CSE 484 / CSE M 584 (Winter 2009)

Computer Security and Privacy

Tadayoshi Kohno

Thanks to Dan Boneh, Dieter Gollmann, John Manferdelli, John Mitchell, Vitaly Shmatikov, Bennet Yee, and many others for sample slides and materials ...

High-level information

- Instructor: Tadayoshi Kohno (Yoshi)
  - Office: CSE 558
  - Office hours: Mondays, 10:30 to 11:20am (right after class, may change)
  - Open door policy – don’t hesitate to stop by!
- TAs: Jonathan Beall, Alexei Czeskis and Tamara Denning
  - Office/hours: See website (TBD)
- Course website
  - Assignments, reading materials, lecture notes, ...
- Course email list (and blog)
  - Discussions, announcements

Prerequisites (CSE 484)

- Required: Data Structures (CSE 326)
- Required: Machine Org and Assembly (CSE 378)
- Assume: Working knowledge of C and assembly
  - One of the projects involves writing buffer overflow attacks in C
  - You must have detailed understanding of x86 architecture, stack layout, calling conventions, etc.
- Assume: Working knowledge of software engineering tools for Unix environments (gdb, etc)
- Assume: Working knowledge of Java and JavaScript

Prerequisites (CSE 484)

- Recommended: Computer Networks; Operating Systems
  - Will help provide deeper understanding of security mechanisms and where they fit in the big picture
- Recommended: Complexity Theory; Discrete Math; Algorithms
  - Will help with the more theoretical aspects of this course.

Prerequisites (CSE 484)

- Most of all: Eagerness to learn!
  - This is a 400 level course.
  - I expect you to push yourself to learn as much as possible.
  - I expect you to be a strong, independent learner capable of learning new concepts from the lectures, the readings, and on your own.

Prerequisites (CSE M 584)

- All the previous prerequisites, plus
  - Admission to 5-th year Masters program
  - CSE 378 and one of CSE 451 / CSE 461

Course Logistics (CSE 484)

- Lectures: Mon, Wed: 9--10:20am ; Recitations: Thurs: 8:30--9:20am and 9:30--10:20am
- Security is a contact sport!
- Labs (45% of the grade)
  - Labs involve a lot of programming
  - Can generally be done in teams of 3 students (see specific lab descriptions for details)
- Homeworks (30% of grade)
  - Textbook-style questions (15%)
  - Blog entries (15%)
- Final (25% of the grade)

Exceptional work may be rewarded with extra credit

No make-up or substitute exams!
If you are not sure you will be able to take the exam on the assigned date and time, **do not take this course!**

Course Logistics (CSE M 584)

- Same as before, but...
- Labs (40% of the grade)
- Homeworks (25% of grade)
  - Textbook-style questions (12.5%)
  - Blog entries (12.5%)
- Final (25% of the grade)
- Research readings
  - Read research papers (1 per week for first 9 weeks)
  - Present one of these papers to the class (last week of class)

Late Submission Policy

- Late assignments will (generally) be dropped 20% per day.
  - Late days will be rounded up
  - So an assignment turned in 1.25 days late will be downgraded 40%.
  - Some exceptions.
    - Blog postings must be on time
    - See website for additional exceptions
- Everything is generally due on Friday

Course Materials

- Textbooks:
  - Daswani, Kern, Kesavan, "Foundations of Security"
  - Handouts (printed, not available online)
  - Additional materials linked to from course website
- Attend lectures.
  - Lectures will not follow the textbooks
  - Lectures will focus on "big-picture" principles and ideas
  - Lectures will cover some material that is not in the textbook – and you will be tested on it! (Also make sure to read the blog)

Other Helpful Books (all online)

- Ross Anderson, “Security Engineering” (1st edition)
  - Focuses on design principles for secure systems
  - Wide range of entertaining examples: banking, nuclear command and control, burglar alarms
  - You should all at least look at the Table of Contents for this book.
- Kaashoek and Saltzer, “Principles of Computer System Design”
- Menezes, van Oorschot, and Vanstone, “Handbook of Applied Cryptography”

Others books, movies, ...

- Pleasure books:
  - Little Brother by Cory Doctorow
    - Available online here http://craphound.com/littlebrother/download/
    - I highly recommend that everyone reads this
    - Who knows, there might even be an extra credit question or two about Little Brother on the final...
  - Cryptonomicon by Neal Stephenson
- Movies
  - Hackers
  - Sneakers
  - Diehard 4
- Historical
  - The Codebreakers by David Kahn
  - The Code Book by Simon Singh

Ethics

- In this class you will learn about how to attack the security and privacy of (computer) systems.
- Knowing how to attack systems is a critical step toward knowing how to protect systems.
- But one must use this knowledge in an ethical manner.
- In order to get a non-zero grade in this course, you must sign and return the "Security and Privacy Code of Ethics" form by the end of Section on Thursday (Jan 8).
- http://www.cs.washington.edu/education/courses/484/09wi/administrivia/ethics.pdf

Mailing List

- Make sure to sign up for the mailing list
- URL for mailing list on course website:
  - http://www.cs.washington.edu/education/courses/484/09wi/administrivia/email.html
- Used for announcements

Forum

- We’ve set up a forum for this course
  - https://catalysttools.washington.edu/gopost/board/kohno/9176/
- Please us it to discuss the homeworks and labs and other general class materials

Homeworks

- Tentative schedule online (future dates subject to change based on progress, etc)
- General plan:
  - 5 homeworks, once every two weeks
    - Jan 16, Jan 30, Feb 13, Feb 27, March 13
    - First one posted online now
  - Due Fridays at 11am.
  - Submit to Catalyst system (URL on course page)
- http://www.cs.washington.edu/education/courses/484/09wi/homework/index.html

Labs

- Tentative schedule online (future dates subject to change based on progress, etc)
- General plan:
  - 4, once every two weeks
    - Jan 23, Feb 6, Feb 20, March 6
    - First one posted online on Wednesday
  - Due Fridays at 11am.
  - Submit to Catalyst system (URL on course page)
  - Groups of three generally allowed (check each project page for details)
- http://www.cs.washington.edu/education/courses/484/09wi/projects/index.html

Labs

- First lab: Software security
  - Buffer overflow attacks, double-free exploits, format string exploits, ...
- Second lab: Web security
  - XSS attacks, ...
- Third lab: Network security
  - Sniffing packets, probing networks, ...
- Fourth lab: Host security
  - Switchblade (tentative)

Blog

- We’ll talk about this in a little bit...

What does “security” mean to you?

Two key themes of this course

- How to **think** about security
  - The Security Mindset - “new” way to think about systems
  - Threat models, security goals, assets, risks, adversaries
  - Connection between security, technology, politics, ethics, ...
  - The first few lectures, and the blog
    - http://cubist.cs.washington.edu/Security/
    - http://slashdot.org/
- **Technical aspects** of security
  - Attack techniques
  - Defenses

Technical Themes

- Vulnerabilities of computer systems
  - Software problems (buffer overflows); crypto problems; network problems (DoS, worms); people problems (usability, phishing)
- Defensive technologies
  - Protection of information in transit: cryptography, security protocols
  - Protection of networked applications: firewalls and intrusion detection
  - "Defense in depth"

What This Course is Not About

- Not a comprehensive course on computer security
  - Computer security is a broad discipline!
  - Impossible to cover everything in one quarter
  - So be careful in industry or wherever you go!
- Not about all of the latest and greatest attacks
  - Read bugtraq or other online sources instead
- Not a course on ethical, legal or economic issues
  - We will touch on ethical issues, but not focus on them
- Not a course on how to “hack” or “crack” systems
  - Yes, we will learn about attacks … but the ultimate goal is to develop an understanding of attacks so that you can build more secure systems

