

12/4/2021

CentOS

My Lynis reports, including any changes you made to each server and why you made those changes.

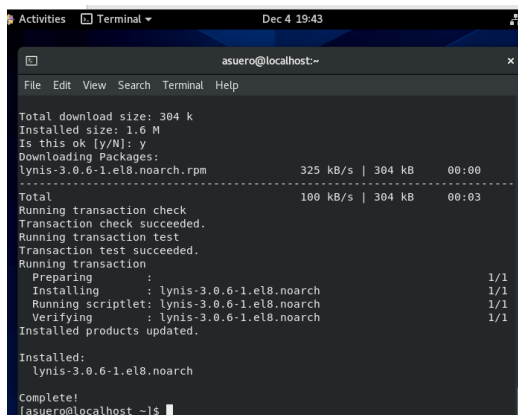
Lynis Installation

Lynis is an open-source security tool. It helps with security scanners on systems running Linux, macOS, and BSD.

To install lynis please type the following command:

yum install lynis

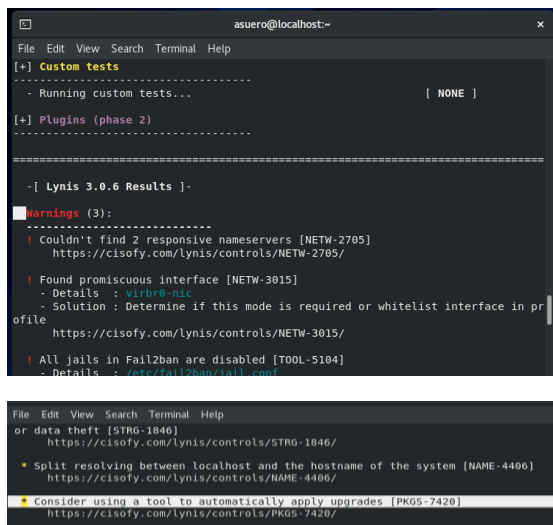
then hit enter, later you will be asked to confirm the download, type **y** for yes. Wait a few seconds depending on your system, and later you will be all set.

A terminal window titled 'Terminal' with a date of 'Dec 4 19:43'. The prompt is 'asuro@localhost:~'. The output of the command 'yum install lynis' is shown. It starts with 'Total download size: 304 k' and 'Installed size: 1.6 M'. It asks 'Is this ok [y/N]: y'. Then it shows 'Downloading Packages:' with a progress bar for 'lynis-3.0.6-1.el8.noarch.rpm' at 325 kB/s | 304 kB | 00:00. A second progress bar shows 'Total' at 100 kB/s | 304 kB | 00:03. It then shows 'Running transaction check', 'Transaction check succeeded.', 'Running transaction test', 'Transaction test succeeded.', 'Running transaction', 'Preparing : lynis-3.0.6-1.el8.noarch 1/1', 'Installing : lynis-3.0.6-1.el8.noarch 1/1', 'Running scriptlet: lynis-3.0.6-1.el8.noarch 1/1', 'Verifying : lynis-3.0.6-1.el8.noarch 1/1', and 'Installed products updated.'. It ends with 'Installed: lynis-3.0.6-1.el8.noarch' and 'Complete! [asuro@localhost ~]\$'.

After you install lynis, you run it to scan your system by typing the following command

sudo lynis audit system -Q

when you run lynis, when the audit to your system is complete you will be suggested if needed with some advice, some warning, and follow up lynis showed me some warning, and I will take them into consideration to keep my server safer from attackers. Below there are a couple of images with my warnings and suggestions.



```
asuro@localhost:~  
File Edit View Search Terminal Help  
[+] Custom tests  
-----  
- Running custom tests... [ NONE ]  
-----  
[+] Plugins (phase 2)  
-----  
=====
```

