Information Systems Security

Lecture 3: Threats And Countermeasures

Prof. Dr. Christoph Karg

Aalen University of Applied Sciences

Department of Computer Science

11.10.2015

Learning Objective

The goal of this lecture is to give an overview over common threats that may harm IT systems and their users.

In particular, the following questions will be addressed:

- What is a buffer overflow?
- Who does a computer virus work?
- Which risks are associated with passwords?
- Which threats lurk in the Internet?
- What does phishing mean?
- Which threats do exist for web applications?

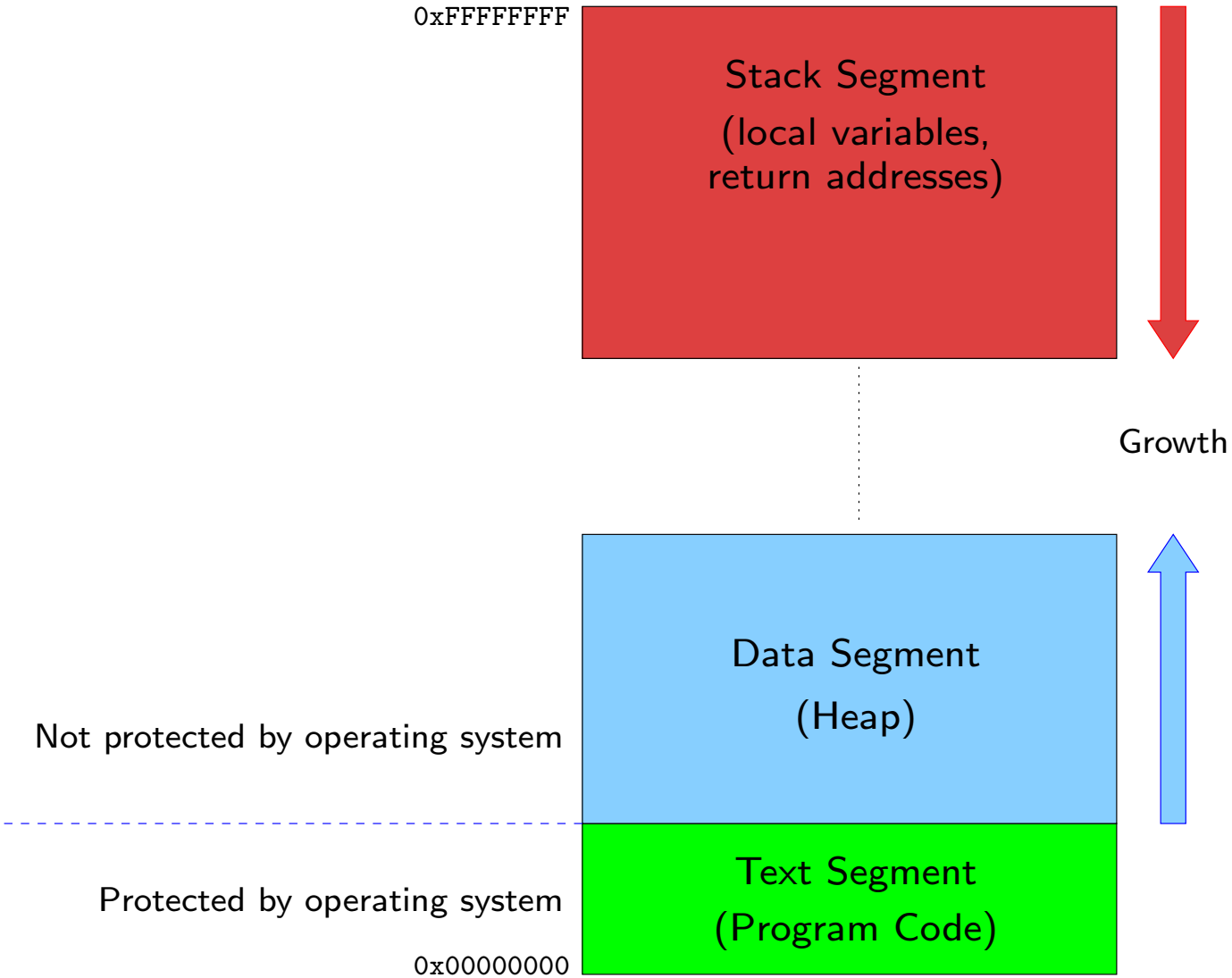
Overview

- Programming Flaws
  - Buffer overflows
  - Heartbleed Bug
- Malware
  - Computer viruses
  - Computer worms
  - Trojan horses
- Weak passwords
- Phishing
- Botnets
- Web Application Issues

Buffer Overflows

- Usage of exploits caused by programming flaws
- Point of interest: software written in C or C++
- Starting point: code fragments which use `strcpy()`
- `strcpy()` uses byte value 0 as the end of string mark
- Bad style: copy a string without length check
- Approach: Overwrite memory with a string which contains malicious code

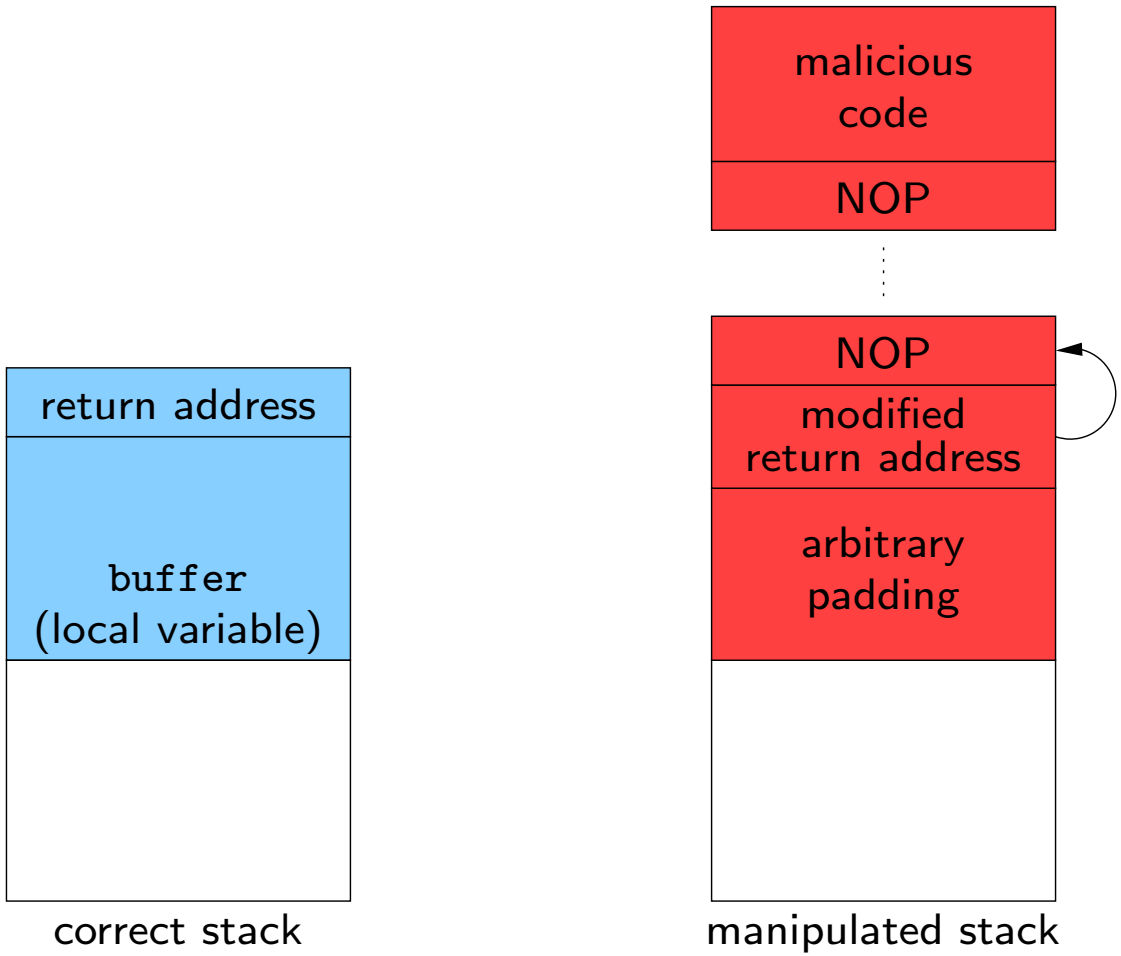
Process Memory Management



Stack Management

- Each process manages its own stack
- Function calls are managed with stack frames
- A stack frame stores:
  - the parameters and local variables of the called function
  - the return address of the calling function
- Important registers (Intel x86 32-Bit):
  - Extended base pointer (ebp) $\rightsquigarrow$ base of the current stack frame (higher address)
  - Extended stack pointer (esp) $\rightsquigarrow$ top of the stack (lower address)

