CSE 484 / CSE M 584: Computer Security and Privacy

Winter 2021

David Kohlbrenner

dkohlbre@cs.washington.edu

Thanks to Franzi Roesner, 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

- Things Due:
  - Ethics form: Due next Monday (1/11)
  - Homework #1: Due next Wednesday (1/13)

- Textbook:
  - Not available digitally, apologies
  - I'll be posting alternative readings as well that are freely available

- Any logistics questions at this point?

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

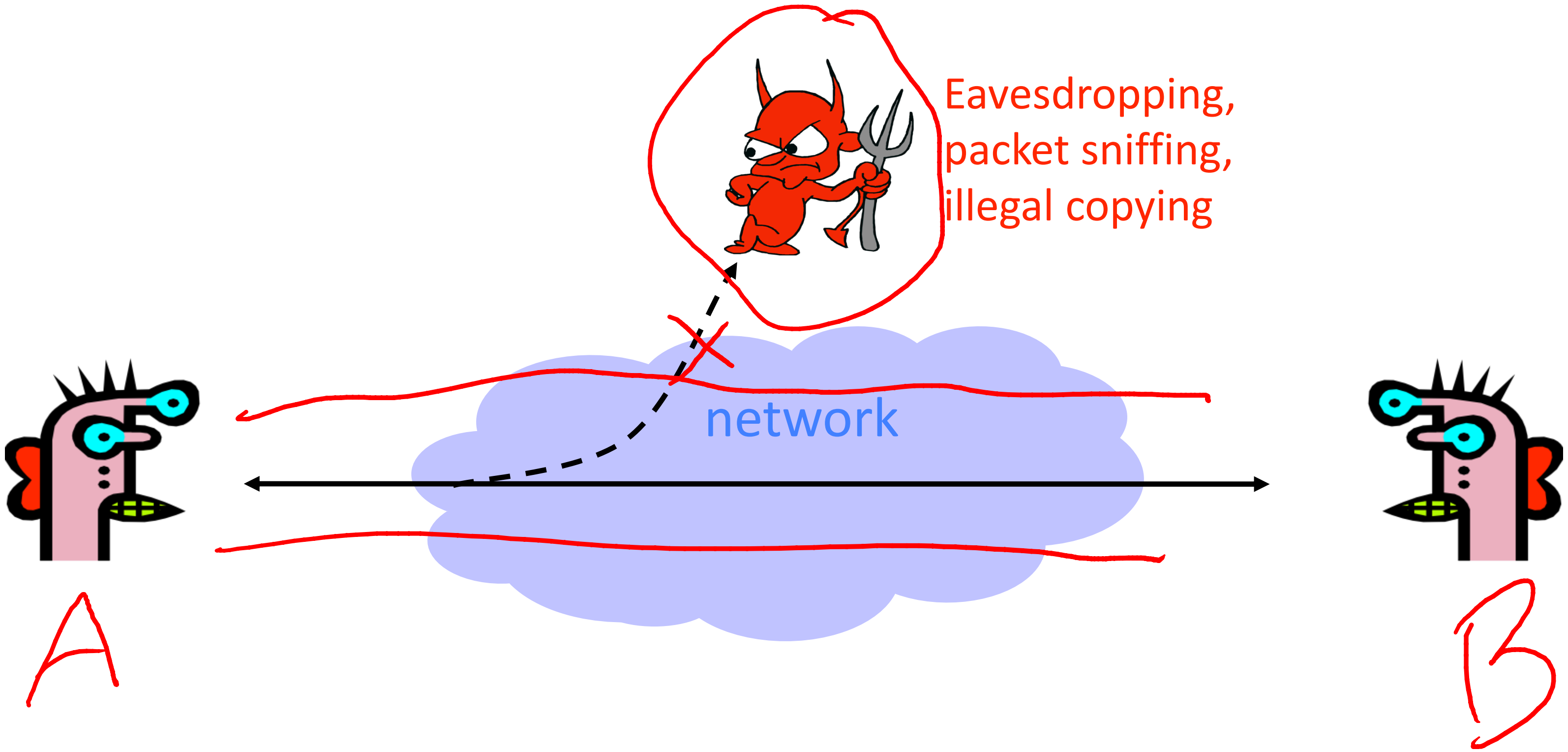
What’s *Security*, Anyway?

- Common general security goals: “CIA”
  - Confidentiality
  - Integrity
  - Availability

- Or the extension: C~~P~~c IAAU (Parkerian Hexad)
  - ~~Control~~
  - Authenticity
  - ~~Utility~~

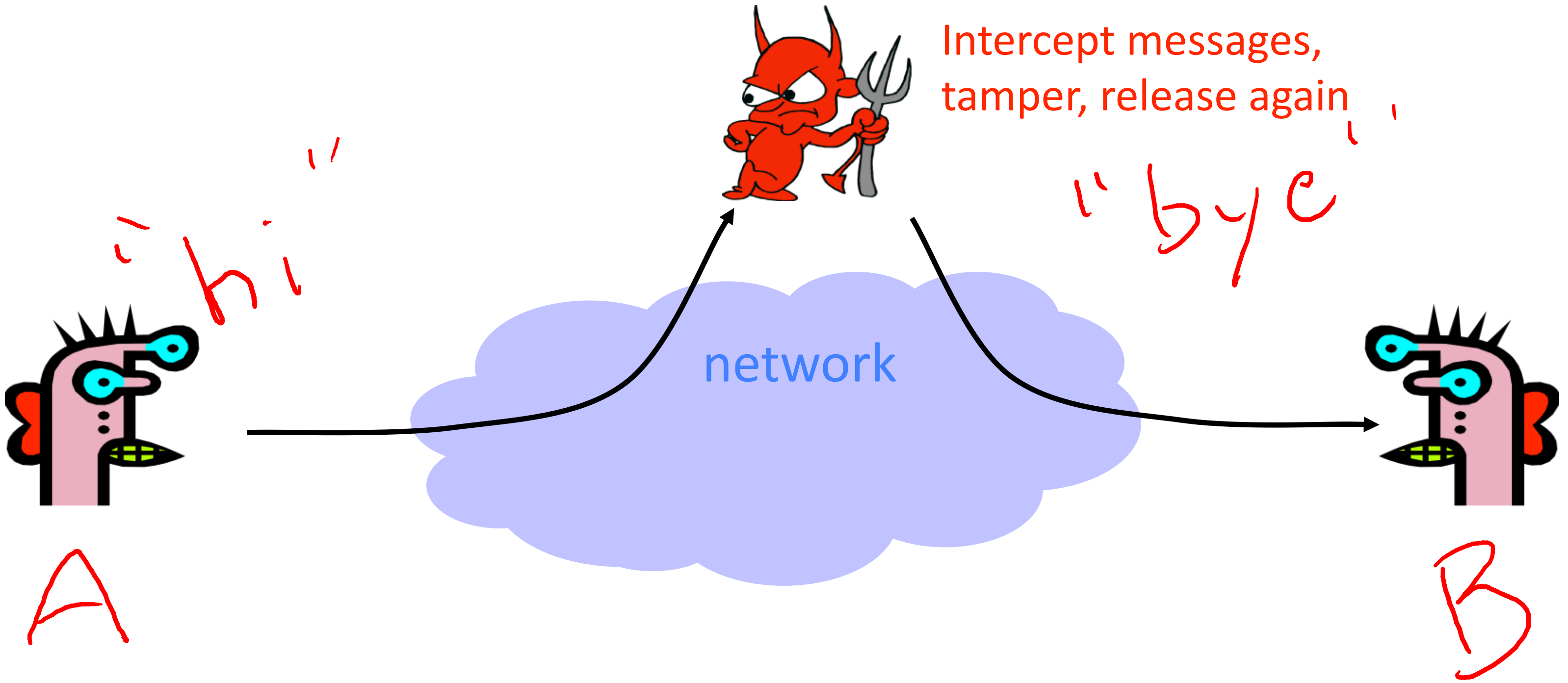
Confidentiality (Privacy)

