CSE P 564 (Autumn 2012)

Computer Security and Privacy

Tadayoshi Kohno

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

High-level information

- Instructor:
  - Tadayoshi Kohno (Yoshi)
- TA:
  - Alex Takakuwa
- Course website
  - Readings
- Course email list (not created yet, due to course name issues)
  - Urgent announcements
- Course forums
  - Announcements
  - Discussion
    - Recommendation: turn on email notifications or check announcements frequently

High-level information

- Class
  - Thursday, 6:30-9:20pm, Johnson 111
- Alex’s office hours
  - Thursdays, 5:30-6:20, CSE 216
- Yoshi’s office hours
  - Schedule by email

This Course

- It does not replace your company's internal security training (if your company has such training)
- It is designed to help raise your awareness and understanding of key computer security issues and concepts
  - The security mindset
  - How attackers think and what motivates them
  - Some contemporary issues, including Web security, cryptography, authentication, consumer device security, threat modeling, and so on

Prerequisites

- Computer security is a broad field
- Expected background: strong undergraduate computer science background
- Eagerness to learn!

- We will cover mathy things and more systemsy things

Course Logistics

- Lectures: Thursdays 6:30-9:20pm
  - We will have at least one short break
  - If you need to get up and move around, I encourage you to do so
- Readings (30% of grade)
- Assignments (30% of grade)
- Final (40% of grade)

Late Submission Policy

- Late assignments (not readings, not the final) will (generally) be dropped 20% per calendar day.
  - Late days will be rounded up
  - So an assignment turned in 26 hours late will be downgraded 40%
  - See website for exceptions -- some assignments must be turned in on time
- Assignments generally due on Monday nights

Course Materials

- Research papers
  - We will be reading from current and some classic research papers

- Textbook:
  - Daswani, Kern, Kesavan, “Foundations of Security”

- Additional materials linked to from course website

- Mix of lectures and paper discussions

Other Helpful Books (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.
  - (2nd edition available for purchase)
- Menezes, van Oorschot, and Vanstone, "Handbook of Applied Cryptography"
- Many many other useful books exist (not all online)

Others books, movies, …

- Pleasure books include:
  - Little Brother by Cory Doctorow
    - Available online here http://craphound.com/littlebrother/download/
  - Cryptonomicon by Neal Stephenson
- Movies include:
  - Hackers
  - Sneakers
  - Die Hard 4
  - WarGames
  - …
- Historical texts include:
  - The Codebreakers by David Kahn
  - The Code Book by Simon Singh

Mailing List

- Make sure you're on the mailing list
  - We'll send a test mail next week; everyone enrolled should receive it
- URL for mailing list (also on course website):
  - https://mailman1.u.washington.edu/mailman/listinfo/csep590a_au12
- Used for urgent announcements

- Some potential problems due to the fact that we were only assigned a course number today

Discussion Board

- We've set up a forum for this course to discuss assignments
  - https://catalyst.uw.edu/gopost/board/kohno/29821/
- Please use it to discuss the course, and also to post comments on research papers

Assignments

- General plan (tentative):
  - Approximately 3 assignments (timeline TBD, most likely due on Mondays)
  - Submit to Catalyst system (URL to be posted on course page)
- Expected topics to include:
  - Threat modeling
  - Cryptography
  - Web security
  - Possibly a second threat modeling assignment or another assignment toward the end of the course
- Also possibly in-class exercises, due at end of class

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, economics, ...
- **Technical and research aspects** of security
  - Attack techniques
  - Defenses
  - Current and classic research directions and results
- Computer security is a broad field
  - Impossible to cover everything
  - But possible to become conversant in key issues and contemporary topics

How to think about security

- Several approaches for developing “The Security Mindset” and for exploring the broader contextual issues surrounding computer security
    - First assignment
    - In class discussions (including focused discussion today)
    - Discussion in forum, critiquing papers, discussing current events, and so on

Current events and security reviews

- Past blog URL: http://cubist.cs.washington.edu/Security/
- Past Security Reviews: http://cubist.cs.washington.edu/Security/category/security-reviews/

What This Course is <u>Not</u> About

- <u>Not</u> a comprehensive course on computer security
  - Computer security is a <u>broad</u> discipline!
  - Impossible to cover everything in one quarter
  - So be careful in industry or wherever you go!
- <u>Not</u> about all of the latest and greatest attacks
  - Read bugtraq or other online sources instead
- <u>Not</u> a course on ethical, legal, or economic issues
  - We will touch on ethical issues, but the topic is huge
- <u>Not</u> 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

How Systems Fail

- Systems may fail for many reasons, including
- 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, revenge, curiosity, politics, terror....
- Fake websites, identity theft, steal money
- Control victim’s machine, send spam, capture passwords
- Industrial espionage and international politics
- Attack on website, extort money
- Wreak havoc, achieve fame and glory
- Access copy-protected movies and videos

Security is a Big Problem

- Security very often on the “front page” of the news

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

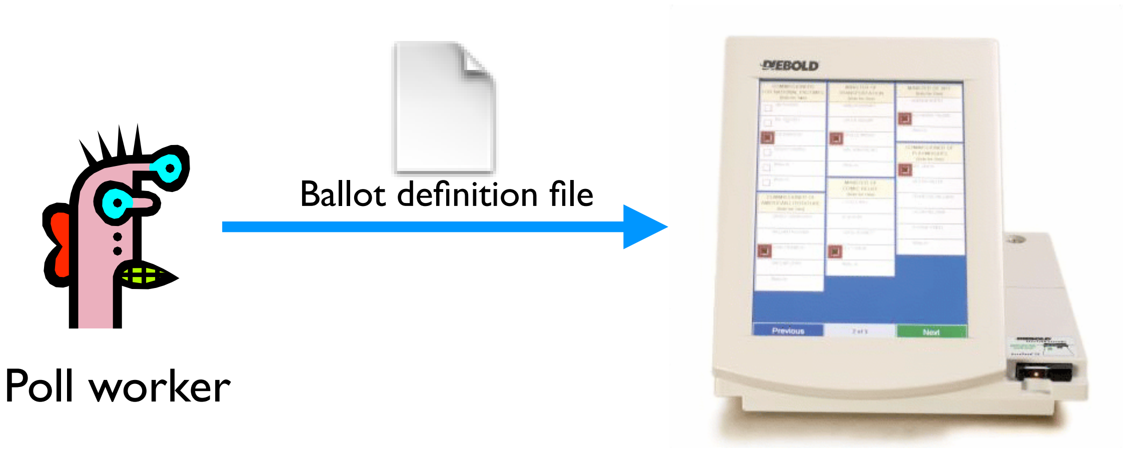
- The threat of a response may be enough to deter some attackers

