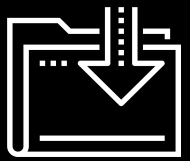
Backups and Restoring Data with tar

Cybersecurity

5. 1 Archiving and Logging Data



Class Objectives

By the end of today's class, you will be able to:

Identify and describe use cases for the three kinds of backups.

Create (tar) and archive from existing files and directories.

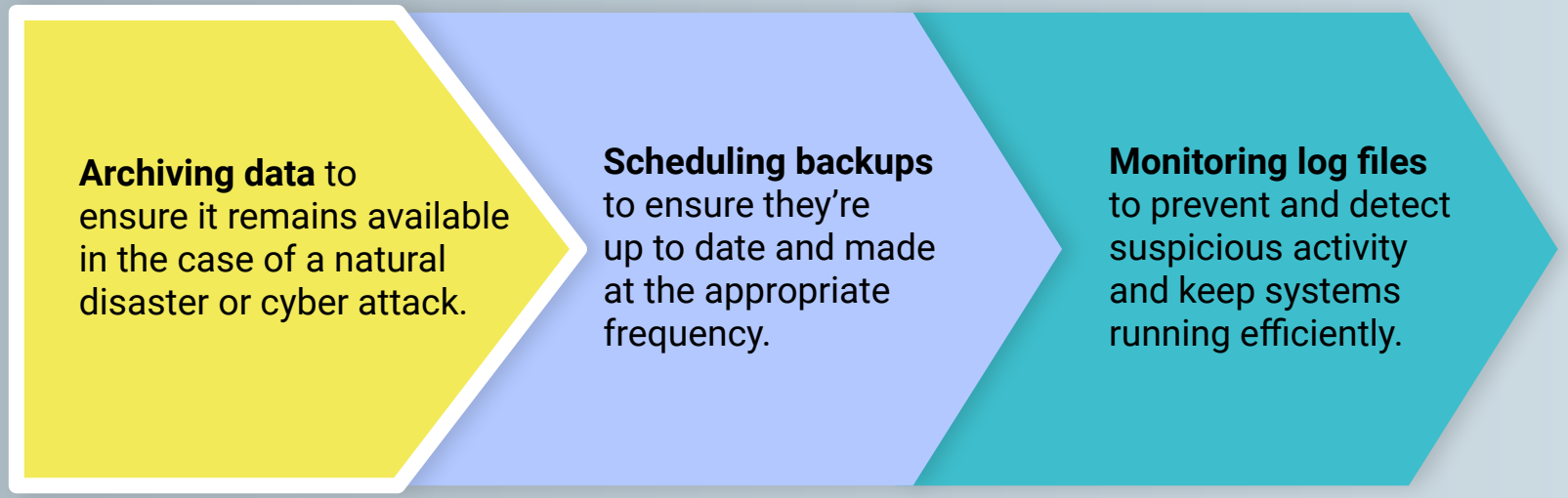
List and search the contents of an existing archive.

Extract the contents of an archive.

Describe and demonstrate two ways to exploit the `tar` command.

Archiving and Logging Data

In this unit, we'll continue to explore how Linux system administrators keep their system secure by:



Archiving and Logging Data

These skills are used by system administrator to accomplish the following tasks:

- Overseeing or conducting data backup and recovery.
- Determining how long to retain data and the frequency of backups.
- Monitoring and troubleshooting backups and restoration.
- Providing security for compliance requirements.

Archiving with `tar`

Archives are essential in IT security for maintaining regulatory compliance in industries like finance and health.

Finance

In finance, the common standard for data archiving is the **Sarbanes-Oxley Act (SOX)**. It requires all business records and communication to be retained for five years.

Before the signing ceremony of the Sarbanes–Oxley Act, President George W. Bush meets with Senator Paul Sarbanes, Secretary of Labor Elaine Chao and other dignitaries in the Blue Room at the White House on July 30, 2002

Finance

The **Markets in Financial Instruments Directive (MiFID II)** requires the European Union's financial firms to retain and reproduce records of all activity from telephone conversations and electronic communications, including instant messages and social media interactions.

Health

The most common standard is **Health Insurance Portability and Accountability Act (HIPAA)**, which requires healthcare providers to keep records for six years.

Introduction to the `tar` Command

Cyber professionals must ensure that data is always protected against threats of data loss and interruptions caused by attacks or natural disasters.

Ensuring Availability of Data

When important institutions are under threat, data must remain available.

In 2019, hackers seized important government machines during a ransomware Attack in Baltimore, Maryland.

vox.com

Ensuring Availability of Data

When important institutions are under threat, data must remain available.

In 2019, Hurricane Michael caused massive flooding, destroying homes, businesses, and strategic defense data at Tyndall Air Force Base.

Hurricane-Damaged Air Force Base Has an Opportunity to Rebuild for Resilience

Experts see the recovery effort as a test case for how the U.S. military prepares for climate change

By Courtney Columbus, E&E News on November 9, 2018

READ THIS NEXT

SPACE
On Earth: Stardust from 7.5 Billion Years Ago
1 hour ago — Caleb A. Scharf

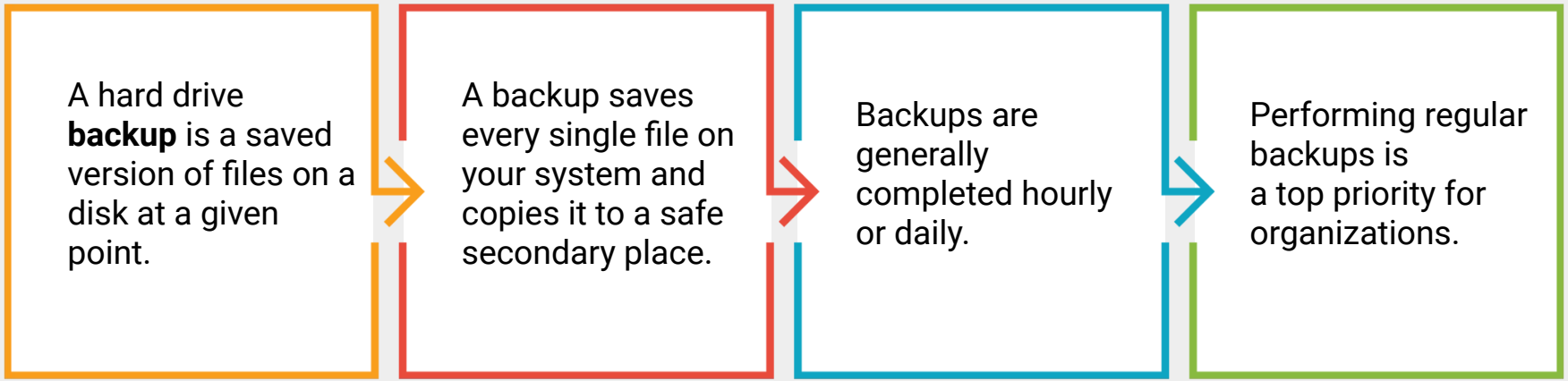
COGNITION
My Go-To Arguments for Free Will
3 hours ago — John Horgan | Opinion

WELLNESS
7 Benefits of Swimming and How to Get Them

Source

Backing Up Data

System administrators backup data so organizations can quickly recover lost assets and quickly restore systems.



