ACCURATE

A CENTER FOR
CORRECT, USABLE,
RELIABLE, AUDITABLE,
AND TRANSPARENT ELECTIONS

# Overview of and Perspectives on UOCAVA Voting

David Wagner
UC Berkeley

# The current state of UOCAVA

Of overseas voters:

- 22% requested but never received their ballot
- 39% received ballot late (Oct 15 or later)
- 24% had questions or problems when registering

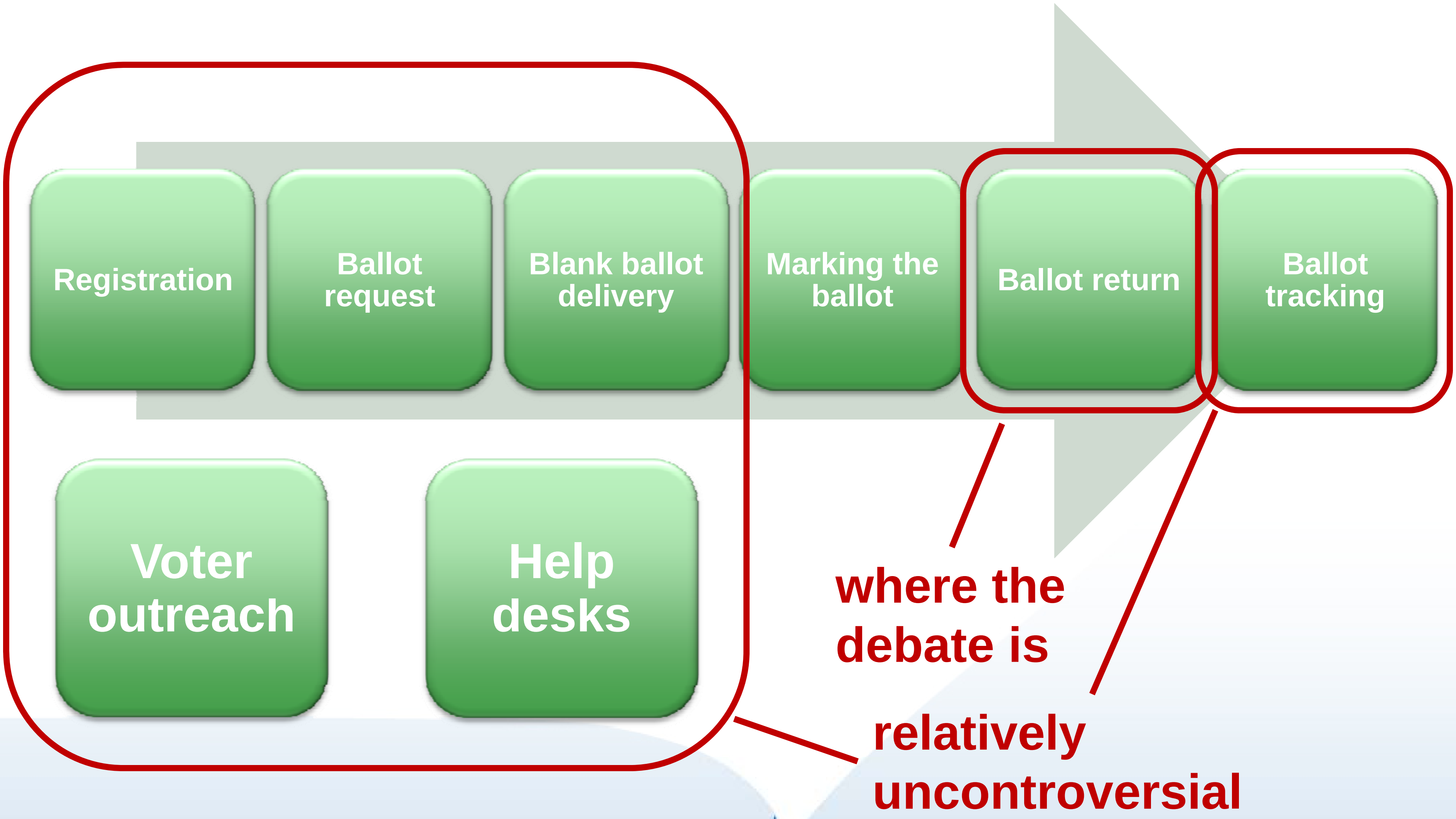
# The current state of UOCAVA

Of overseas voters:

- 22% requested but never received their ballot
- 39% received ballot late (Oct 15 or later)
- 24% had questions or problems when registering
- **~ 30% tried to vote, but their vote didn't count**

This is a travesty.

# Where can technology help?



Registration

Ballot request

Blank ballot delivery

Marking the ballot

Ballot return

Ballot tracking

Voter outreach

Help desks

where the debate is

relatively uncontroversial

ACCURATE

# Let’s talk about software

All software has bugs.

All software has bugs we don’t know of.

Unknown bugs can have unbounded consequences.

In many voting systems, a single bug can undetectably change the election outcome.