- Confidentiality is concealment of information.



Integrity

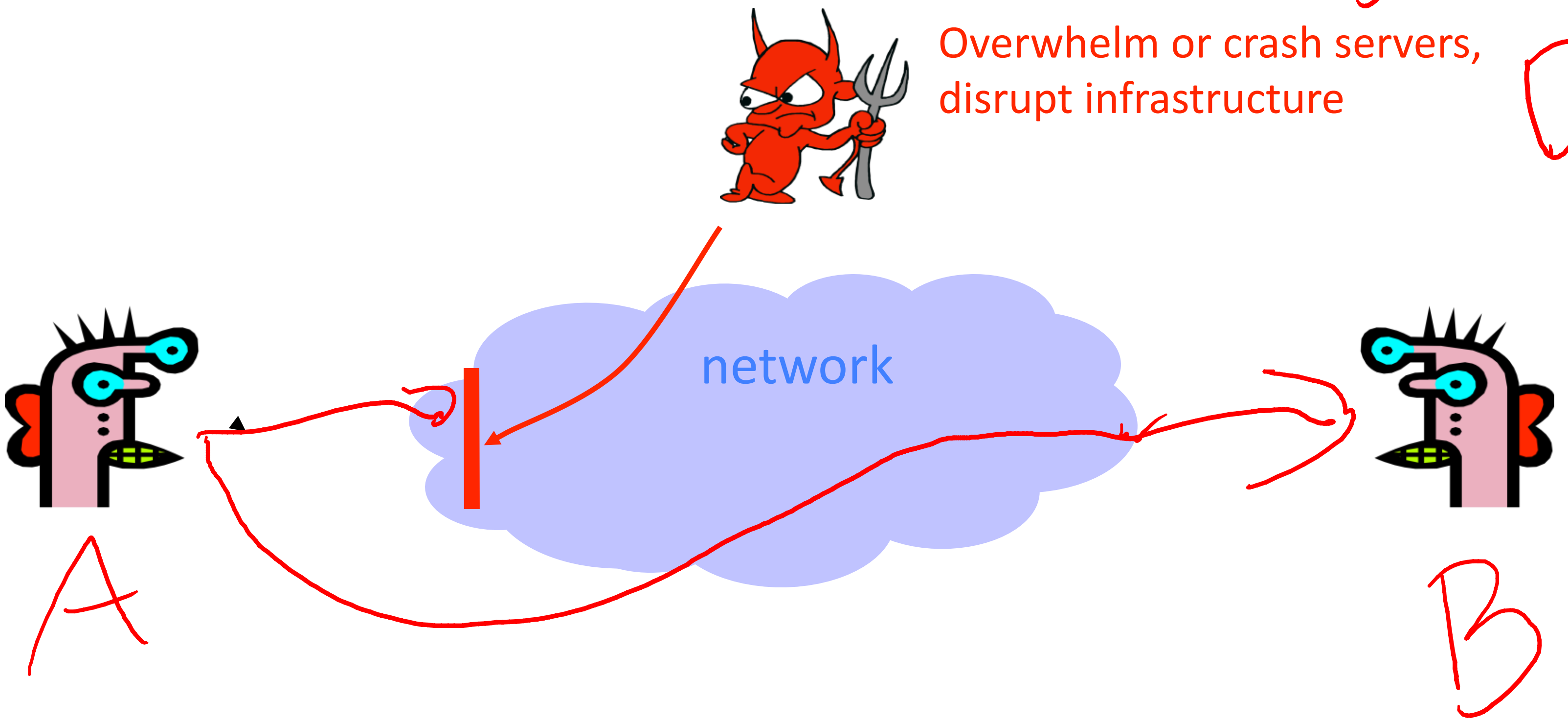
- Integrity is prevention of unauthorized changes.



