Bullet Proof Penguin

You have been hired by the XYZ company as a consultant to harden the Bulletproof Penguin, an old server that's never been hacked (as far as we know). As you arrive, the company's IT crew hands you a vulnerability scan report that was recently made against the server, and asks you to implement solutions to each finding.

Redis Server with no Password

The remote Redis server is not protected with a password. This issue may be exploited by a remote attacker to gain access to sensitive information or modify system configuration.

Configuring the password can be done by editing the redis.conf file

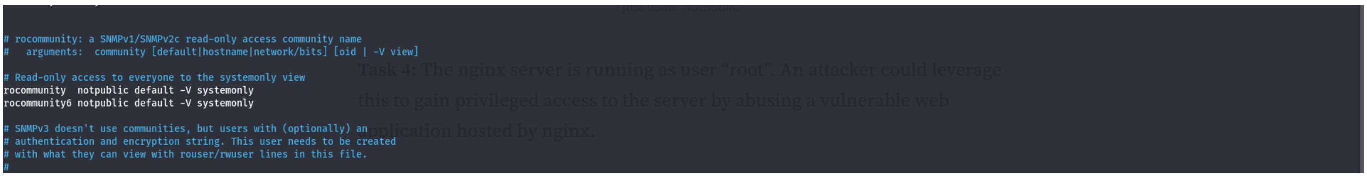
> requirepass myStrongPassword123!

Simple Network Management Protocol(SNMP)

Simple Network Management Protocol (SNMP) is a protocol which can be used by administrators to remotely manage a computer or network device.

Changes can be done through the

> /etc/snmp/snmpd.conf

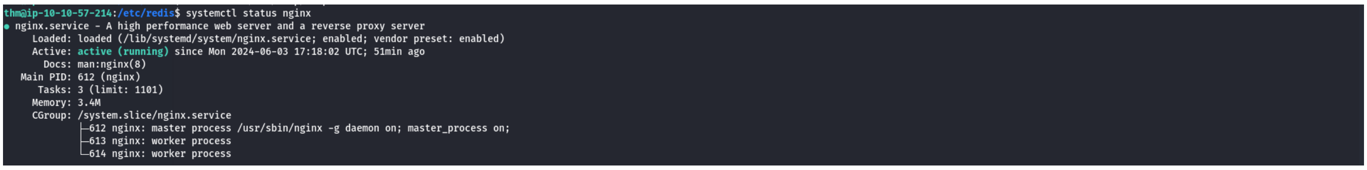


```
# rocommunity: a SNMPv1/SNMPv2c read-only access community name
#   arguments:  community [default|hostname|network/bits] [oid | -V view]

# Read-only access to everyone to the systemonly view
rocommunity  notpublic default -V systemonly
rocommunity6 notpublic default -V systemonly

# SNMPv3 doesn't use communities, but users with (optionally) an
# authentication and encryption string. This user needs to be created
# with what they can view with rouser/rwuser lines in this file.
#
```

Nginx

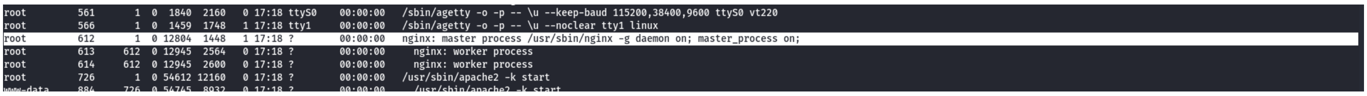
It is a web server that can also be used as a reverse proxy, load balancer, mail proxy and HTTP cache.

In this case it is running as root, this can be exploited and allow the user to gain root privileges.



```
thm@ip-10-10-57-214:/etc/redis$ systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
     Loaded: loaded (/lib/systemd/system/nginx.service; enabled; vendor preset: enabled)
     Active: active (running) since Mon 2024-06-03 17:18:02 UTC; 51min ago
       Docs: man:nginx(8)
   Main PID: 612 (nginx)
      Tasks: 3 (limit: 1101)
     Memory: 3.4M
     CGroup: /system.slice/nginx.service
             ├─612 nginx: master process /usr/sbin/nginx -g daemon on; master_process on;
             ├─613 nginx: worker process
             └─614 nginx: worker process
```

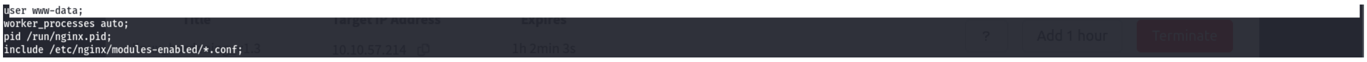
We can now check the proccesses to which the permissions



```
root         561       1  0  1840  2160   0 17:18 ttyS0    00:00:00  /sbin/agetty -o -p -- \u --keep-baud 115200,38400,9600 ttyS0 vt220
root         566       1  0  1459  1748   1 17:18 tty1     00:00:00  /sbin/agetty -o -p -- \u --noclear tty1 linux
root         612       1  0 12804  1448   1 17:18 ?        00:00:00  nginx: master process /usr/sbin/nginx -g daemon on; master_process on;
root         613     612  0 12945  2564   0 17:18 ?        00:00:00    nginx: worker process
root         614     612  0 12945  2600   0 17:18 ?        00:00:00    nginx: worker process
root         726       1  0 54612 12160   0 17:18 ?        00:00:00  /usr/sbin/apache2 -k start
www-data     884     726  0 54745  8932   0 17:18 ?        00:00:00    /usr/sbin/apache2 -k start
```

