CSE 484 / CSE M 584: Computer Security and Privacy

Spring 2020

Franziska (Franzi) Roesner
franzi@cs.washington.edu

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

Announcements

- Online course logistics
  - **Participation:** We will try something today, and will follow up with a concrete plan for grading
    - Don't worry about this week, we are all adapting!
  - We will try **non-random breakout rooms** in the future (stay tuned for instructions)
  - **Recordings:** Includes student speech/video/chat (don't share if you don't want to), won't be shared outside the class
- Things Due:
  - Ethics form: Due next Wednesday (4/8)
  - Homework #1: Due next Friday (4/10)
    - Start forming groups, feel free to continue using Ed forum

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

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 or goals**?
- Perfect security does not exist!
  - Security is not a binary property
  - Security is about risk management

Current events, security reviews, and other discussions are designed to exercise our thinking about these issues.

Two Key Themes of this Course

1. How to **think** about security
   - The "Security Mindset" – a "new" way to think about systems
2. **Technical aspects of security**
   - Vulnerabilities and attack techniques
   - Defensive technologies
   - Topics including: software security, cryptography, malware, web security, web privacy, smartphone security, authentication, usable security, anonymity, physical security, security for emerging technologies
   - (There's a lot we are not covering!)

Theme 1: Security Mindset

- Thinking critically about designs, challenging assumptions
- Being curious, thinking like an attacker
- "That new product X sounds awesome, I can't wait to use it!" versus "That new product X sounds cool, but I wonder what would happen if someone did Y with it…"

- Why it's important
  - Technology changes, so learning to think like a security person is more important than learning specifics of today
  - Will help you design better systems/solutions
  - Interactions with broader context: law, policy, ethics, etc.

Learning the Security Mindset

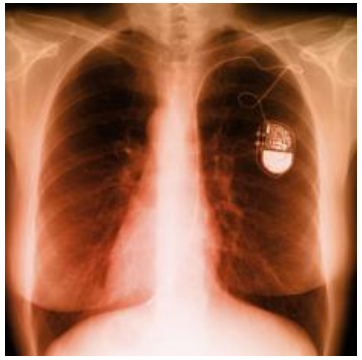
- Several approaches for developing "The Security Mindset" and for exploring the broader contextual issues surrounding computer security
  - Homework #1
    - Current event reflections and security reviews
    - Groups up to 3 people (lots of value in discussing security with others!)
  - In class discussions and activities
  - Participation in Ed discussion board (e.g., critiquing movies)

Security: Not Just for PCs

smartphones

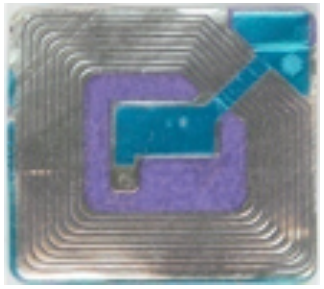
voting machines

EEG headsets



medical devices

wearables



RFID

mobile sensing platforms

cars

game platforms

airplanes

THREAT MODELING

Threat Modeling

- There's no such thing as perfect security
  - But, attackers have limited resources
  - **Make them pay unacceptable costs to succeed!**
- Defining security per context: identify assets, adversaries, motivations, threats, vulnerabilities, risk, possible defenses

Threat Modeling (Security Reviews)

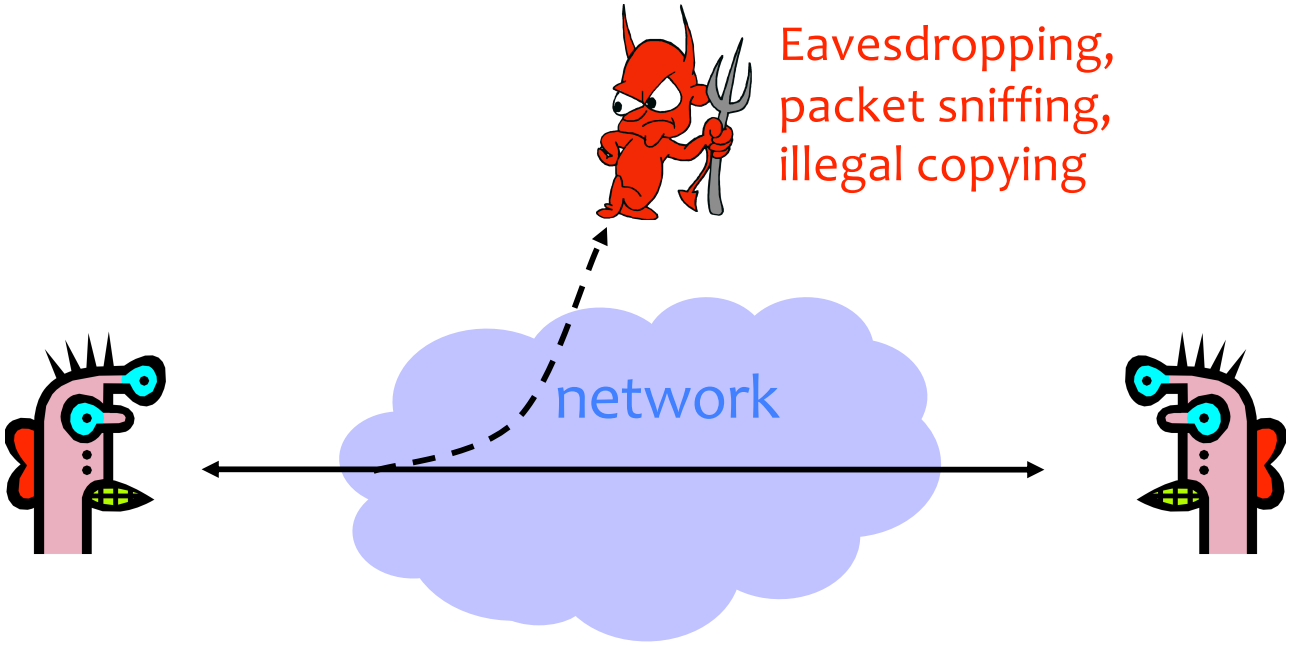
- Assets: What are we trying to protect? How valuable are those assets?
- Adversaries: Who might try to attack, and why?
- Vulnerabilities: How might the system be weak?
- Threats: What actions might an adversary take to exploit vulnerabilities?
- Risk: How important are assets? How likely is exploit?
- Possible Defenses

What's *Security*, Anyway?

