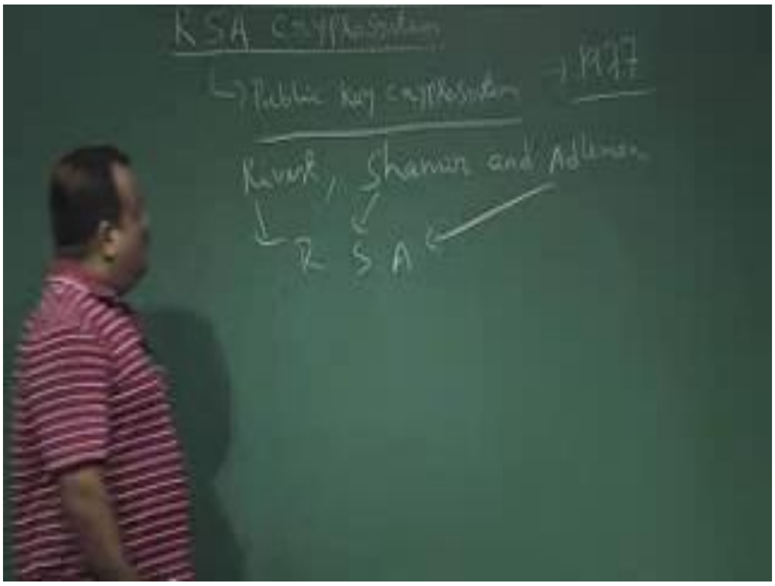
# Internetwork Security

**Prof. Sourav Mukhopadyay**
**Department of Mathematics**
**Indian Institute of Technology, Kharagpur**

## Lecture – 25
## RSA Cryptosystem

We talk about RSA cryptosystem which is basically a public key cryptosystem and which was invented by 3 cryptographers at MIT – Rivest, Shamir and Adlemen.

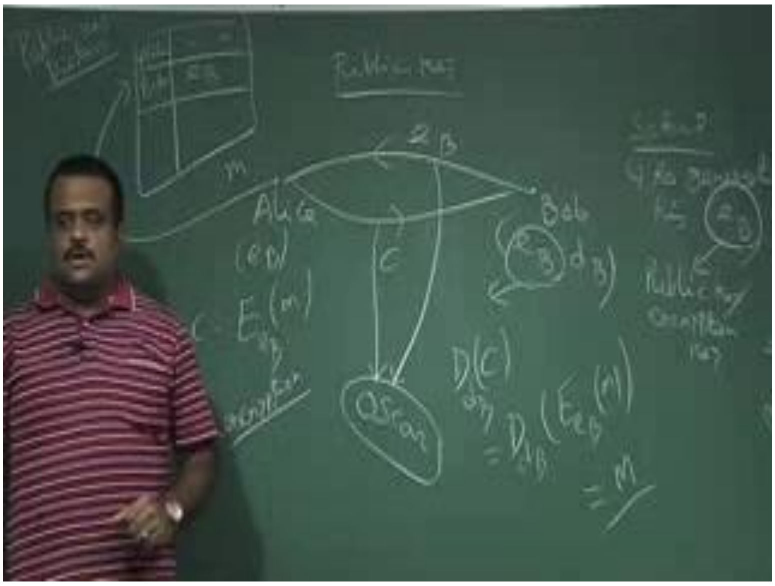
(Refer Slide Time: 00:24)



So, based on their name RSA this cryptosystem is named as RSA based on their name and it was in 1977; after the work of protocol this was invented by this 3 cryptographer at MIT.

So this is basically a public key cryptosystem. So, in public key we know that each party has two pair of keys one is encryption key which is called public key which has to make it public and another one is decryption key.

