/03/22

Digital signatures

- RSA
- ElGamal
- ECC
- pairing based cryptography

Keypair (SK, PK)

$SK \downarrow$ secret

$PK \rightarrow$ public

$\boxed{\begin{matrix} APK \\ BPK \end{matrix}}$

(ASK, APK) $\boxed{A} \longrightarrow \boxed{B}$ (BSK, BPK)

msg-m.

$\boxed{E}$

$c \leftarrow E[BPK(m)]$

$m \leftarrow D[BSK(c)]$

In this scenario what is the need of digital signature?

An E can replace B's public key with its public

signer – signs the document using SK & PK

verifier – verifies the ~~document~~ signature

Signer :- $h \leftarrow H(m)$

H – hash function

m – message

h – hash message

H is a one way function

The hash message is unique i.e $H(m) \neq H(m')$; $m \neq m'$

size of h is fixed.

$\boxed{\sigma \leftarrow h^d \bmod n}$ d – SK of 'B' verifier:-

→ verifies the document
→ has (m, σ, PK of sender)

1) $h \leftarrow H(m)$
2) $h' \leftarrow \sigma^{PK} \bmod n$.
3) if $h = h'$; valid
   else invalid

$h' \leftarrow h^{SK \cdot PK} \bmod n$

integer factorisation.

RSA is secure because of we know 'n' but we don't know p, q

Elgamal is secure because of discrete logarithm problem.

$y = g^x \bmod p$.

unknown - x
x is private key

$x \Rightarrow Sk$
$y \Rightarrow Pk$

Digital signature using Elgamal:-

Guide to ECC - chapter 1 - RSA, Elgamal, ECC.

Gen ← (x, y)
x → SK, y → PK

$y = g^x \bmod P$ → PK

sign(x, m)

1) $K \in_R [1, q-1]$
2) $T = g^k \bmod P$
3) $h \leftarrow H(m)$
4) $r = T \bmod q$ if $r = 0$ goto step ①
5) $s = k^{-1}(h + xr) \bmod q$
   if $s = 0$ then goto step ①

(r, s) - signature

public domain parameter

(p, q, g)

$\underbrace{p, q}$ → bit length set by the customer.

Say, p length → l.
q length → j
l, j are bit length such as 1024 etc...

⇒ q divides p-1 where g is generator w.r.t q.
i.e, $q \mid p-1$ and $g^q \equiv 1 \bmod P$.

what is the relation between r and s ?

both the values are calculated using mod q hence the value of r and s have boundry of $[0, q-1]$

$r, s \in [0, q-1]$.

Verify the signature :-

1. if not, $0 < r < s < q-1$ not valid.
2. $h = H(m)$
3. $w = s^{-1} \bmod q$
4. $u_1 = hw \bmod q$ and $u_2 = rw \bmod q$
5. $T = g^{u_1} \cdot y^{u_2} \bmod p$ $y \rightarrow$ is public key of signee.
6. $r' = T \bmod q$
7. if $r = r'$ valid,

   else not valid

proof of correctness :-

$s = k^{-1}(h + xr)$ $(ab)^{-1} = b^{-1}a^{-1}$ —— (1)

$(k^{-1})^{-1} = k$. —— (2)

$w = s^{-1} \bmod q$ $y = g^{x} \bmod p$ —— (3)

$= (k^{-1}(h+xr))^{-1} \bmod q$

$= (h+xr)^{-1} k \bmod q$ —— using (1) and (2)

$u_1 = h(h+xr)^{-1} \times k \bmod q$

$u_1 = h(h+xr)^{-1} k \bmod q$

$u_2 = r(h+xr)^{-1} k \bmod q$.

