Malicious Code

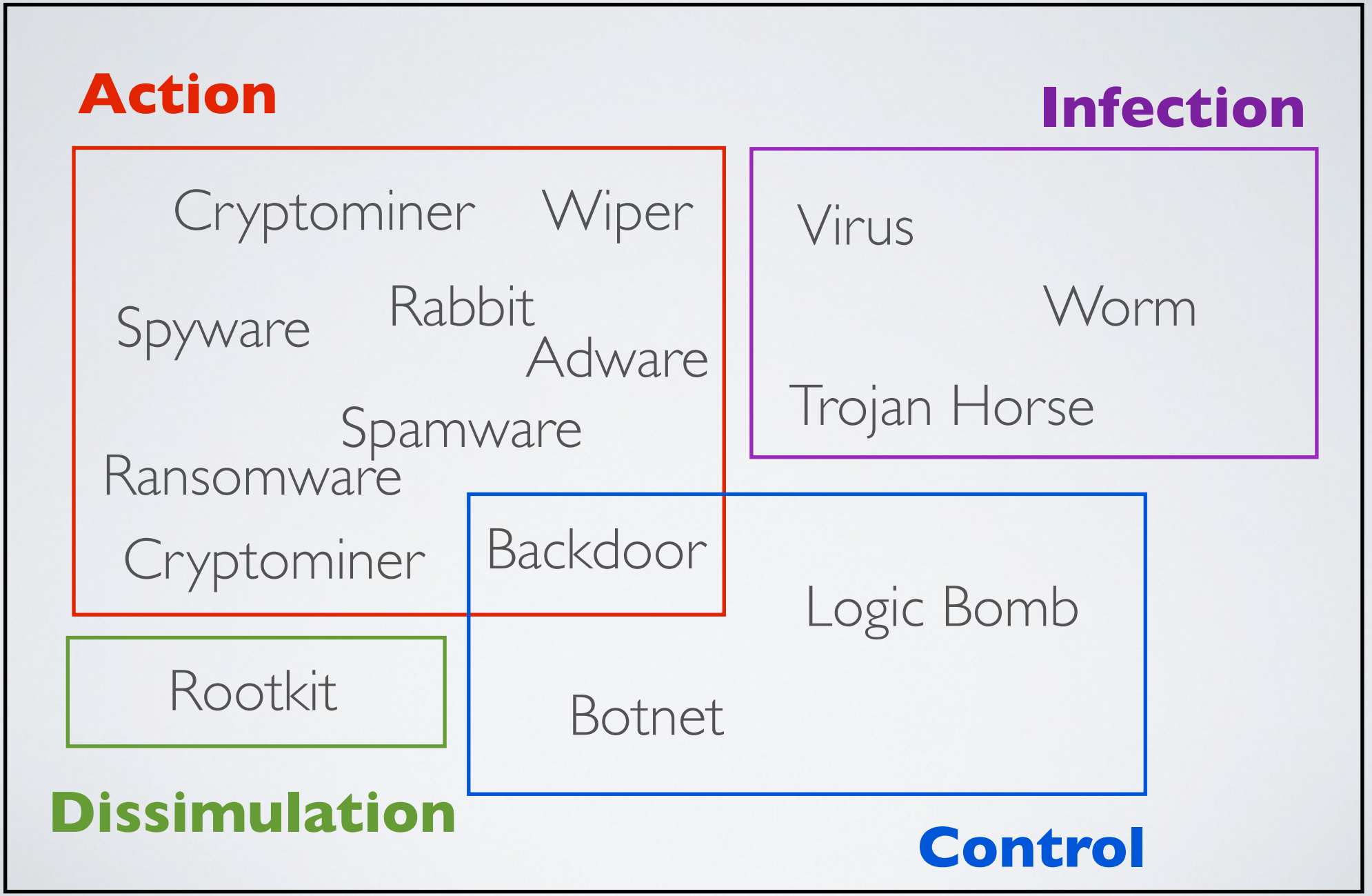
Thierry Sans

Outline

- Malware Taxonomy
- The Evolution of Malware Through History
- Anatomy of Modern Malware
  - **Payload** - What the malware does?
  - **Infection Vector** - how malware infects the system?
  - **Persistence Mechanisms** - how malware sticks to the system?
  - **Exfiltration and Control** - How the malware can be exfiltrate data and/or be controlled remotely?
  - **Evasion Technique** - How the malware stays undetectable and/or hard to analyze?
- [Bonus] A Malware Story (2014)

Malware Taxonomy

Malware


Action
Infection
Cryptominer
Wiper
Virus
Rabbit
Spyware
Worm
Adware
Trojan Horse
Spamware
Ransomware
Backdoor
Cryptominer
Logic Bomb
Rootkit
Botnet
Dissimulation
Control

Action

- performs unsolicited operations on the system

- **Rabbit** exhausts the hardware resources of a system until failure
- **Wiper** destroys data and take down services
- **Backdoor** allows an attacker to take control of the system bypassing authorization mechanisms
- **Spyware** collects information
- **Spamware** uses the system to send spam
- **Ransomware** restricts access to system's data and resources and demands for a ransom
- **Cryptominer** runs crypto-mining bots
- **Adware** renders unsolicited advertisement

Dissimulation

- avoid detection by anti-malware programs

Rootkit hides the existence of malicious activities

Infection

- penetrate a system and spread to others

Replication

- copy itself to spread

- **Virus** contaminates existing executable programs
- **Worm** exploits a service's vulnerability

Subterfuge

- based on user's credulity

- **Trojan Horse** tricks the user to execute the malicious code

Control

- activate the malicious code

- **Logic Bomb** activates the malicious code when certain conditions are met on the system
- **Backdoor** communicates with command & control servers allowing an attacker to control the virus
- **Botnet** is a network of compromised systems that can be controlled by a single attacker

The Evolution of Malware Through History

Chronology

- 70's - The era of the first self-replicating programs
- 80's - The era of maturity and first pandemics
- 90's - The era of self-modifying virus and macro viruses
- 00's - The era of Trojan horses and internet worms
- 10's - The era of cyber-warfare malware and ransomware
- 20's - The era of AI malware?

