



# Cyber Security

## Lecture 1.1 Overview of the Course

# Lecture Agenda



- **Course Information and Protocol**
- **Cyber Security: A Bigger Picture**
- **Categories of Cyber Security**
- **History of Cyber Attacks**
- **Cybersecurity Roles**
- **The New Battle Field: AI Meets Cybersecurity**

# Course Info & Protocols



# About The Instructor



Dr. Muhammad Arif Butt  
Asst. Prof. Department of Data Science  
University of Punjab, Lahore

[arif@pucit.edu.pk](mailto:arif@pucit.edu.pk)  
[https://arif.excaliat@gmail.com](mailto:https://arif.excaliat@gmail.com)  
<https://www.linkedin.com/in/dr-arif-butt/>  
<https://youtube.com/learnwitharif>  
<https://github.com/arifpucit>  
<https://huggingface.co/arif-butt>

- **Education:**

- Graduation from Pakistan Military Academy, Kakul
- MPhil CS from University of Punjab, Lahore
- PhD CS from University of Punjab, Lahore

- **Experience:**

- Served in field/staff/instructional posts in Pakistan Army
- Assistant Professor, Department of Data Science
- Founder Excaliat (<https://excaliat.com/en>)
- Founder FalconHunt (<https://falconhunt.org/>)
- Co-Founder Tbox Solutionz (<https://tboxsolutionz.com/>)

- **Research Interest:**

- Embedded and Real-Time Operating Systems
- Vulnerability Analysis, Binary Exploitation & Exploit Development
- AI-Driven Cybersecurity and Securing AI systems



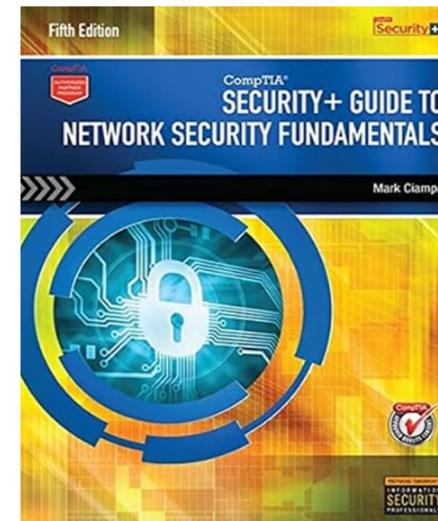
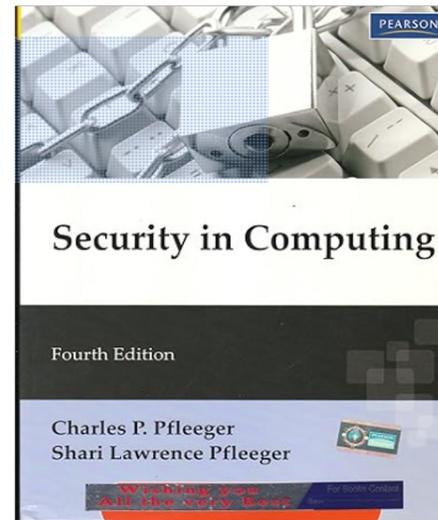
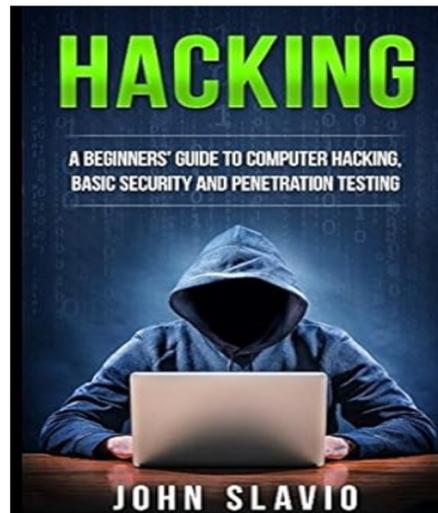
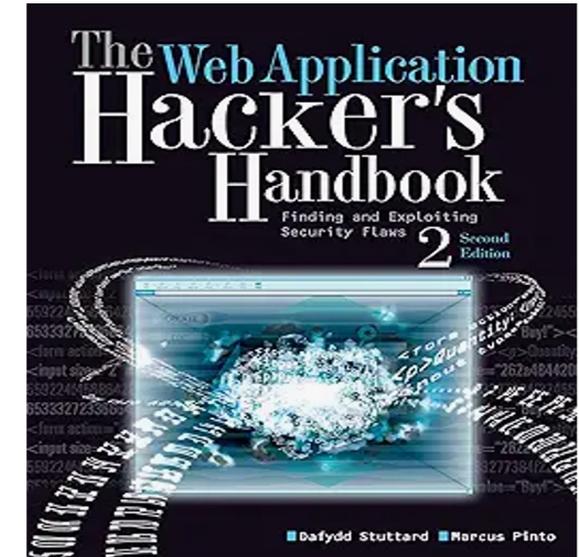
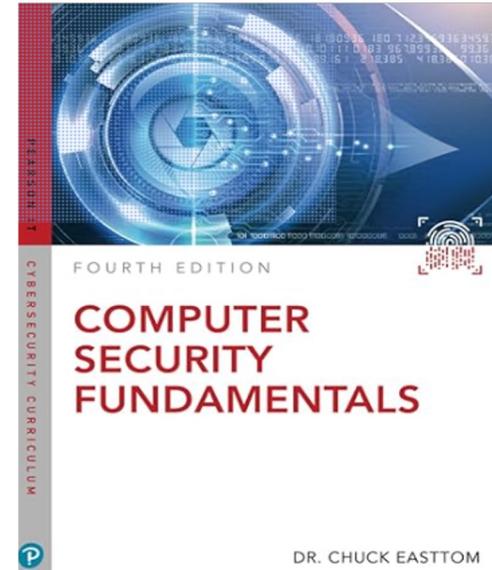
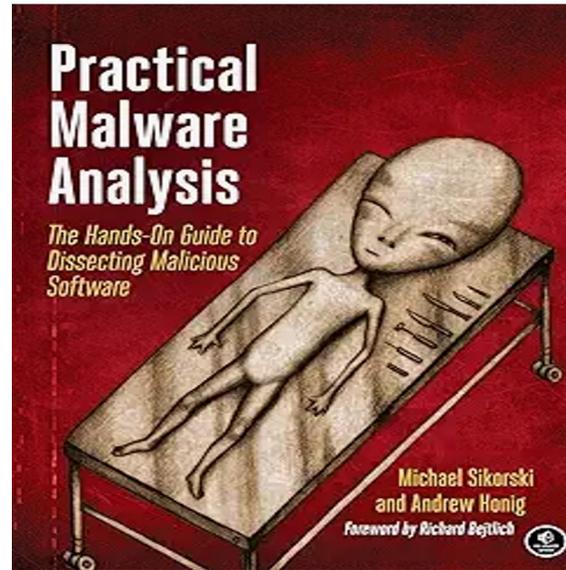
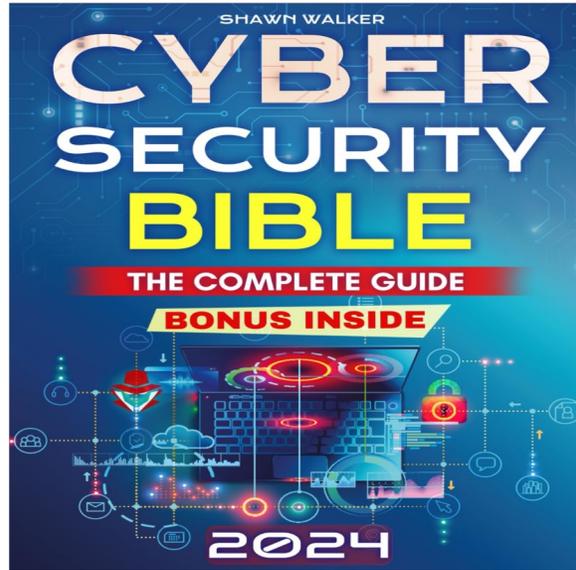
# Course Information



- Lectures Slides/Handouts Available at: <https://arifbutt.me>
- Video Lectures Available at: <https://youtube.com/learnwitharif>
- Codes Hosted at: <https://github.com/arifpucit>
- Grades Website: <https://online.pucit.edu.pk>
- Prerequisites:
  - OS and Internetworking with Linux
  - Basic programming skills in Python, C, and Assembly
- Office: Building-C, FCIT (NC)
- Students Counseling hours:
  - Monday: 0900 – 1000, and 1201 - 1300
  - Wednesday: 0900 – 1000, and 1201 - 1300
- 24 hour turnaround for email: [arif@pucit.edu.pk](mailto:arif@pucit.edu.pk)



# Text and Reference Books





# Who cares to get an A



- **Final-Term Exam:** 40
- **Mid-Term Exam:** 35
- **Sessional Activities:** 25
  - **Assignments: 30%**
  - **Quizzes: 40%**
  - **Class Activities: 30%**





# Lecture Format



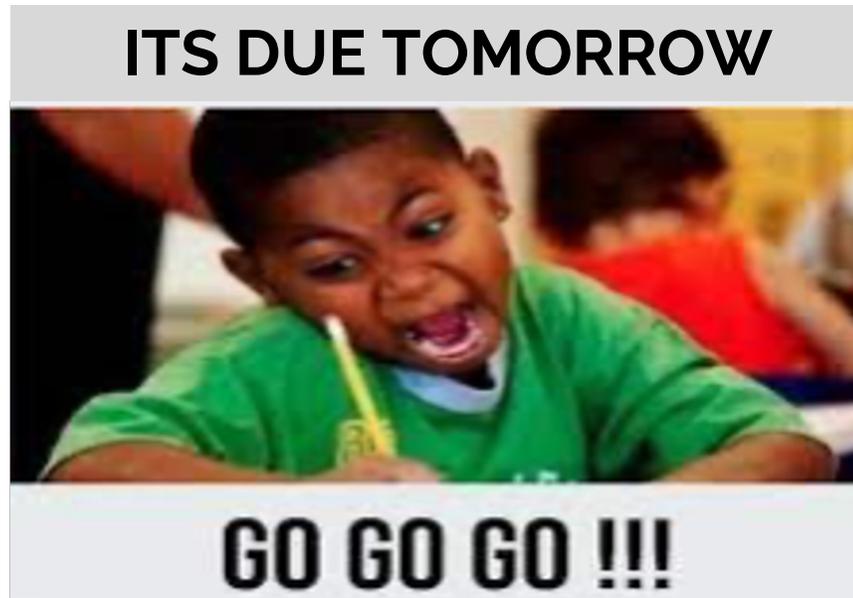
Instructor: Muhammad Arif Butt, PhD



# Late submission guidelines protocol



- Late Assignment submissions will not be accepted!
- There will be no retake on exams and quizzes!



Start working on your tasks early and submit well before time.



# Cheating Policy



- Academic integrity
- Copying an Assignment with or without permission. Both the cheater and the student who aided the cheater will be held responsible for the cheating
- The instructor may take actions such as:
  - require repetition of the subject work,
  - assign '**zero**' or may be '**negative**' marks for the subject work,
  - for serious offenses, assign an **F** grade for the course



# Cyber Security

## The Big Picture

# What is Cyber Security?



- **Cybersecurity** is the practice of protecting systems, networks, devices and data from from cyber threats such as malware, ransomware, phishing, data breaches and so on to safeguard both individual and organizational digital assets.
- It encompasses a wide range of domains, including information security, network security, application security, and incident response.
- It involves the application of technologies, processes, and practices to ensure the **confidentiality**, **integrity**, and **availability** of information.



# CIA Triad



**CIA Triad** is the foundational model in IS, representing three core principles that ensure the protection of information



**Integrity** ensures the accuracy, consistency and trustworthiness of information by protecting it from unauthorized modification, deletion or corruption

**Availability** ensures that information and resources are accessible to authorized users when needed

**Confidentiality** ensures that sensitive information is accessible only to authorized users.

## Opposite of CIA is DAD

- **Disclosure** means someone not authorized is getting access to the system.
- **Alteration** means your data has been altered.
- **Destruction** means your data or system have been destroyed.

**100\$ Question:** Finding the right mix

- Ensuring too much **C**, **A** will suffer.
- Ensuring too much **I**, **A** will suffer.
- Ensuring too much **A**, both **C** & **I** will suffer

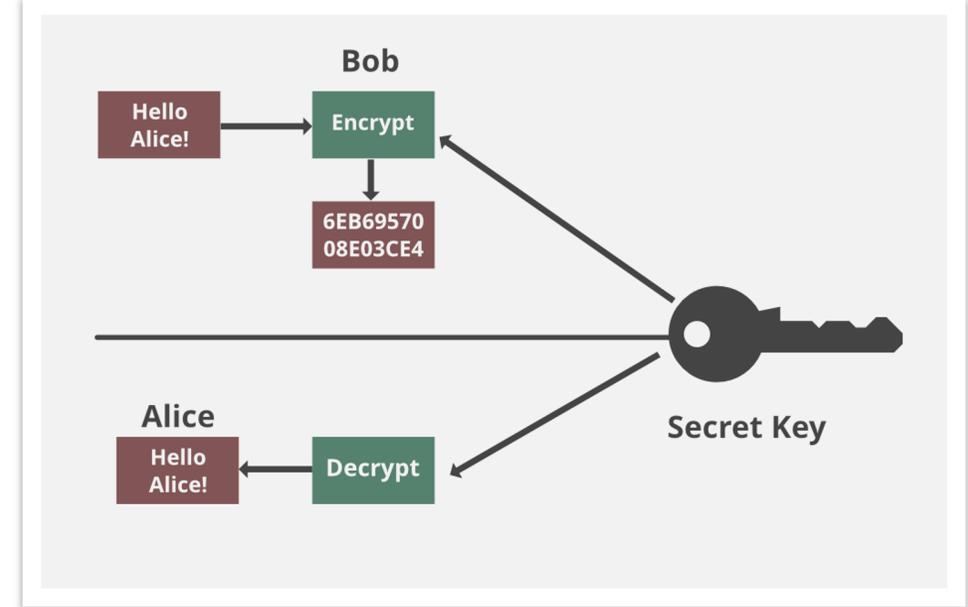
# CIA Triad: Confidentiality



**Confidentiality** ensures that sensitive information is accessible only to authorized users.

## Measures to achieve Confidentiality:

- **Encryption:** (3DES, Blowfish, AES, RSA, DSS, PGP)
- **Secure Transport Protocols:** (SSL, TLS, IPSec)
- **Access Control Mechanisms:** (DAC, MAC, RBAC)
- **Authentication Mechanisms:** (MFA, Biometrics)
- **NW Security Controls:** (Fire Walls, VPNs, IDS/IPS)
- **Least Privilege Principle**
- **Physical Security**
- **End-user Training**



## Threats to Confidentiality:

- Social engineering/Phishing.
- Unauthorized NW access & Port scanning.
- Eavesdropping and MitM attacks.
- Password dump stealing and attack on your encryption (cryptoanalysis).
- Authorized users may abuse their access to retrieve sensitive data.

