Embedded Linux
Exploring Raspberry Pi

Norman McEntire

Copyright (c) 2023 Servin Corp

References

- https://www.raspberrypi.com/
- https://tldp.org/LDP/sag/html/root-fs.html
- https://www.gnu.org/software/bash/manual/bash.html
- https://en.wikipedia.org/wiki/Application_binary_interface

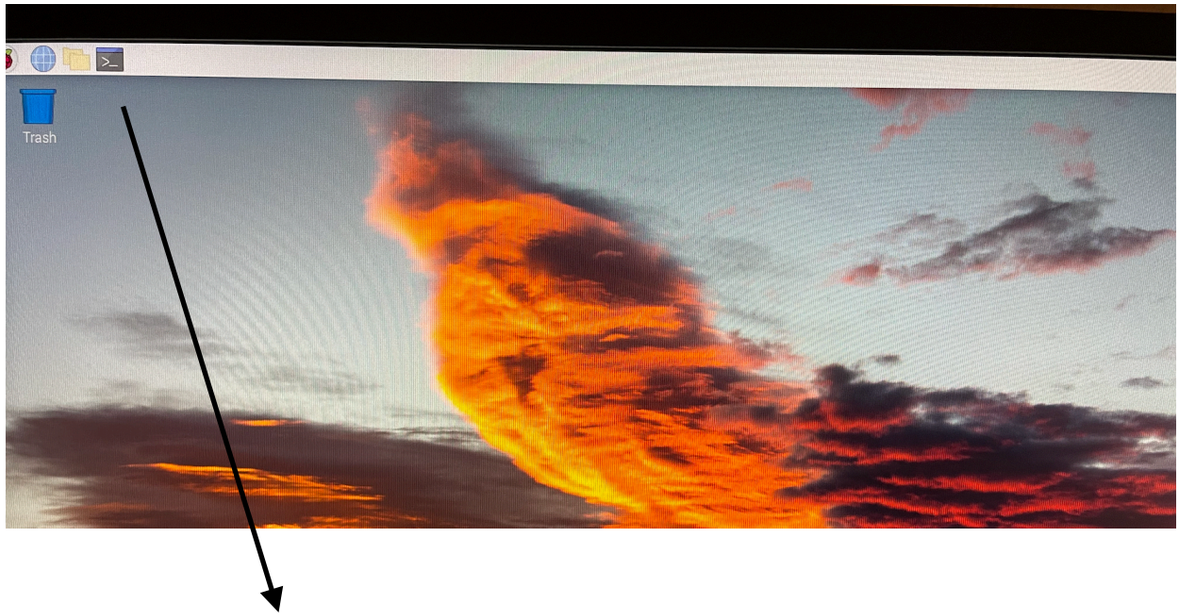
Copyright (c) 2020 Servin Corp

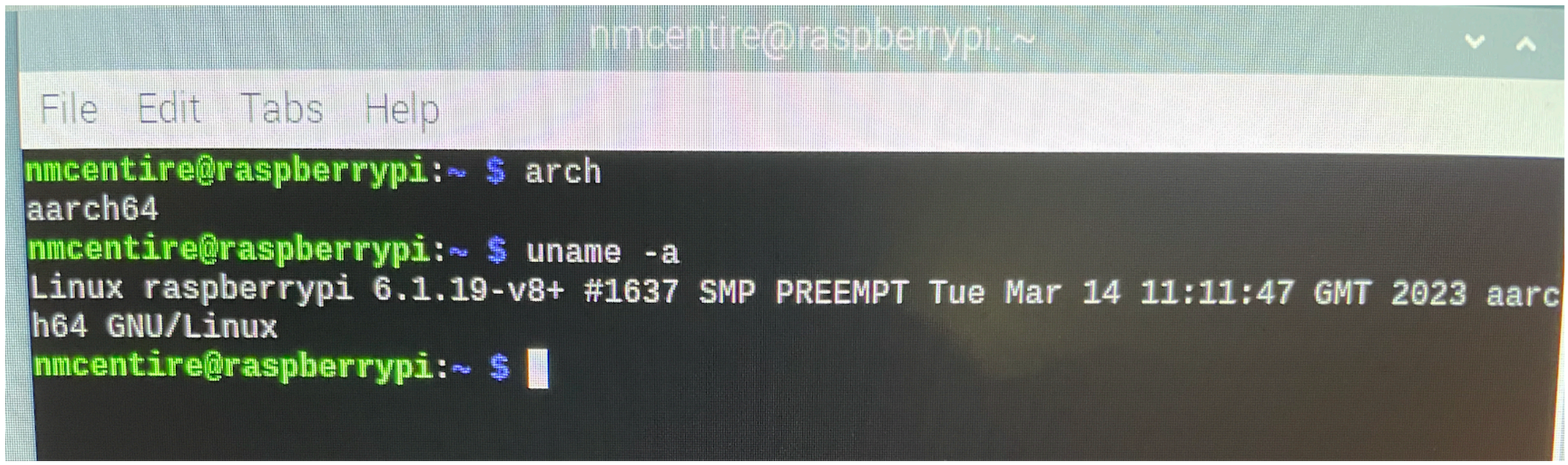
Quick Review:

ssh Into Your RPi

Copyright (c) 2023 Servin Corp

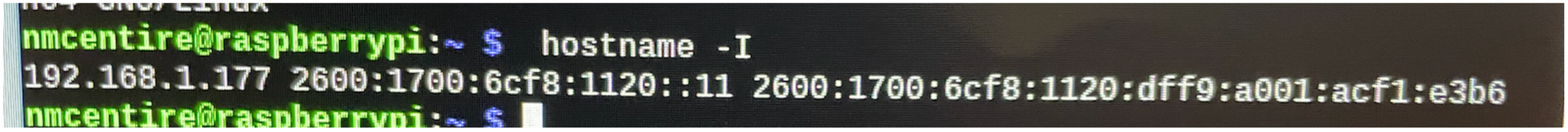
Raspberry Pi Desktop





Copyright (c) 2023 Servin Corp

Finding your RPI IP address (for remote ssh)



