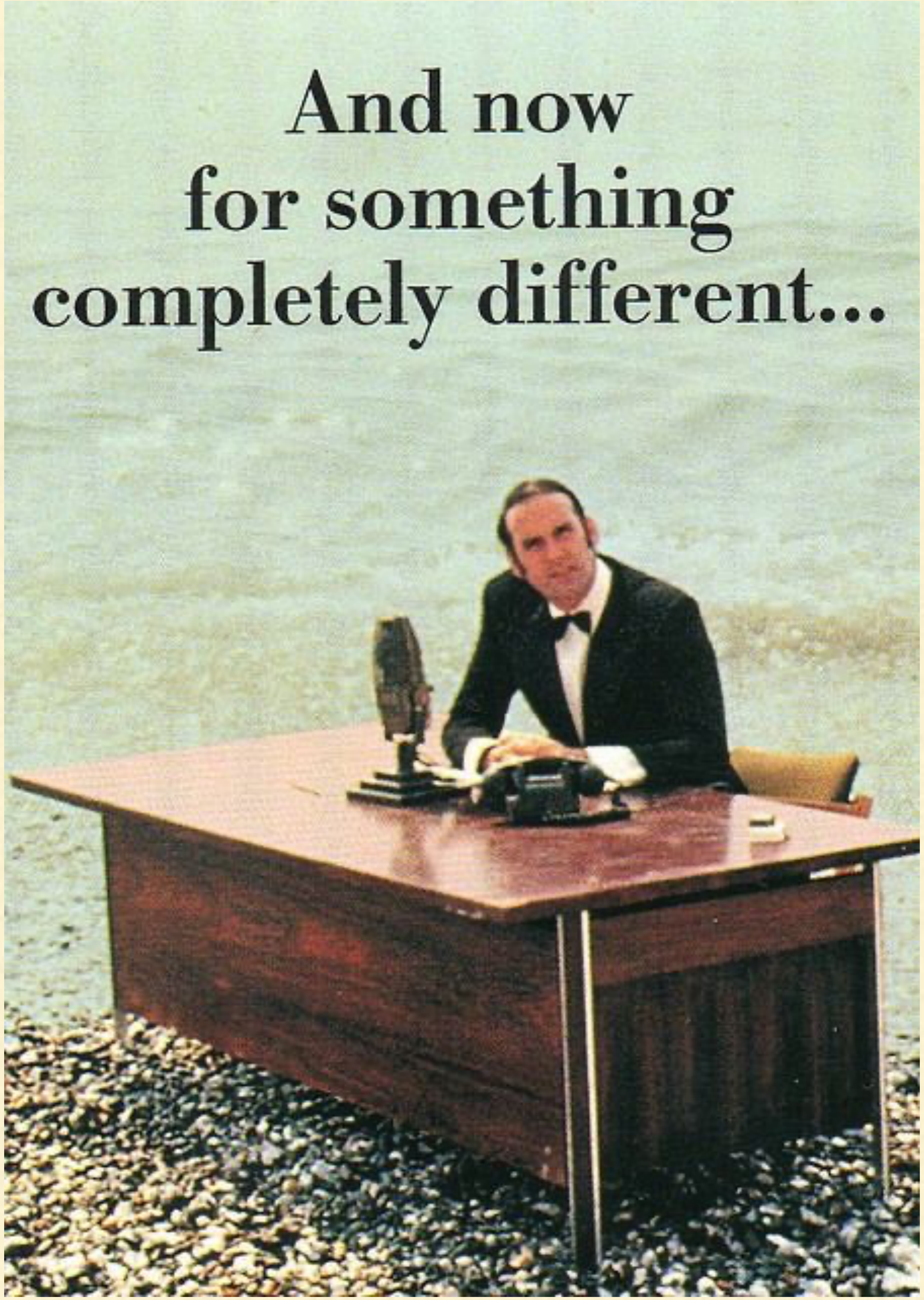
COL876: SPECIAL TOPICS IN FORMAL METHODS

Formal verification of security protocols

Lecture 10, 16 October 2023

RECAP

- Saw how to formally model and verify security protocols
- Various kinds of abstract models: transition systems, applied-pi, multiset rewriting…
- Different classes of properties: trace, equivalence
- Many tools built using symbolic verification: ProVerif, Tamarin…


And now
for something
completely different...

