Secure Ideas®
professionally evil

FINAL SUMMARY REPORT

Web Application Penetration Test of Canvas

Prepared for: Instructure
November 27, 2013

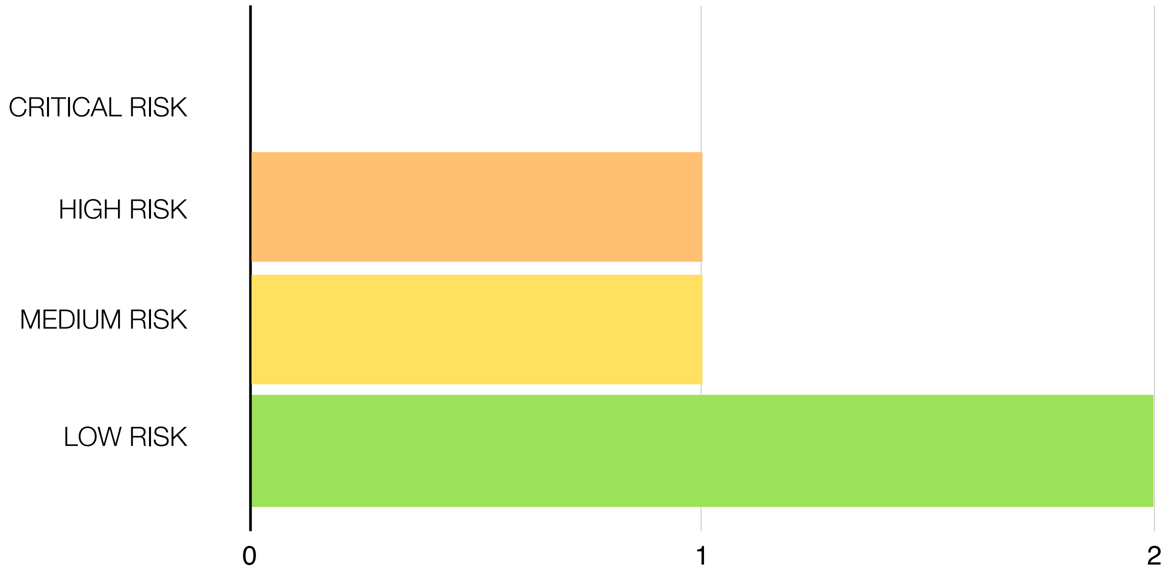
EXECUTIVE SUMMARY

Overview

Secure Ideas performed a Web Application Penetration Test of Instructure's open source "Canvas" application during November of 2013. The scope of the testing was constrained to a set of URLs where Instructure had configured a test instance of Canvas specifically for this purpose.

Secure Ideas found that with a few minor exceptions the quality and coverage of security controls in the Canvas application were very solid. The Finding Severity Profile (see Figure 1 below) compared to a similar 2012 assessment reaffirms a continued effort towards improvement in the security of the application and coding practices over previous years.

Finding Severity Profile



Procedure

In conducting this test, Secure Ideas used a combination of automated and manual methods to attempt to circumvent the existing controls. In addition, since Canvas is an open source application, Secure Ideas obtained the current source code from GitHub. This code was then run through the Brakeman static analysis engine (see http://brakemanscanner.org/) and also reviewed manually for specific types of flaws commonly found in Ruby/Rails applications. The results were evaluated against the live application. Similar steps where followed for the Flash (i.e. .swf files) components of the application. SWFScan was used to decompile and analyze those.

The authentication mechanism of Canvas is well written but could be improved. Secure Ideas found the login page to be susceptible to username harvesting. Secure Ideas observed that an account lockout mechanism was in place and would hinder an adversary from attempting dictionary attacks on a password once usernames are harvested. Secure Ideas also observed that email addresses where used for usernames. When email addresses are used for this purpose, harvested usernames may be used in targeted phishing attacks.

Secure Ideas tested for injection flaws which result in vulnerabilities such as Cross-Site Scripting (XSS), SQL Injection, command injection, etc. This was done through both automated scanning scripts as well as targeted manual testing based on the source code analysis results and experience. Secure Ideas found that user-provided input was consistently validated or encoded across most of the application, rendering it quite resilient to XSS vulnerabilities. One exception to this was found with html file attachments, which has also been reported in previous years as an ”Arbitrary File Upload” vulnerability. Database communication was found to consistently leverage parameterized queries, eliminating the risk of SQL Injection.

Secure Ideas found that the Canvas application consistently and effectively leverages tokens to prevent replay of requests, effectively preventing Cross Site Request Forgery (CSRF) attacks. In addition, Canvas strictly follows session management best practices. These combined measures rendered a variety of attack attempts such as privilege escalation and business logic unproductive. However, Secure Ideas found the session timeout seemed rather long for administrators, and in one case it was observed to be valid after several hours of inactivity. Secure Ideas recommends a shorter session timeout for administrator accounts.

The table on the following page summarizes these findings.

FINDINGS SUMMARY TABLE

Ref#	Severity	Finding	Summary	Status
1	**High**	Persistent Cross-Site Scripting	**Description:** User-supplied input that is stored on the system and rendered in another user's browser at a later time may contain malicious executable scripts.	Open, Risk accepted under 2012 finding "Arbitrary File Upload"
			Recommendation: Use output encoding and/or input validation to prevent embedded scripts from being executed and/or control the output so that it is downloaded instead of rendered in the browser.	
2	**Medium**	Username Harvesting	**Description:** Username harvesting is a flaw that allows an attacker to verify that a username is valid and in use within the system. This is caused by the system reacting somehow differently for a valid user name then for an invalid user name.	Remediated as of December 2013
			Recommendation: Modify application code to return the same response regardless of whether or not the user name exists within the authentication system.	
3	**Low**	Password Field with AUTOCOMPLETE Enabled	**Description:** The Canvas application does not explicitly disable the autocomplete feature of the client browsers to store password information.	Open, Risk Accepted under 2012 finding by same name
			Recommendation: Modify application code include the "AUTOCOMPLETE=OFF" setting for the password field.	
4	**Low**	Session Expiration on Administrative Functionality	**Description:** The session does not expire within a short time while authenticated in an administrative role.	Open
			Recommendation: Administrative sessions should be set to expire after 15 minutes of inactivity.	

FINDING CLASSIFICATIONS

Each finding is classified as a High, Medium, or Low risk based on Secure Ideas’ considerations of potential threats, the likelihood of attack, and the possible impact of a successful attack against Instructure’s “Canvas” application. Each of these factors is assessed individually and in combination to determine the overall risk designation. These assessments are based on Secure Ideas’ professional judgment and experience providing consulting services to enterprises across the country. This report outlines the findings Secure Ideas collected from the testing, as well as Secure Ideas’ recommendations that will assist Instructure in reducing its risks and helping remove the vulnerabilities found.