Introduction to Ethical Hacking

Ethical Hacking

Alessandro Brighente

07/10/2024

Master Degree on Cybersecurity

Università degli Studi di Padova

SPRITZ Security & Privacy Research Group

What is Hacking?

- A hacker is an individual skilled in IT, with knowledge on bugs and exploits
- Hackers attack information systems to find vulnerabilities to gain advantage over the victim

(Desired) Effects

- Flaws in IT systems may include outdated software or misconfiguration
- The consequence of cyberattacks are different and depend on the target victim
- Effects include:
  - Gaining access to private information
  - Gain control of a machine/group of machines
  - Causing service unavailabilities

Famous Cyberattacks

- Wannacry (2017)

Famous Cyberattacks

SPRITZ
SECURITY & PRIVACY
RESEARCH GROUP

UNIVERSITÀ
DEGLI STUDI
DI PADOVA

- Wannacry (2017): ransomware that rapidly spread from large to small (local) networks
  - Four-day long knocked out more than 200,000 computers in 150 countries
  - Encrypted devices including medical equipment, and some factories were forced to stop production

- Darkhotel (2004): Spyware
  - Victims were prompted to install a seemingly legitimate update
  - Immediately got infected with the DarkHotel spyware,

Not Only IT

- Hacking also extends to other technologies
- When IT networks are connected with other networks (e.g., Operational Technology) the attack may exploit peculiarities of these last devices
- Effects on both the network and the sensors/actuators

Famous Cyberattacks (2)

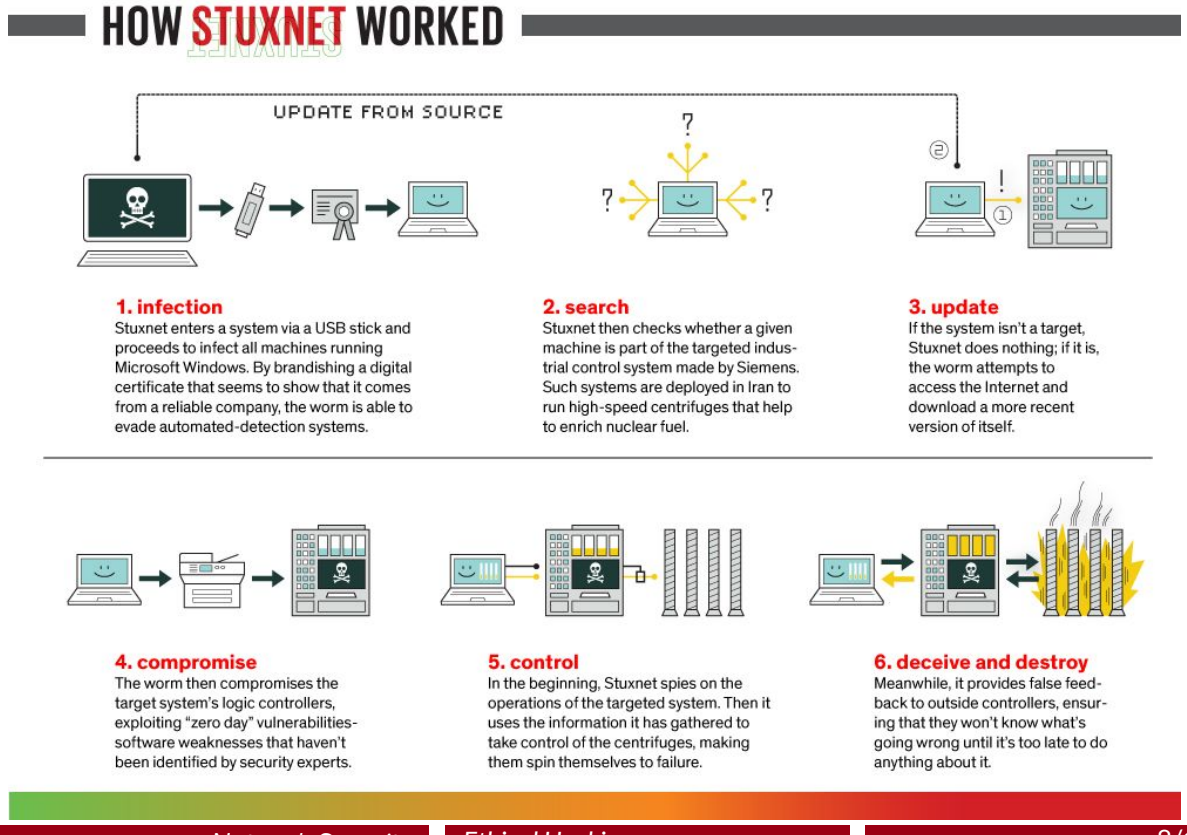
SPRITZ
Security & Privacy
Research Group

