Steganografia nei file Audio

Simona Di Ienno

1. La steganografia nei file audio

La steganografia nelle comunicazioni audio viene utilizzata sin dai tempi delle guerre mondiali, per via dell'esigenza di comunicazioni sicure. Queste tecniche sono arrivate fino ai giorni nostri in quanto i principali canali di comunicazione (linee telefoniche, trasmissioni radio, ...) trasmettono segnali accompagnati da qualche tipo di rumore che ben si presta a nascondere messaggi steganografati.

Il segnale audio è per sua natura un segnale analogico, vale a dire un segnale che varia in modo continuo nel tempo. Non ci sono salti bruschi. Inoltre l'ampiezza può assumere un'infinità di valori passando da uno minimo ad uno massimo. Questa onda non è riproducibile sui computer moderni per il fatto che è una grandezza continua e i computer gestiscono invece l'informazione digitalmente.

E' dunque possibile rappresentare un segnale audio, ma bisogna campionarlo, ossia prelevarne, ad intervalli regolari, il valore, che si presenta sotto forma di un segnale elettrico che varia nel tempo.

La funzione non sarà più continua, ma assumerà dei valori numerici discreti, e sicuramente presenterà dei salti fra un valore e l'altro, poiché abbiamo assegnato un singolo valore ad un intervallo di tempo che prima ne presentava infiniti.

In quest'operazione viene inevitabilmente introdotto del rumore, legato al fatto che l'approssimazione della curva analogica originale tramite una spezzata non è perfetta.

Il segnale da considerare perfetto sarà quello di qualità CD, con passo di campionamento pari a 44100 Hz e quantizzazione dei campioni a 16 bit.

Le ridondanze che si creano nei file audio digitali li rendono adatti a recitare il ruolo di cover. Come per le immagini, la tecnica più usata per nascondere messaggi segreti nei files audio consiste nel rimpiazzare i bits meno significativi. Lo "spazio" che le tecniche steganografiche utilizzano per nascondere i file nasce da come il file audio digitale viene manipolato per essere memorizzato su disco in modo da renderlo più compatto.

Nel corso degli ultimi anni abbiamo assistito ad una grossa rivoluzione tecnologica nel campo della compressione audio.

L'obiettivo di queste tecniche è quello di ridurre lo spazio necessario ad immagazzinare determinati dati o la banda necessaria per trasmetterli. Nel comprimere dati come testi, documenti o programmi non ci si può permettere la perdita di nessun bit di informazione, per cui dovremo utilizzare necessariamente tecniche lossless come quelle adottate dallo Zip, nel caso dell'audio, delle immagini e dei filmati, un certo livello di degradazione è un compromesso accettabile per ridurre (e di molto) l'occupazione o la banda richiesta dal file. Le tecniche di compressione audio sono di tipo lossy.

Le procedure di compressione dell'audio sono numerose ed utilizzano tecniche anche molto differenti l'una dall'altra. Esistono però tre categorie principali:

- **codifica nel dominio del tempo**,
- **codifica per modelli,**
- **codifica nel dominio delle frequenze**.

Generalmente, le prime due categorie di algoritmi vengono usate per comprimere il segnale vocale, mentre alla terza appartengono algoritmi come **MP3**, **WMA**, **ATRAC-3** e **AAC**, ottimi per la compressione della musica.

La codifica nel dominio del tempo è un algoritmo che elabora il segnale campionato direttamente, senza estrarre le informazioni spettrali (frequenze).

La codifica per modelli è una tecnica legata ad una particolare sorgente sonora (in questo caso la voce), che si tenta di emulare tramite un modello più o meno semplificato.

Infine, l'ultimo tipo di codifica analizzata è caratterizzato dal fatto di esaminare il segnale non nel dominio del tempo, ma nel dominio della frequenza.

