Foundations of Cryptography
Prof. Ashish Choudhury
Department of Computer Science
Indian Institute of Science –Bangalore

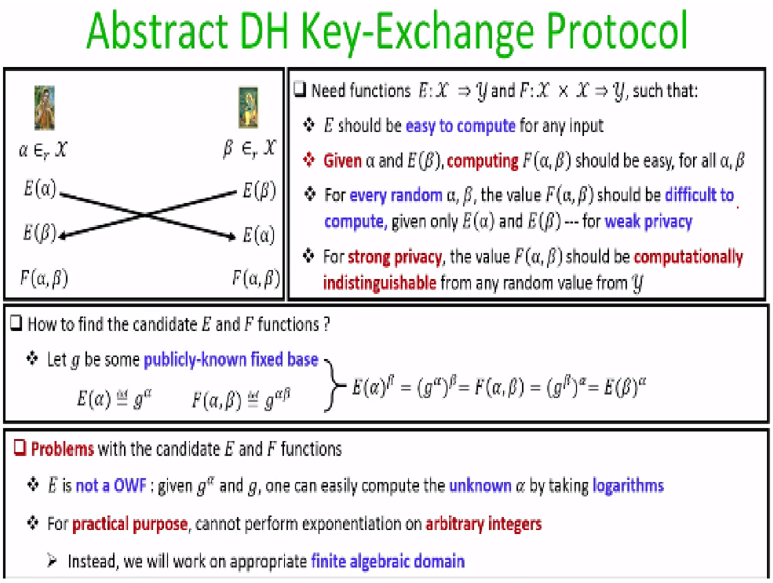
Lecture - 38
Cyclic Groups

Hello everyone. Welcome to this lecture.

(Refer Slide Time: 00:30)

The plan for this lecture is as follows. In this lecture we will introduce the concept of cyclic groups. You will see various definitions and properties and the reason we are interested to study cyclic groups is that later on we will introduce some cryptographic hardness assumptions in the context of cyclic groups which will further help us to design or develop the basic ideas which are used to design the Diffie–Hellman Key Exchange Protocol.

(Refer Slide Time: 01:00)



So remember the abstract Diffie–Hellman key Exchange Protocol which we have explained assuming that we have some special functions *E* and *F*. Just to recall what exactly the requirement from the function *E* and *F* were, well we need the following properties: we need that function *E* should be easy to compute for any input. We also need that if you are given any α from the domain of *E* and a function output E(β), then without knowing the value β it should be easy to compute the value of function *F* on the input pair (α, β) and this should hold for any (α, β).

If you want to achieve weak privacy then the property that we require here from the function *E* and *F* is that for every random (α, β) it should be difficult to compute to value of *F*(α, β) if you are just given the value of *E*(α) and *E*(β). Whereas for strong privacy we need that the value of *F* (α, β) should be computationally indistinguishable from any random value from the co-domain even if you know the value of *E*(α) and you know the value of *E*(β). Now the question is that how do we instantiate this function *E* and *F?*

It could be as follows: so imagine we take exponentiation to the public base g. So you assume that g is some publically known fixed base and we can take the candidate function *E* to be as follows: *E*(α) is defined to be this base g^{α} and this function *F*(α, β) can be defined to be the exponentiation of this with respect to this base g and a power exponentiation power is α times β. So that is my candidate *F*(α, β).

It is now easy to see that if I define my *E* function and *F* function like this. Then I am able to satisfy one of the requirements of the function *E* and *F* that I am interested in. Namely if I am given $E(\alpha)$ and if I want to compute $F(\alpha, \beta)$ then what I have to do is I have to just raise that $E(\alpha)^{\beta}$ that will give me the value of $F(\alpha, \beta)$. In the same way if I possess $E(\beta)$ and I do not know β then just knowing α and $E(\beta)$ I can compute $F(\alpha, \beta)$ by raising that value $E(\beta)^{\alpha}$.

