Wireless Pers Commun
DOI 10.1007/s11277-015-3063-1

Safe Data Transmission Architecture Based on Cloud for Internet of Things

Seokhoon Kim[1] · **Wonshik Na**[2]

Abstract Nowadays, cloud computing technology, and IoT technology are one of the inevitable core trends. These technologies are key technology of various ICT convergence services, and it will create a value-added business and market. Although it is in its infancy, there is no doubt that it will be a game changer. However, cloud computing technology and IoT technology have some limitations especially in openness and standardization because they have been independently evolved. This is why they expose some weak-points, which are difficult to expand to various services, and it also has some problems to interwork other devices or services. In the safe network case, current safe network service has a restriction because it is based on closed networks. In addition, the key technology in the safe networking has an independent scheme or architecture. Due to the fact that we need new architecture to integrate entire networks, there are no certain alternatives so far. It is because most of related works are focused on the providing to just the safety in the networks. To solve these problems, we propose a safe data transmission architecture for the IoT ecosystems. The proposed architecture is a kind of software defined network base cloud, and it can provide the safe data transmission with QoS/QoE. Basically, the proposed architecture has three states, install, start and working, and four phase, negotiation, initialization, configuration and execution. In the install state, the devices set some parameters such as a synchronization time between node in cloud and IoT device. Based on the install state, data transmission will be started in the start state. In the start state, the nodes start data transmission by the set parameters in the install state. And then, the nodes keep the working state. However, it can be changed whenever user's request to upgrade the

✉ Wonshik Na
winner@nsu.ac.kr

Seokhoon Kim
shkim@cs.ac.kr

[1] Department of Software Engineering, Changshin University, 262, Paryong-ro, MasanHoiwon-gu, Changwon-si, Gyeongsangnam-do 630-764, Republic of Korea

[2] Department of Computer Science, Namseoul University, 91, Daehak-ro, Seonghwan-eup, Seobuk-gu, Cheonan-si, Chungcheongnam-do, Republic of Korea

Published online: 19 October 2015

transmission security. The proposed architecture has some enhanced synchronization algorithms to provide a safe data transmission and to guarantee a QoS/QoE. Based on the architecture, we verified that the proposed architecture outperforms than legacy architectures in various aspects, and we are shown these simulation results by using OPNET.

Keywords Safe network · IoT · Cloud · SDN

1 Introduction

Cloud computing paradigm, that has been strongly promoted by the United States from 2009, is now irreversible trends, and has become a leading computing paradigm of the political, economic, social, cultural and scientific fields. The modern IT services have some important and common features, which should have to support smart computing services and to provide safe use of data. There are no exceptions in these trends. It is not irrelevant to a recent cyber terror and a leakage of private information. Due to the fact that it has a widespread side-effects, the computer network's role is now more and more important. According to the trends, various researches about safe networks are ongoing now, and an intellectual safe network technologies, which can be assigned a network resources dynamically, are also one of the limelight. In these fields, there are hot topics, which we have to immediately solve the problems. For instance, there are IP address hiding routing technology, control and management system technology, protected wireless/mobile technology, safe mobile OS technology, next generation encryption software, enhanced security technology, safe network technology with QoS/QoE, and so on [1–5].

As a result, the safe network technology and related technology have been highlighted. There has been also some various efforts that interwork the safe network technology and the Internet of Things (IoT) in a cloud base computing service. It could be included a Software Defined Network (SDN) technologies. Although the most important thing is to support an efficient and safe data store/transmission/utilization in these trends, it is true that the efforts aren't enough to provide these features until now.

On the one hand, the necessity of cloud base IoT ecosystem implementation are steadily increasing, because Bring Your Own Device and One Person Multi-Device, hand-held devices with various sensors are rapidly deployed. It is mainly caused by the current computing environments which are showing limitation of the existing systems, wasting energy, leaking or losing private information, ineffective management and so on. If the cloud base IoT ecosystems can be successfully implemented in near future, it will be one of the best alternatives which we can take a lot of good effects such as increasing business efficiency, decreasing carbon emission quantity, creating various new business, etc. The most important thing in the implementation of cloud base IoT ecosystems is guaranteeing a service efficiency, continuity, safety, and reliability. Based on the ecosystem, we will meet a new computing paradigm soon [6–9].

Basically, IoT/Machine-to-Machine (M2M) technology is defined as an Internet infrastructure technology, which can interconnect to each thing by using Internet and provide various services. It usually uses Radio Frequency Identification, Near Field Communication, Wireless-Fidelity, Bluetooth, ZigBee, mobile communication technology such as LTE, satellite communication technology and so on. As previously mentioned, the current ICT ecosystem and environment are dramatically shifting to a new paradigm, it is made and will make various platforms. The cloud and IoT ecosystem are placed in the core of the changes. The creation of various platforms have many implications we have to consider. The existing M2M technology, which has been used to achieve a specific goal in a specific scope, is now evolving to IoT technology. In addition, the implementation methods are changing during the evolution periods. In other words, the existing M2M devices usually depend on some specific chipset vendors, the IoT devices, however, are made by using an energy efficiency and cost effective chipsets with standardized design and implementation. Therefore, the IoT devices don't depend on the specific chipset vendors anymore. It can be an example that is preferred to use platform environment in the development process. For a detailed instance, most of the international mobile communication carriers and module vendors usually use "Oracle JAVA ME Embedded" as a development platform [10, 11].

