**Sicurezza su Reti** **Anno Acc. 1997-98**

PROF. ALFREDO DE SANTIS

# CRITTOGRAFIA VISUALE

*a cura di Margherita Bresco, Claudio Chimera e Michele de Rosa* *24 Luglio 1998*

## Indice

# 1 Introduzione

La crittografia visuale si occupa del problema di cifrare delle immagini (testi, appunti scritti a mano, figure, etc) in maniera perfettamente sicura in modo che queste possano essere decodificate direttamente dal sistema visivo umano. Il modello base consiste in una pagina stampata di testo cifrato (la quale può essere inviata via mail o fax) e un lucido, che rappresenta la chiave segreta, sul quale compaiono dei pixel che sembrano distribuiti in modo casuale. Il testo in chiaro viene rivelato sovrapponendo il lucido con la chiave, sulla pagina con il testo cifrato. Il sistema ricalca lo schema tipico di un crittosistema a chiave privata con la differenza che la sua semplicità ne permette l'uso senza alcuna conoscenza di crittografia e senza alcun bisogno di computazioni crittografiche. Inoltre essa è simile ad un one-time pad, se ciascuna pagina di testo in chiaro viene codificata con un lucido diverso (ossia con una chiave diversa). Dato che questo modello base può essere esteso ad una variante visuale del problema della condivisione di un segreto con soglia $(k, n)$, esponiamo le nozioni di base che riguardano un tale schema. In generale, un metodo a soglia $(k, n)$ è un metodo per condividere un segreto $S$ tra un'insieme di $n$ partecipanti in modo che :

- ogni $k$ partecipanti possono computare il valore di $S$;
- nessun gruppo di $q$ partecipanti, dove $q < k$, possono computare alcuna informazione circa il valore di $S$.

Il segreto è conosciuto da un partecipante speciale $D$, detto dealer. Quando $D$ divide il segreto $S$ tra gli $n$ partecipanti, egli dà a ciascun partecipante un'informazione parziale detta **share**. Le share vengono distribuite in modo sicuro, così che nessun partecipante conosca la share data ad un altro partecipante. Sia $P$ l'insieme dei partecipanti e sia $B \subseteq P$ il sottoinsieme dei partecipanti che, successivamente, uniscono le loro share tentando di ricostruire il segreto $S$. Se $|B| \geq k$, allora essi sono in grado di ricostruire il segreto $S$ dalle share che essi posseggono. Se $|B| < k$, allora essi non sono in grado di ricostruire $S$, nè di risalire ad alcuna informazione riguardante $S$. Successivamente è stata introdotta l'idea della condivisione di segreto per strutture di accesso generali. Una struttura di accesso consiste di tutti i sottoinsiemi di partecipanti che sono abilitati alla ricostruzione del segreto. Analogamente a quanto esposto per la condivisione dei segreti, uno schema di crittografia visuale a soglia $(k, n)$ è un metodo per codificare un'immagine segreta $SI$ in $n$ immmagini "ombra", le share, tali che ogni $k$ o più share permettano il recupero visuale dell'immagine segreta, ma dall' ispezione di meno di $k$ share non si ottiene alcuna informazione sull'immagine segreta. In questo caso il "recupero visuale" consiste nella stampa delle share su lucidi e nella perfetta sovrapposizione di questi ultimi. Come primo esempio consideriamo uno schema visuale a soglia $(2, 2)$. In uno schema visuale a soglia $(2, 2)$ ci proponiamo di condividere un'immagine segreta tra due utenti. Tale immagine è costituita da pixel bianchi e neri che devono essere codificati singolarmente in maniera opportuna. L'insieme dei pixel codificati è detta share. Le due share vengono distribuite ai due partecipanti al segreto. Una prima idea potrebbe essere quella di codificare un pixel bianco dell'immagine originale con un singolo pixel bianco su ciascuna delle share, mentre un pixel nero può essere codificato in due modi, con un pixel nero sulla prima share ed uno bianco nella seconda o viceversa come mostrato in Figura 1.

Tale soluzione porta ad uno schema che non risulta essere sicuro, in quanto su entrambe le share si otterrebbero informazioni circa l'immagine originale.

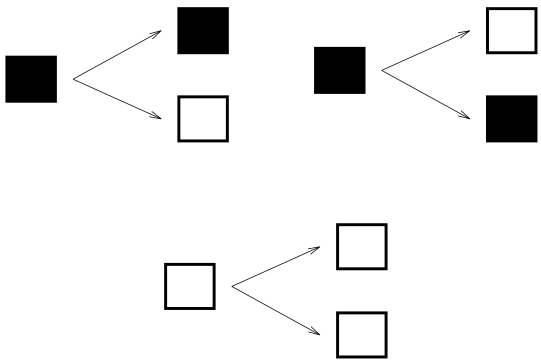
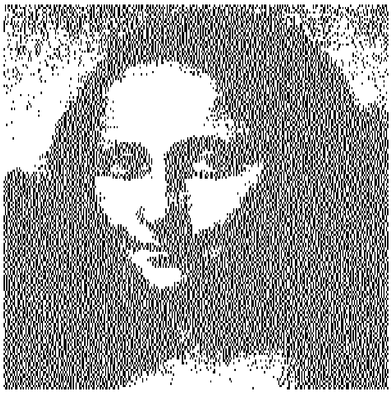


Figura 1: Soluzione di uno schema $(2,2)$.

**Esempio 1.1** *Scegliendo di codificare un pixel nel modo descritto precedentemente, allora si potrebbe ottenere la situazione mostrata in Figura 1.1:*

*Immagine originale*

*Share data al partecipante 1*



*Share data al partecipante 2*

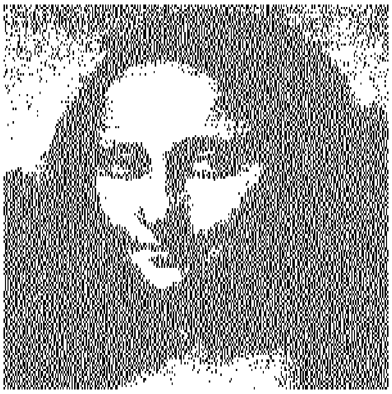


*Immagine risultante dalla sovrapposizione delle due share*

Notiamo che entrambe le share singolarmente rivelano l'immagine segreta, quindi un tale schema non è utilizzabile. A risolvere tale problema pensarono Naor e Shamir che in [8] proposero la costruzione di uno schema visuale a soglia $(2,2)$. Tale costruzione è mostrata in Figura 2:

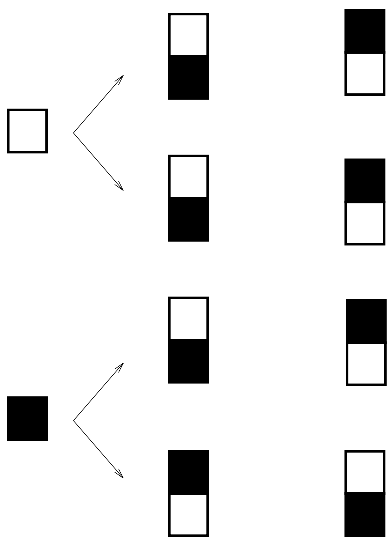


Figura 2: Soluzione di uno schema $(2,2)$.

Come si può notare ogni pixel $p$ è suddiviso in due sottopixel in ognuna delle due share. Se $p$ è un pixel bianco, allora casualmente viene scelta una delle prime due righe in figura 2. Il pixel $p$ è codificato con i due sottopixel della riga scelta in ogni share. Verifichiamo la sicurezza di tale schema. Siano $s_1$ ed $s_2$ le due share. Poniamo la nostra attenzione su $s_1$, come si può notare uno dei due sottopixel è nero e l'altro è bianco e, per come è stato costruito lo schema, le combinazioni bianco-nero e nero-bianco sono equamente probabili indipendentemente dal fatto che il corrispondente pixel nell'immagine segreta sia bianco o nero. Questo schema gode della proprietà di sicurezza perfetta, poichè nessuna informazione sull'immagine segreta può essere ricavata guardando le due share separatamente. Nello schema descritto, quando due share vengono sovrapposte, se il pixel $p$, nell'immagine originale, è nero, allora vedremo due sottopixel neri, mentre se $p$ è bianco vedremo un sottopixel bianco ed un sottopixel nero.

Uno svantaggio di tale soluzione è quello di non mantenere le proporzioni tra l'immagine originale e l'immagine ricostuita sovrapponendo le share. Per ovviare a tale problema, nella figura 3, proponiamo un esempio di realizzazione di uno schema $(2,2)$

Tutto ciò viene formalizzato definendo due matrici $S_0$ ed $S_1$:

$$S_0 = \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix} \qquad S_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

dove ogni matrice ha dimensione $n \times m$, con $n$ numero dei partecipanti ed $m$ espansione di un pixel. Ciascuna riga rappresenta la codifica di un pixel in una share, là dove uno rappresenta un sottopixel nero e zero un sottopixel bianco. È da notare che uno schema $(2,2)$ può essere costruito mediante una qualsiasi permutazione delle colonne di $S_0$ ed $S_1$. Dalle permutazione delle colonne della matrice $S_0$ ricaviamo tutte le possibili codifiche di un pixel bianco, mentre da quelle di $S_1$ ricaviamo tutte le possibili codifiche di un pixel nero. Nel nostro caso, poichè tutte le possibili permutazioni delle colonne sono $2! = 2$, avremo che questo è il numero di possibili codifiche di un pixel.

In generale gli schemi di crittografia visuale sono caratterizzati da due parametri :

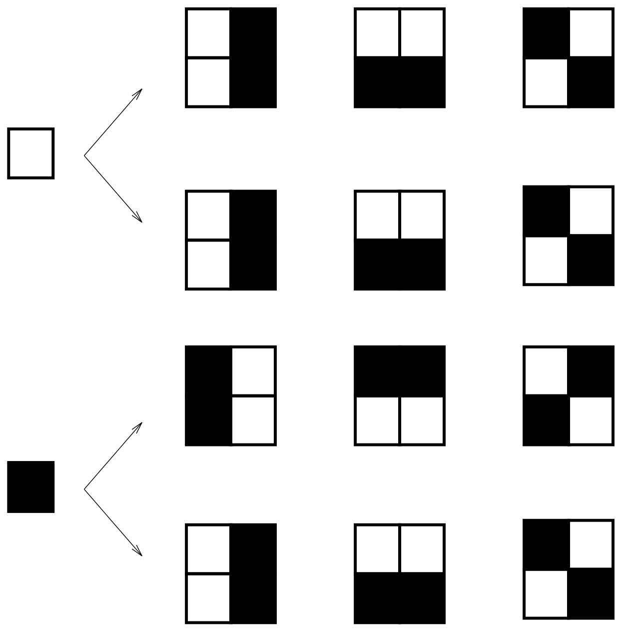


Figura 3: Soluzione di uno schema $(2, 2)$.

