CSE 484 / CSE M 584 (Winter 2010)

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, 9:30 to 10:20am (right after class, may change)
  - Open door policy – don't hesitate to stop by!
- TAs: Slava Chernyak (labs), Alexei Czeskis (discussion sections), Miro Enev (homeworks)
  - Office/hours: See website (TBD)
- Course website
  - Assignments, reading materials, ...
- Course email list
  - Announcements
- Course forum
  - Discussion

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)

- Strongly 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 (machine organization and assembly language) and one of CSE 451 / CSE 461 (OS / networking)

Course Logistics (CSE 484)

- Lectures: Mon, Wed, Fri: 8:30--9:20am ;
  Recitations: Thurs: 8:30--9:20am and 9:30--10:20am
- Security is a contact sport!
- Labs (40% 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 (25% of grade)
- Participation (10% of grade)
- 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 (35% of the grade)
- Homeworks (20% of grade)
- Participation (10% of grade)
- Final (25% of the grade)
- Research readings (10%)
  - Read research papers (1 per week for first 9 weeks)
  - Possibly present one of these papers to the class (depending on enrollment)

Late Submission Policy

- Late assignments will (generally) be dropped 20% per day.
  - Late days will be rounded up
  - So an assignment turned in 26 hours late will be downgraded 40%.
  - See website for exceptions
- Everything is generally due on Friday

Participation Grade

- Regular contributions to class forum
- Participation in class
  - We will have a seating chart … at least until I learn everyone's names.
  - On Wednesday, please pick a seat that you'd like to have for at least the first part of the quarter

Small class in a large class

- This class has ~60 enrolled students
- Hard to have 1-on-1 interactions; not very personal
- Coffee / tea?
  - Approximately once a week for the first half of the quarter, let's go as a small group for coffee or tea (~8 or 9 students and me)
  - Not required.
  - But an opportunity for all of us to get to know each other better, to discuss security, the broader context, thoughts about the course, current movies, ...
  - Sign up form will be on the website soon

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 include:
  - Little Brother by Cory Doctorow
    - Available online here http://craphound.com/littlebrother/download/
    - I highly recommend that everyone reads this
  - Cryptonomicon by Neal Stephenson
- Movies include:
  - Hackers
  - Sneakers
  - Diehard 4
  - Wargames
- Historical texts include:
  - 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 class on Friday (Jan 8).
- http://www.cs.washington.edu/education/courses/484/10wi/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/484/10wi/administrivia/email.html
- Used for announcements

Forum

- We've set up a forum for this course
  - https://catalysttools.washington.edu/gopost/board/kohno/14597/
- 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 (tentative):
  - 4 homeworks, approximately once every two weeks
    - Jan 22, Feb 5, Feb 19, March 4
    - First one posted online by Friday
  - Due Fridays at 11am.
  - Submit to Catalyst system (URL on course page)
- http://www.cs.washington.edu/education/courses/484/10wi/homework/index.html

Labs

- Tentative schedule online (future dates subject to change based on progress, etc)
- General plan (tentative):
  - 3 labs
    - Jan 29, Feb 12, March 12
    - First one posted online by next Monday
  - 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/484/10wi/projects/index.html

Labs (tentative plan)

- First lab: Software security
  - Buffer overflow attacks, double-free exploits, format string exploits, ...
- Second lab: Web security
  - XSS attacks, ...
- Third lab: Botnets (tentative)
  - Build a botnet, command and control, leasing, crypto, ...

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 forum
    - http://cubist.cs.washington.edu/Security/ (last year)
    - http://slashdot.org/
- **Technical aspects** of security
  - Attack techniques
  - Defenses

How to think about security

- Several approaches for developing "The Security Mindset" and for exploring the broader contextual issues surrounding computer security
  - Forum: Current event reflections
  - Forum: Security reviews
  - Science fiction prototyping
  - In class discussions
  - Additional participation in forums

Forum: Current events and security reviews

- One current event posted by Feb 5 (at 3pm)
- One security review posted by Feb 5 (at 3pm)
- 12 points each
- 1 point extra credit for each week that you are early
- May work in groups of up to 3 people.
  - Working in groups is actually encouraged.
  - Recall: security is a contact sport -- lots of value in discussing security with other people
- Please participate in follow-up discussions on forum

Forum: Current events and security reviews

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

Science Fiction Prototyping

- Science fiction prototyping: new techniques for exploring potential implications of new technologies
- In many ways, a perfect match for security

- Key ideas:
  - Take new technology
  - Place new technology in the context of people, society and explore that with a story
  - Use that story to draw lessons about the technologies themselves
- Jan 22: Guest lecture to bootstrap this effort from science fiction author Brian David Johnson
- Tentative plan: Deadlines on Feb 12 and Feb 29.
- Your security reviews and current events articles might give you good ideas for possible technologies to explore

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 the topic is huge
- 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

