# Threshold Secret Sharing: Information-Theoretic

# Threshold Secret Sharing

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

**Def:** Let $1 \leq t \leq m$. **$(t, L)$-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.

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

With the Random String Method:
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?

With the Random String Method:
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!**

# Secret Sharing With Polynomials

We do $(3,6)$-Secret Sharing.

1. Secret $s$. Zelda picks prime $p \sim s$, Zelda works mod $p$.
2. Zelda gen rand numbers $a_2, a_1 \in \{0, \ldots, p-1\}$
3. Zelda forms polynomial $f(x) = a_2x^2 + a_1x + s$.
4. Zelda gives $A_1$ $f(1)$, $A_2$ $f(2)$, ..., $A_6$ $f(6)$ (all mod $p$). These are all of length $\sim |s|$.

1. Any 3 have 3 points from $f(x)$ so can find $f(x)$, $s$.
2. Any 2 have 2 points from $f(x)$. Constant term $(s)$ **anything!**.

# Example

$s = 20$. We'll use $p = 23$.

1. Zelda picks $a_2 = 8$ and $a_1 = 13$.
2. Zelda forms polynomial $f(x) = 8x^2 + 13x + 20$.
3. Zelda gives $A_1$ $f(1) = 18$, $A_2$ $f(2) = 9$, $A_3$ $f(3) = 16$, $A_4$ $f(4) = 16$, $A_5$ $f(5) = 9$, $A_6$ $f(6) = 18$.

If $A_1, A_3, A_4$ get together and want to find $f(x)$ hence $s$.
$f(x) = a_2x^2 + a_1x + s$.
$f(1) = 18$: $a_2 \times 1^2 + a_1 \times 1 + s \equiv 18 \pmod{23}$
$f(3) = 16$: $a_2 \times 3^2 + a_1 \times 3 + s \equiv 16 \pmod{23}$
$f(4) = 16$: $a_2 \times 4^2 + a_1 \times 4 + s \equiv 16 \pmod{23}$
3 linear equations in, 3 variable, over mod 23 can be solved.
**Note:** Only need constant term $s$ but can get all coeffs.

# What if Two Get Together?

What if $A_1$ and $A_3$ get together:
$f(1) = 18$: $a_2 \times 1^2 + a_1 \times 1 + s \equiv 18 \pmod{23}$
$f(3) = 16$: $a_2 \times 3^2 + a_1 \times 3 + s \equiv 16 \pmod{23}$
Can they solve these to find $s$ **Discuss**.

# What if Two Get Together?

What if $A_1$ and $A_3$ get together:
$f(1) = 18$: $a_2 \times 1^2 + a_1 \times 1 + s \equiv 18 \pmod{23}$
$f(3) = 16$: $a_2 \times 3^2 + a_1 \times 3 + s \equiv 16 \pmod{23}$
Can they solve these to find $s$ **Discuss**.

No. However, can they use these equations to eliminate some values of $s$? **Discuss**.

# What if Two Get Together?

What if $A_1$ and $A_3$ get together:
$f(1) = 18$: $a_2 \times 1^2 + a_1 \times 1 + s \equiv 18 \pmod{23}$
$f(3) = 16$: $a_2 \times 3^2 + a_1 \times 3 + s \equiv 16 \pmod{23}$
Can they solve these to find $s$ **Discuss**.

No. However, can they use these equations to eliminate some values of $s$? **Discuss**.

No. ANY $s$ is consistent. If you pick a value of $s$ you then have two equations in two variables that can be solved.

# What if Two Get Together?

What if $A_1$ and $A_3$ get together:
$f(1) = 18$: $a_2 \times 1^2 + a_1 \times 1 + s \equiv 18 \pmod{23}$
$f(3) = 16$: $a_2 \times 3^2 + a_1 \times 3 + s \equiv 16 \pmod{23}$
Can they solve these to find $s$ **Discuss**.

No. However, can they use these equations to eliminate some values of $s$? **Discuss**.

