CSE 484 / CSE M 584: Computer Security and Privacy

Fall 2022

Franziska (Franzi) Roesner
franzi@cs

UW Instruction Team: David Kohlbrenner, Yoshi Kohno, Franziska Roesner. Thanks to Dan Boneh, Dieter Gollmann, Dan Halperin, John Manferdelli, John Mitchell, Vitaly Shmatikov, Bennet Yee, and many others for sample slides and materials ...

Announcements

- Things Due:
  - Ethics Form: Due Monday
    - Note: sign in with your CSE Google accounts, not UW
  - Homework #1: Due next Friday
    - Use the Ed discussion board (or Discord) to find groups
    - Join a group on Canvas
  - Research Readings (CSE M 584): Due next Thursday (and every Thursday thereafter)
- Apologies to yesterday's 1:30pm (AA) Section!
  - Slides posted; onward to next week ☺

Ethics

- To learn to defend systems, you will learn to attack them. You must use this knowledge ethically.

- In order to get a non-zero grade in this course, you must electronically sign the "Security and Privacy Code of Ethics" form by 11:59pm on Monday, October 3.

(Linked from the course schedule)

We will also repeatedly consider ethics (more generally) as part of our curriculum throughout course (see HW1, for example).

Late Submission Policy

- 5 free late days, no questions asked
  - Cumulative, throughout the quarter
  - Use up to 3 for one submission
  - All group members use days at once
- After that, late assignments will 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 -- a small number of assignments must be turned in on time
- **Please write on the assignment how many late days you are using!**

THREAT MODELING

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
- Not "traditional" threat modeling, but important:
  - Benefits: Who might the system benefit, and how?
  - Harms: Who might the system harm, and how?

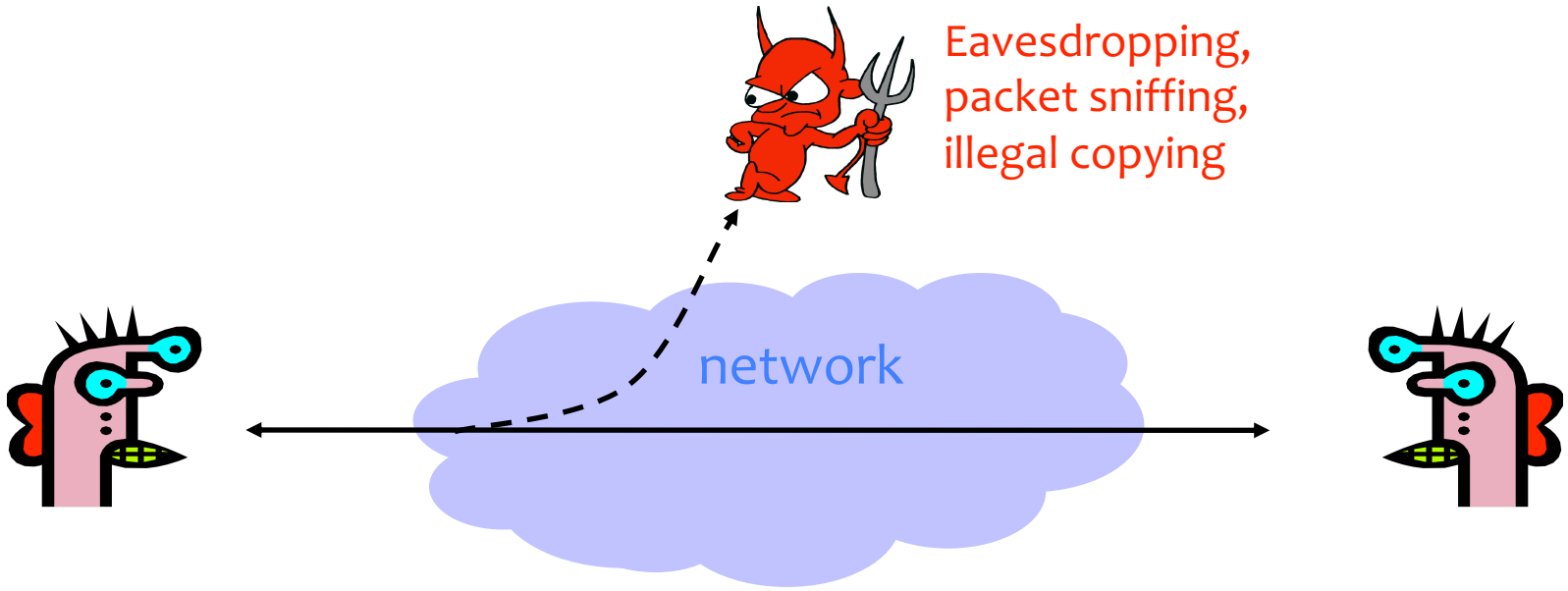
What's *Security*, Anyway?

