COMP482
Cybersecurity
Week 2 - Friday

Dr. Nicholas Polanco

(he/him)

Attendance Trivia

Q: What is the smallest country in the world by land area?

Q: Which superhero is known as the “Caped Crusader”?

Q: Mount Everest is located in which mountain range?

Q: In the movie Inception, what object does Dom Cobb (Leonardo DiCaprio) use to tell if he’s in a dream?

Q: What’s the name of the theoretical point in a black hole where gravity is thought to be infinite?

CSE482 - Attendance Quiz (Programming)

Attendance Trivia

Q: What is the smallest country in the world by land area?
A: Vatican City

Q: Which superhero is known as the “Caped Crusader”?
A: Batman

Q: Mount Everest is located in which mountain range?
A: The Himalayas

Q: In the movie Inception, what object does Dom Cobb (Leonardo DiCaprio) use to tell if he’s in a dream?
A: A spinning top

Q: What’s the name of the theoretical point in a black hole where gravity is thought to be infinite?
A: Singularity

CSE482 - Attendance Quiz (Programming)

Important Notes

1. The project AND presentation deliverable are moved to **Friday of Week 3 (April 18th)**.
    a. If you are working in a group and sending 1 representative, I will need all those group members to email me so I have something in writing that you approve of this.
        - If you have already met with me (in my office), I have made note of this requirement :)

Important Dates (Week 3)

Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
				Reflection: Week 2 Project Deliverable: Meeting with Dr. Polanco Topic Deliverable: Topic Selection Activity: Keylogger or Buffer Overflow		

Outline

1. Zero-Day Vulnerabilities
2. Buffer Overflow
3. Keylogger / Ransomware
4. Race Conditions
5. Antivirus Software
6. Activity: Keylogger/Buffer Overflow