70's - The era of the first self-replicating programs

Context : the era of big computer mainframes

The era of the first self-replicating programs (70's)

ANIMAL (a popular game)

- Replication through the filesystem
- No effect

Simple Joke

Creeper (and **Reaper**) on Tenex OS (Arpanet)

- Replication through a modem and copied itself to the remote system
- Displays the message
  `I'M THE CREEPER : CATCH ME IF YOU CAN`

Disruptive

The **Rabbit** program

- Replication through the filesystem
- Reduces system performance till crashing

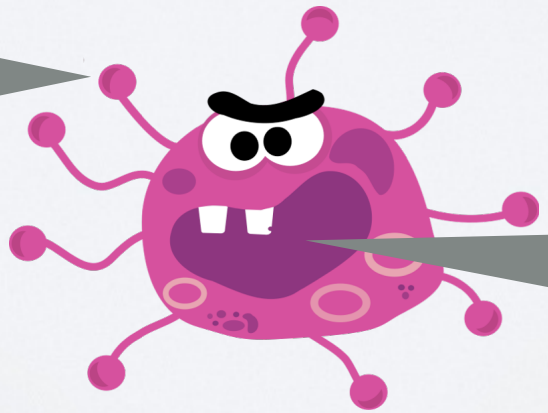
Destructive

Anatomy of a Virus

A **virus** can be

- a malicious code embedded in an existing program and replicates itself by infecting other programs through the filesystem or the network
- a program that exists by itself and replicates through the filesystem or network

Infection vector
how the virus penetrate the system



The payload
what the virus does

80's - The era of maturity and first pandemics

Context : the era of personal computers

1987 - the beginning of pandemics

Jerusalem (MS-DOS)

- Destroys all executable files on infected machines upon every occurrence of Friday the 13th

SCA (Amiga)

- Displays a text every 15th boot
- 40% of the Amiga owners were infected

Christmas Tree EXEC (IBM/PC)

- Displays a snow flow animation
- Paralyzed several international computer networks in December 1987

The first anti-virus softwares (end of 80's)

Virus scanner (detection)

- Signature based
  Using a signature database of existing viruses
- Behavior based
  Looking for suspicious code patterns that can be used by viruses

Virus removal tools (sanitation)

- Cleaning the memory and the filesystem

Avoiding detection

Cascade (1987)

- The virus encrypts itself with a cryptographic key and changes this key when replicating itself

✓ Each instance of the virus does not look the same

➡ This is the emergence of polymorphic viruses

90's - The era of self-modifying virus and macros viruses

Context : the era of gaming and office applications

The era of self-modifying virus (90's)

The **Chameleon** family (1990)

Ply (1996)

- DOS 16-bit based complicated polymorphic virus with built-in permutation engine

Anatomy of a "polymorphic" virus

A **polymorphic virus** mutates when replicating (but keeps the original algorithm intact)

- By using cryptography
- By injecting garbage code
- By doing permutations within certain instructions or block of instructions
- By using code obfuscation technique

How to detect it?

➡ By detecting code patterns used for the self-modification

Macro Viruses

A **macro virus** is written in scripting languages used by some office applications (can be cross-platform)

- Written in VBS, embedded in a MS-office document, activated when the document is open (autoload function)

Concept (1995)

Melissa (1999)

- March 26 1999, Melissa shut down e-mail systems that got clogged with infected e-mails

00's - The era of Trojan horses and internet worms

Context : the era of internet adoption by the public

Anatomy of a Trojan horse

A **Trojan horse** is a program that disguise itself as a legitimate program or file

➡ In most cases, Trojan horses replicate themselves through emails

The big stars among trojan horses

VBS/Loveletter ILOVEYOU (2000)

- Caused 5.5 to 10 billion dollars in damage

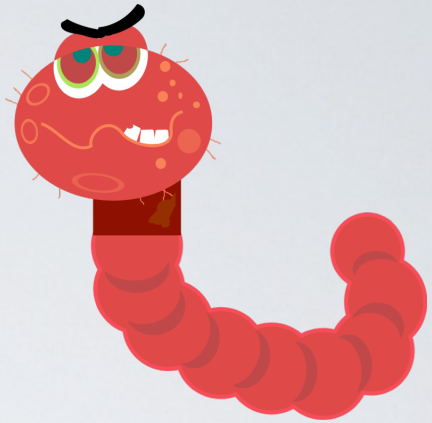
Sobig (2002)

- Sobig.F set a record in sheer volume of e-mails

MyDoom (2002)

- Broke the record set by Sobig.F

Anatomy of a worm



A **worm** exploits a security flaw (often of a network service) to infect the machine and replicates itself through the network

- ➡ Very fast infection (does not need the user to be activated)
- ➡ Has a payload as well (more or less harmful)

