FOGG: A Fog Computing Based Gateway to Integrate Sensor Networks to Internet

Sripiya S. Adhatarao, Mayutana Arumaiturai and Xiaoming Fu
Institute of Computer Science, University of Göttingen, Germany.
Email: {adhatarao,arumaiturai,fx}@cs.uni-goettingen.de

Abstract—Internet of Things (IoT) is a growing topic of interest along with 5G. Billions of IoT devices are expected to connect to the Internet in the near future. These devices differ from the traditional devices operated in the Internet. We observe that Information Centric Networking (ICN), is a more suitable architecture for the IoT compared to the prevailing IP based network. However, we observe that recent works that propose to use ICN for IoT, either do not cover the need to integrate Sensor Networks with the Internet to realize IoT or do so inefficiently. Fog computing is a promising technology that has many benefits to offer especially for IoT. In this work, we discover a need to integrate various heterogeneous Sensor Networks with the Internet to realize IoT and propose FOGG: A Fog Computing Based Gateway to Integrate Sensor Networks to Internet. FOGG uses a dedicated device to function as an IoT gateway. FOGG provides the needed integration along with additional services like name/protocol translation, security and controller functionalities.

Keywords—Fog computing, Sensor Networks, IoT, ICN

I. INTRODUCTION

Internet of Things (IoT) refers to a network of devices like machines, vehicles, electronic appliances and also wearables like Radio Frequency IDentification tags (RFID), Step-counter, etc. These devices are usually embedded with sensors, actuators, memory and network connectivity. They are mainly used for sensing, monitoring and controlling various applications. IoT is a growing topic of interest especially with 5G and has already drawn attention of academia and Industry. Billions (50 Billion by 2020 [1]) of IoT devices are expected to be connected in the near future to communicate, sense and gather data. However, it is important to understand the difference between IoT and Sensor Networks. There are various heterogenous Sensor Networks that operate in their own private network. The goal of IoT is to provide the Sensor Networks access to the Internet to realize IoT [2].

Currently, IoT are designed to operate with IP [3]. However, IoT networks often contain many resource constrained devices with smaller memory, limited computational capacity and power supply (mostly a battery). Many IoT applications require the devices to operate for longer periods of time and in remote locations with no facilities e.g. forests. Due to power constraints, the IoT devices are equipped with technologies like IEEE 802.15.4 and Bluetooth LE; and hence, they operate with a much smaller MTU than the current MTU used in the Internet. They also incur several other challenges like limited IP address space, while point-to-point connectivity is also heavy for these resource constrained devices. Additionally, security is another critical aspect in many IoT applications and is expensive (induces overhead) to achieve with IP leading to complexity in operation and resource consumption. There is therefore a need for an efficient design for the IoT devices that is scalable, efficient and provides a secure mechanism for communication to gather data for monitoring and/or controlling the devices. Shang et al. [4] discuss many of such issues in detail.

We observed that IoTs are usually studied/researched as separate entities. However, that should not be the case. One has to also consider all the potential issues in the Internet. Isolating them might lead to unforeseen consequences. IoT needs well connected networks to communicate and pass information/control messages to other devices in the network. One might also question, what is the benefit of integrating the Sensor Networks with the Internet? We believe that by integrating the Sensor Networks with the Internet, both the networks can benefit. Primarily, Sensor Networks will benefit from the existing features of the Internet thus incorporating the ubiquitous Sensor Networks into the IoT world. At the same time, we see a scope for the Internet to widen.

Information-Centric Networking (ICN) [5]–[7] is a new networking paradigm that treats content as the first-class entity. Nodes exchange information based on the Names of the content instead of the IP addresses of the end points that request or provide the information. ICN is growing rapidly with a highly active community. Recent works like, Named Data Networking (NDN [5]), MobilityFirst [8], etc., focus on how IoT could benefit from the future Internet architectures. Additionally, it is feasible for these architectures to be deployed in a closed domain like IoT compared to a full-fledged deployment in the Internet. Similar to ICN, in IoT the devices are interested in the content and not their location i.e., IoT’s are information centric in nature. The design of ICN suits fairly well for IoTs. There is no longer a need for maintaining point-point communication. Recent ICN proposals such as NDN adopt human-readable, hierarchically structured Names. The namespace is unbounded and can easily support billions of devices and even more. There is an increased availability of data due to in-network caching. Security is enhanced since ICN embeds security in the content rather than on the communication link.