# CIA Triad: Integrity

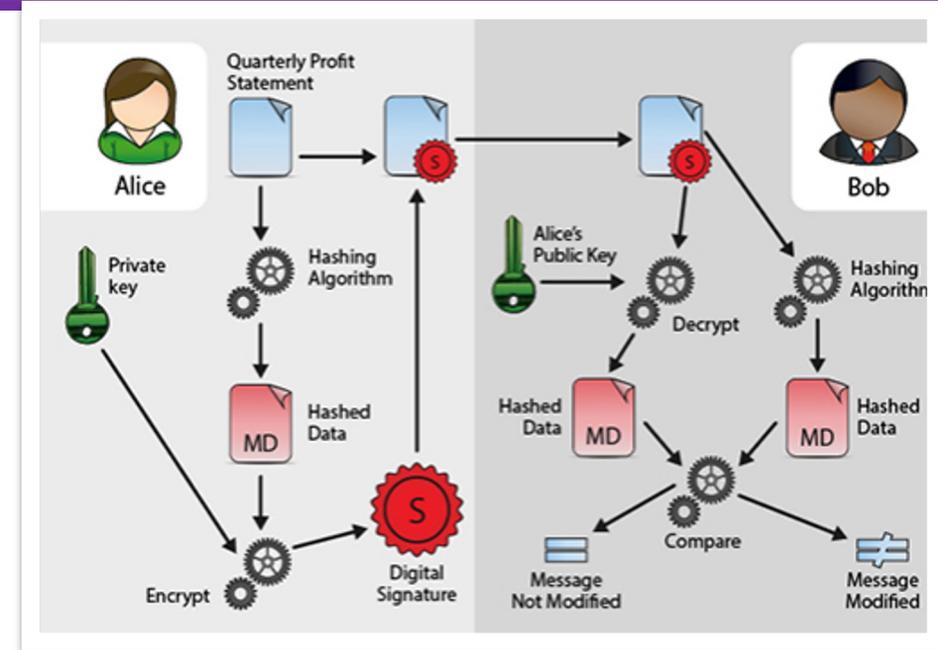


**Integrity:** ensures the accuracy, consistency and trustworthiness of information by protecting it from unauthorized modification, deletion or corruption

## Measures to achieve Integrity:

- **Hashing**, generating a fixed size hash value for data , so that any alteration is easily detectable. (MD5, SHA-256, SHA-512)
- **Checksums**, using checksums to detect errors in data communication or storage. (CRC-32, Adler32)
- **Digital Signature**, is used to verify the authenticity and integrity of a message or document. (RSA, PGP)
- **Version Control**, is used to track changes to document or code, allowing roll back if unauthorized changes are detected. (Git, SVN)
- **Active Logging**, maintaining logs that track data changes, system access and transactions (Splunk, Elastic Stack)

Instructor: Muhammad Arif Butt, PhD



## Threats to Integrity:

- MitM for tempering
- Data corruption by malware
- Malicious code injection
- Ransomware
- Deleting/altering DB records by SQLi
- DNS Spoofing / Cache Poisoning
- Replay attacks

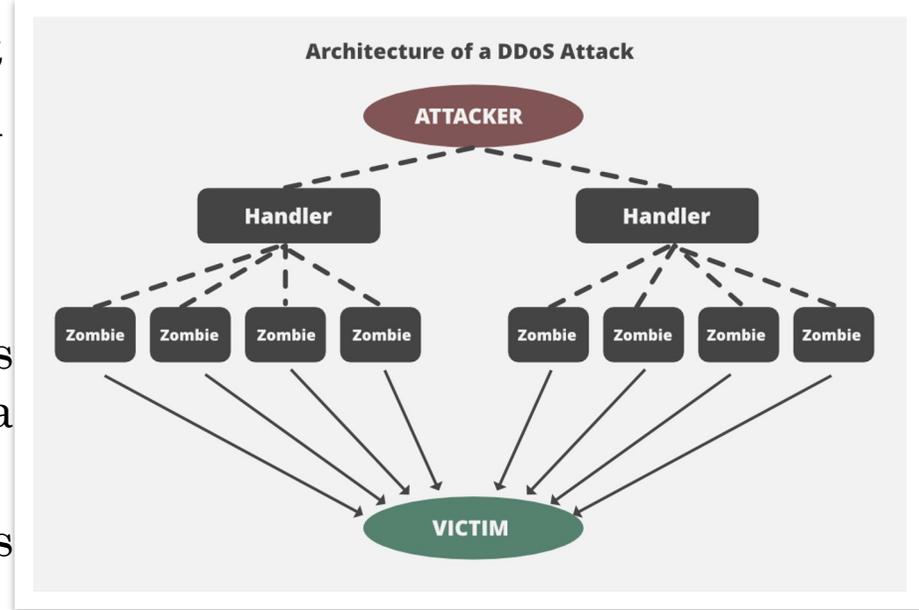
# CIA Triad: Availability



**Availability:** Achieving Availability ensures that information and resources are accessible to authorized users when needed.

## Measures to achieve Availability:

- **Redundancy and Failover**, Implementing redundant systems and automatic failover mechanisms (RAID, Load balancers, Data Center Failover)
- **DDoS Protection**, using security measures to ensure continuous availability of services. (Cloudflare, Akamai)
- **Backup and Recovery**, Regularly backing up data and maintaining recovery procedures to restore systems in case of a disaster. (Veeam, Acronis, AWS Backup)
- **High Availability Architecture**, Using HA designs in systems avoiding single points of failure. (Clustering, Virtualization, Container Orchestration (Kubernetes, Docker).
- **Patch Management**, Keeping systems and applications updated to prevent downtime caused by security vulnerabilities or bugs.



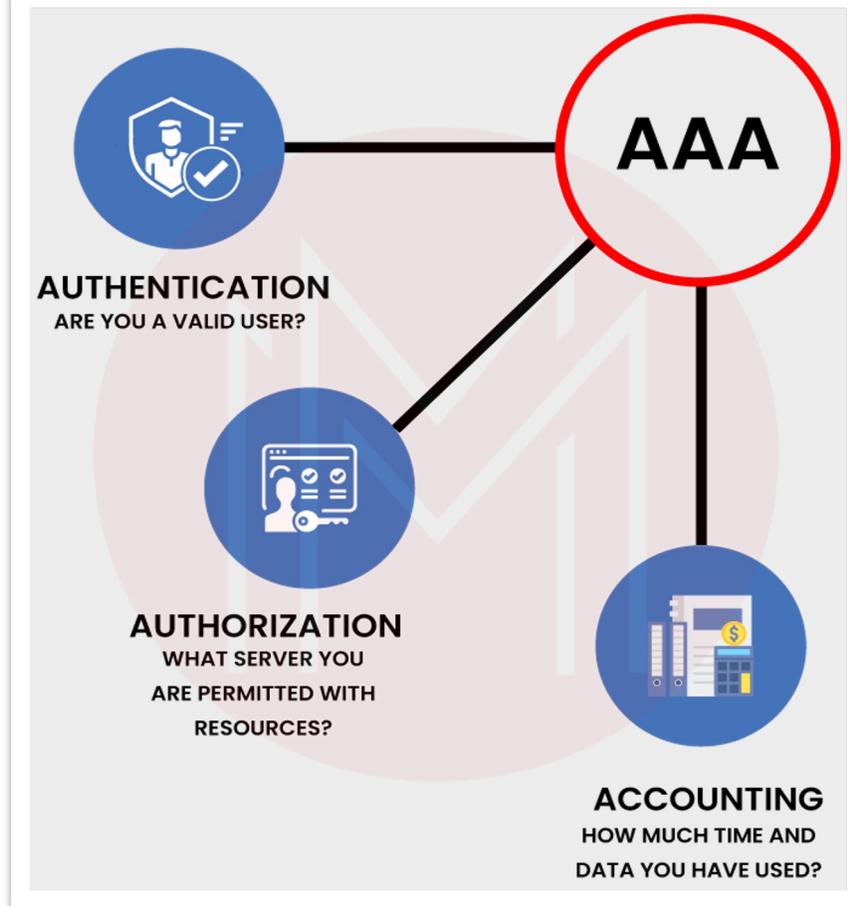
## Threats to Availability:

- DDoS
- Ransomware (for availability).
- H/W or S/W Failure.
- Natural disasters.
- Sabotage / Insider attacks.
- Resource exhaustion attacks.
- Logic bombs.

# AAA Architecture



- **Authentication** is the process of verifying the identity of a user or system trying to access a resource.
  - Something you know (password, passphrase, PIN)
  - Something you have (NIC, ATM, Passport)
  - Something you are (Biometrics)
  - Somewhere you are (Geographic location, IP, MAC address)
  - Something you do (signatures, pattern unlock)
- **Authorization** determines what an authenticated user or system is allowed to do, specifying access levels or permissions based on the user role or identity. (DAC, MAC, RBAC)
- **Accountability**, also known as auditing, involves tracking and recording user activities and resource usage. This information is used for monitoring, analysis, and compliance purposes.



# Cyber Security

## Major Categories

# Categories of Cybersecurity



## Information Security

Governance, Policies, Risk Management, Defense Strategy Across All Domains

## Application Security

Web, Mobile, API Security,  
Code Review, Browser  
Hardening, Firmware Security

## Operating System Security

Kernel Hardening, Access Control,  
Memory Protections, Sandboxing,  
SELinux

## Network Security

Firewalls, Segmentation, IDS/IPS,  
Routing, Zero Trust, VPN Architecture

## Cloud Security

VPCs, IAM, Workload Isolation, Storage Security, Secrets &  
Key Management

## IoT Security

Embedded Systems, Gateways, Radio/BLE Security, Firmware  
Integrity, Secure Boot

# Information Security



- **Information Security (InfoSec)** is the practice of protecting information from unauthorized access, disclosure, disruption, modification, or destruction.
- It aims to ensure the confidentiality, integrity, and availability (CIA Triad) of data, whether it's in storage, processing, or transit, through the use of policies, procedures, and technologies.
- Information security encompasses a wide range of security practices, including risk management, cryptography, access controls, and incident response, to protect both digital and physical information assets.



# Application Security



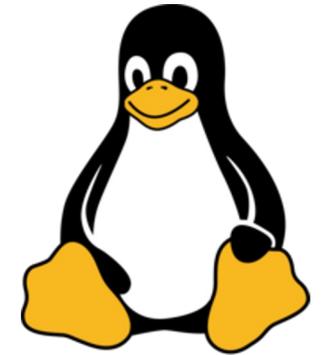
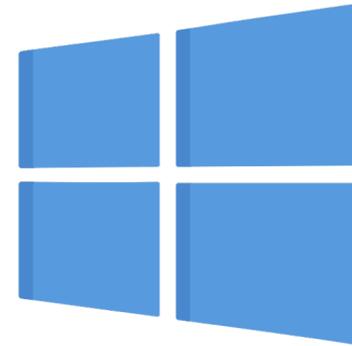
- **Application security** focuses on protecting software applications from threats and vulnerabilities that could lead to unauthorized access, data breaches, or other forms of exploitation.
- This includes techniques like secure coding, input validation, authentication, authorization, encryption, and regular security testing (such as vulnerability scanning and penetration testing).
- Application security aims to identify and fix vulnerabilities, such as SQL injection or cross-site scripting (XSS), to prevent malicious attacks.



# Operating System Security



- **OS security** focuses on protecting the operating system from vulnerabilities and threats, ensuring that the system operates securely and is resistant to attacks.
- It is achieved through following steps:
  - Regularly applying updates and patches to address vulnerabilities and improve security.
  - Configuring the OS to reduce its attack surface by disabling unnecessary services and features.
  - Access Control.
  - User Authentication.
  - File System Security.



Linux™



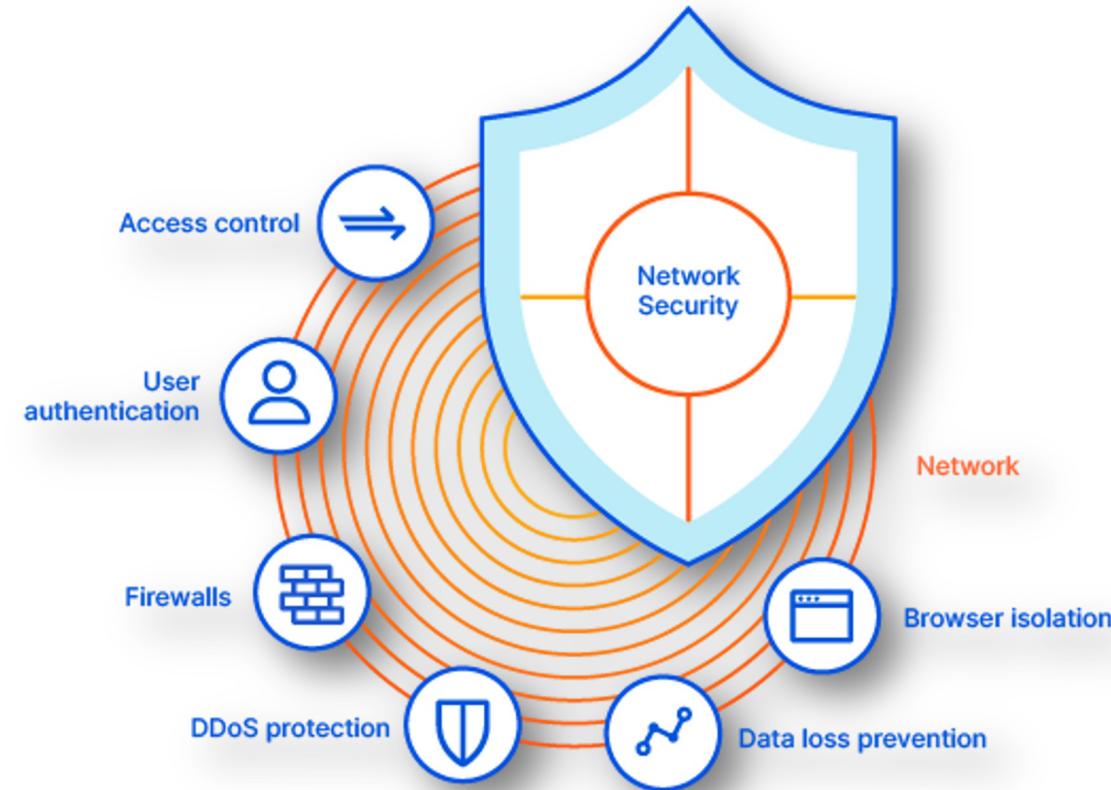
Mac™ OS



# Network Security



- **Network Security** is the practice of protecting the confidentiality, integrity, and availability of data and resources as they are transmitted or accessed across a network.
- It involves a combination of policies, procedures and technologies to prevent unauthorized access, misuse, modification, or disruption of network infrastructure.
- Network security includes measures such as encryption, security protocols, firewalls, IDS/IPS, and access controls to safeguard communication between devices and protect networks from threats like cyberattacks, malware, and data breaches.



# Cloud Security

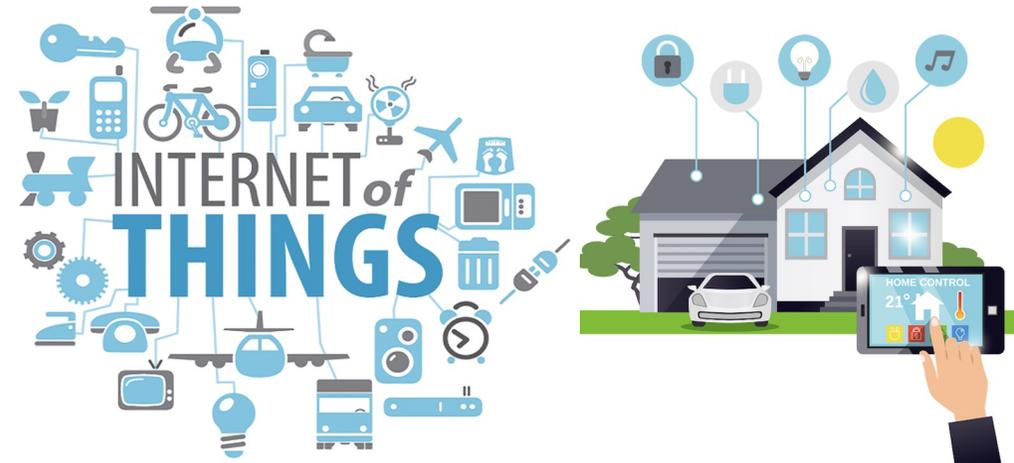


- **Cloud security** is the practice of safeguarding data, applications, and services hosted in cloud environments from unauthorized access, data breaches, and other cyber threats.
- Cloud security measures include encryption, Identity and Access Management (IAM), network security, Data Loss Prevention (DLP), and regular security monitoring.
- It addresses both the responsibilities of cloud providers (infrastructure security) and customers (secure configuration and data protection) to prevent threats like data leaks, account hijacking, and misconfigurations.



- **IoT security** is the practice of protecting Internet of Things (IoT) devices and the networks they connect to from cyber threats, unauthorized access, and vulnerabilities.
- Key IoT security practices include device authentication, encryption, secure firmware updates, network segmentation, and monitoring for unusual activity.
- IoT security is critical because these devices often have limited processing power, making them more susceptible to attacks, and they can serve as entry points for threats like malware or unauthorized control in broader network environments.

