



Access Controls and Managing Services

Cybersecurity

4.3 Managing Permissions and Services



Class Objectives

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



Inspect and set file permissions for sensitive files on the system.



Manage and monitor services on the system, and remove unused services.



Create and assign users for services.

Access Controls



Like Google Docs, Linux has **access controls**, which grant permission to access documents and files on a host.

Managing Access Controls in Linux

Managing files, programs, and files as items.



Item

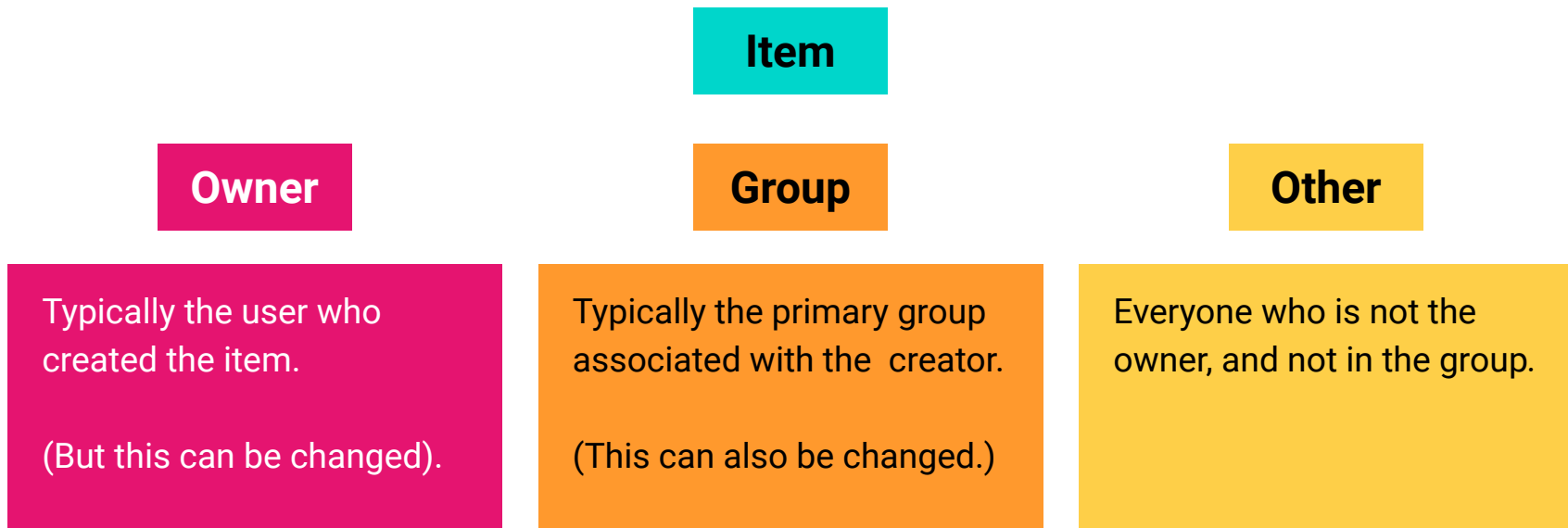
Managing Access Controls in Linux

Each process has a number of *permissions* on the item, the *group* associated with the item, and *others*.



