



Computer Security and Cryptography

CS381

来学嘉

计算机科学与工程系 电院3-423室

34205440 1356 4100825 laix@sjtu.edu.cn

2016-05



Contents



- Introduction -- What is security?
 - Cryptography
 - Classical ciphers
 - Today's ciphers
 - Public-key cryptography
 - Hash functions/MAC
 - Authentication protocols
 - Applications
 - Digital certificates
 - Secure email
 - Internet security, e-banking
- Network security
 - SSL
 - IPSEC
 - Firewall
 - VPN
 - Computer security
 - Access control
 - Malware
 - DDos
 - Intrusion
 - Examples
 - Bitcoin
 - Hardware
 - Wireless



contents



- Malicious Software
- Virus Countermeasures
- Flame



Malicious software



- Malicious software requiring a host program

Viruses	Attaches itself to a program and propagates copies of itself to other programs
logic bombs	Triggers action when condition occurs
Backdoors	Program modification that allows unauthorized access to functionality

- Independent malicious software

Worm	Program that propagates copies of itself to other computers
Zombie	Program activated on an infected machine that is activated to launch attacks on other machines



Infection mechanisms



- Email
- P2P file sharing
- Embedding in data files like music, video, doc
- Remote exploitation of system/software vulnerability
- USB token
- System installation/update



Viruses



- a piece of **self-replicating** code attached to some other code (cf biological virus)
- as code to perform some covert **task**
- virus phases:
 - dormant – waiting on trigger event
 - propagation – replicating to programs/disks
 - triggering – by event to execute task
 - execution – of task



Virus Structure



```

program V :=
  {goto main;
  1234567;
  subroutine infect-executable := {loop:
    file := get-random-executable-file;
    if (first-line-of-file = 1234567) then goto loop
    else prepend V to file; }
  subroutine do-damage := {whatever damage is to be
  done}
  subroutine trigger-pulled := {return true if condition holds}
  main: main-program := {infect-executable;
    if trigger-pulled then do-damage;
    goto next;}
  next:
}
  
```

An infected version of a program is **longer** than the uninfected one.

Variation: compress the executable file so that both the infected and uninfected versions **are of identical length**



Macro Virus



- **macro code** attached to some **data file**
 - interpreted by program using file, eg Word/Excel macros
 - esp. using auto command & command macros
 - code is now platform independent
 - blur distinction between data and program files
 - classic trade-off: "ease of use" vs "security"
- **Email Virus** spread using email with **attachment containing a macro virus**, cf Melissa
 - triggered when user opens attachment
 - or worse even when mail viewed by using scripting features in mail agent
 - hence propagate very quickly



Logic Bomb



- one of oldest types of malicious software
- code embedded in legitimate program
- activated when specified conditions met
 - presence/absence of some file
 - particular date/time
 - particular user
- when triggered typically damage system
 - modify/delete files/disks, halt machine, etc
- Protection
 - Don't install software without testing it and checking it
 - Keep regular backups so that you can restore your data.



Trojan Horse



program with hidden side-effects

- which is usually superficially attractive
 - eg game, s/w upgrade etc, pictures
- when run performs some additional tasks
 - allows attacker to indirectly gain access they do not have directly
- often used to propagate a virus/worm or install a backdoor
- Key-logger; screen capture; or simply to destroy data
- Automatic update;



Backdoor or Trapdoor



- secret entry point into a program
- allows those who know access bypassing usual security procedures
- have been commonly used by developers
- a threat when left in production programs allowing exploited by attackers
- very hard to block in O/S
- requires good s/w development & update



Zombie



- program which secretly takes over another networked computer
- used to perform malicious tasks [under remote direction](#); e.g. launch distributed denial of service (DDoS) attacks
- exploits known flaws in network systems
- have been used extensively to send e-mail spam



Worms



- replicating but not infecting program
 - **Viruses** : **attach** themselves to other programs.
 - **Worms**, in contrast, are **stand-alone** programs that do not need to attach to other programs.
- Can propagate like viruses through e-mail, and so on
- Effect
 - Cause High Net Traffic, create **zombie**
 - Mischief/Spyware
- Spread
 - Over Networks
 - Actively
- major issue is lack of security of permanently connected systems, esp PC's



