What Every Computer Scientist Should Know About Security

David Evans
University of Virginia
http://www.cs.virginia.edu/evans/

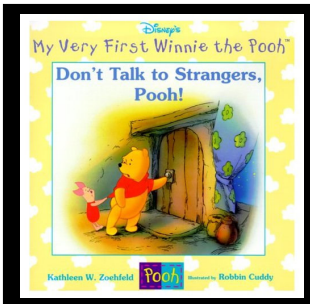
cs290 Spring 2008
21 February 2008

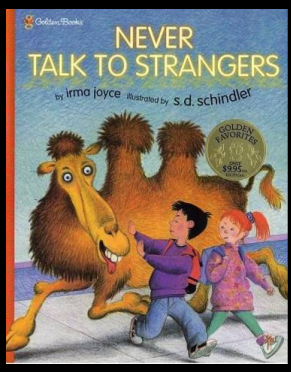
What Every *Human* Should Know About Security

David Evans
University of Virginia

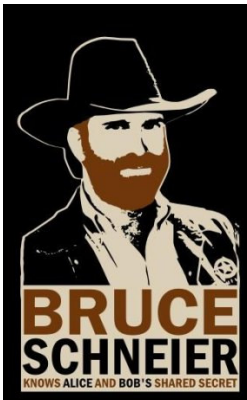
cs290 Spring 2008
21 February 2008

www.cs.virginia.edu/evans/

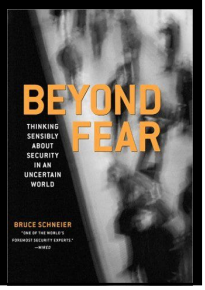


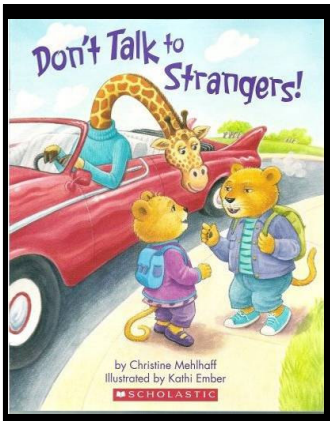


4



"Many children are taught never to talk to strangers, an extreme precaution with minimal security benefit."





"Emma Lion loves to make new friends, but Mama tells her to be careful and never talk to strangers. Emma sees new people to meet everywhere she goes. How will she know who is a stranger?"

6

Security Research

study of systems in the presence of *adversaries*

about what happens when people **don't** follow the rules

7

Security

- Technical questions
  - Figuring out who is not a "stranger" (*authentication*)
  - Controlling access to resources (*protection* and *authorization*)
- Value judgments
  - Managing risk vs. benefit (*policy*)
- Deterrents
  - If you get caught, bad things happen to you

Protecting assets from misuse

8

Quiz

Authentication, Protection, Authorization, Policy, or Deterrent?

9

Authentication, Protection, Authorization, Policy, or Deterrent?

10

Authentication, Protection, Authorization, Policy, or Deterrent?

Charlottesville Airport, Dec 2001

Authentication, Protection, Authorization, Policy, or Deterrent?

British Parliament, Dec 2007

Principles for Designing and Building Secure Systems

13

Principle 0:

Know what you are protecting, and what the threats are.

14

Principle 1:

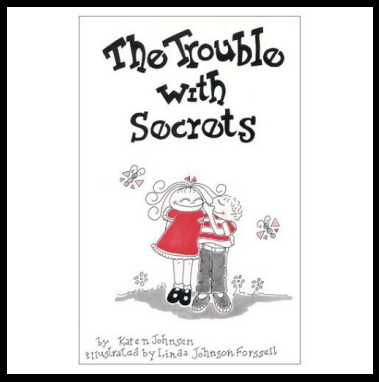
Keeping secrets is hard.

Design systems so that they don't depend on secrets being kept from adversaries.

If you need a secret, make it small and well defined.

