# Lecture 1: Introduction to Computer Security

RK Shyamasundar

# Aims

- Provide a thorough understanding of
  - Policy (what are being protected)
  - Mechanisms (authentication, authorization, auditing/monitoring, …)
  - Attacks (vulnerabilities, malware, …)
  - Assurance: How much can we assure and when?

# Principles of Data and System Security: Syllabus

- CS 745

# References

- **Text Books**
- Security Engineering: A Guide to Building Dependable Distributed Systems, Ross Anderson, 2nd Edition, Wiley, 2008, SBN: 978-0-470-06852-6
- Cryptography and Data Security – Dorothy Denning, Addison Wesley, 1988
- **Research Papers/ chapter**s

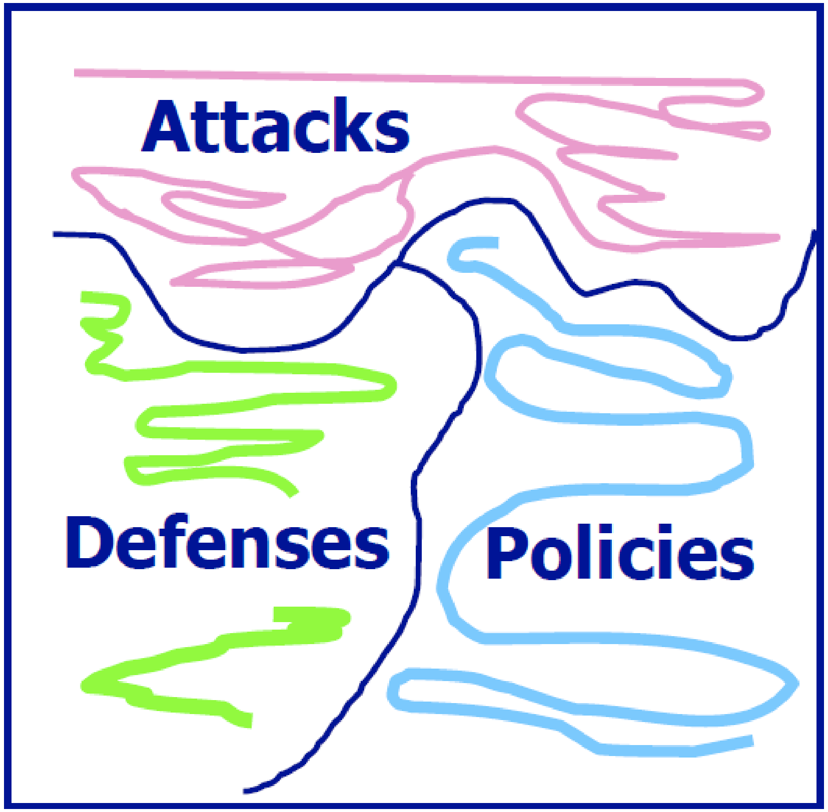
# TAs

– **BS Radhika, radhikabs@cse.iitb.ac.in**

– **Amit Goel, amitgoyal@cse.iitb.ac.in**

– **Mayukh Rath mayuk@cse.iitb.ac.in**

# Security Is All About




**Features**:

- Classes of policies
- Classes of attacks
- Classes of defenses

**Relationships**:

"Defense class D enforces policy class P despite attacks from class A."

"Defense D + Defense D′ = …"

# Objectives

- By the end of the course,
  - you should be able to design policies and mechanisms to protect a system from a given threat model

# Principles of Data and System Security

## Assessment

- Two Exams: Midterm (30%) + Final (35%)
- 1 Group Project (15%) – Presentation/Demo
- 3 Assignments (20% ) – One of them in the Lab
- Attendance Necessary

Note 1: You may collaborate when solving the assignments, however when writing up the solutions you should do so on your own.

Note 2: Group Projects: Everyone should contribute but must be aware of the whole solution

Note 3: Give credit to all assistance (with proper citations): literature, persons.

