Ph.D Thesis

Submitted in partial fulfillment for the degree of

Doctor of Philosophy

University of Lille 1 – Sciences and Technologies

Discipline: Civil Engineering

by

Elie SEMAAN NASR

Laboratory of Civil and Geo-Environmental Engineering

Security of Smart City Network Infrastructures: Design and Implementation Application to “Sunrise – Smart City” Demonstrator

Defended on November 29, 2017, before the committee:

Isam SHAHROUR	Prof. Université de Lille 1 – France	Advisor
Aziz SOULHI	Prof. Ecole National Supérieur de Mines – Rabat	Reviewer
Abdelmounaim ABDALI	Prof. Université Cadi Ayyad – Marrakech	Reviewer
Antoine HARFOUCH	MCF Paris Ouest Nanterre La Défense - France	Examiner
Said HAYYAT	Directeur de Recherche, IFSTTAR – France	Examiner
Sana OUNAIS	MCF Université Lille 1 - France	Examiner

Dedication

Exceptional cheers and thankfulness go to my beloved family who has encouraged me to complete each and every process of this study and has been able to be my backbone during the entire experience. My deepest love and respect to my wife Liliane who has been very encouraging, compassionate and supportive; along with my children Joe and Christopher who believed in my potentials.

&

to the soul of my Mother…

Acknowledgments

This thesis would not have seen the light without the trust and support of a number of wonderful people. My thanks and appreciation to all of them for being part of this journey. I would first and foremost like to single out Mrs. Hiam Sakr President of the American University of Science and Technology (AUST), I want to thank you for your motherly generosity and for all of the opportunities I was given to conduct my research and further my thesis. Nonetheless, I would also acknowledge that the research presented in this dissertation has been supported by the financial and moral assistance provided through the American University of Science & Technology, the educational institution that I grew up in and with.

I would particularly thank Dr. Riad Sakr, Vice President of the American University of Science and Technology. You were a core contributor to my mission by providing the best, most rigorous, and most intellectually exciting education.
I owe my everlasting gratitude to you, the sole person who had foreseen my potentials to conduct advanced research, paved my way and eliminated all the obstacles that faced me. You have always believed that the future is brighter when investing in human capitals.

To my dearest supervisor, Dr. Issam Shahrour, you definitely provided me with the tools, constructive criticism, support, and expertise needed to choose the right direction and successfully complete my thesis. You have supported me greatly, encouraged me continuously, and gave me invaluable suggestions throughout my PhD journey. Your close mentorship, deep insights, and intuition have greatly shaped my views on research in the area of smart city. Dr. Shahrour, I am so honored and pleased to have met and worked with you.

Next, I will never forget the contribution of Dr. Nabil Haidar, Provost of AUST. I thank you for writing the motivational letter that backed my acceptance in the PhD program. Your wise and prudent knowledge is a great asset for all of us at AUST.

Last but not least, I would like to thank my dissertation committee members for their time, commitment, and valuable feedback on my work.

Finally, I would like to thank my family and friends for being supportive, understanding, and loving. To my wife Liliane, I thank you for spending long hours with me, bearing the burden of understanding the

technical and complex security acronyms and paradigms to be able to edit and proofread this document. To my mother, may your soul rest in peace, you left us before you witnessed this moment. To Joe & Christopher, I thank God for his precious gift.

Résumé

Le but de cette thèse est de concevoir et mettre en œuvre une stratégie de renseignement sur les menaces cyber afin de soutenir les décisions stratégiques. L'alerte précoce et la détection des violations sont décisives, ce qui signifie que l'accent de la cyber sécurité a évolué vers l'intelligence des menaces. Pour cette raison, nous avons créé, analysé, mis en œuvre et testé deux solutions.

La première solution agit comme un mécanisme prédictif et proactif. C'est un nouveau cadre utilisé pour analyser et évaluer quantitativement les vulnérabilités associées à un réseau de villes intelligentes. Cette solution utilise le modèle de chaîne de Markov pour déterminer le niveau de gravité de vulnérabilité le plus élevé d'un chemin d'attaque potentiel du réseau. Le niveau de gravité élevé amènera l'administrateur système à appliquer des mesures de sécurité appropriées à priori aux attaques.

La deuxième solution agit comme un mécanisme défensif ou auto-protecteur. Ce cadre atténue les attaques par disponibilité zero-day basées sur Identification, Heuristics et Load Balancer dans un délai raisonnable. Ce mécanisme défensif a été proposé principalement pour atténuer les attaques par déni de service distribué (DDoS) car elles sont considérées comme l'une des attaques de disponibilité les plus sévères qui pourraient paralyser le réseau de la ville intelligente et provoquer une panne complète. Cette solution repose sur deux équilibreurs de charge dans lesquels le premier utilise une approche heuristique et le second agit comme une sauvegarde pour produire une solution dans un délai raisonnable.

Abstract

Combining multiple technological trends play a crucial role in achieving successful smart city. Cities are becoming smarter depending on the amount of new technologies they use. The building blocks of any smart city solution embrace the hardware and the software components. As technology continues to evolve, it brings ever greater threats. Keeping intruders out cannot be guaranteed, and yet no unique overarching cyber security architecture exists for all types of networks and its associated devices. For instance, at present, the Internet of Things (IoT) is considered one of the pillars of smart city through technological integration and collaboration. It can be difficult to ensure end-to-end security because most sensors and low-powered devices do not have sufficient computing power to support an encrypted network link. Cyber criminals are working on new techniques for getting through the security of established cities and organizations to cause damage, disrupt services and steal intellectual property. This is considered one of the essential challenges smart cities are currently facing. Technology becomes a central point for cybercrime, industrial espionage, and cyber-attacks. Therefore, protecting it is of paramount priority. So far, several cyber security incidents took place and several conventional security measures have been proposed and implemented but in fact these measures have proven to be inefficient due to the lack of consistent guidelines or widely accepted cyber security standards that are aligned with smart city's network needs.

The purpose of this thesis is to design and implement a cyber-threat intelligence strategy to support strategic decisions. Early warning and detection of breaches are decisive to being in a state of readiness, meaning that the emphasis of cybersecurity has changed to threat intelligence. For that reason we created, analyzed, implemented, and tested two solutions. The first solution acts as a predictive *and proactive mechanism.* It is a novel framework used to analyze and evaluate quantitatively the vulnerabilities associated with a smart city network. This solution uses the Markov Chain Model to determine the highest vulnerability severity level of a particular potential attack path in the attacks graph of the network. High severity level of a potential attack path will lead the system administrator to apply appropriate security measures a priori to attacks occurrence. The second solution acts as a *defensive or self-protective mechanism.* This framework mitigates the zero-day availability attacks based on Identification, Heuristics and Load Balancer in a reasonable time frame. This defensive mechanism has been proposed mainly to mitigate Distributed Denial of Service (DDoS) attacks since they are considered one of the most severe availability attacks that could paralyze the smart city's network and cause complete black out. This solution relies on two load balancers in which the first one uses a heuristic approach, and the second acts as a backup to produce a solution in a reasonable time frame.

Results of the first method have been proven to be robust and efficient. Security administrators can now use our method as a tool to predict and calculate the severity level of any potential attack sequence in the network and thus, applies the appropriate security measures as long as the severity level exceeds 40% which is considered in the range of medium to high. Simulation results obtained after testing the second solution showed that the heuristic approach associated with the backup load balancer led to substantial accuracy in mitigating DDoS attacks. Only one false negative outcome has been identified and recorded. As such, despite the fact that a regular user is not an attacker; she/he was not able to reach the service either. That's because only trusted sources are allowed to pass and use the service and this user is not marked as trusted in the backup load balancer rules.

Research Overview and General Introduction

Smart cities are highly digitized cities by nature. They are characterized by large volumes of data stored digitally and large numbers of physical objects with online connection to the Internet. In a smart city, almost all critical services like electricity, transportation, water, communication are increasingly becoming interconnected and dependent on smart technology. A resilient city must be able to tolerate disruptions, to anticipate, and to recover form disturbance [41]. As such, choosing the right technology is far a straightforward mission as it integrates multiple technological trends together. Ubiquitous computing, Networking, Open data, Internet of Things (IoT), Big data, Geographic Information System (GIS), Cloud computing, when all combined together play a crucial role in achieving successful smart city[42].

As the use of connected IoT devices constantly increase, so do the security concerns. The amount of networked IoT device will certainly increase in years to come. Currently there are up to 15 billion internet connected IoT devices in use and it is predicted to become 200 billion in 2020. We expect to see in the future more damaging IoT DDoS attacks because the source code of the Mirai malware is freely available on the Internet [81]. It has been published on a community hack forum as open-source by its suspected author Paras Jha using the moniker online name “Anna-senpai” [82].

Challenges

The biggest challenge that we face nowadays is the use of IoT devices (camera, DVR, thermostat, etc.) to launch a high-profile Distributed Denial of Service attacks as it happened lately on September 2016 with the Mirai botnet (generated up to 1.2Tbps of severe wave of network traffic). These attacks have brought businesses down, destroyed the economy of a nation and even led to governments being changed (See chapter 3 for details). This kind of botnet not only affected the IoT devices themselves but also everybody connected to the Internet. The Availability attack or what is commonly known as Denial of Service attacks have the greatest destructive effects and are considered the main cause for a city ***blackout.***

Thus, the need for cyber security to protect the network, the devices, the information exchange, and the privacy of citizens’ data becomes a must. The increasing number of attacks and the effects of these happenings show the importance of researching such types of attacks.

The focus on security can get lost if organizations rush to launch their IoT products, prioritizing speed-to-market over security. For most organizations, security is not the core focus of the IoT product development process. Securing a product from cyber-attacks is a critical element of the product development process in an IoT world.

Expanding Security Problem

Conventional wireless security measures are not enough to touch the IoT threats as they provide security solutions focusing on defending network perimeter. These security methods focus on establishing fences around an information system and then implementing security within each fence. For instance, Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS) solutions are decent enough at monitoring and preventing attacks coming over the network but they are weak at protecting IoT related attacks as these devices might use non-conventional communication protocols (Bluetooth, NFC, RFID, Z-Wave, ZigBee or 2G/3G/4G protocols). System administrators should not rely totally on the aforementioned protection mechanisms as they could not: a) compensate for weak identification and authentication mechanisms; b) conduct investigations of attacks without human intervention; c) perceive the contents of your organizational security policy; d) compensate for weaknesses in network protocols; e) analyze all of the traffic on a busy network; f) deal always with some problems involving packet-level attacks. Yet, no perfect mitigation for this problem exists [86, 87, 88].

The lack of common standards of interconnecting a city's various dimensions stances tremendous challenges on implementing the applications and their respected security aspects.

Originality of Solution and Contributions

In this study, we are aiming at designing and implementing an original cyber-threat intelligence strategy to support strategic decisions. Early warning and detection of breaches are decisive to being in a state of readiness especially for the smart city network, meaning that the emphasis of cyber-security has changed to threat intelligence.

Therefore our work identifies the necessary precursor steps through a twofold solution:

First, a predictive and proactive approach used to detect the highest vulnerability severity level of a particular attack path in the attacks graph of the network. This solution helps the security administrator predict and identify the severe vulnerabilities of all the devices and applications on the network, and thus providing the corresponding security measures prior to the attack.

Second, a defensive or self-protective approach used to lessen the impact of zero-day distributed denial of service attacks on the smart city network (IoT devices) and hence to avoid entire blackout.

Having attained the aforementioned solutions is considered as a step forward contribution to securing and protecting a smart city network as for:

1 - Demonstrating how predictive analytics and metric capabilities are the keys to boosting the security intelligence of smart cities

2 - Providing a smart city proactive and not reactive security approach; preventative and not remedial. A framework which seeks to anticipate privacy concerns rather than seeking to resolve privacy infractions once they have incurred

3 - Creating a novel predictive framework to analyze and evaluate quantitatively the vulnerabilities associated to the wired and wireless network in a smart city.

4 - Creating another novel defensive mechanism to mitigate Denial of Service attacks based on Identification, Heuristics and Load Balancer.

Table of Contents

Page

Chapter 1 – Smart City State of the Art

Chapter 2 – Proposed Cyber Security Solutions for Smart City

Chapter 3 – Analysis and Implementation of Proposed Solutions for Smart City Protection

List of Figures

List of Tables

Chapter 1

Smart City State of the Art

1.1 - Introduction

While this study focuses mainly on network infrastructure security and implementation of smart city, we believe that is it essential to first introduce and portray a generalized concept of smart city, its spanning industries, applications, technologies, network infrastructure and cyber security.

“When we thought we had all the answers, suddenly, all the questions changed”

Mario Benedetti

The rate of technology adoption in cities has witnessed considerable increase in the last decade. Cities are becoming smarter depending on the amount of new technologies they use. Smart cities need robust and resilient technologies to automate and improve city services, making citizens’ lives better. For a city to become smart, it must deploy new services like: smart traffic control, smart parking, smart public transportation, smart water management, smart energy management (smart grid), smart waste management and most importantly smart information and network infrastructure security. The building blocks of any smart city solution embrace the hardware and the software components. M2M (Machine to Machine) i.e, devices (machines) talking to each other, making decisions automatically, and sensors (which feed smart city systems with data are both considered the core part of a smarter city. The technology stack deployed in edge devices and gateways constitutes a multi-tiers architecture spread across: The communication protocols, the real operating system, the service in which the device communicate with, the software application (mobile apps), and the DBMS which supports storage and data analysis. This wide range of technologies and interdependencies that form a combination of local public and private industries is capable of working together to deliver complex smart solutions. The complexities of these smart solutions resulting from the widespread use of new technologies drive new and distinctive threats.

According to Gartner [1], by the end of 2020 there will be around 25 billion connected devices. Other estimations point out [2] that by year 2050, 70% of the world population is expected to live in urban cities. The increasing interconnectedness of intelligent and connected devices and infrastructure due to this gigantic demographic shift makes the cybersecurity of cities extremely significant and vital issue.

ISO/IEC27032 Guidelines for cybersecurity have defined cyber-security as preservation of confidentiality, integrity and availability of information in cyberspace. The fundamental building block of cybersecurity relies on information security, application security, network security, and Internet security. Therefore, the cybersecurity guidelines in the context of Smart Cities are required to provide guidance for improving the state of cybersecurity of smart cities [3].

Information security is hard. As technology continues to evolve, it brings ever greater threats. Keeping intruders out cannot be guaranteed and yet no unique overarching cyber security architecture for all types of networks and its associated devices. Cyber criminals are working on new techniques for getting through the security of established organizations to cause damage, disrupt sensitive data and steal intellectual property. This is considered one of the essential challenges smart cities are currently facing. Technology becomes a central point for cybercrime, industrial espionage, and cyberattacks. Therefore, protecting it is of paramount priority.

For instance, internet of things (IoT) is at present considered one of the pillars of smart city through technological integration and collaboration. It can be difficult to ensure end-to-end security because most sensors and low-powered devices do not have sufficient computing power to support an encrypted network link [4]. Where encryption is used, security issues can arise due to how it is operated [5].

No silver bullet exists to protect our smart city. Threats from several cyber-attack vectors like malware, botnets, and ransomware are apparent. Cybersecurity is not optional. It must be part of the design of every product and every gadget, and must also underpin every technology be it through broadband or local networks, wireless signals, or anywhere data is stored.

Research by cybersecurity experts has exposed how smart city systems have been established with no or minimal security. Moreover, city governments and vendors of smart city technologies often deploy them without undertaking cybersecurity testing [5]. For example, using the *Shodan* search engine (www.shodan.io) [6] it is possible to find all kinds of devices and control systems connected to the internet – from networked thermostats for heating systems to traffic control systems and command-and-control centers for nuclear power plants – many of which have been found to have little to no security (such as no user authentication, or using default or weak passwords, e.g., ‘admin’, ‘1234’). [7]

Chapter 1 starts with an informative introduction about cybersecurity in smart city and its conspicuous security challenges which emerge every day, a concise review of literature, a slight explanation about the suggested predictive and real-time defensive solutions, and a depiction of the thesis contributions. Then, a thorough elaboration on smart city state of the art which includes smart city concept, smart city industries, applications, stakeholders, standardization, technologies involved, network architecture, communication protocols, data layer, sensor layer, and IoT vs M2M. A generalized concept of cybersecurity mechanisms adopted in a smart city and in a smart building, types of cyber-attacks, their cause of vulnerabilities along with their security solutions. Finally, a typical threat model of public transport in smart city is referenced and criticized.

Chapter 2 proposes two efficient cyber security mechanisms for smart city as mentioned in chapter 1. It depicts the problem statement and portrays the gaps of the existing solutions and provides a detailed technical point of view of the vulnerabilities and risks threatening smart cities nowadays. A detailed explanation about my possible solutions and why we chose such solution are deeply investigated. Security concepts, indicators and metrics used to implement the solutions are examined in depth as they are considered the corner stone that is used to measure the vulnerability severity level.

Chapter 3 demonstrates a detailed implementation of the two proposed security mechanisms introduced in chapter 2 to protect the smart city. The solutions are provided through two research papers in which one of them has been accepted in an international conference and the second one is in the process of being published. The first published paper - Evaluating Wireless Network Vulnerabilities and Attack Paths in Smart Grid: Comprehensive Analysis and Implementation - details our first proposed solution: ***A preventive and proactive approach***: It is a novel framework used to analyze and evaluate quantitatively the vulnerabilities associated to the wired and wireless network in a smart city. It is a probabilistic approach and uses the Markov Chain Model to determine the likelihood of states transitions sequences of vulnerabilities attributed to attack paths using the Common Vulnerability Scoring System (CVSS) scores. The quantitative results have been investigated and examined to prioritize and detect the highest vulnerability severity level of a particular attack path in the attacks graph of the network.

Nonetheless, the second paper details our second proposed solution: ***A defensive or self-protective approach:*** This framework mitigates the zero-day availability attacks Based on Identification, Heuristics and Load Balancer in a reasonable time frame. The Distributed Denial of Service (DDOS) attacks are considered one of the most severe availability attacks that could paralyze the smart city's network and cause complete black out. Traditional security mechanisms like firewall, IPS and IDS could not prevent such type of zero day attacks. To reveal the severity of such attack, it is imperative to consider the case of the Dyn DDoS attack which was one of the biggest distributed denial of service attacks ever launched [8].

1.2 - Smart City Concept

Although many definitions of smart city exist [9], yet no universally specific definition has been adopted and recognized. Approaches vary as widely as the philosophy, values, culture, priorities, principles, needs, and histories of cities themselves. Smart Cities Europe, British Standards Institution, Smart Cities

Council**,** Boyd Cohen, ARUP, Cisco, Academia and others, each proposed its own definition of smart city from different perspectives but the singleton juncture among all definitions characterized the smart city as: " the use of advanced information and communication technologies to collect, communicate, and analyze data to improve the design and operations of a city's core systems and programs, as well as citizen engagement, for greater efficiency and effectiveness, thus improving the city's sustainability, resilience, bottom line and quality of life" [10].

To avoid confusion, various "smart city" term synonyms have been used extensively and broadly in books, research papers, reports, articles, and presentations. By and large, the Wikipedia encyclopaedia (*https://en.wikipedia.org/wiki/Smart_city)* depicted the following substitutes to "smart city" term: 'cyberville ', 'digital city', 'electronic communities', 'flexicity', 'information city', 'intelligent city', 'knowledge-based city, 'MESH city', 'telecity, 'teletopia', 'ubiquitous city', 'wired city'. However in this study we used only the terms "smart city", "intelligent city", and "digital city" interchangeably.

The importance of urban areas is a global phenomenon, as confirmed by the diffusion of megacities of more than 20 million people which are gaining ground in Asia, Latin America and Africa (UN, 2008). Most resources are nowadays consumed in cities worldwide. According to statistics more than 50% of the world population (3,5 bln) are now living in the cities and by the year 2050 the percentage will raise to 70%. This fact contributes to the economic and social importance of cities, but also to their poor environmental sustainability. With this large part of population moving to urban cities, developing and managing infrastructure of sustainable economy turned out to be a complex task. The importance of urban areas is a global phenomenon, as confirmed by the diffusion of megacities of more than 20 million people which are gaining ground in Asia, Latin America and Africa (UN, 2008). As a result, most resources are nowadays consumed in cities worldwide. This fact contributes to the economic and social importance of cities, but also to their poor environmental sustainability. As the consequences and challenges of urbanization became the major concern of the contemporary and modern cities, the perception of smart city comes as a solution. A subtle example which portrays the importance of smart city concept can be depicted after the study that was conducted by the Fraunhofer Institute for Systems and Innovation Research commissioned by the BITKOM digital association,. The study found that the complete networking and digitization of the German energy sector, up to and including the use of smart grids, could generate savings of approximately €9 billion per year.

As ABI Research stated that while $8.1 billion was spent on smart city technologies in 2010, by 2016 that number is likely reached $39.5 billion [Schelmetic, 2011]. As of today, there are 102 smart city projects worldwide, says ABI, with Europe leading the way at 38 cities, North America at 35, Asia Pacific at 21, the Middle East and Africa at six, and Latin America with two.

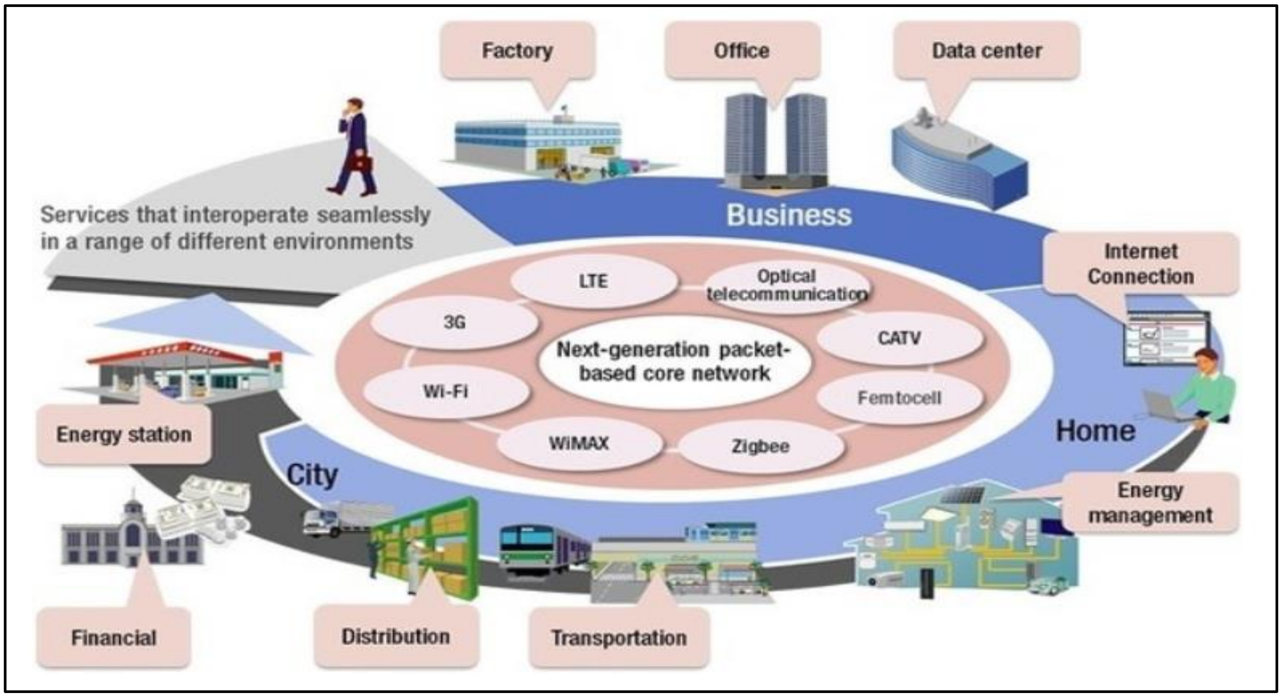


Figure 1.1 – Smart City Concept
Source: https://www.quora.com/What-is-the-concept-of-a-smart-city

The smart city concept spans around the city dimensions and industries, applications and key technologies (see figure 1.1). However the lack of common standards of interconnecting a city's various dimensions stances tremendous challenges on implementing the applications using ubiquitous and diverse technologies while taking into consideration their respected security aspects. Accordingly, the concept of smart city is not fixed and its ever-changing and adaptable nature makes the achievement of its operative stage even more complex [11].

There is a wide body of literature on the topic of smart cities. Some papers provided guidelines, policies, and real-life experimentation on new smart technologies (IoT devices), and some other papers tackled in depth smart city cybersecurity. Papers [12, 13] provided valuable guidelines for policy makers seeking to better define and drive smart city strategy and planning actions. Other papers [8] described the deployment and experimentation architecture of the Internet of Things (IoT) so under real-life conditions.
Paper [14] revealed the privacy risks associated with advances in the standardization of the smart grid, and verified the effectiveness of privacy impact assessment, with reference to privacy risks in the smart city. [15] provided a mobile-cloud-based smart city framework to avoid data over-collection in order to

improve user's data security. [16, 17] carried out a series of studies on reducing the risks of cyber intrusions and detection of several types of attacks on the smart grid, which can be considered as an essential partition of the smart city, and developed algorithms and visualization techniques for cyber trust in a smart grid system. [18]

Cerrudo [5] provided an overview of current real threats and possible cyber-attacks that could have a huge impact on cities by highlighting the vulnerability related to Traffic Control Systems, Smart Street Lighting, City Management Systems, Sensors, Public Data and Mobile Applications.

The research paper in chapter 3 - Evaluating Wireless Network Vulnerabilities and Attack Paths in Smart Grid: Comprehensive Analysis and Implementation – constitutes several literature reviews pertaining to the research's subject.

1.2.1 – Smart City Industries and Applications

What makes a city smart is converting the existing city's legacy industries into new innovative smart systems. The holistic approach to smart cities means that data from different dimensions and industries - smart energy, smart water, smart transportation, smart buildings, smart government, smart people, smart economy, smart mobility, smart living, smart health care, ***smart security*** and others - can be applied in a citywide context to maximize efficiency while minimizing costs. Key conceptual components of smart city were categorized in the following core factors: technology (infrastructures of hardware and software), people (creativity, diversity, and education), and institutions (governance and policy). Given the connection between the factors, a city is smart when investments in human/social capital and IT infrastructure fuel sustainable growth and enhance a quality of life through participatory governance [19].

The smart city industries conceive the city's infrastructure elements that play a major role in the life quality of the inhabitants and residents' happiness while providing sustainable development. In essence, the implementation of a smart city relies on the subsequent essential infrastructure elements: a) Adequate water supply; b) Assured electricity supply; c) Sanitation, including solid waste management; d) Efficient urban mobility and public transport; e) Affordable housing, especially for the poor; f) Robust connectivity and digitalization; g) Good governance, especially e-governance and citizen participation; h) Sustainable environment; i) Safety and security of citizens, particularly women, children and the elderly; Health and education. Preserving and maintaining smart city's infrastructure elements require rethinking of the business models and using disruptive technologies like Internet of Things (IoT) and Machine to Machine

(M2M) to create intelligent software integrated hardware products leading to smart, efficient, robust, and secure internetworking system.

Shifting from traditional city trends to a smart city requires converting data acquired by the hardware or software instrumentation(s) (sensors, meters, system automation) to intelligence. This transformation is possible by harmonic integration of “devices”, “network infrastructure”, and “administration”. Figure 1.2 shows how data is to be collected, integrated and analyzed to predict and make smarter decisions.

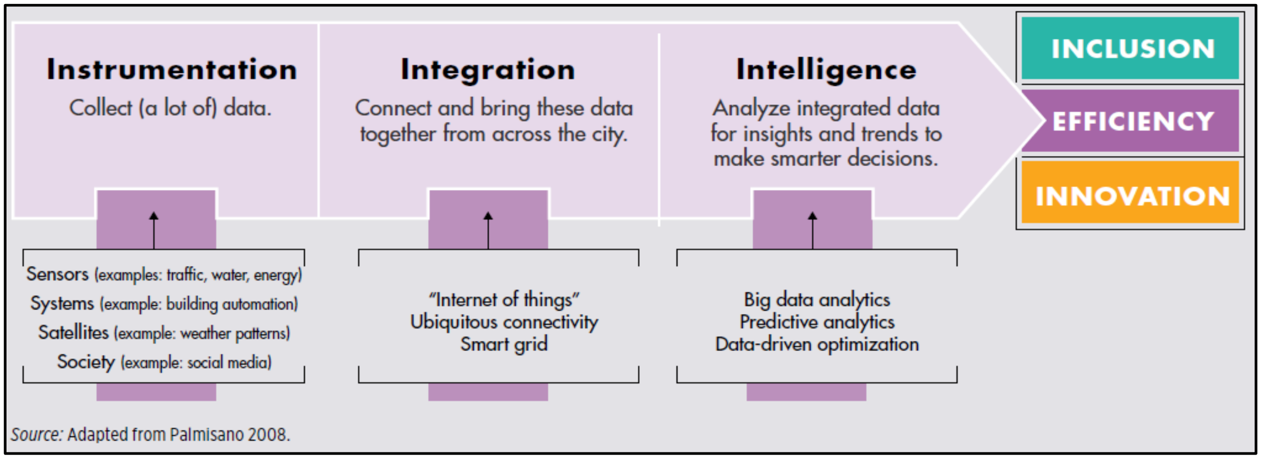

Instrumentation
Collect (a lot of) data.
Integration
Connect and bring these data together from across the city.
Intelligence
Analyze integrated data for insights and trends to make smarter decisions.
INCLUSION
EFFICIENCY
INNOVATION
Sensors (examples: traffic, water, energy)
Systems (example: building automation)
Satellites (example: weather patterns)
Society (example: social media)
“Internet of things”
Ubiquitous connectivity
Smart grid
Big data analytics
Predictive analytics
Data-driven optimization
Source: Adapted from Palmisano 2008.

Figure 1.2 – From Data Collection to Intelligence
Source: Adapted from Palmisano 2008.

Inasmuch as smart city strategies and policies are aimed at making better approaches and creating new ways to deliver public services, mechanisms must be be set up to guarantee that there will be open interest for these new services and products. For that reason, and to better understand the overall city’s intelligent management system, the following table summarizes the essential smart city’s dimensions, applications and their concerned technologies.

Table 1.1 – Smart city’s Dimensions and Technology

Dimension	***Applications***	***Technology Involved***
Smart Energy	Energy Saving Systems, Renewable Energy Integration, Electrical Vehicles (EV), Deliver on Demand Consumption, Periodic Energy Monitoring, Dynamic Pricing, Data Acquisition System, Big Data Applications, Knowledge Discovery Databases,	Wireless Automatic Meters, Remote Sensing Monitoring System, Fault Diagnostic, Distribution Automation, IoT/M2M device types.

	Forecasting and Scheduling Loads, smart grid,	
Smart Water	Consumption Monitoring, Physical Water Scarcity, Economic Water Scarcity, Water Treatment, Flood Management, Weather Forecasting, Modeling Water Channel Behavior System, Geographic Database, Thematic Data	Integrated Smart Water Resource Management (IWRM), Automatic Metering Infrastructures (AMI), Decision Support System (DSS), Pollution and Water Quality Control, Chemical and Physical Sensing. Decentralized Waste Water Techniques, Recharge Pits in Parking Lots and Green Areas
Smart Waste Management	Waste Composition, Waste Sorting, Waste Treatment, Waste Collection, Waste Transportation, Digital Management Infrastructure	Sensor Networks, Dust bins with Electronic Sensors, Vehicle with GPS Tracking System, IT based MIS System Connected to Citizen Complaint System.
Smart Building	Building Automation, Advanced HVAC Control, Lighting Equipment and Control, Fire Alarm, Data Network, Voice network, Power Management, Video Surveillance and Distribution, Access Control	IoT Sensing, Building Automation System Integration, Analytics, Remote Monitoring Services, Digital Signage, Fire Detection and Suppression Equipment, Office Lighting Application Report, Smart Connected Lighting, Metering, IoT Software Services, Connectivity and Devices, Photovoltaic and Renewable Energy, Security Services
Smart Healthcare	eHealth Systems, Intelligent and Connected Medical Devices, Self Tracking/Progress Feedback, Behavior Modification Tools, Physician Guidance, Clinical Decision Tools, Electronic Health Records. Smart Pills and Packaging, Remote Patient Monitoring, Diagnostic Tools	Wireless Blood Pressure Monitor, Passive and Active Tracking Devices, Activity and Sleep Wristband Tracker, Recommendation Engines, Smart Devices Integration, Safety Software, Interoperability, Drug Library

	and Call Center	
Smart Technology/Seamless Connectivity	Broadband Penetration Rate of Over 80%, Location Based Services, Augmented Reality, GPS Enabled Devices/Phones	Cognitive Systems, Internet of things, Machine to Machine, Robotics, Next Generation Security, 3D Printing, Cloud, Big Data Analytics, Mobility
Smart Security and Safety	Simulation Modeling and Crime Prediction, Incident Response, Network Penetration Testing	Surveillance, Biometric, CCVT Cameras, Helpline Numbers, Encryption, Authentication, Key Sharing, Confidentiality
Smart Transportation/Mobility/ Smart parking	Low Emission Mobility, Multimodal Transport, Traffic Monitoring and Management, Congestion Management, Road User Charging, Car Sharing, Emergency Response, Public Information Systems, Smart Parking, Solar Power Street Lights, Integrated Traffic Light Management	Traffic Surveillance, Smart Card for Public Transport Station Intelligent Transportation Systems, EV Charging Systems, Road Use Pricing Systems, Sensors Networks, Monitoring and Management Parking, Traffic Monitoring, Predictive Analytics, Vehicle Telematics, Public Portals and Smart Apps, Open Data Platforms.
Smart Governance	e-Government e-Education Disaster Management solutions City Performance Dashboards to Monitor the Performance of City Subsystems Through Digital Technology, GIS Based Workforce and Resource Management Solutions.	Sensor Networks, Cloud Computing Services, Data Analytics, Open Data Platforms, Lighting Networks, Emergency Response Systems
Smart Logistics	Packages, Letters, Containers, Bulkware, Tanks, Automated Pick Up Sheet Replenishment,	Robots, Validates, Classifies & Routes Requests.

	Driven by Production Cadence (Line Consumes a Truck – Replenish a Truck), Orders to Consumption - Build sequence, Automated VMA (Visualization, Monitoring, Alerting), Receiving Manning Reduction, Payback Measured in Months	Visualization, Monitoring & Alerting Applications
Smart Citizen	Use of Green Mobility Options Smart Life Style Choices Energy Conscious	Civic Digital Natives

While this table presents the various dimensions, applications and technologies constituting the smart city, the following section elaborates upon the stakeholders involved in developing the smart city.

1.2.2 - Smart City Stakeholders

At present, there is no universally defined list of stakeholders. Cisco and ITU each defines the main stakeholders based upon their roles in managing the city's assets. When selecting any new project or initiative, it is essential to consider policy makers and other stakeholders. In essence, creating a sustainable smart city requires harmonized and coherent political, managerial, and technical cohesion of all stakeholders who are only involved in the building process. They are responsible to cooperate together to make the city smarter either by using new technologies, or by combining existing technologies in new ways. The development process cannot but seen as an open chain structure constituting the member stakeholders. Figure 1.3 shows a generic topology of a mix of stakeholders that are needed to develop a sustainable smart city.

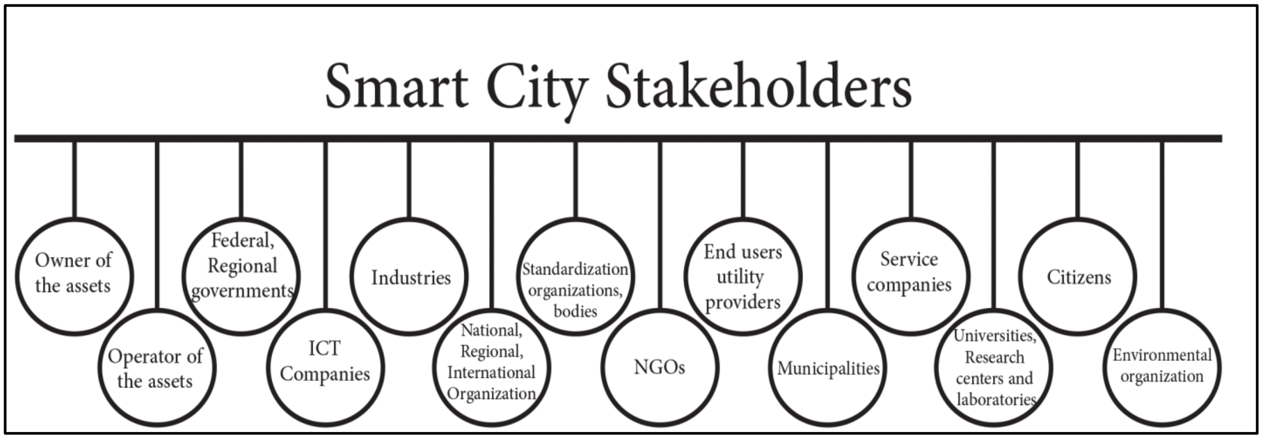

Smart City Stakeholders
Owner of the assets
Operator of the assets
Federal, Regional governments
ICT Companies
Industries
National, Regional, International Organization
Standardization organizations, bodies
NGOs
End users utility providers
Municipalities
Service companies
Universities, Research centers and laboratories
Citizens
Environmental organization

Figure 1.3 – Smart City Stakeholders

The following section describes and demonstrates technically various smart devices alongside the network infrastructures adopted and implemented in most of smart city industries.

1.2.3 - Smart City Technologies

First and foremost, the smart city is established based on a set of requirements and solutions which are considered as a mixture of today's independent technologies and innovations. The rapid growth of smart city's network infrastructure and devices interconnectivity accents tremendous challenges on the city's platform components and its fragmented model of sectorial applications. The challenges are stretched to devices' portability, inter-networking manageability, systems and software applications interoperability, and information security applicability, caused by the absence of unified standardized policies and procedures. The stakeholders aim at tearing down these challenges and integrate all city systems into a horizontal platform through the use of predictive analytics, intelligent system, and emerging communication trends, thus creating standard and scalable platform which facilitates the growth of sustainable smart city ecosystem. ICTs play a major role in making optimal use of all the interrelated information of a smart city to better control the use of limited resources, and facilitate the aggregation of city's industries' information for data analysis. Ultimately, one of the essential key factors of ICT in smart city is the ability to capture, govern, share and manipulate information in real-time so that cities can take appropriate decisions before the problem occurs or starts to escalate.

Currently, the smart city concept is established on a set of prevailing standalone technologies, namely, Internet of Things (IoT), Cloud Computing, Big Data, Machine to Machine (M2M), Data Analytics, Business Intelligence, Building Automation, Wireless Sensors Network, Next Generation Devices, E-cards, Information Technology Security and many others. However, the decisive solution of smart city is to invest on disruptive technologies which fuel sustainable and rapid economic growth and a high quality of citizens' life. **Techopedia (https://www.techopedia.com) defines** disruptive technology as "an enhanced or completely new technology that replaces and disrupts an existing technology, rendering it obsolete." Today, the four interchangeable terms: Internet of Thing, Internet of Everything (IoE), Network of Things (NoT), and Machine to Machine (M2M) are considered as disruptive technologies and they act as fundamental components of the smart city's network.

To better understand the role of disruptive technology in a smart city, it is essential to explain the concept of ubiquitous computing in addition to the smart city network architecture, and particularly the hardware, software, devices components, and their communication protocols.

1.2.3.1 – Ubiquitous/Pervasive Computing

Ubiquitous or pervasive computing is when at any moment and using any device, tailored information is automatically and proactively conveyed to inhabitants – certainly those who wish to accept it - without the need to request it or search for it themselves. Internet of Things, the growing trend of embedding capability into everyday object, is considered the enabler of ubiquitous computing. The personalized information is collected based on citizens' preferences and profiles. Information is controlled through open and secure platforms where public and private businesses can manage to innovate or improve their operations.

1.2.3.2- Framework for Smart Cities Standardization

The standards are principle based and not rule based. They provide framework for performing and promoting audit and clear demarcation of scope of roles and responsibility for participating standard development organizations. Although standard are mandatory requirements, no unified standard exit for smart city network architecture.

