CSSE7014 Distributed Computing
Assignment 2
Semester 1, 2017

Paul Kogel (44644743), Ramdas Ramani (44743767)

May 6, 2017

Contents

1 Introduction **3**

2 Architectures and Models **3**

3 Common Issues **3**

3.1 Networking 3

3.2 Quality of Service-related Issues 3

3.3 Application Development 4

3.4 Security and Privacy 4

4 Applications **4**

5 References **5**

1 Introduction

Fog computing is a new, exciting computing paradigm [1].

The introduction is clear with several definitions of the computing paradigm under study for comparison. The structure of the report is presented.

2 Architectures and Models

Compare and contrast different architectures and models with examples to back the arguments.

RAM

3 Common Issues

Though still an emerging field, previous research, such as Yi et al.'s "survey on fog computing" [2], has been able to identify multiple potential issues related to fog computing.

In this section, we present the major technical challenges that have been identified by them. In addition, we elaborate on further possible problems related to these issues, and discuss possible solutions. As research in this field appears to be fairly limited, though, the latter aspect is only briefly covered.

We organise the issues that have been found by Yi et al. around the following areas: network, Quality of Service (QoS), application development, and security and privacy.

3.1 Networking

In order for the fog to function properly, the network has to provide nodes with connectivity, and additional services, such as routing. Due to the specific characteristics of the fog, though, efficiently providing these functions is considered challenging [2].

In contrast to more traditional networks, for example, routing cannot be implemented centralised in a specific subset of nodes. Instead, as each node acts as router, a decentralised routing mechanisms must be used. Due to the mobility of nodes, this mechanism has to be able to deal with frequent changes in the network topology. Real-time requirements of many applications, and the prevalent use of unreliable wireless links make the fast and reliable delivery of packets even more difficult.

In their proposal of a general architecture for the Fog, Bonomi et al. [3] state that the fog uses virtualisation for "key resources", including networking. As example, they suggest Software Defined Networking (SDN). As pointed out by Peng et al. [4], however, SDN does not put "a high emphasis" on distribution. Similarly, Yi et al. state that the "design [of a] distributed SDN system that meet the harsh requirement of fog computing" is still an open issue.

3.2 Quality of Service-related Issues

As described in XX, fog nodes are mainly connected by unreliable wireless links. Availability of resources at these nodes, and capacity of the links can greatly vary across the network. In addition, as the network is highly dynamic, these properties are subject to constant change. Providing nodes with a specific quality of service is therefore challenging.

In the following, we define several possible dimensions of QoS, and discuss techniques to achieve high values for them.

According to Tanenbaum and Steen, QoS comprises "timing (and other nonfunctional) requirements" [5]. Based on Yi et. al's assessment of quality of service-related issues for the fog, we identify the following basic requirements:

- Service availability
- Resource usage

- Delay
- Reliability

"Service availability" describes the availability of specific services to a node. According to [2], this is greatly influenced by the network topology. This makes intuitively sense: if a node is, for example, connected to a router over an unreliable link, it will probably not be able to access network services at all time.

Naturally, resources as bandwidth and storage are limited in the fog. [2] suggest that placing data effectively can help addressing this issue. In addition, they discuss several techniques for "computation offloading", i.e., the movement of tasks from resource-limited devices like mobile phones to more powerful nodes that are located in the fog, or the cloud. This offloading, however, introduces several new issues. Mainly, these are centred around the difficulty of deciding which parts of the application should be offloaded, while accounting for the constant changes in the network due to node mobility.

Since many applications for fog computing are set to process data in real-time, maintaining a low delay is critical. The aforementioned techniques of data placement, and choice of network topology can be used here.

Finally, to improve reliability, [2] suggest that replication might be used. Traditional techniques like checkpointing, or rescheduling, in contrast, are argued to introduce much delay, and are therefore unfit for real-time applications.

3.3 Application Development

As stated in section XX, the fog is dynamic in regards to network topology, and resource availability. In addition, fog nodes might by based on different platforms and system architectures. Resulting from this, developing applications in a fog computing context is expected to be challenging [2]. To counter this issue, Bonomi et al. [3] propose a "fog abstraction layer" that hides the underlying heterogeneity, and provides developers with a "uniform and programmable interface". Yi et al. make a similar suggestion by calling for a "unified interfacing and programming model" [2].

Though the necessity of these abstraction layers is obvious, we expect that the initially mentioned characteristics of the fog will make an actual implementation highly challenging. Moreover, if devices from different vendors are set to work together, an industry-wide standard has to be developed.

3.4 Security and Privacy

Many applications that have been proposed for fog computing are safety-critical, and/or process sensitive data. For example, in vehicle-to-vehicle communication, an insecure system that allows attackers to remotely control the car could have disastrous consequences. In home automation, users might be worried that unwanted parties could learn about their daily routine.

Stojmenovic and Wen [6] find that providing authentication throughout the system is one of the "main security issues" for fog computing . As an example, they describe a smart meter that is modified by a user, and reports then, due to a lack of authentication, false readings. As a possible solution, they suggest encryption at node-level. For this, the meter would encrypt its data, and another node would decrypt it before further forwarding the data.

Encryption could also be used to protect the user's privacy.

4 Applications

Various examples (across different disciplines) provided with clear arguments why they are relevant.

5 References

[1] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the internet of things," in *Proceedings of the first edition of the MCC workshop on Mobile cloud computing*, pp. 13–16, ACM, 2012.

[2] S. Yi, C. Li, and Q. Li, "A survey of fog computing: concepts, applications and issues," in *Proceedings of the 2015 Workshop on Mobile Big Data*, pp. 37–42, ACM, 2015.

[3] F. Bonomi, R. Milito, P. Natarajan, and J. Zhu, "Fog computing: A platform for internet of things and analytics," in *Big Data and Internet of Things: A Roadmap for Smart Environments*, pp. 169–186, Springer, 2014.

[4] M. Peng, S. Yan, K. Zhang, and C. Wang, "Fog-computing-based radio access networks: issues and challenges," *IEEE Network*, vol. 30, no. 4, pp. 46–53, 2016.

[5] M. Van Steen and A. S. Tanenbaum, *Distributed Systems*. Pearson Higher Education, 2013.

[6] I. Stojmenovic and S. Wen, "The fog computing paradigm: Scenarios and security issues," in *Computer Science and Information Systems (FedCSIS), 2014 Federated Conference on*, pp. 1–8, IEEE, 2014.