CORSO DI SICUREZZA

Laurea Magistrale in Informatica

Facoltà di Scienze Matematiche Fisiche e Naturali

Università degli Studi di Salerno

Via Ponte Don Melillo 84081

Fisciano (SA) - Italy

Distribuzioni Linux per l'analisi forense: post-mortem forensic

Giugno 2011

Umberto Annunziata 0522500013

Claudio Gargiulo 0522500011

Prof. del corso: Alfredo De Santis

Sommario

Introduzione

Per “computer forensic” si intende la scienza che mira al recupero e all'analisi dei dati informatici che è possibile recuperare da un computer, al fine di utilizzare i risultati in un processo giuridico o in un'indagine[1].

Tramite la post-mortem forensic, nel dettaglio, l'operatore mira al recupero e all'analisi dei dati su una macchina spenta, ossia di tutti i dati leggibili dalle memorie di massa (tralasciando quindi i dati presenti in RAM e in transito sulla rete, interesse della live forensic, branca complementare alla post-mortem nella computer forensic)[1].

A supporto delle procedure di post-mortem forensic sono stati sviluppati negli ultimi anni molti strumenti software: obiettivo principale di questa relazione è fornire una panoramica esauriente sullo stato dell'arte degli strumenti open source a disposizione degli operatori che intendono eseguire post-mortem forensic, utilizzando un caso di studio appositamente creato.

Verranno descritti i vantaggi e gli svantaggi derivanti dall'utilizzo di sistemi operativi Linux e, più in generale, dei tool open source.

Fra i maggiori vantaggi è possibile indicare:

• Flessibilità: molti tool che verranno presentati consentono di eseguire la stessa operazione con procedure molto differenti fra loro;

• Costo dei tool: la quasi totalità dei tool viene distribuita gratuitamente;

• Codice sorgente accessibile: l'utente più esperto ha la possibilità di modificare e ampliare il codice dei tool, portando i sui studi e la sua esperienza a beneficio di tutta la comunità.

I vantaggi presentati tuttavia possono rivelarsi facilmente degli svantaggi, infatti:

• La flessibilità comporta differenti utilizzi dei tool da parte degli operatori, con conseguente mancanza di una standardizzazione delle procedure da eseguire. Tale carenza può, durante un processo, consentire alla controparte di contestare il lavoro svolto dall'operatore.

• Sebbene l'utilizzo di tool gratuiti consenta un notevole risparmio economico,

solitamente tali tool sono più complessi rispetto ai tool commerciali: gran parte di essi non fornisce un'interfaccia grafica e un errore di battitura sulla linea di comando può distruggere in modo irreversibile l'evidenza digitale. Nel caso in cui sia necessario ricevere supporto tecnico dagli sviluppatori, inoltre, il costo di tali consulenze è in genere superiore alla norma. Infine occorre precisare che se durante un processo risulti necessaria la presenza di uno sviluppatore del tool utilizzato a supporto della tesi presentata al termine dell'indagine, ottenere tale assistenza può risultare molto difficile.

• Codice sorgente accessibile: avendo accesso al codice sorgente di un tool la controparte ha la possibilità di eseguire un'analisi approfondita dello stesso e, scoprendo eventuali bug o lacune, può contestare il lavoro presentato e far cadere la tesi presentata.

Viene riportata di seguito l'organizzazione generale di questa tesina:

Nel **capitolo 1** verranno presentate, nell'ambito degli strumenti open source, le distribuzioni linux più utilizzate e rinomate per compiere attività di computer forensic.

Nel **capitolo 2** saranno presentati i tool più importanti disponibili nelle distribuzioni linux.

Nel **capitolo 3** verrà descritto il caso di studio utilizzato per eseguire l'analisi dei tool di acquisizione e dei tool di analisi Autopsy e PTK. Inoltre verranno illustrati i risultati ottenuti nelle misurazioni delle acquisizioni e le conclusioni alle quali si è giunti.

Nel **capitolo 4** verrà illustrata la teoria alla base del funzionamento del caching del browser web Internet Explorer; grazie a tali fondamenta teoriche, nel **capitolo 5** sarà possibile comprendere a pieno il funzionamento dei tool in grado di automatizzare il recupero delle pagine web e la loro ricostruzione.

Infine, nel **capitolo 6**, verranno riassunti brevemente gli obiettivi raggiunti al termine degli studi svolti e verranno elencate le risorse consultate.

1. Distribuzioni linux per l'analisi forense

Le distribuzioni linux “forensic oriented” più utilizzate sono quattro:

- Caine (Computer Aided Investigative Environment)
- Deft (Digital Evidence Forensics Toolkit)
- Backtrack
- Helix

Fatta eccezione per la Backtrack che viene fornita come immagine DVD bootable, tutte vengono distribuite come immagine CD bootable.

Le distribuzioni citate sono tutte basate su Ubuntu e, allo stato attuale, su kernel 2.6.3x.

Le motivazioni che hanno spinto gli sviluppatori a basare le loro distribuzioni su Ubuntu sono molteplici:

- è ritenuta universalmente come la distribuzione linux più user-friendly disponibile;
- è supportata dalla comunità di sviluppatori più grande al mondo; ciò consente di avere rilasci e correzioni di bug più rapidamente rispetto ad altre distribuzioni;
- è facilmente personalizzabile grazie al suo gestore di pacchetti.

E' opportuno precisare che la distribuzione Backtrack non è stata sviluppata specificatamente per l'analisi forense: essa infatti si colloca nel contesto più ampio dell'information gathering e penetration testing che, spesso, comprende attività che hanno come unico obiettivo il trovare falle di sicurezza nei sistemi e nelle reti; per questo motivo la Backtrack è considerata dai più come la distribuzione linux d'eccellenza dei c.d. “black hat”. Nonostante ciò essa risulta essere dotata di molti dei tool recensiti in questo documento e, come verrà mostrato successivamente, risulta essere fra le più veloci in fase di acquisizione delle evidenze.

Verranno presentati ora nei dettagli i tool di sviluppo più importanti forniti in dotazione con le distribuzioni linux citate.

Essi possono essere suddivisi in quattro macro-categorie:

• Tool di acquisizione: sono tutti quei tool che consentono di acquisire le evidenze digitali da una macchina.

• Tool di analisi: fanno parte di questa categoria tutti quei tool che consentono di analizzare le evidenze ottenute in fase di acquisizione; a loro volta si suddividono in tool di analisi di basso livello e tool di analisi di alto livello.

Fra i tool di analisi di alto livello è possibile individuare, inoltre, le seguenti categorie:

• Tool di cracking: vi appartengono quei tool che consentono di recuperare, a partire da un'evidenza cifrata, l'evidenza originale e/o la password utilizzata per la cifratura.

• Tool integrati: l'indagine condotta dall'operatore può risultare, anche nei casi più semplici, molto lunga e composta da innumerevoli dati da tracciare e riportare nelle relazioni finali; i tool appartenenti a questa categoria supportano l'operatore nel riportare in modo dettagliato le informazioni rilevanti trovate.

Nel capitolo successivo vengono descritti, per ogni categoria, i tool più importanti.

2. I tool

In questo capitolo vengono presentati i tool disponibili nelle distribuzioni Linux per eseguire analisi forense post-mortem.

2.1 I tool di acquisizione

I tool di acquisizione più importanti disponibili nelle distribuzioni linux sono principalmente a linea di comando e sono i seguenti:

• dd

dd è il comando di acquisizione e trasferimento dati per eccellenza sui sitemi Unix. Esso è presente sin dalle prime versioni del sistema operativo ed eredita la sua sintassi da quella utilizzata nei mainframe IBM-360:

dd [if=file] [of=file] [ibs=bytes] [obs=bytes] [bs=bytes]

[cbs=bytes] [skip=blocks] [seek=blocks] [count=blocks]

[conv={ascii,ebcdic,ibm,block, unblock,lcase,ucase,swab,noerror,notrunc,sync}]

Le opzioni elencate assumono i seguenti significati:

if – input file: indica il file (o la device) dal quale deve essere letto l'input. Se assente, legge dallo standard input;

of – output file: indica il file (o la device) sul quale deve essere scritto l'output. Se assente, scrive sullo standard output;

ibs – indica il numero di bytes che devono essere letti per ogni operazione di lettura;

obs – indica il numero di bytes che devono essere scritti per ogni operazione di scrittura;

bs – se definito utilizza il suo valore come parametro di ibs e obs;

cbs – setta i buffer di conversione da ASCII a EBCDIC o da un device non a blocchi ad uno a blocchi (necessaria solo in rarissimi casi);

skip – consente di eseguire la lettura dell'input tralasciando, partendo dall'inizio, i

blocchi specificati;

seek – tralascia gli ultimi nbl blocchi del file. Tale opzione ha senso solo se associata all'opzione notrunc;

count – effettua la copia del numero di blocchi specificati dal valore di questa opzione. Se assente copia tutto il file;

conv – è un'opzione che consente di effettuare diverse conversioni durante l'esecuzione. I suoi valori sono i seguenti:

- ascii/ebcdic/ibm: esegue una conversione dal formato originale verso il formato specificato;
- block: allunga tutti i record terminati da un newline alla lunghezza di cbs, sostituendo al newline degli spazi;
- unblock: toglie gli spazi bianchi e aggiunge un newline;
- lcase/ucase: converte il testo da maiuscolo/minuscolo a minuscolo/maiuscolo;
- swab: effettua, a due a due, lo swap di due byte di input;
- noerror: prosegue l'operazione anche in caso di errori;
- sync: allunga i blocchi di input alla lunghezza indicata da ibs, aggiungendo dei NULL;

• dcfldd

dcfldd è, insieme a dc3dd, una versione di dd ampliata con nuove funzionalità. Essa deve il suo nome al Defense Computer Forensics Lab, una sezione del Defense Cyber Crime Center americano che è un dipartimento analogo alla Polizia Postale italiana. Fra le funzionalità aggiuntive di questo tool vi sono:

• Hashing on the fly;

• Progress bar per sapere a che punto è il trasferimento;

• Set di pattern per eseguire il wiping;

• Verifica bit a bit dell'originale con la copia eseguita;

• Output multipli su supporti differenti (Read once, write many);

• Output partizionabile in file multipli;

• Possibilità di redirezionare l'output in pipeline verso altre applicazioni.

•dc3dd

Anche dc3dd, come dcfldd, è una versione ampliata di dd. A differenza di dcfldd però, questo tool non è un fork del dd originale ma si presenta come patch; ciò significa che gli aggiornamenti al dd originale sono immediatamente disponibili in dc3dd, mentre dcfldd ha uno scheduling delle release indipendente.

Oltre a questa differenza, i tool sono pressoché identici da un punto di vista funzionale. Fra le poche differenze presenti occorre segnalare che dcfldd è supportato sulla piattaforma cygwin, dispone di un numero maggiore di algoritmi di hashing e ha la possibilità di scrivere pattern random in output, funzionalità non disponibili in dc3dd.

• guymager

a differenza dei tool descritti in precedenza, Guymager è un tool di acquisizione dotato di interfaccia grafica. I punti di forza di questo prodotto sono la compatibilità e la velocità. Oltre al formato raw, con il formato EWF (Expert Witness Format) utilizzato, ad esempio, nei tool EnCase ed FTK. Tale formato risulta essere uno standard 'de-facto' per l'acquisizione delle evidenze digitali in quanto largamente utilizzato dalle forze di polizia. Inoltre Guymager è dotato di un motore multi thread che consente di sfruttare al meglio i processori multicore e dotati di tecnologia hyperthreading, con conseguente aumento di velocità rispetto a dd.

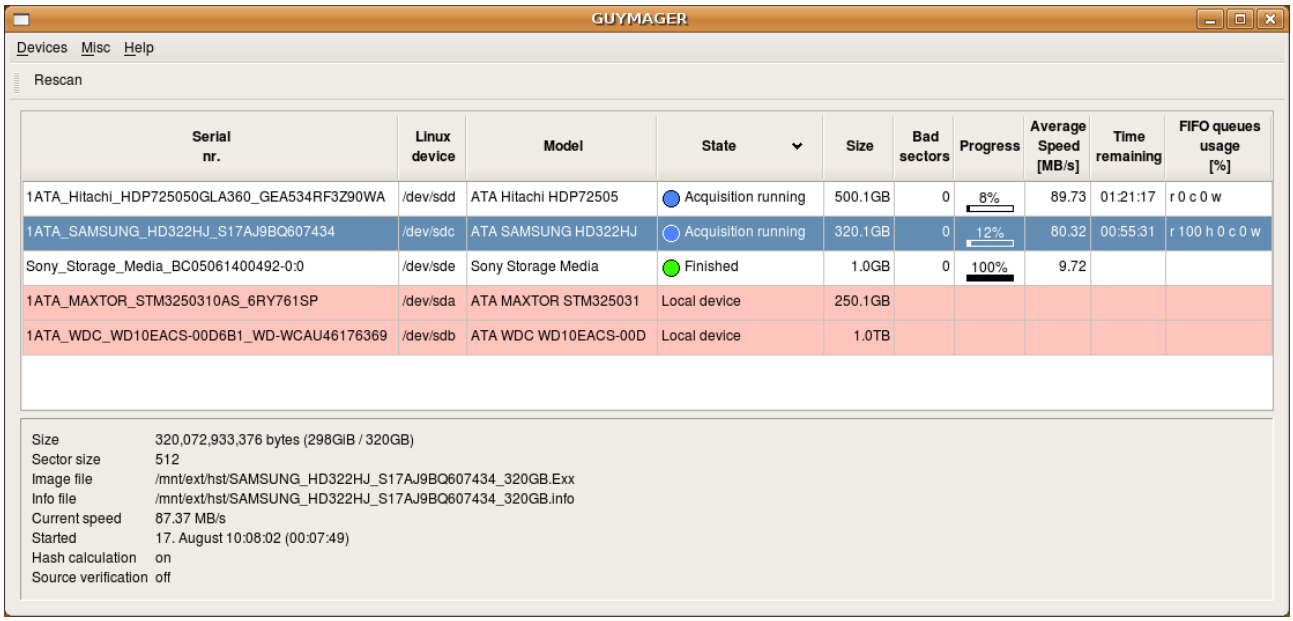


Figura 2.1 - Schermata principale di Guymager

• **FTK Imager**

FTK Imager è un prodotto commerciale sviluppato dalla AccessData, utilizzabile sia in ambiente Microsoft Windows che in Linux (tramite il motore Wine). E' considerato uno dei migliori software di acquisizione dati in quanto supporta molteplici formati (fra cui il formato SMART della AsrData) e consente inoltre, grazie ad una tecnologia chiamata Isobuster, di acquisire rapidamente anche supporti rimovibili e salvarli come file BIN/CUE.

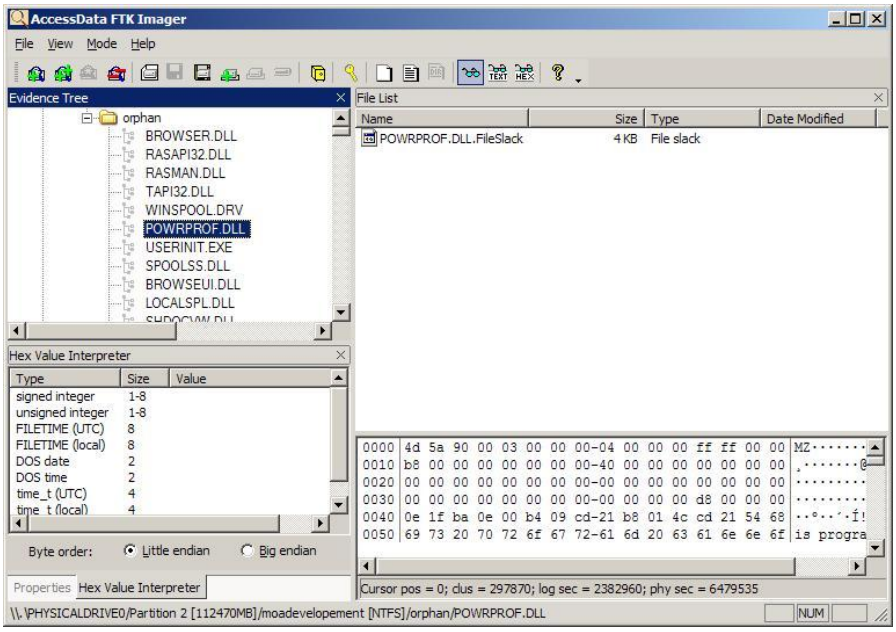


Figura 2.2 - Interfaccia grafica di FTK Imager

• **Scalpel**

Più che essere un tool di acquisizione, Scalpel è un tool gratuito ed open source che consente di recuperare i files cancellati dall'immagine acquisita di un'evidenza digitale (operazione di carving). Scalpel è un tool file-system independent che consente il recupero sia di file interi o frammenti di essi in base alle informazioni lette nelle definizioni degli header e dei footers.

Nonostante sia un tool cross-platform, gli stessi sviluppatori considerano Linux come la miglior piattaforma su cui eseguire Scalpel.

Riguardo all'acquisizione e al recupero di files cancellati occorre dare alcuni dettagli.

Spesso tali file possono essere recuperati soltanto in parte perché possono essere soggetti a sovrascritture o a danni presenti nella struttura del filesystem.

Tuttavia tool di carving (tool che consentono il recupero di file cancellati) differenti utilizzano algoritmi per l'analisi dello spazio libero diversi fra loro, con conseguente differenza nei risultati di output. L'operatore che intende recuperare quante più informazioni possibili dallo spazio libero deve utilizzare più tool di carving sulla stessa evidenza e, al termine, confrontare i risultati per avere una visione accurata dell'evidenza digitale.

Altro punto da precisare riguarda la presenza di file crittografati sull'evidenza: sebbene esistano molti tool di password cracking che consentono di recuperare le chiavi utilizzate per la cifratura, spesso essi risultano inefficaci a causa della complessità della password o dell'algoritmo di cifratura utilizzato dall'utente.

Nonostante ciò, la sola segnalazione della presenza di file crittografati sull'evidenza potrebbe essere d'aiuto al committente dell'indagine e, quindi, va riportata in modo dettagliato.

2.2 I tool di analisi

I tool di analisi osservati, si distinguono sommariamente in tre categorie: quelli che permettono di fare una analisi di basso livello, ossia analizzare direttamente il volume (il file system); quelli che permettono di effettuare una analisi più di alto livello, ossia a livello delle applicazioni; e quelli che ci permettono di visualizzare e analizzare una timeline, ossia una linea del tempo che riporta tutte le operazioni effettuate sui file.

2.2.1 I tool di analisi di basso livello

Tra la molteplicità dei tool di analisi di basso livello disponibili, si evidenziano:

- **Catfish:** è un tool che permette di ricercare file a partire da parole chiavi.

Una guida breve e chiara può essere consultata all'indirizzo:
http://www.ubuntugeek.com/catfish-file-search-tool-that-support-several-different-engines.html

- **Large Text File Viewer:** utile per aprire e visualizzare istantaneamente file di testo

molto grandi, di taglia > 1 GB.

È possibile scaricarlo, visualizzare diversi screenshots, e consultare la guida all'indirizzo http://www.swiftgear.com/ltfviewer/features.html

- **GHex:** è un semplice editor binario, permette all'utente di visualizzare e scrivere un file binario sia in ascii che in esadecimale.

È possibile avere altre informazioni e scaricarlo all'indirizzo http://www.icewalkers.com/Linux/Software/535950/GHex.html

- **Digital Framework Forensic(DFF):** è sia un tool di investigazione sia una piattaforma di sviluppo. Consiste di un insieme di moduli, librerie, tool, interfacce utente.

È possibile scaricarlo e avere maggiori informazioni all'indirizzo http://www.digital-forensic.org/

- **WinAudit:** tool che permette di fare un inventario delle caratteristiche hardware e software di un computer.

È possibile scaricarlo e avere maggiori informazioni all'indirizzo http://it.kioskea.net/download/scaricare-21-winaudit

- **RegScanner:** è una piccola utility che permette di eseguire la scansione del registro di sistema, trovando i valori desiderati che corrispondono a criteri di ricerca specificati.

È possibile scaricarlo e avere maggiori informazioni all'indirizzo http://it.kioskea.net/download/scaricare-881-regscanner

- **PhotoRec:** è un programma di recupero di dati destinato a recuperare foto perse dalla scheda di memoria di una fotocamera. È possibile recuperare tutti i tipi di file persi come file video, documenti o file memorizzati su un HDD o CD-ROM.

È possibile scaricarlo e avere maggiori informazioni all'indirizzo

http://it.kioskea.net/download/scaricare-80-photorec

- **WhatInStartup:** visualizza l'elenco di tutte le applicazioni che vengono lanciate automaticamente all'avvio di Windows. Per ogni applicazione vengono riportate una serie di informazioni.

È possibile scaricarlo e visualizzare più informazioni all'indirizzo:

http://it.kioskea.net/download/scaricare-874-whatinstartup

2.2.2 I tool di analisi di alto livello

Per quanto riguarda i tool di analisi di alto livello, sono stati analizzati analizzatori di cache del browser, dei cookies, della cronologia, della posta elettronica, dei log di chat. Verrà fatta ora una panoramica di tali tool, con l'obiettivo di fornire al lettore un quadro generale delle possibilità che le distribuzioni Linux offrono per eseguire analisi forense di tipo post-mortem. Si tiene a precisare che questi tool vengono utilizzati in ambiente Linux attraverso l'ambiente Wine (http://www.winehq.org).

• *Cache Analysis*

- **ChromeCacheView:** piccola utility che legge la cache di Google Chrome e visualizza la lista di tutti i file di essa.

Download e maggiori informazioni all' indirizzo
http://www.nirsoft.net/utils/chrome_cache_view.html

- **IECacheView:** piccola utility simile a quella precedente, analizza la cache di Internet Explorer.

Download e maggiori informazioni all'indirizzo
http://www.nirsoft.net/utils/ie_cache_viewer.html

- **Mozilla Cache View:** piccola utility simile alle precedenti, analizza la cache di Firefox.

Download e maggiori informazioni all'indirizzo
http://www.nirsoft.net/utils/mozilla_cache_viewer.html

- **MUICacheView:** ogni volta che si usa una nuova applicazione, Windows estrae automaticamente l'application name del file exe, e lo memorizza in una chiave di registro nota come MUICache, per utilizzarlo successivamente. L' utility permette di visualizzare e modificare la lista di tutti gli elementi nella MUICache.

Download e maggiori informazioni all' indirizzo
http://www.nirsoft.net/utils/muicache_view.html

- **OperaCacheView:** piccola utility che permette di analizzare il contenuto della cache del browser web Opera.

Download e maggiori informazioni all' indirizzo
http://www.nirsoft.net/utils/opera_cache_view.html

- **VideoCacheView:** utility che permette di estrarre dalla cache i file video che sono stati salvati dopo averli visti da browser web, in modo da poterli rivedere in futuro. Effettua automaticamente la scansione della cache di Internet Explorer e Mozilla Firefox estraendo tutti i video che sono memorizzati nella cache.

Download e maggiori informazioni all'indirizzo
http://www.nirsoft.net/utils/video_cache_view.html

• *Cookies Analysis*

- **IECookiesView:** piccola utility che visualizza i dettagli di tutti i cookies memorizzati da Internet Explorer.

Download e maggiori informazioni all' indirizzo
http://www.nirsoft.net/utils/iecookies.html

- **MozillaCookiesView:** visualizza i dettagli di tutti i cookies memorizzati nel file dei cookies (cookies.txt). Permette inoltre di cancellare cookies indesiderati e il backup / ripristino del file dei cookies.

Download e maggiori informazioni all'indirizzo
http://www.nirsoft.net/utils/mzcv.html

• *Browser history analysis*

- **IEHistoryView:** utility che legge tutte le informazioni della cronologia e visualizza la lista di tutte le URL visitate negli ultimi giorni.

Download e maggiori informazione all' indirizzo
http://www.nirsoft.net/utils/iehv.html

- **MozillaHistoryView:** utility simile alla precedente che opera su Firefox.

Download e maggiori informazioni all' indirizzo
http://www.nirsoft.net/utils/mozilla_history_view.html

• *Analizzatori di email e chat*

- **BulkExtractor:** utility che ci permette di estrarre archivi protetti da password.

Download e maggiori informazioni all' indirizzo
http://bstdownload.com/reviews/bulk-extractor/

- **LibPST:** utility che contiene delle librerie che permettono di accedere alle cartelle di Outlook.

Download e maggiori informazioni all'indirizzo
http://freshmeat.net/projects/libpst/

- **Live Contacts View:** tool che permette di aprire il database dei propri contatti Messenger senza accedere al programma di messaggistica di Microsoft, che di default registra i contatti nel file contacts.edb, bloccato quando il programma è aperto.

Download e maggiori informazioni all'indirizzo

http://www.downloadblog.it/post/10428/vedere-i-contatti-messenger-con-live-contacts-view

- **SkypeLogView:** tool che permette di leggere i file di log creati da Skype e visualizzare i dettagli sulle chiamate in entrata / uscita, messaggi di chat, e trasferimenti di file.

Download e maggiori informazioni all'indirizzo
http://www.softpedia.com/get/Internet/Chat/Other-Chat-Tool/SkypeLogView.shtml

- **SkypeHistoryViewer:** tool che permette di visualizzare l' intera cronologia delle conversazioni avvenute tramite Skype.

Download e maggiori informazioni all' indirizzo
http://www.fratellogeek.com/come-vedere-le-conversazioni-su-skype-con-history-viewer/

• *Analisi delle vulnerabilità*

- **ClamAv:** è un antivirus open source che permette di rilevare virus, trojan, malware, ed altri oggetti maliziosi.

Download e maggiori informazioni all'indirizzo
http://www.clamav.net/lang/en/

- **Rootkit revealer:** tool che effettua la scansione di file e registro e informa l'utente su cosa è stato trovato di sospetto. Per non essere identificato dai rootkit esegue le proprie scansioni da una copia di se stesso rinominata in modo casuale, che viene eseguita come servizio di Windows.

Download e maggiori informazioni all' indirizzo
http://www.notrace.it/Download/Sicurezza/Anti-Rootkit/rootkit-revealer.htm

- **Sophos anti-rootkit:** software free che rileva e rimuove ogni rootkit nascosto utilizzando una tecnologia avanzata.

Download e maggiori informazioni all' indirizzo
http://www.sophos.com/en-us/products/free-tool/sophos-anti-rootkit.aspx

- **Rootkit hunter:** software che effettua una scansione dei rootkit eventualmente presenti nel sistema.

Download e maggiori informazioni all' indirizzo
http://www.geekissimo.com/2009/08/30/rkhunter-facciamo-diventare-linux-ancora-piu-sicuro/

• *Password recovery /cracking*

- **IE PassView:** piccola utility di gestione delle password che rivela le password memorizzate dal browser web Internet Explorer, e consente di eliminare le password che non servono più. Supporta tutte le versioni di Internet Explorer, dalla 4.0 alla 8.0.

Download e maggiori informazioni all' indirizzo
http://it.kioskea.net/download/scaricare-788-ie-passview

- **Opera PassView:** piccola utility che permette il recupero delle password memorizzate dal browser web Opera (nel file wand.dat).

Download e maggiori informazioni all' indirizzo
http://www.nirsoft.net/utils/opera_password_recovery.html

- **ChromePass:** piccola utility che permette di effettuare il recupero delle password memorizzate dal browser web Google Chrome.

Download e maggiori informazioni all' indirizzo
http://www.nirsoft.net/utils/chromepass.html

- **Mail PassView:** piccola utility che permette di recuperare password e altre informazioni di alcuni client di posta elettronica come Outlook Express.

Download e maggiori informazioni all' indirizzo
http://www.nirsoft.net/utils/mailpv.html

- **MessenPass:** tool che permette il recupero delle password memorizzate da diverse applicazioni di messaggeria istantanea.

Download e maggiori informazioni all' indirizzo
http://www.nirsoft.net/utils/mspass.html

- **VNCPassView:** tool che permette di recuperare le password memorizzate dal tool VNC. Recupera 2 tipi di password: quelle memorizzate dall'utente correntemente loggato e quelle memorizzate per tutti gli utenti.

Download e maggiori informazioni all' indirizzo
http://www.nirsoft.net/utils/vnc_password.html

- **OphCrack:** applicazione Open Source che permette di recuperare password smarrite. Esso cracka le password di sistemi operativi Windows avvalendosi degli hash LM ed NTLM attraverso le Rainbow Tables.

Download e maggiori informazioni all' indirizzo
http://ophcrack.sourceforge.net/

- **John the ripper:** ottimo strumento per testare la robustezza delle password di un sistema. Il suo utilizzo non è complesso e può aiutare ad individuare, fra le proprie password, quelle troppo semplici.

Download e maggiori informazioni all' indirizzo
http://openskill.info/infobox.php?ID=642

- **PDFCrack:** tool online che permette di sbloccare i file pdf protetti da password.

Download e maggiori informazioni all' indirizzo
http://informaticafree.ilbello.com/?p=1727

- **fcrackzip:** tool che permette di recuperare password di file zippati protetti, a forza bruta o attraverso un attacco basato su dizionario.

Download e maggiori informazioni all' indirizzo
http://www.ubuntugeek.com/fcrackzip-password-cracker-for-zip-archives.html

- **Dialupass:** utility che permette di recuperare nome utente, password e dominio di tutte le voci dialup/VPN. Può essere utilizzate per recuperare la password smarrita di una connessione Internet o VPN.

Download e maggiori informazioni all' indirizzo
http://www.nirsoft.net/utils/dialupass.html

- **PstPassword:** piccola utility che permette il recupero di password smarrite di Outlook.

Download e maggiori informazioni all' indirizzo
http://www.nirsoft.net/utils/pst_password.html

- **Hydra:** permette di effettuare degli attacchi a forza bruta ad un servizio di autenticazione remota.

Download e maggiori informazioni all' indirizzo
http://www.darknet.org.uk/2007/02/thc-hydra-the-fast-and-flexible-network-login-hacking-tool/

2.2.3 I tool integrati

Per "tool integrati" si intendono tutti quei tool che consentono, attraverso un'interfaccia grafica user friendly, di accedere in modo semplice e rapido a più tool di basso livello e che, inoltre, permettono di stilare dei report del lavoro svolto dall'utente in modo da supportarlo nella stesura delle relazioni finali valide come prove scientifiche.

I tool elencati di seguito, Autopsy e PTK, sono entrambe presenti nella distribuzione linux Backtrack; essi verranno descriti in modo dettagliato nei capitoli successivi.

- **Autopsy Forensic Browser:** oltre alle consuete attività di investigazione, ci aiuta in fase di report grazie alle sue funzionalità di case management.