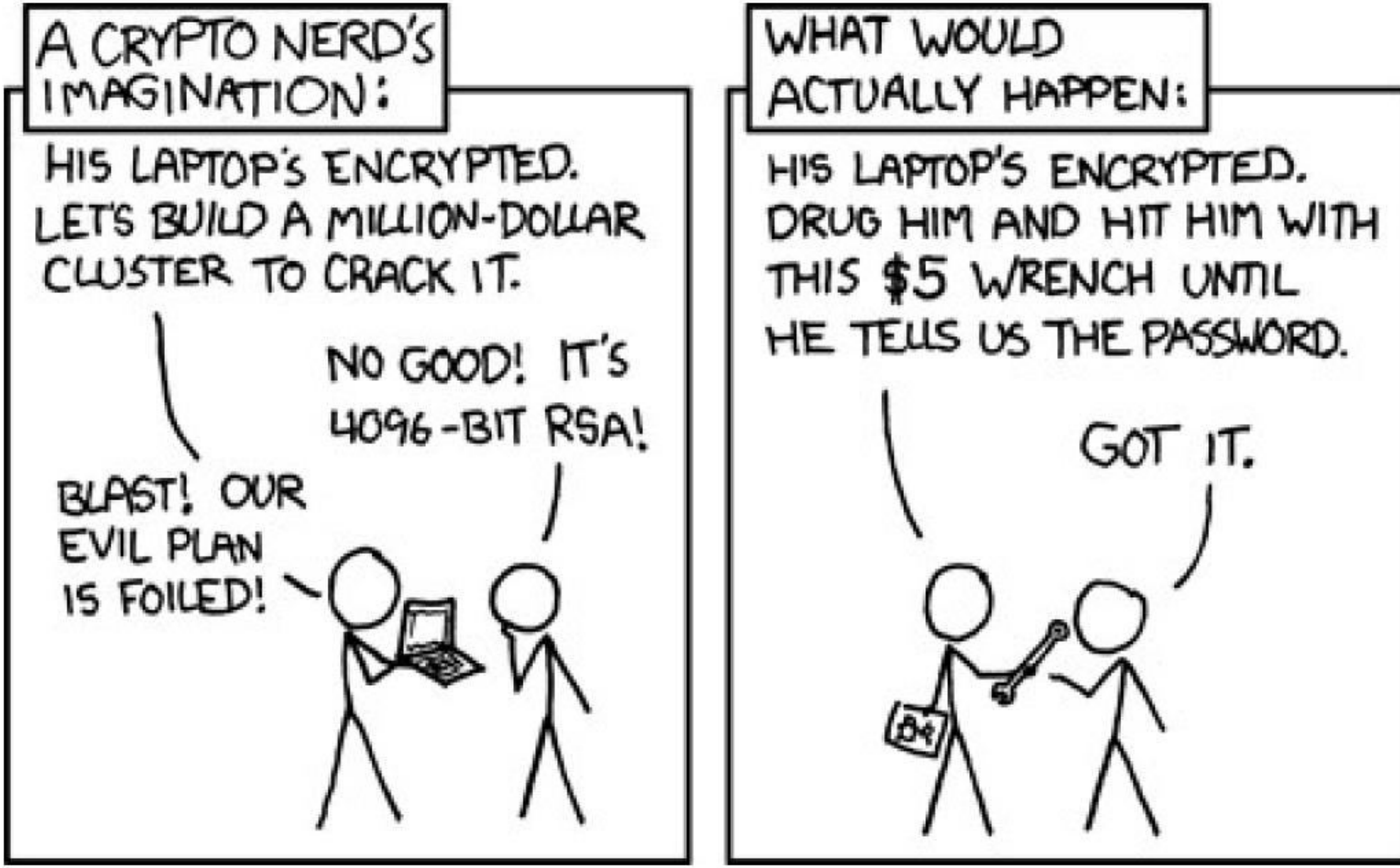
Note 4: Lab Experiments could be Via Cloud access

# What is Security?

- Computers are as secure as real-world systems, and people believe it.
- Most real-world systems are not very secure by any absolute standard
- Why  tolerate such poor security in real-world systems?
- Real world security is not about perfect defenses against determined attackers.
  - Instead, it’s about value, locks, and punishment.
  - The purpose of locks is to raise the threshold of casual break-in
- Why Not Perfect Defense? TOO COSTLY

Whoever thinks his problem can be solved using cryptography, doesn’t understand his problem and doesn’ t understand cryptography.

ATTRIBUTED BY ROGER NEEDHAM AND BUTLER LAMPSON TO EACH OTHER


A CRYPTO NERD'S IMAGINATION:
HIS LAPTOP'S ENCRYPTED. LET'S BUILD A MILLION-DOLLAR CLUSTER TO CRACK IT.
NO GOOD! IT'S 4096-BIT RSA!
BLAST! OUR EVIL PLAN IS FOILED!
WHAT WOULD ACTUALLY HAPPEN:
HIS LAPTOP'S ENCRYPTED. DRUG HIM AND HIT HIM WITH THIS $5 WRENCH UNTIL HE TELLS US THE PASSWORD.
GOT IT.

# What is Computer Security

- Cryptography is nearly perfect; Can computer security be as well?
- NO
  - Software – Complicated Almost never perfect
  - Security set-up gets in the way
  - No quantifiable output

# What is Computer Security

The science of managing malicious intent and behaviour that involves information and communication technology.