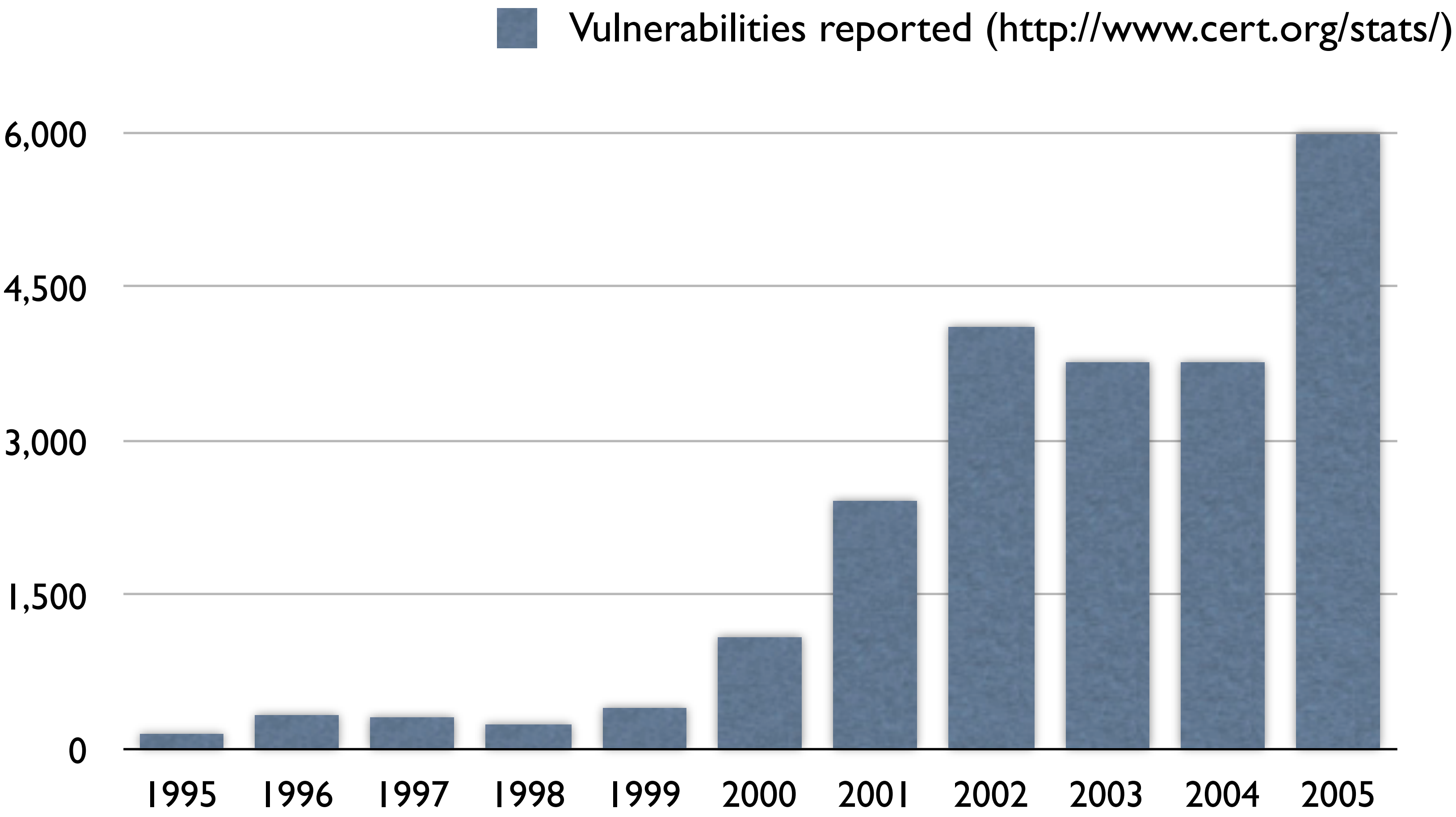
What is Computer Security?

- Systems may fail for many reasons
- Reliability deals with accidental failures
- Usability deals with problems arising from operating mistakes made by users
- Security deals with intentional failures created by intelligent parties
  - Security is about computing in the presence of an adversary
  - But security, reliability, and usability are all related

What Drives the Attackers?

- Adversarial motivations:
  - Money, fame, malice, curiosity, politics, terror....
- Fake websites, identity theft, steal money and more
- Control victim’s machine, send spam, capture passwords
- Industrial espionage and international politics
- Access copy-protected movies and videos
- Attack on website, extort money
- Wreak havoc, achieve fame and glory

Growing Problem



Challenges: What is "Security?"

- What does security mean?
  - Often the hardest part of building a secure system is figuring out what security means
  - What are the assets to protect?
  - What are the threats to those assets?
  - Who are the adversaries, and what are their resources?
  - What is the security policy?
- Perfect security does not exist!
  - Security is not a binary property
  - Security is about risk management

From Policy to Implementation

- After you've figured out what security means to your application, there are still challenges
  - How is the security policy enforced?
  - Design bugs
    - Poor use of cryptography
    - Poor sources of randomness
    - ...
  - Implementation bugs
    - Buffer overflow attacks
    - ...
  - Is the system usable?

Don't forget the users! They are a critical component!

Many Participants

- Many parties involved
  - System developers
  - Companies deploying the system
  - The end users
  - The adversaries (possibly one of the above)
- Different parties have different goals
  - System developers and companies may wish to optimize cost
  - End users may desire security, privacy, and usability
  - But the relationship between these goals is quite complex (will customers choose not to buy the product if it is not secure?)

Other (Mutually-Related) Issues

- Do consumers actually care about security?
- Security is expensive to implement
- Plenty of legacy software
- Easier to write “insecure” code
- Some languages (like C) are unsafe

Approaches to Security

- Prevention
  - Stop an attack
- Detection
  - Detect an ongoing or past attack
- Response
  - Respond to attacks

- The threat of a response may be enough to deter some attackers

Blog and Security Reviews

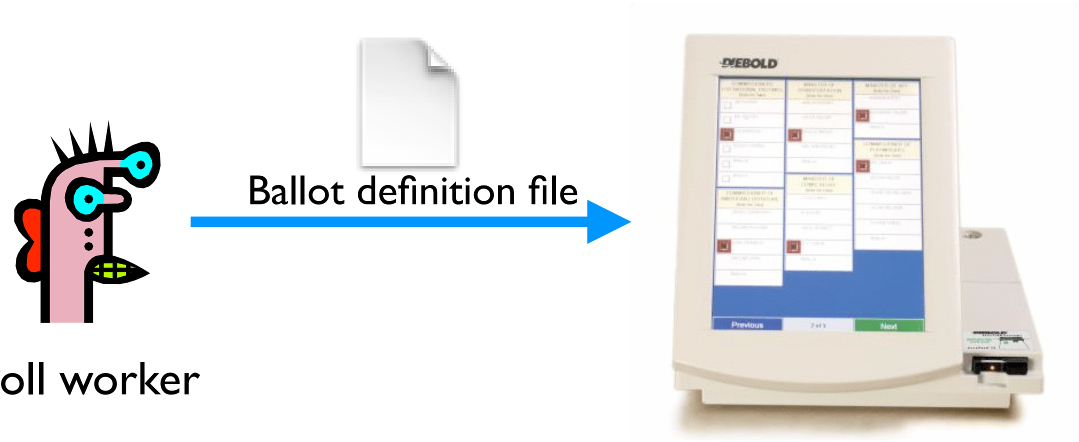
- Previous courses looked at
  - Nike+iPod Sport Kit
  - Wireless keyboards
  - iPhone
  - Zune
  - SlingBox
  - Nintendo Wii
  - Dodgeball
  - Netflix
  - …
- Blog URL: http://cubist.cs.washington.edu/Security/
- Past Security Reviews: http://cubist.cs.washington.edu/Security/category/security-reviews/