# Let’s talk about security

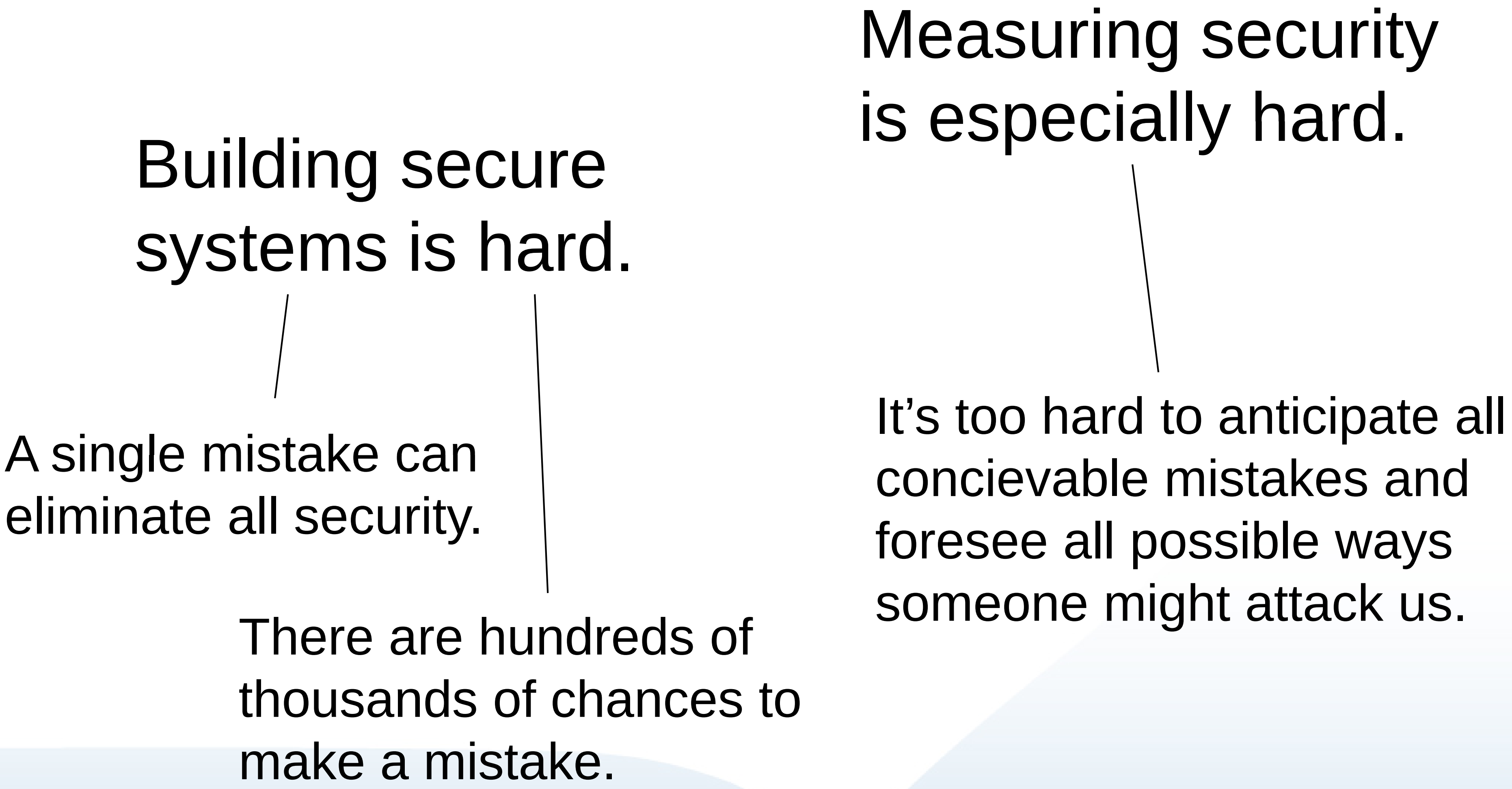
All software has security bugs.

All software has security bugs we don’t know of.

Unknown security bugs can have unbounded consequences.

In many voting systems, a single security bug can undetectably change the election outcome.

# Why is security hard?



Building secure systems is hard.

A single mistake can eliminate all security.

There are hundreds of thousands of chances to make a mistake.

Measuring security is especially hard.

It’s too hard to anticipate all concievable mistakes and foresee all possible ways someone might attack us.

# Why is security hard?

Building secure systems is hard.

Imagine trying to write a tax code that's free of loopholes.

Measuring security is especially hard.

