Linux users and groups

Accounts and groups

CONCEPTS

usermod -G alone replaces all supplementary groups. Always write -aG to add one. Group changes apply from the next login.

useradd -m defaults from login.defs

↓ writes

/etc/passwd name, UID, home, shell

↓ plus

/etc/shadow hash, aging, expiry

↓ plus

/etc/group own group, same name

↓ copies

/home/<user> from /etc/skel

ID ranges (typical)

Range	Used for
0	root
1-999	system users (-r)
1000+	people
65534	nobody

WHO IS WHO

id
Your UID, GID and groups

id *<user>*
Same for another user

groups *<user>*
List a user's groups

whoami
Print your effective user name

getent passwd *<user>*
Look up an account entry

getent group *<group>*
Show a group and its members

who
Users logged in now

w
Logged in users and what they run

last -n 10
Recent logins

CREATE USERS

useradd -m *<user>*
Create a user with a home dir

useradd -m -s /bin/bash *<user>*
Set the login shell too

CREATE USERS (cont.)

useradd -m -G *<g1>*,*<g2>* *<user>*
Add supplementary groups

useradd -m -u *<uid>* *<user>*
Pick a fixed UID

useradd -g *<group>* *<user>*
Set the primary group

useradd -r -s /usr/sbin/nologin *<user>*
Create a system user for a service

Debian and Ubuntu also offer adduser, an interactive wrapper that creates the home and asks for a password. Scripts should use useradd.

MODIFY USERS

usermod -aG *<group>* *<user>*
Add to a group, keep the rest

usermod -G *<g1>*,*<g2>* *<user>*
Replace all supplementary groups

gpasswd -d *<user>* *<group>*
Remove a user from one group

usermod -s *<shell>* *<user>*
Change the login shell

usermod -L *<user>*
Lock the password

usermod -U *<user>*
Unlock the password

usermod -e *<YYYY-MM-DD>* *<user>*
Set the account expiry date

usermod -l *<new>* *<old>*
Rename the login

usermod -d *<dir>* **-m** *<user>*
Move the home directory

DELETE USERS

userdel *<user>*
Delete the account, keep files

userdel -r *<user>*
Delete account, home and mail

GROUPS

groupadd *<group>*
Create a group

groupadd -g *<gid>* *<group>*
Create a group with a fixed GID

GROUPS (cont.)

gpasswd -a *<user>* *<group>*
Add a user to a group

PASSWORDS AND AGING

passwd
Change your own password

passwd *<user>*
Set another user's password

passwd -S *<user>*
Password status: P, L or NP

passwd -e *<user>*
Force a change at next login

chage -l *<user>*
Show aging and expiry info

chage -E *<YYYY-MM-DD>* *<user>*
Expire the account on a date

Lock or expire?

Command	Blocks
passwd -l	password logins
usermod -L	password logins
usermod -e 1	all logins, incl. SSH keys
passwd -e	nothing; forces a new password

FILES

/etc/passwd fields

#	Field
1	login name
2	x = hash in shadow
3	UID
4	primary GID
5	comment (GECOS)
6	home directory
7	login shell

/etc/group
name:x:GID:member,member

SWITCH USER

su - *<user>*
Log in as user, full environment

sudo -iu *<user>*
Login shell as user via sudo

sudo -u *<user>* *<cmd>*
Run one command as user

su needs the target's password; sudo needs yours plus a sudoers rule. Prefer sudo, keep root's password locked.