Copyright (c) 2023 Servin Corp

Ping your remote RPi

```
nmcentire@nmcentire-Galago-Pro:~$ ping -c3 192.168.1.177
PING 192.168.1.177 (192.168.1.177) 56(84) bytes of data.
64 bytes from 192.168.1.177: icmp_seq=1 ttl=64 time=100 ms
64 bytes from 192.168.1.177: icmp_seq=2 ttl=64 time=3.02 ms
64 bytes from 192.168.1.177: icmp_seq=3 ttl=64 time=142 ms
```


Copyright (c) 2023 Servin Corp

Try to ssh into RPI

```
nmcentire@nmcentire-Galago-Pro:~$ ssh nmcentire@192.168.1.177
ssh: connect to host 192.168.1.177 port 22: Connection refused
```

Default has sshd (Secure Shell Daemon) not installed

Copyright (c) 2023 Servin Corp

To install sshd on RPI

sudo apt install openssh-server

sudo systemctl enable ssh

sudo systemctl start ssh

Copyright (c) 2023 Servin Corp

Logging onto RPi remotely via ssh

```
nmcentire@nmcentire-Galago-Pro:~$ ssh nmcentire@192.168.1.177
The authenticity of host '192.168.1.177 (192.168.1.177)' can't be established.
ECDSA key fingerprint is SHA256:LkPSMjbzuiR5WImRorOjt6W7N5nFploB+lTOTyIpZ98.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.1.177' (ECDSA) to the list of known hosts.
nmcentire@192.168.1.177's password:
Linux raspberrypi 6.1.19-v8+ #1637 SMP PREEMPT Tue Mar 14 11:11:47 GMT 2023 aarch64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sun Mar 26 19:30:39 2023
nmcentire@raspberrypi:~ $
```


Copyright (c) 2023 Servin Corp

Embedded Linux using the RPi

Copyright (c) 2023 Servin Corp

Next up is gcc GNU C Compiler

Copyright (c) 2023 Servin Corp

Is gcc on the RPi?

```
$ gcc
gcc: fatal error: no input files
compilation terminated.
```

```
$ which gcc
/usr/bin/gcc
```

```
$ dpkg --list gcc
Desired=Unknown/Install/Remove/Purge/Hold
| Status=Not/Inst/Conf-files/Unpacked/halF-conf/Half-inst/trig-aWait/Trig-pend
|/ Err?=(none)/Reinst-required (Status,Err: uppercase=bad)
||/ Name           Version         Architecture Description
+++-==============-===============-============-===============================
ii  gcc            4:10.2.1-1+rpi1 armhf        GNU C compiler
```


Copyright (c) 2023 Servin Corp

Let’s Build Hello World C Executable

Copyright (c) 2023 Servin Corp

hello.c - Part 1

```
#include <stdio.h>

int main() {
        puts("Hello World");
        return 0;
}
```

Note: To Edit:
Option 1: vi hello.c
Option 2: nano hello.c

Copyright (c) 2023 Servin Corp

hello.c - Part 2

```
$ gcc -Wall -o hello hello.c
```

```
$ file hello
hello: ELF 32-bit LSB executable, ARM, EABI5 version 1 (SYSV),
 dynamically linked, interpreter /lib/ld-linux-armhf.so.3,
BuildID[sha1]=31cd63ca6cbb0712b1cf6cc4b740d7a9cd44f8de,
for GNU/Linux 3.2.0, not stripped
```

ELF = “Executable and Linking Format”

LSB = “Linux Standard Base” Executable”

ARM = “Advanced RISC Machines”

RISC = “Reduced Instruction Set Computing”

EABI = “Embedded Application Binary Interface”

dynamically linked (uses shared objects)

Interpreter is /lib/ld-linux-armhf.so.3

Build ID (sha1 Hash of the code)

Copyright (c) 2023 Servin Corp

hello.c - Part 3

```
$ ./hello
Hello World
```

```
$ ldd hello
  /usr/lib/arm-linux-gnueabihf/libarmmem-${PLATFORM}.so => /usr/lib/arm-linux-
gnueabihf/libarmmem-v8l.so (0xf79dc000)
  libc.so.6 => /lib/arm-linux-gnueabihf/libc.so.6 (0xf7875000)
  /lib/ld-linux-armhf.so.3 (0xf79f1000)
```

ldd = List Dynamic Dependencies

Copyright (c) 2023 Servin Corp

We created binary executable - how about a shared object?