In figura 2.3 mostriamo la timeline che viene visualizzata da Autopsy:

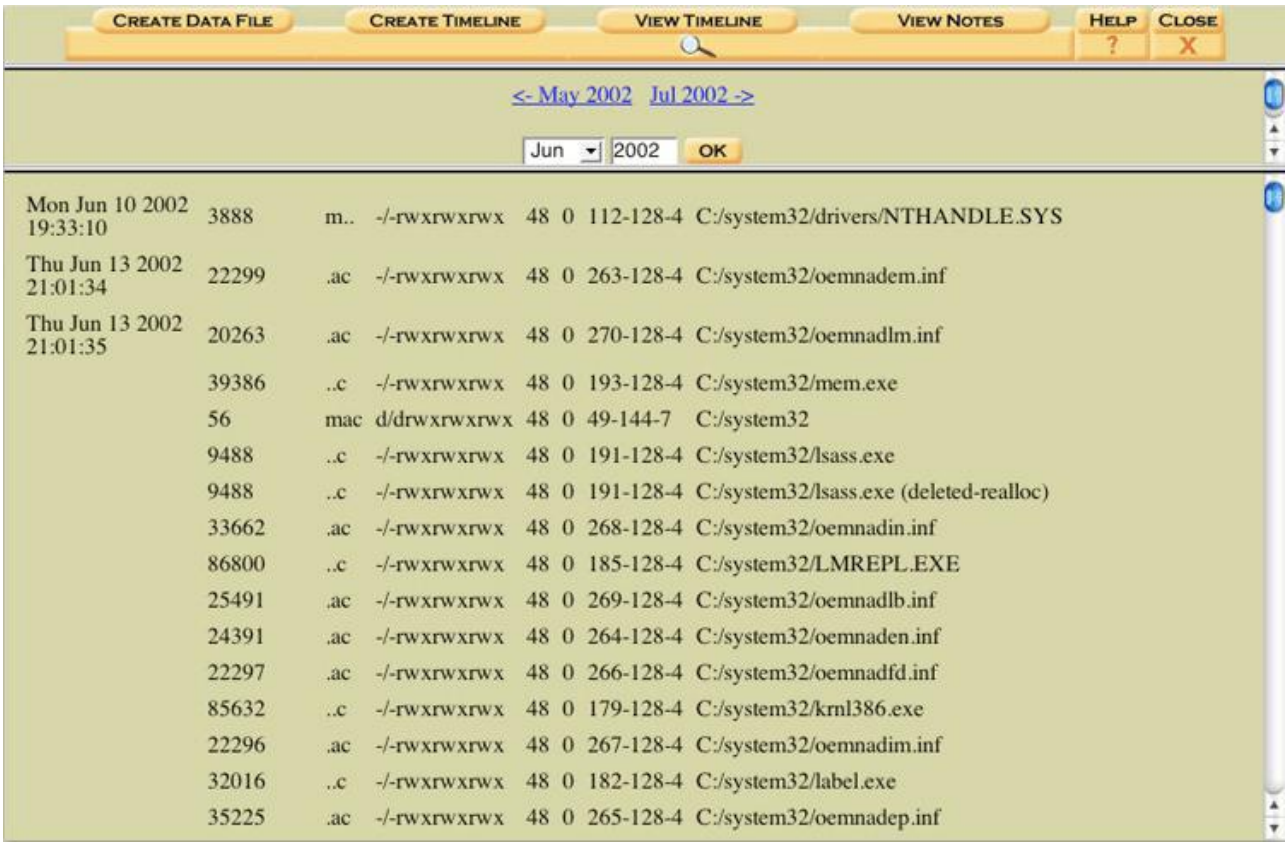


Figura 2.3

Download e maggiori informazioni all' indirizzo
http://www.sleuthkit.org/autopsy/download.php

- PTK Framework: è come Autopsy ma

- è dotato di una GUI più veloce tramite Ajax
- è modulare, permette l'aggiunta di nuovi tool
- è dotato di un potente motore di indicizzazione
- permette di aggiungere bookmark da poter condividere con gli altri investigatori

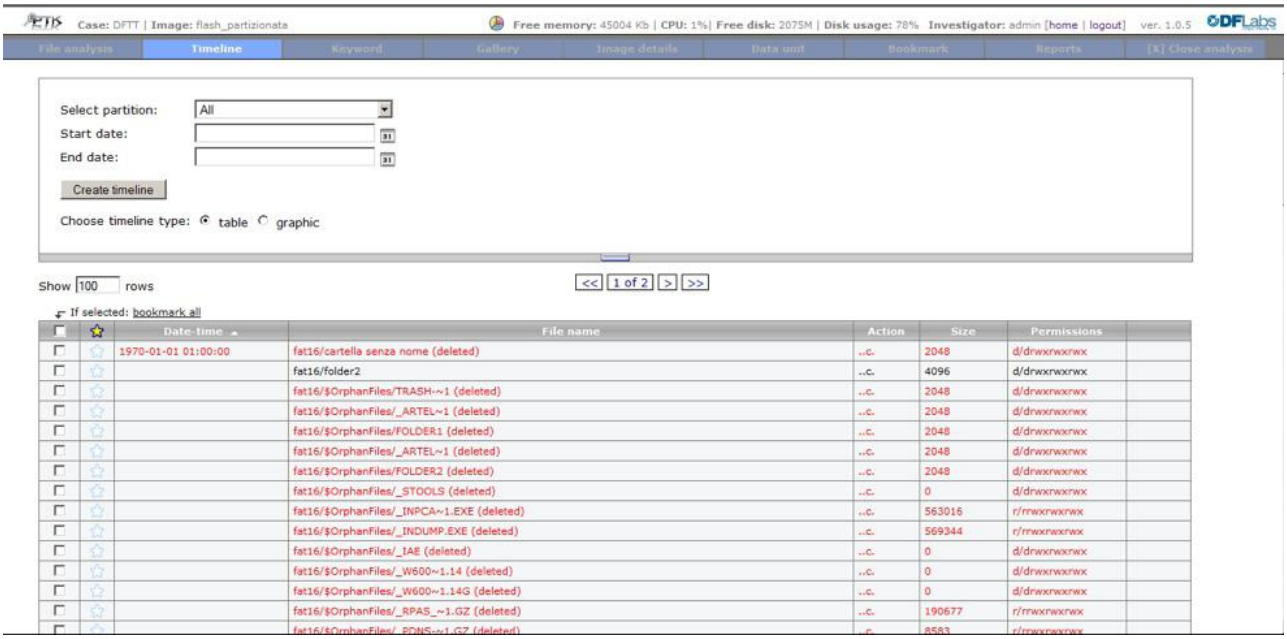


Figura 2.4 - la timeline che viene visualizzata da PTK

Download e maggiori informazioni all'indirizzo

http://ptk.dflabs.com/

Dopo aver presentato i tool open source più utilizzati in ambito forense si è proceduto alla creazione di un caso di studio e ad un utilizzo effettivo degli strumenti descritti allo scopo di valutarne le prestazioni e le funzionalità in modo più dettagliato.

3. Il caso di studio

Verrà simulata l'analisi post-mortem di un computer utilizzato con sistema operativo Microsoft Windows 7 con utente singolo. Gli obiettivi dell'analisi del caso di studio sono quelli di fornire una panoramica dei tool di acquisizione e di analisi disponibili, così come fornire le indicazioni da seguire nell'utilizzo degli stessi. Inoltre, il caso di studio creato verrà utilizzato per simulare la ricostruzione delle pagine web visualizzate dall'utente nel passato, a partire dai dati temporanei mantenuti dal browser web sul computer.

Per la creazione del caso di studio si è simulato il normale utilizzo di un computer da parte di un utente, ossia: navigazione web; visualizzazione e download di immagini; navigazione su social network; chat e telefonata in VoIP tramite Skype.

L'installazione del sistema operativo è stata effettuata su una partizione di 19.4GB di un hard disk collegato tramite interfaccia parallela FireWire 800, utilizzando una macchina virtuale.

L'utilizzo di una macchina virtuale non ha comportato particolari problemi infatti, sebbene l'ambiente sia stato virtualizzato, le scritture su disco sono state rese persistenti in quanto il virtualizzatore è stato configurato in modo da utilizzare la partizione dedicata alla simulazione in modalità raw e non virtualizzata su file.

Inoltre l'utilizzo di una macchina virtuale ha consentito di ibernare l'hardware in determinate condizioni e riprendere l'analisi in momenti distanti fra loro, velocizzando il lavoro.

L'unico svantaggio di tale approccio è il decadimento di prestazioni dovuto alla virtualizzazione dell'hardware; in seguito verranno mostrate nel dettaglio le prestazioni di lettura e scrittura da macchina virtuale ed i risultati saranno confrontati con le prestazioni ottenute eseguendo in nativo i tool.

Sul sistema da analizzare sono state effettuate le seguenti operazioni:

• Navigazione web con Internet Explorer

• Navigazione web sul social network Facebook

• Configurazione e accesso a Live Messenger

• Download di due fotografie dal sito web deviantart.com, una delle quali è stata nascosta e modificata nel nome da .jpg a .txt

• Chat e chiamata tramite Skype

Il dettaglio delle operazioni è stato riportato in uno storico, del quale viene mostrato un estratto:

- Ore 11.34 - avviata la macchina virtuale
- Ore 11.35 - avviato internet explorer: connessione alla homepage http://it.msn.com/?ocid=iehp
- Ore 11.36 - ricerca su bing: facebook
- Ore 11.36 - http://www.facebook.com
- Ore 11.37 - login su facebook (memorizza password, ricorda login)
- Ore 11.38 - navigazione profilo di Umberto Annunziata
- Ore 11.40 - inizio conversazione fb con Umberto Annunziata
- Ore 11.43 - fine conversazione
- Ore 11.43 - visualizzazione gruppo Quelli di SOII
- Ore 11.44 - rimossa discussione da fb con Umberto Annunziata
- Ore 11.45 - lasciato post su fb nel gruppo Quelli di SOII
- Ore 11.46 - ricevuto un "mi piace" da Giovanni Mastroianni e Umberto Annunziata
- Ore 11.46 - ricevuto commento in risposta al post da Umberto Annunziata
- Ore 11.47 - lettura delle ultime due notifiche
- Ore 11.47 - passaggio dalla pagina del gruppo alla home di facebook
- Ore 11.48 - logout da facebook e chiusura di Internet Explorer
- Ore 11.49 - apertura di Live Messenger, configurazione account e accesso
- Ore 11.50 - eseguito login su messenger, accesso area MSN
- Ore 11.50 - caricata area MSN con notizia in evidenza "Corona punta tutto su Belen"
- Ore 11.51 - inizio conversazione con Umberto Annunziata
- Ore 11.56 - invio messaggio con allegato tramite web a umberto287@libero.it
- Ore 12.01 - apertura internet explorer
- Ore 12.01 - ricerca su bing "deviantart"
- Ore 12.02 - navigazione sul sito web http://www.deviantart.com
- Ore 12.03 - navigazione sulla foto www.deviantart.com/#/d3hk9e6
- Ore 12.03 - download dell'immagine JPG sul desktop: foto.JPG
- Ore 12.04 - ritorno nella home di deviantart cliccando sul logo
- Ore 12.05 - navigazione nella categoria "digital art"
- Ore 12.05 - ritorno sulla home di deviantart
- Ore 12.06 - navigazione sulla foto www.deviantart.com/#/d3hk8i3
- Ore 12.07 - salvataggio su disco della foto sul desktop: testo.jpg
- Ore 12.07 - chiusura di internet explorer
- Ore 12.09 - rinominato il file testo.jpg in testo.txt
- Ore 12.10 - impostato il file testo.txt come nascosto (attraverso l’opzione apposita di Windows 7)
- Ore 12.11 - disconnessione e uscita da live messenger
- Ore 12.11 - apertura di skype
- Ore 12.12 - chiusura inaspettata di skype
- Ore 12.13 - arresto di windows
- Ore 15.38 - accensione macchina virtuale

- Ore 15.41 - login su skype
- Ore 15.48 - telefonata su skype con umberto
- Ore 15.52 – disinstallazione di skype
- Ore 16.00 - installazione di skype
- Ore 16.01 - navigazione su www.yahoo.it, poi su youtube.com
- Ore 16.03 - installazione del flash player
- Ore 16.03 - visualizzazione video su youtube v=4bujaPd4wuc
- Ore 16.06 - fine visualizzazione video
- Ore 16.08 - chiusura di Internet Explorer
- Ore 16.08 - arresto del sistema

3.1 Acquisizione della partizione

L'acquisizione della partizione da analizzare è stata effettuata tramite dc3dd e, nelle distribuzioni dove non era disponibile, è stato utilizzato dcfldd.

Per valutare le prestazioni delle distribuzioni linux in fase di acquisizione è stato effettuato un profiling del dd, variando la dimensione del BlockSize da 512 a 168384 Byte).

Le misurazioni sono state effettuate anteponendo il comando "time" al comando dd.

Essendo il fine ultimo di queste acquisizioni la misurazione delle prestazioni, si è scelto di utilizzare come interfaccia di output del comando dd il device /dev/zero. Così facendo si è riusciti a misurare i tempi di read della partizione senza alterazioni dovute alla scrittura dell'input su un disco di output.

Le distribuzioni analizzate sono:

• Ubuntu 11.04

• Backtrack 5

• Deft 6 (Digital Evidence Forensics Toolkit)

• Caine (Computer Aided Investigative Environment)

Le misurazioni, fatta eccezione per la Ubuntu 11.04, sono state effettuate da macchina virtuale; si ritiene lecito pensare che le differenze di prestazioni misurate fra le varie distribuzioni in virtuale siano evidenziabili anche in nativo.

L'hard disk su cui è stata creata la partizione di 19.4GB da analizzare è un hard disk

di 200GB Seagate Momentus II da 7200rpm, collegato alla macchina tramite interfaccia FireWire 800.

I risultati ottenuti in nativo tramite Ubuntu sono riportati in figura 3.1.

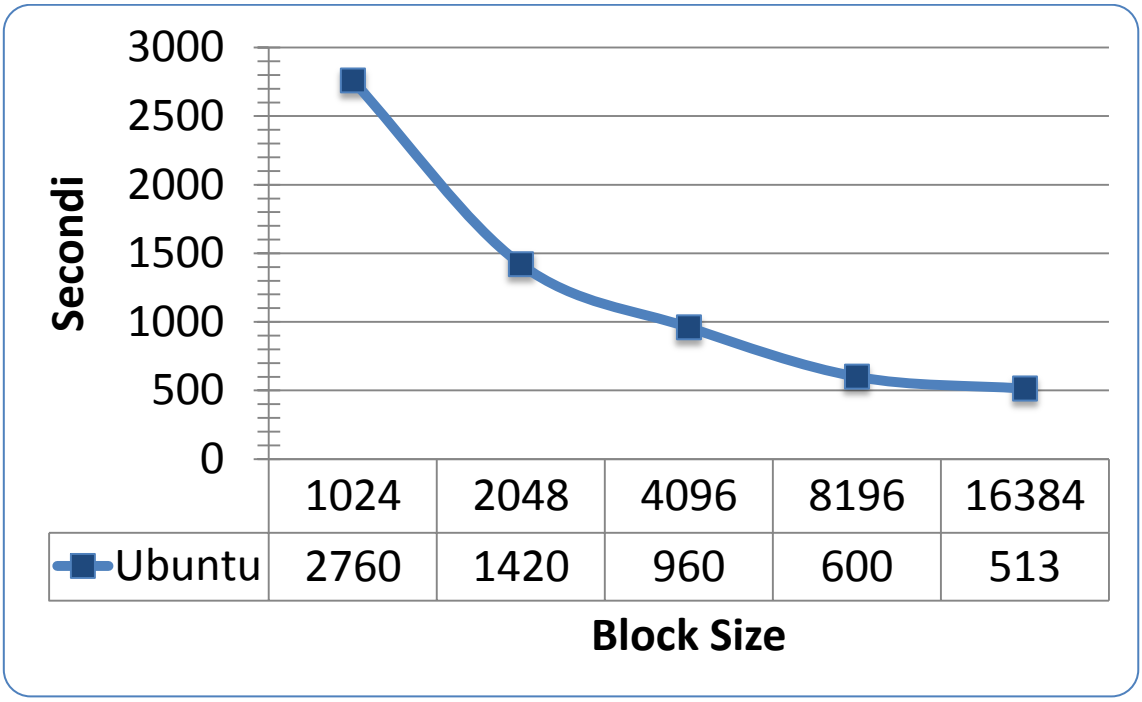


Figura 3.1 - dd con Ubuntu 11.04 (distribuzione nativa)

I test di acquisizione effettuati con le distribuzioni Linux sono riportati nelle figure 3.2 (in forma tabellare) e in figura 3.3 (in grafico).

BlockSize	Deft	Caine	Backtrack
1024	8016	7945,8	7357,63
2048	3876	3726	3973,12
4096	2754	2166	2207,29
8192	1514,4	1467	1460,1
16384	501,6	502,8	797,81

Figura 3.2 - Tabella dei risultati ottenuti con le varie distribuzioni

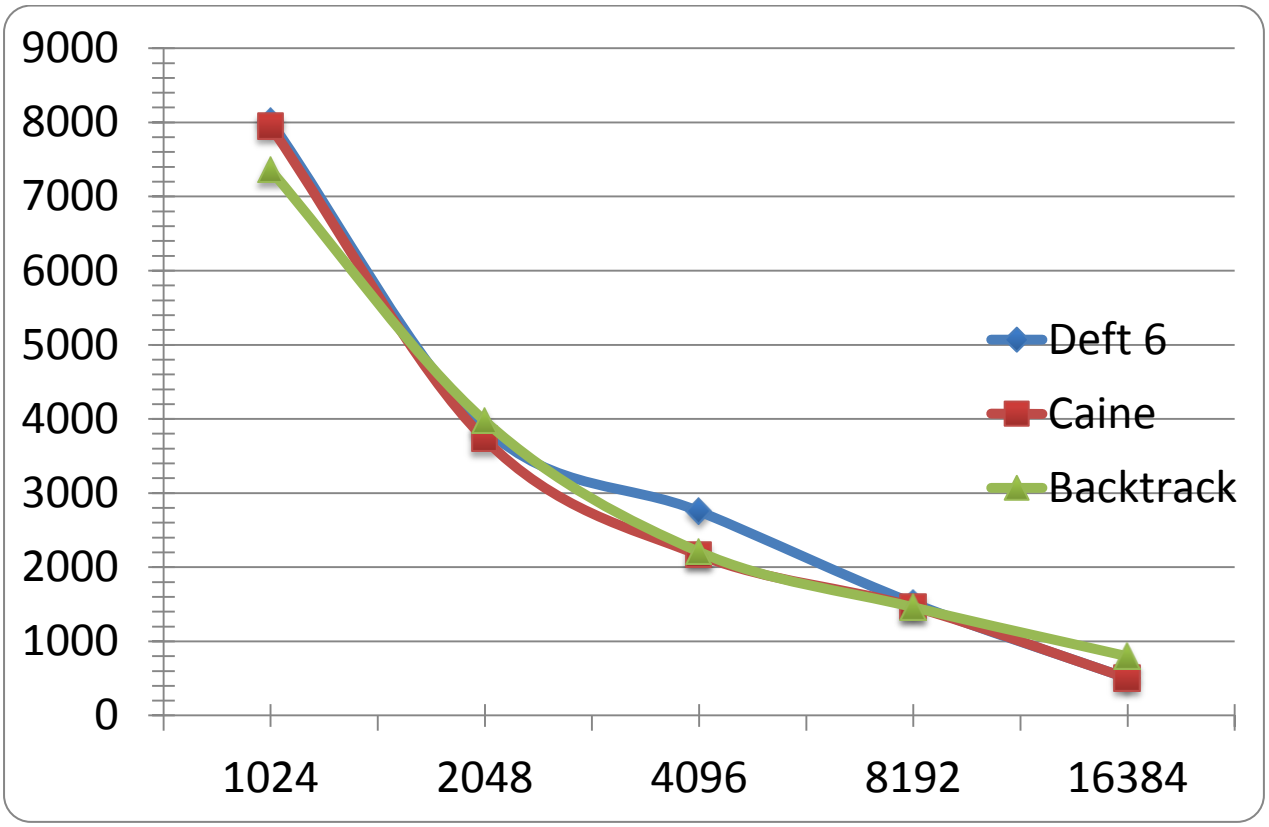


Figura 3.3 - dd con le distribuzioni Linux avviate tramite macchina virtuale

La distribuzione Linux Helix non è stata testata a causa del termine del periodo di prova di cui si era in possesso.

Le conclusioni alle quali si è giunti sono le seguenti:

• per effettuare copie forensi (con blocksize piccolo) la distribuzione linux più adatta risulta essere la Backtrack 5, mentre con blocksize più elevato conviene utilizzare la Deft;

• nessuna delle distribuzioni utilizzate riconosce la Firewire 800, infatti i test mostrano un top di 40 MB/s, anche in nativo, rispetto ai 100 MB/s disponibili.

• la scelta a favore di una distribuzione deve essere basata non solo sulle prestazioni di acquisizione del singolo hard disk, ma anche sul numero di dischi da acquisire e sulla loro dimensione; questo perché sul singolo hard disk il tempo guadagnato in fase di acquisizione potrebbe essere minore rispetto al tempo necessario per cambiare distribuzione per effettuare l'analisi. Se il numero di dischi da acquisire invece è elevato, allora utilizzare la distribuzione linux più efficiente può determinare un considerevole risparmio di tempo.

- Ulteriori analisi sull'acquisizione tramite distribuzioni linux

A sostegno delle supposizioni fatte confrontando i tempi di acquisizione fra le varie distribuzioni linux tramite macchina virtuale, sono stati rieseguiti i "dd-test" utilizzando nativamente i live cd delle distribuzioni "forensic oriented".

I risultati ottenuti sono riportati in figura 3.4:

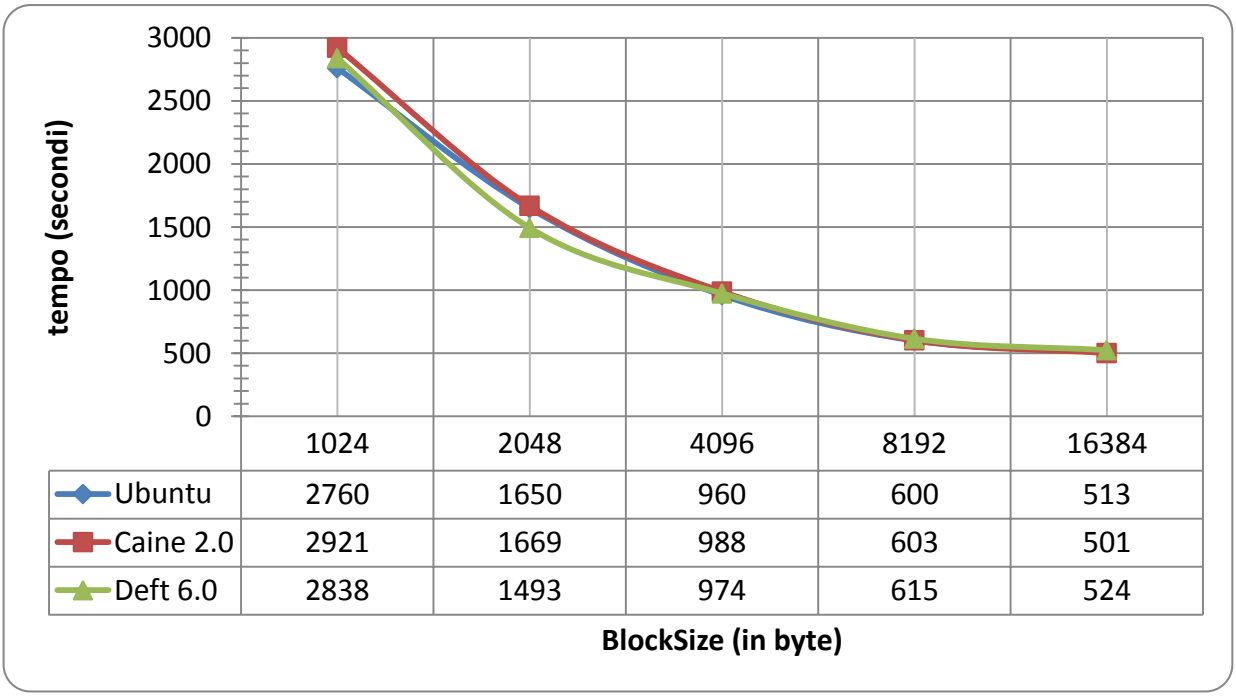


Figura 3.4

Da essi è possibile notare come i rapporti fra le distribuzioni misurati in precedenza in virtuale non siano rispettati in nativo. La spiegazione di tale differenza è da ricercare nelle differenti versioni del kernel adottate nelle distribuzioni e dei driver utilizzati per accedere all'hard disk da acquisire.

Si può osservare inoltre come i tempi di acquisizione fra live CD e distribuzione nativa non abbiano sensibili variazioni: ciò è ragionevole in quanto una volta caricati in memoria il sistema operativo ed il dd non è necessario alcun accesso al CD che ne rallenta l'esecuzione; ciò che varia sono, chiaramente, i tempi di avvio delle distribuzioni stesse.

L'utilizzo di una distribuzione in modalità live, sebbene rappresenti una rapida alternativa all'installazione su hard disk, è in letteratura fortemente sconsigliato (dove possibile) in quanto spesso l'utente dimentica di configurare in maniera opportuna spazi di swap su cui effettuare swapping in caso di saturazione della memoria RAM; occorre infatti ricordare che i tool di analisi di alto livello possono saturare molto rapidamente tale memoria e costringere il sistema operativo ad effettuare swap dei dati.

3.2 Analisi di basso livello con Autopsy

Completata la fase di acquisizione, si è proceduto ad un'analisi di basso livello dell'immagine ottenuta tramite il tool Autopsy, presente nella distribuzione Deft 6.

Autopsy è un'interfaccia grafica per il set di comandi TSK (The Sleuth Kit): lo Sleuth Kit è un insieme di librerie e tool a linea di comando che consentono di eseguire l'investigazione di un hard disk e di analizzarne il filesystem.

Nell'avvio di Autopsy dal live CD della Deft occorre porre particolare attenzione allo storage sul quale si intende effettuare il salvataggio dei dati d'interesse, in quanto di default Autopsy tenta di salvare i dati sul filesystem del cd (che ovviamente è read-only).

Per avviare Autopsy in modo che utilizzi una cartella presente su un hard disk writable (SEMPRE diverso da quello che si intende analizzare) si esegue da terminale il comando

autopsy –d </…/path-to-root-location>

Onde evitare di saturare l'hard disk di output (evento non segnalato da Autopsy!), si consiglia di utilizzarne uno capiente almeno quanto l'hard disk da analizzare.

Eseguendo il comando di avvio si aprirà il browser di default (Firefox su Deft) mostrando la schermata iniziale di Autopsy. Attraverso questa schermata è possibile definire il nome del caso al quale si lavora e inserire i dati degli investigatori che ci lavorano.

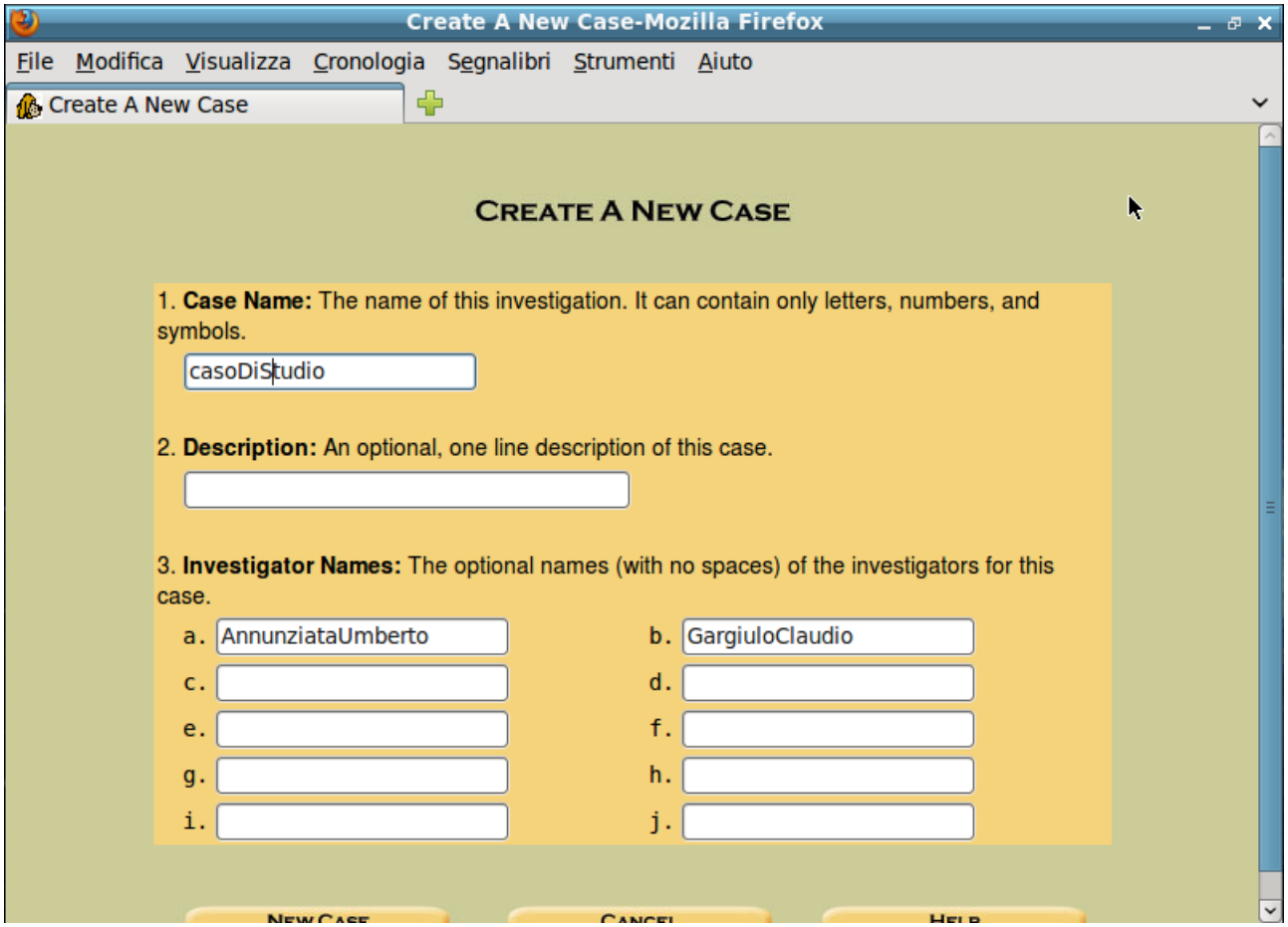


Figura 3.6

Dopo aver creato il caso è possibile aggiungere uno o più host ad esso. Ogni host rappresenta una delle macchine che deve essere analizzata per il caso in questione.

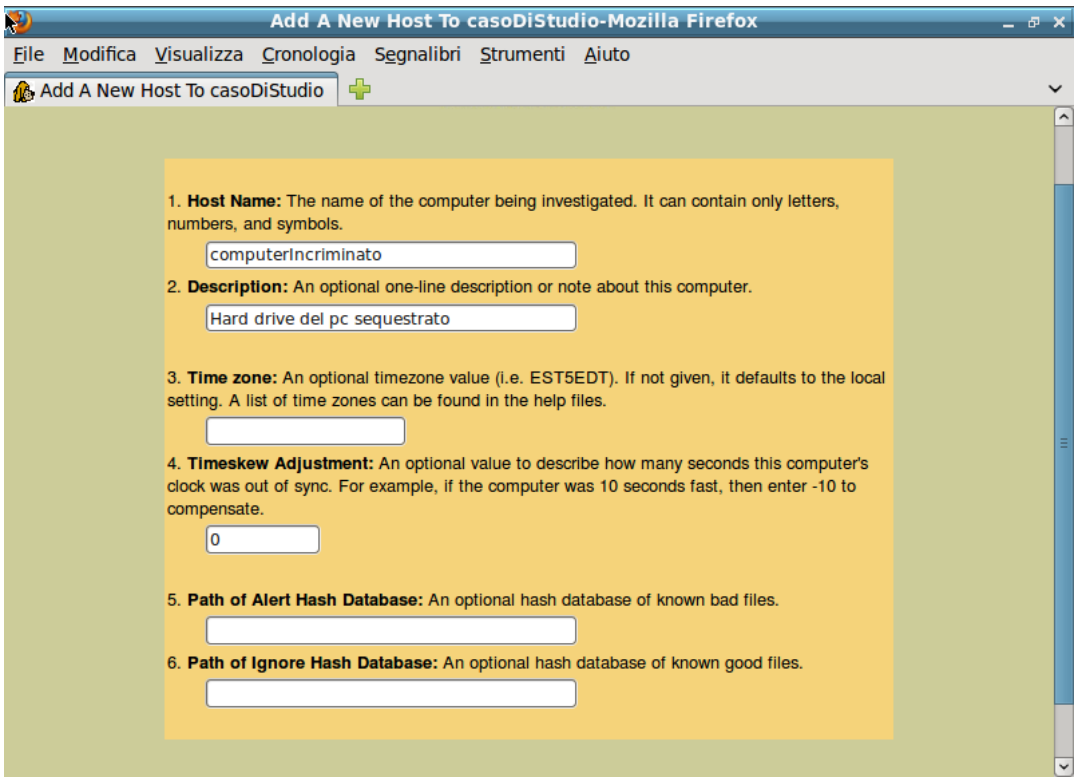


Figura 3.7

Dopo aver dato un identificativo ed una breve descrizione dell'host, viene mostrata la schermata principale di Autopsy:

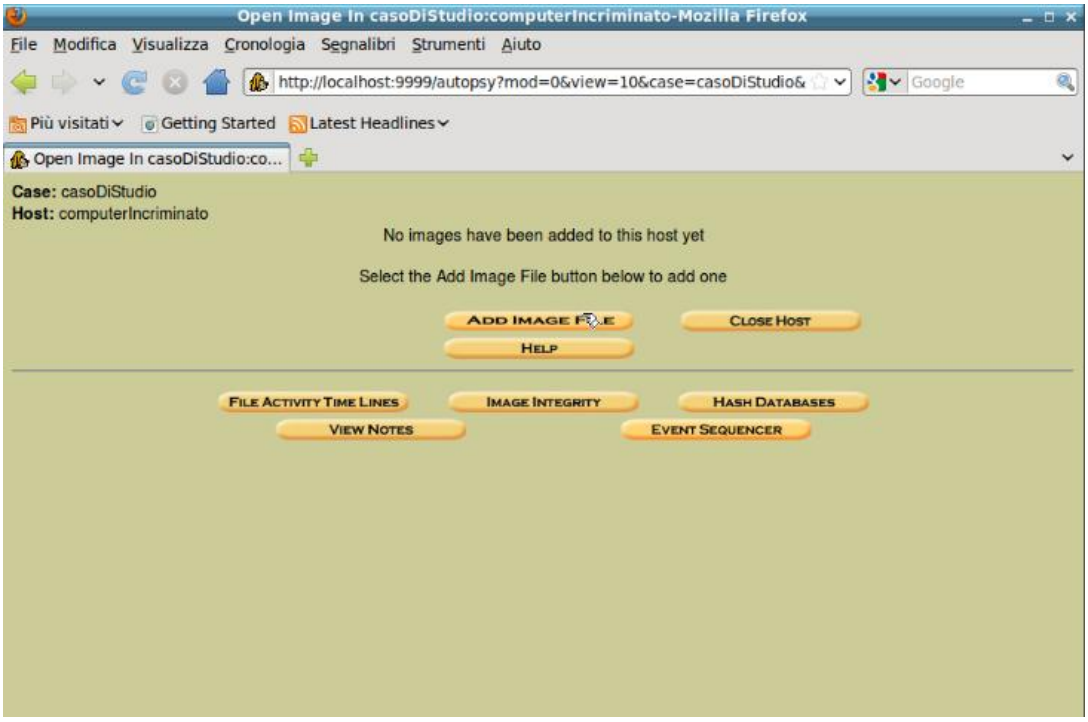


Figura 3.8

I comandi ai quali è possibile accedere sono i seguenti:

- File Activity Timeline: crea una timeline sulla base delle modifiche apportate ai file.
- Image integrity: verifica l'integrità di un'evidenza.
- Hash databases: aggiunge le firme di file noti per evitarli durante l'analisi.
- View notes: visualizza le note riguardanti l'host aperto.
- Event sequencer: permette di aggiungere note legate ad uno (o più) timestamp.

