# SOLVE-IT Exercise: Tool testing dataset creation

## Overview for instructors

| | |
|---|---|
| Version | 0.1 DRAFT |
| Author | Chris Hargreaves |
| Prerequisites | ● An introduction to the SOLVE-IT knowledge base<br>● Instruction on methods for dataset creation<br>● Instruction on tool testing methods and best practise |
| Intended Learning Outcomes (ILOs) | At the end of this exercise you will be able to:<br>● Create error focused datasets based on weaknesses in the SOLVE-IT knowledge base<br>● Evaluate a tool or process using that dataset |
| Instructor notes | This exercise is designed to encourage students to think about how tools can go wrong, and how we can capture that in a dataset, and use that dataset to test tools.<br><br>The exercise is worded that the students can either have free choice over the technique tested or you can select one for them.<br><br>Not all techniques are populated so you may need to provide content updates for SOLVE-IT if you want to use a currently incomplete technique for the exercise.<br><br>Some example currently suitable populated techniques are:<br><br>T1059: Identify partitions<br>T1060: Process file system structures<br>T1063: Identify file types<br>T1064: File carving<br>T1069: Browser examination<br>T1072: Chat app examination<br><br>The second part involves using the created dataset to test tools. Again, this can be left to the student or you can specify. |

# SOLVE-IT Exercise: Tool testing dataset creation

## 1. Aim

The aim of this exercise is for you to create datasets that focus on testing tool functionality and correct processing of data to present correct results.

## 2. Overview of the Tasks

This exercise involves several steps:

1. Technique selection
2. Weakness selection
3. Dataset generation
4. Tool testing

### 2.1 Technique Selection

Select a technique from SOLVE-IT that you wish to use as the basis of the dataset creation and testing. Your instructor may specify a particular technique to use, for example *T1072: Chat app examination*.

Otherwise to help with your selection, within the SOLVE-IT Excel spreadsheet, the worksheet “Techniques” has a column to indicate the number of weaknesses indexed for each technique.

### 2.2 Weakness(es) Selection

Once a technique is selected, you can review the weaknesses listed. The aim is to select one or more weaknesses for which a dataset can be created that would test if a weakness is present or not in a particular tool.

For example, within *T1072: Chat app examination*, you could select the weakness *W1102: Failure to display that a message had a previous state*. This would then form the basis of the dataset that needs to be generated.

Depending on your instructor, you may be free to propose a weakness that is not listed for testing. If this is the case, look at the ‘Propose New Weakness STANDARD’ template in the SOLVE-IT issue tracker[1] for guidance. This can also be submitted for review and inclusion in the main SOLVE-IT repository.

[1] https://github.com/SOLVE-IT-DF/solve-it/issues

## 2.3 Dataset Generation

Using the weaknesses identified above, you now need to create a dataset that allows this weakness to be tested for.

This could be a disk image, set of files, or a phone extraction. It could be created manually, or programmatically.

You will also need to identify the specific technology that you are testing. For example *T1060: Process file system structures*, you might choose FAT, NTFS, APFS or others for your dataset generation. The operating system you use to create the data may also be relevant due to file system driver implementation differences. Alternatively, for *T1072: Chat app examination*, you would need to select a specific chat app, on a specific operating system, and select specific versions of those pieces of software to test.

You should carefully document the ground truth of the dataset you create. Ground truth includes both:

- The set of actions carried out to create the data along with details such as times of actions
- The expected results from a tool processing your dataset (focused on the potential weakness you are testing)

If you are looking at multiple weaknesses you might decide to capture multiple weaknesses in a single dataset, or provide separate data for each weakness.

## 2.4 Tool Testing

Once this dataset is created you can now use it to test digital forensic tools. You will need to note the tools, and the specific version, and on what platform they were run.

You should already have your expected results recorded from the previous tasks. Using your forensic tools of choice, you will now need to process the data you generated, extract the actual results from the tool, and compare the two.

You should report on how the tool performs in terms of producing results that align with your expected results.

If they do not match, an extension exercise is to examine the raw data more closely and see if you can determine in what manner, and why, the results are not correct.

# 3. Related Reading

Brunty, J., 2023. Validation of forensic tools and methods: A primer for the digital forensics examiner. *Wiley Interdisciplinary Reviews: Forensic Science*, *5*(2), p.e1474.

Hargreaves, C., Nelson, A. and Casey, E., 2024. An abstract model for digital forensic analysis tools-a foundation for systematic error mitigation analysis. *Forensic Science International: Digital Investigation*, *48*, p.301679.

Lyle, J., 2002. Testing disk imaging tools. *DFRWS 2002 USA, https://dfrws.org/wp-content/uploads/2019/06/2002_USA_paper-testing_disk_imaging_tools.pdf*