Computer Security and Privacy

AIMS Senegal
Winter 2016

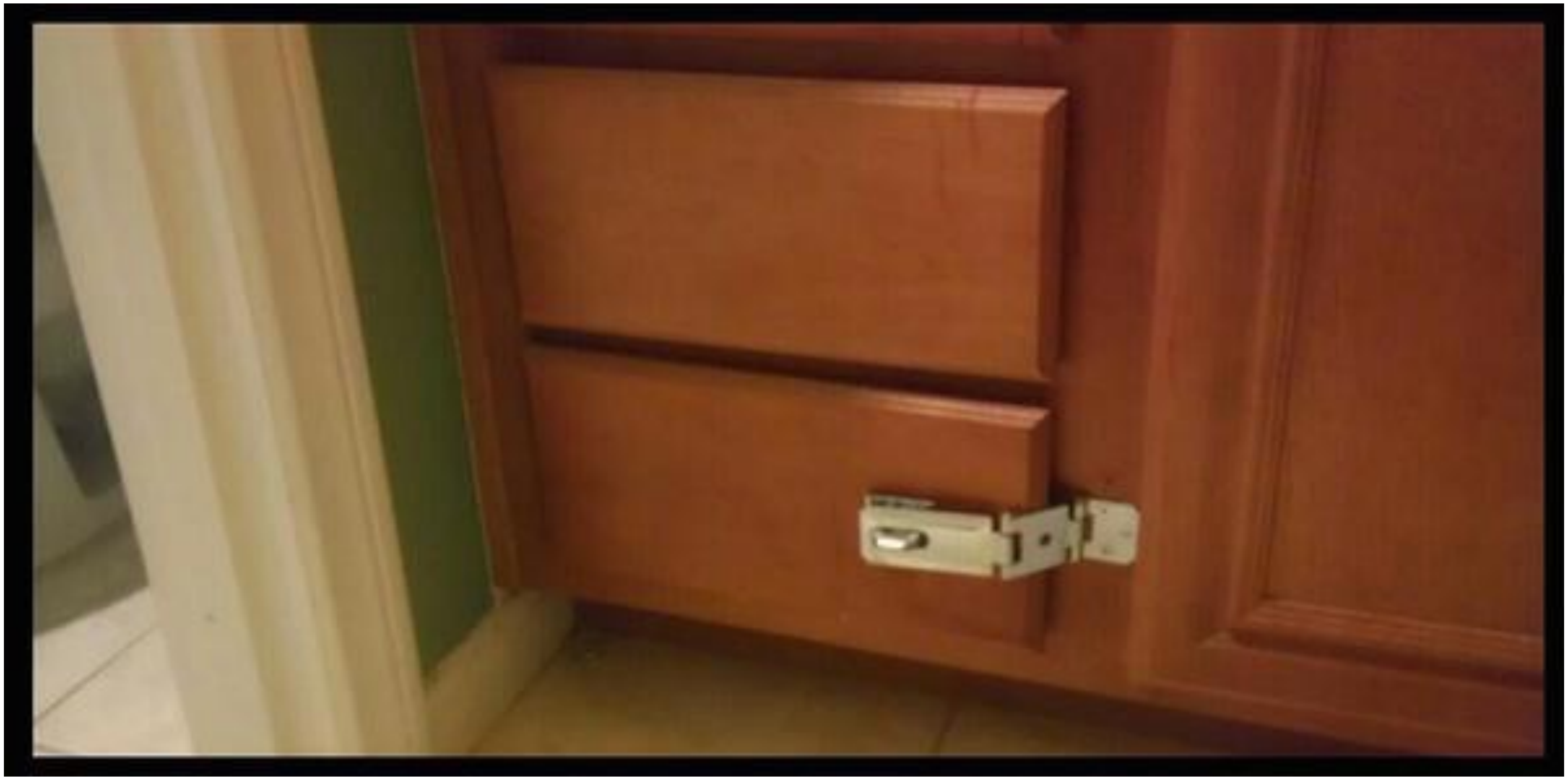
Tara Whalen
Google and the Stanford Center for Internet and Society (USA)
tjwhalen@gmail.com

Computer Security and Privacy

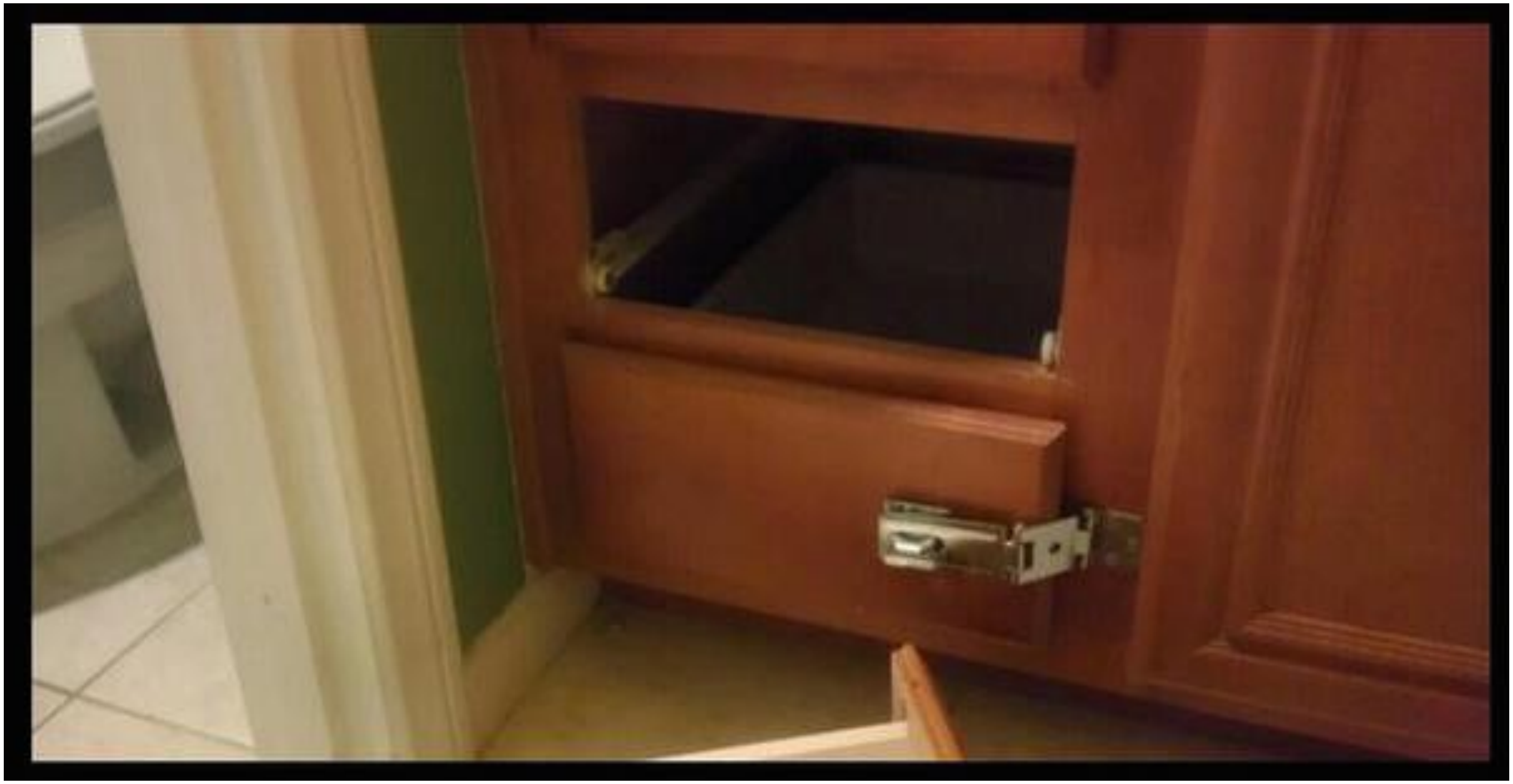
AIMS Senegal
Winter 2016

Topic #1:
Course Introduction

What’s Wrong With This Picture?



What’s Wrong With This Picture?