(Refer Slide Time: 01:48)



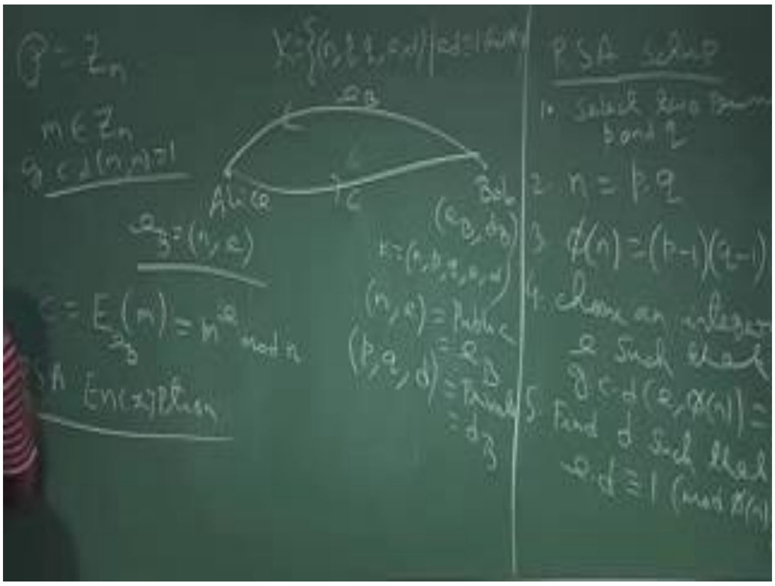
So, we have two party - Alice and Bob. So, in public key setup, if Alice wants to send a message to Bob then Alice needs to get Bob public key, so what Bob has to do? Bob has to run the setup phase to generate the setup to generate his public key, private key pair. So, this is a Bob public key or it is called encryption key because this key will be used to encrypt their massage by the sender and this is the public key or the encryption key and this is the secret key or the decryption key or the private key. So, at the setup phase Bob has to generate this pair of keys and then Bob has to make this public key public, Bob can send this public key over public channel to Alice or if there is a public key directory, public key directory then Bob can store suppose public key have stored here then Bob can store the public key here.

So, Alice can access this directory if Alice wants to send the message to Bob. So, Alice can access these directory and can face the public key of Bob e B can get the public key of Bob and then Alice will encrypt this message using the public key of Bob and generate the ciphertext c hence this Alice will send it to Bob.

And what Bob will do? This is the encryption this is the encryption done by the sender. So, what Bob will do? Bob will receive this message c ciphertext c and Bob will apply the decryption algorithm this is the public key decryption algorithm using its secret key and this is basically D of d B, c is nothing but E of e B of m and this should give us m. So, this public key should, public key decryption and this key should be chosen in such a way that they should give us m and the key should be chosen in such a way that because this key is public. So, by seeing this public key one should not able to guess what is the secret key otherwise if Oscar is having this access of this c and Oscar is also having this e, from this e if Oscar can get the secret key or the decryption key then Oscar can also do this decryption.

So, in public key setup these getting the decryption key or guessing the secret key from public key should be computationally hard. So, it should not be feasible or computationally infeasible to get the, ok. So, now we will talk about RSA cryptosystem which is a public key cryptosystem and we will talk about the key generation for this, this is the key setup phase of this RSA cryptosystem.

(Refer Slide Time: 06:04)



$\mathcal{P} = \mathbb{Z}_n$
$m \in \mathbb{Z}_n$
$gcd(m, n) = 1$
Alice
c
Bob
$e_B$
$(e_B, d_B)$
$e_B = (n, e)$
$c = E_{e_B}(m) = m^e \bmod n$
RSA Encryption
$(n, e) = $ Public $= e_B$
$(p, q, d) = $ Private $= d_B$
RSA Setup
1. Select two primes p and q
2. $n = p \cdot q$
3. $\phi(n) = (p-1)(q-1)$
4. Choose an integer e such that $gcd(e, \phi(n)) =$
5. Find d such that $e \cdot d \equiv 1 \pmod{\phi(n)}$

So, again suppose Alice wants to send a message to Bob. So, this setup has to be done by Bob. So, RSA setup and these has to be done by Bob because Alice wants to send a message to Bob. So, Alice needs to get Bobs public key. So, there are few step - first step is Bob will choose select two prime number primes p and q unusual this has to be large prime, why large? We will come to know when we talk about security of this scale. Then Bob compute this product p into q and then Bob compute this phi m what is basically p minus 1 into q minus 1, and now Bob will choose an integer, choose an integer e such that gcd of e and phi n should be 1; that means, they should be relatively prime.

