Computer Security (CM2025)

Course Notes

Felipe Balbi

October 27, 2020

Contents

Week 1

Key Concepts

- Understand the central goals and aspects of computer security.
- Understand and explain the differences between a range of malware types.
- Identify key examples of malware and their historical significance.

Reading: Ethics in information security

Read the following article:
Ethics in information security, IEEE Security & Privacy, vol.15 May/June 2017, pp.3–4.
You should also familiarise yourselves with the ACM ethics guidelines.

Defining computer security

With the advent of the Internet, computer security became a very important field of research.

Computer Security is the *protection of computer-related assets against danger, loss or loss of control of something valuable.*

Security has three main goals:

Prevention The safeguarding of assets from threats;

Detection Systems that alert if malicious activity is, or is about to, take place;

Reactiion Definition of procedures that enables us to deal with an attack.

Security can also be split into three main components:

Policy deals with confidentiality, integrity and availability of data;

Threat Model assumptions about those involved with malicious activity;

Mechanism The SW/HW used to make sure the policies are enforced using the assumptions made during Threat Modelling.

There are five important terms that need to be defined:

Attack activities harmful to computer systems, data, software, and hardware;

Risk the possibility of damage or loss of digital assets in case of an attack;

Zero-day (vulnerability) a vulnerability used by an attacker before its discovery by the developer of the SW;

Exploit SW used to take advantage of a bug or vulnerability;

Hacker Subdivided into three groups

White Hats Find vulnerabilities with a goal of fixing them before its discovery by an attacker. They work under the permission of the owner of the computer system being attacked;

Black Hats Try to penetrate the system to gain unauthorized access. Often, their motivation is harm operations or steal sensitive operations;

Gray Hats They fall somewhere in-between White Hats and Black Hats and work with varying combinations of good and bad intentions.

Types of malicious software 1

Malware is a piece of softwarte designed to disrupt, damange and destroy an information system. There are many types of malwares, some of which are discussed in the following subsections

Viruses

They self-replicate by inserting themselves into other files, programs, documents, etc. Can spread through emails, USB sticks and downloads from unknown sources.

The *Creeper* is considered to be one of the first computer viruses. Developed in 1971, it infected computers connected to the ARPANET, the internet prototype.

Another early virus was the *Elk Cloner*. Written by a high-school student to infect Apple II computers using floppy disks in 1982. Every 50th time the computer booted, it would display a poem written by the hacker.

Not all viruses are harmless. The *I Love You* virus in 2000 caused around $10 billion worth of damages by affecting nearly 10% of all computers around the globe.

Worms

They can replicate without attaching themselves to existing software. The *Stuxnet* is a well-known worm, considered to be one of the most destructive worms ever created. It was designed to attack Programmable Logic Controllers by Siemes.

PLC devices are used for the automation of processes in machinery. In this case, it targetted centrifuges in Iranian nuclear power plants and altered the speed of the machine, causing it to tear itself apart.

It's estimated that *Stuxnet* destroyed 20% of Iranian's nuclear power plant centrifuges.

Adware

These display advertisements on your screen during browsing or online shopping. Possibly the most visible form of malware one can encounter. Its main purpose is to collect user data.

Trojans

Trojans are named after the famous ancient Greek tale of the invasion of the city of Troy by the Greek during the Trojan War.

After trying and failing several times to get access to the city, the Greeks came up witha plan where squatter soldiers would hide in the joint wooden status of a horse presented as a gift.

During the night, after entiring the city within the horse, the Greeks broke out of the horse to attack the city.

Trojans have a similar way of working: they hide themselves inside an application or program data and spread based on specific user action.

Spyware

Designed to spy on the target machine or the user, it collects information and sends it back to the hacker for further use or for sale on the dark web.

The *Dark Hotel* spyware is one famous case which used Hotel Wi-Fi to target the personal systems of government officials, business tycoons and political leaders to extract sensitive information.

Types of malicious software 2

We have a look at *Keyloggers*, *Ramsonware*, *Botnets*, and *Rootkits*.

A *Keylogger* records every keystroke from the user. This may include messages being typed, emails, confidential information such banking credentials, users, passwords, etc.

Olympic Vision is a keylogger used for Business Email Compromise (BEC) attacks. It also uses several other pieces of malicious software to steal sensitive information and spy on business transactions. Nowadays it's very easy to get a hold of a keylogger.

In a *Ramsonware* attack, the victim's data is encrypted, backup files are deleted and the people responsible demand money in exchange for decryption of this data. In other words, the victims are held to *ramsom* for renewed access to their data. The *WannaCry* attack is a recent example that took place in May 2017.

