



O15

# Linux User Administration

Pete Davis, Sr. Instructor, IBM

**IBM @server xSeries**  
**Technical Conference**

Aug. 9 - 13, 2004

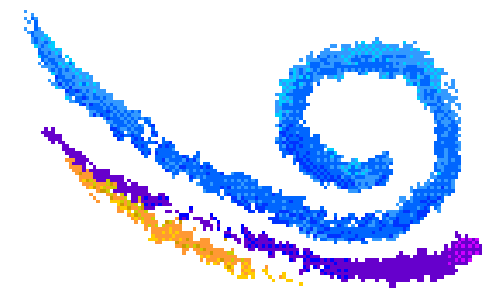
Chicago, IL

# Objectives

---

After completing this unit, students should be able to:

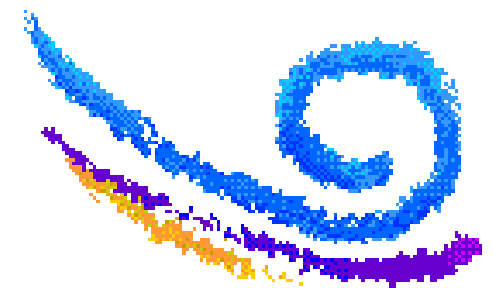
- Add, change and delete users and groups
- Manage user passwords
- Discuss authentication and verification
- Limit user space using Quotas
- Understand all the permission bits