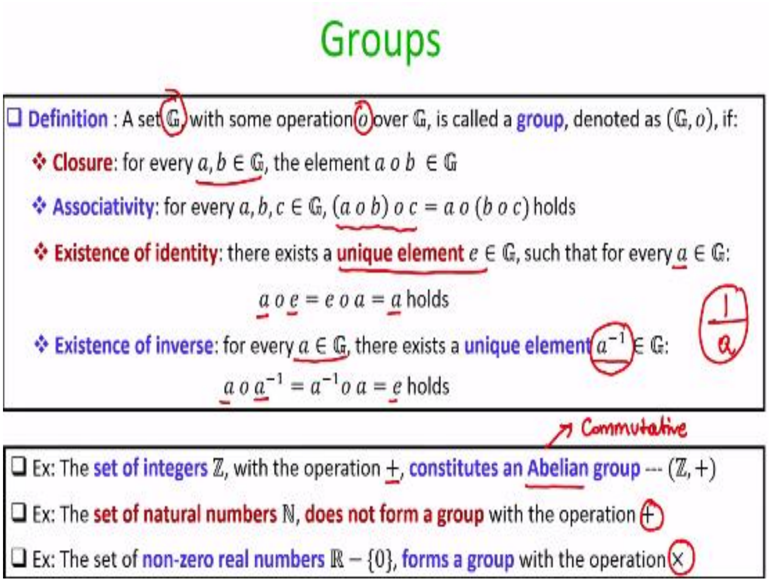
So that satisfies one of the key requirements that I need from my function *E* and function *F*, but it turns out that taking this integer exponentiation is not sufficient to instantiate that abstract Diffie–Hellman Key Exchange Protocol because there are several others security problem with this function *E* and *F*.

The first major problem is that is function E is not a candidate one-way function and to see that imagine you know the description of the base g and you know the value of $E(\alpha)$ namely g^{α} and your goal is to find out the α then it is very easy to compute the unknown α by just taking the natural logarithm. And computing natural logarithm is a computationally easy task. So the function *E* at the first place itself is not a one-way function and that means I cannot achieve this notions of weak privacy and strong privacy.

Not only the function *E* is a one-way function, the problem here is that I cannot use it for practical purposes. I cannot use this candidate *E* function and candidate *F* function for practical purposes because here my α and β can be any arbitrary integers. Whereas if I want to instantiate and implement this function *E* and *F*, I cannot work on a function or domain at the range which consist of arbitrary integers and which could be of infinite size.

Rather we will be interested to work on domains which are finite in nature so that is why we will now try to look for candidate *E* and *F* functions which are not only one-way functions and not only satisfies the requirement in *E* and *F* functions that we need for this abstract Diffie–Hellman Key Exchange Protocol, but we are also interested that those functions should be from appropriate finite algebraic domain.

(Refer Slide Time: 05:25)



For that now we have to recall the theory of groups that we had seen when we constructed our information theoretic MACs namely when we constructed universal function. So let me quickly go through the definition of groups. So a group which I denote by this symbol $\mathbb{G}$ is a set with some appropriate operation over the elements of the $\mathbb{G}$ set and we say that set $(\mathbb{G}, o)$ is a group if it satisfies the following axioms namely it should satisfy the closure property which states that for every a, $b \in \mathbb{G}$, the element $a\ o\ b \in \mathbb{G}$.

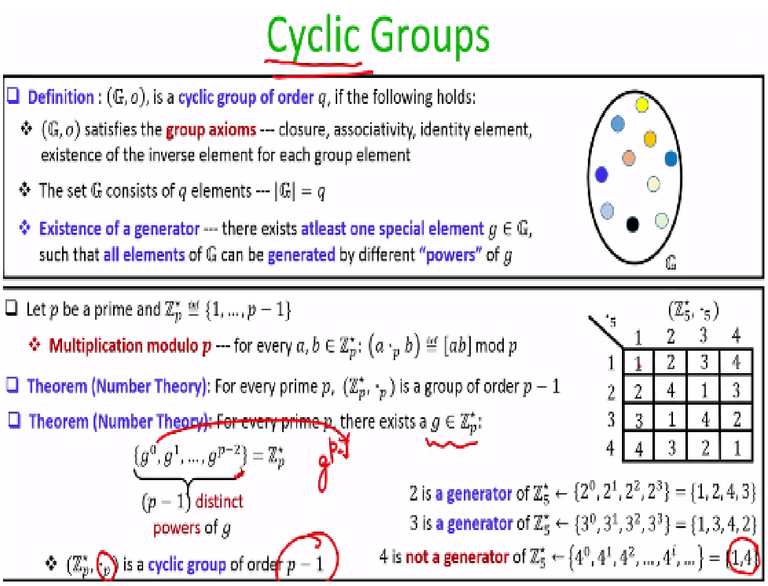
The second property is that the operation o should satisfy the associativity property namely for every a, b, $c \in \mathbb{G}$, $(a\ o\ b)\ o\ c = a\ o\ (b\ o\ c)$ holds. We need the existence of an identity element namely there should exist a unique element which I denote as e in the set $\mathbb{G}$ such that $e \in \mathbb{G}$, such that for every $a \in \mathbb{G}$ such that $a\ o\ e = e\ o\ a = a$ holds.