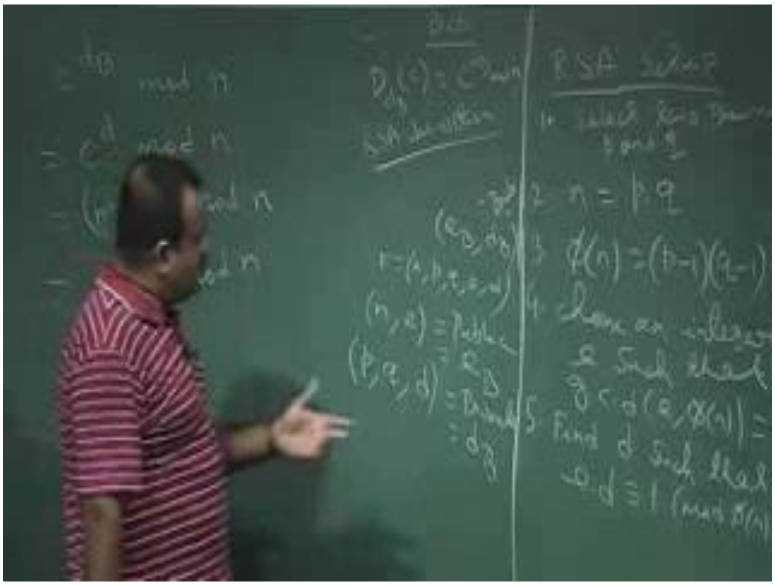
So, these we need because we want to get the inverse of e under mode phi n. So, if they are relatively prime then the inverse will exists. So, for in order to getting the inverse of e this condition is required gcd of e coma phi should be 1 and then. So, if this is the case then inverse exists then Bob choose the inverse find the inverse of e find d such that d is the inverse of e such that e into d is congruent to 1 naught phi n. So, basically Bob is choosing the finding the inverse of e which we are denoting by d. So, this is the setup phase. Now what is the key? So, key is basically, key is basically consists of n, p, q, e, d, such that e d is congruent to 1 mod phi n.

So, each key k is this tuple. So, this is the key space each key k is the tuple in p, q, e, d. So, among these who are public n and e is public this is the public key of Bob this we denoted by e of B public key of Bob and who are the private key p, q, d and this is the secret key or the private key of Bob. So, this is we denoted by e B. So, Bob generate this e B, d B. So, this is d B this is e B and then Bob sends this e B to Alice because Alice wants to send a message to Bob. So, Alice got Bob's public key which is nothing but n comma e. Now the message space is basically Z n, so Alice choose a message m from Z n such that this is relatively prime to n why we need this we will come to know soon then we will talk about the decryption is working.

So Alice selects the message and Alice wants to encrypt the message. So, what is the encryption procedure? Encryption is basically Alice compute, so encryption of m using Bob public key is basically m to the power e mod n, m to the power e mod n. So, e is public, e is known to Alice, n is known to Alice which is the public key of Bob which Alice is getting over this public channel or from the public key directory some way Alice is getting this and Alice is generating this c. So, this is the ciphertext Alice is sending to Bob. So, this is the ciphertext Alice is sending to Bob.

So, now, Bob has to get back the message, so how Bob can decrypt it? So, this is the encryption procedure RSA encryption this is RSA encryption now, what is the decryption? So, for decryption, decryption has to be done by Bob. So, Bob is receiving this c.

(Refer Slide Time: 12:16)



So, Bob is receiving this c this is doing by Bob and Bob is having its own, this is Bob private key, Bob is having its own secret key. So, what Bob will do? Bob will compute through this is the decryption, decryption using Bob secret key of c. So, Bob will compute c to the power d B mod m, Bob will compute c to the power d b mod m and this is suppose to give us n we need to verify that whether we are getting m or not.

But anyway this is the decryption process, so Bob is getting c from Alice, Bob is getting c from Alice which is the ciphertext and then Bob is having this Bob secret key d B. So, Bob will simply compute, c to the power the ciphertext to the power d B mod n. So, this is the decryption, RSA decryption and now the question is why this is equal to m, why this will give us the original message. So, that part we need to verify.

So, let us talk about that part - why this is n. So, this is the decryption c to the power e to the power b mod n. So, c is basically what? C is basically ciphertext which is basically m to the power e sorry, e B is basically d. So, this is c to the power d mod n; now c is basically m to the power e, so m to the power e to the power d mod n. Now this is basically m to the power e d mod n. Now e d we know e d is basically congruent to 1 mod phi n.

