Steganografia

L'arte della scrittura nascosta

Graziella Montemarani

5 giugno 2005

Indice

Introduzione

Quando si parla di crittografia-dal greco $\kappa\rho\upsilon\pi\tau\varsigma$ (*kryptos*), che significa nascosto e $\gamma\rho\alpha\phi\epsilon\iota\nu$ (*graphein*, scrivere)-ci si riferisce a quell'arte che fornisce uno strumento adatto a mantenere segrete tutte quelle informazioni che non si vogliono divulgare pubblicamente, in maniera tale che la possibilità di accedervi sia data soltanto ad uno o ad un ristretto numero di persone autorizzate.
La steganografia-dal greco $\sigma\tau\epsilon\gamma o$ (*stego*, occultare, nascondere) e $\gamma\rho\alpha\phi\epsilon\iota\nu$ (*graphein*, scrivere)-rappresenta invece l'insieme delle tecniche che consente a due o più persone di comunicare in modo tale da nascondere non tanto il contenuto (come nel caso della crittografia), ma la stessa esistenza della comunicazione agli occhi di un eventuale osservatore, tradizionalmente denominato "nemico".
La debolezza della crittografia è proprio il sistema usato per criptare il messaggio: una volta abbattuto quello, si ha libero accesso a tutti i segreti trasmessi. La steganografia invece parte dal presupposto che nessuno dovrebbe sapere che esiste un messaggio segreto nascosto, e quindi risulta impossibile trovarlo. L'importante è quindi escogitare un metodo sicuro per poter nascondere il proprio segreto.
Lo studio della steganografia nella letteratura scientifica si deve a Simmons che nel 1983 formulò il "Problema dei prigionieri". In questo scenario Alice e Bob sono in prigione e devono escogitare un piano per fuggire: tutti i loro messaggi vengono scambiati tramite il guardiano Willie. Se Willie scopre che essi si scambiano messaggi cifrati metterà uno di loro in isolamento ed il piano fallirà. Quindi essi devono trovare un metodo per nascondere il loro testo cifrato in un testo apparentemente innocuo. Nonostante lo studio della steganografia abbia radici relativamente vicine si tratta di un'idea tutt'altro che nuova e che, anzi, vanta origini molto lontane. Nel corso dei secoli, sono stati escogitati infatti numerosi metodi steganografici, tutti molto diversi tra loro: può quindi essere utile accennare ad alcuni esempi tra i più interessanti ed ingegnosi...

1 Cenni storici

1.1 Storie di Erodoto

Una vecchia storia, ambientata alcuni secoli prima di Cristo, ci proviene dalle scritture di Erodoto. Il mezzo di scrittura del tempo era costituito da tavolette di legno ricoperte da cera sulla quale si incidevano i messaggi. Un esule greco stabilitosi in una città persiana, avendo saputo che Xerxes, re dei persiani, voleva invadere la Grecia, nonostante fosse in esilio, trovò uno stratagemma per avvisare i suoi compatrioti: sollevò la cera da una di queste tavolette, incise la notizia sul legno sottostante e la ricoprì di cera. La tavoletta che conteneva la soffiata appariva come inutilizzata, così riuscì ad oltrepassare le ispezioni e a raggiungere i greci.
In un'altra storia Erodoto racconta di un nobile persiano che fece tagliare a zero i capelli di uno schiavo fidato al fine di poter tatuare un messaggio sul suo cranio; una volta che i capelli furono ricresciuti, inviò lo schiavo alla sua destinazione, con la sola istruzione di tagliarseli nuovamente.

1.2 L'antica Cina

Nell'antica Cina invece si dipingeva il messaggio su striscioline di seta finissima, che venivano appallottolate e coperte di cera. Per evitare che i messaggi fossero intercettati le palline erano inghiottite dal messaggero.

1.3 Griglie di Cardano

Un altro strumento usato a scopi steganografici erano le griglie di Cardano che non erano altro che fogli di materiale rigido nei quali venivano ritagliati fori rettangolari ad intervalli irregolari. Questa griglia veniva appoggiata su un foglio di carta bianca, il messaggio segreto veniva scritto nei buchi (ciascun buco poteva contenere una o più lettere), dopodiché si toglieva la griglia e si cercava di completare la scrittura del resto del foglio in modo da ottenere un messaggio di senso compiuto, il quale poi veniva inviato a destinazione.

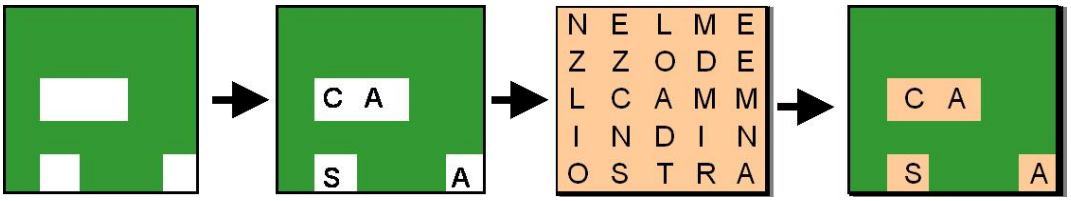

Figura 1: Esempio di griglia di Cardano

Per poter leggere il testo nascosto si doveva applicare sul messaggio ricevuto una copia esatta della griglia originaria.

1.4 Cifre nulle

Quella delle cifre nulle era un'altra tecnica steganografica che consisteva nell'inserire il messaggio nascosto in un altro messaggio di testo. Fu usata nella seconda guerra mondiale, in particolare per comunicazioni via radio. I messaggi trasmessi venivano registrati e poi filtrati in modo opportuno per ricavare il messaggio nascosto. Tecnicamente il messaggio trasmesso veniva composto in modo tale che, unendo le prime lettere di ogni capoverso o con altre tecniche, si ottiene un messaggio di senso compiuto. Il seguente, ad esempio, è un testo realmente inviato da una spia tedesca durante la seconda guerra mondiale:

A**p**parently n**e**utral's p**r**otest i**s** t**h**oroughly d**i**scounted a**n**d i**g**nored. Is**m**an h**a**rd h**i**t. B**l**ockade i**s**sue a**f**fects p**r**etext f**o**r e**m**bargo o**n** b**y** p*r*oducts, e**j**ecting s**u**ets a**n**d v**e**getable o**i**ls.