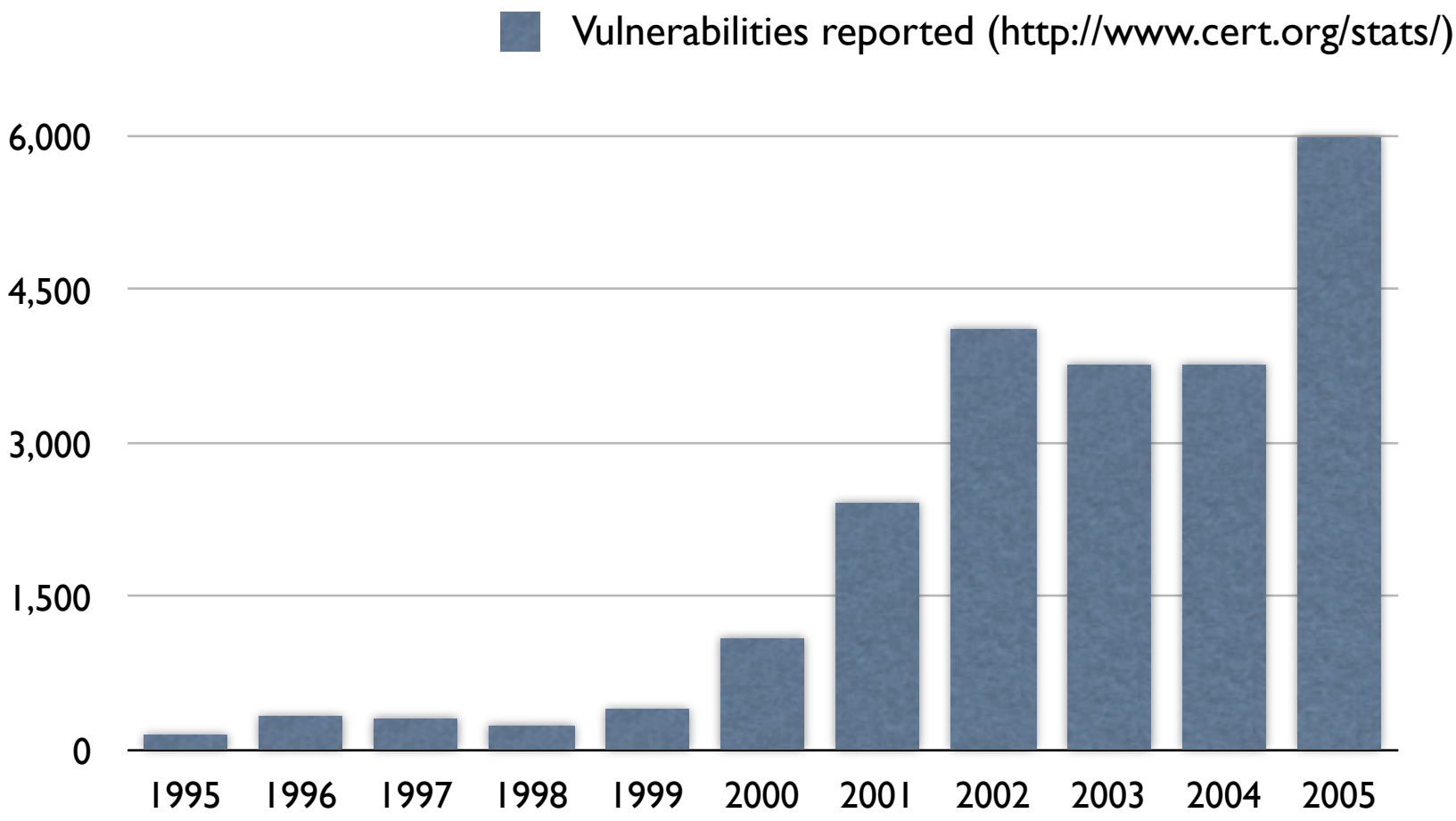
What is Computer Security?

- 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

Current events, security reviews, and science fiction prototyping all designed to exercise our thinking about these issues

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

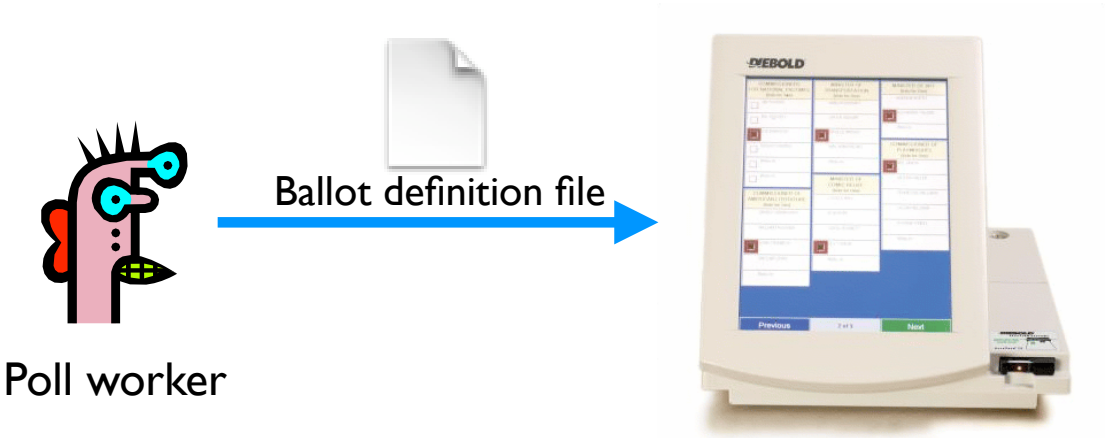
- The threat of a response may be enough to deter some attackers