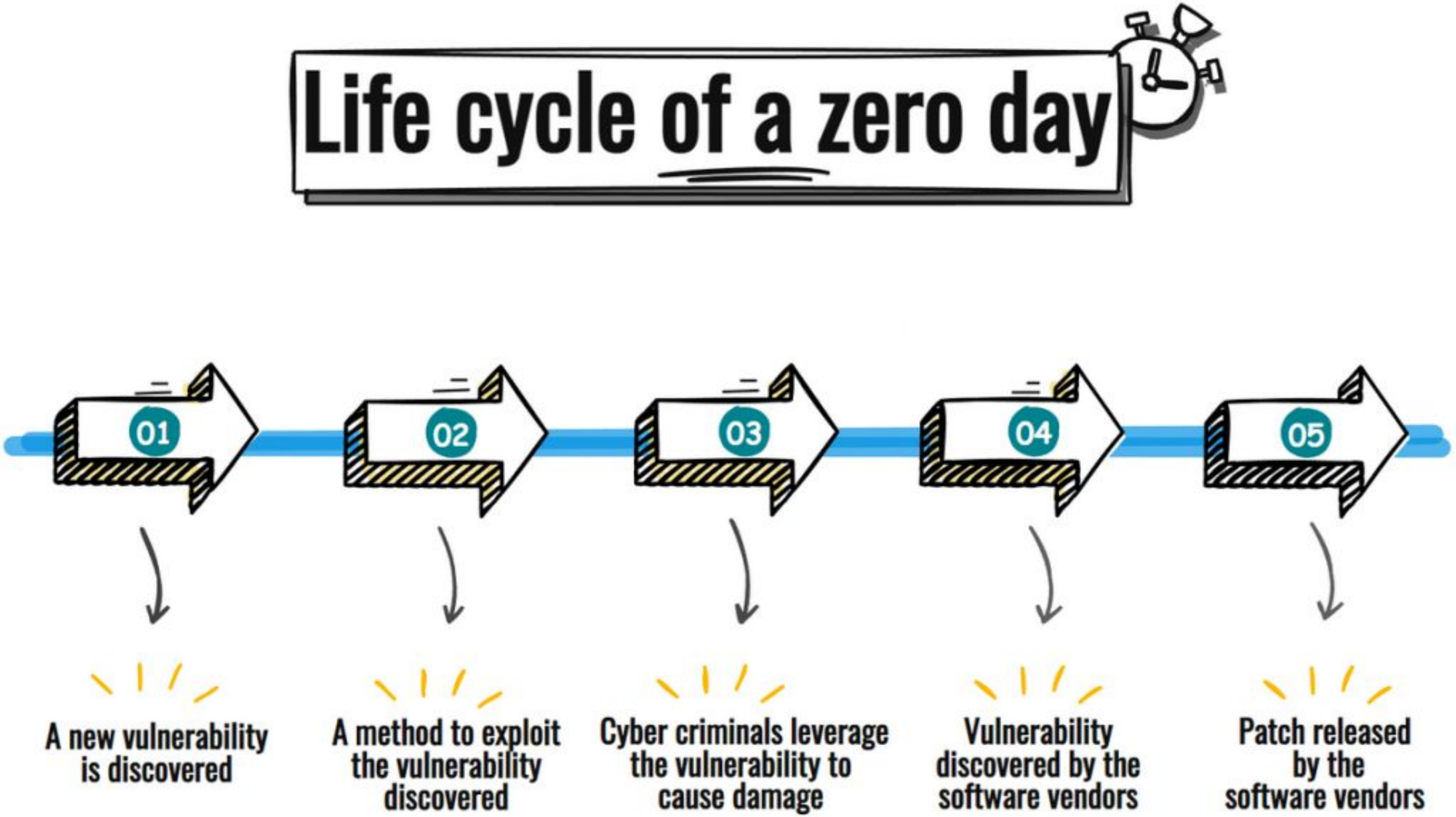
Zero-Day Vulnerability

Zero-Day Vulnerability

A zero-day vulnerability refers to a security flaw in software or hardware that is **unknown to the vendor or developer.** This means patch or fix available, making it especially dangerous.

The term "zero-day" comes from the fact that the developers have had zero days to fix the issue.

Image Credit



Life cycle of a zero day

01 A new vulnerability is discovered

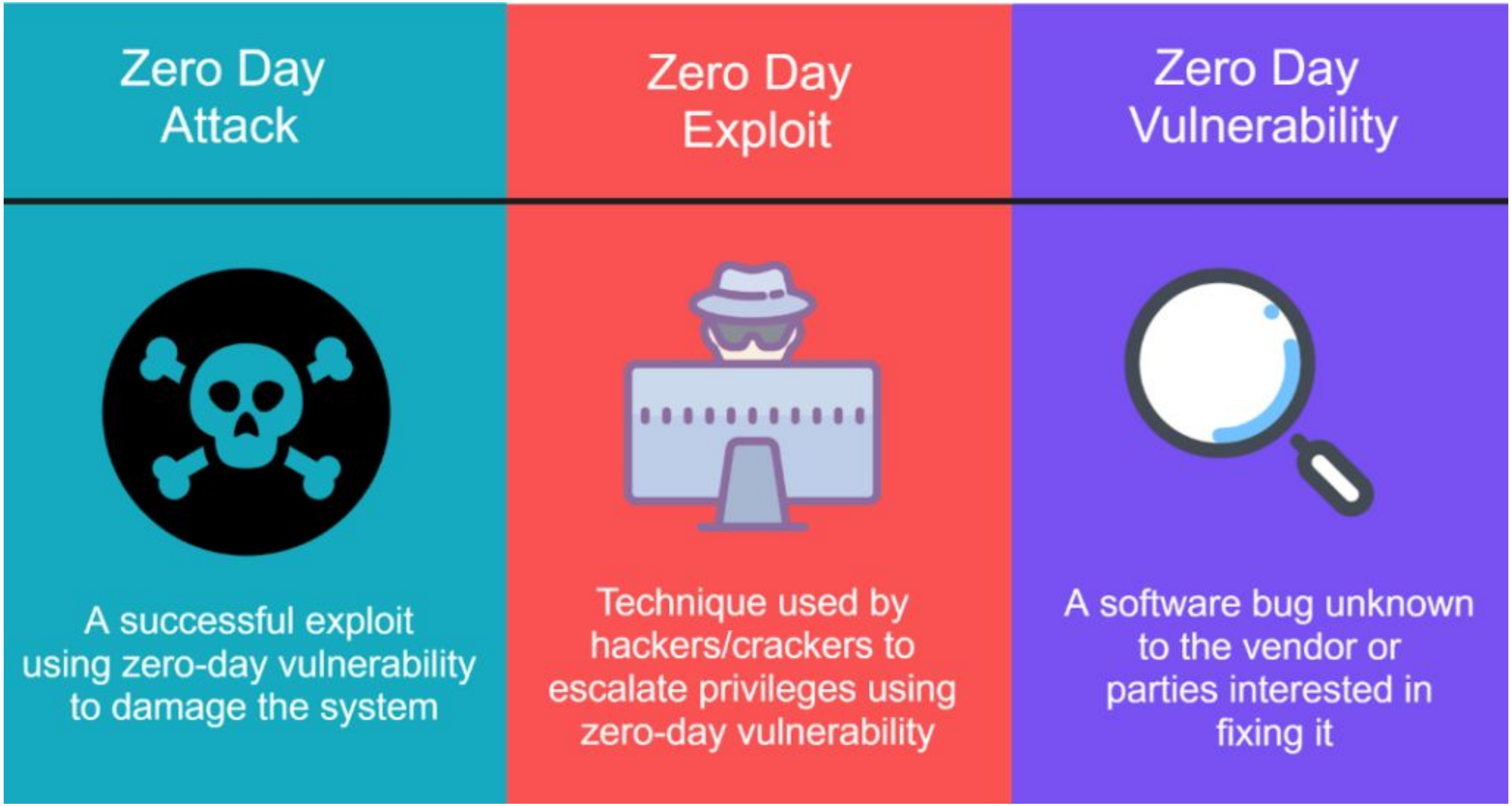
02 A method to exploit the vulnerability discovered