Copyright (c) 2023 Servin Corp

First let's create an object file (.o)

```
// math.h

#ifndef _MATH_H_
#define _MATH_H_


int sum(int a, int b);
int prod(int a, int b);

#endif
```

```
// math.c

#include "math.h"

int sum(int a, int b) {
   return a + b;
}

int prod(int a, int b) {
   return a * b;
}
```

```
gcc -Wall -c math.c
```

```
$ file math.o
math.o: ELF 32-bit LSB relocatable, ARM, EABI5 version 1 (SYSV), not stripped
```


Copyright (c) 2023 Servin Corp

math-test.c

```
$ gcc -Wall -o math-test math-test.c
/usr/bin/ld: /tmp/ccKAXBHB.o: in function `main':
math-test.c:(.text+0x14): undefined reference to `sum'
/usr/bin/ld: math-test.c:(.text+0x30): undefined reference to `prod'
collect2: error: ld returned 1 exit status
```

```
$ gcc -Wall -o math-test math.o math-test.c
```

```
$ ./math-test
value: 30
value: 200
```

```
$ ldd math-test
   /usr/lib/arm-linux-gnueabihf/libarmmem-${PLATFORM}.so =>
 /usr/lib/arm-linux-gnueabihf/libarmmem-v8l.so (0xf7a1b000)
   libc.so.6 => /lib/arm-linux-gnueabihf/libc.so.6 (0xf78b4000)
   /lib/ld-linux-armhf.so.3 (0xf7a30000)
```


Copyright (c) 2023 Servin Corp

libmath.so

```
$ gcc -Wall -shared -o libmath.so math.c
```

```
$ file libmath.so
ibmath.so: ELF 32-bit LSB shared object, ARM, EABI5 version 1 (SYSV),
dynamically linked,
BuildID[sha1]=a268dcd293b017e28884667572225f02d17c02f, not stripped
```


Copyright (c) 2023 Servin Corp

math-test.c

```
// math-test.c
//
#include <stdio.h>

#include "math.h"

int main() {
   int value = sum(10,20);
   printf("value: %d\n", value);

   value = prod(10,20);
   printf("value: %d\n", value);

   return 0;
}
```


Copyright (c) 2023 Servin Corp

math-test.c and libmath.so

```
$ gcc -Wall -o math-test math-test.c -lmath
/usr/bin/ld: cannot find -lmath
collect2: error: ld returned 1 exit status
```

```
$ gcc -Wall -o math-test math-test.c -lmath -L.
```

```
$ ldd math-test
   /usr/lib/arm-linux-gnueabihf/libarmmem-${PLATFORM}.so =>
/usr/lib/arm-linux-gnueabihf/libarmmem-v8l.so (0xf7d71000)
\          libmath.so => not found
           libc.so.6 => /lib/arm-linux-gnueabihf/libc.so.6 (0xf7c0a000)
           /lib/ld-linux-armhf.so.3 (0xf7d86000)


        $ LD_LIBRARY_PATH=. ldd math-test
           /usr/lib/arm-linux-gnueabihf/libarmmem-${PLATFORM}.so =>
        /usr/lib/arm-linux-gnueabihf/libarmmem-v8l.so (0xf7f8f000)
           libmath.so => ./libmath.so (0xf7f7d000)
           libc.so.6 => /lib/arm-linux-gnueabihf/libc.so.6 (0xf7e16000)
           /lib/ld-linux-armhf.so.3 (0xf7fa4000)
```

```
$ ./math-test
./math-test: error while loading shared libraries: libmath.so: cannot open shared object file: No such
file or directory
```

```
$ LD_LIBRARY_PATH=. ./math-test
value: 30
value: 200
```


Copyright (c) 2023 Servin Corp

You can also load shared object dynamically at runtime (e.g. plugins)

Copyright (c) 2023 Servin Corp

math-test-dynamic.c

```
// math-test.c
//
#include <stdio.h>
#include <dlfcn.h>

#include "math.h"

int main() {

        void *handle = dlopen("libmath.so", RTLD_LAZY);
   if (!handle) {
      fprintf(stderr, "%s\n", dlerror());
                  return 1;
   }

   int (*sum)(int a, int b);
   sum = dlsym(handle, "sum");
   if (!sum) {
      fprintf(stderr, "%s\n", dlerror());
   }
   else {
      int value = sum(10,20);
      printf("value: %d\n", value);
   }

         dlclose(handle);

   return 0;
}
```


Copyright (c) 2023 Servin Corp

math-test-dynamic.c

```
$ gcc -Wall -o math-test-dynamic math-test-dynamic.c
/usr/bin/ld: /tmp/cct6hfqx.o: in function `main':
math-test-dynamic.c:(.text+0x14): undefined reference to `dlopen'
/usr/bin/ld: math-test-dynamic.c:(.text+0x30): undefined reference to `dlerror'
/usr/bin/ld: math-test-dynamic.c:(.text+0x58): undefined reference to `dlsym'
/usr/bin/ld: math-test-dynamic.c:(.text+0x78): undefined reference to `dlerror'
/usr/bin/ld: math-test-dynamic.c:(.text+0xb8): undefined reference to `dlclose'
collect2: error: ld returned 1 exit status
```

```
$ gcc -Wall -o math-test-dynamic math-test-dynamic.c -ldl
```

```
$  ldd math-test-dynamic
   /usr/lib/arm-linux-gnueabihf/libarmmem-${PLATFORM}.so =>
/usr/lib/arm-linux-gnueabihf/libarmmem-v8l.so (0xf7b6d000)
   libdl.so.2 => /lib/arm-linux-gnueabihf/libdl.so.2 (0xf7b46000)
   libc.so.6 => /lib/arm-linux-gnueabihf/libc.so.6 (0xf79f2000)
   /lib/ld-linux-armhf.so.3 (0xf7b82000)
```

```
$ ./math-test-dynamic
libmath.so: cannot open shared object file: No such file or directory
```

```
$ LD_LIBRARY_PATH=. ./math-test-dynamic
value: 30
```


Copyright (c) 2023 Servin Corp

Any questions about creating executables files or object files or shared object files?

