Access Control and Privacy Policies (4)

Email: christian.urban at kcl.ac.uk
Office: S1.27 (1st floor Strand Building)
Slides: KEATS (also homework is there)

Unix-Style Access Control

- Q: "I am using Windows. Why should I care?"
  A: In Windows you have similar AC:

  administrators group
  (has complete control over the machine)
  authenticated users
  server operators
  power users
  network configuration operators

- Modern versions of Windows have more fine-grained AC than Unix; they do not have a setuid bit, but have `runas` (asks for a password).

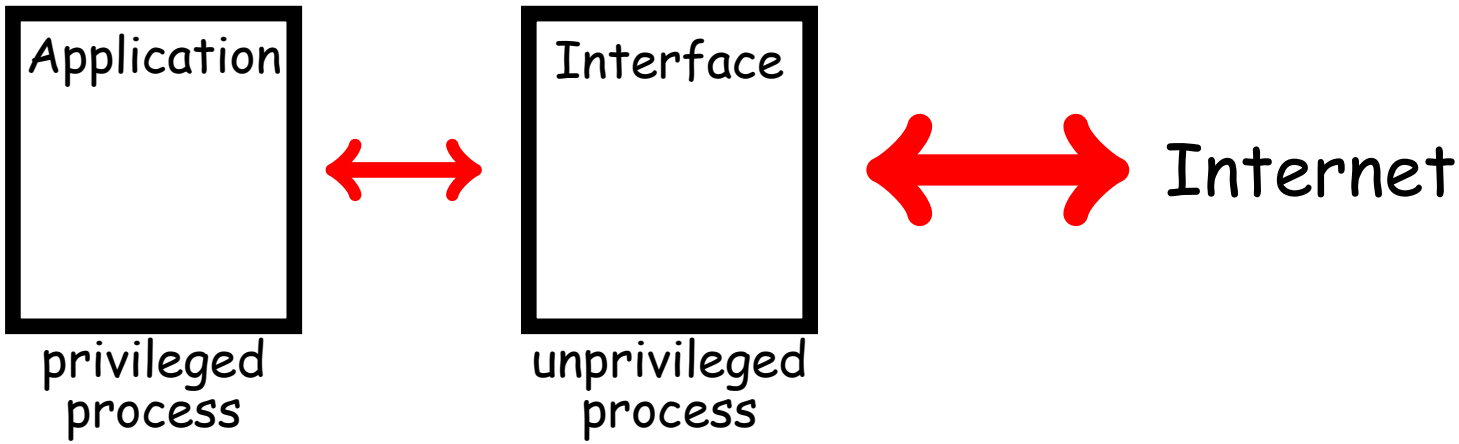
Unix-Style Access Control

- Q: "I am using Windows. Why should I care?"
  A: In Windows you have similar AC:

  administrators group
  (has complete control over the machine)
  authenticated users
  server operators
  power users
  network configuration operators

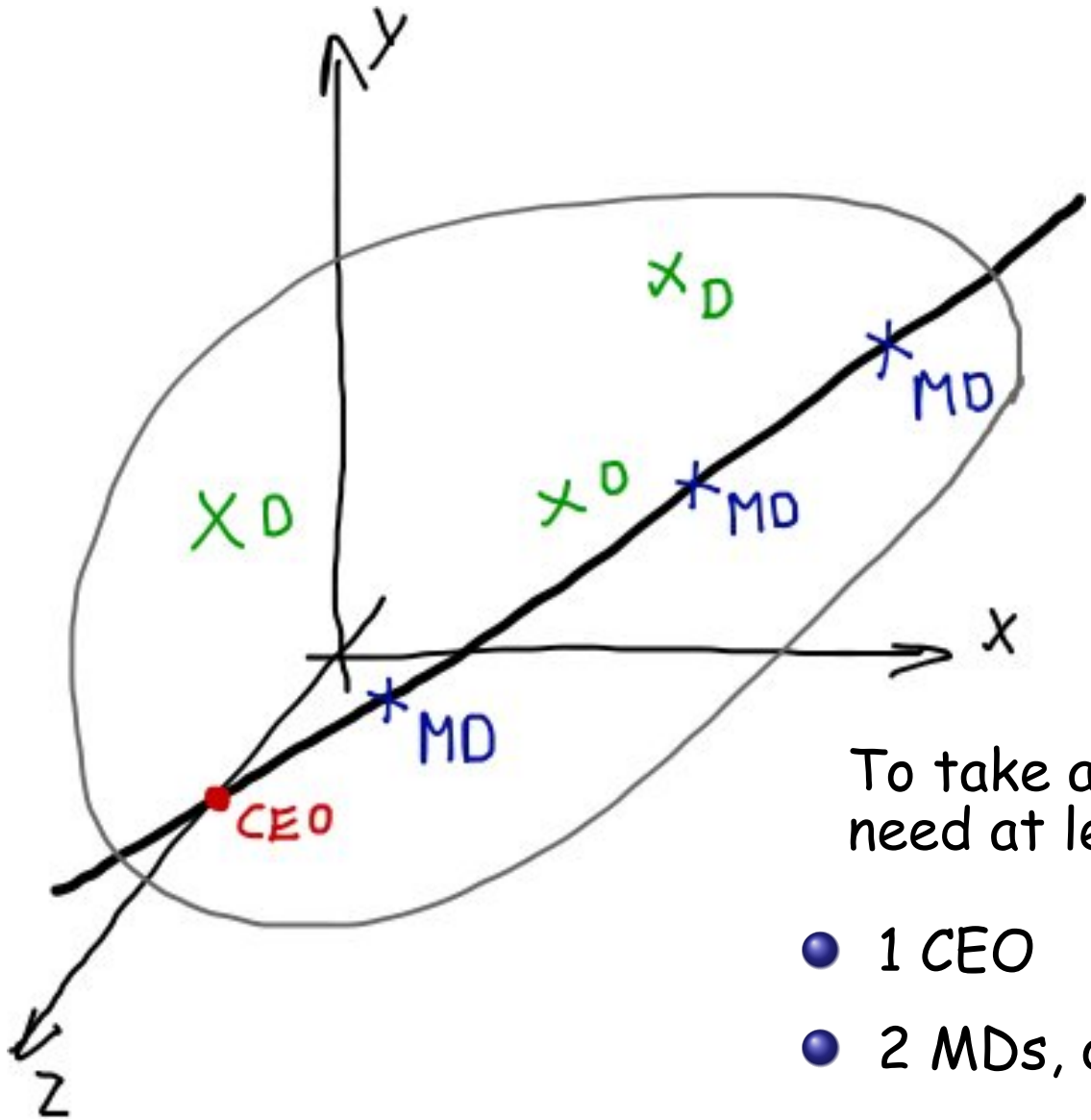
- Modern versions of Windows have more fine-grained AC than Unix; they do not have a setuid bit, but have `runas` (asks for a password).
- OS-provided access control can **add** to your security.