Factors

- The wide adoption of internet
- The global network is a good medium for virus pandemics
- The multiplication of internet applications and services
- Fast publication of program vulnerabilities
- Slow release of corrective patches
- Slower adoption of these patches (not automatic)

Code-Red (2001)

- Exploits a security flaw (buffer overflow) of Microsoft IIS web server (MS01-033) patched one month earlier
- In few days, 359 000 machines infected

Nimda (2001)

- Exploits another security flaw of MS-IIS
- The Internet's most widespread worm so far (the most part of the infection was done in 22min)

Klez (2001)

- Exploits a security flaw of Microsoft Internet Explorer layout engine used by Outlook and IE
- Infection through email attachment however the user does not have to open this attachment to get infected

SQL-Slammer (also called **Sapphire**) (2002)

- Exploits a security flaw in MS-SQL servers for which a patch had been released six months earlier (MS02-039)
- Infected 75,000 machines in 10 minutes causing caused a massive denial of service and dramatically slowed down global Internet traffic

Sasser (2002)

- Exploiting a buffer overflow of Microsoft LSASS on Windows 2000 and XP systems
- Many companies had to shut down their services

Blaster (also known as **Lovesan**) (2003)

- Exploits a security flaw in DCOM-RPC services on Windows 2000 and XP
- Was supposed to do SYN flood on August 15, 2003 against port 80 of windowsupdate.com

Welchia (also known as **Nachia**) (2003)

- Exploits the same security flaw than Blaster
- Corrects the security flaw by patching the system

Conficker (2008)

- Exploits a security flaw in NetBIOS
- Disables auto-update
- Embeds a dictionary password cracker and a backdoor to turn the machine into a "bot"
- Believed to be originated from Ukraine and/or Russia

10's - The era of cyber-warfare malware & Ransomware

Context : the era of cloud computing, bitcoin and connected physical infrastructure

The first cyber-warfare virus

W32.Dozor (2009)

- A virus that created a botnet dedicated to perform a DDoS attack South Korea and US government website on July 4th
- Believed to be originated from China and/or North Korea

The awakening

Stuxnet (2010)

- A very sophisticated virus that targets SCADA systems (supervisory control and data acquisition)
- Believed that it took down 4000 nuclear centrifuges in Iran
- Believed to be originated from the USA and Israel

Flame also called **Skywiper** (2012)

- An *espionage* virus that embeds sophisticated spywares
- Believed to be originated from the USA (*Olympic Games* defense program)

Espionage

Sunbust used in *SolarWinds* attack (2020)

- Considered one of the most sophisticated supply-chain intrusions
- Impacted US government agencies and Fortune 500 companies
- Believed to be originated from China

Volt Typhoon (2023)

- Targeted US critical infrastructure: telecom, energy, water, transportation
- Believed to be originated from China

Sabotage

Shamoon (2012)

- Targeted oil and gas infrastructure in Qatar and Saudi Arabia
- Destroyed 30,000 machines

CrashOverride (2015)

- Targeted the Ukrainian Power Grid
- First malware to turn off electricity at scale

Another trend - Ransomware

Reveton (2012)

- Displays a message from the law enforcement agency saying that you have pirated software and child pornography on your machine
- Ask you to pay a fine using a prepaid cash service

CryptoLocker (2013)

- Encrypt specific files on your machine with a 2048 RSA key
- Ask you to pay a ransom with Bitcoins

"Ransomware attacks grew by 500% in 2013 and turned vicious"

source : *Symantec Internet Security Threat Report 2014*

... and it turned vicious

WannaCry and **NotPetya** (2017)

- Use the *EternalBlue* vulnerability found in the NSA hacking toolkit leak
- Researcher has found a "kill switch"
- Paralyzed hospitals in UK and trains in Germany
- Caused $10B+ damage, largest cyber-incident to date

LockBit 3.0 (2024)

- Advanced RaaS (Ransomware as a Service)

Late 10's - the emergence of IoT malware and Cryptominers

Mirai (2016)

- Infects IoT devices
- Most powerful DDoS attacks to date

Coinhive (2018)

- Javascript embedded in website (either legitimately or not) and popular malware as well

Crypto Drainers (2023)

- Target custodial and non-custodial crypto wallets
- Uses Telegram and Discord bots for C2

20's - The era of AI malware

Context : the era of adoption of generative AI by the public

Using Generative AI

WormGPT (2024)

- Use AI to build its own payload

Promptflux malware family (2025)

- Use AI to rewrite itself and evade detection

Anatomy of Modern Malware

Type of adversaries

Cybercrime-as-a-Service

- The goal is to compromise machines at a large scale, build a botnet and rent such capability

Cybercrime Groups

- The goal is to monetize attacks using ransomware, cryptominers and infostealers

Nation-State Groups

- The goal is to infect targeted systems and remain undetected
- Often advanced and coordinated attacks

Payload - What the malware do?

Backdoor

- Allows the attacker to take control of a system

Wiper

- Destroys data and take down services

Ransomware

- Encrypts data and ask for a ransom paid in crypto

Cryptominer

- Runs crypto-mining bots

Spyware (including **keyloggers** and **infostealers**)

- Key logging, credential harvesting, file stealer, screen & camera capture, browsing monitoring, geo tracking,

Infection Vector

How the malware infects the system?

Social Engineering (the most common vector)

- Trick or convince the user to install the malware on the system

Credential Stuffing

- Use stolen credentials to get legitimate access to systems

Exploit

- Exploit a vulnerability on the system

Macros

- Embed malicious code into office documents

Supply Chain Attack

- Infect software libraries that will be embedded with software

Persistence Mechanism

How malware sticks to the system?

