

# SystemD

The two files below are the merger of other Minecraft SystemD service files I came across on the web and the screen-based method I originally used for running Minecraft instances.

Here's a breakdown of the processes that facilitate automatically loading the Minecraft server instance(s) when the system boots up:

1. The systemd minecraft@ instance specific **service** launches **screen**. (ie: systemctl start minecraft@creative1)
2. **screen** launches the script **start-server-in-screen-systemd** (as instructed by the SystemD service).
3. **start-server-in-screen-systemd** launches "**java fabric-server-launch.jar**" (with a lot of other arguments for optimization)
4. **java** loads fabric-server-launch.jar
5. fabric-server-launch.jar takes care of loading your Minecraft server instance

I chose to use screen a bash script instead of launching java directly from systemd for these two reasons:

1. **screen** allows you to directly interact with the Minecraft server console without needing an rcon application, and its the way I'm used to dealing with server instances.
2. Using a bash script to launch the Minecraft server instance was easier to deal with per-instance customizations. You can facilitate the same thing directly from systemd, I just didn't want to take the time to learn how to do.

## Setting up your environment

Below is an example of the files and folders that are needed for this setup.

```
# create minecraft systemd service and socket files
# copy contents of files from contents further down the page
vi /etc/systemd/system/minecraft@.service
vi /etc/systemd/system/minecraft@.socket

# example for server-instance creative1

# required files for server-instance creative1
# copy contents of file from contents further down the page
vi /opt/minecraft/server-instances/creative1/start-server-in-screen-systemd
```

```
# systemctl commands for server-instance creative1
systemctl enable minecraft@creative1.service
systemctl start minecraft@creative1.service
systemctl status minecraft@creative1.service
```

## Prep commands

```
MCROOT=/opt/minecraft
mkdir -p $MCROOT/server-instances $MCROOT/tools

# change ownership if desired
chown -R minecraftuser:minecraftuser $MCROOT/server-instances

# git and make mcrcon
cd $MCROOT/tools
git clone https://github.com/Tiiffi/mcrcon.git
cd mcrcon
make
# only install it if you want, otherwise just run the binary from here
# sudo make install

# create server instance folders
MCINSTANCE1=$MCROOT/server-instances/creative1
mkdir -p $MCINSTANCE1
chown -R minecraftuser:minecraftuser $MCINSTANCE1
```

## /etc/systemd/system/minecraft@.service

```
[Unit]
Description=Minecraft Server - %i
Wants=network-online.target
After=network-online.target
ConditionPathExists=/opt/minecraft/server-instances/%i

[Service]
WorkingDirectory=/opt/minecraft/server-instances/%i

User=minecraftuser
Group=minecraftuser
```

Sockets=minecraft@%i.socket

StandardInput=socket

StandardOutput=journal

StandardError=journal

PrivateUsers=true

ProtectSystem=full

ProtectHome=true

ProtectKernelTunables=true

ProtectKernelModules=true

ProtectControlGroups=true

NoNewPrivileges=true

WorkingDirectory=/opt/minecraft/server-instances/%i

Type=simple

ExecStart=screen -D -m -S minecraft-%i ./start-server-in-screen-systemd

ExecStop=/bin/sh -c "/bin/echo stop > /opt/minecraft/server-instances/%i/systemd.stdin"

Restart=on-failure

RestartSec=60s

[Install]

WantedBy=multi-user.target

## /etc/systemd/system/minecraft@.socket

[Unit]

BindsTo=minecraft@%i.service

[Socket]

ListenFIFO=/opt/minecraft/server-instances/%i/systemd.stdin

Service=minecraft@%i.service

SocketUser=minecraftuser

SocketGroup=minecraftuser

RemoveOnStop=true

SocketMode=0600

## /opt/minecraft/server-instances/%i/start-server-in-screen-systemd

The systemd service launches screen and tells it to run this file per instance, which actually runs  
SystemD Minecraft@ Service -> Screen -> start-server-in-screen-systemd -> Java -> Fabric ->  
Minecraft

```
#!/bin/bash
```

```
# Simple command to start fabric server with diffent RAM options
```

```
#java -Xms4096M -Xmx4096M -jar fabric-server-launch.jar nogui
```

```
#java -Xms6144M -Xmx6144M -jar fabric-server-launch.jar nogui
```

```
# 20211210
```

```
# Added the following line to fix a security bug announced today
```

```
# -Dlog4j2.formatMsgNoLookups=true
```

```
# Launching Minecraft with a specific java binary using Aikar's server flags
```

```
/usr/lib/jvm/java-17-oracle/bin/java -Xms8G -Xmx8G -XX:+UseG1GC -XX:+ParallelRefProcEnabled -  
XX:MaxGCPauseMillis=200 -XX:+UnlockExperimentalVMOptions -XX:+DisableExplicitGC -XX:+AlwaysPreTouch -  
XX:G1NewSizePercent=30 -XX:G1MaxNewSizePercent=40 -XX:G1HeapRegionSize=8M -  
XX:G1ReservePercent=20 -XX:G1HeapWastePercent=5 -XX:G1MixedGCCountTarget=4 -  
XX:InitiatingHeapOccupancyPercent=15 -XX:G1MixedGCLiveThresholdPercent=90 -  
XX:G1RSetUpdatingPauseTimePercent=5 -XX:SurvivorRatio=32 -XX:+PerfDisableSharedMem -  
XX:MaxTenuringThreshold=1 -Dusing.aikars.flags=https://mcflags.emc.gs -Daikars.new.flags=true -jar fabric-  
server-launch.jar nogui
```

```
#end
```

---

Revision #1

Created 21 May 2023 20:07:01 by bluecrow76

Updated 21 May 2023 21:09:23 by bluecrow76