21/12/21

Voglio generalizzare secret sharing, invece di threshold (t, n)
Voglio più "libertà" (es (3,4) ~> (2,3) se ho 3 persone)

# Linear Secret Sharing
# &
# Access Control Matrix

# Revisiting Shamir Scheme

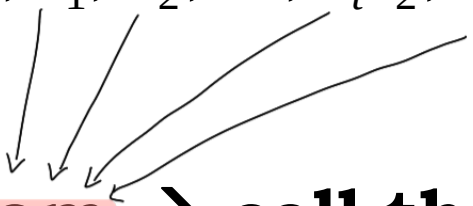
➔**Secret s**

➔**Share i:** $y_i = s + a_1 x_i + a_2 x_i^2 + \cdots + a_{t-2} x_i^{t-2} + a_{t-1} x_i^{t-1}$

è un prodotto tra vettori!

➔**Vector interpretation: scalar product**

$$y_i = [1, x_i, x_i^2, \cdots, x_i^{t-2}, x_i^{t-1}] \bullet [s, a_1, a_2, \cdots, a_{t-2}, a_{t-1}]$$



➔**Coefficients $a_i$ are random → call them $r_i$**

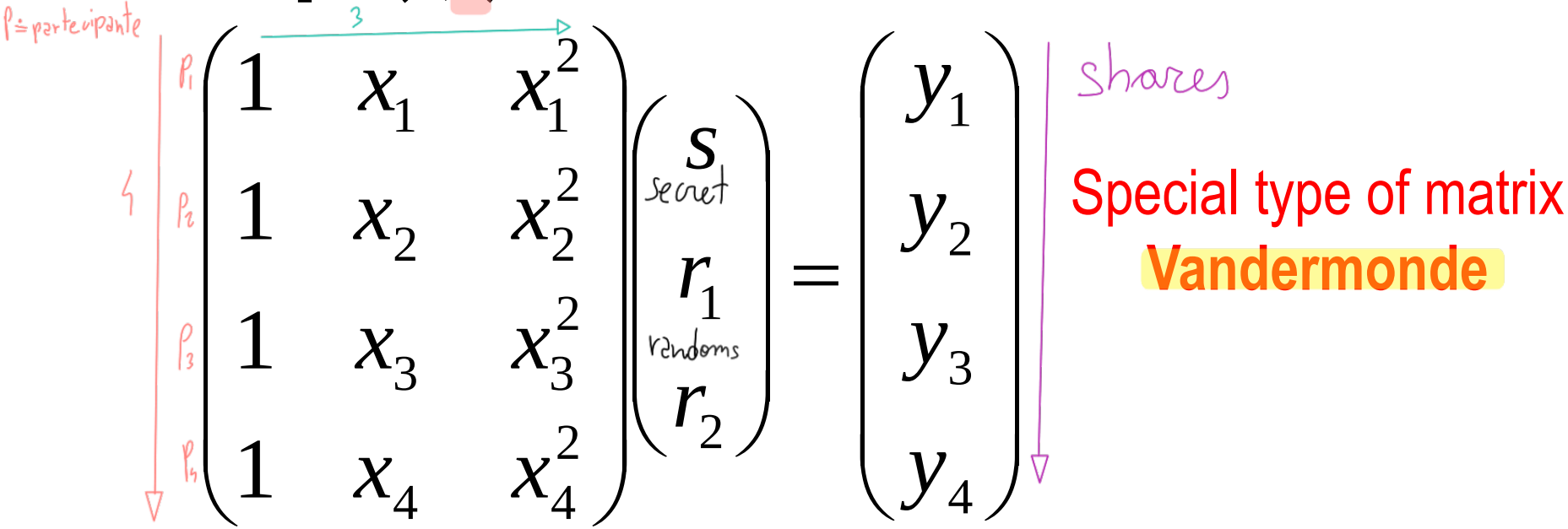
$$y_i = [1, x_i, x_i^2, \cdots, x_i^{t-2}, x_i^{t-1}] \bullet [s, r_1, r_2, \cdots, r_{t-2}, r_{t-1}]$$

# Shamir scheme in matrix form

$$Ax = b$$

- A → matrix, n × t **GIVEN**
- x → vector, t **[secret, rand, rand,…]**
- b → vector, n **resulting shares**