Stack Manipulation



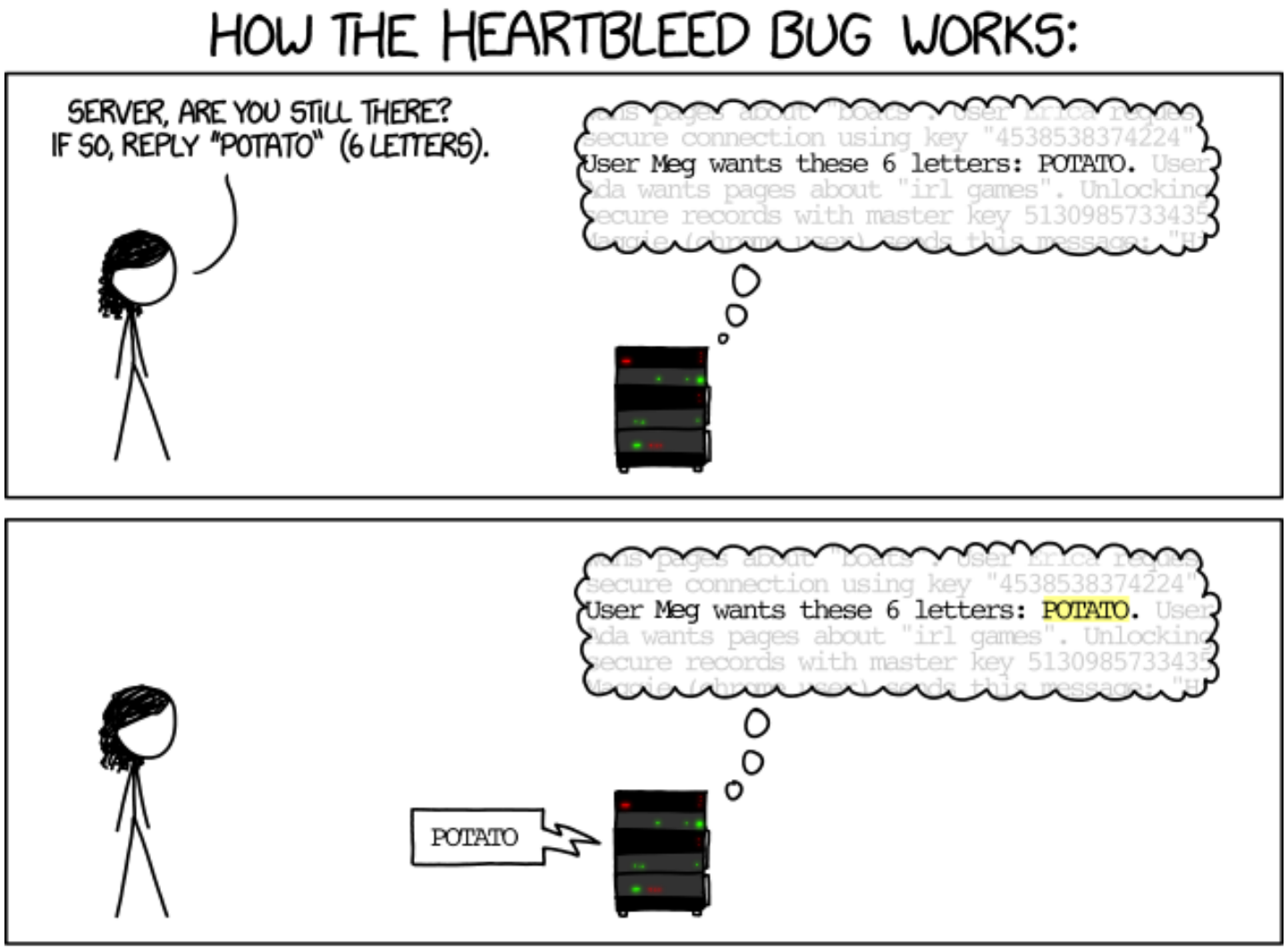
Countermeasures

- Develop your software carefully!
- Use a programming language with advanced security features
- Use safe libraries which offer security enhanced implementations of `strcpy()` and friends
- Do not disable the security mechanisms provided by your compiler and operating system
- Use code analyzers to search for insecure code fragments
- Use trusted versions of your operating system or enable additional security features

Heartbleed Bug

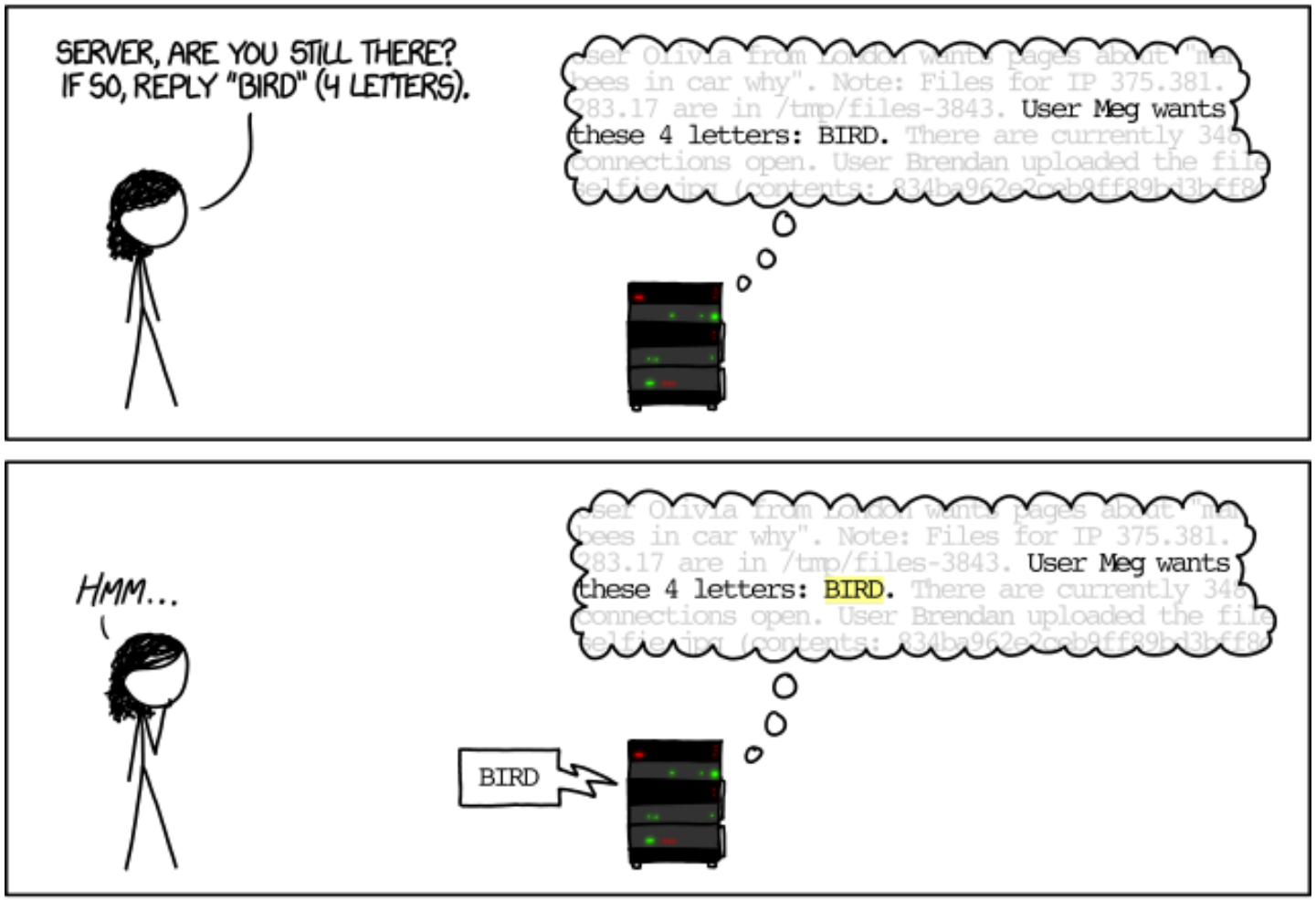
- The heartbeat extension of the TLS protocol can be used to check whether a connection to a server still exists.
- Approach:
  1. The client sends a message together with its length to the server.
  2. The server sends the message back to the client.
- Heartbleed Bug
  - ▷ The OpenSSL implementation of the heartbeat extension does not check whether the length of the message is equal to the given size.
  - ▷ If the size value is greater than the message length, then the OpenSSL implementation returns parts of its internal memory.

Explanation



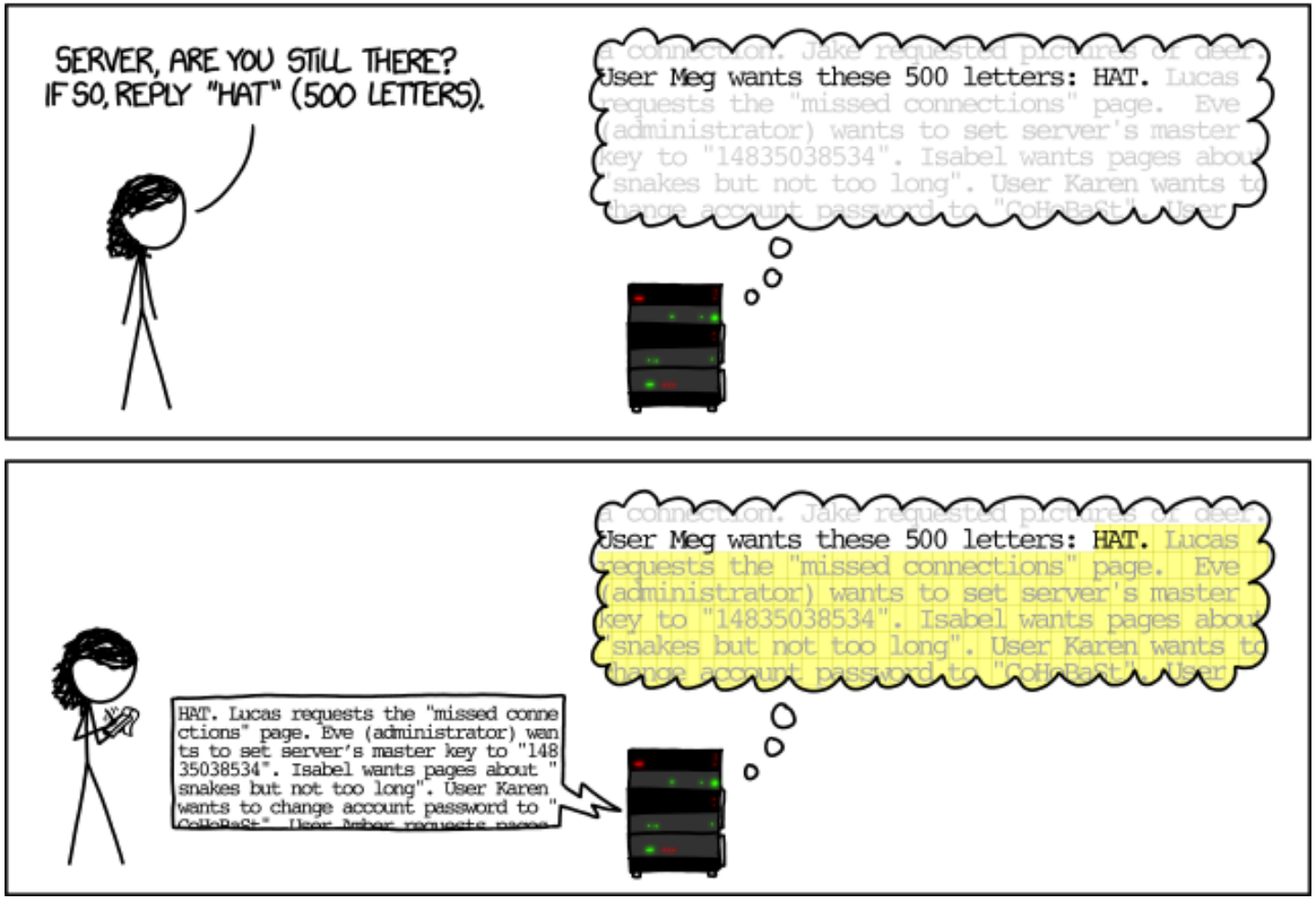
[xkcd]

