Emergency Communication in Disaster Scenario over KubeEdge and NDN

Minh Ngoc Tran, Young Han Kim*
Soongsil University

mipearlska1307@dcn.ssu.ac.kr, younghak@ssu.ac.kr*

Abstract

Accurate and reliable communication between emergency teams when dealing with disaster situations plays a decisive part in an effective disaster recovery phase for any country. However, because network infrastructure is usually heavily damaged when these scenarios happen; establishing a quick, flexible, and scalable network connection for emergency communication is a challenging issue. In this paper, we introduce the use of two promising network technology for edge infrastructure – KubeEdge and Named Data Networking to solve the mentioned task.

Ⅰ. INTRODUCTION

When disasters happen, network infrastructure may be partially destroyed and fragmented. This problem causes a huge challenge for government and related emergency agencies to maintain a reliable communication connection for exchanging disaster information in time for rescuing people. Carrying network devices to disaster areas and creating a temporary network for communication might be a solution. However, hardware dependencies and time-consuming setting up time can be another issue. With virtualization and edge technologies are on the rise, using an edge platform to deploy virtualized network functions instead of setting up real devices is an effective strategy especially in disaster scenarios. In our implementation, we propose to use KubeEdge – a lightweight Edge platform to perform fast and low-cost virtualized network infrastructure installation.

Besides the need of a fast and lightweight deployment strategy, a new network solution instead of IP-based network is also necessary as current network infrastructure shows limitations when dealing with communication in emergency scenarios [1]. Emergency teams usually have massive sub teams, dynamic role changes and high mobility during their missions. This requires network to support scalability, location independency, and built-in mobility which Named Data Networking (NDN) can offers.

In this paper, we present the KubeEdge-NDN system architecture, which shows how these two technologies are integrated to solve current challenges in emergency communication.

Ⅱ. KUBEEDGE

KubeEdge is a lightweight opensource edge computing platform for Kubernetes introduced by Huawei [2], which brings cloud containerized application orchestration to edge. KubeEdge turns master-worker architecture in original Kubernetes into Cloudcore-Edgecore architecture which allows control plane to operate at the cloud and applications to run at multiple edge nodes. There are two characteristics of KubeEdge which are well-suited to support emergency communication in disaster scenario. First, KubeEdge Cloud-Edge connection is designed with At-Least-Once Delivery protocol (Figure 1) to ensure transmission reliability between cloud and edges in intermittent network condition. This allows messages from cloud can be fully delivered to edge without any loss when connection between them are recovered. Every messages or tasks from cloud need to receive acknowledgement after execution at Edge before being removed from cloud message queue. Besides connection reliability supports, Edge nodes in KubeEdge environment can work offline, maintain application running inside edge when being disconnected from cloud.

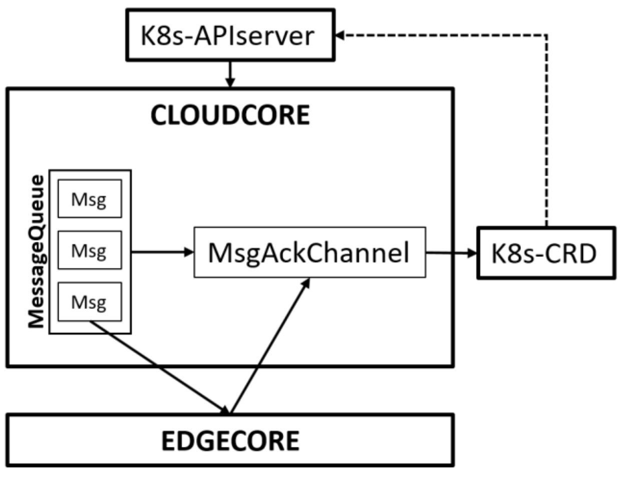

Figure 1. KubeEdge Cloud-Edge transmission reliability [3]

Second, KubeEdge optimizes kubelet at edge side in resources utilization, which makes KubeEdge memory usage at Edge very low compared with original Kubernetes. This feature saves significant setup and operation cost when deploying containerized network function, which meets one of the main requirements mentioned in the first part.