Useful Information

- Course website: https://sites.google.com/site/secprivaims/
  - Copies of slides, optional reading materials, links to resources (free books)

Prerequisites

I assume no background in computer security

- Necessary: Ability to understand code fragments (no particular languages)
  - Important for software security module in particular: mostly about common concepts
  - But relax! This is not a programming course

Prerequisites

- Helpful: computer networks; operating systems
  - Will help provide deeper understanding of security mechanisms and where they fit in the big picture
- Also helpful: complexity theory; discrete math; algorithms
  - Will help with the more theoretical aspects of this course, especially the cryptography module

Prerequisites

- Most of all: eagerness to learn!
  - You are expected to push yourself to learn as much as possible.
  - You are expected to be a strong, independent learner capable of learning new concepts from the lectures, the readings, and on your own.

Other Helpful Books (Online)

- Ross Anderson, "Security Engineering" (2nd edition)
  - Focuses on design principles for secure systems
  - Wide range of entertaining examples: banking, nuclear command and control, burglar alarms

- Menezes, van Oorschot, and Vanstone, "Handbook of Applied Cryptography"
  - Reference for cryptography (goes far beyond this course!)

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.
- Experimentation encouraged but be careful in testing!

What Does “Security” Mean to You?

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

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, web security, web privacy, authentication, usable security, special topics (e.g., mobile security)

What This Course is *Not* About

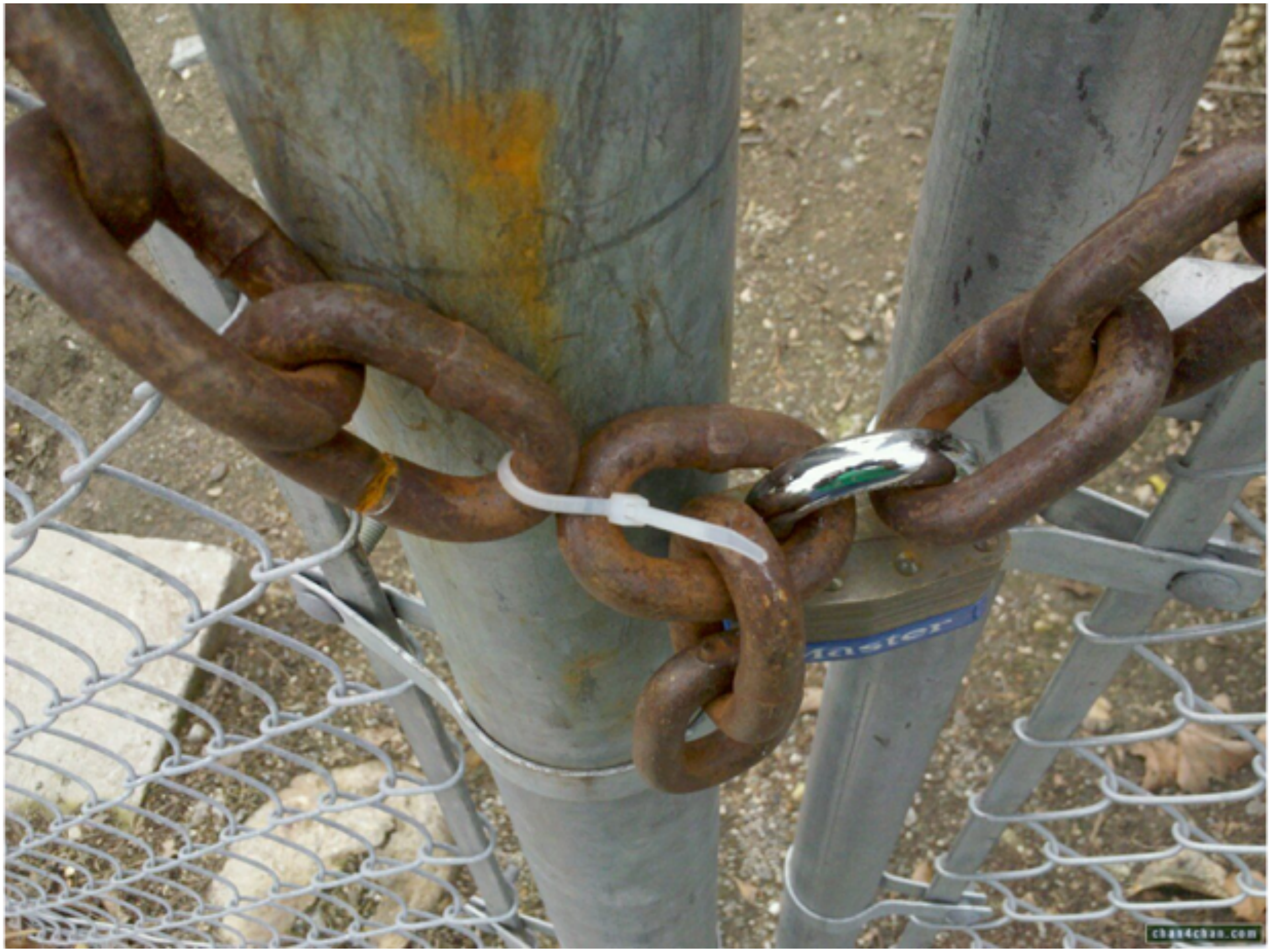
- Not a comprehensive course on computer security
  - Computer security is a broad discipline!
  - Impossible to cover everything in a few weeks
  - So be careful in industry or wherever you go!
- Not about all of the latest and greatest attacks
  - Read news
- Not a course on ethical, legal, or economic issues
  - We will touch on these 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

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.

Example



Learning the Security Mindset

- Several approaches for developing “The Security Mindset” and for exploring the broader contextual issues surrounding computer security
  - Security reviews
    - **lots of value in discussing security with others**
  - In class discussions

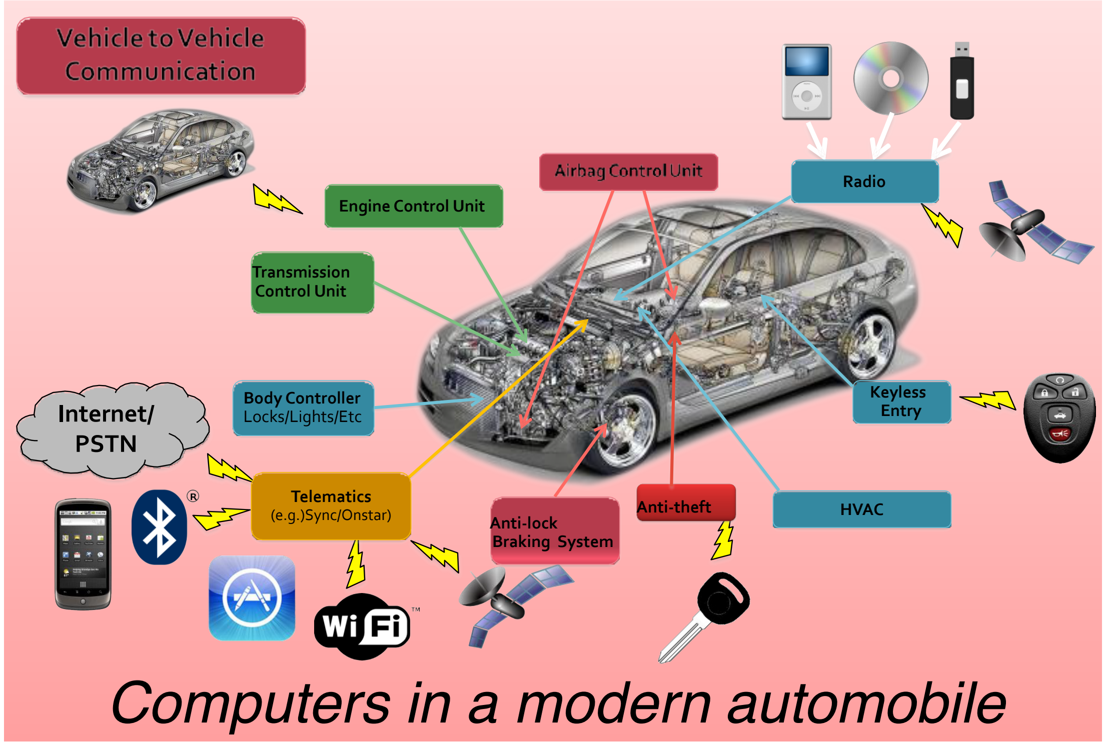
  Outside class:

  - Reading (research papers, news stories)

Example: Modern Automobiles

Modern automobiles contain dozens of computers.

