Proceedings of the 13th APAN Research Workshop 2016

ISBN 978-4-9905448-6-7

1st August 2016, The University of Hong Kong Hong Kong

Editor-in-Chief: Yoshiaki Kasahara, Kyushu University, Japan
Teck Chaw Ling, University of Malaya, Malaysia

Publisher: Asia-Pacific Advanced Network (APAN)

APAN-NRW 2016 Committee Members

Workshop Co-Chair and Technical Program Committee Co-Chair

Yoshiaki Kasahara, Kyushu University, Japan
Teck Chaw Ling, University of Malaya, Malaysia

Publicity and Local Arrangement

Manesha Hettipathirana, APAN Secretariat
N. S. Weerakoon, APAN Secretariat

Technical Program Committee

J. Adinarayana, IIT Bombay, India
Navaneethan C Arjuman, NLTVC Sdn. Bhd., Malaysia
Chaodit Aswakul, Chulalongkorn University, Thailand
Jun Bi, Tsinghua University, China
Nevil Brownlee, The University of Auckland, New Zealand
ByungRae Cha, GIST, Korea
Chalermpol Charnsripinyo, NECTEC, Thailand
Boncheol Goo, KAIST GSCT, Korea
Shigeki Goto, Waseda University, Japan
Ho Seong Han, Seoul National University Bundang Hospital, Korea
Dongsoo Har, KAIST, Korea
Andrew Howard, Australian National University, Australia
Eiji Kawai, NICT, Japan
JongWon Kim, GIST, Korea
Takuji Kiura, NARO, Japan
Yong-moo Kwon, KIST, Korea
HyunYong Lee, ETRI, Korea
Eueung Mulyana, Institut Teknologi Bandung, Indonesia
Motonori Nakamura, National Institute of Informatics, Japan
Suhaimi Napis, Universiti Putra Malaysia, Malaysia
Faridah Noor, University of Malaya, Malaysia
Koji Okamura, Kyushu University, Japan
Sanghoon Park, Hanwha Thales, Korea
Sun Park, GIST, Korea
Rungsun Rerknimitr, Chulalongkorn University, Thailand
Shuji Shimizu, Kyushu University Hospital, Japan
Wang-Cheol Song, Jeju National University, Korea
Kei Tanaka, NARO, Japan
Nguyen Huu Thanh, Hanoi University of Science and Technology, Vietnam
Denis Villorente, Department of Science and Technology, Philippines
Ye-Nu Wan, NCHU, Taiwan
Yufeng Xin, RENCI, USA
Ma Yan, Beijing University of Posts and Telecommunications, China
Kitamura Yasuichi, Kyushu University Hospital, Japan
Eric Yen, Academia Sinica Grid Computing Centre, Taiwan
Shigetoshi Yokoyama, National Institute of Informatics, Japan

Contents

Proceedings of the APAN – Research Workshop 2016
ISBN 978-4-9905448-6-7

Detecting Drive-by-Download Attacks based on HTTP Context-Types

Ryo Kiire, and Shigeki Goto.

***Abstract*—Recently, Drive-by-Download attacks have been prevailing. A user's PC may be infected with a malware derived from tampered web pages. Malicious attackers easily construct Drive-by-Download websites using a software tool, called Exploit Kit. This paper proposes a new method for detecting Drive-by-Download attacks and preventing download of malwares. Our method is based on fine-grained analysis of Drive-by-Download attacks based on HTTP Context-Types. We also evaluate a new detection method for detecting Drive-by-Download attacks, whose effectiveness is proved by the experimental results.**

***Index Terms*— Network Security, Malware, HTTP Header, Drive-by-Download Attacks, Packet Analysis**

I. Introduction

A. Background

The threat of web-based malwares has been increasing recently. A typical web-based malware is known as Drive-by-Download attacks [1]. If a user's browser or plug-in has some vulnerability, attackers may exploit it. When a user begins to browse some web pages, he/she will be automatically guided to download a malware. A malware called *ransomware* has recently become popular [2]. Ransomware encrypts user files on an infected machine and requests the user to pay some amount of money in order to decrypt the files.

There are some methods for detecting Drive-by-Download attacks, which are effective against *known* web-based attacks. Using these methods, we can compile blacklists and match them against known signatures. However, Drive-by-Download attacks have diversified over time and have resulted in *new* attacks. For example, attackers may frequently change their IP addresses and domain names to escape the blacklists. A malicious script may be obfuscated to avoid signature pattern matching.

Ryo Kiire and Shigeki Goto are with the Department of Computer Science and Engineering, Waseda University, Shinjuku, Tokyo 169-8555 Japan
e-mail: (see http://www.goto.info.waseda.ac.jp).

Thus, we certainly need a new method for detecting *new* attacks.

B. New Approach

This paper proposes a new defense method for malware downloading, which is based on a fine-grained analysis of Drive-by-Download attacks. Our new method combines several detection mechanisms that are effective for each stage of Drive-by-Download attacks.

The new method can detect attacks in real time because it judges an attack during HTTP communications. It uses common features in web-based malware, and is effective against new (unknown) malwares.

The rest of the paper is organized as follows. Section II describes the mechanism of Drive-by-Download attacks. The HTTP header is explained in section III. Section IV presents a proposal of our new method. Our evaluation results of evaluation are shown in section V. Section VI concludes this paper with possible future research.

II. Mechanism of Drive-by-Download

A. Mechanism of Drive-by-Download Attacks

Drive-by-Download attacks exploit vulnerabilities in browsers and plug-ins. Malware that infects user PCs by these attacks is called web-based malware. It is not easy for users to defend themselves from these attacks even if they are very cautious. A user's PC is attacked when the user simply looks at a website.

Drive-by-Download attacks have targeted PC browsers. Now, Android OS has become a target for Drive-by-Download attacks [3].

Fig. 1 shows the mechanism of Drive-by-Download attacks. It starts at some website. The landing page is the first step to be hit by the attack, which may originally be a benign page that is tampered to inject a malicious script. Shortened URLs in SNS are becoming increasingly popular [4].

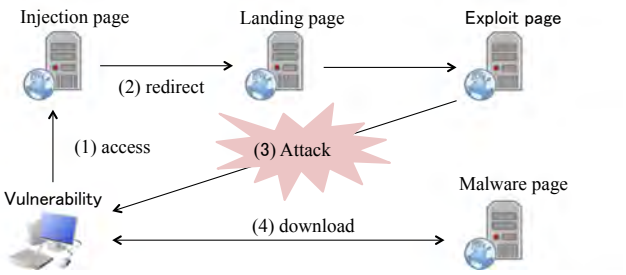


Fig. 1. Drive-by-Download Attack

B. *Exploit Kit of Drive-by-Download Attacks*

Exploit Kit is the generic name of the tool kits that cyber criminals use for Drive-by-Download attacks. They induce victims to a website where an Exploit Kit is installed. There are various types of Exploit Kits which use a wide variety of vulnerabilities [5].

According to the IBM Tokyo SOC report [6,7], 66.1% of Drive-by-Download attacks observed in the first half of 2014 used vulnerability of Java Runtime Environment (JRE), and 15.5% used vulnerability of the Adobe Flash player. The trend is changing rapidly, with 99.0% of Drive-by-Download attacks in the first half of 2015 having used vulnerabilities of the Adobe Flash player. This trend change might have resulted from the improvement in the security level of JRE. This indicates that the type of Exploit Kit and the vulnerability used for attacks is evolving.

C. *Related Works*

There are several related works for detecting Drive-by-Download attacks. We summarize here two related papers.

Sakai et al. [8] proposed a detection method based on HTTP headers. In their method, three fields in the HTTP header—X-Powered-By, Content-Type and Date—are used to detect Drive-by-Download attacks. X-Powered-By shows the PHP version, Content-Type indicates the file type, and Date is the timestamp of HTTP response. Their approach realizes a lightweight process because they do not analyze the data part (payload) of the HTML packets and executable files. The limitation of their method is the low detection accuracy and the low true positive rate. The accuracy of their method is 88.2%, and the true positive rate is 80.0%.

Kasama et al. [9] investigated several Exploit Kits. They attempted to detect Drive-by-Download attacks by using the obtained characteristics of Exploit Kits. If a Drive-by-Download attack has the characteristics of an existing Exploit Kit within their composition, then this approach can detect unknown types of attacks. However, their approach cannot detect attacks made by a new Exploit Kit and attacks without Exploit Kits. Thus, their method is entirely based on the analysis of Exploit Kits.

III. HTTP HEADER

A. *HTTP Protocol [10]*

HTTP is a well-known protocol that is used to send and receive contents between a client and an HTTP server. HTTP communications include HTTP requests and responses. A client sends an HTTP request to a server based on the URL information in a browser. The server sends back an HTTP response to the client when it receives the HTTP request.

B. *HTTP Request*

In Fig. 2, a client sends an HTTP request to a server to get contents indicated by a URL. The server that receives the HTTP request understands that the client has used the GET method in the HTTP protocol and has requested test.html in hoge.com. Some additional information such as User-Agent and Referer is also included in the header. The server processes the HTTP request based on the header information and returns the HTTP response.

```
GET /test.html HTTP/1.1
Accept: */*
Accept-Language: en-us
User-Agent: Mozilla/4.0 (compatible; MSIE 8.0; Windows
NT 5.1; Trident/4.0)
Accept-Encoding: gzip, deflate
Referer: http://hoge.com/index.html
Host: hoge.com
```

Fig. 2. HTTP request header

1) *User–Agent in HTTP request*

The User–Agent field in an HTTP header specifies the name of the client who sends the HTTP request. The server processes the request according to this User–Agent information. Furuhata [11] explained that most browsers contain a string of “Mozilla” in the User–Agent field. There is a historical reason for this string. When Netscape Navigator was highly popular among browsers, it has the “Mozilla” string in the User–Agent field. Other browsers simply followed them. The Opera browser does not have this “Mozilla” string in the User–Agent field.

2) *Referer in HTTP request*

The Referer field identifies the information of the website that is linked to the resource being requested. By checking the Referer field, the current website can see where the request came from. In Fig. 2, the HTTP request is sent to http://hoge.com/test.html through http://hoge.com/index.html.

C. *HTTP Response*

Server sends an HTTP response according to the HTTP request from the client. Fig. 3 illustrates an HTTP response header corresponding to the HTTP request header in Fig. 2.

```
HTTP/1.1 200 OK
Date: Mon, 21 Dec 2015 04:23:30 GMT
Server: Apache
X-Powered-By: PHP/5.4.44
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Pragma: no-cache
Connection: close
Content-Type: text/html
```

Fig. 3. HTTP response header

1) HTTP Status Code in HTTP response

HTTP status code is a three-digits code that classifies the responses. It indicates the response type by the first one digit number. For example, 200s means a success and 400s stands for a client error. Table 1 is a list of the HTTP status codes. In Fig. 3, the HTTP status code is 200, which means that the HTTP response is successful.

TABLE I
HTTP STATUS CODE

Status Code	Description
1xx	Informational
2xx	Success
3xx	Redirection
4xx	Client Error
5xx	Server Error

2) Content-Type in HTTP response

The Content-Type field specifies the type of the file that the server sends to the client. The client processes the file according to this field. The Content-Type field is represented as a MIME type: "type/subtype." In Fig. 3, MIME type is "text/html," which means a *text* file. MIME type may sometimes be made up of multiple types in a single file format. For example, javascript files have two MIME types "text/javascript" and "application/javascript".

D. Other HTTP header fields

In this paper, we use other HTTP header fields in addition to the ones described above. Table II shows, the HTTP header fields used in this paper.

TABLE II
HTTP HEADER FIELDS

Field Name	Description
User-Agent	Client Name
Content-Type	Type of File
Content-Disposition	Handling Information
X-Powered-By	Version of PHP

IV. PROPOSED METHOD

A. New Approach

We propose a new method to prevent the downloading of web-based malwares. Our method is an extended version of Sakai [8] and covers more header fields and provides more detailed clarification of clarifies discrimination criteria than Sakai [8]. Our method achieves high accuracy rate during detection. The new method is a combination of three detection mechanisms that utilize HTTP Content-Types. Section IV-B, deals with *text* files that have the string of "text" in Content-Type. Section IV-C, describes the detection mechanism that handles *executable* files, whose Content-Type is listed in Table III.

TABLE III
EXECUTABLE FILES FORMAT

Content-Type
application/octet-stream
application/x-msdownload
application/x-download
application/x-msdos-download

B. Detection mechanism for text files

Text files have a specific Content-Type field in an HTTP header. Content-Type has a string of "text." If a text file satisfies at least one of the following conditions, it is judged as malignant.

- There is an X-Powered-By field in the HTTP header, which meets at least one of the following conditions:
  - There is a string of "eval" in HTTP data
  - There is a string of "slice" in HTTP data
  - There is a string of "iframe" in HTTP data
- The "User-Agent" field has a string of "Java"

The first condition tries to detect landing pages in Drive-by-Download attacks. The major role of a landing page is to redirect a user's access to an exploited page, as shown in Fig. 1. In many cases, a JavaScript that performs the redirection is obfuscated, and it is not easy to analyze an obfuscated JavaScript. Our method tries to detect functions such as "eval" or "slice," which are often seen in an obfuscated JavaScript. This is the reason why we find a focus on "eval" and "slice". There is another redirection method that displays a web page by "iframe," and is often seen in Drive-by-Download attacks. Our condition covers "iframe" as well.

It is necessary to investigate HTTP data, which are usually larger than the HTTP header, to search these strings. To reduce the analysis cost, we investigate HTTP data only when the header has an X-Powered-By field. This field is known to be effective in detecting Drive-by-Download attacks [8]. By using this field, we realize a high-accuracy detection while suppressing the analysis cost by restricting the target HTTP data.

The second condition focuses on Java which is a typical vulnerable spot used by Drive-by-Download attacks. This condition aims to detect attacks using Java. The false positive rate of this condition is very small because benign HTTP communications whose User–Agent field has "Java" are unlikely.

C. Detection mechanism for executable file

This section proposes a detection mechanism for an exploited page shown in Fig. 1. We have already surveyed the

executable files format in Table III, which have a specific Content-Type in the HTTP header. If an executable file meets at least one of the following conditions, it is judged as malignant.

- There is an X-Powered-By field in the HTTP header
- There is a Content-Disposition field in the HTTP header, and the value of this field is “inline”

The first condition is explained by the same reason as the detection mechanism for text files. Note that we investigate only HTTP headers to analyze executable files, and do not investigate HTTP data (payload) at all.
In the second condition, “Content-Disposition” is a field for displaying an inline element or for downloading a file as an attachment element. If a HTTP server sets 'inline' to this field and gives a filename, his/her browser will immediately display the files as an inline element.

In a Drive-by-Download attack, this field is used to run a malicious file automatically as an inline element. Focus on this Content-Disposition field realizes an effective detection mechanism.

D. Detection mechanism for IP addresses

If a malicious file is detected by the two detection mechanisms explained above, we record the IP address of the server and put it into a blacklist. Afterwards, when an HTTP communication starts with the registered IP address in the blacklist, the server site is classified as malicious.

V. EVALUATION

A. Data set for Experiment

We use a dataset of Drive-by-Download attacks: D3M2012 to D3M2015 [12—15][16] and Threatglass [17]. D3M is a series of datasets of Drive-by-Download attacks collected by Marionette [18], which is a highly interactive web client honeypot of NTT Secure Platform Laboratories. The OS of Marionette is Windows XP and the web browser is Internet Explorer 6.0. They have implemented several plug-ins, including Adobe Reader, Flash Player, Win Zip, Quick Time, and JRE. Marionette does not include the communications data by the malware itself after infection because it does not allow the execution of malware.

Threatglass was invented by Barracuda Labs, which have posted considerable information on their website about Drive-by-Download attacks. We use a certain amount of data from Threatglass.

We also captured packets for benign data in a campus network. The collected data may include malicious data. Therefore, we extract only the normal communications data by filtering the packets under the conditions in Fig. 4.

The HTTP header satisfies at least one of the following
- There is no Accept field
- User–Agent does not include Mozilla or Opera
- User–Agent includes strings reminiscent of the bot
  - api, application, bat, bot, crawl, exe, hunny, pot, program

Fig. 4. Condition of malicious data

B. Experiment outline

To evaluate the performance of the proposed method, we conduct different experiments for malicious dataset and benign dataset. Experiment 1 uses malicious dataset to calculate the detection rate and experiment 2 uses benign dataset to calculate the error detection rate.

In experiment 1, it is necessary to count the accurate number of attacks in the dataset before applying the new method. We conduct a preliminary experiment to sort out the dataset that consists of captured packets. The result contains a series of successful attacks whose number is precisely counted. Then, we will conduct experiment 1 to measure the detection rate of Drive-by-Download attacks.

The preliminary experiment is divided into three steps. First, we count the number of pairs of HTTP request and response pairs. Next, we label the page transition in Drive-by-Download attacks. Finally, we count the number of successful malware downloading.

In experiment 2, it is also necessary to count the accurate number of pairs of HTTP request and HTTP response pairs in the benign dataset. Then, we can measure the error rate of the newly proposed method.

C. Number of Drive-by-Download Attacks

We conduct a preliminary experiment to extract meaningful data from the dataset.

1). Count the number of pairs of HTTP request and HTTP response pairs
2). Estimate the source and the destination of a transition
3). Count the number of attacks

1) *Count the number of pairs of HTTP request and response*

An HTTP communication is an HTTP request and response pair. A pair is represented as a vector with four elements (source IP, destination IP, source port, destination port). If a client uses the same port number more than once for the same destination IP address, the first port number in chronological order is used. Other duplicated port numbers are not used in the experiment.

If there are no HTTP responses in the dataset, we will use only HTTP request as a pair. If there is no HTTP requests, while there is a certain HTTP response, we remove the response from the dataset.

2) Estimating the source and the destination of a transition

A malicious dataset consists of Drive-by-Download attacks. An attack triggered by a redirection which has a transition source page and a destination page. We estimate the source and destination pages by the URL and the IP address before or after the transition according to the relationship.

- Referer URL
  - If there is a Referer field in and HTTP request, the URL indicates the page before the transition.
- Location URL

- If there is a Location field in and HTTP response, the URL in the value is the new page after the transition.

- URL String
  - If there is a URL String in an HTTP data, this URL is the new page after the transition.
- IP Address
  - If an earlier HTTP response in chronological order has the same client IP address, the earlier response is the previous page before the transition.

Our method refers to the earlier work by Takata [19].

3) *Count the number of attacks*

Based on the estimated transition relations, we can extract successful Drive-by-Download attacks from the dataset. In Fig. 5, a file icon represents a file in an HTTP communications, including, HTML, javascript, CSS, and image. An arrow in Fig. 5 indicates a page transition, including a redirection. At the top of the figure a large icon of file A represents an HTTP data retrieved from the server URL.

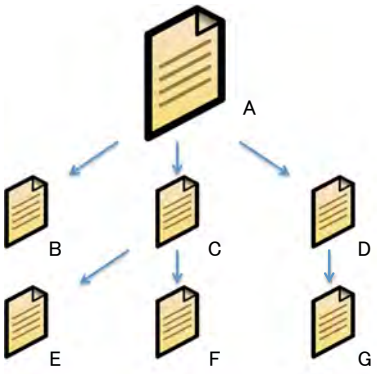


Fig. 5. Successful HTTP communication

Our concern is whether it is possible to prevent the download of malware caused by a Drive-by-Download attack. However, the D3M dataset contains interrupted communications which are unsuccessful attacks. They do not include the malware download. It is necessary to exclude such unsuccessful attacks.

D3M dataset and Threatglass dataset consist of Drive-by-Download attacks. If there is an executable file in the dataset, it is expected to be a malware. It is possible to check whether an HTTP communication contains a malware. If an HTTP communication contains an executable file, it has a malware. There are three conditions to find an executable file:

- A file extension is .exe
- Content-Type is shown in Table III
- A PE format file is an executable file in MS Windows

If a file meets at least one of the above conditions, and the associated HTTP status code is 2XX (success), then the file is regarded as an executable file. We extract successful attacks from the dataset with one or more executable files.

D. Experiment 1: detection rate for malicious dataset

The performance evaluation is conducted by applying the proposed method to the malicious dataset prepared in the previous section.

It is better to explain the meaning of preventing malware download. Fig. 6 is an example of a Drive-by-Download attack, where file G is a malware. If we block the downloading of file C, it also blocks downloading file E and file F. However, this blocking is not related to file G which is malicious. It is necessary to determine whether file D is malicious or file G is malicious. If any file in a series of transitions to reach the malware is found malignant by the proposed method, it is called a successful "prevention," and an unsuccessful "prevention" otherwise.

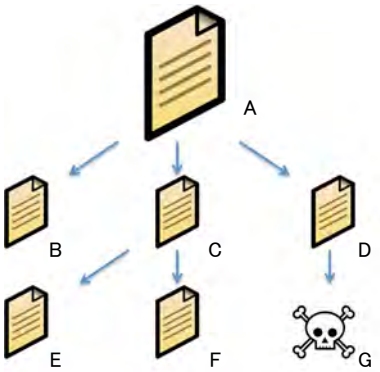


Fig. 6. Example of Drive-by-Download Attack

Using the term "prevention," the results of the evaluation experiments are shown in Table IV.

TABLE IV
RESULTS OF EXPERIMENT 1

Dataset	Prevention	Number of Data
D3M2015	13	21
D3M2014	42	43
D3M2013	19	19
D3M2012	29	30
Threatglass	41	51
Total	144	164

The true positive rate is 87.8%. There are eight unsuccessful preventions in D3M2015, six of which are attacks that uses the same Exploit Kit. It is not easy to detect these attacks because the transition to a malware is implemented by an obfuscated javascript. It can also not be detected by the X-Powered-By field test. We will visit this issue in the Conclusion of this paper.

E. Experiment 2:error detection rate for benign dataset

In experiment 2, the proposed method is applied to the HTTP communication data that was prepared in Section V-C from the benign dataset. The performance evaluation includes the evaluation of the number of false positives. The results are shown in Table V.

TABLE V
RESULTS OF EXPERIMENT 2

Content-Type	False Positive	Number of Data	False Positive Rate [%]
Text	21	1353	1.55
Executable	1	65	1.53
All	171	6386	2.68

The false positive rate for both text files and executable files is around 1.5%. The total false positive rate is increased to about 2.7%. The reason of this 2.7% is the error detection by IP addresses which is described in Section IV-D of this paper.

F. Summary of Results

The results of experiment 1 and 2 are presented in Table VI.

TABLE VI
RESULTS OF EXPERIMENT 1 AND EXPERIMENT 2

		Result	
	Total	Malicious	Benign
Total	6550	164	6386
Malicious	164	144	20
Benign	6386	171	6215

The accuracy of our detection mechanism is 97.1%. Thus, our detection mechanism exhibits improved accuracy and TPR compared to Reference [8].

VI. CONCLUSION

A. Summary

This paper focuses on the structure of Drive-by-Download attacks and tries to prevent malware download by the detection mechanism for each attack stage. The results show that the proposed method is effective for attacks in datasets.

The proposed method realizes a low-cost detection, and also enables real-time detection, during web-browsing communications.

This method is based on the features of a wide range of Drive-by-Download attacks, and is not limited to a specific Exploit Kit. Therefore, it would be effective against unknown new attacks.

B. Future Research

1) Improvement of estimation of redirect structure

We used the method presented in Section V-C to extract successful attacks. However, there may be a page transition that is not estimated correctly. For example, there is an operation of reading a particular file after a certain period of time by the "settimeout" function of javascript. The function can be obfuscated. In this operation, the URL does not appear because of obfuscation. We use an IP black list in the proposed method; however, IP addresses may change with time. If an HTTP communication occurs for other files by the same IP address during a specified predetermined time by the "settimeout" function, an incorrect estimation may occur.

We also find an incorrect estimation about a URL as the transition destination, and hence need more data analyses to improve our method's accuracy.

2) Improvement in the proposed method

The proposed method adopts the features that are discussed in various related literature. By improving the proposed method to analyze the characteristics of various Exploit Kits, a more precise detection can be expected. In particular, there is room for improving signature matching with HTTP communications data.

Our proposed method can be used for real-time detection, and can be implemented it as a plug-in to a web browser.

3) Verification of a various dataset

In this paper, we use D3M2012 to D3M2015 and Threatglass as the datasets for Drive-by-Download attacks. We have not yet investigated the appropriate Exploit Kit to be used for each attack. Therefore, it is important to evaluate the performance of the various Exploit Kits in the study.

ACKNOWLEDGEMENTS

We thank all the members of the IPSJ MWS (anti-Malware Engineering Workshop) community. A part of this work was supported by JSPS Grant-in-Aid for Scientific Research B, Grant Number 16H02832.

REFERENCES

[1] JPCERT. (2013, June.). Attention on the site falsification. JPCERT. Available: https://www.jpcert.or.jp/english/at/2013/at130027.html
[2] JPCERT. (2015, May.). Attention on the ransomware. JPCERT. Available: https://www.jpcert.or.jp/english/at/2015/at150015.html
[3] Axelle Apvrile. (2014, February.). New Drive-By Download Android Malware. FORTINET. Available: *http://blog.fortinet.com/post/new-drive-by-download-android-malware*
[4] Atsushi Yoshizawa. (2011, March.). 14.3% of report of illegall attack site is short URL, not stop Drive-by-Download Attack. CNET Japan. Available: http://japan.cnet.com/news/business/20426859/
[5] Brooks Li. (2014, December.). What's New in Exploit Kit 2014. TREND MICRO. Available: http://blog.trendmicro.com/trendlabs-security-intelligence/whats-new-in-exploit-kits-in-2014/
[6] Tokyo SOC. (2014, August.). 2014 the firsts half of Tokyo SOC information analysis report. Tokyo SOC. Available: https://www-304.ibm.com/connections/blogs/tokyo-soc/resource/PDF/tokyo_soc_report2014_h1.pdf?lang=ja
[7] Tokyo SOC. (2015, August.). 2015 the firsts half of Tokyo SOC information analysis report. Tokyo SOC. Available: https://www-304.ibm.com/connections/blogs/tokyo-soc/resource/PDF/tokyo_soc_report2015_h1.pdf?lang=ja
[8] Hiroaki Sakai, Ryoichi Sasaki, "Proposal of detection method based on HTTP headers against Drive by Download Attack," JPSJ, Vol.2013-DPS-154, No.29, pp.1-6, March. 2013.
[9] Takahiro Kasama, Masaki Kamizono, Daisuke Inoue, "Drive-by-Download Attack Detection based on Characteristics of Exploit Kit," Computer Security Symposium 2013, October. 2013.
[10] RFC 7231. Available: https://tools.ietf.org/html/rfc7231
[11] UserAgentString.com. (2005-2011.). List of all Browsers. OpenSpace. Available: http://www.useragentstring.com/pages/Browserlist/
[12] anti Malware engineering WorkShop 2012(MWS2012). Available: http://www.iwsec.org/mws/2012/en.html

[13] anti Malware engineering WorkShop 2013(MWS2013). Available: http://www.iwsec.org/mws/2013/en.html
[14] anti Malware engineering WorkShop 2014(MWS2014). Available: http://www.iwsec.org/mws/2014/en.html
[15] anti Malware engineering WorkShop 2015(MWS 2015). Available: http://www.iwsec.org/mws/2015/en.html
[16] Mitsuhiro Hatada, Mitsuaki Akiyama, Takahiro Matsuki, Takahiro Kasama, "Emprowering Anti-malware Research in Japan by Sharing the MWS Datasets (Preprint)," IPSJ, 1882-7764, Sep. 2015.
[17] Threatglass. Barrcuda Labs. Available: http://www.threatglass.com/
[18] Mitsuaki Akiyama, et al, "Design and Implementation of High Interaction Client Honeypot for Drive-by-Download Attacks," IEICE Transactions on Communication. Vol.E93-B No.5. pp.1131 – 1139, May. 2010.
[19] Yuta Takata, Tatsuya Mori, Shigeki Goto, "Redirect Analysis of web-based malware," Proceedings of the APAN – Network Research Workshop 2011, August. 2011.

Ryo Kiire Ryo Kiire received the B.S. degree in Computer Science and Engineering from Waseda University in March, 2016. He is now a master student at Department of Computer Science and Communications Engineering, Waseda University. His research interest Cyber Security.

Shigeki Goto Shigeki Goto is a professor at Depart-ment of Computer Science and Engineering, Waseda University, Japan. He received his B.S. and M.S. in Mathematics from the University of Tokyo. Prior to becoming a professor at Waseda University, he has worked for NTT for many years. He also earned a Ph.D in Information Engineering from the University of Tokyo. He is the president of JPNIC. He is a member of ACM and IEEE, and he was a trustee of Internet Society from 1994 to 1997.

Proceedings of the APAN – Research Workshop 2016
ISBN 978-4-9905448-6-7

Design and Implementation of a DMARC Verification Result Notification System

Naoya Kitagawa, Toshiki Tanaka, Masami Fukuyama and Nariyoshi Yamai

***Abstract*—Damages caused by spoofed e-mails as sent from a bank, a public organization and so on become serious social problems. In such e-mails attackers forge the sender address to defraud receivers of their personal and/or secret information. As a countermeasure against spoofed e-mails, sender domain authentication methods such as SPF and DKIM are frequently utilized. However, since most spoofed e-mails do not include DKIM signature in their e-mail header, those e-mails cannot be authenticated by the conventional system. Additionally DKIM has a problem that cannot determine whether the attached signature is legitimate. In this paper, we propose a method to detect spoofed e-mails and alert the user without DKIM signature by utilizing DMARC and implement a system that sends DMARC verification results to receivers. By utilizing this system, the users can obtain alerts for spoofed e-mails that the existing systems cannot warn.**

***Index Terms*—Anti spam, DKIM, DMARC, Sender Domain Authentication, SPF, Spoofed E-mail**

I. Introduction

E-mail communication is one of the most widely used service on the Internet. However, various malicious usages of e-mail have been becoming a serious social problem over the years. For instance, MITM (Man In The Middle) attack and DDoS (Distributed Denial of Service) attack are typical abuse examples of e-mail communication. In addition, phishing mails, that aim to defraud receivers of their personal and/or secret information under the guise of a bank or a public organization and so on, are frequently in circulation. Such e-mails are called spoofed e-mails since the most senders spoof their addresses or display names. Moreover, the damages have been growing by fraud caused by spoofed e-mails, therefore many police agencies around the world such as the FBI have been alerting [1].

Sender domain authentication methods have been proposed as countermeasure mechanisms against spoofed e-mails. As typical sender domain authentication method, SPF (Sender Policy Framework) [2] and DKIM (DomainKeys Identified Mail) [3] are widely utilized. SPF examines the validity of the sending mail server using the IP address. DKIM examines whether the message has not been tampered and whether the message has sent from proper sender using the digital signature. However, since most spoofed e-mails are considered to be sent without DKIM signature in the mail header, they cannot be verified by DKIM.

In this paper, we propose a method to distinguish spoofed e-mails without DKIM signature by using DMARC (Domain-based Message Authentication, Reporting, and Conformance) [4]. Although DMARC is utilized for the administrator of sender's domain to obtain the aggregate report or authentication failure report in general, our system notifies the receivers of spoofed e-mail by utilizing DMARC. To realize this method, we implemented a system that performs sender domain authentication using DMARC, and notifies the receiver of the authentication result according to the contents of DMARC policy to each receivers.

The rest of the paper is organized as follows. In Section II, we present existing methods. In Section III, we describe the design of our spoofed e-mail alert system. Then, Section IV shows an implementation method of the system. Section V shows notification examples of DMARC verification results and an alert example of an actual received spoofed e-mail. Finally, we present concluding remarks and suggestions for future study.

Submitted Date: 27th May 2016.

Naoya Kitagawa is with Division of Advanced Information Technology & Computer Science, Department of Institute of Engineering, Tokyo University of Agriculture and Technology, Koganei, Tokyo 184-0026 Japan (e-mail: nakit@cc.tuat.ac.jp).

Toshiki Tanaka is with Department of Computer and Information Sciences, Faculty of Engineering, Tokyo University of Agriculture and Technology, Koganei, Tokyo 184-0026 Japan (e-mail: t-tanaka@net.cs.tuat.ac.jp).

Masami Fukuyama is with Department of Computer and Information Sciences, Graduate School of Engineering, Tokyo University of Agriculture and Technology, Koganei, Tokyo 184-0026 Japan (e-mail: mfuk@net.cs.tuat.ac.jp).

Nariyoshi Yamai is with Division of Advanced Information Technology & Computer Science, Department of Institute of Engineering, Tokyo University of Agriculture and Technology, Koganei, Tokyo 184-0026 Japan (email: nyamai@cc.tuat.ac.jp).

```
Return-Path: <sender@example.com>
(snip)
DKIM-Signature:v=1; a=rsa-sha256;
      c=relaxed/relaxed;
      d=example.net;
      s=20120113;
      h=mime-version:date:message-id:subject:from:to:
      content-type;
      bh=YzODIQzFL5CIwg3H6lYD6ZafgsQR/7HxA6gRkSc7Vvg=;
      b=Jd6cf0fJGsMyekr7dUL6jjxVywqRXhkKeBcdFYdk/KzuHKZisyg/3
      iJMNlQq7wtDT6wU9uijAoEnPQirUwCHLFCJHqkliiDBva56Ec5nuGX
      AxsjLCU3XwwMQ1ABcGSepSl+e5kozZFBG7ItOZ5eXBXEyAAvChoLgu
      jjnUHJtS6uYOuSC6pVlHpyg1uzm+bVk97/w0dxc64Z8xaWMneN6KBL
      od28r7KORNgU8K6GKkwjfcYi1lkm1KBuW3X9YR8nVmhXjsRIyEhz25
      6a3WLYqKbC7cPHaK8lxFVHzE1AoZwhsgMRCswRCR9026OkWSvpuVvk
      +qN5CsarxWxmA==
(snip)
From: <sender@example.com>
To: receiver@example.org
```

Fig. 1. A sample of E-mail Header

II. EXISTING METHODS

A. Sender Domain Authentication

Currently, SPF and DKIM have been widely utilized as sender domain authentication methods.

SPF is an authentication method using the IP address of the sender's SMTP (Simple Mail Transfer Protocol) server and the domain of Envelope-From address. In order to use the verification method, a sender domain publishes an SPF record at its own DNS (Domain Name System) server in advance. The SPF record indicates the servers that may send messages with the sender address of the domain. Then, a receiver obtains the sender's SPF record and investigates whether the IP address of the sender's SMTP server is included in the SPF record. However, SPF has a problem that is not able to authenticate forwarded messages properly. This is because the IP address of the SMTP server becomes the IP address of the relay server rather than that of the original server is used for authentication, which does not match the SPF record.

Secondly, DKIM is a method using digital signature. In order to use this method, a sender domain prepares a pair of a private key and a public key in advance. Then, the sender domain publishes the public key at their DNS server. At the time of mail sending, the sender domain creates a digital signature from the mail header and the body using the secret key, and adds it to the mail header as the DKIM signature. In Fig.1, the value of "b=" tag shows the DKIM signature. Then, a receiver queries the public key to the authoritative DNS server of sender's domain that obtained from the "d=" tag of DKIM signature header. Subsequently the receiver compares the hash value obtained from the digital signature by using the public key with the hash value, that is the value of "bh=" tag. As a result, DKIM verification will be success when these values are the same. With such a mechanism, unlike SPF, DKIM can verify forwarded messages properly.

However, DKIM permits even a "d=" tag domain (example.net in Fig.1) different from the domain of Envelope-From address (example.com in Fig.1). Thus, if a spammer sends spoofed e-mails from the address of his/her own domain with the DKIM signature, the DKIM verification will be success.