Copyright (c) 2023 Servin Corp

Background Processes/ Daemons

- An Embedded Linux system has many programs running in the background
- Often called daemons
- Often start up automatically at boot time

Copyright (c) 2023 Servin Corp

dataserverd.c - Part 1

```
// dataserverd.c

#include <stdio.h>
#include <string.h>
#include <unistd.h>
#include <stdbool.h>

#define VERSION "1.0.0"

static bool g_run_as_daemon = true;

void usage(char *cmd) {
  printf("Usage: %s [--version | --no-daemon]\n", cmd);
}
```


Copyright (c) 2023 Servin Corp

dataserverd.c - Part 2

```
void handle_arguments(int argc, char *argv[]) {
  // Handle each argument
  for (int i = 1; i < argc; i++) {

    if (0 == strcmp("--version", argv[i])) {
      puts(VERSION);
    }
    else if (0 == strcmp("--no-daemon", argv[i])) {
      g_run_as_daemon = false;
    }
    else {
      usage(argv[0]);
    }
  }
}
```


Copyright (c) 2023 Servin Corp

dataserverd.c - Part 3

```
int main(int argc, char *argv[]) {

  // Handle arguments if any
  if (argc != 1) handle_arguments(argc, argv);

  if (g_run_as_daemon) {
          //int daemon(int nochdir, int noclose);
    int result = daemon(0, 1);
    if (0 != result) {
      perror("daemon");
      return 1;
    }
  }
  else {
    puts("Running in foregrond");
  }

  while (1) {
    sleep(1);
  }

  return 0;
}
```


Copyright (c) 2023 Servin Corp

Building/Running dataserverd

```
$ gcc -Wall -o dataserverd dataserverd.c
```

```
$ ./dataserverd --version
1.0.0
```

```
$ ./dataserverd --no-daemon
Running in foregrond
^C
```

```
$ ./dataserverd
$
```

```
$ ps aux | head -1
USER       PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
```

```
$ ps aux | grep dataserverd
nmcenti+  5578  0.0  0.0   1748    64 ?        Ss   22:41   0:00 ./dataserverd
```

```
$ pkill dataserverd
```


Copyright (c) 2023 Servin Corp

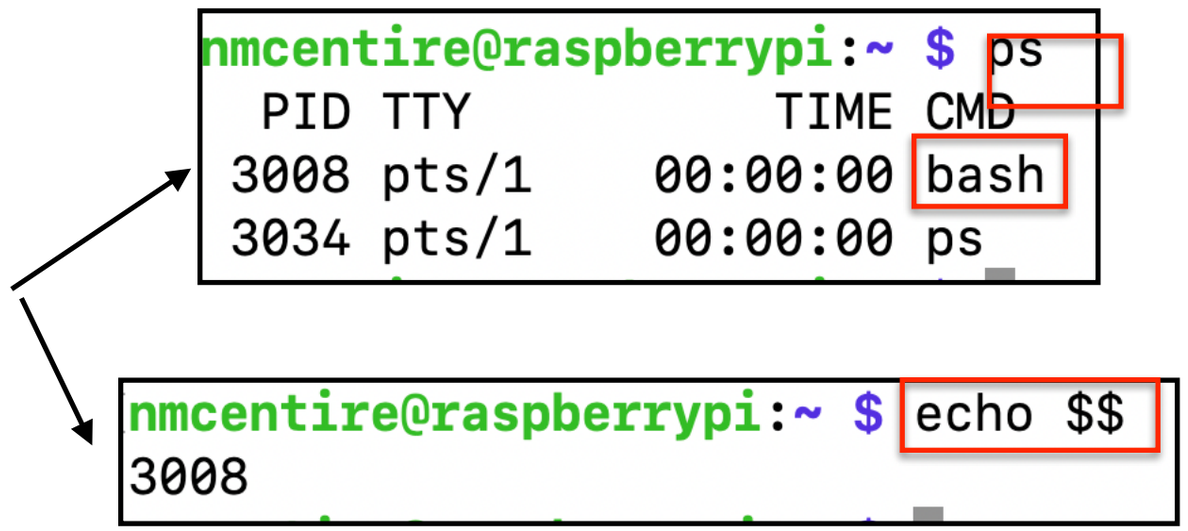
Questions about Background Processes / Daemons?

Copyright (c) 2023 Servin Corp

Use the ps (Process Status) command to explore processes

Copyright (c) 2023 Servin Corp

You are using the bash shell



ps:
process
status

PID:
process
ID

TTY:
Teletype
(old terminal)

CMD:
Command

Copyright (c) 2023 Servin Corp

ps aux

```
nmcentire@raspberrypi:~ $ ps aux
USER         PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root           1  0.0  0.4  33816  8852 ?        Ss   15:36   0:03 /sbin/init splash
root           2  0.0  0.0      0     0 ?        S    15:36   0:00 [kthreadd]
root           3  0.0  0.0      0     0 ?        I<   15:36   0:00 [rcu_gp]
root           4  0.0  0.0      0     0 ?        I<   15:36   0:00 [rcu_par_gp]
root           5  0.0  0.0      0     0 ?        I<   15:36   0:00 [slub_flushwq]
root           6  0.0  0.0      0     0 ?        I<   15:36   0:00 [netns]
avahi        343  0.0  0.1   6992  3208 ?        Ss   15:36   0:10 avahi-daemon: running [raspberrypi-2.local]
root         345  0.0  0.1   8196  2244 ?        Ss   15:36   0:00 /usr/sbin/cron -f
message+     346  0.0  0.2   8088  3932 ?        Ss   15:36   0:00 /usr/bin/dbus-daemon --system --address=systemd:
avahi        348  0.0  0.0   6752   260 ?        S    15:36   0:00 avahi-daemon: chroot helper
root         360  0.0  0.3  40972  7264 ?        Ssl  15:36   0:00 /usr/libexec/polkitd --no-debug
nmcenti+     744  0.0  0.2   8640  3848 tty1     S+   15:36   0:00 -bash
```