Now imagine trying to verify that a draft tax code is loophole-free

(when it was written by your arch-rival)

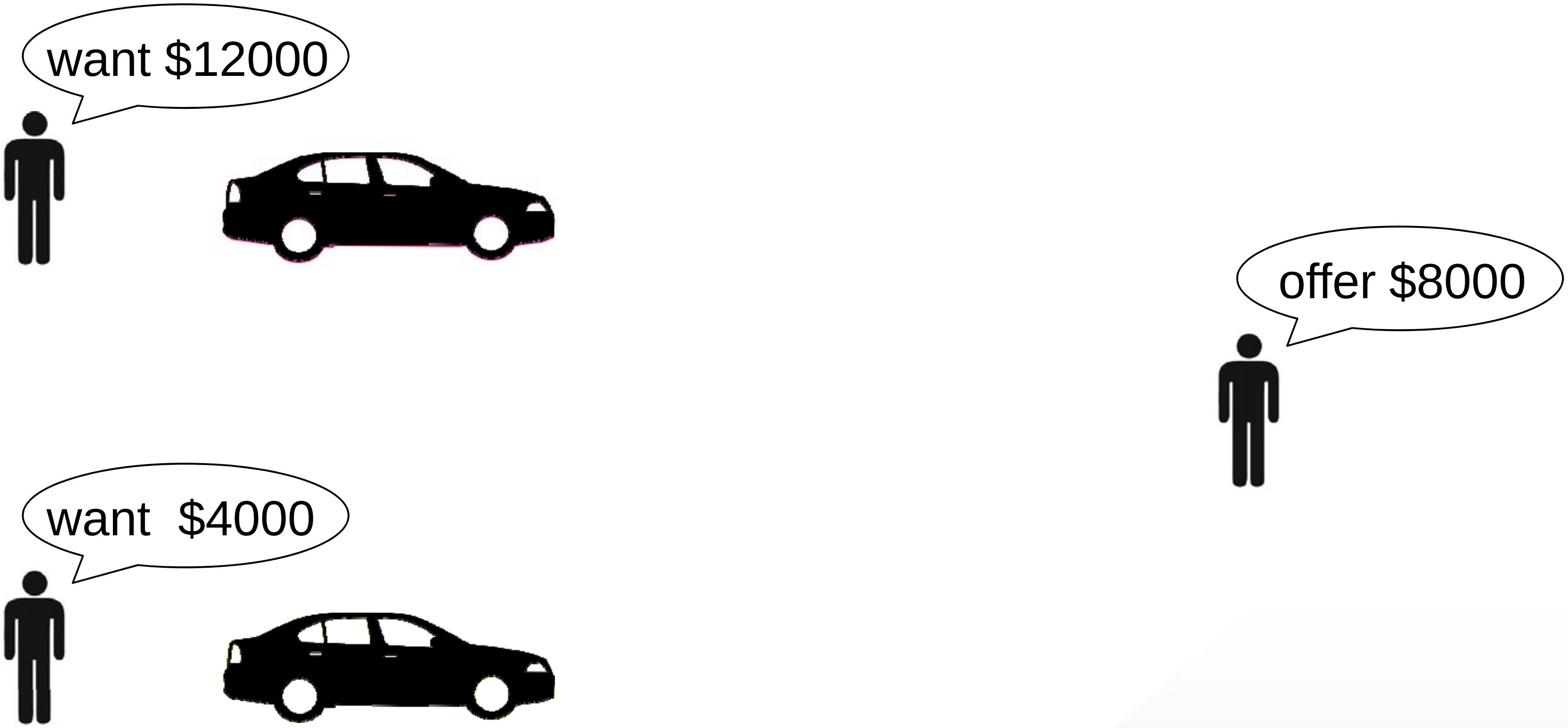
# Measuring security of software

- State-of-the-art: Architectural risk analysis + security code review
- Cost: \$2-\$20 per line of code
  (modern voting system: ≥ 100K lines of code)
- Rarely authoritative; subjective; qualitative
  (“it’s insecure”, “dunno”, “might be OK”)
- Often not so convincing for third parties
  (thus ill-suited to voting, where we must convince everyone that the system is trustworthy)