- Common general security goals: "CIA"
  - Confidentiality
  - Integrity
  - Authenticity
  - Availability

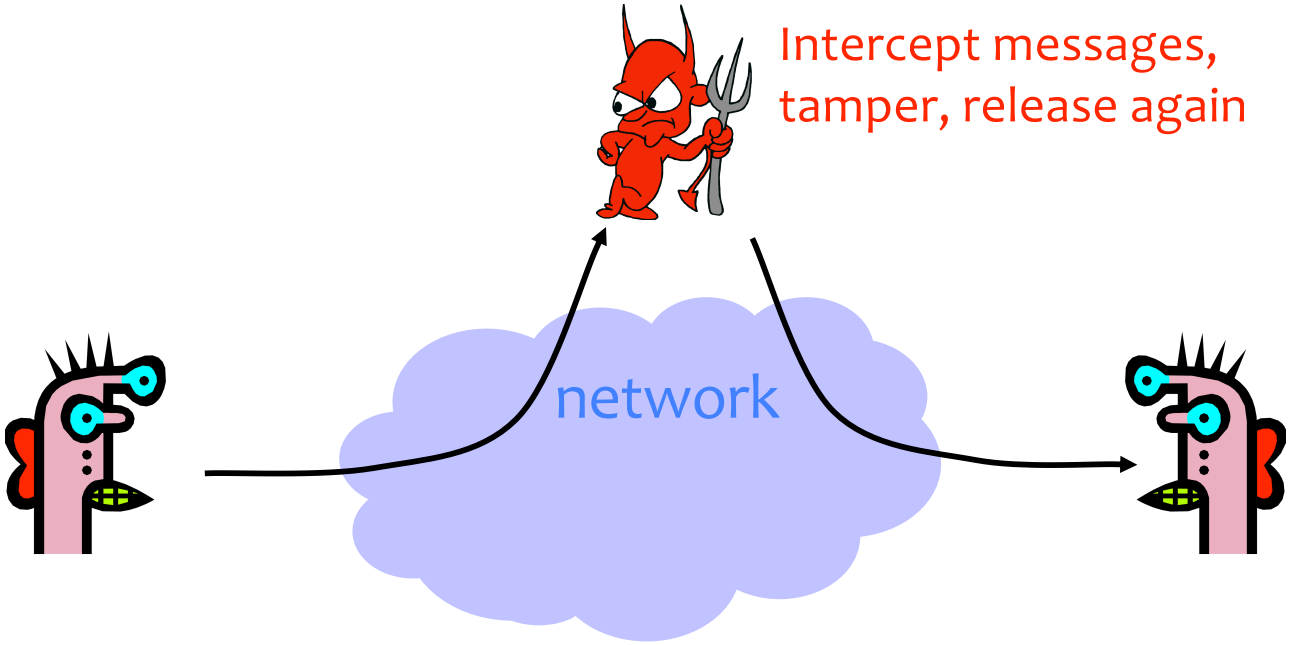
Confidentiality (Privacy)

- Confidentiality is concealment of information.



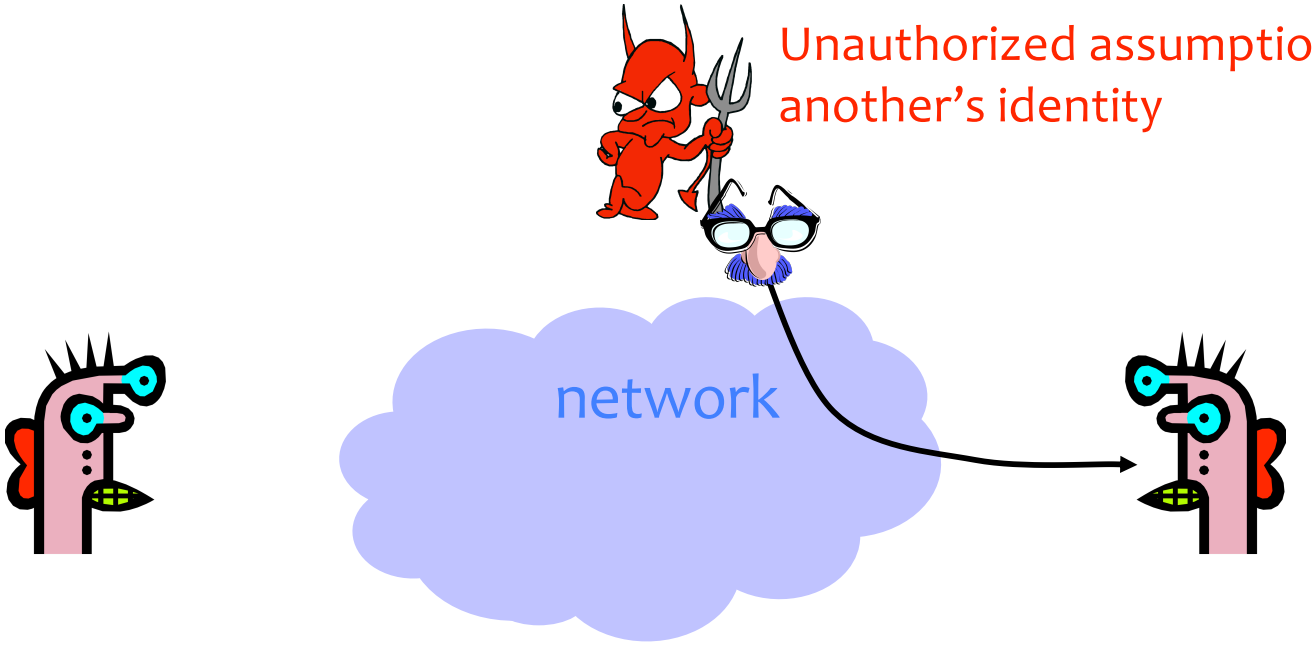
Integrity

- Integrity is prevention of unauthorized changes.