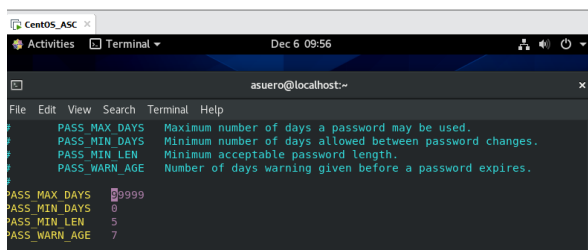
```
-[ Lynis 3.0.6 Results ]-  
Warnings (3):  
-----  
! Couldn't find 2 responsive nameservers [NETW-2705]  
https://cisofy.com/lynis/controls/NETW-2705/  
  
! Found promiscuous interface [NETW-3015]  
- Details : virbr0-nic  
- Solution : Determine if this mode is required or whitelist interface in pr  
ofile  
https://cisofy.com/lynis/controls/NETW-3015/  
  
! All jails in Fail2ban are disabled [TOOL-5104]  
- Details : /etc/fail2ban/jail.conf
```

```
File Edit View Search Terminal Help  
or data theft [STRG-1846]  
https://cisofy.com/lynis/controls/STRG-1846/  
  
* Split resolving between localhost and the hostname of the system [NAME-4406]  
https://cisofy.com/lynis/controls/NAME-4406/  
  
* Consider using a tool to automatically apply upgrades [PKG5-7420]  
https://cisofy.com/lynis/controls/PKG5-7420/
```

For my system I will follow some suggestion and warnings, some of them are:

- Set up a tool for automatic update
- Work with my password security, for example, configure minimum password requirement.
- Install a malware scanner to periodically scan my system (for now, lynis is good for me)
- Enable the jail in fail2ban

For now, we are going to work with the password requirement, we are going to set up password length and password expiration. To change the password expiration time, we must go to the following file with the root privileges **vi /etc/login.defs**, without root privileges you won't be able to change anything in this file.



```
CentOS_ASC  
Activities Terminal Dec 6 09:56  
asuro@localhost:~  
File Edit View Search Terminal Help  
# PASS_MAX_DAYS Maximum number of days a password may be used.  
# PASS_MIN_DAYS Minimum number of days allowed between password changes.  
# PASS_MIN_LEN Minimum acceptable password length.  
# PASS_WARN_AGE Number of days warning given before a password expires.  
#  
PASS_MAX_DAYS 9999  
PASS_MIN_DAYS 0  
PASS_MIN_LEN 5  
PASS_WARN_AGE 7
```

Type **sudo /etc/login.defs** later change the pax max days to 90. And **save and exit**. To save and exit you can hit the **key scape** and later type **:wq!**

```
Activities Terminal Dec 6 10:14
asuro@localhost:~
File Edit View Search Terminal Help
# Password aging controls:
#
# PASS_MAX_DAYS Maximum number of days a password may be used.
# PASS_MIN_DAYS Minimum number of days allowed between password changes.
# PASS_MIN_LEN Minimum acceptable password length.
# PASS_WARN_AGE Number of days warning given before a password expires.
#
PASS_MAX_DAYS 90
PASS_MIN_DAYS 0
PASS_MIN_LEN 5
PASS_WARN_AGE 7
#
```

As we can see in the image below the password max day **was 99999**, we are going to change that to **90**

To change the password length please go to the following file **sudo vi /etc/security/quality.conf** it has an **8** minimum requirement we are going to change it to **10**.

```
asuro@localhost:~
File Edit View Search Terminal Help
Defaults:
#
# Number of characters in the new password that must not be present in the
# old password.
# difok = 1
#
# Minimum acceptable size for the new password (plus one if
# credits are not disabled which is the default). (See pam_cracklib manual.)
# Cannot be set to lower value than 6.
# minlen = 8
#
```

```
Activities Terminal Dec 6 10:44
asuro@localhost:~
File Edit View Search Terminal Help
# Defaults:
#
# Number of characters in the new password that must not be present in the
# old password.
# difok = 1
#
# Minimum acceptable size for the new password (plus one if
# credits are not disabled which is the default). (See pam_cracklib manual.)
# Cannot be set to lower value than 6.
# minlen = 10
#
```

Also, we are going to force the user to use at least one upper letter and at least one lower letter. So, change the 0 on the **#ucredit = 0** to 1, and the same with the **#lcredit = 0** change it to 1

```
asuro@localhost:~
File Edit View Search Terminal Help
# Minimum acceptable size for the new password (plus one if
# credits are not disabled which is the default). (See pam_cracklib manual.)
# Cannot be set to lower value than 6.
# minlen = 10
#
# The maximum credit for having digits in the new password. If less than 0
# it is the minimum number of digits in the new password.
# dcredit = 0
#
# The maximum credit for having uppercase characters in the new password.
# If less than 0 it is the minimum number of uppercase characters in the new
# password.
# ucredit = 1
#
# The maximum credit for having lowercase characters in the new password.
# If less than 0 it is the minimum number of lowercase characters in the new
# password.
# lcredit = 0
#
```

```
asuario@localhost:~  
File Edit View Search Terminal Help  
# credits are not disabled which is the default). (See pam_cracklib manual.)  
# Cannot be set to lower value than 6.  
# minlen = 10  
#  
# The maximum credit for having digits in the new password. If less than 0  
# it is the minimum number of digits in the new password.  
# dcredit = 0  
#  
# The maximum credit for having uppercase characters in the new password.  
# If less than 0 it is the minimum number of uppercase characters in the new  
# password.  
# ucredit = 1  
#  
# The maximum credit for having lowercase characters in the new password.  
# If less than 0 it is the minimum number of lowercase characters in the new  
# password.  
# lcredit = 1  
#
```