No. ANY $s$ is consistent. If you pick a value of $s$ you then have two equations in two variables that can be solved.

**Important:** Information-Theoretic Secure: if $A_1$ and $A_3$ meet they learn NOTHING. If they had big fancy supercomputers they would still learn NOTHING.

# A Note About Linear Equations

The three equations below, over mod 23, can be solved:

$a_2 \times 1^2 + a_1 \times 1 + s \equiv 18 \pmod{23}$
$a_2 \times 3^2 + a_1 \times 3 + s \equiv 16 \pmod{23}$
$a_2 \times 4^2 + a_1 \times 4 + s \equiv 16 \pmod{23}$

Could we have solved this had we used mod 24?

**VOTE**

1. YES
2. NO

# A Note About Linear Equations

The three equations below, over mod 23, can be solved:
$a_2 \times 1^2 + a_1 \times 1 + s \equiv 18 \pmod{23}$
$a_2 \times 3^2 + a_1 \times 3 + s \equiv 16 \pmod{23}$
$a_2 \times 4^2 + a_1 \times 4 + s \equiv 16 \pmod{23}$

Could we have solved this had we used mod 24?
**VOTE**

1. YES
2. NO

**NO**
Need a domain where every number has a mult inverse.
Over mod $p$, $p$ primes, all numbers have mult inverses.
Over Mod 24 even number do not have mult inverse.

# Threshold Secret Sharing With Polynomials: Ref

Will be on next few slides.
Due to Adi Shamir
**How to Share a Secret**
**Communication of the ACM**
**Volume 22, Number 11**
**1979**

# Threshold Secret Sharing With Polynomials

Zelda wants to give strings to $A_1, \ldots, A_L$ such that

Any $t$ of $A_1, \ldots, A_L$ can find $s$. Any $t-1$ learn **NOTHING**.

1. Secret $s$. Zelda picks prime $p \sim s$, Zelda works mod $p$.
2. Zelda gen rand $a_{t-1}, \ldots, a_1 \in \{0, \ldots, p-1\}$
3. Zelda forms polynomial $f(x) = a_{t-1}x^{t-1} + \cdots + a_1 x + s$.
4. For $1 \le i \le L$ Zelda gives $A_i$ $f(i)$ mod $p$.

1. Any $t$ have $t$ points of $f(x)$ so can find $f(x)$ and $s$.
2. Any $t-1$ have $t-1$ points of $f(x)$. Constant term $(s)$ could be **anything!**.

# We Used Polynomials. Could Use...

What did we use about degree $t-1$ polynomials?

1. $t$ points determine a the polynomial (we need constant term).
2. $t-1$ points give **no information** about constant term.

Could do geometry over $\mathbb{Z}_p^3$. A **Plane** in $\mathbb{Z}_p^3$ is:

$$\{(x,y,z) : ax+by+cz=d\}$$

1. 3 points in $\mathbb{Z}_p^3$ determine a plane.
2. 2 points in $\mathbb{Z}_p^3$ give **no information** about $d$.

This approach is due to George Blakely, **Safeguarding Cryptographic Keys**, **International Workshop on Managing Requirements**, **Vol 48**, **1979**.
We will not do secret sharing this way, though one could.

# We Used Polynomials. Could Use...

We won't go into details but there are two ways to use the **Chinese Remainder Theorem** to do Secret Sharing.

Due to:
C.A. Asmuth and J. Bloom. **A modular approach to key safeguarding. IEEE Transactions on Information Theory Vol 29**, **Number 2**, **208-210**, **1983**.

And Independently

M. Mignotte **How to share a secret**, **Cryptography: Proceedings of the Workshop on Cryptography, Burg Deursetein**, **Volume 149 of Lecture Notes in Computer Science**, **1982**.

# Features and Caveats of Poly Method

Imagine that you've done $(t, L)$ secret sharing with polynomial, $p(x)$. So for $1 \leq i \leq L$, $A_i$ has $f(i)$.

