Public-Key Cryptography

Lecture 10

DDH Assumption

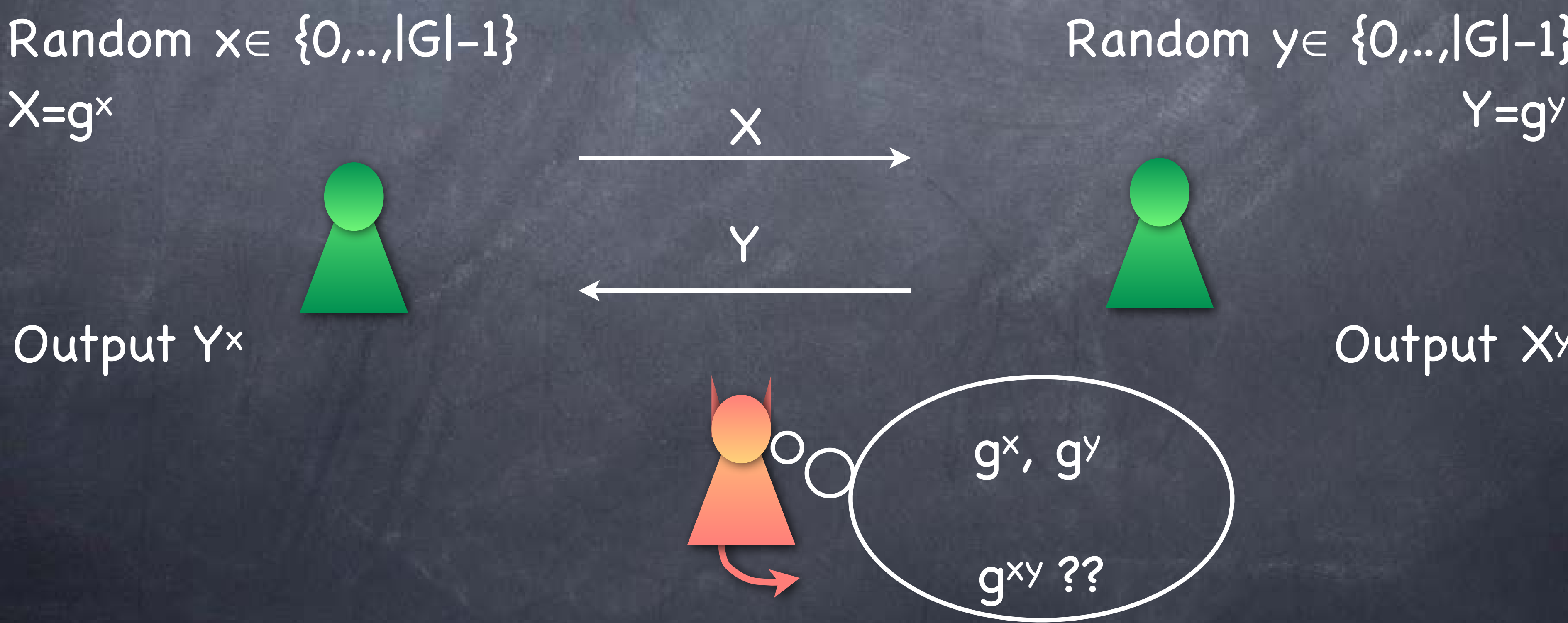
El Gamal Encryption

Public-Key Encryption from Trapdoor OWP

RECALL

Diffie-Hellman Key-exchange

- "Secure" if $(g^x, g^y, g^{xy}) \approx (g^x, g^y, g^r)$



RECALL

Discrete Log Assumption

Repeated squaring

- Discrete Log (w.r.t g) in a (multiplicative) cyclic group G generated by g: $DL_g(X)$:= unique x such that $X = g^x$ ($x \in \{0,1,...,|G|-1\}$)
- In a (computationally efficient) group, given integer x and the standard representation of a group element g, can efficiently find the standard representation of $X=g^x$ (How?)
  - But given X and g, may not be easy to find x (depending on G)
  - DLA: Every PPT Adv has negligible success probability in the DL Expt: (G,g)←GroupGen; X←G; Adv(G,g,X)→z; $g^z=X$?

