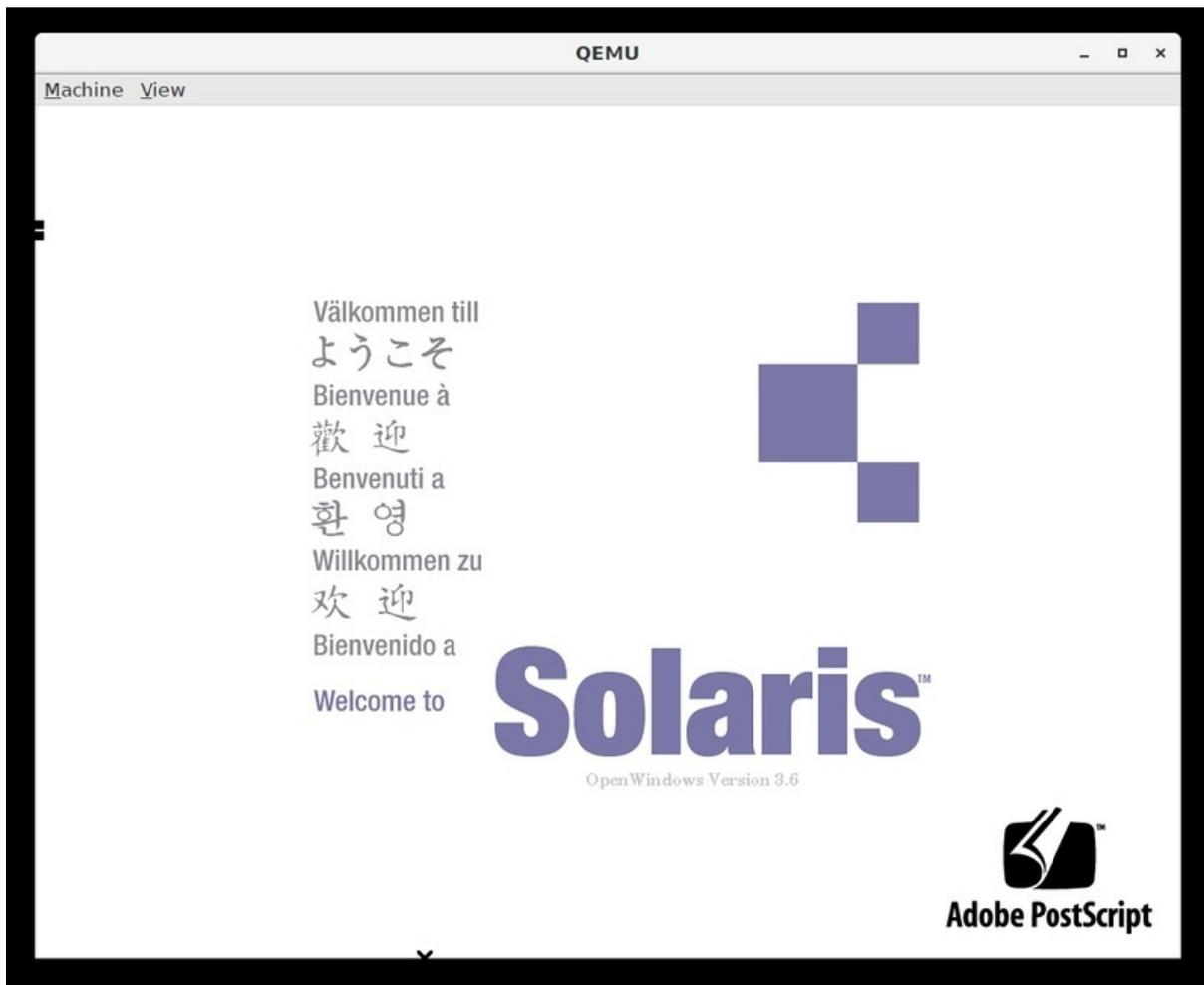




Build your own SPARC workstation with QEMU and Solaris

Created by Matt G



<https://learn.adafruit.com/build-your-own-sparc-with-qemu-and-solaris>

Last updated on 2024-06-03 02:44:17 PM EDT

Table of Contents

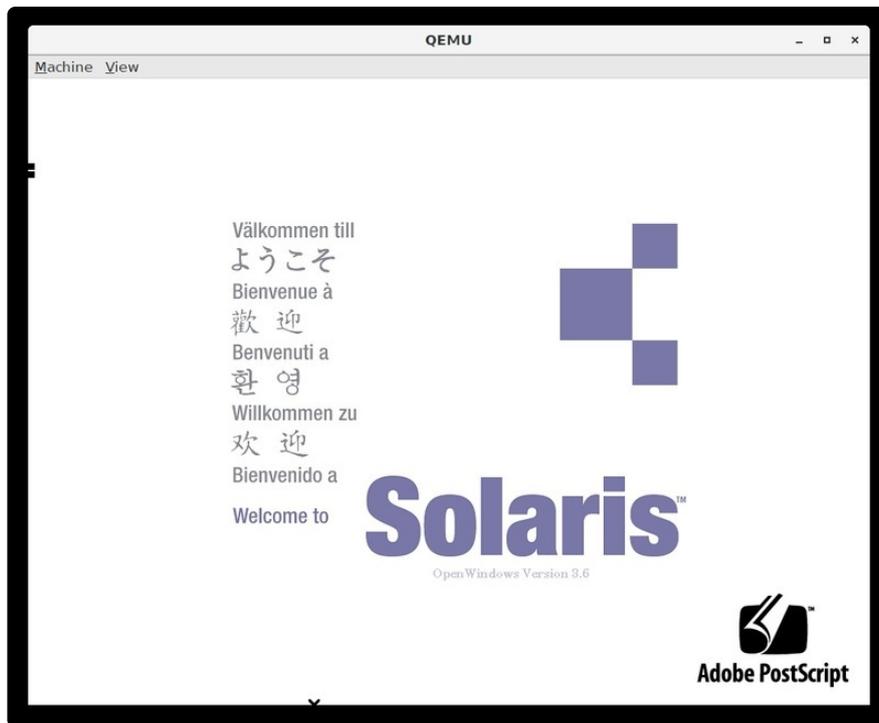
Overview	3
Set up QEMU	5
Create a disk image	5
Install Solaris	7
Install Solaris part 2	11
User Setup	13