B. DKIM Verification System Using POP Proxy

Our research group has proposed a system to perform a sender domain authentication by DKIM using a POP proxy [5]. Although DKIM verification is usually performed by mail service provider's server, this system verifies messages by using a POP proxy installed by each organization. In addition the system reports the verification results to each user. Even if the receiving mail server that is operated by universities, companies, ISPs, and so on does not support DKIM verification, the verification gets available independently by introducing this system at each organization. In this system, when the proxy receives a retrieval request from a mail client, the proxy gets messages from the mail server and performs DKIM verification. Then, the proxy puts the verification result into the mail header. Based on the verification result, proxy or MUA (Mail User Agent), such as Outlook, notifies the result to each user. Of course, since this system notifies based on DKIM verification result, the system cannot perform the verification for the messages without DKIM signature.

C. DMARC

DMARC is a framework of reporting and declaration of policy control using two sender domain authentication mechanisms, SPF and DKIM, and that has been spreading recently. The reporting function notifies the authentication failure reports and the aggregate reports to the administrator of the sender's domain. The administrator is able to know whether the authentication has been performed as intended by this report. On the other hand, in the policy declaration function, a sender can specify the e-mails handling method in case of sender domain authentication failure.

In addition, DMARC has the concept of "alignment". This concept means that DMARC verification gets failed even if the domain for verification (SPF and DKIM) is different from the sender's Header-From domain. SPF and DKIM verification need not be the same the Header-From domain and Envelope-From domain for SPF or the domain for signature for DKIM. Moreover, attackers can spoof the Header-From address easily. By taking advantage of alignment, a receiver can confirm the validity of the Header-From domain. A sender domain can specify the strictness of relationships between these domains using DMARC record. If a sender specifies "strict" as the alignment, DMARC verification will fail unless the domain of the Header-From address and the domain for SPF or DKIM verification match completely. On the other hand, if a sender specifies "relaxed" as the alignment, DMARC verification will succeed if the subdomains of the domain are identical.

In order to use this mechanism, a sender domain needs to support SPF and/or DKIM. Additionally, the sender domain must publish the DMARC record at its DNS contents server. DMARC record shows the recipient e-mail address for

TABLE I
VALUES OF "P=" TAG AND CORRESPONDING HANDLING METHODS

"p=" tag	How to Handle failed-messages
None	Inaction even if the authentication failed.
Quarantine	Quarantine the authentication failure mails.
Reject	Do not receive the authentication failure mails.

verification result reports and indicates the e-mails handling method in case of sender domain authentication failure. A receiver domain performs sender domain authentication of both SPF and DKIM, and applies the policy when the both of verifications are failed.

As mentioned above, a sender domain specifies the handling method for the verification-failed messages at the "p=" tag of DMARC record as the DNS contents server. TABLE I shows the values of "p=" tag and processing details corresponding to the each policy.

For example, let us consider the case where a sender domain (example.com) is supporting DMARC. Then, we assume that the sender domain is publishing the DMARC record as a TXT record of "_dmarc.example.com" in the following manner.

v=DMARC1\; p=none\; rua=mailto:reports@example.com

In this example, since the value of "p=" tag is "none", the administrator of example.com requests not to perform the isolation or reception rejection of the e-mails even if the DMARC verification is failed. Additionally, the administrator requests to send the reports to "reports@example.com" as shown in the "rua=" tag.

Fig.2 shows the flow of DMARC verification.

1) A sender domain supports own domain to the SPF and/or DKIM.
2) The sender domain also publishes the DMARC record as a TXT record of its DNS contents server.
3) SPF and DKIM verifier on receiver mail server sends a query to the DNS contents server and gets the SPF record and the public key for DKIM. Then it performs the SPF and the DKIM verification.
4) SPF and DKIM verifier sends the verification results to the DMARC verifier.
5) DMARC verifier sends a query for DMARC record to the DNS contents server of the Header-From domain.
6) If DMARC verifier obtains the DMARC record, it applies the DMARC policy based on the verification results of SPF and DKIM, and whether the sender domain matches the "alignment".
7) DMARC verifier creates an aggregate report containing the verification results and the applied policy, and sends it to the e-mail address as shown in the "rua=" tag.

TABLE II shows the percentage of each DMARC policy based on the number of domains that we have observed. As shown in table, since the most of domains' DMARC policies

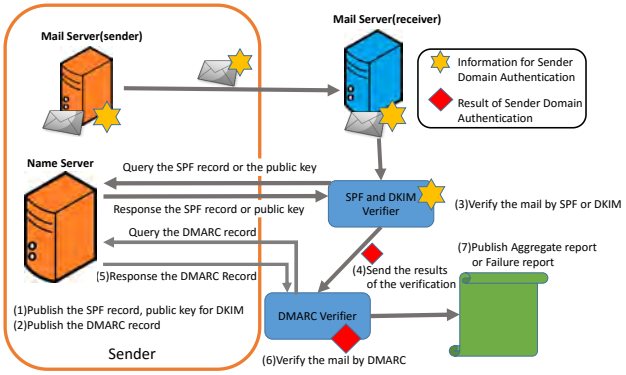


Fig. 2. Flow of DMARC Verification

TABLE II
SURVEY OF DMARC POLICY

Polocy	2016/2	2016/3	2016/4
none	1,473 (81.65%)	1,261 (82.31%)	1,821 (77.79%)
quarantine	123 (6.82%)	93 (6.07%)	209 (8.93%)
reject	192 (10.64%)	170 (11.10%)	305 (13.03%)
error	16 (0.89%)	8 (0.52%)	6 (0.26%)
Total	1,804 (100%)	1,532 (100%)	2,341 (100%)

are published as "none", the receiver will accept the verification-failed messages without rejection or quarantine. We can consider from this survey that many DMARC compliant sender domains hope receiver domains to accept spoofed messages as are and only to send aggregate reports. Hence the isolation or rejection effect of DMARC against spoofed e-mails is currently limited.

III. DESIGN OF DMARC VERIFICATION RESULT NOTIFICATION SYSTEM

A. *Summary of the System*

As described in Section II-A and Section II-B, DKIM cannot perform the verification for the e-mails that do not attach the digital signature. In other words, even if a received e-mail is from a domain that should have with a DKIM signature, DKIM cannot determine the e-mail that does not exist a DKIM signature as spoofed e-mail. To solve the problem, we propose a system to warn of such e-mails by utilizing DMARC. This system does not focus on creating and sending the reports explained in Section II-C.

Our proposed system performs sender domain authentication and notification of DMARC perform on users' terminals. By performing on users' terminals, PC users can easily adopt the sender domain authentication mechanisms and/or DMARC verification even if the user's mail receiving server does not support these mechanisms. The system obtains the mail receiving server information required for SPF from "Received" field of the mail header. After that, the system determines the boundary of the internal and external organization, and the system uses IP address of the nearest

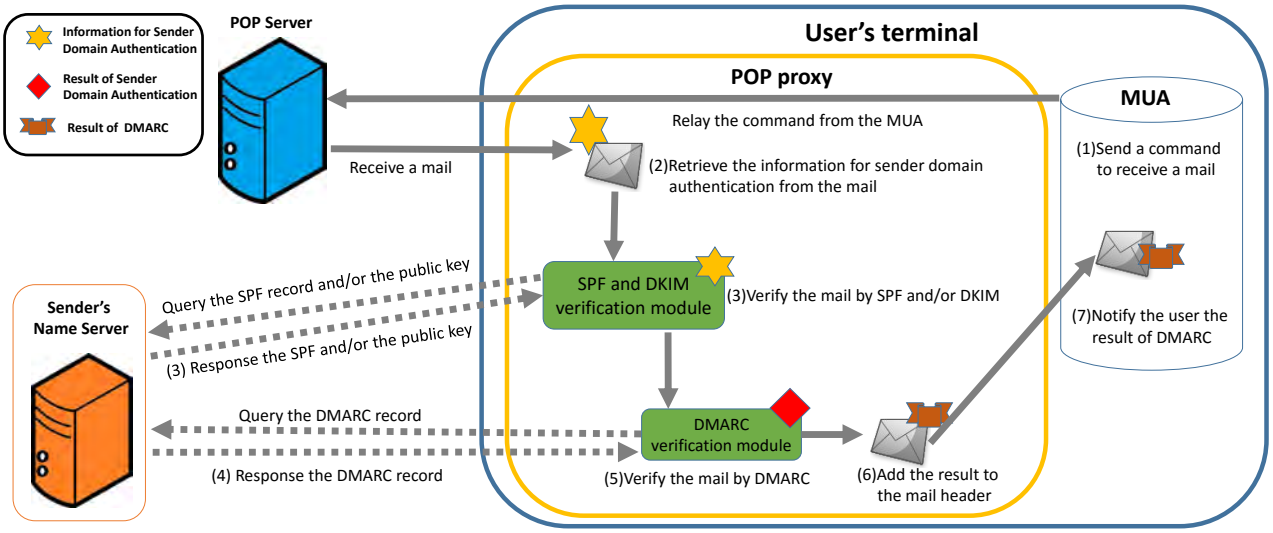


Fig. 3. Flow of DMARC Verification by Our System

external organization to the boundary and the e-mail address indicated by "Return-Path" for SPF verification.

SPF and DKIM verification are performed by the verification module shown in Fig.3. DMARC verification module receives the results of sender domain authentication and determines whether to apply the DMARC policy. DMARC verification module judges "pass" or "fail" as the verification result. Subsequently the system notifies the verification result to MUAs.

B. *Summary of the System*

Fig.3 shows the behavior of the POP proxy and a client in this system.

1) When the POP proxy received a message acquisition command from a MUA, the proxy relays the command to the POP server.
2) The proxy retrieves the information required for authentication from the header of the acquired e-mail, and inputs the information to the SPF and DKIM verification module.
3) SPF and DKIM verification module performs sender domain authentications based on the information obtained from the header.
4) DMARC verification module queries to the sender's DNS contents server, and acquires the DMARC record.
5) DMARC verification module applies the DMARC policy based on the result of the sender domain authentications.
6) The proxy adds the DMARC verification result to the mail header, and delivers the e-mail to the MUA.
7) The MUA notifies the user the DMARC verification result.

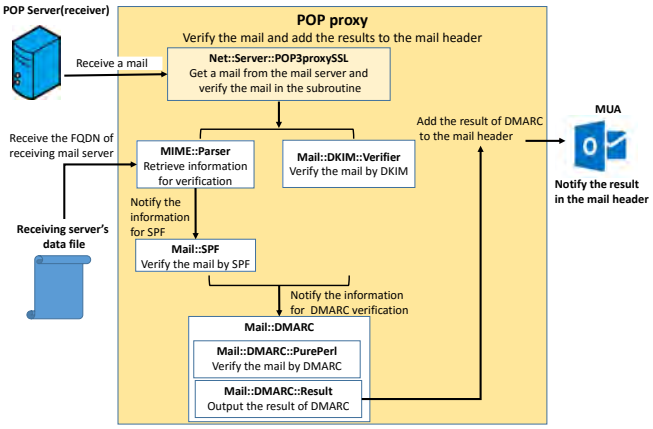


Fig. 4. Structure of Our System

IV. Implementation of the DMARC Verification Result Notification System

Based on the design described in Section II-C, we have developed the system using Perl. In order to perform DMARC verification, this system is configured by using Mail::DMARC and Mail::DMARC::PurePerl [6] that are modules published on CPAN. We used MIME::Parser [7] and Net::Server::POP3proxySSL, that was created based on Net::Server::POP3proxy, to obtain the information required for the verification from the mail header. In addition, by implementing them all on Cygwin, our proposed system works on a user's terminal.

First, we describe an implementation method of the part to obtain the information required for verification from a mail header. The parts necessary for verification are Return-Path, DKIM signature, From:, and To: in a mail header shown in Fig.1.

TABLE III
THE FIELDS THAT CAN BE OBTAINED BY MAIL::DMARC::RESULT

Field	Contents
result	DMARC verification result. (pass. fail)
disposition	DMARC policy when the result field is “fail”.
reason	The reason of the verification failure when the result field is “fail”.
dkim	The result of DKIM verification.
dkim_align	The degree of coincidence with the DKIM signature domain and the Header-From domain.
spf	The result of SPF verification.
spf_align	The degree of coincidence with the envelope-From domain and the Header-From domain.

Fig.4 shows the structure of our proposed system. The operation of POP proxy in this system can be divided into five of 1) obtaining a message from the POP server, 2) analysis of the mail header, 3) execution of the sender domain authentication, 4) execution of DMARC verification, 5) addition of the verification result. We describe about the implementation method for each of these steps.

1) In order to implement the POP proxy and obtain a message, we used Net::Server::POP3proxySSL that was created based on Net::Server::POP3proxy. This module receives a message from the POP server and stores it in “$[0]”. By passing the variable to “flterAction” that is a subroutine function, this module can perform processing on the message.
2) We used the MIME::Parser for the header of the analysis. This module isolates the mail header and the body, and extracts the necessary information using regular expression. Additionally, the module retrieves the sender information to be used for SPF verification from the “Received” field. The sender's information used for SPF verification is indicated on “Received” field the server located in the boundary of the internal and external organization is added to the header. Then, the system reads the receiving server's data file that retains the information of own organization's receiving server, and scans “Received” field. By preparing the external file, each organization is able to specify the receiving server without modifying the program code. When the appropriate “Received” field is specified and the source IP address is obtained, the module terminates.
3) The system performs SPF verification by using the information that was extracted with 2). We utilized Perl module Mail::SPF [8] for SPF verification. The system performs SPF verification by passing the sender IP address and Envelope-From address to this module. On the other hand, DKIM need to use the entire message for the verification. By passing “$_[0]”, that contains the entire message, to Perl module Mail::DKIM::Verifier [9], the system performs DKIM verification.
4) By using the information extracted in 2) and the result of authentication performing in 3), the system performs DMARC verification by Mail::DMARC::PurePerl which is a method of Perl module Mail::DMARC. The system performs the verification by passing the sender IP address, Envelope-From address, Header-From address, and verification results of SPF and DKIM to Mail::DMARC::PurePerl.
5) The system appends the DMARC verification result obtained in 4) to the mail header, and delivers to the MUA. The system receives DMARC verification result from DMARC::Mail::Result method. TABLE III shows the fields about verification results that are possible to obtain by this method. The system adds the verification result and Header-From domain regardless of the verification result to the mail header. Moreover, when the verification result is “fail”, this method can obtain the failure reason from “reason” field. Therefore even though the verification result was “fail”, this system can obtain “no_policy” as the failure reason from the “reason” field when the sender domain was not supported DMARC. In addition, as described in Section II-C, DMARC is different from DKIM, the verification will be failed when the domain indicated by the “d=” tag and the domain indicated by the Envelope-From address are different.
   The “spf_align” and “dkim_align” field indicates “strict” when the header from domain and the domain for each verification are completely consistent, and when each of these domains is the relationship of the subdomains, the field indicates “relaxed”. On the other hand, these fields do not have information when DMARC verification failed due to these domains are different.
   The system appends the domain of the receiver's e-mail address, SPF verification result and Envelope-From, DKIM verification result and its signed domain, DMARC verification result, and Header-From domain to the mail header. Additionally, when the verification fails, the system appends the reason. Therefore RFC7001 allows the freely description in the parentheses, the system adds the DMARC policy and the reason of verification failure as shown in Fig.5.

```
Authentication-Results: user.example.com;
  spf=pass smtp.mailfrom=sender@example.net;
  dkim=pass header.d=example.net;
  dmarc=fail (p=reject comment=no match)
  header.from=sender@example.com
```

Fig. 5. Addition of the Verification Result to the Mail Header

V. NOTIFICATION OF DMARC VERIFICATION RESULT

We implemented the notification function of DMARC verification results by using the label of Microsoft Outlook 2013 as a user's MUA. The system notifies the four types shown in the lower part of the Fig.6 based on DMARC

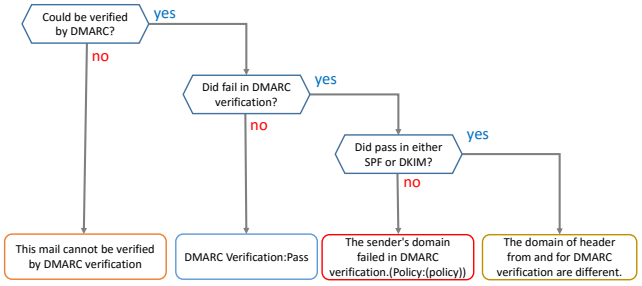


Fig. 6. Flow of the Label Addition

verification result.

Moreover, Fig.7-10 show the actual label additional examples in the MUA. The system appends a blue label when succeeding in the verification (Fig.7), and the system adds a yellow label when the verification failed and the sender domain indicated “none” or “quarantine” as the DMARC policy (Fig.8). Furthermore, the system appends an orange label when the sender does not correspond to DMARC (Fig.9), and the system adds a red label when the verification failed and the sender domain indicated “reject” as the DMARC policy (Fig.10).

Additionally, when the applied policy was “reject”, that represents such e-mails did not attach the DKIM signature even though all of the legitimate transmissions that send from the domain are supposed to be attached the signature. Otherwise, such e-mails mean that failed to the verification. In any case such mails are extremely high possibility of spoofing or falsification, therefore the system alerts by pop-up window in addition to the red label notification as shown in Fig.11.

VI. DISCUSSION

In general usage of DMARC, a receiver does not handle spoofed e-mails unless the sender's DMARC policy is “reject” or “quarantine”. However, as shown in TABLE II, about 80% of the DMARC compliant domains publish “none” as the policy. Therefore, the existing systems cannot isolate or reject the e-mails even if those are very high probability of being spam mails. On the other hand, by giving various warnings according to each policy, our system enables alerting to spoofed e-mails that the conventional systems cannot warn.

Moreover, since DMARC can be expected to spread more widely in the future, the effectiveness of this system will be increased.

VII. CONCLUSION

In this paper, we proposed a system that distinguishes spoofed e-mails utilizing DMARC. Our proposed system can alert spoofed e-mails that do not attach the DKIM signature even though all of the legitimate transmissions that send from the domain are supposed to be attached the signature. A

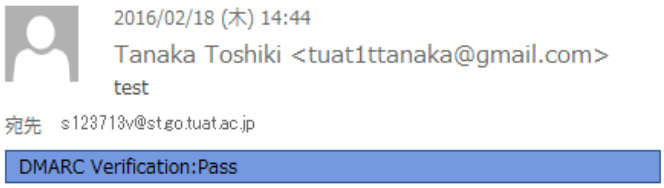


Fig. 7. Addition of “pass” Label

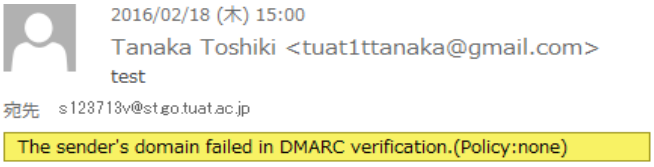


Fig. 8. Addition of “none” Label

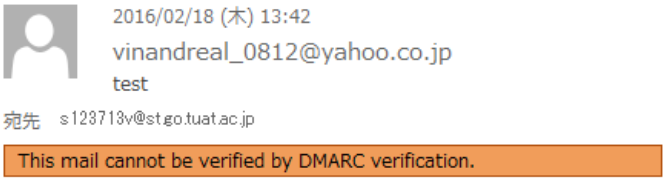


Fig. 9. Addition of “Non-DMARC-compliant” Label

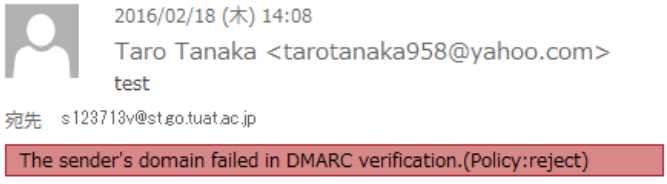


Fig. 10. Addition of “reject” Label

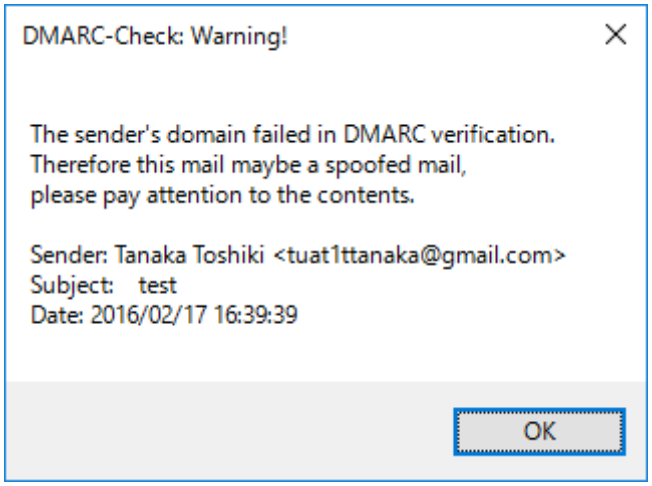


Fig. 11. Pop-up Alert Window

remarkable point of the system is to implement the all functions of sender domain authentication, DMARC verification, and the result notification on a user's PC. By implementing on each user's PC, users can install a spoofed e-mail alert system even if their receiving server does not support DMARC verification. Generally DMARC is used for administrators of sender domain receives the report of sender domain authentication. However, this system is able to alert the spoofed e-mails by visually notifying DMARC verification result to each recipient. Moreover, even when the sender domain was publishing “none” as the DMARC policy, our system can prevent a recipient overlooking the spoofed e-mails by the notification.

This system performs sender domain authentication and DMARC verification in POP proxy, thus the system is only compatible with POP. Therefore e-mail receiving via IMAP has been widely utilized in recent years, support of the mechanism described in this paper to the IMAP environment is a future subject.

ACKNOWLEDGMENT

We would like to thank Mr. Ayachika Kitazaki, who is the vice chairman of anti spam committee of The Internet Association Japan, for providing us data of DMARC policy statistics.

REFERENCES

[1] FBI (Federal Bureau of Investigation): Public Service Announcement, E-mail Account Compromise. [Online]. Available: http://www.ic3.gov/media/2015/150827-2.aspx
[2] M. Wong and W. Schlitt. (2006, Apr.). Sender Policy Framework (SPF) for Authorizing Use of Domains in E-Mail, RFC4408, IETF
[3] D. Crocker, T.Hansen, M. Kucherawy. (2011, Sep.). DomainKeys Identified Mail (DKIM) Signatures, RFC6376, IETF
[4] M. Kucherawy, E. Zwicky. (2015, Mar.). Domain-based Message Authentication, Reporting, and Conformance (DMARC), RFC7489, IETF
[5] M. Fukuyama, M. Oiwa, N. Yamai, N. Kitagawa, “Implementation of DKIM Verification System Using POP Proxy,” IPSJ Technical Report, 2015-IOT-28, No.2, 2015, pp.1-6 (in Japanese).
[6] CPAN:MAIL::DMARC.pm. [Online]. Available: http://search.cpan.org/~msimerson/Mail-DMARC-1.20150527/lib/Mail/DMARC.pm
[7] CPAN:MIME::Parser.pm. [Online]. Available: http://search.cpan.org/~dskoll/MIME-tools-5.506/lib/MIME/Parser.pm
[8] CPAN:Mail::SPF.pm. [Online]. Available: http://search.cpan.org/~jmehnle/Mail-SPF-v2.9.0/lib/Mail/SPF.pm
[9] CPAN:Mail::DKIM::Verier.pm. [Online]. Available: http://search.cpan.org/~jaslong/Mail-DKIM/lib/Mail/DKIM/Verifier.pm
[10] M. Kucherawy. (2013, Sep.). Message Header Field for Indicating Message Authentication Status, RFC7001, IETF

Naoya Kitagawa received his B.Sc. and M.Sc. degree in information science from Chukyo University, Toyota, Japan, in 2009 and 2011 respectively, and his Ph.D. degree in information science from Nagoya University, Nagoya, Japan, in 2014.
In April 2014, he joined Information Technology Center, Nagoya University as a postdoctoral fellow. Since October 2014, he has been an assistant professor in the Institute of Engineering, Tokyo University of Agriculture and Technology. His research interests include the Internet, network security, and distributed system. He is a member of IPSJ.

Toshiki Tanaka received B.E. in computer and information science from Tokyo University of Agriculture and Technology, in 2016.

Masami Fukuyama received B.E. in computer and information science from Tokyo University of Agriculture and Technology, in 2016. Since April 2016, he has been a graduate student in the Graduate School of Engineering, Tokyo University of Agriculture and Technology.

Nariyoshi Yamai received his B.E. and M.E. degrees in electronic engineering and his Ph.D. degree in information and computer science from Osaka University, Osaka, Japan, in 1984, 1986 and 1993, respectively.
In April 1988, he joined the Department of Information Engineering, Nara National College of Technology, as a research associate. From April 1990 to March 1994, he was an Assistant Professor in the same department. In April 1994, he joined the Education Center for Information Processing, Osaka University, as a research associate. In April 1995, he joined the Computation Center, Osaka University, as an assistant professor. From November 1997 to March 2006, he joined the Computer Center, Okayama University, as an associate professor. From April 2006 to March 2014, he was a professor in the Information Technology Center (at present, the Center for Information Technology and Management), Okayama University. Since April 2014, he has been a professor in the Institute of Engineering, Tokyo University of Agriculture and Technology. His research interests include distributed system, network architecture and Internet. He is a member of IEICE, IPSJ and IEEE

Fingerprinting Attack on Tor Anonymity using Deep Learning

Kota Abe and Shigeki Goto

***Abstract*— Tor is free software that enables anonymous communication. It defends users against traffic analysis and network surveillance. It is also useful for confidential business activities and state security. At the same time, anonymized protocols have been used to access criminal websites such as those dealing with illegal drugs. This paper proposes a new method for launching a fingerprinting attack to analyze Tor traffic in order to detect users who access illegal websites. Our new method is based on Stacked Denoising Autoencoder, a deep-learning technology. Our evaluation results show 0.88 accuracy in a closed-world test. In an open-world test, the true positive rate is 0.86 and the false positive rate is 0.02.**

***Index Terms*— Network Security, Tor, Fingerprinting Attack, Deep Learning, Autoencoder**

I. INTRODUCTION

The Onion Router (Tor) is free software that enables anonymous communication. [1, 2]. It defends users against traffic analysis and network surveillance. It is also useful for confidential business activities and state security. At the same time, anonymized protocols have been used to access criminal websites such as those dealing with illegal drugs. There is a need to develop a method that can identify websites when anonymized protocols are used.

This paper proposes a new method for launching a fingerprinting attack to analyze Tor traffic in order to detect users who access illegal websites. Using a fingerprinting attack, we can identify a website that a user accesses on the basis of traffic features such as packet length, number of packets, and time. We can analyze this information from captured packets regardless of encryption. Our new method for fingerprinting attacks is based on Stacked Denoising Autoencoder (SDAE), a deep-learning technology. Our evaluation results show 0.88 accuracy is in a closed-world test. In an open world test, the true positive rate (TPR) and false positive rate (FPR) are 0.86 and 0.02, respectively.

The remainder of this paper is organized as follows. Section II explains the technical background. Section III describes related work. Our new method is proposed in Section IV. Section V shows the evaluation results. Section VI concludes the paper.

Kota Abe and Shigeki Goto are with the Department of Computer Science and Engineering, Waseda University, Shinjuku, Tokyo 169-8555 Japan e-mail: (see http://www.goto.info.waseda.ac.jp).

II. TECHNICAL BACKGROUND

A. *Tor Anonymity*

Tor [1, 2] is a popular anonymized protocol. Figure 1 shows an example of a Tor configuration. At the initial setting, there are three nodes between a user and a web server, as shown in Figure 1. Tor traffic data is encrypted using Transport Layer Security (TLS) between a user and each Tor node. Thus, Tor nodes do not know the original plain data, with one exception. The closest node to the web server can read the original data without encryption. In a Tor configuration, each node knows only the Internet Protocol (IP) addresses of adjacent nodes that are directly connected to the node.

In the Tor protocol, content data is encapsulated into a series of *cells*, each with a fixed length of 512 bytes. It is difficult to estimate the original content only from the packet length.

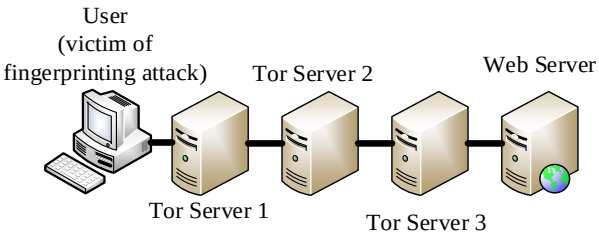


Fig. 1. Configuration of Tor

B. *Fingerprinting Attacks on a Website*

1) *Fingerprinting*

A website fingerprinting attack aims to detect a website even if the traffic is encrypted using Tor or a virtual private network (VPN). We cannot specify the website by inspecting the encrypted payload. However, we can utilize the packet information, such as packet length, number of packets, and time. In a fingerprinting attack, we can specify a website by providing the packet information.

There are two methods for capturing traffic data in Tor. In the first method, an attacker (analyzer) prepares an entry node

of Tor and captures the traffic through this node. However, the Tor protocol selects nodes at random. It is unlikely that a specific victim connects to the attacker's node. In the second method, an attacker (analyzer) is a network operator, such as an Internet service provider (ISP). He or she can capture traffic packets between a victim and the entry node of Tor. This is a realistic scenario. This paper proposes a new approach using the second method.

2) Closed- and Open-World Tests

There are two evaluation schemes for fingerprinting attacks. The first scheme is a *closed-world* test. It conducts a test in which a victim can access only a limited number of websites, which the attacker attempts to detect. For example, an attacker might prepare 100 monitored sites and investigate the features of these 100 websites. The victim can access only these 100 websites.

The second scheme is an *open-world* test. In such a test, a victim can freely access any websites on the Internet. The attacker must be able to determine whether a website is monitored or non-monitored. If it is a monitored website, the attacker must be able to determine which website among the 100 monitored sites it is. This paper uses two evaluation schemes, closed and open.

C. SDAE

Deep learning is an attractive method in machine learning. It is called *deep* because it utilizes a multiple-layered neural network. An autoencoder is a deep-learning technique. This paper uses SDAE.

An autoencoder is a neural network that consists of input, hidden, and output layers. Figure 2 shows an example of an autoencoder. It calculates weights on directed edges in Figure 2 by learning from input data. One specific autoencoder feature is that the input data (vector) and the output data (vector) must be equal.

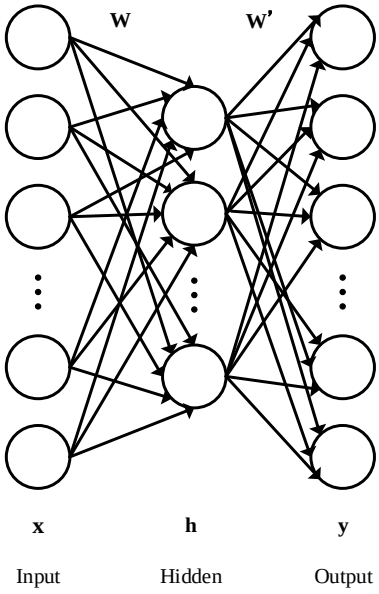


Fig. 2. Structure of an Autoencoder

An autoencoder is represented by a mathematical formula. In formula (1), the input layer is represented as a vector $\boldsymbol{x}$, the output of the hidden layer as a vector $\boldsymbol{h}$, and weights from the input layer to the hidden layer as a matrix $\boldsymbol{W}$ and vector $\boldsymbol{b}$. The vector $\boldsymbol{b}$ represents bias terms. We also define an activation function f. Data propagation from the input layer to the hidden layer is calculated using formula (1).

$$\boldsymbol{h} = f(\boldsymbol{W}\boldsymbol{x} + \boldsymbol{b}) \quad (1)$$

Similarly, we define the output from the output layer as a vector $\boldsymbol{y}$, and the weights from the hidden layer to the output layer are represented as a matrix $\boldsymbol{W}'$ and vector $\boldsymbol{b}'$. The vector $\boldsymbol{b}'$ consists of bias terms. We also define an activation function f'. Data propagation from the hidden layer to the output layer is calculated using formula (2).

$$\boldsymbol{y} = f'(\boldsymbol{W}'\boldsymbol{h} + \boldsymbol{b}') \quad (2)$$

The autoencoder determines the weights $\boldsymbol{W}$ and $\boldsymbol{W}'$ that equalize the input $\boldsymbol{x}$ and output $\boldsymbol{y}$. The weights are calculated using formula (3), which minimizes the difference between the input data $\{\boldsymbol{x_i}, \ldots\}$ and output $\boldsymbol{y}$.

$$\min_{\boldsymbol{W},\boldsymbol{b},\boldsymbol{W}',\boldsymbol{b}'} \sum_i \|\boldsymbol{x_i} - f'(\boldsymbol{W}'f(\boldsymbol{W}\boldsymbol{x_i} + \boldsymbol{b}) + \boldsymbol{b}')\|_2^2 \quad (3)$$

Using an autoencoder, we can decrease the dimensions of data vectors. The dimension of $\boldsymbol{h}$ is less than that of $\boldsymbol{x}$ or $\boldsymbol{y}$. The output vector $\boldsymbol{h}$ of the hidden layer is used as a feature vector in machine learning.

We can combine multiple autoencoders by overlapping a hidden layer as an input of the second autoencoder. This type of autoencoder is called Stacked Autoencoder (SAE). Figure 3 shows an example.

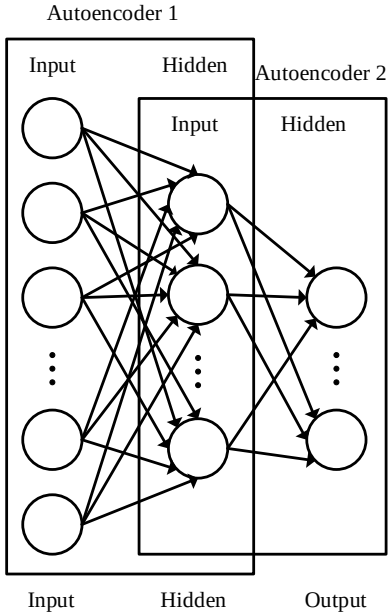


Fig. 3. Structure of a Stacked Autoencoder

It can be meaningful to add *noise* to an input vector. This type of autoencoder is called a denoising autoencoder (DAE). By adding noise data, an autoencoder can avoid overlearning or overfitting, with the result that formula (3) is satisfied only for the training data. Noise is sometimes useful to generalize the training data. A DAE can attain higher accuracy.

We can further combine multiple DAEs similarly to SAEs, This type of autoencoder is called SDAE. This paper uses SDAEs. We use Pylearn2 software [3] as a deep-learning tool.

III. RELATED WORK

A. Optimal String Alignment Distance (OSAD)

In 2013, Wang and Goldberg [4] conducted a fingerprinting attack using OSAD. In their method, a sequence of Tor cells is treated as a string. If two instances of a cell string are captured for the same site, the distance between the two instances is small. If they are captured for two different sites, the distance of the two instances is large. Wang and Goldberg used OSAD in an algorithm to calculate the distance.

Wang and Goldberg used this distance as the *kernel matrix* in a support vector machine (SVM). They defined the distance and the kernel by formulas (4) and (5), respectively. s_1 and s_2 are two strings, and the distance between s_1 and s_2 is $D(s_1, s_2)$.

$$D'(s_1, s_2) = \frac{D(s_1, s_2)}{min(|s_1|, |s_2|)} \quad (4)$$

$$K(s_1, s_2) = e^{-D'(s_1, s_2)^2} \quad (5)$$

When $D' = 0$, two strings are equal, and K becomes one. When the distance between two strings is large, K becomes small. When $D \to \infty$, the limit of K becomes zero. Therefore, we can use K as the kernel matrix of an SVM. Wang and Goldberg used the one-against-one method in their SVM. This method is used for multi-class classification by repeating two-class classifications and by performing majority voting.

B. k-Nearest Neighbor Algorithm (k-NN)

In 2014, Wang et al. [5] proposed another fingerprinting attack using the k-nearest neighbor (k-NN) algorithm. In their new method, they extract features from captured packets.