Network Applications: Privilege Separation



- the idea is make the attack surface smaller and mitigate the consequences of an attack

Shared Access Control



To take an action you need at least either:

- 1 CEO
- 2 MDs, or
- 3 Ds

Lessons from Access Control

Not just restricted to Unix:

- if you have too many roles (i.e. too finegrained AC), then hierarchy is too complex
  you invite situations like. . . let's be root

- you can still abuse the system. . .

A "Cron"-Attack

The idea is to trick a privileged person to do something on your behalf:

- root:

```
rm /tmp/*/*
```

A "Cron"-Attack

The idea is to trick a privileged person to do something on your behalf:

- root:

  `rm /tmp/*/*`

  the shell behind the scenes:

  $\texttt{rm /tmp/dir}_1\texttt{/file}_1\ \texttt{/tmp/dir}_1\texttt{/file}_2\ \texttt{/tmp/dir}_2\texttt{/file}_1\ \ldots$

  this takes time

A "Cron"-Attack

1. attacker (creates a fake passwd file)
   `mkdir /tmp/a; cat > /tmp/a/passwd`
2. root (does the daily cleaning)
   `rm /tmp/*/*`

   records that `/tmp/a/passwd`

   should be deleted, but does not do it yet
3. attacker (meanwhile deletes the fake passwd file, and establishes a link to the real passwd file)
   `rm /tmp/a/passwd; rmdir /tmp/a;`
   `ln -s /etc /tmp/a`
4. root now deletes the real passwd file

A "Cron"-Attack

1. attacker (creates a fake passwd file)
   `mkdir /tmp/a; cat > /tmp/a/passwd`
2. ro
   rm

To prevent this kind of attack, you need additional policies (don't do such operations as root).

should be deleted, but does not do it yet

3. attacker (meanwhile deletes the fake passwd file, and establishes a link to the real passwd file)
   `rm /tmp/a/passwd; rmdir /tmp/a;`
   `ln -s /etc /tmp/a`
4. root now deletes the real passwd file

Schneier Analysis

There is no absolutely secure system and security almost never comes for free.

- What assets are you trying to protect?
- What are the risks to these assets?
- How well does the security solution mitigate those risks?
- What other risks does the security solution cause?
- What costs and trade-offs does the security solution impose?

Example: Credit Cards

You might have the policy of not typing in your credit card online. Worthwhile or not?

Example: Credit Cards

You might have the policy of not typing in your credit card online. Worthwhile or not?

- What assets are you trying to protect?
  your credit card number

Example: Credit Cards

You might have the policy of not typing in your credit card online. Worthwhile or not?

- What assets are you trying to protect?
- What are the risks to these assets?

  With credit cards you loose a fixed amount £50. Amazon £50.

Example: Credit Cards

You might have the policy of not typing in your credit card online. Worthwhile or not?

- What assets are you trying to protect?
- What are the risks to these assets?
- How well does the security solution mitigate those risks?

  Well, hackers steal credit cards from databases. They usually do not attack you individually.

Example: Credit Cards

You might have the policy of not typing in your credit card online. Worthwhile or not?

- What assets are you trying to protect?
- What are the risks to these assets?
- How well does the security solution mitigate those risks?
- What other risks does the security solution cause?

  None (?)

Example: Credit Cards

You might have the policy of not typing in your credit card online. Worthwhile or not?

- What assets are you trying to protect?
- What are the risks to these assets?
- How well does the security solution mitigate those risks?
- What other risks does the security solution cause?
- What costs and trade-offs does the security solution impose?

  Internet shopping is convenient and sometimes cheaper.