Fortunatamente anche il nostro orecchio non è perfetto e questo è un gran vantaggio. In prima analisi esso è sensibile in misura diversa alle diverse frequenze. L'orecchio umano è maggiormente sensibile alle frequenze comprese fra 2 e 4 KHz, che richiedono pochissimi dB per essere percepite.

Per percepire quindi suoni molto bassi o alti (nel senso delle frequenze), abbiamo bisogno che questi suoni siano molto intensi, in sostanza si tagliano le alte frequenze e le bassissime frequenze.

Supponiamo che vengano diffusi un tono alla frequenza di 1 KHz, detto tono maschera, tenuto fisso a 60 dB (volume alto) ed un secondo tono, detto tono di test: risulterà necessario alzare il volume del tono test per riuscire a distinguerlo.

Oltre al mascheramento in frequenza, esiste un altro tipo di mascheramento, quello temporale. Supponiamo di avere al solito due toni, uno forte e l'altro, che gli è vicino in frequenza, piuttosto debole. Sappiamo già che il nostro orecchio sente solo il tono più forte, se improvvisamente il tono maschera cessasse di esistere, non si avvertirebbe subito il tono più debole.

In conclusione, l'effetto complessivo del mascheramento è che molti toni non saranno mai udibili perché collocati nel dominio della frequenza e del tempo troppo vicino a toni forti.

Per questo motivo, risulta importante la scelta della sorgente audio più appropriata, perciò sarà più semplice nascondere informazioni in un file audio di heavy-metal rispetto ad uno di musica leggera.

Quando i segnali audio devono essere trasmessi, bisogna considerare il mezzo di trasmissione attraverso il quale il segnale dovrà essere propagato.

Esistono quattro diversi ambienti di trasmissione:

- ***Ambiente digitale***. Se un file sonoro è copiato direttamente da una macchina ad un'altra, senza mai essere modificato in alcun modo, esso passerà attraverso questo ambiente: il campione del segnale resta lo stesso fra la sorgente e la destinazione.
- ***Ambiente di ricampionatura con aumento/decremento.*** Il segnale viene rimodellato secondo un nuovo tasso di campionatura più alto oppure più basso, ma resta digitale dal principio alla fine. Benché l'ampiezza assoluta e la fase della maggior parte del segnale restino invariate, le caratteristiche temporali del segnale cambiano.
- ***Trasmissione analogica e ricampionatura.*** Questo accade quando il segnale viene convertito in uno stato analogico e rimodellato. L'ampiezza assoluta del segnale, la quantizzazione del campione ed il tasso di campionatura temporale non vengono conservati, mentre in generale la fase resta invariata.
- ***Ambiente "over the air".*** Questo avviene quando il segnale viene "suonato nell'aria" e rimodellato con l'aiuto di un microfono. In tal caso il segnale potrà essere soggetto a possibili modifiche non lineari, che causeranno variazioni di fase e di ampiezza, accumulo di componenti di frequenza diverse, eco, …

La rappresentazione digitale del segnale ed il mezzo di trasmissione dello stesso sono i due principali elementi che devono essere presi in considerazione nell'analisi di una procedura di data hiding in file sonori. La quantità di informazioni che si potranno nascondere dipenderà dal tasso di campionatura e dal tipo di suono che dovrà essere modificato.

Il processo di occultamento dei dati in file audio si può effettuare in diversi modi.

Il primo è il cosiddetto metodo di **codifica nei bit "bassi"**: tale procedura sostituisce i bit di informazione meno significativi di ogni punto campione del segnale con una stringa binaria codificata. Può però introdurre un fastidioso rumore di fondo.

Il maggiore inconveniente di tale metodo è la scarsa resistenza ad eventuali manipolazioni, quali l'introduzione di segnali di disturbo nel canale di trasmissione o la ricampionatura del segnale, a meno dell'utilizzo di tecniche di ridondanza.

Un secondo metodo di data hiding in file audio è la **codifica delle fasi**: tale procedura funziona sostituendo la fase di un segmento iniziale del file audio con una fase di riferimento che rappresenta i dati.