The reference architecture of Information Technology (lT) infrastructure in smart city suggested by National Institute of Standards and Technology (NIST) serves as a common starting point for system planning while promoting interoperable functional building blocks, which are required in a smart city. See figure 1.4 below:

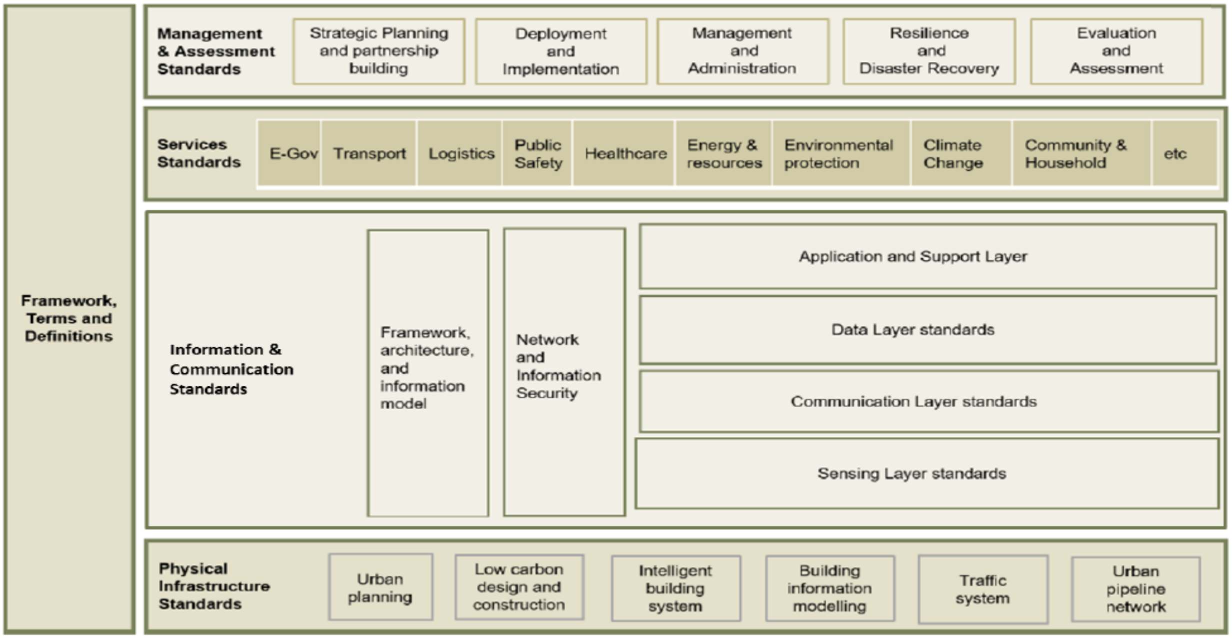

Framework, Terms and Definitions
Management & Assessment Standards
Strategic Planning and partnership building
Deployment and Implementation
Management and Administration
Resilience and Disaster Recovery
Evaluation and Assessment
Services Standards
E-Gov
Transport
Logistics
Public Safety
Healthcare
Energy & resources
Environmental protection
Climate Change
Community & Household
etc
Information & Communication Standards
Framework, architecture, and information model
Network and Information Security
Application and Support Layer
Data Layer standards
Communication Layer standards
Sensing Layer standards
Physical Infrastructure Standards
Urban planning
Low carbon design and construction
Intelligent building system
Building information modelling
Traffic system
Urban pipeline network

Figure 1.4 - Smart City Standardization Framework
(Source: https://www.mcmc.gov.my/skmmgovmy/media/General/pdf/Framework-on-Smart-Cities)

Another document on smart city standardization framework has been published by the ITU FG-SSC 0097 specifications document on smart cities infrastructure. This document has been referenced and adopted by several other major standard development organizations to include IEEE and ISO/IEC [20].

1.2.3.3 –Smart City Network Architecture

Smart city Network architecture can be seen through three different scopes:

1- Cyber-Physical Systems
2- High level Architecture
3- Low Level Architecture

1.2.3.3.1 – The Cyber-Physical Systems or the Urban Layer

Cyber-Physical Systems (CPS) or "smart" systems are co-engineered interacting networks of physical and computational components. The CPS is also called the urban layer in which the physical and digital infrastructures meet. Examples include: Smart Buildings, Smart Grid (Utilities – Water, Electricity, Gas), Smart Waste and Smart Mobility [21]. Figure 1.5 hereunder exhibits a typical framework of a CPS.

Figure 1.5 –Smart City Cyber Physical System
(Source: https://www.nist.gov/el/cyber-physical-systems)

These integrated systems will provide the foundation of the critical network infrastructure.

1.2.3.3.2 – The High Level Architecture

A generic high level architecture of smart city is composed of: Data Acquisition Layer (sensors, or IoT devices), Network Layer (edge routers), Aggregation Layer (gateway), and Data Analytics Layer (Data Center). The reference architecture of Information Technology (IT) infrastructure in Smart city suggested by National Institute of Standards and Technology (NIST) serves as a common starting point for system planning while promoting interoperable functional building blocks, which are required in a smart city.

Figure 1.6 shows a generic high level conceptual design of smart city ICT network

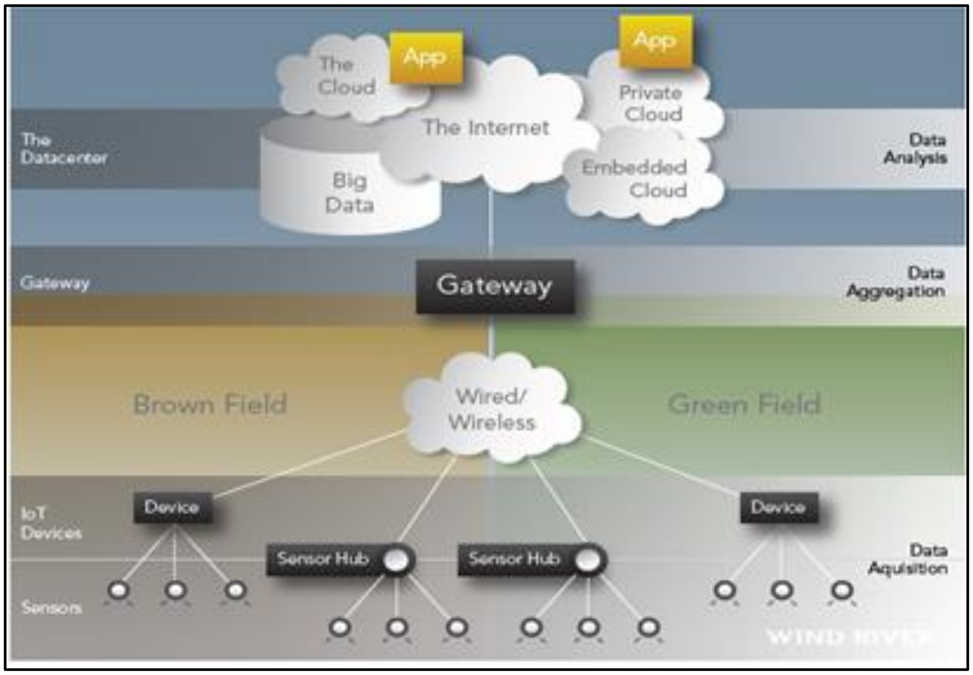


Figure 1.6 – High Level Conceptual Design of Smart City Network
(Source: Wind River)

1.2.3.3.3 – The Low Level Architecture

The generic low level architecture of smart city generally consists of four layers - a sensing layer, a communication layer, a data layer and an application layer, and these four layers are overseen by the smart city security system. Architecture of Information Technology systems deployed in smart city needs to be open, interoperable and scalable.

The message exchange between various applications in the smart city should be fully encrypted and authenticated. Any application outside the Data Centre (DC) should talk to the applications hosted in the datacenter through only predefined APIs. Figure 1.7 depicts a selected low level technical design of the ICT network.

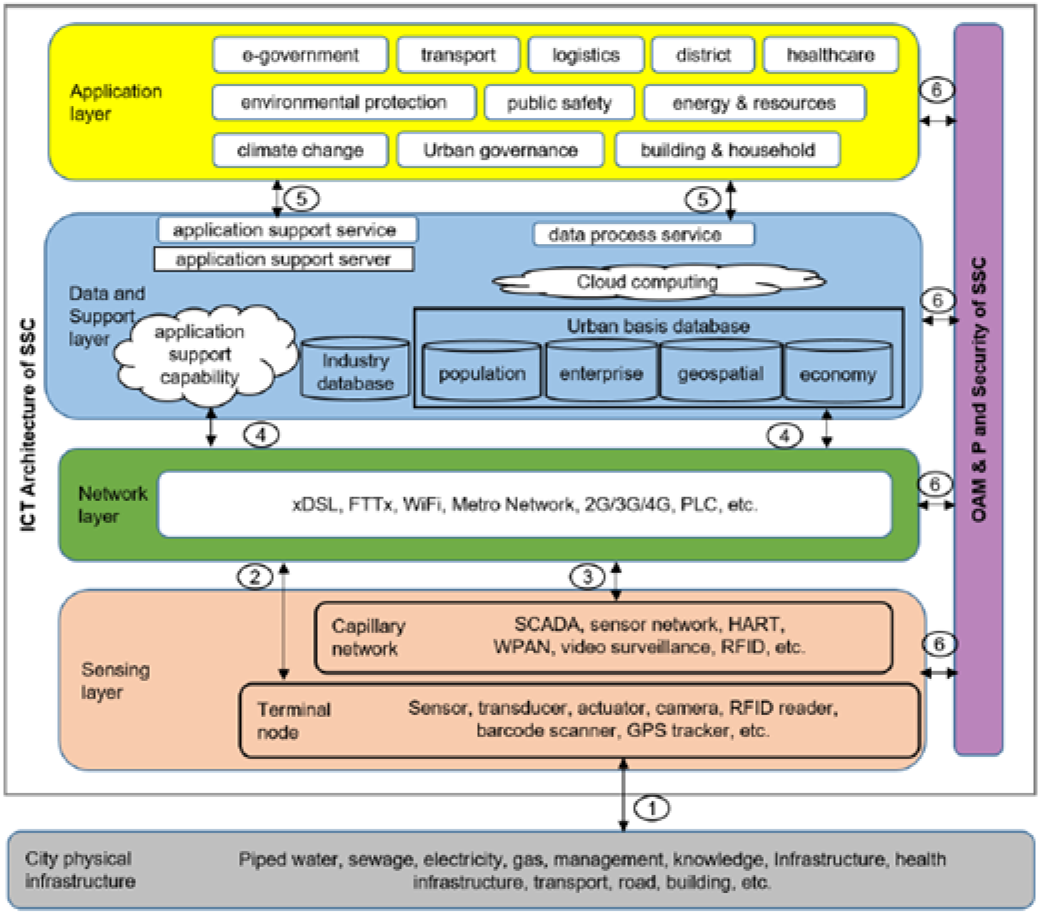

Application layer
e-government
transport
logistics
district
healthcare
environmental protection
public safety
energy & resources
climate change
Urban governance
building & household
5
5
application support service
application support server
data process service
Cloud computing
Data and Support layer
application support capability
Industry database
Urban basis database
population
enterprise
geospatial
economy
4
4
Network layer
xDSL, FTTx, WiFi, Metro Network, 2G/3G/4G, PLC, etc.
2
3
Sensing layer
Capillary network
SCADA, sensor network, HART, WPAN, video surveillance, RFID, etc.
Terminal node
Sensor, transducer, actuator, camera, RFID reader, barcode scanner, GPS tracker, etc.
6
6
6
6
ICT Architecture of SSC
OAM & P and Security of SSC
1
City physical infrastructure
Piped water, sewage, electricity, gas, management, knowledge, Infrastructure, health infrastructure, transport, road, building, etc.

Figure 1.7 – A Selected Low Level Technical Architecture of Smart City ICT Network (Source: https://www.mcmc.gov.my/skmmgovmy/media/General/pdf/Framework-on-Smart-Cities)

The above figure shows how sensing devices, network, data, and application layers interact with each other. The sensing layer senses signals from the city's physical infrastructure components and deliver it to the network layer. The network layer adopts the appropriate protocol to transfer information to be stored in the cloud or in the fog or in the corresponding applications' database management system. The application attributed to a specific industry retrieves the data from the storage for processing and for further analysis.   (For more details see chapter two)

The smart city architecture should be capable of managing heterogeneous data which would be continuously communicated through numerous devices following different protocols. ln order to ensure that the flow of data between devices does not run into latency issues, appropriate protocols need to be deployed so as to minimize latency.

1.2.3.4 – Smart City Layers Demystified

1.2.3.4.1 - The Sensing Layer

A *sensor* is an electronic device which measures physical properties or states in the cyber-physical system: Sensors employ various effects at an interface to a controlled process or open environment: A multitude of sensor types exists: For instance, air pollution, vibrations, accelerometer, energy usage, temperature, gyroscope, heart rate, blood pressure, material conditions speed, water leakage, pollution, etc. Sensors and networked devices with mesh technologies (see chapter two), ultimately improving the city's resilience [28].

The sensing layer consists of terminal node and capillary network. Terminals (sensor, transducer, actuator, camera, RFID reader, barcode symbols, GPS tracker, etc.) sense the physical world. This layer has the ability and the intelligence for monitoring and controlling the physical infrastructure within the city.

The capillary network is a local network that uses short-range radio-access technologies connects various terminals to network layer, providing ubiquitous and omnipotent information and data.

It is recommended to use the standardisation of the following technologies (but not limited to) in the sensing layer of smart cities: a) IEEE 1451 smart transducer interface; b) ISO/IEC JTC 1 SC 31 and AIM PDF417 barcode symbols; c) ISO/IEC JTC 1 SC 31 and Electronic Product Code (EPC) global Radio Frequency Identification (RFID); d) ZigBee; e) IPv6 over low power wireless personal area networks (6LoWPAN); f) wireless MBus; g) Global positioning system (GPS); h) video surveillance; and i) smart metering. [3]

The smart meter acts as an example of smart sensing/actuating device. It provides direct readings, updates the readings to achieve energy savings, allows remote on/off control of supply and/or power limitations, supports advance tariff systems, provides secure data communications and performs fraud detection and prevention.

Also, Deloitte [22] provides an exceptional sensing example through the smart lamppost. The lamppost is the most valuable real estate asset in the city for the purpose of installing sensors that form a city wide IoT backbone. Of all objects, lampposts are best distributed across the entire city and they already are connected to power networks. There are almost 116.000 lampposts in Amsterdam.

Security is a concern for sensors if their data is tampered with, stolen, deleted, dropped, or transmitted insecurely so it can be accessed by unauthorized parties. Building security into specific sensors may or may not be cost effective;

1.2.3.4.2 - Communication Protocols/Network Layer

The network layer refers to the network provided by telecommunication operators. Communication channels move data between computing, sensing, and actuation. No standardized communication channel protocol is assumed
Smart city communication networks are divided into 4 categories: fixed broadband, mobile broadband, M2M/IoT (explained below) and iBeacons networks.

Three types for fixed broadband networks are available and they could provide internet access at a bandwidth of 100 Mbps to 1 Gbps and higher: Fiber optics, Coax and Twisted pair networks. Mobile broadband networks: 3G (5-10 Mbps), 4G/LTE (up to 50 Mbps), and 5G networks(up to 1 Gbps) M2M/IoT networks (LoRa): Machine-to-Machine (M2M) and Internet of Things (IoT) communications have specific charcteristics: Long range (several kilometers), Low energy (Battery operated), Low Bitrate (0.5Kbps to 50 Kbps). iBeacons are small, battery operated devices that use Bluetooth Low energy to transmit a unique identifier in a range of 50-70 meters.

Several communication protocols could be used for the different layers for data flow. (see chapter 2 for more details)

1.2.3.4.3 - Data Layer and Big Data

Data Layer is fundamental in making the city "smarter" and capable of communicating with various types of sensors/ devices and their management platforms/applications. Data layer contains data center from industries, departments, enterprises, as well as the municipal dynamic data center and data. Three Data Analytics solutions are provided in this layer: (1) Descriptive, which uses business intelligence and data mining to ask: "What has happened?" (2) Predictive, which uses statistical models and forecasts to ask: "What could happen?" and (3) Prescriptive (includes Cognitive), which uses optimization and simulation to ask: "What should we do?" [23]

Big Data refers to extremely large data sets collected in real time. this explosion of "Big Data" is fed, in art, by the growth of what is called "the Internet of Things" (IoT): the infusion of data-collecting sensors and actuators in common everyday objects such as parking meters, pacemakers, and thermostats. predictive capability can be provided by applying Machine Learning to the data. Information collected by the city from the various domains and verticals, when studied and analyzed in aggregate, can provide insights to city officials previously not identified. A Smart City, as a "system of systems", can potentially generate vast amounts of data, especially as cities install more sensors, gain access to data from sources such as mobile devices, government, and other agencies make more data accessible [22, 23, 24].

1.2.3.4.4 - Application Layer

The application layer includes various applications that manage the smart city and deliver the smart city services. It is considered as an open society's infrastructure system. The development of applications needs the support of common knowledge of Smart Cities. Examples of application services span various smart city industries like e-government, transport and logistics, healthcare, public safety, energy and resources, community and household, among others.

To understand better the concept of smart city network architecture, and to investigate how different layer communicate among each other, consider the example of smart energy provided by USDN [25].

"Smart" energy efficiency arises when a building energy disclosure ordinance is coupled with the following:

1. Energy use data generated through meters (data generation)
2. The data is transmitted in real time through a broadband connection (distribution of data)
3. Data from multiple buildings is aggregated in a dedicated data server(s) (aggregation)

4. Aggregated data is analyzed for trends and patterns such as peak loads, high energy users, and efficient energy users (analysis)
5. Software applications are developed to enable smarter decisions on energy efficiency across all commercial buildings in the city. For example, a "heat map" can be created showing which buildings have the lowest utility costs. This information can be used to drive the energy efficiency market.

1.2.3.5 - Internet of Things versus Internet of Everything and Machine to Machine

Yet, no universally accepted and workable definition exists to the question, "What is IoT?"
Cisco Systems refers to IoT as the "Internet of Everything, while Bruce Schineier[1] recently referred to two new colloquial terms – World Spanning Robot and Benign Organization. There is also the term "Skynet" in reference to the Terminator movies that is frequently discussed in Blog and online postings/jargon. As figure 1.8 depicts, Continental Automated Buildings Association (CABA) stated that Internet of Things (only IP addressable devices) acts as a subset of Internet of Everything (IoE).

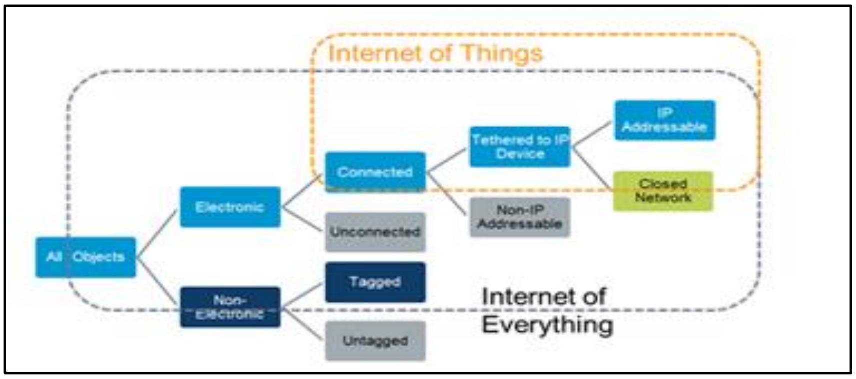


Figure 1.8 - Internet of Things vs. Internet of Everything
Source: CABA Continental Automated Buildings Association

CABA classified the devices and objects into four categories:

1- Unconnected Objects: Desk, chair, animal collar, building…
2- Unconnected Electronic Device: Calculator, streetlight, vending machine, …
3- Connected/Tethered Electronic Devices: Audio headset, printer, DVD,…
4- IP-addressable Devices: Tablet, PC, smartphone, EV charging station, …

All these four categories form the Internet of Everything when internet protocols (IPs) are assigned to each object and device.

[1] Bruce Schneier is an internationally renowned security technologist, called a "security guru" by *The Economist*. (https://www.schneier.com)

Several research papers and organizational definitions [26] have been proposed. In fact, they all lead to one opening statement "*This technology employs a mixture of sensing, communication, computation, and actuation."* Also, IoT has been perceived as an instantiation of a Network of Things (NoT), more specifically, it has its things (electronics devices, software, sensors, smart objects,…) tethered to the Internet where it forms what is called machine-to-machine (M2M) communication.

Machine-to-Machine communication is non-interactive, automated, and bi-directional information exchange in operational systems, performed between peers or between satellite systems and their supporting backend services

As the use cases emerge across several industries, the landscape of Internet of Things or Network of Things or M2M will become more applicable and trustworthy. Thus, the relevance of integrating third party IoT applications rises, and confidence grows and expected to usher in automation in nearly all fields, paving the way for advanced applications like a smart grid, and accordingly a smart city.

Statistics showed how promising and vigorous is the IoT technology to our future. Cisco is expecting 50 billion devices connected to the internet by 2020. Cisco also states that IoE creates $14.4 Trillion of value at stake for companies and industries. TechNavio, a tech-focused research firm expected the global Internet of Things market to grow at a Compound Annual Growth Rate (CAGR) of 31.72 percent from 2014-2019.

As Figure 1.9 shows, the International Telecommunication Union (ITU) reveals the outstanding significance of integrating the IoT technology into the various smart city industries. IoT transforms the unconnected, closed vertical silos (industries) of functionally oriented service providers into innovative and collaborative new models that connect these vertical silos.

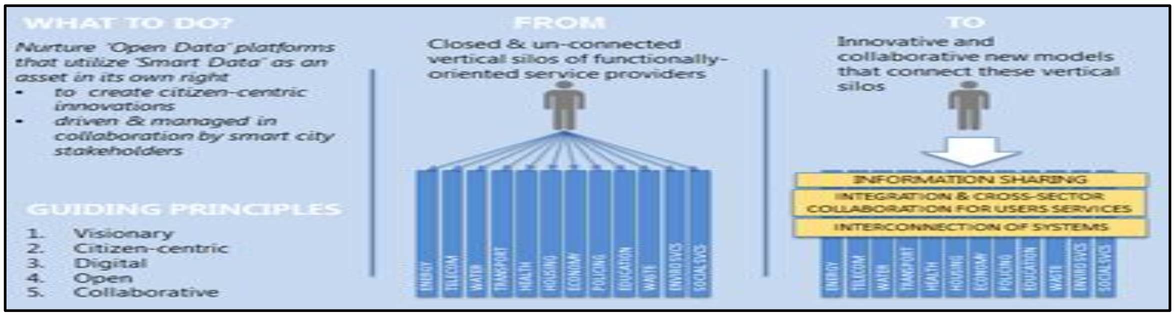

WHAT TO DO?
Nurture 'Open Data' platforms that utilize 'Smart Data' as an asset in its own right
- to create citizen-centric innovations
- driven & managed in collaboration by smart city stakeholders
GUIDING PRINCIPLES
1. Visionary
2. Citizen-centric
3. Digital
4. Open
5. Collaborative
FROM
Closed & un-connected vertical silos of functionally-oriented service providers
ENERGY
TELECOM
WATER
TRANSPORT
HEALTH
HOUSING
ECONOMY
POLICING
EDUCATION
WASTE
ENVIRO SVCS
SOCIAL SVCS
TO
Innovative and collaborative new models that connect these vertical silos
INFORMATION SHARING
INTEGRATION & CROSS-SECTOR COLLABORATION FOR USERS SERVICES
INTERCONNECTION OF SYSTEMS
ENERGY
TELECOM
WATER
TRANSPORT
HEALTH
HOUSING
ECONOMY
POLICING
EDUCATION
WASTE
ENVIRO SVCS
SOCIAL SVCS

Figure 1.9 - Towards IoT – Enabled Smart Communities
Source: ITU 2016

As a matter of fact, the IoT endorses a new third generation of the Internet progress. The first generation provided fixed wired connections to the Internet. The second generation connected a wide array of mobile devices to the Internet. The third new IoT generation connects a much larger number of things to the Internet, when converting a plethora of unconnected object to IP addressable smart objects.

Figure 1.9 below from Beecham Research demonstrates how the Internet of Things can be integrated in most of the segments within the public/private sectors and ordinary consumers. Public sector entities such as hospitals may have some level of interaction within all other service sectors; ranging from energy, healthcare and industry elements of hospitals, to levels of research, retail entities, transportation, and IT/Networks. [27]

Indeed, figure 1.10 splits the smart city components into tracks and sectors aiming at demonstrating how the IoT technology is incorporated and mapped to service elements. From inner to outer, the tracks range from services, application groups, locations to devices. The service track for example ranges from buildings, energy, consumer & home, healthcare & life science, industrial, transportation, retail, security & public safety to IT & networks.

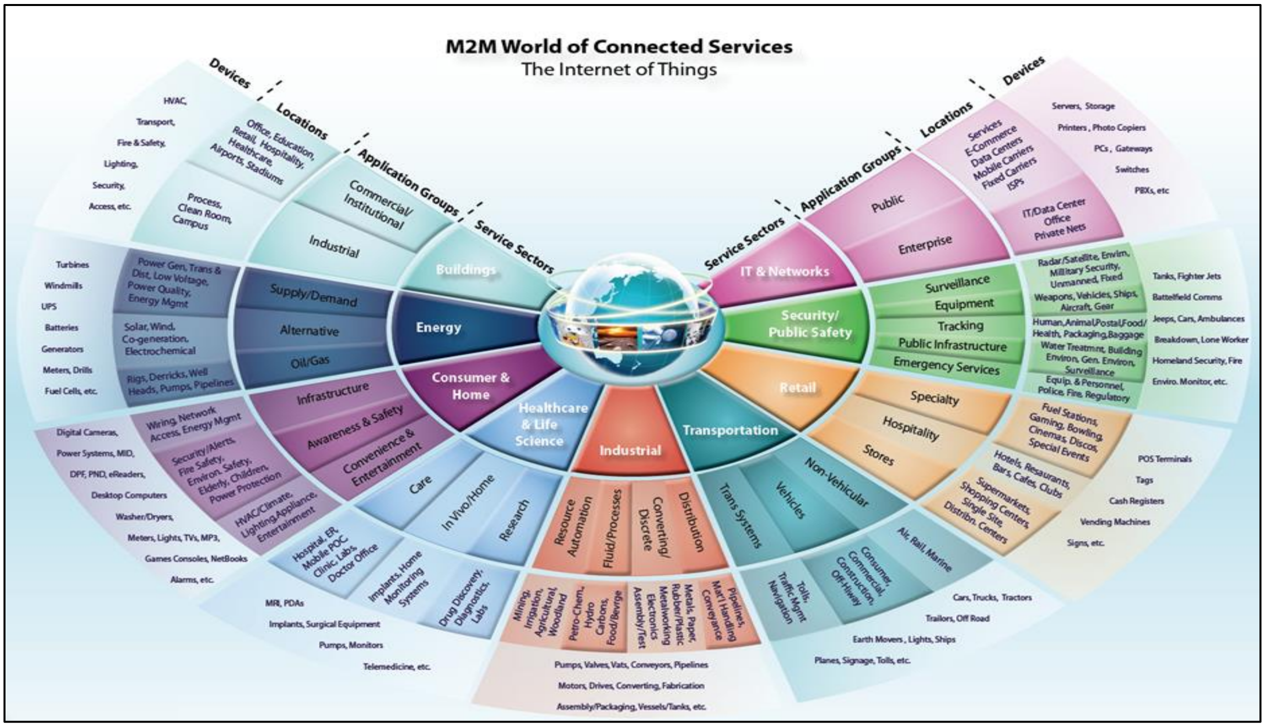

M2M World of Connected Services
The Internet of Things
Devices
Locations
Application Groups
Service Sectors
Service Sectors
Application Groups
Locations
Devices
Buildings
Commercial/ Institutional
Industrial
Office, Education, Retail, Hospitality, Healthcare, Airports, Stadiums
Process, Clean Room, Campus
HVAC, Transport, Fire & Safety, Lighting, Security, Access, etc.
Energy
Supply/Demand
Alternative
Oil/Gas
Power Gen, Trans & Dist, Low Voltage, Power Quality, Energy Mgmt
Solar, Wind, Co-generation, Electrochemical
Rigs, Derricks, Well Heads, Pumps, Pipelines
Turbines
Windmills
UPS
Batteries
Generators
Meters, Drills
Fuel Cells, etc.
Consumer & Home
Infrastructure
Awareness & Safety
Convenience & Entertainment
Wiring, Network Access, Energy Mgmt
Security/Alerts, Fire Safety, Environ. Safety, Elderly, Children, Power Protection
HVAC/Climate, Lighting,Appliance, Entertainment
Digital Cameras, Power Systems, MID, DPF, PND, eReaders, Desktop Computers Washer/Dryers, Meters, Lights, TVs, MP3, Games Consoles, NetBooks Alarms, etc.
Healthcare & Life Science
Care
In Vivo/Home
Research
Hospital, ER, Mobile POC Clinic, Labs, Doctor Office
Implants, Home Monitoring Systems
Drug Discovery, Diagnostics, Labs
MRI, PDAs
Implants, Surgical Equipment
Pumps, Monitors
Telemedicine, etc.
Industrial
Resource Automation
Fluid/Processes
Converting/ Discrete
Distribution
Mining, Irrigation, Agricultural, Woodland
Petro-Chem, Hydro Carbons, Food/Bevrge
Metals, Paper, Rubber/Plastic Metalworking Electronics Assembly/Test
Pipelines, Mat'l Handling Conveyance
Pumps, Valves, Vats, Conveyors, Pipelines
Motors, Drives, Converting, Fabrication
Assembly/Packaging, Vessels/Tanks, etc.
Transportation
Trans Systems
Vehicles
Non-Vehicular
Tolls, Traffic Mgmt Navigation
Consumer, Commercial, Construction, Off-Hiway
Air, Rail, Marine
Cars, Trucks, Tractors
Trailors, Off Road
Earth Movers , Lights, Ships
Planes, Signage, Tolls, etc.
Retail
Specialty
Hospitality
Stores
Fuel Stations, Gaming, Bowling, Cinemas, Discos, Special Events
Hotels, Resaurants, Bars, Cafes, Clubs
Supermarkets, Shopping Centers, Single Site, Distribn. Centers
POS Terminals
Tags
Cash Registers
Vending Machines
Signs, etc.
Security/ Public Safety
Surveillance
Equipment
Tracking
Public Infrastructure
Emergency Services
Radar/Satellite, Envirn. Military Security, Unmanned, Fixed
Weapons, Vehicles, Ships, Aircraft, Gear
Human,Animal,Postal,Food/ Health, Packaging,Baggage
Water Treatmnt, Building Environ, Gen. Environ, Surveillance
Equip. & Personnel, Police, Fire, Regulatory
Tanks, Fighter Jets
Battelfield Comms
Jeeps, Cars, Ambulances
Breakdown, Lone Worker
Homeland Security, Fire
Enviro. Monitor, etc.
IT & Networks
Public
Enterprise
Services E-Commerce Data Centers Mobile Carriers Fixed Carriers ISPs
IT/Data Center Office Private Nets
Servers, Storage
Printers , Photo Copiers
PCs , Gateways
Switches
PBXs, etc

Figure 1.10 – M2M/IoT Sector Map
Source: Beecham Research

Because of its advanced functionalities, the IoT smart meter device can be considered as a tangible example to demonstrate how the IoT technology is in fact associated with each sector in the smart city.
This device may be used across all sectors and in particular in the energy sector. It can provide several intelligent functionalities: Emits direct readings; updates the readings frequently to achieve energy savings; allows remote on/off control of supply and/or power limitations; supports advance tariff systems; provides secure data communications and fraud detection and prevention.

Chapter two elaborates more on the technical aspects of the IoT technology, two case studies have been implemented to showcase the relevance of IoT in various sectors of the smart city.

1.3 - Cyber Security

"More connections to more devices mean more vulnerabilities. If you control the code, you control the world." *–Marc Goodman*

"All inventions have unintended consequences." *–Marc Goodman*

Smart cities are highly digitized cities by nature, characterized by large volumes of data stored digitally and large numbers of physical objects with online connection to the Internet. This can be used in a positive way, by contributing to societal goals, but it can also be abused by hackers and used probably for criminal purposes.

For smart city there are two main key security facets. The first is the security of network infrastructure and all connected devices, and the second is the security of the data produced, warehoused, pooled and shared across all smart city dimensions or industries.

Various city systems and services use complex network of heterogeneous devices that have been produced by different vendors. Due to the lack of global standardized and efficient testing security strategy for smart cities, most devices are assumed vulnerable and prone to being compromised by hackers. Any networked device which relies on software is considered as an entry point whether it is from a wired or wireless interface, and then the whole network that is, the aggregation of the entire interconnected computer systems and devices becomes vulnerable to cyber-attacks.
Due to the lack of global standardized and efficient testing security strategy for smart cities, most devices are assumed vulnerable and prone to being compromised by hackers. Any networked device which relies on software is considered as an entry point whether it is from a wired or wireless interface, and then the

whole network that is, the aggregation of the entire interconnected computer systems and devices becomes vulnerable to cyber-attacks.

For smart city there are two main key security facets. The first is the security of network infrastructure along with all its connected devices and the second is the security of all installed software, the data produced, warehoused, pooled and shared across all smart city dimensions or industries. Smart devices are mainly relying on software thus; they are prone to being hacked. Whether it is from a wired or wireless interface, the whole network that is, the aggregation of the whole interconnected computer systems and devices becomes vulnerable to cyber-attacks.

The number of potential entry point multiplies across the smart city network, and cyber-attacks can be performed locally as well as remotely.

Cyber-attacks tend to disrupt, paralyze, destroy, and deceive both the interconnected computer systems and the data in transit or at rest. Generally, cyber-attacks target the three security services as they are known by the Confidentiality, Integrity, and Availability (CIA) triad. Confidentiality attacks seek to reveal confidential information as decrypting or guessing user password, credit cards credentials, and any other critical personal or industrial/business type information.

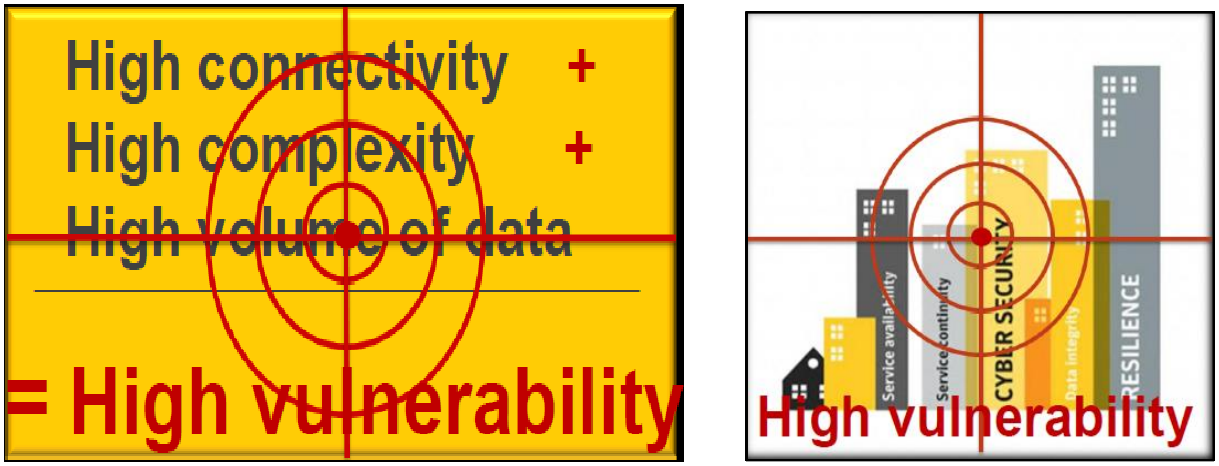


Figure 1.11 – Cause of High Vulnerabilities
Source: Symantec, http://bit.ly/106cTM8

Integrity attacks tend to intrude the computer or device to modify and delete critical information such as creating/erasing administrator account, or installing new programs as for example the stuxnet malware, etc…; however, the availability attacks are considered the most dangerous as they paralyze or put down the whole network and services for instance, the recent Distributed Denial of Service (DDoS) attack Dyn, sprang on Friday, October 21, 2016, is considered one of the biggest distributed denial of service attacks ever launched. The attack affected the availability of major internet services [28].

In addition to the three security services (CIA) triad, it should be acknowledged that attackers will always look for the weakest link; thus, it is not unlikely that attack vectors include not only technology and application, but also employees. Therefore, it is not sufficient that technology and application are designed from the ground up to be secure, it is also necessary that employees are aware of cyber security threats and well trained to act properly.

1.3.1 - Smart City Cybersecurity Challenges

In today's digital world, smart city's cybersecurity challenges open up new vulnerabilities (weaknesses or holes in security systems) and present an opaque universe of threats that have escalating and catastrophic effects as one vulnerable device can lead to other vulnerable devices.
There are many smart city cybersecurity challenges [28, 29], including, but not limited to:

Insecure Hardware**:** Smart cities or smart buildings sensors are insecure due to lack of standardization of IoT devices. most of the sensors use an unencrypted link to communicate. Intruders can hack them and feed fake data, causing signal failures, or system shut downs, etc...
Larger Attack Surface: A network of networks has made data accessible everywhere and any time. The number of potential entry points is multiplied in Smart Cities. By compromising a single device, it is possible to attack the entire system or network.
Bandwidth Consumption: Thousands of sensors, or actuators, trying to communicate to a single server can bring down the server. The bandwidth consumption from billions of devices will put a strain on the spectrum of other wireless communications, radio, television, emergency services, etc…
Application Risk*:* While apps accelerate the integration of mobile devices within our daily lives, they also increase the risk of supporting Bring Your Own device (BYOD) in corporate environments as organizations allow employees to bring their own devices to access work-related data. This may open the door in front of malicious app to propagate and infect the organization network.
Simple Bugs with Huge Impact: A simple software bug can have huge impact on the smart city network.
Blind Trusting Device Vendors: Organizations are blindly trusting vendors and buying technology without requiring any security testing or security benchmarking information requirements. This must be addressed to stop vulnerable and insecure technologies from being deployed into production.

1.3.2 - Internet of Things Security and Challenges

Smart cities can be seen as an application domain of IoT. “Internet of Things”, and “Machine to Machine” (M2M) are considered the pillars of the smart city technologies and enabler of the key term “smart” when used in conjunction with strict physical and logical security measures. In fact IoT is about smart objects. For an object to be ‘smart’ it must have three properties:

- An Identity (to be uniquely identifiable – via iPv6)
- A communication mechanism (i.e. a radio) and
- A set of sensors / actuators

There are two essential branches in IoT: industrial and consumer. The industrial IoT technology is used in the context of manufacturing, utilities, energy, infrastructure, and any other field, for the purpose of adding value to businesses whereas consumer IoT technology is used at an individual context which reflects entertainment devices, wearables, smart appliances, which are in their embryonic stage of development. Figure 1.12 below from Robeco Trends Investings shows the industrial IoT system where layers with IoT content are labeled with ‘IoT’ tag.

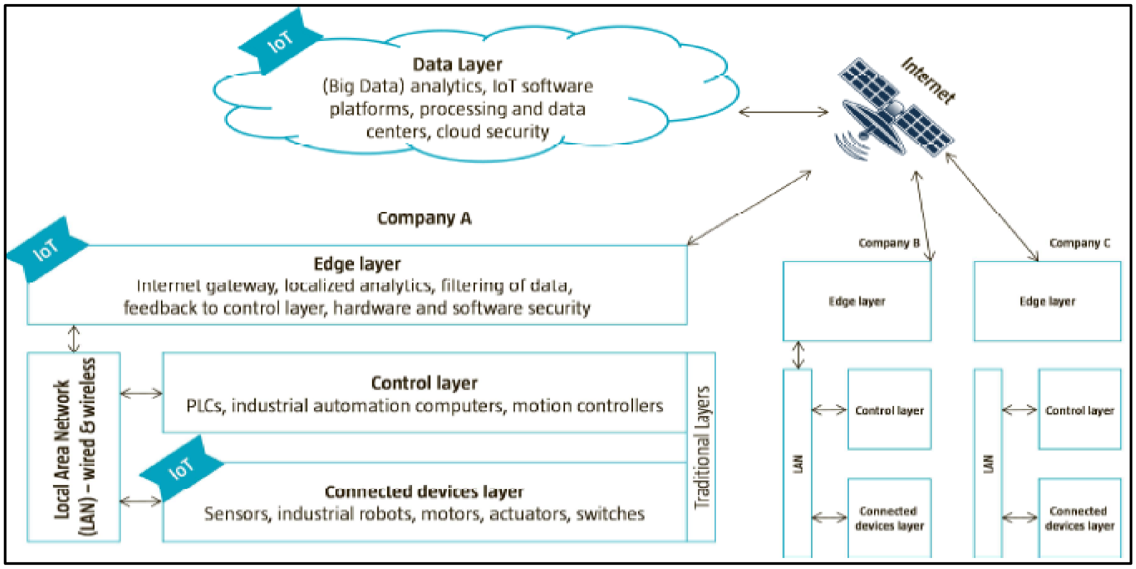

IoT
Data Layer
(Big Data) analytics, IoT software platforms, processing and data centers, cloud security
Internet
Company A
IoT
Edge layer
Internet gateway, localized analytics, filtering of data, feedback to control layer, hardware and software security
Local Area Network (LAN) – wired & wireless
Control layer
PLCs, industrial automation computers, motion controllers
IoT
Connected devices layer
Sensors, industrial robots, motors, actuators, switches
Traditional Layers
Company B
Edge layer
LAN
Control layer
Connected devices layer
Company C
Edge layer
LAN
Control layer
Connected devices layer

Figure 1.12 – Industrial IoT System
Source: Robeco Trend Investing

Securing an IoT system, whether it is industrial or consumer, is considered a big challenge because of its multiple points of vulnerability. When it comes to vulnerabilities with IoT, expect them. Vulnerabilities lead to attacks. The attack surface in an IoT system is expanded to the extent that vulnerabilities can be found in: a) the IoT device itself; b) the embedded software and data residing in it; c) the communication channels; d) the data aggregation platform, and e) data centers.