Example: Electronic Voting

- Popular replacement to traditional paper ballots

AP

Pre-Election

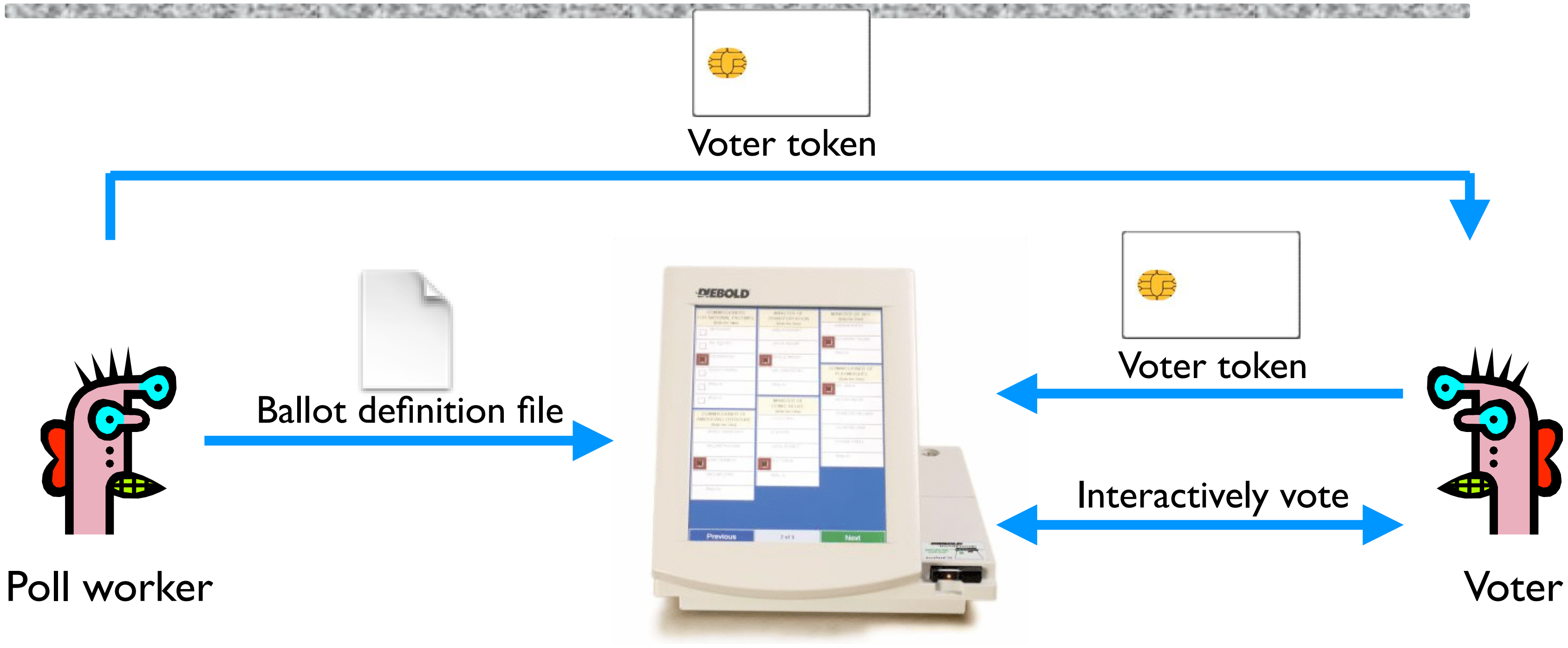


Ballot definition file

Poll worker

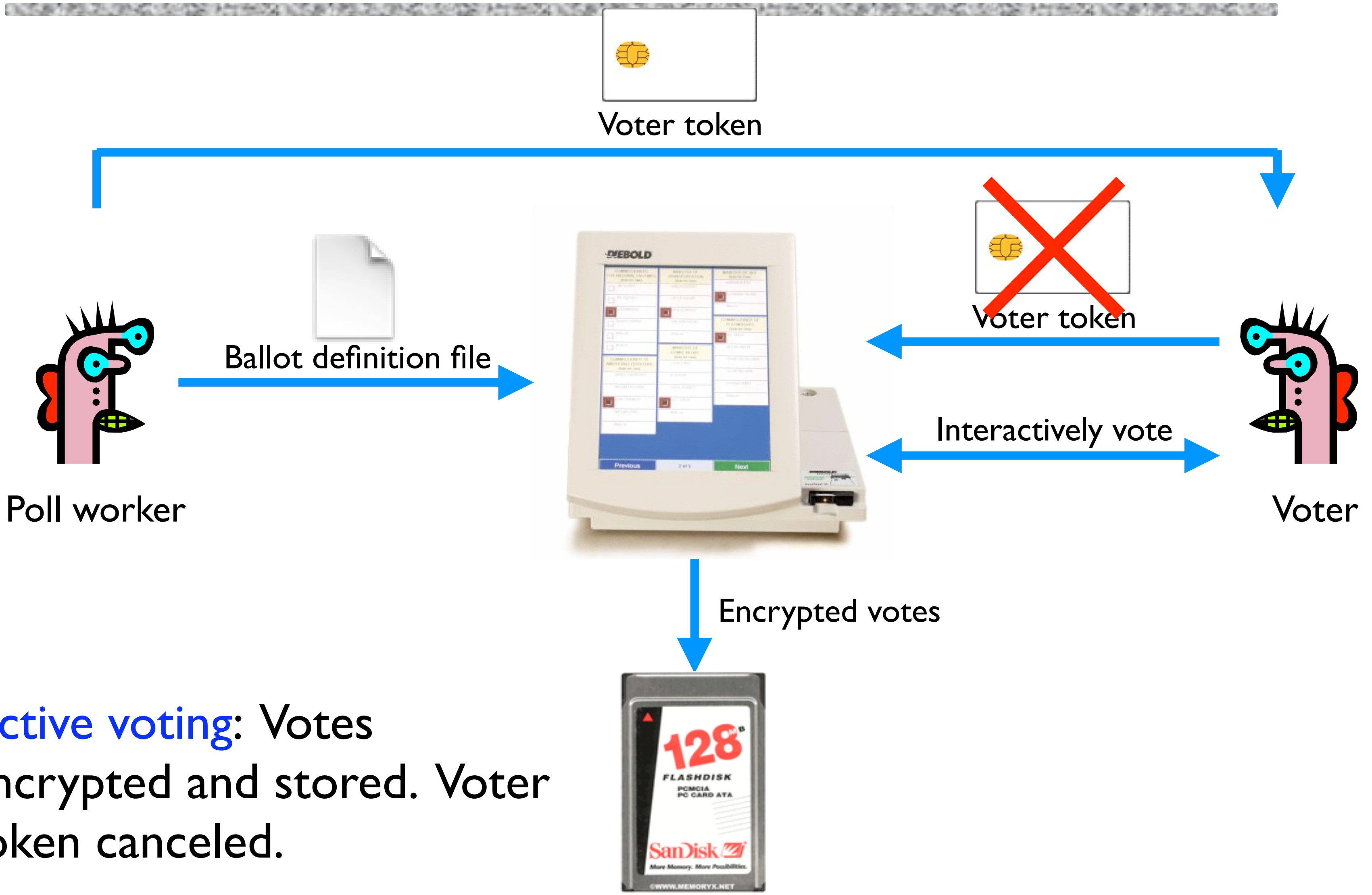
Pre-election: Poll workers load “ballot definition files” on voting machine.

Active Voting



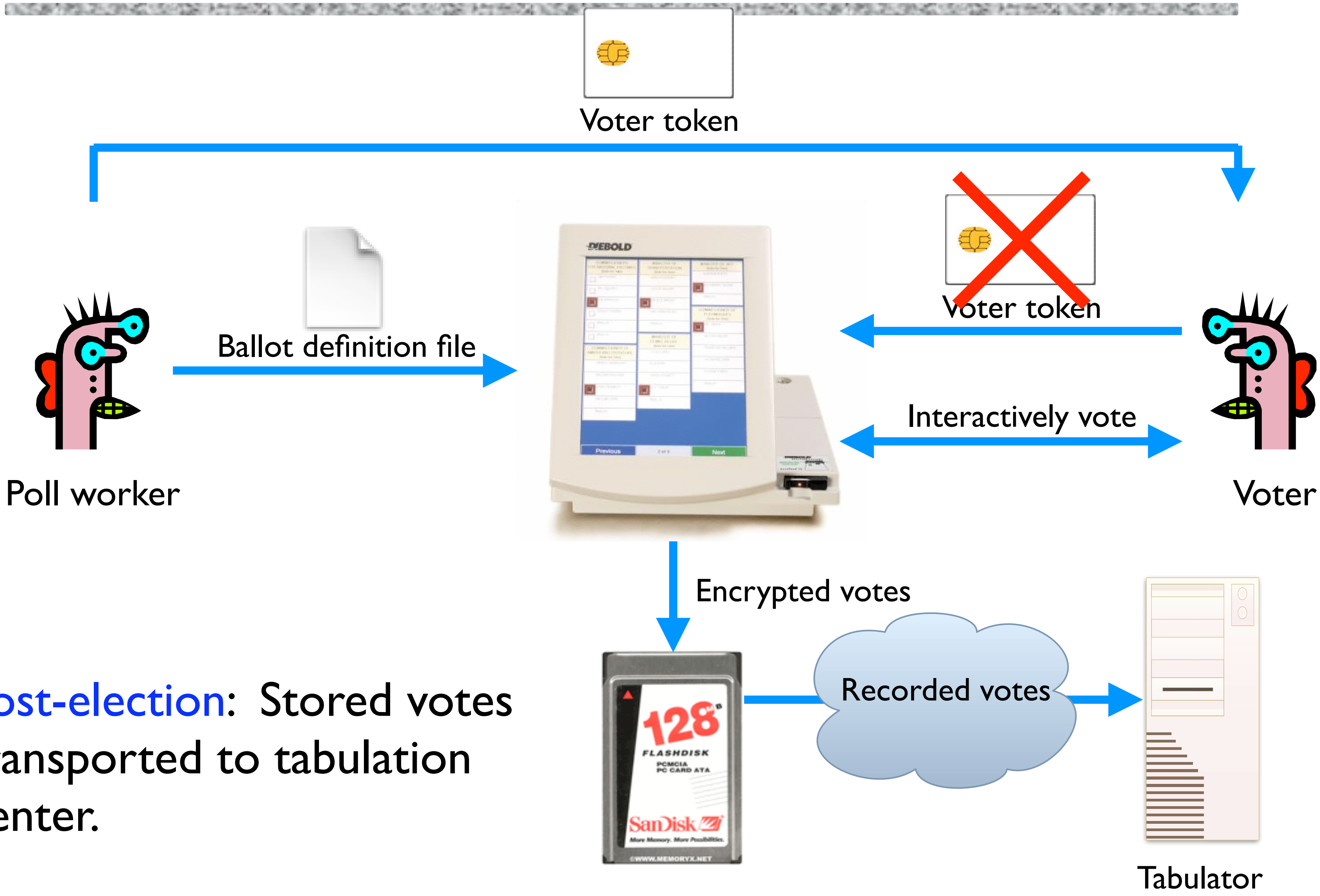
Active voting: Voters obtain single-use tokens from poll workers. Voters use tokens to active machines and vote.

Active Voting



Active voting: Votes encrypted and stored. Voter token canceled.