To run the script

- You need root privileges
- Also, you need to install sysstat

You may need to specify the full script path. Type the following command assuming you are on a Linux machine **ssh USER@HOST 'bash -s' < SCRIPT**. In my case, I am currently on a windows machine, and I used putty to run the script remotely.

My system health report is saved in the same place I created my script in my case is in my **home** directory.

```
asuario@localhost:~  
File Edit View Search Terminal Help  
[asuario@localhost ~]$ ls  
datebook  lynis.log  Net_Scr  oputout.txt  Templates  
Desktop  lynis-report.dat  Pictures  
Downloads  MonitoringScript.txt  Public  
Downloads  MonitorScriptout.txt  redi-release-8.rpm  
~/Downloads  MonitorScriptout.txt  redi-release-8.rpm  
[asuario@localhost ~]$ pwd  
/home/asuario  
[asuario@localhost ~]$
```

What you picked, why you picked those commands and how they are used.

Commands:

Ifconfig: I used this command to get my IP address.

Hostname: I used this command to get my hostname.

Hostnactl: I used this command to get my kernel version and my operative system information because I think a good health report should include that information about your system.

lstat -c: I used this command to get the information about the CPU because it gave me the information, I need to use in my report such as the CPU used by the system, the CPU used by users, etc.

Ps: This command provides me information about the active process running in the system, it is good to know what is running on your server.

Debian

Installation the lynis

To install lynis on centos type the following command **sudo apt-get install lynis** and hit enter. Then you will be prompted to select **y/n** to confirm type **y** for yes.

```
arseniasec@debian:~$ sudo apt-get install lynis
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  lynis
Suggested packages:
  apt-listbugs debsums debsums-tripwire scanmail-xfs menu-l3m-gksu
  | kde-clip-tools | kxtools
The following NEW packages will be installed:
  lynis
0 upgraded, 1 newly installed, 0 to remove and 23 not upgraded.
Need to get 636 kB of archives.
After this operation, 3,371 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://deb.debian.org/debian bullseye/main amd64 lynis all 3.0.2-1 [263 kB]
Get:2 http://deb.debian.org/debian bullseye/main amd64 lynis amd64 3.0.2-1 [373 kB]
Fetched 636 kB in 1s (879 kB/s)
Selecting previously unselected package lynis.
(Reading database ... 34209 files and directories currently installed.)
Preparing to unpack .../archivos/lynis_3.0.2-1_all.deb ...
Unpacking lynis (3.0.2-1) ...
```

To audit or check the status of your server type the following command: **lynis audit system**

