



CENTOS

 5

How to Create and Run New Service Units in Systemd Using Shell Script

by Ioannis Koustoudis | Published: February 17, 2016 | Last Updated: September 7, 2016



Download Your Free eBooks NOW - [10 Free Linux eBooks for Administrators](#) | [4 Free Shell Scripting eBooks](#)



(247newsaktuell.de)

Few days ago, I came across a **Centos 7** 32-bit distro and I felt the desire to test it on an old 32-bit machine. After booting I realized that it had a bug and it was losing the network connection, which I had to turn it **“up”** manually every time after boot. So, the question was how could I set a script doing this job, running every time I boot my machine?

Well, this is very simple and I'll show you the systemd-way using service units. But first a small introduction to service units.

In this article, I'm going to explain what a **“service unit”** in systemd is, how easy is to create and run one. I will try to simplify what **“targets”** are, why we call them **“collections of units”** and what are their **“wants”**. Finally we are taking

advantage of a service unit to run our own script after the boot procedure.

It's obvious that your computer is useful due to the services it offers and in order to have this functionality, many services have to be called as the computer boots and reaches different levels. Other services are called to be executed when the computer reaches for example the rescue level (**runlevel 0**) and others when it reaches the multi-user level (**runlevel 3**). You can imagine these levels as **targets**.

In a simple way target is a collection of service units. If you want to have a look at service units running in your graphical.target level, type:

```
# systemctl --type=service
```

```
colord.service      loaded active running Manage, Install and Generate Color Profil
crond.service       loaded active running Command Scheduler
cups.service        loaded active running CUPS Printing Service
dbus.service        loaded active running D-Bus System Message Bus
firewalld.service   loaded active running firewalld - dynamic firewall daemon
gdm.service         loaded active running GNOME Display Manager
gssproxy.service    loaded active running GSSAPI Proxy Daemon
iscsi-shutdown.service loaded active exited Logout off all iSCSI sessions on shutdown
kmod-static-nodes.service loaded active exited Create list of required static device nod
ksm.service         loaded active exited Kernel Samepage Merging
ksmtuned.service    loaded active running Kernel Samepage Merging (KSM) Tuning Daem
libstoragemgmt.service loaded active running libstoragemgmt plug-in server daemon
libvirtd.service    loaded active running Virtualization daemon
lvm2-lvmetad.service loaded active running LVM2 metadata daemon
lvm2-monitor.service loaded active exited Monitoring of LVM2 mirrors, snapshots etc
lvm2-pvscan@8:17.service loaded active exited LVM2 PV scan on device 8:17
lvm2-pvscan@8:18.service loaded active exited LVM2 PV scan on device 8:18
lvm2-pvscan@8:19.service loaded active exited LVM2 PV scan on device 8:19
lvm2-pvscan@8:2.service loaded active exited LVM2 PV scan on device 8:2
lvm2-pvscan@8:33.service loaded active exited LVM2 PV scan on device 8:33
lvm2-pvscan@8:34.service loaded active exited LVM2 PV scan on device 8:34
lvm2-pvscan@8:37.service loaded active exited LVM2 PV scan on device 8:37
ModemManager.service loaded active running Modem Manager
netcf-transaction.service loaded active exited Rollback uncommitted netcf network config
network.service     loaded active exited LSB: Bring up/down networking
NetworkManager.service loaded active running Network Manager
nfs-config.service  loaded active exited Preprocess NFS configuration
packagekit.service loaded active running PackageKit Daemon
polkit.service       loaded active running Authorization Manager
postfix.service      loaded active running Postfix Mail Transport Agent
rhel-dmesg.service   loaded active exited Dump dmesg to /var/log/dmesg
rhel-import-state.service loaded active exited Import network configuration from initram
rhel-readonly.service loaded active exited Configure read-only root support
```

