## LESSON: Introduction to Linux

**Primer**

This is the first module for the Integrating Infrastructure Security course, and it contains two labs: the first lab focuses on “Meet Linux”, while the second lab is related to “Basic Folder and Files Manipulation”. We highly recommend that instructors plan ahead to allocate sufficient time for these labs to ensure students have ample opportunity to practice.

For this lesson and upcoming lessons, instructors are required to ensure the following activities are completed for each lesson

* Review the “Lesson Opener” and “Real World Scenario” with the learners prior to starting the module.
* Throughout the module, you will find “Consider the Real World Scenario” slides. Review the questions found on these slides, tie the concepts back to the scenario discussed at the start of the lesson as well as content you are presenting, and encourage the learners to share their thoughts.
* For each lesson, you will find a “Pulse Check” slide which is the opportunity for instructors to open a poll to gather feedback from the learners. Leave the poll open for about 1 minute and after you close the poll, share the results with the learners. Encourage the learners to share their thoughts. This information will help the instructors as well as the learners better understand where they are with regards to the lesson.
* Labs are to be demonstrated live for each module. The demonstration of labs is the top priority for the lead instructor. While demonstrating each lab, encourage students to participate and explore.
* At the end of each lesson, it is important to take a few minutes to review the key concepts for the lesson, provide guidance on what the learners can do to prepare for the next lesson, and wrap up with Q&A.

### Summary

In this lesson, learners will explore Linux, an open-source OS created by Linus Torvalds in the 1990s. They will uncover its distinct approach of emulating UNIX while adhering to the GNU General Public License. From versatile distributions rooted in a common kernel to efficient memory and process management, Linux's flexibility and reliability will be highlighted. Learners will grasp the hierarchical file structure, essential navigation and management commands, and the power of the command-line interface (CLI). By understanding Linux's core principles and commands, such as "whoami" and "uname," learners will lay the foundation for effective system interaction and administration.

### Objectives

* Explain Linux’s key features and characteristics.
* Identify popular Linux distributions and their main characteristics.
* Describe the basic components of Linux distributions.
* Explain the advantages of dedicated environments.
* Define Linux’s file system structure.
* Illustrate and analyze Linux directory structure.
* Explain the naming conventions for files and directories.
* Explain the use of commands and the CLI in Linux.
* Define Linux shells.
* Compare and contrast the sh and bash shells.
* Illustrate the command syntax.
* Explain Linux system commands.
* Identify the commands used for navigating the file system.
* Illustrate the commands used for creating, deleting, moving, renaming, and duplicating files and folders.

