SVEUČILIŠTE U ZAGREBU

FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

RAČUNALNA FORENZIKA

PE FORMAT (.EXE, .DLL)

Marko Veizović

Zagreb, siječanj 2017.

Sadržaj

1. Uvod

U suvremenom svijetu u gotovo svim aspektima ljudskog života tehnologija ima veliku ulogu. Za bilježenje svega onoga što se u nekoj djelatnosti napravilo i utvrđivanje onoga što će se tek uraditi, ili za jednostavan način objave nekakve novosti ili izvršavanje neke akcije koristi se digitalni dokument ili datoteka kao osnovni koncept za pohranu, manipuliranje i razmjenu informacija između ljudi i računala [2]. S obzirom na jednostavnost pristupa i brzinu komunikacije, taj je koncept opće prihvaćen. Međutim, tehnološki propusti u sigurnosnom smislu otvaraju mogućnost za prijetnje i napade od treće strane koja inače nema pravo pristupa nekoj datoteci. Razvojem računalne forenzike čiji je zadatak otkriti digitalne tragove i ustvrditi okolnosti neke digitalne radnje razvila se i teorijska podloga koja na detaljan način opisuje datoteke, kao i brojni alati koji mogu kvalitetno analizirati nekakvu datoteku i utvrditi, npr. je li promijenjena; ako je tko ju je promijenio i kada i sl. Sami postupak analize datoteke jest prepoznavanje vrste sadržaja datoteke s obzirom na to da svaki format nosi sa sobom drugačiji sadržaj. Potom se pokušavaju izvući metapodaci, tj. podaci koji opisuju samu datoteku (autor datoteke, vrijeme nastanka, vrijeme zadnje promjene…), a konačno se nastoji ustvrditi postoje li neki ostaci informacija koji ukazuju da je datoteka mijenjana [2]. Na temelju toga donosi se nekakva pretpostavka i radi izvještaj. U ovom seminarskom radu opisat će se karakteristike PE (eng. *Portable Executable*) formata datoteka i dat će se konkretan primjer analize tog tipa datoteke alatom *pev*.

2. PE format

Portable Executable (PE) format jest vrsta datoteke koja se koristi na operacijskim sustavima MS Windows. To je format za izvršne datoteke, objektni kod, DLL-ove (eng. *Dynamic-link library*), datoteke za fontove (FON) i sl. Podatkovna struktura tog formata omogućava operacijskom sustavu da upravlja izvršnim kodom, primjerice kod dinamičkog referenciranja za povezivanje, uvoza i izvoza tablica i upravljanja resursima [3].

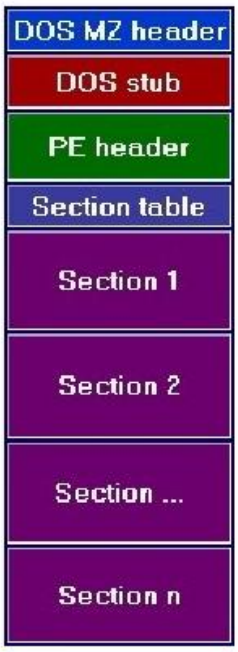
Na Slici 1 a) nalaze se pojednostavljeni, a na Slici 1 b) složeniji prikaz podatkovne strukture PE datoteka. PE datoteka sastoji se od većeg broja zaglavlja i sekcija koji govore operacijskom sustavu kako da mapira datoteku u memoriju, odnosno koje mu dijeljene knjižnice trebaju za okolinu izvođenja. Tada OS kopira taj sadržaj iz trajne pohrane u RAM. Prvo zaglavlje naziva se DOS zaglavlje (eng. *DOS Header*) i zauzima prva 64 bajta svake PE datoteke. Najznačajnija polja tog zaglavlja su: Signature i pokazivač na PE zaglavlje [4]. Polje Signature predstavlja magični broj kojim se ustvrđuje radi li se o PE datoteci [2]. Za taj format to je sekvenca 0x4D5A. Pokazivač na PE zaglavlje omogućuje preskok sljedećeg po redu zaglavlja (DOS stub). DOS stub zaglavlje je 16-bitno polje koje najčešće sadrži poruku „This program cannot be run in DOS mode.“ kojom de upozorava korisnika koji želi koristiti Windows program na DOS-u [4]. PE zaglavlje sadrži skup polja koja opisuju kako izgleda ostatak datoteke, npr. veličinu i lokaciju koda. Neka od njegovih osnovnih polja su: Signature koji služi kao dodatan potpis kako bi ga OS lakše razumio (vrijednost: PE00), Machines koji označava na kojem se stroju izvodi program, NumberOfSections koji definira veličinu tablice sekcija, SizeOfOptionalHeader koji definira veličinu opcionalnog zaglavlja za izvršnu datoteku. Među zastavicama Characteristics nalazi se Image_File_dll koja ima vrijednost 0x2000 ako se radi o DLL datoteci. Polje Image_Optional_Header sadrži važne informacije o datoteci kao što su početna veličina stoga, početak samog programa, verzija operacijskog sustava i sl. Postoje još brojna polja koja bolje opisuju okolinu u kojoj se izvršna datoteka izvodi (virtualna adresa, relativna virtualna adresa, poravnanje sekcija, ukupna veličina, podatkovni direktoriji koji sadrže podatke o sekcijama unutar PE datoteke, kao što su sigurnost, iznimke, resursi i sl.) [4]. Nakon opcionalnog zaglavlja nalazi se tablica sekcija u kojoj se nalaze polja kao što su VirtualSize za veličinu podataka sekcija, SizeOfRawData za veličinu na disku, PointerToRawData kao offset od početka datoteke do podataka sekcija.

