POSTER: Towards Exposing Internet of Things: A Roadmap

Vinay Sachidananda
iTrust, Singapore University of Technology and Design
Singapore 487372
sachidananda@sutd.edu.sg

Jinghui Toh
iTrust, Singapore University of Technology and Design
Singapore 487372
jinghui_toh@sutd.edu.sg

Shachar Siboni
CSRC, Ben-Gurion University of the Negev
Beer-Sheva, 84105, Israel
sibonish@post.bgu.ac.il

Asaf Shabtai
CSRC, Ben-Gurion University of the Negev
Beer-Sheva, 84105, Israel
shabtaia@post.bgu.ac.il

Yuval Elovici
iTrust, Singapore University of Technology and Design
Singapore 487372
yuval_elovici@sutd.edu.sg

ABSTRACT

Considering the exponential increase of Internet of Things (IoT) devices there is also unforeseen vulnerabilities associated with these IoT devices. One of the major problems in the IoT is the security testing and analysis due to the heterogeneous nature of deployments. Currently, there is no mechanism that performs security testing for IoT devices in different contexts. In addition, there is a missing framework to be able to adapt and tune accordingly with various security testing perspectives. In this paper, we propose an innovative security testbed targeted at IoT devices and also briefly introduce Adaptable and Tunable Framework (ATF) for testing IoT devices.

1. INTRODUCTION

Currently, there are several types of IoT devices available in the market each with different capabilities. Today IoT devices are chosen based on their specs and price alone. Security does not play a role since there is no clear way to estimate it. Apparently, security is a major challenge of IoT. Since IoT devices will have: a) an internet connection, implying that a hacker can get access to the device and b) a connection to the physical devices. Shodan [1] the IoT search engine shows the dark side of the connected IoT devices. Many devices ranging from cameras to industrial controllers that are connected to the internet could be easily manipulated [2]. This indicates that IoT devices are very easily prone to attacks and it is trivial to consider security measures for IoT devices. In that respect there is no common security standard for all IoT devices. There is a need for dedicated testbeds for performing security testing and analysis on IoT [3].

In this paper, we are proposing a fully functional IoT testbed for security analysis and we also briefly introduce ATF: Adaptable and Tunable Framework for security testing of various IoT devices. We provide an overview of the design setup of the testbed, brief overview of ATF and some use cases that has been accomplished in the testbed.

CCS'16 October 24-28, 2016, Vienna, Austria

ACM ISBN 978-1-4503-4139-4/16/10.

DOI: http://dx.doi.org/10.1145/2976749.2989046

2. THE SECURE IOT TESTBED

The security testbed for IoT devices has been designed and setup in the iTrust lab of Singapore University of Technology and Design, Singapore. The IoT testbed is where various IoT devices such as smart home devices, smart wearables, etc. and Wireless Sensor Networks (WSNs) are tested against a set of security requirements, with predefined test cases and provide the test results. The IoT testbed emulates the different type of testing environments and accordingly stimulates different types of sensors (GPS, movement, Wi-Fi, etc.) and performs predefined and customized security tests. The testbed also collects data while performing the security test for conducting security forensic analysis. Furthermore, the report is provided with the type of IoT device, connectivity, communication protocols supported, security test cases executed and their status i.e. PASS or FAIL.

2.1 The Testbed Setup

The testbed consists of three machines to run and support the tests. The IoT devices, measurement tools and the shielded room are part of the setup. The shielded room is as shown in the Figure 1. The communication capabilities of the testbed can be via WIFI, Bluetooth or Zigbee. We have setup an access point within the shielded room, so that all the IoT devices can connect to the internet and there can be no interference with any signals outside the shielded room. The server has been setup to store any test results, reports, conducts and maintain any project details.

