Administrace uživatelů, identita, práva administrátora

Úvod do administrace operačních systémů

Jan Žďárek

České vysoké učení technické v Praze
FEL, katedra počítačů

(Program STM – kód Y36UAD)

Úvod (1)

UNIXové systémy běžně slouží mnoha uživatelům.

Konvence:

- Všechny soubory a procesy v UNIXu vlastní konkrétní uživatel (uživatelský účet).
- Bez povolení uživatele (majitele účtu) nemohou ostatní uživatelé k těmto objektům přistupovat.

Důsledek: tato konvence pomáhá chránit uživatele před chybou či zlým úmyslem jiných uživatelů.

Operační systémy umožňují uživatelské účty spravovat. Otázkou je, jak efektivně.

Úvod (2)

Systémové procesy a soubory vlastní speciální uživatel (*superuživatel,* `root`). Může se stát libovolným jiným uživatelem, resp. působit jako vlastník libovolného procesu nebo souboru (jen pro něj neplatí uvedená konvence). Chceme-li provést v systému něco, co není dovoleno běžnému uživateli, musíme použít *identitu* uživatele `root`, UID=0.

- Nesprávná správa účtů na počítači s mnoha uživateli může usnadnit napadení systému.
- I když máme nástroje pro automatickou práci s účty, je nutné rozumět změnám, jež v systému provádějí.

Identita v UNIXu

UNIX tradičně rozlišuje vnitřní a vnější identitu uživatele.

- Vnější identita je skutečná identita uživatele. Je využita např. při
  1. přihlašování se do systému,
  2. pro název domovského adresáře,
  3. pro e-mailový účet.
- Vnitřní identita je identita procesu nebo souboru v OS. Vnitřní identita je pro nás zajímavá, umožňuje řídit přístup k prostředkům.

Práva a vlastnictví (1)

1. Každý objekt (soubor, proces) – vlastník, skupina vlastníka.
2. Pouze vlastník: právo měnit vlastnosti (např. práva přístupu).
   (Může odebrat práva sám sobě? Projeví se to a jak? Kdo může tento stav změnit?)
3. UID, GID.
   (Co je pro jádro primární informace (12867, `novaj1`; 1000, `users`?)
4. `/etc/passwd; /etc/group`

Např. pro soubory existují na většině UNIXů rozšíření – ACL (*Access Control List*).

Práva a vlastnictví – opakování (2)

Kolik *identit* má proces?

1. Vlastník a skupina vlastníka: RUID, RGID.
2. Efektivní vlastník a skupina vlastníka: EUID, EGID.
3. Většina moderních UNIXů navíc: *saved set-UID/set-GID*.

- Proces superuživatele si může libovolně změnit UID/GID, včetně R-ID – úplná změna identity. (`login`)
- Proces běžného uživatele to udělat nemůže, pokud nemá nastavený některý *set-ID bit*. I tehdy ale může pouze nastavit E-ID na R-ID nebo saved-set ID a zpět.

(Proč proces obyčejného uživatele nemůže změnit reálnou identitu, není to zbytečné omezení?)

Práva a vlastnictví – opakování (3)

SetUID/setGID/sticky:

1. Nastaveno na souboru:
   - SetUID bit: Při spuštění souboru (programu) – EUID=UID vlastníka souboru.
   - SetGID bit: Při spuštění souboru (programu) – EGID=GID skupiny vlastníka souboru.
   - Sticky bit: Dnes již nemá význam.
2. Nastaveno na adresáři:
   - SetUID bit: Nemá význam.
   - SetGID bit: Při vytváření souborů EGID=GID skupiny vlastníka adresáře.
   - Sticky bit: Speciální pravidlo pro mazání souborů v adresáři.

Práva a vlastnictví – opakování (4)

SetUID/setGID/sticky:
UNIXový způsob, jak běžným uživatelům dobře definovaným a spolehlivým způsobem povolit využití některých pravomocí superuživatele.

`$ /bin/passwd` (Přesto pro provedení změny vyžaduje znalost hesla uživatele.)

`# /bin/passwd novaj1` (Od skutečného `root`a znalost hesla uživatele nepožaduje.)

(Proces může snadno zjistit, kým byl spuštěn.)

(Je nutné, aby se superuživatel jmenoval `root`? Je nutné, aby měl UID=0? Co je potřeba zajistit, aby tomu tak bylo? Viz též otázky na str. 5. Kdo/co vynucuje respektování práv: souborů, procesů, prováděných systémových funkcí, operací procesoru?)

