ECAM
BRUSSELS ENGINEERING SCHOOL

I5020 Computer Security

Session 4

Malware and Protection

Sébastien Combéfis

Fall 2019

Objectives

- Discovering the notion of malware

  Definition, classification, threat and countermeasure

- Characterisation of different types of malwares
  - Propagation mechanism on several targets
  - Different payload types and associated threats

- Design and deployment of countermeasures

  Criterion for a good countermeasure and examples

Malware

Malware

- Malware is the main threat on computer systems

  Affects different programs (application, kernel, compiler...)

"A program that is inserted into a system, usually covertly, with the intent of compromising the confidentiality, integrity, or availability of the victim's data, applications, or operating system or otherwise annoying or disrupting the victim."

Classification

- Inspection of threats and countermeasures related to malwares

  Also present in servers, forged spam emails...

- Two main ways to categorise malwares
  - Depending on how they are propagating
  - Depending on their action type or payload used once in place

Malware Tour (1)

- **Advanced Persistent Threat** (APT)
  - Cybercrime directed towards business and political targets
  - Persistent threats over an extended period of time
- **Adware**

  Advertising integrated in a software (popup, HTTP redirection...)
- **Attack kit**

  Set of tools generating malware automatically
- **Auto-rooter**

  Hacker tools to penetrate machines remotely

Malware Tour (2)

- **Backdoor** (trapdoor)

  Mechanism that overrides a normal security check

- **Downloader**

  Code that installs something on a machine being attacked

- **Drive-by-download**

  Code that exploits browser vulnerability to attack clients

- **Exploits**

  Code specific to one (a set of) vulnerability(ies)

Malware Tour (3)

- **Flooders** (DoS client)

  Generate large volume of data to attack a networked system

- **Keyloggers**

  Capture keys pressed on a system

- **Logic bomb**

  Sleeping code inserted in malware, waking up under conditions

- **Macro virus**
  - Virus that uses macro/script, embedded in a document
  - Enabled when the document is open and replicates in others

Malware Tour (4)

- **Mobile code**
  - Software that can be send on heterogeneous platforms
  - Does not need to be modified and same semantic execution
- **Rootkit**

  Set of tools for after introduction and getting root access
- **Spammer programs**

  Sending a large volume of unsolicited emails
- **Spyware**

  Collection/transmission of information about system activity

Malware Tour (5)

- **Trojan horse**

  Software with useful function that hides malicious code

- **Virus**

  Malware that duplicates itself in other code

- **Worm**

  Software running independently and that can spread

- **Zombie**, bot

  Program on infected machine attacking other machines

Attack Kit

- Creation and deployment of malware requires technical skills

  First malwares were real artworks

- Emergence of attack kits to create malwares

  Also known as crimeware (Zeus, for example)

- Modules with propagation mechanism and payload
  - Construction by composition, selection and deployment
  - Exploitation of opportunity window after discovery

Attack Source

- Initially attackers were individuals

  Motivated to show their skills to their peers

- More organised and dangerous attack sources
  - "Political" attackers, criminals and organised crime
  - Organisation selling services to companies and nations

- Development of an underground economy

  Attack kits sale, compromised host access/stolen information...

Advanced Persistent Threat (APT)

- APT attributed to organisations sponsored by states
  - Application of intrusion technologies and malwares
  - Rather business or political target type
- Very different from other types of attack
  - Very rigorous selection of the target
  - Persistent and stealthy intrusion efforts over a long period
- Two main goals for this type of attack
  - Intellectual property theft, data on infrastructure
  - Interference or physical interruption of the infrastructure

Propagation

Propagation

- Two approaches to classify propagation mechanism
  - Need or no need to have a host program
  - Possibility or not to replicate itself
- Several existing mechanisms for the propagation
  - Infection of existing executables with viruses
  - Exploitation of software vulnerabilities by worms
  - Drive-by-downloads to enable malware replication
  - Social engineering types of attack

Infected Content

- Parasitic fragment attaching itself to an executable content

  Affects application, utility, system program, bootcode...

- Executes itself secretly when the host is executed

  Initially easy because no access control

- Can take the form of a script for active content

  Microsoft Word document or Adobe PDF, Excel spreadsheet...

