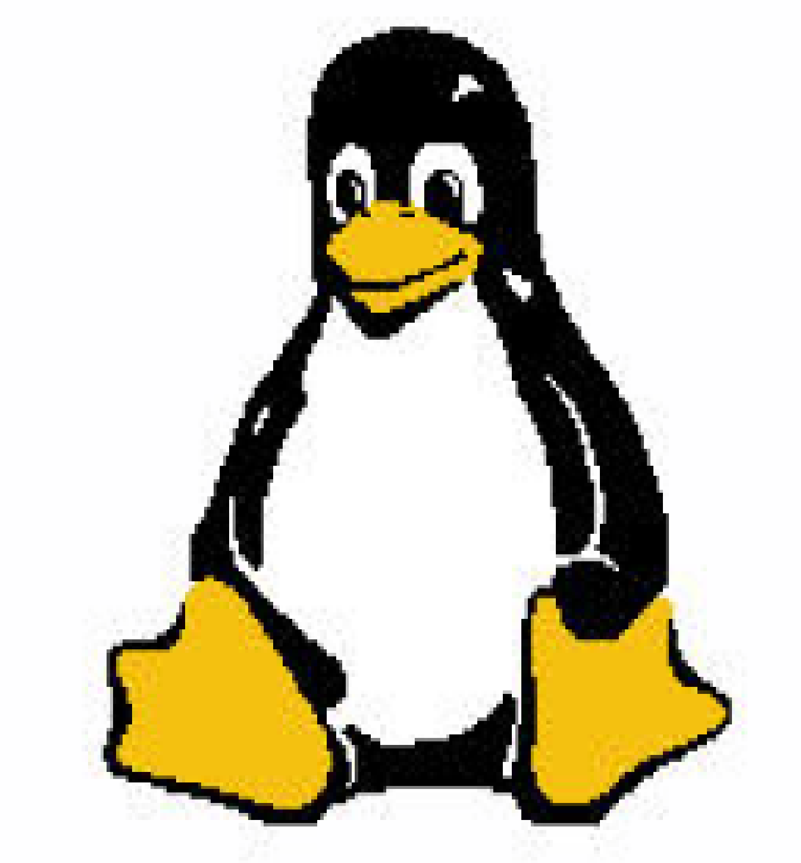
Linux Computer Security



Disabling Unnecessary Services

Reduce the Attack Surface!

What is an Attack Surface?

Attack Surface

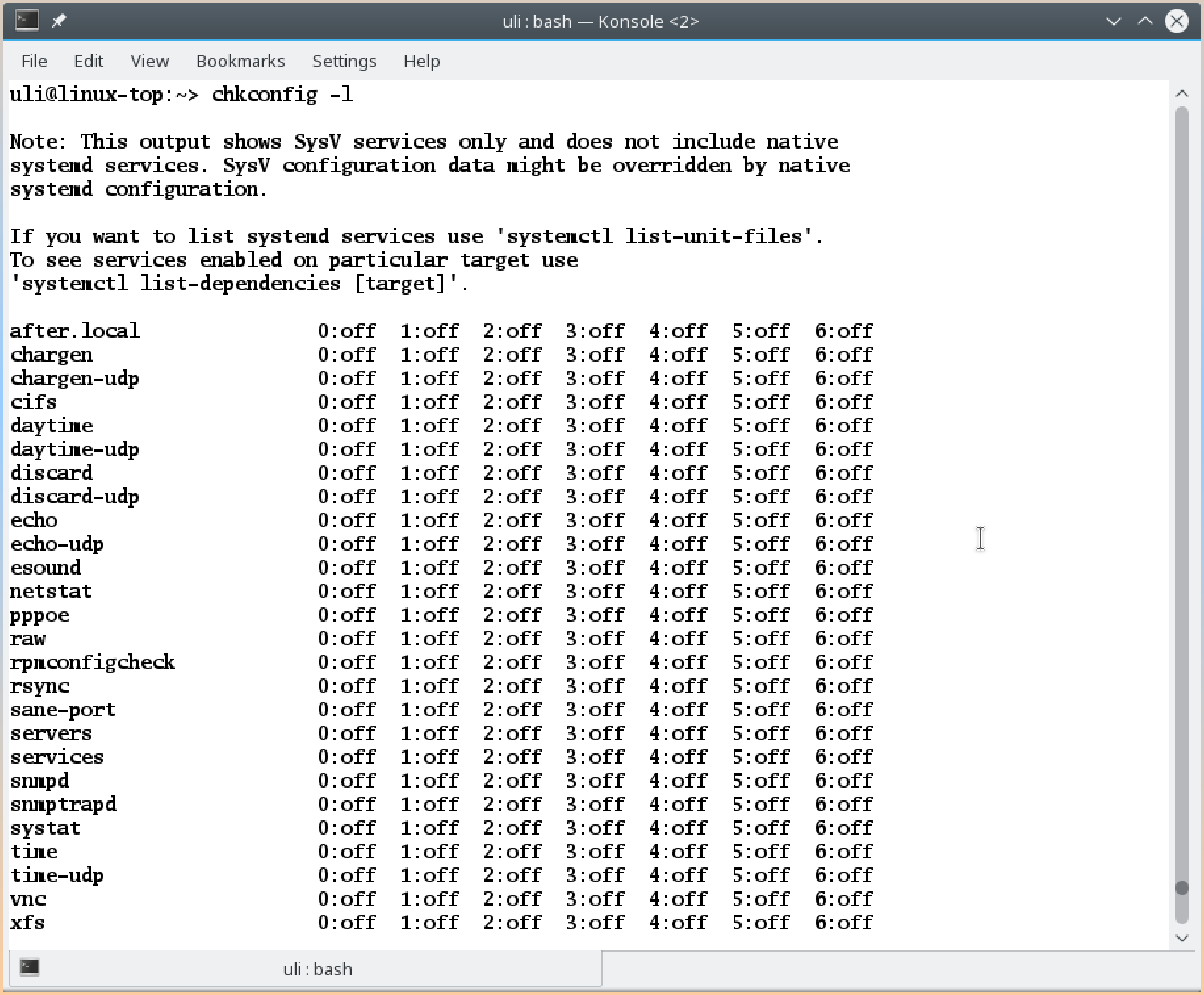
To reduce the possibility of an attack the attack surface has to be as small as possible.

One possibillity to reduce the attack surface on your computer is to run only services (programmes) that you need. If you have installed more services you should either uninstall or disable them.

What services are running on our computer?

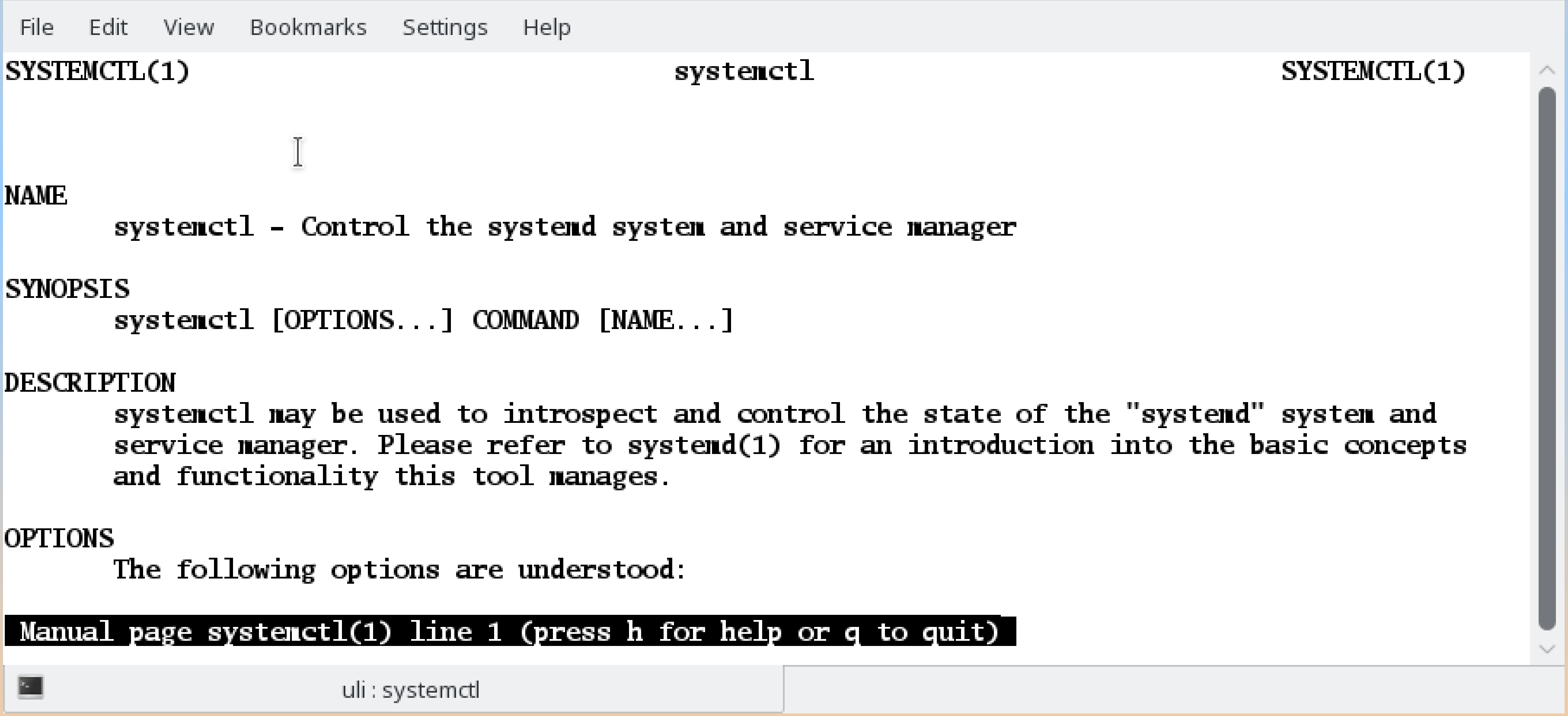
On old systems …….