Kerckhoffs' principle, 1883

16



17

Authorization with "Secrets"

18

Authentication/Authorization using Secrets

Who are you?

ID=#5187230

Generate random number r

Prove it by encrypting r

Compute $x=E_K(r)$

x

Check $x=E_K(r)$

19

Encryption

- E is an encryption function: algorithm for scrambling bits in a way that depends on K
- K is a secret key shared between card and reader (backend database)

$$x=E_K(r)$$

Threat: Eavesdropper cloning cards.
Requirement: attacker who overhears x and r (even many pairs) should not be able to guess K

20

Recall Principle 1:

Keeping secrets is hard.

Design systems so that they don't depend on secrets being kept from adversaries.

If you need a secret, make it small and well defined.

Security should depend only on keeping K secret (not on keeping algorithm or protocol secret)

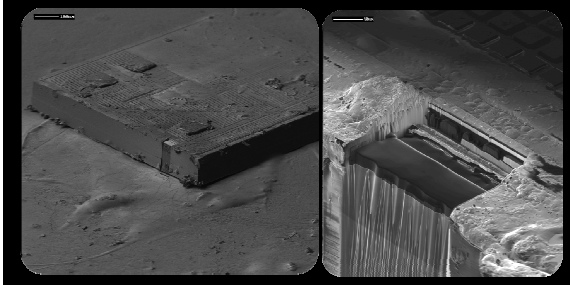
21

NXP Mifare Classic RFID Tag

- Billions of units deployed
- Cost < $1
- Planned for nationwide deployment in Netherlands
- Uses secret encryption algorithm

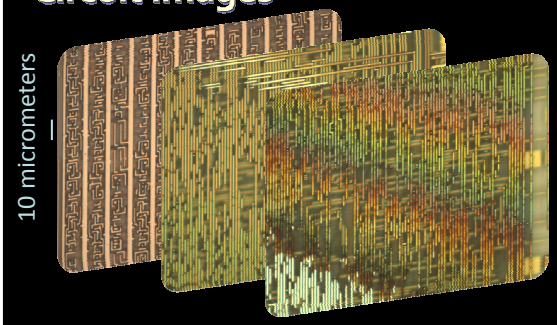
Reverse-Engineering a Cryptographic RFID Tag.
Karsten Nohl, David Evans, Starbug, Henryk Plötz, Feb 2008.

22



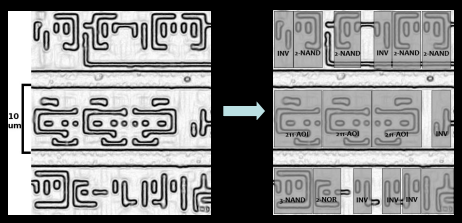
23

Circuit Images



24

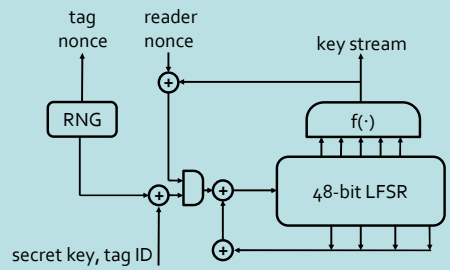
Reverse Engineer Circuit



Very error-prone and tedious process (but can be automated)

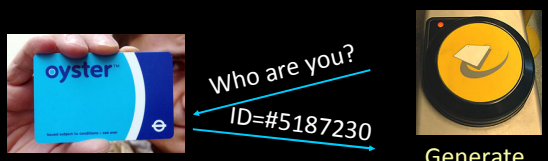
25

Mifare Crypto-1



26

Does knowing E matter?



Who are you?