Il comando Add Image File, mostrato nella figura seguente, consente di aggiungere le immagini degli hard disk acquisiti nella fase precedente.

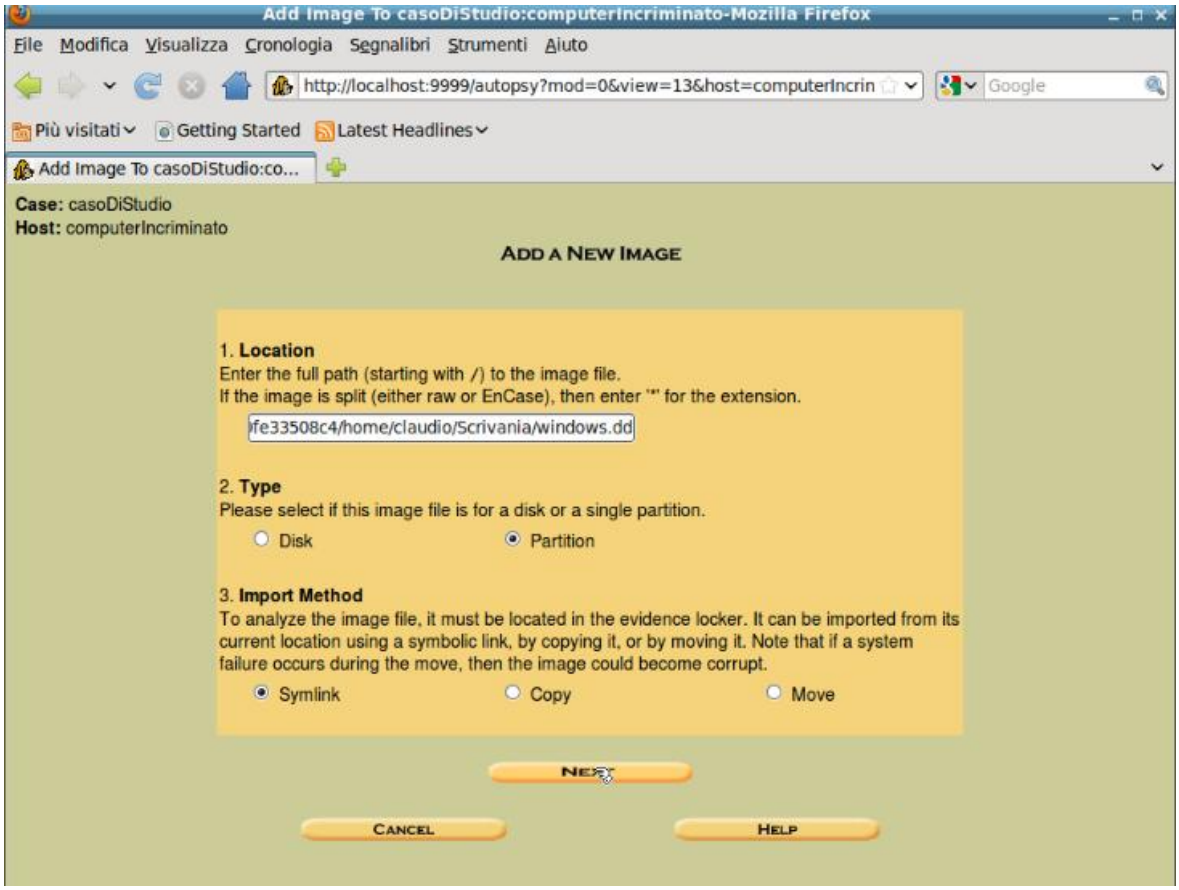


Figura 3.9

Premendo "next", è possibile specificare ulteriori opzioni di inserimento:

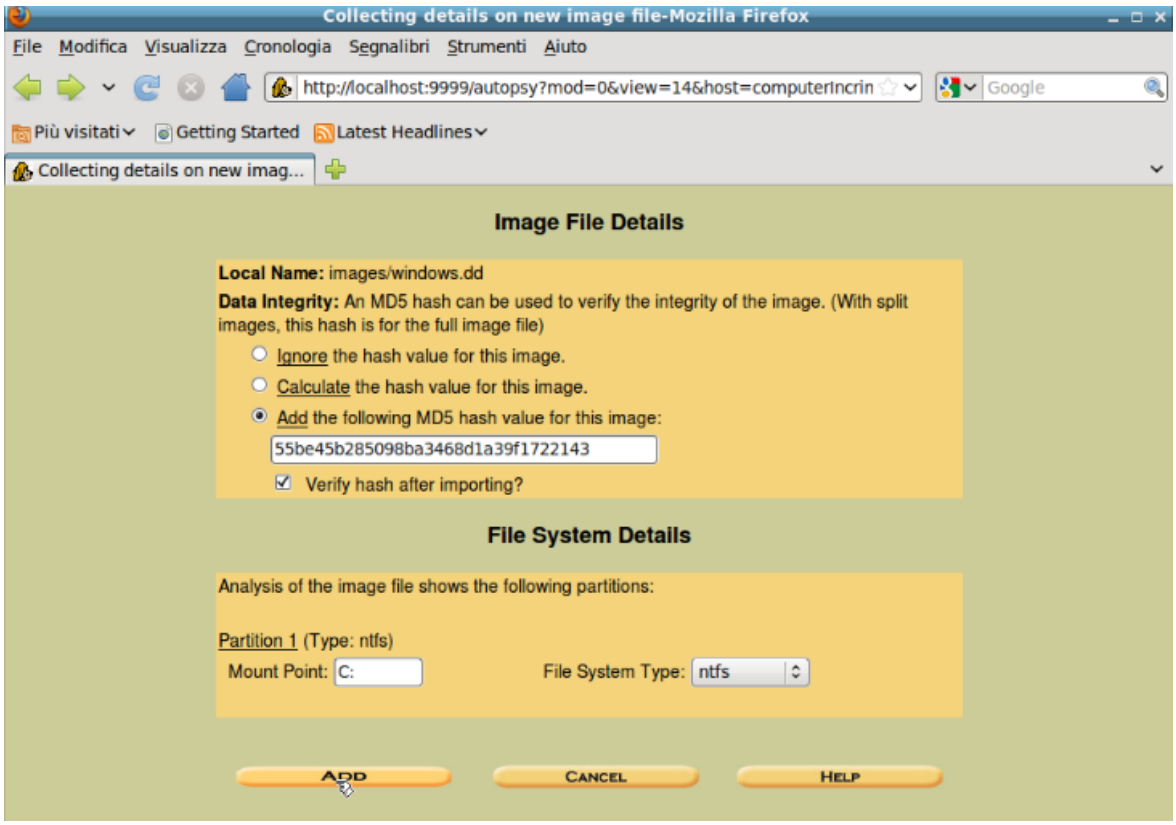


Figura 3.10

Una volta aggiunta l'immagine è possibile procedere all'analisi vera e propria attraverso i seguenti comandi:

• File Analysis: consente di navigare il filesystem dell'immagine selezionata e visualizzare i dettagli di ogni file. Tuttavia tale comando non mette in evidenza alterazioni eseguite dei nomi dei file e lo stato visibile/invisibile di ognuno di essi.

Ad esempio, il file Testo.txt mostrato nella figura seguente potrebbe sembrare un semplice file testuale, mentre aprendolo si scopre che esso, in realtà, è un file immagine avente estensione modificata.

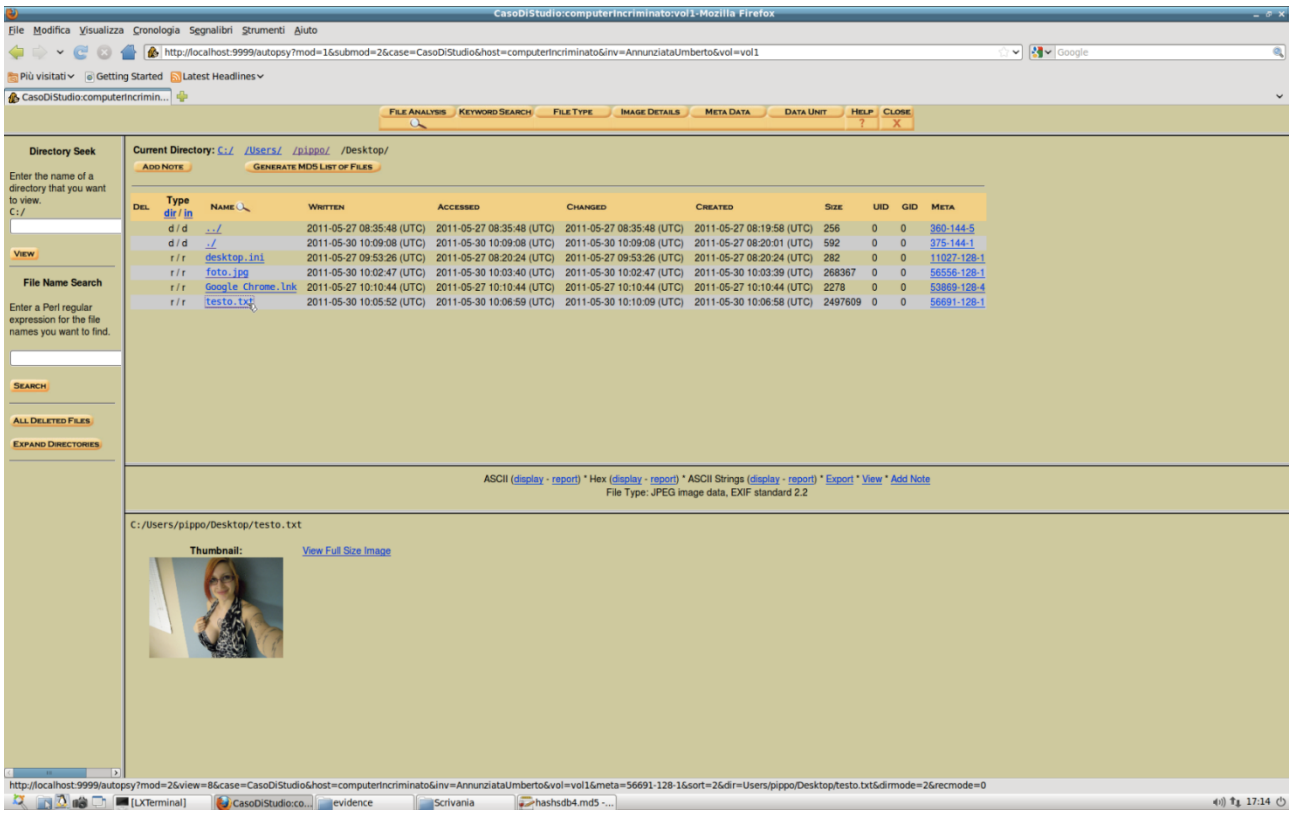


Figura 3.11

• Keyword Search: consente di eseguire una ricerca testuale di basso livello sull'immagine. Consente di utilizzare espressioni regolari grep per meglio indirizzare la ricerca verso il risultato desiderato.

• File Type sorting: salva in un archivio web i nomi dei file categorizzati per tipo e, se selezionata l'opzione appropriata, salva una copia dei file stessi. Questa operazione effettua l'extension ed il file type validation, assenti nel file manager mostrato in precedenza.

Di seguito viene riportato l'esito del File Type Sorting sul caso di studio portato in esempio:

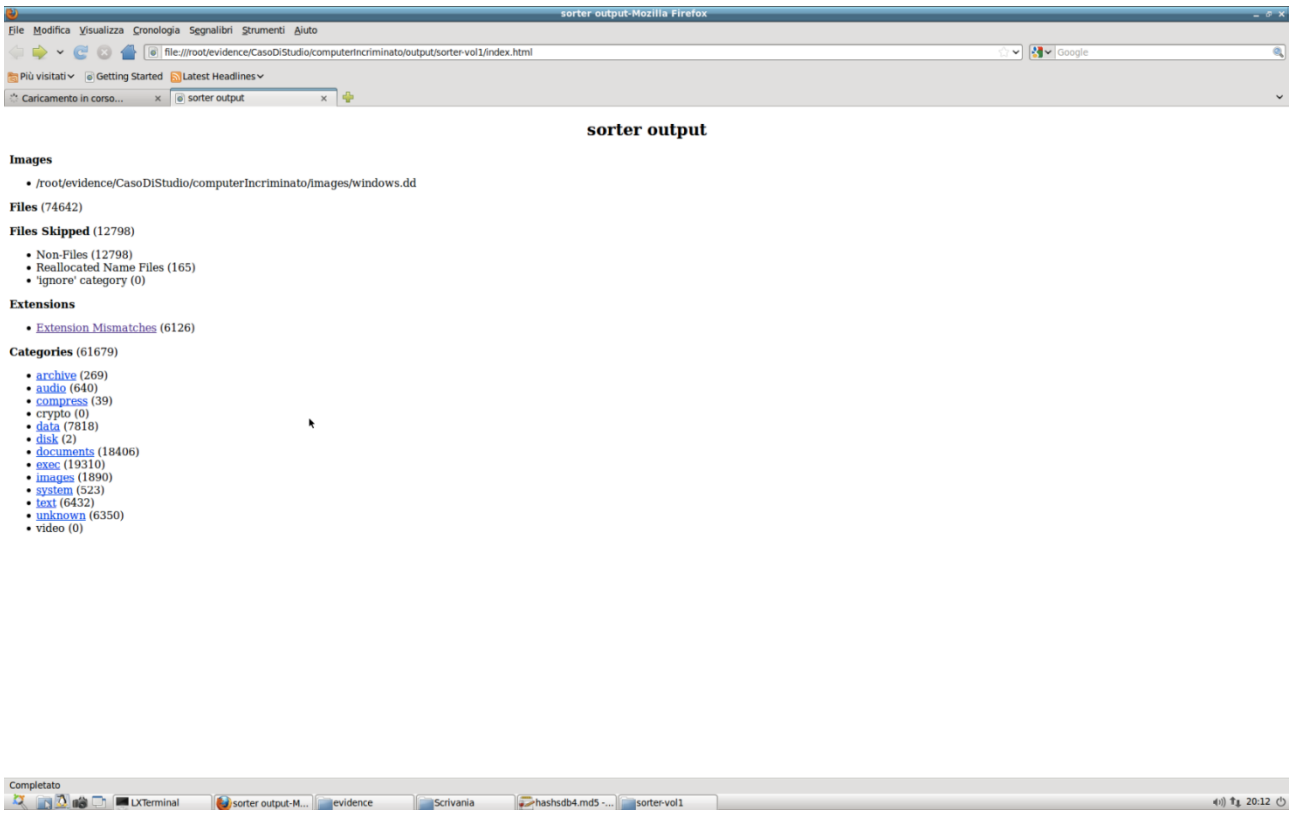


Figura 3.12

E' possibile notare l'elevato numero di extension mismatches (6126) rilevati: tuttavia soltanto uno, ossia il file immagine modificato da .jpg a .txt, è un extension mismatch "reale"; tutti gli altri sono falsi positivi di file del sistema operativo.

Si ritiene che un tool che miri ad assistere l'investigatore nella sua indagine debba essere dotato di un archivio di file da ignorare durante la scansione (attraverso controlli approfonditi come hash o firma degli stessi) in modo tale da evitare la rilevazione di falsi positivi, con conseguente perdita di tempo per l'investigatore.

• Timeline

Autopsy consente di rilevare, sulla base della data di modifica dei file, gli intervalli di tempo in cui la macchina in analisi è stata utilizzata dall'utente.

Prima che Autopsy possa procedere alla creazione della timeline è richiesta però la creazione di una struttura dati chiamata DataFile; tale procedura permette ad Autopsy di analizzare il filesystem e creare un unico file contenente tutte le informazioni necessarie per il processing successivo.

Una volta completata la creazione, i risultati possono essere consultati accedendo al file di log oppure tramite l'interfaccia di Autopsy (riportata nella figura sottostante).

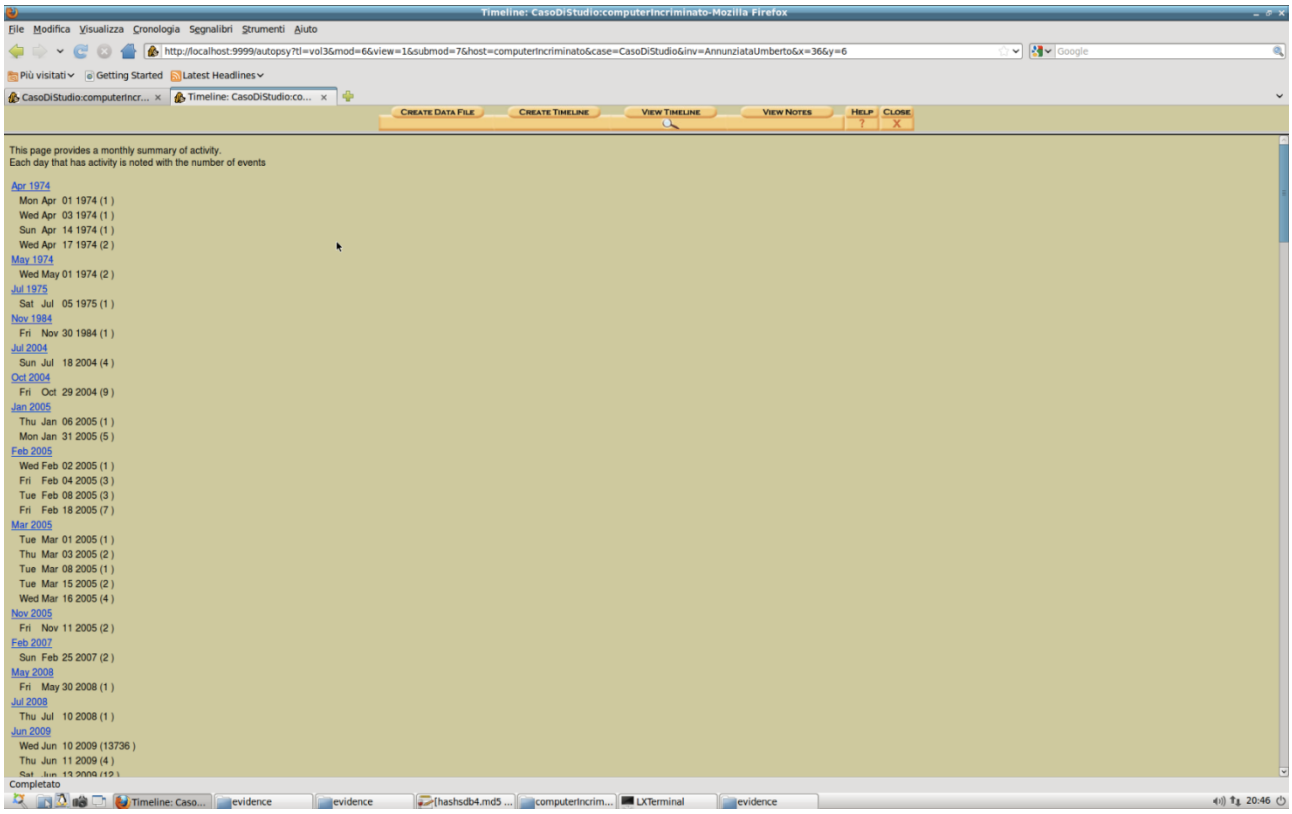


Figura 3.13

3.3 Analisi di basso livello con PTK

Il framework PTK è incorporato all'interno della distribuzione Linux Backtrack 5, la quale ne facilita l'utilizzo. Ovviamente può essere installato anche su altre distribuzioni. Come abbiamo già detto, esso è un miglioramento di Autopsy, in quanto: offre un'interfaccia grafica migliore; utilizza tecnologia Ajax; offre un motore di indicizzazione che mappa su un database tutte le informazioni dell' evidenza da analizzare, in modo da rendere le query più veloci ed ottimizzate in fase di analisi. Inoltre mette a disposizione la funzionalità di generazione automatica di report. I requisiti software per far funzionare PTK sono i seguenti: sistema Linux, server Apache, server MySql. I requisiti hardware sono i seguenti: Pentium 4 2.33 GHz, 512 MB di RAM, 10 GB di memoria su disco (varia a seconda del numero di casi da gestire).

3.4 Installazione di PTK

L'installazione di PTK è guidata da Backtrack, basta scegliere nome utente e password per accedere al server MySql; nome utente e password per poter accedere a PTK; un indirizzo di posta elettronica sul quale ci verrà inviata la licenza free. Una volta che avremo ricevuto la licenza (è un file con estensione .lic), basterà copiarla nella cartella /var/www/ptk/config.

3.5 Utilizzo di PTK

Una volta installato PTK, è possibile avviarlo accedendo da browser in localhost a http://localhost/ptk/index.php come indicato in Figura 3.14.

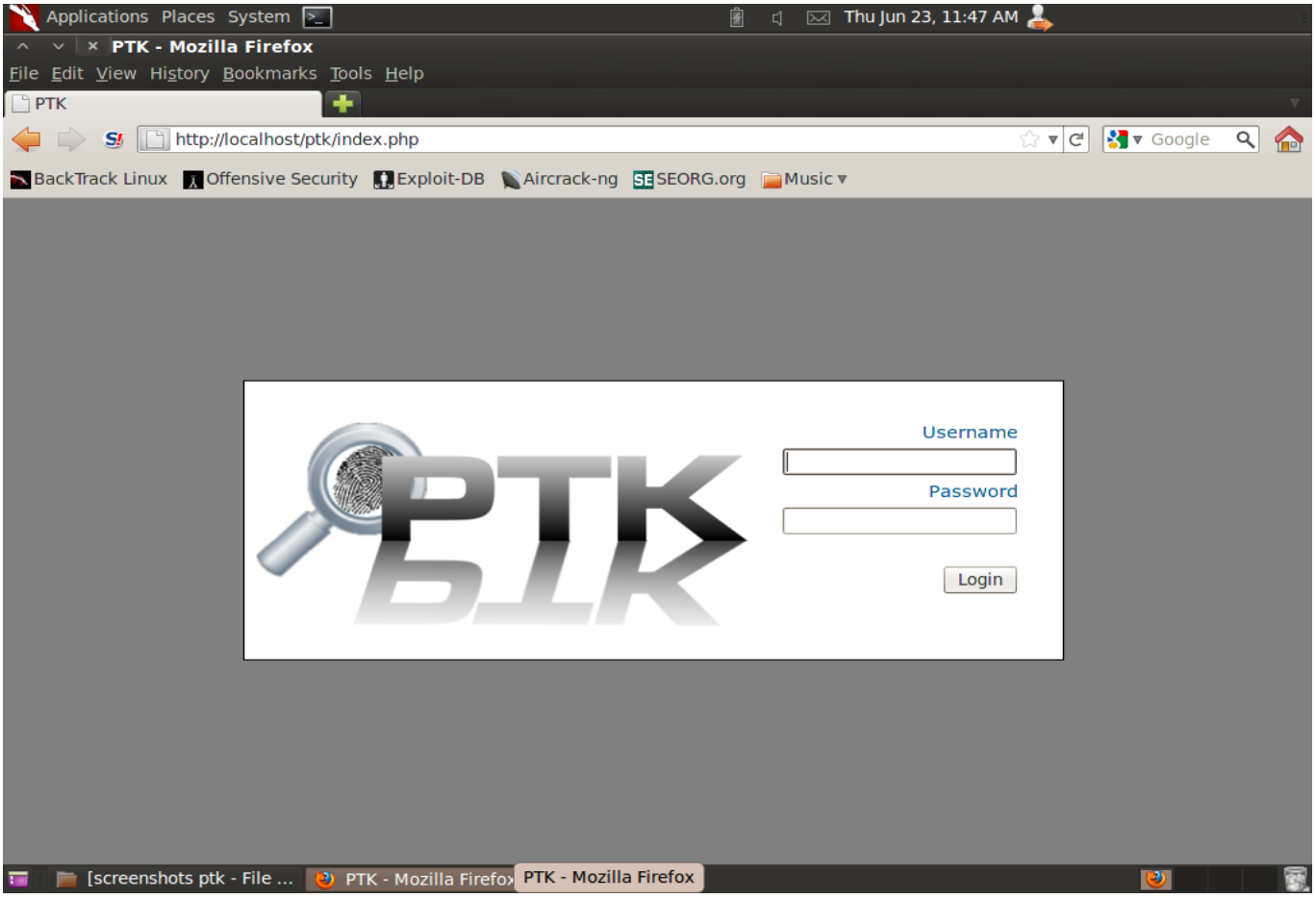


Figura 3.14 – Schermata di avvio PTK

Dopo aver inserito nome utente e password scelti durante l'installazione, compare la schermata in figura 3.15, la quale permette la gestione dei casi.

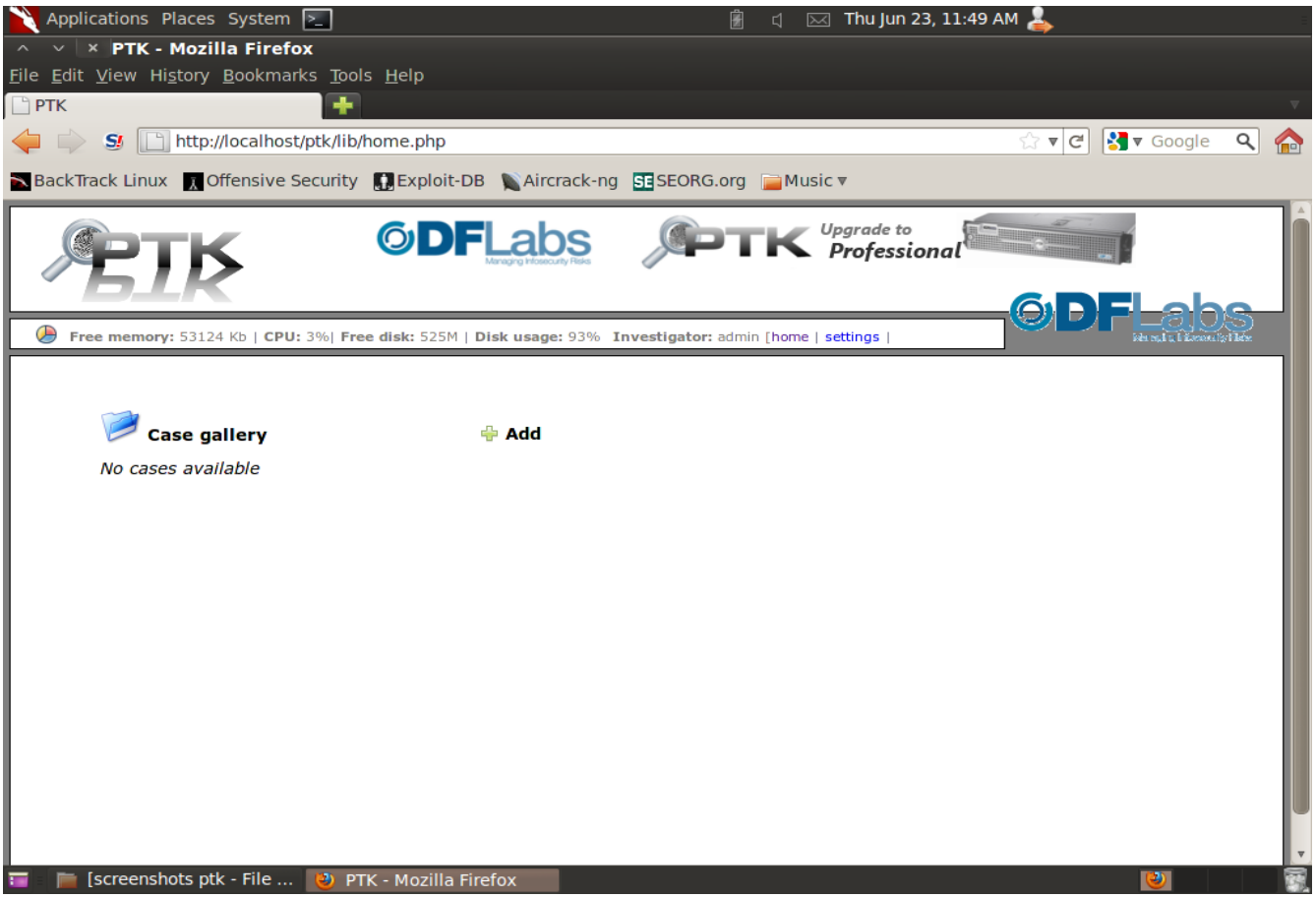


Figura 3.15 – Case management

È quindi possibile creare un nuovo caso cliccando su “Add” e inserendo, poi, le varie informazioni del caso, come in figura 3.16.

