Characterizing and Classifying IoT Traffic in Smart Cities and Campuses

Paper's analysis
ITPA 2019-2020

Andrea Graziani - 0273395

Università degli Studi di Roma "Tor Vergata"
FACOLTA' DI INGEGNERIA
Corso di Laurea Magistrale in Ingegneria Informatica

December 2, 2020

Research's goal - Part 1

According to Sivanathan et al. [6], research's goal is to:

> *"[...] develop a classification method that can not only distinguish IoT from non-IoT traffic, but also identify specific IoT devices with over 95% accuracy."*

What is the reason according to which is important to profile IoT traffic?

1. To understand IoT devices "*normal*" **traffic pattern** in terms of their **activity pattern** (traffic rate, idle durations, etc.) and **signalling overheads** (`DNS`, `NTP`, etc.).
2. To enhance **cyber-security** involving IoT devices which administration belong to **different authorities**.
   According to Sivanathan et al. [6], is possible to improve security deploying a network-level security mechanisms which, analysing traffic patterns, is capable to **identify attacks knowing the normal traffic pattern of monitored IoT devices**.

Research's goal - - Part 2

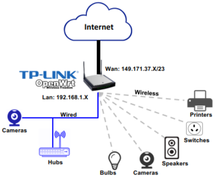
In other words, research's goal is to build an classification model for IoT devices based on **machine learning** techniques, which building passes through following steps:

1. Collect data from an IoT environment.
2. Characterize traffic pattern corresponding to the various IoT devices.
3. Develop a classification technique that learns the behaviour of an IoT device and is able to identify it based on its traffic pattern.

The "Smart Environment" - Overview - Part 1

- In order to collect data, Sivanathan et al. [6] built a "*Smart Environment*" to simulate a real usage scenario including:
  - 21 unique IoT devices representing different categories, like **cameras**, **healthcare devices**, **hub**, **air quality sensors** and so on.
  - A router, the TP LINK ARCHER C7[1]
  - Several **non-IoT** devices were also used, such as laptops, mobile phones and tablet.

Figure: "Smart environment"'s scheme



[1] https://www.tp-link.com/it/home-networking/wifi-router/archer-c7/#overview

The "Smart Environment" - Overview - Part 2

Several IoT devices used by Sivanathan et al. [6] for their experiments **are battery operated**.
For instance:

- The *Withings Smart scale* device is powered by 4 1.5 V alkaline cells (AAA).[a]
- Similarly, the *Netatmo Weather station* device is powered by 2 1.5 V alkaline cells (AAA) with an **estimated autonomy of about 2 years**.[b]
- The *Blipcare blood pressure meter* device is powered by an internal battery.[c]

[a] https://www.withings.com/it/en/body
[b] https://www.netatmo.com/it-it/weather/weatherstation/specifications
[c] http://www.blipcare.com/

The "Smart Environment" - Architecture - Part 1

The architecture of the "Smart Environment" can be splitted into:

Front-end which contains the *router* and the *IoT/non-IoT devices*. Both *wireless* and *wired* interfaces are used.

Back-end represented by the *cloud*.
IoT devices also rely on cloud back-end services for **data storage**, **backup**, **firmware updates**, **remote access and integration**, and other services, like **media streaming** [3].

- Proposed architecture is intended for **cloud-native** IoT applications and services [4].
- Cloud resources are exploited through so-called **cyber-foraging techniques** in order to overcome the very strictly constrains of any IoT devices. [4].

The "Smart Environment" - Architecture - Part 2

The implementation of the "*Smart Environment*" is based on a **star network topology**.

The use of a star network topology allows us to:

- **Preserve battery life** of IoT devices because they **do not have to forward** other nodes data; in other words, *any IoT device receives, or transmits, only its own data* [2].
- **Decrease the complexity** of the network [2] making infrastructure deployment cost low [**nokia**].

Also the implementation of `LoRaWAN` network is based on the star network topology, and mostly, stars-of-stars network [2].
As known `LoRaWAN` network belongs to `LPWAN` category, *which are specifically designed to achieve the need for low power, long-range, low bit error rate, and low cost* needed in IoT context.

The "Smart Environment" - Wireless Technologies - Part 1

According to vendor's specifications regarding the TP Link Archer C7, is possible to know that aforementioned router supports following protocols:

- IEEE 802.11ac/n/a at 5 GHz
- IEEE 802.11n/b/g at 2.4 GHz

Researchers use IEEE 802.11 as **media access control** (MAC) and **physical layer** (PHY) protocol.

Researchers did *not* specify which **version** of IEEE 802.11 standard has been effectively used.
We don't know with which **frequencies** data has been transmitted.