1. **Feature:** If more people come FINE- can extend to $(t, L+a)$ by giving $A_{L+1}$, $f(L+1)$, ..., $A_{L+a}$, $f(L+a)$.
2. **Caveat:** If $L > p$ then you run out of points to give people. We will always assume $L < p$.
3. **Caveat:** If $L > p$ there are still ways to do this, but we won't get into that.

# Length of Shares

$s = 1111$, length 4. This is 15 in base 10, so we go to smallest prime $> 15$, namely 17.

We use $p = 17$. $s = 1111$, $|s| = 4$.

Elements of $\mathbb{Z}_{17}$ are represented by strings of length 5

1. Everyone gets at least one share.
2. All shares length 5, even though $s$ is length 4.

Can we always get get length $n$? Length $n + 1$?

# Length of Shares

If $|s| = n$ want prime $p$ with $2^n < p$.
**Known:** For all $n$ there exists prime $p$ with $2^n \le p \le 2^{n+1}$.

**Upshot:** The secret is length $n$, the shares are of length $n+1$.

**Good News:** Every $A_i$ gets ONE share.

**Bad News:** That share is of length $n+1$, not $n$.

**VOTE:** Can Zelda do threshold secret sharing where every student gets ONE share of length $n$?

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

# Length of Shares

If $|s| = n$ want prime $p$ with $2^n < p$.
**Known:** For all $n$ there exists prime $p$ with $2^n \leq p \leq 2^{n+1}$.

**Upshot:** The secret is length $n$, the shares are of length $n + 1$.

**Good News:** Every $A_i$ gets ONE share.

**Bad News:** That share is of length $n + 1$, not $n$.

**VOTE:** Can Zelda do threshold secret sharing where every student gets ONE share of length $n$?

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

**YES**

# Why Did We Use Primes?

We used $\mathbb{Z}_p$ since need to inverses.

**Def:** A **Field** is a set $F$ together with operations $+, \times$ such that

1. There is a 0 element such that $(\forall x)[x + 0 = x]$.
2. There is a 1 element such that $(\forall x)[x \times 1 = x]$.
3. $(\forall x, y)[(x + y = y + x) \wedge (x \times y = y \times x)]$.
4. $(\forall x, y, z)[x \times (y + z) = x \times y + x \times z]$.
5. $(\forall x)(\exists y)[x + y = 0]$.
6. $(\forall x \neq 0)(\exists y)[x \times y = 1]$. (This one is KEY.)

**WE USED:** $p$ prime iff $\mathbb{Z}_p$ a field.

# Can we use a different field?

**KEY:** There is a field of size $p^n$ for all primes $p$ and $n \geq 1$.

**WE USE:** For all $n$ there is a field on $2^n$ elements.
If secret is $s$ of length $n$, use the field on $2^n$ elements. All elements of it are of length $n$.

**Upshot:** For threshold there is a secret sharing scheme where everyone gets ONE share of size EXACTLY the size of the secret.

# Example: A Field of 32 elements

$\mathbb{Z}_2[x]$ is the set of polys over $\mathbb{Z}_2$. $x^5+x^2+1$ is irreducible in $\mathbb{Z}_2[x]$.

Field on $2^5$ elements:

1. The elements are polys in $\mathbb{Z}_2[x]$ of degree $\le 4$.
2. Addition and subtraction are as usual.
3. Mult is MOD $x^5+x^2+1$. So Mult two polys together and
   Replace $x^5$ with $-x^2-1=x^2+1$
   Replace $x^6$ with $-x^3-x=x^3+x$
   Replace $x^7$ with $-x^4-x^2=x^4+x^2$
   Replace $x^8$ with $-x^5-x^3=x^5+x^2 \equiv 2x^2+1$
4. One can show that this is a Field- mult has inverses. For that proof need that the poly $x^5+x^2+1$ is irreducible.