Figure 1.13 below shows the horizontal spectrum of the expanded attack surface of an IoT system.

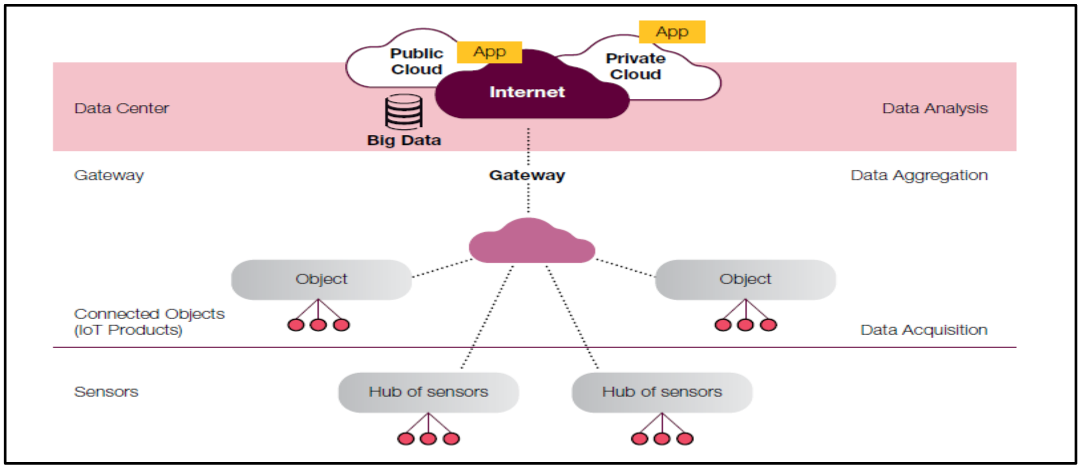
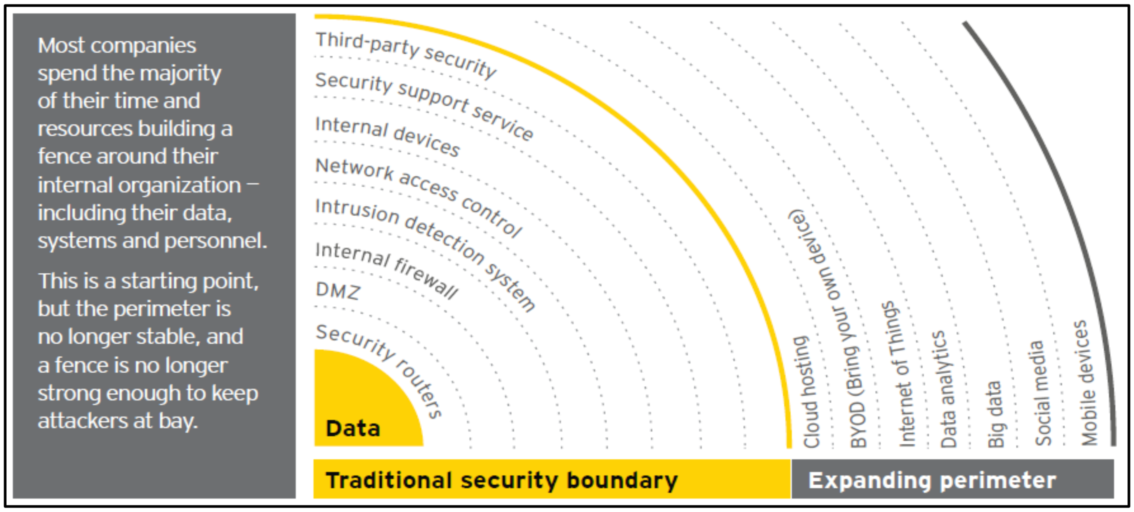


Figure 1.13 – Attack Surface of an IoT System
Source: Capgemini Cyber Security Service Line

Traditional security methods focus on establishing fences around an information system and then implementing security within each fence. These longstanding techniques do not readily scale to handle complex IoT systems situations, where traditional security methods and techniques of individual systems may undermine the security of the whole. Security Routers, DMZ, Internal firewall and so on, need further updates and patches as new smart non-secure systems emerge.



Most companies spend the majority of their time and resources building a fence around their internal organization – including their data, systems and personnel.

This is a starting point, but the perimeter is no longer stable, and a fence is no longer strong enough to keep attackers at bay.

Figure 1.14 – Traditional Security Boundaries and the Expanding Perimeters of New IoT Systems
Source: The Internet of Things in insurance: shaping the right strategy, managing the right risks

Figure 1.14 shows that the traditional security boundaries are no longer feasible and a fence is no longer strong enough to keep attackers at bay due to the expanding security requirements for new IoT systems.

The challenges in security are intensified by the IoT paradigm. It encompasses a wide range of tasks, including embedding keying material during the manufacturing process of the device, provisioning of new keying material during operation, establishing access control policies for access to networks and services, processes for secure software development, use of hardware security modules to protect keys against tampering, software update management, and development and selection of efficient cryptographic primitives [30]. The security of the IoT product is a key element of the overall security of an IoT system.

As a matter of fact, we can deduce two types of security challenges for IoT systems: Non-technical and technical. Non-technical challenges focus on the security policy adopted when producing, installing and using this IoT system by customers. Therefore, the focus on security can get lost if organizations rush to launch their IoT products, prioritizing speed-to-market over security. For most organizations, security is not the core focus of the IoT product development process. Securing a product from cyber-attacks is a critical element of the product development process in an IoT world. Michael Murray, the Director of GE Healthcare's Cyber Security Consulting and Assessment division, highlights this when he says: "It's all about building these sensitive medical systems and devices with cyber security in mind, rather than as an afterthought". Survey [31] showed that only 48% of companies focus on securing their IoT products from the beginning of the product development phase. In addition, only 36% are working towards modifying their IoT development process to focus more on security from the earliest stages of product design. Also to prevent cyber-attacks, organizations must ensure that they educate consumers about the correct security procedures to be followed while using an IoT system.

Technical IoT challenges reflect the vulnerabilities and threats resulting from the considerations for the redesign of networks to IPv6 implementations with regards to IoT that may come as more demand for traditional IPv4 addresses, changes on how bandwidth as consumed, quality of service, and prioritizing network traffic through new designs. And further, the redesign of networks may also take into account of how firewalls and IDS/IPS may handle IoT traffic when considering IPv6. [32]

The IoT system may inherit some "well-known" vulnerabilities from the traditional non-IoT system. It may be more often to find default, weak, and hardcoded credentials (usernames passwords) within IoT devices. The issue of upgrading firmware to counter vulnerabilities may be dependent upon how devices are designed during development; issues may occur that upgrading may break functionality. For this reason, vendors may hesitate or refuse to render support in product lines and make adjustments during the next design phase of projects. [33]

Certain IoT devices with embedded web services may also be subject to the same vulnerabilities that commonly plague web server platforms today. Also, with the premise that updating such functionality may run into the same issues such as buffer overflows are quite common vulnerabilities within technology infrastructure, same with IoT [34]. Devices may also at times use protocols that transmit credentials in the clear, in addition to having open ports, DOS/DDOS attacks may be the results in hacking or hijacking IoT devices on network(s). It is also possible that through misconfigurations of IoT devices, such “attacks” may be false positives and cause business disruption.

The issue of physical attacks of IoT devices may result in tampering to inject malicious code or make Hardware modifications to IoT devices. In addition, impersonating or counterfeiting devices may be issues when safeguards are not in place to protect physical security. [35]

Figure 1.15 from nebbiolotechnologies shows typical threats attributed to IoT devices related to the healthcare system.

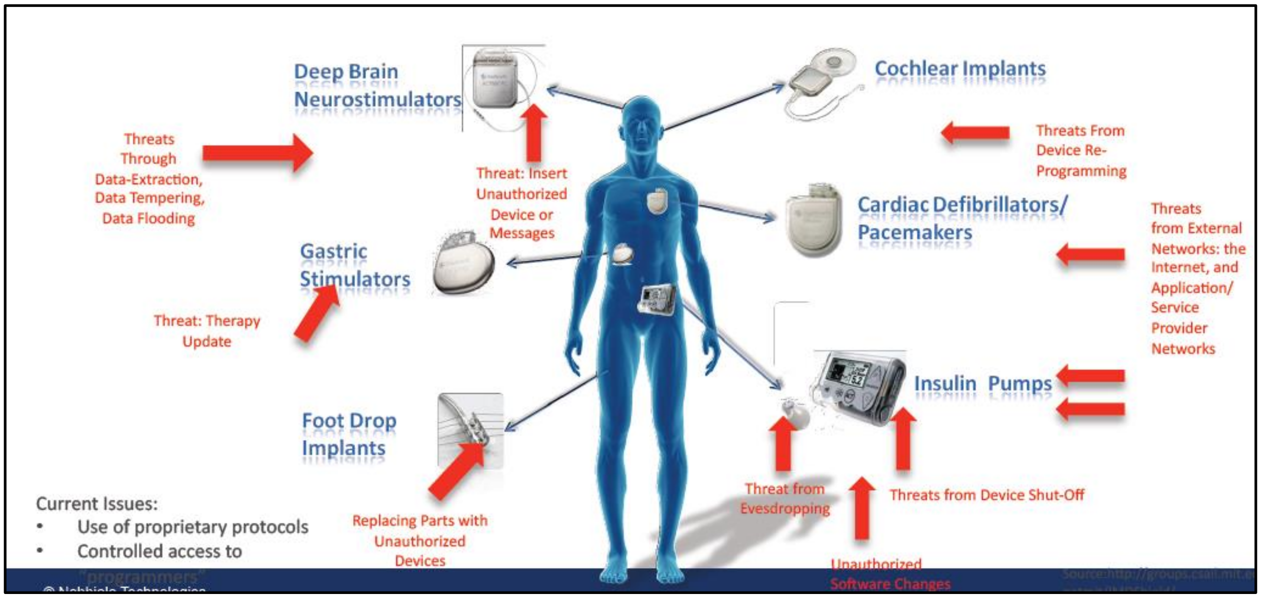

Deep Brain Neurostimulators
Cochlear Implants
Threats Through Data-Extraction, Data Tempering, Data Flooding
Threat: Insert Unauthorized Device or Messages
Threats From Device Re-Programming
Cardiac Defibrillators/ Pacemakers
Threats from External Networks: the Internet, and Application/ Service Provider Networks
Gastric Stimulators
Threat: Therapy Update
Insulin Pumps
Foot Drop Implants
Current Issues:
• Use of proprietary protocols
• Controlled access to "programmers"
Replacing Parts with Unauthorized Devices
Threat from Evesdropping
Threats from Device Shut-Off
Unauthorized Software Changes
© Nebbiolo Technologies
Source:http://groups.csail.mit.e

Figure 1.15 – End Point Security in IoT
Source: nebbiolotechnologies, 2016

Infiltration through non-traditional communication protocols such as Bluetooth, Zigbee, Zwave, Sigfox, NFC, 6LowPAN, and other types of non-traditional wireless communication outside of Wifi Communication protocols.

Ideally, various security protocols have already been standardized, and adapting them to cater for the required security functionalities for use in IoT would be beneficial. Such a standardized protocol when deployed on IoT devices, they can interoperate more easily with existing Internet infrastructure and services. Conversely, designing a completely new security protocol for IoT seems like reinventing the wheel, and new wave of vulnerabilities will be merging and difficult to circumvent [30].

The following table summarizes some major attack surface and its vulnerabilities in the IoT cyber-security ecosystem.

Table 1.2 – IoT Attack Surface and its Vulnerabilities
Source: OWASP

Attack Surface	*Vulnerability*
Ecosystem Access Control	Implicit trust between components Enrolment security Decommissioning system Lost access procedures
Local Data Storage	Unencrypted data Data encrypted with discovered keys Lack of data integrity checks
Device Memory	Cleartext usernames Cleartext passwords Third-party credentials Encryption keys
Third-party Backend APIs	Unencrypted PII sent Encrypted PII sent Device information leaked Location leaked
Device Physical Interfaces	Firmware extraction User CLI Admin CLI Privilege escalation Reset to insecure state Removal of storage media
Vendor Backend APIs	Inherent trust of cloud or mobile application

	Weak authentication Weak access controls Injection attacks
Device Web Interface	SQL injection Cross-site scripting Cross-site Request Forgery Username enumeration Weak passwords Account lockout Known default credentials
Update Mechanism	Update sent without encryption Updates not signed Update location writable Update verification Malicious update Missing update mechanism No manual update mechanism
Device Firmware	Hardcoded credentials Sensitive information disclosure Sensitive URL disclosure Encryption keys Firmware version display and/or last update date
Ecosystem Communication	Health checks Heartbeats Ecosystem commands Deprovisioning Pushing updates
Device Network Services	Information disclosure User CLI Administrative CLI Injection Denial of Service Unencrypted Services Poorly implemented encryption Test/Development Services Buffer Overflow UPnP Vulnerable UDP Services DoS
Mobile Application	Implicitly trusted by device or cloud Username enumeration Account lockout Known default credentials Weak passwords Insecure data storage

	Transport encryption Insecure password recovery mechanism Two-factor authentication
Administrative Interface	SQL injection Cross-site scripting Cross-site Request Forgery Username enumeration Weak passwords Account lockout Known default credentials Security/encryption options Logging options Two-factor authentication Inability to wipe device
Cloud Web Interface	SQL injection Cross-site scripting Cross-site Request Forgery Username enumeration Weak passwords Account lockout Known default credentials Transport encryption Insecure password recovery mechanism Two-factor authentication
Network Traffic	LAN LAN to Internet Short range Non-standard

1.3.3 - Smart Building and its Security Concerns

A smart building integrates the different physical systems present in a building (such as Building Automation System (BAS) - HVAC & Energy Management, Lighting Control System, Fire & Life Safety Control Systems, Parking Guidance and Management Systems) in an intelligent manner to ensure that all the different systems in a building act together in an optimized and efficient manner [36].

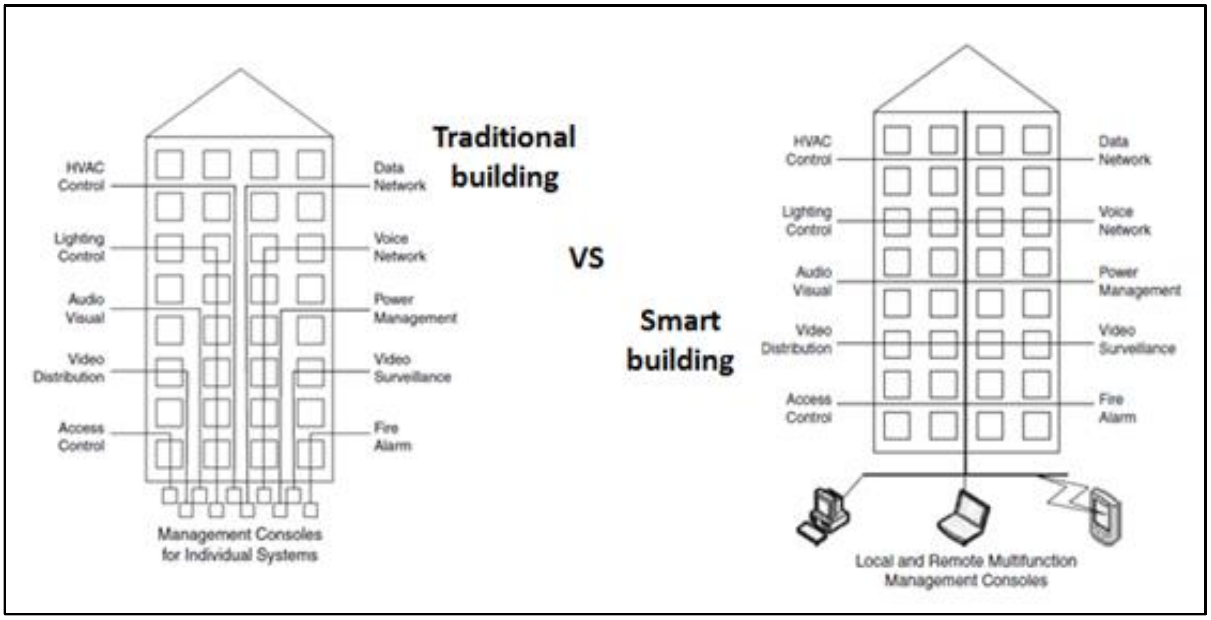


Figure 1.16 – Multiple Proprietary Building Traditional Buiding System vs Intergarted Smart Building Systems
Source: Reference [40]

Figure 1.16 shows a comparison between the traditional and the smart building systems where new IoT devices are installed and attached to an improved network infrastructure to form a unified smart building management system. Smart building management systems can improve building energy efficiency, reduce wastage, and ensure optimum usage of water with operational effectiveness and occupant satisfaction.

It is estimated that implementing smart building solutions [37] could save as much as 30 percent of water usage, 40 percent of energy usage, and could reduce overall building maintenance costs by 10-30 percent. It has been found that energy use in existing buildings can be reduced by up to 50% through simple retrofit programs.

One major concern is that building management systems are often considered as property services rather than IT services and cybersecurity is not an imperative issue. Consequently, the system is weakly configured and potential attacks may take place. Attacks include but not limited to default, weak, and hardcoded credentials, difficult to update firmware and OS, lack of vendor support for repairing vulnerabilities, clear text protocols and unnecessary open ports, physical theft and tampering.

The vulnerabilities of building management systems pose two main threats. The first is that if there are hacked building, operations could be disrupted and safety risks could be created. The second is that they provide a potential route for breaking into enterprise business systems and critical company data if they share the same network. [38]

1.4 - A Typical Threat Model for Public Transport in Smart City

To better acquire and understand the concept of cybersecurity in a smart city, it is worth considering the example of threat model for public transport in smart city provided in the reference [39] where authors tackled in depth the security concerns i.e, threats, vulnerabilities, risks and attacks of a typical public transport. In short, they explained the threat categories in the context of intelligent transport system in smart city taking into account Availability, Integrity, Authenticity, Confidentiality and Non-Repudiation/accountability threats. They also provided a simplified view of the ICT architecture of smart cities with all its corresponding functional layers namely: smart processing data aggregation, data processing, data transmission network and field components. Authors explained as well the threats that are coming out of intentional attacks such as Eavesdropping, Theft , Tampering/alteration, Unauthorized use/access, Distributed Denial of Service (DDoS), Loss of Reputation. (For more information refer to reference [39])

1.5 - Thesis Statement and Proposed Solution

Paying attention to security should be a part of the smart city's everyday activities. Where do we even start?

Several smart city security solutions have been currently suggested and expected to comply with basic security requirements to guarantee confidentiality, integrity, availability and authenticity of the information such as [29]: a) Strong cryptography to protect data, both at rest and in transit (wired and wireless); b) Authentication capabilities to enforce one-time passwords, certificates or biometric-based authentication; c) Authorization capabilities to impose proper permissions before performing any actions; d) Automatic and secure update of software and firmware to patch security vulnerabilities; e) Auditing, alerting, and logging capabilities to be able to track all users' activities; f) Anti-tampering mechanism to prevent devices' tampering by unauthorized sources; g) No backdoor/undocumented/hardcoded accounts to prevent and disable hardcoded vendor's passwords that can't be changed.

However, those solutions are considered static defensive mechanisms. They cannot ensure a state of readiness as they fail to prevent: 1) traffic analysis; 2) re-transmitted packets; 3) delayed packets; 4) packets from being jammed; and 5) malicious insiders and captured nodes.

We are aiming at designing and implementing a cyber-threat intelligence strategy to support strategic decisions. Early warning and detection of breaches are decisive to being in a state of readiness, meaning that the emphasis of cybersecurity has changed to threat intelligence. No organization can ever predict or prevent all (or even most) attacks; but they can reduce their attractiveness as a target, increase their resilience and limit damage from any given attack [28].

Therefore, our work identifies the necessary precursor steps through a twofold solution:

1- ***A predictive and proactive approach***: It is a novel framework used to analyze and evaluate quantitatively the vulnerabilities associated to the wired and wireless network in a smart city. It is a probabilistic approach and uses the Markov Chain Model to determine the likelihood of states transitions sequences of vulnerabilities attributed to attack paths using the Common Vulnerability Scoring System (CVSS) scores. The quantitative results have been investigated and examined to prioritize and detect the highest vulnerability severity level of a particular attack path in the attacks graph of the network. (See the published research paper in chapter three)

2- ***A defensive or self-protective approach:*** This framework mitigates the zero-day availability attacks Based on Identification, Heuristics and Load Balancer in a reasonable time frame. The Distributed Denial of Service (DDOS) attacks are considered one of the most severe availability attacks that could paralyze the smart city's network and cause complete black out. Traditional security mechanisms like firewall, IPS and IDS could not prevent such type of zero day attacks. To reveal the severity of such attack, it is imperative to consider the case of the Dyn DDoS attack which was one of the biggest distributed denial of service attacks ever launched. The Dyn attack, which took place on 21st October of 2016, is one of the largest data breaches in history. The source of the attack was the Mirai botnet. This botnet is unlike other botnets, consisting of so called Internet-of-Things (IoT) devices such as internet protocol (IP) cameras, printers, digital video recorders. A few major US websites including Paypal, Spotify, Twitter and Amazon faced connectivity issues. The various other web services of companies such as BankWest, HSBC and Ticketmaster were also affected [8]. According to Bitsight [8], approximately 8% of the Dyn DNS customer base terminated their contract after the attack. (See the research paper in chapter three)

1.6 - Thesis Contributions

The main contributions of this research are:

1- Demonstrating how predictive analytics and metric capabilities are the keys to boosting the security intelligence of smart cities

2- Providing a smart city proactive and not reactive security approach; preventative and not remedial. A framework which seeks to anticipate privacy concerns rather than seeking to resolve privacy infractions once they have incurred

3- Creating a novel predictive framework to analyze and evaluate quantitatively the vulnerabilities associated to the wired and wireless network in a smart city. It is a probabilistic approach and uses Markov Chain Model to determine the likelihood of states transitions sequences of vulnerabilities attributed to vulnerable attack paths using the Common Vulnerability Scoring System (CVSS) scores. We then investigate and examine the quantitative results to prioritize and detect the highest vulnerability severity level of a particular attack path in the attacks graph of the network.

4- Creating another novel defensive framework to mitigate Denial of Service Attacks Based on Identification, Heuristics and Load Balancer.

7 - Conclusion

Conventional security mechanisms have proven that they cannot provide efficient state of readiness of operating a smart city. They have failed to prevent severe attacks related to Denial of Service attacks which had paralyzed in many occasions the essential functionalities and operations. Therefore, two solutions have been proposed. The first is predictive in nature, whereas, the second is defensive and self-protective.

By applying the aforementioned solutions, we would have been contributing to boosting the security intelligence of a smart city, anticipating privacy, and ensuring steady services availability. The next chapter proposes two efficient cyber security mechanisms to protect a smart city network. Security concepts, indicators and metrics used to implement the solutions are examined in depth as they are considered the corner stone that is used to measure the vulnerability severity level.

Chapter 2

Proposed Cyber Security Solutions for Smart City

2.1 - Statement of the Problem

Regardless of how the smart city's architecture looks like and regardless of what the key performance indicators of the three core domains of sustainability (social, economic, environmental) that make a city smart are, one essential characteristic to recognize is that the information and communications technology ecosystem serves as a foundational base to all smart city dimensions. It ultimately aims at improving the economic and the political efficiency leading to social, cultural, and urban development. A resilient city must be able to tolerate disruptions, anticipate, and recover form disturbance [41]. As such, choosing the right technology is far from straightforward mission as it integrates multiple technological trends like Ubiquitous computing, Networking, Open data, Internet of Things (IoT), Big data, Geographic information system (GIS), Cloud computing, and more. When all combined together, they play a crucial role in achieving successful smart city [42].

Marc Goodman said: "More connections to more devices means more vulnerabilities. If you control the code, you control the world [43]."

In a smart city, almost all critical services like electricity, transportation, water, communication among others are increasingly becoming interconnected and dependent on smart technology. With the use of Internet of Thongs (IoT) and disruptive technologies, large volumes of data are stored digitally and a plethora of physical objects require online connection to the Internet and the role of humans dwindles. This may lead to many possible security breaches which can be exploited for criminal purposes [44]. Thus, the need for cyber security to protect the network, the devices, the information exchange, and the privacy of citizens' data becomes a must.

To be able to understand the types of attacks that smart cities are currently facing, it is imperious to give some example of cyber-attacks which had happened not long ago and had caused massive impact on some governments, infrastructure, citizens, and society as a whole. The following selected examples are inspired from the Trend Micro Research paper [45]. These examples motivated us to implement a smart cyber security solution which is based on studying and analyzing the vulnerabilities of the entire smart city's network and its associated devices and predicting potential critical attack paths, and hence creating a secure model to avoid real-time attacks manifestation.

In 2016, The San Francisco Transportation Agency was hit by a ransomware attack that encrypted 2,000 of its computers, allowing passengers to ride for free until the problem was fixed [46].

In November 2013, the Bay Area Rapid Transit (BART) closed down due to a software glitch after a server upgrade. People were stuck in trains and track switching had to be done manually [47].

A solar software and analytics company issue patches for its power meters to protect them against command injection and remote code execution vulnerabilities [48].

Gogoro partnered with Coup to provide 200 electric scooters in Berlin [49]. Preregistered members use a mobile app to locate the nearest available scooters. Once payment is made, scooters are unlocked and made ready for use. Yugo provides a similar service in Barcelona. Previous research [50] suggested vulnerabilities in Gogoro's Bluetooth stack. These have since been fixed but new bugs can be abused to give attackers free rides. Attackers use ransomware attacks to disrupt vendors' services.

Smart traffic lights can also collect data for analytical use. Traffic lights that collect tons of data can be hijacked and the stolen data monetized. Hackers can also sell "always green" services by abusing open and unencrypted radio signals to control traffic lights. Newer traffic lights connected via Long-Term Evolution (LTE) may be prone to a downgrade attack that can cause city-wide chaos as well [51].

Many websites like Opentopia and Insecam list public cameras using public IP addresses with no or default passwords. Hackers can take advantage of the huge number of surveillance equipment. IP cameras have been targeted by malware like Mirai in 2016 [52].

As we have seen so far, threats and vulnerabilities (holes or weaknesses in the ICT ecosystem) appear to be multifaceted and directed against information/data, applications and their underlying protocols, devices, network (OSI/TCP layers) infrastructure, and may be organizational and managerial structure relevant to the entire smart city framework. However, the scope of this research is to tackle only the cyber security concerns related to the technical ICT aspects which form the corner stone of the smart city's network.

Several cyber security measures are being implemented. However, measures are very diverse as there are neither consistent guidelines, nor widely accepted cyber security standards that are aligned with smart city network needs. Security standards are still in their embryonic stage.

The current cyber security approach used nowadays has been based on model that alleviates attacks targeting the security services: Availability, Integrity, Authenticity, Confidentiality and Non-repudiation/accountability of the stored data and/or in transit. Nonetheless, these security services cannot prevent attacks like traffic analysis, re-transmitted packets, delayed packets, packets from being jammed, malicious insiders and captured nodes.

The above argument imposed a need for a robust predictive and preventive solution that would identify and apply the necessary precursor steps prior to any attack.

The proposed solution is twofold:

1- A predictive and proactive approach

It is a novel framework used to analyze and evaluate quantitatively the vulnerabilities associated to the wired and wireless network in a smart city. It is a predictive security approach in which suspicious activities, events and incidents can be predicted using a security indicator being the Common Vulnerability Scoring System (CVSS) which is used to measure the vulnerability severity level of potential attack paths and hence, enabling the security administrator to take appropriate security measures to patch the vulnerabilities before the breach occurs. (Refer to section four for detailed explanation)

2- A defensive or self-protective approach

This framework mitigates the zero-day availability attacks based on Identification, Heuristics and Load Balancer in a reasonable time frame. (Refer to section 2.4.3 for detailed explanation)

The above two approaches use the CVSS score as security indicator or metric. Such security indicators help to evaluate the strengths and weaknesses of smart city network. Investigating the level of severity of a certain attack path using the CVSS score help us determine the most and the least critical vulnerable areas. The determination of the vulnerability severity level assists the system administrator to set security measures priorities. Moreover, CVSS indicators act as monitoring tool to assess the security measures over a certain period of time and after several security actions have been implemented.

2.2 – Smart City and IoT Ecosystem Security

Since the smart city's network infrastructure depends heavily on the Internet of Things (IoT) architecture, it becomes necessary to study the attacks and analyze the threats and vulnerabilities pertaining to the whole IoT ecosystem.

The high level IoT layers consist of (see figure 2.1): The IoT device or thing (sensor, actuator/controller), the network, the application, the mobile, and the cloud (API or Web).

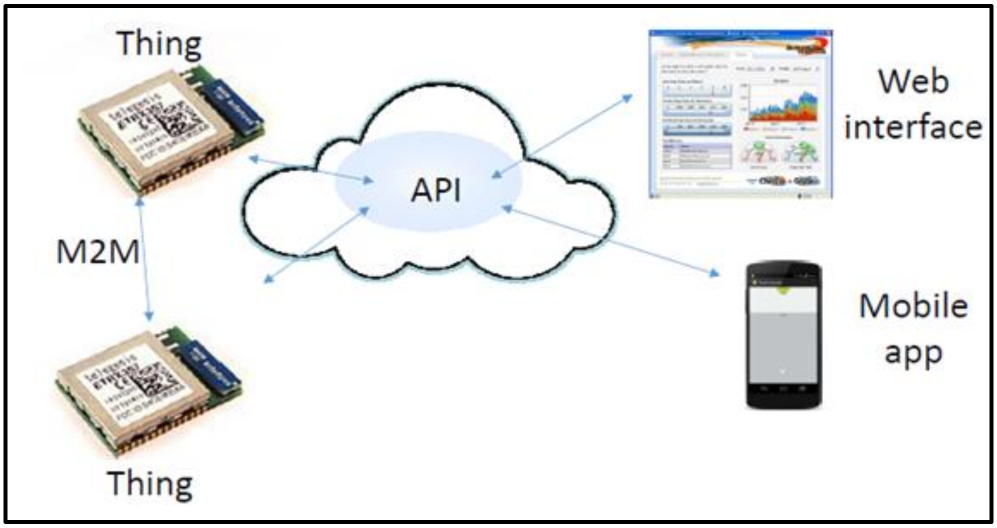
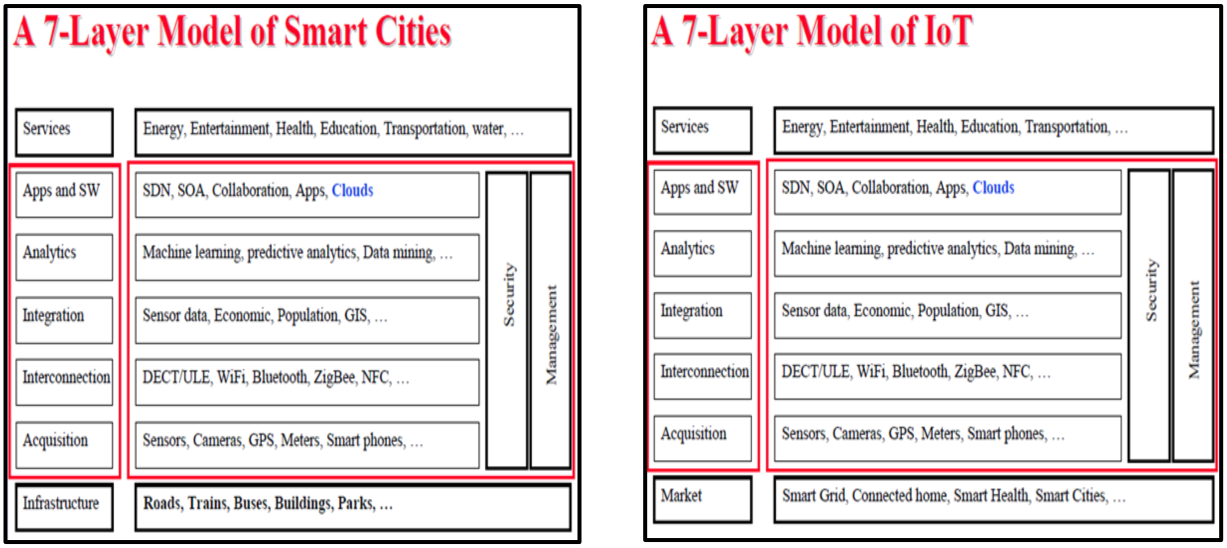


Figure 2.1 – IoT Layers Architecture.
Source: AppSec Labs

Two seven layer Models for the ICT architecture one for smart city and the other for IoT have been proposed by Saint Louis University in Washington showing the technologies involved in each layer. This model shows that security mechanisms must be integrated in each layer, for services to be delivered safely in the context of smart city in which it depends deeply on IoT technologies.


A 7-Layer Model of Smart Cities

Services	Energy, Entertainment, Health, Education, Transportation, water, …		
Apps and SW	SDN, SOA, Collaboration, Apps, Clouds	Security	Management
Analytics	Machine learning, predictive analytics, Data mining, …		
Integration	Sensor data, Economic, Population, GIS, …		
Interconnection	DECT/ULE, WiFi, Bluetooth, ZigBee, NFC, …		
Acquisition	Sensors, Cameras, GPS, Meters, Smart phones, …		
Infrastructure	Roads, Trains, Buses, Buildings, Parks, …		

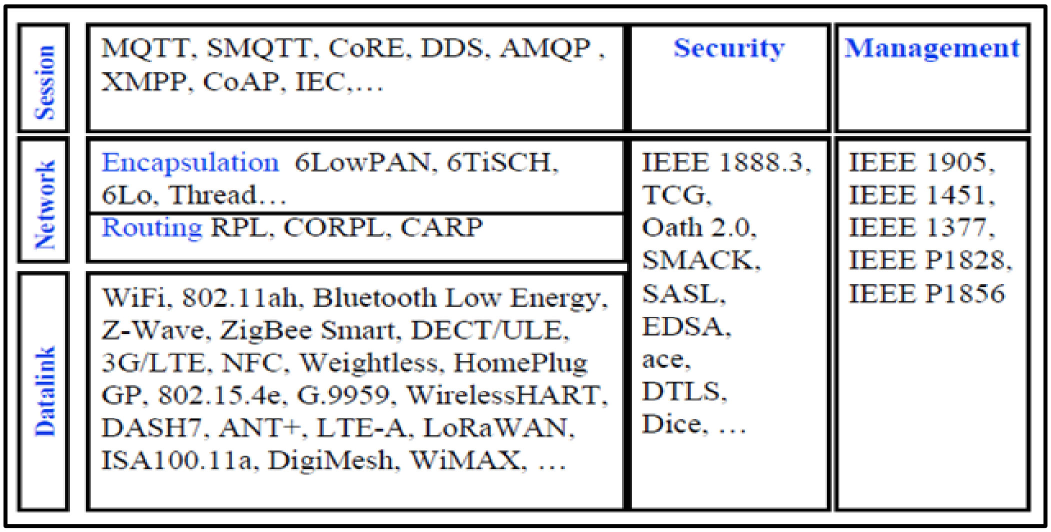
A 7-Layer Model of IoT

Services	Energy, Entertainment, Health, Education, Transportation, …		
Apps and SW	SDN, SOA, Collaboration, Apps, Clouds	Security	Management
Analytics	Machine learning, predictive analytics, Data mining, …		
Integration	Sensor data, Economic, Population, GIS, …		
Interconnection	DECT/ULE, WiFi, Bluetooth, ZigBee, NFC, …		
Acquisition	Sensors, Cameras, GPS, Meters, Smart phones, …		
Market	Smart Grid, Connected home, Smart Health, Smart Cities, …		

Figure 2.2 – Seven Layer Model for Smart City and for IOT
Source: Washington University in Saint Louis, Rai Jain, 2016

All layers in both models are similar. However, one difference is noted, the infrastructure layer in the smart city model becomes market layer in the IoT model. Thus, the smart grid, smart health, smart transportations, etc., that rely on IoT form the smart city dimensions as explained in chapter 1. The infrastructure layer which consists of trains, buses, buildings, parks, etc., becomes smart due IoT technology usage.

To be able to analyze the threats and vulnerabilities of the IoT technology we must first understand the underlying low level protocols (see figure 2.3) and how these protocols communicate among each other.



Session	MQTT, SMQTT, CoRE, DDS, AMQP , XMPP, CoAP, IEC,…	Security	Management
Network	Encapsulation 6LowPAN, 6TiSCH, 6Lo, Thread…	IEEE 1888.3, TCG, Oath 2.0, SMACK, SASL, EDSA, ace, DTLS, Dice, …	IEEE 1905, IEEE 1451, IEEE 1377, IEEE P1828, IEEE P1856
	Routing RPL, CORPL, CARP		
Datalink	WiFi, 802.11ah, Bluetooth Low Energy, Z-Wave, ZigBee Smart, DECT/ULE, 3G/LTE, NFC, Weightless, HomePlug GP, 802.15.4e, G.9959, WirelessHART, DASH7, ANT+, LTE-A, LoRaWAN, ISA100.11a, DigiMesh, WiMAX, …		

Figure 2.3 – IoT Protocols
Source: Washington University in Saint Louis, Rai Jain, 2016

For the purpose of demonstrating how the IoT protocols interact with each other, we have written a research paper which was accepted in the ICTO 2017, International Conference in Paris on March 16-17, 2017. This paper has been entitled ***"A Pervasive IoT Scheme to Vehicle over Speed Detection and Reporting Using MQTT Protocol"*** in which we solved a major problem to detect over speed without the use of Lidar, or Radar, or camera. Our main contributions in this paper were: a) Using IoT protocols to develop a new smart IoT solution which helps governmental authorities in supervising vehicle overs peed in a less costly manner; b) Issuing and collecting car tickets autonomously; c) Implementing a cost efficient and accurate solution for over speed road control. For more details about the IoT protocols implementation read the full paper provided ***in Appendix A.***

Today a numerous number of smart IoT devices is available and heavily used in smart homes, smart buildings, and smart grids. As an example, smart IoT devices might include: Thermostats, light bulbs, hubs, locks, TVs, webcams, home thermostats, remote power outlets, smoke detectors, sprinkler controllers, IP cameras, and so.

Usually these IoT devices use a back-end (see figure 2.1) cloud service to monitor usage or to allow users to remotely control them. Users can access the device through a mobile application or web portal. Connections to the internet are achieved through a central router, which may contain basic firewall filtering functionality. IoT devices may support several communication protocols like Powerline, Z-Wave, Zigbee, Bluetooth 4.0, and other radio frequency (RF) protocols.

As the use of connected IoT devices constantly increases, so do the security concerns. The number of smart techniques in which IoT devices can help our society is boundless, but unfortunately, they are susceptible to many of the same traditional types of attacks like Distributed Denial of Service Attacks (DDOS). This was demonstrated on September 2016 when attackers used the Mirai botnet to infect hundreds of thousands of vulnerable IoT devices from consumer and corporate environments to disrupt the operations of other devices and networks.

Conventional wireless security measures are not enough to touch the IoT threats as they provide security solutions focusing on defending network perimeter. Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS) solutions are decent enough at monitoring and preventing attacks coming over the network but they are feeble at protecting IoT related attacks as these devices might use non-conventional communication protocols as we stated earlier (Bluetooth, NFC, RFID, Z-Wave, ZigBee or 2G/3G/4G protocols). Protecting network perimeter is no longer enough. Enterprises must work on protecting their assets against threats covering the entire RF spectrum emerging on both legacy and tomorrow's IoT protocols.

2.2.1 - IoT Attack Surface

The IoT attack surface spans the entire IoT ecosystem (see figure 2.4): The Things or IoT devices themselves, the wireless access protocols (Zigbee, RFID…), the IoT gateway (the mobile device, …), the home LAN (Wifi, Ethernet,...), the IP Network (DNS, Routers, …), the higher layers protocols (MQTT, COAP, …), the cloud or fog services, the web interface, the life cycle management (booting, pairing, updating, …)

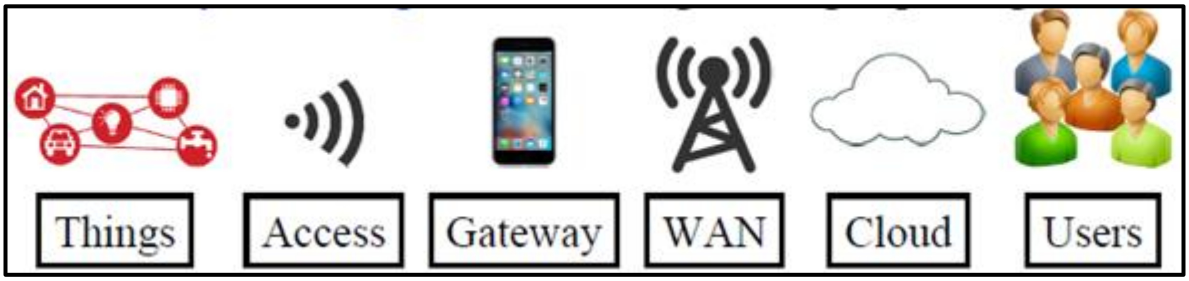


Figure 2.4 – IoT Attack Surface

The Open Web Application Security Project's (OWASP) List of Top Ten Internet of Things Vulnerabilities put together most of the concerns and attack vectors surrounding this category of devices:

- Insecure web interface
- Insufficient authentication/authorization
- Insecure network services
- Lack of transport encryption
- Privacy concerns
- Insecure cloud interface
- Insecure mobile interface
- Insufficient security configurability
- Insecure software/firmware
- Poor physical security

An Internet of Things Security study conducted by Craig Smith and Daniel Miessler, from Hewlett-Packard on June 2015 showed that 80% percent of IoT devices raised privacy concerns (See figure 2.5) which encompass collecting some form of personal information such as name, health information, credit card numbers, and those concerns are multiplied when cloud and mobile application services are added. For more information about ***privacy concerns vulnerabilities,*** refer to https://www.owasp.org/index.php /Top_10_2014-I5_Privacy_Concerns.

