Week 13: Lecture B

Election Cybersecurity

Thursday, November 21, 2024

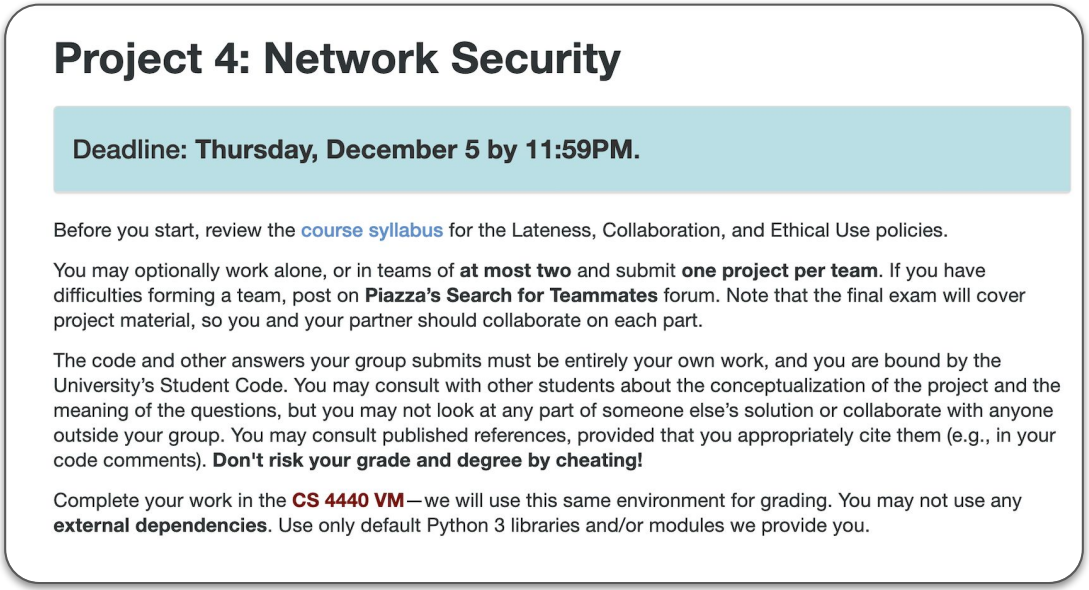
Announcements

- **Project 3: WebSec** regrades posted
  - If your team submitted a regrade request, you'll see a comment on **Canvas**
  - If you don't see one, **let me know!**

- **Questions?** See me after lecture

Announcements

- **Project 4: NetSec** released
  - **Deadline:** Thursday, December 5th by 11:59PM



Project 4: Network Security

Deadline: Thursday, December 5 by 11:59PM.

Before you start, review the course syllabus for the Lateness, Collaboration, and Ethical Use policies.

You may optionally work alone, or in teams of **at most two** and submit **one project per team**. If you have difficulties forming a team, post on **Piazza's Search for Teammates** forum. Note that the final exam will cover project material, so you and your partner should collaborate on each part.

The code and other answers your group submits must be entirely your own work, and you are bound by the University's Student Code. You may consult with other students about the conceptualization of the project and the meaning of the questions, but you may not look at any part of someone else's solution or collaborate with anyone outside your group. You may consult published references, provided that you appropriately cite them (e.g., in your code comments). **Don't risk your grade and degree by cheating!**

Complete your work in the **CS 4440 VM**—we will use this same environment for grading. You may not use any **external dependencies**. Use only default Python 3 libraries and/or modules we provide you.

Project 4 Progress

Working on Part 1

0%

Finished Part 1, working on Part 2

0%

Finished both Part 1 and Part 2

0%

None of the above

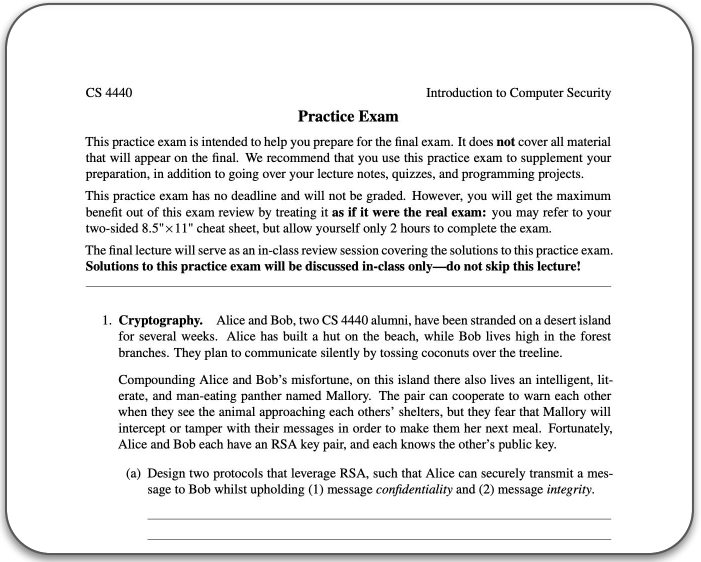
0%

Final Exam

- **Save the date: 1–3PM** on **Tuesday, December 10**
  - **CDA accommodations:** schedule exam via CDA Portal
- **High-level details** (more to come):
  - One exam covering all course material
  - Similar to project/quiz/lecture exercises
- **Cheat Sheet**
  - **One 8.5”x11” paper** with handwritten/typed notes on **both** sides
  - **Suggestion:** Don’t just use someone else’s—you’ll learn better making **your own**!
  - **Suggestion:** Don’t just paste lecture slides—you’ll learn better by **writing/typing** it!

Practice Exam

- **Practice Exam** released
  - See **Assignments** page on the CS 4440 website
- **Final lecture** will serve as a **review session**
  - Solutions discussed **in-class only**—don't skip!



CS 4440 Introduction to Computer Security

Practice Exam

This practice exam is intended to help you prepare for the final exam. It does **not** cover all material that will appear on the final. We recommend that you use this practice exam to supplement your preparation, in addition to going over your lecture notes, quizzes, and programming projects.

This practice exam has no deadline and will not be graded. However, you will get the maximum benefit out of this exam review by treating it **as if it were the real exam:** you may refer to your two-sided 8.5"×11" cheat sheet, but allow yourself only 2 hours to complete the exam.

The final lecture will serve as an in-class review session covering the solutions to this practice exam. **Solutions to this practice exam will be discussed in-class only—do not skip this lecture!**

1. **Cryptography.** Alice and Bob, two CS 4440 alumni, have been stranded on a desert island for several weeks. Alice has built a hut on the beach, while Bob lives high in the forest branches. They plan to communicate silently by tossing coconuts over the treeline.

   Compounding Alice and Bob's misfortune, on this island there also lives an intelligent, literate, and man-eating panther named Mallory. The pair can cooperate to warn each other when they see the animal approaching each others' shelters, but they fear that Mallory will intercept or tamper with their messages in order to make them her next meal. Fortunately, Alice and Bob each have an RSA key pair, and each knows the other's public key.

   (a) Design two protocols that leverage RSA, such that Alice can securely transmit a message to Bob whilst upholding (1) message *confidentiality* and (2) message *integrity*.

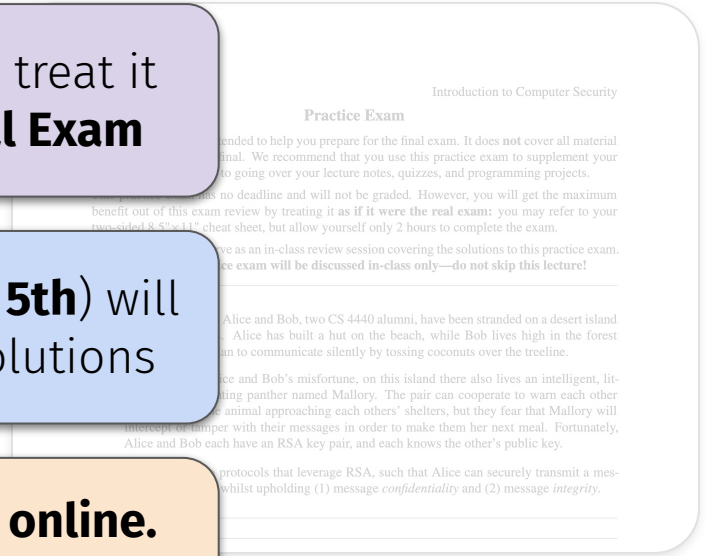
Practice Exam

- **Practice Exam** re
  - See **Assignmen**
- **Final lecture** will serve as a review session
  - Solutions discu

To get the most out of this, treat it just **as you would the Final Exam**

Last lecture (**Thursday, Dec. 5th**) will go over the exam review solutions

Solutions won't be posted online.
(Reminder: attendance/participation makes up **5%** of your course grade)



Introduction to Computer Security

Practice Exam

ended to help you prepare for the final exam. It does **not** cover all material inal. We recommend that you use this practice exam to supplement your to going over your lecture notes, quizzes, and programming projects.

s no deadline and will not be graded. However, you will get the maximum benefit out of this exam review by treating it **as if it were the real exam:** you may refer to your two-sided 8.5"×11" cheat sheet, but allow yourself only 2 hours to complete the exam.

ve as an in-class review session covering the solutions to this practice exam. **ce exam will be discussed in-class only—do not skip this lecture!**

Alice and Bob, two CS 4440 alumni, have been stranded on a desert island Alice has built a hut on the beach, while Bob lives high in the forest an to communicate silently by tossing coconuts over the treeline.

ce and Bob's misfortune, on this island there also lives an intelligent, lit- ting panther named Mallory. The pair can cooperate to warn each other e animal approaching each others' shelters, but they fear that Mallory will intercept or tamper with their messages in order to make them her next meal. Fortunately, Alice and Bob each have an RSA key pair, and each knows the other's public key.

protocols that leverage RSA, such that Alice can securely transmit a mes- whilst upholding (1) message *confidentiality* and (2) message *integrity*.

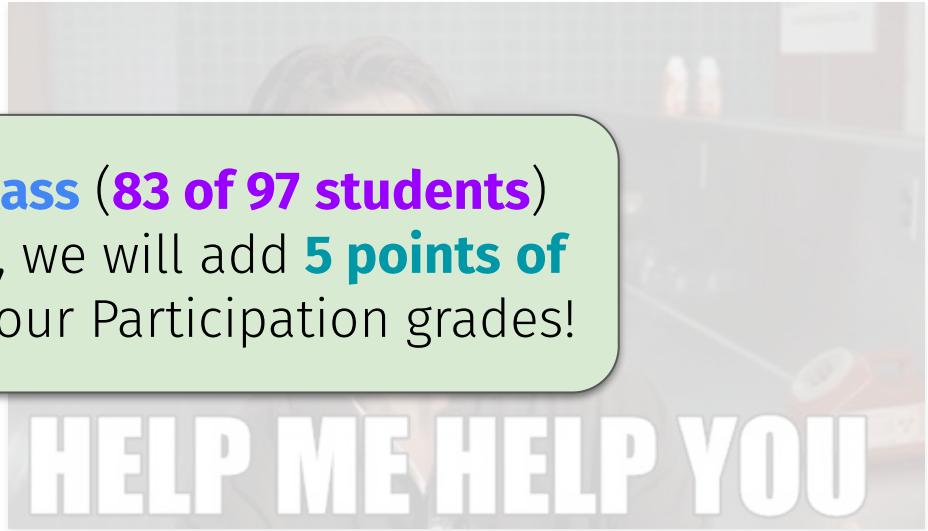
End-of-semester Course Evals

- **I want your feedback!**
  - 3rd time teaching this course 😃
  - **Help me improve the class!**
- Due by **December 19th**
  - https://scf.utah.edu
  - **Please please please!**

End-of-semester Course Evals

- **I want your feedback!**
  - 3rd time teaching this course 😃
  - **Help me improve the class!**
- Due by **Dece**
  - https://sc
  - **Please pl**

If 85% of the class (**83 of 97 students**) submits an eval, we will add **5 points of extra credit** to your Participation grades!



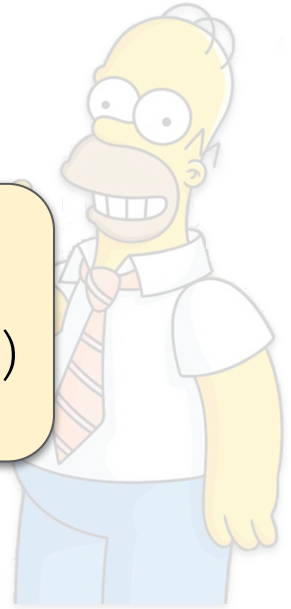
Reminders: Participation Extra Credit

- **Piazza:** 5 points per top-10 student contributors
  - Answering peers' questions
  - Providing helpful resources
- **Wiki Contributions:** 1 point per approved contribution
  - Must be cleared in advance
- **Course Evals:** 5 points if 85% of class submits evals
  - Will be released soon on **scf.utah.edu**

Reminders: Participation Extra Credit

- **Piazza:** 5 points per top-10 student contributors
  - Answering peers' questions
  - Providing helpful resources
- **Wiki Contri**
  - Must be
- **Course Eva**
  - Will be released soon on

Final deadline for extra credit will be the last day of class (**Thursday, December 5th**)



Announcements

Guest Industry Lecture!

Join ACM and Recursion Pharmaceuticals:

- Recursion Integrates technology to revolutionize and automate drug discovery.
- Partnering with leading industry organizations to advance breakthroughs in TechBio

RECURSION
pharmaceuticals

Thurs, Nov 21, 5pm

MEB 3515

Please RSVP for headcount

Announcements

utahsec

See Discord for meeting info!

utahsec.cs.utah.edu

No Class Next Week

Questions?

Last time on CS 4440...

Side Channels
Hardware Security
Hardware Supply Chain Attacks

Side Channels

- What are some potential sources of **indirect info** emitted by your computer?
  - **Additional channels** of information beyond what is directly visible/accessible to you

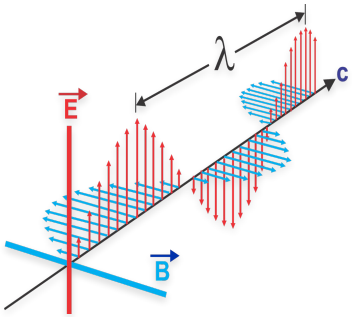
What are some sources of indirect info emitted by a computer?

