Secret Sharing

CS/ECE 407

Today's objectives

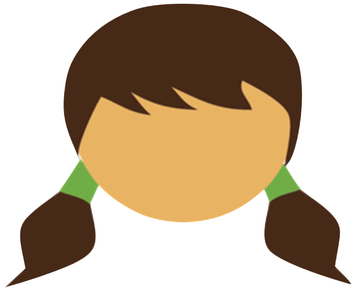
Introduce the notion of a secret-sharing scheme

Construct simple additive secret sharing scheme

Construct distributed public key encryption schemes/ signature schemes

Construct threshold secret sharing scheme

Secret Sharing



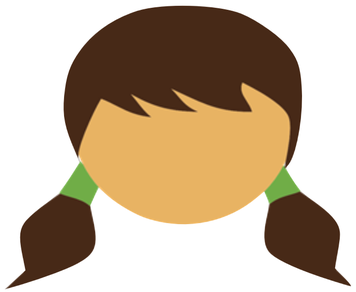
Alice

secret

Bob

Charlie

Secret Sharing



Alice

secret

???

Bob

secret0

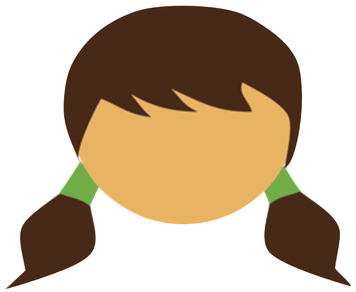
???

Charlie

secret1

Secret Sharing

Very roughly, a mechanism by which to distribute trust



Alice

secret

???

Bob **secret0**

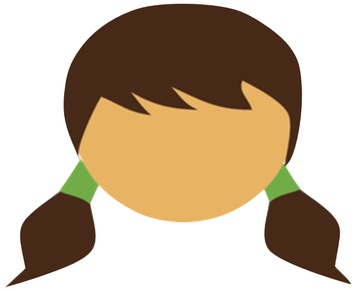
???

Charlie **secret1**

Secret Sharing

Very roughly, a mechanism by which to distribute trust

Access control, secure data storage, as a primitive for more complex tasks



Alice

secret

???

Bob **secret0**

???

Charlie **secret1**

Secret Sharing Scheme

Share(m) outputs n shares

Reconstruct(s_0,…,s_{t-1})
takes any t shares and outputs a message

Correctness:

Secret Sharing Scheme

Share(m) outputs n shares

Reconstruct($s_0,\ldots,s_{t-1}$)
takes any t shares and outputs a message

Correctness: $(s_0,\ldots,s_{n-1}) \leftarrow$ Share(m)

$(s'_0,\ldots,s'_{t-1})$ is an arbitrary subset of $(s_0,\ldots,s_{n-1})$

Reconstruct($s'_0,\ldots,s'_{t-1}$) = m

Secret Sharing Scheme

Privacy

Secret Sharing Scheme

Privacy

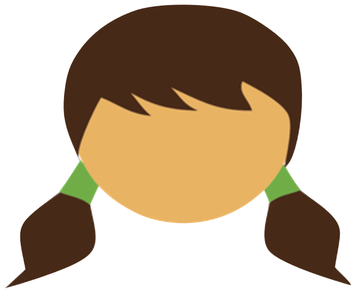
For all m0, m1 and for all $k < t$,

$$\left\{ (s'_0, \ldots, s'_{k-1}) \;\middle|\; \begin{array}{l} (s_0, \ldots, s_{n-1}) \leftarrow \mathrm{Share}(m_0) \\ (s'_0, \ldots, s'_{k-1}) \text{ is an arbitrary subset of } k \text{ shares} \end{array} \right\}$$

$$\equiv$$

$$\left\{ (s'_0, \ldots, s'_{k-1}) \;\middle|\; \begin{array}{l} (s_0, \ldots, s_{n-1}) \leftarrow \mathrm{Share}(m_1) \\ (s'_0, \ldots, s'_{k-1}) \text{ is an arbitrary subset of } k \text{ shares} \end{array} \right\}$$

Schnorr Signature



Alice

$sk \leftarrow\$ Z_q$

$r \leftarrow\$ Z_q$

g^r

$c = H(g^r, m)$

$s = r + sk \cdot c$

Bob

$pk = g^{sk}$

$g^r \cdot pk^c \stackrel{?}{=} g^{r+sk \cdot c}$

Threshold Secret Sharing Scheme

A size-t subset of shares can reconstruct the secret

ElGamal Public Key Encryption

Let g be the generator of some cyclic group G of order q

```
KeyGen():
  sk ←$ Z_q
  pk ← g^sk
  return (pk, sk)
```

```
Enc(pk, m ∈ G):
  y ←$ Z_q
  s ← pk^y
  c_1 ← g^y
  c_2 ← m·s
  return (c_1, c_2)
```

```
Dec(sk, (c_1, c_2)):
  s ← c_1^sk
  return c_2·s^-1
```

Today's objectives

Introduce the notion of a secret-sharing scheme

Construct simple additive secret sharing scheme

Construct distributed public key encryption schemes/ signature schemes

Construct threshold secret sharing scheme