Example: Electronic Voting

- Popular replacement to traditional paper ballots

AP

Pre-Election

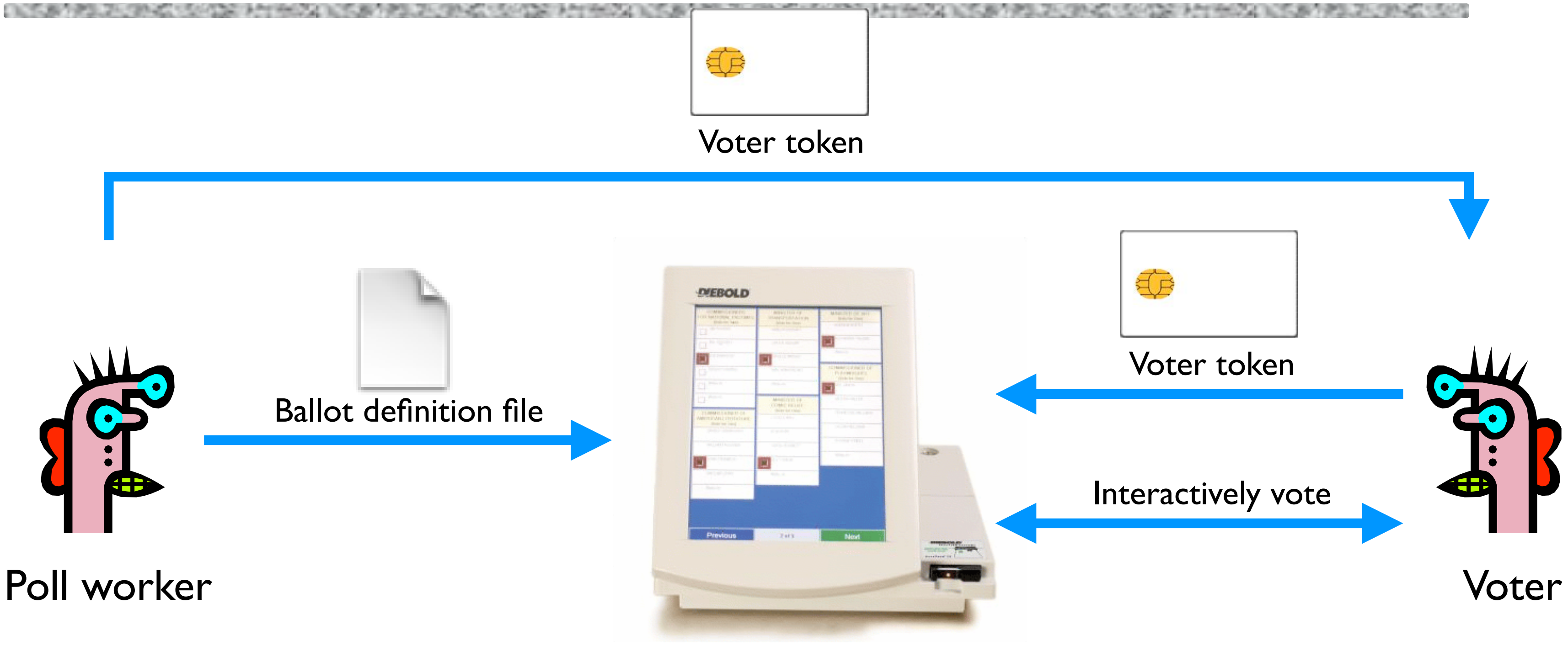


Ballot definition file

Poll worker

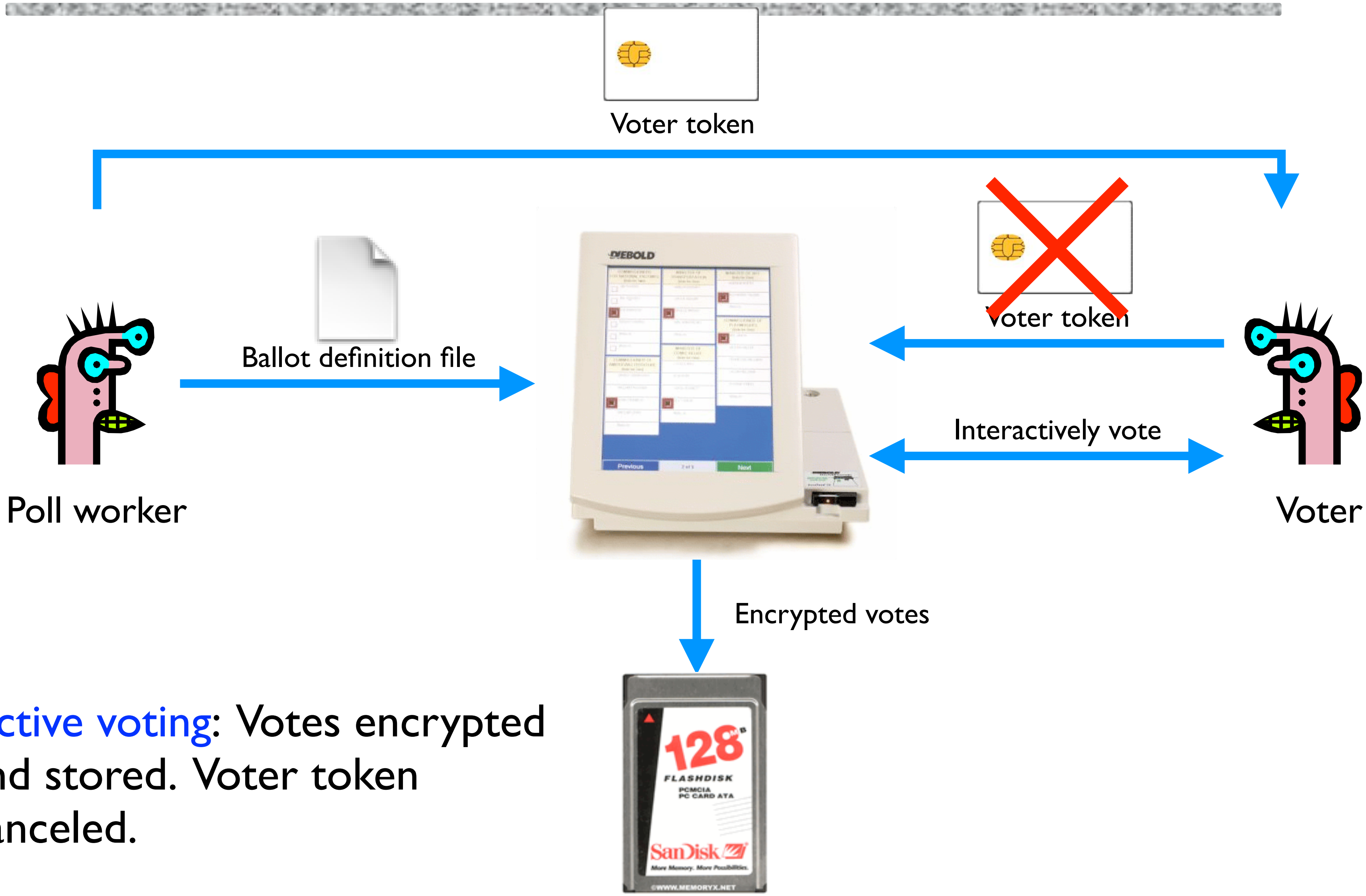
Pre-election: Poll workers load “ballot definition files” on voting machine.

Active Voting



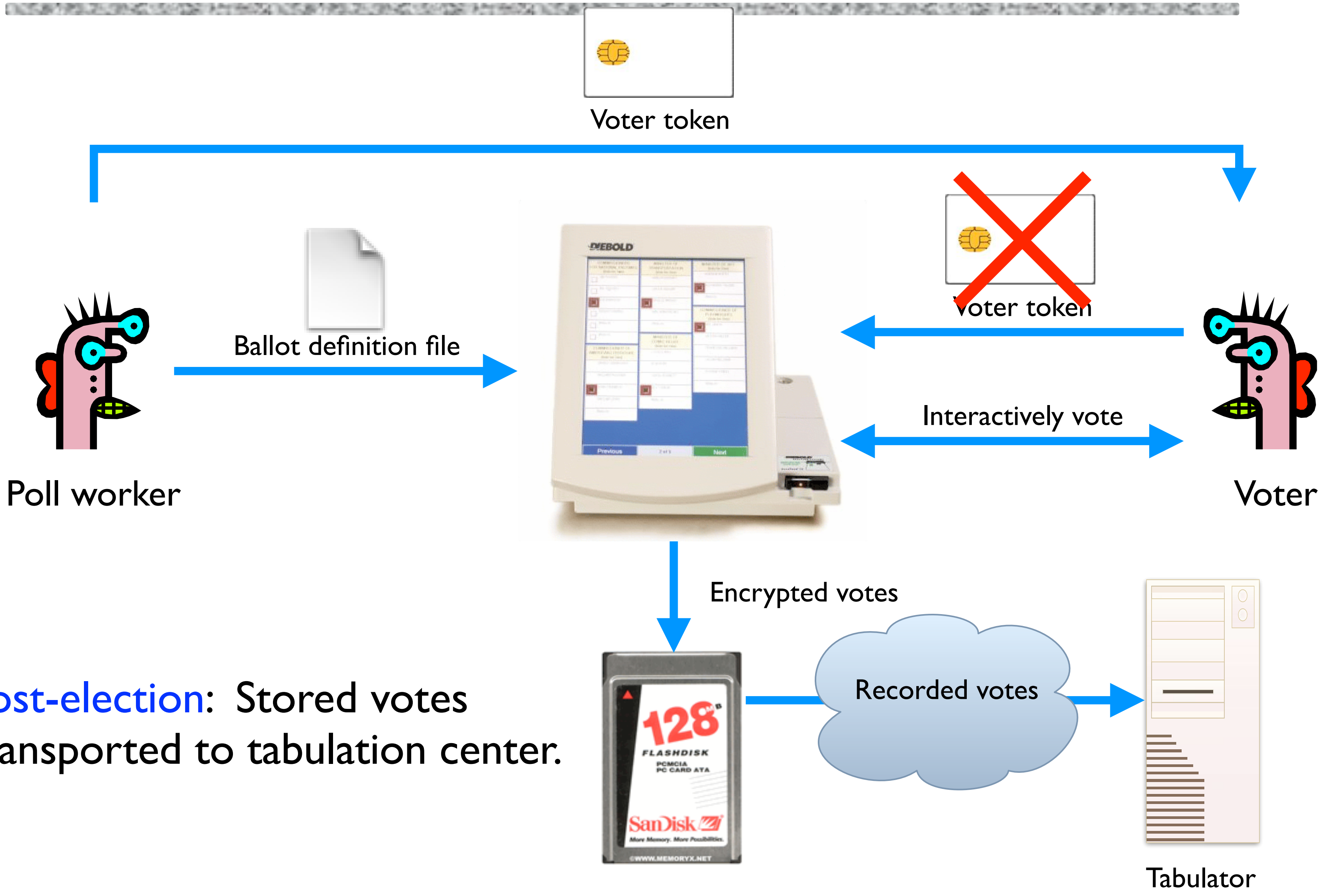
Active voting: Voters obtain single-use tokens from poll workers. Voters use tokens to active machines and vote.

