Search for: Search ...

Linux Audit

Linux security: Auditing, Hardening and Compliance

Menu

- Home
- Lynis
- Contact
- About Linux Audit

June 11, 2015 Michael BoelenMalware, System Administration 30 comments

Installing ClamAV on CentOS 7 and Using Freshclam

Install and Configure ClamAV on CentOS 7

Including the usage of Freshclam

To get ClamAV on CentOS installed, we have to use the EPEL repository (Extra Packages for Enterprise Linux). Fortunately, the Fedora project provides this with an easy installation. Unfortunately the default configuration is not properly working. In this post we collect some of the issues and required changes.

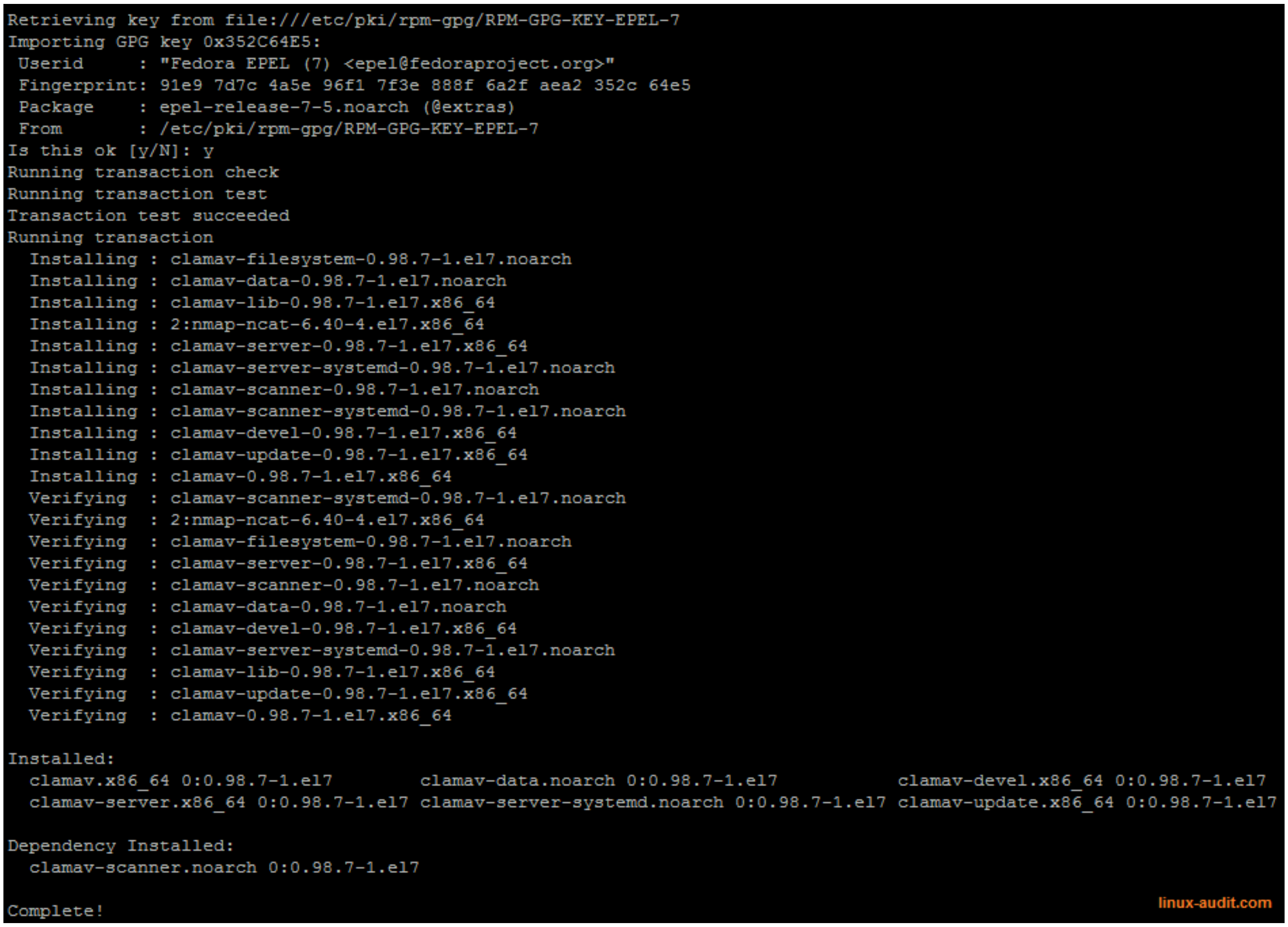
Let’s start with installing the EPEL support.

```
yum install epel-release
```