Collection of interconnected devices that communicate and transfer data through the Internet



# The Future: Security of Gen-AI Models



## *Weaknesses in LLM applications that attackers can exploit to steal data, manipulate outputs, or harm systems*

- OWASP released the “Top 10 LLM Vulnerabilities” list to highlight how AI systems can be attacked through prompts, data, or integrations.
- Attacks against AI systems can be categorized into three broad categories:

- ✓ **Prompt-Level Attacks:** Attackers manipulate what you type to trick the AI into breaking its rules or revealing secrets. (LLM01, LLM06, LLM07)
- ✓ **Data-Level Attacks:** Attackers poison the training data the AI learns from, or exploit vulnerabilities in third-party components, causing the AI to produce wrong, biased, or harmful outputs. (LLM02, LLM03, LLM04)
- ✓ **System-Level Attacks:** The AI's outputs or actions aren't properly checked before being used, allowing attackers to inject malicious code, cause excessive resource usage, or manipulate how the AI retrieves information. (LLM05, LLM08, LLM09, LLM10)

## 2025

LLM01: Prompt Injection Attacks
LLM02 : Sensitive Information Disclosure
LLM03 : Supply Chain
LLM04 : Data and Model Poisoning
LLM05 : Improper Output Handling
LLM06 : Excessive Agency
LLM07 : System Prompt Leakage
LLM08 : Vector and Embedding Weaknesses
LLM09 : Misinformation
LLM10 : Unbounded Consumption

# History of Cyber Attacks

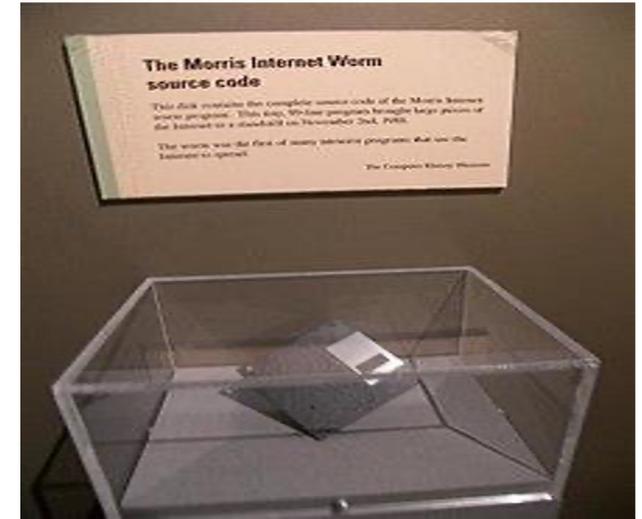
# Famous Cyber Attacks



**Morris Worm (1988):** Often considered one of the first major worms, created by Robert Tappan Morris. It exploited sendmail service to achieve RCE and exploited BoF vulnerability in the finger daemon to retrieve user information in Unix systems and spread across the internet. It infected around 6,000 computers, causing significant disruption by slowing down systems and making them unusable.

**Melissa Virus (1999):** The Melissa virus was a computer virus that spread quickly through email attachments in 1999. It primarily exploited vulnerabilities in Microsoft Word Macros and Outlook users, and was one of the first viruses to raise awareness of the risks of opening unsolicited emails.

**ILOVEYOU Worm (2000):** The ILOVEYOU Worm, also known as the Love Bug, spread through email with a subject line that read "I Love You." It used a social engineering trick to lure recipients into opening an infected attachment. It caused widespread damage, affecting 10 million computers globally, and led to billions of dollars in damages.

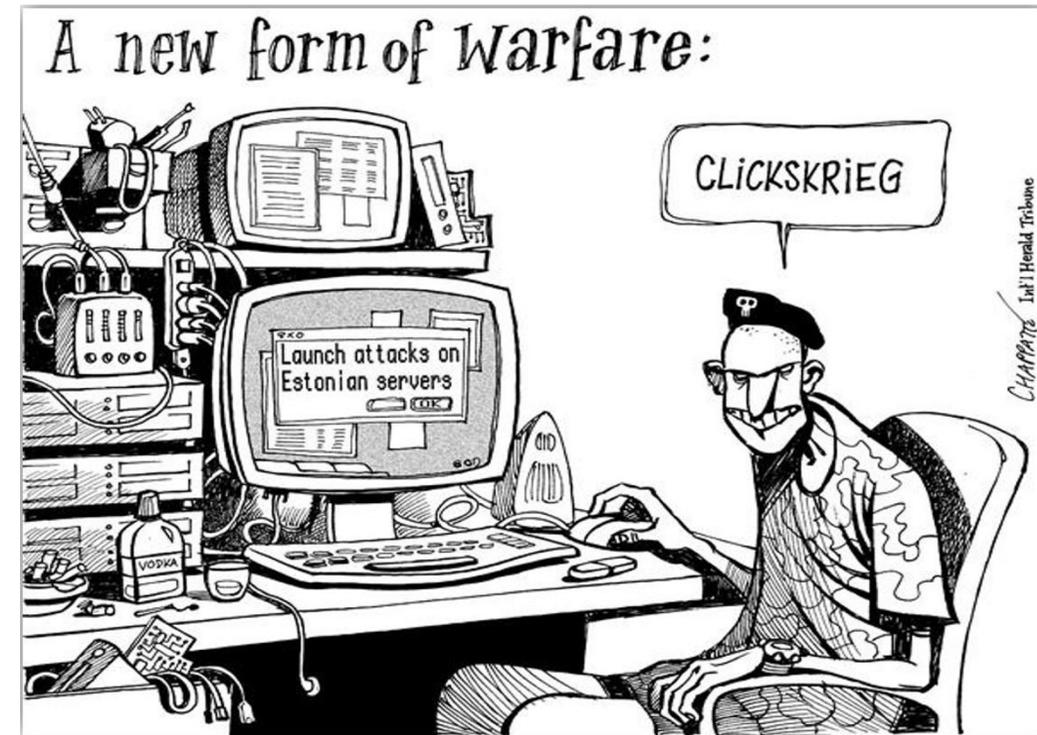


# Famous Cyber Attacks: Estonia



**Cyber Attacks against Estonia (2007):** In 2007, Estonia experienced a series of major cyber attacks, widely regarded as one of the first large-scale, politically motivated, state-sponsored cyber warfare campaigns.

- These attacks targeted government institutions, banks, media outlets, and other critical infrastructure, effectively disrupting the nation's online services for several weeks.
- It involved DDoS (Distributed Denial of Service) attacks, overwhelming servers with traffic to shut down websites and services.
- Russia was widely suspected of being behind the attacks.

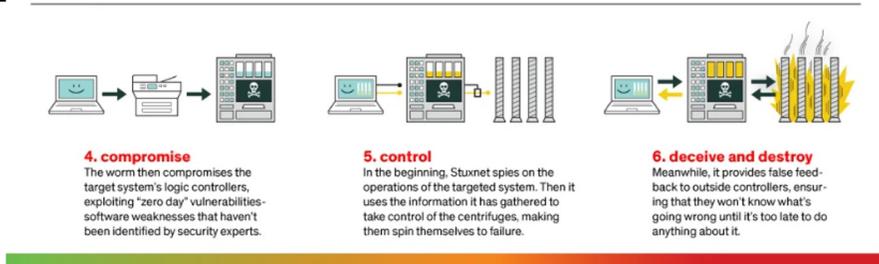
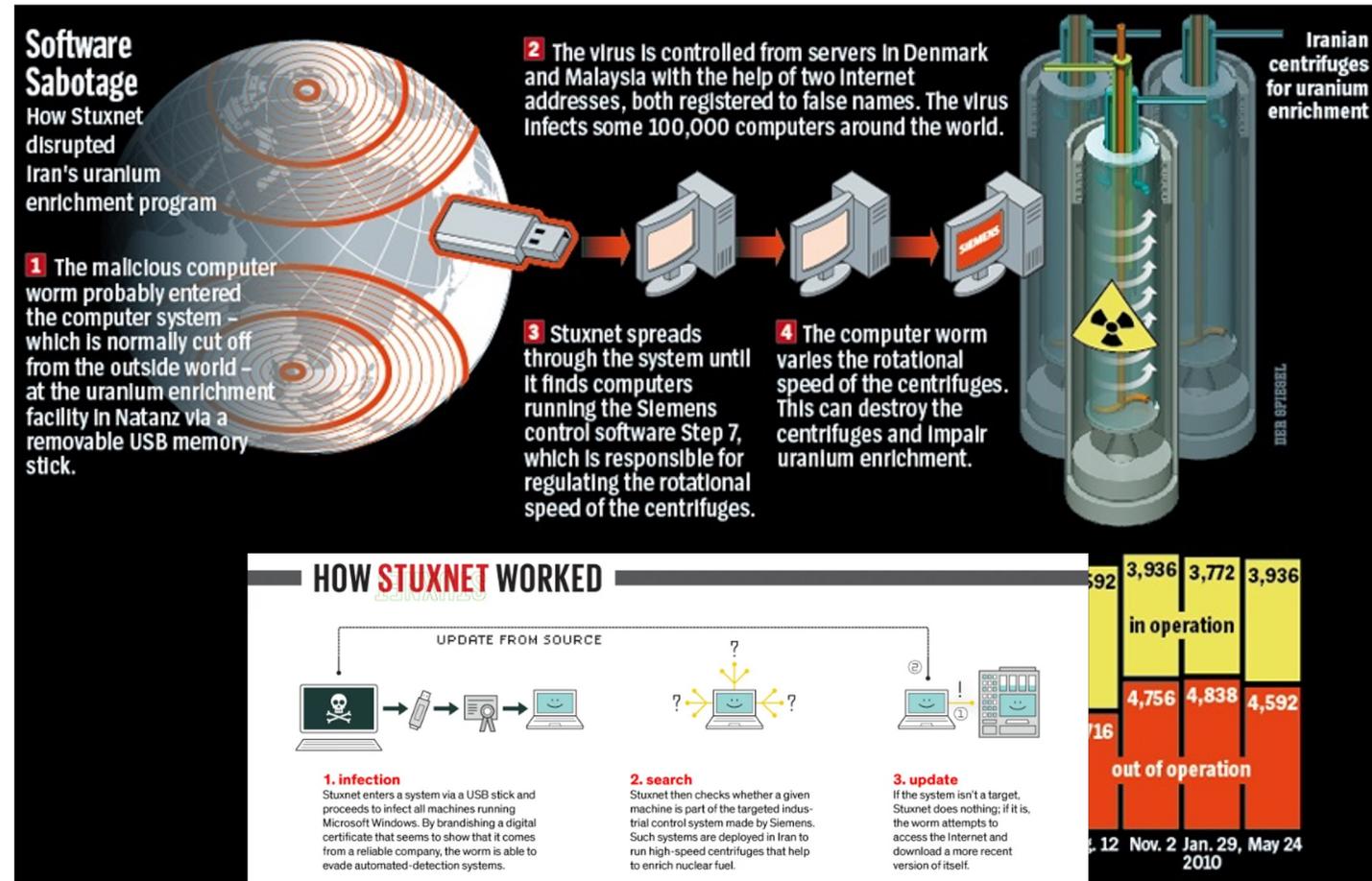


# Famous Cyber Attacks: Stuxnet



## Stuxnet (2010):

- Stuxnet was a sophisticated worm believed to be created by the U.S. and Israeli governments to target Iran's nuclear facilities. Worm spread through Windows computers, but its primary function was to cause physical damage to the centrifuges by altering their speeds without detection, while simultaneously sending normal operating signals to monitoring systems.
- It is considered one of the first known cyber weapons and demonstrated the potential for cyber-attacks to cause physical damage.



# Famous Cyber Attacks



## Sony PlayStation Network Hack (2011):

- Attackers compromised Sony's PlayStation Network (*PSN Hack*), gaining access to personal information of around 77 million users.
- They used a combination of phishing techniques and SQL injection to gain unauthorized access to Sony's servers. Data Breach forced Sony to shut down the PSN for nearly a month.
- Sony faced financial loss of \$171 million approx., making it one of the largest security breaches of its time.
- The breach led to the suspension of the network, significant financial losses, and a major hit to Sony's reputation.



# Famous Cyber Attacks: WannaCry Ransomware



## WannaCry Ransomware Attack (2017):

- WannaCry attacked computers worldwide by exploiting a vulnerability in Microsoft's Server Message Block protocol.
- Vulnerability ID: CVE-2017-0144
- Exploit used: EternalBlue, developed by NSA, leaked by shadow brokers hacking group.
- It encrypted users' files and demanded ransom payments in Bitcoin.
- The attack affected more than 300,000 computers across 150 countries, with total damage ranging from hundreds of millions to billions of dollars.
- Attack caused significant financial losses, halted critical services, and highlighted the dangers of outdated software and unpatched security vulnerabilities.



# Famous Cyber Attacks: Lockbit



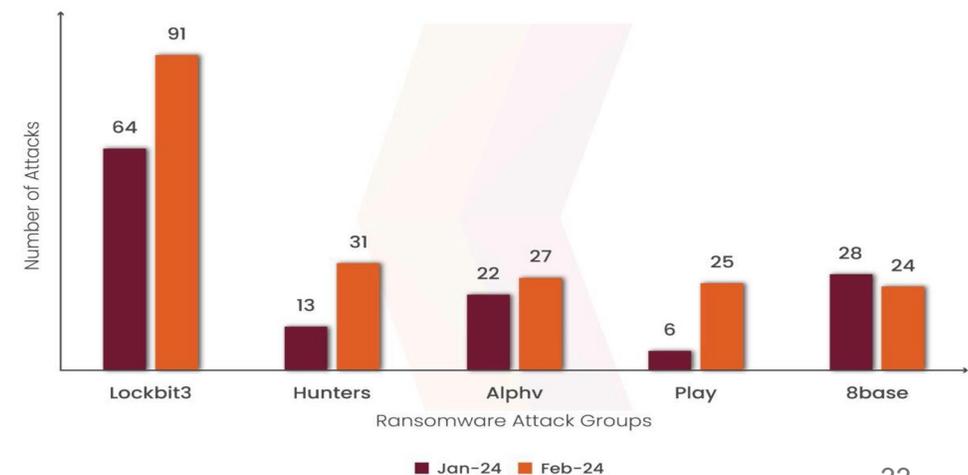
## LockBit (2019):

- A notorious ransomware group known for its multiple versions of ransomware, including LockBit1.0 (2019), LockBit2.0 (2021), LockBit3.0 (2022) and LockBit Black (2023).
- Each version has become more sophisticated, with enhanced encryption and better evasion tactics. The group operates under a ransomware-as-a-service (RaaS) model, allowing other cybercriminals to use their software for a fee.
- Attacked financial services, healthcare and transportation sectors.
- Used in approximately 1,700 ransomware attacks in US between January 2020 to and May 2023, with US\$91 million paid as ransom.
- Also declared as the world's most prolific ransomware in 2022.