Explanation (Cont.)



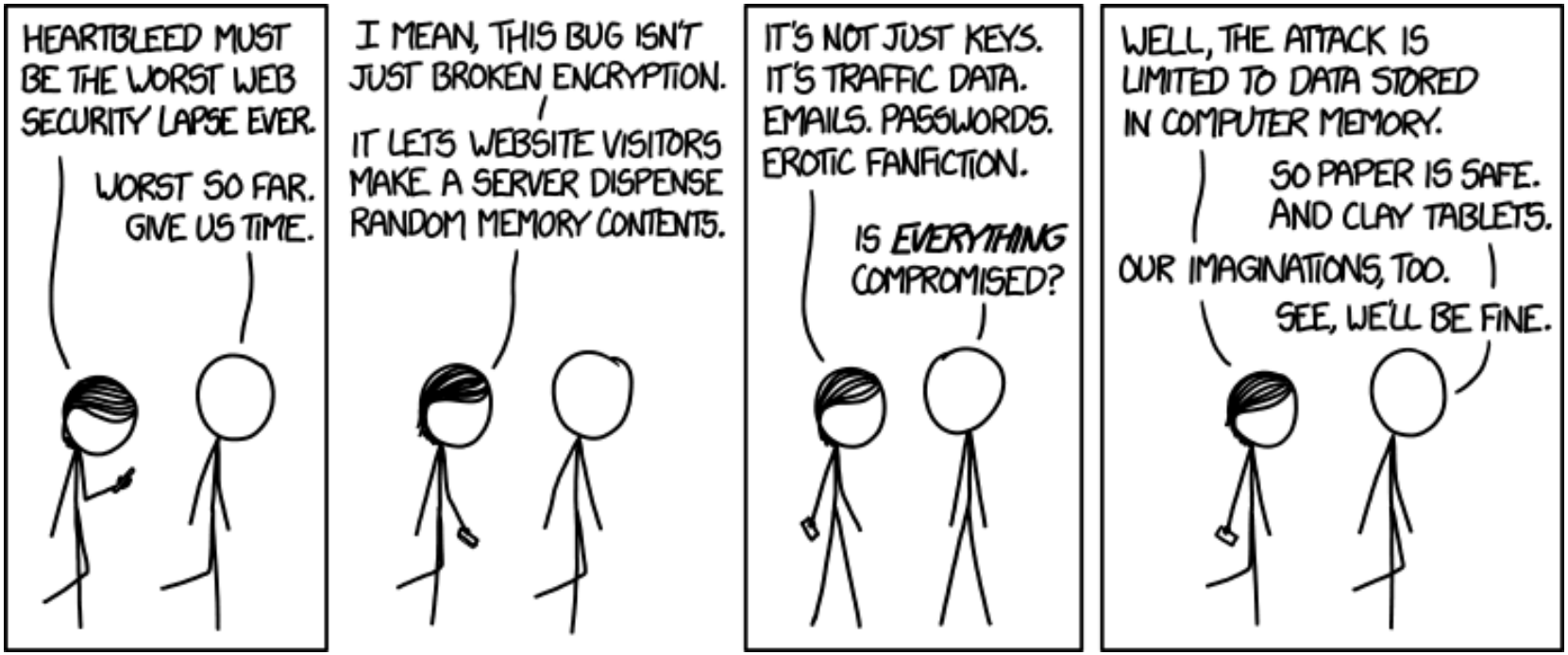
[xkcd]

Explanation (Cont.)



[xkcd]

Consequences



[xkcd]

Consequences (Cont.)

- OpenSSL from version 1.0.1 to 1.0.1f contains the heartbleed bug.
- A large number of online services was affected by the heartbleed bug.
- The bug was patched with version 1.0.1g on 7. April 2014.
- Bruce Schneier's remark:

  "Catastrophic" is the right word.
  On the scale of 1 to 10, this is an 11.

  For more details: [Schneier-Heartbleed]

Computer Viruses

- A computer virus is a code fragment which needs an host program for execution.
- The virus infects other files with his code (reproduction).
- The virus contains a part with malicious code which can cause severe damage to the host system.
- A mutant virus modifies his code during reproduction.
- The term "computer virus" was introduced by Adleman and Cohen in 1984.

Blueprint Of A Virus

A computer virus consists of the following parts:

1. Identification $\rightsquigarrow$ information that marks an object as "infected"
2. Infection $\rightsquigarrow$ code which searches for objects which can be infected and copies the virus code
3. Malicious Code $\rightsquigarrow$ code which performs harmful activities such as password sniffing, deleting the hard disk, ...

   Note: the execution may be triggered by the occurrence of a certain event.

   Example: on Friday, 13th, delete the system's hard disk
4. Host application starter $\rightsquigarrow$ code to execute the host application

Types Of Viruses

First generation

- Program virus
  - Virus is embedded in an executable file
  - The host application must be started to execute the virus
- Boot virus
  - Virus infects the boot loader of the hard disk
  - Virus is started at system start up

Second generation

- Macro virus
  - The virus is part of a document such as an attachment of an email or a office document (PDF, spreadsheet)
  - The virus is executed on processing the document

Example: Sobig Virus

Infos

- Rapid spreading in 2003
- Infection of Windows operating systems
- Usage of social engineering

Functioning

- Virus was transmitted as part of a Program Information File (PIF) via email ("Please see the attached file for details")
- Sobig created a file `winmgm32.exe` in the Windows system folder and modified the registry to execute this file on system start-up
- On execution, Sobig sent an infected picture to each of the user's Outlook contacts
- Sobig installed a key logger to sniff the user's passwords
- The gathered information was sent periodically to geocities.com

Countermeasures

Preventive methods

- Use an anti-virus software (and keep it up-to-date)
- Be careful when downloading software from the Internet
- Restrict the user's access control to prevent modifications of system files
- Use monitoring tools to observe the system critical files
- Perform backups of your data or virtual machines

Recovery methods

- Use a cleaning software
- Restore a backup which is not infected
- If nothing other helps: re-install the operating system

Computer Worms

- A computer worm is a program which is executable and able to reproduce itself.
- A worm consists of several code parts, so called worm segments.
- The reproduction is done automatically, usually by communication with other worm segments.
- Worms are spread via the Internet, for instance by access of a malicious web page.
- Common entry points are buffer overflows.

Example 1: ILOVEYOU

- Spread by a Visual Basic script as an email attachment
- If received with MS Outlook, the script was automatically executed and sent an email to all contacts of the user
- ILOVEYOU destroyed image, audio and video files and searched for passwords on the local hard disk

Example 2: Lovesan/Blaster Worm

- Goal: Distributed Denial of Service (DDoS) attack on the server *windowsupdate.com*
- Infection by usage of a buffer overflow in the Windows DCOM RPC service
- On 16.8.2003, a SYN-flood attack should be performed. Luckily, Microsoft removed the DNS entry of the above server before this date

Countermeasures

- Use anti-virus and anti-spyware software
- Keep your system up-to-date
- Use a firewall and prevent unrestricted access from outside into your network
- Use access control mechanisms

Trojan Horses

- A Trojan horse is a piece of software where the real functionality deviates from the described functionality
- The name of this kind of malware is derived from the war of Trojan saga of the Greek mythology
- Trojan horses are threats for any part of a computer system
- Common techniques
  - Text processors which copy documents without authorization
  - Databases which provide sensible informations to unauthorized persons
  - System applications which grant unrestricted access to the attacker
  - Software which scans for password data

Example: Zeus Trojan horse

- The Zeus Trojan horse is a malware to manipulate bank transfers
- The code exists in several variations
- The code contains stealth mechanisms to prevent detection by virus scanners
- Windows is the main target
- Mobile variants do exist for Android, Blackberry, Symbian, and Windows Mobile

Countermeasures

- Be careful while downloading software from the Internet
- Restrict the user authorization as much as possible
- Do not trust informations of untrustworthy origin
- Do not store passwords and other confidential information in plain text
- Use an external device to store confidential information

