TauNet Software Test Plan

Table of Contents

1. **Introduction**
2. **Tests**

1. Introduction

a. The purpose of this document is to record the results of tests run on the TauNet system. It will help to demonstrate the usability of the system.

2. Tests

a. Ciphersaber 2 tests

i. Decryption with human memorable component

1. Input:
    a. Message: “Al Dakota buys”
    b. Key: “Al”
2. Expected output:
    a. “mead”
3. Result
    a. expected output achieved

ii. Encryption and decryption with echo server (picture in repo)

1. Input:
    a. Message: “testing…” to encryption
    b. Key: “password”
2. Expected output:
    a. Return message containing header info
3. Result:
    a. expected output achieved

b. **Client tests**
    i. **Maximum message length**
        1. Input: message size less than 1 kb
        2. Expected output: Error message and new attempted
        3. Result: expected output achieved
    ii. **User list**
        1. Input: userlist file is changed
        2. Expected output: userlist in client changes to reflect file
        3. Result: expected output achieved
    iii. **All flow control elements work as expected**
        1. Input: visual inspection of all possible flow control executions
        2. Expected output: flow control works correctly
        3. Result: expected output achieved
    iv. **User cannot log in without correct key**
        1. Input: visual inspection of several incorrect keys
        2. Expected output: no access to client
        3. Result: expected output achieved
c. **Server tests**
    i. **Multiple messages can be received in quick successions**
        1. Input: two clients with messages sent to server as rapidly as possible
        2. Expected output: both messages displayed by server
        3. Result: expected output achieved
    ii. **Blank messages are discarded**
        1. Input: test function to send blank message to server
        2. Expected output: “blank message” message from server
        3. Result: expected output achieved
    iii. **User cannot log in without correct key**
        1. Input: visual inspection of several incorrect keys
        2. Expected output: no access to client
        3. Result: expected output achieved