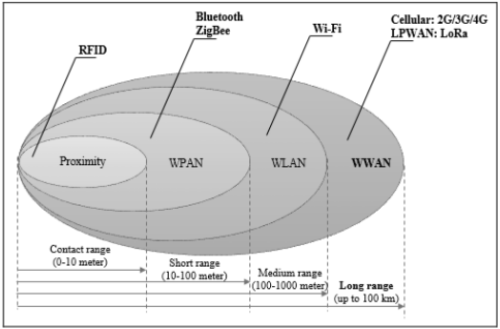
The "Smart Environment" - Wireless Technologies - Part 2

Above wireless technologies are **not** fully optimized for IoT business models and devices used in **smart cities** and **campuses**.

- Despite high reliability, low latency, and high transfer rates (using `802.11ac` 5 GHz we can achieve 1300 Mbps), due to their **inherent complexity** and **energy consumption** are generally not suitable for all IoT nodes [7].
  Generally, *constrained* IoT devices requires **low power** and **less data-rate** [**nokia**]; therefore `IEEE 802.15.4`, `IEEE 802.11ah` or `LoRaWAN` can be better choice [7].
- These technologies provide a **short/medium coverage** with **100-to-1000** meters range [2]. Provided coverage range can be not enough to fulfil all use cases.
  - This is due to **mid/high frequencies** used by these protocol which are **vulnerable to several side effect during signal propagation** (*blocking*, *reflection*, *refraction* and so on) [5].

The "Smart Environment" - Wireless Technologies - Part 3

Utilizing sub-1 GHz bands is possible to provide **better propagation characteristics** in outdoor scenarios.
Low frequencies signal are less affected by obstacles presence.



The “Smart Environment” - Wireless Technologies - Part 3

Adopted wireless technologies are suitable only for *unconstrained* IoT devices in **short-medium** range scenarios. Experimental setup not cover all smart city and campus use cases.

A combination of *low-band*, *mid-band* and *high-band* spectrum is desirable to manage all possible use cases.

IoT Traffic - Part 1

According to their experimental results, Sivanathan et al. [6] stated that:

> *"[...] if we consider only the load imposed by the IoT devices, then there is a dramatic reduction in the peak load (*1 *Mbps) and average loads (*66 *Kbps), [...], implying that traffic generated by IoT devices is small compared to traditional non-IoT traffic."[6, par. IV.A]*

> *"the traffic pattern of one IoT device [...] a pattern of active/sleep communication emerges. [...] IoT active time [...] decays rapidly initially (only* 5% *of sessions last longer than* 5 *seconds), with the maximum active time being* 250 *seconds in our trace. This shows that IoT activities are short-lived in general."[6, par. IV.A]*

IoT Traffic - Part 2

Since many IoT devices are battery powered, **maximize energy efficiency**, in order to **preserve devices lifetime**, is critical.

According to Sivanathan et al. [6]'s results, the power management approach adopted by IoT devices is based on **periodic sleep**, during which radio transceiver are turned off.

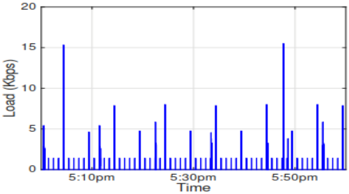
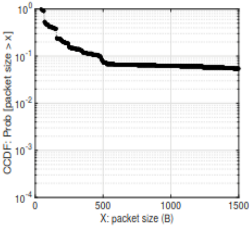


Figure: Load of LiFX light bulb device.

IoT Traffic - Part 3

- A very interesting observation by Sivanathan et al. [6] made by regard packet size, according to which only the 10% of packets are larger than 500 Bytes.
  - **header-overhead**, caused by **short packets transmission**, can occur frequently.

Figure: "Smart environment" 's scheme



The Most Dominant Application Layer Protocols - Part 1

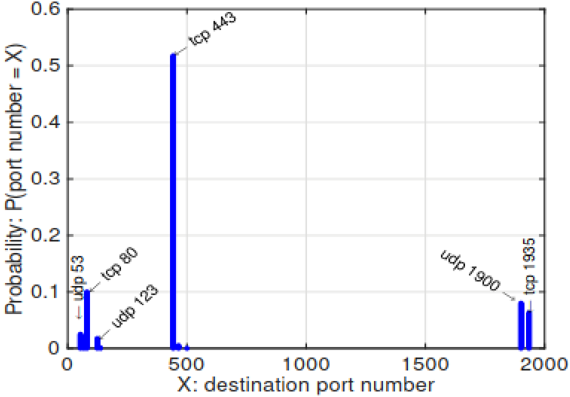


Figure: Probability histogram of destination port numbers for IoT packets destined to both the local network and the Internet.