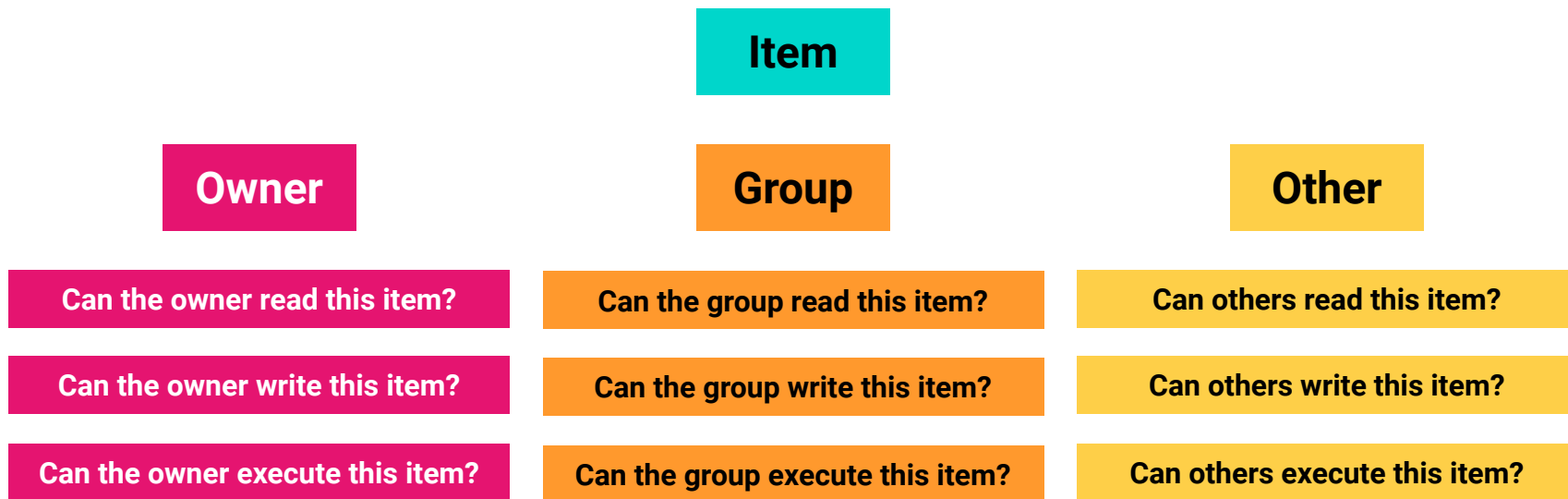
Managing Access Controls in Linux

Each file or directory has a *user* associated with it, the *group* associated with the item, and *others*.



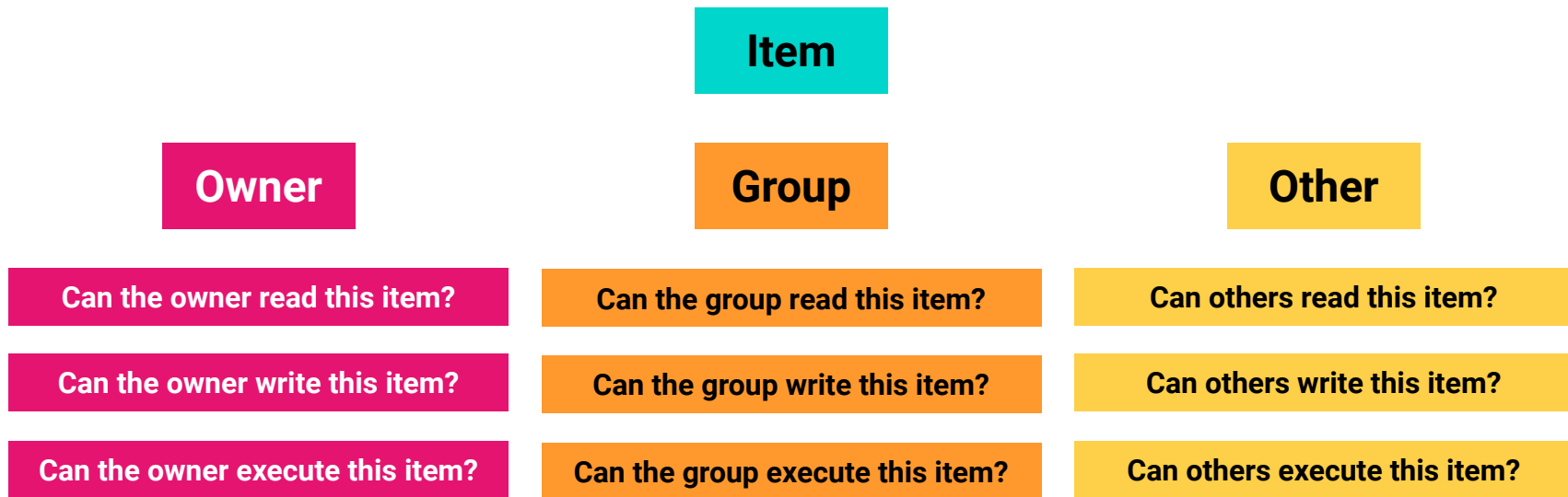
Managing Access Controls in Linux

Managing the access policies, rules and restrictions that we can allow or prevent:
read, write, execute.



Managing Access Controls in Linux

Assigning these permissions is called **Discretionary Access Control (DAC)**. It is *discretionary* because item permissions are flexible and can be arbitrarily modified by someone with sufficient privileges.