OWF collection: Raise(x;G,g) = (g^x;G,g)

- If DLA broken, then Diffie-Hellman key-exchange broken
  - Eve gets x, y from g^x, g^y (sometimes) and can compute g^{xy} herself
    - A "key-recovery" attack
  - Note: could potentially break pseudorandomness without breaking DLA too

RECALL

Decisional Diffie-Hellman (DDH) Assumption

- $\{(g^x, g^y, g^{xy})\}_{(G,g)\leftarrow\mathbf{GroupGen};\ x,y\leftarrow[|G|]} \approx \{(g^x, g^y, g^r)\}_{(G,g)\leftarrow\mathbf{GroupGen};\ x,y,r\leftarrow[|G|]}$
- At least as strong as Discrete Log Assumption (DLA)
  - DLA: Raise(x; G,g) = (g^x; G,g) is a OWF collection
  - If DDH assumption holds, then DLA holds [Why?]
- But possible that DLA holds and DDH assumption doesn't
  - e.g.: DLA is widely assumed to hold in $\mathbb{Z}_p^*$ (p prime), but DDH assumption doesn't hold there! (coming up)
- Today: a candidate group for DDH

A Candidate DDH Group

- Consider $\mathbb{QR}_P^*$: subgroup of Quadratic Residues ("even power" elements) of $\mathbb{Z}_P^*$

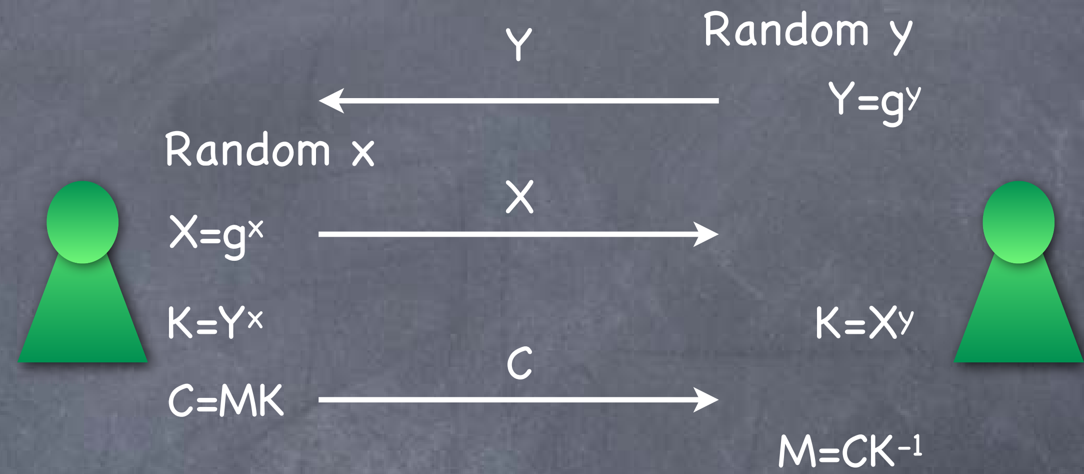
- Easy to check if an element is a QR or not: check if raising to |G|/2 gives 1 (identity element)
- DDH does not hold in $\mathbb{Z}_P^*$: g^{xy} is a QR w/ prob. 3/4; g^z is QR only w/ prob. 1/2.
- How about in $\mathbb{QR}_P^*$?
  - Could check if cubic residue in $\mathbb{Z}_P^*$!
    - But if (P-1) is not divisible by 3, all elements in $\mathbb{Z}_P^*$ are cubic residues!
  - "Safe" if (P-1)/2 is also prime: P called a safe-prime

(P-1)/2 called a Sophie Germain prime

DDH Candidate:
$\mathbb{QR}_P^*$
where P is a random k-bit safe-prime

El Gamal Encryption

- Based on DH key-exchange
  - Alice, Bob generate a key using DH key-exchange
  - Then use it as a one-time pad
- Bob's "message" in the key-exchange is his PK
- Alice's message in the key-exchange and the ciphertext of the one-time pad together form a single ciphertext



KeyGen: PK=(G,g,Y), SK=(G,g,y)