We need to change the account running the nginx service

> /etc/nginx/nginx.conf



```
user www-data;
worker_processes auto;
pid /run/nginx.pid;
include /etc/nginx/modules-enabled/*.conf;
```

> sudo systemctl restart nginx

Clear Text Protocols

The remote host is running a Telnet service that allows cleartext logins over unencrypted connections. An attacker can uncover login names and passwords by sniffing traffic to the Telnet service.

We need to diable telnet and TFTP(port 69)

> sudo nano /etc/inetd.conf

We need to hash out the telnet and TFTP lines

Unistalling telent

```
sudo dpkg -l | grep telnet - checking if its installed
sudo apt-get purge telnet - unistalling telnet client
sudo apt-get purge telnetd - unistalling telnet server

sudo apt-get autoremove
sudo apt-get autoclean
```

> sudo systemctl restart inetd

Weak SSH Keys

SSH is secure as compared to its counter-part telnet

- The remote SSH server is configured to allow / support weak key exchange (KEX) algorithm(s). An attacker can quickly break individual connections.
- The remote SSH server is configured to allow / support weak encryption algorithm(s).
- The remote SSH server is configured to allow / support weak MAC algorithm(s).

We can edit the ssh config file

> /etc/ssh/sshd_config

Anonymous FTP Login

A host that provides an FTP service may additionally provide Anonymous FTP access as well. Under this arrangement, users do not strictly need an account on the host. Instead, the user typically enters 'anonymous' or 'ftp' when prompted for username. Although users are commonly asked to send their email address as their password, little to no verification is actually performed on the supplied data.

> sudo nano /etc/vsftpd.conf

Here you change the anonymous login to NO

> systemctl restart vsftpd.service

Weak Passwords

Strong passwords are key for security.

changing password for an account

> sudo passwd username

deleting an account

> sudo userdel -r username #-f to force issues

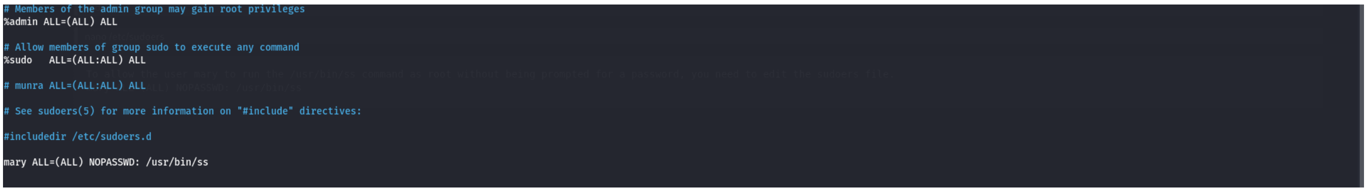
Sudo Permissions

Ensure that permissions to execute elevated commands via sudo are granted only to users that strictly require it.

> sudo -l

> nano /etc/sudoers

```
To allow the user mary to run the /usr/bin/ss command as root without being
prompted for a password, you need to edit the sudoers file.
mary ALL=(ALL) NOPASSWD: /usr/bin/ss
```



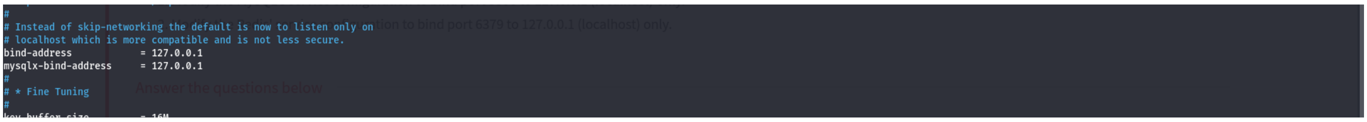
```
# Members of the admin group may gain root privileges
%admin ALL=(ALL) ALL

# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL

# munra ALL=(ALL:ALL) ALL

# See sudoers(5) for more information on "#include" directives:

#includedir /etc/sudoers.d

mary ALL=(ALL) NOPASSWD: /usr/bin/ss
```

Exposed Database Ports

While not a vulnerability in itself, exposing database ports makes them prone to brute-force attacks and other exploits. Ensure that access to the reported database ports is restricted to the minimum necessary.

> sudo nano /etc/mysql/mysql.conf.d/mysqld.cnf

> sudo systemctl restart mysql



```
#
# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
bind-address            = 127.0.0.1
mysqlx-bind-address     = 127.0.0.1
#
# * Fine Tuning
#
key_buffer_size         = 16M
```

> sudo nano /etc/redis/redis.conf