PID = Process ID

VSZ = Virtual Memory Size

RSS = Resident Storage Size

TTY = “?” means no TTY (daemon)

Copyright (c) 2023 Servin Corp

Embedded Linux and Bash Shell

- Many Embedded Linux Projects use bash shell as “glue” to connects different parts of system together
- Bash shell is included on almost all Embedded Linux systems
  - /bin/bash
- Many Embedded Linux Projects use a combination of bash shell plus C executables

Copyright (c) 2023 Servin Corp

Info on the bash shell

```
nmcentire@raspberrypi:~ $ which bash
/usr/bin/bash
```

```
nmcentire@raspberrypi:~ $ file /usr/bin/bash
/usr/bin/bash: ELF 32-bit LSB executable, ARM, EABI5 version 1 (SYSV), dynamically linked, interpreter /lib/ld-linux-armhf.so.3, Bu
ildID[sha1]=f12e6d40fb262ad0037b6ec43162208b76d4da71, for GNU/Linux 3.2.0, stripped
```

ELF = “Executable and Linking Format”

LSB = “Linux Standard Base” Executable”

ARM = “Advanced RISC Machines”

RISC = “Reduced Instruction Set Computing”

EABI = “Embedded Application Binary Interface”

dynamically linked (uses shared objects)

Interpreter is /lib/ld-linux-armhf.so.3

Build ID (sha1 Hash of the code)

Copyright (c) 2023 Servin Corp

ldd
(List Dynamic Dependencies)

```
nmcentire@raspberrypi:~ $ ldd /usr/bin/bash
        /usr/lib/arm-linux-gnueabihf/libarmmem-${PLATFORM}.so => /usr/lib/arm-linux-gnueabihf/libarmmem-v8l.so (0xf7b00000)
        libtinfo.so.6 => /lib/arm-linux-gnueabihf/libtinfo.so.6 (0xf7abb000)
        libdl.so.2 => /lib/arm-linux-gnueabihf/libdl.so.2 (0xf7aa7000)
        libc.so.6 => /lib/arm-linux-gnueabihf/libc.so.6 (0xf7953000)
        /lib/ld-linux-armhf.so.3 (0xf7b15000)
```

libc - C Library

.so - shared object

Copyright (c) 2023 Servin Corp

man TOPIC
(Manual Pages)

```
BASH(1)                                  General Commands Manual                                 BASH(1)

NAME
       bash - GNU Bourne-Again SHell

SYNOPSIS
       bash [options] [command_string | file]

COPYRIGHT
       Bash is Copyright (C) 1989-2020 by the Free Software Foundation, Inc.

DESCRIPTION
       Bash  is  an  sh-compatible  command  language interpreter that executes commands read from the standard input or from a
       file.  Bash also incorporates useful features from the Korn and C shells (ksh and csh).

       Bash is intended to be a conformant implementation of the Shell and Utilities portion of the  IEEE  POSIX  specification
       (IEEE Standard 1003.1).  Bash can be configured to be POSIX-conformant by default.
```

Example: man bash

NOTE: Press “q” to quit

Copyright (c) 2023 Servin Corp

Key Observations

- The bash Shell provides a command-line interface
  - CLI = Command-Line Interface
- The bash Shell can also be used to write shell scripts
  - Shell scripts often used to start up and manage an Embedded Linux system

Copyright (c) 2023 Servin Corp

hello.sh

```
nmcentire@raspberrypi:~ $
nmcentire@raspberrypi:~ $
nmcentire@raspberrypi:~ $ vim hello.sh
-bash: vim: command not found
nmcentire@raspberrypi:~ $ vi hello.sh
nmcentire@raspberrypi:~ $ cat hello.sh
#!/bin/bash

echo Hello World

nmcentire@raspberrypi:~ $ bash hello.sh
Hello World
nmcentire@raspberrypi:~ $
nmcentire@raspberrypi:~ $ chmod +x hello.sh
nmcentire@raspberrypi:~ $
nmcentire@raspberrypi:~ $ ls -l hello.sh
-rwxr-xr-x 1 nmcentire nmcentire 31 Mar 28 16:00 hello.sh
nmcentire@raspberrypi:~ $
nmcentire@raspberrypi:~ $ ./hello.sh
Hello World
nmcentire@raspberrypi:~ $
```


Copyright (c) 2023 Servin Corp

Bash Shell Written In C

- The bash shell provides a high-level CLI
- However, the bash shell itself is written in C
- Let's write Hello World in C