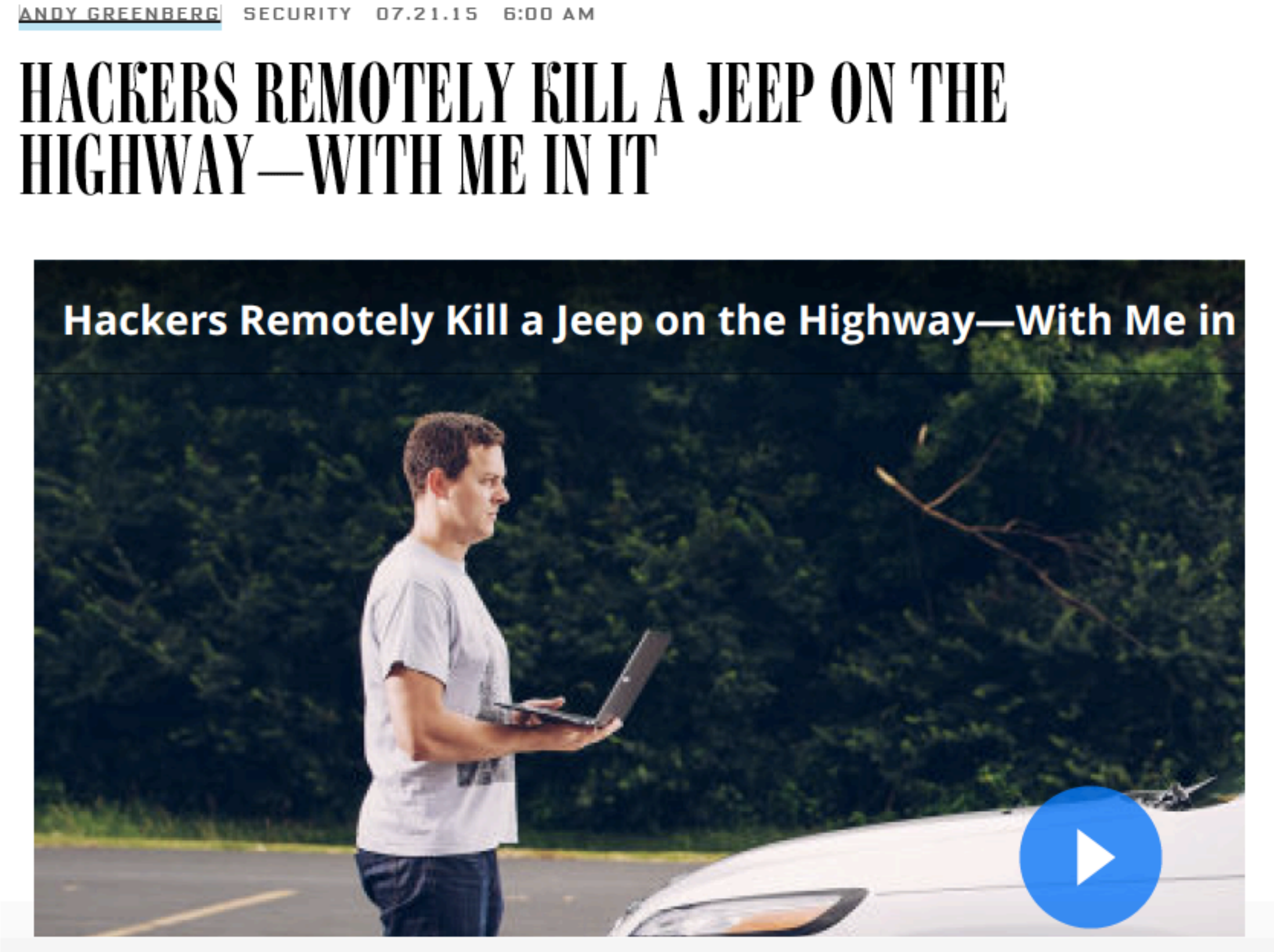
Those computers control nearly everything in the car, including locks, lights, brakes, the engine, the airbags, etc.


Vehicle to Vehicle Communication
Engine Control Unit
Airbag Control Unit
Radio
Transmission Control Unit
Body Controller Locks/Lights/Etc
Internet/ PSTN
Keyless Entry
Telematics (e.g.)Sync/Onstar)
Anti-lock Braking System
Anti-theft
HVAC
Wi Fi

Computers in a modern automobile

Who might want to attack? Why, and how?

From the news (Wired, July 2015)



ANDY GREENBERG SECURITY 07.21.15 6:00 AM

HACKERS REMOTELY KILL A JEEP ON THE HIGHWAY—WITH ME IN IT

Hackers Remotely Kill a Jeep on the Highway—With Me in

From the news (Wired, July 2015)

- remote exploit: 10 miles away from Jeep Cherokee
- cut transmission, brakes
- can track Jeep's GPS (surveillance)
- vulnerability: Uconnect cellular connection
    - can connect by knowing vehicle's IP address
    - place exploit in firmware that controls car
- very high-profile attack!
    - designed to bring attention to the problem
    - ethics?
    - Chrysler issued recall, firmware patch
    - one of those hackers now works for Tesla

The story so far…

- Importance of the security mindset
  - (challenging design assumptions, thinking like an attacker)
- There's no such thing as perfect security
- Defining security per context: identify assets, adversaries, motivations, threats, vulnerabilities, risk, possible defenses

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 Drives the Attackers?

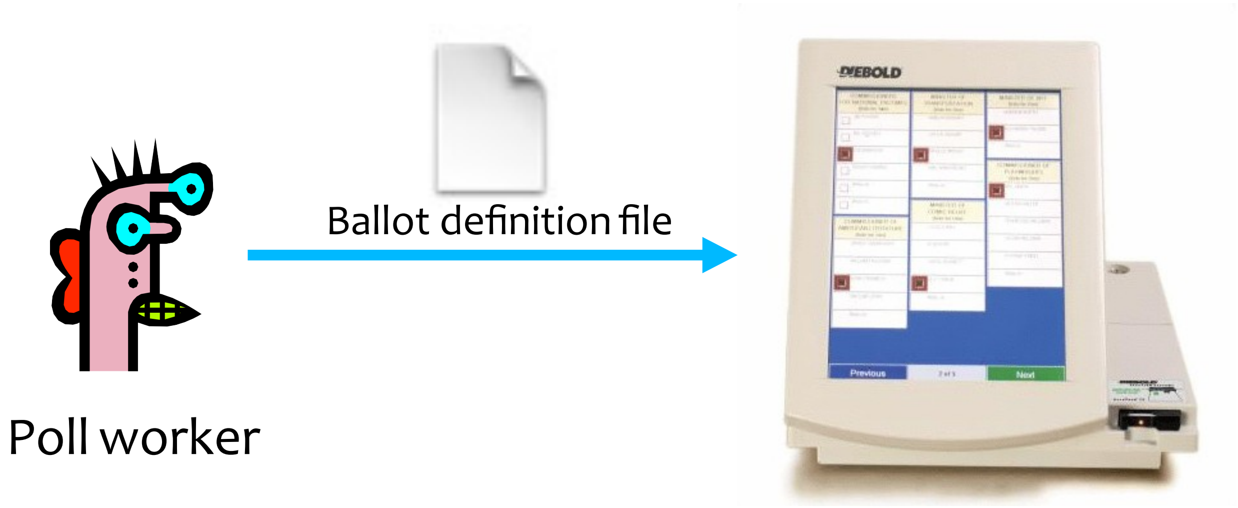
- Adversarial motivations:
  - Money, fame, malice, revenge, curiosity, politics, terror….
- Fake websites: identity theft, steal money
- Control victim's machine: send spam, capture passwords
- Industrial espionage and international politics
- Attack on website, extort money
- Wreak havoc, achieve fame and glory
- Access copy-protected movies and videos, entitlement or pleasure

