Secret Sharing

A secret sharing scheme allows a dealer to share a secret s among n parties $P_1, P_2, \ldots, P_n$ such that any authorized subset of parties can use their shares to reconstruct the secret

- All other subsets learn nothing about the secret from their shares

This is useful in cases where we need to distribute required trust while also allowing reconstruction even if some of the parties fail

We focus on a common access structure whcih is t out of n, or threshold secrete sharing

- Authorized subsets are all those of size at least t

t out of n Secret Sharing Scheme

A t -out-of-n secret sharing scheme consists of a pair of algorithms:

- `Share` is a randomized algorithm that takes as input a messages m and outputs a n-tuple of shares
- `Reconstruct` is a deterministic algorithm that given a t -tuple of shares, outputs a message m

Two properties have to hold:

- **Correctness**: the probability of reconstructing from any subset of size t is 1
- **Security**: for every m, m' and every subset I of size less than t, the distribution of the secrets looks the same for both m and m'

n out of n Secret Sharing Scheme

For a messages l bits long we have:

- `Share`: Choose at random $s_1, \ldots, s_n \in \{0,1\}^l$ such that XORing all of the messages equals m
- `Reconstruct`: XOR everything together

t out of n Secret Sharing Scheme

Try #1:

- For simplicity, suppose $t = 2$
- To share m, every pair of distinct parties generate 2-out-of-2 shares of m
  - Each party has $n - 1$ shares
- **Problem**:
  - If we do this for larger t, this results in the size scaling blowing up

Shamir Solution:

- Suppose we wish to share a message that is a single bit
  - To extend to ℓ bits, we will use the scheme ℓ times
- Choose a prime p
- `Share`:
  - Choose a random $t-1$ degree polynomial f in the field modulo p such that $f(0) = m$
    - This can be done by choosing a_1 through a_{t-1} randomly and setting $a_0 = m$
  - Output $(f(1), f(2), \ldots, f(n))$
- `Reconstruct`:
  - Solve a system of t linear equations that has t variables
  - These t players can noy only recover the secret m, but in fact the entire polynomial

Reed-Solomon Codes

The above secret-sharing scheme is similar to Reed-Solomon error-correcting codes

- A messages $m = (m_0, m_1, \dots, m_{t-1})$ is viewed as a unique degree $t-1$ polynomial f_m
- The codeword corresponding to the message is the evaluation of f_m on all points in the field:
  - $\mathrm{ECC}(m) = (f_m(0), f_m(1), \dots, f_m(p-1))$
- The codeword can then be uniquely decoded even if all but t of the coordinates in the codeword were erased