You should have an inverse element for every element in the set $\mathbb{G}$ namely for every element a from the set $\mathbb{G}$ there should exist a unique element which I denote by a^{-1} such that $a\ o\ a^{-1} = a^{-1}\ o\ a = e$ holds. I stress that this element a^{-1} does not mean the numeric 1/a. It is just an interpretation or just a notation for the special inverse element corresponding to the element a that I need from my set $\mathbb{G}$.

So, again just to recall we had also seen some examples of groups. The set of integers $(\mathbb{Z}, +)$ constitutes an Abelian group. What is an Abelian group? An Abelian group is a special group which satisfies all the group axioms and on top of that it should satisfy the commutative property. Namely your operation o should satisfy the commutative property and it is easy to see that the set of integers along with this operation + that satisfies the group axioms. If you take any 2 integers add them, you obtain an integer, addition is associative, 0 is the identity element because if you add 0 to any integer you obtain that integer. If you take any integer a, the corresponding inverse is –a.

Whereas it turns out that the set of natural numbers does not form a group with respect to the + operation because we do not have the additive inverse. Because additive inverse of the element, say 2 is – 2, but -2 does not belong to the set of natural numbers. In the same way the set of non-zero real numbers $\mathbb{R}-\{0\}$ form a group with respect to the multiplication operation because multiplication of any 2 real numbers gives you a real number, so closure is satisfied, multiplication is associative, the element 1 is the identity element. For every nonzero element a, the corresponding inverse is the numeric 1/a because the real number a multiplied with inverse 1/a will give you the identity element 1 and that is why the set of non-zero real numbers forms a group.

(Refer Slide Time: 08:59)



Now we are interested in special types of groups which we call as cyclic groups. So, let us see what exactly a cyclic group is? A group $\mathbb{G}$ with respect to an operation o is called a cyclic group of order q if the following conditions holds: first of all, ($\mathbb{G}$, o) should satisfy the group axioms namely closure, associative property, existence of identify, existence of the inverse for each group element. If it is a Abelian group then it is fine, but we do not need it to be an Abelian group.

So pictorially imagine that we have certain elements from this set $\mathbb{G}$ and this operation o satisfies all this group axioms. The second proper requirement here is that since I am saying that the group $\mathbb{G}$ is a cyclic group of order q. By order q, I mean that there are q number of elements in my set $\mathbb{G}$. So, that is what I mean by the order of the group to be q.

And why it is called cyclic group? It is called a cyclic group because I need a special element which I call a generator which I denote by say g. This g should be one of the elements from this set $\mathbb{G}$ such that all the elements from set $\mathbb{G}$ can be generated by this special element g by different powers. Here "powers", it will be clear to you very soon what exactly I mean by powers here.

Basically, the idea here is that this special element g which I call as generator it has the capability, it has the capacity to generate all the elements in your set $\mathbb{G}$. If I have at least one such special generator element which I call as generator, then my group $\mathbb{G}$ is called as a cyclic, but I do not have any such special element g then my group $\mathbb{G}$ is not called a cyclic group.

So, let me explain to you what exactly I mean by generating different elements by different powers of the element generator. So let p be a prime and I define a set Z_p^* which consists of the element 1 to p – 1 and now let me define a multiplication operation which is different from the normal arithmetic multiplication so the operation that I am going to define is it is denoted by $\cdot_p$ and which is also called as multiplication modulo p.