As previously described, there are two big axis leading computing paradigm. One is the cloud computing, and the other is the IoT technology. But, they have common features which have to support a safe data transmission. Until now, although they have been constantly evolved to the next stage, it is not only best time to converge each other, but also best chance to provide a safe network services. To achieve these goals, various researches are ongoing. However, there are still lacks and remain some issues. Among these issues, one of the most important thing is intellectual safe network technology which can provide a safe data transmission with QoS/QoE. This is why we propose the safe data transmission architecture. The proposed architecture use some SDN technology to support safe data transmission. In addition, the proposed architecture is designed for the IoT ecosystems based on cloud. That is because it has a powerful effect than other technologies.

The rest of this paper is organized as follows. Section 2 describes the related works such as SDN, IoT, and safe network. Section 3 discusses the proposed safe data transmission architecture in detail, and Sect. 4 shows the performance evaluations of the proposed architecture. Finally, we offer concluding remarks in Sect. 5.

2 Related Works

2.1 Software Defined Network (SDN)

In modern computing paradigm, there are some keywords we have to be considered, such as cloud computing, bigdata analyzing, high definition video streaming and so on. Due to the fact that the technologies have very different traffic attributes when compare with the legacy traffic attributes of IP networking era, we have been confronted with a new networking era. That is, the adoption of cloud, bigdata, and distributed system have been dramatically increasing the East–West traffic (within switches which have been connected to the servers) than the North–South traffic (between client and server). We need a flexible architecture, which can be dynamically controlled to cope with the paradigm shift, and we usually called that it is a Software Defined Environment (SDE). The SDE includes a Software Defined Compute, Software Defined Storage, and SDN. Among these technology concepts, SDN is the most limelight technology in these days.

Generally, the SDN is a kind of cost-effective network infrastructure technologies by implementing separate control plane and data plane. Based on this scheme, the SDN can support a best path of transmitted data in the entire networks, and provide a flexibility and optimization to the users by using open APIs which can control/manage a physical and logical networks. As is well known, the SDN can be categorized to a hierarchical classification based on an architecture. It can help to easily grasp the element technology of SDN such as application layer, control layer, and infrastructure layer, as shown as Fig. 1. However, it has some difficulties to understand an usage of the technology.

There are three layers, as shown as Fig. 1. First of all, the infrastructure layer consists of various OpenFlow switches which are take charge of packet forwarding or packet manipulation. In this layer, a received packet is classified by a flow, and then the packet is forwarding by defined rule in the flow table. These functionalities are controlled by a command of controller which has a software defined forwarding functions. The infrastructure layer basically supports a OpenFlow standard specification of Open Networking Foundation (ONF), and some protocols, which will be defined by various standardization organizations such as IETF, will be included in near future. Secondly, the control layer is a kind of data plane resource set, and it has a common library form to support various topologies, host metadata, status abstractions and so on. Each network device's connection is configured by a self-organized information. Based on this, it is established a best path for each network traffic, and then this information will be transmitted to a OpenFlow switch in the infrastructure layer. Finally, the application layer is implemented network business requirements such as virtual network overlay, network slice, tenant recognition broadcast, application recognition path/policy/security/traffic engineering and etc. In most cases, the API between the control layer and application layer is supported by the control layer (Fig. 1).

There are various standardization organizations which is related to a SDN technology. The ONF makes effort to establish enhancement and expansion of standard SDN functionality based on OpenFlow protocols. The Network Functions Virtualization Industry Specification Group, and ITU are leading a NFV standard and SDN standard for cooperation of each nations. In addition, the IETF will be defined a SDN standard for legacy devices. Besides that, OpenDaylight, Open vSwitch, and Open vSwitch Database Management Protocol are also one of the important thing, we should be taking note of the SDN standardizations [12–15].

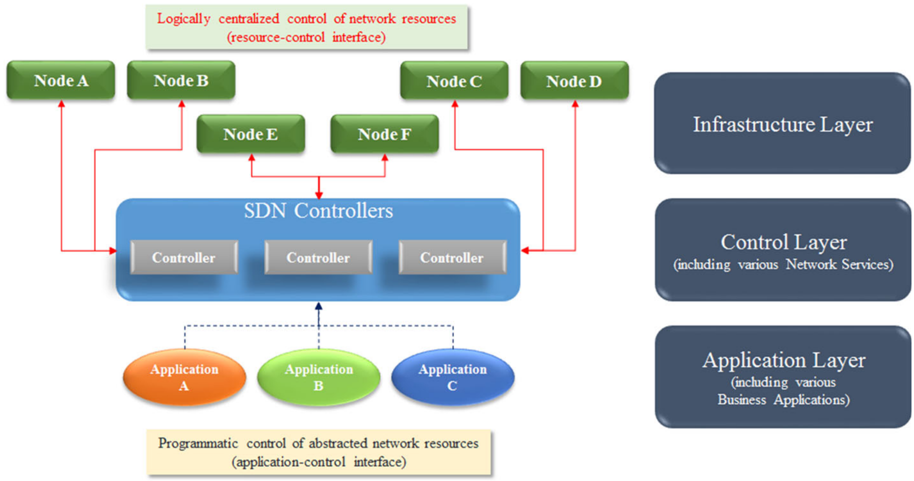
Logically centralized control of network resources
(resource-control interface)
Node A
Node B
Node C
Node D
Node E
Node F
SDN Controllers
Controller
Controller
Controller
Application A
Application B
Application C
Programmatic control of abstracted network resources
(application-control interface)
Infrastructure Layer
Control Layer
(including various Network Services)
Application Layer
(including various Business Applications)

