TauNet Project Evaluation Report

Document Version 1.0

1.0 Abstract

This TauNet implementation was an overall success. Collaboration, good leadership, and structured process resulted in the success of this project. Factor that made this project difficult included scheduling, deviating for the design process, and code bugs.

2.0 Summary

My design process started with a c++ prototype that sent plaintext messages over a local network. After confirming that c++ was a viable option for implementation I switched gears and focused on documentation. With class collaboration a protocol was developed that standardized what it mean for a program to speak TauNet. With this document in hand I developed a Software Requirements Specification (SRS). By developing the SRS I was able to clearly define what it meant to be a functional TauNet node. Using the SRS and my prototype it was trivial to develop the Software Design Specification (SDS) that defined how this project would be implemented.

With most of the proper documentation in hand I began coding. For the most part, because of proper planning, the implementation went well. I followed my design document and it led me to a state of near completion. At this time I had a TauNet implementation that sent non-plaintext messages between nodes; this was very exciting. However, when it became time to message the Echo Server and confirm that my client was indeed speaking TauNet, bad things happened. The messages I decrypted from the Echo Server were not in plaintext. It became clear that something was wrong with my encryption protocol. I spent much time debugging, reimplementing, and testing. This process finally led me to a bug - my rc4 prng was not producing a proper keystream. This bug was not a result of poor coding, but a result of not having a complete understanding of the CipherSaber protocol. My assumption was that the rc4 function was supposed to behave like a prng and return a keystream of numbers. This was not the case. The keystream was supposed to by random bytes and the c++ int variable was not appropriate to use. After fixing this bug everything went smoothly to completion.

3.0 Successes

The largest factor that influenced success was collaboration between students. With the combined knowledge of all of us we were able to help everyone develop a clear understanding of what we were to build and some of the challenges we might encounter. The leadership of our professor Bart Massey kept the project from going wildly astray. By following a formal process I was able to, rather painlessly, develop a non-trivial software intensive system.

4.0 Failures

A user requirements document was never drafted. While the user requirements were verbalized and fairly easy to remember, I feel that documentation of these requirement would have been

helpful and should not have been omitted. A project timeline was also omitted. This was less of a detriment, but perhaps with a timeline I would have made an encryption prototype and avoided the headache of debugging a full program.

5.0 Test Plan Execution

The unit test were extremely helpful to have in place when I was debugging my encryption. These test allowed me to easily confirm that my encryption and decryption were indeed still inverse functions. Unit tests also existed that decrypted known good encrypted values. These were essential for knowing when I had fixed my bug.