Overview



Back in the late 80s and through the 90s, Unix workstations were super powerful, super cool, and super expensive. If you were making 3D graphics or developing applications, you wanted a high-performance workstation and Sun made some of the best ones. But unless you worked for a huge company, university, or government, they were probably too expensive.

More than twenty years later, we have much more powerful and affordable computers, so let's emulate the old systems and see what it was like to run some of the coolest computers you could buy in the 90s.



Sun workstations started out running SunOS, based on BSD Unix (like NeXTSTEP), but in 1991 they replaced it with Solaris, based on Unix System V Release 4 (like AIX and HP-UX).

The first version of Solaris was actually Solaris 2, and to try and make sense they went back and named SunOS Solaris 1, but in the operating system it still refers to itself as SunOS. So Solaris 1 is SunOS 4, and Solaris 2 is SunOS 5, and Solaris 2.6 that we'll run is SunOS 5.6. Confused yet?

Ignore the naming conventions and just know that we'll run Solaris 2.6 from 1997. For comparison, at that time a PC would be running Windows 95 and Apple released Mac OS 8 the same year.

You'll need:

- The latest version of [QEMU](#), for this guide version 3.1.0 worked well.
- A system to run it on - Windows, Linux, or Mac
- [Solaris 2.6 ISO](#)
- For added fun, a Sun Type 5 keyboard converted to USB with [Drakware's Sun2USB](#) or DIY with the [SPARC keyboard specification](#) and a small Arduino like the Trinket MO

Set up QEMU



Rather than using a virtual machine to run the x86 version of Solaris, we'll use QEMU to emulate the actual SPARC hardware used by Sun machines back in the 90s. SPARC stands for Scalable Processor Architecture and is a RISC (reduced instruction set computing) design.

For Windows, you can download the latest binary from the [QEMU site \(https://adafru.it/EEe\)](https://adafru.it/EEe). I used the 20190218 build.

For macOS you can use homebrew to install it:

```
brew install qemu
```

If you're running Linux, your distribution might have an older version of QEMU so you'll need to get the source and compile it. That goes a bit beyond this guide so take a look at the [download page \(https://adafru.it/EEf\)](https://adafru.it/EEf) for more info.

Create a disk image

The disk format that QEMU uses is called qcow2. We'll make a 9.1GB disk, that's plenty to hold Solaris and have a lot of room left over.

```
qemu-img create -f qcow2 sparc.qcow2 9663676416
```

Disks in Solaris need to be labelled before they can be used, so we can't install just yet. First we need to boot into Solaris to format and label the disk we just made.

```
qemu-system-sparc -M SS-5 -m 128 -drive file=sparc.qcow2,bus=0,unit=0,media=disk -drive file=solaris_2.6_598_sparc.iso,bus=0,unit=2,media=cdrom,readonly=on
```

Let's take a look at the options:

qemu-system-sparc - run the emulator for a 32-bit SPARC system

-M SS-5 - emulate a SPARCstation 5

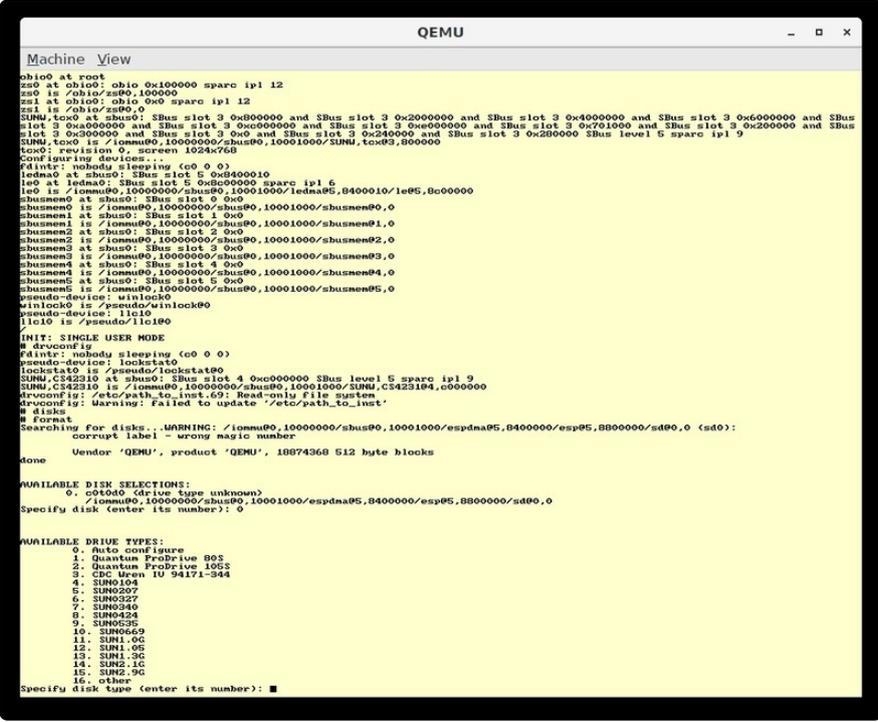
-m 128 - 128 MB of RAM

-drive file=sparc.qcow2,bus=0,unit=0,media=disk - Use our disk image, put it on SCSI bus 0, unit 0