Availability

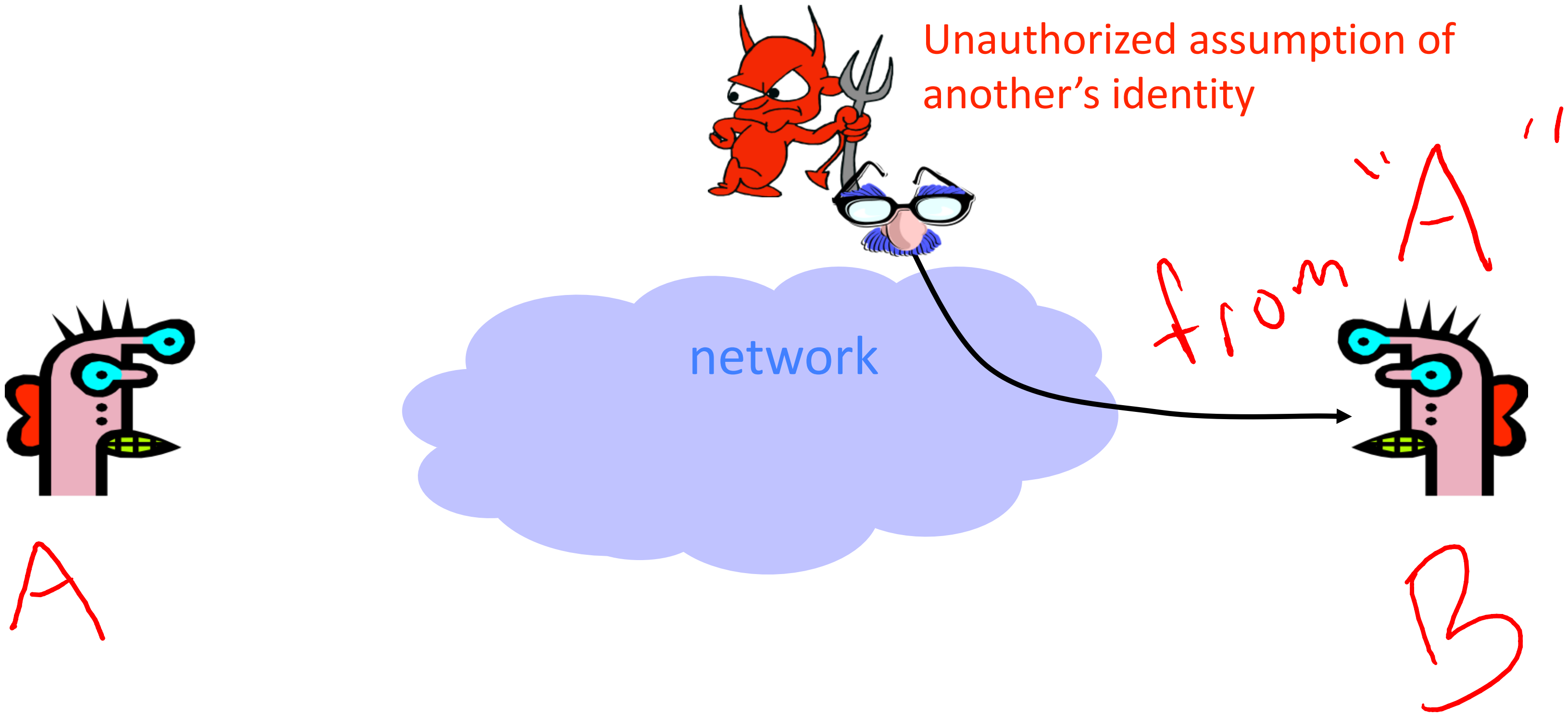
denial of service
DDOS

- Availability is ability to use information or resources.



Authenticity

- Authenticity is knowing who you're talking to.

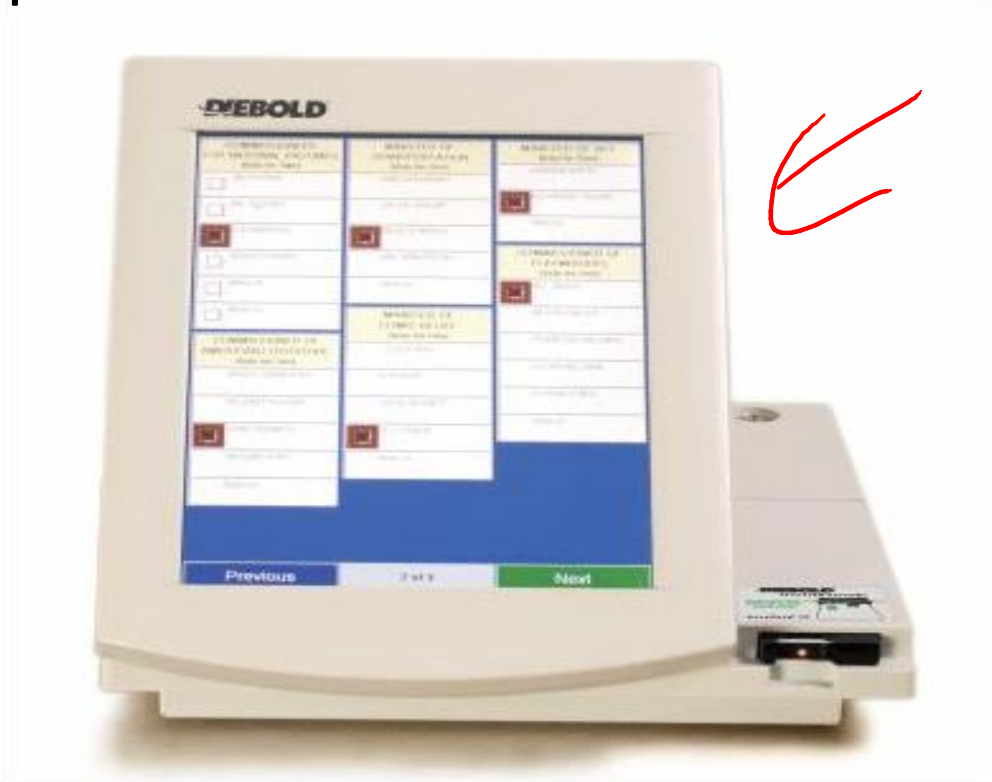


Threat Modeling

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

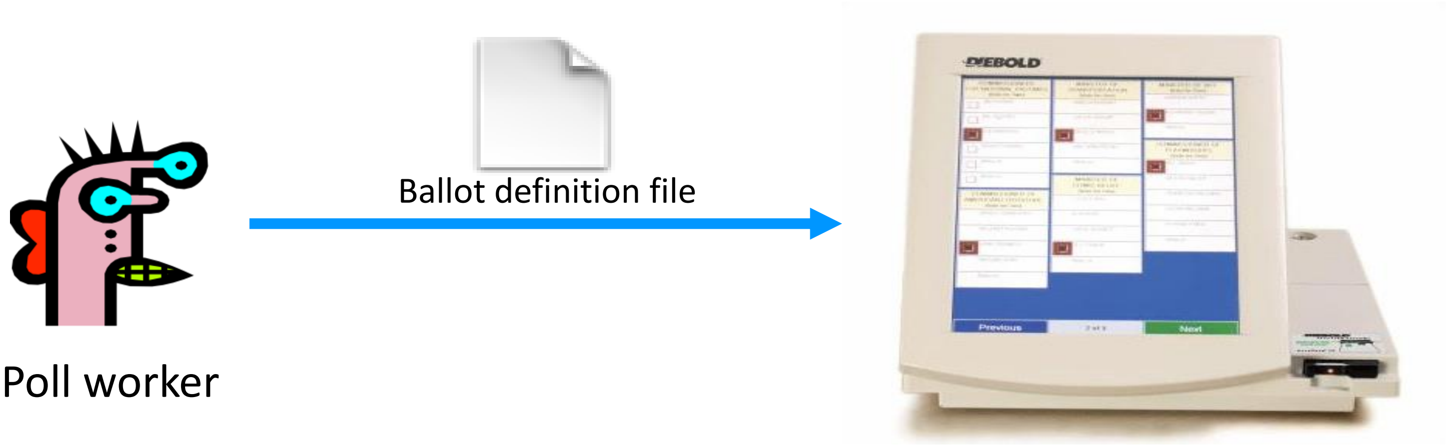
Threat Modeling Example: Electronic Voting

