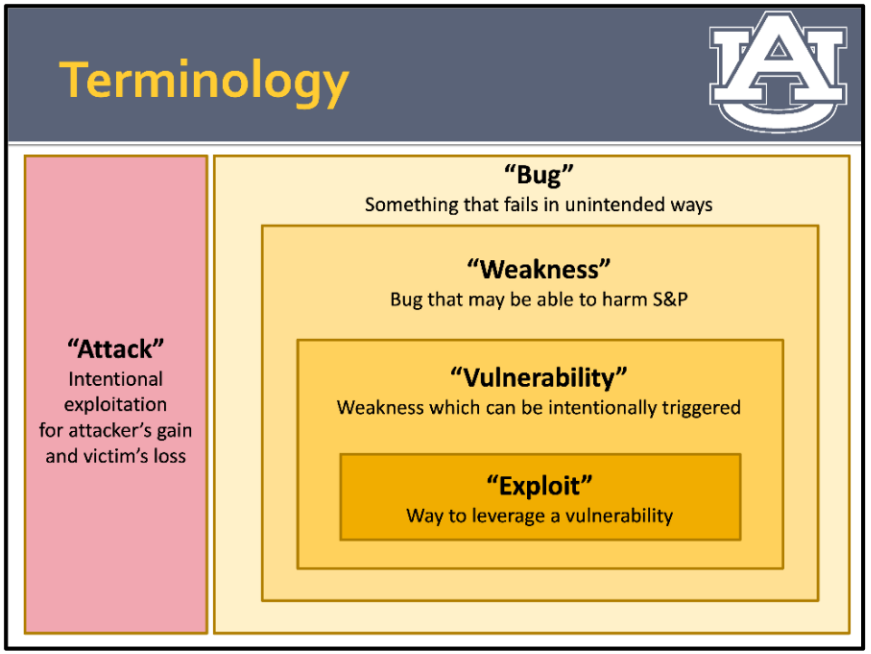
Computer and Network Security

Lecture 14:
OS Security & Isolation

COMP-5370/6370
Fall 2024

Software Facts of Life

- Software has bugs
- Some bugs are weaknesses
- Some weaknesses are vulnerabilities
- Some vulnerabilities can be exploited
- Someone has an interest in exploiting others for gain



Do you trust other people's code?

- Linux kernel: 27.8M lines of C
  - Pointer math, raw byte buffers, etc.
  - C99 compliant code lacks features like "smart pointers", threads, implicit bounds-checking,…
  - Written, reviewed, and tested by those strange and mythical people known as "kernel devs"

Do you trust other people's code?

- Do you use an HP computer?
  - Android OS or Google's Android Apps?
  - Google services (GMail, Docs, Calendar, etc)?

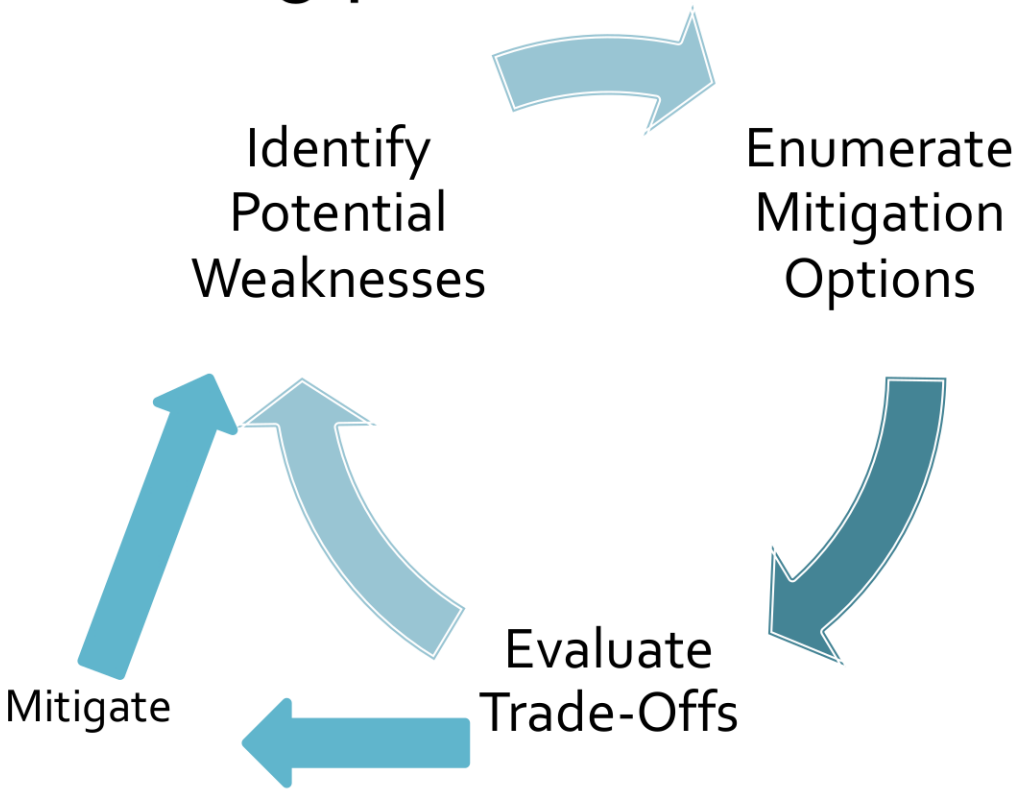
Finster Professor

Shady Professor

Profesor Turbio

Threat Modeling

A systematic approach to analyzing and understanding potential weaknesses.



During Planning

- Create a system that is defendable

Principle of Least Privilege

The **Principle of Least Privilege** is that access to data & resources is limited those who need for routine, authorized purposes.

- "Specific users can only run specific apps"
- "Normal users aren't root-users"
- "routine" means **ROUTINE**
  - Having everyday access for something that happens once every 3 years is not recommended

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 validates that app Y is allowed to use permission Z

Applying Principles

- **Security Model** – An abstraction to *subjects*, *permissions*, and *objects* to allow reasoning about S&P properties.
- **Security Policy** – The mapping of *subjects,* permissions, and *objects* to implement the security model.
- **Security Mechanism** – The technical measure that enforces the security policy.

Access Control Model

Formal Models

- Discretionary Access Control (DAC)
- Mandatory Access Control (MAC)
- Role-based Access Control (RBAC)
- Bell-LaPadula
- <many, many more>

The real-world is a mixture of all.