-drive file=solaris_2.6_598_sparc.iso,bus=0,unit=2,media=cdrom,readonly=on - Use the Solaris 2.6 ISO as a CD drive, SCSI bus 0, unit 2

Once you run that, the OpenBIOS ROM should start first and bring you to a prompt, type:

```
boot cdrom:d -vs
```



```
Machine View
obio at root
z80 at cbio0: obio 0x100000 sparc ipl 12
z80 is /obio/z80,100000
z81 at cbio0: obio 0x0 sparc ipl 12
z81 is /obio/z80,0
SUNW.tcx0 at sbus0: SBUS slot 3 0x800000 and SBUS slot 3 0x2000000 and SBUS slot 3 0x4000000 and SBUS slot 3 0x6000000 and SBUS slot 3 0xa000000 and SBUS slot 3 0xc000000 and SBUS slot 3 0xe000000 and SBUS slot 3 0x0000000 and SBUS slot 3 0x201000 and SBUS slot 3 0x200000 SBUS level 5 sparc ipl 9
SUNW.tcx0 is /iommu0,10000000/sbus0,10001000/SUNW,tcx09,800000
tcx0: revision 0, screen 1024x768
Configuring devices...
fdintr: nobody sleeping (c0 0 0)
ledma0 at sbus0: SBUS slot 5 0x8400010
led0 is /ledma0,SBUS slot 5 0xc000000 sparc ipl 6
led0 is /iommu0,10000000/sbus0,10001000/ledma05,8400010/led05,8c00000
sbusmem0 at sbus0: SBUS slot 0 0x0
sbusmem0 is /iommu0,10000000/sbus0,10001000/sbusmem0,0
sbusmem1 at sbus0: SBUS slot 1 0x0
sbusmem1 is /iommu0,10000000/sbus0,10001000/sbusmem01,0
sbusmem2 at sbus0: SBUS slot 2 0x0
sbusmem2 is /iommu0,10000000/sbus0,10001000/sbusmem02,0
sbusmem3 at sbus0: SBUS slot 3 0x0
sbusmem3 is /iommu0,10000000/sbus0,10001000/sbusmem03,0
sbusmem4 at sbus0: SBUS slot 4 0x0
sbusmem4 is /iommu0,10000000/sbus0,10001000/sbusmem04,0
sbusmem5 at sbus0: SBUS slot 5 0x0
sbusmem5 is /iommu0,10000000/sbus0,10001000/sbusmem05,0
pseudo-device: winlock0
winlock0 is /pseudo/winlock0
pseudo-device: llic10
llic10 is /pseudo/llic10
INIT: SINGLE USER MODE
# drvconfig
fdintr: nobody sleeping (c0 0 0)
pseudo-device: lockstat0
lockstat0 is /pseudo/lockstat0
SUNW.C342310 at sbus0: SBUS slot 4 0xc000000 SBUS level 5 sparc ipl 9
SUNW.C342310 is /iommu0,10000000/sbus0,10001000/SUNW.C3423104,c000000
drvconfig: /etc/path_to_inst.69: Read-only file system
# disks
drvconfig: Warning: Failed to update /etc/path_to_inst
# format
Searching for disks...WARNING: /iommu0,10000000/sbus0,10001000/espdaa05,8400000/esp05,8800000/sd00,0 (sd0):
corrupt label - wrong magic number
Vendor 'QEMU', product 'QEMU', 18874368 512 byte blocks
done
AVAILABLE DISK SELECTIONS:
0. c0t0d0 drive type unknown)
/iommu0,10000000/sbus0,10001000/espdaa05,8400000/esp05,8800000/sd00,0
Specify disk (enter its number): 0
AVAILABLE DRIVE TYPES:
0. Auto configure
1. Quantum ProDrive 805
2. Quantum ProDrive 105
3. CDC Urean IU 94171-344
4. SUN0104
5. SUN0297
6. SUN0327
7. SUN0340
8. SUN0424
9. SUN0635
10. SUN0669
11. SUN1_0C
12. SUN1_0E
13. SUN1_3C
14. SUN2_1C
15. SUN2_5C
16. other
Specify disk type (enter its number): #
```

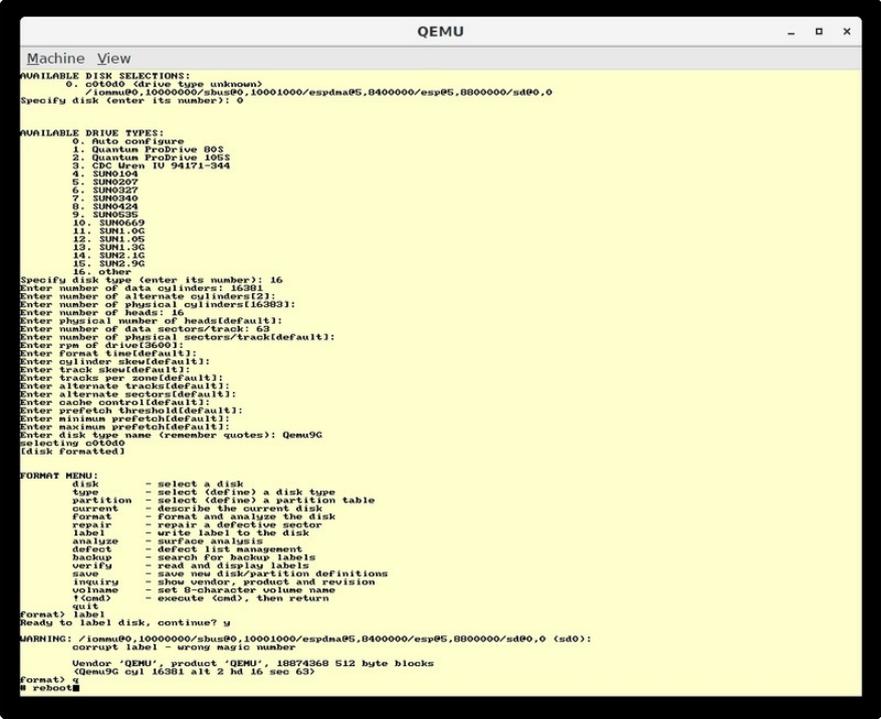
The system should boot into single user mode and bring you to a root prompt (#). First we run a few commands to prep the system, then we'll format the disk.

```
# drvconfig
# disks
# format
```