- **Example: (3,4)**



# Reconstructing secret

➔**Linear system**

⇨t entries

⇨E.g. 3 shares out of 4

$$\begin{pmatrix} 1 & x_1 & x_1^2 \\ 1 & x_2 & x_2^2 \\ 1 & x_4 & x_4^2 \end{pmatrix} \begin{pmatrix} s \\ r_1 \\ r_2 \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ y_4 \end{pmatrix}$$

faccio $A^{-1}$ e ritrovo $\langle s, r_1, r_2 \rangle$

**Reconstruction coefficients:**

- **Previous: Lagrange formula**
- **Now: MIGHT solve as linear system**
  - **Solution for s gives usual Laplace formula, of course**

$$\begin{pmatrix} s \\ r_1 \\ r_2 \end{pmatrix} = \begin{pmatrix} 1 & x_1 & x_1^2 \\ 1 & x_2 & x_2^2 \\ 1 & x_4 & x_4^2 \end{pmatrix}^{-1} \begin{pmatrix} y_1 \\ y_2 \\ y_4 \end{pmatrix}$$

# Reconstructing secret

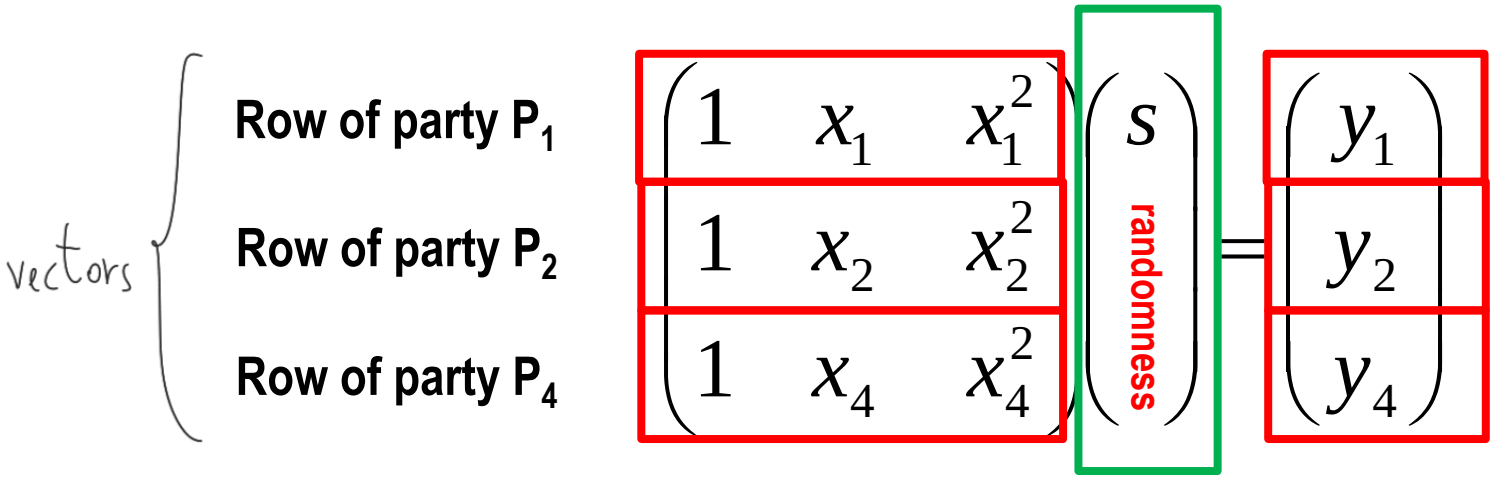
$$\begin{pmatrix} 1 & x_1 & x_1^2 \\ 1 & x_2 & x_2^2 \\ 1 & x_4 & x_4^2 \end{pmatrix}^{-1} = \begin{pmatrix} \frac{x_2 x_4}{(x_1 - x_2)(x_1 - x_4)} & \frac{x_1 x_4}{(x_2 - x_1)(x_2 - x_4)} & \frac{x_1 x_2}{(x_4 - x_1)(x_4 - x_2)} \\ \dots & \dots & \dots \\ \dots & \dots & \dots \end{pmatrix}$$

$$s = y_1 \frac{x_2 x_4}{(x_1 - x_2)(x_1 - x_4)} + y_2 \frac{x_1 x_4}{(x_2 - x_1)(x_2 - x_4)} + y_4 \frac{x_1 x_2}{(x_4 - x_1)(x_4 - x_2)}$$

esistono altri modi?