Filesystem Example

Subjects

Users

Groups

Permissions

Read

Write

Execute

Objects

"Files"(data, binary, device)

Hierarchies (multiple files)

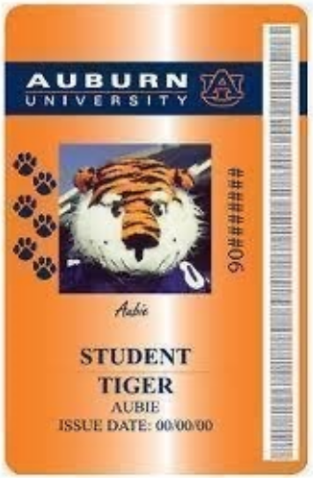
Access Control Policy

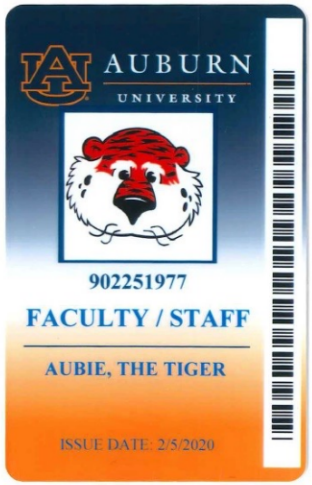
Create a logical representation of who has access to what file.

- Enumerate users
- Enumerate files
- Give specific users specific access to specific files
- Default: no-access

		Objects			
		File 1	File 2	File 3	File 4
Subjects	Alice	read	read/ write	no access	no access
	Bob	read	read/ write	no access	no access
	Carol	read	write	read/ write	read/ write
	Wendy	read/ write	read/ write	read	read

During Operation





- Create a system that is defendable
- Maintain the mechanisms for defending it

Subject Tracking Mechanism

Implement the policy in the real-world.

- Users assigned **user id**
- Users can **masquerade** as other users
- Users assigned to groups for simplicity (logic/mgmt)
- Groups are **meta-users** in some but not all aspects

```
user@desktop:~$ whoami; id -u
user
1000
```

```
user@desktop:~$ sudo whoami
[sudo] password for user:
root
```

```
user@desktop:~$ groups
user adm faculty
```

```
user@desktop:~$ $ sudo -u adm bash -c 'hi'
Sorry, user adm is not allowed to execute '/usr/bin/bash
user@desktop:~$ $ sudo -u root bash -c 'hi'
hi
```

User Permission Tracking

- 3 permission bits per object (RWX)


```
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
```

OS Mediation of File Access

- Who is trying to act?

```
user@desktop:~$ whoami; id -u
user
1000
```

- What are they trying to act on?

```
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
```

- What are they trying to do?

- Allow action or not?

Processes as Subjects

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

Do you trust other people's code?

- Do you use an HP computer?
  - Android OS or Google's Android Apps?
  - Google services (GMail, Docs, Calendar, etc)?

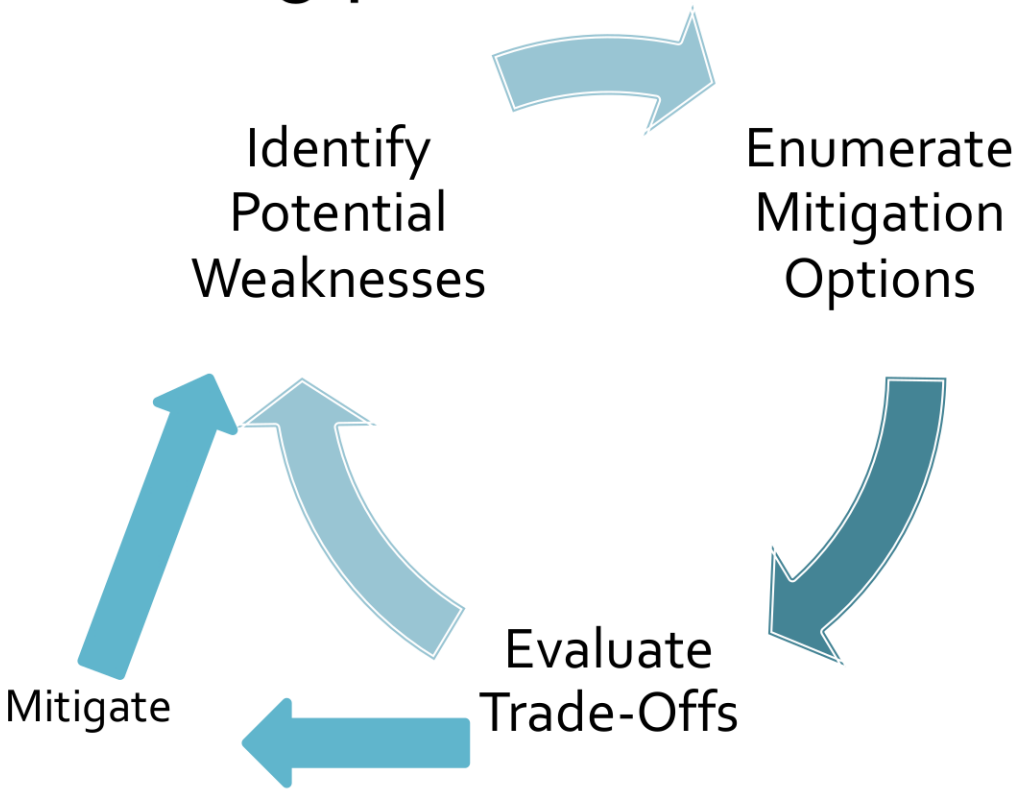
Finster Professor

Shady Professor

Profesor Turbio

Threat Modeling

A systematic approach to analyzing and understanding potential weaknesses.

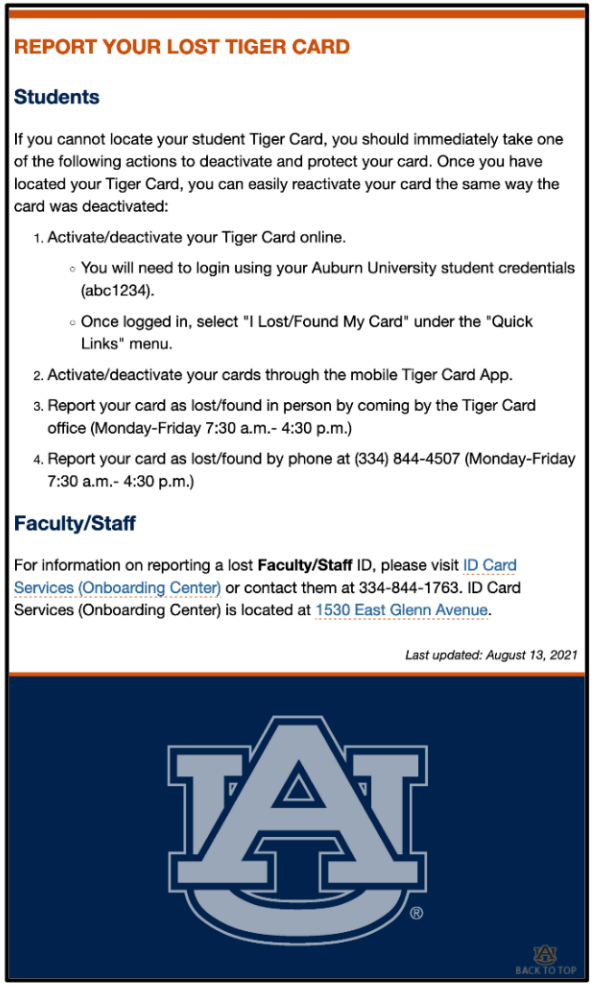


Architectural Defenses

An **architectural defense** is one which is both generic in terms of implementation and focused on *isolating* a potential misbehaving application or process.