Many recent works that advocate the use of ICN for IoT focus on aspects such as data retrieval patterns [9], routing [10], caching [11] and architectural changes [12]. The
Internet houses powerful devices that are capable of running the full fledged ICN protocols [13], [14]. However, the Sensor Networks do not need the entire ICN stack and would require a lighter version of the ICN protocols. Hence like recent works [4] we assume the Sensor Networks will operate with lighter versions of the ICN protocols like CCN-lite/NDN-lite [15] since it is one of the most promising future technology.

Traditional cloud based architectures direct user requests for storage and computing to nearby datacenters. However, with increased focus towards low-latency and high bandwidth requirements form applications like high definition video streaming, technologies like Fog computing [16] have emerged. Fog computing is a promising new architecture that utilizes the multitude of end-user/edge-devices to carry out various operations like, storing, computing, controlling, etc. Fog computing can improve the Quality of Service (QoS) experienced by users through deploying the required storage at the edge of the network close to the users. Many applications including but not limited to smart grids, sensor networks, industrial automation have been realized with Fog computing. Deployment of the future architectures like ICN can also greatly benefit with Fog computing as it expedites the process of integration with the existing technologies and exploits the heterogeneity of the devices available at the edge.

Even though the recent works like [10], [12], etc., have focused on solving many different problems of IoT, they have not considered the crucial aspect of the need to integrate the Sensor Networks with the Internet in order to pave a way for them to join the IoT family. Therefore in this work, we address this vital need. The main goal of this work is to exploit the benefits provided by Fog computing to connect the multitude of Sensor Networks to the Internet. In this work, we propose FOGG: A Fog Computing Based Gateway to Integrate Sensor Networks to Internet. FOGG is a Fog computing based IoT gateway. Its primary focus is to bridge the Internet (running on IP, ICN or other future protocols) with the IoT domains running ICN protocols. We look at the various requirements for realizing such a Fog based – IoT=Internet=SN design. FOGG introduces Fog Gateways to integrate the Sensor Networks with the Internet. To this end, FOGG is designed to provide Fog based services such as name/protocol translation, security (secure onboarding), controller functionality and etc. In this preliminary work, we focus on Fog based name/protocol translation service. The contributions in this paper include:

- **Analysis of the requirements to integrate Sensor Networks with the Internet.**
- **FOGG design:** To integrate Sensor Networks to the Internet with Fog computing.
- **Communication protocol for the IoT networks.**

The rest of the paper is organized as follows: §II identifies potential usecases and derives requirements from them. §III discusses the related work and §IV discusses the proposed FOGG design. §V realizes the usecases identified in §II with the proposed FOGG design and §VI concludes the paper.

II. USE CASES AND REQUIREMENTS

In this section we formalize the essential use cases and derive the requirements for integrating Sensor Networks to the Internet. They further assist in building FOGG.

A. Use Case Formalization

We consider the example of a smart industry with many divisions equipped with Sensor Networks to build our use cases. Let us consider the various applications that an industry can house e.g. monitoring the temperature of equipments like boilers, taking readings, sensor based actuators, etc., as representative Sensor Networks. The smart industry has several equipments, each equipped with a sensor device. There is a Base Station Controller (BSC) that gathers the inputs like temperature readings and gives instructions for operating the actuators from each room once in every 30mins. The BSC raises an alarm or notifies the administrator when abnormalities occur. Further, the Sensor Network1 (SN1) constitutes the sensor devices operating in the division1 and Sensor Network2 (SN2) constitutes the sensor devices operating in the division2 of the industrial unit.

**Scenarios:** We can consider three important use case scenarios in our smart industry example. In the first scenario, A user in the Internet is interested in the temperature of equipment1 in division1. In the second scenario, a sensor equipment E3 in SN1 needs some content e.g. the accurate viscosity of the liquid it boils from the equipment E3 in division2 in SN2. Finally, in the third scenario, all the sensor devices need to transfer their readings periodically to the cloud for storage and further processing.

B. Requirement Analysis

We assume that Sensor Networks operate with a lighter version of the ICN protocol like the CCN-lite (or NDN-lite) while the Internet can operate with IP and/or ICN protocols like the NDN. We choose NDN as a representative for ICN, similar requirements will apply to other future architectures like MobilityFirst [8], XIA [17], etc.

