

Foundations of Cryptography
Dr. Ashish Choudhury
Department of Computer Science
Indian Institute of Science – Bengaluru

Lecture - 56
Secret Sharing

(Refer Slide Time: 00:29)



Hello everyone. Welcome to this lecture. So in this lecture we will discuss about interactive protocols namely our focus till now was on solving the problem of secure communication where we had two entities a sender and a receiver and we extensively discussed how to design algorithms to solve the problem of secure communication. But now what we are going to discuss scenario is a well-known problem where we have multiple entities.

And our goal is to design cryptography protocol which requires interaction among the entities. So, specifically the roadmap for this lecture is as follows. We will introduce the problem of secret sharing. We will see additive secret sharing, replicated secret sharing and then we will see the classic construction of secret sharing scheme due to Adi Shamir.

(Refer Slide Time: 01:17)

Secret Sharing : Motivation



❑ Access to the locker:

❖ Only if at least two managers come together and enter their password

❖ No access, if only a single manager enters its password



So let us see the motivation of secret sharing. So imagine we have a banking applications say where the locker in the bank it is accessible only by the managers in the following way. The password for the locker is shared among the three managers and it shared in such a way that if only two of the managers go together and enter their respective password the lockers can be accessed.

But if only a single manager tries to enter, enter the password and access the locker the access is not possible, right. So, for instance if the second manager goes and tries to open the locker it should not able to do that. In the same way, the third manager go it should fail, but if we take any set of two or more number of managers and they go and enter their respective password they should be able to access the locker. So that is what we required here.

(Refer Slide Time: 02:14)

Secret Sharing : Motivation

Access to Russia's Nuclear Weapons in 1990's



Okay. In the same consider another real-world scenario. This is real-world scenario which really happened during 90s. So this is regarding how Russia's Nuclear Weapons was accessible by the top leaders of the countries. So it is believed that the password to launch the Russia's nuclear weapon it was shared between top three entities of the country namely the president, prime minister and defense minister in such a way that the weapon could be accessed or launched only if at least 2 of the 3 entities come together and enter their password whereas if only single entity try to launch or access the weapon the access will be denied.

(Refer Slide Time: 02:55)

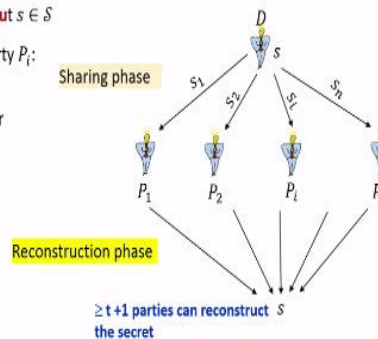
(n, t) Secret Sharing Scheme [Shamir 1979, Blakley 1979]

- ❑ A set of parties $\mathcal{P} = \{P_1, \dots, P_n\}$ connected by **pair-wise private and authentic channels**
- ❑ A **designated dealer** $D \in \mathcal{P}$, with a **secret input** $s \in \mathcal{S}$
- ❑ Goal: to **distribute a share** s_i of s to every party P_i :

❖ **Should be impossible** for any set of t or less number of share-holders to pool their shares and **reconstruct back** s

- Perfect-secrecy
- Computational-secrecy

❖ **Should be possible** for any set of $(t + 1)$ share-holders to pool their share and **reconstruct back** s



So both these applications can be extracted by the following problem which we call as n, t Secret Sharing and this problem was independently formulated by Shamir in 1979 and Blakley in 1979.

So what we are given here is we are given the following setting. We have a set of n parties P_1 to P_n and they are connected by pair-wise private and authentic channel. What it means is, if any information P_i wants to send it to P_j .

We assume that it has a dedicated channel with which it is connected to P_j and anything P_i sends over that channel to P_j it will be received correctly and securely by P_j . If you are wondering how exactly that such channels are available in real-world well, we can use any of the well-known secure communication protocol that we have extensively discussed till now to ensure that such channels are available between every pair of parties.

Now apart from these n parties we have a designated party among those n parties and everyone will know the identity of that party and it is called as dealer and dealer has some private input, a secret little s from a bigger space which is a set of all possible secrets. Now the goal of this dealer is to distribute its secret among the n parties by coming up or computing a share S_i for each of the party and giving a share to each of this party.

