Announcements

- No Problem Set this week
- My goal is to have Exam 3 ready by 11/23, due on 12/3
  Know the end of semester will be weird with exam period remote after Thanksgiving
- Explicitly think about what your take-aways from Crypto this semester will be

  In addition to content, think about the way of thinking / problem-solving and how it might apply to other academic pursuits and how you approach problems in general

  In other words, how has Math 202 changed your life?

Digital Signature Algorithm, 160-bit

Key creation - Alice

- Find 1024-bit prime p,
  160-bit prime q where q divides $p-1$
- Find $\alpha \in \mathbb{Z}_p^*$ where $\text{ord}(\alpha) = q$
- Choose private d where $0 < d < q$
  Compute $\beta \equiv \alpha^d \mod p$
- Publish (p, q, α, β)

Sign message x - Alice

- Choose ephemeral k_E where $0 < k_E < q$
- Compute
  $$r \equiv \left(\alpha^{k_E} \mod p\right) \mod q$$
  $$s \equiv \left(\text{SHA}(x) + dr\right) k_E^{-1} \mod q$$
- Send $(x, (r, s))$

Verify signature - Bob

- Compute
  $$w \equiv s^{-1} \mod q$$
  $$u_1 \equiv w \cdot \text{SHA}(x) \mod q$$
  $$u_2 \equiv w \cdot r \mod q$$
  $$v \equiv \left(\alpha^{u_1}\beta^{u_2} \mod p\right) \mod q$$
- If $v = r$ then valid
  If $v \neq r$ then invalid

Some shortcomings of digital signatures

- Every single message between Alice and Bob should be signed
- In particular, every 128-bit AES block should be signed
- Signatures are necessarily asymmetric (e.g. RSA, DSA) and less efficient than symmetric like AES
- Motivation for *Message Authentication Codes*, or MACs

Message Authentication Codes

- Uses symmetric keys so faster in implementation
- Keys used for only that one session
- Based on hash functions or block ciphers, like AES
- Assumes symmetric key has been securely exchanged
- Also called *keyed hash functions*

Example: HMAC-SHA256, keyed-hash message authentication code

- Assume Alice and Bob have shared symmetric message key k
- For Alice to create MAC m for message x, concatenate k with x and hash:

$$m = \text{SHA2-256}(k||x)$$

  Alice sends (x, m)

- Bob can verify m since they have shared message key k
- Provides
  - Integrity: Can determine if message modified
  - Authentication: Only Alice has shared message key k
- Does not provide non-repudiation

WhatsApp uses HMAC-SHA256

- Signal protocol (`https://signal.org/docs/`) has super-clever idea of using a *chain key* and "ratcheting" forward after each use.

  Essentially,

  - Hash (chain key ||0x01) to get message key for HMAC-SHA256 with message
  - Hash (chain key with ||0x02) to get chain key to use with next message

- If key compromised, cannot work backwards to use with previous messages

- Search for "WhatsApp Encryption Overview" for technical white paper

- Some strong arguments for using Signal rather than WhatsApp