Spam E-Mails

- Spam are e-mails which contain junk or unsolicited commercial advertisements
- Spam mails are also used for password phishing
- Spamhaus project: estimates that 95% of the e-mail traffic in America and Europe is spam
- Nucleus Research: Each employee of a company in the U.S. receives 13.3 spam mails per day and spends 6.5 minutes per day on processing them $\rightsquigarrow$ financial damage of 120 millions USD in 2004
- Countermeasure: Use a spam filter
- But: In an industrial environment, automated processing of spam e-mails is subject to legal restrictions

Phishing

Goal: Get confidential access data directly from the user

Approach:

- Setup a bogus web page for "account recovery"
- Send an email with an message concerning the "loss" of the user's account data
- Phish the account data through the web page

Targets:

- Bank accounts
- Credit card data
- Web shop accounts
- Accounts of social media platforms

Example 1: EBay Phishing

Dear eBay Member,

We regret to inform you that your eBay account could be suspended if you don't re-update your account information.
To resolve this problem please visit link below and re-enter your account information:

https://signin.ebay.com/ws/eBayISAPI.dll?SignIn&sid=verify&co_partnerId=2&siteid=0

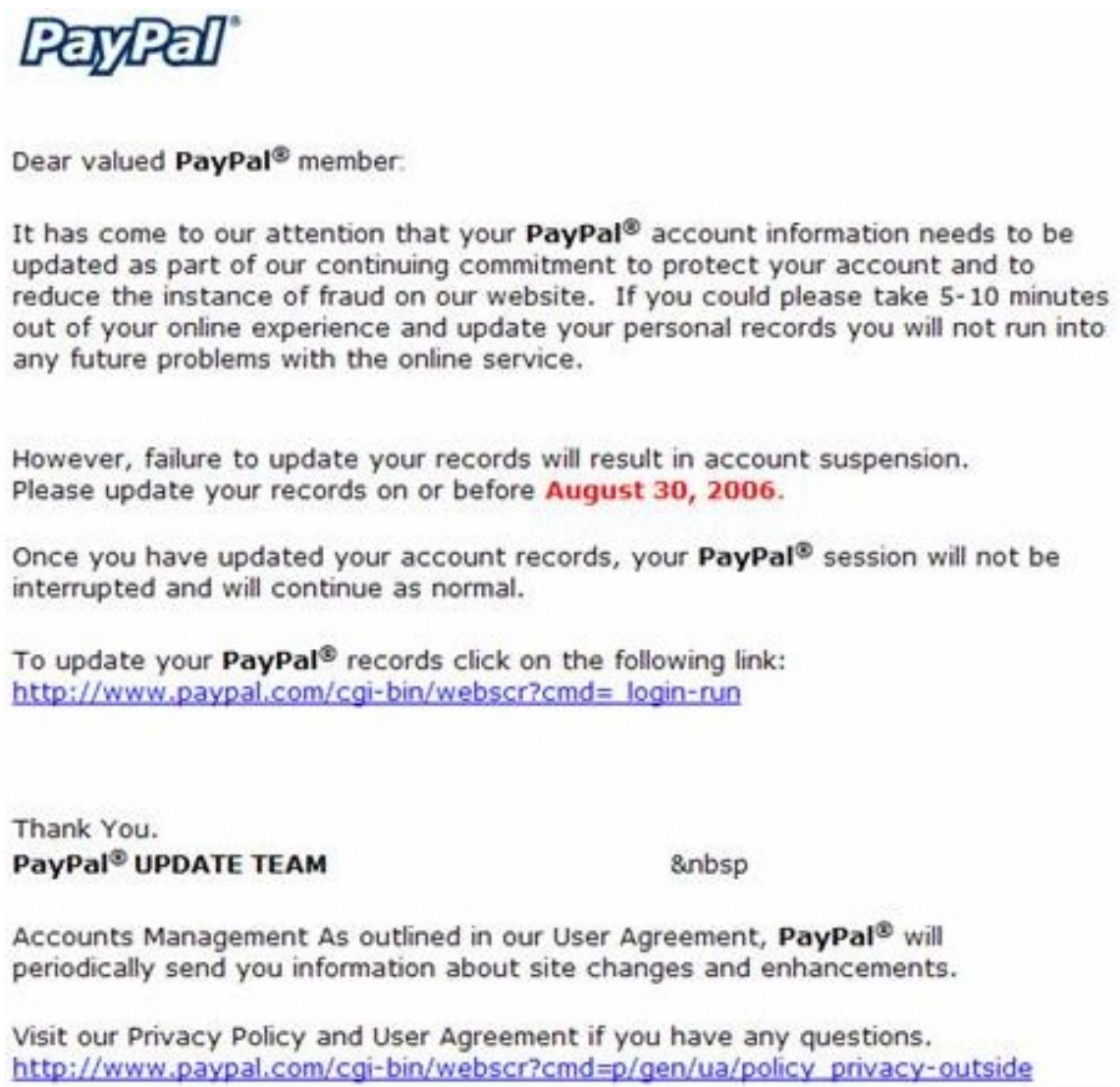
If your problems could not be resolved your account will be suspended for a period of 24 hours, after this period your account will be terminated.

For the User Agreement, Section 9, we may immediately issue a warning, temporarily suspend, indefinitely suspend or terminate your membership and refuse to provide our services to you if we believe that your actions may cause financial loss or legal liability for you, our users or us. We may also take these actions if we are unable to verify or authenticate any information you provide to us.

Due to the suspension of this account, please be advised you are prohibited from using eBay in any way. This includes the registering of a new account. Please note that this suspension does not relieve you of your agreed-upon obligation to pay any fees you may owe to eBay.

Regards,
Safeharbor Department eBay, Inc
The eBay team
This is an automatic message, please do not reply

Example 2: Paypal Phishing



PayPal®

Dear valued **PayPal®** member.

It has come to our attention that your **PayPal®** account information needs to be updated as part of our continuing commitment to protect your account and to reduce the instance of fraud on our website. If you could please take 5-10 minutes out of your online experience and update your personal records you will not run into any future problems with the online service.

However, failure to update your records will result in account suspension.
Please update your records on or before **August 30, 2006.**

Once you have updated your account records, your **PayPal®** session will not be interrupted and will continue as normal.

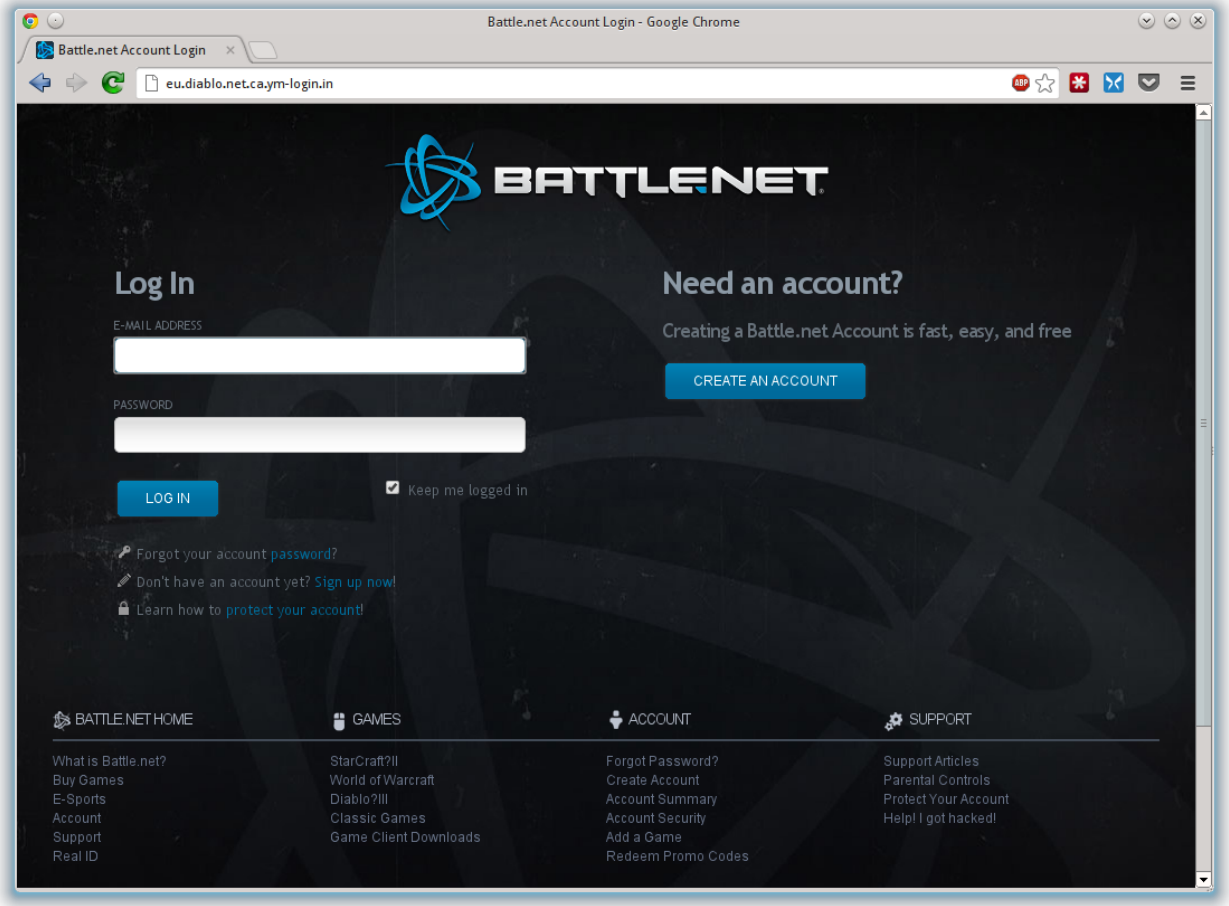
To update your **PayPal®** records click on the following link:
http://www.paypal.com/cgi-bin/webscr?cmd=_login-run

Thank You.
PayPal® UPDATE TEAM

Accounts Management As outlined in our User Agreement, **PayPal®** will periodically send you information about site changes and enhancements.