- Popular replacement to traditional paper ballots



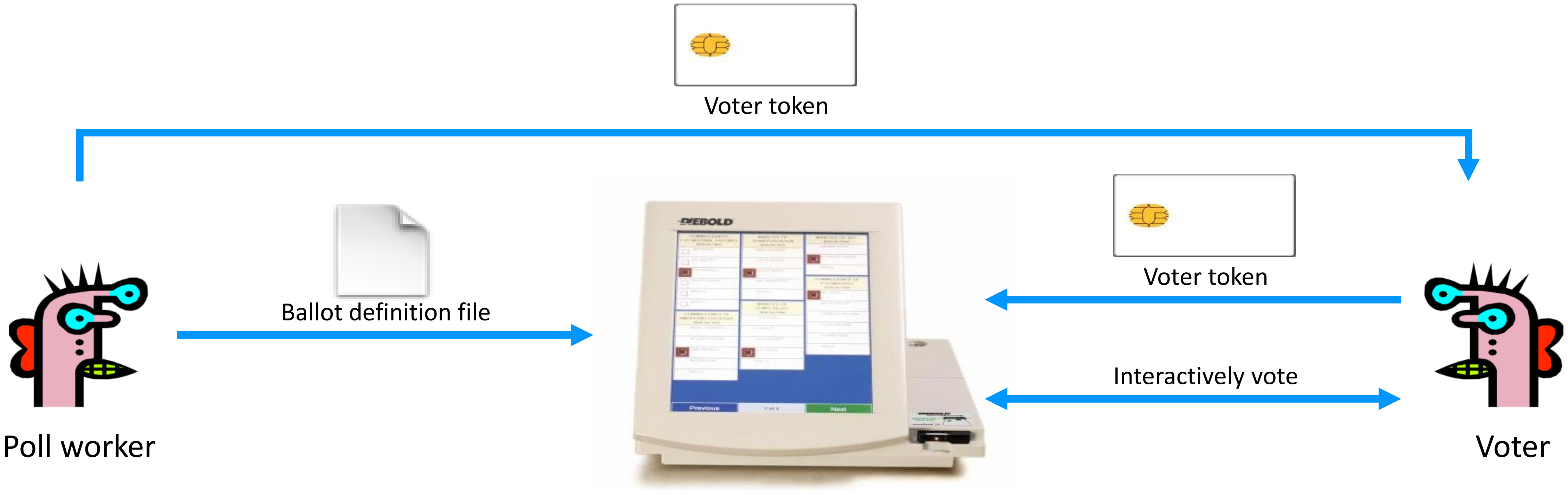
2000

Pre-Election



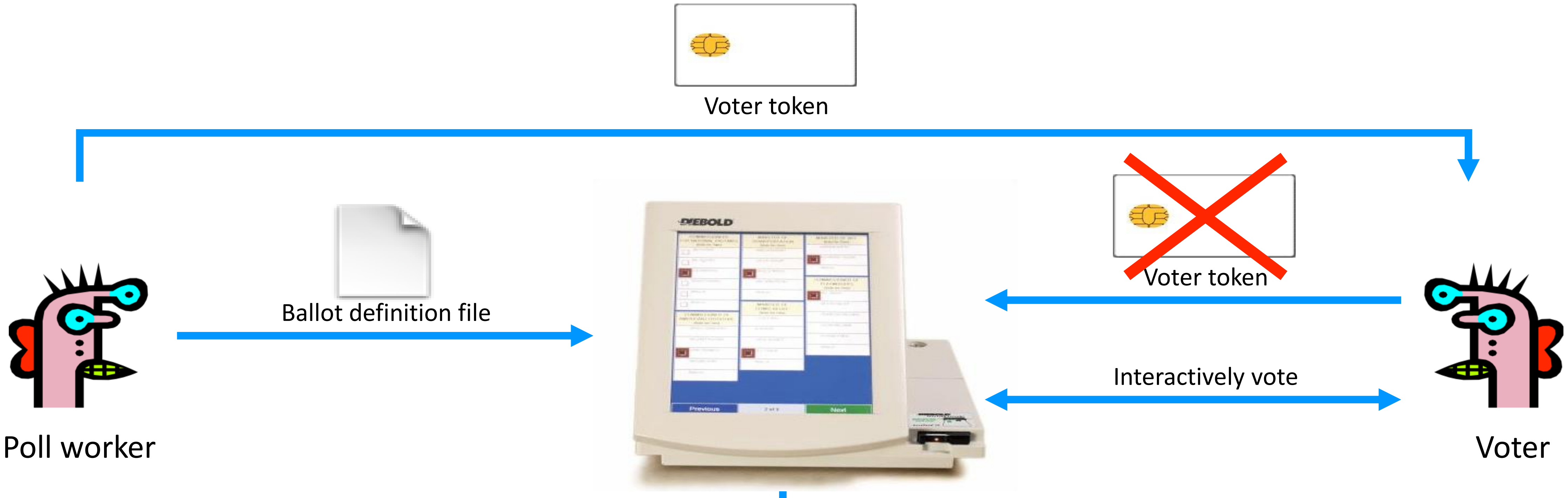
Pre-election: Poll workers load “ballot definition files” on voting machine.

