CU CS 4995-03: Internet Systems Programming

Unix System Data

Overview

- password file
-

Authentication

- user-supplied password
- other mechanisms:
- proliferation of non-login identities:
  - `.htaccess` for web
  - CVS
  - user directories (LDAP)
  - ...

Password File

`username:passwd:UID:GID:full_name:directory:shell`

user name	`pw_name`
encrypted password	`pw_passwd`
numerical user id	`pw_uid`
numerical group id	`pw_gid`
comment field	`pw_gecos`
initial directory	`pw_dir`
initial shell	`pw_shell`

flat file or `gdbm` (FreeBSD)

Looking up Users

```
#include <sys/types.h>
#include <pwd.h>

struct passwd *getpwuid(uid_t uid);
struct passwd *getpwnam(const char *name);
```

- also `getpwent()` as iterator

Shadow Password Files

- `/etc/passwd`: world-readable $\longrightarrow$ bad
- `/etc/shadow`: use '*' in `passwd`, only root readable
- doesn't help much if NIS is being used (`ypcat passwd`)

Group File (/etc/group)

```
struct group {
  char  *gr_name;
  char  *gr_passwd;
  gid_t gr_gid;
  char  **gr_mem;
};
```

- users can belong to more than one group

Primary User Groups

```
#include <sys/types.h>
#include <grp.h>

struct group *getgrgid(gid_t gid);
struct group *getgrname(const char *name);

struct group *getgrent(void);
```

Supplementary Groups

```
int getgroups(int gidsetsize, gid_t grouplist[]);
int setgroups(int ngroups, const gid_t grouplist[])
```

Other Configuration Files

`/etc/nsswitch.conf` governs lookups

`/etc/services`	network servers
`/etc/protocols`	network protocols
`/etc/networks`	local networks
`/etc/hosts`	host names (don't use)

NT: registry
also: `/etc/rc.*` for startup

PAM (Pluggable Authentication Module)

- old: just passwords
- PAM = loadable libraries for authentication
- can be configured (/etc/pam.conf)

Login Accounting

Record login and logout, `utmp` for current users

```
struct utmpx {
  char    ut_user[32];   /* user login name */
  char    ut_id[4];      /* inittab id */
  char    ut_line[32];   /* device name */
  pid_t   ut_pid;        /* process id */
  short   ut_type;       /* type of entry */
  struct timeval ut_tv;  /* time entry was made */
  int ut_session;        /* session ID */
  char    ut_host[257];  /* remote host name */
};
```

System Identification

`uname`:

```
struct utsname {
 char sysname[];
 char nodename[];
 char release[];
 char version[];
 char machine[];
}
```

`gethostname`