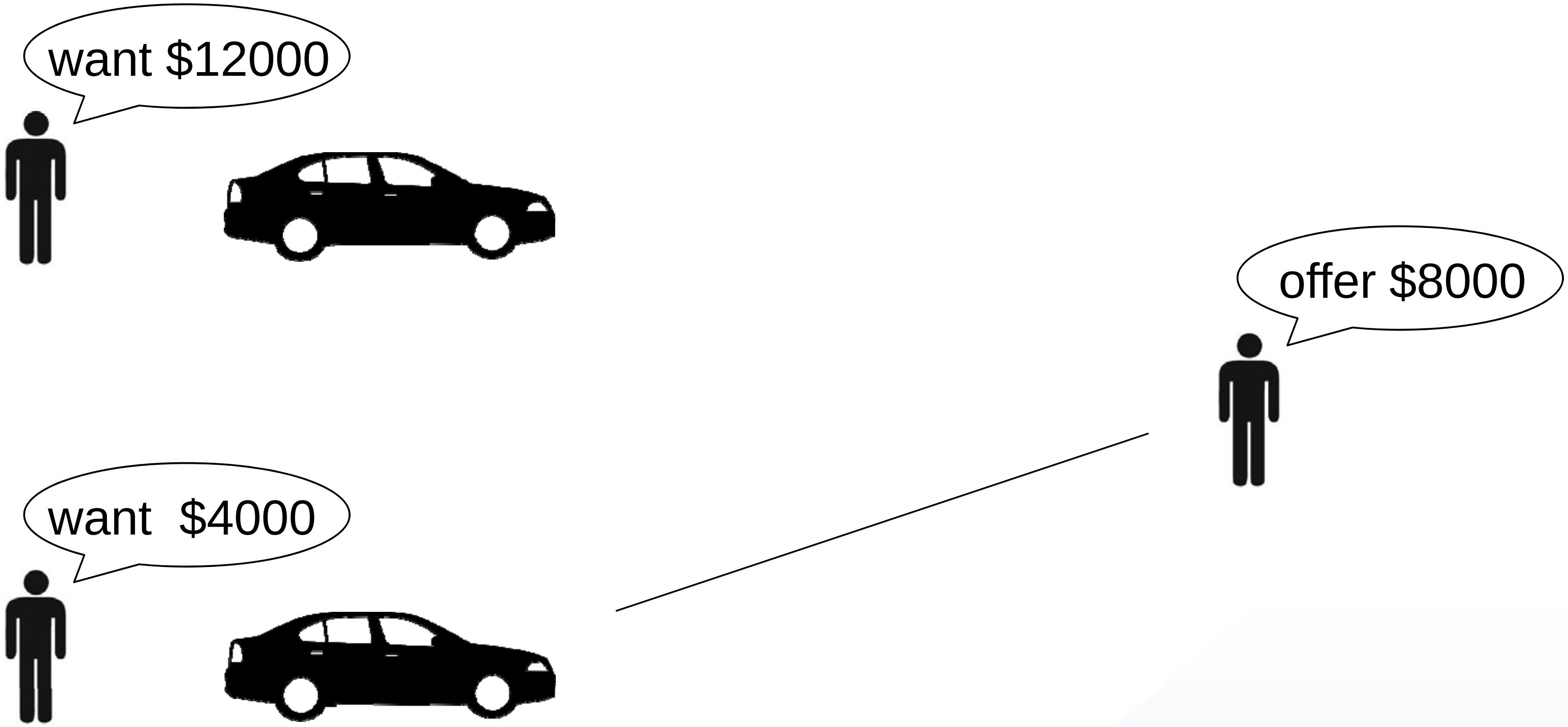
# In short

We have no plausible metrics for measuring security.