Copyright (c) 2023 Servin Corp

hello.c

```
nmcentire@raspberrypi:~ $ vi hello.c
nmcentire@raspberrypi:~ $
nmcentire@raspberrypi:~ $ cat hello.c
#include <stdio.h>

int main() {
        puts("Hello World");
        return 0;
}

nmcentire@raspberrypi:~ $
nmcentire@raspberrypi:~ $ gcc -Wall hello.c -o hello.c
gcc: fatal error: input file 'hello.c' is the same as output file
compilation terminated.
nmcentire@raspberrypi:~ $
nmcentire@raspberrypi:~ $ gcc -Wall hello.c -o hello
nmcentire@raspberrypi:~ $
nmcentire@raspberrypi:~ $ ./hello
Hello World
nmcentire@raspberrypi:~ $ ldd hello
        /usr/lib/arm-linux-gnueabihf/libarmmem-${PLATFORM}.so => /usr/lib/arm-linux-gnueabihf/libarmmem-v8l.so (0xf7fa6000)
        libc.so.6 => /lib/arm-linux-gnueabihf/libc.so.6 (0xf7e3f000)
        /lib/ld-linux-armhf.so.3 (0xf7fbb000)
nmcentire@raspberrypi:~ $ echo $PLATFORM

nmcentire@raspberrypi:~ $ file hello
hello: ELF 32-bit LSB executable, ARM, EABI5 version 1 (SYSV), dynamically linked, interpreter /lib/ld-linux-armhf.so.3, BuildID[
a1]=31cd63ca6cbb0712b1cf6cc4b740d7a9cd44f8de, for GNU/Linux 3.2.0, not stripped
nmcentire@raspberrypi:~ $
```


Copyright (c) 2023 Servin Corp

Shared Objects that make up hello

```
nmcentire@raspberrypi:~ $ ls -l /lib/ld-linux-armhf.so.3
lrwxrwxrwx 1 root root 30 Oct 18 07:24 /lib/ld-linux-armhf.so.3 -> arm-linux-gnueabihf/ld-2.31.so
```

Symbolic Link

```
-rwxr-xr-x 1 root root 1319784 Oct 18 07:24 /lib/arm-linux-gnueabihf/libc-2.31.so
nmcentire@raspberrypi:~ $ ls -l /lib/arm-linux-gnueabihf/ld-2.31.so
-rwxr-xr-x 1 root root 146888 Oct 18 07:24 /lib/arm-linux-gnueabihf/ld-2.31.so
nmcentire@raspberrypi:~ $
```

145K

```
nmcentire@raspberrypi:~ $ ls -l /lib/arm-linux-gnueabihf/libc.so.6
lrwxrwxrwx 1 root root 12 Oct 18 07:24 /lib/arm-linux-gnueabihf/libc.so.6 -> libc-2.31.so
nmcentire@raspberrypi:~ $ ls -l /lib/arm-linux-gnueabihf/libc-2.31.so
-rwxr-xr-x 1 root root 1319784 Oct 18 07:24 /lib/arm-linux-gnueabihf/libc-2.31.so
```

1.3MB

Copyright (c) 2023 Servin Corp

/lib/ld-linux-armhf.so.3

- Dynamic linker/loader for armhf (ARM Hard Float) architecture
- Responsible for dynamic linking and loading shared objects at runtime

Copyright (c) 2023 Servin Corp

/lib/gnueabihf/libc-2.31.0

- The C Runtime Library
- Notice the size of this library
  - About 1.3MB
  - This is why you may want use a smaller C library (e.g. uClibc, etc.)

Copyright (c) 2023 Servin Corp

Additional Skills

Copyright (c) 2023 Servin Corp

After You Login You Are At Your Home Directory

```
nmcentire@raspberrypi:~ $ pwd
/home/nmcentire
```

```
nmcentire@raspberrypi:~ $ ls
Bookshelf  Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos
```

```
nmcentire@raspberrypi:~ $ ls -a
.              .bash_logout  .cache    Documents  Music      Public      .Xauthority
..             .bashrc       .config   Downloads  Pictures   Templates   .xsession-errors
.bash_history  Bookshelf     Desktop   .local     .profile   Videos      .xsession-errors.old
```

Note: Files that begin with “.” are hidden files.

Copyright (c) 2023 Servin Corp

Bash Keeps A Command-Line History

```
nmcentire@raspberrypi:~ $ ls -a
.               .bash_logout   .cache    Documents  Music      Public      .Xauthority
..              .bashrc        .config   Downloads  Pictures   Templates   .xsession-errors
.bash_history   Bookshelf      Desktop   .local     .profile   Videos      .xsession-errors.old
```

```
nmcentire@raspberrypi:~ $ history | tail
    7  sudo systemctl start ssh
    8  sudo poweroff
    9  pwd
   10  ls
   11  ls -a
   12  ls
   13  ps
   14  echo $$
   15  ls -a
   16  history | tail
```


Copyright (c) 2023 Servin Corp

To Keep Unlimited Bash Command-Line History

- Edit ~/.bashrc
- Add these lines to end of file
  - # Unlimited Bash history
    HISTSIZE=-1
    HISTFILESIZE=-1
- After editing
  - source ~/.bashrc

Copyright (c) 2023 Servin Corp

Bash Scripts - Hello World

```
nmcentire@raspberrypi:~ $ cd Documents/
nmcentire@raspberrypi:~/Documents $ ls
nmcentire@raspberrypi:~/Documents $
nmcentire@raspberrypi:~/Documents $ vi hello.sh
nmcentire@raspberrypi:~/Documents $ chmod +x hello.sh
nmcentire@raspberrypi:~/Documents $ bash hello.sh
Hello World
nmcentire@raspberrypi:~/Documents $ ./hello.sh
Hello World
nmcentire@raspberrypi:~/Documents $ cat hello.sh
#!/bin/bash
#
# The classic hello world in bash
#

echo Hello World
```


Copyright (c) 2023 Servin Corp

Bash Script - Prompt for Name

```
nmcentire@raspberrypi:~/Documents $ vi prompt.sh
nmcentire@raspberrypi:~/Documents $ chmod +x prompt.sh
nmcentire@raspberrypi:~/Documents $ ./prompt.sh
Enter your name: Norman
Hello, Norman
nmcentire@raspberrypi:~/Documents $ cat prompt.sh
#!/bin/bash

echo -n "Enter your name: "
read ANSWER
echo "Hello, $ANSWER"
```