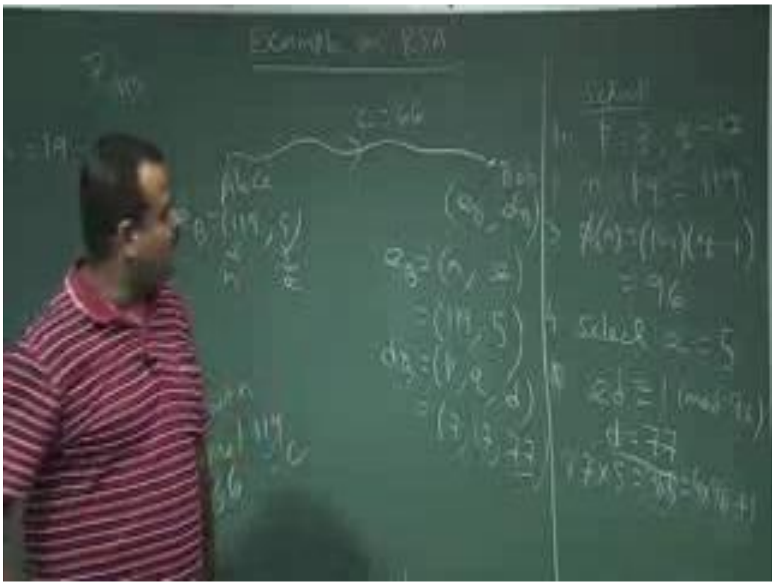
(Refer Slide Time: 14:49)

So, from here what we can write we can write e d is equal to some k phi n plus 1 where k is an integer. So, then we want to use this. So, this is basically m to the power k phi n plus 1 mod n. Now this is basically m to the power phi n to the power k into m mod n. So, these we can write m to the power phi n mod n to the power k m mod m, but m we are choosing from anyway, so into m mod n. So, now, these result we know this is basically we have seen this is the Euler's theorem and what is phi here? Phi is the Euler phi function, Euler phi function means phi n is the Euler phi function is the number of prime which is less than a number of integer which is less than m and which is co prime to n. So, that you have seen.

So, this is basically, this is coming from what is called Euler's theorem, Euler's theorem is telling because this phi n is basically what? Phi n is basically a Euler's phi functions, phi function which is defined as number of integer, number of integer less than n which is defined number of integer less than n which are co prime to n, which are that mean gcd of n and that number is one which are co prime to n this is Euler phi function and if n is equal to p q where p is prime we know if n is prime then phi n is p minus 1 because, so you have seen that now if n is equal to p q product of two prime this is also you have seen phi n is basically p minus 1 into q minus 1, this result you have seen. So, this is basically Euler phi function - this p minus 1 into q minus 1, this one is Euler phi function.

So, now Euler’s theorem is telling that, telling that if we have a let m and n - two integers such that they are relatively prime then m to the power phi n is congruent to 1 mod m to the power phi n is congruent to 1 mod n. So, this result this is basically Euler’s theorem which we have proved when we discussed the number theory, then this result we can use here. So, this will be 1 congruent to 1, so 1 to the power k. So, this is one. So, this will basically give us m - the message or the plaintext send by Alice. So, this is the correctness of the RSA decryption. So, this is the way we can decrypt the RSA. So, this is RSA decryption for reducing the Euler’s theorem.

(Refer Slide Time: 19:45)



We can take an example quick example on this RSA. So, let us take a quick example. So, you have to choose the prime. So, example on RSA - same thing Alice wants to send a message to Bob. So, Bob has to run this setup RSA setup. So, Bob has to choose the prime. So, suppose Bob is choosing two prime number 7 and q is equal to 17 these two prime number Bob is choosing and now Bob is computing n which is basically p into q, which is basically 119.

So, now, if Bob is calculating the Euler’s phi function phi n which is basically p minus 1 into q minus 1 which is basically 96 - 6 into 16, so 96. Now Bob has to calculate, Bob has to choose e such that gcd of e and such that e is relatively prime to 96 because that will guaranty that e has inverse. So, select e which is relatively so we select e is equal to

5, Bob is choosing e is equal to 5 which is relatively prime to 96. So, that will guarantee that e has a inverse mod 96.