Types of Backups

There are three different types of backup: full, incremental, and differential.

Full backups are more reliable and offer complete restoration , but require large storage space.

Incremental and **differential backups** include only data that changed since the last full or incremental backup. They are fast and use less disk space.

Performing a Backup

`tar`

The `tar` command is a Linux utility that system administrators use to create a backup.

`tar` (tape archive) takes files we want to back up and creates an archive of them.

An archive is a special file that contains a concatenation of files and directories.

Archives created with `tar` are called Tarballs and use the extension `.tar`.

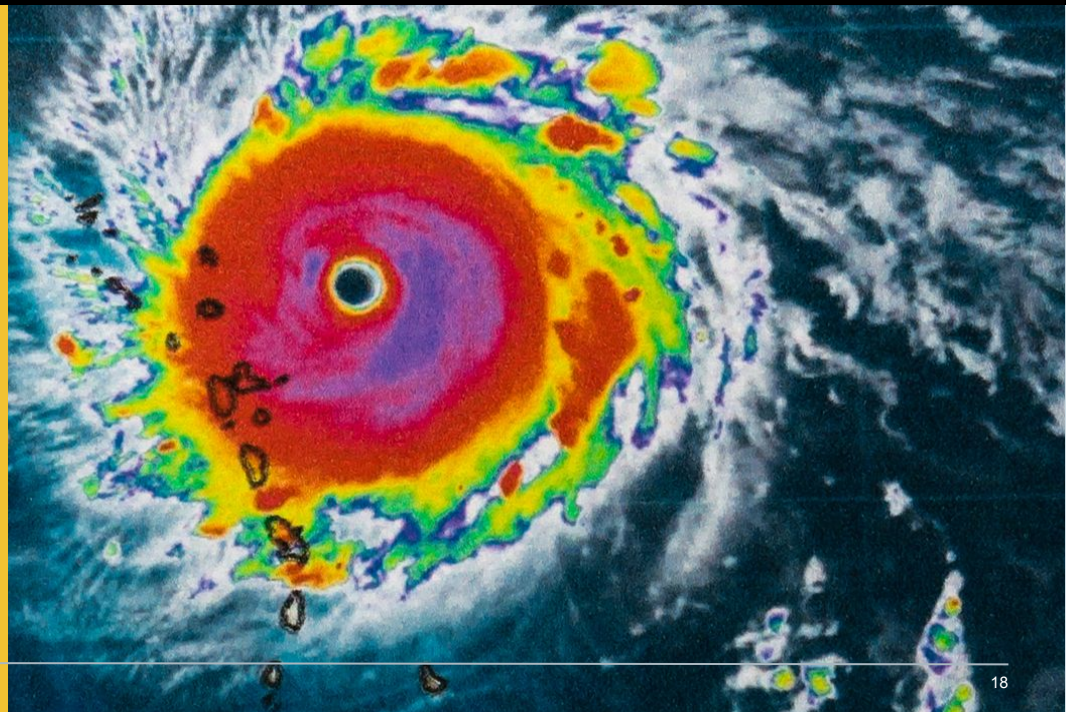
Creating an Archive

The general syntax of the **tar** command is:

```
tar [option(s)] [archive_name] [objects_to_archive]
```

Let's apply this command to following scenario:

The Systems Operations Center at Tyndall Air Force Base is backing up all system data with the approaching threat of Hurricane Michael.



Creating an Archive

Create a **full backup** of all log files in `/var/log` using the `tar` command:

```
sudo tar cvf hurricane-backup-10-11-2018.tar /var/log
```

`sudo` because /var/log contains system files, we'll need administrator rights in order to make a backup of them.

It would compromise security if non-administrative users were able to make backups of system files.

Creating an Archive

Create a **full backup** of all log files in `/var/log` using the `tar` command:

```
sudo tar cvf hurricane-backup-10-11-2018.tar /var/log
```

`tar` is the Linux command to initiate the backup.

Creating an Archive

Create a **full backup** of all log files in `/var/log` using the `tar` command:

```
sudo tar cvf hurricane-backup-10-11-2018.tar /var/log
```

`cvf` are short form options.

c	stands for *create*. Always needed when creating an archive.
v	stands for *verbose*. It will print progress status and useful information as it's running.
f	indicates the title of the archive.

Creating an Archive

To create a **full backup** of all the log files in /var/log directory, use the following tar command:

```
sudo tar cvf hurricane-backup-10-11-2018.tar /var/log
```

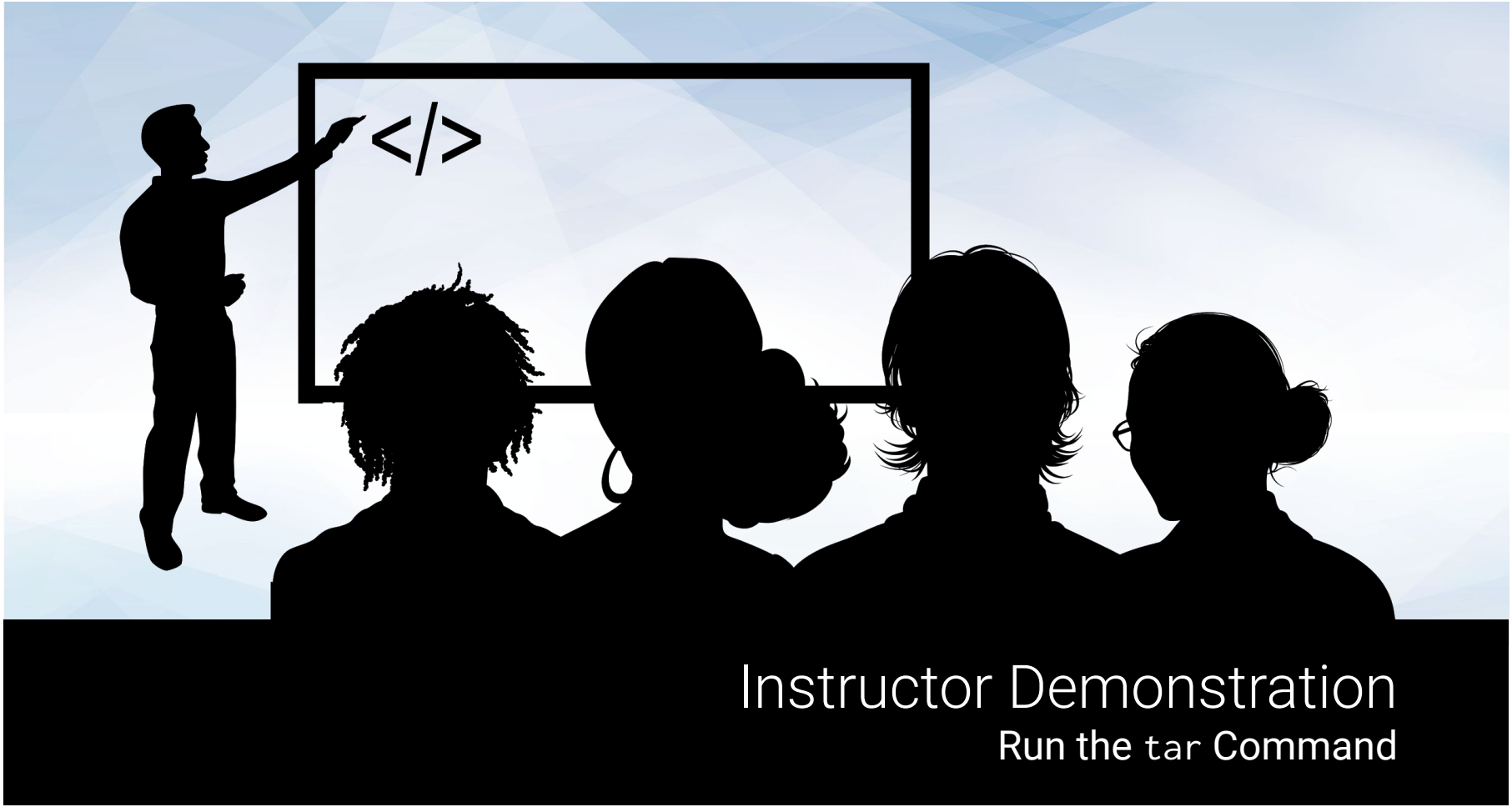
`hurricane-backup-10-11-2018.tar`
is the archive name, indicating the title.