Considerando in sequenza la seconda lettera di ogni parola, si ottiene il messaggio:

Pershing sails from NY June 1

(anche se in realtà c'è una "r" di troppo e la "i" alla fine viene interpretata come 1).

1.5 Shakespeare VS Bacon

Esempio di steganografia molto interessante è rappresentato dalle opere di Shakespeare. Secondo molti studiosi, infatti, alcune opere dell'inglese possono essere attribuite al noto scrittore e statista Francis Bacon. Questo perché all'interno di tali scritti vi sono diversi testi nascosti che contengono il nome di Bacon stesso. A rafforzare questa ipotesi contribuiscono interessanti retroscena che accomunano Shakespeare e Bacon.

1.6 Inchiostri invisibili

Altro antico mezzo steganografico è costituito dagli inchiostri invisibili. Gli antichi romani usavano sostanze come succhi di frutta, latte e urine per scrivere messaggi segreti. Quando queste sostanze venivano riscaldate, divenivano leggibili. Gli inchiostri invisibili (o simpatici) sono stati usati anche nella II guerra mondiale.

1.7 Micropunti monografici

La tecnica dei micropunti fotografici fu inventata dal direttore dell' F.B.I. durante la seconda guerra mondiale, si tratta di fotografie della dimensione di un punto dattiloscritto che, una volta sviluppate e ingrandite, possono diventare pagine stampate di buona qualità.

1.8 Le immagini di Al-Queda

Anche oggi la steganografia viene utilizzata come veicolo politico-militare. Ad esempio, nel famoso quotidiano americano "USA Today" del 10 luglio del 2002 si legge: "Ultimamente al-Queda ha inviato centinaia di messaggi crittografati nascosti in fotografie digitali sul sito eBay.com. Molti dei messaggi sono stati inviati da café pakistani e librerie pubbliche di tutto il mondo...". E ancora: "Ufficiali americani dicono che azzam.com contiene messaggi crittografati nelle sue immagini e nei suoi testi (pratica conosciuta come steganografia). Essi affermano che i messaggi contengono istruzioni per i nuovi attacchi di al-Queda".

2 Modelli steganografici

Caratteristica della steganografia, come si è visto, è l'esistenza di due frammenti di informazione: il contenitore (o cover) ha il compito di nascondere il messaggio segreto (embedded), racchiudendolo al suo interno e rendendolo invisibile o, più correttamente, difficilmente percepibile. I principi che stanno alla base dei software steganografici rispondono alle caratteristiche appena descritte. Come si può facilmente immaginare, le nuove tecnologie e in particolar modo i sistemi per l'elaborazione dell'informazione, hanno consentito anche nel caso della steganografia la progettazione di nuove tecniche, sempre più sofisticate, sicure e pratiche da usare. Naturalmente esistono diversi approcci che distinguono i software in varie famiglie.

2.1 Steganografia iniettiva e generativa

Alcune tecniche, probabilmente le più numerose, consentono di "iniettare" il messaggio segreto dentro un messaggio contenitore già esistente, modificandolo in modo tale sia da contenere il messaggio sia da risultare, al livello al quale viene percepito dai sensi umani, praticamente indistinguibile dall'ori-

ginale. Indichiamo l'insieme di queste tecniche con il termine *steganografia iniettiva*.

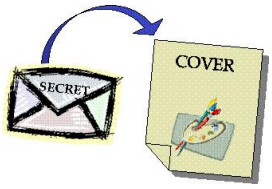

Figura 2: Steganografia iniettiva

Le tecniche di questo tipo sono di gran lunga le più diffuse, tanto che in genere con il termine steganografia ci si riferisce implicitamente a esse.

Esistono tuttavia altre tecniche steganografiche che hanno capacità proprie di generare potenziali messaggi contenitori e utilizzano il messaggio segreto per "pilotare" il processo di generazione del contenitore. Per queste tecniche adottiamo il termine *steganografia generativa*.

Figura 3: Steganografia generativa

2.2 Un'altra classificazione

Secondo un diverso sistema di classificazione le tecniche steganografiche possono essere ripartite in tre classi: *steganografia sostitutiva*, *steganografia selettiva* e *steganografia costruttiva*.

2.2.1 Steganografia sostitutiva

È senz'altro la tecnica steganografica più diffusa, tanto che spesso quando si parla di steganografia ci si riferisce implicitamente a quella di questo tipo. Alla base di questa tecnica c'è un'osservazione: la maggior parte dei canali di comunicazione (linee telefoniche, trasmissioni radio, ecc.) trasmettono segnali che sono sempre accompagnati da qualche tipo di rumore. Questo rumore può essere sostituito da un segnale (il messaggio segreto) che è stato trasformato in modo tale che, a meno di conoscere una chiave segreta, è indistinguibile dal rumore vero e proprio, e quindi può essere trasmesso senza destare sospetti. Quasi tutti i programmi si basano su questa idea, sfruttando la grande diffusione di file contenenti una codifica digitale di immagini, animazioni e suoni; spesso questi file sono ottenuti da un processo di conversione analogico/digitale e contengono qualche tipo di rumore. Per esempio, uno scanner può essere visto come uno strumento di misura più o meno preciso. Un'immagine prodotta da uno scanner, da questo punto di vista, è il risultato di una specifica misura e come tale è soggetta a essere affetta da errore. Lo stesso discorso lo si può fare analogamente per un file sonoro che evidentemente è stato acquisito tramite una scheda sonora. La tecnica impiegata nella maggior parte dei programmi è concettualmente molto semplice: sostituire i bit meno significativi dei file digitalizzati con i bit che costituiscono il file segreto (i bit meno significativi, infatti, corrispondono ai valori meno significativi, importanti ed evidenti di una misura, cioè proprio quelli che possono essere facilmente affetti da errore!). Quello che succede quindi è che il file contenitore risultante, dopo un'iniezione steganografica, si presenta in tutto e per tutto simile all'originale, con differenze difficilmente percettibili e quindi, a meno di confronti approfonditi con il file originale

