ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ

ΔΙΑΤΜΗΜΑΤΙΚΟ ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ

«ΠΛΗΡΟΦΟΡΙΚΗ ΚΑΙ ΥΠΟΛΟΓΙΣΤΙΚΗ ΒΙΟΙΑΤΡΙΚΗ»

Vulnerability Assessment in Internet of Things Devices

GEORGE SKEPARNAKOS

MASTER'S THESIS

SUPERVISOR

Dr. STAMOULIS GEORGIOS

Lamia,2016

ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ

ΔΙΑΤΜΗΜΑΤΙΚΟ ΜΕΤΑΠΤΥΧΙΑΚΟ ΠΡΟΓΡΑΜΜΑ ΠΛΗΡΟΦΟΡΙΚΗ ΚΑΙ ΥΠΟΛΟΓΙΣΤΙΚΗ ΒΙΟΙΑΤΡΙΚΗ

ΚΑΤΕΎΘΥΝΣΗ .....

«ΠΛΗΡΟΦΟΡΙΚΗ ΜΕ ΕΦΑΡΜΟΓΕΣ ΣΤΗΝ ΑΣΦΑΛΕΙΑ, ΔΙΑΧΕΙΡΙΣΗ ΜΕΓΑΛΟΥ ΟΓΚΟΥ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΠΡΟΣΟΜΟΙΩΣΗ»

ΕΚΤΙΜΗΣΗ ΕΥΠΑΘΕΙΩΝ ΣΕ ΣΥΣΚΕΥΕΣ INTERNET OF THINGS

Σκεπαρνάκος Γεώργιος

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Επιβλέπων

Δρ.Σταμούλης Γεώργιος

ΕΚΤΙΜΗΣΗ ΕΥΠΑΘΕΙΩΝ ΣΕ ΣΥΣΚΕΥΕΣ INTERNET OF THINGS

Σκεπαρνάκος Γεώργιος

Τριμελής Επιτροπή:

Δρ. Σταμούλης Γεώργιος

Δρ. Κίκιρας Παναγιώτης

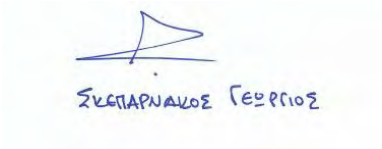
Δρ. Φιλιππόπουλος Ιωάννης

Λαμία 20/10/2016

«Υπεύθυνη Δήλωση μη λογοκλοπής και ανάληψης προσωπικής ευθύνης»

Με πλήρη επίγνωση των συνεπειών του νόμου περί πνευματικών δικαιωμάτων, και γνωρίζοντας τις συνέπειες της λογοκλοπής, δηλώνω υπεύθυνα και ενυπογράφως ότι η παρούσα εργασία με τίτλο [«τίτλος εργασίας»] αποτελεί προϊόν αυστηρά προσωπικής εργασίας και όλες οι πηγές από τις οποίες χρησιμοποίησα δεδομένα, ιδέες, φράσεις, προτάσεις ή λέξεις, είτε επακριβώς (όπως υπάρχουν στο πρωτότυπο ή μεταφρασμένες) είτε με παράφραση, έχουν δηλωθεί κατάλληλα και ευδιάκριτα στο κείμενο με την κατάλληλη παραπομπή και η σχετική αναφορά περιλαμβάνεται στο τμήμα των βιβλιογραφικών αναφορών με πλήρη περιγραφή. Αναλαμβάνω πλήρως, ατομικά και προσωπικά, όλες τις νομικές και διοικητικές συνέπειες που δύναται να προκύψουν στην περίπτωση κατά την οποία αποδειχθεί, διαχρονικά, ότι η εργασία αυτή ή τμήμα της δεν μου ανήκει διότι είναι προϊόν λογοκλοπής.

Ο ΔΗΛΩΝ



20/10/2016

I would like to thank professor dr.Panayotis Kikiras who accepted to give advices and the guidance to make this master thesis. Also through his lectures put the bases and the ideas to make this master thesis.

ABSTRACT

The adequacy of efficient wireless protocols, improved sensors, cheaper processors, and a bevy of startups and established companies developing the necessary management and application software has finally made the concept of the Internet of Things (IoT) mainstream. The connected devices on internet have surpassed the population of human beings in Earth in 2011. By 2020 these devices are expected to number between 25 billion and 50 billion. These will include all the types of consumer electronics, machine tools, industrial equipment, cars, appliances, and a number of devices likely not yet invented.

For the new technological era that we are passing, the cyber security for these devices must grow rapidly, with the same speed rate as the devices grow. Otherwise a lot of issues with serious impact will occur.In this master thesis' following chapters the cyber security of Internet of Things devices will be analyzed.

In the first chapters we will analyze the internet of things at every aspect. Then we will continue to mention the connection models of IoT.At the next chapters we will mention IoT attack surface, top IoT vulnerabilities and countermeasures and methodologies of penetration testing.

The following chapters will include and analyze the necessity of cyber security for IoT devices, IoT vulnerabilities and a research use case where the findings of vulnerable IoT devices will be represented. Security challenges, such as confidentiality and privacy issues will be analyzed and almost every aspect of the cyber security issues surrounding the internet of things device.

At the last chapter will be made a use case research for vulnerable devices which will be represented geographically and a proof which will show that full access was gained to them.

ΠΕΡΙΛΗΨΗ

Καθώς η ανθρωπότητα τεχνολογικά εξελίσσεται με γοργούς ρυθμούς, ταυτόχρονα συμβαδίζει με ακόμη γρηγορότερους ρυθμούς η νέα τεχνολογική εποχή που ονομάζεται η εποχή των Internet of Things ή αλλιώς, το ίντερνετ των πραγμάτων. Στις επόμενες σελίδες γίνεται η εισαγωγή αυτού που τις διαβάζει στην νέα αυτήν τεχνολογική εποχή, αναλύοντας έννοιες και απαντώντας σε ερωτήματα όπως τι είναι το ίντερνετ των πραγμάτων , που θα είναι χρήσιμο για την ανθρωπότητα , γιατί είναι αναγκαία η μετάβαση μας σε αυτήν την νέα εποχή, πως θα ωφεληθούν οι επιχειρήσεις από αυτό και άλλα αρκετά ενδιαφέροντα ερωτήματα.

Η εργασία όμως αυτή έχει ως σκοπό όμως να αναλύσει και να εξηγήσει στο μέγιστο δυνατόν τα προβλήματα που θα δημιουργηθούν ταυτόχρονα με αυτήν την ραγδαία ανάπτυξη του ίντερνετ των πραγμάτων όσον αφορά στην ιδιωτικότητα και στην κυβερνοασφάλεια. Θα εξηγηθούν οι ευπάθειες αυτώ των συσκευών αναλυτικά, τι μπορούν να κάνουν οι ειδικοί ασφαλείας για να τις εξαλείψουν όσο γίνεται, τις επιπτώσεις που θα έχουν αυτές οι ευπάθειες και σχεδόν κάθε τι που μπορεί να προκύψει και αναφέρεται στην ασφάλεια του ίντερνετ των πραγμάτων.

Στο τελευταίο μέρος της εργασίας γίνεται με ειδικές μηχανές αναζήτησης και λειτουργικά συστήματα κυβερνοασφάλειας γεωγραφική εύρεση ευπαθών τέτοιων συσκευών και συστημάτων. Επίσης σε κάποιες από αυτές δίνεται μέσα από στιγμιότυπα οθόνης η απόδειξη ότι καταφέραμε να αποκτήσουμε μη εξουσιοδοτημένη είσοδο σε αυτά πραγματοποιώντας αλλαγή σε ρυθμίσεις και έχοντας πλήρη έλεγχο των συσκευών, είτε βρίσκονταν σε οικιακό περιβάλλον είτε σε δημόσιους χώρους είτε ακόμη και σε βιομηχανικό περιβάλλον. Για λόγους ιδιωτικότητας και ασφάλειας των προσωπικών δεδομένων των χρηστών έχει γίνει απόκρυψη ορισμένων στοιχείων.

KEYWORDS:

IOT, VULNERABILITY ASSESSMENT, CYBER SECURITY ,

Πίνακας περιεχομένων

CHAPTER 1

1.1 INTERNET OF THINGS OVERVIEW

The internet has revolutionized the computers and the communications the last decades. Since 1960s with the first launch of ARPANET with many few users, the internet nowadays is used by more of the 40 % of the entire population of the earth.

With so many devices connected and people connected to the Internet, what will be the future? The future is called “The Internet of Things”. And what is the Internet of Things? This is the first question we need to answer.

The term Internet of Things refers to the use of standard Internet protocols for the human-to-thing or thing-to-thing communication in embedded networks. Sometimes the Internet of Things referred to as ubiquitous networking and computing .Another simple approach of the term Internet of Things is the network of physical objects—devices, vehicles, buildings and other items embedded with electronics, software, sensors, and network connectivity that enables these objects to collect and exchange data

The original concept and more scientific term was proposed by Kevin Ashton at the Auto-ID Center at MIT in 1999. The Internet of Things is an informational network that allows the look-up of information about real-world objects by means of a unique ID called Electronic Product Code (EPC) and a resolution mechanism (ONS), to a network of sensors, actuators and autonomous objects interacting with each other directly.

Despite the variety of definitions of the Internet of Things, the concept is similar. All of the definitions describe scenarios in which network connectivity and computing capability extends to a constellation of objects, devices, sensors and everyday items that are not ordinarily considered to be computers. This allows the devices to generate, exchange and consume data, often with minimal human intervention.

For the companies the current potential market for the Internet of Things is huge but not so visible in our everyday lives. It is considered from a survey from Cisco that in 2020 there will be around 50 billion devices connected to the Internet and all these will be potential IoT devices and sensors. So in every second of the next 4-5 years 57.000 devices will be connected to the Internet. In financial terms, the market is measured in Trillions of dollars with estimates at 9 Trillion dollars by 2020. In the short-term, growth to 2018, current projections are for a 300% growth in profits from incremental profits due to IoT. Furthermore, this is being projected and reported across all the major vertical markets such as Banking, Retail, Health, Transport, Manufacturing, Utilities and Government. Therefore, in both the short and long term, financial growth projections are staggering. Currently, Asia leads the way with 40% of Machine-to-Machine connections (M2M). This isn’t surprising, because China has already committed $603 Billion towards machine-to machine connections leaving the USA and EU well behind. Recently, in March 2015 the UK launched a major government sponsored initiative to encourage IoT research,

development and implementation. The goal of the UK IoT project is to encourage business to adopt this new revolutionary technology.

Below are some Internet of Things real world sector applications:

- Smart Cities – Traffic Management, Waste Management, Structural Health, noise urban maps, intelligent transportation systems
- Smart Environment – Earthquake early detection, Forest fire detection, Air quality and Pollution detection, Avalanche and landside prevention
- Smart Water – Water quality, leakage prevention, reservoir level management, river floods detection and prevention
- Security and Emergencies – Perimeter access controls, Radiation and liquid Detection, Explosive and Hazardous gases detection, emergency service management
- Smart Retail – Supply chain control, NFC payments, smart product management, Vending machine remote management
- Smart Logistics – Quality of shipment condition, item tracking, fleet tracking, geo-positioning, shipment/deliver management
- Industrial Control – M2M application, Environment control (HVAC), temperature control, ozone presence, vehicle auto-diagnosis, Warehouse stock location
- Smart agriculture – wine quality monitoring, crop irrigation, green house control, park management
- Smart Animal Farming – offspring care, animal tracking, environment monitoring, toxic gas levels, animal health care monitoring, food history management
- Smart Homes – temperature and humidity control, remote automation, lightning and ambiance control, energy efficiency, intrusion detection systems, fire and safety alarms
- eHealth – fall detection, sports monitoring, patient surveillance, equipment monitoring, health and fitness monitors, ultra-violet detection monitors.

These are just some of the more obvious IoT application already in common use today and they are evolving rapidly as the only bounds are the limits of innovation and creativity.

It is not just industry and consumer vertical markets that stand to be revolutionized, one other major business sector is already repositioning itself to reap the benefits, as it understands very well the concept of risk versus reward, and that is marketing.
Analyze the picture and write the internet of things market

Internet of things as it is mentioned above will grow and will be a part to a lot of critical aspects of everyday life and critical infrastructures.

Banking and finance sectors will use insurance based monitoring and billing, smart payments and smart loan applications and processing.

At public services the defense and the homeland security of a country will use internet of thins device.

Manufacturing sector can use internet of things devices for efficiency monitoring, failure analysis, proactive maintenance, supply chain optimization and security,

robotics, RFID logistics and connected devices, industrial control systems and video monitoring.

In Smart cities internet of things will increase applications for smart parking, environmental monitoring, smart lighting and watering, traffic management, police command and control and security monitoring.

At retail service internet of things devices will be used for automated checkout, sensors on shelves, smart fitting rooms and smart mirrors, proximity advertising, smart vending machines, security alarm and environmental sensors.

At energy infrastructures internet of things will bring the development of smart grid, demand on response, safety monitoring and fault detection, industrial control systems and security monitoring.

With all the above applications is presumed that internet of things will be the “heart” of a lot of critical infrastructures and either we see them or not, behind in every aspect in our everyday life in the near future.

The web so far has gone through four evolutionary stages. The first stage was the research stage when the web was called the Advanced Research Projects Agency Network (ARPANET) and was used for academic and research purposes.
The second stage of the evolution of the web was called the “gold rush”. At this phase of evolution almost every company focused on the needs to share information on the Internet so that people had the chance learn about their products and services. The third evolution stage transformed the web from static data to transactional information, where products and services could be sold and bought and services delivered. This phase was called the “dot-com” boom and bust.

The fourth phase of evolution where we are now is the “social” or “experience” web where big companies popular and gain profit by allowing people to communicate each other connect and share information. Together with the “social” and “experience” web the internet of things can attach and make the user’s experience more efficient by adding machine to machine communication and human to machine communication among with the human to human communication. Internet of thins become critical for human progression because people desire to live healthy fulfilling, and comfortable lives for themselves, their families, and those they care about. By combining the ability of the next evolution of the Internet (IoT) to sense, collect, transmit, analyze, and distribute data on a massive scale with the way people process information, humanity will have the knowledge and wisdom it needs not only to survive, but to thrive in the coming months, years, decades, and centuries.

All these makes the current potential market for Internet of Things huge. The confluence of efficient wireless protocols, improved sensors, cheaper processors, and a number of startups and established companies developing the necessary management and application software has finally made the concept of the Internet of Things mainstream. A number of companies and research organizations have offered a wide range of projections about the potential impact of IoT on the Internet and the economy during the next five to ten years. Cisco, for example, projects more than 24 billion Internet–connected objects by 2019, Morgan Stanley,

however, projects 75 billion networked devices by 2020. Looking out further and raising the stakes higher, Huawei forecasts 100 billion IoT connections by 2025.McKinsey Global Institute suggests that the financial impact of IoT on the global economy may be as much as $3.9 to $11.1 trillion by 2025. However, with all these new connecter devices, cyber security for Internet of Things becomes an essential aspect for every vendor, organization and consumer who use them.More attack vectors and more possibilities for harmful attacks will target this devices and their users. So it is critical to move fast to address this rising security concern in order not to face inevitable disaster.

At the summer of 2016 Vodafone published its global survey of business sentiment regarding innovation and investment in the Internet of Things. The survey was conducted by Circle Research in April and May 2016 and involved more than 1,096 companies around the world. The results were:

- 89% of companies investing in IoT have increased their budgets over the last 12 months
- 76% of all companies interviewed believe that taking advantage of IoT technologies will be critical for the future success of any organisation
- 63% of IoT adopters are seeing “significant” returns on investment, up from 59% in last year’s Repor
- IoT investment now accounts for 24% of the average IT budget, on a par with cloud computing or data analytics.
- 48% of companies interviewed are using IoT technologies to support large-scale business transformation, rising to 61% in the Asia-Pacific region
- 52% of consumer electronics companies interviewed are using IoT technologies as the basis for a new generation of applications for connected homes
- 46% of all companies interviewed said they intend to develop new IoT-based products and services over the next two years.

For improving the security issues of the IoT the companies answered:

- 42% will train their staff
- 41% will recruit security specialists
- 38% will establish a clear contingency plan
- 45% will establish clear security best practice and guidelines for staff
- 42% will make security a major part of request for proposal requirements
- 40% will work with a specialist security provider

Reviewing the results concerning the improve of security for IoT vendors it is disappointing that even the half of them takes the security concerns serious, even the investments rates are rapidly growing up.

1.2 INTERNET OF THINGS COMMUNICATION MODELS

1.2.1 DEVICE TO DEVICE COMMUNICATIONS

The device to device communication model represents two or more devices that directly connect and communicate between one another, through an

intermediary application server. These devices communicate with many types of networks, including IP networks or the Internet. Often, however these devices use protocols like Bluetooth, Z-Wave, or ZigBee to establish direct device-to-device communications.

These devices to device networks allow devices to be attached to a particular communication protocol to communicate and exchange messages to achieve functionality. This communication model is commonly used in applications like home automation systems, which typically use small data packets of information to communicate between devices with relatively low data rate requirements. Residential IoT devices like light bulbs, light switches, thermostats, and door locks normally send small amounts of information to each other (e.g. a door lock status message or turn on light command) in a home automation scenario. This device to device communication approach illustrates many of the functionality challenges that the IOT manufactures face. From the user's point of view, this often means that underlying device-to-device communication protocols are not compatible, forcing the user to select a family of devices that employ a common protocol. For example, the family of devices using the Z-Wave protocol is not natively compatible with the ZigBee family of devices. While these incompatibilities limit user choice to devices within a particular protocol family, the user benefits from knowing that products within a particular family tend to communicate well.

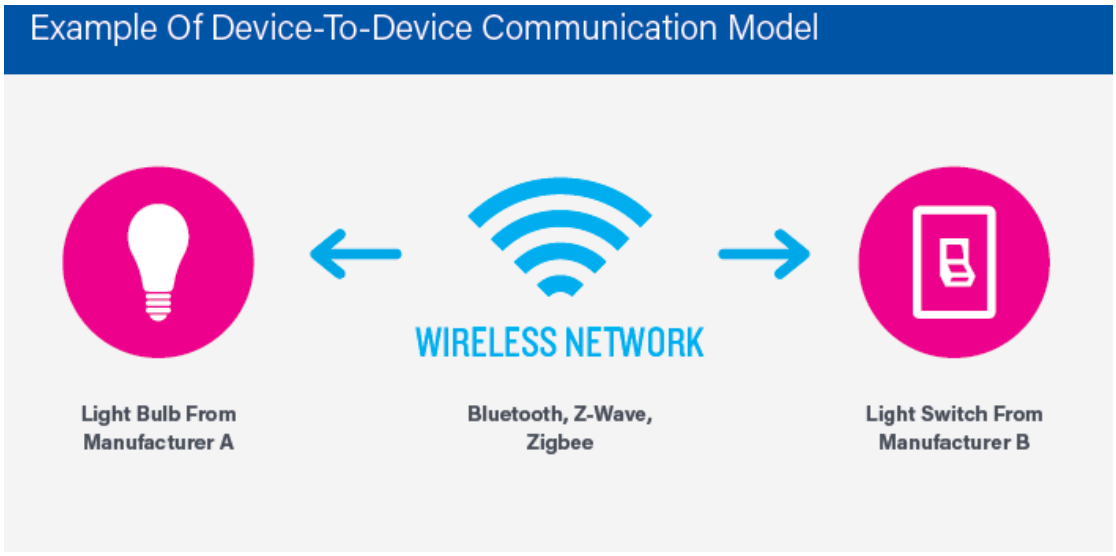


Image 1.Device to Device Communication Model

1.2.2 DEVICE TO CLOUD COMMUNICATIONS

In a device to cloud communication model, the IoT device connects directly to an Internet cloud service like an application service provider to exchange data and control message traffic. This approach frequently takes advantage of existing communications mechanisms like traditional wired Ethernet or Wi-Fi connections to establish a connection between the device and the IP network, which ultimately connects to the cloud service. This communication model is employed by some popular consumer IoT devices like smaet *Thermostats* and *Smart televisions*. In the case of the *Thermostats*, the device transmits data to a cloud database where the

data can be used to analyze home energy consumption. Further, this cloud connection enables the user to obtain remote access to their thermostat via a smartphone or Web interface, and it also supports software updates to the thermostat. The same process exists in the *Smart televisions* technology, the television uses an Internet connection to transmit user viewing information to the television vendor for analysis and to enable the interactive features of the TV. In these cases, the device to cloud model adds value to the end user by extending the capabilities of the device beyond its native features. However, functionality challenges can arise when attempting to integrate devices made by different manufacturers. Frequently, the device and cloud service are from the same vendor.

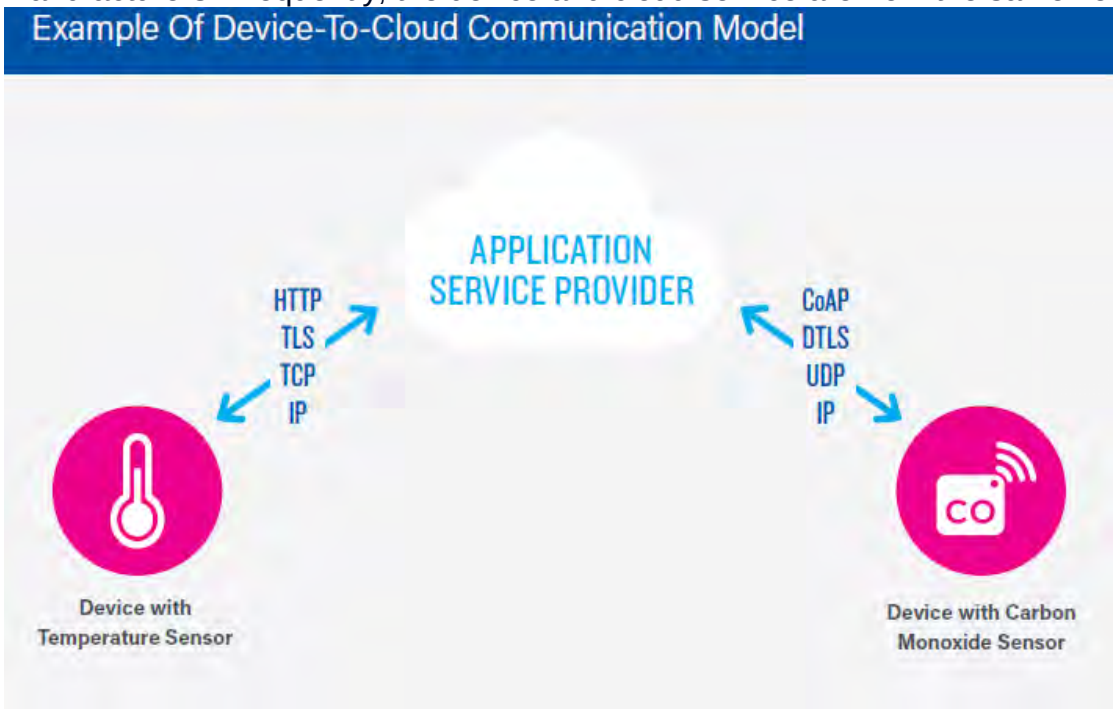


Image 2.Device to Cloud Communication Model

1.2.3 DEVICE TO GATEWAY MODEL

In the device to gateway model or the device to application layer gateway (ALG) model, the IoT device connects through an ALG service as a channel to reach a cloud service. This means that there is an application software operating on a local gateway device, which acts as a proxy between the device and the cloud service and provides security and other functionality such as data or protocol translation. Several forms of this model are found in consumer devices. In many cases, the local gateway device is a smartphone running an app to communicate with a device and relay data to a cloud service. This is often the model which is employed in popular consumer items like personal fitness trackers. These devices do not have the native ability to connect directly to a cloud service, so they frequently rely on smartphone application software to serve as a middleman gateway to connect the fitness device to the cloud.

The other form of this device to gateway model is the emergence of “hub” devices in home automation applications. These are devices that serve as a local gateway between individual IoT devices and a cloud service, but they can also bridge the communication functionality gap between devices themselves. This communication model is used in situations where the smart objects require

interoperability with non-IP devices. Sometimes this approach is taken for integrating IPv6-only devices, which means a gateway is necessary for legacy IPv4-only devices and services. In other words, this communications model is frequently used to integrate new smart devices into a legacy system with devices that are not natively interoperable with them.

A downside of this approach is that the necessary development of the application-layer gateway software and system adds complexity and cost to the overall system.

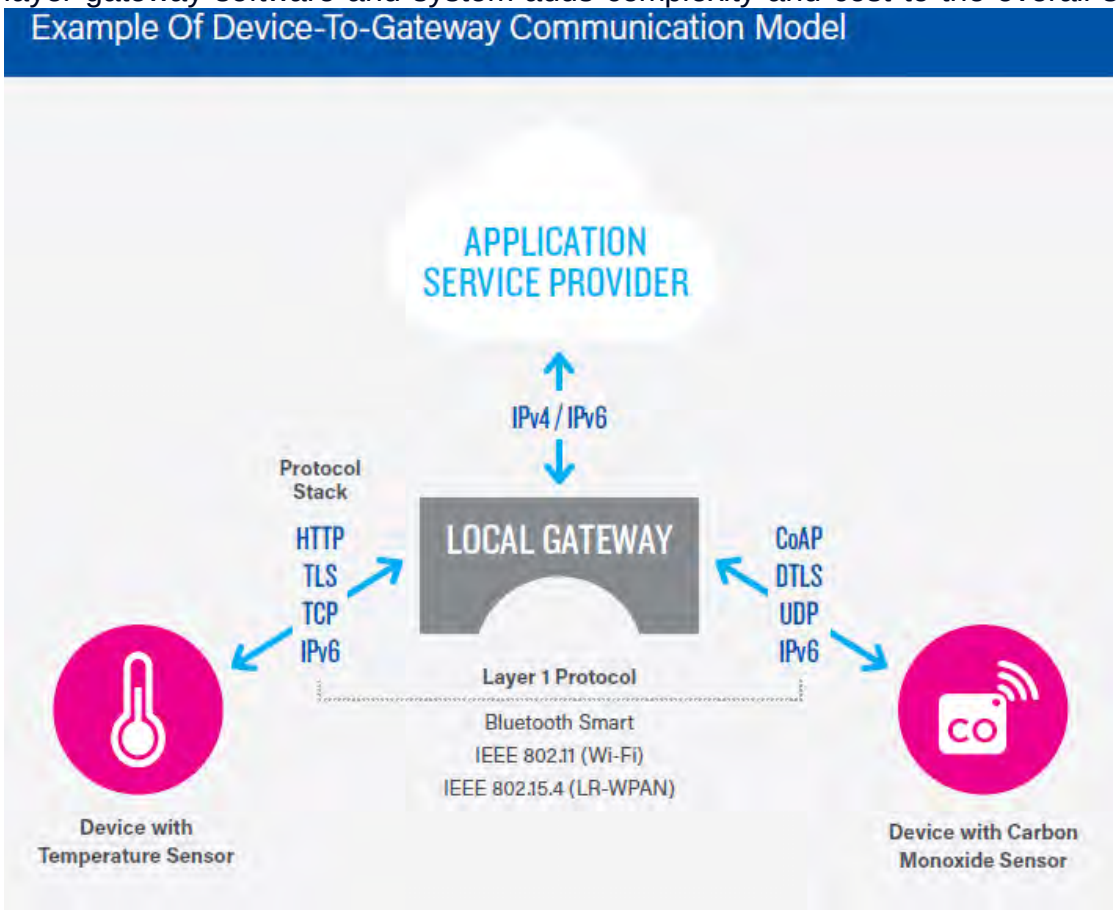


Image 3. Device to Gateway Communication Model

1.2.4 BACK END DATA SHARING MODEL

The backend data sharing model refers to a communication architecture that enables users to export and analyze smart object data from a cloud service in combination with data from other sources. This architecture supports the user's desire for granting access to the uploaded sensor data to third parties. This approach is an extension of the single device to- cloud communication model, which can lead to data silos where IoT devices upload data only to a single application service provider. A backend sharing architecture allows the data collected from single IoT device data streams to be aggregated and analyzed. For example, a corporate user in charge of an office complex would be interested in consolidating and analyzing the energy consumption and utilities data produced by all the IoT sensors and Internet enabled utility systems on the premises. Often in the single device to cloud model, the data each IoT sensor or system produces sits in a standalone data silo. An effective back-end data sharing architecture would allow the company to easily

access and analyze the data in the cloud produced by the whole spectrum of devices in the building. Also, this kind of architecture facilitates data portability needs. Effective backend data sharing architectures allow users to move their data when they switch between IoT services, breaking down traditional data silo barriers. The backend data sharing model suggests a federated cloud services approach cloud applications programmer interfaces (APIs) are needed to achieve interoperability of smart device data hosted in the cloud.