Next step is installing all ClamAV components.

```
yum install clamav-server clamav-data clamav-update clamav-filesystem clamav clamav-scanner-systemd clamav-devel clamav-lib clamav-server-systemd
```

The output should be similar to:



```
Retrieving key from file:///etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-7
Importing GPG key 0x352C64E5:
 Userid     : "Fedora EPEL (7) <epel@fedoraproject.org>"
 Fingerprint: 91e9 7d7c 4a5e 96f1 7f3e 888f 6a2f aea2 352c 64e5
 Package    : epel-release-7-5.noarch (@extras)
 From       : /etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-7
Is this ok [y/N]: y
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : clamav-filesystem-0.98.7-1.el7.noarch
  Installing : clamav-data-0.98.7-1.el7.noarch
  Installing : clamav-lib-0.98.7-1.el7.x86_64
  Installing : 2:nmap-ncat-6.40-4.el7.x86_64
  Installing : clamav-server-0.98.7-1.el7.x86_64
  Installing : clamav-server-systemd-0.98.7-1.el7.noarch
  Installing : clamav-scanner-0.98.7-1.el7.noarch
  Installing : clamav-scanner-systemd-0.98.7-1.el7.noarch
  Installing : clamav-devel-0.98.7-1.el7.x86_64
  Installing : clamav-update-0.98.7-1.el7.x86_64
  Installing : clamav-0.98.7-1.el7.x86_64
  Verifying  : clamav-scanner-systemd-0.98.7-1.el7.noarch
  Verifying  : 2:nmap-ncat-6.40-4.el7.x86_64
  Verifying  : clamav-filesystem-0.98.7-1.el7.noarch
  Verifying  : clamav-server-0.98.7-1.el7.x86_64
  Verifying  : clamav-scanner-0.98.7-1.el7.noarch
  Verifying  : clamav-data-0.98.7-1.el7.noarch
  Verifying  : clamav-devel-0.98.7-1.el7.x86_64
  Verifying  : clamav-server-systemd-0.98.7-1.el7.noarch
  Verifying  : clamav-lib-0.98.7-1.el7.x86_64
  Verifying  : clamav-update-0.98.7-1.el7.x86_64
  Verifying  : clamav-0.98.7-1.el7.x86_64

Installed:
  clamav.x86_64 0:0.98.7-1.el7        clamav-data.noarch 0:0.98.7-1.el7           clamav-devel.x86_64 0:0.98.7-1.el7
  clamav-server.x86_64 0:0.98.7-1.el7 clamav-server-systemd.noarch 0:0.98.7-1.el7 clamav-update.x86_64 0:0.98.7-1.el7

Dependency Installed:
  clamav-scanner.noarch 0:0.98.7-1.el7

Complete!
```

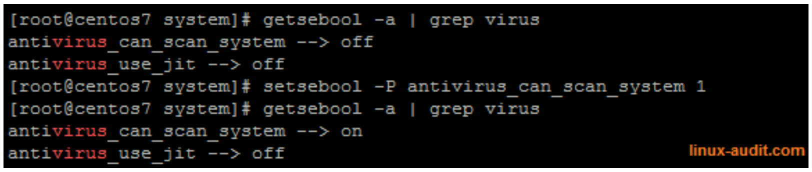
Installing ClamAV with help of EPEL repository

Configure SELinux for ClamAV

If you are using ClamAV on CentOS, together with SELinux, we should configure it a little bit. This way ClamAV can access all files on disk, and update its data definition files.