03 Cyber criminals leverage the vulnerability to cause damage

04 Vulnerability discovered by the software vendors

05 Patch released by the software vendors

Image Credit
https://www.balbix.com/insights/what-is-a-zero-day-exploit/



Zero Day Attack

A successful exploit using zero-day vulnerability to damage the system

Zero Day Exploit

Technique used by hackers/crackers to escalate privileges using zero-day vulnerability

Zero Day Vulnerability

A software bug unknown to the vendor or parties interested in fixing it

Image Credit
https://accuknox.com/blog/zero-day-attacks-what-why-how

Buffer Overflow

Buffer Overflow

A buffer overflow attack occurs when a program writes more data to a buffer than it can hold. This extra data can overflow into adjacent memory, potentially overwriting valid data, changing program behavior, or even executing malicious code.

Programs written in low-level languages like C or C++ often use fixed-size buffers to store data. If proper bounds-checking isn't enforced, an attacker can exploit this

Image Credit

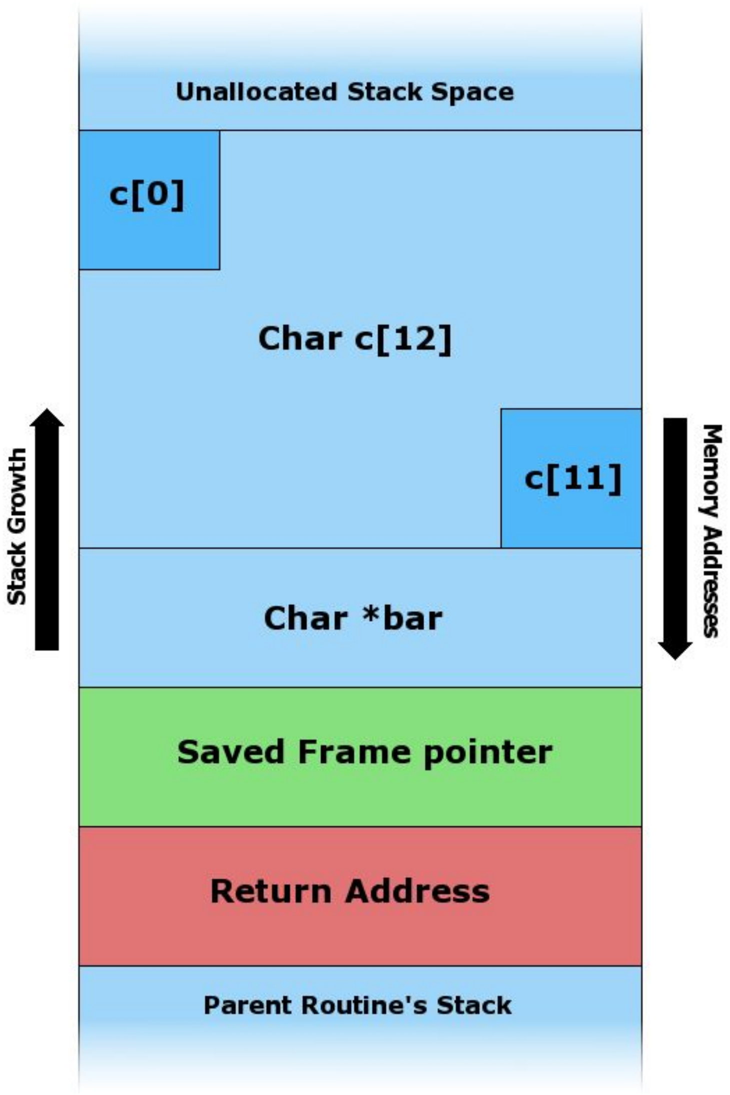


Image Credit
https://en.wikipedia.org/wiki/Stack_buffer_overflow#/media/File:Stack_Overflow_2.png

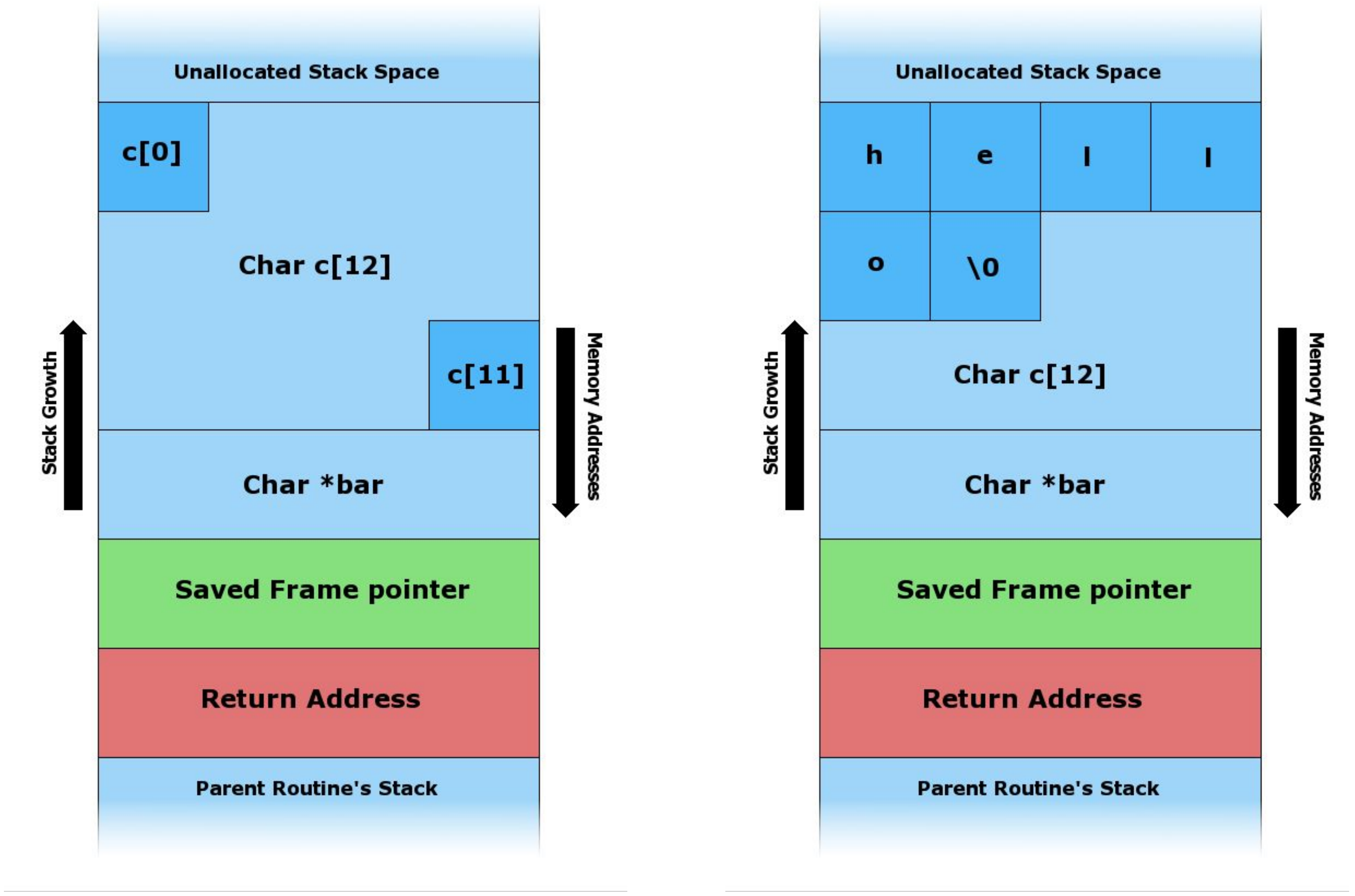
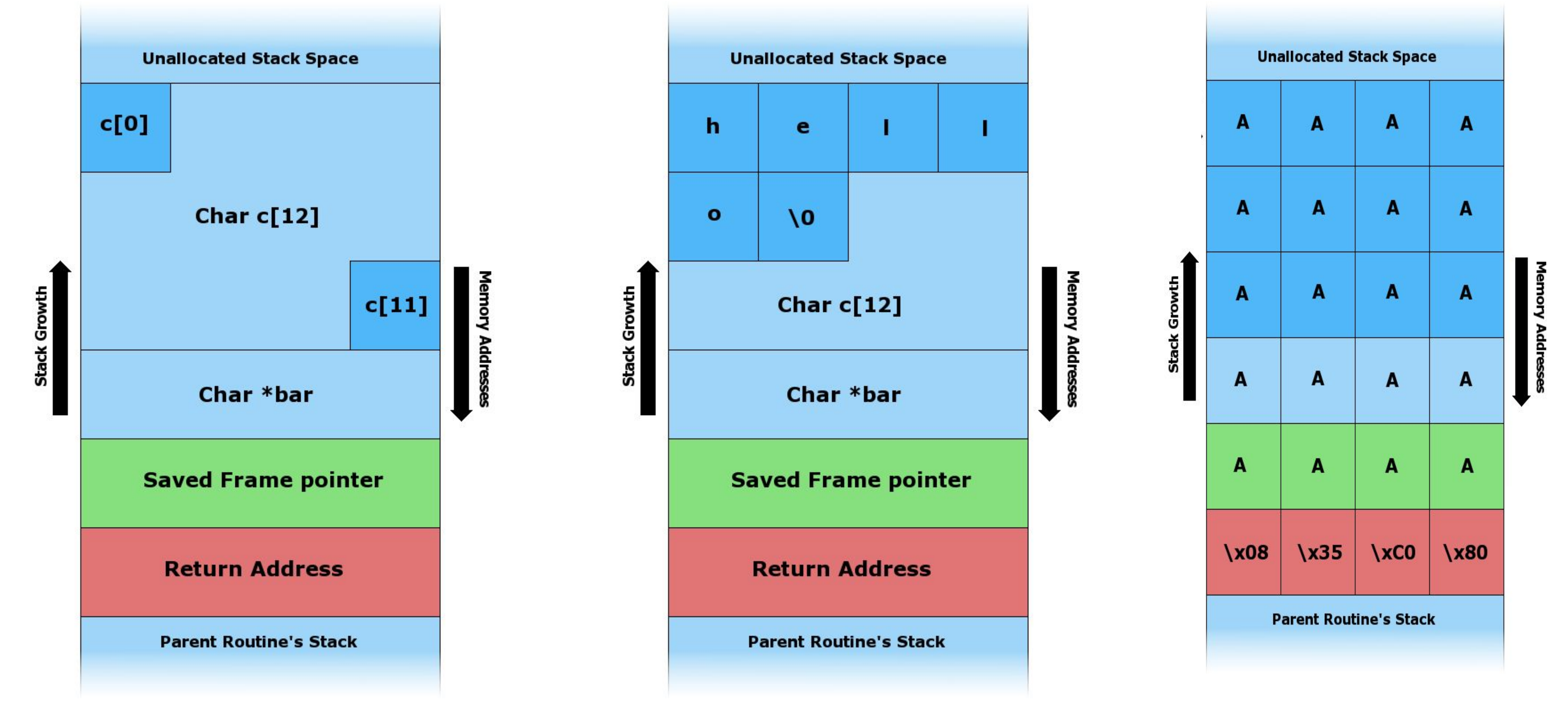


Image Credit
https://en.wikipedia.org/wiki/Stack_buffer_overflow#/media/File:Stack_Overflow_2.png



Input: AAAAAAAAAAAAAAAAAAAA\x08\x35\xC0\x80

Image Credit
https://en.wikipedia.org/wiki/Stack_buffer_overflow#/media/File:Stack_Overflow_2.png

A Simple C Example

```
void vulnerable_function() {
    char buffer[10];
    gets(buffer);  // DANGER: no bounds checking
}
```

Image Credit

Why Buffer Overflow Happens?

Lack of bounds checking

- Older or poorly written code (especially in C/C++) often doesn’t check if inputs fit in the memory allocated (e.g., using gets(), strcpy(), etc.).

Contiguous memory layout

- In a stack-based buffer overflow, local variables (buffers) are stored right next to the return address. Overflowing the buffer lets you overwrite that return address.

Predictability of memory layout (historically)

- Before modern protections, stack memory layout was predictable, making it easier for attackers to aim their payloads.

Pause: Code Injection Attack

A code injection attack occurs when a malicious actor inserts or injects unauthorized code into a program or system with the goal of modifying its behavior, accessing sensitive data, or gaining control. These attacks exploit poor input validation or insecure code execution mechanisms.

Image Credit

Keylogger / Ransomware

Keylogger

A keylogger is a type of surveillance technology or malicious software that records every keystroke made on a keyboard.

They *can* be used for legitimate purposes, such as monitoring employee activity or recovering lost data, but they are more commonly associated with malicious intent, particularly in cybersecurity threats like identity theft or credential harvesting.

Image Credit

Keylogger Types

Kernel-Based

- These are like “rootkit keyloggers” that operate at the OS kernel level; hard to detect, hooks into OS input handlers.

Form Grabbers

- These capture keystrokes submitted in web forms, even before SSL/TLS encryption.

Javascript-Based

- They are injected into web pages via XSS or browser extensions and use JS listeners like onKeyDown.

Clipboard Loggers

- They log clipboard content (e.g., when users copy-paste passwords).

Screen Scrapers

- The goal is to capture screenshots periodically or when specific keys are pressed.

Keylogger Defense

Two-Factor Authentication (2FA)

- This is a small additional to other defenses we have talked about, but adds an extra layer of security.
  - This ensures that even if a password is taken, another additional login tool is required.

On-Screen Keyboards:

- The users can make use of virtual keyboards for password entry.
  - How many of you use your virtual keyboards? How many people do you know that use this?

Image Credit

Ransomware

A ransomware attack is a type of cyberattack where threat actors use malware to encrypt a victim’s files or systems, rendering them inaccessible. The attackers then demand a ransom payment, usually in cryptocurrency (like Bitcoin), in exchange for a decryption key that may restore access to the locked data.

Image Credit

Ransomware Types

Crypto Ransomware

- These encrypts files (e.g., WannaCry) and make it so the user cannot access their files

Locker Ransomware

- This locks the user out of the entire system, rendering it unusable until paid.

Double Extortion

- The attackers steal data and threaten to leak it if the ransom isn't paid.

Ransomware Payment and Risk

These payments are often demanded in cryptocurrency to maintain attacker anonymity.

*There's no guarantee that paying the ransom will result in data recovery.

*This makes it so organizations that pay may become repeat targets

Image Credit

Ransomware Sample Scenario

A K-12 school district is attacked by ransomware, disrupting access to student records and online learning platforms. The district is considering whether to engage law enforcement or pay the ransom to expedite recovery. Additionally, the school system is underfunded and lacks sufficient cybersecurity resources to fully protect its network, raising questions about the responsibility of educational institutions in cybersecurity preparedness.

Discussion Focus:

- What legal and ethical considerations should the school district take into account when deciding whether to pay the ransom or seek law enforcement involvement?
- How can educational institutions address the challenges of limited budgets while implementing effective cybersecurity measures to prevent such attacks in the future?

Image Credit

Race Conditions

Race Conditions

A race condition in cybersecurity refers to a situation that occurs in a computer system or application when multiple processes or threads access shared resources (like memory, files, or databases) concurrently, and the final outcome depends on the order in which the processes or threads are executed.

What are products/companies/websites that may run into situations where they need to be careful about race conditions and making sure "shared resources" are safe?

Image Credit

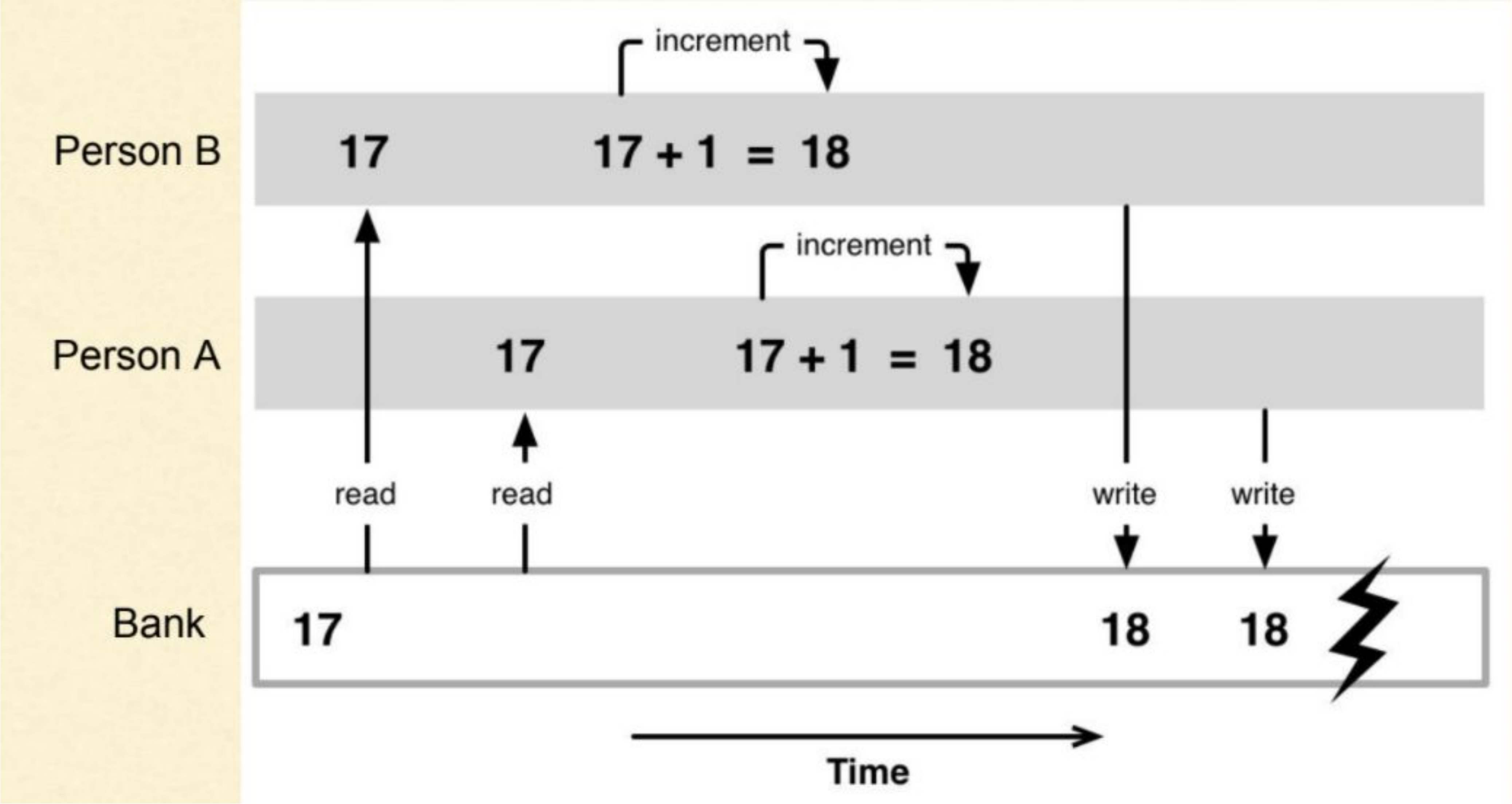


Image Credit
https://cloudxlab.com/blog/race-condition-and-deadlock/

Race Conditions (continued)

The term "race" comes from the idea that the processes are "racing" to access and manipulate the shared resource, and if one process gets there first, it can affect the behavior of the others.

Race conditions are often bugs or vulnerabilities in software that can lead to unpredictable or insecure behavior, and they occur when there is insufficient synchronization or control over how resources are accessed and modified by multiple processes.

Race Conditions Dangers

Data Corruption

- If multiple processes write to a file or database without proper coordination, it could result in inconsistent or corrupted data.
  - What aspect of the CIA triad would be in danger here?

Privilege Escalation

- An attacker may be able to exploit a race condition to gain higher privileges or bypass authentication.

Image Credit

Race Conditions Prevention

What can we do to our shared resources to make sure no issues occur?

Image Credit

Race Conditions Prevention

What can we do to our shared resources to make sure no issues occur?

Locks

- We make sure only one thread or process can access the resource at a time.

Atomic Operations

- An operation is completed entirely before another one can begin.

Transaction Integrity

- Using transactions that guarantee a series of operations happen completely or not at all, ensuring consistency.

Image Credit

Antivirus Software

Antivirus Software

Antivirus software is designed to detect, prevent, and remove malware from computer systems. It provides essential protection by identifying harmful files and activities that could compromise system integrity, data, and privacy.

In modern systems, antivirus solutions are part of a broader endpoint protection strategy, which includes additional layers like firewalls and intrusion detection systems.

Pause: Endpoint Protection Strategies

Endpoint protection strategies are designed to safeguard devices (endpoints) such as computers, smartphones, tablets, and other networked devices from cyber threats and attacks.

These strategies focus on preventing, detecting, and responding to various types of security risks that might target endpoints, such as malware, ransomware, phishing attacks, and unauthorized access.

Image Credit

KASPERSKY
ANTI•VIRUS
N
AVG
McAfee
bitdefender
secure your every bit
F-Secure
ca
INTERNET
SECURITY
SUITE
2010
Microsoft
Security Essentials
avast!
be free
AVIRA
eset
NOD32
antivirus

Image Credit
https://www.wepc.com/deals/where-to-buy-antivirus-software/

How Antivirus Software Works

1. Installation - The user installs the software on a device (e.g., computer, tablet, phone).

2. Scanning and Detection - The antivirus software will then scan and try to find malicious software (often upon installation). It can use a few strategies to find malicious activity.
   a. Signature-Based Detection: As files are accessed, the antivirus scans them for known malicious signatures. This is the most straightforward form of detection.
   b. Heuristic Detection: This applies if an occurrence doesn't match a known signature but behaves suspiciously (e.g., attempts to modify system settings or send out mass emails), the antivirus may flag it as a potential threat.
   c. Behavioral Analysis: This is for malware that attempts to execute a harmful action (such as encrypting files in a ransomware attack), the antivirus can stop the process in real time.

How Antivirus Software Works (continued)