# Field on $p^a$ Elements

$p$ a prime.
$\mathbb{Z}_p[x]$ is the set of polynomials over $\mathbb{Z}_p$.
$f(x)$ is irreducible in $\mathbb{Z}_p[x]$, and of degree $a$

Field on $p^a$ elements:

1. The elements are polys in $\mathbb{Z}_p[x]$ of degree $\le a-1$.
2. Addition and subtraction are as usual.
3. Mult is MOD $f(x)$. So Multiply two polys together and mod down to degree $\le a-1$ by assuming $f(x)=0$.
4. One can show that this is a Field- mult has inverses. For that proof need that the poly $f(x)$ is irreducible.

# Practical and Pedagogical Point

1. We **could** from now on, on HW and exams and slides and notes, work over the field on $2^n$ elements and have shares of length **exactly** the size of the secret.

# Practical and Pedagogical Point

1. We **could** from now on, on HW and exams and slides and notes, work over the field on $2^n$ elements and have shares of length **exactly** the size of the secret.
2. That would be madness! Madness I say!

# Practical and Pedagogical Point

1. We **could** from now on, on HW and exams and slides and notes, work over the field on $2^n$ elements and have shares of length **exactly** the size of the secret.
2. That would be madness! Madness I say!
3. For pedagogue we work over $\mathbb{Z}_p$ for some well chosen $p$.

# Practical and Pedagogical Point

1. We **could** from now on, on HW and exams and slides and notes, work over the field on $2^n$ elements and have shares of length **exactly** the size of the secret.
2. That would be madness! Madness I say!
3. For pedagogue we work over $\mathbb{Z}_p$ for some well chosen $p$.
4. We will **cheat and lie**. We will say **the shares are the same length as the secret** when may be off by 1 (YES, just by 1) because we use primes instead of $GF(2^n)$ (Whats that? Galois Field on $2^n$ elements. Duh :-) )

# Practical and Pedagogical Point

1. We **could** from now on, on HW and exams and slides and notes, work over the field on $2^n$ elements and have shares of length **exactly** the size of the secret.
2. That would be madness! Madness I say!
3. For pedagogue we work over $\mathbb{Z}_p$ for some well chosen $p$.
4. We will **cheat and lie**. We will say **the shares are the same length as the secret** when may be off by 1 (YES, just by 1) because we use primes instead of $GF(2^n)$ (Whats that? Galois Field on $2^n$ elements. Duh :-) )
5. In the real world they use primes.

# Practical and Pedagogical Point

1. We **could** from now on, on HW and exams and slides and notes, work over the field on $2^n$ elements and have shares of length **exactly** the size of the secret.
2. That would be madness! Madness I say!
3. For pedagogue we work over $\mathbb{Z}_p$ for some well chosen $p$.
4. We will **cheat and lie**. We will say **the shares are the same length as the secret** when may be off by 1 (YES, just by 1) because we use primes instead of $GF(2^n)$ (Whats that? Galois Field on $2^n$ elements. Duh :-) )
5. In the real world they use primes. I think.

# Practical and Pedagogical Point

1. We **could** from now on, on HW and exams and slides and notes, work over the field on $2^n$ elements and have shares of length **exactly** the size of the secret.
2. That would be madness! Madness I say!
3. For pedagogue we work over $\mathbb{Z}_p$ for some well chosen $p$.
4. We will **cheat and lie**. We will say **the shares are the same length as the secret** when may be off by 1 (YES, just by 1) because we use primes instead of $GF(2^n)$ (Whats that? Galois Field on $2^n$ elements. Duh :-) )
5. In the real world they use primes. I think. I'll ask Putin.