(comunque non effettuabili ad occhio nudo) è difficile dire se le eventuali perdite di qualità siano da imputare al rumore od alla presenza di un messaggio segreto steganografato. Inoltre il più delle volte il file originale non è disponibile e quindi effettuare questo confronto è pressoché impossibile. Il principale difetto della steganografia sostitutiva è che le sostituzioni possono alterare le caratteristiche statistiche del rumore nel media utilizzato. Se il nemico, infatti, possiede un modello del rumore, può utilizzarlo per testare se i file sono conformi al modello; se non lo sono probabilmente si è in presenza di un messaggio steganografato. Il problema di questo tipo di attacco però sta nella difficoltà di costruire un modello che tenga conto di tutti i possibili errori o rumori. In casi molto specifici comunque questo tipo di attacco ha avuto buon successo e la steganografia selettiva e costruttiva hanno proprio lo scopo di evitarlo.

2.2.2 Steganografia selettiva

La steganografia selettiva ha valore puramente teorico e non viene utilizzata nella pratica. L'idea su cui si basa è quella di procedere per tentativi, ripetendo una stessa misura fintanto che il risultato non soddisfa una certa condizione. Si fissi una funzione hash semplice, da applicare a un'immagine in forma digitale (una funzione hash è una qualsiasi funzione definita in modo da dare risultati ben distribuiti nell'insieme dei valori possibili; tipicamente questo si ottiene decomponendo e mescolando in qualche modo le componenti dell'argomento); per semplificare al massimo, diciamo che la funzione vale 1 se il numero di bit uguali a 1 del file che rappresenta l'immagine è pari, altrimenti vale 0 (si tratta di un esempio poco realistico ma, come dicevamo, questa discussione ha valore esclusivamente teorico). Così, se vogliamo codificare il bit 0 procediamo a generare un'immagine con uno scanner; se il numero di bit dell'immagine uguali a 1 è dispari ripetiamo di nuovo la generazione, e continuiamo così finché non si verifica la condizione opposta. Il punto cruciale è che l'immagine ottenuta con questo metodo contiene effettivamente l'informazione segreta, ma si tratta di un'immagine

"naturale", cioè generata dallo scanner senza essere rimanipolata successivamente. L'immagine è semplicemente sopravvissuta a un processo di selezione (da cui il nome della tecnica), quindi non si può dire in alcun modo che le caratteristiche statistiche del rumore presentano una distorsione rispetto a un modello di riferimento. Come è evidente, il problema di questa tecnica è che è troppo dispendiosa rispetto alla scarsa quantità di informazione che è possibile nascondere.

2.2.3 Steganografia costruttiva

La steganografia costruttiva affronta lo stesso problema della steganografia sostitutiva nel modo più diretto, tentando di sostituire il rumore presente nel media utilizzato con l'informazione segreta opportunamente modificata in modo da imitare le caratteristiche statistiche del rumore originale. Secondo questa concezione, un buon sistema steganografico dovrebbe basarsi su un modello del rumore e adattare i parametri dei suoi algoritmi di codifica in modo tale che il falso rumore contenente il messaggio segreto sia il più possibile conforme al modello. Questo approccio è senza dubbio valido, ma presenta anche alcuni svantaggi.

Innanzitutto non è facile costruire un modello del rumore: la costruzione di un modello del genere richiede grossi sforzi ed è probabile che qualcuno, in grado di disporre di maggior tempo e di risorse migliori, riesca a costruire un modello più accurato, riuscendo ancora a distinguere tra il rumore originale e un sostituto. Inoltre, se il modello del rumore utilizzato dal metodo steganografico dovesse cadere nelle mani del nemico, egli lo potrebbe analizzare per cercarne possibili difetti e quindi utilizzare proprio il modello stesso per controllare che un messaggio sia conforme a esso. Così, il modello, che è parte integrante del sistema steganografico, fornirebbe involontariamente un metodo di attacco particolarmente efficace proprio contro lo stesso sistema.

3 Information hiding

Il processo di "data hiding", o occultamento di dati, consente di nascondere informazioni segrete nei testi, nelle immagini e nei segnali audio. Nell'utilizzo di tali tecniche, è opportuno cercare di preservare i messaggi da un eventuale deterioramento, nonché mantenerli il più lontano possibile da attacchi e manipolazioni. Per questo motivo, è usuale cifrare le informazioni prima di nasconderle, cioè combinare la crittografia con la steganografia, al fine di avere un maggiore livello di sicurezza. Così, sebbene non sia necessario, i messaggi da nascondere vengono spesso precedentemente cifrati. Questa è un'esplicita richiesta del cosiddetto "principio di Kerckhoff" in crittografia, che fu formulato in modo definitivo nel 1883 dallo stesso linguista olandese Auguste Kerckhoff von Nieuwenhof nel trattato dal titolo La Cryptographie Militaire: la sicurezza del crittosistema deve basarsi sull'ipotesi che il nemico abbia piena conoscenza dei dettagli di progetto ed implementazione del crittoalgoritmo; la sola informazione di cui il nemico non deve disporre è una sequenza di numeri casuali che costituisce la chiave segreta senza la quale, osservando un canale di comunicazione, non deve avere neanche la più piccola possibilità di verificare che è in corso una conversazione segreta.