- Not tied to a specific application or attack
- ***Can defend against unknown attacks in the future due to generality***

During Un-Wanted Events



REPORT YOUR LOST TIGER CARD

Students

If you cannot locate your student Tiger Card, you should immediately take one of the following actions to deactivate and protect your card. Once you have located your Tiger Card, you can easily reactivate your card the same way the card was deactivated:

1. Activate/deactivate your Tiger Card online.
   - You will need to login using your Auburn University student credentials (abc1234).
   - Once logged in, select "I Lost/Found My Card" under the "Quick Links" menu.
2. Activate/deactivate your cards through the mobile Tiger Card App.
3. Report your card as lost/found in person by coming by the Tiger Card office (Monday-Friday 7:30 a.m.- 4:30 p.m.)
4. Report your card as lost/found by phone at (334) 844-4507 (Monday-Friday 7:30 a.m.- 4:30 p.m.)

Faculty/Staff

For information on reporting a lost **Faculty/Staff** ID, please visit ID Card Services (Onboarding Center) or contact them at 334-844-1763. ID Card Services (Onboarding Center) is located at 1530 East Glenn Avenue.

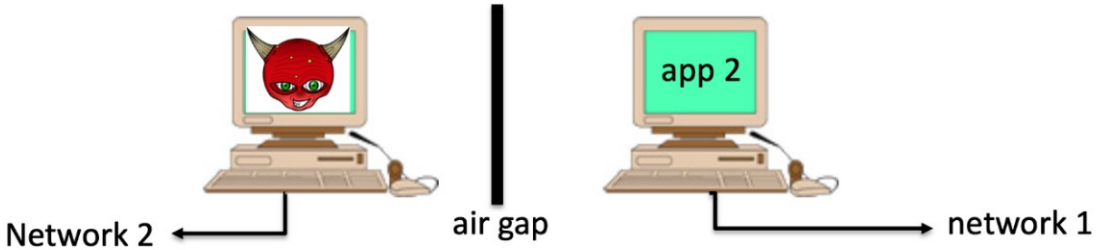
Last updated: August 13, 2021

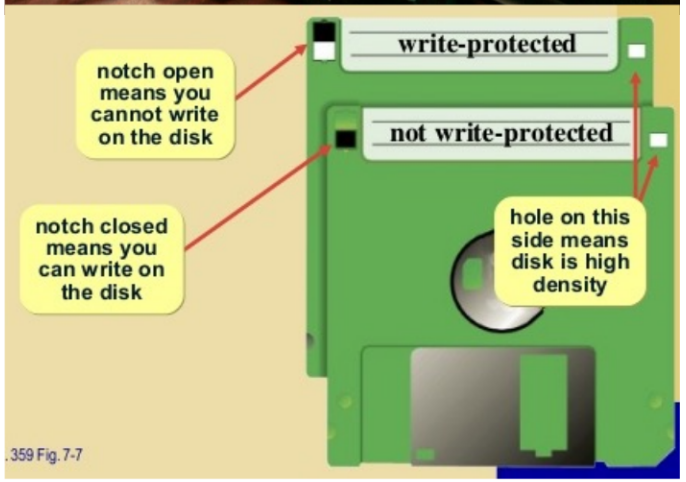
BACK TO TOP

- Create a system that is defendable
- Maintain the mechanisms for defending it
- Train and educate non-experts
- Make safety easy

Defense: Air Gap Hardware

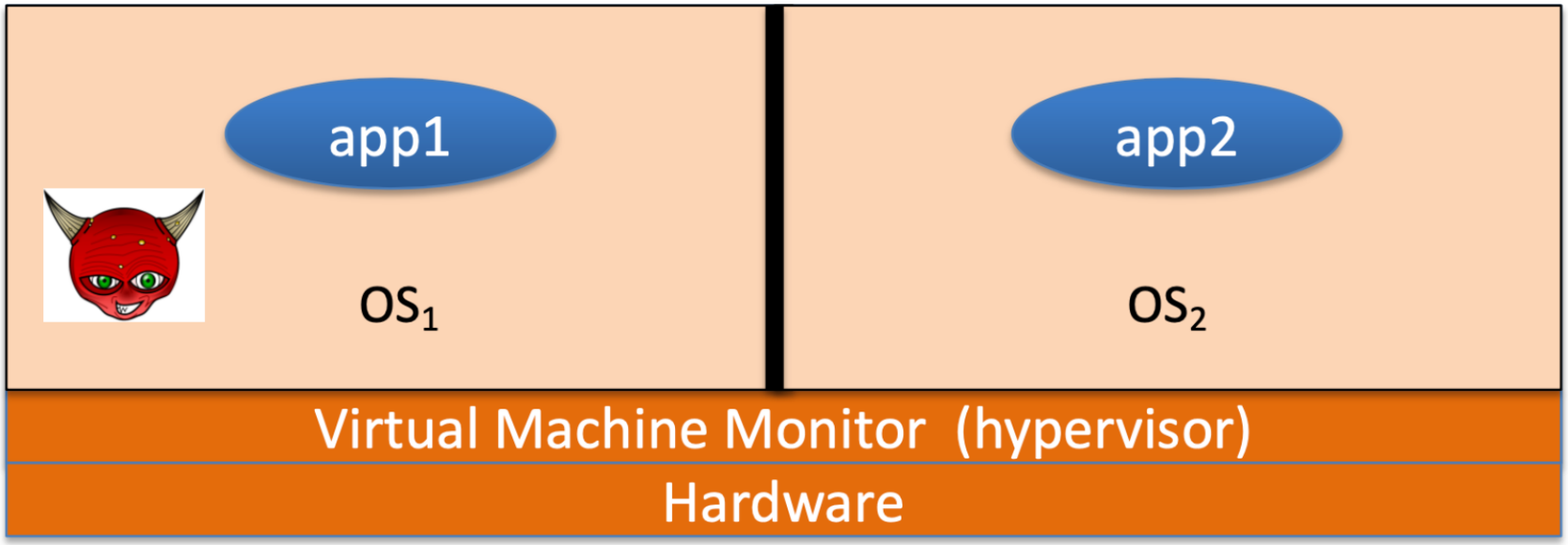
- Physically separated systems/networks
- No logical interaction across boundary
- Physical data transfer across boundary





