IoT Metrics and Automation for Security Evaluation

Thomas Setzler
Department of Computer Science
College of Charleston, Charleston SC, USA
setzlerte@g.cofc.edu

Xenia Mountrouidou
Network To Code
New York NY, USA
xenia.mountrouidou@networktocode.com

***Abstract*—Internet of Things (IoT) devices are ubiquitous, with web cameras, smart refrigerators, and digital assistants appearing in homes, offices, and public spaces. However, these devices are lacking in security measures due to their low time to market and insufficient funding for security research and development. In order to improve the security of IoTs, we have defined novel security metrics based on generic IoT characteristics. Furthermore, we have developed automation for experimentation with IoT devices that results to repeatable and reproducible calculations of security metrics within a realistic IoT testbed. Our results demonstrate that repeatable IoT security measurements are feasible with automation. They prove quantitatively intuitive hypotheses. For example, an large number of inbound / outbound network connections contributes to higher probability of compromise or measuring password strength leads to a robust estimation of IoT security.**

***Index Terms*—IoT, Security Metrics, Automation.**

I. Introduction

The Internet of Things (IoT) is a rapidly growing sector of the digital world consisting of internet-connected devices. To keep up with the rapid growth, many IoT devices are being produced as quickly and as cheaply as possible, which raises serious concerns for security. In order to secure these IoT devices, a means of evaluation needs to exist specifically for assessing their security.

There are several challenges in evaluating security and specifically security of IoT devices. We list two of these challenges below:

1) ***How can we measure something that does not exist?*** Pfleeger et. al. in [1] define security evaluation as the measurement of the absence of an attribute. To this end, we propose a set of novel metrics that are based on known device behavior, not abstract, unknown concepts, such as the adversary's behavior or zero day attacks.
2) ***Quantified security is a false hypothesis.*** Verendel [2] has an extensive study regarding security evaluation techniques. The missing link in quantitative security work are repeatable, automated experiments that verify operational metrics' validity and realism. We develop automation to conduct repeatable experiments to evaluate IoT security metrics and a realistic testbed for experimentation.

To our knowledge, one of the few attempts to evaluate IoT security with quantitative metrics is presented in [3]. The authors evaluate End to End secure key management for health devices using third parties to handle the keys. However, this work focuses only on the specialized area of key management IoT security evaluation. In his survey on IoT security, Kouicem identifies two design approaches to securing IoT devices, unification and integration [4]. Babar et. al. [5] propose a general threat model IoT devices whereas Yin et. al. take a different approach analyzing attacks in the wild with an IoT Honeypot [6]. Acar et. al. [7] discover original IoT Web based attacks that can find IoTs even if they are behind a NAT, through studying their web portal responses. One of the most standardized efforts to evaluate security is the Common Vulnerability Scoring System (CVSS) [8]. Even though the NIST CVSS is an industry standard, there is opportunity to introduce additional metrics that are continuous and more granular than the LMH scale metrics. Abraham et. al. [9] introduce a non-homogeneous Markov model for security predictions, created based on: attack graphs, CVSS, Vulnerability lifetime and Frei's model to integrate time in the exploitability parameter.

Our work differs from previous works due to its focus on generic IoT security principles, not specific attacks. Our contributions are summarized below:

- *Development of novel security metrics for IoT:* we propose novel security metrics for IoT devices based on their fundamental characteristics and security principles. Our metrics are objective, repeatable, and reproducible.
- *Experimental evaluation of metrics:* we evaluate the metrics and show that they can be quantified and lead conclusions about the likelihood of attacks.
- *Automation for IoT security measurements:* because of the limited interfaces and APIs offered to access IoTs, automation and experimentation is challenging. We create novel automation that conducts repeatable experiments and collects security measurements.

II. Methodology

In this Section, we first specify the process to extract metrics based on fundamental device characteristics and security principles. Then we analyze in detail the metrics developed and evaluated in this paper. Finally, we describe the testbed used for experimental evaluation of IoT security metrics.