Active Voting



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

Post-Election



Post-election: Stored votes transported to tabulation center.

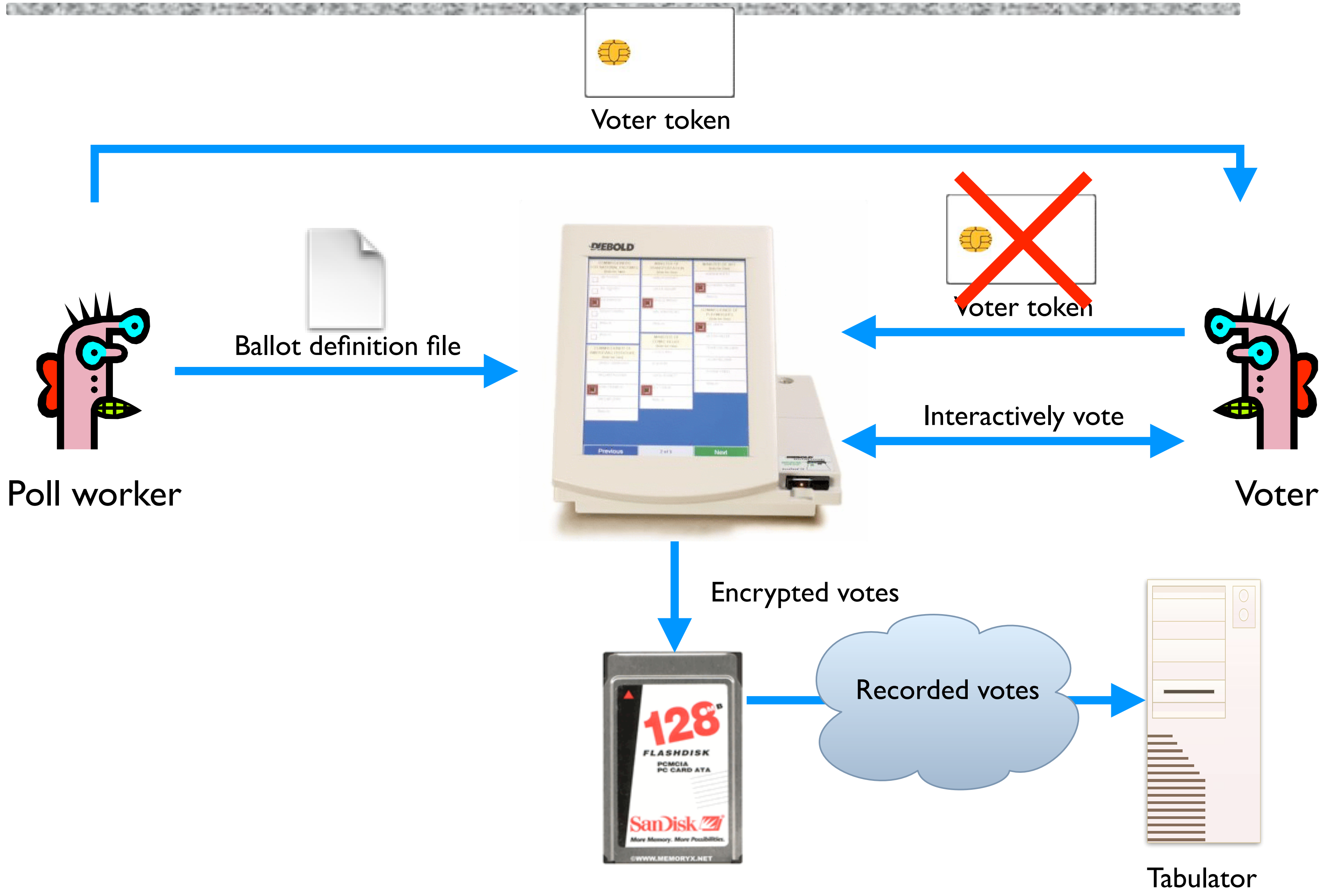
Security and E-Voting (Simplified)

- Functionality goals:
  - Easy to use
  - People should be able to cast votes easily, in their own language or with headphones for accessibility

Security 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?