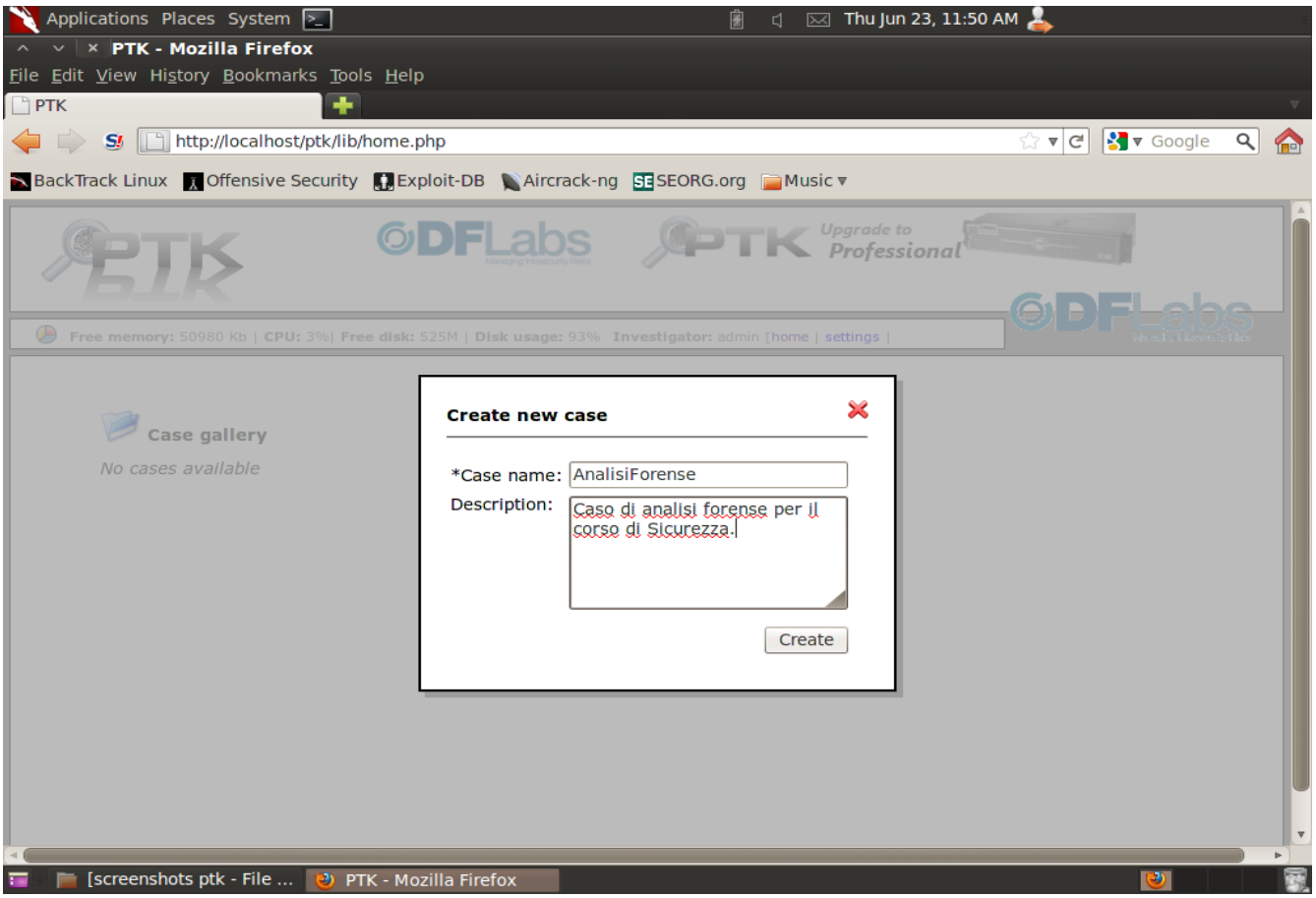


Figura 3.16 – Creazione nuovo caso

Cliccando poi su “Create”, il caso viene creato.

Figura 3.17 – Lista casi creati

Ci viene, quindi, mostrata la gallery dei casi esistenti (Figura 3.17). Per ogni caso ci viene mostrato il suo nome ed una serie di operazioni che possiamo effettuare sul caso. Il cerchietto verde ci permette di gestire le evidenze importate per quel caso, il cerchietto grigio ci permette di aggiungere o rimuovere investigatori al caso (Figura 3.18), il cerchietto azzurro ci permette di visualizzare a video diverse informazioni sul caso (Figura 3.19), il cestino ci permette di eliminare il caso, il lucchetto ci permette di chiudere il caso una volta che l'analisi è stata portata a termine.

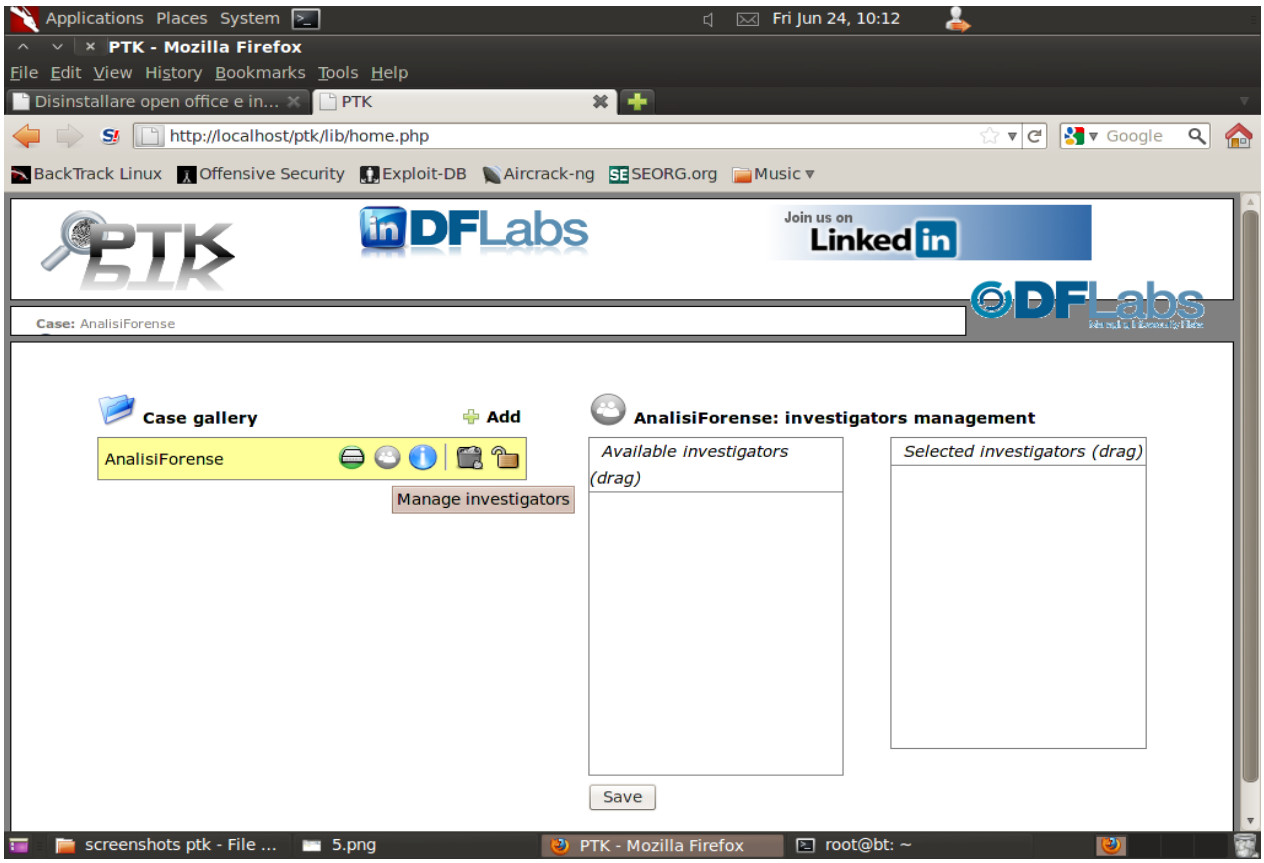


Figura 3.18 – Gestione investigatori

Figura 3.19 – Informazioni sul caso

Come già detto, il cerchietto verde ci permette di gestire le evidenze importate oppure importare una nuova evidenza.

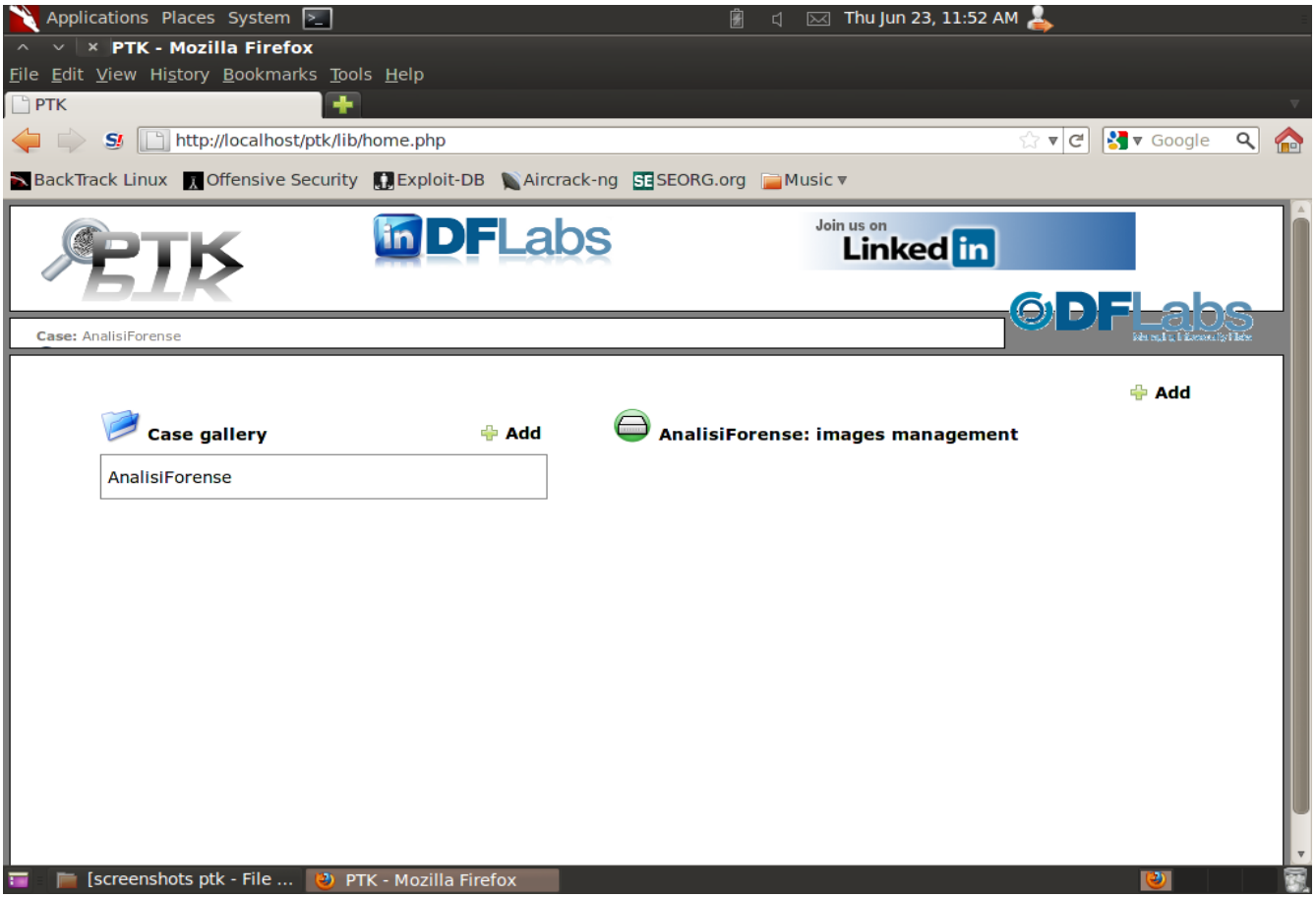


Figura 3.20 - Gestione delle immagini

Cliccando sul pulsante “Add” (in alto a destra) in figura 3.20, è possibile aggiungere una nuova immagine al caso.

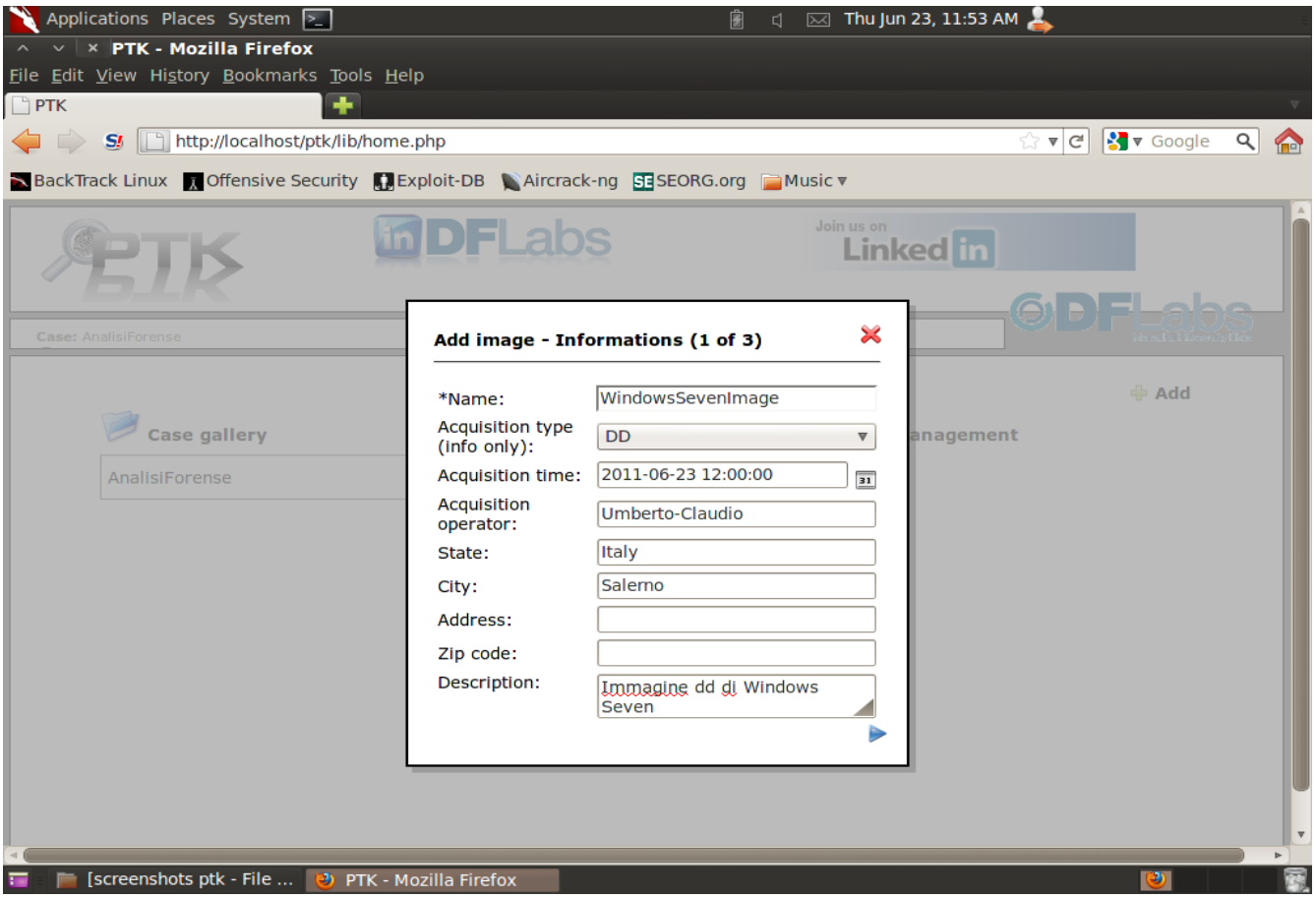


Figura 3.21 – Inserimento informazioni immagine

La figura 3.21 mostra la schermata che ci viene visualizzata per poter inserire le informazioni relative all'immagine da importare. Le figure 3.22 e 3.23 ci mostrano le schermate che ci vengono visualizzate per poter inserire il path dell'immagine che si vuole importare.

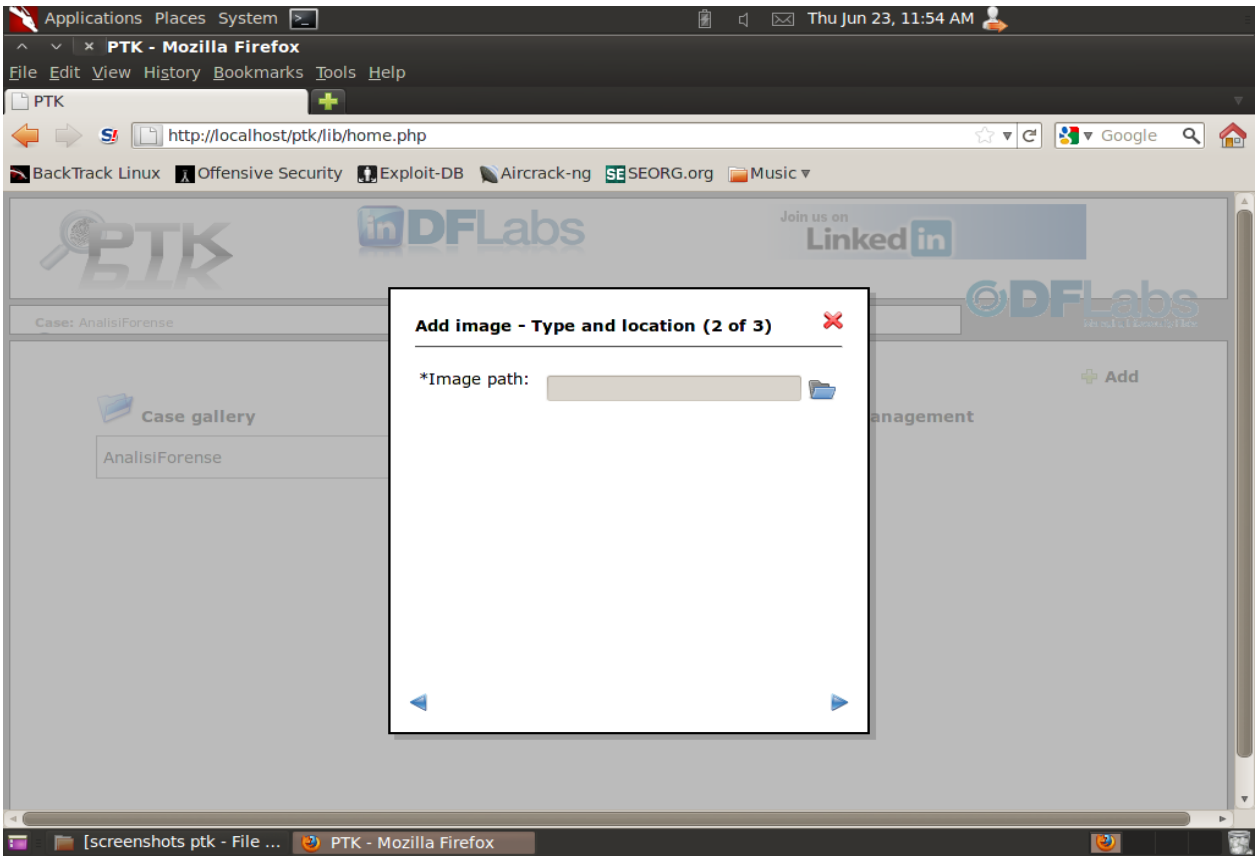


Figura 3.22 – Path dell'immagine

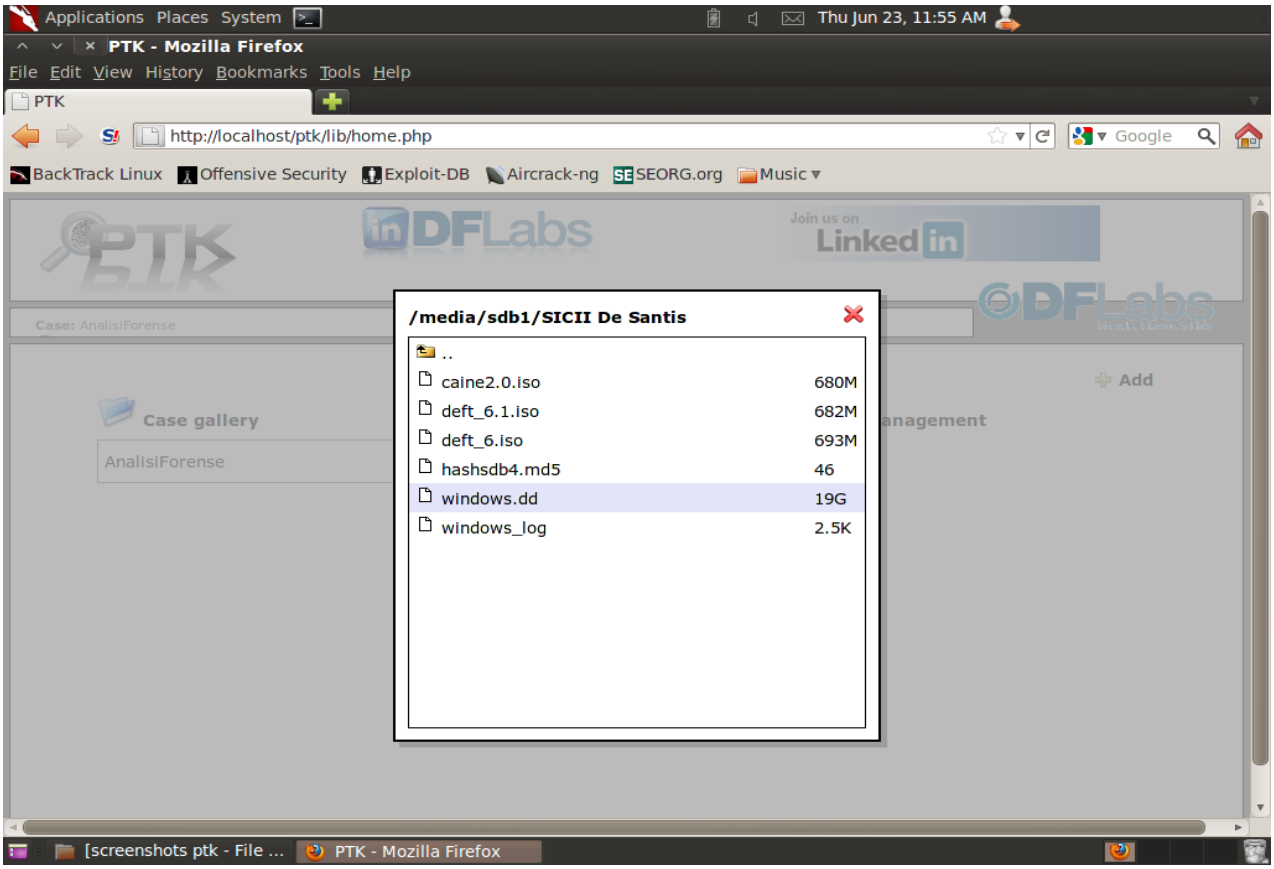


Figura 3.23 – Path dell'immagine windows.dd

Possiamo poi scegliere il metodo con cui effettuare la copia (Figura 3.24), attraverso link simbolico oppure copia vera e propria.

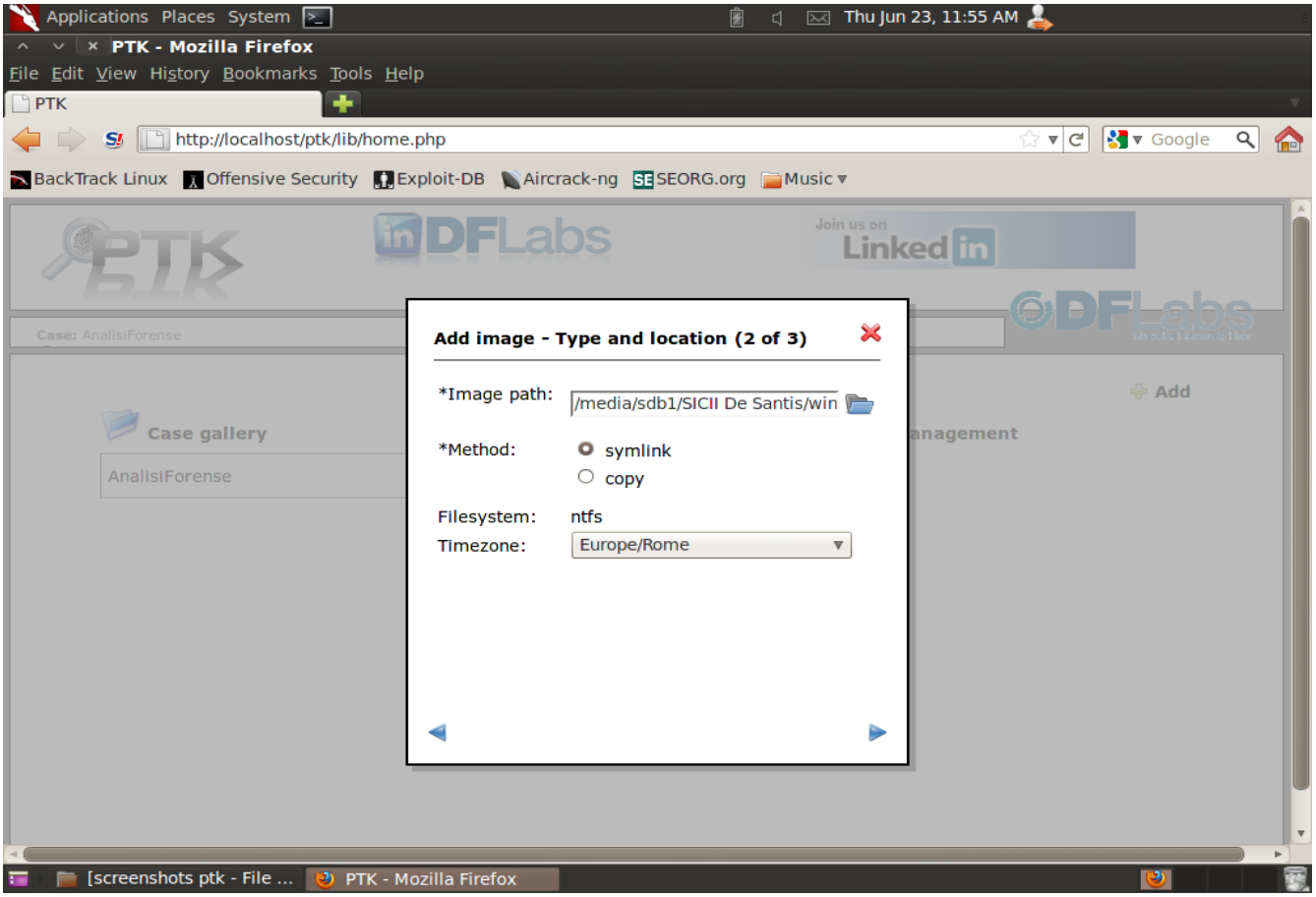


Figura 3.24 – Tipologia di copia

Possiamo poi decidere se calcolare o meno gli hash md5 e sha1 sull'immagine (Figura 3.25)

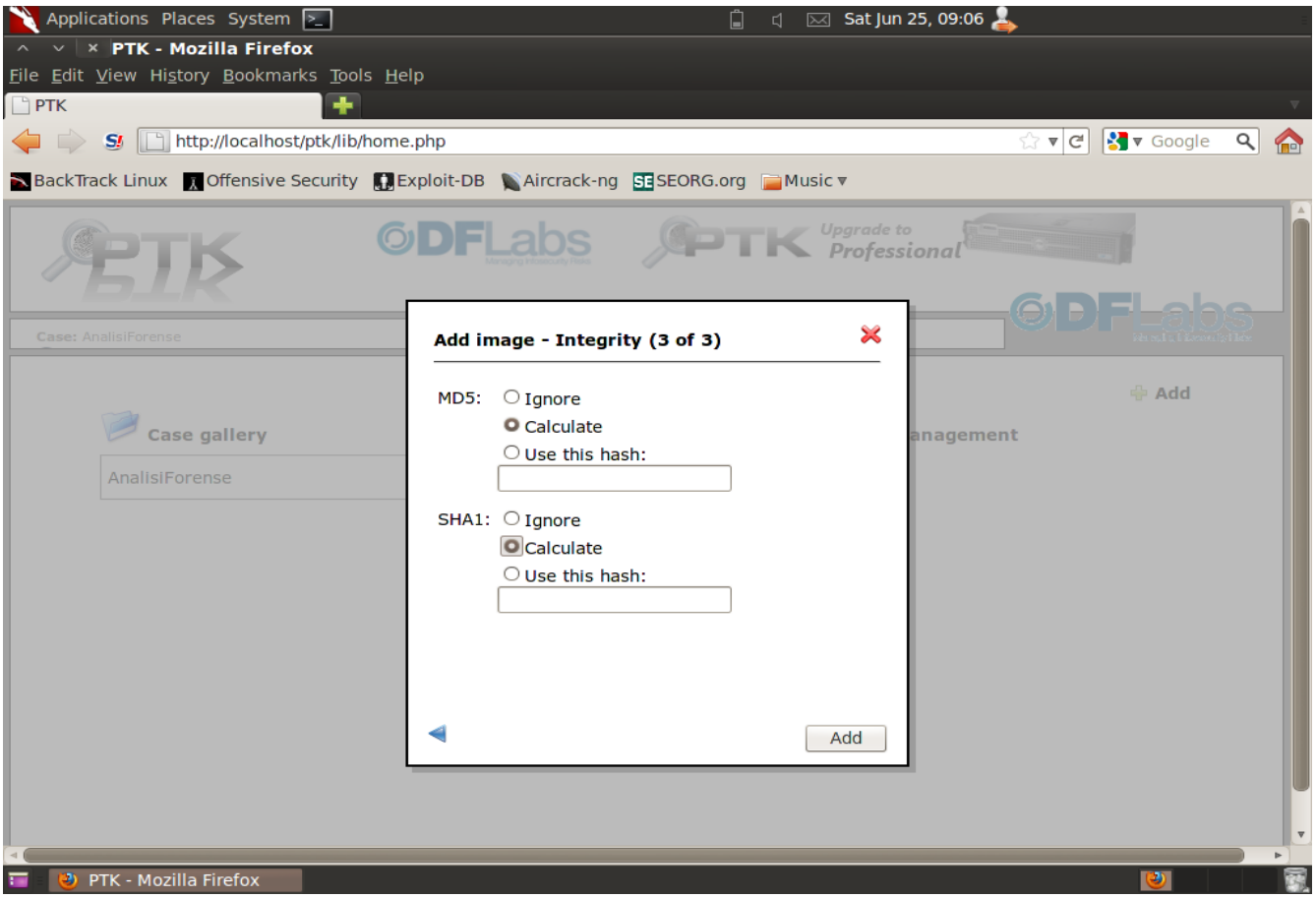


Figura 3.25 – Scelta calcolo hash

A questo punto, come indicato in Figura 3.26, ci viene mostrata l'immagine appena importata e le operazioni che è possibile effettuare su di essa. Il cerchietto rosso ci permette di effettuare l'indicizzazione dell'immagine (Figura 3.27); il cerchietto grigio ci permette di effettuare altre operazioni sull' immagine, come la timeline, la ricerca per parole chiave, ecc.; il pallino verde ci permette di effettuare il controllo di integrità dell'immagine; il pallino celeste ci permette di visualizzare le informazioni sull'immagine; il cestino ci permette di eliminare l'immagine.

Figura 3.26 – Operazioni possibili per l'immagine importata

In figura 3.27 ci viene mostrata la schermata visualizzata durante l'indicizzazione dell'immagine.

