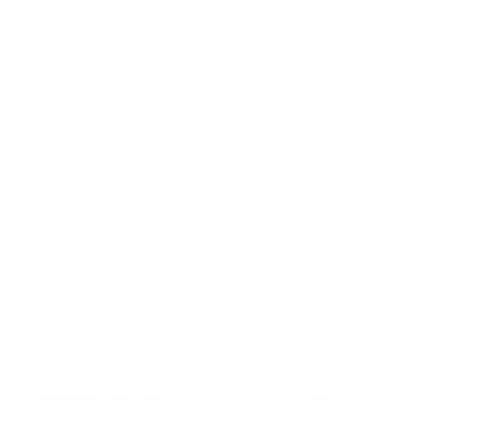
**Lab Development Guidelines for - TWINE**





# Table of Contents:

[Table of Contents: 2](#_Toc147155420)

[CSI Linux Lab Guidelines for - TWINE 3](#_Toc147155421)

[1. Purpose: 3](#_Toc147155422)

[2. Factions: (a.ka. The Criminowls) 3](#_Toc147155423)

[3. Evidence: 3](#_Toc147155424)

[4. Knowledge Domains: 3](#_Toc147155425)

[5. Simulations: 4](#_Toc147155426)

[6. TWINE Documentation Template 7](#_Toc147155427)

[7. Operation QC Checklist: 8](#_Toc147155428)

# CSI Linux Lab Guidelines for - TWINE

## Purpose:

The purpose of this document is to define the criteria for creating content using Twine. Twine is a solution that can be used online and on the desktop; both available at <<https://twinery.org>>.

CSI Linux themed Labs and Simulations are synonymously called Operations; and hereinafter referred to as operations. Operations are intended to support the learning process and.....

## Factions Catalogue:

Factions are groups that are the subject/s of an Operation(Case). In all cases, the SUBJECT are Owls. In some instances, a faction is an analog for a real world group that is of like mind.

Factions will grow over time, however, unless absolutely necessary, all creators are asked to consider existing factions before recommending the creation of a new one. Where necessary, all new factions must be reviewed and approved before publication, by submission in the Operation report.

Affiliations are small groups or individuals that are in association with a larger faction.

Bodies are individuals or groups that participate in the completion of the operation. This includes, but not limited to Law Enforcement agencies, Locations, detectives, etc.

For an up to date list of factions and their associated bodies, see the **CSI Linux – Factions Catalog**; which, for the time, will be provided in the creators package at time of request.

## Evidence:

The evidence for operations should be in the following formats:

* Forensic images of full OS
* Forensic images of a USB
* Functional websites
* Captured website images (e.g. SingleFile, Hunchly, etc...)

## Knowledge Domains:

Knowledge domains are defined as the curated information that is presented by CSI Linux in the training of its participants. These are areas that can be challenged, and which align to training that exists in the Academy at CSI Linux <[csilinux.com](https://csilinux.com/academy)>.

For an up to date list of knowledge domains, see the **CSI - Knowledge Domains**; which, for the time, will be provided in the creators package at time of request.

## Simulations:

The TWINE operations shall align with the template provided in the creator package. This template contains the passages and layout that is required for all operations and is to be used as the starting place in the creation of content for use by CSI Linux.

The template is created in TWINE using an IDE of the creators choosing but must produce an export that is the standard `.twee` file formatted. The following is intended to profide specific guidance that would be lost by only importing the TWEE TEMPLATE into the IDE, and must be adhered to as deviation may delay the release of training content.

* Story **MUST** be created in Twine ( https://twinery.org )
  + Twine (version 2)
  + Story Format: SugarCube v2.35 or greater (Required)



