Internetwork Security
Prof. Sourav Mukhopadhyay
Department of Mathematics
Indian Institute of Technology, Kharagpur

Lecture - 37
Digital Signature

We will talk about Digital Signature scheme. So, this is used for authentication purpose. So, this is under public key setup.

(Refer Slide Time: 00:29)

We have 2 party; Alice and Bob and they are communicating over a public channel, now Alice wants to send a message to Bob and Bob wants to ensure that the message is really coming from Alice. So, this authentication has to be preserved. So, now, the digital signature is the public key setup, now Alice is sending a message to Bob. So, Alice has to sign on the message. So, this setup phase has to done by Alice. So, in the setup phase, Alice has to generate this e A, d A. So, Alice public key and Alice private key and then Alice can encrypt this message using the Alice public key I sorry, Alice public key private key. So, this is the signature. So, this is the Alice signature, on this message and Alice is sending to this message to Bob and Bob can just Bob has this Alice public key and Bob can decrypt it and verify this message is coming from Alice, this is the authentication.

Now let us talk about RSA signature scheme. So, in RSA; we have to do the RSA setup. So, that setup; who will do now because Alice is wants to send a message to Bob. So, Alice as to get her own public key private key pair, so, this setup has to be done by Alice then only Alice can sign after generating the public key, private key pair then only Alice can sign on the message using Alice private key. So, this is the setup phase or key generation phase done by the Alice. So, first of all Alice will choose 2 prime p q and compute this is the RSA cryptosystem.

Compute this product n, n equal to p q and then compute the phi n Euler phi function e minus 1, q minus 1 and then Alice choose a key such that gcd of e comma phi n is 1. So, this is required to ensure that this e as a inverse under mod phi m and that inverse we denote by d. So, we find the inverse of e and that will exist because this gcd is one. So, Alice has to find this d And this d can be get by the using the extended Euclidean algorithm such that e d is congruent to 1 mod phi n. So, this is the setup phase. So, once the setup phase is done, once Alice run this setup phase. So, Alice is having this key; key set. So, this key set is basically consists of this n p q e d this e d such that e d is congruent to 1 mod phi n; 1 mod phi n.

A key is basically is basically this topple. So, p is p q, so, oh n p q then e d. So, among this which are the public only n and e are public key of Alice. So, this is e of A public key of Alice, this is n and e and remaining this p q d All are secret key of Alice. So, that is basically d of a, d of a, secret key of Alice, this is p q d so, but here we can change the role because we need this e d is congruent to 1 mod phi m. So, this is the beauty of RSA cryptosystem, this is very symmetric, we can keep one as a public key other as a private key anyway. So, we are using e as a public key and d As a secret key. So, now, this is the key setup is done, now the plaintext phase; plaintext is coming from z n. So, Alice chooses this plaintext m this message, now Alice wants to en encrypt this message, Alice wants to sign on this message. So, how Alice will sign? Alice will do the RSA encryption using the Alice secret key that is the secret to the Alice. So, that Alice only can do; digital signature.

The signature can be done only that person so; that means we have to do something with the secret which is typically secret to that person. So, that is typically the signature and that is the secret key. So, Alice will do the RSA encryption on this message using Alice secret key. So, this is basically m to the power d mod n. So, this is the RSA; RSA

encryption, but using the secret key of Alice. So, this is the signature part. So, Alice is signing on the message. So, this is Alice's signature. So, then Alice is sending this signature, this is now y to Bob. So, Bob is having Alice's public parameter. So, this is n e because this is public, anybody can verify this, anybody can do this verification. So, Oscar also having this e of a, so Oscar also can verify these messages coming from Alice, so authentication can be done by other; Oscar also.

Bob is receiving this now after receiving this. So, Bob is getting y; this signature or the cipher text we can say. So, now, what Bob will do Bob will decrypt it using the public key of Alice, this is the RSA decryption. So, Bob will decrypt this y using the public key of Alice. So, this is the RSA decryption, this is basically y to the power e mod n. So, now, how it will give us m because this is y is basically what? Y is m to the power d now to the power e mod n. So, this is basically m to the power e d mod n, now this is the RSA decryption.