The way this multiplication is going to be performed is as follows: If I take any 2 elements a, b from the set Z_p^* then multiplication modulo p of a and b is same as you multiply a and b numerically and take the reminder with respect to the modulo p. Whatever reminder you obtain that will be a number in the range 0 to p-1 and that is the way we define this multiplication modulo p operation.

Now I am going to state a result which is a standard result which follows from number theory. I am not going to prove that, but if you are interested in the proof of this theorem then you can refer to any standard reference from the number theory. So the theorem basically states that for every prime number p the set Z_p^* along with the operation multiplication modulo p is a group or it constitute a group consisting of p – 1 elements. That means the group order is p – 1.

So we are not going to prove that, but I am going to demonstrate that indeed this is true if p is your prime. So I am taking p = 5 here and Z_5* basically consist of elements {1, 2, 3, 4}. So what I have done is along the rows I have written down the element 1, 2, 3, 4 and along the columns I have denoted the elements 1,2, 3, 4 and this is like a matrix here.

And what I have done here is the $(i, j)^{th}$ entry denotes the results of (i . j) modulo 5 here. If I consider this entry, this is the result of (2 . 2) modulo 5. In the same way (4 . 3) modulo 5 is going to give you 2 and so on. So you can see from this matrix that your closure property is satisfied. You take any i and any j in the range which belongs to Z_5* and perform the operation (i . j) modulo 5, you are going to obtain the number in the range 1 to 4.

The multiplication operation that we have defined here it satisfies the associativity property that means if I take $(i, j)^{th}$ entry, then it does not matter in what order I multiply $(i, j)^{th}$ entry and take modulo 5, I am going to obtain the same reminder. The identity element is the element 1 here because if you see this matrix here and if you focus on the column under 1 here then under that 1, 1 dot 1 modulo 5 gives you 1, 2 dot 1 modulo 5 gives you 2, 3 dot 1 modulo 5 gives you 3 and 4 dot 1 modulo 5 gives you 4. So the element 1 serves as the identity element here.

Now under each element you will have the corresponding inverse element. So the inverse of 1 is 1 here because 1 dot 1 gives you 1. The inverse of 2 is 3 because 2 dot 3 modulo 5 gives you the identity element 1, the inverse of 3 is 2 because 3 dot 2 modulo 5 gives you the identity element 1 and inverse of 4 is 4 because 4 dot 4 modulo 5 gives you the identity element 1.

So you have all the axioms satisfied and now you can see, you have some special generator elements present here as well. I am going to demonstrate that as well. Before going into whether the generator element here exist or not, let me first define what we mean by group exponentiation in this set Z_p*. For any element g belonging to set Z_p*, I define g^0 to be 1. And I define g^1 to be g because indeed g^0 modulo p you are going to obtain 1 modulo p which is same as 1 and g^1 if g is an element of Z_p*. Of course, it is less than p and if you do g^1 and then take mod p then the effect of mod p does not effect. You are going to obtain g only. Whereas I define g^i to be the multiplication modulo p operation applied i – 1 times.

And it turns out that it does not matter whether I take the reminder at the end or if I take on reminder after performing each individual **.** p operation i – 1 times, the results will be same because that comes from the associativity property of my group Z_p*. So I can define g^i to be the same as you perform the numeric g^i and then you take a final mod p to bring back the result in the range 0 to p – 1.

So, because of the way g^i is defined for the case i to be 0, i to be 1 and i to be a generic i, I can say that I can define my g^i. I can use the notation g^i in the set Z_p* to denote the value numeric g^i modulo p so that is a notation I am going to use here and that is what I mean by power of an element g in the set Z_p* here.

So again I am going to state another well known result from the number theory which states that for every prime number p, there exist at least to one element g in the set Z_p* such that when you compute when you raise g to different powers and perform modulo p operation namely you do g^0 mod p, g^1 mod p up to g^{p-2} mod p, then you are going to obtain all the elements in the set Z_p* in some arbitrary order. That means p – 1 distinct powers of this special element g is going to give you all the elements in Z_p* and that means you have at least one generator present in this set Z_p*. That is what I mean by generating all the elements of the set $\mathbb{G}$ by different powers.

Your $\mathbb{G}$ here in this particular example is Z_p* and the claim from the number theory is that there exist at least one element in this Z_p* such that if you raise g to the different powers here and perform the group operation namely the multiplication modulo p operation you are going to obtain all the elements of Z_p*. Again I am not giving a proof of this, but if you are interested in the proof you can see any standard reference.

