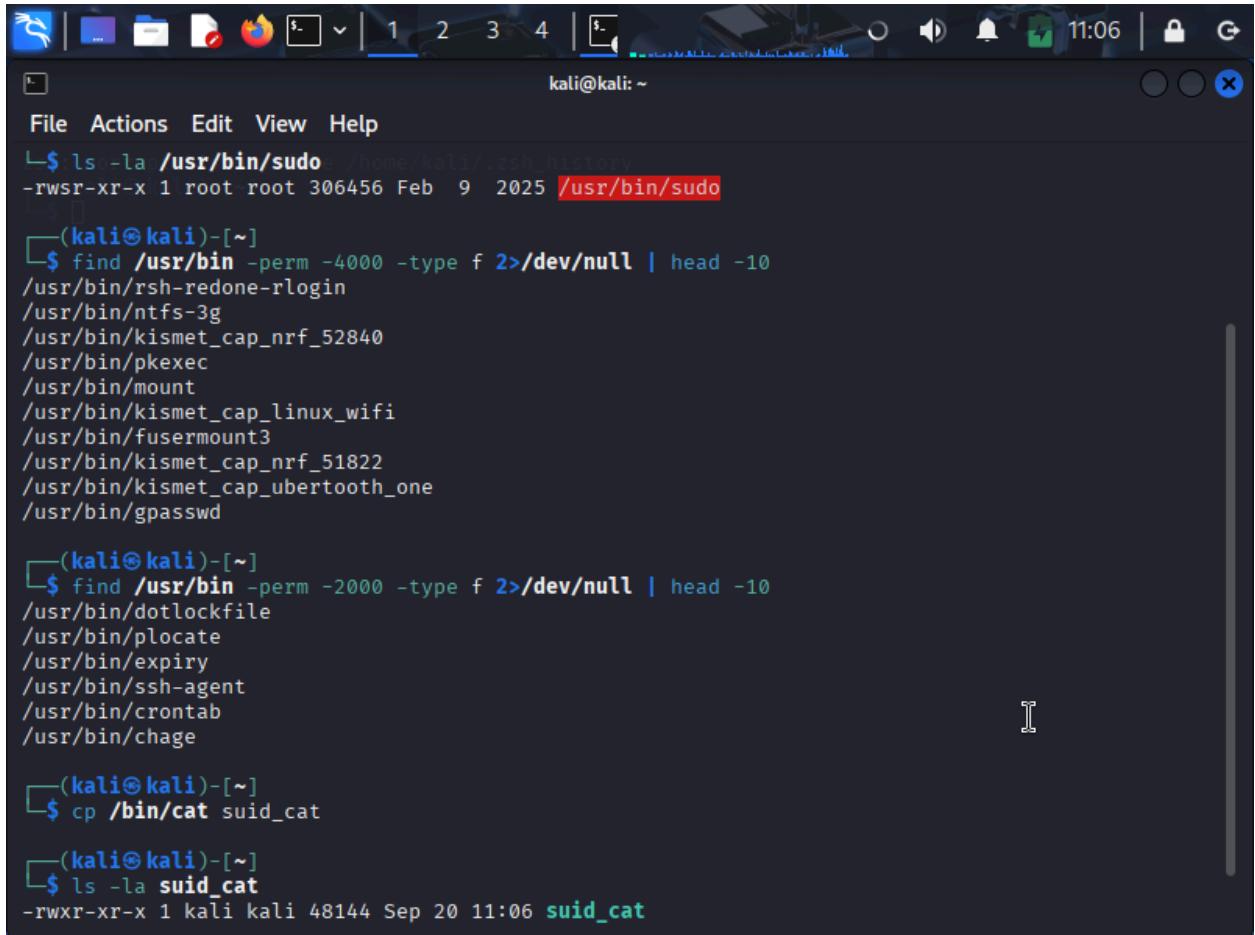
(kali㉿kali)-[~/cybersec-labs/permissions-lab]
$ sudo chown root:root admin_config.txt