Nobody has responded yet.

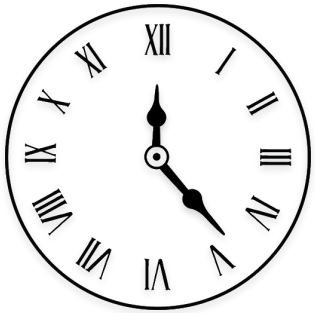
Hang tight! Responses are coming in.

Side Channels

- What are some potential sources of **indirect info** emitted by your computer?
  - **Additional channels** of information beyond what is directly visible/accessible to you



Emitted Radiation

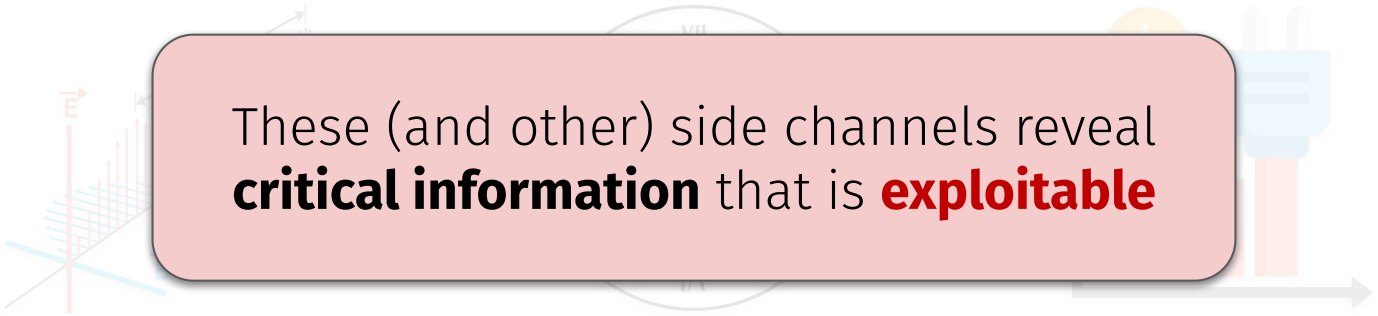


Execution Time

Power Consumption

Side Channels

- What are some potential sources of **indirect info** emitted by your computer?
  - **Additional channels** of information beyond what is directly visible/accessible to you

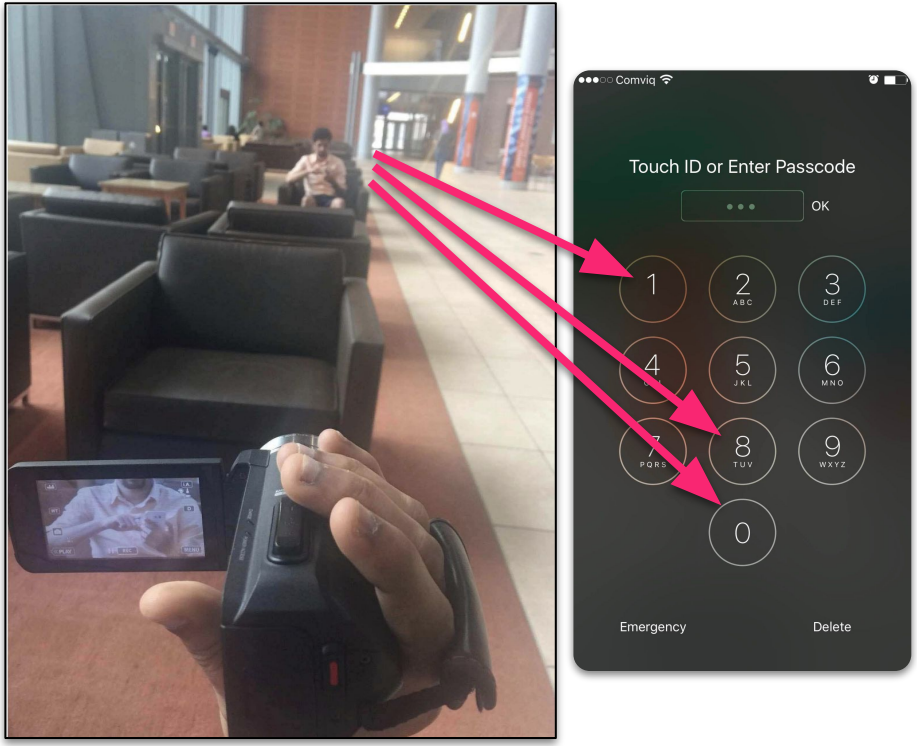


These (and other) side channels reveal **critical information** that is **exploitable**

Emitted Radiation | Execution Time | Power Consumption

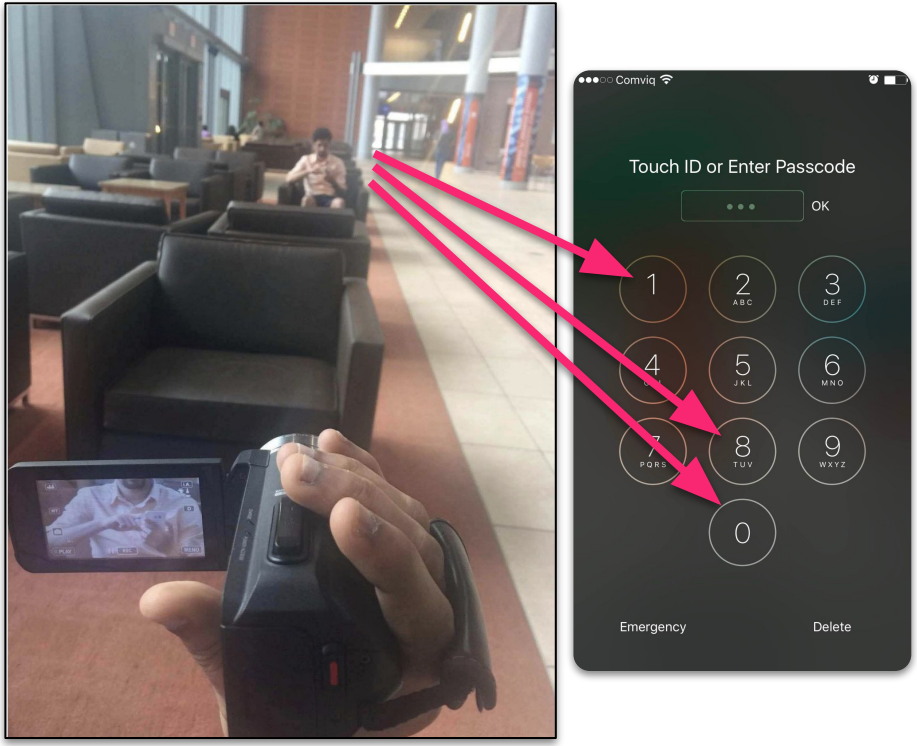
Optical Side Channels

- **Stealing passwords via gestures**
  - ???



Optical Side Channels

- **Stealing passwords via gestures**
  - Capture visible **hand movements**
  - Assume attacker **knows (or can easily guess)** the key interface
  - Attacker **maps movements** to pressed keys on the interface



Acoustic Side Channels

- **Stealing passwords via key press noises**
  - ???

Acoustic Side Channels

- **Stealing passwords via key press noises**
  - Build model of key press noises
    - Consider microphone
    - Consider ambient noise
  - Use model to infer entered data
    - Passwords
    - Usernames
    - Phone numbers

Other Side Channels

- How `memcmp()` works under the hood:

```
bool checkPW(char *testPW, char *realPW, int len) {

    for (int i = 0; i < len; i++) {

        if (testPW[i] != realPW[i]) {
            return false;
        }
    }

    return true;
}
```

What is the side channel here?

Other Side Channels

- How `memcmp()` works under the hood:

```
bool checkPW(char *testPW, char *realPW, int len) {

    for (int i = 0; i < len; i++) {

        if (testPW[i] != realPW[i]) {
            return false;
        }
    }

    return true;
}
```

Password Login Attempts:

`ABCDEFGH == PASSWORD`
- **False** on first iteration

`PASSEFGH == PASSWORD`
- **True** on iterations **1–4**
- **False** on **fifth** iteration

More code executed
for a **correct** symbol!

Other Side Channels

How can this **side channel** be **exploited**?

Other Side Channels

How can this **side channel** be **exploited**?

Attacker: CRCDEF

Server: False
Server took 2ms to respond

Attacker: CHIDEF

Server: False
Server took 4ms to respond

Other Side Channels

How can this **side channel** be **exploited**?

Attacker: `CHIEFS`

Server: `True`
Server took **7ms** to respond

Through **timing analysis**, attacker can infer the **correctness** of individual **password symbols**!

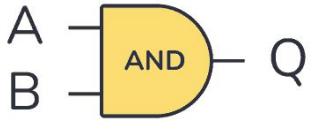
Avoiding Side Channels

- **Solution:**
  - ???

Avoiding Side Channels

- **Solution:**
  - **Constant-time** implementation (e.g., using bitwise **AND**-ing)

```
bool checkPW(char *testPW, char *realPW, int len) {
    bool result = 1; // integer equiv of "true"
    for (int i = 0; i < len; i++) {
        result &= ca[i] == cb[i];
        return result;
    }
}
```



A	B	Q
0	0	0
0	1	0
1	0	0
1	1	1

Avoiding Side Channels

- **Solution:**
  - **Constant-time** implementation (e.g., using bitwise **AND**-ing)

```
bool checkPW(char *testPW, char *realPW, int len) {

    bool result = 1; // integer equiv of "true"

    for (int i = 0; i < len; i++) {

        result &= ca[i] == cb[i];

        return result;
    }
}
```

Guess: PASSEFGH
Bit: 11110000
Result: **False**

Password Login Attempts:

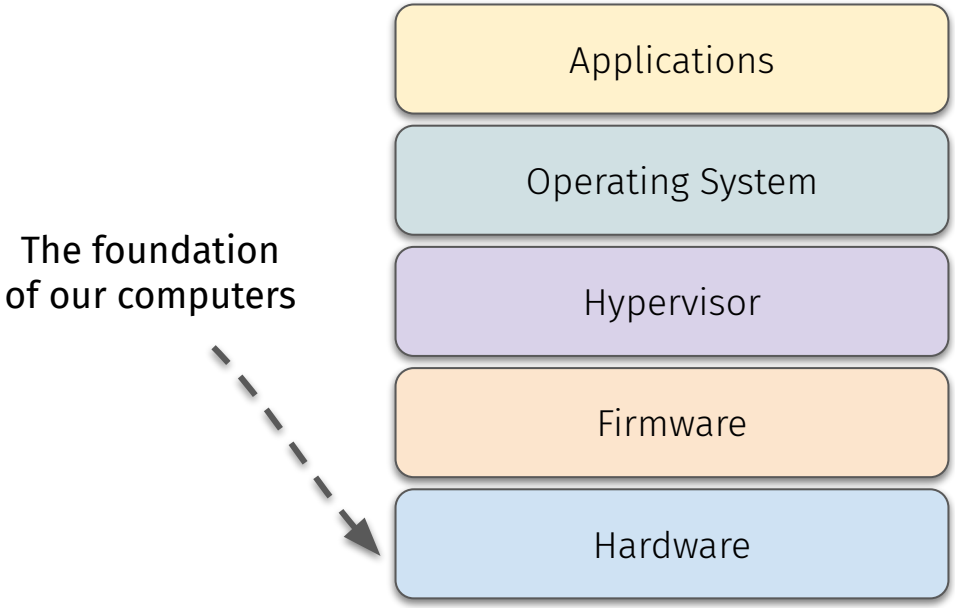
ABCDEFGH == PASSWORD
- **False** on **last** iteration

PASSEFGH == PASSWORD
- **False** on **last** iteration

PASSWORD == PASSWORD
- **True** on **last** iteration

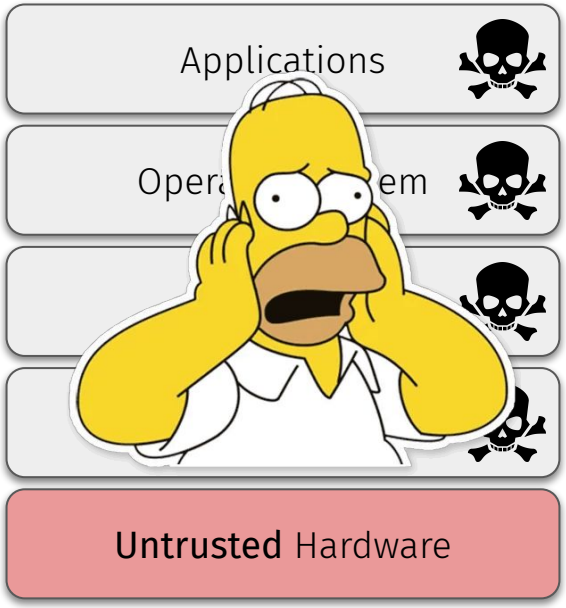
True and **False** run for **identical time**!

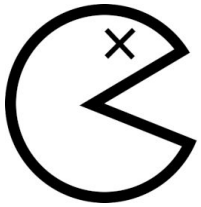
Hardware Threats



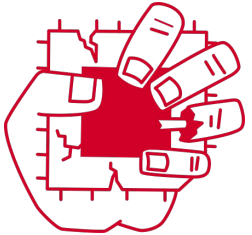
Hardware Threats

The foundation of our computers





Weaknesses weaken the entire system

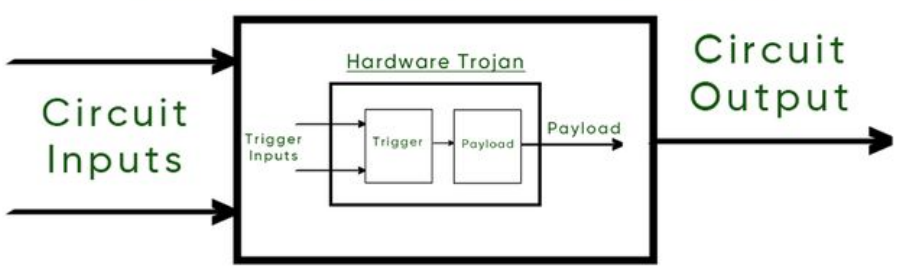


Hardware Threats

- **Hardware Trojans:**
  - **???**

Hardware Threats

- **Hardware Trojans:**
  - Attack pre-inserted into chip
  - Will be **exploited** at **run time**
  - **Remotely triggered** by attacker
    - Small
    - Stealthy
    - Controllable



Israeli sky-hack switched off Syrian radars countrywide

Backdoors penetrated without violence

Lewis Page — Thu 22 Nov 2007 / 13:57 UTC

More rumours are starting to leak out regarding the mysterious Israeli air raid against Syria in September. It is now suggested that "computer to computer" techniques and "air-to-ground network penetration" took place.