A. Development of IoT Security Metrics

In order to define IoT security metrics, we used the following criteria:

Security Metric	IoT Feature	CIANA Principle
Password Strength	Communication	C
# incoming connections	Communication	C, Availability
# of outgoing connections	Communication	I
# of active services	Purpose	C, Authenticity
% of successful attacks	Communication, Purpose, Mobility	C, I, A, N, A & Safety
% historically vulnerable	Communication, Purpose, Mobility	C, I, A, N, A & Safety

TABLE I
IOT SECURITY METRICS CORRESPONDING TO DEVICE CHARACTERISTICS AND CIANA PRINCIPLES.

1) **IoT features:** IoT devices represent a diverse ecosystem, therefore standardization of their access interfaces, software, and hardware features, is challenging. We based our metrics on the IoT taxonomy presented in [10]. This verifiable taxonomy defines the following shared features between IoT devices: communication, mobility, and purpose. We based our metrics on these common characteristics.
2) **Security pillars:** The fundamental security principles of Confidentiality, Integrity, Availability, Non-Repudiation, and Authenticity (CIANA) offer a simple yet robust base for defining security metrics. In addition, safety is another consideration for IoT devices, since they may affect humans physically and cause harm if operated by a malicious actor.

Table I shows the defined security metrics in combination with the IoT features and security principles that inspired these. Note that the metrics are not indicators of compromise, on the contrary they are means to predict a potential compromise. For example, if the password strength is low, there is a higher likelihood for an IoT portal or SSH service to be compromised. This assumption helps us relax the strict rules of absolute metrics, while maintaining the rigorousness of evaluating security with objective, reproducible, and quantitative measurements.

Next, we analyze each metric in Table I:

1) Password Strength: A large number of IoT devices operate with a default username and password combination [11]. Furthermore, passwords may be hard-coded and thus it is impossible to harden these.

In order to evaluate the strength of a password, we use password's information as it is calculated by information entropy [12] $I = -\log_2 prob_i$, where I is the information and $prob_i$ is the probability of the event i to occur. For the calculations of the value of a password information, an event may either be the occurrence of a word, or letter, or number, or character. The higher the value of I, the more complex and secure the password is.

To evaluate password complexity we need to consider if it can be successfully cracked via a brute force, a dictionary, or rainbow table attack. To this end, we combine statistical password guessability metrics as described in [13] with information entropy a simple formula to calculate the strength of

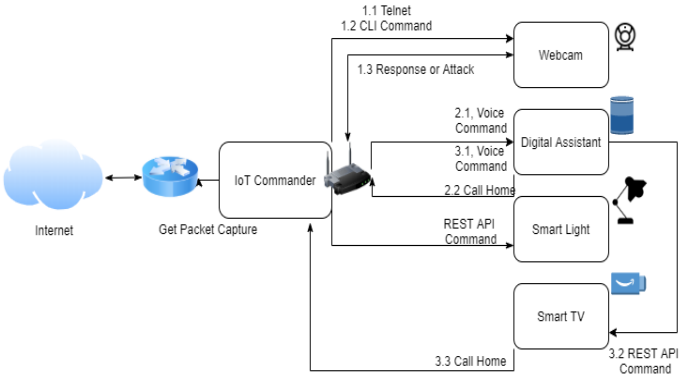

Fig. 1. Automation workflow where the IoT Commander initiates requests, triggers events, and collects data. Messages are marked with numbers.

an IoT password introducing the following metric:

$$Password\_Strength = \frac{I}{Password\_Guessability}, \quad (1)$$

where I is the information entropy defined above and $Password\_Guessability$ is a factor that indicates how easy it is to brute force a password given a specific algorithm. For simplicity, since a large number of IoT devices have hard-coded default passwords, we specify the ratio: $Password\_Guessability = \frac{1}{total\_guesses}$. This is an intuitive metric that is not affected by the algorithm used to guess the password and takes in account the worst case scenario of the password being at the end of a dictionary or file.