$\mathrm{Enc}_{(G,g,Y)}(M) = (X=g^x,\ C=MY^x)$

$\mathrm{Dec}_{(G,g,y)}(X,C) = CX^{-y}$

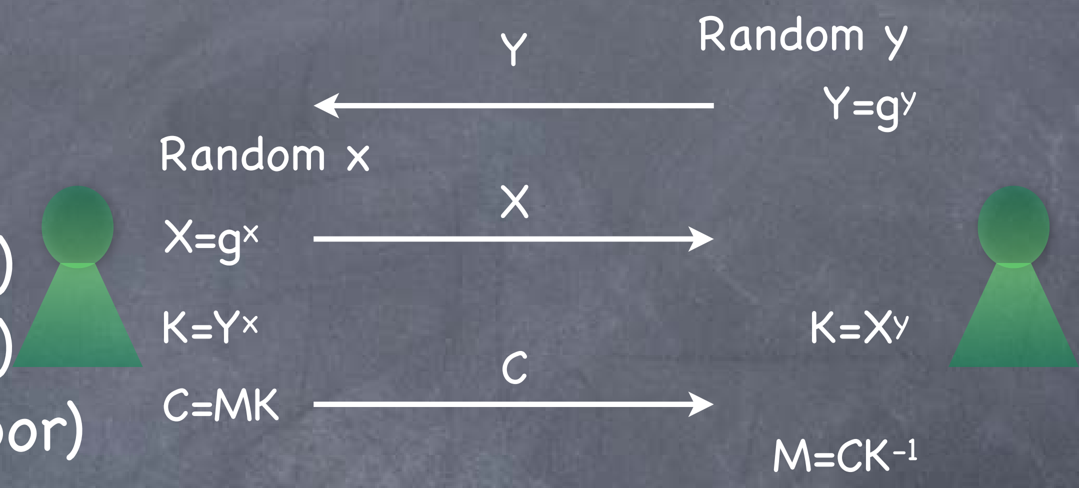
- KeyGen uses GroupGen to get (G,g)
- x, y uniform from $\mathbb{Z}_{|G|}$
- Message encoded into group element, and decoded

Security of El Gamal

- El Gamal IND-CPA secure if DDH holds (for the collection of groups used)
  - Construct a DDH adversary A^* given an IND-CPA adversary A
  - $A^*(G,g; g^x,g^y,g^z)$ (where $(G,g) \leftarrow$ GroupGen, x,y random and z=xy or random) plays the IND-CPA experiment with A:
    - But sets $PK=(G,g,g^y)$ and $Enc(M_b)=(g^x,M_b g^z)$
    - Outputs 1 if experiment outputs 1 (i.e. if b=b')
  - When z=random, A^* outputs 1 with probability = 1/2
  - When z=xy, exactly IND-CPA experiment: A^* outputs 1 with probability = 1/2 + advantage of A.

Abstracting El Gamal

- Trapdoor PRG:
  - KeyGen: a pair (PK,SK)
  - Three functions: $G_{PK}(.)$ (a PRG) and $T_{PK}(.)$ (make trapdoor info) and $R_{SK}(.)$ (opening the trapdoor)
    - $G_{PK}(x)$ is pseudorandom even given $T_{PK}(x)$ and PK
    - $(PK, T_{PK}(x), G_{PK}(x)) \approx (PK, T_{PK}(x), r)$
    - $T_{PK}(x)$ hides $G_{PK}(x)$. SK opens it.
      - $R_{SK}(T_{PK}(x)) = G_{PK}(x)$
- Enough for an IND-CPA secure PKE scheme (e.g., Security of El Gamal)



KeyGen: $PK=(G,g,Y)$, $SK=(G,g,y)$

$Enc_{(G,g,Y)}(M) = (X=g^x, C=MY^x)$

$Dec_{(G,g,y)}(X,C) = CX^{-y}$

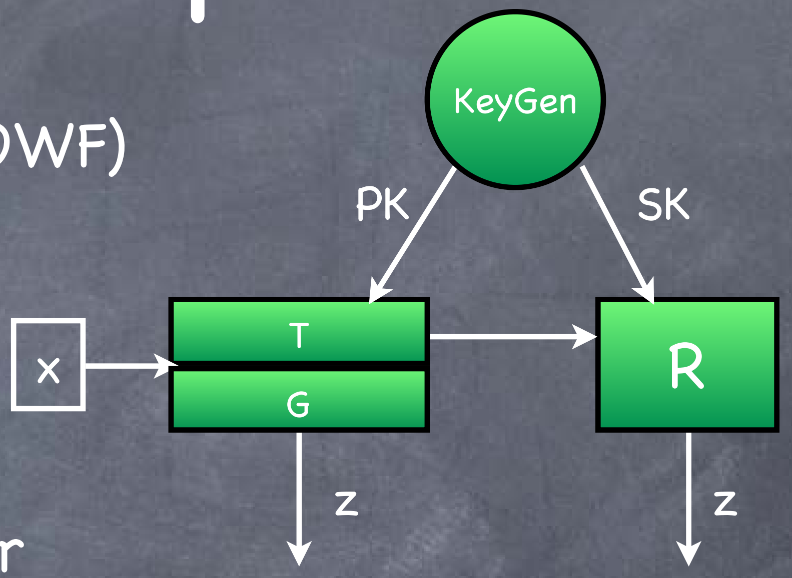
KeyGen: (PK,SK)

$Enc_{PK}(M) = (X=T_{PK}(x), C=M.G_{PK}(x))$

$Dec_{SK}(X,C) = C/R_{SK}(T_{PK}(x))$

Trapdoor PRG from Generic Assumption?