Lines 14-46/80 59%

As you can see some services are active and “**running**” all the time, while others run one-time and terminate (exited). If you want to check the status of a service, type:

```
# systemctl status firewalld.service
```

```
[root@fws ~]# systemctl status firewalld.service
● firewalld.service - firewalld - dynamic firewall daemon
   Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled; vendor preset: enabled)
   Active: active (running) since Tue 2016-02-16 08:37:58 EET; 3h 41min ago
   Main PID: 811 (firewalld)
   CGroup: /system.slice/firewalld.service
           └─811 /usr/bin/python -Es /usr/sbin/firewalld --nofork --nopid

Feb 16 08:37:41 fws.ach.sch.gr systemd[1]: Starting firewalld - dynamic firewall daemon...
Feb 16 08:37:58 fws.ach.sch.gr systemd[1]: Started firewalld - dynamic firewall daemon.
[root@fws ~]#
```

Check Status of Service in CentOS 7

As you can see I checked the status of `firewalld.service` (tip: you can use the auto-complete for the name of the service). It informs me that **firewalld** service is running all the time and it is enabled.

Don't Miss: [How to Configure FirewallD Service in CentOS 7](#)

Enabled and disabled means the service will be permanently loaded or not, during the next boot respectively. On the other hand to start and stop a service has the limitation of the present session and it's not permanent.

For example, if you type:

```
# systemctl stop firewalld.service
# systemctl status firewalld.service
```

```
[root@fws ~]# systemctl stop firewalld.service
[root@fws ~]# systemctl status firewalld.service
● firewalld.service - firewalld - dynamic firewall daemon
   Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled; vendor preset: enabled)
   Active: inactive (dead) since Tue 2016-02-16 12:30:14 EET; 1min 49s ago
   Process: 811 ExecStart=/usr/sbin/firewalld --nofork --nopid $FIREWALLD_ARGS (code=exited, status=0/SUCCESS)
   Main PID: 811 (code=exited, status=0/SUCCESS)

Feb 16 08:37:41 fws.ach.sch.gr systemd[1]: Starting firewalld - dynamic firewall daemon...
Feb 16 08:37:58 fws.ach.sch.gr systemd[1]: Started firewalld - dynamic firewall daemon.
Feb 16 12:30:13 fws.ach.sch.gr systemd[1]: Stopping firewalld - dynamic firewall daemon...
Feb 16 12:30:14 fws.ach.sch.gr systemd[1]: Stopped firewalld - dynamic firewall daemon.
Feb 16 12:31:49 fws.ach.sch.gr systemd[1]: Stopped firewalld - dynamic firewall daemon.
[root@fws ~]#
```

Manage Services in CentOS 7

You can see that the `firewalld.service` is inactive (dead) but it is still enabled, which means that during next boot it will be loaded. So if we want a service to be loaded during boot time in the future we must enable it. What a

great conclusion! Lets create one, it's easy.