Creating an Archive

To create a **full backup** of all the log files in /var/log directory, use the following tar command:

```
sudo tar cvf hurricane-backup-10-11-2018.tar /var/log
```

`/var/log` are the objects to archive, indicating the files or directories we want to backup.



Instructor Demonstration

Run the `tar` Command

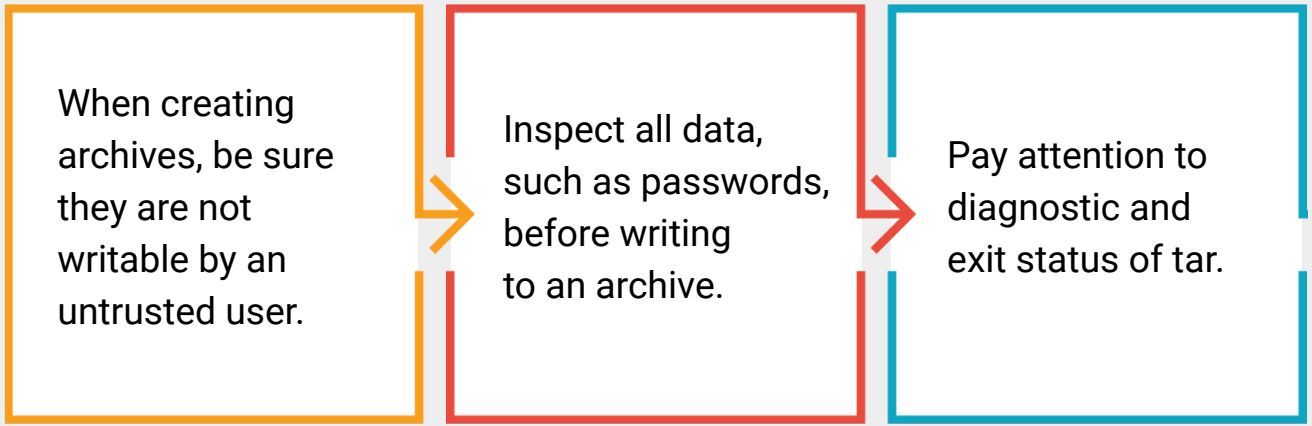
Compressing archives with gzip

`gzip` is a command used to create a compressed tar archive, indicated with a `tar.gz` extension

- gzip compression is useful when disk space is an issue.
- Additionally, moving smaller files sizes uses less bandwidth, making transfer times faster.
- `gzip 2018-10-12-hurricane-backup.tar` will compress the archive with a .gz extension.
- `gunzip 2018-10-12-hurricane-backup.tar.gz` will unzip an archive and remove the .gz extension, restoring the archive back to its original state.

Tar: Best Practices

In order to maintain confidentiality, privacy, and integrity:

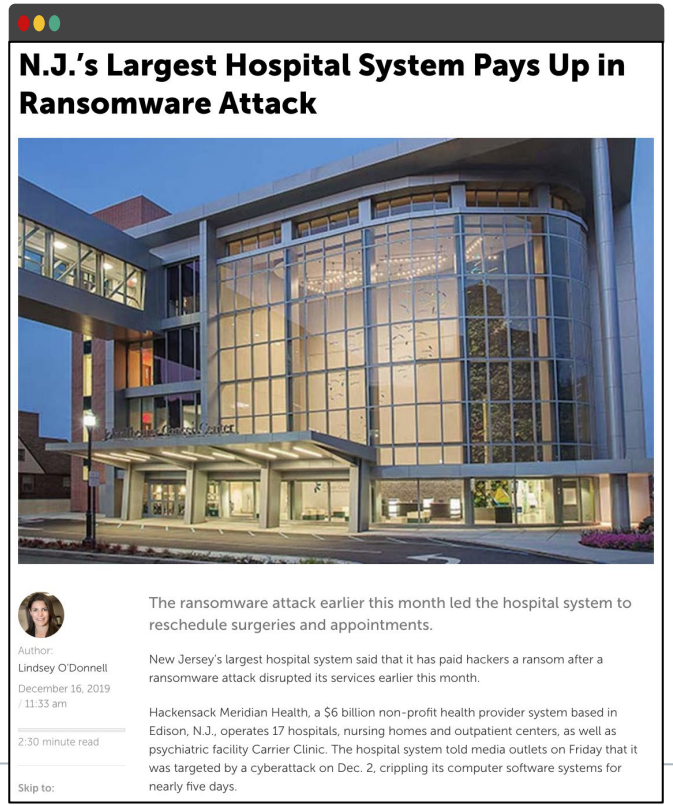


Restoring Data with `tar`

Case Study: Backing Up Hackensack

In 2019 Hackensack Meridian Health systems were compromised by a ransomware attack.

- The attack crippled computer systems and forced hospitals to reschedule surgeries and appointments.
- The attackers offered to decrypt the data in exchange for payment, effectively ransoming the hospital's data.
- **Unfortunately, the hospital was not prepared and had to pay the attackers.**



N.J.'s Largest Hospital System Pays Up in Ransomware Attack

The ransomware attack earlier this month led the hospital system to reschedule surgeries and appointments.

Author:
Lindsey O'Donnell
December 16, 2019 / 11:33 am

2:30 minute read

Skip to:

New Jersey's largest hospital system said that it has paid hackers a ransom after a ransomware attack disrupted its services earlier this month.

Hackensack Meridian Health, a $6 billion non-profit health provider system based in Edison, N.J., operates 17 hospitals, nursing homes and outpatient centers, as well as psychiatric facility Carrier Clinic. The hospital system told media outlets on Friday that it was targeted by a cyberattack on Dec. 2, crippling its computer software systems for nearly five days.

Case Study: Backing up Hackensack

In this demonstration, we'll respond to a similar ransomware attack that hit hospital systems on the morning of May 11, 2019.