$T = g^{h(h+xr)^{-1} k \bmod q} \cdot y^{r(h+xr)^{-1} k \bmod q} \bmod p.$

$= \left(g^{h(h+xr)^{-1}k} \cdot y^{r(h+xr)^{-1}k}\right) \bmod p$

$= \left(g^{h(h+xr)^{-1}k} \cdot g^{rx(h+xr)^{-1}k}\right) \bmod p$ —— using (3)

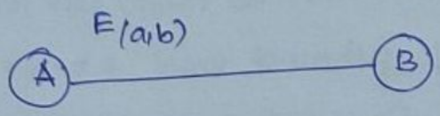
$$= \left(g^{h(h+xr)^{-1}k + rx(h+xr)^{-1}k}\right) \bmod p$$

$$= \left(g^{k(h+xr)^{-1}(h+rx)}\right) \bmod p$$

$r = g^k \bmod p \longrightarrow$ same as step 2 in digital signature using Elgamal.

$\cancel{r' = r \bmod q =} \; g^k \cancel{\bmod p \bmod q}$

24|03|2022 Digitial Signature using Ecc



$y^2 = x^3 + ax + b \in F_q$ where q is prime.

$E(F_q) = \{ (x,y) \in F_q \times F_q \} \cup \underline{\{0\}}$
$\downarrow$
point at infinity.

$\# E(F_q) = \underset{\sim}{<P>} = n \;;\; nP = 0$

The braces means the generator.

<u>for a:</u> $a \in [1, n-1] \rightarrow SK$

$P_a = aP \rightarrow PK$

<u>for b:-</u> $b \in [1, n-1] \rightarrow SK$

$P_b = bP \rightarrow P_K$

<u>Signature Generation</u> :-

1) choose $k \in_R [1, n-1]$ → point/scalar multiplication
2) $R = kP$ } P is a point that lies on curve $P \in E(F_q)$
3) $r = x(R)$ → Taking only x-coordinate for signature
4) $s = k^{-1}(H(m) + dr)$ → d is the secret key of signer

Signature $-(r,s)$

$n \rightarrow$ no of points generated by the curve over field F_q.

<u>Verify</u> :-

1) $w = s^{-1} \bmod n$
2) $u = H(m) w \bmod n$
3) $v = rw \bmod n$
4) $R = uP + vQ \implies$ Q is public key of signer $Q = dP$
5) $r = x(R)$ valid, else not valid

collection of point satisfies additive group not multiplicative group.

correctness proof of verification :-

$$w = s^{-1} \bmod n$$

$$= (k^{-1}(H(m)+dr))^{-1} \bmod n$$

$$= (H(m)+dr)^{-1}\, k \bmod n.$$

$$u = H(m)\,(H(m)+dr)^{-1} k \bmod n$$

$$v = r(\cancel{k}\; H(m)+dr)^{-1} k \bmod n$$

$$R = H(m)(H(m)+dr)^{-1} kP + r(H(m)+dr)^{-1} kQ$$

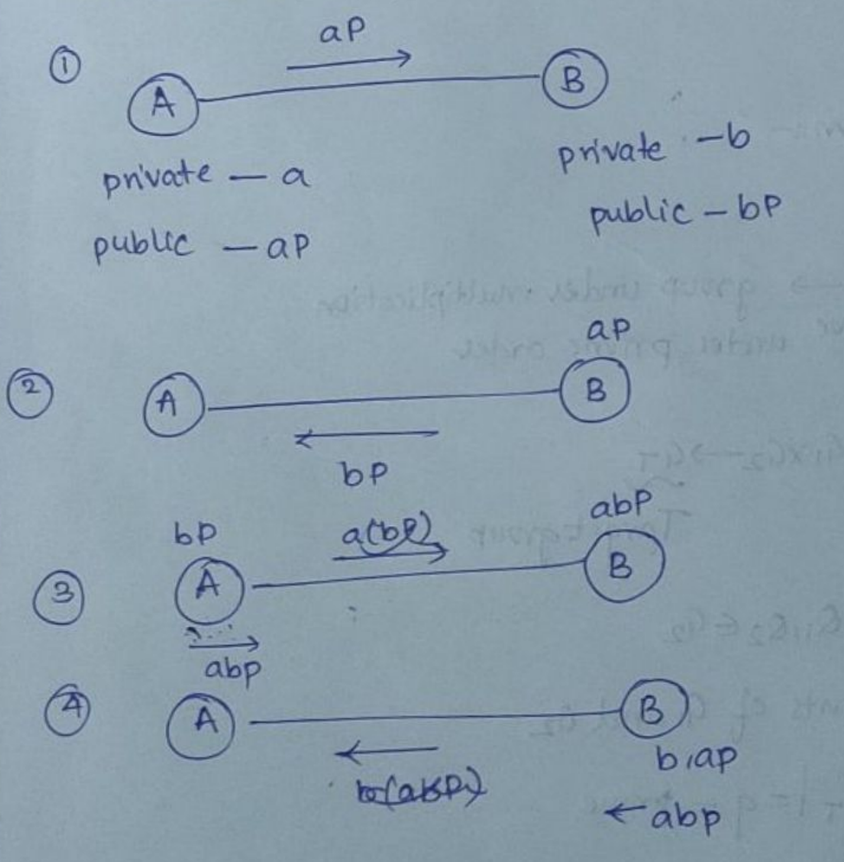
$$= H(m)(H(m)+dr)^{-1} kP + r(H(m)+dr)^{-1} kdP$$

