EE / CprE / SE 491 – sddec20-05

An Advanced Networking Outreach Activity for Kids

Bi-Weekly Report #2
2/3/2020 – 2/16/2020
Client & Faculty Advisor: Dr. Tom Daniels

Team Members

Grayson Cox | Chief Engineer - UI/Control Systems | Report Manager
Austin Dvorak | Chief Engineer - Networking Systems | Test Engineer
Ryan Newell | Meeting Leader | Progress Tracker
Spencer Parry | Chief Engineer - UI/Control Systems
Ross Thedens | Meeting Scribe | Hardware Manager

Reporting Period Summary

Over the past two weeks, we began planning our design for the project in earnest. In a meeting with our advisor, we constructed a rudimentary design diagram defining the basic types of nodes in our network (see addendum). Based on this diagram and discussion, we continued researching potential technologies to use for our project. In particular, we began finding and testing out routing distributions for Linux and ad-hoc networking protocols. Our discussions from working on our first lightning talk and the design document led us to do further research into appropriate single-board computers, network adapters, and other hardware for our project. At this point, we have a clear picture of what major features our project may entail and the highest-level components it entails. We are beginning to figure out the lower level details as we explore the existing technologies and discover what is feasible.

Reporting Period Accomplishments

- Networking Investigations (Austin)
    - Started research on possible ad-hoc networking protocols (BATMAN and OLSR)
        - Found out that Vanilla Ad-Hoc networking is no longer used, Some sort of advanced mesh networking protocol such as BATMAN and OLSR is used to aid in ease of use
    - Found possible routing distributions (OpenWRT and IPFire)
        - The distro we use will dictate which protocol we will use (OpenWRT would use BATMAN and IPFire would use OLSR)
- Single Board Computer Research (Ryan)

    - Researched several possible boards for use as nodes in the project
        - Rock Pi S
            - Has onboard ethernet and WiFi, 256-512 MB RAM sizes should be sufficient for wireless networking, Extremely small 1.7-inch dimensions, Would be very cheap (~$20)
            - Less documentation available; may find less support online when trying to work with peripherals (cameras, sensors, etc.)
        - Raspberry Pi Zero W
            - Has onboard WiFi; 512 MB RAM should be sufficient for wireless networking, Would be even cheaper (~$10 for board only), Lots of documentation available; known compatibility with OpenWrt
            - About 2.5 inches long, 1 inch wide, ethernet available via USB adapter
    - Decided Raspberry Pi family is clearly the best choice; may need to opt for Raspberry Pi 2/3B+ boards if multiple USB peripherals are needed
- Networking and Hardware Investigations (Ross)
    - Tested video streaming over UDP using a C socket program
    - Installed OpenWrt on a Raspberry Pi 3B+
        - Found that installation procedure is very simple; OS images exist preconfigured for Raspberry Pi
    - Tested using a Raspberry Pi ribbon cable camera module (v1.3) in OpenWrt
        - Found that OpenWrt's opkg package manager does not contain the "raspicam" package available through apt on Raspbian. Attempts to install the package through python/pip were not successful due to a missing library file.
        - Discovered that packages can be built into the OpenWrt image, but the procedure is not well documented and could be troublesome if the OS needs to be updated or reflashed.
        - Determined that OpenWrt contains several driver packages related to USB webcams that allow for streaming. These will likely be our best bet for a camera module.
- UI Design (Grayson & Spencer)
    - Experimented with different UI layouts (Grayson)
        - Made a demo layout with a side navigation bar for seeing all components in the network
    - Looked for ways to implement UI components with minimal memory usage (Spencer)

Pending Issues

- Determine video/sensor feasibility in routing distributions considering incompatibility of Raspberry Pi camera module (Ross)

- Find out what video/sensor options are supported in OpenWrt and IPFire (USB webcams, etc.)
- After researching, procure several of these options for testing; use this information when making a final hardware/OS choice so that video and sensor data collection does not become a major roadblock

Individual Contributions

Team Member	Contribution	Reporting Period Hours	Total Hours
Grayson Cox	Researched Angular and Typescript for UI design, software development standards slide in lightning talk, wrote sections 1.6 (Assumptions & Limitations), 1.7 (Expected End Product & Deliverables) in the design document, updates to team website, Coordinated a meeting with Dr. Daniels to discuss design document, Reserved meeting rooms for weekly team meetings	12	24
Austin Dvorak	Research on ad-hoc networking protocols and routing distributions, Testing with OpenWrt, Functional requirements slide for lightning talk, work on sections 2.1 (Proposed Approach) and 2.2 (Design Analysis) for design document	12	24
Ryan Newell	Single Board Computer suitability research for network nodes, environmental requirements slide for lightning talk, sections 1.4 (Requirements) and 2.3 (Development Process) for design document	12	24
Spencer Parry	Researched Angular and Typescript for UI design, hardware standards slide and final production for lightning talk, contributed to sections 1.3 (Operational Environment), 1.5 (Intended Users and Uses) for design document	12	24
Ross Thedens	Video Streaming UDP Socket in C for network nodes, OpenWrt camera investigations for network nodes, economic requirements slide for lightning talk, sections 1.2 (Problem and Project Statement) and 2.4 (Conceptual Sketch) for design document, took and compiled notes from meeting with Dr. Daniels (advisor)	12	24

Plans for Next Period

In the coming two weeks, we will be finishing the first version of our project's design document and continuing to refine our design ideas. Having written a rough draft, we will meet with Dr. Daniels early in the week to discuss our work so far and use his feedback to construct a final "first draft" for Feb. 23. Since sections 3-5 of the design document are expected to have substantive detail in the second version, we will be working to solidify which features we want to implement and in what order. To this end, we will continue exploring GUI and network routing technologies to assess the feasibility of our planned features and to estimate required effort.

- Design document (version 1)
    - Finish rough drafts of relevant sections before meeting with Dr. Daniels
    - Meet with Dr. Daniels to discuss our writing
    - Finish final draft
- Networking (Austin)
    - Start experimenting with BATMAN and try using IPFIRE on Raspberry Pi
- Networking and Hardware (Ross)
    - Find, purchase, and eventually test a suitable camera that works in OpenWrt on Raspberry Pi
    - Research temperature, power consumption, battery level sensors; research battery packs and cases to create the system nodes
- GUI (Grayson, Spencer)
    - Experiment more with Angular UI components
    - Practice making lightweight UI's that take up little memory
- Hardware Support (Ryan)
    - Work with Ross to determine which Raspberry Pi board is best suited to our nodes, and whether multiple types may be warranted (i.e. 3B+ for camera nodes, Zero W for the most barebones nodes)

Addendum: Advisor Meeting Summary

On Feb. 4, we met with Dr. Daniels to discuss the general architecture of the project. During the meeting, we created a basic diagram of the system. It is shown below. All the blocks labeled “node” will consist of single board computers, while the Instructor Station will be a laptop. The Admin display will likely be the laptop screen, while the Audience Display will be an external monitor or projector. One of the nodes (Network Master Node) will serve as a “base station” that allows the Instructor station access to the network configuration via a control program we will develop. Besides the diagram, we discussed visual cues for the Raspberry Pis to indicate if nodes are connected (via LEDs), a registration system for operating two sets of the nodes in a single network or two separate networks, and our general thoughts on potential protocols for the network (primarily B.A.T.M.A.N.).