# Tools for Security Senario Setup

Submitted by

**Sun Mingyang**


In partial fulfilment of the

z requirements for the Degree of

Bachelor of Engineering (Computer Engineering)

National University of Singapore

B.Eng. Dissertation

# Tools for Security Senario Setup

Orchestration Tool for IT Network Traffic Generation
and Human Behavior Simulation in Web Browsing

By

Sun Mingyang

Department of Computer Science
School of Computing
National University of Singapore

2018/2019

Project ID: H0041300

Project Supervisor: Assoc Prof Change Ee-chien

Deliverables:

- Report: 1 Volume
- Manual: 2 Volumes
- Program: 1 Git Repository

**Abstract**

Normally, during Cyber Defence Exercise (CDX), it would be relatively easy to perform an attack or deploy an exploit once the vulnerability is known, however, setting up an appropriate security senario or target system will be rather difficult. In a particular example, during a Cyber Defence Exercise targeting a virtual enterprise-scale network, the attacking traffic performed by the Red Team (who will conduct the attacks) may be too obvious to be spotted or identified by the Blue Team (who will defense the network) when the attacks are performed in the an 'empty' virtual network where there is no background noise or "normal" web traffic.

The goal of this project is to build tools that could help to simulate the real-world scenario and generate normal and benign network traffics in the virtual environment during the CDX similar to what has been described above.

This project will consist of two compoents, while the first one is to build an orchestration tool for generating a sufficiently large amount of network traffic in a distributed system senario, and the second one is to simulate human behaviors in web browsing to make the web traffic more realistic.

The evaluation and demonstration of this project will be conducted in the virtual network provided by the National Cyber-Security R&D Lab (NCL) and the final product may have international impacts on the CDX held by KYPO Cyber Range in Czech Republic.

**Subject Descriptor:**

C.2.4: Distributed Systems

D.2.2: Design Tools and Techniques

D.2.3: Coding Tools and Techniques

D.2.13: Reusable Software

G.3: Probability and Statistics

**Keywords:**

Cyber Defence Exercise Setup, Orchestraction Framework, IT Network Traffic Generation, Network Security, Distributed System, Human Behavior Simulation

**Implementation Software:**

Ubuntu Linux 18.04 Bionic Beaver, Python 3.6, Flask 1.0, Flask-Marshmallow 0.10, Flask-SQLAlchemy 2.3, Marshmallow-SQLAlchemy 0.16, Selenium 3.141, SQLAlchemy 1.3, osBrain 0.6, APScheduler 3.5, ntlk 3.4, React 16.6+, Ant Design 3.10+

# Acknowledgements

I wish to express my sincere thanks to Chang Ee-Chien, Associate Professor from School of Computing, National University of Singapore, for providing me with all the necessary information and facilities for this project.

I place on record, my sincere thank you to Dr. Jan Vykopal from National Cyber-Security R&D Lab (NCL), for the continuous help. I am extremely thankful and indebted to him for sharing expertise, and sincere and valuable guidance and encouragement extended to me.

I am also grateful to Ivo Nutar from the Institute of Computer Science, Masaryk University, Brno, Czech Republic, for sharing with me lots of relevant work and past experience while I am developing the project.

Finally, I will also take this opportunity to express gratitude to all of the members in NCL who everlent their hands to me along this project for their help and support.

# Contents

# Chapter 1

# Introduction

## 1.1 Background

Practical Cyber Defence Exercise (CDX) creates virtual network environment and helps participants experience and understand how attacks against the infrastructures in their organizations are performed. The Blue Team (i.e. defenders) in CDX are normally supposed to come up with countermeasures and deploy some defending actions against a large variety of attacks that will be performed by the Red Team (i.e. attackers), as well as to monitor the context or the environment, a virtual network in most of the case, for any suspicious traffics or actions, or to recover from some successful exploits.