**Gateway:** We observe from the use cases, there are multitude of users scattered across hybrid Sensor Networks and Internet. However these networks do not follow the same protocol. An efficient way to interface networks operating different communication protocols is via a Gateway. The Gateway should run all the protocols of the networks it serves.

**Naming:** We observe from the use cases a need for a naming schema for the IoT devices that is more catered towards the operation of the constrained devices in the Sensor Networks. The rationale for this requirement is, the MTU in the Sensor Networks is much smaller (127B) compared to the MTU (1280B) used by the devices in the Internet. There is a need for smaller ICN names that can not only fit into the MTU of the sensor devices but also requires less storage on the forwarding engines of the sensing devices. This will ensure efficient use of resources and scalability in IoT.

**Communication Protocol:** We observe from the use cases that, unlike IP, the users in Sensor Networks are interested in...
the content similar to ICN irrespective of their location. With an efficient communication protocol we can ensure efficient utilization of the constrained resources in these networks. **Security:** Security is a greater concern in many if not all of the Sensor Networks and Internet. There has been a growing concern that IoT are designed without addressing many of the associated security concerns [18]. However, security induces additional overhead especially in the Sensor Networks. There is a need to analyze and provide some security measures that meet the security requirement of the Sensor Networks and are not an overkill with regards to their constrained resources. **Controller:** The Sensor Networks generate a lot of raw data that needs further processing. However, the sensor devices are heavily constrained in resources and cannot perform such operations. Additionally, transferring this raw data for computation and storage in cloud is necessary but time consuming. This can prove to be expensive for the sensor devices as the cloud might be operating with another protocol. Further, in latency sensitive applications like in the industry, if the computation has to be carried out in the cloud to raise an alarm for a fire it might lead to heavy damage and even risk lives. Additionally, the increasing amount of data generated by the Billions of hybrid IoT devices is already a concern in the Big Data community [19] along with the increasing amount of mobile data traffic [20]. However, most of the data generated by these devices is redundant and can be eliminated with some pre-processing before storing in the cloud. There is a need for a dedicated controlling node equipped with the necessary intelligence and data structures that can handle complex operations on behalf of the resource-constrained Sensor Networks. The controller node should also perform a near transparent flow of traffic between the Sensor Networks and the clouds.

III. RELATED WORK

In this section we mention some of the works that focus on Fog computing, Edge computing and ICN for IoT.

**A. Fog Computing**

We can divide the related work into two categories: ICN+computing which includes technologies used with ICN for computation and Edge/Fog computing.

**ICN+computing:** The authors in [21] show that cloud computing needs a stable infrastructure and networking. They explore an ICN protocol NetInf [22] for supporting networking and storage in cloud computing. They analyze the benefits that could be exploited by the cloud computing from utilizing a name based networking infrastructure especially for management, storage and deployment in dynamic networking environments. While Authors in [23] mention that with the increased amount of information exchange in modern day devices, Fog computing and ICN can help in managing the cloud environment. They propose a framework of ICN as an API for ubiquitous computing. They leverage the caching from ICN and use Fog computing with names instead of IP addresses. They incorporate Fog in the content store (cache) in ICN at the edge.

**Edge/Fog computing:** The authors in [24] explore Fog computing for supporting IoT. They focus on the IoT applications and services and how Fog computing can be used for them. Authors in [25] discuss the need for edge computing with the increasing popularity of IoT and cloud services. In [26], authors propose an architecture named NetFATE that places VNF at the edge of the network.

B. ICN based IoT

We can broadly classify the ICN based IoT related works into four categories namely, Architectural, Routing/Caching, Protocol and Security.

**Architectural:** The authors in [12] propose an initial high-level design for IoT using NDN architecture. They divide the NDN layer into two planes: Data plane and management & control plane. The data plane handles query/response while the control plane re-engineers the current NDN routing plane. In [4], the authors analyze the current TCP/IP solutions for supporting IoT. They argue that existing TCP/IP solutions are inefficient and propose that IoT can benefit by using ICN.