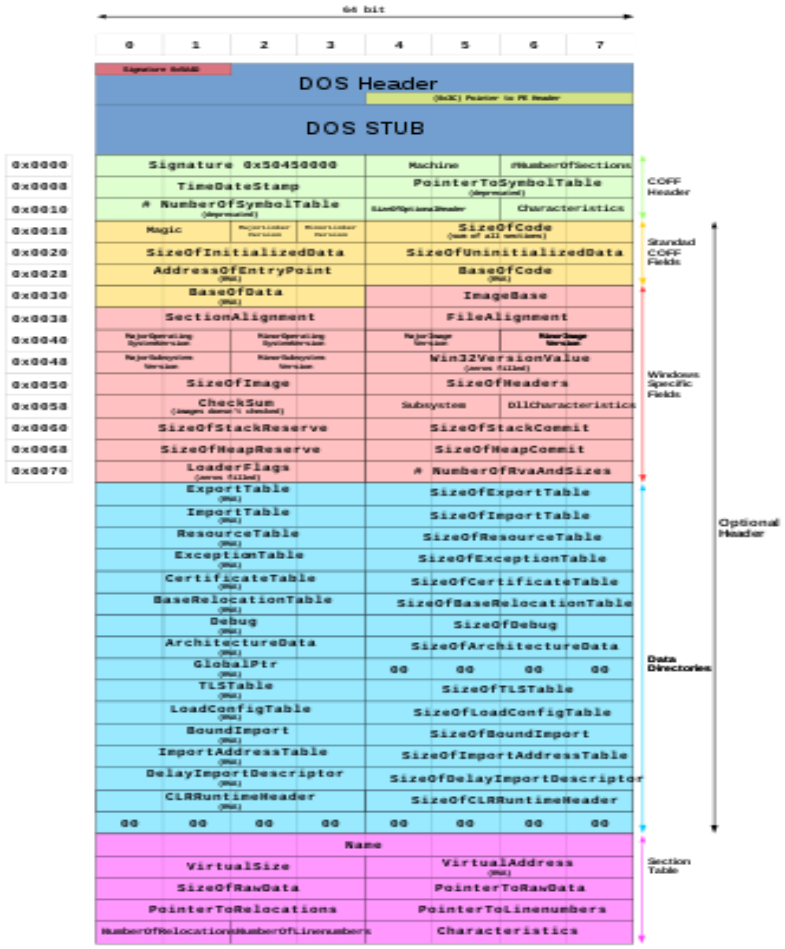
Potom se nalaze PE sekcije, tj. 9 predefiniranih sekcija. Ovisno o aplikaciji, ne koriste se uvijek sve sekcije.

Sekcije su sljedeće:

- .text - sadrži kod
- .bss - neinicijalizirani podaci
- .rdata - podaci samo za čitanje (eng. *read-only*)
- .data -statički podaci definiran i u kodu
- .rsrc - informacije o resursima modula
- .edata - podaci o izvezenim funkcijama
- .idata - podaci o uvezenim funkcijama
- .pdata - podaci o iznimkama
- .debug - podaci o pogreškama

Windows operacijski sustav također dopušta višedretvenost za izvođenje jednog procesa, gdje svaka dretva sprema svoj kontekst zasebno. To se naziva Thread Local Storage (TLS) [4].