Permissions Demo

In the upcoming demo, we'll create a file and a directory, observing default permissions. Then, we will change the permissions to deny certain groups and users access.

To read and manipulate these file permissions, we'll use these commands:

<code>ls -l</code>	Show the permissions info.
<code>chmod</code>	Change the permissions info.
<code>chown</code>	Change the owner and group of a file.



Instructor Demonstration

Permissions

Inspecting File Permissions

-rw-r--r--

Item type (- for file, d for directory)

Inspecting File Permissions

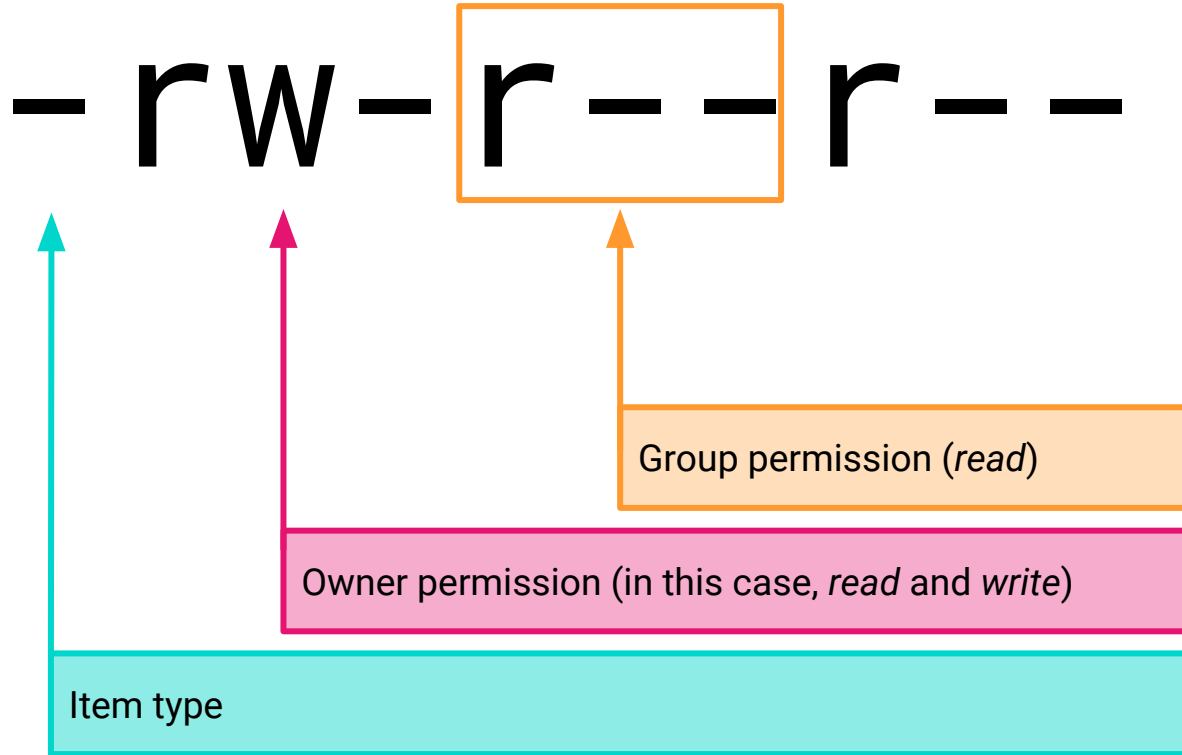
