# Reverse Bayesian poisoning:

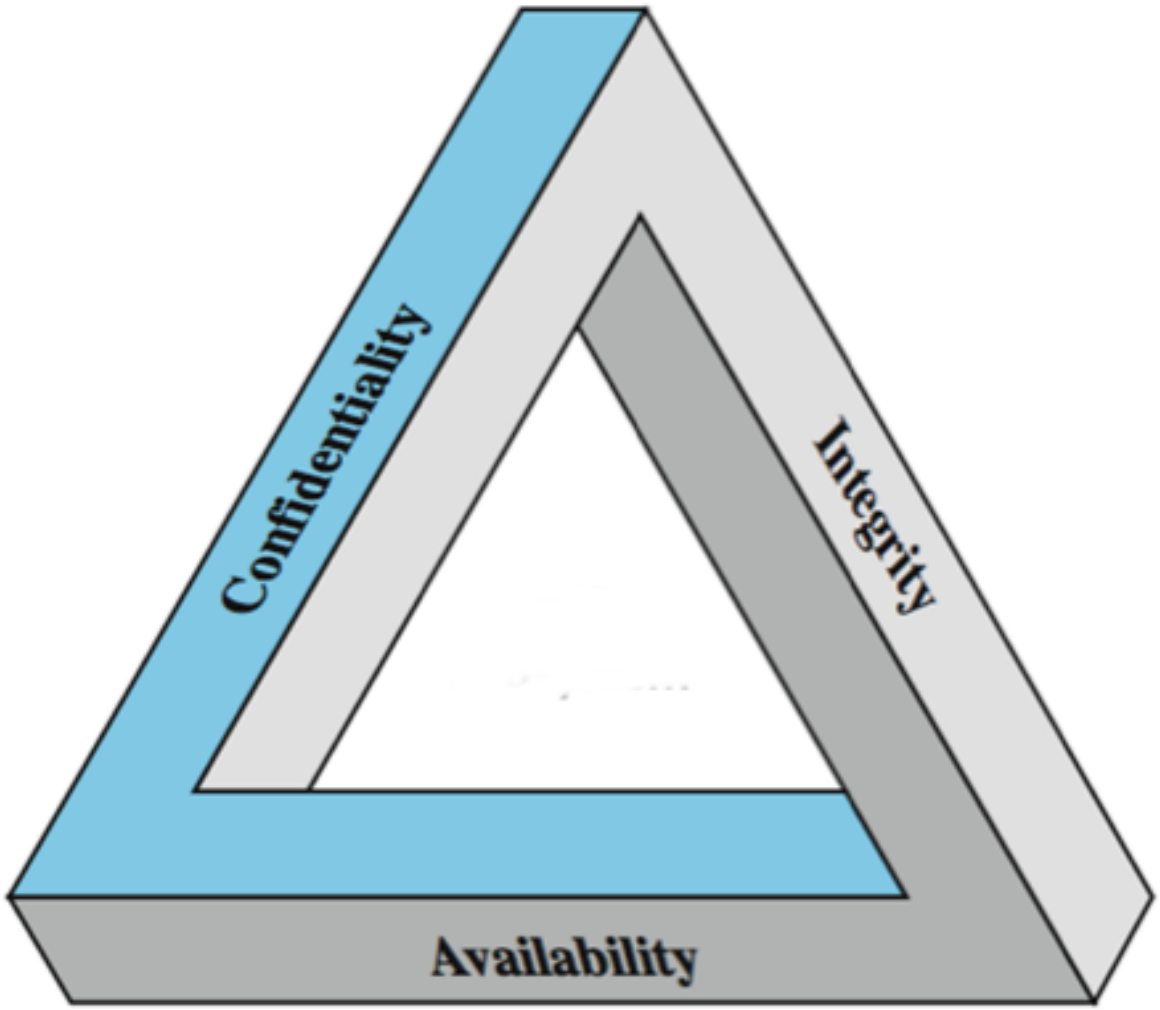
# How to use spam filters to manipulate online elections