Instructor: Muhammad Arif Butt, PhD



## Lockbit Attack on the Rise

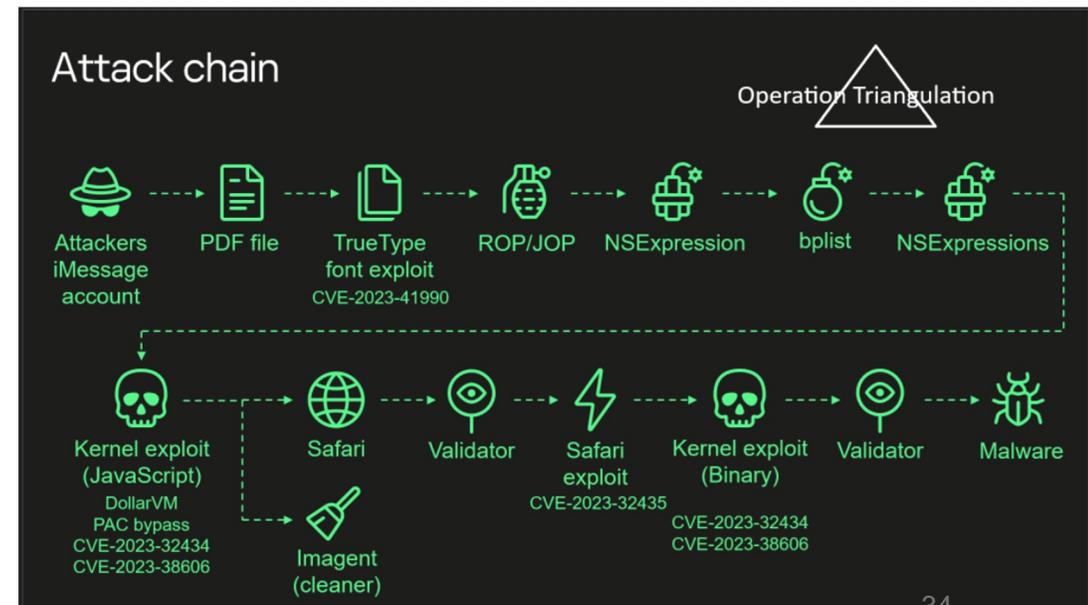
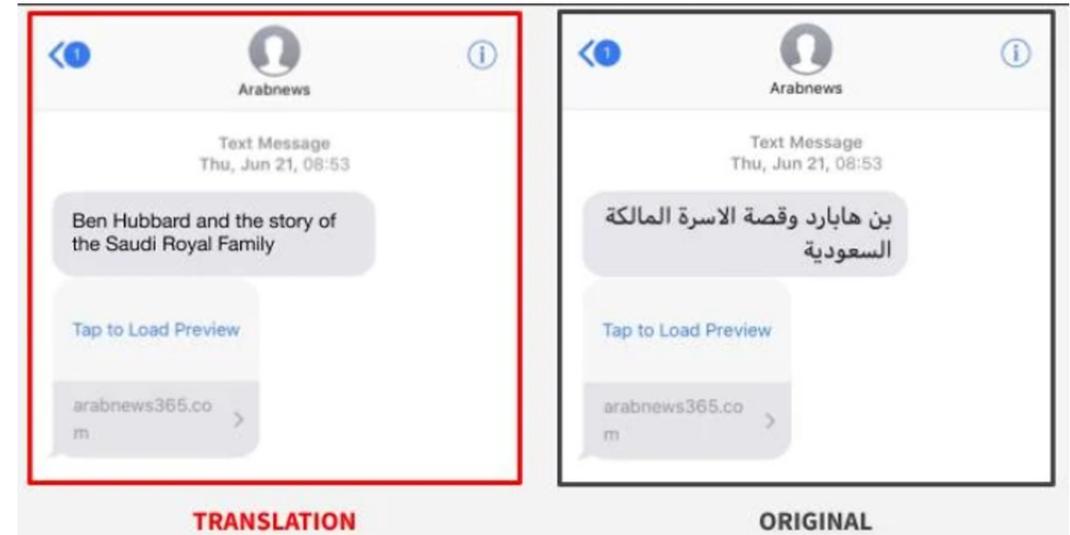


# Famous Cyber Attacks: Pegasus



**Pegasus (Forced Entry) (2021):** Pegasus (Forced Entry) refers to a specific exploit of the Pegasus spyware, developed by Israeli company NSO group. It explored a zero-day in Apple's iMessage, allowing the spyware to be installed on a device without any interaction from the victim. This attack didn't require the victim to click on a malicious link or open a compromised file; it could be silently triggered just by receiving a specially crafted message. Once installed, Pegasus could monitor calls, messages, emails, and even activate microphones and camera

**Operation Triangulation (2023):** This operation utilized Pegasus spyware to silently compromise mobile devices and gain access to sensitive data. Operation Triangulation is a targeted cyberattack on iOS devices conducted using a chain of four zero-day vulnerabilities.



# Famous Cyber Attacks: Mirari Botnet



## IoT (Internet of Things) Attacks

- **Mirai Botnet (2021):** The Mirai botnet, which was initially used for DDoS attacks, continues to exploit vulnerabilities in IoT devices to launch large-scale attacks.
- IoT device vulnerabilities can lead to widespread disruptions and can be used to launch attacks on various services. It stresses the importance of securing connected devices and updating their software regularly.



# Famous Cyber Attacks (Cont..)



**Deepfake AI:** Deepfake AI poses significant challenges to cybersecurity, as it can be used to create highly convincing fake audio, video, and text content that can deceive individuals and manipulate public opinion

- Social Engineering Attacks
- Phishing and Fraud
- Reputation Damage
- Misinformation and Disinformation Campaigns
- Authentication and Trust Issues



# Famous Cyber Attacks (Cont..)



ispr.gov.pk

## Hacked By Benaam Pakistani

Pakistan Zindabad

Dear Pakistani's

**In the memories of those who sacrifice their life for us,**

**It is to inform that Pakistan Armed Forces are the best in the World.**

**They've made a lot of sacrifices for this country and were the true defenders of this motherland.**

**But, they must not misuse their power against their own people.**

**They should work for the betterment of this country rather than killing it like a parasite.**

**Meanwhile, the politicians too are required to obey by their oath.**

**They should work to make the country a much better place to live in.**

**Islamic principles and teachings should be opted in every field especially in the economic aspect (interest free).**

**Moreover, the citizens are also advised to rectify their character and code of conduct. They must leave all the activities which tends out to be a harm for the country. It is time to unite!**

## Over 2 million Pakistanis' data hacked from restaurants database up for sale



**KARACHI:** Bykea, a popular ride-hailing, mobility, and delivery app used by many Pakistanis, experienced a disturbing hacking incident on Tuesday, causing widespread concern among users.

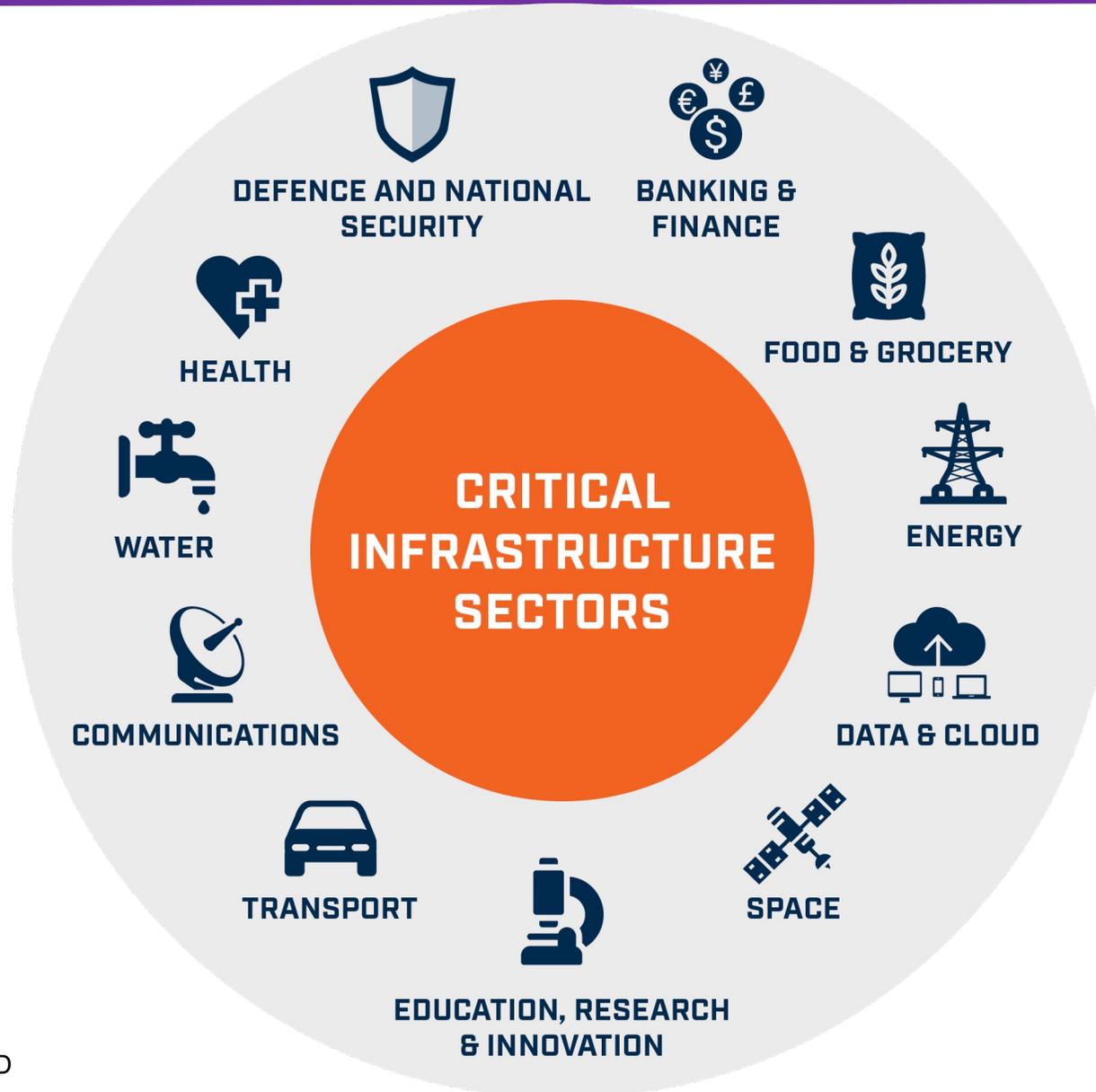
The ordeal began when users started receiving highly inappropriate text messages, alarming and unsettling them. Bykea promptly addressed the issue and issued an apology for the offensive messages.

# Impacts of Cyber Attacks



- **Financial Losses:** Direct financial losses due to theft, ransom payments, and fraud, as well as indirect costs from downtime, recovery efforts, and legal fees.
- **Operational Disruption:** Interruption of business operations, affecting productivity and service delivery. Critical infrastructure attacks can lead to widespread disruptions in essential services.
- **Reputational Damage:** Loss of customer trust and damage to the organization's reputation, which can affect future business prospects and customer relationships.
- **Data Privacy Concerns:** Exposure of personal and sensitive data, leading to increased risks of identity theft, phishing, and privacy violations.
- **Regulatory and Legal Consequences:** Increased scrutiny from regulators and potential legal actions, resulting in fines and compliance costs.
- **National Security Risks:** Espionage and attacks on critical infrastructure can have implications for national security and geopolitical stability.

# National Critical Infrastructure



# Cybersecurity Roles

# Security Battle: Red Team vs Blue Team



A red team/blue team exercise is a simulated cybersecurity battle where offensive experts (red team) emulate real-world attackers to breach defences, while defensive responders (blue team) detect, analyze, and counter their attacks to strengthen organizational resilience.



Instructor: Muhammad Arif Butt, PhD

RED TEAM	VS	BLUE TEAM
<p>Simulated adversary, attempting to identify and exploit potential weaknesses within the organization's cyber defenses...</p> 		<p>Incident response consultants guide the IT security team on where to make improvements to stop sophisticated types of cyberattacks and threats...</p> 
<p>...identifying an attack path that breaches the organization's security defense through real-world attack techniques</p>		<p>...leaving the IT security team responsible for maintaining the internal network against various types of risk</p>



# Cybersecurity Roles & Domains



- Penetration Testing
- Reverse Engineering
- Binary Exploitation
- Vulnerability Research & Exploit Development
- Malware Development & Analysis

# Penetration Testing



Penetration testing (pen testing) is a systematic process used to identify and exploit vulnerabilities in Networks, Websites, Mobile apps, LLM apps and Cloud Services

## Reconnaissance & Information Gathering:

*Collect public and internal information to build an attack surface and plan tests*

- **Networks:** Enumerate IP ranges, open ports, services, ASNs, DNS, and exposed management interfaces using tools like Wireshark, Nmap, host, nslookup, dig, whois, traceroute, netdiscover, shodan
- **Websites:** Gather hostnames/subdomains, directories, frameworks, API endpoints, public parameters, robots.txt, DNS records using tools like WhatWeb, theHarvester, KnockPy, wafw00f, OWASP Amass, Google dorking, OSINT framework
- **Mobile Apps:** Collect app metadata, package names, endpoints used by the app, permissions, backend hosts, certificate pinning detection using tools like MobSF, jadx, apktool, frida,, drozer, objection, sherlock
- **LLM Apps:** Gather information about exposed endpoints, RAG stores, repository secrets, connectors, prompt templates, sanitizers, using tools like OWASP Amass, Gitleaks, OSINT framework, custom API-probing scripts

Recce & Info  
Gathering

Scanning & Vuln  
Analysis

Exploitation &  
Gaining Access

Post Exploitation &  
C2

Covering Tracks

Analysis &  
Reporting

# Penetration Testing



Penetration testing (pen testing) is a systematic process used to identify and exploit vulnerabilities in Networks, Websites, Mobile apps, LLM apps and Cloud Services

## Scanning & Vulnerability Analysis:

*Actively probe and detect vulnerabilities and misconfigurations identified during recce*

Recce & Info  
Gathering

Scanning & Vuln  
Analysis

Exploitation &  
Gaining Access

Post Exploitation &  
C2

Covering Tracks

Analysis &  
Reporting

- **Networks:** Port/service scanning, vulnerability fingerprinting (CVEs), weak TLS/SSL configs, open SMB/RDP, unsecured SNMP, banner grabbing using tools like Nmap scripting engine, Nessus, OpenVAS, Tripwire, MSF
- **Websites:** Active scans for injection flaws, insecure session management, insecure direct object references (IDOR), CSRF, XXE, security misconfigs using tools like Burp Suite, OWASP ZAP, sqlmap, Nikto
- **Mobile Apps:** Scan components for insecure data storage, dynamic testing for insecure network traffic, authentication bypass, insecure IPC using tools like MobSF, Frida, Drozer, Burp/ZAP
- **LLM Apps:** Test for prompt injection, jailbreaks, data leakage, insecure output handling, training data poisoning risks, model extraction vulnerabilities using tools like Adversarial Robustness Toolbox (ART), TextAttack, Foolbox, Gitleaks, custom API fuzzers

Penetration testing (pen testing) is a systematic process used to identify and exploit vulnerabilities in Networks, Websites, Mobile apps, LLM apps and Cloud Services

## Exploitation & Gaining Access:

*Exploit identified vulnerabilities to gain unauthorized initial access*

Recce & Info  
Gathering

Scanning & Vuln  
Analysis

Exploitation &  
Gaining Access

Post Exploitation &  
C2

Covering Tracks

Analysis &  
Reporting

- **Networks:** Exploit CVEs, brute-force credentials, exploit misconfigurations, leverage default credentials, using tools like MSF, Cobalt Strike, msfvenom, Hashcat, John the Ripper
- **Websites:** Execute SQL injection, XSS attacks, authentication bypass, file upload exploitation, API abuse using tools like Burp Suite Intruder/Repeater, custom scripts, sqlmap, BeEF
- **Mobile Apps:** Exploit insecure authentication, API vulnerabilities, reverse engineering for hardcoded secrets, runtime manipulation using tools like Frida, Objection, Drozer, SSL unpinning tools, debuggers
- **LLM Apps:** Perform prompt injection attacks, bypass content filters, extract training data, manipulate agent behavior, exploit RAG vulnerabilities using tools like Adversarial Robustness Toolbox (ART), TextAttack, Gitleaks, custom exploit scripts

Penetration testing (pen testing) is a systematic process used to identify and exploit vulnerabilities in Networks, Websites, Mobile apps, LLM apps and Cloud Services

## Post Exploitation & C2:

*Escalate privileges and establish control/command channels to move laterally and exercise deeper compromise*

- **Networks:** Escalate using weak service configs, kernel exploits, credential reuse, and maintain access via C2 channels for lateral movement using tools like Metasploit, Cobalt Strike, BloodHound
- **Websites:** Establish webshells, create backdoor admin accounts, identify paths for re-entry using custom webshells, session hijacking tools
- **Mobile Apps:** Root/jailbreak device, install persistent backdoors, access sensitive storage by device rooting, and hooking frameworks persistence
- **LLM Apps:** Escalate by chaining prompt injections across components (e.g., induce RAG retrieval of sensitive docs, then prompt model to expose them), or inducing agent behaviors that call external connectors (email, webhooks) to exfiltrate

Recce & Info  
Gathering

Scanning & Vuln  
Analysis

Exploitation &  
Gaining Access

Post Exploitation &  
C2

Covering Tracks

Analysis &  
Reporting

# Penetration Testing



Penetration testing (pen testing) is a systematic process used to identify and exploit vulnerabilities in Networks, Websites, Mobile apps, LLM apps and Cloud Services

## Covering Tracks:

*Remove or obfuscate evidence of compromise and restore systems to pre-test state*

