COSC 458-647
Application Software Security

Today

- What is "software security"?
- The problem of software insecurity
- The causes of the problem
- Security concepts
- The solution to the problem?

Quiz

- What do *web-sites, web-browsers, operating systems, wifi access points, network routers, mobile phones, PDAs, smartcards, firewalls, intrusion detection systems, and videoconferencing equipment* have in common?

- Why can all these things be hacked, if we are not very careful?

Why software security?

- Software plays a major role in providing security, and is a major source of security problems.
  - Software is possibly the weakest link in the security chain (human-software-hardware), with the possible exception of “the human factor”

- Software security does not get much attention
  - in other security courses, or
  - in programming courses,

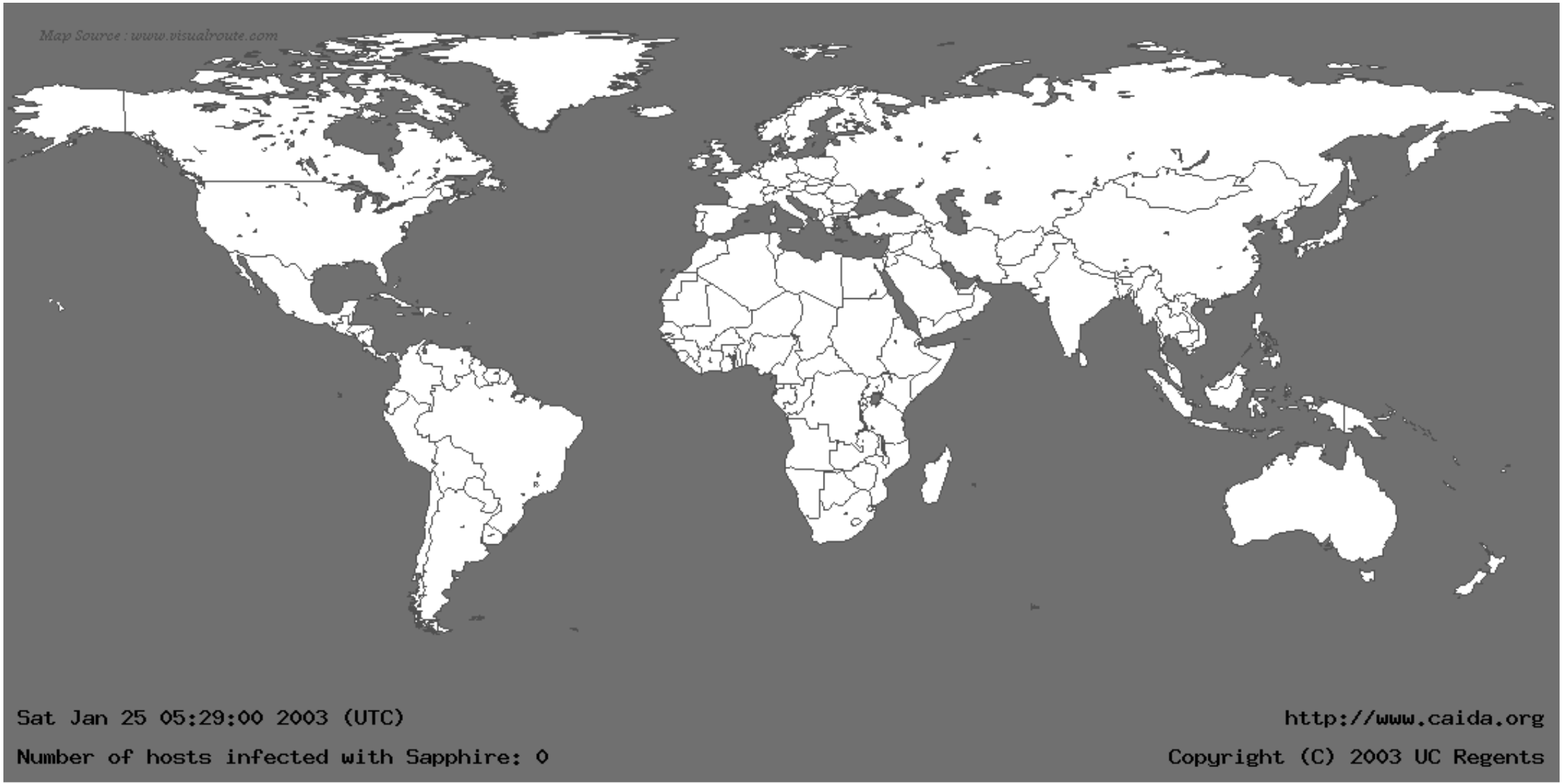
or indeed, in much of the security literature!

