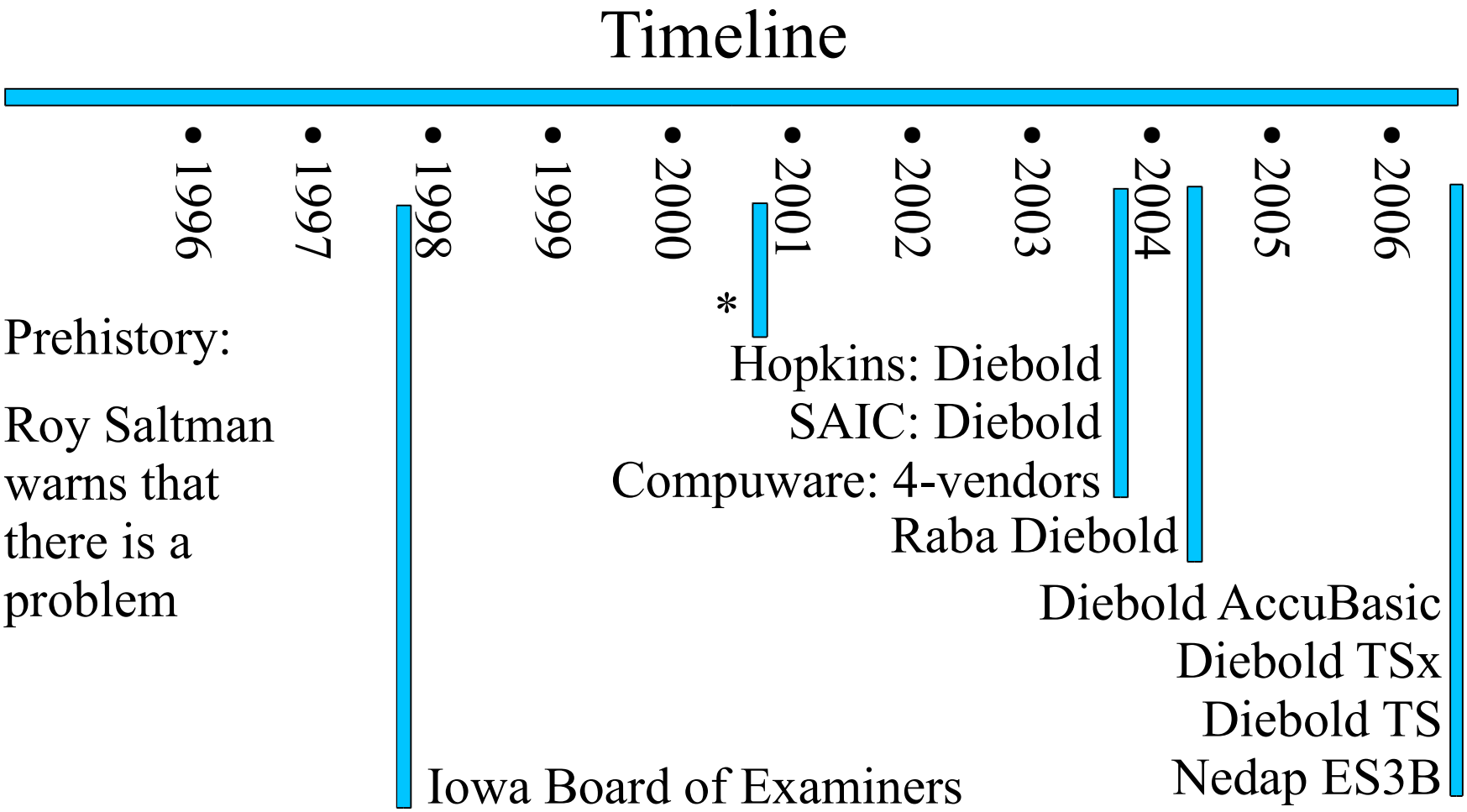
Voting Security
A Technical Perspective

Douglas W. Jones
THE UNIVERSITY OF IOWA
Department of Computer Science

A talk presented at the University of South Carolina
Cybersecurity Symposium
October 27, 2005, Columbia, South Carolina

Supported in part by NSF Grant CNS – 052439
A Center for Correct, Usable, Reliable, Auditable, and Transparent Elections
(ACCURATE)

Discovering the DRE Problem



1997, the Iowa Board of Examiners

"Dr. Jones also expressed concern about data encryption standards used to guarantee the integrity of the data on the machine. DES requires the use of electronic keys to lock and unlock all critical data. Currently all machines use the same key. Dr. Jones stated that this is a security problem. However, the use of a single key for all machines is not a condition that would disqualify the system under Iowa law." [Minutes, Nov. 6, 1997 meeting]

The Hopkins Report

- *Analysis of an Electronic Voting System*
  Kohno, Stubblefield, Rubin & Wallach,July 2003

Showed that Diebold code last used in late 2002:

- Still used DES
- Still set all DES keys to a hard-coded constant
- Had essentially no smart-card security

In sum, worse than I knew in 1997, and I thought the security of the system was incredibly naive.

