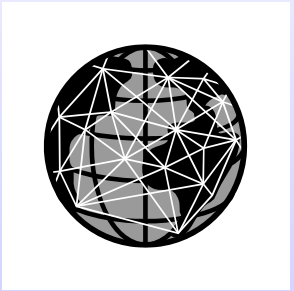
Hot Topics in Computer Security

Iliano Cervesato

http://www.qatar.cmu.edu/~iliano

Computer Security

- Networked computer systems
  - Provide fast access to lots of information
    - Information society
  - Higher productivity
  - Much higher convenience
- Substantial opportunity for abuse

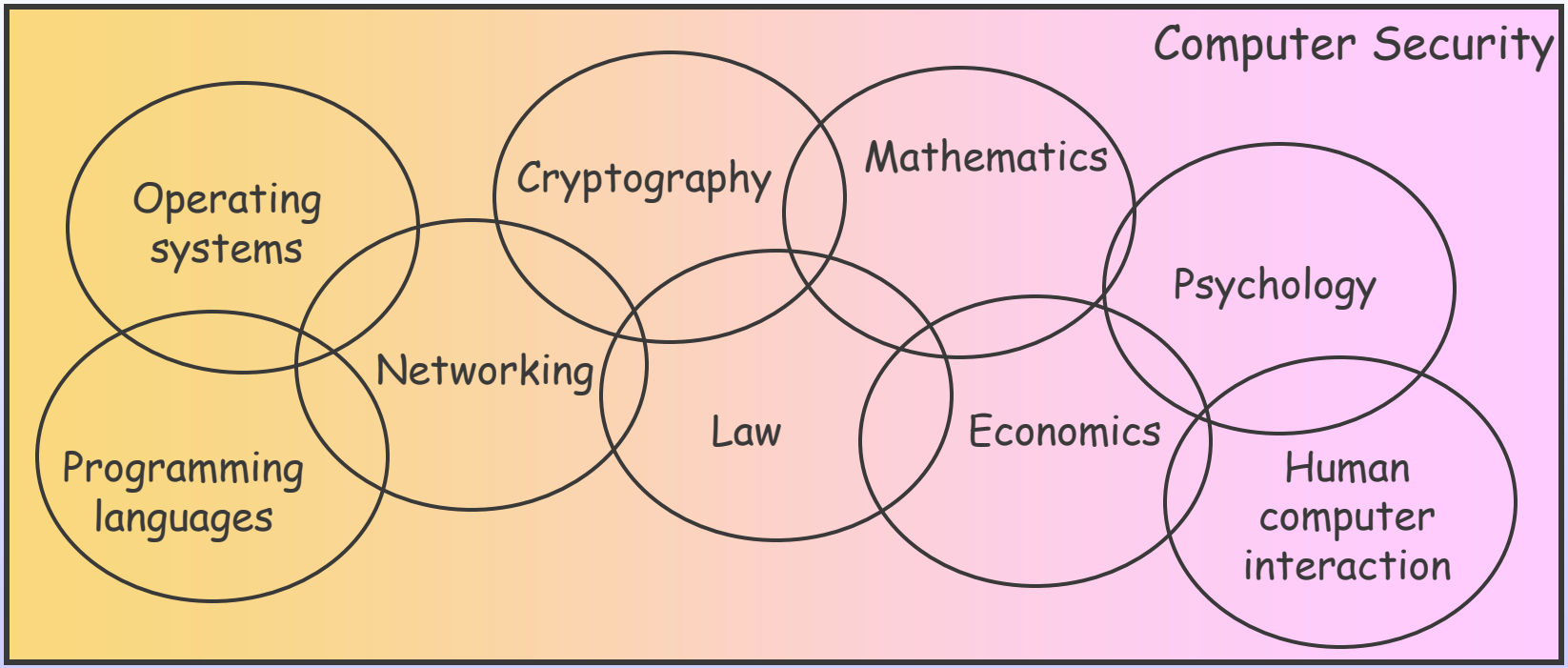


- Computer security
  - Mitigate risk
  - Prevent disruption, fraud, …

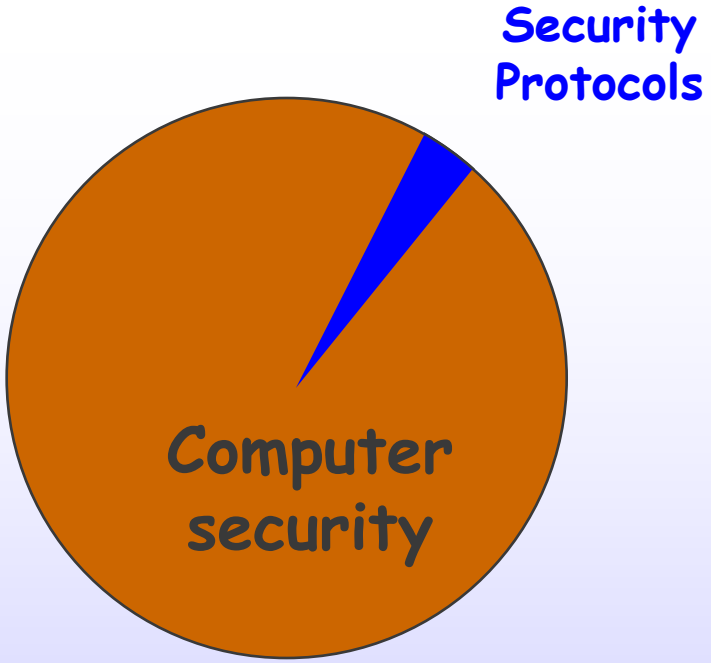
Is Cryptography the Solution?

Cryptography is not the same as security

- No crypto today
- 85% of all CERT advisories cannot be fixed by crypto
- 30-50% of recent security holes from buffer overflow



Computer Security is a Big Field!



- We are going to look at a tiny speck
- Security Protocols