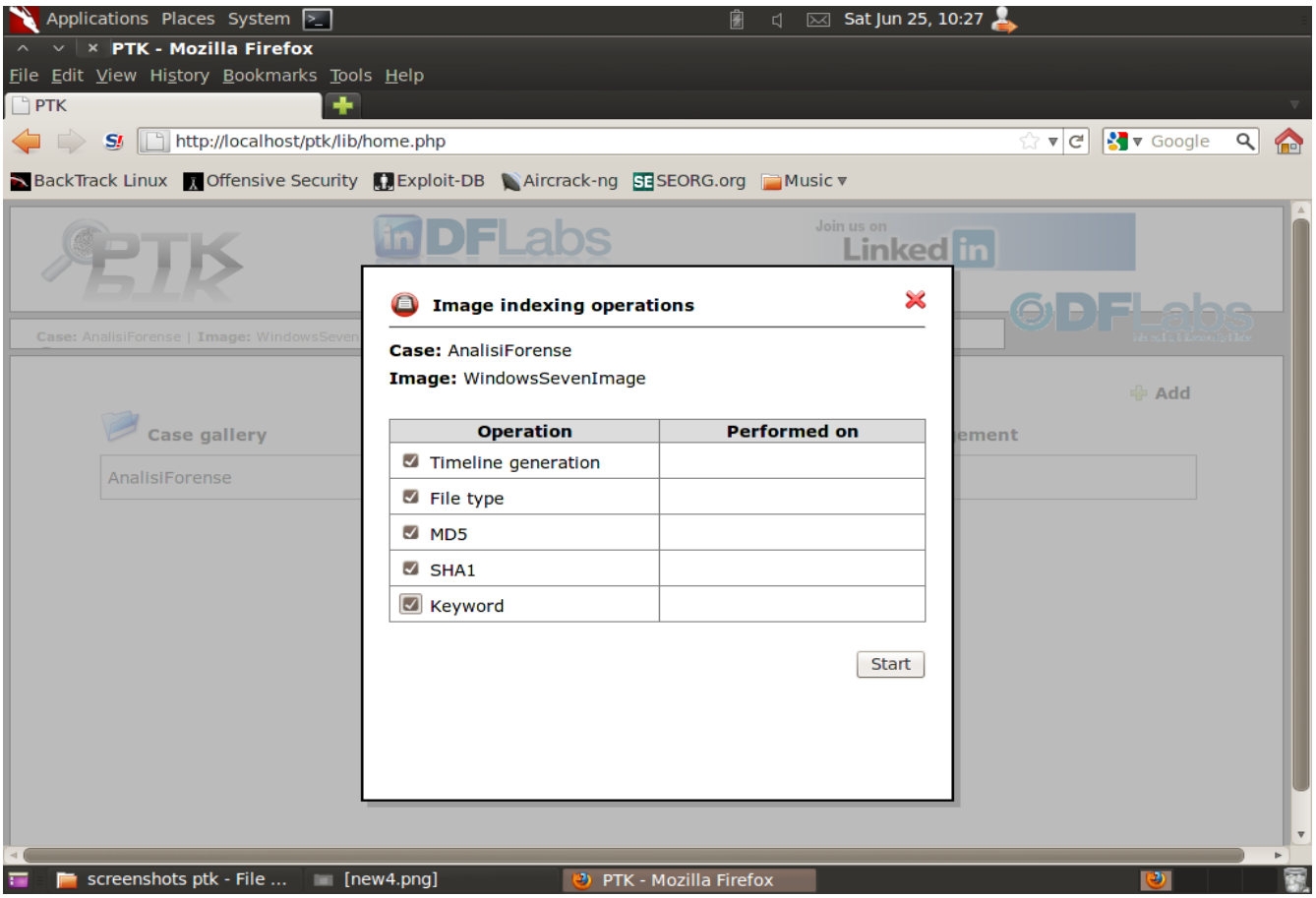


Figura 3.27 – Indicizzazione dell'immagine

Cliccando sul cerchietto grigio nella schermata visualizzata in figura 3.25, ci viene visualizzata la schermata che mostriamo in figura 3.28. A partire da sinistra verso destra notiamo tutte le funzionalità che ptk ci mette a disposizione per poter effettuare un'analisi: file analysis (permette di navigare all'interno del file system della partizione da analizzare), timeline (permette di visualizzare una timeline delle operazioni effettuate), keyword (permette di effettuare una ricerca per parole chiave), gallery (permette di visualizzare tutte le immagini in una cartella), image details (permette di visualizzare delle informazioni sull'immagine da analizzare), data unit (permette di fare un'analisi di più basso livello sui blocchi del disco), bookmark (permette di creare dei bookmark), reports (permette di generare in automatico il documento di report).

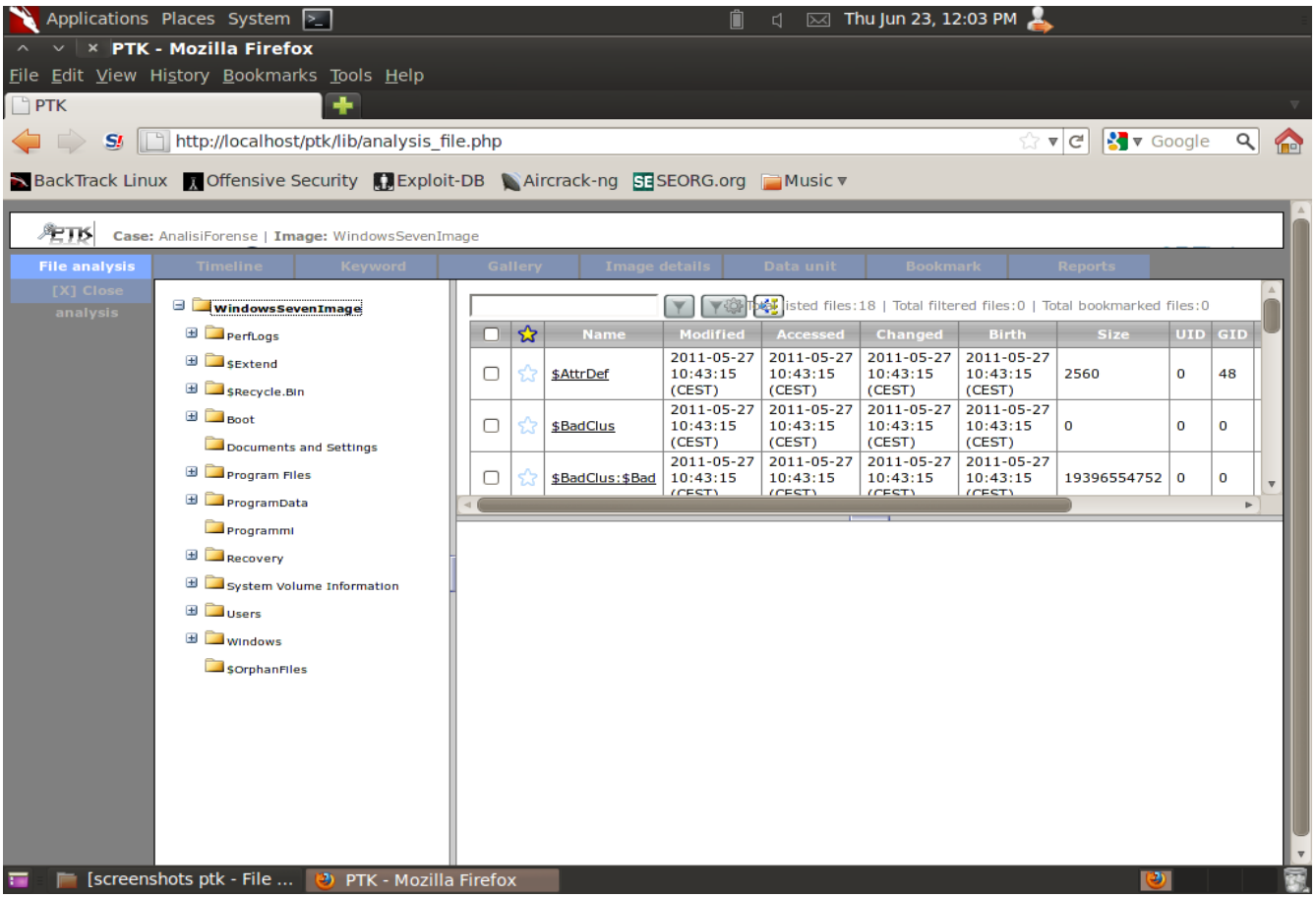


Figura 3.28 – Analisi dell'evidenza

Come prima operazione della nostra analisi abbiamo cercato nella nostra partizione il file testo.txt attraverso la funzionalità di file analysis, e abbiamo scoperto che si tratta di una immagine e non di un file di testo come avrebbe voluto farci credere la sua estensione. Eravamo stati noi stessi, come abbiamo precisato nella prima presentazione, a cambiare l'estensione da .jpg a .txt per impedire ad un investigatore di trovare facilmente l'immagine. L'operazione viene mostrata in figura 3.29.

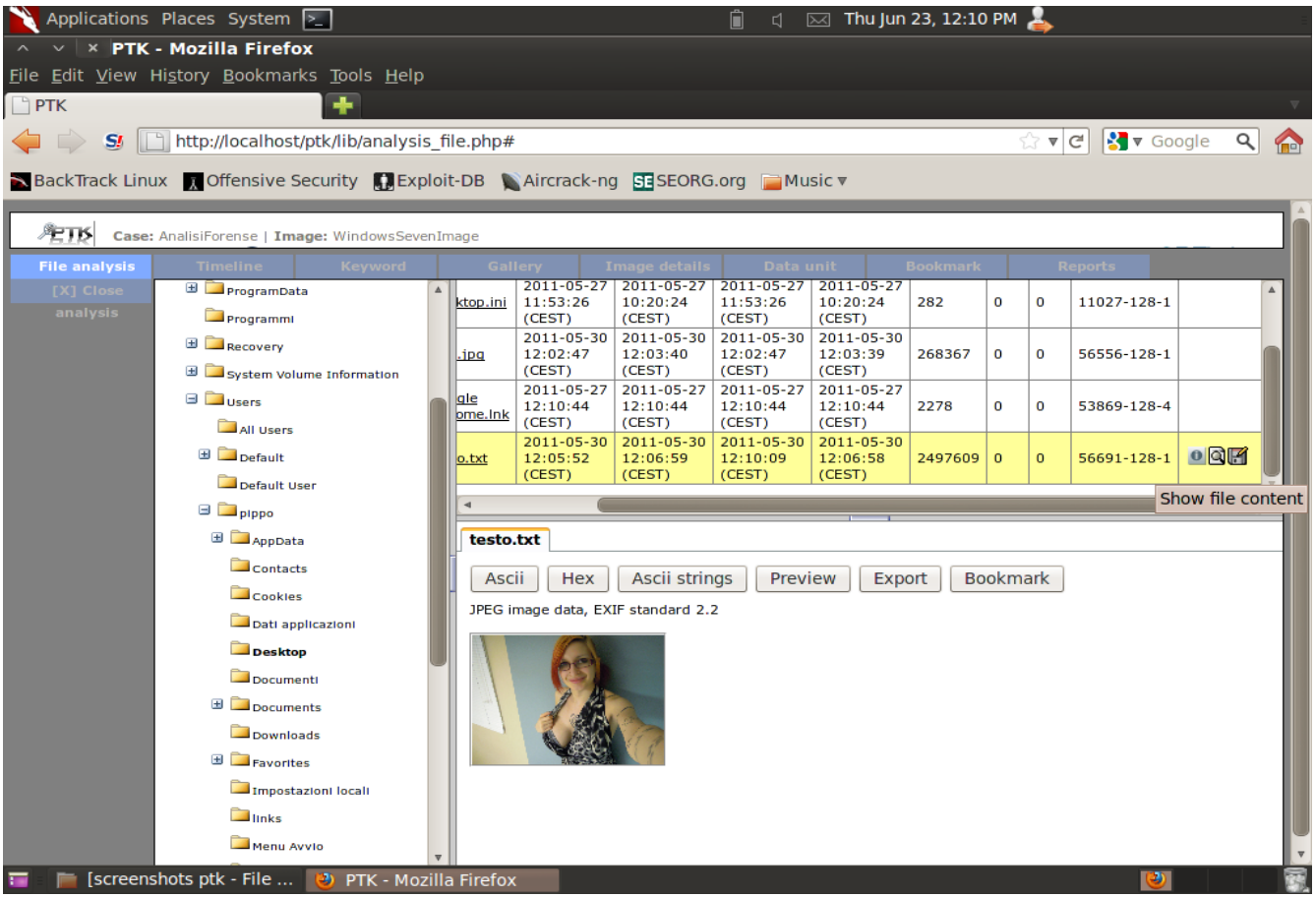


Figura 3.29 – Recupero dell'immagine testo.txt

Abbiamo poi utilizzato la funzionalità di timeline come mostrato in figura 3.30, notiamo che è possibile indicare una data di inizio e una data di fine e, in tal caso, ci viene restituita la lista di quelle operazioni che sono state effettuate in quell'intervallo di tempo. Nel nostro caso non abbiamo inserito le due date e, quindi, ci vengono mostrate tutte le operazioni effettuate in forma tabellare.

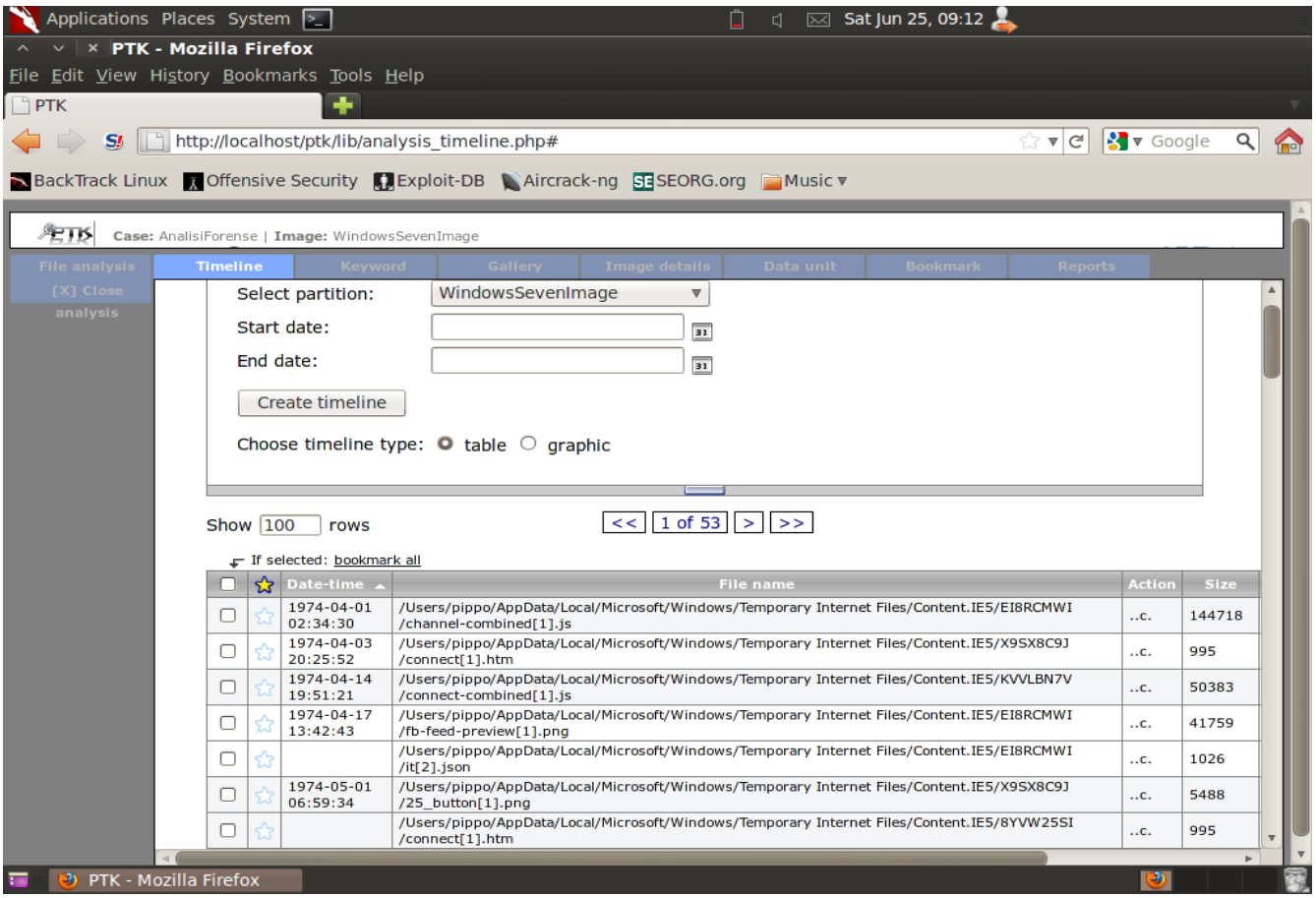


Figura 3.30 – Timeline

Abbiamo poi visualizzato la timeline anche in formato grafico (figura 3.31), ci viene mostrato un grafico nel quale vengono riportati gli anni sull'asse delle ascisse e, sull'asse delle ordinate con colori diversi, il numero di file acceduti (viola), il numero di file modificati (blu), il numero di file creati (verde).

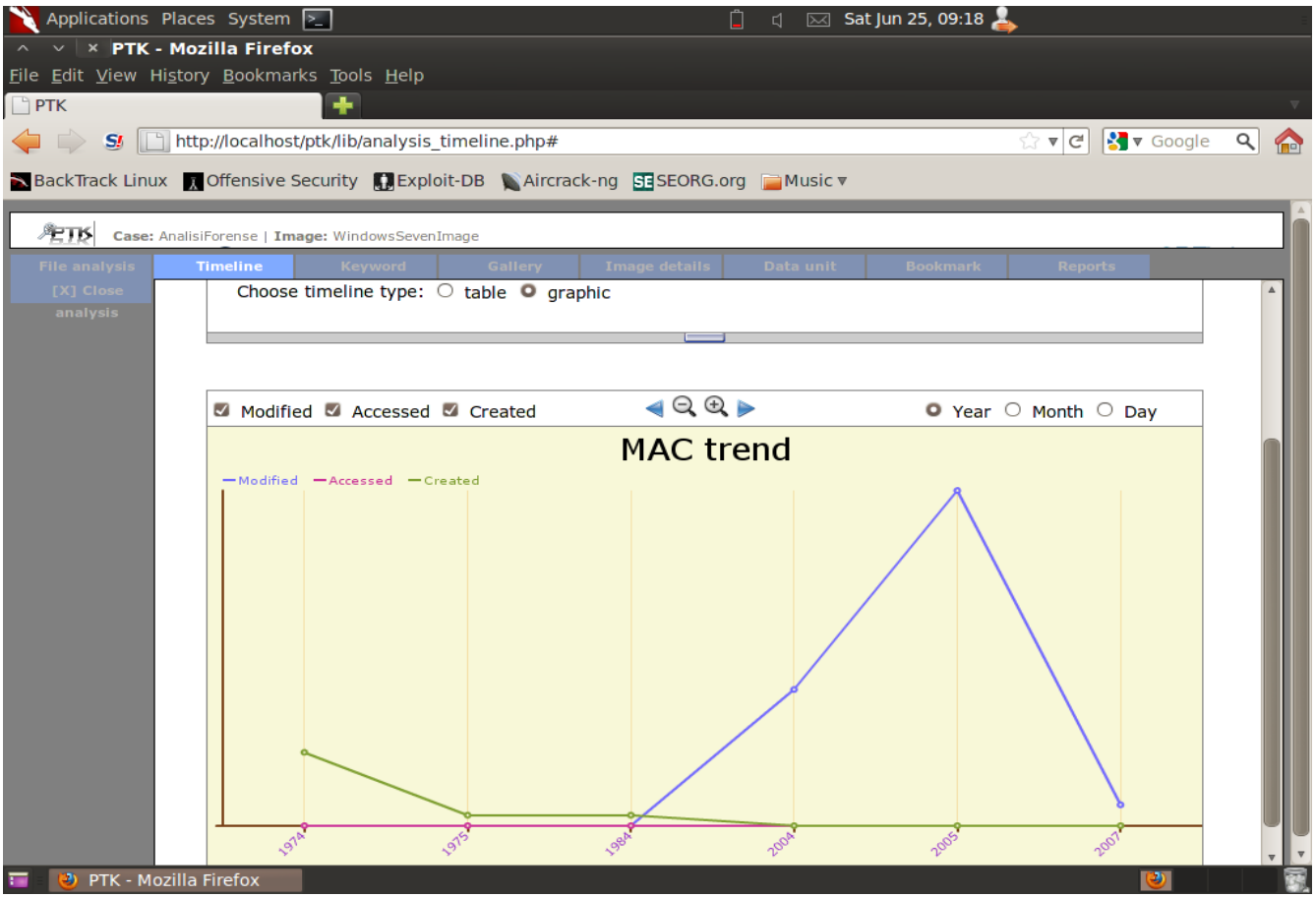


Figura 3.31 – Timeline in formato grafico

Abbiamo poi utilizzato la funzionalità di keyword (Figura 3.32) utilizzando come parola chiave "skype". È possibile inoltre indicare delle espressioni regolari per la ricerca e vengono messi a disposizione alcuni template di espressioni regolari come il numero di carta di credito, indirizzo email, indirizzo ip, etc.

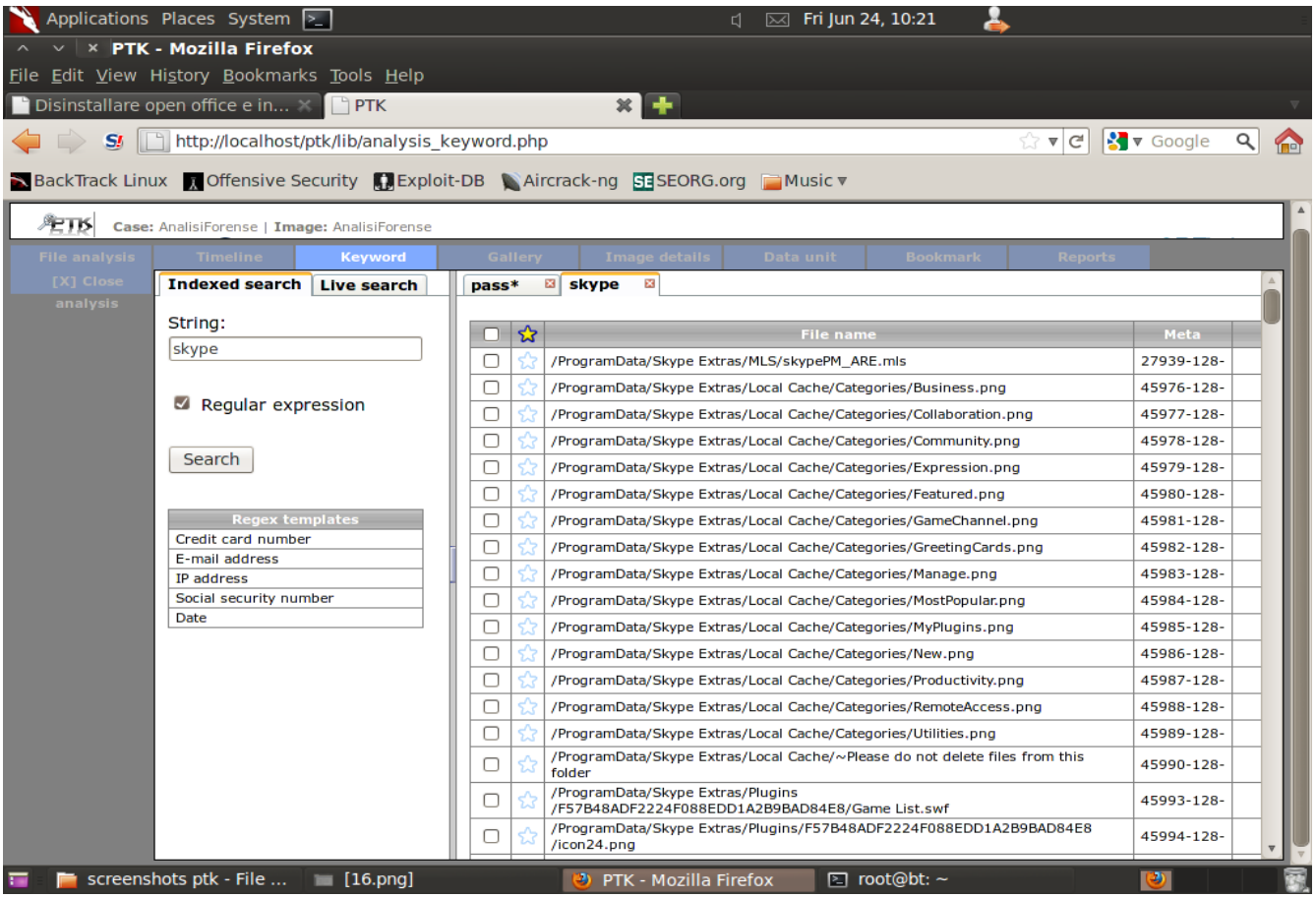


Figura 3.32 – funzionalità keyword

Abbiamo poi utilizzato la funzionalità di Gallery per poter visualizzare tutte le immagini che si trovano sul Desktop della partizione (Figura 3.33), notiamo che ci viene visualizzata anche l'immagine testo.txt, nonostante sia stata manipolata la sua estensione.

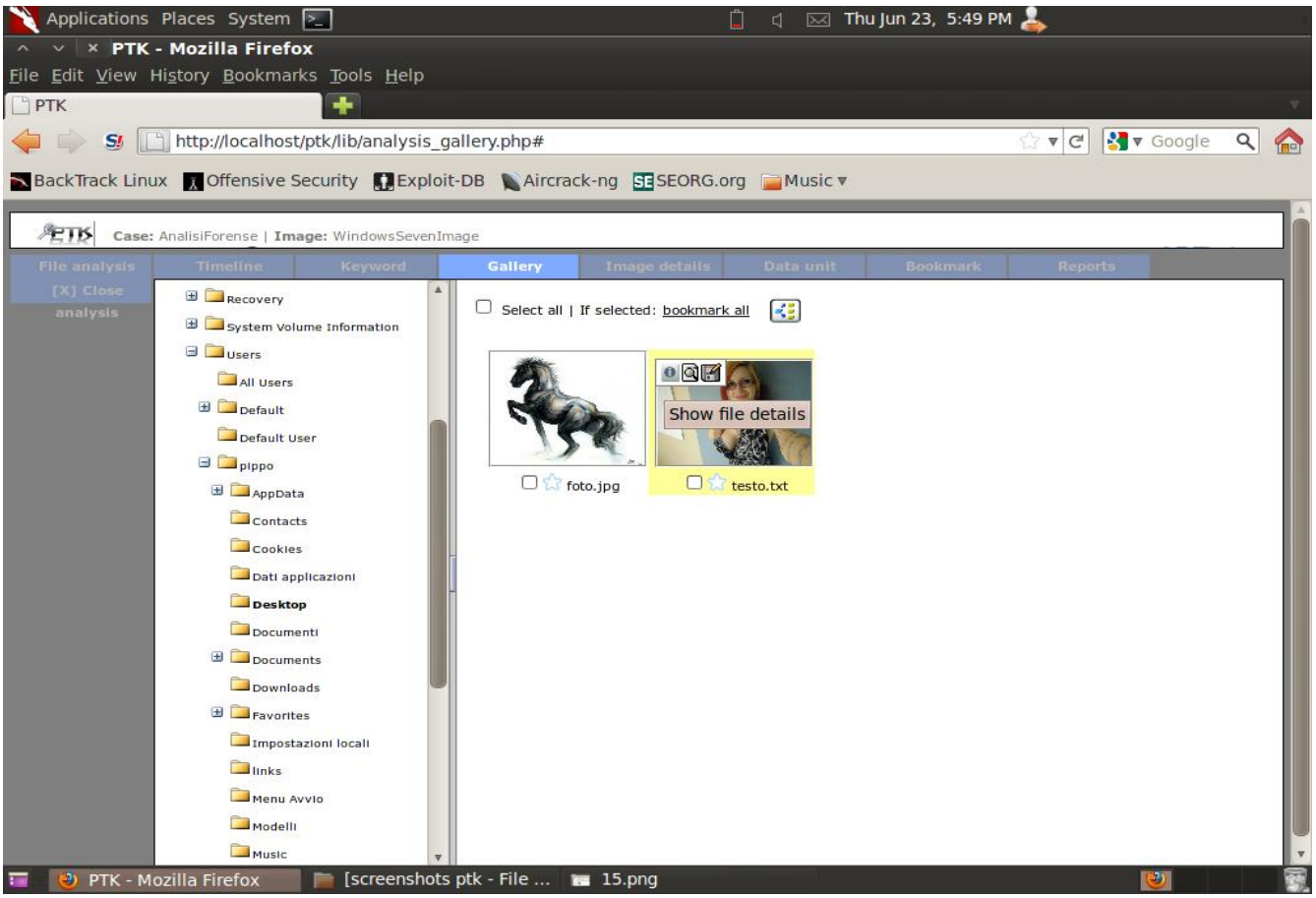


Figura 3.33 – Gallery

Abbiamo poi utilizzato la funzionalità di “Image details” (Figura 3.34) che ci dà tutte le informazioni sulla partizione che abbiamo analizzato. Non risultano corretti il Volume Serial Number (problema non rilevante in quanto corrispondente ad una partizione virtualizzata) e la versione del sistema operativo (non è windows xp ma windows 7).

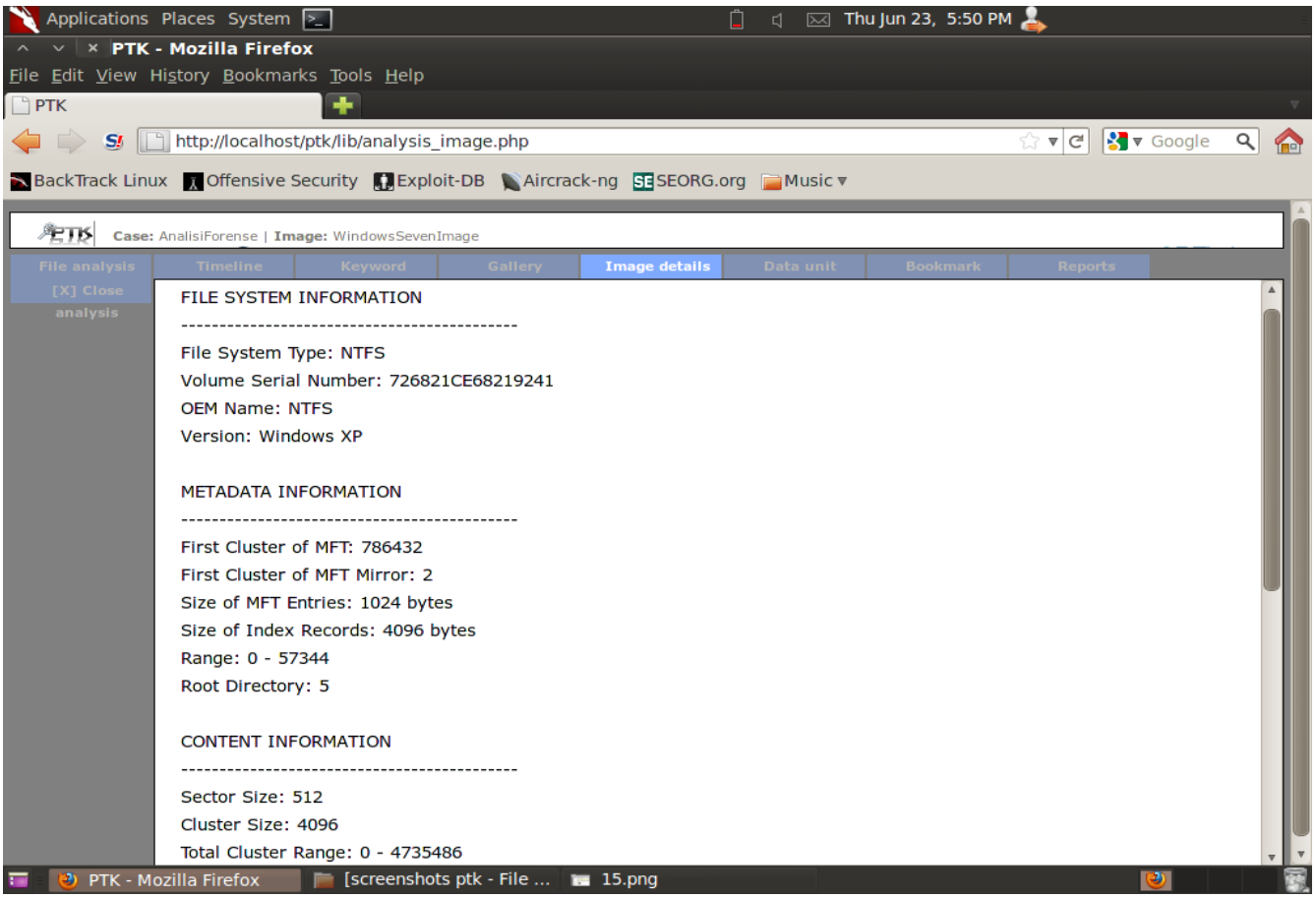


Figura 3.34 – Image details

In figura 3.35 viene mostrata la schermata relativa alla funzione di data unit che ci permette di effettuare una ricerca più a basso livello indicando il sector number di partenza e il numero di settori da visitare.