Fig. 1 SDN architecture

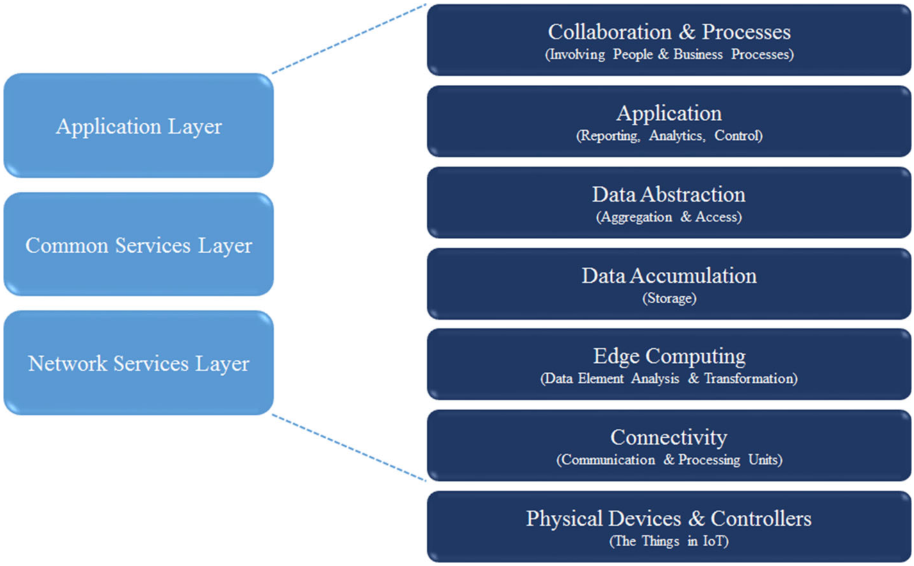
Application Layer
Common Services Layer
Network Services Layer
Collaboration & Processes
(Involving People & Business Processes)
Application
(Reporting, Analytics, Control)
Data Abstraction
(Aggregation & Access)
Data Accumulation
(Storage)
Edge Computing
(Data Element Analysis & Transformation)
Connectivity
(Communication & Processing Units)
Physical Devices & Controllers
(The Things in IoT)

Fig. 2 The oneM2M Layered model and IoT Reference Model

2.2 Internet of Things

Recently, the oneM2M established a technical specification drafts in 2014. In the technical specifications, they define several main features and functionalities. Among these things, the followings are layered model and functional architecture, which are related to the proposed architecture. According to the technical drafts, there are three layers, Application Layer, Common Services Layer, and Network Services Layer. The each layer details are following, as shown in Fig. 2.

- Application Layer: it comprises oneM2M applications and related business and operational logic.
- Common Services Layer: it consists of oneM2M service functions that enable oneM2M applications (e.g. management, discovery and policy enforcement).
- Network Services Layer: it provides transport, connectivity and service functions.

In addition, the technical drafts define an architecture reference model. In these architecture reference model, the functional architecture is followings. In addition, the reference model of oneM2M is very similar to IoT reference model, which was announced by Cisco Systems in IoTWF 2014 (Internet of Thing World Forum 2014).

- Application Entity (AE): Application entity represents an instantiation of application logic for end-to-end M2M solutions. Each AE is identified with a unique AE-ID. Examples of the AEs can be an instance of a fleet tracking application, a remote blood sugar monitoring application, a power metering application, or a controlling application.
- Common Services Entity (CSE): A Common Services Entity represents an instantiation of a set of “common service functions” of the M2M environments. Such service functions are exposed to other entities through reference points Mca

and Mcc. Reference point Mcn is used for accessing Underlying Network Service Entities. Each CSE is identified with a unique CSE-ID. Examples of service functions offered by CSE include: data Management, Device Management, M2M Subscription Management, and Location Services. Such “sub-functions” offered by a CSE may be logically and informatively conceptualized as Common Services Functions (CSFs).

- Network Services Entity (NSE): A Network Services Entity provides services from the underlying network to the CSEs. Examples of such services include device management, location services and device triggering. No particular organization of the NSEs is assumed [16, 17].

2.3 Safe Network

Safe network means a network which has a network architecture and equipment. However, the architecture and equipment has some solutions which can solve the structural weak points of IP base networks. In addition, it can efficiently use a legacy network. The element technology of safe networks are as followings.