Sjouke Mauw
University of Luxembourg

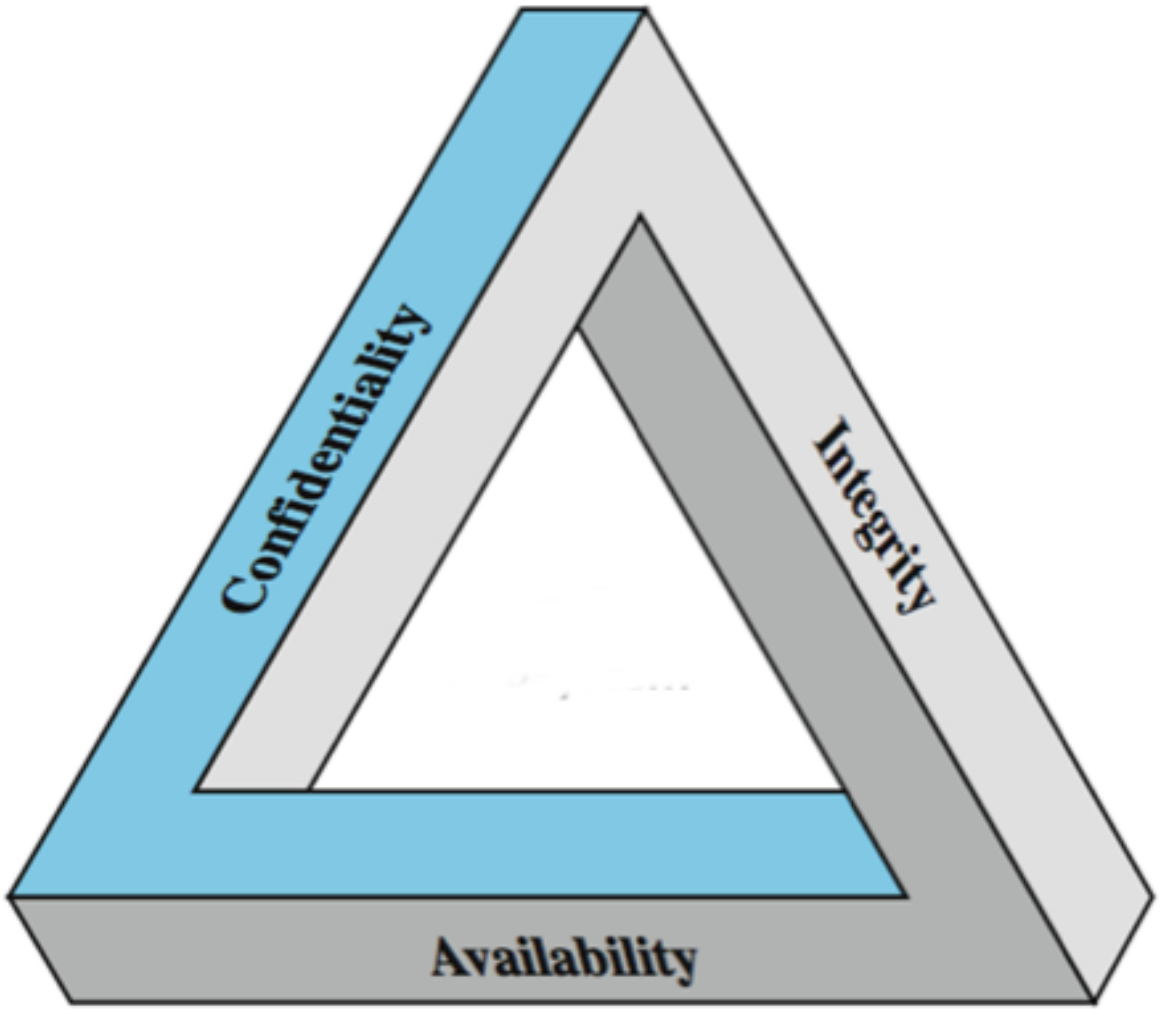
(joint work with Hugo Jonker, Tom Schmitz)

E-Vote-ID'17, 27 October 2017, Bregenz

# E-voting and the CIA triad



# E-voting and the CIA triad




Largely ignored

# Literature on Denial-of-Service attacks in e-voting

- Considered a serious threat to e-voting by some.
- But mostly ignored in security analysis.
- Studied from a generic point of view.
- Considered easily detectable.
- Focus on disruption of the election process, not on influencing the outcome.
- E-Vote-ID'17: the solution of a DDoS prevention provider introduces new vulnerabilities.