- **rw-** r - - r - -



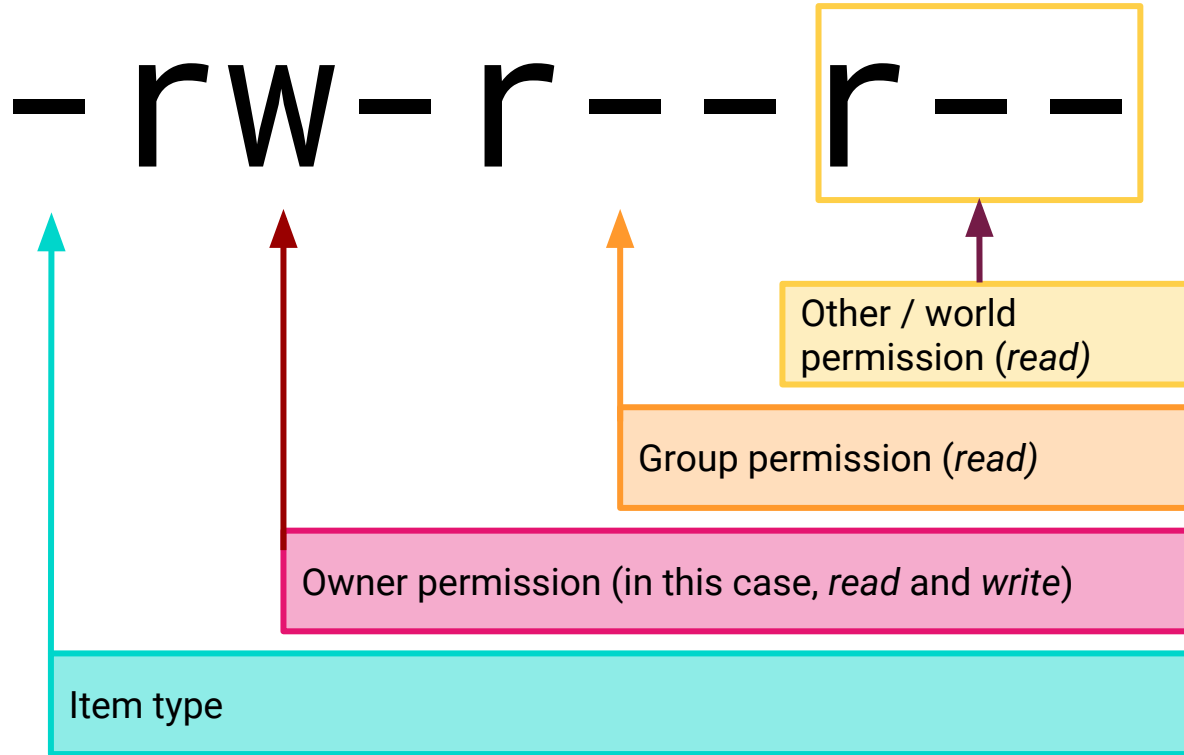
Owner permission (in this case, *read* and *write*)

Item type

Inspecting File Permissions



Inspecting File Permissions



Changing File Permissions

File permissions can be set using two different notations: **symbolic** and octal.

Symbolic Notation	
r	read
w	write
x	execute

rwX **rw-** **r--**

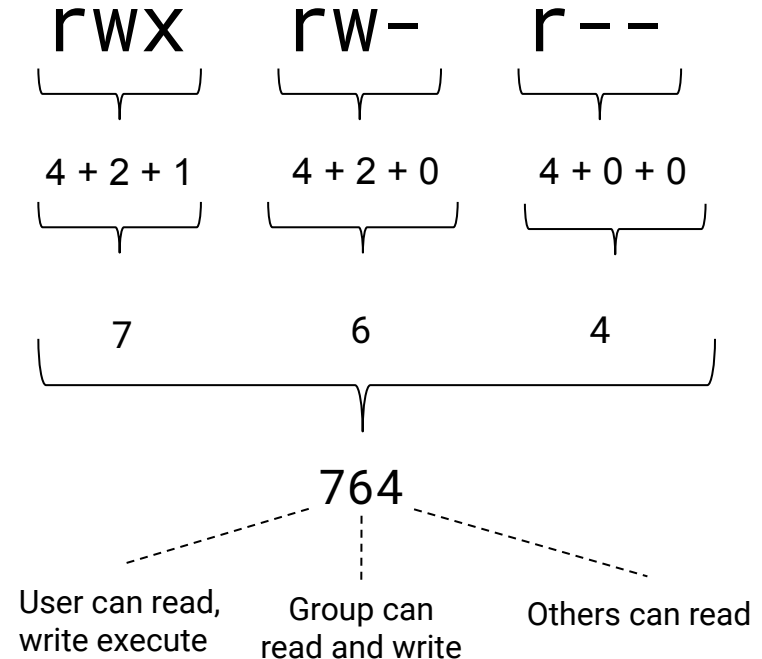
⏟ ⏟ ⏟

User can read,
write execute Group can
read and write Others can read

Changing File Permissions

File permissions can be set using two different notations: symbolic and **octal**.