Example: Electronic Voting

- Popular replacement to traditional paper ballots

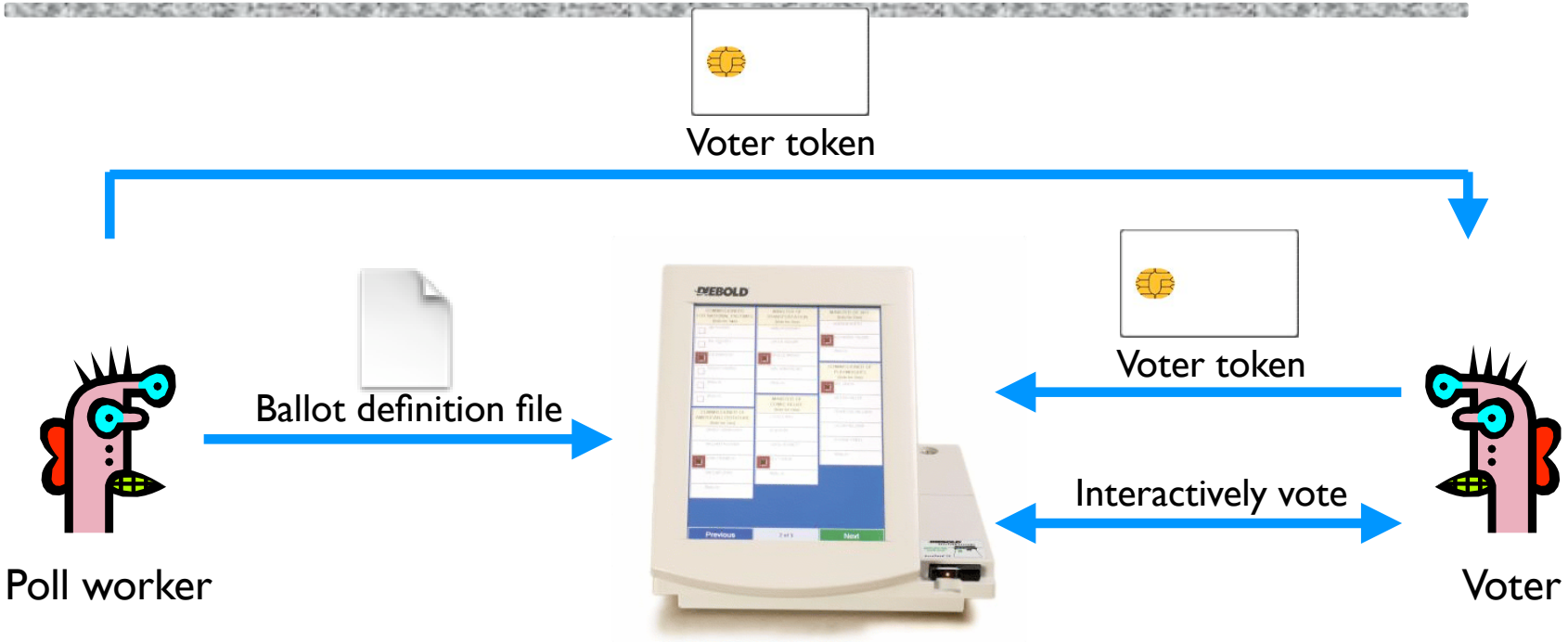
AP

Pre-Election



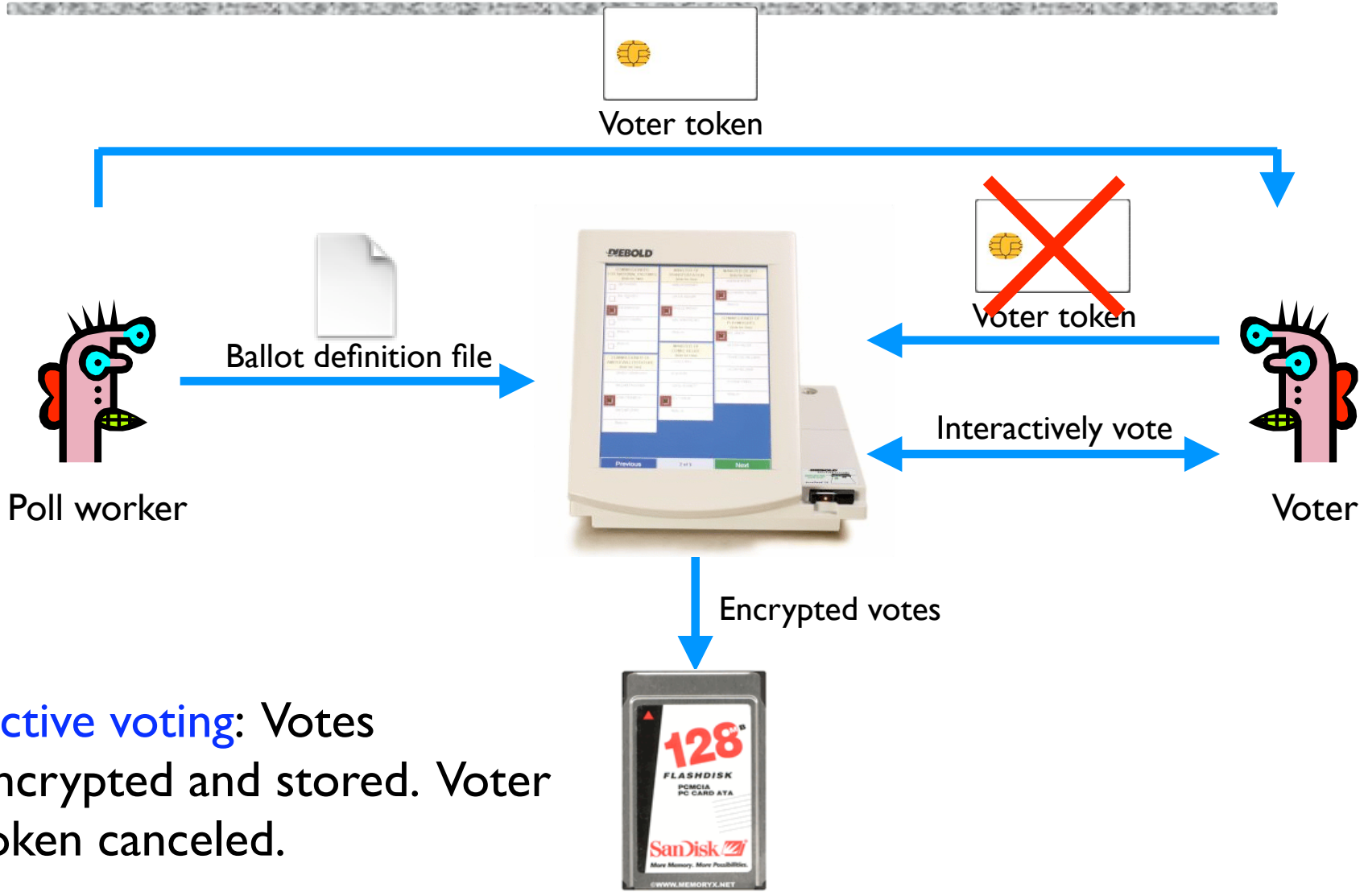
Pre-election: Poll workers load "ballot definition files" on voting machine.

Active Voting



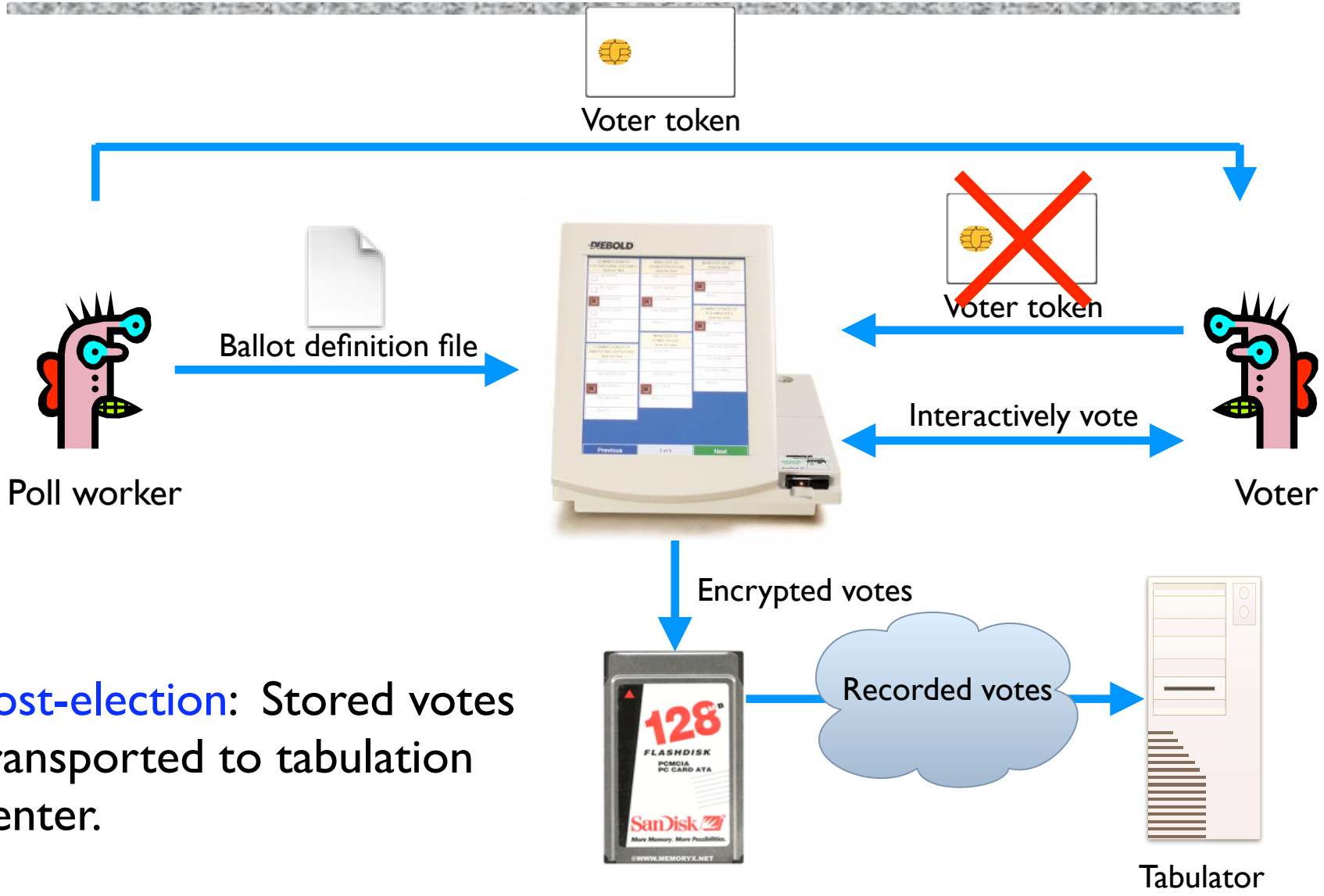
Active voting: Voters obtain single-use tokens from poll workers. Voters use tokens to active machines and vote.