**Routing/Caching:** The authors in [10] discuss the shortcomings of CCN protocol for IoT and propose a routing protocol with O(1) and almost no control traffic. They exploit the caching and data path in ICN and show that CCN-lite uses 80% less memory compared to IP. Whereas in [11], the authors study the benefit of caching with ICN for IoT in terms of energy consumption and bandwidth utilization in comparison with IP.

**Protocol:** The authors in [9] focus on a specific type of data retrieval pattern called Multi-Source Data retrieval. They argue that current NDN architecture does not support this type of communication. In the proposed solution consumers use multi-source interest to retrieve data from multiple producers. They propose to delete the PIT entry based on parameter like Interest life time (TTL). Whereas in [27], the authors study the potential for using ICN for Wireless Sensor Area Networks (WSAN) and how ICN for WSAN’s is different from ICN for Internet. They use flat names and continuous Interest to receive data sensed by multiple sensors as multiple sensors in WSAN sense the same data and respond to the Interest.

**Security:** The authors in [28] propose a protocol for authenticating and authorizing new devices joining IoT mesh networks in ICN. They show 87% improvement in communication and 66% reduction in energy consumption compared to ZigBee-IP solutions. While authors in [29] compare two approaches based on Asymmetric and Symmetric key encryptions for deploying new IoT devices in ICN.

All of the related works discussed above leverage the benefits of either Edge/Fog or ICN or both for supporting IoT. Similar to these works FOGG also exploits the benefit of ICN and Fog computing for supporting IoT. In fact, FOGG extends these works by providing an avenue to connect the Sensor Networks to the Internet to realize IoT through a dedicated Gateway. FOGG utilizes the features of Fog computing and provides improved gateway functionality as we will see in next sections.
IV. FOGG DESIGN

In this section we describe the FOGG design shown in Figure. 1 for integrating the various hybrid Sensor Networks with the Internet using Fog computing.

The Internet could be running on IP or future protocols such as the ICN protocol while the different Sensor Networks that represent various applications like environmental monitoring, smart houses, etc. could be running on lighter versions of IP, ICN or other protocols. The users are spread across both the Internet as well as the Sensor Networks. The aim of FOGG is to let the devices operate in the Sensor Networks as they desire but extend their availability and control by integrating them with the Internet by introducing a Fog Gateway in the Sensor Networks that provides the necessary intelligence and processing capability. The design allows the users in the Internet to access/control the IoT devices using the Internet. A key to achieve this integration is through Fog Gateways at the edge of the Sensor Networks. Below, we provide details on how FOGG could leverage the computing provided by Fog nodes to better support the integration of different Sensor Networks with the Internet.

A. Controller Functionality

In FOGG, we envision a powerful node that provides the functionalities of a Gateway. We call this node Fog Gateway. In addition to the functionalities of the gateway, the Fog Gateway is also capable of performing the tasks similar to that of a controller. FOGG can collect the data periodically from the sensor devices in the Sensor Network and can respond to the queries from Internet and other Sensor Networks. Similarly, Fog Gateway can also receive control messages from other networks and authorize and carry out the execution of controlling the devices.

B. Protocol Translation

The Fog Gateway is a powerful component that has many roles to play. Every Sensor Network is associated with one or more Fog Gateways that sits at the edge of the Sensor Network and is responsible for seamlessly integrating the respective Sensor Network with the Internet. The entire traffic between these two worlds will flow through the Fog Gateway and is transparent to the users in both the networks. The Fog Gateway runs the protocols used by both the Internet and the Sensor Networks in order to optimally translate between the two and in some cases behave transparent to the communication.

The Fog Gateway maintains a mapping table to map the source/destination addresses or content names (in case of ICN) to facilitate communication between the two sides. Moreover, even when both the Internet and Sensor Networks run the same protocol, e.g. ICN, FOGG will support the mapping of the lengthy, unbounded names that could be used in the Internet running an ICN protocol such as NDN to their equivalent short names in the Sensor Networks running a lighter version of ICN such as the CCN-lite protocol. It is clear that since the IPV6 MTU is 1280B while the IEEE 802.15.4 can support only 127B, the authors in the IETF standard [3] suggest that header compression in IPV6 is unavoidable. However, the content generated in the Sensor Network is assumed to be small. FOGG could therefore play an active role in header compression too.