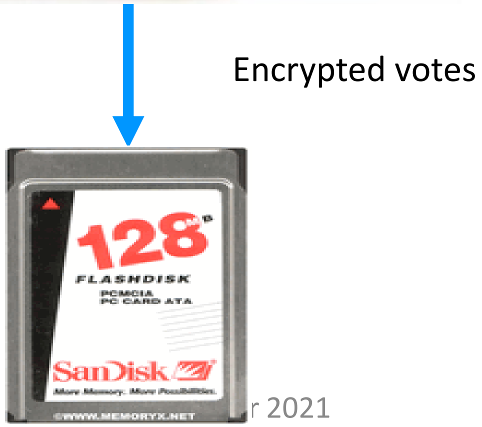
Active Voting



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

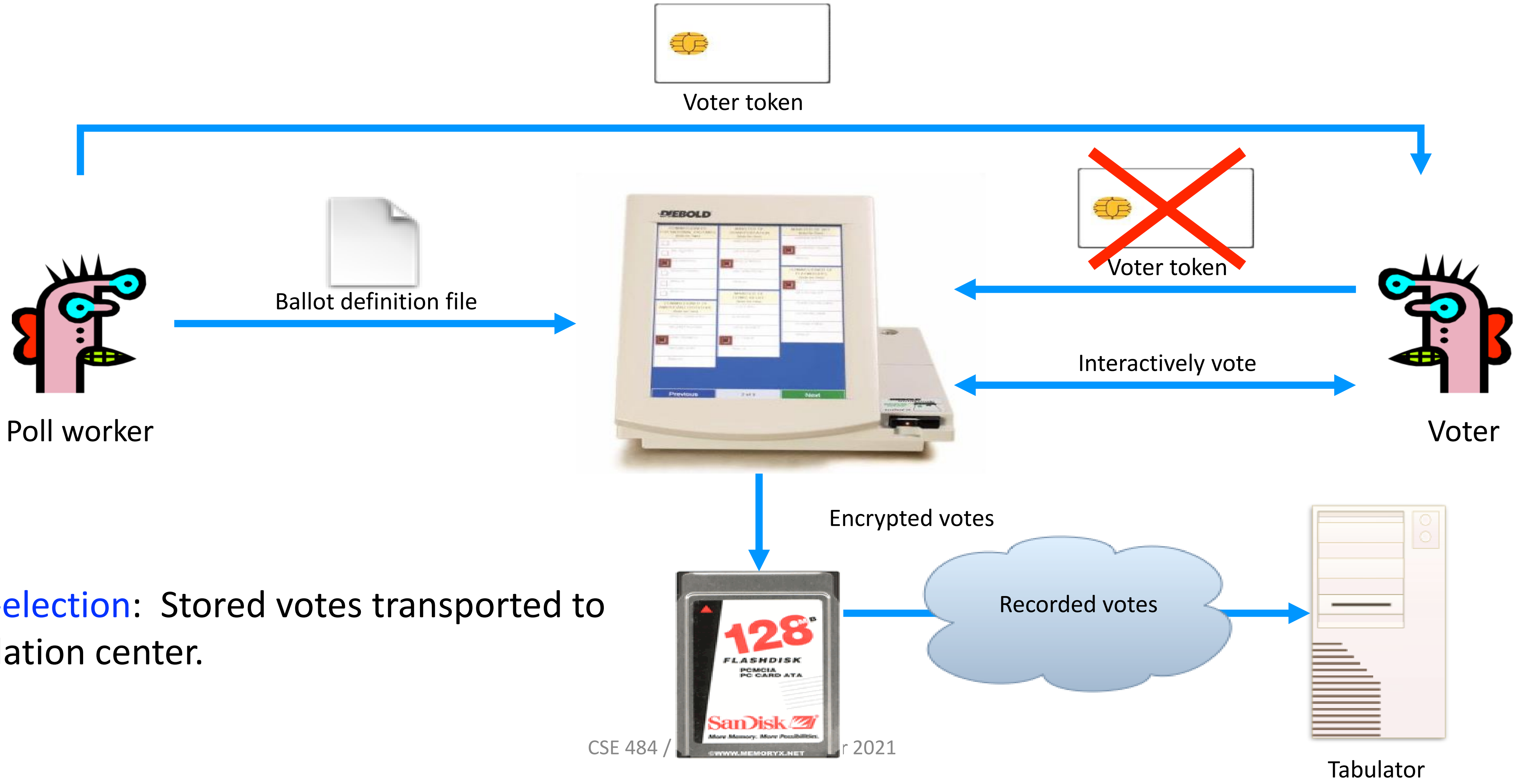
Active Voting





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

Post-Election



Post-election: Stored votes transported to tabulation center.

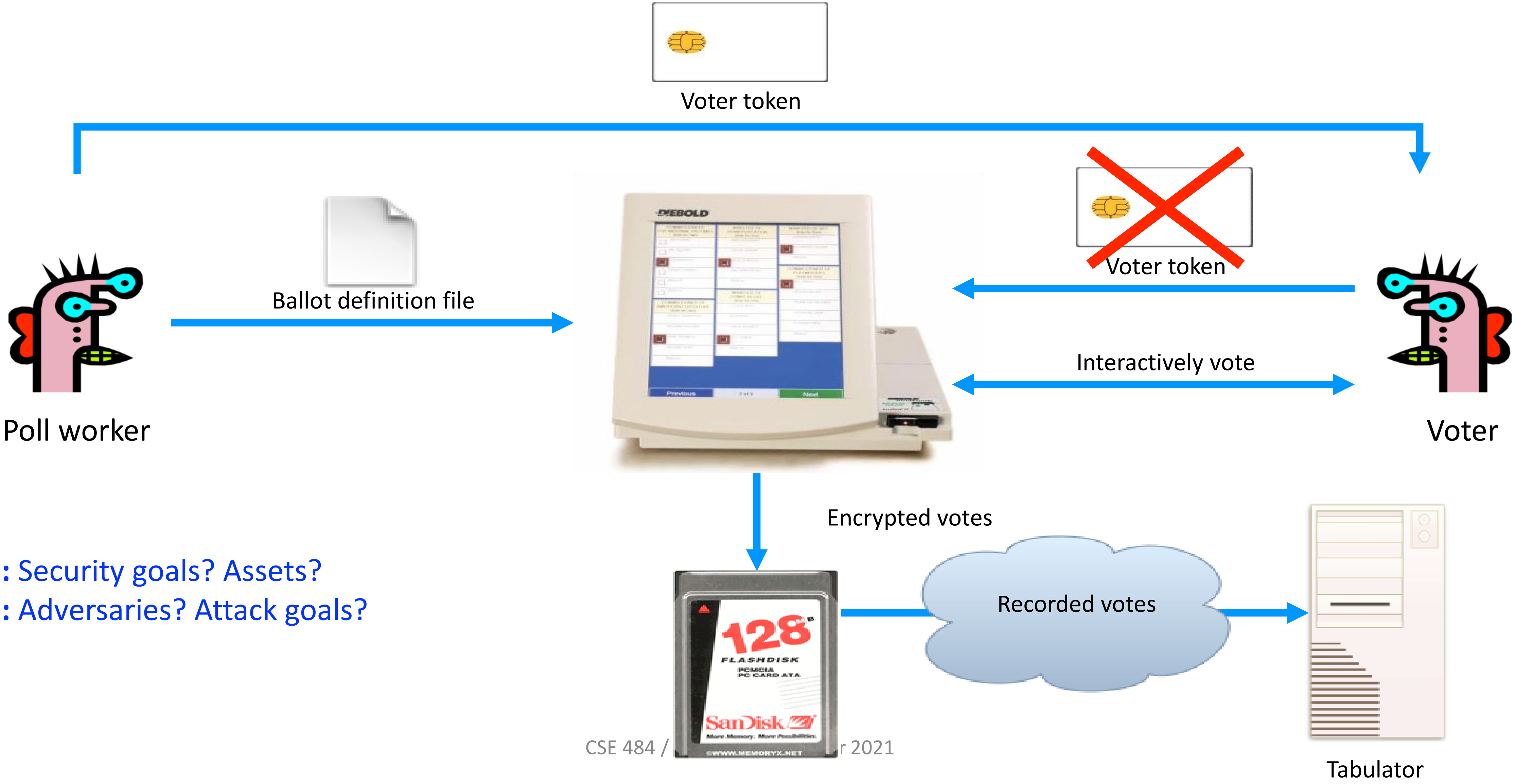
In-Class “Worksheet” Experiment

- Go to Canvas -> Quizzes -> “In-Class Activity – Jan 5”

    (I will also always post the link in the chat.)

- 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?



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

Security and E-Voting (Simplified)

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

adversary cannot change outcome
ensure single vote (counted 1x)
privacy of vote
physical ~~vote~~ storage
availability of vote

Potential Adversaries

rationales
- trolls
- foreign govts
- political opponents

actors
- poll workers / officials
- voters
- engineers (sw / hw / fabrication)

Lets talk about concrete problems

What Software is Running?