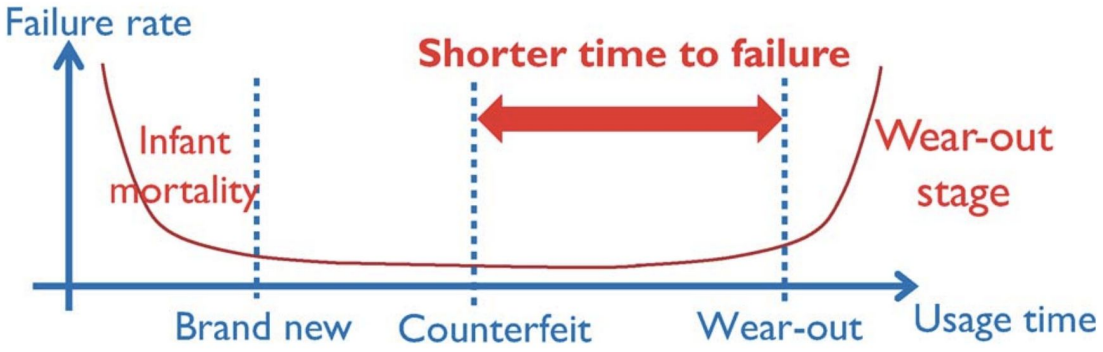
The latest revelations are made by well-connected *Aviation Week* journalists. Electronic-warfare correspondent David Fulghum says that US intelligence and military personnel "provided advice" to the Israelis regarding methods of breaking into the Syrian air-defence network.

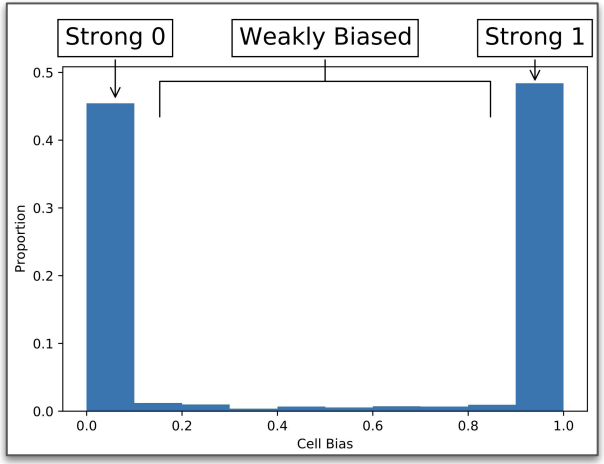
Hardware Threats

- **Counterfeit and recycled chips:**
  - **???**

Hardware Threats

- **Counterfeit and recycled chips:**
  - Have a **shorter** lifespan—leads cell bias and/or earlier wear-out
  - Absolutely dangerous for security-critical use cases





Questions?

This time on CS 4440...

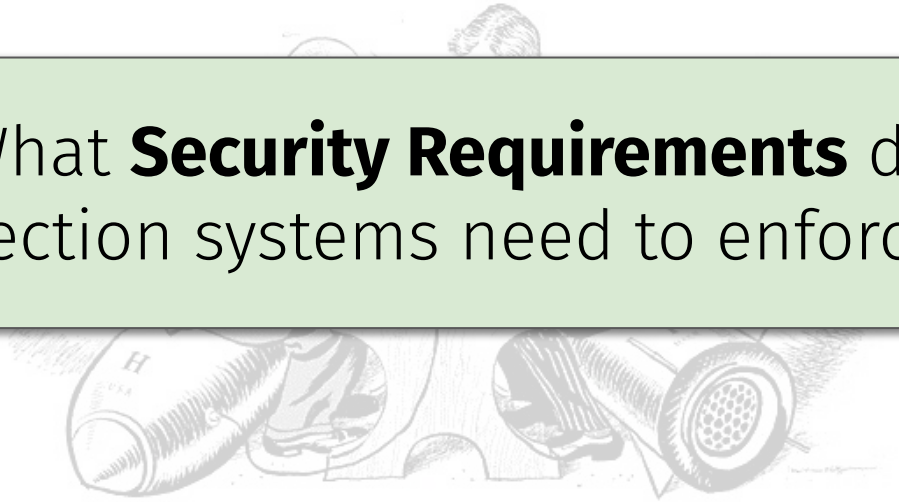
Election Cybersecurity
Voting Technology
Computerized Voting
Attacking Voting Systems

Elections

- Why have them?

Elections

- Why have them?



What **Security Requirements** do election systems need to enforce?

What security requirements must election systems enforce?

Nobody has responded yet.

Hang tight! Responses are coming in.

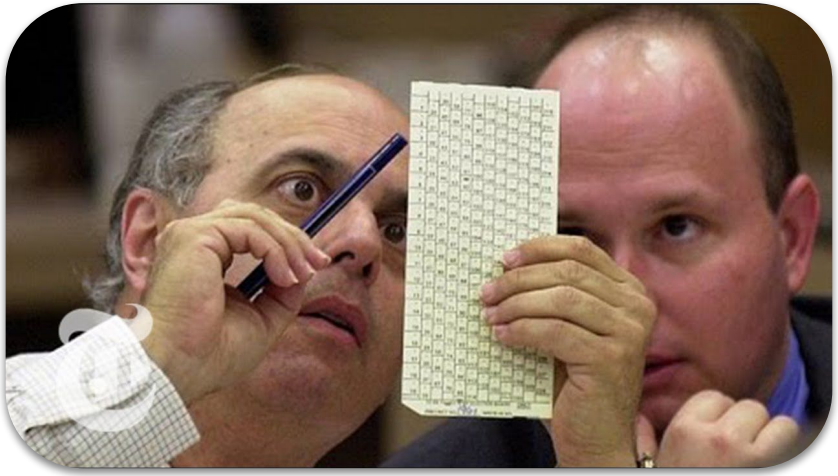
Requirement #1: Integrity

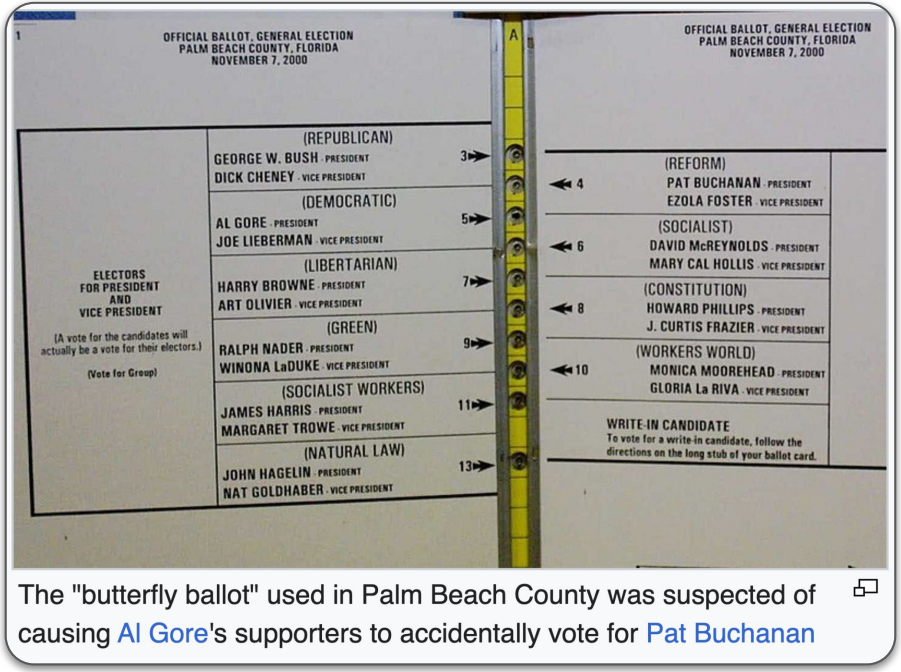
- **Goals: ???**

Requirement #1: Integrity

- **Goals:** outcome matches voter's **intent**
  - Votes are cast as intended
  - Votes are counted as cast

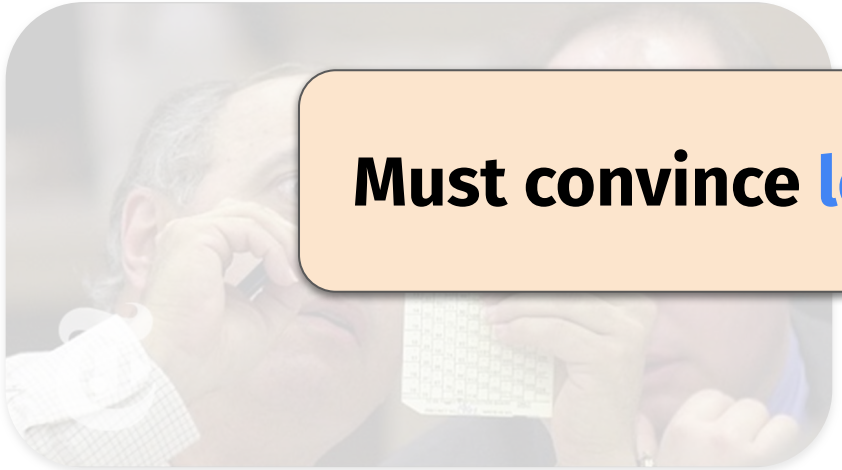
Requirement #1: Integrity





The "butterfly ballot" used in Palm Beach County was suspected of causing Al Gore's supporters to accidentally vote for Pat Buchanan

Requirement #1: Integrity



Must convince loser that they lost

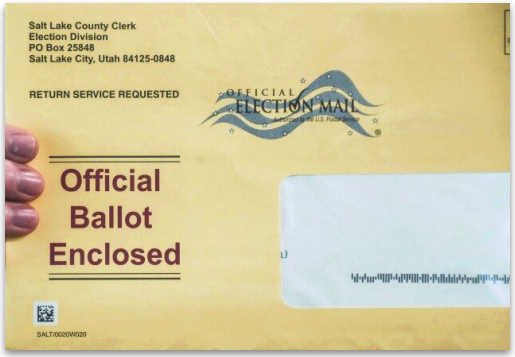
The "butterfly ballot" used in Palm Beach County was suspected of causing Al Gore's supporters to accidentally vote for Pat Buchanan

Requirement #2: Confidentiality

- **Goals: ???**

Requirement #2: Confidentiality

- **Goals:** nobody can figure out **how** you voted
  - … even if you try to prove it to them

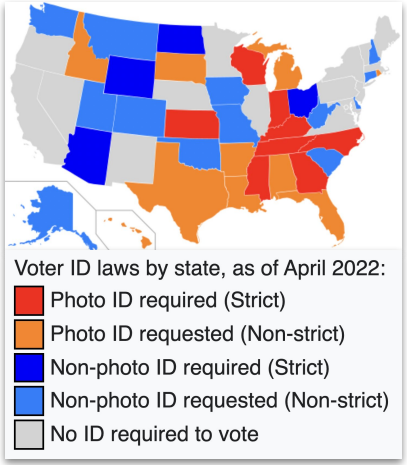


Requirement #3: Authentication

- **Goals: ???**

Requirement #3: Authentication

- **Goals:**
  - Only **authorized voters** can cast votes
  - Each voter can cast **at most one** vote

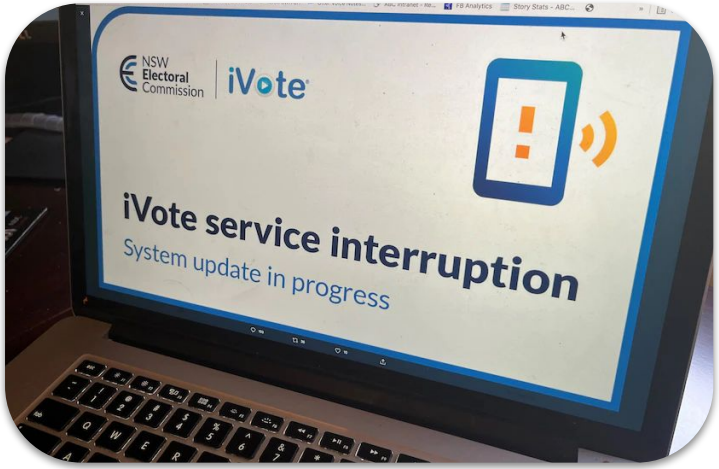


Requirement #4: Availability

- **Goals: ???**

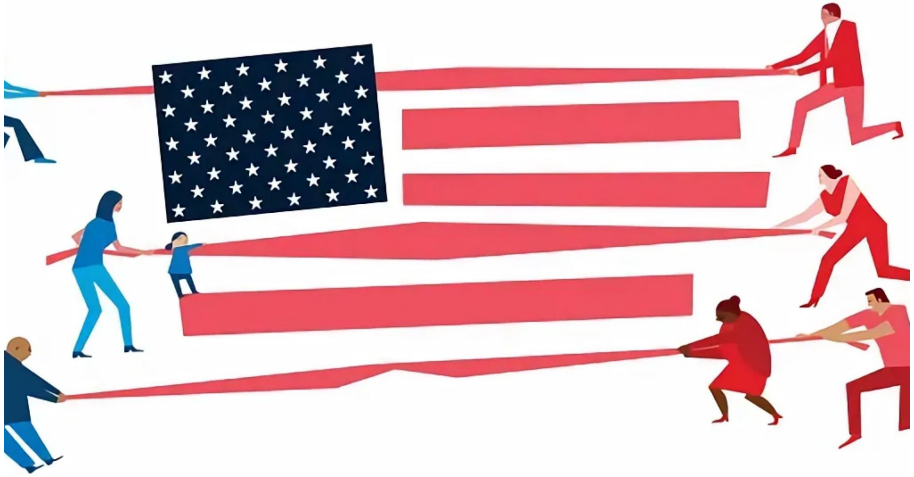
Requirement #4: Availability

- **Goals:**
  - All authorized voters have **opportunity** to vote
  - System is able to **accept all votes** on schedule
  - System can produce results in a **timely manner**