(kali㉿kali)-[~/cybersec-labs/permissions-lab]
$ sudo chmod 600 admin_config.txt

(kali㉿kali)-[~/cybersec-labs/permissions-lab]
$ echo "user_data" > user_file.txt

(kali㉿kali)-[~/cybersec-labs/permissions-lab]
$ ls -la *config.txt user_file.txt
-rw----- 1 root root 25 Sep 17 10:47 admin_config.txt
-rw-rw-r-- 1 kali kali 10 Sep 17 10:49 user_file.txt
```

Initially, I examined the current ownership. After, I changed the ownership as well as the group ownership. Then I created a few ownership scenarios. Next, I will be making an ownership analysis. That will then cover this section. For then I will gain a greater understanding of the sudo command in the next exercise.



A screenshot of a Kali Linux terminal window. The terminal shows a shell session with the user kali@kali:~. The command history at the bottom of the window includes:

```
ls -la /usr/bin/sudo
-rwsr-xr-x 1 root root 306456 Feb  9 2025 /usr/bin/sudo

find /usr/bin -perm -4000 -type f 2>/dev/null | head -10
/usr/bin/rsh-redone-rlogin
/usr/bin/ntfs-3g
/usr/bin/kismet_cap_nrf_52840
/usr/bin/pkexec
/usr/bin/mount
/usr/bin/kismet_cap_linux_wifi
/usr/bin/fusermount3
/usr/bin/kismet_cap_nrf_51822
/usr/bin/kismet_cap_ubertooh_one
/usr/bin/gpasswd

find /usr/bin -perm -2000 -type f 2>/dev/null | head -10
/usr/bin/dotlockfile
/usr/bin/plocate
/usr/bin/expiry
/usr/bin/ssh-agent
/usr/bin/crontab
/usr/bin/chage

cp /bin/cat suid_cat
ls -la suid_cat
```

Here, I have gotten familiar with SUID and SGID. I then created a SUID example as a demonstration.

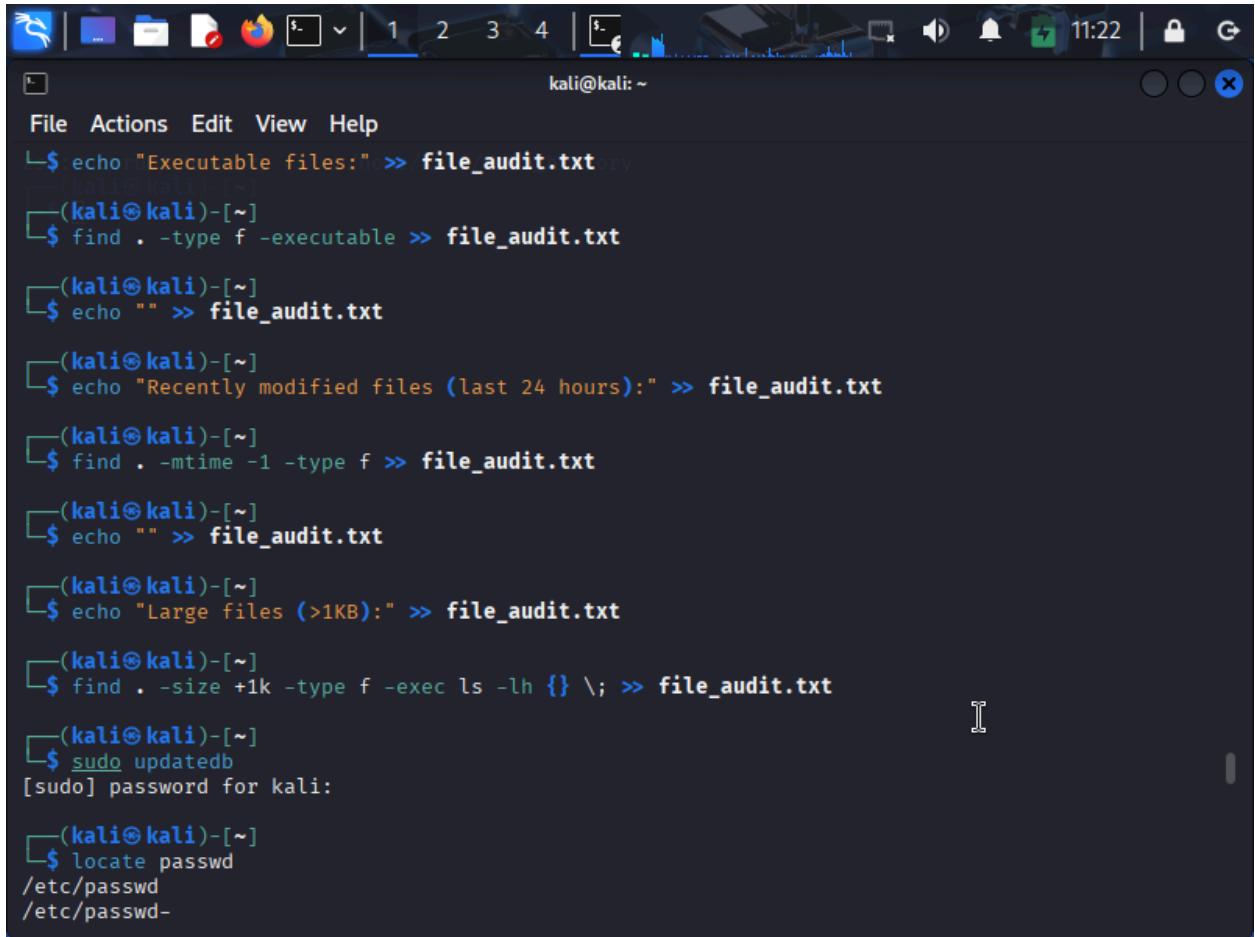
```
File Actions Edit View Help
./discover/notes/git.txt  /home/kali/.zsh_history
./discover/notes/snmp.txt
./discover/notes/PowerShell/PS-Attack.txt
./discover/notes/PowerShell/basic.txt
./discover/notes/PowerShell/PowerView.txt
./discover/notes/PowerShell/Empire.txt
./discover/notes/bash.txt
./discover/notes/terraform.txt
./discover/notes/osx.txt
./discover/notes/buffer-overflows.txt
./discover/notes/mobile.txt
./discover/notes/exploits.txt
./discover/notes/shodan.txt
./discover/notes/metasploit.txt
./discover/notes/active-directory.txt
./discover/notes/ssl.txt