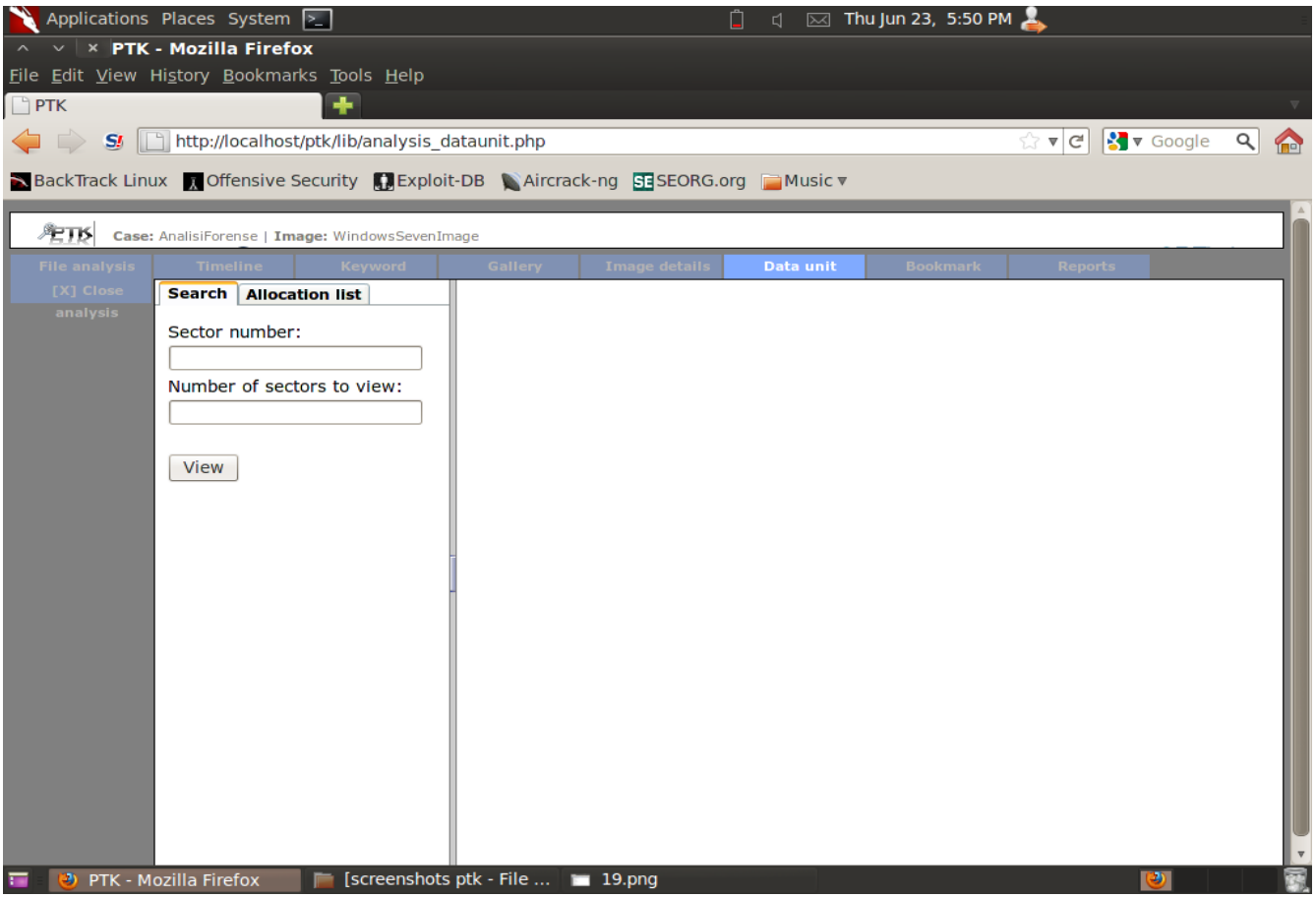


Figura 3.35 – Funzione di Data Unit

La funzione di Reports ci permette di generare in automatico un documento di report il cui nome sarà della forma data_ora_AnalisiForense.pdf (figure 3.36 e 3.37) che riporta tutte le informazioni sul caso e sull'evidenza analizzata.

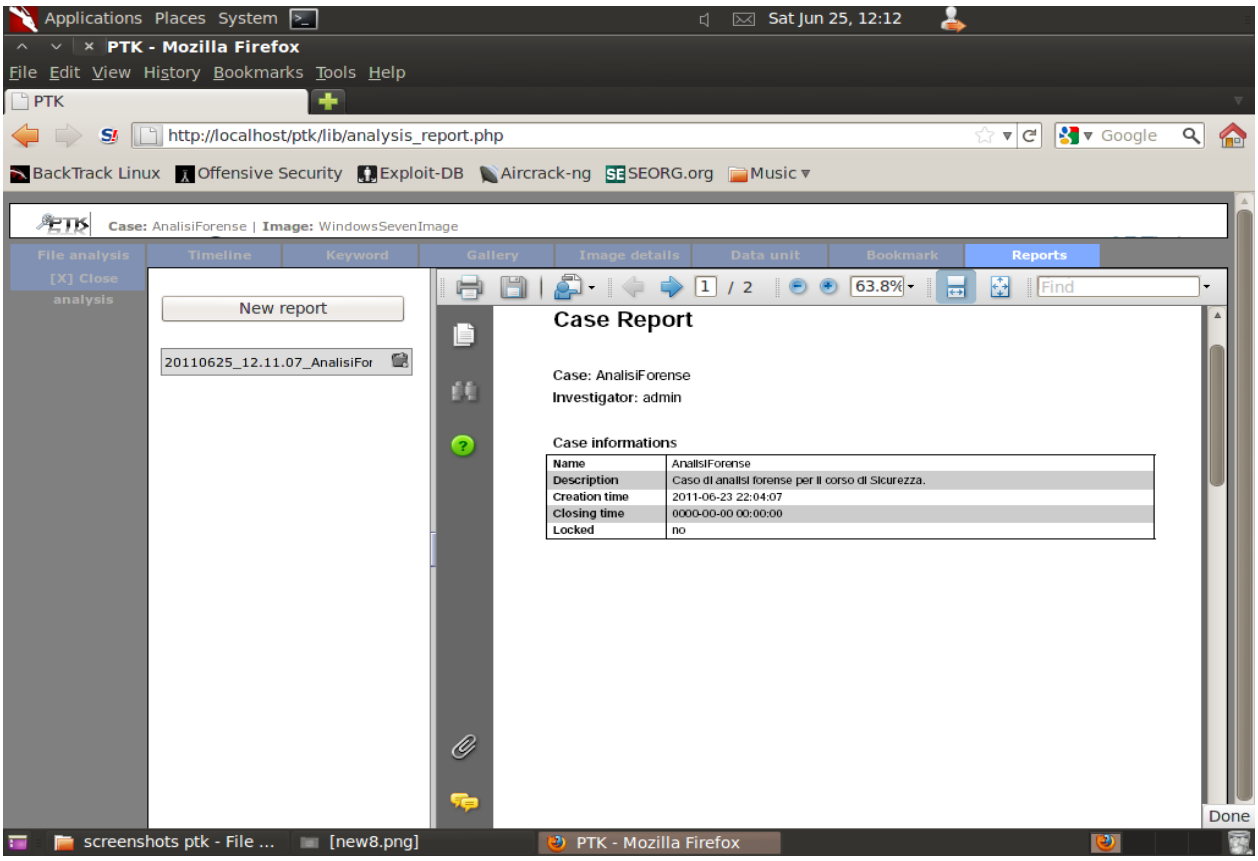


Figura 3.36 – Prima pagina del report

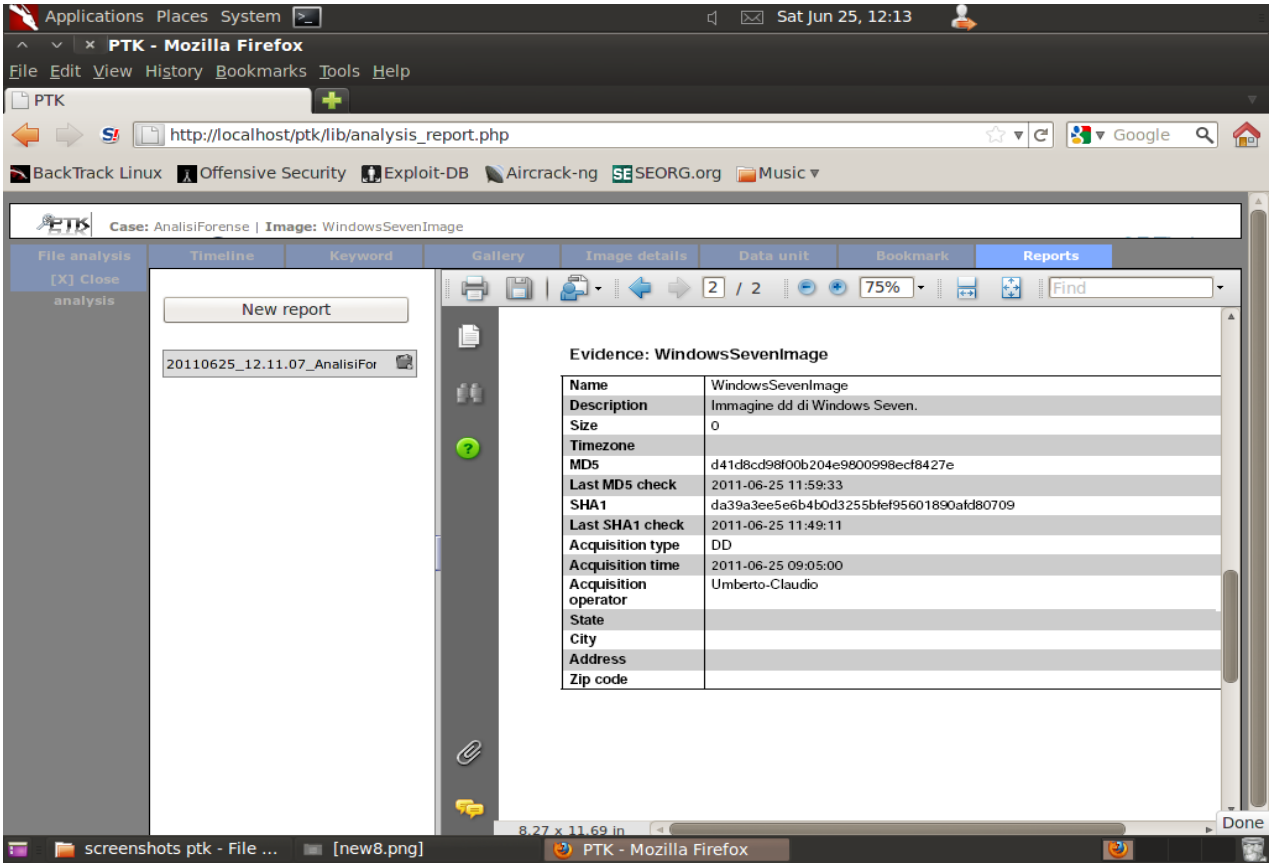


Figura 3.37 – Seconda pagina del report

Ovviamente, come già detto, ptk ci permette di effettuare un image integrity check, mostrato in figura 3.38, sia con md5 sia con sha1.

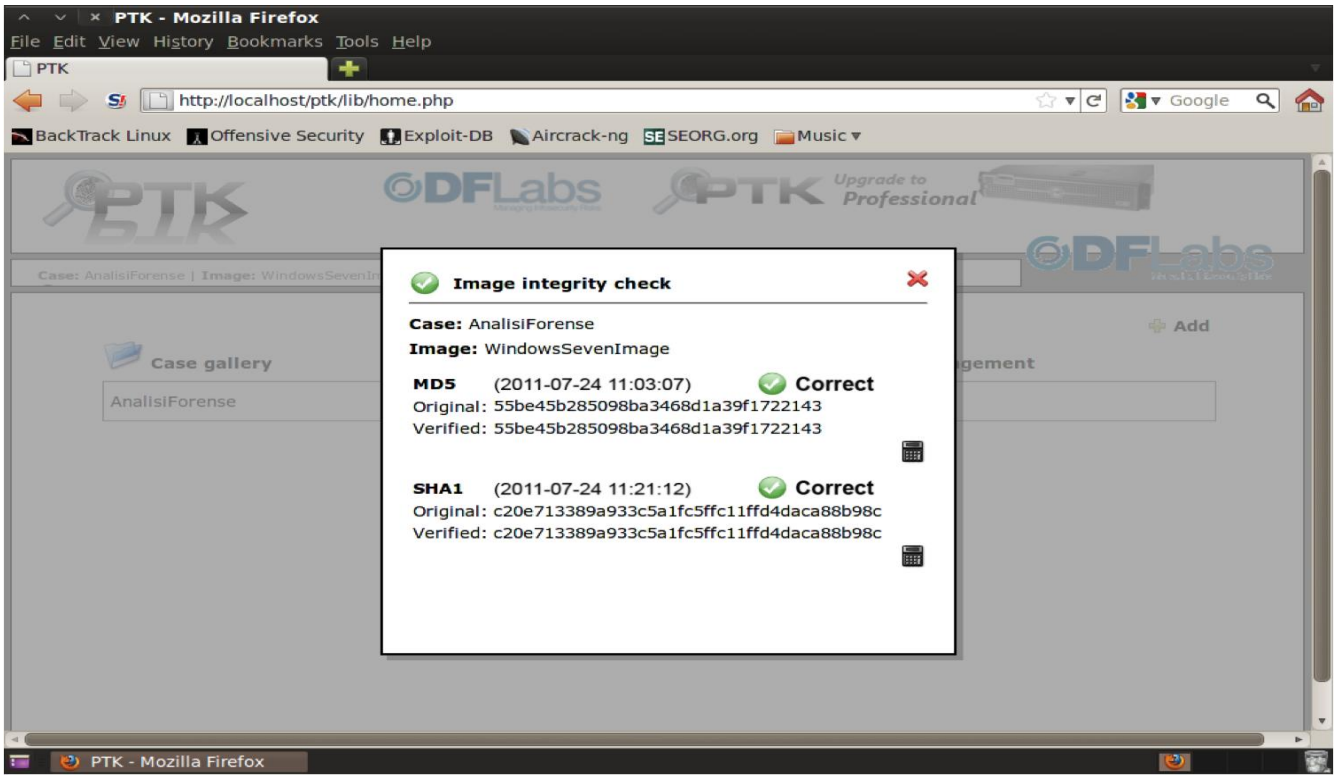


Figura 3.38 – Image integrity check

3.6 Confronto tra Autopsy e Ptk

Dopo aver utilizzato sia Autopsy che Ptk siamo giunti alla conclusione che Ptk offre una interfaccia grafica migliore rispetto ad Autopsy che, invece, ha una interfaccia più spartana. Ptk offre poi un motore di indicizzazione dell'immagine che mappa su database tutte le informazioni dell'evidenza da analizzare in modo da rendere, in fase di analisi, le query molto più veloci. Rispetto ad Autopsy, Ptk ha qualche funzionalità in più come la generazione automatica dei report e la visualizzazione della timeline in formato grafico. Fra le funzionalità presenti in Autopsy e non in ptk si trova il salvataggio dei file in un archivio categorizzati per tipo (visualizzando le extension mismatches).

4. Web Browser Forensic

Nel capitolo precedente sono stati mostrati alcuni tool open source e alcune tecniche che è possibile utilizzare per eseguire analisi forense di basso livello. Tuttavia l'analisi forense post-mortem di un computer richiede un'analisi estremamente approfondita dei dati; essi infatti non vanno visti solo nel loro insieme e nelle loro caratteristiche dettate dal filesystem, ma anche nella loro struttura e nel loro contenuto interno che, chiaramente, è dipendente dall'applicazione che l'ha creato.

Fra le operazioni più importanti dell'analisi di alto livello si trova senz'altro l'analisi dei dati relativi alla navigazione web effettuata dall'utente.

Problema non marginale per l'investigatore è cercare di risalire alle pagine web visitate e ricostruirne l'aspetto della pagina così come visualizzato dall'utente, anche a distanza di molto tempo.

In questo capitolo viene presentata la teoria alla base del file index.dat di Internet Explorer, che rappresenta il cuore del sistema di memorizzazione dei dati relativi ai siti web visitati, ed alcuni esempi pratici per una sua analisi tramite un editor esadecimale. Nel capitolo successivo invece verranno mostrati alcuni tool in grado di recuperare i dati temporanei di navigazione lasciati da Internet Explorer in Windows 7 in modo completamente automatico; infine verrà illustrato un tool in grado di ricostruire graficamente le pagine web così come sono state visualizzate dall'utente al momento della navigazione.

I dati temporanei di Internet Explorer

Internet Explorer conserva i dati temporanei in directory e file di registro; essi sono così organizzati:

- Cache:
  C:\Users\<user name>\AppData\Local\Microsoft\Windows\Temporary Internet Files
- Cookies:
  C:\Users\<user name>\AppData\Roaming\Microsoft\Windows\Cookies
- History:
  C:\Users\<user name>\AppData\Local\Microsoft\Windows\History
- Indirizzi digitati nella URL bar:
  HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\TypedUrls
- Dati dei form (autocomplete forms):

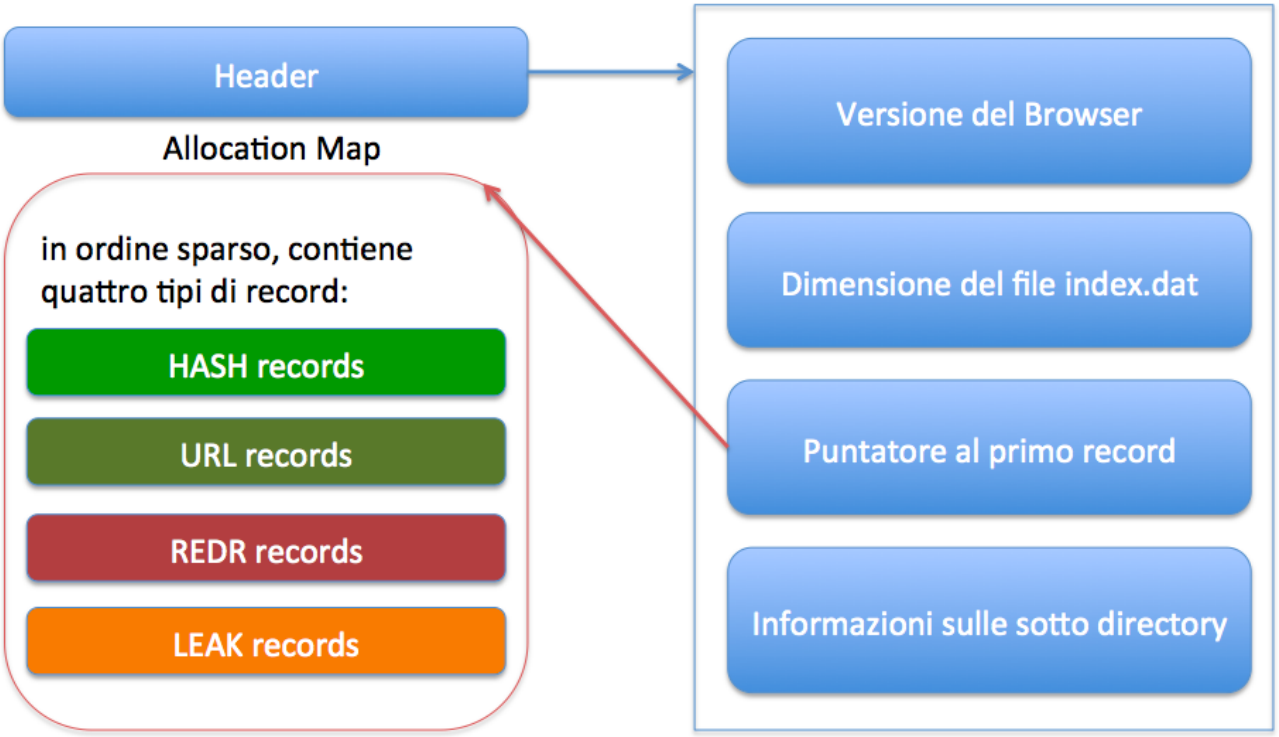
HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\IntelliForms\Storage1

HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\IntelliForms\Storage2

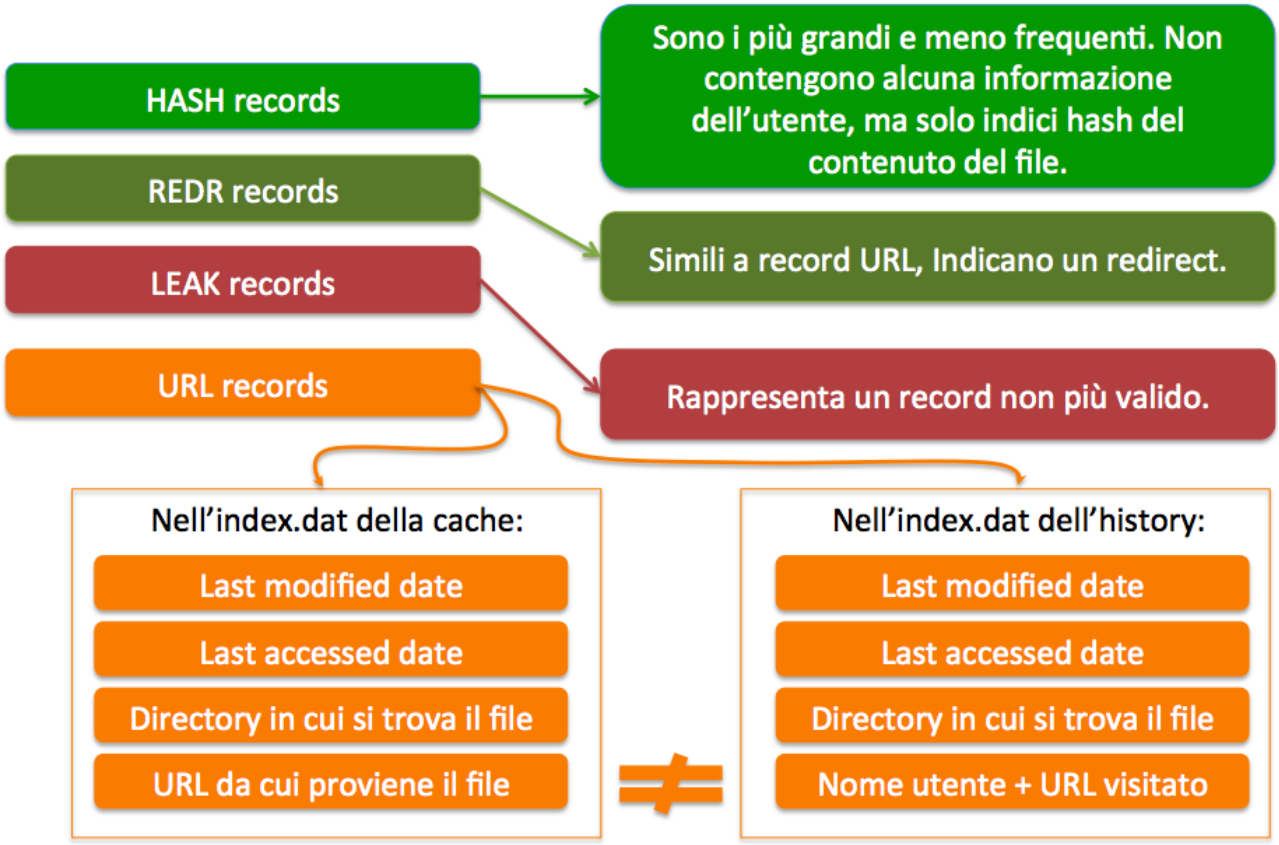
4.1 Struttura dell'index.dat - analisi di alto livello

La directory "Temporary Internet Files" contiene le pagine e le immagini visualizzate dall'utente ed è organizzata in sottodirectory.

Nel seguente schema viene mostrata una struttura di alto livello, indipendente dalla sottodirectory in cui esso è contenuto:



Più nel dettaglio, ogni entry dell'allocation map è strutturata come indicato nella seguente figura:

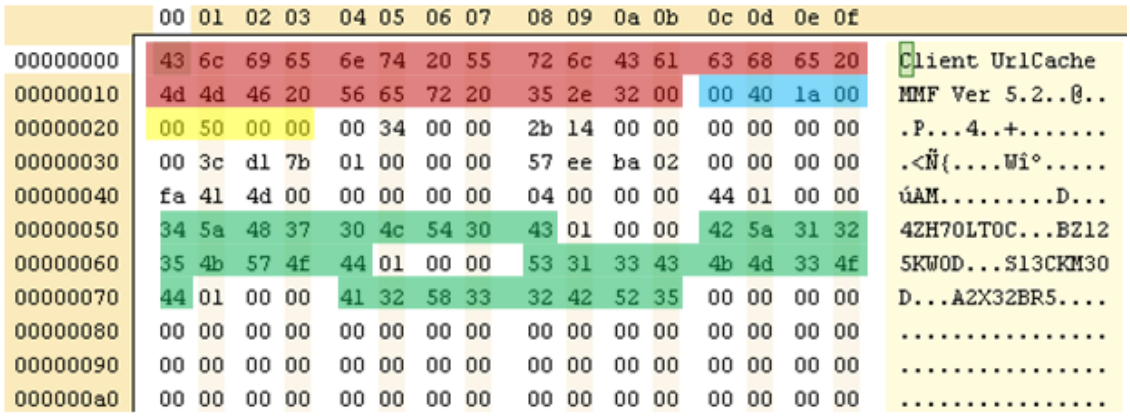


E' possibile notare che se il file index.dat a cui ci si riferisce è estratto dalla cartella di cache e non dalla cartella dei file temporanei, il quarto campo indicato dell'URL record è differente fra history e cache.

4.2 Esempio di lettura dell'index.dat tramite editor esadecimale

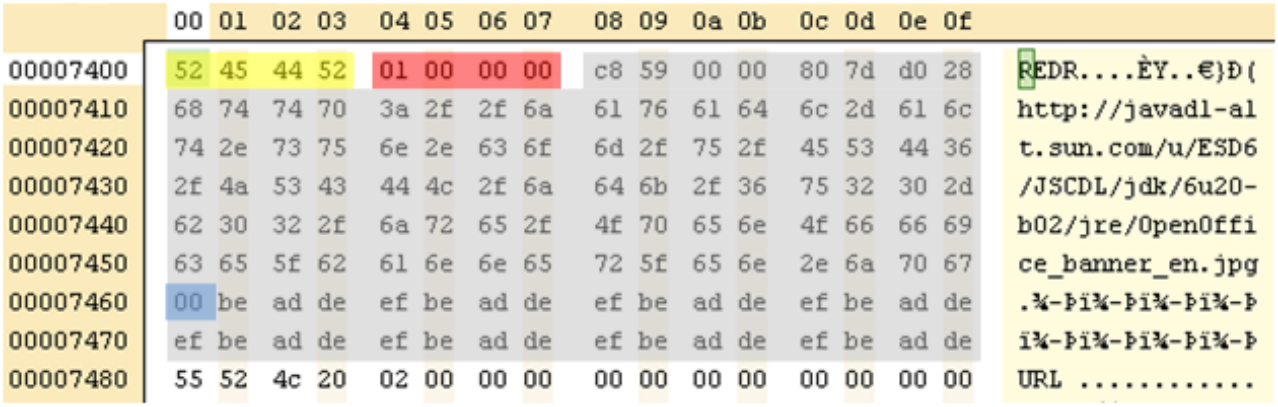
Viene mostrata ora la lettura di un file index.dat tramite editor esadecimale, al fine di far comprendere al meglio la filosofia di analisi dell'index.dat che è alla base degli strumenti che verranno utilizzati e mostrati nel capitolo successivo, sul caso di studio presentato nel capitolo 3.

Nella seguente figura è possibile osservare l'header di un file index.dat



- Il campo evidenziato in rosso indica una stringa standard in ogni file index.dat "Client UrlCache MMF Ver", seguita da una sigla numerica che varia in base alla versione del browser.
- Il campo evidenziato in azzurro indica la dimensione del file, in little endian.
- Il campo evidenziato in giallo è il puntatore al primo record valido dell'allocation map. In prima istanza può sembrare inutile questo campo, ma analizzando nel dettaglio il funzionamento dell'allocation map (nel paragrafo successivo) e delle entry valide e non valide se ne capirà l'utilità.
- Infine, i campi evidenziati in verde indicano i nomi delle sottocartelle utilizzate per contenere i file temporanei.

Nella figura seguente viene evidenziato l'header che caratterizza ogni record dell'allocation map:

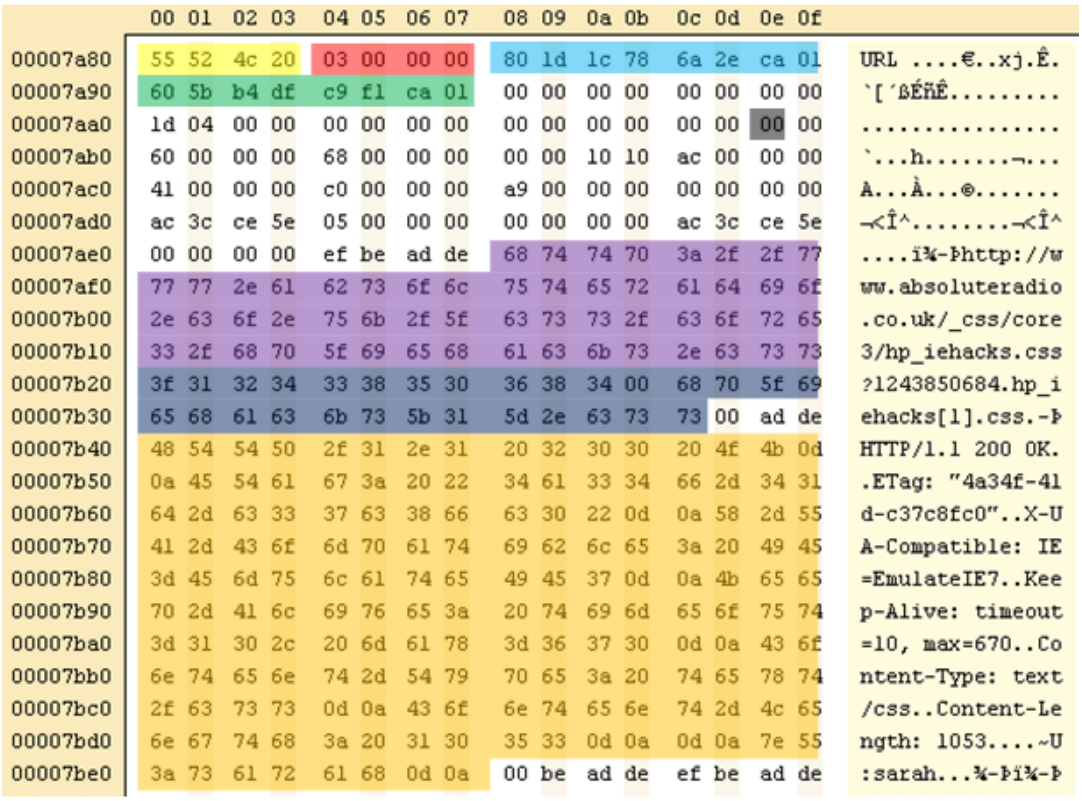


- Il campo evidenziato in giallo indica il tipo di record;
- Il campo evidenziato in rosso indica la lunghezza del record, espressa come numero di blocchi da 128byte;
- Il campo di colore grigio indica i dati (in questa figura generici) del record, terminati da un NULL terminator (evidenziato in celeste);

- Tutti i campi successivi al NULL terminator sono riempiti con dati "junk" e sono necessari per mantenere fissa la lunghezza del record nell'allocation map, per ridurre la frammentazione.

Per ciò che riguarda i dati di ogni singolo record, è necessario presentarli sia nel caso in cui si stia compiendo l'analisi di un file index.dat contenuto nella cartella di cache, sia nel caso in cui esso venga effettuato sull'index.dat della history.