* + - DO NOT USE ANY OTHER FORMAT (e.g. Harlowe, Chapbook, Snowman, etc.)
  + All passage names MUST/SHALL be unique.
  + Use this template to build from:
    - https://csilinux.com/files/stories/Owl\_Simulation\_Template.twee
  + REF:
    - <https://twinery.org/cookbook/html/storyformats/sugarcube.html>
    - <https://www.motoslave.net/sugarcube/2/docs/>
  + Using the CSI Linux Twine Operation Template.
    - This is a `.twee` file and should be obtained new for each operation; as this will be a living template.
* Story **CAN** be:
  + Your own operation.
  + A continuation of an existing/previous operation.
* Story **MUST**:
  + Align with the themes and teachings of CSI.
    - Any deviation from this may result in the creation being heavily modified or refused for publication.
  + Be presented in written form before a creator begins producing TWINE content. (*See Section 6 for Operation Template.*)
    - This is to ensure that the concept can be supported by CSI Leadership/Staff.
    - **Note**: *It is known that portions of the Operation Template will be revised and matured over the creation of the TWINE content. This is expected but will never eliminate the requirement to present Operation writeup before proceeding with its creation*.
* Story **SHALL** consist of:
  + Correct path/s
    - Correct path/s should leave no question as to it being the proper path
    - Building on or explaining the success is encouraged.
  + Wrong path/s
    - Wrong path/s must be explained
  + A definitive end
    - There should be no doubt that they have reached or been redirected to the end.
  + After Action Report (AAR)
    - This is the final screen, regardless of outcome from the operation, that provides feedback to the player on all the areas challenged.
    - For every ‘improper or poor’ choice a variable should trigger or increment, and the AAR will provide an assessment of each.
    - For every ‘correct or proper’ choice a variable should trigger or increment, and the AAR will provide an assessment of each.
* Story **MAY** contain:
  + Rabbit Holes
    - This is a path in the operation/lab where, as the creator, you can take them on a path that leads to a lesson they didn’t need to learn but the choice is real.
    - All rabbit holes must have an exit and a reason.
    - Do not send a participant down a path that they cannot exit.
    - Do not create a rabbit hole that has no learning purpose.
    - Rabbit holes CAN take a player to a completely different outcome for the operation/lab, BUT the player needs to be told that they didn’t follow a proper path in the after action report.
  + Mulligan (similar to rabbit hole)
    - A point in the operation/lab that is not terminal but also doesn’t require a redo/restart.
    - This is a place where a player can be taught a lesson because they selected an option that allows them to reconsider their action.
    - Allow the user to take a new approach, BUT they are taken through what would have gone wrong had they proceeded. Then they are placed back in the operation/lab at the point where they made the decision that put them in this spot.
  + Redo/Restart
    - A point in the operation/lab where the player reaches a point a termination event BUT is less than 25% through operation/lab.
    - This is a choice within the operation/lab that the player can select.
  + Terminations
    - Termination is a point within an operation/lab where there is no way to recover to a good point.
    - If the operation/lab has terminations configured into it there must be a detailed explanation of the terminal event.
    - Never leave a player wondering what was wrong about what they did.
* Story **SHALL** have:
  + An intended lesson.
  + Be very detailed and engaging.
  + All passage names shall be unique.
  + Use this template to build from:

https://csilinux.com/files/stories/Owl\_Simulation\_Template.twee

* + Each page should have a 16x4 image banner.
    - Here are some you to choose from if you do not make your own:
    - <https://www.dropbox.com/scl/fo/9ics39i2rs7gbf130ktoh/h?rlkey=iu81okciwammlllzflmwe9o80&dl=0>
* Coding for the story; within TWINE
  + For items users must type exactly to be correct, use this code:

<<set $answer001 to "">>

<<set $name1 to "something">>

Please input the value.

<<textbox "$answer001" "" autofocus>>

<<button "Verify Name">>

<<if $answer001 is "">>

/% They have not supplied an answer. %/

<<script>>UI.alert("You did not supply a name!");<</script>>

<<else>>

<<set$answer001 to $answer001.trim().toLowerCase()>>

<<if $answer001 is $name1>>

/% Send to the “passagename” passage. %/

<<goto "passagename">>

<<else>>

/% Send to the “wrong\_passage” passage. %/

<<goto "wrong\_passage">>

<</if>>

<</if>>

<</button>>

## TWINE Documentation Template

The following defines the sections of the TWINE documentation template named:

“**Operation – Twine Template**”

This template will be available through a version-controlled path and is to be acquired new each operation to ensure that the latest template is used.

In each instance of use the creator is asked to replace the title in the format that it is presented, and to leave the prompts for all of the remaining sections as an line or item in that section until the document is accepted for publication.

