Advanced Tools from Modern Cryptography

Lecture 2 First Tool: Secret-Sharing

Secret-Sharing

Dealer encodes a message into n shares for n parties

Privileged subsets of parties should be able to reconstruct the secret
<u>Access Structure: Set of all privileged sets</u>

View of an unprivileged subset should be independent of the secret

Ø Very useful

- Direct applications (distributed storage of data or keys)
- Important component in other cryptographic constructions
 - Secure multi-party computation
 - Attribute-Based Encryption
 - Leakage resilience ...

Threshold Secret-Sharing

@ (n,t)-secret-sharing

Divide a message m into n shares s₁,...,s_n, such that
 any t shares are enough to reconstruct the secret
 upto t-1 shares should have no information about the secret
 e.g., (s₁,...,s_{t-1}) has the secret

Recall last time: (2,2) secret-sharing

e.g., (s₁,...,s_{t-1}) has the same distribution for every m in the message space

Threshold Secret-Sharing

Construction: (n,n) secret-sharing

Additive Secret-Sharing

Message-space = share-space = G, a finite group
e.g. G = Z₂ (group of bits, with xor as the group operation)
or, G = Z₂^d (group of d-bit strings)
or, G = Z_P (group of integers mod p)
Share(M):

@ Pick $s_1, ..., s_{n-1}$ uniformly at random from G

 $Det s_n = - (s_1 + ... + s_{n-1}) + M$

Claim: This is an (n,n) secret-sharing scheme [Why?]

Additive Secret-Sharing: Proof

Share(M):

PROOF

O Pick s_1, \dots, s_{n-1} uniformly at random from G

Claim: Upto n-1 shares give no information about M

Proof: Let T ⊆ {1,...,n}, |T| = n-1. We shall show that { s_i }_{i∈T} is distributed the same way (in fact, uniformly) irrespective of what M is.

For T = {1,...,n-1}, true by construction. How about other T?

For concreteness consider T = {2,...,n}. Fix any (n-1)-tuple of elements in G,

 $(g_1,...,g_{n-1}) \in G^{n-1}$. To prove $Pr[(s_2,...,s_n)=(g_1,...,g_{n-1})]$ is same for all M. Fix any M.

 $(s_2,...,s_n) = (g_1,...,g_{n-1}) \Leftrightarrow (s_2,...,s_{n-1}) = (g_1,...,g_{n-2}) \text{ and } s_1 = M-(g_1+...+g_{n-1}).$

So $Pr[(s_2,...,s_n)=(g_1,...,g_{n-1})] = Pr[(s_1,...,s_{n-1})=(a,g_1,...,g_{n-2})], a:=(M-(g_1+...+g_{n-1}))$ But $Pr[(s_1,...,s_{n-1})=(a,g_1,...,g_{n-2})] = 1/|G|^{n-1}$, since $(s_1,...,s_{n-1})$ uniform over G^{n-1} Hence $Pr[(s_2,...,s_n)=(g_1,...,g_{n-1})] = 1/|G|^{n-1}$, irrespective of M.

An Application

Gives a "private summation" protocol (for <u>commutative</u> groups)

Clients with inputs



Secure against <u>passive</u> corruption" (i.e., no colluding set of servers/clients learn more than what they must) if at least one server stays out of the collusion

Linear Secret-Sharing

Another look at additive secret-sharing

Working with a commutative group here. Multiplication by ±1 and 0 well-defined in a group. But more broadly, we shall consider a field.