- General features (total transmission size, total transmission time, and numbers of incoming and outgoing packets)
- Packet ordering
- Concentration of outgoing packets
- Bursts

Some features are more meaningful than others. Then, they determine the weights of features. Finally, they classify test data using the k-NN method with features and weights.

IV. NEW METHOD

A. Dataset for Learning and Evaluation

This paper uses the same dataset as that of Wang [6] in our evaluation experiment. This dataset contains 100 sites as *monitored* web sites and 9,000 sites as *non-monitored* sites. Monitored sites are used in the closed-world test. Non-monitored sites are used in the open-world test. Each monitored site has 90 instances (cells), and each non-monitored site has one instance. Monitored sites consist of porn sites, Bit Torrent trackers' sites, and sites that have religious or political contents. Access to these sites is blocked in China, United Kingdom, and Saudi Arabia. Non-monitored sites consist of Alexa's list [7], which covers ordinary popular web pages. In Figure 4, the first column records when a cell is captured. The timestamp unit is seconds. The time at which the first cell is sent is 0.0. The second column indicates the direction of a cell. When a cell is sent from a victim (target) to a Tor node, it is represented as 1. When a cell is sent from a Tor node to a victim, it is represented as -1. This time sequence starts when the web page begins loading and ends when the last cell is sent.

0.0	1
0.0	1
0.116133928299	1
0.499715805054	-1
0.499715805054	-1
0.782404899597	-1
0.969846963882	-1
0.969846963882	-1
0.969846963882	-1
0.969846963882	-1

Fig. 4. Example of dataset.

We count the number of cells in a packet. Since the size of a cell is fixed at 512 bytes, the number of cells is counted by dividing the packet length by 600. We use not 512 but 600 because we consider inter-cell headers and the overhead [10] .Tor sends cells for flow control at regular intervals. Such a control cell is called a SENDME cell. SENDME cells are not useful in fingerprinting attacks. We exclude SENDME cells from the dataset.

B. Proposed Method

First, an attacker (analyzer) collects training data for machine learning. The attacker accesses websites he or she wants to monitor through Tor fingerprinting and then captures the traffic data repeatedly, e.g.,., 100 times. The attacker also collects traffic data from a large number of other websites. The data is used for the open-world test. Since this paper uses Wang's dataset, we can omit the data collection phase.

Next, the attacker extracts Tor cells from the captured data. These are used as input to the autoencoder. Again, we can omit this phase, because we use the same dataset as that in Wang's method. Tor cells are already extracted. Then, we sort out data to create an input vector to the autoencoder. This paper uses the direction of a cell as an element of an input vector. It is a simple method. We do not use other features. It should be noted here that input vectors have a fixed length (dimension). The original traffic data have a variety of lengths (dimensions) according to the traffic pattern. We truncate a sequence of cells if the length is greater than 5,000. If the number of cells is less than 5,000, we put 0 as dummy padding to create a vector of size 5,000. Figure 5 shows an example of input data corresponding to Figure 4.

1, 1, 1, −1, −1, −1, −1, −1, −1, −1, −1,
−1, −1, 1, −1, 1, 1, … , 0, 0, 0, 0, 0

Fig. 5. Example of input data

After preparing the data, the attacker conducts training using the SDAE. In our experiment, we specifically use a multilayer perceptron (MLP) that has two layers of SDAEs and an output layer realized by a *softmax* function. The parameters of the SDAE and MLP will be shown in the next section (V). Before inputting the training data, we randomize the order of the training vectors in the data set. If a *batch* has many similar vectors, the efficiency of learning might be decreased.

The *test* data for evaluation is prepared similarly to the *training* data.

V. Evaluation

A. Environment

Table 1 shows the experimental environment. We use Compute Unified Device Architecture (CUDA) [8] to accelerate the training using a graphical processing unit (GPU). Table 1 shows the machine specification, which includes a GeForce GTX 750 Ti graphics card by NVIDIA.

TABLE 1
ENVIRONMENT OF EXPERIMENT

OS	Ubuntu 14.04.03 LTS
CPU	Intel Core i7-4790
RAM	32 GB
GPU	NVIDIA GeForce GTX 750 Ti

B. Closed-World Test

1) Overview

In the closed-world test, the dataset contains 100 monitored sites, with each site containing 90 cell instances. Seventy-two instances are used for training data, and 18 instances for test data. This closed-world test is a multi-class classification. We labeled monitored websites as class 0 to class 99.

2) Layer

We used an MLP with two layers of SDAEs and with the output layer realized by a softmax function. Parameters of Pylearn2 are shows in Tables 2 and 3.

Nvis and *nhid* are the dimensions of the input and hidden layers of the Autoencoder, respectively. Learning rate is a coefficient during the weight-training phase.

TABLE 2
PARAMETERS OF SDAE (CLOSED-WORLD TEST)

Parameter	First Layer	Second Layer
nvis	5000	500
nhid	500	125
learning_rate	0.001	0.001
batch_size	50	50

TABLE 3
PARAMETERS OF MLP (CLOSED-WORLD TEST)

Parameter	MLP
Nvis	5000
n_classes	100
learning_rate	0.005
batch_size	200

3) Results

We conducted a series of closed-world tests while changing the number of learning sessions (*max_epoch*) every five times. Values of *max_epoch* in each DAE and the output layer are the same. Figure 6 shows the accuracy of the closed-world tests. The highest accuracy of 0.88 is attained when the number of learning sessions is 50.

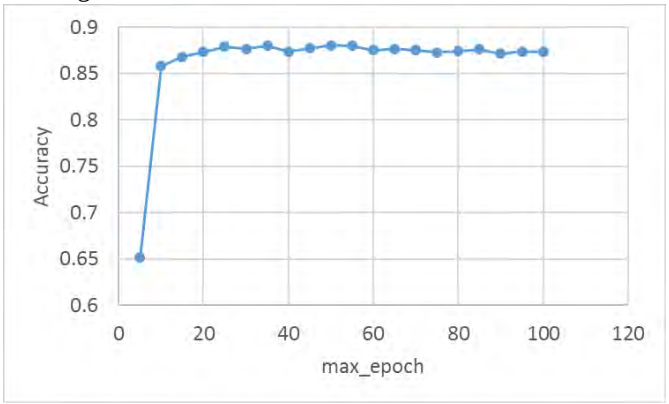


Fig. 6. Relation between max_epoch and accuracy in closed world test.

We also conduct a series of closed-world tests by changing the dimensions, *nvis* and *nhid*. We fix the *max_epoch* value as 50. The results are shown in Table 4. There is no major change in the accuracy by the dimension parameters of the hidden layer of the SDAE. The maximum accuracy is attained when the *nhid* values of the first and second layers are 1,000 and 500, respectively. When the *nhid* values are 500 for the first layer and 125 for the second, the results are similar.

TABLE 4
RESULTS WHEN CHANGING NVIS AND NHID
(CLOSED-WORLD TEST)

Accuracy		1st layer			
		250	500	750	1000
2nd	125	86.4	88.1	86.9	86.9
layer	250	87.1	87.2	87.6	87.6
	500	-	87.3	87.2	88.2
	750	-	-	87.9	87.3
	1000	-	-	-	87.6

4) Execution time

Table 5 shows the execution time when *max_epoch* is set as 50. In this experiment, the autoencoder can use the weight that is already learned. Then, the test time is very short. The training time is also short, because the autoencoder does not need to perform multiple layers of backpropagation.

TABLE 5
EXECUTION TIME (CLOSED-WORLD TEST)

Process	Description	Time [s]
Data Transmission	Time to convert train data and test data to Pylearn2 format.	124.4
Learning Time	Time to train using 7200 train data.	163.0
Test Time	Time to test 1800 Test data	3.0

5) Three-layer SDAE

The above results are obtained for the closed-world test by

the two-layer SDAE. It is worthwhile investigating the performance of the three-layer SDAE.

We conduct another closed-world test using a three-layer SDAE. Table 6 shows the parameters of the new SDAE. The parameters of the MLP are the same as those in Table 3.

TABLE 6
PARAMETERS OF THREE-LAYER SDAE (CLOSED-WORLD TEST)

Parameter	1st Layer	2nd Layer	3rd Layer
nvis	5000	750	500
nhid	750	500	250
learning_rate	0.001	0.001	0.001
batch_size	50	50	50

Figure 7 shows the results while changing the learning intervals (*max_epochs*) every ten times. The maximum accuracy is 0.88, the same accuracy achieved by the two-layer SDAE. However, when three layers are used, the convergence of learning becomes slow compared with the two-layer SDAE.

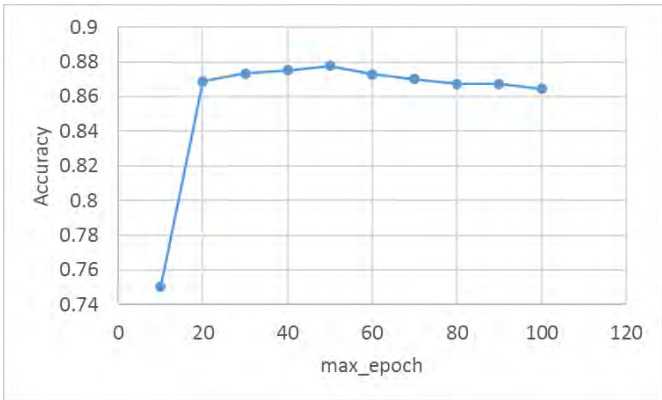


Fig. 7. Relation between max_epoch and accuracy in closed-world test.

The learning and test times also become slow. For the three-layer SDAE, when *max_epoch* is 50, the learning time becomes 241.9 s and the testing time becomes 3.6 s.

C. Open-World Test

1) Overview

In the open-world test, we use the data not only of 100 monitored sites, but also those of 9,000 non-monitored sites. The data of the monitored sites is divided into 72 instances for training data and 18 instances for testing data. We use 1,800 instances of non-monitored sites as the testing data.

A non-monitored website never appears in the training data. The victim can access a new website that the attacker does not expect. We label monitored websites as classes 0 to 99 and all the non-monitored websites as a single class 100.

2) Layer

In the open-world test, we use an MLP that has an input layer with a dimension of 5,000 and a two-layer DAE. The output layer is realized by a softmax function. The parameters of Pylearn2 are showed in Tables 7 and 8.

TABLE 7
PARAMETERS OF SDAE (OPEN-WORLD TEST)

Parameter	First Layer	Second Layer
Nvis	5000	500
Nhid	500	125
learrning_rate	0.001	0.001
batch_size	50	50
max_epoch	30	30

TABLE 8
PARAMETERS OF MLP (OPEN-WORLD TEST)

Parameter	MLP
Nvis	5000
n_classes	101
learning_rate	0.005
batch_size	200
max_epoch	50

3) Results

We investigate the TPR, i.e., the rate at which non-monitored websites are classified correctly, and the FPR, i.e.,., the rate at which a non-monitored website is classified as a monitored site. The TPR is shown in Figure 8, and the FPR is shown in Figure 9.

In Figure 8, when the number of training data instances of non-monitored sites is larger, the TPR is lower. The maximum TPR is 0.87, when the number of the training data of non-monitored sites is 1,000, and the minimum TPR is 0.86, when the number is 7,000. In Figure 9, in addition to the TPR, when the number of training data instances of non-monitored sites is larger, the FPR is lower. The minimum FPR is 0.02, when the number of training data of non-monitored sites is 7,000.

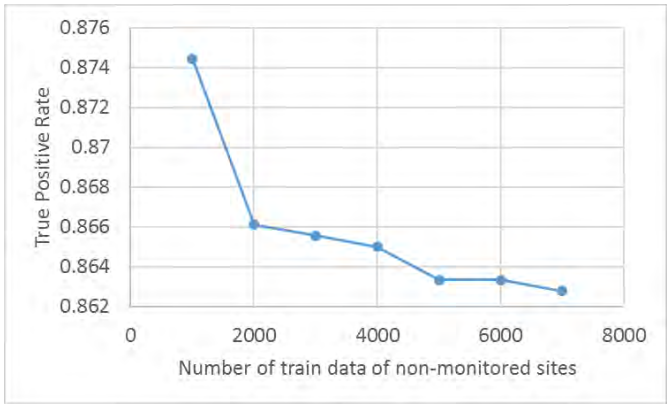


Fig. 8. Relation between number of training data of non-monitored sites and TPR in the open-world test.

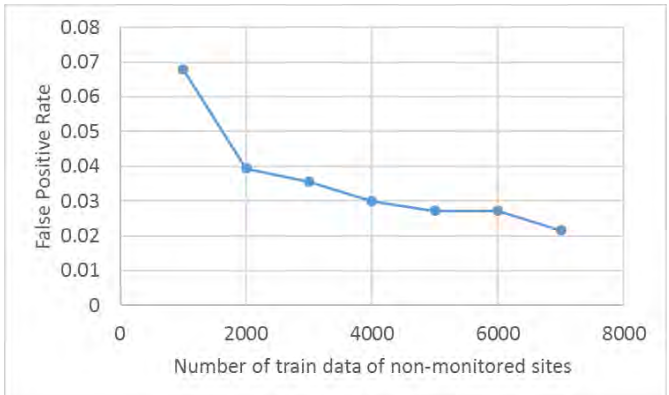


Fig. 9. Relation between number of training data of non-monitored sites and FPR in open-world test.

There is a trade-off between TPR and FPR. It is better to have a high value of TPR, while keeping the FPR value low.

Comparison with Related Work

Wang et al. showed the results of the OSAD [4] and k-NN methods [5] using the same dataset. Table 9 shows the comparison with previously known methods and our method.

In our proposed method, the accuracy in the closed-world test is 0.88, slightly lower than those of OSAD and k-NN. In the open-world test, our TPR (0.86) is higher than that of OSAD, and our FPR (0.02) is lower than that of OSAD. However, our FPR is higher than that of the k-NN method.

TABLE 9
COMPARISON WITH EXISTING METHODS

Method	Accuracy in Closed World Test	TPR in Open World Test	FPR in Open World Test
Our Method	0.88	0.86	0.02
OSAD Method	0.90	0.83	0.06
k-NN Method	0.91	0.85	0.006

VI. CONCLUSION

A. Summary

Here we propose a new method for fingerprinting attacks on Tor anonymity using SDAE. The input vector takes a very simple form, with elements 1, −1, or 0. The evaluation results show an accuracy of 0.88 in the closed-world test and TPR and FPR values of 0.86 and 0.02, respectively, in the open-world test. It is the advantage of our method that we can realize a high accuracy without selecting the features manually. Out method is based on mechanical Deep Learning.

This paper shows that deep-learning technology can be applied to fingerprinting attacks on Tor communications to have results comparable to those of existing technologies.

B. Future Research

It may be meaningful to combine our method with other methods proposed in related work. For example, the output of SDAE can be used as features in Wang's method and used for training by k-NN.

There are convolutional neural networks (CNNs) and recurrent neural networks (RNNs) in deep learning. CNN has been used in pattern recognition. RNN can handle time-series data. It may be possible to improve the accuracy of our method by applying other neural network technologies as well.

ACKNOWLEDGEMENTS

A part of this work was supported by JSPS Grant-in-Aid for Scientific Research B, Grant Number 16H02832.

REFERENCES

[1] The Tor Project, "Tor Project: Anonymity Online," https://www.torproject.org/, referred Jan. 20, 2016.
[2] Roger Dingledine, Nick Mathewson and Paul Syverson, "Tor: the second-generation onion router," in *Proceedings of the 13th USENIX Security Symposium*, 2004, pp. 303–320.
[3] the LISA lab, "Welcome — Pylearn2 dev documentation," http://deeplearning.net/software/pylearn2/, referred Jan. 20, 2016.
[4] Tao Wang and Ian Goldber, "Improved Website Fingerprinting on Tor," WPES '13 Proceedings of the 12th ACM workshop on Workshop on privacy in the electronic society, 2013, pp. 201–212.
[5] Tao Wang, Xiang Cai, Rishab Nithyanand, Rob Johnson and Ian Goldberg, "Effective attacks and provable defenses for website fingerprinting," in *23th USENIX Security Symposium*, 2014, pp. 143–157.
[6] Tao Wang, "Website Fingerprinting," https://cs.uwaterloo.ca/~t55wang/wf.html, referred Jan. 20, 2016.
[7] Alexa, "Alexa - Actionable Analytics for the Web," http://www.alexa.com/, referred Jan. 20, 2016.
[8] NVIDIA, "Parallel Programming and Computing Platform − CUDA − NVIDIA − NVIDIA," http://www.nvidia.com/object/cuda_home_new.html, referred Jan. 20, 2016.
[9] Andriy Panchenko, Lukas Niessen, Andreas Zinnen and Thomas Engel, "Website Fingerprinting in Onion Routing Based Anonymization Networks," in *WPES '11 Proceedings of 27 the 10th annual ACM workshop on Privacy in the electronic society*, 2011, pp. 103–114.
[10] Xiang Cai, Xin Cheng Zhang, Brijesh Joshi, and Rob Johnson, "Touching from a distance: website fingerprinting attacks and defenses," *in CCS '12 Proceedings of the 2012 ACM conference on Computer and communications security*, 2012, pp. 605–616.
[11] Hideki Asoh, Muneki Yasuda, Shiniti Maeda, Daisuke Okanohara, Takayuki Okatani, Yotaro Kubo and Danushka Bollegala, "Deep Learning," Kindai kagaku sha, Tokyo, 2015.
[12] Takayuki Okatani and Masaki Saito, "Deep Learning," IPSJ SIG-CVIM: Computer Vision and Image Media, 2013, pp.1–17.

Kota Abe Kota Abe received the B.S. degree in Computer Science and Engineering from Waseda University in March, 2016. He is now a master student at Department of Computer Science and Communications Engineering, Waseda University. His research interest covers Cyber Security.

Shigeki Goto Shigeki Goto is a professor at Depart-ment of Computer Science and Engineering, Waseda University, Japan. He received his B.S. and M.S. in Mathematics from the University of Tokyo. Prior to becoming a professor at Waseda University, he has worked for NTT for many years. He also earned a Ph.D in Information Engineering from the University of Tokyo. He is the president of JPNIC. He is a member of ACM and IEEE, and he was a trustee of Internet Society from 1994 to 1997.

Proceedings of the APAN – Research Workshop 2016
ISBN 978-4-9905448-6-7

An Analysis of Botnet Attack for SMTP Server using Software Define Network (SDN)

Mohd Zafran Abdul Aziz[1,2], Koji Okamura[3]
[1]Faculty of Electrical Engineering,Universiti Teknologi Mara,40450, Shah Alam, Selangor. Malaysia
[2]Department of Advanced Information Technology, Graduate School of Information Science and Electrical Engineering, Kyushu University, Japan
[3]Research Institute for Information Technology, Kyushu University, Japan
E-Mails: zafran.fke@gmail.com, oka@ec.kyushu-u.ac.jp

***Abstract*— SDN architecture overwhelms traditional network architectures by software abstraction for a centralize control of the entire networks. It provides manageable network infrastructures that consist millions of computing devices and software. In this work, we present multi-domain SDNs architecture with an integration of Spamhaus server. The proposed method allows SDN Controllers to update the Spamhaus server with latest detected spam botnet signatures. It can prevent any botnet attack from entering other SDN domains. We also discussed a method for analyzing SMTP traffic using a decision tree algorithm. We used Mininet tool to simulate the multi-domain SDNs with the Spamhaus server. The simulation results have shown that a packet Retransmission Timeout (RTO) between server and client can detect the SMTP spam frames.**

***Index Terms*—SDN, Software Define Network, SMTP, Spam, Botnet, SDN Security, OpenFlow, Mininet**

I. INTRODUCTION

SDN is an architecture for multi devices communication in integrated networks. It provides manageable network infrastructures that consist millions of computing devices and software. Due to growing of device connectivity and speeds, tradition networks such as LANs and WANs are no longer capable of optimizing all connectivity (e.g. network routing) and to secure networks from multi-faceted security threats. Traditional firewall and IDS are not capable of preserving a large network such as monitoring all inbound and outbound packets because the internet data is too huge to be monitored. Cloud Computing, Bigdata and IoT create deadly network traffics for the traditional network architecture, which it will cause an obsoleting and soon it will cripple the existing network functionality. SDN is one of a promising architecture that allows huge WANs/MANs to be controlled using a high-level of abstraction. The SDN architecture splits the centralize control of the entire networks (control plane) from an actual network data and routing process (data plane). All network behavior will be programmed in the centralize control using programmatic software such as SDN Application and Controller. The SDN architecture also provides a centralized security control that can help to prevent illegitimate access or network attacks such as DDos.

In this work, we present multi-domain SDNs architecture with an integration of Spamhaus server. The proposed method allows SDN Controllers to update the Spamhaus server with latest detected spam signatures. It can help to prevent any spam email from entering others SDN domains. We also discussed the method for analyzing SMTP spam frames using a decision tree algorithm. We divided this work into six sections. The first Introduction section provides an introduction to SDN and traditional network architecture. It follows the Related Works section that discusses SDN and STMP attack using botnets. After that, we discuss methodology adopted to prevent spam in SMTP protocol in the Methodology section. In the Simulation Setup section, we simulate the proposed method using an actual data in Mininet tool. We present simulation results and discussion using the Mininet in the Results and Discussion section. Finally, we conclude this work and propose a future work in the Conclusion section.

II. RELATED WORKS

This section presents related works:

A. Software Define Network (SDN)

SDN is an architecture for multi devices communication in integrated networks. In the initial stage, it allows multiple LANs devices and systems to be integrated into WAN networks. The first SDN began after Java language released by Sun Microsystem, which AT&T Labs Geoplex project used Java to program APIs to implement middleware networking [1]. The Geoplex provided open networking

standard for network integrations and communications such as system managements and provisions, integrated security and system authentication, network monitoring etc. The most prominent functionality of the Geoplex is it allows network IPs to be mapped to one or many system and services [2]. In 2008, research and development for SDN continue by UC Berkeley and Stanford University [3]. By 2011, Open Networking Foundation (ONF) continues to develop OpenFlow for SDN [4]. The ONF provides SND resources (e.g. switch specification) for product manufacturer and software developer to implement SDN using the OpenFlow's standard and protocol [5].

Figures 1 and 2 show a general SDN architecture and its stacks. In SDN topology, all network nodes or devices are controlled using a control plane. The architecture splits the control plane from actual network data and routing process (data plane). The infrastructure layer communicates with SDN Controller using Control Data Plane (CDP) API (e.g. OpenFlow). All nodes or routers in the SDN network will use the CDP API for all control plane communication. The control layer consists of SDN Control Software or Controller, which extract information from the infrastructure layer such as a list of all devices in the SDN network and its states. It does not provide the entire information of all connected devices, but it provides an abstract view of the SDN network and topology. The application layer uses information from the control layer for a network abstraction administrative such as network analytics; network, system and topology managements etc. [6,7].

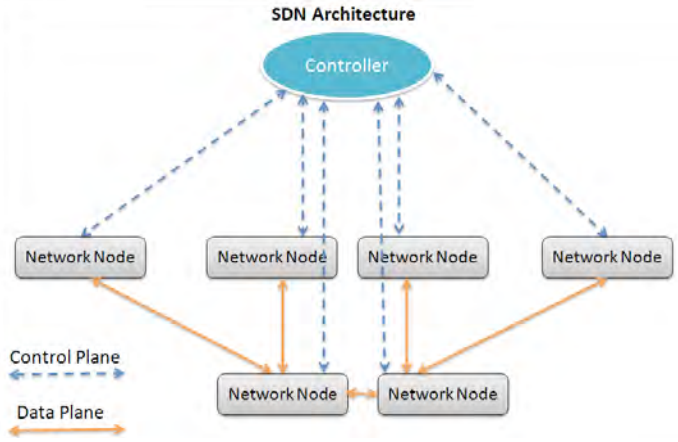


Fig. 1. SDN architecture [8].

Many SDN runs over a virtualized architecture, which the application and control layers may execute in various devices that including a virtual machine in cloud computing [10,11]. This allows application and control layers to be distributed on various computing platforms, which it will increase flexibility, mobility and computing power using the virtualized architecture, system and devices [12–14].

In this work, we will not discuss the advantage of SDN in distributed systems, but we want to assess a network security through SDN. The next subsection will discuss further the network security and threats in the SDN.

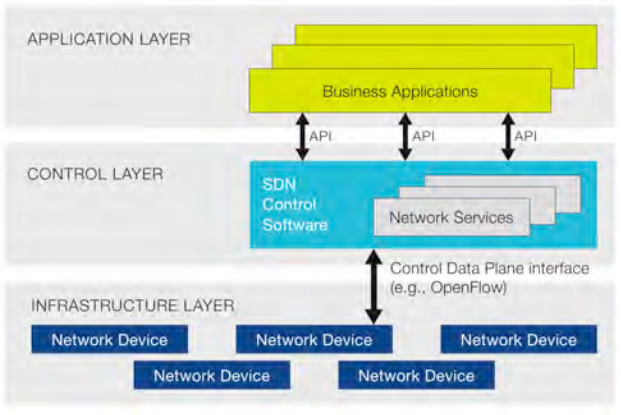


Fig. 2. SDN's stacks [9].

B. *Network Security by SDN*

Distributed systems such as cloud computing and Internet of Things (IoT) are not the main factors for organizations to migrate theirs network infrastructure into SDN, another main reason is a network security that offered by the SDN [15,16]. The SDN allows an abstraction of network security that provides a central authority in a network, which previously hard to be done by traditional distributed networking systems and infrastructures [4,5]. There are also new security problems introduces by an implementation the SDN in network infrastructure, but we are not going to discuss in this publication and one may refer to [16–19] for further examinations regarding these security problems. The following paragraphs will discuss security threats and its countermeasures using SDN.

N. Hoque et al. [20] discuss tools use by attackers and network administrators in SDN. Major attacks on SDN are Dos and DDos [21] that mounted by botnets [22]. Most botnets will try to prevent access to computing resources in the SDN by draining computing capability of the target computing system. An attacker(s) frequently used SYN-Flooding Attack [23], which sends a flood of TCP/SYN packets (by zombie machines) and leave the 3-ways TCP handshake protocol hang-up without ACK packets. This attack applied to all application protocols that are used TCP based connections such as SMTP, FTP, HTTP, DNS etc. Traditional network security systems and infrastructures rely on Intrusion Detection System (IDS) and firewall to protect LAN, WAN from the internet. It might work well for a small and manageable network such as LAN, but not for multi-WANs in a large organization (or a join of multiple organizations) in distance geographical locations. Furthermore, applying SDN for the entire internet is far away than a current topic, which requires, at least a successful implementation of SDN for multi-WANs. We skipped this part, but we want to narrow down our discussion that to improve an efficiency for botnet attack detections on SMTP protocol. The next paragraph will explore the existing methods in preventing the botnet attacks on SMTP protocol.

The most common way to detect botnet attacks are using a signature-based of known attacks [24], and a real-time detection of network anomalies [24,25] using IDS. Both

methods used congestion control and drop packet to block DDos attacks, which called Pushback method [21]. The signature-based requires others systems to provide the signature of known attacks, which can be derived from the integrate the multi-domain SDNs with Spamhaus server. S. Seeber et al [33] proposed to use the existing database (spam signatures) to secure SDN domain. We propose to integrate the Spamhaus server with multi-domain SDNs, which allow

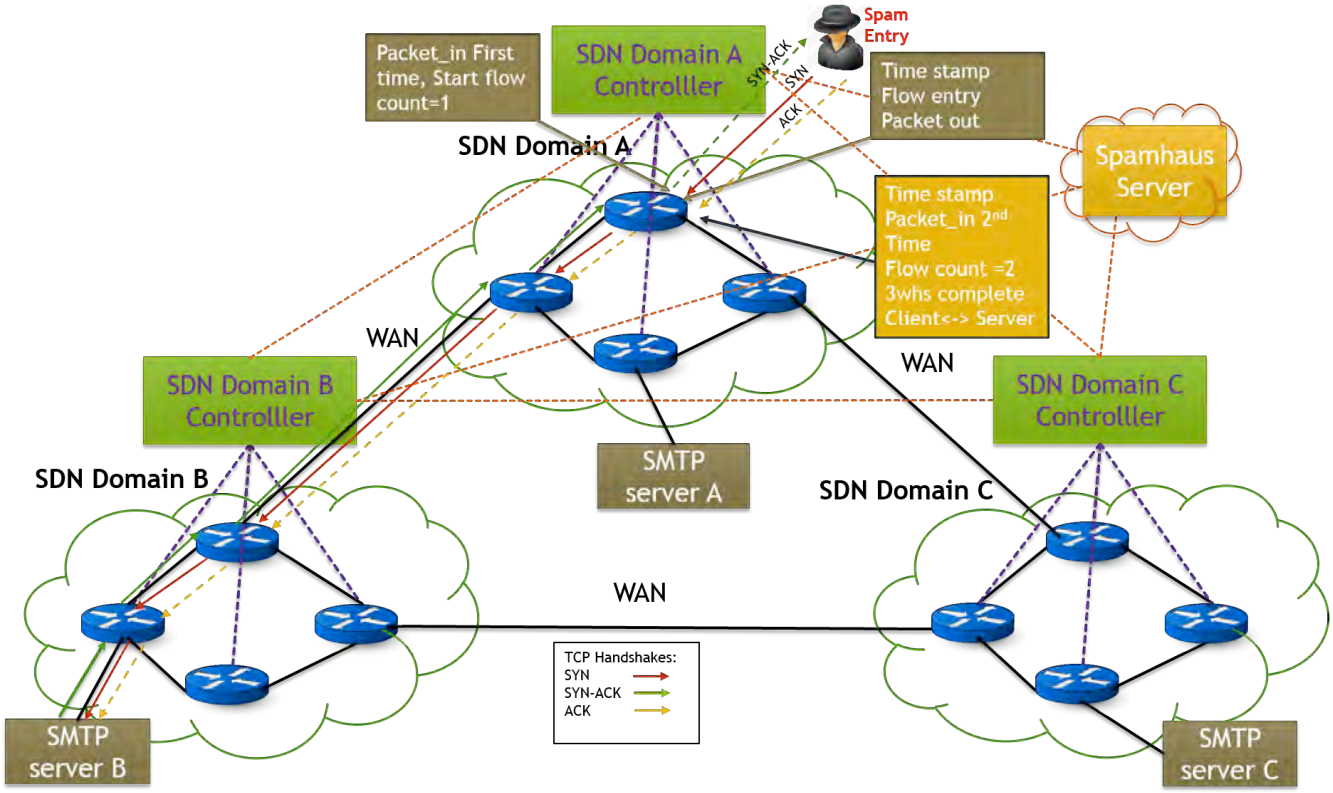


Fig. 3. Integrated Spamhaus in multi-domain SDNs.

real-time detection from a shared database. Routers within the same LANs/WANs may share or distribute attack signatures, for examples: a list of blacklisted source and destination IPs, payloads, Time-to-Live (TTL) [26] etc. Another method to detect potential attacks is using a network traffic classification. It can help to identify packets send by botnets at local and enterprise networks [27]. This method may be integrated into the real-time detection method.

In this work, we used Round-Trip Time (RTT) and Retransmission Timeout (RTO) to detect an anomaly in SMTP traffic, which similar to works done by [27–32]. We enhance the existing detection methods using a new decision tree algorithm for improving detection efficiency. Second, we integrated Spamhaus [33] into SDN for botnet detections using botnet controller lists (BCL). The BCL is shared among SDN domains. The Spamhaus server will serve all SDN Domain Controllers with up-to-date BCL. We discuss the proposed solutions in the Methodology section.

III. Methodology

This section will present the problem statements and proposed solutions in this work. Based on latest literature as aforementioned, this work explores botnet detection on SMTP protocol in SDN. RTT and RTO are used for anomaly detection in SMTP traffics. However, the literature did not SDN Controllers to update the Spamhaus with latest BCL. This will mitigate any botnet attack on SMTP server from entering others SDN domains because all SDN domains will have the latest BCL from the Spamhaus server.

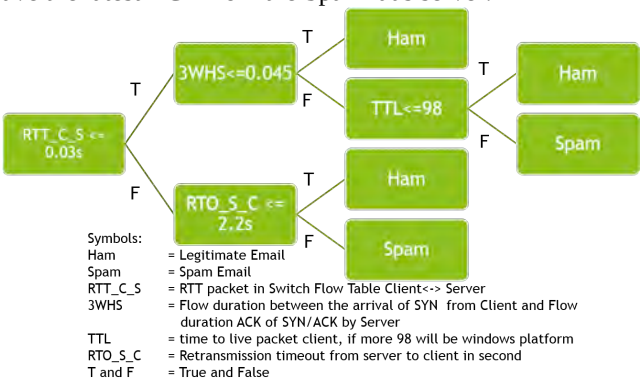


Fig. 4. Decision tree

Figure 3 show the proposed method for the Spamhaus implementation in multi-domain SDNs. For example, a bulk botnet attack on SMTP server being executed by botnets in Domain A. SDN Controller in the Domain A will verify all SMTP frames using information from the Domain A Controller. The Domain A will have the latest BCL because the Domain A Controller is directly connected to the Spamhaus server. Meanwhile, SDN Controller in the Domain

A begins to learn and detect anomaly traffics in the Domain A. The SDN Controller will use the existing algorithms and also the proposed decision tree algorithm to analyze the SMTP frames as shown in Figures 4 and 5. The SDN controller in Domain A will block all anomaly traffics using both the existing and proposed algorithms. Latest updates on anomaly signatures for SMTP frames are forwarded to the Spamhaus server. This will enable BCL sharing among multi-domain SDNs.

Module 0
If dst port==25, Then
  Forward to controller
  Packet_in Flow count
  Go to module 1
Else drop the packet

Module 1 (RTT Client-Server)
If rtt client <-> server between two switch t>= 0.0087s, Then
  Go to module 2
Else
  Go to module 3

Module 2 (3 ways handshake flow count and time)
If flow count packet_in == 2, same src ip, same dist ip, time arrival for 2nd flow <= 0.087 between client <-> server, Then
  Install the flow in flow table, forward the next packet
Else
  Go to module 4

Module 3 (RTO Server-Client)
If RTO from server < 2.2 second, Then
  Install the flow in the flow table and forward the next packet
Else
  Blacklist the ip source send information to Spamhaus

Module 4 (TTL)
If ip TTL<= 96, Then
  Install the flow in the flow table and forward the next packet
Else
  Blacklist the ip source send information to Spamhaus

Fig. 5. Decision tree algorithm

IV. Simulation Setup

This section discussed the simulation setup using Mininet [34]. It allows one to create a virtual network and its components. The Mininet being used by OpenFlow for SDN simulation [35]. Figure 3 shows the overview architecture of simulation setup for this work. The simulation used the internet traffic dataset from University New Brunswick (UNB), Canada [36]. The same dataset was used by E. B. Beigi et al. [32] for botnet detection in their publication. Figures 6 and 7 show the simulation of the dataset using Mininet.

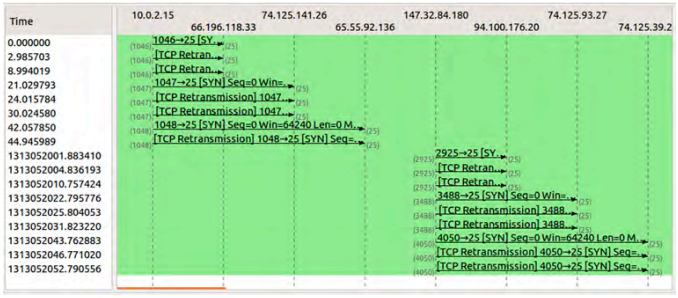


Fig. 6. A flow graph of SYN flood

Figures 8 and 9 show two traffics from seven traffic datasheets that were tested in the simulation. Figures 10 and Table 1 show the summary of max RTT and RTO for seven traffic datasheets. These results can be used to identify botnet SMTP attack packets in a network. Refer to the decision tree in Fig. 4, any packet does not satisfy the decision tree is dropped from the SDN domain.

Refer to the Botnet training and testing columns, any packet RTO between server and client greater than 2.2 seconds (a baseline from botnet training), the packet must be dropped.