Active Voting



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

Post-Election

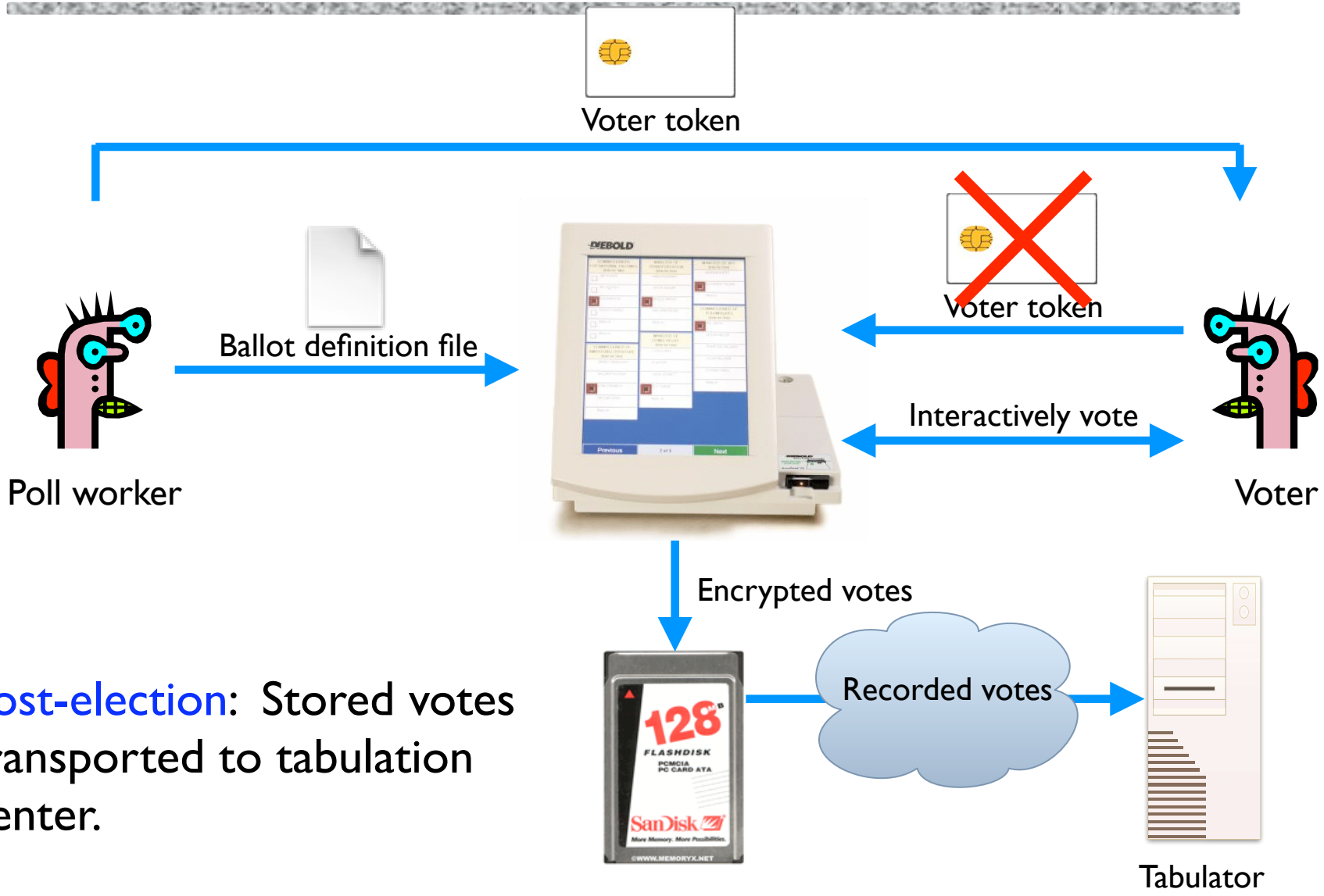


Post-election: Stored votes transported to tabulation center.