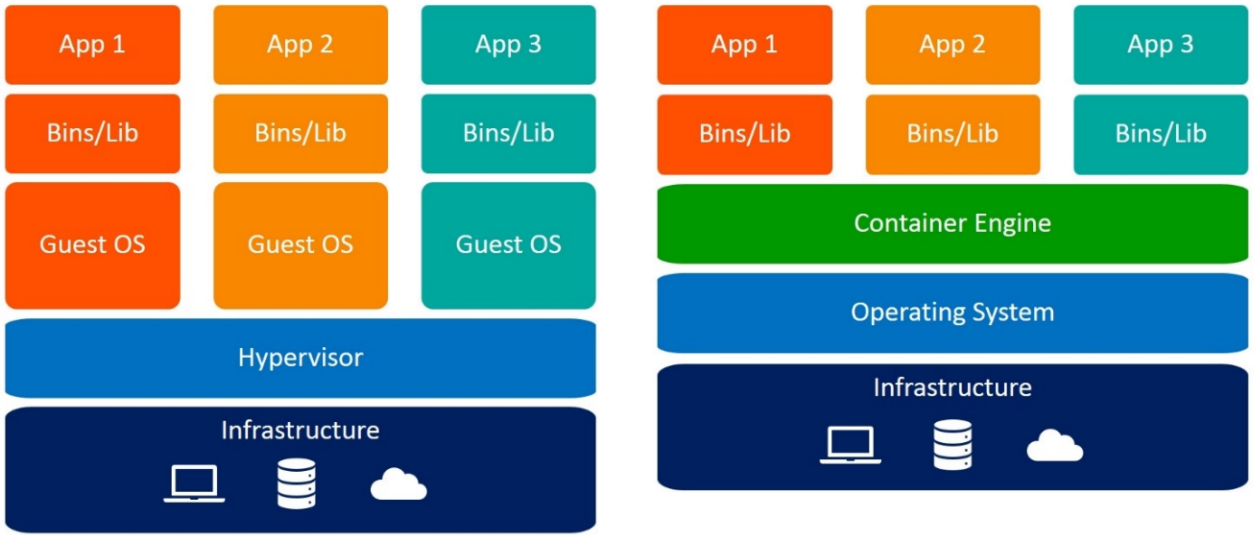
Defense: Virtual Machines

- OS-level isolation
  - Similar to dual-boot but simultaneous
- A “hypervisor” coordinates hardware access
- OSes are isolated (Linux VM on Windows)



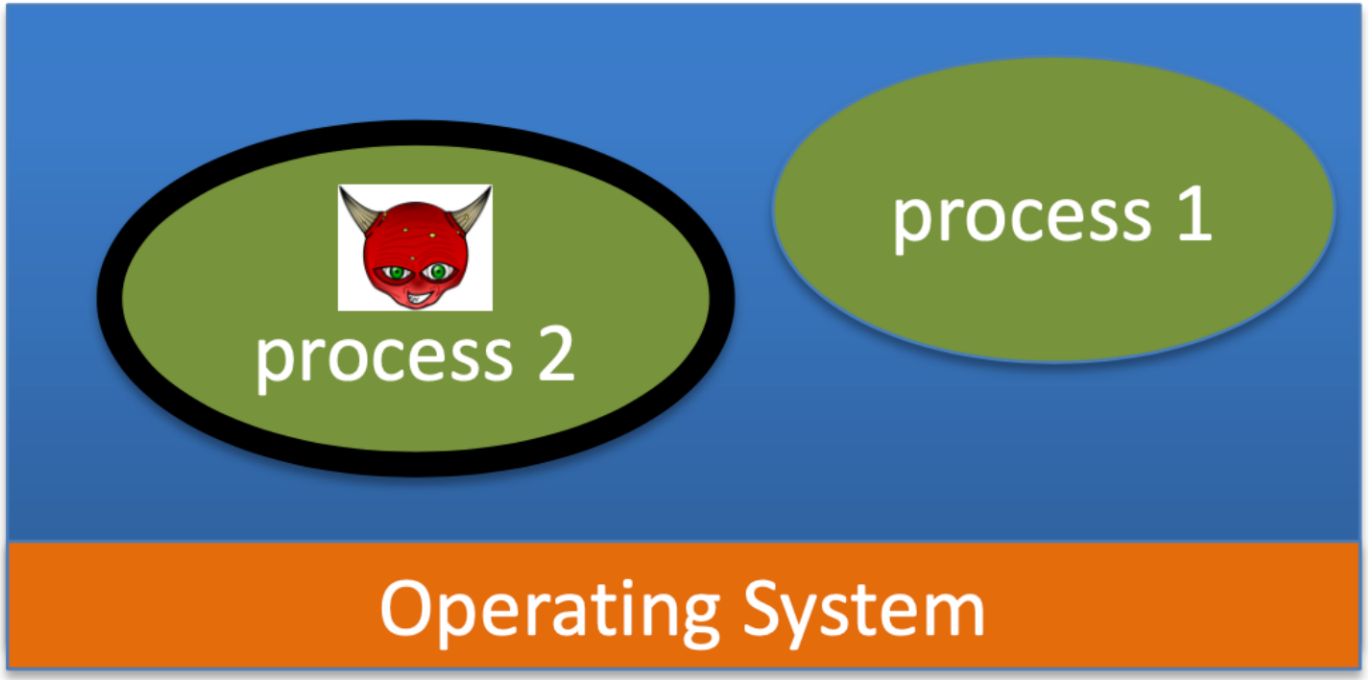
Defense: Containers

- Perspective-level isolation
- Shared kernel w/ isolated perspective
  - Each container thinks it's the only thing running on the entire computer



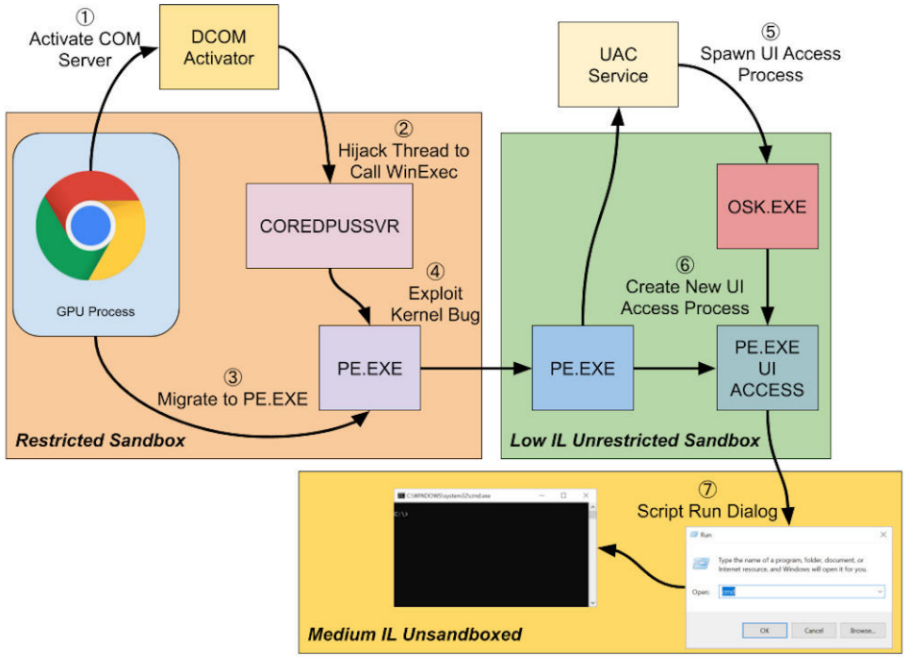
Defense: Process Isolation

- Memory-level isolation
- Managed via OS scheduler
- Extremely efficient due to virtual memory



