ALE - ATLANTA LINUX ENTHUSIASTS

- http://ale.org/ - sign up for our email lists
  - General email list - ale
  - This Study Group list - ale-study
- Meetup.com Group
  https://www.meetup.com/ALE-Atlanta-Linux-Enthusiasts/ (use google)

We are volunteers. This is our first attempt at this.

Thanks for coming out!

Notes for Wk 2

Notes from Linux Administration I PDF file

System and Users v4.0

What does a System Admin do?

What does a System Admin do?

Admins are different from end-users.

We enable computers for end-users to get their work done, while maintaining the system, files, and security of the system overall. There are thousands of tiny tasks for an admin to accomplish.

A few are:

Setup the system, backup the system, maintain a “run book” which explains the setup in a reproducible manner.

Maintain system security, prevent users from being hacked as much as possible, setup and maintain networking as needed, provide storage when possible, monitor logs for issues.

Monitor system performance to be proactive BEFORE there are user-impacting issues.

Communicate to management, end-users and other admins when changes are necessary.

System management is usually handled by modifying or creating text files in a specific location, in a specific format. Because text files are used, than means we can do things manually over very slow connections or use higher-level tools to maintain the settings through a CLI or GUI or DevOps tool. The choice is ours.

The Privileged root Account

The Privileged root Account

The root account is needed to manage a system.

Historically, this account could edit anything on the system anything. Even if it isn't a good idea, root can change it.
The root account is really any account that has a uid of 0 (zero). We can use this fact in some interesting ways.

Root is also called "super user".

Su is the tool to change users.

Sudo is the tool to allow other users, non-root, access to different accounts, including root. This tool, through configuration, can allow 1 user to run a command with only specific options as an other userid, for example. It isn't just for gaining root or installing software.

Permissions

Permissions are the way we:

a) keep users from trampling each other’s work

b) keep users from accidentally messing up a system

c) allow groups of users to more easily work together by allowing them to read + write to the same files and directories. This is a selective thing.

Each user can choose how much their files are locked down from others being able to write, read based on the normal Unix file and directory permissions model.

Only use root when it is needed.

Only use sudo when it is needed.

Exercises (interactive)

What is the difference between a user and an administrator?

Name examples for tasks and actions (and suitable commands) that are typically performed from a user account and the root account, respectively.

Why should you, as a normal user, not use the root account for daily work?

What are some reason?

What about access control on your computer at home?

Single account?
Multiple accounts?
Multiple people?

Do you work from an administrator account for non-admin tasks?

Login as Admin/root

How to login as admin?   What are the different ways?

How to login as admin?

A few ways:
a) login as root. Avoid doing this for a GUI session
b) login as a normal user, then su – to root
c) login as a normal user, then sudo -i to root (or sudo su -)

There are others. There are usually 10-500 different ways to accomplish anything on Unix.

Why run

/bin/sh –

instead of just

su?

Why run

/bin/sh –

instead of just

su?

There is a time for each.
Minimal “root” environment or just change your current e-uid to root.

What does

su – pete

do?

What does

su – pete

do?

Need to know pete’s password, if not root. The result is a login shell with pete as the userid

HOME is correctly set
Pete’s startup config is run.

What does
sudu -i -U pete
do differently?

What does

```
sudu -i -U pete
```

do differently?

Only need to know my password.

The result is a login shell with pete as the userid
* HOME is correctly set
* Pete’s profile and other setup files are run

Without the -i, HOME is left to my userid which will likely cause file permissions errors since pete cannot write to files in my HOME.

What is the /etc/securetty file used for?

What is the /etc/securetty file used for?

Lists allowed devices that root can directly login using.

Added as a way to prevent remote root attacks/exploits.

Why does Ubuntu NOT enable the root account?

Why does Ubuntu NOT enable the root account?

Because they are pansies? NO!

Direct access to root provides easy methods for attackers to gain complete control over a system, especially when they know the name of the root userid.

Home users wouldn’t know about this other account nor would they understand the purpose.

They wouldn’t disable remote root, which is a best practice and having a weak root password is extremely common.

Some other reasons?

Things to know - Shell prompts: $ vs #?

$	is for a normal userid.
#	is for root – or you are expected to use sudo in the required manner.

Any questions? We will use this all the time.

I will use **$ sudo** almost always. Habit.

Why should access to root/sudo be limited?

Too many cooks.

Accountability for mistakes.

Some level of knowledge is really necessary.

Why would using **git** (or any VCS) for /etc/ or **etckeeper** be a good idea?

Why would using git (or any VCS) for /etc/ or etckeeper be a good idea?

Can see all changes made to system config files over time.

Possible to restore to a previously working setup with minimal hassle.

Put things back after a mistake.

Versioned backups can fulfill the same requirements.

What are the good and bad things about using GUIs for system admin tasks?

Risks?

~/.config/

Wrong ownership.

Is downloading and running a shell script with sudo smart?

Why Not?

Which distro(s) normally have end-users running as root?

Puppy Linux

Exercises 5 min – work in teams

1.4 What methods exist to obtain administrator rights?   Which method is better?   Why?

1.5 On a conventionally configured system, how can you recognize whether you are working as root ?

1.6 Log in as a normal user (e. g., test). Change over to root and back to test. How do you work best if you frequently need to change between both these accounts (for example, to check on the results of a new configuration)?

1.7 Log in as a normal user and change to root using su . Where do you find a log entry documenting this change? Look at that message.

1.4 Distribution-specific Administrative Tools

Some distro-specific admin tools are awesome!
Some suck. List a few of each.

Webmin
SuSE
Ubuntu
RHEL / CentOS

Webmin

The danger is that inexperienced administrators will use an administration tool to attempt tasks which do not look more complicated than others but which, without adequate background knowledge,   may endanger the safety and/or reliability of the system.

Most GUI tools don’t include version control for the settings. That needs to be addressed outside the tool.

Why is point-n-click administration bad?

Why is point-n-click administration bad?

Pron to mistakes.

Managing 20-2,000 machines is next to impossible.

Not consistent.

Too slow.

Did you modify all the machines?

Requires a GUI, which uses lots of bandwidth. Do not expect physical access to any systems.

Training wheels are for noobs. Learn the true management of the system through files.

Are manual changes to settings reflected in the admin tool for your distro or are they ignored?

For just a few machines, the GUI ease-of-use may overrule the bad.

Avoid being distro-specific with your knowledge.

Expertise is needed in yast, yum, dnf, dpkg, apt – all of them. It doesn’t directly transfer.

Don’t become dependent on specific user/group admin GUIs, LDAP, printer setup and other distro specific tools.   Learn enough to do what needs to be accomplished.

This one time, at a prior job ……   6,000 printers for 12 systems.

Are manual changes to settings reflected in the admin tool GUI for your distro or are they ignored?

That is the $64,000 question.

If you use webmin, how should it be secured, in detail?

* Only allow localhost to access it.
* Use an ssh tunnel to make a tunnel for access.

How do administration teams ensure all the admins know what is going on for every server they manage?

* Runbooks.
* Talking.
* DevOps code.

How are server management and desktop management different?

* Desktop management can be on-the-fly for a single user system.
* Desktop management tends to have GUI tools.
* Server management needs to be predictable, deliberate, cautious, tested.
*

With tools like webmin, there is no need to learn old-school Linux admin skills.

* Not true.
* Someone with old-school admin skills needs to setup webmin
* AND secure it.
* AND fix it when it breaks.
* AND fix the system when webmin breaks something.

Next Time

* Wk 3 – User Administration
* Read through Pg 60 of the 101 Admin book.