Milestone #4
NetBSD Wifi Browser Project 2022
Dylan Roy, Stephen Loudiana, Kevin McGrane

1 General Progress Report

Progress on the API front again has been slow, but with the minor progress and the implementation of the passphrase hashing, it has become apparent how close the API is to v1.0 completion. Moving into the next milestone, the implementation of manual configuration will be changed to both simplify the implementation and to make configuration more expandable (and able to better accommodate the large amount of available configuration fields). An unfortunate piece of information discovered about `wpa_supplicant` was that password hashing for WPA-EAP networks is not worth while considering that the accepted hashing algorithm does not work for passwords longer than 14 characters, requiring them to be stored in plain text as a result. This can be somewhat circumvented with better file permissions, which will be considered during the determination of circumventing the necessity of root privileges.
The actual implementation of the NetBSD TUI application is now under way, the next few days will be spent starting the TUI and building out the design and layout while also working more with ncurses to design an optimal layout for the TUI. Once the layout and design is at a place where we feel comfortable with it we will start to implement the connections with the API which will be the final part of the TUI implementation

Accomplishments

- API supports passphrase hashing

- API has generalized and improved makefile

- API can edit already configured networks

- TUI Scaffolding and Design

- TUI Devise testing strategy

Targets Not Met:

- Use API without root privileges

- TUI Support for current API features

- TUI Create Tests

2 Git Branch Report

One additional branch was used to implement the passphrase hashing (branch `crypto`) in order to prevent compile issues for the others while figuring out the details of needing extra dependencies and the solution not being trivial. Beyond this, most work was performed on `master`.

3 Individual Progress

Kevin: 4.5 hours

- Completed Ncurses test file

- Started implementation of NetBSD Wifi Browser TUI

- Tested design and layout ideas

- Created testing scheme for TUI

Dylan: 12 hours, 10 minutes

- Implemented the ability to edit already configured networks

- Determined and changed strategy for passphrase hashing

- Implemented passphrase hashing in API

- Improved Makefile

- Cleaned up and abstracted test file

- Solve lab machine issues (there were many the first week of milestone 4)

Stephen: 30 minutes

- Organized CLI to be more neat

- Additional security testing of functions