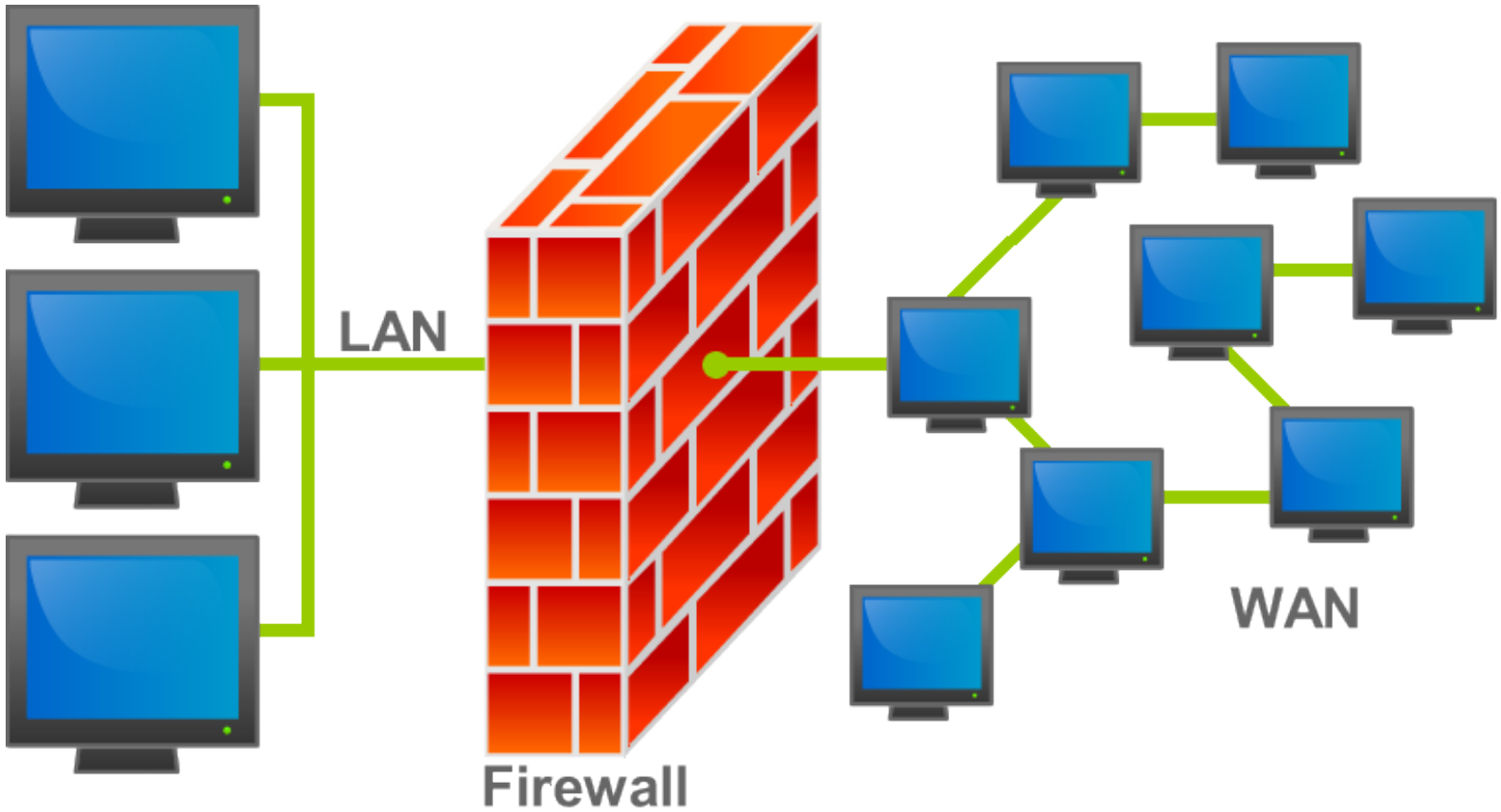
Example: Credit Cards

You might have the policy of not typing in your credit card online. Worthwhile or not?

- What assets are you trying to protect?
- What are the risks to these assets?
- How well does the security solution mitigate those risks?
- What other risks does the security solution cause?
- What costs and trade-offs does the security solution impose?

No!

Example: Firewalls



A firewall is a piece of software that controls incoming and outgoing traffic according to some rules.

Example: Firewalls

- What assets are you trying to protect?

  Whatever is behind the firewall (credit cards, passwords, blueprints, ...)

Example: Firewalls

- What assets are you trying to protect?
- What are the risks to these assets?

  With a small online shop you are already at risk. Pentagon, definitely.

Example: Firewalls

- What assets are you trying to protect?
- What are the risks to these assets?
- How well does the security solution mitigate those risks?

  Well, at home so not much. Everywhere else, if properly configurated then it does.

Example: Firewalls

- What assets are you trying to protect?
- What are the risks to these assets?
- How well does the security solution mitigate those risks?
- What other risks does the security solution cause?

  There might be backdoors or bugs in the firewall, but generally they are secure. You choose to prevent certain traffic.

Example: Firewalls

- What assets are you trying to protect?
- What are the risks to these assets?
- How well does the security solution mitigate those risks?
- What other risks does the security solution cause?
- What costs and trade-offs does the security solution impose?

  Minimal to modest. Firewalls are part of free software. You need a knowledgeable person to set them up.

Example: Firewalls

- What assets are you trying to protect?
- What are the risks to these assets?
- How well does the security solution mitigate those risks?
- What other risks does the security solution cause?
- What costs and trade-offs does the security solution impose?

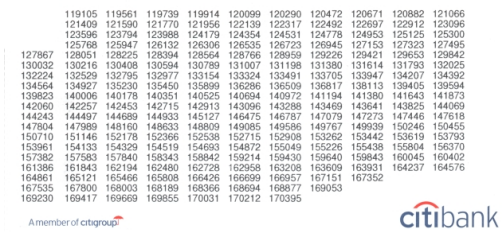
Example: Firewalls

- What assets are you trying to protect?
- What are the risks to these assets?
- How well does the security solution mitigate those risks?
- What other risks does the security solution cause?
- What costs and trade-offs does the security solution impose?

  Yes!

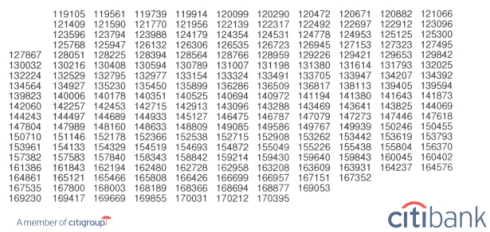
Ex: Two-Factor Authentication

Google uses nowadays two-factor authentication. But it is an old(er) idea. It is used for example in Germany and Netherlands for online transactions.



Ex: Two-Factor Authentication

Google uses nowadays two-factor authentication. But it is an old(er) idea. It is used for example in Germany and Netherlands for online transactions.



Or nowadays by SMS (restricts the validity of the numbers) or with a secure generator

Ex: Two-Factor Authentication

- What assets are you trying to protect?

  Your bank account.

Ex: Two-Factor Authentication

- What assets are you trying to protect?
- What are the risks to these assets?

  Nowadays pretty high risk.

Ex: Two-Factor Authentication

- What assets are you trying to protect?
- What are the risks to these assets?
- How well does the security solution mitigate those risks?

  It prevents problems when passwords are stolen. Man-in-the-middle attacks still possible.

Ex: Two-Factor Authentication

- What assets are you trying to protect?
- What are the risks to these assets?
- How well does the security solution mitigate those risks?
- What other risks does the security solution cause?