- l'**espansione dei pixel m** che è il numero di sottopixel necessari per codificare un pixel dell'immagine originale;
- il **contrasto** $\alpha$ che misura la differenza tra un pixel bianco ed uno nero nell'immagine ricostruita.

Nel caso di Figura 2 avremo un livello di grigio pari ad uno se un pixel è nero e un livello di grigio pari ad $\frac{1}{2}$ se $p$ è bianco,pertanto ci sarà una perdita di contrasto del 50% nell'immagine ricostruita.

Le nozioni qui esposte rappresentano una base informale per la conoscenza di un sistema di crittografia visuale e saranno approfondite nei successivi paragrafi.

# 2 VCS per strutture a soglia

La più semplice versione del problema di condivisione visuale del segreto (VCS), descritta da Naor e Shamir, assume che il messaggio consista di una collezione di pixel bianchi e neri ed ogni pixel venga manipolato separatamente. Ogni pixel dell'immagine originale appare in $n$ versioni modificate (chiamate share), una per ciascun lucido. Ogni share è una collezione di $m$ sottopixel bianchi e neri, che sono stampati uno vicino all'altro cosicchè il sistema visivo umano fonde i loro contributi individuali bianco/nero in modo da visualizzare un unico pixel di una certa tonalità di grigio.

## 2.1 Il modello

In questa sezione formalizzeremo il concetto del quale ci siamo occupati in modo informale. Come si é visto, ogni share é una collezione di $m$ sottopixel bianchi e neri, la cui sovrapposizione ricrea il pixel originario. La struttura risultante puó essere descritta da una matrice booleana $S = [s_{i,j}]$ di dimensioni $n \times m$, dove $s_{i,j} = 1$ se e solo se il $j$-esimo sottopixel nell'$i$-esimo

lucido è nero. Quando i lucidi $i_1, i_2, \cdots, i_r$ sono sovrapposti, il nostro sistema visivo percepisce una share che è la combinazione di sottopixel neri di ciascun lucido. Tale combinazione è equivalente all'or booleano delle righe $i_1, i_2, \cdots, i_r$ di $S$. Il livello di grigio risultante è proporzionale al peso di Hamming (cioè il numero di 1) del vettore $V$, che rappresenta l'or booleano di $i_1, i_2, \cdots, i_r$. Detto $H(V)$ tale peso, il livello di grigio è interpretato dal sistema visivo come nero se $H(V) \geq d$ e come bianco se $H(V) < d - \alpha m$ per una data soglia $d$ tale che $1 \leq d \leq m$ e differenza relativa $\alpha > 0$. Intuitivamente, non è possibile avere uno schema nel quale un pixel bianco è rappresentato da una collezione di sottopixel completamente bianchi e un pixel nero è rappresentato da una collezione di sottopixel completamente neri, così dobbiamo usare una soglia $d$ e una differenza relativa $\alpha > 0$ per distinguere tra i due colori.
Per costruire una soluzione per uno schema $(k, n)$, definiamo la collezione di matrici $C_0$ e $C_1$.

**Definizione 2.1** *Una soluzione allo schema $(k, n)$ consiste di due collezioni di matrici booleane, $C_0$ e $C_1$, di dimensione $n \times m$. Per condividere un pixel bianco il dealer sceglie casualmente una delle matrici in $C_0$ e per condividere un pixel nero il dealer sceglie casualmente una delle matrici in $C_1$. La matrice scelta definisce il colore degli $m$ sottopixel in ciascuno degli $n$ lucidi. La soluzione è considerata valida se le seguenti tre condizioni sono verificate:*

1. *per ogni matrice $S$ in $C_0$, il vettore $V$ risultante dall'or di un qualsiasi sottoinsieme di $k$ righe delle $n$ soddisfa la relazione $H(V) < d - \alpha \cdot m$;*

2. *per ogni matrice $S$ in $C_1$, il vettore $V$ risultante dall'or di un qualsiasi sottoinsieme di $k$ righe delle $n$ soddisfa la relazione $H(V) \geq d$;*

3. *per ogni sottoinsieme $\{i_1, i_2, \cdots, i_q\}$ di $\{1, 2, \cdots, n\}$ con $q < k$, le due collezioni di matrici $D_t$ per $t \in \{0, 1\}$, di dimensioni $q \times m$, ottenute dalla restrizione delle matrici in $C_0$ e $C_1$, di dimensione $n \times m$, alle righe $i_1, i_2, \cdots, i_q$ sono indistinguibili, nel senso che esse contengono le stesse matrici con le stesse frequenze.*

Le condizioni 1 e 2 servono ad assicurare almeno una differenza di contrasto $\alpha$ tra un pixel bianco ed uno nero nell'immagine ricostruita. La condizione 3 implica che, ispezionando meno di $k$ share, un crittoanalista con a disposizione una potenza computazionale infinita non riuscirebbe a stabilire se un pixel dell'immagine originale è bianco o nero. I parametri importanti di uno schema sono:

- $m$, il numero di sottopixel in una share. Tale valore rappresenta la perdita in risoluzione dell'immagine originale rispetto a quella condivisa. Vorremmo che $m$ sia il più piccolo possibile;

- $\alpha$, la differenza relativa, in peso, fra le share combinate derivanti da un pixel bianco e quelle derivanti da un pixel nero dell'immagine originale. Tale valore rappresenta la perdita in contrasto e vorremmo che $\alpha$ sia il più grande possibile;

- $r$, la dimensione delle collezioni $C_0$ e $C_1$. $\log_2 r$ rappresenta il numero di bit casuali necessari a generare le share.

## 2.2 Soluzioni efficienti per piccoli k e n

Lo schema $(2,2)$ visto in Figura 2 può essere generalizzato, ora, ad uno schema $(2,n)$, mediante l'ausilio delle seguenti collezioni di matrici di dimensione $n \times n$:

$$C_0 = \{\text{tutte le matrici ottenute dalla permutazione delle colonne di } \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 1 & 0 & 0 & \cdots & 0 \\ 1 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & 0 & \cdots & 0 \end{bmatrix} \}$$

$$C_1 = \{\text{tutte le matrici ottenute dalla permutazione delle colonne di } \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix} \}$$

Ogni riga di una matrice rappresenta un sottopixel nero ed $n-1$ sottopixel bianchi. Notiamo che la sovrapposizione di due righe qualsiasi di una matrice appartenente a $C_0$, dà come risultato un sottopixel nero ed $n-1$ sottopixel bianchi. Viceversa la sovrapposizione di due righe qualsiasi di una matrice appartenente a $C_1$, dà come risultato due sottopixel neri ed $n-2$ sottopixel bianchi. Dunque nel primo caso interpreteremo il pixel risultante come bianco, mentre nel secondo caso come nero.
Una soluzione al problema di un VCS $(3,3)$ è data dalle seguenti collezioni di matrici:

$$C_0 = \{ \text{ tutte le matrici ottenute dalla permutazione delle colonne di } \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix} \}$$

$$C_1 = \{ \text{ tutte le matrici ottenute dalla permutazione delle colonne di } \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix} \}$$

Le sei share definite dalle righe di $C_0$ e $C_1$ sono date in figura 4; e sono costruite a partire dall'angolo superiore sinistro e procedendo in senso orario.

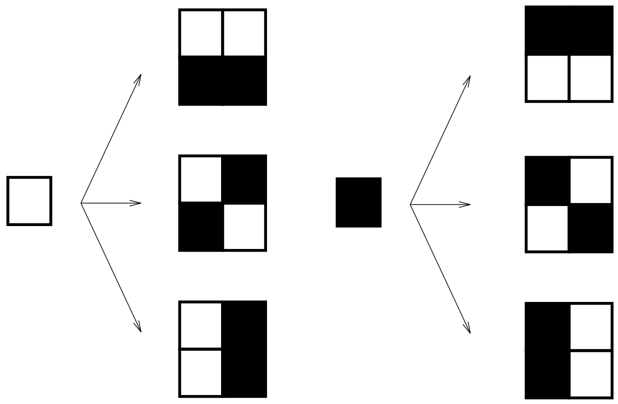


Figura 4: Soluzione di uno schema $(3,3)$.

Come si può notare sovrapponendo tre share da $C_0$, otteniamo tre sottopixel neri su quattro, mentre effettuando la stessa operazione con $C_1$ otteniamo un pixel completamente nero.

Viceversa ogni share in $C_0$ e $C_1$ contiene due sottopixel neri e due bianchi e se sovrapponiamo solo due share otteniamo sia per $C_0$ che per $C_1$ un sottopixel bianco e tre sottopixel neri e, dunque, non riusciamo a distinguere il nero dal bianco.
Esiste un semplice metodo per generalizzare tale schema ad uno schema $(3, n)$, per $n \geq 3$. Sia $B$ la matrice, di dimensione $n \times (n-2)$, che contiene solo 1, cioè $B_{i,j} = 1$, $1 \leq i \leq n$ e $1 \leq j \leq n-2$ e sia $I$ la matrice identità di dimensione $n \times n$, cioè $S_{i,i} = 1$, $1 \leq i \leq n$ e $S_{i,j} = 0$, $i \neq j$. Denotiamo, ora, con $BI$ la matrice, di dimensione $n \times (2 \cdot n - 2)$, ottenuta concatenando $B$ ed $I$ e sia $c(BI)$ il complemento booleano di $BI$. Allora:

$C_0 = \{$ tutte le matrici ottenute permutando le colonne di $c(BI)$ $\}$

$C_1 = \{$ tutte le matrici ottenute permutando le colonne di $BI$ $\}$

Le matrici appartenenti a $C_0$ e $C_1$ godono delle seguenti proprietà :

- ogni singola share contiene un'arbitraria collezione di $n-1$ sottopixel neri ed $n-1$ sottopixel bianchi;
- ogni coppia di share ha $n-2$ sottopixel comuni neri e 2 sottopixel "spaiati" neri;
- ogni tripla di share sovrapposte prese da $C_0$ ha $n$ sottopixel neri, mentre ogni tripla di share sovrapposte prese da $C_1$ ha $n+1$ sottopixel neri.

**Esempio 2.2** *Supponiamo di voler costruire uno schema* $(3, 4)$ *utilizzando lo schema appena descritto. Poichè* $n = 4$, *allora* $B$ *avrà dimensioni* $4 \times 2$ *e* $I$ *avrà dimensioni* $4 \times 4$. *Seguendo lo schema si ottiene:*

$$B = \begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \end{bmatrix} \qquad I = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$S_1 = BI = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 \end{bmatrix} \qquad S_0 = c(BI) = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix}$$

*Quindi:*

$C_0 = \{$ *tutte le matrici ottenute permutando le colonne di* $S_0$ $\}$

$C_1 = \{$ *tutte le matrici ottenute permutando le colonne di* $S_1$ $\}$

## 2.3 Soluzione ottimale per schemi visuali a soglia $(k, k)$

In questa sezione costruiremo uno schema visuale con parametri $m = 2^{k-1}$ e $r = 2^{k-1}!$. Consideriamo un'insieme di base $W = \{e_1, e_2, \cdots, e_k,\}$ di $k$ elementi (per esempio $W = \{1, 2, \cdots, k\}$) e sia $\pi_1, \pi_2, \cdots, \pi_{2^{k-1}}$, la lista di tutti i sottoinsiemi di $W$ di cardinalità pari e sia $\sigma_1, \sigma_2, \cdots, \sigma_{2^{k-1}}$, la lista di tutti i sottoinsiemi di $W$ di cardinalità dispari (l'ordine non è importante). Le liste definiscono le seguenti matrici $S^0$ e $S^1$:

per $1 \leq i \leq k$ e $1 \leq j \leq 2^{k-1}$ sia $S^0[i,j] = 1$ se e solo se $e_i \in \pi_j$ e $S^1[i,j] = 1$ se e solo se $e_i \in \sigma_j$.