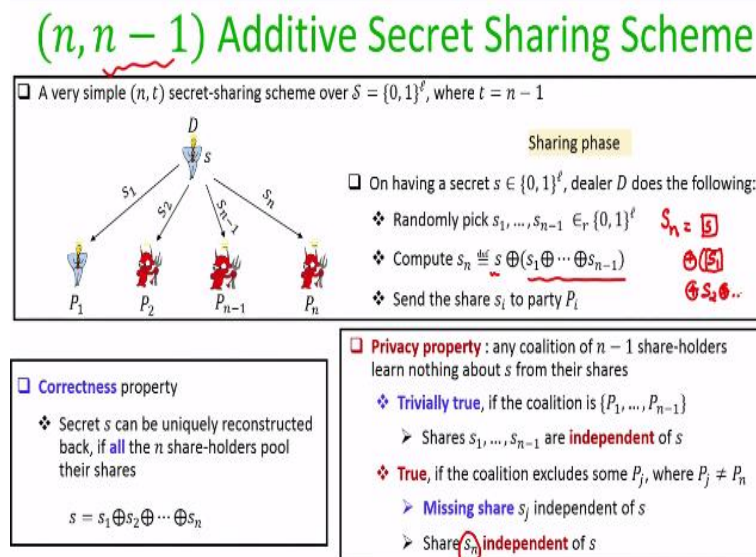
So first shareholder or party 1 gets a share as 1, second party should get a second share; the n th party should get the n th share. And these shares are distributed over the pair-wise channel with which the dealer is connected to the individual shareholders. Now we require the following properties from these distributions of shares. We require that it should be impossible for any set of t or less number of shareholders to exchange their shares and reconstruct back the secret s .

And depending upon whether the set of t shareholders who are trying to reconstruct back the secret whether they are computationally bounded or they are computationally unbounded, we achieve either perfect secrecy or computational secrecy. So that is a first requirement from this distribution of shares. On the other hand, if any set of $t + 1$ or more shareholders pull together their shares then it should be possible to reconstruct back the secret s .

So the parameter t here is acting as a threshold for you that mean we require a sharing mechanism such that if any set of t or less number of shareholders they come together, they should fail to access or they should fail to reconstruct the secret. The share should be completely

independent from the underlying secret. On the other hand, if any set of $t + 1$ or more number of shareholders come together the secret should be reconstructed. It should be possible to reconstruct back the secret.

(Refer Slide Time: 05:39)



So let us see a very simple construction of $n, n - 1$ additive secret sharing scheme. So basically here my threshold is t , that means I require a sharing mechanism where all the n shareholders should come together to reconstruct back the secret. But if any single shareholder is missing then the secret should not be; it should not be possible to reconstruct back the secret. And why it is called additive secret sharing it should be clear to you very soon.

So here my secret space is a set of all possible albeit strings. And as I said my threshold $t = n - 1$. The sharing algorithm is as follows. So imagine dealer has a secret s consisting of little albeit. To share it, it picks $n - 1$ shares randomly from the set of, the set of albeit strings. That means the first share is a random albeit string, the second share is a random albeit string and in the same way the n th share is also a random albeit string.

Now once the first $n - 1$ shares are fixed by the dealer the n th share is computed to be the xor of the share and a secret s . And once the n shares are computed the dealers sends the respective shares to the respective shareholders namely the i th share S_i is given to the party P_i over the dedicated secure and authentic channel between the dealer and the i th party. So the correctness

property here is trivial for this secret sharing, namely if all the n shareholders come together and exchange their shares with each other then indeed they can perform the xor of all the n shares and they will uniquely get back the underlying secret s which was shared by the dealer.

The next formally proved the privacy property here. So for privacy our goal is to show that if among this n shareholders any $n - 1$ shareholders come together and pull their shares they should not learn anything about underlying secret s . So we divide the proof into two cases. So consider the case when the set of $n - 1$ shareholders who are corrupted and they are trying to reconstruct the secret are the first $n - 1$ shareholders.

It turns out that if the first $n - 1$ shareholders are corrupt then from their shares they learn absolutely nothing about underlying secret s because if you see the sharing algorithm the first $n - 1$ shares they are picked independently of the actual secret of the dealer. So that means if the adversary controls the first $n - 1$ shareholders' and access their shares the adversary launch absolutely nothing about the dealer secret. That is the case one.

On the other hand, consider the case when the set of $n - 1$ shareholders definitely include the n th shareholder, because the n th shareholder its share is a function of a secret s and a remaining $n - 1$ shares. So the second case is when the n -; the coalition of $n - 1$ shareholders excludes some party P_j where P_j is definitely different from the n th party. That means the n th party is definitely in the coalition here.

So for simplicity you can imagine that my $P_j = P_1$ that means the adversary is corrupting the last $n - 1$ shareholders here. It turns out that the missing share which the set of $n-1$ shareholders are missing in this example the share as 1 that is independent of the secret s because that was picked randomly by the dealer and that ensures that even though the n th shareholders learns S_n which is computed as the xor of the actual secret of the dealer and a remaining $n-1$ shareholders.