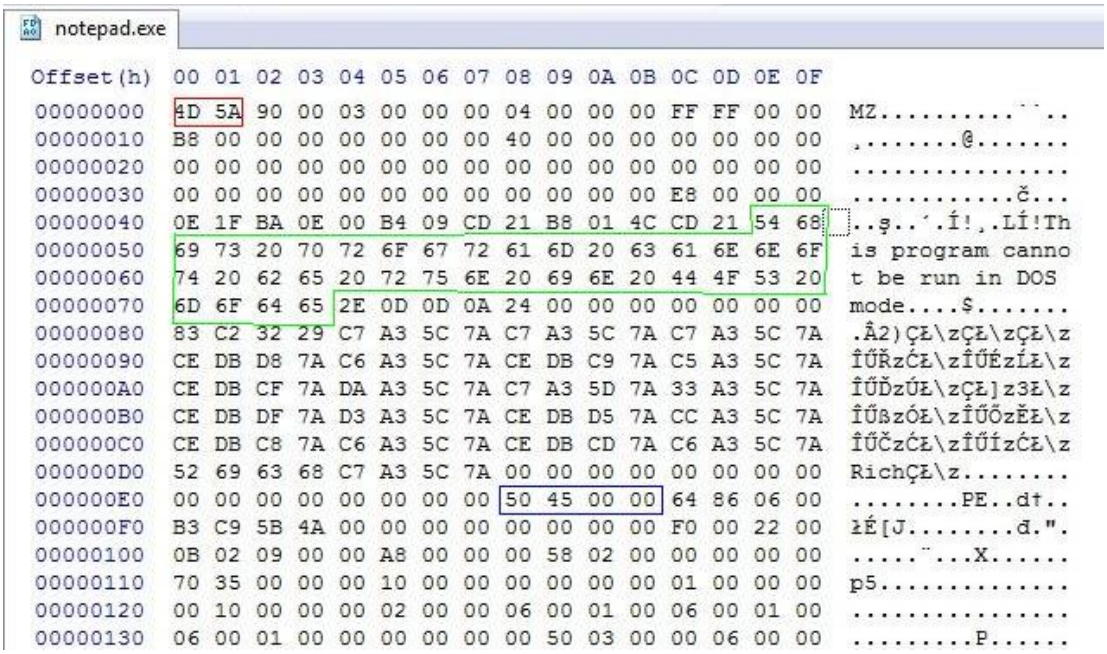
Slika 1 a) pojednostavljeni prikaz PE Slika 1 b) složeniji prikaz PE

2.1. EXE i DLL datoteke

Ovisno o potrebi, PE datoteka može biti EXE (eng. *Executable*) ili DLL (eng. *Dynamic-link library*). Dok EXE datoteka predstavlja samostojeći program, DLL predstavlja skup funkcija i procedura koje mogu koristiti drugi programi. U najosnovnijem programskom paketu najčešće se nalazi jedna EXE datoteka te nijedna, jedna ili više DLL datoteka, ovisno o potrebama programa. EXE datoteke imaju ulaznu točku (eng. *entrypoint*) na kojoj se samostalno počinje izvoditi program, dok DLL datoteke to nemaju i ne mogu se izvoditi samostalno. Osnovna namjena DLL-ova jest da se mogu ponovno koristiti u drugim programima, što znači da se za razliku od EXE datoteka koje su pisane za specifičniju primjenu, DLL datoteke pišu tako da ih može koristiti više programa sa sličnim potrebama i područjem primjene. Nadalje, za razliku od EXE datoteka, DLL-ovi nemaju vlastiti memorijski prostor, već koriste onaj od aplikacije koja ih je pozvala. Zbog toga imaju ograničen pristup resursima [5].

2.2. Prikaz EXE datoteke u HxD-u

Na Slici 2 se nalazi prikaz datoteke notepad.exe u HxD programu (HEX Editoru). Crvenom bojom označen je magični broj, zelenom DOS stub, a plavom potpis (Signature) u PE zaglavlju.



Slika 2 - prikaz notepad.exe u HxD-u

3. Analiza EXE datoteke alatom *pev*

Pev je jednostavni, open source toolkit za analizu PE datoteka. Može se jednostavno instalirati s poveznice „pev.sourceforge.net/#download“.

U nastavku su dani neki primjeri korištenja alata *pev* za analizu PE datoteke volatility.exe. Primjeri su preuzeti i prilagođeni s poveznice „pev.sourceforge.net“.