- We focus on software security, but don't forget that security is about, in no particular order,

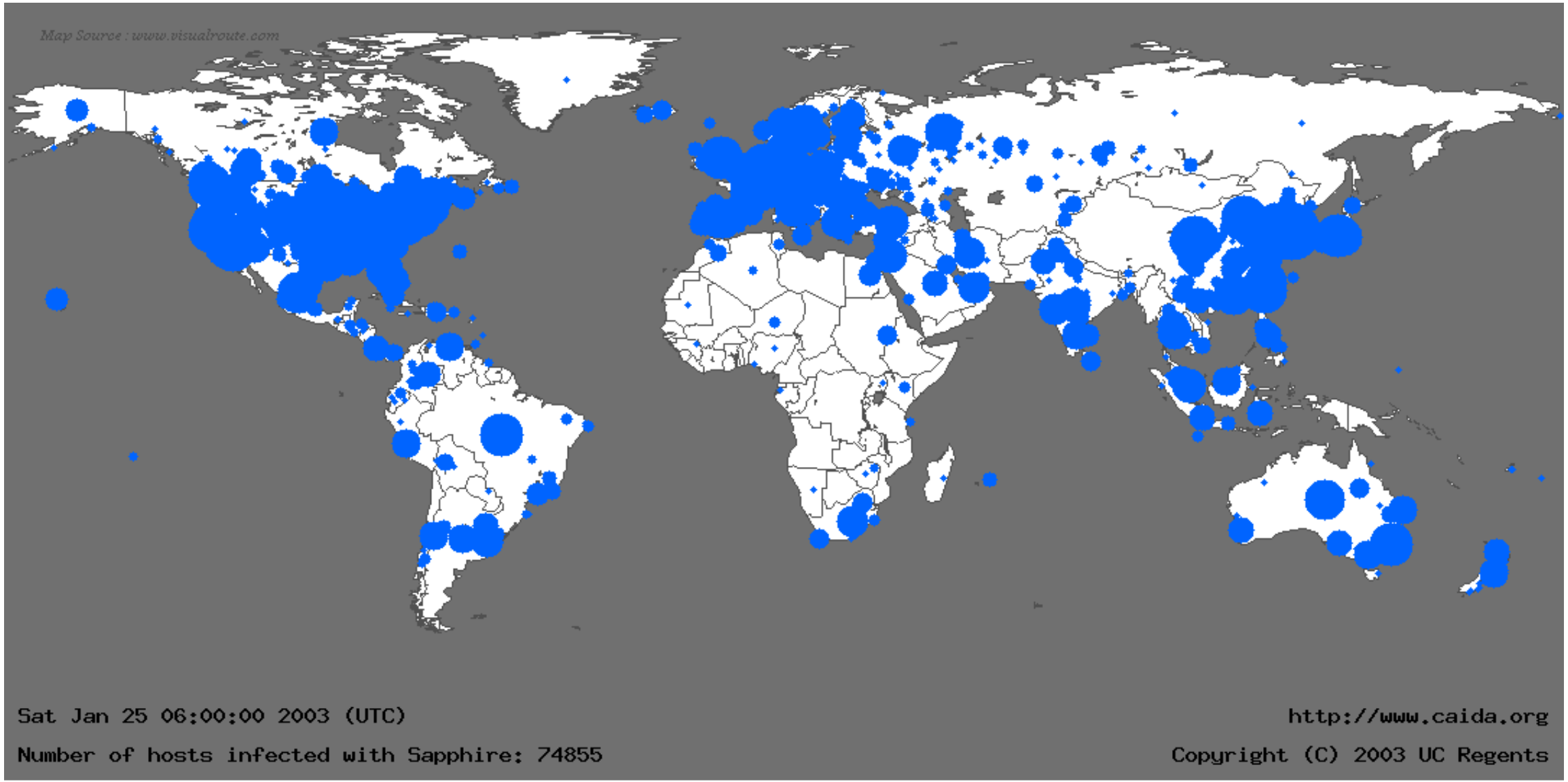
  people (users, employees, sys-admins, programmers,...), access control, passwords, biometrics, cryptology, protocols, policies & their enforcement, monitoring, auditing, legislation, persecution, liability, risk management, incompetence, confusion, lethargy, stupidity, mistakes, complexity, ***software***, bugs, verification, hackers, viruses, hardware, operating systems, networks, databases, public relations, public perception, conventions, standards, physical protection, data protection, ...