2) Number of Inbound and Outbound Connections: Incoming connections to network devices provide insight regarding security for the following reasons: i. if these are unscheduled and unsolicited, they increase the probability of an attack through reconnaissance, i.e., scanning, or password brute forcing, ii. increased number of incoming connections may conceal a malicious data transfer attempt or a Denial of Service. Similarly, a high number of outgoing connections may demonstrate potential for data ex filtration or data tampering. Both these metrics are derived by the communications characteristic of IoT devices.

These two metrics need to be interpreted within a specific context using the following criteria: i. the incoming connection is unsolicited or from unknown sources, ii. the inter-arrival rate is high, and iii. the number of incoming connection is statistically deviant from a regular traffic baseline.

3) Number of Active Services: IoTs often use open ports to expose key services for their functionality. Open ports increase the attack surface especially if they are open persistently. Thus, this metric considers a high number of open ports may increase the probability for IoT security breaches. This metric is derived from the purpose of IoT devices, since this correlates to the services that they are running.

4) Percentage of successful attacks: In order to define an objective number of successful attacks metric, we need to provide the definition of a successful attack. For an attack to be successful, it has to have a noticeable impact on the IoT device

with regards to the CIANA security pillars. The most obvious indication of a successful attack is the inability of a device to operate during or after the attack, i.e., the device availability is affected. Another implication of a successful attack may affect the physical safety since devices include actuators, such as robotic arms, smart cars, etc. Finally, an attack may succeed in stealing or tampering with information, i.e., confidentiality and integrity may be affected by a successful attack.

The metric that we propose is the percentage of successful attacks:

$$\%\_successful\_attacks = \frac{number\_of\_successful\_attacks}{total\_number\_of\_attacks}, \quad (2)$$

that can be calculated via penetration testing or running a standardized benchmark of attacks against IoT devices. This metric is correlated to the purpose of the device and affects all security pillars. The higher the percentage, the higher the probability of a successful attack from an adversary.

In the following Sections we discuss our methodology to evaluate and present a proof of concepts of calculating these metrics with automated software calls in a local testbed.

B. Automation Design

Manually operating IoT devices to extract repeatable security metrics can be a challenging task. To this end, we have designed automation to calculate baselines of normal IoT device operations and attacks with goal to evaluate the proposed metrics. The automation leverages existing Application Programming Interfaces (APIs) and python's OS library in order to run commands on devices without user interaction. The current automation targets home IoT devices and is divided into modules, each tasked with performing a single operation on the targeted device. These modules were combined into four scenarios to mimic real IoT usage: "wake up", "house party", "enterprise normal hours", and "enterprise after hours". "Wake up" scenario mimics a typical routine for waking up by turning on lights, playing music, and casting youtube videos via chromecast. "House party" sets the lamp to perform a strobe light effect while music is played, videos are played on a smart tv, and the webcam is quickly checked. "Enterprise normal hours" and "Enterprise after hours" imitates functions of a businsess such as turning lights on / off or checking cameras for security. The time intervals for each operation are extracted from random poisson distribution to produce realistic scenarios.

Automation was developed for the following attacks: Denial of Service (DoS) attack against an IoT, port scan, brute force password attack, and large ping initiated by an IoT that hypothetically is part of a botnet. Denial of service attacks against IoTs were implemented using hping3[1], scans were implemented using nmap[2], brute force password attacks used hydra[3]. Figure 1 shows the design of our automation with all the modules and interactions. A centralized "IoT commander" acted as an orchestrator for initiating automation calls and performing attacks. To initiate operations the commander either logs in to IoT devices or performs API calls. The commander also collects data and stores it in csv files for post analysis. There is one module that controls webcameras by establishing a remote telnet connection and sending commands as well as receiving responses, a second module that plays voice commands that control the digital assistant and extents to a third module that triggers the smart TV. Finally, REST API commands are sent through a module to the light bulb.

In order to test the vulnerabilities of IoT devices, we constructed a testbed of IoTs on a private subnet. The reason for this is to allow for thorough testing and experimentation without corrupting the metrics. To achieve this, our testbed was created by connecting a pfSense router to the local network and attaching a Netgear to the pfSense machine to function as a wireless access point for the bed. This simple setup combined with orchestration results to a robust monitoring mechanism for automated collection of metrics.