  Your mobile phone or credit card/pin might be stolen. SIM card becomes more valuable.

Ex: Two-Factor Authentication

- What assets are you trying to protect?
- What are the risks to these assets?
- How well does the security solution mitigate those risks?
- What other risks does the security solution cause?
- What costs and trade-offs does the security solution impose?

  Banks need to establish an infrastructure.

  For you it might be inconvenient.

Ex: Two-Factor Authentication

- What assets are you trying to protect?
- What are the risks to these assets?
- How well does the security solution mitigate those risks?
- What other risks does the security solution cause?
- What costs and trade-offs does the security solution impose?

Ex: Two-Factor Authentication

- What assets are you trying to protect?
- What are the risks to these assets?
- How well does the security solution mitigate those risks?
- What other risks does the security solution cause?
- What costs and trade-offs does the security solution impose?

  Yes!

Security Seals

According to Ross Anderson: "... is a tamper-indicating device designed to leave non-erasable, unambiguous evidence of unauthorized entry or tampering."

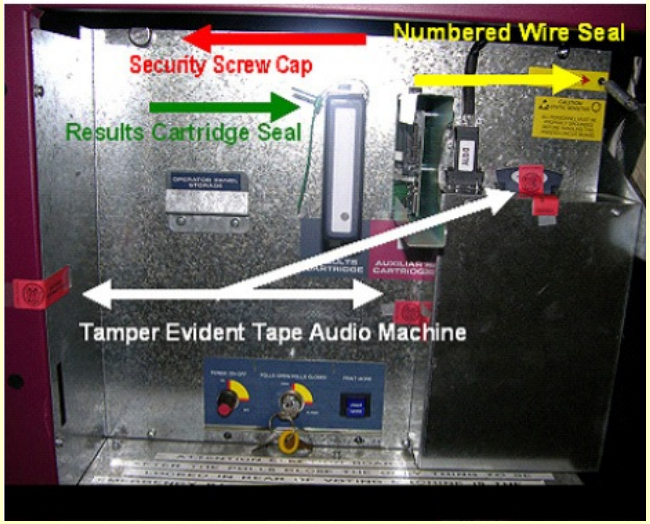
They also need some quite sophisticated policies (seal regiment).

Security Seals (2)

- at the Argonne National Laboratory they tested 244 different security seals
  - meantime to break the seals for a trained person: 100 s
  - including 19% that were used for safeguard of nuclear material

- Andrew Appel defeated all security seals which were supposed to keep voting machines safe

Security Seals (2)



- The tamper-indicating tape can be lifted using a heat gun.
- The security screw cap can be removed using a screwdriver, then the serial-numbered top can be replaced (undamaged) onto a fresh (unnumbered) base.
- The wire seal can be defeated using a #4 wood screw.
- The plastic strap seal can be picked using a jeweler's screwdriver.

Example: Security Seals

- What assets are you trying to protect?

  Voting machines, doors.

Example: Security Seals

- What assets are you trying to protect?
- What are the risks to these assets?

  Casual thieves, insider attacks.

Example: Security Seals

- What assets are you trying to protect?
- What are the risks to these assets?
- How well does the security solution mitigate those risks?

  Needs a quite complicated security regiment.

Example: Security Seals

- What assets are you trying to protect?
- What are the risks to these assets?
- How well does the security solution mitigate those risks?
- What other risks does the security solution cause?

  You might not notice tampering.

Example: Security Seals

- What assets are you trying to protect?
- What are the risks to these assets?
- How well does the security solution mitigate those risks?
- What other risks does the security solution cause?
- What costs and trade-offs does the security solution impose?

  The "hardware" is cheap, but indirect costs can be quite high.

Example: Security Seals

- What assets are you trying to protect?
- What are the risks to these assets?
- How well does the security solution mitigate those risks?
- What other risks does the security solution cause?
- What costs and trade-offs does the security solution impose?

Example: Security Seals

- What assets are you trying to protect?
- What are the risks to these assets?
- How well does the security solution mitigate those risks?
- What other risks does the security solution cause?
- What costs and trade-offs does the security solution impose?

  No! Though in some areas they work: airports, swimming pools, ...

Ex: Security-by-Obscurity

You might think it is a good idea to keep a security relevant algorithm or software secret.

- What assets are you trying to protect?

  Source code, an algorithm and things that depend on it

Ex: Security-by-Obscurity

You might think it is a good idea to keep a security relevant algorithm or software secret.

- What assets are you trying to protect?
- What are the risks to these assets?

  Can be pretty high (Oystercards).

Ex: Security-by-Obscurity

You might think it is a good idea to keep a security relevant algorithm or software secret.

- What assets are you trying to protect?
- What are the risks to these assets?
- How well does the security solution mitigate those risks?

  Not really. The source code can be reverse engineered, stolen, coerced ...

Ex: Security-by-Obscurity

You might think it is a good idea to keep a security relevant algorithm or software secret.

- What assets are you trying to protect?
- What are the risks to these assets?
- How well does the security solution mitigate those risks?
- What other risks does the security solution cause?

  You prevent scrutiny and independent advice. You also more likely than not to get it wrong.

Ex: Security-by-Obscurity

You might think it is a good idea to keep a security relevant algorithm or software secret.

- What assets are you trying to protect?
- What are the risks to these assets?
- How well does the security solution mitigate those risks?
- What other risks does the security solution cause?

No!

Voting as Security Problem

What are the security requirements of a voting system?

Voting as Security Problem

What are the security requirements of a voting system?

- Integrity

- The outcome matches with the voters' intend.
- There might be gigantic sums at stake and need to be defended against.

Voting as Security Problem

What are the security requirements of a voting system?

- Integrity
- Ballot Secrecy

Voting as Security Problem

What are the security requirements of a voting system?

- Integrity
- Ballot Secrecy

- Nobody can find out how you voted.
- (Stronger) Even if you try, you cannot prove how you voted.

Voting as Security Problem

What are the security requirements of a voting system?

- Integrity
- Ballot Secrecy
- Voter Authentication

- Only authorised voters can vote up to the permitted number of votes.