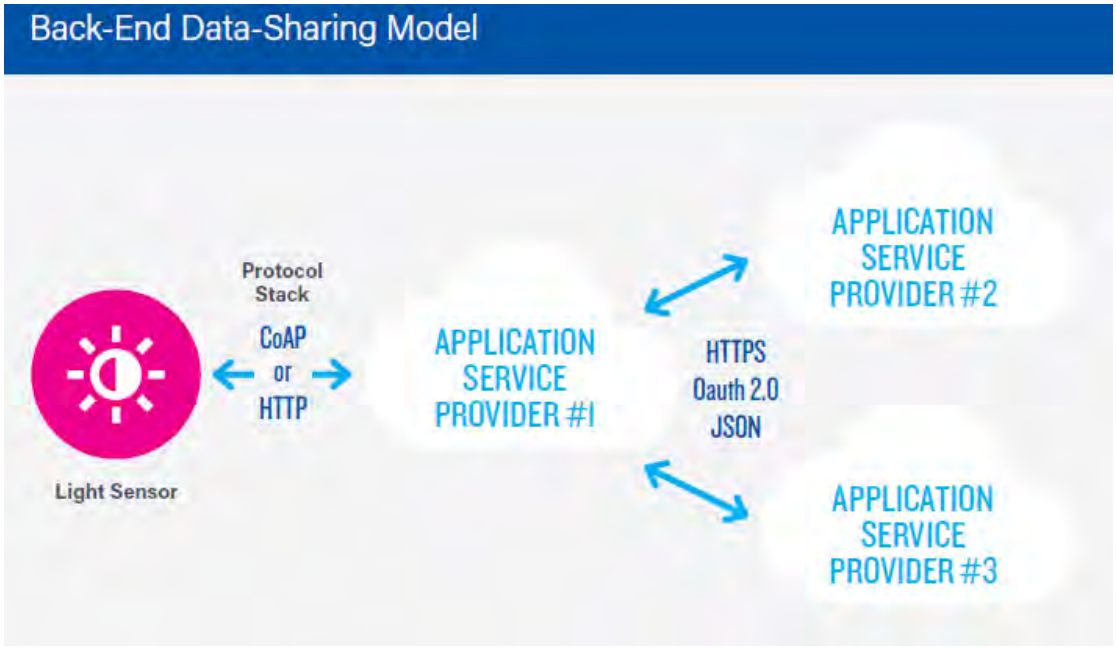


Image 4. Back-end Data Sharing Mode

1.2.5 IPv6 AND THE INTERNET OF THINGS

Although there is not a specific total amount of IoT device that will exist by 2025, specialists considered to be 100 billion. As the Internet of Things continues to grow, devices that require true end to end Internet connectivity will not be able to rely on IPv4, the protocol most Internet services use today. They will need a new enabling technology. IPv6 is a long-anticipated upgrade to the Internet's original fundamental protocol IPv6 is a long-anticipated upgrade to the Internet's original fundamental protocol the Internet Protocol (IP), which supports all communications on the Internet. The IPv6 is necessary because the Internet is running out of original IPv4 addresses. While IPv4 can support 4.3 billion devices connected to the Internet, IPv6 with 2 to the 128th power addresses, is for all practical purposes inexhaustible. This represents about 340 trillion addresses, which satisfies the demand of the estimated 100 billion IoT devices going into service in the near future. Key challenges for IoT developers are that IPv6 is not natively interoperable with IPv4 and most low cost software that is available for embedding in IoT devices implements only IPv4. Many experts believe, however, that IPv6 is the best connectivity option and will allow IoT to reach its potential.

1.3 THE DATA OF INTERNET OF THINGS

Because the generation and analysis of data is so essential to the IoT, consideration must be given to protecting data throughout its lifecycle. Managing information at this level is complex because data will flow across many administrative boundaries with different policies and intents. Individuals will surely have different

privacy goals than corporate entities, which in turn will have different goals than government or other organizations. Oftentimes, data is processed or stored on edge devices that have highly limited capabilities and are vulnerable to sophisticated attacks.

Privacy implications must also be considered to include developing an understanding of potential privacy issues when many different sources aggregate to a single point. Privacy controls are required at various points across the IoT ecosystem, particularly at point of user consent to data capture, transfer of data between IoT partners and at the points within the system that the data is stored and used.

Given the various technological and physical components that truly make up an IoT ecosystem, it is good to consider the IoT as a system-of-systems. The architecting of these systems that provide business value to organizations will often be a complex undertaking, as enterprise architects work to design integrated solutions that include edge devices, applications, transports, protocols, and analytics capabilities that make up a fully functioning IoT system. This complexity introduces challenges to keeping the IoT secure, and ensuring that a particular instance of the IoT cannot be used as a jumping off point to attack other enterprise information technology (IT) systems.

1.4 THE IOT ECOSYSTEM

The Internet of Things ecosystem will be described below ,containing processors types, operating systems and other things that give the ability to a device to be smart and internet connected:

- Processors: Arm, Cortex-M, ARC, Quark etc.
- Operating Systems : Embedded Linux , uCLinux, Android Auto, Yokto , freeRTOS, QNX, OpenWRT, CarPlay, Snappy Ubuntu, RIOT , Contiki, mbed, Android, TinyOS
- Platforms: Rapsberry Pi, Arduino, BeagleBone
- Device Types: Virtual Things, Access points, Routers ,Aggregators,ZETA Platforms
- IOT Protocols:CoAp, LWM2M, One M2M, NFC, 802.15.4 Zigbee, XMPP-Iot, HTTP, Zwave, 6LowPAN, MQTT, Bluetooth, Ethernet, SATCOM, PKE ,AMQP, DSRC, DDS,802.11 Wifi
- Integration Frameworks: Apple HOMEKIT, Temboo, CROWNSet, Thingspeak ,Wemo
- APIs: COSM,IOBridge.

CHAPTER 2

2.1 COMMON IOT SECURITY PROBLEMS

The mobility and the easy communication between internet of things device is a big advantage, but simultaneously a big disadvantage too.

In this section we will assess the security of some of the common IoT technologies that use to communicate. For the purpose of this assessment, we assume that the attacker is within range of the device's wireless transmission and can interact with it. These attacks can be achieved from outside of the building, for example in a parking lot, with an antenna. Some of the attacks require the attacker to be on the same local wireless network. All of the following technologies mentioned are potentially prone to radio jamming, allowing an attacker to disrupt connectivity to the device.

2.1.1 Wi-Fi networks (802.11)

Getting access to the home's Wi-Fi network allows an attacker to perform attacks against any connected device. The Wi-Fi standard Wired Equivalent Privacy (WEP) is considered to be insecure and should not be used. Even though Wi-Fi Protected Access II (WPA2) encryption is widely adapted, attackers can still brute-force weak passwords with a dictionary attack and get access to the network. Some broadband providers do not allow the user to change the Wi-Fi password, potentially helping attackers to brute-force accounts. Some vendors use Wi- Fi Protected Setup (WPS), which has long been found to be vulnerable to WPS PIN brute-forcing. Some manufacturers implemented client isolation security mode for Wi-Fi access points, but internet providers don't usually enable this option in home routers to allow devices to interoperate within a home network. As a result, devices connected to the network can typically access each other, not just the gateway, which is a good and desired layout.

2.1.2 Z-Wave protocol

The Z-Wave protocol itself is considered to be secure. However, researchers have previously found implementation flaws affecting specific manufacturers that allowed them to take full control of devices in Z-Wave networks. "This vulnerability was not due to a flaw in the Z-Wave protocol specification, but because of an implementation error in disabling the use of temporary key after initial network key exchange during inclusion of a node to the network," stated the research paper's authors Behrang Fouladi and Sahand Ghanoun. Similar implementation pitfalls may affect other smart home device manufacturers.

2.1.3 ZigBee

Similarly, to Z-Wave, the ZigBee protocol is considered secure from its ZigBee PRO version onwards. There have been some security concerns regarding support for plain text over-the-air (OTA) key exchange in certain profiles, which is meant to be used by manufacturers when provisioning units for the first time. Researchers have found that certain manufacturers have misused this feature. Another security concern lies in the protocol's shared network key. By stealing one of the nodes of a ZigBee network, an attacker could dump the node's internal memory and retrieve this

network key, giving them access to the network. Such a scenario may be particularly dangerous in certain configurations used for home networks that have sensors deployed outside of the house, such as an external lamp.

2.1.4 Powerline

The two main home automation protocols that make use of Powerline are:

• X10 (also supported over RF)

• Insteon (A hybrid of RF and Powerline)

One of the main concerns around these Powerline protocols is that signals can easily bleed over to the next connected networks, allowing people near the network, such as neighbors, to spy on these communications. In order to counter this, these protocols and other Powerline-based systems typically support encryption.

2.1.5 Bluetooth Low Energy

Bluetooth Low Energy, also known as Bluetooth Smart, is often used for smart home devices that do not require an internet connection, such as door locks or light bulbs. Users can typically control these devices using a mobile phone and a dedicated app. The Bluetooth Smart standard is quite flexible and leaves space open for faulty implementations that could allow attackers to remotely control these devices. For example, recently, the Bluetooth LE implementation of a wearable fitness bracelet had been completely reverse-engineered, allowing exposing the device to attack.

2.1.6 Other RF protocols

Some vendors have implemented their own radio protocol for their devices. This may result in protocols that are vulnerable to similar attacks, as with the previously described standards. For example, LightwaveRF is considered to be vulnerable to replay attacks.

2.2 MAJOR SECURITY CONCERNS

Below are some key findings which an analysis of them raises major security concerns for Internet of things Devices. During Symantec's research, they found issues such as following:

• Around 19 percent of all tested mobile apps that are used to control IoT devices did not use Secure Socket Layer (SSL) connections to the cloud
• None of the analyzed devices provided mutual authentication between the client and the server
• Some devices offered no enforcement and often no possibility of strong passwords
• Some IoT cloud interfaces did not support two-factor authentication (2FA)
• Many IoT services did not have lock-out or delaying measures to protect users' accounts against brute-force attacks
• Some devices did not implement protections against account harvesting
• Many of the IoT cloud platforms included common web application vulnerabilities
• We found ten security issues in fifteen web portals used to control IoT devices without performing any deep tests. Six of them were serious issues, allowing unauthorized access to the backend systems.

• Most of the IoT services did not provide signed or encrypted firmware updates, if updates were provided at all

As a conclusion, there are still many devices that do not use encrypted communications or proper authentication. It is crucial that smart home devices, or any IoT devices for that matter, use mutual authentication and encryption. IoT devices often have less memory and slower CPUs, so they may be unable to use the same encryption methods as a traditional computer does, but that is no excuse for the lack of strong encryption. There are efficient cryptographic methods designed for small scale devices, such as Elliptic Curve Cryptography (ECC), which can be used. that is run on a smart device, be it the firmware or application, should be verified through a chain of trust.

Protecting the code and securing the device creates a trusted baseline. Vendors should provide a simple and automated way for users to update their device in order to ensure that common security issues can be fixed quickly and efficiently. IoT devices should only accept signed firmware as standard. Where applicable, security analytics features should be provided in the overall device management strategy. Cloud control interfaces present another weak point of many IoT. Users should not be forced to use cloud setups if all they want to do is to do basic tasks such as turning on the lights in their homes. Vendors need to allow strong, complex passwords to be used. Restricting authentication to simple four-digit PIN codes does not sufficiently protect the device, especially if this issue is combined with the lack of any brute-force protection mechanism. Even when strong passwords are use, we found that common web application vulnerabilities, such as SQL injection or remote file inclusion, are often present in these cloud control portals as well. Vendors need to ensure that their services are not vulnerable to the OWASP's top ten web application vulnerabilities. For IoT devices such as smoke alarms, it is also crucial that the vendor has considered what happens when there is a power outage or the network gets jammed. Will the user be notified or will the malfunctioning safety device go unnoticed?

In the near future, a lot of people could have a variety of devices connected to their home networks. This will lead to smarter smart hubs that allow commands based on logical conditions, such as "if this, then that". This adds to the complexity of the problem, as now a problem in one device can trigger the shutdown of another.

There are already applications available which allow you to do exactly this. In order to perform the actions, the application needs to be authorized to access the smart devices. This makes the smart hub an ideal central point of attack, as changing such rules could have a catastrophic effect on all devices connected to the network. With all of these issues affecting the devices on different levels, it is currently not easy to deploy multiple smart devices in a secure fashion at home.

2.3 PRIVACY CONSIDERATIONS

2.3.1 INTERNET OF THINGS PRIVACY BACKGROUND

Respect for privacy rights and expectations is integral to ensuring trust in the Internet, and it also impacts the ability of individuals to *speak*, *connect*, and *choose* in meaningful ways. These rights and expectations are sometimes framed in terms of ethical data handling, which emphasizes the importance of respecting an individual's expectations of privacy and the fair use of their data. The Internet of Things can challenge these traditional expectations of privacy. IoT often refers to a large network of sensor enabled devices designed to collect data about their environment, which frequently includes data related to people.

This data presumably provides a benefit to the device's owner, but frequently to the device's manufacturer or supplier as well. IoT data collection and use becomes a privacy consideration when the individuals who are observed by IoT devices have different privacy expectations regarding the scope and use of that data than those of the data collector. Seemingly harmless combinations of IoT data streams also can jeopardize privacy. When individual data streams are combined or correlated, often a more invasive digital portrait is painted of the individual than can be realized from an individual IoT data stream.

For example, a user's Internet-enabled toothbrush might capture and transmit innocuous data about a person's tooth-brushing habits. But if the user's refrigerator reports the inventory of the foods he eats and his fitness-tracking device reports his activity data, the combination of these data streams paint a much more detailed and private description of the person's overall health. This data-aggregation effect can be particularly potent with respect to IoT devices because many produce additional metadata like time stamps and geolocation information, which adds even more specificity about the user.

In other situations, the user might not be aware that an IoT device is collecting data about the individual and potentially sharing it with third parties. This type of data collection is becoming more prevalent in consumer devices like smart televisions and video game devices. These kinds of products have voice recognition or vision features that continuously listen to conversations or watch for activity in a room and selectively transmit that data to a cloud service for processing, which sometimes includes a third party. A person might be in the presence of these kinds of devices without knowing their conversation or activities are being monitored and their data captured. These kinds of features may provide a benefit to an informed user, but can pose a privacy problem for those who are unaware of the presence of the devices and have no meaningful influence over how that collected information is used. Independent of whether the user is aware of and consents to having their IoT data collected and analyzed, these situations highlight the value of these personalized data streams to companies and organizations seeking to collect and capitalize on IoT information.

The demand for this information exposes the legal and regulatory challenges facing data protection and privacy laws. These kinds of privacy problems are critical to address because they have implications on our basic rights and our collective

ability to trust the Internet. From a broad perspective, people recognize their privacy is intrinsically valuable, and they have expectations of what data can be collected about them and how other parties can use that data. This general notion about privacy holds true for data collected by Internet of Things devices, but those devices can undermine the user's ability to express and enforce privacy preferences. If users lose confidence in the Internet because their privacy preferences aren't being respected in the Internet of Things, then the greater value of the Internet may be diminished.

2.3.2 UNIQUE PRIVACY ASPECTS OF THE INTERNET OF THINGS.

Generally, privacy concerns are amplified by the way in which the Internet of Things expands the feasibility and reach of surveillance and tracking. Characteristics of IoT devices and the ways they are used redefine the debate about privacy issues, because they dramatically change how personal data is collected, analyzed, used, and protected. The traditional "notice and consent" online privacy model, in which users assert their privacy preferences by interacting directly with information presented on a computer or mobile screen (e.g. by clicking "I agree"), breaks down when systems provide no mechanism for user interaction. IoT devices frequently have no user interface to configure privacy preferences, and in many IoT configurations users have no knowledge or control over the way in which their personal data is being collected and used. This causes a gulf between the user's privacy preferences and the data-collecting behavior of the IoT device. There might be less incentive for IoT vendors to offer a mechanism for users to express their privacy preferences if they regard the data collected as being non-personal data. However, experience shows that data not traditionally considered personal data might actually be personal data or become personal data when combined with other data.

Assuming an effective mechanism can be developed to enable a user to express informed consent of their privacy preferences to IoT devices, that mechanism needs to handle the large number of IoT devices a user must control. It is not realistic to think that a user will directly interact with each and every IoT device they encounter throughout the day to express their privacy preferences. Instead, privacy interface mechanisms need to be scalable to the size of the IoT problem, while still being comprehensive and practical from a user perspective. The Internet of Things can threaten a person's expectations of privacy in common situations. There are social norms and expectations of privacy that differ in public spaces versus private spaces, and IoT devices challenge these norms. For example, IoT monitoring technologies like surveillance cameras or location tracking systems that normally operate in public spaces are migrating into traditionally private spaces like the home or personal vehicle in which our expectations of privacy are very different. In doing so, they challenge what many societies recognize as the "right to be left alone" in one's home or private space. Also individuals' expectations of privacy in spaces they consider to be public (e.g. parks, shopping malls, train stations) are being challenged by the increased nature and extent of monitoring in those spaces.

IoT devices often operate in contexts in which proximity exposes multiple people to the same data collection activity. For example, a geolocation tracking

sensor in an automobile would record location data about all occupants of the vehicle, whether or not all the occupants want their location tracked. It may even track individuals in nearby vehicles. In these kinds of situations, it might be difficult or impossible to distinguish, much less honor, individual privacy preferences. Big data analytics applied to aggregated personal data already represents a substantial risk of privacy invasion and potential discrimination. This risk is amplified in the Internet of Things by the scale and greater. intimacy of personal data collection. IoT devices can collect information about people with an unprecedented degree of specificity and pervasiveness; aggregation and correlation of these data can create detailed profiles of individuals that create the potential for discrimination and other harms. The sophistication of this technology can create situations that expose the individual to physical, criminal, financial or reputational harm. The ubiquity, familiarity, and social embrace of many IoT devices might create a false sense of security and encourage individuals to divulge sensitive or private information without full awareness or appreciation of the potential consequences of doing so.

2.3.3 IOT PRIVACY QUESTIONS

These privacy issues would be challenging even if the interests and motivations of all of the participants in the IoT ecosystem were well aligned. However, we know that there can be unbalanced or unfair relationships and interests between those who are exposed to personal data collection and those who aggregate, analyze, and use the data. The data source might see an unwelcome intrusion into private space, often without consent, control, choice, or even awareness. The data collector, however, might consider this a beneficial resource that can add value to products and services as well as provide new revenue streams. Because IoT challenges our notions of privacy in new ways, key questions need to be asked when re-evaluating online privacy models in the context of IoT. Some questions that have been raised include:

- How do we resolve the marketplace relationship between data sources and data collectors in the context of IoT? Personal data has personal and commercial value that sources and collectors value differently, both individually and in aggregate; both parties have legitimate interests that may conflict. How might those distinct interests be expressed in a way that leads to fair and consistent rules for both sources and collectors concerning access, control, transparency, and protection?
- How can privacy policies and practices be made readily available and understandable in the context of IoT? What are the alternatives to the traditional “notice and consent” privacy model that will address the unique aspects of the Internet of Things? What is an effective model for expressing, applying, and enforcing individual privacy preferences and multi-party preferences? Could such a multi-party model be constructed, and if so, what would it look like? How might it be applied to specific circumstances involving individual privacy preferences? Is there a market for outsourcing the management of privacy settings to commercial services designed to put users’ preferences into effect? Is there a role for a privacy proxy that would express and enforce a user’s preferences across an array of devices, while eliminating the need for direct interaction with each one?
- Privacy norms and expectations are closely related to the social and cultural context of the user, which will vary from one group or nation to another. Many

IoT scenarios involve device deployments and data collection activities with multinational or global scope that cross social and cultural boundaries. What will that mean for the development of a broadly applicable privacy protection model for the Internet of Things? How can IoT devices and systems be adapted to recognize and honor the range of privacy expectations of the users and different laws?

- How can we encourage IoT device manufacturers to integrate privacy-by-design principles into their core values? How do we foster the inclusion of consumer privacy considerations in every phase of product development and operation? How do we reconcile functionality and privacy requirements? In principle, manufacturers should expect that privacy-respecting products and practices build long-term customer trust, satisfaction, and brand loyalty. Is that a sufficiently compelling motivation, when matched against the competing desires for design simplicity and speed to market? Should devices be designed with default settings configured for the most conservative data collection mode (i.e. opt out of data collection by default)?
- How should we protect data collected by IoT that appears not to be personal at the point of collection or has been “de-identified”, but may at some point in the future become personal data (e.g. because data can be re-identified or combined with other data

The Internet of Things creates unique challenges to privacy that go beyond the data privacy issues that currently exist. Strategies need to be developed to respect individual privacy choices across a broad spectrum of expectations, while still fostering innovation in new IoT technology.

2.4 The IOT security Challenge

The term security subsumes a wide range of different concepts. In the first place, it refers to the basic provision of security services including confidentiality, authentication, integrity, authorization, non-repudiation, and availability. These security services can be implemented by means of different cryptographic mechanisms, such as block ciphers, hash functions, or signature algorithms. For each of these mechanisms, a solid key management infrastructure is fundamental to handling the required cryptographic keys.

Ensuring the security, reliability, resilience, and stability of Internet applications and services is critical to promoting *trust* and use of the Internet. As users of the Internet, we need to have a high degree of trust that the Internet, its applications, and the devices linked to it are secure enough to do the kinds of activities we want to do online in relation to the risk tolerance associated with those activities. The Internet of Things is no different in this respect, and security in IoT is fundamentally linked to the ability of users to trust their environment. If people don’t believe their connected devices and their information are reasonably secure from misuse or harm, the resulting erosion of trust causes a reluctance to use the Internet. This has global consequences to electronic commerce, technical innovation, free speech, and practically every other aspect of online activities. Indeed, ensuring security in IoT products and services should be considered a top priority for the sector.

As we increasingly connect devices to the Internet, new opportunities to exploit potential security vulnerabilities grow. Poorly secured IoT devices could serve as entry points for cyberattack by allowing malicious individuals to re-program a device or cause it to malfunction. Poorly designed devices can expose user data to theft by leaving data streams inadequately protected. Failing or malfunctioning devices also can create security vulnerabilities. These problems are just as large or larger for the small, cheap, and ubiquitous smart devices in the Internet of Things as they are for the computers that have traditionally been the endpoints of Internet connectivity. Competitive cost and technical constraints on IoT devices challenge manufacturers to adequately design security features into these devices, potentially creating security and long-term maintainability vulnerabilities greater than their traditional computer counterparts.

Along with potential security design deficiencies, the sheer increase in the number and nature of IoT devices could increase the opportunities of attack. When coupled with the highly interconnected nature of IoT devices, every poorly secured device that is connected online potentially affects the security and resilience of the Internet *globally*, not just locally. For example, an unprotected refrigerator or television in the US that is infected with malware might send thousands of harmful spam emails to recipients worldwide using the owner's home Wi-Fi Internet connection.

Day by day, we become more connected and dependent on IoT devices for essential services, and we need the devices to be secure, while recognizing that no device can be absolutely secure. This increasing level of dependence on IoT devices and the Internet services they interact with also increases the pathways for wrongdoers to gain access to devices. Perhaps we could unplug our Internet-connected TVs if they get compromised in a cyber-attack, but we can't so easily turn off a smart utility power meter or a traffic control system or a person's implanted pacemaker if they fall victim to malicious behavior. This is why security of IoT devices and services is a major discussion point and should be considered a critical issue. We increasingly depend on these devices for essential services, and their behavior may have global reach and impact.

When thinking about Internet of Things devices, it is important to understand that security of these devices is not absolute. IoT device security is not a binary proposition of secure or insecure. Instead, it is useful to conceptualize IoT security as a spectrum of device vulnerability. The spectrum ranges from totally unprotected devices with no security features to highly secure systems with multiple layers of security features. In an endless cat-and-mouse game, new security threats evolve, and device manufacturers and network operators continuously respond to address the new threats. The overall security and resilience of the Internet of Things is a function of how security risks are assessed and managed. Security of a device is a function of the risk that a device will be compromised, the damage such compromise will cause, and the time and resources required to achieve a certain level of protection. If a user cannot tolerate a high degree of security risk as in the case of the operator of a traffic control system or person with an implanted, Internetenabled medical device, then she may feel justified in spending a considerable amount of

resources to protect the system or device from attack. Likewise, if she is not concerned that her refrigerator might be hacked and used to send spam messages, then she may not feel compelled to pay for a model that has a more sophisticated security design if it makes the device more costly or complicated.

Several factors influence this risk assessment and mitigation calculation. Factors include having a clear understanding of the present security risks and the potential future risks; the estimated economic and other costs of harm if the risks are realized; and the estimated cost to mitigate the risks. While these kinds of security tradeoffs are often made from an individual user or organizational perspective, it is also important to consider the interrelatedness of IoT devices as part of a larger IoT ecosystem. The networked connectivity of IoT devices means that security decisions made locally about an IoT device can have global impacts on other devices. As a matter of principle, developers of smart objects for the Internet of Things have an obligation in ensuring that those devices do not expose either their own users or others to potential harm. As a matter of business and economics, vendors have an interest in reducing their cost, complexity, and time to market. For example, IoT devices that are high–volume, low–margin components that already represent a cost added to that of the product in which they are embedded are becoming quite common; adding more memory and a faster processor to implement security measures could easily make that product commercially uncompetitive. In economic terms, lack of security for IoT devices results in a negative externality, where a cost is imposed by one party (or parties) on other parties. A classic example is pollution of the environment, where the environmental damage and cleanup costs (negative externalities) of a polluter's actions are borne by other parties. The issue is that the cost of the externality imposed on others is not normally factored into the decision-making process, unless, as is the case with pollution, a tax is imposed on the polluter to convince him to lower the amount of pollution. In the case of information security, as discussed by Bruce Schneier, an externality arises when the vendor creating the product does not bear the costs caused by any insecurity; in this case, liability law can influence vendors to account for the externality and develop more security products.

Iot devices tend to differ from traditional computers and computing devices in important ways that challenge security issues:

- Many Internet of Things devices, such as sensors and consumer items, are designed to be deployed at a massive scale that is orders of magnitude beyond that of traditional Internet connected devices. As a result, the potential quantity of interconnected links between these devices is unprecedented. Further, many of these devices will be able to establish links and communicate with other devices on their own in an unpredictable and dynamic fashion. Therefore, existing tools, methods, and strategies associated with IoT security may need new consideration.
- Many IoT deployments will consist of collections of identical or near identical devices. This homogeneity magnifies the potential impact of any single

security vulnerability by the sheer number of devices that all have the same characteristics. For example, a communication protocol vulnerability of one company's brand of Internet-enabled light bulbs might extend to every make and model of device that uses that same protocol or which shares key design or manufacturing characteristics.

- Many Internet of Things devices will be deployed with an anticipated service life many years longer than is typically associated with high-tech equipment. Further, these devices might be deployed in circumstances that make it difficult or impossible to reconfigure or upgrade them; or these devices might outlive the company that created them, leaving orphaned devices with no means of long-term support. These scenarios illustrate that security mechanisms that are adequate at deployment might not be adequate for the full lifespan of the device as security threats evolve. As such, this may create vulnerabilities that could persist for a long time. This is in contrast to the paradigm of traditional computer systems that are normally upgraded with operating system software updates throughout the life of the computer to address security threats. The long-term support and management of IoT devices is a significant security challenge.
- Many IoT devices are intentionally designed without any ability to be upgraded, or the upgrade process is cumbersome or impractical. For example, consider the 2015 Fiat Chrysler recall of 1.4 million vehicles to fix a vulnerability that allowed an attacker to wirelessly hack into the vehicle. These cars must be taken to a Fiat Chrysler dealer for a manual upgrade, or the owner must perform the upgrade themselves with a USB key. The reality is that a high percentage of these autos probably will not be upgraded because the upgrade process presents an inconvenience for owners, leaving them perpetually vulnerable to cybersecurity threats, especially when the automobile appears to be performing well otherwise.
- Many IoT devices operate in a manner where the user has little or no real visibility into the internal workings of the device or the precise data streams they produce. This creates a security vulnerability when a user believes an IoT device is performing certain functions, when in reality it might be performing unwanted functions or collecting more data than the user intends. The device's functions also could change without notice when the manufacturer provides an update, leaving the user vulnerable to whatever changes the manufacturer makes.
- Some IoT devices are likely to be deployed in places where physical security is difficult or impossible to achieve. Attackers may have direct physical access to IoT devices. Anti-tamper features and other design innovations will need to be considered to ensure security.