# The market for lemons



# The market for lemons

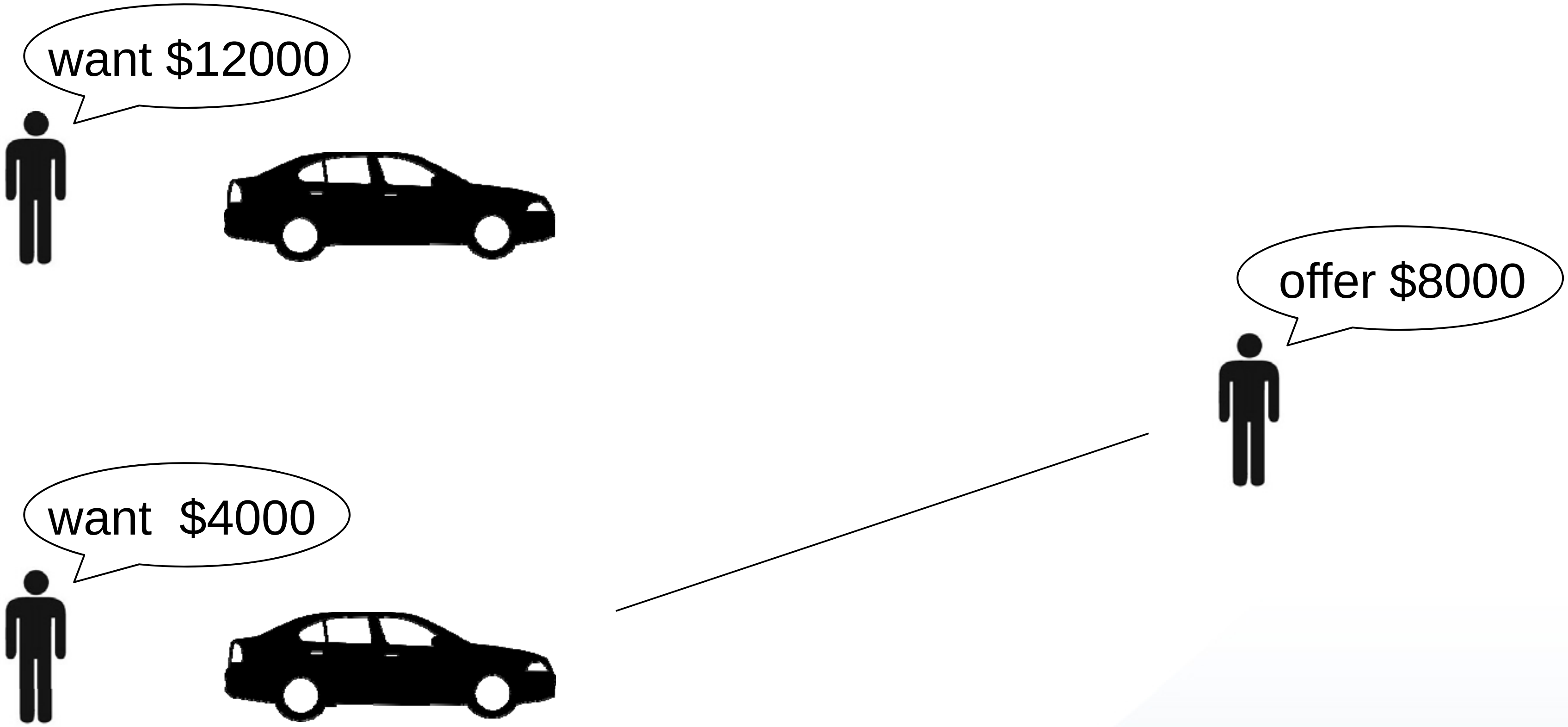


want $12000

want  $4000

offer $8000

# The market for lemons



# The market for lemons

# The market for lemons

# In short

We have no plausible metrics for measuring security.

Tends to drive market towards lowest common denominator security.

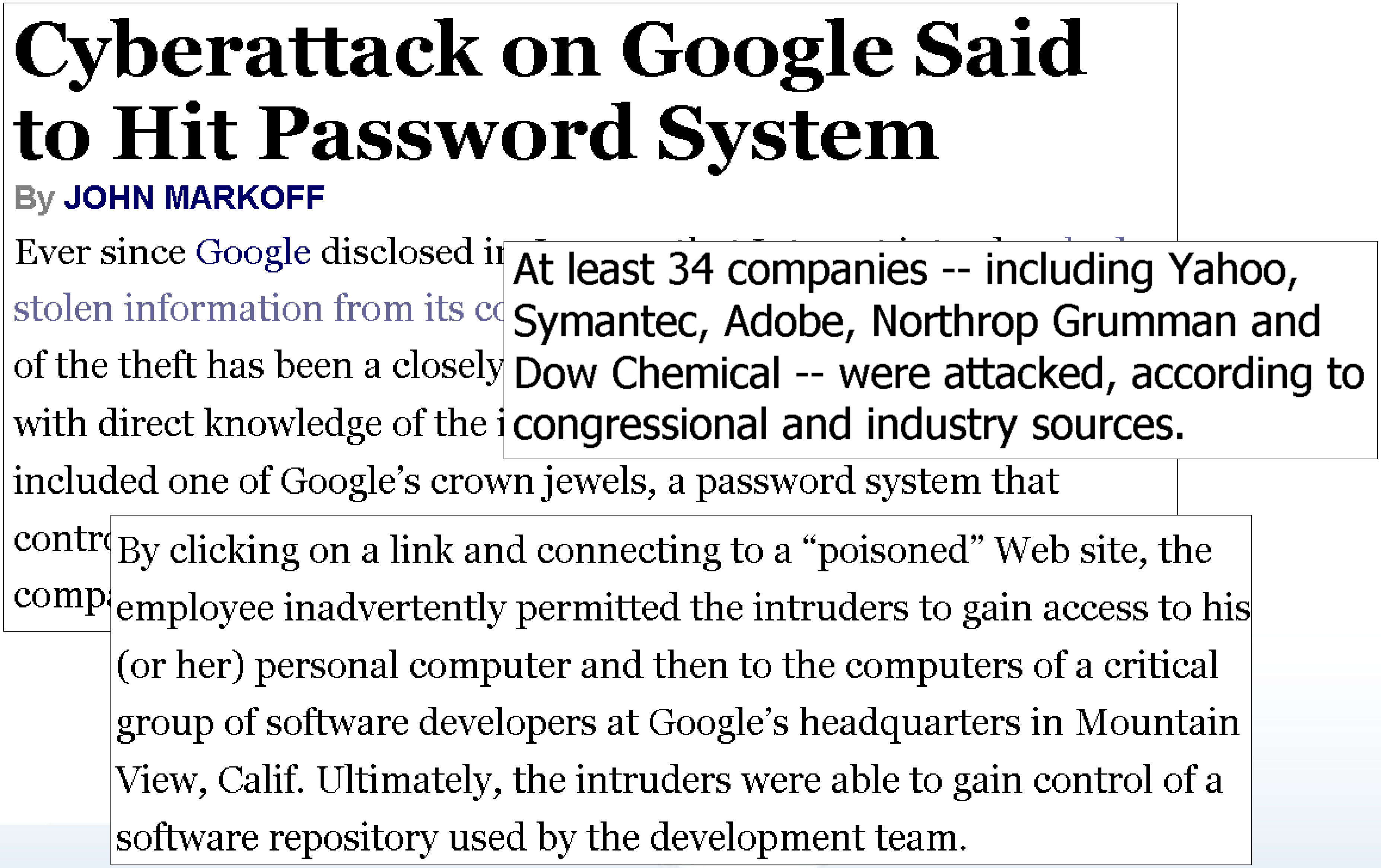
# Why is security hard?

