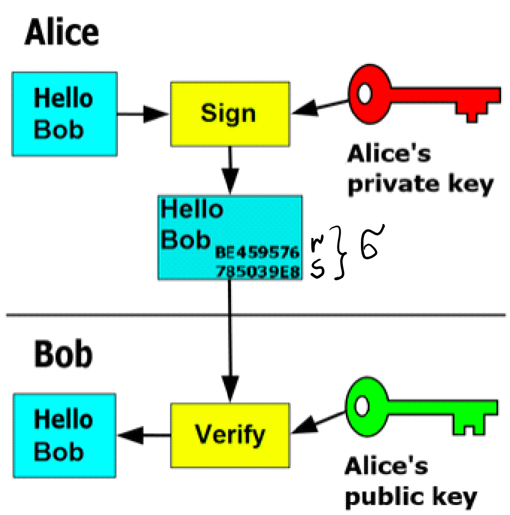
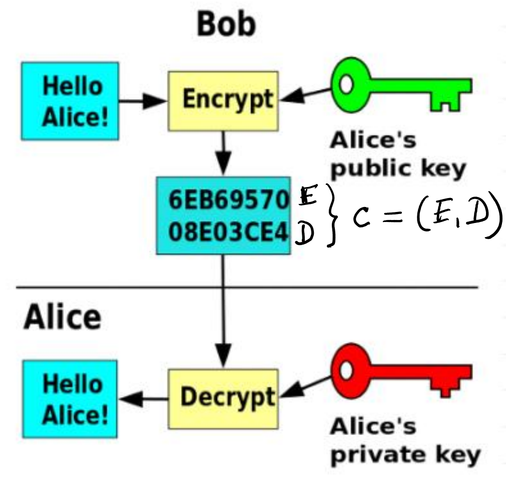
Signature creation-verification

Message encryption-decryption





p=268435019; g=2.

Public and Privete keys generation

Alice

```
>> x=int64(randi(p-1))
x = 27493765
>> a=mod_exp(g,x,p)
a = 38199862
```

Bob

```
>> y=int64(randi(p-1))
y = 2691421
>> b=mod_exp(g,y,p)
b = 28687908
```

ElGamal Signature

1.***Signature creation by* Alice**

To sign any finite message ***M*** the signer performs the following steps using public parametres **PP**.

- Compute **h=H(*M*)**.
- Choose a random ***k*** such that $1 < k < p - 1$ and gcd$(k, p - 1) = 1$.
- k^{-1} **mod (p-1)** computation: k^{-1} **mod (p-1) exists** if gcd$(k, p - 1) = 1$, i.e. **k** and **p-1** are relatively prime.

k^{-1} can be found using either Extended Euclidean algorithmt or Euler theorem or .....

>> k_m1=mulinv(k,p-1) % k^{-1}mod (p-1) computation.

```
>> m='Hello Bob'
m = Hello Bob
>> h=hd28(m)
h = 198770750
>> k=int64(genprime(28))
k = 179693671
>>
>> gcd(k,p-1)
ans = 1
>>  k_m1=mulinv(k,p-1)
k_m1 = 182658757
>> mod(k*k_m1,p-1)
ans = 1
```

- Compute **r=g^k mod p** ($r = g^k \bmod p$)

```
>> r=mod_exp(g,k,p)
```

- Compute **$r=g^k \bmod p$**
- Compute **$s=(h-xr)k^{-1} \bmod (p-1)$ -->**

$h=xr+sk \bmod (p-1)$,

Signature **$\sigma=(r,s)$**

2.***Signature Verification by* Bob**

A signature **$\sigma=(r,s)$** on message **M** is verified using Public Parameters **PP**=(**p**, **g**) and **PuK_A=a**.

1. Bob computes **h=H(M)**.
2. Bob verifies if **$1<r<p-1$** and **$1<s<p-1$.**
3. Bob calculates **$V1=g^h \bmod p$** and

$V2=a^r r^s \bmod p$, and verifies if **V1=V2.**

The verifier Bob accepts a signature if all conditions are satisfied and rejects it otherwise.

```
>> r=mod_exp(g,k,p)
r = 232941370
>> xr=mod(x*r,p-1)
xr = 151841508
>> hmxr=mod(h-xr,p-1)
hmxr = 46929242
>> s=mod(hmxr*k_m1,p-1)
s = 112441390

>> mm='I hate you'
mm = I hate you
>> h=hd28(mm)
h = 51721800
>>
>> v1=mod_exp(g,h,p)
v1 = 57746599
>> a_r=mod_exp(a,r,p)
a_r = 233505079
>> r_s=mod_exp(r,s,p)
r_s = 207550501
>> v2=mod(a_r*r_s,p)
v2 = 16540280
```

```
>> m='Hello Bob'
m = Hello Bob
>> h=hd28(m)
h = 198770750
r = 232941370
s = 112441390
```

```
>> v1=mod_exp(g,h,p)
v1 = 16540280
>> a_r=mod_exp(a,r,p)
a_r = 233505079
>> r_s=mod_exp(r,s,p)
r_s = 207550501
>> v2=mod(a_r*r_s,p)
v2 = 16540280
```

ElGamal Encryption

$B: \quad t \leftarrow randi(\mathcal{I}_p^*)$

$$\left.\begin{array}{l} E = m \cdot a^t \bmod p \\ D = g^t \bmod p \end{array}\right\} c=(E,D) \longrightarrow$$

$$(-x) \bmod (p-1) = (0-x) \bmod (p-1) =$$
$$= (p-1-x) \bmod (p-1)$$

$c=(E,D)$

1.***Message encryption by* Bob**

```
>> m=111222
m = 111222
>> t=int64(randi(p-1))
t = 3638073
>> a_t=mod_exp(a,t,p)
a_t = 68855447
>> E=mod(m*a_t,p)
E = 57869183
>> D=mod_exp(g,t,p)
```

$c = (E, D)$

>> E=mod(m*a_t,p)
E = 57869183
>> D=mod_exp(g,t,p)
D = 67024666

$\mathcal{A}$: is able to decrypt $c = (E, D)$ using her $PrK_A = x$.

1. $D^{-x \bmod (p-1)} \bmod p$
2. $E \cdot D^{-x} \bmod p = m$

1.***Message decryption by*** **Alice**

x = 27493765
>> mx=mod(-x,p-1)
mx = 240941253
>> mod(x+mx,p-1)
ans = 0
>> D_mx=mod_exp(D,mx,p)
D_mx = 231840357
>> mb=mod(E*D_mx,p)
mb = 111222

$\mathcal{A}$: M - message to be encrypted

$|M| = 1\ GB$

$k \leftarrow randi(\mathbb{Z}_p^*)$

$Enc(b, k) = c = (E, D)$

$AES(k, M, 'e') = C$

$\xrightarrow{c,\ C}$

$\mathcal{B}$:

$Dec(y, c) = k$

$AES(k, C, 'd') = M$

$\mathcal{I}_o$: forging M to M'

$k' \leftarrow randi(\mathbb{Z}_p^*)$

$Enc(b, k') = c' = (E', D')$

$AES(k', M', 'e') = C'$

$\xrightarrow{c',\ C'}$

$\mathcal{B}$: obtains M' by decryption C'.

Avoidance of MiM Attack.

$\mathcal{A}$: $Sign(x, c) = \sigma_c$

$h = H(M)$

$Sign(x, h) = \sigma_C$

$\xrightarrow[\sigma_c,\ \sigma_C]{c,\ C}$

$\mathcal{B}$: verifies signatures σ_c, σ_C on c, C and if verification passes then decrypts c and obtains k

decrypts C and obtains M.