



# DASISH Web Annotator (DWAN)

# Test Plan for Security

## Test Plan Identifier

DWAN release 1.0 Test Plan for Security version 0.9 (draft)

## Introduction

Security testing verifies whether a system under test allows only designated users and processes access to business functionality and data.

## Features and Functions to Test

SE1 - High
Input data received from the client must be parsed to make sure that it does not contain "out of bounds" or “buffer overflow” input data (e.g. a client enters a value of 13 or 99999999999999999999999999 for a month).

SE2 - High
Check that the user is locked out for 30 minutes after 3 failed attempts to log in.

SE3 - High
Check that the user is logged out after 30 minutes of inactivity.

SE4 - High
Check that disabling/enabling any client-side setting (e.g. JavaScript) does not allow the client to circumvent any security measure.

SE5 - High
Check that using any combination of bookmark/favorite, back/forward, history or go navigational jump does not allow the client to circumvent any security measure.

SE6 - High
Check that all sensitive data is encrypted during transmission, this includes cookies (session or persistent), hidden tags on an HTML form or via “long URLs” (e.g. using the HTTP Get command).

SE7 - High
When user annotates private or secured content the copy of the Web page (cached representation) must not be available for public.

## Features and Functions not to Test

This test plan will not include any security tests designed to ensure that the DWAN server infrastructure (as opposed to the client-side Plug-In) is protected from attacks by hackers (internal or external).

## Test Deliverables

The following documents will be generated as a result of these testing activities:

- Test plan for Security (this document)
- Test log for each testing effort
- Automated test scripts and supporting test data

Under normal testing conditions, the incident reports would be produced in GitHub.
With the exception of the automated test scripts, all documents will be delivered as PDF documents.

## Test Environment

Testing is performed on the client side with operating system Windows 7, Windows 8, Mac OS X or Linux. For testing of the browser plugin the latest Mozilla Firefox version (29 or later) is used. For collecting data about HTTP requests and responses the LoadUIWeb 2.99 is used. For the testing of the server API the Python programming environment with the unit testing framework and the package Requests 2.3.0 (https://pypi.python.org/pypi/requests/) is used.