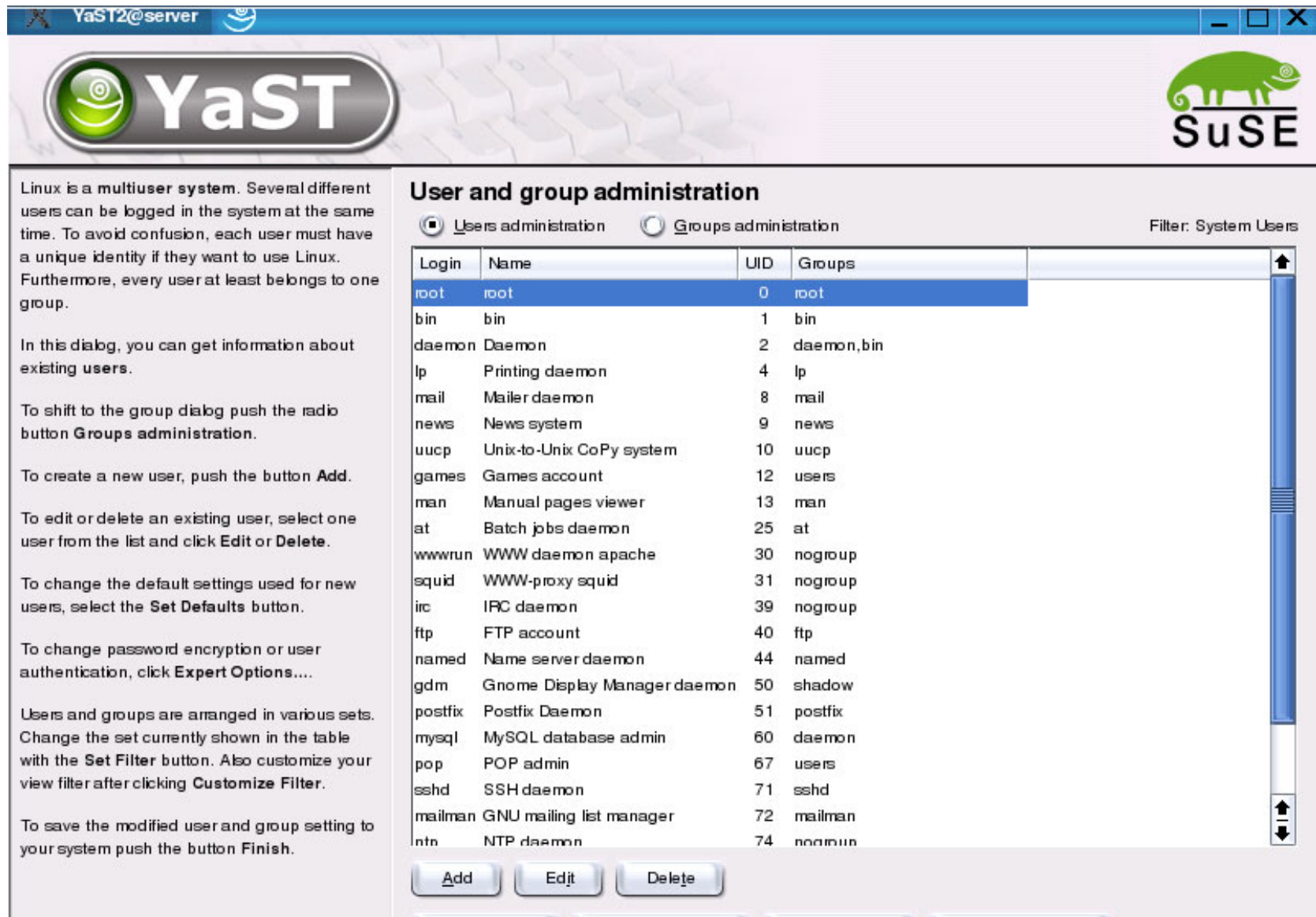
# Command Line User Tools

---

- Add a user account
  - # ***useradd username*** (newusers)
  - # ***password username***
- Delete a user account
  - # ***userdel username***
  - # ***rm -r /home / username***
  - or
  - # ***userdel -r username***
- Change a user account
  - # ***usermod -G othergroups username***
- Locking and unlocking a user account
  - # ***usermod -L username***
  - # ***usermod -U username***



# YaST User admin



The screenshot shows the YaST User and group administration interface. The window title is "YaST2@server". The YaST logo is on the left, and the SuSE logo is on the right. The main content area is titled "User and group administration" and has two radio buttons: "Users administration" (selected) and "Groups administration". A filter "Filter: System Users" is visible. A table lists system users with columns for Login, Name, UID, and Groups. Below the table are buttons for "Add", "Edit", and "Delete".

Linux is a multiuser system. Several different users can be logged in the system at the same time. To avoid confusion, each user must have a unique identity if they want to use Linux. Furthermore, every user at least belongs to one group.

In this dialog, you can get information about existing users.

To shift to the group dialog push the radio button **Groups administration**.

To create a new user, push the button **Add**.

To edit or delete an existing user, select one user from the list and click **Edit** or **Delete**.

To change the default settings used for new users, select the **Set Defaults** button.

To change password encryption or user authentication, click **Expert Options**....

Users and groups are arranged in various sets. Change the set currently shown in the table with the **Set Filter** button. Also customize your view filter after clicking **Customize Filter**.

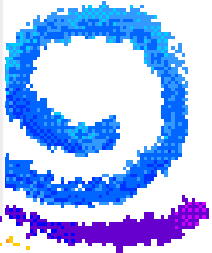
To save the modified user and group setting to your system push the button **Finish**.