Visit our Privacy Policy and User Agreement if you have any questions.
http://www.paypal.com/cgi-bin/webscr?cmd=p/gen/ua/policy_privacy-outside

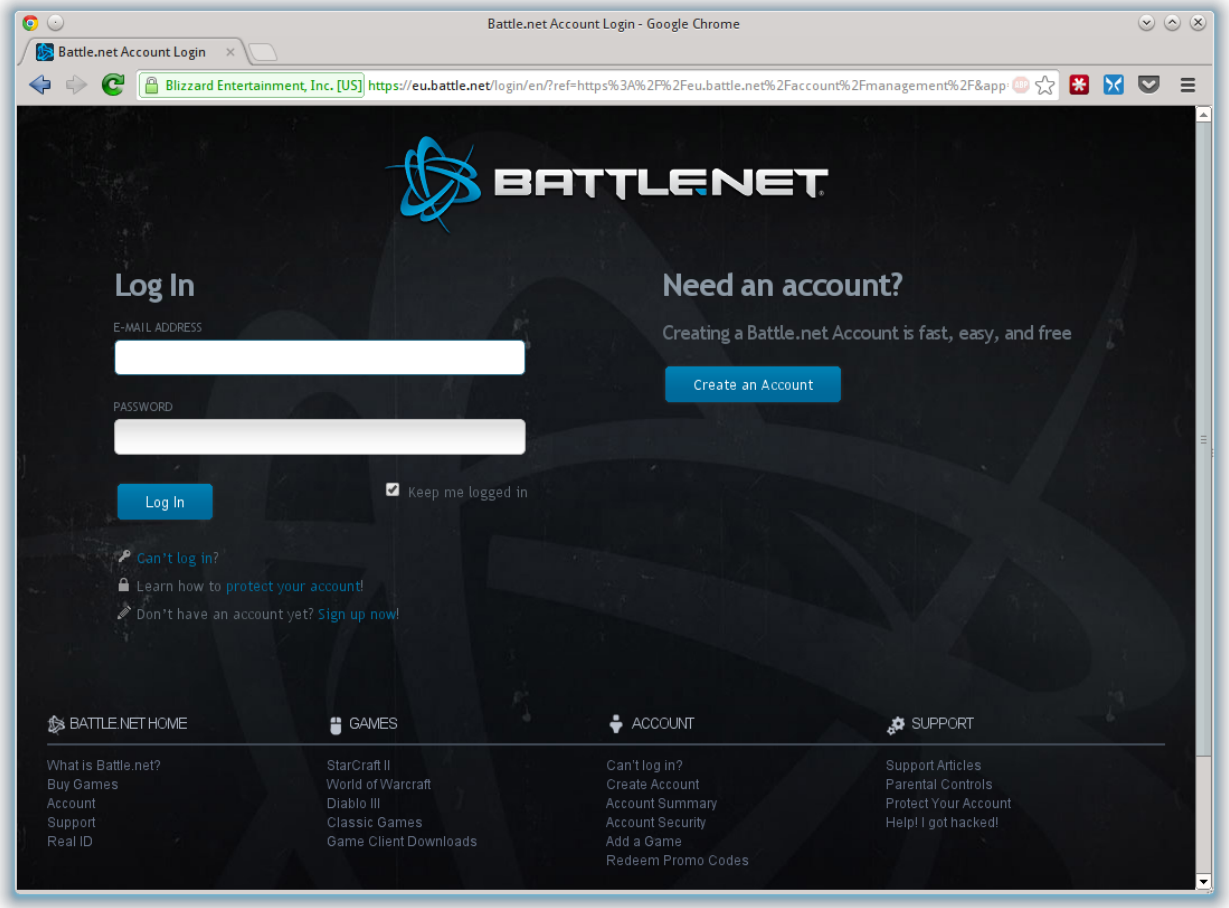
Example 3: Battlenet Phishing



URL of this page: `http://eu.diablo.net.ca.ym-login.in`

Example 3: Battlenet Phishing (Cont.)

The original page looks like this:



Countermeasures

- Use a virus scanner
- Use a web browser which is up-to-date
- Use your brain
  - Links in HTML mails and PDF documents might be bogus
    $\rightsquigarrow$ `http://www.google.com`
  - Serious companies do not request passwords or other confidential information via email
- For a list of current phishing attacks, take a look at `http://www.phishtank.com`

Usage Of Passwords

- Passwords are the most common method of authentification
- Advantages
  - Simple to generate
  - Simple to use
  - Cheap
- Disadvantages
  - Good passwords are hard to keep in mind
  - Security does not depend solely on the user
  - Password must be entered into the system

The RockYou Incident 2009

- RockYou was a webpage for social networking.
- Users had the possibility to save data from other accounts sch as MySpace or Facebook.
- RockYou's user database saved the passwords in plaintext.
- December 2009: SQL injection attack against RockYou.
- Consequence: 32 millions of stolen passwords.
- An anonymized list of passwords was published.
- For more details see: [Techcrunch Link]

Analysis Of The RockYou Passwords

- 30% of the passwords had a length of at most 6 characters
- 20% of the passwords were included in a dictionary with 5000 easy to be guessed passwords
- 40% of the users chose passwords consisting only of lower case letters
- Only 0.2% of the passwords fulfilled the following (typical) requirements:
  - Length of at least 8 characters
  - Mixture of upper and lower case letters, digits and special symbols
- For details see: [Imperva document]

Passwords Frequently Used

Rank	*Password*	*Frequency*
1	123456	290731
2	12345	79078
3	123456789	76790
4	Password	61958
5	iloveyou	51622
6	princess	35231
7	rockyou	22588
8	1234567	21726
9	12345678	20553
10	abc123	17542

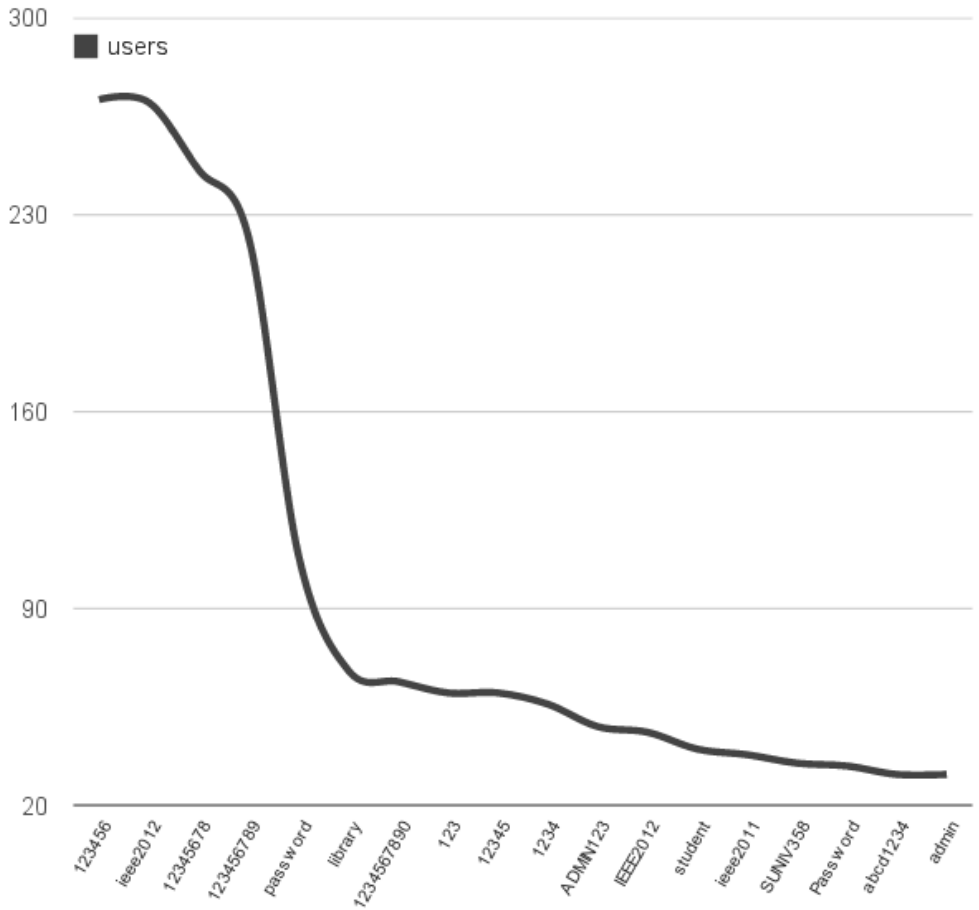
Rank	*Password*	*Frequency*
11	Nicole	17168
12	Daniel	16409
13	babygirl	16094
14	monkey	15294
15	Jessica	15162
16	Lovely	14950
17	michael	14898
18	Ashley	14329
19	654321	13984
20	Qwerty	13856

Top 20 of the passwords of RockYou users

Example 2: Stolen IEEE Log Data 2012

- Data breach at IEEE.org in September 2012
- Log data of IEEE.org was available publicly on an FTP server
- Log data time span: 1. August 2012 to 18. September 2012
- Gathered information:
  - Total number of log entries: 376.021.496
  - Log entries with password details: 411.308
  - 99.979 distinct username values
- Consequence: almost 100000 compromised users
- Users from Apple, Google, IBM, Oracle, Samsung, NASA, Stanford university, ...
- For details see: `http://ieeelog.com`

Most Used Passwords



Most Used Passwords (Cont.)