The RTO and RTO2 (2nd time runs of the RTO) provided significant results for a botnet detection. The 3WHS is expected to be less or equal to 0.045 second, which provides an unimportant timing for a botnet detection.

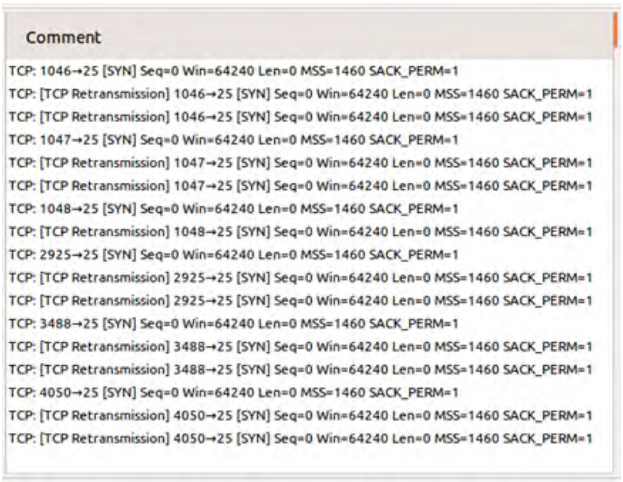


Fig. 7. A flow graph botnet for SYN flood (comment)

V. Results & Discussion

Refer to the Jun-12 until Jun-16 columns, the RTT between client and server must be less or equal to 0.03 second. The RTO and RTO2 are less than zero second, which provides an insignificant timing for a botnet detection. The 3WHS is expected to be less or equal to 0.045 second, which also provides an unimportant timing for a botnet detection. Refer to Figure 11 and Table 2, the TTL for botnet training and testing are equal to 128.

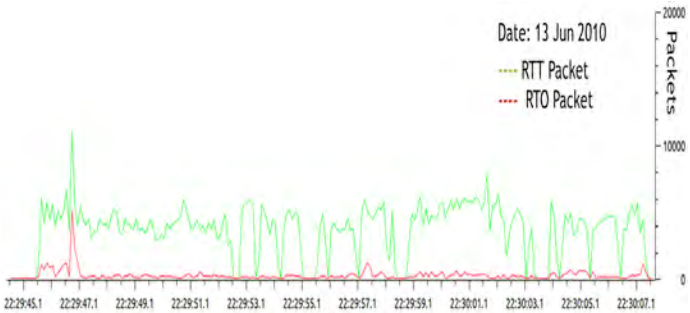


Fig. 8. A total of SMTP packets per second on 13 Jun 2010

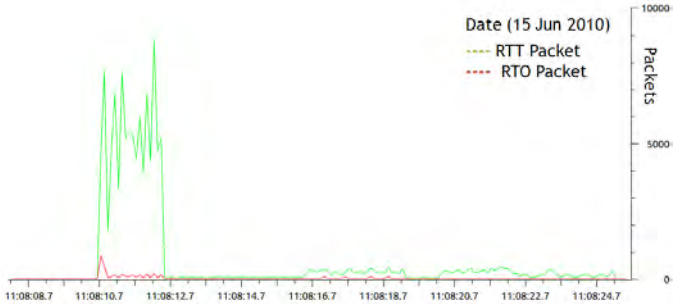


Fig. 9. A total of SMTP packets per second on 15 Jun 2010

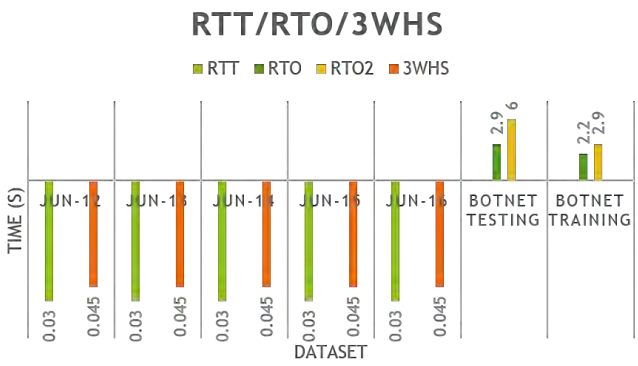


Fig. 10. A summary of max RTT and RTO for seven traffic datasheets

Table 1
A summary of max RTT and RTO for seven traffic datasheets

DATASET	RTT (s)	RTO (s)	RTO2 (s)	3WHS (s)
Jun-12	0.03	0	0	0.045
Jun-13	0.03	0	0	0.045
Jun-14	0.03	0	0	0.045
Jun-15	0.03	0	0	0.045
Jun-16	0.03	0	0	0.045
Botnet Testing	0	2.9	6	0
Botnet Training	0	2.2	2.9	0

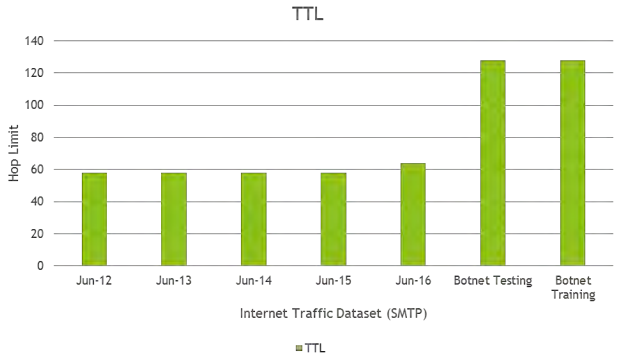


Fig. 11. A graph of average TTL for packet for seven traffic datasheets

Table 2
A table of average TTL for packet for seven traffic datasheets

DATASET	TTL
Jun-12	58
Jun-13	58
Jun-14	58
Jun-15	58
Jun-16	64
Botnet Testing	128
Botnet Training	128

VI. CONCLUSION

We have presented multi-domain SDNs with Spamhaus server. The proposed method allows SDN Controllers to update the Spamhaus server using up-to-date botnet controller lists (BCL). The BCL is updated and shared among SDNs by a Spamhaus server. This will help prevent botnet attacks on SMTP server as well as spreading into other SDN domains. We also discussed the method for analyzing SMTP traffics flow using a decision tree algorithm. The method utilized RTO packets between server and client to detect the SMTP traffics flow. We plan to implement the multi-domain SDNs with Spamhaus server as the future work. The upcoming experiment will deliver a better solution in securing the multi-domain SDNs from botnet attacks on SMTP server.

REFERENCES

[1] G. Vanecek, GeoPlex: Universal Service Platform for IP Network-based Services, 1997. http://www.cerias.purdue.edu/news_and_events/events/security_seminar/details/index/56218-noZCKZKV1F3c-372-hq15nTbsPk31bQ8W.

[2] N.V. Michah Lerner, George Vanecek, Middleware Networks: Concept, Design and Deployment of Internet Infrastructure, Kluwer Academic Publishers Norwell, 2000.

[3] S. Shenker, Gentle Introduction to Software-Defined Networking, 2012. https://www.youtube.com/watch?feature=player_detailpage&v=eXsCQdshMr4&t=168.

[4] Open Networking Foundation, Software-Defined Networking: The New Norm for Networks, 2012.

[5] O.N. Foundation, OpenFlow, (2016). https://www.opennetworking.org/sdn-resources/openflow/57-sdn-resources/onf-specifications/openflow?layout=blog (accessed January 29, 2016).

[6] S.H. Park, B. Lee, J. You, J. Shin, T. Kim, S. Yang, RAON: Recursive abstraction of OpenFlow networks, Proc. - 2014 3rd Eur. Work. Software-Defined Networks, EWSDN 2014. (2014) 115–116. doi:10.1109/EWSDN.2014.29.

[7] V.K. Gurbani, M. Scharf, T. V. Lakshman, V. Hilt, E. Marocco, Abstracting network state in Software Defined Networks (SDN) for rendezvous services, IEEE Int. Conf. Commun. (2012) 6627–6632. doi:10.1109/ICC.2012.6364858.

[8] Mouli, Why SDN Concepts Need To Extend Into The Wan, (2016). http://www.aryaka.com/blog/why-sdn-concepts-need-to-extend-into-the-wan/ (accessed January 31, 2016).

[9] SDxCentral, Inside SDN Architecture, (2016). https://www.sdxcentral.com/resources/sdn/inside-sdn-architecture/ (accessed January 31, 2016).

[10] S. Azodolmolky, SDN-based cloud computing networking, in: Transparent Opt. Networks, 2013: pp. 2–5. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6602678.

[11] R. Jain, S. Paul, Network virtualization and software defined networking for cloud computing: A survey, IEEE Commun. Mag. 51 (2013) 24–31. doi:10.1109/MCOM.2013.6658648.

[12] A. Dixit, F. Hao, S. Mukherjee, Towards an elastic distributed sdn controller, in: Proc. …, 2013: pp. 7–12. doi:10.1145/2491185.2491193.

[13] P. Berde, M. Gerola, J. Hart, Y. Higuchi, M. Kobayashi, T. Koide, et al., Onos, in: Proc. Third Work. Hot Top. Softw. Defin. Netw. - HotSDN '14, 2014: pp. 1–6. doi:10.1145/2620728.2620744.

[14] R. Beverly, K. Sollins, Exploiting Transport-Level Characteristics of

Spam, (2008). http://18.7.29.232/handle/1721.1/40287.
[15]S. Scott-Hayward, G. O'Callaghan, S. Sezer, Sdn Security: A Survey, 2013 IEEE SDN Futur. Networks Serv. (2013) 1–7. doi:10.1109/SDN4FNS.2013.6702553.
[16]R. Kl, P. Smith, OpenFlow: A Security Analysis, in: 21st IEEE Int. Conf. Netw. Protoc., 2013. doi:10.1109/ICNP.2013.6733671.
[17]R.L. Smeliansky, SDN for network security, Sci. Technol. Conf. (Modern Netw. Technol. (MoNeTeC), 2014 First Int. (2014) 1–5. doi:10.1109/MoNeTeC.2014.6995602.
[18]S. Lange, S. Gebert, T. Zinner, P. Tran-Gia, D. Hock, M. Jarschel, et al., Heuristic Approaches to the Controller Placement Problem in Large Scale SDN Networks, IEEE Trans. Netw. Serv. Manag. 12 (2015) 4–17. doi:10.1109/TNSM.2015.2402432.
[19]Z.Y. and F.B. Wang Shuling, Li Jihan, Research on SDN Architecture and Security, Telecommun. Sci. 29 (2013) 117–122.
[20]N. Hoque, M.H. Bhuyan, R.C. Baishya, D.K. Bhattacharyya, J.K. Kalita, Journal of Network and Computer Applications Network attacks : Taxonomy , tools and systems, 40 (2014) 307–324.
[21]J. Ioannidis, S.M. Bellovin, Implementing Pushback: Router-Based Defense Against DDoS Attacks, 2014. doi:10.1007/s13398-014-0173-7.2.
[22]S. Lim, J. Ha, H. Kim, Y. Kim, S. Yang, A SDN-oriented DDoS blocking scheme for botnet-based attacks, 2014 Sixth Int. Conf. Ubiquitous Futur. Networks. (2014) 63–68. doi:10.1109/ICUFN.2014.6876752.
[23]R.K. Sahu, N.S. Chaudhari, A performance analysis of network under SYN-flooding attack, IFIP Int. Conf. Wirel. Opt. Commun. Networks, WOCN. (2012) 2–4. doi:10.1109/WOCN.2012.6335561.
[24]M.-S.K.M.-S. Kim, H.-J.K.H.-J. Kong, S.-C.H.S.-C. Hong, S.-H.C.S.-H. Chung, J.W. Hong, A flow-based method for abnormal network traffic detection, 2004 IEEE/IFIP Netw. Oper. Manag. Symp. (IEEE Cat. No.04CH37507). 1 (2004) 1–14. doi:10.1109/NOMS.2004.1317747.
[25]P. Nevlud, M. Bures, L. Kapicak, J. Zdralek, Anomaly-based Network Intrusion Detection Methods Keywords Detection of Network Anomalies, (2013) 468–474. doi:10.15598/aeee.v11i6.877.
[26]B. Xiao, W. Chen, Y. He, E.H.M. Sha, An active detecting method against SYN flooding attack, Proc. Int. Conf. Parallel Distrib. Syst. - ICPADS. 1 (2005) 709–715. doi:10.1109/ICPADS.2005.67.
[27]H. Chen, C. Mao, S. Tseng, An Approach for Detecting a Flooding Attack Based on Entropy Measurement of Multiple E-Mail Protocols, 18 (2015) 79–88. doi:10.6180/jase.2015.18.1.10.
[28]C. Schafer, Detection of Compromised Email Accounts Used by a Spam Botnet with Country Counting and Theoretical Geographical Travelling Speed Extracted from Metadata, 2014 IEEE Int. Symp. Softw. Reliab. Eng. Work. (2014) 329–334. doi:10.1109/ISSREW.2014.32.
[29]H. Luo, B. Fang, X. Yun, Anomaly detection in SMTP traffic, Proc. - Third Int. Conf. onInformation Technol. New Gener. ITNG 2006. 2006 (2006) 408–413. doi:10.1109/ITNG.2006.34.
[30]G. Kakavelakis, J. Young, Auto-learning of SMTP TCP Transport-Layer Features for Spam and Abusive Message Detection., Lisa. (2011). http://static.usenix.org/events/lisa11/tech/slides/beverly.pdf.
[31]T. Sochor, Overview of e-mail SPAM Elimination and its Efficiency, Res. Challenges Inf. Sci. (RCIS), 2014 IEEE Eighth Int. Conf. (2014) 1 – 11.
[32]E. Biglar Beigi, H. Hadian Jazi, N. Stakhanova, A.A. Ghorbani, Towards effective feature selection in machine learning-based botnet detection approaches, 2014 IEEE Conf. Commun. Netw. Secur. (2014) 247–255. doi:10.1109/CNS.2014.6997492.
[33]S. Seeber, L. Stiemert, Towards an SDN-Enabled IDS Environment, in: Commun. Netw. Secur., 2015: pp. 751–752.
[34]Mininet Team, Mininet, (2016). http://mininet.org/ (accessed February 22, 2016).
[35]M. Gupta, J. Sommers, P. Barford, Fast, accurate simulation for SDN prototyping, Proc. Second ACM SIGCOMM Work. Hot Top. Softw. Defin. Netw. - HotSDN '13. (2013) 31. doi:10.1145/2491185.2491202.
[36]U.N. Brunswick, CTU-Malware-Capture-Botnet-1, (2015). http://mcfp.felk.cvut.cz/publicDatasets/CTU-Malware-Capture-Botnet-1/ (accessed May 20, 2012).

Mohd Zafran Abdul Aziz has received his first Bachelor Degree (B. Eng of Electrical and Computer Science) from Kumamoto University ,Japan on March 2001 and obtained his Master Degree (MSc of Engineering) from Tokyo University Of Technology ,Japan on March 2008. He also has 6 years in industrial as project engineer in several multinational company focus on industrial automation and instrument engineer. Currently on study leave as lecturer from Computer Department of Universiti Teknologi MARA, Shah Alam ,Selangor , Malaysia .

He is currently a PhD candidate and belong to Department of Advanced Information Technology, Graduate School of Information Science and Electrical Engineering, Kyushu University, Japan.

Koji Okamura is a Professor at Research Institute for Information Technology, Kyushu University and Director of Cybersecurity Centre Kyushu University, Japan. He received B.S and M.S. Degree in Computer Science and Communication Engineering and Ph.D. in Graduate School of Information Science and Electrical Engineering from Kyushu University, Japan in 1988,1990 and 1998,respectively.

He has been a researcher of MITSUBISHI Electronics Corporation Japan for several years and has been a Research Associate at the Graduate School of Information Science ,Nara Institute of Science and Technology ,Japan and Computer Centre, Kobe University , Japan. He's area of interest is Future Internet and Next Generation Internet, Multimedia Communication and Processing, Multicast/IPV6/QoS , Human Communication over Internet and Active Network. He is a member of WIDE, ITRC , GENKAI , HIJK project and Key person of Core University Program on Next Generation Internet between Korea and Japan sponsored by JSPS/KOSEF.

Proceedings of the APAN – Research Workshop 2016
ISBN 978-4-9905448-6-7

Design and Implementation of Monitoring Schemes for Software-Defined Routing over Federated Multi-domain SDN Testbed

Pang-Wei Tsai, Aris Cahyadi Risdianto, Teck Chaw Ling, JongWon Kim and Chu-Sing Yang

***Abstract* — Emerging Software-Defined Networking (SDN) paradigm has been widely affecting most networking fields. However, the real-world SDN application for inter-domain routing management is still limited since the routing exchange among wide-area networks is quite complicate due to the extreme scale of global Internet connectivity. Several SDN-leveraged routing ideas are being proposed to improve the routing exchange among wide-area networks. Thus, in this paper, an on-going experience for experimenting and validating the inter-domain routing proposals over OF@TEIN federated testbed in Asia is shared. By focusing on the design and implementation of monitoring deployment for visibility support, we try to identify practical key points and provide improved monitoring for validating the performance and anomaly of the exchange. Other design considerations are also discussed together with possible future research directions.**

***Index Terms* — SDN-leveraged routing, federated multi-domain testbed, monitoring, measurement, visibility.**

I. Introduction

As the Software-Defined Networking (SDN) [1] enables the possibility on programming network operation with softwarized methods, it becomes a popular topic in research and education fields. The most significant advantage of SDN is its flexible controllability on network provision, and it also offers a framework to manage the network status. In the legacy network architecture, the network interaction depends on protocol negotiation among network nodes, and operators may need to setup proper configurations on network devices to let them know how to recognize the forwarding information. However, in SDN, this work can be done by the controller well, and the protocol negotiation among SDN devices is no longer needed. The SDN controller is able to centrally instruct network devices how to treat incoming packets. Furthermore, optimization mechanisms such as traffic engineering, fault-tolerance, security and measurement can be easily implemented on SDN-enabled network due to its characteristic of centralized control.

Nowadays, SDN is gradually applied on network field [2]. Jarschel et al. [3] summarized several use cases, pointing out the key attributes of SDN, and reviewed some start-up cases toward large-scale SDN appliance on Wide Area Network (SD-WAN). For example, Google [4] used software-based control mechanism running on commodity servers to perform the simplified coordination for changes of network status. For another, Microsoft [5] designed a new architecture to carry more traffic and support flexible adaption with the software-driven mechanism. Moreover, in exchanging Internet traffic between Internet Service Providers, Gupta et al. [6] addressed the difficulties of making WAN traffic delivery, and they also proposed an experimental design with Software-Defined Exchange (SDX) method to process traffic exchange among routers participated in the internet exchange point. These novel researches are the instances for improving network operation with the SDN techniques over the wide-area environment.

Even the SDN is already applied on research, educational and commercial solutions, deploying SDN techniques into WAN environment is still a crucial task. Sezer et al. [7] presented a discussion about the difficulties when SDN is raised in different aspects. Due to the characteristic of WAN, any routing effects has to be careful to keep the network work with stability and consistently. Operation such as fluctuating and symmetric route should be avoided. The improper configurations may lead packet transmission corrupted. Nevertheless, there are several integration issues need to be considered. For example, the routing decisions made by SDN controller must be reliable. Moreover, establishing a valid supervisor for monitoring network behavior is necessary. Hence, how to acquire the measurement and monitoring is an

Pang-Wei Tsai and Chu-Sing Yang are with the Institute of Computer and Communication Engineering, Department of Electrical Engineering, National Cheng Kung University, Taiwan (e-mail: {pwtsai,csyang}@ee.ncku.edu.tw)

Aris Cahyadi Risdianto and JongWon Kim are with the School of Information and Communications, Gwangju Institute of Science and Technology (GIST), Korea. (e-mail: {aris,jongwon}@nm.gist.ac.kr)

Teck Chaw Ling is with the Department of Computer System & Technology, Faculty of Computer Science and Information Technology, University of Malaya, Malaysia. (e-mail: tchaw@um.edu.my)

important issue in routing exchange of SDN.

Due to the necessity on exploring and verifying SDN innovations, the network testbed plays an important role on research and education purposes. There are many colleagues and institutes build their testbeds for conducting experiments. For creating a testing environment approaching to wide-area network, the OF@TEIN testbed [8] provides such a service, and authors of this paper manage part of testbed sites collaborated with the OF@TEIN testbed. Because of the newly implemented SDN method, the testbed nodes are able to use public IPs to make data transmission with each other. The testbed network is built as integrated control architecture with OpenFlow [9] and BGP [10]. In this circumstance, network operation on delivering packets is similar to the WAN-based routing. The operation for inter-connecting testbed sites is claimed to be the Software-Defined Routing (SD-Routing) Exchange by the authors. Because of the importance of system operation and management, there is demand to monitor the experiment network of the testbed. This paper introduces the plan for making measurement and observation on the inter-connection among testbed sites. The paper aims to share the experience for planning and designing a method to monitor BGP routing status initially, identifying the possible issues and challenges for future development.

The remainder of the paper is organized as follows. Section II brings with a brief review of background and related work. Section III has made a discussion of purposes and concerns. Section IV interprets the operation environment, explaining the design of system. Section V shows the initial evaluation results of the implementation. The plan for future developments is presented in Section VI. Finally, the conclusions are given in Section VII.

II. Background and Related work

This section describes the background knowledge and related work. It explains the concerns for designing and developing issues of SD-Routing Exchange in the OF@TEIN testbed.

A. Wide-area Routing in SDN

For reaching a better network architecture with flexibility and adaption, the SDN is recognized as an innovative way [11] for network provision. The most significant different between SDN-enable network and legacy network is that the SDN uses centralized control method to manage the network. It separates the control plan and data plane by utilizing out-of-band control channel to make control communication between network operation system (i.e., controller) and device. While the essence of the control plane in SDN network is more fragile than the legacy network. The reason is that the SDN devices (e.g., switch and router) have to be instructed by the controller to process the packets. Therefore, the design issues for securing the control plane are an important thing in SDN research.

Since the BGP [10] is the most wide-spread protocol for managing wide-area routing operation, to stitch local SDN network with wide-area network, the SDN controller has to recognize and advertise routes from/to the legacy network. The provision is usually called Hybrid SDN [12]. For exchanging routes between SDN and legacy network, there are many researches focusing on SD-Routing issues. For example, the BGPMux [13] in GENI project had delivered a software router in VINI [14] testbed to carry real traffic into experiments. On the other hand, Lin et al. [15], Rothenberg et al. [16] as well as Thai and Oliveira [17] proposed their ideas and practices for integrating BGP and SDN. To determine the convergence time between SDN and legacy network, Gämperli et al. [18] illustrated a testing mechanism for further evaluations. For studying SDN traffic engineering and loading balance issues, Hong et al. [19] proposed a hybrid design to make approaches, and they also presented the evaluation result with discussion.

B. Trans-Eurasia Information Network and OF@TEIN

The Trans-Eurasia Information Network (i.e., TEIN) is a high speed network for research and education which connected among Europe and Asia areas [20]. There are more than fifty colleagues have joined the operation and established peering in TEIN. The TEIN network plays an important role of inter-continental traffic exchanges among Europe and Asia countries In 2012, the OpenFlow at TEIN (i.e., OF@TEIN [8]) collaboration community was established to carry out the SDN research and education purposes. It is expected to have further education activities in Cloud computing, SDN, Cyber-Security, WSN, Multi-media, IoT and other networking issues. There is also a federated testbed built for network experiments. The infrastructure of testbed is consisted of the SmartX [21] and domestic systems to provide not only computing but networking resources for creating experiments. Currently, the OF@TEIN testbed is constituted of more than a dozen sites and services on this testbed is free for academic use [22].

C. Software-Defined Routing Exchange on the OF@TEIN

Owing to the TEIN is a public network which carries the production traffic, therefore, in the beginning, the FlowVisor [23] and the VXLAN [24] are integrated with OpenFlow switch to deliver the inter-traffic among testbed sites. The VXLAN is able to create tunnels for establishing connections. As the result, this implementation solves the problems of traffic isolation and delivery well, while it also has some limitations. For example, the tunnels make intermediate routes being masked, it is hard to realize the status of underlay network. Especially for some routing experiments required to evaluate the network status closing to real network, this shortcoming is prohibited on their purpose.

To enhance the network control ability and use SDN-ways to establish inter-connections among each site in the testbed, a new development with SD-Routing Exchange method is under development. By designing a routing exchange utilizing ONOS [25] controller and software-based BGP router [26], the experiment network of OF@TEIN testbed is acting as a semi-controlled hybrid SDN, operating as an layer 3 overlay network for inter-connecting testbed sites. The routing exchange allows testbed operators advertise their domestic IP prefixes to the experiment network of the testbed. Due to the

TEIN network provides the Internet Exchange service for collaborators, namely, the inter-traffic among OF@TEIN testbed sites is count on it for delivery.

III. Discussion

The prototype development of SD-Routing Exchange in OF@TEIN makes the improvement on emulation for the testbed, while it also brings the demands on supervising and management for testbed operators. There are several concerns about planning and designing the monitoring mechanism for routing status on the testbed. This section makes a brief discussion for illustrating the key research issues.

A. Network Stitching

The PlanetLab [27] is the classic network testbed which runs across the WAN. It uses virtualization techniques to create multiple spaces for users, and the related spaces are linked [28] together as slices. Therefore, PlanetLab users are able to use assigned slices for conducting their experiments. The infrastructure of OF@TEIN testbed is partially similar to the PlanetLab, it also consisted of multiple sites from different countries. While the network architectures of PlanetLab and OF@TEIN testbed are different. First of all, for OF@TEIN testbed, the platform is able to allow testbed users to assign public IP addresses to their nodes in experiments. Second, to link inside (SDN) and outside (legacy) networks, the edge BGP router is directly connected to one or more legacy gateways. By doing this, the edge BGP router is able to forward node packets between production network and testbed network on each site.

B. Network Transparency

For a large-scale network testbed with distributed infrastructure, the testbed network is usually collaborated by following methods: light-path, native VLAN, VPN and layer 3 connection. Owing to the light-path and VLAN are expensive in establishment and maintenance, these two solutions are not considered. By contrast, the VPN is an essential way to build site-to-site connection, while it shields the hop information during packet transmission. Therefore, for providing experiment network with more reality, the OF@TEIN testbed using the layer 3 IP routing for packet transmission among testbed sites. However, due to the network environment, the testbed sites are separated by legacy network. The SDN controller is not able to manage the traffic in the intermediate area. For enhancing the scalability, we plan to use an alternative way to let testbed user configure their cross-site experiment network passively. Since several testbed sites are attached with multiple RENs and domestic ISPs, it might be possible to determinate site-to-site routes by assigning different IP domains on the nodes. By doing this, testbed users are able to select network path among their nodes indirectly. For achieving this, developing a measurement method to realize the possible routes in legacy network is necessary.

C. Measurement and Monitoring

In order to realize the system status, there is an existing health check mechanism [29] for monitoring hardware utilization and availability. It uses active ways (with probe packets) to measure data plane of the OF@TEIN testbed. Due to the method is concentrating on end-to-end test, there is a shortcoming to realize the network status between testbed sites. For this problem, there are already several novel researches use open and flexible ways [30] to design measurement methods in SDN measurement. Even so, since the testbed sites are distributed and semi-controlled with hybrid architecture, deploying a suitable and effective way for observing the traffic with visibility is a challenge. Hence, as stated above, there is a requirement to collect relevant information for presenting the inter-network of sites and sharing the results for management.

IV. System Design and Development

This section provides an overview of monitoring the experiment network in OF@TEIN testbed. The design and development issues are introduced in this section.

A. Network Environment

The forwarding policies for outgoing traffic of testbed nodes in the OF@TEIN can be divided as three parts: LAN traffic, Intra-traffic and Inter-traffic. When a node sends packets to another node in the same LAN, the packets are delivered as directly. On the other hand, when packets are sending to the node located in another local network (sub-net) in the same site, the gateway will forward packets according to its routing table. In this part, the gateway routers are stitching to a OpenFlow switch, and the underlay packet processing action is made by the OpenFlow controller. For inter-traffic, the edge routers is used to manage routing exchange among testbed sites. Their next-hop attributes is forced to set as default gateway addresses of legacy networks. By contrast, the edge router of production network also has to setup reverse routes with IP prefixes used by testbed nodes.

As mentioned above, the testbed network in OF@TEIN testbed acts as layer 3 overlay architecture through TEIN and other domestic RENs. On account of requiting extra devices, there are only three sites in OF@TEIN testbed has been equipped with this new architecture, and the local routes also advertise IP prefixes for routing exchange experimentally. For the time being, the development is implemented on GIST (Korea), UM (Malaysia) and NCKU (Taiwan) sites initially. Figure 1 is an example of routing architecture in one of sites. The deployed SDN controller is inherited from SDN-IP of ONOS [25]. For delivering incoming packets on the local SDN network, the data links of related devices are stitched to an OpenFlow switch, and then the controller setups corresponding flow rules for transmission.

B. Routing Exchange

For inter-traffic among testbed sites, a logical transit AS is developed to exchange route information. Routers in this AS are running with Route Reflector (RR) configuration to provide redundancy, which is shown as Figure 2. Due to the connectivity is relying on the Internet, therefore, the advertised routes are setup with next-hop attribute values as the default

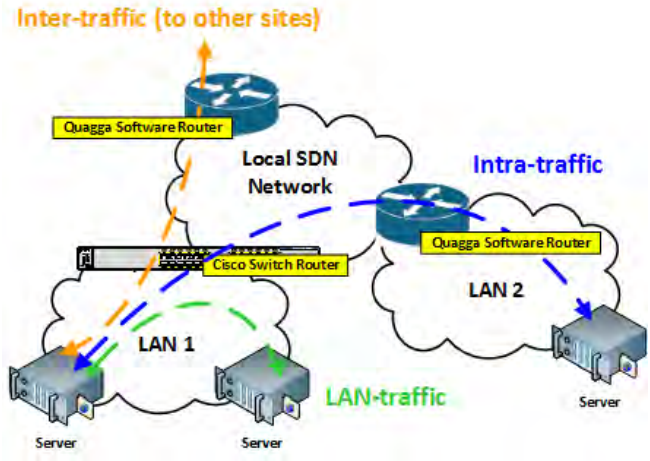


Fig. 1. The traffic forwarding at NCKU site.

gateway. The key point here is that there is no interaction between legacy network and SDN network in this hybrid architecture. The connection established by legacy network can be seen as links among routers in this logical transit AS. By doing this, the packets of testbed network are delivered by production network forcibly in the intermediate areas. In this operation scenario, if the BGP session for one router is abnormal, the routes are still able to be learned from another reflector. By measuring the status of RR routers, the monitoring of overall available routes can be easily made. However, there is no replication to guarantee routing exchange currently. Authors are still finding better solutions to enhance the reliability by adding more failover mechanisms.

C. *Development of Monitoring Schemes*

For monitoring and measurement this SD-Routing exchange, there are several monitoring indexes are required initially in control plane, such as BGP session, advertised IP prefixes, routing information base (RIB) as well as forwarding information base (FIB). Moreover, if the next-hop information can be recorded by the probe packet, it is possible to realize the transparent hops in the intermediate network. There is no way to control the path in the WAN by OF@TEIN testbed, while the routing path can be presented to testbed users, and let them realize more details about the status of experiment network.

Due to most networking devices are supporting standard SNMP-based query, the traffic statistics in data plan are able to be collected easily. For objectively observing the testbed network, the network management system (NMS) is setup at the third-party location to collect data (located at Amazon EC2 ap-northeast-1 region, Tokyo zone). All the routers and OpenFlow switches are required to enable SNMP daemon for pulling information by the NMS. There two common NMS tools are used to gather, analysis and present measured data: Cacti [31] and Smokeping [32]. Some customized scripts are also developed to provide measured data to the tools. We would like to implement different measuring ways to generate various logs as more as possible.

V. Initial Evaluation

To have a quickly view of work-in-progress tasks, this section makes a brief summary of prototyping developments in current stage. The presentation can be roughly divided as three parts: route advertisement, weathermap and alert notification.

A. *Route Advertisement*

To monitor BGP status, it is commonly and elastically to setup an observer router for receiving advertisement passively. Therefore, the IP prefixes announced from other routers are able to be received by this router, and the route information can also be recorded. However, considering the transit AS for passing advertisements is configured as RR way, it is not able to tracking details for diagnosis advertisement precisely on a single observer. Hence, the designed method is sampling each BGP routers separately. The measured data provides a change for further investigation in future development. By combination with the cacti plugin [33], the monitoring results network can be visualized clearly.

B. *Traffic Weathermap*

Since the original SmartX system already has a mechanism for monitoring the point-to-point throughput among server nodes, therefore, the proposed method in this paper is focusing on the experiment network of OF@TEIN sites. By making combination with the visualization tool [34], a web-based interface for illustrating the statistics of inter-traffic is being built, and the presentation is shown as Figure 3. On this weathermap website, a simple view of inter-traffic is briefly showed tentatively.

C. *Abnormality Notification*

To let site operators of realize of abnormal events, a mailing list is established to forward alert messages to OF@TEIN community members. Currently, the health check of network nodes and BGP session are two major indexes under supervision. If the heartbeat of a network node is unreachable, the detected mechanism will trigger the alert, sending email/SMS to site operators with warnings.

VI. Possible Research Directions in Future Development

On account of that there is only three sites have implemented SD-Routing, the routing topology is simple and clear currently. While there are still many sites in OF@TEIN testbed using the original way for inter-connection. In order to migrate routing functionality as well as develop monitoring schemes to extended sites, several planned and work-in progress tasks are described as follows.

A. *Exchange Centre*

When the number of joined sites grows up, the peering among testbed sites is getting much more complicated. How to manage a reliable routing exchange becomes a crucial task. For elastic and flexible configuration in joining new testbed sites, there is an idea to develop a control centre with automatic update ability is a possible solution. After registration, site

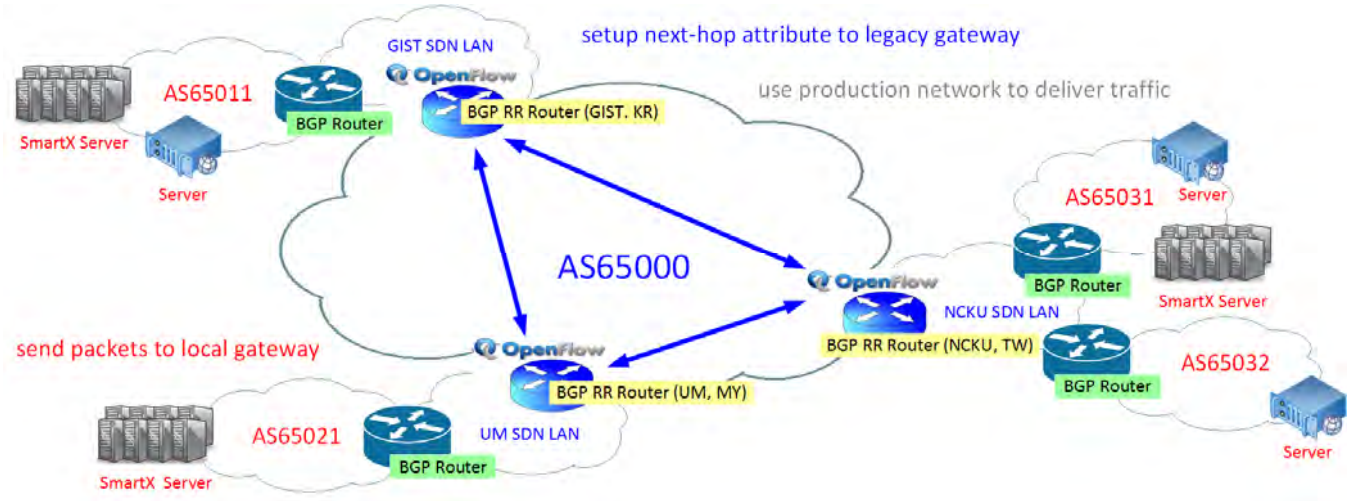


Fig. 2. The BGP routing deployment in OF@TEIN testbed.