- **Networks:** Clear logs, tamper timestamps, remove created accounts or artifacts using log clearing/detection-evasion and MITRE Defense Evasion techniques.
- **Websites:** Remove web shell traces or obfuscated payloads, manipulate application logs or monitoring hooks.
- **Mobile Apps:** Clear app logs/telemetry, delete local artifacts in a controlled lab to demonstrate detection gaps.
- **LLM Apps:** Erase injected prompts from prompt history/logs, remove malicious entries from vector DBs, or obfuscate previous queries — important to test retention, audit logging and log-integrity controls. Defensive focus: robust immutable audit trails and retention policies

Recce & Info  
Gathering

Scanning & Vuln  
Analysis

Exploitation &  
Gaining Access

Post Exploitation &  
C2

Covering Tracks

Analysis &  
Reporting

# Penetration Testing



Penetration testing (pen testing) is a systematic process used to identify and exploit vulnerabilities in Networks, Websites, Mobile apps, LLM apps and Cloud Services

## Analysis & Reporting:

*Document findings, assess risk, and provide actionable remediation guidance*

Follow industry reporting standards (CVSS v3.1+, NIST frameworks) including:

- Executive summary
- Methodology and scope
- Detailed findings with severity ratings (CVSS scores)
- Proof-of-concept evidence
- Remediation recommendations
- Risk assessment and business impact

Recce & Info  
Gathering

Scanning & Vuln  
Analysis

Exploitation &  
Gaining Access

Post Exploitation &  
C2

Covering Tracks

Analysis &  
Reporting

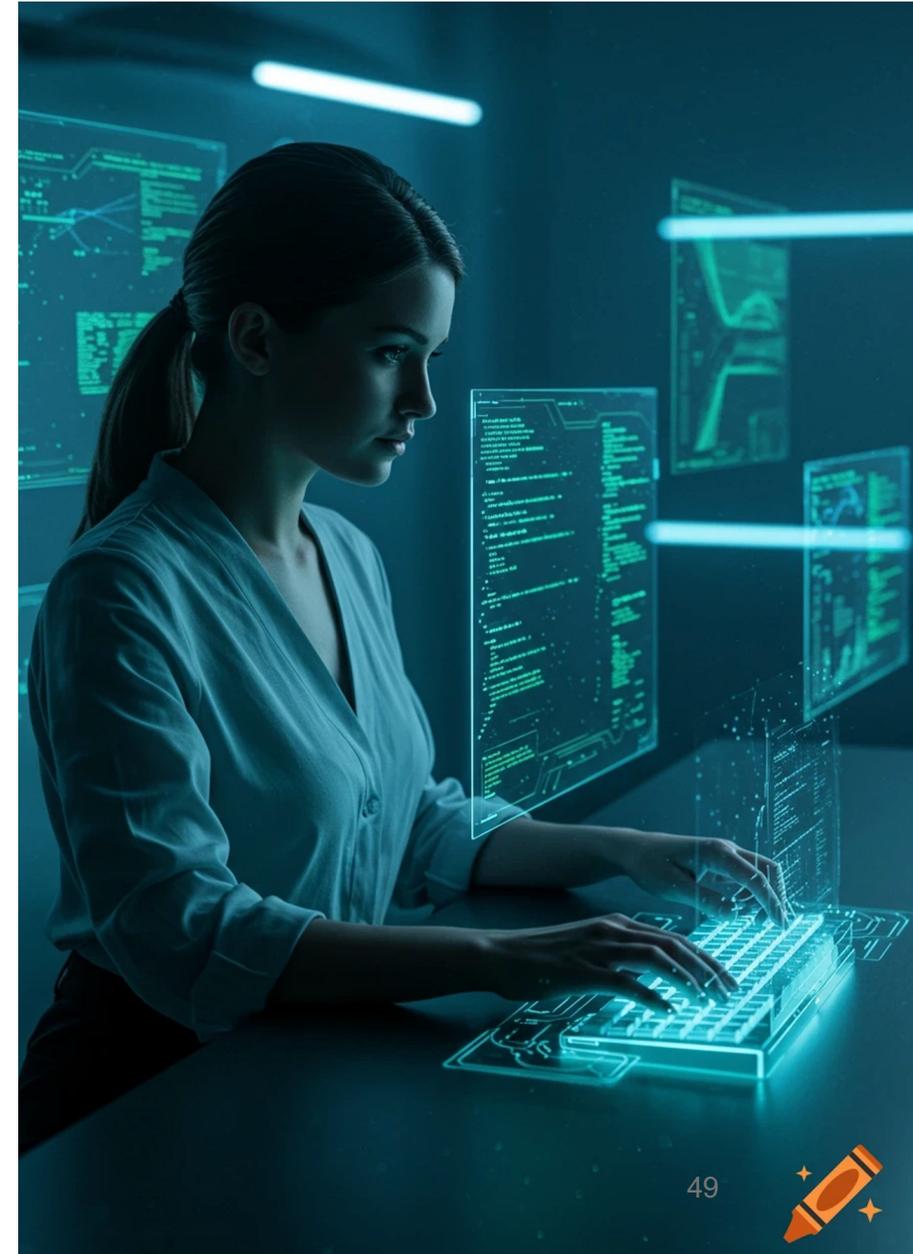
# Reverse Engineering



- **Reverse engineering** in cybersecurity refers to the process of analyzing and disassembling software, hardware, or protocols to understand their inner workings, often with the goal of extracting valuable information, identifying vulnerabilities, malicious behavior or weaknesses.
- Key purposes of reverse engineering are:
  - Closed source code analysis
  - License reversing
  - Vulnerability research
  - Malware analysis
  - Digital forensics

## RE Tools:

- Ghidra
- IDA Pro
- Binary Ninja
- Cutter / radare2
- Jadx, apktool, MobSF, Frida





# Vulnerability Research & Exploit Development



Opcode	Description	EAX	ECX	EDX	EBX	ESP	EBP	ESI	EDI	EIP
CALL	Call subroutine	FF	D0	D1	D2	D3	D4	D5	D6	D7
DEC	Decrement	48	49	4A	4B	4C	4D	4E	4F	
INC	Increment	40	41	42	43	44	45	46	47	
JMP	Unconditional Jump	EB	00h-7Fh: Forward jump, 80h-FFh: Reverse jump							
MOV	Move (Copy)	BB	B9	BA	BB	BC	BD	BE	BF	
PUSH	Push onto Stack	68	50	51	52	53	54	55	56	57
POP	Pop from Stack	58	59	5A	5B	5C	5D	5E	5F	
XCHG	Exchange eAX with	91	92	93	94	95	96	97		
XOR	Logical exclusive OR	33	C0	C9	D2	DB	E4	ED	F6	FF
ADD	Add									05
AND	Logical and									25
CMP	Compare eAX									3D
INT	Interrupt	CD								
INT3	Interrupt 3	CC								
JNA	Jump if above	77								
JB	Jump if below	72								
JBE	Jump if below or equal	76								
JCXZ	Jump if CX zero	E3								
JE	Jump if equal	7E								
JL	Jump if less than	7C								
JLE	Jump if less than or equal	7E								
JNB	Jump if not below	73								
JNL	Jump if not less	7D								
JNLE	Jump if not less or equal	7F								
JNO	Jump if no overflow	71								
JNS	Jump if no sign (positive)	79								
JNZ	Jump if not zero	75								
JO	Jump if overflow	70								
JR	Jump if sign (negative)	78								
JZ	Jump if zero	74								
LOOP	Loop operation	E2								
NOP	No operation	90								
NOT	Invert each bit									
OR	Logical or									0D
RET	Return from subroutine	C3								
RET +OFF	Return + Offset	C2								
SUB	Subtract									2D
WAIT	Suspend exec until BUSY # pin	98								

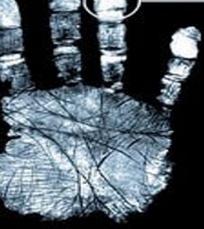
```
get all shellcode on binary file from objdump
objdump -d binary.file.o | grep '[0-9a-f]' | grep -v 'file' | cut -f2 -d: | cut -f1-6 -d: | tr -s ' ' | tr '\t' ' ' | sed 's/ / /g' | paste -d ' ' -s | sed 's/ / /g' | sed 's/ / /g'
```

de	hx	char	description	dec	hx	char	dec	hx	char	dec	hx	char
0	0	NUL	null	32	20	Space	64	40	@	96	60	
1	1	SOH	start of header	33	21	!	65	41	A	97	61	a
2	2	STX	start of text	34	22	"	66	42	B	98	62	b
3	3	ETX	end of text	35	23	#	67	43	C	99	63	c
4	4	EOT	end of trans	36	24	\$	68	44	D	100	64	d
5	5	ENQ	Enquiry	37	25	%	69	45	E	101	65	e
6	6	ACK	Acknowledge	38	26	&	70	46	F	102	66	f
7	7	BEL	Bell	39	27	'	71	47	G	103	67	g
8	8	BS	Backspace	40	28	(	72	48	H	104	68	h
9	9	TAB	horizontal tab	41	29	)	73	49	I	105	69	i
10	0a	LF	line feed	42	2a	*	74	4a	J	106	6a	j
11	0b	VT	vertical tab	43	2b	+	75	4b	K	107	6b	k
12	0c	FF	form feed	44	2c	,	76	4c	L	108	6c	l
13	0d	CR	carriage return	45	2d	-	77	4d	M	109	6d	m
14	0e	SO	shift out	46	2e	.	78	4e	N	110	6e	n
15	0f	SI	shift in	47	2f	/	79	4f	O	111	6f	o
16	10	DLE	data link escape	48	30	0	80	50	P	112	70	p
17	11	DC1	device control 1	49	31	1	81	51	Q	113	71	q
18	12	DC2	device control 2	50	32	2	82	52	R	114	72	r
19	13	DC3	device control 3	51	33	3	83	53	S	115	73	s
20	14	DC4	device control 4	52	34	4	84	54	T	116	74	t
21	15	NAK	negative ack	53	35	5	85	55	U	117	75	u
22	16	SYN	synchronous idle	54	36	6	86	56	V	118	76	v
23	17	ETB	end trans block	55	37	7	87	57	W	119	77	w
24	18	CAN	Cancel	56	38	8	88	58	X	120	78	x
25	19	EM	end of medium	57	39	9	89	59	Y	121	79	y
26	1a	SUB	Substitute	58	3a	:	90	5a	Z	122	7a	z
27	1b	ESC	Escape	59	3b	;	91	5b	[	123	7b	[
28	1c	FS	file separator	60	3c	<	92	5c	\	124	7c	\
29	1d	GS	group separator	61	3d	=	93	5d	]	125	7d	]
30	1e	RS	record separator	62	3e	>	94	5e	^	126	7e	^
31	1f	US	unit separator	63	3f	?	95	5f	_	127	7f	_

```
python struct()
Format C Type Python type Size
x pad byte no value -
c char string len(1) 1
b signed char integer 1
B unsigned char integer 1
? Bool bool 1
h short integer 2
H unsigned short integer 2
i int integer 4
I unsigned int integer 4
l long integer 4
L unsigned long integer 4
q long long integer 8
Q unsigned long long integer 8
f float float 4
d double float 8
s char[] string -
c char[] string -
p void * -
>>> pack('hhi', 1, 2, 3)
>>> unpack('hhi', '\x00\x01\x00\x02\x00\x00\x00\x03')
```

```
SCAPY packet = IP(dst="<dest>", src="<src>") / TCP(dport=80, flags="S")
unans, ans = sr(packet)
is() List all handled proto
is(<PROTO>()) List <proto> defaults
fuzz() Randomize specified proto field
send() Send packet
sr(packet, retry=5, timeout=1.5, iface="eth0", filter="host 1.2.3.4 and port 80")
sr() Send/Receive packet
sr1() Returns only 1 packet answered
srp() Send layer 2 proto packets
srp1() Returns only 1 packet answered
ans.show() Formatted packet contents/flags
ans.summary() Summary of handled packets in ans
tracertool()
tracertool(["10.1.99.2", "www.google.com", "www.citrix.com"], maxttl=15)
sniff() Packet sniffer
packets = sniff(filter="host <host>", count=<num>)
rdpcap() Read pcap
packets = rdpcap("filename.pcap")
wpcap() Write pcap
wpcap("filename.pcap", packets)
```

```
get all shellcode on binary file from objdump
objdump -d binary.file.o | grep '[0-9a-f]' | grep -v 'file' | cut -f2 -d: | cut -f1-6 -d: | tr -s ' ' | tr '\t' ' ' | sed 's/ / /g' | paste -d ' ' -s | sed 's/ / /g' | sed 's/ / /g'
```



Format	C Type	Python type	Size
x	pad byte	no value	-
c	char	string len(1)	1
b	signed char	integer	1
B	unsigned char	integer	1
?	Bool	bool	1
h	short	integer	2
H	unsigned short	integer	2
i	int	integer	4
I	unsigned int	integer	4
l	long	integer	4
L	unsigned long	integer	4
q	long long	integer	8
Q	unsigned long long	integer	8
f	float	float	4
d	double	float	8
s	char[]	string	-
c	char[]	string	-
p	void *	void *	-

```
Nmap - additional flags
--send-eth/--send-ip: send using raw ethernet frames or ip packets
--script "http-": Scripts with wildcard
--source-port <portnumber> Spoof source port number
--state-every <time> Update state during runtime at set <interval>
--open-reason: Return only open services, and reason why
--data-length 0: 0 for no random or protocol-specific payload
--randomize-hosts: Default is sequential hosts
--data hex string: Append custom binary data to sent packets
e.g. --initial-rtt-timeout 250ms --max-retries 2 --host-timeout 15m
```

```
GDB - basic commands
display/ Sp would display the next instruction after each step
disassemble Disassemble the current function or
disassemble <where> given location.
info args Print arguments to function of the current stack frame.
info breakpoints Print informations about the break- and watchpoints.
info display Print informations about the "displays".
info locals Print local variables in the currently selected stack frame.
info registers Print of register registers and their contents
info sharedlibrary List loaded shared libraries.
info signals List all signals and how they are currently handled.
info threads List all threads.
show directories Print all directories that GDB searches for source files.
show listsize print how many are shown in the "list" command.
whatis variable name print type of named variable.
w/nfu <address> print memo
```



PE HEADER	DOS MZ Header	DOS Stub	PE File Header	PE Signature	Image Optional Header	Section Table	Array of Image Selection Headers
							Data Directories
							Sections
							.idata
							.rsc
							.data
							.text
							.src

```
Scapes NTAccessCheckAndAuditAlarm Egghunter
%x00\x01\xca\xff\x01 OR dx, 0xffff
%x42 inc edx
%x52 push edx
%x6a\x02 push byte +0x02
%x58 pop eax
%xcd\x2e int 0x2e
%x3c\x05 cmp al, 0x05
%x5a restore edx
%x2d\xef jmp xxxxx
%ab\x5d\x30\x30\x57 mov eax, 0x57303054
%8b\x4a mov edi, edx
%af scasd
%75\x0a jnz xxxxxxxx
%af scasd
%75\x02 jmp xxxxxxxx
%af\x07 jmp to inc edi if only one egg found
%af\x07 jmp to beginning of shellcode
```

```
SSH Tunneling
- execute from source: ssh -f -N -i <source> <dest> <port> user@relay
- execute from dest: ssh -f -N -R <source> <port> <dest> <port> user@relay
```

```
Netcat Listeners
nc -l -p <port> -s <addr>
ncat -l -k -p <port>

Basic CLI one-liners
perl -e 'print "A" x <buffer-space>'
python -c 'print "A" * <buffer-space>'
ruby -e 'put "A" * <buffer-space>'
```

```
PEDA - basic commands
Show/set ASLR setting of GDB
checksec Check for security options of binary
dumprags Display args passed to function
dumprop Dump all ROP gadgets in mem range
elfheader Get headers info from ELF file
elfsymbol Get non-debug symbol info from ELF file
lookup Search for all addr/ref to addr in mem range
patch Patch mem start at addr: str/hex/int
pattern Gen, search, write pattern to mem
proclinfo Display various info from /proc/pid/
pshow Show PEDA options & other settings
pset Set PEDA options & other settings
readelf Get headers information from ELF file
ropgadget Get common ROP gadgets of binary or library
ropsearch Search for ROP gadgets in mem
searchmem [flag] Search for a pattern in memory; support regex search
shellcode Gen or download common shellcode
skeleton Gen python exploit code template
vmmmap Get virtual mapping address ranges of section[s] in debugged process
xormem XOR a memory region with a key
```

```
python struct()
Format C Type Python type Size
x pad byte no value -
c char string len(1) 1
b signed char integer 1
B unsigned char integer 1
? Bool bool 1
h short integer 2
H unsigned short integer 2
i int integer 4
I unsigned int integer 4
l long integer 4
L unsigned long integer 4
q long long integer 8
Q unsigned long long integer 8
f float float 4
d double float 8
s char[] string -
c char[] string -
p void * -
>>> pack('hhi', 1, 2, 3)
>>> unpack('hhi', '\x00\x01\x00\x02\x00\x00\x00\x03')
```

```
Nmap - additional flags
--send-eth/--send-ip: send using raw ethernet frames or ip packets
--script "http-": Scripts with wildcard
--source-port <portnumber> Spoof source port number
--state-every <time> Update state during runtime at set <interval>
--open-reason: Return only open services, and reason why
--data-length 0: 0 for no random or protocol-specific payload
--randomize-hosts: Default is sequential hosts
--data hex string: Append custom binary data to sent packets
e.g. --initial-rtt-timeout 250ms --max-retries 2 --host-timeout 15m
```

```
SSH Tunneling
- execute from source: ssh -f -N -i <source> <dest> <port> user@relay
- execute from dest: ssh -f -N -R <source> <port> <dest> <port> user@relay
```