Enable **antivirus_can_scan_system**:

> setsebool -P antivirus_can_scan_system 1



```
[root@centos7 system]# getsebool -a | grep virus
antivirus_can_scan_system --> off
antivirus_use_jit --> off
[root@centos7 system]# setsebool -P antivirus_can_scan_system 1
[root@centos7 system]# getsebool -a | grep virus
antivirus_can_scan_system --> on
antivirus_use_jit --> off
```

If you don’t perform this step, Freshclam will log something like:

```
During database load : LibClamAV Warning: RWX mapping denied: Can't allocate RWX Memory: Permission denied
```

Configuration of Clam daemon

Copy a the clamd.conf template, in case you don’t have a configuration file yet.

> cp /usr/share/clamav/template/clamd.conf /etc/clamd.d/clamd.conf
> sed -i ‘/^Example/d’ /etc/clamd.d/clamd.conf

Change **/etc/**clamd**.d/**clamd**.conf** file and define if you want to run the scanner as root, or a specific user. Check your /etc/passwd file for the related Clam user.

Change the following two options:

```
User clamscan
LocalSocket /var/run/clamd.<SERVICE>/clamd.sock
```

Enable Freshclam

Freshclam helps with keeping the database of ClamAV up-to-date. First delete the related “Example” line from **/etc/freshclam.conf**.

```
cp /etc/freshclam.conf /etc/freshclam.conf.bak
sed -i ‘/^Example/d’ /etc/freshclam.conf
```

Check the other options in the file, and change it to your preferred settings.

Missing systemd service file

We didn’t get a systemd service file, so creating a quick file here. The process should be forking itself and start freshclam in daemon mode. In this case we configure it to check 4 times a day for new files.

Create a new file **/usr/lib/systemd/system/clam-freshclam.service**

```
# Run the freshclam as daemon
[Unit]
Description = freshclam scanner
After = network.target
[Service]
Type = forking
ExecStart = /usr/bin/freshclam -d -c 4
Restart = on-failure
PrivateTmp = true
[Install]
WantedBy=multi-user.target
```

Now enable and start the service.

```
systemctl enable clam-freshclam.service
```

```
systemctl start clam-freshclam.service
```

Check the status.

```
[root@centos7 system]# systemctl status clam-freshclam.service
clam-freshclam.service - freshclam scanner
Loaded: loaded (/usr/lib/systemd/system/clam-freshclam.service; enabled)
Active: active (running) since Thu 2015-06-11 11:09:24 CEST; 1s ago
Process: 3158 ExecStart=/usr/bin/freshclam -d -c 4 (code=exited, status=0/SUCCESS)
Main PID: 3159 (freshclam)
CGroup: /system.slice/clam-freshclam.service
└─3159 /usr/bin/freshclam -d -c 4
```

Change service files

By default, the service files seem to be messy and not working.

These are the files bundled:

```
[root@centos7 system]# ls -l /usr/lib/systemd/system/clam*
-rw-r--r--. 1 root root 136 Apr 29 20:38 /usr/lib/systemd/system/clamd@scan.service
-rw-r--r--. 1 root root 231 Apr 29 20:38 /usr/lib/systemd/system/clamd@.service
```

When enabling the clamd service, we would see something like this:

```
[root@centos7 system]# systemctl enable /usr/lib/systemd/system/clamd@.service
 Failed to issue method call: Unit /usr/lib/systemd/system/clamd@.service does not exist.
```

So let’s fix it. First rename the **/usr/lib/systemd/system/clamd@.service** file.

Rename the clamd@ file.

```
mv /usr/lib/systemd/system/clamd@.service /usr/lib/systemd/system/clamd.service
```

Now we have to change the clamd@scan service as well, as it refers to a non-existing file now. Change this line in **/usr/lib/systemd/system/clamd@scan.service** and remove the @ sign.

```
.include /lib/systemd/system/clamd@.service
```

Next step is changing the clamd service file **/usr/lib/systemd/system/clamd.service**

```
[Unit]
Description = clamd scanner daemon
After = syslog.target nss-lookup.target network.target
[Service]
Type = simple
ExecStart = /usr/sbin/clamd -c /etc/clamd.d/clamd.conf --nofork=yes
Restart = on-failure
PrivateTmp = true

[Install]
WantedBy=multi-user.target
```

Move into the directory.

```
cd /usr/lib/systemd/system
```

Start all services.

```
[root@centos7 system]# systemctl enable clamd.service
[root@centos7 system]# systemctl enable clamd@scan.service
[root@centos7 system]# systemctl start clamd.service
[root@centos7 system]# systemctl start clamd@scan.service
```

Checking the status

With all these changes, ClamAV on CentOS 7 should be running now. The easiest way to check, is using the *ps* command and see if *freshclam* and *clamd* are running.