Post-Election

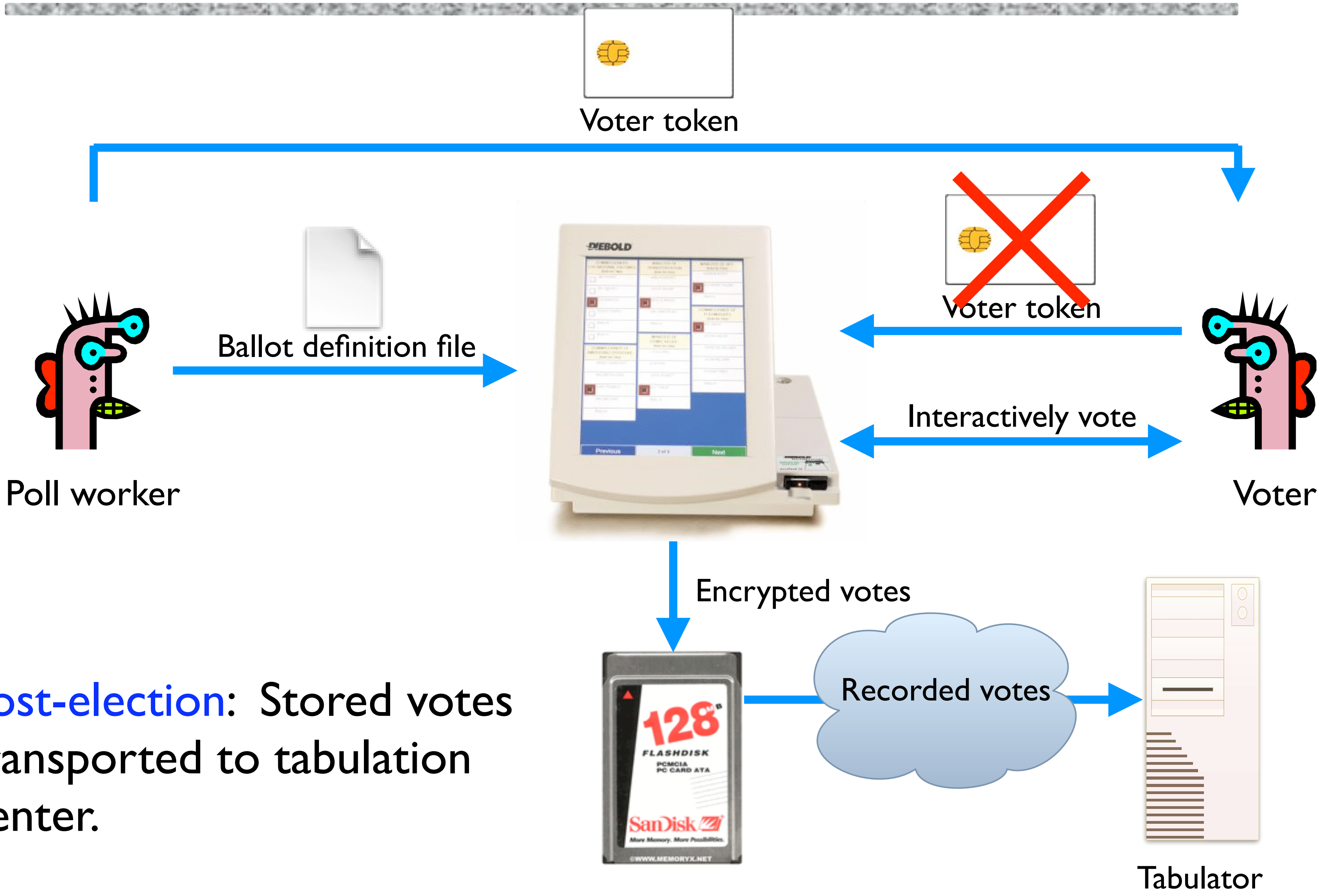


Post-election: Stored votes transported to tabulation center.

Security and E-Voting (Simplified)

- Functionality goals:
  - Easy to use
  - People should be able to cast votes easily, in their own language or with headphones for accessibility
- Security goals:
  - Adversary should not be able to tamper with the election outcome
    - By changing votes
    - By denying voters the right to vote
  - Is it OK if an adversary can do the above, assuming you can catch him or her or them?
  - Adversary should not be able to figure out how voters vote

Can You Spot Any Potential Issues?



Post-election: Stored votes transported to tabulation center.

Potential Adversaries

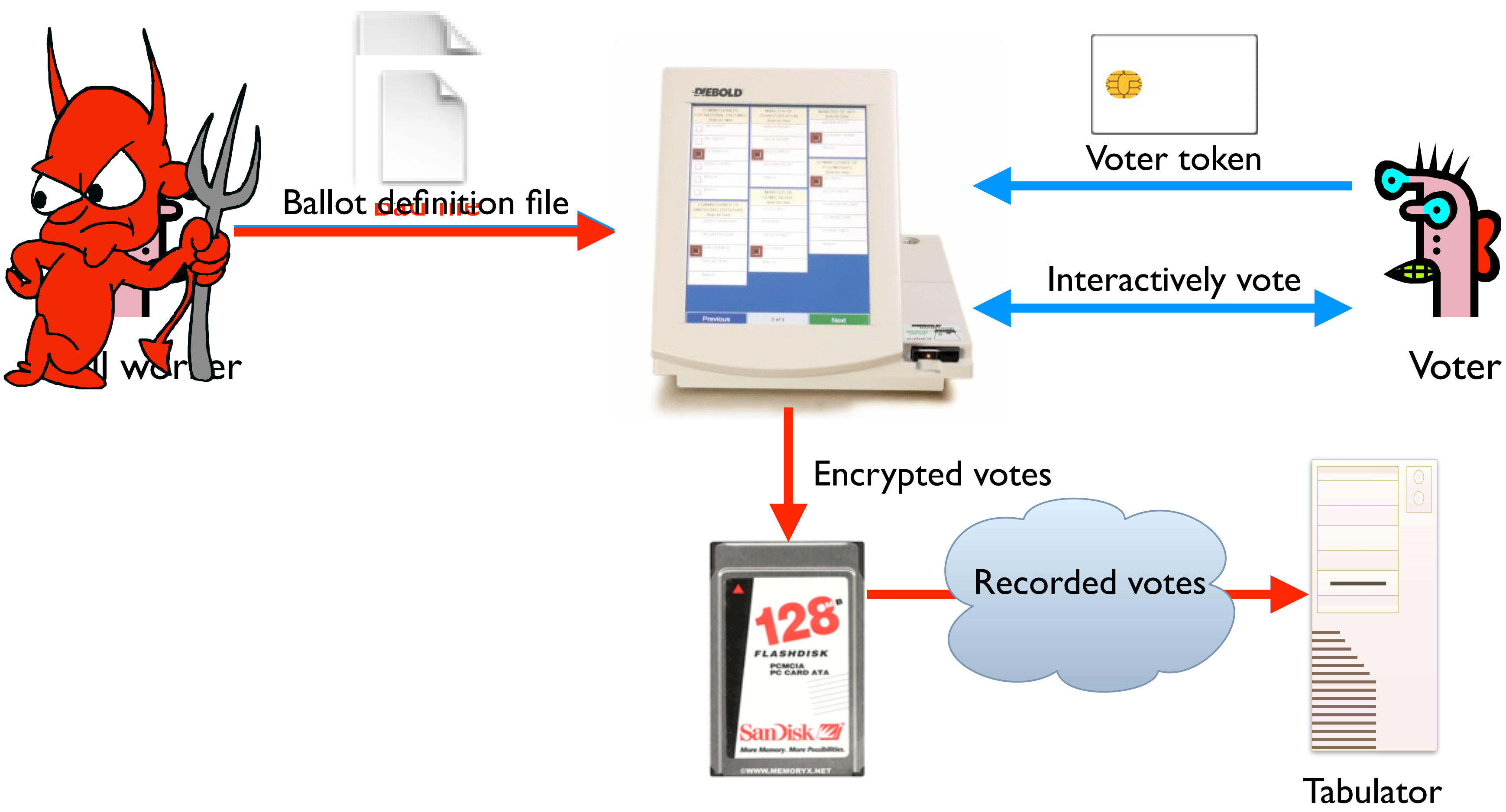
- Voters
- Election officials
- Employees of voting machine manufacturer
  - Software/hardware engineers
  - Maintenance people
- Other engineers
  - Makers of hardware
  - Makers of underlying software or add-on components
  - Makers of compiler
- …
- Or any combination of the above

What Software is Running?

Problem: An adversary (e.g., a poll worker, software developer, or company representative) able to control the software or the underlying hardware could do whatever he or she wanted.