Come nelle costruzioni precedenti, le collezioni $C_0$ e $C_1$ sono ottenute permutantdo le colonne delle corrispondenti matrici, $S^0$ e $S^1$.

**Lemma 2.3** *Il precedente schema è uno schema $(k,k)$ con parametri $m = 2^{k-1}$, $\alpha = 1/2^{k-1}$ e $r = 2^{k-1}!$.*

L'ottimalità dello schema precdente è data dal seguente teorema, dimostrato da Naor e Shamir:

**Teorema 2.4** *In ogni schema a soglia $(k,k)$ si ha $\alpha \leq 1/2^{k-1}$ e $m \geq 2^{k-1}$.*

Per schemi visuali $(k,n)$, Naor e Shamir hanno dimostrato il seguente teorema:

**Teorema 2.5** *Per ogni $n$ e $k$ esiste uno schema visuale a soglia $(k,n)$ , per la condivisione dei segreti, con $m = \log n \cdot 2^{O(k \log k)}$ e $\alpha = 2^{-\Omega(k)}$.*

Attualmente non si conosce se il limite su $m$ nel precedente teorema è stretto, Naor e Shamir hanno congetturato che $m = \log n \cdot 2^{O(k)}$

## 2.4 Un esempio di share per strutture a soglia

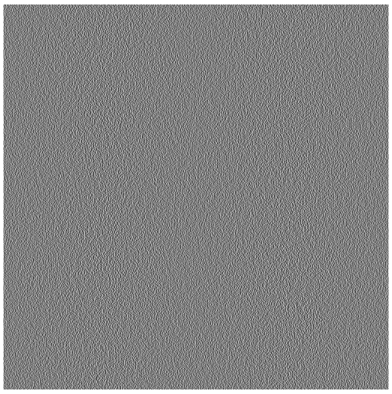
Diamo, ora, un esempio visivo di un VCS per strutture a soglia $(2,2)$ che utilizza le seguenti matrici di base :

$$S_0 = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \end{bmatrix} \quad S_1 = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \end{bmatrix}$$

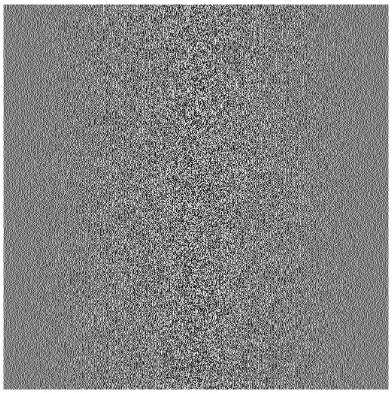
Le share risultanti sono le seguenti :

Immagine originale

Share data al partecipante 1



Share data al partecipante 2 Immagine risultante dalla sovrapposizione delle due share



# 3 VCS per strutture di accesso generali

Sinora ci siamo occupati di strutture a soglia $(k, k)$ e $(k, n)$ per le quali la ricostruzione dell'immagine segreta era possibile mediante la sovrapposizione di $k$ share qualsiasi. Ora vogliamo introdurre uno schema che permetta la ricostruzione dell'immagine segreta solo da parte di alcuni sottoinsiemi di partecipanti.

## 3.1 Il modello

Sia $\mathcal{P} = \{ 1, \cdots, n \}$ un insieme di partecipanti, tale insieme può essere partizionato in sottoinsiemi, denoteremo con $2^{\mathcal{P}}$ la famiglia di tali sottoinsiemi. Detto ciò denotiamo con $\Gamma_{qual} \subseteq 2^{\mathcal{P}}$ e con $\Gamma_{forb} \subseteq 2^{\mathcal{P}}$, $\Gamma_{qual} \cap \Gamma_{forb} = \Phi$, la famiglia dei sottoinsiemi di partecipanti qualificati a ricostruire il segreto e la famiglia dei sottoinsiemi di partecipanti non qualificati rispettivamente. La coppia $(\Gamma_{qual}, \Gamma_{forb})$ è detta struttura d'accesso dello schema.
Denotiamo, inoltre, con $\Gamma_0$ la famiglia minimale di tutti i sottoinsiemi qualificati:

$$\Gamma_0 = \{A \in \Gamma_{qual} : A' \notin \Gamma_{qual} \ \forall A' \subset A\}$$

**Definizione 3.1** *Un partecipante* $P \in \mathcal{P}$ *è un partecipante* **essenziale** *se esiste un'insieme* $X \subseteq \mathcal{P}$ *tale che* $X \cup \{P\} \in \Gamma_{qual}$ *ma* $X \notin \Gamma_{qual}$.

Se un partecipante non è essenziale, allora possiamo costruire un VCS dandogli una share completamente bianca. Infatti un partecipante non essenziale non ha bisogno di partecipare attivamente nella ricostruzione dell'immagine, poichè l'informazione che egli possiede non è necessaria a nessun insieme in $\mathcal{P}$ per recuperare l'immagine condivisa. Assumeremo che tutti i partecipanti siano essenziali.

**Definizione 3.2** *Una struttura di accesso* $(\Gamma_{qual}, \Gamma_{forb})$ *è detta* **forte** *se l'insieme* $\Gamma_{qual}$ *è monotono crescente e l'insieme* $\Gamma_{forb}$ *è monotono decrescente. In questo caso l'insieme* $\Gamma_0$ *è chiamato* **base**.

**Definizione 3.3** $\Gamma_{qual}$ *è detta* **monotona crescente** *se* $\forall X \in \Gamma_{qual}$, *e* $\forall p \in \mathcal{P} \setminus X \Rightarrow X \cup \{p\} \in \Gamma_{qual}$*;* $\Gamma_{forb}$ *è detta* **monotona decrescente** *se* $\forall X \in \Gamma_{forb}$, *risulta che* $\forall X' \subset X \Rightarrow X' \in \Gamma_{forb}$.

In strutture di accesso forti, $\Gamma_{qual} = \{C \subseteq \mathcal{P} : \exists\ B \subseteq C$ tale che $B \in \Gamma_0\}$, e diciamo che $\Gamma_{qual}$ è la chiusura di $\Gamma_0$.

**Esempio 3.4** *Supponiamo che $n = 4$ e che, dunque, $\mathcal{P} = \{\ 1, 2, 3, 4\ \}$. Definiamo:*

$$\Gamma_{qual} = \{\{1,2\},\{2,3\},\{3,4\},\{1,2,3\}\}$$

*e*

$$\Gamma_{forb} = \{\{1\},\{2\},\{3\},\{4\},\{1,3\},\{1,4\},\{2,4\}\}$$

*Da tali definizioni si ricava subito che:*

$$\Gamma_0 = \{\{1,2\},\{2,3\},\{3,4\}\}$$

**Definizione 3.5** *Sia $(\Gamma_{qual}, \Gamma_{forb})$ una struttura d'accesso su un insieme di $n$ partecipanti. Due collezioni di matrici booleane $C_0$ e $C_1$, di dimensione $n \times m$, costituiscono uno schema di crittografia visuale $(\Gamma_{qual}, \Gamma_{forb})$-VCS se esiste $\alpha(m)$ e un'insieme $\{(X, t_X)\}_{X \in \Gamma_{qual}}$ tale che:*

1. *ogni insieme qualificato $X = \{i_1, i_2, \cdots, i_p\} \in \Gamma_{qual}$ può recuperare l'immagine condivisa sovrapponendo le trasparenze dei partecipanti appartenenti all'insieme.*
   *Formalmente, per ogni $M \in \mathcal{C}_0$, l' "or" $V$ delle righe $i_1, i_2, \cdots, i_p$ di $M$ soddisfa $w(V) \leq t_X - \alpha(m) \cdot m$; mentre, per ogni $M \in \mathcal{C}_1$ risulta $w(V) \geq t_X$.*

2. *Ogni insieme non qualificato $X = \{i_1, i_2, \cdots, i_p\} \in \Gamma_{forb}$ non ha nessuna informazione sull'immagine condivisa.*
   *Formalmente, le due collezioni di matrici $\mathcal{D}_t$, con $t \in \{0, 1\}$, ottenute restringendo ogni matrice $n \times m$ in $\mathcal{C}_t$ alle righe $i_1, i_2, \cdots, i_p$ sono indistiunguibili, nel senso che contengono le stesse matrici con la stessa frequenza.*

Notiamo che se un'insieme di partecipanti $X$ contiene un insieme di partecipanti $X' \in \Gamma_{qual}$, cioè $X' \subset X$, i partecipanti possono recuperare l'immagine condivisa considerando solo le share dell'insieme $X'$. Questo non esclude la possibilità che i partecipanti in $X$, sovrapponendo tutte le loro trasparenze, non rileveranno nessuna informazione riguardo l'immagine originale.

## 3.2 Matrici di base

La maggior parte delle costruzioni che analizzeremo è fatta utilizzando due matrici $S_0$ e $S_1$, di dimensione $n \times m$, chiamate **matrici di base**, che soddisfano la seguente definizione:

**Definizione 3.6** *Sia $(\Gamma_{qual}, \Gamma_{forb})$ una struttura di accesso su un insieme si $n$ partecipanti. Un $(\Gamma_{qual}, \Gamma_{forb})$ - VCS, con differenza relativa $\alpha$ e insieme di soglie $\{(x, t_x)\}_{x \in \Gamma_{qual}}$ è realizzato utilizzando le due matrici di base, di dimensione $n \times m$, $S_0$ ed $S_1$ se le seguenti due condizioni sono verificate:*

1. *Se $X = \{i_1, i_2, \cdots, i_p\} \in \Gamma_{qual}$, cioè se $X$ è un insieme qualificato), allora l'or $V$ delle righe $i_1, i_2, \cdots, i_p$ di $S_0$ soddisfa la realzione $w(V) < t_x - \alpha \cdot m$, mentre per $S_1$ risulta che $w(V) \geq t_x$;*

2. *Se* $X = \{i_1, i_2, \cdots, i_p\} \in \Gamma_{forb}$ *(cioè se* $X$ *è un insieme non qualificato), allora le due matrici* $p \times m$*, ottenute restringendo* $S_0$ *e* $S_1$ *alle righe* $i_1, i_2, \cdots, i_p$ *sono indistinguibili (cioè sono uguali a meno di una permutazione delle righe).*

La seconda condizione formalizza il concetto di incapacità nel ricavare qualche informazione, circa l'immagine segreta, se vengono sovrapposti i lucidi appartenenti ad un insieme non qualificato.

