UNIVERSITA' DEGLI STUDI DI PERUGIA
Facoltà di Scienze Matematiche, Fisiche e Naturali
Corso di Laurea Specialistica in Informatica

Seminario Sicurezza Informatica:
Steganografia

Studente
Mancinelli Luca

Docente
Prof. Stefano Bistarelli

STEGANOGRAFIA

Definizione “Steganografia”:

La parola **steganografia** deriva dall'unione dei due vocaboli greci στεγο (rendo occulto, nascondo) e γραφη (la scrittura). Steganografia è dunque "la scrittura nascosta" o meglio ancora 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".

Differenza Steganografia-Crittografia:

Mentre con il termine “Crittografia” 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; con il termine “Steganografia” si vuole nascondere la stessa esistenza della comunicazione agli occhi di eventuali osservatori.
Per rendere più esplicite le differenze tra questi due concetti possiamo osservare che, mentre nel caso della crittografia è consentito al nemico di rilevare, intercettare e modificare i messaggi senza però avere la possibilità di violare le misure di sicurezza garantite dallo specifico sistema crittografico (cioè senza poter accedere all'informazione vera e propria e quindi leggerne il contenuto), l'obiettivo della steganografia è invece quello di nascondere un messaggio **dentro** un altro messaggio, dall'aspetto innocuo, in modo che il nemico non possa neppure rilevare l'esistenza del primo messaggio.

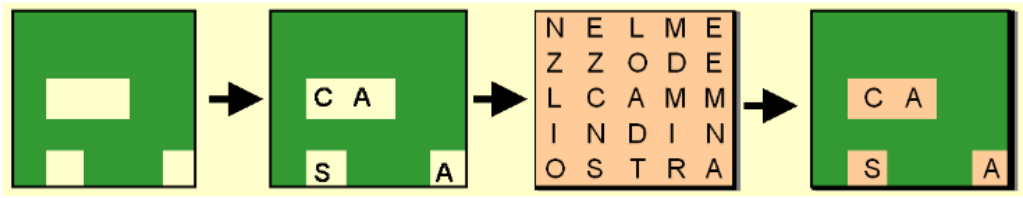
Metodi Steganografici nella Storia:

Nel corso dei secoli sono stati escogitati numerosi metodi steganografici:

1) **Erodoto** racconta la storia 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.

2) Un **acrostico** è una poesia – o un testo di qualsiasi tipo – composta intenzionalmente in modo tale che, unendo le prime lettere di ogni capoverso, si ottiene un messaggio di senso compiuto.

3) Le **griglie di Cardano** erano fogli di materiale rigido nei quali venivano ritagliati fori rettangolari a intervalli irregolari; applicando la griglia sopra 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. Applicando sul foglio una copia esatta della griglia originaria, era possibile leggere il messaggio nascosto.



4) Gli **inchiostri invisibili** (o inchiostri simpatici) sono sostanze che, in condizioni normali, non lasciano tracce visibili se usate per scrivere su un foglio di carta, ma diventano visibili (rivelando in tal modo la scrittura) se il foglio viene sottoposto a una fonte di calore. È così possibile scrivere il messaggio segreto negli spazi compresi tra le righe di un messaggio dall'aspetto innocuo, quest'ultimo scritto con un inchiostro normale. (Per accedere al messaggio segreto occorre letteralmente "saper leggere tra le righe"...). Le sostanze più usate a questo scopo sono molto comuni: succo di limone, aceto, latte, ma durante la seconda guerra mondiale sono state impiegate sostanze più sofisticate, come ad esempio gli inchiostri al cobalto, che possono essere resi visibili solo mediante l'uso di particolari reagenti chimici.

5) **Micropunti fotografici**: 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.