Primjer 1 - ispis polja određenog zaglavlja

Naredba: readpe --header coff header volatility.exe

```
C:\Users\Korisnik\Desktop\pev-0.80-win32>readpe --header coff header volatility.
exe
COFF/File header
    Machine:                        0x14c IMAGE_FILE_MACHINE_I386
    Number of sections:             4
    Date/time stamp:                1364077614 (Sat, 23 Mar 2013 22:26:54 UTC)
    Symbol Table offset:            0
    Number of symbols:              0
    Size of optional header:        0xe0
    Characteristics:                0x103
    Characteristics names
                                        IMAGE_FILE_RELOCS_STRIPPED
                                        IMAGE_FILE_EXECUTABLE_IMAGE
                                        IMAGE_FILE_32BIT_MACHINE
```

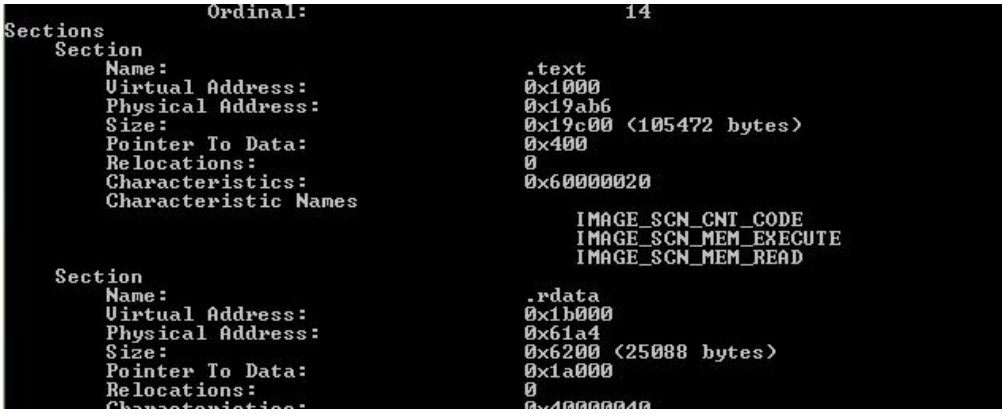
Primjer 2 - ispis polja određenog zaglavlja u XML obliku

Naredba: readpe --format xml --header coff header volatility.exe

```
C:\Users\Korisnik\Desktop\pev-0.80-win32>readpe --format xml --header coff heade
r volatility.exe
<document cmdline="readpe --format xml --header coff header volatility.exe">
    <object name="COFF/File header">
        <attribute name="Machine">0x14c IMAGE_FILE_MACHINE_I386</attribute>
        <attribute name="Number of sections">4</attribute>
        <attribute name="Date/time stamp">1364077614 (Sat, 23 Mar 2013 22:26:54
UTC)</attribute>
        <attribute name="Symbol Table offset">0</attribute>
        <attribute name="Number of symbols">0</attribute>
        <attribute name="Size of optional header">0xe0</attribute>
        <attribute name="Characteristics">0x103</attribute>
        <array name="Characteristics names">
            <attribute>IMAGE_FILE_RELOCS_STRIPPED</attribute>
            <attribute>IMAGE_FILE_EXECUTABLE_IMAGE</attribute>
            <attribute>IMAGE_FILE_32BIT_MACHINE</attribute>
        </array>
    </object>
</document>
```

Primjer 3 - ispis svih polja zaglavlja i sekcija (na slikama prikazane funkcije i sekcije)