Ci occuperemo dunque delle procedure di occultamento dei dati nei seguenti tipi di oggetto:

- file di testo
- immagini
- file audio

Le tecniche di data hiding possono essere classificate rispetto alle modalità di estrazione:

- procedure che, al fine di estrarre le informazioni segrete, necessitano sia del documento originale che di quello modificato da tale inserzione;
- metodi in grado di riconoscere il messaggio nascosto senza l'aiuto del file originale.

Di solito, i secondi sono più comodi a causa della difficoltà di distribuire copie autenticate dell'originale. Ci occuperemo dapprima in dettaglio di una tecnica di steganografia testuale tardo-medievale, la cifratura di Trithemius, che solo di recente però è stata svelata. Per quanto riguarda le moderne tecniche di data hiding nei testi, analizzeremo tre procedure: codifica line-shift, codifica word-shift e codifica feature. Ognuna di queste è stata progettata per combattere la distribuzione illegale di documenti di testo, marcandoli con caratteristiche riconoscibili, oppure spostandone le righe, o ancora variando lo spazio tra le parole, o infine alterando alcune fattezze delle lettere stesse.

Nascondere i dati all'interno di immagini è divenuto negli ultimi anni la più diffusa tecnica di steganografia, anche grazie all'infinità di file grafici che si possono trovare sulla rete. La procedura più comune è l'inserzione del messaggio segreto al posto dei bit meno significativi del file "ospite". Esistono però anche metodi più complessi, come il "masking & filtering", mascherare e filtrare, algoritmi e trasformazioni matematiche che offrono maggiore resistenza agli attacchi esterni, come il metodo Patchwork, che sfrutta le debolezze dell'occhio umano rispetto alle variazioni della luminosità. Le procedure di data hiding nei file audio, infine, sono analizzate attraverso lo studio di come un segnale audio viene immagazzinato e di come esso viene trasmesso. Anche in questo caso, esamineremo le principali procedure: l'inserzione nei bit meno significativi, la codifica delle fasi, la codifica mediante lo spread spectrum e l'occultamento dei dati nell'eco. Queste procedure di data hiding hanno un comune denominatore, il fatto che l'informazione nascosta debba

essere contenuta in un oggetto che non sollevi alcun sospetto. Per esempio, un sito di notizie on line potrebbe utilizzare tecniche steganografiche per celare nelle immagini della pagina web elementi che riguardano i diritti d'autore, consentendo così ai creatori del sito web di scovare eventuali copie delle stesse immagini presenti in rete non dotate di una firma digitale che le identifichi.

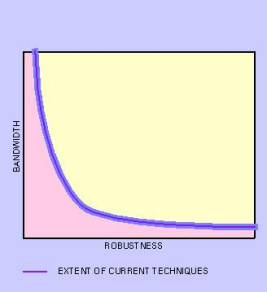

Figura 4: curva

Un'ultima caratteristica comune riguarda il fatto che l'affidabilità di un algoritmo steganografico è inversamente proporzionale alla quantità di dati che è in grado di nascondere senza che tale modifica diventi visibile o comunque percettibile per il nemico. In altre parole, se un metodo di occultamento di dati è robusto ed in grado di resistere a diversi tipi di manipolazione, allora esso potrà supportare la presenza di una piccola quantità di informazioni segrete; al contrario, una procedura che consente di celare grandi moli di dati, è meno affidabile. Il grafico qui sopra rappresenta questo rapporto. In generale, vale la formula:

(quantità di dati da nascondere)x(affidabilità della tecnica di occultamento)
= costante

4 Steganografia testuale

4.1 Il codice di Trithemius

Il primo libro sulla crittologia, *Polygraphiae libri sex* (Sei libri di poligrafia), fu scritto da Johannes Trithemius (1462-1516), un abate tedesco che si rivelò uno degli intellettuali più importanti del suo tempo; i sei tomi furono pubblicati dopo la morte dell'autore, nel 1518.

Figura 5: Trithemius

Il primo libro contiene 384 colonne di parole latine, due colonne per ogni pagina: ogni parola rappresenta una lettera dell'alfabeto. Qui sotto forniamo una parte della prima pagina:

a	Deus	a	clemens
b	Creator	b	clementissimus
c	Conditor	c	pius

.....

Figura 6: Steganografia generativa

Così, sostituendo tali parole al posto delle lettere che compongono un messaggio segreto, si ottengono dei passaggi che suonano come innocue preghiere: ad esempio, cifrando le lettere della parola *abbot* (abate), si ottiene la seguente frase in latino: *Deus clementissimus regens aevum infinivet.* I restanti libri di *Polygraphia* introducono ulteriori schemi crittografici ed ingegnose tecniche di occultamento delle informazioni. Ma *Polygraphia* non fu l'unica opera di Trithemius: nel 1499, egli compose un volume cifrato, molto controverso, intitolato *Steganographia*, una trilogia che per anni circolò privatamente in forma manoscritta e fu infine pubblicata solo nel 1606. Fu poi collocata nella lista ufficiale dei libri proibiti nel 1609, con il pretesto che spiegava come servirsi degli spiriti per inviare messaggi segreti.

I primi due libri di *Steganographia* contengono numerosi metodi per nascondere messaggi all'interno di altre scritture più composite e talvolta prive di significato: ad esempio, nella frase `PARAMESIEL OSHURMI DELMUSON THAFLOIN PEANO CHARUSTREA MELANY LYAMUNTO` ... la prima parola specifica il sistema crittografico che deve essere usato, dopodiché a partire dal secondo termine, leggendo una lettera no ed una sì e saltando una parola, si ottiene la seguente espressione: *`Sum tali cautela ut ...`*

La terza parte della trilogia, di cui nella figura qui sotto forniamo un campione, è invece apparentemente un libro di astrologia occulta: è composta da tabelle di numeri, le cui colonne sono sovrastate da simboli zodiacali e planetari, che appaiono come dati astronomici, appunto.

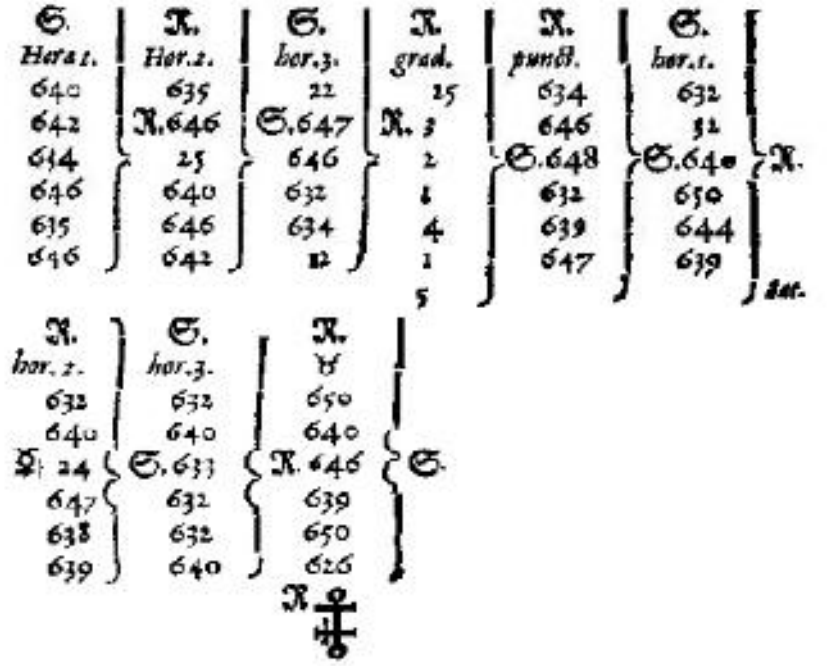

Figura 7: parte del libro terzo della trilogia Steganographia

Per secoli, gli studiosi hanno discusso sul fatto che il terzo libro di *Steganographia* contenesse o meno messaggi segreti; molti hanno concluso che era soltanto un manuale di operazioni magiche, mentre due ricercatori, Thomas Ernst e Jim Reeds erano convinti che il terzo tomo nascondesse dati in codice. Ernst, attualmente professore di tedesco a *La roche College* di Pittsburgh, risolse l'enigma quando era uno studente all'Università di Pittsburgh e scrisse un articolo in tedesco, descrivendo la sua soluzione, che fu pubblicata sulla rivista olandese Daphnis nel 1996, ma che non suscitò l'interesse dovuto.

Jim Reeds, un matematico crittoanalista della *AT&T Labs*, a Florham Park, New Jersey, assunse come ipotesi iniziale il fatto che le tabelle numeriche dovessero essere lette verticalmente e suddivise inoltre le righe successive in blocchi di 25 numeri ciascuno: fu una fortunata intuizione. Egli riscrisse la prima tabella numerica convertendo le colonne in righe, eliminando i simboli non numerici e sostituendoli col segno /. Forniamo qui di seguito un modello di tale scomposizione:

644 650 629 650 645 635 646 636 632 646 639 634 641 642 649 642 648 638 634 647 632 630 642 633 648 650 655 626 650 644

638 633 635 642 632 640 637 643 638 634 / 669 675 654 675 670 660 675 661 651 671 664 659 666 667 674 667 673 663 659

672 657 655 667 658 673 675 660 651 675 669 663 658 660 667 637 665 662 668 663 659 / 694 700 679 700 695 685 696 686

Egli notò che i simboli / dividevano i primi 160 numeri in quattro blocchi di esattamente quaranta numeri ognuno. Inoltre, quasi tutti i numeri di ciascun blocco erano compresi in un particolare intervallo. Reeds scrisse i quattro blocchi da quaranta numeri su quattro righe, una sotto l'altra, per vedere se poteva scorgere qualche similarità nella struttura delle righe:

644	650	629	650	645	635	646	636	632	646	...
669	675	654	675	670	660	675	661	651	671	...
694	700	679	700	695	685	696	686	632	696	...
719	725	704	725	720	710	721	711	707	721	...

Si accorse che, tranne alcune eccezioni, sommando 25 ai numeri della prima riga, otteneva i numeri della riga sottostante. Calcolò poi la frequenza di apparizione di ciascuno dei 25 numeri di ogni riga:

626	1	631	0	636	1	641	1	646	2
627	0	632	3	637	1	642	4	647	1
628	0	633	2	638	3	643	1	648	2
629	1	634	3	639	1	644	2	649	1
630	1	635	2	640	1	645	1	650	4

Ulteriori calcoli sperimentali rivelarono un alfabeto di 22 lettere capovolto: $650 = A$, $649 = B$, e così via. Ottenne il seguente alfabeto: A, B, C, D, E, F, G, H, I, L, M, N, O, P, Q, R, S, T, U, X, Y, Z ed infine le tre lettere greche α (alpha), β (beta) e γ (gamma). Applicata ai 40 numeri della prima riga, questa cifratura conduceva al seguente messaggio: *gazafrequenslibicosduyitca?[gamma]agotriumphos*. Altri indizi spinsero Reeds nella giusta direzione al fine di svelare lo schema usato da Trithemius: ad esempio, il simbolo β rappresenta la sequenza di lettere *sch* nella lingua tedesca e la lettera che lo studioso aveva identificato con x era in realtà w. Egli scoprì infine che la α stava per il digramma *tz* e che γ stava per il *th*.

Un colpo di fortuna finale completò la rottura del codice di Trithemius. Un giorno, mentre il matematico stava ricercando su Internet le due parole *gaza frequens*, s'imbatté proprio nel seguente passaggio latino: Gaza frequens Libycos duxit Carthago triumphos, che confermò che la lettera γ rappresentava il *th* e suggerì che la lettera y che compariva nell'alfabeto cifrato di Reeds era in realtà una x.

Egli poté così concludere che si trattava di un cifrario di sostituzione numerica, con equivalenti numerici multipli per ogni lettera del testo in chiaro. Nella seguente figura forniamo la tabella originale alla base della cifratura di Trithemius.

