AI Server Initial Build

Ok, I have an old server, and old

- CPU: Intel(R) Xeon(R) CPU E5-2660 0 @ 2.20GHz
- CPU Count: 2
- Cores: 8
- Memory: 64G
- Disk: 2ea 960G SSD, RAID-1
- GPU: None
- OS: Devuan Excalibur

As you can see, this is well below what most people use. For real work, I'd want at least one GPU, and I guess I'd want my disks set up as a RAID-0.

Why Devuan and not Debian? SysV instead of System-D. Reliability. Personal preference, but it means when a system (like Ollama) is set up for system-d, I have to do (a bit more) work.

Installation

First, make sure curl and cat-certificates installed, then install Ollama, test.

```
sudo apt update
sudo apt install curl ca-certificates
# next step takes a while
curl -fsSL https://ollama.com/install.sh | sh
# test it. Don't worry if it fails, but this should work
ollama --version
ollama serve
# In another terminal, try to connect
curl http://127.0.01:11434
# in the other terminal still, install llama3.2
ollama pull llama3.2:3b
ollama run llama3.2:3b
# /bye will exit
# in main terminal, ^c will exit the ollama service
```

At this point, you have a working server. YAY. For why I chose Ollama and llama3.3:3B, read the bottom

Devuan Configuration

I'm assuming the Ollama install would set up a service under System-D (from what I read), but I'm using Sys-V, so I need to build an init script and configuration file manually. Ignore this whole section if you are not using Devuan.

Create init script

Create the file ollama in /etc/init.d. So

```
# use your favorite editor instead of 'edit'
edit /etc/init.d/ollama
```

Now, copy/paste the following in there.

ollama

```
#!/bin/sh
### BEGIN INIT INFO
# Provides:          ollama
# Required-Start:    $remote_fs $network
# Required-Stop:     $remote_fs $network
# Default-Start:     2 3 4 5
# Default-Stop:      0 1 6
# Short-Description: Ollama local LLM server
### END INIT INFO

PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
DAEMON=/usr/local/bin/ollama
NAME=ollama
USER=ollama
GROUP=ollama
RUNDIR=/run/$NAME
PIDFILE=$RUNDIR/$NAME.pid

. /lib/lsb/init-functions
[ -f /etc/default/ollama ] && . /etc/default/ollama

: "${OLLAMA_HOST:=127.0.0.1:11434}"
: "${OLLAMA_MODELS:=/var/lib/ollama/models}"
: "${OLLAMA_LOG:=/var/log/ollama.log}"

ensure_dirs() {
    install -d -o "$USER" -g "$GROUP" -m 0755 "$RUNDIR"
    install -d -o "$USER" -g "$GROUP" -m 0755 /var/lib/ollama
    install -d -o "$USER" -g "$GROUP" -m 0755 "$OLLAMA_MODELS"
    install -d -m 0755 /var/log
    touch "$OLLAMA_LOG"
    chown "$USER:$GROUP" "$OLLAMA_LOG"
}

start() {
    log_daemon_msg "Starting $NAME"
    ensure_dirs

    start-stop-daemon --start --background --make-pidfile \
```

```
        --pidfile "$PIDFILE" \
        --chuid "$USER:$GROUP" \
        --startas /bin/sh -- -c \
        "exec env OLLAMA_HOST=\"$OLLAMA_HOST\"
OLLAMA_MODELS=\"$OLLAMA_MODELS\" \"$DAEMON\" serve >>\"$OLLAMA_LOG\"
2>&1"

    RET=$?
    if [ $RET -eq 0 ]; then
        sleep 2
        if kill -0 "$(cat "$PIDFILE")" 2>/dev/null; then
            log_end_msg 0
        else
            log_end_msg 1
        fi
    else
        log_end_msg $RET
    fi
}

stop() {
    log_daemon_msg "Stopping $NAME"
    start-stop-daemon --stop --pidfile "$PIDFILE" --retry 5
    RET=$?
    rm -f "$PIDFILE"
    log_end_msg $RET
}

status() {
    if [ -f "$PIDFILE" ] && kill -0 "$(cat "$PIDFILE")" 2>/dev/null;
then
        echo "$NAME is running"
        exit 0
    else
        echo "$NAME is not running"
        exit 3
    fi
}

case "$1" in
    start) start ;;
    stop) stop ;;
    restart) stop; start ;;
    status) status ;;
    *)
        echo "Usage: $0 {start|stop|restart|status}"
        exit 1
        ;;
esac
```

```
exit 0
```

Add a user to your system, then set the init levels. Do **not** attempt to start the service yet.

```
sudo adduser --system --group --home /var/lib/ollama ollama
sudo chmod 755 /etc/init.d/ollama
sudo update-rc.d ollama defaults
```

Create a configuration file

Finally, on Devuan, some base configurations are done in /etc/default/servicename, so let's create the file /etc/default/ollama

Note: The default is for the service to be available only on localhost. However, I am building this in a LAN, and want my other machines to be able to access it, so I chose to make it available to the entire LAN. You'll see the *OLLAMA_HOST=10.10.10.10:11434* entry in the sample config. That assumes my server is running on IP 10.10.10.10 and making itself available to anyone on the subnet. If you want it only accessible from localhost, change it to 127.0.0.1, ie *OLLAMA_HOST=127.0.0.1:11434*

```
edit /etc/default/ollama
```

ollama

```
OLLAMA_HOST=10.10.10.10:11434
OLLAMA_MODELS=/var/lib/ollama/models
OLLAMA_LOG=/var/log/ollama.log
```

Start the service

Finally, it is all set up, so start it up

```
sudo service ollama start
```

cli availability

If you set OLLAMA_HOST to anything but 127.0.0.1, you will need to send each cli command by setting an environmental variable first. So, for example, to do a simple *ollama list*, you would need.

```
# works for localhost
ollama list
# works in my setup (ie, tied to an external IP
OLLAMA_HOST=10.10.10.10:11434 ollama list
```

I'm lazy, so I set it up permenantly for the root shell

```
echo 'export OLLAMA_HOST=10.10.10.10:11434' >> /root/.bashrc
source /root/.bashrc
ollama list
```

The first line above creates an environmental variable in /root/.bashrc, so the OLLAM_HOST is set each time you log in. The second line simply runs it so you don't have to log out and log back in. The last line tests.

Test

```
ollama list # should show you the model you have installed
ollama run llama3.2:3b
```

The second command connects you to llama3.2:3b's shell. You can muck around in there for a while. When you are through, type **/bye** to exit.

Decisions, Decisions

I was, and remain, a newbie here. I have used AI for software development and systems administration, but never did anything with the actual workings of it.

My first decision was, what do I install. That seems a deeper question, so I'm going to answer it in a different article here.

From:
https://kb.unixservertech.com/ - **Unix Server Tech Knowledge Base**

Permanent link:
https://kb.unixservertech.com/software/ai/initialbuild?rev=1793068737

Last update: **2026/10/27 02:38**