So, we have already seen, this is just we know that e d is congruent to 1 mod phi n so; that means, e d can be written as k phi n plus 1. So, if we write this then this is basically m to the power k phi n plus 1 mod n. So, this can be written as m to the power phi n mod n to the power k into m, this is mod n; mod n, now if we choose the n such that n and m are relatively prime, they are co prime then by a this is the Euler by Euler's theorem this is 1. So, this is basically n.

So, now Alice got the message by Bob got the message and if Alice is sending message also along with this Bob can compare this and Bob can ensure that the message is coming from Alice. So, this is a typical RSA signature scheme. So, now, we will formally define, what do you mean by a signature scheme? So, let us just formally define, what is a signature scheme? So, it is basically a phi topple like encryption scheme it was a phi topple in encryption scheme.

(Refer Slide Time: 10:30)

We have p c cipher text space e space this is the encryption scheme. So, in encryption scheme, we have a plaintext space, cipher text space, key space and we have set of encryption function and we have set of decryption function this phi topple, so, such that for every key from the key space, there must exists an encryption algorithm from this set of all algorithm; encryption algorithms such that and a decryption; corresponding to decryption algorithm d k. So, better we use capital K capital E, E K and B K from this decryption set of all decryption algorithm such that such that D K of E K of m should give us n and this should be true for all m belongs to the plaintext. So, this is the encryption scheme.

So, if an provided this E K D K should be easy functional, easy function means it should be polynomically it should not be hard this E K D K should be feasible function i mean it should not be a hard problem like one way function like that. So, it should be a this E K D K calculating E K D K. So, if you have a message you should be able to encrypt it. So, it could be d A s e A s or it could be any public key. So, encryption algorithm should be should give us the cipher text and on this decryption algorithm should give us the plaintext back.

(Refer Slide Time: 12:53)

This is the encryption scheme on the other hand so; signature scheme is also a phi topple. So, if the signature scheme signature scheme is also phi topple. So, we have plaintext space and instead of cipher text space we have a signature space as I mean the signature all possible signatures. So, this is basically set of all possible signatures; set of all possible signature; signatures and we have key space and we have the signature algorithm or signing algorithm and we have verification algorithm P. So, this is key space set of all possible key and this is the all possible signature algorithm set of all possible signature algorithm and this is similarly set of all possible verification algorithm; verification algorithms.

So, this phi topple such that for a given k key. So, for each k coming from this key space there must exist a signature algorithm from this s and a corresponding verification algorithm from this v such that now signature this is the mapping from where to where the signature of k it is basically a function form plaintext page to this set of all signatures and the verification is basically it is taking the plaintext space and the signature and it is giving us basically a either true or false authenticated or not authenticated.

Now, this verification should be done like this, so following.

(Refer Slide Time: 16:06)

For every x plaintext and for every signature, signature is coming from a y. So, this must be true verification of x comma y is true, if this y is the signature of x, so, if y or sign y x was sign and this is the signature and false if this is not 1. So, this is the formal definition of a signature scheme. So, it is basically a phi topple and for each sorry this is the plaintext phase. So, this is the set of all possible signature and this is the key space this is the set of all possible signature algorithm and this is the set of all the verification algorithm. So, if a message was signed by a key using a key and it is give a signature y then we should able to verify it. So, it should give us a true if it is signature basically otherwise it should give us a false.

Now, let us talk about this what we have discussed RSA signature based on this phi topple. So, what are they for RSA signature scheme, so for RSA signature.

(Refer Slide Time: 18:26)

This is the formal way we are defining the RSA signatures K. So, in RSA signature P is equal to a is equal to. So, this is a is equal to Z p sorry Z n. So, an n is equal to the 2P into Q2 prime and the key space is basically this set n comma P comma Q comma e d such that e d is congruent to 1 mod phi n and we know this is these and these are public parameter and this is the private parameter.

Now if we choose a key n p q e d, so, how we define the signature? So, signature using this key on the message x where x is coming from x is coming from Z p. So, this is basically x to the power a sorry, x to the power d. So, d is the public key sorry, d is the secret key of the sender that is Alice mod n. So, this is the signature now, how 1 can do the verification. So, verification algorithm is. So, if we have given x y. So, it is true if and only if x is congruent to y to the power. So, d e y to the power e mod n and y is basically x y both are coming from Z n. So, this is the RSA signature scheme in the formal way.

