**Project Title:** Creating a secure testing environment using the cloud

**Team Members:** Adam Sturchio, Lisa Blackerby, Sam Riley (In collaboration with Warnitris May and Ashley Mains)

**Date:** 2 February 2019; 17 February 2019

**Objective:**

Our goal is to automate the process of creating a virtual machine in the cloud in order to create a secure testing environment for web applications. The process of sending the web application to the cloud should be automated as well.

**Current Practices:**

There are many services that currently exist that provide web application testing as apart of their wide array of services.

**#1) Resourcology**

Resourcology is a full-cycle software development and testing company with expertise in mobile app testing, security testing, and software application testing. Their robust methodology ensures the application is error-free. They have successfully tested applications for Fortune 500 companies as well as SMB’s. They provide companies with a free trial to experience the difference in testing solutions and quality assessment.[2]

**Visit the website:**[**Resourcology.com**](https://www.resourcology.com/qasolutions.php)

**#2) CresTech**

Testing Services for all needs – From Startups to Fortune 500 enterprises. Testing solutions provided include – product quality, application performance, test optimization and transformation, test automation, application security, and mobile testing.[2]

**Visit the website:** **[CresTech Software Systems](http://crestechglobal.com/" \t "_blank)**

**#3) TESTRIQ QA Lab**

An independent Software Test Consulting and Services company. Powered by a workforce of 30 proficient QA professionals and average 12+ years of experience in the field of software testing, they aim to provide a complete range of software testing services, while keeping the customer satisfaction at its core.[2]

**Visit the website:** [**TESTRIQ.com**](https://www.testriq.com/)

While this list is not comprehensive, a common characteristic of all these services is that they are built for local testing, none of these companies do secure testing in a cloud environment. This sets our service apart allowing for the versatility of cloud computing in our testing environment.

**Proposed approach:**

By using a virtual machine created through automation with a cloud database and web servers, our Java tool will try to successfully perform a series of SQL Injections and browser automation tests into a web application using batch scripting, the Selenium Webdriver, and Selenium IDE. While this isn’t the limit of our services ability, it will demonstrate how a testing environment is created and actual tests are executed.

Due to the scope and complexity of penetration testing, our main goal for half of this project is to test the requirements needed for Defense 1. By using SQL injection attacks that target if a system is using proper prepared statements, small companies can utilize this tool to make sure their developers are writing their database queries properly.

By using Selenium IDE we can display the robustness of automation testing on a cloud platform by expanding into the realm of input validation, since the IDE does not require advanced coding knowledge to build and execute test scripts.

**Stakeholders/Beneficiaries:**

Any business or organization that needs to test software ever. Additionally, by providing automated validation tests via web browser, we can showcase the benefits of automation testing on a cloud platform to Software Quality Assurance Teams.

**Impacts and Dissemination**

If implemented properly, this should be able to replace most of a software testing team. Only one member of that team should be needed to implement the test and provide us with all necessary information.

**Costs & Duration**

|  |  |
| --- | --- |
| Timeframe | Activity |
| February 18 | Proposal |
| February 18 - March 25 | Literature Survey, Research Work, and Building Two Customer Web Applications |
| March 25 - March 27 | Progress Report Presentation for Cloud Computing |
| March 27 - April 28 | Implementation Testing |
| April 29 | Presentation |

**Risks and Payoffs**

The risks of this project are a matter of redundancy and complexity. As mentioned before, many services already permeate the industry. If we are successful, it wouldn’t take a long time for them to implement a cloud testing environment as well.

However, the payoffs would be HUGE! This would allow companies with an extensive QA team to allocate those resources to create a better experience for the consumer.

**Measure of Progress**

By March 27th, an automated virtual testing environment should be ready to place on the cloud for automation testing of Defense #1 and basic Quality Assurance testing. By April 15th, automation of Defense #1 should be at the end of the test & measure phase. By April 25th, a presentation should be completed demonstrating automation of Defense #1.

**References:**

[1] SQL Injection Prevention Cheat Sheet. (2018, February 06). Retrieved October 15, 2018, from https://www.owasp.org/index.php/SQL\_Injection\_Prevention\_Cheat\_Sheet#Defense\_Option\_1:\_Prepared\_Statements\_.28with\_Parameterized\_Queries.29

[2] SoftwareTestingHelp. (2019, February 13th). Top 20 Software Testing companies - 2019. Retrieved from https://www.softwaretestinghelp.com/software-testing-service-providers/

[3] Abu-Dabaseh, F., & Alshammari, E. (2018). AUTOMATED PENETRATION TESTING: AN OVERVIEW. Retrieved October 15, 2018, from <http://www.academia.edu/36703007/AUTOMATED_PENETRATION_TESTING_AN_OVERVIEW>. Copyright: CS&IT-CSCP 2018

[4]Stock, A., Gonçalves, I. R., & Correa, J. (2018, August 25). OWASP Top Ten Cheat Sheet. Retrieved October 18, 2018, from <https://www.owasp.org/index.php/OWASP_Top_Ten_Cheat_Sheet#A1_Injection>

**References for Web Application Development and Selenium IDE Scripting:**

[5]“Django Tutorial: The Local Library Website.” *MDN Web Docs*, developer.mozilla.org/en-US/docs/Learn/Server-side/Django/Tutorial\_local\_library\_website.

[6] “Setting up a Django Development Environment.” *MDN Web Docs*, developer.mozilla.org/en-US/docs/Learn/Server-side/Django/development\_environment.

[7] “Django Web Framework (Python).” *MDN Web Docs*, developer.mozilla.org/en-US/docs/Learn/Server-side/Django.

[8] Winter, Mark, and Mark Winter. “Creating Websites Using React and Django REST Framework.” *Hacker Noon*, Hacker Noon, 25 July 2017, hackernoon.com/creating-websites-using-react-and-django-rest-framework-b14c066087c7.

[9] Jordan.irabor. “Build a To-Do Application Using Django and React.” *Scotch*, 19 Feb. 2019, scotch.io/tutorials/build-a-to-do-application-using-django-and-react.

[10] “Getting Started · Selenium IDE.” *Selenium Blog Posts*, [www.seleniumhq.org/selenium-ide/docs/en/introduction/getting-started/](http://www.seleniumhq.org/selenium-ide/docs/en/introduction/getting-started/).

[11] Haeffner Dave, "The New Selenium IDE: See It Live in Action! ", 20 Feb 2019, <https://applauncher.gotowebinar.com/#webinarOver/5748903595052620034/en_US>

[12] Mitchel, Justin. 30 Days of Python | Unlock your Python Potential. Copyediting: All Levels, Udemy. 26 Dec 2018- 05 May 2019. <https://www.udemy.com/30-days-of-python/learn/v4/overview>.

[13] Nauvalazhar. “Nauvalazhar/My-Login.” GitHub, 11 Nov. 2018, github.com/nauvalazhar/my-login.

[14] Kozubek-Krycuń, Agnieszka. “How to Store Authentication Data in a Database. Part 1.” Vertabelo, 3 Mar. 2016, www.vertabelo.com/blog/technical-articles/how-to-store-authentication-data-in-a-database-part-1.