In a *Botnet* attack, computers connected to the internet are taken over by an attacker which remotely controls the computers using a Command And Control (CNC) server to carry out Distributed Denial Of Service (DDOS) attacks. A DDOS attack is when a given server is flooded with so much traffic at one moment, that it collapses.

EchoBot is a botnet used to exploit over 59 known vulnerabilities and launch a number of attacks, such as DDOS attack, steal sensitive information, conduct corporate espi-

onage, and infects a wide range of Internet Of Things connected devices. Furthermore, it also scans for old vulnerabilities in legacy systems for future exploitation.

Finally, we have *Rootkits*. These remain hidden in a target computer and activate in secret. They can perform several activities rangin from giving attackers remote access to a computer all the way to stealing sensitive information such as a password or credit card details. They can also use a compromised computer to launch any of the other attacks described before.

Week 2

Key Concepts

- Explain the key differences between static and dynamic analysis.
- Explain the usage of sandboxes in malware analysis.
- Understand the need for a variety of methods of malware analysis.

Malware analysis and techniques

Static Malware Analysis is one of the techniques used to analyze and combat the types of malware discussed previously.

More generally, Malware Analysis is a set of processes and techniques that help a Security Analyst understand the functionality, origin, impact, and intent of malicious software.

The goal of this activity is find the *Indication Of Compromise* (IOC) that depicts the behavior of malicious software. IOCs are also used to develop signatures of malware. The two techniques used for such analysis are Static Malware Analysis and Dynamic Malware Analysis.

In Static Malware Analysis, the executable files are examined without being executed. We can determine if a file is clean or malicious and also discover information about its functionality.

With the information collected during Static Malware Analysis, allows us to determine signatures, which are a collection of distinguishing features that can be used to recognize malware.

Two essential techniques used in static analysis are antivirus scanning and hashing. Antivirus scanning is the traditional method of running a file through an antivirus scanner to try to detect whether the file contains malware. Hashing is an algorithm that produces a value referred to as a *Hash* which is a unique fingerprint for a given file. Any modification to a file will result in a different hash fingerprint, including infection by a malware.

While static analysis methods are useful as a starting point for some more basic types of attack, it can be powerless against more recent and advanced types of malware. These have found ways to circumvent the detection methods used during static analysis.

The solution to this is Dynamic Malware Analysis.

Malware analysis 2

Dynamic Analysis or Behavioral Analysis is where we execute the malware in a controlled environment known as a sandbox. This allows a Security Professional to observe the behavior of malicious software, help to understand its functionality and, hopefully, find the Indicator Of Compromise.

This method overcomes some of the limitations of Static Analysis with regards to catching the more advanced and adaptive forms of malware.

Dynamic Analysis is an efficient method for analyzing malware because it helps uncover the functionality of the malware, which is not entirely possible with static analysis.

A Sandbox is a virtualized environment that contains a virtual network, services, drives, etc, to ensure that the malware behaves exactly as it would in a real environment.

There are two main types of sandboxes: Agent-based and Agent-less. Agent-based sandboxes require software to be installed on every computer that needs to be monitored. Well known examples are cuckoo, threat expert, bit blase, and Comodo. Conversely, an Agent-less sandbox monitors computers on the network from afar without needing to be installed on every device. Popular examples are VMRay, Analyzer, and SNDBOX.

Security Researchers use both types of sandboxes, but some research suggests that agent-less sandboxes are more efficient.

Another common tool for dynamic analysis on Windows machines is process monitor (Procmon). It's used to monitor the registry, file system, network, running processes, etc.

Ethics

Ethics is really important in the field of Computer Security.

Because we frequently work with computers, we may be exposed to security issues and vulnerabilities. When that occurs, we may be able to fix the issue ourselves by setting a rules in our *iptables* or blocking traffic from a certain port for instance.

There may be, however, wider ramifications to the problem we have discovered. It could be something worth mentioning to the maker of the faulty software.

That's where **Responsible Disclosure** comes into the picture. The term itself is somewhat subjective, but there may be legal ramifications related to the disclosure of a security issue.

In a situation where we find an issue in a popular software package, e.g. a popular Operating System, it may be the case that many other users are affected. This means that we have some responsibility to the Company or Service Provider in terms of disclosing the issue.

It is common practice to identify the bug to the provider and offer them enough time to fix the issue before disclosing the problem publicly. There are considerations about transparency.

Some companies may refuse to patch issues or even seek legal action against those exploiting or even simply highlighting the issue.