- Malicious behaviour can include
  - Fraud/theft – unauthorised access to money, goods or services
  - Vandalism – causing damage for personal reasons (frustration, envy, revenge, curiosity, self esteem, peer recognition, . . . )
  - Terrorism – causing damage, disruption and fear to intimidate
  - Warfare – damaging military assets to overthrow a government
  - Espionage – stealing information to gain competitive advantage
  - Sabotage – causing damage to gain competitive advantage
  - "Spam" – unsolicited marketing wasting time/resources
  - Illegal content – child pornography, Nazi materials, . . .
- Security vs safety engineering:
    - focus on intentional rather than accidental behaviour, presence of intelligent adversary.

# Trustworthy Computer System

- Exhibit all of the functionality users expect,
- Not exhibit any unexpected functionality, and
- Be accompanied by some compelling basis to believe that to be so,

Despite failures of system components, **attacks**, operator errors, and the inevitable design and implementation flaws found in software.

-

# Dependability vs Security

- Dependability = reliability + security
- Reliability and security are often strongly correlated in practice
- But malice is different from error!
  - Reliability: “ Co-author will be able to read this file”
  - Security: “The Pakistan Government won’t be able to read this file”
- Beyond Byzantium
- Proving a negative can be much harder …

# Computer Security

- Focuses on resisting attacks -- one of the factors of Trustworthiness
- Practical Security
  - Tradeoff between Protection and the risk of loss
- Fascinating intellectual discipline, practically a very important area with an enormous number of engineering challenges.

# The computer security problem

Two factors:

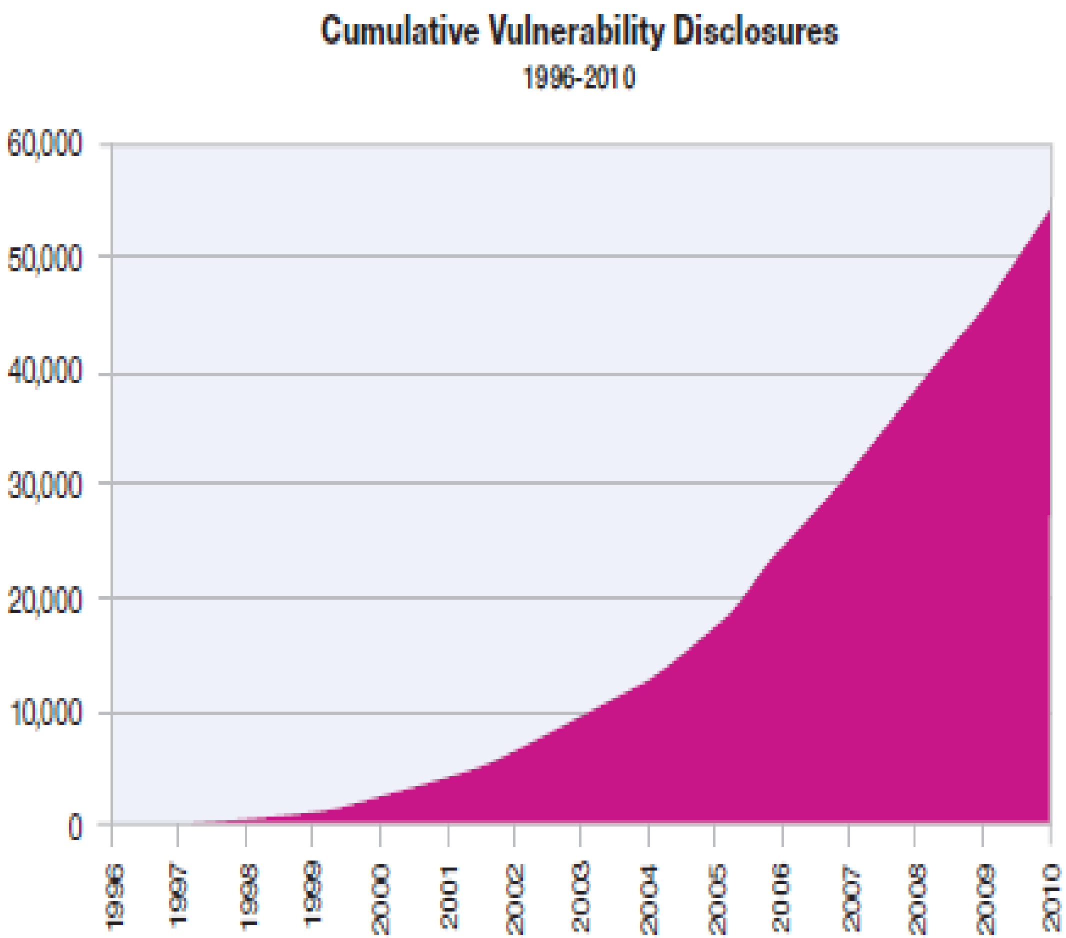
- **Lots of buggy software** (and gullible users)
- **Money can be made from finding and exploiting vulnerabilities**.