The problem

Slammer Worm (Jan 2003)



Slammer Worm (cont'd)



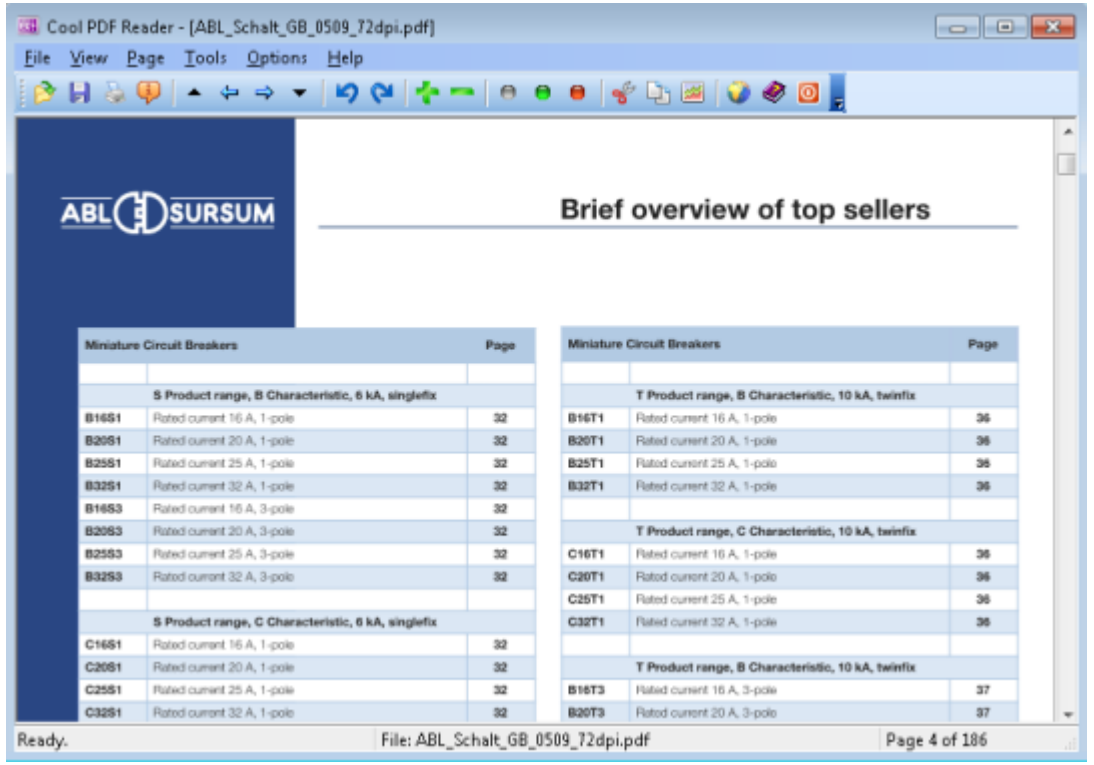
Slammer Worm

- Was the fastest computer worm in the history
- Doubled size in every 8.5s, infected 90% of vulnerable hosts.
- Exploited a *buffer overflow* vulnerability in computers running Microsoft's SQL Server or Microsoft SQL Server Desktop Engine (MSDE) 2000
  - An underlying indexing service; discovered in July 2002
- Spread out based on random scanning
  - Selected IP addresses at random to infect
- Infected > 75K hosts & caused network outages → canceled flights, inference with election, ATM failures, etc.