The **`chkconfig`** command option `-l` for listing services.



```
uli@linux-top:~> chkconfig -l

Note: This output shows SysV services only and does not include native
systemd services. SysV configuration data might be overridden by native
systemd configuration.

If you want to list systemd services use 'systemctl list-unit-files'.
To see services enabled on particular target use
'systemctl list-dependencies [target]'.

after.local               0:off  1:off  2:off  3:off  4:off  5:off  6:off
chargen                   0:off  1:off  2:off  3:off  4:off  5:off  6:off
chargen-udp               0:off  1:off  2:off  3:off  4:off  5:off  6:off
cifs                      0:off  1:off  2:off  3:off  4:off  5:off  6:off
daytime                   0:off  1:off  2:off  3:off  4:off  5:off  6:off
daytime-udp               0:off  1:off  2:off  3:off  4:off  5:off  6:off
discard                   0:off  1:off  2:off  3:off  4:off  5:off  6:off
discard-udp               0:off  1:off  2:off  3:off  4:off  5:off  6:off
echo                      0:off  1:off  2:off  3:off  4:off  5:off  6:off
echo-udp                  0:off  1:off  2:off  3:off  4:off  5:off  6:off
esound                    0:off  1:off  2:off  3:off  4:off  5:off  6:off
netstat                   0:off  1:off  2:off  3:off  4:off  5:off  6:off
pppoe                     0:off  1:off  2:off  3:off  4:off  5:off  6:off
raw                       0:off  1:off  2:off  3:off  4:off  5:off  6:off
rpmconfigcheck            0:off  1:off  2:off  3:off  4:off  5:off  6:off
rsync                     0:off  1:off  2:off  3:off  4:off  5:off  6:off
sane-port                 0:off  1:off  2:off  3:off  4:off  5:off  6:off
servers                   0:off  1:off  2:off  3:off  4:off  5:off  6:off
services                  0:off  1:off  2:off  3:off  4:off  5:off  6:off
snmpd                     0:off  1:off  2:off  3:off  4:off  5:off  6:off
snmptrapd                 0:off  1:off  2:off  3:off  4:off  5:off  6:off
systat                    0:off  1:off  2:off  3:off  4:off  5:off  6:off
time                      0:off  1:off  2:off  3:off  4:off  5:off  6:off
time-udp                  0:off  1:off  2:off  3:off  4:off  5:off  6:off
vnc                       0:off  1:off  2:off  3:off  4:off  5:off  6:off
xfs                       0:off  1:off  2:off  3:off  4:off  5:off  6:off
```

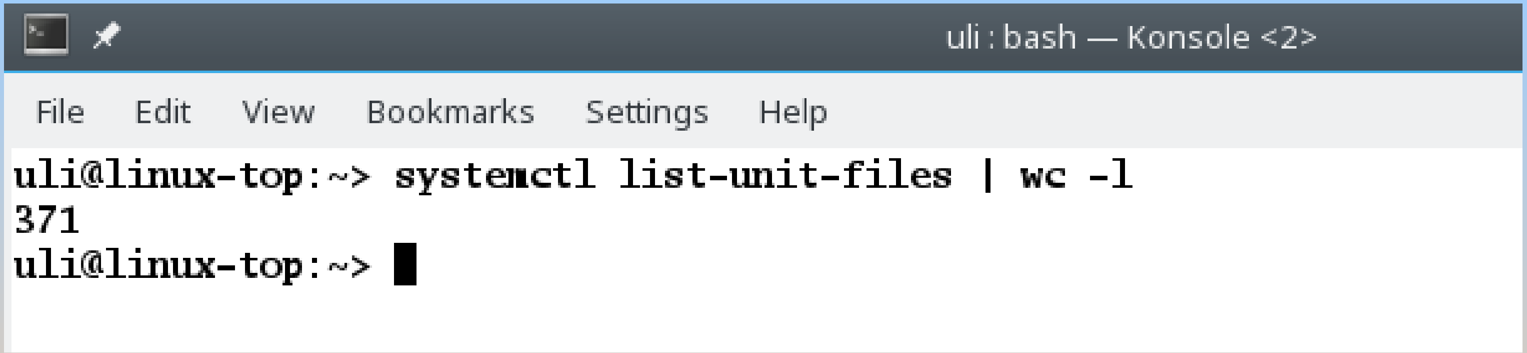
Today the `systemctl` command is used



```
SYSTEMCTL(1)                          systemctl                          SYSTEMCTL(1)



NAME
       systemctl - Control the systemd system and service manager

SYNOPSIS
       systemctl [OPTIONS...] COMMAND [NAME...]

DESCRIPTION
       systemctl may be used to introspect and control the state of the "systemd" system and
       service manager. Please refer to systemd(1) for an introduction into the basic concepts
       and functionality this tool manages.

OPTIONS
       The following options are understood:

 Manual page systemctl(1) line 1 (press h for help or q to quit)
```

How many services are installed?

To check how many services are installed we can pipe the **`systemctl list-unit-files`** command through **`wc -l`** (word count with the lines option).

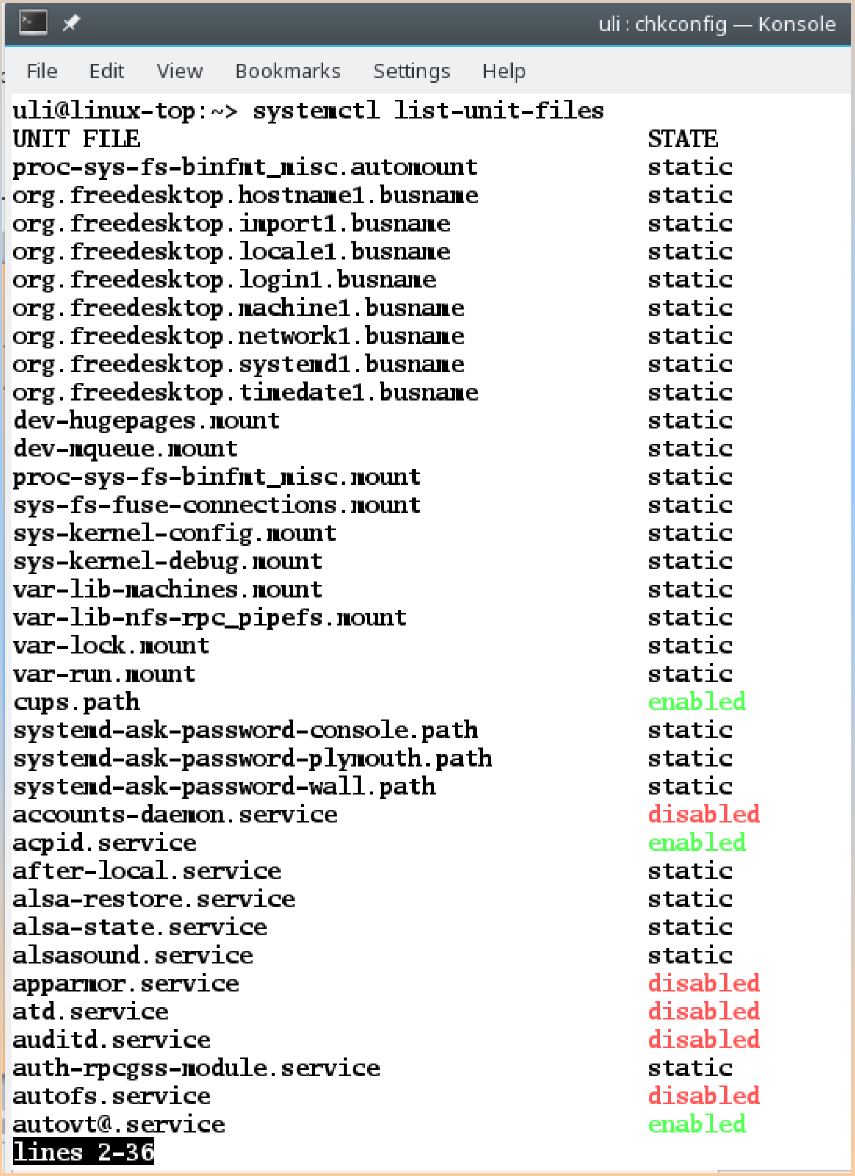


```
uli@linux-top:~> systemctl list-unit-files | wc -l
371
uli@linux-top:~> 
```

How many services are running?

