Cristina Nita-Rotaru

CY2550: Foundations of Cybersecurity
Section 03

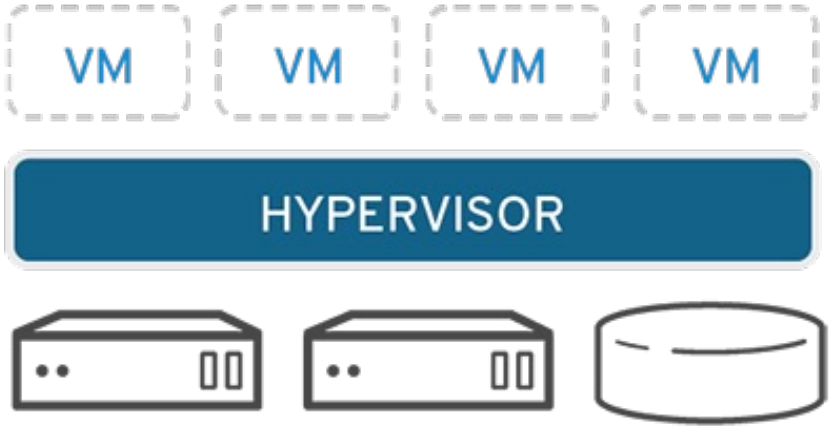
Linux. Git

Linux shell and commands

Linux

- Operating system (OS): the program which starts up when you turn on your computer and manages all the available resources on a computer (CPU, memory, hard disk, etc)
  - Controls hardware
  - Runs applications
  - Manages data and files
- UNIX was created in the late 1960s, originally created and implemented by Ken Thompson of Bell Labs, later many versions and mutations
- Linux is a freely distributed UNIX-based OS created by Linus Torvalds in 1991
  - Many distributions: Ubuntu, Debian, etc
  - Packet managers that make installation easier: apt for Linux

Virtualization



- Concept created around 1960, but adopted around 2000
- VM - virtual machine or guest
- HYPERVISOR (virtual machine monitor) – program that runs on the host OS and separates the physical resources from the VM
- Examples of hypervisors: VMWare, KVM, QEMU, VirtualBox

Working environment for class

- Linux Ubuntu 20.4 running in VirtualBox
- For M1 MAC users – VMWare or UTM

What is a “shell” ?

- Shell: A program that provides a "command line" interface which allows the user to enter commands which are translated by the shell into commands that are understood by the kernel so the kernel can execute them
  - Available with the program called terminal
- Default shell is bash
- Several shells, the user can chage their shell with command **chsh**
  - tcsh - Turbo C SHell
  - csh - C SHell
  - ksh - Korn SHell
  - bash - Bourne Again SHell
  - sh - SHell

Aliasing commands

- The **alias** command assigns a command, possibly with many options and flags, to another name. Usually it is a shorter name or one that is easier to remember.

Setting up an alias:

- **tcsh** syntax:
  - **alias <aliased name> <original command>**
- **bash** syntax:
  - **alias <aliased name>=<original command>**

Environment variables

- The shell maintains information about its current state in the environment variables hold this information.
- You can view them with the command env
- You can change them with the command export (in bash)

Important environment variables

- **HOME** - your home directory.
- **USER** and **LOGNAME** - your login ID.
- **HOSTNAME** - the name of the host computer.
- **PWD** - the current working directory.
- **MAIL** - where your mail is located.
- **PATH** - a list of directories in which to look for executable commands.

- Certain applications and commands may communicate with the shell and reference the environment variables that it maintains.

Bin directories

- Program files or commands, also called binary executable files and script files
- Typically stored in bin (short for binary) directories throughout the system.
- If you take a look at the paths are stored in your $PATH environment variable, you will notice that many of these directories end in .../bin.

bash

- Almost any home installation of Linux defaults to the bash shell
- bash is one the many GNU.org (http://www.gnu.org) projects
- Each shell has a configuration file that is run when you start the shell, for bash it is called bashrc, it is a hidden file you can see with ls –a
- **bash manuals:**
  - A comprehensive online manual is provided at https://www.gnu.org/software/bash/manual/bash.html

Shell scripting

- Usually shells are interactive that mean, they accept command as input from users and execute them.
- Shells also take files called **Shell Scripts** that contain a set of commands to be executed
- Shell scripts are written in shell scripting languages that are shell-specific.

UNIX commands

command options(s) filename(s)

- *Command:* name of the utility or program to execute
  - Convention in UNIX, always lowercase
- *options* modify the way the command works
  - typical to have a hyphen followed by a single character, such as **-a**.
  - a common convention under Linux to have options that are in the form of 2 hyphens followed by a word or hyphenated words, such as **--color** or **--pretty-print**.
- The *filename* is the last argument for a lot of UNIX commands. It is simply the file or files that you want the command to work on.
- Not all commands work on files

Man Pages

- The **man** command allows you to access the MANual pages for a UNIX command.
- To get additional help on any of the commands listed below, you can always type **man name_of_command** at the command prompt.
- Examples:
  - **man ssh**
  - **man passwd**

Commands

- **ls :** lists the contents of a directory
  - l : long directory listing
  - a : lists all files, including files which are normally hidden
  - F : distinguishes between directories and regular files
- **pwd** : prints the current working directory
- **cd** : changes directories
  - The difference between relative and absolute paths.
  - Special characters **.**, **..**, and **~**.
- **mkdir** : creates a directory
- **rmdir** : removes a directory (assuming it is empty)
  - If you get an error that the directory isn't empty even though it looks empty, check for hidden files.

