

1. Simona Miboučiková

a.) Setuid a Setgid pre spustiteľné súbory

↑  
platí pre užívateľa

↑  
platí pre skupinu

- prístupové práva umožnia spustiť súbor. (X - execute) X majú programy a empty

Po vykonaní exec sa UID procesu zmení na UID vlastníka súboru

Ak je vlastníkom root, súbor je spustiteľný i nastavené setuid

po exec proces beží pod root (napr. použijeme pri zmene prihlasovacej hesla)

b.) setgid pre adresáre

všetky nové súbory v adresári majú pridelenú rovnakú skupinu

Novovytvorené podadresáre majú zdedený setgid bit. ako adresár

3. ls - ukazuje nám zoznam súborov v aktuálnom adresári

4.

a.)  $ls \$X$  '\$X' echo interpretuje X ako slovo

b.)  $ls -l$  - vypisala všetky súbory v adresári ~~interpretuje~~ (echo \* & ls)  
 $\$X$  dereferencuje premennú x

d.) úvodzovky odšpeciálnujú všetko okrem \$

\*  $\$X$  = znakový súbor  $\Rightarrow$  ~~ls~~  $\$X$  \* = echo \*

$ls$  echo echo \*  
↖  
vypis súborov

c.)  ~~$\$X$  reference~~ na

$\$(echo *)$

↖  
vykonaj príkaz ktorý je v zátvorke

echo \$X - vykoná príkaz \$X  
vypis

⑤

run-test

script1.py

~~test-in.txt~~

#!/bin/bash

NA2O V = \$(echo "\$1" | sed 's/\./py\$/1')

#

if test ! -d NA2O V

then

mkdir NA2O V

~~fi~~

~~cp test-in.txt~~

cp "\$1" ~~to~~ test-in.txt

cat \$1 > test-out.txt

fi

⑥

ask

def txt-to-words(txt):

x = txt.split()

for i, item in x

if x[i] == " " or x[i] == " "

x.remove(i)

continue

new\_x = x

- (1) (10 bodov) Vysvetlite, čo znamenajú prístupové bity  
 (a) `setuid` a `setgid` pre spustiteľné súbory,  
 (b) `setgid` pre adresáre.
- (2) (10 bodov) Na základe vašej znalosti použitých systémových funkcií *vysvetlite*, čo a ako robí funkcia `forkmap`. Na riadkoch 35–40 vidíte príklad jej použitia. Vaša odpoveď by sa nemala skladať z viet typu „na riadku *n* sa volá funkcia *foo*“ (pretože taká odpoveď preukazuje iba to, že viete čítať) ale mala by preukazovať vaše porozumenie veci.  
 Je to napísané v Pythone ale to by pre Vás nemal byť problém.  
 Ako dlho (rádovo) si myslíte, že ten program beží?

```

1  import os      importujeme module
2  import sys
3  import time
4
5  def forkmap(f,zoz):
6
7      read_end,write_end=os.pipe() ← vytvára rúru s čítavým a zapisujúcim
8      read_file=os.fdopen(read_end) ← do premennej read_file      deskriptorom
9      write_file=os.fdopen(write_end,"w") ← otvára čítavý koniec
10
11  → for i,item in enumerate(zoz): ← otvára súbor na zápis
12      if os.fork()==0: nastane, ak sme v child procese
13          write_file.write("%d %d\n" % (i,f(item))) ← zapis do súboru
14          write_file.flush() ← vymazanie
15          sys.exit() ← ukončí proces "medzi pamäť"
16
17      write_file.close() ← zatvorenie súboru na zápis
18
19      for x in zoz:
20          os.wait() ← čaká na dokončenie child procesu
21
22      zoz_ret=[0]*len(zoz)
23
24      while True:
25          line=read_file.readline() ← z read_file sa číta po riadkoch
26          if not line: ← ak už nie je čo prečítať program sa ukončí
27              break
28          i_s,result_s=line.split() ← reže v riadku rozparsuje
29          zoz_ret[int(i_s)]=int(result_s) ← do zoznamu sa uložia jednotlivé znaky
30
31      return zoz_ret
32
33  def square(x):
34      # Dlhý výpočet
35      time.sleep(5)
36      return x*x
37
38  print(forkmap(square,range(100)))

```

ty' -0 zoznamy

1  
2

funkcia zapisuje do súboru čísla od 0 po 100 a aplikuje na ne mocniny

- (3) (10 bodov) Uvažujme takúto situáciu: máme spustený bash na termináli, príkaz `ls` vypisuje toto:

1

0\*0  
1\*1  
:  
n\*n



- (1) (10 bodov) Vysvetlite, čo znamenajú prístupové bity  
 (a) `setuid` a `setgid` pre spustiteľné súbory,  
 (b) `setgid` pre adresáre.
- (2) (10 bodov) Na základe vašej znalosti použitých systémových funkcií *vysvetlite*, čo a ako robí funkcia `forkmap`. Na riadkoch 35–40 vidíte príklad jej použitia. Vaša odpoveď by sa nemala skladať z viet typu „na riadku *n* sa volá funkcia *foo*“ (pretože taká odpoveď preukazuje iba to, že viete čítať) ale mala by preukazovať vaše porozumenie veci.  
 Je to napísané v Pythone ale to by pre Vás nemal byť problém.  
 Ako dlho (rádovo) si myslíte, že ten program beží?

```

1  import os      importujeme module
2  import sys
3  import time
4
5  def forkmap(f,zoz):
6
7      read_end,write_end=os.pipe()  ← vytvára rúru s čítavým a zapisujúcim
8      read_file=os.fdopen(read_end) ← do premennej read_file      deskriptorom
9      write_file=os.fdopen(write_end,"w")  ← otvára čítavý koniec
10
11     ← pre každý prvok zo zoznamu (iteruje)
12     for i,item in enumerate(zoz):
13         if os.fork()==0: nastane fork a sme v child procese
14             write_file.write("%d %d\n" % (i,f(item)))  ← otvára súbor na zápis
15             write_file.flush()  ← vymazanie  ← zápis do súboru
16             sys.exit()  ← ukončí proces "medzi pamäť"
17
18     write_file.close()  ← zatváranie súboru na zápis
19
20     for x in zoz:
21         os.wait()  ← čaká na dokončenie child procesu
22
23     zoz_ret=[0]*len(zoz)
24
25     while True:
26         line=read_file.readline()  ← z read_file sa číta po riadkoch
27         if not line:  ← ak už nie je čo prečítať program sa ukončí
28             break
29         i_s,result_s=line.split()  ← reže sa slova v riadku rozparuje
30         zoz_ret[int(i_s)]=int(result_s)  ← do zoznamu sa uložia jednotlivé znaky
31
32     return zoz_ret
33
34     def square(x):
35         # Dlhý výpočet
36         time.sleep(5)
37         return x*x
38
39     print(forkmap(square,range(100)))

```

1  
2  
funkcia zapisuje do súboru čísla od 0 po 100 a aplikuje na ne druhú mocninu  
0\*0  
1\*1  
:  
n\*n

- (3) (10 bodov) Uvažujme takúto situáciu: máme spustený bash na termináli, príkaz `ls` vypisuje toto: