Intelligent Cyber-Security
Introduction to Cyber-Security

Dr. Alexandru ARCHIP

Gheorghe Asachi Technical University of Iasi

Artificial Intelligence (MSc., second year) – 2025 – 2026

alexandru.archip@academic.tuiasi.ro

In brief...

1. Brief introduction
   - Cyber-Security
   - Weakness-enabling factors
2. Threat and Defense Models
   - Cyber Kill Chain

Outline

1. Brief introduction
   - Cyber-Security
   - Weakness-enabling factors

2. Threat and Defense Models
   - Cyber Kill Chain

Cyber-Security [1, 2]

A tentative definition [2]

Cyber-Security refers to a set of principles and practices designed to safeguard your IT assets and online information against cyber-attacks/threats.

- a wide term, encompasing both human and non-human factors;
- could be narrowed down to it's core function: protection of devices and services [2]
- increasingly important nowadays: digital technologies have become an integral part of our lives (public and private alike).

Cyber-Threats & Cyber-Attacks

Cyber-Threat

A *cyber-threat* is a potential danger or malicious intent that could compromise various digital systems and/or data.

Cyber-Attack

A *cyber-attack* is a specific, intentional action meant to exploit different vulnerabilities and weaknesses to compromise digital systems and/or data.
A *cyber-attack* is carreid out by a *threat actor*.

Weaknesses & Vulnerabilities

Weakness [3]

A *weakness* is a condition in a software, firmware, hardware, or service component that, under certain circumstances, could contribute to the introduction of vulnerabilities.

Vulnerability

A *vulnerability* is a flaw in a software, firmware, hardware, or service component resulting from a weakness that can be exploited, causing a negative impact to the confidentiality, integrity, or availability of an impacted component or components [3].

A *vulnerability* is an instance of one or more weaknesses in a Product that can be exploited, causing a negative impact to confidentiality, integrity, or availability; a set of conditions or behaviors that allows the violation of an explicit or implicit security policy [4].

Weaknesses & Vulnerabilities

Examples:

Weakness

CWE-120/CWE-121: buffer-overflow / improper bounds checking [3]

Vulnerability

CVE-2017-11882 is a memory-corruption bug in Microsoft Office's Equation Editor (EQNEDT32.exe). An attacker can craft an Office file (e.g., a malicious RTF/DOC file) that triggers a buffer overflow when the Equation Editor copies a font/name or other field into a too-small local buffer. Successful exploitation lets the attacker overwrite the saved frame pointer/return address and execute arbitrary code as the victim user [4].

Weaknesses & Vulnerabilities

Examples:

Weakness

CWE-89: Improper Neutralization of Special Elements used in an SQL Command [3]

Vulnerability

CVE-2024-24213 – Supabase PostgreSQL v15.1: SQL injection in /pg_meta/default/query. Summary: The product's /pg_meta/default/query component allowed specially crafted input that could be interpreted as SQL, enabling an attacker to inject arbitrary SQL (data exfiltration, unauthorized queries, or other database impact). The CVE entry describes the SQL-injection classification and affected versions; vendor fixes/patches are listed on the CVE record 4.

Weakness-enabling factors

Non-human / Technical factors

- Flaws in hardware design: faulty hardware design (e.g., weak isolation, poor material choices)
- Flaws in software design: OS or application flaws (e.g., insufficient process memory protection)
- Flaws in protocol specifications: incomplete mitigation for large numbers of simultaneous requests; ambiguous error handling
- Misconfiguration and insecure defaults
- Legacy components and unmaintained dependencies
- Insufficient runtime protections (e.g., missing ASLR/DEP, weak sandboxing)

Weakness-enabling factors

Human factors

- Lack of security awareness / training
- Social engineering (phishing, pretexting)
- Poor operational practices
  - Weak / reused passwords, shared accounts
  - Inadequate change management
- Delayed patching and poor vulnerability management
- Insider mistakes or malicious insiders
- Incomplete or unenforced policies and procedures

Weakness-enabling factors

Examples of non-human / technical factors:

HTTP/1.1: RFC 9110 & 9112 [5, 6]

- HTTP is a *stateless protocol* – processes only complete requests;
- specifications deal with slow / incomplete requests, but *do not enforce timeout duration*;
- *weakness*: slow / incomplete requests might render the server inaccessible (Slow DOS / Slowloris).

IP Protocol suite [7, 8, 9]

- TCP/IP communications require *source* and *destination* addresses (common sense):
  - IPv4: source address is an *ordinary* 32-bit header field;
  - IPv6: source address is an 128-bit header field;
- *weakness*: no authentication / source verification steps are provided by the specs; any 32-bit or 128-bit value cat be set, and the protocol does not verify it.