Ⅲ. NAMED DATA NETWORKING

Named Data Networking is the new network architecture which forward packages using their names instead of their location in IP-based network. Each piece of data in network has its own unique name (ex: /korea/seoul/ssu/slideX.pptx). Whenever a user wants to fetch any data, he sends Interest packet with data name encapsulated in packet header. Any NDN node inside NDN network that has the copy of the same name data can send the data back to user by using the Data packet. This feature makes NDN independent from data location which is a key challenge in emergency communication. Instead of having to know the IP address of receiver, which frequently changes due to high mobility mission in emergency mission, sender only need to know receiver's prefix. Besides, by using names as first entity inside network, NDN can solve scalability and flexibility of IP-based network in disaster scenario. Each user can be assigned a new name, which is easier to manage than IP, when his role changes during mission. Another benefit of NDN over IP-based network in emergency communication is NDN ability in aggregate same requests/Interests at intermediate nodes. Interests with the same name when come to the same router will be aggregated into one entry in router's Pending Interest Table, then only one Interest packet will be forwarded to outgoing face. When data coming back to router, copies of data will be delivered back to all incoming faces of that Interest. This feature allows NDN to save significant amount of bandwidth, which is limited in disaster recovery phase. With all mentioned advantages above, NDN emerges as a great solution for IP-based network problems in emergency communication.

IV. KUBEEDGE-NDN SYSTEM ARCHITECTURE

In our proposed system, one KubeEdge Cloudcore is responsible for managing edge nodes. Given that Cloudcore node information is available at edges, each KubeEdge edge node can join the KubeEdge environment by executing "keadm join" command in its terminal. Then each NDN node will be deployed as a virtualized NDN router and be containerized inside each KubeEdge Edgecore node. Containerized NDN routers after being deployed will establish a temporary NDN network for emergency communication. User messaging delivery application will use NDN Interest-Data packets to register user's name prefix, and exchange messages through NDN

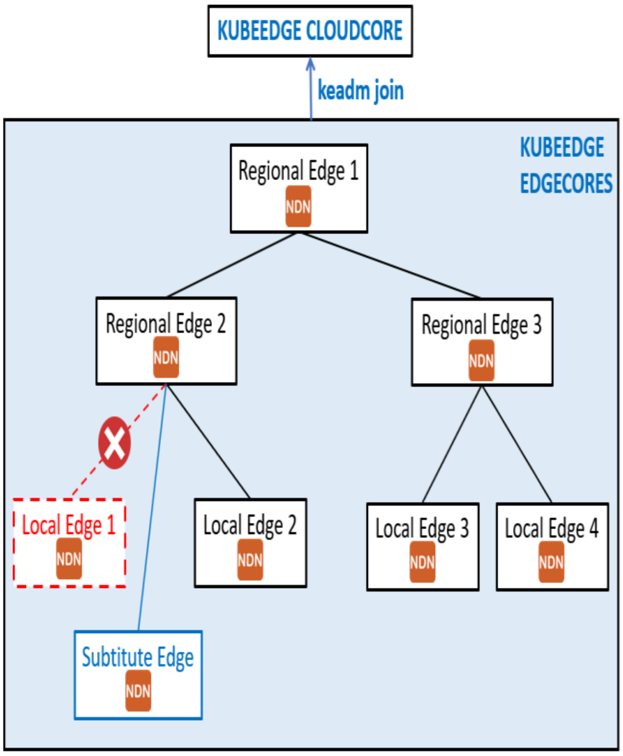

Figure 2. KubeEdge NDN general network architecture

network. The general network architecture is shown in Figure 2

In case of any node inside network is damaged or destroyed when disaster happens, it can be quickly substituted by installing new network node with KubeEdge and NDN image pre-installed. Thanks to KubeEdge's lightweight size, new nodes can be portable network devices and easily be carried to disaster areas for quick installing. Given network connection is recovered for the new node by ethernet cables or WiFi, it can send "keadm join" request to Cloudcore node. After that, containerized NDN router will be deployed at new node and re-establish NDN connection to neighbor nodes based on pod's NDN environment configuration from Cloudcore.

For application side, an NDN distributed messaging delivery mechanism is designed to support flexibility, mobility requirement of emergency communication in disaster scenario. Figure 3 demonstrates how messages are delivered using our designed NDN namespaces for emergency communication.

There are two prefixes used in message delivery. Room prefix in the form of "scenarioName/syncNumber" is responsible for synchronizing room message list in each user's application. In balance case when everyone in messaging group share the same message list, each user's application will send the same syncNumber Interest. When one user publishes a new message, other users will be responded with the new syncNumber. Message prefix in the form of