Università
degli Studi
di Padova

- Stuxnet (2010)
  - Target: industrial power plant
  - Disabled uranium-enrichment centrifuges in Iran
  - Set the rotational speed of the uranium-enrichment centrifuges too high, physically destroying them

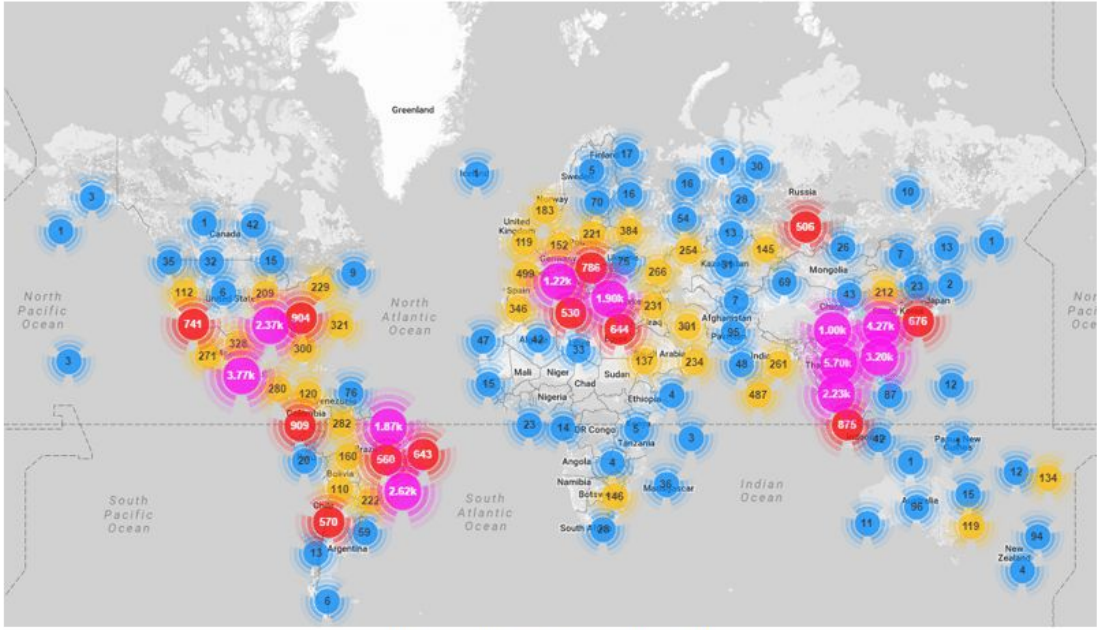
Famous Cyberattacks (2)

- Stuxnet (2010)



Famous Cyberattacks (3)

- Mirai (2016)
  - Botnet: a “robot network”
  - Targeted low capability devices such as IoT
  - Launched a Distributed Denial of Service affecting PayPal, Twitter, Netflix, Spotify, PlayStation online services



[Image courtesy of Incapsula.com]

What is Ethical Hacking

- Ethical hackers test systems to find and exploit vulnerabilities and weaknesses
- Basic difference with an hacker: they own the victim's permission
- Why do we need ethical hackers?

Need for Ethical Hackers

SPRITZ
SECURITY & PRIVACY
RESEARCH GROUP

UNIVERSITÀ
DEGLI STUDI
DI PADOVA

- Ethical hackers are able to identify the vulnerabilities of a system before a malicious user does
- By reporting the vulnerability to the client, they allow for immediate action and attack avoidance
- Of course, ethical hackers have some boundaries…

Classification

- We can generally distinguish between
  - **Black hat hackers:** launching cyberattacks for monetary reward or hacktivism
  - **White hat hackers:** doing hacking to provide security solutions
  - **Gray hat hackers:** not necessarily malicious, but don't have the victim's permission

What it is NOT hacking

SPRITZ
SECURITY & PRIVACY
RESEARCH GROUP

UNIVERSITÀ
DEGLI STUDI
DI PADOVA

[Best Scene Ever]

When does it come to play

- Companies can make a great use of ethical hackers
- They are usually grouped in two teams
  - Red team: is composed by a group of security experts that use their expertise to attack the company's system
  - Blue Team: is composed by security experts that need to defend (also proactively) the company's system

When does it come to play

- Thanks to these teams companies can
  - Identify misconfigurations and coverage gaps
  - Strengthen network security to detect targeted attacks and improve breakout time
  - Raise healthy competition among security personnel and foster cooperation among the IT and security teams
  - Elevate awareness among staff as to the risk of human vulnerabilities which may compromise the organization's security
  - Build the skills and maturity of the organization's security capabilities within a safe, low-risk training environment

