Pairings-based Cryptography

Dario Catalano
Università di Catania
Italy

Cryptography Summer School - Bucharest
July 2014

Identity Based Encryption

- Goal: Realize a cryptosystem that can use *any* string as possible public key.
  - Alice wants to encrypt M using Bob's identity bob@uni-bob.edu as pk.
- This simplifies the management of public keys (sometimes).
- To decrypt, needs to be able to get a (secret) key for bob@uni-bob.edu.

Why is this so cool?

- Alice can send encrypted msgs to Bob even if he does not create his public key.
- Easy implements “expiring” or temporary keys.
  - bob@uni-bob.edu||this year
- Key revocation becomes easier to deal with (sometimes).
- Key Escrow comes for free.
- Actually IBEs have many more applications!

The IBE context

- Users gets the secret keys from a (trusted) authority Auth
- Auth knows a master secret key msk that allows to compute local secret keys.
- Without msk it should be impossible to do the same.
- This new setting complicates things a bit...

Definition (ind-cpa security)

- AE=(KeyGen, Enc, Dec)

$Esp_{AE}^{ind\text{-}cpa\text{-}1}(A)$	$Esp_{AE}^{ind\text{-}cpa\text{-}0}(A)$
$(pk,sk) \leftarrow_R KeyGen$	$(pk,sk) \leftarrow_R KeyGen$
$b \leftarrow A^{Enc_{pk}(LR(.,.,1))}$	$b \leftarrow A^{Enc_{pk}(LR(.,.,0))}$
Return b	Return b

$$Adv^{ind\text{-}cpa}(A)=Pr[Esp_{AE}^{ind\text{-}cpa\text{-}1}(A)=1] - Pr[Esp_{AE}^{ind\text{-}cpa\text{-}0}(A)=1]$$

The IBE context – II

- Adversaries might *already* have a certain number of secret keys.
- These might be used to derive sk corresponding to *new* identities.
- A good IBE should remain secure also with respect to this kind of adversary.

More in detail

- We formalize this by providing A with a new oracle.
- Extract(ID) → SK_{ID}
- A can ask a limited number of extract queries.
- Next A asks to be challenged on some ID* of its own choice
  - A cannot ask extract queries on such an ID*
- The rest is like in the usual (ind-cpa) security.

Definition (ind-id-cpa-security)

– IBE=(Setup,KeyDer,Enc,Dec)

$Esp^{ind\text{-}id\text{-}cpa\text{-}1}$ (A)	$Esp^{ind\text{-}id\text{-}cpa\text{-}0}$ (A)
(pk,msk) $\leftarrow_R$ Setup	(pk,msk) $\leftarrow_R$ Setup
b $\leftarrow A^{Enc_{pk}(id^*,LR(.,.,1)),KeyDer_{msk}(.)}$	b $\leftarrow A^{Enc_{pk}(id^*,LR(.,.,0)),KeyDer_{msk}(.)}$
If A cheats Return 0	If A cheats Return 0
else return b	else return b

$Adv^{ind\text{-}id\text{-}cpa}(A)=Pr[Esp^{ind\text{-}id\text{-}cpa\text{-}1}(A)=1] - Pr[Esp^{ind\text{-}id\text{-}cpa\text{-}0}(A)=1]$

A cheats if asks $KeyDer_{msk}(.)$ on id*.

Boneh-Franklin

- Realizing IBEs has been a long standing open problem for many years.
- BF proposed a solution that uses parings.

Bilinear Maps (as considered in Crypto)

- G_1, G_2, G_T finite groups such that

1. G_1 and G_2 additive groups, G_T multiplicative one
2. G_1, G_2 groups of (prime) order q.
   - g generator of G_1; g' generator of G_2
3. $\rho : G_2 \rightarrow G_1$ isomorphism such that $\rho(g') = g$
4. $e : G_1 \times G_2 \rightarrow G_T$ has the following properties
   1. Bilinear: $\forall U \in G_1, \forall V \in G_2, \forall a,b \in Z, e(U^a, V^b) = e(U,V)^{ab}$
   2. Non degeneracy: $e(g,g') \neq 1_{G_T}$
   3. Efficiency: Group ops, ρ and e are efficiently computable.