Example: Morris Worm



- First known worm - November 2, 1988
- Author - Robert Tappan Morris
- Infected BSD Unix systems
- Son of Robert Morris, the former chief scientist at the National Computer Security Center, a division of the National Security Agency (NSA).
- Morris received his Ph.D. in computer science from Harvard University in 1999 and is a professor at MIT.
- Robert Morris is the first person convicted under the 1986 Computer Fraud and Abuse Act

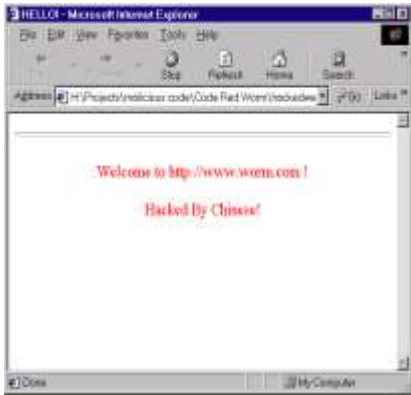




Example: The Code Red Worm(2001)




- attacked computers running Microsoft's IIS web server
- Spread **rapidly**: > 2,000 hosts/min
- Evaded automated detection
 - Detectable more easily by humans than scanners
 - Resident only in memory, no disk writes
- Defaced home page of infected server



熊猫烧香 (2006)



- “熊猫烧香”是一个由Delphi编写的蠕虫，终止反病毒软件和防火墙软件进程。
- 病毒删除扩展名为gho的文件，使用户无法使用ghost软件恢复操作系统
- “熊猫烧香”感染系统的.exe、.com、.pif、.src、.html、.asp文件
- 添加病毒网址，导致用户一打开这些网页文件，IE就会自动连接到指定的病毒网址中下载病毒。
- 在硬盘分区生成autorun.inf和setup.exe，可通过U盘/移动硬盘传播，利用Windows自动播放功能运行，搜索硬盘中的.exe可执行文件并感染，
- 感染后文件图标变成“熊猫烧香”图案。“熊猫还可以通过共享文件夹、系统弱口令等多种方式进行传播



Virus Countermeasures



- best countermeasure is prevention, but in general not possible
- hence need to do one or more of:
 - **detection** - of viruses in infected system
 - **identification** - of specific infecting virus
 - **removeal** - restoring system to clean state



Anti-Virus Software



- **first-generation**
 - scanner uses virus signature to identify virus
 - or change in length of programs
- **second-generation**
 - uses heuristic rules to spot viral infection
 - or uses crypto hash of program to spot changes
- **third-generation**
 - memory-resident programs identify virus by actions
- **fourth-generation**
 - packages with a variety of antivirus techniques
 - eg scanning & activity traps, access-controls



Advanced Anti-Virus Techniques



- generic decryption
 - use CPU simulator to check program signature & behavior before actually running it
 - digital immune system (IBM)
 - general purpose emulation & virus detection
 - any virus entering org is captured, analyzed, detection/shielding created for it, removed
- Behavior-Blocking Software
 - monitors program behavior in **real-time** for malicious actions
 - **blocks** potentially malicious actions before they have a chance to affect the system
 - **Drawback**: the malicious code must actually run on the target machine **before** all its behaviors can be identified



APT



- **APT-Advanced Persistent Threat**: usually refers to a group, with both the capability and the intent to **persistently** and effectively target a **specific entity**.
- long-term pattern of sophisticated hacking attacks
- **Advanced** -combine multiple targeting methods, e.g., telephone-interception, satellite imaging, travel records, hobby, shopping, GPS, GSM,...



- **Persistent**- the attackers are guided by external entities, through continuous monitoring and interaction in order to achieve the defined objectives. “low-and-slow”
- **Threat** –have a specific objective by coordinated human actions, and are **skilled, motivated, organized and well funded**.
- 典型例： Stuxnet, Duqu, and **Flame**



Stuxnet - 1



- **Stuxnet**: a computer worm discovered in June 2010.
- **target only Siemens (SCADA) systems for control and monitor specific industrial processes**
 - Stuxnet reportedly ruined almost one-fifth of Iran's nuclear centrifuges
 - On 1 June 2012, an article in *The New York Times* said that Stuxnet is part of a U.S. and Israeli intelligence operation called "Operation Olympic Games", started under President Bush and expanded under President Obama.
 - only attacks systems that spin between 807 Hz and 1210 Hz, periodically modifies the frequency to 1410 Hz and then to 2 Hz and then to 1064 Hz
 - a **man-in-the-middle** attack, report **normal operation** to user
 - Stuxnet has a valid, but abused digital signature signed with a stolen key from C-MEDIA, Taipei



Stuxnet - 2



- Stuxnet has three modules:
 - a **worm** that executes the attack;
 - a **link file** that auto-executes the copies of the worm;
 - a **rootkit** for hiding all malicious files and processes, preventing detection of the presence of Stuxnet.^[6]
- Stuxnet spreads by infected **USB flash drive**.
 - The virus then propagates across the network,
 - scanning for Siemens Step7 software on computers controlling a PLC (Programmable Logic Controller) .
 - Then introduces the rootkit, modifying the codes, giving commands to the PLC
 - while returning a loop of normal operations system values back to the users

23



Duqu



- **Duqu** was found on 1 September 2011,
- it “nearly identical to Stuxnet, but with a completely different purpose”, is designed to capture information, such as keystrokes and system information.
- Stuxnet and Duqu both were built on, originated in 2007.
- has a valid, but abused digital signature signed with a **stolen key** from C-MEDIA, Taipei

24



Flame



- **Flame**, also known as **sKyWlper**
- identified in **May 2012** by MAHER Center of Iranian CERT,
- infected approximately 1,000 machines, with victims including governmental organizations, educational institutions and private individuals. countries most affected were Iran, Israel, Sudan, Syria, Lebanon, Saudi Arabia, and Egypt, with a "huge majority of targets" within Iran.
- Flame is an uncharacteristically large program for malware at **20 megabytes**
- Flame supports a "kill" command which wipes all traces, stopped operating after exposure, and the "kill" command was sent
- MS issued certificate update in **June 2012**.

25



Naming



```
FROG.Payloads.ServiceBuffer
start /wait RunDll32.exe %windir%\temp\~ZFF042.ocx,DDEnum
del /q %windir%\temp\~ZFF042.ocxJ
FROG.Payloads.Flame0InstallationBat
InstallFlame
FROG.DefaultAttacks.A InstallFlame Description
AGENT
FROG.DefaultAttacks.A InstallFlame AgentIdentifier
FROG.DefaultAttacks.A InstallFlame ShouldRunCMD
T<&
%temp%\fib32.bat
FROG.DefaultAttacks.A InstallFlame CommandLine
FROG.DefaultAttacks.A InstallFlame ServiceTimeout
FROG.DefaultAttacks.A InstallFlame AttackTimeOut
FROG.DefaultAttacks.A InstallFlame DeleteServicePayload
FROG.DefaultAttacks.A InstallFlame DeleteUploadedFiles
FROG.DefaultAttacks.A InstallFlame SampleInterval
FROG.DefaultAttacks.A InstallFlame MaxRetries
FROG.DefaultAttacks.A InstallFlame RetriesLeft
FROG.DefaultAttacks.A InstallFlame TTL
FROG.DefaultAttacks.A InstallFlame HomeID
FROG.DefaultAttacks.A InstallFlame FilesToUpload.size
```



Targets



Flame's Modules



Name	Description
Flame	Modules that perform attack functions
Boost	Information gathering modules
Flask	A type of attack module
Jimmy	A type of attack module
Munch	Installation and propagation modules
Snack	Local propagation modules
Spotter	Scanning modules
Transport	Replication modules
Euphoria	File leaking modules
Headache	Attack parameters or properties
...(≈ 20)	