Octal Notation				
	4	2	1	
0	-	-	-	No permission
1	-	-	x	Only execute
2	-	w	-	Only write
3	-	w	x	Write and execute
4	r	-	-	Only read
5	r	-	x	Read and execute
6	r	w	-	Read and write
7	r	w	x	Read, write, and execute





Activity: Access Controls and Permissions

In this activity, you will inspect and set file permissions on a few of the most sensitive items on a Linux system.

Suggested Time:

25 Minutes



Time's Up! Let's Review.

Questions?



Recap: Permissions

How permissions apply to each specific file and folder with `r`, `w`, and `x`.

Symbolic Notation	
<code>r</code>	read
<code>w</code>	write
<code>x</code>	execute

`rwX` `rw-` `r--`

⏟ ⏟ ⏟

User can read,
write execute Group can
read and write Others can read

Permissions

How to view and apply permissions to an item's user, group, and other.

Users

Every file and program on a Linux system has permissions.

These permissions tell the system which users can access a file or run a program.

Groups

Users can be placed in groups, which can have special permissions that apply to all members of the group.

Root

File and program permissions apply to all users *except* the root.

The root user (or super user) has complete access to the system and can perform any task.

Permissions

We can use `sudo` user to invoke the `root` user and bypass any permissions.

<code>ls -l</code>	To show the permissions info.
<code>chmod</code>	To change the permissions info.
<code>chown</code>	To change the owner and group of a file.

Permissions

We can assign `sudo` for a specific command for a specific user.

<code>whoami</code>	To determine the current user.
<code>su</code>	To switch to another user, in this case the root user.
<code>sudo</code>	To invoke the root user for one command only.
<code>sudo -l</code>	To list the <code>sudo</code> privileges for a user.
<code>visudo</code>	To edit the <code>sudoers</code> file.



Managing Services

A perspective view of a server room with rows of server racks on both sides. The racks are filled with electronic equipment, including circuit boards, switches, and numerous small, colorful indicator lights (red, green, blue, yellow). The floor is a light-colored, reflective material, and the ceiling has recessed lighting fixtures. The overall atmosphere is dimly lit with a strong blue glow from the server equipment.

Servers are computers that offer services to other computers.

Managing Services

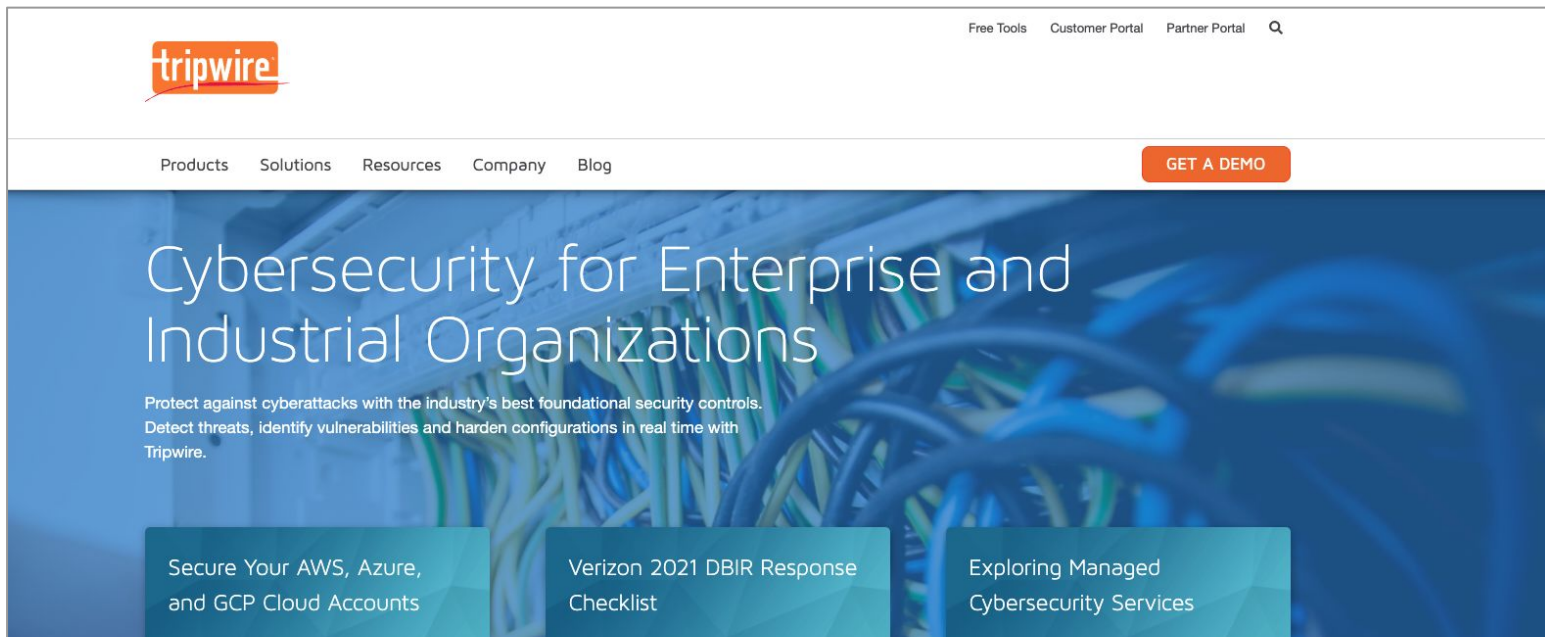
A service is a function or capability that a machine makes available to another.