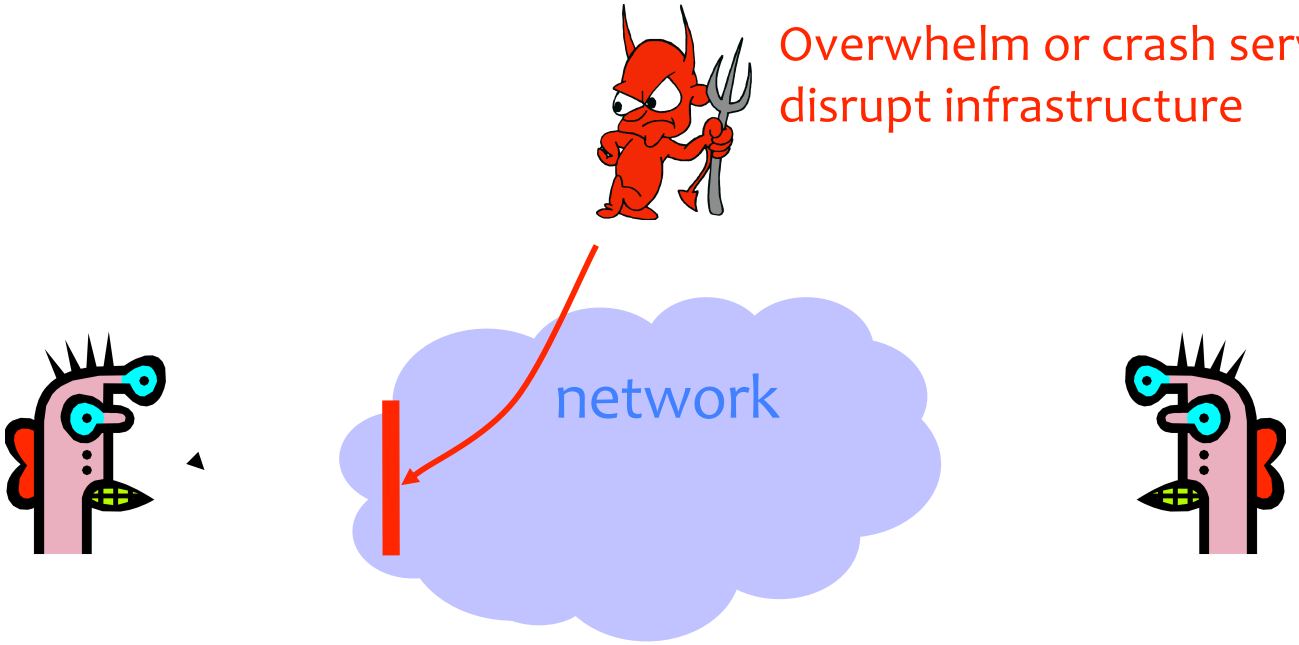
Authenticity

- Authenticity is knowing who you're talking to.



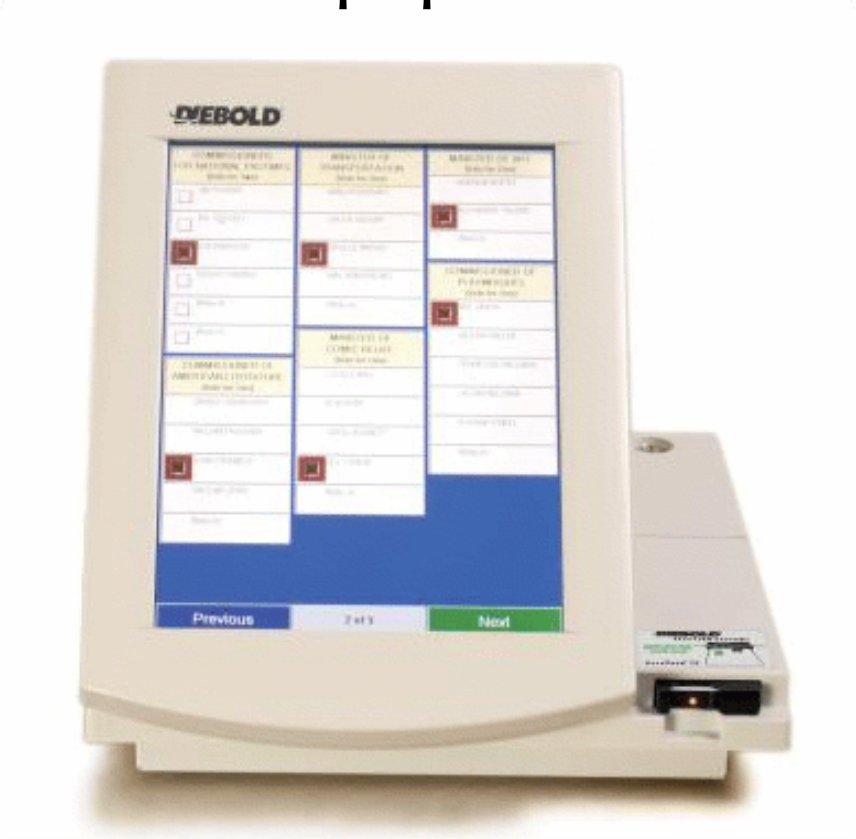
Availability

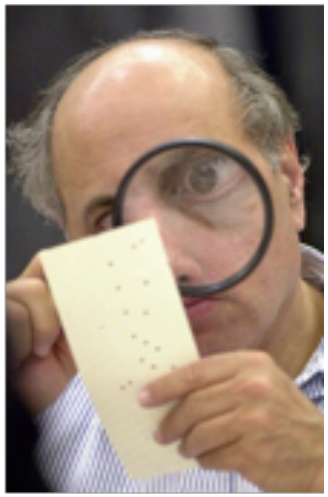
- Availability is ability to use information or resources.