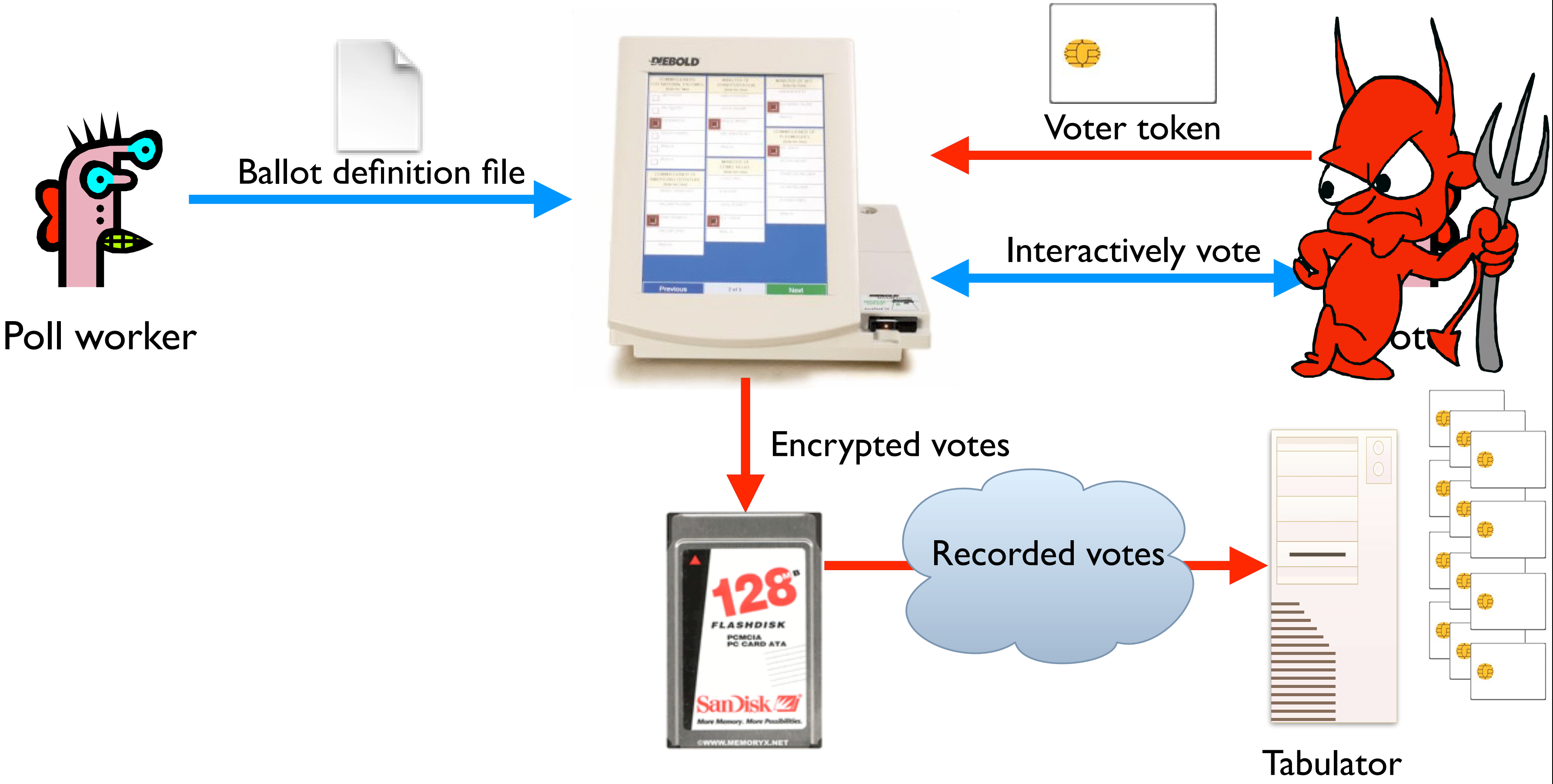
Problem: Ballot definition files are not authenticated.

Example attack: A malicious poll worker could modify ballot definition files so that votes cast for "Mickey Mouse" are recorded for "Donald Duck."



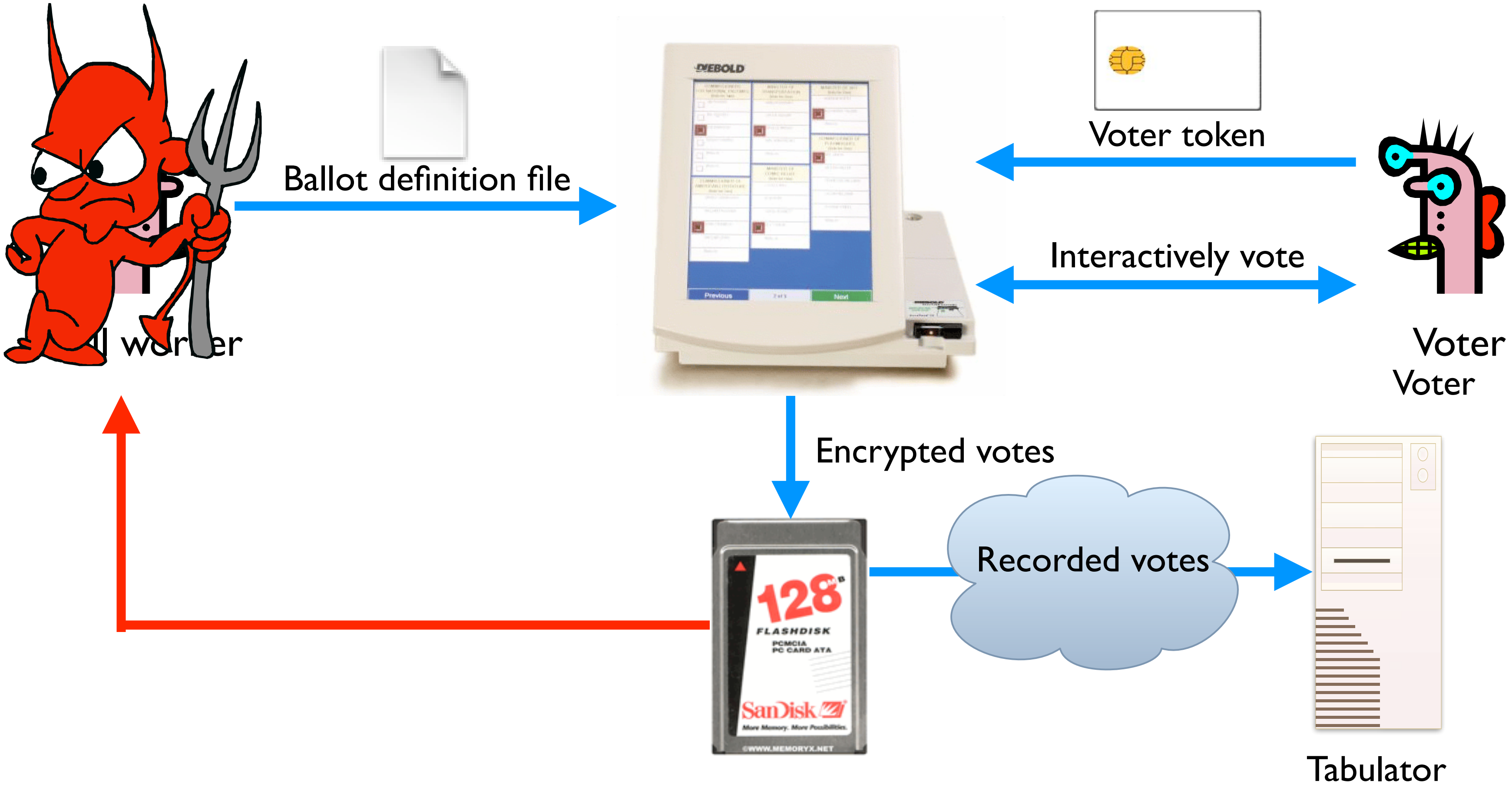
Problem: Smartcards can perform cryptographic operations. But there is no authentication from voter token to terminal.

Example attack: A regular voter could make his or her own voter token and vote multiple times.



Problem: Encryption key (“F2654hD4”) hard-coded into the software since (at least) 1998. Votes stored in the order cast.

Example attack: A poll worker could determine how voters vote.