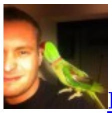
Useful resources for debugging are the systemctl status command, followed by the service. Then there is logging in /var/log/messages, which usually will reveal when and why something is (not) running.

More tips? Leave them in the comments!

- centos 7
- clamav
- clamd
- clamscan
- freshclam

Show me!

30 comments

- Rocco
  July 19, 2015

  4:49 am

  Thx a lot for this article! I searched a lot to get clamd work on CentOS 7!
  Nice greetings from Vienna!

  Reply

  - Michael Boelen
    July 19, 2015

    1:05 pm

    Great to hear it helped for you as well Rocco. Any other topics you like to read on our blog?

    Reply
- Azzinar
  July 28, 2015

  4:21 am

  Hi Mic,

  thank you very much for your posting. i have do that sequence but i have get error

  “Jul 28 09:13:10 localhost.localdomain freshclam[12408]: During database load : LibClamAV Warning: RWX mapping denied: Can’t allocate RWX Memory: Permission denied”

  the error disappear after type command “setsebool -P clamd_use_jit on”

  Best regards / Azzinar

  Reply
- Luis
  August 12, 2015

  8:37 pm

  Hi

  Every time I use this line sed -i ‘/^Example/d’ /etc/freshclam.conf the ‘ gets change to . and it fails and if I use the correct character I get the same error

  Any idea?

  Reply

  - Michael Boelen
    August 15, 2015

    11:59 am

Copy the line manually in a text browser and replace the quotes with single ones. It might be due to WordPress.

Reply

- Emiliano A.
  December 23, 2015

  3:13 am

  You can use vim editor and remove this line with dd command

  Reply

- Felipe
  October 9, 2015

  10:59 pm

  Hi there!

  Very good tutorial! I’m begginer in linux but I could follow this almost complete…
  My doubts are:
  1. You said to change this to any user I want, but which should be better for security matters, root or clamscan?
  2. sed -i ‘/^Example/d’ /etc/clamd.d/clamd.conf (What should happen after using that command?)
  3. You said to change this: LocalSocket /var/run/clamd./clamd.sock. To point I did not understand is, change what and for what value in there?

  Thank you!

  Reply

  - Michael Boelen
    October 10, 2015

    5:24 pm

    Best is using a non-privilged user, to reduce the chance of attacks succeeding and having full permissions.
    The sed command will remove the “Example” line.
    The socket file should point to the right directory, which is determined by the service name in the related example.

    Did that help?

    Reply

    - Felipe
      October 10, 2015

      10:18 pm

      Hi, thanks for the reply!

      To be honest, no XD

      Let me clarify…
      2. sed -i ‘/^Example/d’ /etc/clamd.d/clamd.conf (What should happen after using that command?)
      This command didn’t delete the Exampe (I think…)

      When I get this point “LocalSocket /var/run/clamd./clamd.sock”, the problem is that I didn’t understood what should be changed there and under /var/run the only folder about clam is clamd.scan which there is nothing inside. It means that those socket file doesn’t exist.

      Until now that are my problems! :/

As you can see I’m newbie… trying to lean XD

Reply

- Robert Nadon
  October 13, 2015

  10:12 pm

  As per above, just add this line:
  LocalSocket /var/run/clamd.scan/clamd.sock

  It will create your clamd.sock

  Reply
- Amy Tebbe
  November 24, 2015

  5:08 pm

  Thanks for the directions. I was able to get it working, however, I end up with 2 clamd processes running:

  ```
  # ps -ef|grep clam
  clamupd+ 11639 1 0 Nov23 ? 00:00:10 /usr/bin/freshclam -d -c 4
  clamscan 25171 1 1 09:42 ? 00:00:17 /usr/sbin/clamd -c /etc/clamd.dclamd.conf –nofork=yes
  clamscan 26210 1 99 10:02 ? 00:00:04 /usr/sbin/clamd -c /etc/clamd.dclamd.conf –nofork=yes
  ```

  stopping clamd.service will kill one and I can manually kill -9 the other process, but it always restarts another process.

