Zásuvné Autentifikační Moduly

Ondřej Caletka

O.Caletka@sh.cvut.cz
http://www.pslib.cz/caletka

Zásuvné Autentifikační Moduly

- Anglicky Pluggable Authentication Modules – zkratka PAM
- Implementace DCE-RFC 86.0, October 1995
- Používá RedHat a jeho klony
- V Gentoo USE flag “pam” - standardně zapnutý

Tradiční přístup

UNIXová hesla

- všemocný /etc/passwd
- soubor s (šifrovanými) hesly všech uživatelů čitelný všem
- šifra poměrně snadno prolomitelná

Stínová hesla

- hesla v /etc/shadow
- soubor shadow čitelný pouze pro roota
- přibyly možnosti vypršení hesel (man chage)

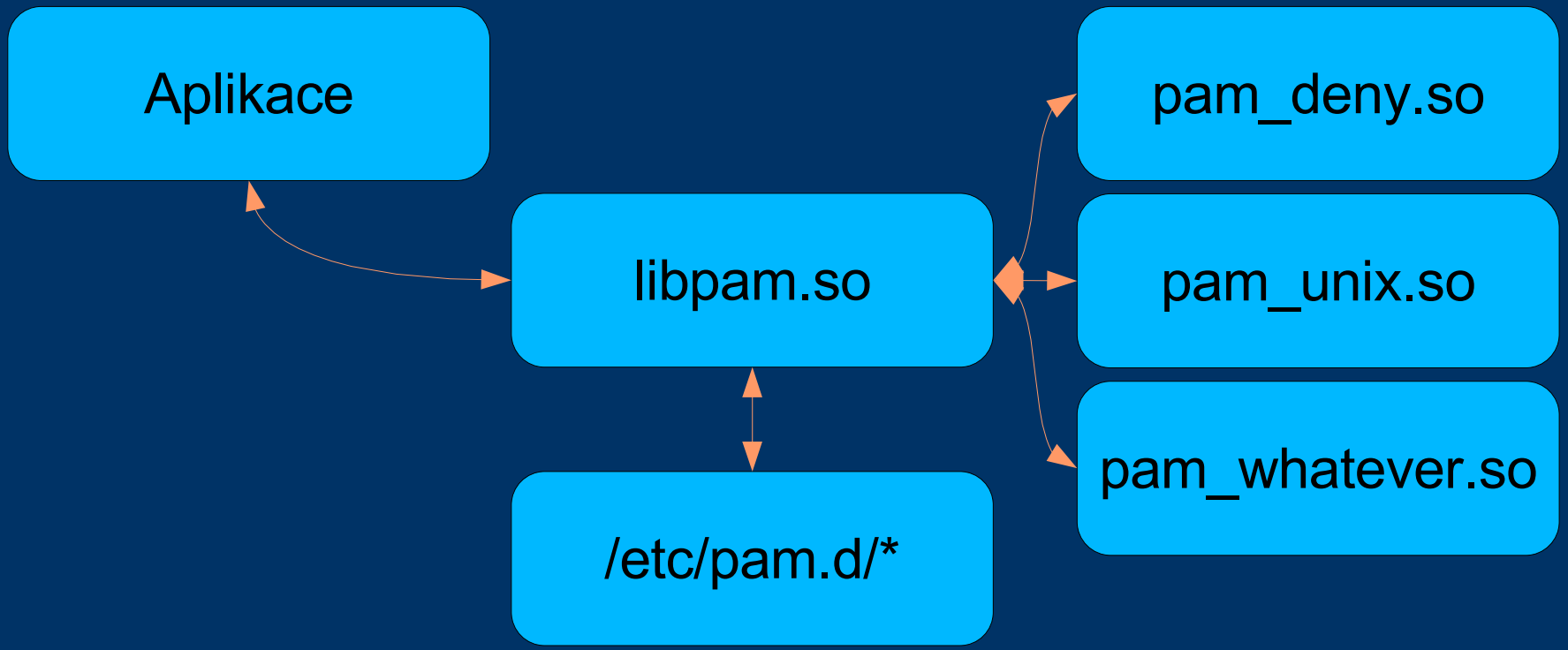
Tradiční přístup – společné znaky

- implementace v cílové aplikaci (např. login, su, passwd, chfn, chage) prostřednictvím knihovny

```
include “shadow.h”
```

- konfigurace prostřednictvím souboru login.defs
- při požadavku na jiný druh autentizace je nutno všechny aplikace, které autentizaci používají, překompilovat s novou knihovnou

