A SOFTWARE DEFINED BASED NETWORK INFRASTRUCTURE FOR THE SMART GRID

MOHAMMAD HOSSEIN YAGHMAEE

[1] Ferdowsi University of Mashhad
Email: [1]hyaghmae@um.ac.ir

Abstract - The Smart Grid (SG) which has been known as a big evolution in the traditional power network, provides different applications and services such as: Advance Metering Infrastructure (AMI), Demand Side Management (DSM), Electric Vehicles (EV), Distributed Automation (DA) and Energy Management Systems (EMS). Each applications has unique traffic patterns and needs different certain amount of Quality of Service (QoS). Software Defined Networking (SDN) is a major trend in the telecommunication industry today. In the SDN, the control and data planes in each network equipment (NE) is separated and logically centralized. In this case, each NE just forwards the traffic and enforces policy according to instructions received from the controller. In this paper we present a SDN based communication infrastructure to manage and control different traffic flows in the smart grid, centrally

Keywords - Smart Grid; Software Defined Networking; Virtualization.

I. INTRODUCTION

The Smart Grid (SG) has been recently developed and utilized in many power system around the world. Distinctive types of applications in the smart grid, especially in Neighbor Area Network (NAN) and Home Ares Network (HAN) have been widely used. This last mile network is made of highly limited devices interconnected by fairly unstable low-quality links that cause different Quality of Service (QoS) requirements, which is not the same as the traditional IP networks [1]. Also, the QoS is an essential component of the overall architecture in the smart grid [2]. Some data, such as alerts or control signals, have real-time requirements.

Therefore, the networking infrastructure should somehow guarantee the quality of service, for example, decrease the end-to-end delay. Software Defined Networking (SDN) is a major trend in the telecommunication industry today. In the SDN, the control and data planes in each network equipment (NE) is separated and logically centralized. In this case, each NE just forwards the traffic and enforces policy according to instructions received from the controller. This makes the network programmable in a way that promises to be more flexible than the currently managed paradigm. In software-defined networks, the interface between applications and networks has been changed.

This makes it more suitable for developing different applications in the smart grid which need a higher degree of network awareness. Ease of configuration and management, cross-domain content-based networking, and virtualization and isolation are some opportunities of SDN in smart grid networks [3]. Virtualization of Sensor Network (VSN), on the other hand, is a quite new research approach. A virtual sensor network is formed by providing logical connectivity among a subset of sensor nodes that are dedicated to a certain task or application at a given time [4]. Virtual sensors are software sensors that are built on top of actual physical sensors. They have some data processing functions for complex queries. These functions combine and process sensed data from a group of heterogeneous sensors [5]. Each virtual sensor is created from one or more physical sensors based on the task they perform. Users can freely create and use virtual sensors as if they owned them. These sensors are accessed on-demand and they are deployed when they are an active request for using data [6].

SDN utilizes the concepts of software based centralized control in the telecommunication networks and can be leveraged in the smart grid communication to respond quickly to changing any communication requirements via a centralized control console. In this paper we present a SDN based communication infrastructure to manage and control different traffic flows in the smart grid, centrally. The structure of the proposed framework is depicted in Fig.1. As it can be seen in this figure, the proposed architecture consists of the following layers:

- **Application layer:** In this layer, different applications and services are developed for the smart grid. SDN controller uses northbound Application Programming interfaces (APIs) to communicate with these applications. This layer helps network administrators to programmatically shape traffic and deploy services.

- **SDN Control layer:** It offers a centralized view of the overall network, and enables network administrators to dictate to the underlying systems how the forwarding plane should handle network traffics in the smart grid. Software-defined networking uses standard southbound APIs to relay information to the underlying network elements. OpenFlow is the first popular standard in the SDN and remains as one of the most common protocols.