Voter token

Poll worker

Ballot definition file

~~Voter token~~

Interactively vote

Voter

Encrypted votes

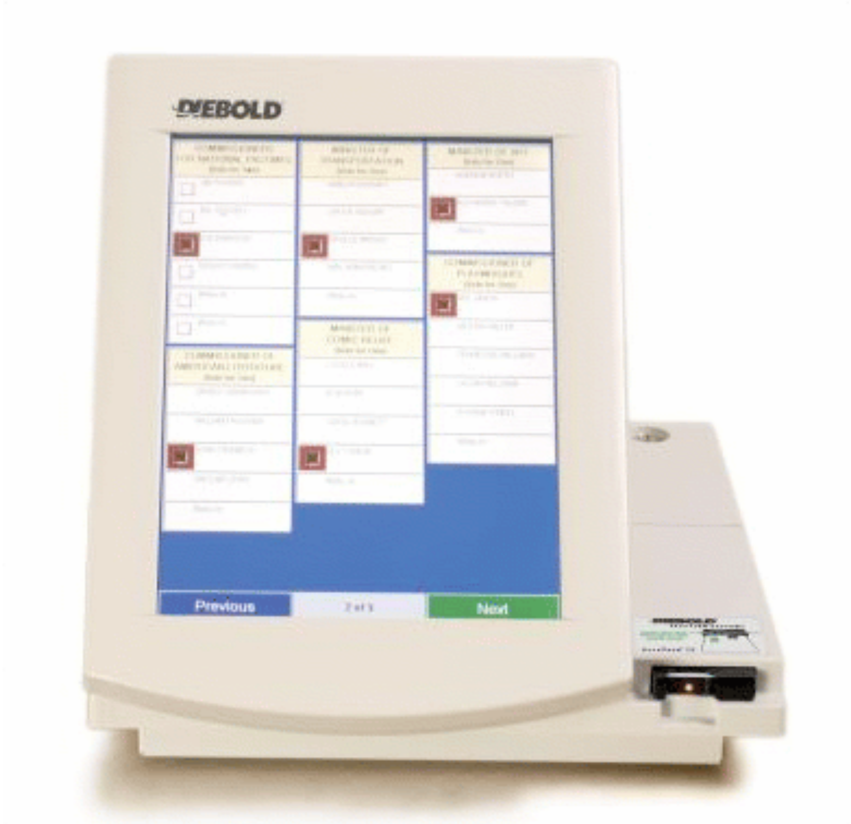
Recorded votes

Tabulator

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

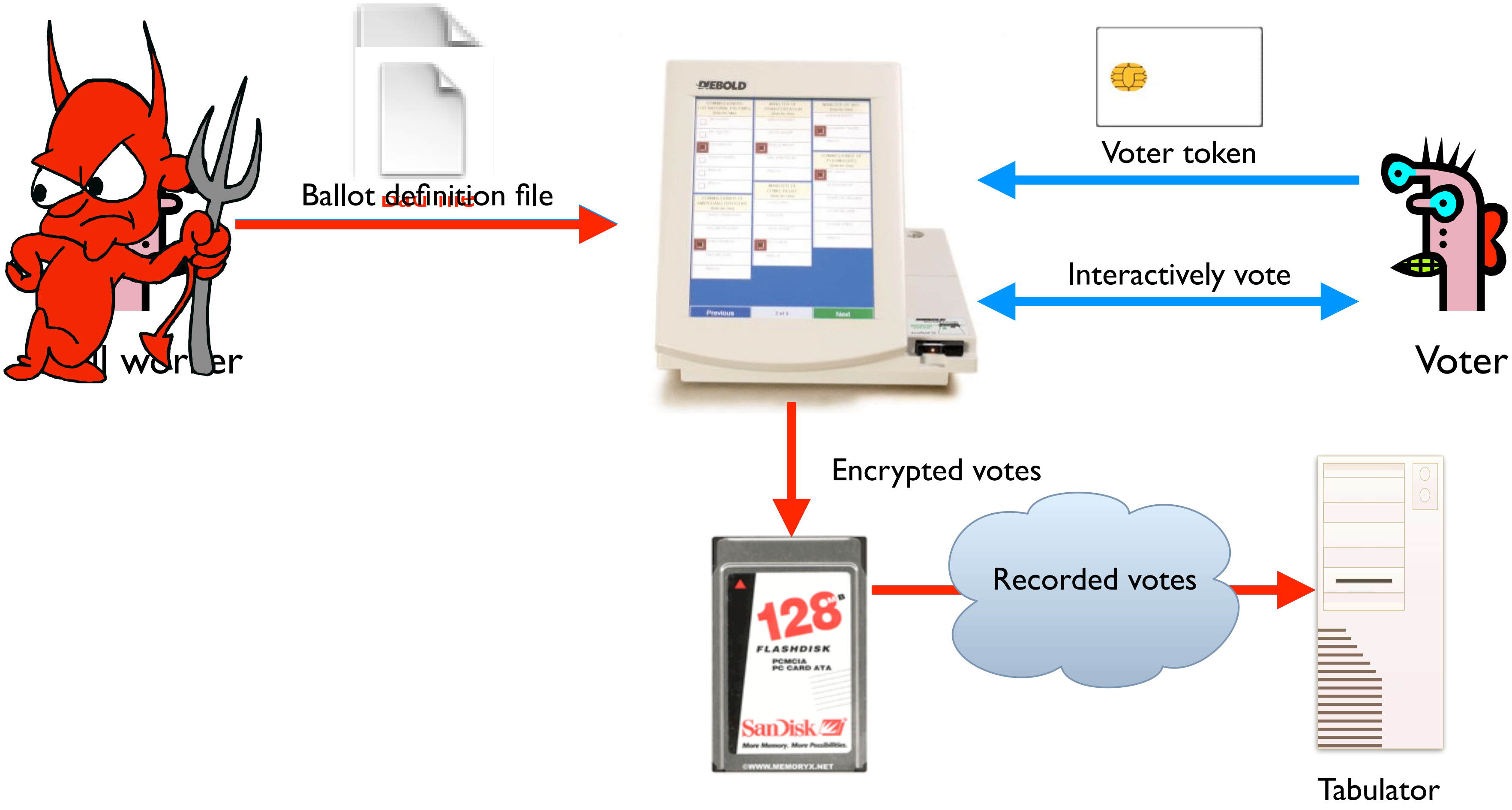
What Software is Running?



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.

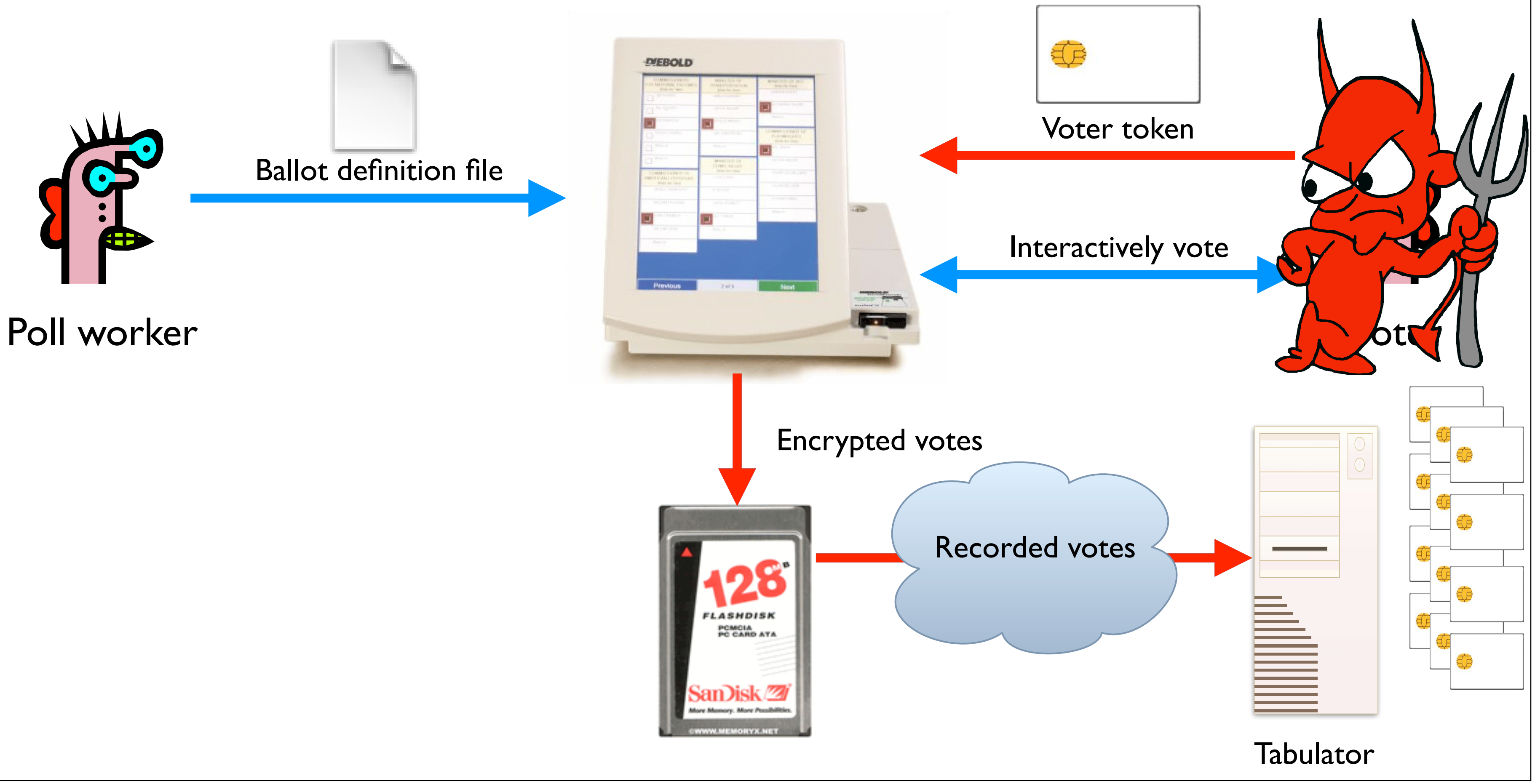
Problem: Ballot definition files are not authenticated.

