



**UAF Cyber Security Club**

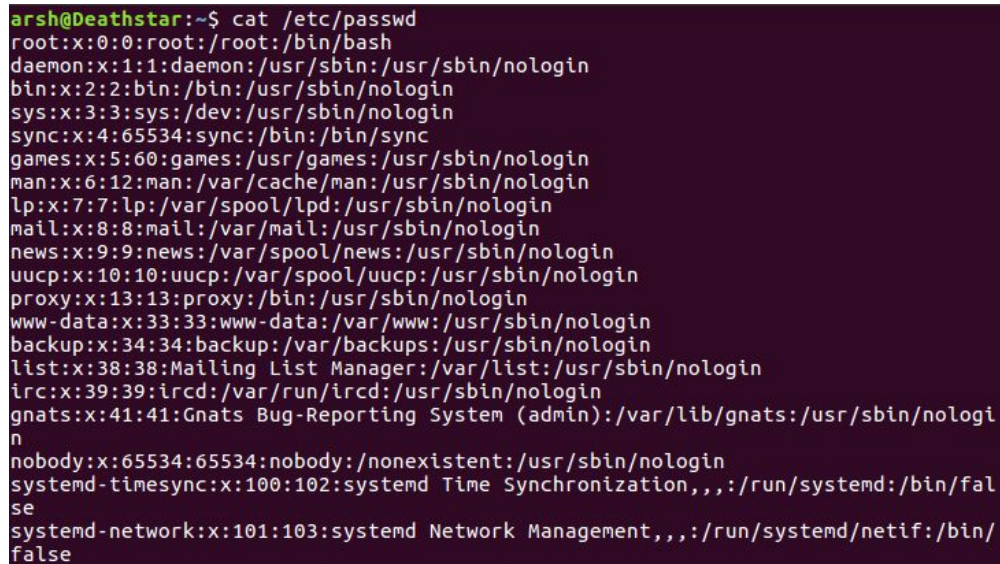
## **Linux User Management**

**Arsh Chauhan**

**11/18/2017**

# Why is it important?

Any computer today has multiple users, even if you think you only have one user. Each user needs access to their own files while also trusting the system that their files will not be accessible by others (Well sometimes we want to share files, more on this later). I mentioned earlier that even systems where you think you have one user have multiple users, let take a look at this.

A terminal window with a dark purple background. The prompt is 'arsh@Deathstar:~\$'. The command 'cat /etc/passwd' has been executed, displaying the contents of the /etc/passwd file. The output lists system users and regular users, each with their username, UID, GID, name, home directory, and shell. Users listed include root, daemon, bin, sys, sync, games, man, lp, mail, news, uucp, proxy, www-data, backup, list, irc, gnats, nobody, systemd-timesync, and systemd-network.

```
arsh@Deathstar:~$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-timesync:x:100:102:systemd Time Synchronization,,,:/run/systemd:/bin/false
systemd-network:x:101:103:systemd Network Management,,,:/run/systemd/netif:/bin/false
```

The image above shows the **passwd** file for my computer. This file hold information on all users on a linux system. You'll notice there are a lot of users with weird names, these are system users that are crucial for the system to function normally and don't need access to my files so it's important that the OS has a system to manage user privileges, bad things like [DirtyCOW](#) happen when these system have flaws or are incorrectly configured. Let go look at User management on Linux and Windows.

## User Management in Practice

### Linux

I'm going to technically incorrect and use Linux as an OS term (It's technically a Kernel) but every Linux distro I've seen does users in the same way. The first thing we'll talk about is the Sudoers file or the wheel or sudo group.

## Sudo

The command **sudo** lets a non-root user execute commands as root. It also lets the user become root and then become another user on the system. So, a user with **sudo** privileges can basically circumvent any user management policies. This is why controlling **sudo** access is supercritical to securing a Linux machine.

Sudo access is controlled by adding/removing users from the sudo or wheel group. Add a user to a group by using the [usermod](#) command

```
sudo usermod -a -G sudo bob #add user bob to sudo
```

This is a chicken and egg problem, you either need to be root or in the sudo group to add a new user to sudo and forgetting a flag like **-a** results in the following



Forgetting to add the **-a** flag just replaces all your groups with the ones you just added yourself to. In the above picture, I added myself to a group called **dialout** and removed myself from everything else (including sudo).

## The Sudoers File

The sudoers file located in **/etc/sudoers** controls how sudo works. Messing with this file is an easy way for attackers to bypass sudo or a stupid sysadmin can basically get rid of user privilege control by wrongfully editing this file. This file can be edited by

```
sudo visudo /etc/sudoers
```

Always use **visudo** to edit the sudoers file. The sudoers file is syntax dependent and **visudo** checks for any syntax issues before saving your changes.

**NOTE:** Messing up the sudoers file breaks sudo. Dr Orion Lawlor suggests always having a root terminal open before editing **sudoers** (I try to follow this and highly suggest making this a habit)

# Lab

## Scenario

You're a cyber security analyst for AllSafe CyberSecurity, the security contractor for E-Corp. You've been assigned to audit the user accounts on some of their machines after the F-Society hack and Tyrrell Wellick being fired.

Attached is an organization chart for everyone that matters. Based on this information, complete the following tasks and write a report for Gideon Goddard.

- 1) Provide a list of all user accounts
- 2) Provide a list of all sudo users
- 3) List users you removed from sudo (if any)
- 4) Add the current CTO to sudo
- 5) Add a user for Gideon Goddard with sudo
- 6) Any weird configuration issues

## Org Chart

Full Name	Position	Status
Phillip Price	CEO	Employed
Terry Colby	CTO	Fired 02/15
Tyrell Wellick	Interim CTO/ Ladies man	Fired
Scott Knowles	CTO	Employed
Angela Moss	Risk Management/ Pretty face	Employed
Elliot Anderson	AllSafe Analyst	Employed

## Creds

Full Name	Username	Password
Philip Price	pprice	god
Terry Colby	tcolby	fsociety
Tyrell Wellick	twellick	sharon

Scott Knowles	sknowles	revenge
Angela Moss	amoss	badmen
Elliot Anderson	eanderson	mrrobot
fsociety	fsociety	anarchy