Ballot definition file

Voter token

Interactively vote

ll wor er

Voter

Voter

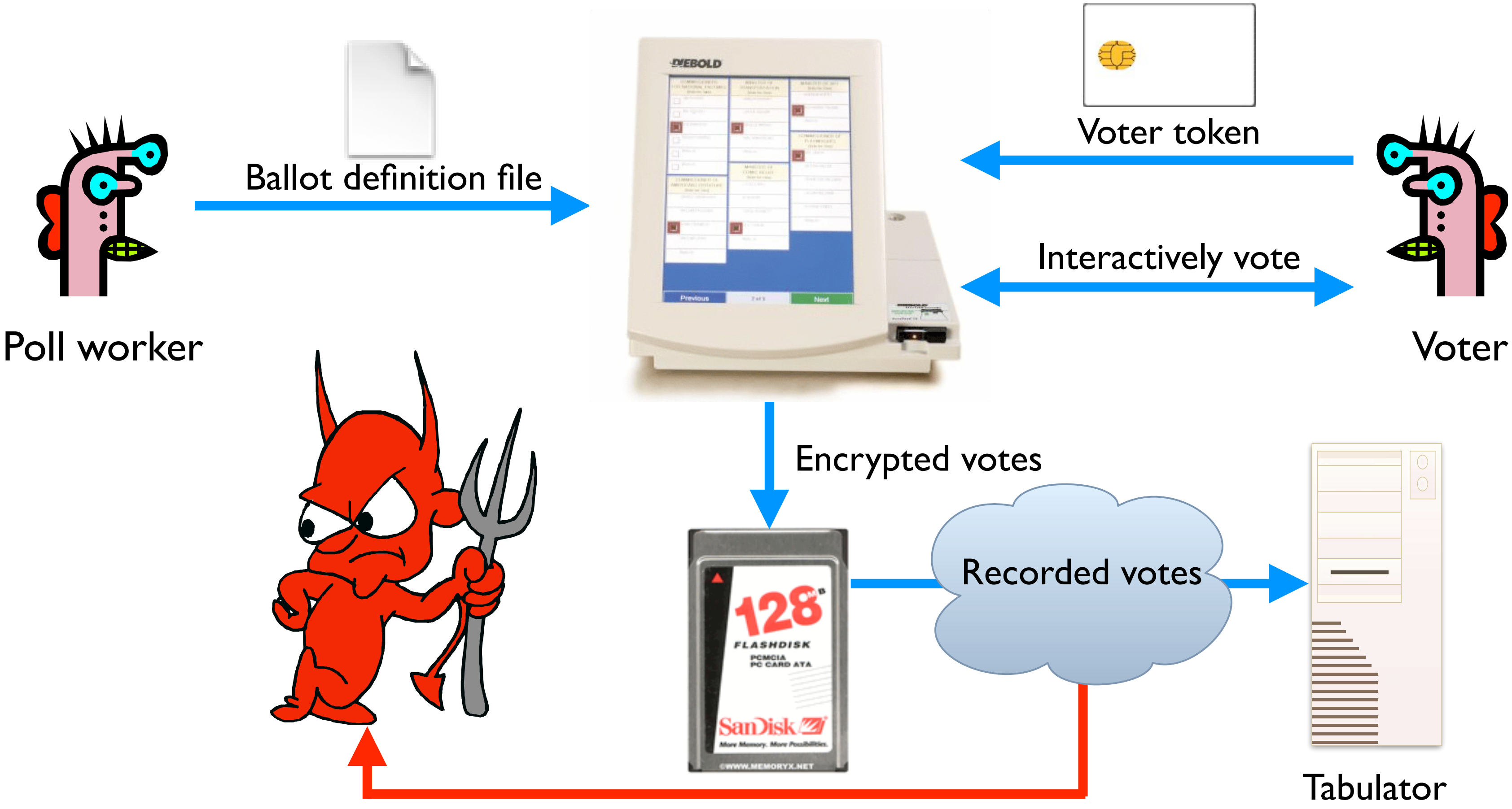
Encrypted votes

Recorded votes

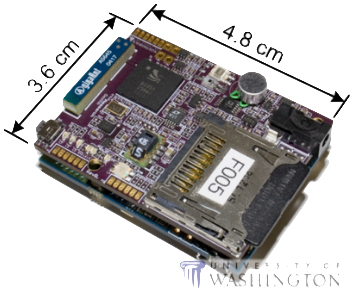
Tabulator

Problem: When votes transmitted to tabulator over the Internet or a dialup connection, they are decrypted first; the cleartext results are sent the the tabulator.

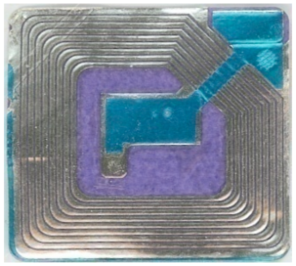
Example attack: A sophisticated outsider could determine how votes vote.



Security not just for PCs



mobile sensing platforms



RFID

EEG Gaming

large displays

ambient displays

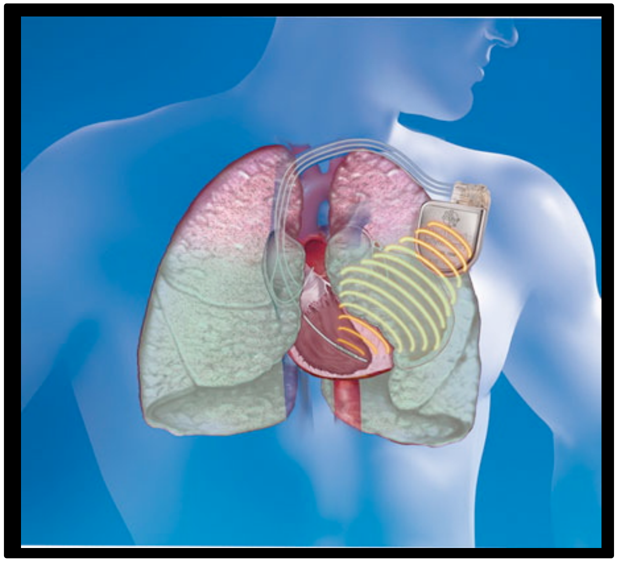
smart phones

wearables

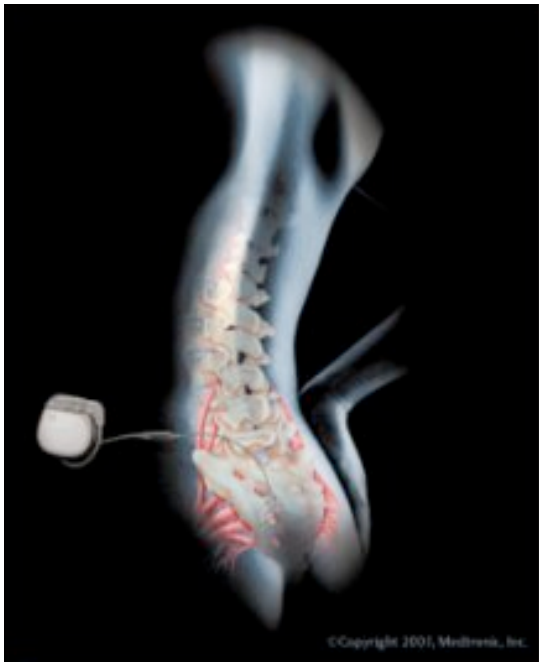
health displays

Implantable Medical Devices

Glucose
monitors



Pacemakers and
defibrillators



Neurostimulators

pumps

Whole-System is Critical

- Securing a system involves a whole-system view
  - Cryptography
  - Implementation
  - People
  - Physical security
  - Everything in between
- This is because “security is only as strong as the weakest link,” and security can fail in many places
  - No reason to attack the strongest part of a system if you can walk right around it.

Analyzing the Security of a System

- First thing: Summarize the system as clearly and concisely as possible
  - Critical step. If you can't summarize the system clearly and concisely, how can you analyze it's security?
- Next steps:
  - Identify the assets: What do you wish to protect?
  - Identify the adversaries and threats
  - Identify vulnerabilities: Weaknesses in the system
  - Calculate the risks

Assets

- Need to know what you are protecting!
  - Hardware: Laptops, servers, routers, PDAs, phones, ...
  - Software: Applications, operating systems, database systems, source code, object code, ...
  - Data and information: Data for running and planning your business, design documents, data about your customers, data about your identity
  - Reputation, brand name
  - Responsiveness
- Assets should have an associated value (e.g., cost to replace hardware, cost to reputation, how important to business operation)

Adversaries

- National governments
- Terrorists
- Thieves
- Business competitors
- Your supplier
- Your consumer
- New York Times
- Your family members (parents, children)
- Your friends
- Your ex-friends
- ...

Threats

- Threats are actions by adversaries who try to exploit vulnerabilities to damage assets
  - Spoofing identities: Attacker pretends to be someone else
  - Tampering with data: Change outcome of election
  - Denial of service: Attacker makes voting machines unavailable on election day
  - Elevation of privilege: Regular voter becomes admin
- Specific threats depend on environmental conditions, enforcement mechanisms, etc
  - You must have a clear, simple, accurate understanding of how the system works!

Threats

- Several ways to classify threats
  - By damage done to the assets
  - By the source of attacks
    - (Type of) insider
    - (Type of) outsider
    - Local attacker
    - Remote attacker
    - Attacker resources
- I like to think of a matrix
  - Adversaries on one axis
  - Assets on the other axis

Vulnerabilities

- Weaknesses of a system that could be exploited to cause damage
  - Accounts with system privileges where the default password has not been changed (Diebold: 1111)
  - Programs with unnecessary privileges
  - Programs with known flaws
  - Known problems with cryptography
  - Weak firewall configurations that allow access to vulnerable services
  - …
- Sources for vulnerability updates: CERT, SANS, Bugtraq, the news(?)

Risks

- Quantitative risk management
  - Example: Risk = Asset × Threat × Vulnerability
  - Monetary value to assets
  - Threats and vulnerabilities are probabilities
  - (Yes: Difficult to assign these costs and probabilities)
- Qualitative risk management
  - Assets: Critical, very important, important, not important
  - Vulnerabilities: Has to be fixed soon, should be fixed, fix if convenient
  - Threats: Very likely, likely, unlikely, very unlikely