Solaris only knows about a few specific disk types, so we need to specify our own geometry.

```
Specify disk (enter its number): 0
Specify disk type (enter its number): 16
```

```
Enter number of data cylinders: 16381
...defaults are fine...
Enter number of heads: 16
...
Enter number of data sectors/track: 63
...
Enter disk type name (remember quotes): Qemu9G
```



```
QEMU
Machine View
AVAILABLE DISK SELECTIONS:
0. cot0d0 (drive type unknown)
 /iommu0,10000000/sbus0,10001000/espdaa05,04000000/esp05,8B000000/sd00,0
Specify disk (enter its number): 0

AVAILABLE DRIVE TYPES:
0. Auto configure
1. Quantum ProDrive 80S
2. Quantum ProDrive 10S
3. CDC Urem IU 94171-344
4. SUN0104
5. SUN0207
6. SUN0227
7. SUN0340
8. SUN0428
9. SUN0635
10. SUN0669
11. SUN1_0C
12. SUN1_0S
13. SUN1_3C
14. SUN2_1C
15. SUN2_5C
16. other
Specify disk type (enter its number): 16
Enter number of data cylinders: 16381
Enter number of alternate cylinders[12]:
Enter number of physical cylinders[16381]:
Enter number of heads: 16
Enter physical number of heads[default]:
Enter number of data sectors/track: 63
Enter number of physical sectors/track[default]:
Enter ppm of drive[3600]:
Enter format line[default]:
Enter cylinder skew[default]:
Enter tracks per zone[default]:
Enter alternate tracks[default]:
Enter alternate sectors[default]:
Enter cache control[default]:
Enter prefetch threshold[default]:
Enter minimum prefetch[default]:
Enter maximum prefetch[default]:
Enter disk type name (remember quotes): Qemu9G
selecting cot0d0
[disk formatted]

FORMAT MENU:
disk          - select a disk
type         - select (define) a disk type
partition    - select (define) a partition table
current      - describe the current disk
format       - format and analyze the disk
repair       - repair a defective sector
label        - write label to the disk
analyze      - surface analysis
defect       - defect list management
backup       - search for backup labels
verify       - read and display labels
save         - save new disk/partition definitions
inquiry      - show vendor, product and revision
setname      - set 8-character volume name
?cmd        - execute ?cmd, then return
quit        - quit

format) label
Ready to label disk, continue? y
WARNING: /iommu0,10000000/sbus0,10001000/espdaa05,04000000/esp05,8B000000/sd00,0 (sd0):
corrupt label - wrong magic number
Vendor: 'QEMU', product 'QEMU', 18874368 512 byte blocks
Qemu9G cyl 16381 alt 2 hd 16 sec 63
format) q
# reboot
```

Now label the disk - it's really easy:

```
format> label
Ready to label disk, continue? y
```

All set. Now type **q** to quit the format utility and type **reboot** to restart the system. It's time to install!

Install Solaris

The reboot should take you back to OpenBIOS. Boot from the CD again but this time into the installer:

```
boot cdrom:d
```

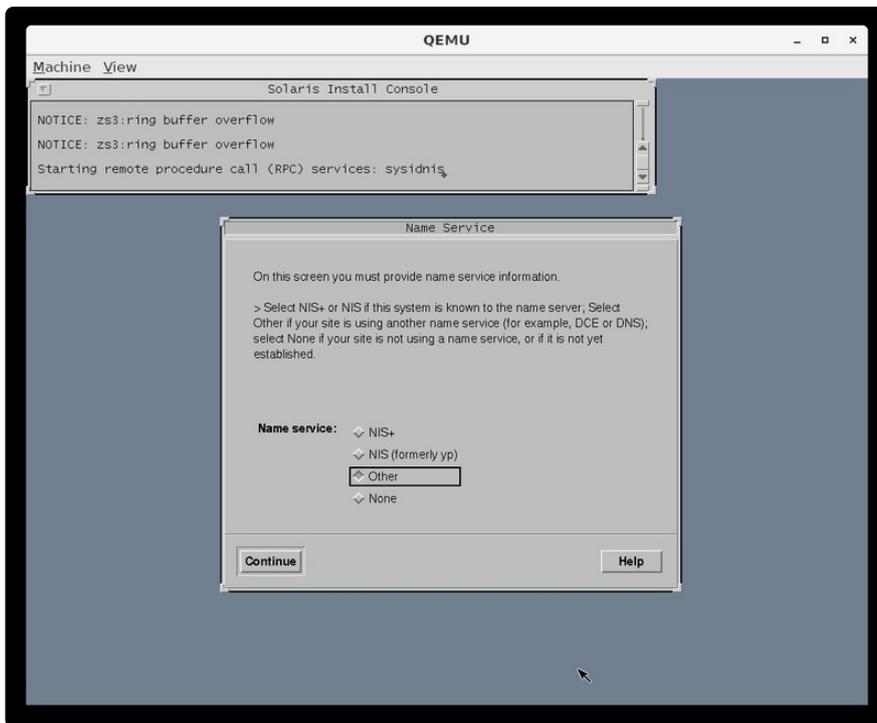


Time to install! For the most part, the defaults are fine, but there are a few places where you'll want to make changes.