**Esempio 3.7** *Supponiamo che* $n = 4$ *e che, dunque,* $\mathcal{P} = \{\ 1, 2, 3, 4\ \}$. *Definiamo:*

$$\Gamma_{qual} = \{\{1,2\}, \{2,3\}, \{3,4\}, \{1,2,3\}\}$$

*e*

$$\Gamma_{forb} = \{\{1\}, \{2\}, \{3\}, \{4\}, \{1,3\}, \{1,4\}, \{2,4\}\}$$

*Da tali definizioni si ricava subito che:*

$$\Gamma_0 = \{\{1,2\}, \{2,3\}, \{3,4\}\}$$

*le 2 matrici di base* $S_0$ *ed* $S_1$ *sono:*

$$S_0 \quad = \quad \begin{bmatrix} 1 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \end{bmatrix} \qquad S_1 \quad = \quad \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$$

*In questo schema* $\alpha = \frac{1}{3}$ *e dunque il contrasto risulta essere uguale a* 1. *Verifichiamo, ora, che le condizioni* 1 *e* 2 *sono verificate per* $\Gamma_{qual}$ *e* $\Gamma_{forb}$.

| *Partecipanti in* $\Gamma_{Qual}$ | $w(V)$ *in* $S_0$ | $w(V)$ *in* $S_1$ |
|---|---|---|
| $\{1,2\}$ | *2* | *3* |
| $\{2,3\}$ | *1* | *2* |
| $\{3,4\}$ | *2* | *3* |
| $\{1,2,3\}$ | *2* | *3* |
| *Partecipanti in* $\Gamma_{Forb}$ | $w(V)$ *in* $S_0$ | $w(V)$ *in* $S_1$ |
| $\{1\}$ | *2* | *2* |
| $\{2\}$ | *1* | *1* |
| $\{3\}$ | *1* | *1* |
| $\{4\}$ | *2* | *2* |
| $\{1,3\}$ | *2* | *2* |
| $\{1,4\}$ | *3* | *3* |
| $\{2,4\}$ | *2* | *2* |

*Ogni elemento della tabella rappresenta il peso di Hamming del vettore ottenuto dall'or delle righe corrispondenti ai partecipanti. Notiamo che per insiemi di partecipanti appartenenti a* $\Gamma_{qual}$ *tale peso differisce a seconda che ci riferiamo al nero o al bianco (*$S_1$ *o* $S_0$*). Viceversa tale peso non cambia nel caso che l'insieme dei partecipanti appartenga a* $\Gamma_{forb}$.

Le collezioni $C_0$ e $C_1$ sono ottenute permutando le colonne delle corrispondenti matrici di base ($S_0$ per $C_0$ ed $S_1$ per $C_1$) in tutti i modi possibili. Notiamo che, in questo caso, la cardinalità delle due collezioni è la stessa, cioè $m!$ ed essa è denotata con $r$. L'algoritmo per il VCS basato sulla precedente costruzione delle collezioni $C_0$ e $C_1$ richiede poca memoria (deve solo mantenere le matrici $S_0$ ed $S_1$) ed è efficiente (per scegliere una matrice in $C_0$ o $C_1$ genera una permutazione delle colonne di $S_0$ o $S_1$ rispettivamente).

## 3.3 Costruzioni generali

In questo paragrafo presenteremo due tecniche di costruzione per relizzare VCS per strutture generali.

### 3.3.1 Una costruzione per VCS che usa l'array cumulativo

La prima costruzione considerata è basata sul metodo dell'array cumulativo.
Sia $\Gamma = (\Gamma_{qual}, \Gamma_{forb})$ una struttura di accesso forte sull'insieme di partecipanti $\mathcal{P} = \{1, 2, \cdots, n\}$. Denotiamo con $Z_M$ la collezione degli insiemi massimali non qualificati di $\Gamma$:

$$Z_M = \{B \in \Gamma_{forb} : B \cup \{i\} \in \Gamma_{qual}\ \forall i \in \mathcal{P} \setminus B\}.$$

Una **mappa cumulativa** $(\beta, T)$ per $\Gamma_{qual}$ è un insieme finito $T$ e una mappa $\beta : \mathcal{P} \to 2^T$ tale che per $Q \subseteq \mathcal{P}$ abbiamo che :

$$\bigcup_{a \in Q} \beta(a) = T \Leftrightarrow Q \in \Gamma_{qual}.$$

Possiamo costruire $(\beta, T)$ per ogni $\Gamma_{qual}$ usando la collezione di insiemi massimali non qualificati $Z_M = \{F_1, \cdots, F_t\}$ nel modo seguente. Sia $T = \{T_1, \cdots, T_t\}$ e per ogni $i \in \mathcal{P}$ sia

$$\beta(i) = \{T_i \mid i \notin F_j, 1 \leq j \leq t\}.$$

È facile verificare che per ogni $X \in \Gamma_{qual}$ abbiamo

$$\bigcup_{i \in X} \beta(i) = T,$$

mentre in ogni insieme $X \in \Gamma_{forb}$ mancherà almeno un $T_j \in T$. Dalla mappa cumulativa $\beta(i)$ per $\Gamma_{qual}$, possiamo ottenere un **array cumulativo** per $\Gamma_{qual}$ nel modo seguente. Un array cumulativo è un matrice booleana di dimensioni $|\mathcal{P}| \times |T|$, denotata con $CA$, tale che $CA(i, j) = 1$ se e solo se $i \notin F_j$.

**Esempio 3.8** *Sia* $\mathcal{P} = \{1, 2, 3, 4\}$, $\Gamma_0 = \{\{1, 2\}, \{2, 3\}, \{3, 4\}\}$, $Z_M = \{\{1, 4\}, \{1, 3\}, \{2, 4\}\}$, *e siano* $F_1 = \{1, 4\}$, $F_2 = \{1, 3\}$ *e* $F_3 = \{2, 4\}$. *Da ciò si ricava* $T_1 = \{1, 2\}$, $T_2 = \{2, 3\}$ *e* $T_3 = \{3, 4\}$, *quindi* $|T| = 3$. *La mappa cumulativa risultante sarà , dunque, la seguente:*

$$\beta(1) = \{T_2, T_3\}, \quad \beta(2) = \{T_3\}, \quad \beta(3) = \{T_1\}, \quad \beta(4) = \{T_1, T_2\}$$

*L'array cumulativo, che ne deriva per* $\Gamma_{qual}$*, è il seguente:*

$$CA = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

A questo punto possiamo realizzare un VCS per qualsiasi struttura di accesso forte. La tecnica è basata sullo schema a soglia $(n, n)$. Sia $Z_M$ la famiglia degli insiemi massimali non qualificati e sia $T = |Z_M|$. Sia $CA$ l'array cumulativo per $\Gamma_{qual}$ ottenuto usando la mappa cumulativa. Siano $\hat{S}^0$ e $\hat{S}^1$ le matrici di base per un VCS a soglia $(t, t)$. Le matrici di base $S^0$ e $S^1$ per una struttura di accesso $(\Gamma_{qual},\ \Gamma_{forb})$ può essere costruita nel modo seguente:

> per ogni $i$ fissato siano $j_{i,1}, \cdots, j_{i,g}$ gli interi $j$ tali che $CA(i, j) = 1$. L'i-esima riga di $S^0$ ($S^1$ rispettivamente) consiste dell'or delle righe $j_{i,1}, \cdots, j_{i,g}$ di $\hat{S}^0$ ($\hat{S}^1$ rispettivamente). Un esempio aiuterà ad illustrare questa tecnica.

**Esempio 3.9** *Sia* $\mathcal{P} = \{1, 2, 3, 4\}$*,* $\Gamma_0 = \{\{1, 2\}, \{2, 3\}, \{3, 4\}\}$*,* $Z_M = \{\{1, 4\}, \{1, 3\}, \{2, 4\}\}$ *quindi* $|T| = 3$*. Siano* $\hat{S}^0$ *ed* $\hat{S}^1$*:*

$$\hat{S}^0 = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

$$\hat{S}^1 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix}$$

*Le matrici di base* $S_0$ *ed* $S_1$ *in un VCS con struttura di accesso forte con base* $\Gamma_0$ *sono:*

$$S^0 = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 \end{bmatrix}$$

$$S^1 = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix}$$

*La seconda riga di* $S^0$ *è l'or delle righe* 1 *e* 2 *di* $\hat{S}^0$*, cioè :*

$$[0, 1, 1, 1] = [0, 0, 1, 1] \ or\ [0, 1, 0, 1]$$

*e la terza riga di* $S^0$ *è l'or delle righe* 1 *e* 3 *di* $\hat{S}^0$*.*

### 3.3.2 Tecniche di composizione

Ora presenteremo una costruzione per VCS che utilizza piccoli schemi come blocchi elementari nella costruzione di schemi più grandi. Siano $(\Gamma'_{qual}, \Gamma'_{forb})$ e $(\Gamma''_{qual}, \Gamma''_{forb})$ due strutture di accesso su un insieme di $n$ partecipanti $\mathcal{P}$. Se un partecipante $i \in \mathcal{P}$ non è essenziale per $(\Gamma'_{qual}, \Gamma'_{forb})$, assumeremo che $i \in \Gamma'_{forb}$ e che $i$ riceve una share con nessuna informazione. Lo stesso vale per $(\Gamma''_{qual}, \Gamma''_{forb})$. Supponiamo che esiste $(\Gamma'_{qual}, \Gamma'_{forb}, m')$-VCS e un $(\Gamma''_{qual}, \Gamma''_{forb}, m'')$-VCS con matrici di base $R^0$, $R^1$ e $T^0$, $T^1$, rispettivamente. Mostreremo come costruire un VCS per una struttura di accesso $(\Gamma_{qual}, \Gamma_{forb}) = (\Gamma'_{qual} \cup \Gamma''_{qual}, \Gamma'_{forb} \cap \Gamma''_{forb})$. Dalle matrici $R^0$, $R^1$ e $T^0$, $T^1$ costruiamo due coppie di matrici $(\hat{R}^0, \hat{R}^1)$ e $(\hat{T}^0, \hat{T}^1)$, ognuna di $n$ righe nel modo seguente. Mostriamo prima come costruire $\hat{R}^0$. Per $i = 1, \cdots, n$, l'$i$-ma riga di $\hat{R}^0$ ha tutti zeri se il partecipante $i$ non è un partecipante essenziale per $(\Gamma'_{qual}, \Gamma'_{forb})$; altrimenti è la riga di $R^0$ corrispondente al partecipante $i$. Le matrici $\hat{R}^1$ ,$\hat{T}^0$ e $\hat{T}^1$ sono costruite analogamente. Infine, la matrice di base $S^0$ ($S^1$ rispettivamente) per $(\Gamma_{qual}, \Gamma_{forb})$ sarà realizzata concatenando le matrici $\hat{R}^0$ e $\hat{T}^0$ ($\hat{R}^1$ e $\hat{T}^1$ rispettivamente). Cioè $S^0 = \hat{R}^0 \circ \hat{T}^0$ e $S^1 = \hat{R}^1 \circ \hat{T}^1$, dove con $\circ$ denotiamo l'operatore di concatenazione di due matrici.