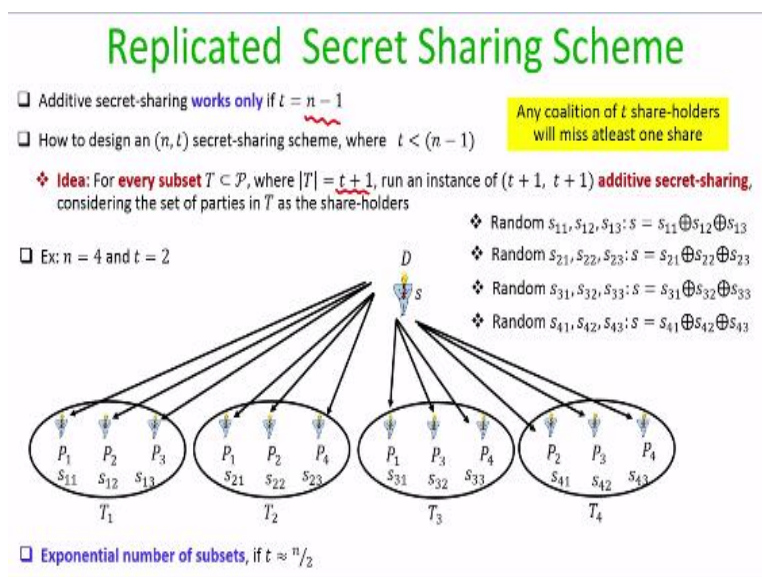
It ensures that n th share is also independent of the secret s . Because for instance, if we are considering the case where P_1 is missing in the coalition then from the view part of the attacker, attacker knows that the value of S_n which it has learnt is equal to some unknown secret s , so I

am highlighting; I am putting this s in the box here to show, signify that it is unknown to the attacker and even the first share is also not known to the attacker here.

Because I am considering the case where all but the first party is under the control of the adversary but rest of the values is known to that; unknown to the attacker. So more or less we are exactly in the same situation as we were in the onetime paired scheme. So you can imagine that this S of n is nothing but a onetime paired encryption of the message s with a key; where the key is nothing but the xor of the last $n - 1$ pieces which are known to the adversary plus the random value which is not known to the attacker.

That means even though adversary is seeing S of n it cannot figure out whether S of n is actually a share corresponding to the secret s or corresponding to a secret s prime. Because it does not know the value of the missing share S_1 which was randomly picked and independently picked; and which was randomly picked and independent of the actual secret of the dealer. That ensures that even if the adversary controls a n th shareholder it will also absolutely nothing about the underlying secret s . And that is why the secret sharing satisfies the requirement of an $n, n - 1$ secret sharing scheme.

(Refer Slide Time: 11:01)



So it turns out that the additive secret sharing that we have discussed it works only if my threshold is $n - 1$. But in general I might be interested to design a secret sharing where my

threshold may not be $n - 1$, my threshold could be strictly less than $n - 1$. So now let us see a solution, a naive solution of coming up with a secret sharing scheme for any threshold t which is less than $n - 1$.

So the idea here is we take every proper subset of the set of n parties say t ; say big T where a size of big T is $t + 1$. So we take every possible subsets of size $t + 1$ of the end of parties and run a dedicated independent instance of additive secret sharing among that subset of $t + 1$ parties as the shareholders with the threshold be $t + 1$ and idea here is that if we do this for every subset of size $t+1$ then when it comes to the actual coalition of t shareholders who might try to learn about the secret that coalition of t shareholders.

They miss at least one share to reconstruct back the actual shared secret. So what I am trying to say is a rest demonstrated by this example. So imagine my $n = 4$ and I want to design a scheme where my threshold t should be 2. That means any subset of three shareholders should be able to reconstruct back the secret, but any set of two shareholders if they try to pull their share they should fail to reconstruct back to secret.

So the idea here is that the dealer treats this set of four parties into different subsets of size three parties. So, say t_1 is the first subset, t_2 is the first subset and like that we have the fourth subset t_4 . Now in the first subset dealer consider P_1, P_2, P_3 to be the shareholders and it runs an instance of additive secret sharing scheme with the threshold being $t = 2$. But considering only the parties P_1, P_2, P_3 to be the shareholders, namely dealer picks random shares s of 11, s of 12, s of 13 such that their sum is equal to the secret s and since shares are given to the respective shareholder P_1, P_2, P_3 .

Independently dealers runs another instance of additive secret sharing for the same secret s and picks or computes three shares such that their sum is equal to the secret s and respective shares are given to P_1, P_2, P_4 who are the shareholders in this second subset t_2 . In the same way dealer creates another instance of the additive secret sharing for the subset t_3 and another instance of additive secret sharing for the shareholders in the set t_4 .

Now what is the total share for the party P1, the total share of the party P1 is all the shares which it is getting from the respective subset in which it is present. So the share of P1 will be the piece S11 because it is present in the subset t1 as well as the piece as t2 because it is present in the set t2, and this will be the total shares for the party P1. In the same way the share for P2 will be S12, S22 and so on, and S41 and so on.