Voting as Security Problem

What are the security requirements of a voting system?

- Integrity
- Ballot Secrecy
- Voter Authentication
- Enfranchisement

- Authorised voters should have the opportunity to vote.

Voting as Security Problem

What are the security requirements of a voting system?

- Integrity
- Ballot Secrecy
- Voter Authentication
- Enfranchisement
- Availability

- The voting system should accept all authorised votes and produce results in a timely manner.

Ballot Boxes

Problems with Voting

Integrity vs. Ballot Secrecy

Authentication vs. Enfranchisement

Problems with Voting

Integrity vs. Ballot Secrecy

Authentication vs. Enfranchisement

Further constraints:

- costs
- accessibility
- convenience
- intelligibility

E-Voting

- The Netherlands between 1997 - 2006 had electronic voting machines
  (hacktivists had found: they can be hacked and also emitted radio signals revealing how you voted)
- Germany had used them in pilot studies
  (in 2007 a law suit has reached the highest court and it rejected electronic voting on the grounds of not being understandable by the general public)
- UK used optical scan voting systems in a few polls

E-Voting

- US used mechanical machines since the 30s, later punch cards, now DREs and optical scan voting machines (fantastic "ecosystem" for study)
- Estonia used in 2007 the Internet for national elections (there were earlier pilot studies in other countries)
- India uses e-voting devices since at least 2003 ("keep-it-simple" machines produced by a government owned company)
- South Africa used software for its tallying in the 1993 elections (when Nelson Mandela was elected) (they found the tallying software was rigged, but they were able to tally manually)

A Brief History of Voting

- Athenians
  - show of hands
  - ballots on pieces of pottery
  - different colours of stones
  - "facebook"-like authorisation

  problems with vote buying / no ballot privacy

- French Revolution and the US Constitution got things "started" with paper ballots (you first had to bring your own; later they were pre-printed by parties)

Ballot Boxes

Security policies involved with paper ballots:

1. you need to check that the ballot box is empty at the start of the poll / no false bottom (to prevent ballot stuffing)
2. you need to guard the ballot box during the poll until counting
3. tallied by a team at the end of the poll (independent observers)

Paper Ballots

What can go wrong with paper ballots?

Paper Ballots

What can go wrong with paper ballots?

William M. Tweed, US Politician in 1860's

"As long as I count the votes, what are you going to do about it?"

Paper Ballots

What can go wrong with paper ballots?

Chain Voting Attack

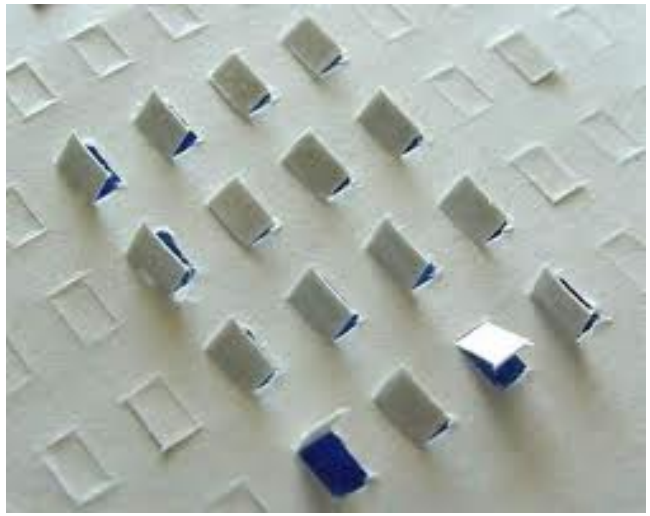
1. you obtain a blank ballot and fill it out as you want
2. you give it to a voter outside the polling station
3. voter receives a new blank ballot
4. voter submits prefilled ballot
5. voter gives blank ballot to you, you give money
6. goto 1

Mechanical Voting Machines

- Lever Voting Machines (ca. 1930 - 1990)

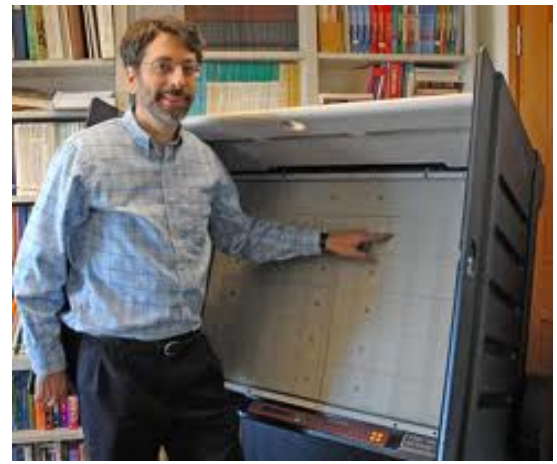
Mechanical Voting Machines

- Lever Voting Machines (ca. 1930 - 1990)
- Punch Cards (ca. 1950 - 2000)



Electronic Voting Machines

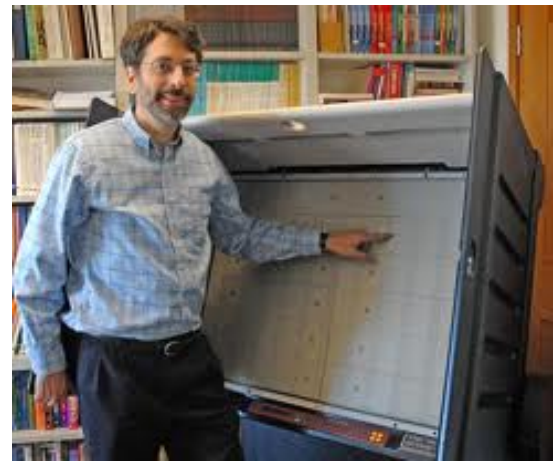
DREs



Optical Scan

Electronic Voting Machines

DREs



Optical Scan

all are computers

DREs

Direct-recording electronic voting machines
(votes are recorded for example memory cards)
typically touchscreen machines
usually no papertrail (hard to add: ballot secrecy)

Diebold Machines

The work by J. Alex Halderman:

- acquired a machine from an anonymous source
- the source code running the machine was tried to keep secret

Diebold Machines

The work by J. Alex Halderman:

- acquired a machine from an anonymous source
- the source code running the machine was tried to keep secret
- first reversed-engineered the machine (extremely tedious)
- could completely reboot the machine and even install a virus that infects other Diebold machines
- obtained also the source code for other machines

Diebold Machines

What could go wrong?