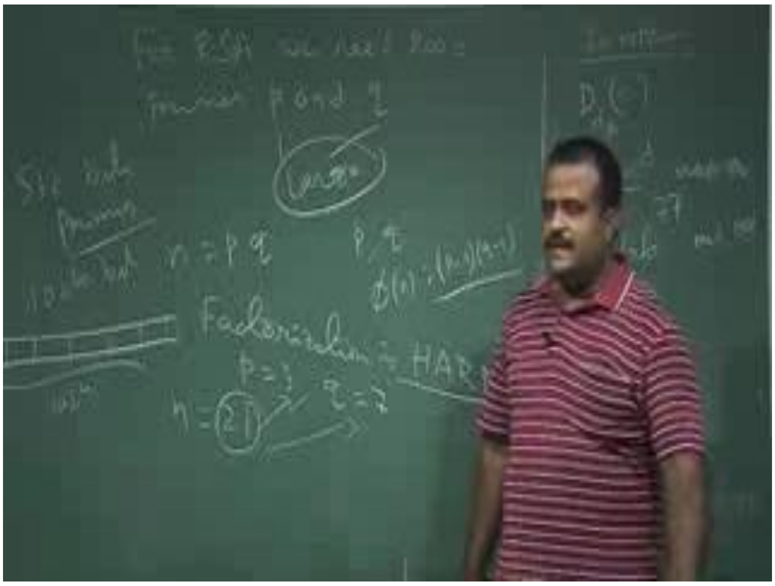
So, now we choose d we have to calculate d. So, d is the basically inverse of e, what we choose d such that e into d is congruent to 1 mod phi n mod 96. So, this inverse we can calculate which we have seen in the extended equilibrium algorithm. So, if we apply that we will be getting d is equal to 77. This we can cross multiply, we can multiply d with this and then we check whether this is equal to 1 mod this or not. So, 77 into 5 is basically how much? 385 sorry; 385, 385 can be written as 4 into 96 plus 1. So, this d is can be calculated from which we have seen in the extended equilibrium algorithm. So, we calculate d. So, d is this.

Now set up phase is complete. So, Bob's generate is public key, private key (Refer Time: 22:45) so, Bob generates e B, d B, basically e B is basically what? e B is the Bob public key and this is basically n comma e which is basically n is here 119 comma e is 5. So, this is sending to Alice. So, this is known to Alice; not only Alice this is public, everybody know this comma 5, this is n this is e.

And then what is a secret key? Secret, all are secret p, q, d, basically p, q, d phi n. So, this is basically 7, 17 and d is 77. So, this is the secret key of Bob. So, now, Alice has to choose a message which is coming from Z n, n is here 119. So, Alice say, Alice choose n is equal to 19 and this should be co-prime to 119 in order to Bob's perform the decryption algorithm. So, if you do that then, so this is the message Alice wants to send to Bob. So, then what is the encryption? So, encryption is this is the RSA encryption E of n e B which is basically m to the power e mod n. So, which is basically n is 19, so 19 to the power e is 5 and mod n is 119. So, this will give us 66, if you just calculate this, so this is also another issue how we can get, how we can do the exponent for the large number, I mean we will come to that.

So, this is 66. So, this 66 is the ciphertext and this is our c and this is sending to Bob.

(Refer Slide Time: 25:46)



So, now, how Bob will decipher it or Bob will decrypt it. So, this is the encryption, now the decryption which is done by Bob upon receiving the ciphertext c. So, the decryption is D of d B. So, d B is basically Bob secret key. So, Bob basically use this d. So, c which is basically c to the power d mod n, so c is basically 66 to the power d, d is 77, mod n, n is 119. So, this is basically 19. So, these we have to calculate and if you calculate this we can easily check this is 19. So, Bob is getting the plaintext. So, this is one example.

Now the question is there are many issues here. So, computational aspect; that means, how to; so this is very big number, this is very big number. So, how we can calculate such a - a to the power b where a b are two big number, this exponent, how we can calculate?

And another issue is how to get the primes, because for RSA we need primes. So, we need two primes p q. So, these are the issues here. So, let us just discuss some of the issues. So, for RSA we need two prime. So, RSA we need two prime’s p, q; p and q and which should be large, why large? Because, so RSA security on RSA is based on factorization is hard, we have given n which we know is an integer which is a product of two prime we know. So, this factorization if we can do this factorization then we know the p and q, if one can do the factorization then we know the p and q, Oscar knows p and q. So, once we know p q then we know phi n because phi n is basically p minus 1 q minus 1 and once we know phi n we can easily compute this because we know e, we can easily compute the inverse.