Here we see 3 state options for services: enabled, disabled or static.

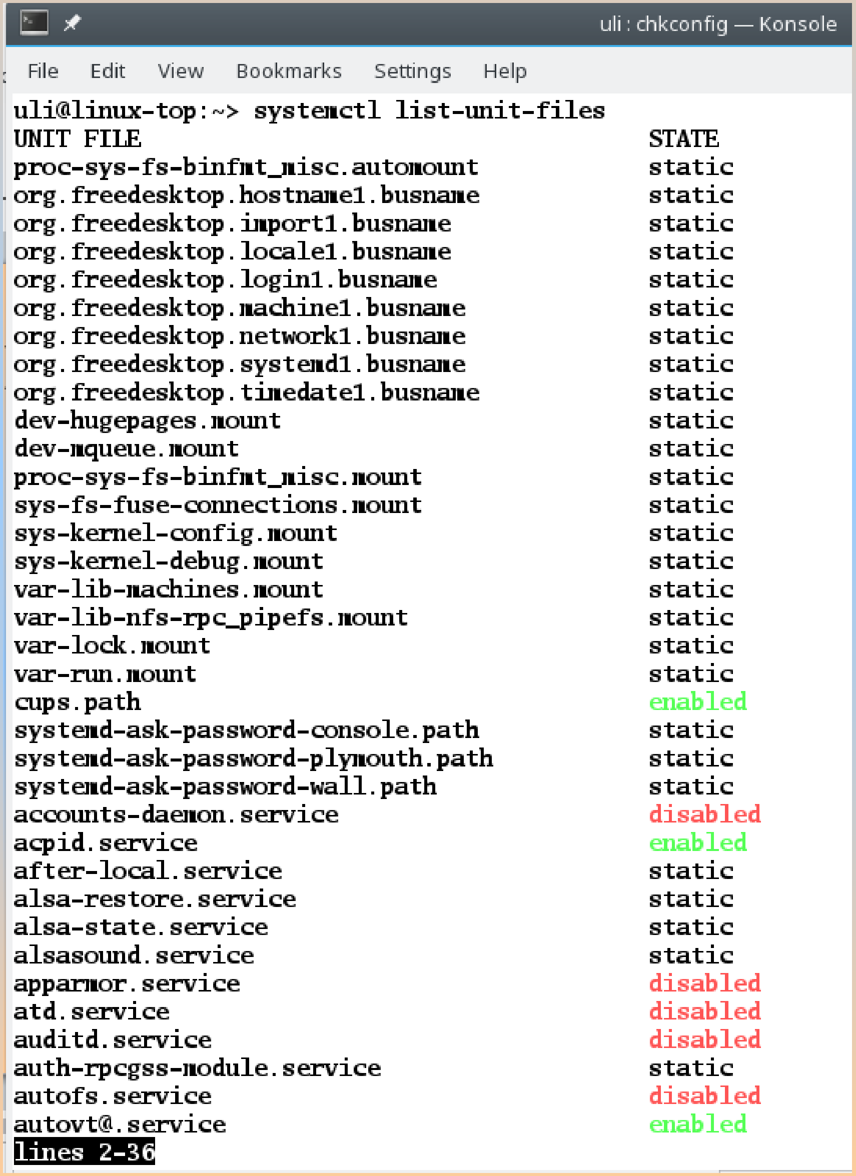
enabled means the service is running, disabled means the service is not running.



```
uli@linux-top:~> systemctl list-unit-files
UNIT FILE                                    STATE
proc-sys-fs-binfmt_misc.automount            static
org.freedesktop.hostname1.busname            static
org.freedesktop.import1.busname              static
org.freedesktop.locale1.busname              static
org.freedesktop.login1.busname               static
org.freedesktop.machine1.busname             static
org.freedesktop.network1.busname             static
org.freedesktop.systemd1.busname             static
org.freedesktop.timedate1.busname            static
dev-hugepages.mount                          static
dev-mqueue.mount                             static
proc-sys-fs-binfmt_misc.mount                static
sys-fs-fuse-connections.mount                static
sys-kernel-config.mount                      static
sys-kernel-debug.mount                       static
var-lib-machines.mount                       static
var-lib-nfs-rpc_pipefs.mount                 static
var-lock.mount                               static
var-run.mount                                static
cups.path                                    enabled
systemd-ask-password-console.path            static
systemd-ask-password-plymouth.path           static
systemd-ask-password-wall.path               static
accounts-daemon.service                      disabled
acpid.service                                enabled
after-local.service                          static
alsa-restore.service                         static
alsa-state.service                           static
alsasound.service                            static
apparmor.service                             disabled
atd.service                                  disabled
auditd.service                               disabled
auth-rpcgss-module.service                   static
autofs.service                               disabled
autovt@.service                              enabled
lines 2-36
```

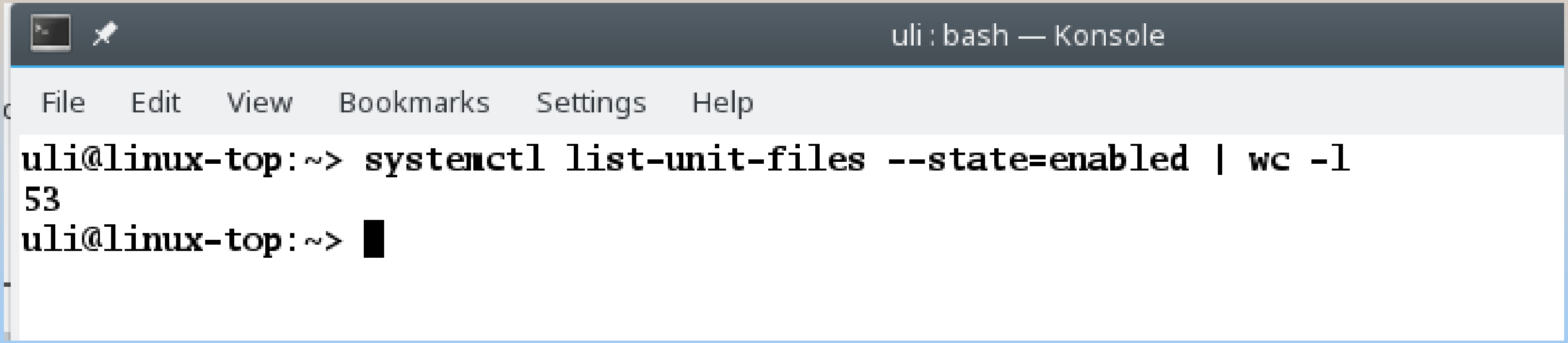
How many services are running?

static means that the unit file does not contain an "install" section, which is used to enable a unit. As such, these units cannot be enabled. Usually, this means that the unit performs a one-off action or is used only as a dependency of another unit and should not be run by itself.



```
uli@linux-top:~> systemctl list-unit-files
UNIT FILE                                    STATE
proc-sys-fs-binfmt_misc.automount            static
org.freedesktop.hostname1.busname            static
org.freedesktop.import1.busname              static
org.freedesktop.locale1.busname              static
org.freedesktop.login1.busname               static
org.freedesktop.machine1.busname             static
org.freedesktop.network1.busname             static
org.freedesktop.systemd1.busname             static
org.freedesktop.timedate1.busname            static
dev-hugepages.mount                          static
dev-mqueue.mount                             static
proc-sys-fs-binfmt_misc.mount                static
sys-fs-fuse-connections.mount                static
sys-kernel-config.mount                      static
sys-kernel-debug.mount                       static
var-lib-machines.mount                       static
var-lib-nfs-rpc_pipefs.mount                 static
var-lock.mount                               static
var-run.mount                                static
cups.path                                    enabled
systemd-ask-password-console.path            static
systemd-ask-password-plymouth.path           static
systemd-ask-password-wall.path               static
accounts-daemon.service                      disabled
acpid.service                                enabled
after-local.service                          static
alsa-restore.service                         static
alsa-state.service                           static
alsasound.service                            static
apparmor.service                             disabled
atd.service                                  disabled
auditd.service                               disabled
auth-rpcgss-module.service                   static
autofs.service                               disabled
autovt@.service                              enabled
lines 2-36
```

How many services are running?