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

- Or the extension: CPIAAU (Parkerian Hexad)
  - Confidentiality
  - Possession or Control
  - Integrity
  - Authenticity
  - Availability
  - Utility

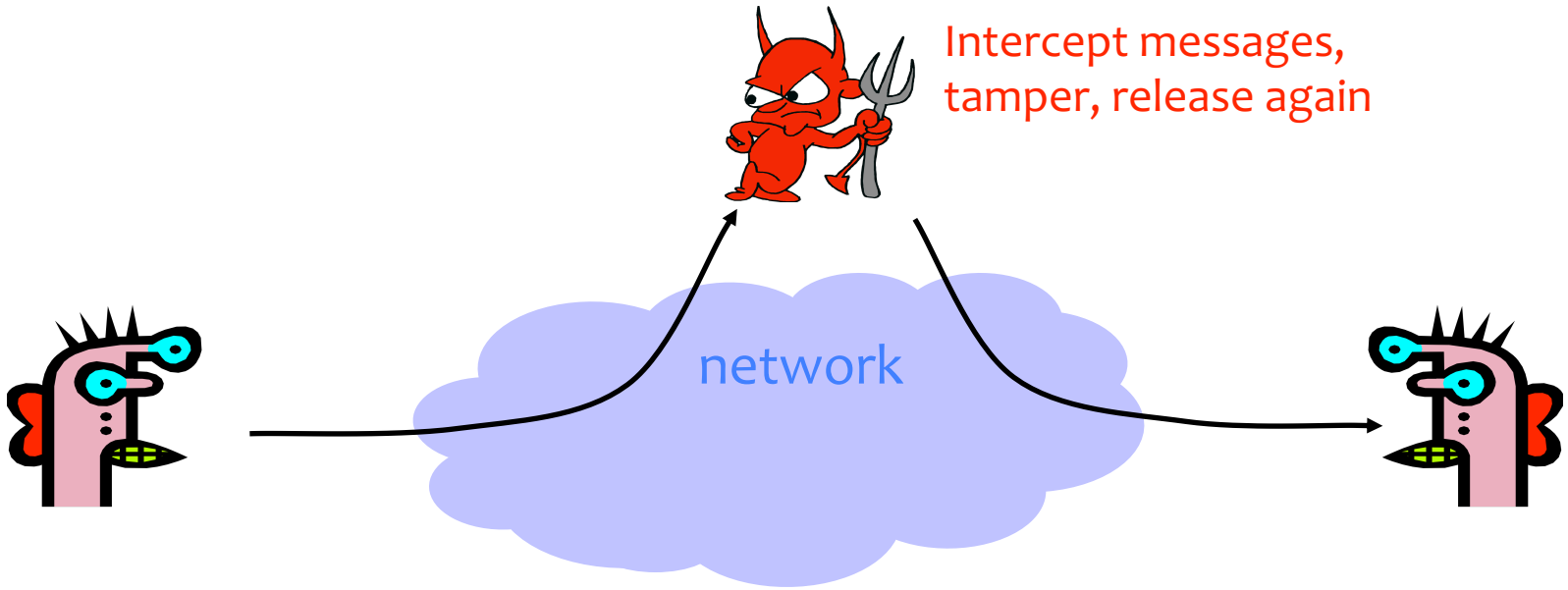
Confidentiality (Privacy)

- Confidentiality is concealment of information.