#1 polinomiale shamir
#2 linear system
#3 ... ?

# An alternative interpretation




**Each row = vector**

**$v_1$, $v_2$, $v_3$**

**Such vectors "span" a 3D space**

**$c_1v_1+c_2v_2+c_3v_3$**

$$c_1\begin{pmatrix} 1 \\ x_1 \\ x_1^2 \end{pmatrix} + c_2\begin{pmatrix} 1 \\ x_2 \\ x_2^2 \end{pmatrix} + c_3\begin{pmatrix} 1 \\ x_4 \\ x_4^2 \end{pmatrix}$$

solo se vale questa condizione il segreto può essere ricostruito !!

# An alternative interpretation

➔**Vector [1,0,0] included in "span"**

⇨Exists linear combination

$c_1v_1+c_2v_2+c_3v_3=[1,0,0]$

moltiplicato

$$c_1\begin{pmatrix}1 & x_1 & x_1^2\end{pmatrix}+c_2\begin{pmatrix}1 & x_2 & x_2^2\end{pmatrix}+c_3\begin{pmatrix}1 & x_4 & x_4^2\end{pmatrix}=\begin{pmatrix}1 & 0 & 0\end{pmatrix}$$

➔**Found by solving different linear system**

➔**Result… guess what ☺**

$$\begin{pmatrix}1 & 1 & 1\\ x_1 & x_2 & x_4\\ x_1^2 & x_2^2 & x_4^2\end{pmatrix}\begin{pmatrix}c_1\\ c_2\\ c_3\end{pmatrix}=\begin{pmatrix}1\\ 0\\ 0\end{pmatrix}$$

$$c_1=\frac{x_2x_4}{(x_1-x_2)(x_1-x_4)};\quad c_2=\frac{x_1x_4}{(x_2-x_1)(x_2-x_4)};\quad c_3=\frac{x_1x_2}{(x_4-x_1)(x_4-x_2)}$$

# An alternative interpretation

**But…**

appena visto!

[1, 0, 0]

$$\left\{c_1\begin{pmatrix}1 & x_1 & x_1^2\end{pmatrix}+c_2\begin{pmatrix}1 & x_2 & x_2^2\end{pmatrix}+c_3\begin{pmatrix}1 & x_4 & x_4^2\end{pmatrix}\right\}\begin{pmatrix}s\\ r_1\\ r_2\end{pmatrix}=\begin{pmatrix}1 & 0 & 0\end{pmatrix}\begin{pmatrix}s\\ r_1\\ r_2\end{pmatrix}=s$$

$$c_1\begin{pmatrix}1 & x_1 & x_1^2\end{pmatrix}\begin{pmatrix}s\\ r_1\\ r_2\end{pmatrix}+c_2\begin{pmatrix}1 & x_2 & x_2^2\end{pmatrix}\begin{pmatrix}s\\ r_1\\ r_2\end{pmatrix}+c_3\begin{pmatrix}1 & x_4 & x_4^2\end{pmatrix}\begin{pmatrix}s\\ r_1\\ r_2\end{pmatrix}=s$$

share

$$c_1y_1+c_2y_2+c_3y_4=s$$

$y_2$ $y_3$

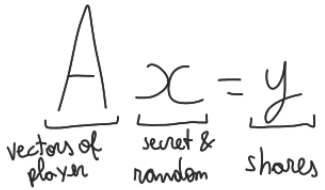
infatti abbiamo visto che:

$$\begin{bmatrix}1 & x_1 & x_1^2\\ 1 & x_2 & x_2^2\\ \vdots & & \end{bmatrix}\begin{pmatrix}s\\ r_1\\ r_2\end{pmatrix}=\begin{pmatrix}y_1\\ y_2\\ \vdots\end{pmatrix}$$

se trovo 3 vettori che hanno span [100] ok, senno non risolvibile!

**SUMMARY: shamir secret sharing treated as "span" problem**

# Much more general!




➔ **Linear Secret Sharing Scheme (LSSS)**

⇨Matrix can be ARBITRARY!

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \\ a_{41} & a_{42} & a_{43} \end{pmatrix} \begin{pmatrix} s \\ r_1 \\ r_2 \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{pmatrix}$$

➔ **Equivalence with (monotone) "span programs"**

➔ **Amos Beimel theorem: LSSS = MSP** (span problem)

# Trivial Secret Share is LSSS

## ➔Example: (3,3)

implementa

$$\begin{matrix} P_1 \\ P_2 \\ P_3 \end{matrix}\begin{pmatrix} 1 & -1 & -1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}\begin{pmatrix} s \\ r_1 \\ r_2 \end{pmatrix} = \begin{pmatrix} s - r_1 - r_2 \\ r_1 \\ r_2 \end{pmatrix}$$

## ➔Span program

trovare $\langle c_1, c_2, c_3 \rangle$ :

- vectors = partecipanti

$$c_1(1 \quad -1 \quad -1) + c_2(0 \quad 1 \quad 0) + c_3(0 \quad 0 \quad 1) = (1 \quad 0 \quad 0)$$

$$c_1 = c_2 = c_3 = 1$$

- se avessi solo due vettori? $c_1(1,-1,-1) + c_2(0,1,0) = (1,0,0)$ NO SOLUTION!

## ➔Share rec.

$$c_1(s - r_1 - r_2) + c_2 r_1 + c_3 r_2 = (s - r_1 - r_2) + r_1 + r_2 = s$$

# Any LSSS is homomorphic

mascondo

$$x_a = (s_a, r_{1a}, r_{2a}, \cdots) \quad y_a = (share_{1a}, share_{2a}, \cdots)$$

mascondo

$$x_b = (s_b, r_{1b}, r_{2b}, \cdots) \quad y_b = (share_{1b}, share_{2b}, \cdots)$$

$Ax_a = y_a$ dealer per secret 'a'

$Ax_b = y_b$ dealer per secret 'b'

shares of the sum, prima solo shamir, ora esteso!

sum of shares

$$y_a + y_b = Ax_a + Ax_b = A(x_a + x_b) = A(s_a + s_b, rand, rand, \cdots)$$

Se avessi $P_4$ ? schema (3,4) → NO threshold!

Se $P_4 = (0, 1, 1)$ → $y_4 = r_1 + r_2$, se avessi SOLO $P_1$ e $P_3$ : $c_1P_1 + c_4P_4 = (1,0,0)$ sarebbe

$c_1 \langle 1, -1, -1 \rangle + c_4 \langle 0, 1, 1 \rangle$, se $c_1 = c_4 = 1$ trovo $\langle 1, 0, 0 \rangle$ ✓

Ho ricostruito sottospazio dove c'è $\begin{bmatrix} 1 \\ 0 \end{bmatrix}$, non spazio 3D. Ora è POLICY BASE, $P_4$ può essere "più forte" di $P_2$ e $P_3$!

Qui posso usare $(P_1 \wedge P_2 \wedge P_3) \vee (P_1 \wedge P_4)$

# Monotone Span Programs

## (example in GF2, other fields OK)

aritmetica 0/1 : come XOR !

· secret : 1 bit : 0/1
· 5 partecipanti, 4 dimensioni

ho 2 vettori

| | | | | |
|---|---|---|---|---|
| $P_2$ | 1 | 1 | 0 | 1 |
| $P_2$ | 0 | 1 | 1 | 0 |
| $P_1$ | 0 | 1 | 1 | 0 |
| $P_3$ | 1 | 1 | 0 | 0 |
| $P_4$ | 0 | 0 | 1 | 1 |
| | 1 | 0 | 0 | 0 |

quali combinazioni danno questo risultato?

regola:

The program accepts a set *B* partecipante
iff
the rows labeled by *B* span the target vector. (cioè se posso ottenere quel result)

# Monotone Span Programs

| | | | | |
|---|---|---|---|---|
| $P_2$ | 1 | 1 | 0 | 1 |
| $P_2$ | 0 | 1 | 1 | 0 |
| $P_1$ | 0 | 1 | 1 | 0 |
| $P_3$ | 1 | 1 | 0 | 0 |
| $P_4$ | 0 | 0 | 1 | 1 |
| | 1 | 0 | 0 | 0 |

| | | | |
|---|---|---|---|
| 1 | 0 | 1 | 1 |

⊕

| | | | |
|---|---|---|---|
| 0 | 0 | 1 | 1 |

| | | | |
|---|---|---|---|
| 1 | 0 | 0 | 0 |

{P2,P4} accettata

# Monotone Span Programs

| | | | | |
|---|---|---|---|---|
| $P_2$ | 1 | 1 | 0 | 1 |
| $P_2$ | 0 | 1 | 1 | 0 |
| $P_1$ | 0 | 1 | 1 | 0 |
| $P_3$ | 1 | 1 | 0 | 0 |
| $P_4$ | 0 | 0 | 1 | 1 |
| | 1 | 0 | 0 | 0 |

| | | | |
|---|---|---|---|
| 1 | 1 | 0 | 1 |
| 0 | 1 | 1 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 |