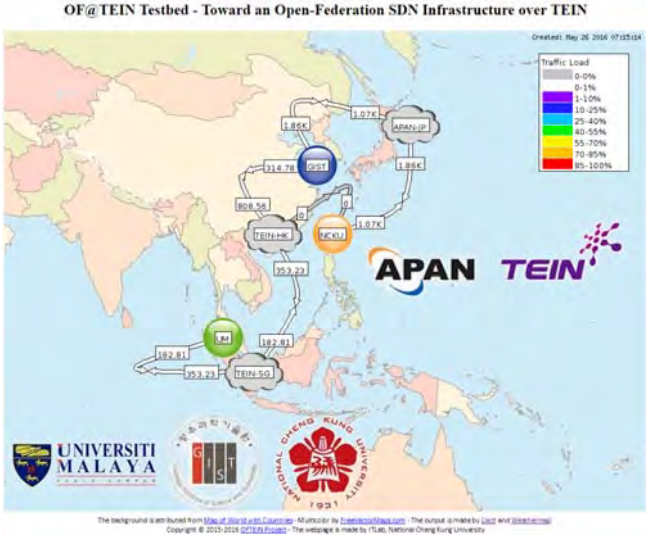


Fig. 3. The output screen of weathermap exported on website.

operators are able to submit their latest IP prefixes and configured policies to the exchange centre, and the exchange centre will push instructions to the routers for updating their configuration periodically. The testbed portal can also have the global view of routing advertisement on the centre.

B. *Security*

Due to the testbed network in intermediate area is in-band control, the routers are exploited with public access. For securing the system, a classic way is using access control rules to filter irrespective messages for the routers, Furthermore, only the authorized operators are allowed to access the routers, and the configuration of each router should be verified before activated. Integrating security mechanism into exchange centre is probably a choice, the unify authorization and authentication in updating routing configuration can be made with no difficulty.

C. *Supporting Multi-layer Monitoring*

In current stage, the measurement on the OF@TEIN testbed only watched few entries to satisfy the lowest requirement, such as live ping, traffic statistics, routing advertisement. For better viewing on system status, authors would like to extend the watched layers, collecting measurement data from not only the hardware but software framework. Building a shared database to store from all sites is also a thought. By doing this, the abnormal detection is able to be implemented on all the sites, and the global visibility is easier to make approaches.

D. *User API and Toolkit*

To provide testbed users more transparency in their allocated resource in experiments, sharing the measured results in database is a way, there is a plan to organize measured data into some divisions. By using Application Programming Interface (API) and defining query parameters, the authorized testbed users are able to get statistics information about nodes and networks in their experiments. For toolkits on the testbed, they are also able to access the shared database for gathering information. This development is expected to support more exploration for measurement and management issues for testbed users in the future.

VII. CONCLUSIONS

This paper introduces the implementation of SD-Routing Exchange in OF@TEIN testbed, and it also illustrates the management requirements and concerns which met by authors. The proposed method is trying to monitor the networking status of a hybrid SDN architecture with distributed sites. The design considerations and related issues are discussed in the paper, and the possible research directions are also introduced for future development. The aim of this paper is to introduce the experience on evaluation, trying identifying the issues for practicing the monitoring schemes over federated multi-domain testbed.

ACKNOWLEDGEMENT

This research is financially supported by the Ministry of Science and Technology of Taiwan (under grants No.104-3115-E-194-001 and 104-2218-E-001-002) for which authors are grateful. Authors are also grateful to the National Center for High-Performance Computing, TWAREN NOC, KOREN NOC and MYREN NOC for their support.

REFERENCES

[1] T. D. Nadeau and K. Gray, SDN: software defined networks. " O'Reilly Media, Inc.", 2013.
[2] A. Lara, et al., "Network innovation using openflow: A survey," Communications Surveys & Tutorials, IEEE, vol.16, no.1, pp.493-512, 2013.
[3] M. Jarschel, T. Zinner, T. Hoßfeld, P. Tran-Gia, and W. Kellerer, "Interfaces, attributes, and use cases: A compass for sdn," Communications Magazine, IEEE, vol. 52, no. 6, pp. 210–217, 2014.
[4] S. Jain, A. Kumar, S. Mandal, J. Ong, L. Poutievski, A. Singh, S. Venkata, J. Wanderer, J. Zhou, M. Zhu et al., "B4: Experience with a globally-deployed software defined wan," in ACM SIGCOMM Computer Communication Review, vol. 43, no. 4. ACM, 2013, pp. 3–14.
[5] C.-Y. Hong, S. Kandula, R. Mahajan, M. Zhang, V. Gill, M. Nanduri, and R. Wattenhofer, "Achieving high utilization with software-driven wan," in ACM SIGCOMM Computer Communication Review, vol. 43, no. 4. ACM, 2013, pp. 15–26.
[6] A. Gupta, L. Vanbever, M. Shahbaz, S. P. Donovan, B. Schlinker, N. Feamster, J. Rexford, S. Shenker, R. Clark, and E. Katz-Bassett, "Sdx: A software defined internet exchange," ACM SIGCOMM Computer Communication Review, vol. 44, no. 4, pp. 551–562, 2015.
[7] S. Sezer, S. Scott-Hayward, P.-K. Chouhan, B. Fraser, D. Lake, J. Finnegan, N. Viljoen, M. Miller, and N. Rao, "Are we ready for sdn? implementation challenges for software-defined networks," Communications Magazine, IEEE, vol. 51, no. 7, pp. 36–43, 2013.
[8] "Web portal of of@tein." [Online]. Available: http://oftein.net/
[9] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, "Openflow: enabling innovation in campus networks," ACM SIGCOMM Computer Communication Review, vol. 38, no. 2, pp. 69–74, 2008.
[10] Y. Rekhter and T. Li, "A border gateway protocol 4 (bgp-4)," 1995.
[11] D. Kreutz, F. M. Ramos, P. Esteves Verissimo, C. Esteve Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-defined networking: A comprehensive survey," Proceedings of the IEEE, vol. 103, no. 1, pp. 14–76, 2015.
[12] S. Vissicchio, L. Vanbever, and O. Bonaventure, "Opportunities and research challenges of hybrid software defined networks," ACM SIGCOMM Computer Communication Review, vol. 44, no. 2, pp. 70–75, 2014.
[13] V. Valancius and N. Feamster, "Multiplexing bgp sessions with bgpmux,"in Proceedings of the 2007 ACM CoNEXT conference. ACM, 2007, p. 44.
[14] A. Bavier, N. Feamster, M. Huang, L. Peterson, and J. Rexford, "In vini veritas: realistic and controlled network experimentation," in ACM SIGCOMM Computer Communication Review, vol. 36, no. 4. ACM, 2006, pp. 3–14.
[15] P. Lin, J. Bi, and H. Hu, "Internetworking with sdn using existing bgp,"in Proceedings of The Ninth International Conference on Future Internet Technologies. ACM, 2014, p. 21.
[16] C. E. Rothenberg, M. R. Nascimento, M. R. Salvador, C. N. A. Corrˆea, S. Cunha de Lucena, and R. Raszuk, "Revisiting routing control platforms with the eyes and muscles of software-defined networking," in Proceedings of the first workshop on Hot topics in software defined networks. ACM, 2012, pp. 13–18.
[17] P. W. Thai and J. C. De Oliveira, "Decoupling bgp policy from routing with programmable reactive policy control," in Proceedings of the 2012 ACM conference on CoNEXT student workshop. ACM, 2012, pp. 47–48.
[18] A. Gäamperli, V. Kotronis, and X. Dimitropoulos, "Evaluating the effect of centralization on routing convergence on a hybrid bgp-sdn emulation framework," in ACM SIGCOMM Computer Communication Review, vol. 44, no. 4. ACM, 2014, pp. 369–370.
[19] D. K. Hong, Y. Ma, S. Banerjee, and Z. M. Mao, "Incremental deployment of sdn in hybrid enterprise and isp networks."
[20] "The cooperation center of the trans-eurasia information network."[Online]. Available: http://www.teincc.org/
[21] J. Kim, B. Cha, J. Kim, N. L. Kim, G. Noh, Y. Jang, H. G. An, H. Park, J. Hong, D. Jang et al., "Of@ tein: An openflow-enabled sdn testbed over international smartx rack sites," Proceedings of the Asia-Pacific Advanced Network, vol. 36, pp. 17–22, 2013.
[22] T. Na and J. Kim, "Inter-connection automation for of@ tein multipoint international openflow islands," in Proceedings of The Ninth International Conference on Future Internet Technologies. ACM, 2014, p. 12.
[23] R. Sherwood, G. Gibb, K.-K. Yap, G. Appenzeller, M. Casado, N. McKeown, and G. Parulkar, "Flowvisor: A network virtualization layer," OpenFlow Switch Consortium, Tech. Rep, pp. 1–13, 2009.
[24] M. Mahalingam, D. Dutt, K. Duda, P. Agarwal, L. Kreeger, T. Sridhar, M. Bursell, and C. Wright, "Virtual extensible local area network (vxlan): A framework for overlaying virtualized layer 2 networks over layer 3 networks," Internet Req. Comments, 2014.
[25] P. Berde, M. Gerola, J. Hart, Y. Higuchi, M. Kobayashi, T. Koide, B. Lantz, B. O'Connor, P. Radoslavov, W. Snow et al., "Onos: towards an open, distributed sdn os," in Proceedings of the third workshop on Hot topics in software defined networking. ACM, 2014, pp. 1–6.
[26] P. Jakma and D. Lamparter, "Introduction to the quagga routing suite," Network, IEEE, vol. 28, no. 2, pp. 42–48, 2014.
[27] L. Peterson and T. Roscoe, "The design principles of planetlab," ACM SIGOPS operating systems review, vol. 40, no. 1, pp. 11–16, 2006.
[28] A. C. Bavier, M. Bowman, B. N. Chun, D. E. Culler, S. Karlin, S. Muir, L. L. Peterson, T. Roscoe, T. Spalink, and M. Wawrzoniak, "Operating systems support for planetary-scale network services." in NSDI, vol. 4, 2004, pp. 19–19.
[29] "Visibility-tools on of@tein." [Online]. Available: http://oftein.net/projects/visibility-tools/wiki
[30] A. Yassine, H. Rahimi, and S. Shirmohammadi, "Software defined network traffic measurement: Current trends and challenges," Instrumentation & Measurement Magazine, IEEE, vol. 18, no. 2, pp. 42–50, 2015.
[31] "Cacti: the complete rrdtool-based graphing solution," 2010. [Online]. Available: http://www.cacti.net/
[32] T. Oetiker and N. Tyni, "The smokeping website," 2005. [Online]. Available: http://oss.oetiker.ch/smokeping/index.en.html
[33] "Monitoring of bgp session via quagga daemon in linux." [Online]. Available: http://forums.cacti.net/viewtopic.php?f=12&t=51271
[34] "Network weathermap: Open source network visualisation." [Online]. Available: http://network-weathermap.com/

Pang-Wei Tsai received the Bachelor's degree in Electronic Engineering and the Master's degree in Computer and Communication Engineering from National Cheng Kung University. His research interest is on software-defined networking, cloud computing, virtualization and network management. He is also experienced in designing the large-scale network testbed. His current research is focus on developing the future Internet testbed on TaiWan Advanced Research and Education Network.

Aris Cahyadi Risdianto received the B.S. degree from Telkom University and M.S. degree from Institut Teknologi Bandung (ITB), Indonesia in Telecommunication Engineering. He is currently pursuing the Ph.D. degree in School of Electrical Engineering and Computer Science at Gwangju Institute of Science and Technology (GIST), South Korea. His research interests include Cloud-Computing, Software-Defined Networking and Future Internet Architecture.

Teck Chaw Ling is an Associate Professor at the Faculty of Computer Science and Information Technology, University of

Malaya and also the chairperson of Malaysia Research and Education Network-MYREN Network & Distributed Systems Working Group . His research areas include Software Defined Networking, Green computing, Core network research, inter-domain Quality of Service (QoS), Voice over IP (VoIP), cloud computing, and network security.

JongWon Kim received the B.S., M.S. and Ph.D. degrees from Seoul National University (Seoul, Korea), in 1987, 1989 and 1994, respectively, all in Control and Instrumentation Engineering. In 1994–2001, he was a faculty member of KongJu National University (KongJu, Korea) and University of Southern California (Los Angeles, USA). From September 2001, he has joined Gwangju Institute of Science & Technology (Gwangju, Korea), where he is now a full Professor. Since April 2008, he is leading GIST SCENT (Super Computing & Collaboration Environment Technology) Center as a director. He is also leading Networked Computing Systems Lab. (recently renamed from Networked Media Lab.) which focuses on Dynamic & Resource-aware Composition of Media-centric Services employing Programmable/Virtualized Computing/Networking Resources. His recent research interests cover topics such as software defined networking (SDN)/cloud computing for Future Internet testbed and smart media-centric services employing heterogeneous SmartX nodes.

Chu-Sing Yang is a Professor of Electrical Engineering in the Institute of Computer and Communication Engineering at National Cheng Kung University, Tainan, Taiwan. He received the B.Sc. degree in Engineering Science from National Cheng Kung University in 1976 and the M.Sc. and Ph.D. degrees in Electrical Engineering from National Cheng Kung University in 1984 and 1987, respectively. He joined the faculty of the Department of Electrical Engineering, National Sun Yat-sen University, Kaohsiung, Taiwan, as an Associate Professor in 1988. Since 1993, he has been a Professor in the Department of Computer Science and Engineering, National Sun Yat-sen University. He was the chair of the Department of Computer Science and Engineering, National Sun Yat-sen University from August 1995 to July 1999, and the director of the Computer Center, National Sun Yat-sen University from August 1998 to October 2002. He joined the faculty of the Department of Electrical Engineering, National Cheng Kung University, Tainan, Taiwan, as a Professor in 2006. He participated in the design and deployment of Taiwan Advanced Research and Education Network and served as the deputy director of National Center for High-performance Computing, Taiwan from January 2007 to December 2008. His research interests include future classroom/meeting room, intelligent computing, network virtualization.

Proceedings of the APAN – Research Workshop 2016
ISBN 978-4-9905448-6-7

Deploying and Evaluating OF@TEIN Access Center and Its Feasibility for Access Federation

Aris Cahyadi Risdianto[1], Phyo May Thet[2], Azeem Iqbal[3], Nurul Ainaa Binti Muhamad Shaari[4], Hari Krishna Atluri[5], Galih Nugraha Nurkahfi[6], Apichart Wantamanee[2], Rifqy Hakimi[6], Uzzam Javed[3], Muneeb Ahmad[3], Chaodit Aswakul[2], Muhammad U. Ilyas[3], Teck Chaw Ling[4], Arumugam Paventhan[5], Eueung Mulyana[6], and JongWon Kim[1]

1) Gwangju Institute of Science and Technology (GIST), Korea.
Email: jongwon@nm.gist.ac.kr
2) Chulalongkorn University, Thailand.
3) National University of Sciences and Technology (NUST), Pakistan.
4) University of Malaya, Malaysia.
5) ERNET (Education and Research Network) India, India.
6) Institut Teknologi Bandung (ITB), Indonesia.

***Abstract* – For the emerging software-defined infrastructure, to be orchestrated from so-called logically centralized DevOps Tower, the shared accessibility of distributed playground resources and the timely interaction among operators and developers are highly required. In this paper, by taking OF@TEIN SDN-Cloud playground as a target environment, we discuss an access center effort to address the above requirements. In providing the developer presence via the proposed access center, the inherent heterogeneity of internationally dispersed OF@TEIN resources is setting a unique challenge to cope with the broad spectrum of link bandwidths and round-trip delays. The access capability of deployed access center is experimentally verified against a wide range of access network conditions, which would be extended for futuristic access federation with appropriate identity management and resources abstraction for multiple developers and operators.**

***Index Terms* — DevOps-based automation, SDN-Cloud playground, software-defined infrastructure, multi-domain resource federation, and federated resource access.**

I. INTRODUCTION

SINCE launched in 2012, OF@TEIN has extended its capability from OpenFlow-enabled into SDN-Cloud-enabled playground[1]. As shown in Fig. 1 [1], the playground resources are deployed on the top of underlay network infrastructure that is distributed over multiple international sites and is involved with multiple network administrator domains. The playground resources are managed by logically-centralized DevOps (i.e., Development and Operations) Tower, which allows both developers and operators to perform various developments and resource management.

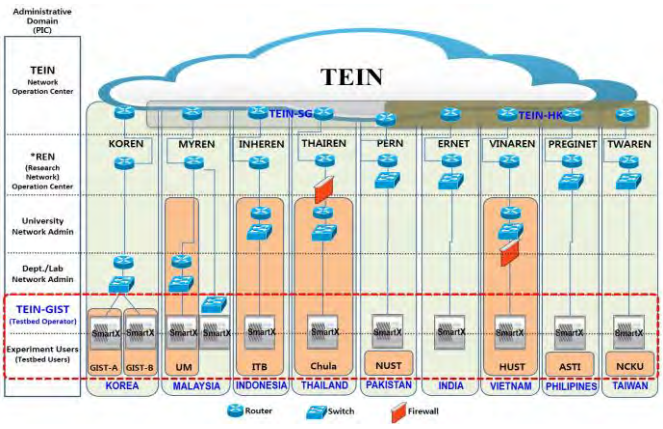


Figure 1: OF@TEIN Playground: A multi-domain SDN-Cloud testbed environment.

[1] Note that we intentionally use the term „playground" instead of „testbed" to highlight the software-driven flexibility of shared resource pools.

In order to allow developers to perform diverse experiments, we need to provide a systematic access to the shared resources of OF@TEIN playground directly from the different access networks of all developers. Since each access network can belong to different administrative domain, we need to apply a different set of rules to manage (i.e., allow and restrict) the playground access. In order to allow the playground access, the playground operators are required to identify the diverse reachability between OF@TEIN playground and developer access networks. However, the reachability can be still limited due to bandwidth availability, routing configuration, and firewall policy of underlying access networks. Considering these limitations (or difficulties), so-called access center deployment is needed to support multiple access methods by leveraging well-known protocol/port combinations, secure access schemes, and remote desktop access schemes. This access center may be able to help the developers to utilize the playground resources, overcoming several access limitations, without the burden of additional configuration.

II. Access Center: Requirements and Related Work

There are several access requirements and related efforts that motivated the deployment of OF@TEIN access center.

A. Access Center: Requirements

The developers may need full access to the playground for playground customization and experiment execution. The access should be open any time and any location regardless of developer sites with different time zones, institutions, and access networks. The common access hurdles are troublesome IPAM (IP address allocation and management including firewall blocking) issues and protocol-/time-of-day-based bandwidth limitations. The access may not be restricted in the sense of remote visualization. For example, the developer of video streaming experiments may want faster GUI-based access. Furthermore, remote access from different types of devices needs to be considered. Also, flexible configuration to support interactive and programmable access is required to help developers with different remote access needs.

The other access hurdles are unified (i.e., federated) authentication and heterogeneous resource abstraction with extended/expanded capability. The unified identity federation is required for centralized, policy-driven authentication to assign different levels of privileges. A simplified abstraction for heterogeneous resources is also required so that it can facilitate resource pooling by integration dispersed resources at specific locations (e.g., Chulalongkorn University Thailand, UNINET Thailand, and ERNET India).

B. Related Work

Akarsu and others discuss about gateways for seamless desktop access to high performance resources [2]. The gateway will retrieve data from different resources and allocate computational resources to process data. Thus, it hides system management and coordination from the developers. Treder et al. discuss about desktop applications for both remote and local accesses that support flexible access and remote desktop capability according to user requirements [3]. BonFire provides centralized broker service that interfaces users and federated resources of heterogeneous cloud and network testbeds. It offers user‟s access to resources through SSH gateway by VPN [4]. Also, Wahle et al. propose a reference-point gateway concept to federate independent testbed islands [5]. They provide web services to query and request the testbed resources.

From another perspective, Leandro et al. discuss about identity federation for access control in multi-tenancy cloud environment using Shibboleth, an authentication and authorization infrastructure based on SAML [6]. SAML standard assertion carries credentials across trusted domain boundaries, also known as tokens. Furthermore, Bhatt et al. try to maintain the confidentiality, integrity, and authentication over insecure internet connection [7]. As the communication was secured by SSLv3 and TLSv1 with CA server and Smart card client, the overall authentication time will be doubled.

III. OF@TEIN Access Center: Design

This section discusses about the design issues for access center. It includes access parameters to be considered, solution candidates, and the selection flexibility of users.

A. Key Design Issues

There are two aspects for OF@TEIN access center design. They are developer and access network requirements, respectively.

1) Developer Requirement

As discussed above, user‟s (developer‟s) requirements are important to be considered as the developers need to do experiments without constraints. The developer‟s access should be provided regardless the access locations (e.g., campus network, dormitory, or internet) as they may be wired and wireless connected. The access should be flexible with different access methods such as CLI-based, web-based GUI, and remote display for their specific visual experiment and verification.

2) Access Network Requirement

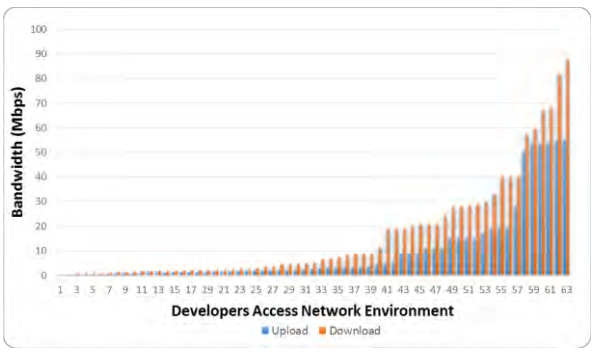


Figure 2: Developer access network: available bandwidth.

In order to provide the developer access from diverse network environments, several network parameters are considered between the developer access network and the access center, such as *i)* basic connectivity/reachability to verify the routing configuration, *ii)* access limitations or firewall policies to ensure access scheme provided, and *iii)* the available bandwidth to predict the access responsiveness. As depicted in Fig. 2, OF@TEIN access center is tested to provide multiple access schemes from various network environments (i.e., 21 networks with 3 different access times).

B. Design for Access Center

By considering many aspects of access center design and requirements, several access solutions or approaches have been selected such as *i)* multiple points of entry for access center, *ii)* multiple access schemes with different solutions, and *iii)* access center components for deployments. The detailed design of OF@TEIN access center is shown in Fig. 3.

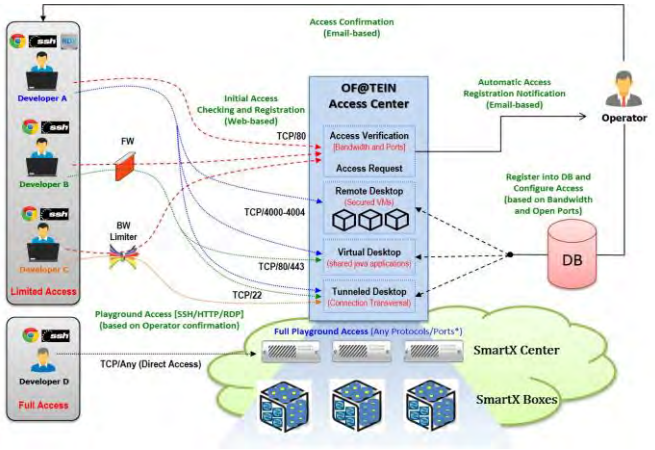


Figure 3: OF@TEIN Access Center: Design.

1) Multiple Points of Entry

Point of entry is a point where all the playground resources/services are accessible from the developers. Currently, the design considers entry from the R&E (research and education) network with unlimited access and from internet with some selected access schemes only. It is decided to provide developer access from their campus network as well as external network.

2) Multiple Access Methods

The script-/program-based and GUI-based accesses are required for the developers for their experiments. Several protocols or ports are however blocked/filtered due to their security performance policies. For this, the designed access center provides several access schemes such as using well-known application protocols and ports (e.g., HTTP web browsing), encrypted protocols communications (e.g., SSH remote access, HTTPs secure socket layer, or trusted Java web application), and/or remote pre-configured secured desktops (e.g., VRDP – virtual remote desktop protocol).

3) Selected Components

Several components are selected to provide the flexible access that satisfies all the considerations and requirements. The selected components for preliminary access center are:

- *Web-based portal* to verify the connectivity from developer‟s access network and for online access registration.
- *Bandwidth measurement web applications* to verify the available bandwidth from developer‟s access network.
- *Port scanner web applications* to check the access list applied in the developer‟s access network.
- *Connection transversal (tunneled desktop)* to provide access by translating into well-known protocol and ports.
- *Secure access for shared (virtual desktop) web application* to provide access through secure communication protocols (e.g., HTTPS).
- *Remote access to secured desktop or workstation (remote desktop)* to provide the accesses to secure dedicated virtual machine for each developers.

IV. OF@TEIN Access Center: Deployment and Verification

A. Deployment

The main task of the access center deployment is a physical box that is accessible from R&E networks (and Internet). This box is running on Ubuntu Linux OS with additional software and services (i.e., Open SSH server for secure remote access and Apache web server for web portal interface). For bandwidth measurement, we utilize *Ookla* Speedtest Mini [8] and *jnetscan* [9] port scan (as a java application). Three different access schemes are utilizing different software combinations: *i) Open SSH server* for tunneled desktop access through well-known ports, ii) Community version of *Ulteo Virtual Desktop* [10] for java-based shared virtual desktop web application, and iii) *Oracle Virtual Box* [11] for remote desktop to secure pre-configured virtual machines.

B. Measurement

In order to measure the quality of service (QoS) and quality of experience (QoE) of this preliminary access center deployment, several developers are tested and measured the access performance through several different access networks. In total, it is involved with 6 different countries (i.e., Korea, Thailand, Pakistan, Indonesia, Malaysia, and India), 10 developers, and 21 networks including R&E and public networks. The QoS measurement calculates the access setup time duration and number of packets or bytes required for the access. It simply utilizes *Wireshark* [12] network analyzer to capture all access related packets from developer‟s terminal to the access center. Based on the captured (*pcap* formatted) file, the traffic statistic (e.g., number of packets, number of bytes, duration between first packet and last packet, and average rate) for specific access scheme can be analyzed. The QoE measurements polled the developer‟s experiences during testing and measurements for different access schemes and quick feedbacks to match with their requirements.

C. Result and Analysis

As mentioned above, two different measurements are completed (i.e., QoS and QoE) and analyzed for further improvements in next deployment.

1) Quality of Service based on Pcap Packet Analysis

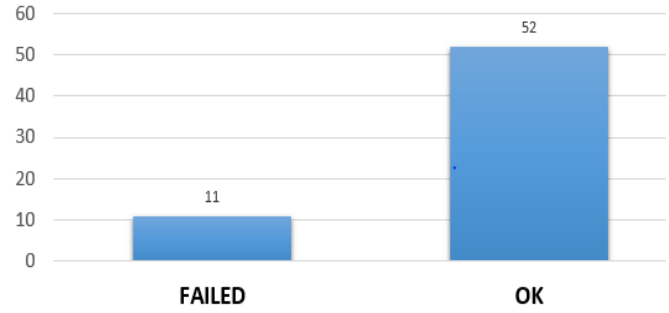


Figure 4: Access status result.

The overall testing result of all access schemes is acceptable, because approximately 83 % of accesses are successful, as depicted in Fig. 4. Less than ~17 % access problems are caused by several different aspects (e.g., developer‟s operating system issue, java/browser compatibility issue, and some unknown reasons), which may not be directly related with current access center deployment. However, it still needs to be considered for future work extension.

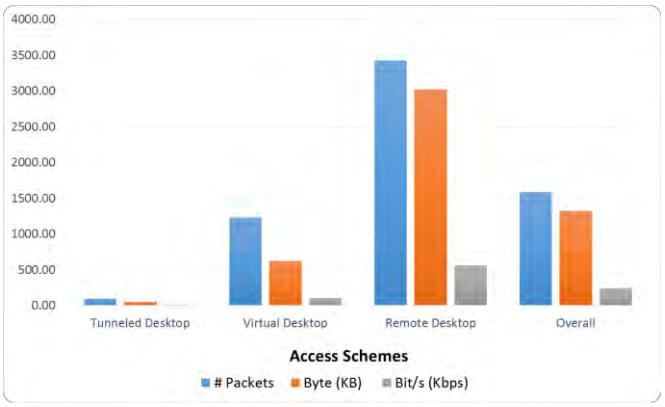


Figure 5: QoS comparison for all access schemes.

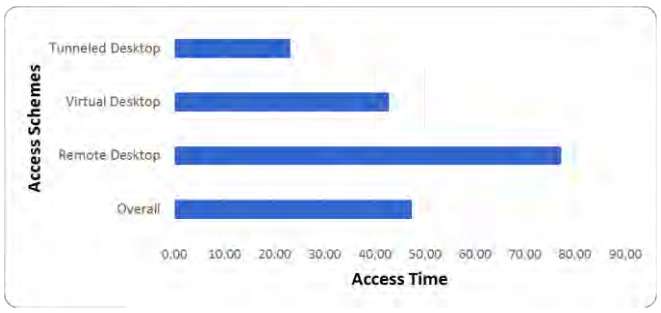


Figure 6: Access time comparison for all access schemes.

Fig. 5 and Fig. 6 show the comparison between the access schemes based on the measured access parameters. As expected, the tunneled desktop requires smallest number of packets and number of bytes, and also with fastest access setup time. It is then followed by virtual desktop and remote desktop accesses that require more packets and bytes and longer access setup time. However, tunneled desktops are only suitable for well-known applications and static TCP ports (e.g., web page, remote access, and others). Also it is required for specific configuration (e.g., TCP port translation, Linux scripting, and unique access links). Virtual and remote desktops are more useful for GUI-based experiments such as video or java-based applications.

2) Quality of Experience based on Developers Polling

As QoE verification from the developers, MOS-like (similar with "*mean opinion score*") value is one representative parameter for this purpose. In order to get the developer‟s opinion score about their access experiences, simple polling method is selected for 10 developers from different countries and access networks. The result is shown in Table 1, where the highest score for tunneled desktop followed by remote desktop and virtual desktop. However, note that overall the score is around 3.57, which are in between *"Fair"* and *"Good"* quality [13].

TABLE 1: DEVELOPERS ACCESS QUALITY OF EXPERIENCE (QOE)

NO	ACCESS SCHEMES	QOE (MOS)
1	Tunneled Desktop	4.43
2	Virtual Desktop	2.86
3	Remote Desktop	3.43
	Overall	**3.57**

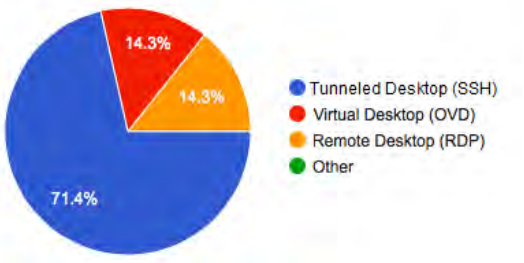


Figure 7: Preferred access schemes based on developer's experiences.

With this simple polling method, the developers also select the most preferred access *"anonymously"* in order to ensure independency. As depicted in Fig. 7, surprisingly that most all the developers are preferred tunneled desktop (based on connection transversal) access scheme than the other schemes. It is concluded that tunneled desktop is "lighter" and "faster" for accessing some specific resources in OF@TEIN playground. However, it needs to be verified continuously by observing and measuring *"real"* playground accesses during developer‟s experiments.

V. Feasibility Of Access Center For Access Federation

Since 2015, the focus of OF@TEIN collaboration is shifted into establishing an open collaboration consortium amongst existing and new potential collaborators, and developing a reference model to build and operate SDN-Cloud-leveraged open/shared infrastructure. Also, it hopes to establish a federation-based multi-domain SDN-Cloud playground and also a distributed support center to provide technical guide for all collaborators. Aligned with this transition, more playground resources, network domains, and developers are expected to increase and federate, while simple access and strong authentication/authorization are still needed to be considered. This section discusses more about the challenges of current work to be leveraged in the federation environment.

A. Access Federation Challenges

This work mainly focuses on "successful" access schemes to the playground for all developers and operators if required. Currently, it is not covering *authentication* and *authorization*, which defines the portions of playground resources accessible to different developers. Another important aspect for access federation is an abstraction for all federated resources into logical entities that are accessible from the access center. Additionally, the favorable access should give less resource requirements (e.g., terminal specification, bandwidth, software installation, etc.). In summary, access federation needs to solve following challenges:

- *Single identity management* to provide single-sign-on with group-based access control for multi-tenancy environment. It is possible to build trust relationships between institutions (as identity providers) and OF@TEIN playground (as resource provider).
- *Resources authorization* to provide OF@TEIN playground access policies for developers, operators to avoid resource monopoly or collision.
- *Playground abstractions* to present overlay entities of OF@TEIN playground over underlying multi-domain federated resources.
- *Access offload or network-style proxy* to minimize developer terminal requirements and speed-up the developer access to the playground resources.

B. Access Center for Access Federation

Based on the analysis on the requirements and current work results, the important items to extend in near future are:

a) *Access center (deployment enhancement)*: This is the main piece of access federation as it will federate (i.e., aggregate) all types of access (e.g., GUI or CLI) from the developers to the playground resources. But, improvements are required based on current measurement results and developer‟s experiences feedbacks. It also needs enhancements to provide an overlaid playground abstraction on top of underlay playground infrastructure.

b) *ID Federation:* This is an important component to provide single-sign-on authentication and resources authorization for different developers/operators. It should cover both SDN and Cloud resources. Several approaches are evaluated such as keystone federation and slice-based federation architecture (SFA).

c) *Access Box:* This additional deployment is required for access offload or network-style proxy to solve the current access limitations to the resources (e.g., low bandwidth, public IP address limitation, lack of graphical interface). It can be implemented as lightweight physical box (Pi or NUC) or virtual box (VM or container) with pre-configured operating systems and additional software.

VI. Conclusion

In this paper, OF@TEIN access center deployment is preliminary verified to provide multiple access schemes for OF@TEIN developers from multiple countries with different types of access networks. Based on QoS and QoE measurements, the preferred access scheme is tunneled desktop (based on connection transversal) access, preferred over virtual/remote desktop accesses. However, current deployment needs to be improved by considering other aspects such as single authentication and authorization, resource abstraction, and access offload, in order to provide the complete design of access federation for overlaid OF@TEIN playground over OF@TEIN underlay infrastructure.

Acknowledgement

This research was supported by 'Software Convergence Technology Development Program', through the Ministry of Science, ICT and Future Planning (S1004-14-1045). We also thank all OF@TEIN collaborators who are involved with open collaboration around OF@TEIN playground.

References

[1] Risdianto, Aris Cahyadi, et al. "OF@ TEIN: A Community Effort towards Open/Shared SDN-Cloud Virtual Playground." Proceedings of the Asia-Pacific Advanced Network 40 (2015), pp. 22-28.

[2] E. Akarsu et al., "Using Gateway system to provide a desktop access to high performance computational resources," High Performance Distributed Computing, 1999. Proceedings. The Eighth International Symposium on, Redondo Beach, CA, 1999, pp. 294-298. doi: 10.1109/HPDC.1999.805309

[3] Treder, Terry Noel, Christopher Fleck, Adam Marano, Anil Roychoudhry, and Richard James Mazzaferri. "Methods and systems for maintaining desktop environments providing integrated access to remote and local resources." U.S. Patent 9,239,666, issued January 19, 2016.

[4] David Garcia-Perez, Juan Angel Lorenzo Del Castillo, Yahya Al-Hazmi, Josep Martrat, Kon-stantinos Kavoussanakis, et al.. Cloud and Network facilities federation in BonFIRE. Federative and interoperable cloud infrastructures, August 2013, Aachen, Germany. Euro-Par 2013: Parallel Processing Workshops, 8374, Lecture Notes in Computer Science.