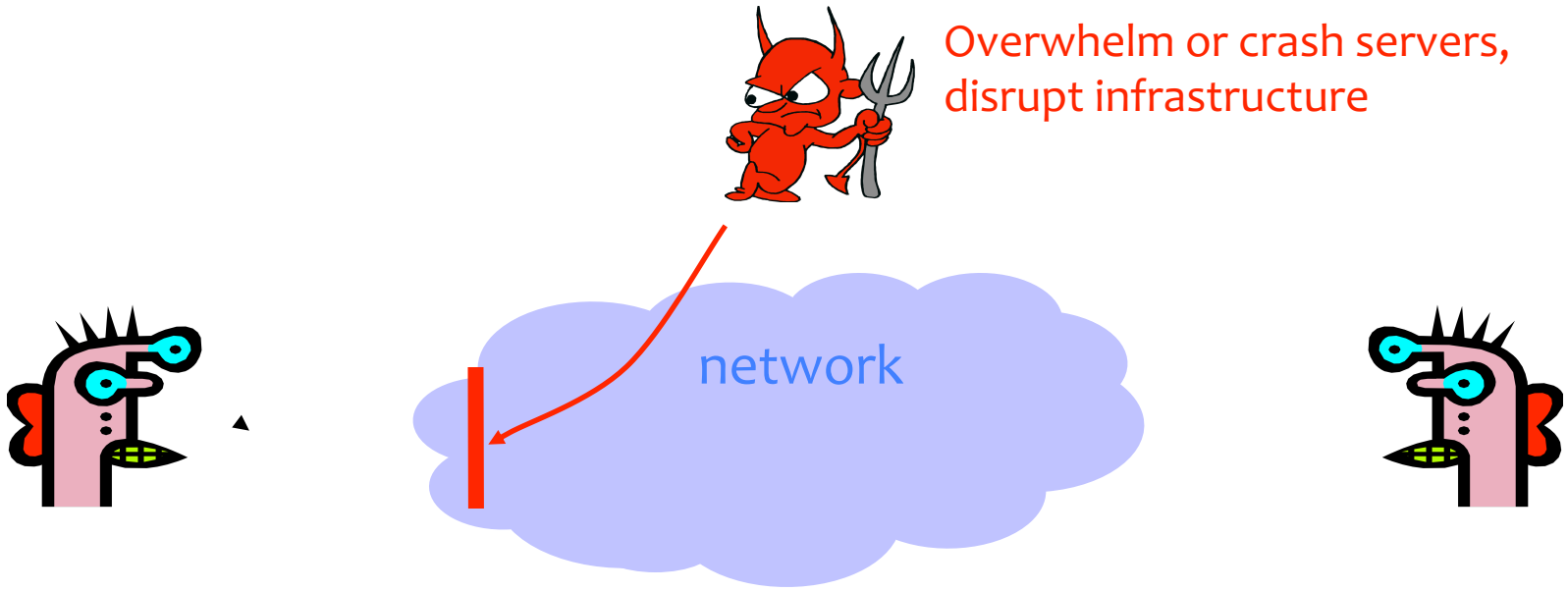
Integrity

- Integrity is prevention of unauthorized changes.



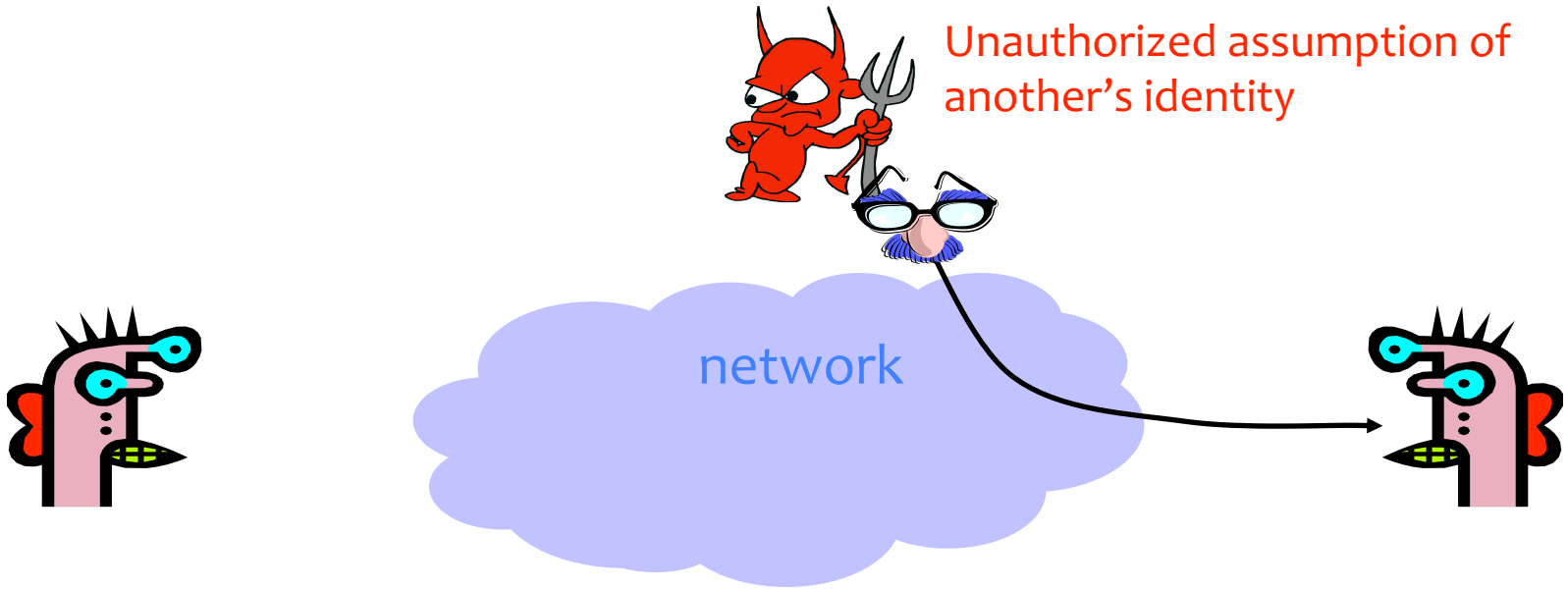
Availability

- Availability is ability to use information or resources.



Authenticity

- Authenticity is knowing who you're talking to.