The SAIC Report

- *Risk Assessment Report, Diebold AccuVote-TS*
  Science Applications Int'l Corp., September 2003

Commissioned by state of Maryland:

- A response to the Hopkins Report
- Released only in redacted form
- Confirms conclusions of Hopkins Report
- Shows that Maryland procedures were inadequate
- Offered constructive criticism of state and vendor

State response:

We're doing what they say.

The Compuware Report

- *DRE Technical Security Assessment Report*
  Compuware Corporation, November 2003

Commissioned by the State of Ohio

- Response to Hopkins Report
- Looked at Diebold, ES&S, Hart, Sequoia
- Found significant flaws in all systems examined
- Made constructive suggestions to vendors, state

State Response

- We're doing what they say

Problem: No long term solution, short term patches!

The Raba Report

- *Trusted Agent Report, Diebold AccuVote-TS*
  RABA Technologies LLC, January 2004

Commissioned by Maryland

- Followup on Hopkins/SAIC
- Things are even worse than they seemed
- Physical locks and keys terrible
- Short term fixes suggested for election 2004

State response:

- We are completely exhonerated

The VSTAAB Report

- *Security Analysis of ... Diebold AccuBasic ...*
  Wagner, Jefferson, Bishop, February 2006

Commissioned by the State of California

- A response to a report by Harri Hurst
- Shows that Diebold's coding practices are bad
- Serious buffer overflow vulnerabilities
- Serious potential of attack from PCMCIA cards

State Response:

- More tamper evident security seals

The Hursti II Report

- *Diebold TSx Evaluation – Security Alert*
  Harri Hursti, May 2006

Black Box Voting at invitation of a Utah County

- Diebold TSx "firmware" easy to change
- Proposes PCMCIA virus threat

The Princeton Report

- *Security Analysis of … Diebold AccuVote-TS …*
  Feldman, Halderman & Felton, September 200

Felton's group got a Diebold TS DRE machine
- Demonstrated Hursti's proposed virus
- Created demo virus that attacks election

Diebold Response
- There is no problem
- Polling place procedures defend us adequately

The Nedap Report

- *Nedap/Groenendaal ES3B ... a security analysis*
  Gonggrijp & Hengeveld, October, 2006

They got a Nedap ES3B voting machine

- Sold in US by Liberty Voting Solutions
- All machines are keyed alike
- They reverse engineered the memory cartridge
- They made a fraudulent cartridge

No response from vendor yet [that I have seen]

Diagnosis

The status quo today is
Security through obscurity

Every time any vendor's implementation has been subject to close examination, serious flaws have been found.

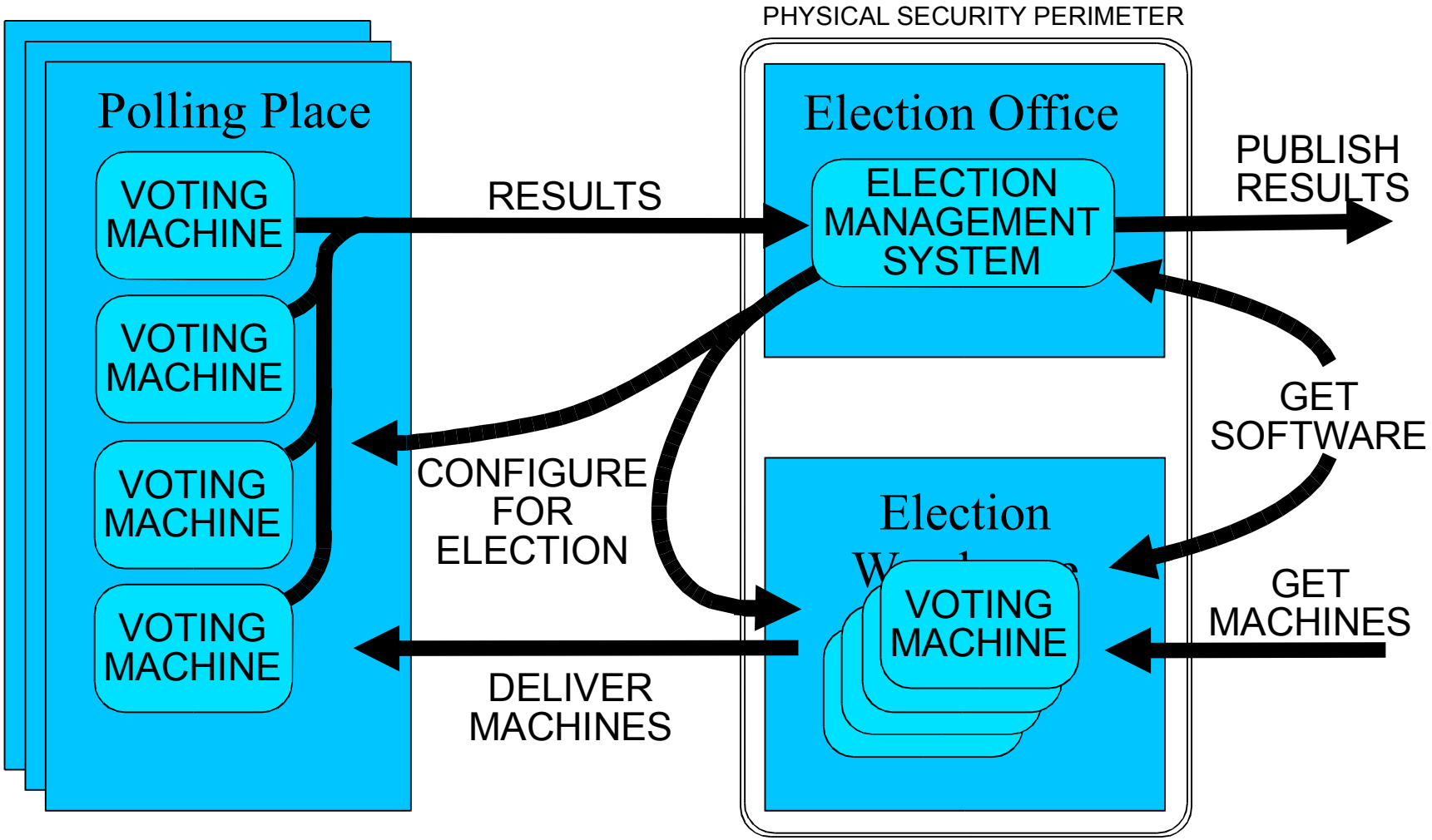
In fact, when proprietary documents of vendors are examined, they also show evidence of very low awareness of issues.