Example: Electronic Voting

- Popular replacement to traditional paper ballots

Voting example from Yoshi Kohno, UW

Pre-Election

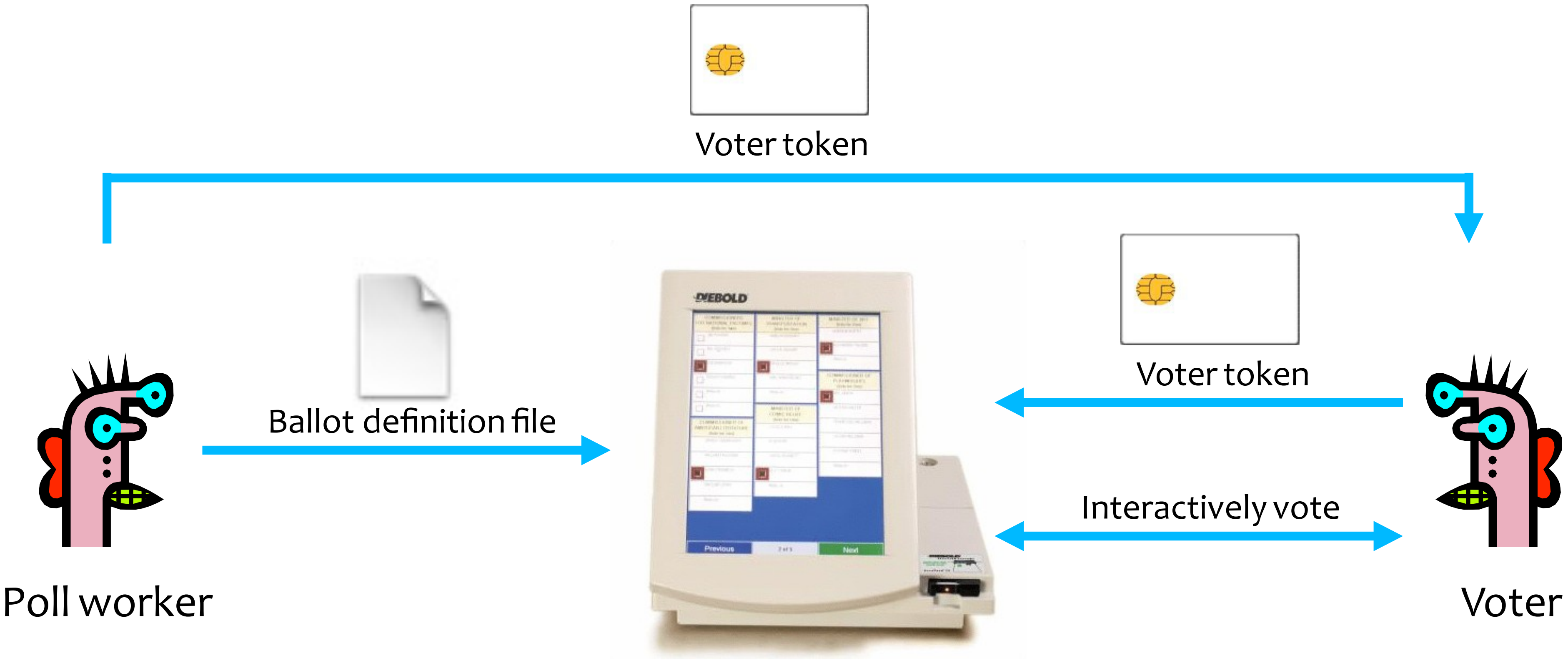


Ballot definition file

Poll worker

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

Active Voting



Voter token

Ballot definition file

Voter token

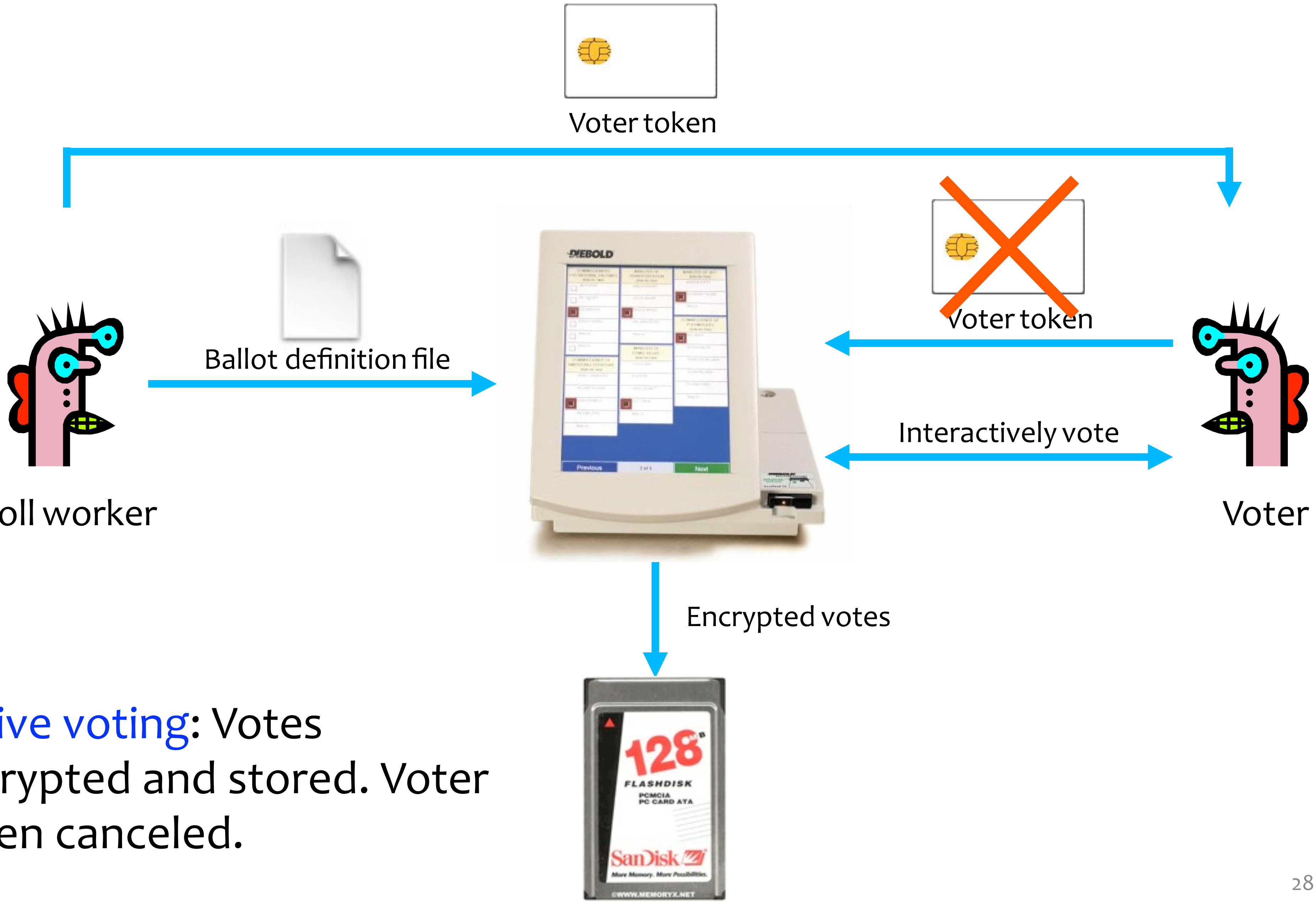
Interactively vote

Poll worker

Voter

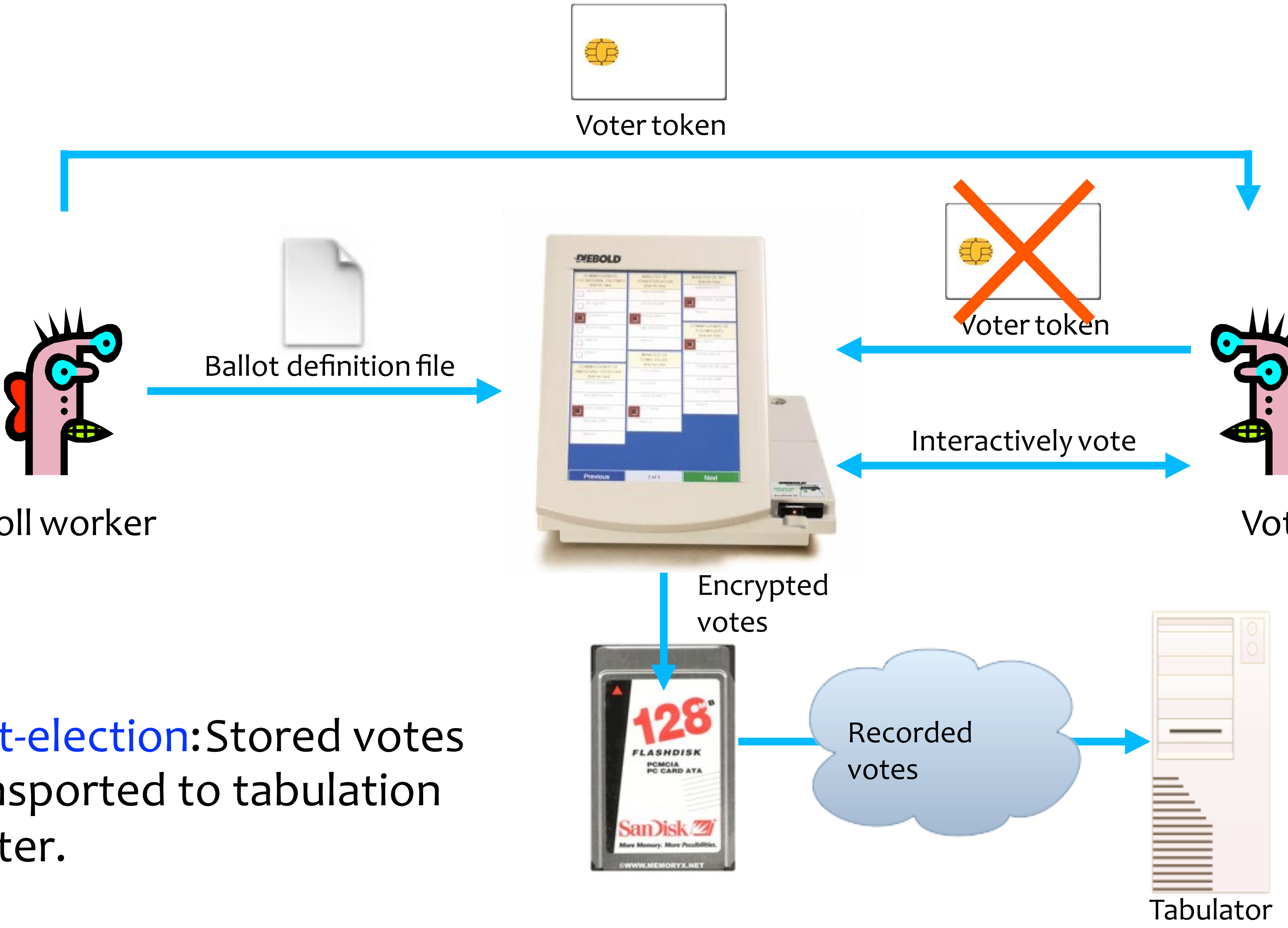