- Network Hiding Technology: Basically, this technology is one of the core issue in the Future Internet. Due to the fact that the Future Internet is not currently available network, most of network hiding technologies are based on the black core network which use a secure signaling and overlay base network security device. However, these approaches are one of the closed network services, which aren’t applied the real network hiding technology. Therefore, there are various researches and developments to hide a destination IP address or server address in an open networks, including trusted network and untrusted network. Meanwhile, The Onion Routing is very similar research when it comes to compare with the network hiding technology.
- High Trusted VPN Technology: There are a lot of needs about VPN which can support a mobility, expandability, ease of manageability, supporting a multi-domain and so on. However, the legacy simple VPN technologies (e.g.: IPSec, SSL, GRE, L2TP, MPLS VPN, etc.) can’t easily support the above needs. So, the recent researches and developments are focused on to support a real-time user and service recognition, traffic classification/transmission by user and service. One of the representative example is Safe VPN.
- Intellectual Network Technology: The rapid deployment of new services (e.g: cloud service, bigdata service, and etc.) based on a various multimedia contents are very common services in these days. In these services are needed elaborate control to provide high quality of service. Therefore, many global companies such as Cisco Systems, Juniper Networks, Alcatel Lucent have researched various high quality transmission services based on cloud network and enterprise network (e.g: Cloud Intelligent Network, MX series 3D universal Edge, and etc.).
- Identification and Authority Management (IAM) technology: This is one of the inevitable technology for supporting a smart work service. The IAM can assign and distribute a resource to a user by a pre-defined authority. It is usually used to support a functionality of IAM to a mobile device. The representative examples are Tivoli, Safe Authentication Manager, and so on [18–20].

3 Safe Data Transmission Architecture (S-DTA)

As previously mentioned in the above sections, the proposed S-DTA is secure data transmission scheme which is based on a synchronization between source and destination. Basically, the target system of the proposed S-DTA is a network which is based on cloud networks with various IoT devices. In addition, the proposed S-DTA use IAM for secure data transmission and resource allocation. To provide a secure data transmission, the proposed S-DTA makes a synchronization path between source device and destination device. The proposed S-DTA uses a time synchronization method during the synchronization process. This synchronization process is most important process in the synchronization process, because the proposed S-DTA has an on-time bypassing transmission scheme. The proposed S-DTA uses IEEE 1588 protocol as a Precision Time Protocol (PTP) to provide an on-time bypassing transmission scheme. As known as about the IEEE 1588 protocol, the protocol supports high precision and accuracy. That is, it means that the proposed S-DTA with IEEE 1588 protocol provides an on-time bypassing transmission scheme to the devices. Nonetheless, IEEE 1588 protocol has some weak points such as asymmetric uploading and downloading time. However, these weak points can overcome by utilizing a SDN technique, and this is why we use IEEE 1588 protocol as a PTP for the synchronization process in S-DTA [21–24].

Actually, the synchronization concept in the proposed S-DTA is almost same as the Packet fOrwarding Scheme based on Interworking Architecture (POSIA) [25], the QoS-aware Data Forwarding Architecture (QDFA) [26], the Information Exchange Architecture (IEA) [27], and the Efficient P2P Data Forwarding Scheme (EP-DFS) [28]. Actually, the synchronization concept in the proposed S-DTA is almost same as the POSIA, the QDFA, the IEA, and the EP-DFS. However, there are some differences between S-DTA's synchronization process and them. First of all, the proposed S-DTA uses IAM techniques to assign resources which are used to the transmission. Secondly, the proposed S-DTA uses a

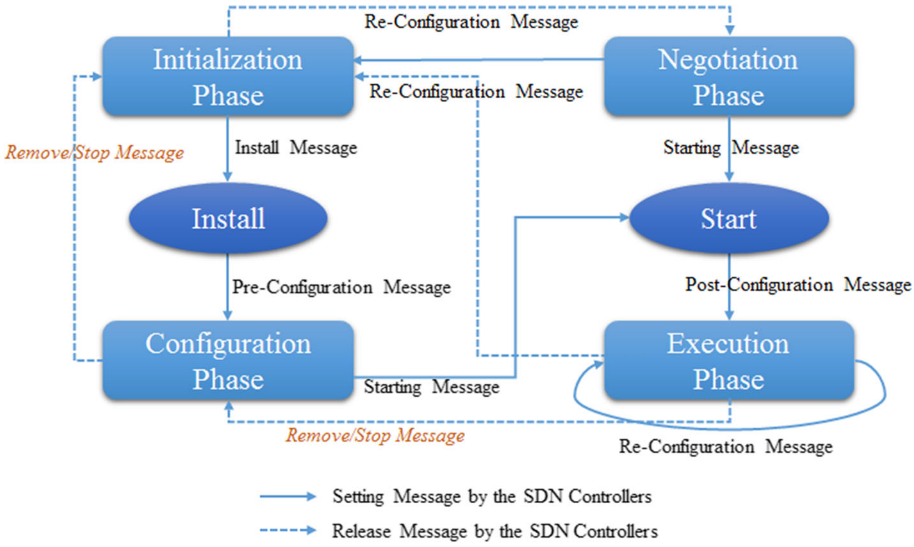

Fig. 3 The S-DTA state diagram

random variable to support a secure data transmission. That is, all nodes in the S-DTA transmit the data at the pre-defined time which has been negotiated by a random variable. It means that the data which are used the S-DTA's transmission path will be ensured to the safe data transmission, because the data transmission time is always changed. Finally, the proposed S-DTA can handle IoT traffic attributes (e.g: required bandwidth, transmission time, session time, and so on). The previously mentioned data forwarding scheme is usually focused on the multimedia traffic attributes. However, the IoT traffic attributes are certainly different to the multimedia traffic attributes (Fig. 3).

