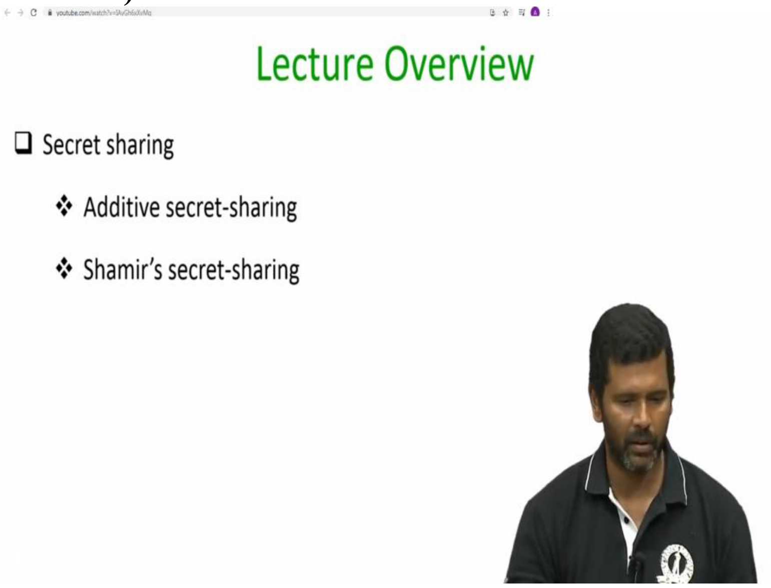
Discrete Mathematics
Prof. Ashish Choudhury
International Institute of Information Technology - Bangalore

Lecture - 71
Foundations of Cryptography

(Refer Slide Time: 00:23)



Lecture Overview

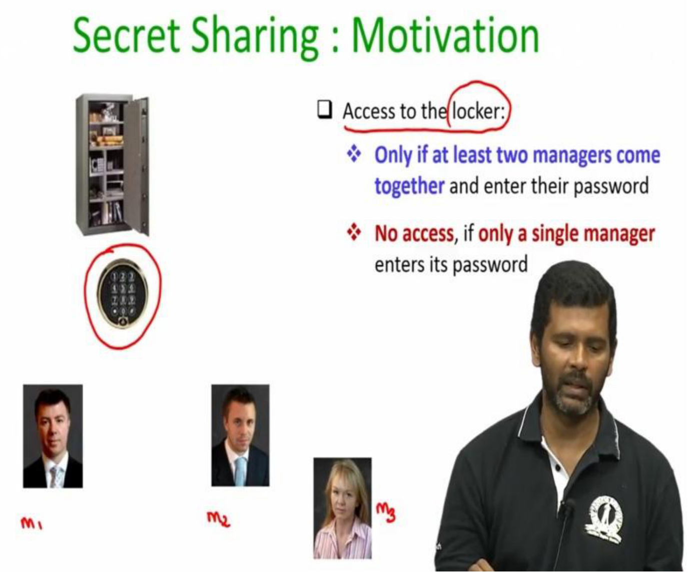
❑ Secret sharing

❖ Additive secret-sharing

❖ Shamir's secret-sharing

Hello everyone, welcome to this lecture. The plan for this lecture is as follows. So, in this lecture we will see a very nice cryptographic application based on the concepts related to finite fields, namely secret sharing. So, earlier my plan was to cover both additive secret-sharing, as well as Shamir's secret-sharing. We will see depending upon the availability of time whether we are going to cover both of these topics or not.

(Refer Slide Time: 00:50)



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

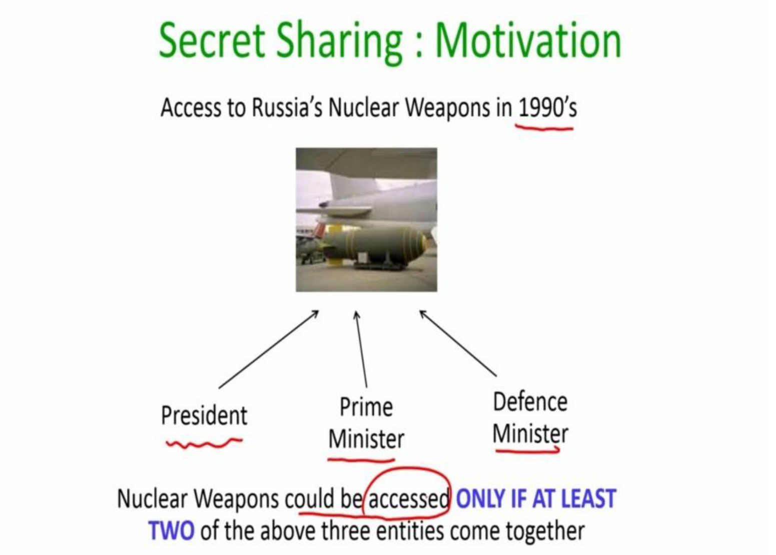
So, let us start with the problem of secret sharing what exactly it is motivation real world application. So, for that imagine a banking application and the way a locker is operated. So, I do not know whether you have a locker account in a bank or not, but I do have and the way locker account is maintained or operated in the bank is as follows. Whenever you want to open or get access to your locker, you have to go along with your key. And apart from your key there is another key which is held by the manager.