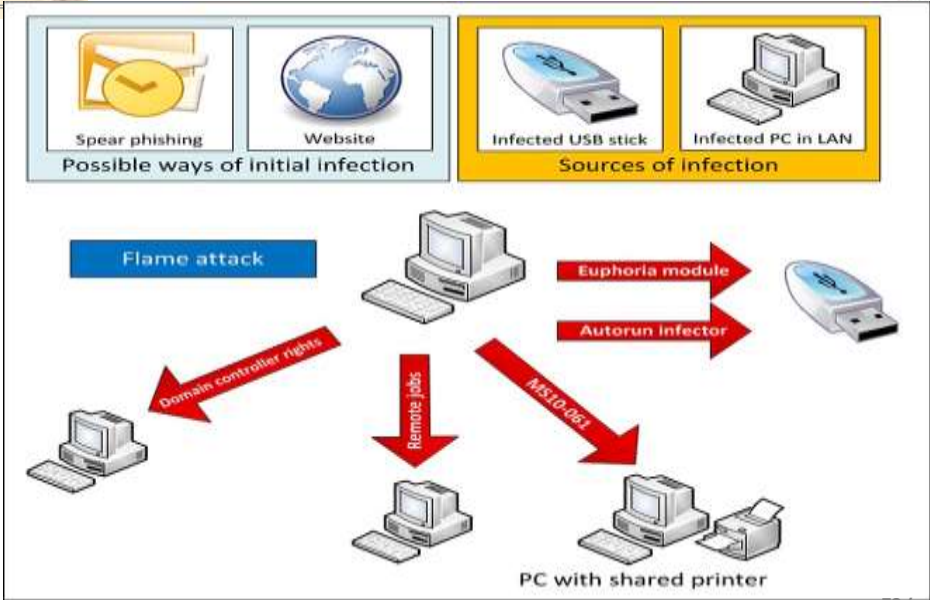
Flame -espionage



- Flame spreads over a local network (LAN) or via USB stick.
- It can record audio, screenshots, keyboard activity and network traffic, also records Skype conversations
- Can turn infected computers into Bluetooth beacons which attempt to download contact information from nearby Bluetooth-enabled devices. This data, along with locally stored documents, is sent on to one of several command and control servers that are scattered around the world. The program then awaits further instructions from these servers
- Flame appears to have been written purely for espionage



Infection



507



Flame Abstract



Date: 28 May 2012(discovery) **Area:** Middle-East

OS: Windows

Language: Lua & C++

Size: ≈20 MB(zlib, libbz2, ppmd, sqlite3, vm)

Functions: Espionage

...

**Backdoor, Trojan,
Worm...**

31 / 59



Flame Operations



- Five different encryption methods;
- SQLite;
- Process inject;
- Memory pages are protected;
- Exploits two of the same security vulnerabilities used by **Stuxnet**;
- Antivirus software adaption;
- **Signed with a fraudulent certificate.**

” ...some components of the malware have been signed by certificates that allow software to appear as if it was produced by Microsoft...”

32 / 59



counterfeit certificate



- Flame was signed with a **fraudulent certificate** from the Microsoft Enforced Licensing Intermediate PCA certificate authority.
- a Microsoft Terminal Server Licensing Service certificate that still used the weak **MD5 hashing** algorithm,
- produced a counterfeit certificate that was used to **sign malware** to make them appear to have originated from Microsoft.
- A successful collision attack against a certificate was previously demonstrated in 2008, but Flame implemented a **new variation of the chosen-prefix collision attack**

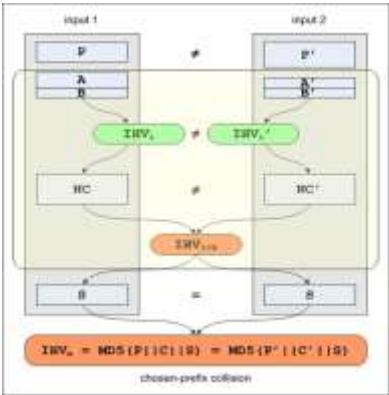
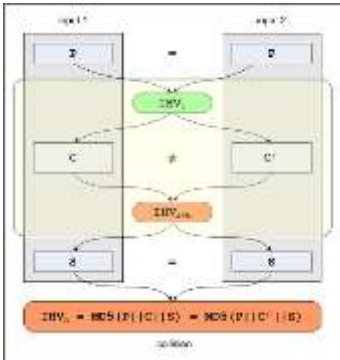
33