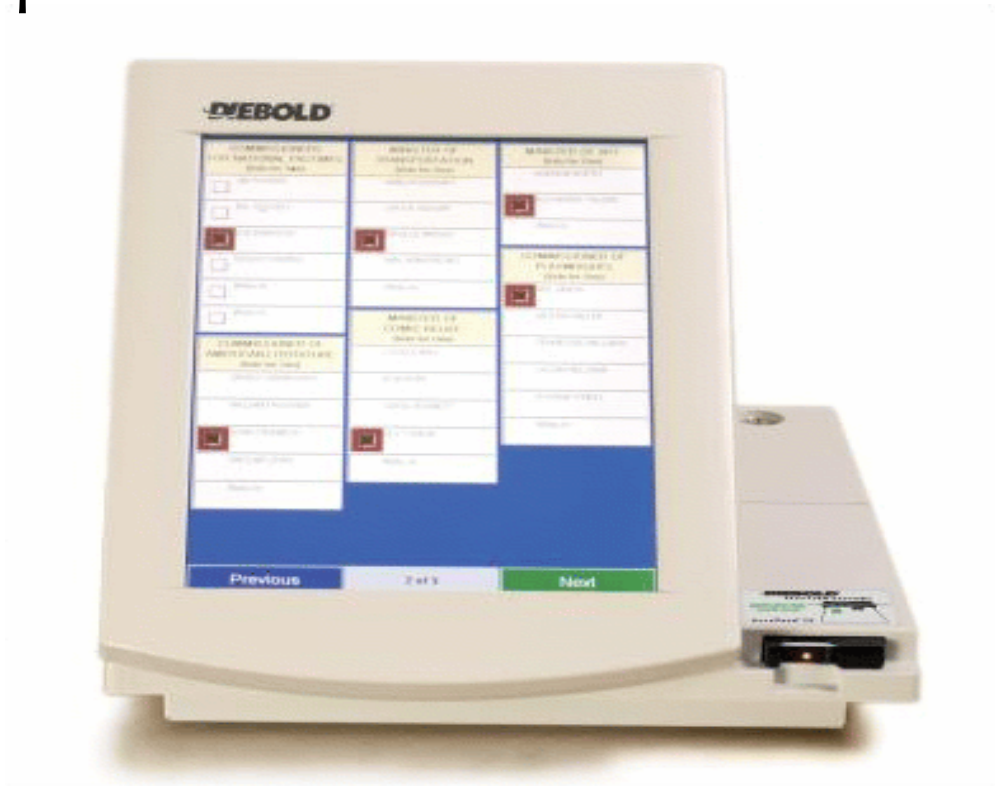


Threat Modeling

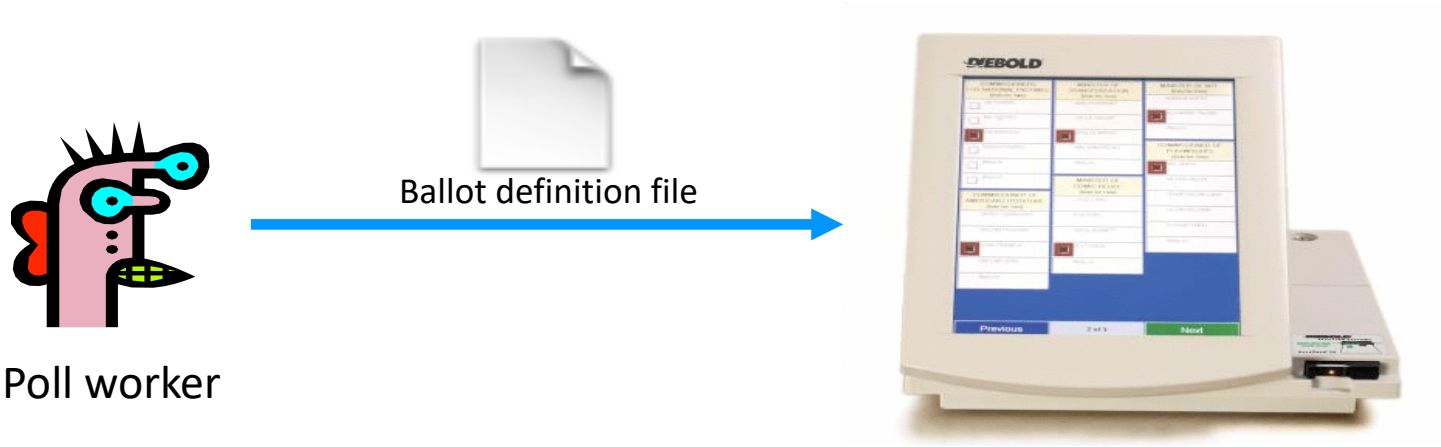
- There's no such thing as perfect security
  - But, attackers have limited resources
  - **Make them pay unacceptable costs / take on unacceptable risks to succeed!**
- Defining security per context: identify assets, adversaries, motivations, threats, vulnerabilities, risk, possible defenses