Tension Between these Properties

Ballot Integrity ◄- - - - - ► **Ballot Confidentiality**

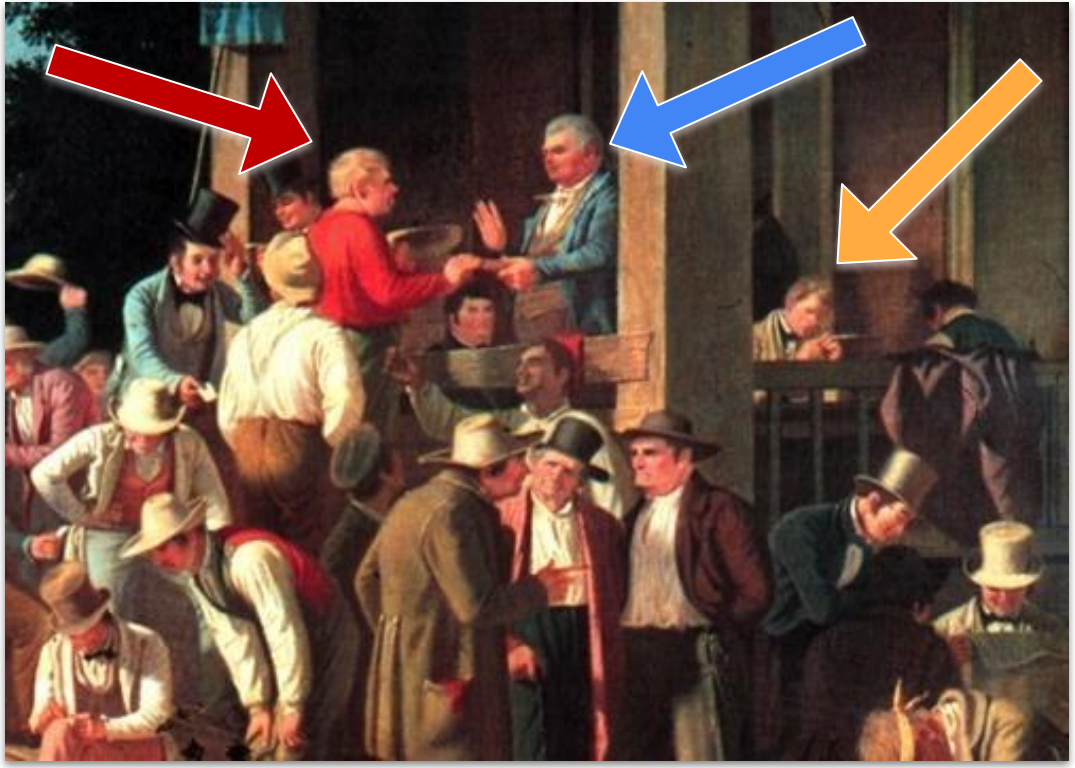


Voting Availability ◄- - - - - ► **Voter Authentication**

Early Voting Technology

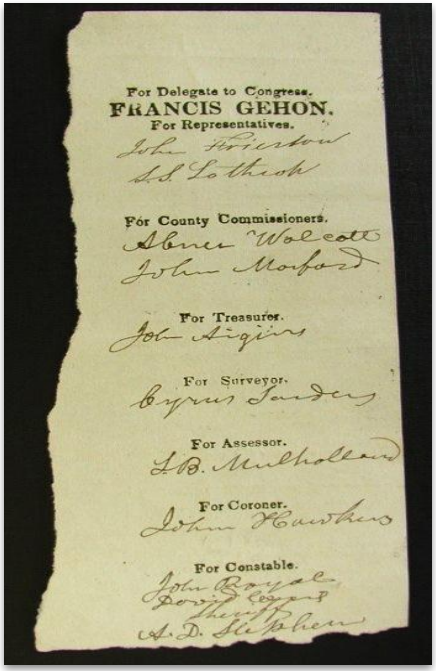
Voice Voting

Voice Voting

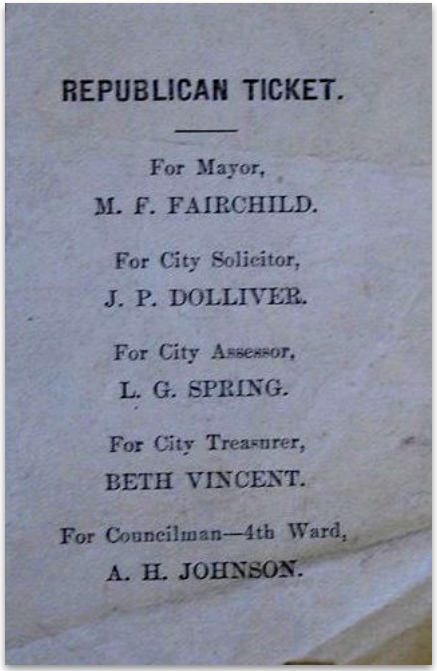


Voice Voting

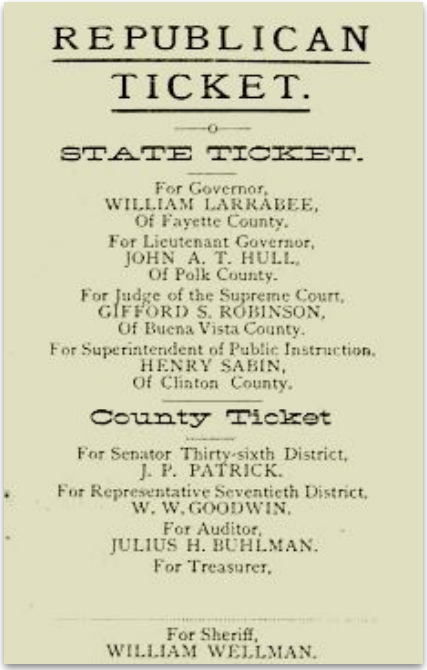
Voting by Ballots



1839

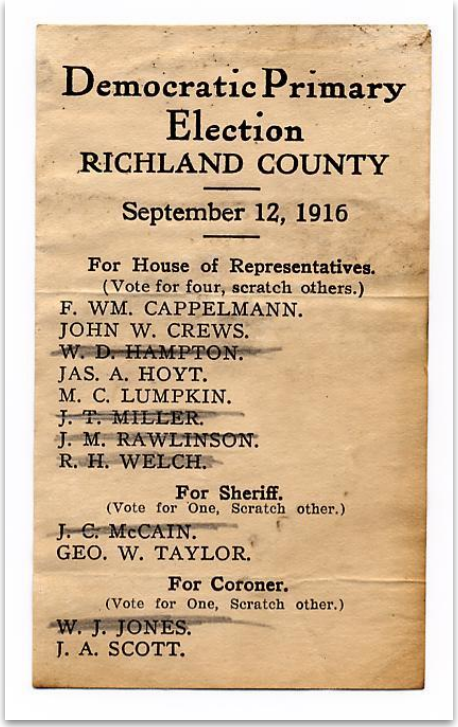


1880



1888

Voting by Ballots



Democratic Primary Election
RICHLAND COUNTY

September 12, 1916

For House of Representatives.
(Vote for four, scratch others.)

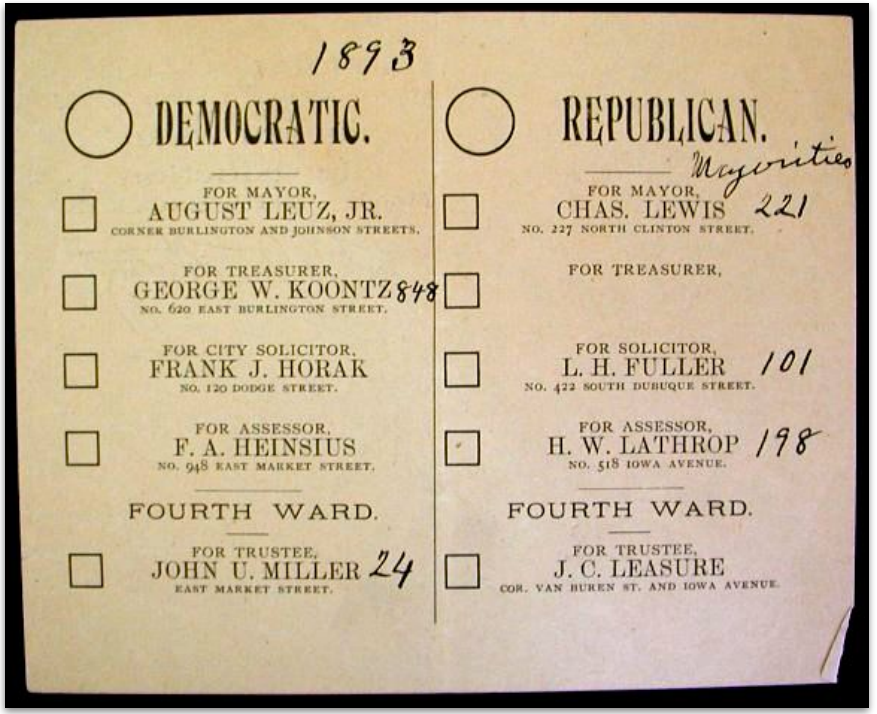
F. WM. CAPPELMANN.
JOHN W. CREWS.
W. D. HAMPTON.
JAS. A. HOYT.
M. C. LUMPKIN.
J. T. MILLER.
J. M. RAWLINSON.
R. H. WELCH.

For Sheriff.
(Vote for One, Scratch other.)

J. C. McCAIN.
GEO. W. TAYLOR.

For Coroner.
(Vote for One, Scratch other.)

W. J. JONES.
J. A. SCOTT.



1893

DEMOCRATIC.

FOR MAYOR,
AUGUST LEUZ, JR.
CORNER BURLINGTON AND JOHNSON STREETS.

FOR TREASURER,
GEORGE W. KOONTZ 848
NO. 620 EAST BURLINGTON STREET.

FOR CITY SOLICITOR,
FRANK J. HORAK
NO. 120 DODGE STREET.

FOR ASSESSOR,
F. A. HEINSIUS
NO. 948 EAST MARKET STREET.

FOURTH WARD.

FOR TRUSTEE,
JOHN U. MILLER 24
EAST MARKET STREET.

REPUBLICAN.

Majorities

FOR MAYOR,
CHAS. LEWIS 221
NO. 227 NORTH CLINTON STREET.

FOR TREASURER,

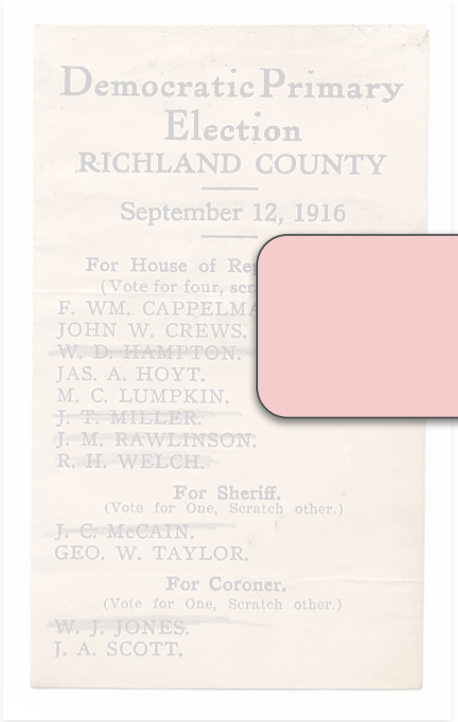
FOR SOLICITOR,
L. H. FULLER 101
NO. 422 SOUTH DUBUQUE STREET.

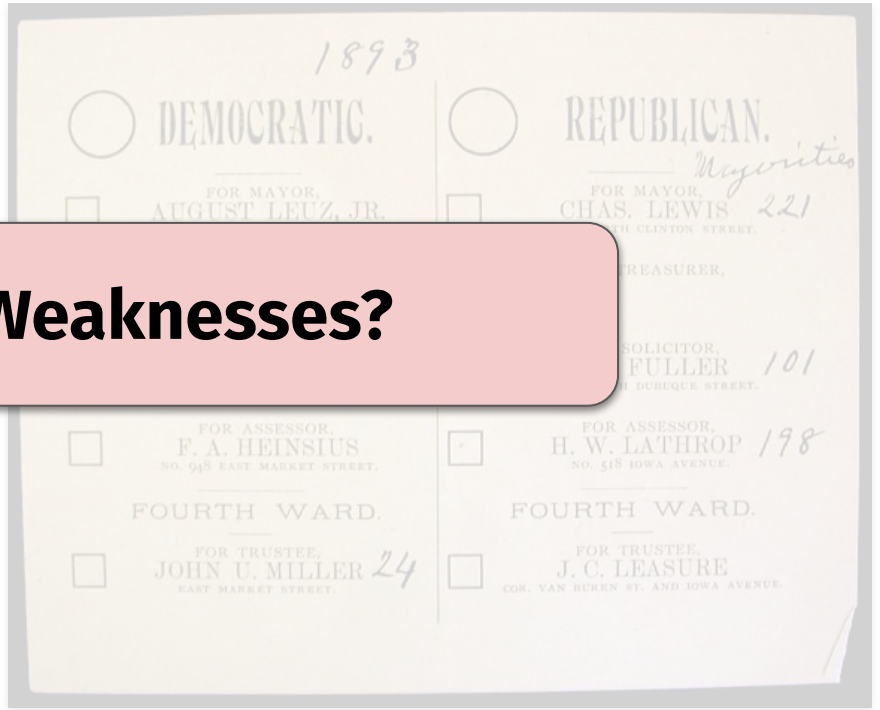
FOR ASSESSOR,
H. W. LATHROP 198
NO. 518 IOWA AVENUE.

FOURTH WARD.

FOR TRUSTEE,
J. C. LEASURE
COR. VAN BUREN ST. AND IOWA AVENUE.

Voting by Ballots





Weaknesses?