$$= kP\,(H(m)+dr)(H(m)+dr)^{-1}$$

$$= kP.$$

$R = kP \Rightarrow$ step2 in signature generation Hence proved ☺

Defman Algorithm :



Between 3 persons:-

Bilineal Defmann Algorithm:-

$\{\hat{e}, G_1, G_2, G_T\}$

G_1, G_2 → groupunder additive

G_T → group under multiplication under prime order

$\hat{e}$ → bilinear mapping $\hat{e} : G_1 \times G_2 \rightarrow \underbrace{G_T}_{\text{Target group}}$

$P_1, P_2 \in G_1$ ~~&~~ $Q_1, Q_2 \in G_2$

$P_1, Q_1 \in$ elements of G_1 and G_2

$|G_1| = |G_2| = |G_T| = q = \text{prime}$

① $\hat{e}(P_1+P_2, Q_1) = \hat{e}(P_1, Q_1)\cdot\hat{e}(P_2, Q_1)$

② $\hat{e}(P, Q_1+Q_2) = \hat{e}(P, Q_1)\cdot\hat{e}(P, Q_2)$

③ $\hat{e}(0, Q) = \hat{e}(P, 0) = 1$ $\Longrightarrow$ 0 is point of infinity

④ $\hat{e}(aP, bQ) = \hat{e}(P, Q)^{ab} = \hat{e}(bP, aQ) = \hat{e}(P, abQ) = \hat{e}(abP, Q)$ $a, b \in \mathbb{Z}$

⑤ $\hat{e}(-P, Q) = \hat{e}(P, Q)^{-1} = \hat{e}(P, -Q)$

⑥ $\hat{e}(P, Q) \neq 1$, $P = Q \neq 0$

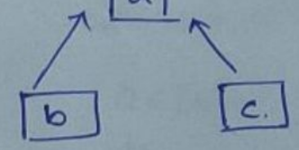
proving ③ based on assuming ① is true.

$$\hat{e}(P, Q) = \hat{e}(P+0, Q)$$
$$= \hat{e}(P, Q)\cdot\hat{e}(0, Q)$$
$$\hat{e}(0, Q) = 1$$

proving ⑤ based on 1

$$\hat{e}(P+(-P), Q) = 1$$
$$\hat{e}(P, Q)\cdot\hat{e}(-P, Q) = 1$$
$$\hat{e}(P, Q) = \hat{e}(-P, Q)^{-1}$$

→ a, bP, cP $\hat{e}(bP, cP)^a = \hat{e}(P, P)^{abc}$



→ $\{\hat{e}, G_1, G_2, G_T\}$

$$\hat{e}: G_1 \times G_2 \rightarrow G_T$$

① $T_1 = G_1 = G_2$

② $T_2 = G_1 \neq G_2$ $\phi . G_2 \rightarrow G_1$

③ T_3 $G_1 \neq G_2$ not homomorphism $\phi(x+y) = \phi(x)\cdot\phi(y)$

$\Rightarrow G_1 = G_2 = Z_5 = \{0, 1, 2, 3, 4\}$

Z_5 is additive group.