And only when both I enter my key as well as the manager enters the key the locker can be opened, that means in my absence, the manager cannot open the locker and the same way I alone cannot go and open the locker myself just using my copy of the key. So now, in the same way, here in this particular example, I consider a scenario where you can imagine that the locker can be opened by pressing the key word or the key phrase, whatever you can call and the key phrase is not available with a single person.

So I imagine here that we have 3 managers: manager 1, manager 2 and manager 3. And the way this system works here is the following only when at least 2 of the managers come together and enter their respective passwords, the locker can be opened. But if only a single manager goes and tries to open the locker by entering his or her password, the locker should not be opened. So for instance, if m_1 just simply goes and try to enter, the locker should not open.

Similarly, if m_2 goes and try to open the locker alone, he should fail. If the manager m_3 alone tries to open the locker, she should fail and so on. But say if the first and second manager goes together and enter their respective passwords, then the locker should be opened. In the same way the first and third manager goes together and enter their respective passwords the locker should be opened.

(Refer Slide Time: 03:12)



Secret Sharing : Motivation

Access to Russia's Nuclear Weapons in 1990's

President

Prime Minister

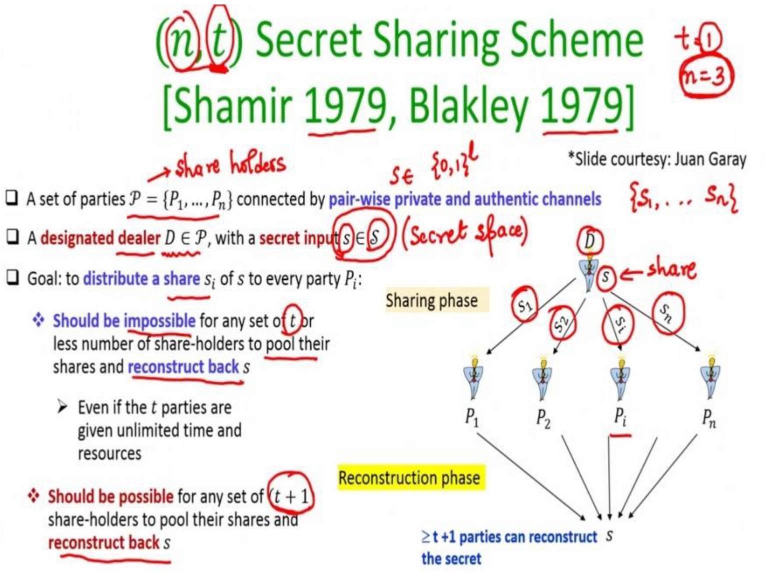
Defence Minister

Nuclear Weapons could be accessed **ONLY IF AT LEAST TWO** of the above three entities come together

Another interesting motivation for secret sharing problem is the following. So it is believed that in the 1990’s, the access to Russia's nuclear weapon was done in the following fashion. So the password or the credential for launching the nuclear weapon was shared among 3 top entities of the country, namely the President, Prime Minister and the Defence minister. And it was shared in such a way that nuclear weapon could be launched or could be accessed only if at least 2 of the above 3 entities come together and enter their respective credentials or passwords. But if only 1 entity comes and try to launch the nuclear weapon, then that entity should fail.

So, in some sense, you can imagine that this is the kind of system gives you more security more robustness in the sense that if one of the 3 entities say either the President or the Prime minister or the Defence minister gets compromised and leaks the password, then an enemy country can launch the nuclear weapon. But if we operate our nuclear weapon in this kind of system, then in order that the system gets compromised, the enemy country has to compromise at least 2 entities. That means, if it just compromises one of the entities say either the President, or the Prime minister, or the Defence minister, then still no harm will be caused only when 2 of the entities are compromised the harm can be caused. And the system in that sense, the system is more secure.

(Refer Slide Time: 04:57)



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

$t = 1$

$n = 3$

*Slide courtesy: Juan Garay

→ share holders

$s \in \{0,1\}^l$

- □ A set of parties $\mathcal{P} = \{P_1, \dots, P_n\}$ connected by pair-wise private and authentic channels $\{s_1, \dots s_n\}$
- □ A designated dealer $D \in \mathcal{P}$, with a secret input $s \in \mathcal{S}$ (Secret space)
- □ 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
    - ➢ Even if the t parties are given unlimited time and resources
  - ❖ Should be possible for any set of $(t + 1)$ share-holders to pool their shares and reconstruct back s

So, now, let us abstract both the examples that we had seen by this general problem of what we call as (n, t) secret sharing and this problem or this primitive was independently introduced by Turing award winner, Adi Shamir in 1979 and by Blakley in the same year. So, I am taking this slide from Juan Garay. So, the problem definition is the following you have a set of parties P_1 to P_n. So, n is a given parameter everyone knows the identity of the parties, we call this parties as shareholders.

