Univerzitet u Nišu, Elektronski fakultet
Katedra za računarstvo

Forenzika storage medijuma
Digitalna forenzika

Mentor: Prof. dr Bratislav Predić **Student: Jovan Jurić, 1206**

Niš, 2021

Apstrakt

Digitalna forenzika predstavlja proces skupljanja, čuvanja i analiziranja dokaza preuzetih sa digitalnog medijuma. Ukoliko je datotečni sistem na bilo koji način oštećen, korumpiran i sl. treba primeniti tehnike za restauraciju fajlova (eng. *File carving*). File carving je metodologija koja se koristi za prikupljanje i preuzimanje podataka sa memorije koja nije alocirana. Usled skorašnjih napredovanja i istraživanja file carving je postala primarna tehnika za vraćanje podataka kao i pri istrazi u digitalnoj forenzici.

Ključne reči: [Digitalna forenzika, File carving, rekonstrukcija fajlova, datotečni sistem, vraćanje podataka, fajl]

UVOD

Čovek je u zadnjih 30 godina počeo svoje podatke da digitalizuje i ova praksa se sve intenzivnije primenjuje kako sve više ljudi ima pristup računaru, internetu i mobilnim telefonima. Kreiranje i distribucija digitalnih podataka postala je naša svakodnevnica tako da se naše društvo može nazvati i *društvo informacija* [1].

Digitalni svet u kome živimo jeste tu da nam olakša svakodnevne poslove međutim nalazimo se u situaciji gde imamo mač sa dve oštrice. Neke od posledica faktora digitalizacije jeste prilika za kriminalce da profitiraju ilegalnim aktivnostima kao što su *online* prevare, *phishing* napadi, *malware*, *ransomware* i slično. Problem sa kojim se digitalni forenzičari suočavaju jesu velike količine podataka čiji red veličine može dostizati nekoliko stotina gigabajta pa čak i terabajta. Po statistici broj krivičnih dela počinjenih uz pomoć računara se povećava dok se procenat rešenih slučajeva postepeno smanjuje što pokazuje da "sajber" kriminalci postaju sve pametniji i snalažljiviji samim tim potrebno je i napredovanje u digitalnoj forenzici.

Digitalni podaci mogu se smeštati na različitim medijumima kao što su HDD, SDD, SD kartice, USB, računarima u oblaku itd. Datotečni sistem (eng. *File system*) je proces koji upravlja kako i gde se podaci smeštaju na fizičkim medijumima kao i o autorizaciji za pristup podacima. Ukoliko želimo da povratimo izgubljene podatke sa diska neophodno je da se poseduje znanje o datotečnim sistemima.

Koraci za dobijanje celih fajlova od sekvence bajtova prikazan je na Slici 1. Prvo je potrebno grupisati sekvence bitova u sektore. Sektor se smatra se najmanjom mogućom količinom podataka (sekvencom bajtova) koja može da se pročita u upise na disk jednovremeno. Usled toga termin *klaster* predstavlja grupu sektora odnosno najmanju količinu podataka koja se adresira kada se pristupa datotečnom sistemu. Neki datotečni

sistemi klaster nazivaju blokovima. Fragmenti se sastoje od najmanje jednog klastera, dok se termin veličina bloka (eng. *Block size*) odnosi na broj bajtova koji čini jedan blok. U zavisnosti od tipa datotečnog sistema fajlovi se dele na nekoliko fragmenata ili mogu biti "nefragmentisani" (eng. *Unfragmented*) ukoliko se sastoje samo od jednog fragmenta.

Slika 1. Načini za grupisanje bajtova [2]

Fajlovi se identifikuju pomoću potpisa zaglavlja i podnožja (eng. Header i footer). Potpisi fajlova su sekvene bajtova i karakteristični su za taj tip fajla. Primera radi prva dva bajta PNG zaglavlja su 0x89 0x50.

Metode rekonstrukcije fajlova

U nastavku poglavlja opisani su neki od savremenih metoda rekonstrukcije fajlova.

Vraćanje podataka zasnovano na meta podacima

U metodi gde restauriramo fajlove na osnovu meta podataka podaci se gledaju kao na strukture koje odgovaraju datotečnom sistemu u kome se nalaze. U zavisnosti od vrste datotečnog sistema (npr. NTFS, FAT) obrisani ili oštećeni podaci i njihov redosled može ostati nepromenjen usred njihovog brisanja [2].