**Esempio 3.10** *Sia* $\mathcal{P} = \{1,2,3,4,5\}$ *e sia* $\Gamma_0 = \{\{1,2\},\{2,3\},\{3,4\},\{4,5\},\{1,5\},\{2,5\}\}$. *Possiamo costruire un VCS per una struttura di accesso forte* $(\Gamma_{qual}, \Gamma_{forb})$ *avente base* $\Gamma_0$ *usando i VCS per le strutture di accesso forti con base* $\Gamma'_0 = \{\{1,2\},\{1,5\}\}$ *e* $\Gamma''_0 = \{\{2,3\},\{3,4\},\{4,5\},\{2,5\}\}$, *rispettivamente*

$$R^0 = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 0 \end{bmatrix} \qquad R^1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{bmatrix}$$

$$T^0 = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 1 & 0 \end{bmatrix} \qquad T^1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$

*dalle precedenti matrici otteniamo le matrici* $\hat{R}_0$, $\hat{R}_1$, $\hat{T}_0$, $\hat{T}_1$:

$$\hat{R}^0 = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \end{bmatrix} \qquad \hat{R}^1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}$$

$$\hat{T}^0 = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 1 & 0 \end{bmatrix} \qquad \hat{T}^1 = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$

*Concatenando le matrici* $\hat{R}_0$ *con* $\hat{T}_0$ *e le matrici* $\hat{R}_1$ *e* $\hat{T}_1$, *otteniamo le seguenti matrici di base* $S^0$ *e* $S^1$ *per un VCS per la struttura di accesso forte con base* $\Gamma_0$:

$$S^0 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 \end{bmatrix} \qquad S^1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}$$

La correttezza di tale costruzione è data dal seguente teorema.

**Teorema 3.11** *Siano* $(\Gamma'_{qual}, \Gamma'_{forb})$ *e* $(\Gamma''_{qual}, \Gamma''_{forb})$ *due strutture di accesso su un insieme di* $n$ *partecipanti* $\mathcal{P}$. *Supponiamo che esista un* $(\Gamma'_{qual}, \Gamma'_{forb}, m')$*-VCS e* $(\Gamma''_{qual}, \Gamma'_{forb}, m'')$*-VCS con matrici di base* $R^0$, $R^1$ *e* $T^0$, $T^1$, *rispettivamente. Allora la precedente costruzione fornisce un* $(\Gamma'_{qual} \cup \Gamma''_{qual}, \Gamma'_{forb} \cap \Gamma''_{forb}, m' + m'')$*-VCS. Se le strutture di accesso originali sono entrambe forti, allora la risultante struttura d'accesso è forte.*

La tecnica di costruzione descritta precedentemente non funziona per VCS generali (cioè se esse non sono state costruite da matrici di base). Cioè , dato un $(\Gamma'_{qual}, \Gamma'_{forb}, m')$-VCS e un $(\Gamma''_{qual}, \Gamma''_{forb}, m'')$-VCS la concatenazione delle matrici dei due schemi non fornisce un $(\Gamma'_{qual} \cup \Gamma''_{qual}, \Gamma'_{forb} \cap \Gamma''_{forb}, m' + m'')$-VCS. Infatti, consideriamo le collezioni $C_0$ e $C_1$ di un VCS $(2,2)$ denotato con $\Sigma$, ottenute come segue.
La collezione $C_0$ è realizzata considerando le matrici ottenute permutando le colonne delle seguenti matrici:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \end{bmatrix}$$

mentre la collezione $C_1$ è realizzata considerando le matrici ottenute permutando le colonne delle seguenti matrici:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Supponiamo di voler utilizzare $\Sigma$ per realizzare un VCS per strutture di accesso forti avente basi $\{\{1,2\}\}$ e $\{\{2,3\}\}$. Per costruire le collezioni $C_0$ e $C_1$ di un VCS per strutture di accesso forti avente basi $\{\{1,2\}, \{2,3\}\}$, non possiamo semplicemente concatenare le matrici dei due schemi, infatti è facile verificare che

$$M = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \end{bmatrix} \in C_0 \qquad M' = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} \in C_1$$

Da ciò ricaviamo che $w(M_{1,2}) = w(M'_{1,2}) = 4$ che contraddice la proprietà 1 della definizione 3.5. Quindi, la tecnica di costruzione sviluppata nel teorema precedente non funziona per strutture generali VCS.

## 3.4 Un esempio di share per un VCS con strutture d'accesso

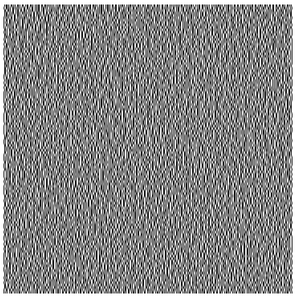
L'esempio, che segue, ha come insieme di partecipanti $\mathcal{P} = \{1,2,3\}$ e insieme di partecipanti qualificati $\Gamma_{qual} = \{\{1,2\},\{2,3\}\}$. Le matrici di base risultanti saranno, quindi, le seguenti :

$$S_0 = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 0 \end{bmatrix} \qquad S_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{bmatrix}$$

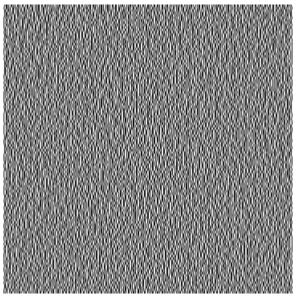
Da tali matrici di base si ottengono le seguenti share :

Immagine originale

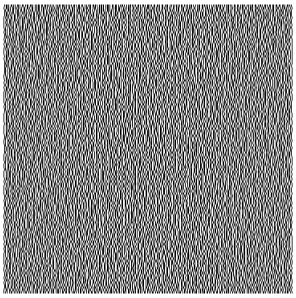
Share data al partecipante 1



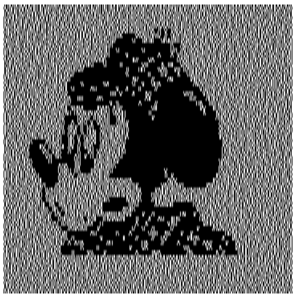
Share data al partecipante 2



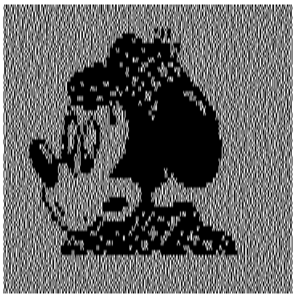
Share data al partecipante 3



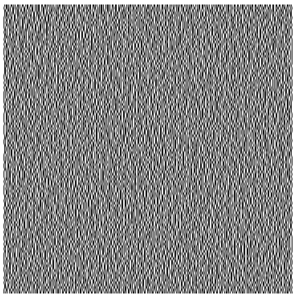
Sovrapposizione delle share 1 e 2



Sovrapposizione delle share 2 e 3



Sovrapposizione delle share 1 e 3



# 4 Crittografia Visuale Estesa

In questa sezione introdurremo un'estensione dei VCS per le strutture di accesso introdotte precedentemente.

## 4.1 Introduzione

Nei VCS visti finora le share date ai partecipanti erano composte da una collezione casuale di pixel bianchi e neri. Con la crittografia visuale estesa (EVCS) le share date ai partecipanti contengono delle immagini "banali" (una casa, un cane, un albero, $\cdots$). Sia $\mathcal{P} = \{1, 2, 3, 4\}$ e consideriamo la struttura di accesso $\Gamma_{qual} = \{\{1, 2\}, \{2, 3\}, \{1, 2, 3\}\}$ e supponiamo che tutti i rimanenti sottoinsiemi di $\mathcal{P}$ sono non qualificati. Vogliamo condividere la figura $\boxed{\text{S}}$ in modo tale che la share del partecipante 1 sia la figura $\boxed{\text{A}}$, la share del partecipante 2 sia la figura $\boxed{\text{B}}$ e la share del partecipante 3 sia la figura $\boxed{\text{C}}$. Questa distribuzione delle share dovrebbe avere la proprietà che, quando i partecipanti 1 e 2, i partecipanti 2 e 3, o i partecipanti $1, 2$ e 3 sovrappongono le loro share ottengono l'immagine segreta $\boxed{\text{S}}$.

Un $(\Gamma_{qual}, \Gamma_{forb,m})$-EVCS con espansione di pixel $m$, per una struttura di accesso $(\Gamma_{qual}, \Gamma_{forb})$ su un insieme di $n$ partecipanti, è una tecnica per codificare $n$ immagini "banali" in modo tale che, quando sovrapponiamo i lucidi associati ai partecipanti di un insieme $X \in \Gamma_{qual}$ otteniamo l'immagine segreta con nessuna traccia delle immagini originali, ma ogni $X \in \Gamma_{forb}$ non ha nessuna informazione sull'immagine condivisa. Comunque, dopo che le immagini "banali" sono codificate, devono essere significative, cioè ogni utente riconosce l'immagine sul suo lucido.