Since the CDX should simulate the real operational network environment as much as possible, instead of attacking traffics, there should be plenty of benign and normal web traffic existing during the CDX as well. In such case, the quality and accuracy of benign traffic has a significant impact on the learning experience and the effectiveness of the whole security exercise because usally we do not want the attacks to be the only traffics in the virtual network. Therefore, ensuring that the quality and volume of background traffic is able to camouflage the attacking traffics from any automated intrusion detection systems will be an very essential goal to set up the CDX virtual environment, or otherwise, it will be too trivial for the Blue Team to detect intrusive traffics.

To achieve the goal above, a large amount of background traffic that simulates actions per-

formed by normal users or actual humans should be guaranteed. Many current practice of simulation of normal traffics in CDX may engage human beings or hard-coded traffic generators, for example, to use dedicated scripts or softwares on every individual machine, which are neither scalable nor efficient, especially when there is a dramatic system requirement change. Besides, as the scales of many CDX are enlarging quickly, more operators or clients will be needed to produce enough realistic network traffic. As a result, automating the process of generating sufficient amount of high-quality benign traffic according to a reasonable amount of configurations is becoming neccessary. And this is also the main goal of this project.

To fulfill the requirements illustrated above, we are going to propose a brand-new framework to simulate abd generate normal traffic during CDX, based on some state-of-the art works on automotive traffic generating and some studies on real human behavior. This framework will involve multiple client nodes and a controller node. Each of the clients will play a role of a fictitious user to simulate a normal network user who is performing regular actions such as web browsing and sending emails, while the controller node will play a role of an orchestrator, who is going to instruct all the clients based on some configurations, as well as to monitor system status on each client machine.

As for the types of traffic that will be generated by this framework, we will primarily focus on email and web traffic in this project. This is because these two kinds are of the most common activities that the internet users will have, while the portion of web traffic in the whole internet traffic mix has became dominant (Gebert et al., 2012). But we will also open the options and possibilities by design and implementing a highly extensible framework for the future for more types of traffic. Furthermore, since there is a much larger degree of randomness in web browsing behaviors of humans, we will also design and implement an algorithm or a mathematical model to predict or reproduce the realistic human web browsing traffic.

In this report, chapter 2 and chapter 3 will mainly focus on the scope of the design of orchestration tool, while chapter 4 and chapter 5 will mainly talk the methods that will be used to simulate human behaviors. Last but not least, chapter 6 will address on the actual implementation.

## 1.2 Illustration of actions to be done by fictitious users in CDX

Based on the background we illustrated before, here, we are going to elaborate a more specific scenario where a ”fictitious” user or a client is created and what it is supposed to do in order to generate normal traffic during a CDX. The fictitious users will reside in a virtual network created during CDX, and there will be numerous of them existing simultaneously to ensure the traffic volume. The actions specified below describes the actions needed to be performed by a certain client during CDX, and these actions will be executed on a routine base (i.e. repeat every day or every few hours) and will stop until the CDX ends.

| Client LAN IP: 10.0.26.4 | | | Browser: Chrome | |
|---|---|---|---|---|
| **Time** | **Actions** | **Data** | **Destination application/service** | **CDX actions** |
| 8:00 to 9:00 | Browsing news sites | N/A | www.bbc.com www.cnn.com | Red Team will disable DNS in the virtual network. |
| 9:00 to 10:00 | Send emails to a list of email address | email account credentials | admin1@ncl.org admin2@ncl.org | Red Team performs MITM attack and hijack the email traffic |
| 10:00 to 11:00 | Check whether some internal web services is accessible (if not, send email to admin) | email account credentials | www.nclweb1.org www.nclweb2.org | Some web services in Blue Team virtual network will be under DoS attack by Red Team. |
| 11:00 to 11:30 | Check external websites | N/A | www.google.com | The website will be defaced by the Red Team. |

## 1.3 The need of an automatic traffic generator to simulate human behaviours in CDX

Since most of the actions specified in the previous section are actually achieved by actual humans during a CDX nowadays. These tasks are nearly impossible to be directly repaced by automation scripts or softwares since they include lots of randomness in terms of human behaviors. However, this method will become really expensive when the scale of CDX becomes larger. Thus, we would like to propose a traffic simulator which could not only orchestrate a large scale distributed system, but also be able to include modules that could simulate human behaviors.

