Socket Files and IOCTL's, More Debugging

UNX511 Week 4 Class 2

John Sellens

May 28, 2025

Seneca Polytechnic

Outline

Introduction to IPC with Sockets

More on IOCTLs

Just a Little More on Debugging

Introduction to IPC with Sockets

IPC – Inter Process Communication

- One process often needs to communicate with another
    - Consider your web browser getting a page from a web server
- Sometimes on the local machine, local network, or across the internet
- We will look at several IPC methods in this course
- Today – an introduction to sockets

What Is a Socket?

- Sockets allow bi-directional communication between processes
- They can be local only, or available across the network
- There are many different socket types (or families) – see `socket(2)`
- The most common are AF_INET (IPv4 internet protocols) and AF_INET6
- Today we will look at AF_UNIX – “UNIX Domain Sockets”
  - For local communication on a single machine
- A UNIX domain socket appears in the file system
- Similar to named pipes (FIFOs), but named pipes are unidirectional
- Sockets Tutorial: `https://www.linuxhowtos.org/C_C++/socket.htm`

How to Use Sockets

- The general method for using sockets is similar across families
- Connections are made by a client process connecting to a server process
- The server process gets ready
  - socket() – returns a file desriptor
  - bind() – attach to a network port or UNIX domain socket
  - listen() – wait for a client to ask to connect
  - accept() – accept a connection, returns a read/write file descriptor
- The client process initiates a connection to the server
  - socket() – returns a file desriptor
  - bind() – only if network, establishes local network port
  - connect() – connect to a server
- Processes then read/write until the connection until `close()`

UNIX Domain Socket Code Example

- Let’s have a look
- `week4_2/2_socket_example` – simple client/server pair

More on IOCTLs

Network Interface IOCTLs

- Last week we looked at disk driver IOCTLs
- Now let's get information about a network interface
- Call socket() to open in AF_INET socket
  - SOCK_STREAM for TCP/IP – reliable, ordered, packet stream
  - SOCK_DGRAM for UDP – User Datagram Protocol
    - Sometimes called Unreliable Datagram Protocol
    - Toss a packet onto the network, hope it gets delivered
- Use the file description returned by socket() to query an interface
- Most machines have 2 or more network interfaces
- `week4_2/2_ioctl_example` – etherCtrl2.cpp example

Just a Little More on Debugging

Another Debugging Example

- Last week we talked in general about debugging and looked at `gdb`
- As we discussed, sometimes we just want to spit out some output
- Here's an example using a `#define DEBUG` and `#ifdef DEBUG`
    - Wrapping debug code like this is very common
    - You don't have to include your debugging code in your production code
- This examples opens a new file, and sends stderr there
    - Remember that you can use `tail -f filename` on a log file
- `week4_2/3_debugging` – conversions code with debugging

Summary

- Introduction to sockets – UNIX domain sockets
- Some network-related IOCTLs
- A little more on debugging techniques