## 4.2 Schemi di Crittografia Visuale Estesa

In un EVCS, abbiamo $n+1$ immagini (le prime $n$ sono associate con gli $n$ partecipanti mentre l'ultima è l'immagine segreta), e vogliamo costruire le $n$ share significative, cioè , ogni utente deve essere capace di riconosce l'immagine di partenza nel suo lucido. Quindi, ogni tecnica per implementare gli EVCS deve prendere in considerazione i colori dei pixel nelle $n$ immagini "banali" e il colore del pixel nell'immagine segreta che vogliamo ricostruire. Nel seguito ci riferiremo al colore di un pixel bianco (nero) come un $w$ pixel ($b$ pixel). In generale, denotiamo con $C_c^{c_1,\cdots,c_n}$, dove $c, c_1, \cdots, c_n \in \{b, w\}$, la collezione delle matrici dalle quali il dealer sceglie una matrice per cifrare un pixel $c$, per $i = \{1, \cdots, n\}$, i sottopixel in $c_i$ appartengono alla share del partecipante $i$, sovrapponendo i sottopixel di tutti i partecipanti in $X \in \Gamma_{qual}$ otteniamo il pixel $c$ dell'immagine segreta. Quindi, per realizzare un EVCS dobbiamo costruire $2^n$ coppie di tali collezioni $(C_w^{c_1,\cdots,c_n}, C_b^{c_1,\cdots,c_n})$, una per ogni possibile combinazione di pixel bianchi e neri nelle $n$ immagini originali. Assumeremo che non vi siano partecipanti isolati nella struttura di accesso. Questa assunzione non è restrittiva, poichè non ha senso considerare partecipanti isolati in un EVCS. Detto ciò diamo una definizione formale di un EVCS. Se permettiamo alla struttura di acceso di contenere partecipanti isolati in EVCS, allora ciò significa dall'immagine "banale" (quella data al aprtecipante isolato) siamo capaci di ottenere il segreto semplicemente guardandola, senza effettuare nessuna computazione crittografica. Chiaramente questo è impossibile, a meno che la figura posseduta dal partecipante isolato non sia il segreto stesso.
Un EVCS per una struttura di accesso $\Gamma$ è definito come segue:

**Definizione 4.1** *Sia $(\Gamma_{qual}, \Gamma_{forb})$ una struttura di accesso generale di $n$ partecipanti. Una famiglia di $2^n$ coppie di collezioni (multiinsiemi) di matrici booleane di dimensioni $n \times m$ $\{(C_w^{c_1,\cdots,c_n},\ C_b^{c_1,\cdots,c_n})\}_{c_1,\cdots,c_n \in \{b,n\}}$ costituisce un forte $(\Gamma_{qual}, \Gamma_{forb}, m)$-EVCS se esiste $\alpha(m)$ e $\{t_X\}_{X \in \Gamma_{qual}}$ che soddisfa le seguenti condizioni:*

1. *Ogni insieme qualificato $X \in \Gamma_{qual}$ può recuperare l'immagine segreta.*
   *Formalmente, per ogni $X \in \Gamma_{qual}$ e per ogni $c_1, \cdots, c_n \in \{b, w\}$ la soglia $t_X$ e la differenza relativa $\alpha(m)$ sono tali che per ogni $M \in C_w^{c_1,\cdots,c_n}$ abbiamo che $w(M_X) \leq t_X - \alpha(m) \cdot m$; mentre per ogni $M \in C_b^{c_1,\cdots,c_n}$ risulta che $w(M_X) \geq t_X$.*

2. *Ogni insieme non qualificato $X = \{i_1, \cdots, i_n\} \in \Gamma_{forb}$ non ha alcuna informazione sull'immagine condivisa.*
   *Formalmente, per ogni $c_{i_1}, \cdots, c_{i_p} \in \{b, w\}$ la coppia di collezioni $\cup_{i \in \{1,\cdots,n\} \setminus X} \cup_{c_i \in \{b,w\}}$ $\mathcal{D}_t^{c_1,\cdots,c_n}$ con $t = \{b, w\}$, dove $\mathcal{D}_t^{c_1,\cdots,c_n}$ è ottenuto restringendo ogni matrice di dimensione $n \times m$ in $C_t^{c_1,\cdots,c_n}$ alle righe $i_1, \cdots, i_p$, sono indistinguibili nel senso che contegono le stesse matrici con le stesse frequenze.*

3. *Dopo che le immagini "banali" sono codificate sono ancora significative, cioè ogni utente riconosce l'immagine sulla sua trasparenza.*

*Formalmente, per ogni* $i \in \{1, \cdots, n\}$ *e ogni* $c_1, \cdots, c_{i-1}, c_{i+1}, \cdots, c_n \in \{b, w\}$ *risulta che:*

$$\min_{M \in \mathcal{M}_b} w(M_i) > \max_{M \in \mathcal{M}_w} w(M_i),$$

*dove* $\mathcal{M}_b = \cup_{c_1,\cdots,c_{i-1},c_{i+1},\cdots,c_n \in \{b,w\}} C_w^{c_1,\cdots,c_{i-1},b,c_{i+1},\cdots,c_n}$
*e* $\mathcal{M}_w = \cup_{c_1,\cdots,c_{i-1},c_{i+1},\cdots,c_n \in \{b,w\}} C_w^{c_1,\cdots,c_{i-1},w,c_{i+1},\cdots,c_n}$.

4. *Per ogni scelta di* $c_1, \cdots, c_n \in \{b, w\}$*, la coppia di collezioni* $(C_w^{c_1,\cdots,c_n}, C_b^{c_1,\cdots,c_n})$ *costituisce un* $(\Gamma_{qual}, \Gamma_{forb}, m)$*-VCS.*

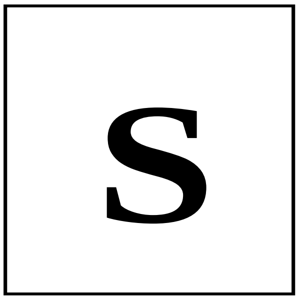
La prima condizione afferma che un insieme di utenti qualificati, appartenenti a $\Gamma_{qual}$, sovrapponendo le immagini può recuperare correttamente il segreto. La seconda condizione è collegata alla sicurezza dello schema, essa implica che, ispezionando solo le share e le immagini originali associate a un insieme di partecipanti non qualificato, non si riesce ad ottenere nessuna informazione sull'immagine segreta. La terza condizione implica che le immagini originali non sono "modificate", cioè dopo aver codificato le $n$ immagini "banali" originali utilizzando le $2^n$ coppie di collezioni $(C_w^{c_1,\cdots,c_n}, C_b^{c_1,\cdots,c_n})$, dove $c_1, \cdots, c_n \in \{b, w\}$, ogni utente riconoscerà l'immagine sul suo lucido. La quarta condizione è anch'essa collegata alla sicurezza dello schema, essa implica che, ispezionando le immagini associate a un sottoinsieme non qualificato di partecipanti, non si riesce ad ottenere nessuna informazione sull'immagine segreta, anche se si conoscono le immagini originali di tutti gli $n$ partecipanti. Questo è dovuto al fatto che, per ogni $c_1, \cdots, c_n \in \{b, w\}$, la coppia di collezioni $(C_w^{c_1,\cdots,c_n}, C_b^{c_1,\cdots,c_n})$, costituisce un VCS.
Il prossimo esempio mostra come realizzare uno schema $(2, 2)$-EVCS.

**Esempio 4.2** *Le collezioni* $C_c^{c_1,c_2}$*, dove* $c, c_1, c_2 \in \{b, w\}$*, di uno schema a soglia* $(2, 2)$*-EVCS sono ottenute permutando le colonne delle seguenti matrici:*

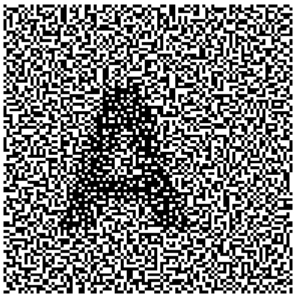
$$\begin{array}{llll}
S_w^{ww} = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix} & & S_b^{ww} = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix} \\
S_w^{wb} = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 \end{bmatrix} & & S_b^{wb} = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 \end{bmatrix} \\
S_w^{bw} = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix} & & S_b^{bw} = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix} \\
S_w^{bb} = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 \end{bmatrix} & & S_b^{bb} = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{bmatrix}
\end{array}$$

## 4.3 Un esempio di share per un EVCS

Immagine originale



Share data al partecipante 1



Share data al partecipante 2

Share data al partecipante 3

Immagine risultante dalla sovrapposizione delle share

# 5 Identificazione ed Autenticazione Visuale

L'autenticazione e l'identificazione sono tra i principali argomenti trattati in crittografia. In un protocollo di autenticazione un utente A cerca di trasmettere alcuni messaggi ad un'altro utente B. Un nemico, O, controlla il canale di comunicazione che gli utenti A e B usano per comunicare e può cambiare il messaggio trasmesso sul canale. Alla fine del protocollo, l'utente B deve stabilire se ritenere autentico il messaggio. Se il nemico non altera la comunicazione, allora l'utente B accetta il messaggio come autentico. Se, d'altra parte, il nemico cambia il messaggio, l'utente B dovrebbe accorgersene con alta probabilità . In un protocollo di identificazione, un utente A vuole provare la sua identità ad un verificatore. Un nemico che tenta di identificarsi come A al verificatore può riuscirci con una bassa probabilità .
Supponiamo che l'utente B in un protocollo di autenticazione o l'utente A in un protocollo di identificazione siano umani, quindi non possono effettuare calcoli complicati o memorizzare grandi quantità di dati. Supponiamo inoltre che queste persone non hanno nessun dispositivo di calcolo eccetto le loro naturali capacità . Gli uomini non possono effettuare quei calcoli che sono facilmente eseguiti dai computer, tuttavia la percezione visiva umana può facilmente effettuare calcoli che possono essere considerati "calcoli complicati". I sistemi presentati utilizzano le capacità visuali degli uomini, inoltre supponiamo che, nei protocolli, entrambe le parti che si scambiano messaggi hanno un'informazione segreta in comune, precisamente un'immagine su un lucido.

## 5.1 Modello e definizioni per l'Autenticazione Visuale

**Definizione 5.1 scenario per l'autenticazione visuale** *Ci sono tre entità in uno scenario di autenticazione visuale: U (Umberto), P (Pasquale) ed S (Serena). U è umano ed ha capacità umane. Queste capacità devono includere l'abilità di identificare l'immagine risultante*

*dalla composizione di due share di uno schema visuale a soglia* $(2,2)$, *di verificare che un'area è nera, di stabilire se due immagini sono simili, etc.*
*C'è un parametro di sicurezza,* $n$, *tale che, la capacità di memorizzazione e la potenza di calcolo di* $S$ *e* $P$ *sono polinomiali in* $n$.
*Nella fase di inizializzazione,* $S$ *calcola una stringa casuale* $r$, *e crea una trasparenza* $T_r$ *e qualche informazione ausiliaria* $A_r$. *La loro dimensione è polinomiale in* $n$. $S$ *invia* $T_r$ *e* $A_r$ *ad* $U$ *attraverso un canale privato al quale* $P$ *non ha accesso.* $S$ *invia ad* $U$ *anche un'insieme di istruzioni che* $U$ *dovrà eseguire nel protocollo. Le istruzioni possono consistere nel controllare se una determinata area in un'immagine è nera, confrontare due aree, etc. Queste istruzioni sono pubbliche e possono essere conosciute da* $P$, *ma* $P$ *non deve essere capace di cambiarle. Dopo la fase di inizializzazione tutte le comunicazioni tra* $U$ *e* $S$ *sono fatte usando il canale controllato da* $P$, *il quale può cambiare i messaggi.*

**Definizione 5.2 protocollo di autenticazione visuale** $S$ *vuole comunicare ad* $U$ *un' informazione* $m$, *il cui contenuto è conosciuto a* $P$.

- $S$ *invia un messaggio* $c$, *che dipende da* $m$ *ed* $r$, *ad* $U$.
- $P$ *può cambiare il messaggio, da* $c$ *in* $c'$, *prima che* $U$ *lo riceva.*
- *ricevendo il messaggio* $c'$, $U$ *decide se accettare* $m$ *basandosi su* $c'$ *e sulle informazioni segrete* $T_r$ *e* $A_r$.