The Most Dominant Application Layer Protocols - Part 2

HTTPS (TCP port 443) is the dominant protocol used by the IoT devices since it represents over the 55% of total IoT traffic.

HTTP (TCP port 80) represent the second most dominant application layer protocol constituting the 11% of total traffic.

SSDP (UDP port 1900) is the next most dominant application layer protocol representing the 8% of traffic.

- SSDP, which stands for **Simple Service Discovery Protocol**, is used to for *advertisement* and *discovery* purposes of network services without the assistance of server-based configuration mechanisms, such as DHCP or DNS.

RTMP (TCP port 1935) represent the fourth most dominant protocol representing the 7% of traffic

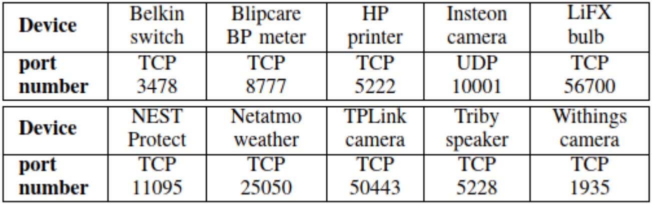
- RTMP, which stands for **Real-Time Messaging Protocol**, is a proprietary protocol used for streaming audio, video and data over the Internet, generally used by cameras. It is owned by Adobe.

The Most Dominant Application Layer Protocols - Part 3

DNS (UDP port 53) represents less than 5/4% of total traffic.

NTP (UDP port 123) constitutes less than 2/3% of IoT traffic.

Application specific Sivanathan et al. [6]'s results shows that, regarding remaining IoT traffic, each IoT device use an own **application-specific** protocol.
In table reported below, are reported most frequent *transportation protocol* and *port number*.



Device	Belkin switch	Blipcare BP meter	HP printer	Insteon camera	LiFX bulb
port number	TCP 3478	TCP 8777	TCP 5222	UDP 10001	TCP 56700

Device	NEST Protect	Netatmo weather	TPLink camera	Triby speaker	Withings camera
port number	TCP 11095	TCP 25050	TCP 50443	TCP 5228	TCP 1935

The role of HTTP - Part 1

Why results that HTTPS and HTTP are the most used protocols by IoT devices?

- A very important aspect of an urban, or campus, IoT infrastructure is the **necessity to make data collected by the urban IoT devices easily accessible** by both authorities and citizens [7].
- In order to achieve this objective, IoT devices adopt a very well known web-based paradigm called **Representational State Transfer** (**ReST**), which plays a very important role into **Web of Things Architecture** (**WoT**) [7][1].

The role of HTTP - Part 2

- Exploiting REST paradigm, HTTP and HTTPS are used very frequently because they facilitate both the **integration of IoT devices** with existing services currently available on the Web and the **Web applications development** [7][1].
- HTTP and HTTPS offer a **direct access** for users to IoT devices data and services, without the need for installing additional software [7].
  In fact, using a Web browser (or any HTTP library in the case of a software client) client are able to to directly extract, save and share smart things data and services.
  This ensures the usability of the architecture and minimizes the entry barriers for final users [1].

The disadvantages of HTTP

- The **verbosity** and **complexity** of native HTTPS/HTTP make them **unsuitable** for constrained IoT devices [7].
  - In fact, the **human-readable format** of HTTP, which has been one of the reasons of its success in traditional networks, turns out to be a limiting factor due to the large amount of heavily correlated (and, hence, **redundant**) data [7].
- HTTPS/HTTP rely upon the TCP transport protocol that, however, does not scale well on constrained devices, yielding poor performance for small data flows in lossy environments [7].

Security Problems Due To Unencrypted Traffic - Part 1

- According to Sivanathan et al. [6], about 45% of IoT traffic is **not** sent over HTTPS to the servers.
  - Since the traffic transmitted using other protocols are typically not encrypted, Sivanathan et al. [6]'s results indicate that a sizeable fraction of IoT traffic is **not** being securely transported over the Internet.
  - The use of unencrypted protocols can leak sensitive information about users [3].

Security Problems Due To Unencrypted Traffic - Part 2

Why IoT devices transmit unencrypted data?

There may be various reasons according to which data are transmitted unencrypted:

- Due to **limitations and constrains** in the IoT device itself.
- As noted by Mazhar and Shafiq [3], IoT devices vendors may be hesitant to move to HTTPS if their products use any third-party resources that are HTTP-only. These resources are typically **ads** and **trackers** [3].
- Bad design.

Other considerations

Experiment's results show following differences among IoT and Non-IoT traffic:

DNS traffic IoT devices initiate DNS queries for only a limited number of domains while non-IoT device, such as a laptop, looks for more than 300 domain names in a course of a few hours.

