ALGEBRA

KATEDRA ZA OPERACIJSKE SUSTAVE

Operacijski sustavi

Lab 06 – NTFS datotečni sustav

REV 1.0

Sadržaj

Uvod

U ovoj vježbi ćemo se koncentrirati na pohranu podataka na *storage* medije. Dotični mediji spadaju u kategoriju uređaja koji trajno pamte podatke (eng. *Nonvolatile memory*). Predmet proučavanja u ovoj vježbi će bit datotečni sustavi, kao najniža razina apstrakcije koju, barem trenutno, ima smisla proučavati. Prije nego se pozabavimo datotečnim sustavima i njihovim karakteristikama, podsjetimo se pojmova koji se koriste pod Windows operacijskim sustavima:

- **disk** je fizički uređaj za pohranu podataka, npr. čvrsti disk, CD ROM medij ili floppy medij. Disk je podijeljen u sektore.
- **sektori** su hardverski adresirljivi blokovi fiksne veličine koju određuje hardver. Npr. čvrsti diskovi pod x86 sustavima imaju (uglavnom) sektore veličine 512 B, dok CD ROM sektori imaju veličinu 2048 B. Ipak, Windowsi podržavaju rad i sa sektorima čvrstih diskova većih od 512 B.
- **particije** (eng. *Partition*) su skupovi sekvencijalnih sektora na disku. Particijska tablica sadrži informacije o početnom sektoru particije, veličinu i ostale informacije. Tablica se nalazi na istom disku kao i sama particija.
- **volumeni** (eng. *Volume*) su objekti koji predstavljaju sektore objedinjene iz više particija (eng. *Multipartition volumes*), a koje datotečni sustav tretira kao jednu jedinicu pohrane podataka.
- **datotečni sustav** (eng. *File System*) definira način pohrane podataka na disku, ali definira i mogućnosti pohrane podataka. Npr. datotečni sustav koji ne podržava prava pristupa za datoteke i direktorije ne može imati implementiranu sigurnost u zadovoljavajućoj mjeri. Također, postoje ograničenja i na veličinu datoteka koju je moguće zapisati na datotečni sustav. Ne zaboravimo da datotečni sustavi imaju optimizacije – naime, neki su optimizirani za pohrani velikih (ili većih) datoteka, a neki za pohranu manjih. NTFS i exFAT su primjeri datotečnih sustava koji, ovisno o scenariju upotrebe, nude različit set mogućnosti.
- **klasteri** (eng. *Cluster*) su adresirljivi blokovi podataka koje koriste datotečni sustavi kako bi efikasnije iskorištavali diskovni prostor. Veličina klastera je umnožak veličine sektora – npr. jedan klaster je veličine 4 sektora. Potencijalni problem kod klastera velike veličine je neiskorištavanje diskovnog prostora, do čega dolazi kada se na disk pohranjuje datoteka čija veličina nije jednaka idealnom umnošku veličine sektora. Takav gubitak prostora se zove **interna fragmentacija**.
- **metapodaci** (eng. *Metadata*) su podaci pohranjeni na volumenu koji pomažu pri upravljanju s datotečnim sustavima. Najčešće nisu dostupni aplikacijama, a sadrže informacije koje definiraju položaj datoteka i direktorija na volumenu, te ostalo.

Podržani datotečni sustavi

Windowsi podržavaju nekoliko datotečnih sustava, kao što su: CDFS, UDF, FAT12, FAT16, FAT32 i NTFS. Svaki od navedenih datotečnih sustava zahtjeva upravljački program (eng. *Driver*) kako bi se mogli registrirati za upotrebu. Popis registriranih datotečnih sustava možemo vidjeti pomoću **Winobj** aplikacije iz Sysinternals grupe alata. Winobj čita sadržaj objekata iz U/I upravitelja (komponenta kernel moda Windowsa). Pregledajte registrirane datotečne sustave na sljedeći način:

1. Preuzmite WinObj s linka:
   https://docs.microsoft.com/en-us/sysinternals/downloads/winobj
   te raspakirajete komprimiranu datoteku.
2. Pokrenite aplikaciju **Winobj** iz direktorija gdje ste raspakirali preuzetu datoteku.
3. Kliknite na mapu **FileSystem** i pregledajte koji su datotečni sustavi registrirani u vašem virtualnom računalu.
4. Zatvorite aplikaciju Winobj.

Registrirane upravljačke programe također možete vidjeti i pomoću **MSinfo32** aplikacije, koju ste imali prilike upoznati na prošlim vježbama. Kad ju otvorite, kliknite na **Software Environment** i zatim na **System Drivers**.

----------NAPOMENA----------

Obje aplikacije prikazuju registrirane upravljačke programe neovisno o tome da li su lokalni ili mrežni. Primjerice, Npfs (eng. Named Pipe File System) je mrežni API koji se „ponaša" kao upravljački program datotečnog sustava.