A System Perspective

What the voting system industry lacks

- Examine all inputs and outputs to system
- Identify potential attacks to all of them
- Construct appropriate defenses for all
- Do not permit any undocumented data paths

An Information Flow Perspective



Voting System Certification Process

- Independent Testing Labs Certify systems to Federal Standards
- Standards set by
  - Federal Election Commission, 1990, 2002
  - Election Assistance Commission, 2005
- States may set additional standards
  - New York's new standards look good, *on paper*
  - Many states consider Federal standards enough

But Why?

- Why have vendors delivered sub-par systems?
  - Why are vendors not held responsible for flaws?
- Why have ITAs approved sub-par systems?
  - Why are ITAs not held responsible for oversights?
- Why have states accepted sub-par systems?
- Why do counties brush off reports of problems?

Regulatory Capture

- "Gamekeeper turns poacher or, at least, helps poacher." [The Economist]
- Richard Posner of the University of Chicago argued that "REGULATION is not about the public interest at all, but is a process, by which interest groups seek to promote their private interest... Over time, regulatory agencies come to be dominated by the industries regulated."
- Voting system vendors have always played at this

Election Official Buy-In

- Once you spend public money on something, you cannot afford to be wrong.
- If you are tied to a single source for a decade, you will avoid asking hard questions.
- Public confidence in elections is very important, so by all means, keep all criticism private.

The result? Election officials are predisposed to:

- Believe what the vendors tell them.
- Discount what critics say.

Some Hard Facts

Security people need to *emphasize* hard truths:

- Hardware verification is impossible
- Software verification is impossible
- Malware detection is impossible

Reliance on procedural defenses is doomed

- Computer Security is a pretty grim business!

Hardware Verification

Is the hardware delivered the hardware specified?

- Can answer by reverse engineering

Opaque components make this hard

- ICs, multilayer circuit boards, etc.

Programmable components make this hopeless

- FPGAs are a killer – how can you prove content?

Software Verification

Is the software delivered the software specified?

- Can answer by reverse engineering object code
- Cannot answer by source code analysis

Opaque components prevent this

- Single chip microcontrollers with copy protect

Crypto authentication

- Only helps if the authenticator is trusted!

Malware Detection

Theorem: There cannot exist an effective* malware detection algorithm.

Proof:

* No false positives
No false negatives

- Let *A* be such an algorithm
- Construct program *P* using *A* such that
  - *P* behaves safely when *A* returns MALWARE
  - *P* behaves as malware when *A* returns SAFE
- Apply *P* to itself, *A* must fail, CONTRADICTION

Procedural Defenses

- Chain of custody
- Tamper evident security seals
- Paper trails documenting custody

These fail when the procedures get complex.

- People make mistakes!
- Procedures themselves create vulnerabilities
  - Security seal broken by accident
  - Signature mismatch because of age or illness

Defense in Depth

Procedural defenses

- We admit potential flaws
- We demand known flaws be addressed

Technical defenses

- We admit potential flaws
- We demand known flaws be addressed

Preferably, multiple layers of each defense type