Simplified notation

- We will assume that $G_1 = G_2$
  - Usually referred as the symmetric setting

$$e : G \times G \rightarrow G_T$$

- Will use multiplicative notation everywhere

Boneh Franklin - Details

q prime, G, G_T groups of order q, g gen. of G

$e : G \times G \rightarrow G_T$ admissible bilinear map

Messages are elements in G_T.

Setup: Auth randomly chooses a in Z_q. Pk: $h=g^a$

Key Extraction: Secret key for user id is set as

$Q_{id}=H_1(id)$; // H_1: $\{0,1\}^* \rightarrow G$, *Special Hash function*

$d_{id}=Q_{id}^a$

Boneh-Franklin – Details (cont.)

$\mathbf{Enc}(m, \mathtt{id}, Pk)$ // $M = G_T$

$r \leftarrow_R \{1, \ldots q\}$; $Q_{\mathtt{id}} \leftarrow H_1(\mathtt{id})$

$k \leftarrow e(Q_{\mathtt{id}}, h)^r$;

$c_1 \leftarrow g^r$; $c_2 \leftarrow m \cdot k$

Return (c_1, c_2)

$\mathbf{Dec}((c_1, c_2), d_{\mathtt{id}})$

$k \leftarrow e(d_{\mathtt{id}}, c_1)$; $m \leftarrow c_2 \cdot k^{-1}$

Return m

Why does this work? (Correctness)