NTFS datotečni sustav

NTFS datotečni sustav ima karakteristike kao što su višestruki podatkovni *streamovi*, Unicode kodirana imena, indeksiranje datoteka, hard i simbolički linkovi, kompresija i enkripcija na razini datotečnog sustava, evidentiranje promjena (eng. *Change logging*), diskovne kvote, POSIX podrška i ugrađen mehanizam defragmentacije. Neke od tih svojstava ćemo upoznati kroz sljedeće primjere.

Hard linkovi

Hard link je mehanizam koji omogućuje višestruko referenciranje jedne datoteke. Pojednostavljeno, hard link je svojevrsna **prečica** (eng. *Shortcut*) do datoteke. Ipak, za razliku od „pravih" prečica, hard linkovi referenciraju isključivo lokalne datoteke. Stoga, ne mogu „pokazivati" na datoteke koje se se protežu na više volumena ili koje se nalaze na mrežnom računalu. Detaljnije, hard link je evidencija na razini datotečnog sustava koja omogućuje da se za pristup datoteci koristi više imena.
Primjerice, ukoliko stvorite hard link s imenom C:\Documents\primjer.doc koji referencira datoteku C:\Users\Administrator\Documents\primjer.doc, obje putanje pokazuju na istu datoteku na disku. Stoga se datoteka može modificirati korištenjem bilo koje putanje. Hard linkove mogu stvoriti procesi korištenjem WIN32 API funkcije *CreateHardLink* ili korištenjem *In* POSIX funkcije. Od Windows 7 verzije podržano je i korisničko stvaranje hard linkova korištenjem **fsutil hardlink** ili **mklink** naredbama.

----------NAPOMENA----------
Puna sintaksa naredbe mklink je **MKLINK [[/D] | [/H] | [/J]] Link Target,** gdje je:
/D - kreira simbolički link za direktorij. Predefinirano, mklink kreira simbolički link za datoteku.
/H - kreira hard link umjesto simboličkog linka.
/J - kreira direktorijski čvor (eng. *Directory Junction*).
Link - specificira ime novog simboličkog linka.
Target - specificira putanju koju referencira novi simbolički link.

Isprobajmo:

1. Pokrenite **Command Prompt**
2. Pozicionirajte se **u C:** direktorij
3. Upišite naredbu **md link**
4. Upišite naredbu **cd link**
5. Upišite naredbu **echo hello > test.txt**
6. Upišite naredbu **mklink hard.txt test.txt /H**
7. Upišite naredbu **dir *.txt**
8. Proučite ispis naredbe; da li su stvorene datoteke identične po pitanju prava pristupa i veličine? Koliko prostora na disku zauzimaju obje datoteke? Koje je vrijeme stvaranja datoteka?
9. Upišite naredbu **notepad hard.txt**
10. Dodajte bilo gdje u dokument riječ **World** i spremite promjene.
11. Upišite naredbu **notepad test.txt**. Da li je tekst World prisutan?
12. Obrišite datoteku hard.txt upisivanjem naredbe **del hard.txt**. Da li je obrisana datoteka test.txt?
13. Upišite naredbu **mklink hard.txt test.txt /H**

14. Upišite naredbu **del test.txt**
15. Što se dogodilo s datotekom hard.txt? Da li je i ona izbrisana?

Simbolički linkovi

Uz prethodno opisane hard linkove, NTFS datotečni sustav podržava još jedan način referenciranja datoteka korištenjem višestrukih putanja, odnosno aliasa. **Simbolički** ili **soft linkovi** su stringovi koji mogu pokazivati na bilo koji volumen na lokalnom računalu, ali i na dijeljene mape na mrežnim računalima. Simbolički linkovi koriste relativnu ili apsolutnu putanju i ne povećavaju broj datoteka (eng. *File Count*). Primjerice, ukoliko obrišete datoteku do koje pokazuje simbolički link dolazi do gubitka podataka (za razliku od hard linkova gdje ne dolazi). Sam simbolički link se tada ne briše automatski; on ostaje kao datoteka s neispravnom referencom do datoteke na koju je izvorno pokazivala. I za kraj, spomenimo da simbolički linkovi mogu pokazivati na direktorije, za razliku od hard linkova. Upravo ova mogućnost čini ih praktičnima za uporabu mnogo češće od hard linkova.
Primjerice, ukoliko je postavljen simbolički link **C:\Driveri** na direktorij **%SystemRoot%\System32\Drivers**
aplikacija koja želi pročitati datoteku **C:\Driveri\Ntfs.sys** zapravo čita datoteku s putanje **%SystemRoot%\System32\Drivers**. Možemo zaključiti da ćete simboličke linkove koristiti kad želite ubrzati pristup direktorijima koji su duboko u strukturi (hijerarhiji) direktorija, kao gore spomenuti Drivers direktorij.
Kao i hard linkovi, simbolički se linkovi mogu kreirati programski, pozivom funkcije *CreateSymbolicLink* iz WIN32 API-ja ili korisnički, putem mklink naredbe.
Sljedeći primjer demonstrira neke od razlika između hard i soft linkova.

