Auditing and Policies

Linux (Ubuntu)

NORTHROP GRUMMAN

Cyber Security Challenge UK

Introduction

This section will explain some of the auditing and policies you can set up on linux (in this case Ubuntu). These auditing and policy changes can be used to further secure a system, and log changes or malicious behaviour. I will also talk about the use cases for these policies, and why they make a system more secure.

Level 1


CYBERCENTURION
EDUCATE. CHALLENGE. INSPIRE
YOUTH CYBER DEFENCE COMPETITION

Firewall - UFW

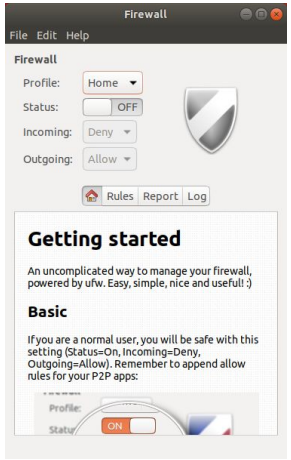
The most common way to set up a firewall in Ubuntu is by using Uncomplicated Firewall (UFW) which comes preinstalled with Ubuntu. UFW by itself is a command line based tool, but you can install a GUI for it called GUFW to make it more user friendly.

GUFW can be installed using the "**apt-get install gufw**" command. It can then be opened by typing "**gufw**" into the terminal.

For more information on using apt, see page 7 of the software module of these linux training materials.

```
harry@ubuntu:~$ sudo apt-get install gufw
[sudo] password for harry:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  gufw
0 upgraded, 1 newly installed, 0 to remove and 97 not upgraded.
```

```
harry@ubuntu:~$ gufw
```

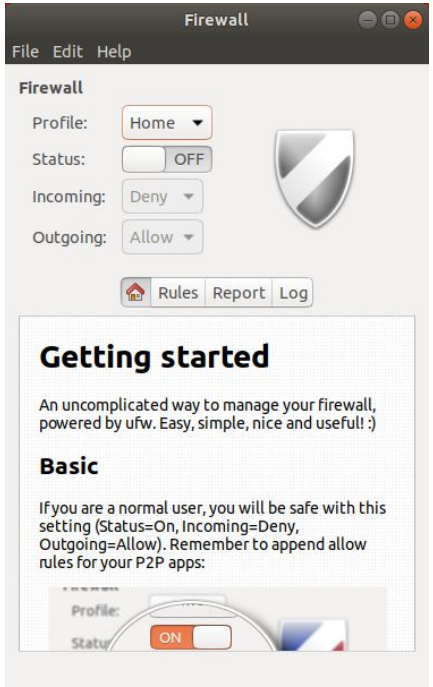


Firewall - GUFW

To enable the firewall, click the slider to turn it to "ON".

In this software you can set the base rules for incoming and outgoing packets, as well as create custom rules, which can be useful for allowing (or blocking) certain applications through the firewall.

For more information on setting up the firewall, go to: https://helpubuntu.com/community/Gufw



Level 2


CYBERCENTURION
EDUCATE. CHALLENGE. INSPIRE
YOUTH CYBER DEFENCE COMPETITION

SSH

Secure Shell (SSH) is a service that allows remote access to machines over a network, which could be a Local Area Network (LAN) such as a business network, or Wide Area Network (WAN) such as the internet. You might want to create an ssh server to allow managing a work server from home for example

Allowing remote access to a machine over a network obviously comes with some inherent security risks. I will cover some simple ways to reduce these.

Standard Ubuntu comes preinstalled with an SSH client, but not an SSH server.

For more information on SSH see: https://www.ssh.com/ssh/

SSH Server - Security

If a machine has an SSH server installed and running, there are a few changes to its configuration to make it more secure.

The configuration file for SSH Servers on Ubuntu is in the directory **/etc/ssh/sshd_config**.

A rundown of all of the options that can be changed can be found at:
https://linux.die.net/man/5/sshd_config

Password Policies

In order to ensure passwords are secure, a password policy should be enforced. This should enforce the length, complexity and age of passwords. It should also only allow a certain number of login attempts before the user is timed out. This is to stop brute-force attempts to gain access to an account.

You should take care when enforcing these, especially during the Cyber Centurion competition, as you can unintentionally trigger your own lockout policy for example.

Password Policies - Login Config

To enforce a maximum password age, the config file at **/etc/login.defs** contains an option to enforce this on new passwords. Note that this will not change existing passwords of users, the properties of these must be changed manually, or all users must be required to change their password next login.

"**PASS_MAX_DAYS**" sets the maximum age of a password. The default is "99999". I recommend this is set to a value between 30 and 120 for a balance between security and convenience.

"**PASS_MIN_DAYS**" sets the minimum time between password time. This should be set to a small number such as 2, just to avoid abuse of passwords.