Among the affected systems were:

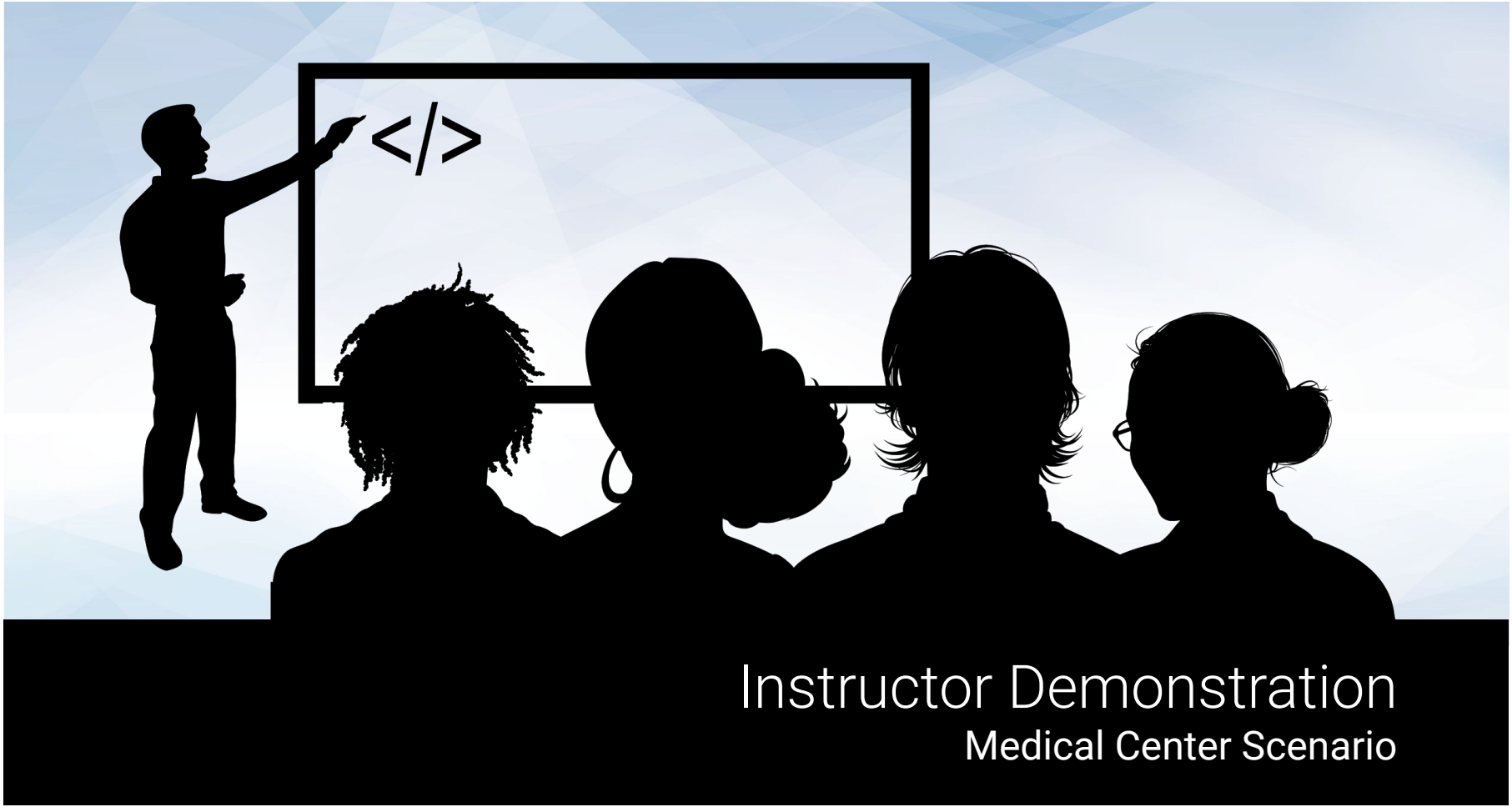
- Doctors
- Patients
- Treatments

After taking the infected systems offline, we'll need to:

- Restore the operating system and applications using the full backup.
- Restore email data using the full backup.

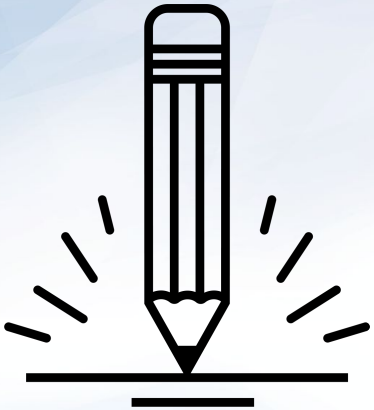
Restoring the email data will require using the tar command in order to:

- List the contents of the latest full backup to locate the email data.
- Extract the email data from the archive to a directory on the new system.



Instructor Demonstration

Medical Center Scenario



Activity: Creating and Restoring Data with `tar`

In this activity, you will use the `tar` command to create backups and restore lost files.

Suggested Time:
25 Minutes

Time’s Up! Let’s Review.

Incremental Backups with `tar`

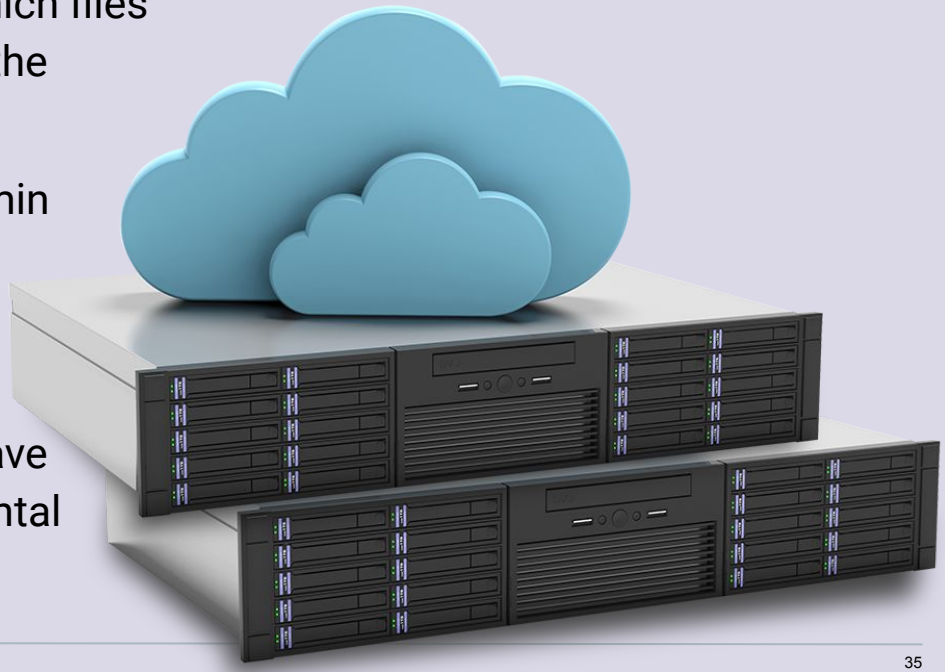
The file sizes for full backups are very large, because they include everything on a system.

Depending on the size of the file system, doing a full backup takes a very long time.

Incremental Backup

Incremental backups are completed after a full backup is performed on a system, only capturing what has changed since the last incremental backup.

- Incremental backups store a list of which files have changed in a snapshot file, with the extension `.snar`.
- The snapshot is created when the admin creates the initial level 0 backup.
- Every time an incremental backup completes, a new snapshot is created that contains only files that have changed since the last full or incremental backup.