Ballot Boxes

Ballot Boxes

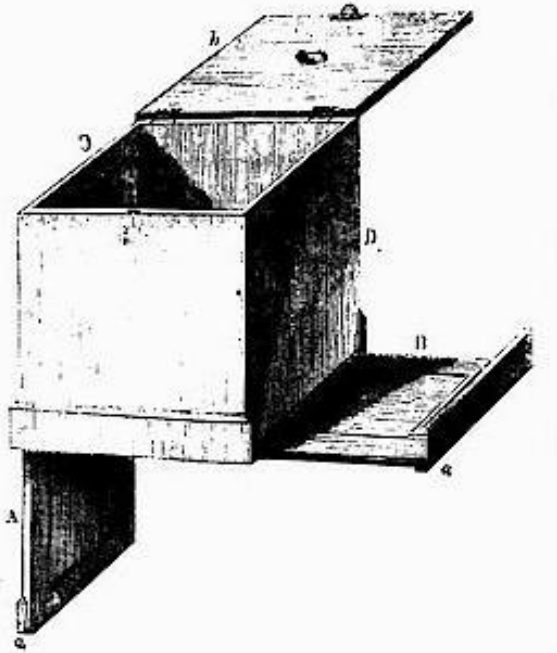
THE STUFFER'S BALLOT-BOX CLOSED UP.

STUFFER'S BALLOT-BOX.

WE give three views of the "Stuffer's Ballot-Box," which will give the reader a clear idea of the *modus operandi* of conducting the elections in San Francisco, and probably in some of our northern cities. The drawings were made from the box now in possession of the Vigilance Committee. It was from ballots taken from this box that Yankee Sullivan made out the election returns that secured Casey his office of Supervisor. The box is about two feet long and fourteen inches wide, and a foot deep, and painted on the outside a dark sky-blue color. It had moulding or cleets around the bottom, and at the top next the lid. The lock, which looked like an ordinary one, is so constructed that though it is worked with a key, it might also be opened by a peculiar pressure upon one side of the lid. There was an auger hole in the middle of the lid, and some of the wax with which it had been sealed at the closing of the polls when last used, was still remaining. It seems that the box was used last at a primary election in the Seventh ward, and the votes were still in it. On looking at the ballot-box, few would suspect the contrivances about it; but on further and minute examination it was found that it had a false bottom and a false side, sliding in grooves, under and behind which were packed quantities of spurious votes all ready for an election.

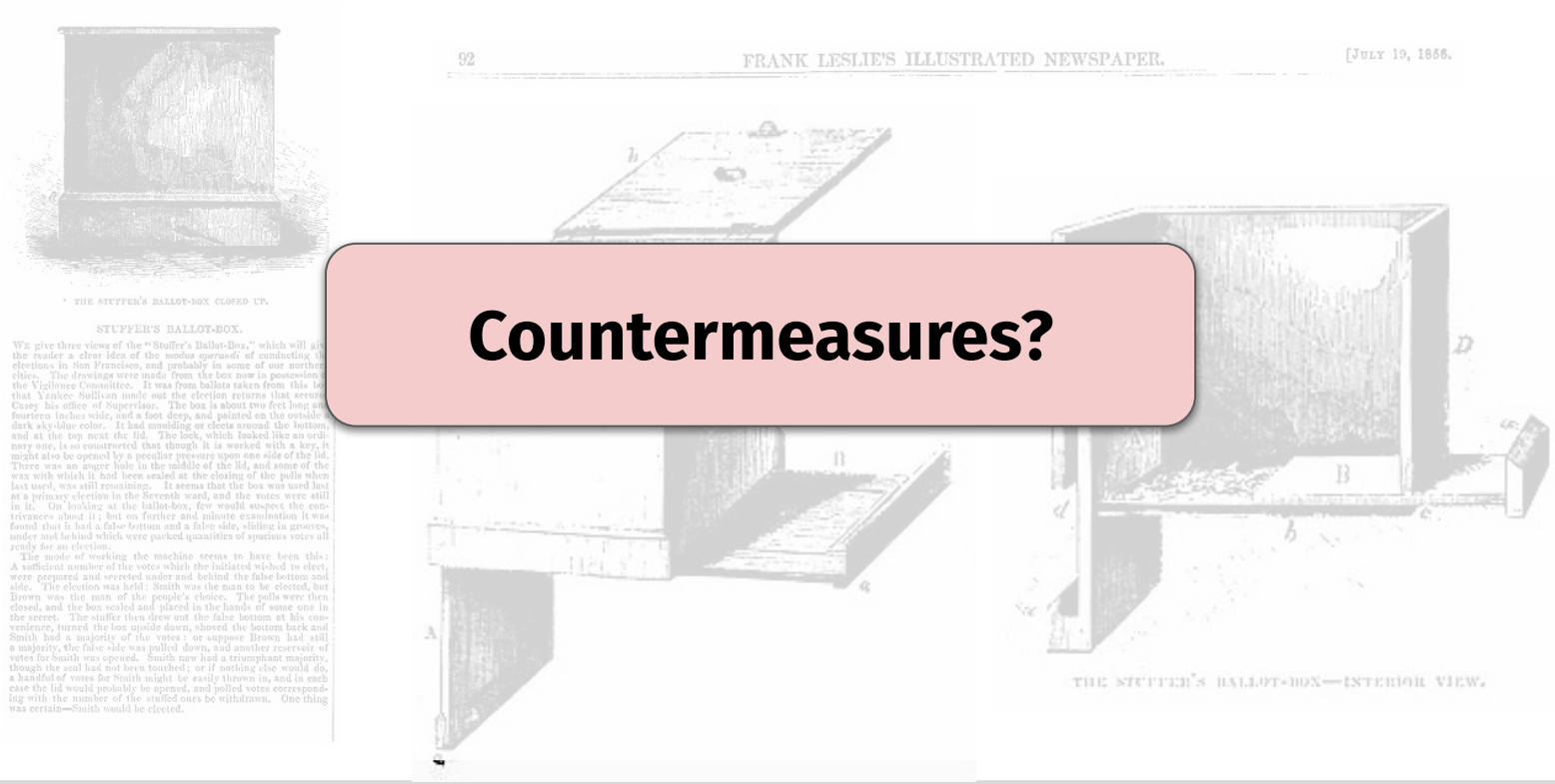
The mode of working the machine seems to have been this: A sufficient number of the votes which the initiated wished to elect, were prepared and secreted under and behind the false bottom and side. The election was held: Smith was the man to be elected, but Brown was the man of the people's choice. The polls were then closed, and the box sealed and placed in the hands of some one in the secret. The stuffer then drew out the false bottom at his convenience, turned the box upside down, shoved the bottom back and Smith had a majority of the votes: or suppose Brown had still a majority, the false side was pulled down, and another reservoir of votes for Smith was opened. Smith now had a triumphant majority, though the seal had not been touched; or if nothing else would do, a handful of votes for Smith might be easily thrown in, and in each case the lid would probably be opened, and polled votes corresponding with the number of the stuffed ones be withdrawn. One thing was certain—Smith would be elected.

92 FRANK LESLIE'S ILLUSTRATED NEWSPAPER. [JULY 19, 1856.



THE STUFFER'S BALLOT-BOX—INTERIOR VIEW.

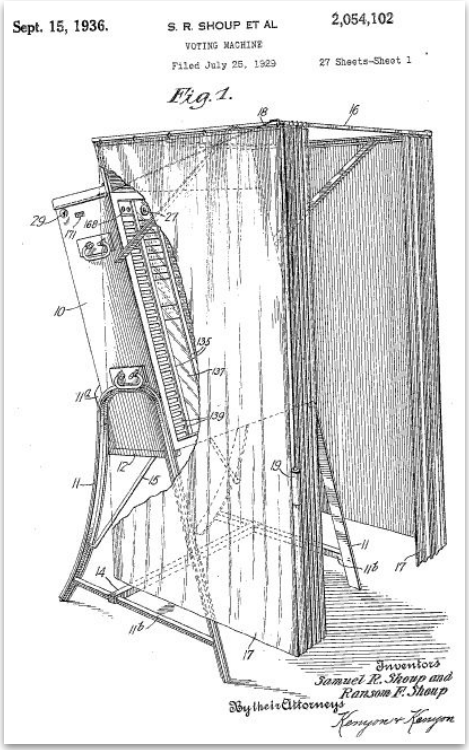
Ballot Boxes



Countermeasures?

