freja.nordsiek@gwdg.de

Freja Nordsiek

User Management

Table of contents

1 Introduction

2 Controlling Sources

3 Local Files

Why have more usernames

- Running everything as `root` is **DANGEROUS!**
- Giving everyone access to `root` or some other shared username is **DANGEROUS!**
- Some services use different usernames
  - Reduces attack surface and improves security
  - Remember setting up `munge` and `slurm` for Slurm
- Give each user their own username
  - Separate credentials
  - Separate HOME and other directories
  - Avoid giving them admin access
  - Harder to harm the system (e.g. `rm -rf /`)
  - etc.

POSIX user/group model

- user account
  - Has an ID number (UID), currently uint32
  - Has a name (username)
  - Has a $HOME directory
  - Has a login shell
  - Has a password
  - Member of a primary group
  - Member of zero or more secondary groups
- group
  - Has an ID number (GID), currently uint32
  - Has a name (group name)
  - Zero or more users are members of it

User/Group sources

- Local to the machine (files)
  - `/etc/passwd`, `/etc/group`, `/etc/shadow`, `/etc/gshadow`
  - Allowed SSH keys defined in user's `HOME/.ssh/authorized_keys` file
  - Great for system groups and usernames
  - Works OK when the number of users is small
- Remote
  - Usernames and groups defined on a remote server
  - Client usually SSSD (`https://sssd.io`)
    - Replaces NSCD and NIS (formerly Yellow Pages)
  - Server
    - LDAP
    - Active Directory
  - Organization specific
    - They will tell you the server/s parameters

The Files

- `/etc/nsswitch.conf`
  - Name Service Switch
  - Controls the sources for each database
  - Format: `DATABASE: SOURCE1 SOURCE2 ...`
  - Sources used in order given
  - Databases like `passwd`, `group`, etc.
  - Sources like `files`, `sss` (SSSD), etc.
- `/etc/pam.d/*`
  - Pluggable Authentication Module (PAM)
  - Control the process of authentication once information is looked up

Authselect

- Possible to manually set `/etc/nsswitch.conf` and `/etc/pam.d/*`
- Authselect can do most of the work
  - Set a profile and features
  - `authselect select PROFILE [FEATURE1 ...]`
  - Then do manual edits for the few changes required
    - e.g. Slurm PAM Adopt
  - Note, does not make SSSD config for you
- Examples
  - `authselect select minimal with-faillock`
  - `authselect select sssd with-pam-u2f-2fa`

Groups

Defined in `/etc/group`, one per line with colon separated fields

```
NAME:x:GID:SECONDARYMEMBERS
```

- Only usernames who have the group as a secondary group are listed.
- Secondary members are separated by commas
- The "x" in the second field
  - Technically, groups can have passwords
  - Indicates that the password is in `/etc/gshadow`
    - Entry is almost always `NAME:::` or `NAME:!::`
    - Passwords very rarely used
- High level tools
  - `groupadd`
  - `groupdel`
  - `groupmod`

Users (Part 1)

Defined in `/etc/passwd`, one per line with colon separated fields

`USERNAME:x:UID:PRIMARYGID:GECOS:HOME:SHELL`

- `PRIMARYGID` is the GID of the primary group
- `GECOS` is an optional field to hold info (often name or email address)
- `HOME` the absolute path to the user's HOME
- `SHELL` path to the user's login shell
  - e.g. `/bin/bash`
  - Use `/sbin/nologin` to disable normal login (still other ways to run stuff)
- The "x" in the second field
  - Indicates that the password is in `/etc/shadow`
- High level tools
  - `useradd`
  - `userdel`
  - `usermod`

Users (Part 2)

Encrypted passwords stored in `/etc/shadow`, one per line with colon separated fields

```
USERNAME:PASSWORD:LASTCHANGED:::
```

- `PASSWORD` is the encrypted and salted password
  - `!!` means password authentication disabled
  - **Password should almost always be disabled**
    - SSH key authentication for users is safer
- `LASTCHANGED` when it was last changed (days since 01.01.1970)
  - `0` means must change on next login
- Last 3 fields are for password expiration and often blank
- Tools to work with passwords directly
  - `passwd`
  - `mkpasswd`

Warewulf

Warewulf provides some tooling to help manage locally defined users and groups

- Create groups and users on Warewulf node
- `syncuser` overlay
  - Merges contents of container's and Warewulf node's files
    - `/etc/group`
    - `/etc/passwd`
  - Does not merge `/etc/shadow` and `/etc/gshadow`
    - You have to set passwords manually in the containers
- `/etc/profile.d/ssh_setup.*sh`
  - Profile script that that is run on login
  - Creates SSH keys for user if they don't exist
  - Adds those keys to the user's `HOME/.ssh/authorized_keys`
  - Useful for SSH-ing between different nodes

More on passwords

- Current best practices
  - Store only a hash of the password
  - Salt the hash (prevents rainbow table attacks)
  - Make hash computationally expensive to brute force
- Can generate encrypted passwords with `mkpasswd -m METHOD -R ROUNDS`
  - `METHOD`, usually pick `yescrypt`, `scrypt`, or `sha512crypt`
  - `ROUNDS` determines compute time (some are logarithmic)
- `mkpasswd -m help` to see supported methods
- Paste generated password into one of the following
  - `useradd -p 'PASSWORD' ...`
  - `/etc/shadow`
- If you set a `root` password, it should be very slow to hash