C. Communication Techniques

There are two important modes of communication in the Internet and IoT: Query/Response (Q/R) and Publish/Subscribe (pub/sub). The Fog Gateway can support both forms of communications. For Q/R the Fog Gateway will perform the protocol translation between Internet protocol like NDN to the CCN-lite protocol in Sensor Networks. Whereas for pub/sub, the Fog Gateway can assist in maintaining long term subscriptions. When a publication packet reaches the Fog Gateway it will perform the protocol translation into respective protocol of the target network and forward the publication packet to the subscribers.

D. Secure Onboarding

The Fog Gateway speeds up the deployment of new devices in the Sensor Network by automating the configuration to a large extent. The Fog Gateway uses a registration procedure for every device in the Sensor Network for secure-onboarding. Each sensor device upon entering the Sensor Network must register itself with the Fog Gateway. The Fog Gateway provides an ID to each newly added sensor device. Since in many applications the devices tend to sleep, based on the location of the device, the controller in the Fog Gateway sends information about sleeping patterns to the device for configurations to ensure a stable network at all times. This is essential as in a multi-hop network, all the intermediate devices should be awake simultaneously to ensure successful communication. In case of ICN, the sensor device also registers the short name and long name of the content that it wishes to serve. These entries will be added to the mapping table maintained in the Fog Gateway. The mapping table should be updated whenever there are any changes in the content served by the sensor devices.

E. Data: Pre-fetching, Caching, Filtering

Further, the applications interested in the data generated by the sensor devices need the data to be processed before use. Due to the controller intelligence available at FOGG, it
could proactively pre-fetch and cache content that would be required in the near future. Further, the Fog Gateway could perform complex operations on the data such as filtering of raw data and storage of the processed data on behalf of the resource constrained sensor devices. The enormous amount of data produced by the IoT devices [30] is already a concern for Big Data. However, not all of the data generated by the sensor devices is important to be transmitted and stored. e.g., the devices that periodically sense the temperature of an equipment need to store only the abnormalities. Fog Gateway can greatly reduce the required storage by pre-processing to identify the important data for storage.

F. Security

Since, security is also a major concern, FOGG satisfies this requirement by utilizing ICN security features which provides data-centric security instead of securing the end to end communication channel like in IP. If the devices cannot handle the computational complexity of generating a signature for the content they produce, during registration the devices can receive a signature from Fog Gateway for the content they produce.

Although we speak about one Fog Gateway between the Internet and each Sensor Network, there is no restriction on the number of Fog Gateways. As the traffic exchange between the Internet and Sensor Networks increases, the burden on a single Fog Gateway also increases. Hence, multiple Fog Gateways have to be used to form a Fog network and distribute the load.

V. REALIZATION OF USE-CASES WITH FOGG

In this section, we present examples of how FOGG could support the various use-cases resulting from the scenarios we defined in §II-A. Consider the example of a smart industry with the FOGG shown in Figure. 2. There is a Fog Gateway in each Sensor Network that collects the readings sensed by the devices periodically every 30 mins. The Fog Gateway also receives control messages and forwards it to devices in the Sensor Network. Let us assume that the Internet is running on top of NDN and the IoT network is running on top of CCN-lite. The Fog Gateway therefore runs both the NDN and CCN-lite protocols for interconnecting the Sensor Networks and the Internet.

A. Operation

We use the term Inbound traffic for the traffic entering the Sensor Network and the term Outbound traffic for the traffic leaving the Sensor Network. The inbound traffic from the users can be either a request for data or a request containing a control message. There are two possible kinds of outbound traffic. One containing the reply to the inbound traffic and the other is the request traffic generated inside the Sensor Network. The two types of outbound traffic should be distinguished from one another as the reply traffic needs a name change through a mapping table lookup. This can be easily achieved by using any one bit available field in the packets.

When the Fog Gateway receives inbound traffic it is basically a NDN Interest packet. The Fog Gateway scans the mapping table to find the equivalent short name. Since the Fog Gateway periodically collects the data from the sensing devices, the Fog Gateway checks if it already has the data and returns the data if available. Otherwise, the Fog Gateway creates a CCN-lite Interest packet with the short name and forwards it to the Sensor Network. Upon receiving a CCN-lite Data packet from the Sensor Network, the Fog Gateway performs a lookup in the mapping table to find the equivalent long name and creates a NDN Data packet with the long name, extracts the content from the CCN-lite Data packet and inserts it into the NDN Data packet and forwards it in the Internet. In case of the outbound traffic generated inside the Sensor Network, the Fog Gateway merely performs a protocol translation by generating a NDN Interest with the same content name as in the CCN-lite Interest. Upon receiving the data it converts it to a CCN-lite Data packet and forwards it to the Sensor Network.