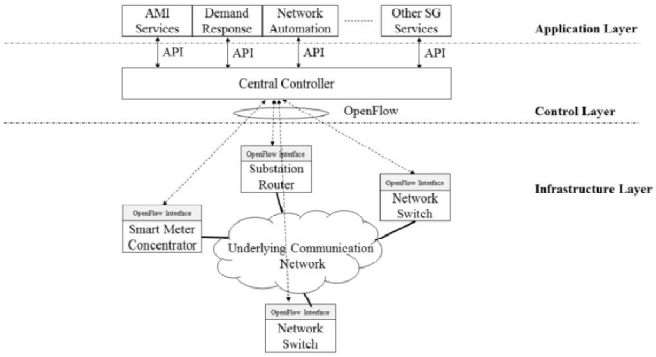


Fig. 1. The proposed SDN based infrastructure for the smart grid communication

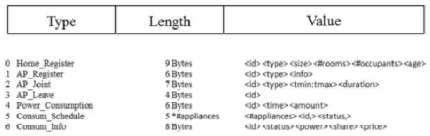


Fig. 2. The structure of DR messages

- **Infrastructure layer:** This layer consists of the network components such as: AMI concentrators, substation router and network switch/routers. Note that these network components are SDN enable which means they have equipped with an OpenFlow interface which provides communication capabilities between central controller and the network elements.

II. PROPOSED ARCHITECTURE

In this section, we describe the components of the proposed architecture. As the demand response programs are based on customer's profile, it is very important to know how many appliances exist in the customer home. In the proposed architecture, there are some SDN gateways at the customer home which provide communication interface between the customer's home and the SDN controller. For this purpose, we define some messages to be exchanged between customers and the SDN controller. The structure of the proposed DR protocol is shown in Fig.2. Each message contains three fields: Type, Length and Value which indicate type of message, its length and the data part of the message.

As shown in Fig.3, the proposed DR message are encapsulated in the TCP/IP packets.

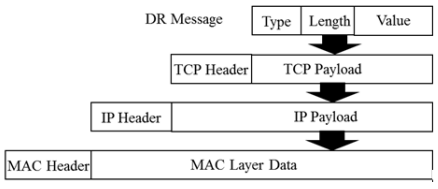


Fig. 3. Message encapsulation

A. Home Appliances Discovery

As the proposed DR program is located at the server side, it is necessary to transfer the number of active appliances and also their energy profile consumption to the server. Each new appliances which is connected to the home network, should be registered in the network. For this purpose, the appliance sends the following message to its local SDN gateway:
AP_Register<id><type><info>

where <id> is a unique identification code for each appliance in the network. Note that <id> consists of two parts, <user id> and <appliance id> which represent the customer and appliance identification, respectively. <type> represents the type of appliances,which will be explained shortly. Finally, <info> represents some power consumption information about that particular appliance. When registration phase is completed, the central controller at the server has complete information about number and type of appliances are currently used in each customer's home. This information is stored in the particular data base. Note that the DR program is based on active appliances which are currently used and joint to the home network. Any active appliance which has been already registered in the system, can joint to the home network by using the following message.

AP_Joint<id><type><tmin:tmax><duration>

As mentioned earlier, <type> represents the type of appliance. We define the following there different appliances which can be used in any home network:
- Fixed: Those appliances which are connected to the local power and consume energy constantly must be kept ON for a certain period of time. Whenever the consumer needs to use these, they should be turn ON and can't be shifted to the other hours. TV, lighting and cooking are some examples of this kind of appliances. Suppose $f_{i,t}$ presents the total load of all fixed appliances of customer $i \in N$ at time slot $t \in T$.

where f_i^{Min} and f_i^{Max} are the minimum and maximum values of all fixed load of customer $i \in N$ at time slot $t \in T$.

- Shiftable non-interrupted: This kind of appliances can be shifted to the other time, but whenever they turned on, they can't be stopped until they finish their jobs. For this type of appliances, the desired operation time is determined by <tmin:tmax>. This means that the appliance should be turn on after <tmin>, and be finished before <tmax>. The duration of operation depends on how much energy it needs. Washing machine and dish washer are some examples of this type of appliances.