In the context of the table above, our proposed framework will help to schedule and execute all the entries appeared in the ”Actions” coloumns, where the users of our framework (e.g. CDX cordinators) will only need to provide the information appeared in ”Time”, ”Data” and ”Destination” coloumns.

## 1.4 System design requirements

In this part, we would like to define all major requirements that our proposed traffic generating orchestration framework and the spsecific math model for human behavior simulation should satisfy.

### 1.4.1 Requirements for Traffic Generating Orchestration Tool

1. **Modular architecture**: Modular software architecture should be achieved to enable extensibility and efficient development of individual components in parallel.

2. **Distributed system with controller**: There should be a controller acting as an orchestrator to instruct a number of clients that acts as ficticious users distributed at various hosts in virtual environment, and the controller and the clients should be able to communicate seamlessly.

3. **Real clients and tools are used**: Real clients and tools are used for generating realistic traffic. In other words, forged traffic or crafted packets should not be injected to network,

but interactive sessions between clients and web services are established.

4. **Extensibility of protocols is supported**: Framework can be extended to generate traffic of various network and application protocols. For this project, we will only involve protocols that are related to web browsing (e.g. HTTP/HTTPS) and email (e.g. POP3/IMAP) first. But we will also ensure the high extensibility for the framework.

5. **Client-side cross-platform supported**: Client side framework should be deployable at least at Linux and Windows operating systems.

6. **High-level script of actions supported**: Traffic will be generated according to a human-readable high-level script of actions (which will be explained in more details in a later section) provided to the controller node by an human operator beforehand. The operator is able to specify fictitious users with their routine or scheduled actions, but not network packets nor flows. For example, valid actions may refer to browsing the news website, sending an email to X, or "check web service Y, if not accessible, send email to admin, and if accessible, go to ther service and upload file Z".

7. **Scheduled/repeatable/conditional actions**: Actions defined in the high-level script should be able to be executed either according to a scheduled timeline (repeatable/non-repeatable) or certain conditions specified by the operator.

8. **On-demand actions**: Traffic can be also generated on-demand during the execution of the scheduled script. Framework provides a simple way for the operator how to instruct all clients from the controller.

9. **Graphical user interface**: The framework should provided a graphical user interface for the controller, enabling the operator to instruct and monitor the clients status and traffic status in real time with minimum effort.

10. **Dynamic plan for taffic generation**: The framework should give some flexibility for the operators to dynamically change the traffic generation plans in the middle of generation stage.

### 1.4.2 Requirements for Human Web-browsing Behavior Simulation Model

1. **Continuous generation of traffic**: The model designed should be able to continously generate web traffic without any interruptions unless specified or instructed.

2. **Simulation of human behavior**: The model should be able to simulate how a real human being browses web pages to a certain degree.

9. 10. The framework should be able to produce seemingly random traffic to better simulate real users and their behaviors in cyberspace. This requirement can be either achieved by the framework itself, i.e. by defining new actions, or achieved by our mathematical models, i.e. by allowing certain randomness when predicting human behaviors.

# Chapter 2

# Related Work: Traffic Generating Orchestration Tool

# Chapter 3

# System Design: Traffic Generating Orchestration Tool - Autraff

# Chapter 4

# Literature Review: Human Web Browsing Simulation

# Chapter 5

# Mathmatical Modeling: Human Web Browsing Simulation

# Chapter 6

# Implementation

# Chapter 7

# Testing and Validation

# Chapter 8

# Conclusions

## 8.1 Smmary

## 8.2 Limitation

## 8.3 Future work

# Bibliography

Gebert, S., Pries, R., Schlosser, D., and Heck, K. (2012). Internet access traffic measurement and analysis. In Pescap, A., Salgarelli, L., and Dimitropoulos, X., editors, *Traffic Monitoring and Analysis*, pages 29–42.