  Any idea why I’m getting 2 clamd processes? Thanks.

  Reply
  - fassl
    April 20, 2016

    2:23 pm

    I had the same problem, turns out you must not enable both services,

    just enable clamd@scan.service:

    systemctl enable clamd@scan.service
    systemctl disable calmd.service

    Cheers, thans for the tut

    Reply
  - fassl
    April 20, 2016

    2:25 pm

    systemctl disable clamd.service

    then restart

    Reply
- mailpop3
  November 27, 2015

  11:50 pm

  Hi All,
  Thanks to Michael Boelen for this training

  Reply to friends about the following problem:
  #LocalSocket /var/run/clamd./clamd.sock

1) Please first run the command:
# ls ls -la /var/run/

2) Now find the following folder clamd.???
drwx–x—. 2 clamscan clamscan 80 Feb 01 10:27 clamd.scan

3) Open the file /etc/clamd.d/clamd.conf

4) Finde the line: #LocalSocket /var/run/clamd./clamd.sock
5) Now change to: LocalSocket /var/run/clamd.scan/clamd.sock

Thanks.

Reply

- Amy Tsui
December 30, 2015

6:30 pm

I am not sure why you would need to change the name of /usr/lib/systemd/system/clamd@.service
I assume this is a subprocess that is called when /usr/lib/systemd/system/clamd@scan.service is started

You would run into no error if you enable and start /usr/lib/systemd/system/clamd@scan.service without renaming clamd@.service

Thank you for your article. It was helpful.

Reply

- Amy Tsui
December 30, 2015

6:48 pm

if you do:

systemctl list-unit-files –type=service

you will see:

….
clamd@.service static
clamd@scan.service enabled
…..

In particular, "static" means "enabled because something else wants it". Think by analogy to pacman's package install reasons:
enabled :: explicitly installed
static :: installed as dependency
disabled :: not installed

So you should not need to change or enable the clamd@.service

The dependency will just work if you enable and start clamd@scan.service

Reply

- Theo
January 13, 2016

11:13 pm

Nice work. Got me up and running. I wouldn't mind a little more at the end with a few basic commands or to know if we are all set on a reboot to auto-start, how I will know if a virus is detected and such. But I can break out the manual. Their install instructions were all kinds of wrong so glad your post was here.

Reply

  - Michael Boelen
January 15, 2016

4:52 pm

Great to be of help here and good feedback. Let’s help others: can you share the commands you used (after you got things set up)? Then I will add them to the article!

Reply

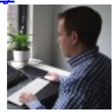
- Chad Vondra
  February 1, 2016

  9:29 pm

  I am dealing with a constant error when the service starts. I have touched clamd.sock then chown’d the directory and the socket to clamupdate:clamupdate. Immediately after I do that it loads properly. Howewver, upon reboot when it fixes the stale socket it the service fails to start again until I re-touch and chown. Any suggestions would be great!

  ```
  Feb 1 14:24:01 mail clamd[4906]: Log file size limited to 1048576 bytes.
  Feb 1 14:24:01 mail clamd[4906]: Reading databases from /var/lib/clamav
  Feb 1 14:24:01 mail clamd[4906]: Not loading PUA signatures.
  Feb 1 14:24:01 mail clamd[4906]: Bytecode: Security mode set to “TrustSigned”.
  Feb 1 14:24:14 mail clamd[4906]: Loaded 4244905 signatures.
  Feb 1 14:24:16 mail clamd[4906]: LOCAL: Socket file /var/run/clamd.scan/clamd.sock could not be bound: Permission denied
  Feb 1 14:24:16 mail clamd: ERROR: LOCAL: Socket file /var/run/clamd.scan/clamd.sock could not be bound: Permission denied
  Feb 1 14:24:16 mail systemd: clamd.service: main process exited, code=exited, status=1/FAILURE
  Feb 1 14:24:16 mail systemd: Unit clamd.service entered failed state.
  ```

  Reply

  - Michael Boelen
    February 3, 2016

    8:01 am

    Check if your temporary directory is properly created via a file in /etc/tmpfiles.d/.