For example, file sharing services allow computers to send and receive data.



Managing Services

Some services, like Tripwire, are only run locally on the server and are not provided to other computers. These services are packages that can be installed and removed just like other programs.



Services and Security

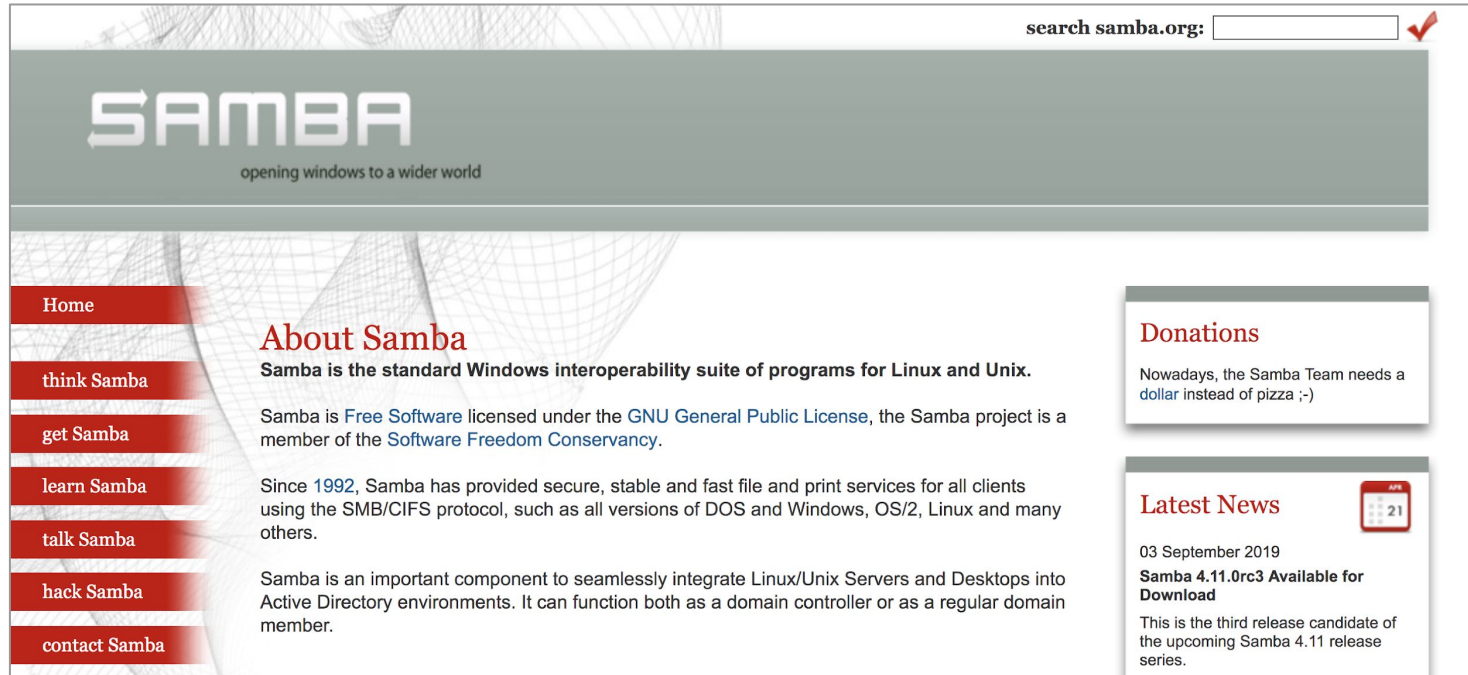
Services and Security

Attackers can manipulate vulnerable services into doing things that they are not designed to do.