- Depends on the targeted system
  - **Windows** : registry keys, scheduled tasks, *DLL* hijacking, system services
  - **Linux** : cron jobs, *systemd* services, `LD_PRELOAD` tricks
  - **MacOS** : *LaunchAgents*

Exfiltration and Control

How the malware can be exfiltrate data and/or be controlled remotely?

You either need

- **Data Exfiltration Channel** for a spyware (unidirectional)
- **Command & Control (C2)** for a bot (bidirectional)

- The goal is to prevent traffic from being detected
  - HTTP, SMTP, DNS tunneling
  - SSH tunnels, TOR
  - Remote drive, *Github*, *PasteBin*, *Twitter*, *Youtube*, *Reddit*
  - *Slack* Bots, *Telegram* bots, *Discord* WebHooks
  - Web3 (blockchain)

How to analyze malware?

➡ Understand what a program does by looking at system calls, library calls, IO operations and so on

Two complementary approaches:

1. **Static Analysis**
   Analyze binary code and infer its behavior (control flow and data flow approaches)
2. **Dynamic Analysis**
   Run program in a sandbox and observe its behavior

How to automate malware analysis?

➡ Anti-malware tools combine static and dynamic analysis but use different approaches to make a decision

- **Signatures**
  Detect based on signatures database (known patterns)
  See *Yara* rules: https://yara.readthedocs.io/en/latest/

- **Machine learning**
  Detection based on similarities with a collection of known malware

Evasion Technique - How the malware stays undetectable and/or hard to analyze?

Living-Off-The-Land (LOLbins)

- Reuse legitimate tools for payload, exfiltration and C2

Malware Packing

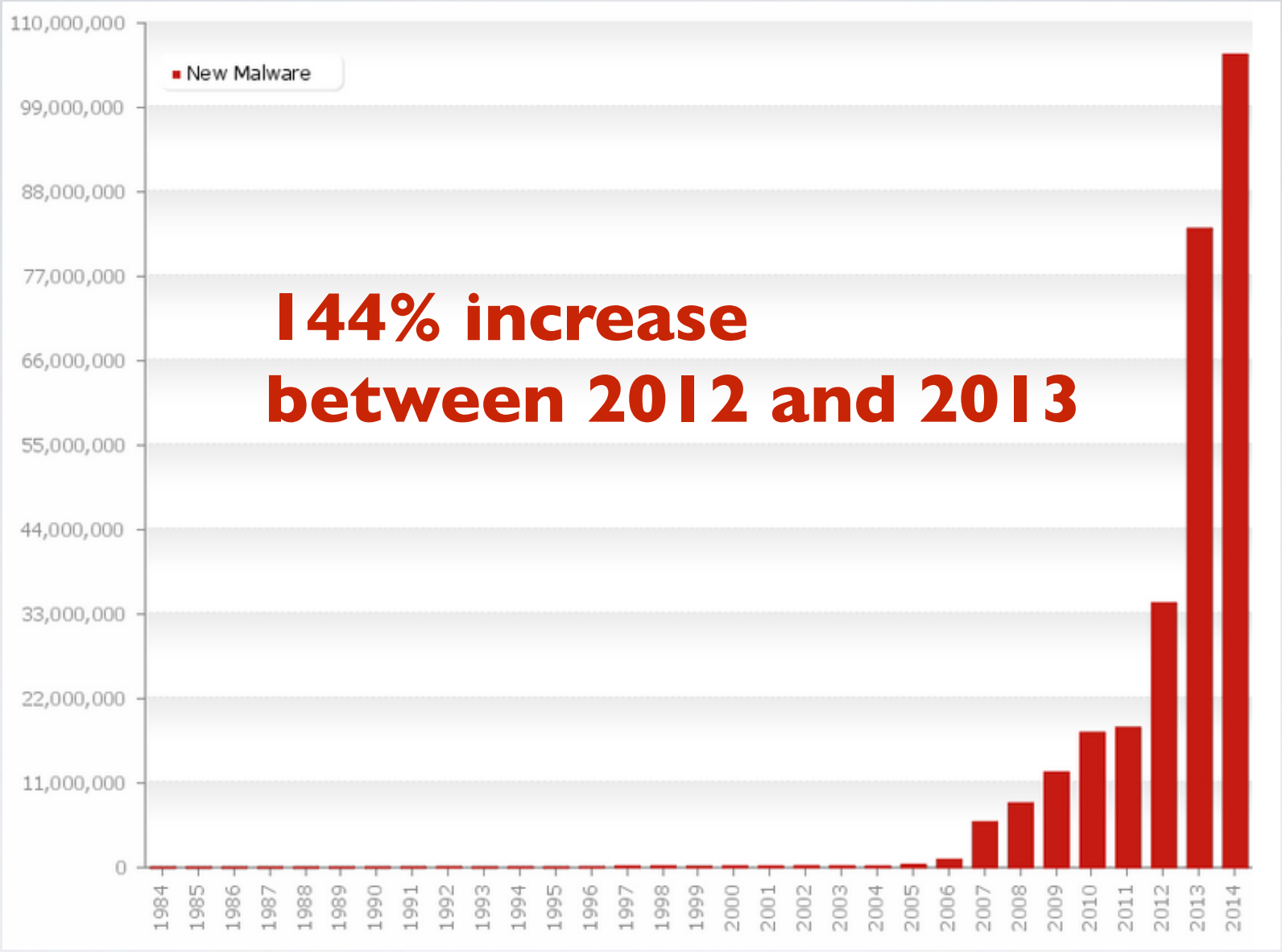
➡ The goal is to evade common detection techniques

- Encryption
- Code obfuscation
- Rewrite engines
- Stealth mode to detect and bypass VMs and sandboxes

Generative AI (newest trend)

- Use AI to dynamically generate payload or rewrite itself

[Bonus] A Malware Story (2014)



The Explosion of Unknown Malware

AV-TEST Institute

av-test.org

Why?