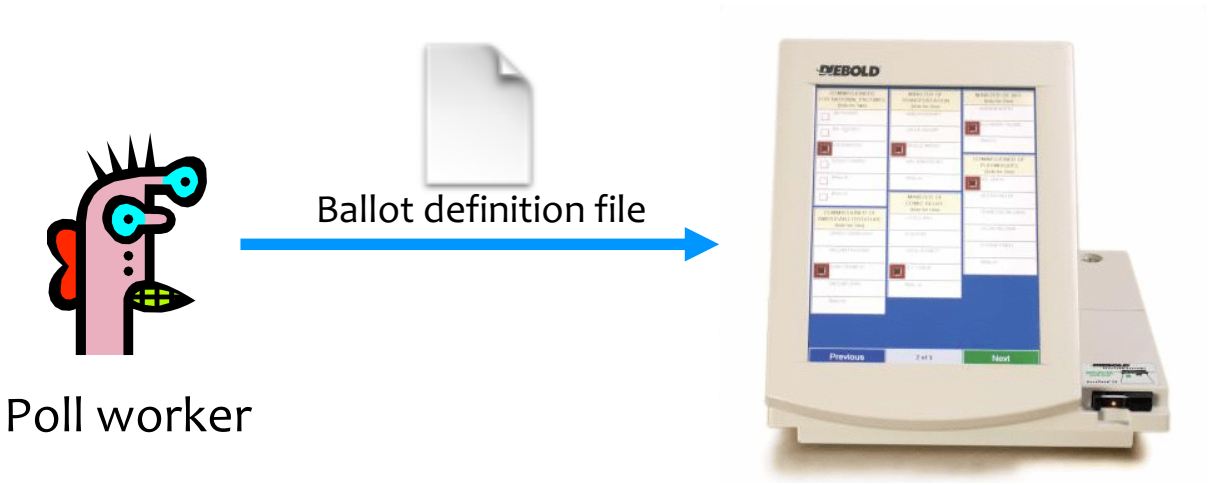
Threat Modeling Example: Electronic Voting