Zásuvné Autentifikační Moduly - princip



Zásuvné Autentifikační Moduly - princip

- aplikace, která požaduje autentifikaci, požádá knihovnu libpam.so
- knihovna najde v /etc/pam.d/* konfigurační soubor, jehož název odpovídá názvu služby (např. login, su...)
- v tomto souboru knihovna zjistí, který modul z /lib/security/*.so je potřeba zavolat
- výstup z modulu se předá do původní aplikace

Zásuvné Autentifikační Moduly - výhody

- systém lze konfigurovat za chodu – není třeba pro změnu nastavení nic překompilovávat.
- existují moduly pro autentifikaci proti všemožným databázím (např. RADIUS, LDAP, Kerberos)
- spousta podpůrných modulů pro zvýšení bezpečnosti, nebo naopak pohodlí

Příklad - soubor /etc/pam.d/login

```
#%PAM-1.0

auth       required     /lib/security/pam_securetty.so
auth       required     /lib/security/pam_stack.so service=system-auth
auth       required     /lib/security/pam_nologin.so

account    required     /lib/security/pam_stack.so service=system-auth

password   required     /lib/security/pam_stack.so service=system-auth

session    required     /lib/security/pam_stack.so service=system-auth
```

- struktura:

<oblast> <důležitost> <jméno modulu> [parametry]

- Pokud je k jedné oblasti víc řádků, vyhodnocují se v napsaném pořadí

Syntax konfiguračního souboru - <oblast>

- auth – požívá se pro ověření, zda daný uživatel má potřebné oprávnění (zpravidla zadáním jména a hesla)
- account – zjišťuje, zda daný uživatel existuje, má vypršené heslo, ...
- password – používá se, pokud chce uživatel změnit ověřovací údaje (zpravidla heslo)
- session – ověřuje se před spuštěním shellu a po jeho ukončení

Syntax konfiguračního souboru - <důležitost>

- required – modul musí skončit úspěšně
- optional – na výstupu modulu nezáleží
- sufficient – pokud modul skončí úspěšně, další se už nevolají a je vyhlášen úspěch
- requisite – pokud modul neskončí úspěšně, další se nevolají a je vyhlášen neúspěch

Praktický příklad

- Protože obvykle chceme nastavovat vše na jednom místě, většina souborů v /etc/pam.d jen odkazuje na soubor *system-auth*, ten je tedy nejzajímavější:

Praktický příklad – soubor system-auth

```
auth        required      pam_env.so
auth        sufficient    pam_unix.so likeauth nullok
auth        required      pam_deny.so

account     required      pam_unix.so

password    required      pam_cracklib.so difok=2
  minlen=8 dcredit=2 ocredit=2 retry=3
password    sufficient    pam_unix.so nullok md5
  shadow use_authtok
password    required      pam_deny.so

session     required      pam_limits.so
session     required      pam_unix.so
session     optional      pam_ssh.so debug
```

Vychytávky s PAM

- *pam_xauth* – jednoduchý přenos práv k X serveru
- *pam_console* – práva pro uživatele „na konzoli“
- *pam_wheel* – test na uživatele „u volantu“
- *pam_ssh* – autentifikace pomocí ssh klíčů

pam_xauth - úvod

- Aby mohl uživatel spouštět aplikace na X serveru, musí mít v souboru ~/.Xauthority správnou „sušenku,“ kterou při autentifikaci předloží
- Co když použiju su? Jaktože i uživatel, na kterého jsem se změnil může spouštět grafické aplikace?
  - Je tam totiž xauth, který vše potřebné zařídí...

pam_xauth - použití

- V souboru /etc/pam.d/su je mimo jiné:

```
session    optional                pam_xauth.so
```

- Díky tomu se při každém použití su předají „sušenky“

pam_console - úvod

- Pokud si v Gentoo chcete pustit zvuk, musíte být ve skupině audio, abyste mohl zapisovat do /dev/snd*
- Každý, kdo je ve skupině audio může pouštět zvuk – kdykoli odkudkoli (třeba přes ssh)
- Chceme, aby ho mohl spouštět jen a jen ten, kdo u počítače skutečně sedí.