ID=#5187230

Generate random number r

Prove it by encrypting r

Compute $x=E_K(r)$

x

Check $x=E_K(r)$

27

Failure of Security through Obscurity

- If there are enough possible keys, it shouldn't matter if attacker knows E (if it is a good encryption algorithm)
- Mifare algorithm has several weaknesses that made it easier to find K without needing to try all possible keys

28

"The debate about the OV-Chipcard is symbolic, as we discovered yesterday at the hearing, for the choice: open or closed software. The bankruptcy of closed source software shone solidly in the limelight yesterday. GroenLinks [a Dutch political party] has urged for a long time that the government should work with open-source software. Yesterday it became painfully obvious that we may pay a high price due to the fact that this advice was not followed for the OV-Chipcard."
– Dutch parliamentary discussion, January 16th 2008

29

Principle 2: Need to Know Basis

Once you release information, there is no way to get it back.

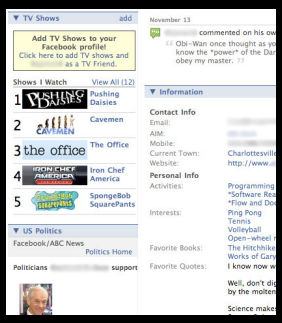
Give out as little as necessary to enable useful things.

Principle of "Least Authority"

30

Facebook Platform

Generated by "untrusted" third party



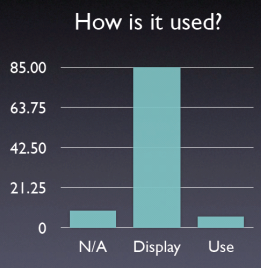
Generated by Facebook

Privacy-by-Proxy. **Adrienne Felt**, David Evans, Feb 2008.

31

32

Application Needs



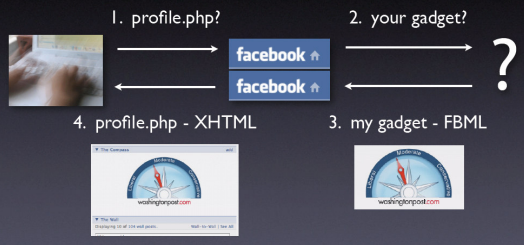
Recall Principle 2: Need to Know Basis

Once you release information, there is no way to get it back. Be careful about what you release.

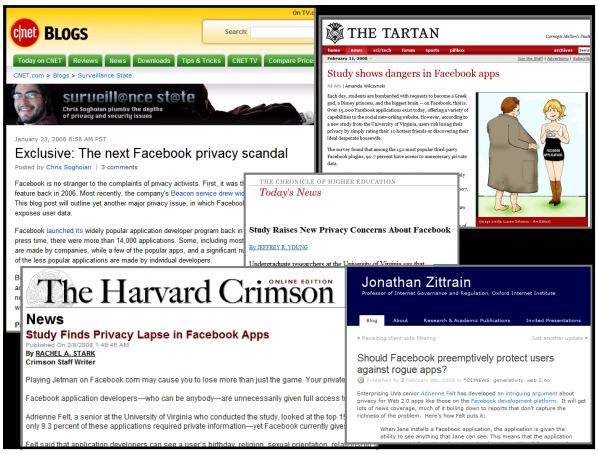
No need to give applications access to information they don't use

34

Proxying



36



from c|net article:

I asked Facebook's [chief privacy officer] Kelly what his company is doing to ensure that application developers do not violate the rules by saving a copy of user data that passes through their servers. He cited "extensive security mechanisms operating behind the scenes," although, he refused to expand on this, due to "security reasons." He wasn't too happy when I accused him of practicing **security though obscurity**, a concept widely mocked in security circles. He dismissed my charge as a mischaracterization.

38

Recap

- Principle 0: Know what you are protecting, and what the threats are.
- Principle 1 (security-through-obscurity doesn't work): Design systems so that they don't depend on secrets being kept from adversaries.
- Principle 2 (least authority): Give others what they need to do useful work, but not more.

Not everything you need to know about security

39

Questions

http://www.cs.virginia.edu/evans
My home page
http://www.cs.virginia.edu/wheel
Research group blog

If you steal property, you must report its fair market value in your income in the year you steal it unless in the same year, you return it to its rightful owner.

Your Federal Income Tax, IRS Publication 17 (p. 90)

40