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

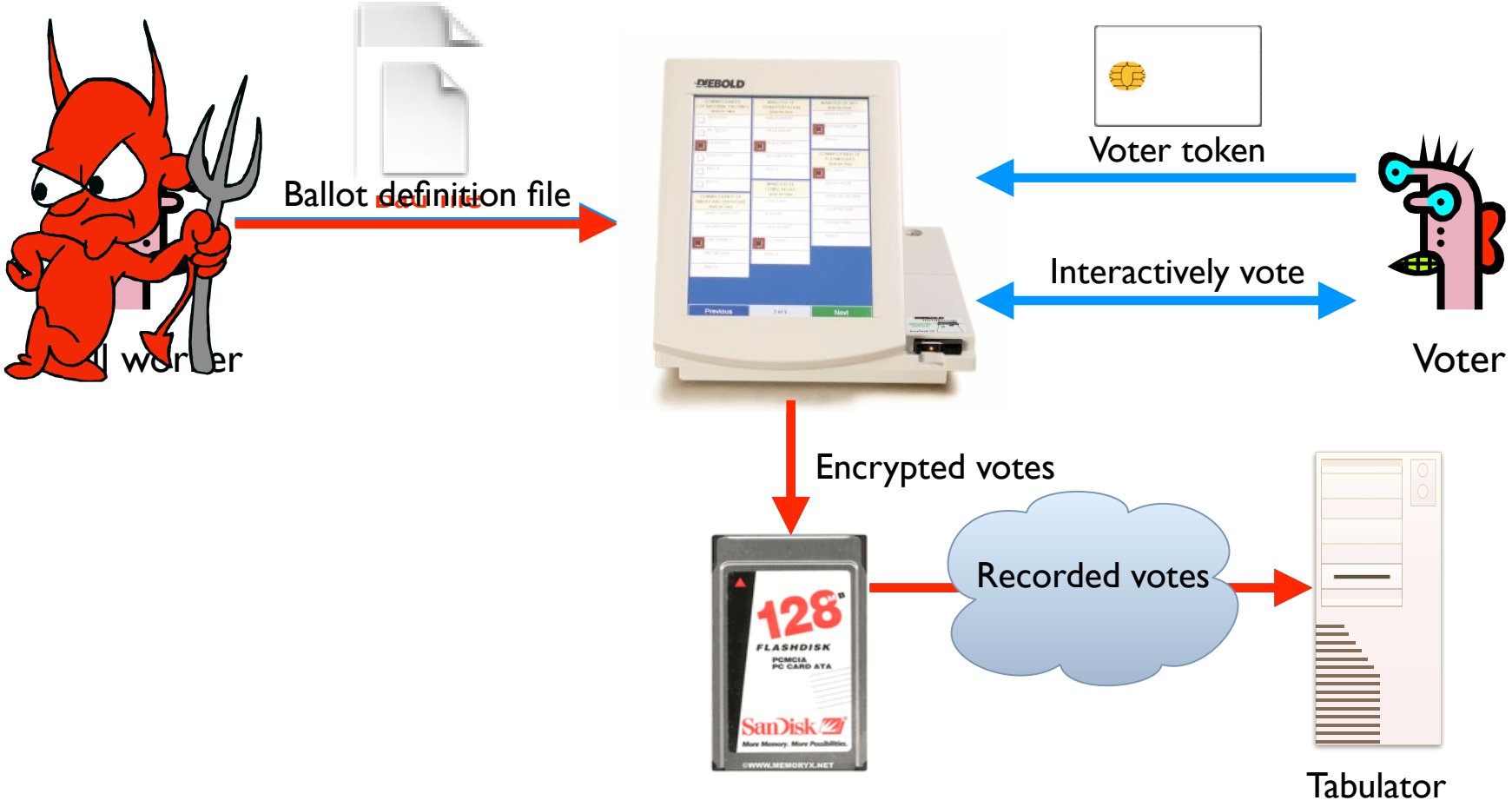
- Voters
- Election officials
- Employees of voting machine manufacturer
  - Software/hardware engineers
  - Maintenance people
- Other engineers
  - Makers of hardware
  - Makers of underlying software or add-on components
  - Makers of compiler
- …
- Or any combination of the above

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.