If you go to the folder:

```
# cd /etc/systemd/system
# ls -l
```

```
[root@fwsn system]# cd /etc/systemd/system/
[root@fwsn system]# ls -l
total 12
drwxr-xr-x. 2 root root 54 Dec 9 11:57 basic.target.wants
drwxr-xr-x. 2 root root 30 Dec 9 11:54 bluetooth.target.wants
lrwxrwxrwx. 1 root root 41 Dec 9 11:54 dbus-org.bluez.service -> /usr/lib/systemd/system/bluetooth.service
lrwxrwxrwx. 1 root root 41 Dec 9 11:52 dbus-org.fedoraproject.FirewallD1.service -> /usr/lib/systemd/system/firewalld.service
lrwxrwxrwx. 1 root root 44 Dec 9 11:56 dbus-org.freedesktop.Avahi.service -> /usr/lib/systemd/system/avahi-daemon.service
lrwxrwxrwx. 1 root root 44 Dec 9 11:57 dbus-org.freedesktop.ModemManager1.service -> /usr/lib/systemd/system/ModemManager.service
lrwxrwxrwx. 1 root root 46 Dec 9 11:52 dbus-org.freedesktop.NetworkManager.service -> /usr/lib/systemd/system/NetworkManager.service
lrwxrwxrwx. 1 root root 57 Dec 9 11:52 dbus-org.freedesktop.nm-dispatcher.service -> /usr/lib/systemd/system/NetworkManager-dispatcher.service
lrwxrwxrwx. 1 root root 36 Dec 9 12:01 default.target -> /lib/systemd/system/graphical.target
drwxr-xr-x. 2 root root 85 Dec 9 11:51 default.target.wants
lrwxrwxrwx. 1 root root 35 Dec 9 11:55 display-manager.service -> /usr/lib/systemd/system/gdm.service
drwxr-xr-x. 2 root root 31 Dec 9 11:51 getty.target.wants
drwxr-xr-x. 2 root root 63 Dec 9 11:51 graphical.target.wants
drwxr-xr-x. 2 root root 4096 Jan 26 09:31 multi-user.target.wants
drwxr-xr-x. 2 root root 25 Dec 9 11:53 printer.target.wants
drwxr-xr-x. 2 root root 30 Dec 18 10:24 remote-fs.target.wants
drwxr-xr-x. 2 root root 4096 Dec 9 11:56 sockets.target.wants
drwxr-xr-x. 2 root root 35 Dec 9 11:56 spice-vdagentd.target.wants
drwxr-xr-x. 2 root root 4096 Dec 18 10:24 sysinit.target.wants
drwxr-xr-x. 2 root root 43 Dec 18 10:24 system-update.target.wants
[root@fwsn system]#
```

SystemD System Files

You can see some link files of unit services and some directories of the **"wants"** of a target. For example: what the multi-user target wants to be loaded when the boot procedure reaches its level, is listed in the directory with name `/etc/systemd/system/multi-user.target.wants/`.

```
# ls multi-user.target.wants/
```

```
[root@fwsn system]# ls multi-user.target.wants/
abrt-ccpp.service  crond.service      mdmonitor.service  smartd.service
abrttd.service     cups.path           ModemManager.service  sshd.service
abrt-oops.service  hypervkvpd.service netcf-transaction.service  sysstat.service
abrt-vmcore.service  hypervvssd.service NetworkManager.service  tuned.service
abrt-xorg.service   irqbalance.service nfs-client.target      vmtotlsd.service
atd.service         ksm.service        postfix.service       vsftpd.service
auditd.service      ksmtuned.service   remote-fs.target
avahi-daemon.service  libstoragemgmt.service  rngd.service
chronyd.service      libvirtd.service     rsyslog.service
```

Multi User Targets Services

As you can see it doesn't contain only **services** but also other **targets** which are also collections of services.

Let's make a service unit with the name **connection.service**.

```
# vim connection.service
```

and type the following (hit **"i"** for insert mode), save it and exit (with **"esc"** and **":wq!"**):

```
[Unit]
Description = making network connection up
After = network.target

[Service]
ExecStart = /root/scripts/conup.sh

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

```
[Unit]
Description = making network connection up
After = network.target

[Service]
ExecStart = /root/scripts/conup.sh

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

Create New Service Units in CentOS 7

To explain the above: we have created a unit of service type (you can also create units of target type), we have set it to be loaded after the **network.target** (you can understand that the booting procedure reaches the targets with a defined order) and we want every-time the service starts to execute a bash script with the name **conup.sh** which we are going to create.

The fun starts with the last part **[install]**. It tells that it will be wanted by **"multi-user.target"**. So if we enable our service a symbolic link to that service will be created inside the **multi-user.target.wants** folder! Got it? And if we disable it that link will be deleted. So simple.