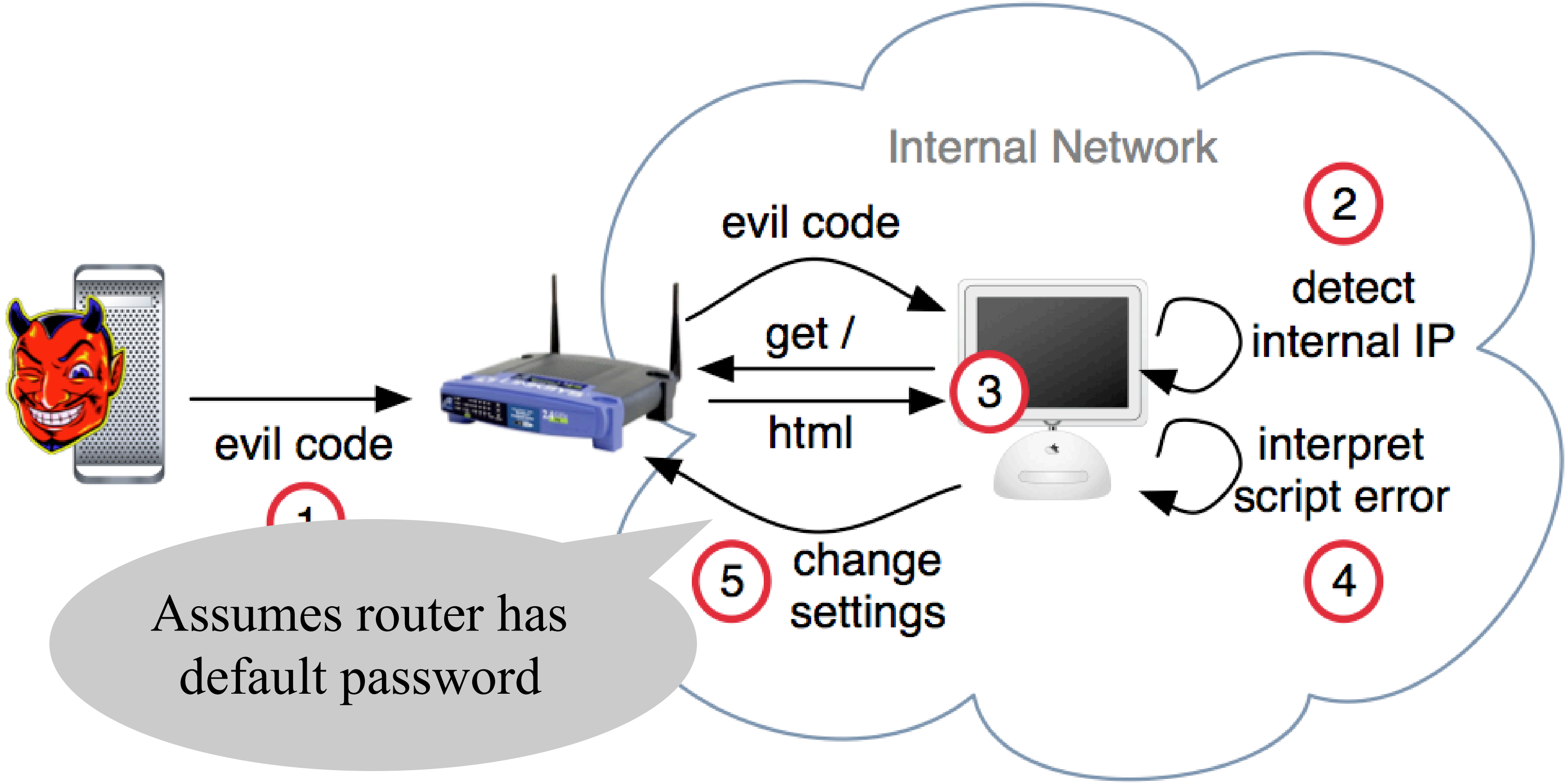
Security is Subtle

- Security attacks can be subtle
- So need to think careful!
  - And keep the whole system in mind
  - Whole system includes the users!
- Phishing one example
  - If attacker can trick user into entering private information, then no protection mechanism will help
  - (So research tries to focus on helping users not be tricked)

Another Example: Drive-By Pharming

- Designed to provide a firewall to external machines (keep the bad guys out)

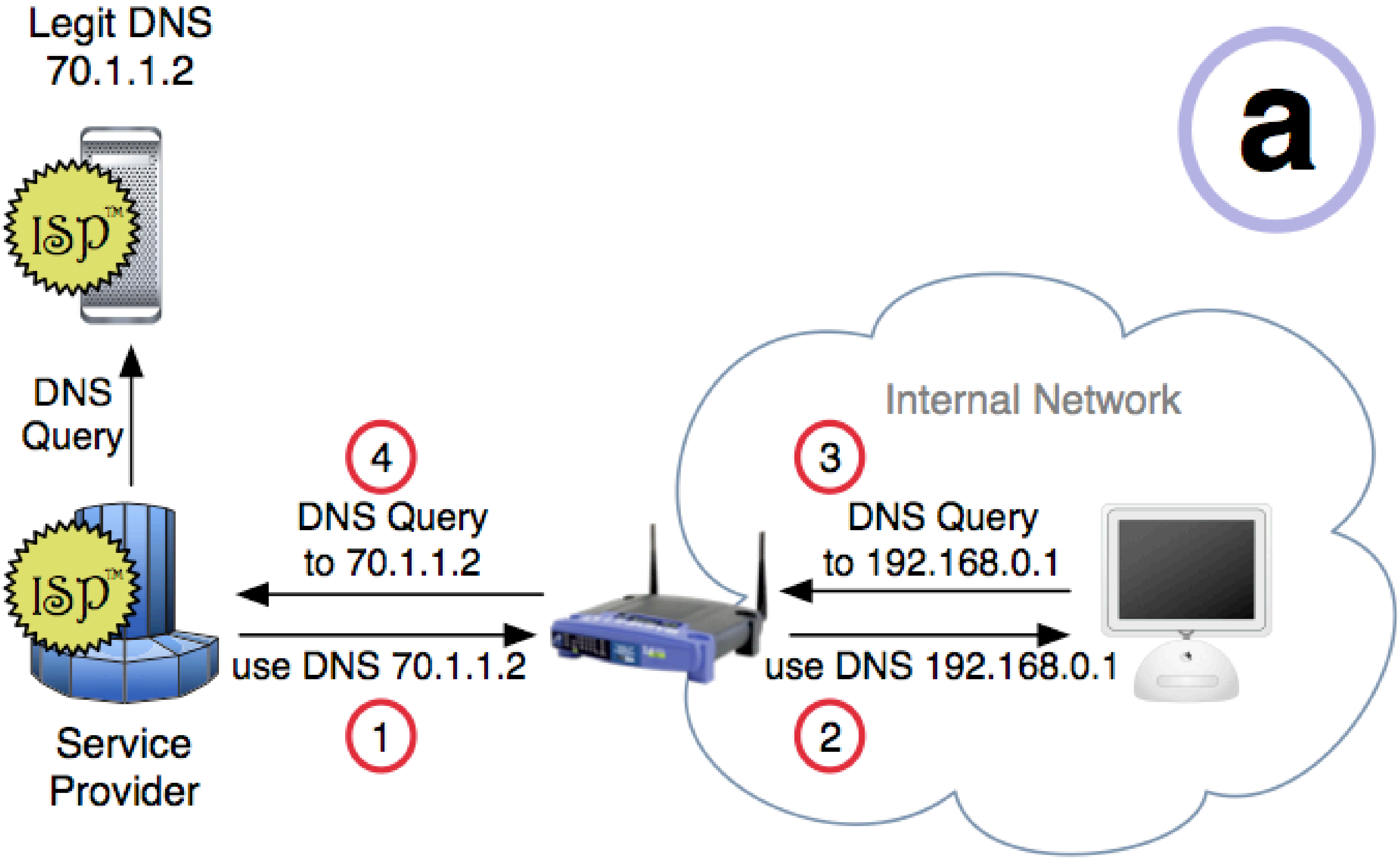
Another Example: Drive-By Pharming



Assumes router has default password

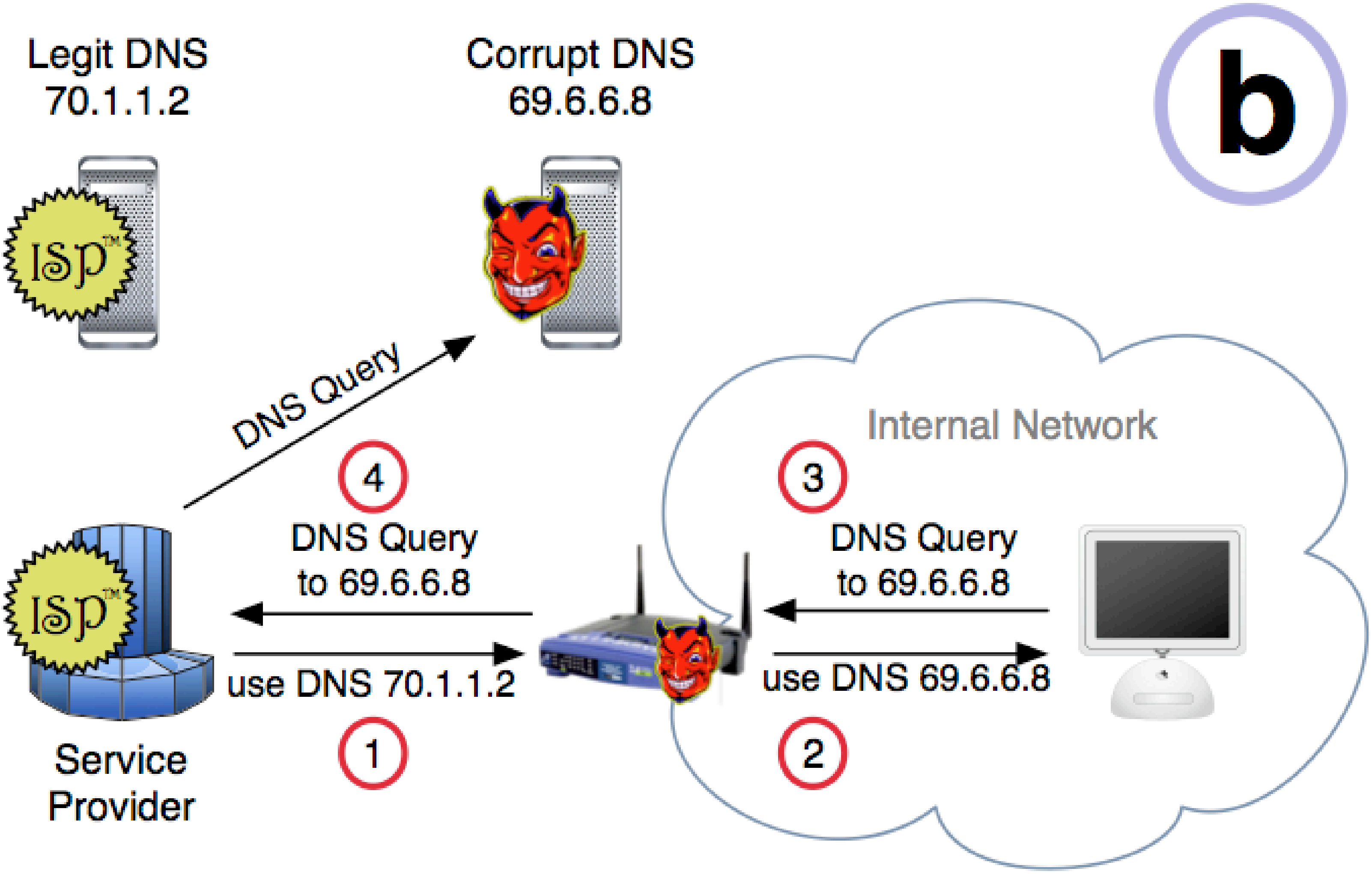
Reference: http://www.cs.indiana.edu/pub/techreports/TR641.pdf