Looking Ahead: Directions and Reflections from the Past

Cyber-security is about **data and information**.
Cyber-security is about **humans**: identifying patterns, spotting loopholes, and correcting them.

Bruce Schneier on cyber-security in [10]

Security is a process, not a product.

Bruce Schneier on AI as hackers in [11]

When AIs start hacking, everything will change. They won't be constrained in the same ways, or have the same limits, as people. They'll change hacking's speed, scale, and scope, at rates and magnitudes we're not ready for.

- **Reflections from the past**: Learn from historical weaknesses and vulnerabilities, human errors, and system design flaws.
- **Directions ahead**: Anticipate emerging threats, consider human & AI factors, and design proactive, adaptive security processes.

Outline

1. Brief introduction
   - Cyber-Security
   - Weakness-enabling factors

2. Threat and Defense Models
   - Cyber Kill Chain

Threat Model

Definition (taken from [12])

Threat modeling is a structured process used to identify, analyze, and prioritize potential threats to a system. It involves understanding the system's architecture, identifying possible attack vectors, and determining countermeasures to mitigate risks. This proactive approach helps in designing secure systems by anticipating and addressing security issues early in the development lifecycle.

Cyber Kill Chain

Characteristics

Type threat model / attack lifecycle framework

Purpose describe the sequence of steps an attacker performs to compromise a target;

Use understand attacks, identify traces and detection points, design mitigations;

Developer Lockheed Martin

Cyber Kill Chain [13]

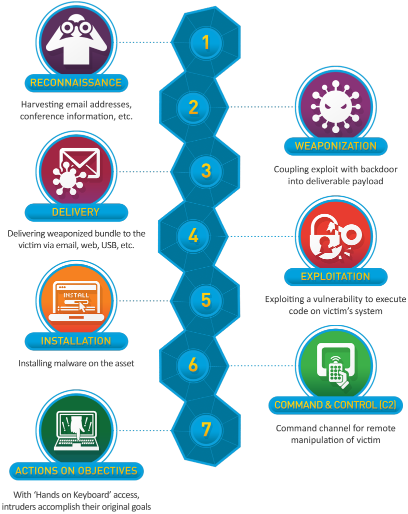
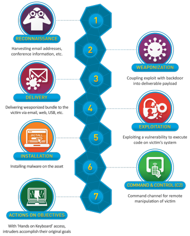

1
RECONNAISSANCE
Harvesting email addresses, conference information, etc.
2
WEAPONIZATION
Coupling exploit with backdoor into deliverable payload
3
DELIVERY
Delivering weaponized bundle to the victim via email, web, USB, etc.
4
EXPLOITATION
Exploiting a vulnerability to execute code on victim's system
5
INSTALL
INSTALLATION
Installing malware on the asset
6
COMMAND & CONTROL (C2)
Command channel for remote manipulation of victim
7
ACTIONS ON OBJECTIVES
With 'Hands on Keyboard' access, intruders accomplish their original goals

Figure 1: *Cyber Kill Chain (taken from [13])*

Cyber Kill Chain [13]


1
RECONNAISSANCE
Harvesting email addresses, conference information, etc.
2
WEAPONIZATION
Coupling exploit with backdoor into deliverable payload
3
DELIVERY
Delivering weaponized bundle to the victim via email, web, USB, etc.
4
EXPLOITATION
Exploiting a vulnerability to execute code on victim's system
5
INSTALLATION
Installing malware on the asset
6
COMMAND & CONTROL (C2)
Command channel for remote manipulation of victim
7
ACTIONS ON OBJECTIVES
With 'Hands on Keyboard' access, intruders accomplish their original goals

1. Reconnaissance

Phase type partially observable, depending on target

Adversary gain data on the target and identify potential weaknesses

focus on both human and non-human factors

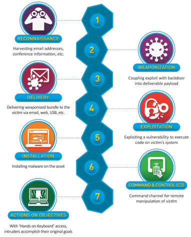
examples: harvest accounts, social media data, identify *internet-facing* services

Defender monitor and identify reconnaissance patterns, gain insights on potential attacks

should include employee monitoring, social media data (*it may sound a bit harsh/unethical, but it may be achieved elegantly*)

examples: monitor app usage, web/internet access patterns

Cyber Kill Chain [13]