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



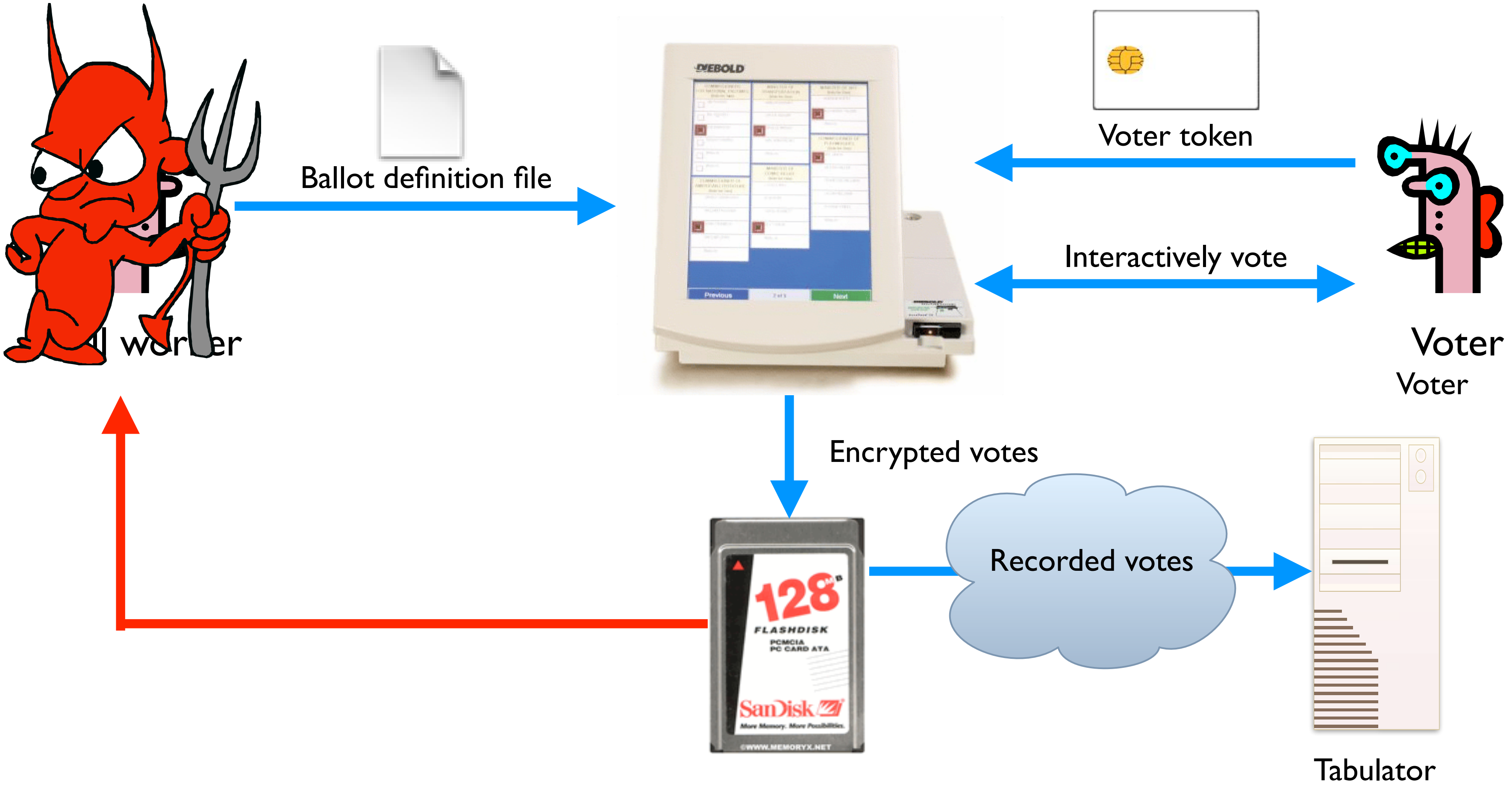
Problem: Smartcards can perform cryptographic operations. But there is no authentication from voter token to terminal.

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



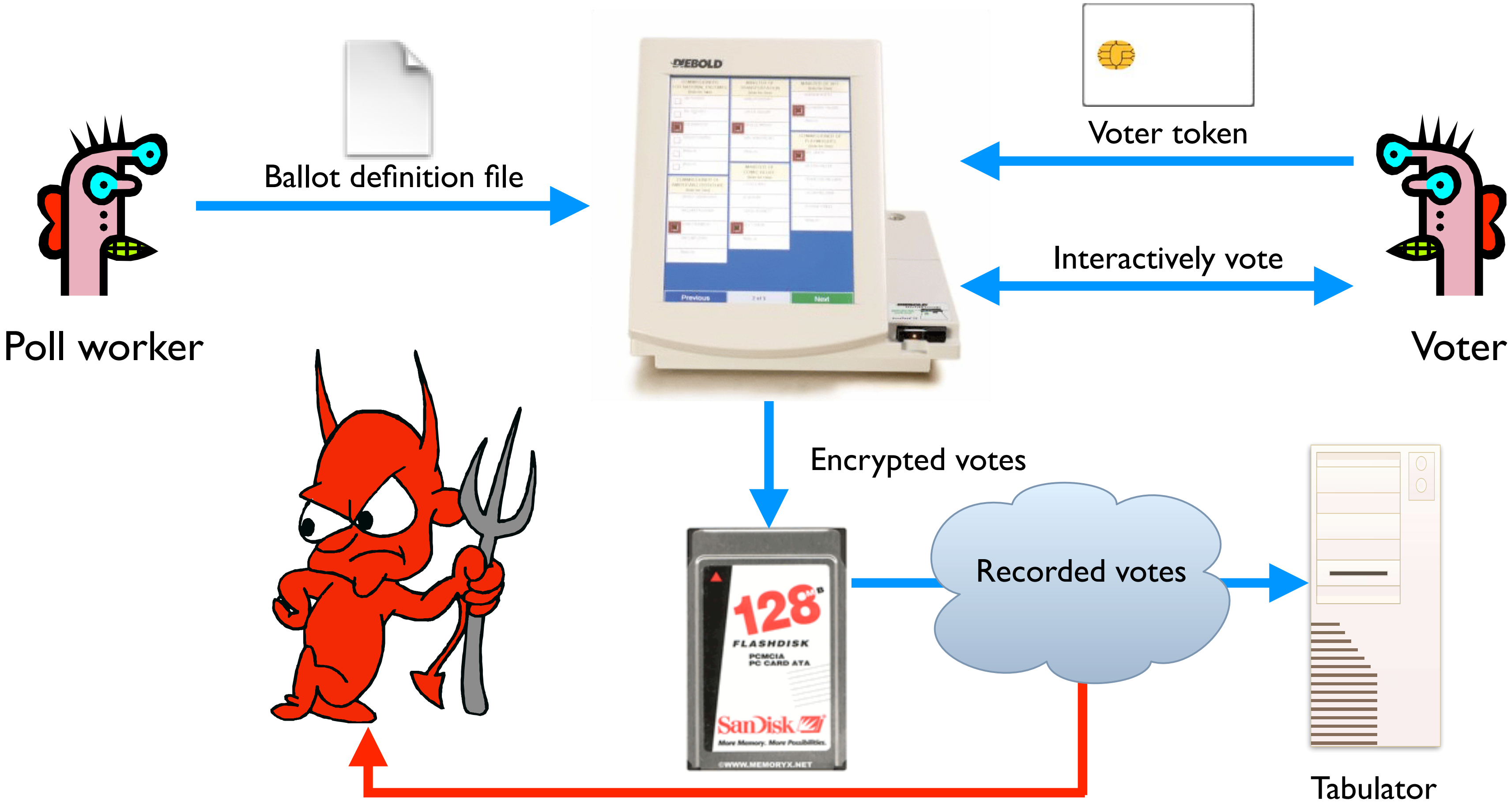
Problem: Encryption key (“F2654hD4”) hard-coded into the software since (at least) 1998. Votes stored in the order cast.

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

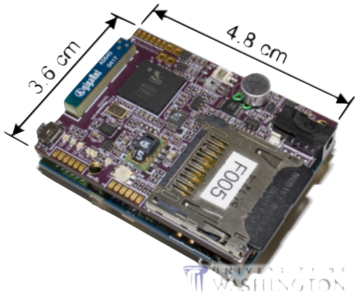


Problem: When votes transmitted to tabulator over the Internet or a dialup connection, they are decrypted first; the cleartext results are sent the the tabulator.

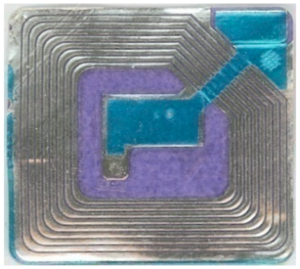
Example attack: A sophisticated outsider could determine how voters vote.



