PAM
Pluggable Authentication Modules

David Morgan

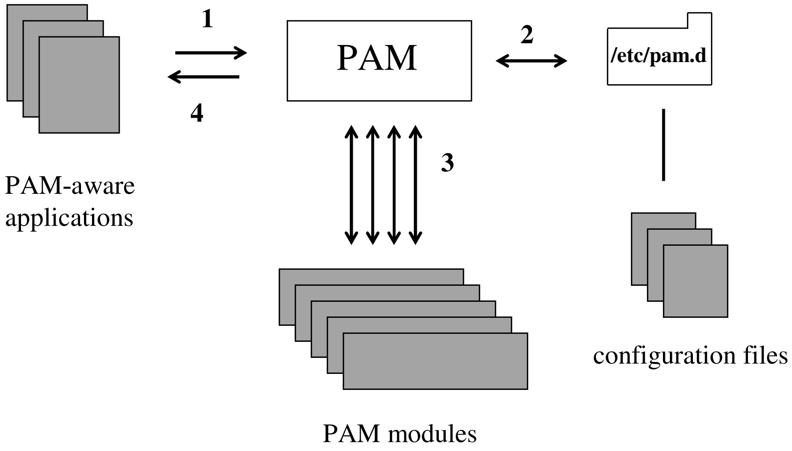
What’s PAM?

- a group of programs that do authentication
- called by other, PAM-aware programs as a service
- to delegate the authentication task

Hypothetical example

- program X
- uses PAM’s module /lib/security/foo
- configured by its config file /etc/pam.d/foo
- to perform authentication action Y

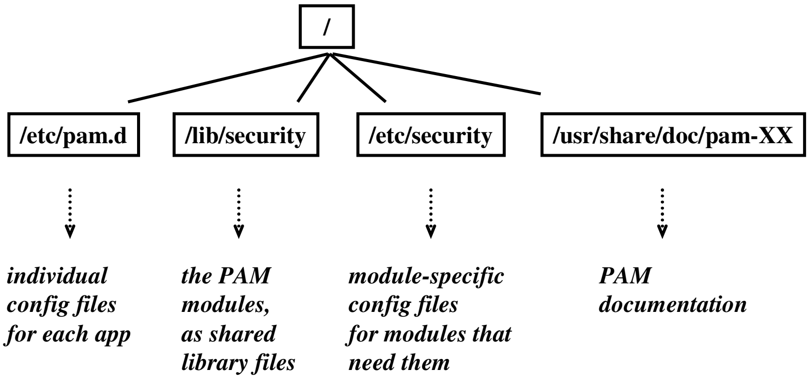
PAM architecture



Operation sequence

- app calls PAM (1)
- PAM reads app's PAM config file (2)
- PAM calls PAM modules as listed in the file (3)
  - each succeeds or fails
- PAM itself succeeds or fails, depending on the modules' outcomes
  - returns its overall outcome to app (4)
- app proceeds (if success) or terminates (if failure)