Let us see whether this theorem holds for the current example that we are considering here. So if I take the element 2 which is an element of Z_5* and perform 2^0 modulo 5, 2^1 modulo 5, 2^2 modulo 5 and 2^3 modulo 5, I am going to obtain the elements 1, 2, 4, 3 respectively. So notice that I am not obtaining all the elements of Z_5* in the exact order. But I am obtaining all the elements in some arbitrary order.

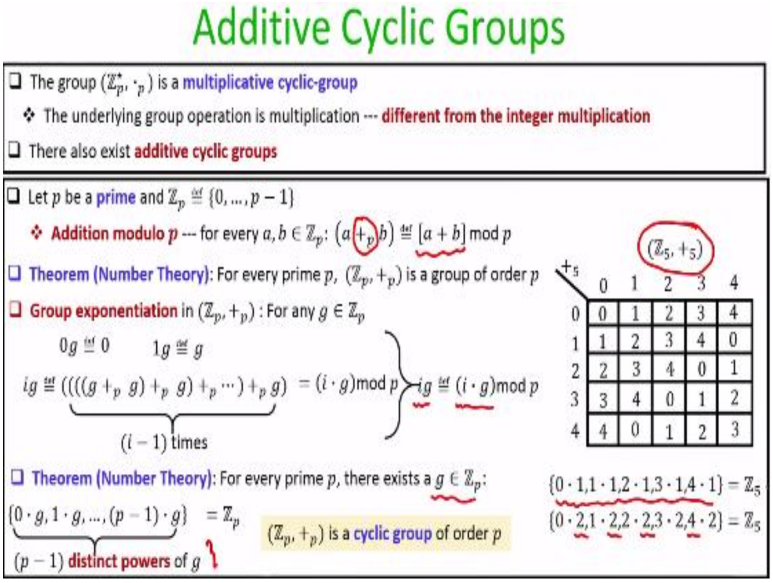
So for the definition of cyclic group, the requirement is that different powers of g should give you all the elements of that set $\mathbb{G}$ in any arbitrary order. The order does not matter here. In the same way the element 3 is also a special element here because 3^0 modulo 5, 3^1 modulo 5, 3^2 modulo 5 and 3^3 modulo 5 is going to give you the element 1,3, 4, 2 namely the entire Z_5*.

But if I take the element 4 and try to raise or compute different powers of 4, I am not able to generate all the elements of Z_5*. I could generate only the elements 1 and 4. So since the number theory result that I am stating here states that I have some special element g which has the capability to generate the entire set Z_p*. That means the set Z_p* along with this multiplication modulo p operation is a cyclic group of order $p - 1$.

Because it has $p - 1$ elements and indeed in this example 2 is the one of the generators of Z_5*, 3 is also one of the generators of Z_5*, but 4 is not a generator of Z_5*. Now you might be wondering that why the name cyclic here. The reason I am calling it cyclic because as soon if you have a generator g say for example for the group Z_p*and then if you compute the next power of g namely g^{p-1} you are going to obtain one of the elements which is already there in Z_p*. Essentially you are going to obtain the element g^0 only.

So again the proof for that follows from the number theory, but I am not going to prove that if you compute g^{p-1} you will get the same value as g^0. The next power of g will give you g^1 and so on. In that sense it is cyclic that means as soon you go up to the limit g^{p-2}, and if you go to start taking the next sequence of powers of g you will start getting back the same cycle. You will obtain the same elements of Z_p* and it will keep on happening and that is why the name cyclic group. So cyclic groups, just to summarize cyclic groups are special types of group which has at least one generator.

(Refer Slide Time: 20:59)



So the group Z_p* with respect to the multiplication modulo p operation constitutes the multiplicative cyclic group because there the operation was multiplication. It was not the natural multiplication. It was not the integer multiplication, but it could be interpreted as a multiplicative operation. It turns out that we can also define cyclic groups based on the notion of addition operation and let us do that.