    You could try to force it in your service file:

    ```
    ExecStartPre=/bin/mkdir -p /var/run/clamd.scan
    ExecStartPre=/bin/chown -R clamuser:clamgroup /var/run/clamd.scan
    ```

    Reply

    - Karbas
      April 13, 2016

      5:16 pm

      Have the same problem. Unfortunately forcing directory creation and owner change with ExecStartPre didnt change much. Have noticed, when i manually launch /usr/sbin/clamd -c /etc/clamd.d/clamd.conf –nofork=yes – clamd starts without any errors and socket file is created.

      /etc/clamd.d/clamd.conf has the following options active:

      ```
      LogSyslog yes
      LocalSocket /var/run/clamd/clamd.sock
      User clamscan
      AllowSupplementaryGroups yes
      ```

      ```
      ls -la /var/run/clamd
      total 0
      drwxr-xr-x. 2 clamscan clamscan 40 Apr 13 11:30 .
      drwxr-xr-x. 28 root root 800 Apr 11 16:18 ..
      ```

Any ideas?

Reply

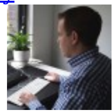
- Temir
  February 5, 2016

  6:02 am

  Hi, when i change line #LocalSocket /var/run/clamd./clamd.sock to:
  LocalSocket /var/run/clamd.scan/clamd.sock
  i get this error ” clamd: ERROR: LOCAL: Socket file /var/run/clamd.scan/clamd.sock is in use by another process.”

  can you tell me where i could make mistake?

  Reply

  - Michael Boelen
    February 7, 2016

    1:31 pm

    Easiest step to validate this is using the *lsof* utility and see what process keeps it open. Let us know when you found it, so it might helps others as well.

    Reply

- Jason Hotchkiss
  February 5, 2016

  4:48 pm

  Hi, thank you so much for this great article! I have the following question:

  How would I configure clamav to automatically kick off a weekly virus scan of the entire hard drive?

  Thanks in advanced!

  Reply

  - Michael Boelen
    February 7, 2016

    1:30 pm

    That would be via a cron job or timer.

    Reply

- Jason Hotchkiss
  February 5, 2016

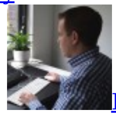
  5:17 pm

  Hello ~ thank you for this great article.

  Now that I have all 3 services up and running, is it possible for me to create an automated weekly scan of the full hard drive (or at least the important areas of the drive)?

Thank you!

Reply



- Michael Boelen
  February 7, 2016

  1:30 pm

  Hi Jason. Sure, you could run the clamscan utility to do a full system scan. Add it to your cronjobs (or timers).

  Reply

- Robert
  February 14, 2016

  10:02 pm

  The freshclam works fine, but when I follow the instructions above here for clamd, it won’t start. When I call the status with: systemctl status clamd.service I see the following:

```
|root@centos72.testdomain.com > /usr/lib/systemd/system |->systemctl status clamd.service
● clamd.service – clamd scanner daemon
Loaded: loaded (/usr/lib/systemd/system/clamd.service; enabled; vendor preset: disabled)
Active: failed (Result: start-limit) since Sun 2016-02-14 21:52:10 CET; 6min ago
Process: 16999 ExecStart=/usr/sbin/clamd -c /etc/clamd.d/clamd.conf –nofork=yes (code=exited,
status=1/FAILURE)
Main PID: 16999 (code=exited, status=1/FAILURE)

Feb 14 21:52:10 centos72.testdomain.com systemd[1]: Unit clamd.service entered failed state.
Feb 14 21:52:10 centos72.testdomain.com systemd[1]: clamd.service failed.
Feb 14 21:52:10 centos72.testdomain.com systemd[1]: clamd.service holdoff time over, scheduling restart.
Feb 14 21:52:10 centos72.testdomain.com systemd[1]: start request repeated too quickly for clamd.service
Feb 14 21:52:10 centos72.rope-parkstad.nl systemd[1]: Failed to start clamd scanner daemon.
Feb 14 21:52:10 centos72.rope-parkstad.nl systemd[1]: Unit clamd.service entered failed state.
Feb 14 21:52:10 centos72.rope-parkstad.nl systemd[1]: clamd.service failed.
```

  Can someone tell me what is going wrong here? I Get the same error when i want to start clamd@scan

  Reply
- James Daniel
  February 15, 2016

  6:39 am

  Hi Michael. This post includes a big error, the service is not missing and didn’t need to be created. Take a look here, it’s solved my problems:
  https://www.adminsys.ch/2015/08/21/installing-clamav-epel-centosred-hat-7-nightmare/

  Reply

- Michael Boelen
  February 15, 2016

  5:07 pm

  Thanks, that might be helpful for others. Not everyone will do their installation the same way, so keeping things up for readers to consider what option they prefer.