80% failed to require passwords of sufficient complexity and length due to insufficient authentication and authorization. Thus, most of the cloud and mobile components are allowing passwords such as "1234" or "123456". An attacker can use these vulnerabilities through several password recovery mechanisms to gain access to a device. For more information about ***insufficient authentication and authorization vulnerabilities,*** refer to https://www.owasp.org/index.php/Top_10_2014- I2_Insufficient_Authentication /Authorization.

70% of the devices failed to encrypt network services transmitting data via the local network and the Internet. For more information about ***lack of transport encryption vulnerabilities,*** refer to https://www.owasp.org/index.php/Top_10_2014-I4_Lack_of_Transport_Encryption.

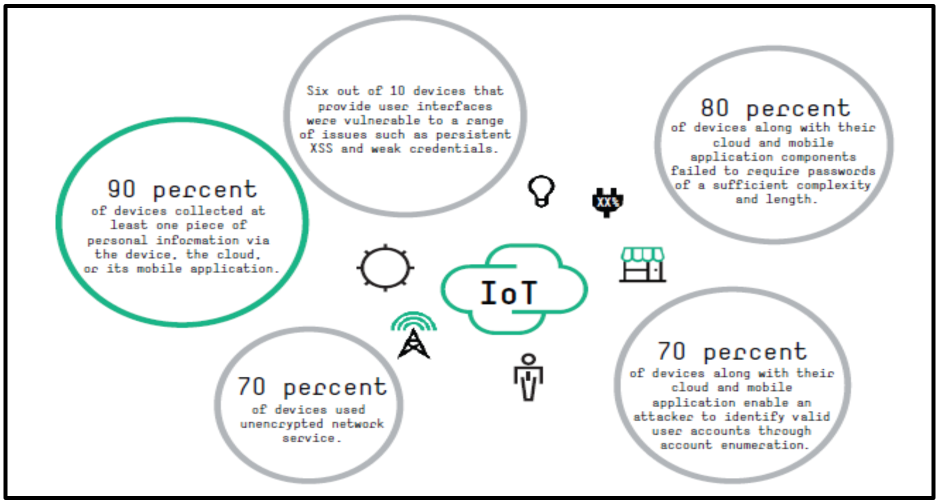

Six out of 10 devices that provide user interfaces were vulnerable to a range of issues such as persistent XSS and weak credentials.
80 percent of devices along with their cloud and mobile application components failed to require passwords of a sufficient complexity and length.
90 percent of devices collected at least one piece of personal information via the device, the cloud, or its mobile application.
XXS
IoT
70 percent of devices used unencrypted network service.
70 percent of devices along with their cloud and mobile application enable an attacker to identify valid user accounts through account enumeration.

Figure 2.5 – Security Concerns
Source: Hewlett Packard Enterprise, Internet of Things Research Study, 2015

60% raised security concerns with their user interfaces such as persistent cross site scripting, poor session management, and weak default credentials. For more details about ***web interface vulnerabilities*** refer to https://www.owasp.org/index.php/Top_10_2014-I1_Insecure_Web_Interface.

60% did not use encryption when downloading software updates. As such, some downloads were intercepted, extracted, and mounted as a file system in Linux® where the software could be viewed or modified. For more details about ***Insecure software and firmware,*** refer to https://www.owasp.org/index.php/Top_10_2014-I9_Insecure_Software/Firmware.

2.2.2- Typical IoT Network for a Smart City and its Security Considerations

Figure 2.6 shows a typical web based IoT network in which different IoT protocols are integrated with the existing communication infrastructures. A mapping between the unconstrained de-facto protocols for internet communication and their corresponding constrained IoT peers is well depicted. The XML and HTTP over IPv4 correspond to EXI and COAP/UPD over IPv6/6LoWPAN. For more details about the communication protocols refer to [53]. Well, our main concern here is not to explain the communication protocols but to reveal the vulnerabilities (security holes or weaknesses) and explain the potential attack vector associated to this network.

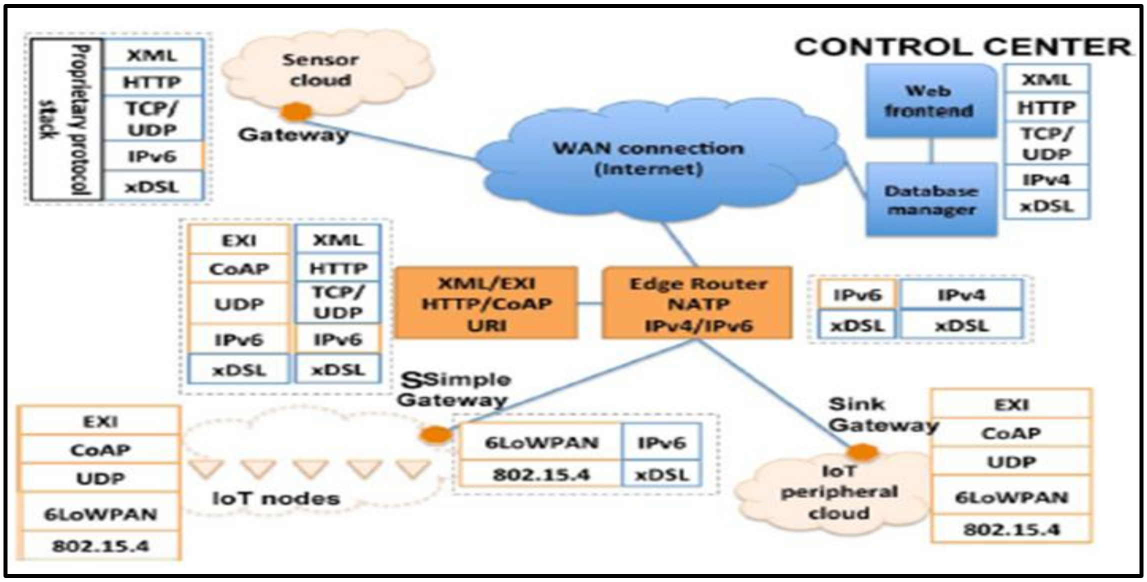


Figure 2.6 – Typical IoT Network for Smart City
Source: Reference [13]

A very vital issue has to be raised first. Why IPv6 and not IPv4? The Internet Assigned Number Authority (IANA), the international organization that assigns IP addresses at a global level, announced the exhaustion of IPv4 address blocks on February 2011. Also ARIN (North America's Regional Internet Registry) announced exhaustion in 2014. All new devices have to use the new IPv6 address system instead. Cisco predicts that by year 2019 there will be 3.9 billon internet users, 24 billion network devices, and 10.5 billion M2M devices all connected to the Internet.
A solution to this problem is offered by the IPv6 standard, which provides a 128-bit address field, making it possible to assign a unique IPv6 address to each node in the network [54].

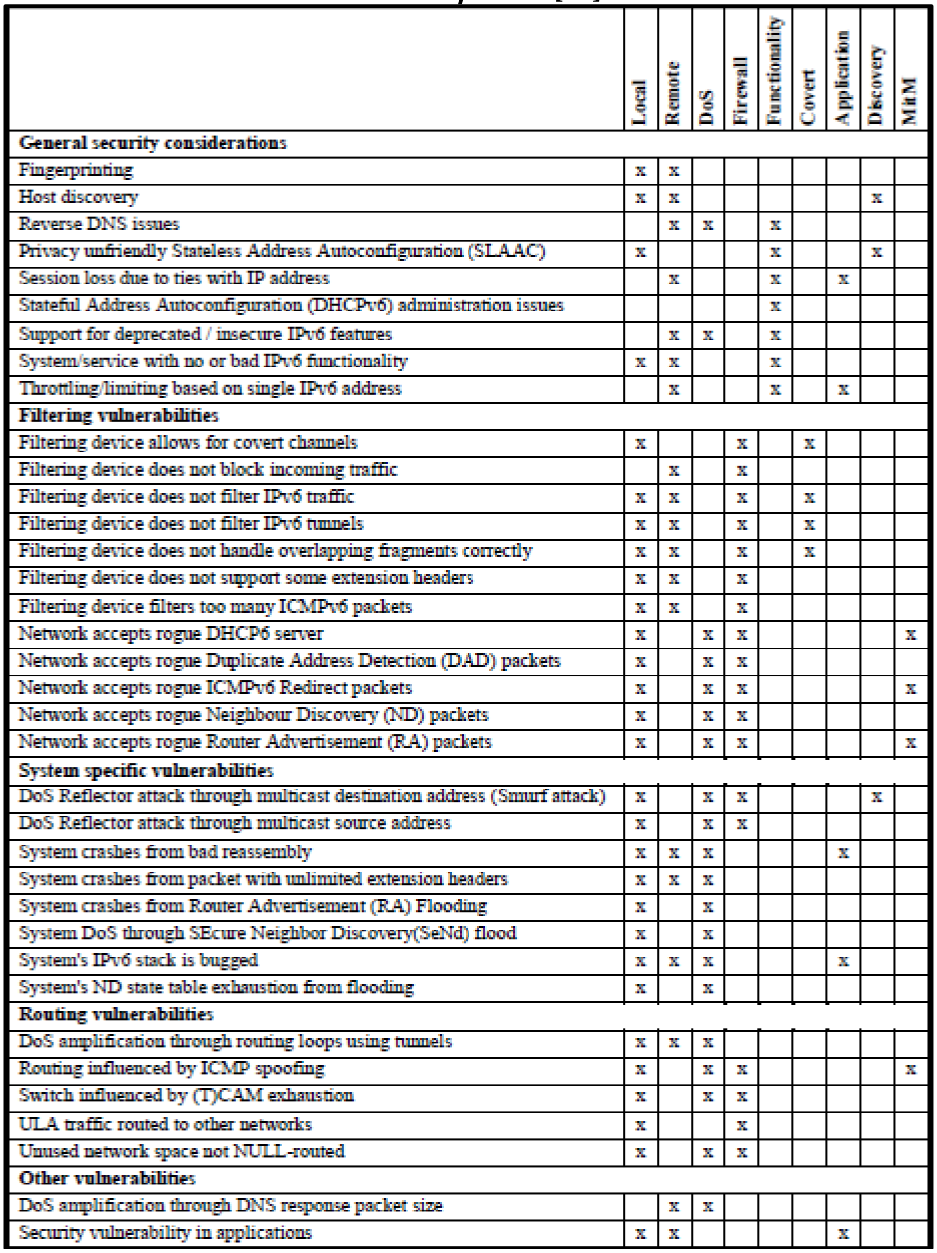
The security concerns about this IoT network architecture are boundless. Vulnerabilities span through the entire entry points in this network leading to a reasonable number of destructive attacks, namely: The Ipv6 attacks, Denial of Service attacks, web based attacks, Malware and Botnets. A summary of each attack and its conventional mitigation methods are provided below.

2.2.3 - IPv6 Vulnerabilities and its Attack Surface

All new IoT devices will be assigned IPv6 address to connect to Internet. Thus, anyone who has your IP address also has your MAC address; ICMPv6 is required for all networks and can’t be blocked since it replaces the ARP protocol in the IPv4 architecture; Many security appliances are not ready for IPv6, so it often bypasses them, for instance Torrents run over IPv6; Some VPN appliances are not ready, so IPv6 connections must bypass them; IPv6 is vulnerable to Packet Amplification attacks like Routing Header Zero, and Ping-pong.

The following table (Table 2.1) summarizes and displays the mappings between the IPv6 vulnerabilities and their corresponding categories or attack surface. Security administrators are obliged to use this table to test their IPv6 network against the below vulnerabilities [55].

Table 2.1 – Ipv6 Vulnerabilities
Source: reference [15]



	Local	Remote	DoS	Firewall	Functionality	Covert	Application	Discovery	MitM
General security considerations									
Fingerprinting	x	x							
Host discovery	x	x						x	
Reverse DNS issues		x	x		x				
Privacy unfriendly Stateless Address Autoconfiguration (SLAAC)	x				x			x	
Session loss due to ties with IP address		x			x		x		
Stateful Address Autoconfiguration (DHCPv6) administration issues					x				
Support for deprecated / insecure IPv6 features		x	x		x				
System/service with no or bad IPv6 functionality	x	x			x				
Throttling/limiting based on single IPv6 address		x			x		x		
Filtering vulnerabilities									
Filtering device allows for covert channels	x			x		x			
Filtering device does not block incoming traffic		x		x					
Filtering device does not filter IPv6 traffic	x	x		x		x			
Filtering device does not filter IPv6 tunnels	x	x		x		x			
Filtering device does not handle overlapping fragments correctly	x	x		x		x			
Filtering device does not support some extension headers	x	x		x					
Filtering device filters too many ICMPv6 packets	x	x		x					
Network accepts rogue DHCP6 server	x		x	x					x
Network accepts rogue Duplicate Address Detection (DAD) packets	x		x	x					
Network accepts rogue ICMPv6 Redirect packets	x		x	x					x
Network accepts rogue Neighbour Discovery (ND) packets	x		x	x					
Network accepts rogue Router Advertisement (RA) packets	x		x	x					x
System specific vulnerabilities									
DoS Reflector attack through multicast destination address (Smurf attack)	x		x	x				x	
DoS Reflector attack through multicast source address	x		x	x					
System crashes from bad reassembly	x	x	x				x		
System crashes from packet with unlimited extension headers	x	x	x						
System crashes from Router Advertisement (RA) Flooding	x		x						
System DoS through SEcure Neighbor Discovery(SeNd) flood	x		x						
System's IPv6 stack is bugged	x	x	x				x		
System's ND state table exhaustion from flooding	x		x						
Routing vulnerabilities									
DoS amplification through routing loops using tunnels	x	x	x						
Routing influenced by ICMP spoofing	x		x	x					x
Switch influenced by (T)CAM exhaustion	x		x	x					
ULA traffic routed to other networks	x			x					
Unused network space not NULL-routed	x		x	x					
Other vulnerabilities									
DoS amplification through DNS response packet size		x	x						
Security vulnerability in applications	x	x					x		

2.3 - Attacks that Cause Smart City Blackout

We have not seen any blackout of a city yet, but we will definitely see in the future. The Denial of Service attacks are considered the main cause for a city blackout. For that reason, the following section describes in details selected types of Denial of Service attacks. The Denial of Service attacks were the core focus which motivated us to create a solution involving innovative load balancing techniques to mitigate this kind of attacks – the subject of the second part of this research. There are countless types of Denial of Service attacks; the following section explores the main types pertinent to the research. Figure 2.7 exposes the DDoS attack vector frequency in the third quarter of year 2016. We notice that the main reason for this successful type of attacks is the UDP protocol. The UDP protocol is a connectionless protocol, it provides no authentication as the receiving party accepts the request without verifying its source IP. The highest percentages are coming from UDP based services (DNS, NTP, …)

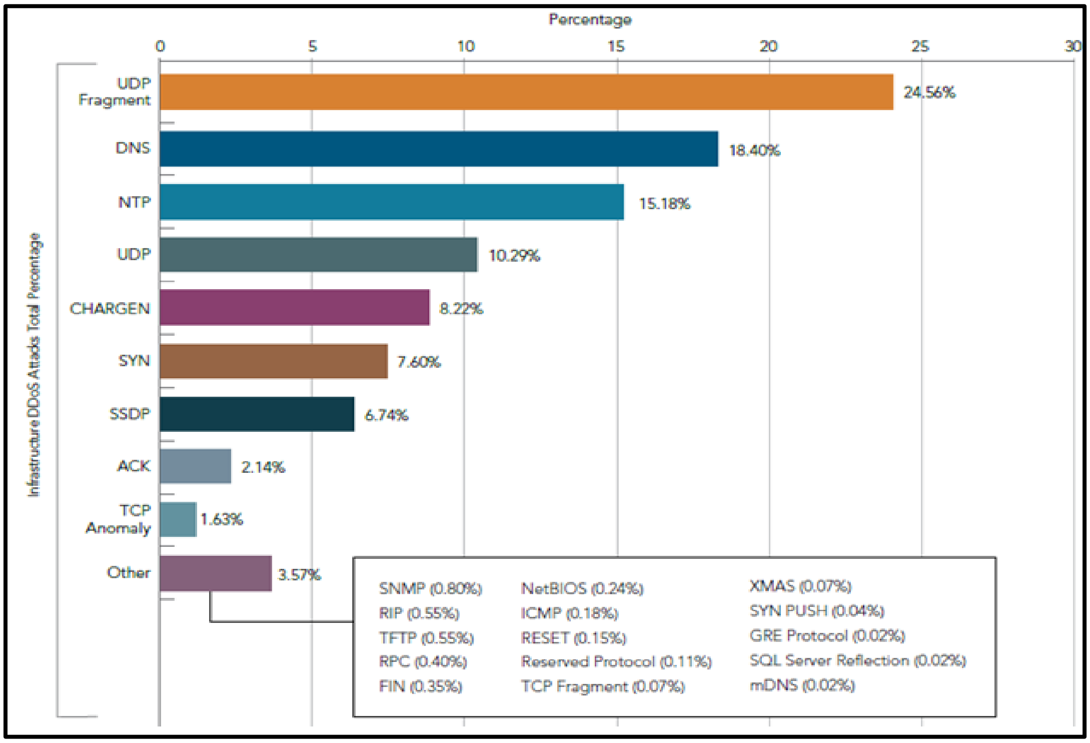


Figure 2.7 – DDoS Attack Vector Frequency, Q3 2016
Source: Akamai's State of the Internet / Security, Q3 2016 Report

Figure 2.8 displays the source of DDoS attack by country for the year 2015-2016. We see that China and U.S.A are the two dominating countries. Statistics provide significant evidence about the dynamics of this cyber-threat. In each quarter, there will be new countries joining the DDoS club. It is worth mentioning that top countries for DDoS attacks are themselves considered the top leading countries that are embracing smart city initiative.

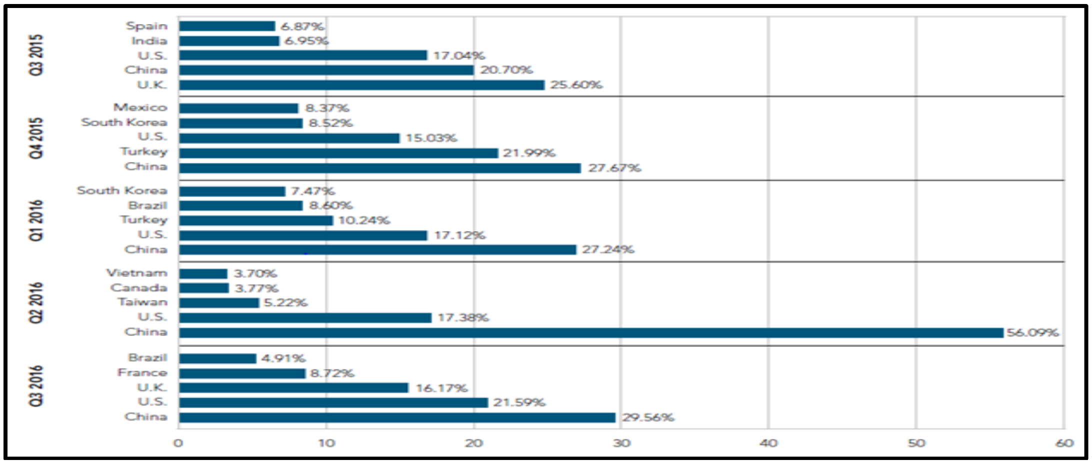


Figure 2.8 – Top 5 Countries for DDOS Attacks, Q3 2015 – Q3 2016
Source: Akamai's State of the Internet / Security, Q3 2016 Report

2.3.1 - Denial of Service Attacks.

The attacker's aim behind these types of attacks is to reduce or interrupt the victim's server availability. The server cannot provide any service anymore. The attack parameters are the network bandwidth and the type of packets to send. The attacker can a) increase the network bandwidth by controlling the resources and tricking others into participating in the attack; b) minimize risk of detection while also maximizing damage to the victim. Various types of Denial of Service attacks are available.

2.3.1.1 – Exploring Asymmetry Attack

Exploiting asymmetry is one of the dangerous Distributed Denial of Service (DDOS) attacks. The attacker uses a botnet which is a network of infected machines collected by the attacker to flood the server with tones of requests (packets). Usually DDoS are powered by volunteers that are available as open source software, they are ready to be downloaded, and launched, for example, Low Orbit Ion Canon (LOIC), and Anonymous.

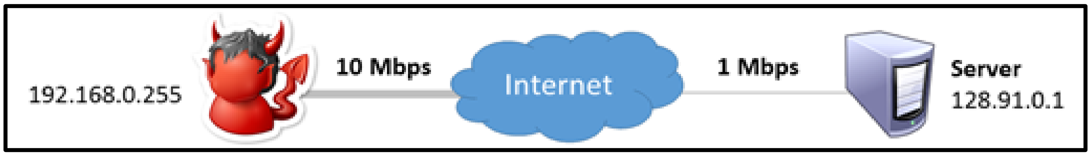


Figure 2.9 – Stage of the Network before the Attack - Courtesy of Wil Robertson

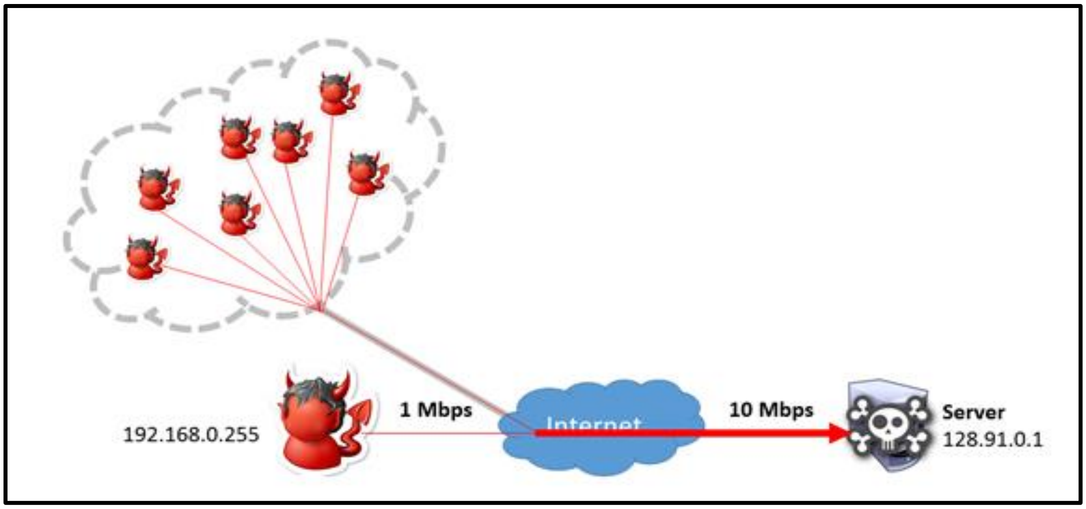


Figure 2.10 – Stage of the Network after the Exploiting Asymmetry Attack

Figure 2.9 shows the network bandwidth of the victim 1 Mbps, while for the attacker is 10Mbps. The attacker wants to distribute these 10 Mbps into the botnet causing each infected machine to use 1 Mbps to flood the server. Figure 2.10 shows the stage of the network after the attack where the botnet showers the server with at least 10 Mbps of bandwidth which are consider enough to paralyze the server and hence, casing denial of service.

3.3.1.2 - Smurf Attack

The attacker spoofs the victim's IP address and uses it to send a ping request to a broadcast address such as 10.7.0.255. The edge router then forwards the request to all /24 subnet hosts.

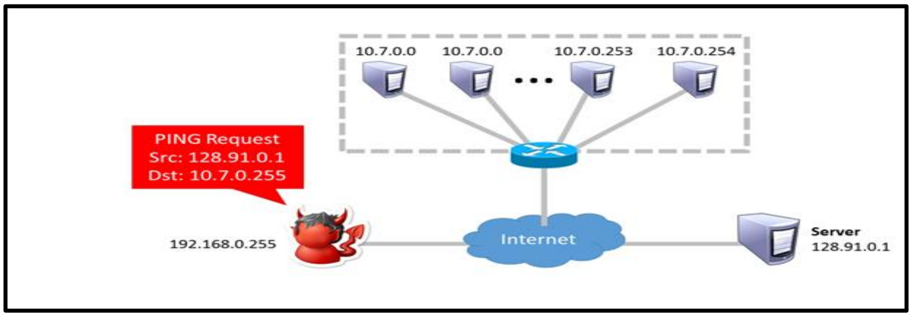


Figure 2.11 – Attacker Spoofs the Server's IP address - Courtesy of Wil Robertson

The 255 in the IP address 10.7.0.255 denotes a broadcast address. This means that the request is going to be broadcasted to the range of IPs starting from 10.7.0.1 to 10.7.0.254 a total 255 hosts (see figure 2.12 a)

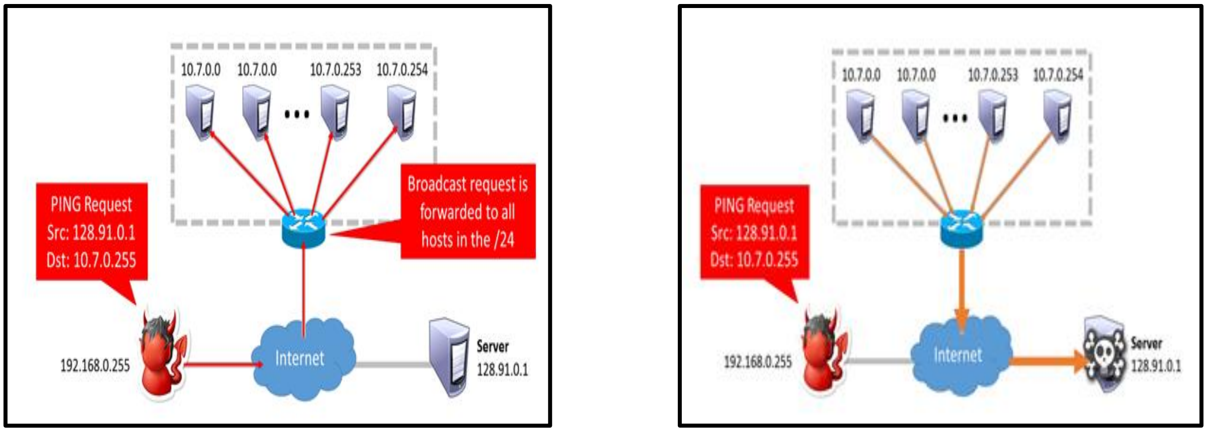


Figure 2.12 (a)

Figure 2.12 (b)

Figure 2.12 – Ping Request is Broadcasted to All 255 Hosts and Back to Server

The 255 hosts (assuming all respond) act now as attackers since they send back a reply to the real server's IP (128.91.0.1) which has already been spoofed by the attacker (see figure 2.12 b).

Smurfing works well since the ICMP protocol does not support authentication. It is a connectionless oriented protocol. Receivers accept the request without verifying its source IP. Attackers also benefit from the smurfing amplification factor:

$$Amplification\ factor\ = \frac{total\ response\ size}{request\ size}$$

The total response size is the number of servers or hosts that respond to broadcast, in our case the total response size is 255 and the request size is 1.

2.3.1.3 - DNS Reflection Attack

In this kind of attacks the spoofed DNS requests are forwarded to many **open** DNS resolvers. The DNS protocol is UDP based protocol. It is connectionless, that is does not support authentication of requests. The open DNS resolvers accept requests from any client including 8.8.8.8 and 8.8.4.4 which are static assigned Corporate DNS IP addresses assigned for Google. Two million open DNS resolvers on the internet witnessed on February, 2014. Attackers have to register their own domains and install very large record just to enable reflection attacks as the smurf attack, but on DNS servers. Spamhaus DDoSed at 300 Gbps via DNS reflection on March 2013 [56, 57]

2.3.2 - Mitigating Denial of Service Attacks

Several methods can be used to mitigate known (non-zero day attacks) denial of service attacks: The system administrator must use to Intrusion Detection System (IDS) to filter the incoming IP broadcasts at the gateway router to drop anything destined to *.*.*.255, and to disable non-essential services like Network Time Protocol (NTP) through UDP port 123, echo on port 7, and chargen on port 13. Also, the services must be configured to respond only to requests coming from the Local Area Network (LAN), and authenticate custom made UDP services [57].

2.3.3 – The Influence of the DDOS Attacks on the Smart City Network

Intruders can use the IoT devices like cameras, thermostat, light bulbs, among others for several suspicious activities to obtain personal information, spy on people, turn off smoke detector, drain the battery to render it unavailable, change the firmware to be able to use this device as a launching point for internal or external attacks, and compromise the device to install a malware as illustrated in the MIRAI case that took place on September 2016 in many areas in the world. These devices might get comprised due to several innate vulnerabilities because of their commercial nature. These devices either lack the basic security features or users lack the preliminary notions of security awareness of how to implement the appropriate safety techniques, namely, changing default password, enforcing authentication password, patching or updating the device frequently, identifying certificate revocations list for outdated devices, and applying seemly encryption mechanisms to protect the network from eavesdropping [58].

2.4 – Proposed Solutions

Two solutions have been proposed as an attempt to ensure state of readiness in delivering efficient smart city services. As stated in the thesis statement in chapter one, the static and conventional defensive mechanisms cannot ensure state of readiness as they fail to prevent 1) traffic analysis; 2) re-transmitted packets; 3) delayed packets; 4) packets from being jammed; and 5) malicious insiders and captured nodes. Today, a smart city can be perceived as a “predictive city” [59] where unusual incidents and events can be predicted making citizens aware of the current situation so that they can take the right decision as the next course of action.

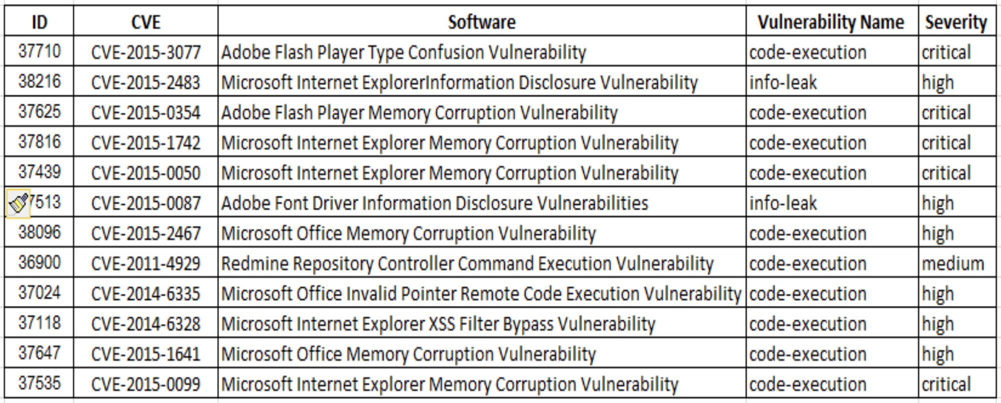
The cyber-security concept has changed to threat intelligence in which early detection of vulnerabilities and breaches are influential to being in a functional state.

Before diving into details, it is empirical to stress on the concept of the security metrics being the Common Vulnerability Scoring System (CVSS) as the core foundation for our solutions.

2.4.1 – Security Metrics

Security metrics are indicators used to provide contextual quantitative measure for the security characteristics of an information system. To improve the security of a system we have to measure it. The Common Vulnerability Scoring System (CVSS) was designed by a team of security professional and by the National Institute of Standard and Technology (NIST). It is a vulnerability scoring system designed to provide a standardized and open framework for rating software vulnerabilities. The Common Vulnerability and Exposure (CVE) is the industry standard for vulnerabilities and exposure names, namely the CVSS and the Common Weakness and Enumeration (CWE) which provides a list of software weaknesses. Each *vulnerability* of publicly known software is recorded and stored in the National Vulnerability Database (NVD) with a unique CVE number. See table 2.2 for NVD records example [20].

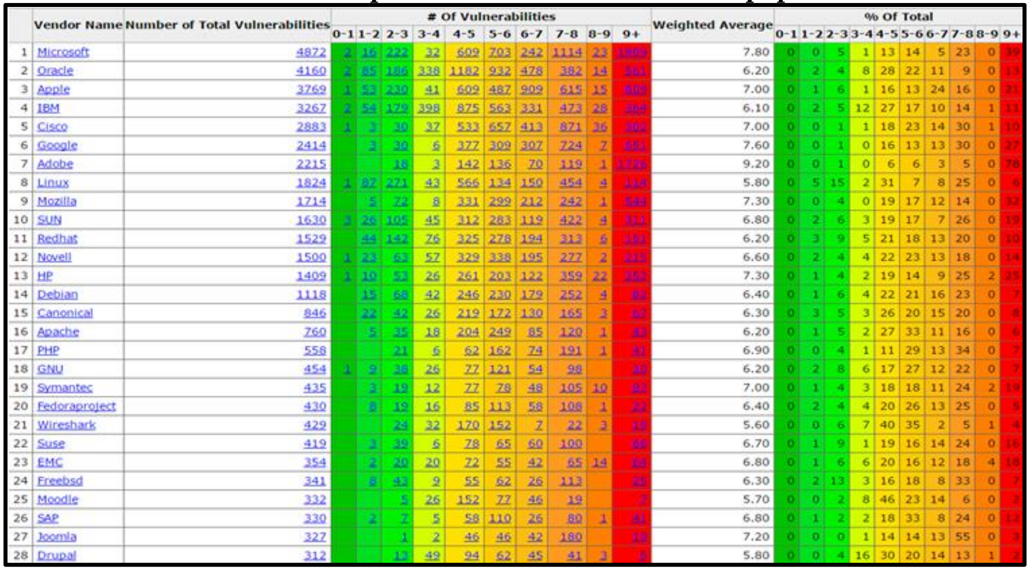
Table 2.2 – Sample of CVE Numbers for Selected Vulnerabilities



ID	CVE	Software	Vulnerability Name	Severity
37710	CVE-2015-3077	Adobe Flash Player Type Confusion Vulnerability	code-execution	critical
38216	CVE-2015-2483	Microsoft Internet ExplorerInformation Disclosure Vulnerability	info-leak	high
37625	CVE-2015-0354	Adobe Flash Player Memory Corruption Vulnerability	code-execution	critical
37816	CVE-2015-1742	Microsoft Internet Explorer Memory Corruption Vulnerability	code-execution	critical
37439	CVE-2015-0050	Microsoft Internet Explorer Memory Corruption Vulnerability	code-execution	critical
[illegible]7513	CVE-2015-0087	Adobe Font Driver Information Disclosure Vulnerabilities	info-leak	high
38096	CVE-2015-2467	Microsoft Office Memory Corruption Vulnerability	code-execution	high
36900	CVE-2011-4929	Redmine Repository Controller Command Execution Vulnerability	code-execution	medium
37024	CVE-2014-6335	Microsoft Office Invalid Pointer Remote Code Execution Vulnerability	code-execution	high
37118	CVE-2014-6328	Microsoft Internet Explorer XSS Filter Bypass Vulnerability	code-execution	high
37647	CVE-2015-1641	Microsoft Office Memory Corruption Vulnerability	code-execution	high
37535	CVE-2015-0099	Microsoft Internet Explorer Memory Corruption Vulnerability	code-execution	critical

Table 2.3 shows the CVSS score distribution for top 50 vendors by total number of distinct vulnerabilities.

Table 2.3 Number of Vulnerabilities of the Top 50 Vendors
Source: https://www.cvedetails.com/index.php



	Vendor Name	Number of Total Vulnerabilities	# Of Vulnerabilities										Weighted Average	% Of Total									
			0-1	1-2	2-3	3-4	4-5	5-6	6-7	7-8	8-9	9+		0-1	1-2	2-3	3-4	4-5	5-6	6-7	7-8	8-9	9+
1	Microsoft	4872	2	16	222	32	609	703	242	1114	23	1909	7.80	0	0	5	1	13	14	5	23	0	39
2	Oracle	4160	2	85	186	338	1182	932	478	382	14	561	6.20	0	2	4	8	28	22	11	9	0	13
3	Apple	3769	1	53	230	41	609	487	909	615	15	809	7.00	0	1	6	1	16	13	24	16	0	21
4	IBM	3267	2	54	179	398	875	563	331	473	28	364	6.10	0	2	5	12	27	17	10	14	1	11
5	Cisco	2883	1	3	30	37	533	657	413	871	36	302	7.00	0	0	1	1	18	23	14	30	1	10
6	Google	2414		3	30	6	377	309	307	724	7	651	7.60	0	0	1	0	16	13	13	30	0	27
7	Adobe	2215			18	3	142	136	70	119	1	1726	9.20	0	0	1	0	6	6	3	5	0	78
8	Linux	1824	1	87	271	43	566	134	150	454	4	114	5.80	0	5	15	2	31	7	8	25	0	6
9	Mozilla	1714		5	72	8	331	299	212	242	1	544	7.30	0	0	4	0	19	17	12	14	0	32
10	SUN	1630	3	26	105	45	312	283	119	422	4	311	6.80	0	2	6	3	19	17	7	26	0	19
11	Redhat	1529		44	142	76	325	278	194	313	6	151	6.20	0	3	9	5	21	18	13	20	0	10
12	Novell	1500	1	23	63	57	329	338	195	277	2	215	6.60	0	2	4	4	22	23	13	18	0	14
13	HP	1409	1	10	53	26	261	203	122	359	22	352	7.30	0	1	4	2	19	14	9	25	2	25
14	Debian	1118		15	68	42	246	230	179	252	4	82	6.40	0	1	6	4	22	21	16	23	0	7
15	Canonical	846		22	42	26	219	172	130	165	3	67	6.30	0	3	5	3	26	20	15	20	0	8
16	Apache	760		5	35	18	204	249	85	120	1	43	6.20	0	1	5	2	27	33	11	16	0	6
17	PHP	558			21	6	62	162	74	191	1	41	6.90	0	0	4	1	11	29	13	34	0	7
18	GNU	454	1	9	38	26	77	121	54	98		30	6.20	0	2	8	6	17	27	12	22	0	7
19	Symantec	435		3	19	12	77	78	48	105	10	83	7.00	0	1	4	3	18	18	11	24	2	19
20	Fedoraproject	430		8	19	16	85	113	58	108	1	22	6.40	0	2	4	4	20	26	13	25	0	5
21	Wireshark	429			24	32	170	152	7	22	3	19	5.60	0	0	6	7	40	35	2	5	1	4
22	Suse	419		3	39	6	78	65	60	100		68	6.70	0	1	9	1	19	16	14	24	0	16
23	EMC	354		2	20	20	72	55	42	65	14	64	6.80	0	1	6	6	20	16	12	18	4	18
24	Freebsd	341		8	43	9	55	62	26	113		25	6.30	0	2	13	3	16	18	8	33	0	7
25	Moodle	332			5	26	152	77	46	19		7	5.70	0	0	2	8	46	23	14	6	0	2
26	SAP	330		2	7	5	58	110	26	80	1	41	6.80	0	1	2	2	18	33	8	24	0	12
27	Joomla	327			1	2	46	46	42	180		10	7.20	0	0	0	1	14	14	13	55	0	3
28	Drupal	312			13	49	94	62	45	41	3	5	5.80	0	0	4	16	30	20	14	13	1	2

In this research two security solutions have been proposed and they both rely on the CVSS scores. The following sections illustrate the two proposed solutions in details.

2.4.2 – Proposed Solution #1

We proposed a predictive approach to analyze and evaluate quantitatively the vulnerabilities associated to the wired and wireless network. It could be applied to both the smart city network and smart grid network.

A study has been conducted to alleviate the vulnerabilities that could not be mitigated using the conventional security services namely the public key infrastructure (PKI), and private key cryptography mechanisms. For instance, cryptography can't prevent attacks related to traffic analysis, re-transmitted packets, delayed packets, packets from being jammed, malicious insiders and captured nodes.