1. Marketplace for vulnerabilities
2. Marketplace for owned machines (PPI)
3. Many methods to profit from owned client machines

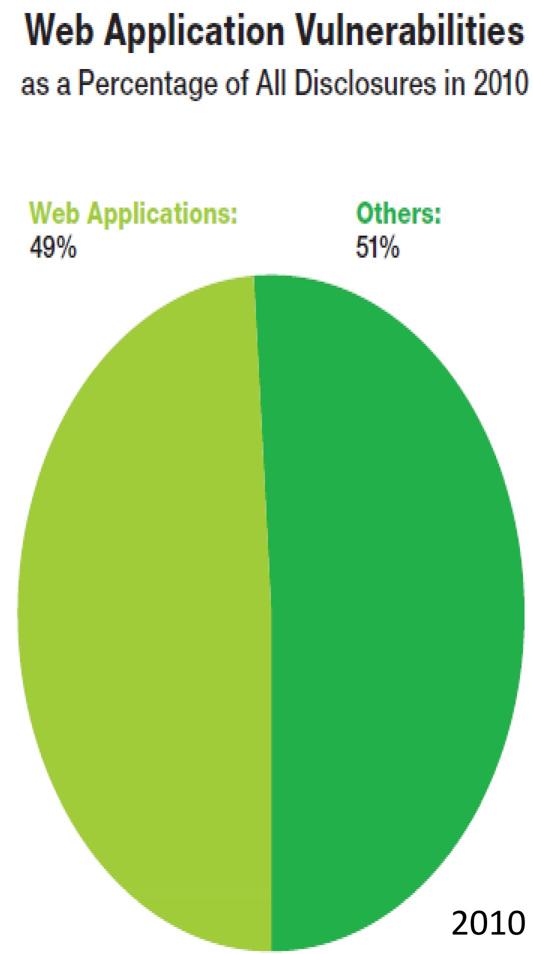
current state of computer security

# MITRE tracks vulnerability disclosures

Cumulative Disclosures




Percentage from Web applications

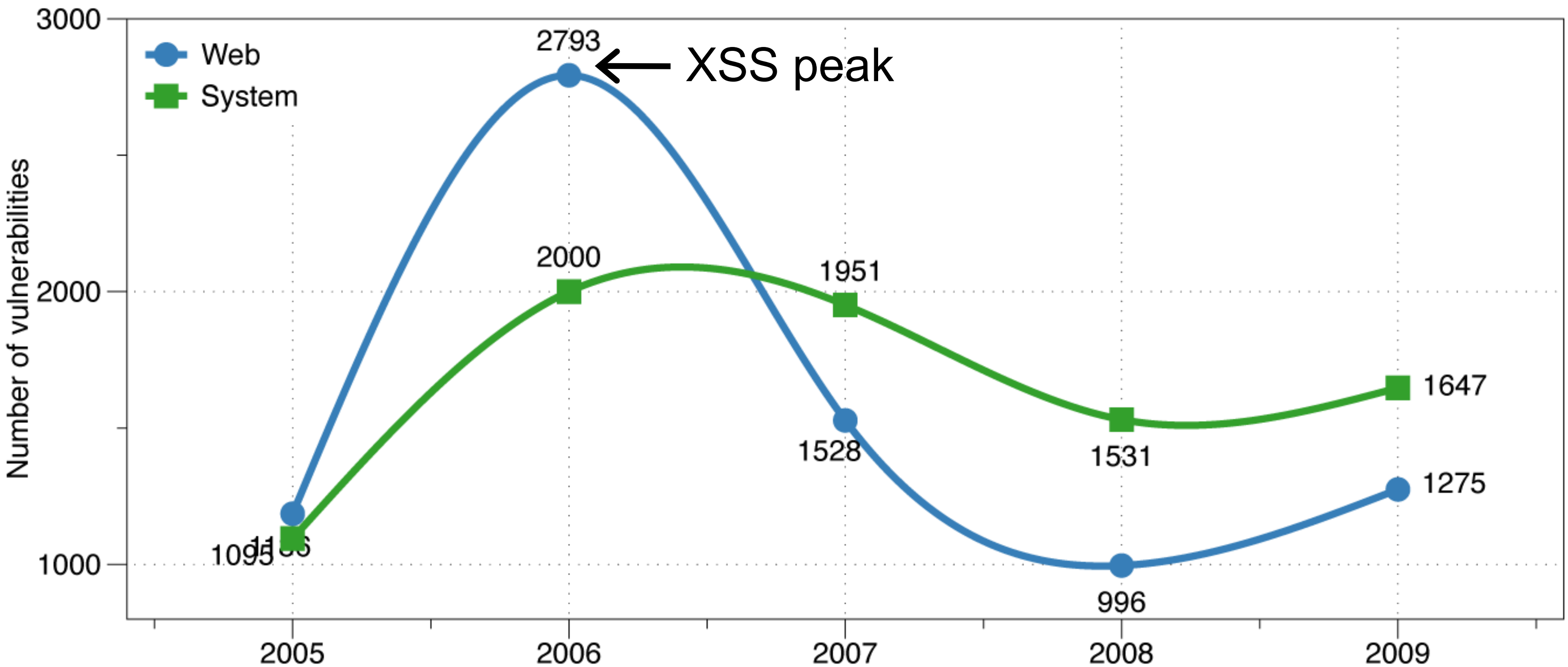



2010

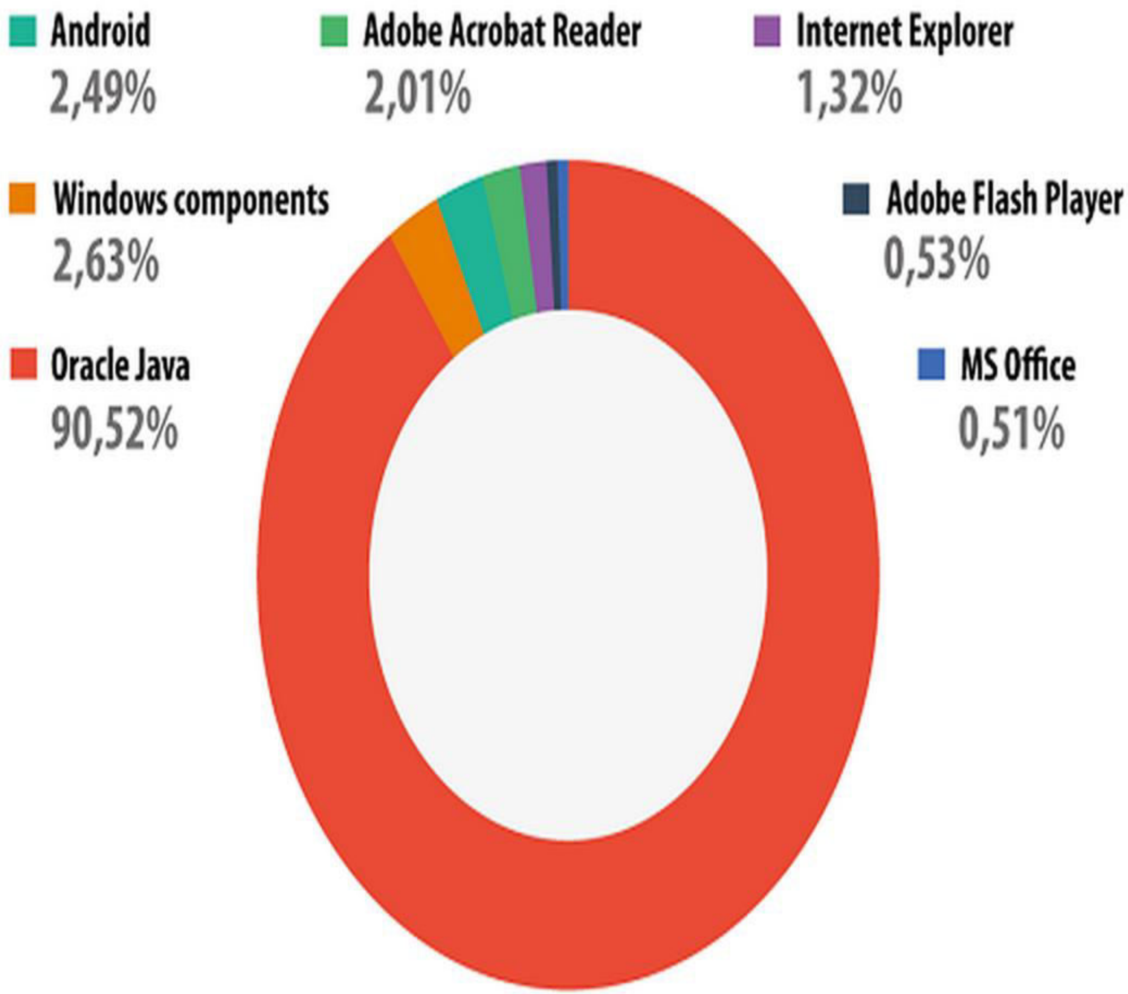
Source: IBM X-Force, Mar 2011

Data: http://cve.mitre.org/

# Web vs System vulnerabilities



# Vulnerable applications being exploited




Source: Kaspersky Security Bulletin 2013

# Marketplace for Vulnerabilities

**Option 1**: bug bounty programs (many)

- Google Vulnerability Reward Program: up to 20K $
- Microsoft Bounty Program: up to 100K $
- Mozilla Bug Bounty program: 500$ - 3000$
- Pwn2Own competition: 15K $