### User and group administration

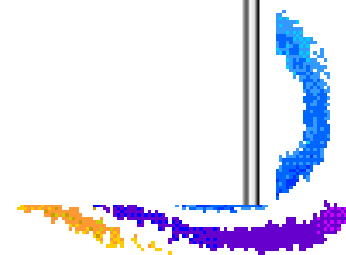
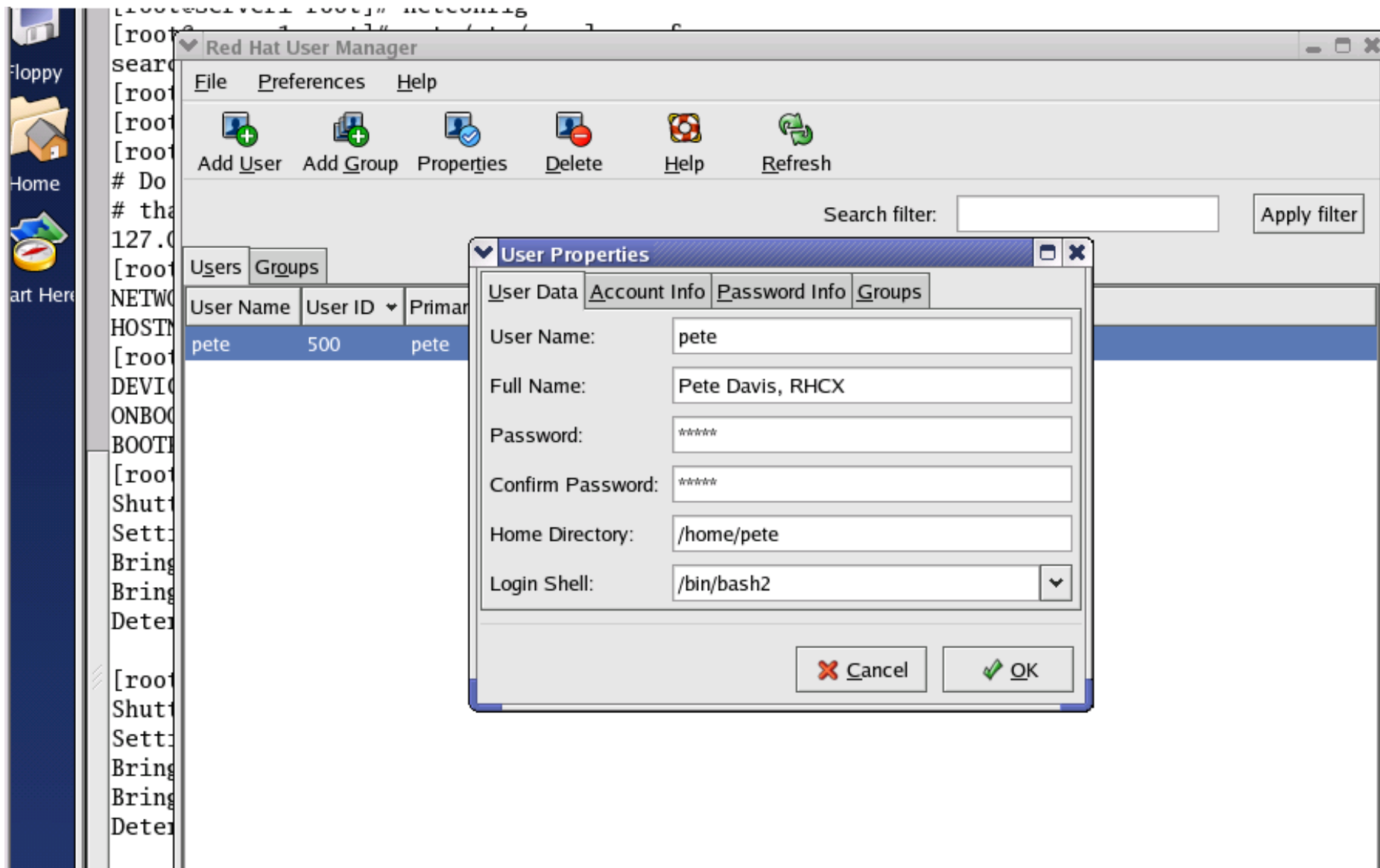
Users administration    Groups administration   Filter: System Users

Login	Name	UID	Groups
root	root	0	root
bin	bin	1	bin
daemon	Daemon	2	daemon,bin
lp	Printing daemon	4	lp
mail	Mailer daemon	8	mail
news	News system	9	news
uucp	Unix-to-Unix CoPy system	10	uucp
games	Games account	12	users
man	Manual pages viewer	13	man
at	Batch jobs daemon	25	at
wwwrun	WWW daemon apache	30	nogroup
squid	WWW-proxy squid	31	nogroup
irc	IRC daemon	39	nogroup
ftp	FTP account	40	ftp
named	Name server daemon	44	named
gdm	Gnome Display Manager daemon	50	shadow
postfix	Postfix Daemon	51	postfix
mysql	MySQL database admin	60	daemon
pop	POP admin	67	users
sshd	SSH daemon	71	sshd
mailman	GNU mailing list manager	72	mailman
ntn	NTP daemon	74	nogroup

**Add**   **Edit**   **Delete**



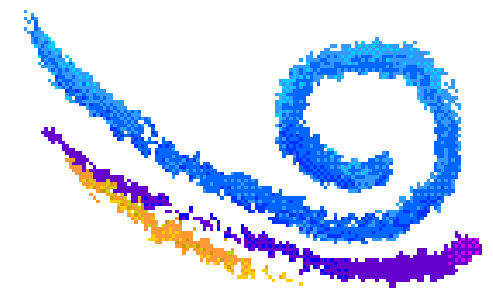
# redhat-config-users



# User Private Groups

---

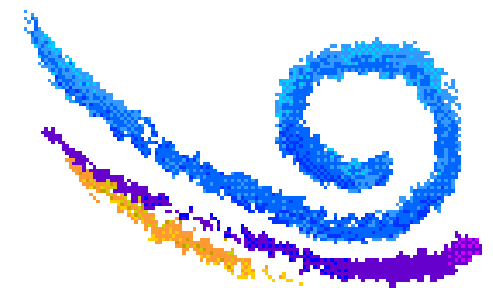
- User's primary group is the same name as the user
- Restricts write access to files and directories created by the user by other users
- The traditional UNIX unmask of 022 is not used: instead Linux uses unmask of 002 for users
- Advantage: New files do not belong to a public group
- Disadvantage: May encourage making files "world"