{P1,P2}


Con P2 e P4 avremmo potuto pensare che la soglia thresold fosse "2", ma in questo nuovo esempio capiamo che non dobbiamo parlare di thresold, bensì di "policy", dove alcuni partecipanti possono essere più "importanti di altri"

rifiutata per Policy, no threshold!

# Span Programs → Secret Sharing

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| $P_2$ | 1 | 1 | 0 | 1 | $s$ | | $s+r_2+r_4$ | P2 |
| $P_2$ | 0 | 1 | 1 | 0 | $r_2$ | = | $r_2+r_3$ | P2 |
| $P_1$ | 0 | 1 | 1 | 0 | $r_3$ | | $r_2+r_3$ | P1 |
| $P_3$ | 1 | 1 | 0 | 0 | $r_4$ | | $s+r_2$ | P3 |
| $P_4$ | 0 | 0 | 1 | 1 | | | $r_3+r_4$ | $P^4$ |

share da dare

Example $s$=1,$r2$=$r3$=0, $r4$=1

(possono essere 1 o 0)

| | |
|---|---|
| 0 | P2 |
| 0 | P2 |
| 0 | P1 |
| 1 | P3 |
| 1 | $P^4$ |

Slides from A. Beimel

# Span Programs ⇒ Secret Sharing

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| $P_2$ | 1 | 1 | 0 | 1 | | $s$ | | $s+r_2+r_4$ | P2 |
| $P_2$ | 0 | 1 | 1 | 0 | | $r_2$ | | $r_2+r_3$ | P2 |
| $P_1$ | 0 | 1 | 1 | 0 | = | $r_3$ | = | $r_2+r_3$ | P1 |
| $P_3$ | 1 | 1 | 0 | 0 | | $r_4$ | | $s+r_2$ | P3 |
| $P_4$ | 0 | 0 | 1 | 1 | | | | $r_3+r_4$ | $P^4$ |
| | 1 | 0 | 0 | 0 | | | | $s$ | |

{P2,P4}

$P_2 + P_2 + P_4 = s + r_2 + r_4 + r_2 + r_3 + r_3 + r_4$



Slides from A. Beimel

# LSSS and access structure

posso dire chi deve esserci, non posso dire chi NON deve esserci!

➔ **Every (monotone) access structure can be realized**

$$\text{participan ts}: P = \{P_1, P_2, P_3, P_4\}$$

$$\text{monotone access structure}: A \subseteq 2^P$$

valido anche per Shamir

$$example: \overset{\text{secret}}{A} = \{\{\overset{\text{presenti}}{P_1}, P_2\}, \{\overset{\text{presenti}}{P_1}, P_3, P_4\}\}$$ vale anche $\{P_1, P_2P_3P_4\}$

➔ **Consequence: every boolean predicate (without negation) may be supported**

$$example: P_1 \wedge (P_2 \vee (P_3 \wedge P_4))$$ POLICY da tradurre in S.S.S.

➔ **Scheme May NOT be ideal**

⇨ May entail more than 1 share per partner

dipende da come scrivo la policy!