Measuring security is especially hard.

Building secure systems is hard.

Operating a system securely is expensive.

A single mistake can eliminate all security.



# Cyberattack on Google Said to Hit Password System

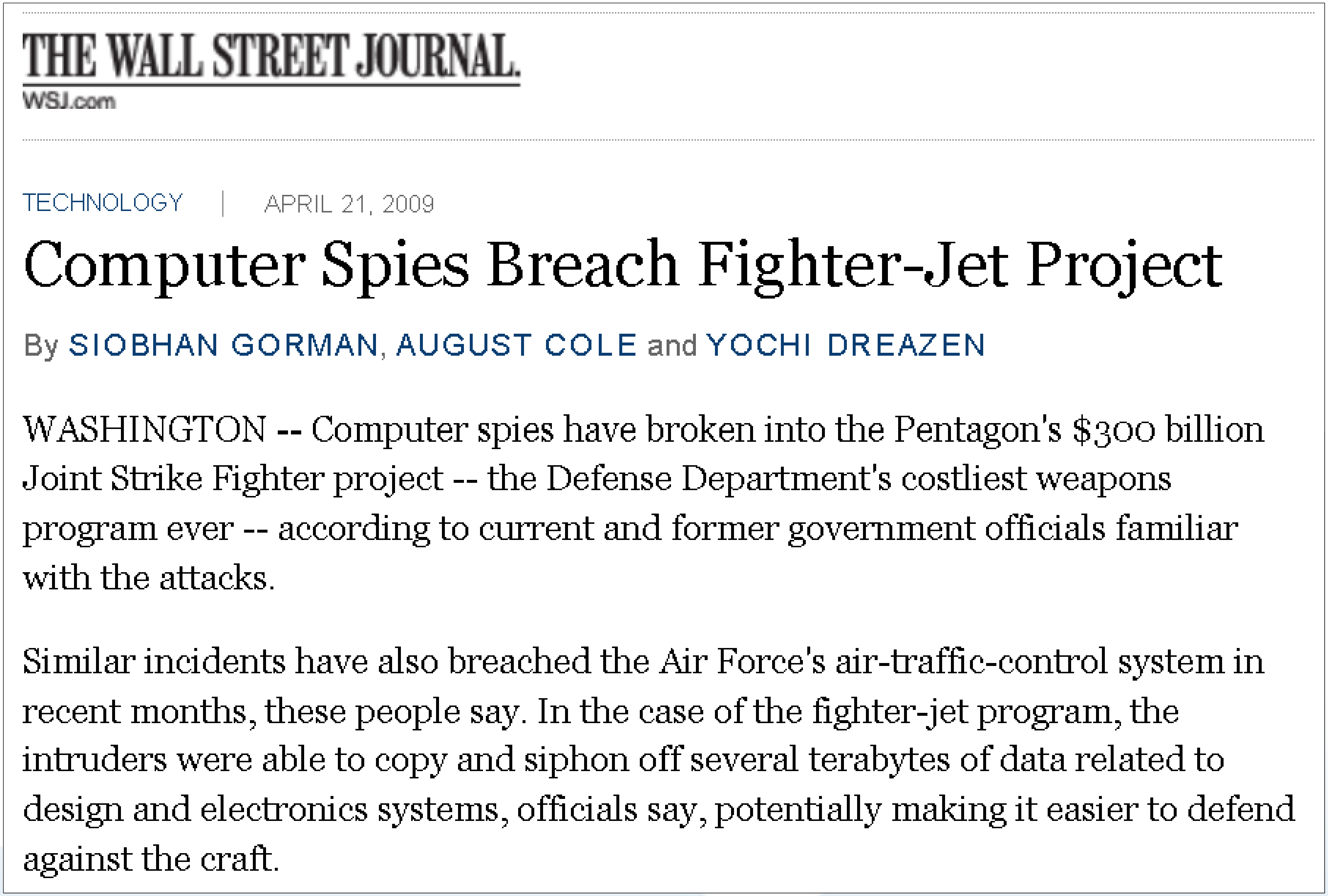
By JOHN MARKOFF

Ever since Google disclosed i
stolen information from its c
of the theft has been a closely
with direct knowledge of the i
included one of Google's crown jewels, a password system that
contr
comp

At least 34 companies -- including Yahoo, Symantec, Adobe, Northrop Grumman and Dow Chemical -- were attacked, according to congressional and industry sources.

By clicking on a link and connecting to a "poisoned" Web site, the employee inadvertently permitted the intruders to gain access to his (or her) personal computer and then to the computers of a critical group of software developers at Google's headquarters in Mountain View, Calif. Ultimately, the intruders were able to gain control of a software repository used by the development team.

ACCURATE



THE WALL STREET JOURNAL.

WSJ.com

TECHNOLOGY | APRIL 21, 2009

# Computer Spies Breach Fighter-Jet Project

By SIOBHAN GORMAN, AUGUST COLE and YOCHI DREAZEN

WASHINGTON -- Computer spies have broken into the Pentagon's $300 billion Joint Strike Fighter project -- the Defense Department's costliest weapons program ever -- according to current and former government officials familiar with the attacks.