# /etc/skel

---

- Directory with skeleton files that users should have in their home directory.
- After creating the user account, the files in /etc/skel are copied to the home directory of that user
- You can place files here which you want every user to have in their home directory



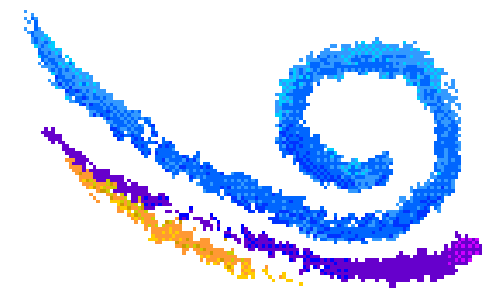
# Files Associated with User Login

---

- /etc/profile
- /etc/profile.d/\*.sh
- \$HOME/.bash\_profile (or .profile, .login)
- \$HOME/.bashrc
- /etc/bashrc

## Non-login shell scripts

- /etc/profile.d/\*.sh
- \$HOME/.bashrc
- /etc/bashrc



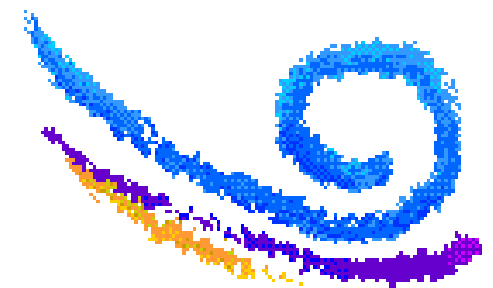


# Switching Accounts

---

***# su [-] [username] { -c command }***

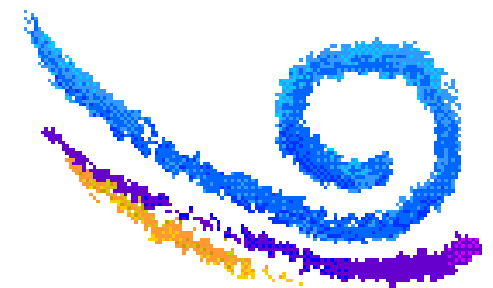
- "root" is the default username
- the dash recreates a login shell environment



# Command Line Group Tools

---

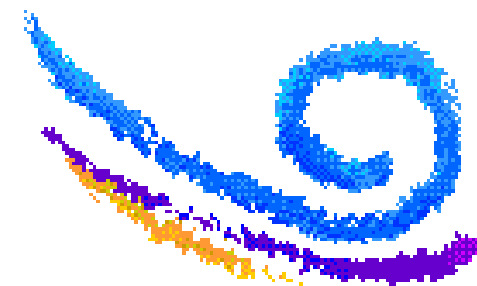
- Add a group  
*# groupadd -g some\_number groupname*
- Delete a group  
*# groupdel groupname*
- Change a group  
*# groupmod -n new\_name groupname*



# Passwords

---

- Change a user's password with:  
***# passwd user***
- Checked for strength
  - Dictionary Check
  - Minimum length
- Stored in /etc/shadow
- Options for MD5 and 'shadow' should be used
- To generate a random password:  
***# mkpasswd [username]***

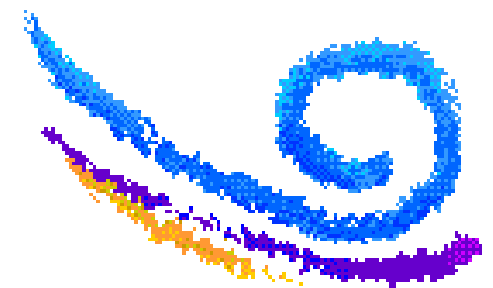


# Password Aging Policies

---

- By default, passwords do not expire
- Use the chage command (or gui tool) to set:
  - min/max days
  - lastday, warn days
  - inactive, expiredate