# Can Shares be SHORTER than Secret?

**Thm** There is a $(t, L)$ scheme, $|s| = n$, all shares $\le \frac{2n}{t}$.

Zelda's secret is $s = s_0 s_1 s_2 \cdots s_{t-1}$ where each $s_i$ is of length $\frac{n}{t}$.
Zelda uses $\mathbb{Z}_p$, $p \sim 2^{n/t}$. Zelda gen rand $k$ of length $n$.
$k = k_0 k_1 \cdots k_{t-1}$, $|k_i| = \frac{n}{t}$. Zelda creates two polynomials:

$$f(x) = (s_{t-1} \oplus k_{t-1})x^t + \cdots (s_1 \oplus k_1)x + (s_0 \oplus k_0)$$

$$g(x) = k_{t-1}x^t + \cdots k_1 x + k_0$$

For $1 \le i \le m$ Zelda gives $A_i$ $(f(i), g(i))$.
**Note:** Everyone gets a share of size $\frac{2n}{t}$.
**Note:** Scheme uses all coeffs not just constant.
Next slide on recovery and security.

# Recover and Security

**Recovery:** If $t$ get together they can determine both polynomials (not just the constant term). Hence they all know:

$$s_{t-1} \oplus k_{t-1}, \cdots, s_1 \oplus k_1, s_0 \oplus k_0$$

$$k_{t-1}, \cdots, k_1, k_0$$

From this can easily get $s_{t-1}, \ldots, s_1, s_0$.

**Discuss Security:** $t-1$ people cannot learn **anything**.

# Security

# Security

**You've Been Punked!!**

# You've Been Punked!!

$A_1, \ldots, A_{t-1}$ can get **some** information.
They know that $A_t$ has a share of length $\frac{2n}{t}$.
They do the following:

> $CAND = \emptyset$. $CAND$ will be set of Candidates for $s$.
>
> For $x \in \{0,1\}^{2n/t}$ (go through ALL shares $A_t$ could have)
>
> > $A_1, \ldots, A_{t-1}$ pretend $A_t$ has $x$ and deduce candidates secret $s'$
> > $CAND := CAND \cup \{s'\}$
>
> Secret is in $CAND$. $|CAND| = 2^{2n/t} < 2^n$. So we have **eliminated** many strings from being the $s$

# Can we use even shorter shares?

$|s| = n$, $(t, L)$-secret sharing.
Is there a scheme where someone gets share of size $< n$? We will allow others to get long shares (larger than $n$)
**VOTE**

1. $(\exists)$ scheme, $A_1$ gets size $n - 1$.
2. $(\exists)$ scheme, $A_1$ gets size $\lceil n/2 \rceil$.
3. $(\exists)$ scheme, $A_1$ gets size $\lceil \sqrt{n} \rceil$.
4. $(\exists)$ scheme, $A_1$ gets size $\lceil \log n \rceil$.
5. NO- in ANY scheme $A_1$ MUST get size $\geq n$.

# Can we use even shorter shares?

$|s| = n$, $(t, L)$-secret sharing.
Is there a scheme where someone gets share of size $< n$? We will allow others to get long shares (larger than $n$)
**VOTE**

1. $(\exists)$ scheme, $A_1$ gets size $n-1$.
2. $(\exists)$ scheme, $A_1$ gets size $\lceil n/2 \rceil$.
3. $(\exists)$ scheme, $A_1$ gets size $\lceil \sqrt{n} \rceil$.
4. $(\exists)$ scheme, $A_1$ gets size $\lceil \log n \rceil$.
5. NO- in ANY scheme $A_1$ MUST get size $\geq n$.

NO- proof on next slide.

# Nobody Gets Short Share

They know that $A_t$ has a share of length $n-1$.
They do the following:

$CAND = \emptyset$. $CAND$ will be set of Candidates for $s$.

For $x \in \{0,1\}^{n-1}$ (go through ALL shares $A_t$ could have)

$A_1, \ldots, A_{t-1}$ pretend $A_t$ has $x$ and deduce candidates secret $s'$
$CAND := CAND \cup \{s'\}$

Secret is in $CAND$. $|CAND| = 2^{n-1} < 2^n$. So we have **eliminated** many strings from being the $s$