

SystemD

- [SystemD and Asterisk](#)
- [systemd-resolved](#)

SystemD and Asterisk

Asterisk

Put the following in `/etc/systemd/system/asterisk.service` and then run `"systemctl daemon-reload && systemctl enable asterisk"`

```
[Unit]
Description=Asterisk PBX And Telephony Daemon
Wants=network.target
After=network.target

[Service]
Type=simple
User=root
Group=root
#Environment=HOME=/var/lib/asterisk
#WorkingDirectory=/var/lib/asterisk
ExecStart=/usr/sbin/asterisk -f -C /etc/asterisk/asterisk.conf
ExecStop=/usr/sbin/asterisk -rx 'core stop now'
ExecReload=/usr/sbin/asterisk -rx 'core reload'

LimitNOFILE=65535

# safe_asterisk emulation
Restart=always
RestartSec=10

[Install]
WantedBy=multi-user.target
```

Paste below into a terminal to setup the files:

```
cat << EOF > /etc/systemd/system/asterisk.service[Unit]
Description=Asterisk PBX And Telephony Daemon
Wants=network.target
After=network.target

[Service]
Type=simple
User=root
Group=root
#Environment=HOME=/var/lib/asterisk
#WorkingDirectory=/var/lib/asterisk
ExecStart=/usr/sbin/asterisk -f -C /etc/asterisk/asterisk.conf
ExecStop=/usr/sbin/asterisk -rx 'core stop now'
ExecReload=/usr/sbin/asterisk -rx 'core reload'

LimitNOFILE=65535

# safe_asterisk emulation
Restart=always
RestartSec=10

[Install]
WantedBy=multi-user.target

EOF
```

Loading Dahdi

After some digging, for my purposes I found the best way to load the transcoding module is to use the systemd module loader to load the wctc4xpp transcoding module and UDEV to run dahdi_cfg once the module is loaded.

Prerequisite: Configure the /etc/dahdi/modules and /etc/dahdi/system.conf files as you normally would.

```
cat << EOF > /etc/dahdi/modules
# /etc/modules-load.d/dahdi is symlinked here so systemd will load it on startup
# /etc/udev/rules.d/dahdi-wctc4xpp.rules instructs udev to run dahdi_cfg after wctc4xpp module is loaded
```

```
# Digium TC400B: G729 / G723 Transcoding Engine
wctc4xxp

EOF

ln -s /etc/dahdi/modules /etc/modules-load.d/dahdi.conf

# update udev to run dahdi_cfg after loading the transcoding module
cat << EOF > /etc/udev/rules.d/dahdi-wctc4xxp.rules
KERNEL=="wctc4xxp" RUN+="/usr/sbin/dahdi_cfg"
EOF
```

The next time you reboot, systemd will load the module, udev will run dahdi_cfg, and then systemd will load asterisk. Granted this is only really needed if you haven't migrated away from MeetMe yet...

Single shot

Haven't had time to work on this. [Here's](#) the information I'm starting from.

systemd-resolved

systemd-resolved is a systemd service that provides network name resolution to local applications via a D-Bus interface, the resolve NSS service, and a local DNS stub listener on 127.0.0.53.

Traditionally, /etc/resolv.conf has been used to handle name resolution. Below is an example of this file on a system that uses systemd-resolved:

```
# Dynamic resolv.conf(5) file for glibc resolver(3) generated by resolvconf(8)
#   DO NOT EDIT THIS FILE BY HAND -- YOUR CHANGES WILL BE OVERWRITTEN
# 127.0.0.53 is the systemd-resolved stub resolver.
# run "systemd-resolve --status" to see details about the actual nameservers.

nameserver 127.0.0.53
search localdomain.loc
```

The systemd specific configuration file is /etc/systemd/resolved.conf. An example is shown below that uses Quad9 for primary DNS resolution with fallback to Google's Public DNS. The Ubuntu man page for [resolve.conf](#) is located [here](#).

```
# This file is part of systemd.
#
# systemd is free software; you can redistribute it and/or modify it
# under the terms of the GNU Lesser General Public License as published by
# the Free Software Foundation; either version 2.1 of the License, or
# (at your option) any later version.
#
# Entries in this file show the compile time defaults.
# You can change settings by editing this file.
# Defaults can be restored by simply deleting this file.
#
# See resolved.conf(5) for details

[Resolve]
# Use Quad9 for normal resolution
DNS=9.9.9.9 149.112.112.112
# Use Google Public DNS in case of a failure of Quad9
FallbackDNS=8.8.8.8 8.8.4.4
```

Domains=bluecrow.net

LLMNR=no

MulticastDNS=no

#DNSSEC=no

#Cache=yes

#DNSStubListener=yes