└─(kali㉿kali)-[~]
└─$ find . -type f -name "*config"
./cybersec-labs/permissions-lab/config_file.txt
./cybersec-labs/permissions-lab/admin_config.txt
./.config/cherrytree/config.cfg
./discover/.git/config
./discover/report/pages/config.htm

└─(kali㉿kali)-[~]
└─$ find . -type d -name "*test*"
./cybersec-labs/permissions-lab/test_directory

└─(kali㉿kali)-[~]
└─$
```

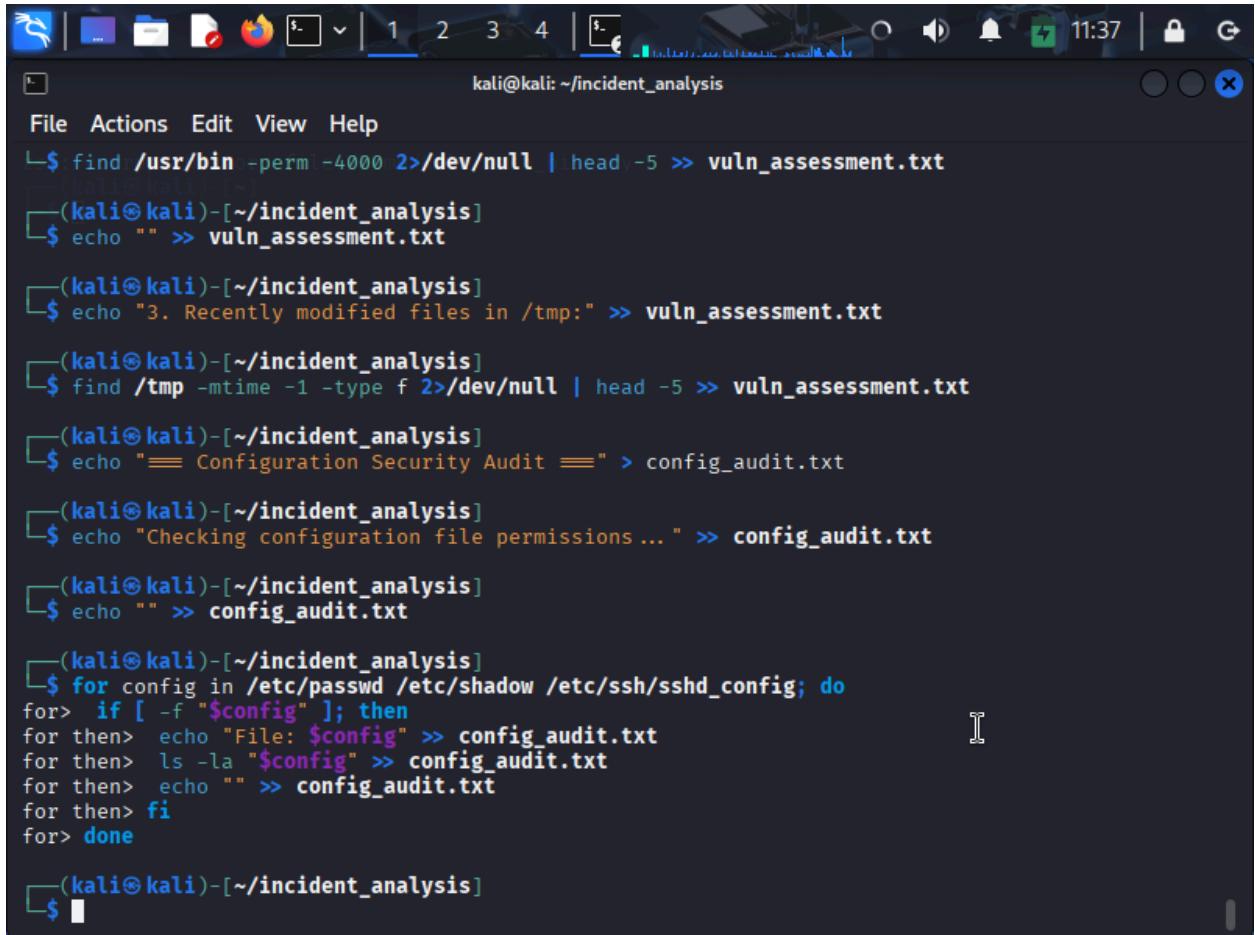
Out of frame, I created a SUID and SGID analysis. Now, I used some basic find operations.



A screenshot of a terminal window titled "kali@kali: ~". The terminal shows a series of commands being run to audit files on a system. The commands include echoing text into a file, finding executable files, listing recently modified files, and finding large files. It also includes a sudo command to update a database and a locate command to search for 'passwd'.

```
$ echo "Executable files:" >> file_audit.txt
[(kali㉿kali)-~]
$ find . -type f -executable >> file_audit.txt
[(kali㉿kali)-~]
$ echo "" >> file_audit.txt
[(kali㉿kali)-~]
$ echo "Recently modified files (last 24 hours):" >> file_audit.txt
[(kali㉿kali)-~]
$ find . -mtime -1 -type f >> file_audit.txt
[(kali㉿kali)-~]
$ echo "" >> file_audit.txt
[(kali㉿kali)-~]
$ echo "Large files (>1KB):" >> file_audit.txt