# Malware Development & Analysis

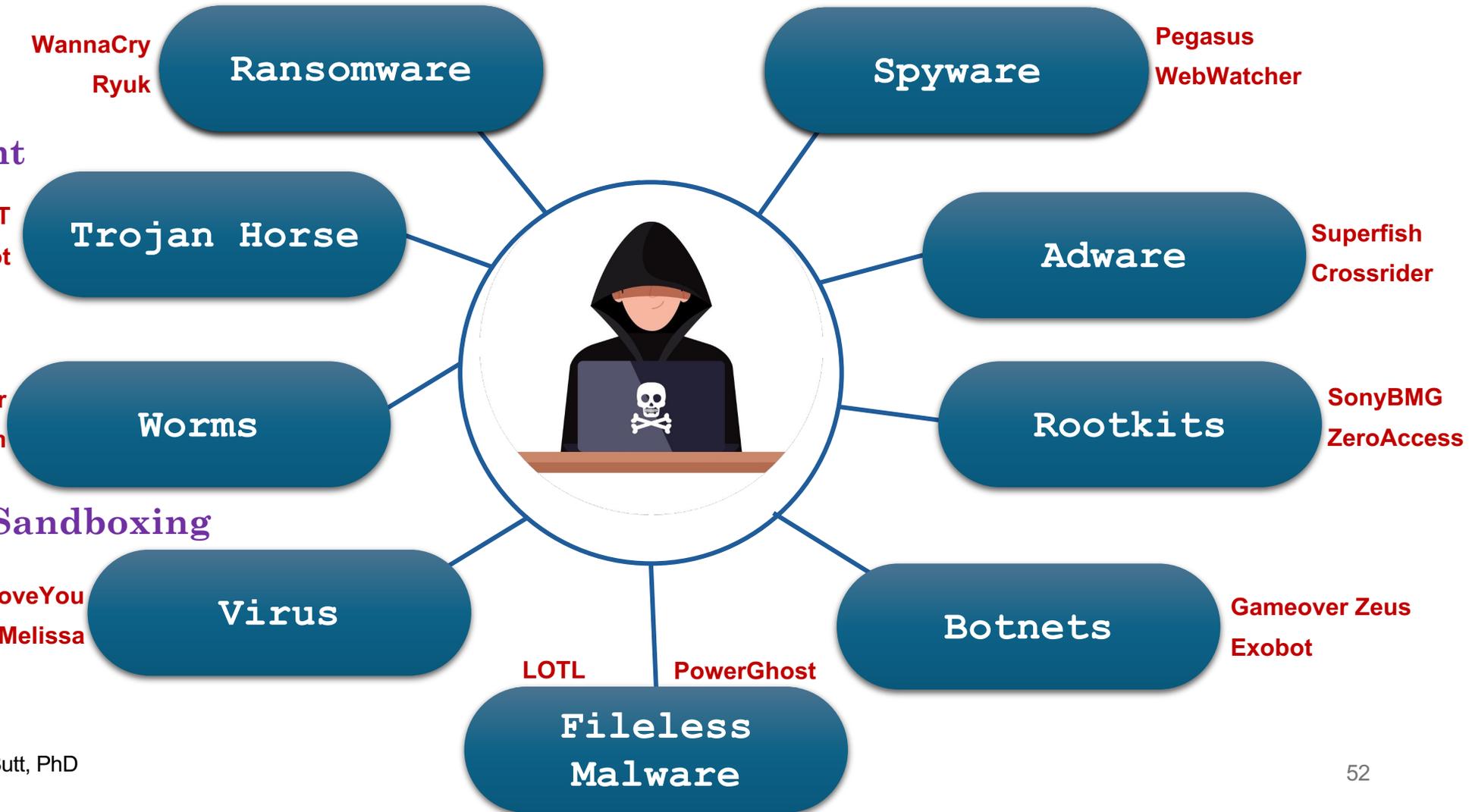
Malware development refers to the process of creating malicious software designed to infiltrate, damage or steal data from, or disrupt the operation of computers, NW or devices

## Malware Development

- Msfvenom
- Veil-Framework
- Unicorn

## Malware Analysis & Sandboxing

- FLARE\_VM
- Cuckoo Sandbox
- YARA
- REMnux





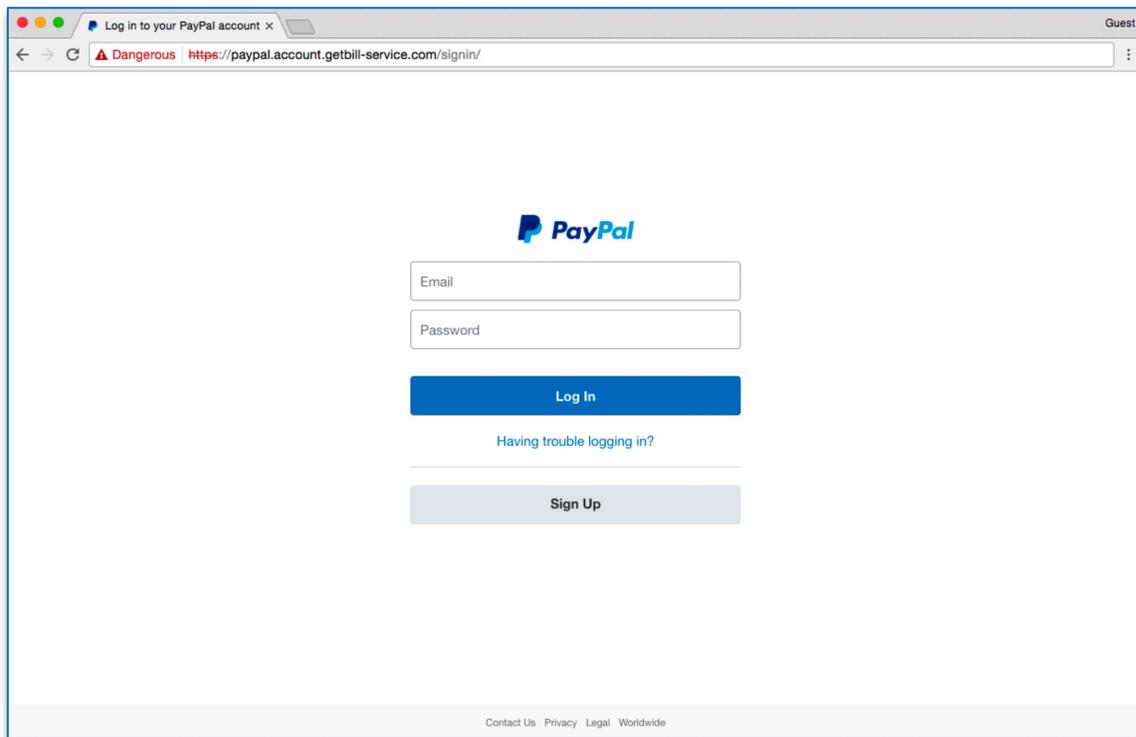
# AI meets Cybersecurity

## The New Battlefield

# Applications of GPT4 Vision in Cybersecurity



# Prompt: Check if the website is genuine

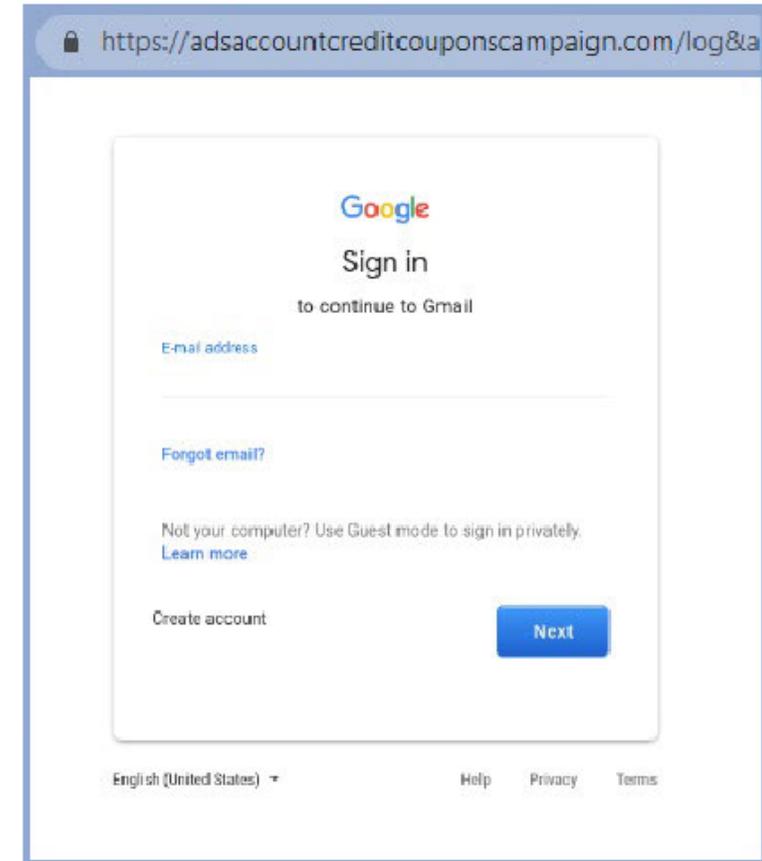
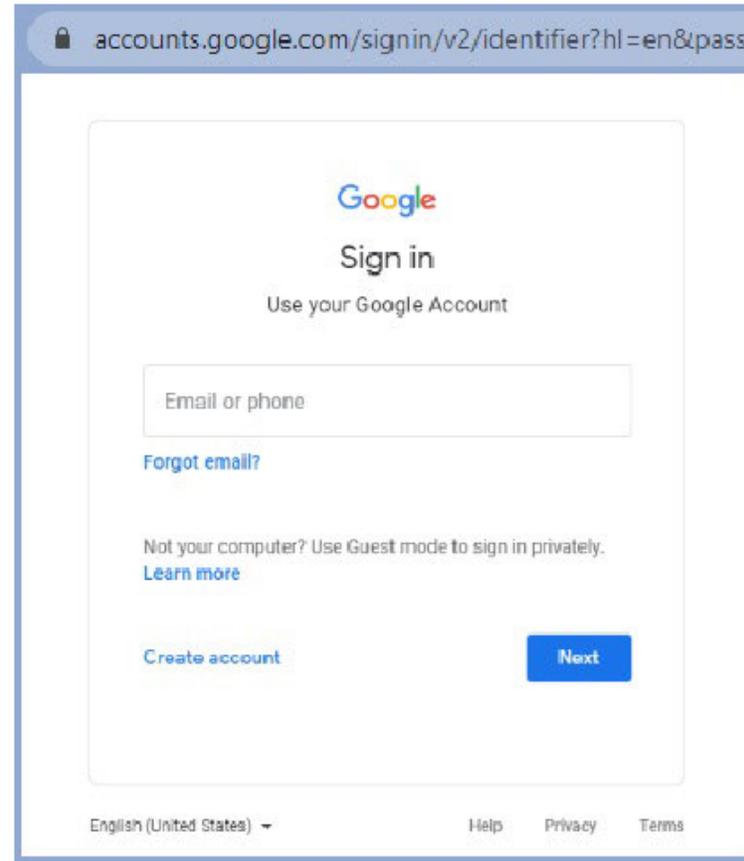
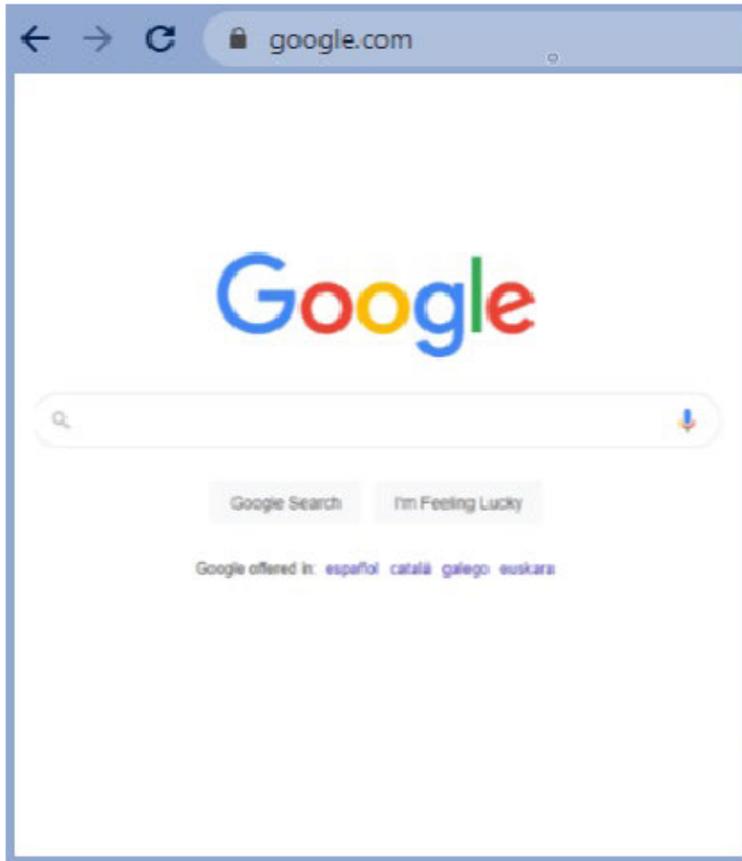


Based on the image you've provided, there are several indicators that suggest the website may not be genuine:

1. **Domain Name:** The URL "<https://paypal.account.getbill-service.com/signin/>" contains extra words that are not associated with the official PayPal domain. PayPal's official domain is "paypal.com", and they would not use secondary domains like "getbill-service.com" for their main services.
2. **Dangerous Warning:** The browser has flagged the website as "Dangerous," which means the website might be harmful or has been reported for phishing activities.
3. **HTTPS:** While the site does use HTTPS, indicating a secure connection, this alone does not guarantee the authenticity of a website. Many phishing sites also use HTTPS to appear more legitimate.

Based on these factors, it's highly advisable not to enter any personal or financial information on this website. Always double-check the domain and ensure you're on the official website before entering any sensitive information. If you suspect a phishing attempt, it's a good practice to report it to the respective company (in this case, PayPal).

# Prompt: Evaluate the three Screenshots



## 1. google.com:

- **URL:** The URL is "google.com," which is the legitimate domain for Google.
- **Appearance:** The page looks like the standard Google homepage.
- **Verdict:** This appears to be a genuine page.

## 2. accounts.google.com:

- **URL:** The URL is "accounts.google.com," which is the legitimate domain for Google Account services.
- **Appearance:** The design resembles the genuine Google sign-in page.
- **Verdict:** This appears to be a genuine page.

## 3. adsaccountcreditcouponscampaign.com:

- **URL:** The URL "adsaccountcreditcouponscampaign.com" is not a recognized Google domain. Google's official services, including Gmail, use the google.com domain or subdomains thereof.
- **Appearance:** While the design looks like a Google sign-in page, the domain name gives away that it's not legitimate.
- **Verdict:** This appears to be a fake page.