Una terza tecnica è lo **spread spectrum** che diffonde i dati codificati attraverso lo spettro delle frequenze, rendendo difficile scovare le informazioni, a meno di non conoscere la chiave pseudocasuale. Il ricevente dovrà conoscere oltre alla chiave anche i punti di inizio e di fine del messaggio nascosto nel file audio.

Lo spread spectrum introduce un rumore di fondo casuale al suono al di sotto della soglia di percezione; inoltre, opportuni codici di correzione dell'errore garantiscono l'integrità dei dati. La quantità di informazioni che si possono celare mediante questo metodo è di circa 4bps.

L'ultima procedura è l'**echo data hiding** che inserisce le informazioni in un segnale ospite introducendo un'eco. I dati vengono celati modificando tre parametri dell'eco: l'ampiezza iniziale, il tasso di indebolimento del suono ed il ritardo.

Mentre il ritardo tra il suono originale e l'eco diminuisce, i due segnali si mescolano fino a che l'orecchio umano non può più distinguerli.

Usando due diversi ritardi possiamo codificare le cifre 1 o 0. Per codificare più di un bit, il segnale originale viene diviso in parti, su ognuna delle quali viene introdotta un'eco.

1.1. ESEMPIO File WAV

Ad esempio i file sonori WAV in Windows sono immagazzinati usando 8 o 16 bit (che vengono poi eventualmente convertiti da una scheda sonora). Un file da 8 bit implica che i valori possono essere in un range da 0 a 255. Il range dei 16 bit varia invece da 0 a 65535. Tutto quello che un programma steganografico esegue, è distribuire la sequenza di bit che corrispondono al file da nascondere nei bit meno significativi del file sonoro.

Ad esempio, ammettiamo che siano presenti i seguenti otto byte di informazione da qualche parte:

132	134	137	141	121	101	74	38

che in binario diventano:

10000100	10000110	10001001	10001101
01111001	01100101	01001010	00100110

Supponendo di voler nascondere il byte 11010101 (cioè 213) in questa sequenza, il programma semplicemente sostituisce i bit meno significativi di ogni byte con il corrispondente bit estratto dal byte che si vuole nascondere. In questo modo la sequenza originaria è diventata:

133	135	136	141	120	101	74	39

che in binario è:

1000010**1**	1000011**1**	1000100**0**	10001101
0111100**0**	01100101	01001010	0010011**1**

Come si può quindi vedere, i valori del file sonoro sono cambiati, al massimo, di un valore per ciascun byte. Questa differenza non sarà assolutamente percepibile all'orecchio umano.

Un file wav mono, campionato a 44100 Hz a 16 bit, per esempio, indica un file che è stata generata una stringa di 16 bits ogni 1/44100 di secondo. Nel caso di un wav stereo, le stringhe di 16 bits ottenute sono due

Per avere un'idea, un file wav stereo di un minuto è grande:

16 bits x 44100 Hz x 60 sec = 42336000 bits = 5168 Kb circa,

da raddoppiare perché il file è stereo, quindi 10336 Kb.

Se decidessimo di utilizzare i 2 bits meno significativi per ogni stringa di 16 bits otterremmo una disponibilità di

84762000 bits / 16 bits x 2 = 10595250 bits = 1293 Kb circa.

1.2. Usi Illegali

Purtroppo, come per ogni innovazione tecnologica, esiste sempre qualcuno che la utilizza per scopi illegali. In particolare nei file audio vengono nascosti messaggi subliminali in modo che ascoltandoli al contrario vengano fuori altri messaggi rispetto a quelli delle canzoni che vi sono incise. Casi noti sono quelli di canzoni rock che ascoltate al contrario forniscono rituali satanici, ma ne esistono molti altri, come ad esempio alcune canzoni dei Beatles e altri di cui non se ne può stabilire l'autenticità, nella maggior parte dei casi si tratta infatti di pura casualità, altre volte di metodi pubblicitari.