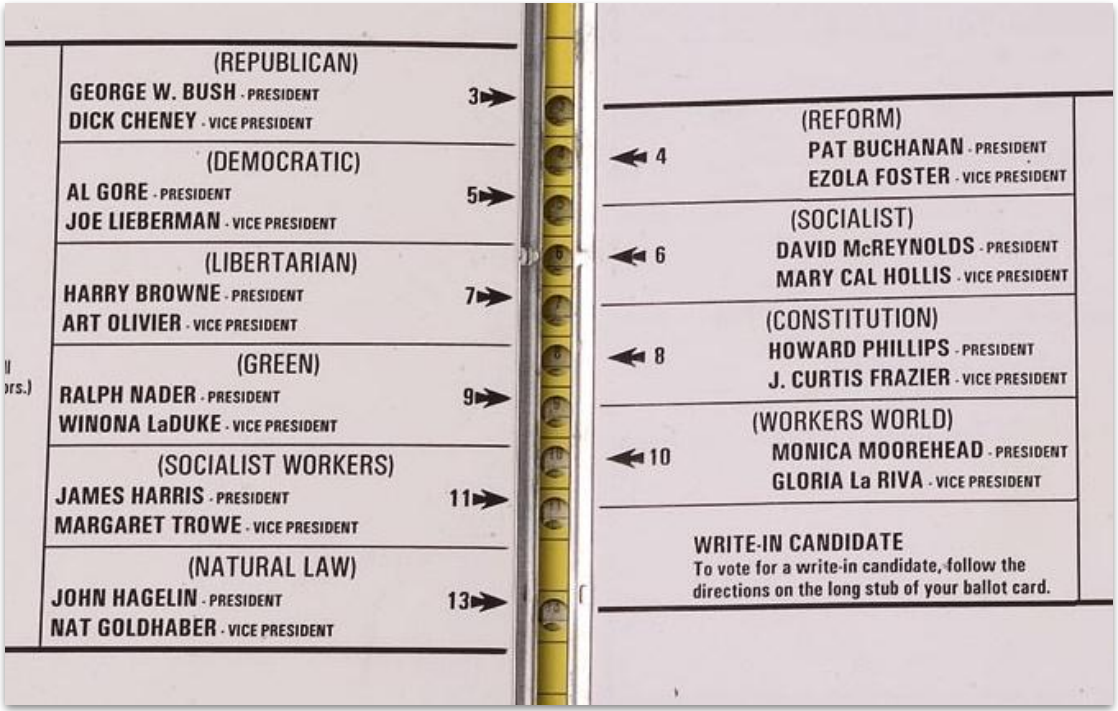
Ballot Boxes

Voting Machines

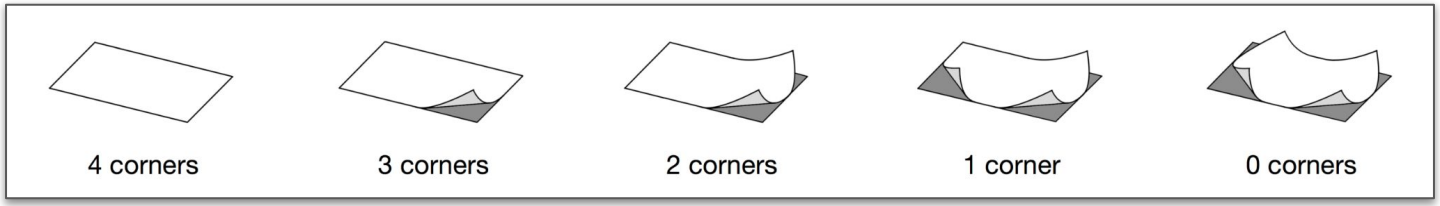


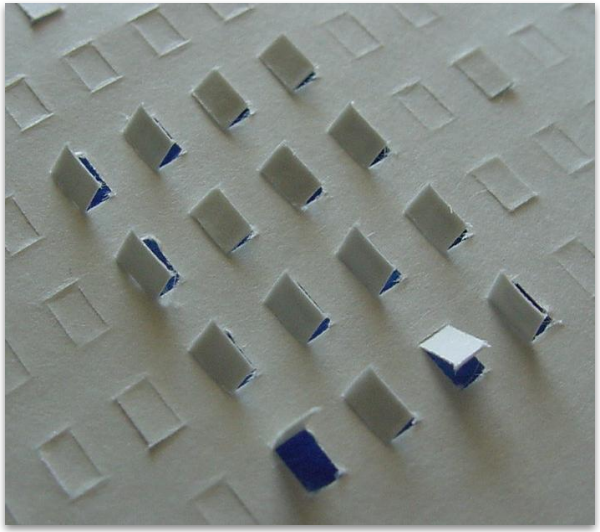
Voting Machines

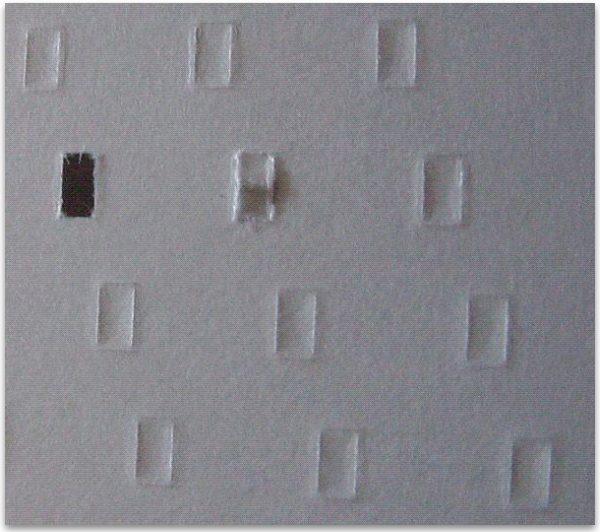
Voting Machines



Voting Machines

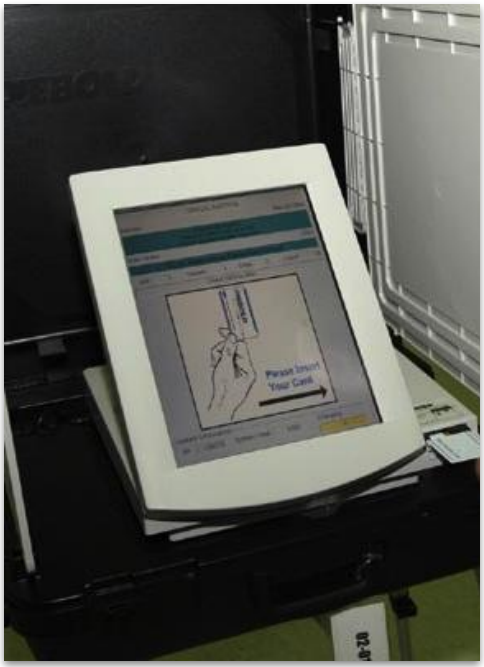






Computerized Voting

Early Computer-based Voting



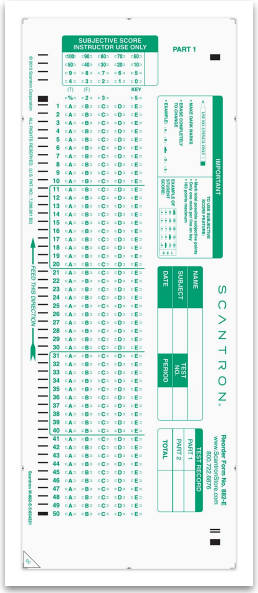
DRE Machine

Optical Scanner

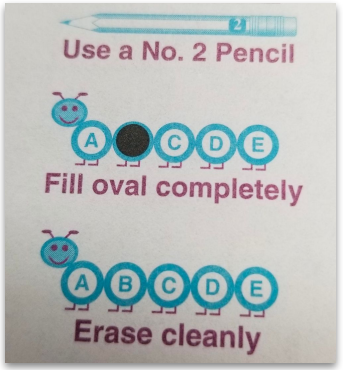
Optical Scanning

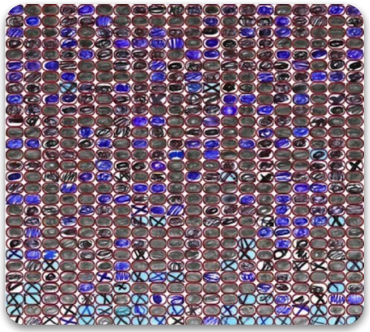
=

+

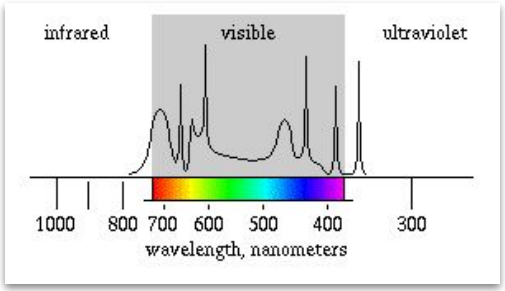


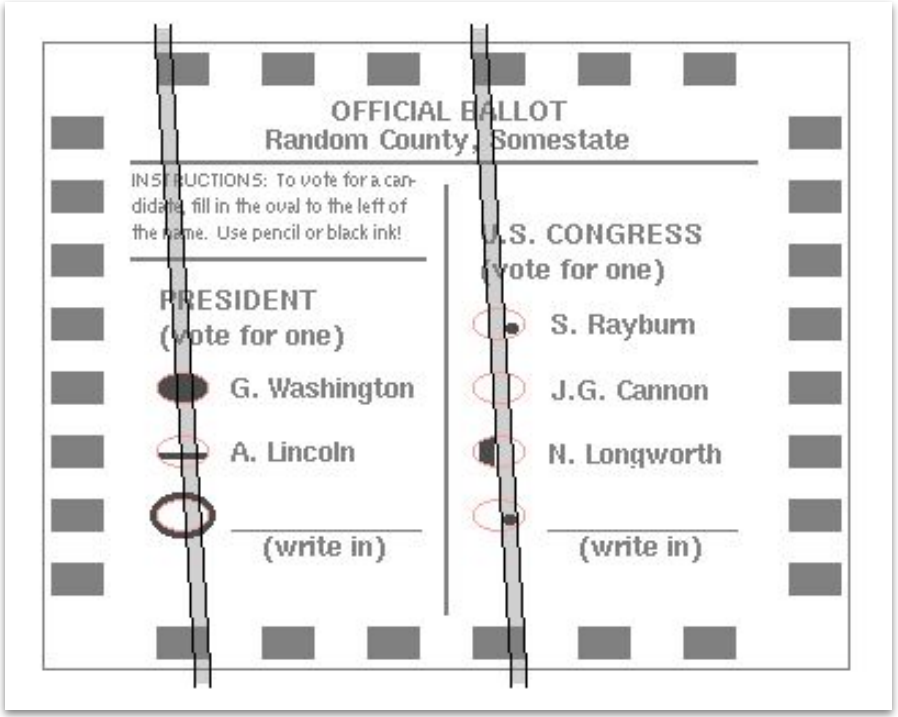
Optical Scanning





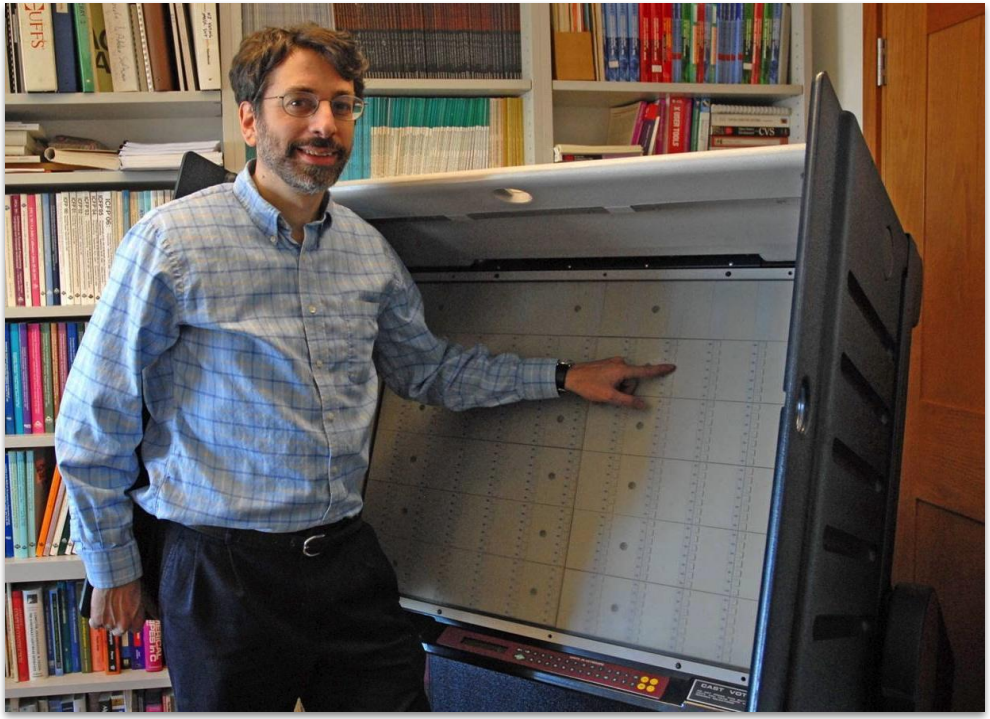
Optical Scanning



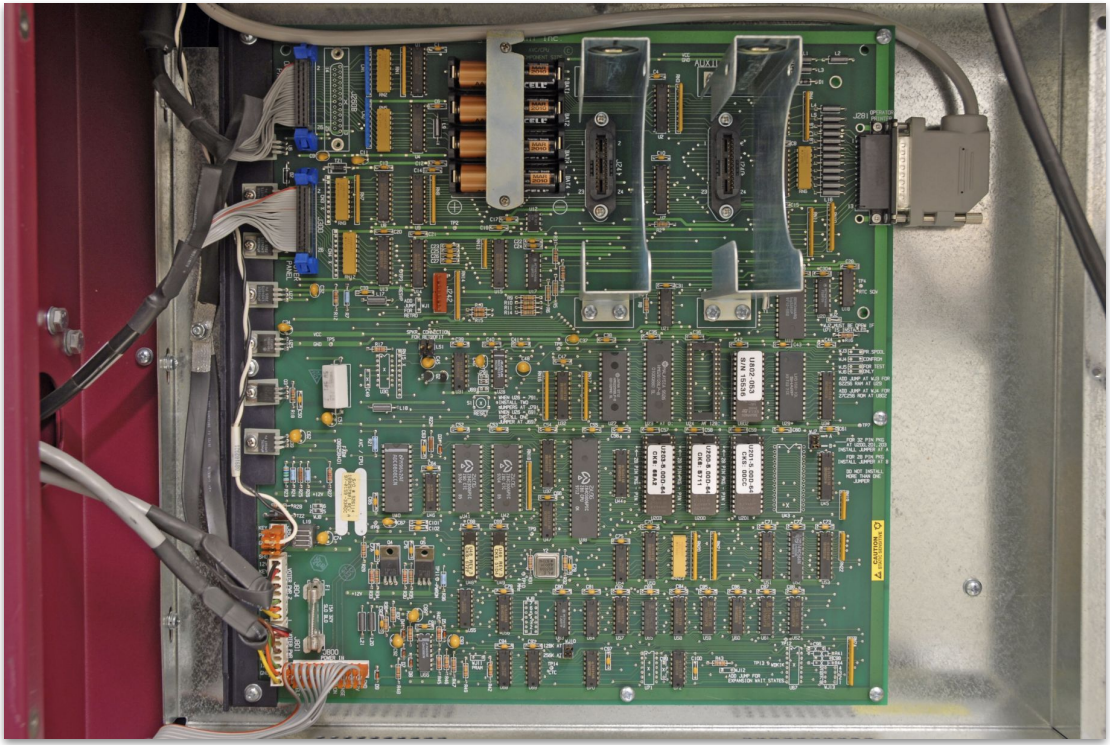


Attacks against Computerized Voting

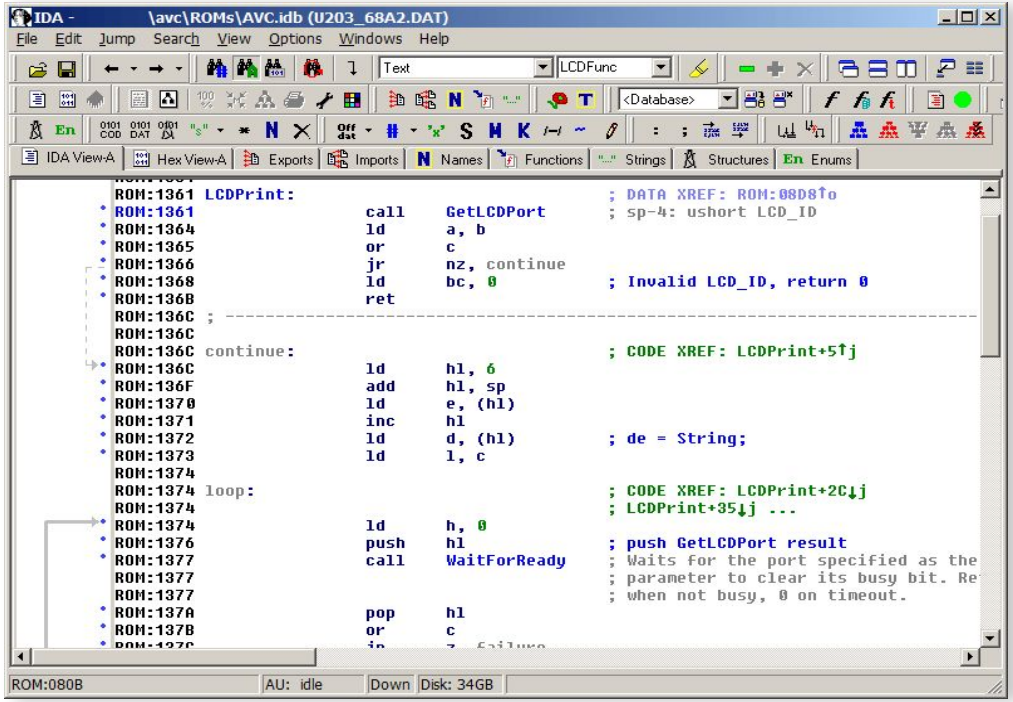
Sequoia AVC



Sequoia AVC



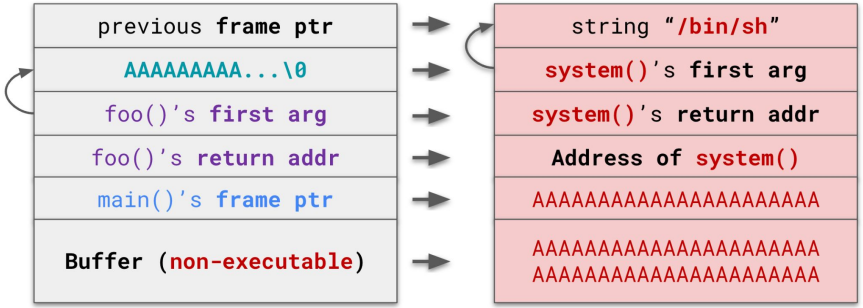
Attacking the Sequoia AVC



Attacking the Sequoia AVC

Return-oriented Programming (ROP)

Use code gadgets to achieve functionality



Can DREs Provide Long-Lasting Security?
The Case of Return-Oriented Programming and the AVC Advantage

Stephen Checkoway, *UC San Diego*
Ariel J. Feldman, *Princeton*
Brian Kantor, *UC San Diego*
J. Alex Halderman, *U Michigan*
Edward W. Felten, *Princeton*
Hovav Shacham, *UC San Diego*