# Attacker can manipulate election results if. . .

- DoS focused on selected voters.
- Stealthy.

# Attacker can manipulate election results if...

- DoS focused on selected voters.
- Stealthy.

Reverse Bayesian Poisoning

# Attacker can manipulate election results if...

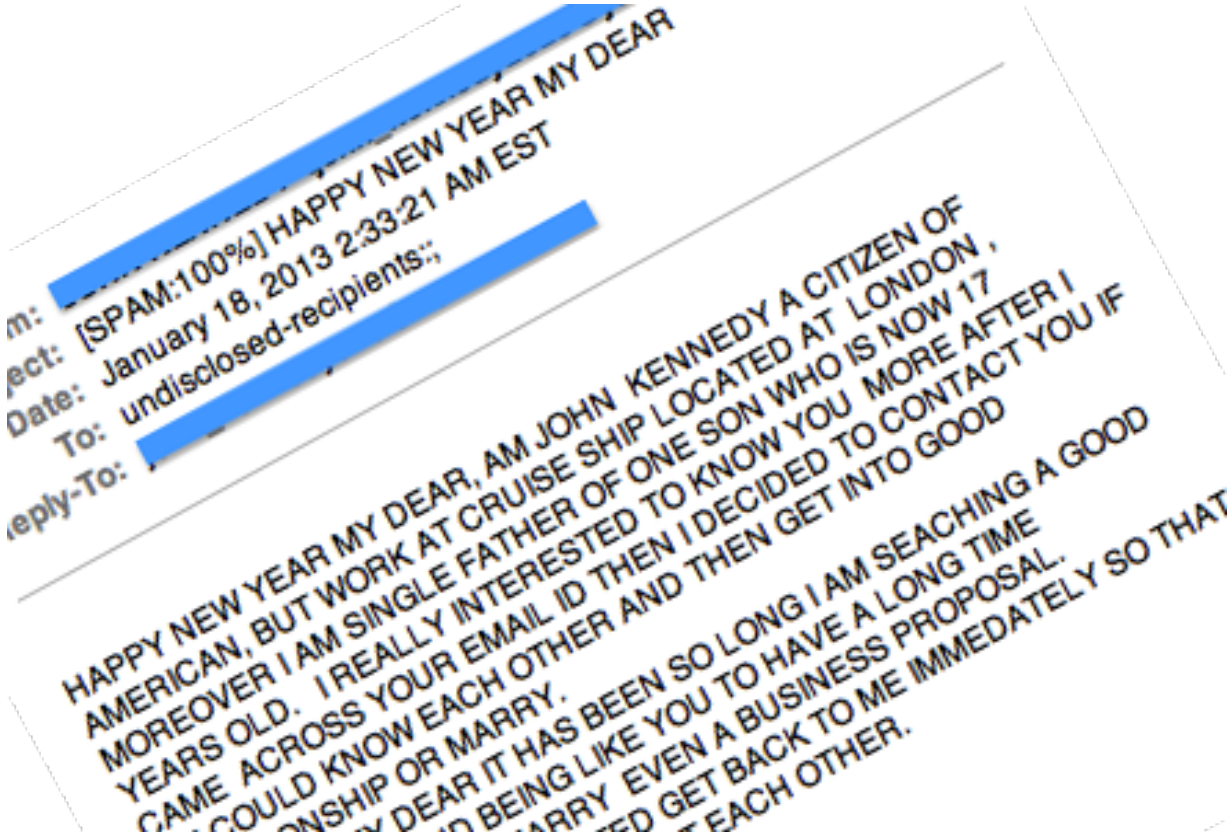
- DoS focused on selected voters.
- Stealthy.

