<https://www.cnblogs.com/lenmom/p/9191966.html>

配置参数说明如下:

**a)  Unit文件格式**：  
/etc/systemd/system：系统管理员和用户使用；  
/usr/lib/systemd/system：发行版打包者使用；  
默认单位是秒；通常由三部分组成：

[Unit]:定义与unit类型无关的通用选项；用于提供unit的描述信息，unit行为及依赖关系；

常用的选项：  
Description：描述信息  
After：定义unit的启动次序，表示当前unit应该晚于哪些unit启动，其功能与Before相反；  
Requires：依赖到的其它units，强依赖，被依赖的units无法激活时，当前unit即无法激活；  
Wants：依赖到的其它units，弱依赖；  
Conflicts：定义units间的冲突关系。

**b) [Service]:与特定类型相关的专用选项；此处为service类型；**

常用选项：  
Type：定义影响ExecStart及相关参数功能的unit进程启动类型；  
simple：默认值，这个daemon主要由ExecStart接的指令串来启动，启动后常驻于内存中；  
forking：由ExacStart启动的程序透过spawns延伸出其它子程序来作为此deamon的主要服务。原生父程序在启动结束后就会终止。  
oneshot：与simple类似，不过这个程序在完成工作后就结束，不常驻内存；  
dbus：与simple类似，但这个daemon必须要在取得一个D-Bus的名称后，才会继续运作。因此通常也要同时设定BusName=才行；  
notify：在启动完成后会发送一个通知消息。还需要配合NotifyAccess来让Systemd接收消息；  
idle：与simple类似，要执行这个daemon必须要所有的工作都顺利执行完毕后才会执行。这类的daemon通常是开机到最后才执行即可的服务。  
EnvironmentFile：环境配置文件；  
ExecStart：指明启动unit要运行命令或脚本的**绝对路径**；  
ExecStartPre：在ExecStart之前运行的**绝对路径**；  
ExecStartPost：在ExecStart之后运行的**绝对路径**；  
ExecStop：指明停止unit要运行的命令或脚本的**绝对路径**；  
Restart：当设定Restart=1时，则当次daemon服务意外终止后，会再次自动启动。

PrivateTmp:true/false表示是否给服务分配独立的临时空间

**c) [Install]：定义由”systemctl enable”以及”systemctl disable”命令在实现服务启动或禁用时用到的一些选项,可设置为多用户。**  
Alias：别名，可使用systemctl command Alias.service  
RequiredBy：被哪些units所依赖，强依赖；  
WantedBy：被哪些units所依赖，弱依赖；  
Also：安装本服务的时候还要安装别的相关服务。