Diebold Machines

What could go wrong? Failure-in-depth.

Diebold Machines

What could go wrong? Failure-in-depth.

A non-obvious problem:

- you can nowadays get old machines, which still store old polls
- the paper ballot box needed to be secured during the voting until counting; e-voting machines need to be secured during the entire life-time

Paper Trail

Conclusion:

Any electronic solution should have a paper trail.

Paper Trail

Conclusion:
Any electronic solution should have a paper trail.

You still have to solve problems about Voter registration, voter authentification, guarding against tampering

E-Voting in India

Their underlying engineering principle is "keep-it-simple":

E-Voting in India

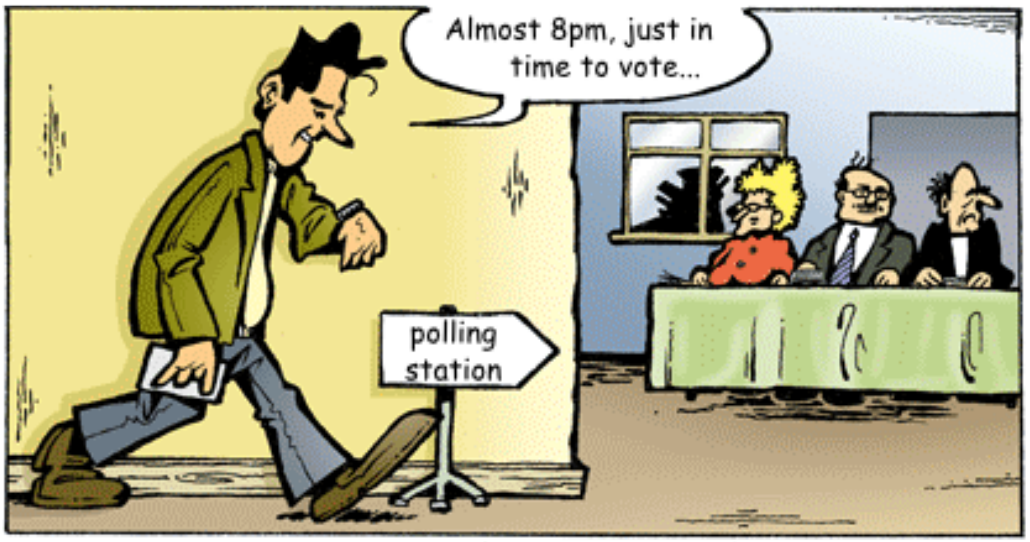
Their underlying engineering principle is "keep-it-simple":

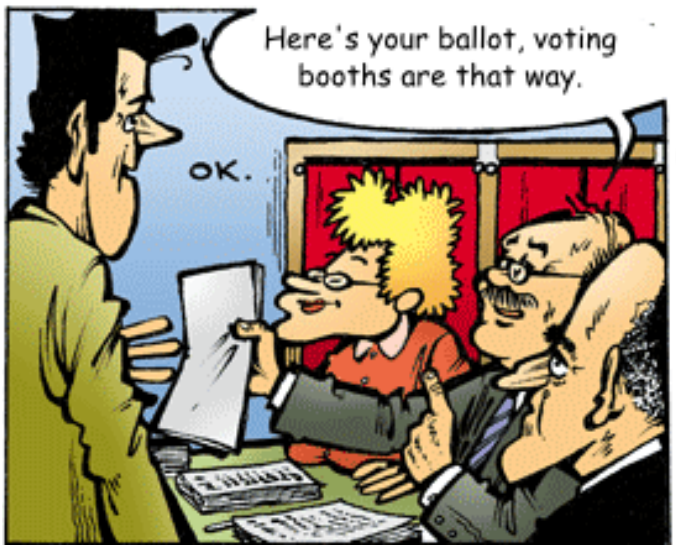
Official claims: "perfect", "tamperproof", "no need for technical improvements" , "infallible"

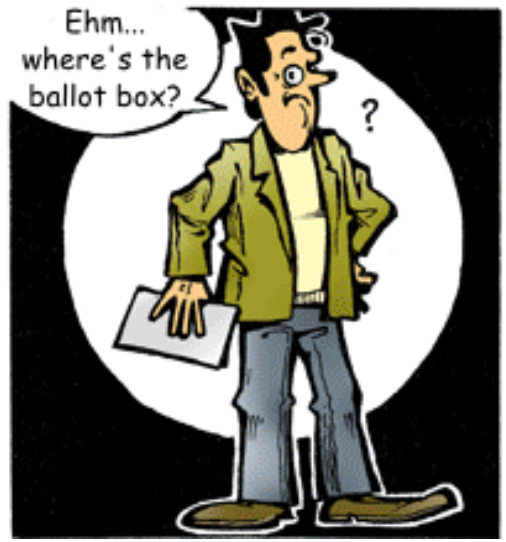
Lessons to be Learned

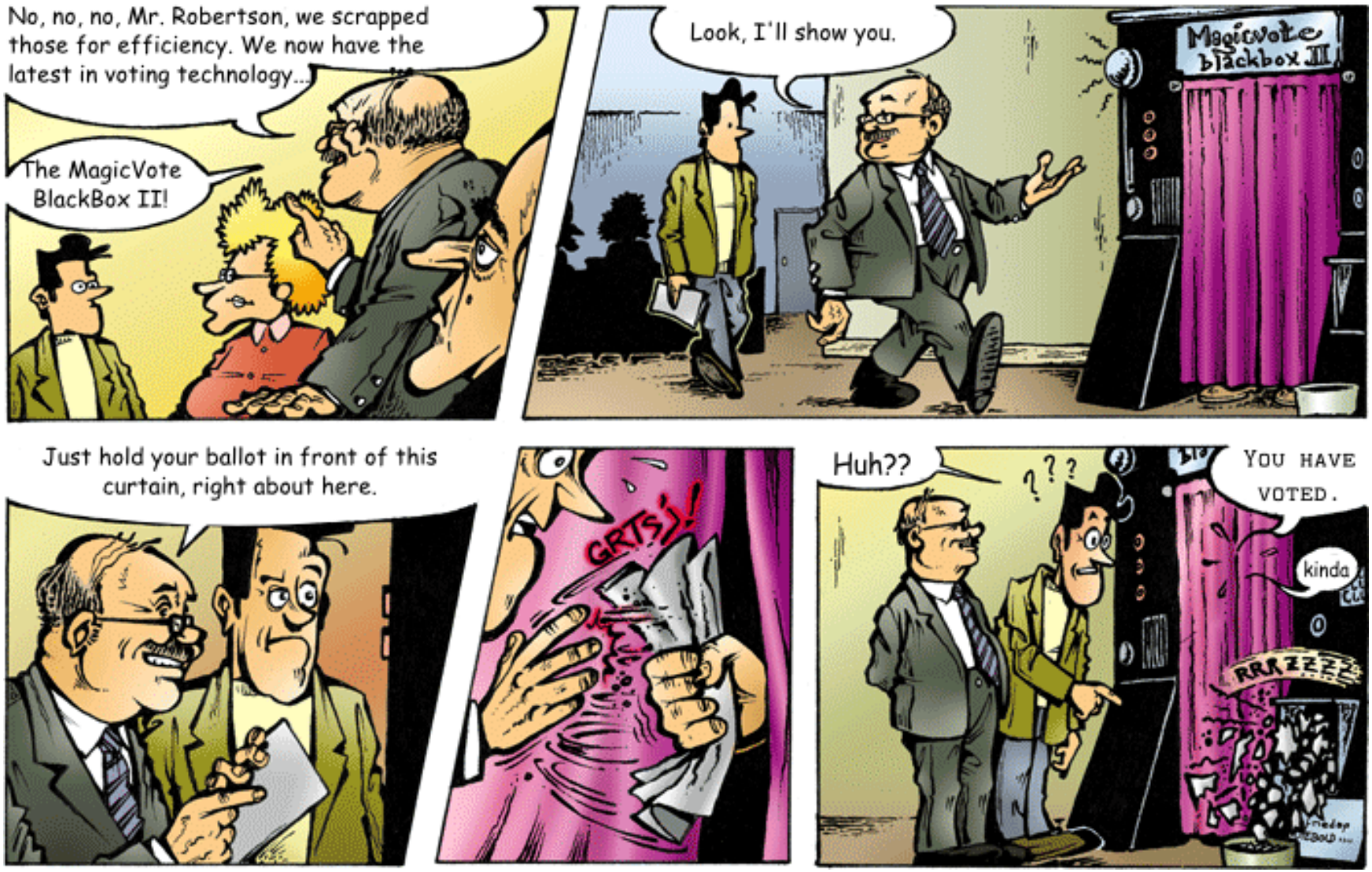
- keep a paper trail and design your system to keep this secure
- make the software open source (avoid security-by-obscurity))
- have a simple design in order to minimise the attack surface

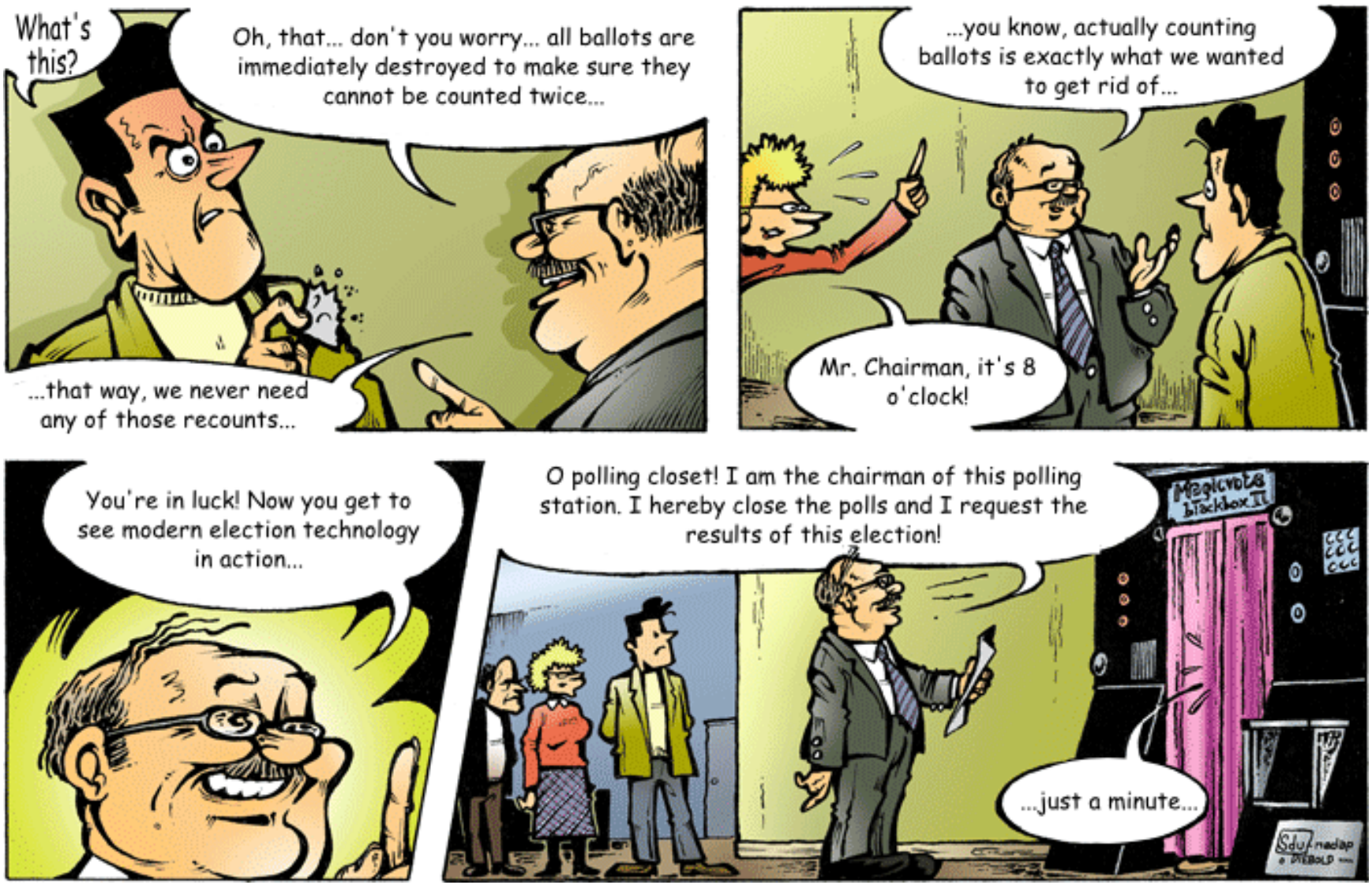
The adventures of citizen Michael C. Robertson