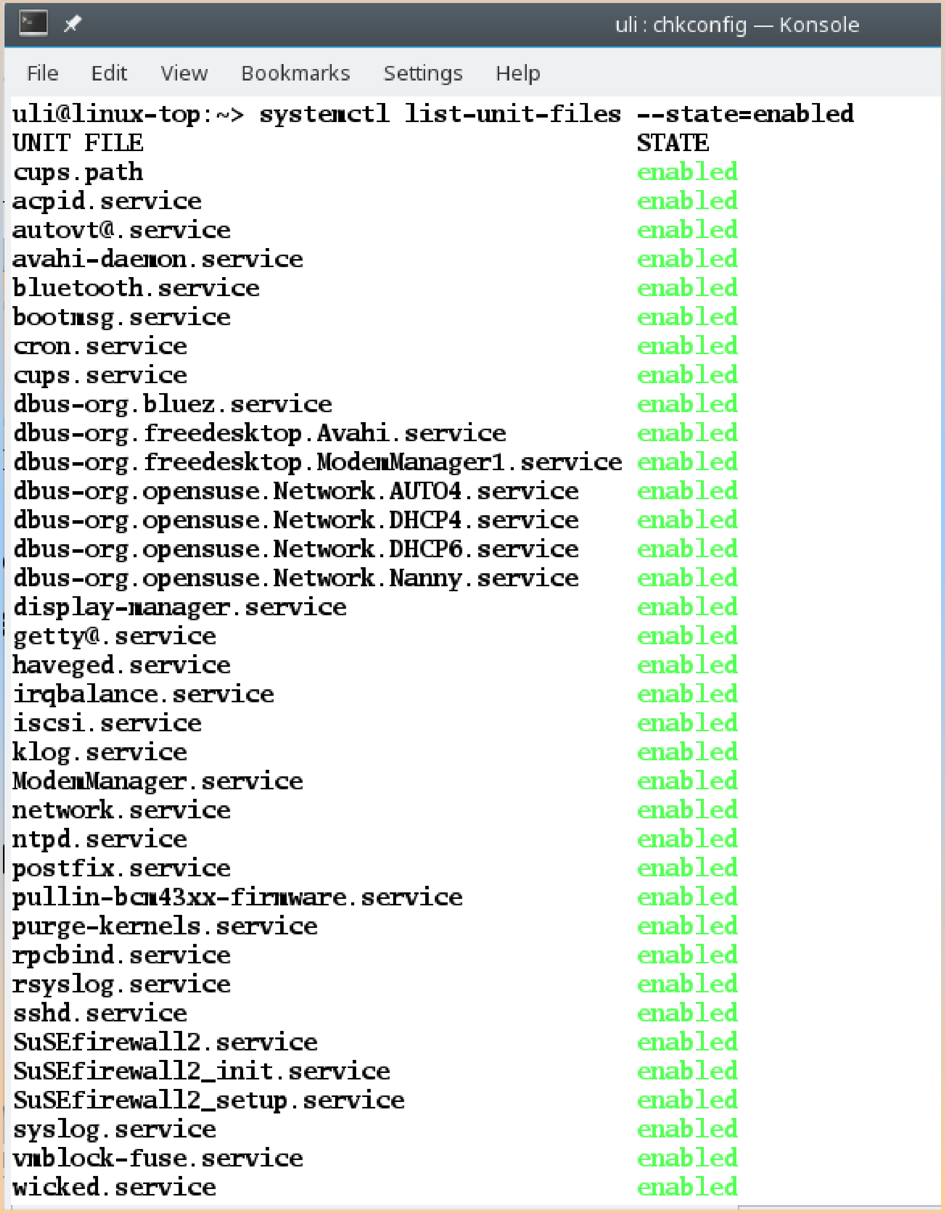


```
uli@linux-top:~> systemctl list-unit-files --state=enabled | wc -l
53
uli@linux-top:~> 
```

Here we can see that on this system we have 53 services running. The next step is to check if these services are really necessary.

Enabled Services

Here we can see the services. E.g. cups stands for ‘common unix printing’. If your computer has no printer installed all those services with cups are not required.

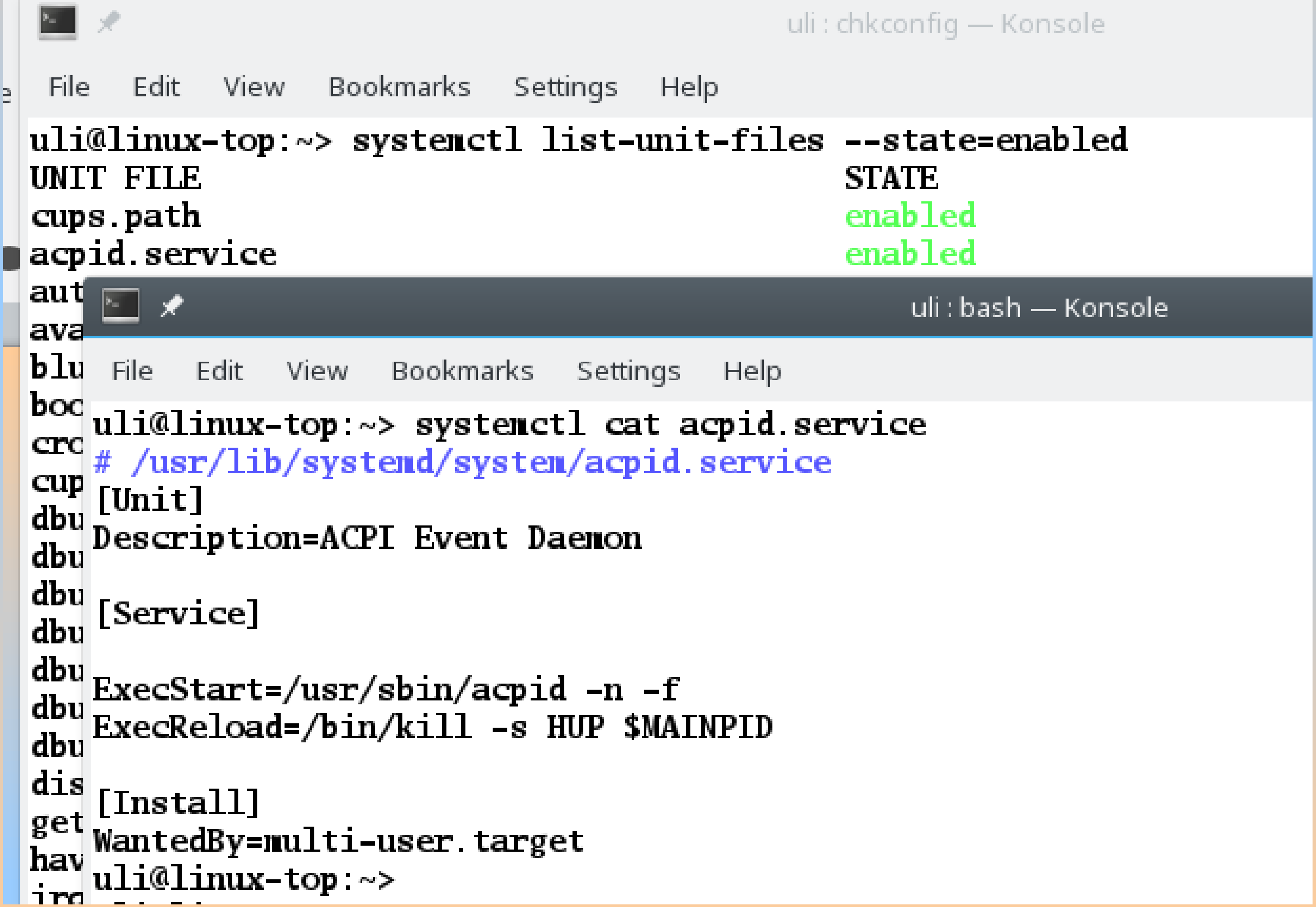


uli : chkconfig — Konsole
File Edit View Bookmarks Settings Help

```
uli@linux-top:~> systemctl list-unit-files --state=enabled
UNIT FILE                                  STATE
cups.path                                  enabled
acpid.service                              enabled
autovt@.service                            enabled
avahi-daemon.service                       enabled
bluetooth.service                          enabled
bootmsg.service                            enabled
cron.service                               enabled
cups.service                               enabled
dbus-org.bluez.service                     enabled
dbus-org.freedesktop.Avahi.service         enabled
dbus-org.freedesktop.ModemManager1.service enabled
dbus-org.opensuse.Network.AUTO4.service    enabled
dbus-org.opensuse.Network.DHCP4.service    enabled
dbus-org.opensuse.Network.DHCP6.service    enabled
dbus-org.opensuse.Network.Nanny.service    enabled
display-manager.service                    enabled
getty@.service                             enabled
haveged.service                            enabled
irqbalance.service                         enabled
iscsi.service                              enabled
klog.service                               enabled
ModemManager.service                       enabled
network.service                            enabled
ntpd.service                               enabled
postfix.service                            enabled
pullin-bcm43xx-firmware.service            enabled
purge-kernels.service                      enabled
rpcbind.service                            enabled
rsyslog.service                            enabled
sshd.service                               enabled
SuSEfirewall2.service                      enabled
SuSEfirewall2_init.service                 enabled
SuSEfirewall2_setup.service                enabled
syslog.service                             enabled
vmblock-fuse.service                       enabled
wicked.service                             enabled
```

