SVEUČILIŠTE U SPLITU
FAKULTET ELEKTROTEHNIKE, STROJARSTVA I BRODOGRADNJE

IZVJEŠTAJ

LABORATORIJSKA VJEŽBA 2

Razumijevanje hash funkcija i ekstenzija

Danica Majić

Split, lipanj 2022.

U sklopu ove vježbe student će se upoznati sa radom hash funkcija kao i sa hexadecimalnom reprezentacijom/notacijom.

1. ZADATAK

Cilj ove vježbe je razumijeti da datoteke imaju jedinstvena zaglavlja na osnovu tipa datoteke. Pokazat ćemo kako datoteke ne trebaju imati ekstenziju za koje se trenutno prikazuju. Veoma bitno je kod računalne forenzike detektirati datateke koje imaju promijenjenu ekstenziju, jer one mogu ukazivati na potencijalno skrivanje informacije.

- Iz direktorija Download sačuvajte datoteku Lab2_download_1.zip te je raspakirajte.
- Vaš zadatak je saznati ekstenziju navedenih datoteka korištenjem python programskog jezika.

HINT: upotrebljavajte python-magic biblioteku:

```
>>> import magic
>>> magic.from_file("testdata/test.pdf")
'PDF document, version 1.2'
# recommend using at least the first 2048 bytes, as less can produce incorrect identification
>>> magic.from_buffer(open("testdata/test.pdf", "rb").read(2048))
'PDF document, version 1.2'
```

- Skinite na vaše računalo te instalirajte Winhex program: http://www.winhex.com/winhex/hex-editor.html

Datoteke kojima treba saznati ekstenziju:

file1	6/28/2022 10:47 PM	Datoteka	13 KB
file2	6/28/2022 10:47 PM	Tekstni dokument	1 KB
file3	6/28/2022 10:47 PM	Datoteka	78 KB

Python kod:

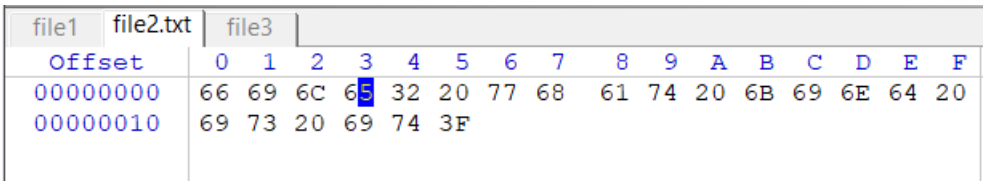
C: > Users > DanicaMajić > Downloads > lab2(1) > lab2 > zad1.py > ...

```python
import magic
import glob

BLOCK_SIZE = 65536
#print("1. file:")
#print(magic.from_file('Lab2_download_1/file1'))
#print(magic.from_buffer(open("Lab2_download_1/file1", "rb").read(2048)))
#print("2. file:")
#print(magic.from_file("Lab2_download_1/file2.txt"))
#print(magic.from_buffer(open("Lab2_download_1/file2.txt", "rb").read(2048)))
#print("3. file:")
#print(magic.from_file("Lab2_download_1/file3"))
#print(magic.from_buffer(open("Lab2_download_1/file3", "rb").read(2048)))

filenames=glob.glob('Lab2_download_1/*', recursive = True)
for filename in filenames:
    print(filename)
    print(magic.from_file(filename))


```

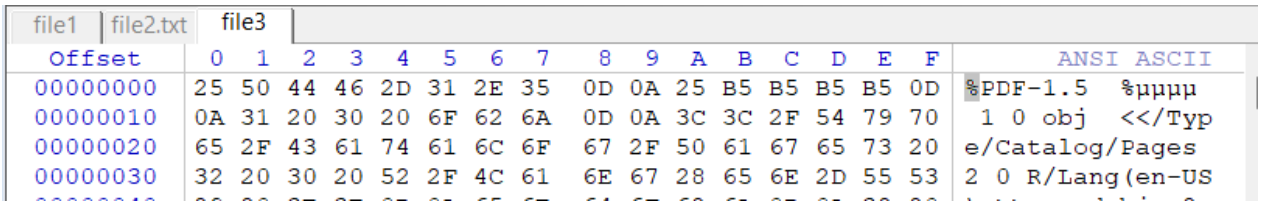
WinHex:

file1 | file2.txt | file3

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	ANSI ASCII
00000000	50	4B	03	04	14	00	06	00	08	00	00	00	21	00	09	24	PK ! $
00000010	87	82	81	01	00	00	8E	05	00	00	13	00	08	02	5B	43	‡, Ž [C
00000020	6F	6E	74	65	6E	74	5F	54	79	70	65	73	5D	2E	78	6D	ontent_Types].xm
00000030	6C	20	A2	04	02	28	A0	00	02	00	00	00	00	00	00	00	l ¢ (
00000040	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000050	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	



Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00000000	66	69	6C	65	32	20	77	68	61	74	20	6B	69	6E	64	20
00000010	69	73	20	69	74	3F										



Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	ANSI ASCII
00000000	25	50	44	46	2D	31	2E	35	0D	0A	25	B5	B5	B5	B5	0D	%PDF-1.5 %µµµµ
00000010	0A	31	20	30	20	6F	62	6A	0D	0A	3C	3C	2F	54	79	70	1 0 obj <</Typ
00000020	65	2F	43	61	74	61	6C	6F	67	2F	50	61	67	65	73	20	e/Catalog/Pages
00000030	32	20	30	20	52	2F	4C	61	6E	67	28	65	6E	2D	55	53	2 0 R/Lang(en-US

2. ZADATAK

Cilj ove vježbe je pokazati kako dvije datoteke kreirane u na različitim uređajima imaju iste hash otiske ukoliko je njihov sadržaj identičan. Također ćemo pokazati iako je sadržaj datoteke identičan (razlikuje se po kapitalizaciji) hash otisak će u tom slučaju biti isti.