# Prompt: Which security policy is being violated?



The image depicts a violation of several security policies related to unauthorized access, physical security, and potential data exfiltration. Specifically:

1. **Unauthorized Access:** The individual in the foreground, dressed in a disguise, appears to be attempting to infiltrate a computer system without permission. This is a direct breach of policies that restrict system access to authorized personnel only.
2. **Physical Security:** The individual is physically present in a space where they likely don't belong, based on their attire and behavior. Physical security protocols are meant to prevent unauthorized individuals from accessing secure areas.
3. **Potential Data Exfiltration:** The individual is holding a USB device, suggesting they might be attempting to extract or install data maliciously. Many organizations have strict policies against the use of external storage devices without proper authorization to prevent data theft and malware introduction.
4. **Awareness and Reporting:** The other individuals in the image appear unaware of the security breach happening right in front of them. This may highlight a lack of training or awareness about security threats and the importance of reporting suspicious activities.

In summary, the scene shows a clear violation of security protocols related to unauthorized physical and digital access, as well as potential data theft.

# Prompt: Is there any issue in this image?

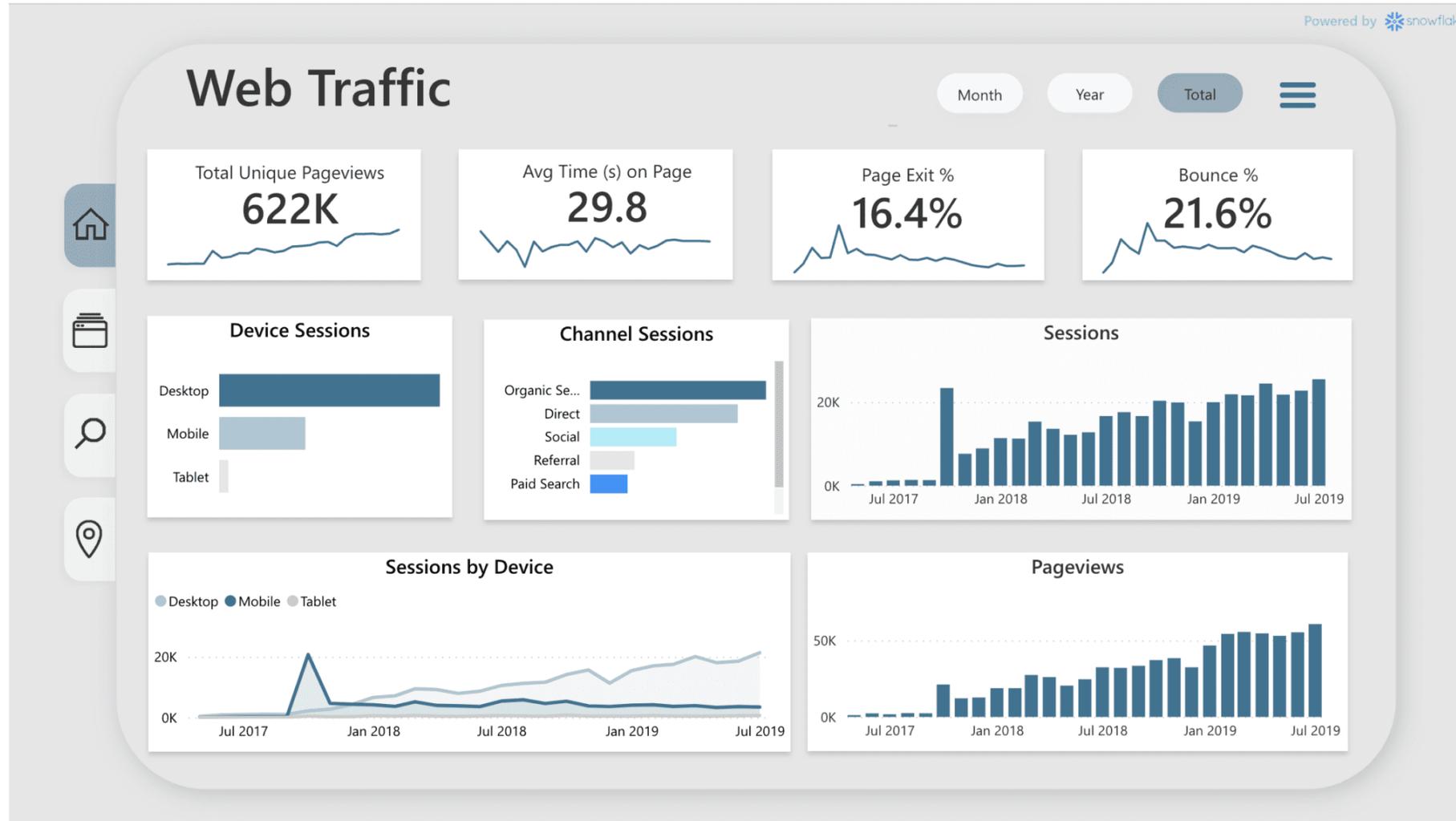


Yes, there appears to be a significant security issue in this image. The individual is taking a photo of a computer screen displaying the word "CONFIDENTIAL." This behavior can be a violation of the following security protocols:

1. **Data Leakage:** Taking photographs of sensitive or confidential information can lead to unintentional or intentional data leakage, which can be detrimental for an organization.
2. **Breach of Confidentiality:** If the information on the screen is indeed confidential, capturing or sharing it without proper authorization breaches confidentiality agreements.
3. **Violation of Electronic Use and Security Policies:** Many organizations have policies prohibiting the use of personal devices to capture or transmit workplace information. This ensures data integrity, confidentiality, and compliance with data protection regulations.

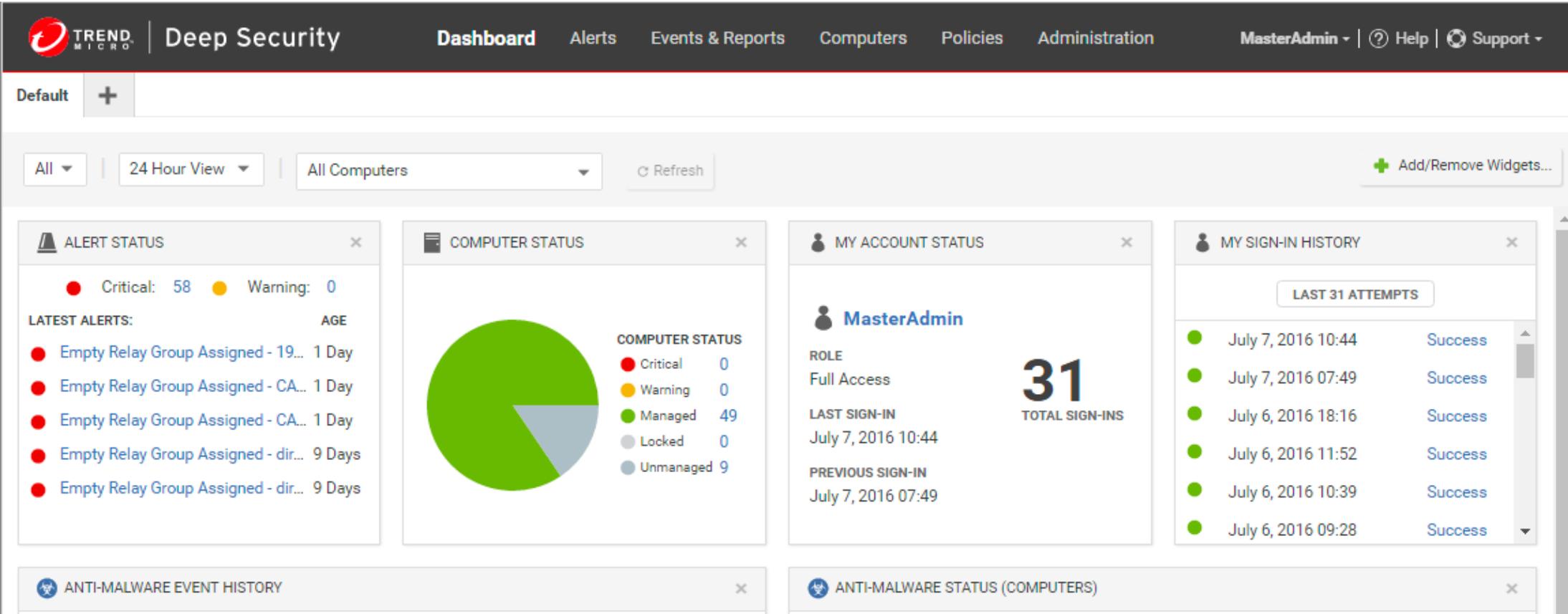
In a secure workplace environment, such actions should be immediately reported to ensure the confidentiality and security of sensitive information.

# Prompt: Which device has maximum number of sessions.



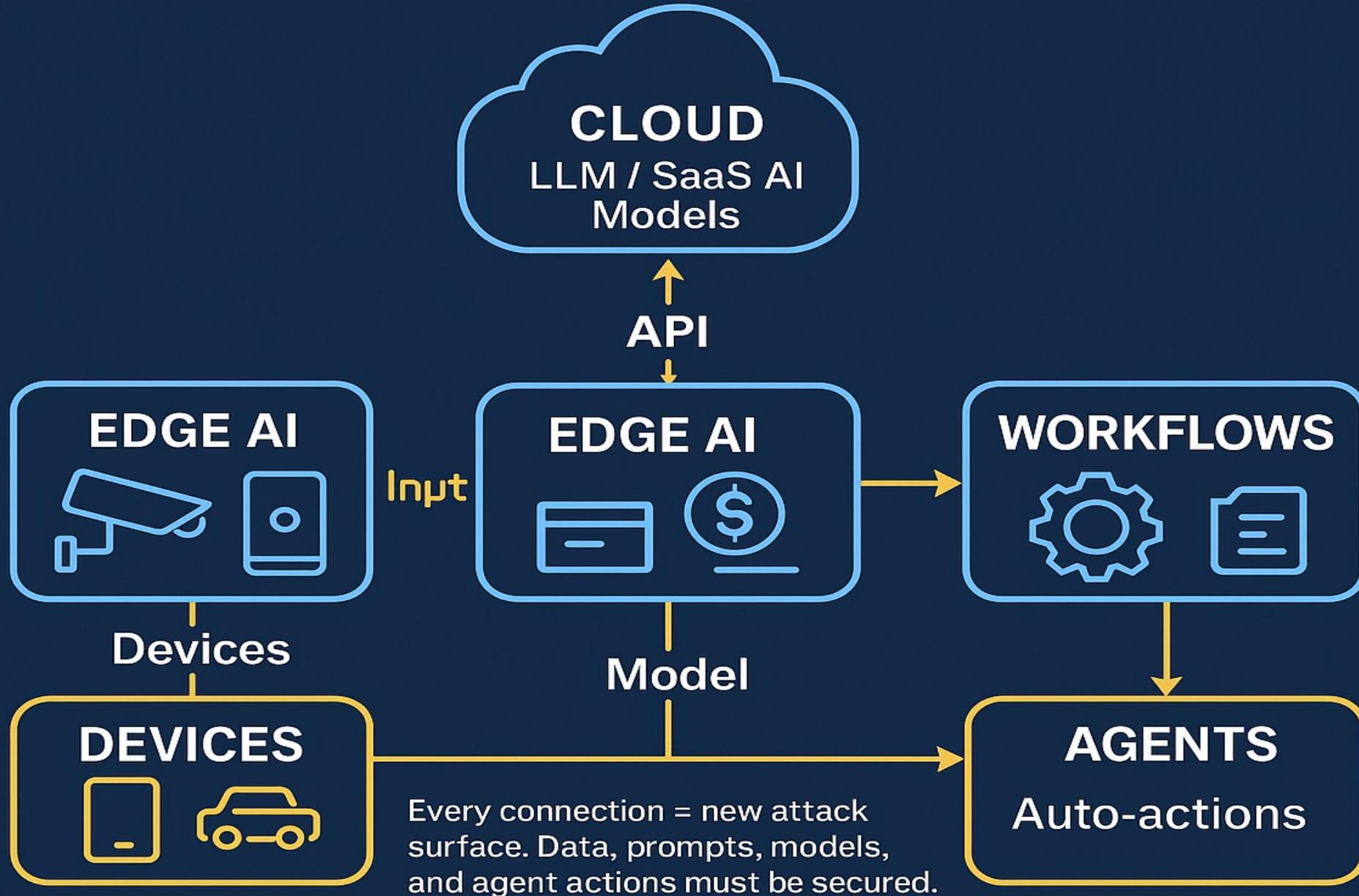
The device with the maximum number of sessions, as displayed in the "Device Sessions" section of the provided image, is "Desktop."

# Prompt: Briefly highlight the issues in the Dashboard



1. 58 critical alerts: "Empty Relay Group Assigned".
2. 9 unmanaged computers.
3. Anti-malware events peak around 09:00.
4. Top infected computer has 16 issues. 60

# AI Expands the Attack Surface



# How AI Expand the Attack Surface?

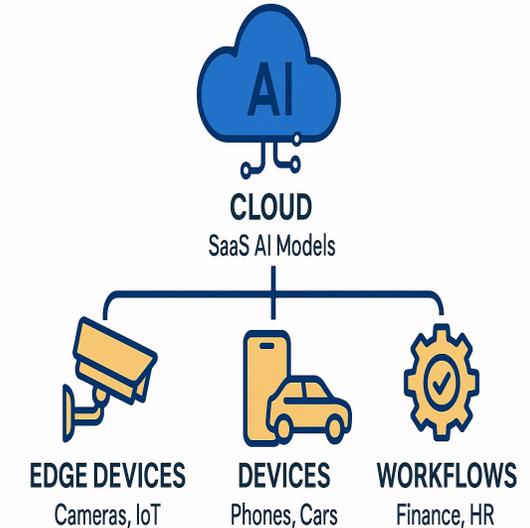


## AI is now embedded into cloud, edge devices, and workflows

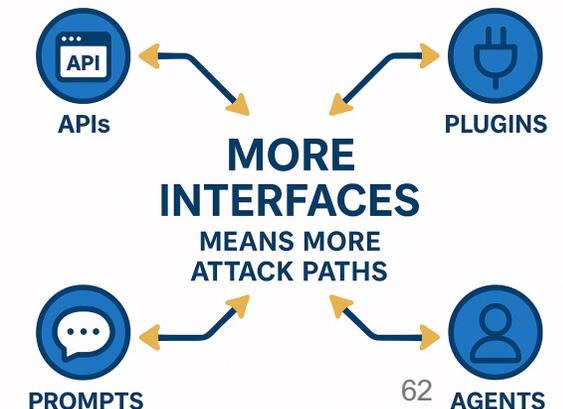
AI is no longer only in big data centers. It runs **everywhere**—from ChatGPT in the cloud to AI models embedded in **phones, cameras, cars, routers, medical devices**, etc.

### Example:

- **Cloud:** MS Copilot or Google Gemini inside enterprise environments.
- **Edge Devices:** Smart CCTV that uses AI for face detection locally.
- **Consumer Devices:** Phones doing on-device speech recognition.
- **Workflows:** Companies using AI assistants to generate documents, create invoices, approve financial requests.



**Why this matters for security:** Each environment becomes a potential attack surface. If the model is compromised in *any one place*, the decision-making pipeline is compromised.



# AI Increase Scale, Automation & Believability



AI makes it possible to:

- Generate thousands of phishing emails, each personalized to the victim.
- Clone voices and faces to impersonate real people.
- Automatically write malware variants that evade detection.