[5] Sebastian Wahle et al., "Technical Infrastructure for a Pan-European Federation of Testbeds", 5th International Conference on Testbeds and Research Infrastructures for the Development of Networks & Communities and Workshops, 2009. TridentCom 2009, pp. 1-8, doi: 10.1109/TRIDENTCOM.2009.4976205

[6] Leandro, M. A., Nascimento, T. J., dos Santos, D. R., Westphall, C. M., & Westphall, C. B. (2012). Multi-tenancy authorization system with federated identity for cloud-based environments using shibboleth. In Proceedings of the Eleventh International Conference on Networks, pp. 88-93.

[7] Bhatt, Deep Vardhan, Stefan Schulze, and Gerhard P. Hancke. "Secure Internet access to gateway using secure socket layer." IEEE Transactions on Instrumentation and Measurement, 55.3 (2006), pp. 793-800.

[8] "OOKLA Speedtest Mini." [Online]. Available: http://www.speedtest.net/mini.php

[9] "jnetscan." [Online]. Available: https://code.google.com/archive/p/jnetscan/

[10] "Open virtual Desktop Community Edition." [Online]. Available: http://www.ulteo.com/get-it/

[11] "Oracle Virtual Box". [Online]. Available: https://www.virtualbox.org

[12] "Wireshark." [Online]. Available: https://www.wireshark.org

[13] "Wikipedia: Mean opinion score." [Online]. Available: https://en.wikipedia.org/wiki/Mean_opinion_score

Aris Cahyadi Risdianto received the B.S. degree from Telkom University and M.S. degree from Institut Teknologi Bandung (ITB) in Telecommunication Engineering. He is currently pursuing the Ph.D. degree in School of Information and Communication at Gwangju Institute of Science and Technology (GIST), South Korea. His research interests include Cloud-Computing, Software-Defined Networking and Future Internet Architecture.

Phyo May Thet received B.Eng degree in Electronic Engineering from Technological University (Thanlyin), Myanmar, in 2011. From 2012 to 2014, she worked as a telecommunication engineer in Thailand and Myanmar. She is currently a Master student in the field of Wireless Network & Future Internet (STAR) Research Group at Department of Electrical Engineering, Chulalongkorn University, Thailand. From 2014 to present, she is a recipient of scholarship for International Graduate Students in ASEAN countries, Chulalongkorn University, Thailand. Her research interests include Cloud Computing, Future Internet Technology and Software Defined Networking.

Azeem Iqbal received the B.S. degree from University of Management and Technology (UMT), Pakistan in Electrical Engineering. He is currently pursuing the M.S. degree in School of Electrical Engineering and Computer Science at National University of Sciences & Technology (NUST), Pakistan. His research interests include Software Defined Networking, Data Analytics and Cloud Computing.

Nurul Ainaa Binti Muhamad Shaari received her B. Sc. degree from University of Technology MARA (UiTM) and MCS degree from University of Malaya (UM) in Data Communication and Computer Network. She is a lecturer at a private college and currently pursuing the Ph.D. degree in Faculty of Computer Science and Information Technology at University of Malaya (UM), Malaysia. Her research interests include Cloud-Computing, Software-Defined Networking and Future Internet Architecture.

Hari Krishna Atluri is working as a Project Engineer at ERNET, India's national education and research network (NREN). He has worked on R&D projects on the topics of "Mobile IPv6 testbed for mobility management over heterogeneous access networks", "Managing 6LoWPAN wireless sensor networks" and "Cloud computing based Educational Services". Prior to working at ERNET, Hari spent 2 years working for Supercomputer Education and Research Center (SERC), Indian Institute of Science. His current research interests are in areas of Internet protocols, software defined networking, cloud computing and future internet.

Galih Nugraha Nurkahfi received his B.S from physics and M.S degree in Telecommunication Engineering from Institut Teknologi Bandung (ITB), Indonesia. He is lecturer in School of Computing, Telkom University (Tel-U), and also IT consultant. His research interests are in Software Defined Networking, data science and big data.

Apichart Wantamanee received B.Eng in Telecommunication Engineering from King Mongkut's Institute of Technology Ladkrabang, Thailand, in 2010. Currently, he is a master student in the field of Wireless Network & Future Internet (STAR) Research Group at Department of Electrical Engineering, Chulalongkorn University, Thailand and works as senior engineer at Thaicom, satellite operator. His research interests include Cloud platform and Future Internet Technology.

Rifqy Hakimi received his B.S and M.S degree in Telecommunication Engineering from Institut Teknologi Bandung (ITB), Indonesia. He is a junior lecturer in School of Electrical Engineering & Informatics, Institut Teknologi Bandung (ITB) since 2015. His research interests are in Computer Networking field such as Internet Technologies and Software Defined Networking.

Uzzam Javed received his B.S. degree in Electrical Engineering from National University of Computer and Emerging Sciences (FAST-NUCES) Islamabad, Pakistan in 2013, and his M.S. degree in Electrical Engineering from School of Electrical Engineering and Computer Sciences (SEECS) of National University of Science and Technology (NUST), Islamabad, Pakistan in 2016. His research interest includes Software-Defined Networking, Data Science and Cloud Computing.

Muneeb Ahmad received his B.S. degree in Electrical Engineering from Air University (AU) Islamabad, Pakistan in 2013, and is currently pursuing his M.S. degree in Electrical Engineering from School of Electrical Engineering and Computer Sciences (SEECS) of National University of

Sciences & Technology (NUST), Islamabad, Pakistan. His research interests include Software-Defined Networking, Network Function Virtualization and Cloud Computing.

Chaodit Aswakul received the B.Eng. degree (1st class honor) in Electrical Engineering from Chulalongkorn University, Thailand, in 1994. In 2000, he received Ph.D. in Communications Networking from the Imperial College of Science, Technology and Medicine, University of London, U.K. He is currently an associate professor at Department of Electrical Engineering, Chulalongkorn University, Thailand, and head of the university‟s Wireless Network & Future Internet (STAR) Research Group. Chaodit was a recipient of the Ananda Mahidol Foundation Scholarship, Thailand.

Muhammad Usman Ilyas is currently appointed Assistant Professor of Electrical Engineering at the School of Electrical Engineering and Computer Science (SEECS) of the National University of Sciences and Technology (NUST), Islamabad, Pakistan. Prior to that he was a Post-doctoral Research Associate appointed jointly by the Electrical & Computer Engineering (ECE) department and the Computer Science & Engineering (CSE) department at Michigan State University (MSU). He worked under the joint supervision of Dr. Hayder Radha (IEEE Fellow) and Dr. Alex Liu at East Lansing, MI. Dr. Ilyas received his Ph.D. and MS degrees in Electrical Engineering from Michigan State University in 2009 and 2007, an MS in Computer Engineering from the Lahore University of Management Sciences (LUMS) at Lahore, Pakistan in 2004, and a BE (Honors) in Electrical Engineering from NUST, Pakistan in 1999. He is interested in network science, future Internet architectures and social networks.

Teck Chaw Ling is an Associate Professor at the Faculty of Computer Science and Information Technology, University of Malaya and also the chairperson of Malaysia Research and Education Network-MYREN Network & Distributed Systems Working Group. His research areas include Software Defined Networking, Green computing, Core network research, inter-domain Quality of Service (QoS), Voice over IP (VoIP), Cloud computing, and network security.

Arumugam Paventhan is Additional Director (R&D) with ERNET (India‟s national research education network) At ERNET, he has handled R&D projects, in the domains of Internet of Things, Mobile IPv6, EU funded MyFIRE Future Internet and Educational Cloud. Prior to joining ERNET, he held scientific and research positions at National Informatics Centre (HQ), New Delhi, C-DAC, Bangalore and Rutherford Appleton Laboratory (RAL), Oxford, UK. He has received M.Tech Computer Applications from IIT Delhi and PhD in the field of Gridcomputing from the University of Southampton, UK under the research scholarship award within the School of Engineering Sciences. He is a Senior Member of IEEE, and member of Internet Society (ISOC) and ACM. His current research interests include protocols and standards for Internet of Things (IoT), Cloudcomputing and Software Defined Networking.

Eueung Mulyana is an Assistant Professor in the School of EE and Informatics at Institut Teknologi Bandung (ITB), Indonesia. He received a Bachelor in EE from ITB, M.Sc. in Communication Channels Modeling from Universitatet zu Karlsruhe (now Karlsruhe Institute of Technology, KIT Germany) in 2001. He received his PhD (Dr.-Ing.) degree in Communication Networks from TUHH (Technische Universitaet Hamburg-Harburg), Germany in 2006. His current research focuses on Advanced Networking Technology and Services, including SDN, NFV and other emerging cloud-related infrastructure and applications.

JongWon Kim received the B.S., M.S. and Ph.D. degrees from Seoul National University (Seoul, Korea), in 1987, 1989 and 1994, respectively, all in Control and Instrumentation Engineering. In 1994–2001, he was a faculty member of KongJu National University (KongJu, Korea) and University of Southern California (Los Angeles, USA). From September 2001, he has joined Gwangju Institute of Science & Technology (Gwangju, Korea), where he is now a full Professor. Since April 2008, he is leading GIST SCENT (Super Computing & Collaboration Environment Technology) Center as a director. He is also leading Networked Computing Systems Lab. (recently renamed from Networked Media Lab.) which focuses on Dynamic & Resource-aware Composition of Media-centric Services employing Programmable/Virtualized Computing/Networking Resources. His recent research interests cover topics such as software defined networking (SDN)/cloud computing for Future Internet testbed and smart media-centric services employing heterogeneous SmartX nodes.

Proceedings of the APAN – Research Workshop 2016
ISBN 978-4-9905448-6-7

POTENTIAL APPLICATIONS OF SPACE TECHNOLOGY ON WATER RESOURCES MANAGEMENT IN THE NILE RIVER OF EAST AFRICA

Carlos M. Pascual

***Abstract*— A critical state-of-the-art review was conducted on the potential applications of space technology, particularly the altimetry radar technology for 20 years in progress in the Nile River systems in East Africa. Altimetry radar data obtained from ERS (ENVISAT) and Topex/Poseidon space missions were compared with gauge data from selected river systems. With satellite remote sensing and GIS supported validation tools, the water levels were estimated with low root mean square error and good accuracy far to use for operational hydrology but good enough for tracking variation trend. Such is a promising and very valuable tool for monitoring water levels of river network thereby contributing to some confidence for decision support systems for Nile River Basin initiative being implemented by various government and non-government institutions such as hydropower as affected by climate change. Case studies on the use of radar altimetry in Lake Tana, Ethiopia and Roseries Dam, Sudan will be presented. Such method gives some insights on the innovative applications of space technology radar altimetry products for regional water resources management, specifically on flood monitoring/hazard mitigation and water resources management, especially in remote and poorly gauged river basins. Moreover, AASTU being a university for industry will also present a transformation plan for the promotion of space sciences and technologies in the community for climate change resiliency towards sustainable development.**

***Index Terms*—Space technology, GIS, radar altimetry, remote sensing, Nile River Basin.**

I. Introduction

The application for satellite radar altimetry in hydrology is increasing. However, few studies have been devoted up to now for scrutiny on the quality of the altimetry data over the continental waters. Moreover, the availability of abundant water resources in develop and developing countries is very important and needs regular with judicious water level monitoring though space and time. Most countries are building its capacity to operate network for water resources development, mitigation of floods and other water-related disasters with national and international collaboration, like that of the Nile River Basin in Africa. The Nile River, is the longest river in the world having a total length of about 6700 km, traversing an extremely wide band of latitude, from 4o South to 32o North (Fig. 1). The area draining into the Nile River systems of about 3 million km^2 extends over ten African countries which are home and source of livelihood to approximately 180 million people. The two main river systems that feed the Nile are the White Nile, and the Blue Nile, with its sources in the Ethiopian highlands and with Tekeze-Setit-Atbara river systems contribute to the flow further downstream of Khartoum in Sudan. The annual runoff potential of about 85 billion cubic meters where average annual runoff estimates varied depending upon the length of records used for the estimation.

With much competing of such scarce water resources, it is important to develop methods to observe and measure spatial and temporal variations of water levels in the Nile River, among its river network, reservoirs and tributaries. The traditional stage-water gauging stations have becoming obsolete, time-consuming, and unreliable for continuous data gathering and recording. In the previous years, radar altimetry has proven to be an efficient alternative to monitor periodic measurements of large bodies of water globally. Varying retracking algorithms were developed by researchers which enabled to re-process, analyze and interpret radar altimetry echoes from Earth's surface, including surface of water bodies using data from National Aeronautics and Space Administration (NASA) and the European Space Agency (ESA) missions. This innovation remote sensing technology paved the way for searching the possibilities of retracking water level from radar echoes reflected from inland water bodies and to estimate river discharges. However, important

This work was supported in part by the LEGOS, France, Future University, Khartoum, Sudan and AASTU, Addis Ababa, Ethiopia, East Africa.
C.M. Pascual is with the Addis Ababa Science and Technology University Addis Ababa, Ethiopia (e-mail: cmpascual123@yahoo.com).

issues need to be addressed for radar altimetry data. These include the validation of measurements and identification of possible applications for solving practical problems related to water resources development and management, like the Nile River network in Sudan and other services areas of Africa. This paper will present a critical state-of-the-art review of recent initiatives and researches on radar altimetry studies in the Nile River Basin of Sudan and other nearby areas in Africa. Such review will somehow add to the geo-spatial dimension of validation studies and other potential applications of radar remote sensing, coupled with other emerging geographic information systems science and remote sensing technologies. Compared to other major river basins, the Nile Basin's disparity in water availability differs sharply among sub-basins which are complex and need continuous hydrometric measurements using remote sensing data, as an alternative to in-situ hydrometeorological data. As a case study, the potential applications of altimetry radar in water level measurements and volume estimation using altimetry radar in river systems covering northern-part of Sudan was explored. Altimetry data obtained from ERS (ENVISAT) and Topex/Poseidon (T/P) missions are compared with gauge data for selected river systems.

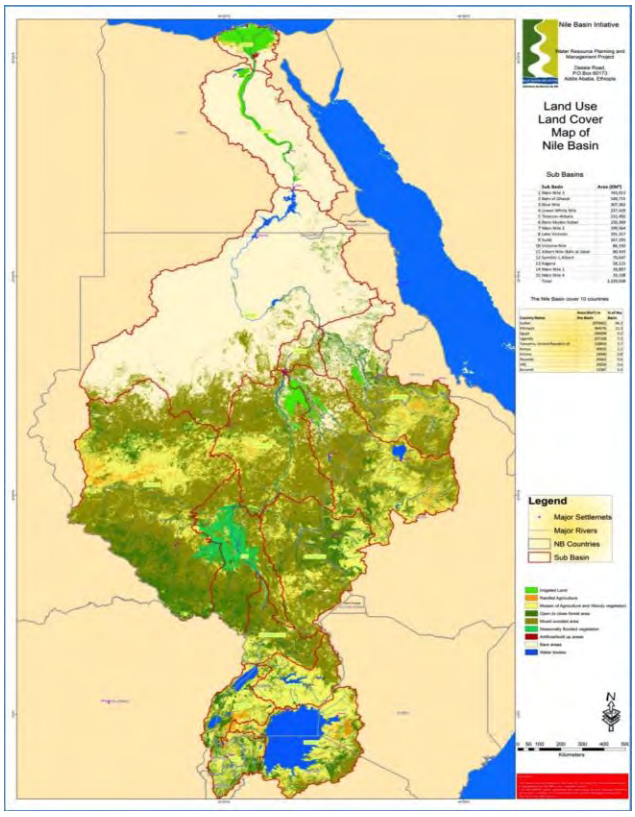
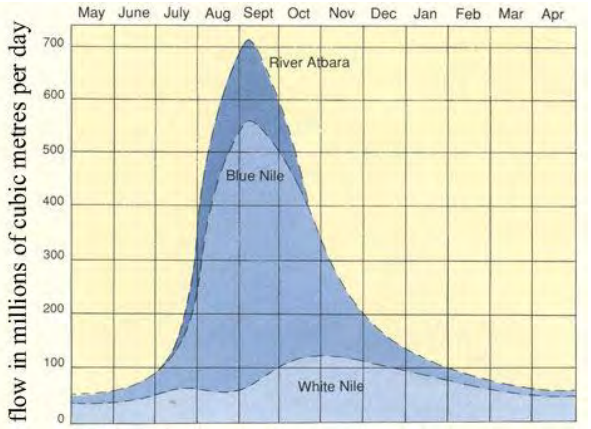


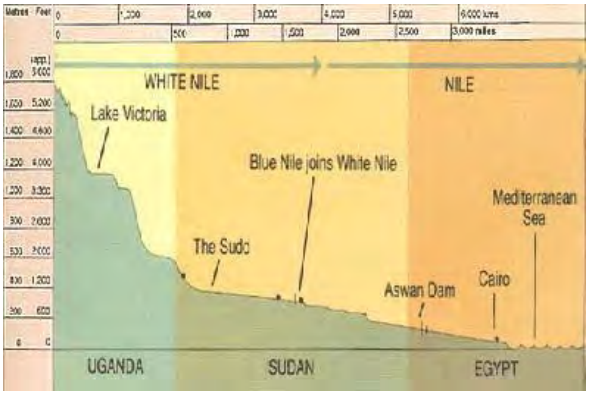
Fig 1. Land use/land cover, major river networks at the Nile River Basin covering ten countries in Africa.

II. THE NILE RIVER AND WATER RESOURCES CHALLENGES

The Nile River is the longest river in the world. From its major source, Lake Victoria in east central Africa, the White Nile flows generally north through Uganda and into Sudan where it meets the Blue Nile at Khartoum, which rises in the Ethiopian highlands. From the confluence of the White and Blue Nile, the river continues to flow northwards into Egypt and on to the Mediterranean Sea. From Lake Victoria to the Mediterranean Sea the length of the Nile is 5584 km (3470 mi). From its remotest headstream, the Ruvyironza River in Burundi, the river is 6671 km (4145 mi) long. The river basin has an area of more than 3,349,000 km^2. The Nile River's average discharge is about 300 million cubic metres per day (Fig. 2) [1]. From Khartoum the Nile flows northeast. 322 km below Khartoum it is joined by the Atbara River.



(a) Average monthly river flow



(b) height above sea level of the Nile

Fig. 2. (a) Monthly river flows of 3 tributaries in Nile; and (b) height of Nile at different points.

The black sediment brought down by the Atbara and Blue Nile Rivers used to settle in the Nile delta making it very fertile. This process historically occurred during the annual flooding of the Nile in the summer months. However, the opening of the Aswan High Dam in the early 1970s allowed for control of the flooding and reduced sediment deposits in the river as these now settle in Lake Nasser. During its course from the confluence of the Atbara through the Nubian Desert, the river makes two deep bends. From Khartoum to Aswan there are six

cataracts. The Nile is navigable to the second cataract, a distance of 1,545 km. The delta of the Nile is 190 km wide. The water level behind the Aswan Dam fell from 170 m in 1979 to 150 m in 1988, threatening Egypt's hydroelectric power generation. There are serious challenges in managing the Nile waters. Very little hydrological information and in particular time-series data has been shared by basin countries. Consequently, it is extremely difficult to accurately understand the behavior of the river system. When studies are conducted on the system there has not been a platform through which technical experts from each basin country can access these publications or share analyses. The Nile basin suffers from a high variability of rainfall and resulting floods and droughts. There is a current lack of sufficient storage infrastructure that could help to alleviate these impacts. Lack of common policy frameworks and even lack of transboundary water policies impact Nile countries ability to effectively cooperate on development programs. There has never existed a basin-wide analytic system which countries could access to openly and transparently share information and aid the understanding of broader impacts. Furthermore, some basin countries have limited technical capacity and financial resources to adequately address these technical challenges.

III. RADAR ALTIMETRY

Radar altimeters measure the distance between the satellite and the sea surface (E). The distance between the satellite and the reference ellipsoid (S) is derived by using the Doppler effect associated with signals emitted from marker points on the Earth's surface as the satellite orbits overhead. Variations in sea surface height (SS, ie S-E), are caused by the combined effect of the geoid (G) and ocean circulation (dynamic topography, DT) as shown in Fig. 3. The speed of electromagnetic waves multiplied by half of total time gives the range height from the satellite to the ocean surface.

An important quality of radar is its accuracy on its measurement ranges. However, previous investigation shows some errors in small inland water bodies due to variations of vegetations and other geographic features. Various algorithms were developed by NASA and ESA and already in used for continued research to improve the margin of errors in retracking methods

Various altimetry missions were employed since 1970s such as the joint ESA and Centre National Etudes Spatiales (CNES) of France which launched the Topex/Poseidon (temporal resolutions of 10 days). Other altimetry missions were the ERS-2, GFO, Jason¬1, and ENVISAT (temporal resolutions of 35 days).

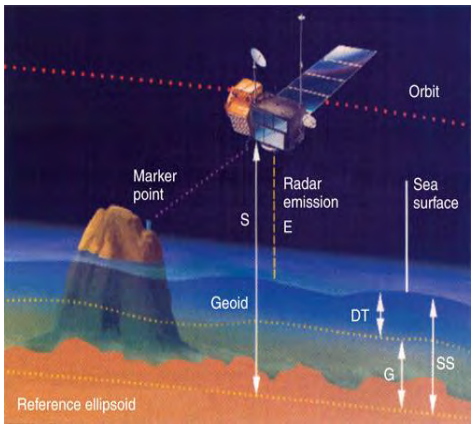


Fig. 3. Working principles of radar altimetry water level measurement.

IV. CASE STUDY APPLICATIONS

This section presents radar altimetry initiatives on access of databases and case studies in radar altimetry applications.

4.1. LEGOS Hydrological Database Initiative

Over the past years, the Laboratoire d'Etudes en Géophysique et Océanographie Spatiales (LEGOS) has developed a web database (HYDROWEB:http://www.LEGOS.obs-mip.fr) containing time series over water levels of large rivers, lakes and wetlands on a global scale [2]. Due to technical problems on the satellite, GEOSAT data are not suitable to estimate water height over continental water, and for ERS-1 and 2, the GDRs (Geophysical Data Records) provided by ESA are limited to ocean surface. Moreover the performances of Jason-1 and GFO satellites over narrow inland water are also poor, which has limited the use of these satellites to only big lakes where it has provided accurate lake height variations.

The lake water levels are based on the mergedTopex/Poseidon, Jason-1 and 2, ENVISAT and GFO data. Almost 150 lakes and reservoirs monthly level variations deduced from multi-satellite altimetry measurements are freely provided. Potentially the number of lakes monitored could be significantly increased. Figs. 4 and 5 show location of water level of rivers and lakes by radar altimetry in Africa (http://www.legos.obs¬mip.fr/fr/soa/hydrologie).

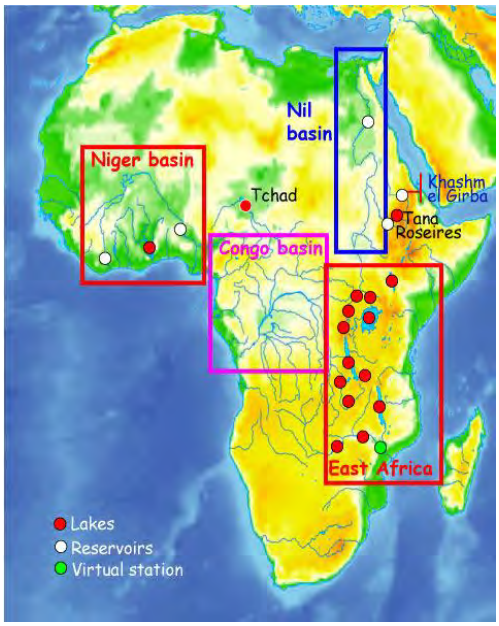
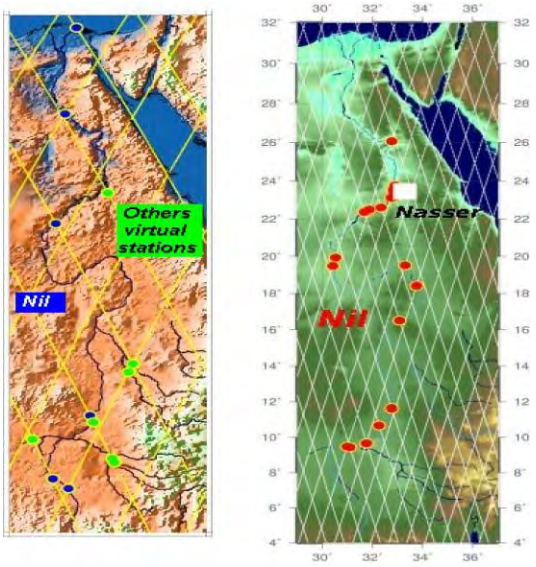


Fig. 4. Location stations of water level of rivers and lakes in Africa by satellite radar altimetry.



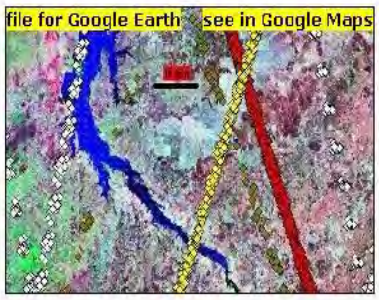
(a) Topex/ Poseidon (b) Envisat

Fig. 5. Sample virtual stations under satellite ground tracks in Nile River Basin.

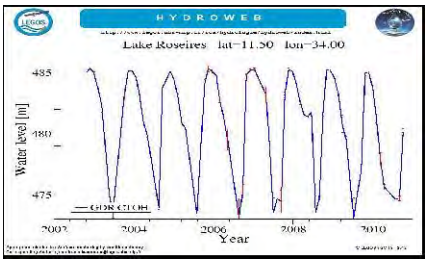
The Topex/Poseidon and Jason-1 have a 10-day orbital cycle and 350 km equatorial inter-track spacing. GFO has a 17-day orbital cycle and 170 km equatorial intertrack spacing. The combined global altimetry data set has more than decade-long history and is intended to be continuously updated in the coming decade. Combining altimetry data from several in-orbit altimetry missions increases the spatio-temporal resolution of the sensed hydrological variables.

4.2. Radar Altimetry Applications

A number of satellite radar altimetry applications on water levels measurements were conducted in major rivers and lakes worldwide, such as in Nile River Basin, covering large areas in Sudan, Africa as shown in Figure 6 in Lake Roseries, a part of Nile River basin in Sudan and major lakes/rivers in Africa [3]; [4] . The lake levels are based on merged Topex/Poseidon, Jason, ENVISAT and GFO data provided by ESA, NASA and CNES data centers.



(a) Satellite radar image



(b) Water level variations

Fig. 6. Sample database of satellite radar altimetry (a) and water level variation (b) of Lake Roseries in Sudan from HYDROWEB.

Potential radar instrument biases between different satellites are removed using T/P data as reference. Then lake levels from the different satellites are merged on a monthly basis (recall that the orbital cycles vary from 10 days for T/P and Jason, to 17 days for GFO and 35 days for ERS and Envisat). We generally observe an increased precision of lake levels when multi-satellite processing is applied (http://www.legos.obs-mip.fr/fr/soa/hydrologie/hydroweb/General_Info.en.htm

l. Another major and important case studies conducted recently was the Nile River Basin, an initiative to for trans-boundary water management covering ten countries in Africa. Where major lake and river are currently covered by T/P and ENVISAT missions, the water level are easily available on the HYDROWEB of LEGOS (Fig. 7). Such hydrological data from radar satellites provide accurate water level heights measurements not affected by some political and logistical considerations, which can empower water and policy decision makers to

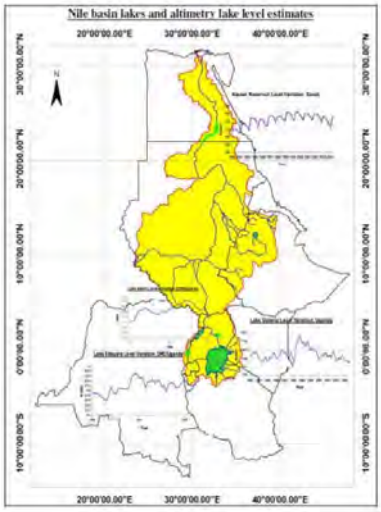


Fig. 7. The Nile River network and water sources covered by radar altimetry.

Validation study made by [3] on altimetry data obtained from ENVISAT and T/P missions in Lake Tana revealed that there is strong agreement among radar altimetry measurements with gauged water level data with RMSE of 0.019 and 0.091, respectively (Fig. 8). A hybrid of water level data sets improved the temporal resolution to an average of seven days but with RMSE of 0.11. With remote sensing validation technique, the water level in the lake was estimated with an RMSE of 0.72 and accuracy of ±1.41 m.

Available data gathered at radar altimetry data from 2002-2012 at reservoir of Lake Roseries, Sudan taken from LEGOS (2012) were compared to the gauged water levels in meters taken from local ministries of irrigation and water resources in Khartoum, Sudan (Fig. 9). Correlation analysis shows a very respectable positive relationship or closeness (correlation, r=0.96) among data taken from different sources across various time the data were taken. However, a closer look on some data revealed some degree of variations. Such might be due to some limitations of radar satellites and human errors in manual measurements. Thus, a more standard quantitative spatial analysis maybe employed for estimating positional accuracy of points on imageries, such as root mean square [3].

4.3 Web Databases

The Hydroweb Web site is hosted by CNES/Legos: www.legos.obs-mip.fr/soa/hydrologie/hydroweb/. This web database is the core of HYDROLARE data centre for Remote Sensing data on lakes and reservoirs with additional information on surface variation for a number of lakes for which precise sets of satellite imagery (from MODIS, Landsat, ASAR, CBERS) have been processed. Hydroweb provides lake level variations in time calculated from satellite radar altimetry (Fig.10). In addition the use of satellite images with adequate resolution (depending on the order magnitude of water extent variability) at different times, ranging from low to high water stage, it is possible to calculate a rating curve function which is simply the relation dh/dS, where dh is the variation of level, and dS is the variation of surface. Applying this rating curve function to all level data obtained from satellite radar altimetry will allow the calculation of the surface variation of the lakes over the time span of the altimetry data (ranging from 1-2 days for big lakes to one month for smaller ones) ([5].

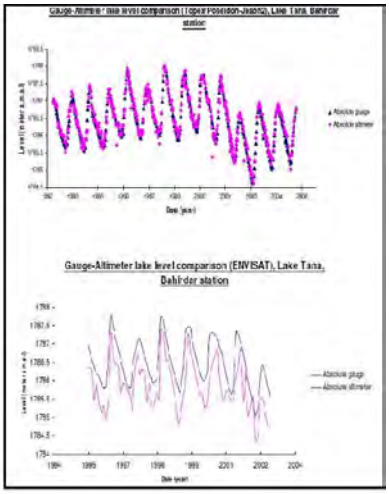


Fig. 8. Comparison of T/P and Envisat radar altimetry on gauge water level data in Lake Tana (Kaba, 2007).

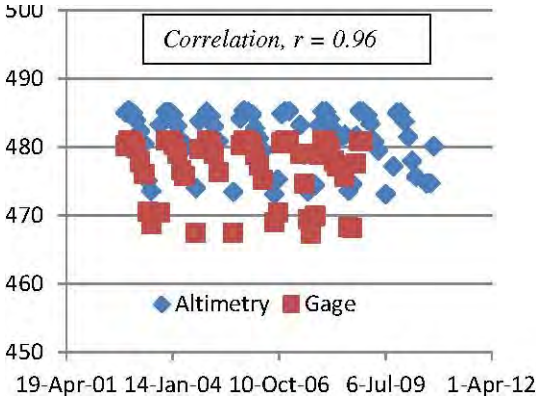
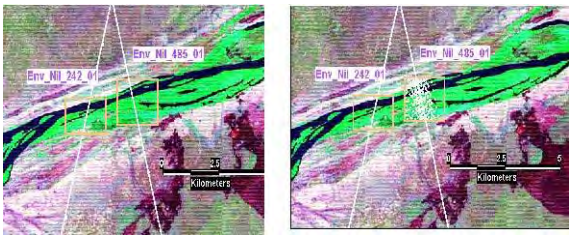
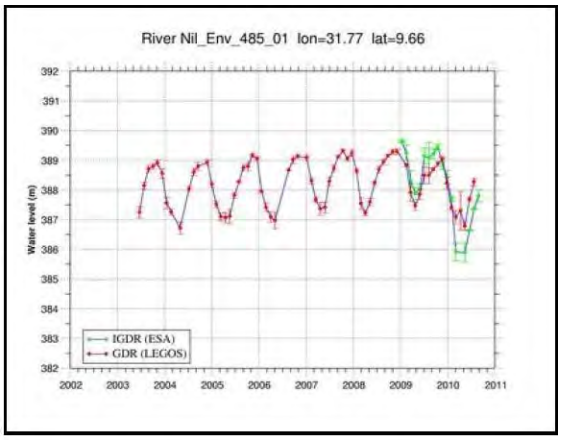


Fig. 9. Variations of T/P radar altimetry and gauged water level (m) reservoir data at Roseries, Sudan.



a) Landsat image showing of virtual stations on right side and satellite tracks on left side over Nile River



(b) Time series of water level of ESA and LEGOS data

Fig. 10. Landsat imageries of satellite tracks of virtual stations (a) and variations of water level of Nile River.

Towards this end and way forward, the AASTU-SCECT proposed to offer Geoinformatics degree courses in 2015 to fast track the promotion and application of space-based technologies among students and professionals in Ethiopia. Such initiatives will explore to collaborate among local, intercontinental and advanced universities and research institutions offering space technology courses and research activities.

V. CONCLUSION

Satellite radar altimetry has proven to be a valuable source of data for a broad range of applications. Looking beyond the missions in operational service today, future satellites will need to provide better spatial and temporal coverage so that we can study meso-scale variations and other phenomena more closely. The existing methods for lakes water level observations are well-developed and sufficient to obtain data accurate enough for further processing and analysis. The results of satellite water level measurements still contain substantial errors exceeding admissible limits. However these data enable one to assess general seasonal and long-term water level trends. Further validation of these data through the lakes and reservoirs ground network and improvement of the technique of satellite water level measurements are necessary. Moreover, to preserve the Nile River and to create a sustainable environment for its people in the approach of climate change, water conservation and management are important.

REFERENCES

[1] Owiro, A.O. (2004). The Nile Treaty: State succession and international treaty commitments. A Case Study of The Nile Water Treaties.

[2] LEGOS, (2006). "*Hydrology from space*", http://www.legos.obsmip.fr/soa/hydrologie/hydroweb/. Retrieved November 2006.

[3] Kaba. (2007). Validation of radar altimetry lake level data and its application in water resource development. MSc Thesis, ITC, The Netherlands. Pp 86

[4] Crétaux, J-F. , W. Jelinski , S. Calmant , A. Kouraev, V. Vuglinski , M. Bergé Nguyen , M-C. Gennero , F. Nino, R. Abarca Del Rio , A. Cazenave , P. Maisongrande, (2011). SOLS: A Lake database to monitor in Near Real Time water level and storage variations from remote sensing data, J. Adv. Space Res.

[5] Vuglinskiy, V. (2009). Water level in lakes and reservoirs, water storage. Global Terrestrial Observing System. FAO, Rome.

Carlos M. Pascual (BSAE'80–MSc'85–PhD'95). Dr. C.M.. Pascual is presently a professor in Hydraulic and GIS Engineering at the School of Civil Engineering and Construction Technology in Addis Ababa Science and Technology University based in Addis Ababa, Ethiopia. He is also a former dean of the School of Geoinformatics, Director of Kush Institute of Space Technology and Asst. President for Research and Development at the Future University in Khartoum, Sudan. He is a Filipino national and worked for 30 years as professor and Dean of Graduate School in Mariano Marcos State University. He initiated the application of space technologies such as remote sensing, GIS, GNSS/GPS and web-based in natural resource assessment, water resource development and renewable energy, as well as in the academe. He is an author for about 17 research papers published and presented about 100 papers in various conferences in local and international venues.