Threat Modeling Example: Electronic Voting

- Popular replacement to traditional paper ballots

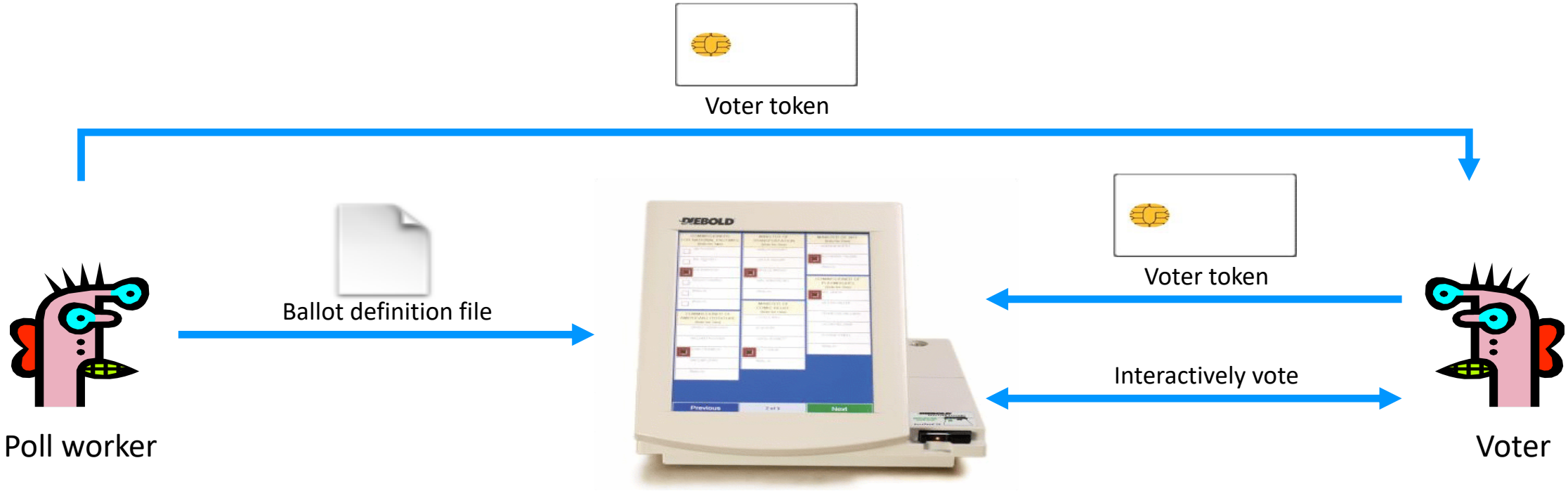


Pre-Election



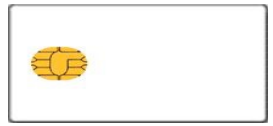
Pre-election: Poll workers load "ballot definition files" on voting machine.

Active Voting



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

Active Voting



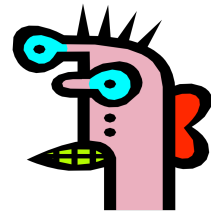
Voter token

Ballot definition file

Poll worker

Voter token

Interactively vote

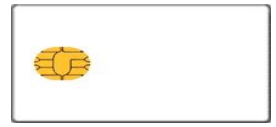


Voter

Encrypted votes

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

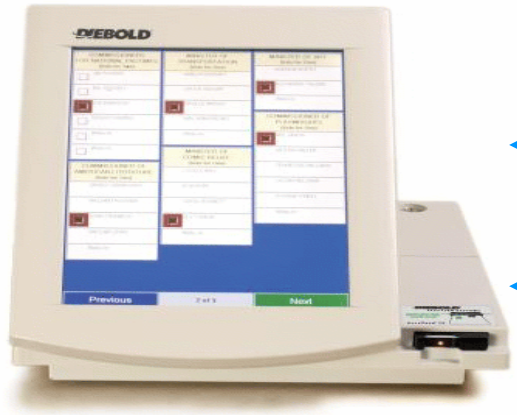
Post-Election



Voter token

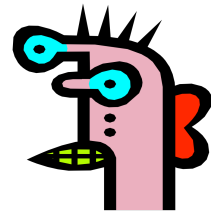
Ballot definition file

Poll worker



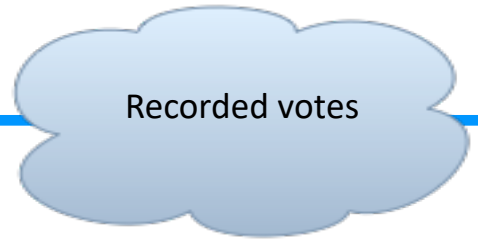
Voter token

Interactively vote



Voter

Encrypted votes



Tabulator

Post-election: Stored votes transported to tabulation center.