- Popular replacement to traditional paper ballots



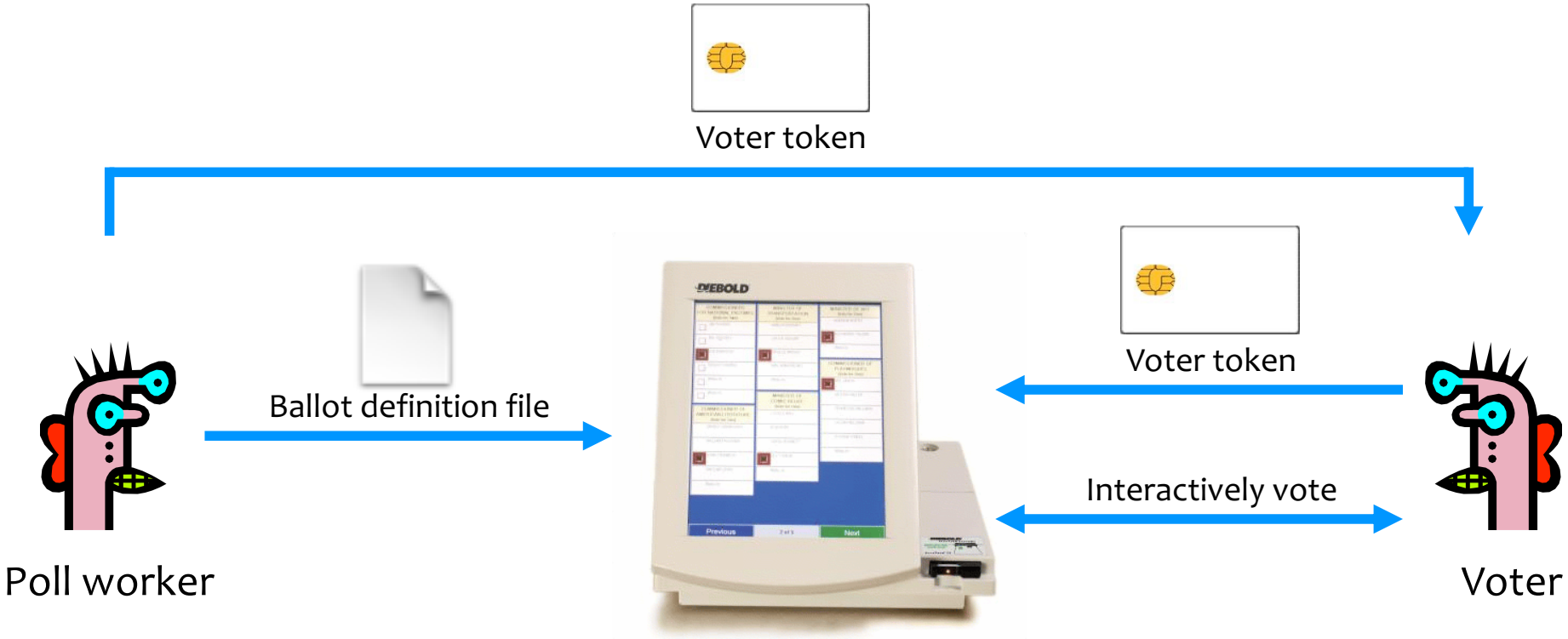


Pre-Election



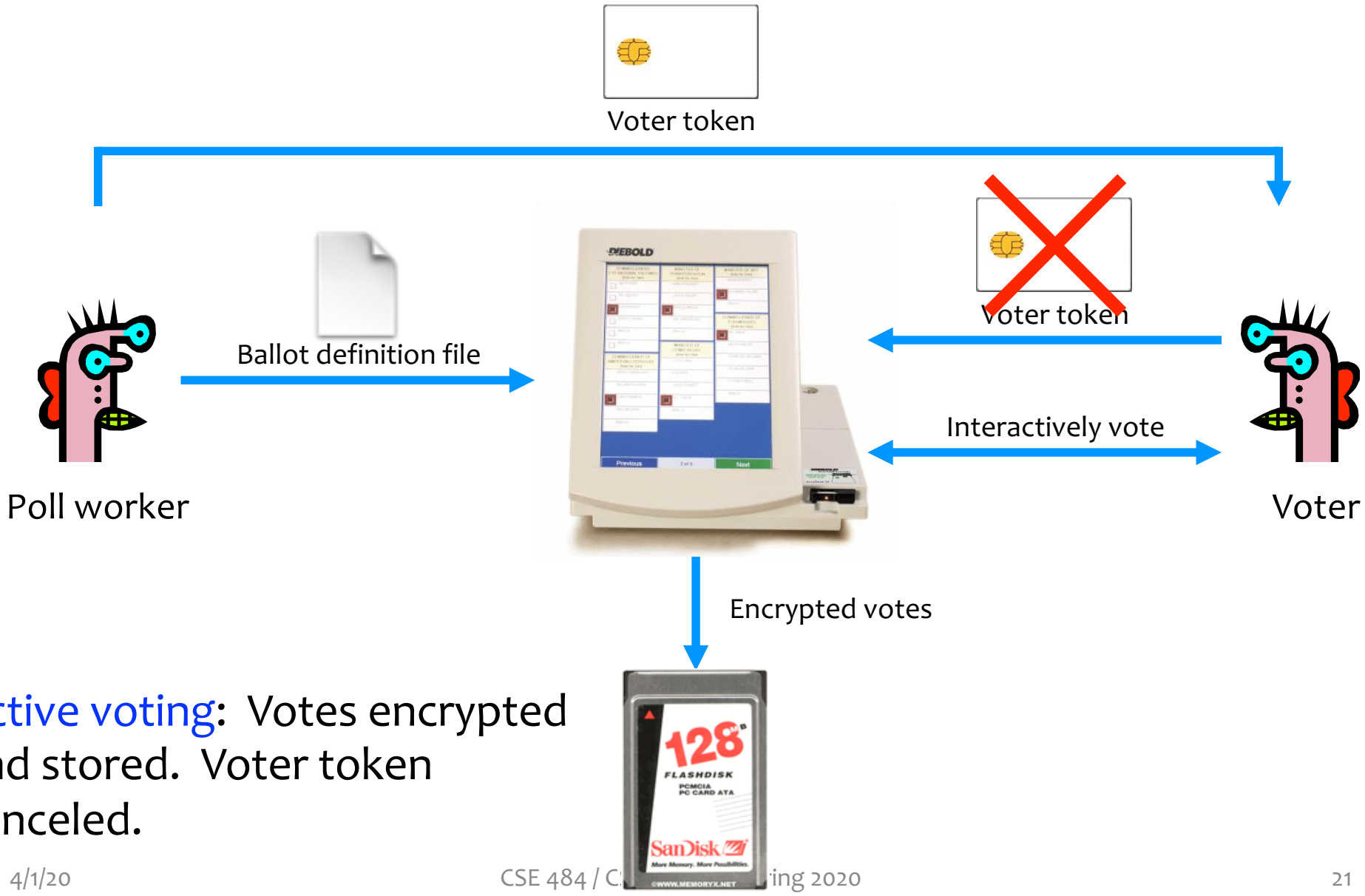
Pre-election: Poll workers load "ballot definition files" on voting machine.

Active Voting



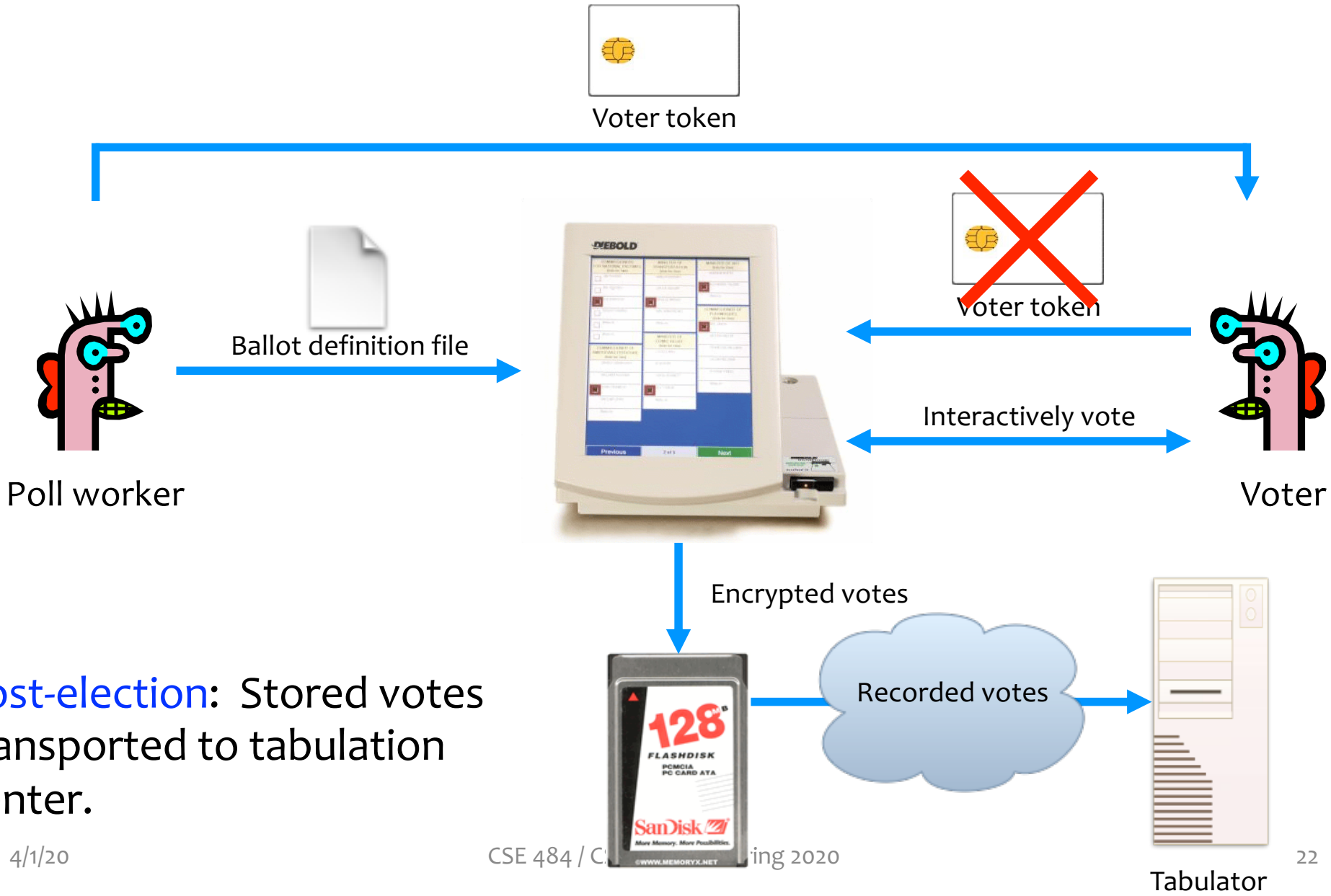
Active voting: Voters obtain single-use tokens from poll workers. Voters use tokens to activate machines and vote.