APAN Proceedings of the APAN – Research Workshop 2016
ISBN 978-4-9905448-6-7

Development of High-Precision 3D Measurement On Agriculture Using Multiple UAVs

Muhammad Haris, Seita Sukisaki, Ryo Shimomura, Zhang Heming, Li Hongyang, and Hajime Nobuhara

***Abstract*—Imaging system for high-precision 3D map on agriculture using UAVs was developed. The system were based on safe and easy UAVs with a ground station application which designed to be the interface between a human operator and the UAVs to carry out mission planning, flight command activation, and real-time flight monitoring. Based on the navigation data, and the way-points generated by the ground station, the UAVs could be automatically navigated to the desired waypoints and hover around each waypoint to collect field image data. By taking only low-resolution image, the proposed system is able to reduce the payload and increase the flight time of the UAVs. The input images then transform into higher-resolution image using reference images, taken by field server or ground-based device, via super-resolution techniques which is able to reduce blurring, blocking, and ringing artifacts especially in edge areas. Finally, we construct high-precision 3D map which proven having error of a millimeter order of magnitude. Our experiment result show that the input low-resolution can be transform into high-resolution image and effective to construct high-precision 3D map. The result indicate that the proposed system provides a reliable method of sensing agricultural field with high-precision 3D map.**

***Index Terms*—UAV, Aerial image, Sparse representation, Monitoring, Agriculture, Super-resolution, Phenotyping, 3D Images**

I. INTRODUCTION

The use of unmanned aerial vehicles (UAVs) in agriculture has increased in recent years due to problems of manual breeding methods in agriculture, which are laborious, time-consuming, unreliable, and often impossible to implement [1]–[5]. For example, high-frequency time series data are almost impossible to obtain without the use of a UAV. Moreover, large-scale, hilly landscapes make it impractical to manually analyze each tree individually using hand-held or ground-based devices. The use of UAVs can overcome such limitations, and UAV imaging offers advantages in terms of high-resolution data and precise 3D imaging.

Examples of some of the advantages offered by the use of UAVs over traditional field-based monitoring methods are listed in Table I. UAV imaging can efficiently provide high-frequency time series data, whereas aircraft and satellite systems are very complicated and their use requires arrangements be made in advance. Hand-held and ground-based devices have short preparation times but require long execution times. In terms of coverage, aircraft and satellites perform well because they can rapidly image several hectares in area, but they produce low-resolution images. In contrast, UAVs can provide better resolution as they have adjustable flight altitudes. Although hand-held and ground-based devices can provide the best resolution because they can observe parts of plants in detail, they cannot be used for large area and coverage or to produce high-frequency time series data. UAVs also require lower expenditures than aircraft or satellite as UAV sensors are much cheaper. As a UAV can be operated autonomously, control by the end user is much simpler. These advantages make UAV utilization in agricultural monitoring quite useful by offering a new perspective from which to monitor the ground with high precision [6].

TABLE I
COMPARISON OF AGRICULTURAL MONITORING SYSTEMS (◯ = SUPERIOR, △ = AVERAGE, × = POOR).

Method	Hand-held	Ground-based	UAV	Aircraft	Satellite
Frequency	×	△	◯	△	×
Coverage	×	×	△	◯	◯
Cost	◯	△	◯	×	×
User friendly	◯	△	◯	×	×
Resolution	◯	◯	△	△	×

The main problems in constructing 3D high-resolution maps using UAV images are flight-time limitations and image quality from the target object. Taking aerial images of a large field will consume a large amount of time, and to reduce time consumption, it is necessary to set an optimum height for UAV flight. However, maximizing the height, which increase the viewing perspective of the UAV and thus potentially reduces the flight time, reduces the optical detail of a target object. Therefore, it is necessary to use a super-resolution (SR) technique to obtain higher-resolution, high-precision images of target objects.

Higher-resolution image also means higher payload for the UAV, because it needs bigger bandwidth to transfer the image data to the local workstation. By using only low-resolution image from the UAV, we can reduce the payload of the UAV which also means increasing the flight time of the UAV. The input low-resolution is captured by the UAV, then downloaded to operator's smartphone and sent to server. This mechanism also enable the user to control many fields remotely. Soon after, it transform into higher-resolution using SR techniques which proposed in our paper [7]. Finally, we construct 3D image using SfM algorithm [8].

Field Server (FS) systems [9]–[12] can be used for ground-

All authors are with the Computational Intelligence and Multimedia Laboratory, Department of Intelligent Interaction Technologies, University of Tsukuba, Tennoudai 1-1-1, Tsukuba 305-8573, Japan e-mail: {mharis, sukisaki, shimomura, zhang, lhy, nobuhara}@cmu.iit.tsukuba.ac.jp

Manuscript received May 1, 2016; revised June 1, 2016; accepted June 30, 2016.

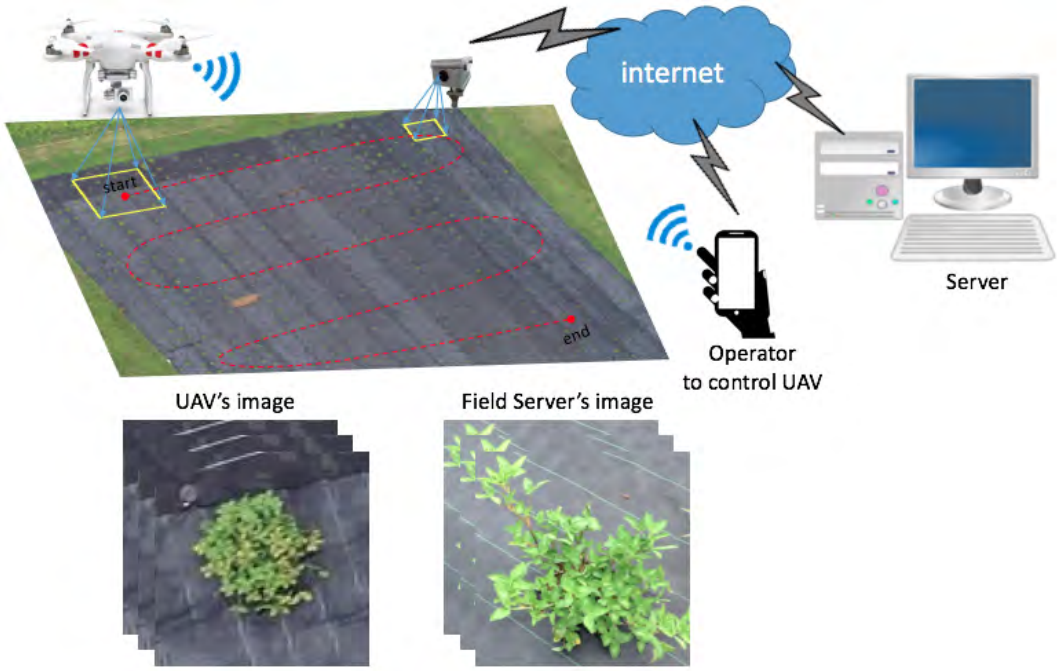


Fig. 1. Proposed system to support High-Precision Agriculture Using UAV and Field Server

based monitoring via a series of small sensor nodes equipped with a Web server that can be accessed via the Internet and communicate, unlike traditional sensor nodes, through a wireless LAN over a high-speed transmission network. An FS system can be easily installed for remotely monitoring field information anywhere. By including the functionality of a Web server in each module, an FS system can collectively manage each module over the Internet, producing high-resolution images that can be used as training images for an SR algorithm.

In this paper, we develop a framework to construct high-precision 3D map for agriculture using UAVs and FS (or ground-based device). We propose the use of low-resolution image taken by UAVs to reduce the payload of the UAVs, and transform it into higher-resolution using proposed SR technique. Finally, we demonstrate the effectiveness of the proposed system in reconstructing 3D map. The proposed system is shown in Fig. 1.

The paper is organized as follows. Section 2 presents an explanation of the proposed system. Section 3 discusses the results of our experiments and analysis. Section 4 shows the effectiveness of the proposed system to construct 3D map. Finally, in section 5 we present our conclusions.

II. PROPOSED SYSTEM

In this section, the proposed system is explained. The proposed system have 4 main components which discussed in the following subsection. The flowchart of the proposed system is illustrated in the Fig. 2.

A. Initialize Environment

Before collecting the data, the initialization of the environment is necessary, including setting the spatial resolution, formation path, and creating waypoints that considered altitude, latitude, longitude, the distance of every turning point, flight speed, and etc. Spatial resolution was the first variable to be considered because it largely determines how much detail can be interpreted from the final image and how many images need to be obtained. Spatial resolution can be defined as how much area is represented by a pixel on the image sensor. The spatial resolution used must be large or small enough to meet the objectives of the application requirements. There are two major factors that influence spatial resolution. One is flying height and the other is the focal length of the sensor. The relationship between these three variables can be expressed as following equation [13].

$$Res = \frac{S_{pixel} \times H_f}{f} \tag{1}$$

Where Res is the spatial resolution; S_{pixel} is the pixel size of the image sensor; H_f is the flying height and f is the lens focal length. The ratio of f/H_f can be defined as image scale, which is the distance between two points on an image to the actual distance between the same two points on the ground. It can be seen from Eq. 1 that the higher the UAV flies, the less ground resolution there will be, using the same image sensor. Similarly, the shorter the focal length, the higher the resolution is. Normally, the resolution and focal length have been selected before a flight. Therefore, Eq. 1 can be used to determine the required flying height to produce the desired resolution. For example, if the focal length is 4.5 mm, the pixel size is 3.12 µm and the expected resolution is 50 mm, then the flying height will be 72 m.

Once the resolution, flying height and the interested area to be captured have been established, the desired lines of flight and the position of the waypoints can be determined. Flight lines are normally orientated in a north-south or east-

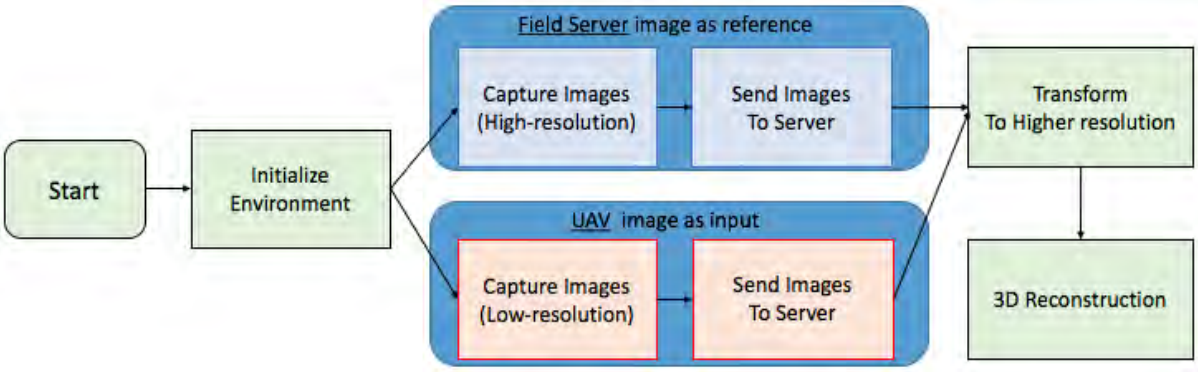


Fig. 2. Proposed System Flowchart

west direction and are usually parallel to each other. To collect images for the desired area on the ground, the UAV flies along the entire length of one strip, then move to the next flight line without changing heading and flies backwards along the entire length of the next adjoining flight line. This procedure is repeated until the desired ground area has been completely covered.

The initialization step also need to determine the formation path, and decide whether it use one, two, or more UAVs to take the image. The formation flight is able to optimize the 3D image result because it can take the image in many different angles. The demo of formation flight is illustrated in Fig. 3.

Fig. 3. Flight formation testing using 2 UAVs.

B. Image Acquisition

There are two types of images acquired in the system: from UAV (low-resolution image) and from field server or ground-based device (high-resolution image). For UAV image, there are many components that need to be set such as image overlapping, camera direction, camera angle.

To map a large field with the UAV system, requires a series of images to be taken along each of the multiple flight lines. To guarantee coverage without gaps throughout the field of interest, the images must contain enough overlaps. Aerial image overlap is the amount by which one image includes the area covered by another image, and is expressed as a percentage. Overlap normally contains two types of overlap along two directions: forward overlap and lateral overlap. The forward overlap is the common image area on consecutive images along a flight strip. The lateral overlap encompasses the overlapping areas of images between adjacent flight lines. Fig. 4 illustrates the forward overlap and lateral overlap between two flight lines. In order to cover as much as possible of the area with the minimum number of images taken by the UAV system, it is necessary to investigate appropriate overlap values.

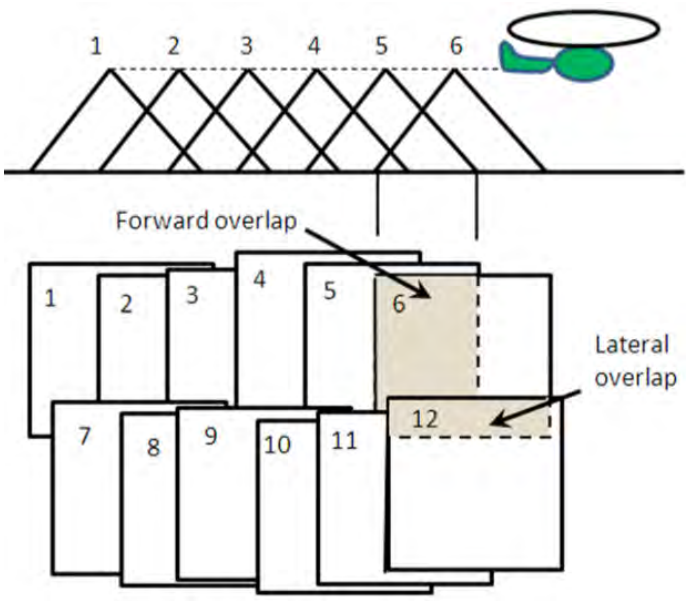


Fig. 4. Aerial image overlapping [14]

To support the formation flight, we also develop automatic shutter for multiple camera which will be installed in each UAV. The illustration is shown in Fig. 5.

C. Super-resolution Technique

The SR algorithm constructs high-resolution images from low-resolution ones. Several methods have been proposed to improve the quality and reduce the computational complexity of SR [15]. The SR algorithm is divided into single and multiple SR. The single SR algorithm constructs an improved image using the parametric image model and prior knowledge/dictionary, whereas the multiple SR algorithm relies on the number of pictures taken as input.

Fig. 5. Camera and gimbal which installed in the UAV

There are many challenges in improving the SR algorithm. In the proposed system, we use a super-resolution algorithm based on adaptive sparse representation via multiple dictionaries for images taken by Unmanned Aerial Vehicles (UAVs) [7]. The super-resolution attainable through the proposed algorithm can increase the precision of 3D reconstruction from UAV images, enabling the production of high-resolution images for constructing high-frequency time series and for high-precision digital mapping in agriculture. The basic idea is to use a field server or ground-based camera to take training images and then construct multiple pairs of dictionaries based on selective sparse representations to reduce instability during the sparse coding process. The dictionaries are classified on the basis of the edge orientation into five clusters: 0, 45, 90, 135, and non-direction. The SR algorithm is expected to reduce blurring, blocking, and ringing artifacts especially in edge areas.

D. 3D Reconstruction

Creating a 3D map with a UAV has become recently popular [16]. A well-known algorithm for this purpose is Structure-from-Motion (SfM) by Furukawa et al. [8]. This algorithm is advantageous to reconstruct and estimate 3D structure from 2D images without any depth information from another sensor such as LiDAR. Using SfM, we only need to rely on the camera and ground control points to construct a 3D image.

The future is wide open for the development of more sophisticated techniques that can recognize known objects or patterns. These advances will help enhance the entire process and make it faster. Currently, we used the GPU, which can modify the process to be even more efficient by replicating image processing tasks and performing many processes in parallel.

III. EXPERIMENTAL RESULTS

To confirm the efficiency of the proposed system, we conducted several experiments. The analysis of these experiments is divided into two subsections: quantitative and qualitative analyses. All experiments were conducted using Matlab R2012b on Win 8.1 64-bit (Intel Core i7@3.2GHz, 8GB). The images used in the experiment were taken at Kazusa DNA Research Institute, Chiba, Japan, red clover tree phenotyping field.

The image dataset consisted of two sub-datasets: training and testing. The training dataset was obtained using a hand-held camera, as shown in Fig. 6. The testing dataset was taken using DJI Phantom 2, as shown in Fig. 7, with an original size of 1280×720 pixels. However, to simplify the process, we divided the image into 320×180 pixel sub-images. In total, we used 300 YCbCr color testing images, as shown in Fig. 8. To conduct the experiment, we enhanced the luminance components (Y) while enhancing the other components using bicubic interpolation. Each resulting image channel was combined to produce a final color image.

Fig. 6. Samples of training images taken by hand-held digital camera.

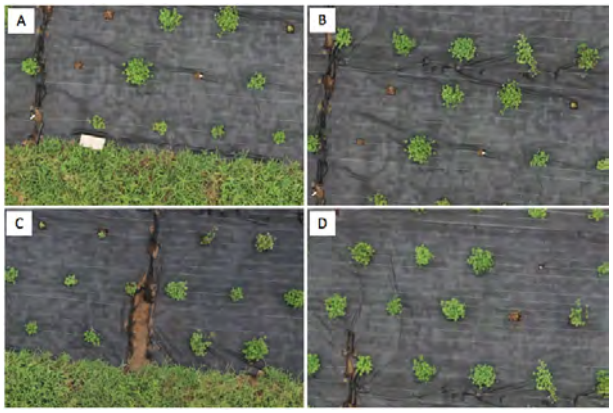


Fig. 7. Images A-D show sample testing images taken by UAV (DJI Phantom 2).

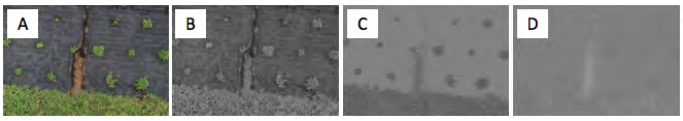


Fig. 8. YCbCr color components. A) Original color image, B) Y component, C) Cb component, D) Cr component.

In the experiments, we obtained images by downsampling and blurring the original images and then enlarging using different methods to $3\times$ magnification. We compared the effectiveness of seven methods: nearest neighbor, bilinear, bicubic, Yang et al. [17], Kim et al. [18], Zeyde et al. [19], and the proposed method. The algorithms associated with

these vary in nature; therefore, in order to produce objective comparisons, all parameters used in the training and testing had to be similar. However, no specific parameter needed to be used for the conventional interpolation methods.

Our proposed method uses 3×3 patches with no overlapping pixels and five pairs of dictionaries. The algorithm of Yang et al. [17] uses 5×5 patches with a 4-pixels patch overlap and a single pair of dictionaries with 1024 atoms with back-projection. The algorithm of Zeyde et al. [19] uses 3×3 patches with 2-pixels patch overlap and a single pair of dictionaries with 1000 atoms. As mentioned above, these algorithms have different characteristics, and therefore obtaining objective comparisons required that all parameters used in training and testing were similar to those recommended in the respective literature.

A. Quantitative Analysis

Methods for measuring the peak signal-to-noise ration (PSNR) [20], structural similarity (SSIM) [21], feature similarity (FSIM) [22], and elapsed time were used for quantitative measurement. The PSNR in decibels (dB) between the original image and the upscaled image is given by [20]. SSIM is a method that measures the quality of images based on the structural content of the original and magnified images. FSIM is based on the fact that the human visual system processes an image mainly in terms of its low-level features. Two features are considered in FSIM computation: the primary feature, i.e., phase congruency (PC), which is a dimensionless measure of a local structures significance; and the secondary feature, i.e., the image gradient magnitude. FSIM combines both of these features to characterize the local quality of an image. Higher values of PSNR, SSIM, and FSIM indicate better quality. CPU time was computed using Matlab functions (tic and toc) to measure the elapsed time for a certain process. All measurements used only the luminance channel (Y) to simplify and objectively calculate the error.

Table II lists the average values from four measurements, with the best values shown in bold. These result confirm that our proposed method clearly out-performs other methods in terms of PSNR, SSIM, and FSIM. Our method obtains a PSNR value of 25.847 dB, which is at least 11% higher than the other methods. Our proposed method also obtains an SSIM value higher by at least 14% than the other methods. In terms of FSIM, our methods outperforms the others by at least 6%. However, it should be noted that PSNR is not suitable for measuring the quality of bicubic and bilinear, as the quantitative and qualitative analysis for both methods produced some anomalies.

Although our proposed method does not provide the lowest computational time, it is still far better in this respect than Yang et al.'s algorithm [17]. Zeyde et al. [19] produced the lowest computational time in our experiments, while our method competes competitively with Kim et al. [18] with a less than 1 s differential. In future applications and research, the use of a graphics processing unit (GPU) application should offer the opportunity to decrease the computational time of the proposed method.

Nearest neighbor, bilinear, and bicubic were all excluded from the time evaluation as these had salient differences in nature to the proposed and other methods; these conventional methods are simple interpolators that do not use prior information or any learning processes. Moreover, their implementations use Matlab built-in functions, making the comparison unfair as these implement the optimization process automatically.

B. Qualitative Analysis

To evaluate the proposed method in terms of visual results, we conducted experiments using 3× magnified images to compare the proposed method to the other five methods: bilinear, bicubic, Kim et al. [18], Yang et al. [17], and Zeyde et al. [19].

Fig. 9 shows a sample of the experimental results. Our method clearly produces sharper and smoother edges and is able to clearly construct the details of a scene. The other methods all produced images with some artifacts, especially in the line and tree areas, while bicubic and bilinear also produced blurring effects in the enlarged image. Although Yang's and Zeyde's methods generate sharp edge, they still suffer from some noise and produce undesired smoothing. By contrast, Kim's method produces too strong of an edge, with unrealistic results.

Fig. 9 also shows the differences between the original images and the results produces by the respective methods. It is seen that our proposed method has the least amount of difference from the original image, which means that the proposed method has produces the least amount of artifacts, as it can clearly reconstruct edges better than the other algorithms.

IV. APPLICATION TO 3D RECONSTRUCTION

High-resolution imaging is necessary in the construction of high-precision 3D images; correspondingly, the resolution of the input image affects the quality of the 3D reconstruction precision. However, we propose the use of low-resolution image to reduce the payload of the UAV. In this section, we describe an application of our proposed system in 3D reconstruction and then compare it with the other methods.

In this experiment, a DJI Phantom 2 UAV was used to take aerial images of boxes in a field oriented at differing angles, directions, and heights. We use only one UAV because the multiple UAVs are still in the development stage. Before collecting the images, we created a flight plan that considered altitude, latitude, longitude, the distance of each turning point, and flight speed. The UAV periodically collected images from different angles in order to create 3D images. Only one operator was required to oversee the autonomous flight because the equipment was configured and the UAV can run in fully autonomous mode using defined parameters (e.g., sensors and flight plan). The flight procedure is show in Fig. 10.

The UAV was set to take original image with size 1280 × 720. Then, we created test image by downsampling the original image into size 430 × 260, and later enlarged it using the proposed SR technique. Next, we implemented the

TABLE II
COMPARISON OF THE AVERAGE QUANTITATIVE RESULTS PRODUCED BY PSNR, SSIM, AND FSIM FOR 3× MAGNIFICATION (BOLD FONT INDICATES THE BEST VALUES).

Methods	PSNR	SSIM	FSIM	Time
Nearest neighbor	22.762 ± 3.85	0.637 ± 0.12	0.736 ± 0.06	-
Bilinear	23.243 ± 3.91	0.650 ± 0.12	0.767 ± 0.06	-
Bicubic	23.361 ± 3.93	0.663 ± 0.12	0.779 ± 0.06	-
Kim et al. [18]	23.205 ± 3.93	0.674 ± 0.11	0.789 ± 0.06	5.568 ± 1.83
Yang et al. [17]	23.213 ± 3.93	0.673 ± 0.11	0.795 ± 0.05	67.189 ± 4.78
Zeyde et al. [19]	23.328 ± 3.93	0.677 ± 0.11	0.794 ± 0.05	**0.669** ± 0.04
Proposed	**25.847** ± 4.35	**0.768** ± 0.09	**0.845** ± 0.05	6.290 ± 1.15

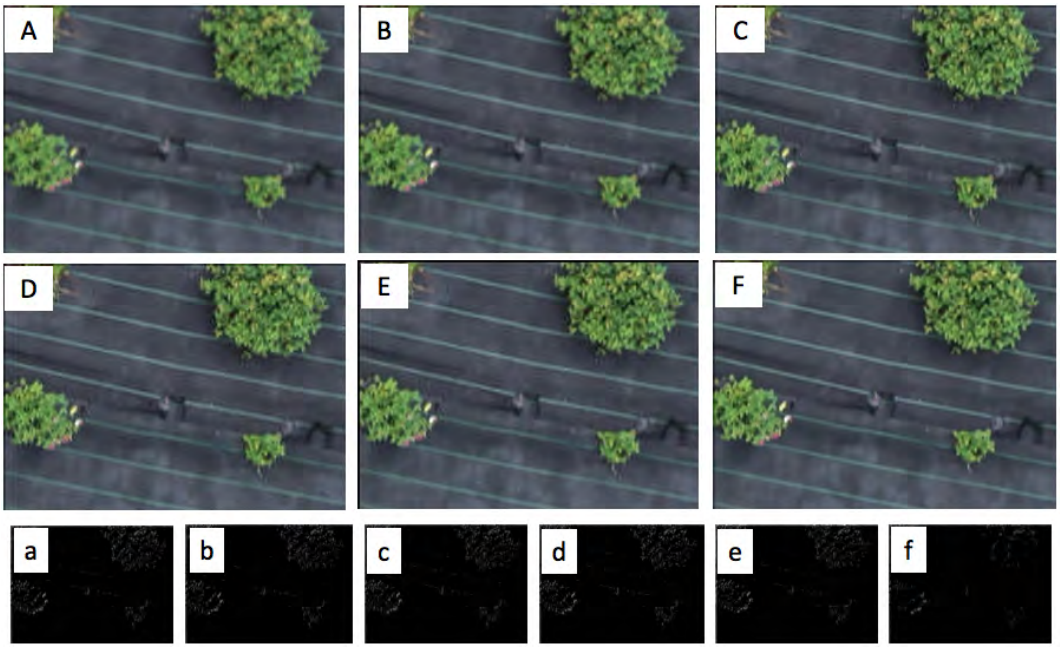


Fig. 9. Results of experiment for 3× magnification (uppercase for color image, lowercase for difference image): A-a) Bilinear, B-b) Bicubic, C-c) Kim et al. [18], D-d) Yang et al. [17], E-e) Zeyde et al. [19], F-f) The proposed method.

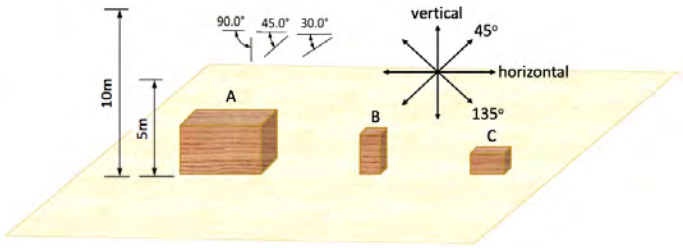


Fig. 10. Flight experimental procedure.

TABLE III
RESULTS OF MATCHING POINTS BETWEEN ORIGINAL IMAGES AND PARTICULAR METHODS.

	Camera angle		
Methods	30^o	45^o	90^o
Bilinear	488	976	727
Bicubic	598	1213	844
Proposed method	1812	2708	2068

Structure from Motion (SfM) algorithm developed by the authors of [8] on a PC (Windows 8.1, 64-bit; CPU: Intel Core i7- 4790, RAM: 32 GB, GPU: GeForce TX780). SfM employs the phenomenon by which humans can recover a 3D structure from a projected 2D (retinal) motion field of a moving object or scene, finding correspondence between images by searching them for features that can be recognized from different angles and distances and matching them in a manner similar to rotating pieces of a jigsaw puzzle until the best match is found. As in a puzzle, the more images that can be sorted and matched, the better the 3D model that can be achieved. The 3D dense cloud image we constructed from various methods is shown in Fig. 11.

Table III lists the result produces by particular methods from matching points with original images. Using a SIFT algorithm, we extracted the feature points from each image and aligned the matching points. The results show that our proposed method produced the highest number of matching points of all of the methods.

Finally, we measured the height and width of each box and then calculated the error by comparing these to the real scale, with the best result indicated by the lowest error value.

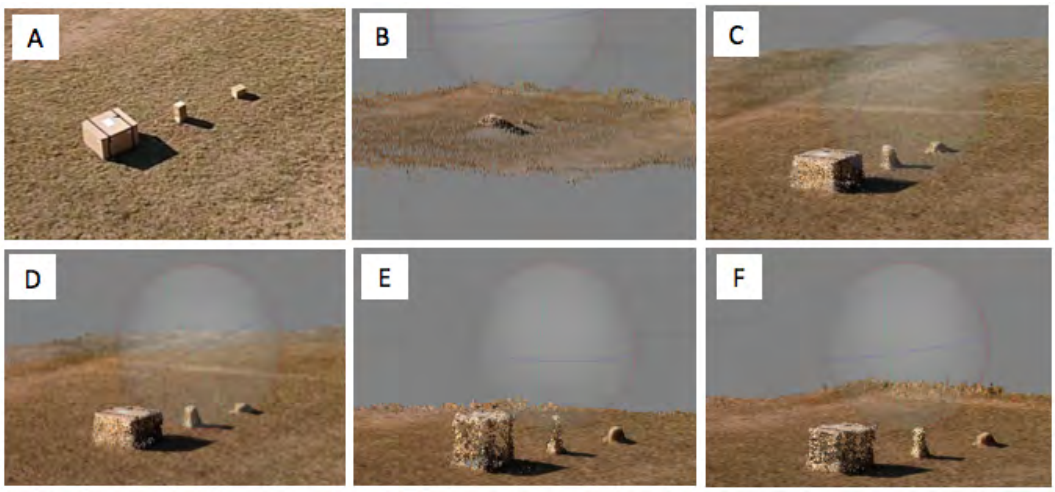


Fig. 11. Results of 3D image reconstruction: (A) Sample of 2D image, (B) From test image, (C) From original image, (D) From the proposed method, (E) From bilinear, (F) From bicubic

TABLE IV
3D MEASUREMENT RESULTS. THE MEASUREMENT IS DETERMINED BY AVERAGING THE DISTANCE OF SIX PAIRS OF POINT'S SAMPLE AND ERROR IS THE DIFFERENCE BETWEEN REAL AND OBSERVED MEASUREMENT (A^* IS USED AS SCALE REFERENCE AND BOLD FONT INDICATE AS THE BEST VALUE.)

			Height (cm)		Width (cm)		
UAV's Height	Methods	A^*	B	C	A	B	C
	Real scale	61	47	25	101	24	32
5m	Original	61	**45.25 (-1.75)**	**22.76 (-2.24)**	106.34 (+5.34)	**22.94 (-1.06)**	**32.27 (+0.27)**
	Proposed	61	42.83 (-4.17)	20.45 (-4.55)	**105.82 (+4.82)**	20.34 (-3.66)	30.26 (-1.74)
	Bicubic	61	41.03 (-5.97)	16.94 (-8.06)	107.54 (+6.54)	20.25 (-3.75)	25.98 (-6.02)
	Bilinear	61	38.93 (-8.07)	17.32 (-7.68)	–	–	–
10m	Original	61	40.71 (-6.29)	20.68 (-4.32)	107.69 (+6.69)	16.73 (-7.27)	26.37 (-5.63)
	Proposed	61	**45.78 (-1.22)**	**28.13 (+3.13)**	**100.37 (-0.63)**	**17.88 (-6.12)**	**28.84 (-3.16)**
	Bicubic	61	40.95 (-6.05)	17.98 (-7.02)	89.73 (-11.27)	12.09 (-11.91)	15.01 (-16.99)
	Bilinear	61	35.18 (-11.82)	–	115.95 (+14.95)	15.06 (-8.94)	–

The height of box A was used as the scale reference to determine the dimensions of the other boxes. The observed measurements were calculated by taking six sample pairs of points and determining the average distances of each pair using the Euclidian distance measure. The measurement results are listed in Table IV. In the case where the width of box A is 10m, it is seen that our proposed method can decrease the measurement error to a millimeter order of magnitude, while other methods have at least an approximate 11cm error. Some results for the bilinear method could not be calculated owing to bad reconstruction results.

In the case where the imaging was performed from a height of 5m, the original image has the highest precision, even better than the proposed method. However, the proposed method can still keep its measurement error lower than the other methods, and it has the least error in measuring the width of box A.

In the case where imaging occurred from 10m, we found that the proposed system produced an error even lower than that of the original image - a striking result. The greater height of the UAV meant that images with lower detail, or lower amount of pixels per centimeter (PPCM), were produced. In this case, an image taken from 10m has around 1 PPCM, while one taken from 5m has around 2 PPCM. Based on this, we know that the images from 10m suffered at least twice the noise of the 5m image, and the results prove that our proposed method is able to recover test images, reinsert high-frequency details, and repair some of the inconsistency in edges owing to a lowered PPCM.

Bigger, well-shaped objects are easy to reconstruct. In this experiment, we used boxes, not trees, to simplify the experiment. However, in the future we will attempt to conduct real field phenotyping. We note that the lowest error was achieved by our proposed method in calculating the width of box A, which did this with an accuracy within a millimeter order of magnitude. However, for smaller dimensions such as the height of C or the width of B, it will be harder to obtain accurate measurements.

V. CONCLUSIONS

In order to provide more reliable of the agriculture image collection, a UAV based system for high-precision 3D map was studied. The UAV is equipped with a user friendly ground station application which designed to be the interface between a human operator and the UAV to carry out mission planning, flight command activation, and real-time flight monitoring. Based on the navigation data, and the way-points generated by the ground station, the UAV could be automatically navigated

to the desired waypoints and hover around each waypoint to collect field image data. By so doing, the aerial images at each point could be captured automatically.

Using only low-resolution, we are able to reduce the payload of the UAV in order to increase the flight time of UAV. The system was tested and proven that it can achieved error of a millimeter order of magnitude. The UAV imaging system developed here offers enhanced capabilities in dealing with agricultural remote sensing needs because, unlike traditional remote sensing systems, it has good spatiotemporal capabilities. When combined with other appropriate sensors like field server, the UAV imaging system has further promise as an agricultural remote sensing platform.

ACKNOWLEDGEMENTS

This work was supported by CREST, Japan Science and Technology Agency. Muhammad Haris would like to thank the Indonesia Endowment Fund for Education (LPDP) Scholarships from the Ministry of Finance, The Republic of Indonesia for doctoral degree scholarship.

REFERENCES

[1] S. K. Seelan, S. Laguette, G. M. Casady, and G. A. Seielstad, "Remote sensing applications for precision agriculture: A learning community approach," *Remote Sensing of Environment*, vol. 88, no. 1, pp. 157–169, 2003.

[2] J. Everaerts *et al.*, "The use of unmanned aerial vehicles (uavs) for remote sensing and mapping," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 37, pp. 1187–1192, 2008.

[3] G. Grenzdörffer, A. Engel, and B. Teichert, "The photogrammetric potential of low-cost uavs in forestry and agriculture," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 31, no. B3, pp. 1207–1214, 2008.

[4] S. C. Chapman, T. Merz, A. Chan, P. Jackway, S. Hrabar, M. F. Dreccer, E. Holland, B. Zheng, T. J. Ling, and J. Jimenez-Berni, "Pheno-copter: a low-altitude, autonomous remote-sensing robotic helicopter for high-throughput field-based phenotyping," *Agronomy*, vol. 4, no. 2, pp. 279–301, 2014.