Another Example: Drive-By Pharming



Reference: http://www.cs.indiana.edu/pub/techreports/TR641.pdf

Another Example: Drive-By Pharming



Reference: http://www.cs.indiana.edu/pub/techreports/TR641.pdf

Many Desirable Security Properties

- Authenticity
- Confidentiality
- Integrity
- Availability
- Accountability and non-repudiation
- Freshness
- Access control
- Privacy of collected information
- …

Syllabus

- Thinking about security; the “big picture”
  - The hardest part: Getting the “security mindset”
- Software security, buffer overflow attacks
- Cryptography
  - Block ciphers, stream ciphers, hash functions, MACs, public key encryption, digital signatures, PKI, key exchange, protocols
- Authentication, passwords, biometrics
- Trusted computing, secure hardware, tamper resistance

Syllabus

- Wireless security, including RFIDs, 802.11, and the future
- Web security and privacy, cross-site scripting, cookies, spyware
- Anonymous communications: Tor, attacks and defenses
- Information leakage and covert channels
- TCP/IP security, routing security, DNS security
- Firewall and intrusion detection systems
- Botnets and worms

Correctness versus Security

- System correctness: system satisfies specification
  - For reasonable input, get reasonable output
- System security: system properties preserved in face of attack
  - For unreasonable input, output not completely disastrous
- Main difference: active interference from adversary
- Modular design may increase vulnerability
  - Abstraction is difficult to achieve in security: what if the adversary operates below your level of abstraction?
- Modular design may increase security: small TCB
- Complexity may increase vulnerability

Bad News

- Security often not a primary consideration
  - Performance and usability take precedence
- Feature-rich systems may be poorly understood
  - Higher-level protocols make mistaken assumptions
- Implementations are buggy
  - Buffer overflows are the "vulnerability of the decade"
- Networks are more open and accessible than ever
  - Increased exposure, easier to cover tracks
- No matter what technical mechanisms you have, people may circumvent them
  - Phishing, impersonation, write down passwords, ...
- Attackers may be very powerful
  - ISPs, governments, ...

Better News

- There are a lot of defense mechanisms
  - We'll study some, but by no means all, in this course
- It's important to understand their limitations
  - "If you think cryptography will solve your problem, then you don't understand cryptography… and you don't understand your problem" -- Bruce Schneier
  - Security is not a binary property
  - Many security holes are based on misunderstanding
- Security awareness and user "buy-in" help
-

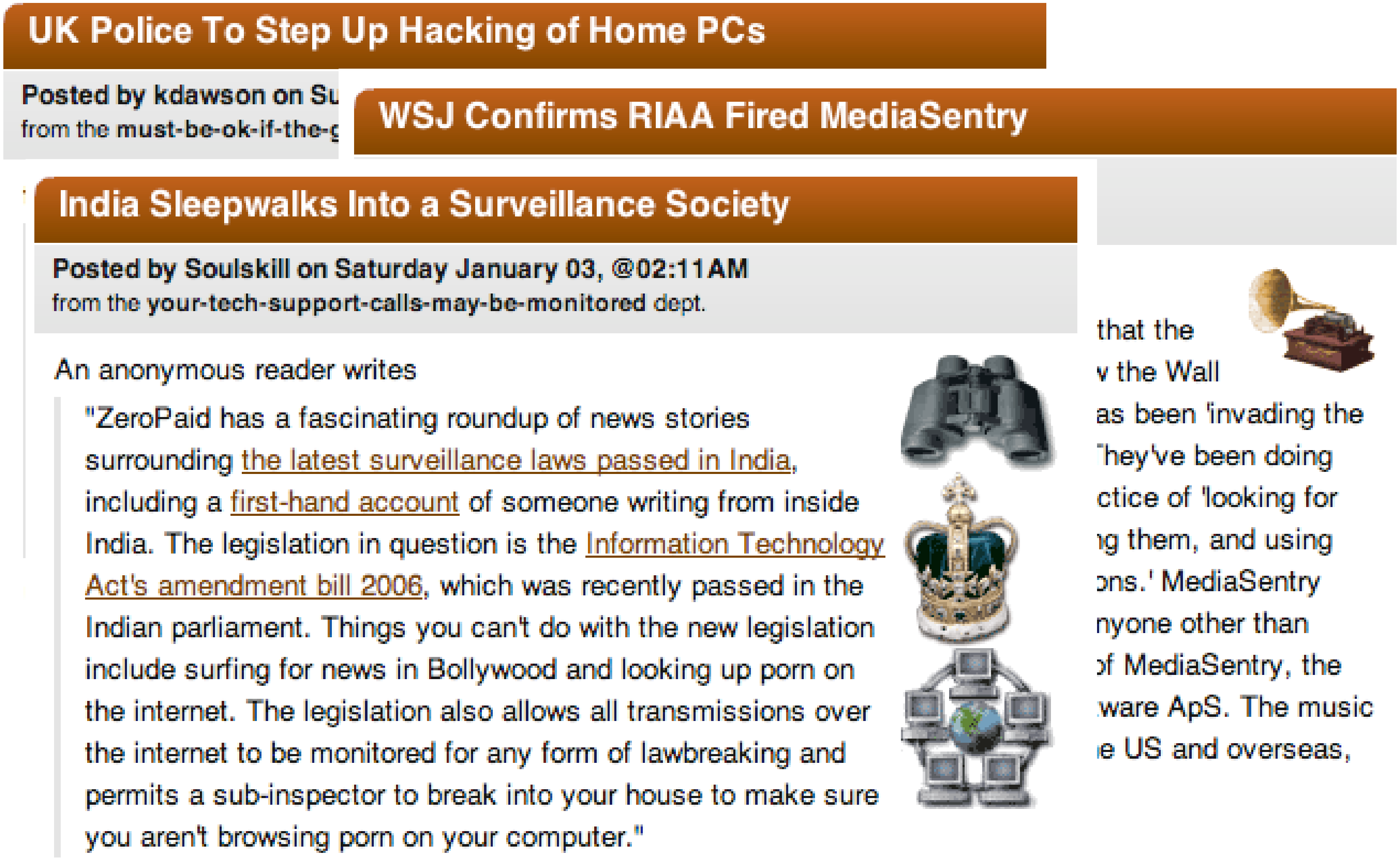
Blog

- Help you develop the “security mindset”
- Best way to learn a foreign language: move to that country and immerse yourself in the language.
- Same thing applies to “security thinking”
- Blog: opportunity to think about security on a regular basis -- outside of class
  - Current events
  - New product announcements
  - While doing regular, day-to-day activities?
    - Do you pass a bank, do you start thinking about how you might break in?

Current Events

- Important for computer security practitioners (and all computer scientists) to be able to
  - Reflect on the broader context of technology
  - Guides future development of technology
  - Guides future policy
- For the course blog
  - Summarize current event
  - Discuss why event arose
  - Reflect on what could have been done prior to the event arising (to prevent, deter, or change consequences)
  - Describe broader issues surrounding current event (ethical, societal)
  - How should people respond to the event (policy makers, the public, companes, etc.)

Current Events



UK Police To Step Up Hacking of Home PCs

Posted by kdawson on Su

from the must-be-ok-if-the-g

WSJ Confirms RIAA Fired MediaSentry

that the
v the Wall
as been 'invading the
hey've been doing
ctice of 'looking for
g them, and using
ons.' MediaSentry
nyone other than
of MediaSentry, the
ware ApS. The music
e US and overseas,

India Sleepwalks Into a Surveillance Society

Posted by Soulskill on Saturday January 03, @02:11AM
from the **your-tech-support-calls-may-be-monitored** dept.

An anonymous reader writes

> "ZeroPaid has a fascinating roundup of news stories surrounding the latest surveillance laws passed in India, including a first-hand account of someone writing from inside India. The legislation in question is the Information Technology Act's amendment bill 2006, which was recently passed in the Indian parliament. Things you can't do with the new legislation include surfing for news in Bollywood and looking up porn on the internet. The legislation also allows all transmissions over the internet to be monitored for any form of lawbreaking and permits a sub-inspector to break into your house to make sure you aren't browsing porn on your computer."

Security Reviews

- Summary of system
- Assets
- Adversaries and threats
- Potential weaknesses (OK to speculate, but make it clear that you are speculating)
- Potential defenses.
- Risks
- Conclusions.

Security reviews

- Let’s try this out

- Possible technologies:
  - IM
  - Facebook
  - ???