Outline

Elections and their security
- System security of electronic voting
- Announcements intermission
- End-to-end verification
- More Unix access control

Elections as a challenge problem
- Elections require a tricky balance of openness and secrecy
- Important to society as a whole
  - But not a big market
- Computer security experts react to proposals that seem insecure

History of (US) election mechanisms
- For first century or so, no secrecy
  - Secret ballot adopted in late 1800s
- Punch card ballots allowed machine counting
  - Common by 1960s, as with computers
  - Still common in 2000, decline thereafter
- How to add more technology and still have high security?

Election integrity
- Tabulation should reflect actual votes
  - No valid votes removed
  - No fake votes inserted
- Best: attacker can’t change votes
- Easier: attacker can’t change votes without getting caught

Secrecy, vote buying and coercion
- Alice’s vote can’t be matched with her name (unlinkable anonymity)
- Alice can’t prove to Bob who she voted for (receipt-free)
- Best we can do to discourage:
  - Bob pays Alice $50 for voting for Charlie
  - Bob fires Alice if she doesn’t vote for Charlie
Election verifiability

- We can check later that the votes were tabulated correctly
- Alice, that her vote was correctly cast
- Anyone, that the counting was accurate
- In paper systems, “manual recount” is a privileged operation

Politics and elections

- In a stable democracy, most candidates will be “pro-election”
- But, details differ based on political realities
- “Voting should be easy and convenient”
  - Especially for people likely to vote for me
- “No one should vote who isn’t eligible”
  - Especially if they’d vote for my opponent

Errors and Florida

- Detectable mistakes:
  - Overvote: multiple votes in one race
  - Undervote: no vote in a race, also often intentional
- Undetectable mistakes: vote for wrong candidate
- 2000 presidential election in Florida illustrated all these, “wake-up call”

Precinct-count optical scan

- Good current paper system, used here in MN
- Voter fills in bubbles with pen
- Ballot scanned in voter’s presence
  - Can reject on overvote
- Paper ballot retained for auditing

Vote by mail

- By mail universal in Oregon and Washington
  - Many other states have lenient absentee systems
  - Some people are legitimately absent
- Security perspective: makes buying/coercion easy
  - Doesn’t appear to currently be a big problem

Vote by web?

- An obvious next step
- But, further multiplies the threats
- No widespread use in US yet
- Unusual adversarial test in D.C. thoroughly compromised by U. Michigan team
DRE (touchscreen) voting

“Direct-recording electronic”: basically just a computer that presents and counts votes

In US, touchscreen is predominant interface
  - Cheaper machines may just have buttons
  - Simple, but centralizes trust in the machine

Adding an audit trail

VVPAT: voter-verified paper audit trail

DRE machine prints a paper receipt that the voter looks at

Goal is to get the independence and verifiability of a paper marking system

Outline

Elections and their security
System security of electronic voting
Announcements intermission
End-to-end verification
More Unix access control

Trusted client problem

Everything the voter knows is mediated by the machine
  - (For Internet or DRE without VVPAT)
  - Must trust machine to present and record accurately
    - A lot can go wrong
      - Especially if the machine has a whole desktop OS inside
      - Or a bunch of poorly audited custom code

Should we use DRE at all?

One answer: no, that’s a bad design

More pragmatic: maybe we can make this work
  - DREs have advantages in cost, disability access
  - If we implemented them well, they should be OK
  - Challenge: evaluating them in advance

US equipment market

Voting machines are low volume, pretty expensive

But jurisdictions are cost-conscious

Makers are mostly small companies
  - One was temporarily owned by the larger Diebold

Big market pressures: regulations, ease of administration
Security ecosystem

- Voting fraud appears to be very rare
  - Few elections worth stealing
  - Important ones are watched closely
  - Stiff penalties deter in-US attackers
- Downside: No feedback from real attacks
- Main mechanism is certification, with its limitations

Diebold case study

- Major manufacturer in early 2000s
  - During a post-2000 purchasing boom
  - Since sold and renamed
- Thoroughly targeted by independent researchers
  - Impolitic statement, blood in the water
- Later state-authorized audits found comprehensive problems
  - Your reading: from California

Physical security

- Locked case; cheap lock as in hotel mini-bar
- Device displays management menu on detected malfunction
  - Can be triggered in booth by unspecified use of paperclip
- Tamper-evident seals? Not a strong protection

Buffer overflows, etc.