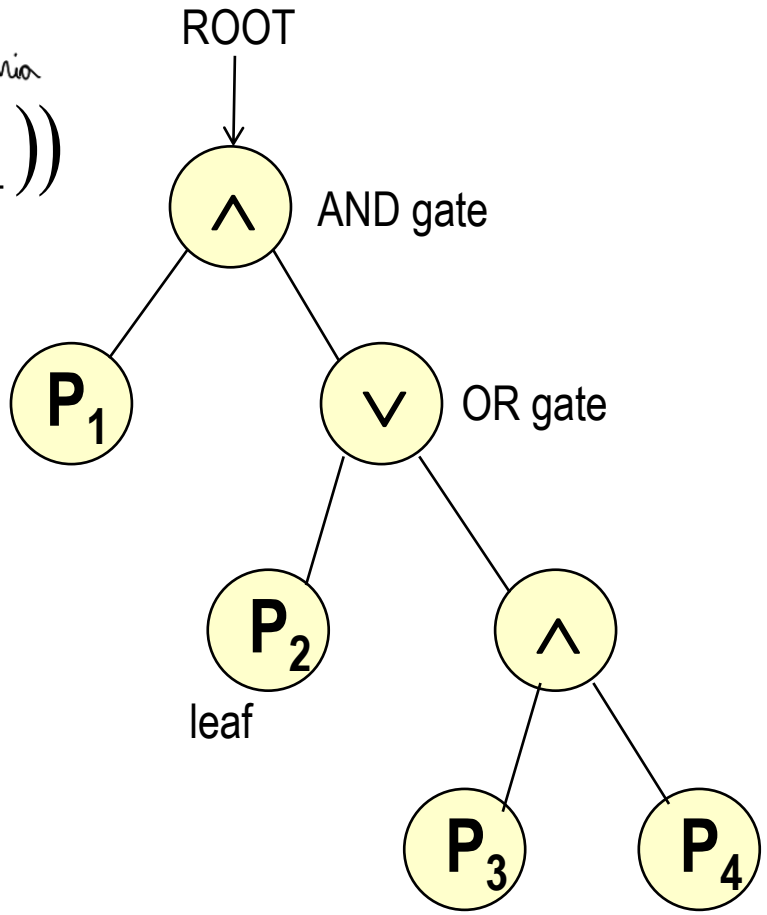
# LSSS matrix from AC Predicate

scrivo come 'root'

Giorgio Marco Giuseppe Maria

$$P_1 \wedge \left(P_2 \vee \left(P_3 \wedge P_4\right)\right)$$

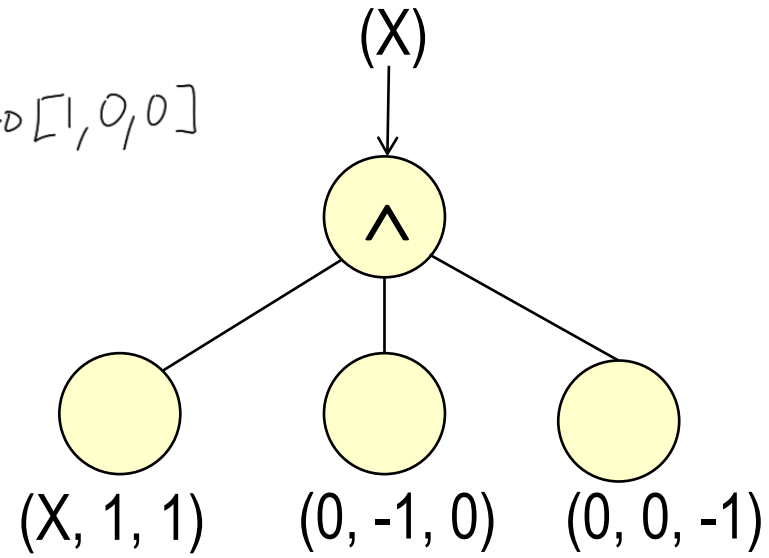
qual è la matrice che rispetta la policy?



Mi concentro: **AND gate**, come implementarlo?

vanno anche bene:

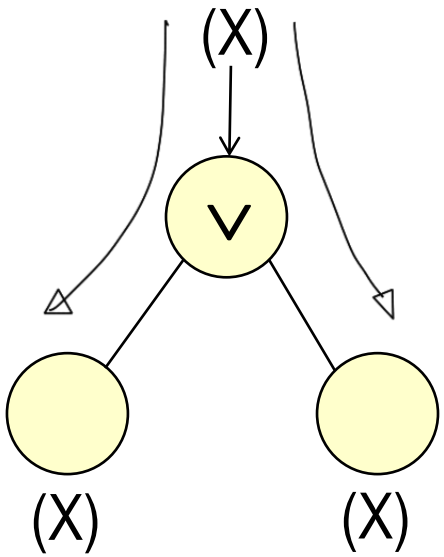
$$\begin{bmatrix} x & -1 & 1 \\ & 1 & \\ & & 1 \end{bmatrix} \begin{matrix} c_1 = 1 \\ c_2 = 1 \\ c_3 = -1 \end{matrix} \rightarrow [1, 0, 0]$$




$$\begin{pmatrix} X & 1 & 1 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{pmatrix}$$

solo avendo tutti e 3 pono ricostruire il segreto! (3,3) S.S.