[(kali㉿kali)-~]
$ find . -size +1k -type f -exec ls -lh {} \; >> file_audit.txt
[(kali㉿kali)-~]
$ sudo updatedb
[sudo] password for kali:
[(kali㉿kali)-~]
$ locate passwd
/etc/passwd
/etc/passwd-
```

Following the last slide, I used a command to conduct further file discovery. I used commands to find permissions, size, and time. Then I did a deeper search with security-focused operations and an advanced file with actions. Afterwards, I created a comprehensive file audit. Now I'm executing some basic locate operations.



The screenshot shows a terminal window titled "kali@kali: ~/incident_analysis". The terminal displays a sequence of commands used for security analysis:

```
$ find /usr/bin -perm -4000 2>/dev/null | head -5 >> vuln_assessment.txt
(kali㉿kali)-[~/incident_analysis]
$ echo "" >> vuln_assessment.txt

(kali㉿kali)-[~/incident_analysis]
$ echo "3. Recently modified files in /tmp:" >> vuln_assessment.txt

(kali㉿kali)-[~/incident_analysis]
$ find /tmp -mtime -1 -type f 2>/dev/null | head -5 >> vuln_assessment.txt

(kali㉿kali)-[~/incident_analysis]
$ echo "==== Configuration Security Audit ===" > config_audit.txt

(kali㉿kali)-[~/incident_analysis]
$ echo "Checking configuration file permissions ..." >> config_audit.txt