Aside: In-Class Participation

- Trying to combine the best of online and in-person
  - In-class discussions, polls, and other online tools
  - More use of the online discussion board
  - Questions live and via pollev
- **Main component: Lightly graded in-class activities**
  - Canvas "quiz" submission (intended for use during class, but can be submitted up until start of next lecture)
  - *Not* a "quiz" in the traditional sense

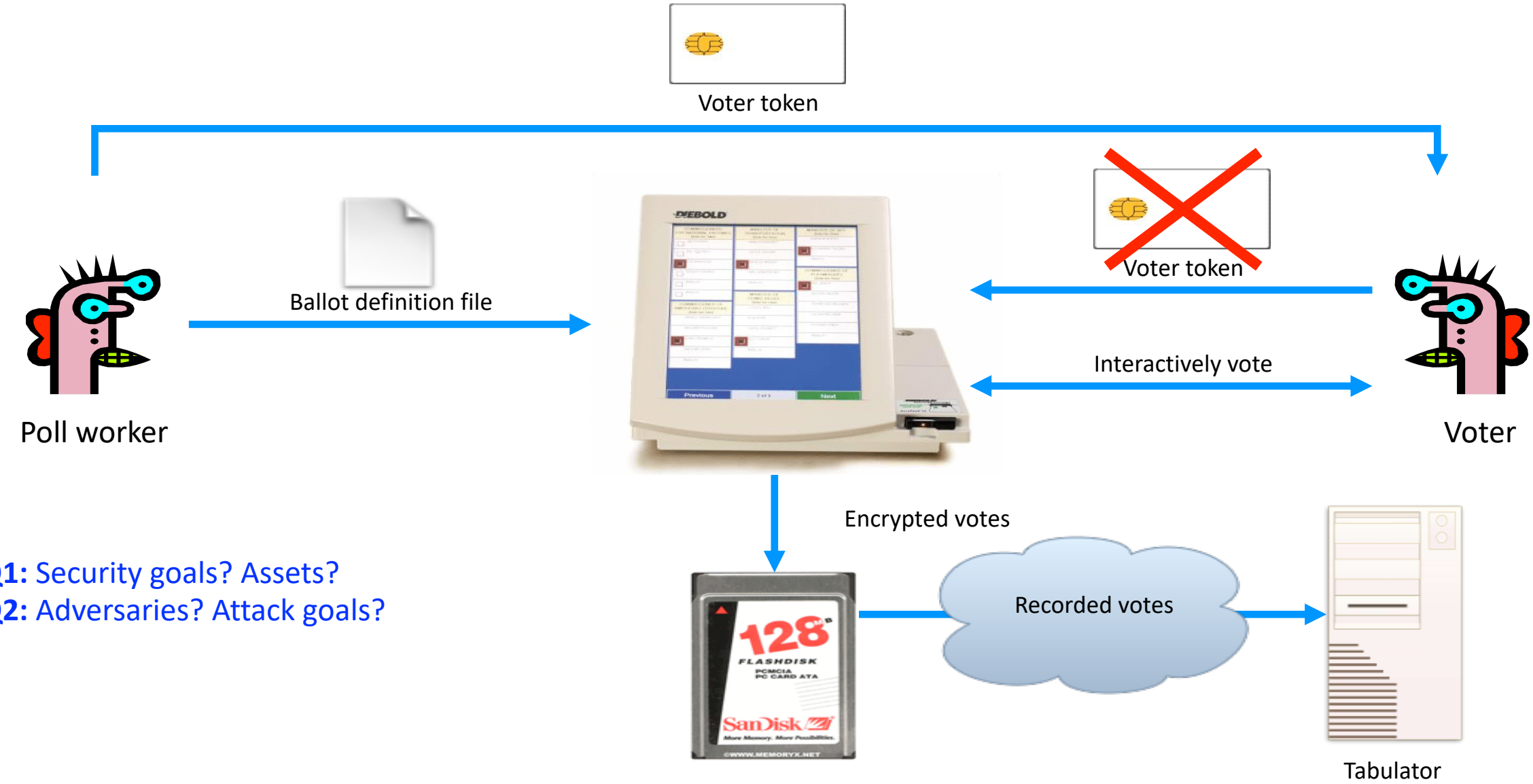
In-Class "Worksheet"

- Go to Canvas -> Quizzes -> "In-Class Activity – Sep 30"
- Fill out the questions while discussing with your neighbor(s)
  - 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
- Q1: What do you think are the **security goals** of the electronic voting system described in class and shown above? What would be some of the **assets** that must be protected?
- Q2: Who are the **adversaries** who might try to attack this electronic voting system? What might be the **attacker's goals**? What potential **threats or vulnerabilities** do you see?

Security and E-Voting (Simplified)

- Functionality goals:
  - Easy to use, reduce mistakes/confusion, make voting more accessible
- Security goals:

Can You Spot Any Potential Issues?