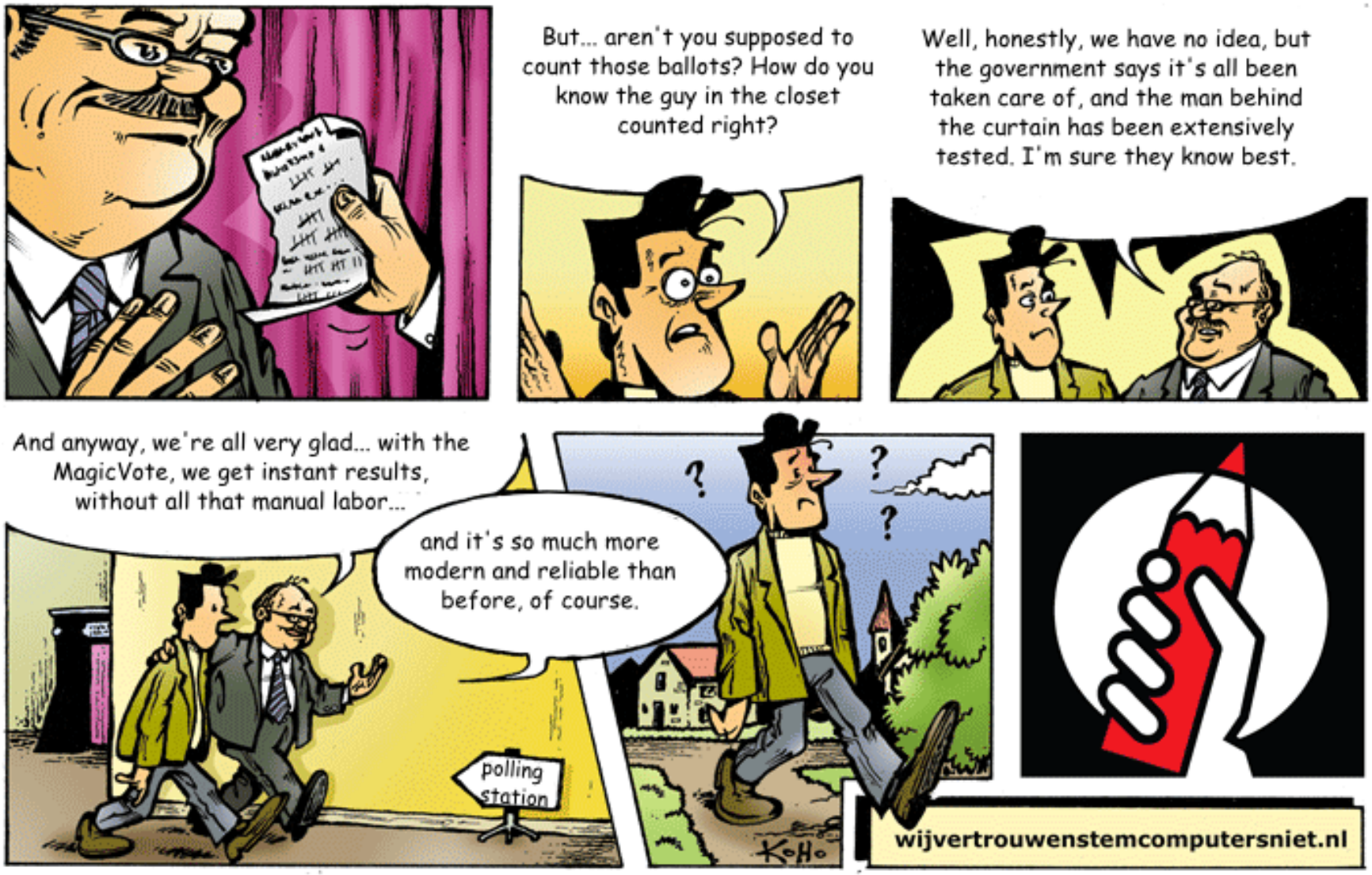







No, no, no, Mr. Robertson, we scrapped those for efficiency. We now have the latest in voting technology...
The MagicVote BlackBox II!
Look, I'll show you.
Magicvote blackbox III
Just hold your ballot in front of this curtain, right about here.
GRTSJ!
Huh??
YOU HAVE VOTED.
kinda
RRRZZZZ


What's this?
Oh, that... don't you worry... all ballots are immediately destroyed to make sure they cannot be counted twice...
...that way, we never need any of those recounts...
...you know, actually counting ballots is exactly what we wanted to get rid of...
Mr. Chairman, it's 8 o'clock!
You're in luck! Now you get to see modern election technology in action...
O polling closet! I am the chairman of this polling station. I hereby close the polls and I request the results of this election!
...just a minute...
Magicvote blackbox II



Drawings: Koen Hottentot — Story: Rop Gonggrijp / Barry Wels — Color: Adam Swiecky — Translation: Jaap Weel