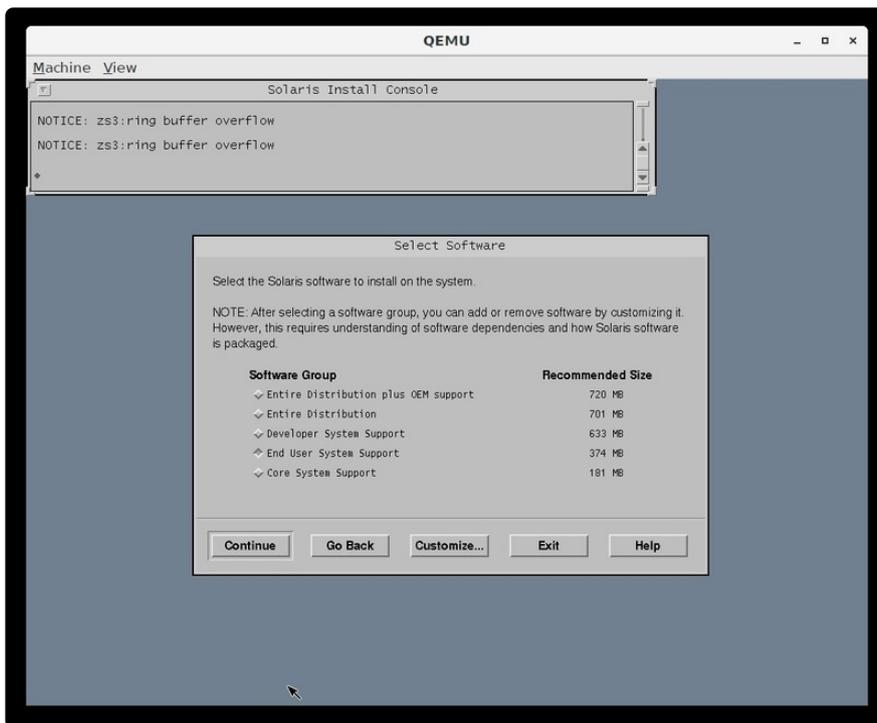
When you reach the networking options, pick a hostname and check **Yes** for Networked. Use 10.0.2.15 for the IP address, QEMU has its own internal network.

Name service: **Other**

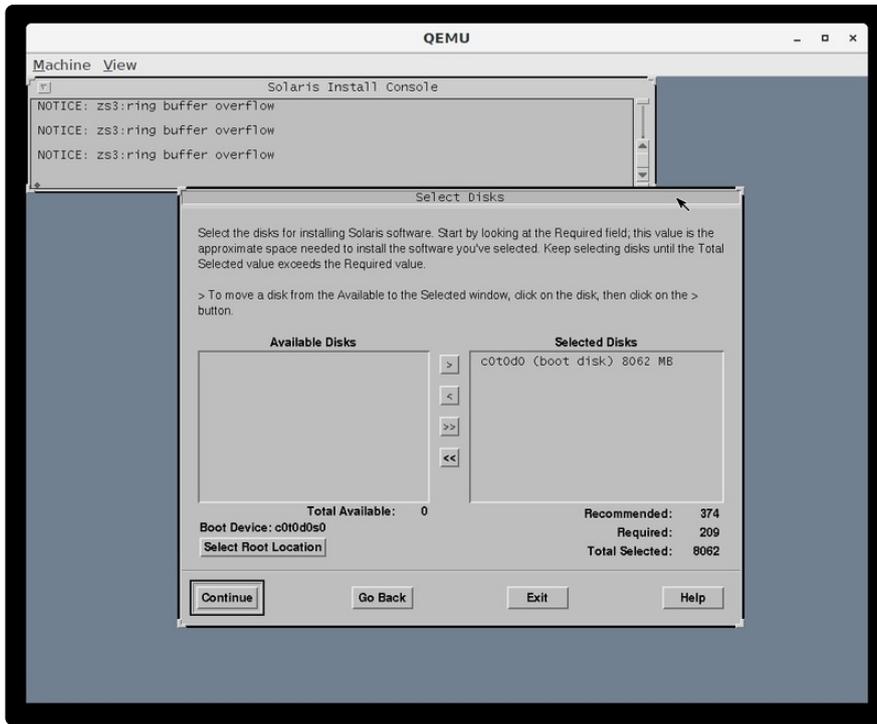
Part of a subnet: **No**



Now set the time zone, time, and date. When you get to "Allocate Client Services" just click **Continue**. Pick an additional language if you'd like one.