Enabled Services

You can use **`systemctl cat <application>.service`**

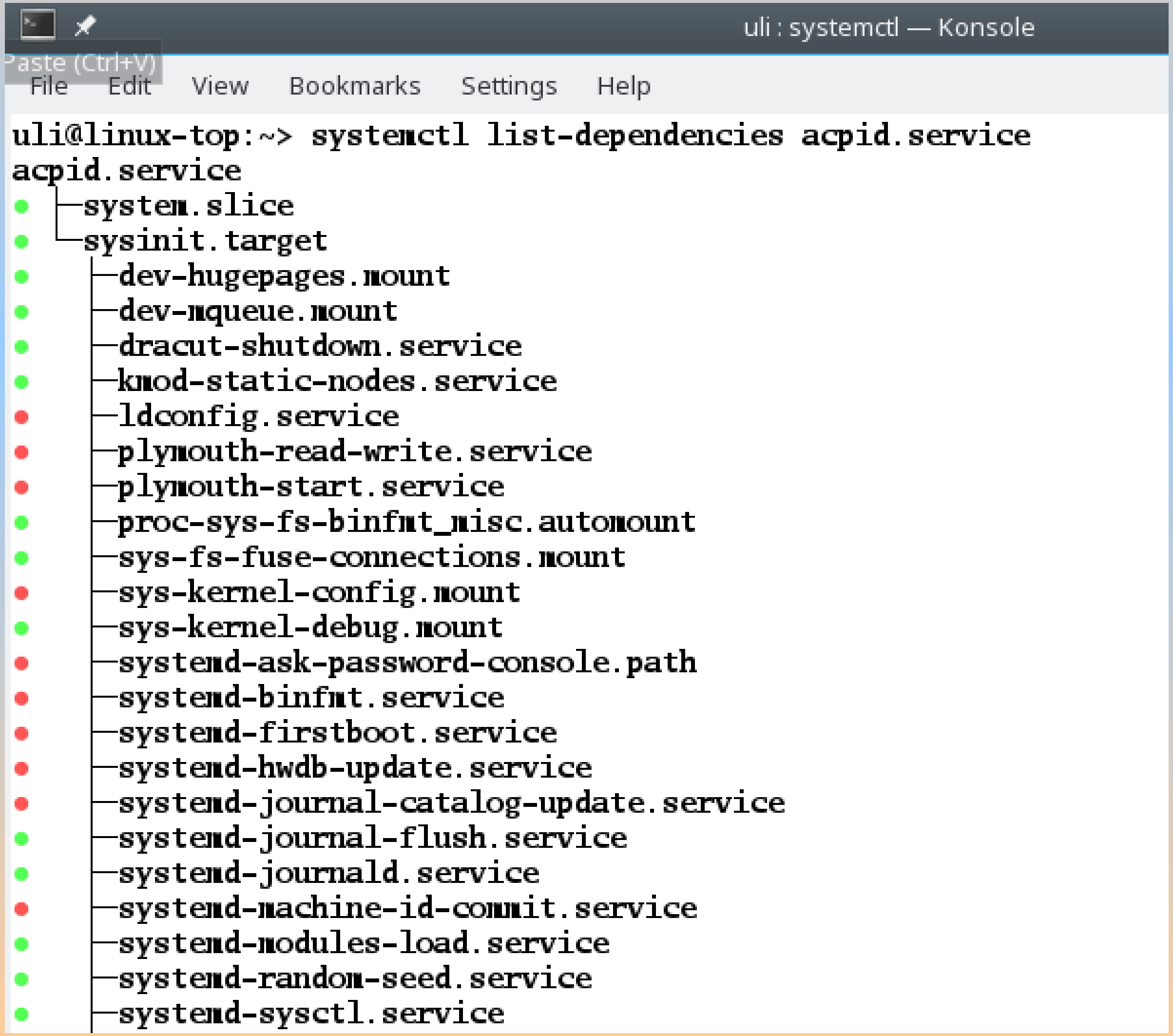


```
uli@linux-top:~> systemctl list-unit-files --state=enabled
UNIT FILE                                   STATE
cups.path                                   enabled
acpid.service                               enabled
aut
ava
blu
boo
cro
cup
dbu
dbu
dbu
dbu
dbu
dbu
dbu
dis
get
hav
```

```
uli@linux-top:~> systemctl cat acpid.service
# /usr/lib/systemd/system/acpid.service
[Unit]
Description=ACPI Event Daemon

[Service]

ExecStart=/usr/sbin/acpid -n -f
ExecReload=/bin/kill -s HUP $MAINPID

[Install]
WantedBy=multi-user.target
uli@linux-top:~>
```

Enabled Services

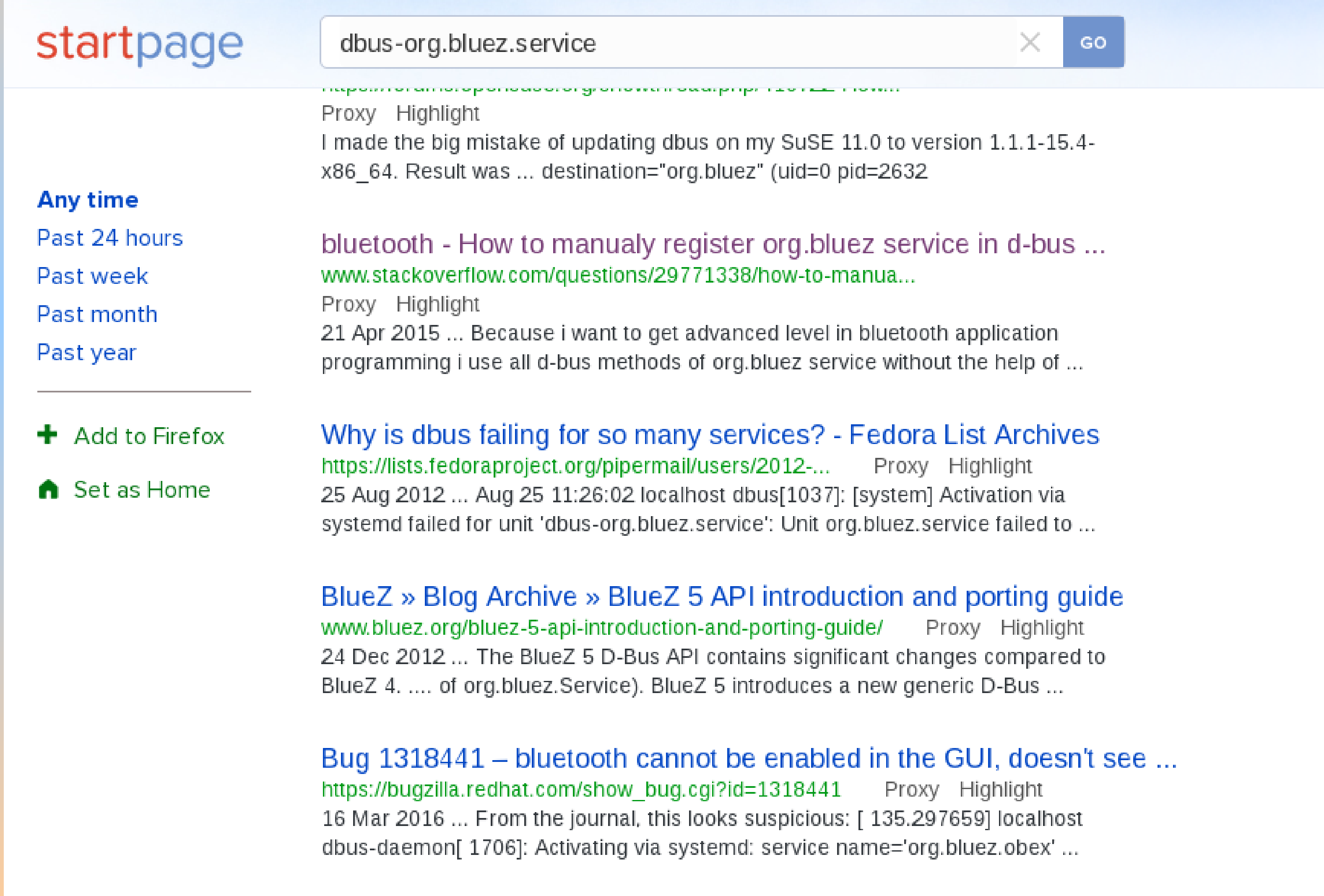
You can find services required because they are dependencies of a required service.



```
uli@linux-top:~> systemctl list-dependencies acpid.service
acpid.service
● ├─system.slice
● └─sysinit.target
●   ├─dev-hugepages.mount
●   ├─dev-mqueue.mount
●   ├─dracut-shutdown.service
●   ├─kmod-static-nodes.service
●   ├─ldconfig.service
●   ├─plymouth-read-write.service
●   ├─plymouth-start.service
●   ├─proc-sys-fs-binfmt_misc.automount
●   ├─sys-fs-fuse-connections.mount
●   ├─sys-kernel-config.mount
●   ├─sys-kernel-debug.mount
●   ├─systemd-ask-password-console.path
●   ├─systemd-binfmt.service
●   ├─systemd-firstboot.service
●   ├─systemd-hwdb-update.service
●   ├─systemd-journal-catalog-update.service
●   ├─systemd-journal-flush.service
●   ├─systemd-journald.service
●   ├─systemd-machine-id-commit.service
●   ├─systemd-modules-load.service
●   ├─systemd-random-seed.service
●   ├─systemd-sysctl.service
```

Enabled Services

You can find out about services on the Internet



dbus-org.bluez.service

Any time
Past 24 hours
Past week
Past month
Past year

https://forums.opensuse.org/showthread.php/410722-How...
Proxy Highlight
I made the big mistake of updating dbus on my SuSE 11.0 to version 1.1.1-15.4-x86_64. Result was ... destination="org.bluez" (uid=0 pid=2632

bluetooth - How to manualy register org.bluez service in d-bus ...
www.stackoverflow.com/questions/29771338/how-to-manua...
Proxy Highlight
21 Apr 2015 ... Because i want to get advanced level in bluetooth application programming i use all d-bus methods of org.bluez service without the help of ...