The three machines are as follows: (1) Orchestrating Machine (OM): is located outside the shielded room. The OM runs National Instruments (NI) TestStand [4] which acts as an orchestrator to run and report the tests. (2) Con-

Figure 1: Shielded Room Setup in the iTrust Lab.

trol and Communication Machine (CCM): is located within the shielded room, which controls and connects the measurement tools and any IoT devices. The CCM runs NI LabVIEW [5], the IoT devices are connected to CCM for purpose such as turning ON/OFF, power control, measure power consumption etc. of IoT devices. (3) Analysis Machine (AM): is also located inside the shielded room. The purpose of the AM is to run various testing tools to support any test cases. All the three machines are interconnected and can speak to each other. For any test cases, the OM starts the test with the sequence being written in NI TestStand, which then sends the commands to CCM for further process of the test. Once CCM has the control and after the tasks are accomplished the required information is sent over SSH to AM for conducting the tests using required set of tools. Finally, the results are pulled out by OM via FTP to display the results on the OM. The results are the report, giving the details of the test.

3. ATF: FRAMEWORK FOR TESTING IOT

Based on our IoT testbed setup we are working towards building an adaptable and tunable framework to test any kind of IoT devices automatically with less human intervention. The framework will be adaptable to any kind of testing environment, IoT devices and communication. On the other hand, the framework will be tunable to test the IoT devices with any kind of pen testing methodologies, various attacks and various defense mechanisms. The brief overview of our framework is as shown in Figure 2.

The framework is compatible accordingly with the testbed setup and consists of three main modules of orchestrating, analysis, communication and control. The orchestrating module contains sub modules of test sequencing and report generation. The orchestrating module is responsible for orchestrating and running any tests in the testbed. The communication and control module is responsible in invoking, connection and control of sub module of IoT device and measurement tools. The analysis module contains various sub modules. The sub modules testing methods and testing tools should run any kind of tests and should run any kind of network testing tools and also be able to be adaptable and tunable to the testbed. The classification sub module will provide which tests can be run on which device based on various classification of services, Operating System (OS) etc. The vulnerabilities sub module will be able to support and provide various existing vulnerabilities for comparison and also provide security metrics to measure the risk level of an IoT device. The attack methods sub module will be able to make various attacks to check the vulnerability of the IoT device and the defense methods sub module will provide every necessary proof of concept to patch the vulnerabilities. All the modules in the ATF works in a plug and play manner, such that it can be adaptable and tunable to various perspectives of IoT security testing.

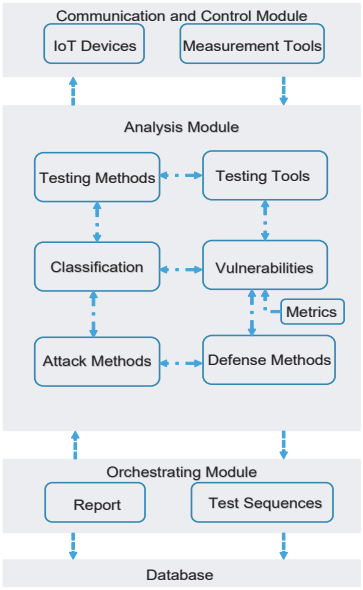

Figure 2: An Overview of Adaptable and Tunable Framework.

3.1 Security Analysis of IoT devices

Based on our setup of IoT testbed and ATF, we have done security analysis with various IoT devices. We briefly explain some of them based on the below chosen testing methods. These use cases chosen for security analysis fall under the testing methods of the analysis module of ATF. Before we start the security analysis of any IoT device, we call a sub module "Check-IoT", which is under the testing methods of analysis module. The Check-IoT will scan the entire shielded room to find the available IoT devices and then lists them on the OM for choosing to which IoT device the user would like to test.

- Port Scanning: is to check for open and vulnerable ports for any IoT device. Port scanning contains sub modules such as "import sub-process" which will conduct an intense scan on the selected IoT device to reveal any open UDP/TCP ports. The output of the import sub-process is the "scan-results". The sub module "IP-check" extracts relevant UDP/TCP ports from the scan-results, evaluate the risk level of the IoT device. The IP-check has two functions "Checker" and "Risk Evaluator". The Checker function sieves out the ports and results are gathered in "Open Port Results" file. The Risk Evaluator function analyzes the results, which is compared against a list of top open vulnerable ports. This list includes the metric scores of the various open ports that have been considered vulnerable by Tenable Network Security in Vulnerability Reporting by Common Ports [7]. The different risk levels are

as follows: Safe, Minor Risk, Major Risk, Critical Risk. Finally, the overall results are sent back to the Orchestrating Module for displaying the report (Figure 3.).