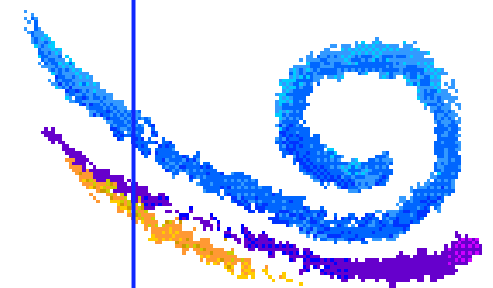
`chage [options] username`



# /etc/passwd

---

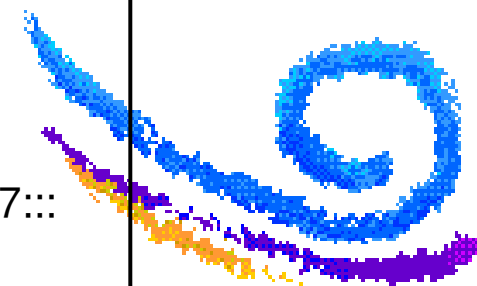
```
[root@hostname /root]# cat /etc/passwd
root:x 0 0:root:/root:/bin/bash
bin:x:l:l:bin:/bin:
daemon x 2:2:daemon:/sbin:
adm:x:3:4:adm:/var/adm:
lp:x 4 7 lp:/var/spool/lpd:
sync x 5 0:sync:/sbin:/bin/sync
shutdown x:6:0:shutdown:/sbin:/sbin/shutdown
halt x 7 0:halt:/sbin:/sbin/halt
mail x 8 12:mail:/var/spool/mail:
news x 9 13:news:/var/spool/news:
uucp x 10:14:uucp:/var/spool/uucp:
operator x:11:0:operator:/root:
games : x: 12 :100 : games : /usr/games :
gopher:x 13:30:gopher:/usr/lib/gopher-data:
ftp:x:14 50:FTP User:/home/ftp:
nobody:x 99:99:Nobody:/:
xfs:x:100:233:X Font Server:/etc/X11/fs:/bin/false
tuxl:x:501:501:Tux the Penguin (1):/home/tuxl:/bin/bash
tux2:x:502:502:Tux the Penguin (2):/home/tux2:/bin/bash
```



# /etc/shadow

---

```
[root@hostname /root]# cat /etc/shadow
root:$1$2psUS/s.$R8ZewWuXql1Z43wsFfalA.:12578:0:99999:8:::
bin:*:12578:0:99999:7:::
daemon *:12578:0:99999:7:::
adm:*:12578:0:99999:7:::
lp:*:12578:0:99999:7:::
sync:*:12578:0:99999:7:::
shutdown:*:12578:0:99999:7:::
halt:*:12578:0:99999:7:::
mail:*:12578:0:99999:7:::
news:*:12578:0:99999:7:::
uucp:*:12578:0:99999:7:::
operator:*:12578:0:99999:7:::
games:*:12578:0:99999:7:::
gopher:*:12578:0:99999:7:::
ftp:*:12578:0:99999:7:::
nobody:*:12578:0:99999:7:::
xfs:*:12578:0:99999:7:::
tuxl:$1$IWa.l8Jz$sk/q1/eGUDLOYuaGqsJzX0:12579:0:99999:7:::
tux2:$1$ISXXp7NHE$q6SiLdnLnc0Kx9Hn/aSU.Q/:12579:0:99999:7:::
```



# /etc/shadow notes ...

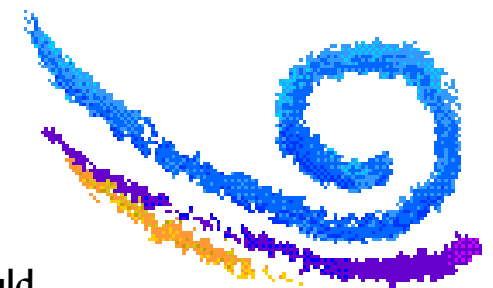
---

Notes:

The passwords of the users are stored in /etc/shadow. This file contains, from left to right:

- o The user name
- o The MD5 encrypted password of the user. MD5 encryption is a one-way encryption, meaning that once encrypted, a password can never be decrypted. To test whether an entered password is correct, the entered password is encrypted too and compared to the encrypted password in /etc/shadow. MD5 encryption is rather new. Older UNIXes, and other Linux distributions might still be using the old crypt algorithm. The real advantage of MD5 is that the allowed password length is increased from 8 to 256 characters.  
A "\*" means that this user does not have a password. That user account can therefore not be used to login.
- o The day the password was last changed (number of days since Jan 1st, 1970).
- o Number of days before the password may be changed again.
- o Number of days after which the password has to be changed again.
- o Number of days the user will be warned of a password expiry.
- o Number of days after expiry, after which the account is disabled.
- o The day the account was disabled.
- o A reserved field.