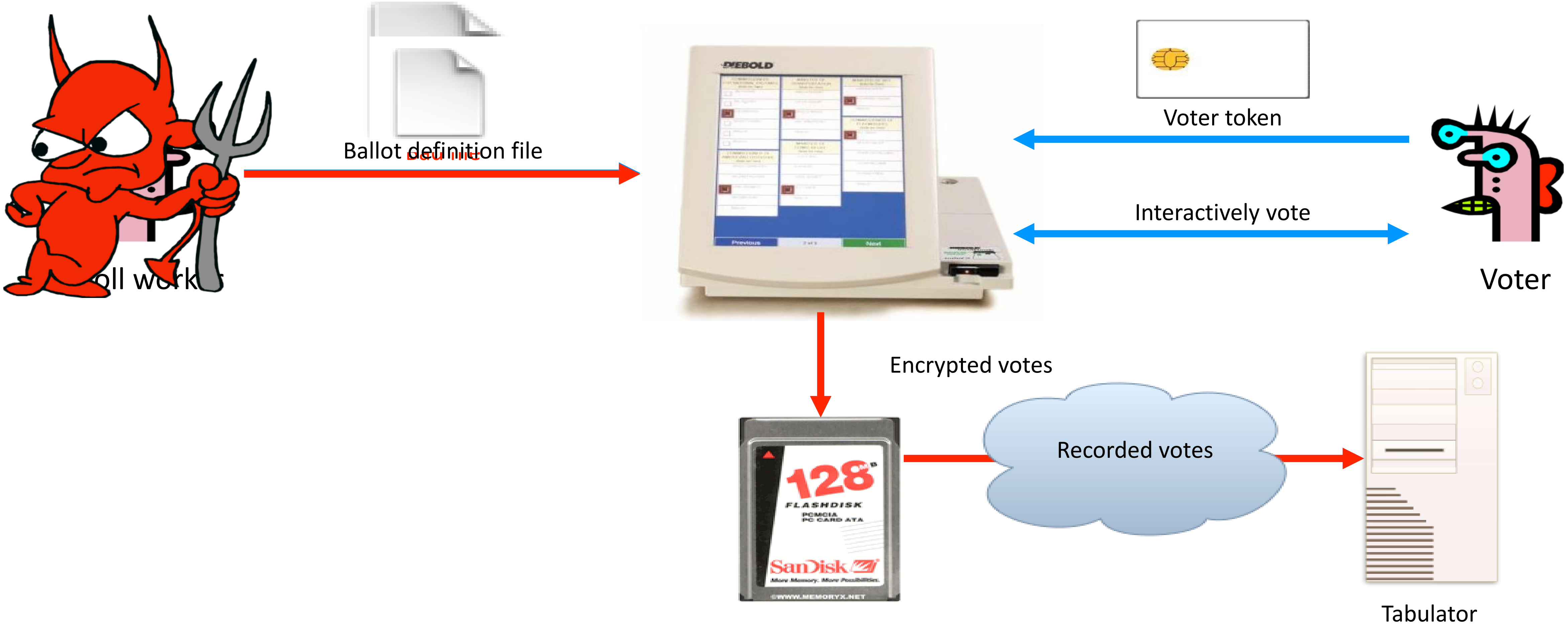
Problem: An adversary (e.g., a poll worker, software developer, or company representative) able to control the software or the underlying hardware could do whatever they wanted.

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.

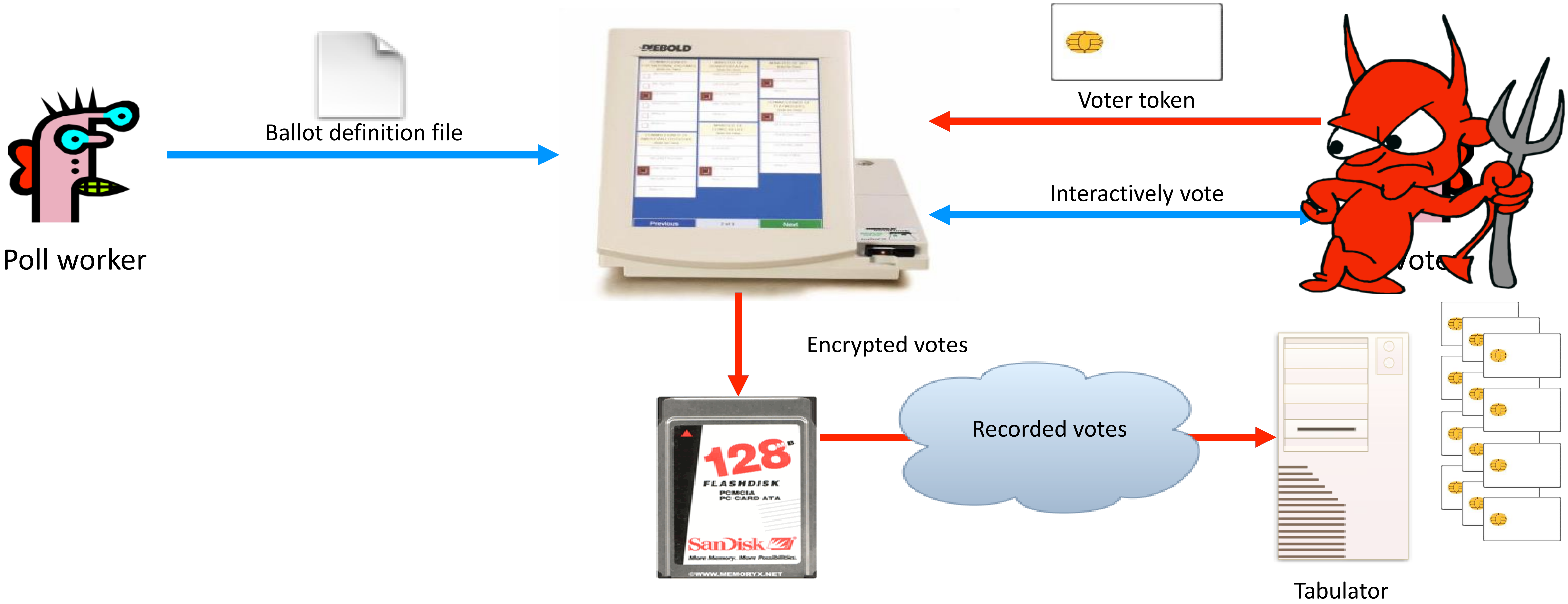
Problem: Ballot definition files are not authenticated.