At the time of this document being written the following are the sections of the operations template file. Each section, where applicable, should meet the requirements noted in the “[ ]” square brackets as well as the guidance below:

1. **Title:**
   1. “**Operation –**“ This section of the name is to identify a top-level operation for an operation. (*e.g. Operation Owl Eye -* )
   2. “**Twine Template**” -- This portion of the name is replaced with a descriptive title of the operation. (*e.g., Seizing Evidence, Silent Vigil, etc.*) These examples are for operations that were named: (*see the difference*)
      1. Operation Owl Eye – Seizing Evidence
      2. Operation - Silent Vigil of Starlight Station
2. **Knowledge Domain/s:**
   1. [list of skills areas that are challenged in this presentation]
3. **Bodies:**
   1. [name of all faction/s, agencies, and individual parties incorporated with the operation]
4. **Prologue:**
   1. [the storyline that sets up the operation]
5. **Brief:**
   1. [the agency brief that delivers the mission to the participant]
6. **Notes:**
   1. [notes to be defined as this template is used and notes are understood their intent]
7. **Websites:**
   1. [web resources that are used in the completion of the challenge]
8. **Evidence Files:**
   1. [list of files that are required to be used in the completion of the challenge]
9. **CSI Training:**
   1. [list of CSI specific training that exists to support learning pre/post challenge]
10. **References:**
    1. [any reference that contributes to the creation or completion of the challenge]
11. **Walk-through/s:**
    1. Correct Path: (step-by-step along the path/s.)
    2. Redo/Restart Path/s: (step-by-step along the path/s.)
    3. Rabbit Hole Path/s: (step-by-step along the path/s.)
    4. Termination Path/s: (step-by-step along the path/s.)
12. **After Action Report:**
    1. [explanation of acter action report]
    2. Variables covered:
       1. [list of actions/variables covered in the AAR]
13. **History Presentation Dates**
    1. Proposal
    2. Twine Complete
    3. Quality Control
    4. Published

## Operation QC Checklist:

The Operation Checklist is for quality control review of all created operations.

1. DOCUMANTATION
   1. Operation template completed.
      1. The operation template is the ‘dotx’ file used in the proposal and documentation of an operation/lab and shall be completed for each.
   2. Knowledge Domains supported by CSI training, currently or in progress.
      1. This is a list of the knowledge areas that the operation/lab covers and whether they are covered by CSI training.
   3. All bodies/parties identified.
      1. These are the characters identified in the operation/lab.
   4. All evidence and supporting content documented.
      1. This action is to ensure that all evidence and supporting documentation necessary to complete the operation/lab is contained.
   5. All walk-through documented.
      1. The walk-through is correct and at a minimum the correct path is fully documented.
      2. It is strongly recommended that all paths be documented at least at the 30,000ft view.
   6. All dates provided represent progress to date.
      1. The history of the operations document is correct.
2. TWINE
   1. Used the applicable twee template.
      1. As TWINE use grows the maturity of the template will change as well. It is required that each new operation be created with a template that is newer than the date that the operation was approved to be produced.
      2. Compare template used to the most recent to ensure that required updates are not missed.
   2. All default menu items are correct and working.
      1. In the TWINE template there are standardized menu items that must be present in all operations.
   3. All default variables initialized in the StoryInit passage.
      1. Each variable used throughout the operation must be initiated in the StoryInit passage.
3. OPERABILITY
   1. All operation variables initialized in the StoryInit passage.
      1. All variables need to be initiated in the StoryInit for practical reasons, but also to ensure that
   2. All correct paths tested and operational.
      1. All paths must be fully tested for correctness and expected outcome.
   3. All improper paths tested and operational.
      1. All paths must be fully tested for correctness and expected outcome.
   4. All rabbit hole paths tested and operational.
      1. All paths must be fully tested for correctness and expected outcome.
   5. All termination paths tested and operational.
      1. All paths must be fully tested for correctness and expected outcome.
   6. All evidence and supporting content are operational.
      1. All evidence must be tested for applicability to the operation/lab.
   7. After Action Report provides proper output dependent on player’s path.
      1. Each after action item in the after action report must accurately communicate the reason for the outcome across the entirety of the operation/lab.

**Thank you for helping grow CSI Linux**