Outline

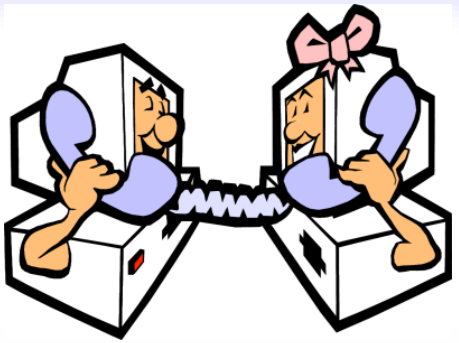
- What are security protocols?
- What can go wrong?
- Where is protocol verification now?
- What are the open questions?

Protocols

Expected behaviors when engaging in communication

- When 2 people want to talk
  - Buying something at the souq
  - Going on a date
  - Calling up your friend, …
- When interacting with an organization
  - Bureaucracy
  - Official visits by head of states, …
- …
- When computers want to talk

Computer Protocols



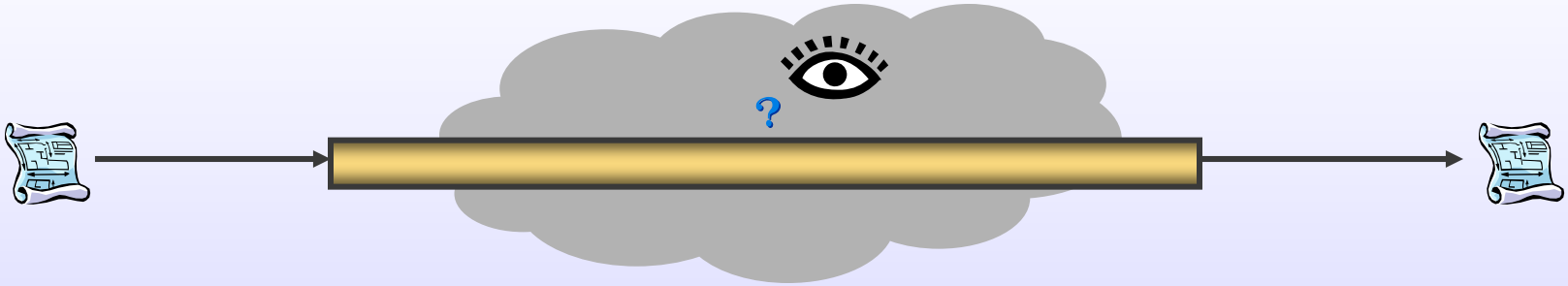
- What sets them apart?
  - No human involved!
    - Automated
    - Inflexible
    - No common-sense
- What protocols are there in a computer?
  - Hundreds!
  - Communication protocols
    - Email, http, Ethernet, …
  - Security protocols