- Shiftable interrupted: This kind of appliances (for example PHEV) can be shifted to the other time. The consumer only concern whether the task can be finished within a certain time period <duration>. For this type of appliances, the desired operation time is determined by <tmin:tmax>. This means that the appliance should be turn on after <tmin>, and be finished before <tmax>. The duration of operation depends on how much energy it needs.

Let $\vec{x}_{i,t}$ denotes the amount of shiftable load of customer $i \in N$ at time slot$t \in T$. This shiftable load should be satisfied between$t \in [\alpha_i, \beta_i]$. Let$\vec{E}_i$ denotes the total energy needed for the operation of shiftable appliances of customer$i \in N$. The consumption scheduling should be applied so that the following equation is satisfied.

$$\vec{x}_i^{Min} \le \vec{x}_{i,t} \le \vec{x}_i^{Max}, \quad \sum_{t=\alpha_i}^{\beta_i} \vec{x}_{i,t} = \vec{E}_i \qquad \forall t \in [\alpha_i, \beta_i], i \in N \qquad (2)$$

where$\vec{x}_i^{Min}$and $\vec{x}_i^{Max}$ are the minimum and maximum values of all fixed load of customer$i \in N$ at time slot $t \in T$.

Whenever an appliance is not further needed to be ON, it should be turn OFF. In this case the AP_Leave<id> message is sent. Note that the dynamic changes in the number of active appliances and their current power usage profile is send to the home appliances discovery functions by the home SDN gateway. As will be explained in the next section, the resource allocation function uses this information to dynamically allocate power resource to each consumer.

B. Resource Allocation Function

We consider the power electricity as a resource and try to allocate this resource to all active customers in the region. Consider a region with N different customers. The key idea behind our proposed framework is to control customers' consumption level by defining a specific function to limit customers' allowable usage level in response to their real-time consumption level. Similar in basis but different in method, in some of the selected studies in pricing methods, the authors considered household features to recognized the consumption level of their customers[7-10]. However, in our proposed architecture we also consider some important features of customers to determine the amount of power that is necessary to be allocated to them for their future consumptions. As the proposed energy allocation and pricing is based on customer's profile, it is so important to know the information about each home and household who are living at home. For this purpose, each customer who is interested in participating in the proposed demand response program, should register his/her home in the system by submitting the following message:

Home_Register<id><type><info>

where<id> is the customer home identification in the region and <info> is the other related customer's information.<type> represents the type of home.

C. Personalized Pricing

The Personalized Pricing [11], is the core of our model in which each customer in the grid pays based on his/her consumption level at each time slot $t \in T$. We consider N customers in the grid who consume electricity power and their hourly consumption level are reported to the power utility company to calculate their cost. To do this, we define the following message

Power_Consumption <id><time><amount>

where <amount> represents the total power consumption measured by the smart meter at the end of time slot <time>. In the Personalized Pricing, the cost of each customer is calculated individually. The Personalized Pricingis based on the amount of electricity that is allocated hourly to each customer individually and the amount of power that each customer consumes in each hour during a day. For this purpose, based on the amount of hourly consumption, we classify grid's customers in two groups: good-behavior customers and bad-behavior customers which are defined as follows:

- Good-behavior customer: customer $i \in N$ is called "good-behavior" if and only if his/her hourly power consumption is less than his/her share.
- Bad-behaviorcustomer: customer $i \in N$ is called "bad-behavior" if and only if his/her hourly power consumption is more than his/her share.

For thegood-behavior customers who consume energy less than their share, the price is always low regardless the peak condition in the grid. In the other words, the customer's consumption costs will not be changed if the consumption level is less than the power allocated limits as defined by appropriate customers and to persuade them with satisfying incentives, their consumption costs will not be increased. For the "bad-behavior" customers who consumes more than their allocated power, in the overload condition the cost will increase by a quadratic function.