Abstract

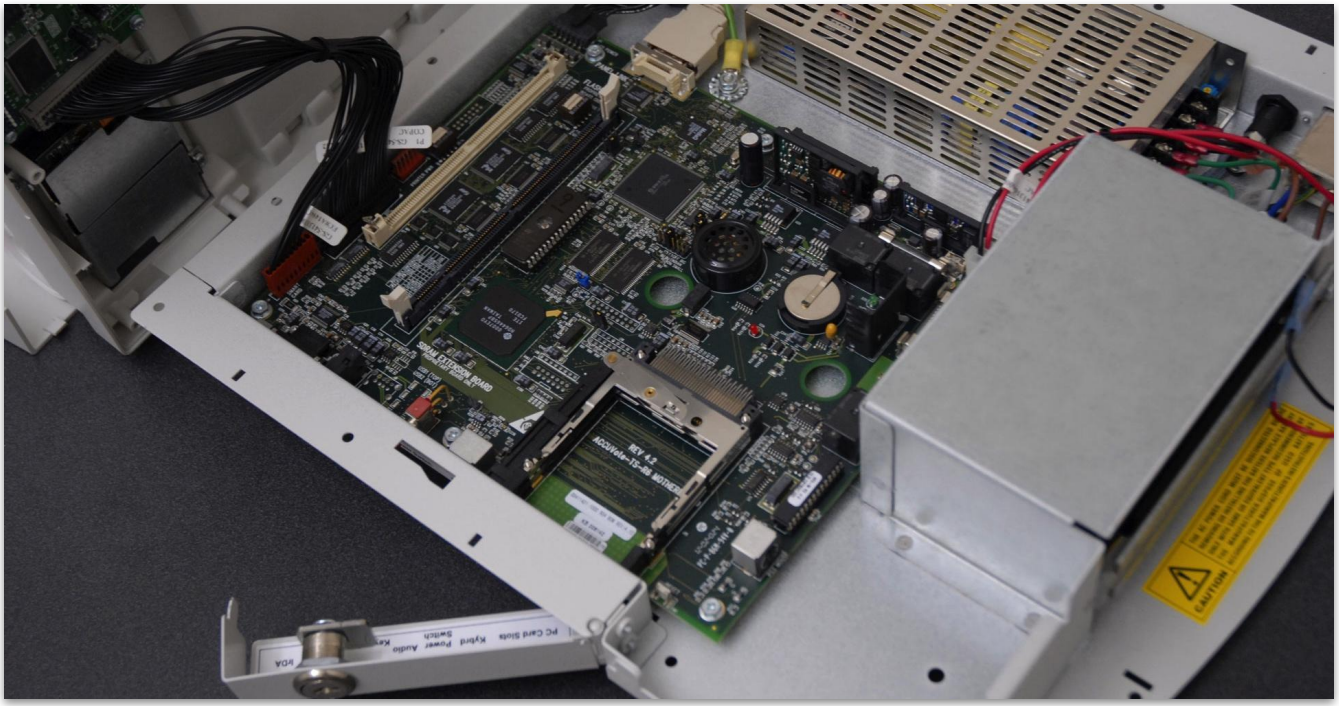
A secure voting machine design must withstand new attacks devised throughout its multi-decade service lifetime. In this paper, we give a case study of the long-term security of a voting machine, the Sequoia AVC Advantage, whose design dates back to the early 80s. The AVC Advantage was designed with promising security features: its software is stored entirely in read-only memory and the hardware refuses to execute instructions fetched from RAM. Nevertheless, we demonstrate that an attacker can induce the AVC Advantage to misbehave in arbitrary ways — including changing the outcome of an election — by means of a memory cartridge containing a specially-formatted payload. Our attack makes essential use of a recently-invented exploitation technique called *return-oriented programming*, adapted here to the Z80 processor. In return-oriented programming, short snippets of benign code already present in the system

The AVC Advantage voting machine we studied.

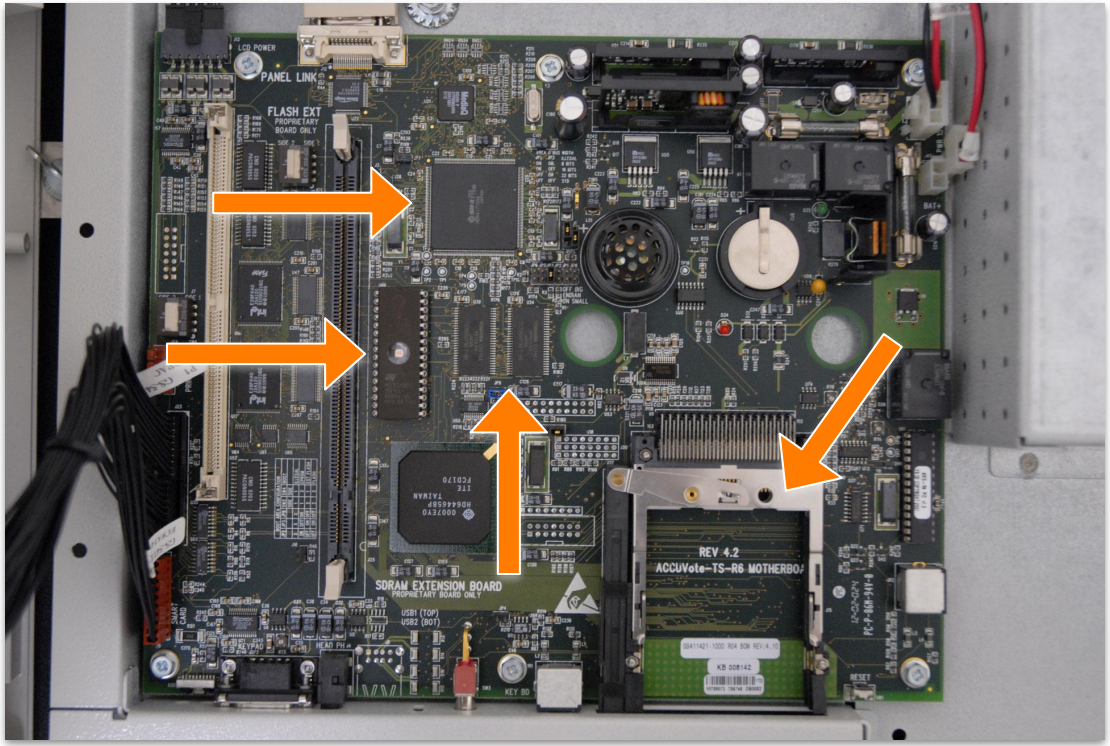
(which does not include the daughterboard) in machines decommissioned by Buncombe County, North Carolina, and purchased by Andrew Appel through a government auction site [2].