Default directories and files



/

/etc/pam.d

individual config files for each app

/lib/security

the PAM modules, as shared library files

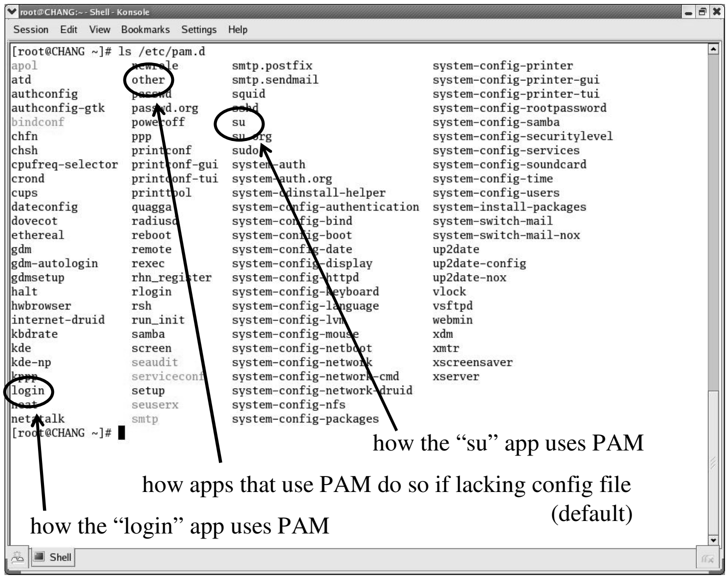
/etc/security

module-specific config files for modules that need them

/usr/share/doc/pam-XX

PAM documentation

PAM config, per app


root@CHANG:~ - Shell - Konsole
Session Edit View Bookmarks Settings Help
Shell

```
[root@CHANG ~]# ls /etc/pam.d
apol              newrole        smtp.postfix                  system-config-printer
atd               other          smtp.sendmail                 system-config-printer-gui
authconfig        passwd         squid                         system-config-printer-tui
authconfig-gtk    passwd.org     sshd                          system-config-rootpassword
bindconf          poweroff       su                            system-config-samba
chfn              ppp            su.org                        system-config-securitylevel
chsh              printconf      sudo                          system-config-services
cpufreq-selector  printconf-gui  system-auth                   system-config-soundcard
crond             printconf-tui  system-auth.org               system-config-time
cups              printtool      system-cdinstall-helper       system-config-users
dateconfig        quagga         system-config-authentication  system-install-packages
dovecot           radiusd        system-config-bind            system-switch-mail
ethereal          reboot         system-config-boot            system-switch-mail-nox
gdm               remote         system-config-date            up2date
gdm-autologin     rexec          system-config-display         up2date-config
gdmsetup          rhn_register   system-config-httpd           up2date-nox
halt              rlogin         system-config-keyboard        vlock
hwbrowser         rsh            system-config-language        vsftpd
internet-druid    run_init       system-config-lvm             webmin
kbdrate           samba          system-config-mouse           xdm
kde               screen         system-config-netboot         xmtr
kde-np            seaudit        system-config-network         xscreensaver
kppp              serviceconf    system-config-network-cmd     xserver
login             setup          system-config-network-druid
neat              seuserx        system-config-nfs
netatalk          smtp           system-config-packages
[root@CHANG ~]# 
```

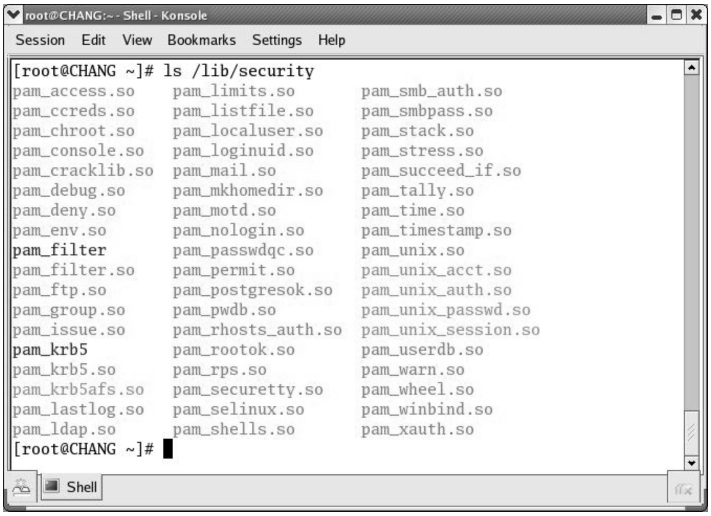
how the "su" app uses PAM

how apps that use PAM do so if lacking config file (default)

how the "login" app uses PAM

© David Morgan 2006

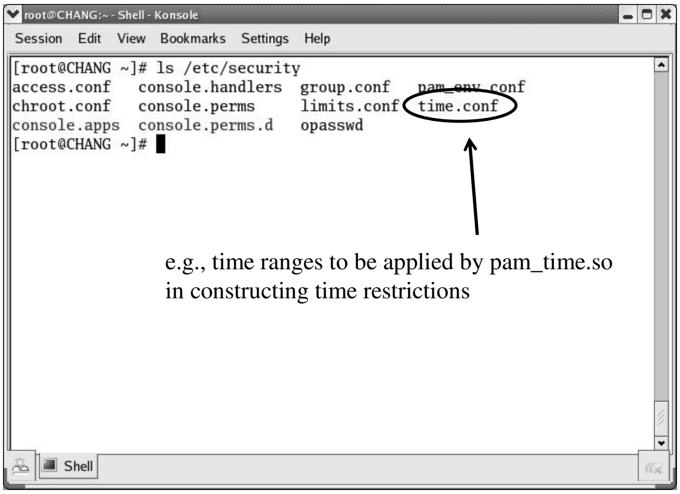
PAM modules themselves (code)