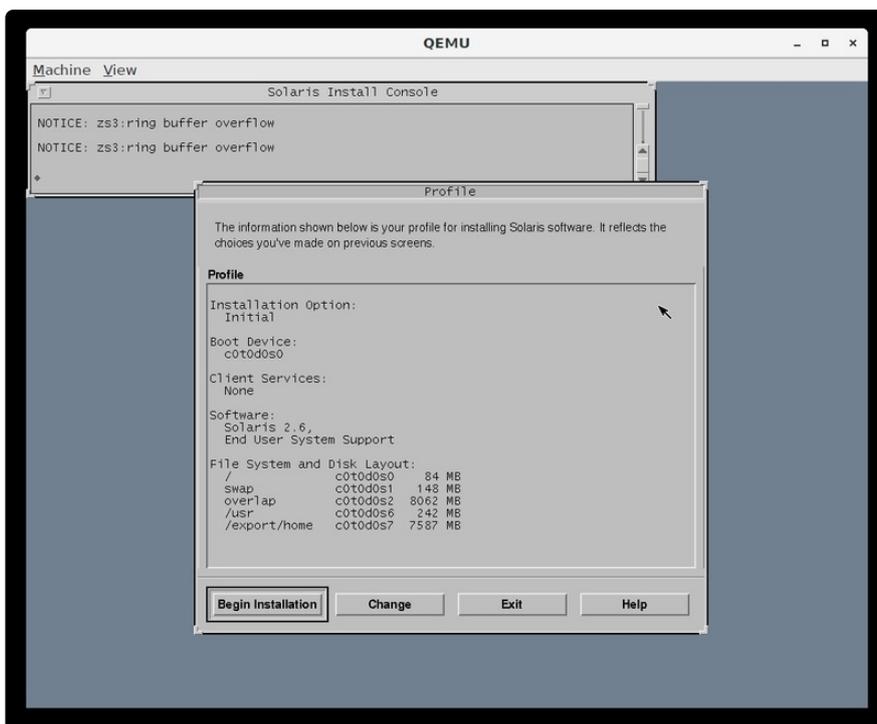
Choose how much you want to install. **End User** is enough to run it, **Developer** will let you build things, and **Entire Distribution** will just install everything. If you really want to, you can **Customize the installation** but it's easy to break things so it's best to stick with the available software groups.



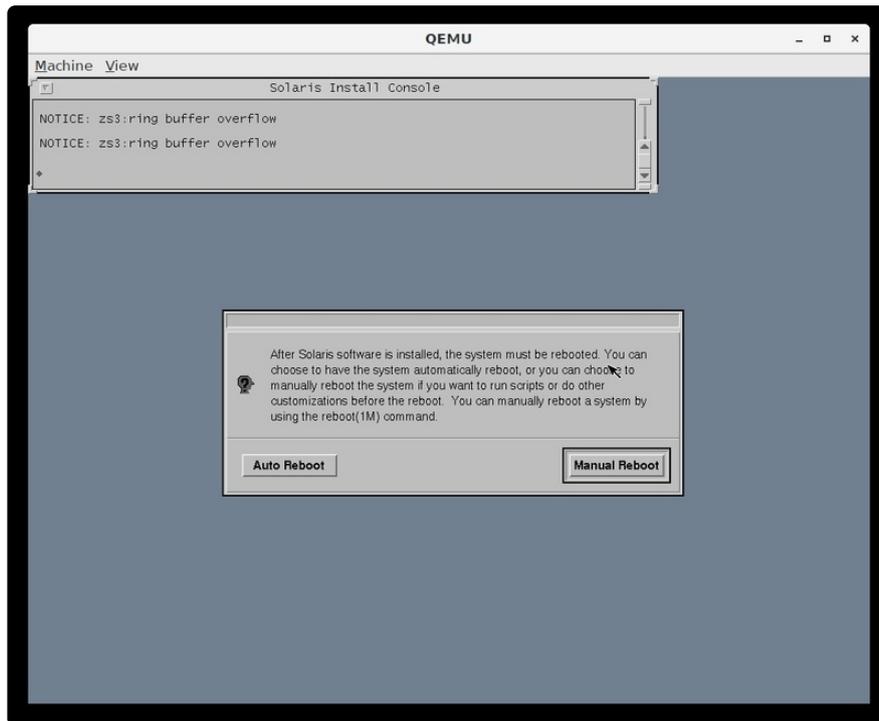
It should auto-select the available disk for you but if not you can select it and click the right arrow to add it to the selected disks.

When it asks about preserving data you can click **continue**. Click **Auto Layout** and **continue**, the defaults are fine. Click continue on the remote file systems page.

Now you can review the last few options. If it looks good click **Begin Installation**.



Make sure you choose **Manual Reboot** when prompted - we need to change some settings or else it won't boot.



Once everything is finished you should see "The finish script log `finish log` is located in `/var/sadm/system/logs` after reboot." followed by a root prompt (`#`).

Install Solaris part 2

Now that the installation has finished we need to adjust a few settings to make everything work, we'll use the console to do it. Below, `^D` means push `Ctrl+D`.

First we'll set the SCSI flags so it'll boot properly.

```
# cd /a/etc
# cat &gt;&gt; system
set scsi_options=0x58
^D
#
```

Next we need some network settings. Let's add a default router, the address that QEMU uses is **10.0.2.2**.

```
# cat &gt;&gt; defaultrouter
10.0.2.2
^D
#
```

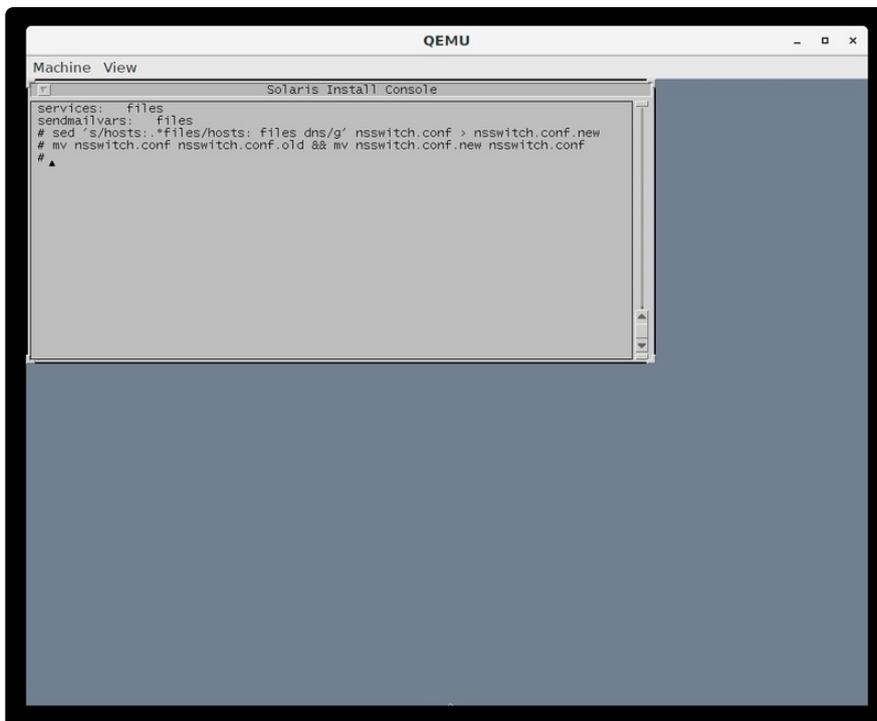
Now configure DNS. The nameserver is **10.0.2.3**.