Services and Security

For example: Samba (SMB), the file sharing protocol, allows users to view, download, and store files remotely.



The screenshot shows the Samba.org website homepage. At the top right is a search bar with the text "search samba.org:" and a red checkmark icon. Below the search bar is a large banner with the Samba logo and the tagline "opening windows to a wider world". On the left side, there is a vertical navigation menu with red buttons labeled "Home", "think Samba", "get Samba", "learn Samba", "talk Samba", "hack Samba", and "contact Samba". The main content area features a section titled "About Samba" with the text: "Samba is the standard Windows interoperability suite of programs for Linux and Unix. Samba is [Free Software](#) licensed under the [GNU General Public License](#), the Samba project is a member of the [Software Freedom Conservancy](#). Since 1992, Samba has provided secure, stable and fast file and print services for all clients using the SMB/CIFS protocol, such as all versions of DOS and Windows, OS/2, Linux and many others. Samba is an important component to seamlessly integrate Linux/Unix Servers and Desktops into Active Directory environments. It can function both as a domain controller or as a regular domain member." To the right of the "About Samba" section, there are two boxes. The top box is titled "Donations" and contains the text: "Nowadays, the Samba Team needs a [dollar](#) instead of pizza ;-)". The bottom box is titled "Latest News" and contains the text: "03 September 2019 Samba 4.11.0rc3 Available for Download This is the third release candidate of the upcoming Samba 4.11 release series." A small calendar icon showing the date 21 is also visible next to the "Latest News" title.

Finding and Stopping SMB Demo

If a malicious user is able to gain access to a shared folder, they can exfiltrate, alter, or delete sensitive files.

- In this example, the server has already been compromised.
- In the following demo, we will stop the SMB service, and then uninstall it from the system.



Finding and Stopping SMB Demo

This will require the following steps:



Listing all running services.



Identifying the Samba service in the list to confirm it's running, then stopping it.



Ensuring Samba doesn't start when the machine is started up.



Ensuring Samba is no longer running.



Uninstalling the Samba service completely.



Instructor Demonstration

Finding and Stopping SMB Demo



Activity: Managing Services

Your senior administrator wants you to audit the services being run by the server and shut down old and unused services.

Suggested Time:

25 Minutes



Time's Up! Let's Review.

Questions?



Service Users



Some services are not run by real users. They are run by specific **service users** that are dedicated to running their own specific service.

Service Users

Typically, when you install a service with the package manager, a service user is automatically created and configured.

Running services under a dedicated user offers several security benefits.

It makes it easier to start, stop, and manage the service, and control which files the service needs to access.



Service Users

A service user usually has a system UID less than 1000 and cannot log in to use a shell.



```
root:x:0:0:root:/root:/bin/bash
```

Service Users

Since service users aren't humans who need to log into and interact with the machine, it's best practice to ensure that users cannot log into an interactive shell using a service username.



Scenario: Setting Up and Adding Service Users Demo

Your senior administrator asked you to follow up on your uninstallation of unused services.

You must now ensure the services' corresponding users have also been removed from the system.

Previously, you disabled `vsftpd`, but its service user, `ftp`, still exists.



Scenario: Setting Up and Adding Service Users Demo

Your senior administrator also plans to install a security service called Splunk to collect and analyze logs for suspicious activity. Like Tripwire, Splunk makes it easier for admins and security personnel to detect and stop malicious behavior.