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



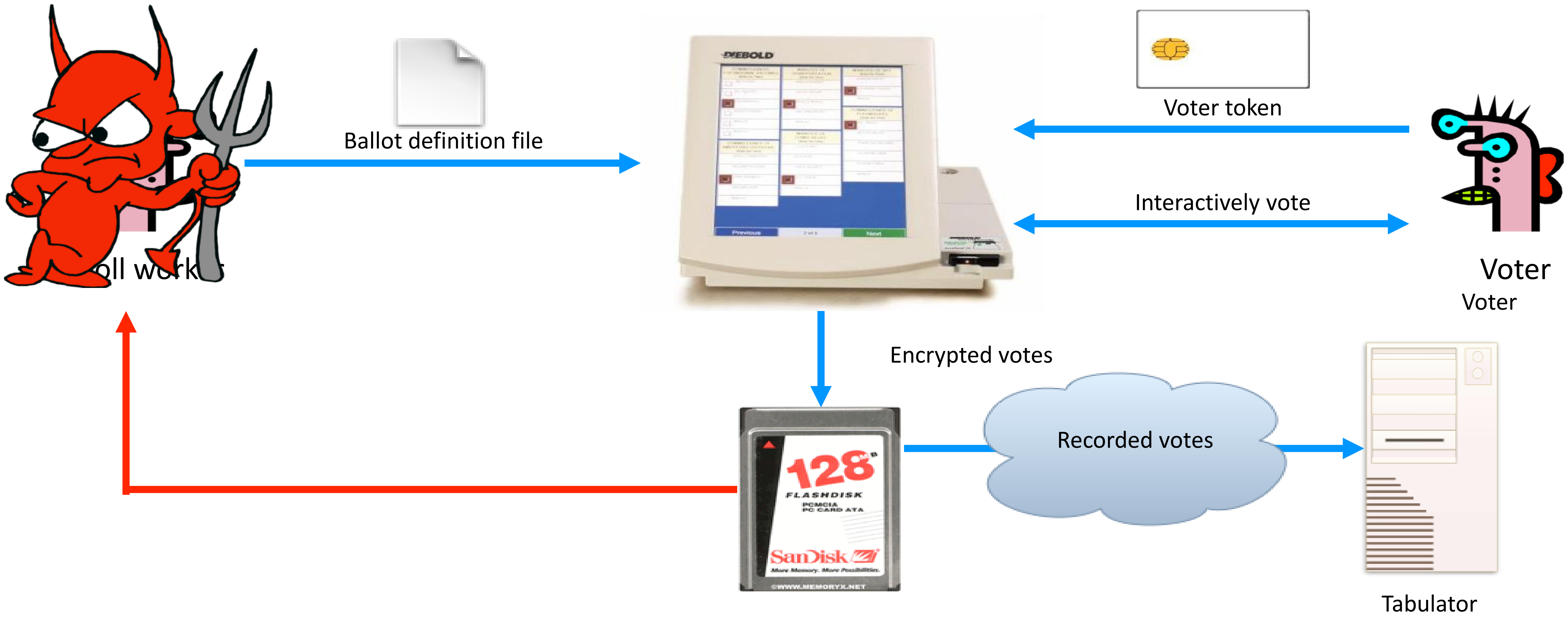
Problem: Smartcards can perform cryptographic operations. But there is no authentication from voter token to terminal.

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



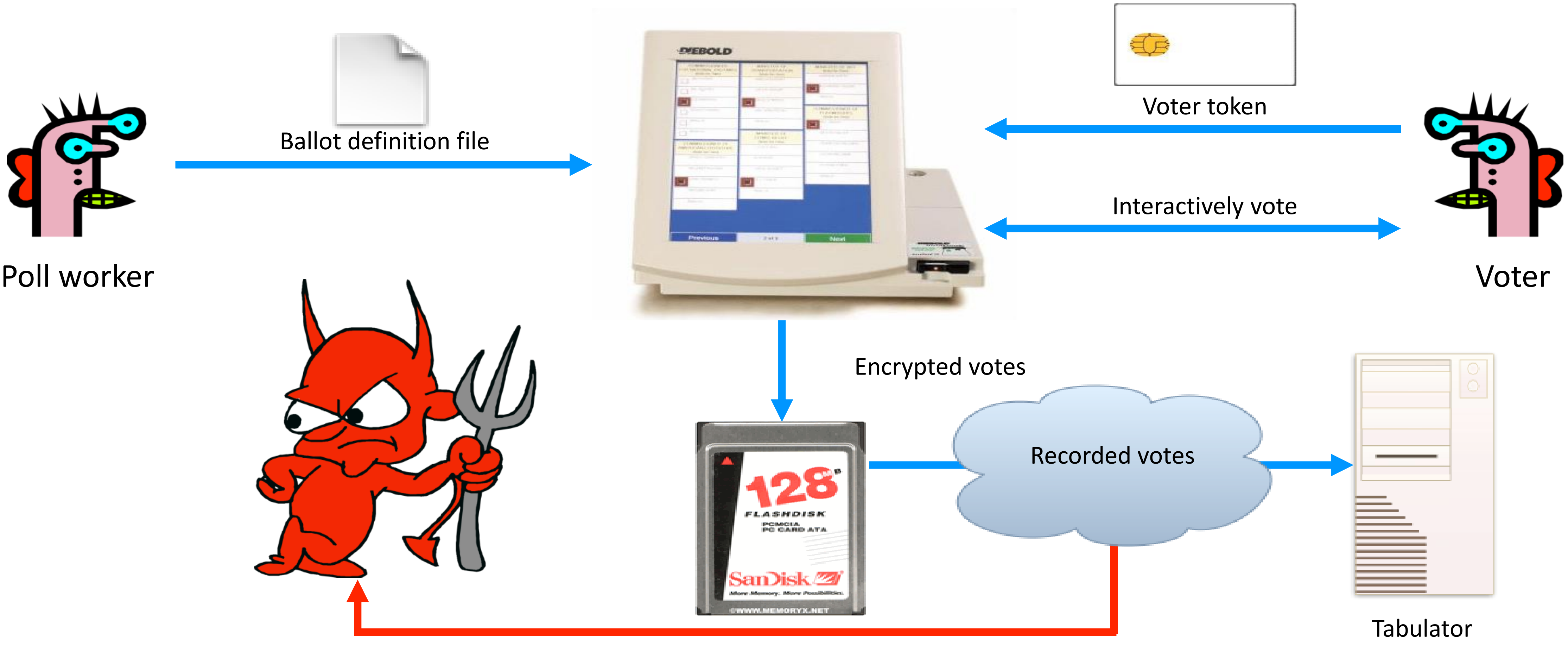
Problem: Encryption key ("F2654hD4") hard-coded into the software since (at least) 1998. Votes stored in the order cast.

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.