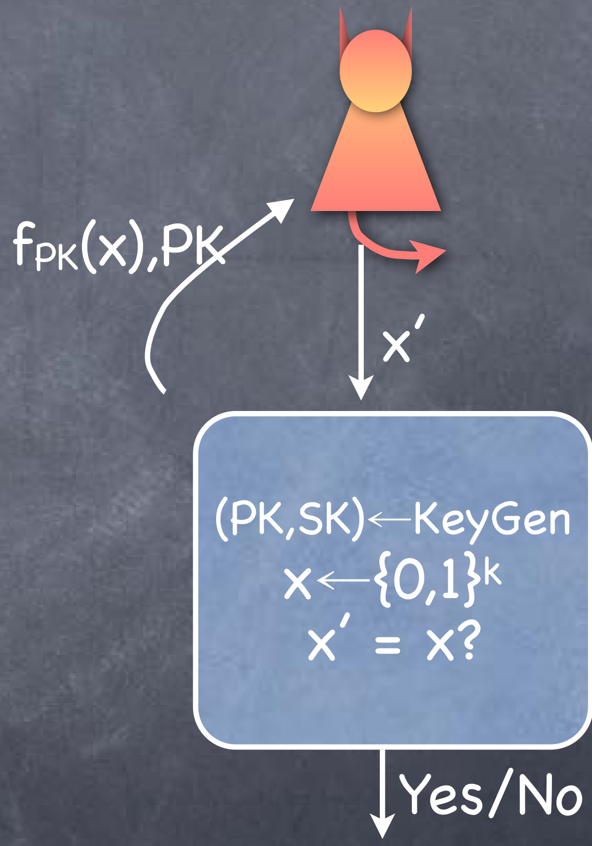
- PRG constructed from OWP (or OWF)
  - Allows us to instantiate the construction with several candidates
- Is there a similar construction for TPRG from OWP?
  - Trapdoor property seems fundamentally different: generic OWP does not suffice
  - Will start with "Trapdoor OWP"



$(PK, T_{PK}(x), G_{PK}(x)) \approx (PK, T_{PK}(x), r)$

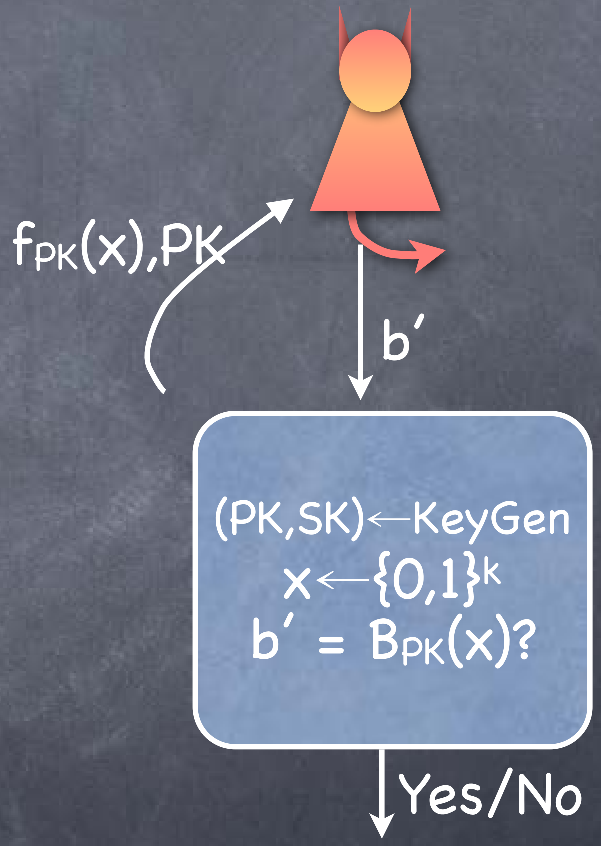
Trapdoor OWP

- (KeyGen,f,f') (all PPT) is a trapdoor one-way permutation if
  - For all (PK,SK) ←KeyGen
    - f_{PK} a permutation
    - f'_{SK} is the inverse of f_{PK}
  - For all PPT adversary, probability of success in the Trapdoor OWP experiment is negligible