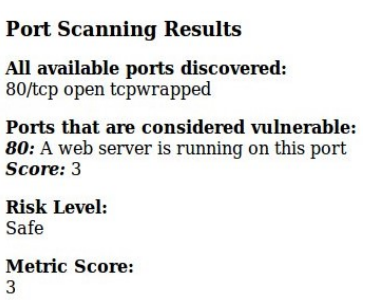

Figure 3: Report of Port Scanning.

- Finger Printing: provides the IoT device′s OS, manufacturer, type of device, IP and MAC address. For finger printing any IoT device, we use "DHCP Dump" sub module, Nmap [8] and Scapy python. We start the finger printing by creating a sub process in the shell using the "Popen" function which outputs the "DHCP Results", which will contain the DHCP dump of any device that has made a DHCP discovery or DHCP request. This process will continuously run in the background while the script is still being executed. With various sub modules and functions such as "Nmap Done Checker", "Deauthentication", "MAC Catcher" and "MAC Finder", we get the "Parameter Request List" of the IoT device. The Parameter Request List is helpful in getting the device OS fingerprint. With function "Chunk Siever" a list of numbers from the Parameter Request List is created, which will be used later on for comparison against the OS fingerprint list provided by "Packet Finder" function. Finally, the output from the Chunk Siever function is compared with "DHCP Fingerprints" and the report is generated (Figure 4.).

Figure 4: Report of Finger Printing.

- Vulnerability Scan: is to check for vulnerabilities on various OS with Common Vulnerability Scoring (CVS) for various IoT devices. The sub module "Check CVE" utilizes multiple python libraries in order to check the vulnerabilities from National Vulnerability Database [8]. The user will have to specify the OS platform, be it Android, Linux, Windows or Las OS. The function "Queryer" creates a string that contains appropriate HTML formatting and also opens the "allitems2005.csv", which contains all the Common Vulnerabilities and Exposures (CVE) and vulnerabilities. Finally, the csv file is parsed line by line to search the CVE number in [7] using the get request to extract the vulnerability details of the specific CVE number (Figure 5.).

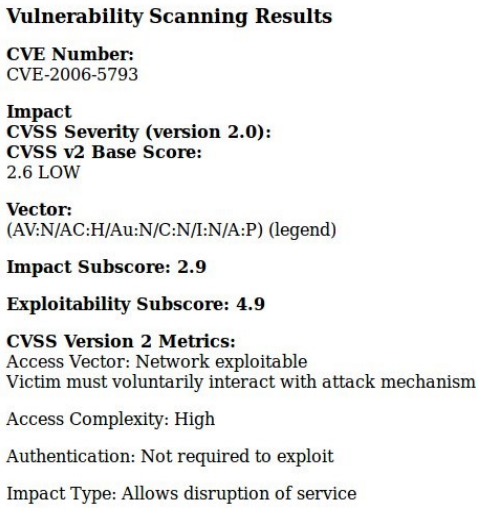

Figure 5: Report of Vulnerability Scanning.

4. CONCLUSION AND FUTURE WORK

In this paper we propose the IoT security testbed and ATF for security testing of various IoT devices. We have shown our primarily work on setting up the testbed and our preliminary architecture of ATF. We also showed the security analysis of IoT devices by choosing few testing methods. Our future work intends in building up the fully functional ATF with more modules to be able to adapt and tune accordingly with the testbed. On the other hand, we have the on going work on exposing various IoT devices available in the market and show how vulnerable these devices are.

5. REFERENCES

[1] SHODAN, https://www.shodan.io/

[2] M. Patton et.al, Uninvited Connections: A Study of Vulnerable Devices on the Internet of Things (IoT). *In Proc. of IEEE JISIC*, pp. 232–235, 2014.

[3] A. Gluhak et.al, A Survey on Facilities for Experimental Internet of Things Research. *IEEE Communications Magazine*, 49 (11):pp.58-67, 2011.

[4] NI TestStand, http://www.ni.com/teststand/

[5] NI LabVIEW, http://www.ni.com/labview/

[6] Tenable, http://www.tenable.com

[7] https://web.nvd.nist.gov/view/vuln/detail?vulnId=.

[8] Nmap, https://nmap.org/