**Option 2**:

- ZDI, iDefense: 2K – 25K $

# Marketplace for Vulnerabilities

**Option 1**:   bug bounty programs (many)

- Google Vulnerability Reward Program:   up to 20K $
- Microsoft Bounty Program:   up to 100K $
- Mozilla Bug Bounty program:  500$ - 3000$
- Pwn2Own competition:   15K $

**Option 2**:

- ZDI,  iDefense:   2K – 25K  $

# Marketplace for Vulnerabilities

**Option 3**: black market

| | |
|---|---|
| ADOBE READER | $5,000-$30,000 |
| MAC OSX | $20,000-$50,000 |
| ANDROID | $30,000-$60,000 |
| FLASH OR JAVA BROWSER PLUG-INS | $40,000-$100,000 |
| MICROSOFT WORD | $50,000-$100,000 |
| WINDOWS | $60,000-$120,000 |
| FIREFOX OR SAFARI | $60,000-$150,000 |
| CHROME OR INTERNET EXPLORER | $80,000-$200,000 |
| IOS | $100,000-$250,000 |

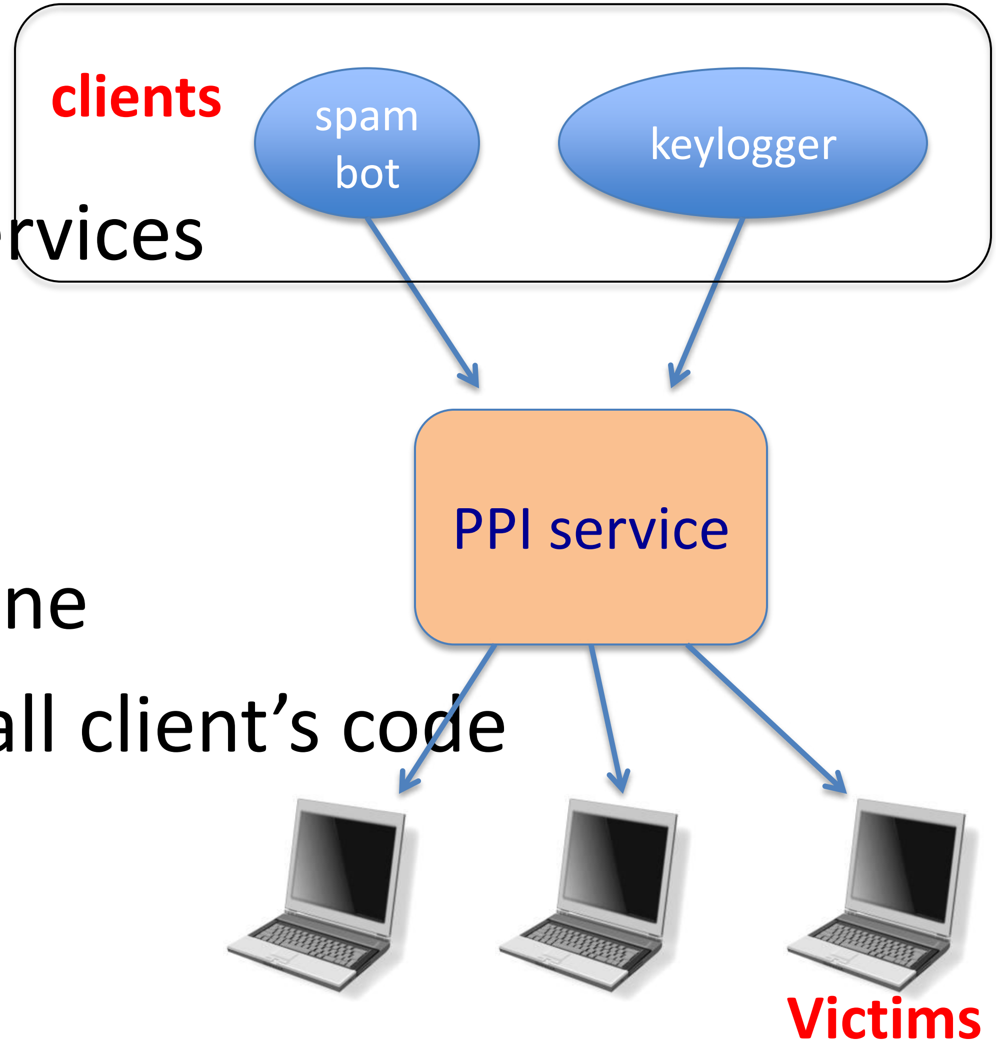
Source: Andy Greenberg (Forbes, 3/23/2012 )