This hypothesis led us to introduce a predictive threat model that captures essential characteristics of decision making to protect the network and avoid smart city blackout. The model uses the Markov Chain process to calculate the probability and assess the vulnerabilities severity level of a certain attack path within the network. The reason we adopt the Markov Chain process is that it uses a metric model as input to represent conditional probabilities of success for exploiting dependent vulnerabilities. This model is based on observed publicly known vulnerabilities provided by the NVD database and tracked with the Common Vulnerabilities and Exposures (CVE) system. Each NVD record provides the CVE id, vulnerability name, CVSS score and the level of severity.

Typical and scalable smart grid network architecture with three attack paths entries has been created. An attacker (hacker) could target the smart grid network through three potential attack paths entries: 1) the wireless 802.11 Wi-Fi networks; 2) the web interface connected to apache web server; and 3) the wireless ZigBee 802.15 IP sensor network. Since our method could be implemented to any type of network, we have chosen the cyber-attacks graph against smart wireless 802.11 Wi-Fi networks.

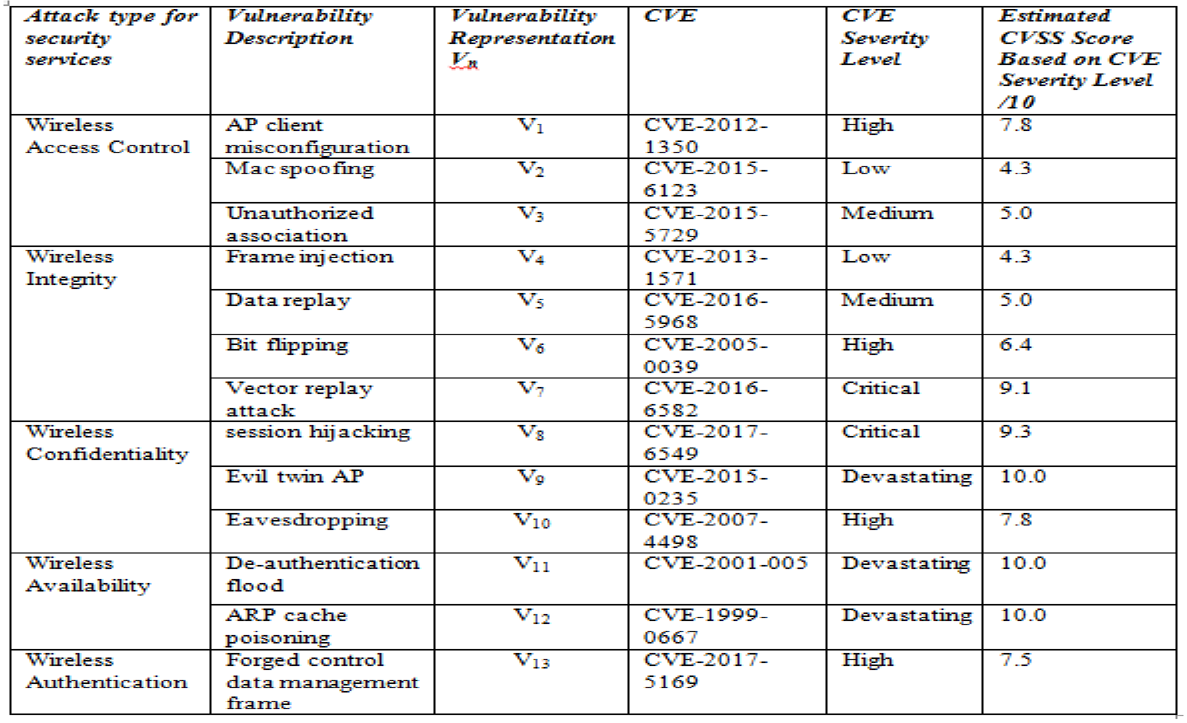
The following are some selected Wi-FI attacks that might lead to a complete blackout:

1) Wireless Access Control Attacks: Can happen through Access Point (AP) /Client misconfigurations, rouge access points, MAC spoofing, unauthorized association and Promiscuous clients; 2) Wireless Integrity Attacks: Such as data frame injections, WEP injections, data replay, vector replay attacks, bit-flipping attacks, AP replay attacks; 3) Wireless Confidentiality Attacks: The attacker practices eavesdropping, session hijacking, honey-pot AP, masquerading, evil twin AP; 4) Wireless Availability

Attacks: Constitute attacks like de-authenticate flood, routing attacks ARP cache positioning and power saving attacks; and 5) Wireless Authentication Attacks: The attackers send forged control, data and management frames over the wireless network to misdirect the wireless devices in order to perform Denial of Service (DOS) attack.

The table 2.4 below displays the corresponding CVSS score which has been deduced from the NVD database

Table 2.4 – Wireless Vulnerabilities Classification
Source: Reference [60]



Attack type for security services	*Vulnerability Description*	*Vulnerability Representation V_n*	*CVE*	*CVE Severity Level*	*Estimated CVSS Score Based on CVE Severity Level /10*
Wireless Access Control	AP client misconfiguration	V_1	CVE-2012-1350	High	7.8
	Mac spoofing	V_2	CVE-2015-6123	Low	4.3
	Unauthorized association	V_3	CVE-2015-5729	Medium	5.0
Wireless Integrity	Frame injection	V_4	CVE-2013-1571	Low	4.3
	Data replay	V_5	CVE-2016-5968	Medium	5.0
	Bit flipping	V_6	CVE-2005-0039	High	6.4
	Vector replay attack	V_7	CVE-2016-6582	Critical	9.1
Wireless Confidentiality	session hijacking	V_8	CVE-2017-6549	Critical	9.3
	Evil twin AP	V_9	CVE-2015-0235	Devastating	10.0
	Eavesdropping	V_{10}	CVE-2007-4498	High	7.8
Wireless Availability	De-authentication flood	V_{11}	CVE-2001-005	Devastating	10.0
	ARP cache poisoning	V_{12}	CVE-1999-0667	Devastating	10.0
Wireless Authentication	Forged control data management frame	V_{13}	CVE-2017-5169	High	7.5

Our motivation is to make predictions about a sequence or set of sequences of vulnerabilities v_1, v_2, v_3, ...v_n , based on models of observed data (Markov Model) which represent the attack paths in a Wi-Fi network. The level of severity (low, moderate, and high) which is represented by the computed probability of the attack path; i.e, the metric associated with each attack step reflects the vulnerability severity level of potential attack path

A detailed implementation of the solution is provided in chapter three as integrated into the paper titled *"**Evaluating Wireless Network Vulnerabilities and Attack Paths in Smart Grid - Comprehensive Analysis and Implementation.**"*

After implementing the solution, results show that there is a very high chance for the attacker to exploit ***Seq 3*** **(**v_4, v_6, v_8, v_9, v_{13}, and v_7) since its probability shows 0.08570=85.7%. In other words, ***Seq 3*** vulnerabilities v_4, v_6, v_8, v_9, v_{13}, and v_7 correspond to the attack path of: Frame injection (v_4) → Bit flipping (v_6) → Session hijacking (v_8) → Evil twin (v_9) → Wireless authentication (v_{13}) → Vector replay (v_7). Consequently, the system administrator has to carry out and apply the appropriate security measures to this sequence of vulnerabilities a priori to attacker's exploitation, and thus avoiding the catastrophic consequences of this cruel and malicious attack path.

2.4.3 – Proposed Solution #2

This approach is titled "**Mitigating Denial of Service Attacks Based on Identification, Heuristics and Load Balancer".**

Throughout the different online attacks obstructing the security of the network, Denial of Service or what is known as DoS has the greatest destructive effects. The increasing number of attacks and the effects of these happenings show the importance of researching such types of attacks. These attacks have led businesses down, destroyed the economy of a nation and even led to governments being changed. As we stated earlier in section 3 (**Attacks that Cause Smart City Blackout),** the Denial of Service attacks are considered the main cause for a city blackout.

DoS attacks are usually classified into several categories and this depends on the style with which they were implemented: 1) Distributed Denial of Service; 2) Low rate TCP targeted Denial of Service; 3) Reflective Denial of service. These categories encompass DoS attacks like: Smurf, Ping Flood and Ping of Death, TCP SYN Flood, and UDP Flood.

Several defensive mechanisms were used to mitigate Dos attacks, for instance, the Use of Access Control Lists, the Use of Rate Limiting, Destination Based Black Hole Filtering, Source Based Black Hole Filtering, and Attack Isolation. We noticed that there are no perfect mitigation procedures for this problem. Thus, we used the heuristic approach to produce a solution in a reasonable time frame that is good enough for solving the above problem.

Our solution consists of keeping track of our visitor's identification (such as IP), and separate our regular visitors from new ones. This could be achieved by adding a smart router / load balancer that distribute data depending on the visitor's reputation as illustrated in figure 1 and figure 2 respectively. Traffic is received by the main load balancer which, based on reputation and heuristics, send the user via the new visitor route or the trusted visitor route.

In case of a DOS attack, the main load balancer will be flooded and gone offline. Hence the backup load balancer will fill the void. The backup load balancer does not have a route for new users and thus blocks all new requests (hence blocks the attack) and safely allow trusted users to enter the network. This solution saves services from going offline, and increase clients trust by staying available regarding the attack. However the downside is that during the attack phase, some legitimate new visitors will be left out as well since they are not in the trust zone yet.

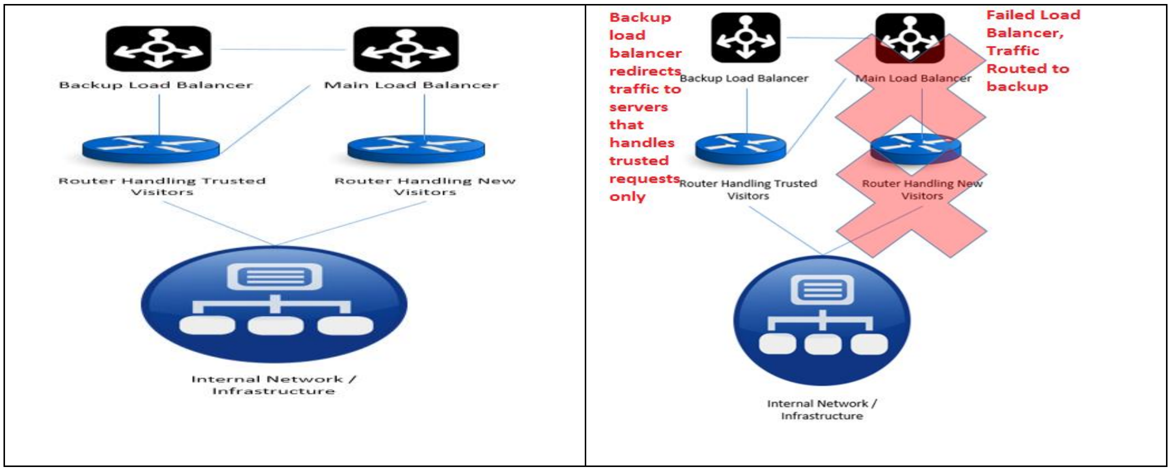

Backup Load Balancer
Main Load Balancer
Router Handling Trusted Visitors
Router Handling New Visitors
Internal Network / Infrastructure
Backup load balancer redirects traffic to servers that handles trusted requests only
Backup Load Balancer
Main Load Balancer
Failed Load Balancer, Traffic Routed to backup
Router Handling Trusted Visitors
Router Handling New Visitors
Internal Network / Infrastructure

Figure 2.13 - Load Balancer Setup

Our solution consists of two main parts: the physical infrastructure of the network and the algorithm identifying legitimate trusted traffic from the poisoned / infected traffic. For the infrastructure, we use two Ubuntu servers. Each of them is running HAPROXY to serve as a load balancer.

Having two (or more) load balancers is a critical step within this solution, as they are configured in a failover manner (load balancer peering), where the main load balancer is always functioning until an attack occur. Once an attack occurs, naturally, the first load balancer will be flooded with requests and hence the backup load balancer will take place. The trick is, the second load balancer only redirect traffic to servers that process only trusted requests and ignore all the others, hence, it will hardly be processing any dummy request and thus remain functioning normally. The second part of the solution is our algorithm to classify traffic as trusted or untrusted. For that, we test the request by having it pass through a set of rules, where each rule has its own weight. For each rule, the request is successfully tested against, the request gain the weight of this rule in an incremental manner. Once it's done, if the request has a specific total number of points, the request is then trusted. (Details are provided in chapter three)

2.5 – Conclusion

As we have seen in the course of this chapter, the smart city's network infrastructure depends heavily on the Internet of Things (IoT) devices. The number of smart techniques in which IoT devices can help our society is boundless, but unfortunately, they are susceptible to many of the same traditional types of attacks like Distributed Denial of Service Attacks (DDOS) and many others. The Denial of Service attacks are considered the main cause for a city blackout. Intruders can use the IoT devices like cameras, thermostat, light bulbs, among others for several suspicious activities to obtain personal information. Therefore, they could spy on people, turn off smoke detector, drain the battery to render it unavailable, change the firmware to be able to use this device as a launching point for internal or external attacks. Conventional wireless security measures are not enough to touch the IoT threats as they provide security solutions focusing on defending network perimeter. Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS) solutions are decent enough at monitoring and preventing attacks coming over the network, but they are weak at protecting IoT related attacks as these devices might use non-conventional communication protocols. Since these devices embrace massive volume of potential threats and vulnerabilities, it becomes imperative to study the IoT attacks, analyze the threats and vulnerabilities, and provide a cyber-security solution. The cyber-security concept has changed to threat intelligence in which early detection of vulnerabilities and breaches are influential for a smart city to being in a functional state.

In this chapter, we have suggested two solutions to ensure smart city services and operations continuity.
1) *A predictive and proactive approach* as a novel framework used to analyze and evaluate quantitatively the vulnerabilities associated with smart city network. This solution uses the Markov Chain Model to determine the highest vulnerability severity level of a particular attack path in the attacks graph of the network. This leads the system administrator to apply appropriate security measures a priori to potential attacks occurrence.
2) *A defensive or self-protective approach* as a framework that mitigates the zero-day availability attacks based on Identification, Heuristics and Load Balancer in a reasonable time frame. This defensive mechanism has been proposed mainly for DDoS protection since the DDoS attack is considered one of the most severe availability attacks that could paralyze the smart city's network and cause complete blackout. This solution relies on two load balancers in which the first one uses a heuristic approach, and the second acts as backup to produce a solution in a reasonable time frame.
Chapter three elucidates both solutions in details. It exposes the main purpose and the scientific approach used to mitigate the security challenges that smart cities are currently facing.

Chapter 3

Analysis and Implementation of Proposed Solutions for Smart City Protection

3.1- Introduction

This chapter explains in details the analysis and implementations of two innovative security solutions that can be applied to any smart city or smart grid network. What motivated us creating the forthcoming two solutions is that conventional security measures like public and private key cryptography failed to protect our network against several severe attacks as Distributed Denial of Service, re-transmitted packets, delayed packets, packets from being jammed, and malicious insiders. These attacks might lead to catastrophic impacts as they could paralyze the smart city network and cause complete black out.

The first solution is a novel framework used to predict, analyze and evaluate quantitatively the vulnerabilities associated with any wired or wireless network in a in a smart city or smart grid. To improve the security of a system we have to measure it. The Common Vulnerability Scoring System (CVSS) was designed by a team of security professional and by the National Institute of Standard and Technology (NIST). It is a vulnerability scoring system designed to provide a standardized and open framework for rating software vulnerabilities. The CVSS vulnerability scores were used to calculate the Markov processes (states) transitions probabilities as the first-order Markov chain allows us to predict the probability of a particular sequence of vulnerabilities [61]. Having the Markovian state diagram permitted us to predict the likelihoods of the vulnerability sequences which represent potential attack paths in a wired or wireless network [62]. Results proved to be noteworthy and alarming as vulnerability indicator. Precisely, our approach is now able to predict the probability of any vulnerability sequence of the attack graph representing any network topology. This tactic allows the system/security administrator to take appropriate decisions to protect the network and adopt provocative security measures against potential attacks.

The second solution is defensive or self-protective. This framework mitigates the zero-day availability attacks based on Identification, Heuristics and Load Balancer in a reasonable time frame. Security mechanisms like firewall, IPS and IDS could not prevent such type of zero day attacks. To reveal the severity of such attack, it is imperative to consider the case of the Dyn DDoS attack that was launched on October 21, 2016 and considered one of the biggest distributed denial of service attacks ever happened. Some major US websites including Paypal, Spotify, Twitter and Amazon faced connectivity issues.

Simulation results proved that the heuristic approach associated with the backup load balancer led to substantive accuracy rate in mitigating DDoS attacks. Sections 3.2 and 3.3 explain in details the implemented solutions, motivation, contributions, methodologies, and results.

3.2 - Solution # 1 – Evaluating Wireless Network Vulnerabilities and Attack Paths in Smart Grid Comprehensive Analysis and Implementation

3.2.1 - Introduction and Background

A smart grid is a union between energy, information technology and telecommunications. New technologies like smart meters, smart appliances, smart cars, etc… operate seamlessly on an intelligent infrastructure that integrates two layers of communications in a smart grid: one is to send electricity in both directions, and the other is to distribute information on energy demand and consumption, both with objectives of efficiently delivering sustainable, economic and secure electricity supplies. The two way flows of electricity and information make up the smart grid infrastructure foundation. The smart infrastructure system incorporates three subsystems: a) smart energy subsystem; b) smart communication system; c) smart information subsystems. Due to the widespread use of standard communication protocols and end user terminal devices, implementing complete security of every single device is hardly possible. Several security solutions using public key infrastructure (PKI), and private key cryptography mechanisms have been previously proposed [63] to solve the cyber-security problems related to confidentiality, integrity, availability and authenticity of the information for its use and services; However, those mechanisms have been proven inefficient since cryptography can't prevent: 1) traffic analysis; 2) re-transmitted packets; 3) delayed packets; 4) packets from being jammed; and 5) malicious insiders and captured nodes.

Given this hypothesis, in this paper we introduce a game-theoretic threat model that captures essential characteristics of decision making to protect the smart grid. Our model uses the Markov Chain Model to evaluate and assess the vulnerabilities of the wireless network. The reason we adopt the Markov Chain Model is that it uses as input a metric model to represent conditional probabilities of success for exploiting dependent vulnerabilities. This model is based on the observed vulnerabilities provided by the Common Vulnerabilities and Exposures (CVE) database. Each CVE comes with some Common Vulnerability Scoring System (CVSS) metrics [64] and parameters, which can be found in Table 3.1. The CVSS scores are used to calculate the likelihood of a particular state sequence and thus, quantifying the vulnerability severity level of any attack path in 802.11 network attacks graph. (See section 3.2.6 for methodology).

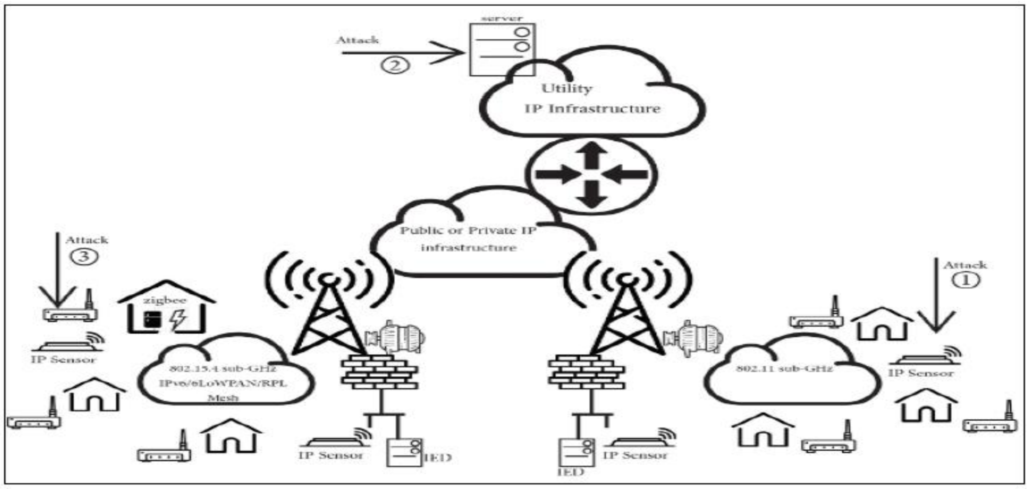


Figure 3.1 - Smart Grid Network - Attack Paths Entries

Figure 3.1 displays typical and scalable smart grid network architecture with three attack paths entries.

An attacker (hacker) could target the smart grid network through three potential attack paths entries: 1) the wireless 802.11 Wi-Fi networks; 2) the web interface connected to apache web server; and 3) the wireless ZigBee 802.15 IP sensor network. However, due to the number of pages limitation devoted for this research, this paper focuses only on the cyber-attacks graph against smart wireless 802.11 Wi-Fi networks and the interaction between the administrator and the attacker.

3.2.3 - Contribution

The main contribution of this paper is a novel framework to analyze and evaluate quantitatively the vulnerabilities associated to the 802.11 network in a smart grid. It is a probabilistic approach and uses Markov Chain Model to determine the likelihood of states transitions sequences of vulnerabilities attributed to vulnerable attack paths using the Common Vulnerability Scoring System (CVSS) scores. We then investigate and examine the quantitative results to prioritize and detect the highest vulnerability severity level of a particular attack path in the attacks graph of the Wi-Fi network.

Consequently, critical vulnerability levels would motivate security administrators to overcome potential attacks by applying appropriate security measures a priori rather than bearing the hassle posteriori.
Section 3.2.4 presents an overview of related work, section 3.2.5 describes the wireless 802.11 attacks and the attack graph chosen for this research, section 3.2.6 discusses the methodology and the theoretical model, section 3.2.7 explains the method simulation and results.

3.2.4 - Related Work

In this section we provide a summary of the state-of- the- art of security metrics used for evaluating networks vulnerabilities and security. Unlike our approach which is used to predict the likelihoods of a sequence or set of sequences of vulnerabilities $v_1, v_2, v_3, \ldots v_n$, based on models of observed data as those sequences represent potential attack paths in the wireless network, a recent study [65] has tackled the network security problem but from different perspective. Instead of predicting the likelihoods of vulnerability sequence, it proposed a model to determine the network security risk (risk=threat x vulnerability x impact) for the entire network based on the Exploitability sub-score and Impact sub-score. By constructing host access graph given firewall rules, and by using Markovian random walk, they prioritize the risk associated with each node via ranking. Summing up the risks associated with all nodes can determine the overall network security risk. Reference [66] proposed security metric of exploits, conditions, or both of the shortest length of attack paths for measuring the amount of security of networks. Alternatively, reference [67] suggested an attack tree labeled with abstract exploitability which is parsed to determine the attack sequences considering the amount of minimum effort needed along the easiest paths as metric. A follow up work assumed the arithmetic mean of all attack paths' lengths as security metric of average attackers' expected efforts in compromising given critical assets [68]. In another recent and similar work [69], the authors used probabilities of attackers reaching the states of an attack graph as states' ranks during a random simulation of the PageRank algorithm. Reference [70] presented a method based on minimal critical set without considering the complicated relation between the attack and the network configuration elements. There exist several standardization efforts on security metrics. The most two widespread are the Common Vulnerability Scoring System (CVSS) and the Common Weakness Scoring System (CWSS) [71]. The first which is used as the basis for this research, focuses on ranking known vulnerabilities, whereas the second on software weaknesses.

3.2.5 – Vulnerability Analysis of the Wireless 802.11 Protocol

In order to discover the vulnerabilities in a Wi-Fi secure communication, it was essential to unleash the source of threats in the design infrastructure of the network. The concerns for wireless security requirements (confidentiality, integrity, availability, authenticity, and accountability.) in terms of threats, and countermeasures, are similar to those found in a wired environment; however, the most significant source of risk in wireless networks is the underlying communications medium.

The following scenario explains the consequences in a typical smart grid wireless 802.11 network (see figure 2) attacks that would violate the Confidentiality, Integrity and Availability (CIA) triad requirements or services.

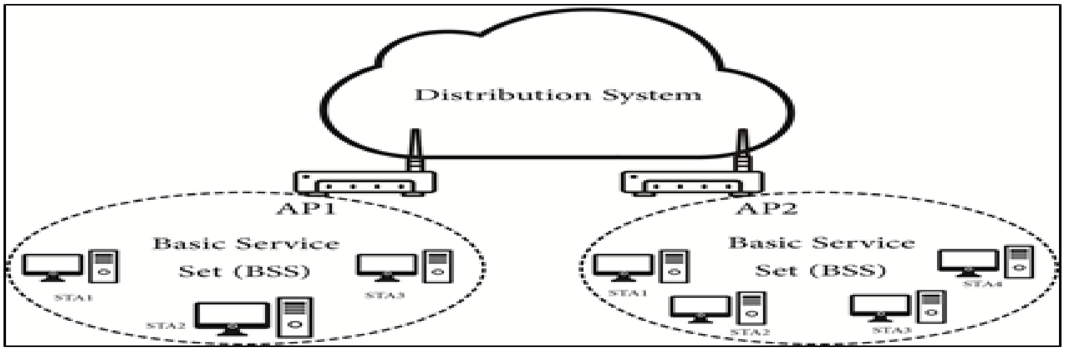


Figure 3.2 - 802.11 Wi-Fi Network in a Smart Grid

Depending on the smart grid's communication design, architecture and implementation, the effects of the below attacks are catastrophic [72]:

1) Wireless Access Control Attacks: Can happen through Access Point (AP) /Client misconfigurations, rouge access points, MAC spoofing, unauthorized association and Promiscuous clients.

2) Wireless Integrity Attacks: Such as data frame injections, WEP injections, data replay, vector replay attacks, bit-flipping attacks, AP replay attacks.

3) Wireless Confidentiality Attacks: The attacker practices eavesdropping, session hijacking, honey-pot AP, masquerading, evil twin AP.

4) Wireless Availability Attacks: Constitute attacks like de-authenticate flood, routing attacks ARP cache positioning and power saving attacks.

5) Wireless Authentication Attacks: The attackers send forged control, data and management frames over the wireless network to misdirect the wireless devices in order to perform Denial of Service (DOS) attack [73].

Consequently, these attacks may lead to an entire blackout in a smart grid. For instance, the smart grid incorporates some new wireless technology, including controls devices that make it possible to monitor electricity use in real-time and make automatic changes that reduce energy waste. This system is vulnerable to manipulation by the third party in smart grid wireless communication. Furthermore, accessing wireless critical access points would let attackers manipulate power-grid data by breaking into substations and intercepting communications between substations, grid operators, and electricity suppliers. If someone wanted to cause a blackout, the load data about how much power is flowing could be used to fool grid operators into overloading parts of the grid. A blackout could then occur before grid operators have the chance to correct for the problem. With similar hacking technique, this data is also used by grid operators to set prices for electricity and to balance supply and demand. Grid hackers could make millions of dollars at the expense of electricity consumers by influencing electricity markets [74].

3.2.6 – Methodology

The rapid development of security exploits in recent years has fueled a strong interest in data analysis tools for computer security. According to AV-TEST (http://www.av-test.org/en/home/) more than 200,000 examples of new malware are sighted daily. The huge number of malicious software witnessed by security experts goes beyond the limits of traditional analysis methods.

Our motivation is to make predictions about a sequence or set of sequences of vulnerabilities v_1, v_2, v_3, ….v_n , based on models of observed data (Markov Model) which represent the attack paths in a Wi-Fi network. The level of severity (low, moderate, and high) which is represented by the computed probability of the attack path; i.e, the metric associated with each attack step reflects the vulnerability severity level of potential attack path. A simple model is that observations are assumed to be independent and identically distributed. Henceforth, in this paper we use the Markov Chain Model as it has proved to be a valuable tool in analyzing and modeling/reproducing observed data.

Markov Chain Model: A Markov chain is a stochastic process with Markov property [75]. Markov property means that, given the present state, future states are independent of the past states. A homogeneous finite state Markov process can be defined as a process in which the current state of the system depends on the previous state (or states) in such a way that the state dependency at a particular time instant is not dependent on the time at which the system is observed. A probabilistic approach is used to determine the future state. Information of the present states influences the evolution of the process.

In our forthcoming example the inputs to the Markov chain are the component metrics, associated with each attack-step node, which indicate the severity of a single vulnerability. The metric represents the conditional probability that a single attack step will succeed when all the prerequisite conditions are met.

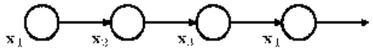
Defining formally the Markov chain will highlight some important characteristics of our proposed approach:

A Markov chain of order K is defined as a process (state) X1, X2, …Xt such that

$\Pr(X_{t+1}|X_1, X_2, \dots, X_t) = \Pr(X_{t+1} \mid X_{t-k+1}, X_{t-k+2}, \dots, X_t)$ *(3.1)*

If the n'th observation in a chain of observations is influenced only by the n-1'th observation, i.e.

$Pr(X_n \,|X_1, \dots, X_{n-1}) = Pr(X_n | X_{n-1})$ *(3.2)*



then the chain of observations is a 1st-order Markov chain, and the joint-probability of a sequence of N observations is

$Pr(X_1, \dots, X_n) = \prod_{n=1}^{N} Pr\,(X_n|X_1, \dots, X_{n-1}) = Pr(X_1) \ \prod_{n=1}^{N} Pr\,(X_n|X_{n-1})$ *(3.3)*

If the distributions p(Xn | Xn-1) are the same for all n, then the chain of observations is a homogeneous 1st-order Markov chain.

For a first-order Markov chain, the state transition matrix A is written as

$$\boldsymbol{A} = (a_{ij}) = \begin{bmatrix} a_{11} & a_{12} \dots a_{1N} \\ a_{21} & a_{22} \dots a_{2N} \\ \vdots & \vdots \\ \vdots & \vdots \\ a_{N1} & a_{N2} \dots a_{NN} \end{bmatrix} \qquad (3.4)$$

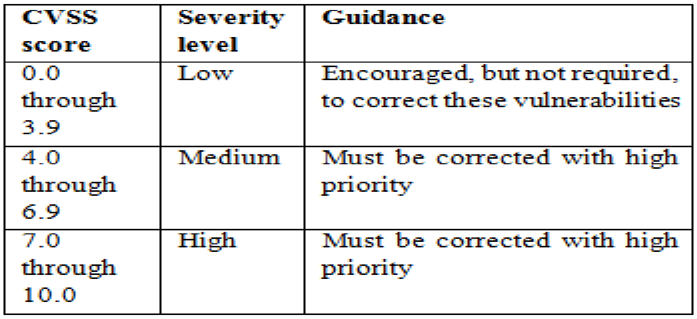
where a_{ij} denotes probability of transitioning from state *i* to state *j*, then for a first-order Markov process we can write

$a_{ij} = Pr(X_{n+1} = i | X_n = j)$ *(3.5)*

and $\sum_{j=1}^{N} a_{ij} = 1$ (stochastic process)

As an example of a first-order Markov chain, let us show how to predict the probability of a particular sequence of vulnerabilities. Based on the severity level of vulnerabilities provided by CVSS as shown in table I, we suppose that there is a three state Markov chain process in which state 1 denotes LOW (L) for CVSS scores 0.0 through 3.9, state 2 denotes MEDIUM (M) 4.0 through 6.9, and state 3 denotes HIGH (H) 4.0 through 6.9.

Table 3.1 – Vulnerability Severity Level



CVSS score	Severity level	Guidance
0.0 through 3.9	Low	Encouraged, but not required, to correct these vulnerabilities
4.0 through 6.9	Medium	Must be corrected with high priority
7.0 through 10.0	High	Must be corrected with high priority

We wish to compute the probability of observing L L M L H L sequence given that the initial state is L. The initial state probability is =1 (since it is the starting state)

Now the state transition matrix **A** shown in (3.4) can be converted to a Markovian state transition diagram (see figure 3) where a_{ij} denotes probability of transitioning from state *i* to state *j*.

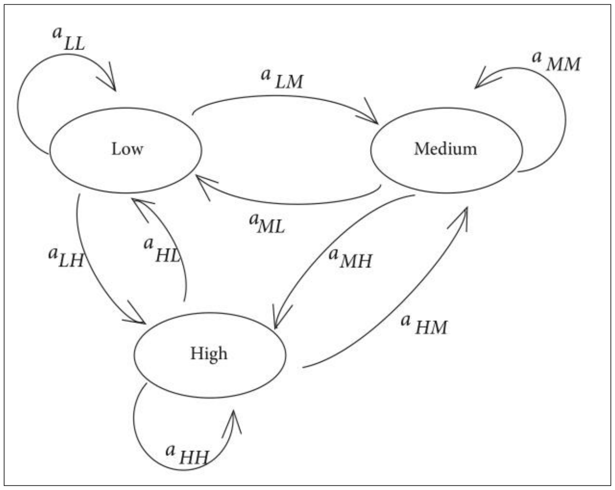


Figure 3.3 - Markovian State Transition Diagram

Computing the probability of observing L L M L H L sequence requires calculating the transition probabilities for all the states in the Markovian state transition diagram.

To be able to calculate the a_{LL}, a_{LH}, a_{LM}, a_{MM}, a_{ML}, a_{MH} and so on, we need to: 1) classify the vulnerabilities of the 802.11 attacks discussed in section 3.2.5 based on severity level; 2) retrieve the corresponding CVE score from the CVS database provided by MITRE; 3) construct possible attack paths through which an attack might happen to compromise most potent target nodes in the Wi-Fi network. For demo purpose, table 3.2 classifies a subset of 802.11 vulnerabilities that would lead to attacks compromising the security services as described in section 3.2.5.

Table 3.2 – Wireless Vulnerabilities Classification

Attack type for security services	***Vulnerability Description***	***Vuln.***	***CVE Name and ID***	***CVE Severity Level***	***Est. CVSS Score /10***
Wireless Access Control	AP client misconfiguration	V_1	CVE-2012-1350	High	7.8
	Mac spoofing	V_2	CVE-2015-6123	Low	3.4
	Unauthorized association	V_3	CVE-2015-5729	Medium	5.0
Wireless Integrity	Frame injection	V_4	CVE-2013-1571	Low	3.4
	Data replay	V_5	CVE-2016-5968	Medium	5.0
	Bit flipping	V_6	CVE-2005-0039	Medium	6.4
	Vector replay attack	V_7	CVE-2016-6582	Critical	9.1
Wireless Confidentiality	session hijacking	V_8	CVE-2017-6549	Critical	9.3
	Evil twin AP	V_9	CVE-2015-0235	Devastating	10.0
	Eavesdropping	V_{10}	CVE-2007-4498	High	7.8
Wireless Availability	De-authentication flood	V_{11}	CVE-2001-005	Devastating	10.0
	ARP cache poisoning	V_{12}	CVE-1999-0667	Devastating	10.0
Wireless Authentication	Forged control data management frame	V_{13}	CVE-2017-5169	High	7.5

A possible attack path of the attack graph for the sequence L L M L H L might include a sequence of vulnerabilities v_1,v_4, v_6, v_8, v_{11}, and v_{13} or any other sequence of vulnerabilities provided that v_i depends on v_{i-1} *for all i=1 …. n.*

802.11 network nodes connectivity and accessibility are the essential requirements of attack graph cconstruction. This needs inspection of wireless routers and firewall controls and the network topology. So now that we have the vulnerability scores (see table 3.2), we can move ahead with the attack graph model.

A typical attack graph can be modeled as shown in figure 3.4

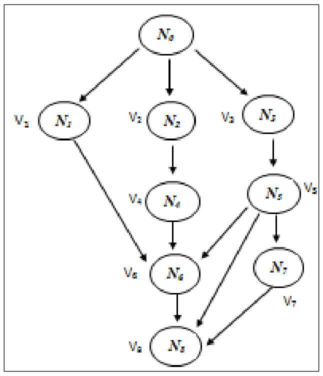


Figure 3.4 - Attack Graph Showing Vulnerable Nodes and Multiple Attack Paths (v1, v2, ….)

Where:

n = number of nodes in the network

n_a = number of nodes in attack graph

V_i = Vulnerability score of ith node

N_0 = be source node from where attack would be launched

Probability that ith node is compromised by the attacker is given by

$$P(N_i) = \frac{V_i}{\sum_{k=1}^{n_a} V_k} \qquad (3.6)$$

Deducing the probability of each node, it becomes straight forward to calculate the transition probabilities for all the states in the Markovian state transition graph shown in figure 3.3 and hence determining the attack path using formula 3.3.

3.2.7 – Simulation and Results

To simulate the theory explained above and for the purpose of estimating the probability of observing L L M L H L sequence, we divided the vulnerabilities shown in table II into three categories. Low = {*v2, v4*}, Medium = {*v3, v5, v6*}, and High = {*v1, v7, v8, v9, v10, v11, v12, v13*}

Now the state transition matrix in 4.4 can be written as:

$$\boldsymbol{A} = (a_{ij}) = \begin{bmatrix} a_{LL} & a_{LM} & a_{LH} \\ a_{ML} & a_{MM} & a_{MH} \\ a_{HL} & a_{HM} & a_{HH} \end{bmatrix}$$

Feeding the CVSS scores provided in Table II to *R* software led to the following matrix results:

$$\boldsymbol{A} = (a_{ij}) = \begin{bmatrix} 0.12 & 0.31 & 0.57 \\ 0.21 & 0.25 & 0.54 \\ 0.01 & 0.19 & 0.8 \end{bmatrix}$$

Applying the formula in 3.3, we can predict the joint probability of any vulnerability sequence and interpret the result.

To demonstrate how to compute the probability of a specific sequence, let us consider the vulnerability sequence $v_2, v_4, v_6, v_2, v_9, v_4$. Perceiving table 3.2, one can obviously identify its corresponding observed severity sequence L L M L H L. Therefore, using the Markovian state transition diagram (see figure 3) the joint probability of *Pr(L, L, M, L, H, L)* can be calculated as:

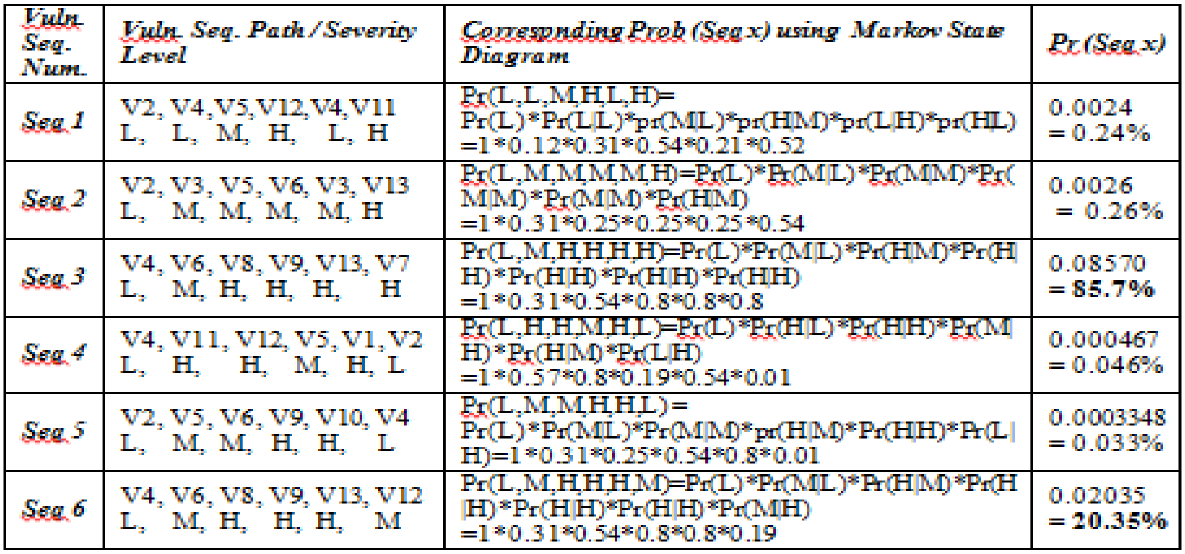
Pr(L, L, M, L, H, L) = Pr(L)Pr(L|L)Pr(M|L)Pr(L|M)Pr(H|L)Pr(L|H)

= 1(0.12)(0.31)(0.21)(0.57)(0.01) =0.0000445

The vulnerability severity level of 0.0000445 is considered negligible or insignificant. This indicates that the security measures applied by the system administrator to the sequence $v_2, v_4, v_6, v_2, v_9, v_4$ of the Wi-Fi network have been made tight and solid (hard to be compromised).

To better understand the proposed framework, table 3.3 shows the results of several selected vulnerability sequences probabilities.