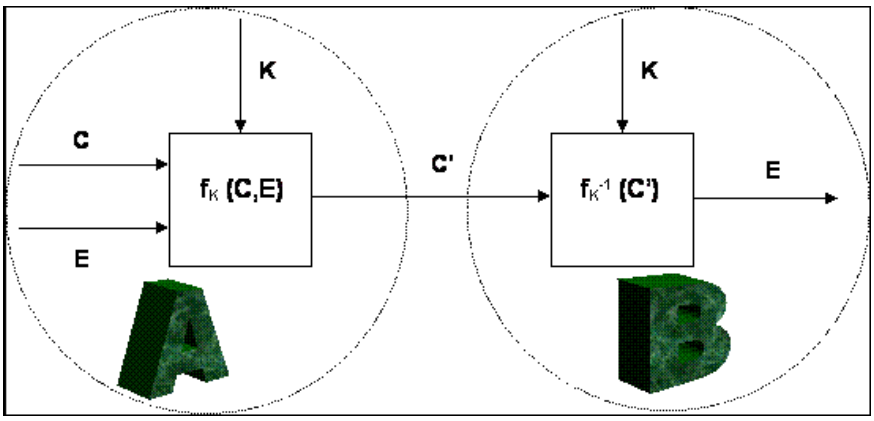
6) **Le immagini di Al-Queda** :Anche oggi la steganografia viene utilizzata come veicolo politico-militare. Ad esempio, nel famoso quotidiano americano "USA Today" del 10 luglio del 2002 si legge: "Ultimamente al-Queda ha inviato centinaia di messaggi crittografati nascosti in fotografie digitali sul 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".

Il sistema steganografico.

Una tecnica adatta a nascondere dati come la steganografia non poteva che essere utilizzata nel digitale. Per cui, quando si parla di sistema steganografico si fa implicito riferimento a questo tipo di informazioni. Il sistema steganografico (stegosistema) prevede che due utenti, A e B, vogliano scambiarsi dati segreti. Per fare ciò, nascondono tali dati in frammenti di informazione che risultano innocui a chi riesca ad intercettarli. I dati da nascondere costituiscono il messaggio segreto (**embedded**) e vengono nascosti (incapsulati) in un altro frammento di informazione detto contenitore (**cover**) dando origine al **frammento stego** che è, appunto, una copia molto simile ma non identica del contenitore.

Per effettuare l'incapsulamento (**embedding**) A utilizza una funzione steganografica **f** che, presi in input il contenitore, il messaggio segreto e una chiave segreta, produce il frammento stego. Una volta effettuato l'incapsulamento, A invia il frammento stego a B che lo riceve e, conoscendo la chiave segreta e la funzione di estrazione f^{-1} (inversa di f), riesce a risalire al messaggio segreto partendo dal frammento stego appena ricevuto.

Il sistema steganografico può essere così schematizzato:



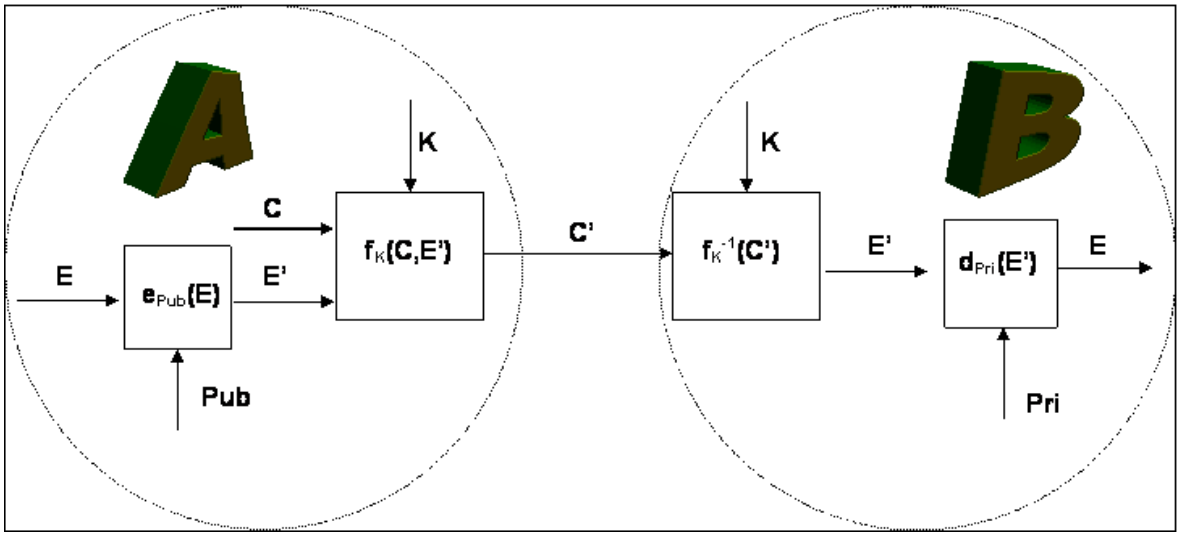
dove:

