

# systemctl cheat sheet

rgdacosta

2015-02-27

systemd is awesome! If you don't like it, accept that [your cheese has been moved](#) and you now have to deal with it. Here are a few systemctl commands to help you get started off.

## Start a service

```
$ systemctl start foo
```

## Stop a service

```
$ systemctl stop foo
```

## Restart a service

```
$ systemctl restart foo
```

## Check the status of a service

```
$ systemctl status foo
```

## Enable a service to persistently start (like chkconfig on)

```
$ systemctl enable foo
```

## Disable a service to persistently stop (like chkconfig off)

```
$ systemctl disable foo
```

## Mask a service (prevent it from being (accidentally) started)

```
$ systemctl mask foo
```

## Unmask a service so that it can be started

```
$ systemctl unmask foo
```

## List all services

```
$ systemctl list-units --type=service
```

## List all targets

```
$ systemctl list-units --type=target
```

## List all units

```
$ systemctl list-units
```

## Find out where a service's definition file is

```
$ systemctl status foo | grep Loaded
```

## **Reload systemd's unit files (required after editing a unit file)**

```
$ systemctl daemon-reload
```

## **List all enabled services**

```
$ systemctl list-units-files | grep enabled
```

## **List all disabled services**

```
$ systemctl list-units-files | grep disabled
```

## **List the dependencies of a service**

```
$ systemctl list-dependencies foo.service
```

## **List the dependencies of a target**

```
$ systemctl list-dependencies foo.target
```

## **List all failed units**

```
$ systemctl --state=failed
```

## **List all failed services**

```
$ systemctl --state=failed --type=service
```

## **List targets (like runlevels)**

```
$ systemctl list-units --type=target
```

## **Change to a different target**

```
$ systemctl isolate desired.target
```

## **Set a target to be the default**

```
$ systemctl set-default desired.target
```

## **View the default target**

```
$ systemctl get-default
```

## **List which targets you're currently using**

```
$ systemctl list-units --type=target
```

## **List which units a target calls**

```
$ systemctl show --property=Wants foo.target
```

## **List what units are called when a target is isolated**

```
$ systemd --test --system --unit=foo.target
```