Copyright (c) 2023 Servin Corp

Bash Script - Select 1, 2, or 3

```
nmcentire@raspberrypi:~/Documents $ cat select1.sh
#!/bin/bash

if [ $# -lt 1 ]; then
        echo "Select 1, 2, or 3"
        exit 1
fi

echo $1

if [ $1 -eq 1 ]; then
        echo "number was one"
        exit 1
fi

if [ $1 -eq 2 ]; then
        echo "number was two"
        exit 2
fi

echo "Something else"
```

```
nmcentire@raspberrypi:~/Documents $ ./select1.sh 1
1
number was one
nmcentire@raspberrypi:~/Documents $ ./select1.sh 2
2
number was two
nmcentire@raspberrypi:~/Documents $ ./select1.sh 3
3
Something else
```

Copyright (c) 2023 Servin Corp

Bash Script - Counting and sleep

```
nmcentire@raspberrypi:~/Documents $ cat count.sh
#!/bin/bash

COUNT=0

while true ; do
        COUNT=$((COUNT+1))
        echo "COUNT: $COUNT"
        sleep 10
done
```

```
nmcentire@raspberrypi:~/Documents $ ./count.sh
COUNT: 1
COUNT: 2
```


Copyright (c) 2023 Servin Corp

Root Filesystem

Copyright (c) 2023 Servin Corp

Root Filesystem

- The last step of the Linux kernel boot process is to mount the Root Filesystem

- Deciding the exact contents of the root filesystem is a major step when building an Embedded Linux system

- Most Embedded Systems are limited in resources, hence keep the root file system as small as possible is a key goal

Copyright (c) 2023 Servin Corp

Use mount command to see what filesystems mounted

```
nmcentire@raspberrypi:~ $ mount
/dev/mmcblk0p2 on / type ext4 (rw,noatime)
devtmpfs on /dev type devtmpfs (rw,relatime,size=682572k,nr_inodes=170643,mode=755)
proc on /proc type proc (rw,relatime)
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime)
securityfs on /sys/kernel/security type securityfs (rw,nosuid,nodev,noexec,relatime)
tmpfs on /dev/shm type tmpfs (rw,nosuid,nodev)
devpts on /dev/pts type devpts (rw,nosuid,noexec,relatime,gid=5,mode=620,ptmxmode=000)
tmpfs on /run type tmpfs (rw,nosuid,nodev,size=378720k,nr_inodes=819200,mode=755)
tmpfs on /run/lock type tmpfs (rw,nosuid,nodev,noexec,relatime,size=5120k)
cgroup2 on /sys/fs/cgroup type cgroup2 (rw,nosuid,nodev,noexec,relatime,nsdelegate,memory_recursiveprot)
pstore on /sys/fs/pstore type pstore (rw,nosuid,nodev,noexec,relatime)
bpf on /sys/fs/bpf type bpf (rw,nosuid,nodev,noexec,relatime,mode=700)
systemd-1 on /proc/sys/fs/binfmt_misc type autofs (rw,relatime,fd=29,pgrp=1,timeout=0,minproto=5,maxproto=5,direct)
mqueue on /dev/mqueue type mqueue (rw,nosuid,nodev,noexec,relatime)
sunrpc on /run/rpc_pipefs type rpc_pipefs (rw,relatime)
tracefs on /sys/kernel/tracing type tracefs (rw,nosuid,nodev,noexec,relatime)
debugfs on /sys/kernel/debug type debugfs (rw,nosuid,nodev,noexec,relatime)
configfs on /sys/kernel/config type configfs (rw,nosuid,nodev,noexec,relatime)
fusectl on /sys/fs/fuse/connections type fusectl (rw,nosuid,nodev,noexec,relatime)
/dev/mmcblk0p1 on /boot type vfat (rw,relatime,fmask=0022,dmask=0022,codepage=437,iocharset=ascii,shortname=mixed,erro
s=remount-ro)
tmpfs on /run/user/1000 type tmpfs (rw,nosuid,nodev,relatime,size=189356k,nr_inodes=47339,mode=700,uid=1000,gid=1000)
gvfsd-fuse on /run/user/1000/gvfs type fuse.gvfsd-fuse (rw,nosuid,nodev,relatime,user_id=1000,group_id=1000)
nmcentire@raspberrypi:~ $
```


Copyright (c) 2023 Servin Corp

Use mount command to see what filesystems mounted

```
nmcentire@raspberrypi:~ $ cat /proc/filesystems
nodev   sysfs
nodev   tmpfs
nodev   bdev
nodev   proc
nodev   cgroup
nodev   cgroup2
nodev   cpuset
nodev   devtmpfs
nodev   configfs
nodev   debugfs
nodev   tracefs
nodev   securityfs
nodev   sockfs
nodev   bpf
nodev   pipefs
nodev   ramfs
nodev   rpc_pipefs
nodev   devpts
        ext3
        ext2
        ext4
        vfat
        msdos
nodev   nfs
nodev   nfs4
nodev   autofs
        f2fs
nodev   mqueue
nodev   binder
nodev   pstore
        fuseblk
nodev   fuse
nodev   fusectl
```


Copyright (c) 2023 Servin Corp

RPi Root Filesystem

```
nmcentire@raspberrypi:~ $ ls / | sort
bin
boot
dev
etc
home
lib
lost+found
media
mnt
opt
proc
root
run
sbin
srv
sys
tmp
usr
var
```


Copyright (c) 2023 Servin Corp

/bin

- Binary files needed at all times

- RPi has lots of files in /bin!

```
nmcentire@raspberrypi:~ $ ls /bin | wc -l
1327
```