So in fact the share of P1 will be S31 as well because P1 is also present in the third subset. And now it is easy to see that irrespective of which two parties get corrupt because my threshold $t=2$, those two shareholders learn absolutely no information about the secret s . So for instance if P1 and P2 gets corrupt they are under the control of the adversary then based on their shares that they learn due to their present in the subset t1 P1 and P2 fails to learn the secret s because the share S13 is missing for P1, P2.

In the same way with respect to the subset t2, P1 and P2 are missing the share as 2 3 and that is why the secret s is will not be known to P1, P2 and so on. So it does not matter which subset of t parties get corrupt based on the assures they fail to learn the actual secret s . So now you might be wondering that we now have a secret sharing scheme for any threshold t with respect to the value of n .


But it turns out that this scheme is inefficient because the number of subsets of size $t + 1$ is $n q t + 1$ which becomes an exponential quantity if t is approximately n over 2. That means dealers basically has to deal with exponential number of values here and same is the case for every shareholder. So this is an inefficient solution.

(Refer Slide Time: 15:53)

Shamir's (n, t) Secret-Sharing Scheme

A. Shamir: How to Share a Secret. Communication of ACM 22(11): 612-613 (1979)

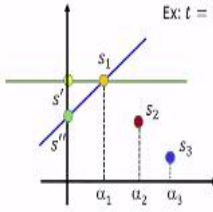
❑ One of the **simplest cryptographic constructions** that one can think of
(My personal favorite)



❑ Idea:

- ❖ Let the secret be the **constant term of a randomly chosen polynomial** $f(X)$ of degree- t
- ❖ Let the **shares be distinct points**, lying on $f(X)$

Ex: $t = 1$



- ❖ $(t + 1)$ **distinct values** of an unknown t -degree polynomial $f(X)$ are **sufficient** to uniquely reconstruct $f(X)$
- ❖ t **distinct values** of an unknown t -degree polynomial $f(X)$ are **not sufficient** to uniquely recover $f(X)$

We perform the above **computations over a finite field F** to achieve **security** and to avoid **working over an infinite domain**

And let us now discuss a very clever solution for n, t Secret-Sharing due to Adi Shamir. And this is one of the simplest cryptographic constructions that you can think of. This is my personal favorite. And this is based on simple arithmetic which you might have learned during your high school. So the idea here is, if you want to share a secret s then to share it you pick a random polynomial $f(x)$ of degree t such that the constant term of the polynomial is a secret which you want to share.

And let the shares be the distinct points at the values lying on that polynomial. So to demonstrate my point, imagine my threshold $t = 1$ and I have a secret s . I am the dealer. What I can do is, to share the secret s since my threshold $t = 1$, I pick a random straight line, it could be any straight line in the plane with the only restriction being that its constant term should be secret s which I want to share. And what I can now do is I can evaluate or I can compute the value of the straight line at some fixed publicly known distinct values.

Say at $x = \alpha_1$, at $x = \alpha_2$ and $x = \alpha_3$. And these are the shares for the first party, second party and third party respectively. That means the first party will have the share S_1 , the second party will have the share S_2 and the third party will have the share S_3 . It will be publicly known to everyone that the first party is obtaining the value of straight line the dealer has picked at $x = \alpha_1$.

So this alpha value they are publicly known and its everyone knows that a alpha 1 is associated with the first party, alpha 2 is associated with the second party and so on. So that is a Shamir's secret sharing scheme. Now let us try to prove that why intuitively it satisfies a requirement of n, t secret sharing. So it is easy to see that if $t + 1$ shareholders come together then they can uniquely reconstruct, tag the t degree polynomial which dealer has picked.

Because $t + 1$ distinct values on a unknown t degree polynomials suffice to uniquely reconstruct back that polynomial. So for example if $t = 1$ and say the first two shareholders is comes up with their share S_1 and S_2 then they can uniquely find the straight line passing through the points alpha 1, S_1 and alpha 2, S_2 by fitting a straight line equation. Once they obtain the straight line they can take the constant term of the straight line namely S to be the recovered secret.

On the other hand, the second fact that we can use for polynomial of degree t start, if you take any t shareholders who are the bad guys and they are trying to reconstruct back the dealer secret they will fail to do that because t distinct values does not suffice to uniquely recover back the unknown t degree polynomial $f(x)$ which is first picked by the dealer. More specifically, in this example since $t = 1$ say the first shareholder is corrupt.

Then from its viewpoint there could be infinite number of straight lines possible in the plane passing through his point alpha 1, S_1 . That means it could be the case that dealer might have picked this blue straight line given the first shareholder the share S_1 , in that case from the viewpoint of the first shareholder it could the case that S double prime is the secret or it might be the case that this red line is the straight line which dealer might have picked such that, that straight line when evaluated at alpha 1 would have given the share S_1 .