Table 3.3 – Predicted Vulnerability Sequence Probabilities



Vuln. Seq. Num.	*Vuln. Seq. Path / Severity Level*	*Correspnding Prob (Seq x) using Markov State Diagram*	*Pr (Seq x)*
Seq 1	V2, V4,V5,V12,V4,V11 L, L, M, H, L, H	Pr(L,L,M,H,L,H)= Pr(L)*Pr(L|L)*pr(M|L)*pr(H|M)*pr(L|H)*pr(H|L) =1*0.12*0.31*0.54*0.21*0.52	0.0024 = 0.24%
Seq 2	V2, V3, V5, V6, V3, V13 L, M, M, M, M, H	Pr(L,M,M,M,M,H)=Pr(L)*Pr(M|L)*Pr(M|M)*Pr(M|M)*Pr(M|M)*Pr(H|M) =1*0.31*0.25*0.25*0.25*0.54	0.0026 = 0.26%
Seq 3	V4, V6, V8, V9, V13, V7 L, M, H, H, H, H	Pr(L,M,H,H,H,H)=Pr(L)*Pr(M|L)*Pr(H|M)*Pr(H|H)*Pr(H|H)*Pr(H|H)*Pr(H|H) =1*0.31*0.54*0.8*0.8*0.8	0.08570 = **85.7%**
Seq 4	V4, V11, V12, V5, V1, V2 L, H, H, M, H, L	Pr(L,H,H,M,H,L)=Pr(L)*Pr(H|L)*Pr(H|H)*Pr(M|H)*Pr(H|M)*Pr(L|H) =1*0.57*0.8*0.19*0.54*0.01	0.000467 = 0.046%
Seq 5	V2, V5, V6, V9, V10, V4 L, M, M, H, H, L	Pr(L,M,M,H,H,L)= Pr(L)*Pr(M|L)*Pr(M|M)*pr(H|M)*Pr(H|H)*Pr(L|H)=1*0.31*0.25*0.54*0.8*0.01	0.0003348 = 0.033%
Seq 6	V4, V6, V8, V9, V13, V12 L, M, H, H, H, M	Pr(L,M,H,H,H,M)=Pr(L)*Pr(M|L)*Pr(H|M)*Pr(H|H)*Pr(H|H)*Pr(H|H)*Pr(M|H) =1*0.31*0.54*0.8*0.8*0.19	0.02035 = **20.35%**

The predicted probabilities of ***Seq 3*** and ***Seq 6*** show ***85.7%*** and ***20.35%*** respectively which are considered very high, significant and alarming. The likelihood for the hacker to attack the network communication system is 85.7% if he/she exploits ***seq3*** vulnerabilities. Moreover, ***Seq 3*** probability **(85.7%)** denotes that almost any security measures have not been implemented for the wireless network system. There is a very high chance for the attacker to exploit ***Seq 3*** and ***Seq 6*** vulnerabilities, intrude the system, escalate privileges, and perform malicious activities. In other words, table II demonstrates that ***Seq 3*** vulnerabilities v_4, v_6, v_8, v_9, v_{13}, and v_7 correspond to the attack path of: Frame injection (v_4) → Bit flipping (v_6) → Session hijacking (v_8) → Evil twin (v_9) → Wireless authentication (v_{13}) → Vector replay (v_7). Consequently, the system administrator has to carry out and apply the appropriate security measures to this sequence of vulnerabilities a priori to attacker's exploitation, and thus avoiding the catastrophic consequences of this cruel and malicious attack path.

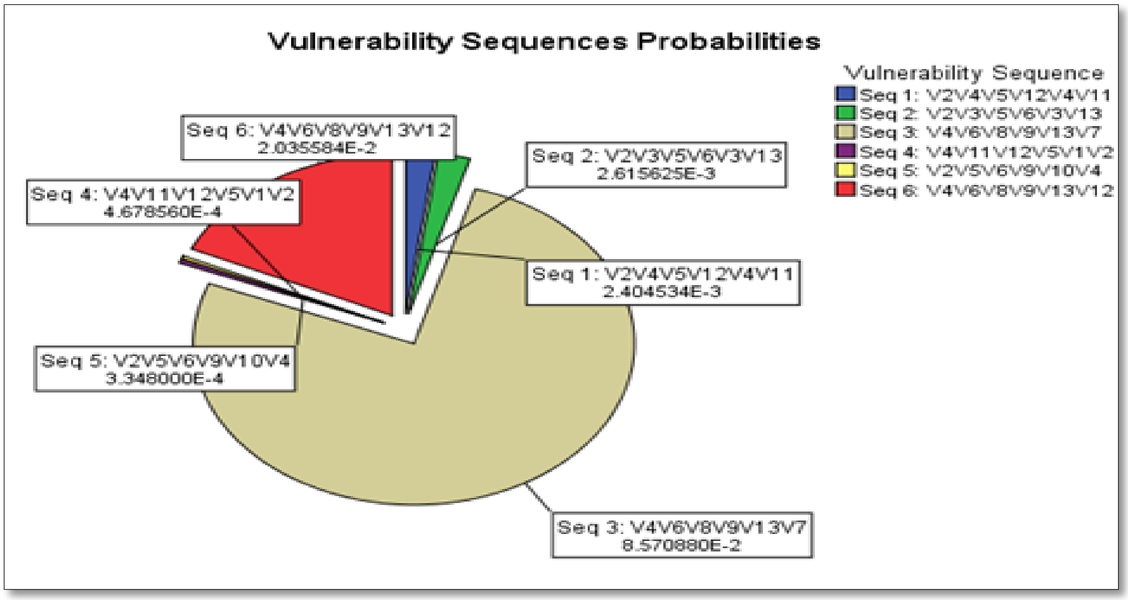


Figure 3.5 - Pie Chart Representation of Sequence Probabilities

On the other hand, figure 3.5 shows tiny slices for ***seq 1***, ***seq 2***, ***seq 4*** and ***seq 5*** which correspond to the likelihood of 0.24%, 0.26%, 0.046% and 0.033% respectively. Those probabilities are considered negligible and hence insignificant. Therefore, there is a very low chance for the hacker to exploit the vulnerabilities of the aforementioned sequences' vulnerabilities attack paths since the system administrator might have already implemented the appropriate security measures to the Wi-Fi network.

3.3 – Solution # 2 Mitigating Denial of Service Attacks Based on Identification, Heuristics and Load Balancer

3.3.1 - Introduction and Background

Regardless of what are the malicious urges of hackers in intruding smart city systems, Denial of Service attacks or availability attacks are considered the most vicious and destructive as they might bring down or paralyze the overall city's network services. The increasing number of attacks and the effects of these happenings show the importance of researching such types of attacks. These attacks have led businesses down, destroyed the economy of a nation and even led to governments being changed. DoS attacks are usually classified into several categories and this depends on the style with which they were implemented: 1) Distributed Denial of Service; 2) Low rate TCP targeted Denial of Service; 3) Reflective Denial of service. These categories encompass DoS attacks like: Smurf, Ping Flood and Ping of Death, TCP SYN Flood, and UDP Flood.

Smart city dimensions rely heavily on Internet of Things (IoT) devices. Thus, exploiting effectively IoT devices vulnerabilities by botnets will definitely lead to a wide range of destructive distributed denial of service attacks.

Unlike conventional devices, IoT devices are exposed to three different types of availability attacks: 1) hardware availability, 2) network availability and 3) cloud availability. Consequently, currently adopted DDoS mitigation techniques against IoT attack vectors for availability have proven inefficient as considerable discrepancies and inconsistency of IoT devices have become apparent. For example, an IoT device can be forced to use more power as a result of exchanging multiple keys over ZigBee smart connection and thereby causing battery drain. Likewise, IoT devices are susceptible to radio jamming due to the use of radio networking (Wifi, Zigbee, Bluetooh, 3G)[76].

The biggest challenge that we face nowadays is the use of IoT devices (camera, DVR, thermostat, etc.) to launch a high-profile Distributed Denial of Service attacks as it lately happened with the Mirai botnet. Recent attacks showed that IOT devices vulnerabilities were most efficiently used by botnets to unveiling a variety of Distributed Denial of Service (DDoS) attacks. This kind of botnet not only affects the IoT devices, but also everybody connected to the Internet. To reveal the severity of the DDoS attacks caused by IoT botnets, we hereby list some recent incidents that took place late 2016.

On September 30, 2016, the blog of the security researcher Brain Kerbs (kerbsOnSecurity.com) started experiencing a 632 Gbps DDoS attack from an IoT botnet. Akamai, the leading Content Delivery Network (CDN) operator which provided the DDoS attack mitigation, was accused after three days by its customers for the scarcity of their internet bandwidth as Akami platform could not handle anymore the technical means required to avoid the attack. [77, 78]

On September 22, 2016, the French cloud computing company (OVH) experienced a series of DDoS attacks with the severest single attack reaching up to 799 Gbps. Later on September 23rd, it experienced another DDoS generating up to1.5Tbps coming from around 146000 cameras, printers and DVRs. The IoT devices were sending from 1 to 30 Mbps of traffic each [79].

On October 21st, Dyn the leading managed DNS provider and Internet Performance Management Company, experienced on its DNS servers several waves of DDoS attacks from 100000 internet connected IoT devices with the severest attack generating 1.2Tbps of traffic. [80]

Regardless of the huge amount of damages brought about from the above three attacks, our challenge is to look forward and find a solution to mitigate such attacks by protecting not only the IoT devices, but the overall network infrastructure. In the three previous attacks all IoT internet connected devices were found mainly infected by Mirai malware.

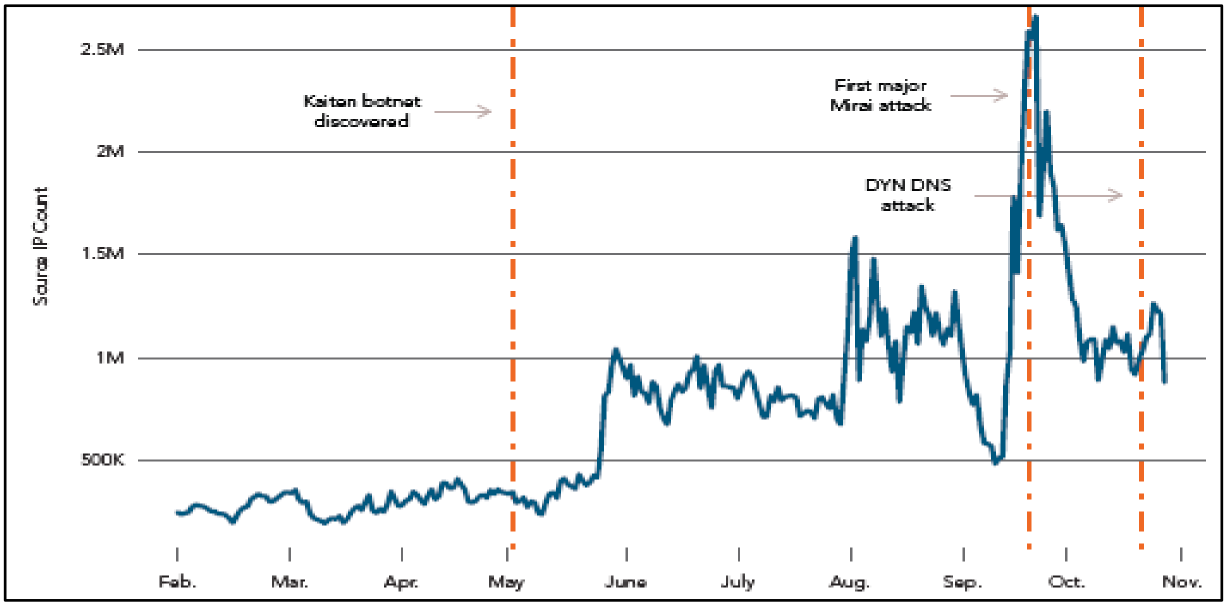


Figure 3.6 - Scans of port 23 and 2323 began on May 13, 2016 as the Mirai botnet attempted to log into unsecure IoT devices
Source: Akamai Q4 – State of the Internet Security Report 2016

The amount of networked IoT device will certainly increase in years to come. Currently there are up to 15 billion internet connected IoT devices in use and it is predicted to become 200 billion in 2020. We expect to see in the future more damaging IoT DDoS attacks because the source code of the Mirai malware is freely available on the Internet [81]. The Mirai source code has been published on a community hack forum as open-source by its suspected author Paras Jha using the moniker online name “Anna-senpai” [82].

In just only two weeks after the release of Mirai source code, the number of infected devices has become more than double as it has increased drastically from 213000 to 483000. After examining the IP addresses of the infected IoT devices, one could notice that the source of the attack of the botnet is distributed over 164 counties. High profile attack densities were found in USA, China, Brazil, and Vietnam. Mirai malware generates not only conventional HTTP, TCP, or UDP traffic, but also exploits legitimate protocols like GRE IP (used for peer to peer VPN), GRE ETH, DNS, STOMP to flood against a specific target during a DDoS attack [83,84].

Mirai is designed to exploit default and hardcoded credentials and self-propagates by scanning the Internet. It utilizes a dictionary of generic and device-specific default credentials to infect IoT devices. Also, non-secure by default ports due to Universal Plug and Play (UPnP) were considered as easy target for the Mirai botnet. Several devices like those were made by the Chinese company XiongMai Technologies, can be accessed through a web interface – (IP/Login.htm) - navigating to "DVR.htm" without device credentials and prior to login [85].

Flashpoint security firm estimated over 515,000 vulnerable devices with hardcoded credentials were found actively in use on early October 2016.

The following sections describe in details the problem statement, the challenges, our contributions, the solution methodology and results.

3.3.2 - Problem Statement

Conventional mitigation methods against DDoS attacks which depend deeply on Intrusion Detection Systems (IDS), Intrusion Prevention Systems (IPS), Network Access Controls (NAC) and firewalls functionalities have proven deficiency, and the attack examples provided above recognize this fact.

Preventive measures from firewalls (use of Access Control Lists, use of rate limiting, destination based black hole filtering, dropped packets, port errors, dictionary logon attacks etc.), IDSs (buffer overflow attacks, fragmentation and replay attacks, cached content change, file integrity check etc.), Secure Socket Layer (certificate errors, DoS attacks, session drop) can jointly be overwhelmed when encountering severe waves of DDoS attacks each generating up to 1.2Tbps of traffic as experienced by the Mira malware.

System administrators should not rely totally on the aforementioned protection mechanisms as they could not: a) compensate for weak identification and authentication mechanisms; b) conduct investigations of attacks without human intervention; c) perceive the contents of your organizational security policy; d) compensate for weaknesses in network protocols; e) analyze all of the traffic on a busy network; f) deal always with some problems involving packet-level attacks. Yet, no perfect mitigation for this problem exists [86, 87, 88].

3.3.3 - Challenges and Contributions

Our main challenging task in this study is to find an efficient, defensive and innovative solution that might lessen the impact of distributed denial of service attacks on the smart city network (IoT devices) to avoid entire blackout.

The defensive mechanism that we proposed for DDoS protection relies on two load balancers in which the first one uses a heuristic approach, and the second acts as backup to produce a solution in a reasonable time frame. In brief, traffic is received by the main load balancer which, based on reputation and heuristics, sends the user via the new visitor route or the trusted visitor route. In case of a DOS attack, the main load balancer will be flooded / gone offline. Hence the backup load balancer will fill the void. The backup load balancer does not have a route for new users, and thus, blocks all new requests (hence blocks the attack) and safely allow trusted users to enter the network. (See details in section proposed solution)

Two very important questions arise here. Why are we adopting the heuristic approach? And why are we using load balancer to stop DoS or DDoS?

3.3.4 - Why Adopting the Heuristic Approach?

Wikipedia defines heuristic as technique designed for solving a problem more quickly when classic methods are too slow, or for finding an approximate solution when classic methods fail to find any exact solution [89].Usually heuristic algorithms are used for problems that have low time complexity and cannot be easily solved [90]. Due to the lack of a unified algorithm (P class) or method to solve the problem of multifaceted DDOS attacks, the problem falls into the NP class. The variety of the DDoS attack unexpected-parameters makes the problem non-deterministic. Class NP consists of all those problems whose solution can be found in polynomial time on a non-deterministic Turing machine. Yet, such machine does not exist. A zero-day attack is a threat aimed at exploiting software or hardware vulnerability before the vendor becomes aware of it and before it becomes known to security experts. These attacks are among the hardest to mitigate, and thus, the use of heuristic technique becomes a must.

3.3.5 - Why Using Load Balancers to Stop DDoS Attacks?

Using a firewall is simply out of the equation as it requires simple pre-defined rules to function. Based on these rules it will take the decision of letting the traffic go or block it.

So the remaining equation is IPS vs Load Balancers. We choose load balancers to specifically stop DOS attacks because we take advantage of their smart way to manage traffic. A DOS attack does not always have a malicious pattern. It could simply be a normal request, repeated so many times, really fast. Hence, an IPS could let it pass. On the other hand, the IPS does not know how to manage traffic in an effective way. While load balancers have memory queues that save requests and split them over other peered load balancers or target servers. This is an advantage for load balancers as they can withhold and survive a huge amount of requests while the IPS might be subject to DOS attack itself and fail quickly.

The main advantage of load balancers is, not only they can be peered together, but they are also able to process requests, change their routes based on a certain condition, and even simply drop them if we wanted. Peered load balancers are also able to forward request to other load balancers that have different conditions and different routes to different servers. Taking advantage of all these features, load balancers will allow us to collect and analyze as many requests as possible, and allow them all to pass into our network.

In case of an attack, the gateway load balancer will fail and hence redirect all traffic to its peered load balancer that have a set of conditions in order to process the requests or drop them. This way, the second load balancer becomes the gateway, and it will only allow requests from trusted users and therefore surviving the attack by dropping all other untrusted requests.

3.3.6 – Understanding the IoT Botnet Structure

The term botnet originates from the terms “roBOT NETwork”. A bot is a malicious-software installed into a PC or a device that runs automated attacks over a network. The infected device itself is knows as a bot or zombie. A botnet is a network of malware-infected devices (zombies), which are controlled by cybercriminals. Intended attackers search for vulnerable internet connected devices, infect them with spam and malware to launch distributed denial-of-service attacks. Up until recently, botnets were formed of contaminated PCs and laptops. What made recent attacks (kerbsOnSecurity.com, OVH, Dyn) different is that the botnets were embraced of what has become recognized as the “Internet of Things” devices.

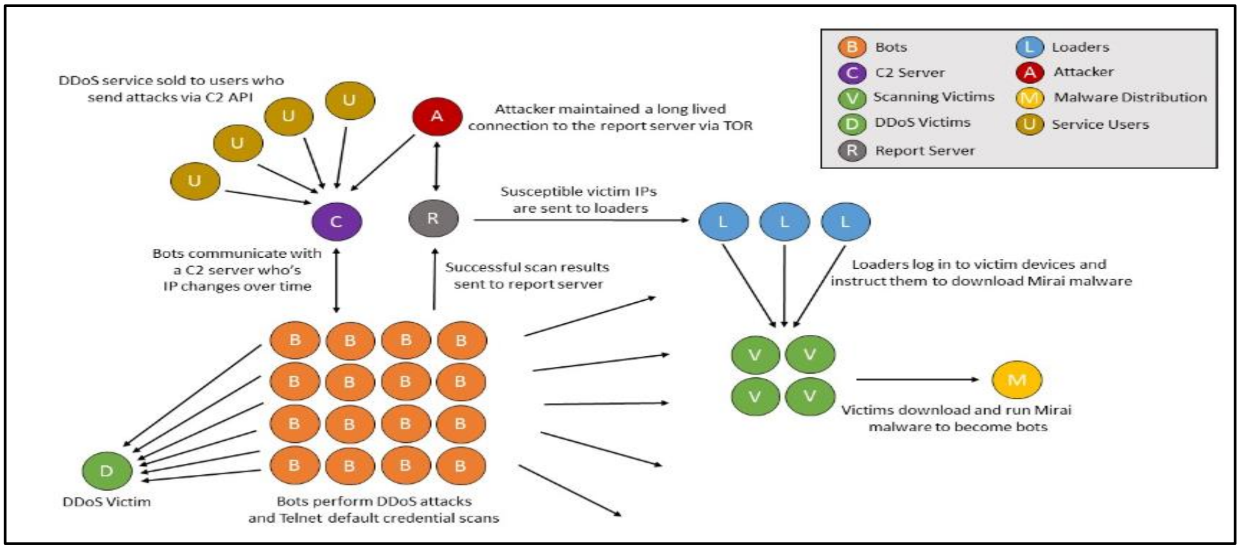
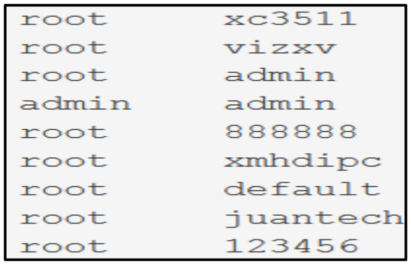

B Bots
C C2 Server
V Scanning Victims
D DDoS Victims
R Report Server
L Loaders
A Attacker
M Malware Distribution
U Service Users
DDoS service sold to users who send attacks via C2 API
U
U
U
U
A
Attacker maintained a long lived connection to the report server via TOR
C
R
Susceptible victim IPs are sent to loaders
L
L
L
Bots communicate with a C2 server who's IP changes over time
Successful scan results sent to report server
Loaders log in to victim devices and instruct them to download Mirai malware
B
V
V
V
V
M
Victims download and run Mirai malware to become bots
D
DDoS Victim
Bots perform DDoS attacks and Telnet default credential scans

Figure 3.7 - Structure of IoT botnet
Source: https://neuropuff.com/c/technology/p/mirai-c-c-botnets/

The infected IoT devices or bots (DVRs, Webcams, Baby monitors, electronic thermostat etc.) are used to perform command based DDoS attacks. Bots communicate with each other with adversaries through Command Control (C2) Servers to either flood various types of traffic or change their IP addresses for different intervals. The Scanning tools are used to scan IoT devices for vulnerabilities For Instance, in the case of Mirai botnets bots were found communicating with the C2 server to scan across TCP port 23 and port 2323 and use different brute force technique for guessing passwords and store them in the reporting server. The attacker uses these guesses through the C2 server to plague the IoT devices with the malicious code (malware).



root	xc3511
root	vizxv
root	admin
admin	admin
root	888888
root	xmhdipc
root	default
root	juantech
root	123456

Figure 3.8 – Weak Default Passwords

Figure 3.8 above shows a sample of weak default passwords grabbed as a result of Mirai's scanning vulnerable IoT devices. At this moment, the loader queries these credentials to log on to vulnerable IoT devices and order them to download, execute the malware and launch the attack. Some IoT devices have no permanent storage media; the malware stays resident into their volatile memory (RAM) and hold onto launching attacks as long as they are not being reset.

3.3.7 - Proposed Solution

Our solution consists of keeping track of our visitor's identification (such as IP), and separate our regular visitors from new ones. Why does IoT device IP such as electronic thermostat, printer, or webcam request Tweeter or Facebook page access? This could be achieved by adding a smart router / load balancer that distribute data depending on the visitor's reputation as illustrated in figure 3.9 and figure 3.10 respectively. Traffic is received by the main load balancer which, based on reputation and heuristics, sends the user via the new visitor route or the trusted visitor route. In case of a DOS attack, the Main Load Balancer will be flooded and gone offline. Hence the backup load balancer will fill the void. The backup load balancer does not have a route for new users, and thus, blocks all new requests (hence blocks the attack) and safely allow trusted users to enter the network. This solution saves services from going offline, and increase clients trust by staying available regarding the attack [91]. However, the downside is that during the attack phase, some legitimate new visitors will be left out as well since they are not in the trust zone yet.

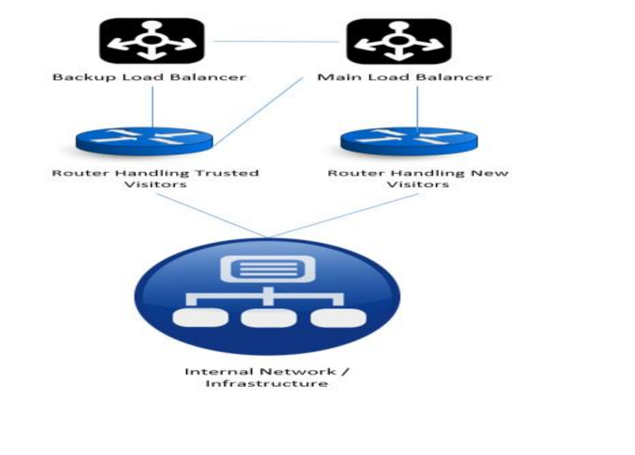

Backup Load Balancer
Main Load Balancer
Router Handling Trusted Visitors
Router Handling New Visitors
Internal Network / Infrastructure

Figure 3.9 - Proposed Solution Infrastructure

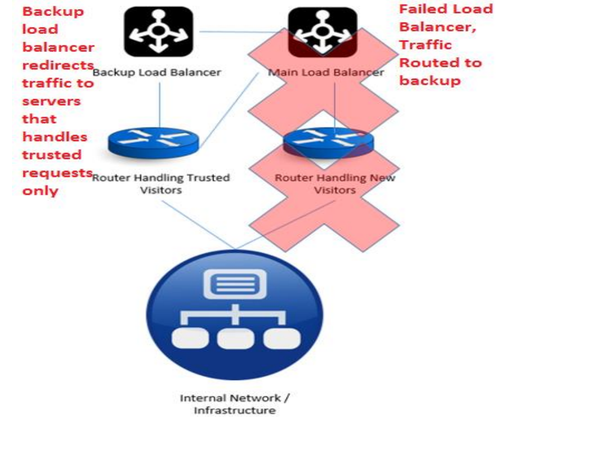
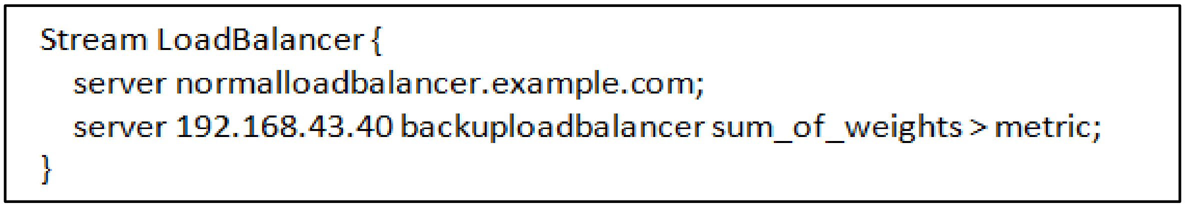

Backup load balancer redirects traffic to servers that handles trusted requests only
Backup Load Balancer
Main Load Balancer
Failed Load Balancer, Traffic Routed to backup
Router Handling Trusted Visitors
Router Handling New Visitors
Internal Network / Infrastructure

Figure 3.10 - Proposed Solution DOS Attack

Our solution consists of two main parts: the physical infrastructure of the network, and the algorithm identifying legitimate trusted traffic from the poisoned / infected traffic. For the infrastructure, we use two Ubuntu servers. Each of them is running HAPROXY to serve as a load balancer. Having two (or more) load balancers is a critical step within this solution, as they are configured in a failover manner (load balancer peering), where the main load balancer is always functioning until an attack occurs (see configuration in figure 3.11). Once an attack occurs, naturally, the first load balancer will be flooded with requests and hence the backup load balancer will take place. The trick is that the second load balancer only redirects traffic to servers that process only trusted requests and ignore all the others [92], hence it will hardly be processing any dummy requests and thus remain functioning normally.

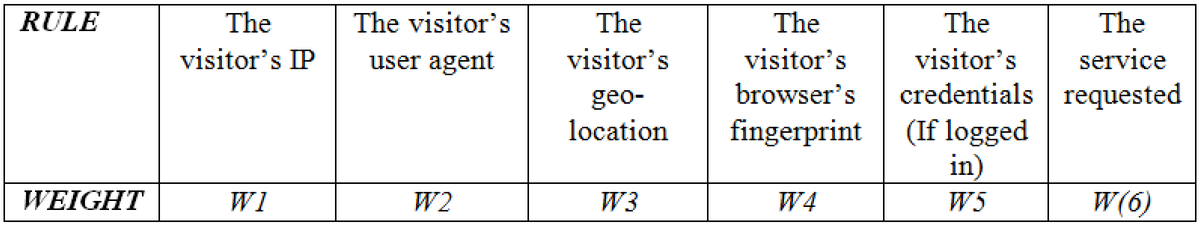


```
Stream LoadBalancer{
   server normalloadbalancer.example.com;
   server 192.168.43.40 backuploadbalancer sum_of_weights > metric;
}
```

Figure 3.11 - Load Balancers Configuration

The second part of the solution is our algorithm to classify traffic as trusted or untrusted. For that, we test the request by having it pass through a set of rules, where each rule has its own weight. For each rule the request is successfully tested, the request gains the weight of this rule in an incremental manner. Once it is done, if the request has a specific total number of points, the request is then trusted. For instance, we define a set of rules as shown in the table below:

Table 3.4 – Load Balancer Rules



RULE	The visitor's IP	The visitor's user agent	The visitor's geo-location	The visitor's browser's fingerprint	The visitor's credentials (If logged in)	The service requested
WEIGHT	*W1*	*W2*	*W3*	*W4*	*W5*	*W(6)*

Where: weight *W(x)* is a defined integer.

Two approaches can be used to assign weights to rules: The first is based on human supervised training, trials and errors and the second is adaptive. (See Section 3.3.8.2 Assigning Weights to Rules)

3.3.8 - Methodology

1- The request is sent from an IP that access our services daily. The request gains *W1* point.
2- The request's User Agent has previously accessed our services. The request gains *W2* point (total is now *W1* + *W2*).
3- The visitor identified is in the same country our services is, and the same location his IP usually requests from, the request gains W3 (total is now *W1* + *W2* + *W3*)
4- The visitor's browser finger print is not identified. The number of points remains W1 + W2 + W3.

5- The visitor is not logged in. The number of points remains *W1 + W2 + W3*.

As a result, if *W1 + W2 + W3 > X*, the request is marked as trusted and thus allowed to pass.

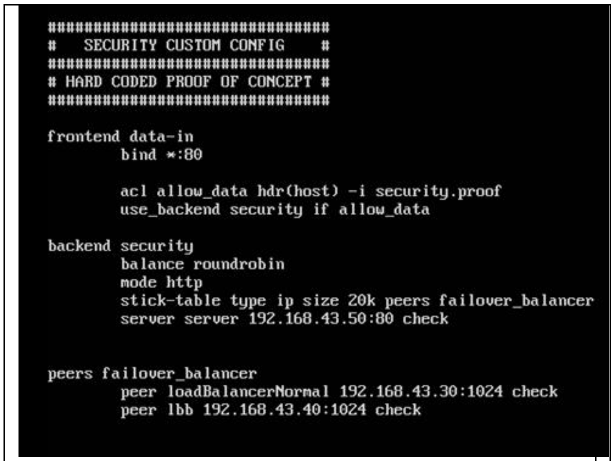
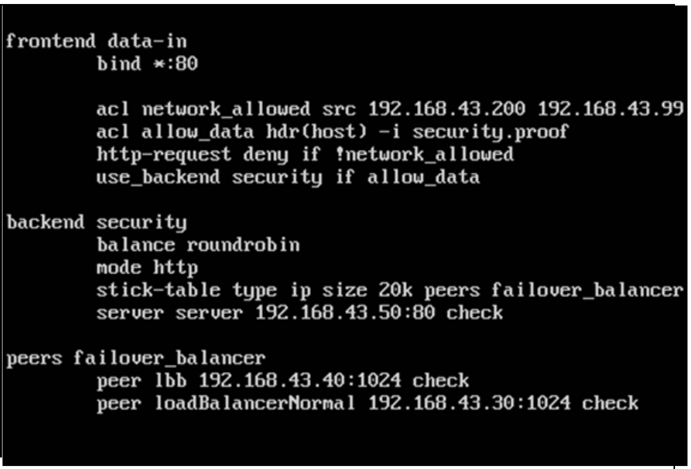
Else, the request is dropped, and further requests from the same source are no longer needed to be tested, they are dropped automatically. Note: the rules and weight are to be customized per service, as each service can provide more evidence and different ways of identifications of legitimate requests.

3.3.8.1 - Configuring Load Balancers

HAProxy (High Availabily Proxy) is free and open source TCP/HTTP Load balancer software run under the Linux platform.

Our network, consisting of two load balancers: a main load balancer (allowing all traffic to be forwarded to the services) and a backup load balancer (protecting our services from attacks, allowing only requests from trusted sources).

The two load balancers are configured as follows:

	
Main Load Balancer: Allowing all traffic to be forwarded to our services.	Backup load balancer, allowing traffic only from trusted sources.

```
###############################
#   SECURITY CUSTOM CONFIG    #
###############################
# HARD CODED PROOF OF CONCEPT #
###############################

frontend data-in
        bind *:80

        acl allow_data hdr(host) -i security.proof
        use_backend security if allow_data

backend security
        balance roundrobin
        mode http
        stick-table type ip size 20k peers failover_balancer
        server server 192.168.43.50:80 check


peers failover_balancer
        peer loadBalancerNormal 192.168.43.30:1024 check
        peer lbb 192.168.43.40:1024 check
```

```
frontend data-in
        bind *:80

        acl network_allowed src 192.168.43.200 192.168.43.99
        acl allow_data hdr(host) -i security.proof
        http-request deny if !network_allowed
        use_backend security if allow_data

backend security
        balance roundrobin
        mode http
        stick-table type ip size 20k peers failover_balancer
        server server 192.168.43.50:80 check

peers failover_balancer
        peer lbb 192.168.43.40:1024 check
        peer loadBalancerNormal 192.168.43.30:1024 check
```

Figure 3.12 – Sample Load Balancers Configuration

Figure 3.12 displays the configuration of both load balancers. We can see that on their input (frontend data-in), both load balancers are bound to port 80 as our test service is web based. Once the load balancers receive a request targeting the endpoint “security.proof”, the code under “backend security” will run.

However, we can see that on the backup load balancer (the right side configuration of figure 3.12), a variable is introduced called “network_allowed” with the source (“src”) parameter, marking 192.168.43.200 as allowed (trusted client). This is the IP of tester 1. In this particular load balancer (backup load balancer), before using the “backend security” the load balancer will deny the request if its source is not in the “network_allowed” list.

In backend security, we notice that the requests are forwarded to the web server on port 80 holding the IP 192.168.43.50. Naturally, the check parameter keeps knowledge whether or not the server is online and available on the selected port.

Finally, the failover_balancer defines the peered load balancers. This section helps the traffic to be redirected from the normal load balancer to the backup load balancer when the normal load balancer fails. When a DOS attack occurs, if the normal load balancer fails, the backup load balancer will take over and drop all requests from untrusted sources, allowing only trusted sources to reach the service and hence protecting it from malicious attacks.

3.3.8.2 - Assigning Weights to Rules

Two methods can be used to assign weights to rules:

a) Pre-assigned: This method is based on human supervised training, trials and error. The security administrator has to be fully aware of all network activities, OS configuration and services vulnerabilities.

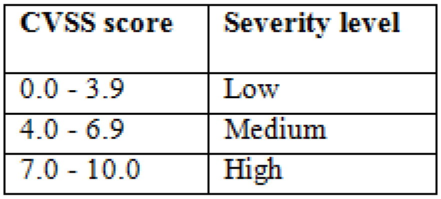
b) Adaptive: This method is considered more robust in which the weights of the load balancer rules are determined based on polling several vulnerability metrics from the internal networks after conducting vulnerability scanning analysis.

Security metrics are indicators used to provide contextual quantitative measure for the security characteristics of an information system. To improve the security of a system, we have to measure it. The Common Vulnerability Scoring System (CVSS) was designed by a team of security professional and by the National Institute of Standard and Technology (NIST). It is a vulnerability scoring system designed to provide a standardized and open framework for rating software vulnerabilities. The Common Vulnerability and Exposure (CVE) is the industry standard for vulnerabilities and exposure names, namely the CVSS and the Common Weakness and Enumeration (CWE) which provides a list of software weaknesses.

To determine the current and historical vulnerabilities associated with our network services, applications, and operating systems we used Nessus vulnerability scanner. We are interested in medium and high vulnerability scores to raise alert of the target service, to determine the attack paths that potential attackers are trying to exploit and thus, to be able to figure out the weight. Nessus can be fed by a variety of vulnerability suppliers like CVE, OSVDB, Cert, NVD to calculate the severity vulnerability score of our custom made network.

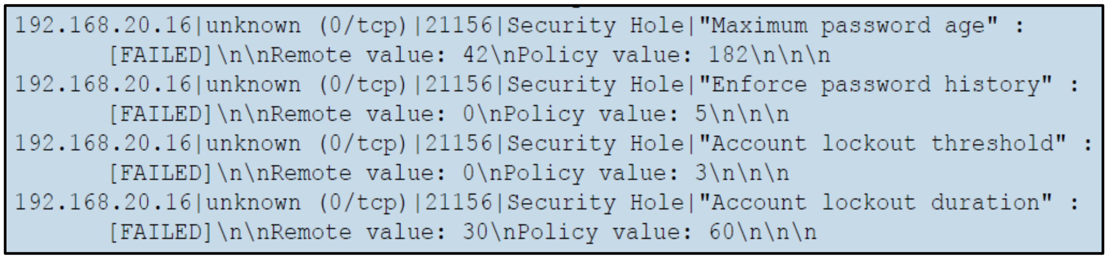
Nessus assigns the following severities to vulnerabilities found based on the CVSS score:

Table 3.5 - CVSS Severity Level



CVSS score	Severity level
0.0 - 3.9	Low
4.0 - 6.9	Medium
7.0 - 10.0	High

Sample output of Nessus Scanning detected non-compliant results



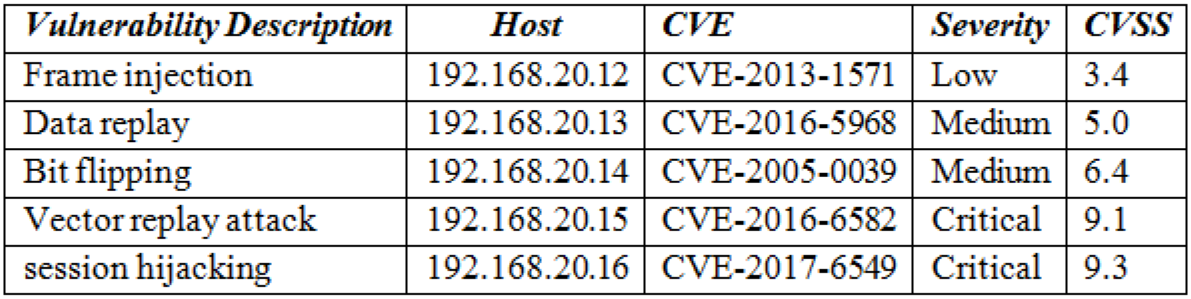
```
192.168.20.16|unknown (0/tcp)|21156|Security Hole|"Maximum password age" :
      [FAILED]\n\nRemote value: 42\nPolicy value: 182\n\n\n
192.168.20.16|unknown (0/tcp)|21156|Security Hole|"Enforce password history" :
      [FAILED]\n\nRemote value: 0\nPolicy value: 5\n\n\n
192.168.20.16|unknown (0/tcp)|21156|Security Hole|"Account lockout threshold" :
      [FAILED]\n\nRemote value: 0\nPolicy value: 3\n\n\n
192.168.20.16|unknown (0/tcp)|21156|Security Hole|"Account lockout duration" :
      [FAILED]\n\nRemote value: 30\nPolicy value: 60\n\n\n
```

Figure 3.13 – Sample Output of Nessus Results

Querying Nessus output using Pandas tool to determine CVE, CVSS and severity level we got:

Hosts = df1.groupby(“Desc’,'Host').agg ({CVE}, {Severity},{'CVSS'})

Table 3.6 – CVE Query Output



Vulnerability Description	*Host*	*CVE*	*Severity*	*CVSS*
Frame injection	192.168.20.12	CVE-2013-1571	Low	3.4
Data replay	192.168.20.13	CVE-2016-5968	Medium	5.0
Bit flipping	192.168.20.14	CVE-2005-0039	Medium	6.4
Vector replay attack	192.168.20.15	CVE-2016-6582	Critical	9.1
session hijacking	192.168.20.16	CVE-2017-6549	Critical	9.3

As a result we correlate vulnerability assessments with load balancer data to determine the weight. The rule weight assigned to the load balancer should be inversely proportional to the CVSS score. Recorded high CVSS scores of services require assigning low weights to rules correlated to those services and thus, the load balancer must not allow the corresponding requests to pass to sever cluster for execution.

3.3.9 - DDoS Attack Simulation and Results

3.3.9.1 - Network Setup and Customization

In order to simulate a DDoS attack and test the reliability of our solution, we made use of the network lab of the Information and Communications Technology department at the American University of Science and Technology.

We customized the network and divided it into two zones (External and Internal), configured the load balancers and integrated them into the network entry points (See figure 9 below).