- Ref: The Spread of the Sapphire/Slammer Worm(http://www.caida.org/publications/papers/2003/sapphire/sapphire.html)

CoolPDF

- Published: 2013-01-26
- Vulnerability No: CVE-2012-4914
- CVSS Severity Score: 9.3
- Vendor/Product: coolpdf



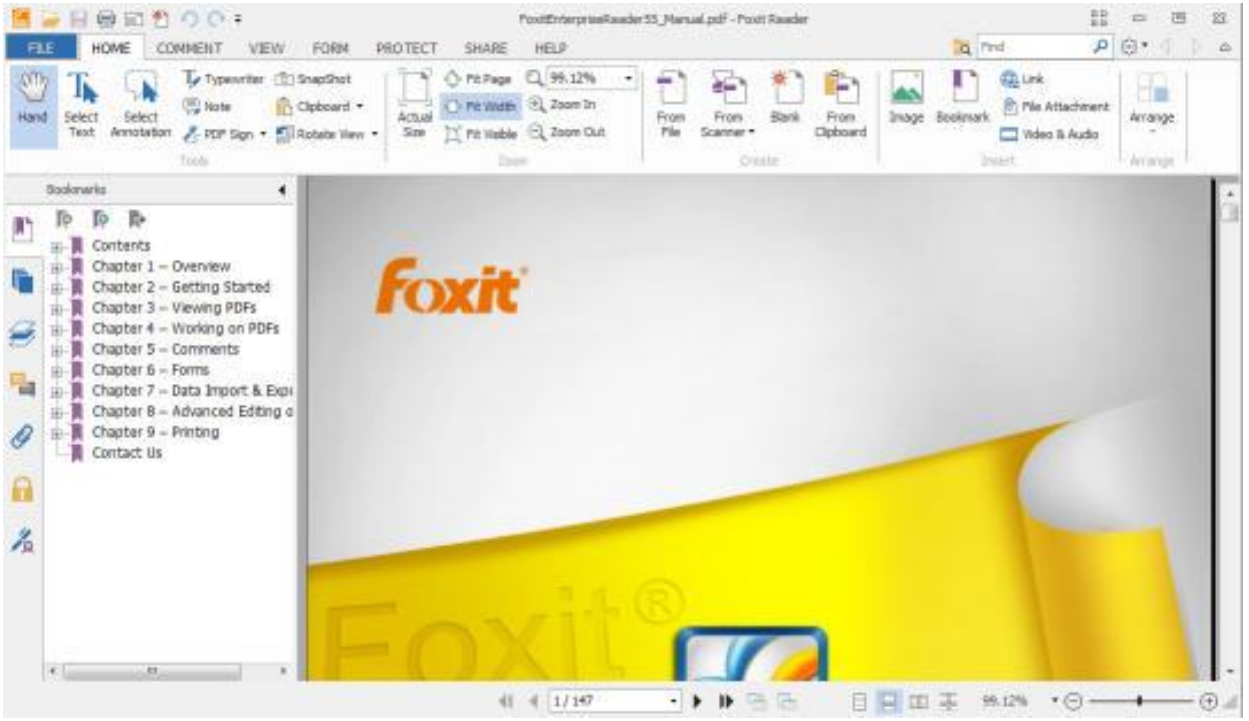
Stack-based buffer overflow in the reader in CoolPDF 3.0.2.256 allows remote attackers to execute arbitrary code via a PDF document with a crafted stream.

[Source http://www.us-cert.gov/cas/bulletins]

[**Common Vulnerability Scoring System** (**CVSS**) is a free and open industry standard for assessing the **severity** of computer system security vulnerabilities]

Foxit PDF editor

- Published: 2013-01-26
- Vulnerability No: CVE-2013-0107
- CVSS Severity Score: 7.6
- Vendor/Product: Foxitsoftware



Stack-based buffer overflow in Foxit Advanced PDF Editor 3 before 3.04 might allow remote attackers to execute arbitrary code via a crafted document containing instructions that reconstruct a certain security cookie.

[Source http://www.us-cert.gov/cas/bulletins]

Oracle jdk

- Published: 2013-01-31
- Vulnerability No: CVE-2012-1489
- CVSS Severity Score: 10.0
- Vendor/Product oracle-jdk

Unspecified vulnerability in the Java Runtime Environment (JRE) component in Oracle Java SE 7 Update 10 and Update 11, when running on Windows using Internet Explorer, Firefox, Opera, and Google Chrome, allows remote attackers to bypass the "Very High" security level of the Java Control Panel and execute unsigned Java code without prompting the user via unknown vectors, aka "Issue 53" and the "Java Security Slider" vulnerability.