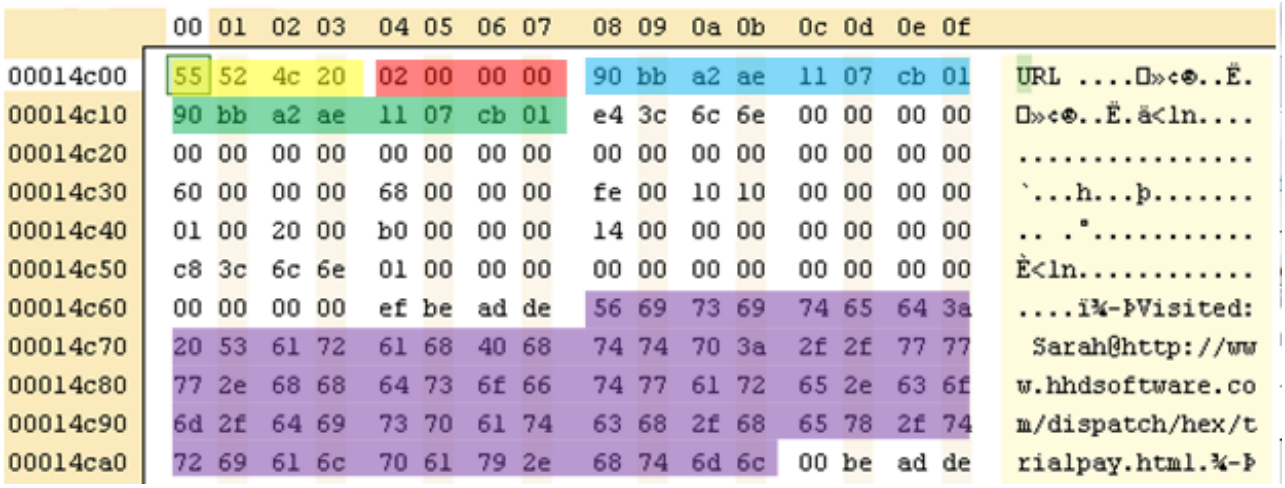
Lettura dei dati di un record dell'index.dat della cache folder



- I campi evidenziati in giallo e rosso sono comuni alla descrizione effettuata precedentemente;
- Il campo di colore celeste indica la data dell'ultima modifica;
- Il campo di colore verde indica la data di ultimo accesso;
- Il campo di colore grigio rappresenta l'indice della sottodirectory nella quale il file è contenuto;
- Il campo di colore viola indica l'URL di origine del file;
- Il campo di colore blu indica il nome dato al file dal browser al momento del salvataggio in cache;
- Il campo di colore arancione rappresenta l'header di risposta HTTP alla richiesta effettuata. Alla fine di questo campo è, inoltre, indicato il nome dell'utente che ha eseguito la richiesta.

Lettura dei dati di un record dell'index.dat della history

Nel caso in cui, invece, si stia effettuando la lettura di un record della history, i campi assumono i seguenti significati:



	00 01 02 03	04 05 06 07	08 09 0a 0b	0c 0d 0e 0f	
00014c00	55 52 4c 20	02 00 00 00	90 bb a2 ae	11 07 cb 01	URL ....□»‹☻..Ë.
00014c10	90 bb a2 ae	11 07 cb 01	e4 3c 6c 6e	00 00 00 00	□»‹☻..Ë.ä<ln....
00014c20	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	....
00014c30	60 00 00 00	68 00 00 00	fe 00 10 10	00 00 00 00	`...h...þ.......
00014c40	01 00 20 00	b0 00 00 00	14 00 00 00	00 00 00 00	.. .°...........
00014c50	c8 3c 6c 6e	01 00 00 00	00 00 00 00	00 00 00 00	È<ln............
00014c60	00 00 00 00	ef be ad de	56 69 73 69	74 65 64 3a	....ï¾-ÞVisited:
00014c70	20 53 61 72	61 68 40 68	74 74 70 3a	2f 2f 77 77	Sarah@http://ww
00014c80	77 2e 68 68	64 73 6f 66	74 77 61 72	65 2e 63 6f	w.hhdsoftware.co
00014c90	6d 2f 64 69	73 70 61 74	63 68 2f 68	65 78 2f 74	m/dispatch/hex/t
00014ca0	72 69 61 6c	70 61 79 2e	68 74 6d 6c	00 be ad de	rialpay.html.¾-Þ

I primi quattro campi sono identici al record illustrato in precedenza, mentre il campo di colore viola rappresenta l'url visitato, preceduto dal nome dell'utente che ha eseguito la richiesta dell'url stesso.

Nel paragrafo successivo vengono mostrati i dettagli tecnici del file index.dat: essi rappresentano un utile punto di partenza per il lettore che ha intenzione di eseguire manualmente l'analisi di un file index.dat senza utilizzare strumenti più "user friendly". Ai fini della totale comprensione dei successivi capitoli, tuttavia, la lettura di tali dettagli non è necessaria.

4.3 Struttura dell'index.dat - analisi di basso livello

Il contenuto della cache è tracciato nel file archivio index.dat: esso è un file binario avente un header di 0x4000bytes, seguito da record successivi di 0x80bytes.

Nel dettaglio, l'header all'offset 0x28 contiene il numero di blocchi allocati; tale blocco ha dimensioni limitate (0x0001ED80 bytes) e ciò limita la dimensione massima del file index.dat a poco meno di 15.4MB. Per superare tale limite nei blocchi 0x48 e 0x4C vengono definite delle cartelle aggiuntive (fino ad un massimo di 32 cartelle) in cui a loro volta è presente un file index.dat allo scopo di indicizzare i dati della sottocartella; nel file index.dat principale, per ogni sottocartella, è indicato il nome ed il numero di files in essa presenti.

Viene riportato di seguito lo schema dell'header del file index.dat:

Offset	Size	Description
0x00	0x1C bytes	signature, necessarily “Client UrlCache MMF Ver 5.2”, including null terminator
0x1C	dword	file size, in bytes
0x20	dword	file offset of first page in hash table, else zero
0x24	dword	total number of blocks following header
0x28	dword	number of allocated blocks
0x2C	4 bytes	apparently unused
0x30	qword	cache limit, in bytes
0x38	qword	cache size, in bytes
0x40	qword	cache usage exempt from scavenging, in bytes
0x48	dword	number of subdirectories in cache
0x4C	0x0180 bytes	array of 0x20 structures, each of 0x0C bytes, to describe subdirectories in cache
0x01CC	0x80 bytes	array of 0x20 dwords, apparently called header data
0x024C	4 bytes	apparently unused
0x0250	0x3DB0 bytes	allocation bitmap for blocks following header

Figura 4.1

4.3.1 Le File Map Entries

Ogni entry dell'allocation map ha un blocco iniziale di otto byte che contiene:

Offset	Size	Description
0x00	dword	signature
0x04	dword	number of blocks allocated to entry

Figura 4.2

In coda ad ogni header è presente il contenuto vero e proprio dell'entry; ciò implica che l'hash table è formata da terne del tipo (TYPE, LENGTH, DATA).

La firma può assumere quattro distinti valori:

"HASH"	0x48534148	page in the hash table
"LEAK"	0x4B41454C	leak entry, actually a modified URL entry
"REDR"	0x52444552	redirection entry
"URL "	0x204C5255	standard URL entry

Figura 4.3

• HASH: indica che l'entry è una hash table (per concatenare hash table differenti);

• LEAK: indica che l'entry referenziata o è stata cancellata o modificata;

• REDR: l'entry rappresenta un redirect;

• URL: l'entry referenzia un URL visitato.

In base al tipo (firma) dell'entry si distinguono strutture dati differenti, delimitate in tutti i casi dalla lunghezza riportata nell'header (campo all'indirizzo 0x04).

4.4 Funzionamento del caching

Quando viene visitata una pagina web il suo contenuto viene ricercato nella cache e, se non trovato, inserito in essa. Per effettuare un recupero/salvataggio efficace delle risorse viene dapprima calcolato un hash a 32bit sull'URL della pagina ma solo i 26 bit più significativi di esso vengono utilizzati come identificativo dell'entry nella tabella. Oltre all'identificativo dell'URL di 26 bit, l'entry dell'hash table ha altri campi riportati nella struttura in Figura 4.4:

Offset	Size	Description
0x00	5 bits	flags
0x00	1 bit	apparently unused
0x00	26 bits	high 26 bits of hash
0x04	dword	file offset of corresponding file-map entry, else 3

Figura 4.4

Al suo interno, il campo flags può assumere i seguenti valori:

0x01	clear: file offset in hash item is of URL entry and hash is of URL
0x02	corresponding URL entry is locked
0x04	corresponding URL entry has trivial redirection

Figura 4.5

Con il flag "URL entry locked" si indica che il contenuto relativo all'URL è in fase di lettura.

Le letture vengono tracciate utilizzando un contatore ricorsivo, ossia incrementato (decrementato) su ogni risorsa ottenuta accedendo all'URL e presente in cache.

Nel caso in cui il flag 0x01 risulti impostato allora i bit 0x01, 0x03 e 0x05 assumono i seguenti significati:

0x01	hash item is free; whole first dword of hash item should be 1
0x03	hash item is unused; whole first dword of hash item should be 3
0x05	file offset is of redirection entry; hash is of original URL

Figura 4.6

Essendo la ricerca effettuata su una linked list, nel caso in cui viene letta una entry con il flag 0x03 settato allora tale entry è l'ultima della lista e non ve ne sono altre presenti; se l'entry desiderata non è stata trovata in precedenza allora viene inserita nella prima entry avente il flag 0x01 impostato (se presente). Nel caso in cui durante la scansione non siano state trovate entry libere, viene creata una nuova entry in coda alla lista, ossia all'entry avente il flag 0x03 settato.

I dati dell'entry relativa ad una URL sono di dimensione fissa (ma variabili in numero) e sono i seguenti:

Offset	Size	Description	
0x08	8 bytes	last modified time, as **FILETIME** structure	
0x10	8 bytes	last access time, as **FILETIME** structure	
0x18	dword	expiry time, as DOS time	
0x1C	dword	potted description needed here!	
0x20	dword	size of local file, in bytes	
0x24	dword	apparently unused, except for explicit initialisation to zero	
0x28	dword	file offset of **GROUP_ENTRY** or **LIST_GROUP_ENTRY**	
0x2C	dword	in URL entry:	exempt delta
		in leak entry:	file offset of next leak entry
0x30	dword	size of structure in excess of **FILEMAP_ENTRY**, in bytes	
0x34	dword	offset from start of structure to URL, as saved in entry after header	
0x38	byte	index of directory containing local file	
0x39	byte	synchronisation count	
0x3A	byte	potted description needed here!	
0x3B	byte	potted description needed here!	
0x3C	dword	offset from start of structure to name of local file, as saved in entry after header	
0x40	dword	cache entry type, as bit flags	
0x44	dword	offset from start of structure to header information, as saved in entry after header	
0x48	dword	size of header information, in bytes	
0x4C	dword	offset from start of structure to file extension, as saved in entry after header	
0x50	dword	last synchronisation time, as DOS time	
0x54	dword	number of times entry has been locked	
0x58	dword	nesting level of locks on entry	
0x5C	dword	creation time, as DOS time	

Figura 4.7

I campi marcati in rosso sono refusi di versioni precedenti della struttura e mantenuti per una questione di compatibilità anche nelle più recenti versioni di Internet

Explorer.
Il tipo di cache entry (mappato all'indirizzo 0x40) può assumere i seguenti valori:

0x00000001	**NORMAL_CACHE_ENTRY**	set initially for all entries in Content container
0x00000004	**STICKY_CACHE_ENTRY**	entry is exempt from scavenging
0x00000008	**EDITED_CACHE_ENTRY**	local file need not be in cache
0x00010000	**SPARSE_CACHE_ENTRY**	potted description needed here!
0x00100000	**COOKIE_CACHE_ENTRY**	set initially for all entries in Cookies container
0x00200000	**URLHISTORY_CACHE_ENTRY**	set initially for all entries in History container
0x00400000	**PENDING_DELETE_CACHE_ENTRY**	set when deletion is attempted while entry is locked
0x10000000	**INSTALLED_CACHE_ENTRY**	
0x80000000	**IDENTITY_CACHE_ENTRY**	

Inutilizzate e non documentate nelle nuove versioni di Internet Explorer

Figura 4.8

Dalla teoria appena esposta si è visto come, attraverso un'analisi dei file index.dat, sia possibile recuperare le informazioni relative alla navigazione effettuata da un utente tramite Internet Explorer.

Tuttavia questa analisi per essere eseguita manualmente richiede grande dimestichezza con la lettura di file binari (attraverso hex editors) e, in ogni caso, è un'operazione estremamente lunga. Per facilitare il compito dell'investigatore sono stati sviluppati vari tool che automatizzano l'analisi ed il reporting del file index.dat: nella parte finale di questo documento vengono presentati i più utilizzati in ambito open source.

4.5 Analisi di Internet Explorer attraverso 3 tool

In questo capitolo vengono presentati i tool che consentono di leggere e recuperare le informazioni contenute nel file index.dat. Essi sono stati testati sul caso di studio presentato nel capitolo 3.

I tool analizzati sono i seguenti:

- Pasco
- Web historian
- NetAnalysis v1.52

Pasco è un software free, lo abbiamo usato su Linux, sebbene esista anche la versione per Windows; anche Web Historian è free e può essere utilizzato solo su Windows; NetAnalysis viene dato in evaluation per 30 giorni e può essere usato su Windows. I primi due tool si limitano ad effettuare una analisi della cache del browser, mentre il terzo, oltre ad analizzare la cache, effettua anche la ricostruzione automatica delle pagine visitate, così come le ha viste l'utente.

4.5.1 Pasco

L'installazione di Pasco in Linux può essere fatta in modo semplice ed immediato, lanciando da terminale

sudo apt-get install pasco

Una volta installato, basta posizionarsi da terminale nella cartella che contiene l'index.dat e lanciare il comando

pasco index.dat > index.csv

il quale va a creare nella cartella in cui è stato lanciato il file index.csv, così come riportiamo in figura 5.1.

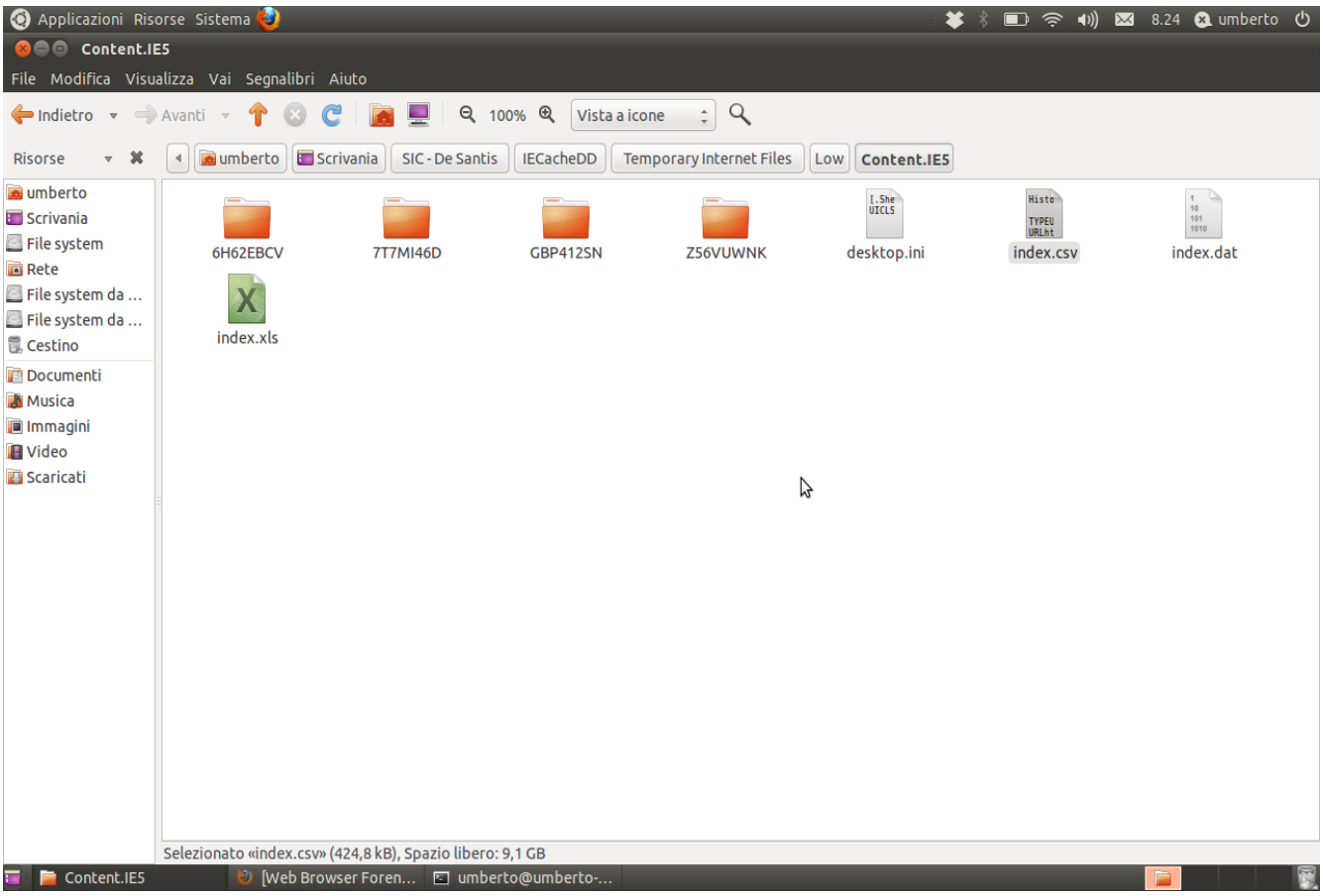


Figura 5.1 – Cartella che contiene l'index.dat

In figura 5.2 mostriamo il contenuto del file index.csv.

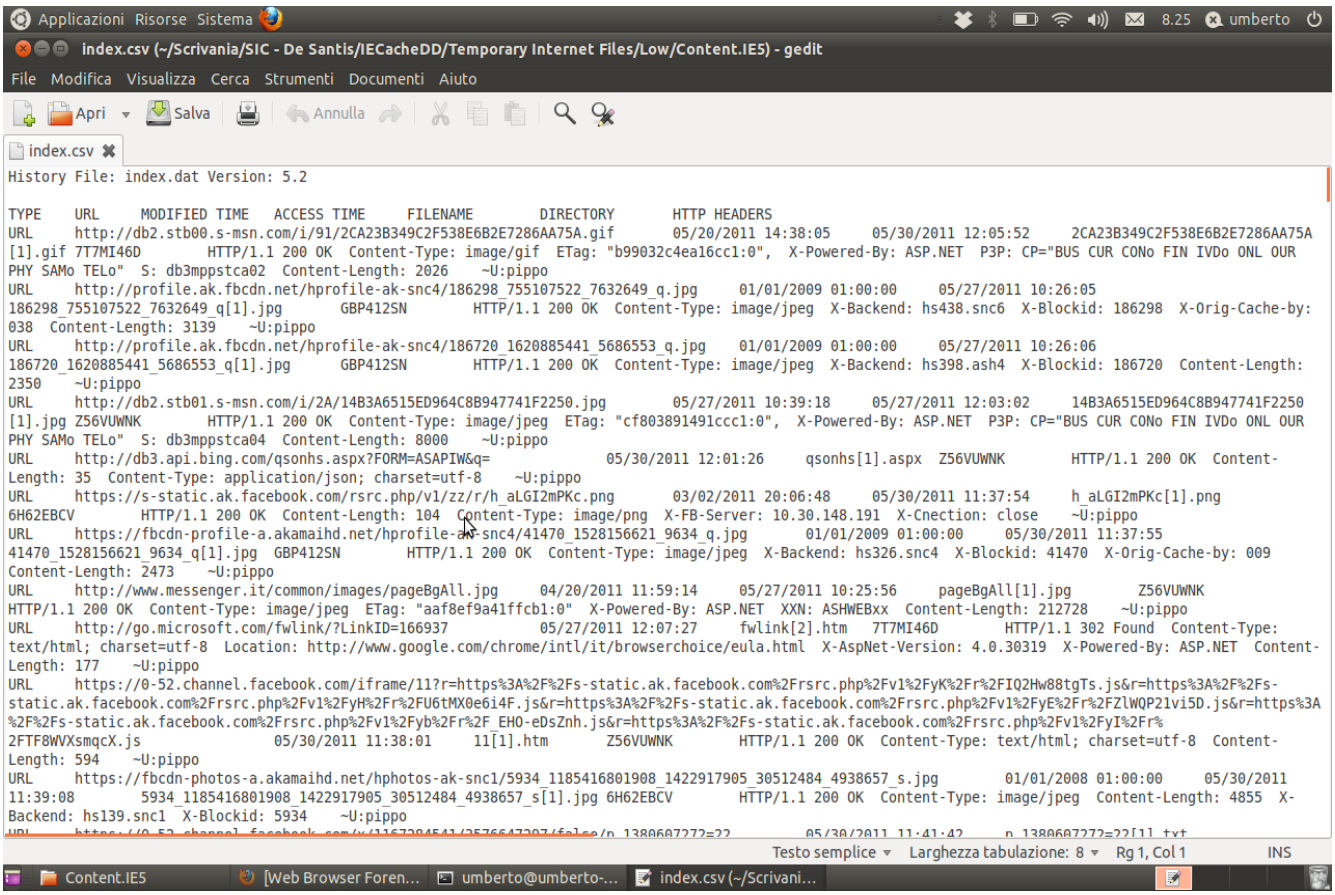


Figura 5.2 – Contenuto del file index.csv

Notiamo che il contenuto viene mostrato in formato tabellare: per ogni voce trovata viene indicato il tipo (url o pagina di redirezione), l'url, data ed ora di modifica, data ed ora di accesso, il nome del file, la directory che contiene il file, gli headers http. Se, ad esempio, facciamo riferimento alla prima voce, possiamo capire che l'URL di tipo TYPE è stato acceduto al tempo ACCESS TIME, e ad esso è associato il file di nome FILENAME che è presente nella cartella DIRECTORY.

4.5.2 Web Historian

Il file di installazione di Web Historian può essere scaricato alla pagina web http://www.mandiant.com/products/free_software/web_historian/download .

Una volta installato ed avviato, basta cliccare su "scan" sotto la voce "File" come da figura 5.3.

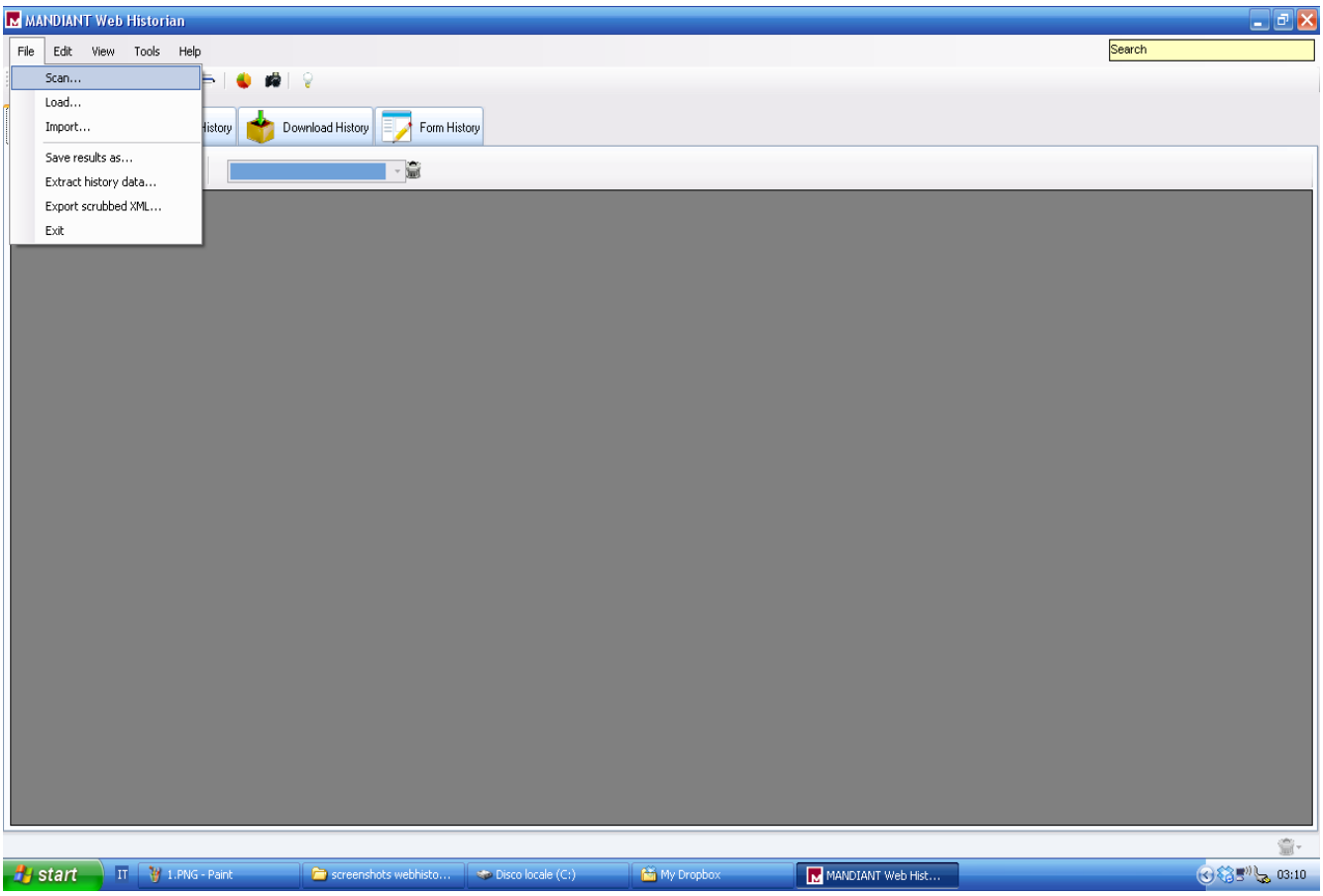


Figura 5.3

Bisogna poi indicare il path dell'index.dat alla voce "History file", come da figura 5.4.

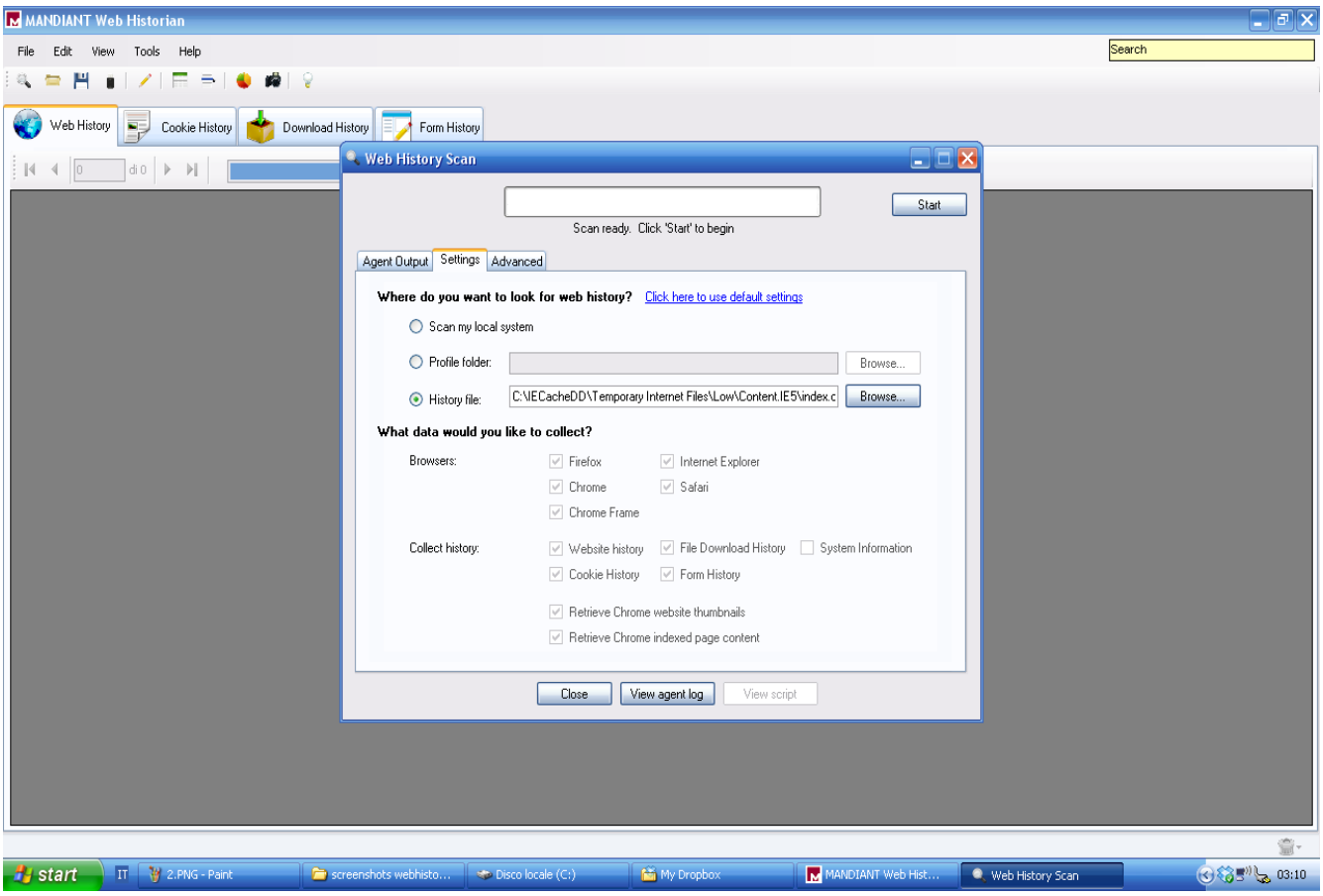


Figura 5.5 – Path dell'index.dat

Quindi, dopo aver cliccato su Start, viene fatta la scansione dell'index.dat indicato e ci viene data in output la history della pagine visitate in formato tabellare, come da figura 5.5.