And among these n parties, there is a designated entity or a special entity whom we call as a dealer denoted by D. And dealer has some private input some secret input, let us denote it by s which is a bit string or it could be any abstract value, but you can always assume that it is a bit string and this value s belongs to a bigger set namely this set $\mathcal{S}$ which I call as the secret space.

So, the knowledge of the secret space is publicly known, namely, it denotes the set of all possible secrets which dealer can have. What exactly is the value of the secret from the secret space which dealer has no one will be knowing, depending upon what kind of prior information we may have about the secret s. But it will be a public knowledge that whatever is the input of the dealer, it is from a bigger space namely this set fancy $\mathcal{S}$ which we call as a secret space.

So, for instance, a simple example of the secret space could be the set of all possible bit strings of length say l bits. That is the case then everyone will be knowing that dealer is going to invoke or use this primitive with some value s belonging to the set of all possible l bit strings. And everyone will know the identity of the dealer. Now, the goal is the following: we want a mechanism according to which dealer should share the secret s among the shareholders. By that I mean we need a mechanism according to which dealer should compute n pieces of information, denote them as s_1 to s_n and ith piece of information namely s_i will be given to the ith party. So, we say that s_i denotes a share of the secret s for the party p_i . That is a requirement, but then if that is just my requirement it is very easy to solve this problem. Dealer can do some kind of distribution, give some piece of information regarding s to the first shareholder some piece of information regarding the secret to the second shareholder and so on.

So, that is why to make the problem interesting and to model what exactly we had seen in the previous 2 examples, we need the following requirements from this sharing mechanism. We require that it should be impossible for any set of t or less number of shareholders to pool their shares and reconstruct back the secret s. So again, like n, t is also some given parameter. So, I need my sharing mechanism to be such that the vector of n shares which I denote as s_1 to s_n, they should be such that if any t shares from this vector are taken, then it should be impossible to reconstruct back the secret s, irrespective of how much time I give to you how much resource I give it to you. And this should hold even if the description of the sharing algorithm, the description of the secret space, everything is publicly known, that is important. I am not assuming here that the sharing mechanism, the algorithm by which the shares are computed they are hidden.

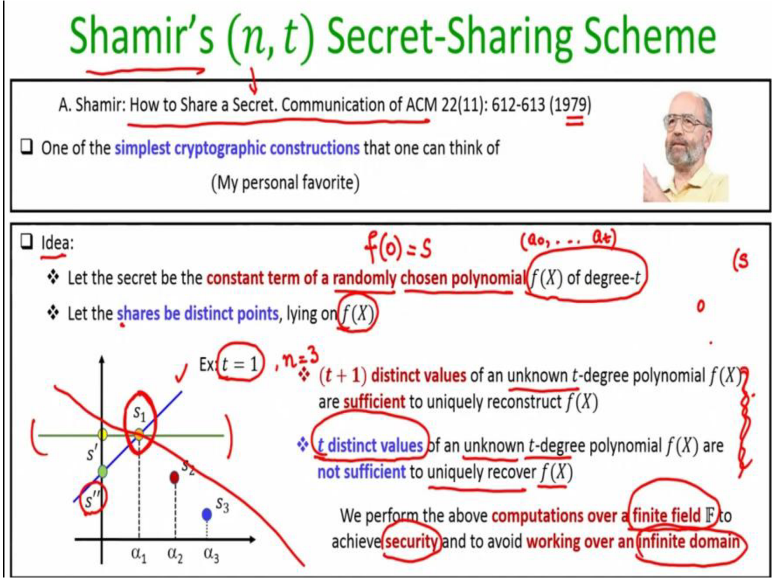
It is known only to dealer no, that is not the case, because dealer could be any party, the end shareholders could be any party I cannot afford to design an algorithm which is known or which is secret and available only with the dealer so, that is my first requirement. The second requirement is that if t + 1 or more shares are available from this vector of n shares, then it should be possible to efficiently and uniquely reconstruct back the secret s.

So, you can imagine here that t is kind of acting as a threshold here. Any number of shares up to t or less will fail to give you back the secret. Any number of shares which are t+1 or more in numbers should give you back the secret s unambiguously. So, for instance, the previous 2 examples that we had seen, there my threshold t was 1. And my n was 3. So, if you take the banking example, there were 3 managers.

So, there are we want a mechanism where the master password should be shared among 3 managers, namely 3 shareholders. So, that if any one of the 3 managers try to access the secret, the manager should fail. In the same way in the nuclear weapon example, we had 3 entities namely the President, the Vice President and the Prime Minister. So, they are my shareholders and my t was 1.

I do not want any one of those 3 entities to be able to access the nuclear weapon, but only when 2 or more numbers of entities come together they should be able to access the nuclear weapons. That was my requirement. So, this is your problem of (n, t) secret sharing. Now, the question is how exactly we can solve this?