III. Results

In this Section we describe the experimentation results and the sample measurements that were taken using the testbed of IoT home devices and automation.

A. Password Strength

We measure password strength within the testbed devices by examining passwords that are entered either via mobile application or web portal using Equation 1. As shown in Table II, IoT devices have consistently weak default passwords, if they even have one. The devices that have default passwords are typically routers and webcams. When calculating the information of default passwords, we assume that the probability of a common word is $1/2000$, considering 2,000 common English words, the probability of a letter is $1/26$, and the probability of a digit is $1/10$. Table II shows the calculated information entropy, guessability, and password strength of default passwords for devices located in the IoT testbed. Guessability varies, from $1/65,000$ if password is picked from rare words, to an estimate of $1/100$ if a password is one of the most common default passwords found in IoT devices. There are several lists that enumerate these common IoT passwords [11]. The most complex password that belonged to the pfsense router, scored considerably better than a 6 digit string of numbers. Intuitively, this indicates that a small change from a default password to a non-common word may improve the security posture of a device dramatically.

B. Number of Incoming and Outgoing Connections

Figure 2 shows a breakdown of regularly occurring communications under normal operation scenarios automated as described in Section II. We have executed the scenarios five times and averaged the metrics to demonstrate that they are repeatable and reproducible. During these scenarios, outbound UDP connections occur more frequently than either inbound or outbound TCP connections. The outbound UDP connections

[1]https://linux.die.net/man/8/hping3

[2]https://nmap.org/

[3]https://tools.kali.org/password-attacks/hydra

IoT Device	Password	Information	Guessability	Strength
Pfsense	pfsense	20.37	1/65,000	1,324,050
Netgear	password	10.97	1/2,000	21,940
Samsung webcam	1234	13.29	1/100	1,329
Avacom portal	1234	13.29	1/100	1,329
Avacom telnet	none	0	0	0
D-Link webcam	123456	19.93	1/100	1,993

TABLE II
INFORMATION ENTROPY AND PASSWORD STRENGTH CALCULATIONS FOR IOT DEVICES IN TESTBED.

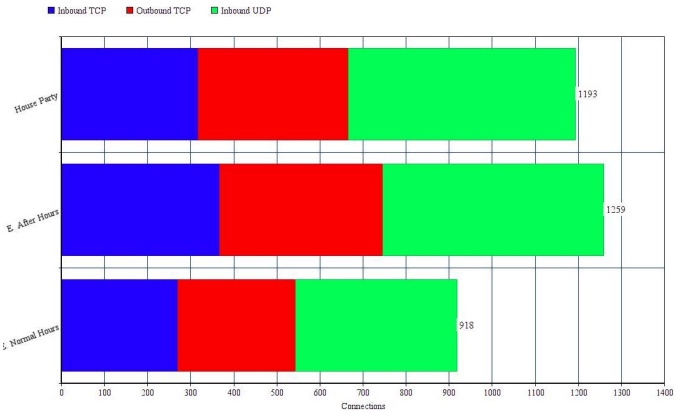

Fig. 2. Number of TCP and UDP connections for all normal operation scenarios.

commonly originate from the Samsung webcam, the Google Home, and the Google Chromecast. Across all scenarios, the number of inbound UDP connections has been very low, equal to one or zero. The number of outbound UDP connections can be attributed to the various devices "calling home" to their manufacturer or central server without an expectation of a response. The lack of UDP inbound packets is due to this lack of responses from devices calling home.

C. Number of Active Services

Table III shows the devices with their open ports and protocols. As shown in the Table, all open ports that were exposed in our testbed are using the TCP protocol. Note that none of the well-known ports (0 to 1023) are in use.

Based on Table III, the highest risk device is the Amazon Firestick with six open ports. The Google Home follows in second place with three open ports. Within the scope of the entire testbed, we present only the devices with consistently open ports. The rest of the devices only opened ports on demand.