# Marketplace for owned machines

Pay-per-install (PPI) services

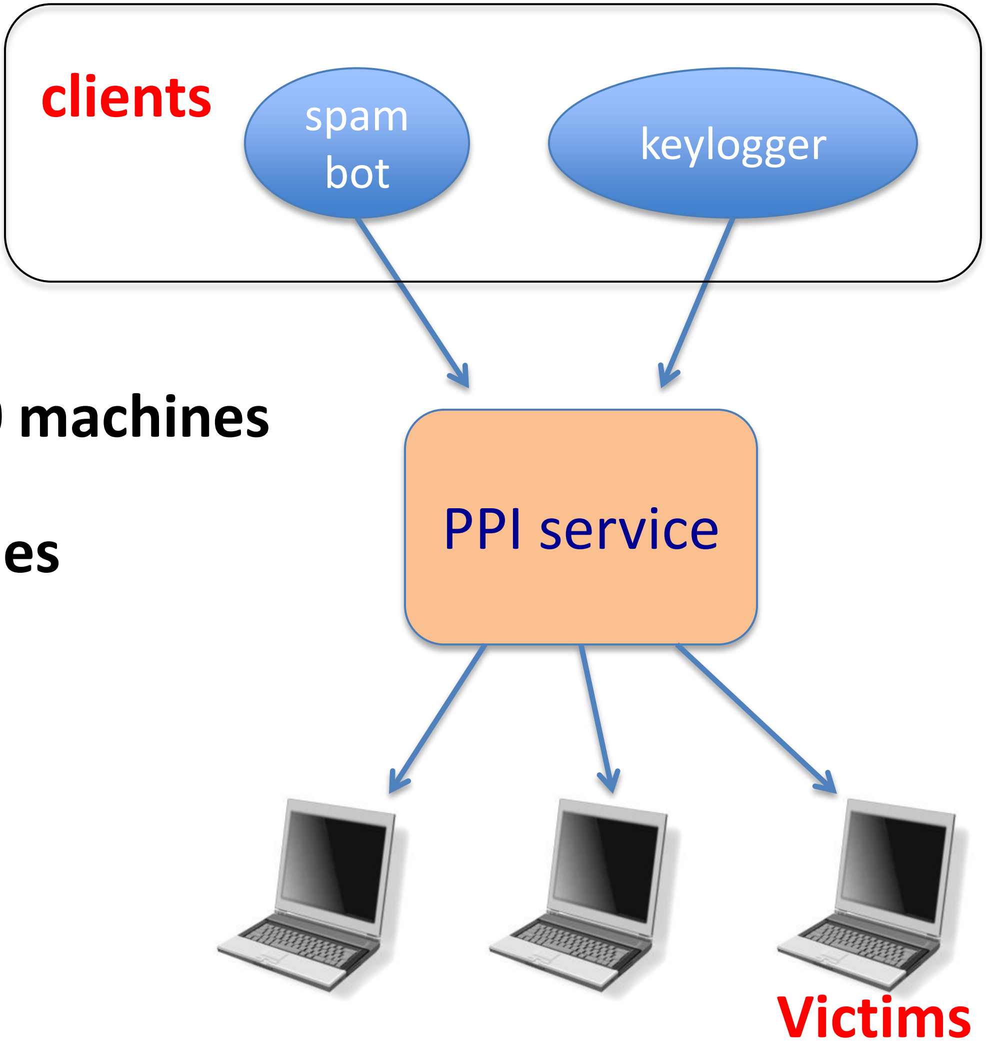
**PPI operation:**

1. Own victim's machine
2. Download and install client's code
3. Charge client




Source: Cabalerro et al. (www.icir.org/vern/papers/ppi-usesec11.pdf)

# Marketplace for owned machines




Cost: **US - 100-180$ / 1000 machines**

**Asia - 7-8$ / 1000 machines**

Source: Cabalerro et al. (www.icir.org/vern/papers/ppi-usesec11.pdf)

# Process of Science

*We never are definitely right;*
*we can only be sure when we are wrong.*

Richard Feynman
*Lectures on the Character of Physical Law*

# Secure or Insecure

**Insecure!**

- *Suppose we have a precisely defined security claim about a system,*

*from which we can derive the consequences which can be tested,*

- *Then **in principle** we can prove that the system is **insecure**.*

**Secure?**

- *Suppose you design a system, derive some security claims, and discover every time that the system remains secure under all tests.*
- *Is the system  then secure?*
- *No, it is simply not proved insecure.*
- *In the future you could refine the security model, there could be a wider range of tests and attacks, and **you might then discover that the thing is insecure.***