Defense: Thread Sandbox

- Logic-level isolation (code logic)
- Threads interact via memory & IPCs
- Tainted threads can be killed and restarted

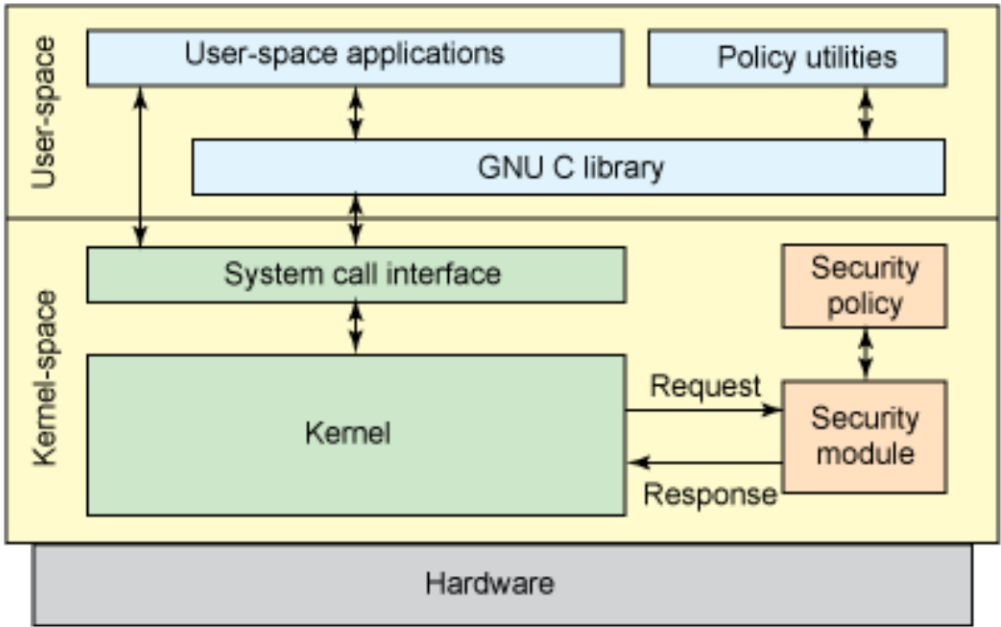


Defense: Thread Sandbox

- "Do one thing and do it safely"
- A "policy engine" can blindly enforce data interactions and data exchanges
- Very useful for web browsers
  - Request content, run JS, render image, etc are *very* different things with predictable inputs and outputs
- Better than nothing but is exceptionally difficult to internally isolate behavior

Defense: SELinux/App Armor

- Permission-level isolation
- **HEAVILY** patched set of kernel modules
- "Know what an application is suppose to do and don't let it do anything else."
  - `ls` doesn't need network access
  - Print driver doesn't need keystrokes



Defense: Use-Specific HW

- Required to install a shady VTC program?
  - Get a $100 tower from 2012 and $20 monitor
- Need to test a really sketchy app?
  - Get a $30 Android phone with Wifi
- Need to install a known-spyware extension?
  - Chromebooks are $100 on-sale




What do you think?