root@CHANG:~ - Shell - Konsole
Session Edit View Bookmarks Settings Help
Shell

```
[root@CHANG ~]# ls /lib/security
pam_access.so     pam_limits.so       pam_smb_auth.so
pam_ccreds.so     pam_listfile.so     pam_smbpass.so
pam_chroot.so     pam_localuser.so    pam_stack.so
pam_console.so    pam_loginuid.so     pam_stress.so
pam_cracklib.so   pam_mail.so         pam_succeed_if.so
pam_debug.so      pam_mkhomedir.so    pam_tally.so
pam_deny.so       pam_motd.so         pam_time.so
pam_env.so        pam_nologin.so      pam_timestamp.so
pam_filter        pam_passwdqc.so     pam_unix.so
pam_filter.so     pam_permit.so       pam_unix_acct.so
pam_ftp.so        pam_postgresok.so   pam_unix_auth.so
pam_group.so      pam_pwdb.so         pam_unix_passwd.so
pam_issue.so      pam_rhosts_auth.so  pam_unix_session.so
pam_krb5          pam_rootok.so       pam_userdb.so
pam_krb5.so       pam_rps.so          pam_warn.so
pam_krb5afs.so    pam_securetty.so    pam_wheel.so
pam_lastlog.so    pam_selinux.so      pam_winbind.so
pam_ldap.so       pam_shells.so       pam_xauth.so
[root@CHANG ~]# 
```

© David Morgan 2006

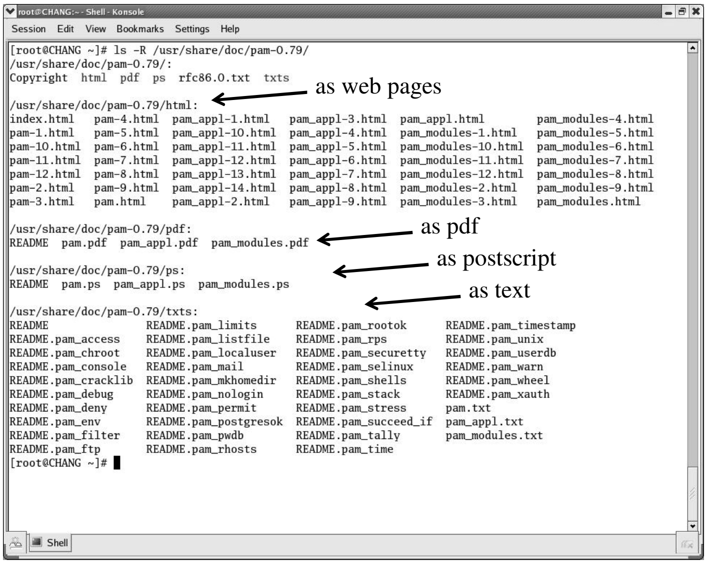
Config for certain PAM modules


root@CHANG:~ - Shell - Konsole
Session Edit View Bookmarks Settings Help
Shell

```
[root@CHANG ~]# ls /etc/security
access.conf   console.handlers  group.conf   pam_env.conf
chroot.conf   console.perms     limits.conf  time.conf
console.apps  console.perms.d   opasswd
[root@CHANG ~]#
```

e.g., time ranges to be applied by pam_time.so
in constructing time restrictions

PAM documentation


root@CHANG:~ - Shell - Konsole
Session Edit View Bookmarks Settings Help
Shell

```
[root@CHANG ~]# ls -R /usr/share/doc/pam-0.79/
/usr/share/doc/pam-0.79/:
Copyright  html  pdf  ps  rfc86.0.txt  txts

/usr/share/doc/pam-0.79/html:
index.html   pam-4.html  pam_appl-1.html   pam_appl-3.html  pam_appl.html        pam_modules-4.html
pam-1.html   pam-5.html  pam_appl-10.html  pam_appl-4.html  pam_modules-1.html   pam_modules-5.html
pam-10.html  pam-6.html  pam_appl-11.html  pam_appl-5.html  pam_modules-10.html  pam_modules-6.html
pam-11.html  pam-7.html  pam_appl-12.html  pam_appl-6.html  pam_modules-11.html  pam_modules-7.html
pam-12.html  pam-8.html  pam_appl-13.html  pam_appl-7.html  pam_modules-12.html  pam_modules-8.html
pam-2.html   pam-9.html  pam_appl-14.html  pam_appl-8.html  pam_modules-2.html   pam_modules-9.html
pam-3.html   pam.html    pam_appl-2.html   pam_appl-9.html  pam_modules-3.html   pam_modules.html

/usr/share/doc/pam-0.79/pdf:
README  pam.pdf  pam_appl.pdf  pam_modules.pdf

/usr/share/doc/pam-0.79/ps:
README  pam.ps  pam_appl.ps  pam_modules.ps

/usr/share/doc/pam-0.79/txts:
README                README.pam_limits      README.pam_rootok      README.pam_timestamp
README.pam_access     README.pam_listfile    README.pam_rps         README.pam_unix
README.pam_chroot     README.pam_localuser   README.pam_securetty   README.pam_userdb
README.pam_console    README.pam_mail        README.pam_selinux     README.pam_warn
README.pam_cracklib   README.pam_mkhomedir   README.pam_shells      README.pam_wheel
README.pam_debug      README.pam_nologin     README.pam_stack       README.pam_xauth
README.pam_deny       README.pam_permit      README.pam_stress      pam.txt
README.pam_env        README.pam_postgresok  README.pam_succeed_if  pam_appl.txt
README.pam_filter     README.pam_pwdb        README.pam_tally       pam_modules.txt
README.pam_ftp        README.pam_rhosts      README.pam_time
[root@CHANG ~]#
```