B. Example of FOGG with Caching and Periodic-fetching

A user (U1) in the Internet is interested in the temperature of equipment1 in the division1 at 12:30. So, U1 will generate an NDN Interest with the long name e.g. /temperature/industry/building1/division1/equipment1/16-5-17/12:30. The network will forward the Interest to Fog Gateway1 (FGW1). FGW1 will check its mapping table for the name. When no match is found, the FGW1 behaves like a router and the NDN module running inside the FGW1 will forward it to the appropriate router in the Internet. If there is a match, since FGW1 periodically collects all the data, if the FGW1 has the content then it will generate the NDN data with the requested content and forward it to the Internet. Otherwise, it generates a CCN-lite Interest with the equivalent short name e.g. /temp/b1/d1/e1/16-5-17/12:30 and forwards it to the device E1 in the Sensor Network1. Upon receiving a CCN-lite data, the Fog Gateway will scan its mapping table to find the short name. If there is a match the FGW1 will extract the content from the CCN-lite Data packet, generate an NDN Data packet with the equivalent long name with the extracted content and forward it to the Internet. When there is no match found in the mapping table the FGW1 will discard the Data packet.
C. Example of FOGG performing Protocol Translation

The sensor device E3 in the SN1 needs the viscosity of the liquid it boils from the sensor device E3 in SN2 operating in division2. It generates a CCN-lite Interest with the long name e.g. /viscosity/industry/building2/division2/equipment3 and forwards it to FGW1. Please note that since this is an outbound traffic for content located outside the Sensor Network this should be indicated by setting any available bit in the Interest packet. For CCN-lite the EXCLUDE field can be used to indicate this. Upon receiving this Interest, the FGW1 identifies it as an outbound traffic by inspecting the one bit field. The Fog Gateway will only perform a protocol translation by generating an NDN Interest with the same name and forward it to the Internet. The content may be located in the Internet or also in another Sensor Network. The producer either from the Internet or any other Sensor Network will reply with the content. Upon receiving a Data packet the FGW1 will again perform a protocol translation by extracting the content from the NDN Data packet and generate the CCN-lite Data packet with the content and forward it to E3.

D. Example of FOGG performing Pre-filtering

All the sensor devices in the Sensor Network 1 & 2 sense some data and send this raw data immediately to the Fog Gateway. The Fog Gateway filters the raw data and performs the necessary computations desired by the application that uses this data. If abnormalities are identified, the Fog Gateway immediately notifies the administrator and raises an alarm in case of emergencies like fire.

E. Example of FOGG performing Controlling:

An equipment E2 in division1 has an actuator connected to it which opens and closes the lid for controlling the air pressure inside the equipment. The sensor device attached to E2 sends periodic readings of the pressure to Fog Gateway for monitoring. The Fog Gateway monitors the reading and when the readings are above a threshold, the Fog Gateway sends a control message to the actuator fitted with the E2 to open the lid for exact amount of time computed by Fog Gateway. The actuator upon receiving the control message performs the required operation and sends an acknowledgement of the action performed. If the action was unsuccessful, the Fog Gateway will notify the administrator as a technical error has occurred and a human intervention is necessary to avoid risks.

VI. Conclusion

In this paper, we discussed IoT and their immanent explosive growth in the near future along with 5G. We discussed the shortcomings of current IoT designs followed by an introduction to Fog computing and ICN. We observed that ICN is more suitable for supporting IoT compared IP. We discussed in detail the importance and requirements for incorporating Sensor Networks into the Internet to realize IoT. We analyzed the various requirements to integrate the Sensor Networks and proposed FOGG with Fog Gateways. We described in detail the responsibilities of such a Fog Gateway. We further described the communication protocol for IoT networks. With the help of use cases we described the functionality of FOGG. As part of future work we intend to explore naming, security aspects and develop and demonstrate a working prototype of FOGG.

ACKNOWLEDGMENT

This work was supported by the joint EU H2020/NICT ICN2020 Project (Contract No. 723014, and NICT No. 184).

REFERENCES


47