Nevertheless, the synchronization procedure is also almost same as previously mentioned schemes, which are POSIA, QDFA, IEA, and EP-DFS. There are some differences in the exchanging parameter values and attributes during the synchronization procedure. That is, we modified the parameter values and attributes to apply various IoT devices for the cloud networks. To achieve this purpose, we set new phase, the negotiation phase, in the proposed S-DTA state, as shown as Fig. 4.

As is well known for IoT device working mechanism, most of IoT devices utilize IP data forwarding scheme to transmit their data. Actually, the real IoT era isn't yet, various IoT devices are deployed and will be deployed. The issue in this era is safe data transmission of the IoT devices. Although IP data forwarding scheme has been extremely evolved than legacy IP forwarding scheme, there are still some lacks is true. Due to the fact that current IP data forwarding scheme is basically based on the data encryption method, it is one of the critical issue about that. That is, an attacker, who knows about the encrypted key or decrypted key, can easily intercept the transmission data whenever they need. To solve this problem, the proposed S-DTA uses a time synchronization to protect the data of IoT devices. In the proposed S-DTA, the transmission time is irregularly set by the SDN controller. In addition, the SDN controller change the IoT device's data transmission time.

The details of the synchronization settings in the S-DTA can be described as follows. The path and the resource reservation are set to have *n* number of switches and routers exist and in this case, the synchronization S-DTA setting parameters that can affect the following.

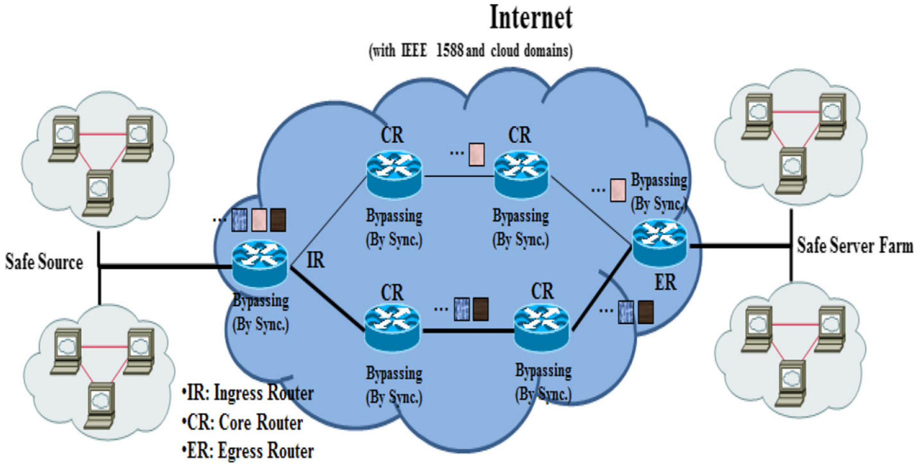

Fig. 4 Performance evaluation topology

- e_0, e_1, …, e_n: Electronic delay. This is the time to go through the circuit in the device. In most cases, this value almost verge on 0.
- p_0, p_1, …, p_n: Processing delay. This is the time to process data in the devices such as end devices, and intermediate devices (e.g: switch, router, computer, and so on). This is main factor to determine the data forwarding performance.
- s_0, s_1, …, s_n: Serialization delay. Packets for transmission over a transmission medium converted into a bit stream. This is the time to convert from analog signal to digital signal or from digital signal to analog signal. Same as the electronic delay, this value almost verge on 0 in most cases.
- t_0, t_1, …, t_n: Transmission delay. Determined by physical laws of the delay time, the transmission medium for sending a signal from the delay time consuming. This is the physical time to transmit the data. Same as the electronic delay and serialization delay, this value almost verge on 0 in most cases.

The Source Device (SD) packet is arithmetically any Destination Device (DD) hop away from the N hop of the incoming queue of DD_{arrive} is expressed as follows.

$$DD_{arrive} = \sum_{n=0}^{N} (e_n + p_n + s_n + t_n) \tag{1}$$

If the expression (1) e_n, p_n, s_n, t_n quantitative value if everyone, *SD* on the path from the *DD* to the synchronization between the routers and destination devices can be easily done. However, in actual network e_n, s_n, t_n is a time value can be expressed as quantified. However, p_n is a packet that passes through the network conditions (congestion, buffering and queuing) varies depending on the quantified value can not be expressed. Therefore, the expression (1) S-DTA can not be used for synchronization.

To solve this problem, S-DTA synchronization settings for setting the synchronization message (*SSSM*: Synchronization Status Set Message) is used. The S-DTA $SSSM_{set}$ used by the *SSSM* and the $SSSM_{init}$ two forms. The synchronization settings in the DD is sent to the *SD*. And $SSSM_{init}$ comprises the following items.