- **E** è il messaggio segreto da nascondere;
- **C** è il contenitore;
- **C'** è il frammento stego ottenuto incapsulando E in C;
- **K** è la chiave segreta che A e B devono conoscere;
- **f_K(C,E)** è la funzione steganografica che nasconde E in C usando la chiave K;
- **f_K^{-1}(C')** è la funzione inversa di f che, sfruttando la chiave K e partendo dal frammento stego C' ricevuto, riesce a risalire ad E.
  Se la chiave fornita è la stessa usata dal mittente per nascondere il messaggio segreto e se C' è lo stesso frammento prodotto dal mittente (potrebbe essere stato modificato da un **attaccante**), allora la funzione di estrazione produrrà effettivamente il messaggio segreto originale E.

Steganografia a chiave pubblica

Il sistema steganografico appena visto assume che esista (o sia esistito) un canale sicuro per far si che mittente e destinatario si siano potuti scambiare la chiave segreta.

Assumiamo che A voglia inviare un messaggio segreto a B senza che nessuno lo sappia, e che essi non abbiano modo di scambiarsi una chiave steganografica prima della trasmissione di tale messaggio. Se B possiede una chiave pubblica e A la conosce, A può cifrare il messaggio segreto con la chiave pubblica di B, nascondendo il testo cifrato in un contenitore e mandando il risultante frammento stego a B. B può estrarre il testo cifrato dal frammento stego ricevuto e decifrarlo con la sua chiave privata.



Rispetto allo schema precedente vi è l'aggiunta del concetto di chiave pubblica:

- **Pub** è la chiave pubblica di B che A conosce;
- **$e_{Pub}(E)$** è la funzione di **encoding** che prende in input la chiave pubblica di B (Pub) e il messaggio nascosto E da cifrare: il risultato di tale operazione è il messaggio E', che costituirà l'input per l'algoritmo di steganografia vero e proprio;
- **E'** è il messaggio segreto e cifrato che viene incapsulato all'interno del cover C;
- **Pri** è la chiave privata di B, grazie alla quale si riesce a risalire ad E partendo da E'.
- **$d_{Pri}(E')$** è la funzione di **decoding** che decifra il messaggio segreto e cifrato E', sfruttando la chiave privata di B (Pri) e che dà in output il messaggio segreto E;

Perché questo metodo funzioni, ognuno ha bisogno di sapere come estrarre il messaggio segreto da un potenziale file stego. Questo algoritmo di estrazione può essere applicato anche ai files che non contengono messaggi nascosti: infatti, non è importante che un file contenga un messaggio segreto o meno, il risultato sarà comunque una stringa di dati random che solo B sarà capace di decifrare con successo.

L'**inconveniente** di questo sistema sta nel fatto che ogni qualvolta si riceve un potenziale file stego si deve estrarre il potenziale testo cifrato e provare a decifrarlo con la propria chiave privata (senza essere sicuri di trovare un messaggio). Più che uno scambio di messaggi, diventa una caccia al tesoro!