Rank	*Password*
1	123456
2	ieee2012
3	12345678
4	123456789
5	password
6	library
7	1234567890
8	123
9	12345
10	1234

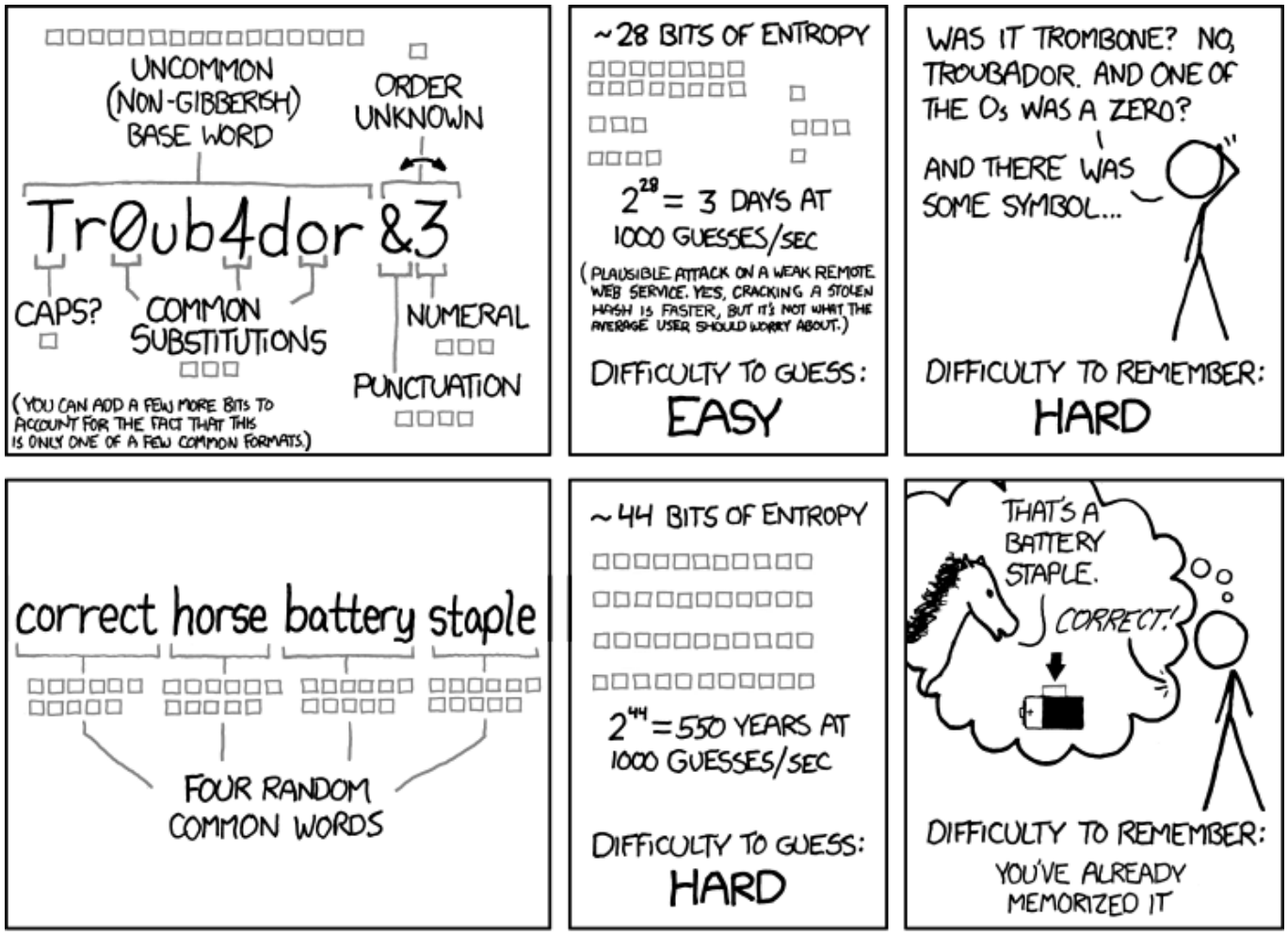
Attacks Against Passwords

- Password hashes can be used for a brute force attack
- A modern PC can compute millions of hashes per second
- The usage of rainbow tables speeds up things
- Cloud services such as Amazon Elastic Cloud provide huge computation power for little money

Recommendations

- Properties of a good password:
  - Length of at least 8 characters
  - Combination of upper and lower letters, digits and special characters
  - Does not include user name, name of a dictionary or an email address
- The quality of a password should depend of the importance of the information to be protected
- Never use a password twice
- Never give an important password to another user
- Change your password frequently
- Use a password safe to store your passwords

Recommendations (Cont.)

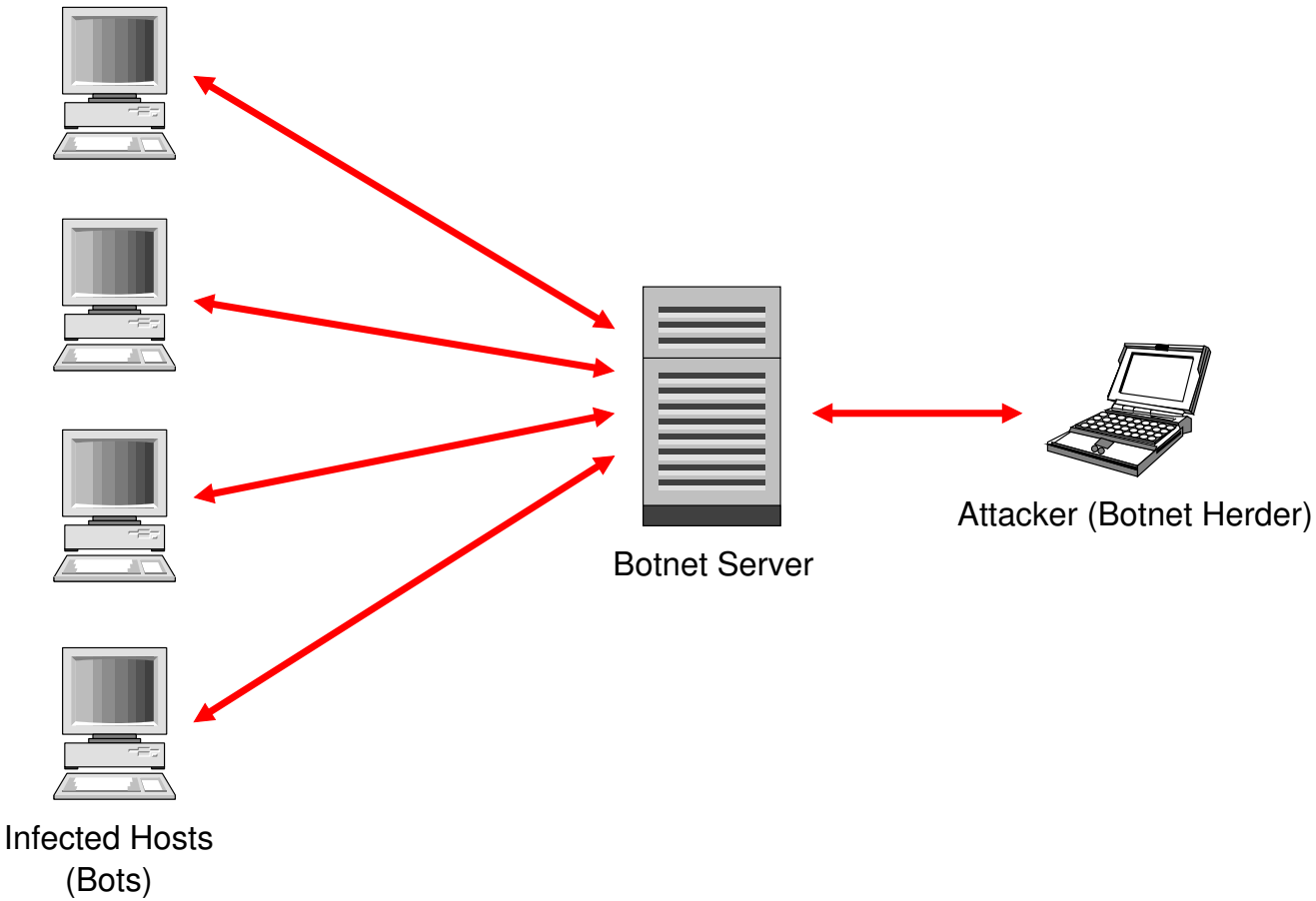


[xkcd]

Botnets

- A botnet is a collection of infected hosts which is controlled by a botnet server.
- The word "bot" is derived from *robot*.
- Botnets may consist of thousands of bots which are distributed over the whole world.
- The malicious code of a botnet is spread via viruses, worms, etc.
- A common way to control the bots is a communication over the Internet Relay Chat (IRC).
- Types of attacks: Distributed Denial of Service, bulk transfer of spam e-mail, password sniffing, etc.

Botnets (Cont.)



Example: Eurograbber

- The Eurograbber attack was a sophisticated attack.
- The attack was multi-dimensional and combined several types attacks such as social engineering and trojan horses.
- In order to perform a successful attack, both the victim's PC and his mobile phone must be manipulated.
- Windows PCs, Android and Blackberry smartphones were the primary targets.

Example: Eurograbber (Cont.)

Step 1: Infecting the victim's systems

[Eurograbber Whitepaper]

Example: Eurograbber (Cont.)