Active voting: Voters obtain single-use tokens from poll workers. Voters use tokens to activate machines and vote.

Active Voting



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

Post-Election

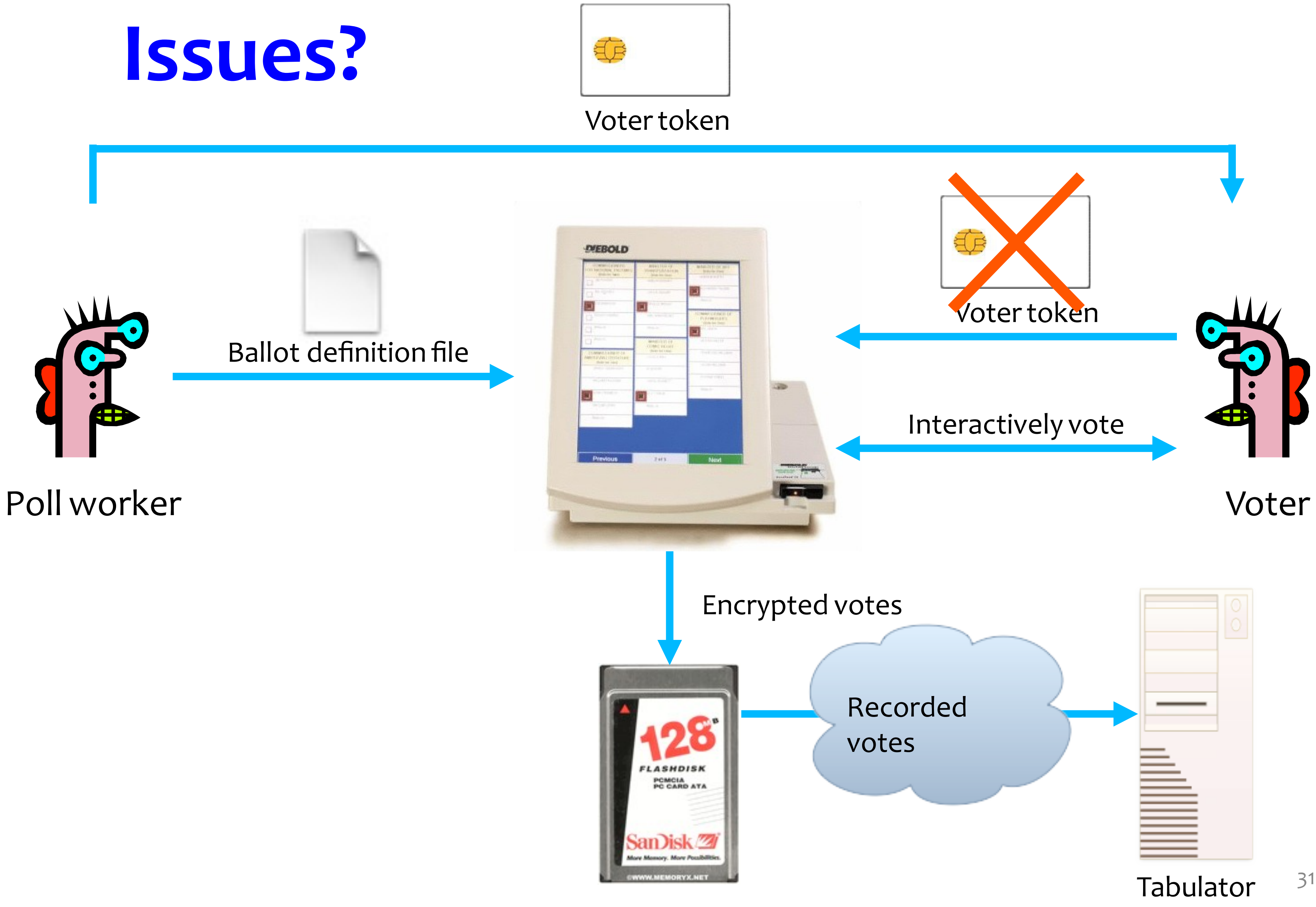


Post-election: Stored votes transported to tabulation center.

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
  - Adversary should not be able to figure out how voters vote

Can You Spot Any Potential Issues?