(Refer Slide Time: 12:07)



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: $t = 1$, $n=3$

❖ $(t+1)$ 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 $\mathbb{F}$ to achieve security and to avoid working over an infinite domain

$f(0) = s$

$(a_0, \ldots a_t)$

So, let me discuss this (n, t) secret sharing scheme due to Shamir. So, this is also called as Shamir's (n, t) secret sharing scheme. He gave a very nice and very elegant solution for solving the (n, t) secret sharing problem. Independently Blakley gave another solution for solving the (n, t) secret sharing problem. But, since we want to see the solution based on applications of or concepts learned for finite field, I am going to demonstrate Shamir's secret sharing scheme for you.

So, this algorithm was published way back in '79, in this very short paper. But this is one of the highly cited research papers in cryptography and this is a very simple and elegant construction. In fact, this is my personal favourite, when I will explain you the solution you cannot even imagine that how can the solution be so simple and at the same time elegant. So, the idea behind Shamir's secret sharing scheme is the following so, imagine dealer as this secret s.

So, to share the secret s what dealer can do is the following: it can pick a polynomial of degree t say in variable x. And it will be a random polynomial, which will be chosen by the dealer. When I say a random polynomial, by that I mean only dealer will be knowing the coefficients of the polynomial f(X). So, say for instance, if f(X) is of the form say $a_0 + a_1$ times X, and like that, the t + 1 th coefficient is $a_t \cdot X^t$.

So, when I say randomly choosing the polynomial by that I mean that the coefficients a_0 to a_t are known only to the dealer and it is not known to any shareholder. This does not violate the assumption that the algorithm description is publicly known. The algorithm is publicly known. What is the algorithm? The algorithm here is choosing a polynomial randomly that is the process, that is the step, that step is publicly known. That means, the shareholders will be knowing the actions of the dealer.

Actions of the dealer, by that I mean they will be knowing that dealer is going to pick a polynomial f(X). But what exactly is the polynomial, the value of the polynomial that will not be known to the shareholders. That will be random, that will be known only to the dealer. Because if everything is known in the public domain regarding what are the values which are picked by the dealer and so on, then how at the first place you can hope to solve this problem.

There has to be some component of randomness in this whole solution and that is incorporated by saying that dealer picks the polynomial f(X) randomly. Now, it is a random polynomial except that the constant term of this polynomial is the secret which dealer wants to share, that means, the coefficients a_1, a_2 and a_t they are randomly chosen but a_0 is not randomly chosen, but rather a_0 is actually the secret s which dealer wants to share.

In some sense, you can also imagine this as if the polynomial f(X) is a random polynomial, except that the polynomial when evaluated at x = 0 will give you the value s because if I pick a polynomial of the form f(X) where the constant term or the coefficient is the secret s. And then you have the remaining coefficients, then that polynomial when I evaluate at x = 0 will give me the value s.

And then, once dealer has chosen this polynomial f(X) randomly, the shares are nothing but distinct points on the polynomial chosen by the dealer. So, let me demonstrate what I am trying to say here. And for demonstration, I assume that I need a secret sharing mechanism where $t = 1$. And imagine dealer's secret is s. So, a polynomial of degree $t = 1$ is nothing but a straight line. So, what I am saying here is that dealer in his mind is picking a random straight line.

And that straight line has the property that, when evaluated at $x = 0$ will give you the value s, namely his secret. And now the shares for the respective shareholders are the following: we imagine here that α_1, α_2, α_3. So by the way, I am assuming here $n = 3$. So, we need to compute 3 shares and for computing 3 shares for the 3 shareholders, we assume here that α_1, α_2 and α_3 they are some publicly known distinct non 0 values.

And the share for the first shareholder is nothing but the value of the straight line at $x = \alpha_1$. Namely, we compute the point (α_1, s_1), we compute a point (α_2, s_2). And we compute a point (α_3, s_3), which is equivalent to saying that I am evaluating the straight line at $x = \alpha_1$, $x = \alpha_2$ and $x = \alpha_3$ and getting the values s_1, s_2 and s_3. And s_1 will be the share given to the first shareholder. s_2 will be the share given to the second shareholder. And s_3 will be the share given to the third shareholder. Again here, everyone will know the value of α_1, α_2 and α_3 that is not hidden. And everyone will know that the first shareholder is getting the value of dealer’s straight line at $x = \alpha_1$. But what exactly was the straight line that is not known, that is important. Similarly, everyone will know that second shareholder is getting the value of dealer’s straight line at $x = \alpha_2$.

And everyone will know that the third shareholder is getting the value of dealer's straight line at $x = \alpha_3$, because the steps of the algorithm are publicly known. Now, let us see here why this constitutes a valid (n, t) secret sharing scheme. So, remember I am considering the case where $n = 3$ and $t = 1$. The reason it constitutes (n, t) secret sharing scheme is because of the following 2 facts.

Since the dealer's polynomial f(X) which was known only to the dealer is unknown and its degree is t, it follows from fundamental properties of polynomials that if I give you $t + 1$ or more number of distinct values of the polynomial, then you will be able to uniquely reconstruct back your polynomial. So, for instance since t = 1 here and the straight line which was chosen by the dealer is not known to you.

