Sprint 3 Planning Document

Team YFFS



Team 22

Brian VerVaet, Max Molnar, Sam Spencer, Wyatt Dahlenburg, Zach Kent

**Table of Contents**

* Sprint Overview 2
* Current Sprint Detail 3-7
  + User Story #1 3
  + User Story #2 4
  + User Story #3 5
  + User Story #4 6
  + User Story #5 7

Sprint Overview

We currently have a working file system that has basic capabilities, such as filename obfuscation, directories, and user permissions. Now we would like to focus on improving the security of our filesystem by making the obfuscation more robust. We would like to add different algorithms to encode the entire contents of our files, and we would also like to fine tune our encryption methods so that they are both fast and efficient. We are also planning on finalizing our product within the lens of an embedded device.

**Scrum Master:** Zach Kent

**Meeting Schedule:** Tuesdays, Thursdays, and Sundays at 3:00pm

**Risks/Challenges:** Because we are going to be encrypting large swathes of text, we could run into situations where we lose data. It is also likely that we could run into problems with our embedded device. We know that the basic functionality works, but when we go into more advanced encryption techniques we may lose the system as a whole.

**User Story #1**

* As an admin, I would like to use this file system on embedded devices.

|  |  |  |  |
| --- | --- | --- | --- |
| # | Task Description | Estimated Time | Owner |
| 1 | research requirements for running on embedded systems. | 3 | Brian |
| 2 | Optimize code for use on embedded systems. | 5 | Brian |
| 3 | Reduce dependencies and executable overhead | 2 | Brian |
| 4 | Deploy on multiple embedded systems | 5 | Brian |
| 5 | Debug and test the filesystem | 7 | Brian |

Acceptance Criteria:

In order to allow use on smaller systems, the code should be optimized to decrease the load while it is running.

To increase the speed of filesystem operations, all unnecessary dependencies will be removed and executable size will be decreased to a minimum.

The filesystem must be able to operate in all situation on a Raspberry Pi.

.

**User Story #2**

* As an admin, I would like secure operations on the filesystem.

|  |  |  |  |
| --- | --- | --- | --- |
| # | Task Description | Estimated Time | Owner |
| 1 | Research possible options for different encryption interfaces. | 5 | Sam |
| 2 | Implement at least two encryption algorithms to obfuscate file names and file contents | 10 | Sam |
| 3 | Prevent unauthorized encryption operations | 10 | Sam |
| 4 | Test consistency and uniqueness of encryption operations across multiple use cases | 5 | Sam |

Acceptance Criteria:

All possible encryption options and their implementations should be clearly defined in documentation and man-pages.

Encryption operations should be consistent and have same result during different executions on a string.

Prevent users from seeing content inside of the filesystem while it is not running.

**User Story #3**

* As an admin,I would like to have appropriate decryption operations for the available encryption options.

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | Research possible security flaws in our encryption that would lead to unauthorized decryption | 5 | Max |
| 2 | Implement decryption operations for each of the encryption options. | 10 | Max |
| 3 | Test the encryption/decryption interface over many uses. | 10 | Max |
| 4 | Prevent unauthorized access to information that may lead to unauthorized decryption | 8 | Max |

Acceptance Criteria:

Given any question on decryption, a user should have all of their questions answered by all relevant man pages.

Given any decryption operation, it should be consistent and have the same result during different executions on the same string.

Given a filesystem that is not currently running, malicious attempts to view the data inside should fail.

**User Story #4**

* As an admin, the encryption interface should work quickly.

|  |  |  |  |
| --- | --- | --- | --- |
| # | Task Description | Estimated Time | Owner |
| 1 | Once encryption is completed I will test the speed of encryption and decryption functions | 4 | Wyatt Dahlenburg |
| 2 | Research methods to optimize encryption/decryption speeds | 8 | Wyatt Dahlenburg |
| 3 | Implement optimizations | 6 | Wyatt Dahlenburg |
| 4 | Compare speed of optimizations and explain the tradeoffs if any | 6 | Wyatt Dahlenburg |

Acceptance Criteria:

Given the encryption option, a file’s contents should be encrypted in a reasonable amount of time.

Given the encryption option, the file’s contents should be protected by the algorithm and not be impacted by optimizations.

Given the decryption option, the file’s contents should be decrypted in a reasonable amount of time.

**User Story #5**

* As a admin, I would like the ability to perform a comprehensive test of the current state of the filesystem.

|  |  |  |  |
| --- | --- | --- | --- |
| # | Task Description | Estimated Time | Owner |
| 1 | Set up a routine to demonstrate the state of the filesystem to others | 4 | Zachary Kent |
| 2 | Write script to thoroughly test all of the filesystem’s available tools | 8 | Zachary Kent |
| 3 | Analyze the efficiency of the filesystem when placed under heavy load (considerably large amount of read/writes, encrypt/decrypts, etc.) | 5 | Zachary Kent |
| 4 | Verify the integrity of the filesystem after a reboot, power failure, and other corner cases | 6 | Zachary Kent |

Acceptance Criteria:

Given a filesystem, running a test script should interact with the system in the same way a user would.

Given a filesystem, the test script should properly list any unexpected errors.

Given a filesystem, the test script should display the time necessary to complete tasks, so that we can easily view how it is operating.