Security Protocols

- Communication protocols ensure that communication actually happens
- Security protocols ensure that communication is not abused
  - Protect contents
  - Protect communicating parties
  - Protect intent of communication
  - Protect possibility of communication

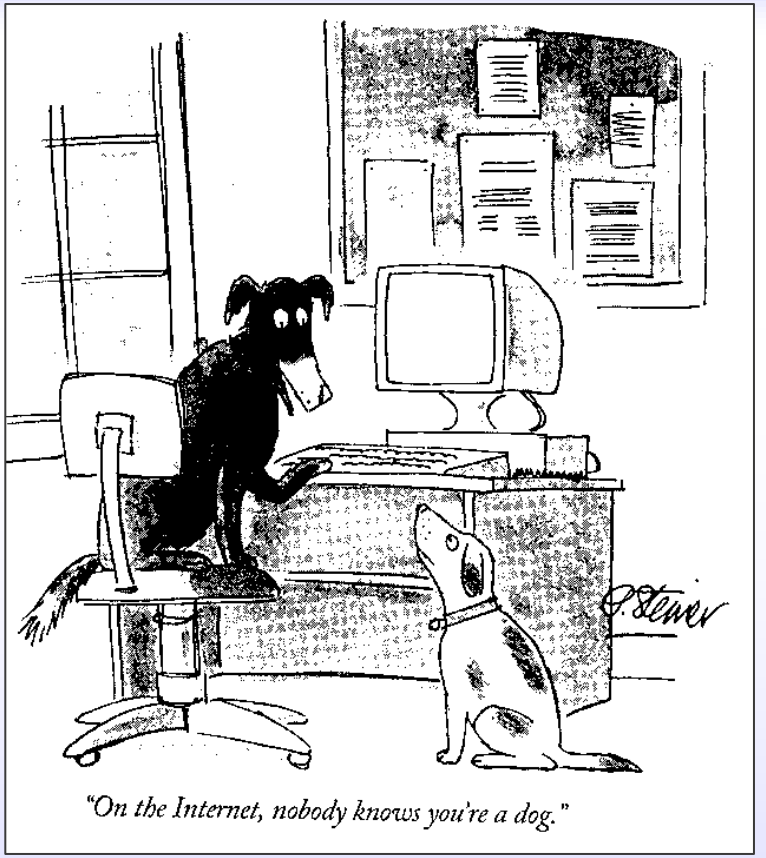
Common Security Goals

- Confidentiality
  - Message cannot be observed in transit



  - Achieved using some form of encryption

Authentication



"On the Internet, nobody knows you're a dog."