Now, we talk about ElGamal signature scheme, so just to quickly talk about ElGamal signature scheme.

(Refer Slide Time: 21:18)

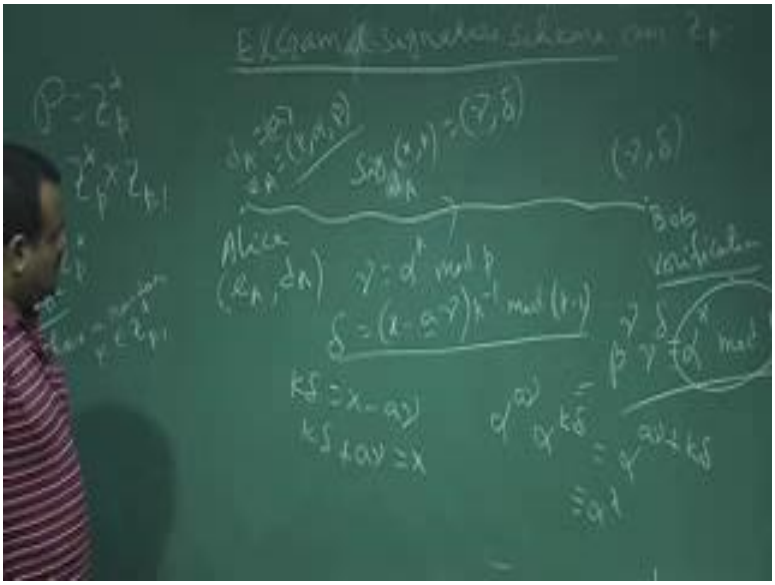
Let us just quickly recap what is the ElGamal cryptosystem. So, for ElGamal cryptosystem, so, we have 2 party Alice and Bob. So, they are communicating over this public channel, now Alice wants to send a message to Bob. So, in ElGamal it is based on the discrete log problem is hard, so for ElGamal, we need to show ElGamal, keep the signature scheme over Z p. So, this is the signature. So, this Alice has to do this setup phase, so Alice and Bob so, Alice has to run this setup phase in order to generate Alice e A d A Alice public key and private key P R.

Alice will choose a prime let P be a large prime so that discrete log problem is hard and so, Alice will so, then Z p star is a cyclic group. So, this is basically Z p minus 0. So, Alice choose a generator alpha is a primitive; primitive element or generator of this group, now Alice choose a secret this is the key generation of the key generation phase of the ElGamal cryptosystem and this has to be done by the setup phase has to be done by the Alice because Alice wants to sign on the message. So, Alice will choose a a which is basically 0 less than a less than or equal to p minus 2 and then Alice will compute this beta. So, beta is equal to alpha to the power a mod p.

This is the key generation step once it is done then the setup phase is ready. So, key generation is done. So, e of A is basically so, a key set this is the key space key space is basically p alpha a beta and beta is basically congruent to alpha to the power a mod p. So, now, this is the. So, a typical key is key is basically p alpha a beta so among this. So,

this is the secret key, so, this is d of a, so d of a is basically a and remaining are all public key of Alice so p alpha beta. So, these are all public key; public key of Alice and only a is the secret key of Alice and this is the; this will be used by the signature purpose because this is the secret to the Alice. So, now, what is the signature? So now, plaintext space is what so case.

(Refer Slide Time: 25:24)