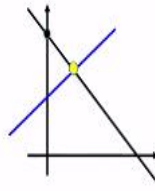
In that case the secret s prime could be the secret from the viewpoint of the first shareholder and so on. That means just based on t shares adversary will completely fail uniquely reconstruct back the dealer's original polynomial and hence the dealer's original secrets. That is the intuitive idea. Here, what it turns out that we have to perform all the above computations over some finite field to achieve security and to avoid working over an infinite domain.

(Refer Slide Time: 20:01)

Polynomials Over a Finite Field

- Let $(\mathbb{F}, +, \cdot)$ be a finite field
- **Definition:** a t -degree polynomial $f(X)$ over \mathbb{F} is of the form

$$f(X) = a_0 + a_1 \cdot X + \dots + a_t \cdot X^t$$
} $a_0, \dots, a_t \in \mathbb{F}$
- **Definition (root of a polynomial):** a value $x \in \mathbb{F}$ is called a **root** of $f(X)$, if $f(x) = 0$
- **Theorem (Abstract algebra):** a t -degree polynomial $f(X)$ over \mathbb{F} has **at most t roots**
- **Theorem (Abstract algebra):** two distinct t -degree polynomials $f(X), g(X)$ over \mathbb{F} **agree on at most t points**
- **Theorem (Abstract algebra):** Let $(x_1, y_1), \dots, (x_{t+1}, y_{t+1})$ be pairs of elements from \mathbb{F} , where x_1, \dots, x_{t+1} are distinct. Then there exists a **unique t -degree polynomial $f(X)$** over \mathbb{F} , such that $f(x_i) = y_i$, for $1 \leq i \leq t+1$



$$\delta_i(X) \triangleq \frac{(X - x_1)(X - x_2) \cdots (X - x_{i-1})(X - x_{i+1}) \cdots (X - x_{t+1})}{(x_i - x_1)(x_i - x_2) \cdots (x_i - x_{i-1})(x_i - x_{i+1}) \cdots (x_i - x_{t+1})}$$

$$f(X) \triangleq \delta_1(X) \cdot y_1 + \dots + \delta_{t+1}(X) \cdot y_{t+1}$$

$\delta_i(x_i) = 1 \quad \delta_i(x_1) = \delta_i(x_2) = \dots = \delta_i(x_{i-1}) = \delta_i(x_{i+1}) = \dots = \delta_i(x_{t+1}) = 0$

So let us try to understand first poly; some basic facts about polynomials over a finite field. So imagine I am given a finite field which has two operations $+$ and \cdot and when I say that I have picked the t degree polynomial $f(x)$ so the field I mean to say that I have picked $t + 1$ coefficients from the field and my polynomial $f(x)$ is defined like this. Next, let me define what we call as a root of a polynomial over a field.

So if I have a polynomial $f(x)$ over the field and if I pick a value x from the field then that value x from the field will be called as the root of that polynomial if the value of the polynomial at the value little x will be 0, is 0. I stress here that all these $+$ operations, \cdot operations that I am performing when I am defining the polynomial over field are the $+$ and the \cdot operation over the field. They are not the numeric $+$ and a numeric \cdot operation.

Now another well-known fact from abstract algebra which we can use here is the following. If you are given a t degree polynomial $f(x)$ over a field, then it can have utmost t roots. More specifically if I take the case of $t=1$ and if I have a straight defined over a field then there could be utmost one x value where I obtain that the value at the straight line or the straight line meets the y -axis here. And this is true for any t degree polynomial over a field.

And based on this theorem we can state that if you have got two distinct t degree polynomials say $f(x)$ polynomial and $g(x)$ polynomial over a field then they can have utmost t common

points. They cannot have $t + 1$ or more number of common points. So for instance if you have 2 distinct straight lines they can intersect at utmost one point. They cannot intersect at 2 points because if they intersect at two common points then the 2 straight lines are basically the same straight line and in general, this generalizes for any value of t .

Again I am not proving this theorem. These are some well-known results from abstract algebra. And a final result which I am going to use for define; for giving the description of summation is the Lagrangian interpolation formula here. So what this theorem basically says is if you given a pair of $t + 1$ x, y values from the field where the x components are distinct. Then there exists a unique degree polynomial say $f(x)$, such that this x, y values lie on the polynomial.

That means $f(x, y)$ value would have given you the corresponding y values. And to see how exactly we can compute this polynomial $f(x)$ satisfying this $p+1$ x, y values. Let me define an i th x polynomial as follows. So this i th polynomial is $\delta_i x$ polynomial, its degree is t because in the numerator I have $(())$ (23:06) terms of the form, variable x -some x_i . And the way I have defined is $\delta_i x$ polynomial is that, it ensures that δ_i of x_i is equal to 1.