1
RECONNAISSANCE
Harvesting email addresses, conference information, etc.
2
WEAPONIZATION
Coupling exploit with backdoor into deliverable payload
3
DELIVERY
Delivering weaponized bundle to the victim via email, web, USB, etc.
4
EXPLOITATION
Exploiting a vulnerability to execute code on victim's system
5
INSTALL
INSTALLATION
Installing malware on the asset
6
COMMAND & CONTROL (C2)
Command channel for remote manipulation of victim
7
ACTIONS ON OBJECTIVES
With 'Hands on Keyboard' access, intruders accomplish their original goals

2. Weaponization

Phase type opaque/unobservable

Adversary prepare and stage attacks and delivery methods

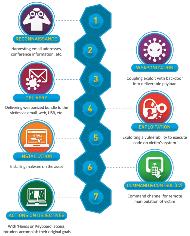
increasingly complex through adoption of AI based tools

examples: stage a scam/phishing campaign, develop new/enhance existing malware, prepare "decoys" to cover actual payload

Defender analyse know threat intelligence data, and monitor threat intelligence suppliers data for updates

examples: analyse known malwares, identify code obfuscation techniques, strengthen logging

Cyber Kill Chain [13]


1
RECONNAISSANCE
Harvesting email addresses, conference information, etc.
2
WEAPONIZATION
Coupling exploit with backdoor into deliverable payload
3
DELIVERY
Delivering weaponized bundle to the victim via email, web, USB, etc.
4
EXPLOITATION
Exploiting a vulnerability to execute code on victim's system
5
INSTALL
INSTALLATION
Installing malware on the asset
6
COMMAND & CONTROL (C2)
Command channel for remote manipulation of victim
7
ACTIONS ON OBJECTIVES
With 'Hands on Keyboard' access, intruders accomplish their original goals

3. Delivery

Phase type observable

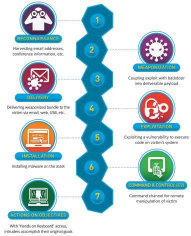
Adversary (as the phase name implies) the malicious payload is actually delivered (directly or indirectly)

examples: perform the web attack, deliver the scam/phishing emails, activate the MitM modules

Defender analyse security applications (e.g., firewall data, antimalware apps) and determine whether payloads have been successfully delivered or not

examples: collect email and web logs, monitor new malicious payloads, perform "forensic" tasks on suspicious files

Cyber Kill Chain [13]


1
RECONNAISSANCE
Harvesting email addresses, conference information, etc.
2
WEAPONIZATION
Coupling exploit with backdoor into deliverable payload
3
DELIVERY
Delivering weaponized bundle to the victim via email, web, USB, etc.
4
EXPLOITATION
Exploiting a vulnerability to execute code on victim's system
5
INSTALLATION
Installing malware on the asset
6
COMMAND & CONTROL (C2)
Command channel for remote manipulation of victim
7
ACTIONS ON OBJECTIVES
With 'Hands on Keyboard' access, intruders accomplish their original goals

4. Exploitation

Phase type observable

Adversary harness vulnerabilities to gain access

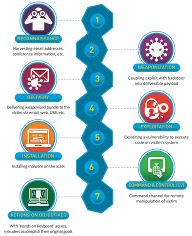
key-term: *zero-day* – a new vulnerability, previously unknown

examples: trigger server-based vulnerbilities, monitor victim triggered events (e.g., a victim clicks on a malicious link)

Defender harden security mechanisms, enhance user awareness, perform security scanning

examples: enforce firewall rules, reduce user privileges, apply apps security patching

Cyber Kill Chain [13]


1
RECONNAISSANCE
Harvesting email addresses, conference information, etc.
2
WEAPONIZATION
Coupling exploit with backdoor into deliverable payload
3
DELIVERY
Delivering weaponized bundle to the victim via email, web, USB, etc.
4
EXPLOITATION
Exploiting a vulnerability to execute code on victim's system
5
INSTALL
INSTALLATION
Installing malware on the asset
6
COMMAND & CONTROL (C2)
Command channel for remote manipulation of victim
7
ACTIONS ON OBJECTIVES
With 'Hands on Keyboard' access, intruders accomplish their original goals

5. Installation

Phase type observable

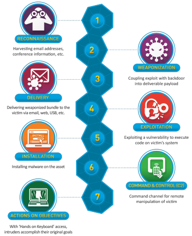
Adversary persist delivered payload, employ evasive techniques

examples: install webshells, add services to victim hosts, "time stop" malware files

Defender deploy different endpoint tools (Host-based Intrusion Prevention System – HIPS) and monitor host activity

examples: perform auditing tasks on installed apps, identify abnormal file creation activities, identify suspicious web traffic and C&C servers