- *HC* (Hop Count): The past number of hops in the path of *SSSM*. The s_n required for synchronization settings used to calculate the parameters.
- *IT* (Interface Type): Sent just before the two-hop router interface in the form of *SSSM*. A router which is connected to the interface is determined according to the type of transmission medium, so the synchronization required for setting the parameters used to calculate the t_n.
- $E_{tolerance}$: Set a tolerance value for each hop in the hop set in the tolerance of each value of the accumulated value. $SSSM_{set}$ was created in *DD* return used to check the synchronization settings.

In expression (1), e_n, s_n, t_n is a time value, so quantified, in any given hop on the path of the incoming $i - 1$ to i. $SSSM_{init}$ hop to reach the delay time ($SSSM_{init\_per\_hop\_delay}$) is as shown in expression 2 expressed.

$$SSSM_{init\_per\_hop\_delay} = e_{i-1} + p_{i-1} + s_{i-1} + t_{i-1} \tag{2}$$

IEEE 1588 is the time stamp of the packet field value is easily calculated using $SSSM_{init\_per\_hop\_delay}$. p_{i-1} can be easily derived, the derived value of i set time hop synchronization ($Sync_{init\_hop\_set}$) to the expression (3) can be derived as be. At this time,

$E_{tolerance\_hop}$ (Error tolerance) is set to be within the scope of the $e_{i-1} + IT_{i-1} < E_{tolerance_{hop}} < p_{i-1}$.

$$Sync_{init\_hop\_set} = SSSM_{init\_per\_hop\_delay} - p_{i-1} + E_{tolerance_{hop}} = C_{hop} + E_{tolerance_{hop}} \quad (3)$$

In any given hop on the path derived from $Sync_{init\_hop\_set}$. The i has the value of the synchronization settings. Similarly, the use of the S-DTA other routers on the path through the process is also the same as described above to the general synchronization settings.

As previously mentioned to the paragraph, the proposed S-DTA has same synchronization concepts to POSIA, QDFA, IEA, and EP-DFS. However, the proposed S-DTA has an expanded synchronization path between source device and destination device. In other words, the proposed S-DTA makes a end-to-end synchronization path to transmit a data. This is most differnet point than the others. Based on these schemes, the proposed S-DTA supports a secure data transmission. In addition, these functionalities are controlled by the SDN controller in the path. Although the above expressions aren't indicated a detail synchronization time set method, it has basically long time range than the other because the S-DTA is an architecture for the various IoT devices.

4 Performance Evalutions

The performance evaluations of the proposed S-DTA are focused on the throughput of each data forwarding scheme, because the throughput attribute is one of the best attribute to compare with the each scheme's performance. So, We have implemented a network topology to evaluate the performance of the proposed S-DTA by using the OPNET, as shown in Fig. 4. Although there are no detailed descriptions about the synchronization in the networks of Fig. 4, the safe networks have the synchronized path which is configured by the S-DTA synchronization procedures. This is big different point than the others. That is, this simulation topology has end-to-end synchronization path between source device and destination device. In these performance evaluations, we compare the S-DTA with DiffServ, IntServ, POSIA and GMPLS which are the best schemes among the current packet forwarding techniques.

In the source side (Safe Source in the Fig. 4) of the network topology, there are 100 safe source nodes, the destination side (Safe Server Farm in the Fig. 4) of the network topology

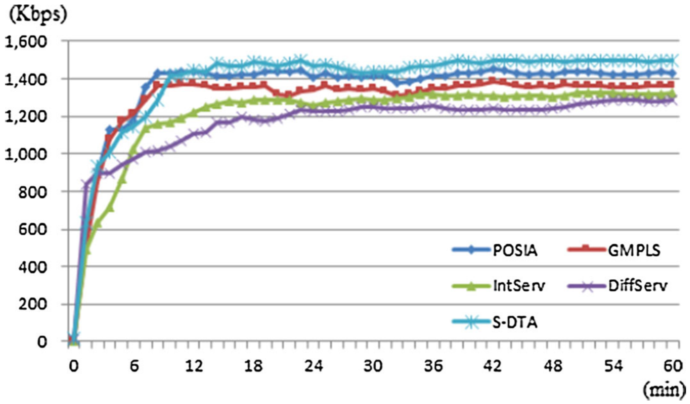

Fig. 5 Throughput comparison of the S-DTA with others

has 6 safe servers. Basically, we use the streaming traffic types in the simulations because the streaming traffic types have high sensitivity to the various parameters such as delay, jitter, and bandwidth. Actually, the streaming traffic attributes might not be suitable for the IoT devices in some aspects, because the traffic attributes don't have any burst traffic features. However, some IoT devices such as CCTV, set top box, are needed to support their functionalities to the users. This is the reason why we have adopted this traffic attribute to the computer simulation.

Figure 5 are the S-DTA and different data forwarding scheme, and the throughput comparison is shown. The Resource Reservation Protocol using the reserved bandwidth is 1500 Kbps, and resource scheduling is One Pass with Advertising model was used. As shown in Fig. 5, the proposed S-DTA has higher throughput than the other data forwarding scheme. It means that the S-DTA's synchronization path can efficiently use the reserved resource than the others. However, the comparison of POSIA don't have remarkable differences, because the POSIA also uses a same synchronization concepts. Although the simulation results of throughput aren't extremely improved than the POSIA's throughput result, we have to take notice of the improved throughput results. That is, the results are improved, even though we are expanded a synchronization path.