Similar incidents have also breached the Air Force's air-traffic-control system in recent months, these people say. In the case of the fighter-jet program, the intruders were able to copy and siphon off several terabytes of data related to design and electronics systems, officials say, potentially making it easier to defend against the craft.

# Personal perspective

- It is not technologically feasible today to make Internet voting safe against attack.
- Operating an Internet voting system safely requires expertise and money way beyond what election officials are likely to have.
- There is no known way to audit Internet voting. UOCAVA votes might fall “under a cloud of suspicion.”

# Security risks of Internet voting

- Insecure clients: Many PCs are insecure. Election officials don’t control voter PCs and can’t assure their security.
- Insecure networks: Man-in-the-middle attacks, denial-of-service attacks.
- Phishing and social engineering: Attacker may be able to fool users into revealing authentication credentials or to fool users into thinking they voted.

# Security risks of Internet voting

- Attacks can be mounted by anyone, from foreign soil, beyond the reach of US law.
- There may be no way to detect attacks.
- Even if attacks are detected, there may be no way to identify the attacker, prosecute the culprit, or recover from the attack.
- The level of sophistication required is within reach of existing attackers.

# Quantitative metrics?

Expected loss =
cost(attack) × Prob(attack)

# Quantitative metrics?

Expected loss =
cost(attack)
× Prob(vulnerability exists)
× Prob(attacker attacks)
× Prob(attack succeeds)

enormous

Let's calculate!

out of our control

≈ 100%?

# Quantitative metrics?

Prob(vulnerability exists) = $1 – (1 – R)^N$

where:

N = # of lines of code

R – security bugs/line of code

# Quantitative metrics?

$\text{Prob(vulnerability exists)} = 1 – (1 – R)^N$

where:

$N$ = # of lines of code ≈ 100,000

$R$ – security bugs/line of code ≈ 0.0002

# Quantitative metrics?

Prob(vulnerability exists) = $1 - (1 - R)^N$
$\approx 0.99999998$

where:
N = # of lines of code ≈ 100,000
R – security bugs/line of code ≈ 0.0002

Lesson: All software has security bugs.

# What could go wrong?

- Selective denial of service: targeted demographic is prevented from voting, affects election outcome
- Central server is hacked, affects election outcome
- Malware/viruses spread that hack voter PCs, affecting election outcome
- Loss of confidence in public elections, public refuses to accept legitimacy of elected leaders
- False-flag operation makes us blame an innocent country, affecting international relations

Lesson: Attacks could have a large impact.

# Quantitative metrics?

Expected loss =
cost(attack)
× Prob(vulnerability exists)
× Prob(attacker attacks)
× Prob(attack succeeds)

≈ cost(attack) × 1 × 1 × 1 = large.

Lesson: Internet voting looks risky, by this metric.

# Metric: Attack team size

- Attack team size = # of people who would have to collude to successfully affect election outcome
- Premise: Large conspiracies are hard to keep secret. Voting systems with large Attack team size are more robust against attack than systems with small Attack team size.

Opinion: This metric seems promising.

# Metric: Expert consensus

- Possible metric: Fraction of independent security experts who are willing to endorse a voting system
- Concept: Proponents could publicly disclose all details of voting system and make the case why their system is safe. Experts with no conflict of interest could evaluate claims.
- Model: AES standardization process.

Opinion: Unclear whether it will work here.

ACCURATE

# Metric: Unauditable votes cast

- Metric: Number of votes cast that cannot be audited (to verify absence of plausible failures that could affect election outcome).
- Premise: Some votes will be cast over insecure media (e.g., fax, email) no matter what; it is the quantity that matters.  If a voting system can be audited, security weaknesses are tolerable.