The screenshot shows the Splunk website homepage. At the top is a navigation bar with the Splunk logo and links for Products, Solutions, Why Splunk?, and Resources. On the right are links for Support, a search icon, a globe icon, a user icon, and a 'Free Splunk' button. The main content area features a large graphic with the text 'Turn Data Into Doing Powering Security, IT and DevOps' and a 'Free Trial' button. To the right is a preview of a Splunk dashboard with various charts and tables. At the bottom, there is a banner for '.conf21 splunk' with the text 'Missed .conf21 Virtual? On-demand content now available!' and a 'Learn More' button.

splunk> Products Solutions Why Splunk? Resources Support Search User Free Splunk

Turn Data Into Doing
Powering Security, IT
and DevOps

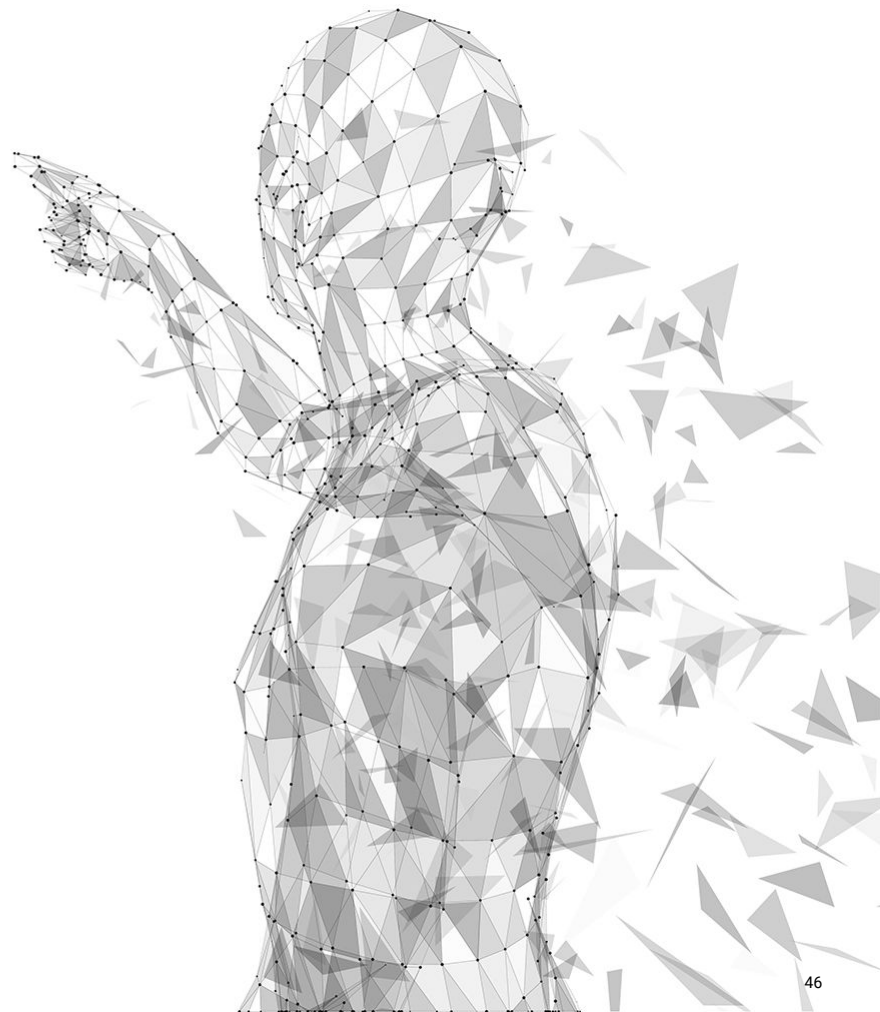
Free Trial

.conf21
splunk>

Missed .conf21 Virtual? On-demand content now available! Learn More

Scenario: Setting Up and Adding Service Users Demo

Your senior administrator told you that they'll handle the installation and configuration themselves, but have requested that you create a service user that they can use later.



Scenario: Setting Up and Adding Service Users Demo

Completing this task will require the following steps:

01

Delete

Deleting an old, unused service user with `deluser/`.

02

Create

Creating and validating a new service user with `adduser`.



Instructor Demonstration

Setting up and Adding Service Users



Activity: Service Users

Your senior administrator would like you to remove any old service users from the system and create a new user dedicated to running Tripwire.

- Use `adduser` and `deluser` with the correct flags to clean up the system and create this new Tripwire user.
- Tripwire can only be run as `root`, so you must add a line to the `sudoers` file to allow this.

Suggested Time:

25 Minutes



Time's Up! Let's Review.

Questions?



Homework

In this week's homework, you will practice all the hardening steps we learned this week, this time on a new system.

You will also run a few new tools: **chkrootkit** and **lynis**.



Questions?



*The
End*