Figure 6 is shown the end-to-end delay results between proposed S-DTA and the other data forwarding scheme. The reason why the result of the end-to-end delay is low than the POSIA in the first part, the proposed S-DTA spend the time to make a synchronization path from source device to destination device. Although the performance of the proposed S-DTA is low than the POSIA in the first part, the total performance is better than POSIA, because the end-to-end synchronization path is established. It means that the proposed S-DTA is not only optimized the synchronization process than the POSIA, but also utilized the reserved bandwidth than the POSIA.

5 Conclusions

In past few years, there has been many changes in IT technology and industry. Among these things, the most dramatic alternation has been caused by computing paradigm based on cloud, bigdata, and IoT. This complexed computing environment is always

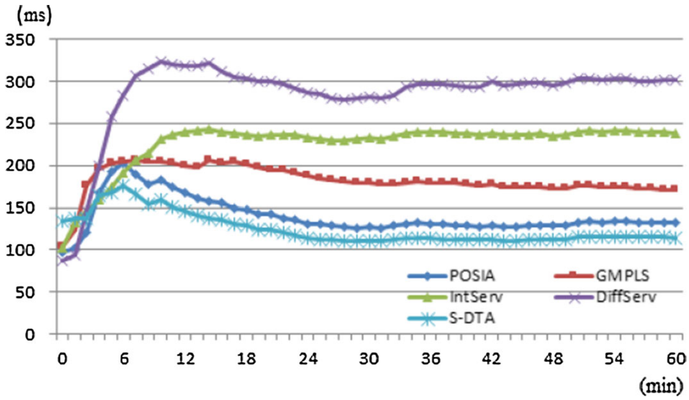

Fig. 6 End-to-end delay comparison of the S-DTA with others

accompanied with various data transmission. Due to the fact that we have to support an efficient and safe data transmission to users and devices. However, we aren't found the best solution yet, because the recent researches and developments have been focused on the deployment and diffusion of the infrastructure and environment.

In this paper, we propose the S-DTA, which is based on the accurate and precise synchronization, for various IoT devices. Based on this scheme, the proposed S-DTA can provide efficient and secure data transmission to the devices in cloud networks. In addition, the proposed S-DTA is adopted various SDN technologies for supporting efficient and secure data transmission. That is, the proposed S-DTA can support the transmission requirements by these features. In the proposed S-DTA, the on-time bypassing data transmission scheme is used a synchronized path and time to forward data. This is a big different thing to the various existing mechanisms. To prove the outperformance of the proposed S-DTA, we performed the computer simulation by OPNET, and we verified the excellence of the proposed architecture.

Although the proposed S-DTA can support various IoT devices, it isn't a customized solution to an light-weight IoT device, because most of light-weight IoT devices don't need a high bandwidth. In addition, due to the fact that the light-weight IoT devices have very light-weight hardware specifications, it isn't easy to apply the SDN technology. That is, although the proposed S-DTA is very efficient and superior aspects in a heavy-weight or fixed IoT device, it has certain limitations to apply to a light-weight IoT device. So, it will be one of the main researches in near future.

Acknowledgments This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2014R1A1A2060035).

References

1. Prasad, A. S., & Rao, S. (2014). A mechanism design approach to resource procurement in cloud computing. *IEEE Transactions on Computers, 63*(1), 17–30.
2. Qu, L., Wang, Y., Orgun, M. A., Liu, L., Liu, H., & Bouguettaya, A. (2015). CCCloud: Context-aware and credible cloud service selection based on subjective assessment and objective assessment. *IEEE Transactions on Services Computing, 8*(3), 369–383.
3. Yin, Z., Yu, F. R., Bu, S., & Han, Z. (2015). Joint cloud and wireless networks operations in mobile cloud computing environments with telecom operator cloud. *IEEE Transactions on Wireless Communications, 14*(7), 4020–4033.
4. Wang, S., & Dey, S. (2013). Adaptive mobile cloud computing to enable rich mobile multimedia applications. *IEEE Transactions on Multimedia, 15*(4), 870–883.
5. Abolfazli, S., Sanaei, Z., Ahmed, E., Gani, A., & Buyya, R. (2014). Cloud-based augmentation for mobile devices: Motivation, taxonomies, and open challenges. *IEEE Communications Surveys & Tutorials, 16*(1), 337–368.
6. Parashar, M., AbdelBaky, M., Rodero, I., & Devarakonda, A. (2013). Cloud paradigms and practices for computational and data-enabled science and engineering. *Computing in Science & Engineering, 15*(4), 10–18.
7. Pandey, V., Singh, S., & Tapaswi, S. (2015). Energy and time efficient algorithm for cloud offloading using dynamic profiling. *Wireless Personal Communications, 80*(4), 1687–1701.
8. Wang, Y., Chen, I.-R., & Wang, D.-C. (2015). A survey of mobile cloud computing applications: Perspectives and challenges. *Wireless Personal Communications, 80*(4), 1607–1623.
9. Kim, S., Kim, G.-J., & Chung, K.-Y. (2014). Performance analysis of non-PC/SC based mini-WiMAX connection manager. *Cluster Computing, 17*(3), 775–789.
10. Xu, B., Da Xu, L., Cai, H., Xie, C., Hu, J., & Bu, F. (2014). Ubiquitous data accessing method in IoT-based information system for emergency medical services. *IEEE Transactions on Industrial Informatics, 10*(2), 1578–1586.