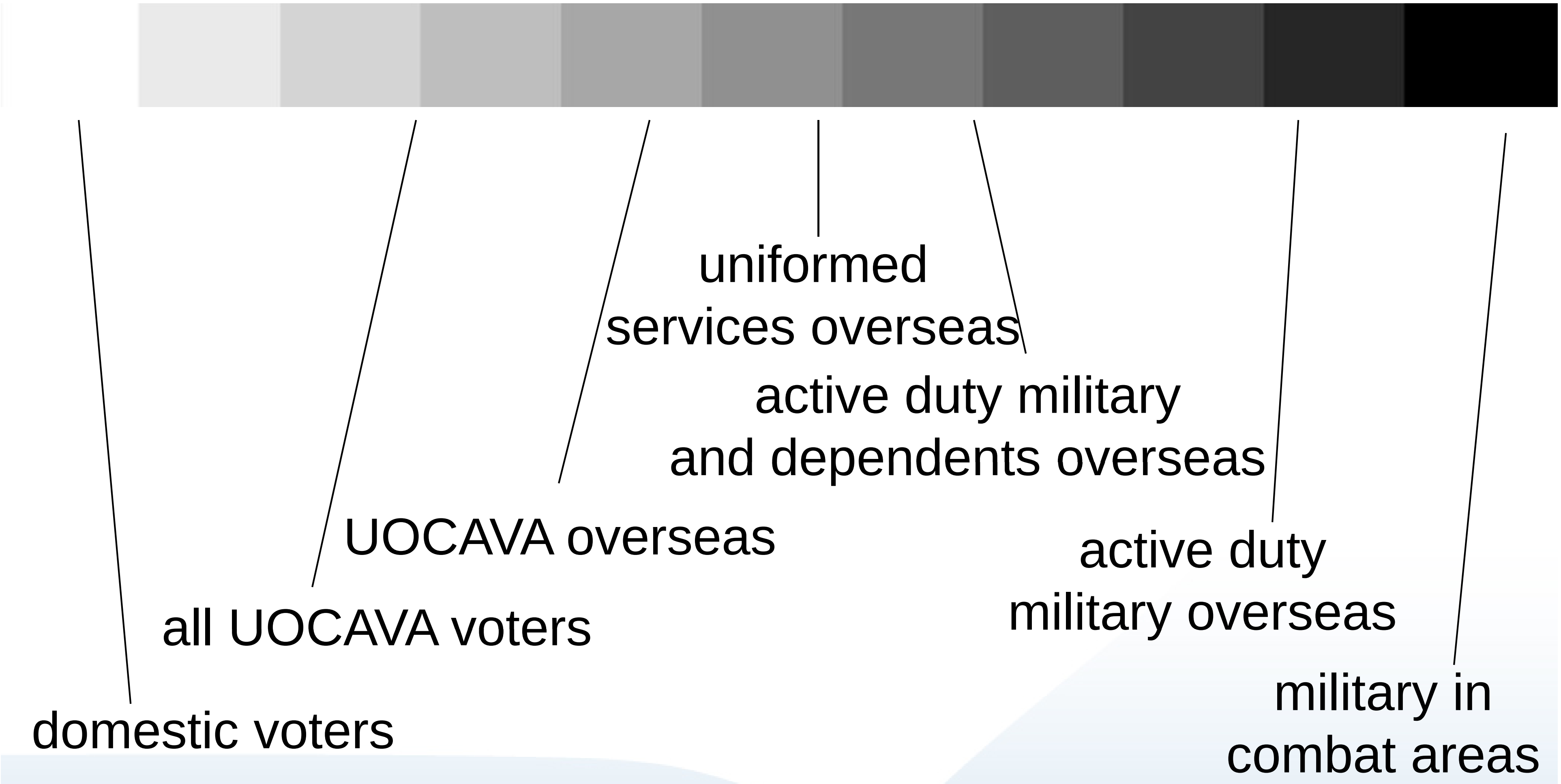
Opinion: Needs discussion.

# Policy levers

What parameters can policymakers control?

- Number of users of a voting system
- Support for technological innovation
- Public disclosure

# Risk is proportional to use ?



domestic voters

all UOCAVA voters

UOCAVA overseas

uniformed
services overseas

active duty military
and dependents overseas

active duty
military overseas

military in
combat areas

ACCURATE

# Technology innovation

Can new breakthroughs address these risks?

- Maybe – see “e2e” (open-audit) voting systems.
- Currently, there is no known e2e system suitable for use with UOCAVA voting. Currently, there appears to be little research focused on UOCAVA voting. However, a sustained research effort might produce dividends.
- Caution: Beware false advertising. Some Internet voting vendors adopt the language, without delivering the goods. Certifying e2e systems will be expensive.

# Public disclosure

- Policymakers could mandate public disclosure of source code, binaries, design docs for all Internet voting systems, before they are certified.
- After disclosure, policymakers could mandate a period for evaluation before system is deployed.

# Under-appreciated alternatives

- Use FWAB for blank-ballot delivery
  - Needs: Publish contests and candidates in open format. Publish address → precinct map. Enable no-excuse FWAB. Accept votes in state and local races on FWAB.
- Accept FWAB as FPCA
  - Why require voters to send in FPCA? If the voter is registered and sends a FWAB, it should be accepted.
- Transmit ballots to voters earlier
- Online registration, ballot request, change of address, ballot tracking.

# Success stories

- Overseas Vote Foundation
  - Non-profit with very modest funding.
  - In 2008: 120,000 voters used OVF web wizard to fill out ballot request or FWAB. 10,000 voters returned voted ballot by FedEx. 5M visits to OVF web pages.
- Minnesota's 2008 UOCAVA initiative
  - Increased number of UOCAVA voters who successfully voted by 2.5×, in two years.
  - Increased # of ballots returned from 27% (2006) to 76%.
  - How? Process reforms, OVF web tools, voter outreach.

# Concluding thoughts

- Internet voting is risky.
- There are alternative ways of using technology to help UOCAVA voters, but they are receiving little attention.
- Don’t expect quantitative, repeatable, affordable metrics for measuring security of software system.

ACCURATE

A CENTER FOR
CORRECT, USABLE,
RELIABLE, AUDITABLE,
AND TRANSPARENT ELECTIONS