Copyright (c) 2023 Servin Corp

/boot

- Files used to boot the system

```
nmcentire@raspberrypi:~ $ ls /boot/
bcm2708-rpi-b.dtb        bcm2710-rpi-2-b.dtb       bcm2711-rpi-cm4.dtb     fixup4.dat     kernel7.img        start4.elf
bcm2708-rpi-b-plus.dtb   bcm2710-rpi-3-b.dtb       bcm2711-rpi-cm4-io.dtb  fixup4db.dat   kernel7l.img       start4x.elf
bcm2708-rpi-b-rev1.dtb   bcm2710-rpi-3-b-plus.dtb  bcm2711-rpi-cm4s.dtb    fixup4x.dat    kernel8.img        start_cd.elf
bcm2708-rpi-cm.dtb       bcm2710-rpi-cm3.dtb       bootcode.bin            fixup_cd.dat   kernel.img         start_db.elf
bcm2708-rpi-zero.dtb     bcm2710-rpi-zero-2.dtb    cmdline.txt             fixup.dat      LICENCE.broadcom   start.elf
bcm2708-rpi-zero-w.dtb   bcm2710-rpi-zero-2-w.dtb  config.txt              fixup_db.dat   overlays           start_x.elf
bcm2709-rpi-2-b.dtb      bcm2711-rpi-400.dtb       COPYING.linux           fixup_x.dat    start4cd.elf
bcm2709-rpi-cm2.dtb      bcm2711-rpi-4-b.dtb       fixup4cd.dat            issue.txt      start4db.elf
nmcentire@raspberrypi:~ $
```


Copyright (c) 2023 Servin Corp

/dev

- Block and Character Devices

```
nmcentire@raspberrypi:~ $ ls /dev
autofs           gpiochip1  loop6          ram0    shm      tty20  tty38  tty55  vchiq     vcsm-cma  video16
block            gpiomem    loop7          ram1    snd      tty21  tty39  tty56  vcio      vcsu      video18
btrfs-control    hidraw0    loop-control   ram10   stderr   tty22  tty4   tty57  vc-mem    vcsu1     video19
bus              hidraw1    mapper         ram11   stdin    tty23  tty40  tty58  vcs       vcsu2     video20
cachefiles       hidraw2    media0         ram12   stdout   tty24  tty41  tty59  vcs1      vcsu3     video21
cec0             hwrng      media1         ram13   tty      tty25  tty42  tty6   vcs2      vcsu4     video22
cec1             i2c-20     media2         ram14   tty0     tty26  tty43  tty60  vcs3      vcsu5     video23
char             i2c-21     media3         ram15   tty1     tty27  tty44  tty61  vcs4      vcsu6     video31
console          initctl    mem            ram2    tty10    tty28  tty45  tty62  vcs5      vcsu7     watchdog
cpu_dma_latency  input      mmcblk0        ram3    tty11    tty29  tty46  tty63  vcs6      vga_arbiter  watchdog0
cuse             kmsg       mmcblk0p1      ram4    tty12    tty3   tty47  tty7   vcs7      vhci      zero
disk             kvm        mmcblk0p2      ram5    tty13    tty30  tty48  tty8   vcsa      vhost-net
dma_heap         log        mqueue         ram6    tty14    tty31  tty49  tty9   vcsa1     vhost-vsock
dri              loop0      net            ram7    tty15    tty32  tty5   ttyAMA0  vcsa2     video10
fb0              loop1      null           ram8    tty16    tty33  tty50  ttyprintk  vcsa3     video11
fd               loop2      port           ram9    tty17    tty34  tty51  uhid      vcsa4     video12
full             loop3      ppp            random  tty18    tty35  tty52  uinput    vcsa5     video13
fuse             loop4      ptmx           rfkill  tty19    tty36  tty53  urandom   vcsa6     video14
gpiochip0        loop5      pts            serial1 tty2     tty37  tty54  v4l       vcsa7     video15
```


Copyright (c) 2023 Servin Corp

/etc

- Configuration Files

Copyright (c) 2023 Servin Corp

/home

- Home Directories

Copyright (c) 2023 Servin Corp

/lib

- Libraries

Copyright (c) 2023 Servin Corp

/lost+found

- Files recovered from filesystem recovery

Copyright (c) 2023 Servin Corp

/media

- Mount point for removable media
- Example: USB Storage, SDCard

Copyright (c) 2023 Servin Corp

/mnt

- Mount point for non-removable media

Copyright (c) 2023 Servin Corp

/opt

- Optional Software

Copyright (c) 2023 Servin Corp

/proc

- Process status and more
- Tons of useful info stored under /proc

Copyright (c) 2023 Servin Corp

/root

- Home directory for root user
- NOTE: Root user is all powerful - be careful how you use this account

Copyright (c) 2023 Servin Corp

/run

- Run info such as PID (process ID)

Copyright (c) 2023 Servin Corp

/sbin

- Superuser Bin
  - Most require running as superuser / root

Copyright (c) 2023 Servin Corp

/sys

- System Info
  - Another interesting subdirectory with lots of dynamic systems info

Copyright (c) 2023 Servin Corp

/tmp

- Temporary Files

Copyright (c) 2023 Servin Corp

/usr

- A whole additional subdirectory for additional programs
  - /usr/bin
  - /usr/sbin
  - /usr/include - Header files for programming
  - /usr/lib

Copyright (c) 2023 Servin Corp

/var

- Variable length files

Copyright (c) 2023 Servin Corp

Questions?

Copyright (c) 2023 Servin Corp