2.5 SECURITY CHALLENGES

There are many challenges to deploying a secure IoT implementation, and many of the existing security technologies on the market will play a role in mitigating IoT risks within an enterprise. However, the IoT also introduces new challenges to security engineering. Many of these would benefit from targeted research or industry collaboration to to determine the optimal long-term approaches to resolution.

2.5.1 Many IoT Systems are poorly designed and implemented, using diverse protocols and technologies that create complex configurations.

The IoT encompasses edge devices, messaging and transport protocols, Application Programming Interfaces (APIs), data analytics, storage, software, and various other technology concepts. Edge devices themselves are complex, consisting of multiple layers of technology and requiring an understanding of hardware, firmware, software and a plethora of protocols. All of this can be applied to myriad use cases across many industries.

Before being able to secure a system, it is important to first understand the functional and technological details of the system to be secured. This will require security engineers to work closely with the developers of the IoT capability to introduce security requirements early in the design process. Using a methodical systems security engineering approach for each IoT implementation within an enterprise is recommended.

Taking a systems security engineering approach to IoT implementations allows designers to identify areas of complexity that can be simplified. As an example, limiting implementations to the use of as few protocols and touch points as possible.

2.5.2 Lack of mature IoT technologies and business processes

Standards supporting the IoT have not yet been fully developed, leaving the market open to competing platforms, protocols, and interfaces. This lack of standards drives increased complexity which can introduce vulnerabilities and provides attackers with a way to infiltrate the enterprise.

2.5.3 Limited guidance for lifecycle maintenance and management of IoT devices

Guidance on the secure configuration of the limited capability operating systems that underlie many IoT edge devices is limited or nonexistent.
Performing firmware, software and patch updates for IoT devices will require a new approach with considerations given to identifying update provisioning obligations and responsibilities throughout the supply chain.

Organization’s procuring IoT assets should also clearly understand and agree on the vendor’s model for licensing to ensure that they are able to continue receiving patches and software updates throughout the course of the IoT asset’s life. If IoT devices fall behind on required security updates, they will be much easier for attackers to exploit. In this regard, organizations should consider the likelihood that IoT devices will eventually become unsupported as phase-out dates come into play from each vendor.

Keeping track of IoT devices and the software and firmware on each device is also an issue. The amount of IoT devices alone introduces a challenge to effectively managing them.

2.5.4 The IoT introduces unique physical security concerns

Many IoT edge devices will be deployed in exposed environments, allowing attackers to more easily acquire them for further lab analysis. This is concerning because most IoT edge devices are limited in capability, requiring software-based solutions for the protection of sensitive material such as cryptographic keys.
Attackers with sufficient resources can reverse engineer these edge devices. Ideally, the use of tamper-resistant protections would be implemented however this may not always be feasible. The fact that many IoT applications desire very low-cost devices causes a conflict with devices' ability to withstand attacks and tampering.

2.5.5 IoT privacy concerns are complex and not always readily evident.

Some privacy concerns are not readily identifiable and some concerns are not solvable by simply enforcing confidentiality protections, identity or location to transactions.

2.5.6 Limited best practices available for IoT developers

Many IoT developers are not yet familiar with secure development best practices. The rush to create new IoT-based capabilities will likely result in limited focus on the security of the new functionality being created.

2.5.7 There is a lack of standards for authentication and authorization of IoT edge devices

Requirement for low-power and wearable devices bring a wealth of new, simpler wireless protocols, which often meshes together and do not implement mature and secure encryption and authentication; these protocols can be attacked "on the fly" and without physical contact
Some IoT devices have no authentication capabilities while others have limited support. Very few have capabilities that support multi-factor authentication. It is also not clear how useful multi-factor authentication for IoT edge devices will be in general. One of the primary benefits of traditional 2-factor authentication is that one of the "factors" is "out-of-band" relative to the other. But, in IoT devices, both of the credential (e.g., keys) may need to be stored in the same device, losing the out-of-band benefit.
Although some standards or commercial options are available (e.g., certificate authentication, commercial or semi-commercial identity providers such as Google, there is a lack of ability to create device-specific profiles and authorization options and the privacy implications of using these services providers has not been fully explored.

2.5.8 There are no best practices for IoT-based incident response activities.

Organizations must be able to plan for the compromise of IoT devices, keys and certificates. This includes performing forensic analysis on compromised systems and devices.

2.5.9 Audit and Logging standards are not defined for IoT components Monitoring

IoT edge devices for security events poses unique difficulties. Many of these edge devices will be single-purpose sensors that may not be capable of tracking all interactions with the device. Other devices may be limited in their ability to instantiate

an RF connection for the purpose of sending audit logs, based on battery constraints. Obtaining near real-time situational awareness of the security posture of IoT devices will be difficult.

Another challenge is aggregating log data from many widespread IoT segments into a single event management system, and then actually being able to derive some intelligence from the activities within each of these segments.

2.5.10 Restricted interfaces available to interact IoT devices with security devices and applications. No focus yet on identifying methods for achieving situational awareness of the security posture of an organization's IoT assets.

Integrating IoT devices into an organization's existing security system would provide situational awareness of the overarching security posture of the organization. Unfortunately, there are typically no interfaces made available to connect with existing SIEM systems, and options are typically limited for connecting with Identity and Access Management systems and other security systems. Given that this is the case, it is likely that intermediary products will soon rise to support brokering between IoT device pools and an organization's security infrastructure.

2.5.11 Security standards for platform configurations involving virtualized IoT platforms supporting multi-tenancy is immature.

This involves use cases where the "cloud" stretches all the way out to the device (e.g., two businesses being hosted as tenants on the same physical IoT platform). This results in the need for lightweight, yet secure virtualization /isolation solutions.

CHAPTER 3

3.1 TOP 10 VULNERABILITIES AND COUNTERMEASURES

At IoT security there is a misconception that It is all about the device, or the network or the clients .There are many surface areas involved and each of these need to be evaluated.

A holistic approach is required and the elements to be considered are:

- The Internet of Things Device
- The Cloud
- The Mobile Application
- The Network Interfaces
- The Software
- Use of Encryption
- Use of Authentication
- Physical Security
- USB ports

Below we will analyze the top 10 categories that covers the entire device and all surface area to get a good assessment of overall security.

1.Insecure Web Interface

Threat Agent: Consider anyone who has access to the web interface including internal and external users(Applciation Specific)

Attack Vectors: Attacker uses weak credentials, captures plain-text credentials or enumerates accounts to access the web interface. Attack could come from external or internal users.(Exploitability easy)

Security Weakness: An insecure web interface can be present when issues such as account enumeration, lack of account lockout or weak credenitals are present. Insecure web interfaces are prevalent as the intent is to have these interfaces exposed only on internal networks, however threats from the internal users can be just as significant as threats from external users. Issues with the web interface are easy to discover when examining the interface manually along with automated testing tools to identify other issues such as cross-site scripting.(Detectability Easy)

Technical Impacts: Insecure web interfaces can result in data loss or corruption, lack of accountability, or denial of access and can lead to complete device takeover.(Impact Severe)

Bussiness Impacts: Consider the business impact of poorly secured web interfaces that could lead to compromised devices along with compromised customers. Could your customers be harmed? Could your brand be harmed?

Is My Web Interface Secure?

Checking for an Insecure Web Interface includes:

- Determining if the default username and password can be changed during initial product setup
- Determining if a specific user account is locked out after 3 - 5 failed login attempts
- Determining if valid accounts can be identified using password recovery mechanisms or new user pages
- Reviewing the interface for issues such as cross-site scripting, cross-site request forgery and sql injection.

How do I make My web Interface Secure?

A secure web interface requires:

1. Default passwords and ideally default usernames to be changed during initial setup
2. Ensuring password recovery mechanisms are robust and do not supply an attacker with information indicating a valid account
3. Ensuring web interface is not susceptible to XSS, SQLi or CSRF
4. Ensuring credentials are not exposed in internal or external network traffic
5. Ensuring weak passwords are not allowed
6. Ensuring account lockout after 3 -5 failed login attempts

Example Attack Scenarios

Scenario #1: The web interface presents "Forgot Password" functionality which upon entering an invalid account informs the attacker that the account does not exist. Once valid accounts are identified, password guessing can begin for an indefinite amount of time if no account lockout controls exist.

Account john@doe.com does not exist.

Scenario #2: Web interface is susceptible to cross-site scripting.

http://xyz.com/index.php?user=<script>alert(123)</script> ... Response from browser is an alert popup

In the cases above, the attacker is able to easily determine if an account is valid or not and is also able to determine that the site is susceptible to cross-site scripting (XSS).

2 Insufficient Authentication/Authorization

Threat Agent: Consider anyone who has access to the web interface, mobile interface or cloud interface including internal and external users.

Attack Vectors: Attacker uses weak passwords, insecure password recovery mechanisms, poorly protected credentials or lack of granular access control to access a particular interface. Attack could come from external or internal users.

Security Weakness: Authentication may not be sufficient when weak passwords are used or are poorly protected. Insufficient authentication/authorization is prevalent as it is assumed that interfaces will only be exposed to users on internal networks and not to external users on other networks. Deficiencies are often found to be present across all interfaces. Many Issues with authentication/authorization are easy to discover when examining the interface manually and can also be discovered via automated testing.

Technical Impacts: Insufficient authentication/authorization can result in data loss or corruption, lack of accountability, or denial of access and can lead to complete compromise of the device and/or user accounts.

Business Impacts: Consider the business impact of compromised user accounts and possibly devices. All data could be stolen, modified, or deleted. Could your customers be harmed?

Is My Authentication/Authorization Sufficient?

Checking for Insufficient Authentication includes:

- Attempting to use simple passwords such as "1234" is a fast and easy way to determine if the password policy is sufficient across all interfaces
- Reviewing network traffic to determine if credentials are being transmitted in clear text
- Reviewing requirements around password controls such as password complexity, password history check, password expiration and forced password reset for new users
- Reviewing whether re-authentication is required for sensitive features

Checking for Insufficient Authorization includes:

- Reviewing the various interfaces to determine whether the interfaces allow for separation of roles. For example, all features will be accessible to administrators, but users will have a more limited set of features available.
- Reviewing access controls and testing for privilege escalation

How Do I Make My Authentication/Authorization Better?

Sufficient authentication/authorization requires:

1. Ensuring that the strong passwords are required
2. Ensuring granular access control is in place when necessary

3. Ensuring credentials are properly protected
4. Implement two factor authentication where possible
5. Ensuring that password recovery mechanisms are secure
6. Ensuring re-authentication is required for sensitive features
7. Ensuring options are available for configuring password controls
8. Ensuring credential can be revoked
9. The app authentication is required
10. The device authentication is required
11. The server authentication is required
12. Manage authenicated user id(credential info.) and the user's device id, the user's app id mapping table in the authentication server
13. Ensuring that the authentication token/session key issuing to client is always different
14. Ensuring that the user id, app id, device id is universally unique

Example Attack Scenarios

Scenario #1: The interface only requires simple passwords.

```
Username = Bob; Password = 1234
```

Scenario #2: Username and password are poorly protected when transmitted over the network.

```
Authorization: Basic YWRtaW46MTIzNA==
```

In the cases above, the attacker is able to either easily guess the password or is able to capture the credentials as they cross the network and decode it since the credentials are only protected using Base64 Encoding.

3. Insecure Network Services

Threat Agent: Consider anyone who has access to the device via a network connection, including external and internal users.

Attack Vectors: Attacker uses vulnerable network services to attack the device itself or bounce attacks off the device. Attack could come from external or internal users.

Security Weakness: Insecure network services may be susceptible to buffer overflow attacks or attacks that create a denial of service condition leaving the device inaccessible to the user. Denial of service attacks against other users may also be facilitated when insecure network services are available. Insecure network services can often be detected by automated tools such as port scanners and fuzzers.

Technical Impacts: Insecure network services can result in data loss or corruption, denial of service or facilitation of attacks on other devices.

Business Impacts: Consider the business impact of devices which have been rendered useless from a denial of service attack or the device is used to facilitate attacks against other devices and networks. Could your customers or other users be harmed?

Are My Network Services Secure?

Checking for Insecure Network Services includes:

- Determining if insecure network services exist by reviewing your device for open ports using a port scanner
- As open ports are identified, each can be tested using any number of automated tools that look for DoS vulnerabilities, vulnerabilities related to UDP services and vulnerabilities related to buffer overflow and fuzzing attacks
- Reviewing network ports to ensure they are absolutely necessary and if there are any ports being exposed to the internet using UPnP.

How Do I Secure My Network Services?

Securing network services requires:

1. Ensuring only necessary ports are exposed and available.
2. Ensuring services are not vulnerable to buffer overflow and fuzzing attacks.
3. Ensuring services are not vulnerable to DoS attacks which can affect the device itself or other devices and/or users on the local network or other networks.
4. Ensuring network ports or services are not exposed to the internet via UPnP for example
5. The abnormal service request traffic should be detected and blocked on service gateway layer

Example Attack Scenarios

Scenario #1: Fuzzing attack causes network service and device to crash.

```
GET %s%s%s%s%s%s%s%s%s%s%s%s%s%s%s HTTP/1.0
```

Scenario #2: Ports open to the internet possibly without the user's knowledge via UPnP.

Port 80 and 443 exposed to the internet via a home router.

In the cases above, the attacker is able to disable the device completely with an HTTP GET or access the device via the internet over port 80 and/or port 443.

4.Lack of Transport Encryption

Threat Agent: Consider anyone who has access to the network the device is connected to, including external and internal users.

Attack Vectors: Attacker uses the lack of transport encryption to view data being passed over the network. Attack could come from external or internal users

Security Weakness: Lack of transport encryption allows data to be viewed as it travels over local networks or the internet. Lack of transport encryption is prevalent on local networks as it is easy to assume that local network traffic will not be widely visible, however in the case of a local wireless network, misconfiguration of that wireless network can make traffic visible to anyone within range of that wireless network. Many Issues with transport encryption are easy to discover simply by viewing network traffic and searching for readable data. Automated tools can also look for proper implementation of common transport encryption such as SSL and TLS.

Technical Impacts: Lack of transport encryption can result in data loss and depending on the data exposed, could lead to complete compromise of the device or user accounts.

Business Impacts: Consider the business impact of exposed data as it travels across various networks. Data could be stolen or modified. Could your users be harmed by having their data exposed

Do I use Transport Encryption?

Checking for Lack of Transport Encryption includes:

- Reviewing network traffic of the device, its mobile application and any cloud connections to determine if any information is passed in clear text
- Reviewing the use of SSL or TLS to ensure it is up to date and properly implemented
- Reviewing the use of any encryption protocols to ensure they are recommended and accepted

How Do I Use Transport Encryption?

Sufficient transport encryption requires:

1. Ensuring data is encrypted using protocols such as SSL and TLS while transiting networks.
2. Ensuring other industry standard encryption techniques are utilized to protect data during transport if SSL or TLS are not available.
3. Ensuring only accepted encryption standards are used and avoid using proprietary encryption protocols
4. Ensuring the message payload encryption
5. Ensuring the secure encryption key handshaking
6. Ensuring received data integrity verification

Example Attack Scenarios

Scenario #1: The cloud interface uses only HTTP.

```
http://www.xyzcloudsite.com
```

Scenario #2: Username and password are transmitted in the clear over the network.

```
http://www.xyzcloud.com/login.php?userid=3&password=1234
```

In the cases above, the attacker has the ability to view sensitive data in the clear due to lack of transport encryption.

5.Privacy Concerns

Threat Agent: Consider anyone who has access to the device itself, the network the device is connected to, the mobile application and the cloud connection including external and internal users.

Attack Vectors: Attacker uses multiple vectors such as insufficient authentication, lack of transport encryption or insecure network services to view personal data which is not being properly protected or is being collected unnecessarily. Attack could come from external or internal users.

Security Weakness: Privacy concerns generated by the collection of personal data in addition to the lack of proper protection of that data is prevalent. Privacy concerns are easy to discover by simply reviewing the data that is being collected as the user sets up and activates the device. Automated tools can also look for specific patterns of data that may indicate collection of personal data or other sensitive data.

Technical Impacts: Collection of personal data along with a lack of protection of that data can lead to compromise of a user's personal data..

Business Impacts: Consider the business impact of personal data that is collected unnecessarily or isn't protected properly. Data could be stolen. Could your customers be harmed by having this personal data exposed?

Does My Device Present Privacy Concerns?

Checking for Privacy Concerns includes:

- Identifying all data types that are being collected by the device, its mobile application and any cloud interfaces
- The device and it's various components should only collect what is necessary to perform its function
- Personally identifiable information can be exposed when not properly encrypted while at rest on storage mediums and during transit over networks
- Reviewing who has access to personal information that is collected
- Determining if data collected can be de-identified or anonymized
- Determining if data collected is beyond what is needed for proper operation of the device (Does the end-user have a choice for this data collection?)
- Determining if a data retention policy is in place

How Do I Prevent Privacy Concerns?

Minimizing privacy concerns requires:

1. Ensuring only data critical to the functionality of the device is collected
2. Ensuring that any data collected is of a less sensitive nature (i.e., try not to collect sensitive data)
3. Ensuring that any data collected is de-identified or anonymized
4. Ensuring any data collected is properly protected with encryption
5. Ensuring the device and all of its components properly protect personal information
6. Ensuring only authorized individuals have access to collected personal information
7. Ensuring that retention limits are set for collected data
8. Ensuring that end-users are provided with "Notice and Choice" if data collected is more than what would be expected from the product
9. Ensuring the role based access control/authorization to the collected data/analyzed data is applied
10. Ensuring that the analyzed data is de-identified

Example Attack Scenarios

Scenario #1: Collection of personal data.

Date of birth, home address, phone number, etc.

Scenario #2: Collection of financial and/or health information.

Credit card data and bank account information.

In the cases above, exposure of any of the data examples could lead to identity theft or compromise of accounts.

6.Insecure cloud Interface

Threat Agent: Consider anyone who has access to the internet.

Attack Vectors: Attacker uses multiple vectors such as insufficient authentication, lack of transport encryption and account enumeration to access data or controls via the cloud website. Attack will most likely come from the internet.

Security Weakness: An insecure cloud interface is present when easy to guess credentials are used or account enumeration is possible. Insecure cloud interfaces are easy to discover by simply reviewing the connection to the cloud interface and identifying if SSL is in use or by using the password reset mechanism to identify valid accounts which can lead to account enumeration.

Technical Impacts: An insecure cloud interface could lead to compromise of user data and control over the device.

Business Impacts: Consider the business impact of an insecure cloud interface. Data could be stolen or modified and control over devices assumed. Could your customers be harmed? Could your brand be harmed?

Is My Cloud Interface Secure?

Checking for an Insecure Cloud Interface includes:

- Determining if the default username and password can be changed during initial product setup
- Determining if a specific user account is locked out after 3 - 5 failed login attempts
- Determining if valid accounts can be identified using password recovery mechanisms or new user pages
- Reviewing the interface for issues such as cross-site scripting, cross-site request forgery and sql injection.
- Reviewing all cloud interfaces for vulnerabilities (API interfaces and cloud-based web interfaces)

How Do I Secure My Cloud Interface?

A secure cloud interface requires:

1. Default passwords and ideally default usernames to be changed during initial setup
2. Ensuring user accounts cannot be enumerated using functionality such as password reset mechanisms
3. Ensuring account lockout after 3- 5 failed login attempts
4. Ensuring the cloud-based web interface is not susceptible to XSS, SQLi or CSRF
5. Ensuring credentials are not exposed over the internet
6. Implement two factor authentication if possible
7. Detect or block the abnormal reqests/attempts

Example Attack Scenarios

Scenario #1: Password reset indicates whether account is valid.

```
Password Reset "That account does not exist."
```

Scenario #2: Username and password are poorly protected when transmitted over the network.

```
Authorization: Basic S2ZjSDFzYkF4ZzoxMjM0NTY3
```

In the cases above, the attacker is able to either determine a valid user account or is able to capture the credentials as they cross the network and decode them since the credentials are only protected using Base64 Encoding.

7.Insecure Mobile Interface

Threat Agent: Consider anyone who has access to the mobile application.

Attack Vectors: Attacker uses multiple vectors such as insufficient authentication, lack of transport encryption and account enumeration to access data or controls via the mobile interface.

Security Weakness: An insecure mobile interface is present when easy to guess credentials are used or account enumeration is possible. Insecure mobile interfaces are easy to discover by simply reviewing the connection to the wireless networks and identifying if SSL is in use or by using the password reset mechanism to identify valid accounts which can lead to account enumeration.

Technical Impacts: An insecure mobile interface could lead to compromise of user data and control over the device.

Business Impacts: Consider the business impact of an insecure mobile interface. Data could be stolen or modified and control over devices assumed. Could your customers be harmed? Could your brand be harmed?

Is My Mobile Interface Secure?

Checking for an Insecure Mobile Interface includes:

- Determining if the default username and password can be changed during initial product setup
- Determining if a specific user account is locked out after 3 - 5 failed login attempts
- Determining if valid accounts can be identified using password recovery mechanisms or new user pages
- Reviewing whether credentials are exposed while connected to wireless networks
- Reviewing whether two factor authentication options are available

How Do I Secure My Mobile Interface?

A secure mobile interface requires:

1. Default passwords and ideally default usernames to be changed during initial setup
2. Ensuring user accounts can not be enumerated using functionality such as password reset mechanisms
3. Ensuring account lockout after an 3 - 5 failed login attempts
4. Ensuring credentials are not exposed while connected to wireless networks
5. Implementing two factor authentication if possible
6. Apply mobile app obfuscation techinque
7. Implement mbile app anti-tempering mechanism
8. Ensuring the mobile app's memory hacking is possible
9. Restrict the mobile app's execution on tempered OS environment

Example Attack Scenarios

Scenario #1: Password reset indicates whether account exist or not.

```
Password Reset "That account does not exist."
```

Scenario #2: Username and password are poorly protected when transmitted over the network.

```
Authorization: Basic S2ZjSDFzYkF4ZzoxMjM0NTY3
```

In the cases above, the attacker is able to either determine a valid user account or is able to capture the credentials as they cross the network and decode them since the credentials are only protected using Base64 Encoding.

8.Insufficient Security Configurability

Threat Agent: Consider anyone who has access to the device.

Attack Vectors: Attacker uses the lack of granular permissions to access data or controls on the device. The attacker could also us the lack of encryption options and lack of password options to perform other attacks which lead to compromise of the device and/or data. Attack could potentially come from any user of the device whether intentional or accidental.

Security Weakness: Insufficient security configurability is present when users of the device have limited or no ability to alter its security controls. Insufficient security configurability is apparent when the web interface of the device has no options for creating granular user permissions or for example, forcing the use of strong passwords. Manual review of the web interface and its available options will reveal these deficiencies.

Technical Impacts: Insufficient security configurability could lead to compromise of the device whether intentional or accidental and/or data loss.

Business Impacts: Consider the business impact if data can be stolen or modified and control over the device assumed. Could your customers be harmed?

Is My Security Configurability Sufficient?

Checking for Insufficient Security Configurability includes:

- Reviewing the administrative interface of the device for options to strengthen security such as forcing the creation of strong passwords
- Reviewing the administrative interface for the ability to separate admin users from normal users
- Reviewing the administrative interface for encryption options
- Reviewing the administrative interface for options to enable secure logging of various security events
- Reviewing the administrative interface for options to enable alerts and notifications to the end user for security events

How Do I Improve My Security Configurability?

Sufficient security configurability requires:

1. Ensuring the ability to separate normal users from administrative users

2. Ensuring the ability to encrypt data at rest or in transit
3. Ensuring the ability to force strong password policies
4. Ensuring the ability to enable logging of security events
5. Ensuring the ability to notify end users of security events

Example Attack Scenarios

Scenario #1: No ability to enforce strong password policies.

Admins and users are allowed to create passwords for their accounts.

Scenario #2: No ability to enable encryption of data at rest.

Password or other sensitive data stored on the device may not be encrypted.

In the cases above, the attacker is able to use the lack of these controls to get access to user accounts with weak passwords or access data at rest which has protection.

9.Insecure Software/Firmware

Threat Agent: Consider anyone who has access to the device and/or the network the device resides on. Also consider anyone who could gain access to the update server

Attack Vectors: Attacker uses multiple vectors such as capturing update files via unencrypted connections, the update file itself is not encrypted or they are able to perform their own malicious update via DNS hijacking. Depending on method of update and device configuration, attack could come from the local network or the internet.

Security Weakness: The lack of ability for a device to be updated presents a security weakness on its own. Devices should have the ability to be updated when vulnerabilities are discovered and software/firmware updates can be insecure when the updated files themselves and the network connection they are delivered on are not protected. Software/Firmware can also be insecure if they contain hardcoded sensitive data such as credentials. Security issues with software/firmware are relatively easy to discover by simply inspecting the network traffic during the update to check for encryption or using a hex editor to inspect the update file itself for interesting information.

Technical Impacts: Insecure software/firmware could lead to compromise of user data, control over the device and attacks against other devices.

Business Impacts: Consider the business impact if data can be stolen or modified and devices taken control of for the purpose of attacking other devices. Could your customers be harmed? Could other users be harmed?

Is My Software/Firmware Secure?

- Note - It is very important that devices first and foremost have the ability to update and perform updates regularly.

Checking for insecure software/firmware updates include:

- Reviewing the update file itself for exposure of sensitive information in human readable format by someone using a hex edit tool
- Reviewing the production file update for proper encryption using accepted algorithms
- Reviewing the production file update to ensure it is properly signed
- Reviewing the communication method used to transmit the update
- Reviewing the cloud update server to ensure transport encryption methods are up to date and properly configured and that the server itself is not vulnerable
- Reviewing the device for proper validation of signed update files

How Do I Secure My Software/Firmware?

Securing software/firmware require:

1. Ensuring the device has the ability to update (very important, need secure update mechanism)
2. Ensuring the update file is encrypted using accepted encryption methods
3. Ensuring the update file is transmitted via an encrypted connection
4. Ensuring the update file does not expose sensitive data
5. Ensuring the update is signed and verified before allowing the update to be uploaded and applied
6. Ensuring the update server is secure
7. Implement the secure boot if possible (chain of trust)

Example Attack Scenarios

Scenario #1: Update file is transmitted via HTTP.

http://www.xyz.com/update.bin

Scenario #2: Update file is unencrypted and human readable data can be viewed.

�v�ñ]��Ü��Qw�û]��˘3DP�Ö�∂]��˘3DPadmin.htmadvanced.htmalarms.htm

In the cases above, the attacker is able to either capture the update file or capture the file and view its contents.

10.Poor Physical Security

Threat Agent: Consider anyone who has physical access to the device.

Attack Vectors: Attacker uses vectors such as USB ports, SD cards or other storage means to access the Operating System and potentially any data stored on the device.

Security Weakness: Physical security weaknesses are present when an attacker can disassemble a device to easily access the storage medium and any data stored on that medium. Weaknesses are also present when USB ports or other external ports can be used to access the device using features intended for configuration or maintenance.

Technical Impacts: Insufficient physical security could lead to compromise of the device itself and any data stored on that device.

Bussiness Impacts: Data could be stolen or modified and the device taken control of for purposes other than what was originally intended. Could your customers be harmed? Could your brand be harmed?

Is My Physical Security Sufficient?

Checking for Poor Physical Security includes:

- Reviewing how easily a device can be disassembled and data storage mediums accessed or removed
- Reviewing the use of external ports such as USB to determine if data can be accessed on the device without disassembling the device.
- Reviewing the number of physical external ports to determine if all are required for proper device function
- Reviewing the administrative interface to determine if external ports such as USB can be deactivated
- Reviewing the administrative interface to determine if administrative capabilities can be limited to local access only

How Do I Physically Secure My Device?

Adequate physical security requires:

1. Ensuring data storage medium cannot be easily removed.
2. Ensuring stored data is encrypted at rest.
3. Ensuring USB ports or other external ports cannot be used to maliciously access the device.

4. Ensuring device cannot be easily disassembled.
5. Ensuring only required external ports such as USB are required for the product to funtion
6. Ensuring the product has the ability to limit administrative capabilities

Example Attack Scenarios

Scenario #1: The device can be easily disassembled and storage medium is an unencrypted SD card.

SD card can be removed and inserted into a card reader to be modified or copied.

Scenario #2: USB ports are present on the device.

Custom software could be written to take advantage of features such as updating via the USB port to modify the original device software.

In both cases, an attacker is able to access the original device software and make modifications or simply copy specific target data.

3.2 PRIVACY ISSUES

The IoT provides organizations with powerful tools for collecting and analyzing data. This data comes in many forms, and in many cases with the IoT, there is residual data that is either collected or can be assembled through careful analysis. As organizations begin to adopt the IoT we will see the placement of sensors, video cameras, and other hardware aimed at collecting information. These IoT components will be deployed pervasively in public spaces as well as private homes, and in some cases even worn by individuals. Many IoT components will include the use of Global Positioning System (GPS) trackers that can provide location-tracking of individuals or those individuals' assets (e.g., cars/telephones). Another aspect of the IoT is that many IoT systems will overlap in regards to the types of data that is collected. As such, the potential to expose sensitive information in aggregate is raised, even if the two collection systems are operated by entirely different entities. In these instances, enterprising marketers or malicious attackers can make use of this aggregate data to meet their objectives, without the knowledge of the individuals being tracked.

One of the unique challenges related to privacy in the IoT is that there will soon be an ability to overwhelm society with data collection devices and sensors. These devices will sometimes be used maliciously and other times may inadvertently capture information about individuals that have not consented to being tracked. From a system-owner perspective it will be important to understand what actions are allowable on the data that is collected inadvertently from individuals. IoT sensors will also be used in ways that enhance a customer experience however. In these instances, the customer will be provided notification that they are interacting with some IoT system.
It must be considered exactly what data persists about each user, and the impact that it stands to have on compliance and privacy regulations. The same applies to

compliance with industry standards such as PCI, which mandates that PII be encrypted both at rest and in transit. In addition to verifying that all sensitive information is protected sufficiently, it is also important to consider risks related to the supply chain. If components that make up your IoT system are compromised in the supply chain, the risk of exposure of sensitive information is high. Another consideration is related to who has access to stored privacy data. This data will likely be provided to third parties and access to any sensitive information should be logged for auditing purposes and checked for compliance against policies.
Given the complexity of the IoT privacy landscape, it is important for any organization offering IoT-based capabilities to expend appropriate resources to ensure the safeguarding of stakeholder sensitive information. When architecting an IoT system, following Privacy-by-Design principles will allow for the integration of appropriate privacy safeguards within the system. These principles can be followed while designing the implementation of the various components that make up an IoT System for any particular organization. The European Union (EU) Article 29 Data Protection Working Party released guidance in September 2014 stating that all IoT stakeholders should adopt these principles to implementations within any region of the world. The following sections provide an IoT-specific view of these principles that organizations can use to bolster their privacy programs to support IoT deployments.

3.2.1 Privacy-by-Design Principles

Users of IoT systems should be made aware of all of the data collected from or about them, and should be given the opportunity to opt out of data collection practices at a granular level. Recognizing the concerns that many of the IoT devices may not have proper user interface, companies should find suitable methods to provide the choice and notice to consumers.

Proactive not Reactive; Preventive not Remedial

Within the context of an IoT System, it is important to consider the potential privacy ramifications to all stakeholders prior to putting the system into an operational state. At the beginning, analysis will focus on data types collected to understand which are sensitive and what regulations apply to each data type. Next, more in-depth analysis should be undertaken to understand the indirect privacy ramifications of the various IoT component operations. As an example, when dealing with applications that track connected vehicles, it would be important to understand whether the tracking would expose driving patterns that, although anonymized, could be traced back to an individual or group when combined with data collected by other systems. Another case in point regards to the collection of data by smart meters that is fed to the utility companies for analysis. If access to this data is not tightly controlled, attackers can deduce when a person is at home exposing opportunity for physical attacks. Looking at privacy of data-in-aggregate vs. privacy of the data collected by a single system will allow for the identification of potentially serious privacy concerns prior to them being exposed or taken advantage of by unscrupulous persons.

Privacy as the default

Organizations that deploy IoT capabilities should take note of this, and ensure that they have built in privacy controls into their systems, on top of the device or application-specific privacy controls provided by any IoT vendor.

Privacy Embedded into Design

Organizations implementing IoT functionality will be faced with first understanding the true privacy concerns of their stakeholders. As such, conducting an analysis to determine the data elements that an IoT system will process is critical. This should ideally be conducted in conjunction with the recommended threat analysis, and early on in the design of the IOT System.

Once a thorough understanding of the potential indirect effects of data collection has been gained, the appropriate safeguards can be designed into the IoT System from the beginning, versus after a privacy concern has been raised or exploited. Also, companies should reevaluate their personal data breach notification program to cover the aspects related to IoT.

Full Functionality — Positive Sum, not Zero-Sum

There is typically a balance between the objectives of functionality and security that must be maintained to ensure that any particular system works correctly, meets business objectives, and is still secure. The same can be said of privacy. In the case of the IoT, it is critically important that trade-offs between functionality, security and privacy be made early on in the design process in order to ensure that all objectives are met equally. Identifying a privacy issue well into the operational life of an IoT system will make the process of retrofitting privacy controls challenging. Adhere to these principles of Privacy-by-Design to identify and implement those trade-offs when the cost of doing so is relatively minor during design of the IoT system.

End-to-End Security — Lifecycle Protection

Within the IoT, data collected will have a long lifespan. It is important to consider the full lifespan of the data collected, both within the collecting organization and within any third parties to which it is provided. Stakeholders should be made aware of when data is provided to third parties, the controls used to secure it, and how and when the data is disposed of.

Lifecycle protection also applies to second-order data (information about people that is inferred or determined based on primary data) as well. For instance, if a sensor in your car collects how far, where, how fast, and other attributes of your driving habits, then someone can infer various things about you, for example, your shopping or working habits, or who you socialize or interact with. The owner of the data (e.g., the car company) may erase your primary data upon sale of your vehicle, but in fact keep all the inferred information (social connection, shopping habits, etc.).

Visibility and Transparency

Stakeholders should be able to easily identify the data collected from them for any particular IoT system, as well as the planned or potential uses for that data. Stakeholders should also be allowed to opt in to data collection, at both a coarse and granular level. As an example, if an application tracks their driving patterns (e.g., for insurance purposes), the user should be able to explicitly authorize the use of their data for that purpose (coarse). The user should also be able to explicitly authorize individual data elements if so desired, for example the storage of driving patterns or history obtained through GPS.

Respect for User Privacy

Maintaining the privacy of stakeholder information will eventually become a discriminating factor for companies in the era of the IoT. With so many opportunities to mishandle user privacy, the organizations that take the necessary steps to safeguard sensitive information will be viewed far more favorably than the ones that do not. Given this, it is important to instill a culture of privacy awareness within the organization. This could include appointing one or more privacy advocates to evaluate the privacy impacts of any new IoT system being implemented. These people would ideally be given the authority to mandate changes to IoT system designs in the event that privacy concerns are identified.
User privacy is also concerning from an indirect perspective. In the case of some IoT devices, for example smart glasses, the user has consented to privacy clauses, but the observed party most likely has not. Further research must be conducted to understand the impacts and regulations required around these type of scenarios.

Privacy Impact Assessment

If it is found that a device collects, processes or stores Privacy Protected Information (PPI), more stringent controls will be required. These controls should be a mix of policy-based and technical. For example:

- Provisioning of the device may require more administrative approvals
- A review by Internal Audit or Compliance should be conducted to determine if it is viable to have PPI data on IoT devices
- Data stored on the device should be encrypted using sufficiently strong cryptographic algorithms
- Data transmitted from/to the device should be encrypted using sufficiently strong cryptographic algorithms
- Access to the device, both physical and logical, should be restricted to authorized personnel

There are various recommendations on privacy requirements that should be considered based on region, including:

- North America
    - Internet of Things, Privacy and Security in a Connected World, Federal Trade Commission (FTC) Staff Report
- Europe
    - Privacy Recommendations for the IoT, WP29 of the EU (European data protection advisory body)

3.3 THREAT MODELLING

Threat modelling in cyber security science is a structured approach to identifying, quantifying and addressing threats. Threat modelling allows system security staff to communicate the potential damage of security flaws and prioritize remediation efforts.

The threat modelling covers the assets, which refers to what data and equipment should be secured, the threats which refers to what an attacked can do to the system

and the vulnerabilities of the device. Below we will approach a threat modelling for IOT devices.

STEP 1 Identify Assets

This is for cataloguing the various components of the IoT System that will be deployed. Consider not only the IoT devices but also the data stores and applications that the devices communicate and the users that interact with the system.

Step 2: Create a System/ Architecture Overview

This step provides a solid foundation for understanding not only the expected functionality of the IoT System, but also how an attacker could misuse the system. Begin with the process of documenting expected functionality and then spend time to consider and document misuse cases for the system. It is also important to create an architectural diagram that details the new IoT System and how the system interfaces with other enterprise computing resources and security systems. This diagram can also serve as the starting point for identifying trust boundaries, authentication and authorization mechanisms as well as logging compos.
The creation of system architecture is aided through use case analysis. The following example use cases from the healthcare sector can provide insight into security considerations for IoT implementations.

1. A person wears some type of monitor that reports through the cloud to his/her physician
    a. Under extreme circumstances, would first responders be automatically dispatched?
    b. Would a new pharmacy prescription be automatically generated (by some rule), or alternatively would the prescription information be routed to several pharmacies that would compete for the purchase?
    c. Would an appointment be auto-scheduled?
    d. Would health records be updated?
    e. If medical response is dispatched is data transferred to an ambulance?

2. An implanted device receives a command
    a. Does the device use PKI? If so, can the device confirm revocation status of the sender?
    b. Can the device validate the message?
    c. Can the device create a secure link or session with the sender?
    d. Can the device request confirmation?

3. A physician establishes a communication session with a smart home/home monitor
    a. Is the communication channel secured with PKI?
    b. Are PII and medical data transferred securely?
    c. Does the physician issue commands to devices? If so, is there integrity checking and nonrepudiation through logging?

4. A hospital transfers a patient's record or diagnosis to a computer or PDA
    a. Can the patient interact with hospital services, such as scheduling another appointment?

b. Can the patient confirm the authenticity of the message?

c. Can the patient effectively remove the message?

5. A patient's blood donation is handled by an online analyzer

a. Is the tracking number for the donor protected locally or centrally?

b. Will the patient be notified directly of any finding?

c. If the patient has an STD which agencies will be notified?

d. What are the trust mechanisms?

e. Will the blood packet be handled by a robot?

f. Will the patient's pharmacy or doctor be messaged on any particular finding?

g. Will a maintenance center be messaged about the state of the analyzer?

7. In an emergency, multiple first responders are dispatched

a. Is medical data transferred securely to the correct ambulance?

b. Can responders communicate patient data securely? Is it through point-to-point or central routing?

c. Is security, trust and privacy managed by multiple trust chains?

8. A pharmaceutical company issues an alert regarding drug infusion pumps

a. Is the pharmaceutical company's message trusted by pharmacies?

b. Does the alert impact a patient's dispensing device?

c. Does a doctor issue controls to the dispensing device?

d. Does the infusion pump have closed loop communications to the controller/monitor?

9. A doctor performs tele surgery using a robot

a. Is the communication channel trusted and secure?

b. Is the robot's distinguished name trusted with the console?

c. Does the communication depend on DNS?

d. What is the strength of the algorithms and key lengths use by the IP VPN?

e. What is the trust chain and CRL management for the entire topology?

f. Are backup communications channels trusted at the same level as the primary?

g. Are pharmaceutical providers and records keeping updated in real time?

10. A government agency issues a health alert that affects implanted devices

a. In what order are stakeholders notified? (doctors, pharmacies, manufacturers, system administrators, etc.)

b. Is the message authenticated and verified?

c. If a device is recalled, what databases need to be updated?

d. Is the inventory managed to ensure that all devices are properly administered?

11. An implanted or wearable device needs updating

a. Is the update remotely managed?

b. Is there two-way trust between the device and the central server?

c. Is the channel secure and trusted?

d. Is the inventory managed to ensure that all devices are properly managed?

e. Are stakeholders notified if procedures or instructions change?

f. Is the pharmacy notified if drugs are involved?

12. The controlling physician for a specific device is replaced by another physician

a. Are credentials managed centrally or locally?

b. Is there a two-way trust between the physician and the device?

c. Can the device be updated remotely to assign a new trust?

13. A manufacturer alters its instructions for a remotely controlled medical device

a. Is configuration management properly maintained, so that stakeholders know the version of devices/instructions?

b. Are medical universities included as part of the stakeholders?

c. Is there an authoritative database for configuration management?

14. In a connected vehicle environment an ambulance/first responder vehicle coordinates patient records with a medical provider

a. Are the communications protected with PKI?

b. Is there two-way trust between the ambulance and the medical provider?

c. Are patient records purged after the patient has been dispatched?

d. Is on-board equipment remotely managed?

15. A patient with an implanted device dials 911

a. Is the patient data made available to the dispatcher?

b. Can the dispatcher route data to a remote provider or doctor?

c. Is a two-way trust relationship established?

d. Are patient records automatically updated?

e. Can information be securely communicated with an ambulance?

16. A private cloud is deployed in South America to serve remote medical communities

a. Is infrastructure auditable to verify that security standards are met?

b. Does the system support remotely connected devices?

c. Is there two-way trust with the remote clients?

d. How are stakeholder identities authenticated?

17. Nano biomedical devices are remotely deployed

a. Are two-way trust relationships established with the central facility?

b. Is each component in the topology trusted?

c. Are recovered modules properly protected from sensitive medical information? (physical security)

d. Is the inventory tracked securely?

Once the logical architecture view is complete, in is important to identify and examine the specific technologies that will make up the IoT System. This includes understanding and documenting lower level details regarding the IoT devices, such

as the processor type and operating system. This will provide information needed to understand the specific types of vulnerabilities that may eventually be exposed and define processes for how and how often patches and firmware updates should be applied. Understanding and documenting the protocols that are used by each IoT device will also allow for updates to the architecture, especially if gaps are found in the encryption applied to the data transmitted throughout the system and the organization.

Step 3: Decompose the IoT System

At this stage, the focus is on understanding the life cycle of data as it flows through the system. Developing this understanding will allow for the identification of vulnerable or weak points within the security architecture that must be addressed. Identify and document the entry points for data within the system. In an IoT system, these entry points are typically sensors of some type. Trace the flow of data from the entry points and document the various components that interact with that data throughout the system. Identify high profile targets for attackers — these may be points within the system that aggregate or store data, or it may be high value sensors that require significant protections to maintain the overall integrity of the system. At the end of this activity a good understanding of the attack surface of the new IoT system will be had.

Once you decompose the IoT system, shouldn't the next step be to design an architecture to protect the system? Why not give them a notional protective architecture? This could be where some of the elements of the SdP can be introduced. Based upon Junaid's comments I have included a notional diagram that we can adapt for the IoT environment

1. Don't allow anything to connect to them
2. Authenticate to the gateways
3. Use Authorization to Elevate Trust
4. Mitigate the theft of keys
5. Deny all connections
6. Require authorization to initiate communication
7. Use independent communication port for admins
8. Audit logging
9. Pin communications and updates the Root
10. Use Secure boot
11. Use hardened OS
12. Application Whitelisting
13. File Integrity Monitoring
14. Audit logging
15. Updates Response

Step 4: Identify and Document the Threats

The popular STRIDE model can be applied to IoT System deployments. Use well known vulnerability repositories to better understand the environment, such as MITRE's Common Vulnerabilities and Exposures database. Uncovering the unique threats to any particular IoT instantiation will be guided by these threat types:

Threat Type	Iot Description
Spoofing Identity	Examine the system for threats related to the spoofing of machine identity and the ability for an attacker to overcome automated trust relationships between devices. Carefully examine the authentication protocols employed to set up secure communications between various devices (M2M) and between devices and applications that make use of data provided by these devices. Examine the process for provisioning of identities to each IoT device and ensure that there are proper procedural controls in place to limit the ability to introduce a rogue device into the system.
Tampering with Data	Examine the path of data across the entire IoT system. Identify points in the system that provide an opportunity to tamper with the data at points of collection, processing, transport and storage. Carefully examine implementation of authorization mechanisms to ensure that data tampering is effectively dealt with.
Repudiation	Examine the IoT system design for nodes within the system that are critical data providers. These are likely sets of sensors that provide various data for analysis. In the case of the IoT, it is important to be able to trace back data to a source and ensure that it was indeed the expected source that provided that data. Examine the IoT system for weaknesses that would allow an attacker to inject a rogue node that would feed bad data into the system in an attempt to confuse upstream processes or take the system out of an operational state. Ensure that attackers are not able to abuse the intended functionality of IoT systems e.g. illegal operations are disabled or not allowed. State changes and time variations (e.g. disrupting message sequencing) should be taken into account.
Information Disclosure	Examine the path of data across the entire IoT system, including the backend processing systems. Ensure that any device that processes sensitive information has been identified and that proper encryption controls have been implemented to guard against disclosure of that information. Identify data storage nodes within the IoT system and ensure that data-at-rest encryption controls have been applied. Examine the IoT system for instances where IoT devices are vulnerable to being physically stolen and ensure that proper controls, such as key zeroization have been considered.
Denial of Service	Perform an activity that maps each IoT system to

	business goals, in an effort to ensure that appropriate Continuity of Operations (COOP) planning has occurred. Examine the throughput provided for each node in the system and ensure that it is sufficient to withstand relevant DoS attacks. Examine the messaging structures (e.g., data busses), data structures, improper use of variables and APIs used within applicable IoT components and determine if there are vulnerabilities that would allow a rogue node to drown out the transmissions of a legitimate node. Implementers of the IoT should also consider rate limiting APIs to mitigate DoS attacks.
Elevation of Privilege	Examine the administration capabilities provided by the various IoT devices that make up an IoT system. In some cases, there is only one level of authentication, which allows for configuration of device details. In other cases, distinct administrator accounts may be available. Identify instances where there are weaknesses in the ability to segregate administrative functions from user-level functions within IoT nodes. Identify weaknesses in the authentication methods employed by IoT nodes in order to design appropriate authentication controls into the system.
Bypassing Physical Security	Examine the physical protection mechanisms offered by each IoT device and plan mitigations where possible against any identified weaknesses. This is especially true for IoT deployments that are placed in public or remote areas.
Social Engineering Intrusions	Train staff to guard against social engineering attempts and regularly monitor assets for suspicious behavior.
Supply Chain Errors	Understand the various technological components that make up IoT devices and systems and keep track of vulnerabilities related to any of these technology layers.
Network Intrusions	Regularly monitor networks for suspicious behavior.

3.4 ATTACK SURFACE

The Iot Attack Surface Areas Project provides a list of attack surface that should be understood by manufacturers ,developers, security researchers and those looking to deploy or implement IoT technologies within their organizations. Below will be referred the attack surface and the vulnerabilities for the specific attack surface

- Ecosystem Access Control
    - Implicit trust between components
    - Enrollment security
    - Decommissioning system
    - Lost access procedures
- Device Memory
    - Clear text usernames
    - Clear text passwords
    - Third-Party credentials
    - Encryption Keys
- Device Physical Interfaces
    - Firmware extraction
    - User command line interface(CLI)
    - Admin command line interface(CLI)
    - Privilege escalation
    - Reset to insecure state
    - Removal of storage media
    - Tamper resistance
    - Debug port
    - Device ID/Serial number exposure
- Device Web interface
    - SQL injection
    - Cross-site scripting
    - Cross-site Request Forgery
    - Username enumeration
    - Weak passwords
    - Account Lockout
    - Known default credentials
- Device Firmware
    - Hardcoded credentials
    - Sensitive information disclosure
    - Sensitive URL disclosure
    - Encryption Keys
    - Encryption (symmetric ,asymmetric)
    - Firmware version display and/or last update date
    - Backdoor accounts
    - Vulnerable services (web, ssh, tftp, etc.)
    - Security related function API exposure
    - Firmware downgrade
- 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
    - Device Firmware OTA update block
    - Replay attack
    - Lack of payload verification
    - Lack of message integrity check
- 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
- Local Data Storage
    - Unencrypted data
    - Data encrypted with discovered keys
    - Lack of data integrity checks
    - Use of static same enc/dec key
- 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

- Third-party Backend APIs
    - Unencrypted PII sent
    - Encrypted PII sent
    - Device information leaked
    - Location leaked
- Update Mechanism
    - Update sent without encryption
    - Updates not signed
    - Update location writable
    - Update verification
    - Update authentication
    - Malicious update
    - Missing update mechanism
    - No manual update mechanism
- 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
- Vendor Backend APIs
    - Inherent trust of cloud or mobile application
    - Weak authentication
    - Weak access controls
    - Injection attacks
    - Hidden services
- Ecosystem Communication
    - Health checks
    - Heartbeats
    - Ecosystem commands
    - Deprovisioning
    - Pushing updates
- Network Traffic
    - LAN
    - LAN to Internet
    - Short range
    - Non-standard
    - Wireless (WiFi, Z-wave, Zigbee, Bluetooth)

    - Protocol fuzzing
- Authentication/Authorization
    - Authentication/Authorization related values (session key, token, cookie, etc.) disclosure
    - Reusing of session key, token, etc.
    - Device to device authentication
    - Device to mobile Application authentication
    - Device to cloud system authentication
    - Mobile application to cloud system authentication
    - Web application to cloud system authentication
    - Lack of dynamic authentication
- Privacy
    - User data disclosure
    - User/device location disclosure
    - Differential privacy
- Hardware (Sensors)
    - Sensing Environment Manipulation
    - Tampering (Physically)
    - Damaging (Physically)

Attack surfaces will now be combined with the most common vulnerabilities that Internet of things face:

- Username enumeration vulnerability. The attack surfaces for this vulnerability are administrative interface, device web interface, cloud interface and mobile application. With this vulnerability it is given the ability to collect a set of valid usernames by interacting with the authentication mechanism.
- Weak passwords vulnerability. The attack surfaces for this vulnerability are administrative interface, device web interface, cloud interface and mobile application. It is given the ability to set account passwords that are easily be cracked like ‘1234’ or ‘123456’ for example.
- Account lockout vulnerability. The attack surfaces for this vulnerability are administrative interface, device web interface, cloud interface and mobile application. With this vulnerability it is given the ability to continue sending authentication attempts after 3-5 failed login attempts
- Unencrypted devices vulnerability. The attack surface for this vulnerability is the device networks services. Network services are not properly encrypted to prevent eavesdropping by attackers.
- Two factor authentication lack vulnerability. The attack surfaces for this vulnerability are administrative interface, cloud web interface and mobile application. The lack of two-factor authentication mechanisms such as a security token or fingerprint scanner.

- Poorly Implemented Encryption vulnerability. It targets device network services. The encryption is implemented however it is improperly configured or is not being properly updated, e.g. SSL v2.
- Update sent without encryption vulnerability. It targets the update mechanism and the updates are transmitted over the network without using TLS or encrypting the update file itself.
- Update location writable, it targets the update mechanism. The storage location for update files is world writable potentially allowing firmware to be modified and distributed to all users.
- Denial of service vulnerability. The attack surface for this vulnerability is the device network service. The service can be attacked in a way that denies service to that service or the entire device.
- Removal of storage media vulnerability. It targets the device physical interfaces and it gives the ability to physically remove the storage media from the device.
- No manual update mechanism, it targets the update mechanism and there is no ability to manually force an update check for the device.
- Missing update mechanism, it targets the update mechanism and there is no ability to update the device.
- Firmware version display and/or last update date. It targets the device firmware and the current firmware version is not displayed and/or the last update date is not displayed.

3.5 FIRMWARE ANALYSIS

The Iot Attack Surface Areas Project provides a list of attack surface that should be understood by manufacturers, developers, security researchers and those looking to deploy or implement IoT technologies within their organizations. Below will be referred the attack surface and the vulnerabilities for the specific attack surface

- Ecosystem Access Control
    - Implicit trust between components
    - Enrollment security
    - Decommissioning system
    - Lost access procedures
- Device Memory
    - Clear text usernames
    - Clear text passwords
    - Third-Party credentials
    - Encryption Keys
- Device Physical Interfaces
    - Firmware extraction
    - User command line interface(CLI)

    - Admin command line interface(CLI)
    - Privilege escalation
    - Reset to insecure state
    - Removal of storage media
    - Tamper resistance
    - Debug port
    - Device ID/Serial number exposure
- Device Web interface
    - SQL injection
    - Cross-site scripting
    - Cross-site Request Forgery
    - Username enumeration
    - Weak passwords
    - Account Lockout
    - Known default credentials
- Device Firmware
    - Hardcoded credentials
    - Sensitive information disclosure
    - Sensitive URL disclosure
    - Encryption Keys
    - Encryption (symmetric, asymmetric)
    - Firmware version display and/or last update date
    - Backdoor accounts
    - Vulnerable services (web, ssh, tftp, etc.)
    - Security related function API exposure
    - Firmware downgrade

- 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
    - Device Firmware OTA update block

    - Replay attack
    - Lack of payload verification
    - Lack of message integrity check

- 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

- Local Data Storage
    - Unencrypted data
    - Data encrypted with discovered keys
    - Lack of data integrity checks
    - Use of static same enc/dec key

- 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

- Third-party Backend APIs
    - Unencrypted PII sent
    - Encrypted PII sent

    - Device information leaked
    - Location leaked

- Update Mechanism
    - Update sent without encryption
    - Updates not signed
    - Update location writable
    - Update verification
    - Update authentication
    - Malicious update
    - Missing update mechanism
    - No manual update mechanism

- 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

- Vendor Backend APIs
    - Inherent trust of cloud or mobile application
    - Weak authentication
    - Weak access controls
    - Injection attacks
    - Hidden services

- Ecosystem Communication
    - Health checks
    - Heartbeats
    - Ecosystem commands
    - Deprovisioning
    - Pushing updates

- Network Traffic
    - LAN
    - LAN to Internet
    - Short range
    - Non-standard
    - Wireless (WiFi, Z-wave, Zigbee, Bluetooth)
    - Protocol fuzzing

- Authentication/Authorization
    - Authentication/Authorization related values (session key, token, cookie, etc.) disclosure
    - Reusing of session key, token, etc.
    - Device to device authentication
    - Device to mobile Application authentication
    - Device to cloud system authentication
    - Mobile application to cloud system authentication
    - Web application to cloud system authentication
    - Lack of dynamic authentication

- Privacy
    - User data disclosure
    - User/device location disclosure
    - Differential privacy

- Hardware (Sensors)
    - Sensing Environment Manipulation
    - Tampering (Physically)
    - Damaging (Physically)

Attack surfaces will now be combined with the most common vulnerabilities that Internet of things face:

- Username enumeration vulnerability. The attack surfaces for this vulnerability are administrative interface, device web interface, cloud interface and mobile application. With this vulnerability it is given the ability to collect a set of valid usernames by interacting with the authentication mechanism.

- Weak passwords vulnerability. The attack surfaces for this vulnerability are administrative interface, device web interface, cloud interface and mobile application. It is given the ability to set account passwords that are easily be cracked like ‘1234’ or ‘123456’ for example.
- Account lockout vulnerability. The attack surfaces for this vulnerability are administrative interface, device web interface, cloud interface and mobile application. With this vulnerability it is given the ability to continue sending authentication attempts after 3-5 failed login attempts
- Unencrypted devices vulnerability. The attack surface for this vulnerability is the device networks services. Network services are not properly encrypted to prevent eavesdropping by attackers.
- Two factor authentication lack vulnerability. The attack surfaces for this vulnerability are administrative interface, cloud web interface and mobile application. The lack of two-factor authentication mechanisms such as a security token or fingerprint scanner.
- Poorly Implemented Encryption vulnerability. It targets device network services. The encryption is implemented however it is improperly configured or is not being properly updated, e.g. SSL v2.
- Update sent without encryption vulnerability. It targets the update mechanism and the updates are transmitted over the network without using TLS or encrypting the update file itself.
- Update location writable, it targets the update mechanism. The storage location for update files is world writable potentially allowing firmware to be modified and distributed to all users.
- Denial of service vulnerability. The attack surface for this vulnerability is the device network service. The service can be attacked in a way that denies service to that service or the entire device.
- Removal of storage media vulnerability. It targets the device physical interfaces and it gives the ability to physically remove the storage media from the device.
- No manual update mechanism, it targets the update mechanism and there is no ability to manually force an update check for the device.
- Missing update mechanism, it targets the update mechanism and there is no ability to update the device.
- Firmware version display and/or last update date. It targets the device firmware and the current firmware version is not displayed and/or the last update date is not displayed.

CHAPTER 4

4.1 PENETRATION TESTING METHODOLOGY

A penetration test, is an attack on a computer system or a device that looks for security weaknesses, potentially gaining access to the computer's features and data. Penetration tests can be automated with software applications or they can be performed manually. Either way, the process includes gathering information about the target before the test (reconnaissance), identifying possible entry points, attempting to break in (either virtually or for real) and reporting back the findings.