[Source http://www.us-cert.gov/cas/bulletins]

Observation 1

- All these problems are due to *(bad) software*

Namely

  - the Linux/Windows/Mac Operating System (OS)
  - the application software
  - the videoconferencing system software
  - the FFmeg graphics engine

    ...

- Such software bugs are why constant patching of system is needed to keep them secure

Observation 2

- All these problems are due to bad software that
  - can be executed/addressed over the network
    - – eg. in case of Slammer worm
  - executes on (untrusted) input obtained over the network
    - – eg. in case of pdf viewers, CMS, routers,...

– or – in the worst case - both

- With ever more network connectivity, ever more software can be attacked

Changing target of attacks

- Traditionally, focus on operating system and network
- “Solutions”
  - regular patching of OS
  - firewalls
  - virus scanners
- Increasingly, focus on
  - web applications
  - web browser
  - mobile devices
    - smartphones, tablet, that pass through firewalls
  - embedded software
    - software in cars, factories, infrastructure...

and targeted attacks on specific organization or person

Changing nature of attackers

- Traditionally, hackers are amateurs motivated by fun
  - publishing attacks for the prestige
- Increasingly, hackers are professional
  - attackers go underground
    - zero-day exploits are worth money
  - attackers include
    - organized crime

      with lots of money and (hired) expertise
    - government agencies

      with even more money & in-house expertise
      - 'Classic' example: Stuxnet (http://www.ted.com/talks/ralph_langner_cracking_stuxnet_a_21st_century_cyberweapon.html)

Software (in)security: crucial facts

- No silver bullets!

  crypto or special security features do not magically solve all problems

  – ***software security ≠ security software***

  – "if you think your problem can be solve by cryptography, you do not understand cryptography and you do not understand your problem" [Bruce Schneier]

- Security is emergent property of entire system

  – just like quality

- (Non-functional) security aspects should be integral part of the design, right from the start

Major causes of (bad) software

- Lack of awareness
  - Was it really a security issue?
- Lack of knowledge
  - I might be aware of it. I just don’t know how to solve it
- Laziness
  - It would take me a day just to fix that, so well…

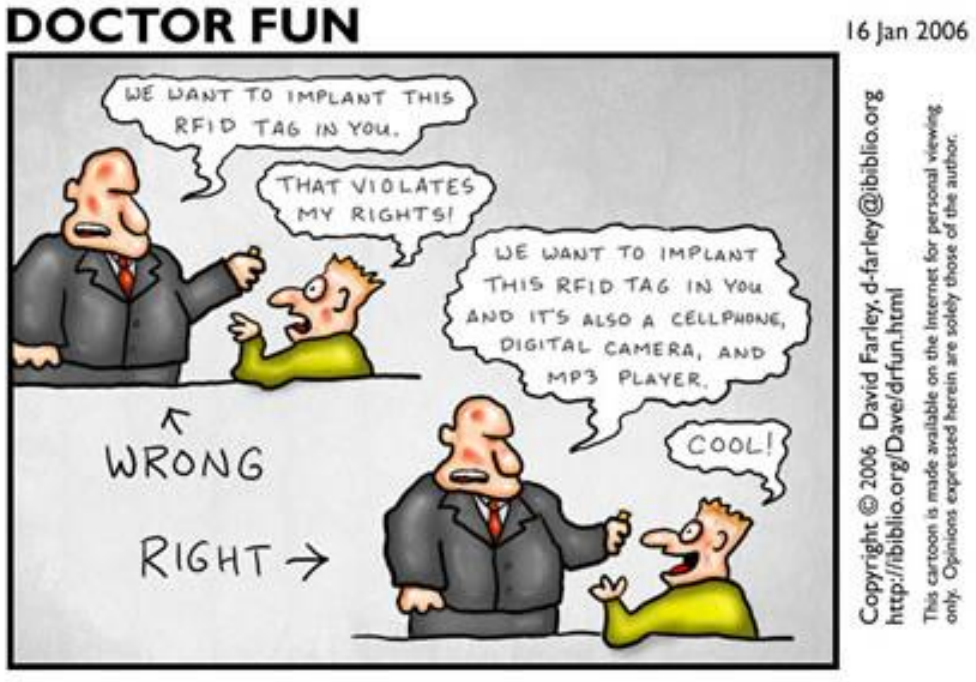
Security is always a secondary concern

- Security is always a secondary concern

  primary goal of software is to provide some functionality or services; managing associated risks is a derived/secondary concern