Tecniche steganografiche:

Ciò che caratterizza la steganografia, come si è visto, è l'esistenza di un secondo messaggio facilmente percepibile, il cui senso è generalmente del tutto disgiunto da quello del messaggio segreto che esso contiene. Nel seguito si indicherà questo secondo messaggio come **messaggio contenitore** o più semplicemente **contenitore**.

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. Le prime definizioni proposte riguardano l'origine del file contenitore: alcune tecniche consentono di "iniettare" il messaggio segreto dentro un messaggio contenitore già esistente, modificandolo in modo tale da contenere sia il messaggio originale che il messaggio “nascosto”, rendendolo praticamente indistinguibile dall'originale. Indichiamo l'insieme di queste tecniche con il termine **steganografia iniettiva**. 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**.

Secondo un sistema di classificazione diverso, le tecniche steganografiche possono essere ripartite in tre classi:

- **steganografia sostitutiva**
- **steganografia selettiva**
- **steganografia costruttiva**

Steganografia Sostitutiva

Le tecniche di **stefanografia sostitutiva** sono di gran lunga le più diffuse. Tali tecniche si basano sulla seguente 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 un'immagine prodotta da uno scanner è soggetta a essere affetta da errore.

La tecnica base impiegata dalla maggior parte dei programmi, consiste semplicemente nel sostituire i "**bit meno significativi**" delle immagini digitalizzate con i bit che costituiscono il file segreto . Spesso l'immagine che ne risulta non è distinguibile a occhio nudo da quella originale ed è comunque difficile dire se eventuali perdite di qualità siano dovute alla presenza di informazioni nascoste oppure all'errore causato dall'impiego di uno scanner poco preciso, o ancora alla effettiva qualità dell'immagine originale prima di essere digitalizzata.

Esempio Steganografia Sostitutiva.

Uno dei modi in cui viene solitamente rappresentata un'immagine prodotta da uno scanner è la codifica **RGB** a 24 bit: l'immagine consiste di una matrice MxN di punti colorati (pixel) e ogni punto è rappresentato da 3 byte, che indicano rispettivamente i livelli dei colori rosso (Red), verde (Green) e blu (Blue) che costituiscono il colore. Supponiamo che uno specifico pixel di un'immagine prodotta da uno scanner sia rappresentato dalla tripla (12, 241, 19) (si tratta di un colore tendente al verde, dato che la componente verde predomina fortemente sulle altre due); in notazione binaria, le tre componenti sono:

==========

12 = 00001100

241 = 11110001

19 = 00010011

==========

quelli che in precedenza abbiamo chiamato i "bit meno significativi" dell'immagine sono gli ultimi a destra, cioè 0-1-1, e sono proprio quelli che si utilizzano per nascondere il messaggio segreto. Se volessimo nascondere in quel pixel l'informazione data dalla sequenza binaria 101, allora bisognerebbe effettuare la seguente trasformazione:

==========

00001100 --> 00001101 = 13

11110001 --> 11110000 = 240

00010011 --> 00010011 = 19

==========

La tripla è così diventata (13, 240, 19); si noti che questo tipo di trasformazione consiste nel sommare 1, sottrarre 1 o lasciare invariato ciascun livello di colore, quindi il colore risultante differisce in misura minima da quello originale. Dato che un solo pixel può contenere un'informazione di 3 bit, un'immagine di dimensioni MxN può contenere un messaggio segreto lungo fino a (3*M*N)/8 byte, per esempio un'immagine 1024x768 può contenere 294912 byte.