So, RSA security is based on this factorization is hard, factorization is hard. So, if you choose two simple prime, I mean two less prime say p is equal to prime and q is equal to 7. So, then product is basically 21 n. So, by seeing 21 we can guess the p is 3, q is 7. So, that is the reason we need to choose two large prime. So, p q should be two large prime large enough. So, that the factorization problem should be hard it should be difficult. Now the question is how we can get the prime number, large prime number, so we need to have say 512 bit prime number each p q should be this, not only this for it could be these bits. So, you need to have a number of these many bits, 01 bits and it should be a prime number.

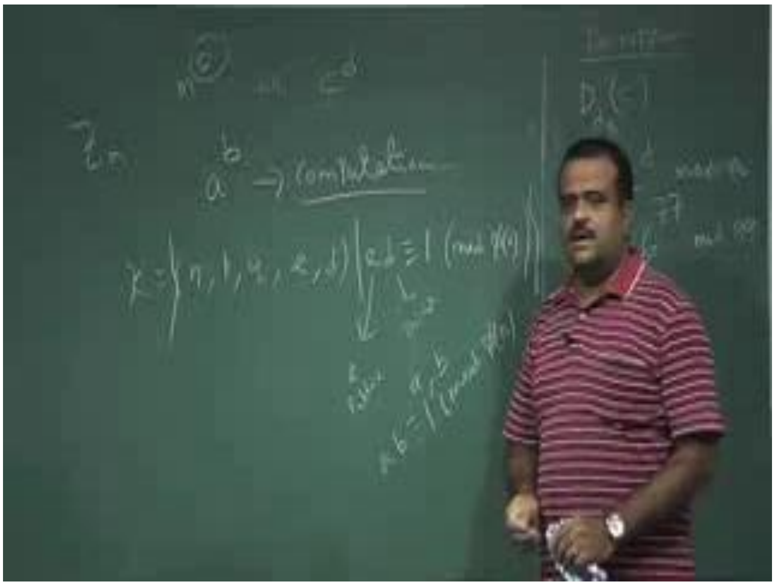
So, now the question is, is there any algorithm for which we can give the length of the bit and it should give us the prime of that bit. So, there is no such algorithm which can do that I mean which can give us the prime number of this much length. So, what we can do? We can do the primality testing, we can choose an odd integer of these many bits of say these many bits and we can check this is the prime or not. So, those are called primality test.

(Refer Slide Time: 30:20)

So, in primality test we give an odd integer n, an in test prime or not. So, this is called primality test there are some algorithm, probabilistic algorithm Miller-Rabin and there are some recent brake through which is determinist, Miller-Rabin is the probabilistic algorithm, deterministic algorithm by Prof. Manindra Agarwal from Kanpur IIT and his group. So, they had designed a case algorithm. So, they have designed, they have given a deterministic pseudo code which can take a n bit input, i (Refer Time: 31:04) which is taking an odd integer and we can test whether this is prime or not in a deterministic way.

We will talk about some primality test in a latest test. Now another issue in RSA is the computing this n to the power e or for encryption or c to the power d, so something to the power b, so this computation, so computing this.

(Refer Slide Time: 31:27)



So, usually these are big numbers. So, m is coming from Z n. So, n is very big number, message is very big and e is also big. So, these are very big number. So, how we can have this exponentials and with two big, with this big number? So, that we have to talk about.

And another property of RSA is, RSA key is basically n, p, q, e, d. So, this is a key space such that e d is congruent to 1 mod phi n. So, now, here we choose e as a public key and d as a private key or the symmetric key. Now this role can be changed actually. So, this is the one beauty of RSA this role can be changed, we can use a d as a public key - we just need to choose two a b such that a into b is congruent to 1 mod phi n, then Bob can declare which one Bob wants to be public and which one Bob wants to be kept secret, so that is up to the Bob.

So, this similarity is there in the RSA cryptosystem. So, this is one of the advantage of RSA cryptosystem we can make use, we can use any of these a b as a public key and other should be the private key.

Thank you.