TauNet Client (“Taurus”)

System Requirements Specification

Revision 2

Table of Contents

1 Introduction

1.1 Purpose of Specification

This document specifies the requirements for a client, tentatively named Taurus, implementing the TauNet protocol. It addresses the hardware and software requirements for operating the software system; assumptions about the network environment, including other TauNet agents; functional features of the user interface; security concerns; and ongoing support. Finally, it describes two situations in which Taurus might be used.

The specification is presented for the consideration of the TauNet project team. When finalized, this document will guide the design and development of Taurus and define criteria for acceptance of the final product.

1.2 Purpose and Scope of Project

Taurus will be a software system which facilitates the transmission and receipt of brief private messages using TauNet, a protocol designed for small communities of moderately technical users. It will allow one person to join a closed group of TauNet users and send brief messages to other members of the same TauNetwork, whether those other members are using Taurus or another client implementing the same version of TauNet. (The protocol version implemented in Taurus will be the most recent at the time of delivery, and is therefore unspecified here.) These messages will be encrypted to protect them from unauthorized inspection.

Taurus is intended to be deployed on a Raspberry Pi 2 Model B running a version of Raspbian Linux which is current and supported at the time of delivery. This combination of hardware and operating system is inexpensive, widely available, and well-documented, precluding the need to trust a hosting service, or other outside entity in order to access TauNet.

The first version of Taurus is to be delivered no later than early December 2015 by a single developer. The breadth of functionality in the first version is constrained accordingly.

1.3 Definitions and References

1.3.1. **Raspberry Pi** is the brand name of a series of tiny computers, typically sold without cases or storage as a low-cost educational or project platform. http://www.raspberrypi.org

1.3.2 **TauNet** is a protocol for the encrypted transmission of messages. http://l.pdx.cat/b0c8ce

1.3.3 A **TauNeighbor** is another member of a shared TauNetwork.

1.3.4 A **TauNetwork** is a group of TauNet clients which share a user table and encryption key.

2 Network

2.1 User Table

2.1.1 Membership in a TauNetwork is defined as sharing two identical pieces of data: a user table and an encryption key. Taurus will store one set of this information. It will also know or infer which user is the one it will be sending and receiving messages for.

2.1.2 Each entry in the user table will contain the data specified in the TauNet protocol for identifying and addressing one TauNeighbor. (As of version 0.1, this consists of a unique username and an IP address.) Because clients share a user table, this includes information about the Taurus instance which is storing the table.

2.1.3 The user table may contain information about anywhere between 2 and 300 TauNeighbors.

2.2 Persistent Connectivity

2.2.1 The Raspberry Pi on which Taurus is running will be presumed to have a consistent connection to the global internet.

2.2.2 The other hosts in the user table will also be presumed to be consistently available on the global internet and running a client implementing the same version of TauNet as Taurus.

2.2.3 The unavailability of any one host will not prevent messages from successfully being delivered to other TauNeighbors.

3 Interface

3.1 Hardware and Setup

3.1.1 All of Taurus's features will function acceptably on a Raspberry Pi 2 Model B.

3.1.2 Taurus is intended to operate in an environment of Raspbian Linux. The user is presumbled to be able to navigate that operating system sufficiently comfortably to either attach peripheral devices to the host or connect to it remotely, install Taurus, and find and run it.

3.1.2 Taurus will not require a graphical user interface. It will assume the user has a keyboard and a monitor (or some equivalents), connected either to the Raspberry Pi host or to a remote machine from which the user connects to the host.

3.2 User Table

3.2.1 During initial setup of Taurus, the user will be able to supply an encryption key and a user table to establish the client's membership in a TauNetwork.

3.2.2 The user table will not be modifiable after creation.

3.2.3. After the user table is initialized, the user can use Taurus to view the list of TauNeighbors.

3.3 Messages

3.3.1 Taurus will allow the user to type and send a message to anyone else on the same TauNet. The length and content type of the message will be determined by the TauNet protocol.

3.3.2 When a message is succesfully sent, Taurus will display a confirmation notice. When transmission is attempted but not completed, Taurus will display an error notice.

3.3.3 Taurus will display a notice when a message has been received, and allow the user to view the decrypted message.

3.3.4 Taurus will display as much message metadata, such as timestamps and sender information, as is useful and available. What metadata is available will be determined by the TauNet protocol.

4 Security

4.1 Hardware

4.1.1 Taurus is designed to be used on a hardware and operating system platform totally under the control of its user. This configuration will be strongly recommended in its documentation.

4.2 Messages

4.2.1 Messages will be sent only to the recipients to whom they are designated. Only messages addressed to Taurus's user will be accepted.

4.2.2 All messages will be encrypted in the manner specified in the TauNet protocol.

5 Support

5.1 Source and Documentation

5.1.1 Taurus's source code will be publicly available and licensed. The source will be accompanied by documentation which explains how to install and use Taurus.

5.1.2 The documentation will assume that the user has already set up a Raspberry Pi with Raspbian and is comfortable operating it.

5.1.3 Issues and pull requests to the source code will be addressed at the author's discretion.

5.2 Service and Availability

5.2.1 Once Taurus has been released, no ongoing service will be provided to its users beyond a best effort to keep the source and documentation up-to-date.

6 Scenarios

6.1 Within a Household

Alice, Bob, and Carol live together, and sometimes wish to send quick messages across the house: “I’m going out, does anyone need anything?” or “I’m expecting Dave, if you hear the door please let him in.” Alice has some extra Raspberry Pis lying around from a robotics project. They create a TauNetwork to allow them to send their personal messages to each other without having to go through an outside service.

1. Each person selects a username.
2. They agree on an encryption key.
3. They create a user table with their usernames and LAN-internal IP addresses.
4. Each person copies the user table and key into their own instance of Taurus.
5. Each Taurus instance shows the other two usernames as TauNeighbors.
6. Alice, Bob, and Carol can select one of their two neighbors and send a message to that person.

6.2 Across a City

Ellen and Finn are the organizers of an urban mesh network. Sometimes the mesh node owners want to check in with each other; for example, if a node goes down, or they want to follow up on a conversation about some neat new radio hardware. There are enough participants in the mesh network that not everyone is comfortable sharing their phone numbers with the entire group. However, a Raspberry Pi is part of the recommended node hardware setup, so almost everyone already has one. Ellen and Finn set up a TauNetwork so everyone can send messages to each other privately.

1. Ellen and Finn select an encryption key and use their existing list of mesh node owners and network information to create a user table.
2. At the next meeting of mesh network participants, they pass around a USB stick with the user table and encryption key. Each node owner makes a copy on their own hardware.
3. Each node owner (including the organizers) load the user table and encryption key into their own copy of Taurus.
4. Each person can see the list of other mesh network participants, but not any more personal data than a name.
5. When a node goes down, the first few people to notice can send messages to the node owner and find out what’s up. If the node owner needs help fixing it, they can send messages to the organizers or other node owners with questions.