*Attacks now look real, adapt fast, and happen at machine speed*



# Use of Agent-Driven Offensive Tools



Hackers no longer run every step manually, rather use **AI agents** that can:

- Scan networks
- Find weak passwords
- Craft payloads
- Try exploits
- Adjust and retry automatically

## Example:

An AI agent finds a vulnerable server →  
Generates exploit code → Deploys payload →  
Moves laterally

*This turns hacking into continuous  
autonomous operations*



# Attackers Automate Attacks using AI



Most security teams still:

- Manually analyze alerts
- Manually investigate incidents

Meanwhile attackers:

- Automate everything they can

**Result** → Attackers scale faster than defenders

*Organizations rely on humans to stop machine-speed attacks*

*Attackers rely on machines to overwhelm human-speed defenses*



# AI-Driven Self-Evolving Malware



*AI-driven polymorphic malware that continuously mutates to bypass defenses*

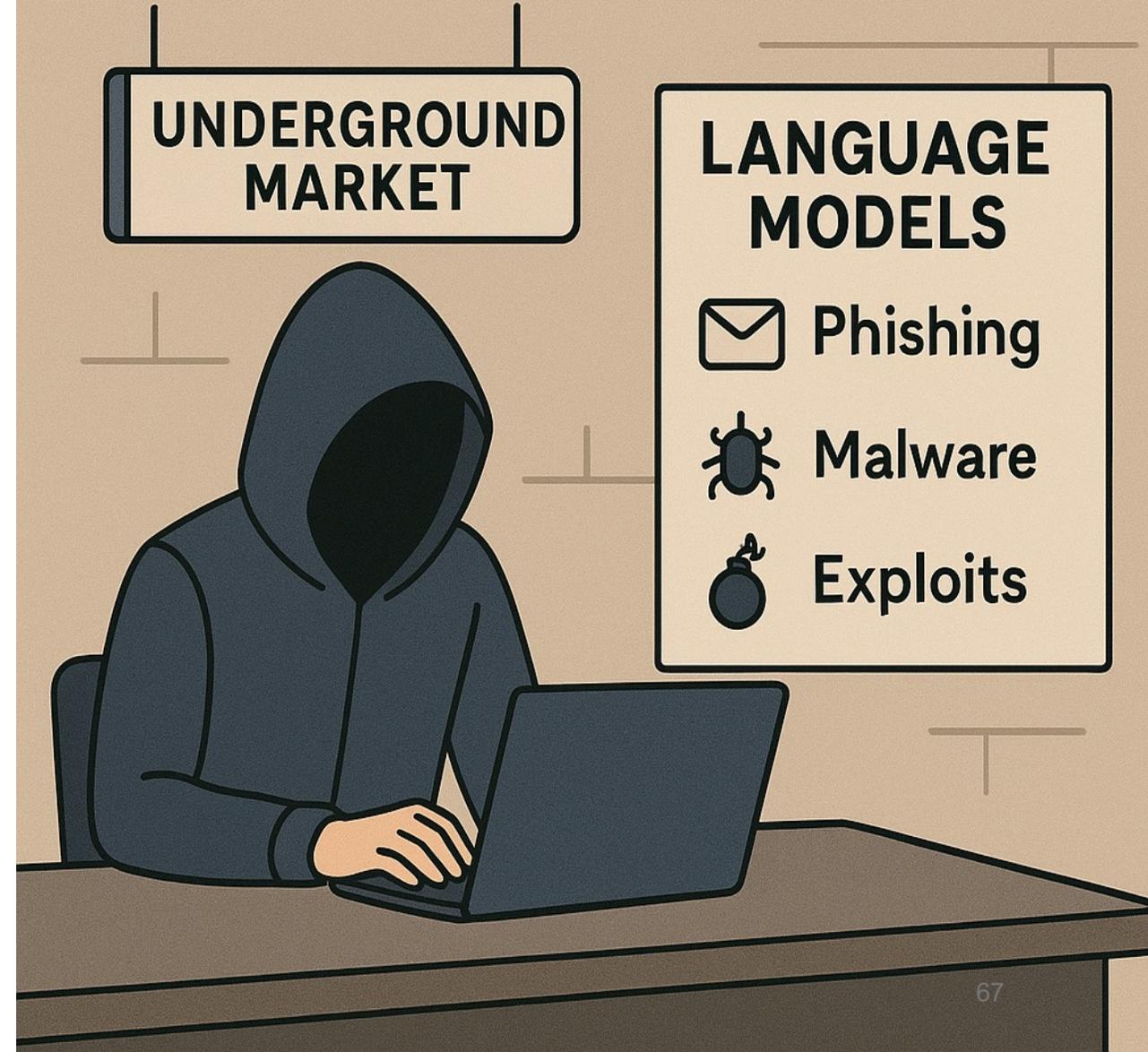
- Offensive operators leverage AI-capable, self-evolving malware to automate payload generation, evasive C2 behaviors and large-scale exploitation.
- Defenders detect and analyze polymorphic/obfuscated threats by combining static pattern engines and hunting rules using YARA, and automated dynamic analysis/sandboxing using Cuckoo Sandbox



# WormGPT / FraudGPT

- WormGPT and FraudGPT are malicious LLMs developed and sold on dark web forums to facilitate cybercrime by removing the ethical guardrails found in legitimate AI chatbots like ChatGPT.
- Designed to launch phishing attacks, generate malware, and exploits

## Case Study: WormGPT / FraudGPT



# Deepfake Executive Fraud

- Case: \$25M stolen/transferred via deepfake CFO call in Hong Kong (2024)
- Victim received a video call appearing to include leadership
- Attackers used generative audio + facial reenactment models

## Case Study: Deepfake Executive Fraud



# Hyper-Realistic Social Engineering



*AI-crafted phishing and Deepfake voice/video impersonation of executives*

<https://www.youtube.com/watch?v=cQ54GDm1eL0>

**Deepfake AI:** Deepfake AI poses significant challenges to cybersecurity, as it can be used to create highly convincing fake audio, video, and text content that can deceive individuals and manipulate public opinion

- Social Engineering Attacks
- Phishing and Fraud
- Reputation Damage
- Misinformation Campaigns
- Authentication and Trust Issues



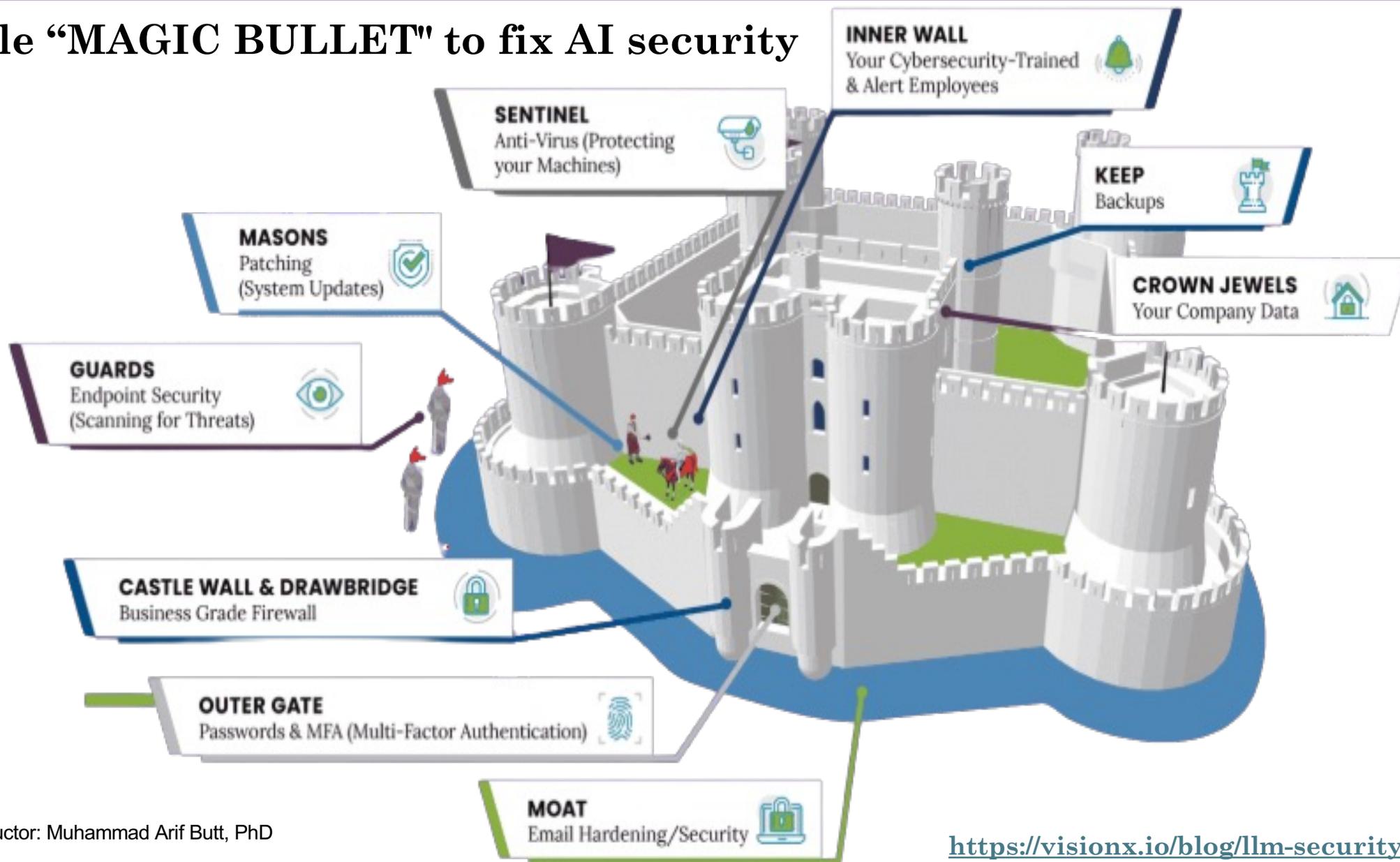
According to recent research from global technology company Entrust, incidents involving deepfake phishing and fraud have skyrocketed by 3,000% since 2022, with a deepfake attempt occurring every five minutes in 2024.



# AI-Powered Cybersecurity

# Designing Defense-in-Depth with AI

No single “MAGIC BULLET” to fix AI security



# Traditional SOC Workflow



## SOC Team (Security Guards)

- Detect
- Investigate
- Confirm Threat
- Handoff to IR

## IR Team (SWAT Team)

- Contain
- Eradicate
- Recover
- Provides feedback to SOC, which updates detection rules

[Link to Example Workflow](#)

- ✓ SOC without IR = Detecting threats but unable to fully respond
- ✓ IR without SOC = No early warning system, responding only after major damage

**Together** = A defense-in-depth strategy that gets stronger with each incident

Instructor: Muhammad Arif Butt, PhD

## What is a SOC?



## Security Operations Center (SOC)



### People

Cybersecurity experts and analysts



### Processes

Continuous monitoring and incident response



### Technology

Tools and systems to monitor and detect threats



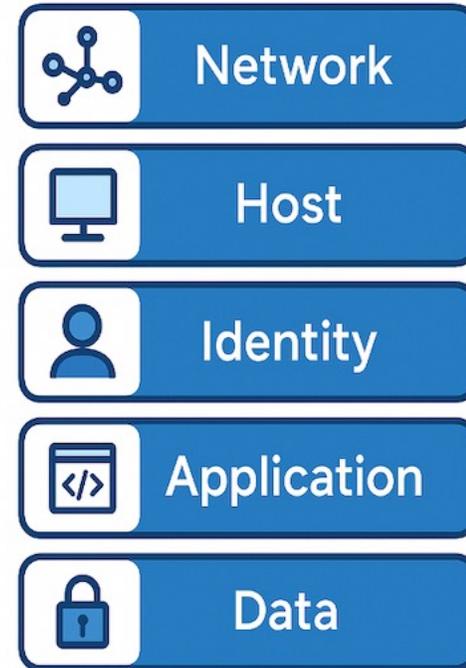
**The command center of cybersecurity.**

# Designing Defense-in-Depth with AI



Think in layers and AI plugs into each layer:

- **Network:** NW security monitoring and detection systems (Zuricata, Zeek)
- **Host:** EDR with ML detection (Defender, SentinelOne)
- **Identity:** Adaptive access control
- **App:** AI WAF (Cloudflare Bot Management, ModSecurity rulesets)
- **Data:** DLP with ML content classification



# AI-Based Threat Hunting Workflow



## *AI-based threat hunting works like a 24/7 digital security team*

- AI-based threat hunting uses AI models and behavioral analytics to proactively search for indicators of compromise and unusual patterns across an organization's digital environment.
- Unlike traditional reactive security measures, AI-powered threat hunting enables organizations to detect threats at machine speed, using autonomous systems to gain access, analyze patterns, and deploy counter-measures:
  1. **EDR/XDR** (Endpoint/Extended Detection & Response) monitor and analyze endpoint activity to detect and respond to threats at the device level and may be NW/cloud infrastructure.
  2. **SIEM** (Security Information and Event Management) solutions collect and analyze data from various sources (NW devices, servers and endpoints), and aggregate into one centralized platform.
  3. **UEBA** (User and Entity Behavior Analytics) powered by ML, flags/spots weird behavior (e.g., an admin account logged in from a foreign VPN at 3 a.m.)
  4. **SOAR** (Security Orchestration, Automation, and Response) solutions jumps into action, w/o waiting for a human. It automatically isolates the suspicious device from the network and forces a password reset for that account, containing the treat within seconds.
  5. **Threat intel** system learns from an incident and updates the firewall rule and EDR signatures across the organization so the same attack can never happen again.

# EDR: Endpoint Detection & Response

*EDR monitor and analyze endpoint activity to detect and respond to threats at the device level*

- **EDR agents installed on endpoints (laptops, servers, mobile devices) continuously:**
  - Monitor process execution and behavior
  - Track file system changes
  - Capture network connections
  - Record registry modifications (Windows)
  - Detect in-memory attacks and fileless malware
- **Leading EDR/XDR Solutions:**
  - CrowdStrike Falcon
  - SentinelOne Singularity
  - Microsoft Sentinel with Logic App
  - Microsoft Defender for Endpoint
  - Palo Alto Networks Cortex XDR

# SIEM:- Data Collection & Aggregation

*SIEM systems aggregate log data from multiple sources into one centralized platform, watching everything and ingesting logs from across the organization*

- **Security Information & Event Management** platforms continuously collect security events from:
  - Endpoints (workstations, servers, mobile devices)
  - Network devices (firewalls, routers, switches)
  - Applications (web servers, databases, SaaS platforms)
  - Cloud infrastructure (AWS CloudTrail, Azure Monitor, GCP Cloud Logging)
- **Leading SIEM Tools:**
  - Microsoft Sentinel
  - Splunk Enterprise Security
  - IBM QRadar
  - Elastic (ELK Stack)
  - Wazuh SIEM

# UEBA: AI-Powered Anomaly Detection



*UEBA technologies analyze the behavior of users and entities (including terminals, applications, networks, servers, connected objects) to identify potential threats*

- **User & Entity Behavior Analytics** solutions analyze behavior using static rules and machine learning algorithms to detect changes in behavior like suspicious operations, access to critical unused applications during holidays, unusual data transfers. Some example scenarios are:
  - An admin account logs in from a foreign VPN at 3 AM (unusual location + unusual time)
  - A user suddenly downloads 100 GiB of data when their baseline is 50 MiB daily
  - Privilege escalation attempts outside normal business hours
  - Lateral movement patterns indicating compromised credentials
- **Leading UEBA Solutions:**
  - Microsoft Sentinel UEBA
  - Securonix UEBA
  - IBM Qradar UEBA
  - Splunk UBA

# SOAR – Security Orchestration, Automation, & Response

*SOAR automatically isolates suspicious devices from the network and forces password resets for compromised accounts, enabling security teams to proactively detect and react to complex security events while automatically interacting with IT and security systems to mitigate incidents*

- **For compromised credentials:** Disable account, force password reset, revoke active sessions, notify user and security team
- **For malware detection:** Isolate endpoint, quarantine file, block C2 communications, trigger forensic data collection
- **For data exfiltration:** Block network egress, suspend user account, alert DLP systems, preserve evidence
- **Leading SOAR Platforms:**
  - Palo Alto Cortex XSOAR
  - IBM Resilient
  - Shuffle SOAR
  - StackStorm

# Advanced Analytics and Threat Intel

*Enrich detection with external threat intelligence and AI-powered analysis*

- **GenAI** can streamline the process by automatically summarizing anomalous activities for each high-risk entity, providing concise yet comprehensive overviews alongside risk scores.
  - GenAI provides contextual information for cybersecurity scenarios and automates common tasks in the analyst's workflow
  - Automated threat report generation
  - Natural language querying of security data
  - Predictive threat modeling
- **AI-Powered Cybersecurity Products**
  - [Microsoft Security Copilot](#)
  - [CrowdStrike Charlotte AI](#)
  - [Securonix Agentic AI](#)
  - [Splunk Enterprise Security](#)
  - [Cortex](#)
  - [IBM Qradar](#)
  - [CheckPoint Infinity](#)
  - [CrowdStrike](#)

# Happy Hacking

## To Do:

Go through Handout # 1.2 at your own and set-up your own Virtual Hacking Lab for all the class activities and assignments using either Virtual Machines or Docker Containers or both 😊