Just enable it and check:

```
# systemctl enable connection.service
```

it informs us that the symbolic link in the **multi-user.target.wants** folder has been created. Check it:

```
# ls multi-user.target.wants/
```

```
[root@fwsn system]# systemctl enable connection.service
Created symlink from /etc/systemd/system/multi-user.target.wants/connection.service to /etc/systemd/system/connection.service.
[root@fwsn system]# ls multi-user.target.wants/
abrt-ccpp.service      connection.service      libvirtd.service       rsyslog.service
abrt-d.service         crond.service          mdmonitor.service      smartd.service
abrt-oops.service      cups.path              ModemManager.service  sshd.service
abrt-vmcore.service    hypervkvpd.service    netcf-transaction.service sysstat.service
abrt-xorg.service      hypervvssd.service    NetworkManager.service tuned.service
atd.service            irqbalance.service    nfs-client.target      vmtoolsd.service
auditd.service         ksm.service           postfix.service        vsftpd.service
avahi-daemon.service   ksmtuned.service      remote-fs.target
chronyd.service        libstoragemgmt.service rngd.service
```

Enable Service in CentOS 7

As you can see “**connection.service**” is ready for next booting, but we must create the script file first.

```
# cd /root
# mkdir scripts
# cd scripts
# vim conup.sh
```

Add the following line inside **vim** and save it:

```
#!/bin/bash
nmcli connection up enp0s3
```

Of course if you want your script to execute something else, you could type whatever you want instead of the second line.

For example,

```
#!/bin/bash
touch /tmp/testbootfile
```

that would create a file inside **/tmp** folder (just to check that your service is working).

We must also make the script executable:

```
# chmod +x conup.sh
```

Now we are ready. If you don't want to wait until next boot (it's already enabled) we can start the service for the current session typing:

```
# systemctl start connection.service
```

Voila! My connection is up and running!

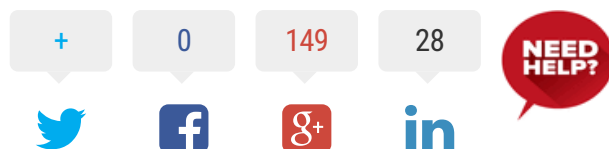
If you 've chosen to write the command **"touch /tmp/testbootfile"** inside the script, just to check its functionality, you will see this file created inside /tmp folder.

```
[artemis@fwsn Desktop]$ ls /tmp/
hogsuspend
ssh-KzoFW4rK6LFh
ssh-sRRi2Jbrs9Qj
ssh-xneg84idaddM
systemd-private-0141d2f78456404d946225b64f4cd8ff-colord.service-J0v6xj
systemd-private-0141d2f78456404d946225b64f4cd8ff-cups.service-ntQXh8
systemd-private-0141d2f78456404d946225b64f4cd8ff-rtkit-daemon.service-txavUo
systemd-private-741ab8f5683f4bae9c0f5de67fdd9328-colord.service-zwQunC
systemd-private-741ab8f5683f4bae9c0f5de67fdd9328-cups.service-sZPF8S
systemd-private-741ab8f5683f4bae9c0f5de67fdd9328-rtkit-daemon.service-fjrbor
systemd-private-ce6fea880a3b47288973ac4ea41374a0-colord.service-2iVndM
systemd-private-ce6fea880a3b47288973ac4ea41374a0-cups.service-Yen0hV
systemd-private-ce6fea880a3b47288973ac4ea41374a0-rtkit-daemon.service-m4Hdkt
testbootfile
tracker-extract-files.1000
tracker-extract-files.1020
[artemis@fwsn Desktop]$
```

Confirm Service Status

I really hope to help you figure out what services, wants, targets and running scripts during booting is all about.

SHARE



If You Appreciate What We Do Here On TecMint, You Should Consider:

1. Stay Connected to: [Twitter](#) | [Facebook](#) | [Google Plus](#)
2. Subscribe to our email updates: [Sign Up Now](#)
3. Use our [Linode referral link](#) if you plan to buy VPS (it starts at only \$10/month).
4. Support us via PayPal donate - [Make a Donation](#)
5. Support us by [purchasing our premium books](#) in PDF format.
6. Support us by taking our [online Linux courses](#)

We are thankful for your never ending support.

Tags: [systemd](#)



Ioannis Koustoudis

[View all Posts](#)

Ioannis Koustoudis is a LFCS Linux sysadmin from Kavala, Greece. He works for the ministry of education and supports almost 200 school units in their infrastructure. If he is not in front of a computer screen, he plays music (he is a multi-instrumentalist) or take care of his two lovely kids.