Reverse Bayesian Poisoning

Feasibility study on Helios, Bogofilter.

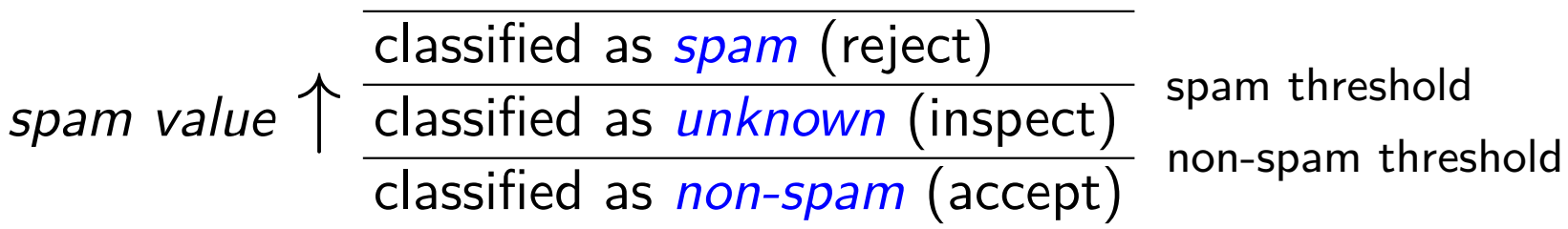
# Spam

- $>$ 50% of email traffic is spam.
- Spam filtering is a necessity.



# Spam classification

Spam filter calculates *spam value* of incoming message.
Based on occurring words, URLs, sender, mime parts, etc.



# Spam classification is not perfect

| | is spam | is not spam |
|---|---|---|
| classified as spam | true reject | false reject |
| classified as unknown | | |
| classified as non-spam | false accept | true accept |

# Spam classification is not perfect

| | is spam | is not spam |
|---|---|---|
| classified as spam | true reject | false reject |
| classified as unknown | | |
| classified as non-spam | false accept | true accept |

- false accept: Spam mail is offered as legitimate mail to user.

# Spam classification is not perfect

| | is spam | is not spam |
|---|---|---|
| classified as spam | true reject | false reject |
| classified as unknown | | |
| classified as non-spam | false accept | true accept |

- false accept: Spam mail is offered as legitimate mail to user.
- false reject: Legitimate mail is discarded as spam.

# Bayesian spam filtering

Suppose an incoming email contains the word “viagra”.
What is the probability that it is spam?

$$P(spam \mid \text{“}viagra\text{”}) =$$

$$\frac{P(\text{“}viagra\text{”} \mid spam) \cdot P(spam)}{P(\text{“}viagra\text{”} \mid spam) \cdot P(spam) + P(\text{“}viagra\text{”} \mid \neg spam) \cdot P(\neg spam)}$$

# Bayesian spam filtering

Suppose an incoming email contains the word “viagra”.
What is the probability that it is spam?

$$P(spam \mid \text{“}viagra\text{”}) =$$

$$\frac{P(\text{“}viagra\text{”} \mid spam) \cdot P(spam)}{P(\text{“}viagra\text{”} \mid spam) \cdot P(spam) + P(\text{“}viagra\text{”} \mid \neg spam) \cdot P(\neg spam)}$$

$P(\text{“}viagra\text{”} \mid spam)$, $P(\text{“}viagra\text{”} \mid \neg spam)$, $P(\neg spam)$ and $P(spam)$, can all be calculated from a spam/non-spam corpus.

# Attacking Bayesian spam filters

false accept: Spam mail considered legitimate.

*Bayesian poisoning:*
Add sufficiently many non-spammy words to a spam message to obtain a low spam score.

# Attacking Bayesian spam filters

false accept: Spam mail considered legitimate.

*Bayesian poisoning:*
Add sufficiently many non-spammy words to a spam message to obtain a low spam score.

false reject: Legitimate mail is discarded as spam.

*Reverse Bayesian poisoning:*
Train the spam filter with spam mails that also contain words from the regular mails that you want to be rejected.