Incremental Backup

Incremental backups are completed after a full backup is performed on a system, only capturing what has changed since the last incremental backup.

Incremental backups have the following drawbacks:

Restoring is slower. You must restore the latest full backup, plus all the previous incremental backups.

There may be a loss of data if a single backup is corrupted.

You must verify an archive was created with no errors, using the `tar W` option.

Creating the Incremental Backup

In this section, we will to use tar in order to create incremental backups:

An incremental backup is a special form of a tar archive.

tar stores additional metadata or information so that the exact state of the file system can be restored when extracting the archive.

The additional information includes which files have been changed, added, or deleted since the last backup, so that the next incremental backup will contain only modified files.

The additional information is stored in a snapshot file.

Incremental Backup

```
$ tar cvvWf emerg_back_sun.tar --listed-incremental=emerg_backup.snar --level=0 emergency
```


`--listed-incremental=emerg_backup.snar` indicates this backup will be part of a series of incremental backups, and specifies that information about files added, changed, or removed should be stored in a snapshot file called **`emerg_backup.snar`**.

Incremental Backup

```
$ tar cvvWf emerg_back_sun.tar --listed-incremental=emerg_backup.snar --level=0 emergency
```

Since this is the first backup in the series, it is both a full backup and an incremental backup.

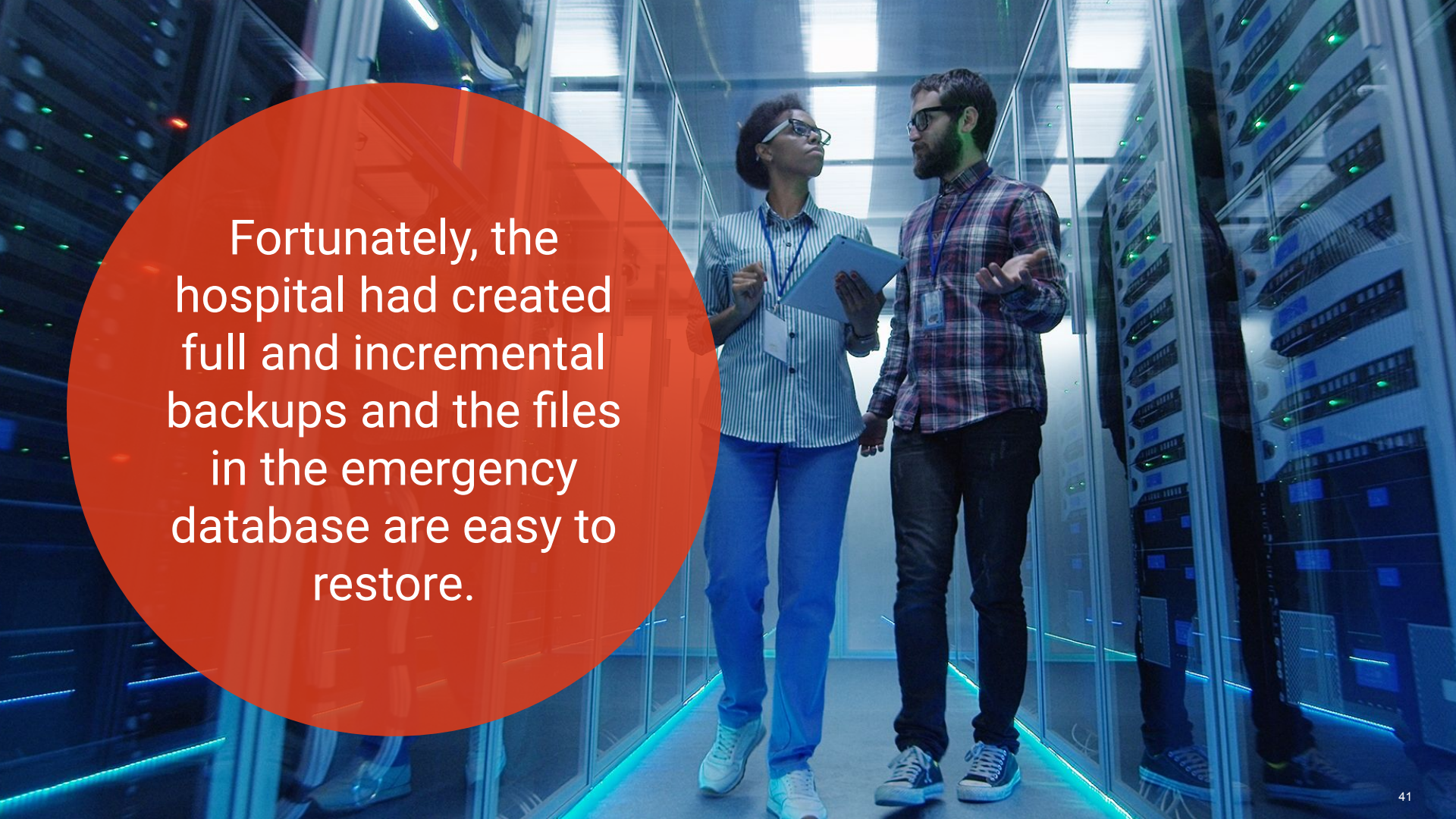
This is because the first backup in a series of incrementals is basically a full backup, also known as a level 0 backup.

Every succeeding backup will be much smaller than this one.

Incremental Backup Scenario

We'll use the following scenario to demonstrate incremental backups:

Sunday	Monday	Tuesday	Wednesday
A hospital's system operations staff performs a full backup of all the data in the emergency database.	The staff runs an incremental backup which stores all changes made on Monday. This backup is smaller than the full backup.	An incremental backup archives only the changes made on Tuesday to the emergency database.	The hospital is hit with a ransomware attack and all the data in the emergency database is encrypted.


Fortunately, the hospital had created full and incremental backups and the files in the emergency database are easy to restore.

Incremental Backup Scenario

We'll use the `tar` command to complete the following steps:

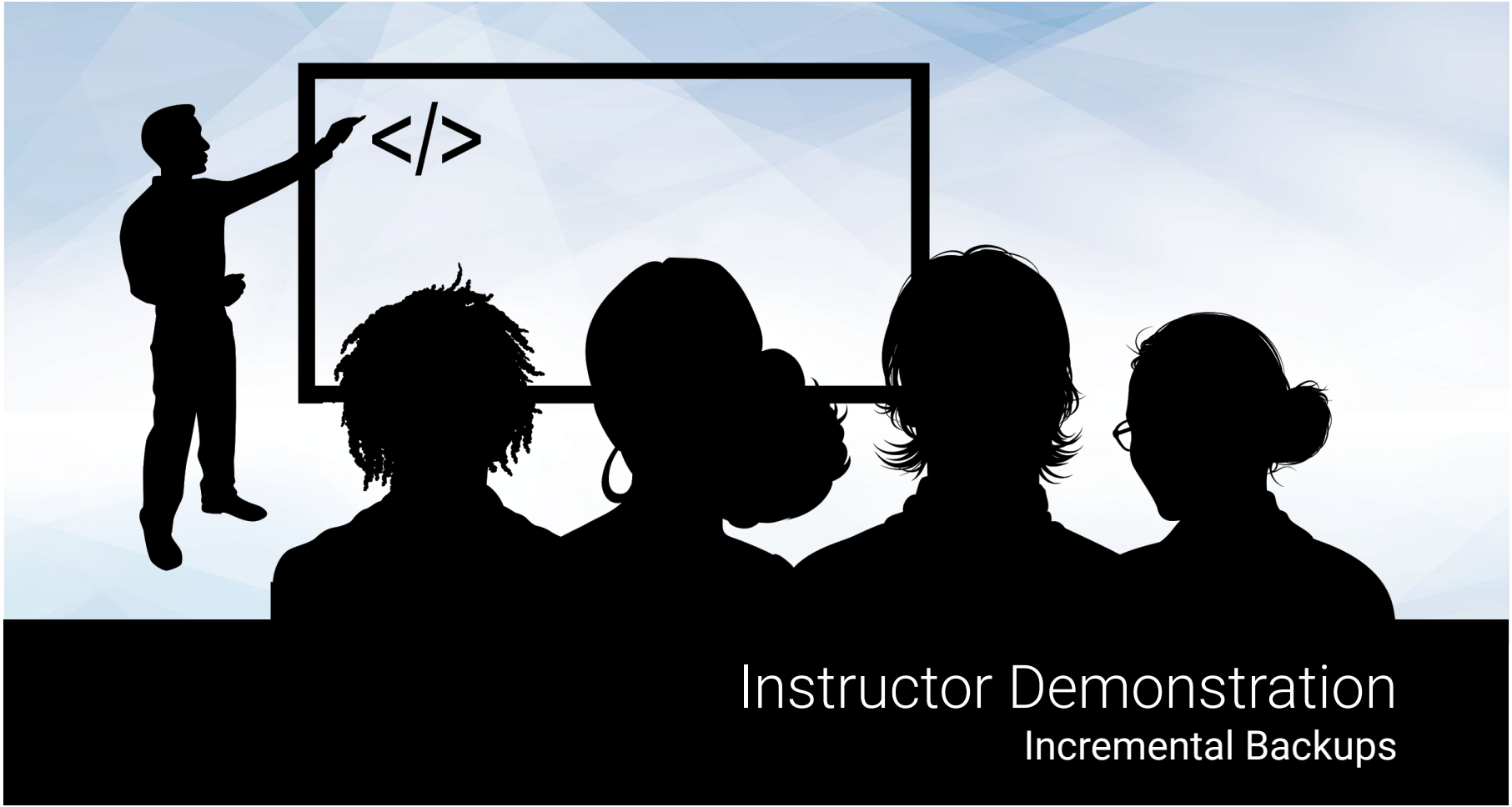
01 **Create** a full backup of the emergency directory and a snapshot file.

02 **Display** the contents of the level 0 backup.

03 **Create** the incremental backups generated on Monday and Tuesday.

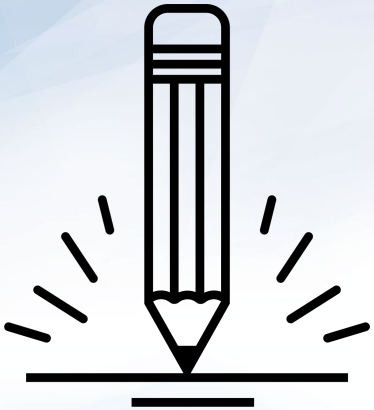
04 **Remove** the emergency directory to mimic the ransomware attack on Wednesday.

05 **Restore** the emergency directory.



Instructor Demonstration

Incremental Backups



Activity: Restoring Data with Incremental Backups

In this activity, you and a partner will work as junior admins tasked with restoring the incremental backup of the patient files in a test environment.

Suggested Time:
20 Minutes

Time’s Up! Let’s Review.


15:00
Break

Exploiting the `tar` Command with the Checkpoint and Wildcard Options

Hackers can combine the tar command with the wildcard character (*) and the checkpoint feature to plant malware on a system.

Using Wildcard with `tar`

Previously, we used the wildcard character (*) when creating an archive.

Sysadmins use the wildcard symbol (*) to specify multiple files in a directory without having to type each filename.

For example, the command `tar cvf archive.tar ./*` would archive the contents of `Documents/ExploitTar`.

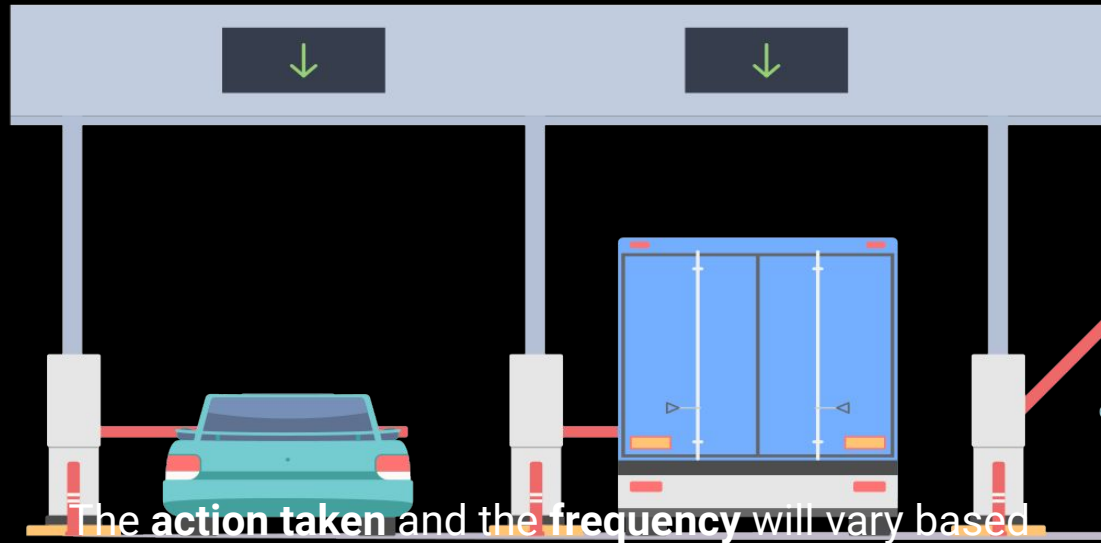
```
cd ~/Documents/ExploitTar
 ls
    f1
    f2
    f3
    f4
    f5
    f6
    f7
    f8
    f9
    ‘important docs’
  tar cvf archive.tar ./*
```

Using Checkpoints

A **checkpoint** is a specific stage of a backup or restoration that can trigger designated actions.

For example:

1. Once the backup reaches 1000 files, a checkpoint triggers the backup to pause and display the amount of remaining disk space.
2. When restoring files from a remote site, a message will print the number of bytes transferred every 5000 files.