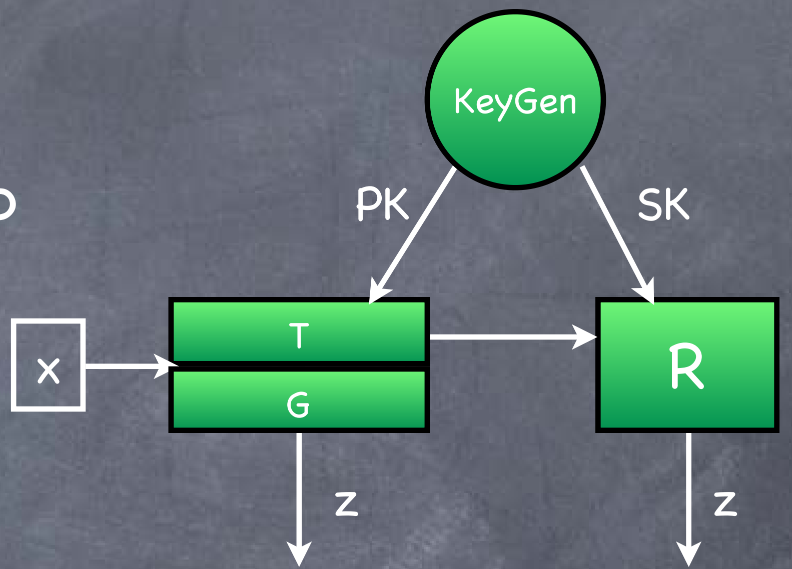
Trapdoor OWP

- (KeyGen,f,f') (all PPT) is a trapdoor one-way permutation if
  - For all (PK,SK) ←KeyGen
    - f_{PK} a permutation
    - f'_{SK} is the inverse of f_{PK}
  - For all PPT adversary, probability of success in the Trapdoor OWP experiment is negligible
  - Hardcore predicate:
    - B_{PK} s.t. $(PK, f_{PK}(x), B_{PK}(x)) \approx (PK, f_{PK}(x), r)$



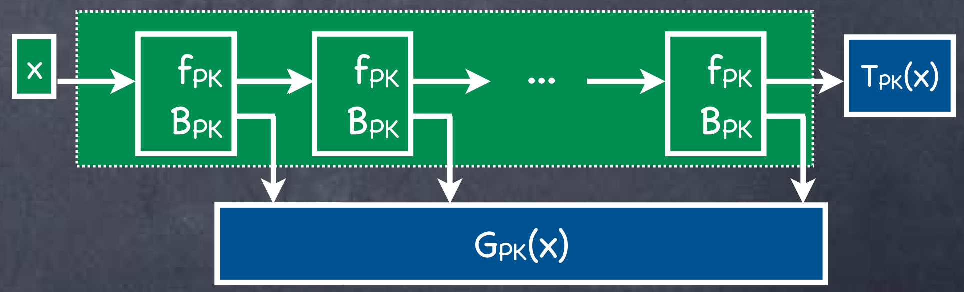
Trapdoor PRG from Trapdoor OWP

- Same construction as PRG from OWP
- One bit Trapdoor PRG
  - KeyGen same as Trapdoor OWP's KeyGen
  - $G_{PK}(x) := B_{PK}(x)$. $T_{PK}(x) := f_{PK}(x)$. $R_{SK}(y) := G_{PK}(f'_{SK}(y))$
    - (SK assumed to contain PK)
- More generally, last permutation output serves as T_{PK}



$(PK, T_{PK}(x), G_{PK}(x)) \approx (PK, T_{PK}(x), r)$

$(PK, f_{PK}(x), B_{PK}(x)) \approx (PK, f_{PK}(x), r)$



Candidate Trapdoor OWPs

- From some (candidate) OWP collections, with index as public-key
- Recall candidate OWF collections
  - Rabin OWF: $f_{Rabin}(x; N) = x^2 \bmod N$, where N = PQ, and P, Q are k-bit primes (and x uniform from {0...N-1})
    - Fact: $f_{Rabin}(.; N)$ is a permutation among quadratic residues, when P, Q are ≡ 3 (mod 4)
    - Fact: Can invert $f_{Rabin}(.; N)$ given factorization of N
  - RSA function: $f_{RSA}(x; N,e) = x^e \bmod N$ where N=PQ, P,Q k-bit primes, e s.t. gcd(e,φ(N)) = 1 (and x uniform from {0...N-1})
    - Fact: $f_{RSA}(.; N,e)$ is a permutation
    - Fact: While picking (N,e), can also pick d s.t. $x^{ed} = x$

Next time