Threat Modeling

- It is a systematic approach to list all the potential way an attacker may attack an application
- Process to look at attacks actively
- The output of this process is a list of threats or probable threat scenarios
- It is a collaborative and repeatable process
- Important terms
  - weakness: software defect or bug
  - vulnerability: weakness that can be exploited
  - attack surface: anything that can be obtained, used, or attacked by a threat actor
  - risk: impact * likelihood

Threat Modeling

- Threat modeling is important for multiple reasons:
  - proactive approach to finding threats
  - increase efficiency by reducing cost
  - better prioritization based on bugs and mitigation plan
  - better understanding of the system
- Depending on the organization, threat modelling should be performed by different actors, comprising architects, developers, testers, and security experts
- Should be performed as early as possible in the software design process
- In agile, should be ideally performed during each sprint

Approaches to Threat Modeling

SPRITZ
SECURITY & PRIVACY
RESEARCH GROUP

UNIVERSITÀ
DEGLI STUDI
DI PADOVA

- **Asset-centric approach**
  - create a list of assets
  - draw assets, components, and data flows
  - for each element, check for threats
- **Attacker-centric approach**
  - create a list of threat actors (motive, means, opportunity)
  - create a list of threats
- **Application-centric approach**
  - draw a diagram of the application
  - list the threats for each element (STRIDE, OWASP TOP 10)
  - rank threats using a classification model

Step 1: Decompose the application

- Understand the application and how it interacts with external entities
- Crate use cases to understand how the application is used
- Identify entry points to see how the attacker can interact with it
- Identify assets, i.e., items or areas that the attacker would be interested in
- Identify trust levels that represent the access rights that the application will grant to external entities
- This information is documented in a *Threat Model* document

Decompose the Application

SPRITZ
SECURITY & PRIVACY
RESEARCH GROUP

UNIVERSITÀ
DEGLI STUDI
DI PADOVA

- Information identifying the threat model typically include
  - Application name
  - Application version
  - Description of the application
  - Document Owner
  - Participants
  - Reviewers

Decompose the Application

SPRITZ
SECURITY & PRIVACY
RESEARCH GROUP

UNIVERSITÀ
DEGLI STUDI
DI PADOVA

Application Version: 1.0

Description: The college library website is the first implementation of a website to provide librarians and library patrons (students and college staff) with online services. As this is the first implementation of the website, the functionality will be limited. There will be three users of the application:

Students, Staff, Librarians

Staff and students will be able to log in and search for books, and staff members can request books.

Librarians will be able to log in, add books, add users, and search for books.

Document Owner: David Lowry

Participants: David Rook

Reviewer: Eoin Keary

Decompose the Application

SPRITZ
SECURITY & PRIVACY
RESEARCH GROUP

UNIVERSITÀ
DEGLI STUDI
DI PADOVA

- The next step is the identification of the external dependencies, which are items external to the code of the application but that are still within the control of the organization
- However, they may not be within the control of the development team
- First area to consider is the production environment and requirements
- Understand how the application is or is not intended to be run
- External dependencies should be document as follows
  - ID: unique identifier assigned to the external dependency
  - Description: textual description of the external dependency

Decompose the Application

ID	Description
1	The college library website will run on a Linux server running Apache. This server will be hardened per the college's server hardening standard. This includes the installation of the latest operating system and application security patches.
2	The database server will be MySQL and it will run on a Linux server. This server will be hardened per the college's server hardening standard. This will include the installation of the latest operating system and application security patches.
3	The connection between the web server and the database server will be over a private network.
4	The web server is behind a firewall and the only communication available is TLS.

Trust Levels

- Represent the access rights that the application will grant to external entities
- They are usually cross-references with the entry points and assets
- Allows us to define the access rights or privileges required at each entry point and those required to interact with each asset
- They are documented as follows
  - ID
  - Name
  - Description

Decompose the Application

SPRITZ
SECURITY & PRIVACY
RESEARCH GROUP

UNIVERSITÀ
DEGLI STUDI
DI PADOVA

ID	Name	Description
1	Anonymous Web User	A user who has connected to the college library website but has not provided valid credentials.
2	User with Valid Login Credentials	A user who has connected to the college library website and has logged in using valid login credentials.
3	User with Invalid Login Credentials	A user who has connected to the college library website and is attempting to log in using invalid login credentials.
4	Librarian	The librarian can create users on the library website and view their personal information.
5	Database Server Administrator	The database server administrator has read and write access to the database that is used by the college library website.
6	Website Administrator	The Website administrator can configure the college library website.