Virus

- A virus is a software that can infect a program

  Will change its content and therefore its behaviour

- Virus Brain released in 1986 against MS-DOS
  - Considered as the first virus for MS-DOS on IBM PC
  - Remplace the boot sector of a floppy disk with a virus copy

- Permanent battle between virus and anti-virus creators

  Countermeasures for existing viruses during creation of new ones

Virus Part

- Computer virus typically consisting of three parts
  - **Infection vector** defines how the virus propagate
  - **Trigger** defines when the payload is activated
  - **Payload** defines what the virus is doing
- Malware also typically includes some of these components

  One or several, and sometimes variants
- Embed machinery to make replications of itself

  Exploit a host with all the permissions it holds

Virus Lifecycle

- Virus lifecycle typically with four phases
  - **Sleeping** does nothing because in idle mode
  - **Propagation** makes copies (sometimes morphs) of itself
  - **Triggering** activated to realise its function
  - **Execution** of the function
- Execution specific to the OS or hardware platform

  Designed to take advantages of weaknesses

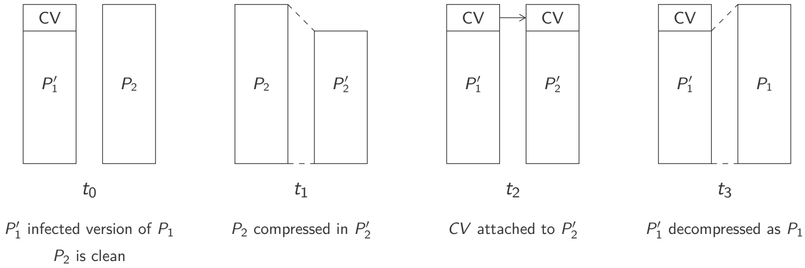
Virus Structure

- Code typically added at the beginning or end of an executable

  Virus must be executed first when the host is running

- Infected program length differs from healthy program

  Possible to compress the executable file



P'_1 infected version of P_1
P_2 is clean

P_2 compressed in P'_2

CV attached to P'_2

P'_1 decompressed as P_1

Virus Classification

- Viruses can be classified according to the target
  - **Boot sector** infection and propagation at startup
  - Infection of **files** considered as executable by the OS
  - Infection of **macros/scripts** executed by an application
  - **Multi-party** infection
- Four main possible concealment strategies

  Encrypted, stealthy, polymorphic or metamorphic viruses

Encrypted and Stealth Virus

- Possibility to encrypt the content of the virus
  - Virus portion creates a random key to encrypt the remainder
  - Random key is stored inside the virus
  - Choice of a different key at each replication
  - No constant bits pattern to observe
- Virus can be designed to hide themselves from detection
  - All the virus, including the payload is hidden
  - Code mutation, compression, rootkit techniques

{Poly,Meta}morphic Virus

- Polymorphic viruses embeds a mutation engine
  - Allows the virus to create variants of itself
  - Mutation engine itself is altered with each use
  - The different versions are functionally equivalent
- Metamorphic viruses also mutate at each infection
  - The virus completely rewrite itself at each iteration
  - Can also change behaviour in addition to appearance

Macro Virus

- Macro virus infects script code in a document

  Exploit the possibility of having document with active content

- Extremely threatening virus for four main reasons
  - Independent of the platform, only linked to the application
  - Attack documents, more easily introduced
  - Much more easily propagated, including by email
  - Bypasses more easily file access control

Vulnerability Exploit

- Worm actively search for other machines to infect

  Infected machine as launch base for attacks to other

- Exploit software vulnerabilities on client and server sides
  - Main goal is to gain access to new systems
  - Broadcast over network connections or removable media

Worm Replication

- Several possible means to access remote system
  - Send oneself by email/messenger, copy on removable media
  - Execution, access o a remote file or login
- Execution of the payload with propagation

  Phases as for viruses: sleeping, propagation, triggering, execution
- Search for access mechanisms to other systems

  Host table, address book, buddy list...

Worm Propagation Model (1)

- Simplified epidemic model classic in biology

$$\frac{dI(t)}{dt} = \beta I(t)S(t)$$

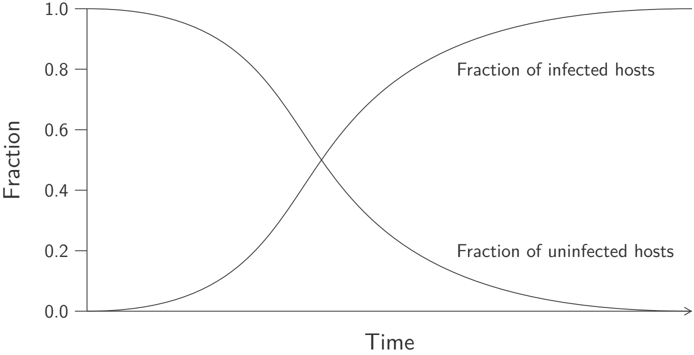
- where:
  - $I(t)$ number of individuals infected at time t
  - $S(t)$ number of individuals likely to be infected at t
  - β the infection rate
  - $N = I(t) + S(t)$ the size of the population

Worm Propagation Model (2)

- Worms propagation in three phases

  Slow start, fast propagation and slow final phase

- Worms end up trying to infect already infected machines



Drive-by-download

- Exploit bug in an application to install a malware

  Common technique consists to go through the web browser

- Downloading malware du malware against the will of the user

  No active propagation, waiting for infected page visit

- Payment to place ads containing a malware

  The attacker targets his/her ads to target websites

Social Engineering

- Trapping the user to compromise his/her own system

  Or reveals his/her private personal information

- Two main techniques of social engineering

  - Sending bulk unsolicited emails

    90% of sent emails are SPAM

  - Hide malicious code inside a Trojan

    Harmful, unwanted function when executed

- Emergence of Trokan for mobile devices

  Broadcast through apps download platforms

Payload

Payload

- A malware contains a payload performing an action

  Spreading, hiding, updating...

- Several types of existing payload action
  - Corruption of the system and its data
  - Service theft to make the system a zombie
  - Information theft like passwords
  - Stealth when the malware hides its presence

System Corruption (1)

- Some malware has the sole purpose of spreading

  But most of them do have a payload

- Endangering the integrity of the attacked system
  - Destruction of data on the infected system
  - Displaying unwanted messages or content
  - Infliction of actual damages

System Corruption (2)

- Data destruction on the disk of the infected system

  Or encryption of the content and ransom to retrieve

- Possibility to cause damages on the infected system

  Rewriting of BIOS boot code, industrial control system

- Logic bomb “explodes” when certain conditions are fulfilled

  Alters or modifies data or files

Attacker

- Disrupt computing and network resources for the attacker

  The infected machine acts like a (ro)bot, zombie, drone...

- Launch or attack management difficult to trace

  Make it difficult to trace the creator of the bot

- Possibility to organise a coordinated attack

  *Setting up a collection of bots called **botnet***

Bot Types (1)

- **Distributed denial of service** (DDoS) attacks

  Causes loss of services of a system for its users

- **Spamming**

  Massive sending of unsolicited emails thanks to botnet

- **Sniff** the traffic

  Clear information watching over compromised machine

- **Keylogging**

  Captures key pressed, better than sniffing if encrypted

Bot Types (2)

- **Propagating** new malwares

  Bot can download and execute files via HTTP/FTP

- Installing **ads add-ons** and *browser helper object* (BHO)

  Fake website with ads and bots to click on them

- Attacks on **IRC chat networks**

  Saturate IRC network of a victim as with DDoS attacks

- **Manipulation** of polls/online games

  Each bot can vote with its own IP address, legitimately

Remote Control

- Difference between worm and bot regarding the control
  - A worm propagates and activates itself
  - Bot controlled by *command-and-control* (C&C) servers
- Several possible communication means for bots
  - Bots join an IRC channel to receive commands
  - Communication channels hidden above HTTP
  - Distributed control mechanism with peer-to-peer protocol

Information Theft

- Harvesting information stored on infected system

  Communicated to attacker for fraudulent use

- Typically retrieving login and password

  For banking, game and other similar applications

- Attacks on the confidentiality of certain information
  - Configuration details and system documents, for example
  - Used for reconnaissance or espionage

Credential Theft

- Sensitive data often sent by HTTPS, POP3S...
  - Attacks against this protection use keyloggers
  - Using filters to obtain relevant information
- Track all the activity of a user with spyware
  - History and browsing activity
  - Redirect users on fake webpages
  - Dynamic of exchanges between browser and serveur

Identity Theft

- Sending a URL in a spam pointing to fake site
  - Fake site completely controlled by the attacker
  - Spam message invokes an emergency
- Phishing requests personal information by a form

  Leverage user trust through social engineering
- Extreme customisation of emails by spear-phishing

  Search on target, citing personal information

Stealth

- Malware has mechanisms to hide its presence

  Allow undercover access to the infected system

- Secret entry point backdoor/trapdoor in a program
  - Getting around security and access control mechanism
  - “Legal” backdoor for maintenance and test

- System access via rootkit with root administrator rights
  - Programs installed undercover to access OS
  - Persistent/in memory, user/kernel mode, VM based

Kernel Rootkit and VM

- Direct modification of the kernel and co-existence with the OS

  Low level presence and much more complicated detection

- Interaction with the kernel via system calls

  Changing the system call table or target

- Rootkit at the level of the VMM or the hypervisor

  Rootkit completely hidden from the kernel code of the target OS

Countermeasure

Countermeasure

- Development of anti-malware countermeasure

  Initially called anti-virus mechanism

- Best solution against malware threat is prevention

  Do not let in and block possibility of changing the system

- Four main elements of prevention

  Politics, awareness, vulnerability and threat mitigation

Technical Mechanism

- Three options to attenuate vulnerability
  - **Detection** of the infection and location of the malware
  - **Identification** of the type of the malware infecting the system
  - **Removing** of all traces of the malware
- Several constraints for a good anti-malware
  - As general as possible and react quickly to limit propagation
  - Resistance to malware evasion techniques
  - Minimising DoS countermeasure, maintaining normal operation
  - Transparency and no modification os OS, application, hardware

Host Scanner

- Anti-virus software installed on all end systems

  Maximum access to all malware information and activity

- Kind of host-based intrusion detection system (IDS)
  - Simple search for signatures and possibility of wildcards
  - Probable malware found with heuristic and integrity check
  - Activity trap actively scans system activities
  - Full-feature protection combines several techniques

Generic Decryption (GD)

- Difficulty to detect polymorphic viruses

  It must nevertheless be decrypted before execution

- Executable files go through a GD scanner
  - CPU emulator is a software virtual computer
  - Scanner of known virus/malwares signatures
  - Target code emulation control module

Host-Based Behaviour-Blocking

- Integration with the host system operating system
    - Real-time monitoring of program behaviour
    - Identification of malicious actions and possible block
- Several types of monitored behaviours
    - File opening, accessing, deleting, modifying
    - Disk format or other unrecoverable operations
    - System critical configuration modification
    - Sending e-mails, instant messages...
    - Initiation of a network communication

Fighting Rootkit

- Anti-rootkit administration tools can be compromised

  Rootkits are therefore very difficult to detect

- Using computer and network level tools
  - Identify rootkit attack signature in incoming traffic
  - Locate keylogger, interception of system calls

- Checking file integrity

  Being able to realise that the system has been modified

Perimeter Scan

- Anti-virus placed at the level of firewall or IDS

  Higher level supervision in the company

- Approach limited to the scan of the content of the malware
  - **Ingress** monitor: network company/internet border
  - **Egress** monitor: output of individual LANs

- Using external firewall orhoneypot

  To place monitoring software

Collective Intelligence

- Using a distributed configuration
  - Harvesting data from a large amount of sources
  - Both based on hosts than on perimeter sensors
- Central intelligent analysis system
  - Able to correlate and analyse data
  - Return signatures and behaviours pattern

Credits

- Faris Algosaibi, January 11, 2014, https://www.flickr.com/photos/siraf72/11885592144.
- Nancy Hoang, October 14, 2017, https://www.flickr.com/photos/nancyhoang/36982441854.
- NASA HQ PHOTO, August 5, 2011, https://www.flickr.com/photos/nasahqphoto/6012617390.
- Dane Erland, September 15, 2012, https://www.flickr.com/photos/lord_dane/7989074153.