By default, Solaris doesn't use a nameserver at all so we need to enable it. The first 3 lines make that change to **nsswitch.conf** and save it in a new file, replace the old file, then set the right permissions.

Next we tell it what nameserver to use by adding a line to the **resolv.conf** file.

```
# sed 's/hosts:.*files/hosts: files dns/g' nsswitch.conf &gt; nsswitch.conf.new
# mv nsswitch.conf nsswitch.conf.old && mv nsswitch.conf.new nsswitch.conf
# chmod 644 nsswitch.conf

# cat &gt; resolv.conf
nameserver 10.0.2.3
^D
```

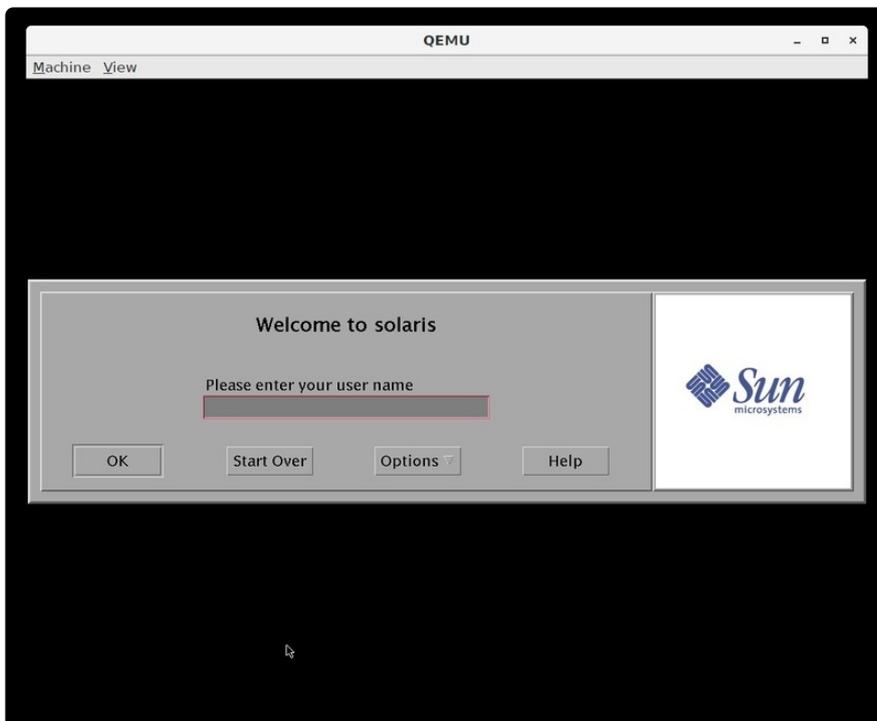


All set! Time to reboot, just type `reboot` and hit enter.

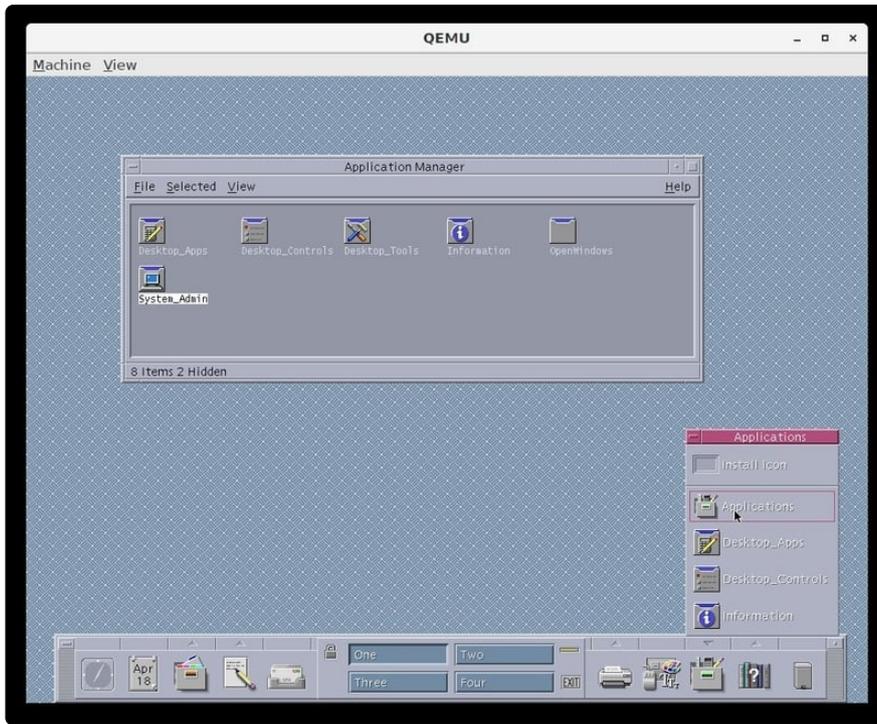
User Setup



The root account has full access to everything, so it's a very good idea to set a password. It's also a good idea to set up a normal user account. We'll do that next.



