**Scope**

This test plan covers full systems test of the Steg-Sleuth application. This includes user procedures, as well as programs and job control.

The first phase will include all ‘must have’ requirements. These and any other requirements that get include must all be tested. At the end of the first phase, a tester must be able to:

1. Create a manual test with as many steps as necessary
2. Save it
3. Retrieve it and have the ability to view it when running the test
4. Enter results and appropriate comments
5. View results

As the team works with the product, they will define the needs for the second phase.

**Definitions**

This clause contains key terms as they are used in this standard.

**carrier:** The file that will hold the hidden file.

**encryption:** The process of obscuring information to make it undecipherable.

**hash:** A value created from a file passing through a hashing formula. Used to verify file integrity.

**hex-dump:** Hexadecimal view of file data.

**pass/fail criteria:** Decision rules used to determine whether a software item or a software feature passes or fails a test.

**payload:** The file to be hidden.

**POSIX-compliant:** A software that is developed following a family of standards specified by the IEEE Computer Society for maintaining compatibility between operating systems.

**software feature:** A distinguishing characteristic of a software item (performance, portability, or functionality).

**software item:** Source code, object code, job control code, control data, or a collection of these items.

**steganography:** The practice of concealing a file or message within another file. Usually done by changing certain bits in the carrier file to match those of the payload.

**test: (A)** A set of one or more test cases, or **(B)** A set of one or more test procedures, or **(C)** A set of one or more test cases and procedures.

**test case specification:** A document specifying inputs, predicted results, and a set of execution conditions for a test item.

**test design specification:** A document specifying the details of the test approach for a software feature or combination of software features and identifying the associated tests.

**test incident report:** A document reporting on any event that occurs during the testing process which requires investigation.

**testing:** The process of analyzing a software item to detect the differences between existing and required conditions (that is, bugs) and to evaluate the features of the software item.

**test item:** A software item which is an object of testing.

**test log:** A chronological record of relevant details about the execution of tests.

**test plan:** A document describing the scope, approach, resources, and schedule of intended testing activities. It identifies test items, the features to be tested, the testing tasks, who will do each task, and any risks requiring contingency planning.

**test procedure specification:** A document specifying a sequence of actions for the execution of a test.

**test summary report:** A document summarizing testing activities and results. It also contains an evaluation of the corresponding test items.

**UNIX:** A multi-user, multitasking computer operating system.

**References**