Serial Port Console under FreeBSD

First, determine what serial port you want to use. For IPMI Serial over Lan (SOL), it is whatever is designated in the firmware (BIOS or UEFI) as the console. For Xen, it is most like the first serial port (ttyu0, aka com1 for Windoze users). You will also need the port address to use (0x3F8 for the first port, 0x2F8 for second port, do a google search for the others).

The following documentation assumes we are installing for SOL on a bare metal machine, using the second serial port (ttyu1, address 0x2F8). If you are doing this for the first serial port, change those to ttyu0 and 0x3F8. It also assumes our serial speed is set to 57600, though modern hardware will want anything from 9600 to 115200; faster is better.

edit /etc/ttys

Find the line for ttyu1 and modify as follows. This sets the port to respond at 57600, using vt100 protocol and requiring a login. **Note:** use **3wire** instead of **std** if your machine does not use hardware handshaking (just leave as the default)

```
ttyu1   "/usr/libexec/getty std.57600"  vt100   onifconsole secure
```

Add parameters to boot.config

This tells the kernel to send messages to the serial port *and* the internal console, and sets the speed to 57600

```
echo "-Dh -S115200" >> /boot.config
```

Edit /boot/loader.conf

Add/edit the following in /boot/loader.conf to set up console redirect when booting.

```
# adjust if needed for the speed
comconsole_speed="115200"
# uncomment one of the following for the port you'll use
# 0x2f8 is the second serial port, 0x3F8 is the first one
#comconsole_port="0x2F8"
comconsole_port="0x3F8"
# allow multiple consoles
boot_multicons="YES"
# one of them is serial
boot_serial="YES"
# use these two consoles
console="comconsole,vidconsole"
```

```
comconsole_port="0x2F8"
# the 0x10 means it is eligible for console. If you know
# which uart you're using, change the other to 0x00
hint.uart.0.flags="0x10"
hint.uart.1.flags="0x10"
```

Hint: change hint.uart.0.flags=“0x00” to hint.uart.0.flags=“0x10” (and set 00 for uart 1) if using the first serial port.

boot_multicons="YES"

Boot from multiple consoles (in our case, the serial and the video)

boot_serial="YES"

Guess what this one means

console="comconsole,vidconsole"

Boot from the serial port AND the video console

comconsole_port="0x2F8"

Sets up to use second serial port, at 0x2F8. use 0x3F8 for the first port, or whatever memory range you are using

hint.uart.0.flags="0x00"

Disable console on first serial port

hint.uart.1.flags="0x10"

Set second serial port (uart.1) as a potential serial console. 0x20 says “always use it as a serial console”

Reboot and view output

You can now reboot and view the output on your SOL connection. If you have issues, you can find them after rebooting with

```
dmesg | grep uart
```

Resources

- https://gist.github.com/pvalkone/9292589
- https://www.freebsd.org/doc/handbook/serialconsole-setup.html
- https://forums.freebsd.org/threads/21856/
- http://www.freebsdwiki.net/index.php/IPMI
- https://www.freebsd.org/cgi/man.cgi?query=sio&sektion=4&manpath=freebsd-release-ports
- https://www.freebsd.org/doc/en_US.ISO8859-1/books/handbook/serialconsole-setup.html

From:
https://wiki.linuxservertech.com/ - **Unix Server Tech Knowledge Base**

Permanent link:
https://wiki.linuxservertech.com/unix/freebsd/ipmi_sol

Last update: **2025/03/21 19:41**