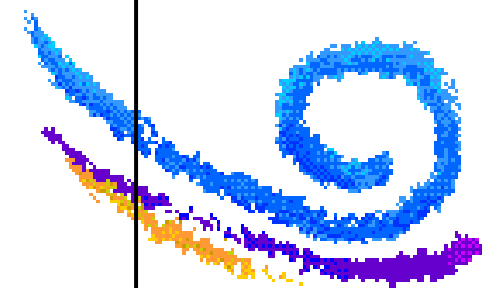
The /etc/shadow password file should be read/writeable by root only. Other users should not be able to read this file at all.



# /etc/group

---

```
[root@pentium /root]# cat /etc/group
root::0:root
bin::1 root,bin,daemon
daemon :2:root,bin,daemon
sys::3 root,bin,adm
adm::4 root,adm,daemon
nobody :99:
users: 100:
floppy x:19:
console:x:101:
utmp:x 102:
pppusers:x:230:
popusers:x:231:
slipusers:x:232:
slocate:x:21:
xfs:x:233:
tux1:x:501:
tux2:x:502:
```

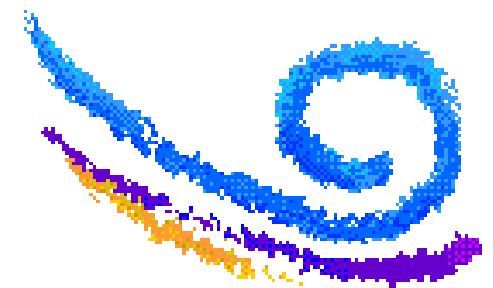




# User-Level Security Overview

---

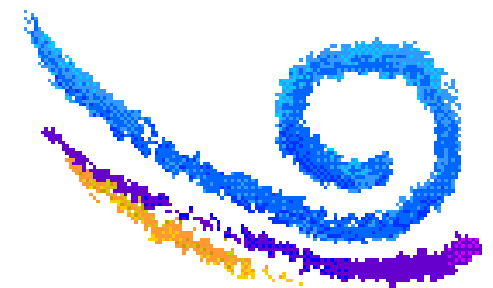
- Authentication: Verifying that you are who you say you are
- Can be based on:
  - Something you only know (for example, password, pin)
  - Something you only have (for example, smartcard, token, key)
  - Something you only are (for example fingerprints, retina scan)
- Authorization: Determining your level of access
  - In Linux implemented using file permissions
    - NIS, LDAP, Kerberos, SMB, ...
- User's "need to know"
- Disk usage and CPU limits
- Train users prior to turning them loose