Your name can also be listed here. Got a tip? [Submit it here](#) to

become an TecMint author.

 **Tecmint**
Linux Training Center

Red Hat RHCSA/RHCE Certification
Preparation Study Guide

 **redhat**

RedHat RHCSA / RHCE 7
RHCSA (EX200) and RHCE (EX300) exams

* EX200 - Red Hat Certified System Administrator (RHCSA)
* EX300 - Red Hat Certified Engineer (RHCE)

Buy Now \$35.00

 **Tecmint**
Linux Training Center

Linux Foundation LFCS / LFCE
Certification Preparation Guide

Linux Foundation Certified System Administrator (LFCS)
Linux Foundation Certified Engineer (LFCE)

Buy Now \$39.99



**The Complete Linux
Administrator Course**
(\$69 Only)



Become an Ethical Hacker (\$49)

NEXT STORY

Deal: Get 3-Year Subscription to Codeanywhere Powerful Cloud-Based Editor \$59

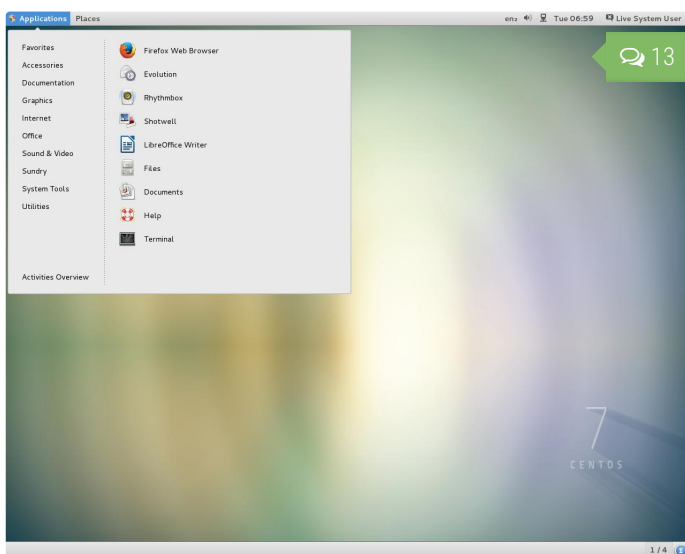


PREVIOUS STORY

5 Ways to Keep Remote SSH Sessions and Processes Running After Disconnection



YOU MAY ALSO LIKE...





How Do I Access or Mount Windows/USB NTFS Partition in RHEL/CentOS/Fedora

7 FEB, 2013

Download of The Day: CentOS 7 Released – Download
DVD ISO Images

8 JUL, 2014

5 RESPONSES

 **Comments** **5**  **Pingbacks** **0**

md  September 8, 2016 at 2:38 am

What you outline in this article wont work until you run 'systemctl daemon-reload'. This needs to be done when you add or change unit files.

Reply



Daniel  September 7, 2016 at 12:28 pm

Could have shared it.. but you spelled systemd wrong in the title. It should not have a capital "D". You could also have executed a command instead of running a script. It is a good example though.

Reply

Fer Nando  May 2, 2016 at 6:58 pm

Very useful!!!! Thanks so much!!!

Reply

Jalal Hajigholamali  February 28, 2016 at 12:28 pm

Hi,

Very useful and interesting article,

Thanks a lot

Reply



Ioannis Koustoudis  February 29, 2016 at 12:24 pm

Thank you Jalal,

feeling great to be useful to you

Reply

GOT SOMETHING TO SAY? JOIN THE DISCUSSION.

Comment

Name *

Email *

Website

☐

Notify me of followup comments via e-mail. You can also [subscribe](#) without commenting.