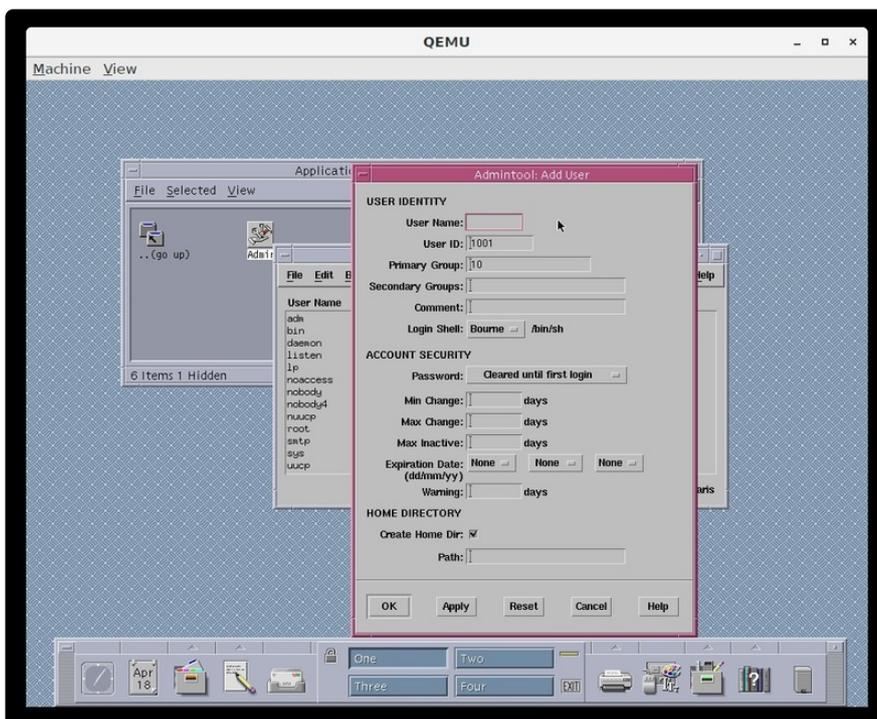
Here's the login screen, go ahead and log in as **root** with the password you just set. Once you push enter or click **OK**, it will ask which desktop environment you'd like. **CDE** is the default and a good choice, OpenWindows is the older environment.



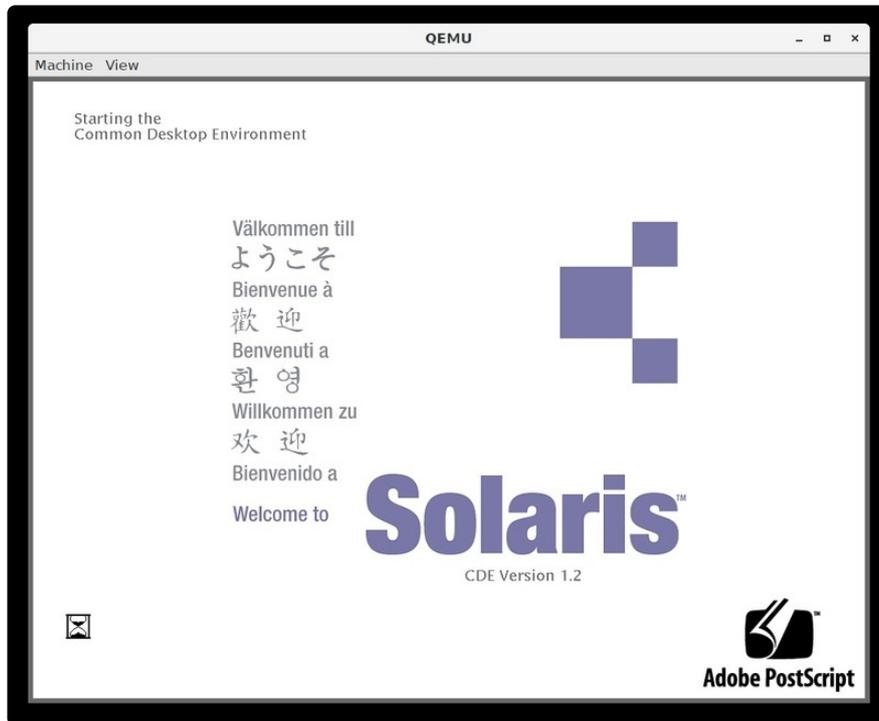
Welcome to CDE!

A bunch of windows will open, you can close them all or take a look if you like. The **Help Viewer** will give you an introduction to the CDE desktop.

The panel on the bottom has everything you need, each icon will launch something or the arrow on top will expand a tray with more things to run. We want to create a new user, so open up **Applications**, go to **System_Admin**, then run **Admintool**.



Go to **Edit, Add**, and add a user for yourself. The defaults are fine for most of it. For the home directory path at the bottom you'll want to use `/export/home/youruser`. Click **OK**, then exit Admintool. Click **exit** down at the bottom of the panel and **OK** to log out. Now you can log in as your new user and the system will prompt you to set a password. The mouse needs to be in the password window to focus it. Once set you'll be able to login.



You'll get all the windows opening again, plus one that wants you to register. You can just click **Never Register**.

When you want to shut down, click **Exit** again, then go to Options and Command Line Login. Press **enter** to get a console login prompt, login as **root**, then type **shutdown now**. After that you can quit QEMU.

Next time you start you can leave out the CD part:

```
qemu-system-sparc -M SS-5 -m 128 -drive file=sparc.qcow2,bus=0,unit=0,media=disk
```

That's it! Enjoy your Solaris system!