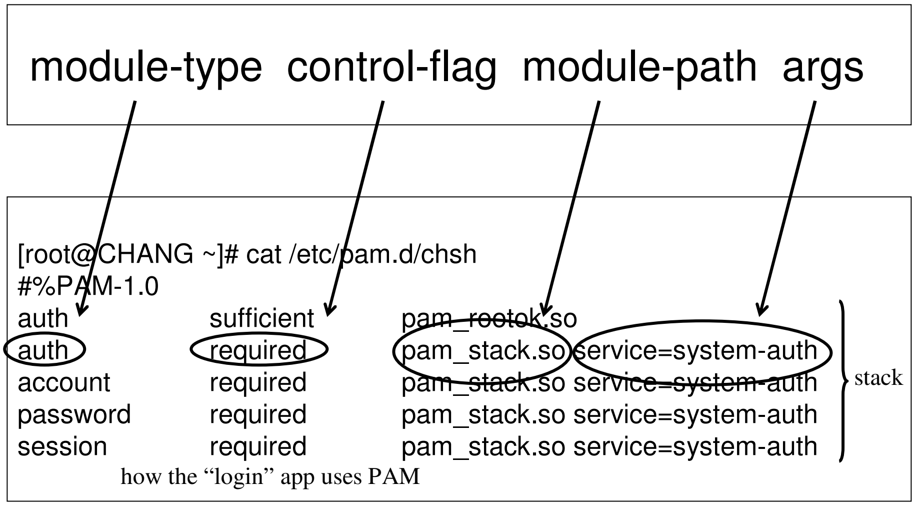
as web pages

as pdf

as postscript

as text

config file line item syntax



module-type control-flag module-path args

```
[root@CHANG ~]# cat /etc/pam.d/chsh
#%PAM-1.0
auth            sufficient      pam_rootok.so
auth            required        pam_stack.so service=system-auth
account         required        pam_stack.so service=system-auth
password        required        pam_stack.so service=system-auth
session         required        pam_stack.so service=system-auth
```

how the “login” app uses PAM

Syntax: the module types

- auth – establishes who the user is (e.g. password)
- account – non-authentication account management (e.g. check time-of-day restriction)
- session – any pre- (e.g. mounting) or post- (e.g. logging actions
- password – update user’s authentication token

Syntax: the control flags

The control-flag is used to indicate how the PAM library will react to the success or failure of the module it is associated with. Since modules can be *stacked* (modules of the same type execute in series, one after another), the control-flags determine the relative importance of each module. The application is not made aware of the individual success or failure of modules listed in the `/etc/pam.conf' file. Instead, it receives a summary *success* or *fail* response from the **Linux-PAM** library. The order of execution of these modules is that of the entries in the /etc/pam.conf file; earlier entries are executed before later ones…. The…syntax for the control-flag is a single keyword defined to indicate the severity of concern associated with the success or failure of a specific module. There are four such keywords: required, requisite, sufficient, optional….

Syntax: the control flags

- required – this test must pass for app to proceed, further tests conducted but then app terminates
- requisite – same, but app terminates immediately
- sufficient – failure is OK, success dispenses with further tests of same type
- optional – app proceeding doesn't depend on this test, unless there are no other successful tests

What some modules do

- pam_cracklib – evaluates password strength
- pam_issue – add text to login prompt
- pam_nologin – determines if /etc/nologin exists
- pam_rootok – determines if user is root
- pam_securetty – determines if current tty listed in /etc/securetty
- pam_time – checks time against allowable times from /etc/security/time.conf

time.conf line item syntax

```
service  ttys  users  time-ranges
```

login ; tty* & !ttyp* ; !root ; !Al0000-2400

all users except for root are denied access to console-login at all times.

games ; * ; !waster ; Wd0000-2400 | Wk1800-0800

games (configured to use Linux-PAM) are only to be accessed out of working hours. This rule does not apply to the user waster.

info

- /usr/share/doc/pam-0.79/pdf/pam.pdf
  or
  /usr/share/doc/pam-0.79/html/index.html
- http://www.kernel.org/pub/linux/libs/pam/index.html