- Ensure that we are talking with who we think
  - Much more subtle than secrecy
  - How to establish a secret channel in the first place
    - Negotiate parameters of channel
    - Ensure channel remains trusted
- Authentication protocols

Other Security Goals

- Non-Repudiation
  - Party cannot claim he didn't do it
  - For auditing, electronic contract signing, …
- Non-Malleability
  - Message cannot be changed en route
  - For electronic voting, …
- Anonymity
  - Hide who is communicating
- Availability
  - User can always get through
- …

Example: Kerberos

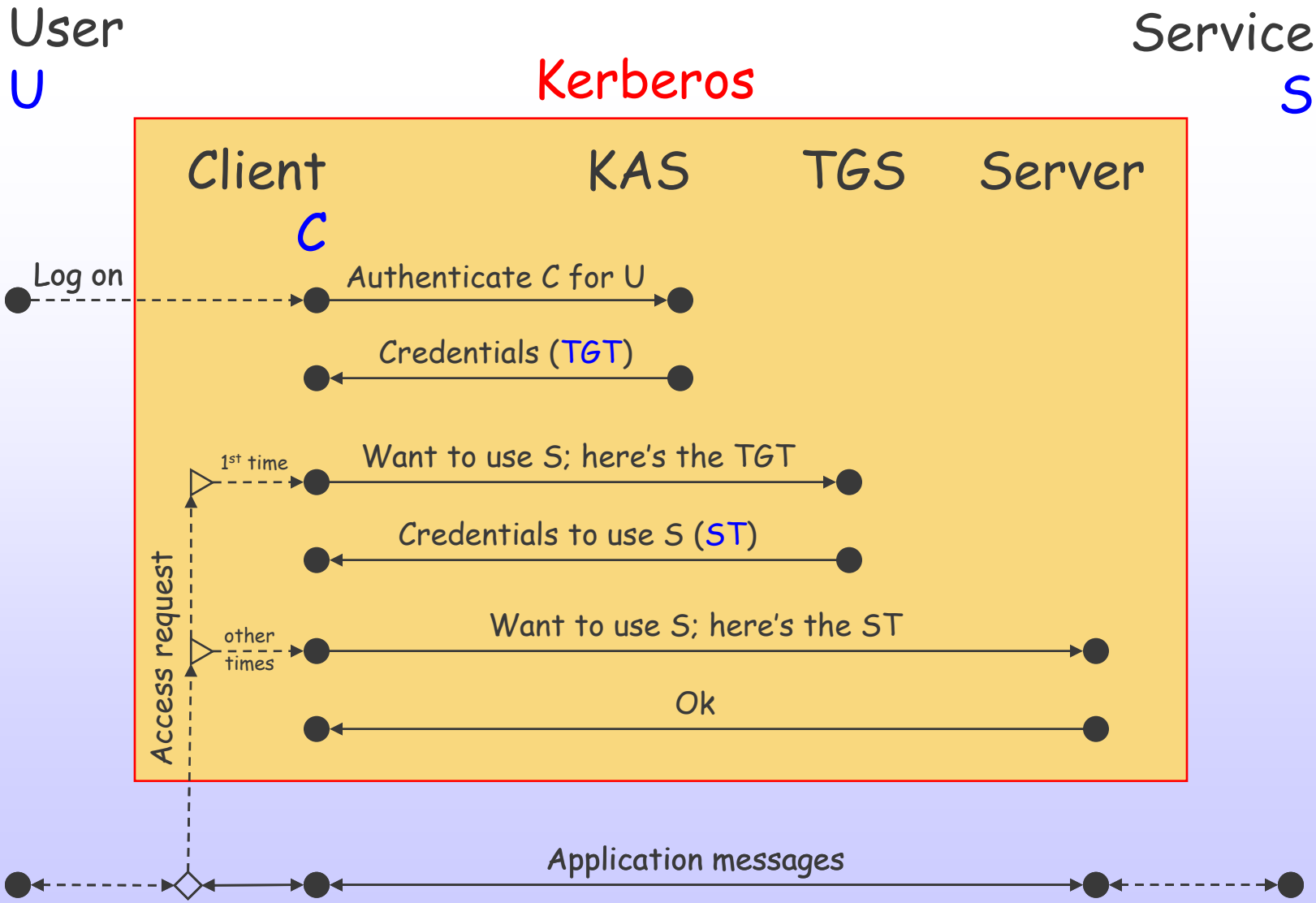
- Log in to your computer
- Access other computers without logging in again
  - Email, "i-drive", printers, directory, …