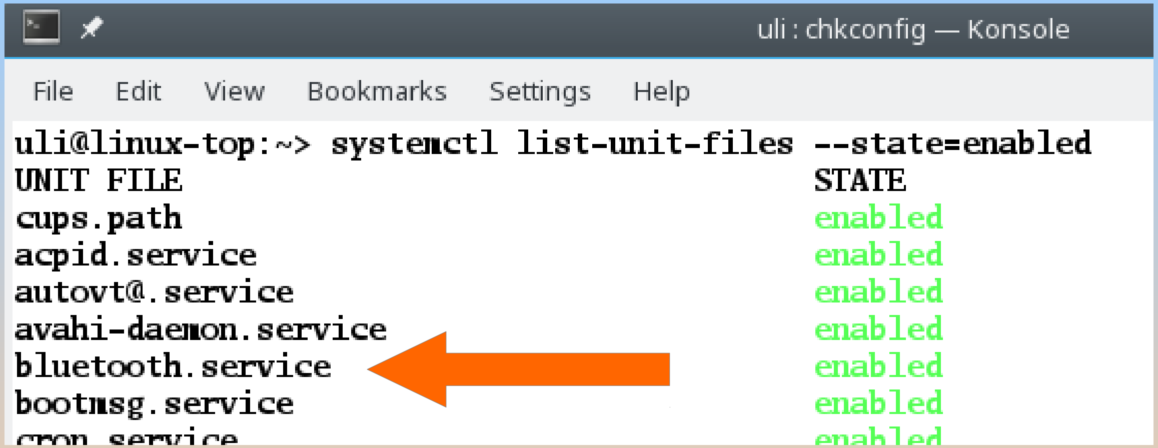
Why is dbus failing for so many services? - Fedora List Archives
https://lists.fedoraproject.org/pipermail/users/2012-... Proxy Highlight
25 Aug 2012 ... Aug 25 11:26:02 localhost dbus[1037]: [system] Activation via systemd failed for unit 'dbus-org.bluez.service': Unit org.bluez.service failed to ...

BlueZ » Blog Archive » BlueZ 5 API introduction and porting guide
www.bluez.org/bluez-5-api-introduction-and-porting-guide/ Proxy Highlight
24 Dec 2012 ... The BlueZ 5 D-Bus API contains significant changes compared to BlueZ 4. .... of org.bluez.Service). BlueZ 5 introduces a new generic D-Bus ...

Bug 1318441 – bluetooth cannot be enabled in the GUI, doesn't see ...
https://bugzilla.redhat.com/show_bug.cgi?id=1318441 Proxy Highlight
16 Mar 2016 ... From the journal, this looks suspicious: [135.297659] localhost dbus-daemon[1706]: Activating via systemd: service name='org.bluez.obex' ...

Enabled Services

If a service is not required on your computer, e.g. if your computer hasn’t got bluetooth, disable the service.

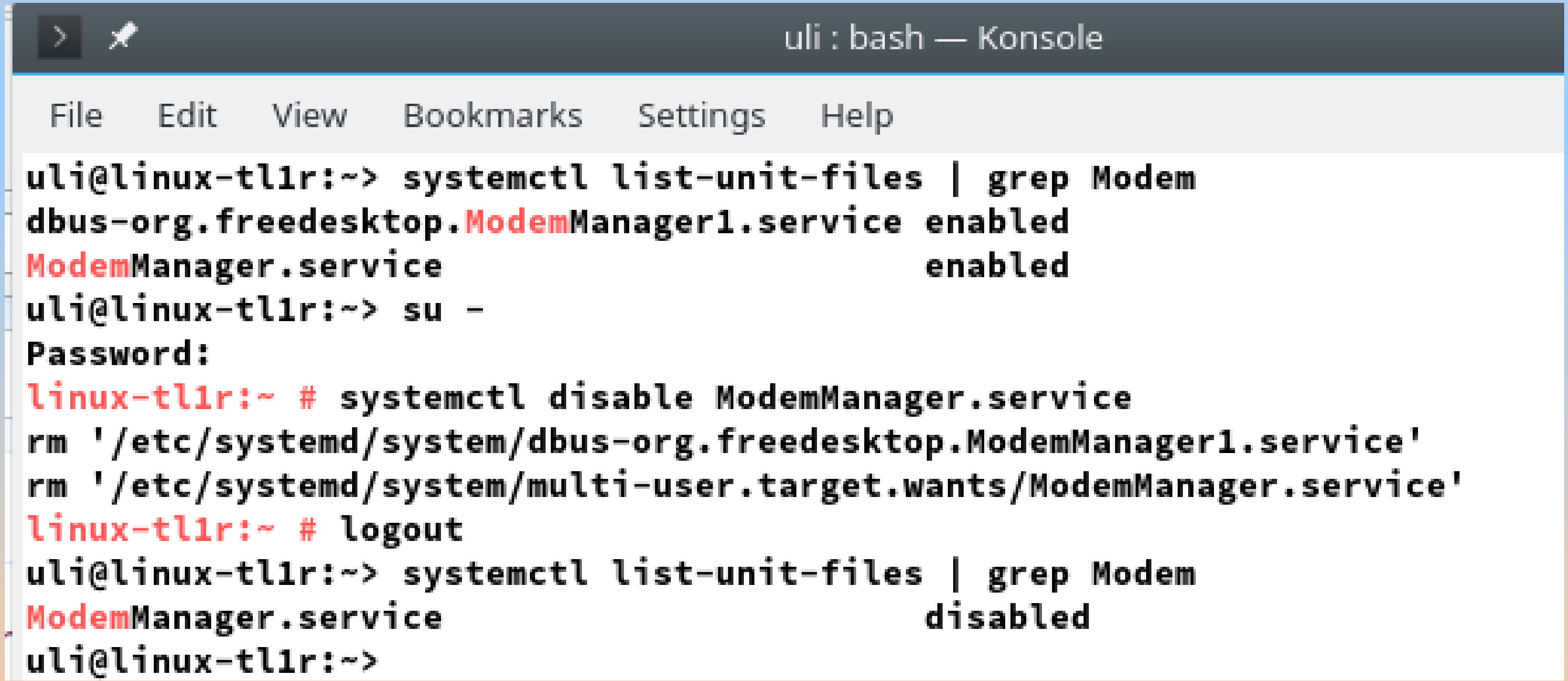


```
uli@linux-top:~> systemctl list-unit-files --state=enabled
UNIT FILE                                   STATE
cups.path                                   enabled
acpid.service                               enabled
autovt@.service                             enabled
avahi-daemon.service                        enabled
bluetooth.service                           enabled
bootmsg.service                             enabled
cron.service                                enabled
```

Disable Services

A service can be disabled with the command (as root):