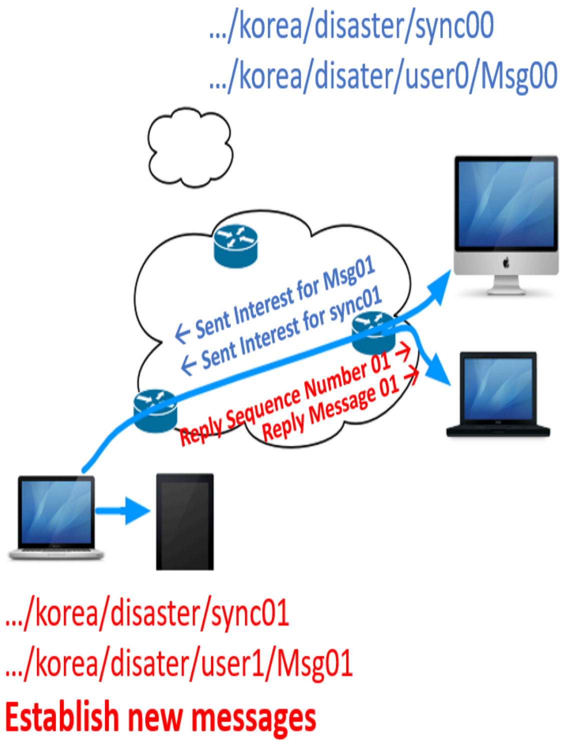

Figure 3. NDN emergency communication message delivery

"scenarioName/userID/MessageNumber" is responsible for naming each message of users. After getting the new syncNumber, each user will send Interest with the new syncNumber and will be responded with the MessageNumber need to be fetched from the published user. Then Message Interest will be sent from receivers to get the new messages from publisher.

To support mobility in our system, Figure 4 shows our proposed advertise client-server program running in user equipment (UE) and border edge (edge node that provides WiFi connection to users). Because UE does not have Named Data Link State Routing (NLSR) [4] protocol, advertise client is responsible for making every user's published prefixes to be reachable from other people in NDN network. Whenever user publishes a new prefix (ex: 2 prefixes when joining a messaging room), advertise client saves it to its local database and sends that prefix to advertise server in border edge which has NLSR to advertise to NDN network. At the same time, advertise client check for changes in UE network connection. When user moves to a new location and connects to a new border edge node, advertise client triggers the ndn-autoconfig [5] client to search for ndn-autoconfig server running at new border edge node to automatically create NDN link between UE and new edge node. Then, advertise client will send all UE prefixes in local database to new border edge node to advertise. This process is transparent to user and is completed right after user connect to a new edge node's WiFi. After that, user can continue to send and receive messages normally without experiencing any interruption in chat.

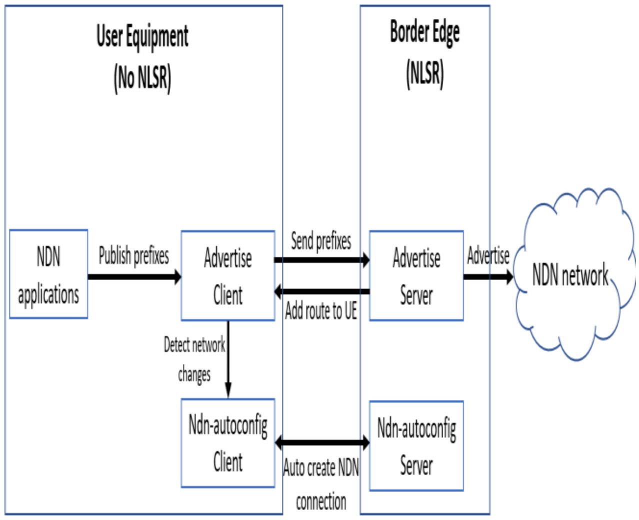

Figure 4. NDN mobility support for user equipment

V. CONCLUSION AND FUTURE WORKS

This paper presents advantages of KubeEdge and Named Data Networking for emergency communication in disaster scenarios and the general architecture of KubeEdge NDN system to deal with the problem. The system shows abilities to provide low installation cost, high flexibility, low communication overhead and mobility support for effective emergency communication. In the next version of this work, we plan to include our testbed , implementation and performance analysis of the system.

ACKNOWLEDGMENT

This work was supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2017-0-00613, Development of Content-oriented Delay Tolerant networking in Multi-access Edge Computing Environment)

REFERENCES

[1] J. Chen, M. Arumaithurai, X. Fu, K. Ramakrishnan. "CNS: Content-oriented Notification Service for Managing Disasters" Proceedings of the 3rd ACM Conference on Information-Centric Networking, pp. 122-131, 2016

[2] KubeEdge, an open platform to Enable Edge computing, https://kubeedge.io/en/

[3] KubeEdge Reliable Message Delivery, https://github.com/kubeedge/kubeedge/blob/master/docs/proposals/reliable-message-delivery.md

[4] NLSR-Named Data Link State Routing Protocol, https://named-data.net/doc/NLSR/current/

[5] ndn-autoconfig, Client tool to run NDN hub discovery procedure, https://named-data.net/doc/NFD/current/manpages/ndn-autoconfig.html