  … for 1 day

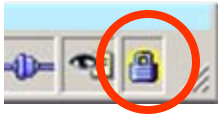
- Goals
  - Repeatedly authenticate a client to multiple servers
  - Transparent to user

- Ubiquitous

How Kerberos works



Other Popular Protocols

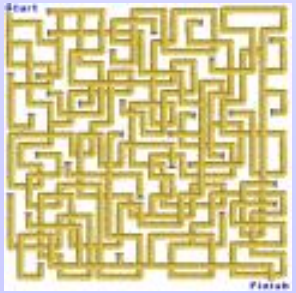


- SSL / TLS protocol
  - Authenticates client to server
  - Encrypts communication
  - HTTPS (secures web page)
  - Secure email download (POP3S, IMAPS)
- SSH protocol
  - PuTTY (Log to remote computer, copy files, …)
- PGP
  - Send encrypted/authenticated email
  - Enigmail

What is there to care about?

The Problem

- Security protocols are extremely hard to get right
  - Minuscule programs
  - Extremely complex interactions
    - Bugs can take years to discover
  - Generally it's not the crypto
  - It's the piping



Correctness vs. Security

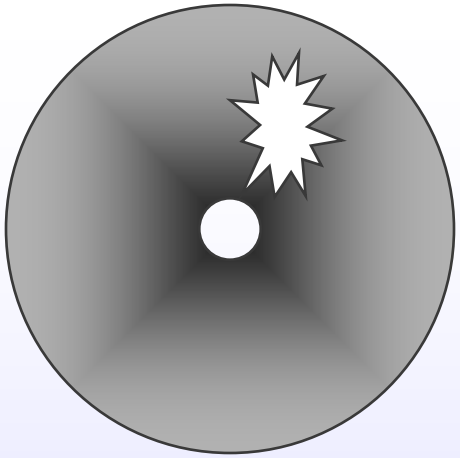
- Correctness: satisfy specifications
  - For reasonable inputs, get reasonable output

- Security: resist attacks
  - For unreasonable inputs, output not completely disastrous

Difference:

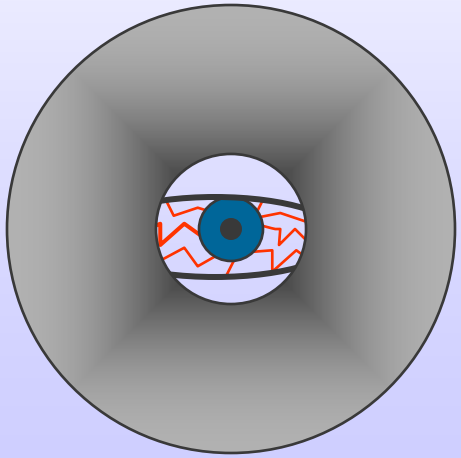
- Random events vs. active attacker

Attacks

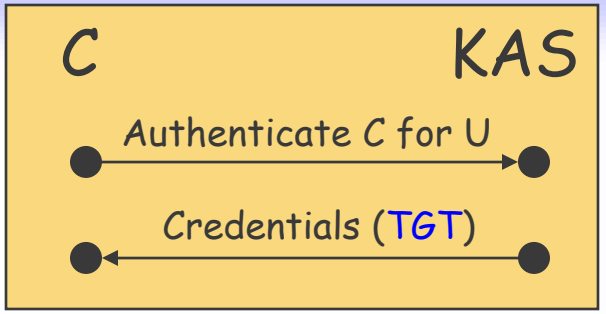


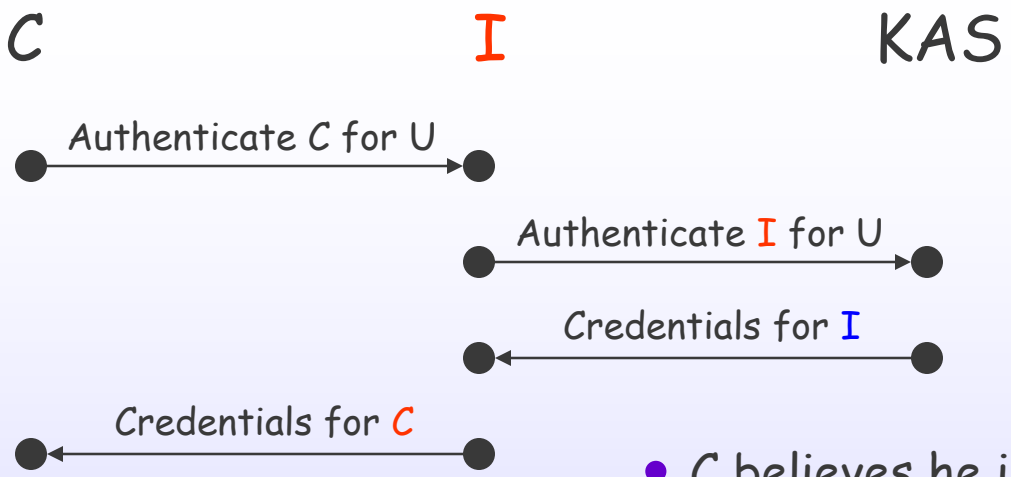
- Attacker can break secrecy of the channel

- Attacker can break authentication
  - Got the piping wrong



Example: Kerberos

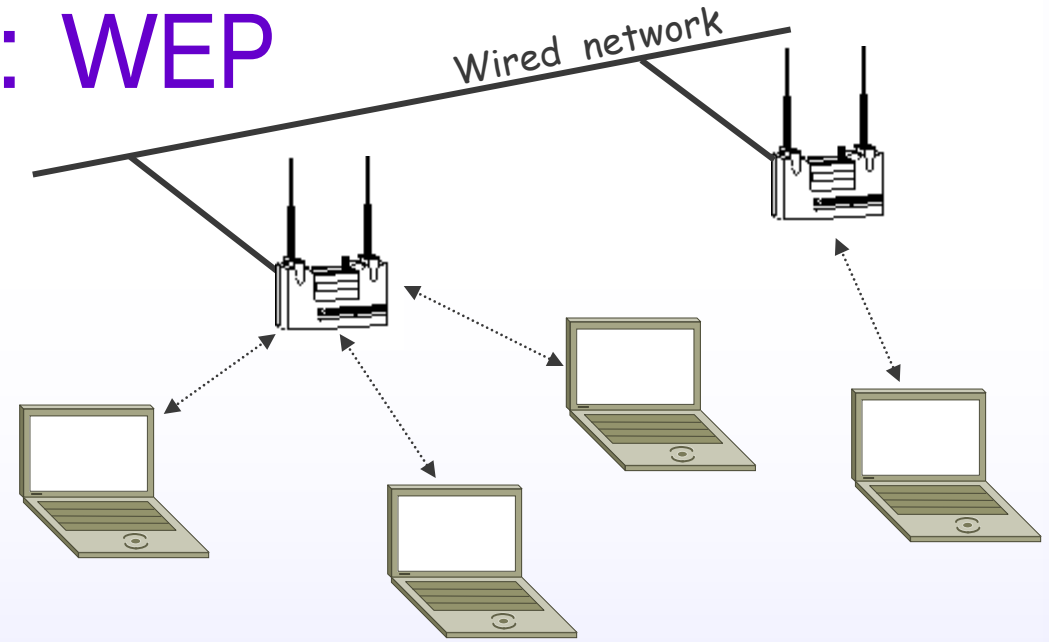




- C believes he is talking to KAS
- KAS believes he is talking to I
- I knows the key that C obtained from KAS

- Discovered 10 years after exchange was designed
- Immediately fixed in all implementations

Another one: WEP



- Standard wireless network
  - Principally a communication mechanism
  - Has built-in security protocol: WEP
    - Confidentiality (prevent eavesdropping)
    - Access control (prevent unauthorized access)
    - Integrity (prevent tampering with messages)

  Fails at all 3!