What I am saying is the following: if I give you 2 distinct points on the straight line which dealer has chosen, then you will be able to uniquely get back the straight line which was chosen by the dealer. And if you can get back the straight line which was chosen by the dealer, well, you will be knowing the secret which dealer has shared because the constant term of that straight line is nothing but the dealer's secret. So, for instance, if I give you the first 2 shares and the first 2 shares are nothing but (α_1, s_1) and (α_2, s_2).

Now, using these 2 points, you will be able to get back this straight line uniquely. And once you get back this straight line, you can get back the secret s. In the same way if I give you say the second share and the third share. You will be able to get back the dealer's straight line uniquely and hence dealer's secret. So, that is the first observation. So, that shows that t + 1 or more number of shares will indeed give you back dealer's secret uniquely that satisfies one of the properties of (n, t) secret sharing.

Now, the second observation here is that if instead of t + 1 shares, I give you only t shares and t shares here are nothing but in this context, they constitute t distinct values of an unknown polynomial whose degree was t. So, the second observation here is that if I give you t distinct values on an unknown t degree polynomial, then you cannot uniquely recover back the polynomial f(X).

So, what does that mean here in the context of this example, where t = 1, it means the following. Suppose, the first shareholder who has the share s_1 and of course, it knows α_1. The question here is it possible for the first shareholder to get back the dealer's secret? Well, no, because it is possible that through the point (α_1, s_1), this blue straight line is actually the line which dealer was which dealer has selected that means, it could be possible that dealer has actually selected this blue straight line for sharing.

And this blue straight line when evaluated at α_1 gives you the value s_1 for namely, the point α_1, s_1 lies on this straight line that could be the case. If that could be the case then the secret would have been s'' where, s'' would be the constant term of this polynomial. Or it could be equally likely the case that dealer has actually used this straight line which also pass through

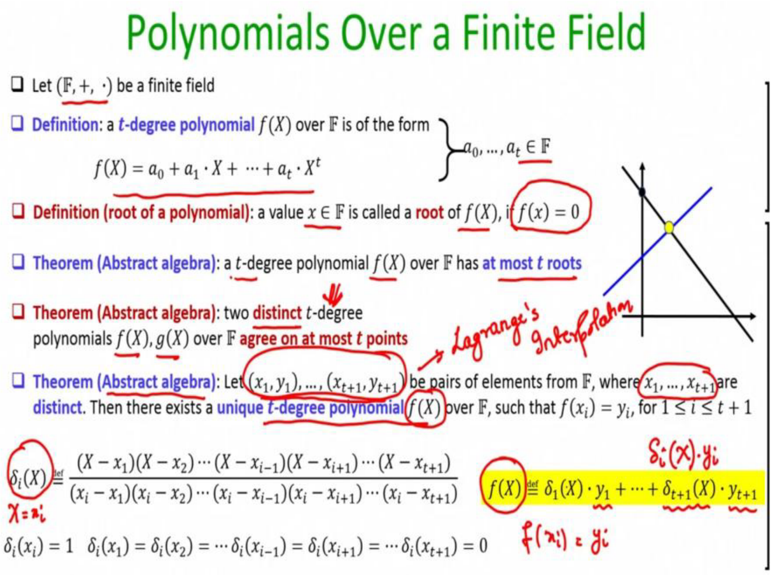
(α_1, s_1) or it could be the case that dealer has used say another straight line which also passes through (α_1 , s_1) and so on.

So, that means, just using the first share s_1, it is simply not possible to exactly identify the straight line which dealer has used and hence it could be any straight line and hence it could be any secret which dealer would have shared. So, that means you do not have sufficient information if you are just given t shares to uniquely reconstruct back the dealer's unknown polynomial.

And hence you cannot uniquely get back dealer's secret. That is the intuitive idea here, a very basic fundamental fact of polynomials of t degree namely, t + 1 or more points are sufficient to get back the polynomial; t or less number of points are not sufficient to uniquely get back the polynomial. Now, in the Shamir secret sharing scheme we perform or we use the above idea where all the computations are performed over a finite field, namely my secret space will be a finite field, my polynomials will be selected over a finite field and my shares also will be elements of a finite field and this is done because of the following 2 reasons.

We need to maintain security, intuitively what do I mean here is that if instead of picking the polynomial over a field, we pick polynomial over integers namely if my coefficients of the polynomial are integers, then based on the magnitude of the shares I may end up revealing some information about the possible range of the secret. I may not be able to leak the exact value of the secret, but I may be leaking some information regarding the possible range of the secret and also if I pick my coefficients of the polynomial and the shares from the integers then actually I will be working over an infinite domain, which I would not like to do. So, that is why everything is embedded to a finite field.