pam_console – co dělá

- Po úspěšné autentifikaci (v hladině *session*) zkontroluje, zda je uživatel *jediný* na konzoli, buď textové, nebo X
- Pokud ano, změní práva určitých souborů podle konfiguračního souboru
- Po ukončení změní práva zpět na defaultní

pam_console – co dělá

- Modul je také možné volat v hladině auth
- V takovém případě zkontroluje, zda je v adresáři /etc/security/console.apps/ soubor se stejným názvem jako je název služby, která autentifikaci požadovala
- Pokud ano, a zároveň je daný uživatel přihlášen na konzoli, vyvolá úspěch
- Používá Red Hat pro vypínání počítače běžnými uživateli přes *console_helper*

pam_console – konfigurační soubor

- /etc/security/console.perms

```
# file classes -- these are regular expressions
<console>=tty[0-9][0-9]* vc/[0-9][0-9]* :[0-9]\.[0-9] :[0-9]
<xconsole>=:[0-9]\.[0-9] :[0-9]

# device classes -- these are shell-style globs
<serial>=/dev/ttyS*
<floppy>=/dev/fd[0-1]* \
         /dev/floppy/* /mnt/floppy*
<sound>=/dev/dsp* /dev/audio* /dev/midi* \

# permission definitions
<console>  0660 <serial>     0660 root.tty
<console>  0660 <floppy>     0660 root.floppy
<console>  0660 <sound>      0660 root.audio
```

pam_console - dodatky

- Kromě příkladu se zvukem se dá také použít pro omezení možnosti připojit určitý filesystem

```
/dev/fd0  /mnt/floppy   auto     noauto,owner    0 0
```

- Pokud se vám zdá, že se vám práva ke speciálním souborům záhadně mění, popř. ztrácí, bude za to nespíš moci špatně zkonfigurovaný pam_console. Ten se totiž spouští jako úplně poslední a tak vlastně rozhoduje o konečném nastavení práv.

pam_wheel - úvod

Určitě se vám stalo, že jste omylem napsali heslo do příkazového řádku...

...mně mockrát...

...a přesto, že je tu jedno elegantní řešení...

...hesla prostě NEPSAT!

pam_wheel - úvod

- pam_wheel je modul, který testuje, zda uživatel je zařazen do skupiny wheel – tedy vyvolených uživatelů – na půl rootů
- Původní účel je naprosto šílený – su může provádět jen člen skupiny wheel – v Gentoo defaultní nastavení.
- Po převrácení na ruby už smysl dává – su může provádět každý, vyvolení nemusí psát heslo

pam_wheel - příklad

- soubor /etc/pam.d/su

```
auth       sufficient   pam_rootok.so
# Uncomment this to allow users in the wheel group to su without
# entering a passwd.
auth       sufficient   pam_wheel.so use_uid trust

# Comment this to allow any user, even those not in the 'wheel'
# group to su
#auth        required     pam_wheel.so use_uid

auth       include                 system-auth
account    include                 system-auth
password   include                 system-auth
session    include                 system-auth
session    required                pam_env.so
session    optional                pam_xauth.so
```

pam_ssh - úvod

- Modul pam_ssh slouží k ověřování uživatelů pomocí hesla k ssh klíči
- Pokud zadané heslo dekryptuje ssh klíč v uživatelově domácím adresáři, je autentifikace úspěšná => nepoužívá se shadow
- Navíc se v oblasti session spustí ssh-agent a přidají se do něj všechny klíče, které zadané heslo úspěšně dekryptovalo

pam_ssh - příklad

- soubor /etc/pam.d/system-auth

```
auth        required      pam_env.so
#auth       sufficient    pam_ssh.so
auth        sufficient    pam_unix.so likeauth nullok
auth        sufficient    pam_ssh.so use_first_pass debug
auth        required      pam_deny.so

---

session     required      pam_limits.so
session     required      pam_unix.so
session     optional      pam_console.so
#session     optional     pam_xauth.so
session     optional      pam_ssh.so debug
```

pam_ssh - dodatky

- Po rozchození a ověření funkčnosti je vhodné zamknout uživatelovo unixové heslo.
- pam_ssh nelze používat v oblasti password, tedy uživatelé nebudou moci používat program passwd pro změnu hesla

Další zajímavé moduly

- pam_limits
- pam_securetty
- pam_time
- pam_usb

Zdroje informací

- Google: PAM
- man pam
- less /usr/share/doc/pam-x.xx/modules/*.gz

Závěr

- Děkuji za pozornost
- Prostor pro dotazy
- Praktické předvedení popisovaných modulů
- Zavaděč GRUB zbude-li čas a chuť