PROBLEMA CON IL FORMATO **JPEG**:

La tecnica appena descritta rappresenta il cuore della steganografia sostitutiva, anche se di fatto ne esistono numerose variazioni. Innanzitutto è ovvio che tutto quello che abbiamo detto vale non solo per le immagini, ma anche per altri tipi di media, per esempio suoni e animazioni digitalizzati. Inoltre – e questo è meno ovvio – lavorando con le immagini come file contenitori non sempre si inietta l'informazione al livello dei pixel, ma si è costretti a operare su un livello di rappresentazione intermedio; è questo il caso, per esempio, delle immagini in formato JPEG, nel quale le immagini vengono memorizzate solo dopo essere state compresse con una tecnica che tende a preservare le loro caratteristiche visive piuttosto che l'esatta informazione contenuta nella sequenza di pixel. Se iniettassimo delle informazioni in una bitmap e poi la salvassimo in formato JPEG, le informazioni andrebbero perdute, poiché non sarebbe possibile ricostruire la bitmap originale. Per poter utilizzare anche le immagini JPEG come contenitori, è tuttavia

possibile iniettare le informazioni nei coefficienti di Fourier ottenuti dalla prima fase di compressione.

In generale il meccanismo alla base delle Serie di Fourier viene utilizzato ogni volta che devo comprimere o trasmettere le informazioni. Ad esempio per creare file **.pdf** o**.jpg** a partire da immagini, o per generare un file **.mp3**.

PROBLEMA CON IL FORMATO **GIF**:

Esiste un altro caso interessante che merita di essere discusso, ed è quello dei formati di immagini che fanno uso di **palette**. La palette (tavolozza) è un sottoinsieme prestabilito di colori. Nei formati che ne fanno uso, i pixel della bitmap sono vincolati ad assumere come valore uno dei colori presenti nella palette: in questo modo è possibile rappresentare i pixel con dei puntatori alla palette, invece che con la terna esplicita RGB. Ciò in genere permette di ottenere dimensioni inferiori della bitmap, ma il reale vantaggio è dato dal fatto che le schede grafiche di alcuni anni fa utilizzavano proprio questa tecnica e quindi non potevano visualizzare direttamente immagini con un numero arbitrario di colori. Il caso più tipico è quello delle immagini in formato GIF con palette di 256 colori, ma le palette possono avere anche altre dimensioni. Come è facile immaginare, un'immagine appena prodotta da uno scanner a colori sarà tipicamente costituita da più di 256 colori diversi, tuttavia esistono algoritmi capaci di ridurre il numero dei colori utilizzati mantenendo il degrado della qualità entro limiti accettabili. Si può osservare che, allo stesso modo in cui avviene con il formato JPEG, non è possibile iniettare informazioni sui pixel prima di convertire l'immagine in formato GIF, perché durante il processo di conversione c'è perdita di informazione (osserviamo anche che questo non vale per le immagini a livelli di grigi: tali immagini infatti sono particolarmente adatte per usi steganografici.) La soluzione che viene di solito adottata per usare immagini GIF come contenitori è dunque la seguente: si riduce il numero dei colori utilizzati dall'immagine a un valore inferiore a 256 ma ancora sufficiente a mantenere una certa qualità dell'immagine, dopodiché si finisce di riempire la palette con colori molto simili a quelli rimasti. A questo punto, per ogni pixel dell'immagine, la palette contiene più di un colore che lo possa rappresentare (uno è il colore originale, gli altri sono quelli simili ad esso che sono stati aggiunti in seguito), quindi abbiamo una possibilità di scelta. Tutte le volte che abbiamo una possibilità di scelta fra più alternative, abbiamo la possibilità di nascondere un'informazione: questo è uno dei principi fondamentali della steganografia. Se le alternative sono **due** possiamo nascondere **un bit** (se il bit è 0, scegliamo la prima, se è 1 la seconda); se le alternative sono **quattro** possiamo nascondere **due bit** (00 -> la prima, 01 -> la seconda, 10 -> la terza, 11 -> la quarta) e così via.