Decompose the Application

SPRITZ
SECURITY & PRIVACY
RESEARCH GROUP

UNIVERSITÀ
DEGLI STUDI
DI PADOVA

- We now need to identify entry points, i.e., interfaces through which potential attackers can interact with the application or supply it with data
- Entry points can be layered, e.g., each web page in a web application
- They are usually identified by points where data enters and leaves the system
- They define a trust boundary
- Should be documented as
  - ID
  - Name
  - Description
  - Trust Level

Decompose the Application

SPRITZ
SECURITY & PRIVACY
RESEARCH GROUP

UNIVERSITÀ
DEGLI STUDI
DI PADOVA

ID	Name	Description	Trust Level
1	HTTPS Port	The college library website will be only be accessible via TLS. All pages within the college library website are layered on this entry point.	(1) Anonymous Web User (2) User with Valid Login Credentials (3) User with Invalid Login Credentials (4) Librarian
1.1	Library Main Page	The splash page for the college library website is the entry point for all users.	(1) Anonymous Web User (2) User with Valid Login Credentials (3) User with Invalid Login Credentials (4) Librarian
1.2	Login Page	Students, faculty members and librarians must log in to the college library website before they can carry out any of the use cases.	(1) Anonymous Web User (2) User with Login Credentials (3) User with Invalid Login Credentials (4) Librarian

Decompose the Application

SPRITZ
SECURITY & PRIVACY
RESEARCH GROUP

UNIVERSITÀ
DEGLI STUDI
DI PADOVA

- The system must have something the attacker is interested in, i.e., assets
- Assets are essentially targets of the attacker
- Can be both physical or abstract
- E.g., the asset of an application might be a list of clients and their personal information
- Assets are documented as follows
  - ID
  - Name
  - Description
  - Trust Levels

Decompose the Application

SPRITZ
SECURITY & PRIVACY
RESEARCH GROUP

UNIVERSITÀ
DEGLI STUDI
DI PADOVA

ID	Name	Description	Trust Level
1	Library Users and Librarian	Assets relating to students, faculty members, and librarians.	
1.1	User Login Details	The login credentials that a student or a faculty member will use to log into the College Library website.	(2) User with Valid Login Credentials (4) Librarian (5) Database Server Administrator (7) web server User Process (8) Database Read User (9) Database Read/Write User
1.2	Librarian Login Details	The login credentials that a Librarian will use to log into the College Library website.	(4) Librarian (5) Database Server Administrator (7) web server User Process (8) Database Read User (9) Database Read/Write User

Step 2: Determine and Rank Threats

- It is fundamental to use a threat categorization methodology
- An example of this is STRIDE or the Application Security Frame (ASF)
- The goal of the threat categorization is to help identify threats both from the attacker and the defensive perspective
- Data flow diagrams produced in step 1 help to identify the potential threat targets from the attacker's perspective
- These threats can be classified further as the roots for threat trees: onea tree for each goal
- The determination of the security risk for each threat can be mode using a value-based risk model such as DREAD, or likelihood-impact

Threat Categorization

- The first step in the determination of threats is adopting a threat categorization
- Provides a set of categories with correspondings examples so that threats can be systematically identified in the application in a structured and repeatable manner
- STRIDE is a threat categorization based on attackers' goals
  - **S**poofing
  - **T**ampering
  - **R**epudiation
  - **I**nformation Disclosure
  - **D**enial of Service
  - **E**levation of Privilege

Threat Categorization

SPRITZ
SECURITY & PRIVACY
RESEARCH GROUP

UNIVERSITÀ
DEGLI STUDI
DI PADOVA

Type	Description	Security Control
Spoofing	Threat action aimed at accessing and use of another user's credentials, such as username and password.	Authentication
Tampering	Threat action intending to maliciously change or modify persistent data, such as records in a database, and the alteration of data in transit between two computers over an open network, such as the Internet.	Integrity
Repudiation	Threat action aimed at performing prohibited operations in a system that lacks the ability to trace the operations.	Non-Repudiation
Information disclosure	Threat action intending to read a file that one was not granted access to, or to read data in transit.	Confidentiality
Denial of service	Threat action attempting to deny access to valid users, such as by making a web server temporarily unavailable or unusable.	Availability
Elevation of privilege	Threat action intending to gain privileged access to resources in order to gain unauthorized access to information or to compromise a system.	Authorization