So, let p be a prime and I define a set Z_p to be the set of integers 0 to p – 1. So the difference between Z_p* and Z_p is that is that the element 0 is not there in Z_p*, but the element 0 is now allowed in Z_p. Now let me define an addition operation in this set Z_p which I call as addition modulo p denoted by this symbol $+_p$ and the addition modulo p of some pair of numbers a, b from the set Z_p is nothing but to perform the numeric or integer addition a and b and take the modulo p. So that the resultant is an element in the set 0 to p – 1.

So again let us take an example here the set Z_5 consist of the integer {0, 1, 2, 3, 4} and what I have done here is I have done the matrix which denotes the result of performing the + modulo 5 operation on any pair of elements in the set Z_5. Again, there is a well-known fact from the number theory which states that if you take any prime p then the set Z_p along with the operation addition modulo p constitutes a group. But now the order of the group is prime namely it has p number of elements because you are now having element 0 to p – 1 whereas the set Z_p* was a multiplicative group of order p – 1.

So now let us see how we can interpret the group exponentiation in this additive group. So we defined 0 times g to be 0 and we defined one times g to be g where g is any element in the set

Z_p. Whereas i times g is defined to be the result of this addition modulo p operation being applied on the element g, i – 1 times. It turns out that it does not matter whether I take the mod at the last or whether I take the mod after every + operation the result is going to be the same because that follows from the associativity property of the + operation and hence I can say that i times g is the same as the integer multiplication of i and g modulo p.

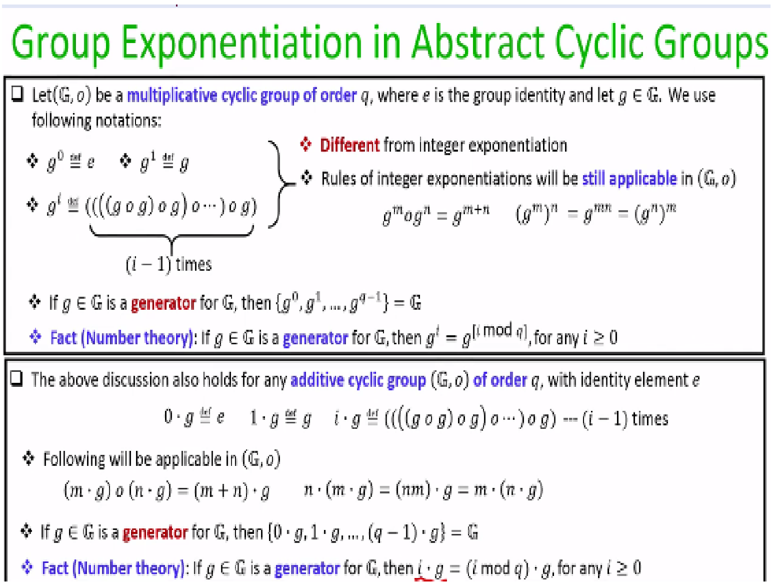
So when I say i times g that did not mean that I am multiplying i and g, i times g is the notation I followed by g and i followed by g is same as the integer multiplication of i and g modulo p. So based on these 3 ways or the way this group exponentiation is defined with respect to this + operation I can use the notation that i **.** g is same as integer multiplication of i and g modulo p.

And again there is a well known result from the number theory which states that for every prime p there exist at least one special element g in the set Z_p such that 0 times g, 1 times g up to p – 1 times g namely the p – 1 distinct powers of g is going to give back all the elements of the set Z_p in some arbitrary order. So now here the exponentiation is basically treated as if you want to compute g^x.

Basically, g^x here is interpreted as if you are performing the + modulo p operation x – 1 number of times. So that is the interpretation of power of g when I am considering the underlying group operation in the additive sets. So again, in the context of Z_5 the element 1 turns out to be one such special element where all the different powers of 1 is going to give you back all the elements of Z_5.

Same holds for 2 as well different powers of 2 is going to give you back all the elements of Z_5. That means the set Z_p along with the operation + modulo p is a cyclic group of order p. So we had seen examples of cyclic groups based on the multiplication operation.

(Refer Slide Time: 25:26)



So we had seen examples of cyclic group based on addition operation. Now let us define what we mean by group exponentiation in abstract cyclic groups. So for explanation I am assuming that my $\mathbb{G}$ is some abstract group where the underlying operation is a multiplicative operation. It need not be an integer multiplication, but it can interpreted in the multiplicative sense and it has an order q namely it has q numbers of elements. Since it is an abstract group, I denote the identity element to be e and let g be an element of this group $\mathbb{G}$.