Post Comment

I ♥ TecMint :



 **BEGINNER'S GUIDE FOR LINUX**  Start learning Linux in minutes →



 **Vi/Vim Editor BEGINNER'S GUIDE**  Learn vi/vim as a Full Text Editor 



[Advertise Here](#)

 **Linux Foundation Certification**  Exam Study Guide to **LFCS** and **LFCE**



How to Add Linux Host to Nagios Monitoring Server Using NRPE Plugin

Nagios 4.2.0 Released – Install on RHEL/CentOS 7.x/6.x/5.x and Fedora 24-19

Install Cacti (Network Monitoring) on RHEL/CentOS 7.x/6.x/5.x and Fedora 24-12

Google Chrome 56 Released – Install on RHEL/CentOS 7/6 and Fedora 25-20

Ebook: Introducing the RHCSA and RHCE Exam Preparation Guide

Install Latest Apache 2.4, MySQL 5.5/MariaDB 10.1 and PHP 5.5/5.6 on RHEL/CentOS 7/6 & Fedora 24-18



The image shows the cover of a study guide for Red Hat RHCSA and RHCE 7 exams. The cover is red with white and black text. At the top is the Tecmint logo. Below it is the title 'Red Hat RHCSA/RHCE Certification Preparation Study Guide'. In the center is the Red Hat logo. Below the logo is the text 'RedHat RHCSA / RHCE 7' and 'RHCSA (EX200) and RHCE (EX300) exams'. At the bottom, it lists the exams: '* EX200 - Red Hat Certified System Administrator (RHCSA)' and '* EX300 - Red Hat Certified Engineer (RHCE)'. A large white box at the bottom says 'Buy Now \$35.00'.

Linux System Administrator Bundle with 7-Courses (96% off)

Add to Cart - \$69

🕒 Ending In: 3 days

Computer Hacker Professional Certification Course (96% Off)

Add to Cart - \$59

🕒 Ending In: 4 days

DOWNLOAD FREE LINUX EBOOKS

- Complete Linux Command Line Cheat Sheet
- The GNU/Linux Advanced Administration Guide
- Securing & Optimizing Linux Servers
- Linux Patch Management: Keeping Linux Up To Date
- Introduction to Linux – A Hands on Guide
- Understanding the Linux® Virtual Memory Manager
- Getting Started with Ubuntu 16.04
- A Newbie's Getting Started Guide to Linux
- Linux from Scratch – Create Your Own Linux OS
- Linux Shell Scripting Cookbook, Second Edition
- Securing & Optimizing Linux: The Hacking Solution
- User Mode Linux – Understanding and Administration



Never Miss Any Linux Tutorials, Guides, Tips and Free eBooks

Join Our Community Of **150,000+ Linux Lovers** and get a weekly newsletter in your inbox

YES! SIGN ME UP

LINUX MONITORING TOOLS

How to Add Linux Host to Nagios Monitoring Server Using NRPE Plugin

8 NOV, 2013

Psensor – A Graphical Hardware Temperature Monitoring Tool for Linux

2 JUL, 2015

Configure Collectd as a Central Monitoring Server for Clients

30 JUN, 2015

Suricata 1.4.4 Released – A Network Intrusion Detection, Prevention and Security Monitoring System

23 JUL, 2013

LINUX INTERVIEW QUESTIONS

10 Useful 'Interview Questions and Answers' on Linux Shell Scripting

26 MAY, 2014

25 Apache Interview Questions for Beginners and Intermediates

8 JAN, 2014

10 Core Linux Interview Questions and Answers

26 JAN, 2014

10 Useful SSH (Secure Shell) Interview Questions and Answers

15 MAR, 2014

OPEN SOURCE TOOLS

Nikto – A Web Application Vulnerability and CGI Scanner for Web Servers

10 JAN, 2014

Suricata 1.4.4 Released – A Network Intrusion Detection, Prevention and Security Monitoring System

23 JUL, 2013

How to Monitor Keyboard Keystrokes Using 'LogKeys' in Linux

23 JAN, 2014

Linux is an Art – Driving Force Behind Linux

20 MAY, 2013



Tecmint: Linux Howtos, Tutorials & Guides © 2017. All Rights Reserved.

This work is licensed under a (cc) BY-NC

The material in this site cannot be republished either online or offline, without our permission.



Check: [Introducing the RHCSA and RHCE Exam Preparation Guide](#)