Th	Sch	Tz	Z	X	W	U	T	S	R	Q	P
01	02	03	04	05	06	07	08	09	10	11	12
26	27	28	29	30	31	32	33	34	35	36	37
51	52	53	54	55	56	57	58	59	60	61	62
76	77	78	79	80	81	82	83	84	85	86	87

O	N	M	L	I	H	G	F	E	D	C	B	A
13	14	15	16	17	18	19	20	21	22	23	24	25
38	39	40	41	42	43	44	45	46	47	48	49	50
63	64	65	66	67	68	69	70	71	72	73	74	75
88	89	90	91	92	93	94	95	96	97	98	99	00

Figura 8: Tabella della cifratura di Trithemius

Ma in cosa consisteva di preciso l'idea di Tritemio? Supponiamo per un istante di voler mandare un messaggio ad un nostro amico Tizio, avvertendolo di non fidarsi di un certo Caio. Abbiamo però paura che Caio legga tutta la nostra corrispondenza e quindi, usando la Steganografia, scriviamo il seguente testo: *"**N**elle **o**re **n**otturne **f**eroci **i**llusioni **d**i **a**ntichi **r**iti **t**ramandati **i**n **d**imenticate **i**sole **c**i **a**ssalgono, **i**vi **o**ra ..."*. Tizio, per leggere il messaggio originale che noi gli volevamo mandare, non dovrà far altro che leggere tutte le iniziali delle parole e comporre il testo nascosto: *"Non fidarti di Caio ..."*. Questo è l'esempio più semplice di tutti gli schemi proposti da Tritemio, che elaborò 40 sistemi principali e 10 sotto-sistemi secondari, sfruttando non solo varie combinazioni di acronimi, ma anche usando dei dischi rotanti (come quello riportato qui di seguito) basati sulla sostituzione mono-alfabetica di Cesare.

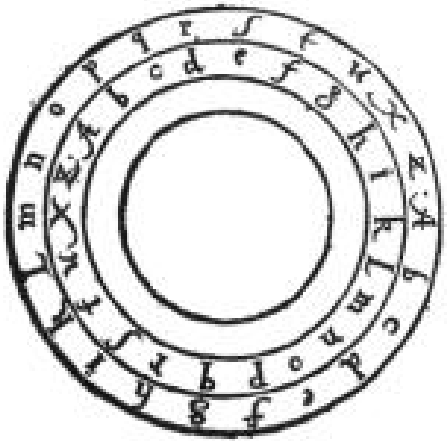

Figura 9: Disco rotante

In questa cifratura il posto di ogni lettera del messaggio è preso dalla lettera che si trova ad una distanza di x posti nell'alfabeto ordinario, dove x, nel caso dell'alfabeto completo di 26 lettere, è un numero compreso tra 1 e 25. Il messaggio *"Mio **z**io è **a**ndato a **Z**urigo non **p**er un **s**emplice incontro **n**otturno di **k**arate, quindi **d**omani si **f**arà il **s**olito giretto **n**el centro **s**torico. Dovrebbe **m**andarmi un **k**imono per **s**abato, e **a**llora..."*, prendendo una parola si e una no, nasconde la stringa, a prima vista illeggibile, *"zazpsnkdfsnsmksa"*. Usando però la trasposizione del disco riportato qui sopra, che quindi tramuta le 'a' in 'o', le 'b' in 'p', le 'c' in 'q' ecc ..., ecco che riusciamo ad ottenere il solito *"nonfidartidicaio"*. L'idea fondamentale di Tritemio era quindi quella di nascondere un testo segreto dentro un messaggio che funzionasse come copertura, senza quindi ricorrere a brutali sistemi fisici come ad esempio la rasatura di capelli ma sfruttando invece abili artifizi matematici e letterari. A meno di non sapere il sistema usato per nascondere il testo, era dunque praticamente impossibile riuscire a estrarre il significato reale del messaggio. Unico inconveniente: il mittente ed il destinatario dovevano avere entrambi il libro di Tritemio per poter conoscere il sistema steganografico usato ...

4.2 Steganografia testuale al giorno d'oggi

Moderni mezzi di comunicazione, come la posta elettronica, permettono a contraffattori di distribuire illegalmente sulla rete copie identiche di documenti originali, senza pagare i diritti d'autore. Per neutralizzare questo tipo di pirateria, esiste un metodo per contrassegnare documenti imprimibili con un'unica parola cifrata indistinguibile per i lettori, ma che può essere usata per identificare il ricevente designato di un documento. Le tecniche qui descritte sono progettate per essere usate come aggiunta alle normali misure di sicurezza: infatti, i documenti dovrebbero comunque essere cifrati prima di essere trasmessi lungo la rete. Un vantaggio di tali procedure è che esse non sono inclini a subire alterazioni durante operazioni di copiatura, quali ad esempio la fotocopiatura, e questa peculiarità è utile per poter facilmente risalire dalle copie alla sorgente originale. L'idea di base è che una parola cifrata, come ad esempio un numero in forma binaria, sia nascosta in un documento, alterando così alcune caratteristiche del testo. Descriviamo ora tre tipi di codifica, detti metodi **open-space** (spaziatura libera), perché sfruttano gli spazi bianchi presenti tra una parola e la successiva per nascondervi informazioni segrete: essi modificano gli spazi fra i termini o gli spazi alla fine di ogni frase.