# Can an attacker remotely train a spam filter?

YES, simply by sending spam to the user,

- If the user is actively marking incoming mails as spam, or
- if the spam filter's *auto-update* feature is enabled.

# Feasibility experiment

- Helios voting system.
- Bogofilter Bayesian spam filter.
- Trained with the Enron email corpus.
- Fully controlled, isolated environment without human interaction.

# Helios

- Attack on administrative emails (outside security features of Helios).
- If user doesn't vote, there is no further action from Helios.
- You can't vote without the credentials sent by Helios.

# Template of a Helios invitation

Dear <voter.name>,

<custom_message>

Election URL: <election_vote_url>
Election Fingerprint: <voter.election.hash>

Your voter ID: <voter.voter_login_id>
Your password: <voter.voter_password>

Log in with your <voter.voter_type> account.

We have recorded your vote with smart tracker: <voter.vote_hash>
You may re-vote if you wish: only your last vote counts.

In order to protect your privacy, this election is configured
to never display your voter login ID, name, or email address to the public.
Instead, the ballot tracking center will only display your alias.
Your voter alias is <voter.alias>.

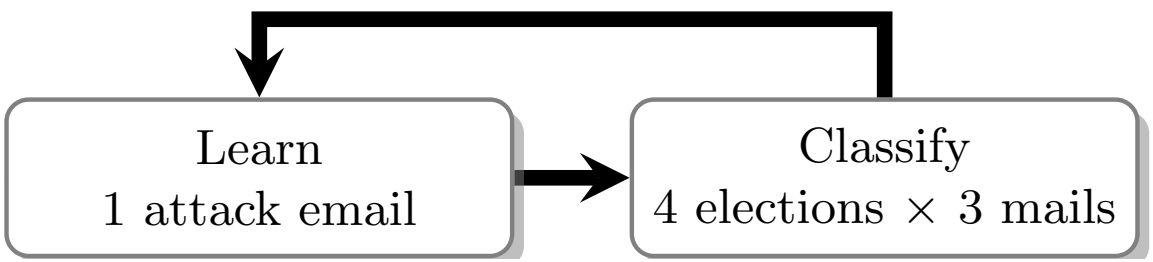
IMPORTANTLY, when you are prompted to log in to vote,
please use your *voter ID*, not your alias.
–
Helios