# OR gate

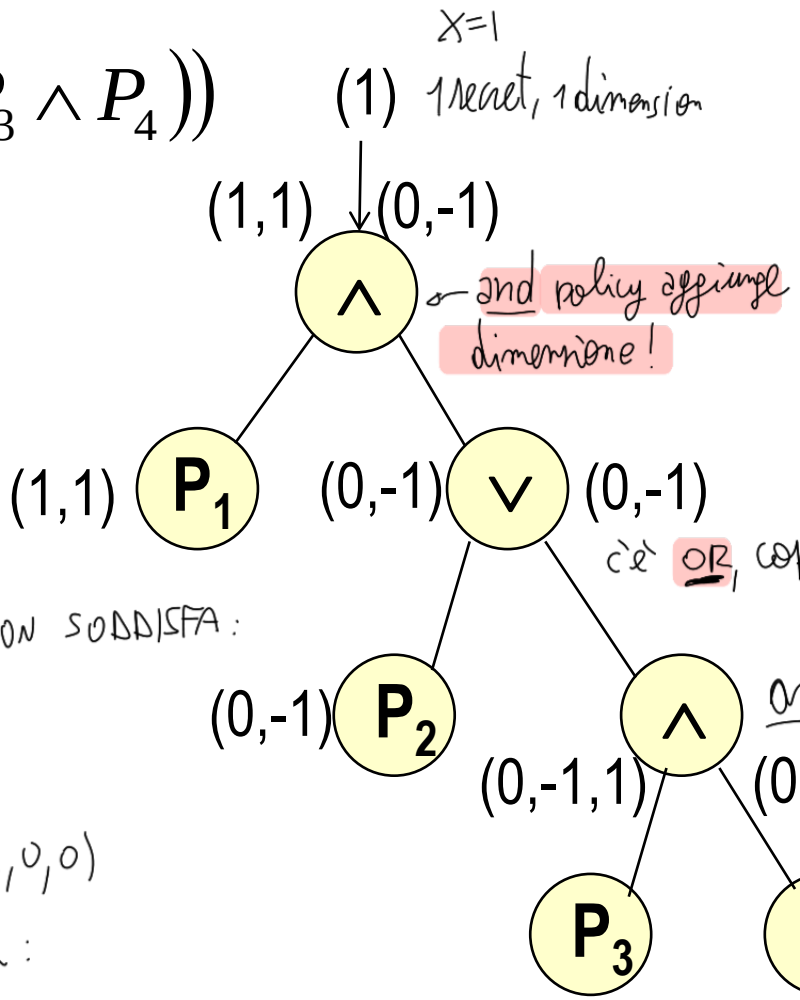



$$\begin{pmatrix} X \\ X \end{pmatrix}$$

do il 'segreto' od entrambi

# Matrix construction

$$P_1 \wedge \left(P_2 \vee \left(P_3 \wedge P_4\right)\right)$$




matrix A

| | | | |
|---|---|---|---|
| $P_1$ | 1 | 1 | 0 |
| $P_2$ | 0 | -1 | 0 |
| $P_3$ | 0 | -1 | 1 |
| $P_4$ | 0 | 0 | -1 |

Verifico A:

- nella policy, $P_1$ e $P_3$ NON SODDISFA:

$\nexists\ c_1, c_2$ :

$c_1 P_1 + c_2 P_2 = (1,0,0)$

- $P_1$ e $P_2$ soddisfa:

$c_1 = c_2 = 1 \rightarrow P_1 + P_2 = (1,0,0)$

$(0,-1,1) + (0,0,-1) = 0,-1,0$

deve corrispondere al ramo precedente!