Lynis suggestions and warnings to my system

```
arseniasec@debian:~$ lynis audit system
https://cisofy.com/lynis/controls/STRG-1846/
* Disable drivers like firewire storage when not used, to prevent unauthorized storage or data theft [STRG-1846]
https://cisofy.com/lynis/controls/PKGS-7370/
* Install debsums utility for the verification of packages with known good database. [PKGS-7370]
https://cisofy.com/lynis/controls/PKGS-7392/
* Update your system with apt-get update, apt-get upgrade, apt-get dist-upgrade and/or unattended-upgrades [PKGS-7392]
https://cisofy.com/lynis/controls/KRNL-5820/
https://cisofy.com/lynis/controls/AUTH-9230/
* Configure password hashing rounds in /etc/login.defs [AUTH-9230]
https://cisofy.com/lynis/controls/AUTH-9262/
* Install a PAM module for password strength testing like pam_cracklib or pam_passwdqc [AUTH-9262]
https://cisofy.com/lynis/controls/AUTH-9282/
* When possible set expire dates for all password protected accounts [AUTH-9282]
https://cisofy.com/lynis/controls/AUTH-9286/
* Configure minimum password age in /etc/login.defs [AUTH-9286]
https://cisofy.com/lynis/controls/AUTH-9286/
* Configure maximum password age in /etc/login.defs [AUTH-9286]
https://cisofy.com/lynis/controls/AUTH-9328/
* Default umask in /etc/login.defs could be more strict like 027 [AUTH-9328]
https://cisofy.com/lynis/controls/AUTH-9328/
* To decrease the impact of a full /home file system, place /home on a separate partition [FILE-6310]

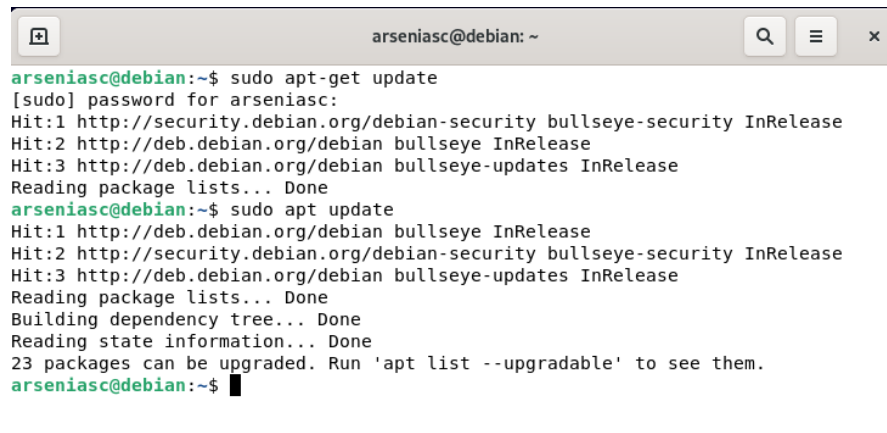
=====
-[ Lynis 3.0.2 Results ]-
Warnings (3):
! Found one or more vulnerable packages. [PKGS-7392]
https://cisofy.com/lynis/controls/PKGS-7392/
! Couldn't find 2 responsive nameservers [NETW-2705]
https://cisofy.com/lynis/controls/NETW-2705/
! iptables module(s) loaded, but no rules active [FIRE-4512]
https://cisofy.com/lynis/controls/FIRE-4512/
- . . . -
```

For my system I will follow some suggestion and warnings, some of them are:

- Work with my password security, for example, configure minimum password requirement.
- Update my system

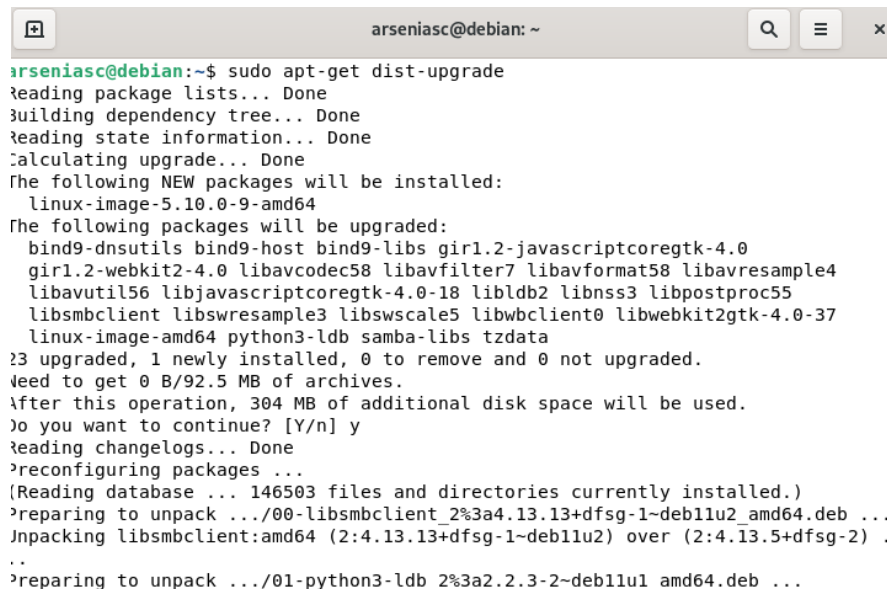
I have a few suggestions and I will work on some of them such as the password requirement set up and updating my system with the apt update.