Potential Adversaries

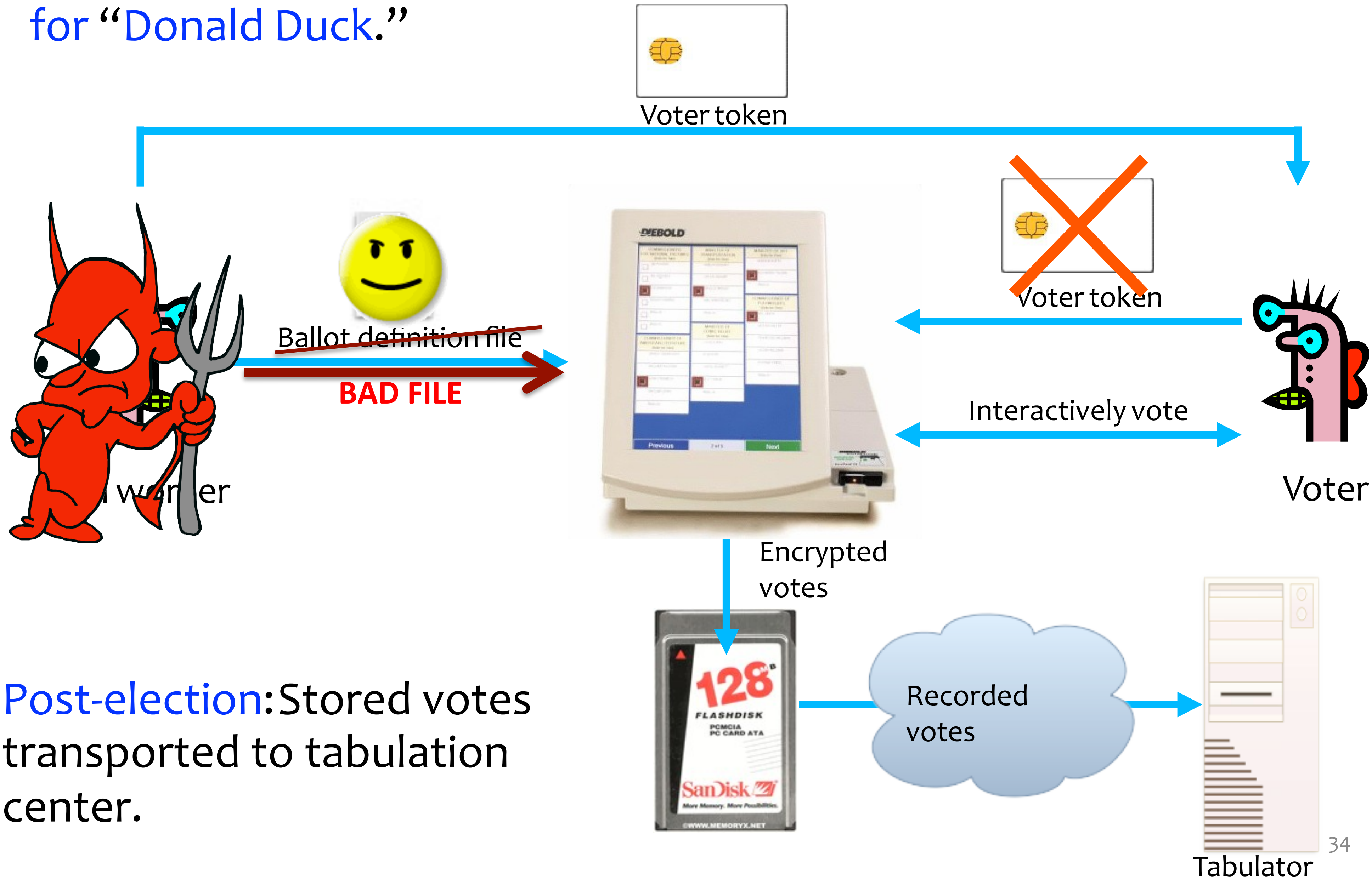
- Voters
- Election officials
- Employees of voting machine manufacturer
  - Software/hardware engineers
  - Maintenance people
- Other engineers
  - Makers of hardware
  - Makers of underlying software or add-on components
  - Makers of compiler
- …
- Or any combination of the above

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.

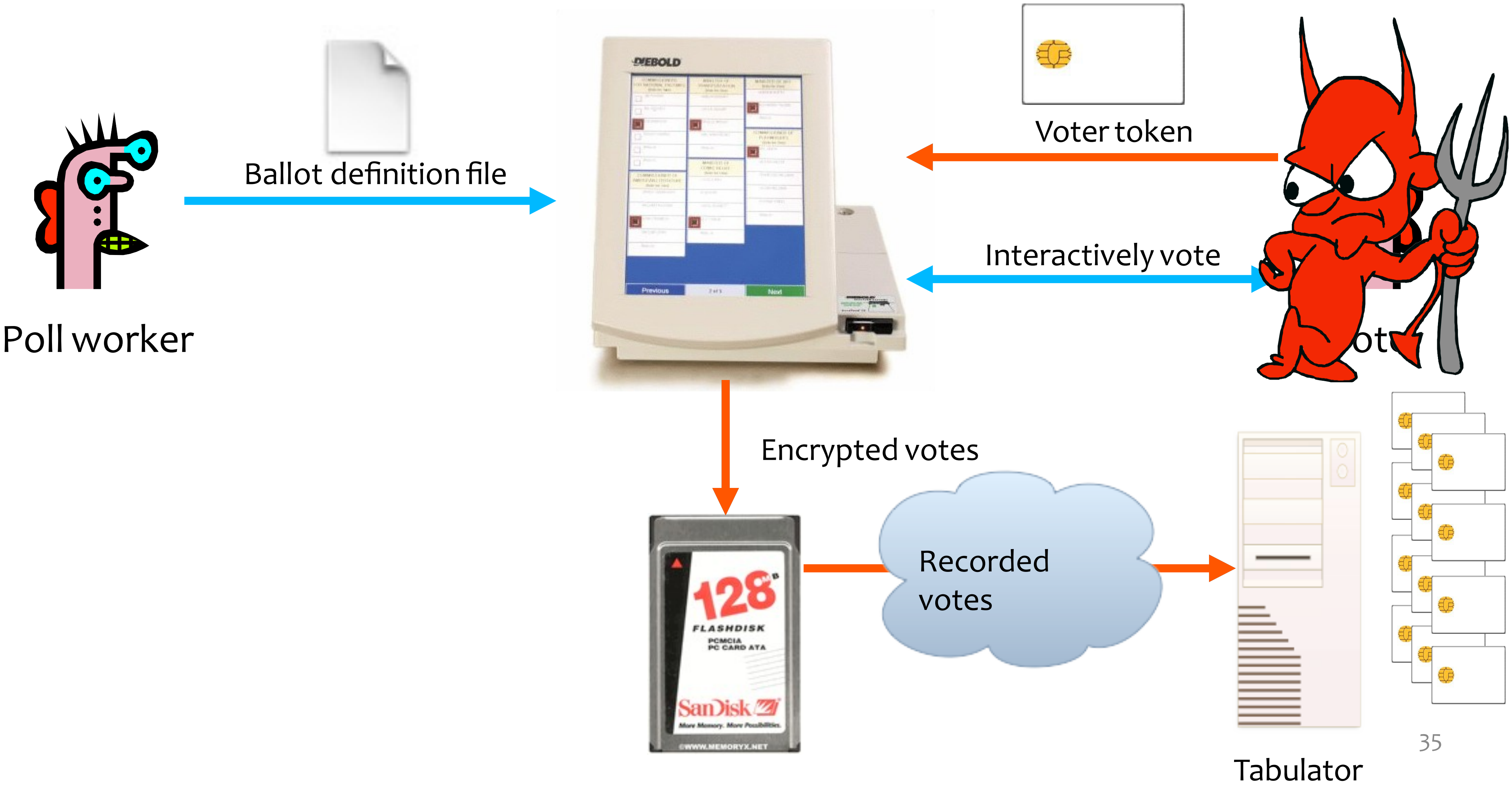
Example attack: A malicious poll worker could modify ballot definition files so that votes cast for "Mickey Mouse" are recorded for "Donald Duck."



Post-election: Stored votes transported to tabulation center.