Ova metoda podržava proces rekonstrukcije fajlova na više načina. Prvi jeste da meta podaci datotečnog sistema sadrže informacije o upotrebljenoj veličini blokova. Veći blokovi podataka povećavaju tačnost u procesu klasifikacije fragmenata. Drugi način jeste da količina podataka koja prolazi kroz proces restauracije može smanjiti ukoliko ne

razmatramo podatke kojima se može pristupiti. Treći način jeste da tip datotečnog sistema pokazuje prirodu fragmentacije fajlova jer svaki datotečni sistem ima drugačiji pristup implementaciji za fragmentaciju fajlova.

Klasifikacija fragmentisanih fajlova

Nakon preprocesiranja podataka, preostali blokovi koji se ne mogu asocirati sa nekim fajlom prosleđuju se procesu klasifikacije fragmentisanih fajlova. U nastavku se u odnosu na pristup reasembliranja podataka određuju tipovi podataka kako bi se dodatno smanjila količina podataka potrebna za reasembliranje ili identifikovanje blokova koji pripadaju specifičnim fajlovima [2].

Pristup traženja po potpisu je koristan za identifikacija početka i kraja fajlova zahvaljujući poznatim sekvencama bajtova hedera i futera. Ova metoda je korisna za nalaženje *tačaka fragmentacije* tj. mesta gde se fragmenti dva različita fajla nalaze jedan pored drugog u memorijskom prostoru [2].

Algoritmi mašinskog učenja kao što su SVM, PCA i neuronske mreže su takođe korišćeni kako bi se otkrili fičeri fajlova čiji tipovi nisu poznati.

Reasembliranje fragmenata

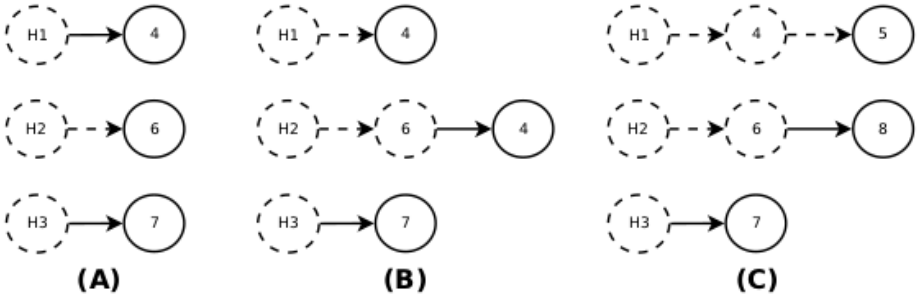
Blokovi koji se uspešno klasifikuju reasembliraju se u ispravnom redosledu. Rezultat reasembliranja fragmenata jeste originalan fajl kom potencijalno fale neki delovi (parcijalno asembliran fajl). Algoritmi reasembliranja mogu se podeliti u tri grupe:

- Pristup baziran fajlovima potpisom
- Pristup baziran na mapiranju (eng. *mapping function*)[3]
- Pristup baziran na grafovima

Algoritmi prve grupe koriste parametre potpise koji se nalaze na hederu i futeru fajla tj. meta podatke.

Drugi algoritam poseduje arhitekturu koja je bazirana na kontrolnoj petlji koja testira validnost oporavljenih fajlova pomoću diskriminatora. Diskriminatori poseduju sposobnost da verifikuju integritet podataka tako što ih sekvencijalno čitaju. Ovaj model zasnovan je na funkciji mapiranja koja mapira ofset unutar fajlova koje treba restaurirati i njihovog ofseta u slikovnom fajlu. Funkcija mapiranja ispunjava tri kriterijuma: prvi je da važi pravilo mapiranja 1:1 gde se mapira ofset slike prema fajlu, drugi kriterijum je da funkcija nije kontinualna zbog fragmentacije fajla i treće je da su funkcije mapiranje nepovratne zato što sektori pripadaju samo jednom fajlu. Ovaj algoritam dobro radi sa podacima koji koriste jedinstvene identifikatore za objekte kojima upravljaju strukture fajlova (kao što su PDF i ZIP fajlovi) [2][3].