WEP Authentication

- Should you stop using WiFi? NO!!!
  - Fine communication suite
  - Use standard protocols on top of it
  - (now replacements to WEP are available)

State of the Art in Protocol Verification

Protocol Analysis

- Ensure that protocol does not have flaws
  - Formal verification
    - Mathematical scrutiny so that nothing bad can happen
  - Secure-by-design
    - Securely compose secure building blocks
  - Testing is not an option!
    - Assumes statistical distribution of errors
    - Security is about worst-case scenario

Formal Verification

- Model checking
  - Show that no bad things can happen
  - Try everything attacker can do to break security goals
    - Fast setup
    - Discovers attacks (but often only partial assurance)
- Theorem proving
  - Show that only good things can happen
  - Mathematical proof that protocol meets security goals
    - Absolute assurance (but no attacks)
    - Extremely time consuming
- Hybrid approaches

Things to Be Made Precise

- What the protocol does
- Security goals
- Attacker capabilities

- Framework to draw general conclusions

Protocol Specification Languages

- Initially, just English
- Till mid 90's: ad-hoc languages
- Since then, several well-understood languages with deep roots in theory
  - MSR

To a large extent, problem solved

Security Goals

- 5 years to define "secrecy"
- 10 for "authentication"
  - Standard notions now well-understood
  - General understanding still shaky
- Usually expressed as logical statements
  - Perfect language has not been found yet

What can an Attacker do?

- Dolev-Yao model
  - Controls the communication medium
  - Can decrypt/encrypt only with known keys
  - ➢ Tractable, but idealizes crypto
- Computational model
  - Can apply computational methods to gain partial information
  - ➢ More precise
  - ➢ But no mathematical tools till recently

What we Know about Security

- Protocol verification is undecidable
  - Apparently decidable for typical protocols
- Dolev-Yao intruder derivable from protocol
- Secrecy and authentication build on each other

What can we Verify?

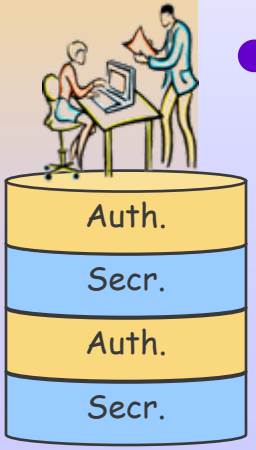
- Lots of toy protocols
  - Now very fast
- A couple in the computational model
- A few commercial protocols manually
  - Kerberos

- Extremely fast progress recently

Open Questions

Understanding Security

- What is protocol security?
  - Much better understanding than 10 years ago in common cases
  - Still pre-scientific stage
- What should the security goals be?
  - General theory
  - Interplay
- Come up with general and usable language for
  - Security goals
  - Security assumptions



Protocol Composition

- Putting 2 good protocols together is no guarantee to get a good protocol
  - When is it the case?
- Modular approach to protocol analysis / construction
  - Start with well-understood building blocks
  - Combine them into desired protocol
- Recent progress in this direction
  - Protocol derivation
  - Still patchy
    - What do basic components do
    - Prove that only good things result from composition

Automation for Large Protocol

- 10 years ago, automated analysis was struggling with toy protocols
  - Now, can verify them very fast
- What about commercial protocols?
  - Threshold situation
    - Tools are almost good enough
  - Manual techniques are there
    - Need to be automated
  - Opportunity to have real-world impact
    - Have a say in protocol design

Qualitative Protocol Analysis

- Current approaches designed to answer yes/no
- Real-world does not work this way
  - Persistent/resourceful attacker can always break crypto
  - Developer can fine-tune parameters to get system more secure
  - Denial-of-Service has no yes/no answer
- Completely ignored by "traditional" protocol analysis research
  - First initial steps

Thank you!