# Importance of Computer Security

**Wide ubiquitous usage of computers and Internet, need to ensure continuous dependable operations:**

- **Business environment**: legal compliance, cash flow, profitability, commercial image and shareholder confidence, product integrity, intellectual property and competitive advantage
- **Military environment**: exclusive access to and effectiveness of weapons, electronic countermeasures, communications secrecy, identification and location information, automated defenses
- **Medical environment:** confidentiality and integrity of patient records, unhindered emergency access, equipment safety, correct diagnosis and treatment information
- **Households**: privacy, correct billing, burglar alarms
- **Society at large:** Utility/Infrastructure services, communications, transport, tax/benefits collection, goods supply, . . .

# Studying Security of a System

- Specification/Policy: What is the system supposed to do?
- Implementation/Mechanism: How does it realize it?
- Correctness/Assurance: Does it really work?

# POLICY: SPECIFYING SECURITY

**Specify the needs of stakeholders**

- **Confidentiality/Secrecy:** Controlling who gets to read information.
- **Integrity:** controlling how information changes
- **Availability:** providing prompt access to information and resources
- Accountability: knowing who has had access to information or resources.

# Aspects of Integrity and Availability Protection

- **Rollback** – ability to return to a well-defined valid earlier state backup, revision control, undo function)
- **Authenticity** – verification of the claimed identity of a communication partner
- **Non-repudiation** – origin and/or reception of message cannot be denied in front of third party
- **Audit** – monitoring and recording of user-initiated events to detect and deter security violations
- **Intrusion detection** – automatically notifying unusual events

- **Optimistic security:** Temporary violations of security policy are tolerated where correcting the situation is easy and the violator is accountable. (Applicable to integrity and availability, but usually not to confidentiality requirements.)

# Dangers Being Protected Against

- Damage to information
- Disruption of service
- Theft of physical resources like money
- Theft of information
- Loss of privacy

- Integrity
- Availability
- Integrity
- Secrecy (confidentiality)
- Secrecy (confidentiality)

# Taxonomy of Cybersecurity Threats

- Incomplete, inquisitive, and unintentional blunders.
- Hackers driven by technical challenges.
- Disgruntled employees or customers seeking revenge.
- Criminals interested in personal financial gain, stealing services, or industrial espionage.
- Organized crime with the intent of hiding something or financial gain.
- Organized terrorist groups attempting to influence U.S. policy by isolated attacks.
- Foreign espionage agents seeking to exploit information for economic, political, or military purposes.
- Tactical countermeasures intended to disrupt specic weapons or command structures.
- Multifaceted tactical information warfare applied in a broad orchestrated manner to disrupt a major military mission.
- Large oganized groups or nation-states intent on overthrowing a government.

# Variants of confidentiality

- Data protection/personal data privacy – fair collection and use of personal data, in Europe a set of legal requirements
- Anonymity/untraceability – ability to use a resource without disclosing identity/location
- Unlinkability – ability to use a resource multiple times without others being able to link these uses together
  - HTTP “cookies” and the Global Unique Document Identifier (GUID) in Microsoft Word documents were both introduced to provide linkability.
- Pseudonymity – anonymity with accountability for actions.
- Unobservability – ability to use a resource without revealing this activity to third parties
  - low probability of intercept radio, steganography, information hiding
- Copy protection
- Information flow control- ability to control the use and flow of information
- Further details: Pfitzmann/Kohntopp: http://www.springerlink.com/link.asp?id=xkedq9pftwh8j752

# MECHANISM: IMPLEMENTING SECURITY

- Security Implementation:
  - Code: The actual program on which the security depends
  - Setup: data that controls the programs' operations: folder structure, access control lists, group memberships, user passwords or encryption keys, and so on.
- Implementation must defend against:
  - Bad, buggy and hostile vulnerabilities

# Broad Defensive Startegies

- Isolate—keep everybody out
  - coarse-grained strategy provides the best security, but it keeps users from sharing info. or services.
  - impractical for all but a few applications.
- Exclude—keep the bad guys out
  - Medium grained strategy makes it all right for programs inside this defense to be gullible. Code signing and firewalls do this.
- Restrict—let the bad guys in, but keep them from doing damage.
  - Fine-grained strategy, also known as sandboxing, can be implemented traditionally with an OS process or with a more modern approach that uses a Java virtual machine.
  - Sandboxing typically involves access control on resources to define the holes in the sandbox. Programs accessible from the sandbox must be paranoid, and it's hard to get this right.
- Recover—undo the damage.
  - Exemplified by backup systems and restore points, doesn't help with secrecy, but it does help with integrity and availability.
- Punish—catch the bad guys and prosecute them.
  - Auditing and police do this.