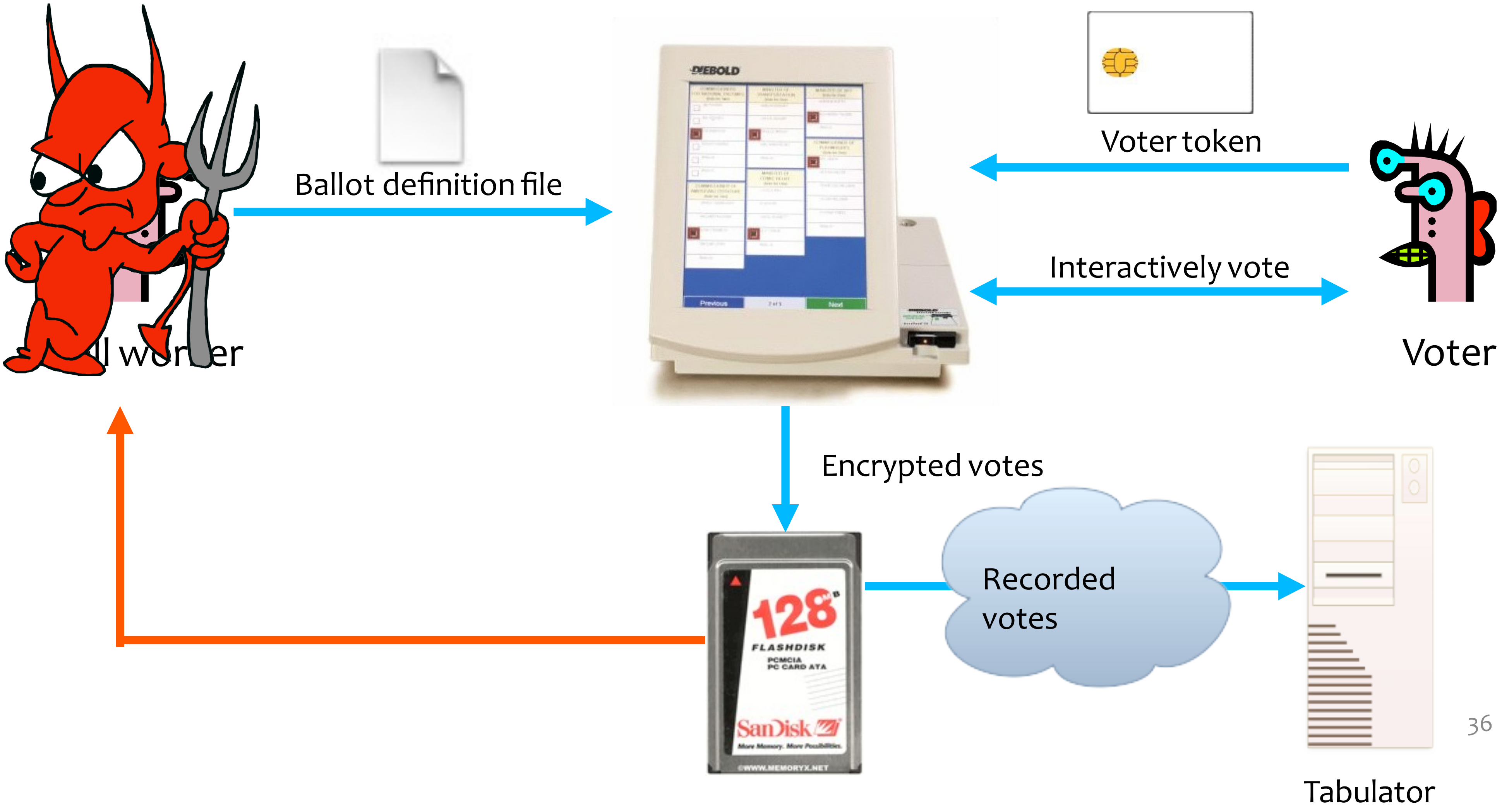
Problem: Smartcards *could* perform cryptographic operations. But if unused: no authentication from voter token to terminal.

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



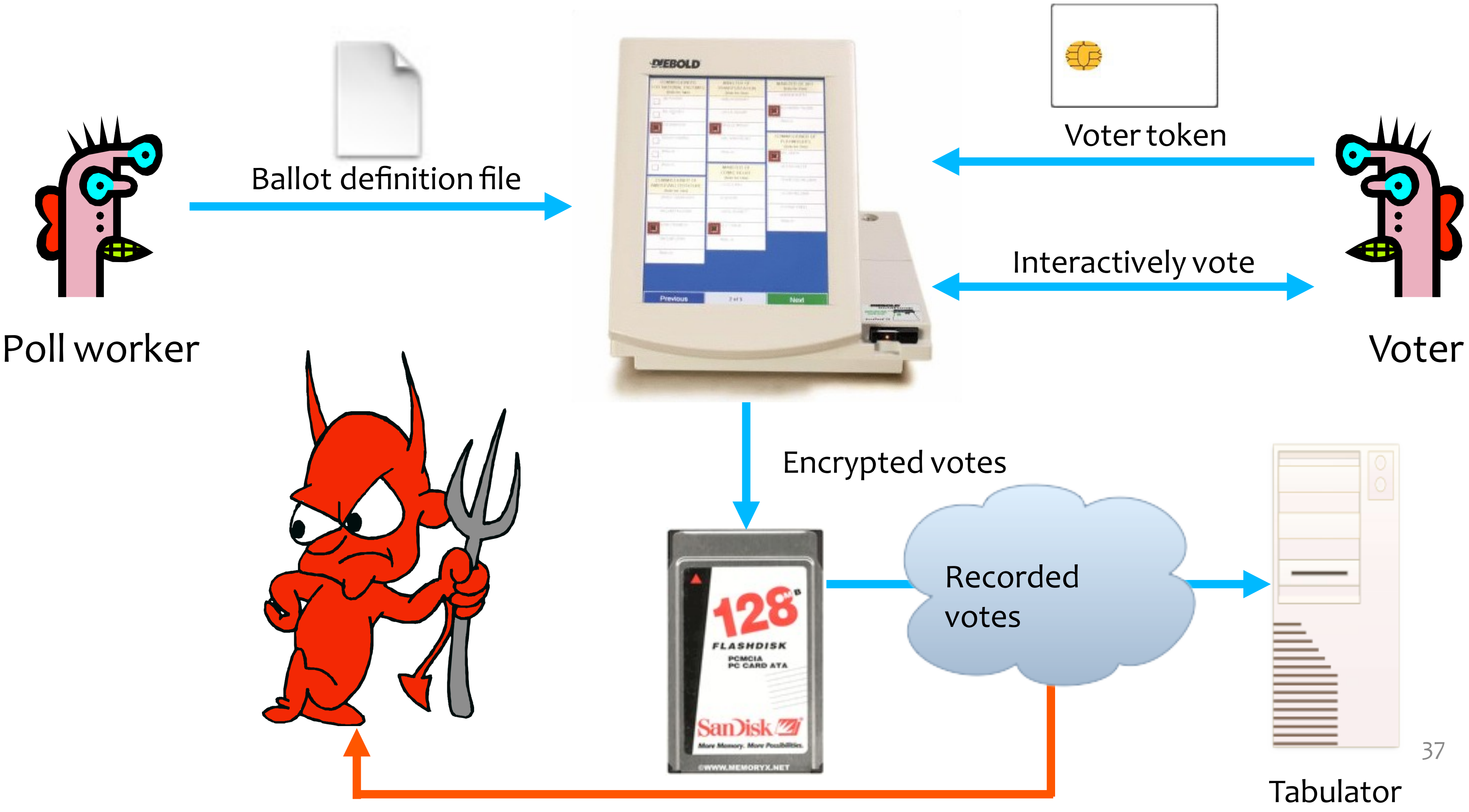
Problem: Encryption key (“F2654hD4”) hard-coded into the software for years. Votes stored in the order cast.

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.

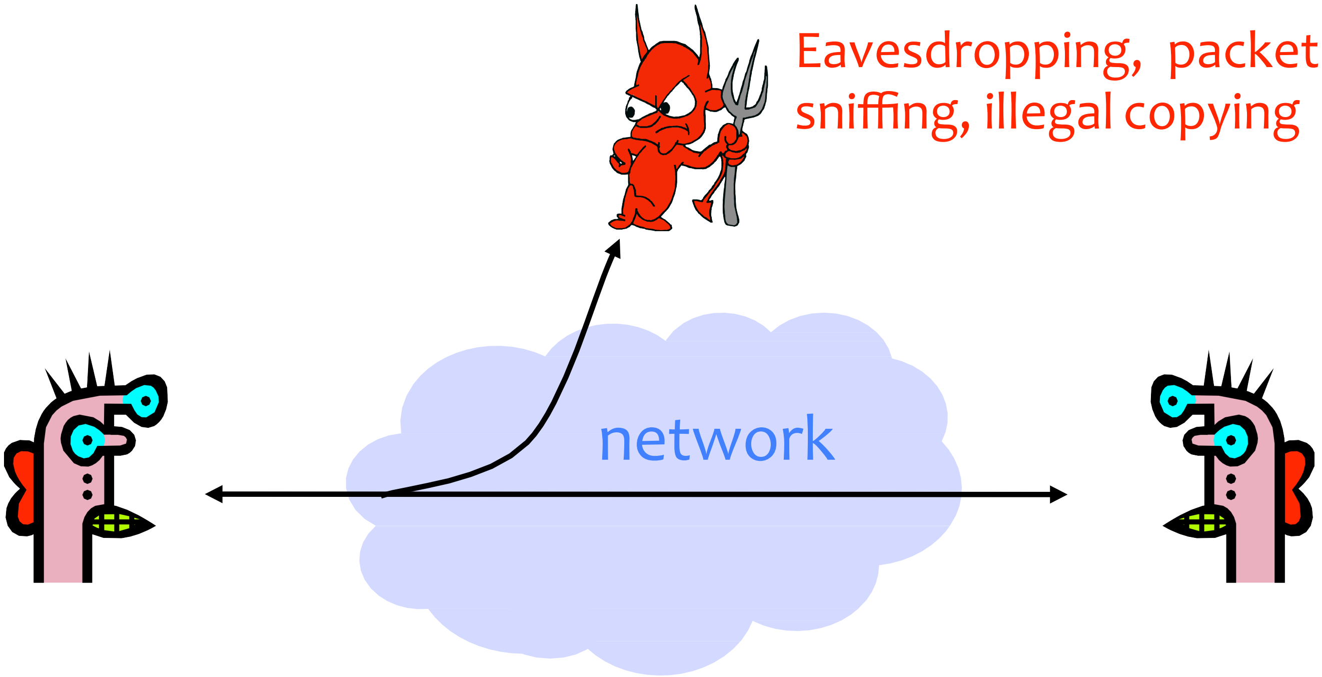
Example attack: A sophisticated outsider could determine how voters vote.