avast

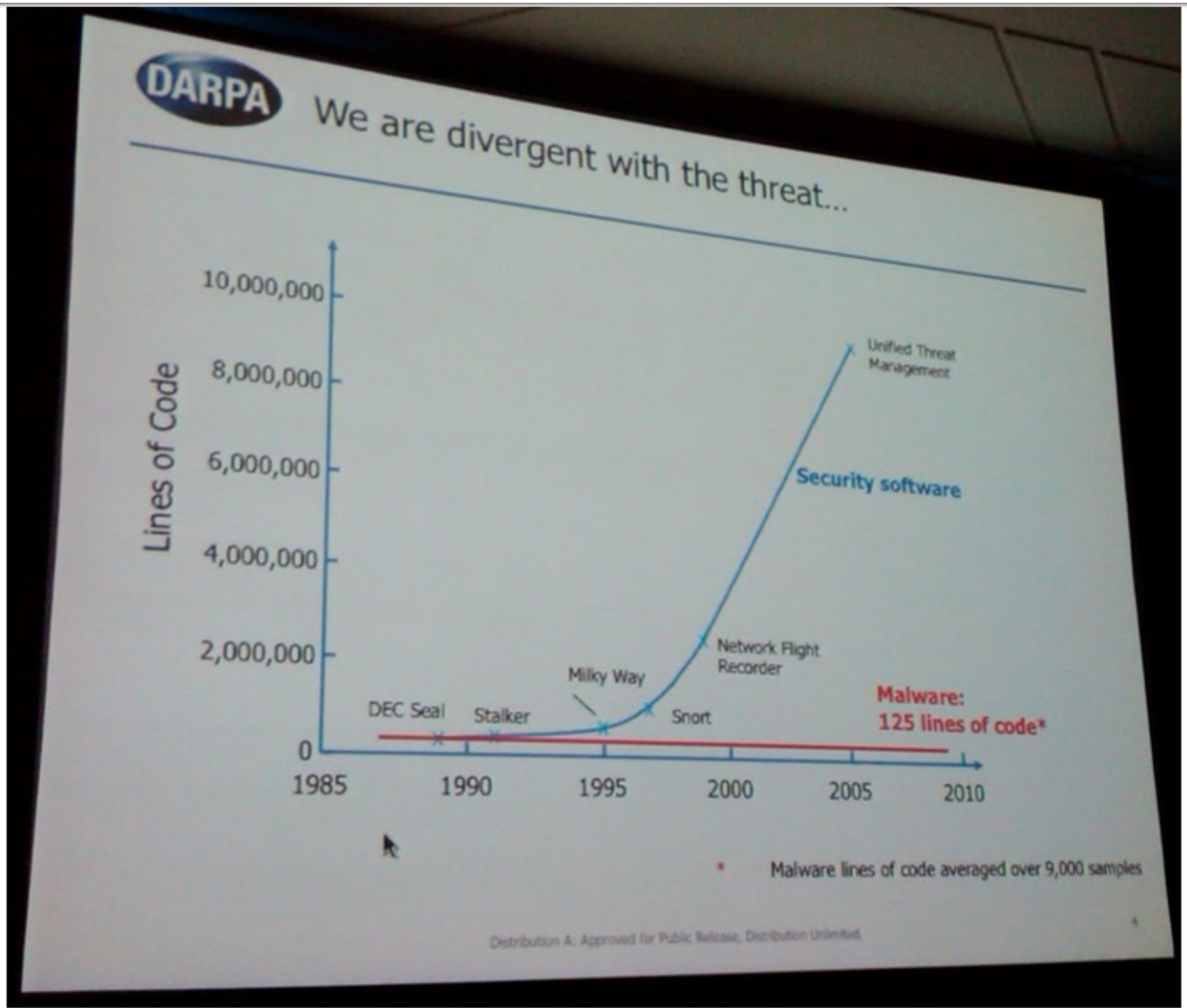
McAfee™

Norton™

Defense: Anti-Virus

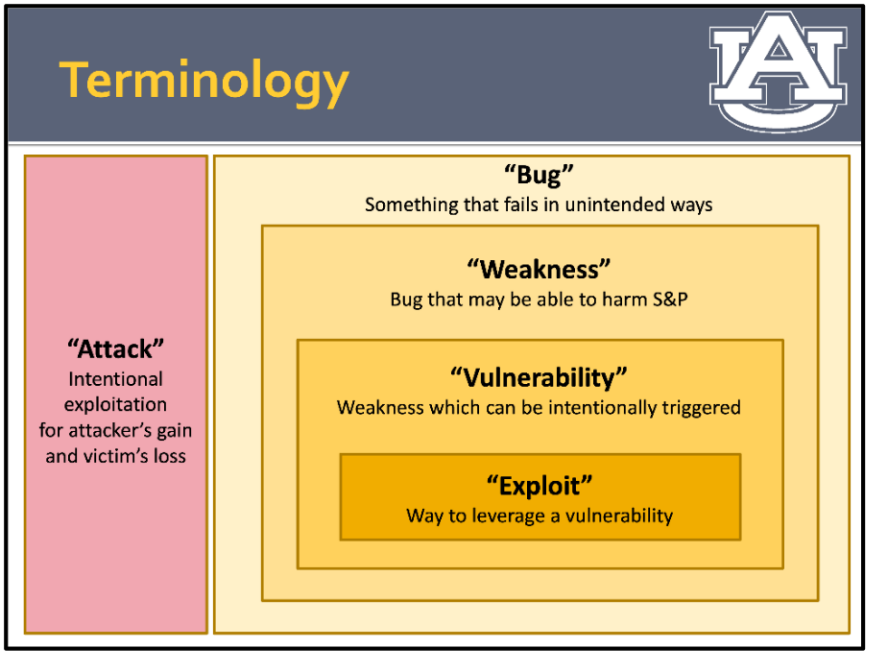
- Anti-Virus and “host-based defense systems” are … complicated…
- **Good** – They can quarantine and alert you when there’s known malware.
- **Maybe** – They can only tell you about known malware and the naïve versions
- **Bad** – They love to quarantine dev-tools and known-benign security tools/apps
- **Worse** – They have to touch **everything**

Defense: Anti-Virus

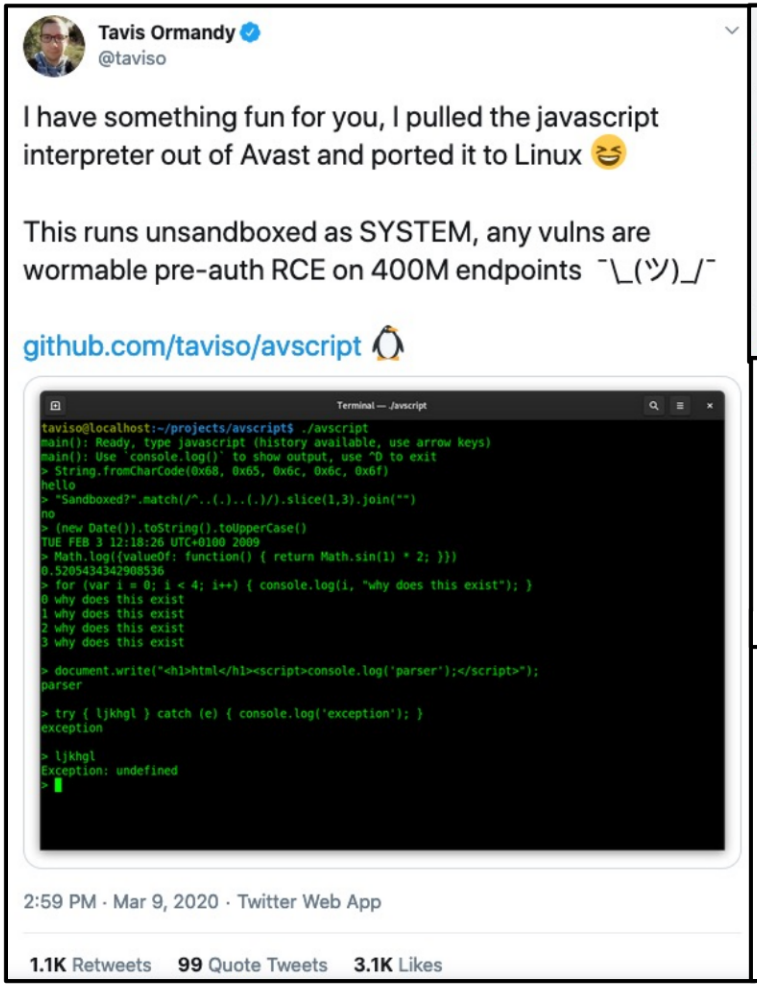


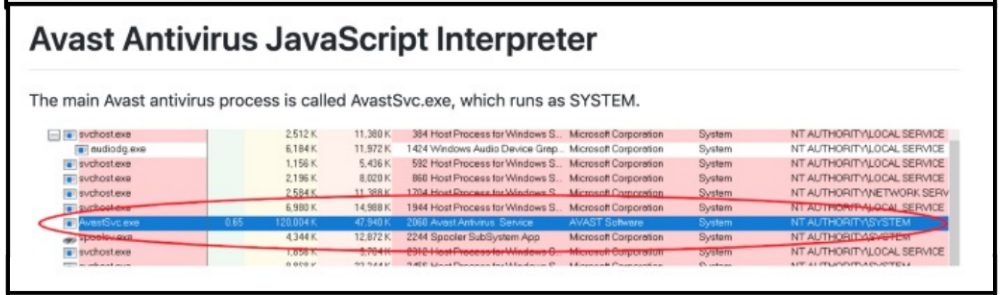
Software Facts of Life

- Software has bugs
- Some bugs are weaknesses
- Some weaknesses are vulnerabilities
- Some vulnerabilities can be exploited
- Someone has an interest in exploiting others for gain