“Malicious Software and its Underground Economy”

joint work with *Omar Abou Selo* (undergrad at CMU) in 2014

Original research problem

➡ how easy is it to hire a hacker or get cutting-edge hacking tools on the internet (hacker’s forums)?

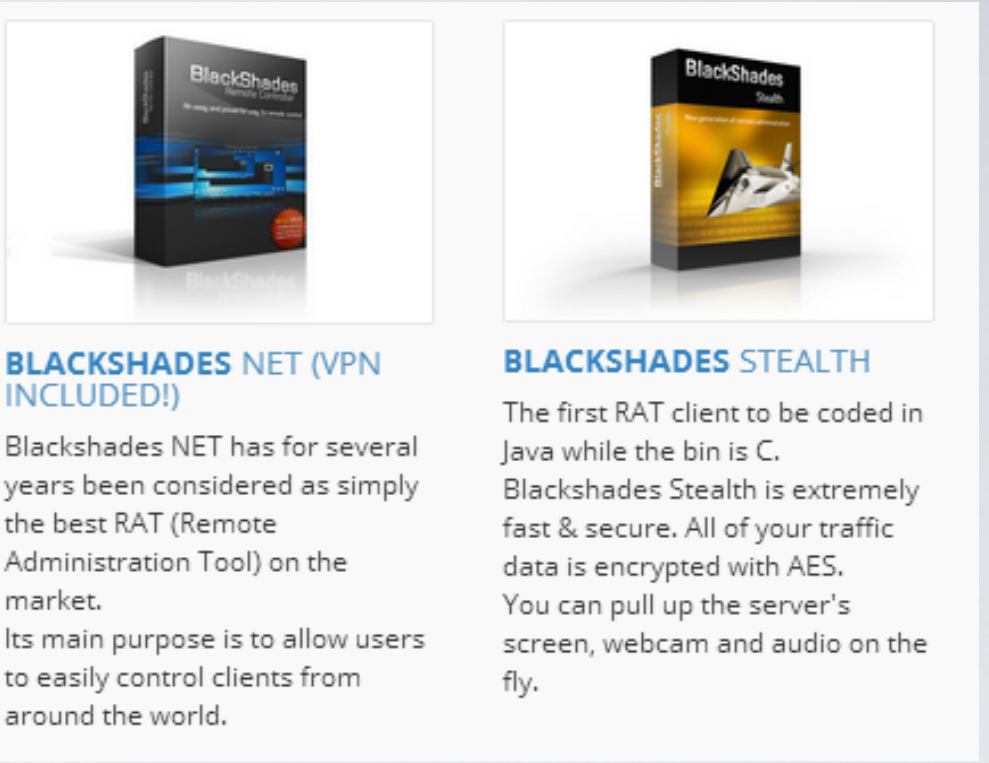
Conclusion

➡ creating a new malware is as simple as assembling pieces available online

How to create a new malware? 3 step process

1. **Create the malware's payload**
2. Make the malware undetectable
3. Spread the malware