SECURITY GOALS (“CIA”)

Confidentiality (Privacy)

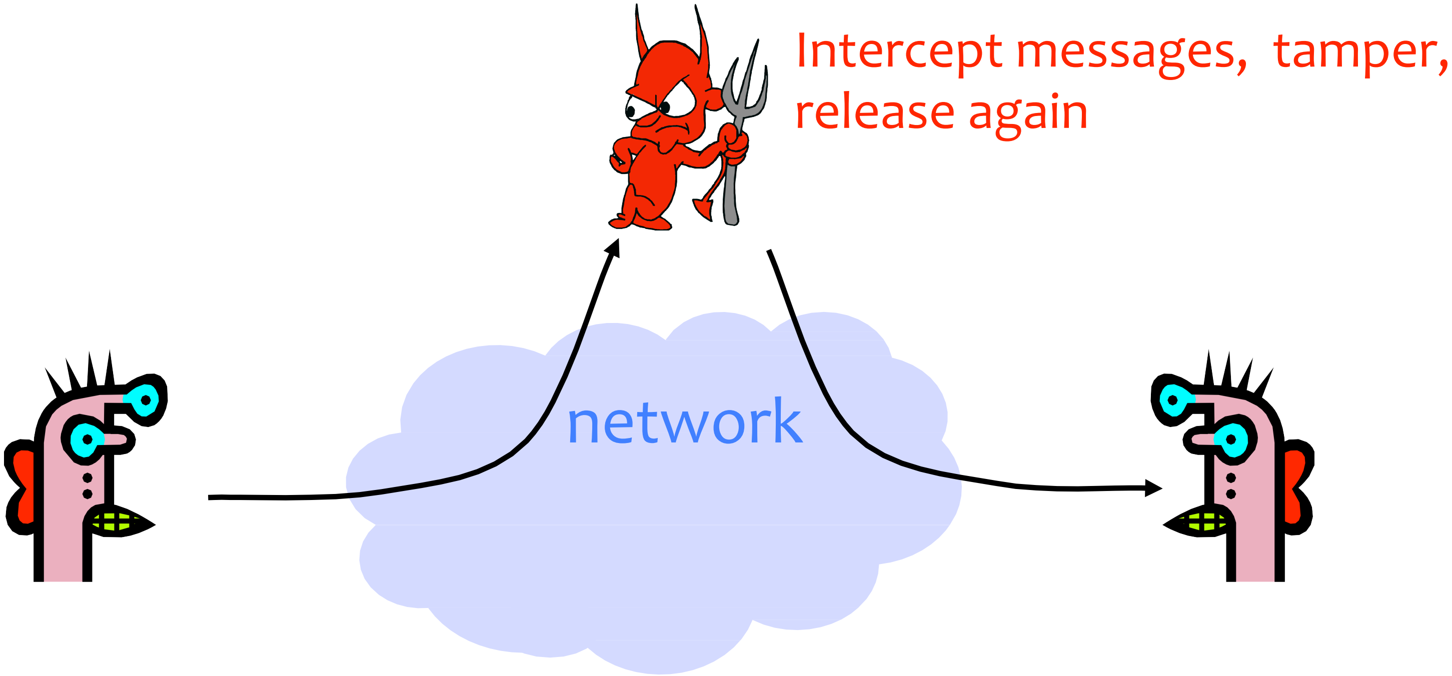
- Confidentiality is concealment of information



Eavesdropping, packet sniffing, illegal copying

Integrity

- Integrity is prevention of unauthorized changes

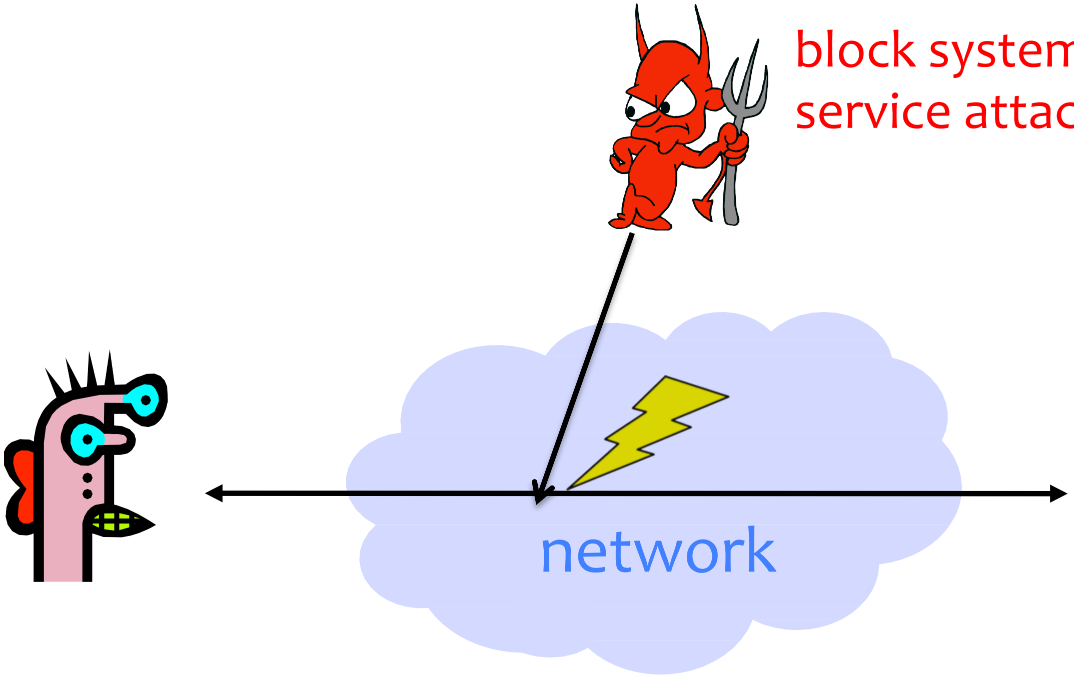


Intercept messages, tamper, release again

network

Availability

- Availability is ensuring information can flow over communications systems; systems remain operational



block systems, denial of service attacks

From Policy to Implementation

- After you've figured out what security means to your application, there are still challenges:
  - Bugs in requirements
  - Incorrect or problematic goals
  - 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 anattack
- Detection
  - Detect an ongoing or past attack
- Response
  - Respond toattacks

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

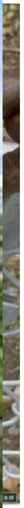
Whole System is Critical

- 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)

Whole System is Critical

- Securing a system involves a whole system view
  - C
  - I
  - P
  - P
  - E
- This
  wea
  plac
  - N you
    c
  - (link)



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
  - Security is not a binary property
  - Many security holes are based on misunderstanding
- Security awareness and user "buy-in" help

Overview of course content

“Theme 2”: Technical aspects of security

Major modules

1. Software security
2. Cryptography
3. Authentication
4. Web security
5. Special topics: usable security; mobile security

subject to some adjustments