- Format string vulnerability
  - "Page %d of %d"
- Was this audited?

```
TCHAR name;
_stprintf(&name, _T("\Storage Card\%s"), findData.cFileName);
```

Web-like vulnerabilities

In management workstation software:

- SQL injection
- Authentication logic encoded only in enabled/disabled UI elements
  - E.g., buttons grayed out if not administrator
  - Not quite as obviously wrong as in web context
  - But still exploitable with existing tools

OpenSSL mistakes

- Good news: they used OpenSSL
- Bad news: old, buggy version
- Insufficient entropy in seeding PRNG
  - Good interface from desktop Windows missing in WinCE
- Every device ships with same certificate and password
Election definitions

- Integrity "protected" by unkeyed, non-crypto checksum
- Can change bounding boxes for buttons
  - Without changing checksum!
- Can modify candidate names used in final report
  - E.g. to fix misspelling; security implication mentioned in comment

Secrecy problems

- Limited, since the DRE doesn’t see registration information
- But, records timestamp and order of voting
- Could be correlated with hidden camera or corrupted poll worker

Voting machine viruses

- Two-way data flow between voting and office machines
- Hijacking vuln’s in software on both sides
- Can write virus to propagate between machines
- Leverage small amount of physical access

Subtle ways to steal votes

- Change a few votes your way, revert if the voter notices
  - Compare: flip coin to split lunch
- Control the chute for where VVPAT receipts go
- Exchange votes between provisional and regular voters

Outline

- Elections and their security
- System security of electronic voting
- Announcements intermission
- End-to-end verification
- More Unix access control

BCLPR design changes

- Avoid unnecessary changes to benign functionality
  - Restricting length or character sets of arguments
  - Though, what is the benign functionality?
- Not a great candidate for privilege separation
  - 500 LOC is not large! (LPRNG 45k, CUPS 371k)
Exercise set 2 due tonight

By 11:55pm, as usual

Midterm exam Tuesday

- Usual location, starting promptly at 2:30
- Open books/notes/printouts, no computers or other electronics
- Sample exam w/solutions (2013’s) posted

Guest lecture on SFI

- Next Thursday (I’ll be out of town)
- Software-based fault isolation
- Conceptually related to part 1, exercise/exam coverage in part 2

Outline

- Elections and their security
- System security of electronic voting
- Announcements intermission
- End-to-end verification
- More Unix access control

End-to-end integrity and verification

- Tabulation cannot be 100% public
- But how can we still have confidence in it?
- Cryptography to the rescue, maybe
  - Techniques from privacy systems, others
  - Adoption requires to be very usable

Commitment to values

- Two phases: commit, later open
  - Similar to one use of envelopes
- Binding property: can only commit to a single value
- Hiding property: value not revealed until opened
Randomized auditing

- How can I prove what’s in the envelope without opening it?
- \(\frac{1}{n}\) envelopes, you pick one and open the rest
  - Chance \(\frac{1}{n}\) of successful cheating
- Better protection with repetition

Election mix-nets

- Independent election authorities similar to Tor nodes
- Multi-encrypt ballot, each authority shuffles and decrypts
- Extra twist: prove no ballots added or removed, without revealing permutation
  - Instance of “zero-knowledge proof”
- Privacy preserved as long as at least one authority is honest

Pattern voting attack

- Widely applicable against techniques that reveal whole (anonymized) ballots
- Even a single race, if choices have enough entropy
  - 3-choice IRV with 35 candidates: 15 bits
- Buyer says: vote first for Bob, then 2nd and 3rd for Kenny and Xavier
  - Chosen so ballot is unique

Fun tricks with paper: visual crypto

- Want to avoid trusted client, but voters can’t do computations by hand
- Analogues to crypto primitives using physical objects
- One-time pad using transparencies:

Scantegrity II

- Designed as end-to-end add-on to optical scan system
- Fun with paper 2: invisible ink
- Single trusted shuffle
  - Checked by random audits of commitments

Outline

- Elections and their security
- System security of electronic voting
- Announcements intermission
- End-to-end verification
- More Unix access control
Special case: `/tmp`
- We'd like to allow anyone to make files in `/tmp`
- So, everyone should have write permission
- But don’t want Alice deleting Bob’s files
- Solution: “sticky bit” 01000

“POSIX” ACLs
- Based on a withdrawn standardization
- More flexible permissions, still fairly Unix-like
- Multiple user and group entries
  - Decision still based on one entry
- Default ACLs: generalize group inheritance
- Command line: `getfacl`, `setfacl`

ACL legacy interactions
- Hard problem: don’t break security of legacy code
  - Suggests: “fail closed”
- Contrary pressure: don’t want to break functionality
  - Suggests: “fail open”
- POSIX ACL design: old group permission bits are a mask on all novel permissions

“POSIX” “capabilities”
- Divide root privilege into smaller (~35) pieces
- Note: not real capabilities
- First runtime only, then added to FS similar to `setuid`
- Motivating example: `ping`
- Also allows permanent disabling

Privilege escalation dangers
- Many pieces of the root privilege are enough to regain the whole thing
  - Access to files as UID 0
  - `CAP_DAC_OVERRIDE`
  - `CAP_FOWNER`
  - `CAP_SYS_MODULE`
  - `CAP_MKNOD`
  - `CAP_PTRACE`
  - `CAP_SYS_ADMIN` (mount)
Legacy interaction dangers

- Former bug: take away capability to drop privileges
- Use of temporary files by no-longer setuid programs
- For more details: “Exploiting capabilities”, Emeric Nasi

Next time

- Good luck on the midterm!