Threshold Secret Sharing: Information-Theoretic

Threshold Secret Sharing

Zelda has a **secret** $s \in \{0,1\}^n$.

Def: Let $1 \le t \le m$. **(t, L)-secret sharing** is a way for Zelda to give strings to $A_1, \ldots, A_L$ such that:

1. If any t get together than they can learn the secret.
2. If any $t-1$ get together they cannot learn the secret.

Cannot learn the secret will be info-theoretic. Even if $t-1$ people have big fancy supercomputers they cannot learn s. We will later look at comp-security.

Applications

Rumor: Secret Sharing is used for the Russian Nuclear Codes. There are three people (one is Putin) and if two of them agree to launch, they can launch.

For people signing a contract long distance secret sharing is used as a building block in the protocol.

$(4,4)$-secret sharing

A_1, A_2, A_3, A_4 such that

1. If all four of A_1, A_2, A_3, A_4 get together they can find s.
2. If any three of them get together then learn **NOTHING**.

An Attempt at $(4, 4)$-Secret Sharing

1. Zelda breaks s up into $s = s_1 s_1 s_3 s_4$ where
$$|s_1| = |s_2| = |s_3| = |s_4| = \frac{n}{4}$$
2. Zelda gives A_i the string s_i.

Does this work?

An Attempt at $(4,4)$-Secret Sharing

1. Zelda breaks s up into $s = s_1 s_1 s_3 s_4$ where

$$|s_1| = |s_2| = |s_3| = |s_4| = \frac{n}{4}$$

2. Zelda gives A_i the string s_i.

Does this work?

1. If A_1, A_2, A_3, A_4 get together they can find s.

An Attempt at $(4, 4)$-Secret Sharing

1. Zelda breaks s up into $s = s_1 s_1 s_3 s_4$ where

$$|s_1| = |s_2| = |s_3| = |s_4| = \frac{n}{4}$$

2. Zelda gives A_i the string s_i.

Does this work?

1. If A_1, A_2, A_3, A_4 get together they can find s. **YES!!**

An Attempt at $(4, 4)$-Secret Sharing

1. Zelda breaks s up into $s = s_1 s_1 s_3 s_4$ where

$$|s_1| = |s_2| = |s_3| = |s_4| = \frac{n}{4}$$

2. Zelda gives A_i the string s_i.

Does this work?

1. If A_1, A_2, A_3, A_4 get together they can find s. **YES!!**
2. If any three of them get together they learn **NOTHING**.

An Attempt at $(4,4)$-Secret Sharing

1. Zelda breaks s up into $s = s_1 s_1 s_3 s_4$ where

$$|s_1| = |s_2| = |s_3| = |s_4| = \frac{n}{4}$$

2. Zelda gives A_i the string s_i.

Does this work?

1. If A_1, A_2, A_3, A_4 get together they can find s. **YES!!**
2. If any three of them get together they learn **NOTHING**. **NO**.
   2.1 A_1 learns s_1 which is $\frac{1}{4}$ **of the secret!**
   2.2 A_1, A_2 learn $s_1 s_2$ which is $\frac{1}{2}$ **of the secret!**
   2.3 A_1, A_2, A_3 learn $s_1 s_2 s_3$ which is $\frac{3}{4}$ **of the secret!**

What do we mean by **NOTHING**?

If any three of them get together they learn **NOTHING**
Informally:

1. Before Zelda gives out shares, if any three A_i, A_j, A_k get together, they know $BLAH_{i,j,k}$.
2. After Zelda gives out shares, if any three A_i, A_j, A_k get together, they know $BLAH_{i,j,k}$.
3. Giving out the shares tells each triple **NOTHING** they did not already know.

If A_i, A_j, A_k have **unlimited computing power**

What do we mean by **NOTHING**?

If any three of them get together they learn **NOTHING**
Informally:

1. Before Zelda gives out shares, if any three A_i, A_j, A_k get together, they know $BLAH_{i,j,k}$.
2. After Zelda gives out shares, if any three A_i, A_j, A_k get together, they know $BLAH_{i,j,k}$.
3. Giving out the shares tells each triple **NOTHING** they did not already know.

If A_i, A_j, A_k have **unlimited computing power** they still learn **NOTHING**.

What do we mean by **NOTHING**?

If any three of them get together they learn **NOTHING**
Informally:

1. Before Zelda gives out shares, if any three A_i, A_j, A_k get together, they know $BLAH_{i,j,k}$.
2. After Zelda gives out shares, if any three A_i, A_j, A_k get together, they know $BLAH_{i,j,k}$.
3. Giving out the shares tells each triple **NOTHING** they did not already know.

If A_i, A_j, A_k have **unlimited computing power**
they still learn **NOTHING**.

Information-Theoretic Security

Is $(4, 4)$-Secret Sharing Possible?

VOTE: Is $(4, 4)$-Secret sharing possible?

1. YES
2. NO
3. YES given some hardness assumption
4. UNKNOWN TO SCIENCE

Is $(4,4)$-Secret Sharing Possible?

VOTE: Is $(4,4)$-Secret sharing possible?