1. Upišite naredbu **echo hello > primjer2.txt**
2. I opet pričekajte 1 minutu ☺
3. Upišite naredbu **mklink soft.txt primjer2.txt**
4. Upišite naredbu **mklink hard.txt primjer2.txt /H**
5. Upišite naredbu **dir *.txt**
6. Proučite ispis naredbe; koje su razlike između hard i simboličkog linka (vrijeme kreiranja, veličina, prava pristupa…)?
7. Upišite naredbu **del primjer2.txt**
8. Što se dogodilo s datotekom **soft.txt**? Da li postoji? Ako da, možete li joj pristupiti?
9. Upišite naredbu **del *.*** i potvrdite s **Y** brisanje sadržaja direktorija.

----------NAPOMENA----------

NTFS podržava i treću vrstu linkova – **direktorijski čvor** (eng. *Directory Junction*). Čvor je, u osnovi, ista vrsta linka kao i simbolički link ali podržava isključivo referenciranje datoteka i direktorija na volumenima lokalnog računala. Ipak, podržan je na starijim verzijama Windows OS-a, dok simbolički linkovi nisu. Naglasimo da korištenje direktorijskog čvora nema niti jednu prednost naspram simboličkih linkova. Koristite ga isključivo za potrebe kompatibilnosti unatrag.

Dostupnost linkova

Hard i simbolički linkovi mogu stvoriti probleme pri radu aplikacija koje nisu nativno pisane za novije verzije Windowsa. Problemi su kudikamo izraženiji ukoliko se koriste linkovi koji referenciraju objekte na mrežnim računalima. Primjerice, takav link bi bio onaj koji na računalu A referencira resurse računala B kojima pristupa računalo C. Stoga su, predefinirano, takvi linkovi pod od Windows 7 verzije isključeni. Dostupne linkove na svom računalu možete doznati putem naredbe

fsutil behavior query SymLinkEvaluation

Predefinirano, omogućeni su linkovi na relaciji **lokalni stroj->lokalni stroj** i **lokalni stroj->udaljeni stroj**. Linkovi **udaljeni stroj ->lokalni stroj** i **udaljeni stroj->udaljeni stroj** su onemogućeni. Možete ih uključiti pomoću naredbi

fsutil behavior set SymLinkEvaluation R2R:1 i **fsutil behavior set SymLinkEvaluation R2L:1**

Dodatni podatkovni *streamovi*

Datoteke pod NTFS sustavom imaju **atribute**, kao što su ime datoteke, vlasnik, vremenski zapis, sadržaj itd. Svaki atribut se sastoji od jednog *streama*, koji je, u osnovi, niz bajtova. Ovakva implementacija omogućava lako dodavanje dodatnih atributa datotekama, odnosno, dodatnih *streamova*. S obzirom da je i sadržaj datoteke samo još jedan atribut, odnosno *stream*, NTFS datoteke i direktoriji mogu sadržavati višestruke *streamove*.

NTFS datoteke imaju, predefinirano, jedan jedini podatkovni *stream* koji čak nema ni ime. Aplikacije mogu stvarati dodatne *streamove*, imenovati ih i pristupati im putem tog imena. Kako bi izbjegli moguće konflikte putem Windows I/O API-ja koji kao argument uzima **ime** datoteke, pristup dodatnim *streamovima* se ostvaruje korištenjem simbola **:** (dvotočka) između imena datoteke i *streama* kojem želimo pristupiti, npr. **mojfile.txt:stream2**.

Jedna od komponenti Windowsa koja koristi višestruke *streamove* je servis **Attachment Execution**. Dotični servis se poziva kada iz aplikacije kao što je Internet Explorer ili Outlook želite spremiti datoteku na disk. Ovisno o internetskoj zoni (My Computer, Intranet ili Untrusted) iz koje je preuzeta datoteka, Windows Explorer će upozoriti korisnika da je datoteka potencijalno opasna, ili će joj čak u potpunosti zabraniti pristup. Internet Explorer također koristi dodatne *streamove* za pohranu zabilježenih web stranica (eng. *Favorites*). I druge aplikacije mogu koristiti višestruke *streamove*. Primjerice, aplikacija za izradu sigurnosnih kopija podataka (eng. *Backup Utility*) može koristiti dodatan *stream* kako bi u njega pohranila informacije o vremenu kada je izražena kopija datoteke.