Then we use the following notation. So since we are using multiplicative notation here for that abstract group, I will use the notation g^0 to denote the identity element and g^1 to denote the identity element. And the notation g^i to denote the element which I obtain by composing or performing the group operation on the element g, i – 1 number of times. I stress that this notation is completely different from the integer exponentiation. This is just a notation, g^i does not mean that I am multiplying g, i – 1 times. It is basically just a notation which I used to represent that I am performing the group operation on the element g, i – 1 number of times.

However, it turns out that the rules of the integer exponentiation are still applicable in this abstract multiplicative group. Namely if I take the group element g^m which is basically the element g composed to itself m – 1 numbers of times as per the group operation and I take the another group element say g^n which is the group element g composed to itself n – 1 number of times. Then if I perform the group operation on these 2 elements then the result will be the same as the element g being composed m + n – 1 number of times.

In the same way, if I take the element g^m and perform the group operation on that element n –

1 number of times then I will obtain the same result which I obtain by performing the group operation on the element g, m+n – 1 number of times and so on.

Moreover, if this element g is a cyclic group then it turns out that different powers of g and again by different powers of g I do not mean the integer exponentiation. Power by different powers of g means the definition of group exponentiation in that abstract sense. So if this g is a generator then different powers of g ranging from the 0^{th} power to the $q-1^{th}$ power is going to give me back all the element of set $\mathbb{G}$ in some arbitrary order.

And finally an interesting fact which we are going to encounter later or use later is the following: if you have any element g which is a generator of the group then the element g^i is the same as the element g^i modulo q, that means you can perform mod q operation in the exponent as well. So for any i which is < q this fact is trivially true because g^i and g^i mod q are same if i < q.

But what this fact says that if i is larger than q, then g to the power that larger power i is going to give you back the same answer as the result which you will obtain by raising g to the index i modulo q. Again, I am not giving you the proof for this you can refer to any standard text on number theory for proof of this. Now this discussion that we have till now here is with respect to a multiplicative cyclic group. We can extend our definition for any abstract cyclic group where the underlying operation is additive.

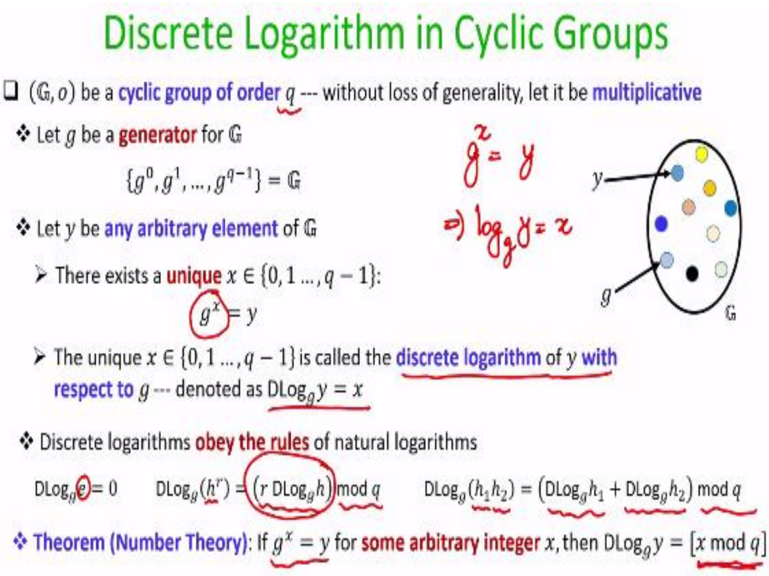
So we can define g to the g times g to be e or the identity element namely the 0^{th} power of g here is the identity element and g^1 in the additive cyclic group will be interpreted as one times g and definition says that 1 times g is going to give you the element g. And g^i in this additive cyclic group will be interpreted as i times g which is defined to be the group operation, or the additive group operation performed on the element g, i – 1 times.

And it turns out that the rules of exponentiation hold in the additive cyclic group as well. Namely if I take the element m times g which is same as g^m in the additive cyclic group and another element n times g which is the equivalent of g^n in the additive cyclic group. If I perform the group operation then the result will be the same as m + n times the element g namely the equivalent of the element $g^{(m+n)}$.