Password Policies - PAM

Ubuntu comes preinstalled with a suite called Pluggable Authentication Modules (PAM) that handles authentication in linux.

You can install more modules, or use the preinstalled ones in Ubuntu to manage passwords.

I recommend installing the **libpam_pwquality** library to easily enforce password complexity. This can be installed with the command "**apt-get install libpam-pwquality**".

The next slide will detail how to configure PAM with this library installed.

```
harry@ubuntu:~$ sudo apt-get install libpam-pwquality
[sudo] password for harry:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  libpam-pwquality
0 upgraded, 1 newly installed, 0 to remove and 98 not upgraded.
Need to get 11.2 kB of archives.
After this operation, 35.8 kB of additional disk space will be used.
Get:1 http://us.archive.ubuntu.com/ubuntu bionic/main amd64 libpam-pwquality amd
64 1.4.0-2 [11.2 kB]
Fetched 11.2 kB in 0s (48.9 kB/s)
Selecting previously unselected package libpam-pwquality:amd64.
(Reading database ... 203273 files and directories currently installed.)
Preparing to unpack .../libpam-pwquality_1.4.0-2_amd64.deb ...
Unpacking libpam-pwquality:amd64 (1.4.0-2) ...
Setting up libpam-pwquality:amd64 (1.4.0-2) ...
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...
```

PAM - Configuration

The password configuration file for PAM can be found at **/etc/pam.d/common-password**. This file will vary depending on what PAM modules you have installed on the system, but the areas I will show you will apply to the **pwquality** module.

The next slide will detail what I consider to be values that will result in a secure password, but they are by no means foolproof and will vary depending on the use case. Please take some time to understand what these arguments change. More information on PAM's pwquality module can be found at https://linux.die.net/man/8/pam_pwquality.

PAM - Configuration

These arguments should be added to the end of the line containing "pam_pwquality.so"

- **retry=3** - This sets the number of times you are prompted to enter a correct password before displaying an error (this does not actually lock the account though).
- **minlen=10** - Sets the minimum length for the new password. This is not quite the same as just the length though, as it counts up characters by their credit values below.
- **lcredit=-1** - Sets the minimum number of lowercase letters the password should contain.
- **ucredit=-1** - Sets the minimum number of upper case letters.
- **ocredit=-1** - Set the minimum number of symbols.
- **dcredit=-1** - Sets the minimum number of digits.
- **reject_username** - Rejects the password if contains the username of the user.
- **difok=3** - Specifies the number of characters that are allowed to be shared with the previous password
- **enforce_for_root**: Enforces these policies even if it is the root user making changes. Be careful when setting this.

PAM - Timeout

A PAM module that comes preinstalled with Ubuntu is **pam_tally2**. This module allows for counting the number of unsuccessful login attempts a user has made, and locking them out if they make too many.

The configuration file in which you can set this functionality is at **/etc/pam.d/common-auth**.

To set this up, this line should be added to the common-auth file:

auth required pam_tally2.so onerr=fail deny=3 unlock_time=300 audit

Take care with PAM timeouts, as if you enter the password wrong too many times in the competition you may lock yourself out!

This line will lockout a user for 300 seconds if they enter an incorrect password 3 times, and audit this error to an audit.log file. For more information on pam_tally2, see https://linux.die.net/man/8/pam_tally2

Shares - Samba

Samba allows for file sharing from linux to windows machines using the SMB protocol.

If Samba is installed on a linux machine, consider if it is necessary, as having a Samba share may unintentionally reduce the security of the machine and the network it is connected to. If it is not needed, think about disabling or removing the Samba service.

For more information on setting up Samba shares and securing them see

https://ubuntu.com/tutorials/install-and-configure-samba#1-overview and

http://linux-training.be/networking/ch21.html respectively

Level 3


CYBERCENTURION
EDUCATE. CHALLENGE. INSPIRE
YOUTH CYBER DEFENCE COMPETITION

Auditing - AuditD

In Ubuntu the easiest way to set up auditing of file changes and execution is using the service called AuditD (short for Audit Daemon)

To Install AuditD, simply use the command "**apt-get install auditd**".

AuditD comes with no rules set up by default, these have to be added manually.

Some of the directories I recommend setting up auditing for include the **/etc/passwd** (users file), **/etc/sudoers** (sudo config) and **/etc/group** (groups file).

For a tutorial on how to set up AuditD for a directory, see https://linoxide.com/how-tos/auditd-tool-security-auditing/

Auditing - Lynis

Lynis is a command line security auditing tool that can show you potential security vulnerabilities and aid with system hardening.

Lynis can be installed using the command "**apt-get install lynis**" and run with the command "**lynis audit system**" to run a basic scan.

This will output a long list of security checks, separated into category. For example, it is showing a problem with SSH PermitRootLogin, as I have not changed this argument to false.

```
sudo apt-get install lynis
```

```
harry@ubuntu:~$ sudo lynis audit system
```