Semi's Tracking Software Test Plan

Prepared by:
Bismar Montano
June 9th 2021

Table of Contents

1.1 Introduction

The goal and aims of the testing is to ensure that the tracking software is able to track all our semi's and ensure all filters are working as intended such as speed and location tracking, direction and cargo(along other tracking features). Ensure that access to the tracking software is locked out from remote access and only available to users in the internal network.

2.0 Objectives And Task

Phase 1 testers, users responsible for everyday tracking, should test the software to ensure proper functionality. All filters displaying the proper output, the software is tracking accurately. Phase 2 to be performed by the IT department ensures that the software is properly configured to not allow any connection via the internet and is locked down only for internal use. Phase 3 also by the IT department, or external consultants, will consist of penetration testing ensuring that although it is locked down from the intern it does not allow external users of any kind and vulnerabilities are addressed as well as ensure users are not able to escalate their own privileges to admin, unless pre approved. Reports bugs to the development team follow up with a second round of testing once they have resolved any bugs and issues identified during testing.

2.1 Objectives

Testers must test the softwares functionality and report problems in order for the development team to address any bugs encountered. Developers should report back once issues and bugs have been addressed should be entered into another round of testing to ensure proper functionality. In house bug reporting methods and communication should be utilized to maximize efficiency in completing testing for deployment.

2.2 Task

Test software filters such as direction, location, cargo tracking and ensure all the data is outputting and displaying correctly. Have testers report any bug via our inhouse bug reporting tools for easy availability. Security and configurations of the software should be tested to ensure safety and locked down of the software from external access and threats.

3.0

Scope

The software will be tested for functionality, which includes all its filters settings while keeping in mind the software performance to ensure our systems are capable of utilizing the software without needing to upgrade. Ensure that the interface is working properly and responding properly. Then move to security settings to lock it down from external access or unauthorized accounts and ensure admin accounts have appropriate privileges.

Tactics

Product testers alongside daily users should have access to the first build of the software to stress test the software for proper functionality. They should take time to test the features and report bugs as they encounter them that will impact the softwares intended use to ensure it is functioning properly and optimally with our hardware. Testing should consist of intended use of the product such as pulling data of routes for analysis.

IT teams should check the configurations to ensure our internal users can access the software without error or conflicts from our firewalls. We can also perform in house, or have external teams, pen test our software to ensure that users are not able to escalate their privileges without proper authorization. Ensure that our tracking software will not be accessible external from either remote authorized users or external threats.

4.0 Testing Strategy

Testing the functionality and features of the software we should give an allotted time of a week to give our feature testers time to test the functionality of the software. Ensuring all the filters are pulling the correct data and ensuring we can pull that data to analyse and report problems. Give time to developers to address bugs countered to be addressed and submitted to retesting.

As testers test the functionality IT department should check configuration to ensure that the firewall does not cause any problems accessing the data and should bugs be encountered they should also be reported and reconfiguration should be assessed that systems are communicating properly and securely.

Although breaking the testing into testing phases from functionality to configurations can slow the process it would allow a better approach to objectively look at the results and cycle back through each phase. Rather than testing alongside it could potentially cause slow downs and create a great round of confusions because it would make it harder to detect what broke where and at what point.

4.1 Unite testing

Definition:

At minimum users should be able to view the correct output of the data based on the filters users are using. Location and cargo should give the trucks location and their cargo type without error or provide requested data. Pulling data from the software should work as intended without error and users can login without error.

Participants:

Production testers, daily users, admin access - should test the actual functionality of the software

Methodology:

Testers will test the software in an as intended or as they would in their workflow to identify bugs that rise and will conflict with their intended workflow and use.

4.2 System and Integration Testing

Definition:

The software should be compatible with the rest of our inhouse tools for easier access and workflow and all security measures such as passwords should meet our organization's standards and integrate Microsoft Directory or easier access for users. The software must also be locked down from all external access. The software must be able to store proper system logs on user activity.

Participants:

IT department, possibly reach out to external pen-testers -to test the configuration and security of the software

Methodology:

Once users have completed a first round of unit testing we can check configurations to ensure that the software communicates well with our firewall and security measures do not break intended workflow and repeat these steps as necessary to ensure proper functionality.

4.3 Performance Stress Testing

Definition:

Users should push the software to see how much of a workload it can perform, such as multiple users accessing the same semi's information and pushing the software to its limit with inputs to access a large variety of data from semi's and ensure it can meet a high demand of request and usage. The IT stresses the security features and configurations of the software to prevent unauthorized access.

Participants:

Product testers and intended daily users will test for workload flow and IT will test for security configs.

Methodology:

Users will intensify their intended workflow to request data to test how much the software can process at any given time and multiple users will attempt to access the same data to test the software output. IT can utilize automated tools to test the software security.

4.4 User Acceptance testing

Definition:

Once the software has been put through its paces and the configurations meet our organization standards we perform a last round of testing to ensure all is working as intended. System logs are working properly and tracking activity.

Participants:

Product testers, intended users and IT department

Methodology:

The software will be put through a full real time testing in a full day of intended use.

4.5 Batch Testing

4.6 Automated Regression Testing

Definition:

Once developers have time to address bugs and issues we can test the major functions of the software and ensure everything is still working as intended and secure.

Participants:

Product testers, intended users and IT department

Methodology:

We can utilize scripts to run through the software functions and access data as intended and check the configurations and ensure nothing has broken in the process.

5.0 Hardware Requirements

Company computer
The tracking software
Cisco routers
modems
Firewalls

6.0 Environment requirements

6.1 Main Frame

Since we are only going to utilize this software internally we can utilize our office space with segmented networks to ensure that while testing we do not interrupt daily workflows until the software is ready to roll out.

6.2 Workstations

Our organization's workstations should be utilized for testing.

7.0 Test Schedule

Expedited testing can take up at least a month.

A first milestone is testing the first version of the product and addressing whatever bugs arise and having proper configurations in place to ensure the software is locked down from external access. We will need a product tester, a handful of intended daily users and IT staff.

8.0 Control Procedures

Problem reporting

Bugs, issues and problems should be logged in our organizations tracking software and meet organizations report procedures.

Change Request

Like reporting bugs, change requests should be submitted to the head developer and management to determine whether the change is necessary and accept possible changes to both the software.

9.0 Features To Be Tested

Loging(system logs), Tracking, and filters should be best tested and configured.

10.0 Features Not To Be Tested

The overall interface of the software is not a primary focus at this point. We are intended to test functionality primarily.

11.0 Resources/ Roles & Responsibility

Development teams are responsible for the building of the software and address bugs. Production testers and intended daily users will be used to test for the software functionality and report bugs. The IT team will be responsible for configuration and security of the software.

12.0 Schedules

Major Deliverables

13. Significantly Impacted Departments (SIDs)

Semi routers, Testers, IT department,

14.0 Dependencies

Initial testing will be isolated from the entire network to limit potential issues and brought into the full network for final testing.

15.0 Risk/ Assumptions

Given the pressure to roll out this software we should expect longer work hours to ensure completion but displays are a possibility given the complexity of the software. Without proper security configurations it could leave our network vulnerable.

16.0 Tools

Tracking software, Workstations, Bugzilla, security logs

17.0 Approvals