- There is often a trade-off/conflict between
  - security
  - functionality & convenience

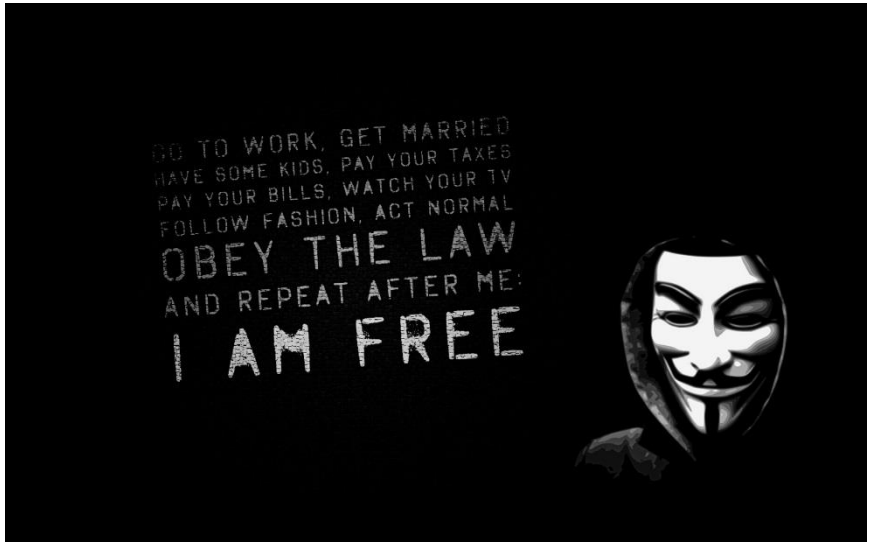
  where security typically looses out
  - more examples of this later...



Functionality vs security

- Functionality is about what software should do
- Security is (also) about what it should not do

- Realizing security threats: Think as a bad guy
  - Unless you think like an attacker, you will be unaware of any potential threats



Functionality vs security: Lost battles?

- operating systems (OSs)
  - with huge OS, with huge attack surface
- programming languages
  - with easy to use, efficient, but very insecure and errorprone mechanisms
- Web browsers
  - with plug-ins for various formats, javascript, ActiveX, VBscript, ...
- email clients
  - which automatically cope with all sorts of formats & attachments..

Weakness

- Software
  - runs on a huge, complicated infrastructure
    - OS, platforms, webbrowser, lots of libraries & APIs, ...
  - is built using complicated languages
    - programming languages, but also SQL, HTML, XML, ...
  - using various tools
    - compilers, IDEs, preprocessors, dynamic code downloads

These may have security holes, or may make the introduction of security holes very easy & likely

Recap

Problems are due to

- lack of awareness
  - of threats, but also of what should be protected
- lack of knowledge
  - of potential security problems, but also of solutions
- compounded by complexity
  - software written in complicated languages, using large APIs , and running on huge infrastructure
- people choosing functionality over security