Security not just for PCs



mobile sensing platforms



RFID

EEG Gaming

large displays

ambient displays

smart phones

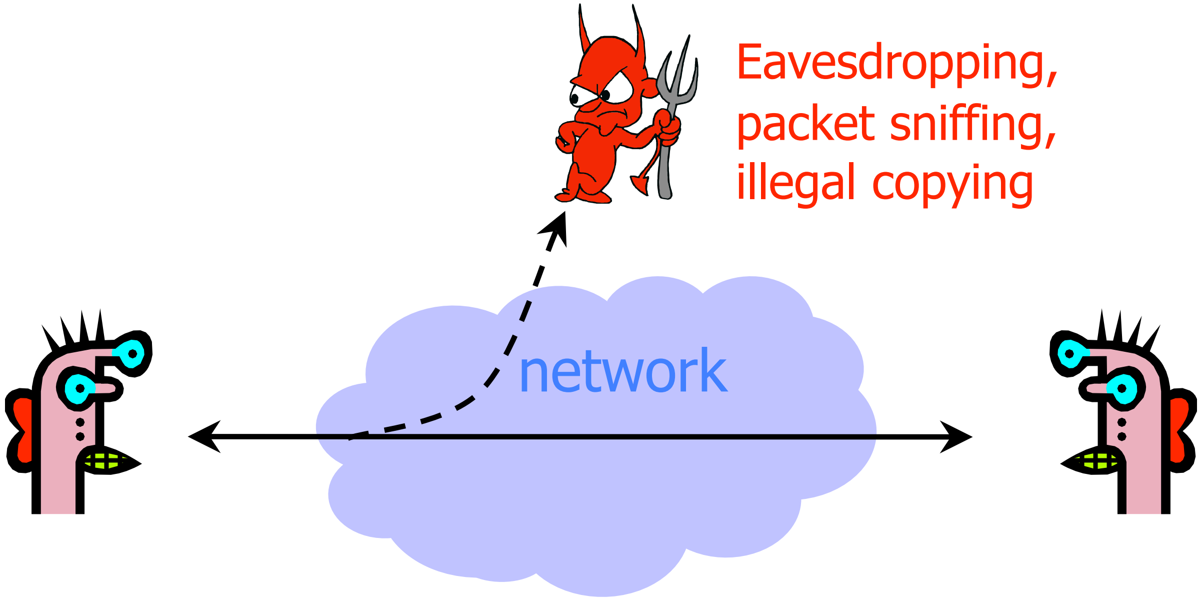
wearables

health displays

Security Goals

Confidentiality (Privacy)

- Confidentiality is concealment of information

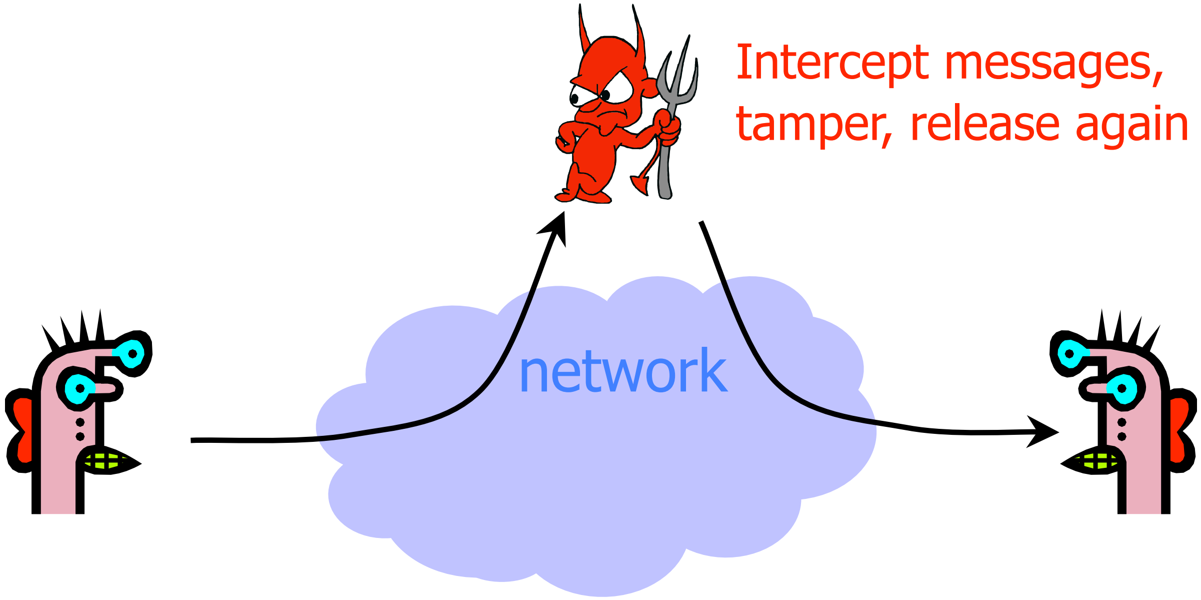


Eavesdropping,
packet sniffing,
illegal copying

network

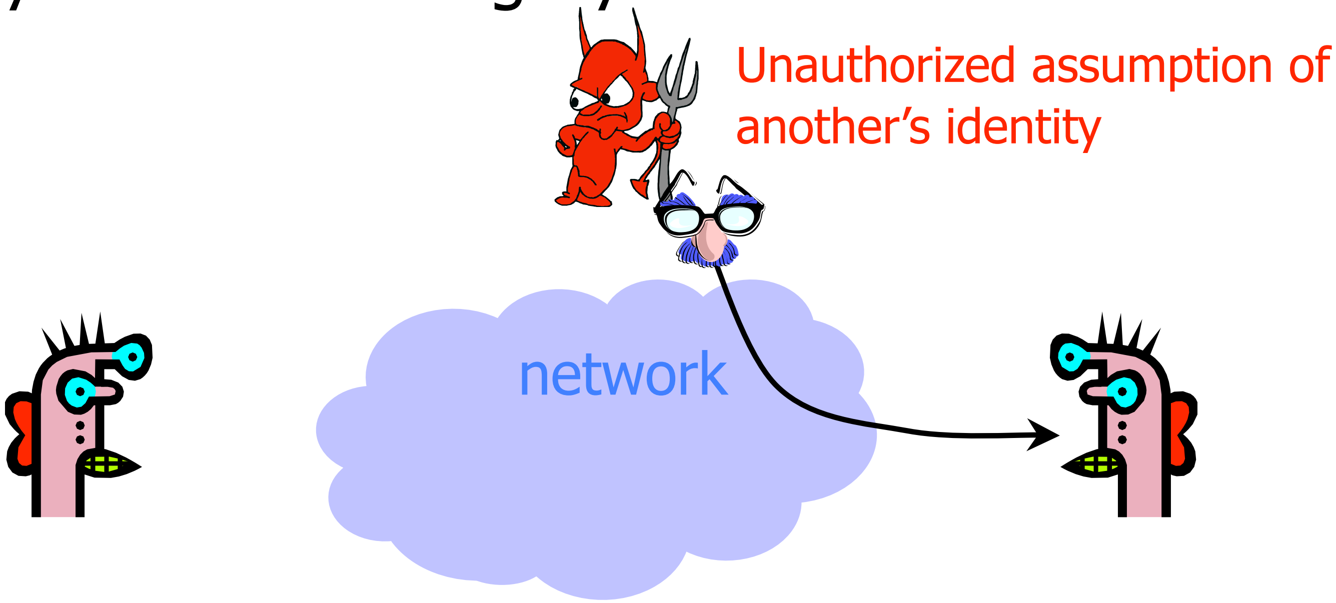
Integrity

- Integrity is prevention of unauthorized changes



Authenticity

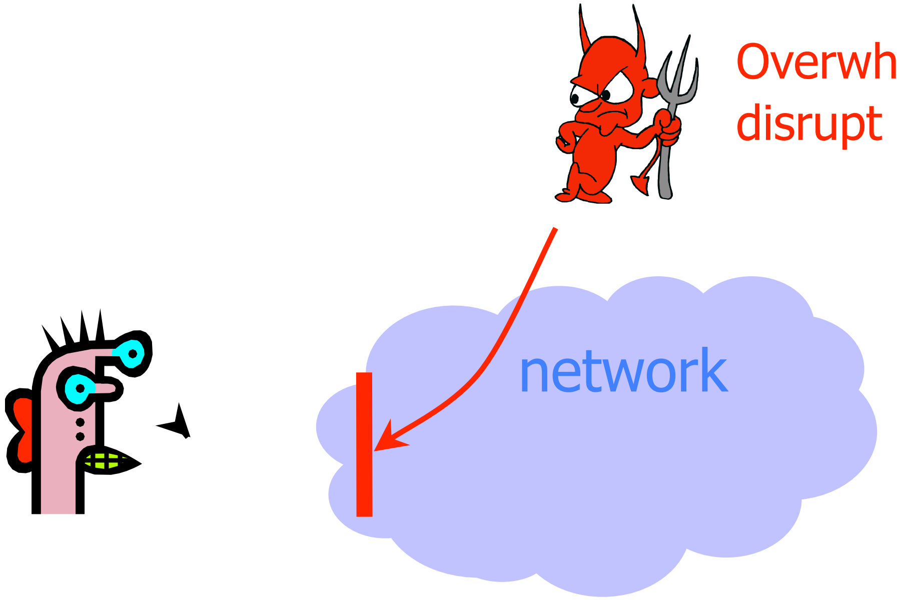
- Authenticity is identification and assurance of origin of information
- Highly related to integrity



Unauthorized assumption of another's identity

Availability

- Availability is ability to use information or resources desired



Overwhelm or crash servers, disrupt infrastructure

Security of a system

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.
  - (Still important to strengthen more than the weakest link)

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?
  - Summary can be hierarchical
- Next steps:
  - Identify the assets: What do you wish to protect?
  - Identify the adversaries
  - Identify the threats
  - Identify vulnerabilities: Weaknesses in the system
  - Calculate the risks

Assets

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

Adversaries

- National governments
- Organized crime
- Terrorists
- Thieves
- Business competitors
- Your supplier
- Your consumer
- The 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
  - Crash machines: 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
    - Confidentiality, Integrity, Availability
  - By the source of attacks
    - (Type of) insider
    - (Type of) outsider
    - Local attacker
    - Remote attacker
    - Attacker resources
  - By the actions
    - Interception
    - Interruption
    - Modification
    - Fabrication

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 implementation flaws
  - Problems with cryptography
  - Weak firewall configurations that allow access to vulnerable services
  - ...
- Sources for vulnerability updates: CERT, SANS, Bugtraq, the news, ...