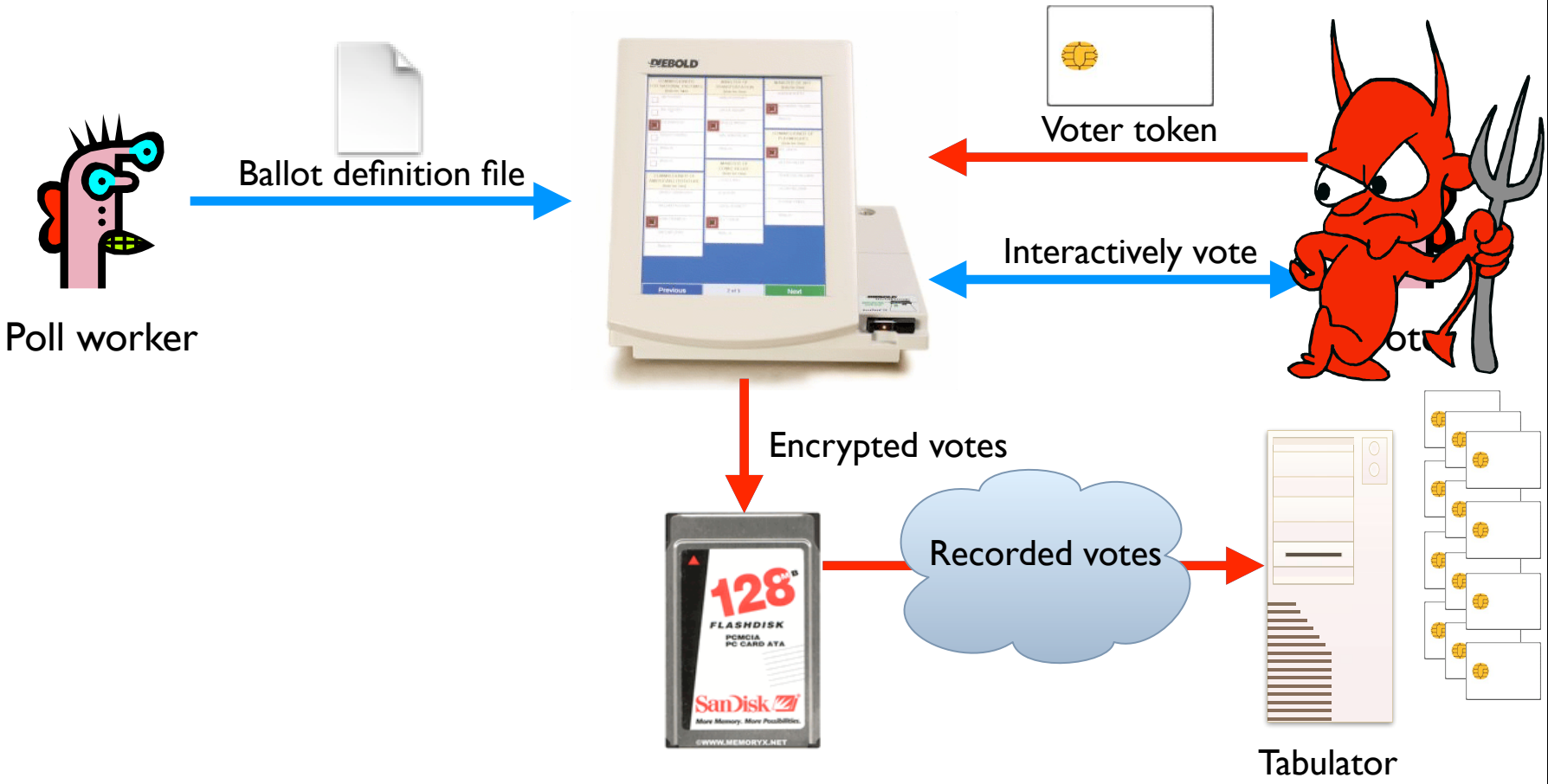
Problem: Ballot definition files are not authenticated.

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



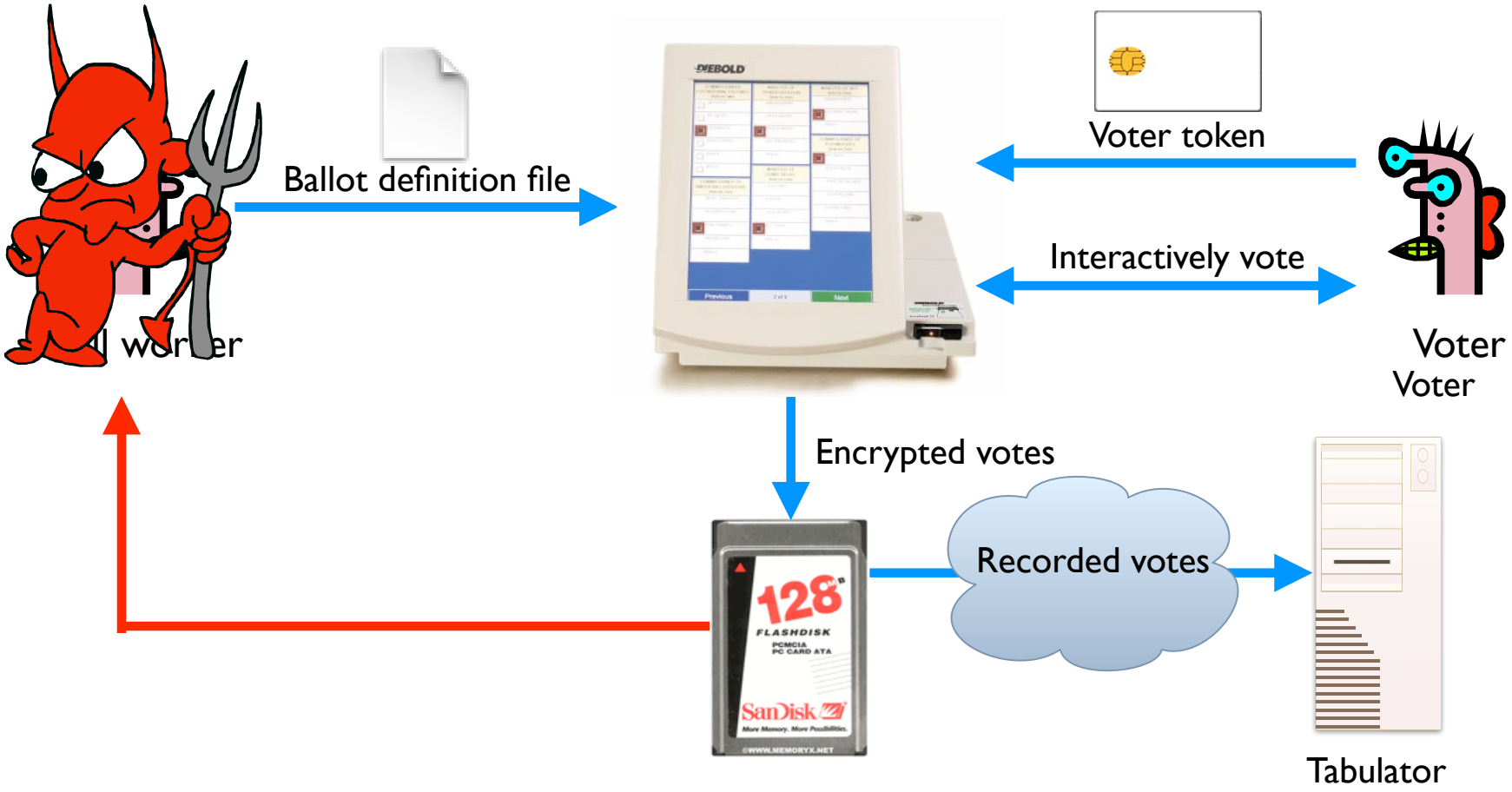
Problem: Smartcards can perform cryptographic operations. But there is no authentication from voter token to terminal.