Účty uživatelů a pseudouživatelů

Na většině UNIXových systémů nalezneme vedle účtu superuživatele a běžných uživatelů také „běžné účty“, které nepatří žádnému člověku. Nedá se na ně ani přihlásit, jsou zablokované.

Virtuální vlastníci SW.
`bin`, `daemon`, `sys`, `apache`, `bind`, `...`

Mnoho programů má dnes nejen „svého“ vlastníka, ale i skupinu (`bind:bind`, `apache:apache`).
Používají se výhradně k řízení přístupu (práva), SW se na ně nepotřebuje přihlašovat.

Opakování – `/etc/passwd`

Databáze uživatelů systému.
Řádek: sedm polí oddělených dvojtečkami.
`user:pass:UID:primGID:gecos:homedir:login shell`

Př.: `root:x:0:1:Super-User:/root:/sbin/sh`

Soubor musí být přístupný všem pro čtení. (Proč?)

Silný HW $\rightsquigarrow$ hádání hesla, je-li k dispozici jeho hash.

Proto všechny dnešní UNIXy podporují uložení hash-hodnot do odděleného souboru (*shadow passwords*, `/etc/shadow`).
Soubor `/etc/shadow` má nastaveno právo jen pro čtení uživatelem `root`, nebo dokonce žádná práva.
(Jak je možné se přihlásit?)

Položky `/etc/passwd` (1)

- Přihlašovací jméno (*login name*):
  - Case-sensitive. (`login`)
  - Volba a prosazení jednotného pojmenovávacího schématu je důležitým úkolem administrátora.
  - Jednotné a jednoznačné (`rsh`; `ssh`) pojmenování uživatele napříč systémy.
  - Odhadnutelnost uživatelského jména (e-mail).
  - `novakj12` někomu sice nepřijde „hezké“, ale
    - zajistí jednoznačnost jména v celé organizaci,
    - bude použitelné i v NIS (limit 8 znaků),
    - zajistí pohodlné a rychlé přihlašování.

    (`Jaroslav.Nováček`)
  - Pokud jsou uživatelská jména provázána na e-mailové účty, tak povolení „uživatelských“ jmen typu `EvilRider` nebo `RubitaDelSur` může zesměšňovat nejen jejich držitele, ale také celou firmu.

Položky `/etc/passwd` (2)

- Hash hesla (bývá zde `x` $\rightsquigarrow$ uloženo v `/etc/shadow`):
  - Při ručním vytváření účtu „`*`“ – zablokuje účet před vytvořením hesla.
  - Pro šifrování pomocí DES je významných jen prvních osm znaků hesla.
  - Pro dnes standardní šifrování pomocí MD5 jsou všechny znaky významné.
  - Hash-řetězce dvou stejných hesel nebudou s vysokou pravděpodobností stejné
    - *salt*, DES: $2 \times [\text{a-zA-Z0-9./}], 2^{12}$, MD5: 2^{128}.
  - Hash-řetězce DES a MD5 poznáme:
    - $|\text{DES řetězec}| = 13$,
    - $|\text{MD5 řetězec}| = 34$, začíná „$1$“;
    - Předpona „$`X`$“ – `X` = id metody.
    - SHA-256/512 „$5/6$“ (`glibc-2.7`), Blowfish „$2a$“, ...

`crypt(3)`

Položky `/etc/passwd` (3)

- UID:
  - typově rozsah `unsigned int`,
  - prakticky (kompatibilita) $\langle 0; 32767 \rangle$.
  - Pro pseudouživatele se vyhrazuje prvních 100–1000 hodnot.
  - UID by měla být stejně jako už. jména jedinečná v rámci organizace.
  - Recyklace UID může způsobit potíže.

    (Obnova ze záloh, NFS.)
- GID:
  - Stejný rozsah jako UID.
  - Odkaz do souboru `/etc/group`.
  - Implicitní hodnota po přihlášení.