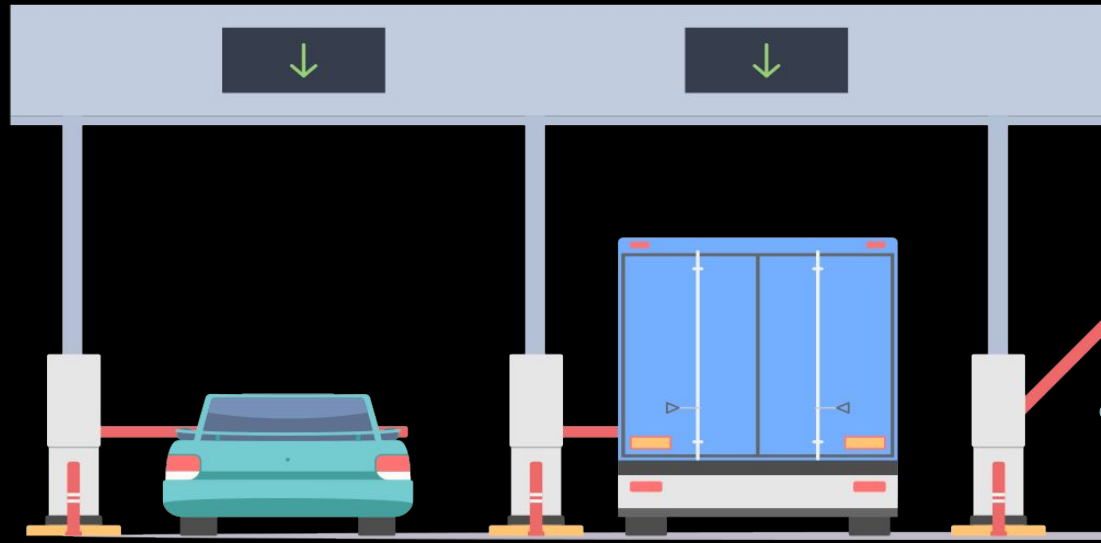


The **action taken** and the **frequency** will vary based on the circumstance.

Using Checkpoints

Including checkpoints in our backup and restoration process is a best practice.

It allows administrators to avoid serious issues, such as accidentally using up all of the space on the backup server, or slowing down the network—both of which can damage data availability.



Using Checkpoints

Checkpoints involve two commands. The first defines the checkpoint, and the second specifies the action to be taken at that checkpoint.

For example:

`--checkpoint=1000` indicates how often the checkpoint occurs.

- At every thousandth file an action will take place.
- Syntax: `--checkpoint=[n]`

`--checkpoint-action=du` indicates the action that will take place:

- `du` displays the available disk space on the machine.
- Syntax: `checkpoint=[ACTION]`

Exploiting System Vulnerabilities

In the previous Linux unit, we learned how sysadmins harden systems against exploits.

01

When a weakness is found, hackers can exploit it by changing existing code or using their own code.

This is also known as arbitrary command execution (ACE).

02

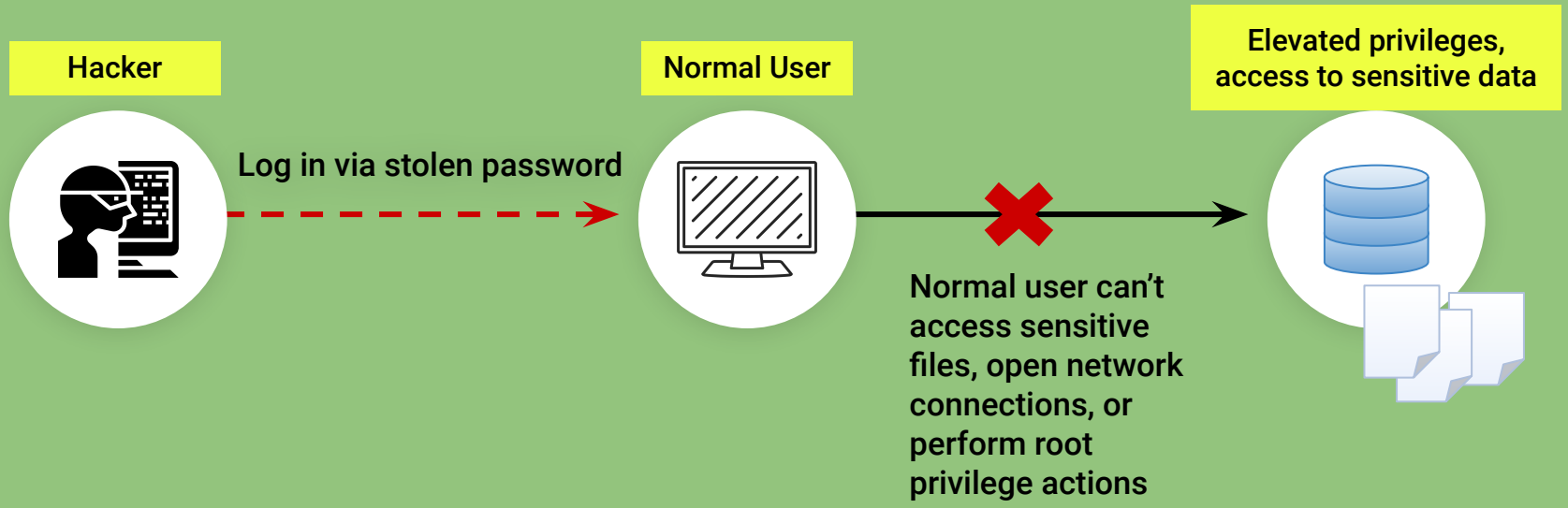
ACE vulnerabilities are extremely dangerous. Attackers can exploit them to accomplish almost anything, including gaining root privileges, which allows them to completely compromise a machine.

03

For our example, hackers have discovered how to use the `tar` checkpoint option with a wildcard to plant malicious code on a system, which they can later run to gain root privileges on the machine.

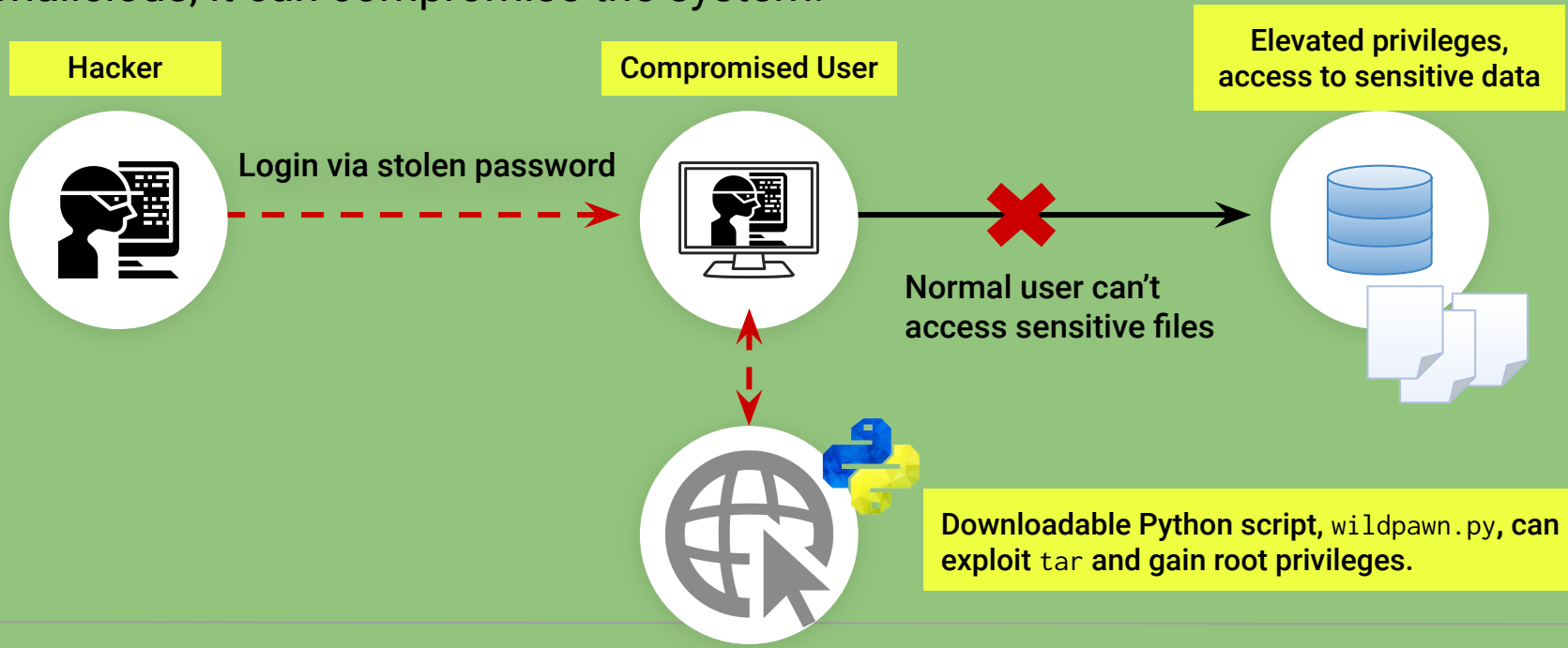
Checkpoint + Wildcard Exploit

First, a hacker gains access to a system by impersonating a normal user.



Checkpoint + Wildcard Exploit

A normal user can typically download and execute files from the internet. If a file is malicious, it can compromise the system.



Checkpoint + Wildcard Exploit

Unsuspecting admins with root privileges run a `tar` command containing a wildcard. This will create an archive containing the directory of the malicious script.

`wildpawn.py` **plants three malicious files on the machine: one hidden, and two files with names like** `tar` **checkpoint commands.**

tar + wildcard

Sysadmin unknowingly creates archive containing planted malicious script.

Hacker can drop into root shell, gaining full access to system.

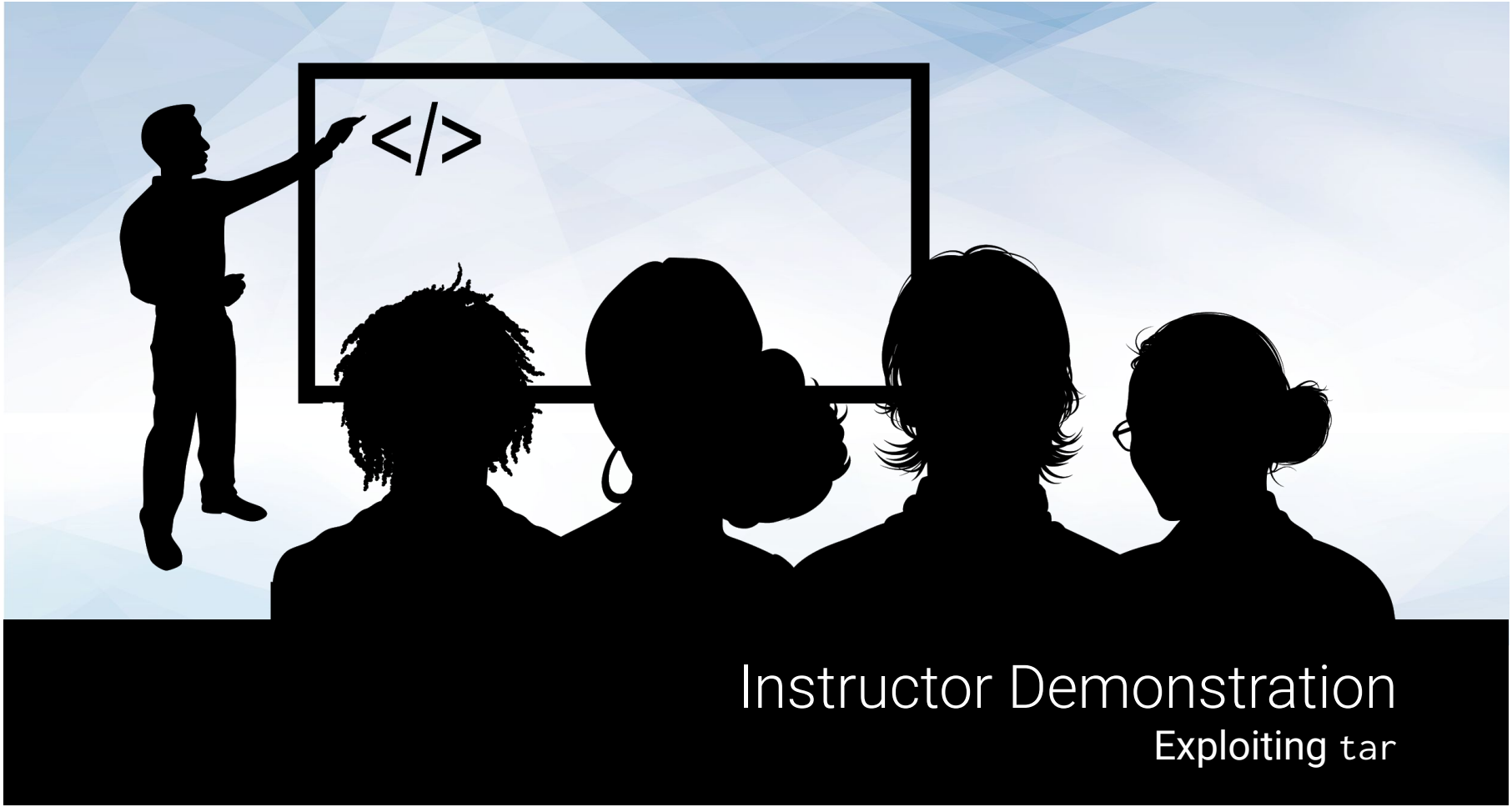
Exploiting `tar`

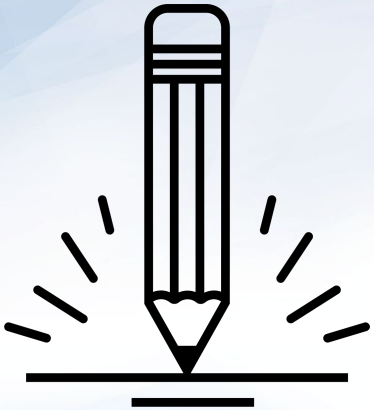
But how do checkpoints and wildcards fit into this exploit?

When a hacker runs `wildpawn.py`, it creates three files.

- The hacker is able to create these files even if they only have normal user privileges.
- The files include a malicious hidden script called `.webscript` and two files whose names appear like checkpoint commands:
  - `--checkpoint=1`
  - `--checkpoint-action=exec=sh .webscript`

After adding the first file to the archive, `tar` will execute the malicious `.webscript`.


Instructor Demonstration
Exploiting tar



Activity: Exploiting `tar`

In this activity, you will play the role of a penetration tester hunting for vulnerabilities in a target system backup procedure.

Use the `wildpwn.py` tool to verify the vulnerability, then research two mitigation strategies to protect the server.

Suggested Time:
25 minutes

Time’s Up! Let’s Review.

Questions?