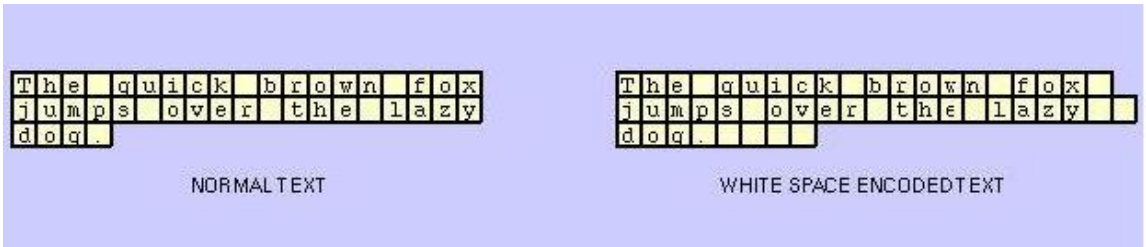

Figura 10: Esempio di dati nascosti con l'inserzione di spazi bianchi

La variazione degli spazi fra le parole di un testo può essere utile per nascondervi messaggi binari, ad esempio: nella figura soprastante, l'aggiunta di un singolo spazio potrebbe codificare uno 0, mentre l' incremento di due potrebbe rappresentare un 1. Questa procedura evidenzia subito la propria inefficienza però, in quanto avremo bisogno di un testo molto lungo per potervi celare solo pochi bit di informazioni segrete. Non tutti gli spazi fra le parole possono però essere utilizzati per archiviarvi dei dati: così, per determinare quali spaziature contengono bit segreti e quali invece fanno parte del testo originale, viene sfruttato il metodo Manchester, che raggruppa i bit a due a due, interpretando le sequenze in questo modo:

$$01 \Rightarrow 0 \qquad 10 \Rightarrow 1 \qquad 00 \Rightarrow NULL \qquad 11 \Rightarrow NULL$$

Nella figura qui sotto, riportiamo un estratto da "A Connecticut Yankee in King Arthur's Court", di Mark Twain, nella quale sono stati nascosti dei dati utilizzando questo tipo di variazione degli spazi fra i termini: sono segnate in nero le spaziature originali del testo, in rosso quelle modificate per cifrare uno zero ed in violetto quelle variate per codificare un uno.

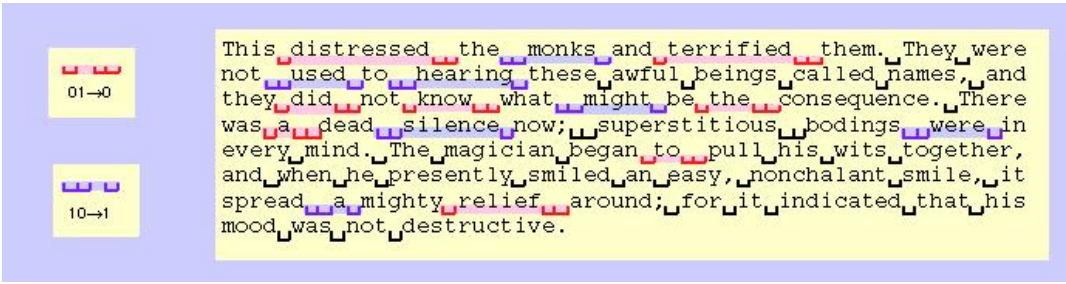

Figura 11: Selezione degli spazi in cui nascondere dati

Lo svantaggio è che una semplice visualizzazione del file su un computer diverso da quello sul quale è stato creato comporterebbe un'automatica formattazione degli spazi del documento e la conseguente modifica del numero delle spaziature provocherebbe la distruzione del messaggio segreto. Ma vediamo in dettaglio le procedure **open-space**:

- **Codifica line-shift** (o spostamento di righe);
- **Codifica word-shift** (o spostamento delle parole);
- **Codifica feature** (codifica delle peculiarità del testo).

Nel primo metodo, le righe di testo vengono spostate verticalmente. Spostando ogni seconda riga-ad esempio-di un file di testo di pochi millimetri in su o in giù, si ottiene una codifica piuttosto efficiente; questa tecnica, però, è certamente la più visibile per un lettore esperto: egli potrebbe infatti effettuare un conteggio manuale oppure automatico del numero di pixel presenti tra una riga e l'altra e scoprire così la manomissione del testo. Un'operazione di rispaziatura casuale o uniforme delle righe del testo danneggerebbe ogni successivo tentativo di rilevare la modifica. Tuttavia, un documento in forma cartacea contrassegnato con tale codifica risulta piuttosto sicuro, in quanto uno stegoanalista dovrebbe riesaminare ogni pagina, apporvi i cambiamenti supposti e ristamparla per verificare l'esattezza delle sue supposizioni. Inoltre, queste operazioni sarebbero ancora più complicate se il testo in esame fosse una fotocopia, nel qual caso si potrebbe incappare anche in macchie o nell'effetto pepe-sale, tipico dei documenti fotocopiati. Nella seconda procedura, le parole in codice vengono cifrate in un documento, trasferendo le posizioni orizzontali delle parole all' interno di righe di testo, mantenendo fra esse l'apparenza di una spaziatura naturale. Il metodo è applicabile esclusivamente a documenti in cui lo spazio fra parole adiacenti è variabile, come i documenti che siano stati adattati al testo che contengono. È dunque necessario conoscere almeno la spaziatura fra i vocaboli nel documento originale. Proponiamo un semplice esempio di come tale tecnica potrebbe funzionare:

si trovino per ogni riga del testo lo spazio maggiore fra due parole e quello minore; al fine di cifrare una riga, la spaziatura maggiore sia ridotta di un certo fattore, mentre la più piccola sia aumentata dello stesso fattore. Questo mantiene la lunghezza della riga e produce un piccolo cambiamento visivo nel testo. Tale metodo di codifica dovrebbe risultare meno visibile rispetto al precedente, in quanto la spaziatura fra vocaboli adiacenti su una riga viene spesso cambiata per avvallare l'adattamento del documento al testo.