Active Voting



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

Post-Election



Post-election: Stored votes transported to tabulation center.

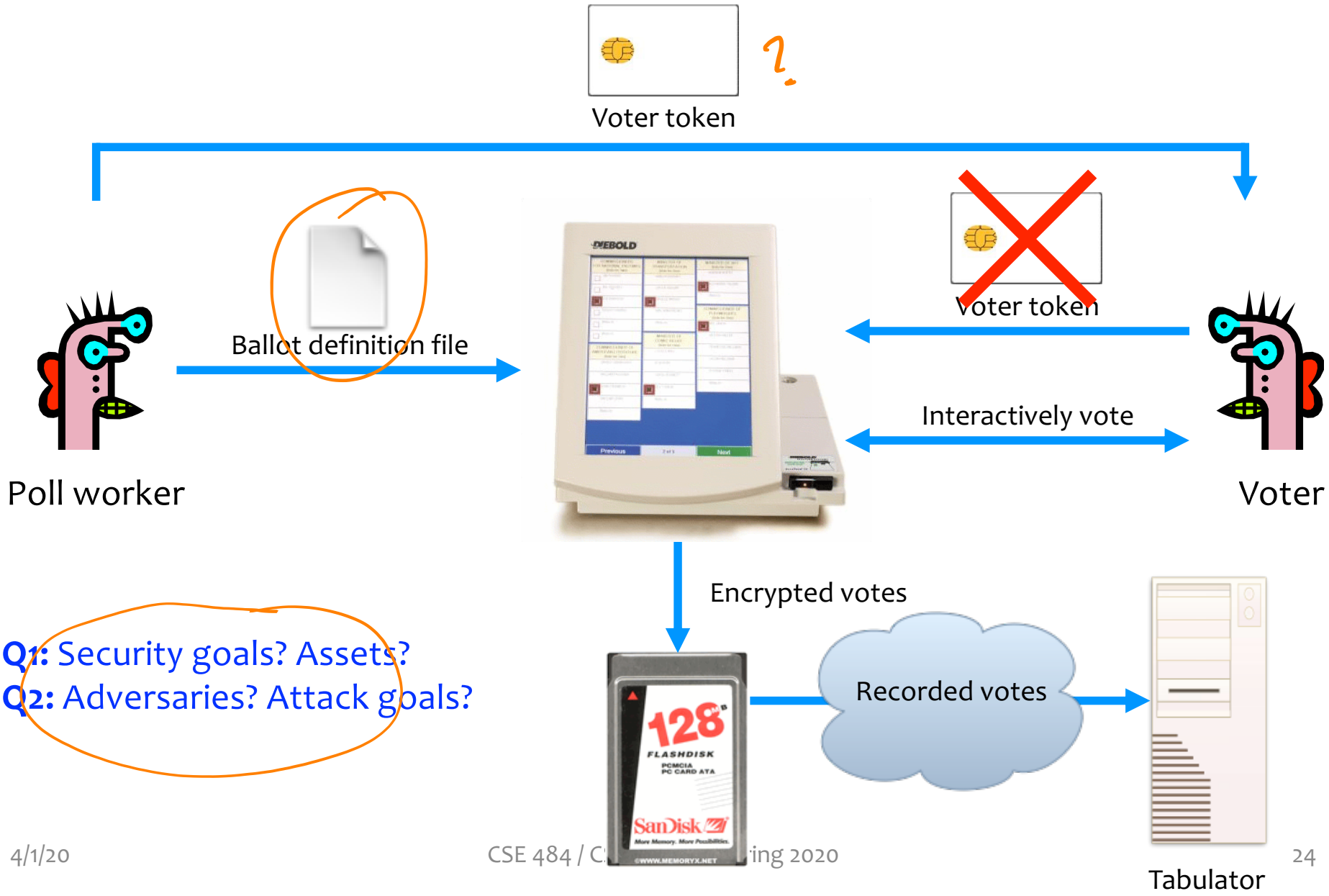
In-Class "Worksheet" Experiment

- Go to Canvas -> Quizzes -> "In-Class Activity - April 1"

  Direct link: https://canvas.uw.edu/courses/1371936/quizzes/1232393

- Fill out the questions while discussing with your breakout group
  - Everyone should submit their own
  - **No need for polish or complete sentences** – jot things down as you would on a piece of paper while chatting in class

Can You Spot Any Potential Issues?