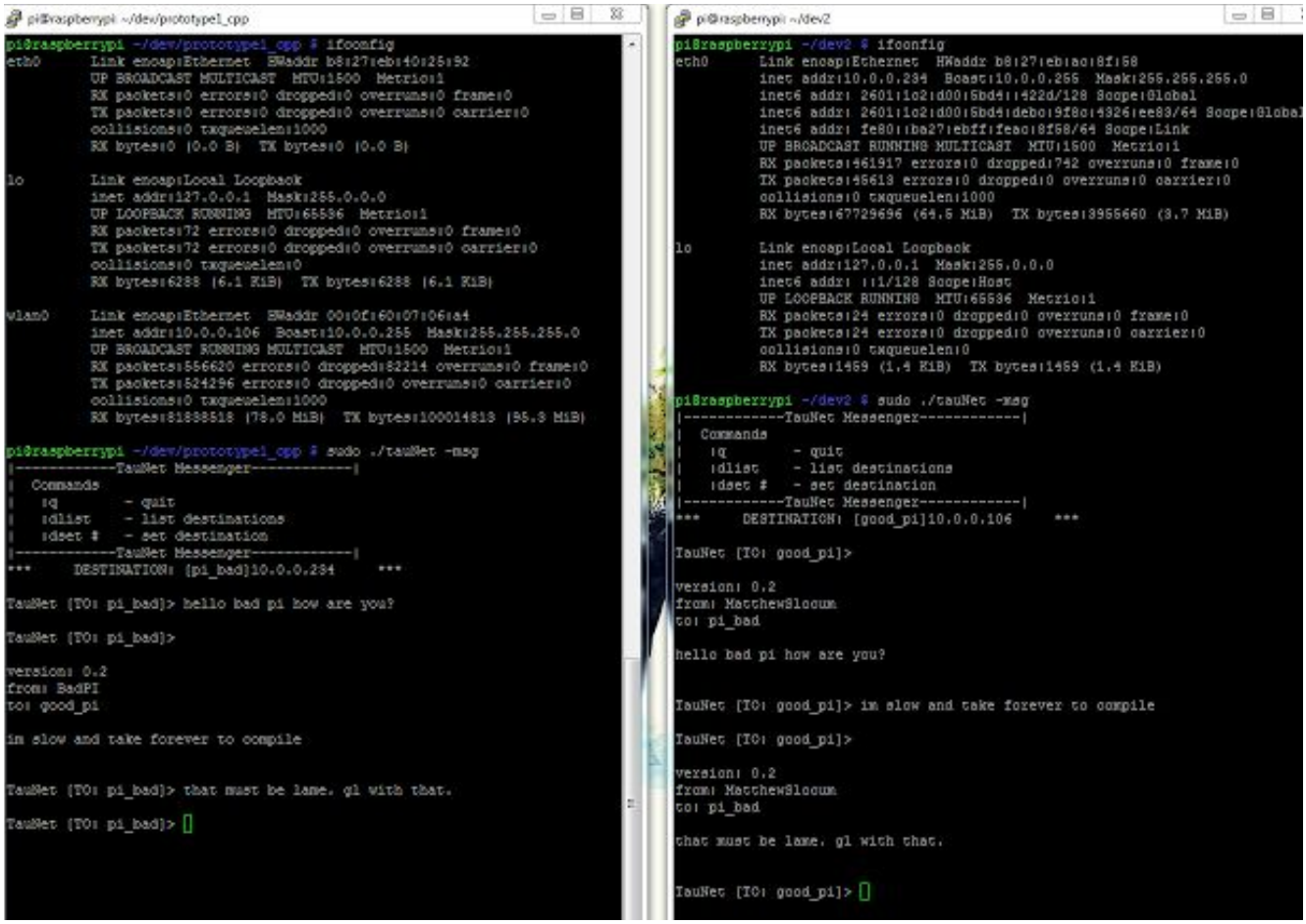
5.1 Unit Tests

At the end of development all unit tests passed.

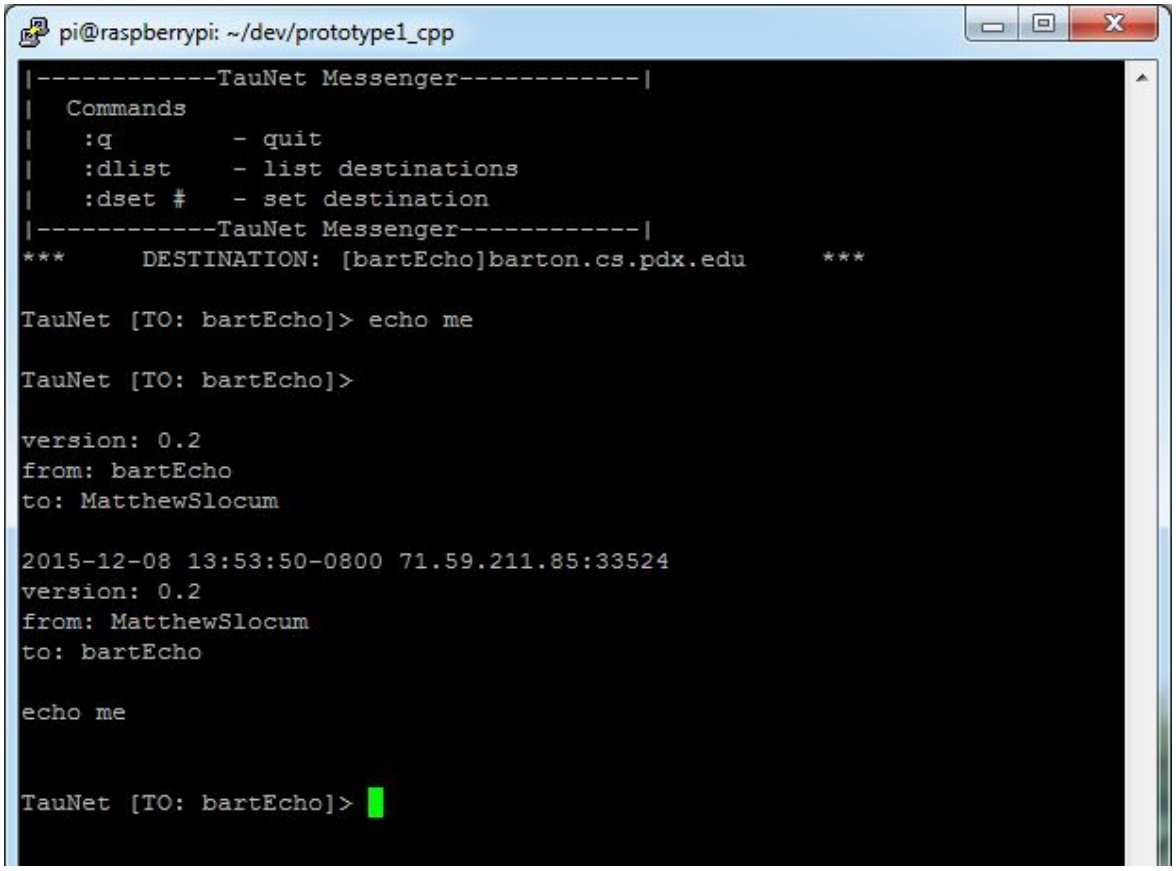
5.2 System Acceptance Tests

At the end of development all System Acceptance tests passed.

5.2.1 Normal Operation Test - Local Pi's



5.2.2 Normal Operation Test - Communication with the TauNet echo server



5.2.3 Normal Operation Test - Communication with other TauNet nodes

This is a partial log file of the interaction.

|----------TauNet Messenger----------|
| Commands
| :q - quit
| :dlist - list destinations
| :dset # - set destination
|----------TauNet Messenger----------|
*** DESTINATION: [castle]castlez.ddns.net ***

TauNet [TO: castle]> I hope this one works

TauNet [TO: castle]>

version: 0.2
from: castlez
to: 71.59.211.85

any luck??

TauNet [TO: castle]> oh ya

TauNet [TO: castle]> lots of luck

TauNet [TO: castle]>

version: 0.2
from: castlez
to: mattslocum

did you get message 2 (this one)?

TauNet [TO: castle]> oh ya

TauNet [TO: castle]> message 2 recieved

TauNet [TO: castle]> have you gotten my like 8 messages?

TauNet [TO: castle]>

version: 0.2
from: castlez
to: mattslocum

here is message 3!!

TauNet [TO: castle]> omg was that message 3!