Buy a RAT as a COTS*

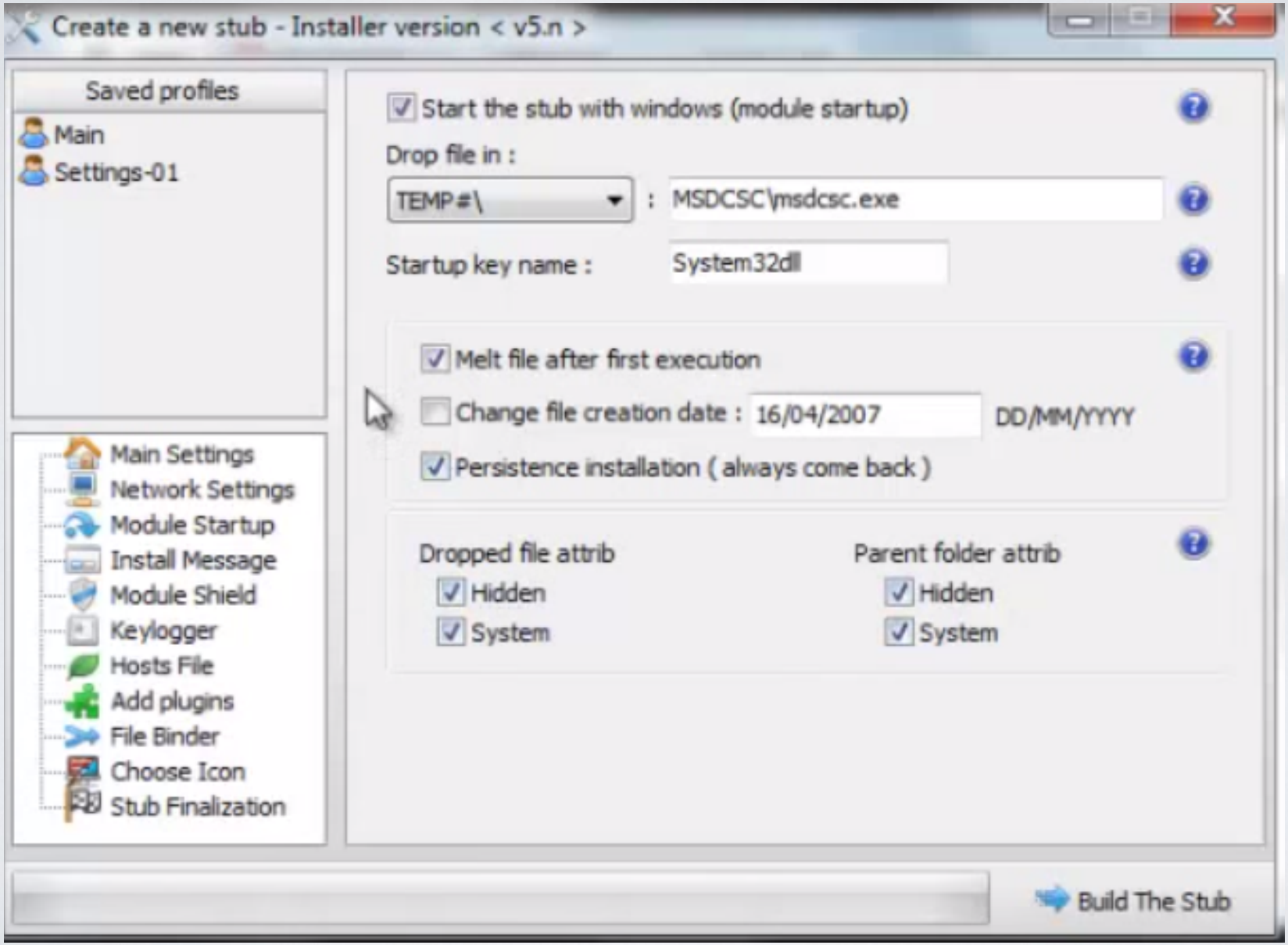


Some RAT Builders

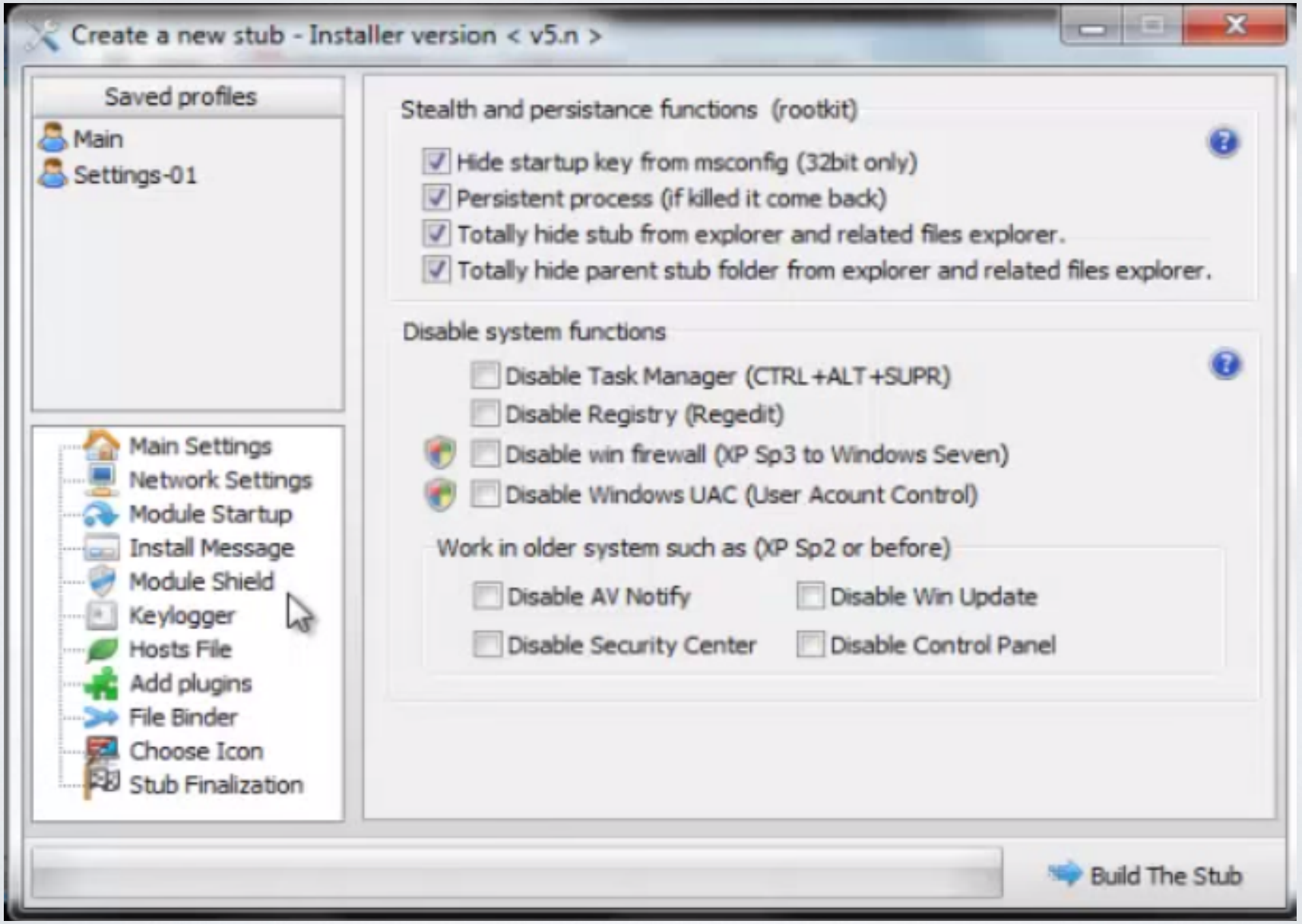
- **Zeus** (2007) initially $700, now open source
- **DarkComet** (2008), open source
- **BlackShades** (2010) can now be purchased from an official company $49 - $56