Un protocollo di autenticazione visuale deve soddisfare i seguenti requisiti:

1. un nemico non può convincere $U$ ad accettare un messaggio $m'$, calcolato a partire da $m$, differente dal messaggio originale $m$,
2. per ogni messaggio $m'$, il nemico non può convincere $U$ ad accettare il messaggio $m'$ come autentico.

Nel nostro modello l'avversario, $P$, può cambiare il messaggio inviato da $S$ ad $U$. Comunque se il messaggio è una share di uno schema visuale per la condivisione del segreto, allora la share dovrebbe contenere esattamente due sottopixel neri in ogni quadrato $2 \times 2$ che rappresenta un pixel. Ci sono due tipi di modifiche che $P$ può apportare:

1. $P$ può cambiare la posizione di due sottopixel neri dentro un quadrato che rappresenta un pixel dell'immagine. Questo cambiamento non può essere notato da $U$.
2. $P$ può inserire o eliminare due sottopixel neri dentro un quadrato che rappresenta un pixel dell'immagine. Questo produce una share illegale. Comunque, la modifica probabilmente non sarà notata da $U$ a meno che questa non venga ripetuta per più pixel.

Assumeremo che l'immagine che l'uomo $U$ vede non varia nel tempo, in particolare non cambia dopo che l'utente ha posto la sua trasparenza sopra l'immagine. Questo può essere facilmente ottenuto se l'immagine è prima stampata e poi usata da $U$ (comunque questo richiede l'uso di una stampante, che può essere troppo costosa per alcune applicazioni). Assumeremo, inoltre, che il contenuto della trasparenza di $U$ rimane segreto. Per esempio, non ci deve essere una telecamera nascosta dietro ad $U$ che legge il contenuto della trasparenza.

## 5.2 Metodo dell'Area Nera e Area di informazione

Dividiamo le immagini in due aree. Definiamo a caso un'area nera e l'altra area come l'area che contiene l'informazione che deve essere inviata ad $U$. $U$ riceve la sua trasparenza, la quale è una share di uno schema per la condivisione del segreto $(2,2)$. Questa trasparenza è divisa in due aree e su ogni area è scritto se quella è l'area nera o l'area con l'informazione. $S$ invia ad $U$, come messaggio, una share di uno schema per la condivisione del segreto $(2,2)$. L'immagine risultante dalla combinazione delle due trasparenze, quella posseduta da $U$ e quella inviata ad $U$ da $S$, ha il messaggio $m$ nell'area informazione e l'area nera che è completamente nera, come mostrato in figura 5. $U$ è informato che se l'area nera non è totalmente nera, allora deve considerare il messaggio ricevuto come un tentativo di frode.

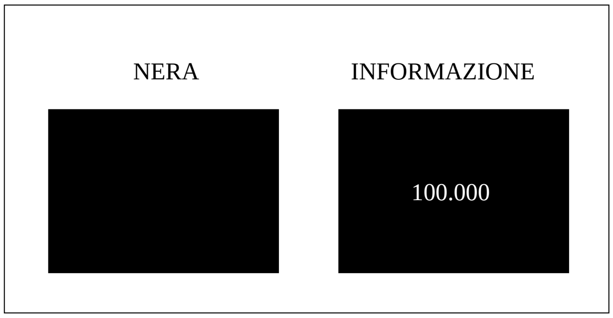



Figura 5: Risultato della composizione della trasparenza dell'utente $U$ e dell'immagine comunicata da $S$.

Se il nemico $P$ volesse cambiare il messaggio visualizzato ad $U$ dovrebbe effettuare alcune modifiche all'area che contiene informazione nell'immagine. Potrebbe effettuare modifiche del 1 tipo (cambiando la posizione di due sottopixel neri in un quadrato $(2,2)$ che rappresenta un pixel), o modifiche del 2 tipo (cambiando il numero dei sottopixel neri in un quadrato $(2,2)$ che rappresenta un pixel). Per trasformare un messaggio $m$ in un messaggio $m'$ effettuando modifiche del 2 tipo, dovrebbero esserci esattamente $|m\Delta m'| = |(m \cap \overline{m'}) \cup (\overline{m} \cap m')|$ cambiamenti. La sicurezza del sistema dipende dalle seguenti due assunzioni circa le capacità di $U$:

1. per qualsiasi coppia di messaggi differenti $m$ e $m'$, $U$ può notare se la share ricevuta da $S$ ha almeno $|m\Delta m'|$ o più pixel che hanno un numero di sottopixel neri maggiore o minore di 2. Questa assunzione è ragionevole poichè se $|m\Delta m'|$ è troppo piccolo allora i due messaggi non sono semanticamente differenti.

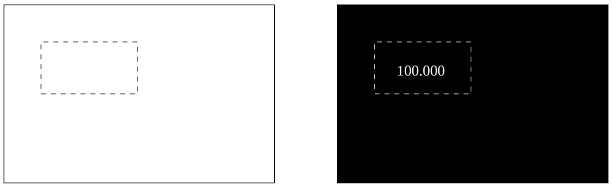
2. $U$ è capace di notare anche un solo sottopixel bianco nell'area nera. Poichè tutti e quattro i sottopixel di un pixel nero sono neri, l'area nera è completamente nera e quindi è ragionevole assumere che $U$ è capace di notare qualsiasi pixel bianco in quest'area.

La prima assunzione impedisce a $P$ di cambiare il messaggio usando solo modifiche del tipo 2. La seconda assunzione impedisce a $P$ di effettuare qualsiasi cambio del tipo 1 nell'area nera. Quindi $P$ deve decidere qual'è l'area informazioni ed effettuare modifiche del tipo 1 solo in quest'area. La probabilità di successo è al più $\frac{1}{2}$.
È importante notare che il metodo descritto è un metodo one-time. Ogni trasparenza può

essere usata solo una volta. Quando una trasparenza è usata una seconda volta, $P$ può confrontare l'immagine che $S$ invia con la precedente immagine inviata e dedurre qualè l'area nera. $P$ può così cambiare il messaggio da visualizzare con un messaggio scelto.
Per ridurre la probabilità di successo di $P$ possiamo usare $k$ aree: ci sono $2^k - 1$ possibilità di dividere le $k$ aree in aree nere e aree d'informazione in modo tale che c'è almeno un'area d'informazione. Una o più di queste aree saranno selezionate a caso e saranno usate come aree d'informazione. L'utente è stato informato precedentemente di quali aree sono nere e quali contengono informazioni. L'immagini osservata dovrebbe essere la stessa in ogni area informazione e tutte le altre aree dovrebbero essere nere. Se $P$ vuole cambiare il messaggio da visualizzare dovrebbe decidere, per ogni area, se è un'area nera o un'area informazione. La sua probabilità di successo è al più $\frac{1}{2^k-1}$. Questo è più efficiente che ripetere lo schema di base per ottenere questa probabilità : ripetendo lo schema di base dobbiamo effettuare $K$ ripetizioni (contemporaneamente), il che richiede $2k$ aree.
C'è una variazione di questo metodo che non richiede che l'utente controlli l'immagine posta sulla trasparenza. La trasparenza contiene tre aree che sono cosiderate, a caso, come area nera, area bianca e area informazione. L'utente conosce qualè il "titolo" di ogni area e si aspetta di vedere il messaggio nell'area informazione, l'area nera completamente nera e l'area bianca completamente bianca. $P$ non conosce quali sono le aree. Se $P$ modifica un pixel di $m$ in modo da avere meno di due sottopixel neri, allora con probabilità $1/3$ questo pixel è nell'area nera e assumendo che $U$ può notare un pixel bianco nell'area nera, $U$ si accorge che $P$ tenta di frodarlo. Analogamente succede se $P$ modifica un pixel di $m$ in modo da avere più di due sottopixel neri. Quindi, per aumentare la probabilità di successo, $P$ è costretto a mantenere i pixel con esattamente due sottopixel neri.

## 5.3 Metodo della Posizione sullo Schermo

Assumeremo che l'immagine è composta da $r \times c$ pixel. Nella fase di inizializzazione $S$ dice ad $U$ la posizione di un rettangolo di dimensione $r' \times c'$ nel quale il messaggio dovrebbe comparire. Sia $(i, j)$ la posizione dell'angolo superiore sinistro del rettangolo. Il perimetro del rettangolo è disegnato sulla trasparenza con linee tratteggiate, così $U$ può facilmente controllare se il messaggio è visualizzato nel rettangolo. Il messaggio è visualizzato in bianco su uno sfondo nero che copre tutti i pixel, dentro e fuori il rettangolo. La figure 6 mostra una trasparenza con un rettangolo e il risultato della sovrapposizione dell'immagine con il messaggio.




(a) (b)

Figura 6: (a) Lucido dell'utente. (b) Risultato della composizione della trasparenza dell'utente $U$ e dell'immagine comunicata da $S$.

Anche se $P$ controlla il computer sul cui schermo è visualizzato il messaggio $m$, è difficile per $P$ visualizzare un messaggio $m'$ diverso da $m$. Sia $m_c = (m \cap m') \cup (\overline{m} \cap \overline{m}')$, i pixel in quest'area hanno lo stesso colore in entrambi i messaggi $m$ ed $m'$, e non dovrebbero essere cambiati da $P$. Sia $m_d = m \Delta m' = (m \cap \overline{m}') \cup (\overline{m} \cap m')$, i pixel in quest'area devono essere invertiti dal nemico, cioè , $P$ deve trasformare un pixel nero in bianco ed un pixel bianco in nero. Questo schema è abbastanza sicuro.

## 5.4 Metodo del Nero e Grigio

La sicurezza di questo schema è esponenziale nella distanza di hamming (che è il numero di bit necessari a cambiare una stringa in un'altra) tra il messaggio che $S$ invia ad $U$ ed il messaggio che $P$ vuole visualizzare ad $U$. Lo svantaggio di questo metodo è che riduce il contrasto dell'immagine visualizzata.
Nello schema visuale per la condivisione del segreto usato, tutti e quattro i sottopixel associati ad un pixel nero sono neri, mentre ci sono due sottopixel neri e due bianchi associati ad un pixel bianco. Possiamo definire un pixel **grigio** come un pixel con tre sottopixel neri ed uno bianco. Denotiamo con $s_1$ e $s_2$ le due share associate ad un pixel. Se le share sono associate ad un pixel nero, è facile costruire da $s_1$ una nuova share $s_1'$ che assieme a $s_2$ formano un pixel grigio. Comunque, se le share sono associate ad un pixel grigio, la probabilità di costruire una share $s_1'$ che sovrapposta a $s_2$ formi un pixel nero è al più $1/4$. Quando il messaggio $m$ è scritto in nero su uno sfondo grigio, è difficile per il nemico cambiare un pixel dello sfondo in un pixel del messaggio. Similmente, quando un messaggio è scritto in grigio su uno sfondo nero è difficile per il nemico "cancellare" un pixel del messaggio e cambiarlo in un pixel dello sfondo. Lo schema proposto visualizza il messaggio in due aree. In un'area lo visualizza in nero su uno sfondo grigio e nell'altra area in grigio sullo sfondo nero. L'utente è informato che deve verificare che il messaggio in entrambe le aree sia uguale.

## 5.5 Schemi riutilizzabili

I tre metodi proposti precedentemente sono tutti one-time, cioè sono sicuri solo per un'autenticazione. Ovviamente è preferibile avere un metodo che sia sicuro per diverse autenticazioni. Una semplice costruzione di uno schema utilizzabile più volte è ottenuta dai precedenti schemi one-time memorizzando diverse copie di essi in differenti aree di una singola trasparenza, dove ogni copia dipende da differenti dati casuali. Il numero di copie in una singola trasparenza dipende dal parametro di sicurezza, che definisce la dimensione dell'area che è usata da ogni copia della schema one-time, e dalla dimensione della trasparenza.

## 5.6 Modello e Definizione per l'identificazione Visuale

Lo scenario per l'identificazione visuale è identico a quello per l'autenticazione visuale, comunque, lo scopo di un protocollo di identificazione è differente: permettere ad un utente umano $U$ di provare la sua identità ad un verificatore $S$ senza l'ausilio di dispositivi di calcolo. Lo scopo del nemico $P$ è di convincere il verificatore che lui è l'utente umano $U$. Se un protocollo di identificazione visuale è utilizzabile solo uno volta esso è inutile in quanto possiamo semplicemente utilizzare un protocollo che fà uso di una password. Quindi considereremo solo protocolli di identificazione utilizzabili più volte, cioè protocolli che possono essere usati per molte identificazioni usando una sola trasparenza. Il protocollo è del tipo **challenge-response** nel quale il verificatore invia una sfida (challenge) all'utente, ed egli dovrebbe rispondere basandori sulle informazioni segrete in suo possesso, in modo da provare la sua identità .

**Definizione 5.3 protocollo di identificazione visuale** *definiremo il protocollo per l'i-ma identificazione di $U$ a $S$:*

- *$S$ invia la sfida $c_i$ ad $U$, $c_i$ è funzione del dato segreto $r$ che è comune ad $S$ ed $U$, e di alcuni bit $r_S$ casuali che sono conosciuti solo a $S$.*
- *$U$ ricevendo $c_i$, elabora una risposta $a_i$ come funzione di $c_i$ e dell'informazione segreta $T_r$ e $A_r$, e la invia a $S$.*
- *$S$ decide se l'altra parte è $U$ basandosi sui messaggi $c_i$ e $a_i$ e sul dato segreto $r$. Successivamente risponde ACCETTO o RIFIUTO.*

Il nemico $P$ può tentare di spacciarsi per $U$. In questo caso può prima tentare di spacciarsi per $S$ ad $U$ e richiedere ad $U$ di provare la sua identità . Successivamente inizia il protocollo di identificazione con il verificatore e invia la risposta che spera convinca $S$ cha l'altra parte è $U$.

**Definizione 5.4 protocollo di identificazione visuale $\ell$-volte $(1-p)$-sicuro** *Un protocollo di identificazione visuale è $\ell$-volte $(1-p)$-sicura se le seguenti due condizioni valgono dopo che il nemico $P$ ha ascoltato al più $l$ identificazioni che sono state fatte da $U$ o anche dopo che $P$ si è spacciato, ad $U$, per il verificatore al più per $l$ identificazioni.*

- *se $U$ risponde ad $S$ in accordo al protocollo, $S$ accetta con probabilità 1;*
- *se un nemico $P$ riceve il messaggio $c_i$ inviato da $S$ e risponde con un messaggio $b_i$ che dipende da $c_i$ e dalle precedenti $l$ comunicazioni $c_{i_1}, b_{i_1}, \cdots, c_{i_l}, b_{i_l}$, allora $S$ accetta con probabilità al più $p$.*

## 5.7 Un metodo di identificazione visuale

Il metodo presentato per l'identificazione visuale non usa schemi visuali per la condivisione dei segreti. Invece $U$ deve provare al verificatore $S$ che conosce alcune proprietà della trasparenza. Useremo trasparenze colorate da dieci differenti colori, che assumeremo essere facilmente distinguibili tra loro: verde, bianco, rosso, azzurro, granata, giallo, nero, marrone, rosa e arancione. Un differente insieme di colori può essere usato ed i parametri di sicurezza variano al variare del numero di colori nell'insieme.
Un vantaggio di questo metodo è che è molto economico rispetto ai metodi di identificazione sicuri usati correntemente i quali richiedono che l'utente consulti un dispositivo di calcolo portatile o richiedono di collegare una carta intelligente (smart card) ad una speciale interfaccia di un computer. I metodi di identificazione visuale sono accessibili a chiunque possiede una stampante a colori (anche una in bianco e nero) col quale costruire schemi di identificazione sicuri, che possono essere usati, per esempio, per permettere l'accesso a determinate aree o ad identificare i partecipanti ad una comunicazione. Inoltre, poichè il world-wide-web (internet) supporta un'interfaccia grafica universale il processo di identificazione visuale che è seguito quando un utente si connette ad un host remoto può usare il browser web per visualizzare l'immagine che il verificatore invia all'utente, e in questo caso non c'è bisogno di istallare nessun software speciale sul computer remoto per eseguire il processo di identificazione.

**Identificazione visuale con il verificatore umano:** il metodo di identificazione visuale suggerito richiede, per il verificatore, poche capacità , quindi i ruoli del verificatore e del provatore posso essere invertiti, cioè il verificatore è umano e verifica l'identità di un computer con il quale comunica. L'uomo può richiedere al computer remoto di provare la sua identità prima di inviargli alcune informazioni confidenziali (ad esempio il numero di una carta di credito).

## 5.8 Uno schema di identificazione sicuro per un singolo verificatore

In questa sezione l'unità di base che useremo nella trasparenza non è un pixel ma un quadrato, il quale è una collezione di pochi pixel (ad esempio un quadrato di $4 \times 4$ pixel). Durante la fase di inizializzazione l'utente $U$ riceve una trasparenza che è divisa in molti quadrati, ognuno dei quali è colorato con un colore scelto a caso tra i possibili colori. L'ordine dei colori che sono stati usati è segreto ed è conosciuto solo da $U$ e dal verificatore $S$ ($S$ conosce l'ordine esplicitamente, oppure può determinarne l'ordine usando l'output di un generatore casuale di numeri di cui conosce il seme).
Sia $N$ il numero di quadrati nella trasparenza, e sia $d$ il numero di quadrati che sono usati nel protocollo, il protocollo di identificazione è il seguente:

- $S$ sceglie casualmente $d$ quadrati ed invia ad $U$ un'immagine che è completamente nera, eccetto per le locazioni dei $d$ quadrati che sono bianchi.
- $U$ mette la sua trasparenza sull'immagine ricevuta da $S$ ed invia ad $S$ i colori nelle locazioni dei quadrati bianchi, in qualche ordine casuale.
- Il verificatore $S$ accetta solo se la risposta di $U$ è corretta per tutti i $d$ quadrati.

È chiaro che $U$ riesce sempre ad identificarsi ad $S$. Viceversa, il nemico $P$ può interagire con $U$ spacciandosi per $S$ al più $l$ volte ottenendo i colori di $dl$ quadrati. $P$ non ha nessuna informazione circa i colori degli altri quadrati, quindi la sua probabilità di successo è al più $\left(\frac{1}{10} + \frac{9dl}{10N}\right)^d$. Una trasparenza con $N$ quadrati può essere usata per $l = \frac{N}{9d}$ identificazioni e la sua sicurezza è maggiore di $1 - 5^{-d}$.

# Riferimenti

[1] G. Ateniese, C. Blundo, A. De Santis, e D. R. Stinson, *Constructions and Bounds for Visual Cryptography* in "23rd International Colloquium on Automata, Languages and Programming" (ICALP '96), F. M. auf der Heide e B. Monien Eds., Vol. 1099 of "Lecture Notes in Computer Science", Springer-Verlag, Berlino, pp. 416-428. 1996.

[2] G. Ateniese, C. Blundo, A. De Santis, e D. R. Stinson, *Visual Cryptography for General Access Structures* Information and Computation, Vol. 129, No. 2, (1996), pp. 86-106.

[3] G. Ateniese, C. Blundo, A. De Santis, e D. R. Stinson, *Extended Schemes for Visual Cryptography* sottomesso a Theoretical Computer Science.

[4] C. Blundo, A. De Santis, e D. R. Stinson, *On the Contrast in Visual Cryptography Schemes* Journal of Cryptology, accepted for publication. Available also at Theory of Cryptography Library, n. 96-13.

[5] C. Blundo, P. D'Arco, A. De Santis, and D. R. Stinson, *Contrast Optimal Threshold Visual Cryptography Schemes* sottomesso.

[6] S. Droste, *New Results on Visual Cryptography* in "Advances in Cryptology - CRYPTO '96", N. Koblitz Ed., Vol. 1109 of "Lecture Notes in Computer Science", Springer-Verlag, Berlino, pp. 401-415, 1996.

[7] T. Hofmeister, M. Krause e H. U. Simon, *Contrast-optimal k out of n Secret Sharing Schemes in Visual Cryptography* COCOON '97, T. Jiang e D. T. Lee, Eds., Vol. 1276 of "Lecture Notes in Computer Science", Springer-Verlag, Berlino, 1997.

[8] M. Naor e A. Shamir, *Visual Cryptography* in "Advances in Cryptology - EUROCRYPT '94", A. De Santis Ed., Vol. 950 of Lecture Notes in Computer Science, Springer-Verlag, Berlino, pp. 1-12, 1995.

[9] M. Naor and A. Shamir, *Visual Cryptography II: Improving the Contrast via the Cover Base* presented at "Security in Communication Networks", 16-17 Settembre, 1996, Amalfi, Italy. Disponibile anche in Theory of Cryptography Library come Technical Report N.7. Una versione preliminare appare in "Security Protocols", M. Lomas Ed., Vol. 1189 of Lecture Notes in Computer Science, Springer-Verlag, Berlino, pp. 197-202, 1997.

[10] M. Naor e Benny Pinkas, *Visual Authentication,* in "Advances in Cryptology - CRYPTO '97", B. Kaliski Jr. Ed., Vol. 1294 of "Lecture Notes in Computer Science" Springer-Verlag, Berlino, pp. 322-336, 1997.

[11] Douglas R. Stinson, *An Introduction to Visual Cryptography* presentato a "Public Key Solutions '97", 28-30 Aprile, 1997, Toronto, Canada.

[12] Douglas R. Stinson, *Visual Cryptography & Threshold Schemes* Dr. Dobb's Journal, Aprile 1998.

[13] E.R. Verheul e H.C.A. van Tilborg, *Constructions and Properties of k out of n visual secret sharing schemes,* Designs, Codes and Cryptography, Vol. 11, 1997, pp. 179-196.