We should also think about security in a distributed way. For example, cloud computing services offer on-demand computing. These services may not be hosted in the locality of the developer or the client. As such, we must consider conforming to the law in **all** of these locations.

As a final thought, actions have consequences. Identity theft and distributed denial of service attacks have real world consequences. People's lives can be destroyed if we do something that can cause harm.

Passwords

Designing truly secure systems is very hard, only made harder by data leaks happening periodically as can be seen in the news. Data is becoming more and more valuable and there are places in the *Dark Web* where one can buy leaked data.

One of the most prolific leaks utilised a simple encryption method where the same encryption key was utilized. This resulted in a one-to-one mapping between plain-text and encrypted passwords. In other words, there was a situation where different users with the same password would end up with the same encrypted string stored in the database.

What this means is that if one password is cracked, all other users who happen to be using the same password were also compromised. To make matters worse, some users used password hints, which were also stored in the database.

Reusing passwords is also a common problem. If a user's password for one service is leaked, there is a probability that the same user employed the exact same password on multiple services which renders all of such services compromised.

There are ways to design more secure systems, however that also has implications. We could require longer passwords and encourage two-factor authentication, but a user may loose their phone or forget that longer, more complex password.

A good system design balances accessibility with security and usability.

Social Engineering

Insecure designs can have far reaching ramifications, however a system is only as secure as its safest link.

Social Engineering is one of the most common attack vectors and it does not rely on technical subtleties of attacks. These attacks rely on the fact that not all staff are properly trained in security and, as such, attackers may exploit gaps in their knowledge.

A company may have a robust security policy in place to handle access control, however **compliance** is a different matter.

Phishing emails are a good example of Social Engineering attacks. They try to trick you into thinking the email comes from a reliable source and convince you to give them the information they're after. Some of these emails may look fairly authentic and we may have to look deeper to determine their authenticity.

Week 3

Key Concepts

- Describe the CIA objectives of network security.
- Use real examples to describe how DoS attacks and DDoS attacks work including those using botnets.
- Describe the levels of security in wireless networks and common attack vectors.

The objectives of network security: confidentiality, integrity and accessibility

Network security has three main objectives:

Confidentiality Read access control, i.e. who can **read** which piece of information

Integrity Write access control, i.e. who can **write** which piece of information

Availability Maintaining function, i.e. guaranteeing that the information will be available to those who can access it

It's composed of a set of policies and practices to protect the network. One example of a policy may be:

Every access to the network is unauthorized unless the user is authenticated with username and password

A practice related to the policy may be the fact that any user must be given a username and password and an authentication server needs to be maintained.

Paper about CIA

Reading about the conflicting aspects of confidentiality, integrity and availability:

K. S. Wilson, Conflicts Among the Pillars of Information Assurance, in IT Professional, vol. 15, no. 4, pp. 44-49, July-Aug. 2013, doi: 10.1109/MITP.2012.24.

The attack surface and the denial of service attack

A *Denial of Service*, or *DoS*, attack is when we flood a server with so many requests that it collapses under the load, therefore **denying service** to the users.

This attack can happen in many of the 7 layers of OSI model. As a quick summary, here are what each of 7 layers represents:

1. Physical Layer

   The hardware pieces. Cables, networking cards, etc. Responsible for the transmission of unstructured raw data. Converts digital bits into electrical, radio, or optical signals.

2. Data Link Layer

   Provides node-to-node transfers. Can detect and maybe correct errors.

3. Network Layer

   Provides the infrastructure for transmission of variable-length packets from one node to another connected in different networks.

4. Transport Layer

   Provides the infrastructure for the transmission of variable-length data sequences from a source to a destination.

5. Session Layer

   Controls the dialogues between computers. Establishes, manages and terminates connections between local and remote application.

6. Presentation Layer

   Establishes context between application-layer entities.

7. Application Layer

   The layer closest to the user. User and OSI interacts directly with the application.

With these in mind, we can look at some example attacks for some of these layers.

ARP Flood Attack is a layer 2 attack where one would keep broadcasting the network with ARP requests consuming a lot of the available processing power of the target machine. ARP requests are broadcast messages used to ask the network which MAC address corresponds to an IP address.

ICMP Ping Flood Attack is a layer 3 attack relying on the *ping* diagnostic message. Hosts are required to respond to ICMP Ping requests. By sending a flood of ping requests, one can keep a server busy processing such ping requests.

TCP-SYN Flood Attack is a layer 4 attack which tries to open several TCP connections by sending a flood of TPC-SYN packets.

The anatomy of a DDOS, botnets and Mirai

A *Distributed Denial of Service* Attack, or *DDoS*, is similar to a *DoS* attack however it's accomplished with many different machines targetting a service, hence the "distributed" in the name.