  Reply

Leave a Reply

Your email address will not be published. Required fields are marked *

Comment

Name *

Email *

Website

Post Comment

About Linux Audit

This blog is part of our mission: help individuals and companies, to secure their systems, and comply with regulations. We simply love Linux security and system auditing.

Besides the blog, we have our security auditing tool Lynis. Open source, GPL, and free to use.

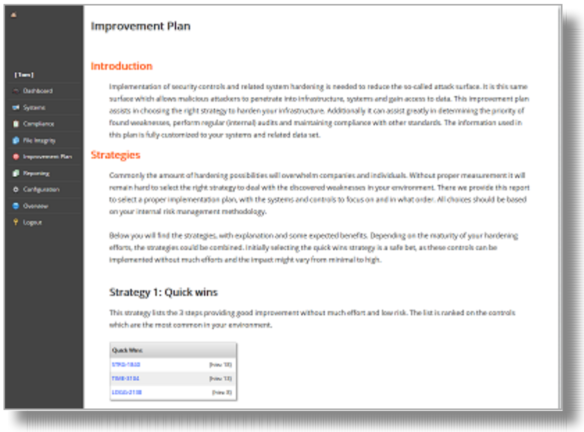
Download Lynis (Free)

For those with enterprise needs, or want to audit multiple systems, there is an Enterprise version.

"*One security solution to audit, harden, and secure your Linux/UNIX systems.*"

Benefits:

- Perform audits within a few minutes
- Central management
- Powerful reporting
- Compliance checks (e.g. PCI DSS)
- Additional plugins and more tests



Improvement Plan

Introduction

Implementation of security controls and related system hardening is needed to reduce the so-called attack surface. It is this same surface which allows malicious attackers to penetrate into infrastructure, systems and gain access to data. This improvement plan assists in choosing the right strategy to harden your infrastructure. Additionally it can assist greatly in determining the priority of found weaknesses, perform regular (internal) audits and maintaining compliance with other standards. The information used in this plan is fully customized to your systems and related data set.

Strategies

Commonly the amount of hardening possibilities will overwhelm companies and individuals. Without proper measurement it will remain hard to select the right strategy to deal with the discovered weaknesses in your environment. There we provide this report to select a proper implementation plan, with the systems and controls to focus on and in what order. All choices should be based on your internal risk management methodology.

Below you will find the strategies, with explanation and some expected benefits. Depending on the maturity of your hardening efforts, the strategies could be combined. Initially selecting the quick wins strategy is a safe bet, as these controls can be implemented without much efforts and the impact might vary from minimal to high.

Strategy 1: Quick wins

This strategy lists the 3 steps providing good improvement without much effort and low risk. The list is ranked on the controls which are the most common in your environment.

Quick Wins	
STRG-1840	[view 13]
TIME-3104	[view 13]
LOGG-2138	[view 3]

Learn More

Enjoy the articles!

Recent Posts

- Vulnerability Scanning: The Destiny to Disappointment?

- How the web changes with HTTP/2: Performance and Security
- How Linux Security Fails to be Simple
- Linux DNS Tuning for Performance and Resilience
- What is the ‘toor’ user on FreeBSD?
- Understanding Linux Privilege Escalation and Defending Against It
- Troubleshooting Linux Time Synchronization with NTP
- The Non-Technical Changelog: Insights of 6 Months Development
- Upgrading External Packages with unattended-upgrade
- Find and Disable Insecure Services on Linux

Receive Updates

Outsmart your colleagues and receive the latest updates.

Email*
you@example.com
Subscribe

Or follow on the social media channels:

Contact

This blog is part of our mission to share valuable tips about Linux security. We are reachable via @linuxaudit

Company Details

CISOfy
De Klok 28,
5251 DN, Vlijmen, The Netherlands
+31202260055

Website: cisofy.com