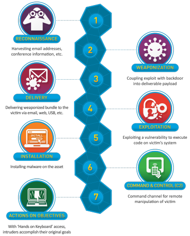
Cyber Kill Chain [13]


1
RECONNAISSANCE
Harvesting email addresses, conference information, etc.
2
WEAPONIZATION
Coupling exploit with backdoor into deliverable payload
3
DELIVERY
Delivering weaponized bundle to the victim via email, web, USB, etc.
4
EXPLOITATION
Exploiting a vulnerability to execute code on victim's system
5
INSTALL
INSTALLATION
Installing malware on the asset
6
COMMAND & CONTROL (C2)
Command channel for remote manipulation of victim
7
ACTIONS ON OBJECTIVES
With 'Hands on Keyboard' access, intruders accomplish their original goals

6. Command & Control (C2)

Phase type	observable
Adversary	establish a connection to the remote victim usually, attackers employ intermediate servers called *C2 servers* the infrastructure could be highly complex and include victim hardware examples of commonly used protocols: HTTP, DNS, email delivery protocols
Defender	discover the C2 infrasturcture block inbound/outbound traffic

Cyber Kill Chain [13]


1
RECONNAISSANCE
Harvesting email addresses, conference information, etc.
2
WEAPONIZATION
Coupling exploit with backdoor into deliverable payload
3
DELIVERY
Delivering weaponized bundle to the victim via email, web, USB, etc.
4
EXPLOITATION
Exploiting a vulnerability to execute code on victim's system
5
INSTALL
INSTALLATION
Installing malware on the asset
6
COMMAND & CONTROL (C2)
Command channel for remote manipulation of victim
7
ACTIONS ON OBJECTIVES
With 'Hands on Keyboard' access, intruders accomplish their original goals

7. Action on objectives

Phase type observable

Adversary perform the actual attack to achieve the final goal

examples: collect user credentials and data, perform internal reconnaissance, exfiltrate data, destroys systems

Defender assess damage and enact damage control/mitigation as soon as possible

if a malicious actor reaches this stage, defenders are quite powerless

data collected at this stage should be used to prevent further attempts

Remarks

- Data collected in stages *1 through 6* yield *Indicators of Attack – IoA*
- Data collected in the final stage (stage 7) yield *Indicators of Compromise – IoA*
- Main issues with security tools:
    - *reactive* rather than *proactive* thinking;
    - usage is often superficial/shallow;
    - lack of thorough analysis and insufficiently known apps.

Main security threats

AI & ML

Main security opportunities

AI & ML

Bibliography

1. Wiam Younes, ***Cyber Security 101***, Carnegie Mellon University, available online
2. ***, ***National Cyber Security Center***, FSSU Training day, available online
3. MITRE. (n.d.), ***Common Weakness Enumeration***, last accessed October 2, 2025, https://cwe.mitre.org/
4. CVE. (n.d.), ***Common Vulnerabilities and Exposures***, last accessed October 2, 2025, https://www.cve.org/
5. Fielding, R., Nottingham, M., & Reschke, J. (2022), ***HTTP Semantics (RFC 9110)***, https://www.rfc-editor.org/rfc/rfc9110.html
6. Fielding, R., Nottingham, M., & Reschke, J. (2022), ***HTTP/1.1 (RFC 9112)***, https://www.rfc-editor.org/rfc/rfc9112.html
7. Postel, J. (1981), ***Internet Protocol (RFC 791)***, https://www.rfc-editor.org/rfc/rfc791.html
8. Braden, R. (1989), ***Requirements for Internet Hosts — Communication Layers (RFC 1122)***, https://www.rfc-editor.org/rfc/rfc1122.html

Bibliography

9. Deering, S., & Hinden, R. (2017), ***Internet Protocol, Version 6 (IPv6) Specification (RFC 8200)***, https://www.rfc-editor.org/rfc/rfc8200.html
10. Schneier, B. (2000), ***Crypto-Gram Newsletter – May 15, 2000***, https://www.schneier.com/crypto-gram/archives/2000/0515.html
11. Schneier, B. (2021), ***When AIs Start Hacking***, https://www.schneier.com/blog/archives/2021/04/when-ais-start-hacking.html
12. OWASP. (n.d.), ***Threat Modeling. Open Web Application Security Project***, https://owasp.org/www-community/Threat_Modeling
13. Lockheed Martin. (n.d.), ***Cyber Kill Chain***, https://www.lockheedmartin.com/en-us/capabilities/cyber/cyber-kill-chain.html