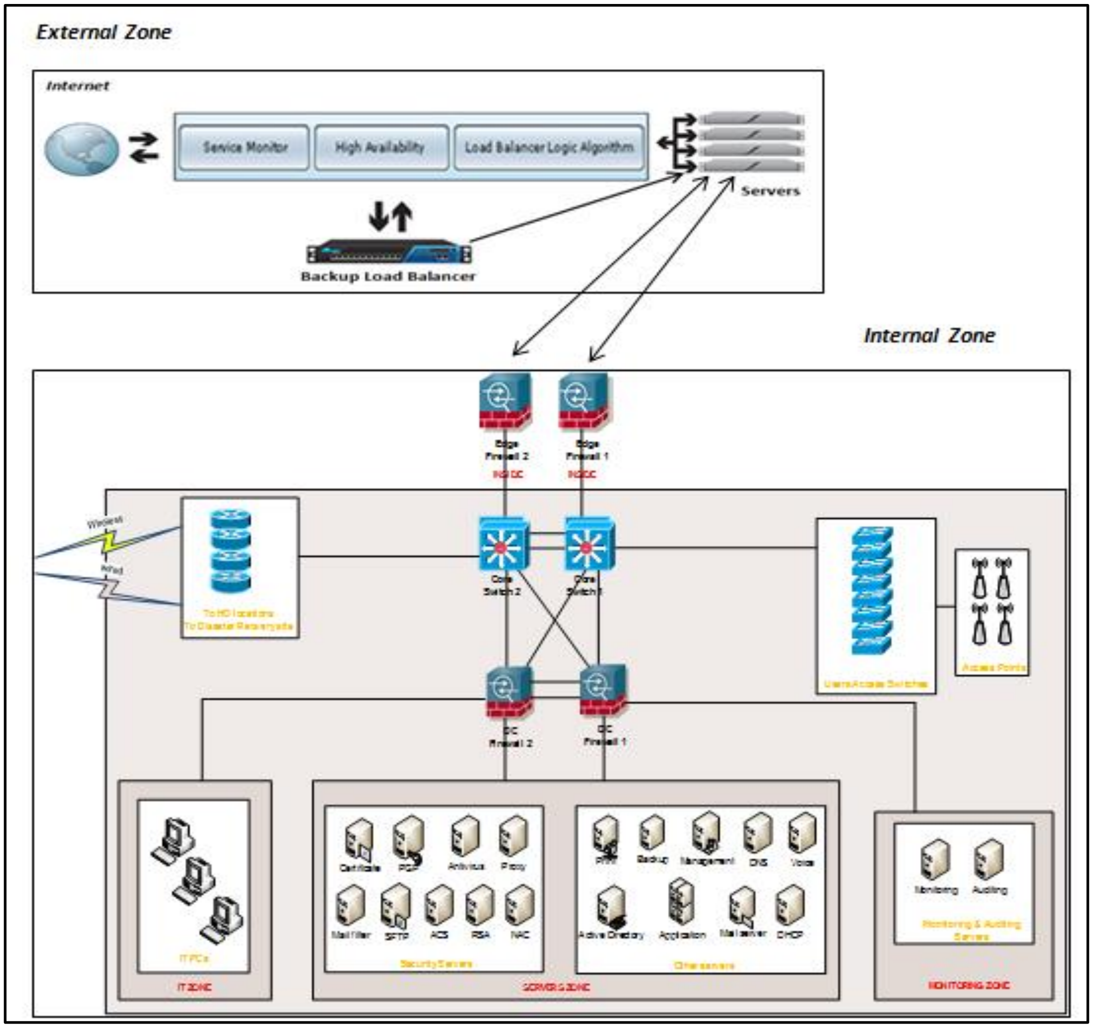


Figure 3.14 – Network Architecture

3.3.9.2 - External Zone

The external zone was configured as follows:

1- A main load balancer used to forward all traffic in a timely manner to the webserver. The main load balancer is called Normal Load Balancer (LBN) on our network and it was assigned the IP of: 192.168.43.30

2- A backup load balancer was also configured. The backup load balancer is peered with the main load balancer. Its job is to forward only the trusted requests to the server and drop all other foreign requests. It was assigned the IP 192.168.43.40

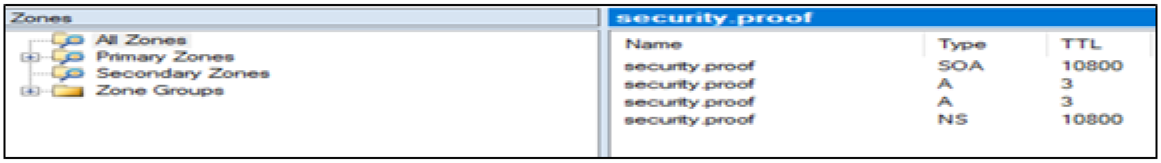
3- A DNS server was added to give the website a domain that the two load balancers will host together, peered one to another for failover. The DNS IP is 192.168.43.99

4- A web server considered as the main server providing the service to users. The webserver's IP is 192.168.43.50

5- A client computer, tagged as untrusted (new visitor). The untrusted client IP is 192.168.43.201

6- A client computer, tagged as trusted. The trusted client IP is 192.168.43.200

In addition, two computers were used to perform an attack over used domain assigned in the DNS server as security.proof



security.proof

Name	Type	TTL
security.proof	SOA	10800
security.proof	A	3
security.proof	A	3
security.proof	NS	10800

Figure 3.15 – DNS Security Proof

3.3.9.3 - Internal Zone

The internal zone was customized as below to mitigate the attack or lessen its impact in case the external zone (our solution) fails to handle it.

- The Edge Firewall internal zone physical interface is connected to the Layer 3 Core switch physical interface.
- Redundant aggregator routers are connected to the core switch. For example, two routers are used for the wired connection and the other two routers are used for the wireless connection. Head office locations and disaster locations are connected to these aggregator routers via an encrypted VPN tunnel connection.

- Physical servers, blade servers, network and security appliances etc. are physically connected to the core switches.
- User access switches or distribution switches are physically connected to the core switches. PCs, printers, faxes and other peripherals are physically connected to these layer 2 switches.
- The wireless access points are connected to the user access switches. The wireless LAN controllers (WLC) are connected to the core switches and are used to monitor the access points.
- An internal firewall or Data Center (DC) firewall is also implemented to segregate the internal corporate zones logically. For example, IT zone, servers zone, monitoring and management zone.

3.3.9.4 - Attack Simulation

In order to instantiate a Denial of Service attack against our network, we made use of the following tools:

1) The open source Low Orbit Ion Cannon (LOIC) platform. LOIC was developed by Praetox Technologies to conduct network stress testing, Denial of Service attack as well as Distributed Denial of Service (DDoS) attacks. It has been widely used by Anonymous as DDOS tool since it can generate huge amount of illegitimate TCP, UDP, or HTTP network traffic which causes performance degradation and potentially a service shut down. Over 30000 downloads were recorded during the month of December 2010 when Anonymous organized attacks on the websites of companies and organizations that opposed Wikileaks [93].

2) The bash script Pentmenu developed by pentbox. It is designed to perform network pen testing functions. It is commonly installed on most linux distributions.

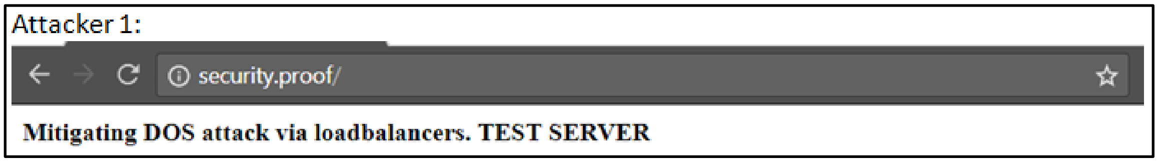
3) The Slowloris DDos attack software developed by Robert “Rsnake” Hansen. It lets a single computer to take down a web server like IIS, Apache 1.x and 2.x.

Two computers were used to attack the domain (load balancer):

1- A computer running windows and LOIC (Low Orbit Ion Cannon) [https://sourceforge.net/projects/loic/] (Attacker 1).
2- Another computer running Kali Linux and using Pentmenu to perform the attack [https://github.com/GinjaChris/pentmenu] (Attacker 2).

Two other computers (tester 1 and tester 2) were used to check the status of the service when being attacked. Tester 1 is considered a trusted client / visitor.

The first step was to check if the service endpoint is reachable by attacker 1, attacker 2, client 1 and client 2. The results were as following:



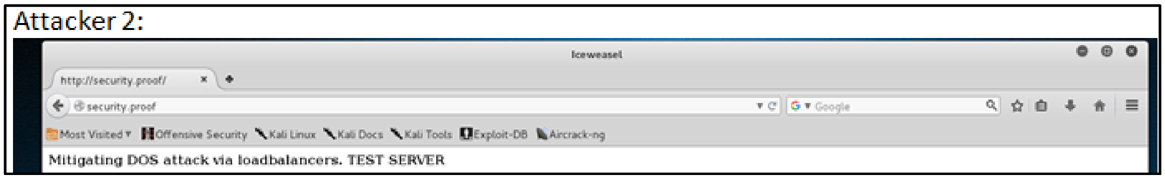
Attacker 1:

security.proof/

Mitigating DOS attack via loadbalancers. TEST SERVER

Figure 3.16 – Screenshot of Testing Endpoint from Attacker 1's PC

Figure 3.16 shows that the endpoint (http://security.proof) is reachable by attacker 1 and all of her/his requests are processed normally before the attack.



Attacker 2:

Iceweasel

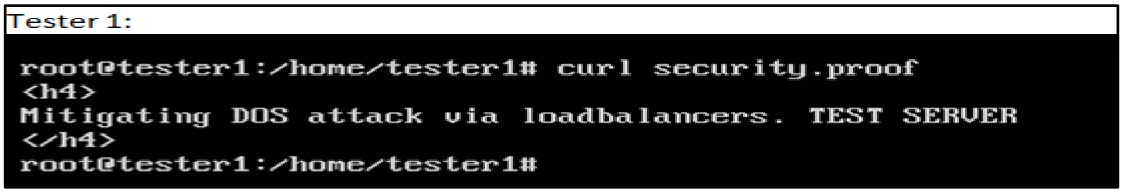
http://security.proof/

security.proof

Mitigating DOS attack via loadbalancers. TEST SERVER

Figure 3.17 – Screenshot of Testing Endpoint from Attacker 2's PC

Figure 3.17 illustrates that the endpoint (http://security.proof) is reachable by attacker 2 and all of her/his requests are processed normally before the attack.

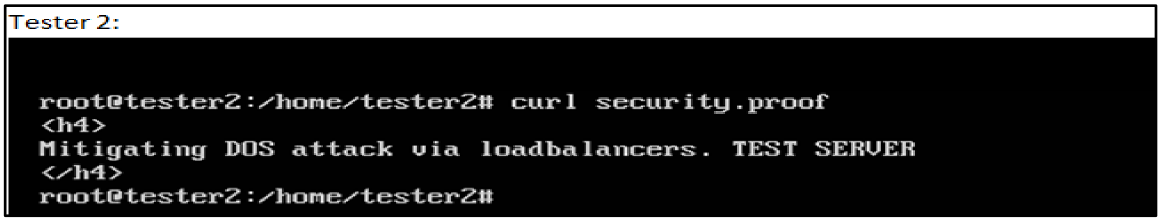


Tester 1:

```
root@tester1:/home/tester1# curl security.proof
<h4>
Mitigating DOS attack via loadbalancers. TEST SERVER
</h4>
root@tester1:/home/tester1#
```

Figure 3.18 – Screenshot of Testing Endpoint from Tester 1's PC

Figure 3.18 demonstrates that the endpoint (http://security.proof) is reachable by tester 1 and all of her/his requests are processed normally before the attack.



Tester 2:

```
root@tester2:/home/tester2# curl security.proof
<h4>
Mitigating DOS attack via loadbalancers. TEST SERVER
</h4>
root@tester2:/home/tester2#
```

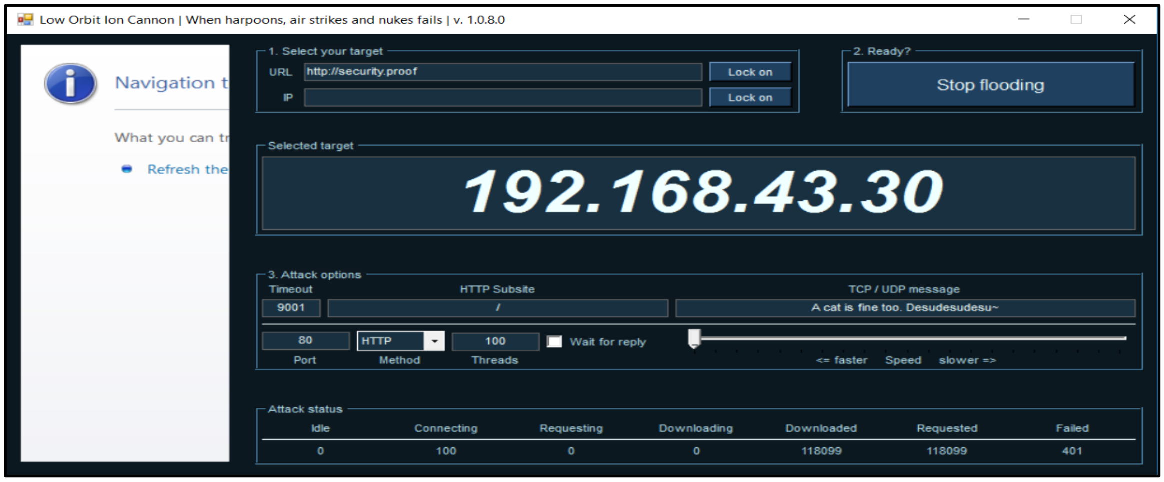
Figure 3.19 – Screenshot of Testing Endpoint from Tester 2's PC

Figure 3.19 shows that the endpoint (http://security.proof) is reachable by tester 2 and all of her/his requests are processed normally before the attack.

The figures 3.16, 3.17, 3.18 and 3.19 demonstrate that all of the parties are able to reach and request the server normally without any issues.

Now that the site is reachable by all parties, attacker 1 and attacker 2 start targeting the endpoint.

We have run multiple instances of both attacking software (LOIC and Pentmenu) in order to make the load balancer fail as shown below:


Low Orbit Ion Cannon | When harpoons, air strikes and nukes fails | v. 1.0.8.0
Navigation t
What you can tr
Refresh the
Lock on
Lock on
Stop flooding

1. Select your target
URL http://security.proof
IP

2. Ready?

Selected target
192.168.43.30

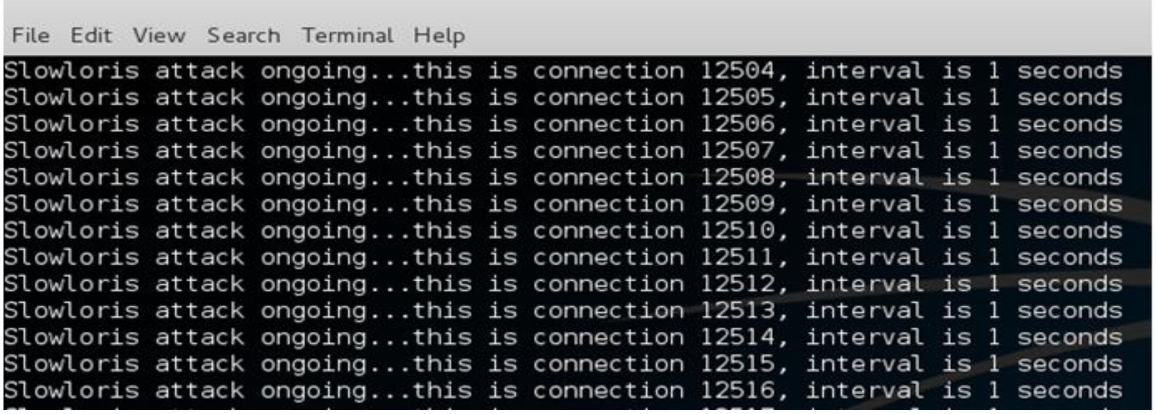
3. Attack options
Timeout 9001
HTTP Subsite /
TCP / UDP message A cat is fine too. Desudesudesu~
Port 80
Method HTTP
Threads 100
□ Wait for reply
<= faster Speed slower =>

Attack status

Idle	Connecting	Requesting	Downloading	Downloaded	Requested	Failed
0	100	0	0	118099	118099	401

Figure 3.20 – LOIC Used by Attacker 1

Figure 3.20 is a screenshot of LOIC used by attacker 1, targeting the endpoint (http://security.proof) with a massive amount of requests (flooding) as fast as possible.



```
Slowloris attack ongoing...this is connection 12504, interval is 1 seconds
Slowloris attack ongoing...this is connection 12505, interval is 1 seconds
Slowloris attack ongoing...this is connection 12506, interval is 1 seconds
Slowloris attack ongoing...this is connection 12507, interval is 1 seconds
Slowloris attack ongoing...this is connection 12508, interval is 1 seconds
Slowloris attack ongoing...this is connection 12509, interval is 1 seconds
Slowloris attack ongoing...this is connection 12510, interval is 1 seconds
Slowloris attack ongoing...this is connection 12511, interval is 1 seconds
Slowloris attack ongoing...this is connection 12512, interval is 1 seconds
Slowloris attack ongoing...this is connection 12513, interval is 1 seconds
Slowloris attack ongoing...this is connection 12514, interval is 1 seconds
Slowloris attack ongoing...this is connection 12515, interval is 1 seconds
Slowloris attack ongoing...this is connection 12516, interval is 1 seconds
```

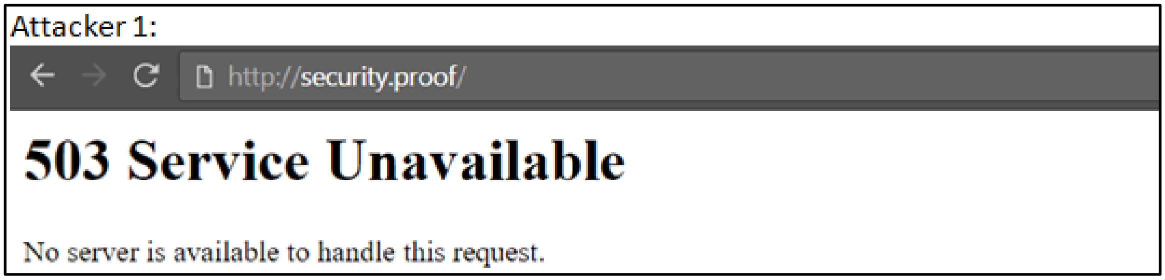
Figure 3.21 – Pentmenu Used by Attacker 2

Figure 3.21 is a screenshot of a Pentmenu used by attacker 2, targeting the endpoint (http://security.proof) with a Slowloris attack. The Slowloris DdoS software is integrated inside Pentmenu bash script.

Slowloris attack is opening as many connections as possible with the server in order to make the service go down.

Figure 3.22 shows that requests have already started to fail. Meaning that, for the attacker, the load balancer failed. This will eventually lead for the backup load balancer to take over.

Once the load balancer was down, we tried again to reach the endpoint from each party on the network. Results came as the following:

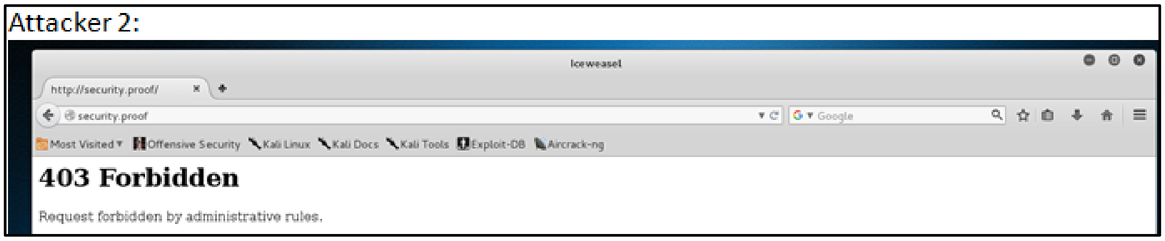


503 Service Unavailable

No server is available to handle this request.

Figure 3.22 – Screenshot of Testing Endpoint from Attacker 1's PC

Figure 3.22 demonstrates that the endpoint (http://security.proof) is no longer reachable by attacker 1 and all the requests are dropped and are no longer forwarded to the service. It might look like the service went offline and the attack was successful, but our other testers will prove otherwise.

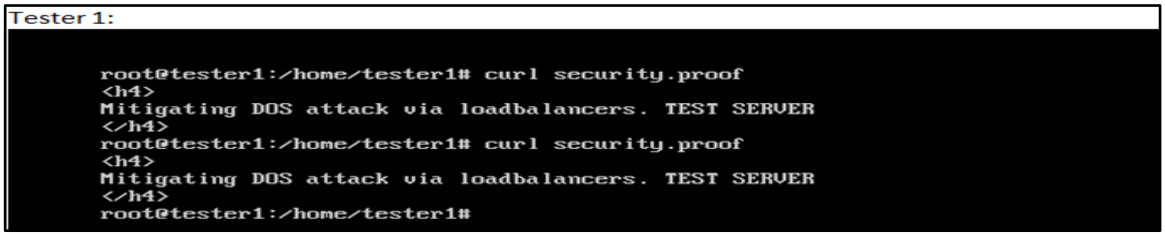


403 Forbidden

Request forbidden by administrative rules.

Figure 3.23 – Screenshot of Testing Endpoint from Attacker 2's PC

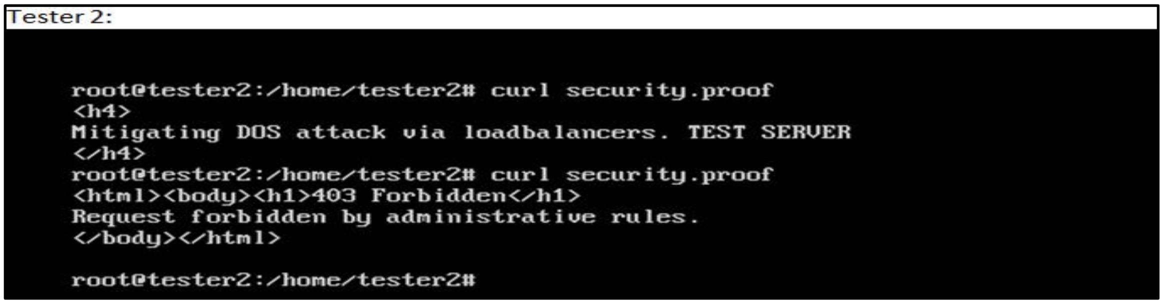
Figure 3.23 shows that our endpoint (http://security.proof) is no longer reachable by attacker 2 and all of his requests are dropped and no longer forwarded to the service. It might look like the service went offline and the attack was successful, but our other testers will prove otherwise.



```
root@tester1:/home/tester1# curl security.proof
<h4>
Mitigating DOS attack via loadbalancers. TEST SERVER
</h4>
root@tester1:/home/tester1# curl security.proof
<h4>
Mitigating DOS attack via loadbalancers. TEST SERVER
</h4>
root@tester1:/home/tester1#
```

Figure 3.24 – Screenshot of Testing Endpoint Tester 1's PC

Figure 3.24 reveals that the endpoint (http://security.proof) is still alive and functioning normally. And since Tester 1 is marked as a trusted user, she/he was still able to reach our service and use it normally without any interruption.



Tester 2:

```
root@tester2:/home/tester2# curl security.proof
<h4>
Mitigating DOS attack via loadbalancers. TEST SERVER
</h4>
root@tester2:/home/tester2# curl security.proof
<html><body><h1>403 Forbidden</h1>
Request forbidden by administrative rules.
</body></html>

root@tester2:/home/tester2#
```

Figure 3.25 – Screenshot of Testing Endpoint from Tester 2's PC

Figure 3.25 indicates that the service is no longer accepting requests from tester 2. Although tester 2 is not an attacker, yet, she/he is a new visitor and not marked as trusted. In order to protect our network, the service closed during the attack and turned into a trust based network allowing only trusted visitors to pass (tester 1).

3.3.9.5 – Results

At first, the endpoint or service (http://security.proof) was reachable by all ends / parties on the network: attacker 1, attacker 2, tester 1 and tester 2.

Tester 1 is the only node marked as a trusted client / visitor. Attacker 1 and attacker 2 start the attack together in order to try to have the service failed. The attack seemed to have succeeded and we can clearly see that attacker 1 and attacker 2 are no longer able to reach the service. However, in reality, the attackers have succeeded to take down the first load balancer which routes all traffic to our service. On the other hand, the backup load balancer took over, allowing only trusted visitors' requests to pass whilst attacker 1 and 2 believed that the service is down. To prove this claim, tester 1 visited the service and the result shows that tester 1 was still able to access the service normally without any interruption.

Alternatively, despite the fact that tester 2 is not an attacker; she/he was not able to reach the service either. That's because only trusted sources are now allowed to pass and tester 2 is not marked as trusted.

3.4 – Conclusion

Security experts and network administrators used to act based on their proficiencies and practices to mitigate network attacks rather than objective metrics and models. This study provided two mechanisms to protect smart city network. The first is predictive in nature relying on a quantitative approach used to measure the vulnerability level of the entire potential attack paths. Cumulative probabilities referring to high vulnerability level in a specific attack path will lead the system administrator to apply appropriate security measures a priori to potential attacks occurrence.

The second mechanism is defensive in nature relying on a heuristic approach associated with load balancers to mitigate DDoS attacks. Based on the fact that IOT devices can easily be exploited and controlled to perform DDOS attacks, we introduced a new way to mitigate DDOS attacks using an array of load balancers. Normal load balancer allows traffic to pass normally until an attack occurs, and backup load balancer uses weighted, dynamically and smartly assigned heuristics to determine whether or not to allow requests to pass. If the heuristic approach classifies the request as trusted, it will reach the protected services, otherwise it shall be dropped.

Over-all Thesis Conclusion and Recommendations

As cyber threats are constantly evolving, protecting our assets requires advanced cyber security mechanisms. By providing real-time threat intelligence, system administrators can infer protection solutions against cyber threats across numerous systems attack vectors. Analyzing information of both internal and external data collected from intelligent devices, servers, networks and applications, stipulates the key factor to identify the risks and threats of the smart city different dimensions. Security experts now become aware of new insights and methods to categorize and detect cyber security threats, and if applied properly will lead to significant results. System administrators relying on their personal experience by using conventional security methods to defend network borders turned out to be a tedious and useless task. Thus, it becomes straightforward to use formal reasoning and proof tools to apply security and to show where security vulnerabilities may lie.

The study also proved that a new way to mitigate Denial of Service (DDoS) attacks using an array of load balancers to create a form of closed network. This solution incorporates two types of load balancers. The first is a normal one which allows traffic to pass normally until an attack occurs. The second is a backup that uses weighted, dynamically and smartly assigns heuristics to determine whether or not to allow requests to pass. If the heuristic approach classifies the request as trusted, it will reach the protected services, otherwise it shall be dropped. However, a proof of concept experiment was performed, DDoSing a webserver, proving that our algorithm is viable and effective.

Having implemented the solutions thoroughly explained in our research, pave the way to a new subset of recommendations. We suggest that if the following recommendations are aligned together with our solutions will definitely lead to significant and robust results in reducing systems vulnerabilities and mitigating attacks.

a) Secure by Design: Security measures should be integrated while designing the IoT products. Survey [31] showed that only 48% of companies focus on securing their IoT products from the beginning of the product development phase.
b) End-to-End protection architecture: Devices and systems must be tolerant to attack, they have to be kept operating under all circumstances.
c) Hardware embedded security: leads to more resilient systems and helps shortening the time needed for services to return to their original states after an attack.

Last but not least, it is worth mentioning that besides technology, another important aspect that we must account for when it comes to securing smart city systems is the human side. Attackers will always look for the weakest link. It is most likely that attack vectors include not only technology and application but also employees. Therefore, it is not sufficient that technology and application are designed from the ground up to be secure, it is also necessary that employees are aware of cyber security threats and well trained to act properly. Therfore, to prevent cyber-attacks, organizations must ensure that they educate consumers about the correct security procedures to be followed while using intelligent IoT systems.

Appendix A

In Reference to Chapter 2 Section 2.2

ICTO 2017: Information and Communication Technologies in Organizations and Society

Paris, March 16th and 17th, 2017

A PERVASIVE IOT SCHEME TO VEHICLE OVERSPEED DETECTION AND REPORTING USING MQTT PROTOCOL

Complete Research

Abstract

One particular concern that Public Safety Organization (PSO) must account for is the excess of speed of vehicles in motion. The high speed is typically responsible for a significant proportion of the mortality and morbidity that result from road crashes. According to the World Health Organization (WHO), "an increase in average speed of 1 km/h generally results in a 3% higher risk of a crash involving injury, with a 4–5% increase for crashes that result in fatalities"[94]. Various ineffective proposed methods and solutions have been implemented to control speed limits; for instance, Speed Detection Camera System (SDCS), Radio Detection and Ranging (RADAR), Light Detection and Ranging (LIDAR). This paper conveys an innovative, pervasive, effective and adaptable Internet of Things (IoT) system to detect and report vehicle overspeed as well as issuing tickets and fines. Our aggregated prototype is composed of five components: IoT vehicle on-board unit, Message Queuing Telemetry Transport (MQTT) broker, application logic server, data storage, and monitoring engine. A software simulation has been implemented and tested as a proof of concept. This novel technique is restricted only for governmental use since it surrogates the contemporary aforementioned speed detection systems paving the way toward a smarter and sustainable solution, and thus ensuring public safety.

Keywords: PSO, WHO, vehicle, overspeed, IoT, Radar, MQTT, ticket.

1- Introduction

Many reasons lead to fatal road crashes that occur disproportionally and are stretched from road state, car situation, weather condition, driver alertness to speed which remains a major safety concern on the nations' roadways. Vehicle high speed is considered a crucial factor that typically leads to a significant increase in the morbidity and mortality rate. Moreover, governments and PSOs invest enormous amount of money and human resources to provide efficient traffic surveillance systems which control the excess of speed and hence enforcing traffic speed laws. The (WHO) organization states that road traffic crashes are predicted to become the 7th leading cause of death by 2030. An adult pedestrian's risk of dying is less than 20% if struck by a car at 50 km/h and almost 60% if hit at 80 km/h. A 30 km/h speed zones can reduce the risk of a crash and are recommended in areas where vulnerable road users are common like residential and schools areas. According to the Association for Safe International Road Travel (ASIRT), nearly 1.3 million people die in road crashes each year, on average 3,287 deaths a day. Road crashes resulting from high speed cost $518 billion globally, constituting 1-2% of individual countries' annual Gross domestic product (GDP) [95].

Many speed devices are currently available: The RADAR, the LIDAR, and the Speed Detection Camera System (SDCS). However, the use of these speed control devices has not resulted in definitive conclusions about their effectiveness as stated in the next section.

Based on the above aforesaid facts, an ample and doable solution for high speed detection becomes a must. Hence, the research question rises up: "How can we provide an innovative, faultless, and effective scheme for overspeed detection?"

To combat the speeding problem, we propose an effective and innovative IoT scheme which depends on the following components: a software component, for instance, an application that relies on the MQTT lightweight protocol to report instantly the speed of the vehicle, and to issue a ticket when overspeed threshold value is defeated; an on-board unit integrated into the vehicles which is responsible for broadcasting the vehicle geolocation information as well as its speed to a processing server; a processing server to retrieve and calculate the speed limit taking into account several factors: state of the road, traffic congestion, weather conditions and the like.

The main contributions of this paper are: (a) Developing a new smart IoT solution which helps governmental authorities in supervising vehicle overspeed. (b) Issuing and collecting car tickets autonomously. (c) Implementing a cost efficient and accurate solution for overspeed road control. (d) Collecting geographical data which can be fed into a data mining engine to extract roads conditions. (e) Providing traffic descriptive statistics reports for some critical areas.

This paper starts with descriptive and inferential statistics about road traffic crashes, high death rates, prorated cost as well as its influence on the individual countries' annual GDP. Sections II, III, and IV, describe the related work, the proposed scheme, the design and implementation respectively. Sections V, VI, and VII expose results, performance, and conclusions and future work.

2 - Related Work

This section discusses and studies the traditional radar systems, their requirements, functionalities and their intrinsic drawbacks.

2.1- RADAR (Radio Detection and Ranging)

There are two types of radar that are commonly used in almost all countries by law enforcement personnel to measure the speed of moving objects. Their functionalities are based on Doppler shifts to measure the speed of vehicle.

Fixed high way radar: It calculates vehicle's speed by means of sensors and capturing still images. This type of radar is considered extremely expensive. Its cost ranges from $20000-$30000 [96].

Mobile inner town radars or radar guns: This device may be hand-held or vehicle mounted. It calculates vehicle's speed by means of sensors, and it needs an operator to capture the images.

Various limitations of the radar system are perceived and not limited to: a) User training and certifications are required, b)Installation and deployment requires planning and mathematical consideration for better field of view, c) Radar can take up to 2 seconds to lock on and hence, cannot detect two excessive speeders simultaneously, d) Large targets close to radar can saturate or hide other smaller objects; therefore, the radar fails to locate the vehicle, e) Radar-triggered cameras are imperfect and can result in tickets being generated for false readings, f) Human intervention is required. They have no mechanism of sending the captured images. The authorities have to make periodic stops to collect the films.

2.2- LIDAR (Light Detection and Ranging)

The Lidar system relies on the principles of time-of-flight of two or more short wave length laser pulses. Sweep for instance, is a $250 Lidar with range of only 40 meters [97]. If we calculate the number of Lidars in a country, their cost will be very high.

Some of the disadvantages are: a) Particles (dust, water) in air can limit range, b) Rounded surfaces, the color black, blue, and violet are poor reflectors, c) Alignment can cause severe error, d) Extreme sunlight can be damaging.

Other technologies can be used to avoid and defeat radar and Lidar systems:

- Laser detectors and radar detectors. They detect if the speed is being monitored and warn the driver.
- Laser jammers and radar jammers. They jam the laser and the radar signal and return a scrambled signal so as the radar speed camera cannot process [98].

2.3- SDCS (Speed Detection Camera System)

SDCS is a camera that uses image processing to detect traffic regulation violations. It can be mounted beside or over a road or installed in an enforcement vehicle.

Some of the perceived drawbacks are: a) SDCS method requires large database for storing video, therefore the cost of this method is higher than of the RADAR and LIDAR, b) In December, 2012, Speed Camera Contractor Xerox Corporation admitted that cameras they had deployed in Baltimore city were producing erroneous speed readings, and that 1 out of every 20 citations issued at some locations were due to errors c) One issue is the potential conflict of interest when private contractors are paid a commission based on the number of tickets they are able to issue.

The purpose of the speed camera program is to improve safety by reducing unsafe speed and must be persuasively and evidently communicated to the public. As a matter of facts, signs announcing the possible presence of speed cameras should be obviously posted throughout the enforcement area. To do otherwise, the suspicion that those cameras are being used mainly for revenue purposes rather than safety reasons [99].

Based on the above discussion, these traditional speed devices need to be replaced by an automated system having better precise outputs, less expensive, and exclude human factor.

3 - The Proposed Scheme

This section outlines our proposed system at a high level scope. The system is composed of the following phases: (a) Service’s registration and IoT device integration, (b) Speed and geo-location reporting (c) Overspeed detection and tickets’ issuing.

3.1- Service’s registration and IoT device integration

The IoT device must be integrated into every vehicle through the on-board Diagnostic (OBDII) plugin standard. After installation, this device automatically acquires the VIN (Vehicle Identity Number) and maps it to its Universal Unique Identifier (UUID). The combination of the VIN and the UUID ensures authentication and genuineness and thus, prevents device's counterfeiting. The first time the device is plugged in into the moving object, its mapping record is then posted to the server's database for registration and consequently, for post-matching. Since the database is managed by the government, then there exist a mapping among the three attributes VIN, plate number, and the vehicle's owner. If the IoT device has been plugged out intentionally or unintentionally then the server can detect the action as described technically in section 4.3 and consequently, the government applies the appropriate measures.

3.2- Speed and geo-location reporting

The IoT device reports repeatedly the speed of the car including its geo-location for a configurable time interval. In order to use the full potential of IoT paradigm, the device reports the data to the main server using the lightweight protocol MQTT over the mobile network. MQTT is an open-source protocol for passing messages between multiple clients through a central broker. The MQTT architecture is broker based, and uses long-lived outgoing TCP connection to the broker. MQTT can be used for two way communications over unreliable networks. It is also compatible with lower consumption devices [100]. In our system, each MQTT message is composed of the vehicle's speed, its geo- location, its VIN, and the date & time. In case the TCP connection is disrupted then all MQTT messages are stored temporarily on a secondary storage device. When the IoT device is reconnected, the stored messages are republished again to the server. This ensures service availability at any time.

3.3- Overspeed detection and tickets' issuing

To be able to issue overspeed ticket, there should be a mechanism to map the vehicle's geo-location to the exact street name or number. Many such services are available in the cloud through well-defined Application Programming Interface (API). But due to the high number of expected transactions published constantly to the server, a bandwidth problem might arise. Therefore, a dedicated server is developed to perform reverse geo-location offline.

Two alternative models have been proposed for overspeed detection: the static model and the adaptable model. The first is implemented by comparing the current vehicle's speed and its associated attributes to a speed record located into a database server provided by the government. The second is achieved through an adaptable solution based on weather forecast, and a universal standard adopted by many states and cities for establishing regulatory speed zones.

With respect to tickets' issuing, if the system detects overspeed then a vehicle record representing a ticket is added to a governmental database dedicated for that purpose (see section 4.3.1 for ticket's attributes). Consequently, the governmental database server maps the vehicle's VIN to its plate number and determines the fine that has to be paid by the car owner (owner of plate number).

Having described the system at a high level scope, the following section illustrates the system's design and describes its implementation in details.

4 - Design and Implementation

4.1- System Architecture

Below is the system architecture of the conceptual model that exposes the model, the behavior, and the views of our proposed system in a sequence.

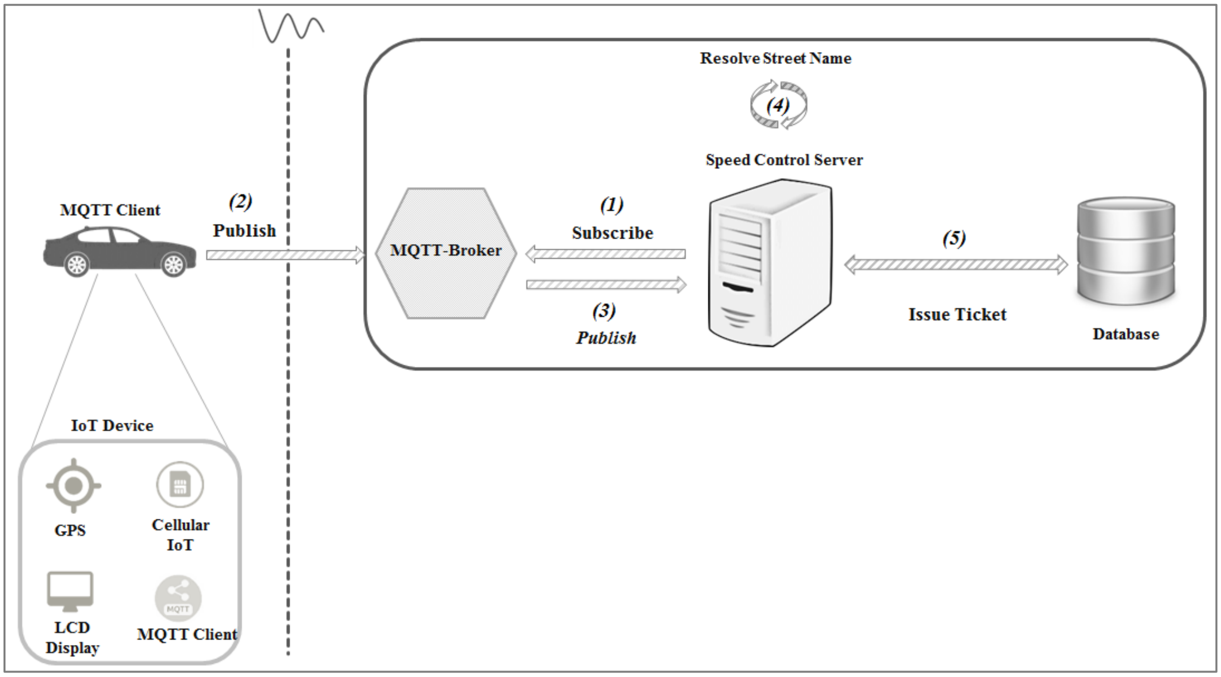


Figure 1. System Architecture

Figure 1 illustrates the consecutive steps of the different components of the system and how they interact with each other:

1. The speed control server which is considered as an MQTT client subscribes to the MQTT-Broker server to receive the messages published by all the moving vehicles.
2. The vehicle publishes its identity (VIN) as well as the speed, the GPS coordinates, and the time stamp.
3. Since the speed control server is already subscribed to the same published topic[2] (see Figure 2) of the vehicle; therefore, it receives the published information.

[2] A topic in MQTT technology represents the key that identifies the information channel to which payload data is published

4. The speed control server is constantly performing reverse geo-location to map the received coordinates against the street name/number and hence, it detects the speed limit.
5. In case of overspeed detection, a ticket is issued spontaneously and stored in a database.

4.2- Device Components

4.2.1 - Cellular IoT

Our IoT device uses the cellular 3G module through a Subscriber Identity Module (SIM) card to establish all kind of wireless communications from and to the server.