Diebold DRE

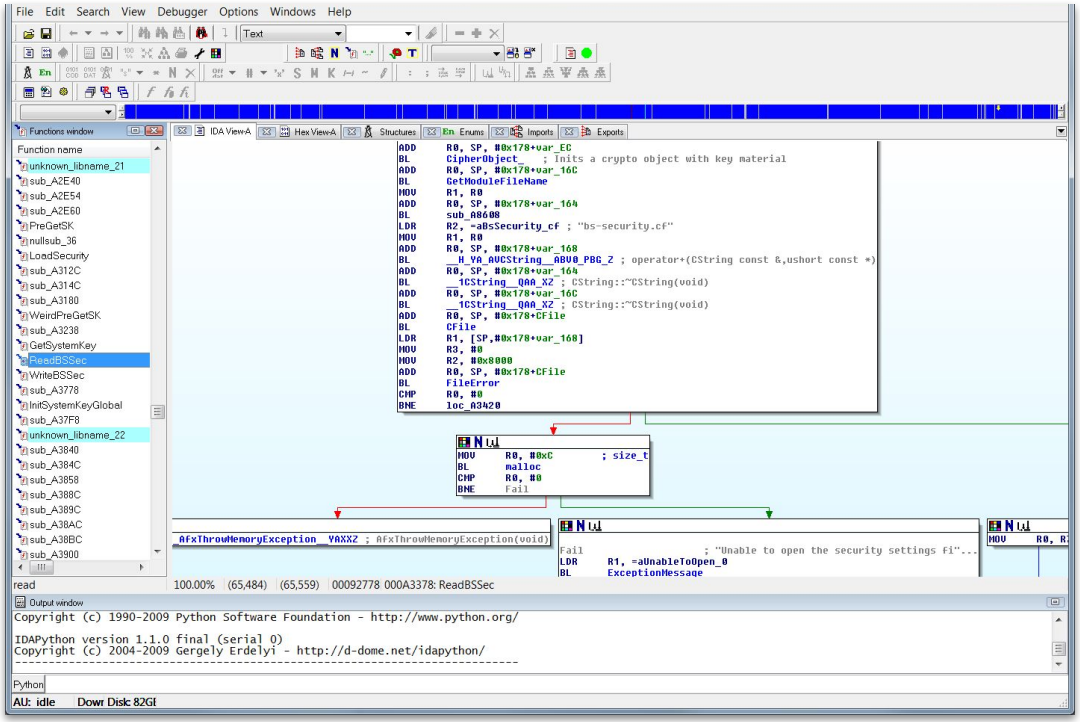
Reverse Engineering the Diebold DRE



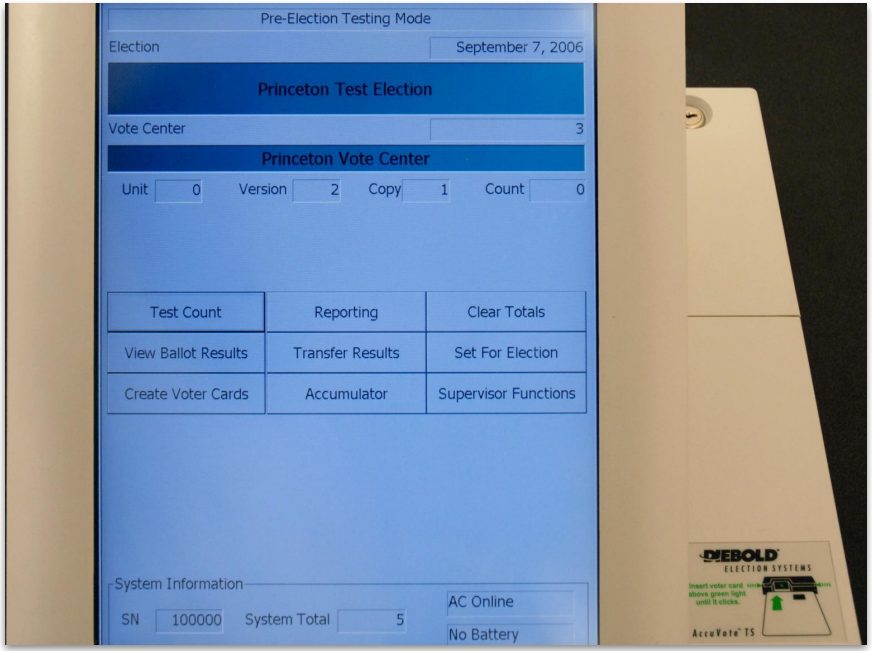
Reverse Engineering the Diebold DRE

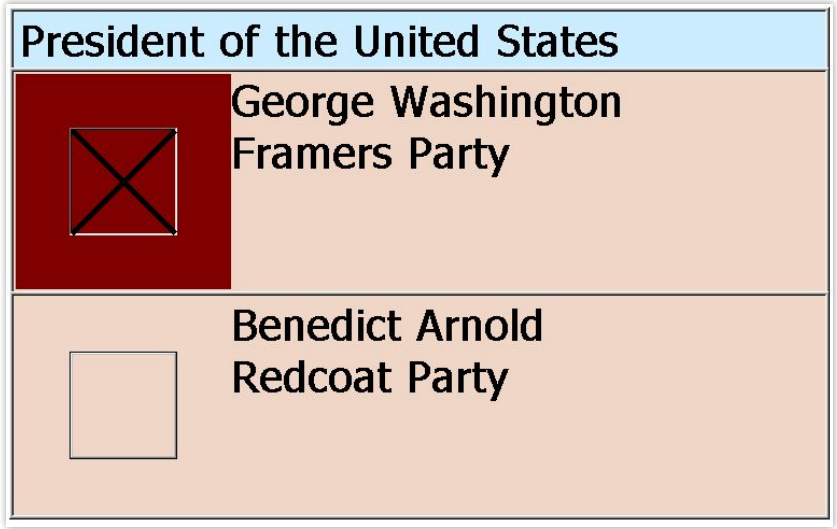


Reverse Engineering the Diebold DRE

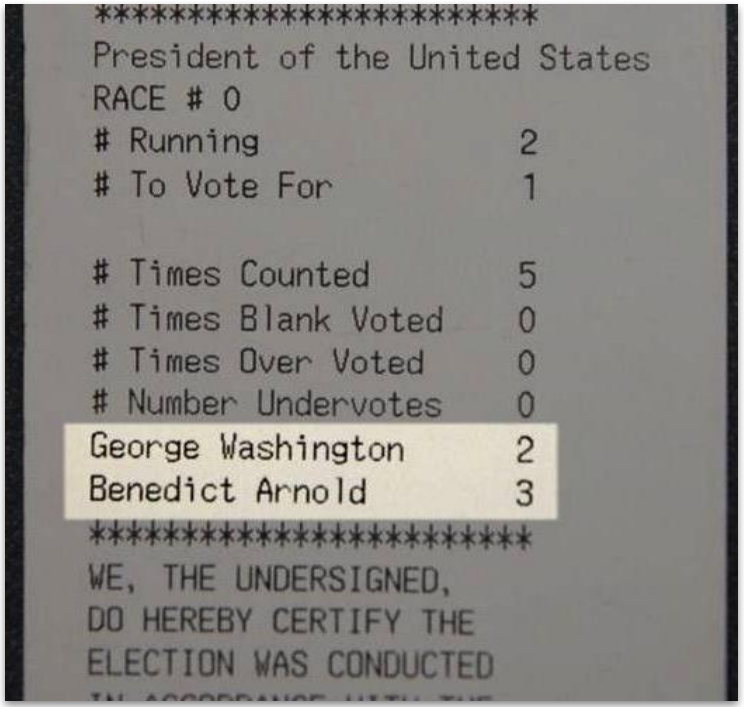


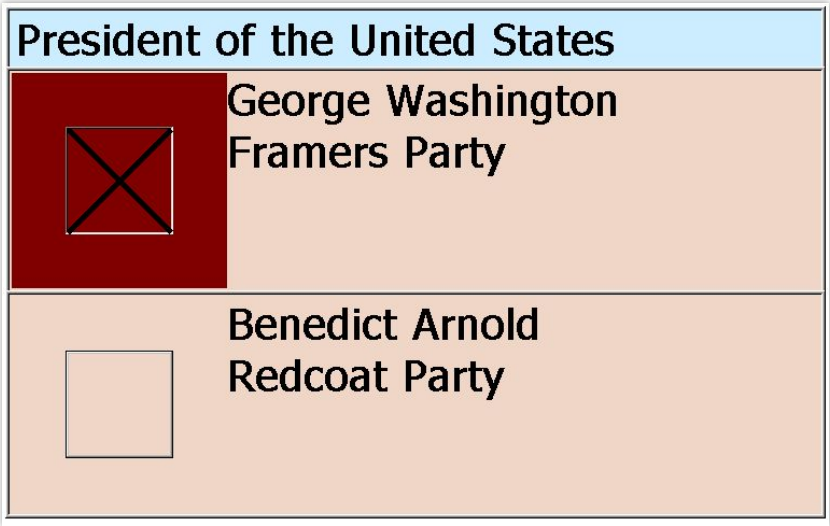
Attacking the Diebold DRE



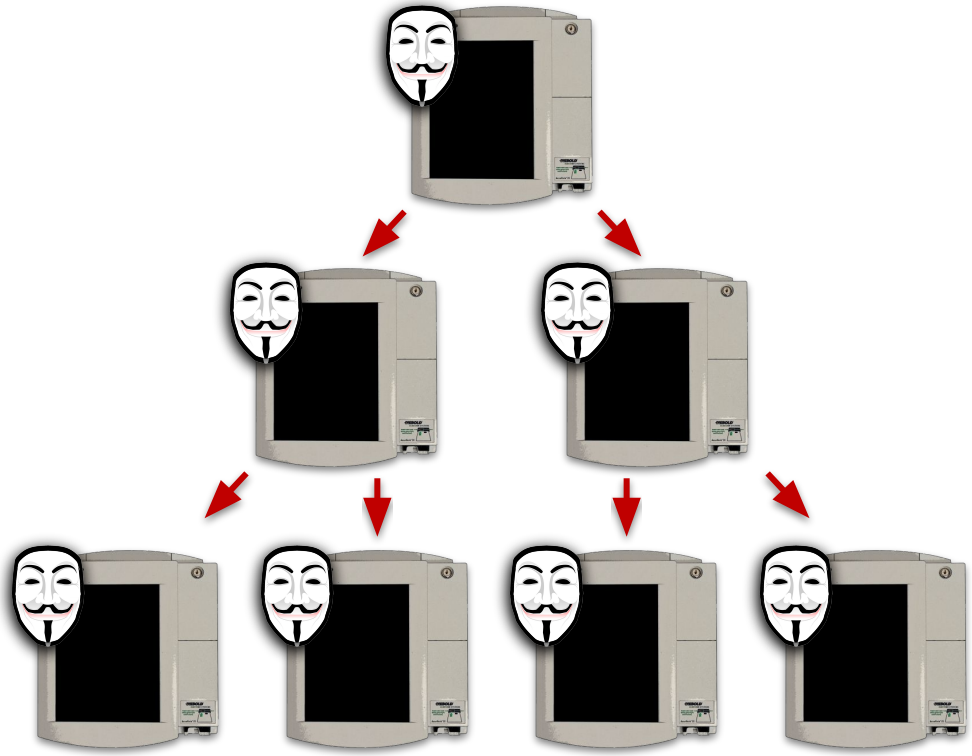


Attacking the Diebold DRE





Attacking the Diebold DRE



Attacking the Diebold DRE

Attacking the Diebold DRE

Attacking the Diebold DRE

Replacement Access Keys

- 2 keys that allow easy service access to the Tally Printer and replacement battery compartment

GS-567311-1000 **$5.90** USD per set
$6.90 CAD per set

Enter a quantity

add to your order ▸

ORDER BY PHONE 800.769.3246

Attacking the Diebold DRE

Security Analysis of the Diebold AccuVote-TS Voting Machine

Ariel J. Feldman*, J. Alex Halderman*, and Edward W. Felten*,†

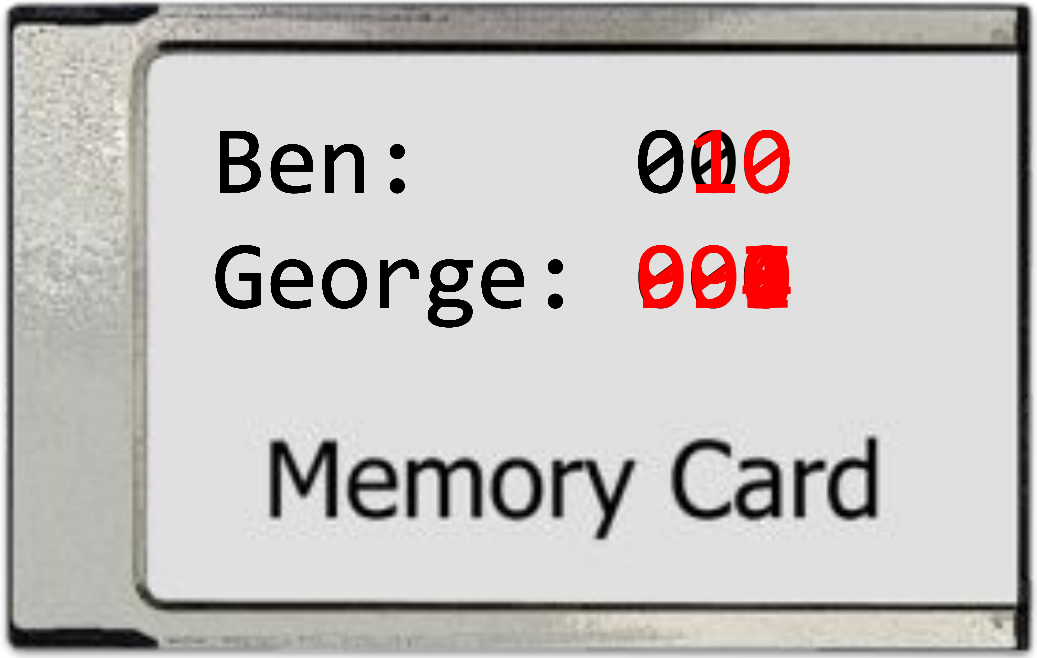
*Center for Information Technology Policy and Dept. of Computer Science, Princeton University
†Woodrow Wilson School of Public and International Affairs, Princeton University
{ajfeldma,jhalderm,felten}@cs.princeton.edu

September 13, 2006

Abstract

This paper presents a fully independent security study of a Diebold AccuVote-TS voting machine, including its hardware and software. We obtained the machine from a private party. Analysis of the machine, in light of real election procedures, shows that it is vulnerable to extremely serious attacks. For example, an attacker who gets physical access to a machine or its removable memory card for as little as one minute could install malicious code; malicious code on a machine could steal votes undetectably, modifying all records, logs, and counters to be consistent with the fraudulent vote count it creates. An attacker could also create malicious code that spreads automatically and silently from machine to machine during normal election activities—a voting-machine virus. We have constructed working demonstrations of these attacks in our lab. Mitigating these threats will require changes to the voting machine's hardware and software and the adoption of more rigorous election procedures.

Attacking the Diebold DRE



Attacking the Diebold DRE

Hursti Hack

Add languages

Article Talk | Read Edit View history Tools

From Wikipedia, the free encyclopedia

The **Hursti Hack** was a successful attempt to alter the votes recorded on a Diebold optical scan voting machine. The hack is named after Harri Hursti.

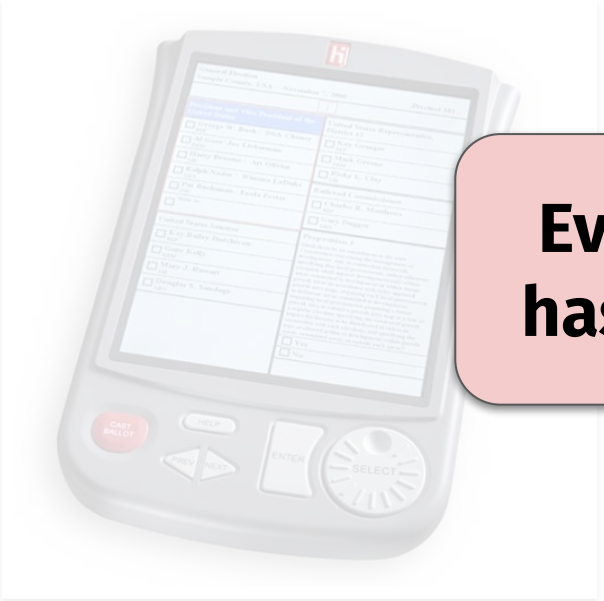
Participants [edit]

The participants were:

- Ion Sancho, Supervisor of Elections, Leon County, Florida.
- Thomas James, Information Systems Officer for Leon County, Florida
- Bev Harris, Black Box Voting founder
- Kathleen Wynne, Black Box Voting Associate Director
- Harri Hursti, computer programmer and security expert
- Hugh Thompson, application security expert and Ph.D. in math
- Susan Bernecker, former Republican candidate for New Orleans city council.
- Susan Pynchon, Director of Florida Fair Elections Coalition

Other Machines

Other Machines



Every examined machine has critical security flaws!

Other Machines

LILY HAY NEWMAN SECURITY 08.28.18 11:04 AM

VOTING MACHINES ARE STILL ABSURDLY VULNERABLE TO ATTACKS

BILL CLARK/GETTY IMAGES

WHILE RUSSIAN INTERFERENCE operations in the 2016 US presidential elections focused on misinformation and targeted hacking, officials have scrambled ever since to shore up the nation's vulnerable election infrastructure. New research, though, shows they haven't done nearly enough, particularly when it comes to voting machines.

Voting Machine Manual Instructed Election Officials to Use Weak Passwords

A vendor manual for voting machines used in about ten states shows the vendor instructed customers to use trivial, easy to crack passwords and to re-use the passwords when changing log-in credentials.

SHARE TWEET

Image: Shutterstodck

States and counties have had two years since the 2016 presidential election to educate themselves about security best practices and to fix security vulnerabilities in their election systems and processes. But despite widespread concerns about election interference from state-sponsored hackers in Russia and elsewhere, apparently not everyone received the memo about security, or read it.

An election security expert who has done risk-assessments in several states since

Latest

The Socialist Memelords Radicalizing Instagram

16 minutes ago

This Guy Wants to Open a DIY Tesla Repair Shop

an hour ago

Scientists Found Antibiotic-Resistant Bacteria In Space

2 hours ago

Supreme Court Weighs Whether Apple's App Store Is

Internet-based Voting

- **Is this safe?**

Risks of internet-based voting?

Nobody has responded yet.

Hang tight! Responses are coming in.

Internet-based Voting

- **Is this safe?**

Web Vulnerabilities

Malware

Fraudsters

Denial of Service

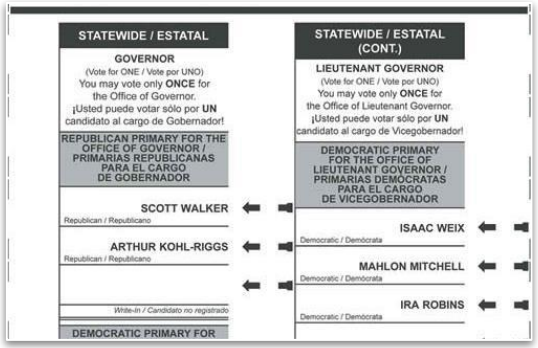
Internet-based Voting

- Is this safe?

Post-election Auditing

Post-election Auditing

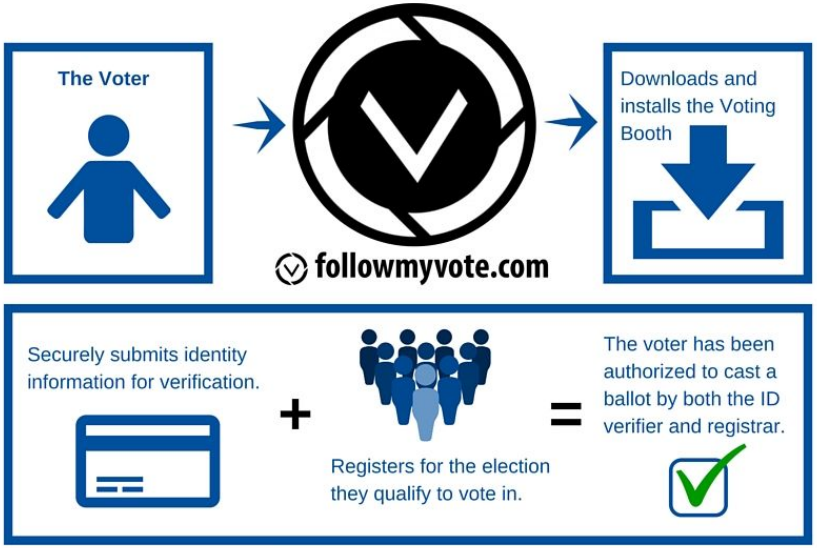
Redundancy + multiple failure modes = greater security

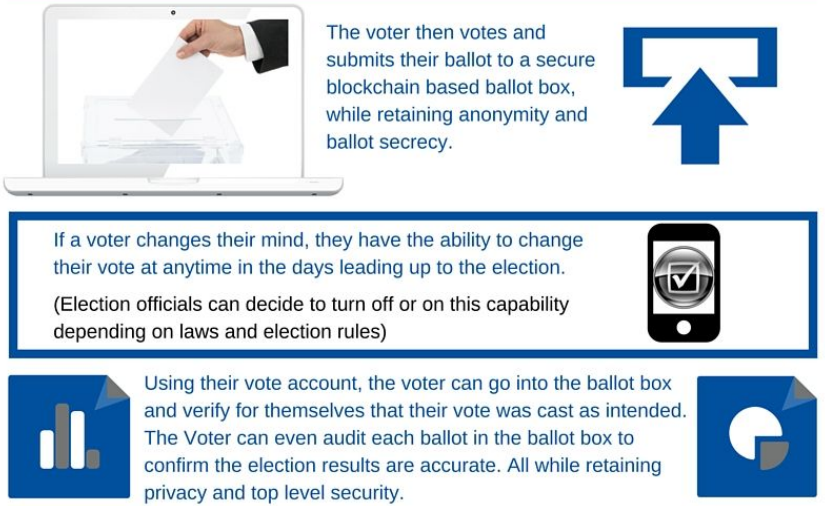


But... redundancy only helps if we use both records!

Post-election Auditing

- **Better ideas?**





Questions?

Next time on CS 4440...

Today's Security Ecosystem
Bug Bounties, CTF Competitions
Career Paths in Cyber Security