# Example attack email (fragment)

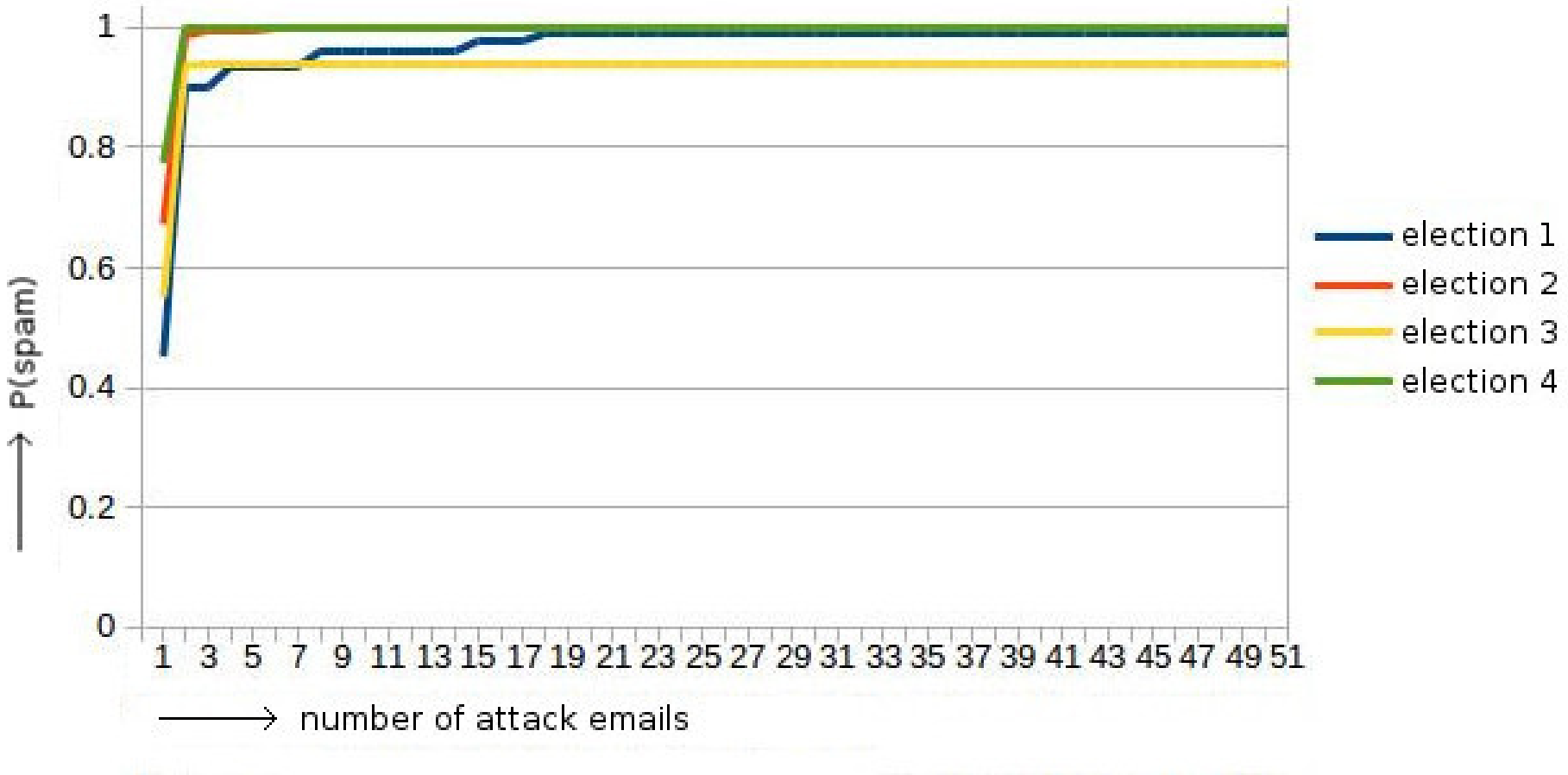
From: Luxury@experience.com
Subject: Lower monthly payment passwords
Remuneration Election Subsidiary Link: payment Dear
Usury – Reapportionment Helios Reply How Syndicate
to Wholesale Vote Return ========= Computer
Election roots URL: Coattail Your Challenger voter
Believe ID: Decide Your Permit password: Advertisement
Log Pamphlets in Broadcast with Downsize your
...

# The experiment

1. Set up 4 elections with Helios.
2. select 3 administrative emails per election.
3. Train Bogofilter using Enron corpus.
4. Train with 50 generated attack emails, one by one.
5. After each attack email classify the administrative emails.



# Results


1
0.8
0.6
0.4
0.2
0
P(spam)
1 3 5 7 9 11 13 15 17 19 21 23 25 27 29 31 33 35 37 39 41 43 45 47 49 51
number of attack emails
election 1
election 2
election 3
election 4

# Observations

- Just a few attack mails suffice to drastically increase the spam value of genuine Helios emails.
- The attack is stealthy. Even not provable after some time if users retrain their spam filter (which is advised to avoid Bayesian poisoning attacks).
- Can be used against group of voters that share a spam filter.
- Assumptions for the attack to have an effect:
  - The attacker knows which victims to attack in order to influence the election results.
  - Victims ignore the fact that they don't receive an invitation from Helios.
  - Election officials don't inform voters to look in their spam box.
  - Elections are not repeated (even if voters complain that their invitation ended up in their spam box).
  - Auto-update feature or active marking by user.

