# Milestone #3

NetBSD Wifi Browser Project 2022

Dylan Roy, Stephen Loudiana, Kevin McGrane

## 1 General Progress Report

Progress on the API for WifiBrowser was still slower than anticipated, but still yielded a lot of success during the last two weeks. The delays experienced for the goals of the milestone that were not met (passkey hashing and network configuration deletion), there was successes in other areas that counteract the failure. As a result of changing strategies from leaning heavily on `wpa_ctrl` for interfacing with `wpa_supplicant` to using File I/O and directly manipulating the configuration file has simplified many things and fixed many of the issues experienced with the prior strategy. Moving forward, progress should be more consistent and *ideally* more substantial as the work required for the goals missed and the upcoming goals for milestone 4 share many similarities in implementation and strategy.

Progress on CLI component has been steady with little to no blocks during development. CLI can interact with API component. The implementation of the TUI is well on its way now, the last couple weeks has been spent on doing some final research on `ncurses` and possible roadblocks or issues that may arise in the coming weeks of implementation. The rest of the time has been used to create a test program to get a working example of how to use `ncurses` and what we need to know to properly implement the needed requirements to get a working TUI.

**Accomplishments**

- API can basically connect to any kind of network (some network configurations might require more fields than what's currently supported, but the change to support it is extremely minimal)
- API actually saves the configuration file
- API testing has been expanded and refactored for easier expansion in the future
- API and CLI components can interact and streamline connection processes for NetBSD
- TUI Scaffolding and Design

**Targets Not Met:**

- API does not support passkey/password hashing yet → solution isn't as trivial as previously thought
- API does not support individual deletion of networks due to a larger delay on isolation of specific information with C string scanning. However, there is support for deleting all of the network configurations
- Targets for CLI component has been met for this milestone. Cleaner organization of CLI will need to be implemented before deployment.
- TUI devised testing strategy was not completed due to the majority of the efforts being put towards the implementation and testing of `ncurses` code in general

## 2 Git Branch Report

All work has been performed on `master` against best practice. It might actually behoove us to work on separate branches due to the experienced difficulties with people working on master at the same time.

## 3 Individual Progress

**Kevin:** 5 hours

- Reformatted and cleaned up time log
- worked more on implementation of `ncurses` in test program
- researched more possible roadblock when implementing the TUI with the API
- Completed a working `ncurses` test program printing to a screen and tasking input from users

**Dylan:** 15 hours, 55 minutes

- API now has to run with `sudo` due to `wpa_supplicant` not behaving as expected
- API Fixed PSK issue → `wpa_supplicant` doesn't throw a fit anymore
- API Added a `cleanConfig()` function for easier testing results checking
- API Refactored tests to be more abstracted; will be refactored more after API is mostly complete
- API Started individual deletion and editing of networks; have to determine best strategy for string scanning.
- API Started modifying list functions to output the ssid's isolated instead of all of the extra `wpa_supplicant` information → this is another case of solving the C string scanning problem
- API Configurations actually save now
- API Supports auto connect to WPA-PSK networks, and manual connection to most networks → the only thing preventing wider support is adding more fields to our configuration struct
- Initialized Milestone 3

**Stephen:** 9 hours, 17 minutes

- Makefile to combine CLI and API components developed
- Additional CLI file to process commands has been added
- API methods are now implemented in CLI component
- CLI has the ability to connect to `wpa_supplicant`
- CLI has the ability to establish manual configuration of a network
- CLI has the ability to establish auto configuration of a network
- CLI has the ability to list available networks
- CLI has the ability to list configured networks