Pristup baziran na grafovima razmatra strukture fajlova i semantičke informacije (npr. vrednosti boja za piksele kod JPEG fajlova). Identifikovani fragmenti se reasembliraju u originalni redosled pomoću pohlepnih (eng. *Greedy*) algoritama. Najveći broj algoritama baziranih na grafovima koristi princip paralelnih jedinstvenih puteva (eng. *PUP – Parallel Unique Paths*). Broj fragmenata (koji su karakterističan podatak na početku fajlova) odlučuju o broju paralelnih puteva. U svakom krugu algoritma traže se fragmenti koji najbolje odgovaraju trenutnoj glavnoj putanji reasembliranja. Na početku heder fragment predstavljeni početak grafa. Kompleksnost PUP algoritama je $O(n^2 \log n)$. Slika 2 pokazuje primenu PUP algoritma. U koraku A, tri heder fragmenta su određena. Nakon toga se na narednim slikama može videti da su pronađeni fragmenti koji pripadaju određenim heder fragmentima. Kada se fragment pridruži nekoj putanji on se briše iz tablice dostupnih fragmenata. Proces se ponavlja sve dok se svi fragmenti ne dodele grafu.



Slika 2. Princip paralelnih jedinstvenih puteva

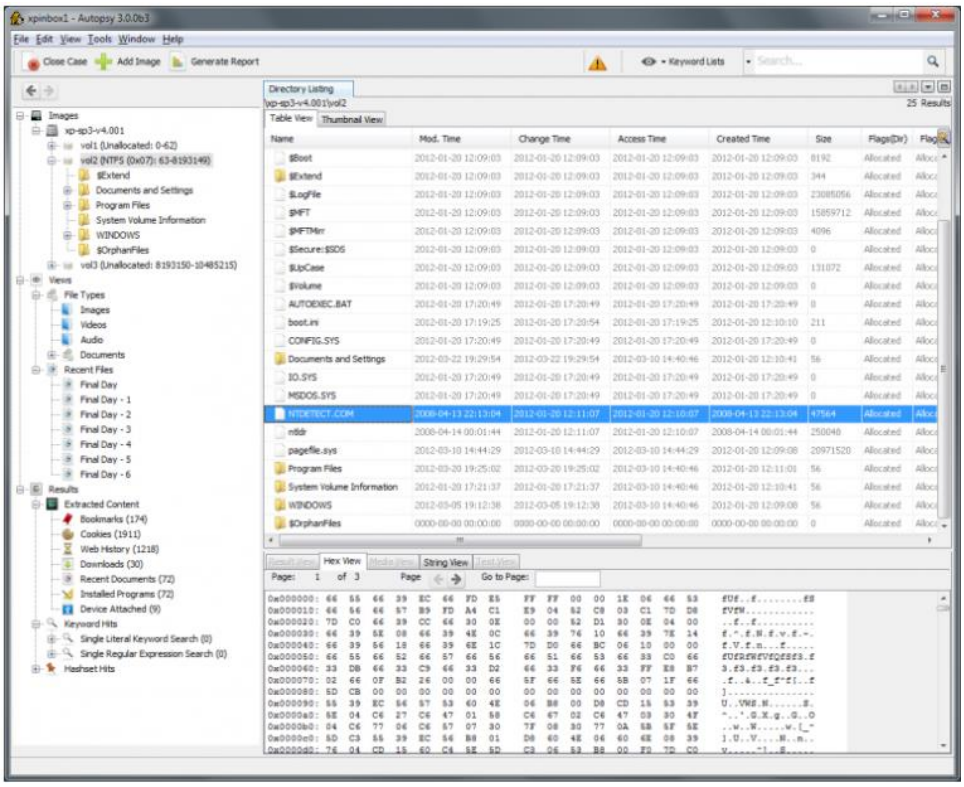
Savremena softverska rešenja

U ovom poglavlju osvrnućemo se na neke od najmodernijih sofverskih alata koji se koriste za digitalnu forenziku i restauraciju fajlova.

Autopsy i Sleuth Kit

Sleuth Kit [4] je biblioteka i kolekcija Unix i Windows baziranih alata koji služe za ekstrakciju podataka sa diska i drugih medijuma za skladištenje podataka kako bi se izvršila forenzička analiza računarskog sistema.

Autopsy [4] je platforma za digitalnu forenziku i interfejs za Slueth Kit i ostale alate koji se koriste za digitalnu forenziku. Koristi se u slučaju sprovođenja zakona, policiji, vojsci pri potrebi istragama na medijumima za skladištenje podataka. Prikaz izgleda Autopsy sofvera može se videti na slici 3.



Slika 3. Izgled Autopsy grafičkog intefejsa

wxHexEditor

Heks editori su programi koji omogućavaju manipulaciju binarnih podataka fajlova kao i datotečnog sistema. Ime "Hex" je izvedeno od heksadecimalnog jer prikazuje podatke u vidu ovog brojnog sistema. Heks editor skuplja sve delove (fragmente) fajla i prikazuje ih ko jedna celina.