La soluzione appena discussa dell'utilizzo di GIF come contenitori è molto ingegnosa ma purtroppo presenta un problema: è facile scrivere un programma che, presa una GIF in ingresso, analizzi i colori utilizzati e scopra le relazioni che esistono tra di essi; se il programma scopre che l'insieme dei colori utilizzati può essere ripartito in sottoinsiemi di colori simili, è molto probabile che la GIF contenga informazione steganografata. Di fatto, questo semplice metodo di attacco è stato portato avanti con pieno successo da diverse persone ai programmi che utilizzano immagini a palette come contenitori, tanto che qualcuno ha finito per sostenere che non è possibile fare steganografia con esse.

Sicurezza nelle Tecniche Steganografiche:

Dopo avere esaminato alcune tecniche steganografiche di tipo sostitutivo, discutiamo adesso i problemi relativi alla loro sicurezza. Innanzitutto premettiamo che le norme che valgono generalmente per i programmi di crittografia dovrebbero essere osservate anche per l'utilizzo dei programmi steganografici.

Per ciò che riguarda le specifiche caratteristiche della steganografia, si tengano presente i seguenti principi:

- in primo luogo si eviti di usare come contenitori file prelevati da siti pubblici o comunque noti (per esempio, immagini incluse in pacchetti software, ecc.);
- in secondo luogo si eviti di usare più di una volta lo stesso file contenitore (l'ideale sarebbe quello di generarne ogni volta di nuovi, mediante scanner e convertitori da analogico a digitale, e distruggere gli originali dopo averli usati).

Come si è visto, queste tecniche consistono nel sostituire un elemento di scarsa importanza (in certi casi di importanza nulla) da file di vario tipo, con il messaggio segreto che vogliamo nascondere. Quello che viene ritenuto il principale difetto di queste tecniche è che in genere la sostituzione operata può alterare le caratteristiche statistiche del rumore presente nel media utilizzato. Lo scenario è il seguente: si suppone che il nemico disponga di un modello del rumore e che utilizzi tale modello per controllare i file che riesce a intercettare. Se il rumore presente in un file non è conforme al modello, allora il file è da considerarsi sospetto. Si può osservare che questo tipo di attacco non è per niente facile da realizzare, data l'impossibilità pratica di costruire un modello che tenga conto di tutte le possibili sorgenti di errori/rumori, tuttavia in proposito esistono degli studi che in casi molto specifici hanno avuto qualche successo.

La steganografia selettiva e quella costruttiva hanno proprio lo scopo di eliminare questo difetto della steganografia sostitutiva.

Steganografia selettiva

La steganografia selettiva ha valore puramente teorico e, per quanto se ne sappia, non viene realmente 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. Facciamo un esempio per chiarire meglio. 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.

Steganografia costruttiva

La steganografia costruttiva affronta lo stesso problema nel modo più diretto, tentando di sostituire il rumore presente nel medium 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.

SVANTAGGI:
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 uno 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.