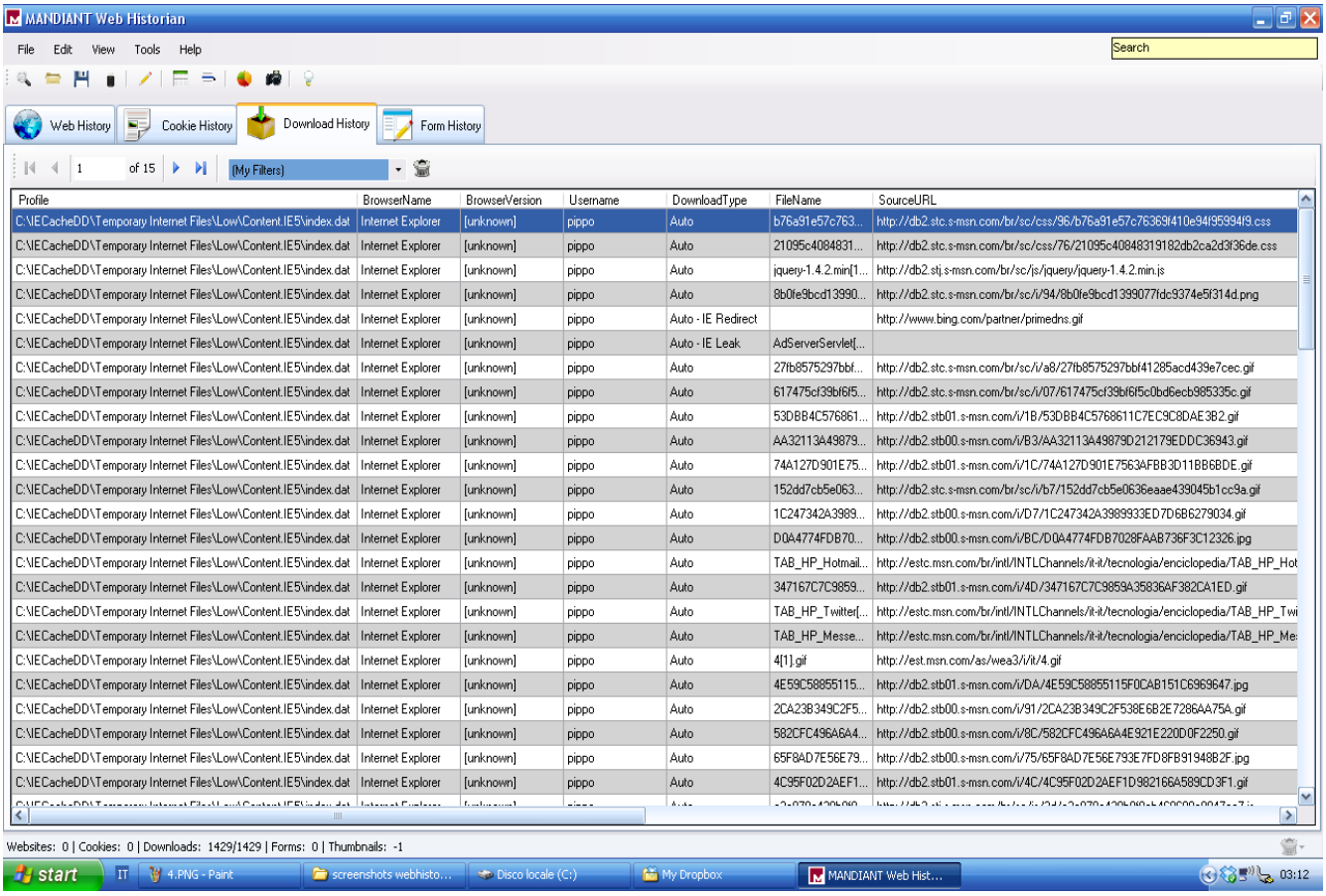
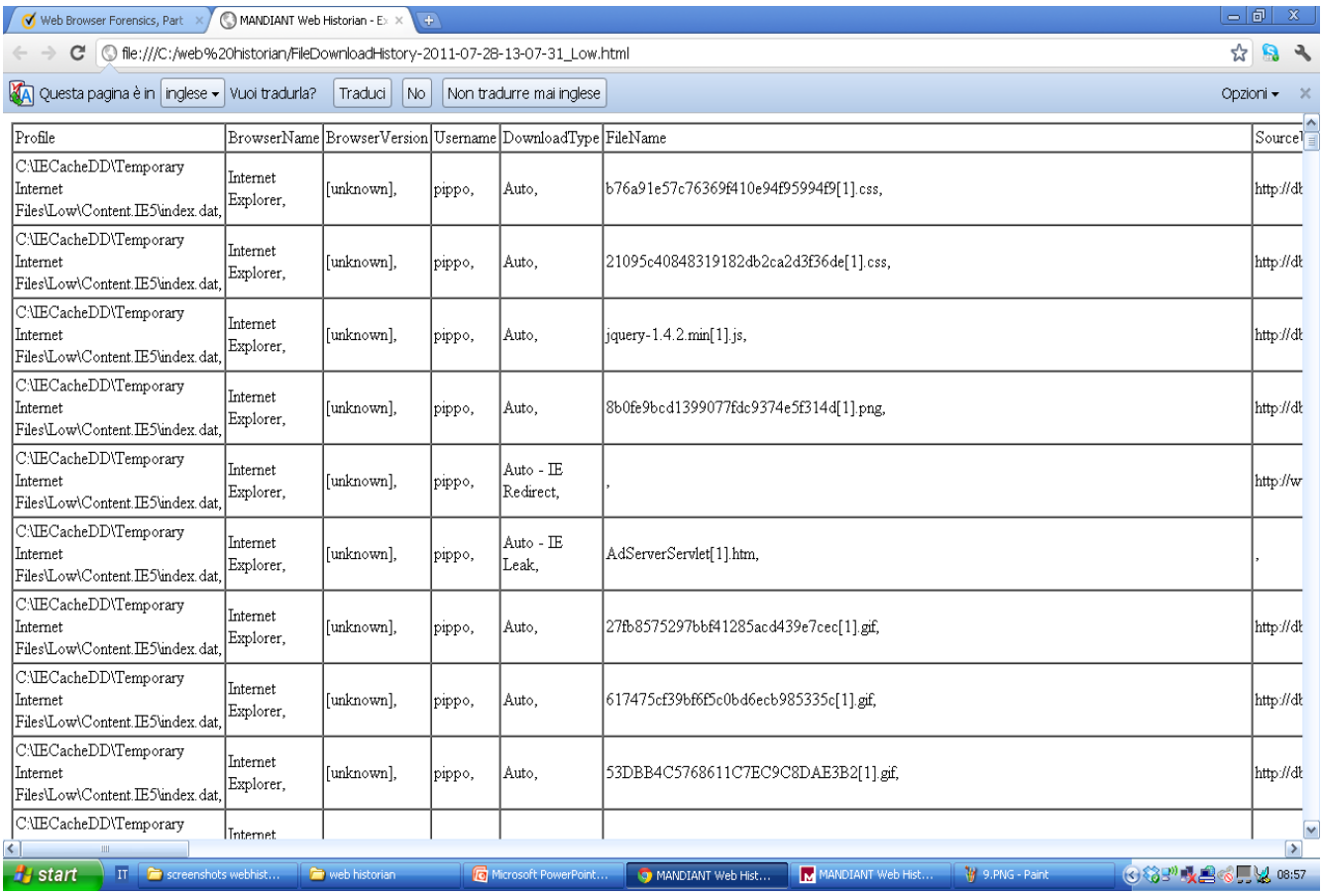


Figura 5.6 – History pagine web

È possibile, quindi, scorrere la tabella per poterla consultare, oppure esportarla in un altro formato (xml, csv o html). Per farlo, basta cliccare su “Save results as...” sotto la voce “File”, scegliere poi il formato di output e la cartella di output e cliccare su “Export”. In figura 5.6 mostriamo la tabella esportata in formato html.



Profile	BrowserName	BrowserVersion	Username	DownloadType	FileName	Source
C:\IECacheDD\Temporary Internet Files\Low\Content.IE5\index.dat,	Internet Explorer,	[unknown],	pippo,	Auto,	b76a91e57c76369f410e94f95994f9[1].css,	http://d
C:\IECacheDD\Temporary Internet Files\Low\Content.IE5\index.dat,	Internet Explorer,	[unknown],	pippo,	Auto,	21095c40848319182db2ca2d3f36de[1].css,	http://d
C:\IECacheDD\Temporary Internet Files\Low\Content.IE5\index.dat,	Internet Explorer,	[unknown],	pippo,	Auto,	jquery-1.4.2.min[1].js,	http://d
C:\IECacheDD\Temporary Internet Files\Low\Content.IE5\index.dat,	Internet Explorer,	[unknown],	pippo,	Auto,	8b0fe9bcd1399077fdc9374e5f314d[1].png,	http://d
C:\IECacheDD\Temporary Internet Files\Low\Content.IE5\index.dat,	Internet Explorer,	[unknown],	pippo,	Auto - IE Redirect,	,	http://w
C:\IECacheDD\Temporary Internet Files\Low\Content.IE5\index.dat,	Internet Explorer,	[unknown],	pippo,	Auto - IE Leak,	AdServerServlet[1].htm,	,
C:\IECacheDD\Temporary Internet Files\Low\Content.IE5\index.dat,	Internet Explorer,	[unknown],	pippo,	Auto,	27fb8575297bbf41285acd439e7cec[1].gif,	http://d
C:\IECacheDD\Temporary Internet Files\Low\Content.IE5\index.dat,	Internet Explorer,	[unknown],	pippo,	Auto,	617475cf39bf6f5c0bd6ecb985335c[1].gif,	http://d
C:\IECacheDD\Temporary Internet Files\Low\Content.IE5\index.dat,	Internet Explorer,	[unknown],	pippo,	Auto,	53DBB4C5768611C7EC9C8DAE3B2[1].gif,	http://d
C:\IECacheDD\Temporary	Internet					

Figura 5.7 – History in formato html

4.5.3 NetAnalysis v1.52

Il file di installazione di NetAnalysis v1.52 può essere scaricato all'indirizzo web http://www.digital-detective.co.uk/downloads.asp.

Una volta che il software è stato installato ed avviato, basta cliccare su "Open all history from folder..." sotto la voce "File", settare il path dell'index.dat e premere OK. Dopo aver effettuato la scansione, ci viene mostrato a video in forma tabellare la history delle pagine web visitate relativa all'index.dat indicato (Figura 5.7).

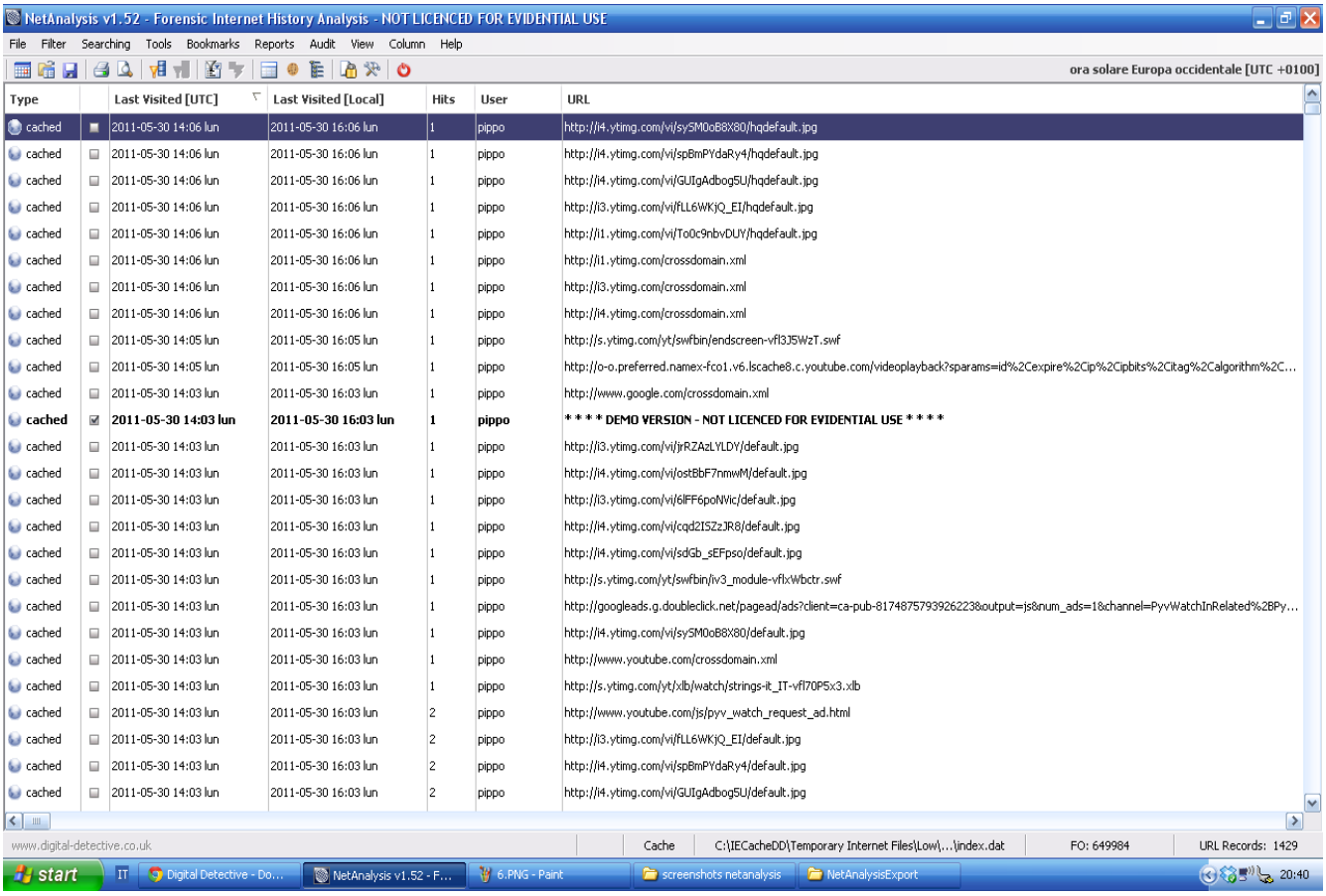


Figura 5.8 – History in formato tabellare

La tabella indica, per ogni voce: il tipo, data ed ora di Greenwich di ultima visita, data ed ora locale di ultima visita (che è quella che ci interessa), il numero di hits (numero che aumenta ad ogni visita dell'url, ma con una formula non indicata da Microsoft), l'utente, l'URL. Sono presenti anche altre colonne che non sono state mostrate in figura 6 e sono: l' host, il path assoluto del file, la cartella in cui risiede il file, il nome del file, il tipo di estensione del file, il response http, il path dell'index.dat associato, la versione del browser.

È possibile, a questo punto, effettuare la ricostruzione automatica delle pagine web visitate a partire dalla history. Per farlo, basta cliccare su "Export/rebuild all cached items..." sotto la voce "Tool"; indicare poi la cartella di output e premere OK. A procedura completa, nella cartella di output indicata in precedenza possiamo notare 4 cartelle, come in figura 5.8.

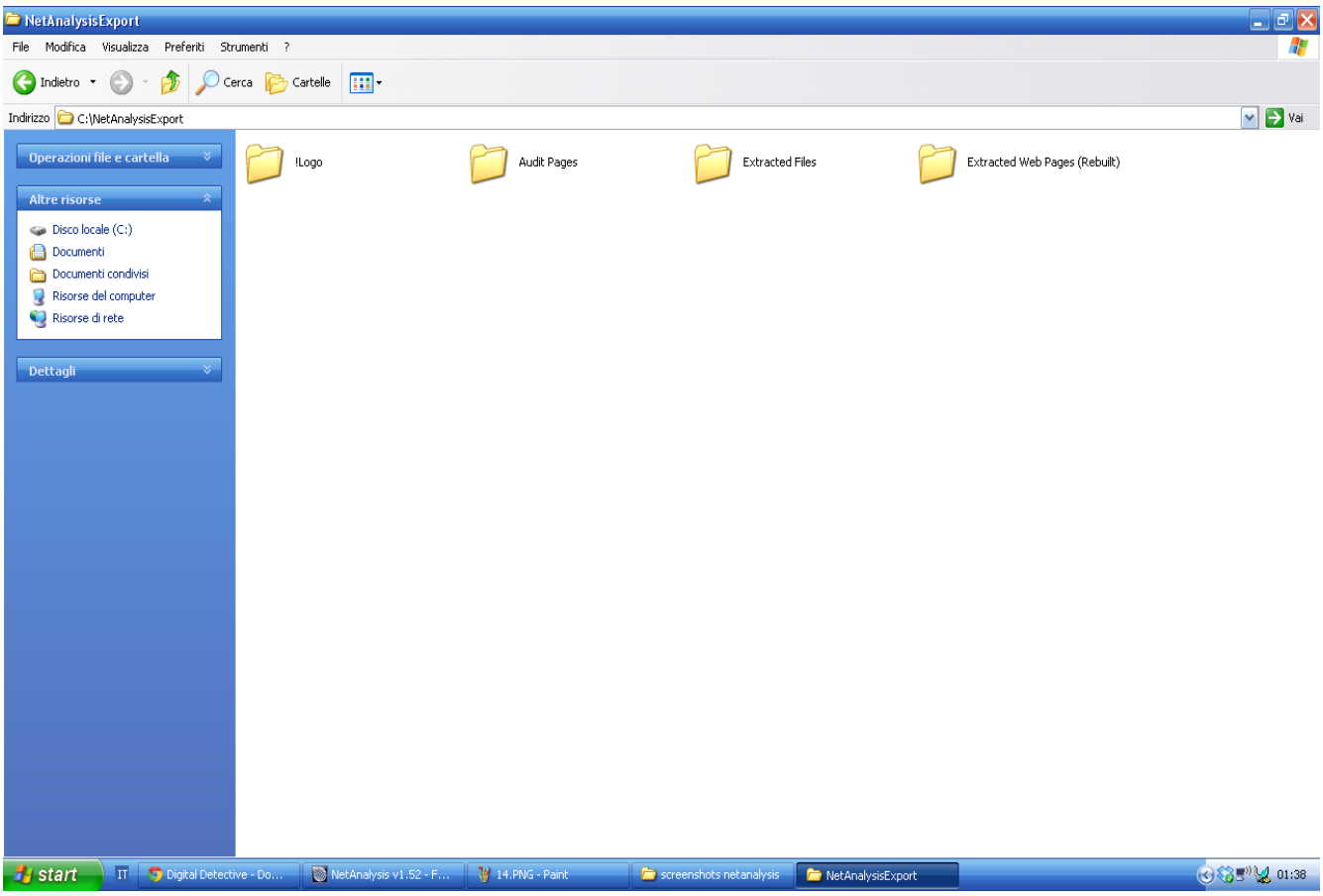


Figura 5.9 – Cartella di output

La cartella “Extracted Web Pages” contiene le pagine html ricostruite; la cartella “Extracted Files” contiene i file che vengono usati dalle pagine web ricostruite; la cartella “Audit Pages” contiene anch’essa delle pagine html, una per ogni pagina web ricostruita, e visualizza una serie di informazioni, tra le quali: il link alla pagina ricostruita (presente in “Extracted Web Pages”), l’URL della pagina, le URL dei file associati alla pagina visitata e ,per ognuno di questi, se ancora è presente o meno in cache. Mostriamo in figura 5.10 una pagina html presente nella cartella “Audit Pages”.

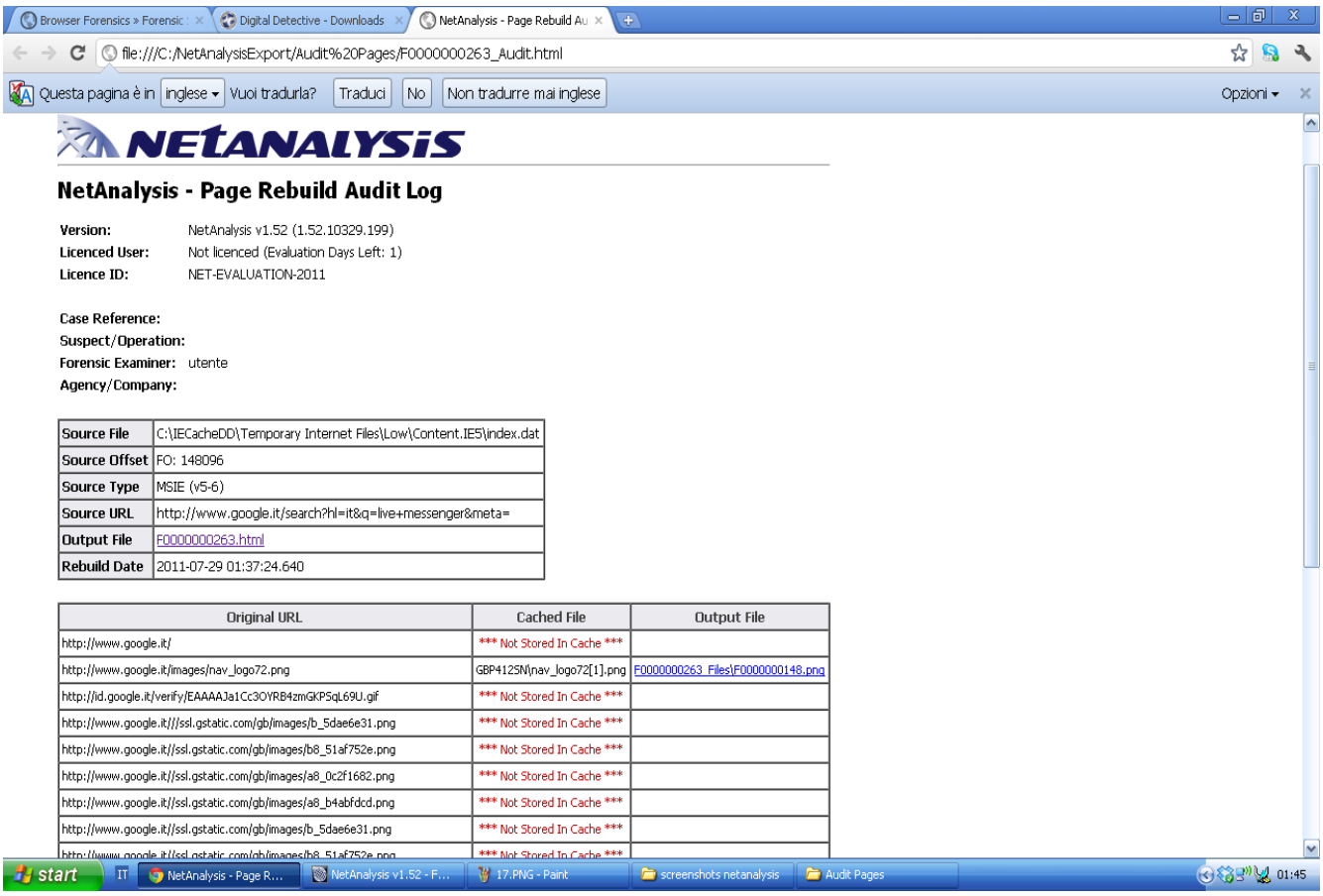


Figura 5.10 – Una pagina html in "Audit Pages"

È possibile quindi fare riferimento al Source URL indicato in figura 5.10 e cercarlo nella history nella colonna URL (vedere figura 5.11).

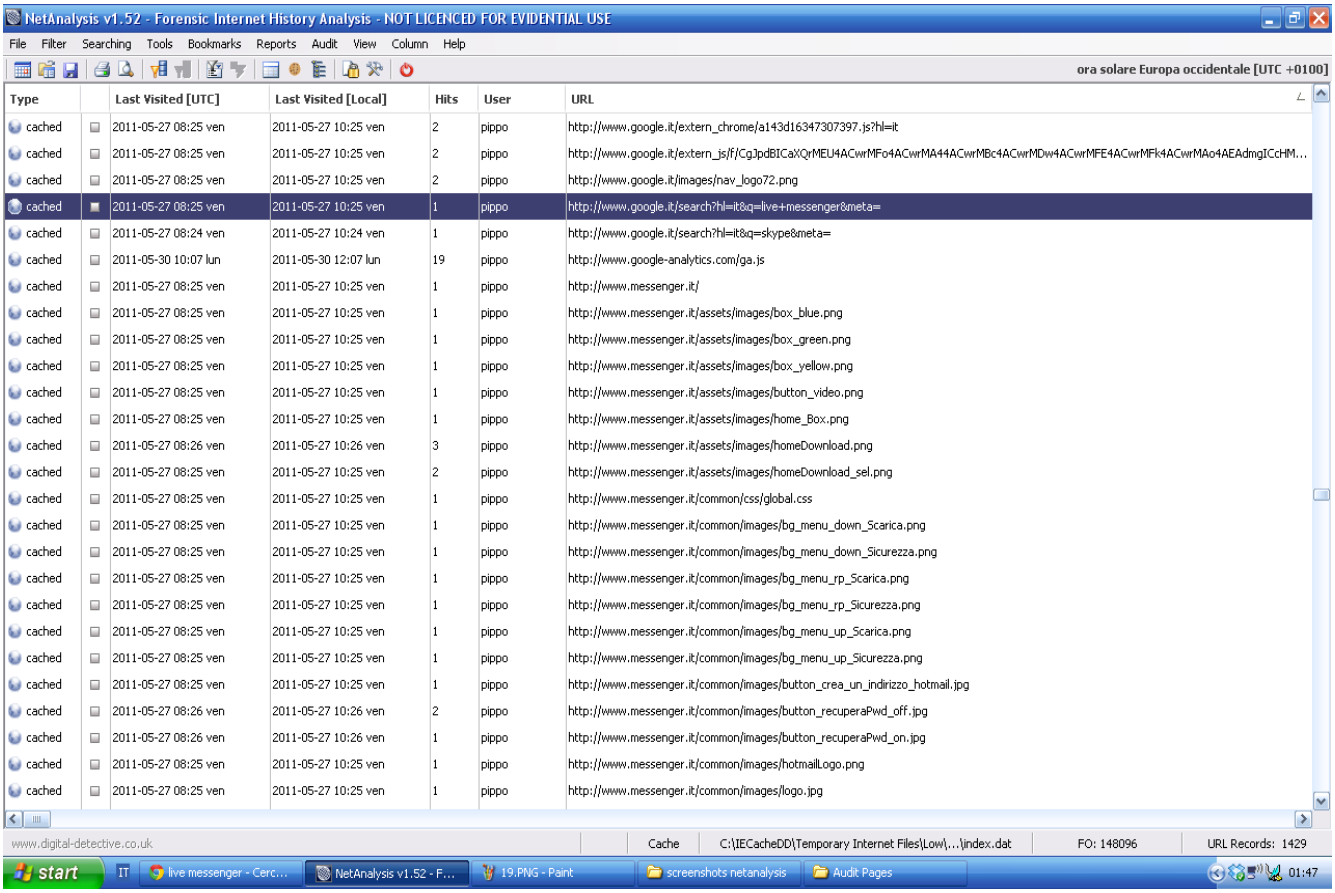


Figura 5.10 – Ricerca manuale dell'URL nella history

In questo modo possiamo capire che la pagina ricostruita F0000000263.html (mostrata in figura 5.11) è stata visitata alle ore 10:25 di venerdì 27-05-2011 dall'utente pippo. Più precisamente a quella data ed ora l'utente pippo ha cercato la stringa "live messenger" su Google.

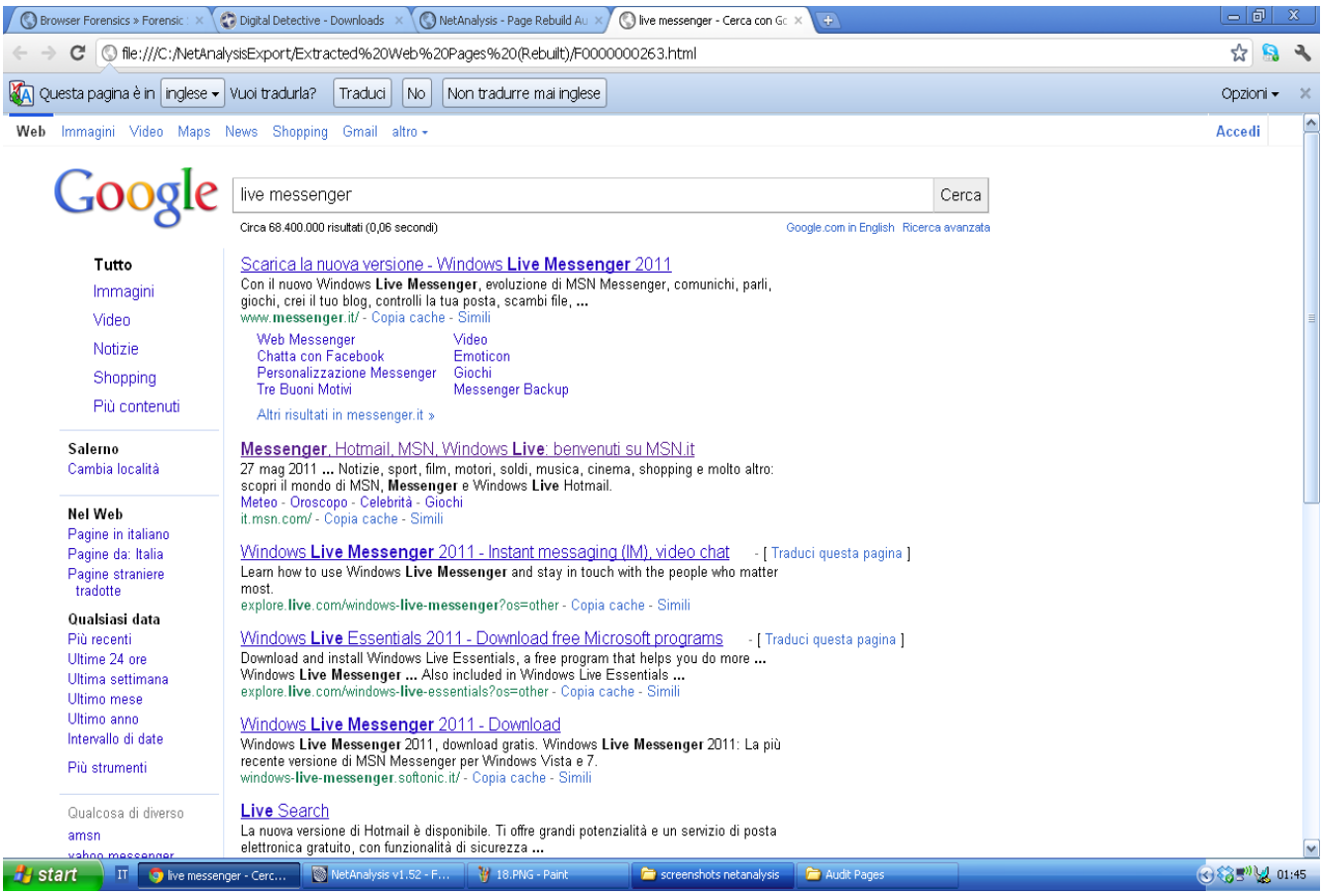


Figura 5.11 – Pagina web ricostruita

5 Riassunto del lavoro svolto

Facciamo ora un riepilogo del lavoro svolto. Inizialmente è stata fatta un'analisi generale delle varie distribuzioni Linux (la Caine, la Deft, la Backtrack e la Helix) valutando i vantaggi e gli svantaggi del loro utilizzo rispetto alle soluzioni proprietarie. È stata poi fatta una lista dei tool presenti, suddividendoli in categorie: acquisizione, analisi, cracking, reporting. Per ognuna di queste categorie abbiamo poi indicato i tool più significativi ed efficaci. Abbiamo creato un caso di studio effettuando diverse operazioni su una partizione con sistema operativo Windows 7. Le operazioni effettuate sono state: download e modifica di file, navigazione sul web, chat e chiamata su Skype. Abbiamo poi acquisito la partizione con le varie distribuzioni, confrontando i tempi e facendo varie considerazioni sul dd da macchina virtuale. Abbiamo scelto ed esposto nei dettagli il funzionamento di due tool che, a nostro avviso, sono i più utili ed efficaci (Autopsy e PTK) mettendo in evidenza le differenze tra i due. Infine abbiamo misurato i tempi di acquisizione dell'evidenza da Live CD, confrontandoli con i tempi impiegati dalla distribuzione locale. Abbiamo visto nel dettaglio la struttura della cache di Internet Explorer e del file index.dat; sulla base di questo studio abbiamo infine fatto una analisi dei dati temporanei di navigazione presenti nel nostro caso di studio con tre tool (Pasco, Web Historian, NetAnalysis) e ricostruito le pagine web così come le ha visualizzate l'utente con l'ultimo dei tool tool elencati.

6 Conclusioni

Il lavoro svolto in questo semestre per il corso di Sicurezza II è stato molto interessante e formativo, in quanto non eravamo a conoscenza di cosa fosse un'analisi forense e ci ha quindi consentito di imparare ad acquisire una evidenza da analizzare e ad usare i tool appropriati per poter effettuare l'analisi. Molto interessante è stato lo studio della struttura della cache del browser e la ricostruzione delle pagine web visitate dall'utente. Inoltre la modalità utilizzata, che è stata diversa da quella tradizionale in quanto non ha previsto una prova scritta ed una orale, ma la preparazione e l'esposizione orale del lavoro eseguito di volta in volta attraverso delle presentazioni con slides, è stata molto formativa in quanto ha migliorato la nostra capacità di parlare e di trasmettere dei concetti acquisiti ad una "platea" di persone.

7 Riferimenti bibliografici

[1] (E. Casey) Handbook of digital forensics and investigation - Academic Prs. 2009
[2] “CatFish” e “fcrackzip”, http://www.ubuntugeek.com
[3] “Large Text File Viewer”, http://www.swiftgear.com
[4] “GHex”, http://www.icewalkers.com
[5] “Digital Framework Forensics”, http://www.digital-forensic.org
[6] “WinAudit”, “RegScanner”, “PhotoRec” e “WhatInstartup” http://it.kioskea.net
[7] “ChromeCacheView”, “IECacheView”, “OperaCacheView”, “VideoCacheView”, “IEHistoryView”, “ChromePass”, “Mail PassView”, “VNCPassView”, “Dialupass” e “PstPassword”, http://www.nirsoft.net
[8] “Portale sulla sicurezza informatica”, http://www.blackhat.com
[9] “Recensioni di tool open source”, http://opensourceforensics.org
[10] “BulkExtractor”, http://bstdownload.com
[11] “LibPST”, http://freshmeat.net
[12] “Seminario Hacker Highschool”, http://www.work4net.it
[13] “Portale di download software”, http://www.softpedia.com
[14] “SkypeHistoryViewer”, http://www.fratellogeek.com
[15] “ClamAv”, http://www.clamav.net
[16] “Analisi su tecniche di anonimato, sicurezza, e difesa della privacy”, http://www.notrace.it
[17] “Soluzioni antivirus e antispam per le aziende”, http://www.sophos.com
[18] “Blog informatico”, http://www.geekissimo.com
[19] “OphCrack”, http://ophcrack.sourceforge.net
[20] “John the ripper”, http://openskill.info
[21] “PDFCrack”, http://informaticafree.ilbello.com
[22] “Hydra”, http://www.darknet.org.uk
[23] “Autopsy”, http://www.sleuthkit.org
[24] “PTK”, http://ptk.dflabs.com
[25] “Web historian”, http://www.mandiant.com
[26] “NetAnalysis”, http://www.digital-detective.co.uk
[27] “Mailing list Computer Forensics Italy”, http://www.cfitaly.net