Secondo recenti studi, i messaggi subliminali non sono efficienti in modo assoluto, come alcuni sostengono, in quanto non si possono cambiare le idee di una persona, se questa non lo desidera e non è possibile costringere qualcuno a fare qualcosa che non vuole con la sola forza del pensiero.

Il nostro inconscio dispone infatti di una libera volontà, lo stesso libero arbitrio di cui dispone la nostra mente, forse anche di più. Esso ha la libertà di accettare o rifiutare un messaggio, sia che si tratti di un ordine, che di un invito, o ancora di un avvertimento.

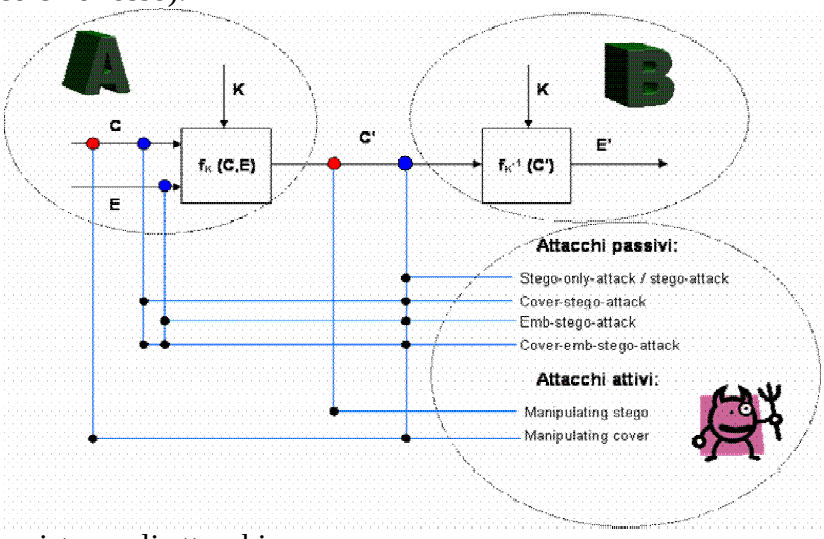
2. La steganalisi

Mentre un crittanalista applica la crittanalisi per cercare di decodificare o rompere messaggi criptati, lo steganalista è colui che applica la steganalisi cercando di determinare l'esistenza di eventuali informazioni nascoste. In crittografia si poteva partire dal confronto di porzioni di testo in chiaro e parti di testo cifrato. Con la steganografia invece, il confronto può essere fatto tra il testo di copertura, il messaggio steganografato ed eventuali porzioni di testo segreto.

Uno stegosistema sicuro potrebbe essere un sistema nel quale nessun intruso possa accorgersi di differenze tra il testo di copertura C e il testo steganografato S contenente l'informazione nascosta E, a meno che non conosca la chiave K.

Questo stegosistema 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 o di uso comune.

3. Future Applicazioni

Attualmente si sta lavorando al concetto di Steganografia di Internet, e cioè alla possibilità di nascondere informazioni ed eventualmente essere in grado di recuperarle dagli header dei pacchetti TCP/IP o da altre trasmissioni di rete. Non bisogna dimenticare che anche se la steganografia e la crittografia sono discipline indipendenti, possono essere impiegate per alterare ed occultare il medesimo testo, garantendo un livello di sicurezza molto più alto.

4. Conclusione

Si potrebbe obiettare che se dei criminali potessero nascondere le informazioni in modi così efficaci la difesa dei cittadini diventerebbe più difficile, ma a questo si può rispondere dicendo che i disonesti non obbediranno mai ad una legge che imponga loro di non utilizzare alcuna forma di steganografia o altro, il problema fondamentale è fornire all'onesto cittadino gli strumenti che gli consentiranno di proteggere la propria privacy: la crittografia e la steganografia.