* Commercial Off-The-Shelf

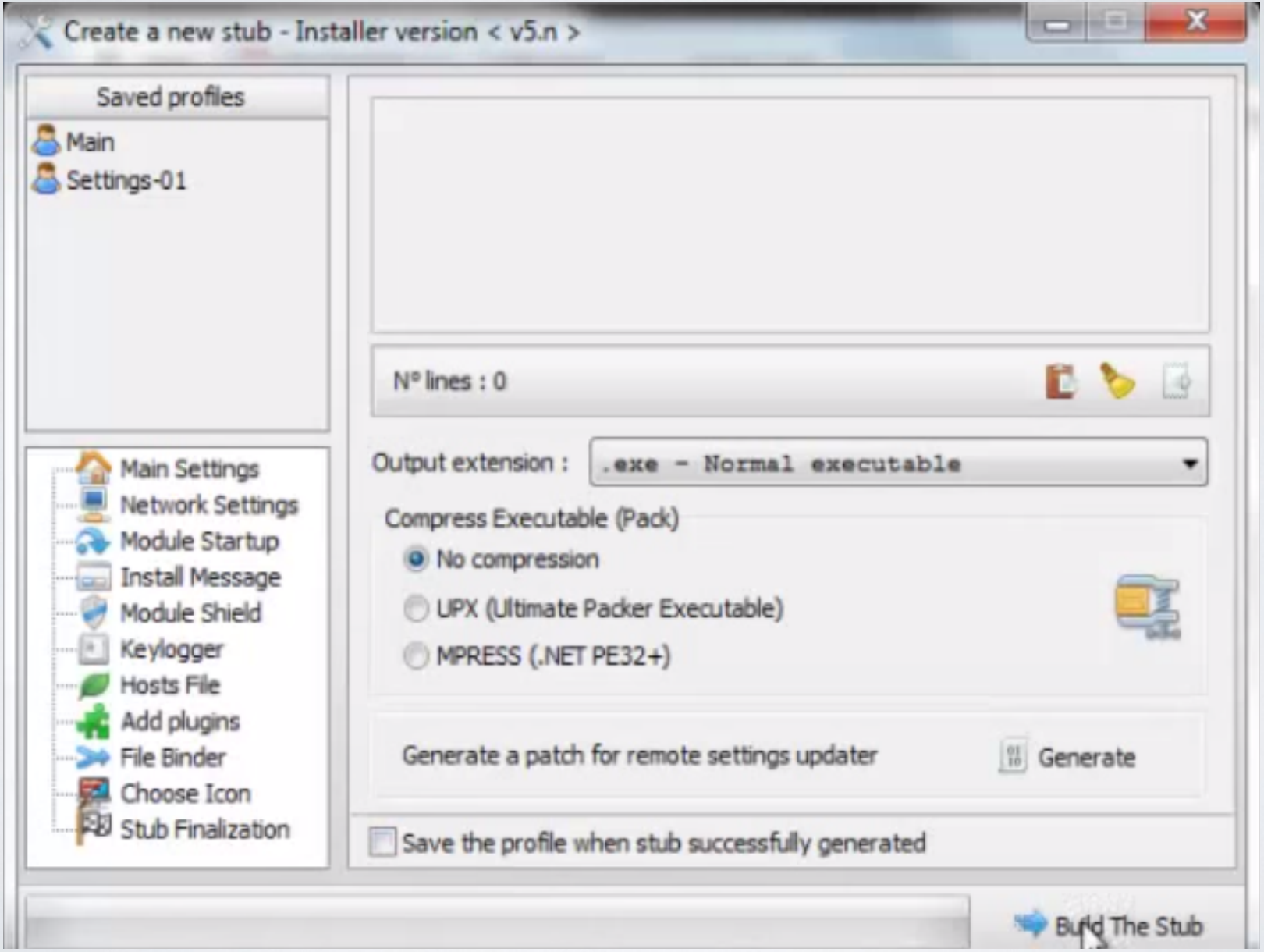
Startup and file options



Stealth and persistence options



Finally building the RAT



Are we done yet?

virustotal

SHA256: 858fb1fc03614802aee5be779b454b16384a1c051f925dbb360e8fcfa12fc6a3

File name: DarkCometRAT531.zip

Detection ratio: 42 / 50

Analysis date: 2014-04-27 11:58:13 UTC (5 hours, 26 minutes ago)

Analysis | Relationships | Additional information | Comments 1 | Votes

Antivirus	Result	Update
AVG	Generic23.AVVP.dropper	20140427
Ad-Aware	Trojan.Generic.KDV.388330	20140427
Agnitum	HackTool.Binder!uc8D13KnW4U	20140427
AntiVir	SPR/Binder.bs.1	20140426
Antiy-AVL	HackTool/Win32.Binder	20140427
Avast	Win32:Malware-gen	20140427