(kali㉿kali)-[~/incident_analysis]
$ echo "" >> config_audit.txt

(kali㉿kali)-[~/incident_analysis]
$ for config in /etc/passwd /etc/shadow /etc/ssh/sshd_config; do
for> if [ -f "$config" ]; then
for then> echo "File: $config" >> config_audit.txt
for then> ls -la "$config" >> config_audit.txt
for then> echo "" >> config_audit.txt
for then> fi
for> done

(kali㉿kali)-[~/incident_analysis]
$
```

Following the series of basic locate operations, I then moved to comparing find vs locate operations. After that, I created a security reconnaissance with locate operations. Then, I made an incident file analysis and a permission vulnerability assessment. Following that, I made a configuration file audit.

The screenshot shows a terminal window titled "kali@kali: ~/cybersec-labs/permissions-lab/secure_configs". The terminal displays the following command-line session:

```
zsh: corrupt history file /home/kali/.zsh_history
heredoc> echo ""[~]
heredoc> echo "Executable files:"
heredoc> find . -type f -executable -exec ls -la {} \;
heredoc> echo ""
heredoc> echo "Files with unusual permissions:"
heredoc> find . -type f \(-perm -4000 -o -perm -2000 \) -exec ls -la {} \; 2>/dev/null
heredoc> EOF

[~(kali㉿kali)-[~/cybersec-labs/permissions-lab/secure_configs]
$ chmod +x security_check.sh

[~(kali㉿kali)-[~/cybersec-labs/permissions-lab/secure_configs]
$ ./security_check.sh
== Automated Security Permission Check ==
Checking for insecure permissions ...

World-writable files (security risk):

World-readable config files (potential risk):
./security_check.sh: line 9: /dev/nul: Permission denied
-rw——— 1 kali kali 28 Sep 20 11:46 ./db_config.conf

Executable files:
-rwx——— 1 kali kali 37 Sep 20 11:47 ./admin_script.sh
-rwxrwxr-x 1 kali kali 661 Sep 20 11:55 ./security_check.sh

Files with unusual permissions:

[~(kali㉿kali)-[~/cybersec-labs/permissions-lab/secure_configs]
$
```

To recap what I've done: I created a private escalation reconnaissance, secure file structure, permission remediation examples, and automated security check scripts. I verified the secure permissions. When I performed the security check, I obtained the output shown above. I did have one hiccup when creating the automated security check script. The output doesn't look correct to me.

```
kali@kali: ~/cybersec-labs/permissions-lab/secure_configs
File Actions Edit View Help
zsh: corrupt history file /home/kali/.zsh_history
[(kali㉿kali)-[~]]$ cd ~/cybersec-labs/permissions-lab
[(kali㉿kali)-[~/cybersec-labs/permissions-lab]]$ cd secure_configs
[(kali㉿kali)-[~/cybersec-labs/permissions-lab/secure_configs]]$ find . -nouser -o -nogroup 2>/dev/null >> incident_files.txt || echo "None found" >> incident_files.txt
[(kali㉿kali)-[~/cybersec-labs/permissions-lab/secure_configs]]$
```

I know the screenshot doesn't show much. I messed up on the very last line of the incident response file patterns and had to close my terminal. But to recap what I've done from the last slide, I found files using multiple different criteria. I then used some security-focused file patterns and complex logical operations.

Knowledge Assessment:

Question 1: True or False: A file with permissions 644 (rw-r--r--) allows the group and others to modify the file.

False

Question 2: Which command would find all SUID files in the /usr/bin directory?

b) find /usr/bin -perm -4000 -type f

Question 3: True or False: The locate command searches the filesystem in real-time, making it slower than find but more accurate.

False

Question 4: What does the command chmod u+s filename accomplish?

b) Sets the SUID bit on the file

Question 5: Which find command would locate world-writable files?

- b) `find . -perm -002`

Conclusion:

With the completion of this lab, I have a greater understanding of manipulating file permissions with various commands. I have now discovered and learned about how to identify and analyze SUID/SGID files for security assessment. I am now able to apply advanced find techniques for security analysis and incident response, as well as locate files with both locate and find commands. After this lab and week of class time, I'm very intrigued by jobs within Digital Forensics. I like the thought of working with local law enforcement and actively investigating and making the world a better place.