Because if I substitute $x = x_i$ then numerator and denominator becomes the same. On the other hand, for every value of x_i different; for every value x_j different from x_i , this polynomial δ_i of that x_j becomes 0, that means this δ_i of x_1 will be 0, δ_i of x_2 will be 0. And in the same way δ_i of x_{t+1} will be 0 because if I substitute for instance $x = x_1$ here in this $\delta_i x$ polynomial then the numerator will become $x_1 - 1$ and hence 0.

In the same way if I substitute $x = x_2$ the numerator $x_2 - x_2$ will become 0 and so on. That means you can consider that this $\delta_i x$ polynomials are such that it survives at x equal to little x_i whereas it vanishes at all other x_i values. Now my goal is to find out that unknown polynomial $f(x)$ passing through the $t + 1$ given pairs of $x_i y_i$ values. And I can represent that unknown polynomials in terms of this $\delta_i x$ polynomials as follows.

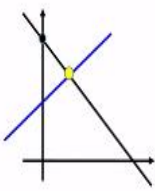
My $f(x)$ polynomial is δ_1 times; the first δ_1 polynomial times y_1 , the second δ_2 polynomial times y_2 and like that the $t + 1$ th δ_{t+1} polynomials times the $t + 1$ th y value.

(Refer Slide Time: 24:43)

Polynomials Over a Finite Field

- Let $(\mathbb{F}, +, \cdot)$ be a finite field
- **Definition:** a t -degree polynomial $f(X)$ over \mathbb{F} is of the form

$$f(X) = a_0 + a_1 \cdot X + \dots + a_t \cdot X^t$$
} $a_0, \dots, a_t \in \mathbb{F}$
- **Definition (root of a polynomial):** a value $x \in \mathbb{F}$ is called a **root** of $f(X)$, if $f(x) = 0$
- **Theorem (Abstract algebra):** a t -degree polynomial $f(X)$ over \mathbb{F} has **at most t roots**
- **Theorem (Abstract algebra):** two distinct t -degree polynomials $f(X), g(X)$ over \mathbb{F} **agree on at most t points**
- **Theorem (Abstract algebra):** Let $(x_1, y_1), \dots, (x_{t+1}, y_{t+1})$ be pairs of elements from \mathbb{F} , where x_1, \dots, x_{t+1} are **distinct**. Then there exists a **unique t -degree polynomial** $f(X)$ over \mathbb{F} , such that $f(x_i) = y_i$, for $1 \leq i \leq t+1$



$$\delta_i(X) = \frac{(X - x_1)(X - x_2) \cdots (X - x_{i-1})(X - x_{i+1}) \cdots (X - x_{t+1})}{(x_i - x_1)(x_i - x_2) \cdots (x_i - x_{i-1})(x_i - x_{i+1}) \cdots (x_i - x_{t+1})}$$

$$f(X) \cong \delta_1(X) \cdot y_1 + \dots + \delta_{t+1}(X) \cdot y_{t+1}$$

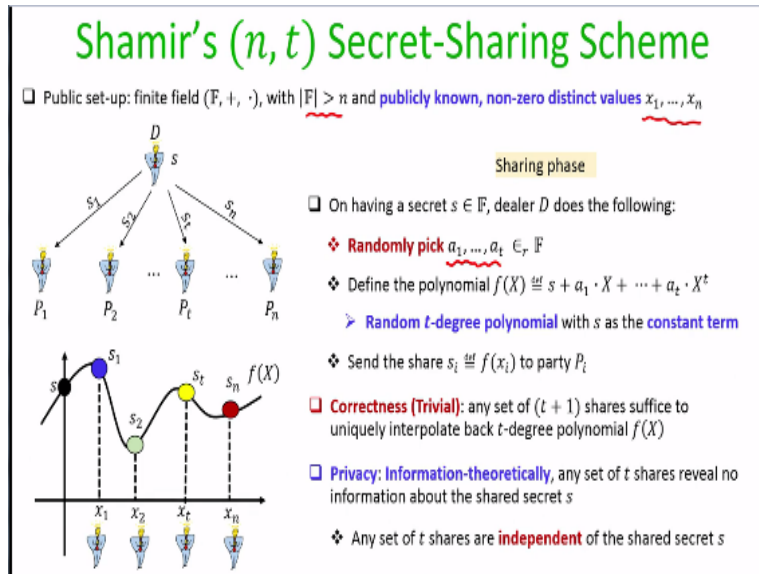
$$\delta_i(x_i) = 1 \quad \delta_i(x_1) = \delta_i(x_2) = \dots = \delta_i(x_{i-1}) = \delta_i(x_{i+1}) = \dots = \delta_i(x_{t+1}) = 0$$

And it is easy to see that indeed $f(x_1)$ will be y_1 , because for $f(x_1)$; because of $x = x_1$ my delta 1 polynomial will survive and give the value 1 and 1 multiplied by y_1 will be y_1 whereas all the other delta polynomials will vanish off. In the same way for $x = x_2$ all my delta polynomials will vanish except the second delta polynomial which will give the value 1 and 1 multiplied by y_2 will give me y_2 which satisfies my condition.