Risks Analyses: Lots of Options

- Quantitative risk analysis
  - Example: $\overset{\text{Risk Exposure}}{\boxed{\text{Risk}}} = \overset{\text{Risk Impact}}{\boxed{\text{Asset}}} \times \overset{\text{Probability}}{\boxed{\text{Threat} \times \text{Vulnerability}}}$
  - Monetary value to assets
  - Threats and vulnerabilities are probabilities
  - (Yes: Difficult to assign these costs and probabilities)
- Qualitative risk analysis
  - Assets: Critical, very important, important, not important
  - Vulnerabilities: Very likely, likely, unlikely, very unlikely
  - Threats: Very likely, likely, unlikely, very unlikely

Helpful Tables

Asset	Confidentiality	Integrity	Availability
Hardware			
Software			
Data			
Personal Safety			
…			

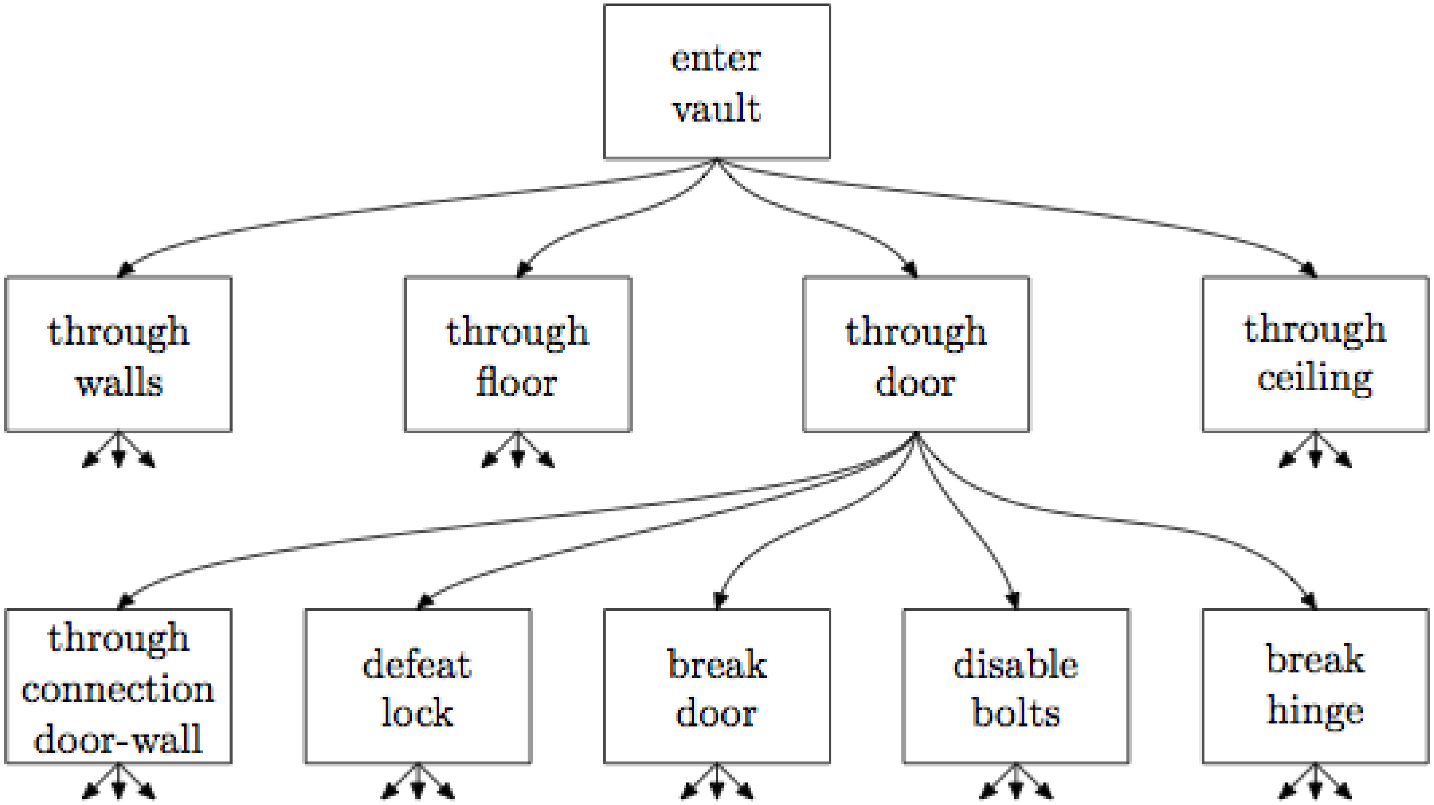
Helpful Tables

	Voter	Election official	…
Privacy of vote			
Integrity of vote			
Availability of voting system			
Confidence in election			
…			

Helpful Tables

	Create New Voter Cards	Decrypt voting record	…
Privacy of vote			
Integrity of vote			
Availability of voting system			
Confidence in election			
…			

Attack Trees



Security is Subtle

- Security attacks can be subtle
- Can't provably and accurately identify / quantify all risks, vulnerabilities, threats.
- So need to think careful!
  - And keep the whole system in mind
- 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)

On Modularity and Complexity

- 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 (trusted computing base)
- Complexity may increase vulnerability

Not So Great News

- Security may not be a primary consideration
  - Performance and usability take precedence
- Feature-rich systems are hard to understand
  - Higher-level protocols make mistaken assumptions
- Implementations can be buggy
  - Buffer overflows, XSS vulnerabilities, ...
- Networks can be left open and accessible
  - Increased exposure, easier to cover tracks
- No matter what technical mechanisms a system has, 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
- 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

Update on Paper Reviews (9/27)

- We will use HotCRP (Conference Management Software)
- You will submit reviews of papers (I will generate review questions in advance).
- We will make all reviews world-readable before the respective class.
- This has the advantage of letting everyone learn from others’ perspectives. But also doesn’t lead to the “rush” to comment first.

First Assignment

First assignment

- 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”
- First assignment: opportunity to think about security outside of class
    - Current events
    - New product announcements
    - Security in your everyday life