How to create a new malware? 3 step process

1. Create the malware's payload

2. **Make the malware undetectable a.k.a packing a malware**

3. Spread the malware

Buy a Crypter as a COTS

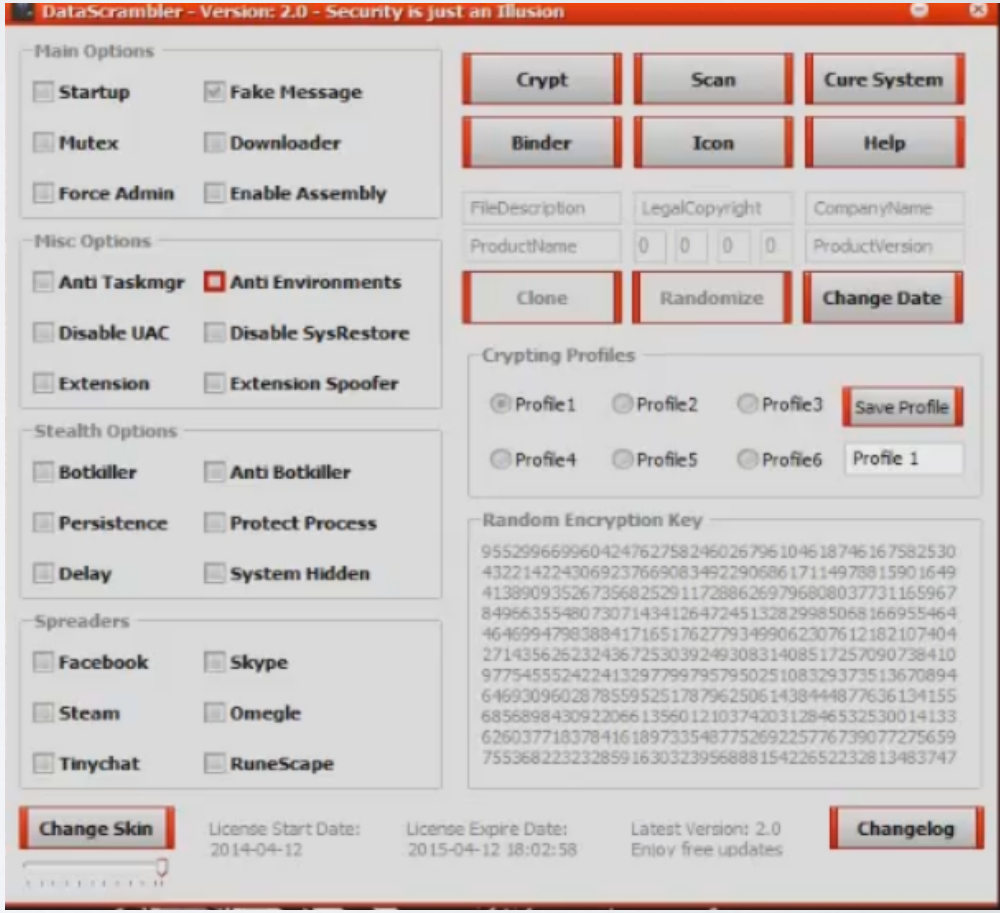
Some available crypters

- **Byte Crypter** $35 for 3 months, $60 for lifetime
- **Datascrambler** $20 for 3 months, $40 for a year
- **BlackShades Crypter** from an official company $60 for 3 months, $100 for a year

A look at Datascrambler

Functionalities include:

- Start malware on startup
- Block sandbox from monitoring
- Kill other bots on victims pc
- Protect from botkiller
- Delay for dynamic analysis
- Persistence
- Binder



How to create a new malware? 3 step process

1. Create the malware's payload
2. Make the malware undetectable
3. **Spread the malware**

Exploits bundle and other services

1. **Exploit bundle** : $25/day, $400/month, up to $3,000
   ➡ program to embed into a webpage

2. **Bulletproof host** : $15–250 per month
   ➡ hosting service to bypass any kind of IP filtering
   anti-spam, anti-virus, anti-malware, law enforcement,
   search engine anti-malware service and so on

3. **Traffic** : $4–10 per 1,000 unique hits
   ➡ attract people to visit the infected webpage

Installs $12 – $550 per 1000 infections
➡ use a spreading service also called Pay-Per-Install (PPI)

Examples of Exploits Kits

http://contagiodump.blogspot.com/2010/06/overview-of-exploit-packs-update.html

- **Blackhole** (2010, latest version in 2013)
  19 CVEs mainly targeting Java and Adobe products
  http://community.websense.com/blogs/securitylabs/pages/black-hole-exploit-kit.aspx

- **Redkit** (2013)
  4 CVEs mainly targeting Java
  http://nakedsecurity.sophos.com/2013/05/03/lifting-the-lid-on-the-redkit-exploit-kit-part-1/

Conclusion

Creating a malware, making it undetectable and spreading it would normally be difficult and require a good deal of expertise

However, the cyber underground market makes this process accessible to the mass given a small amount of money

Consequences

Antivirus “is dead” says Brian Dye, Symantec's senior vice president for information security. **"We don't think of antivirus as a moneymaker in any way."**

Symantec Develops New Attack on Cyberhacking
The Wall Street Journal

Excellent Reference

“Russian Underground 101”

Max Goncharov, Trend Micro Incorporated, 2012

http://www.trendmicro.com/cloud-content/us/pdfs/security-intelligence/white-papers/wp-russian-underground-101.pdf