(Refer Slide Time: 26:34)



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 $f(X) = a_0 + a_1 \cdot X + \cdots + a_t \cdot X^t$ } $a_0, \ldots, a_t \in \mathbb{F}$
- ❑ **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 $(x_1, y_1), \ldots, (x_{t+1}, y_{t+1})$ be pairs of elements from $\mathbb{F}$, where $x_1, \ldots, x_{t+1}$ are **distinct**. Then there exists a **unique t-degree polynomial** $f(X)$ over $\mathbb{F}$, such that $f(x_i) = y_i$, for $1 \leq i \leq t+1$

→ Lagrange's Interpolation

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

$X = x_i$

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

$\delta_i(X) \cdot y_i$

$$f(X) \overset{\text{def}}{=} \delta_1(X) \cdot y_1 + \cdots + \delta_{t+1}(X) \cdot y_{t+1}$$

$f(x_i) = y_i$

So, to understand the actual Shamir’s secret sharing protocol, let us again recap the concepts regarding polynomials over a finite field. So, if you are given an abstract field which is finite with an abstract plus and dot operation, then a polynomial over the field is exactly a polynomial over the integers where the difference is that now, all the coefficients are from the field and all the plus and dot operations are your field operations.

A value x from the field will be called a root of this polynomial f(X), if the polynomial f when evaluated at x gives you the additive identity 0 or the 0 element of the field, then we have actually shown in our discussion on abstract algebra that you take any polynomial of degree t then it can have at most t roots. And using this theorem, we can actually show that you take 2 different polynomials f(X) and g(X) over the field then they can agree on at most t points that means, you can have at most t common points lying on both f(X) as well as g(X).

It is like saying the following you take 2 straight lines, there can be at most 1 point which is lying both on the first straight line as well as on the second straight line, you cannot have 2 points which are common and lying on both the first straight line as well as on the second straight line because if that is the case, that means the 2 straight lines are the same straight line at the first place.

So, that is why we can extend that idea in the context of polynomials over field as well and then conclude that if you take 2 distinct t degree polynomials, they can agree on at most t points. Now, another interesting result from the abstract algebra is the following: if I give you t + 1 number of (x, y) values where the x components are distinct, then you can always find a unique t degree polynomial over the field such that these (x, y) values constitute distinct points on that f polynomial.

So, this is often called as Lagrange’s interpolation theorem. So, basically the process by which we can compute this f polynomial is called as the Lagrange’s interpolation. And here is how we can get back this unique polynomial f(X). I define several t degree polynomials. So, the ith t degree polynomial in X is called as the $\delta_i(X)$ polynomial and it will be of this form. So, why it will be of degree t because in the numerator you have t factors of the form X minus some value and in the denominator you have the product of several differences.

So, since this whole polynomial is over a field do not interpret this division as your numerical division or integer division, the interpretation of this division is the following. In the denominator, I have the terms $(x_i - x_1)$ $(x_i - x_2)$ $(x_i - x_{i-1})$ and so on all these are elements from the field and if I take the differences here they will be elements of the field. So, in the denominator I have the product of several field elements.

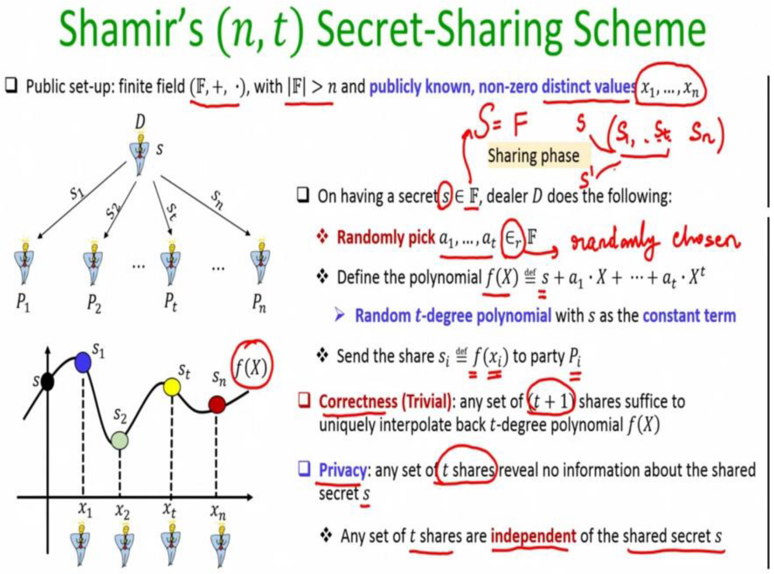
So, let the final result of the product of the elements in the denominator is A then the interpretation of this $\delta_i(X)$ polynomial is that instead of saying I divide the numerator by A I should interpret it as if the numerator is multiplied with A^{-1} where A^{-1} is the multiplicative inverse of the element A because I am performing all the operations over the field. So, in field I have just a plus operation and a dot operation. So, division should be interpreted as if I am multiplying with the multiplicative inverse.

So, the property of this $\delta_i(X)$ polynomial is the following. If I substitute $X = x_i$, then I get the value 1 because if I substitute $X = x_i$, then both the numerator and the denominator becomes same. Whereas, if I substitute X to be any other value of x say x_1, x_2 and so on, then I get a value 0 because one of the factors in the numerator will become 0 and 0 multiplied with any element will give me 0.

So, now, the unknown f(X) polynomial which I can recover from this t + 1 distinct (x, y) pairs is the following: it is the product of these t + 1 δ(X) polynomials multiplied with the corresponding y values. So, the first δ polynomial multiplied with y_1, second δ polynomial multiplied with y_2 and the t + 1 th δ polynomial multiplied with the y_{t+1} value.