1. YES
2. NO
3. YES given some hardness assumption
4. UNKNOWN TO SCIENCE

YES

Random String Approach

Zelda gives out shares of the secret

1. Secret $s \in \{0,1\}^n$. Zelda gen **random** $r_1, r_2, r_3 \in \{0,1\}^n$.
2. Zelda gives A_1 $s_1 = r_1$.
   Zelda gives A_2 $s_2 = r_2$.
   Zelda gives A_3 $s_3 = r_3$.
   Zelda gives A_4 $s_4 = s \oplus r_1 \oplus r_2 \oplus r_3$

A_1, A_2, A_3 A_4 Can Recover the Secret

$$s_1 \oplus s_2 \oplus s_3 \oplus s_4 = r_1 \oplus r_2 \oplus r_3 \oplus r_1 \oplus r_2 \oplus r_3 \oplus s = s$$

Easy to see that if a triple get together they learn NOTHING

$(2, 4)$-Secret Sharing using Random Strings

For each $1 \le i < j \le 4$

1. Zelda generates **random** $r \in \{0, 1\}^n$.
2. Zelda gives A_i the strings $s_{i,(i,j)} = ((i, j), r)$.
3. Zelda gives A_j the strings $s_{j,(i,j)} = ((i, j), r \oplus s)$.

A_i, A_j Can Recover the Secret

A_i takes $((i, j), r)$ and just uses the r.
A_j takes $((i, j), r \oplus s)$ and just uses the $r \oplus s$.
They both compute $r \oplus r \oplus s = s$.

Easy to see that one person learns NOTHING

(L, L)-Random String Method

People: $A_1, \ldots, A_L$. Secret s.

1. Zelda gen rand $r_1, \ldots, r_{L-1}$.
2. A_1 get r_1
   A_2 get r_2
   $\vdots$
   A_{L-1} gets r_{L-1}
   A_L gets $s \oplus r_1 \oplus \cdots \oplus r_{L-1}$
3. If they all get together they will XOR all their strings to get s

We use this as building block for gen case.

(t, L) Secret Sharing

People: $A_1, \ldots, A_L$. $S_1, \ldots, S_m \subseteq \{A_1, \ldots, A_L\}$ are all the sets of size t. $(m = \binom{L}{t})$.

1. For every $1 \le j \le m$ Zelda does (t, t) secret sharing with the elements of S_j but also prepends every string with j.
2. If the people in S_j get together they XOR together strings prepended with j (do not use the j).
3. No subset can get the secret.

PRO: Can always do Threshold Secret Sharing.
CON: You are giving people A LOT of strings!

How Many Strings Does A_i Get in $(5, 10)$-Secret Sharing?

If do $(5, 10)$ secret sharing then how many strings does A_1 get?

A_1 gets a string for every $J \subseteq \{1, \ldots, 10\}$, $|J| = 5$, $1 \in J$.

Equivalent to:

A_1 gets a string for every $J \subseteq \{2, \ldots, 10\}$, $|J| = 4$.

How many sets? **Discuss**

How Many Strings Does A_i Get in $(5,10)$-Secret Sharing?

If do $(5,10)$ secret sharing then how many strings does A_1 get?

A_1 gets a string for every $J \subseteq \{1,\ldots,10\}$, $|J| = 5$, $1 \in J$.

Equivalent to:

A_1 gets a string for every $J \subseteq \{2,\ldots,10\}$, $|J| = 4$.

How many sets? **Discuss**

$$\binom{9}{4} = 126 \text{ strings}$$

How Many Strings Does A_i Get in $(L/2, L)$-Secret Sharing?

If do $(L/2, L)$ secret sharing then how many strings does A_1 get?

A_1 gets a string for every $J \subseteq \{1, \ldots, L\}$, $|J| = \frac{L}{2}$, $1 \in J$.
Equivalent to:

A_1 gets a string for every $J \subseteq \{2, \ldots, L\}$, $|J| = \frac{L}{2} - 1$.

How many sets? **Discuss**

How Many Strings Does A_i Get in $(L/2, L)$-Secret Sharing?

If do $(L/2, L)$ secret sharing then how many strings does A_1 get?

A_1 gets a string for every $J \subseteq \{1, \ldots, L\}$, $|J| = \frac{L}{2}$, $1 \in J$.
Equivalent to:

A_1 gets a string for every $J \subseteq \{2, \ldots, L\}$, $|J| = \frac{L}{2} - 1$.

How many sets? **Discuss**

$$\binom{L-1}{\frac{L}{2}-1} \sim \frac{2^L}{\sqrt{L}} \text{ strings}$$

Thats A LOT of Strings!

Can We Reduce The Number of Strings for $(L/2, L)$?

Thats a lot of strings!

Can We Reduce The Number of Strings for $(L/2, L)$?

In our $(L/2, L)$-scheme each A_i gets $\sim \frac{2^L}{\sqrt{L}}$ strings.

VOTE

1. Requires roughly 2^L strings.
2. $O(\beta^L)$ strings for some $1 < \beta < 2$ but not poly.
3. $O(L^a)$ strings for some $a > 1$ but not linear.
4. $O(L)$ strings but not sublinear.
5. $O(\log L)$ strings but not constant.
6. $O(1)$ strings.

Can We Reduce The Number of Strings for $(L/2, L)$?

In our $(L/2, L)$-scheme each A_i gets $\sim \frac{2^L}{\sqrt{L}}$ strings.

VOTE

1. Requires roughly 2^L strings.
2. $O(\beta^L)$ strings for some $1 < \beta < 2$ but not poly.
3. $O(L^a)$ strings for some $a > 1$ but not linear.
4. $O(L)$ strings but not sublinear.
5. $O(\log L)$ strings but not constant.
6. $O(1)$ strings.

You can always do this problem with 1 string. Really!