It is required to implement cellular IoT 3rd Generation Partnership Project (3GPP) technologies: Extended coverage Global System for Mobile communication (ECGSM), Long Term Evolution (LTE), Long Term Evolution Machine to Machine LTE-M, and the new radio access technology Narrowband IoT (NB-IoT) specifically tailored to form an attractive solution for emerging low power wide area (LPWA) applications [101].

4.2.2 - Global Positioning System (GPS):

The GPS navigation is a component that accurately calculates geographical location by receiving information from GPS satellites [102]. The GPS device is used to send to server the exact vehicle location (longitude and latitude).

4.2.3 MQTT Client:

MQTT client API is installed on the device to enable the "publish/subscribe" model to the MQTT broker located on the server side as shown in Figure 2. Each vehicle's published message includes a topic and its corresponding payload. The topic designates the routing information for the broker. Clients that subscribe to a specific topic receive the message pertaining to the topic's key.

4.3- Software Design

Figure 2 depicts the software design of the system. On the server side, Mosquitto™ which is an open source MQTT message broker is adopted as the system's broker. It is responsible to distribute the messages related to a topic to all its subscribers.

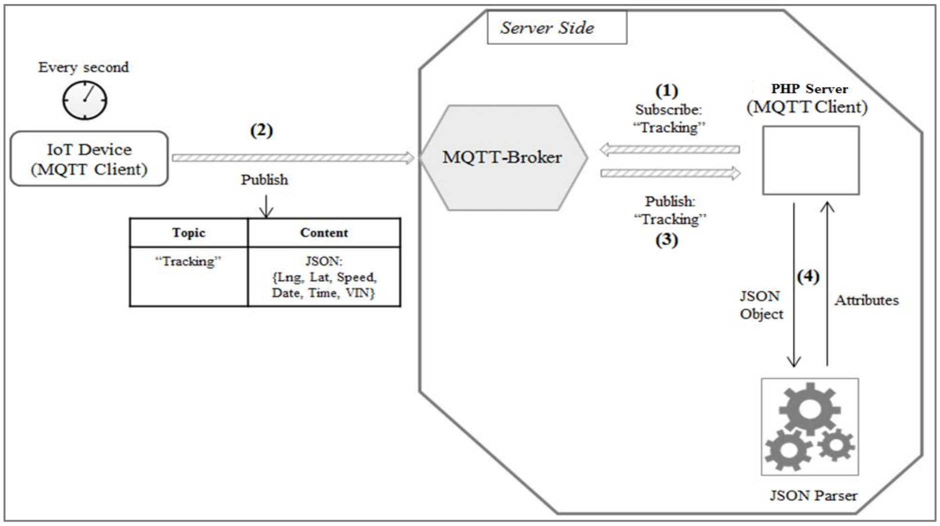


Figure 2. Software Design

In our software design, the vehicle equipped with the IoT device publishes to the topic "Tracking" a JSON object (the topic's payload) containing the following parameters:

a. Longitude: East-West geo coordinate of a point on the earth's surface.
b. Latitude: North-South geo coordinate of a point on the earth's surface.
c. Speed: Using the GPS's internal clock, the speed is calculated by measuring the time the vehicle needs to traverse between two points. As an alternative to GPS, the vehicle speed can also be obtained through the OBDII interface.
d. Date and Time: The National Marine Electronics Association (NMEA) data generated from the GPS is converted to readable format to extract the current date and time.
e. VIN: A unique combination of 17 letters and digits to identify a vehicle.

One of the subscribers to the MQTT broker is the speed control server which uses the Mosquitto-PHP library to become an MQTT client. Once the control server receives a message, it parses it retrieve every attributes separately.

Our software design is based on two alternatives: The first is the static model which relies on offline reverse geocoding for speed limit detection. The second is the adaptable model in which the speed limit parameter is detected dynamically as described in section 4.3.2. This method is based on the continuous weather forecast and the 85% percentile theorem.

4.3.1 - Static model

The static model is illustrated in Figure 3 as shown below:

1- The PHP server sends the longitude and latitude parameters to a java application through PHP-JAVA-Bridge.

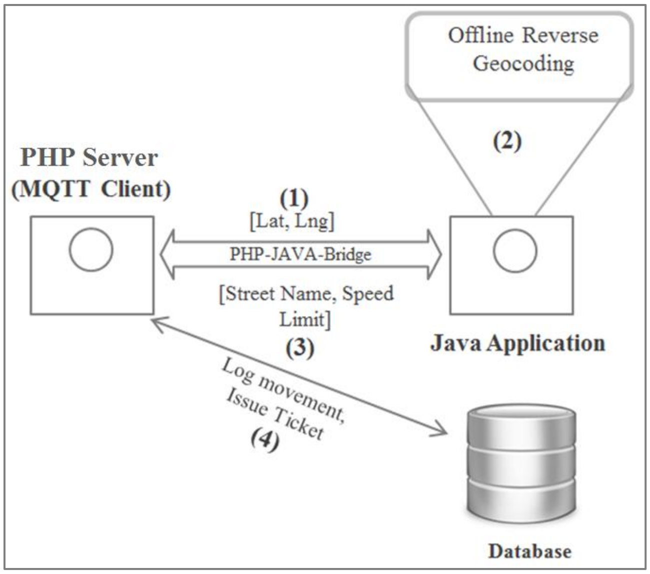


Figure 3. Static Pattern

2- KD-Tree [103] (Figure 4) for nearest neighbor lookup algorithm is implemented in Java to perform offline reverse geocoding. The Java application accepts as input the longitude and the latitude, and then compares them against a database provided by the government containing all country's street names and their corresponding speed limit.

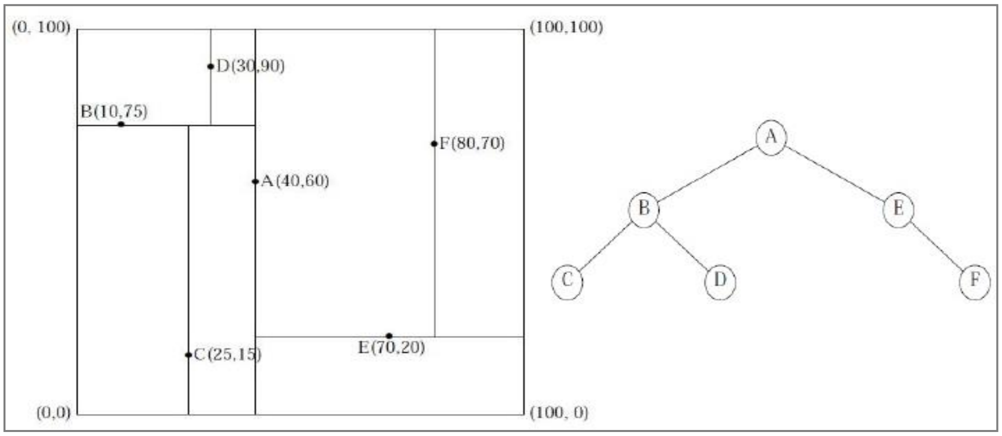


Figure 4. Spatial indexes – KD tree

3- The street name and the speed limit are returned to the PHP server for processing.

4- The PHP server extracts the speed from the JSON object and checks if it exceeds the speed limit. A ticket is issued and stored into the database in case of overspeed detection.

The ticket record's attributes are: VIN, speed limit, longitude and latitude, exceeded speed, date and time.

4.3.2 - Adaptable model

In the adaptable model, the speed limit is considered as variant parameter and is calculated according to two factors: (1) 85th percentile theorem of vehicles speed, (2) continuous weather forecast.

85^{th} percentile

The 85th percentile speed is the speed that 85 percent of vehicles do not exceed. Since the system is collecting all vehicles ‘velocities as well as their locations, then, the 85^{th} percentile theorem can be used to determine the new speed limit. Every ***T*** seconds/minutes (configured per country), the speed control server calculates the new speed limit per cluster and updates the database accordingly. A cluster as shown in Figure 6 is a segment defined by the country in a highway. The dynamic clustering system for highways is still an area under research.

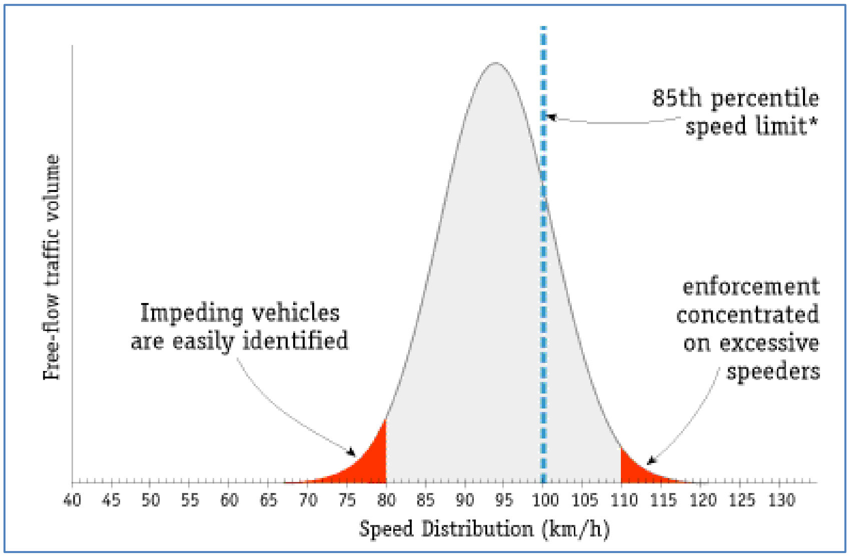


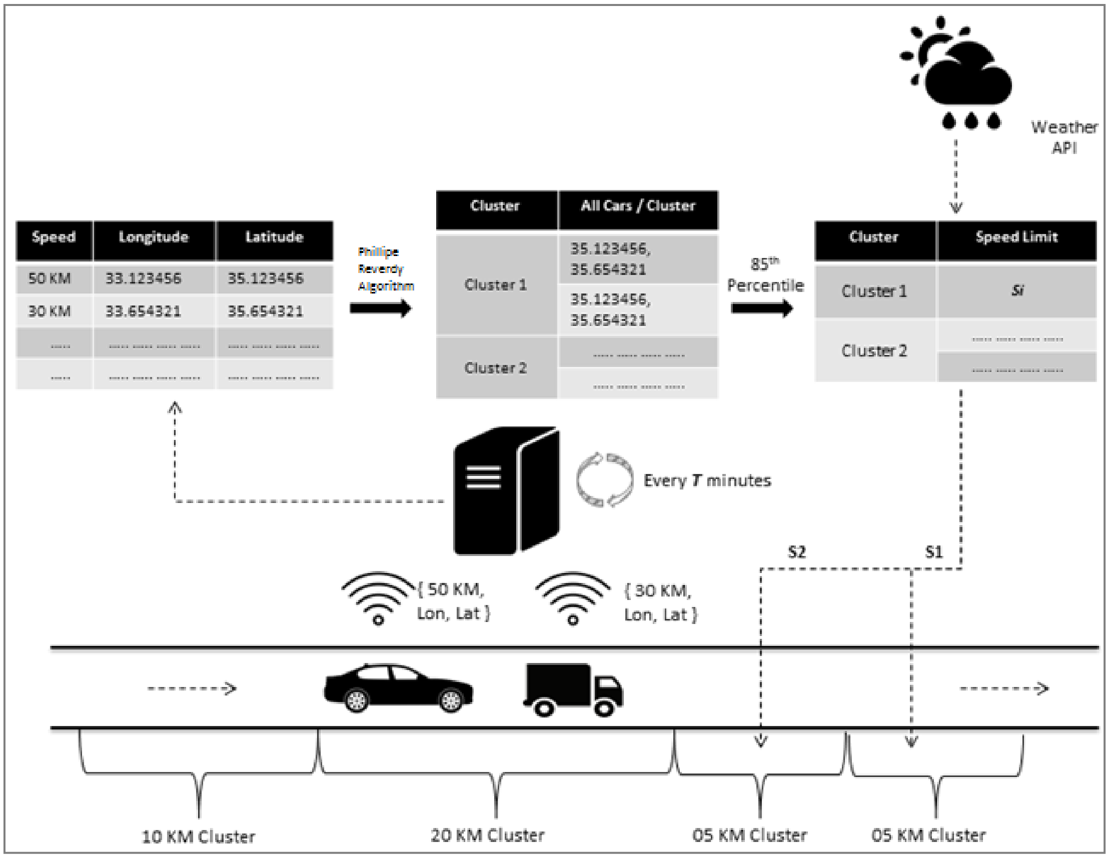
Figure 5. Adaptable Pattern (Source: http://michigandistilled.org)

The use of the 85th percentile speed concept is based on the theory that: 1) the large majority of drivers: Are reasonable and prudent, 2) do not want to have a crash, 3) desire to reach their destination in the shortest possible time.

A speed at or below which 85 percent of people drive at any given location under good weather and visibility conditions may be considered as the maximum safe speed for that location [104].

How to determine if a car lies within a specific cluster?

To determine whether a car lies within a specific road cluster an algorithm proposed by Philippe Reverdy is used. By considering a road cluster as a polygon, this algorithm computes the sum of the angles made between the test point and each pair of points making up the polygon. If this sum is $\mathbf{2\pi}$ then the point is an interior point (vehicle lies within the cluster), if it is **0** then the point is an exterior point (vehicle lies outside the cluster) [105].



Speed	Longitude	Latitude
50 KM	33.123456	35.123456
30 KM	33.654321	35.654321
.....	....	....
.....	....	....

Cluster	All Cars / Cluster
Cluster 1	35.123456, 35.654321
	35.123456, 35.654321
Cluster 2	....
	....

Cluster	Speed Limit
Cluster 1	*Si*
Cluster 2	....
	....

Figure 6. Adaptable Pattern

Weather conditions: Another variable used in the adaptable model is the current weather conditions. The vast majority of most weather-related crashes happen on wet pavement and during rainfall: 73% on wet pavement and 46% during rainfall [106]. In this system, we relied on "Highway Capacity Manual 2000" Chapter 22 as the source for changing the speed according to the weather. The following table summarizes the reduction rates:

Weather Conditions	*Freeway Traffic Flow Reductions*			
	Average Speed	**Free-Flow Speed**	**Volume**	**Capacity**
Light Rain/Snow	3% - 13%	2% - 13%	5% - 10%	4% - 11%
Heavy Rain	3% - 16%	6% - 17%	14%	10% - 30%
Heavy Snow	5% - 40%	5% - 64%	30% - 44%	12% - 27%

Table 1. Speed limit reduction rates according to weather conditions (Source: U.S. Department of Transportation: Road Weather Management Program)

As a matter of fact, the server contacts the Weather Underground API to acquire the weather forecast of the current cluster and consequently, it reduces the speeds accordingly.

5 - Results

In this section we focus on showing the results of only the adaptable model since the static model relies on fixed speed limits and does not depend on dynamic speed detection. To validate the results, we developed an Android mobile application to determine the vehicles' speed within a selected cluster of 06 Kilometres (Figure 7) between two Lebanese towns namely, Jounieh and Jbeil. This mobile application was distributed to 150 drivers using a link on a file server. We used the mobile application knowing that it is infeasible to manufacture 150 IoT devices considering this method is still under research.

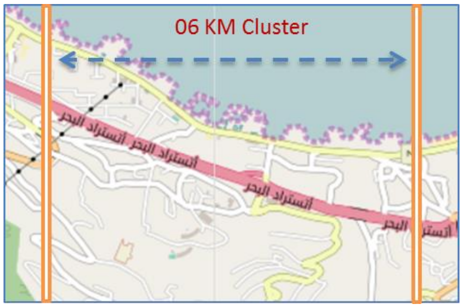


Figure 7. Jounieh – Jbeil Highway Cluster (Lebanon)

The following table exhibits the 150 average speeds calculated by the server. The mobile app publishes the speed to the server every one second; consequently, the server calculates the average speed per car every ***T*** seconds (preconfigured per country).

Speed in Km/h collected from 150 vehicles														
55	85	75	65	95	50	75	40	70	55	73	85	85	85	70
40	40	95	95	85	60	100	70	100	40	100	90	100	100	35
80	65	85	55	40	35	85	75	80	35	80	75	100	95	35
80	90	90	50	80	35	40	50	55	40	40	35	95	35	65
35	72	70	75	65	35	40	45	50	90	40	95	35	55	65
85	80	80	95	85	35	60	80	90	80	70	70	65	60	55
90	40	40	50	95	80	80	35	35	65	60	85	90	45	85
65	75	95	65	60	85	90	45	45	60	50	55	35	65	60
90	90	35	100	70	60	80	35	80	35	77	45	85	95	95
40	70	55	35	50	75	40	60	35	80	60	95	55	80	100

Table 2. Speed collected from the mobile applications distributed to 150 drivers.

For better graph visualization, we calculated the frequencies of the speeds by dividing them into class intervals of width of 05 Km/h each as shown below.

35-39	**40-44**	**45-49**	**50-54**	**55-59**	**60-64**	**65-69**	**70-74**	**75-79**	**80-84**	**85-89**	**90-94**	**95-100**
19	14	5	7	9	10	10	5	8	15	13	10	20

Table 3. Frequency table of speeds per class.

Applying the 85th percentile theorem stated in section 4.3.2 on the above frequency table, the resulting speed limit showed 90.0 Km/h.

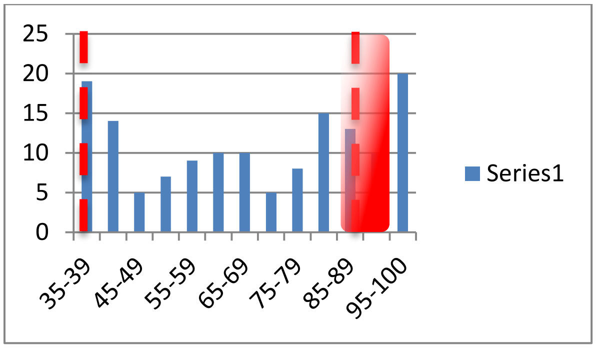


Figure 8. Jounieh – Jbeil Highway Cluster (Lebanon)

The server correlates the maximum speed limits to the current corresponding weather condition, applies the speed reduction rate as depicted in Table 1, and deduces the final maximum speed limit.

As a result, our IoT device considers 90 Km/h as the speed limit and an embedded customized panel displays the maximum allowed speed limit.

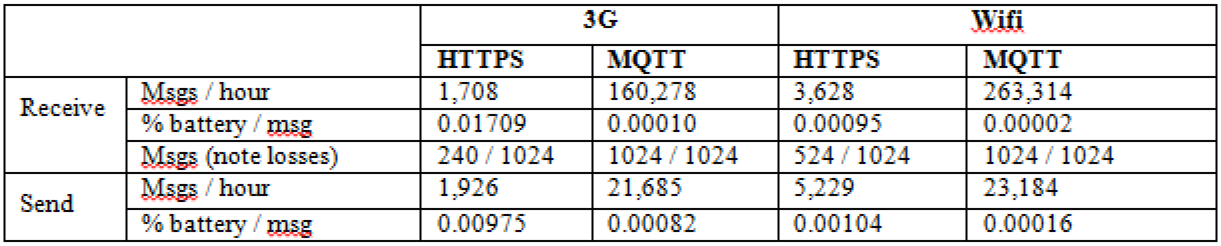
6 - Performance

This IoT device could have been implemented using the HTTP protocol instead of MQTT. But since HTTP is a stateless protocol in which each client request is treated as an independent transaction that is unrelated to any previous request, many additional technical requirements are then needed to maintain session awareness between the client and the server; therefore, more system resources and networks capabilities are required. That would lead to poor system performance, memory leaks, and power drain.

Unlike HTTP, the adopted MQTT is a bidirectional IoT protocol which maintains stateful session awareness. What differentiates MQTT is its lightweight overhead; it requires minimal bandwidth and less power consumption. Thus it helps minimize the resource requirements for the IoT device and ensures reliability and some degree of assurance of delivery with grades of service.

The table below shows a comparative test of a certain number of messages sent and received over HTTPS (HTTP secure) or MQTT and the battery power consumption per message in 3G and WiFi has been recorded.

This test had been conducted by using a smartphone running Android 2.2, and was simulated by sending 1024 messages, of one byte each, to and from the mobile device. Results showed tremendous advantages of the MQTT protocol over the HTTP [107].



		3G		Wifi	
		HTTPS	MQTT	HTTPS	MQTT
Receive	Msgs / hour	1,708	160,278	3,628	263,314
	% battery / msg	0.01709	0.00010	0.00095	0.00002
	Msgs (note losses)	240 / 1024	1024 / 1024	524 / 1024	1024 / 1024
Send	Msgs / hour	1,926	21,685	5,229	23,184
	% battery / msg	0.00975	0.00082	0.00104	0.00016

Table 4. The following table compares messages per hour, and battery usage per message, between HTTP and MQTT networks.

7 - Conclusions and Future Work

In this paper, we proposed and implemented an innovative IoT system which may help the community reducing the death rates resulting from high speed vehicles crashes. It is worth mentioning that the use of this IoT system is restricted for governmental use only due to some privacy concerns related to the drivers' identity and vehicles' tracking. Two models were successfully implemented to detect the speed limit and verified in real environment.

This autonomous system also assists governments in issuing car tickets, collecting fines, controlling the speed limit, and reporting road conditions without human interaction. The results showed that this system is robust and efficient due to the adoption of the MQTT lightweight IoT protocol. Results also showed that system helps the government in: (a) Practicing full control on traffic monitoring and enforce speed laws in all roads and not only highways. (b) Issuing real time tickets without any human interaction. (c) Dismissing traditional devices like RADAR, LIDAR, and SDCS, and thus saving remarkable amount of money. (d) Generating daily records related to the traffic state, to the number of issued tickets in all over the country or in a specific area. These records are stored instantaneously into a dedicated database and can be later fed into a data mining engine for future statistical analysis. (e) Producing daily monetary reports related to fine amount. (f) Customizing the overspeed threshold value conferring to each country's speeding policy and procedures. (g) Reducing the number of policemen in the road. (h) Decreasing the maintenance cost compared to traditional systems. (i) Adopting inexpensive solution to traffic control and traffic issuing. (j) Reducing the number of accidents and lessening the mortality and morbidity rate.

As future work, our intention is to update/upgrade/enhance this system to become an aggregated unit and to embed it into every vehicle during the manufacturing phase. Also, future dynamic road clustering allocation problem is one of our key factors to solve.

This will create a form of closed network, denying the attackers to reach our services. However, a side effect of this approach could be that new users might also be denied from the service until the attack is done. A proof of concept experiment was performed, DDoSing a webserver, proving that our algorithm is viable and effective. Finally, it is worth mentioning that there are no perfect mitigations for DDOS attacks till today. But this could be a solution for applications where the network switches from open to a closed network based on trust is desirable.

References

[1] http://www.gartner.com/newsroom/id/2905717

[2] https://www.qualcomm.com/products/smart-cities

[3] ISO-IEC JTC 1 N11712 CESI contribution on possible work on Smart Cities, ISO/IEC JTC 1, 2014

[4] Article 29 DPWP. (2014). *Opinion 8/2014 on the Recent Developments on the Internet of Things*. Article 29 Data Protection Working Party. Retrieved from http://ec.europa.eu/justice/data-protection/article-29/documentation/opinion-recommendation/files/2014/wp223_en.pdf

[5] Cerrudo, C.: An emerging US (and world) threat: Cities wide open to cyber-attacks.Tech. rep., Securing Smart Cities (2015)

[6] Bodenheim, Roland, Butts, Jonathan, Dunlap, Stephen, and Mullins, Barry. (2014). Evaluation of the Ability of the Shodan Search Engine to Identify Internet-facing Industrial Control Devices. *International Journal of Critical Infrastructure Protection*, 7, 114-123.

[7] Kitchami R., Dodge M., The (in)security of smart cities: vulnerabilities, risks, mitigation and prevention, Published as an open access pre-print on SocArXiv, Ireland, February 2017.

[8] Stephanie Weagle. Financial Impact of Mirai DDoS Attack on Dyn Revealed in New Data. 2017. URL: https: //www.corero.com/blog/797-financial-impact-of-miraiddos- attack-on-dyn-revealed-in-new-data.html.

[9] Albino, Vito, Umberto Berardi, and Rosa Maria Dangelico. "Smart Cities: Definitions, Dimensions, Performance, and Initiatives". Journal of Urban Technology 22.1 (2015): 3-21. Web.

[10] [Online]. Available: http://us.iscvt.org/wp-content/uploads/2017/01/Smart-Cities-RG.pdf. [Accessed: 04- Jun- 2017].

[11] *Kogan, Natalia, Kyoung Jun Lee, Yung Ho Suh and Jae Hong Park. "Thesis for the Degree of Master of Science Exploratory research on success factors and challenges of Smart City Projects." (2014).*

[12] Neirotti, P., De Marco, A., Cagliano, A.C., Mangano, G., Scorrano, F., Current trends in smart city initiatives: Some stylised facts. Cities, 2014

[13] Lazaroiu, G.C., Roscia, M.: De_nition methodology for the smart cities model.Energy ,2012

[14] Seto, Y., Application of privacy impact assessment in the smart city. Electronics and Communications in Japan, 2015

[15] Li, Y., Dai, W., Ming, Z., Qiu, M.: Privacy protection for preventing data over- collection in smart city. IEEE Transactions on Computers, 2015

[16] Matuszak, W.J., DiPippo, L., Sun, Y.L., CyberSAVe, Situational awareness visualization for cyber

security of smart grid systems. In: Proceedings of the 10th Workshop on Visualization for Cyber Security, 2013

[17] Wang, P., Ali, A., Kelly, W., Data security and threat modeling for smart city infrastructure. In: Proceedings of the 2015 International Conference on in Cyber Security of Smart Cities, Industrial Control System and Communications, 2015

[18] Alibasic, A., Al Junaibi, R., Zeyar Aung, Z., Woon. W, and Omar. M., Cybersecurity for Smart Cities: A Brief Review, Institute Center for Smart and Sustainable Systems (iSmart), Abu Dhabi, United Arab Emirates, 2016

[19] Chourabi H, Nam T, Walker S, Gil-Garcia JR, Mellouli S, Nahon K et al. Understanding smart cities: An integrative framework. In Proceedings of the Annual Hawaii International Conference on System Sciences. 2011. p. 2289-2297.

[20] MTSFB, “Report on Framework on Smart Cities and Standardization in Relations to Information and Communications Aspects”, MCMC, 2017

[21] Expanding Participation and Boosting Growth: The Infrastructure Needs of the Digital Economy, World Economic Forum Report, March 2015.

[22] Safegarding the Internet of Things. Deloitte review, issue 17, 2015

[23] Issues paper on “Smart cities and Infrastructure” United Nations Commissions on Science and Technology for Development, Hungary, 2016

[24] Here’s Why ‘The Internet Of Things’ Will Be Huge, And Drive Tremendous Value For People And Businesses, Business Insider, 2013; Emily Adler. H ttp://www.businessinsider.com/growth-in-the-internet-of-things-2013-10

[25] Getting Smart About Smart Cities, USDN Resource Guide, Nutter Consulting and Institute for Sustainable Communities Canada, 2014

[26] "iot.ieee.org/images/files/pdf/networks-of-things_jeff-voas_5-31-2016.pdf"

[27] Cavoukian, A. and Castro, D. (2014) *Big Data and Innovation, Setting the Record Straight: De-identification Does Work*. Information and Privacy Commissioner Ontario, Canada. www2.itif.org/2014-big-data-deidentification.pdf

[28] A report on “Cyber Security**:** A Necessary Pillar of Smart Cities”, India Security Conference, 2016, Retrieved from http://www.ey.com/Publication/vwLUAssets/ey-cyber-security/$FILE/ey-cyber-security.pdf

[29] Cerrudo, Cesar. (2014) Hacking US (and UK, Australia, France, etc.) Traffic Control Systems. *IOActive Blog*, 30 April. http://blog.ioactive.com/2014/04/hacking-us-and-uk-australia-france-etc.html

[30] Keoh, S. L., Kumr, S., and Tschofenig, H., “Securing the Internet of Things – A standardization

Perspective", Intenet of things Journal Vol. 1, No. 3, June 2014.

[31] A report on "Securing the Internet of things. Opportunity: Putting Cybersecurity at the Heart of the IoT", Capgemini Consulting, 2015

[32] Ijaz s. et al., "Smart Cities: A Survey on Security Concerns" *International Journal of Advanced Computer Science and Applications (IJACSA) ,Vol. 7, No. 2, 2016*

[33] https://www.owasp.org/images/7/71/Internet_of_Things_Top_Ten_2014-OWASP.pd

[34] S. Cirani *et al.*, "A scalable and self-con_guring architecture for service discovery in the Internet of Things," *IEEE Internet Things J.*, vol. 1, no. 5, pp. 508_521, Oct. 2014.

[35] Carnot Inst. (Jan. 2011). *Smart Networked Objects and Internet of Things, Carnot Institutes' Information Communication Technologies and Micro Nano Technologies Alliance, White Paper*. [Online]. Available: http://www.internet-of-things-research.eu/pdf/ IoT_Clusterbook_March_2010.pdf, accessed Nov. 28, 2011.

[36] Vijayan, J. (2014) With the Internet of Things, smart buildings pose big risk. *Computer World*, 13 May.www.computerworld.com/article/2489343/security0/with-the-internet-of-things--smart-buildings-pose-big-risk.html

[37] Energy Ensemble (2015), http://energyensemble.com/news_details.php?news_id=240

[38] http://saveonenergy.ca/Business/Program-Overviews/Retrofit-for-Commercial.aspx

[39] LÉVY-BENCHETON C.,Darra E., "Cyber security for Smart Cities: An architecture model for public transport", European Union Agency For Network And Information Security, 2015. www.enisa.europa.eu

[40] Sinopoli J., "Smart Building Systems for Architects, Owners and Builders", Elsevier, 2010

[41] Drobniak, A. , Exploring the Urban Resilience Concept, Presentation Delivered at Regional Studies Association Research Network on Transition and Resilience for Post- Industrial Agglomerations in Central Europe Seminar, Katowice, 30th January 2012.

[42] Akçura, M.T.; Avci, S.B. How to make global cities: Information communication technologies and macro-level variables. Technol. Forecast. Soc. Chang. **2014**, 89, 68–79.

[43] Future Crimes: Everything Is Connected, Everyone Is Vulnerable and What We Can Do About It, Marc Goodman

[44] Safeguarding the Internet of Things, Deloitte Review, Issue 17, 2015

[45] Lin, P., Swimmer, M., Urano, A., Hilt, S., Vosseler, R., "Securing Smart City – Moving toward utopia with Security in Mind ", A TrendLabs research paper, Trend Micro, 2017

[46] Williams, C., "The Register. "Passengers Ride Free on SF Muni Subway After Ransomware Infects Network, Demands $73K." Last accessed on 24 August 2017, http://www.theregister.co.uk/2016/11/27/san_francisco_muni_ransomware/.

[47] Ishimaru, H., ABC 7 News, “Computer Tested Again in BART Tech Problem.” Last accessed on 21 August 2017, http://abc7news.com/archive/9335794/.

[48] Department of Homeland Security. ICS-CERT. “Advisory (ICSA-16-231-01) . Locus Energy LGate C, December, 06, 2016.

[49] Sean O’Kane, S., The Verge. “Gogoro Starts an Electric Scooter-Sharing Program in Berlin.”, 2016 Last accessed on 18 August, 2017, http://www.theverge.com/2016/8/3/12358280/gogoro-electric-scooter-sharing-app-berlin-taiwan.

[50] GD and CSC. “Challenge of BLE Certification Mechanism Design: Take Gogoro Smart Scooter as an Example.” Last accessed on 18 August 2017, https://hitcon.org/2016/CMT/slide/day1-r0-a-1.pdf.

[51] Branden Ghena, B., et.al, “Green Lights Forever: Analyzing the Security of Traffic Infrastructure.” Last accessed on 22 Augustl 2017, https://jhalderm.com/pub/papers/traffic-woot14.pdf.

[52] Trend Micro. (26 October 2016). TrendLabs Security Intelligence Blog. “The IoT Ecosystem Is Broken. How Do We Fix It?”, October 2016, Last accessed on 15 August 2017, http://blog.trendmicro.com/trendlabs-security-intelligence/internet-things-ecosystem-broken-fix/.

[53] A. Zanella, N. Bui, A. Castellani, L. Vangelista and M. Zorzi, "Internet of Things for Smart Cities", *IEEE Internet of Things Journal*, vol. 1, no. 1, pp. 22-32, 2014.

[54] S. Deering and R. Hinden. “Internet Protocol, Version 6 (IPv6) Specification.” RFC2460. s.l. : IETF, 1998.

[55] *New.ipv6-taskforce.nl/cms/wp.../testing_the_security_of_IPv6_implementations.pdf*

[56] Analysis of NTP reflection attacks: http://conferences2.sigcomm.org/imc/2014/papers/p435.pdf

[57] Analysis of DNS and DNSSEC reflection attacks: http://conferences2.sigcomm.org/imc/2014/papers /p449.pdf

[58] Global Threat Intelligence Report, NTT Security, 2017

[59] Personal Communications, Rob van den Dam, IBM Institute for Business Value, ITU Telecom World, 2013

[60] MITRE Corporation, Common Vulnerability and Exposures, CVE List, http://cve.mitre.org /cve/cve.html.

[61] M. Lopez-Benitez and F. Casadevall, “Empirical time-dimension model of spectrum use based on a discrete-time markov chain with deterministic and stochastic duty cycle models,” IEEE Trans. Veh. Technol., vol. 60, no. 6, pp. 2519–2533, 2011.

[62] Common Vulnerabilities and Exposures (CVE),” http://cve.mitre.org, [Online; accessed May-2017].

[63] A. S. Elmaghraby and M. M. Losavio, “Cyber security challenges in Smart Cities: Safety, security and privacy,” J. Adv. Res., vol. 5, no. 4, pp. 491–497, Jul. 2014.

[64] "National Vulnerabilities Database (NVD)," http://nvd.nist.gov, [Online; accessed May-2017].

[65] N. Pokhrel and C. Tsokos, "Cybersecurity: A Stochastic Predictive Model to Determine Overall Network Security Risk Using Markovian Process", Journal of Information Security, vol. 08, no. 02, pp. 91-105, 2017

[66] C. Phillips and L. Swiler. A graph-based system for network-vulnerability analysis. In Proceedings of the New Security Paradigms Workshop (NSPW'98), 1998.

[67] D. Balzarotti, M. Monga, and S. Sicari. Assessing the risk of using vulnerable components. In Proceedings of the 1st ACM QoP, 2005.

[68] W. Li and R. B. Vaughn. Cluster security research involving the model¬ing of network exploitations using exploitation graphs. In Proceedings of the Sixth IEEE International Symposium on Cluster Computing and the Grid, CCGRID '06, pages 26–, Washington, DC, USA, 2006. IEEE Computer Society.

[69] V. Mehta, C. Bartzis, H. Zhu, E. Clarke, and J. Wing. Ranking attack graphs. In Recent Advances in Intrusion Detection 2006, 2006.

[70] Li Q.M. and Zhang. H. Information security risk assessment technology of cyberspace: a review. International Journal on Information, 15 (11):4677–4683, 2012.

[71] The MITRE Corporation. Common weakness scoring system. http://cwe.mitre.org/cwss/, 2010.

[72] Nagarajan V., et al., "Using Power Hoping to Counter MAC Spoofing Attacks in WLAN", in Proc. Of the 7th IEEE Consumer Communications and Networking Conf. (CCNC '10), Las Vegas, USA, pp 1-5, 2010.

[73] Nikbakhsh S., et al., "A Novel Approach for Rogue Access Point Detection on the Client-Side", in Proc. Of the 26th International Conf. on Advanced Information Networking and Applications Workshops (WAINA), Fukuoka, Japan, pp 684-687, 2012.

[74] The Smart Grid Interoperability Panel – Cyber Security Working Group, "Smart grid cyber security strategy and requirements," NIST IR- 7628,Feb. 2010.

[75] S. M. Ross, Stochastic Processes, Second Edition, Wiley, 1996.

[76] A. Mpitziopoulos, D. Gavalas, C. Konstantopoulos and G. Pantziou, "A survey on jamming attacks and countermeasures in WSNs", IEEE Communications Surveys & Tutorials, vol. 11, no. 4, pp. 42-56, 2009.

[77] B. Krebs, "KrebsOnSecurity hit with record DdoS," in KrebsonSecurity, 2016. Source: https://krebsonsecurity.com/2016/09/krebsonsecurity-hit-with-record-ddos/.

[78] B. Krebs, "The Democratization of Censorship," in KrebsonSecurity, 2016. Source: https://krebsonsecurity.com/2016/09/the-democratization-of-censorship/.

[79] R. Millman, "OVH suffers 1.1Tbps DdoS attack," in News, SC Magazine UK, 2016. Source:

http://www.scmagazineuk.com/ovh-suffers-11tbps-ddos-attack/article/524826/.

[80] S. Hilton, "Dyn analysis summary of Friday October 21 Attack," in Dyn, 2016. Source: http://hub.dyn.com/dyn-blog/dyn-analysis-summary-of-Friday-october-21-attack.

[81] Anna-senpai, Mirai Source Code on GitHub, September 2016, Source: https://github.com /jgamblin/ Mirai-Source-Code.

[82] B. Krebs, "Who is Anna-Senpai, the Mirai Worm Author?", January 2017, Source: https://krebsonsecurity.com /2017/01/who-is-anna-senpai-the-mirai-worm-author/.

[83] T. Spring, K. Carpenter, and M. Mimoso, "BASHLITE family of Malware Infects 1 Million IoT devices," in Threat Post, Threatpost, 2016. Source:https://threatpost.com/ bashlite-family-of-malware-infects-1-million-iot-devices/120230/.

[84] B. Krebs, "Source code for IoT Botnet 'Mirai' released," in KrebsonSecurity, 2016. Source: https://krebsonsecurity.com/2016/10/source-code-for-iot-botnet-mirai-released/.

[85] B. Krebs, "Europe to push new security rules amid IoT mess," in *KrebsonSecurity*, 2016. [Online]. Available: https://krebsonsecurity.com/2016/10/131ikipe-to-push-new-security-rules-amid-iot-mess/.

[86] Surviving Distributed Denial of Service Attacks, Liu, S, IEEE Computer, vol. 11, no. 5, pages: 51-53, 2009.

[87] Defeating Distributed Denial of Service Attacks, Geng and Whinston, IT Professional, vol. 2, no 4, pages: 36 – 42, 2006.

[88] Defending Against Flooding Based Distributed Denial of Service Attacks: A Tutorial, Communications Magazine, IEEE, vol. 40, no. 10, pages 42-51, 2002.

[89] https://en.wikipedia.org/wiki/Heuristic_ (computer_science)

[90] 2017. [Online]. Available: https://www.researchgate.net/profile/Natallia_Kokash2. [Accessed: 21- Sep- 2017].

[91] Sustaining availability of Web Services under Distributed Denial of Service Attacks, Wooyong Lee, Jun Xu, Computers IEEE Transactions on, vol. 52, no. 2, pages: 195-208, 2003.

[92] Access Control Lists to Protect a Network from DoS Attacks, Dennis Eck, 2003.

[93] https://security.radware.com/ddos-knowledge-center/ddospedia/loic-low-orbit-ion-cannon/

[94] Who.int « Road safety - Speed ». [online] Available at: http://www.who.int/violence_injury_prevention/publications/road_traffic/world_report/speed_en.pdf

[95] ASIRT. « Annual Global Road Crash Statistics », [online] Available at: https://asirt.org/initiatives/informing-road-users/road-safety-facts/road-crash-statistics

[96] Bole & Wall & Norris (2013) *« Radar and ARPA Manual, 3rd Edition »*.

[97] Weitkamp (2005) « Lidar: Range-Resolved Optical Remote Sensing of the Atmosphere »

[98] Laser Jammers (2012). *How Laser Jammers Work*. [online] Available at: http://www.laserjammer.net/2012/how-laser-jammers-work/

[99] Bhatkar, Shivalkar, Tandale, Joshi. *« Survey of Various Methods used for Speed Calculation of a Vehicle »*

[100] Peter Friess (2013) *« Internet of Things: Converging Technologies for Smart Environments and Integrated Ecosystems »*.

[101] Ericsson (2016) *« Cellular networks for massive IoT »* [online] Available at: http://www.ericsson.com/res/docs/whitepapers/wp_iot.pdf

[102] Alfred Leick (1995) *« GPS Satellite Surveying 2nd »*

[103] Hasan Al-Jabbouli (2011) *« Data clustering using the Bees Algorithm and the Kd-Tree structure»*

[104] Texas Department *« Procedures for Establishing Speed Zones Manual »* [online] Available at: http://onlinemanuals.txdot.gov/txdotmanuals/szn/szn.pdf

[105] Paul Bourke (1987) *« Determining if a point lies on the interior of a polygon »* [online] Available at: http://masters.donntu.org/2009/fvti/hodus/library/article2/article2.html

[106] U.S. Department of Transportation *« Road Weather Management Program »*

[107] IBM. MQTT: Enabling the Internet of Things. [online] Available at: https://www.ibm.com/developerworks/community/blogs/c565c720-fe84-4f63-873f-607d87787327/entry/tc_overview?lang=en