Security and E-Voting (Simplified)

- Functionality goals:
  - Easy to use, reduce mistakes/confusion
- Security goals:

trust in election / gov't

Security and E-Voting (Simplified)

- Functionality goals:
  - Easy to use, reduce mistakes/confusion
- Security goals:
  - Adversary should not be able to tamper with the election outcome
    - By changing votes (**integrity**)
    - By voting on behalf of someone (**authenticity**)
    - By denying voters the right to vote (**availability**)
  - Adversary should not be able to figure out how voters vote (**confidentiality**)

Potential Adversaries

+ Threats

- poll worker mess w/ ballot definition file
- poll worker manipulates voting tokens
- denial of service in specific places

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

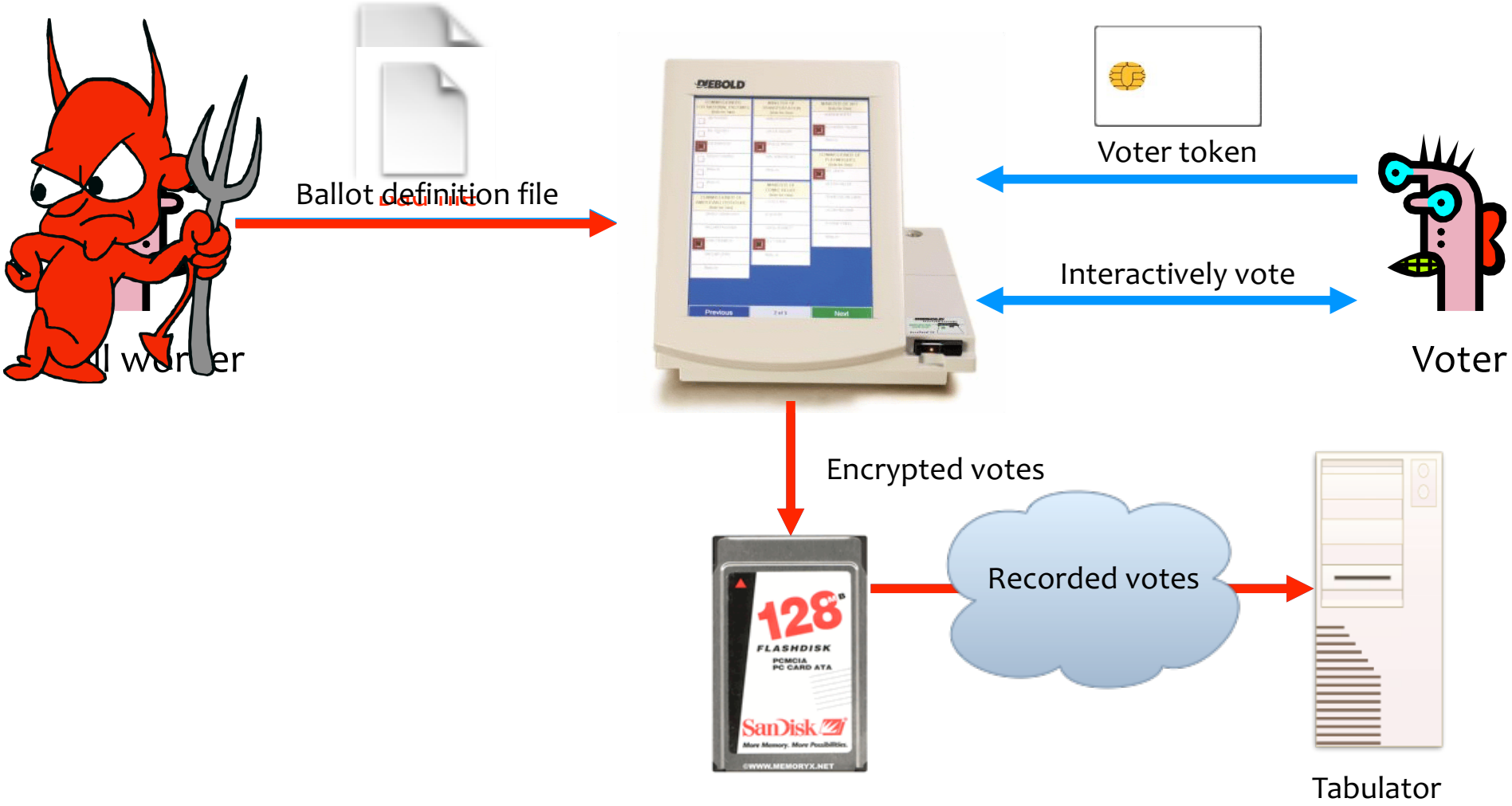
What Software is Running?

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.


KEYS TO THE KINGDOM
Photo taken from Diebold's online store. The keys that open every Diebold touch-screen voting machine. Working copies have been made from the photo.