`systemctl disable <application>.service`



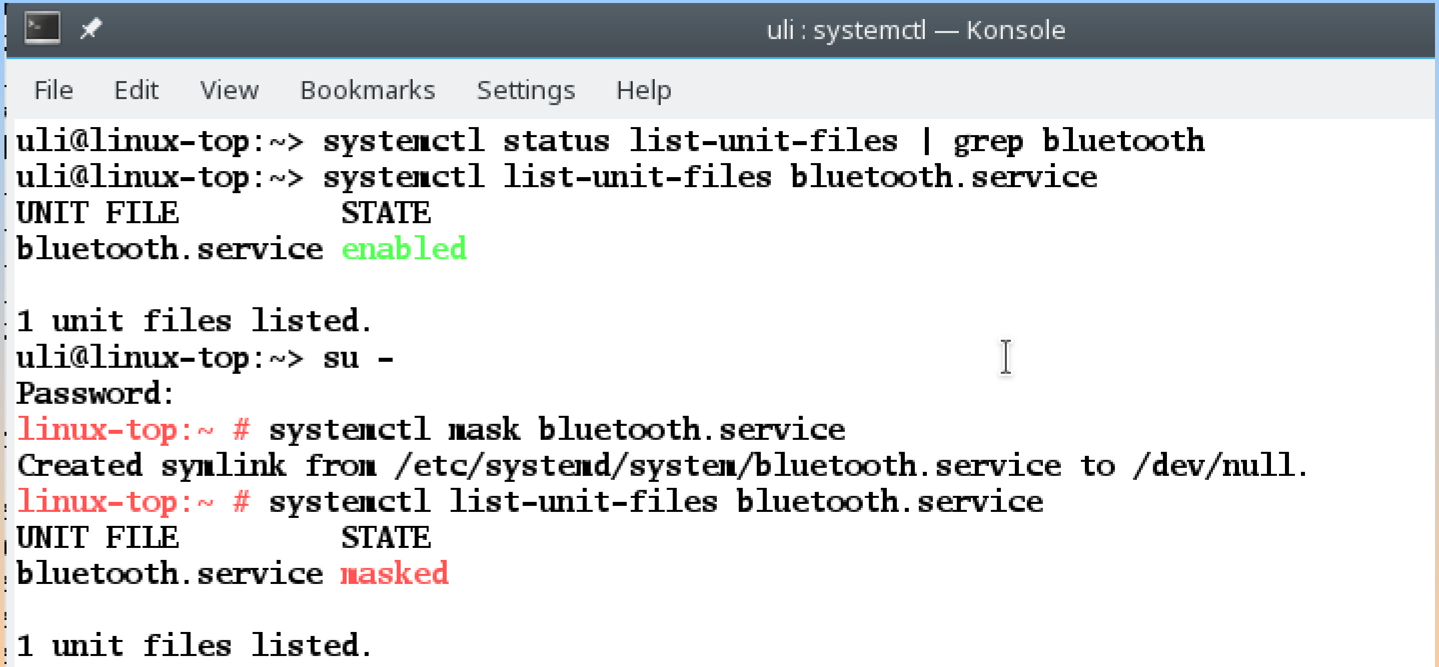
```
uli@linux-tl1r:~> systemctl list-unit-files | grep Modem
dbus-org.freedesktop.ModemManager1.service enabled
ModemManager.service                       enabled
uli@linux-tl1r:~> su -
Password:
linux-tl1r:~ # systemctl disable ModemManager.service
rm '/etc/systemd/system/dbus-org.freedesktop.ModemManager1.service'
rm '/etc/systemd/system/multi-user.target.wants/ModemManager.service'
linux-tl1r:~ # logout
uli@linux-tl1r:~> systemctl list-unit-files | grep Modem
ModemManager.service                       disabled
uli@linux-tl1r:~>
```

Disable Services

Although the disable command disables the service it might be there again after a reboot or if it is called upon as a dependency by another service. To disable a service permanently use the mask command.

Mask Services

systemd has the ability to mark a unit as completely unstartable, automatically or manually, by linking it to **`/dev/null`**. This is called masking the unit, and is possible with the mask command (as root):

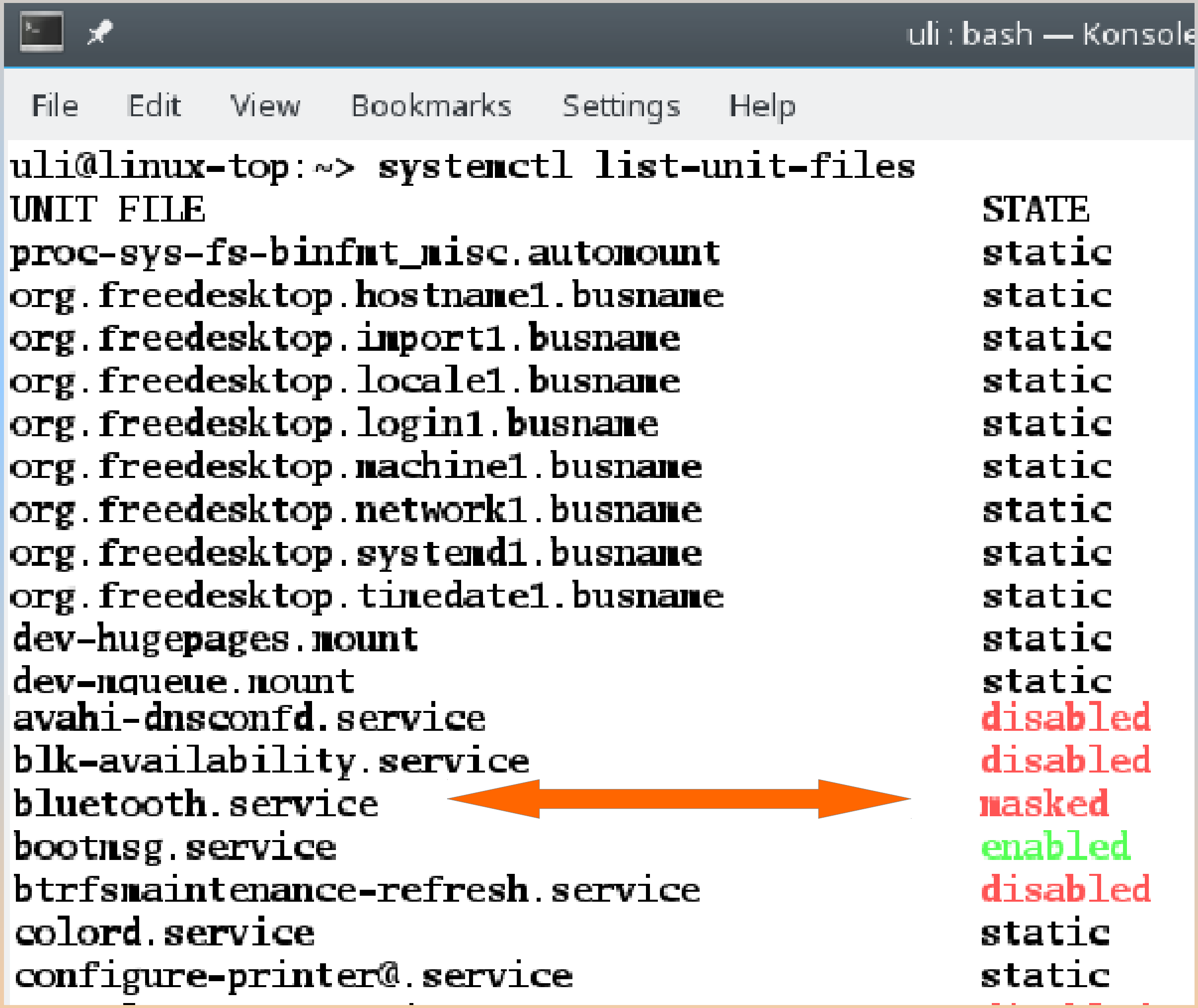


uli : systemctl — Konsole
File Edit View Bookmarks Settings Help

```
uli@linux-top:~> systemctl status list-unit-files | grep bluetooth
uli@linux-top:~> systemctl list-unit-files bluetooth.service
UNIT FILE         STATE
bluetooth.service enabled

1 unit files listed.
uli@linux-top:~> su -
Password:
linux-top:~ # systemctl mask bluetooth.service
Created symlink from /etc/systemd/system/bluetooth.service to /dev/null.
linux-top:~ # systemctl list-unit-files bluetooth.service
UNIT FILE         STATE
bluetooth.service masked

1 unit files listed.
```

Masked Services

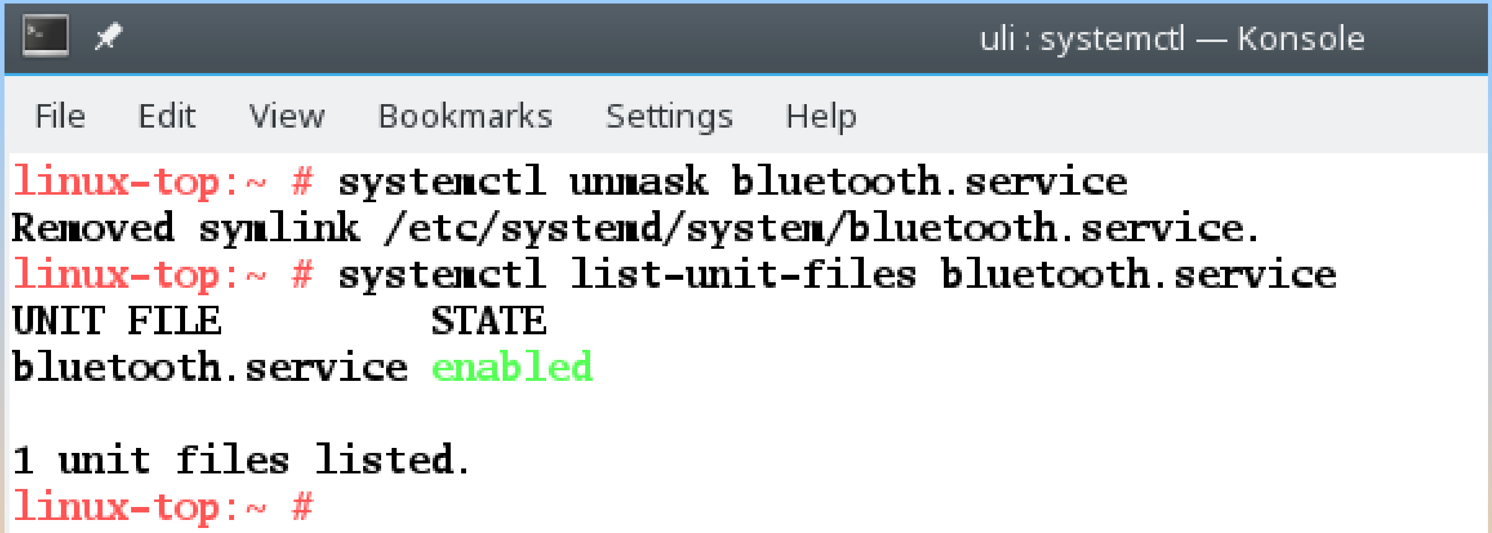
Here we can see bluetooth is masked.



```
uli@linux-top:~> systemctl list-unit-files
UNIT FILE                                    STATE
proc-sys-fs-binfmt_misc.automount            static
org.freedesktop.hostname1.busname            static
org.freedesktop.import1.busname              static
org.freedesktop.locale1.busname              static
org.freedesktop.login1.busname               static
org.freedesktop.machine1.busname             static
org.freedesktop.network1.busname             static
org.freedesktop.systemd1.busname             static
org.freedesktop.timedate1.busname            static
dev-hugepages.mount                          static
dev-mqueue.mount                             static
avahi-dnsconfd.service                       disabled
blk-availability.service                     disabled
bluetooth.service                            masked
bootmsg.service                              enabled
btrfsmaintenance-refresh.service             disabled
colord.service                               static
configure-printer@.service                   static
```