# The Quota System

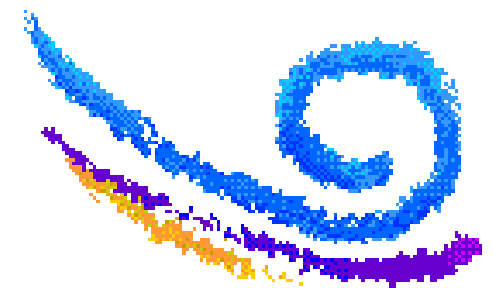
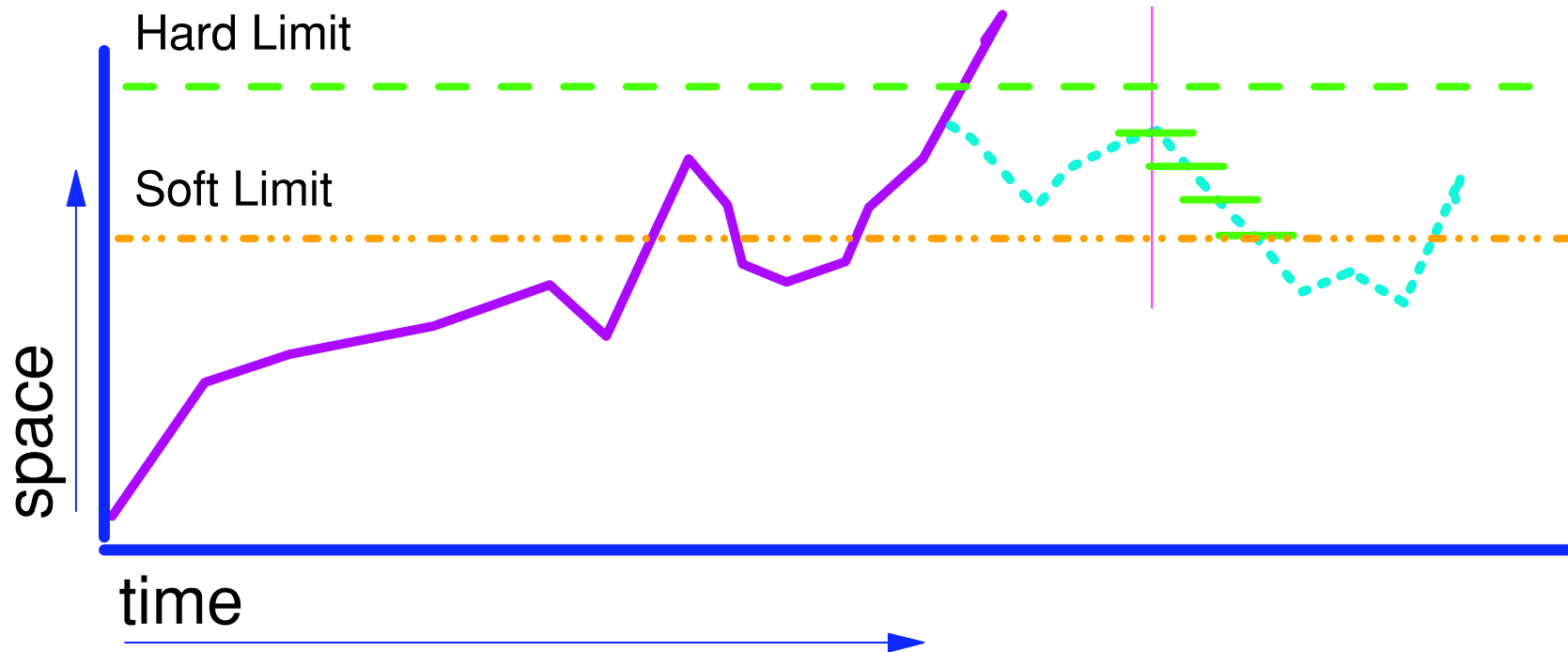
---

- Implementation within kernel, per filesystem
  - # quotaon|quotaoff fileys*
  - # quotacheck*
- Individual policies for users or groups, blocks or inodes
  - # edquota username [-p user1 user2 ...]*
  - # setquota username 2048 3072 20 30 /home*
- Reporting
  - # quota*
  - # repquota*
  - # warnquota*



# Quotas 101

---

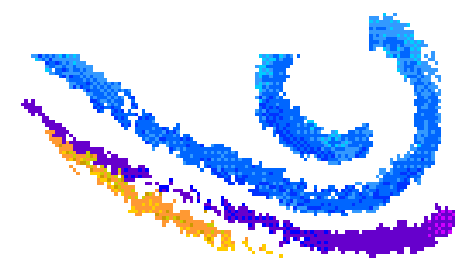


# /etc/fstab

```
root@server1:~ - Shell - Konsole
Session Edit View Bookmarks Settings Help

[root@server1 root]
# cat /etc/fstab
#Device          Mnt Pnt          Type    Options          D F
LABEL=/          /                 ext3    defaults          1 1
LABEL=/home      /home            ext3    defaults,usrquota 1 2
none             /dev/pts         devpts  gid=5,mode=620    0 0
none             /proc            proc    defaults          0 0
none             /dev/shm         tmpfs   defaults          0 0
/dev/sda2        swap             swap    defaults          0 0
/dev/cdrom       /mnt/cdrom       udf,iso9660 noauto,owner,kudzu,ro 0 0
/dev/fd0         /mnt/floppy      auto    noauto,owner,kudzu 0 0

[root@server1 root]
# █
```