To see the available updates on system please type the following command **sudo apt-get update** on the command line.



```
arseniasc@debian:~$ sudo apt-get update
[sudo] password for arseniasc:
Hit:1 http://security.debian.org/debian-security bullseye-security InRelease
Hit:2 http://deb.debian.org/debian bullseye InRelease
Hit:3 http://deb.debian.org/debian bullseye-updates InRelease
Reading package lists... Done
arseniasc@debian:~$ sudo apt update
Hit:1 http://deb.debian.org/debian bullseye InRelease
Hit:2 http://security.debian.org/debian-security bullseye-security InRelease
Hit:3 http://deb.debian.org/debian bullseye-updates InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
23 packages can be upgraded. Run 'apt list --upgradable' to see them.
arseniasc@debian:~$
```

As we can see here, we have 23 packages that can be upgraded. To update the list, type the following command **sudo apt-get dist-upgrade**, and then when prompted type y to confirm the apps upgrades.



```
arseniasc@debian:~$ sudo apt-get dist-upgrade
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
The following NEW packages will be installed:
  linux-image-5.10.0-9-amd64
The following packages will be upgraded:
  bind9-dnswitls bind9-host bind9-libs gir1.2-javascriptcoregtk-4.0
  gir1.2-webkit2-4.0 libavcodec58 libavfilter7 libavformat58 libavresample4
  libavutil56 libjavascriptcoregtk-4.0-18 libldb2 libnss3 libpostproc55
  libsmclient libswresample3 libswscale5 libwbclient0 libwebkit2gtk-4.0-37
  linux-image-amd64 python3-ldb samba-libs tzdata
23 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 0 B/92.5 MB of archives.
After this operation, 304 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Reading changelogs... Done
Preconfiguring packages ...
(Reading database ... 146503 files and directories currently installed.)
Preparing to unpack .../00-libsmclient_2%3a4.13.13+dfsg-1~deb11u2_amd64.deb ...
Unpacking libsmclient:amd64 (2:4.13.13+dfsg-1~deb11u2) over (2:4.13.5+dfsg-2) .
..
Preparing to unpack .../01-python3-ldb_2%3a2.2.3-2~deb11u1_amd64.deb ...
```

When it finished to install the packages, I ran the **sudo apt-get update** again it shows that all packages are up to date.

```
arseniasc@debian: ~  
rseniasc@debian:~$ sudo apt update  
Hit:1 http://deb.debian.org/debian bullseye InRelease  
Hit:2 http://security.debian.org/debian-security bullseye-security InRelease  
Hit:3 http://deb.debian.org/debian bullseye-updates InRelease  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
All packages are up to date.  
rseniasc@debian:~$
```

Now, we are going to set up the password requirements.

First, install **sudo apt install libpam-pwquality** to install password quality checking library.

```
arseniasc@debian: ~  
rseniasc@debian:~$ sudo apt install libpam-pwquality  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
The following NEW packages will be installed:  
  libpam-pwquality  
1 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.  
Need to get 13.8 kB of archives.  
After this operation, 42.0 kB of additional disk space will be used.  
Get:1 http://deb.debian.org/debian bullseye/main amd64 libpam-pwquality amd64 1.4-1 [13.8 kB]  
Fetched 13.8 kB in 0s (65.1 kB/s)  
Selecting previously unselected package libpam-pwquality:amd64.  
(Reading database ... 151274 files and directories currently installed.)  
Preparing to unpack .../libpam-pwquality_1.4-1_amd64.deb ...  
Unpacking libpam-pwquality:amd64 (1.4-1) ...  
Setting up libpam-pwquality:amd64 (1.4-1) ...  
Processing triggers for man-db (2.9.4-2) ...  
rseniasc@debian:~$
```

To change the password expiration, type the following **sudo vi /etc/login.defs**

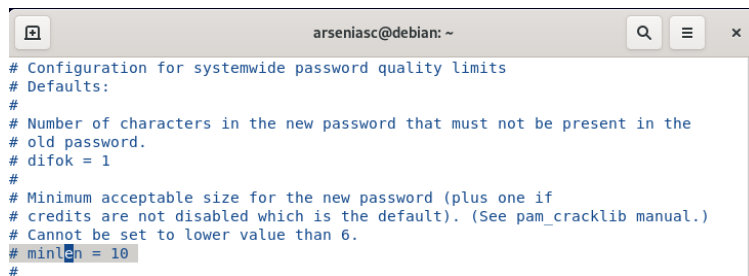
```
UMASK                                022  
  
#  
# Password aging controls:  
#  
#     PASS_MAX_DAYS   Maximum number of days a password may be used.  
#     PASS_MIN_DAYS   Minimum number of days allowed between password changes.  
#     PASS_WARN_AGE   Number of days warning given before a password expires.  
#  
PASS_MAX_DAYS 99999  
PASS_MIN_DAYS 0  
PASS_WARN_AGE 7
```