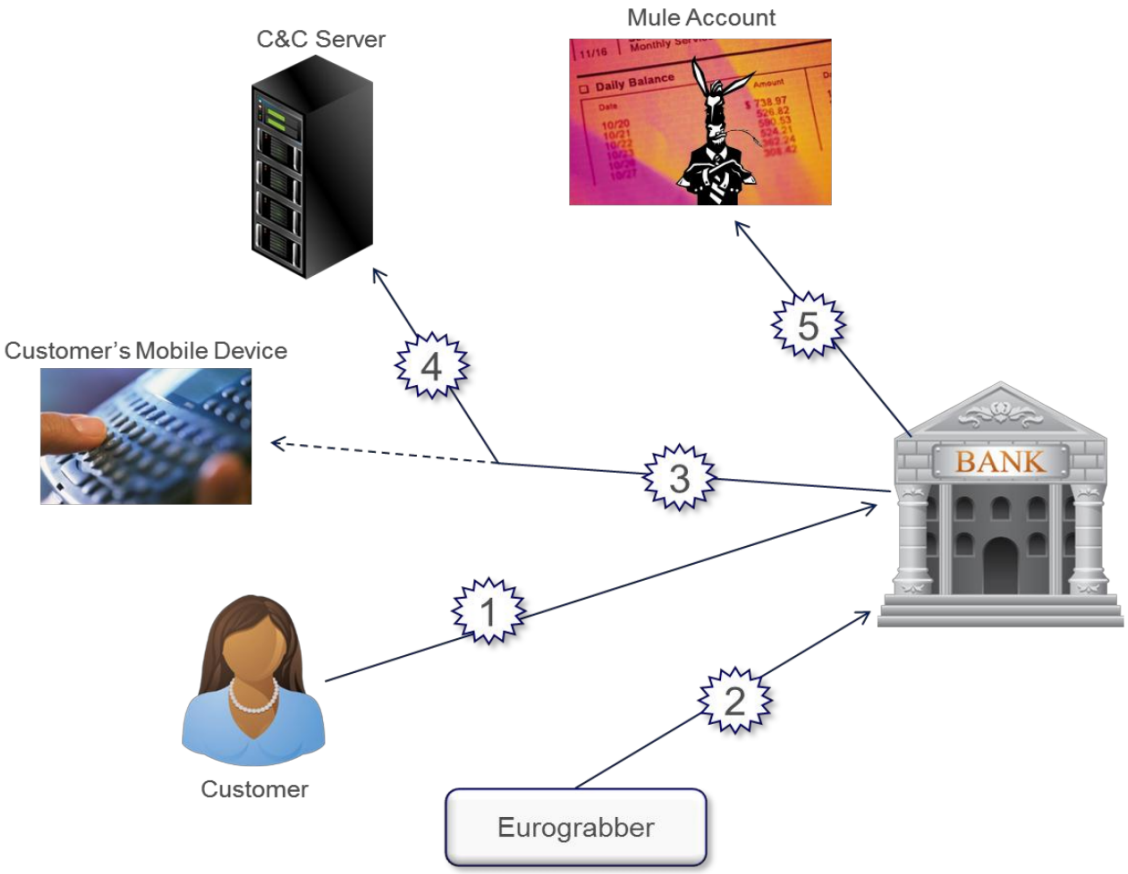
Step 1: Infecting the victim's PC and smartphone

1. By usage of a SPAM Email or a phishing webpage a special version of the Zeus trojan horse is installed on the victim's PC
2. After the victim logs into his online banking account, the trojan horse injects a Javascript code. The script pretends to perform a "security update" and asks for the victim's smartphone type and his phone number
3. The gathered information is stored in a drop zone and is used in the sequel

Example: Eurograbber (Cont.)

4. After receiving the victim's phone number, an SMS is sent to to the victim. The SMS contains a link and instructs the victim to follow the link in order to install an app on his smartphone
5. At the same time, the Trojan horse the victim's PC opens a manual providing a guide for the installation of the app (just in case that the SMS was not received by the victim)
6. After the installation, the app generates a check number and instructs the victim to enter this number into the PC
7. At the end of the installation process the PC displays a success message

Example: Eurograbber (Cont.)



Step 2: Money withdrawal

[Eurograbber Whitepaper]

Example: Eurograbber (Cont.)

Step 2: Money withdrawal

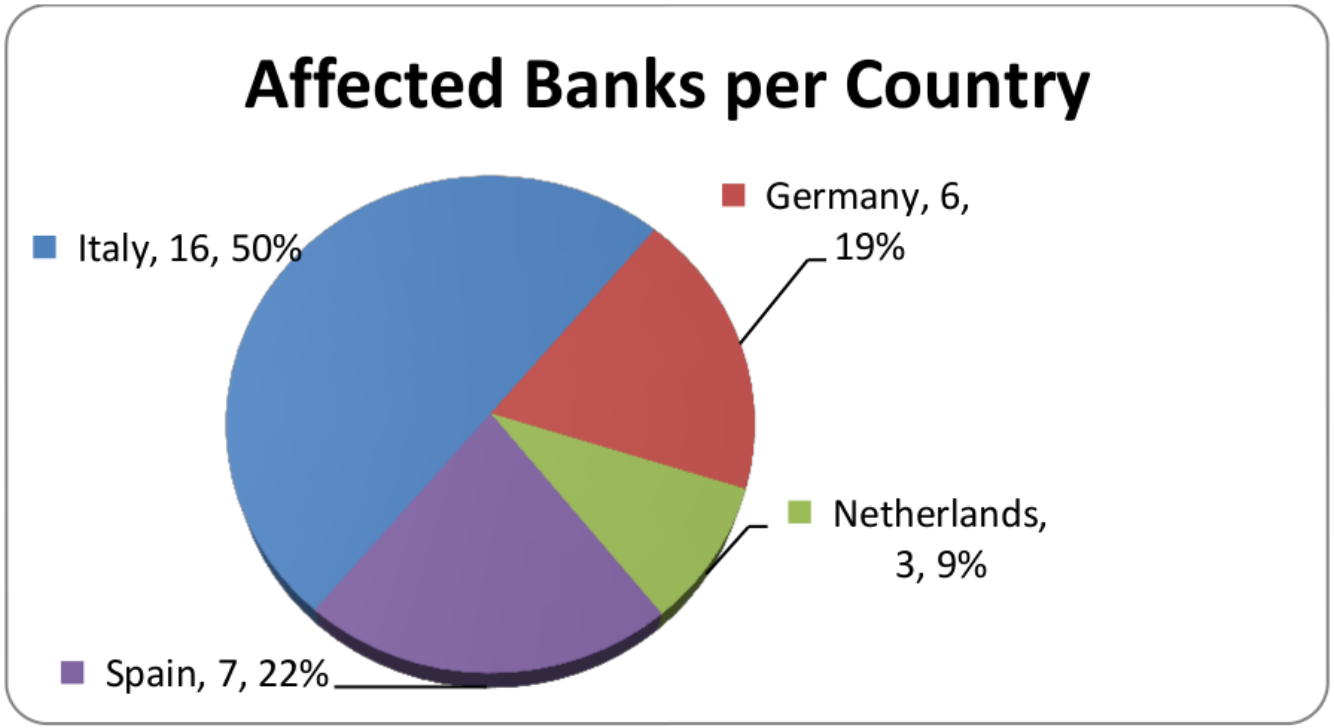
1. The victim logs into his online banking account
2. Immediately after the login, the PC Trojan horse withdraws a percentage of the victim's deposit and transfers the money to the attacker's account
3. The bank sends a Transaction Authorization Number (TAN) to the victim's smartphone
4. The smartphone Trojan intercepts the TAN, hides it from the victim and sends it to the drop zone
5. Der PC Trojan horse pulls the TAN from the drop zone and completes the transaction

Example: Eurograbber (Cont.)

Remarks:

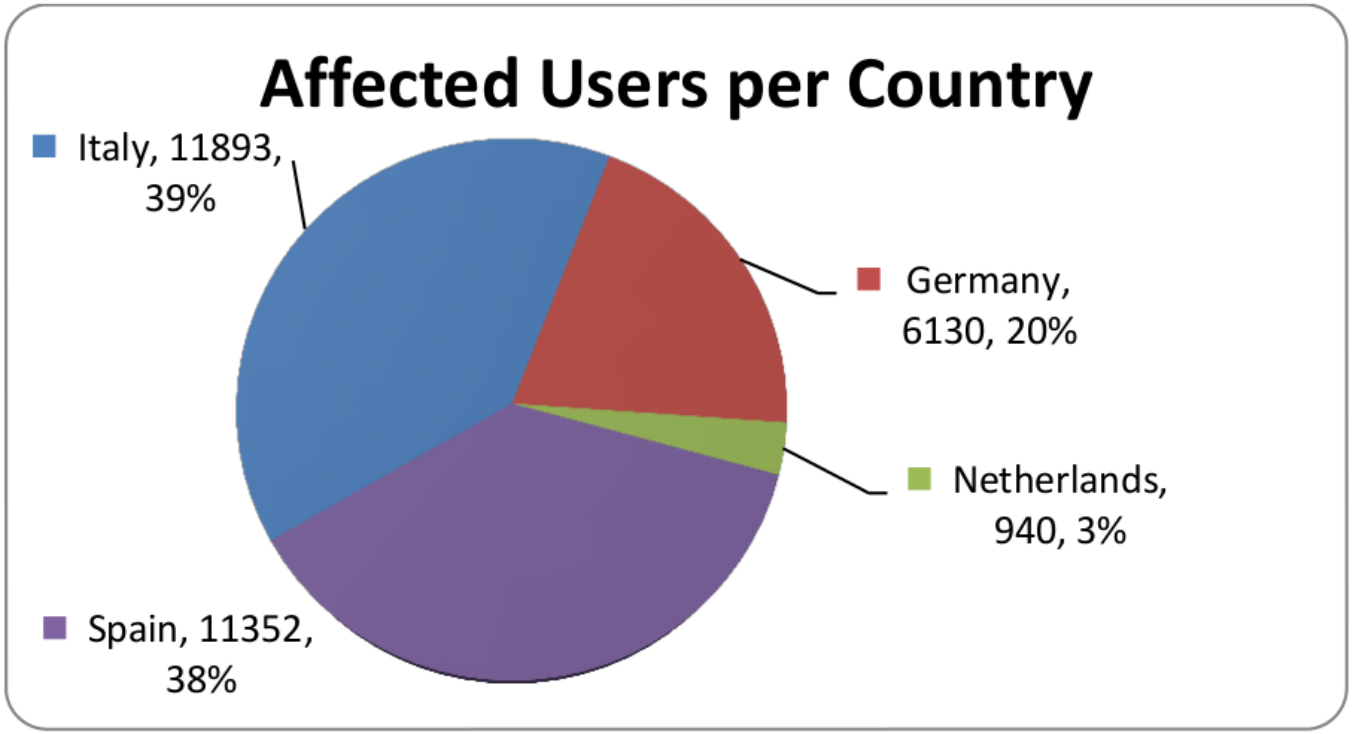
- The complete transaction runs in the background, this is, the customer does not detect it
- On each login a withdrawal is performed
- Eurograbber "earned" at least 36 Million Euros
- More than 30.000 bank customers from multiple banks across Europe were affected by the attack

