ISE Cryptography — Lecture 10

Closing Session

Recap

What did we learn?

What We Covered

- *A Graduate Course in Applied Cryptography* is a graduate textbook (hence the name)
  - You covered it in second year!
- **Classical crypto** (you blew it up in about 10 minutes in the first lab)
- **Symmetric crypto** (stream ciphers, block ciphers)
- **Integrity & hashing** (MACs, hash functions, HMAC)
- **Public-key & signatures** (RSA, DH, ElGamal, digital signatures)
- **Real protocols** (authenticated encryption, KDFs, Signal)

The Last Word on Security

AES Has No Security Proof

- AES (since 2001) and ChaCha20 (since 2008) are trusted because sustained cryptanalysis has found no *practical* break
  - There are good arguments for why they should be secure
    - Such as AES's wide-trail argument
  - But there's no proof they're hard to break!
- There is no ***formal reduction***
  - No theorem says "breaking AES is at least as hard as solving problem X"
- RSA and Diffie-Hellman are a step up
  - Their security ***reduces*** to well-studied problems (factoring, discrete log)
  - One direction only: solving the problem breaks the scheme, but the converse is unproven
- But those problems are only ***conjectured*** to be hard
  - On a large-scale quantum computer, Shor's algorithm solves both efficiently
  - So the assumption fails against a CRQC (but no such machine exists yet)

P vs NP

- All computational crypto rests on an *assumption*, not a theorem
- Some problems are hard to solve...
  - ...but easy to *verify*
- Finding x with $f(x) = y$ is hard for some f...
  - ...but given x, checking it is a single evaluation of f
- P: problems solvable in polynomial time
- NP: problems whose solutions are *verifiable* in polynomial time
- Computational security assumes no PPT adversary can win
- Open question: is $P = NP$?

What Breaks If $P = NP$?

- Factoring: hard to solve, easy to verify (multiply the factors back)
- Factoring is in NP, but RSA assumes it's *not* in P
- If $P = NP$, factoring is in P, and RSA falls
- Discrete log shares the fate: DH and ElGamal collapse too
- Nearly all public-key crypto breaks overnight
- Symmetric crypto is subtler...
  - One-way functions underpin *all* computational security
  - One-way functions existing *implies* $P \neq NP$
  - So $P = NP$ kills symmetric crypto too, just less directly
- One of the seven Millennium Prize problems
  - Open 50+ years; $1,000,000 if you solve it

Does $P = NP$?

- Most experts (80%+) believe $P \neq NP$, with varying confidence
  - We don't actually know
  - A few contrarians take $P = NP$ seriously, e.g. Donald Knuth
- All proofs are equal, but some are more equal than others...
- A proof of $P = NP$ need not hand you an algorithm
  - ***Non-constructive***: shows an algorithm must exist, builds none
  - ***Galactic***: builds one, but the constants make it unusable
- So $P = NP$ could be *true* and change nothing in practice
  - Roughly Knuth's actual position

After the Module

B&S Chapter Exercises

- The B&S exercises at the end of each chapter are excellent
  - This is how you build formal proof-writing ability!
- **Using AI tools productively:** never ask for the answer!
  - Write your attempt first
    - Commit to a direction before asking anything
  - Ask: “Is this argument correct? Where does it break?”
    - Use the model as a judge, not an oracle
  - Iterate
- The goal is to develop the skill by doing the hard part yourself
  - The model is a tutor, not a crutch

Further Reading

- Boneh & Shoup online (toc.cryptobook.us)
- B&S video lectures (available on YouTube)
- IACR ePrint, the preprint archive for cryptography research

Final-Year Projects in Cryptography

- If you're targeting an A1 or A2 and you enjoyed this material, get in touch.
- Topics span a spectrum:
  - **Mostly code:** PQ-secure key exchange, Signal-compatible client, OPAQUE (the password protocol), building a CTF challenge set
  - **Mixed:** benchmarking ML-KEM vs ECDH, side-channel analysis of AES implementations
  - **Mostly maths:** lattice reduction algorithms, formal proofs in the B&S framework, NIST PQC standardisation analysis
- **You have to drive it.** If you wouldn't read a paper on the topic on your own initiative, it's not the right FYP for you.

CryptoHack

- CryptoHack: dedicated crypto challenge platform
  - Structured by topic: stream ciphers, block ciphers, RSA, elliptic curves, hash functions, lattices
  - Archive of crypto challenges from past CTF competitions worldwide
  - Free, beginner-friendly, deep archive
- Start with the **Introduction** challenges, then pick your topic.
- Unlike general CTF platforms, CryptoHack is entirely crypto.
- The skill transfer from this module is as direct as it gets
  - With a bit more focus on cryptanalysis and attacks!

ISE CTF

- Idea: run a crypto CTF event within ISE.
  - Competing
  - Helping build challenges (writing problems, setting up infrastructure)
- Building challenges is a great way to learn and deepen your understanding of a topic
  - You have to understand a concept deeply enough to construct a solvable puzzle around it!
  - Same goes for competing!
- If people would be interested, get in touch

Questions?

Ask now, catch me after class, or email eoin@eoin.ai