So that is the unique t degree polynomial $f(x)$ which you can find out passing through the given $t + 1$ pairs of x and y values where the x components are distinct. Now you might be wondering that why the x components have to be distinct because they have to be distinct to ensure that each of this delta polynomial have a denominator which is non-zero and if denominator is non-zero then basically this numerator divided by denominator should be interpreted as if this numerator is multiplied by the multiplicative inverse of my denominator.

Because I am doing the division here and this division should be interpreted as multiplying the numerator with the multiplicative inverse of the denominator. And the multiplicative inverse of the denominator will exist only if my denominator is non-zero.

(Refer Slide Time: 26:08)



So now let us go to the description of Shamir's Secret Sharing the intuition I have already discussed. So as part of public setup we will be given some finite field where the size of the field be at least n the number of shareholders. And associated with the shareholders will be n publicly known non-zero distinct x values namely x_1 to x_n which are values from the finite field. The sharing algorithm of Shamir secret sharing is as follows.

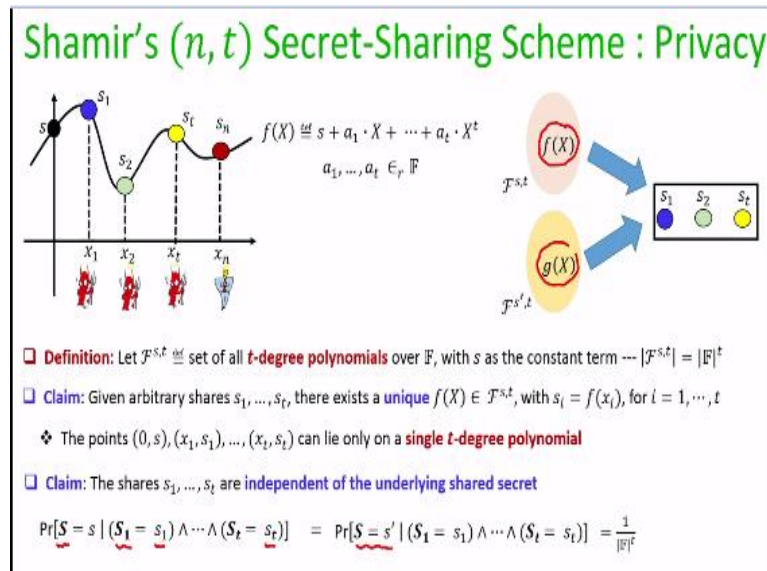
So if the dealer has a secret value little s which he wants to share then it picks a random polynomial over the field by choosing t random elements as the coefficient from the field, such that the constant term of the polynomial which I call as a $f(x)$ is the secret s which dealer want to share. And now the i th shareholder namely the party P_i gets share S_i where S_i is nothing but the value of this polynomial picked by the dealer at $x = x_i$. And that is the share for the i th party.

The correctness of this secret sharing is trivial that means if imagine out of this n shareholders any subset of $t + 1$ shareholders come together by exchanging their shares then they can uniquely re-interpolate back the t degree polynomial $f(x)$ using the Lagrangian interpolation formula that we have discussed in the earlier slide.

And to prove the privacy we are going to prove that if we take any set of t shareholders among this n shareholders then their shares are independent of the underlying share secret s , because

intuitively this is the fact that the coefficients of polynomial which are picked by dealer for sharing the secret are uniformly randomly chosen from the underlying field.

(Refer Slide Time: 27:55)



So let us make it more regress. So let me define a set $\mathcal{F}^{s,t}$ to denote the set of all t degree polynomials selected from the finite field whose constant; constant term is the secret s . So it turns out that the number of such polynomial of degree t whose constant term is the secret s is nothing but the size of the field raise to the power t . Because the number of such polynomials is nothing but the number of candidate a_1 values, the number of candidate a_2 values and the number of candidate a_t values.

Now since each of these coefficients could be randomly chosen from the field I have the number of candidate a_1 values is nothing but a number of elements in the field. In the same way any value from field could have taken the candidate a_2 values and hence the number of candidate a_2 values is nothing but the number of values from the field and the same is true for the t -th coefficient in the polynomial.

So since each of the coefficients could be any of the values from the finite field it turns out that I have size of the field raise to the power t number of polynomials of degree t whose constant term could be the secret s . That is the cardinality of this set $\mathcal{F}^{s,t}$ to the power s, t . Now I want to prove the

secrecy property of Shamir secret sharing. So without loss of generality assume that first t shareholders are corrupt, that means they have seen the shares S_1, S_2, \dots, S_t .