A BotNet is slightly different. An attacker will take over the control of several machines from regular users on the internet, and make those machines target a single service with a flood attack of some kind.

Many *Internet of Things*, or *IoT*, devices on the market have poor security features, which makes them a target for BotNets and carry out DDoS attacks on other servers.

One such case is the Mirai BotNet, which targets certain IoT devices. The most prominent Mirai DDoS attack, took down the DNS provider Dyn resulting in Netflix, Github, Twitter, Reddit and many other major services being rendered inaccessible. After analysis, Dyn claimed that there were up to 100,000 malicious endpoints involved in the attack.

Mirai GitHub and Research Paper

- Analysis of the Mirai botnet:

  H. Sinanović and S. Mrdovic, Analysis of Mirai malicious software, 2017 25th International Conference on Software, Telecommunications and Computer Networks (SoftCOM), Split, 2017, pp. 1-5, doi: 10.23919/SOFTCOM.2017.8115504.

- Mirai Source Code

  `https://github.com/jgamblin/Mirai-Source-Code`

- Recent Paper About the Threat of Botnets

  A. Woodiss-Field and M. N. Johnstone, Assessing the Suitability of Traditional Botnet Detection against Contemporary Threats, 2020 Workshop on Emerging Technologies for Security in IoT (ETSecIoT), Sydney, Australia, 2020, pp. 18-21, doi: 10.1109/ETSecIoT50046.2020.00008.

Wireless attacks: WiFi attack vectors

The Wi-Fi Alliance is a network of companies working on wifi technology and standards. The idea being that device manufacturers want their devices to interoperate, therefore a standard is created which describes the method of communication in the wireless network.

A timeline of the different standards are as follows:

- 802.11a: 1999
- 802.11b: 1999
- 802.11g: 2003

- 802.11n: 2009
- 802.11ac (wifi 5): 2012
- 802.11ax (wifi 6): 2020

Paired with the wireless standards, there a set of wireless security protocols:

- WEP (802.11a/b): 1997
- WPA (802.11g): 2003
- WPA2 (802.11i): 2004
- WPA3: due in 2020

WEP is notorious for having weak encryption in several aspects, which allowed attackers to exploit it and gain access to the network.

There are different attacks that can be carried out on Wireless networks. Some examples:

Dictionary Attacks short keys allow brute force encryption breaking. In other words, if the key is small, it's feasible to try all options

Fluhrer, Mantin and Shamir Attack famous encryption breaking hack for WEP

Replay Attacks replay a sequence of packets, with edits. It may allow an attacker to get a valid session key

PRGA / Packet Tampering Attack allows an attacker to masquerade as another device

Wireless Networking Security

- Classic paper reporting on WEP's vulnerabilities

  Scott R. Fluhrer, Itsik Mantin, and Adi Shamir. 2001. Weaknesses in the Key Scheduling Algorithm of RC4. In Revised Papers from the 8th Annual International Workshop on Selected Areas in Cryptography (SAC '01). Springer-Verlag, Berlin, Heidelberg, 1–24.

- Article about attack vectors for different Wifi protocols such as WEP, WPA and LEAP

  Hal Berghel and Jacob Uecker. 2005. WiFi attack vectors. Commun. ACM 48, 8 (August 2005), 21–28. DOI: `https://doi.org/10.1145/1076211.1076229`

- Very thorough review of security in different wireless technologies used in IoT devices

  K. Lounis and M. Zulkernine, Attacks and Defenses in Short-Range Wireless Technologies for IoT, in IEEE Access, vol. 8, pp. 88892-88932, 2020, DOI: 10.1109/ACCESS.2020.2993553.

Week 4

Key Concepts

- Describe three types of firewall and reason about the appropriate type of firewall to use for a given situation.
- Explain how intrusion detection systems work and give examples of historical and contemporary systems.

Firewalls – our first line of defence

Stateless Firewalls A type of Access Control Lists (or ACL). It checks all traffic against a set of rules;

stateful Firewalls More efficient than Stateless Firewalls. Once a packet session is allowed, no filters need to be applied

Proxy Firewall Carries out external network access.

Intrusion detection systems (IDS)

An Intrusion Detection System, or IDS, is a system which runs on a network and aims at detecting when a intruder has compromised the network. The reason such systems are used is because it's virtually impossible to make any network perfectly secure.

/"Most security experts agree that a completely secure system is impossible to achieve, so we must stay alert for attacks."/. (Kemmerer and Vigna, 2002.)

The IDS is about staying alert for attacks and detecting them as soon as possible.