Change where says **PASS_MAX_DAYS** it has 99999 to 30 days

```
UMASK                                022  
  
#  
# Password aging controls:  
#  
#     PASS_MAX_DAYS   Maximum number of days a password may be used.  
#     PASS_MIN_DAYS   Minimum number of days allowed between password changes.  
#     PASS_WARN_AGE   Number of days warning given before a password expires.  
#  
PASS_MAX_DAYS 30  
PASS_MIN_DAYS 0  
PASS_WARN_AGE 7
```

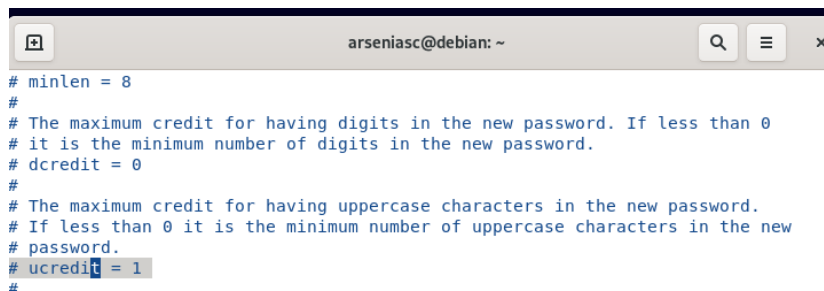
To set up a password length please go to the **sudo vi /etc/security/pwquality.conf**

Look for the line that says **#minlen** it was 8 before I changed it to 10



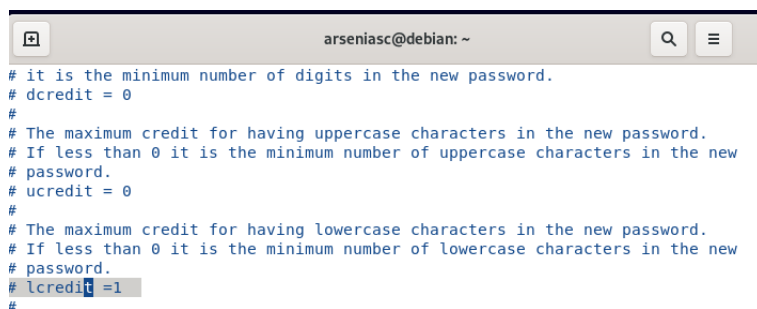
```
arseniasc@debian: ~  
# Configuration for systemwide password quality limits  
# Defaults:  
#  
# Number of characters in the new password that must not be present in the  
# old password.  
# difok = 1  
#  
# Minimum acceptable size for the new password (plus one if  
# credits are not disabled which is the default). (See pam_cracklib manual.)  
# Cannot be set to lower value than 6.  
# minlen = 10  
#
```

To force the user, use at least one upper letter modify where says **#ucredit** I set up it to 1



```
arseniasc@debian: ~  
# minlen = 8  
#  
# The maximum credit for having digits in the new password. If less than 0  
# it is the minimum number of digits in the new password.  
# dcredit = 0  
#  
# The maximum credit for having uppercase characters in the new password.  
# If less than 0 it is the minimum number of uppercase characters in the new  
# password.  
# ucredit = 1  
#
```

To force the user, use at least one upper letter modify where says **#lcredit** I set up it to 1



```
arseniasc@debian: ~  
# it is the minimum number of digits in the new password.  
# dcredit = 0  
#  
# The maximum credit for having uppercase characters in the new password.  
# If less than 0 it is the minimum number of uppercase characters in the new  
# password.  
# ucredit = 0  
#  
# The maximum credit for having lowercase characters in the new password.  
# If less than 0 it is the minimum number of lowercase characters in the new  
# password.  
# lcredit = 1  
#
```

Are they different for each server? the same for both? Are the running instructions any different?

I created my script on CentOS, and when I ran the script, I had issues with the iostat command it did not find this command. I had to install **sudo apt-get install sysstat -y**

Also, I had problem with the **ifconfig**

I installed the **net-tools** and it says the same that the ifconfig command was not found, so when I executed the script with sudo it ran successfully.