- U Notepad-u kreirajte *text* dokument
- U dokument upišite vrijednost test te sačuvajte datoteku pod nazivom test.txt
- Kreirajte novi dokument te u njega upišite vrijednost Test te ga sačuvajte pod nazivom test1.txt
- korištenjem Python programa izračunajte hash vrijednosti navedenih datoteka
- Trebali biste dobiti ove rezultate:

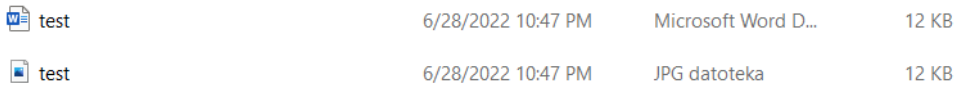
```
--Test.txt--
MD5: 098f6bcd4621d373cade4e832627b4f6
SHA1: a94a8fe5ccb19ba61c4c0873d391e987982fbbd3
SHA256: 9f86d081884c7d659a2feaa0c55ad015a3bf4f1b2b0b822cd15d6c15b0f00a08

--Test1.txt--
MD5: 0cbc6611f5540bd0809a388dc95a615b
SHA1: 640ab2bae07bedc4c163f679a746f7ab7fb5d1fa
SHA256: 532eaabd9574880dbf76b9b8cc00832c20a6ec113d682299550d7a6e0f345e25
```

Dobiveni rezultati:

```
PS C:\Users\DanicaMajić\Downloads\lab2(1)\lab2> python zad2.py
---------zad2\test.txt--------
SHA 256
9f86d081884c7d659a2feaa0c55ad015a3bf4f1b2b0b822cd15d6c15b0f00a08
MD 5
098f6bcd4621d373cade4e832627b4f6
SHA 1
a94a8fe5ccb19ba61c4c0873d391e987982fbbd3
---------zad2\test1.txt--------
SHA 256
532eaabd9574880dbf76b9b8cc00832c20a6ec113d682299550d7a6e0f345e25
MD 5
0cbc6611f5540bd0809a388dc95a615b
SHA 1
640ab2bae07bedc4c163f679a746f7ab7fb5d1fa
PS C:\Users\DanicaMajić\Downloads\lab2(1)\lab2>
```

3. ZADATAK

a) Kreirajte u **Word** programu datoteku te u nju upišite neki sadržaj. Sačuvajte dokument u ekstenziji .docx pod nazivom test na računalu (test.docx). Nakon toga, napravite kopiju word dokumenta te joj promijenite naziv i ekstenziju tako da ime bude identično originalnom dokumentu, dok joj je ekstenzija jednaka ekstenziji slike .jpg (test.jpg). Hoće li hash otisak (**MD5 i SHA1**) obaju dokumenata biti isti.



RIJEŠENJE: Hash otisak (MD5 I SHA1) su isti.

```
PS C:\Users\DanicaMajić\Downloads\lab2(1)\lab2> python zad3.py
---------files\test.docx--------
MD 5
29e29e66fc9636598d28d97ed46c7c1c
SHA 1
78671f9029405fa7956f4a9fedd93b66422f468e
---------files\test.jpg--------
MD 5
29e29e66fc9636598d28d97ed46c7c1c
SHA 1
78671f9029405fa7956f4a9fedd93b66422f468e
```

b) Pretpostavimo da je tvrtka prijavila problem korporativne špijunaže u kojem smatraju da im je ukraden/kopiran tekstualni dokument u PDF-u iznimne važnosti. Budući da tvrtka ne želi otkriti sadržaj dokumenta, forenzični istražitelj dobiva u uvid hash otisak dokumenta:
c15e32d27635f248c1c8b66bb012850e5b342119

```
Found: Dokaz\Secret_file_52.jpg
c15e32d27635f248c1c8b66bb012850e5b342119
c15e32d27635f248c1c8b66bb012850e5b342119
```

Također, sa računala osumnjičene osobe ste izuzeli niz dokumenata koji bi mogli ukazivati na potencijalni dokaz. Dokumente u datoteci Dokaz.zip možete preuzeti iz direktorija Download. Raspakirajte dokumente i napravite analizu te navedite o kojem se dokumentu radi.

KOD:

```python
import glob
import hashlib

BLOCK_SIZE = 65536

filenames=glob.glob('files/test.*', recursive = True)
for filename in filenames:
    print("----------"+filename+"--------")
    sha256_hash = hashlib.sha256()
    md5_hash = hashlib.md5()
    sha1_hash=hashlib.sha1()
    with open(filename, 'rb') as f:
        fb = f.read(BLOCK_SIZE)
        while len(fb) > 0:
            sha1_hash.update(fb)
            sha256_hash.update(fb)
            md5_hash.update(fb)
            fb = f.read(BLOCK_SIZE)
    print("MD 5")
    print (md5_hash.hexdigest())
    print("SHA 1")
    print (sha1_hash.hexdigest())
```

```
print("-----------------------------------------------")
filenames=glob.glob('Dokaz/*', recursive = True)
for filename in filenames:
    sha1_hash=hashlib.sha1()
    with open(filename, 'rb') as f:
        fb = f.read(BLOCK_SIZE)
        while len(fb) > 0:
            sha1_hash.update(fb)
            fb = f.read(BLOCK_SIZE)
    if sha1_hash.hexdigest()=="c15e32d27635f248c1c8b66bb012850e5b342119":
        print(f"\nFound: {filename}")
        print (sha1_hash.hexdigest())
        print("c15e32d27635f248c1c8b66bb012850e5b342119")
```