And they know that these shares are nothing but the value of some unknown $f(x)$ polynomials selected by the dealer evaluated at $x = x_1, x = x_2$ and $x = x_t$, right. So x_1, x_2, x_t , they are publicly known. What is not known to these shareholders that is corruptive shareholder is the exact value of the $f(x)$ polynomial selected by the dealer. And my goal is to show that these t shareholders are this t shares are independent of the actual secret picked by the dealer.

So to prove that I claim here that if I take this t shares or if I take any arbitrary set of t shares S_1 to S_t corresponding to those t shares there exists a unique polynomial $f(x)$ belonging to the set f to the power s, t such that the shares S_1 to S_t would have lied on that polynomial and this is because of the fact that I cannot have two different polynomials $f_1(x), f_2(x)$ from the same set f over f superscript s, t which would have given me the same set of t shares S_1 to S_t .

Because if I have two such distinct polynomials $f_1(x)$ and $f_2(x)$ such that $f_1(x)$ evaluated at $x = x_t$ would have given me S_1 to S_t and for the another polynomial $f_2(x)$ as well it holds that the $f_2(x)$ polynomial evaluated at $x = x_1$ to $x = x_t$ would have given me the value S_1 to S_t . Then it implies that these two different polynomials $f_1(x)$ and $f_2(x)$ have same values at $t + 1$ distance with x values because remember the constant term of this $f_1(x)$ exponential polynomials and $f_2(x)$ polynomials is nothing but the secret s .

That means the point or the value $0, s$ also satisfies both these two polynomials along with the remaining t pairs x_i, s_i which is simply not possible because we have stated one of the fact that two distinct t -degree polynomials can have utmost t common values. So that means there could be only one single $f(x)$ polynomial from the set f superscript s, t which would have given these shares S_1 to S_t .

Now based on all distinct I claimed that the shares S_1 to S_t which are seen by the first corrupt t shareholders or any set of t shareholders. In this example I am considering the case where the first t shareholders are corrupt. So I claim that shares S_1 to S_t they are independent of the actual

secret shared by the dealer and to prove this claim, let us consider another potential candidate secrets S' .

And try to argue with how much probability the shares S_1 to S_t could be the shares of the secret s and with how much probability the same set of t shares could be the shares for S' . And I will show that with same probability this set of shares S_1 to S_t could be the shares of the secret s as well as for the secret S' and that will prove that these set of t shares S_1 to S_t they are independent of the actual secret s .

So for doing that let me introduce $t + 1$ random variables here. So I introduce the random variable S to denote the actual secret which the dealer might have used or picked for sharing. And I denote t additional random variables S_1, S_2, \dots, S_t and bold notation to denote the value of the t shares which the first t shareholders who are corrupt in this analysis are going to obtain and I want to compute the probability that what is the probability that dealer has used the secret s for ensuring given that first t shareholders are seeing the shares s_1, s_2 and s_t .

And I also want to compute what is the probability that dealer has shared the secret s' given that the first t shareholders are seeing the shares s_1, s_2 and s_t . It turns out that both these probabilities are same namely $1/q^t$ or the size of field to the power t . This is because, if dealer the probability that the secret shared by the dealer is s given that first t shares are S_1 to S_t implies there exists unique polynomial say $f(x)$ which dealer has used while sharing or invoking the Shamir secret sharing.

Whereas the probability that dealer has used or shared the secret s' and that resulted in the shares S_1 to S_t implies that dealers has used, the unique polynomial $g(x)$ from the set f to the power s' , t during the invocation of Shamir's secret sharing. But if you remember the step of Shamir secret sharing the polynomial that was used by the dealer is randomly chosen either from the set f to the power s , 2 or f to the power s' , t .

That means the probability that indeed $f(x)$ the polynomial picked by the dealer and S_1, S_t are the resultant share is $1/q^t$ and the same holds with respect to

the secret S prime. That means from the viewpoint of the t shareholders in this case the first t shareholders with equal probability the share S_1 to S_t could have come from the secret s or they could have come from the secret s prime.

And that means the adversary is completely clear whether secret s or whether it is a secret s prime which is shared. And that formally proves that Shamir secret sharing satisfies the requirement of n, t secret sharing. So that brings me to the end of this lecture. Just to summarize. In this lecture we introduced the problem of secret sharing n, t secret sharing and we saw three constructions.

We saw the construction of additive secret sharing where the threshold is $n - 1$. And then using this $n, n - 1$ secret sharing exponential number of times we saw a solution for any n, t secret sharing which we call as replicated secret sharing and finally we saw clever construction of n, t secret sharing which is a poly time solution for any value of n and t due to Shamir.