If we have all the packages needed to run the script will be the same on both servers.

Script.

```
#!/bin/bash
```

```
#Script that will monitor the health of my server.
```

```
#Print the hostname
```

```
echo "
```

```
*****
```

SERVER HEALTH MONITORING REPORT

```
*****"
```

```
#Print an space for a better look of the information
```

```
echo "
```

```
"
```

```
echo "-----General Information-----"
```

```
echo "
```

```
" > MonitoringScript.txt
```

```
#get just the ip add with the command ifconfig
```

```
Ipadd=$( ifconfig | grep "broadcast" | cut -d " " -f 10 | uniq)
```

```
echo "IP Addresses : " $Ipadd
```

```

#Load average
Load=$(cat /proc/loadavg)
echo "Load Average : " $Load

Hostname=$(hostname)
echo "Hostname : " $Hostname

#get just the kernel version
Kernel=$( hostnamectl | awk '/Kernel/{print $2 " " $3}')
echo "Kernel Version : " $Kernel

#get just the System name
OPname=$( hostnamectl | awk '/Operating System/{print $3 " " $4}')
echo "Operating System : " $OPname

echo "
"

#Print cpu statistics about the cpu
echo '-----CPU Information-----'
echo "
"

#Get just the information about the user usage
cpuusedbyuser=$( iostat -c | awk '/ / {print $1}')
echo "Cpu Used by users: " $cpuusedbyuser

#Get just the information about the system
cpuusedbysys=$( iostat -c | awk '/ / {print $3}')
echo "CPU Used By System : " $cpuusedbysys

#Get just the information about the time when the server did not received any request
norequest=$( iostat -c | awk '/ / {print $6}')

```

```
echo "Cpu With no Request : " $norequest
```

```
echo "
```

```
"
```

```
echo '-----Memory Information-----'
```

```
echo "
```

```
"
```

```
#Get just Total memory of the system
```

```
TotalMem=$( awk '/MemTotal/ {print $2}' /proc/meminfo)
```

```
echo "Total Memory: " $TotalMem
```

```
#Get just the free space on the memory
```

```
Freememo=$( awk '/MemFree/ {print $2}' /proc/meminfo)
```

```
echo "Free Memory : " $Freememo
```

```
#Get just the available space
```

```
AvailableSpa=$( awk '/MemAvailable/ {print $2}' /proc/meminfo)
```

```
echo "Available Space : " $AvailableSpa
```

```
echo "
```

```
"
```

```
echo "-----Active Processes-----"
```

```
#Shows the active processes
```

```
echo "
```

```
"
```

```
#get the result Of the PID
```

```
Process=$(ps | awk '{print $1}')
```

```
echo $Process
```

```

#Got the result of TTY
Process2=$(ps | awk '{print $2}')
echo $Process2

#Get the Result of the Time
Process3=$(ps | awk '{print $3}')
echo $Process3

#Get the result of the CMD
Process4=$(ps | awk '{print $4}')
echo $Process4

echo "
"

#print the information about the disk
echo "-----Disk Information-----"

echo "
"

Disk=$(df -h | awk '/Filesystem/{print $1,"\t",$2,"\t",$3,"\t",$4 }')
echo $Disk

Disk1=$(df -h | awk '/sda/{print $1,"\t",$2,"\t",$3,"\t",$4 }')
echo $Disk1

echo "
"

#Print the date, logged users, system uptime and if is Online or Offline
#Print logged user
echo "--Current Logged Users , System Uptime and Date--"
echo "
"

```

```
Who=$(who | awk '{print $1,"\t",$2 }')
echo "Logged User : " $Who
#print the system Uptime
Uptime=$(uptime | awk '{print $3}' | tr -d ",")
echo "System Up Time : " $Uptime
#Print the current date
Date=$(date | awk '{print $1,$2, $3,$4,$6}' | tr -d ",")
echo "Current Date : " $Date
Conection=$(ping -c1 youtube.com &>/dev/null && echo "online" || echo "offline")
echo "Conection Status : " $Conection
echo "
"
```

Link to my Script

https://github.com/elimelec19/Portafolio/blob/main/Server_Health_Monitor.sh

Sources:

https://www.youtube.com/watch?v=RWKD_5rKLnE&t=517s

<https://www.tecmint.com/assign-linux-command-output-to-variable/>

https://www.server-world.info/en/note?os=Debian_10&p=password