Until now, the internet of things penetration testing is not so famous even the vendors everyday distribute to the market new devices. One testing methodology is from OWASP. This testing guidance is developed for the most common and well known vulnerabilities.

INSECURE WEB INTERFACE

To test this vulnerability, we must assess any web interface to determine if weak passwords are allowed. Assess the account lockout mechanism, assess the web interface for cross side scripting attacks, SQLi and CSRF vulnerabilities and other application vulnerabilities and finally assess the use of HTTPS to protect transmitted information.

LACK OF TRANSPORT ENCRYPTION

To test lack of transport encryption we must assess the solution to determine the use of encrypted communication between the devices and internet. Assess the solution to determine if accepted encryption practices are used and if proprietary protocols are avoided and assess the solution to determine if a firewall option is available.

INSUFFICIENT SECURITY CONFIGURABILITY

To test this vulnerability, we must assess the solution to determine if password security options are available, assess the solution to determine if encryption options (Enabling AES-256 where AES-128 is the default setting) are available and assesse the solution to determine if logging for security events.

POOR PHYSICAL SECURITY

To test poor physical security, we must assess the device to ensure it utilizes a minimal number of physical external ports (e.g. USB ports) on the device and assess the device to determine if it can be accessed via unintended methods such as through ab unnecessary USB port.

INSUFFICIENT AUTHENTICATION/AUTHORIZATION

To solve such a problem, we must assess the solution for the use of strong passwords where authentication is needed, assess the solution for implementation two-factor authentication where possible, assess password recovery mechanisms, assess the solution for the option to require strong passwords, assess the solution for

the option to force password expiration after a specific period and assess the solution for the option to change the default username and password.

INSECURE CLOUD INTERFACE

For insecure cloud interface we must assess the cloud interfaces for security vulnerabilities, assess the cloud-based web interface to ensure It disallows weak passwords, assess the cloud-based web interface to ensure it includes an account lockout mechanism, assess the cloud-based web interface to determine if two-factor authentication is used, assess any cloud interfaces for cross side scripting attacks ,SQLi and CSRF vulnerabilities and other vulnerabilities, assess all cloud interfaces to ensure transport encryption is used and assess the cloud interfaces to determine if the option to require strong password is available.

INSECURE SOFTWARE/FIRMWARE

We must assess the device to ensure it includes update capability and can be updated quickly when vulnerabilities are discovered, assess the device to ensure it uses encrypted update files and that the files are transmitted using encryption and assess the device to ensure it uses signed files and then validates that file before installation.

PRIVACY CONCERNS

We must assess the solution to determine the amount of personal information collected, assess the solution to determine if collected personal data is properly protected using encryption at rest and in transit and assess the solution to determine if ensuring data is de-identified or anonymized.

INSECURE MOBILE INTERFACE

We must assess the mobile interface to ensure it disallows weak passwords, we must assess the mobile interface to ensure it includes an account lockout mechanism, we must assess the mobile interface to determine if it implements two-factor authentication, we must assess the mobile interface to determine if the option to require strong passwords is available, assess the mobile interface to determine if the option to force password expiration after a specific period is available, assess the mobile interface to determine if the option to change the default username and password is available and assess the mobile interface to determine the amount of personal information collected.

INSECURE NETWORK SERVICES

We must assess the solution to ensure network services don’t respond poorly to buffer overflow, fuzzing od denial of service attacks and assess the solution to ensure test ports are not present.

4.2 VULNERABLE IOT DEVICES SEARCH

For our research of vulnerable Internet of Things online devices we will use a search engine called Shodan and google dorks.

Shodan Search Engine

Shodan is a computer search engine designed be web developer John Matherly. It is a search engine for Internet connected devices but it is much different than content search engines like Google, Yahoo or Bing. Typical search engine crawl for data on web pages and then they index this data for searching. Shodan interrogates ports and grabs the resulting banners, then indexes the banners rather than the web content for searching. The resulting banners are textual information that describes a service on a device. The content of the banner varies depending the type of service. Below is an example of an HTTP banner:

HTTP/1.1 200 OK
Server: **nginx/1.1.19**
Date: **Sat**, **03 Oct 2015 06**:09:24 **GMT**
Content-Type: **text/html**; **charset**=**utf-8**
Content-Length: **6466**
Connection: **keep-alive**

Below is another banner for the Siemens S7 industrial control system protocol:

Copyright: Original Siemens Equipment
PLC name: S7_Turbine
Module type: CPU 313C
Unknown (129): Boot Loader A
Module: 6ES7 313-5BG04-0AB0 v.0.3
Basic Firmware: v.3.3.8
Module name: CPU 313C
Serial number **of** module: S Q-D9U083642013
Plant identification:
Basic Hardware: 6ES7 313-5BG04-0AB0 v.0.3

The basic operation of Shodan is searching and this can be done by entering search terms into a text boxlike the one below.

For searching terms Boolean operators can be used to include or exclude query terms.

In addition to the banner, Shodan also grabs meta-data about the device such as its geographic, location, hostname, operating system and more. To view and narrow those terms it is possible for a user to use some basic filters:

- After/before: limit results by date
- Country: filter results by two letter country code
- Hostname: filters results by specified text in the hostname of domain
- Net: filter results by a specific IP range or subnet
- Operation system: search for specific operation systems
- Ports: narrow the search for specific services

In our research for vulnerable IoT devices Shodan will be used within an ethical and “white hat” approach.

Google Dorks

Another common method for finding vulnerable online devices, is the use of Google search engine. Either by using Google’s web interface and building queries or using already released Google dorks, it is possible to find and access vulnerable Internet of Things devices. A Google dork query is a search string that uses advanced search operators to find information that is not readily available on a website.

Google dorking, also known as Google hacking can return information that is difficult to locate through simple search queries. That includes information that is not intended for public viewing but that has not been adequately protected. Google dorking is a passive attack method and can return usernames, passwords, and vulnerabilities.

Below are some advanced search parameters examples:

- intitle, allintitle
- inurl, allinurl
- filetype
- allintext
- site
- link
- inanchor
- daterange
- cache
- info
- related

Examples of valid queries that use advanced operators include these:

- _ intitle:Google. This query will return pages that have the word Google in their title.
- _ intitle:“index of”. This query will return pages that have the phrase index of in their title. Remember from the previous chapter that this query could also be given as intitle:index.of, since the period serves as any character.This technique also makes it easy to supply a phrase without having to type the spaces and the quotation marks around the phrase.
- _ intitle:“index of” private .This query will return pages that have the phrase index of in their title and also have the word private anywhere in the page, including in the URL, the title, the text, and so on. Notice that intitle only applies to the phrase index of and not the word private, since the first unquoted space follows the index of phrase. Google interprets that space as the end of your advanced operator search term and continues processing the rest of the query.

- _ intitle:“index of”“backup files” .This query will return pages that have

the phrase index of in their title and the phrase backup files anywhere in the page, including the URL, the title, the text, and so on. Again, notice that intitle only applies to the phrase index of.

4.3 VULNERABLE IOT DEVICES USE CASE

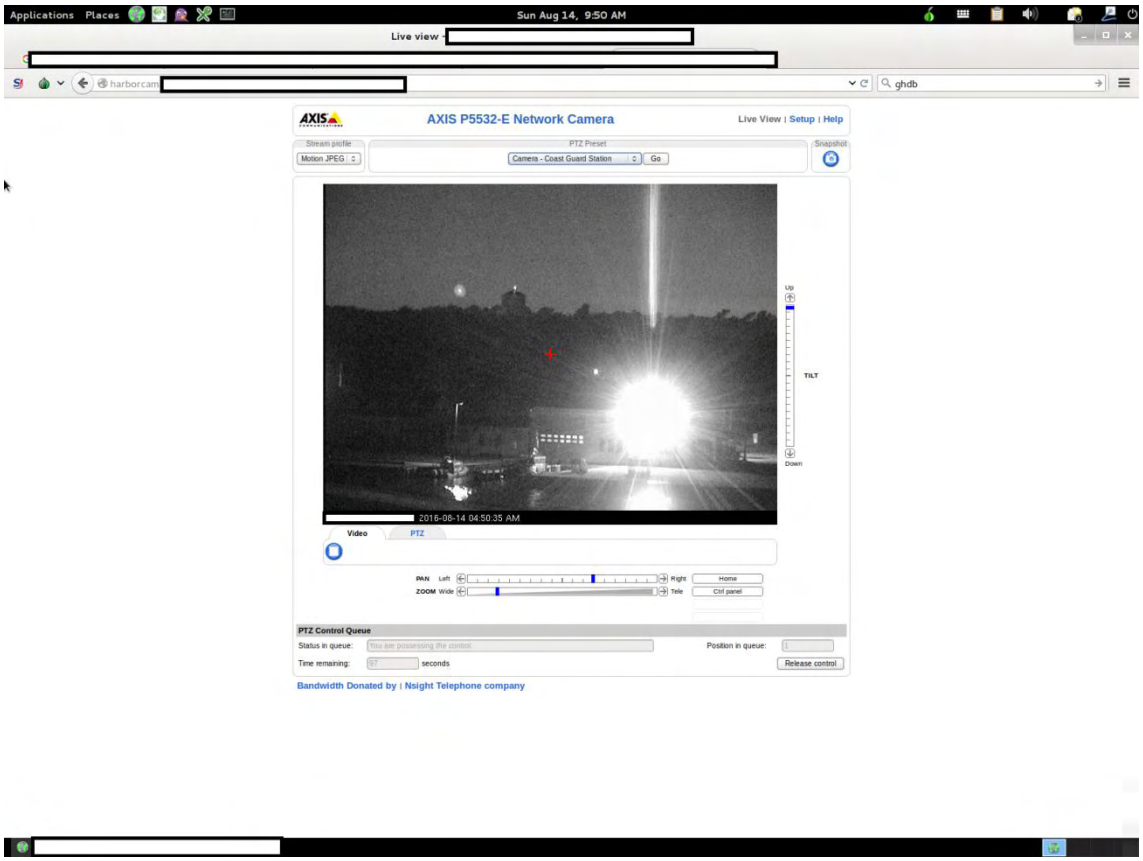
For privacy issues at the following screenshots some of the data has been masked. The operational systems that was used at the research were Kali Linux v2.0, Windows 7 Pro and Parrot Security Linux OS. Below there will be screenshots from our research which they will present the location of the vulnerable IOT devices, the proof of the access that was gained and other relevant information. In the table below are shown the amount of the search results and the Iot device or system that used for the use case

IOT DEVICE	TOTAL VULNERABLE DEVICES
Axis Webcams	918
WebcamXp5	1310
Hp Officejet Printer	513
Ricoh Photocopier	79
Smart Wifi Thermostats	126
Heating System DDC400	125
Heatmiser Smart Thermostat	578
Rapsberry Pi Smart Home	610
Loxone Smart Home	803
Yamaha RX1 Smart Stereo	309
Bticino Legrand Smart Home	1300
Smart Metering	85
CAT Self Drive Dump Trucks	146
GAS Tank Levels	4226
Fibaro Home Center 2	928

Table 1

AXIS WEBCAMS GOOGLE DORK

Our search found 918 results of vulnerable webcams where has gained full access of settings and live view feed.



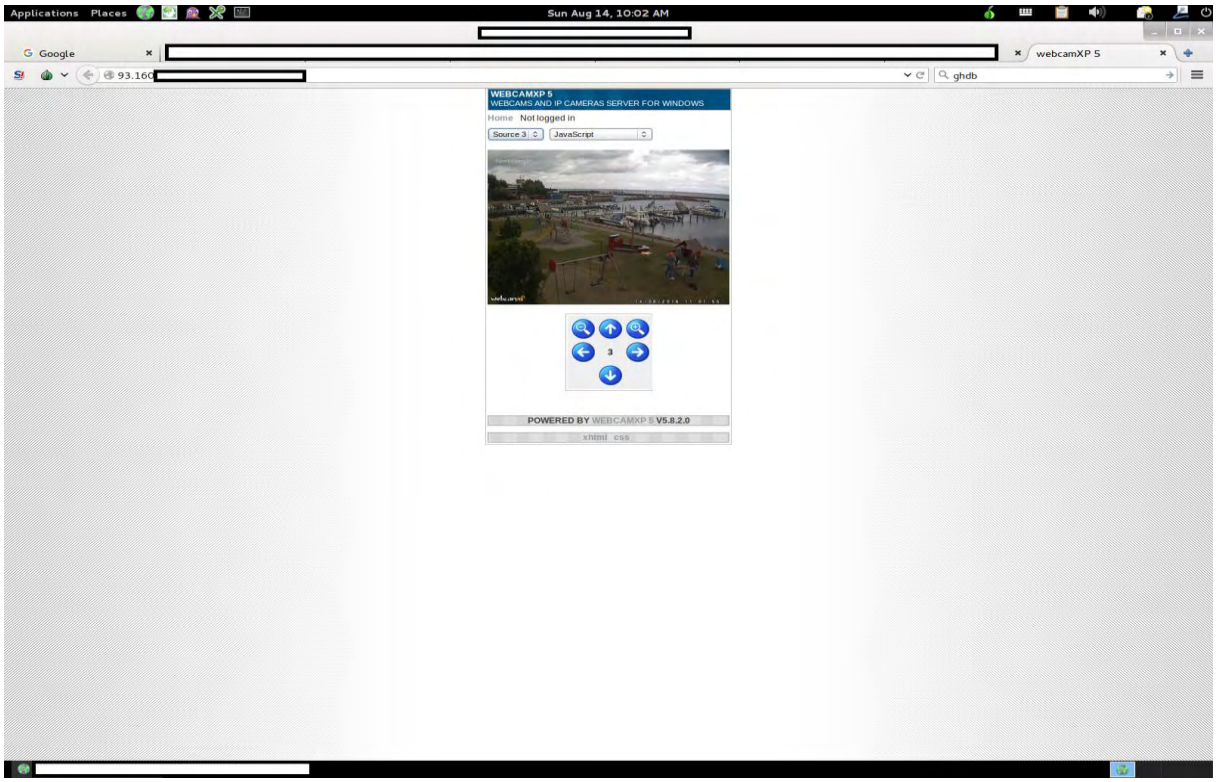
Screenshot 1- Live view and PTZ control access

Screenshot 2-Live view

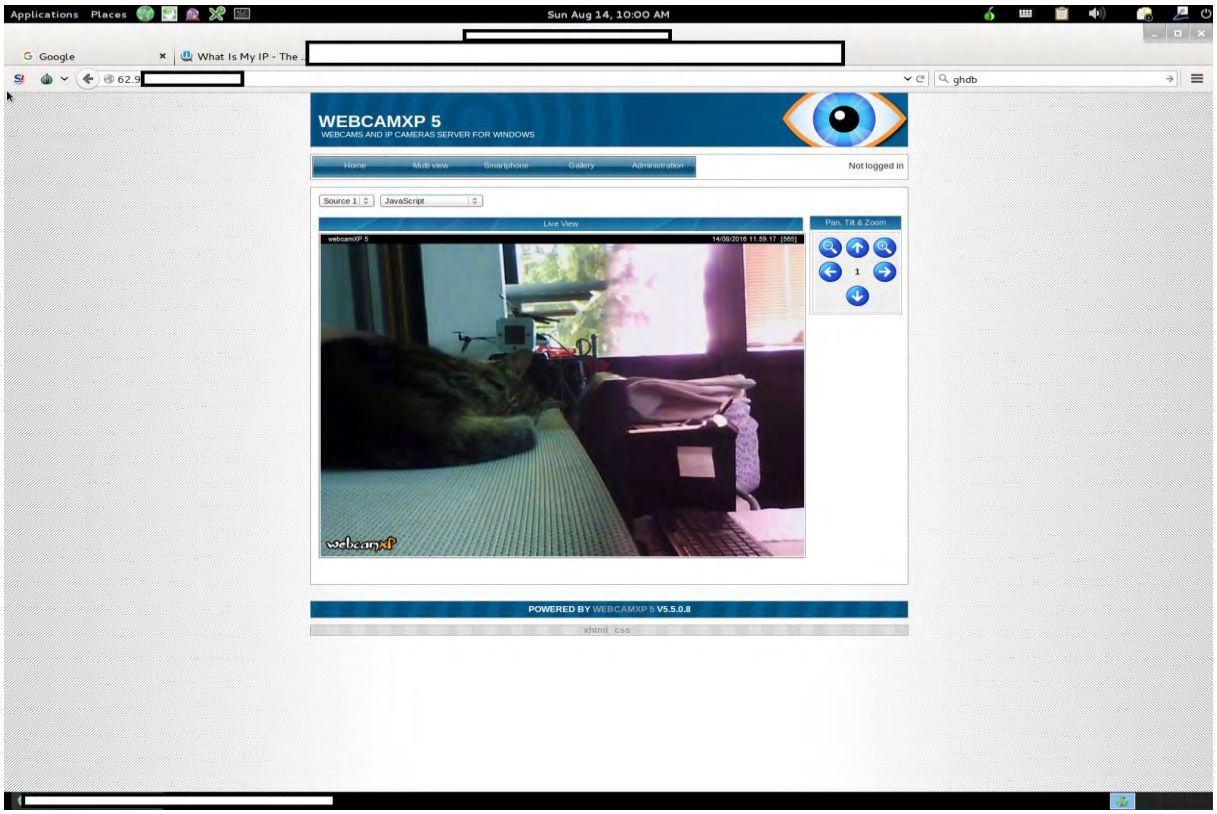
WEBCAMXP5 GOOGLE DORK

Our search found 1310 results of vulnerable webcams. Full access of settings gained and live view feed.

Screenshot 3-Live view and PTZ access

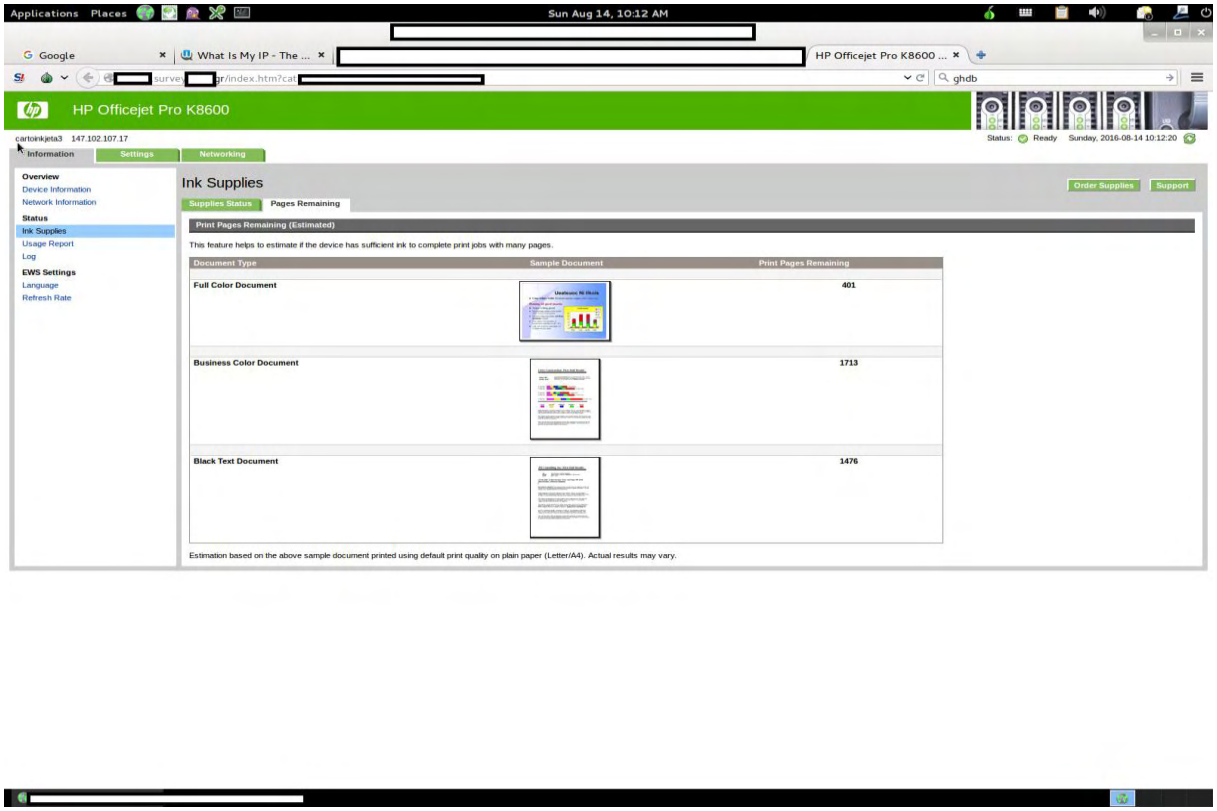


Screenshot 4 - Live view and PTZ access



Screenshot 5- Live view and PTZ access

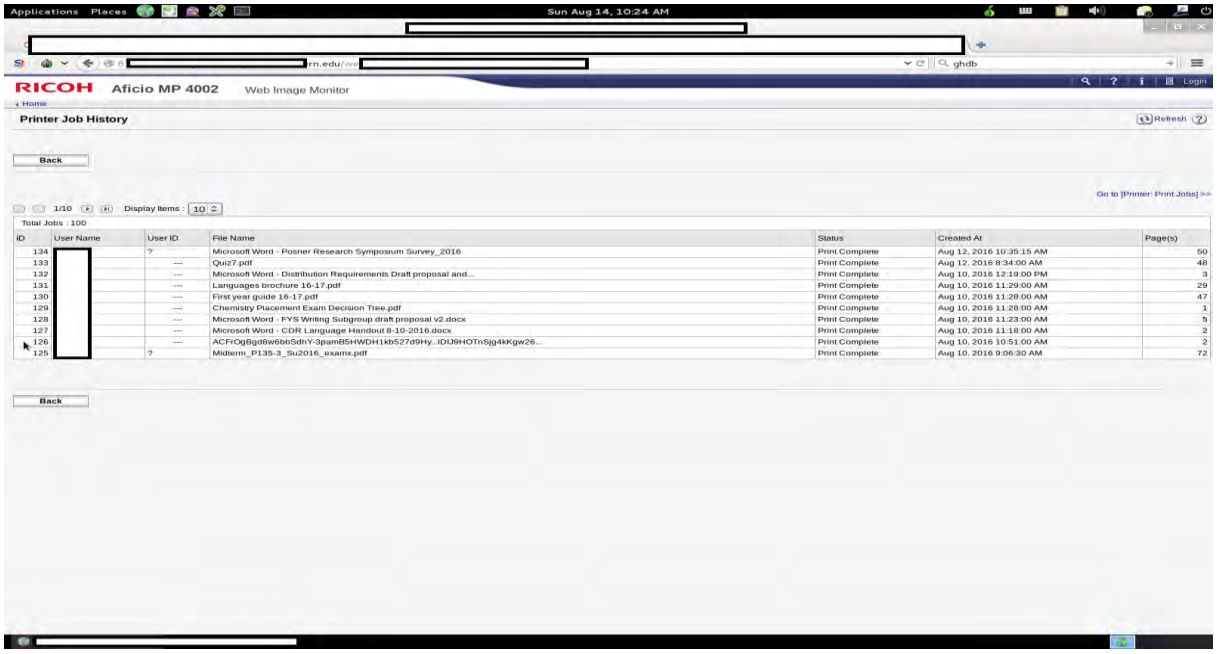
HP OFFICEJET PRINTERS GOOGLE DORK The search found 513 results of vulnerable printers. Some of them located in Greek Universities like the one in the screenshot below.



Screenshot 6 – Access to the printer settings and history

RICOH PHOTOCOPIER GOOGLE DORK

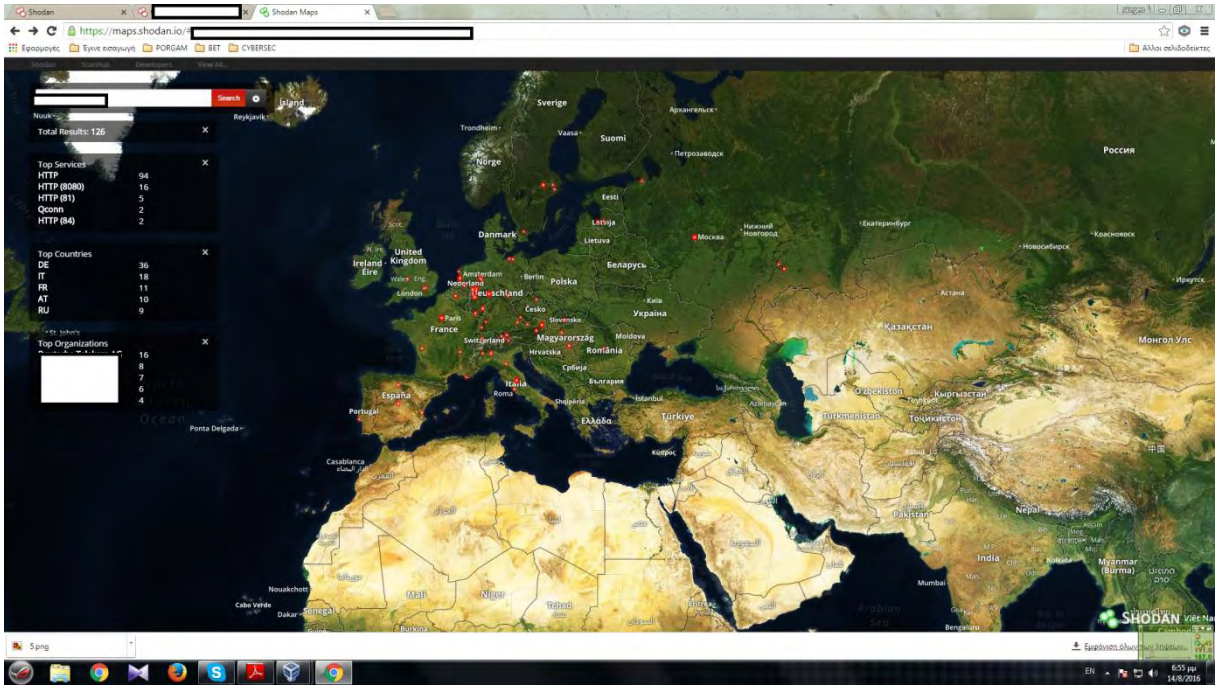
The search returned 79 results which some them belonging to Educational Institutes as the screenshot below.



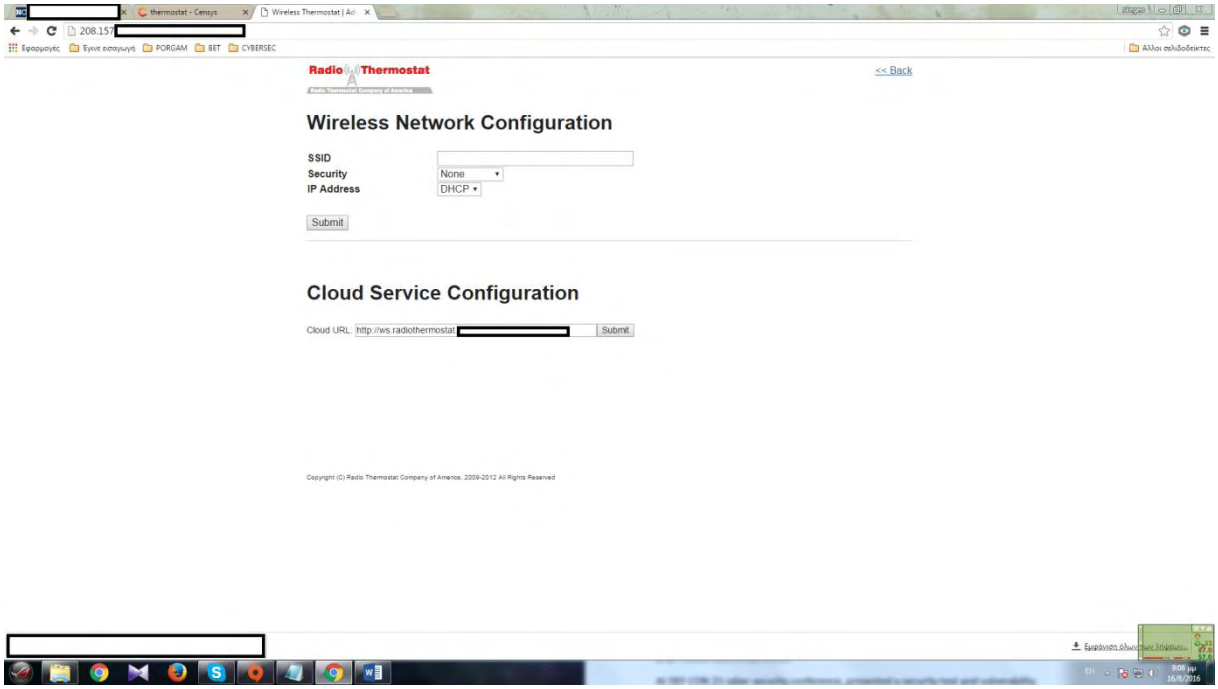
Screenshot 7- Access to the printer settings and history

SMART WIFI THERMOSTATS

The results were 126 mostly from EU. At Germany found 36 devices, at Italy 18 devices, at France 11 devices, at Austria 10 devices and at Russia 9. When access was gained at the interface of the thermostat, it was possible to gain administrator access.