### Lesson Activities and Teaching Strategies

|  |  |  |
| --- | --- | --- |
| Estimated Time | Lesson Portion | Directions |
| 5 min | **Lesson Opener:**  Linux Overview | * Introduce learners to Linux and its importance in Cybersecurity. * Share the background of Linux's creation by Linus Torvald, a Finnish student, in the early 1990s. * Describe the role of the kernel in managing communication between applications and hardware. * Discuss how the kernel's development marked the beginning of Linux and its evolution. * Highlight how Linux's versatility and adaptability have contributed to its widespread adoption. * Introduce the two main interfaces of Linux: The command-line interface (CLI) and the graphical user interface (GUI). * Describe the CLI as text-based, explaining its significance in system administration and configuration. * Explain the GUI's role in providing user-friendly visual interactions. * Mention that not all Linux distributions offer a GUI option. * Define Linux distributions as unique systems created by combining the kernel with software and interfaces. * Provide examples of various distributions tailored to specific needs. |
| 5 min | **Real World Scenario:**  Linux Overview | * Remind learners about the lesson compassion, then review the real world scenario challenge and inform learners that you will be constantly coming back to this scenario throughout the lesson to discover how to solve and apply concepts to this real situation. |
| 2 | **Lesson Companion:**  Linux Overview | * Review the lesson companion, and inform learners that you will be constantly coming back to this scenario throughout the lesson to discover how to solve and apply concepts to this real situation. |
| 20 min | **Cyber Uncovered:**  Linux Distributions | * Begin by explaining the concept of open source, emphasizing that it involves sharing source code for public use, modification, and improvement. * Describe how the open-source philosophy enables the development of diverse Linux distributions tailored to specific needs and use cases. * Explain that Linux is a family of operating systems (OSs) built using UNIX architecture. * Discuss how this philosophy fosters collaboration and innovation among developers and communities. * Relate open-source principles to Linux and its significance in shaping the OS landscape. * Explain how developers can modify and customize distributions to create unique systems. * Discuss the benefits of this diversity in offering solutions for various industries and user preferences. * Introduce the GNU licensing framework and its role in promoting open-source software. * Explain how modified code under GNU licenses can be used privately or shared publicly, contributing to the community's benefit. * Emphasize that Linux-based OSs are integrated into a wide range of products, such as smartphones, computers, cameras, and cars. * Showcase the Linux logo and the symbolic significance of Tux, the penguin mascot, who represents the open-source spirit. * Highlight that most Linux distributions are available for free, aligning with the open-source philosophy. * Emphasize that the kernel is the core component bridging hardware and the OS. * Outline the kernel's key responsibilities:   + Ensuring effective communication between hardware components.   + Allocating and managing system memory.   + Controlling program execution through process management.   + Facilitating requests between the OS and hardware through system calls. * Highlight the significance of Debian's legacy, Ubuntu's user-friendly approach, RHEL's commercial applications, and Fedora's focus on innovation * Present the visual representation of the Linux open-source kernel, available for free download, and emphasize its role as the foundation for creating new distributions. * Be prepared to discuss the implication of the real world scenario presented at the beginning of class on Linux Distributions. There are specific prompts that you should ask learners to reflect on to apply this concept to the real world scenario. |
| 5 | **Real World Scenario:** Linux Distributions | * Review the lesson companion, and inform learners that you will be constantly coming back to this scenario throughout the lesson to discover how to solve and apply concepts to this real situation. |
| **5 min Break** | | |
| 20 min | **Cyber Uncovered:**  File System Structure | * Begin by highlighting the significance of understanding the file system structure in Linux systems. * Explain that the structure is consistent across most Linux distributions, facilitating a unified user experience. * Describe the hierarchical nature of Linux file systems, in which directories are organized in a tree-like structure. * Emphasize that each directory serves a specific purpose and contains files related to that purpose. * Introduce the concept that every operation in Linux is associated with one or more files. * Explain that these files are located within specific folders, contributing to a well-organized system. * Define the root folder as the topmost directory in the file system hierarchy. * Discuss its significance as it contains essential directories that the system relies on for core operations. * Present a visual diagram showcasing the main Linux root directories. * Explain the purpose of each key directory, such as /root for the super user's home directory, /home/ for non-root home folders, /etc for system configuration files, /tmp for temporary files, /boot for the kernel and bootloader, /var and /srv for server data, /proc and /sys for system information, /lib for library directories, and /media and /mnt for external file systems and mount points. * Explain the concept of the single-root inverted tree structure in Linux file systems. * Describe how the file system starts at the root directory indicated by a forward slash (/) and that all paths are delimited by forward slashes. * Emphasize the importance of adhering to naming conventions in Linux. * Explain that all characters are valid in file and directory names, except for the forward slash. * Caution against using special characters and emphasize the need to quote characters when necessary. * Highlight the case-sensitive nature of file and directory names, sharing examples of different variations and their distinctions. * Be prepared to discuss the implication of the real world scenario presented at the beginning of class on File system structure. There are specific prompts that you should ask learners to reflect on to apply this concept to the real world scenario. |
| 5 | **Real World Scenario:** File System Structure | * Review the lesson companion, and inform learners that you will be constantly coming back to this scenario throughout the lesson to discover how to solve and apply concepts to this real situation. |
| 5 | **Pulse Check** | * Before you launch the pulse check, explain each section clearly, and encourage the learners to participate in the survey. * After administering the survey, share the poll results with learners and ask learners to provide feedback * Encourage learners to attend office hours with the associate instructor. |
| **5 min Break** | | |