More generally, a share can have multiple coordinates

Linear Secret-Sharing over a field: message and shares are field elements
 Reconstruction by a set T ⊆ [n] : <u>solve</u> the message from given shares
 i.e., solve W_T $\begin{bmatrix} M \\ r \end{bmatrix} = s_T$ for M

Security of Linear Secret-Sharing

Claim: Every such linear scheme is a secure secret-sharing scheme for some access structure

3 Suppose T \subseteq [n] s.t. M not uniquely reconstructible from \underline{s}_T

of i.e., solution space (of $\underline{z} = \begin{bmatrix} M \\ r \end{bmatrix}$) for W_T·<u>z</u> = <u>s</u>_T contains at least two points with distinct values α and β for M

Then, $\forall y \in F$, the solution space has a point with M=y (e.g., linear combination of the above points with factors $(y-\beta)/(a-\beta)$ and $(a-y)/(a-\beta)$)

Therefore, for any y ∈ F, can add equation M=y and get a solution space of dimension equal to the nullity of the system, say k
i.e., with M=y, exactly |F|^k choices of randomness <u>r</u> that give <u>s</u>_T
i.e., for all <u>s</u>_T and y, Pr[view=<u>s</u>_T | M=y] = |F|^k/|F|^{t-1}

Threshold Secret-Sharing Construction: (n,2) secret-sharing Message-space = share-space = F, a finite field (e.g. integers mod prime) Share(M): pick random r. Let $s_i = r \cdot a_i + M$ (for i=1,...,n < |F|) Reconstruct(s_i, s_j): r = (s_i-s_j)/(a_i-a_j); M = s_i - r · a_i a_i are n distinct, non-zero field elements Each s_i by itself is uniformly distributed.
 irrespective of M [Why?] < Since a_i-1 exists, exactly one solution for r·a_i+M=d, for Geometric interpretation every value of d Sharing picks a random "line" y = f(x), such that f(0)=M. Shares $s_i = f(a_i)$. s_i is independent of M: exactly one line passing 2 3 through (a_i,s_i) and (O,M') for any secret M' But can reconstruct the line from two points!

Threshold Secret-Sharing

(n,t) secret-sharing in a (large enough) field F Shamir Secret-Sharing

Generalizing the geometric/algebraic view: instead of lines, use polynomials

Share(m): Pick a random degree t-1 polynomial f(X), such that f(0)=M. Shares are s_i = f(a_i).

Random polynomial with f(0)=M: $c_0 + c_1X + c_2X^2 + ... + c_{t-1}X^{t-1}$ by picking c_0 =M and $c_1, ..., c_{t-1}$ at random.

Given t points can reconstruct the polynomial. Given < t points, for any M', there are polynomials which pass through (0,M')

Secrecy: Shamir's scheme is linear!

Linearity of Shamir Secret-Sharing

Shamir's scheme is a linear secret-sharing scheme



Which sets T ⊆ [n] can reconstruct? i.e., T s.t. W_T spans [1 0 ... 0]?
 W_T spans [1 0 ... 0] iff |T| ≥ t

The for |T|=t, W_T is a Vandermonde matrix, and is a basis for \mathbb{F}^+

For |T| < t, can add a row [1 0 ... 0] and (optionally) more rows of the form [1 a a²... a[†]] to get a Vandermonde matrix. So [1 0 ... 0] is independent of the rows of W_T

Secrecy: guaranteed for any linear secret-sharing scheme

More General Access Structures

O Idea: For arbitrary monotonic access structure \mathcal{A} , there is a "basis" \mathcal{B} of minimal sets in \mathcal{A} . For each S in \mathcal{B} generate an (|S|,|S|) sharing, and distribute them to the members of S. $|\mathcal{B}| =$ Ø Works, but very "inefficient" \odot How big is \mathcal{B} ? (Say when \mathcal{A} is a threshold access structure) **Total share complexity = \Sigma_{S \in B} |S| field elements. (Compare** with Shamir's scheme: n field elements in all.) More efficient schemes known for large classes of access structures

More General Access Structures

(2,3)

(1,3)

Shares

(2,2)

Shares

of shares

A simple generalization of threshold access structures

A <u>threshold tree</u> to specify the access structure

Can realize by recursively threshold secret-sharing the shares

Check: <u>linear</u> secret-sharing

Fact: Access structures that admit linear secret-sharing are those which can be specified using "monotone span programs"

(2,3)

Today

Secret-sharing schemes (n,t) Threshold secret-sharing Additive sharing for (n,n) Shamir secret-sharing for all (n,t) Optimal (ideal) when message-space is a field with more than n elements Linear secret-sharing