ACME_32_a4_report

Group number: 32
Students:

- Lilit Hasan: 1958055
- Matteo Piermartini: 1802597
- Aurora Polifemo: 1802485

Initial Brainstorming

To configure the syslog service we decided to use the rsyslog tool, because it was already included in the Log Server and in the other hosts.

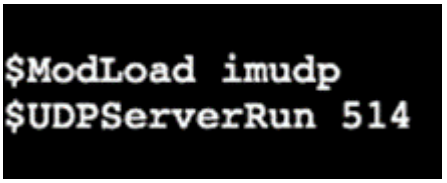
1. Configuration of the Log Server

Firstly, we check if the rsyslog service is in the Log server:

```
systemctl status rsyslog
```

In the file */etc/rsyslog.conf* we have to set the protocol to use for the exchange of log messages:

a. In this way we allow listening on the server's UDP port 514. It's the port the client will send messages to. We set only the UDP port and not the TCP port 514, because the UDP port is faster.



b. *$template* tells the rsyslog daemon to get and write all the log messages of different */var/log* folders based on the name of the hosts (name of the client's machine).
**.*?RemoteLogs* states that all the messages that come from all the different levels of gravity have to use the RemoteLogs model.
With the last line, we are telling the rsyslog to stop the processing of messages once they have been written in a file.

We need to restart the rsyslog service:

```
systemctl restart rsyslog
```

2. Configuration inside a client (Kali)

Inside the clients in which we want to configure rsyslog we have to write a simple line in the */etc/rsyslog.conf* file that allows the client to send log messages to the Log Server:

```
#ADD STUFF

*.* @100.100.1.3:514
```

Furthermore we add some other lines in this file, so that we can send some specific information to the Log Server:

```
$ActionQueueFileName queue
$ActionQueueMaxDiskSpace 1g
$ActionQueueSaveOnShutdown on
$ActionQueueType LinkedList
$ActionResumeRetryCount -1
```

We restart the service:

```
systemctl restart rsyslog
```

At last, we add a firewall rule in the SERVERS interface of the Internal Router, that allows the Log Server to send the various logs.

3. Test of the configuration

To test the service we can go on the Log Server and check the system's logs in the */var/log* folder. To check if the service works it is necessary to find the specific folder of the host we used to configure the log service.

For example:
If from *kali*, which IP is 100.100.2.100, we write the logger command:

```
log from $HOSTNAME
```

Then on the Log Server, we'll have this:

```
root@logserver:/var/log# ls
alternatives.log       btmp              dpkg.log.1      mail.info.1      mail.warn       messages.4.gz  syslog.7.gz
alternatives.log.1     btmp.1            dpkg.log.2.gz   mail.info.2.gz   mail.warn.1     private        user.log
alternatives.log.2.gz  daemon.log       dpkg.log.3.gz   mail.info.3.gz   mail.warn.2.gz  syslog         user.log.1
apt                    daemon.log.1      faillog         mail.info.4.gz   mail.warn.3.gz  syslog.1       user.log.2.gz
auth.log               daemon.log.2.gz   kali            mail.log         mail.warn.4.gz  syslog.2.gz    wtmp
auth.log.1             daemon.log.3.gz   kern.log        mail.log.1       messages        syslog.3.gz
auth.log.2.gz          daemon.log.4.gz   lastlog         mail.log.2.gz    messages.1      syslog.4.gz
auth.log.3.gz          debug             logserver       mail.log.3.gz    messages.2.gz   syslog.5.gz
auth.log.4.gz          dpkg.log          mail.info       mail.log.4.gz    messages.3.gz   syslog.6.gz
```

A new folder has been created, it's called *kali* like the host from which we sent the command. If we go inside the folder we can see some logs that have been created:

```
root@logserver:/var/log# cd kali
root@logserver:/var/log/kali# ls
 CRON.log                  gvfsd.log                          pulseaudio.log       sudo.log
 NetworkManager.log        kernel.log                         rsyslogd.log         systemd-logind.log
 at-spi-bus-launcher.log   lightdm.log                        rtkit-daemon.log     systemd.log
 dbus-daemon.log          'polkitd(authority=local).log'      spice-vdagentd.log   user.log
root@logserver:/var/log/kali#
```

Inside *user.log* we can see the *logger* command:

```
root@logserver:/var/log/kali# cat user.log
Jul  4 12:25:22 kali user: log from kali
```

4. Final Remarks

There were no complications during the configuration of the syslog service.