| 20 min | **Cyber Uncovered:**  Command Structure and Shell Types | * Start the lesson by discussing the significance of commands in Linux for system configuration and data representation. * Explain how commands automate processes and tasks within the operating system, enhancing efficiency. * Emphasize that commands consist of a combination of letters, numbers, and characters. * Illustrate how commands are structured to carry out specific actions, contributing to the functionality of the system. * Introduce the CLI as a text-based interface used for interacting with the system. * Explain its role in receiving textual commands from users and providing textual feedback. * Highlight that while graphical interfaces exist, many system configurations are carried out via the CLI. * Provide context by comparing the Linux CLI to similar interfaces like the command prompt or PowerShell in Windows. * Emphasize the extended functionality and versatility of the Linux CLI. * Define a Linux shell as a command-line interface that facilitates communication with the operating system. * Introduce common shells, specifically the sh (Bourne shell) and bash (Bourne Again SHell). * Discuss the sh shell as one of the earliest UNIX shells with simplicity and wide availability. * Highlight the advanced features of the bash shell, including command-line editing, auto-completion, and more. * Mention other shells like zsh, dash, and tcsh, with their availability based on the distribution. * Explain the syntax of Linux commands as "Command [options] [arguments]." * Discuss the versatility of commands, including displaying data, configuring the system, creating files, and running programs. * Describe how options modify specific actions and control command outputs, indicated by dashes. * Provide a practical example, such as the "ping" command, to test network connectivity. * Break down the example into its components: Command name, options, and arguments. * Explain the purpose of each component, like how the "-c" option specifies the number of packets and the argument represents the target host. * Be prepared to discuss the implication of the real world scenario presented at the beginning of class on Common Structure and Shell Types. There are specific prompts that you should ask learners to reflect on to apply this concept to the real world scenario. |
| 5 | **Real World Scenario:** Common Structure and Shell Types | * Review the real world scenario challenge and inform learners that you will be constantly coming back to this scenario throughout the lesson to discover how to solve and apply concepts to this real situation. |
| 20 min | **Lab:**  Meet Linux | * Remind learners to use this lab to practice and apply the concepts they have learned throughout the day. * Learners will receive direct feedback on their lab to properly assess their knowledge and determine where they might need additional assistance. |
| **5 min Break** | | |
| 20 min | **Cyber Uncovered:**  Introduction to the CLI and Basic Linux Commands | * Begin the lesson by explaining the significance of system commands in Linux, emphasizing their role in executing specific tasks and functions within the operating system. * Illustrate how system commands can be executed in the terminal to achieve various outcomes. * Introduce essential system commands like "whoami" and "uname." * Describe how the "whoami" command displays the currently logged-in user's username, helping to verify identity within the system. * Explain that the "uname" command provides system information, including OS name, kernel version, and hardware details, offering insights into the system's configuration. * Discuss the two main ways to learn about command operations: "--help" and "man [binary]." * Explain that "--help" provides a brief description and usage examples, while "man" displays comprehensive manuals for commands, offering detailed information about their operation. * Provide an overview of file system navigation, comparing it to browsing a file explorer. * Introduce the commands "cd" (change directory) and "pwd" (print working directory). * Emphasize the importance of navigating efficiently to perform tasks and locate files. * Present the various variations of the "cd" command, such as navigating with a full path, relative path, and using virtual paths. * Explain the concept of moving back in directories using ".." and navigating to the home directory using "~". * Introduce the "ls" command to list a directory's content. * Explain common options like "-lh" for human-readable format, "-a" for hidden files, and "-R" for recursive listing. * Mention that modern shells often use color encoding for enhanced visualization. * Explore the detailed output of the "ls -l" command. * Break down each section of the output, including file permissions, hard links, owner, group, file size, modification date, and name. * Provide examples to help students interpret the information effectively. * Explain the process of creating files and folders using the "touch" and "mkdir" commands. * Highlight the ability to create multiple files/folders in a single command. * Introduce the "-p" flag for generating nested folders and emphasize its usage * Describe the usage of the "rm" command for deleting files and folders. * Explain options like "-d" for folders and "-r" for recursive deletion. * Discuss the "rm -rf" command for forceful removal, cautioning about its potential impact. * Present the "mv" command for moving or renaming files and "cp" for copying files or directories. * Describe the structure of both commands and their common syntax: [source] [destination]. * Explain how relative and full paths can be used for source and destination. * Be prepared to discuss the implication of the real world scenario presented at the beginning of class on Introduction to the CLI and Basic Linux Commands. There are specific prompts that you should ask learners to reflect on to apply this concept to the real world scenario. |
| 5 | **Real World Scenario:** Introduction to the CLI and Basic Linux Commands | * Review the real world scenario challenge and inform learners that you will be constantly coming back to this scenario throughout the lesson to discover how to solve and apply concepts to this real situation. |
| 20 min | **Lab:**  Linux CLI | * Remind learners to use this lab to practice and apply the concepts they have learned throughout the day. * Learners will receive direct feedback on their lab to properly assess their knowledge and determine where they might need additional assistance. |
| 15 | **Lesson Closure** | * Encourage learners to read ahead of time * Provide learners additional resources to read / practice and assign homework (e.g., future labs) before you demonstrate the labs during the next class * Spend some time to highlight what are the key takeaways from today’s lesson * Important topics covered during the class includes   + Highlight the key takeaway regarding the Linux overview     - Linux was developed by Linus Torvolds in early 1990s with the goal to emulate the Linx experience     - Linux kernel facilitate communications between the hardware and the OS, memory, and system calls     - Linux is used in many other things such as automotive, and IoT devices   + Highlight the key takeaway regarding Linux distributions     - The open-source philosophy was created to foster collaborations and innovations     - Linux distributions vary in features despite having the same Linux core kernel     - Popular distributions include REH, Debian   + Highlight the key takeaway regarding Linux distributions Command Structure and Shell Types     - Linux supports various types of shells, including Bash (Bourne Again Shell), Zsh (Z Shell), Ksh (Korn Shell). Each shell offers unique features and syntax, catering to different user preferences and scripting needs     - Linux commands provide numerous functions for system administration, such as whoami for displaying the current user's name, uname for showing system information, ls for listing directory contents, rm for removing files, cp for copying files, and touch for creating empty files or updating file timestamps |
|  | Add Additional Time Filler | * Review using Kahoot or other similar platforms * Conduct interview preparation conversations * Continue discussions on real-world scenarios * Demonstrate how to create users in Linux and grant them permissions * Discuss different career paths in cybersecurity and highlight the roles that require Linux skills |