Položky `/etc/passwd` (4)

- GECOS:
  - *General Electric Comprehensive Operating System*, přihlašovací informace pro přenos úloh z UNIXových systémů na mainframe v Bell Labs.
  - Není to jen „komentář“, jak je běžně interpretován.
  - `finger` rozlišuje tyto položky oddělené čárkou:

    celé jméno, číslo kanceláře a budovy, telefon, soukromý telefon.
  - Možnost pole samostatně měnit (`chfn`; `chage`) bývá zakázána.
- Domácí adresář:
  - Při přihlášení je sem proveden `chdir`(2).
  - Nedostupný adresář: na terminálu může projít, `xdm` aj. ale ihned uživatele odhlásí (potřebují zapisovat).
  - Síťové domácí adresáře – server nemusí být dostupný.

Položky `/etc/passwd` (5)

- Přihlašovací (login) shell:
  - Není-li uveden, použije se implicitní, např. `/bin/sh`.
  - Uživatelsky volitelný (`chsh`) z povolených shellů `/etc/shells` (ne všechny distribuce to ale vynucují).

Řízená editace `/etc/passwd`

`chfn`

Co je povoleno/zakázáno měnit: `/etc/login.defs`

`CHFN_RESTRICT frwh`

`chfn -f` *fullname* `-r` *room* `-w` *work_phone* `-h` *home_phone* `-o` *other user*

`chage`

Uživatelé mohou obvykle jen zjistit svoje nastavení

`chage -l` *login*

Soubor /etc/shadow

Pokud je zapnuta podpora `/etc/shadow`, je nutné jej udržovat vzájemně konsistentní s `/etc/passwd`.

Řádky odpovídají záznamům v `/etc/passwd`.

```
novaj12:$1$Af3···Y:13291:0:99999:7:::
```

Položky:

1. Přihlašovací jméno, propojuje `/etc/passwd` s `/etc/shadow`,
2. hash hesla nebo „`!`“ nebo „`*`“,
3. datum poslední změny hesla,
4. minimální a
5. maximální počet dnů mezi změnami hesla,
6. počet dnů, po které bude uživatel varován, že musí heslo změnit,
7. počet dnů po vypršení platnosti hesla, po níž bude účet zablokován,
8. datum vypršení platnosti, neuvádí se v UNIXové sekundové konvenci, ale ve *dnech* od 1. ledna 1970,
9. vyhrazené (dnes prázdné) pole.

`pwconv`, `pwck`

Soubor /etc/group

Obsahuje jména skupin a výčet jejich členů.
Položky:

1. Jméno skupiny (bývalo doporučeno max. 8 znaků),
2. hash hesla (‘x’), není obvykle potřeba (`shadow`: ‘`*`’),
3. GID,
4. výčet uživatelských jmen oddělených čárkami.

```
root::0:
other::1:root
bin::2:root,daemon
sys::3:root,bin,adm
adm::4:root,daemon
```

`sg`, `newgrp`, `grpconv`, `grpck`

Přidávání uživatelů

`useradd`, `usermod`

Přidání uživatele:

1. Vytvořit domovský adresář.
2. Nakopírovat počáteční soubory s uživatelskými konfiguracemi `/etc/skel/*`: `.login`, `.bashrc`, `.vimrc`, `.xsession`, `.xinitrc`,...
3. Úprava `/etc/passwd` (`vipw`) a `/etc/shadow` (někde lze užít `vipw -s`).
4. Přidat členství ve skupinách, `/etc/group` (`vigr`).
5. Nastavit nového vlastníka a práva na celou strukturu domovského adresáře.
6. Vytvoření implicitního hesla, `# /bin/passwd` *username*.

Další možné operace:

1) Nastavit poštovní účet (alias) a adresář.

2) Nastavit diskové kvóty.

3) *Vše zkontrolovat.*

Odebírání uživatelů

Odebrání uživatele: `userdel`

1. Zamknout účet uživatele, aby se nemohl přihlašovat (`passwd -l`).
2. Zlikvidovat všechny dosud běžící procesy uživatele.
3. Projít adresáře, kam mohl uživatel zapisovat a smazat (zálohovat) jeho soubory, to se týká i jeho
   (a) souboru `crontab`,
   (b) úloh naplánovaných přes `at`,
   (c) tiskových úloh (`lpq; lprm`).
4. Smazat řádky v `/etc/passwd` a `/etc/shadow` (`vipw`).
5. Smazat členství ve skupinách, `/etc/group`.
6. Smazat (případně zazálohovat) domovský adresář.

Další možné operace:

1. Smazat poštovní účet (případně nastavit forwarding, smazat soubor s došlou poštou).
2. Smazat nastavení diskových kvót.