Anche questa procedura può essere scoperta ed annullata: se si conoscesse ad esempio l'algoritmo di adattamento del documento al testo, allora si potrebbero misurare gli spazi effettivi tra le parole e, confrontandoli con le spaziature standard, rilevare i dati in codice da tale raffronto. Infine, nell'ultima tecnica di data hiding nei file di testo, alcune caratteristiche del testo vengono alterate, oppure no, a seconda della parola cifrata. Per esempio, si potrebbero nascondere bit di informazione nel testo, allungando o accorciando la barretta verticale di lettere quali "b", "d", "h", ecc. Naturalmente, prima dell'applicazione di questo tipo di codifica, si effettua una scelta casuale dei caratteri da modificare lungo l'intero documento, in modo da evitare un'intuitiva decodifica visiva. Grazie all'incredibile mole di peculiarità di un file di testo che possono essere alterate, questa tecnica è in grado di nascondere una notevole quantità di dati ed è difficilmente distinguibile per un lettore medio. Esistono poi delle tecniche alternative per nascondere dati in un testo; qui di seguito forniamo le più importanti:

- **Metodi sintattici**
- **Metodi semantici**

I primi sfruttano la punteggiatura e le contrazioni grammaticali, mentre i secondi permettono una codifica, manipolando le parole stesse che compongono il testo. Ad esempio, le due frasi qui sotto sono entrambe considerate corrette, sebbene la prima contenga una virgola in più:

$$pane, burro, e \quad latte \qquad\qquad pane, burro \quad e \quad latte$$

L'alternarsi di questi due modi di fare un elenco può essere utilizzato per rappresentare dati in forma binaria. Altri metodi di codifica sintattica coinvolgono l'uso di abbreviazioni grammaticali, ma benché una sintassi ricca di contrazioni sia tipica della lingua inglese (la maggior parte dei documenti in tutto il mondo sono scritti in inglese), in realtà soltanto una scarsa quantità di dati può essere codificata sfruttando questa procedura, circa dell'ordine di pochi bit per ogni kilobyte di testo. L'ultima categoria di occultamento dei dati è rappresentata dai metodi semantici, che utilizzano i sinonimi come "classi di equivalenza" tra le parole, per cifrare un messaggio nascosto, scegliendo una parola al posto di un'altra in una coppia di sinonimi: assegnando dei valori ai sinonimi, i dati segreti possono essere trasformati nelle effettive parole che compongono il testo. Per esempio, al termine *big* potrebbe essere assegnato il valore 1, mentre alla parola *large* il valore 0: ogni volta che si incontrerà il termine *big* nel testo codificato, esso verrà considerato come un 1 nella decodifica. Lo svantaggio che si intuisce in quest'ultima procedura è che talvolta le parole non possono essere scambiate con altre solo approssimativamente equivalenti, senza stravolgere profondamente il senso della frase. Inoltre, talvolta ci sono notevoli differenze a seconda che il testo in analisi sia in inglese formale scritto, o in slang, o che si tratti invece di una conversazione informale, o ancora di sottolinguaggi speciali come quelli utilizzati in matematica o in programmazione: in ognuno di questi casi, una qualunque sostituzione modificherebbe notevolmente il senso delle proposizioni, alterandone le caratteristiche ed attirando subito l'attenzione.

Dunque, entrambe le procedure sono difficilmente utilizzabili in maniera automatica senza risultare visibili: infatti, nella prima, la modifica della punteggiatura è un accorgimento tanto più facilmente individuabile quanto più il testo è lungo, mentre nella seconda non vengono rispettate le sfumature del significato di ciascun termine. É però opportuno evidenziare che l'utilizzo dei metodi sintattici e/o semantici non ostacola l'applicazione delle procedure open-space: di conseguenza, essi possono essere impiegati in parallelo. Esistono molteplici applicazioni delle procedure di data hiding in file di te-

sto, quali ad esempio il controllo del copyright e l'autenticazione: rendere i diritti d'autore inscindibili dal resto del testo è una delle tecniche utilizzate da tutti coloro che vogliono distribuire copie dei loro prodotti in forma elettronica, proteggendone però la legittima proprietà. Inoltre, è possibile ad esempio, impostare un programma di posta elettronica in modo che verifichi l'eventuale esistenza di messaggi nascosti ogni volta che un messaggio viene trasmesso: questo viene respinto o accolto a seconda che venga identificata in esso oppure no la presenza di dati segreti. Questa procedura impedisce che, ad esempio nelle aziende, informazioni riservate vengano inavvertitamente esportate.

Riferimenti bibliografici

[1] 1606. Johannes Trithemius. *Steganographia.*

[2] 1606. Johannes Trithemius. *Clavis Steganographiae.*

[3] 1624. Johannes Trithemius. *Cryptomenytices.*

[4] 1997. Peter Wayner. *Crittografia Invisibile.*

[5] 1999. Simon Singh. *Codici e Segreti.*

[6] http://members.tripod.com/steganography/stego.html

[7] http://www.cl.cam.ac.uk/ fapp2/steganography/

[8] http://www.jjtc.com/Steganography/

[9] http://www.steganos.com/

[10] Ross Anderson. *Stretching the Limits of Steganography.*

[11] 1996. Bruce Schneier. *Applied Cryptography-Protocols, algorithms and source code in C.*

[12] Ivars Peterson. *Cracking a medieval code.*

[13] Neil F. Johnson, Sushil Jajodia. *Steganalysis of images created using current steganography software.*

[14] Frank Sinapsi. *Steganografia.*

[15] http://www-lia.deis.unibo.it/Courses/RetiDiCalcolatori/Progetti00/fortini/project.pdf

[16] 1998. Neil Johnson & Sushil Jajodia. *Exploring Steganography: Seeing the Unseen.*

[17] 1998. Christian Cachin. *An Information-Theoretic Model for Steganography.*

[18] 1998. Ross Anderson & Fabien Petitcolas. *On The Limits of Steganography.*

[19] 1998. Andreas Westfeld & Andreas Pfitzmann. *Attacks on Steganographic Systems.*

[20] http://www.cs.uct.ac.za/courses/CS400W/NIS/papers99/dsellars/stego.htm

[21] 1998. Lisa Marvel & Charles Boncelet & Charles Retter. *Reliable Blind Information Hiding for Images.*

[22] www.Stegoarchive.com