IoT Device	Exposed Ports: Service
Avacom webcamera	10554/TCP
Amazon firestick	8009/TCP, 8888/TCP, 37459/TCP, 42896/TCP, 55443/TCP, 60000/TCP
Google Home	8012/TCP, 8443/TCP, 9000/TCP
Chromecast	8009/TCP, 9000/TCP

TABLE III
OPEN PORTS AND THE SERVICES THAT ARE EXPOSED FOR IOT DEVICES IN TESTBED.

D. Number of Successful Attacks

Our results indicate that the Avacom webcam was successfully attacked by both the DoS attack and the brute force attack, with a noticeable delay occurring during the running of both. Google Home and Echo are resilient to most attacks. This resilience indicates that the home assistants are less of a risk when attacked, due to transfer of computation to the Google and Amazon cloud and minimal hardware. Scans were successful when devices revealed open ports and these devices can be seen in Table III. The robustness of this metric is contingent on the number and types of attacks used and in our case only a sample of attacks were applied.

IV. CONCLUSIONS

We have shown with automation and realistic experimentation that our metrics are easy to evaluate with reproducible, repeatable experiments. Out metrics lead to intuitive results, for example they demonstrate that some devices, such as home assistants, are at low risk against attacks, but high risk of being monitored by the device manufacturer. Our future work includes generic automation for IoT security evaluation that is based on APIs and network automation standards, such as customizeable Ansible playbooks.

REFERENCES

[1] S. Pfleeger and R. Cunningham, "Why measuring security is hard," *IEEE Security Privacy*, vol. 8, pp. 46–54, July 2010.

[2] V. Verendel, "Quantified security is a weak hypothesis: A critical survey of results and assumptions," in *Proceedings of the 2009 Workshop on New Security Paradigms Workshop*, NSPW '09, (New York, NY, USA), pp. 37–50, ACM, 2009.

[3] H. A. Alaba, Othman, "Internet of things security: A survey," vol. 88, pp. 10–28, 2017.

[4] L. Kouicem, Bouabdallah, "Internet of things security: A top-down survey," vol. 141, pp. 199–221, 2018.

[5] B. Sachin, M. Parikshit, S. Antonietta, P. Neeli, and P. Ramjee, "Proposed security model and threat taxonomy for the internet of things (iot)," in *Recent Trends in Network Security and Applications*, pp. 420–429, Springer Berlin Heidelberg, 2010.

[6] Y. Pa, S. Suzuki, K. Yoshioka, T. Matsumoto, T. Kasama, and C. Rossow, "Iotpot: Analysing the rise of iot compromises," 08 2015.

[7] G. Acar, F. Li, A. Narayanan, and N. Feamster, "Web-based attacks to discover and control local iot devices," pp. 29–35, 08 2018.

[8] C. SIG, "Common Vulnerability Scoring System v3.0: Specification Document," tech. rep., First.org, Inc., 2017.

[9] S. Abraham and S. Nair, "Predictive cyber-security analytics framework: A non-homogenous markov model for security quantification," *CoRR*, vol. abs/1501.01901, 2015.

[10] B. M.-R. L. Mountrouidou, Xenia; Billings, "Not just another internet of things taxonomy: A method for validation of taxonomies," *Internet of Things*, vol. 6, no. 7, 2019.

[11] G. Cluley, "These 60 dumb passwords can hijack over 500,000 iot devices into the mirai botnet."

[12] C. E. Shannon, "A mathematical theory of communication.," *Bell Syst. Tech. J.*, vol. 27, no. 3, pp. 379–423, 1948.

[13] B. Ur, S. M. Segreti, L. Bauer, N. Christin, L. F. Cranor, S. Komanduri, D. Kurilova, M. L. Mazurek, W. Melicher, and R. Shay, "Measuring real-world accuracies and biases in modeling password guessability," in *24th USENIX Security Symposium (USENIX Security 15)*, (Washington, D.C.), pp. 463–481, USENIX Association, Aug. 2015.