The current power consumption, power allocation, price and the status of the customer is notified to the customer by the central controller using the following information message:
Send_Info <id><status><power><share><price>

where <id> is the user identification code, <status> is the status of the user (bad-behavioror good-behavior) <power> is the current power consumption , <share> is the customer's share of energyand <price> is the current energy price for that customer. Note that at each customer home, there is an In Home Display which graphically shows the above information to the customer.

D. Virtual networked home energy management (vnhem)

The main idea behind the proposed vNHEM is that instead of using expensive physical NHEM for each customer, develop a virtual central NHEM located at some virtual machines at the cloud servers. By utilizing cloud computing, it becomes possible to gather all consumption related information from all customers in the region and also monitor the network status and resources. The operation of an optimized demand response program requires having all customer power consumption information and power grid resources available. The computing and storage in a cloud facilitates data management and information sharing, among microgrids, end-users, and utilities. Data outage prevention, data management, and protecting user privacy are important security issues in the smart grid which can be solved by utilizing cloud computing.

The vNHEM is in charge of the following two main objectives:

- Find the best schedule for the shiftable appliances so that not only the maximum utility is achieved but also minimum cost is obtained for the customer.
- Determine how much required energy should be consume locally (from the battery) and how much should be bought from the power grid.

The optimal value of power consumption scheduling are informed to the customer by the following message:

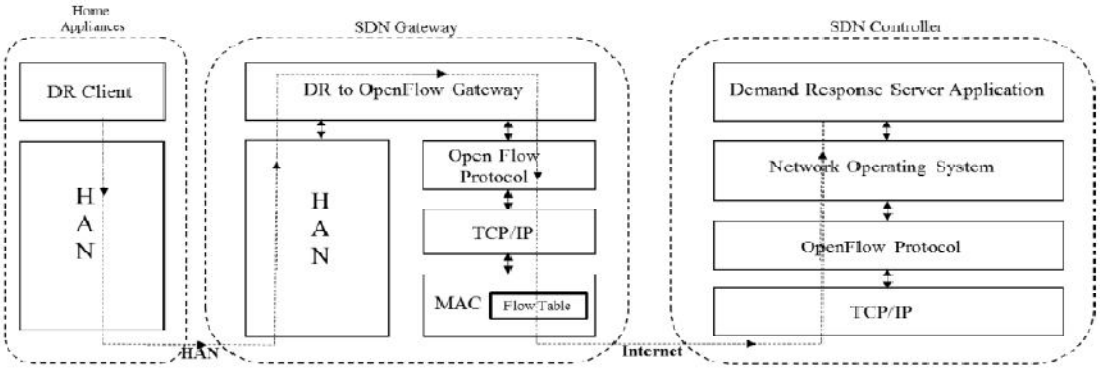
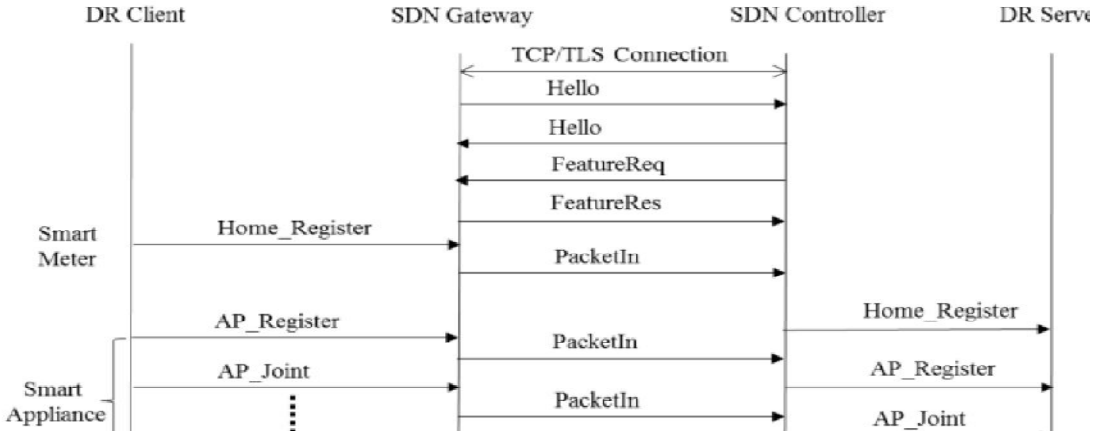