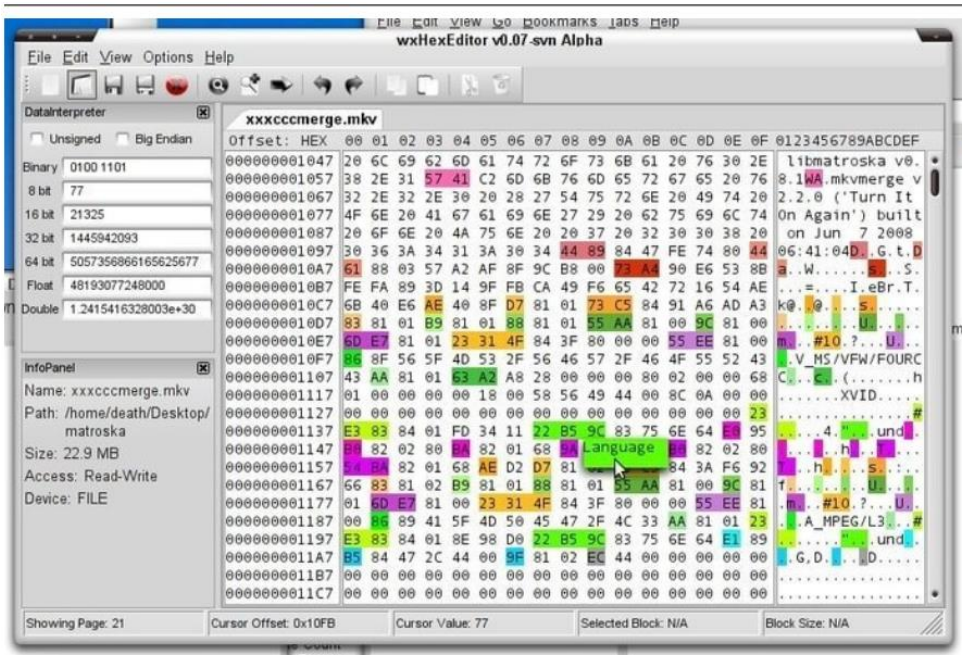
Sa heks editorom, korisnik može videti u uređivati raw podacima. Mogu se koristiti da se isprave oštećeni fajlovi u sistemi ili da reše neke probleme sa aplikacijama gde se vremenski ne bi isplatilo da se napiše program koji ispravlja greške. Može se iskoristiti za "patch"-ovanje izvršnih fajlova (ekstenzija *exe*) fajlova kako bi se promenila ili dodala instrukcija kao alternativa ponovnoj kompilaciji programa.

U većinji heks editora podaci fajla su predstavljene heksadecimalnim vrednostima grupisanim u 4 grupe po 4 bajta (ili 2 grupa po 8 bajtova), i u nastavku sa jednom grupom

16 ASCII karaktera gde svaki karakter obeležava odgovarajući bajt. Karakteri koji se ne mogu odštampati predstavljeni su tačkama.

U odnosu na klasične tekstualne editore, heks editori se efikasno nose sa fajlovima svih veličina jer se učita samo određena količina podatak istovremeno.

wxHexEditor[6] je jedan od najpopularnijih hex ediotra koji može da podrži i najveće količine podataka (do 2 egzabajta), takođe je kompatibilan sa 3 najpopularnija operativna sistema (Windows, Mac OS, Linux), zauzima malu količinu RAM memorije i ne pravi privremene fajlove.



Slika 4. Prikaz wxHexEditor grafičkog korisničkog interfejsa

Photorec

Photorec[5] je open-source softver koji služi za restauraciju podataka i poseduje tekstualni korisnički interfejs. Koristi se za memorije digitalnih foto aparata i hard disk. Može da prepozna do 480 ekstenzija ali moguće je konfigurisati program da traži manje poznate ekstenzije.

Reference

[1] J. Beniger, The Control Revolution: Technological and Economic Origins of the Information Society. Cambridge: Harvard University Press, 1989.

[2] Rainer Poisel, Simon Tjoa, A Comprehensive Literature Review of File Carving, Institute of IT Security Research St. Poelten University of Applied Sciences, St. Poelten, Austria.

[3] M. Cohen, "Advanced carving techniques," Digital Investigation, 2007.

[4] https://www.sleuthkit.org/

[5] https://www.cgsecurity.org/wiki/PhotoRec

[6] https://github.com/EUA/wxHexEditor