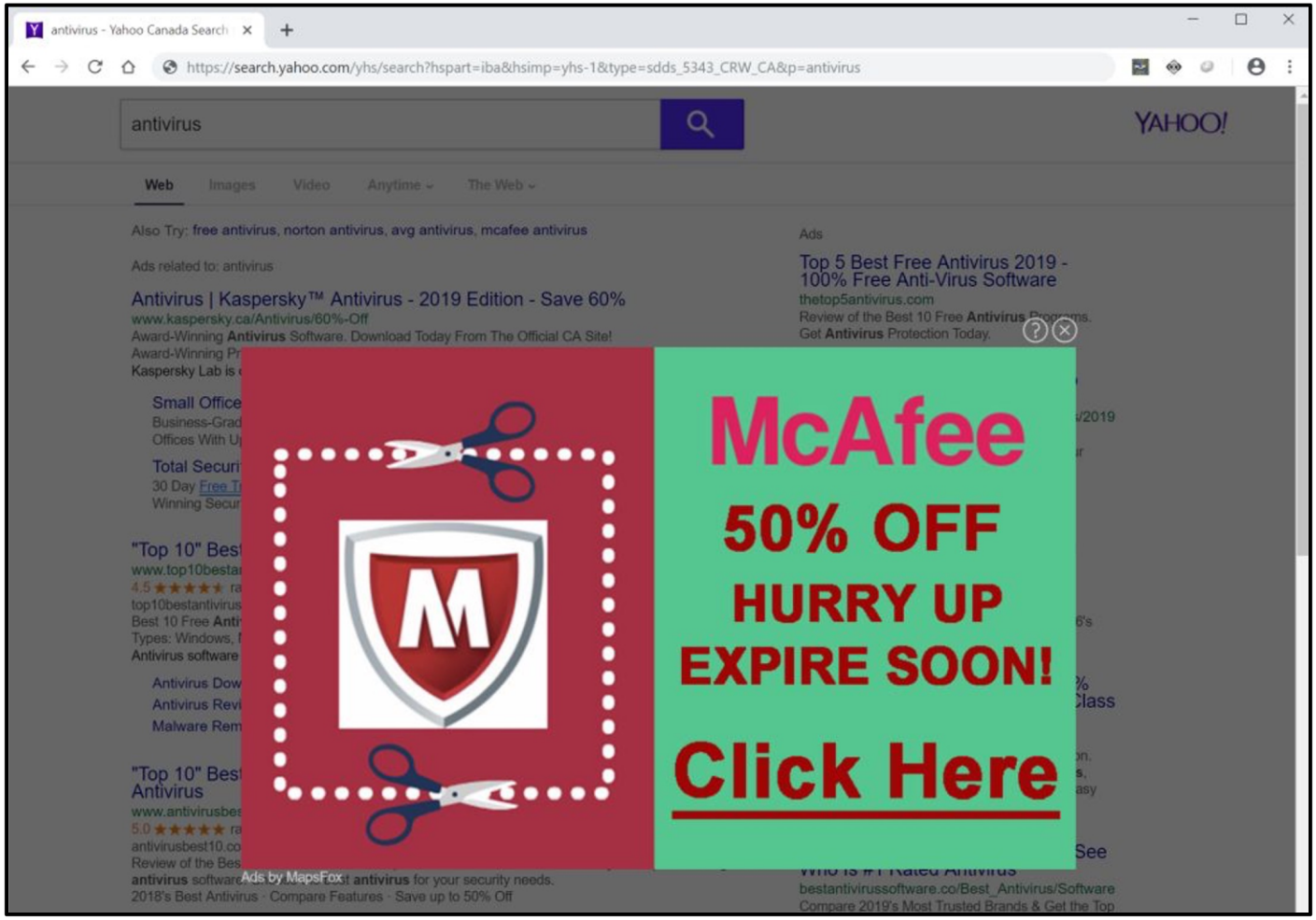
Defense: Anti-Virus





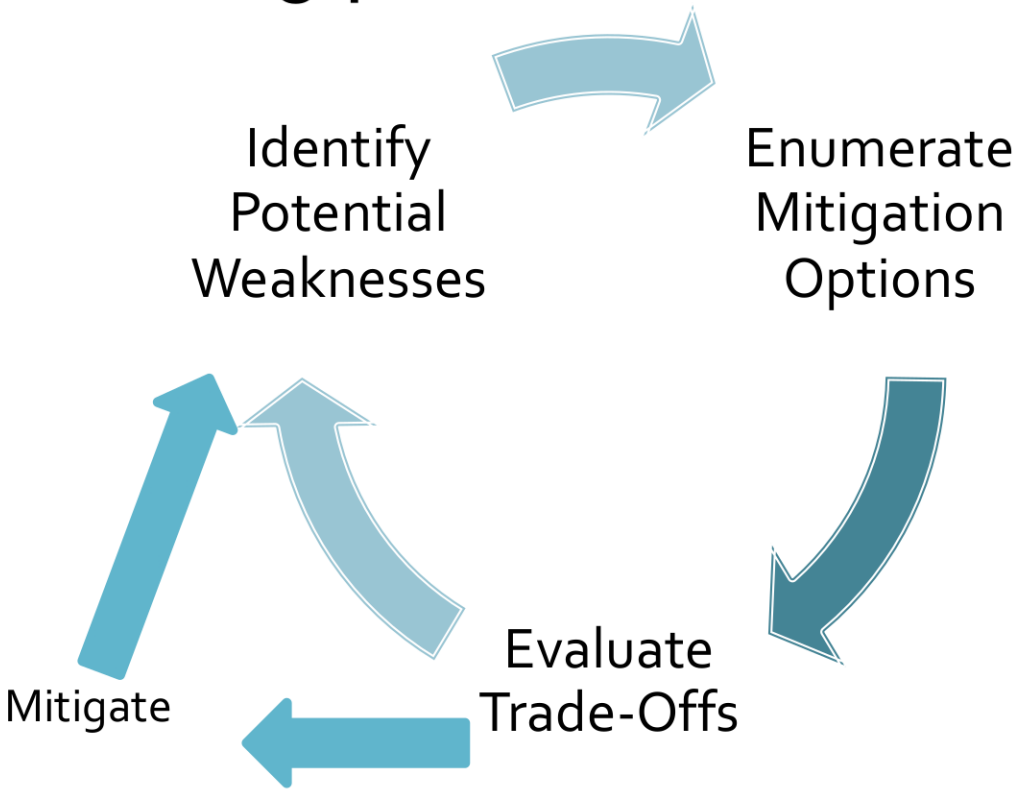
TLS Security	Increased Security	Same Security	Decreased Security	Severely Broken
Client Security Products	0/20	2/20	18/20	10/20
Middleboxes	0/12	1/12	6/12	5/12

Defense: Anti-Virus



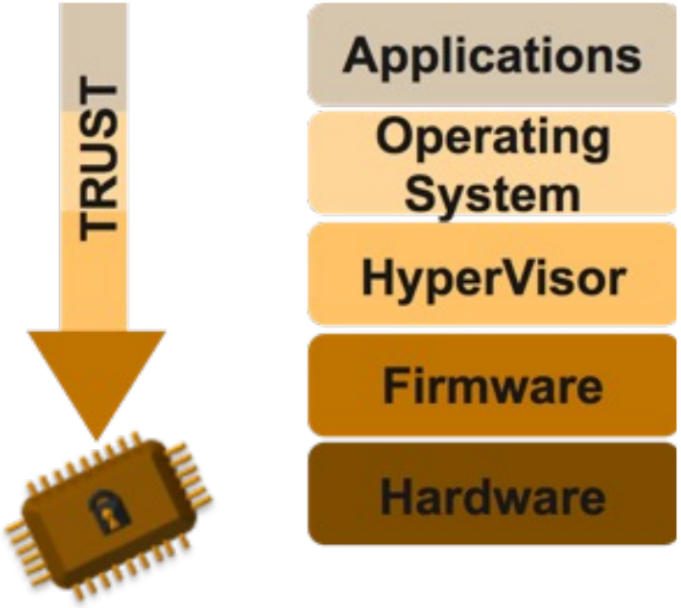
Threat Modeling

A systematic approach to analyzing and understanding potential weaknesses.