11. Papagianni, C., Leivadeas, A., Papavassiliou, S., Maglaris, V., Cervello-Pastor, C., & Monje, A. (2013). On the optimal allocation of virtual resources in cloud computing networks. *IEEE Transactions on Computers, 62*(6), 1060–1071.
12. Jarschel, M., Zinner, T., Hossfeld, T., Tran-Gia, P., & Kellerer, W. (2014). Interfaces, attributes, and use cases: A compass for SDN. *IEEE Communications Magazine, 52*(6), 210–217.
13. Kyoseva, T., Poulkov, V., Mihaylov, M., & Mihovska, A. (2014). Disruptive innovations as a driving force for the change of wireless telecommunication infrastructures. *Wireless Personal Communications, 78*(3), 1683–1697.
14. Miao, W., Agraz, F., Peng, S., Spadaro, S., Bernini, G., Perello, J., … Calabretta, N. (2015). SDN-enabled OPS with QoS guarantee for reconfigurable virtual data center networks. *IEEE/OSA Journal of Optical Communications and Networking, 7*(7), 634–643.
15. Tomovic, S., Pejanovic-Djurisic, M., & Radusinovic, I. (2014). SDN based mobile networks: Concepts and benefits. *Wireless Personal Communications, 78*(3), 1629–1644.
16. Bontu, C. S., Periyalwar, S., & Pecen, M. (2014). Wireless wide-area networks for internet of things: An air interface protocol for IoT and a simultaneous access channel for uplink IoT communication. *IEEE Vehicular Technology Magazine, 9*(1), 54–63.
17. oneM2M. (2014). *oneM2M functional architecture baseline draft*. oneM2M Technical Specification, oneM2M-TS-0001-V-2014-08
18. Li, Q., Xu, M., Yang, Y., Gao, L., Cui, Y., & Wu, J. (2014). Safe and practical energy-efficient detour routing in IP networks. *IEEE/ACM Transactions on Networking, 22*(6), 1925–1937.
19. Kang, W., Sha, L., Berlin, R. B., & Goldman, J. M. (2015). The design of safe networked supervisory medical systems using organ-centric hierarchical control architecture. *IEEE Journal of Biomedical and Health Informatics, 19*(3), 1077–1086.
20. Kang, K., Pang, Z., Da Xu, L., Ma, L., & Wang, C. (2014). An interactive trust model for application market of the internet of things. *IEEE Transactions on Industrial Informatics, 10*(2), 1516–1526.
21. Ferrant, J.-L., & Ruffini, S. (2011). Evolution of the standards for packet network synchronization. *IEEE Communications Magazine, 49*(2), 132–138.
22. Ouellette, M., Ji, J., Liu, S., & Li, H. (2011). Using IEEE 1588 and boundary clocks for clock synchronization in telecom networks. *IEEE Communications Magazine, 49*(2), 164–171.
23. IEEE. (2011). IEEE standard profile for use of IEEE 1588 precision time protocol in power system applications. *IEEE Standard*, *C37.238*, 1–66.
24. Li, C., & Li, L. (2014). Phased scheduling for resource-constrained mobile devices in mobile cloud computing. *Wireless Personal Communications, 77*(4), 2817–2837.
25. Kim, S., & Ryoo, I. (2010). Packet forwarding scheme based on interworking architecture for future internet. *IEICE Transactions on Communications, E93-B*(3), 546–550.
26. Kim, S. (2015). QoS-aware data forwarding architecture for multimedia streaming services in hybrid peer-to-peer networks. *Peer-to-Peer Networking and Applications, 8*(4), 557–566.
27. Ryoo, I., Na, W., & Kim, S. (2015). Information exchange architecture based on software defined networking for cooperative intelligent transportation systems. *Cluster Computing, 18*(2), 771–782.
28. Kim, S., & Suk, J. (2015). Efficient peer-to-peer context awareness data forwarding scheme in emergency situations. *Peer-to-Peer Networking and Applications*. doi:10.1007/s12083-015-0401-8.

Seokhoon Kim received the B.E. and Ph.D. degrees in computer engineering from Kyunghee University, Seoul, Republic of Korea, in 2000 and 2004, respectively. He currently works as an Assistant Professor at Changshin University, Changwon, Korea. His research interests include mobile IPTV, B4G, cross layer, QoS/QoE, Future Internet, and IoT.

Wonshik Na is a Professor in Dept. of Computer Science at Namseoul University. He received the Ph.D. degree in Computer Engineering from Kyunghee University Korea, in 2005, His research interests are in the Network Security, Wireless LAN, Medical Information, Multimedia System.