Q1: Security goals? Assets?
Q2: Adversaries? Attack goals?

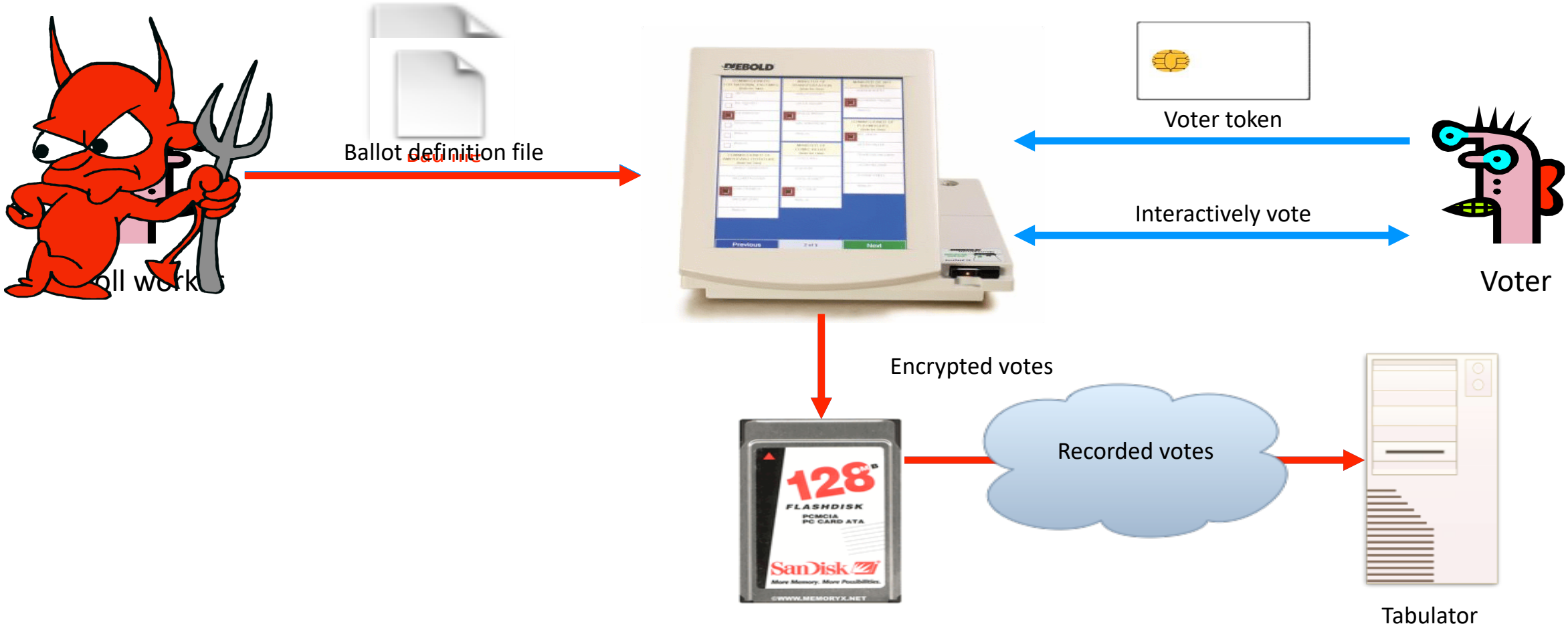
What Software is Running?

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 they 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.