Fig. 4. The protocol stack of theSDN gateway

Consum_Schedule <#appliances><id_i><$status_i$>

where <#appliances> is the number of appliances (including battery), <id_i> and <$status_i$> represent the ith appliances id and its status (ON or OFF).



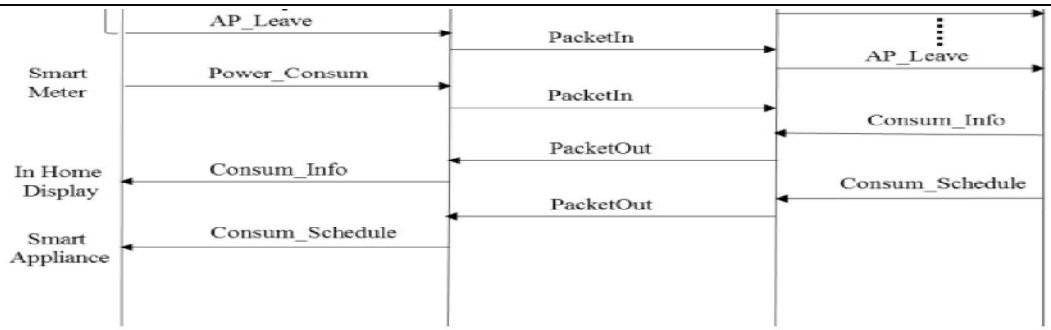


Fig.5. The time sequence of message exchange in the proposed protocol

E. Protocol Stack

All home appliances are connected to the SDN gateway through a Home Ares Network (HAN) protocol such as Zigbee and IEEE 802.15. Each smart appliance is equipped with a HAN protocol stack. At the top of HAN, the proposed DR client service is located. Note that next generation smart appliances can connect to the network and participate in the demand response program. As shown in Fig.4, the SDN gateway has two different protocol stacks. At the left side, it connects to the appliances and smart meter through a home area network. By receiving a DR message from the appliance or smart meter, the SDN gateway encapsulates it into the proper OpenFlow message and forwards it to the OpenFlow protocol for the further actions. The OpenFlow protocol utilizes the TCP/IP protocol stack to forward all messages to the DR server using an Internet connection. In the OpenFlow protocol, the PacketIn message is a way for the SDN switch to send a captured packet to its corresponding controller.

In Fig. 5, the time sequence of messages exchange between DR client and DR server through SDN gateway is depicted. As shown in this figure, at the start of operation, and secure TCP/TLS connection is established between the home SDN gateway and the SDN controller. Then the SDN gateway first submits a Hello packet to alerts the SDN controller, then the first activity is feature determination. The controller sendsthe FeatureReq message to the switch over the transport channel. The switch will then respond to this request with its capabilities. At the home customer side, the smart meter transfers the customer home information to the home SDN gateway, using the Home_Register message. After that, each smart appliance which being used for the first time in the home network should be registered in the system by sending an AP_Regsiter message. Furthermore, each time that an AP is turning ON, it sends an AP_Joint message. All of these messages are encapsulated in the data part of the PacketIn OPenFlow packet and transfer to the SDN controller and DR server. Whenever an appliances is turning OFF, it submits an AP_Leave message which alternatively is encapsulated in and PacketIn packet and be informed to the DR server. After running the optimization problems at the DR server, the consumption information and optimal schedule is sent to the customer home by using Consum_Information and Consum_Schedule messages which are encapsulated in the data part of the PacketOut OpenFlow packet. The consumption information, the current price and the customer status (bad or good behavior) is informed to the customer through these messages. The optimal schedule is sent to the home appliances for further actions.