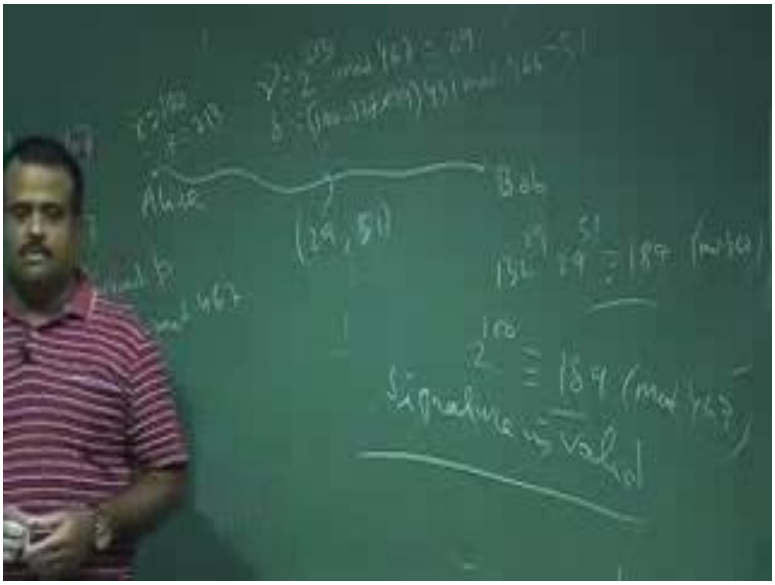
Plaintext space is basically Z n star sorry Z p star they use n anywhere no Z p star and the set of all signatures is basically Z p star plus Z p minus one this is the set of all signatures. So, now, Alice is choosing a k an Alice has to sign on the message using this k. So, suppose Alice is choosing a plaintext from this Z p start this is the message or the plaintext and this Alice wants to sign on this plaintext.

Now, for this, this is capital K, now for ElGamal cryptosystem, so there is another secret, another randomness is there which Alice can choose with which the encryptor can choose. So, here encryptor is Alice or signer. So, Alice can choose a K. So, Alice chooses a random number; choose a random K from Z p minus 1 star. So, this is a random number, Alice is choosing and then Alice is signing on the message. So, here is Bob now, Alice is signing on the message using this capital K. So, using this d of K and x comma K, so this is basically the gamma delta. So, this has 2 parts like ElGamal cryptosystem has 2 part, 2s part y1 y2 for their cipher text. So, here also in the signature, we have 2 part, this is the Z p star cross Z p minus 1. So, this is gamma, this is delta.

Now, what is gamma? Gamma is basically so, the gamma is basically alpha to the power K mod p and delta is basically x minus a gamma K inverse mod p minus 1. So, alpha is with Alice alpha is the public parameter. So, this is the secret thing Alice is using for this signature. So, this is; this 2 is sending to Bob so now, how Bob will verify? So, the verification is basically so Bob is receiving this delta and gamma. So, how Bob will verify? So, Bob will just use this key the secret key the public key of Alice e of Alice and then this x and delta this and Bob will compute this basically beta to the power gamma and gamma to the power delta and if this is congruent to alpha to the power x mod e then the verification is done.

Bob will compute beta, beta is public parameter beta to the power gamma, gamma it is the signature part and gamma to the power delta; it should be congruent to this. So, why it is true? So, basically we can just simplify this. So, beta is basically alpha to the power alpha to the power a. So, it is basically if you just calculate this alpha to the power a gamma into. So, this gamma is basically alpha to the power k. So, it is basically alpha to the power k delta and this is basically alpha to the power e gamma plus k delta, now if you come here in this equation if i multiply both side by k. So, k delta is equal to x minus a gamma under mod p minus one. So, k delta plus a gamma is basically x. So, this should be equal to congruent to alpha to the power x which is basically alpha to the power x mod p. So, the verification is done by this.

(Refer Slide Time: 30:55)



Let us take an quick example for this ElGamal signature scheme, suppose Alice choose this p 467 and Alice choose a generator for this Z p star and Alice choose a secret key 120 127 and Alice compute beta is equal to alpha to the power a mod p. So, this is basically 2 to the power 127 mod 467. So, these are basically 132 and suppose Alice wants to sign a message which is basically a 100.

Now Alice has to choose a K which is a random number. So, Alice is choosing say this K now. So, this K has to be chosen such that it as inverse. So, that also you have to take care. So, now, Alice is computing gamma. So, gamma is basically 2 to the power alpha to the power K mod p. So, mod 467 which is basically 29 and also delta; delta is basically x minus 127 into 29 to 431 mod 466 which is 51. So, these Alice is sending to Bob 29 and 51, this is gamma and delta. So, after receiving this, what Bob will do? Bob will compute this 142 to the power 29 and 29 to the power 51 and check whether this is equal to what this is basically 120 189 mod 467 and also compute alpha to the power x. This is basically also on 7 mod 467. So, these 2 are same. So, the authentication is done. So, this signature is valid. So, the signature is valid. So, this is the ElGamal signature scheme over Z p.

Thank you.