MATH 314 Spring 2024 - Class Notes

4/29/2024

Scribe: William Naylor

Summary: In today's class we went over Signatures and message verification.

Notes:

Signatures:

Alice wants to prove to Bob that the messages are coming from her.

2 Steps for Signatures:

1. A signature is a number that only Alice can produce for a given message m
2. Alice sends (m,s)

Signature Algorithm:

- How Alice creates the signatures

Verifcation Algorithm:

- How Bob checks if the signature is valid for Alice and message

RSA Signatures:

- Basically the same as RSA encryption just "backwards"
- Alice generates a public key in the exact same way (n,e)
- Same public key $\rightarrow n = pq;\ gcd(e,(p-1)(q-1))$

Verification Step:

- Bob computes: S$^e(mod\ n) \rightarrow m \equiv S^e(mod\ n)$
- He checks if this is equal to m. If it is the signature is valid, otherwise the signature is moved.
- Only Alice can produce a valid signatures

El Gamal Signatures:

- Alice produces an El Gamal public key just like regular El Gamal

- Picks a large prime P

$\alpha =$ primitive root (mod p)
picks a random a $\rightarrow 2 \leq \alpha < p-1$
$\beta \equiv \alpha^a (mod\ p)$
public key is (p, α, β)

- If Bob were sending a message, he would pick an ephemeral key b $\rightarrow 2 \leq b < p-1$
$Ciphertext:$
$\texttt{r} = \alpha^b (mod\ p)$
$t = m\beta^b (mod\ p)$

- To sign a message m, Alice picks an ephemeral key k
- She computes:

$2 \leq k < p-1$ and gcd(k,p-1)=1

- Alice sends m along with the signature (r,s) to Bob
- Bob wants to verify this signature
- To verify, Bob checks if:

$\alpha^m \equiv \beta^r r^s (mod\ p)$
if it is the signature is valid, if not its invalid

- Why should this work?

unpack the RHS
$\beta \equiv \alpha^a (mod\ p)$
r $\equiv \alpha^k (mod\ p)$
S $\equiv (m - ar)k^{-1} (mod\ p-1)$