Masked Services

We can reverse this masking (again as root) using the unmask command



```
linux-top:~ # systemctl unmask bluetooth.service
Removed symlink /etc/systemd/system/bluetooth.service.
linux-top:~ # systemctl list-unit-files bluetooth.service
UNIT FILE         STATE
bluetooth.service enabled

1 unit files listed.
linux-top:~ #
```

Further useful commands:

You can get status information through:

```
systemctl status <application>.service
```

You can stop a service through:

```
 systemctl stop <application>.service
```

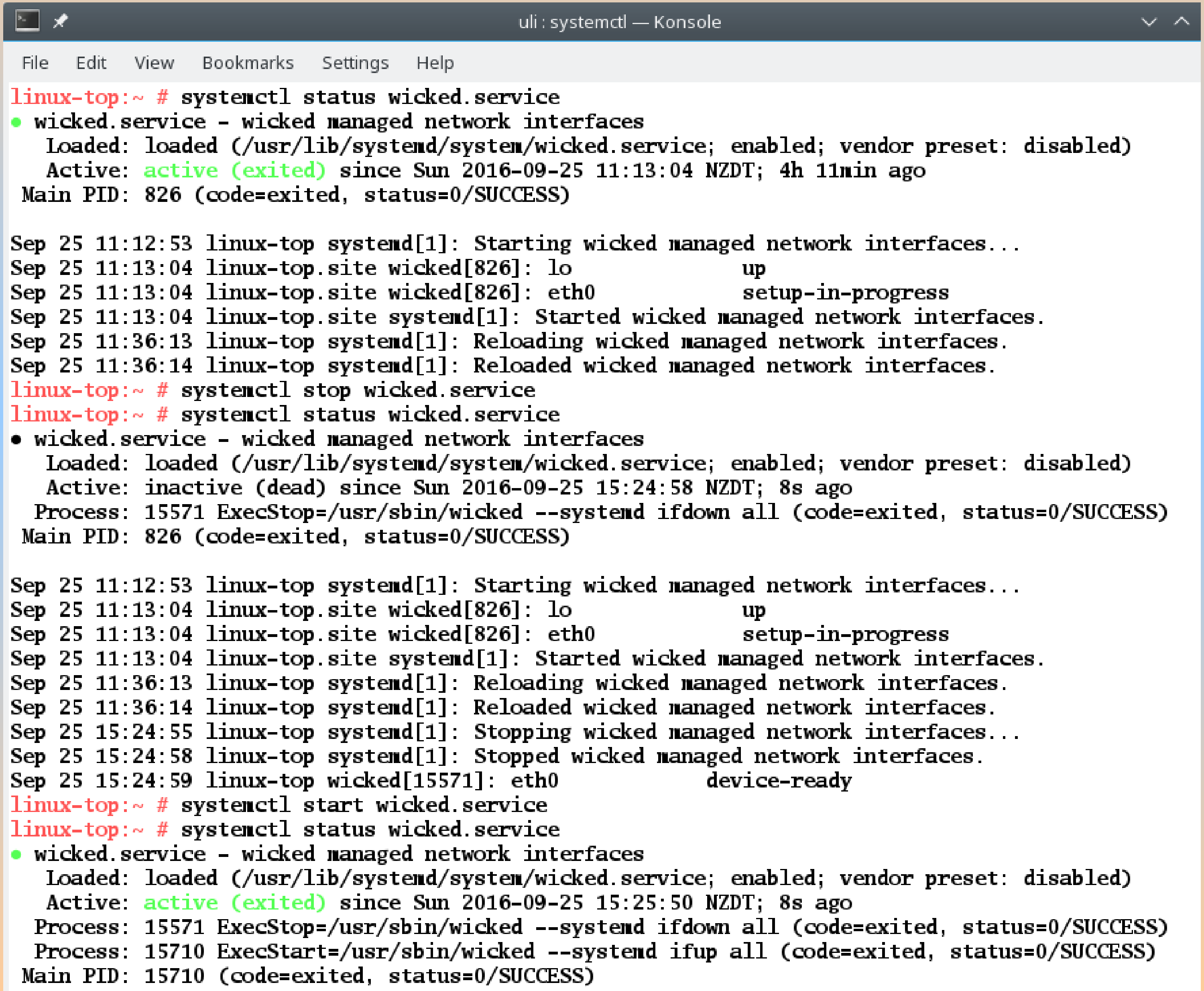
You can start a service through:

```
systemctl start <application>.service
```

You can restart (stop and then start) e.g. after updating a configuration file or making a system update:

```
systemctl restart <application>.service
```

Further useful commands:



```
linux-top:~ # systemctl status wicked.service
● wicked.service - wicked managed network interfaces
   Loaded: loaded (/usr/lib/systemd/system/wicked.service; enabled; vendor preset: disabled)
   Active: active (exited) since Sun 2016-09-25 11:13:04 NZDT; 4h 11min ago
 Main PID: 826 (code=exited, status=0/SUCCESS)

Sep 25 11:12:53 linux-top systemd[1]: Starting wicked managed network interfaces...
Sep 25 11:13:04 linux-top.site wicked[826]: lo              up
Sep 25 11:13:04 linux-top.site wicked[826]: eth0            setup-in-progress
Sep 25 11:13:04 linux-top.site systemd[1]: Started wicked managed network interfaces.
Sep 25 11:36:13 linux-top systemd[1]: Reloading wicked managed network interfaces.
Sep 25 11:36:14 linux-top systemd[1]: Reloaded wicked managed network interfaces.
linux-top:~ # systemctl stop wicked.service
linux-top:~ # systemctl status wicked.service
● wicked.service - wicked managed network interfaces
   Loaded: loaded (/usr/lib/systemd/system/wicked.service; enabled; vendor preset: disabled)
   Active: inactive (dead) since Sun 2016-09-25 15:24:58 NZDT; 8s ago
  Process: 15571 ExecStop=/usr/sbin/wicked --systemd ifdown all (code=exited, status=0/SUCCESS)
 Main PID: 826 (code=exited, status=0/SUCCESS)

Sep 25 11:12:53 linux-top systemd[1]: Starting wicked managed network interfaces...
Sep 25 11:13:04 linux-top.site wicked[826]: lo              up
Sep 25 11:13:04 linux-top.site wicked[826]: eth0            setup-in-progress
Sep 25 11:13:04 linux-top.site systemd[1]: Started wicked managed network interfaces.
Sep 25 11:36:13 linux-top systemd[1]: Reloading wicked managed network interfaces.
Sep 25 11:36:14 linux-top systemd[1]: Reloaded wicked managed network interfaces.
Sep 25 15:24:55 linux-top systemd[1]: Stopping wicked managed network interfaces...
Sep 25 15:24:58 linux-top systemd[1]: Stopped wicked managed network interfaces.
Sep 25 15:24:59 linux-top wicked[15571]: eth0            device-ready
linux-top:~ # systemctl start wicked.service
linux-top:~ # systemctl status wicked.service
● wicked.service - wicked managed network interfaces
   Loaded: loaded (/usr/lib/systemd/system/wicked.service; enabled; vendor preset: disabled)
   Active: active (exited) since Sun 2016-09-25 15:25:50 NZDT; 8s ago
  Process: 15571 ExecStop=/usr/sbin/wicked --systemd ifdown all (code=exited, status=0/SUCCESS)
  Process: 15710 ExecStart=/usr/sbin/wicked --systemd ifup all (code=exited, status=0/SUCCESS)
 Main PID: 15710 (code=exited, status=0/SUCCESS)
```