# Careful with padding...

$$(A \wedge C \wedge D) \vee (B \wedge C)$$

$$\begin{pmatrix} A & 1 & -1 & -1 & . \\ C & 0 & 1 & 0 & . \\ D & 0 & 0 & 1 & . \\ B & 1 & -! & . & -1 \\ C & 0 & +! & . & 1 \end{pmatrix}$$

5 vectors, 2 sono di C

sarebbe stato ideale con: $C \wedge ([A \wedge D] \vee [B])$, con 4 vettori !! $\forall P_i$ ha $x_i$ vettori $\acute{x}_i$ dipende da quante volte è presente $P_i$ nella policy

sarebbe stato errore mantenere lui e aggiungere -1, 1 vicino $\forall$ end-gate extra dimension

# Wrap up

- **LSSS/MSP: generalization to arbitrary access structures**
  - ⇨ Must be monotone
    - → If parties A+B may access, also A+B+C may
    - → Cannot model policies such as A+B+NOT(C)
      - » Issue when revocation needed
- **Sub-optimal**
  - ⇨ Parties may need more than 1 share
  - ⇨ Minimum overhead: open research issue
- **Improved constructions**
  - ⇨ Explicit threshold gates [Liu,Cao, 2010], but prime fields…
- **Applications**
  - ⇨ Dramatic! Most modern crypto
    - → E.g. attribute based cryptography

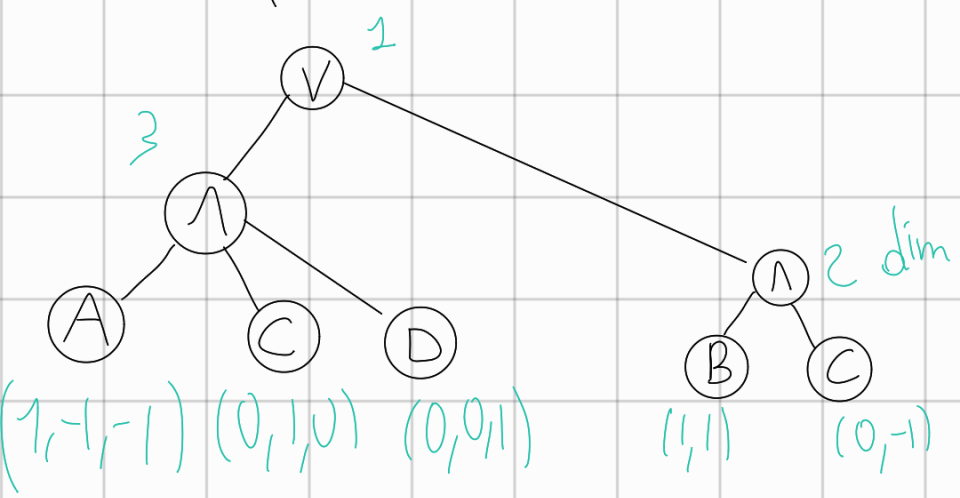
Con policy $(A \wedge B) \vee (B \wedge C) \vee (C \wedge A)$ (scheme 2,3)

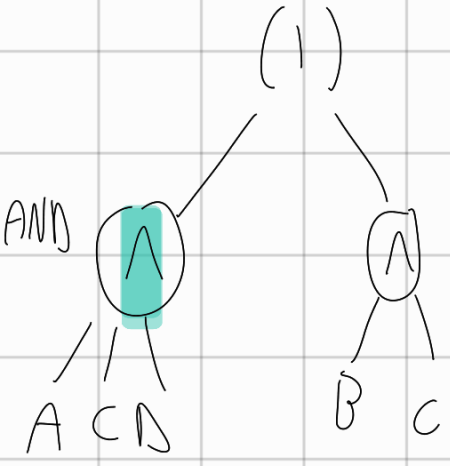
Non posso creare ideal scheme con AND/OR gate, ma con Shamir potrei! Per questo introduco threshold gate

Con policy $(A \wedge B) \vee (B \wedge C) \vee (C \wedge A)$ (scheme 23)

Non posso creare ideal scheme con AND/OR gate, ma con Shamir potrei! Per questo introduco threshold gate

$$(A \wedge C \wedge D) \vee (B \wedge C)$$




faccio 1° AND: parto da 1 dim, ho `2` $\wedge$ → dim = 3

Ma ∃ 2° AND: dim + 1

| | | | | |
|---|---|---|---|---|
| A | 1 | -1 | -1 | 0 |
| C | 0 | 1 | 0 | 0 |
| D | 0 | 0 | 1 | 0 |
| B | 1 | 0 | 0 | -1 |
| C | 0 | 0 | 0 | +1 |

∀ end gate extra dimension!

→ estensione di B, C

4° dimensione

estensione dovuta da A, C, D