# So Who's the Filehog?

---

**# df** -- look for 'full' filesystem  
-80/20 rule ?

**# du -s /home** -- look for filehog

-du -s /home

48        /home/bob

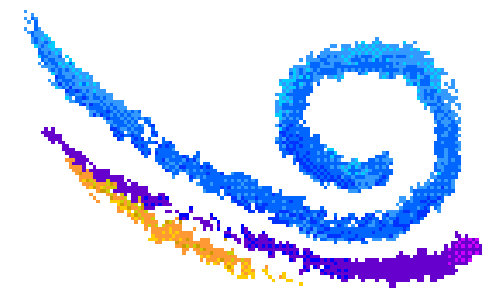
97874   /home/clyde

44       /home/eddie

50       /home/harry

**# ls** -- to prove your suspicion

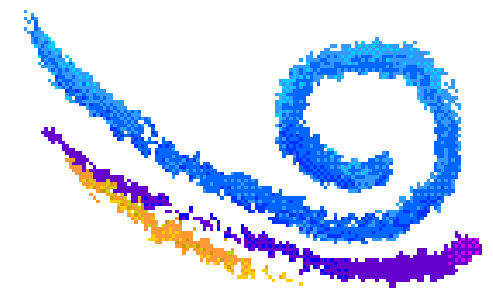
-What options would be valuable?



# File Ownership

---

- Every file has user and group "ownership"
- A new file is owned by the creator and the primary group of that user
- ***chown*** can be used by "root" to modify ownership

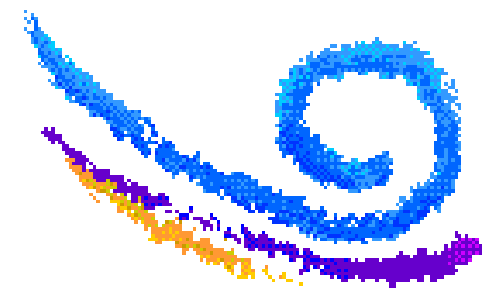


# Linux File Permissions

---

- -rwxrwxrwx
- breaks down into three distinct collections
  - owner (user)
  - group
  - other (the rest of the world)
- Each collection has 'read,write,execute' modes
- Permission given moving left to right

owner	group	other
<b>rwx</b>	<b>rwx</b>	<b>rwx</b>



# SUID / SGID Executables

---

`-rwSr-xr-x 1 root root /usr/bin/passwd`

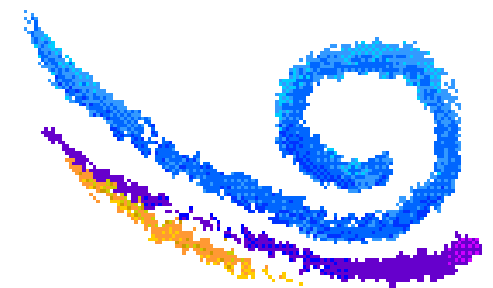
- Allows anyone to "become" root while running the program

`-rwxrwSrwx 1 pete staff /data/ideas`

- Allows all new files to be "owned" by the group 'staff'

`-rwxrwxrwT 12 root root /tmp`

- Allows anyone to add a file to /tmp
- Allows you to delete onely your own files



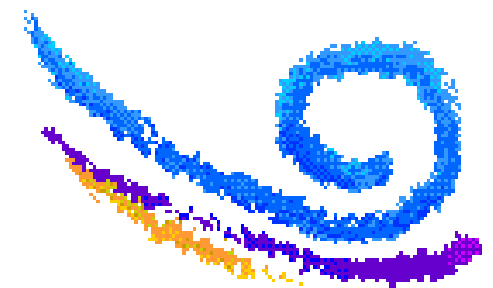


# Default File Permissions

---

- System default is: `rw-rw-rw-`
- Your new file is: `rw-r--r--`
- The difference is: `---w--w-`
- That value is called 'umask'
- `umask nnn`
  
- Directories are effected as well

$$\begin{array}{r} 777 \\ - 022 \\ \hline 755 \end{array}$$



# Access Control Lists

---

- Grant RWX access to files for multiple users|groups

***# mount -o acl***

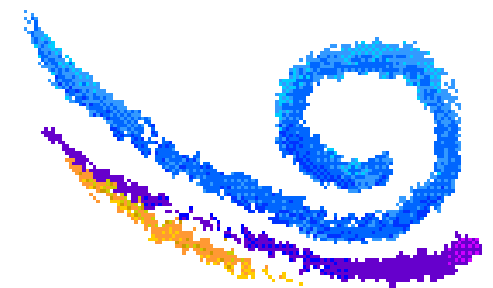
***# getfacl file/dir***

***# setfacl -m u:gandolf:rwX***

***# setfacl -m g:naxgul:rw***

***# setfacl -m d:u:frodo:rw***

***# setfacl -x u:samwise***



# Summary

---

Attendees should now be able to:

- Add, change and delete users and groups
- Manage user passwords
- Discuss authentication and verification
- Limit user space using Quotas
- Understand all the permission bits