Naredba: readpe volatility.exe | more

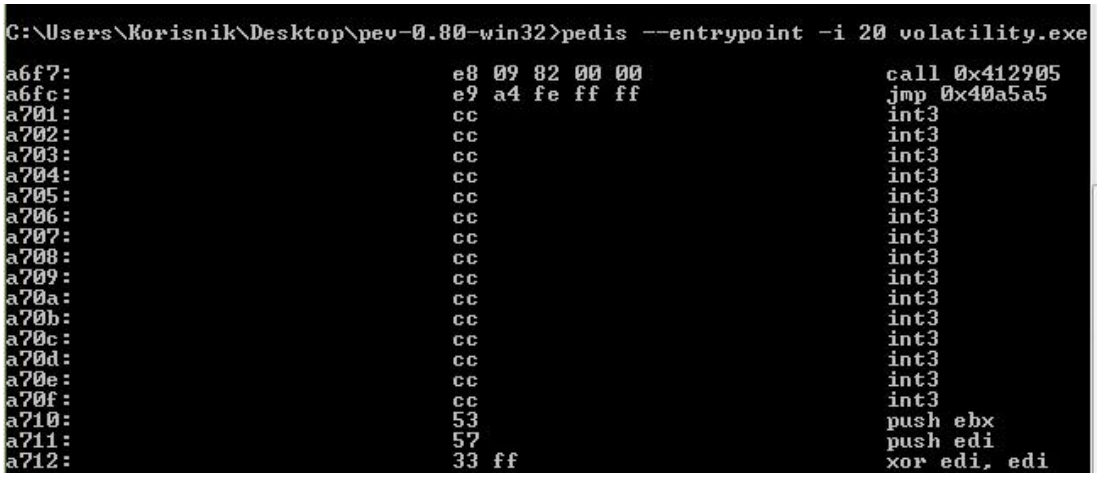


Primjer 4 - izračun hash funkcija za cijelu datoteku i pojedine dijelove

Naredba: pehash -a volatility.exe

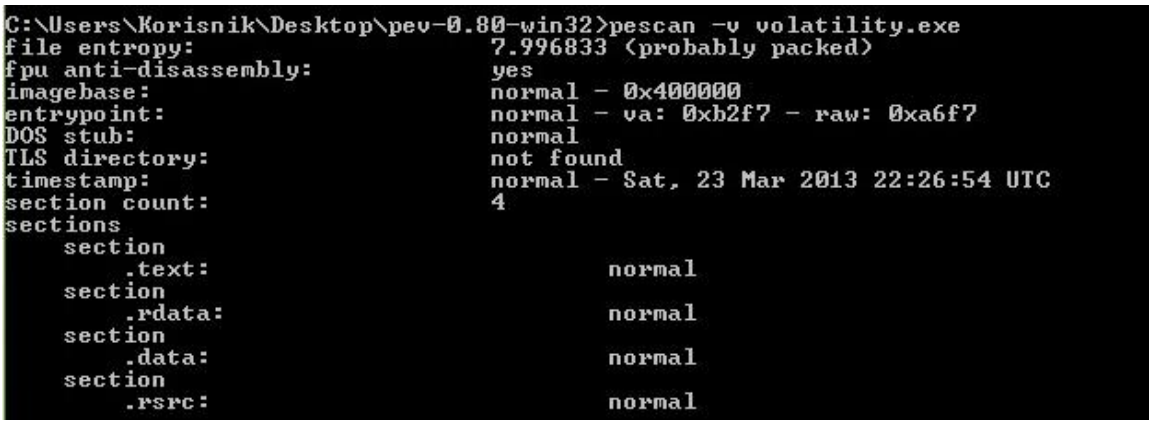
Primjer 5 - pretvorba datoteke u strojni kod

Naredba: pedis --entrypoint -i 20 volatility.exe



Primjer 6 - provjera parametara datoteke

Naredba: pescan -v volatility.exe



Gore navedeni primjeri daju dobar temelj za analizu PE datoteka putem provjere ispravnosti parametara, usporedbom hash funkcija i pregledom svih vrijednosti polja zaglavlja i sekcija.

4. Zaključak

S obzirom na veliku raširenost operacijskih sustava MS Windows, kao i na veliki broj programa koji se razvijaju za široka područja primjene, jasno je kako će se broj datoteka PE formata i dalje povećavati, bilo da se radi o konkretnim samostojećim EXE datotekama ili o popratnim, višestruko upotrebljivim DLL-ovima. Iz toga proizlazi i veliki broj potencijalnih događaja koji će zahtijevati forenzičku analizu kako bi se ustvrdilo tko je kada nešto napravio. Upravo zahvaljujući jednostavnim alatima poput *pev*-a moguće je i osnovnim poznavateljima tehnologije zaći u zanimljivi svijet pronalaska tragova.

5. Literatura

[1] Uvodno predavanje iz predmeta Računalna forenzika, http://www.fer.unizg.hr/_download/repository/RacFor-Uvod-Slides-v10-pp.pdf

[2] Predavanje Forenzika digitalnih dokumenata iz predmeta Računalna forenzika, http://www.fer.unizg.hr/_download/repository/RacFor-Dokumenti-Slides-v13-pp.pdf

[3] Wikipedia - Portable Executable, https://en.wikipedia.org/wiki/Portable_Executable

[4] http://resources.infosecinstitute.com/2-malware-researchers-handbook-demystifying-pe-file/#gref

[5] http://www.differencebetween.net/technology/difference-between-exe-and-dll/