Number of Cloud servers IoT device communicates with less than 10 servers on average per day while non-IoT device contacts about 500 different servers

Machine learning - Part 1

Since Sivanathan et al. [6] adopted a **supervised machine learning algorithms** to build their classification model, is necessary to generate a **data-set** in order to provide an appropriate input during the **learning phase**.

- Sivanathan et al. [6] collected traffic over 3 weeks generated from the "*Smart Environment*".
    - 2 weeks of data was used for *training* and *validation*.
    - last week was used for *test*.
- Is very important to precise that collected data are **time series**, where each instance, indexed by time, contains several **attributes** (or **features**) like *sleep time*, *active time*, *average packet size* and so on.
- Clearly, every instance contains a **label** identifying the IoT device, which is necessary during supervised learning.

An Unbalanced Data-Set - Part 1

The performance and the interpretation of a IoT device classification model **depend heavily on the data** on which it was **trained**.

- Scientific literature showed that classification model, which are trained on *imbalanced datasets*, are highly susceptible to producing inaccurate results.

- Could Sivanathan et al. [6]'s dataset be unbalanced?
- Could Sivanathan et al. [6]'s classification model be unsuitable to correctly classify IoT devices in a real smart-city and campus scenario?

An Unbalanced Data-Set - Part 2

- Generally smart city and campus services are based on a **very heterogeneous set of IoT devices**, generating **very different types of data** that have to be delivered through **suitable communication technologies**.
  - For instance, possible IoT use cases can be:
    - Structural Health of Buildings.
    - Waste Management.
    - Air Quality.
    - Traffic Congestion.
    - Noise Monitoring.
    - City Energy Consumption.
    - Smart Parking.
    - Smart Lighting.

- Proposed "*Smart environment*" simulates too few IoT use cases relating to a smart-city or campus.
- Proposed environment is more suitable for a *smart-home* rather than a smart-city or campus.

An Unbalanced Data-Set - Part 3

- Too few use cases. Only short/medium range use cases.
- It includes **only** *unconstrained protocol stack*, that is protocols that are currently the de-facto standards for Internet communications and are commonly used by regular Internet hosts (HTTP/TCP/IPv4). These protocol are suitable only for unconstrained IoT devices [7].
  - In fact there is a prevalence of HTTPS/HTTP application layer protocol (66% of total IoT traffic according to Sivanathan et al. [6]) and of the TCP transport layer protocol (representing, more or less, the 85% of total transmitted packets according to Sivanathan et al. [6]'s results).
- It does **not** include any *constrained protocol stack*, the low-complexity counterparts of the de-facto standards for Internet, i.e., **Constrained Application Protocol** (CoAP), UDP, and 6LoWPAN, which are suitable even for very constrained devices [7].

Validation technique - Part 1

Sivanathan et al. [6] state that:

> *"Our cross-validation method randomly splits the dataset into training (90% of total instances) and validation (10% of total instances) sets. This cross-validation is repeated 10 times. The results are then averaged to produce a single performance metric."*

To perform validation step, Sivanathan et al. [6] used the **10-fold cross-validation method**.

Validation technique - Part 2

Since their datasets contains **time-series data**, in order to take into account the time-sensitiveness of data, is required a validation technique, that is a *technique which defines a specific way to split available data in train, validation and test sets*, capable **to preserve the temporal order of data**, preventing for example that the testing set contains data antecedent to the training set.

Traditional methods of validation, like 10-fold cross-validation method, are unusable in this context.

References

[1] D. Guinard. “A Web of Things Application Architecture Integrating the Real-World into the Web”. In: 2011.

[2] Dina Ibrahim and Dina Hussein. “Internet of Things Technology based on LoRaWAN Revolution”. In: June 2019. DOI: 10.1109/IACS.2019.8809176.

[3] M. Hammad Mazhar and Zubair Shafiq. *Characterizing Smart Home IoT Traffic in the Wild*. 2020. arXiv: 2001.08288 [cs.NI].

[4] Mahadev Satyanarayanan et al. “The Seminal Role of Edge-Native Applications”. In: July 2019, pp. 33–40. DOI: 10.1109/EDGE.2019.00022.

[5] J.H. Schiller. *Mobile Communications*. Addison-Wesley, 2003. ISBN: 9780321123817. URL: https://books.google.it/books?id=FdojEVT1Oj4C.

[6] A. Sivanathan et al. “Characterizing and classifying IoT traffic in smart cities and campuses”. In: *2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. 2017, pp. 559–564. DOI: 10.1109/INFCOMW.2017.8116438.

[7] Andrea Zanella et al. “Internet of Things for Smart Cities”. In: *Internet of Things Journal, IEEE* 1 (Jan. 2012). DOI: 10.1109/JIOT.2014.2306328.