Threat Analysis

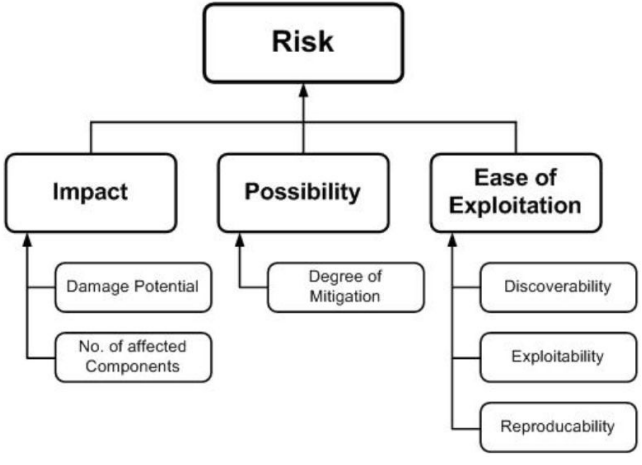
SPRITZ
SECURITY & PRIVACY
RESEARCH GROUP

UNIVERSITÀ
DEGLI STUDI
DI PADOVA

- It is fundamental to understand the generic definition of risk
- Risk is a potential for loss determined by two factors:
  - likelihood that an attack will occur
  - potential impact/cost of that successful attack
- Risk is hence calculated as likelihood x impact
- From a risk management perspective, threat modeling is a systematic and strategic approach for identifying and enumerating threats to an application environment with the objective of minimizing risk and potential impact

Ranking of Threats

- Threats can be ranked from the perspective of risk factors
- Possibility to define a prioritized list of threats to support a risk mitigation strategy



Step 3: Determine Countermeasures and Mitigation

- A vulnerability can be mitigated with the implementation of a countermeasure
- Once a risk ranking is assigned to the threats identified in step 2, it is possible to sort threats from the highest to the lowest and prioritize mitigation efforts
- Options for addressing risks based on business impact include
  - Accept: decide that the business impact is acceptable
  - Eliminate: remove components that make the vulnerability possible
  - Mitigate: add checks or controls that reduce the risk impact

Threat Categorization

SPRITZ
SECURITY & PRIVACY
RESEARCH GROUP

UNIVERSITÀ
DEGLI STUDI
DI PADOVA

Threat Type	Mitigation Technique
Spoofing Identity	1. Appropriate authentication, 2. Protect secret data, 3. Don't store secrets
Tampering with data	1. Appropriate authorization 2. Hashes, 3. MACs, 4. Digital signatures, 5. Tamper resistant protocols
Repudiation	1. Digital signatures, 2. Timestamps, 3. Audit trails
Information Disclosure	1. Authorization 2. Privacy-enhanced protocols 3. Encryption 4. Protect secrets 5. Don't store secrets
Denial of Service	1. Appropriate authentication 2. Appropriate authorization 3. Filtering 4. Throttling 5. Quality of service
Elevation of privilege	1. Run with least privilege

Looking for Vulnerabilities

- Coming up with all possible vulnerabilities for piece of code or asset in you software development life cycle may be a hard task
- You need to be aware of all possible attacks and threat vectors
- Luckily, there exist the CVE Program, which is to identify, define, and catalog publicly disclosed cybersecurity vulnerabilities
- There is one CVE record for each vulnerability in the catalog
- Partners of the CVE program publish CVE records to communicate consistent descriptions of vulnerabilities
- IT and cybersecurity professionals use CVE records to ensure they are discussing the same issue and coordinate their efforts to prioritize and address the vulnerabilities

Scoring System

- In threat modeling we need to assign a level of severity to the identified vulnerabilities
- To help organizations adopting the same metric for common vulnerabilities we use the Common Vulnerability Scoring System (CVSS)
- Free and open industry standard for assessing the severity of computer system security vulnerabilities
- Scores are calculated based on a formula that depends on several metrics that approximate ease and impact of an exploit
- Scores range from 0 to 10, with 10 being the most severe

NVD by NIST

- The NVD is the U.S. government repository of standards based vulnerability management data represented using the Security Content Automation Protocol (SCAP)
- Enables automation of vulnerability management, security measurement, and compliance
- NVD includes databases of security checklist references, security related software flaws, misconfigurations, product names, and impact metrics
- Analysis results in association impact metrics (Common Vulnerability Scoring System - CVSS), vulnerability types (Common Weakness Enumeration - CWE), and applicability statements (Common Platform Enumeration - CPE)

Example: Log4j

- Example of Log4j

OWASP Top 10

- It is a standard awareness document for developers and web application security
- It represents a broad consensus about the most critical security risks to web applications
- Using it is perhaps the most effective first step towards changing the software development culture towards a one that produces more secure code
- It is updated based on the newly identified threats and their impact: mixture between data-driven and community survey