TOWARDS DEFENSES

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

Whole System is Critical

- Securing a system involves a whole-system view
  - Cryptography
  - Implementation
  - People
  - Physical security
  - Everything in between

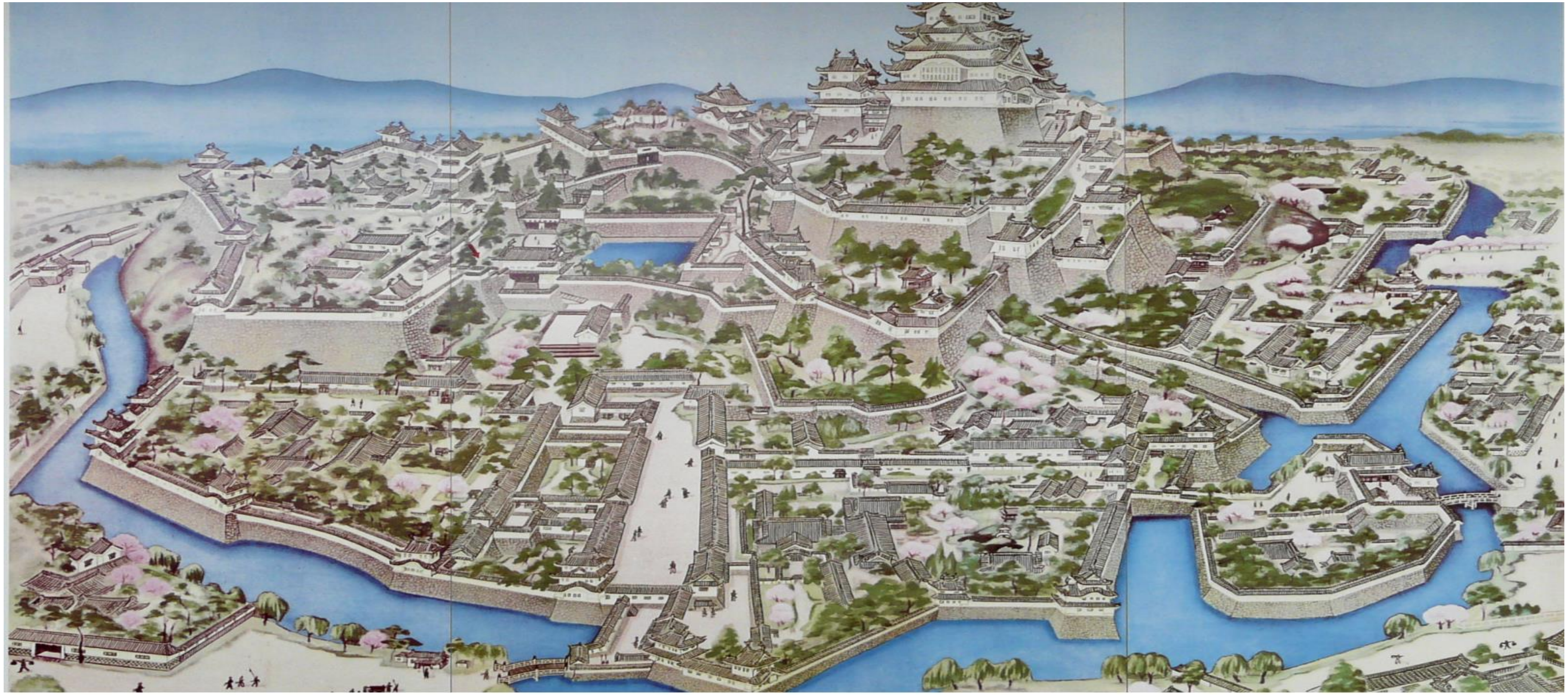
- This is because “security is only as strong as the weakest link,” and security can fail in many places
  - No reason to attack the strongest part of a system if you can walk right around it.

Whole System is Critical

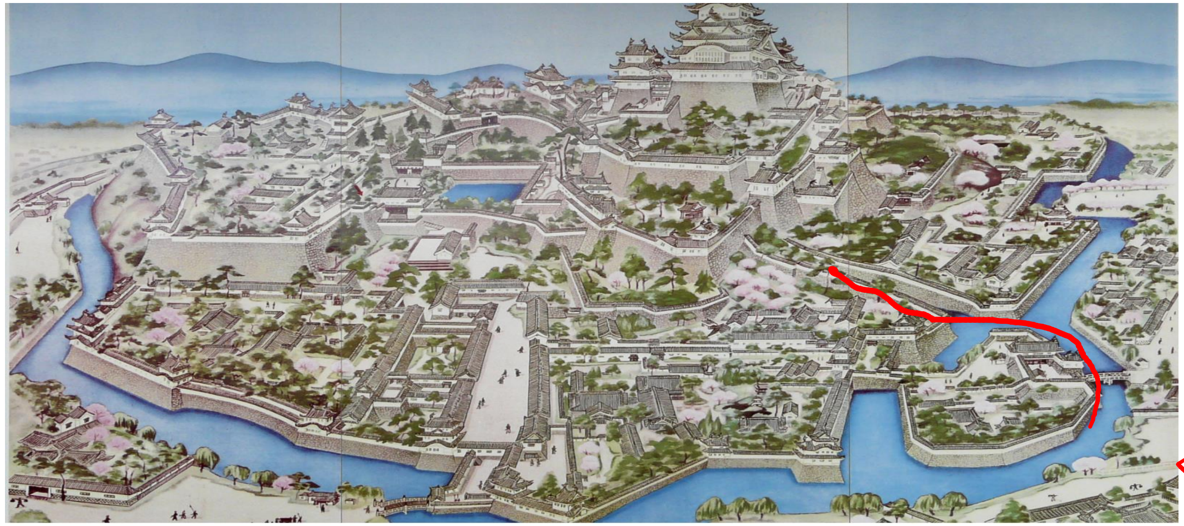
- Securing a system involves a whole-s
  - Cryptography
  - Implementation
  - People
  - Physical security
  - Everything in between
- This is because “security is only as strong as the weakest link,” and security can fail in many places
  - No reason to attack the strongest part of a system if you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

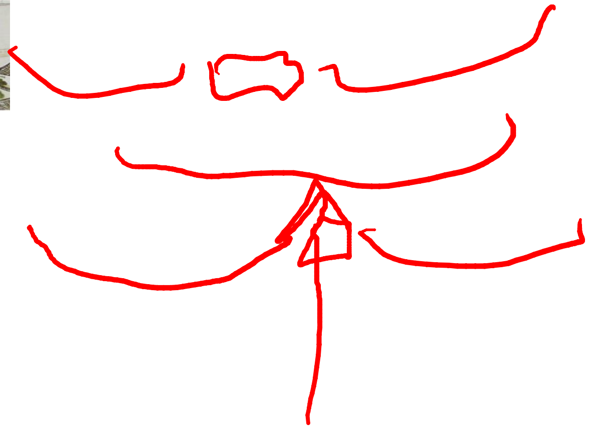
Whole System is Critical

Attacker’s Asymmetric Advantage



Attacker's Asymmetric Advantage





- Attacker only needs to win in one place
- Defender's response: Defense in depth

From Policy to Implementation

- After you've figured out what security means to your application, there are still challenges:
  - Requirements bugs
    - Incorrect or problematic goals
  - Design bugs
    - Poor use of cryptography
    - Poor sources of randomness
    - …
  - Implementation bugs
    - Buffer overflow attacks Classic
    - …
  - Is the system **usable**?

Many Participants

- Many parties involved
  - System developers
  - Companies deploying the system
  - The end users get work done
  - 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 features or security?)

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