G_T (target group) =

$Z_{11}^* = \{0, 1, 2 \ldots 10\}$

$= \{1, 3, 4, 5, 9\}$ is subgroup

as it satisfies:

1. additive identity
2. closure property under $*$
3. multiplicative inverse for all elements.

$\Longrightarrow x \in G_1 \quad y \in G_2$

$\hat{e}(x, y) = 3^{xy}$ on Z_{11}^*

sol:-

$\hat{e}(0, y) = 3^0 = 1$

$\hat{e}(x, 0) = 3^0 = 1$

$\hat{e}(1, 1) = 3$

$\hat{e}(1, 3) = 27 \bmod 11 = 5$

$\hat{e}(1, 2) = 9$

$\hat{e}(1, 4) = 81 \bmod 11 = 4$

$\hat{e}(2, 1) = \hat{e}(1, 1) \cdot \hat{e}(1, 1) = 9.$

$\cancel{\hat{e}(2,2) = \hat{e}(2,0) \cdot \hat{e}(2,2) =}$

$\hat{e}(2, 2) = 4$

$\hat{e}(2, 3) = 3$

$\hat{e}(2, 4) = 5$

$\cancel{\hat{e}(2, 5) \neq 1}$

$\cancel{\hat{e}(2, 6) = 9}$

$\cancel{\hat{e}(2, 7) = 4}$

$\cancel{\hat{e}(2, 8) = 3}$

$\hat{e}(2, 9) = 5$

$\hat{e}(3,1)=5$

$\hat{e}(3,2)=3$

$\hat{e}(3,3)=4$

$\hat{e}(3,4)=9$

$\hat{e}(4,1)=4.$

$\hat{e}(4,2)=5$

$\hat{e}(4,3)=9$

$\hat{e}(4,4)=3$

$\hat{e}(-x,y)=\hat{e}(x,y)^{-1}=\hat{e}(x,-y)$

$\hat{e}(-2,3)=\hat{e}(2,3)^{-1}=\hat{e}(2,-3)$

$\hat{e}(3,3)=3^{-1}(11)=4$

$\hat{e}(ax+by)=\hat{e}(x,y)^{ab}$

$=\hat{e}(x,aby)$

$=\hat{e}(bx,ay)$

$=\hat{e}(bx,ay)$

~~BLS~~ digital signature using Bilinear pairing Mapping :-

$G_1=G_2=\langle P\rangle$, $P\in E(F_p)$

$a\in[1,n-1]$ $\downarrow$ secret key of signer

$aP=A$ $\downarrow$ public key

$M=H(m)$ $M\in E(F_p)$

$S=aM$

parameters that we send to verifier are (P,A,S,M)

P $\rightarrow$ base point

A $\rightarrow$ public key of signer

S $\rightarrow$ signature of signer

M $\rightarrow$ hash message.

verify :- $\hat{e}(P,S) = \hat{e}(A,M)$

$$\hat{e}(P,S) = \hat{e}(P,aM)$$
$$= \hat{e}(aP,M)$$
$$= \hat{e}(A,M)$$

if $\hat{e}(P,S) = \hat{e}(A,M)$ then valid, else invalid.

Batch Signature Verification :-

$$1 \leq i \leq t$$

$\hat{e}(A_i, M_i) \leftarrow i^{th}$ party

- If any one of the signature's is wrong we reject all other signatures ~~also~~ in case of batch signatures.

$$\hat{e}(P,S) = \prod_{i=1}^{t} \hat{e}(A_i, M_i) \quad 1 \leq i \leq t$$

guide: An Introduction to pairing based cryptography