----------NAPOMENA----------

Većina Windows aplikacija ne zna raditi s višestrukim *streamovima*, ali Notepad, kao i naredbe **echo** i **more** znaju. Naredba **echo** je jednostavna naredba koja prikazuje poruke unutar prozora Command Prompta. Sintaksa naredbe je **ECHO [poruka]**. Npr. **echo bok** na ekran ispisuje tekst **bok**.

Naredba **more** ispisuje sadržaj datoteke ili naredbe ekran po ekran. Njena sintaksa je složenija od naredbe echo jer ima brojne prekidače – možete ju saznati pomoću naredbe **help echo** u Command Promptu.

Pomoću naredbi *echo* i *more* ćemo vidjeti kako zapisati i pročitati višestruki *stream* iz datoteke.

1. Pokrenite **Command Prompt** i pozicionirajte se u direktorij **C:**
2. Upišite naredbu **md Stream**
3. Upišite naredbu **cd Stream**
4. Upišite naredbu **echo običan tekst>test.txt**
5. Upišite naredbu **echo super tajni tekst>test.txt:tajno.txt**
6. Upišite naredbu **notepad test.txt**. Koji tekst je prikazan u Notepadu? Prvi ili drugi? Zatvorite Notepad.
7. Upišite naredbu **notepad test.txt:tajno.txt**.
8. Koji je sad tekst prikazan?
9. Zatvorite Notepad.
10. Upišite naredbu **dir /r** i proučite kako su evidentirani *streamovi* datoteke **test.txt**.
11. Upišite naredbu **D:\streams.exe –s C:\Stream**
12. Proučite ispis naredbe.
13. Upišite naredbu **del *.*** i potvrdite s **Y** brisanje sadržaja direktorija.

----------NAPOMENA----------

Uz naredbu **dir/r** i alat **streams**, dodatne *streamove* u datoteci možete vidjeti pomoću GUI aplikacije **Alternate Stream Viewer**. Dotičnu možete besplatno preuzeti s adrese http://www.nirsoft.net/utils/alternate_data_streams.html

Na prethodnom primjeru ste vidjeli kako se u tekstualnu datoteku upisuje dodatni tekstualni *stream*. Ipak, dodatni stream datoteke ne mora biti s istom ekstenzijom – moguće je u .txt datoteku upisati *stream* s .exe ili drugom ekstenzijom. Očito je da ovakvo „maskiranje" izvršnih datoteka unutar tekstualnih datoteka može imati maliciozne namjere. Microsoft je u od Windows Viste onemogućio izvršavanje .exe datoteka iz dodatnih *streamova*.

1. Pozicionirajte se u direktorij **C:**
2. Upišite naredbu **cd Stream**
3. Upišite naredbu **echo test > maskiranje.txt**
4. Upišite naredbu
   type C:\Windows\notepad.exe > C:\Stream\maskiranje.txt:note.exe
   Ta naredba je u datoteku maskiranje.txt „kopirala" benignu (u ovom slučaju) izvršnu datoteku Notepad aplikacije.

5. Upišite naredbu **start maskiranje.txt:note.exe**. Naredba će javiti grešku, zbog prije spomenute nemogućnosti izvršavanja .exe datoteka iz dodatnih podatkovnih *streamova*.
6. Zatvorite prozor s greškom i vratite se u Command Prompt. Ograničenje možemo izbjeći kreiranjem simboličkog linka na dodatni *stream* datoteke.
7. Upišite naredbu
   mklink radi.exe C:\Stream\maskiranje.txt:note.exe
8. Upišite naredbu **radi.exe**. Notepad se neće pokrenuti radi zaštite.
9. Upišite naredbu **del *.*** i potvrdite s **Y** brisanje sadržaja direktorija

----------NAPOMENA----------

Ponovimo da su simbolički linkovi reprezentirani **datotekom**. Dakle, u gornjem primjeru će u direktoriju Stream, uz datoteku maskiranje.txt bit prikazan i datoteka simboličkog linka. Ukoliko ju želite sakriti jednostavno joj promijenite atribut na **Hidden** (desni klik na datoteku, Properties, opcija Hidden). Datoteka s atributom Hidden se ne prikazuje ni u ispisu **dir /r** naredbe.

Što treba znati nakon ove vježbe?

1. Znati što je sektor, klaster i datotečni sustav
2. Objasniti pojam interne fragmentacije
3. Objasniti što su hard link i simbolički link
4. Objasniti razlike na primjeru između hard i simboličkog linka
5. Kreirati hard link
6. Kreirati simbolički link
7. Opisati što je to *stream* i izlistati postojeće streamove u datoteci
8. Zapisati sadržaj u alternativni *stream* datoteke
9. Pročitati sadržaj iz alternativnog *streama* datoteke