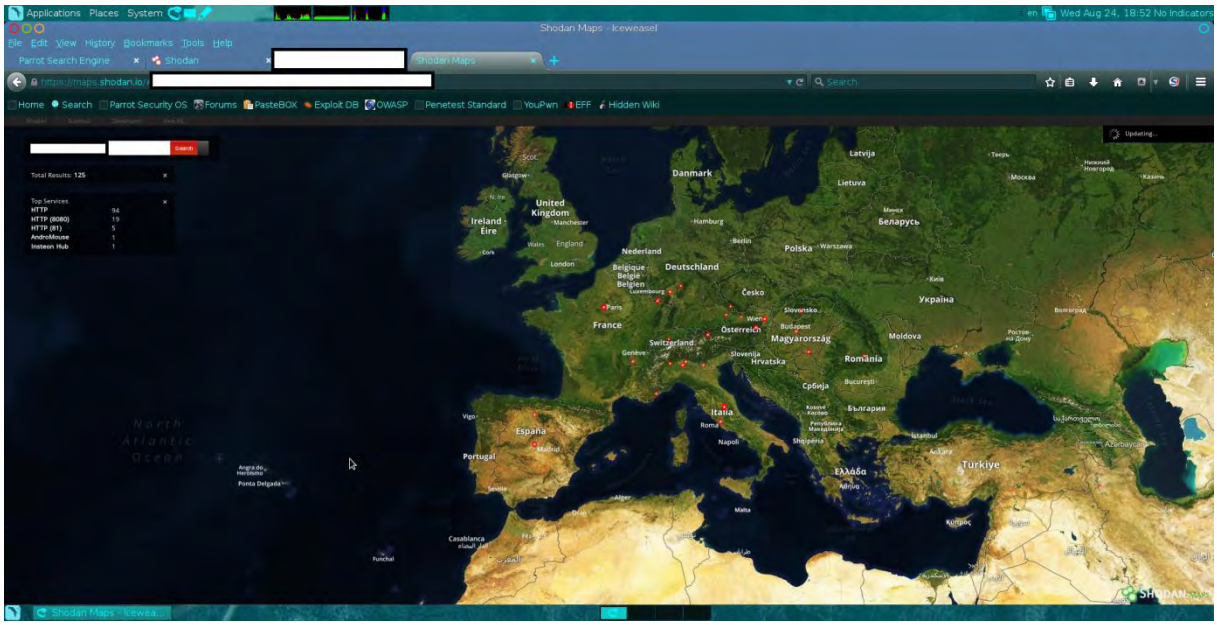
Screenshot 8 – Map results



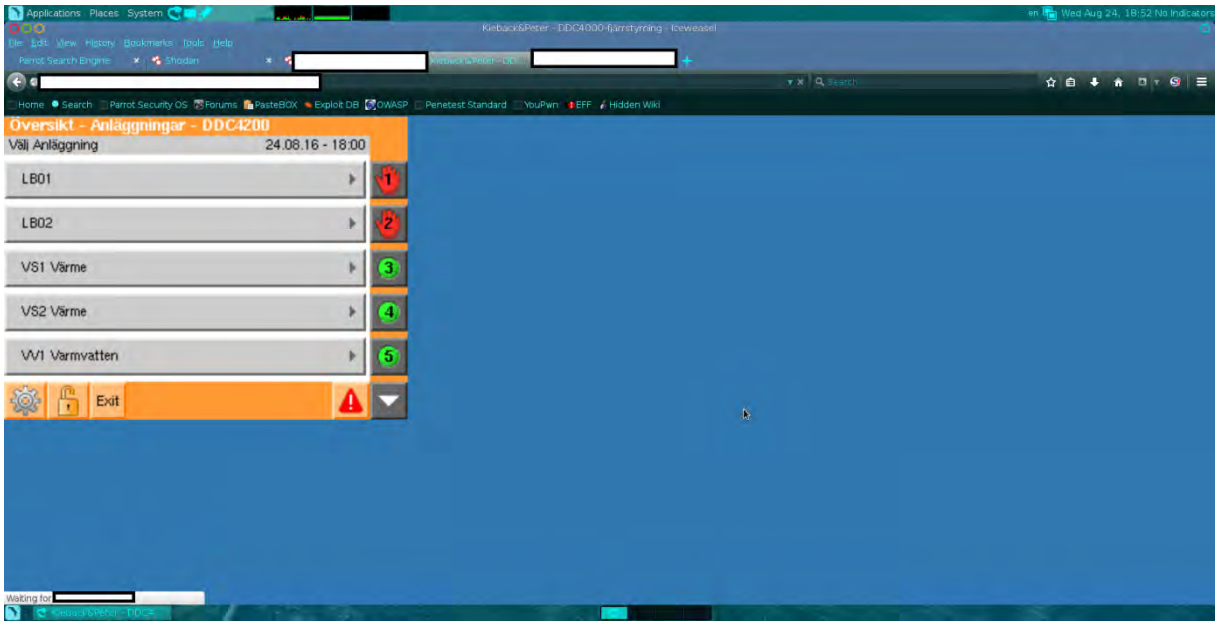
Screenshot 8 – Thermostat's Settings

IOT HEATING SYSTEMS CONTROL PANELS

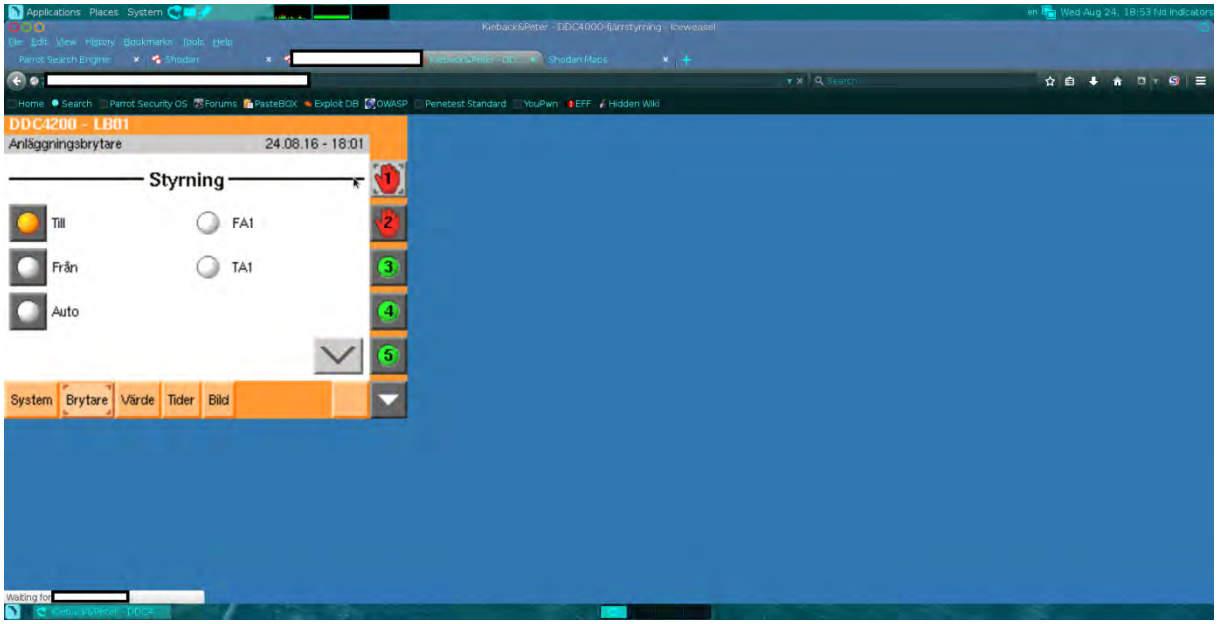
The search returned 125 results all from the EU. It was gained full operationally access and remote access control.



Screenshot 9 – Results Map



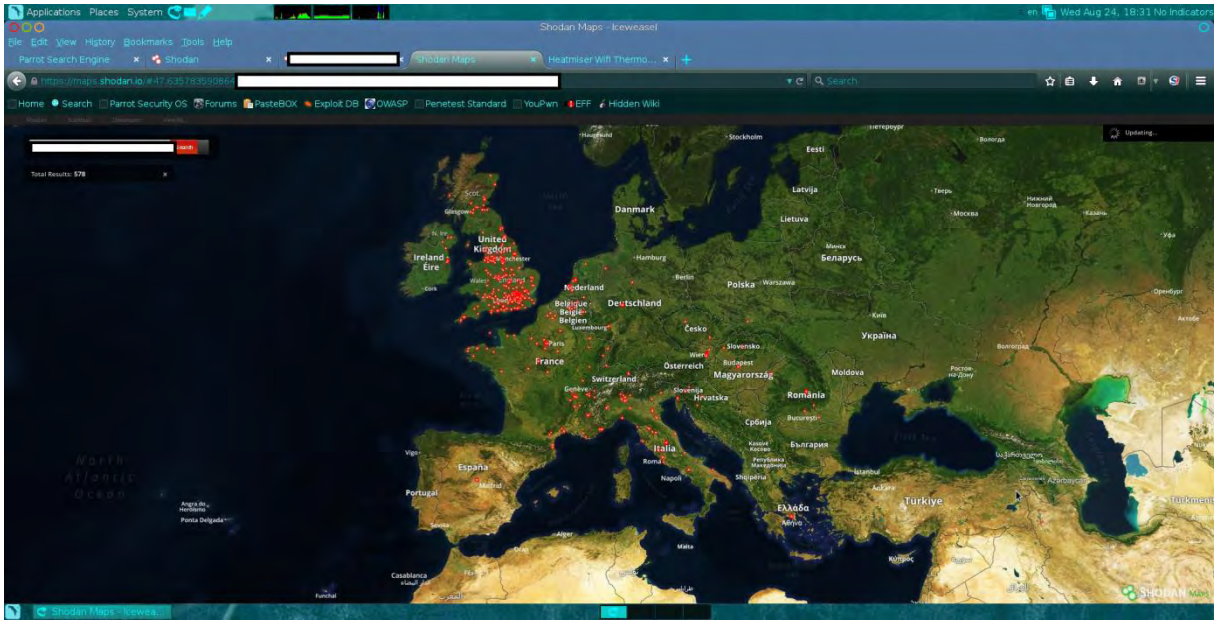
Screenshot 10 – ON/OFF of Heating Systems



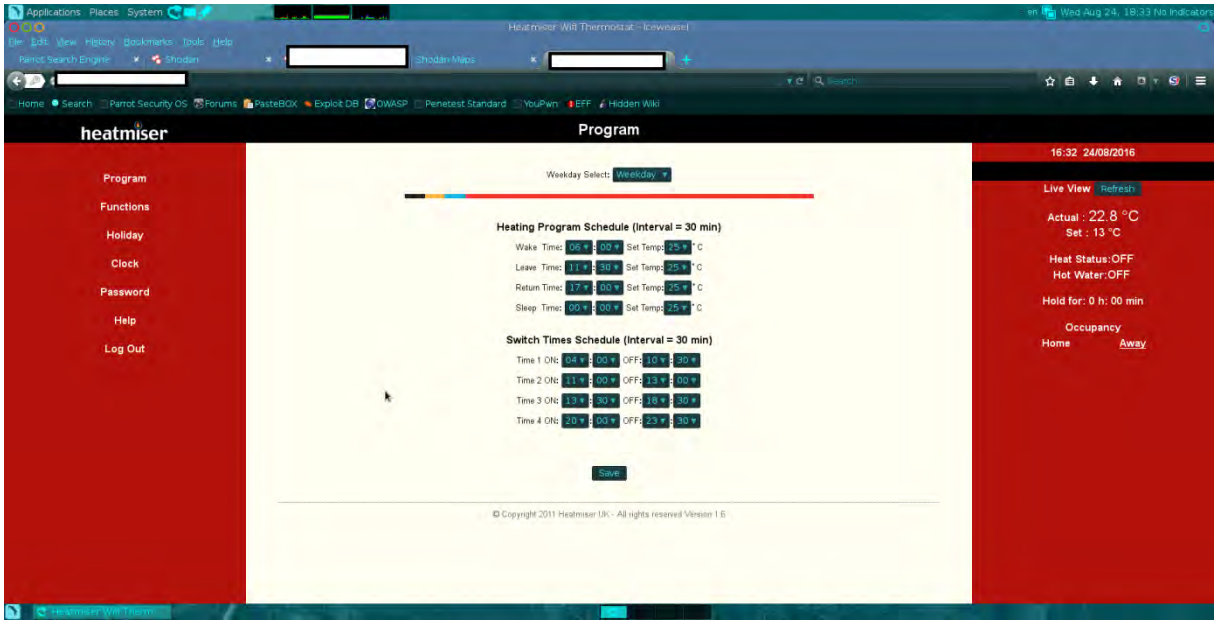
Screenshot 11-System control

HEATMISER SMART THERMOSTATS

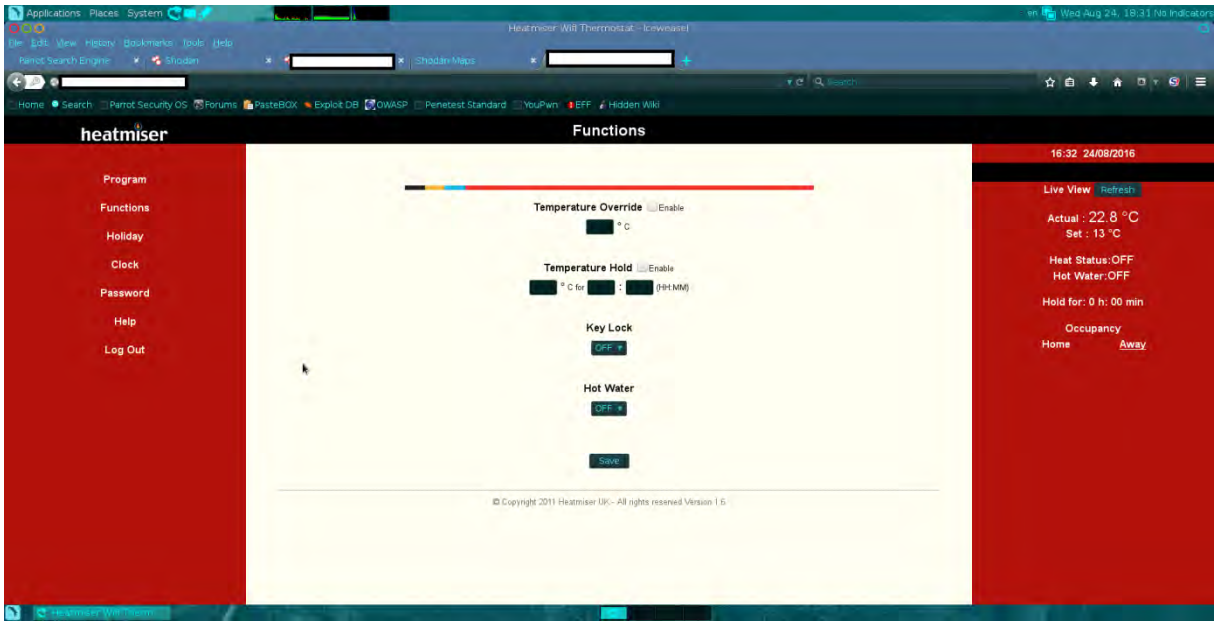
The search results were 578, all from EU and some of them from Greece. Full administrator access was gained.



Screenshot 12 – Map results

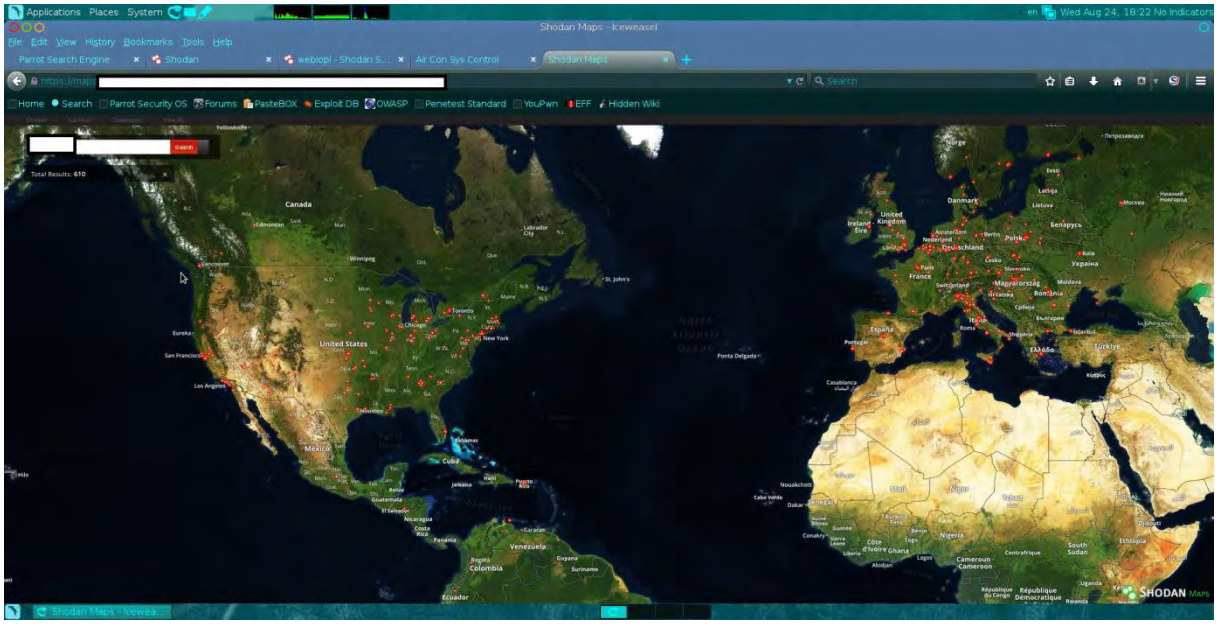


Screenshot 13 – Heating program schedule timer settings

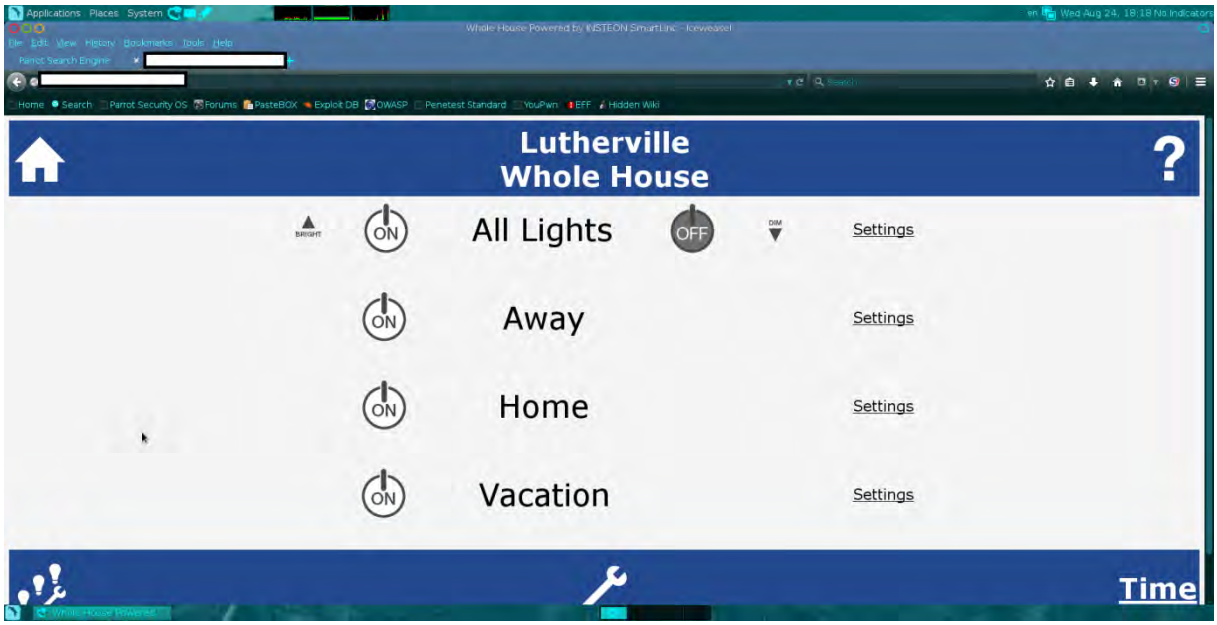


Screenshot 14 – Temperature settings

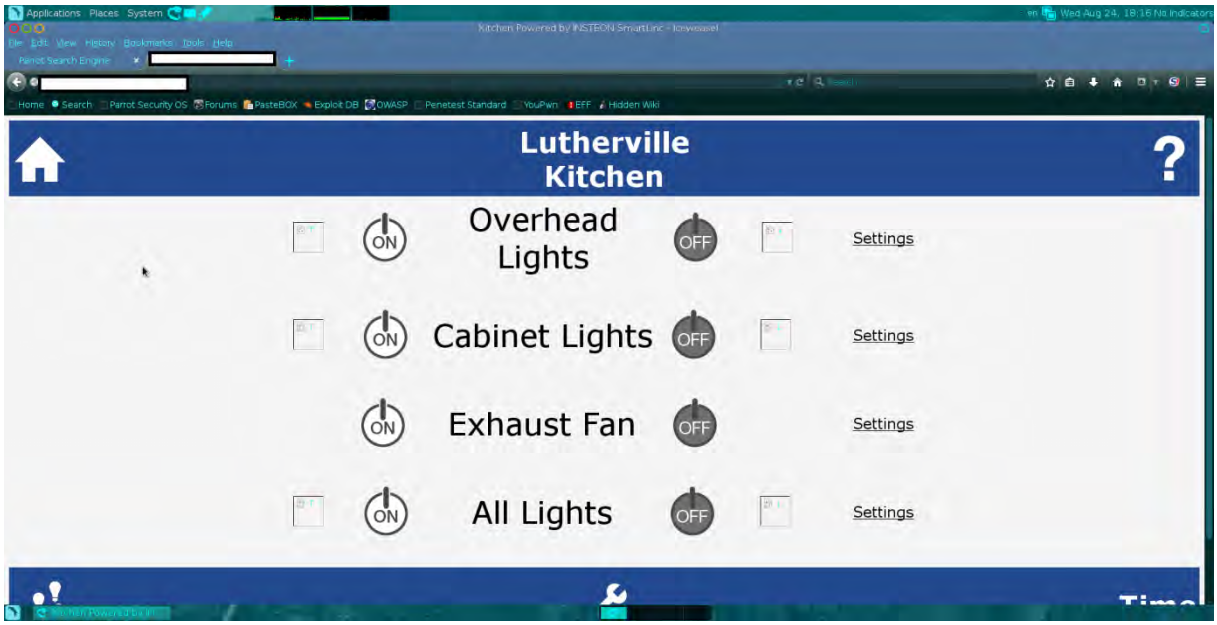
There were 610 results in all over the world and 11 results at Greece. Was gained full administrator access and remote control of the devices that the raspberry pi controlled.



Screenshot 15 - Map results



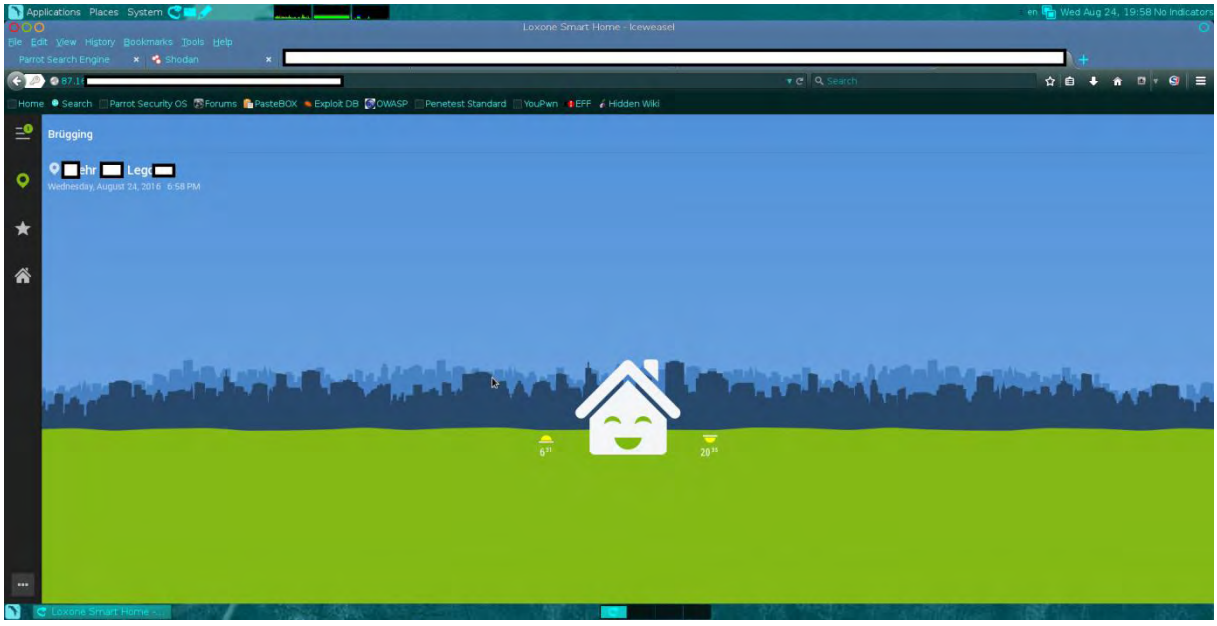
Screenshot 16 - Lighting control



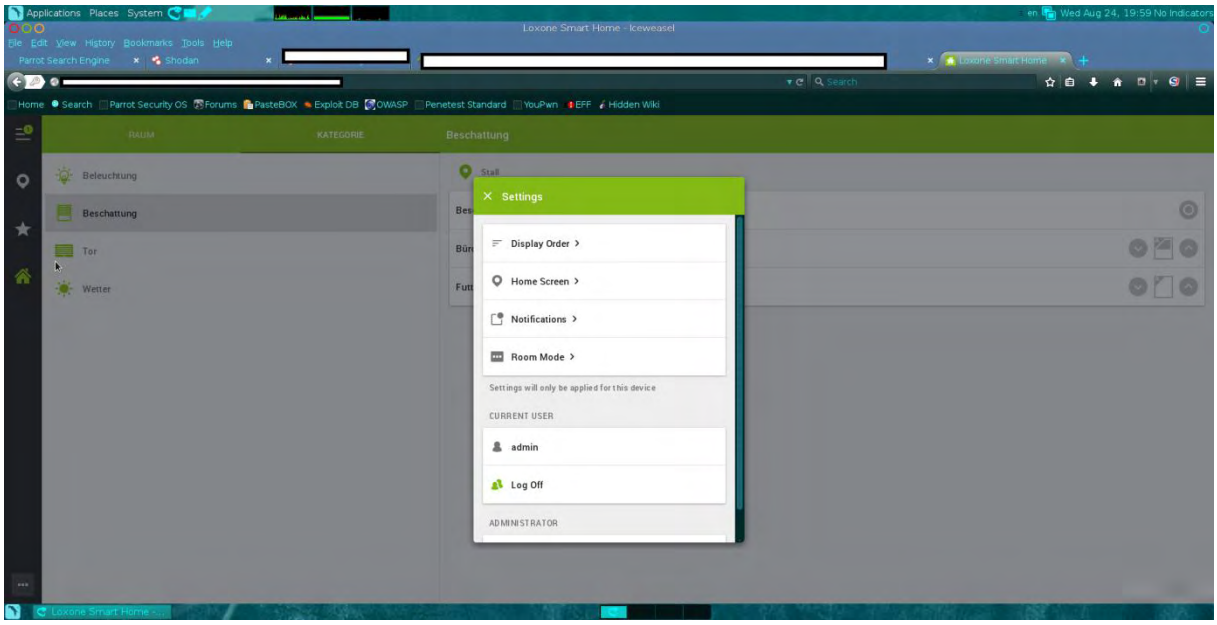
Screenshot 17 – Lighting control

LOXONE SMART HOME

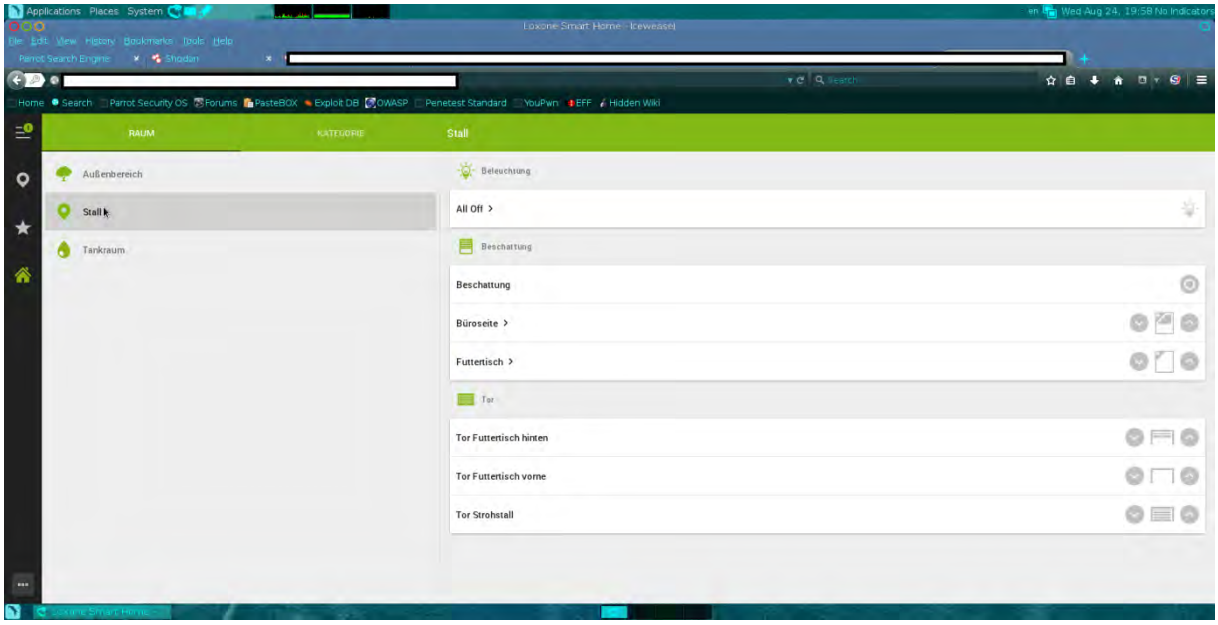
The search returned 803 vulnerable systems which most of them were located in EU. When access gained all settings could be customized, gained full access remotely of the home systems like lighting, door automation, alarm security and others.