And you can check easily here that you take this polynomial f(X) and evaluate it at x_i. You will get the value y_i because when you evaluate this f(X) polynomial at x_i, then only the $\delta_i(X)$ polynomial will survive and give 1 and that will be multiplied with y_i. So, you will get y_i and all other remaining $\delta_i(X)$ polynomials will vanish. That is the idea of Lagrange's interpolation.

(Refer Slide Time: 32:56)



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

❑ Public set-up: finite field $(\mathbb{F}, +, \cdot)$, with $|\mathbb{F}| > n$ and publicly known, non-zero distinct values $x_1, \ldots, x_n$

$S = F$ s $(s_1, . s_t \; s_n)$ s'

Sharing phase

❑ On having a secret $s \in \mathbb{F}$, dealer D does the following:

- ❖ Randomly pick $a_1, \ldots, a_t \in_r \mathbb{F}$ → randomly chosen
- ❖ Define the polynomial $f(X) \overset{\text{def}}{=} s + a_1 \cdot X + \cdots + a_t \cdot X^t$
  - ➢ Random t-degree polynomial with s as the constant term
- ❖ Send the share $s_i \overset{\text{def}}{=} f(x_i)$ to party P_i

❑ Correctness (Trivial): any set of $(t + 1)$ shares suffice to uniquely interpolate back t-degree polynomial $f(X)$

❑ Privacy: any set of t shares reveal no information about the shared secret s

- ❖ Any set of t shares are independent of the shared secret s

So, now, based on this concept from the finite field, here is the actual Shamir's secret sharing scheme. So, the setup here will be the description of a finite field. We require the size of the field to be at least n namely the number of shareholders and there will be n distinct x values from the field which are non 0 and which will be publicly available. To share a secret s which is an element from the field which ensures that my secret space S here is actually the finite field.

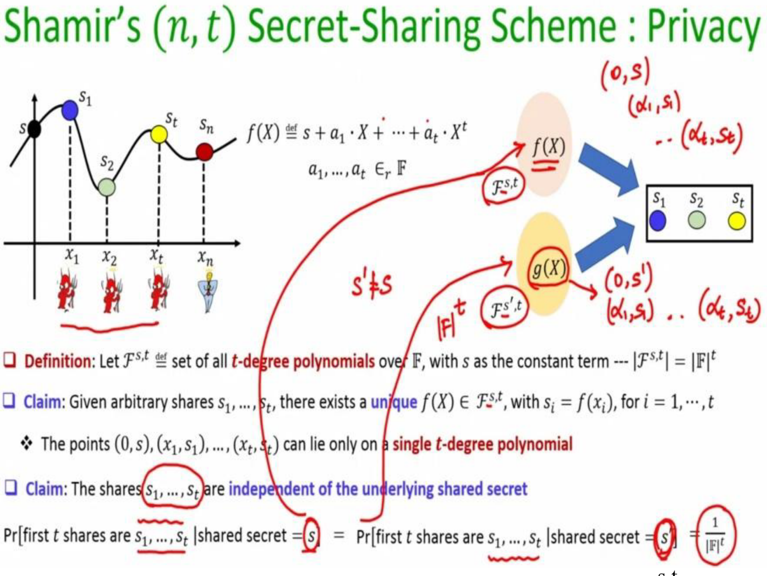
So, if there are multiple elements from the field which dealer wants to share, he has to invoke this protocol multiple times. But imagine he has only 1 element from the field which it wants to share, and to do that dealer is going to do the following. It is going to pick t coefficients for the sharing polynomial randomly from the fields. So, this notation ($\in_r$) belongs to subscript r denotes that the polynomial is randomly chosen. So, the polynomial f(X) is randomly chosen except at its constant term is the secret which dealer wants to share.

And now, the share for the ith party is the evaluation of this polynomial at x = x_i, of course, all operations done over a finite field. Now, let us see whether this scheme satisfies the 2 requirements of secret sharing. The first requirement is that, if you take any t + 1 shares, you should be able to get back the original secret and this comes from our Lagrange's interpolation because t + 1 shares are nothing but t + 1 distinct evaluations of the unknown t degree polynomial.

Why they are distinct evaluations because we have chosen our x values namely the values at which the polynomial is evaluated to be distinct. Whereas for privacy, namely, we have to show that you take any t shares, it leaks no information about the underlying secret s, even if the t shareholders who combine their shares, they are computationally unbounded; they have infinite resources and time. And intuitively, this follows from the fact that, if I take the vector of n shares, which are computed by the dealer.

The property of these n shares is that you take any subset of t shares from this vector, its probability distribution is independent of the actual secret s. That means, say for instance, if I take the first t shares here s_1 to s_t, then what I am saying is, it does not matter whether your secret that was shared was s or the secret that was shared was s'; with equal probability the first t shares could be s_1 to s_t even for the secret s and with equal probability the same set of shares s_1 to s_t could be the shares for the secret s' as well.