Example: Eurograbber (Cont.)



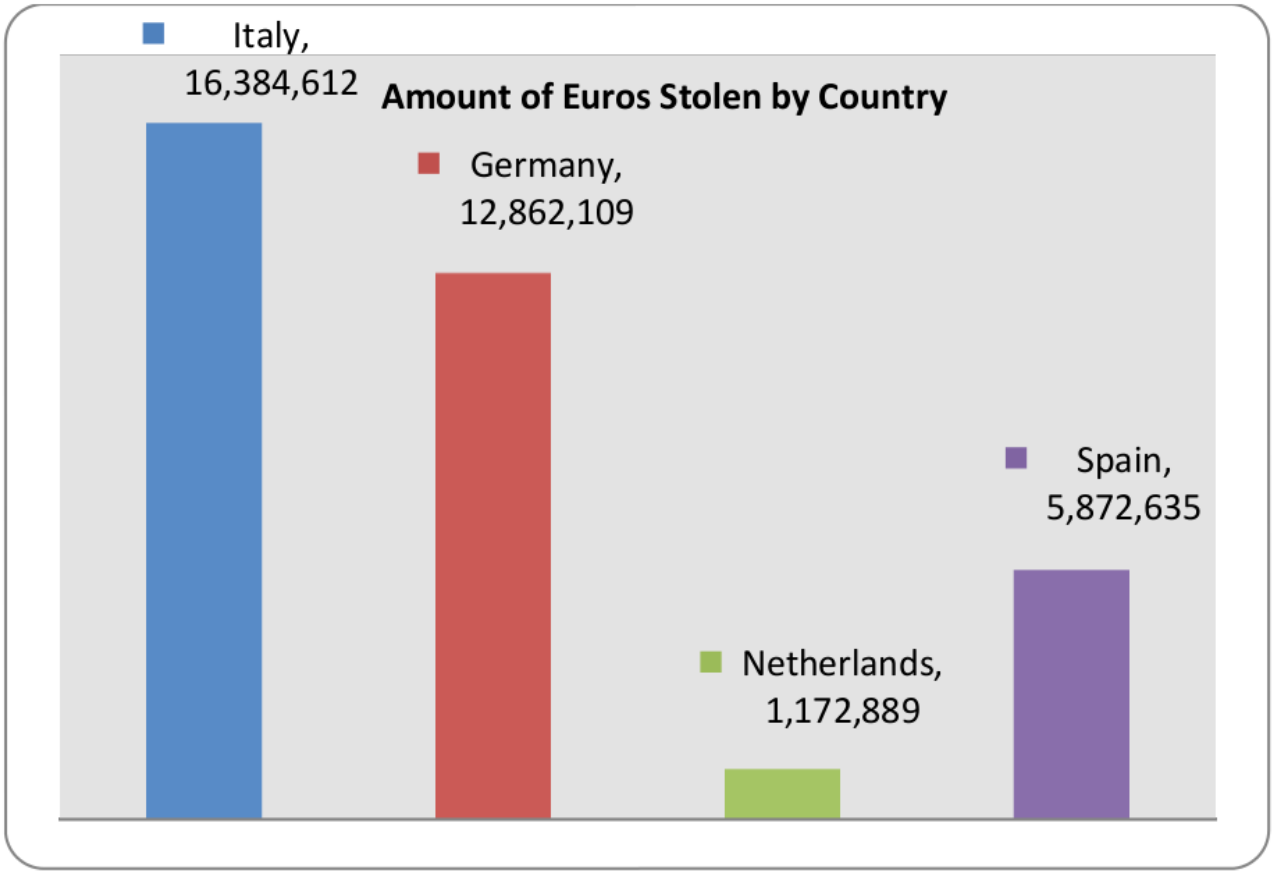
[Eurograbber Whitepaper]

Example: Eurograbber (Cont.)



[Eurograbber Whitepaper]

Example: Eurograbber (Cont.)



[Eurograbber Whitepaper]

Countermeasures

- Perform updates of your operating system and critical software such as web browser, Adobe Flash, Adobe Reader and Java
- Use a virus scanner
- Ignore all emails which ask for bank account information
- Use Linux for online banking
- Install only apps on your smartphone which are provided by the official app store

Web Application Issues

- A web application is a combination of several applications such as a database and web browser.
- The parts of a web application are located on different hosts.
- The analysis of security issues of a web application may be a difficult task.
- The Open Web Application Security Project (OWASP) provides a Top Ten of security risks.

  $\rightsquigarrow$ `https://www.owasp.org/index.php/Top_10_2013-Top_10`

OWASP Top 10 Risks 2013 I

1. Injection: Injection flaws, such as SQL, OS, and LDAP injection occur when untrusted data is sent to an interpreter as part of a command or query. The attacker's hostile data can trick the interpreter into executing unintended commands or accessing unauthorized data.
2. Broken Authentication and Session Management: Application functions related to authentication and session management are often not implemented correctly, allowing attackers to compromise passwords, keys, session tokens, or exploit other implementation flaws to assume other users' identities.

OWASP Top 10 Risks 2013 II

3. Cross-Site Scripting (XSS): XSS flaws occur whenever an application takes untrusted data and sends it to a web browser without proper validation or escaping. XSS allows attackers to execute scripts in the victim's browser which can hijack user sessions, deface web sites, or redirect the user to malicious sites.
4. Insecure Direct Object References: A direct object reference occurs when a developer exposes a reference to an internal implementation object, such as a file, directory, or database key. Without an access control check or other protection, attackers can manipulate these references to access unauthorized data.

OWASP Top 10 Risks 2013 III

5. Security Misconfiguration: Good security requires having a secure configuration defined and deployed for the application, frameworks, application server, web server, database server, and platform. All these settings should be defined, implemented, and maintained as many are not shipped with secure defaults. This includes keeping all software up to date.
6. Sensitive Data Exposure: Many web applications do not properly protect sensitive data, such as credit cards, tax ids, and authentication credentials. Attackers may steal or modify such weakly protected data to conduct identity theft, credit card fraud, or other crimes. Sensitive data deserves extra protection such as encryption at rest or in transit, as well as special precautions when exchanged with the browser.

OWASP Top 10 Risks 2013 IV

7. Missing Function Level Access Control: Virtually all web applications verify function level access rights before making that functionality visible in the UI. However, applications need to perform the same access control checks on the server when each function is accessed. If requests are not verified, attackers will be able to forge requests in order to access unauthorized functionality.
8. Cross-Site Request Forgery (CSRF): A CSRF attack forces a logged-on victim's browser to send a forged HTTP request, including the victim's session cookie and any other automatically included authentication information, to a vulnerable web application. This allows the attacker to force the victim's browser to generate requests the vulnerable application thinks are legitimate requests from the victim.

OWASP Top 10 Risks 2013 V

9. Using Components with Known Vulnerabilities: Vulnerable components, such as libraries, frameworks, and other software modules almost always run with full privilege. So, if exploited, they can cause serious data loss or server takeover. Applications using these vulnerable components may undermine their defenses and enable a range of possible attacks and impacts.
10. Unvalidated Redirects and Forwards: Web applications frequently redirect and forward users to other pages and websites, and use untrusted data to determine the destination pages. Without proper validation, attackers can redirect victims to phishing or malware sites, or use forwards to access unauthorized pages.

Summary

- Threats to IT systems exist since computers entered the mass market.
- The growing complexity of IT systems makes them harder to protect.
- Vulnerabilities of internet applications cause a threat on a large number of IT systems.
- Generally, vulnerabilites cannot be prevented. You have to react on upcoming exploits.

References

- J. R. Vacca: Computer And Information Security Handbook, Morgan-Kaufman, 2010.
- B. Schneier: Secrets And Lies - Digital Security in a Networked World, Wiley, 2004. J. Erickson: Hacking: The Art of Exploitation, No Starch Press, 2007.
- S. Harris: Gray Hat Hacking: The Ethical Hacker's Handbook, McGraw-Hill, 2008.
- E. Kalige, D. Burkey: A Case Study of Eurograbber: How 36 Million Euros was Stolen via Malware, Versafe/Check Point, 2012.