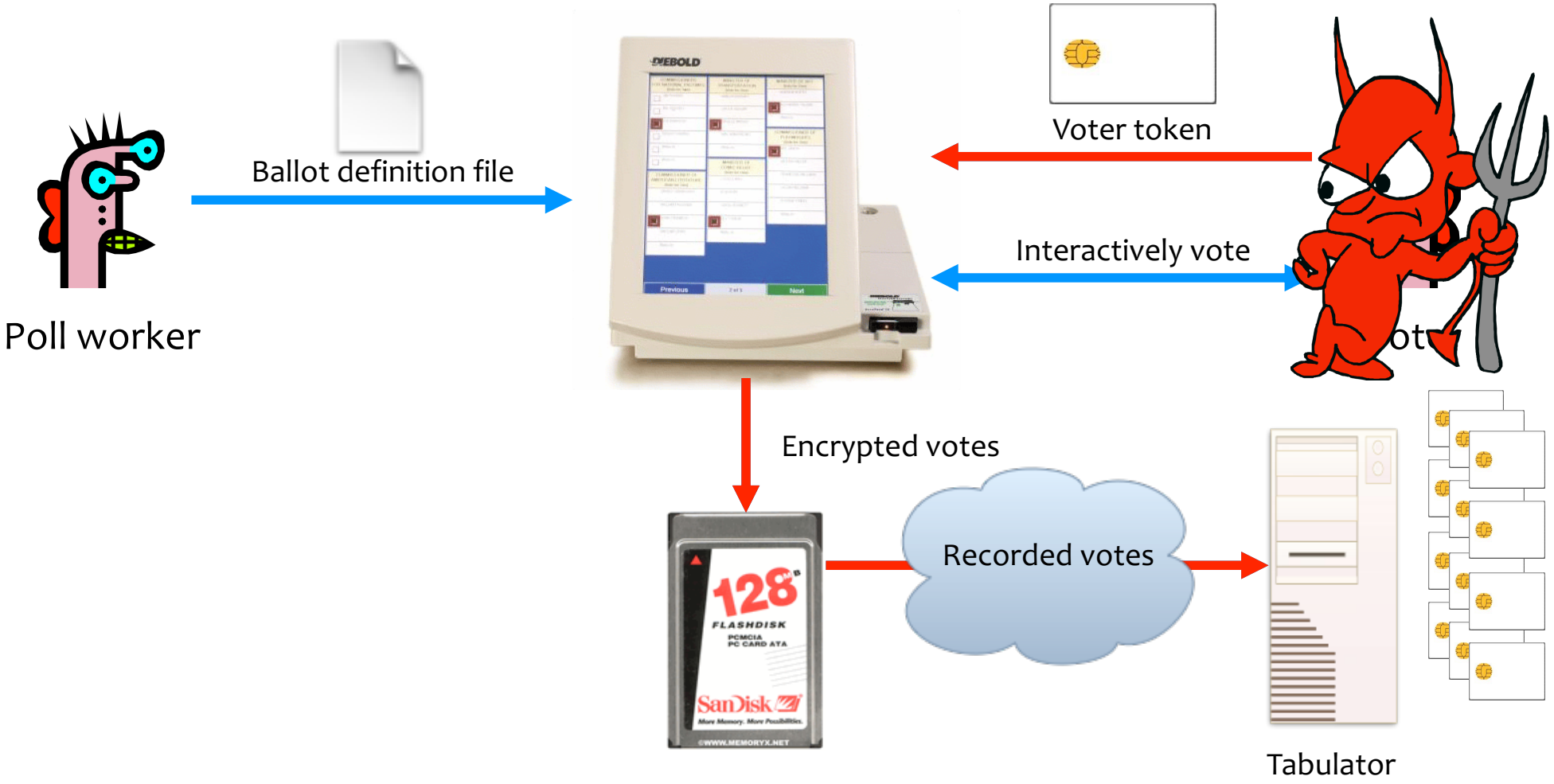
Problem: Ballot definition files are not authenticated.

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



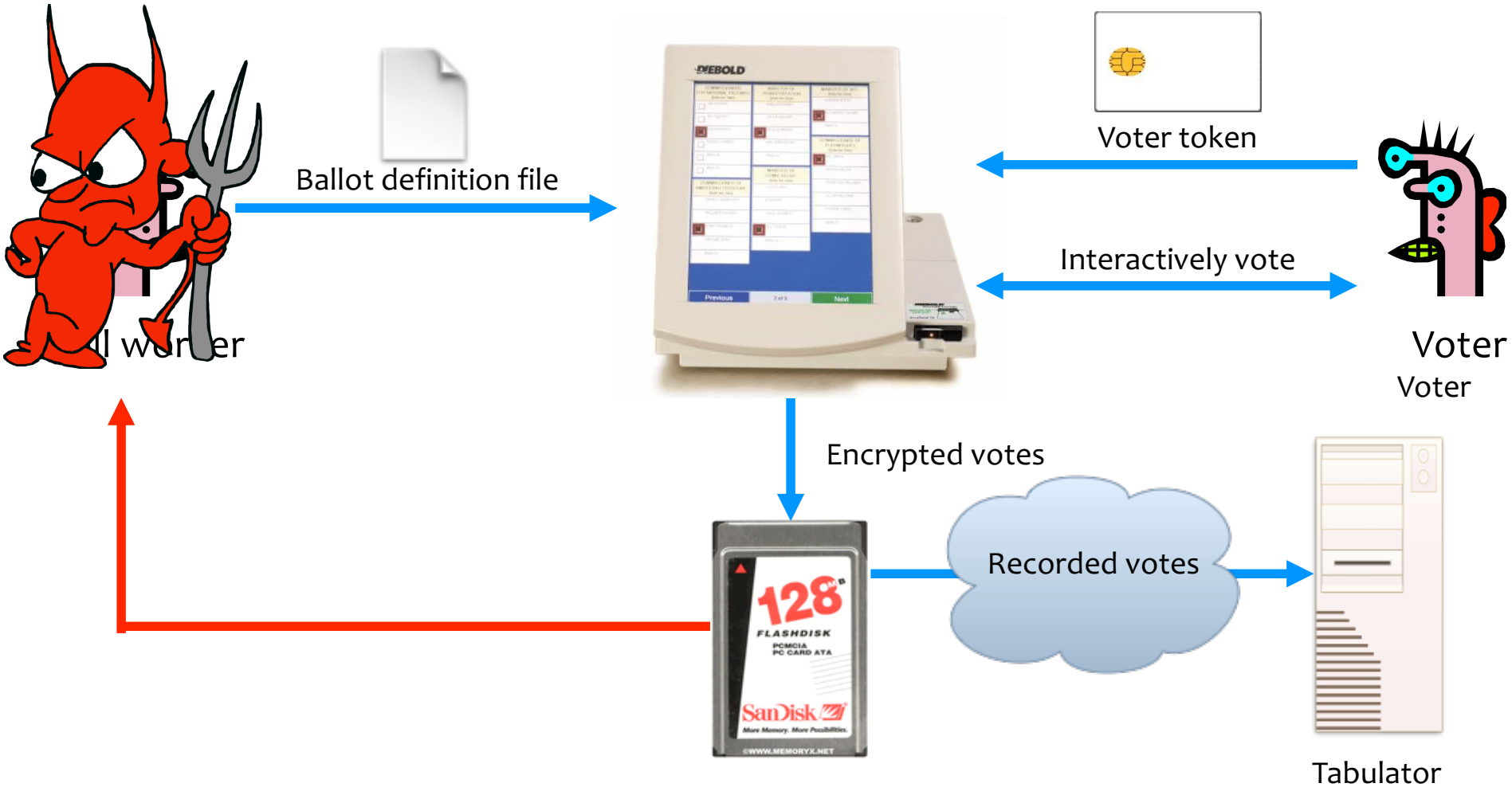
Problem: Smartcards can perform cryptographic operations. But there is no authentication from voter token to terminal.

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.