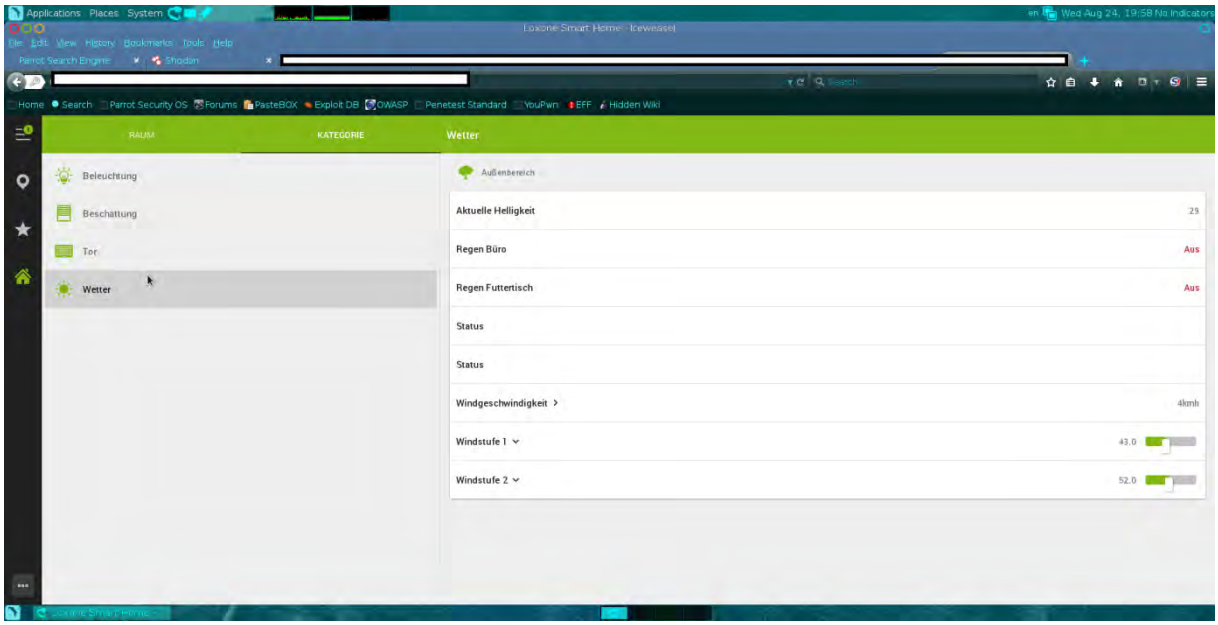
Screenshot 18 – Auto lighting system ON/OFF timer



Screenshot 19 – System settings



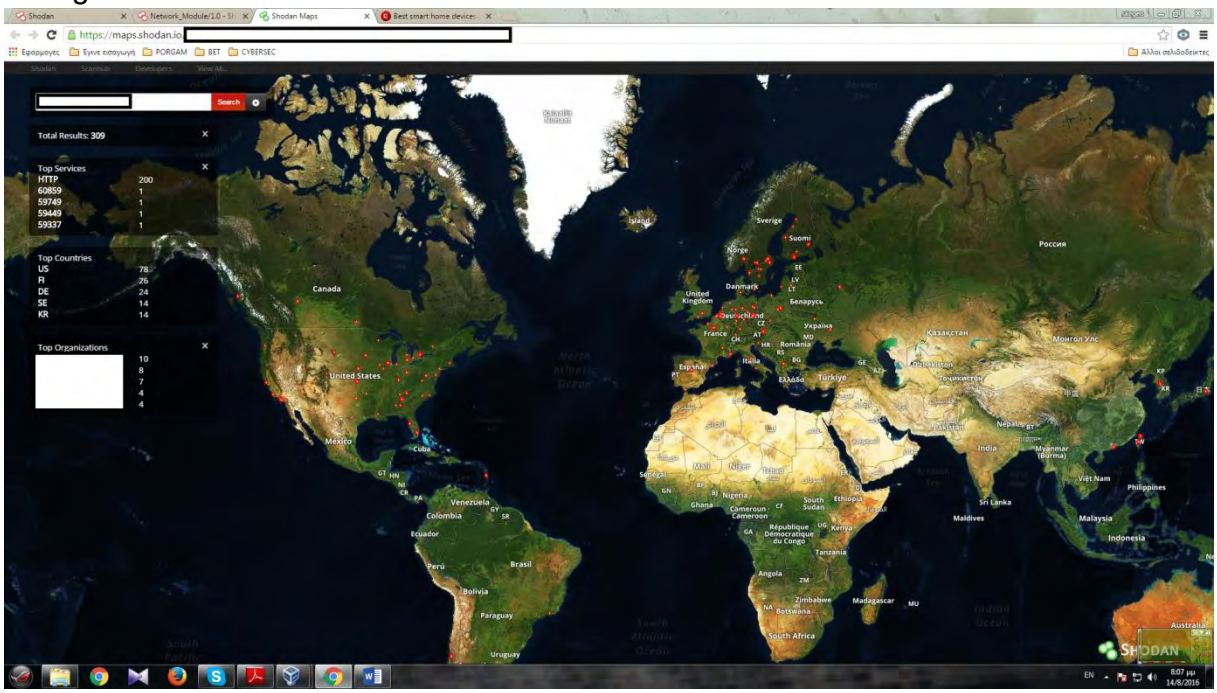
Screenshot 20 – System and lighting settings



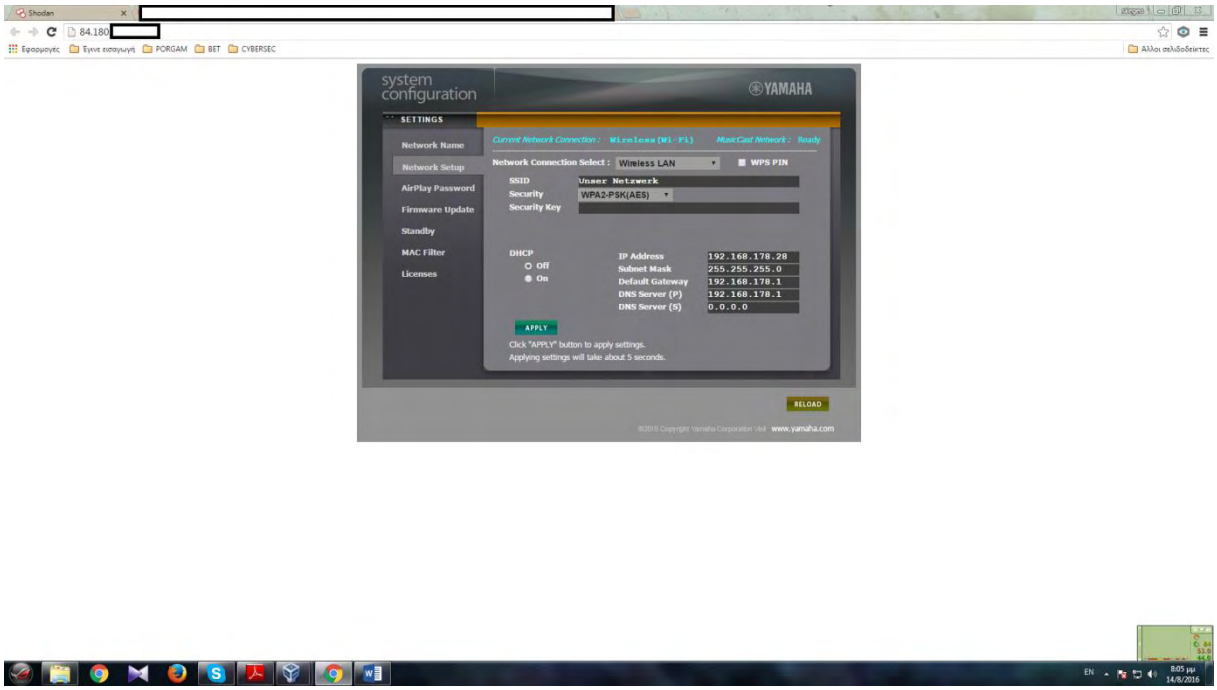
Screenshot 21 – Temperature settings

YAMAHA RX1 SMART STEREO SYSTEM

The result returned 309 results from all over the world. It was gained full remote access of all the features of the device like volume and full admin access at settings configuration.

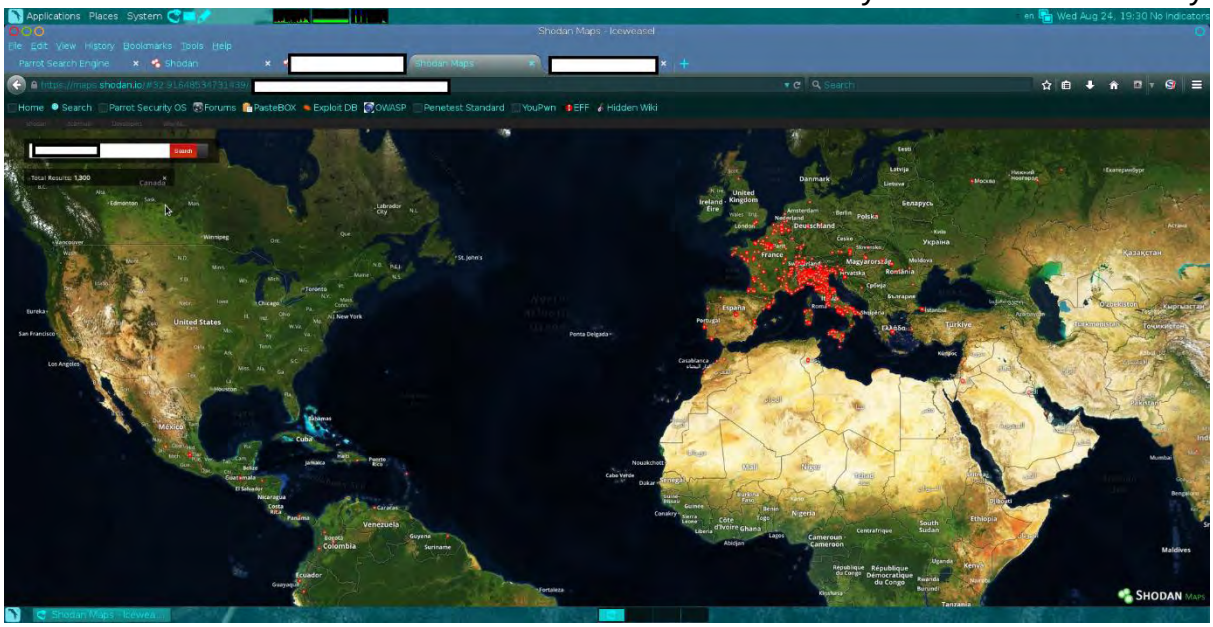


Screenshot 22 - Map results

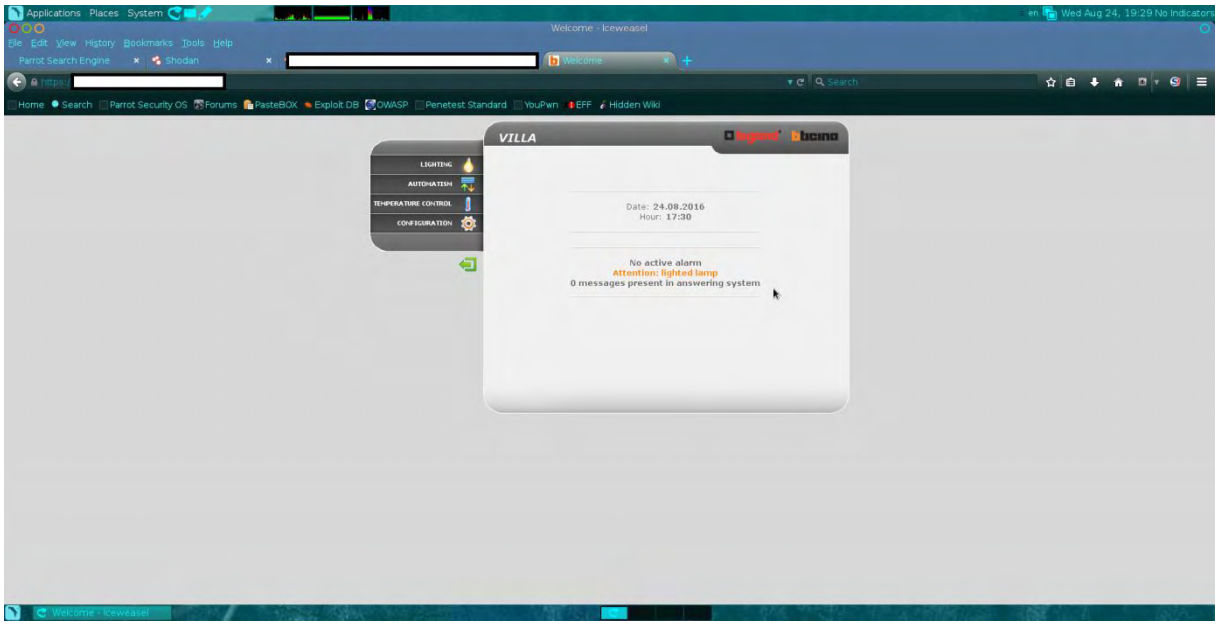


Screenshot 23 – Network access settings

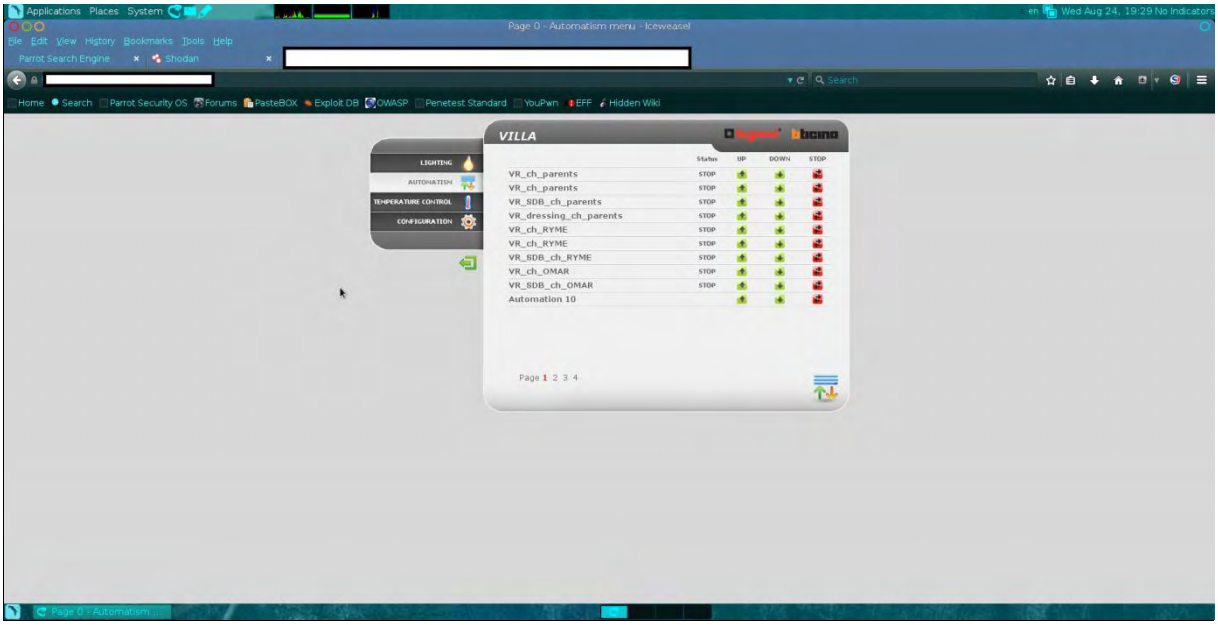
BTICINO LEGRAND SMART HOME There were 1300 results of vulnerable smart systems mostly in EU and 5 of them were in Greece. It was gained full access at lighting, door and window automation ,thermostat and air conditioning, cctv system, door camera and alarm system remotely.



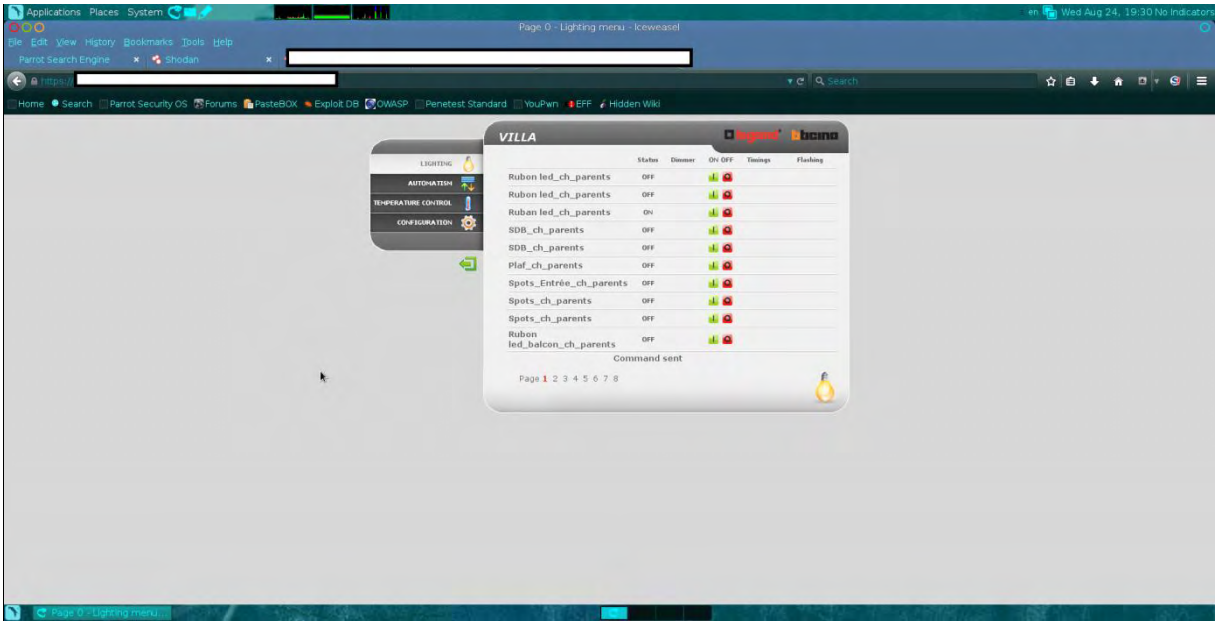
Screenshot 24- Map results



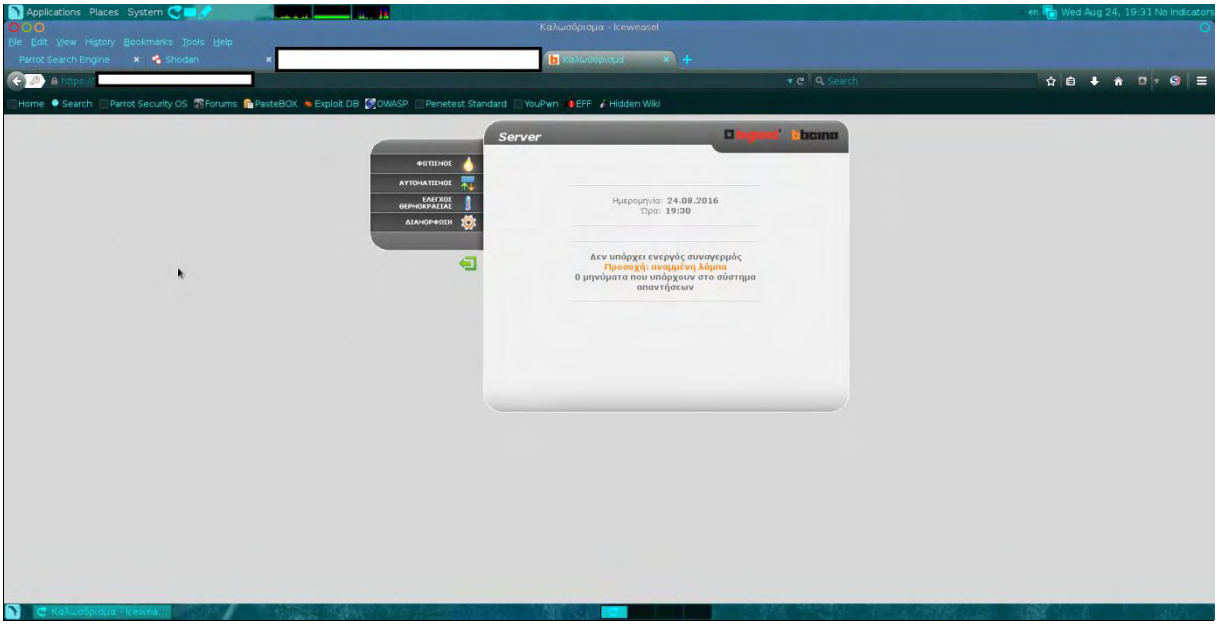
Screenshot 26- Main screen of alarm system



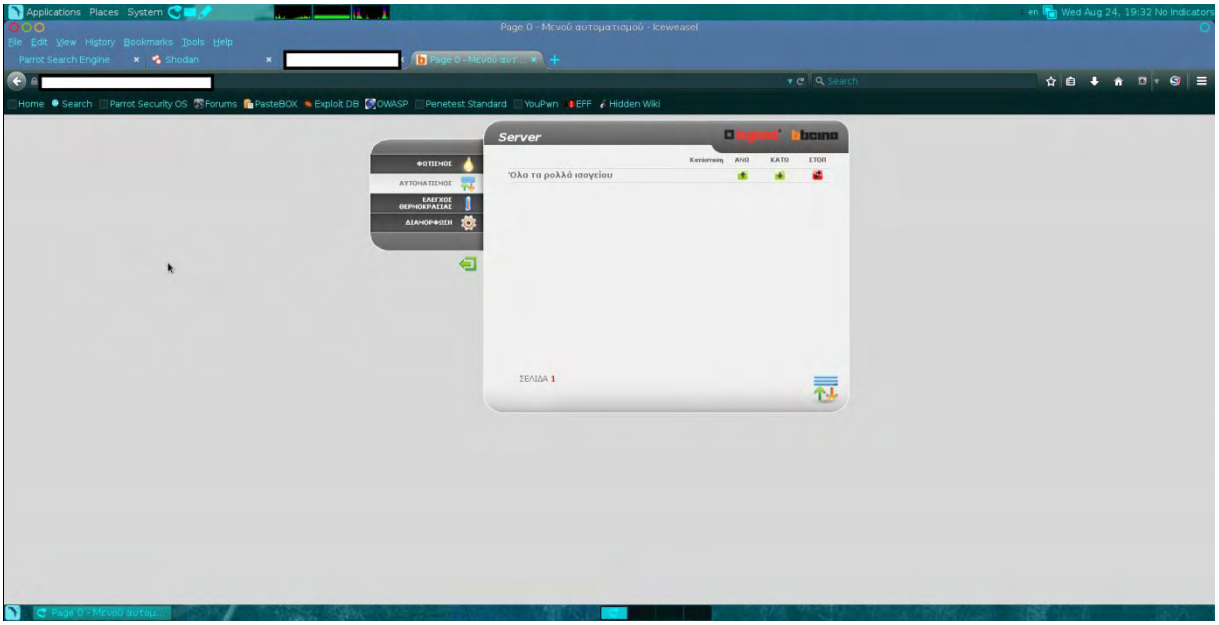
Screenshot 27- Windows controlling system