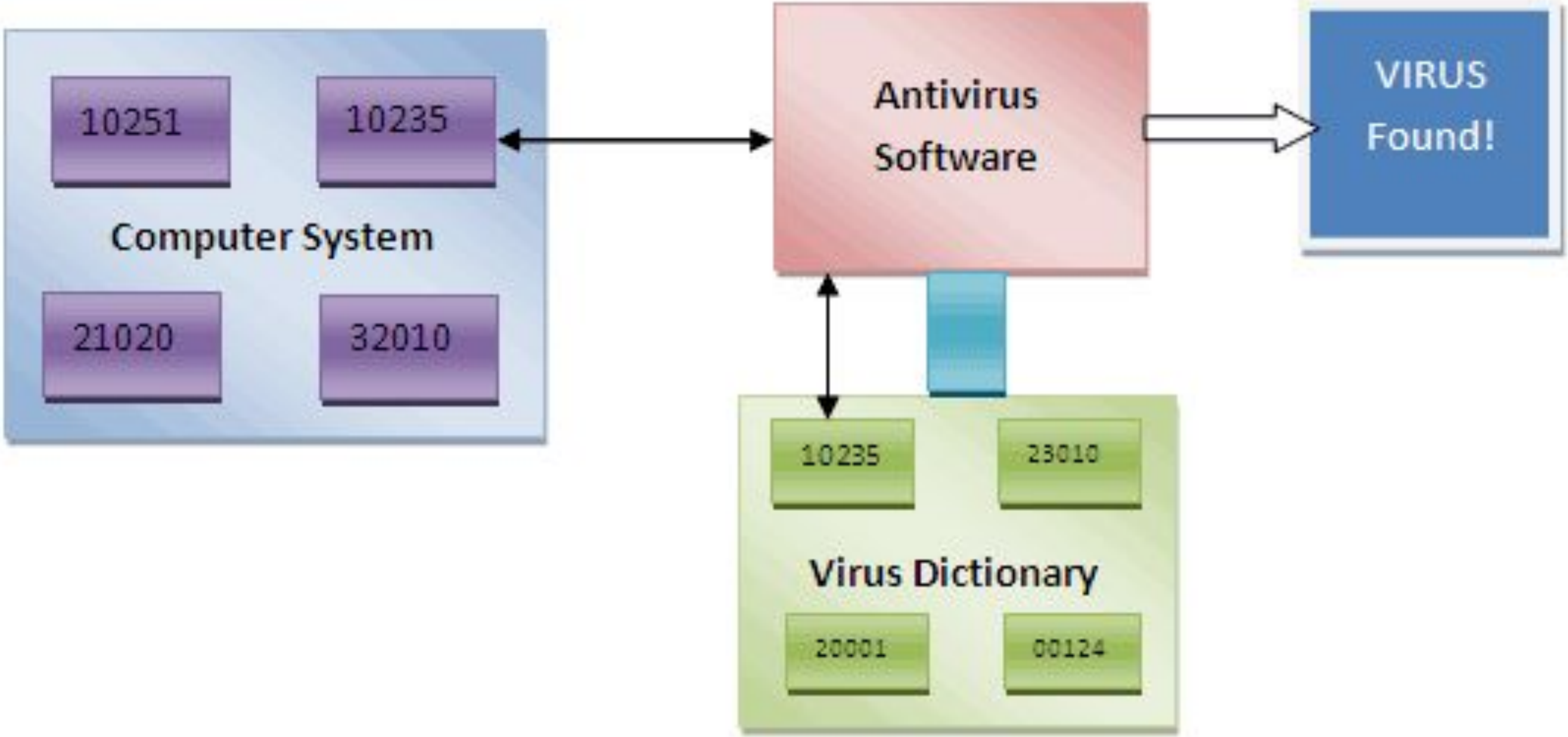
3. Removal or Quarantine - Once a threat is detected, the antivirus software can either:
   a. Quarantine: The suspicious file is isolated so it cannot do harm. This is a precautionary measure in case the file turns out to be a false positive.
   b. Delete: The file is removed from the system entirely.
   c. Repair: In some cases, the software can repair infected files, removing the malicious code while leaving the original file intact.

4. Continuous Updates - As we learned, **new malware is created daily**, antivirus software must continuously update its signature database to remain effective. This is often done automatically via updates from the software vendor, keeping the system protected against the latest threats.

Pause: Signatures

A signature in antivirus terminology is a unique identifier or pattern derived from a known piece of malware. This can include:

- A sequence of bytes from the binary (called a byte pattern).
- Cryptographic hashes (e.g., MD5, SHA-1, SHA-256) of the file or parts of it.
- Specific strings or code fragments (e.g., API call sequences, function names, or suspicious strings).



Signature based Antivirus

Image Credit
https://www.engineersgarage.com/how-antivirus-works/

KALAMAZOO COLLEGE K

Additional Antivirus Features

These can also have:

Cloud-Based Detection: Many use cloud computing to enhance detection. When a new file is found, the antivirus software can send it to a remote server where it's analyzed against the latest database of known threats. This allows faster updates and detection of emerging threats.

Sandboxing: Some antivirus solutions isolate suspicious files in a sandbox environment where they can be safely executed and analyzed. If the file proves to be malicious, it can be removed without risk to the system.

Antivirus Discussion

How much of the effectiveness of antivirus software depends on user behavior? Should antivirus software take a more active role in educating users or enforcing better security practices?

Many users complain that antivirus software slows down their computers. In your opinion, should users prioritize performance or security?

Should antivirus software vendors be held accountable for failing to detect certain types of malware? If so, to what extent?

Activity: Keylogger/Buffer Overflow

Notes for Activity:

- You are to choose **one** of these to implement, you can do both if you decide to do so and I will take the higher of the two grades.
- I will try to be as “hands-off” as possible to allow you to approach this problem in a number of different ways. You are welcome to work in groups, but I want to see the creative ways that you tackle this.
- You will submit a text version of your code, screenshots of your work functioning, and proof of a successful output. I would do this in a Word/Google Doc. This needs to be submitted via a PDF!
  - Do you understand that if you submit a program/script, you are attacking me?!

Activity: Keylogger/Buffer Overflow

Notes for Keylogger:

- **You need to execute the keylogger in a sandboxed mode or a VM to avoid infecting your actual systems.**
- You need to have a “failsafe button”to stop the keylogger.

Image Credit

Activity: Keylogger/Buffer Overflow

Notes for Buffer Overflow:

- You need to show the various outputs in a few different ways.
- You should *try* to get your Buffer Overflow to execute another program.
    - I would suggest a simple “Hello World!”

Questions?