Current Events

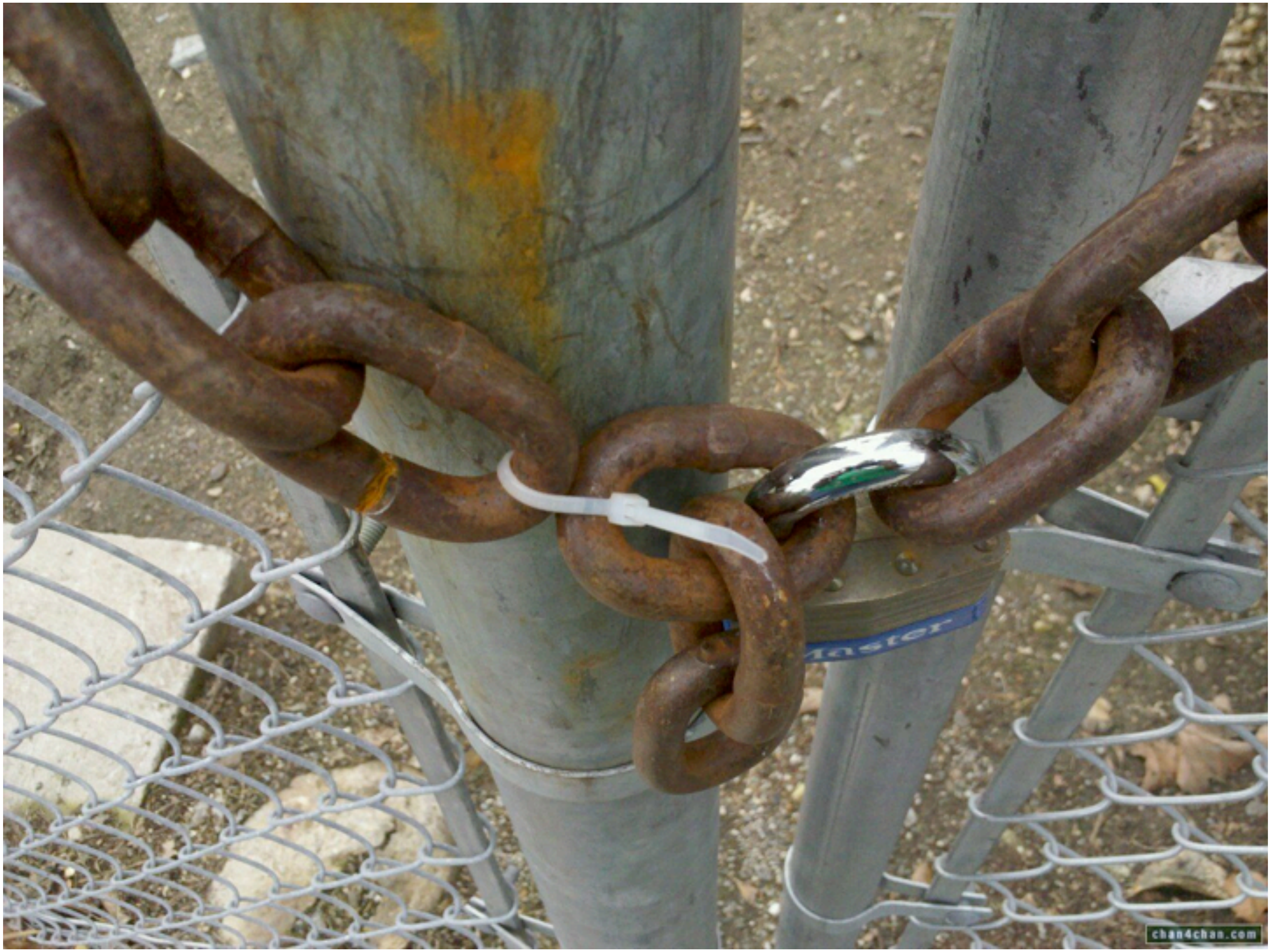
- Important for computer security practitioners (and all computer scientists) to be able to
  - Reflect on the broader context of technology
  - Guide future development of technology
  - Guide future policy
- For the first assignment
  - Summarize current event
  - Discuss why event arose
  - Reflect 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, companies, etc.)

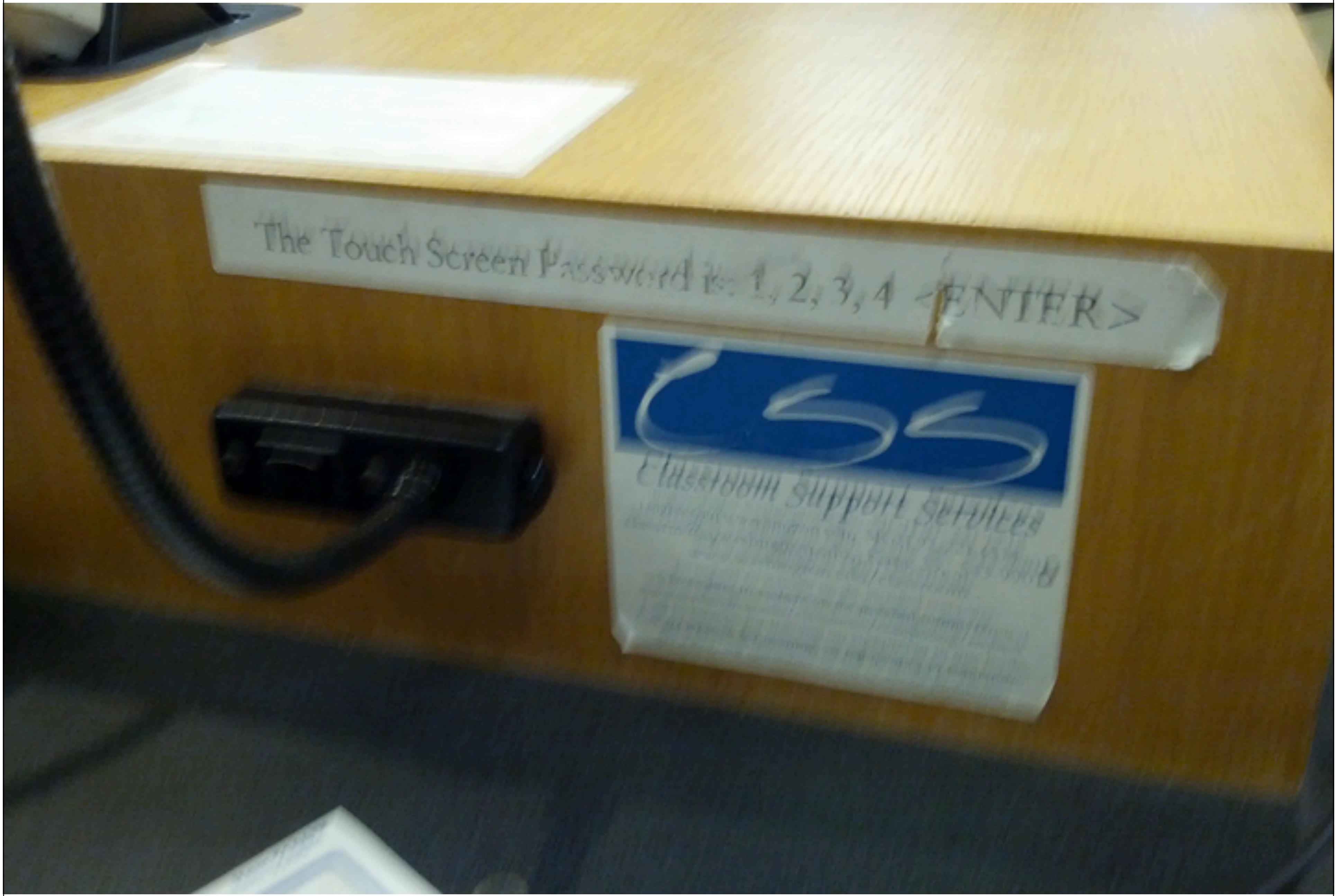
Security Review

- Summary of system/product (if don’t know details, make up something but note that you’re making it up)
- Assets
- Adversaries
- Threats
- Potential weaknesses (OK to speculate, but make it clear that you are speculating)
- Potential defenses
- Risks
- Conclusions

Optional: Security in your life

- Take and share security-related photos and stories and observations (anecdotes, videos, audio, etc.) on the discussion board
- Explain what you were capturing and how it relates to security
- *<b>Stay within legal limits</b>*---for instance, Washington State is a “2-Party State”, which means you can’t record communications without both sides’ consent/notification.
  (All-party for multi-way communications)


Master
chan4chan.com



The Touch Screen Password is: 1, 2, 3, 4 <ENTER>
CSS
Classroom Support Services

Practicum

Security is a contact sport

- Best to learn by doing
- Lots of learning to be done by having discussions with other people -- other people have unique insights and perspectives.

The task

- Break into groups of 3-5 people (ideally 4-5, for more discussion and perspectives)

- 20 minutes: Brainstorm topics for a security review
  - Choose a product that might have interesting security risks
  - Ideally choose something that you're not already familiar with from a security perspective (OK if you've thought about the technology before from a non-security perspective)
  - Try to discuss multiple possible topics, and then discuss why those topics might or might not be interesting from a security perspective

The task (continued)

- 15 minutes: Brainstorm topics for a security review
- 10-12 minutes: Summarize the technology that you decided to focus on from a security perspective
- 30-45 minutes: Conduct a security review
  - Assets (identify at least two)
  - Adversaries (identify at least two)
  - Threats (identify at least two)
  - Potential weaknesses (OK to speculate, but make it clear that you are speculating) (identify at least two)
  - Potential defenses (identify at least two)
  - Risks (are the adversaries, threats, weaknesses above serious or not?)
  - Conclusions (any conclusions from your observations above)
- Report back
- Submit via email your report via email (PDF form). Include names and UWNetIDs on each submission.