以754的权限保存后，即可通过 来启动或停止服务以及配置开机启动。

systemctl start  shadowsocks      #启动

systemctl stop shadowsocks       #停止

systemctl restart shadowsocks    #重新启动

systemctl enable shadowsocks     #开机自动运行

systemctl disable shadowsocks     #禁止开机自动运行

systemctl enable tomcatd.service   #开机自动运行

systemctl start tomcatd                  #启动

systemctl stop tomcatd                  #停止

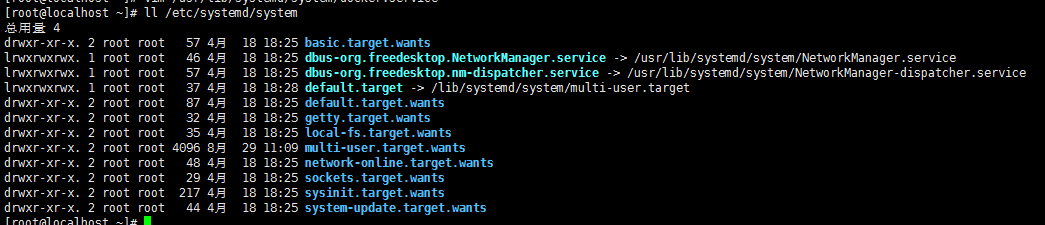
我们在linux中下载了程序，每次需要在程序的相应目录执行启动脚本。

如果想将程序做成服务，然后开机启动。以后就不在需要手动启动，也可以使用systemctl来管理，会方便很多。

开机启动需要rc.local

系统服务一般都放在/usr/lib/systemd/system下面。

开机启动好像放在/etc/system/system/下面的目录中

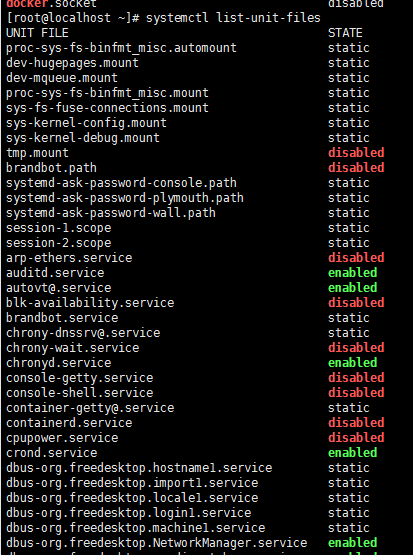


1、systemctl   
用法：systemctl [OPT] COMMAND [NAME]…  
启动服务：systemctl start NAME.service  
停止服务：systemctl stop NAME.service  
重启服务：systemctl restart NAME.service  
服务状态：systemctl status NAME.service  
条件式重启，已启动才重启，否则不作操作：systemctl try-restart NAME.service  
重载或重启服务，先加载再启动：systemctl reload-or-restart NAME.service  
重载或条件式重启：systemctl reload-or-try-restart NAME.service  
禁止自动和手动启动：systemctl mask NAME.service  
取消禁止：systemctl unmask NAME.service  
查看某服务当前激活与否的状态：systemctl is-active NAME.service  
查看所有已经激活的服务：systemctl list-units -t service  
查看所有服务：systemctl list-units -t service -a  
设定某服务开机自启动：systemctl enable NAME.service  
设定某服务开机禁止启动：systemctl disable NAME.service  
查看所有服务的开机自启动状态：systemctl list-unit-files –t service  
列出该服务在哪些运行级别下启用和禁止：ls /etc/systemd/system/\*.wants/sshd.service  
查看服务是否开机启动：systemctl is-enabled NAME.service  
查看服务的依赖关系：systemctl list-dependencies NAME.service  
杀掉进程：systemctl kill （进程名）

2、服务状态：  
loaded：unit配置文件已处理  
active（running）：一次或多次持续处理的运行  
active（exited）：成功完成一次性配置  
active（waiting）：运行中，等待一个事件  
inactive：不运行  
enable：开机启动  
disable：开机不启动  
static：开机不启动，但可以被另一个启用的服务激活。

3、运行级别：  
0 ：关机：poweroff.target  
1 ：单用户模式：rescue.target  
2 ：默认不启用NFS的多用户：multi-user.target  
3 ：完全的多用户：multi-user.target  
4 ：保留：mult-user.target  
5 ：图形：graphical.target  
6 ：重启：reboot.target

systemctl list-unit-files 命令查看当前机器上有哪些服务。



<https://www.fcwys.cc/archives/247.html> ——自定义服务

做成服务，就是写一个service文件。文件中定义unit，service，install。主要是在service部分写执行程序的语句。ExecStart

<https://www.fcwys.cc/archives/371.html> 开机启动的几种方式。总共三种

参考：<https://blog.csdn.net/husong_/article/details/80774706>

要开机启动。使用systemctl enable servicename 这样会建一个连接到mult-user.target.wants。这个目录是开机启动的目录。

查看有哪些target

#systemctl list-units --type target

系统默认启动的target

#systemctl get-default

查看target 依赖的服务

#systemctl list-dependencies

切换target 文字模式

#systemctl isolate multi-user.target

切换target gui模式

#systemctl isolate graphical.target

设置开机启动target

#systemctl set-default multi-user.target

systemctl list-units --type service