CONCLUSION

In this paper, we presented a software defined networking infrastructure for the future smart grid. The proposed architecture consists of the different layers including Application layer, Control layer and Infrastructure layer. In the application layer, different applications and services are developed for the smart grid. SDN controller uses northbound Application Programming interfaces (APIs) to communicate with these applications. This layer helps network administrators to programmatically shape traffic and deploy services. The Infrastructure layer consists of the network components such as: AMI concentrators, substation router and network switch/routers. Note that these network components are SDN enable which means they have equipped with an OpenFlow interface which provides communication capabilities between central controller and the network elements. By utilizing the proposed architecture which is an integration of the software defined networking and smart grid communication, the following benefits would be possible:

- Directly programmability of the smart grid network
- Centralized management of all parts of the smart grid network
- Delivering more agility and flexibility
- Enabling innovation in designing smart grid applications.

REFERENCES

[1] N.T. Long, M.-P. Uwase, J. Tiberghien,, K. Steenhaut,"QoS-aware cross-layer mechanism for multiple instances RPL,"

In: Advanced Technologies for Communications (ATC), 2013 International Conference on 2013, pp. 44-49. IEEE

[2] R.B. Abdessalem, N. Tabbane,"RPL-SCSP: A Network-MAC Cross-Layer Design for Wireless Sensor Networks," In: Proceedings of Ninth International Conference on Wireless Communication and Sensor Networks 2014, pp. 27-35. Springer

[3] J. Zhang, B.-C. Seet, T.-T. Lie, C.H. Foh,"Opportunities for software-defined networking in smart grid," In: Information, Communications and Signal Processing (ICICS) 2013 9th International Conference on 2013, pp. 1-5. IEEE

[4] M.M. Islam, M.M. Hassan, G.-W. Lee, E.-N. Huh,"A survey on virtualization of wireless sensor networks," Sensors 12(2), 2175-2207 (2012).

[5] K. T. Lan, "What's Next? Sensor+Cloud?," in Proceeding of the 7th International Workshop on Data Management for Sensor Networks, pp. 978–971, ACM Digital Library, 2010.

[6] M.M. Rutakemwa,"From Physical to Virtual Wireless Sensor Networks using Cloud Computing," International Journal of Research in Computer Science 3(1), 19 (2013).

[7] I. Laicāne, A. Blumberga, M. Rošā, and D. Blumberga, "Determinants of household electricity consumption savings: A Latvian case study," Agron. Res., vol. 12, no. 2, pp. 527–542, 2014.

[8] J. Heinonen and S. Junnila, "Residential energy consumption patterns and the overall housing energy requirements of urban and rural households in Finland," Energy Build., vol. 76, pp. 295–303, 2014.

[9] R. V. Jones and K. J. Lomas, "Determinants of high electrical energy demand in UK homes: socio-economic and dwelling characteristics," Energy Build., vol. 101, pp. 24–34, 2015.

[10] A. Kavousian, R. Rajagopal, and M. Fischer, "Determinants of residential electricity consumption: Using smart meter data to examine the effect of climate, building characteristics, appliance stock, and occupants' behavior," Energy, vol. 55, pp. 184–194, 2013.

[11] Mohammad Hossein Yaghmaee, Mikhak Samadi Kouhi, Alberto Leon Garcia," Personalized Pricing: A new Approach for Dynamic Pricing in the Smart Grid", in the procedding of IEEE SEGE 2016 conference, 22-24 Augest 2016, Toronto, Canada

★★★