MD5 collision – chosen-prefix collision

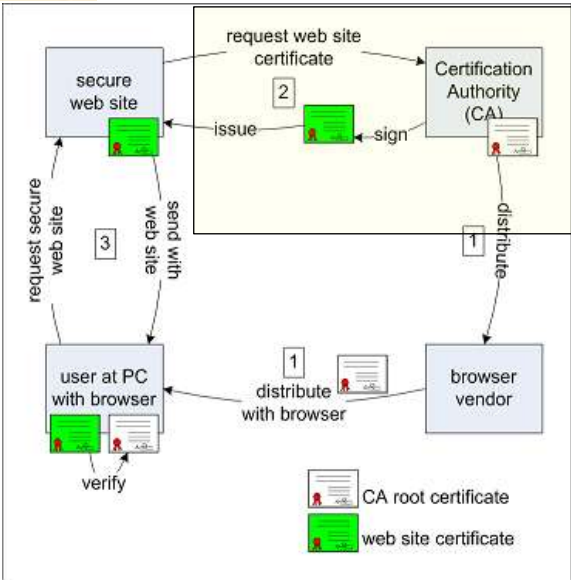


- “rogue certificates” [M. Stevens, <http://eprint.iacr.org/2009/111>]
 - 2 certificates with different data fields (especially CA=TRUE/FALSE) and public-keys, but with same MD5 hash code.
 - free-start collision: $\text{comp.} = 2^{16}$





Normal use of certificates



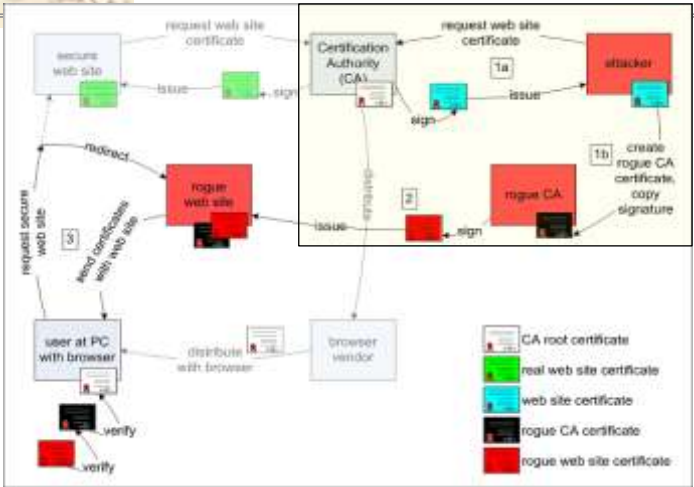
证书验证

- CA根证书置于浏览器中
- CA签发网站证书
- 用户验证网站证书的根CA签名
- 普通网站证书不能用于签发下级证书

37



rouge certificates



- 伪CA证书与普通网站证书有相同的签名（hash 碰撞）
- 伪CA证书可签发新的伪证书（网站，软件）
- 伪证书可通过MS浏览器验证

38



问题出在哪里？



- **SSL**安全性建立在对**浏览器的信任**上
- 信任关系通过**证书**传递
- 证书的安全要求安全（抗碰撞）的**Hash**函数
- **MD5碰撞**--> 深入研究 --> 构造**伪证书**
- 用伪证书签发的软件（**Flame**）可通过**IE**验证



勒索软件（RansomWare）



- Ransomware lock your computer by encrypting your files
- and demand money to be paid to get back to access

Protection:

- Don't click unknown things
- Back up your files





contents



- Summary
 - Malicious Software
 - Virus Countermeasures
 - Flame
- Next
 - DDoS
 - Intrusion



Exercise 17



- Describe the similarities and differences of Virus, Macro virus and Worms
- List the good rules to avoid malware infection.
- How is the hash collision attack used in the Flame malware?
- Deadline: before next lecture