Trusted Computing Base

The **Trusted Computing Base (TCB)** is the collection of all components within a system critical to providing security properties.



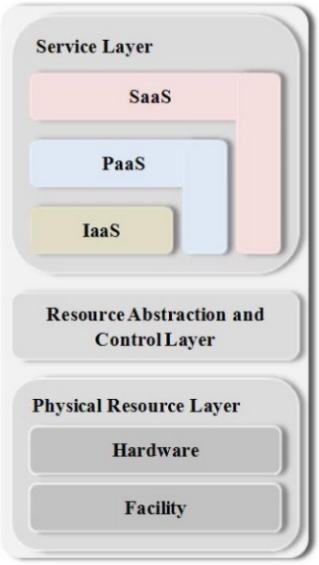
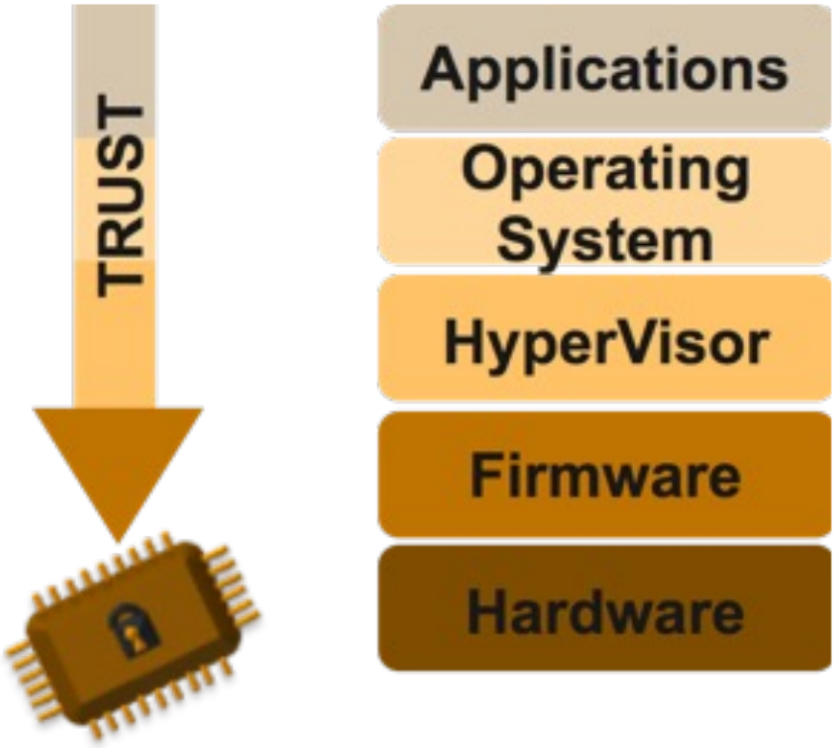


Figure 15: Cloud Provider - Service Orchestration

Local TCB



- Everything needed to run the application safely
- Each layer relies on the layers below it to behave correctly

Turtles All The Way Down

- Attack surface is exponentially larger b/c malicious lower-levels
- Level N bug means levels >=N are untrustworthy
- Bugs and vulns mimic each other due to abstraction

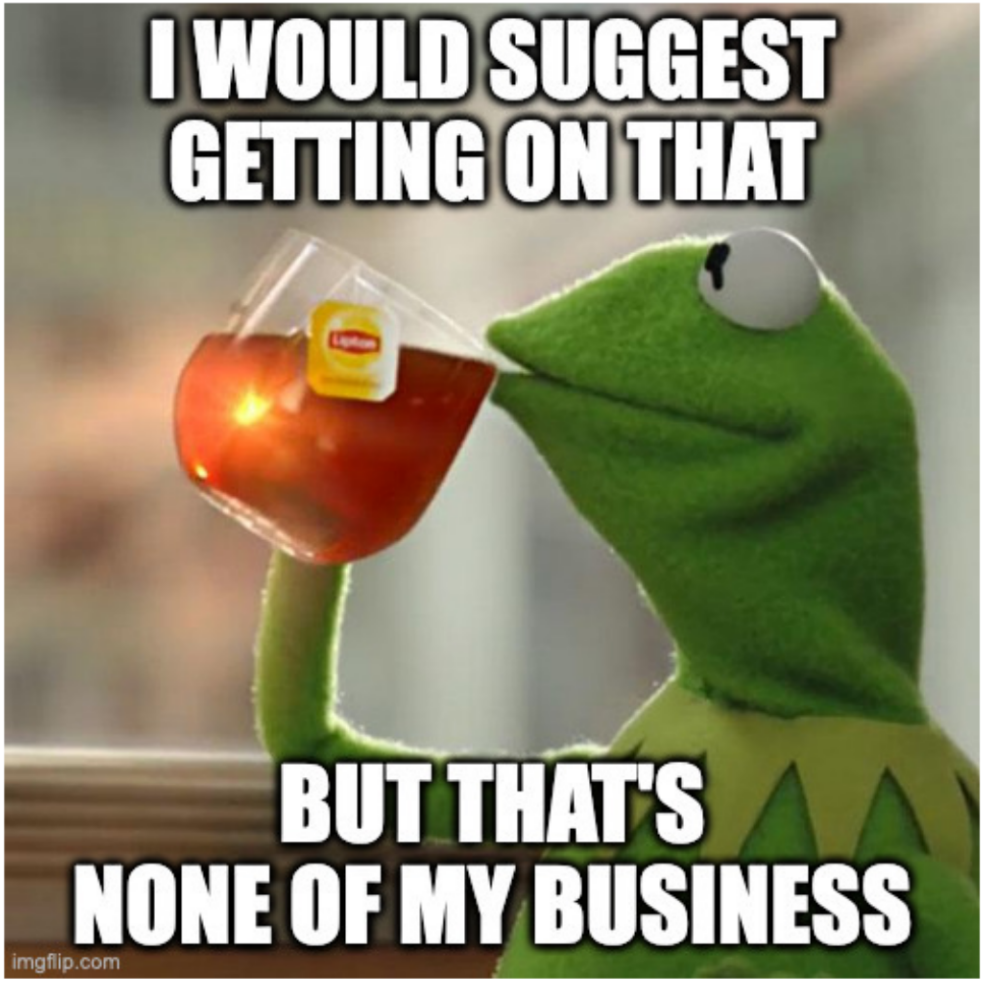
Computer and Network Security

Lecture 14:
OS Security & Isolation

COMP-5370/6370
Fall 2024

Course Notes

- Project 2 due **next Sunday** (13Oct2024)
  - If you haven't setup the VM…


I WOULD SUGGEST
GETTING ON THAT
BUT THAT'S
NONE OF MY BUSINESS
imgflip.com

Computer and Network Security

Lecture 14:
OS Security & Isolation

COMP-5370/6370
Fall 2024