(Refer Slide Time: 36:16)



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

❑ Definition: Let $\mathcal{F}^{s,t} \stackrel{\text{def}}{=}$ set of all t-degree polynomials over $\mathbb{F}$, with s as the constant term --- $|\mathcal{F}^{s,t}| = |\mathbb{F}|^t$

❑ Claim: Given arbitrary shares $s_1, \ldots, s_t$, there exists a unique $f(X) \in \mathcal{F}^{s,t}$, with $s_i = f(x_i)$, for $i = 1, \cdots, t$

❖ The points $(0, s), (x_1, s_1), \ldots, (x_t, s_t)$ can lie only on a single t-degree polynomial

❑ Claim: The shares $s_1, \ldots, s_t$ are independent of the underlying shared secret

Pr[first t shares are $s_1, \ldots, s_t$ |shared secret = s] = Pr[first t shares are $s_1, \ldots, s_t$ |shared secret = s'] = $\frac{1}{|\mathbb{F}|^t}$

So, let us prove this formally. So, for that, let me define this set $\mathcal{F}^{s,t}$, which denotes a set of all possible t degree polynomials over the field, whose constant term is the secret s. And how many such polynomials I can have? Well, I can have order of the field raised to power t ($|F|^t$) such polynomials. This is because any polynomial in this set will be of this form where the constant term is fixed, but you have the flexibility to choose the remaining t coefficients.

And for each of the remaining t coefficients of any polynomial in this set, you have order of field number of options because they could be any element from the field. So that is why you can have these many number of possible polynomials over the field whose constant term is the secret s. Now, my goal is to show that the probability distribution of any t shares computed in an instance of Shamir's secret sharing is independent of the actual secret.

So, for proving that I imagine here that let the first t shareholders are the corrupt shareholders. That means, I want to prove that the probability distribution of the first t shares are independent of the actual secret which is shared. You can prove that the probability distribution on any subset of t shares is independent of the actual secret, but for simplicity, I am proving with respect to the first t shares here.

So, my claim here is the following: if I give you arbitrary value of the first t shares, there exists a unique polynomial of degree t with constant term being the secret s such that this s_1 to s_t could occur as the first t shares. And this is because in order that s_1 to s_t constitutes the shares, or the first t shares for the secret s, it should be the case that there should be a t degree polynomial whose value at 0 should be s, whose value at α_1 should be s_1 whose value at α_2 should be s_2, and like that, whose value at α_t could be s_t. And there could be only one such polynomial, you cannot have 2 different polynomials $f_1(X)$ as well as $f_2(X)$ simultaneously passing through all these points, because you have t + 1 of these points in number. And we have proved that 2 different t degree polynomials can have common values at at most t points. That means at most t points could be common both to $f_1(X)$ as well as $f_2(X)$, you cannot have t + 1 points common to both $f_1(X)$ as well as $f_2(X)$. That is the simple fact here.

Now to prove that the probability distribution of the first t shares is independent of the actual secret, let us take 2 possible candidate secret which could be shared in an instance of Shamir's secret sharing scheme call them as s and s'. My claim is that it does not matter whether the secret is s or whether the secret is s' with equal probability you could get s_1 to s_t as the first t shares in an instance of Shamir's secret sharing scheme.

Basically, I want to calculate the following 2 conditional probabilities. The first conditional probability here is that given that the secret shared is s in an instance of Shamir's secret sharing scheme, what is the probability that the first t shares are s_1 to s_t? And the second conditional probability here is that given that the secret shared is s', where of course, s' is different from s, what is the probability that the same values s_1 to s_t occurs as the first t shares in that instance of Shamir's secret sharing scheme.

And it is easy to see that both these conditional probabilities are exactly the same namely, $1/(|\mathbf{F}|^t)$, because in order that the shares s_1 to s_t occurs in an instance of Shamir's secret sharing, where the shared secret is s, it should be the case that dealer should have chosen the unique t degree polynomial f(X) during the sharing phase for sharing s, which passes through the points (0, s), (α_1, s_1) and like that (α_t, s_t).

But what is the probability that among all possible polynomials from this set, $\mathcal{F}^{s,t}$,. dealer actually chose this polynomial f(X)? Well, it is $1/(|\mathbf{F}|^t)$. In the same way, what is the probability that if dealer wanted to share the secret s', it would result the first t shares to be s_1 to s_t the probability of that is exactly the same as the probability that dealer chooses that unique polynomial of degree t say g(X) from this collection $\mathcal{F}^{s',t}$, passing through the points (0, s'), (α_1, s_1) and like that (α_t, s_t).

But there are field size raised to power t number of polynomials which dealer could have used for secret sharing s'**.** Among those field size raise to power t polynomials, the probability that dealer has actually chosen the polynomial g(X) is again $1/(|\mathbf{F}|^t)$. So, that shows it does not matter whether the secret shared is s or s' with equal probability the shares s_1 to s_t could occur as the first t shares. That means, the probability distribution of those t shares is independent of the actual secret and which formally proves that you are Shamir's secret sharing scheme does not reveal any information if t shares are compromised.

(Refer Slide Time: 42:51)