Commands

- **touch** : creates an empty file with the specified name, or if the file already exists it modifies the timestamp.
- **rm** : removes a file.
  - f : force deletion
  - r : recursive deletion
- **mv** - moves a file, or renames a file
  - f : forces overwrite, if the destination file exists
- **cp** - copies a file, leaving the original intact
  - f : forces overwrite, if the destination file exists
  - r : recursive copying of directories

Commands

- **cat** : shows the contents of a file, all at once
- **more** : shows the contents of a file, screen by screen
- **less** : also shows the contents of a file, screen by screen
- **head** : used to show so many lines form the top of a file
- **tail** : used to show so many lines form the bottom of a file

Commands

- **lpr** : prints a file
- **alias** : creates an alias for a command.
  - Aliases can be placed in your **.cshrc** login script.
  - Example: alias rm 'rm –i'.
- **date** : shows the date and time on the current system
- **who** : used to print out a list of users on the current system
- **hostname** : prints the hostname of the current computer
- **whoami** : prints your current username

passwd

- The **passwd** command changes your UNIX password.
- This command is an example of a “no argument” command. Thus, the format of this command is just the command name itself.

  passwd
- passwd is an interactive command, as once we have typed it, we need to interact with it.

scp

- The **scp** command is a way to copy files back and forth between multiple computers.
- Formats for this command:
  - **scp path/local_name username@hostname:path/remote_name**
  - **scp username@hostname:path/remote_name path/local_name**

UNIX Pipe (|)

- The pipe (|) creates a channel from one command to another, i.e. connects the output from one command to the input of another command
- It is useful to link commands together to perform more complex tasks that would otherwise take multiple steps
- Examples:
  - Count the number of users logged onto the current system.
    - The **who** command will give us line by line output of all the current users.
    - We could then use the **wc -l** to count the number of lines...
    - **who | wc –l**
  - Display long listings in a scrollable page.
    - The **lpq** command will give us a list of the waiting print jobs.
    - **lpq | less**

Commands

- **ps** : lists the processes running on the machine.
  - **ps -u** *username* lists only your processes.
  - **ps -a** : lists all processes running on the machine.
  - The PID column of the listing, provides the information required by the kill command.
- **kill** : terminates a process
  - **kill** *process_id* : sends a terminate signal to the process specified by the *process_id* (PID).
  - In cases where the terminate signal does not work, the command "**kill -9** *process_id*" sends a kill signal to the process.
- **nice** : runs a process with a lower priority.

SSH, HTTPS

ssh (secure shell)

- '*Secure shell is a de facto standard for remote logins and encrypted file transfers.*' [SSH communications inc.]
- Created in 1995 by Tatu Ylonen, a researcher at Helsinki University of Technology, Finland
- **ssh** allows you to securely connect to a remote computer
- Formats for this command:
  - **ssh hostname**
  - **ssh username@hostname**
  - **ssh hostname -l username**
- If you do not specify the username, it will assume that you want to connect with the same username that you have on this local computer
- Several protocols: ssh1, ssh2. openssh

HTTPS

- HTTP – protocol that allows communication for web applications
- HTTPS – HTTP over TLS – a secure version of HTTP
- It it the protocol that is used when you access web applications

- Different applications will require you to use either ssh or HTTPS
- For example when connecting to github you can use either ssh or HTTPS
- Both require public keys for secure communication

Git and github

Version control systems (VCS)

- Provide support for software development
- Backup
- Recovering deleted files
- Access from multiple machines
- Multiple writes on the same repository
- Merging conflicts
- Examples of VCS: Subversion (svn), mercurial, Visual SourceSafe (discontinued), git – the most popular

Git

- Created by Linus Torvalds, creator of Linux, in 2005
  - Came out of Linux development community
  - Designed to do version control on Linux kernel
- Git website: http://git-scm.com/
- Free on-line book: http://git-scm.com/book
- Reference page for Git: http://gitref.org/index.html
- Git tutorial: http://schacon.github.com/git/gittutorial.html
- Git for Computer Scientists:

http://eagain.net/articles/git-for-computer-scientists/

Git and github

- You can run git on your machine to maintain versions of your files and to recover deleted files or different versions
  - git add filename
  - git commit filename –m “comment”
- You can also save your work on a remote repository, such that you can access it from any machine or work collaboratively with somebody else
  - Github is such a repository
- You communicate with the repo with
  - git clone – when you get a copy of the repo
  - git push – when you want to push updates to repo
  - git pull – when you want to get updates from the repo

- git log
  - Shows all the local commits
- git status
  - Shows what files are in the local repository
- git restore filename
  - Restores the filename from the repository; useful if you deleted a file
- git restore –source *identifier* of commit filename
  - Restores the filename from the commit identified by *identifier*

Example of commit identifier

Author: XXX <xxx@hotmail.com>

Date: Sun Sep 11 02:27:53 2022 +0200

Added the final feedback from reviewers

commit f6f5bb6606b61a53976471019ff0a1206dc4ad2c

Working environment for class

- Need to have your work on github
- Link github with gradescope so you can submit your assignments
- Make sure your most recent version is pushed in github before submitting your assignment