Next

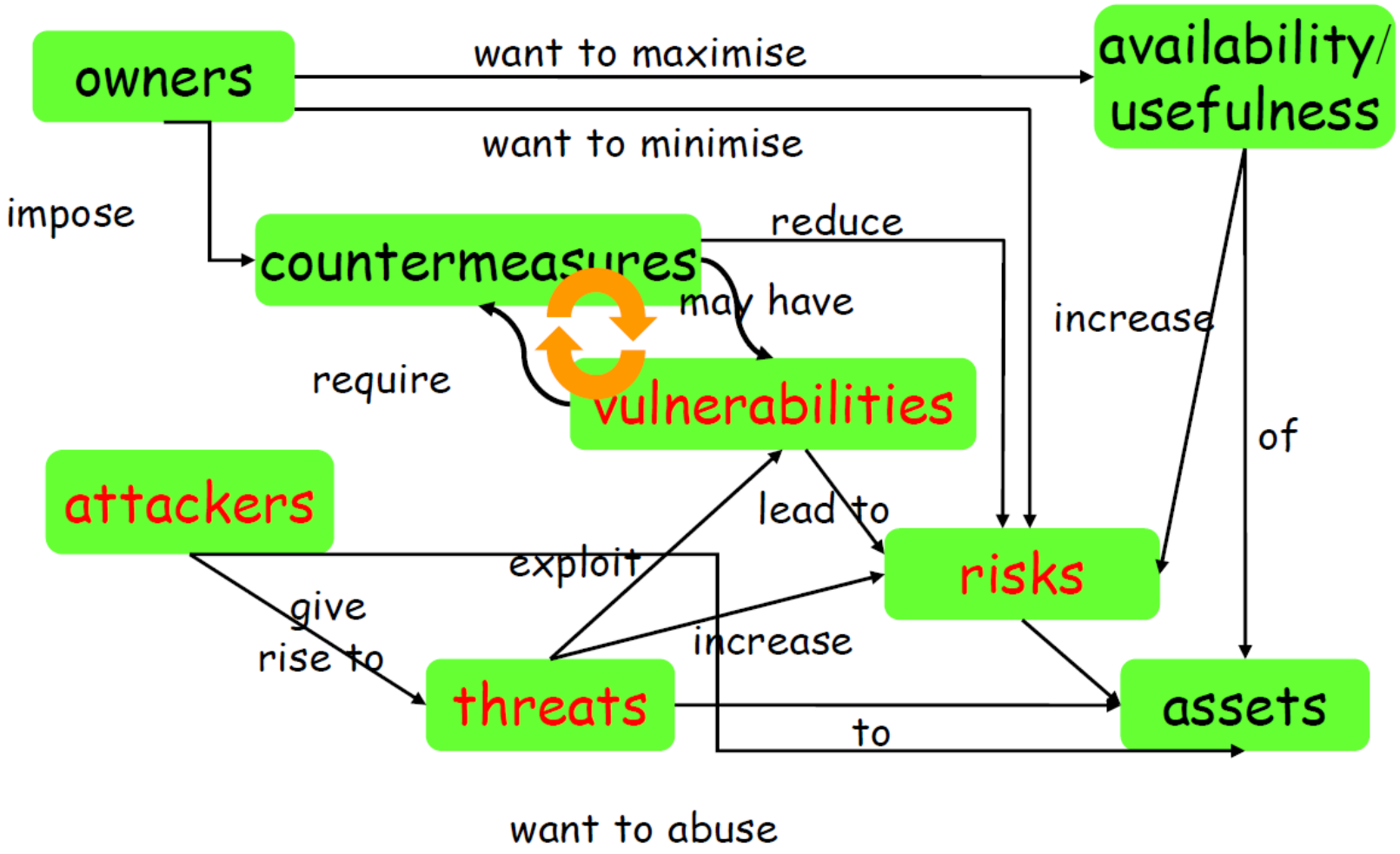
- Security concepts & goals
- Software security
  - Vulnerability
  - Policies
- Setup VMWare Player
- SEED Ubuntu x86 Images

Security concepts & goals

Software and Security

- Security is about regulating access to assets
  - eg. information or functionality
- Software provides functionality
  - eg on-line exam results
- This functionality comes with certain risks
  - eg what are risks of on-line exam results?
- (Software) security is about managing these risks

Security concepts



Starting point for ensuring security

- Any discussion of security should start with an inventory of
  - the stakeholders,
  - their assets, and
  - the threats to these assets

by possible attackers

  - employees, clients, script kiddies, criminals

- Any discussion of security without understanding these issues is meaningless

Security concepts

- Security is about imposing countermeasures to reduce risks to assets to acceptable levels
- A security policy is a specification of what security requirements/goals the countermeasures are intended to achieve
  - secure against what and from whom ?
- Security mechanisms to enforce the policy
- Bottlenecks:
  - expressing what we (don't) want in a policy
  - enforcing this, dynamically or statically