When it comes to implementation of IDSs, Dorothy Denning state, in 1987, that "the model is based on the hypothesis that exploitation of a system's vulnerabilities involves abnormal use of the system; therefore, security violations could be detected from abnormal patterns of system usage". What this means is that an intruder is likely to exhibit abnormal usage patterns on the network.

If one can detect such abnormal usage patterns, then we can detect intruders.

While looking for intruders, we want to avoid false positives (a regular user classified as an intruder) or false negatives (an intruder classified as a regular user).

We also want our IDS to be fast and come to a conclusion using minimal resources. All this while analyzing traffic over the entire network. Modern systems using Docker containers can have a complex network structure and our IDS still needs to be fast while analyzing traffic of complex networking schemes.

Recent IDSs employ state-of-the-art Machine Learning and AI algorithms for improved pattern recognition. Deep-learning is used in many recent research papers in computer security related to IDS.

When developing state-of-the-art IDS using Deep Learning, we need a dataset to train the neural network in order to verify correctness of the system. A common dataset to use for this case is the KDD99 Dataset.

Intrusion Detection Systems

- D. E. Denning, An Intrusion-Detection Model, in IEEE Transactions on Software Engineering, vol. SE-13, no. 2, pp. 222-232, Feb. 1987, doi: 10.1109/TSE.1987.232894.
- R. A. Kemmerer and G. Vigna, Intrusion detection: a brief history and overview, in Computer, vol. 35, no. 4, pp. supl27-supl30, April 2002, doi: 10.1109/MC.2002.1012428.
- S. Wang, C. Xia and T. Wang, A Novel Intrusion Detector Based on Deep Learning Hybrid Methods, 2019 IEEE 5th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing, (HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS), Washington, DC, USA, 2019, pp. 300-305, doi: 10.1109/BigDataSecurity-HPSC-IDS.2019.00062.

Week 5

Key Concepts

- Describe the features of a typical operating system, with specific details about the Windows system.
- Explain the core functionality of a file system and why it needs to be secure.
- Identify security features and flaws of contemporary and historical versions of Windows operating systems.

Operating systems: windows and OS hardening

The Operating System (OS) is a piece of Software that manages hardware resources and provides services to competing programs.

A few important definitions:

OS Security The processes and methods involved in guaranteeing the the integrity, confidentiality and availability of the OS

OS Protection Refers to methods and procedures to protect the OS from intruders and attacks

OS security includes all precautionary control techniques to help protect any computer resources that might be removed or modified if the OS is compromised.

Some key components of an OS are:

Kernal Executes services of the OS at the lowest level

Security Kernal Manages all OS's security processes

Reference Monitor Manages access to the device

The Security Kernal and Reference Monitor, together, for the Trusted Computing Base or TCB, which has everything necessary to enforce OS security policies.

In the Windows OS, the basic security blocks are:

Security Reference Monitor executes access checks

Local Security Authority executes windows local security policies

Security Account Manager database that stores credentials

Active Directory a directory service for Windows domain networks

WinLogon and NetLogon WinLogon manages local input logins; NetLogon manages network-wide logins

Some of the most commonly found client-side vulnerabilities are found in web browsers, office suites and the like. While any software could be attacked, attacks are, in practice, concentrated on the most popular software titles.

In order to overcome attacks, we must *harden* our systems before deployments. Hardening refers to the process of making an operating system more secure. For example, disabling automatic logins, enabling screen lock during screen saver, switching system security on, ensuring wireless connection is disabled if not needed, and so on.

File system and directory structure

A file is a set of linked data stored in non-volatile media. Each file has a logical location for storage and retrieval. In the operating system the *File System* is the data structure used to store, retrieve, and keep track of the files stored on the disk.

Some commonly used filesystems are:

- Windows
  - FAT32
  - exFAT
  - NTFS
- macOS
  - HFS+
  - APFS
- Linux
  - ext3
  - ext4
  - btrfs
  - ffs
- Unix
  - UFS
  - ZFS

There are three main types of files available for use in the OS:

Text File sequence of letters arranged in lines

Object File collection of bits stored in blocks

Source File refers to a variety of operations and activities in the operating system

The file system also maintains a set of metadata for each file, including creation time, update time, the actual volume, etc. All of this information is referred to as the file attributes.

Important attributes are:

File type the type of the file

Permissions who can read, write, and execute the file

Timestamps time and date of creation and last update

A file system also has a directory structure.

Windows Security

- Brief History of Windows Security
- Windows Consumer Security circa Oct 2020:
- Windows Enterprise Security circa Oct 2020