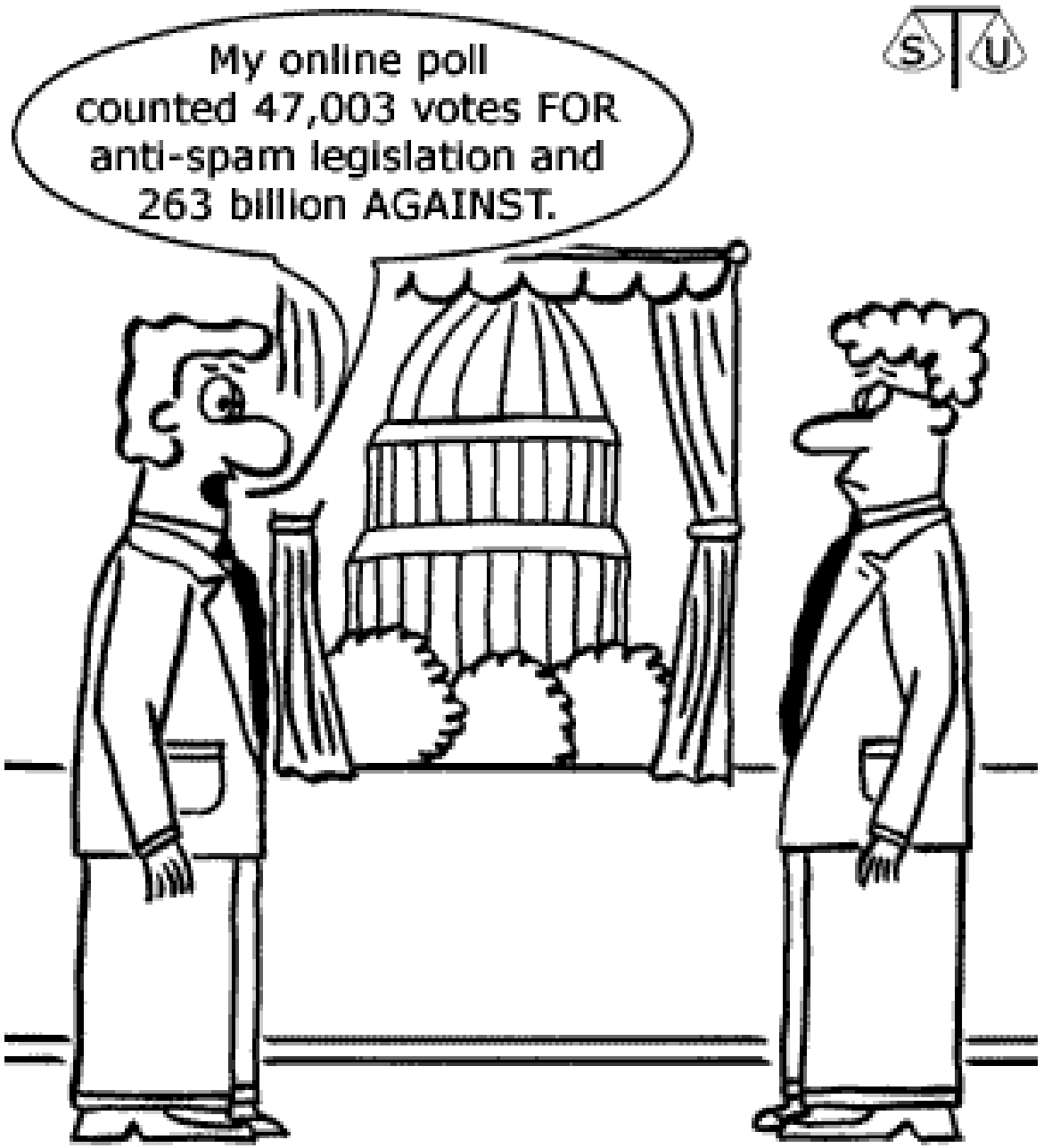
Future:

- Attack can be further optimized. We only showed feasibility.
- Requires field study with real subjects (risk of poisoning e.g. Google's central spam filter).
- Include administrative emails and DoS attacks in the formalization and verification of voting protocols.

Possible mitigation:

- User side (e.g. whitelisting election email address, calendar service for expected invitations from the voting system).
- Central (e.g. multi-channel communication, request notification of receipt).

# Questions?


My online poll counted 47,003 votes FOR anti-spam legislation and 263 billion AGAINST.