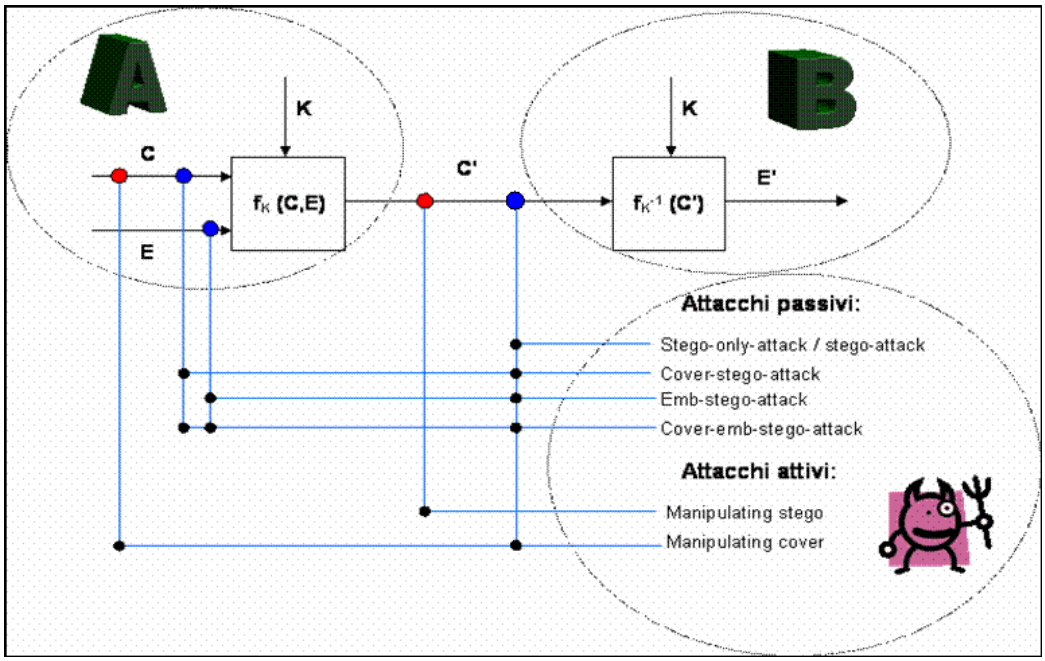
Stegoanalisi

Come con la crittoanalisi per la crittografia, la stegoanalisi è definita come la scienza (nonché l’arte) del rompere la sicurezza di un sistema steganografico. Siccome lo scopo della steganografia è di nascondere l’esistenza di un messaggio segreto, un attacco con successo ad uno stegosistema consiste nello scoprire che un determinato file contiene dati nascosti anche senza conoscerne il loro significato.

Come in crittoanalisi, si assume che il sistema steganografico sia conosciuto dall’attaccante e quindi che la sicurezza dello stegosistema dipenda dal solo fatto che la chiave segreta non è nota all’attaccante (principio di Kerckhoff).

Lo stegosistema esteso

Lo stegosistema visto in precedenza può essere esteso per includere situazioni di attacchi simili agli attacchi crittografici. Nel diagramma seguente un cerchio (rosso o blu) indica un punto in cui un attaccante può avere accesso: i punti in cui l'attaccante ha accesso definiscono il tipo di attacco. C'è una distinzione da fare tra attacchi attivi e attacchi passivi: mentre nel primo tipo gli attaccanti riescono solo ad intercettare i dati (nel diagramma, cerchio blu), nel secondo riescono anche a manipolarli (cerchio rosso).



Ecco in cosa consistono gli attacchi:

- **stego-only-attack**: l'attaccante ha intercettato il frammento stego ed è in grado di analizzarlo. è il più importante tipo di attacco contro il sistema steganografico perché è quello che occorre più di frequente nella pratica;
- **stego-attack**: il mittente ha usato lo stesso cover ripetutamente per nascondere dati. L'attaccante possiede un frammento stego diverso ma originato dallo stesso cover. In ognuno di questi frammenti stego è nascosto un diverso messaggio segreto;
- **cover-stego-attack**: l'attaccante ha intercettato il frammento stego e sa quale cover è stato usato per crearlo. Ciò fornisce abbastanza informazioni all'attaccante per poter risalire al messaggio segreto;
- **cover-emb-stego-attack**: l'attaccante ha "tutto": ha intercettato il frammento stego, conosce il cover usato e il messaggio segreto nascosto nel frammento stego;
- **manipulating the stego data**: l'attaccante è in grado di manipolare i frammenti stego. Il che significa che l'attaccante può togliere il messaggio segreto dal frammento stego (inibendo la comunicazione segreta);
- **manipulating the cover data**: l'attaccante può manipolare il cover e intercettare il frammento stego. Questo può significare che con un processo più o meno complesso l'attaccante può risalire al messaggio nascosto.

Lo **stego-attack** e il **cover-stego-attack** possono essere prevenuti se il mittente agisce con cautela. Un utente non dovrebbe mai usare come cover più volte lo stesso file, né files facilmente reperibili (es. logo di Yahoo) o di uso comune (es. file audio dell'avvio di Windows).