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



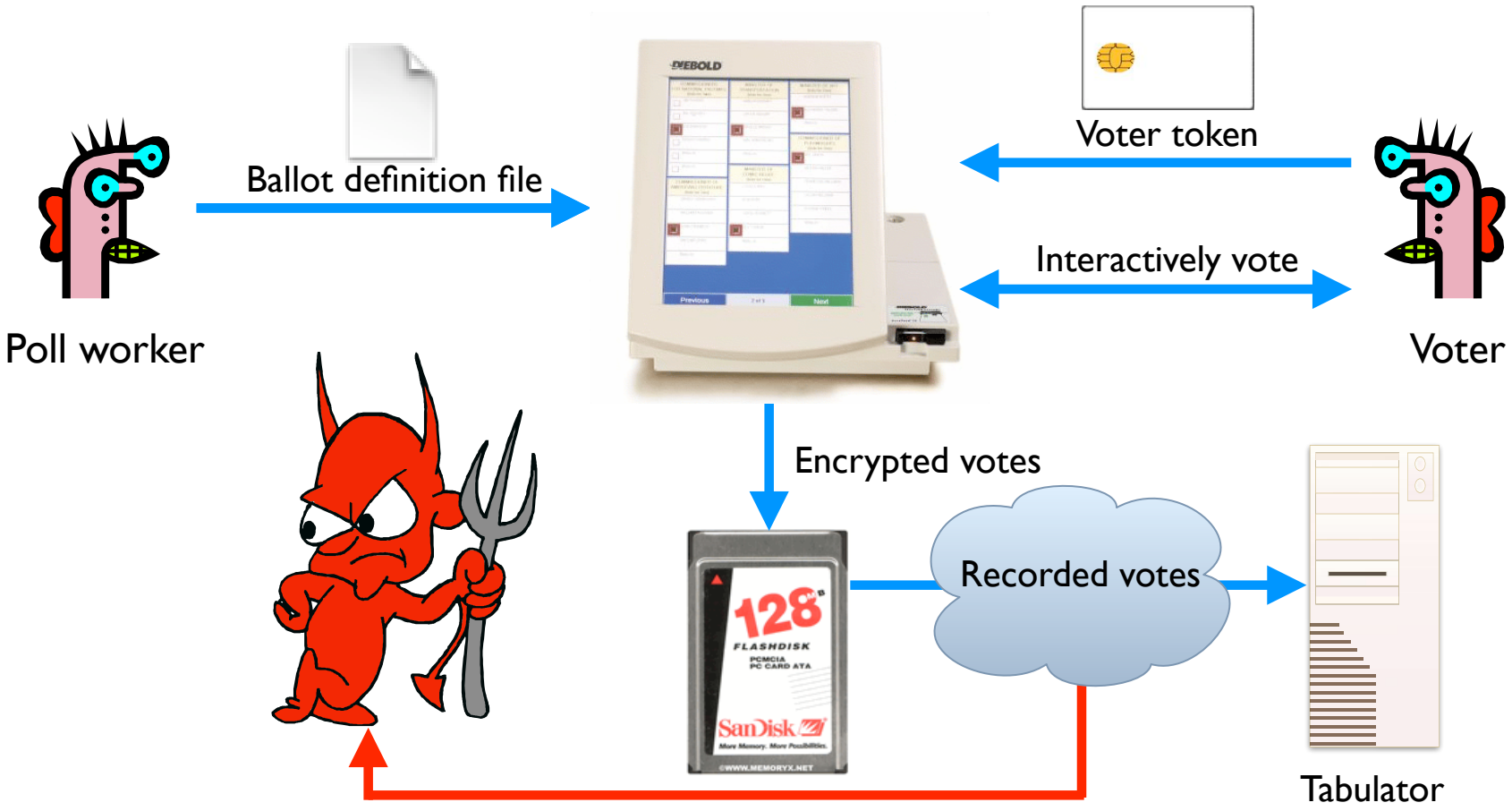
Problem: Encryption key ("F2654hD4") hard-coded into the software since (at least) 1998. Votes stored in the order cast.

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



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

Implantable Medical Devices