And the same holds like this. If I take the element m time g and then if I perform n times that element, then the result will be the same as nm times that element g and so on. Moreover, if the element g is a generator, then different powers of g is going to give me the entire set $\mathbb{G}$ in some arbitrary order and the different powers of g is written as 0 times g, 1 times g and q- 1 times g.

And as it was the case for the multiplicative cyclic group I have a corresponding fact here as well that if g is a generator then any i times g which is the corresponding equivalent of g^i is same as i modulo q times g. That means if i > q and then if you want to compute that i times g then you can first reduce that index i modulo q and then raise that index to the element g to get the resultant answer.

(Refer Slide Time: 31:55)



So now we have defined a notion of cyclic groups and now let us see what we mean by discrete logarithm in the cyclic groups. So imagine you are given a arbitrary cyclic group of order q and without loss of generality assume that the underlying group operation is a multiplicative operation. It need not be an integer multiplication, but for notation purpose we will use the multiplicative notation here.

Since the order is q that means it has finite number of elements. So this color dots denotes the various element in your set $\mathbb{G}$. Since it is a cyclic group it has some generator at least one generator which I denote by g. So I have highlighted that element g here and as per the definition of cyclic group different powers of this element g is going to give you the entire set $\mathbb{G}$ in some arbitrary order.

What does that mean is that if you take any element y from this set $\mathbb{G}$ then there exist some unique index x in the range 0 to q – 1 such that g^x would have given you the element y and remember g^x is performing the group operation on the element g, x – 1 number of times. It does not necessarily mean that I am multiplying g, x number of times. I am performing the underlying multiplicative group operation on the element g, x – 1 number of times.

So the reason there exist a unique x in this range 0 to q – 1 such that g^x would have given you y. It comes from the fact that the element g is a generator. Now this unique x in the range 0 to q – 1 it is called the discrete logarithm of your element y to the base g which we denote by this notation $\text{DLog}^{y}{}_{g} = x$.

And you can consider this discrete log to be an equivalent of the natural log. In the real number world if you have any real number g to the power giving you y then we say we define that $\log^{y}{}_{g} = x$. What we are trying to do here is that we are trying to give an equivalent definition in the discrete world namely in the context of a group where we have some finite number of elements say q numbers of elements and since we are considering a group. The elements here are discrete. Between any two group elements will not be an arbitrary group element coming up. So that is why this logarithm that we are defining the notion of logarithm that we are defining in this cyclic group is called as the discrete logarithm. Interestingly, it turns out that the discrete logarithm obeys the rules of natural logarithm i.e $\text{Dlog}_g{}^{e} = 0$, $\text{Dlog}_g h^r = r\,(\text{Dlog}_g{}^{h}) \bmod q$.

Why we are taking modulo q is that this is the thing that we are having the bracket that may go out of q, that may cross the range 0 to q – 1, but as per the definition of discrete log the index the discrete logarithm has to be in the range 0 to q – 1 and that is why we are taking modulo q here and in the same way $\text{Dlog}_g h_1 h_2 = (\text{Dlog}_g h_1 + \text{Dlog}_g h_2) \bmod q$.

You are going to verify these facts. These are some simple exercises for you and the final fact that we are going to use in the context of discrete logarithm is that if you have some g^x given to be y where x need not be in the range 0 to q – 1. Then the $\text{DLog}^{y}{}_{g}$ is same as x modulo q.

Well if your x that you are given is indeed in the range 0 to q – 1 then x module q is same as x. but the interesting fact here is that if you have if you are given an x which is outside the range 0 to q – 1 such that g^x is given y and if you are interesting to compute the $DLog^y_g$ is same as performing the operation x module q. Again, this is a well known fact from number theory which I am not going to prove here you can see any standard text for the proof of this theorem.

So that brings me to the end of this lecture. To summarize in this lecture, we have introduced the notion of cyclic groups and we have seen the definition of discrete logarithm. In the next lecture we will see some candidate cryptography hardness assumptions based on cyclic groups and then how using those cryptographic hardness assumptions we define the exact Key Exchange Protocols. Thank you!