BUT WHAT ABOUT REAL LIFE?

- Do symbolic guarantees translate into "real-life" guarantees?
- Abstracted out a lot of information, so not necessarily!
- Abstract model may be "correct", but depends heavily on:
  - Perfect cryptography assumption
  - Faithful implementation

COMPUTATIONAL MODEL

- Not much control over implementations
- But we know how to specify protocols computationally
- Also know how to provide computational guarantees
  - Indistinguishability results
- Can we map some equivalences in the symbolic model to indistinguishability in the computational model?
- Easy for pairing. What about symmetric encryption?

ENCRYPTION SCHEMES

- An encryption scheme Π, is a triple of PTIME algorithms (K, E, D)
  - K is the key generation algorithm
    - input: parameter, coins output: key
  - E is the encryption algorithm
    - input: key, string, coins output: ciphertext
  - D is the decryption algorithm
    - input: key, string output: plaintext
  - D(k, E(k, m, r)) = m if m is a valid plaintext, ⊥ otherwise

NEGLIGIBLE ADVANTAGE

- Probabilistic PTIME adversary A
- Need to evaluate advantage in distinguishing between strings from two different distributions D and D'
- Advantage is a function from parameters to reals. Hope that this value is "negligible"
- A function $f: N \rightarrow R$ is negligible if, for all $c > 0$, there exists an N_c such that $f(n) \leq n^{-c}$ for all $n \geq N_c$.
- Advantage $f(n) := Pr[\, x \leftarrow D \mid A(n, x) = 1 \,] - Pr[\, x \leftarrow D' \mid A(n, x) = 1 \,]$

DESIRABLE ASPECTS OF ENCRYPTION

- Repetition concealing: Given ciphertexts c and c′, should not be able to tell if their underlying plaintexts are equal.
- Which-key concealing: If I encrypt messages using various keys, should not be able to tell which messages were encrypted using the same key.
- Message-length concealing: A ciphertext should not reveal the length of its underlying plaintext.

ASPECTS OF ENCRYPTION

- Can have schemes which do not meet one or more of these criteria
- Can have encryption which
  - reveals the length of the plaintext, or
  - reveals which key is being used, or
  - reveals if two ciphertexts are obtained from same plaintext
- Some combinations are better than others!

IMPORTANT

- Have to consider protocols without “encryption cycles”
- Cannot encrypt a key with itself — even via circuitous routes
- Schemes with encryption cycles are breakable (shown by Goldwasser and Micali)
- Fix (K, E, D), a parameter n, and an adversary A

ORACLES (PART 1)

- Pick two keys k, k' from K(n)
- Left oracle: On query m, computes encryption of m using k
- Right oracle: On query m, computes encryption of m using k'
- "Good encryption": Encrypts query m using one of two keys
- Pr_1: Adversary interacts with these oracles and outputs 1

ORACLES (PART 2)

- Pick a key k from K(n)
- Both oracles: On query m, compute encryption of 0 using k
- "Bad encryption": Encrypts 0 using key k
- Pr_2: Adversary interacts with these oracles and outputs 1

ADVANTAGE

- Advantage of adversary: $Pr_1 - Pr_2$
- "How well can the adversary distinguish good encryption from bad?"
- Need this to be negligible!

COMPUTATIONAL SOUNDNESS

- Want to map symbolic terms to distributions over strings
- Map symbolic attacks to non-negligible adversary advantage
- Need to keep track of adversary "view"
- "What can an adversary learn from an encrypted term?" "Patterns"
- Equivalence of patterns == Indistinguishability of ciphertexts