$$k = e(d_{id}, c_1) = e(Q_{id}, g^r) =$$

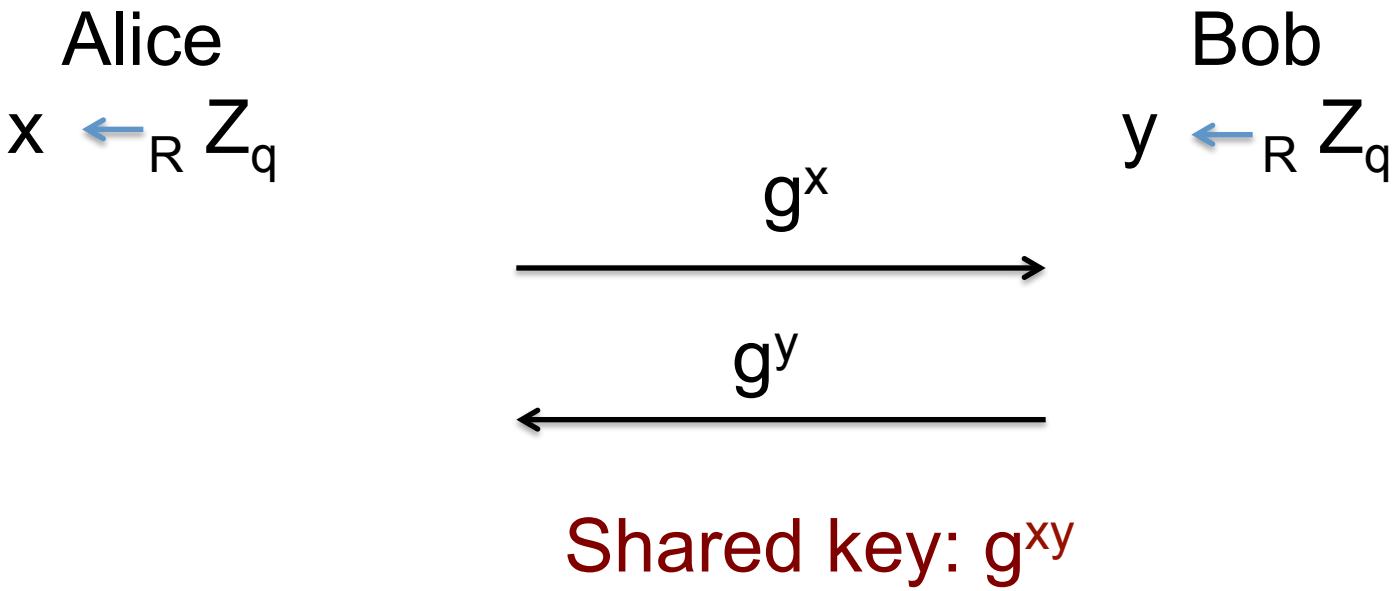
by bilinearity

$$= e(Q_{id}, g^a)^r = e(Q_{id}, h)^r$$

Why is this a good encryption scheme? (Security)

- We'll prove security under the *Bilinear-Diffie Hellman Assumption*.
  - Variant of the standard Diffie-Hellman assumption
- The security proof is in the *random oracle model*.
- The scheme can be generalized to achieve security against stronger advs

Interlude: Diffie-Hellman Problem



Problem: How can we be sure that the key is truly random?

Decisional Diffie-Hellman Problem

G (cyclic) group of order *q*, *g* generator of *G*

$Esp_{G,g}^{ddh\text{-}1}(A)$	$Esp_{G,g}^{ddh\text{-}0}(A)$
$x,y \leftarrow_R Z_q$; $z \leftarrow xy \bmod q$	$x,y,z \leftarrow_R Z_q$;
$X \leftarrow g^x$; $Y \leftarrow g^y$; $Z \leftarrow g^z$;	$X \leftarrow g^x$; $Y \leftarrow g^y$; $Z \leftarrow g^z$;
$d \leftarrow_R A(X,Y,Z)$	$d \leftarrow_R A(X,Y,Z)$
Return d	Return d

$$Adv^{ddh}_{G,g}(A)=Pr[Esp^{ddh\text{-}1}_{G,g}(A)=1]-Pr[Esp^{ddh\text{-}0}_{G,g}(A)=1]$$

Exercise 1

- Show that if you have a bilinear map of the form $e : G \times G \rightarrow G_T$ decisional DH is easy in G.
- Given (g, g^x, g^y, g^z) deciding if $z=xy \bmod q$ is easy

Bilinear Diffie-Hellman Problem

- G, G_T finite groups of order q $e : G \times G \rightarrow G_T$
- Given g^a, g^b, g^c compute $e(g,g)^{abc}$.

Decisional Bilinear Diffie-Hellman Problem

G, G_T groups of order q, g generator of G, $e : G \times G \rightarrow G_T$ bilinear map.

$Esp_{G,G_T,e,g}^{bddh\text{-}1}(A)$	$Esp_{G,G_T,e,g}^{bddh\text{-}0}(A)$
$a,b,c \leftarrow_R Z_q$; $d=abc \bmod q$	$a,b,c,d \leftarrow_R Z_q$;
$T= e(g,g)^d$;	$T=e(g,g)^d$;
$\beta \leftarrow \mathcal{A}(g^a, g^b, g^c, T)$	$\beta \leftarrow \mathcal{A}(g^a, g^b, g^c, T)$
Return β	Return β

$$Adv^{bddh}(\mathcal{A}) = Pr[Esp_{G,G_T,e,g}^{bddh\text{-}1}(\mathcal{A}) = 1] - Pr[Esp_{G,G_T,e,g}^{bddh\text{-}0}(\mathcal{A}) = 1]$$

Exercise 2: Tripartite Diffie-Hellman

- Devise a (non interactive) protocol that allows three users to exchange a common secret key.