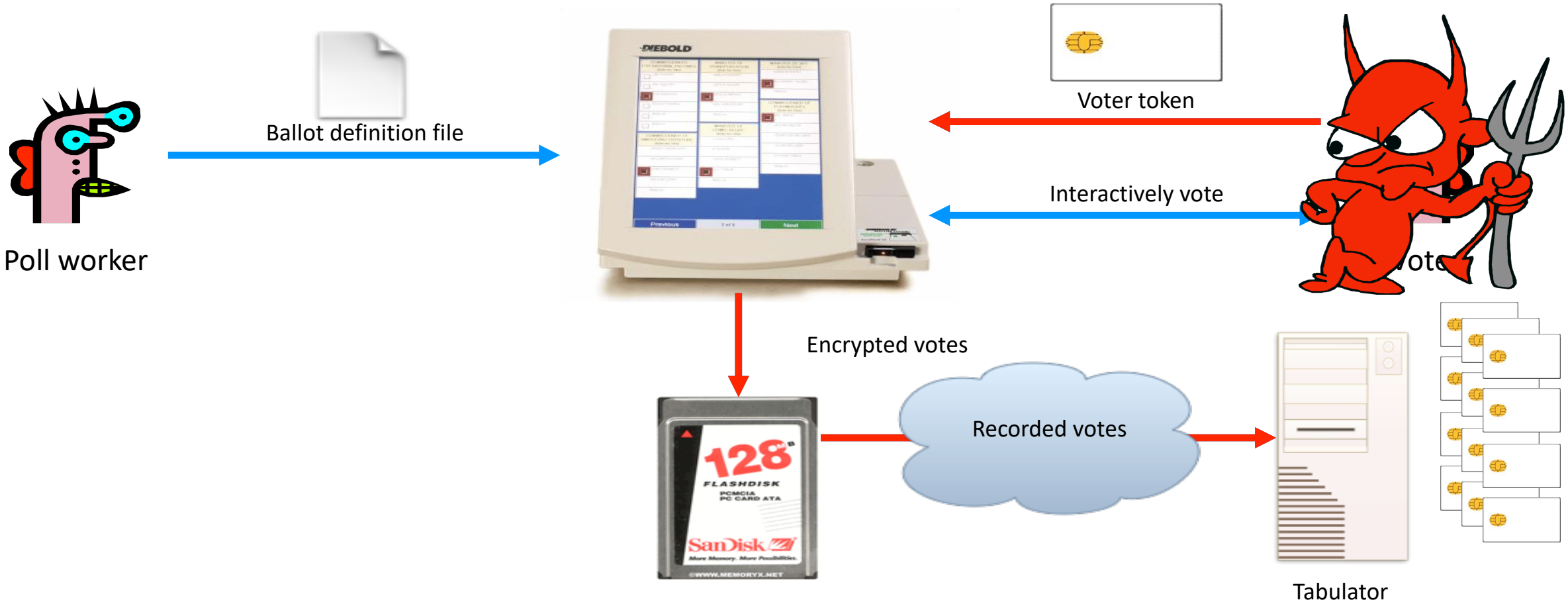
Problem: Ballot definition files are not authenticated.

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



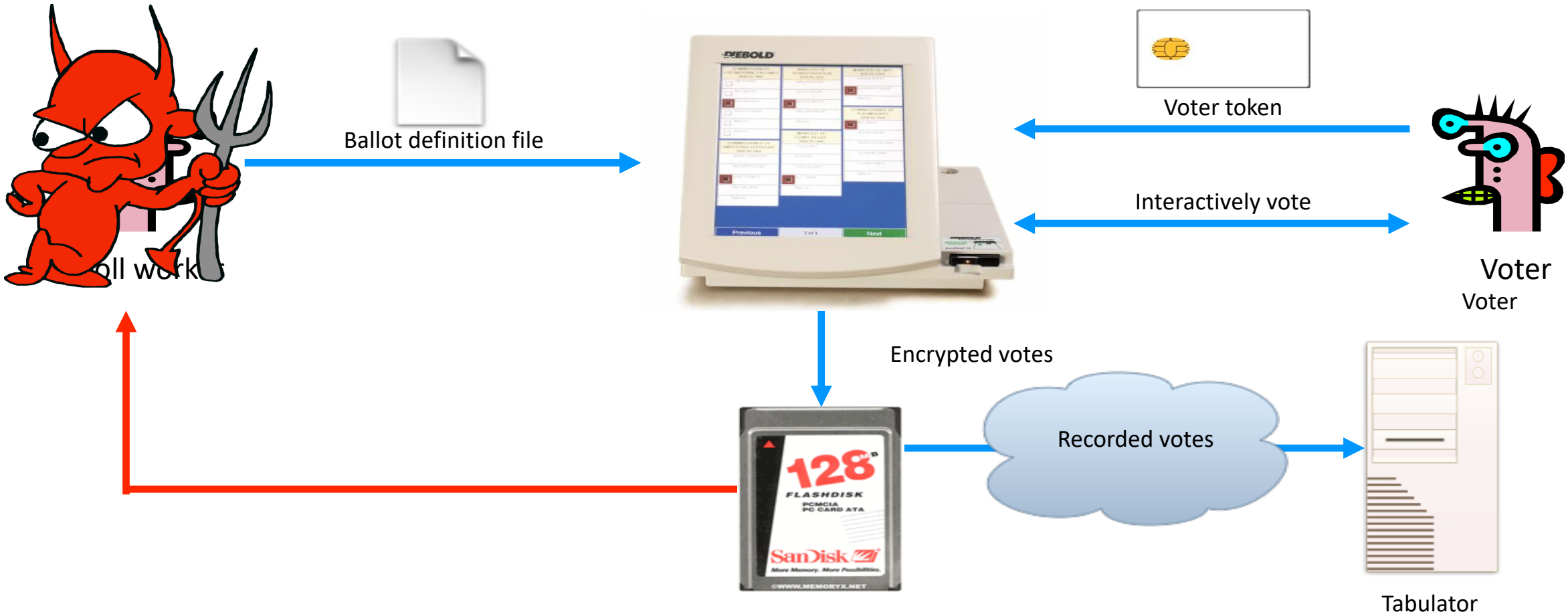
Problem: Smartcards can perform cryptographic operations. But there is no authentication from voter token to terminal.

Example attack: A regular voter could make their 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.