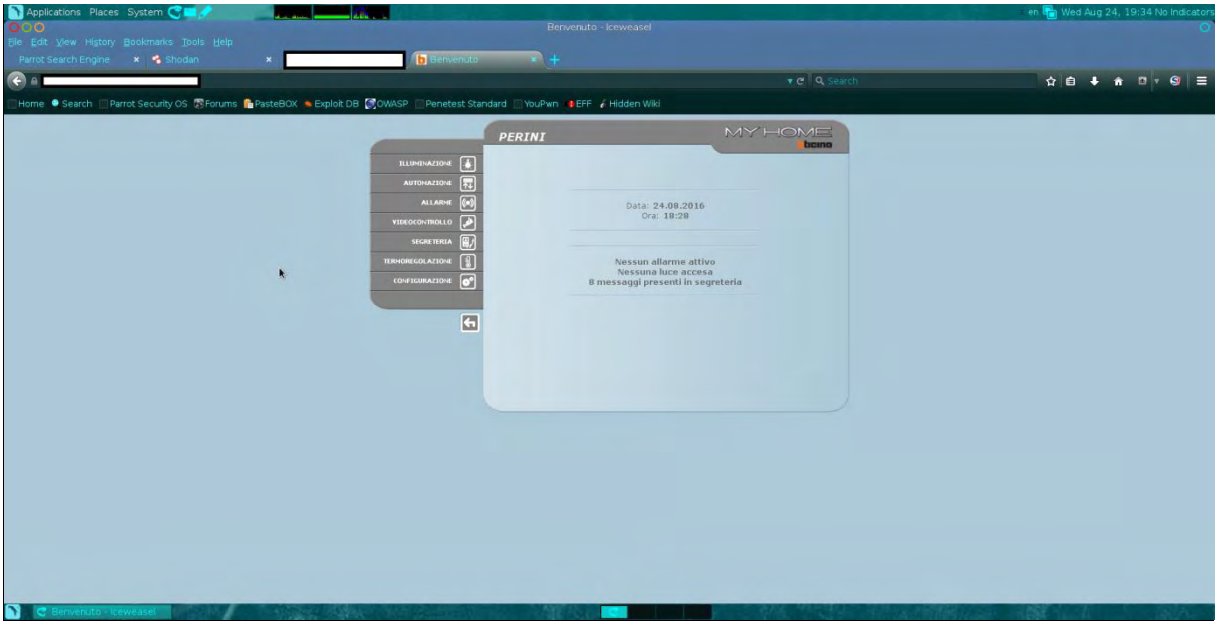
Screenshot 28 – Lighting ON/OFF



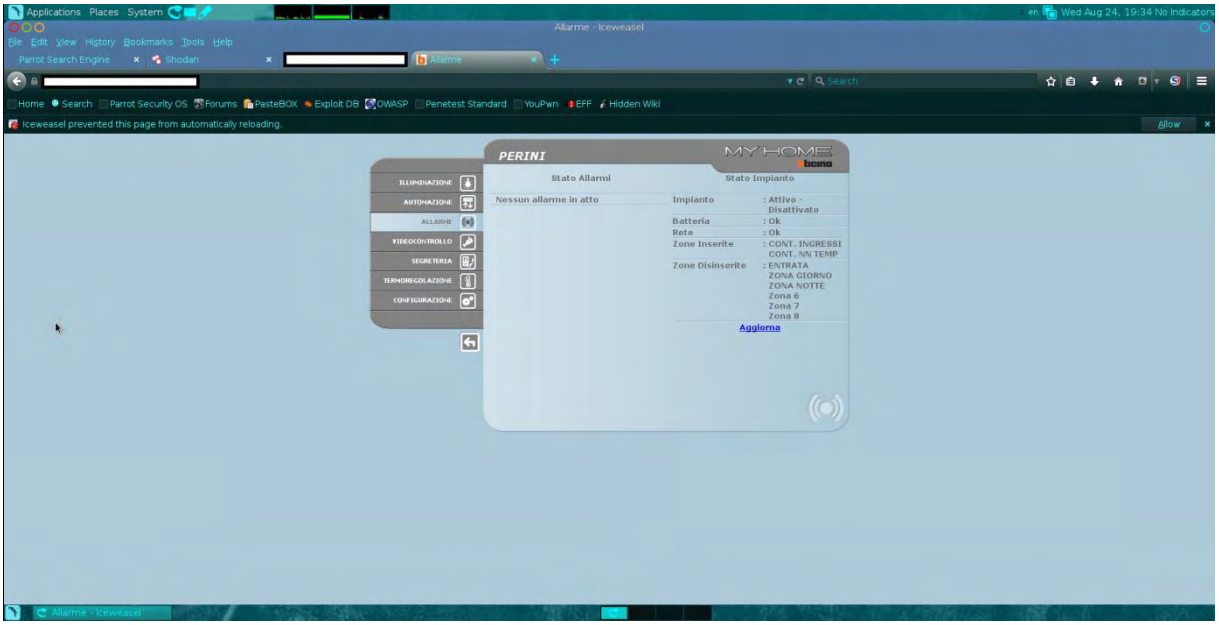
Screenshot 29 – Alarm system



Screenshot 30 – Main building front door system



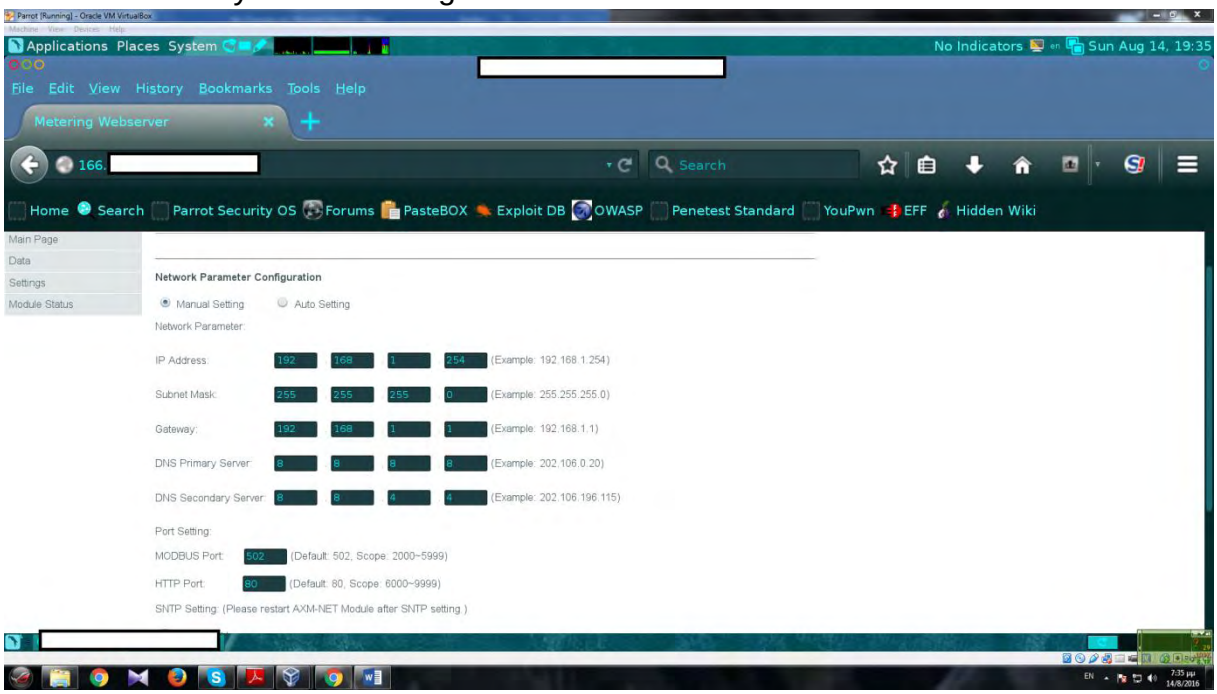
Screenshot 31 – System Home page



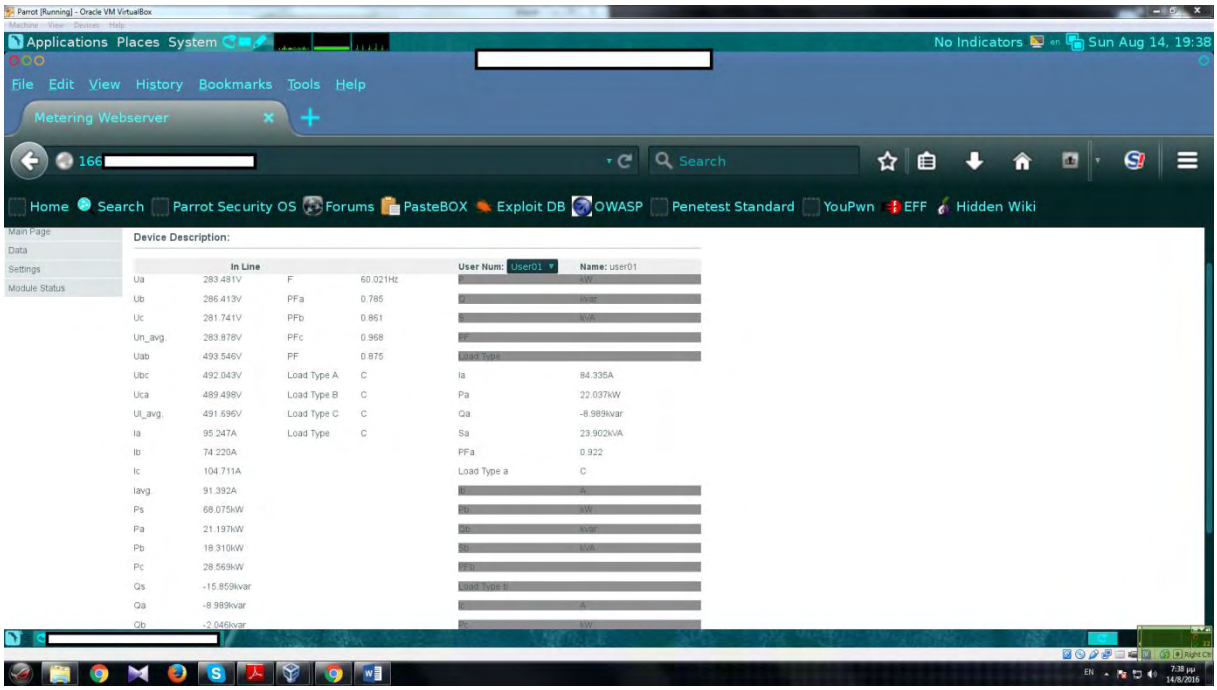
Screenshot 32 – ON/OFF burglar system

SMART METERING

They were found 85 smart meter vulnerable devices mostly in America. It was gained full admin system settings and live feed from the measurements



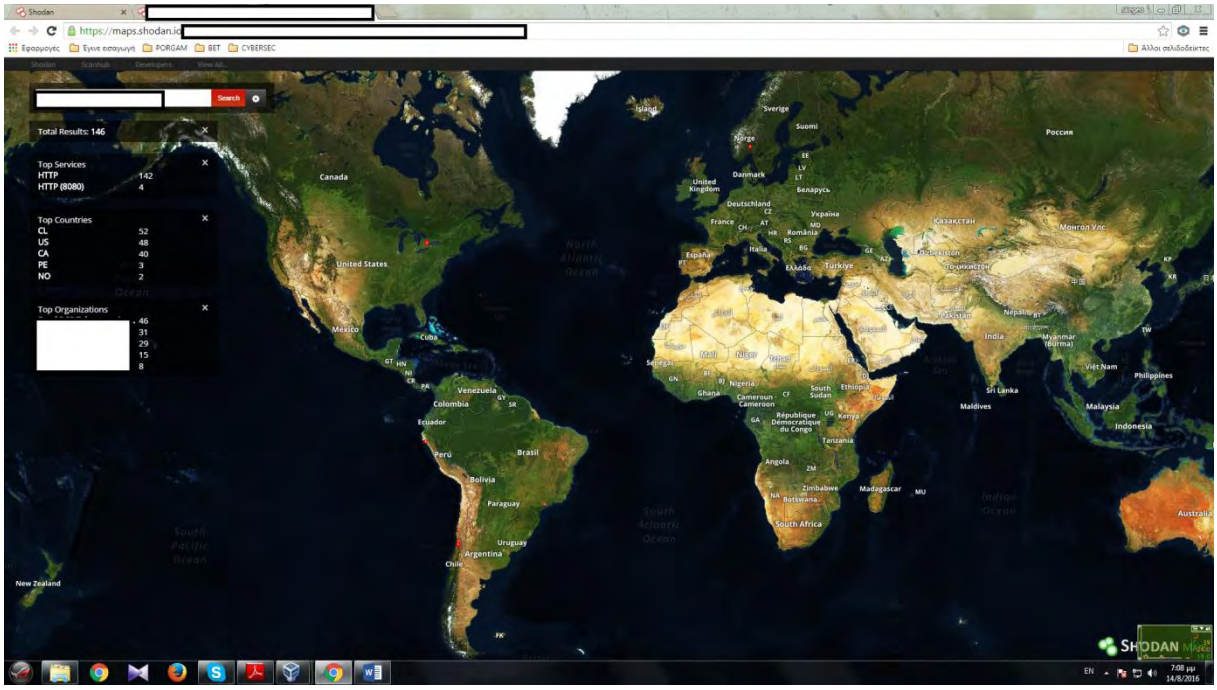
Screenshot 33 – Network access settings



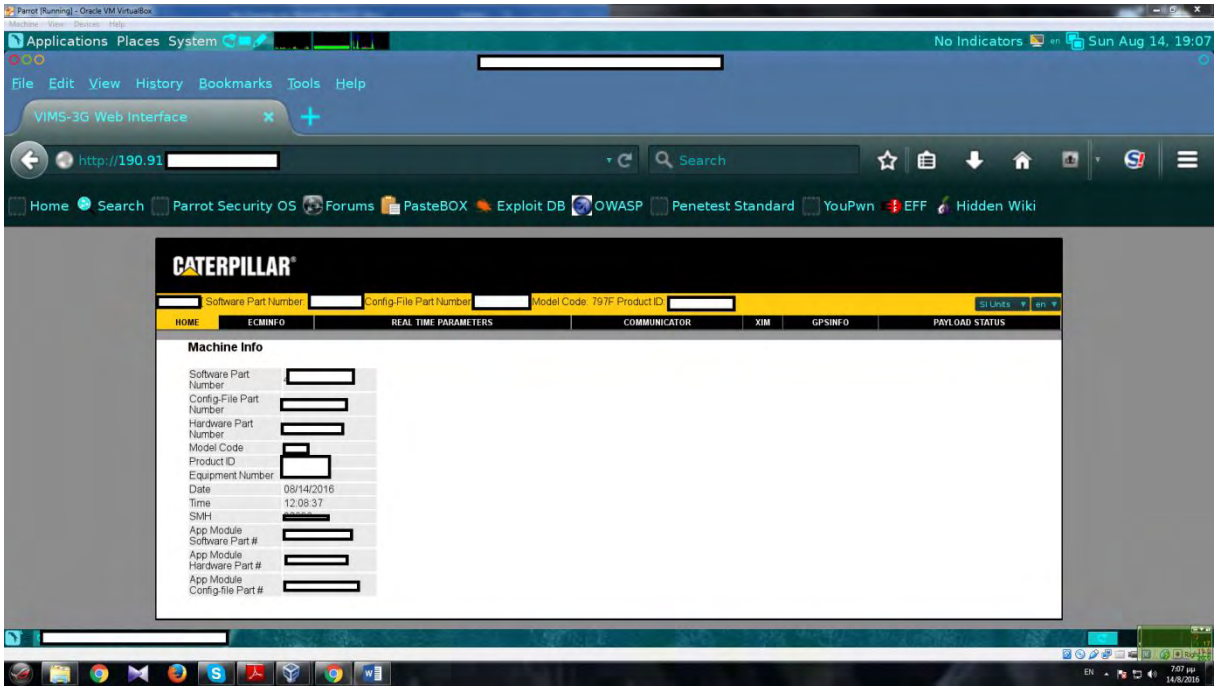
Screenshot 34 – Live metering results and overview

CATERPILAR SELF DRIVE TRUCKS

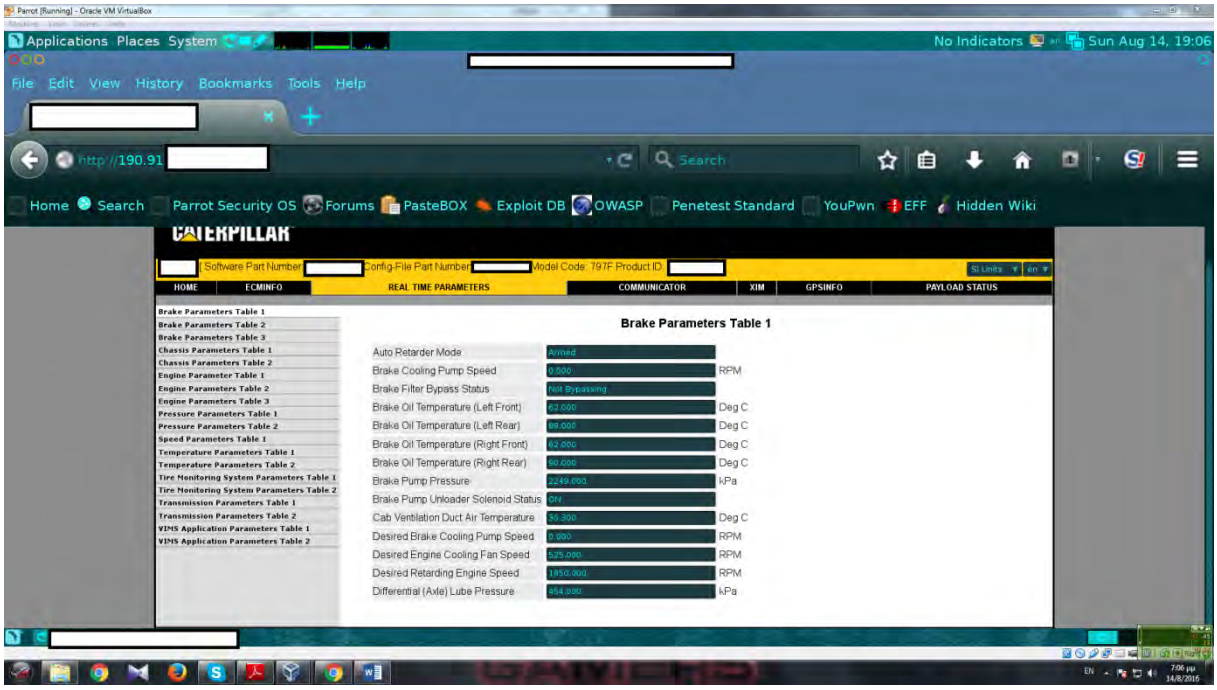
At the research that was done, were found 146 results of self-drive Caterpillar trucks. Access was gained at a lot of settings,real time monitoring and gps tracking.



Screenshot 35 – Map results



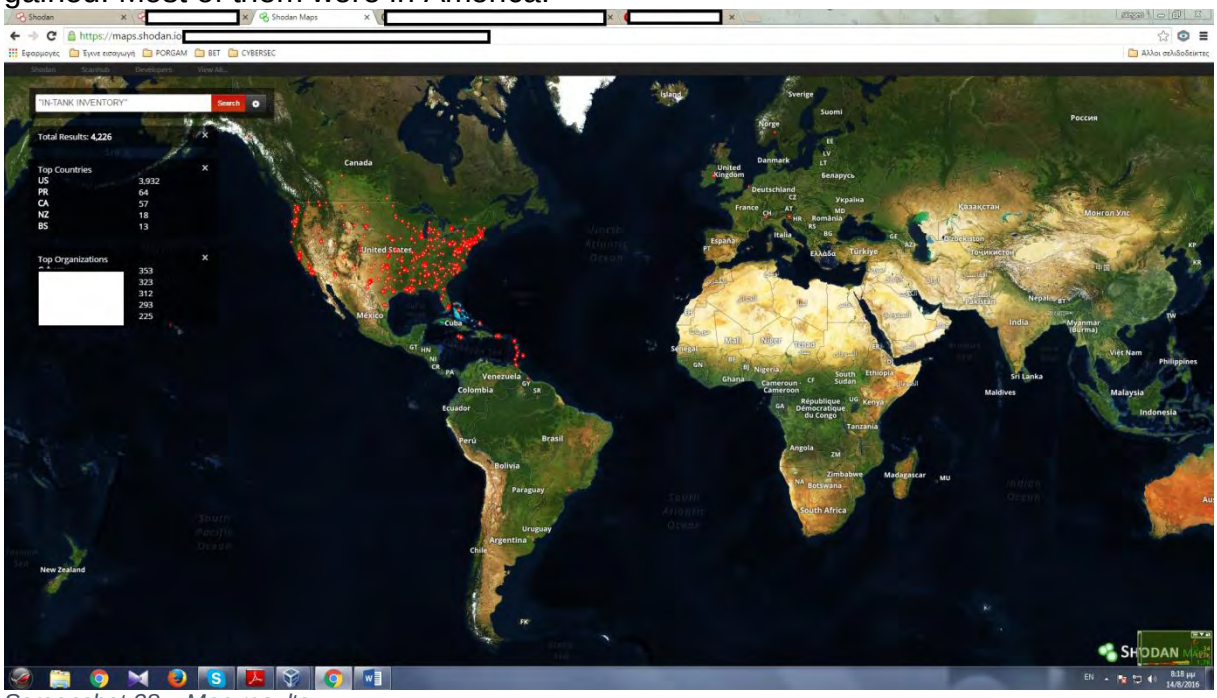
Screenshot 36 – Truck's information



Screenshot 37 – Truck's real time settings ,GPS info

GAS TANK LEVELS

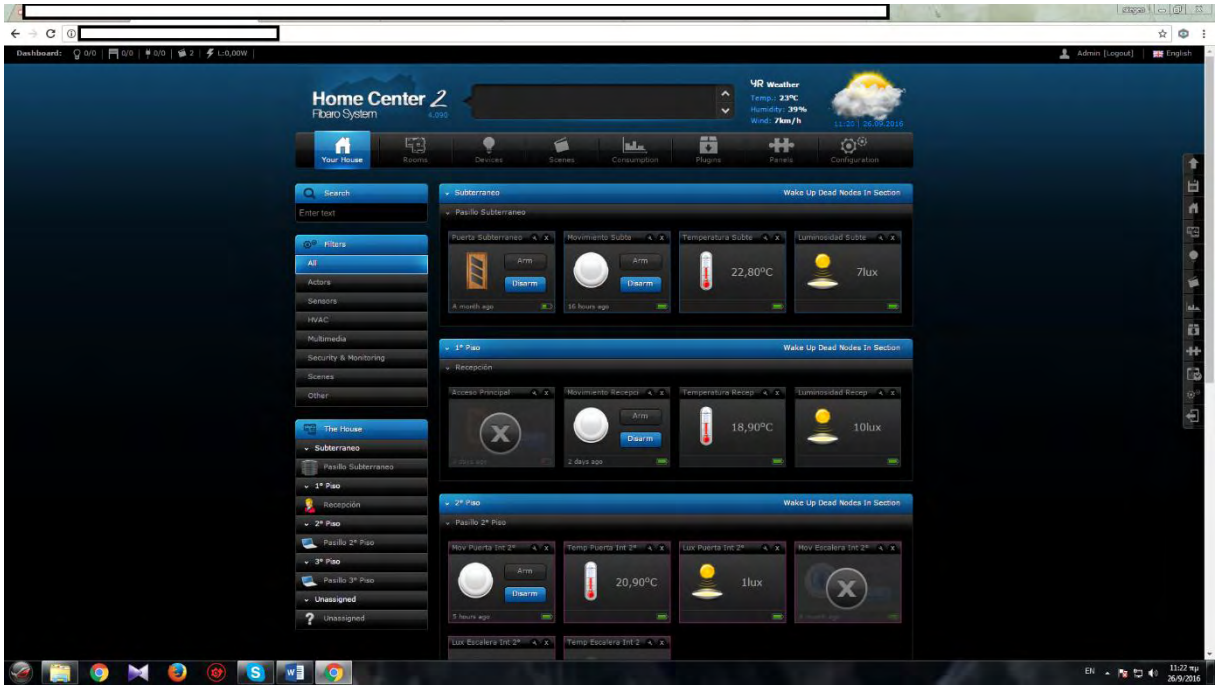
The search results were 4226 gas tanks where real time monitoring access was gained. Most of them were in America.



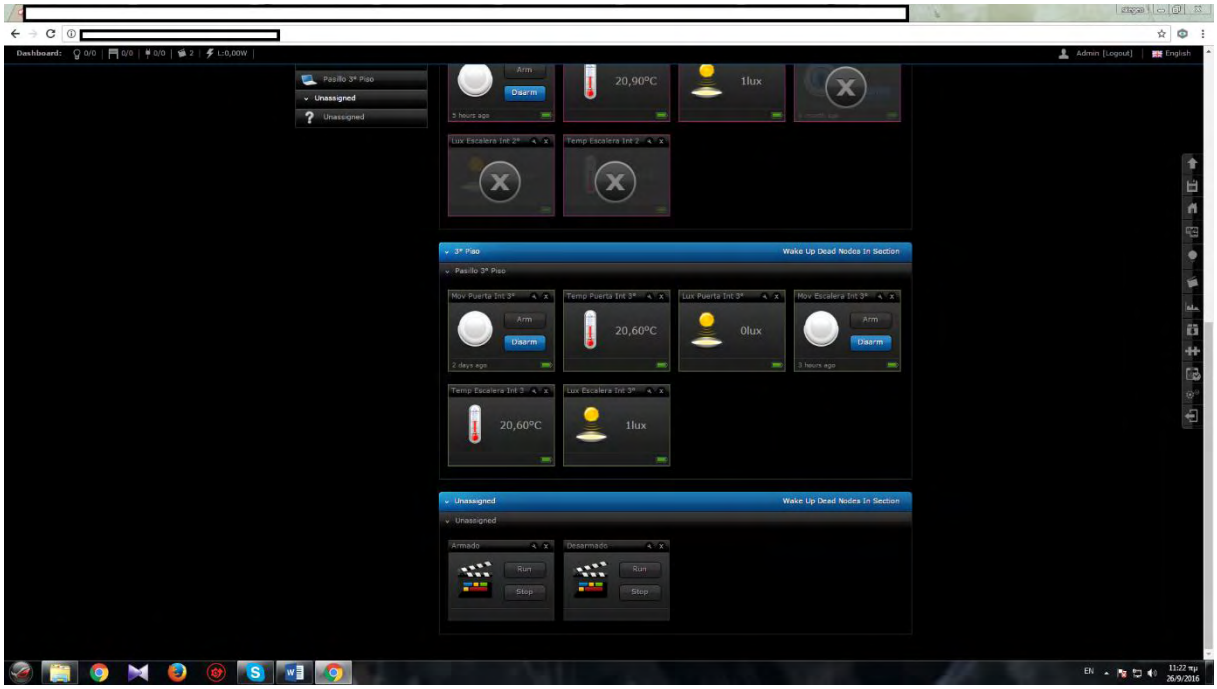
Screenshot 38 – Map results

FIBRARO HOME CENTER 2

The results were 928 from all over the world. After the bypass of the login screen with the use of default credentials,full access administration was achieved.We took control of every device and sensor that was connected such as electrical door locks, lighting, multimedia system, HVAC system and other as shown in the screenshots below.



Screenshot 38 – Door open/close,lighting ON/OFF,thermostat setting



Screenshot 39 – Lighting ON/OFF ,multimedia system

CONCLUSIONS

The Internet of Things continues to march forward and will accelerate with higher pace the coming years. The growth of more networks capable entities will regard to new potential attack surfaces and an explosive increase of cyber security issues. In this master thesis, security challenges and security threats that Internet of Things were analyzed. At the practical research that has be done, although we are at the beginning of the IoT technological era, a large amount of IoT devices are vulnerable. This shows that there is a very little concern from vendors for security design implementation. The vulnerable devices use case search was provided which showed that it is very easily to gain access to these devices. All these security issues conclude that it is important from vendors and end-users to gain deeper understanding of the threats that are facing. Internet of Things devices will be soon a vital part of every man's life and is critical to solve and face the most issues.

The internet of things will bring a better and more convenient life in the near future with many new advances about interacting with our world. However, internet of things will bring many challenges in the world of information technology and cyber security engineers making critical the continuous research and development of new approaches to ensure security and privacy.

Sources, citations and references

- Rethinking the Internet of Things, A Scalable Approach to Connecting Everything – Francis daCosta 2013
- Abusing the Internet of Things – Nitesh Dhanjani 2015
- Designing the Internet of things – Adrian McEwen & Hakim Cassimaly 2014
- Insecurity in The Internet of Things – Mario Ballano Bacena, Sr Threat Analysis Engineer Symantec Corporation 2015
- Internet of Things Top Ten – Open Web Application Security Project
- IOT Testing Guidance - Open Web Application Security Project
- The Internet of Things : A Ciso and Network Security Perspective – Jon Oltsik 2014
- Securing the Internet of Things: Mapping Attack Surface Areas Using the OWASP IoT Top 10 - Daniel Miessler RSA Conference 2015
- http://en.wikipedia.org/wiki/Internet_of_Things
- The Things in the Internet of Things- Stephan Haller, SAP Research Center Zurich
- Security in the Internet of Things – Wind Rivers Systems Inc. 2015
- SHODAN for Penetration Testers – Michael “theprez98” Schearer,Defcon 18
- Iot Attack Surfaces – Daniel Miessler, IOT Defcon Village 2015
- Inteligent Cities,Enabling Tools and Technology – Pethuru Raj & Anupama C. Raman 2015
- Security Guidance for Early Adopters of the Internet of Things – Cloud Security Alliance 2015
- Embedded Devices Security and Firmware Reverse Engineering - Jonas Zaddach & Andrei Costin
- Understanding the Internet of Things Protocols – Stan Schneider
- Overview of Security and Privacy Issues in the Internet of Things – Chris Lu 2014
- Security by Design Principles - Open Web Application Security Project
- Security and Trust in IoT – based Complex Systems - Prof. Dr. Petre Dini
- Complete Guide to Shodan – John Mathelry 2015

The Google Dorks and the Shodan Queries that used in the use case of this master thesis are:

Axis Webcams

inurl:/view/viewer_index.shtml

Webcamxp5

intext:"powered by webcamXP 5"

HP Officejet Printer

inurl:/tcpipv6.htm

Ricoh Photocopier

inurl:"topPage.cgi" | inurl:"mainFrame.cgi" intext:"Web Image Monitor"

Prolifix Thermostat

Title:”Status&Amp;+Control”

Heating System DDC400

Title:”DDC4000”

Rapsberry Pi Smart Home

HomeAssistant/1.0 Python/

CAT Self Drive Dump Trucks

Title:”VIMS-3G Web+Interface”

Smart Metering

Server: FNET+HTTP+-+Freescale+Embedded

Loxone

“loxone smart+home”

Yamaha RX1 Smart Stereo

Network_Module/1.0

Gas Tanks Levels

“IN-TANK INVENTORY”

Heatmiser Thermostat

Title:”Heatmiser Wifi + Thermostat”

Bticino Legrand Smart Home

Location:/iden.php