Security of BF

- Assume there is $\mathcal{A}$ that breaks ind-id-cpa security of BF with advantage $\varepsilon(k)$, making at most q_d derivation queries
- We can build an (equally efficient) $\mathcal{B}$ that breaks DBDH with advantage

$$Adv(B) \geq \frac{\varepsilon(k)}{2e \cdot (1 + q_d)}$$

Proof

- $\mathcal{B}$ receives the DBDH parameters

$$(g, A = g^a, B = g^b, C = g^c, X) \quad (g \in G)$$

- where $X = \begin{cases} e(g,g) \\ \in_R G_T \end{cases}$
- $\mathcal{B}$ runs $\mathcal{A}$ to (try to) break DBDH.

Setup

- Public Parameters:

$$(q, G, G_T, e, g, h = A, H_1)$$

- H_1 random oracle chosen (and controlled) by $\mathcal{B}$.
- Note: no secret key is explicitly known by $\mathcal{B}$.

Random oracle queries

- $\mathcal{B}$ maintains a table H containing tuples of the form $\langle id_i, \nu_i, \beta_i, Q_i \rangle$
- When receiving the query id_i, $\mathcal{B}$ proceeds as follows
- if id_i in H output Q_i
- else flip a bit v_i (s.t. $\Pr[v_i=0]=\delta$), pick $\beta_i \in_R Z_q$ and set

$$X = \begin{cases} g^{\beta_i} & if \quad v_i = 0 \\ B^{\beta_i} & if \quad v_i = 1 \end{cases}$$

Key Derivation queries

- if $\mathcal{A}$ asks for the sk corresponding to id_i, $\mathcal{B}$ answers as follows

1. Run the (just seen) algorithm to answer oracle queries. Let $\langle id_i, \nu_i, \beta_i, Q_i \rangle$ be the resulting tuple.
2. if $\nu_i = 1$ **abort**
3. Otherwise, $Q_i = g^{\beta_i}$, we can compute the secret key $d_i = h^{\beta_i}$

Challenge

- $\mathcal{A}$ outputs m_0, m_1, id^*
- $\mathcal{B}$ "runs" H_1 on input id^*.
- if $v_i^*=0$ **abort**
- Otherwise, $Q^* = B^{\beta^*}$, $\mathcal{B}$ computes the challenge ciphertext as follows

$$C_1 = C^{\beta^{*-1}}$$

$$C_2 = T \cdot m_b$$

Guess

- $\mathcal{A}$ outputs a bit (guess) γ'
- if γ'=b $\mathcal{B}$ outputs 1, 0 otherwise.
- We show that

1. if $\mathcal{B}$ does not abort it solves DBDH with advantage $\mu(k) = \varepsilon(k)/2$
2. It does not abort with probability $\frac{1}{e(1+q_d)}$

- The adv of $\mathcal{B}$ is thus $\frac{\varepsilon(k)}{2e(1+q_d)}$

Point 1

$\Pr[Esp^{dbdh\text{-}1}(\mathcal{B})=1]=\Pr[\mathcal{A}$ guesses b correctly$]=$

$\Pr[\mathcal{A}$ outputs 1 | b=1$]\Pr[b=1]+\Pr[\mathcal{A}$ outputs 0 | b=0$]\Pr[b=0]$

$1/2\ Adv(\mathcal{A})+1/2=\varepsilon(k)/2+1/2$

$\Pr[Esp^{dbdh\text{-}0}(\mathcal{B})=1]=\Pr[\mathcal{A}$ guesses b correctly$]=$

$\Pr[\mathcal{A}$ outputs 1 | b=1$]\Pr[b=1]+\Pr[\mathcal{A}$ outputs 0 | b=0$]\Pr[b=0]$

$1/2\ Adv(\mathcal{A})+1/2=1/2$

$$\mu(k)=\varepsilon(k)/2$$

Point 2

- For each derivation query $\mathcal{B}$ does not abort with probability δ.
- For q_d queries this becomes δ^{q_d}
- For the challenge phase this probability is 1-δ
- Overall probability of not aborting

$$(1-\delta)\delta^{q_d}$$

$$\delta_{opt} = 1 - \frac{1}{q_d + 1} \Rightarrow\approx \frac{1}{e(q_d + 1)}$$