[5] Y. Huang, S. J. Thomson, W. C. Hoffmann, Y. Lan, and B. K. Fritz, "Development and prospect of unmanned aerial vehicle technologies for agricultural production management," *International Journal of Agricultural and Biological Engineering*, vol. 6, no. 3, pp. 1–10, 2013.

[6] C. Zhang and J. M. Kovacs, "The application of small unmanned aerial systems for precision agriculture: a review," *Precision agriculture*, vol. 13, no. 6, pp. 693–712, 2012.

[7] M. Haris and H. Nobuhara, "Super-resolution based on edge-aware sparse representation via multiple dictionaries," in *Proceedings of the 11th Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*, vol. 3, 2016, pp. 40–47.

[8] Y. Furukawa and J. Ponce, "Accurate, dense, and robust multiview stereopsis," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 32, no. 8, pp. 1362–1376, 2010.

[9] T. Fukatsu and M. Hirafuji, "Field monitoring using sensor-nodes with a web server," *Journal of Robotics and Mechatronics*, vol. 17, no. 2, pp. 164–172, 2005.

[10] W. Guo, T. Fukatsu, and S. Ninomiya, "Automated characterization of flowering dynamics in rice using field-acquired time-series rgb images," *Plant methods*, vol. 11, no. 1, p. 7, 2015.

[11] T. Sritarapipat, P. Rakwatin, and T. Kasetkasem, "Automatic rice crop height measurement using a field server and digital image processing," *Sensors*, vol. 14, no. 1, pp. 900–926, 2014.

[12] M. Hirafuji, H. Yoichi, T. Kiura, K. Matsumoto, T. Fukatsu, K. Tanaka, Y. Shibuya, A. Itoh, H. Nesumi, N. Hoshi *et al.*, "Creating high-performance/low-cost ambient sensor cloud system using openfs (open field server) for high-throughput phenotyping," in *SICE Annual Conference (SICE), 2011 Proceedings of*. IEEE, 2011, pp. 2090–2092.

[13] D. P. Paine and J. D. Kiser, *Aerial photography and image interpretation*. John Wiley & Sons, 2003.

[14] H. Xiang and L. Tian, "Development of a low-cost agricultural remote sensing system based on an autonomous unmanned aerial vehicle (uav)," *Biosystems engineering*, vol. 108, no. 2, pp. 174–190, 2011.

[15] S. C. Park, M. K. Park, and K. M. Gi, "Super-resolution image reconstruction: A technical overview," *IEEE Signal Processing Magazine*, vol. 20, pp. 21–36, 2003.

[16] F. Remondino, L. Barazzetti, F. Nex, M. Scaioni, and D. Sarazzi, "Uav photogrammetry for mapping and 3d modeling–current status and future perspectives," *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 38, no. 1, p. C22, 2011.

[17] J. Yang, J. Wright, T. S. Huang, and Y. Ma, "Image super-resolution via sparse representation," *Image Processing, IEEE Transactions on*, vol. 19, no. 11, pp. 2861–2873, 2010.

[18] K. I. Kim and Y. Kwon, "Single-image super-resolution using sparse regression and natural image prior," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 32, no. 6, pp. 1127–1133, 2010.

[19] R. Zeyde, M. Elad, and M. Protter, "On single image scale-up using sparse-representations," in *Curves and Surfaces*. Springer, 2012, pp. 711–730.

[20] M. Irani and S. Peleg, "Motion analysis for image enhancement: Resolution, occlusion, and transparency," *Journal of Visual Communication and Image Representation*, vol. 4, no. 4, pp. 324–335, 1993.

[21] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *Image Processing, IEEE Transactions on*, vol. 13, no. 4, pp. 600–612, 2004.

[22] L. Zhang, L. Zhang, X. Mou, and D. Zhang, "Fsim: a feature similarity index for image quality assessment," *Image Processing, IEEE Transactions on*, vol. 20, no. 8, pp. 2378–2386, 2011.

Muhammad Haris received the B.Sc. in Computer Science from the University of Indonesia, Depok, Indonesia, in 2009. He is currently a doctoral student in Department of Intelligent Interaction and Technologies, University of Tsukuba, Japan. His research interests include image processing, super-resolution, image enhancement, and 3D reconstruction especially for aerial images.

Seita Sukisaki received B.Eng from School of Engineering, University of Tsukuba, Japan. He is currently a master student in Department of Intelligent Interaction and Technologies, University of Tsukuba, Japan. His research interests include robotics especially UAV.

Ryo Shimomura received the B.Sc. in Robotics from the Chiba Institute of Technology, Japan in 2016. He is currently a master student in Department of Intelligent Interaction and Technologies, University of Tsukuba, Japan. His research interests include autonomous mobile robot, navigation, point cloud processing, and cooperative control for outdoor-type robot.

Zhang Heming recieved B.Eng from School of Electrical Engineering, Nantong University, China. He is now a research student in Department of Intelligent Interaction and Technologies, University of Tsukuba, Japan. His current research include position estimation, position control of UAV, and formation flight.

Li Hongyang received the B.Sc. in Electronic Information Science and Technology from the Tianjin Normal University, Tianjin, China, in 2010. He is currently research student in Department of Intelligent Interaction and Technologies, University of Tsukuba, Japan. His research interests include UAV aerial image, 3D reconstruction, especially for formation flight.

Hajime Nobuhara received the Dr.Eng. in Computational Intelligence from the Tokyo Institute of Technology, Japan in 2003. He is currently associate Professor at Department of Intelligent Interaction Technologies, University of Tsukuba, Japan. Before that, he was a post doctoral fellow at University of Alberta, Canada. His research interests include computational intelligence, image processing, web intelligence, bio informatics, and UAV.

Proceedings of the APAN – Research Workshop 2016
ISBN 978-4-9905448-6-7

Development of Wireless Sensor Node for Landslide Detection

Kim, Hyoung Woo

***Abstract*— Landslides have frequently occurred on natural slopes during periods of intense rainfall. With a rapidly increasing population on or near steep terrain in Korea, landslides have become one of the most significant natural hazards. Thus, it is necessary to protect people from landslides and to minimize the damage of houses, roads and other facilities. To accomplish this goal, many landslide prediction methods have been developed around the world. In this study, a prototype of landslide detection is introduced. This system is based on the wireless sensor network (WSN) that is composed of sensor nodes, gateway, and server system. Sensor nodes comprising sensing and communication part are implemented to detect ground movement. A sensing part is designed to measure inclination angle and acceleration accurately, and a communication part is deployed with Bluetooth (IEEE 802.15.1) module to transmit the data to the gateway. To verify the feasibility of this landslide prediction system, a series of experimental studies was performed at a small-scale earth slope equipped with an artificial rainfall dropping device. It is found that sensing nodes plated at slope can detect the ground motion when the slope starts to move. It is expected that the prototype of landslide detection can provide early warnings when landslides occurs.**

***Index Terms*—Enter key words or phrases in alphabetical order, separated by commas.**

I. Introduction

Landslides are a serious geological hazard caused when masses of rock, earth and debris flow down a steep slope during periods of intense rainfall and rapid snow melt. It is reported that landslides happen more repeatedly than before and their damages are increasing due to global warming [1]. Every year, landslides occur at natural slopes in Korea, resulting in loss of human life, destruction of houses and facilities, damage to roads, rail lines and pipelines, vehicle accidents and train derailments, damage to agricultural land, livestock, and forest stands, and many other losses. In order to prevent landslide, hill slopes that are unstable should be strengthened. To mitigate its damage, a system that can predict the occurrence of a landslide at a specific site is required. The immediate detection of landslide activity provided by real-time systems can be crucial in saving human lives and protecting property. The continuous data provided by remote real-time monitoring permits a better understanding of dynamic landslide behavior that enables engineers to create more effective designs to prevent or halt landslides. In this study, a prototype of landslide detection is introduced. This is based on wireless sensor network and designed to detect debris flows that is frequently occur in Korea [2]. To verify the feasibility of this landslide prediction system, a series of experimental studies was performed at a small-scale earth slope increasing soil moisture content. It is found that sensing nodes planted at the slope can detect the ground motion when the slope starts to move. It is expected that the landslide prediction system by wireless senor network will provide early warnings when landslides such as debris flow occurs.

Kim, H.W. is with Enterprise Network Business Performing Department of KT Corp., Seoul, Korea (e-mail: hyoungwoo.kim@kt.com)

II. Debris Flow Instrumentation

A. Sensing debris flow preceding events

Most debris flows are triggered by intense rainfall or rapid snowmelt, therefore monitoring hydrologic conditions can provide advance knowledge of hazardous conditions. Precipitation is often an essential parameter of interest for warning application. After precipitation infiltrates the soil surface, it moves through the soil as both unsaturated and saturated subsurface flow. A variety of instrumentation may be used to monitor subsurface hydrologic conditions that occur prior to debris flow initiation. Most soils that are susceptible to mobilization into debris flows have relatively high permeability and therefore the use of standpipe or Casagrande piezometers with small diameters are preferred. Time domain reflectometry (TDR) that has been recently developed to measure soil moisture and earth deformation can be also used [3].

For reliable debris flow warning systems, it is not enough to only detect debris flow occurrence. Therefore it is essential to sense complementary parameters that can confirm hazardous conditions and help prevent false alarms. In addition to the hydrologic monitoring, ground deformation measurements are

desired. Cable extensometers can be used to measure extension or contraction across tension cracks. They can also be mounted with a cable extended down a borehole and through the failure zone to stable ground beneath a slide. Inclinometers can be also used to make intermittent or manual measurements of a minor deformation of a plastic or aluminum casing that is installed in a borehole [4]. Alternatively, tilt meters can be permanently installed to make continuous measurements. Recently, ground deformation can also be intermittently or continuously measured using geospatial positioning systems (GPS) [5].

B. Measuring debris flow dynamics

Installation and maintenance of sensors that require contact or close proximity to unstable steep slopes where debris flows may occur are dangerous. Alternatively, sensors that can detect debris flow occurrence and do not require close proximity to the area are preferred for long-term reliable operation [5]. To detect and monitor debris flows, the Acoustic Flow Monitor (AFM) was designed by the U.S. Geological Survey Cascades Volcano Observatory. It has been successfully used internationally as part of real-time warning systems in valleys threatened by such flows. The AFM system has also been proven to be an effective tool for monitoring debris flows [6].

Where debris flow is channelized and it is possible to install an overhead boom or cable, distance between a range finder and the flow surface can be sensed. Ultrasonic range finders, also known as distance meters, can measure distances up to 20m or more by emitting ultrasonic or microwave bursts and measuring the round-trip travel time of the emissions that reflect back off the flow surface. Another method utilizes surface velocity sensing with Doppler radar, and this method is similar to ultrasonic range finders. Video cameras and recorders have also been used to record debris flows [5].

C. Features of traditional measuring methods

The assessment of debris flow can be usually undertaken by means of monitoring. The measurement of superficial displacement is the simplest way to observe the evolution of debris flows and to analyze the kinetics of the movement. A variety of surveying techniques have been used to track the superficial movements of unstable areas. Conventionally, tapes and wire devices have been used to measure changes in distance between points. Levels, theodolites, and total station measurements provide the coordinates and changes of target by which the ground motion can be detected [7].

As discussed previously, various prediction sensors have been proposed such as multi-point borehole extensometer, tilt sensors, displacement sensors, and volumetric soil water content sensors. Most of these sensors, however, require drilling 20-30 meter holes into the surface, making the installation very expensive and requiring skilled labor. Furthermore, these are expensive sensors, making wide scale deployment infeasible. Installing a single sensor for monitoring an entire hill side is not sufficient as the properties of the ground change every 100-200 meters. Wiring each sensor to a central data logger is also not feasible in the steep terrain because it requires high maintenance, and is subjected to a single point of failure [8]. In addition, the locations where the data gathering will take place lack electrical and communication infrastructure, making conventional monitoring systems that rely on power grids and wired communication links inappropriate. To solve these problems, emerging technologies to develop landslide monitoring system by wireless sensor network are being studied vigorously throughout the world [8-9].

III. IMPLEMENT OF WIRELESS SENSOR NODE

A. Wireless Sensor Network (WSN)

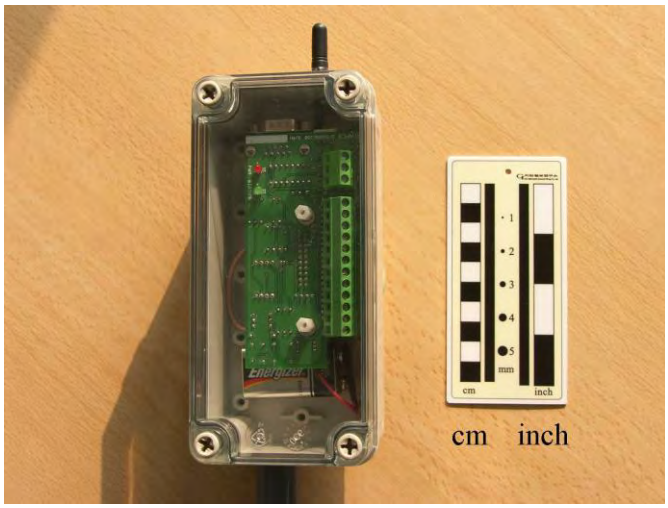
A wireless sensor network (WSN) is a wireless network consisting of spatially distributed autonomous devices using sensors to monitor physical or environmental conditions, such as light, temperature, sound, vibration, pressure, motion or pollutants, at different locations. In addition to one or more sensors, each node in a sensor network is typically equipped with a radio transceiver or other wireless communications device, a small microcontroller, and a power supply, usually a battery. It is anticipated that aggregating sensors into sophisticated computation and communication infrastructures, called wireless sensor networks, will have a significant impact on a wide range applications such as combat field surveillance, intrusion detection, disaster management, structural health monitoring, asset monitoring, and environmental monitoring. The fundamental goal of a wireless sensor network is to produce global information from local data obtained by individual sensors. WSN processes data gathered by multiple sensors to monitor events in an area of interest [10]. By combining sensed data from a large number of distributed sensors, a global monitoring can be performed. This point is a major difference compared with traditional monitoring methods which depend on existing fixed single sensors [8].

The WSN can be divided into three major parts; sensor nodes, the gateway and the server system. Sensors are generally equipped with data processing and communication capabilities. The sensing circuitry measures parameters from the environment surrounding the sensor and transforms them into electric signals. Sensor nodes composed of sensing part and communication part send collected data, usually via radio transmitter, to a gateway (a base station or sink node). In general, the sensing circuitry generates analog signals, and therefore these signals are sampled using A/D converter and stored in the on-board memory as a sequence of digital values. The sensed data can be further processed using data processor (microprocessor) prior to sending them over to the base station [10]. Wireless sensor nodes are mainly to encode and encapsulate the measured parameters, then to transmit them to the gateway via wireless link. Currently available sensors employ one of the following types of radios. The simplest alternative is to use a free band (315/433/868/916 MHz) which has a bandwidth in the range 20-50 kbps. Another models

support an IEEE 802.15.1 (Bluetooth) or 802.15.4 (ZigBee) radio operating in the 2.4 GHz band. The radio range varies with a maximum of about 300m (outdoor) for the first radio type, 30m for the IEEE 802.15.1 radio, and 125m for 802.15.4 radios. The gateway is responsible for polling sensor nodes regularly to acquire and record the relevant data [11-12]. The server system gathers relevant hydrological and geological data via Internet from the threatened region where the landslides are likely to occur. The server system has to fulfill extracting and displaying relevant information to assist authorities to make decisions, and furthermore generating alert or alarm signals if certain thresholds are reached or exceeded. Since the implementation of an alert service is very important for hazard monitoring, such a service may, in general, need the development of intelligent software for evaluating all the measurements from different sensors against predefined thresholds [13].

B. Development of landslide detection sensor

A debris flow involves gravity-driven motion of solid-fluid mixtures with abrupt surge fronts, free upper surfaces, variably erodible basal surfaces, and compositions that may change with position and time. When a debris flow initiates, the rapid initial landslide may continue downslope without confinement. In granular materials this always leads to disintegration, producing flow-like motion [5]. In order to detect such a ground movement, a sensor node based on wireless communication is developed. The sensor node combines a sensing element, or transducer, with A/D conversion, signal processing, memory, radio-frequency communication and battery power supply. For the landslide detection, a low-g (±1g range) ADXL 202 biaxial accelerometer is used (http://www.analog.com). The inclination angle can be obtained from the predefined relationship between inclination angle and acceleration. The sensor node and the view of PCB are shown in Figures 1a and 1b, respectively.



(a)

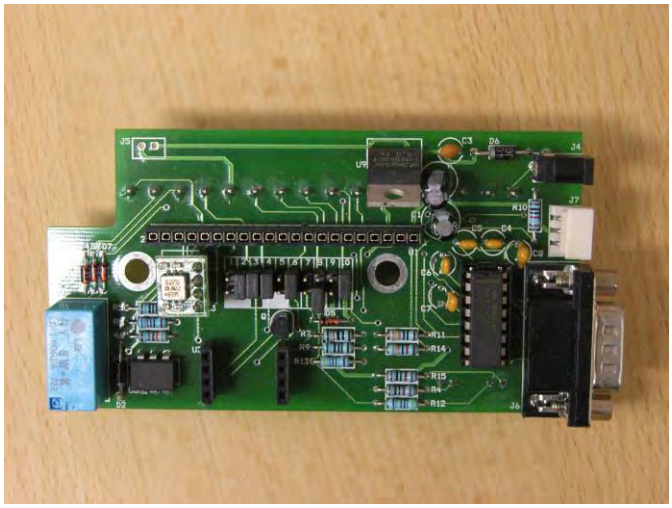


Fig. 1. Photographs of landslide detection sensor node: (a) plastic box containing sensor node; (b) view of PCB.

When the data is ready for communication, the Bluetooth wireless transceiver is utilized. Operating on the ISM (Industrial Scientific and Medical equipment) band, this can communicate with ranges as large as 30 m (line-of-sight). It is found that the maximum power consumption is about 43mA during communication.

C. Experimental studies

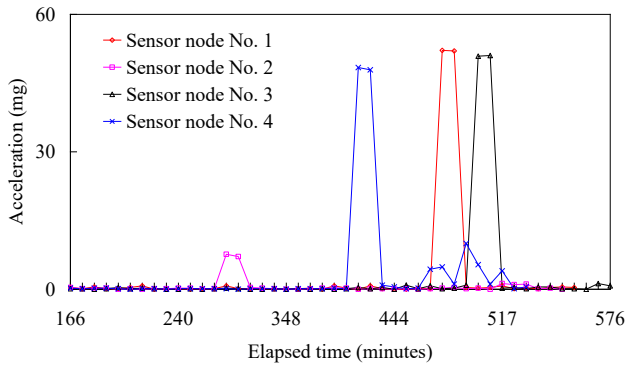
In order to verify the feasibility of landslide detection sensor, a series of experimental studies was performed. A small-scale artificial earth slope has height of 0.8m and base of 1.2m (Figure 2a). Geotechnical laboratory tests show that the soil is classified as a silty-sand, and the cohesion is 0.11 kg/cm² and angle of internal friction is 39.7 degree. Five sensor nodes distributed on the slope as shown in Figure 2b, and water supplied by artificial rainfall dropping device to simulate real storms. The rainfall intensity was uniformly set to 30mm/hr, and the test continued until all the sensor nodes turned over completely due to landslide.

(a)

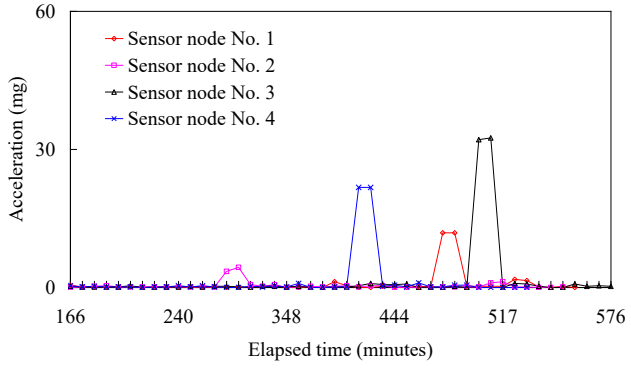
(b)

Fig. 2. Photographs of experiments: (a) view of artificial earth slope and wireless rainfall gauge; (b) array of sensor nodes.

After about four hours, some channels occurred, upper soil is eroded, and granular materials disintegrated and flowed down with water. Simultaneously, the sensor nodes started to incline and move downward slightly. Figure 3 shows x-axis and y-axis acceleration change of each sensor node. Only four acceleration curves are shown because sensor node no. 5 was out of order during the experiment. It is found that the sensor node no. 2 that was installed at the lowest position started to move first and the remaining sensor nodes turned over successively.



(a) x-axis acceleration

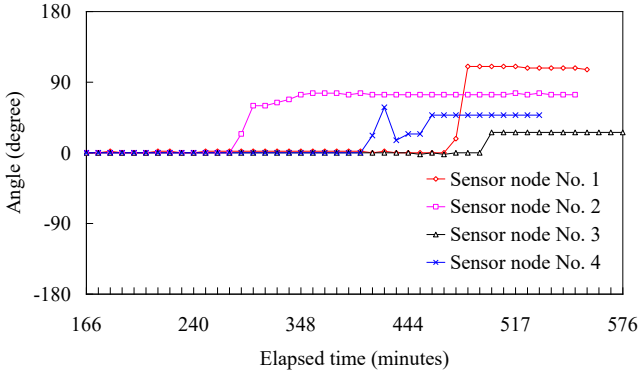


(b) y-axis acceleration

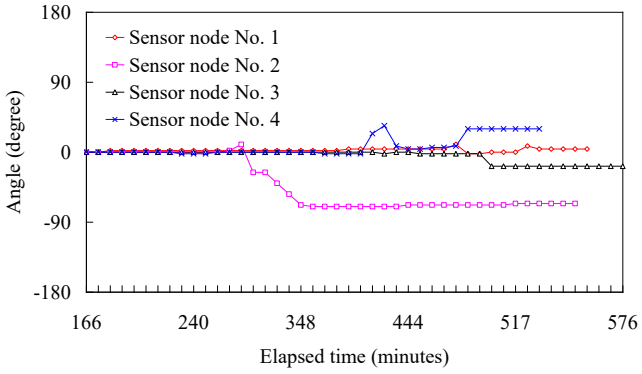
Fig. 3. Acceleration graphs of sensor nodes: (a) x-axis acceleration; (b) y-axis acceleration.

Figure 4 shows x-axis and y-axis inclination angle change of sensor nodes. As previously noticed, it is found that the sensor node no. 2 starts to move first and the remaining sensor nodes are turned over successively. It is also observed that the time of acceleration and inclination change is identical, and from this fact, it is possible to detect ground motion by measuring acceleration and inclination angle. In addition, the magnitude of ground motion can be estimated by acceleration value, and the inclination of slope can be configured by angle data.

It is expected that the landslide detection sensor nodes developed in this study can be applied to the slope where debris flows are likely to happen, because granular materials such as gravels and stones may hit the sensor nodes during debris flow initiation. If the sensor nodes are distributed more densely, the possibility of detecting debris flow will increase higher. Thus, debris flow can be predicted, if the thresholds are predefined according to the slope condition.



(a) x-axis inclination angle



(a) y-axis inclination angle

Fig. 4. Inclination angle graphs of sensor nodes: (a) x-axis inclination angle; (b) y-axis inclination angle.

IV. CONCLUSIONS

In this paper, a prototype of landslide detection by WSN has been described. As information and communication technology (ICT) develops, landslide monitoring systems are becoming more precise and cost-effective. Landslide

monitoring system by wireless sensor network will be an alternative to detect and predict slope failure including debris flows. In order to develop the technology, further studies are needed.

It may be difficult to determine whether the slope is stable or not solely using data collected by landslide monitoring because slope stability depends on soil type and soil condition, groundwater table, soil moisture content, slope failure type, rainfall, and vegetation, etc. If the predefined threshold is set too low, there will be too many false alarms, so that genuine warnings will not be heeded. On the other hand, if the threshold is set too high, events that will cause damage may be ignored (miss-alarm). Therefore, it is necessary to predefine the appropriate thresholds to determine the slope stability, and related work is required.

V. FUTURE WORKS

Since this study was performed several years ago, the wireless sensor node for landslide detection is old fashioned. Due to the recent remarkable improvement of wireless sensor, communication transmission distance between sensor nodes is longer than before and can be operated by low power. Actually, wireless sensor node introduced in this study cannot be applied to real landslide, because there are many problems such as power supply, protection from lightning, and lack of threshold. Conventional landslide detection method can provide more reliable results by analyzing several composite sensors such as soil moisture content sensor, inclinometer, rainfall gauge, and tension meters, and so on. In addition, threshold that is used for issuing warning should be established at the landslide case by case. To do this work, geotechnical engineer and ICT engineer should make cooperation each other. Since the speed of landslide propagation is very fast, it is very difficult to detect the mechanism at an early stage, and to issue alarm for evacuation. In order to solve this problem, numerous studies for landslide detection are being performed and deployed at various sites in Korea. It is expected that this work will contribute to provide a small example.

REFERENCES

[1] Geertsema, M., Clague, J.J., Schwab, J.W., and Evans, S.G. "An overview of recent large catastrophic landslides in northern British Columbia, Canada," *Engineering Geology*, vol.83, pp.120-143, 2006.

[2] Kim, H.W. "Landslide prediction system by wireless sensor network," *Proceedings of the ITFE Summer Conference*, pp.191-195, 2007.

[3] Alimi-Ichola, I., and Gaidi, L. "Influence of the unsaturated zone of soil layer on the solute migration," *Engineering Geology*, vol.85, pp. 2-8, 2006.

[4] USGS. Landslide Hazards Program [Online]. Available: http://landslides.usgs.gov/monitoring/hwy50/U.S

[5] Jakob, M, and Hungr, O. *Debris-flow Hazards and Related Phenomena*, Chichester, UK, Praxis Publishing, Springer, 2005.

[6] USGS. "Acoustic Flow Monitor System – User Manual," Open-File Rep. 02-429, 2005.

[7] Gili, J.A., Corominas, J., and Rius, J. "Using Global Positioning System techniques in landslide monitoring," *Engineering Geology*, vol.55, pp.167-192, 2000.

[8] Sheth, A., Tejaswi, K., Mehta, P., Parekh, C., Bansai, R., Merchant, S., Singh, T.N., Desai, U.B., Thekkath, C.A., and Toyama, K. Poster Abstract, "A Sensor Network Based Landslide Prediction System," *In Proceedings of Sensys 2005*, 2005.

[9] Terzis, A., Anandarajah, A., Moore, K., and Wang, I.J. "Slip Surface Localization in Wireless Sensor Networks for Landslide Prediction," *In Proceedings of Sensys 2006*, 2006.

[10] Olariu, S., and Xu, Q. "A simple and robust infrastructure for massively deployed wireless sensor networks," *Computer Communications*, vol.28, pp.1505-1516, 2004.

[11] Baronti, P., Pillai, P., Chook, V.W.C., Chessa, S., Gotta, A., and Hu, Y.F. "Wireless sensor networks: A survey on the state of the art and the 802.15.4 and ZigBee standards," *Computer Communications*, vol.30, pp.1655-1695, 2007.

[12] Lee, R.G., Chen, K.C., Lai, C.C., Chiang, S.S., Liu, H.S., and Wei, M.S. "A backup routing with wireless sensor network for bridge monitoring system," *Measurements*, vol.40, pp.55-63, 2006.

[13] Effen, M.C., Quintela, D.H., Jordan, R., Westhoff, W., and Moreno, W. "Wireless Sensor Networks for Flash-Flood Alerting," *Proceeding of the Fifth IEEE International Caracas Conference on Devices, Circuits and Systems*, Dominican Republic, pp.142-146, 2004.

How Smooth is an ISP Changeover Process?

Waiting W. T. Fok and Rocky K. C. Chang
Department of Computing
The Hong Kong Polytechnic University
Kowloon, Hong Kong
Email: {cswtfok|csrchang}@comp.polyu.edu.hk

***Abstract*—For various reasons, an enterprise may change to a new ISP for network connectivity, and the ISP changeover will inevitably interrupt the normal network operations. This paper is the first to document this process and to measure the impact of ISP changeover on the network performance for HARNET which connects the eight major universities in Hong Kong. We set up a network measurement platform to monitor the performance and route changes during the one-week long migration process. Our continuous monitoring results clearly show the different phases of the migration and the impact on the end-to-end performance. Except for short periods of instability and disruptions, the entire ISP changeover process is considered quite smooth.**

I. Introduction

Changing Internet Service Provider (ISP) does not happen often for enterprise networks, but it does happen from time to time for various reasons (e.g., lower cost and better performance). A possible barrier to migrating to a new ISP is perhaps the impact on the networks during the changeover [1]. However, there is no formal study on measuring the impacts and documenting the changeover process. This paper is the first to fill this gap by reporting a recent ISP changeover for HARNET (The Hong Kong Academic and Research NETwork) [2] which connects the campus networks of the eight major universities in Hong Kong which has over 100,000 staff and students. This study details the key steps of the changeover process and reports the results of the network measurement designed specifically for this changeover event.

Besides connecting among the eight universities, HARNET also provides Internet connectivity to the universities through (1) a designated ISP in Hong Kong, (2) HKIX, an Internet exchange hosted in a university interconnecting most networks in Hong Kong, and (3) peering to other overseas research and academic networks, such as CERNET, Internet2, and TEIN3. Since we deployed a network monitoring platform for HARNET in January 2009, we experienced three planned ISP changeovers and numerous network events, including local router faults, route discriminations and route changes, and submarine cable outages [3]. We correlate the measurements from data plane, such as round-trip time (RTT) and packet loss events, and that from control plane, including forward and reverse path routes and BGP information, to identify the root causes [4].

In this paper, we present our measurement results and path information collected during the ISP migration. During the one-week planned network change, all of the aforementioned network connections were reconfigured one after the other. Some of them were disconnected temporarily, relying on alternate routes to recover accessibility during the short transition period, and then re-connected again. Others involved adding a new route, adjusting the incoming and outgoing routing preferences to divert traffic to the new route, and finally removing the old route. Although the ISP changeover is a planned event, we have observed that the network configuration changes create similar disruptions as those from network equipment failures and misconfigurations.

The remaining of the paper is organized as follows. Section II documents the main steps in the ISP changeover process and the expected outcomes. Section III describes our platform for monitoring the network performance during the migration. Section IV reports the measurement results and evaluates the impact of the changeover on the network availability and performance. After discussing the related works in Section V, we conclude the paper in Section VI.

II. ISP Changeover Plan

Figure 1 illustrates the layer-3 network diagram of HARNET as of April 2015 after the changeover. Two routers CR$[1|2]$ are hosted by each of the eight universities of HARNET to connect themselves internally and externally to two optical hubs through an optical metro network. The optical network connects to four network switches, two at each optical hub. The core routers HR* and network switches (not shown in the diagram) are hosted in optical hubs in CUHK and HKU, two members of HARNET. Routers HR$[IX|1|2]$ are located in CUHK, and HR$[3|4]$ in HKU. Internally, they are linked to the two network switches at the site connecting to the optical metro network. A dedicated 10 Gbps link connects HR1 and HR3 as shown in the diagram. Not shown in the diagram for simplicity, HRIX router at CUHK is connected to a network switch at HKU to link up HRIX and HR$[3|4]$. Routers HR* also connect to external networks, including HKIX, ISP, CERNET, and TEIN3. The high-level HARNET network structure before and after the transition are identical except

- TEIN3 connects to HR4 in the new configuration, instead of HR3,
- ISP connects to HR1 in the new configuration, instead of HR2, and
- CERNET connects to HR2 in the new configuration, instead of HR1.

Unused connections between routers, optical networks, and switches are not shown in Figure 1.

Moreover, the network connections can be categorized into four groups:

1) Within HARNET, i.e., between CR$[1|2]$ and HR$[1|3|IX]$ routers. The link capacity is 10 Gbps.

Steps	Dates and Times (GMT)	Tasks
1 HKIX Preparation	4/3 1:00-9:30	Installed a new HRIX router and new connections to HKIX and HARNET network switches, and verified the connections to HKIX.
2 HKIX Migration	4/3 22:00-22:30	Switched then existing physical connections to the HRIX new router, switched BGP neighbors to the new HRIX router and connections.
3 TEIN Migration	5/3 22:00 - 6/3 0:00	Shut down the old TEIN3 connection to HR3, relocated the connection to HR4, installed new network port module to HR3, installed new connection between HR3 and HR4, and installed connection between the new ISP and HR3.
4 CERNET Migration	6/3 22:00 - 7/3 0:00	Shut down the old CERNET connection to HR1, configured the new CERNET connection to HR2, installed new network port module to HR1, and installed connection between the new ISP and HR1.
5 ISP Migration	9/3 16:00	Announced prefix to ISP, raised the priority of the new ISP connection over the old ISP connection.

TABLE I. THE ISP CHANGEOVER SCHEDULE.

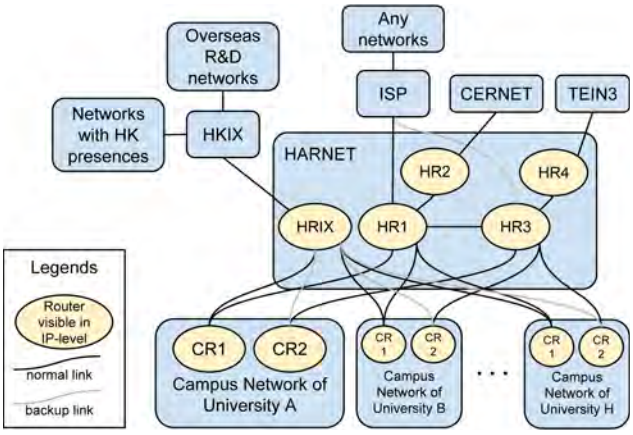


Fig. 1. The HARNET's network connectivity after the ISP changeover on 9 March 2015.

2) Peering with other local networks and overseas research networks at HKIX with link capacity of 20 Gbps.
3) Around 100 Mbps connections with CERNET and TEIN3 networks through special peering at HR2 and HR4, respectively.
4) General Internet connectivity through ISP. It is a paid service, and the eight universities share a bandwidth of around 4 Gbps.

HARNET prioritizes the connections 1-3 over connection 4 to optimize the bandwidth usage.

The one-week ISP changeover schedule is listed in table I. To minimize the disruption to users, most of the migration tasks were completed between 0:00 AM and 8:00 HKT (i.e. between 16:00 and 0:00 GMT). The detailed tasks and expected outcomes in each of the five steps are discussed below.

A. HKIX Preparation

Before the transition, the HRIX router was connected internally to two HARNET core network switches (not shown in Figure 1) at 10 Gbps each, and externally to HKIX at 4 Gbps. Since the optical hub and HKIX are hosted at the same building in CUHK, the HRIX-HKIX network connections are physically short and closely monitored by CUHK. At first, a new HRIX router was installed, with new 2 x 10 Gbps connections to HKIX. After performing ping test to verify the fiber connection, the interfaces on the new HRIX router was shut down to disable the new HRIX-HKIX connection. Connections between new HRIX and network switches of optical hubs was set up then.

At this stage, since the new HRIX router was not in used, and the old router and its connection remained unchanged, no disruption was expected.

B. HKIX Migration

The BGP neighbors of the old HRIX router was shut down. Physical cables connecting HKIX networks and HRIX router were transferred from the old router to a new one. The disabled interface on the new HRIX router were re-enabled and then the BGP neighbors.

At this stage, both the re-arrangement of cables and re-configuration of BGP neighbors result in a disconnection of the HRIX-HKIX links. We therefore expect a short disruption of traffic between the HARNET members, and Hong Kong and overseas R&D networks. The outgoing traffic should be disrupted during this disconnection but resumed shortly after the cables were transferred to the new HRIX router. The incoming traffic should be subjected to short disruption too. Moreover, before the BGP announcement through the new HRIX router reaches the remote networks, the incoming traffic should then be re-routed through other links to the HARNET. Thus, the RTT is expected to fluctuate. The traffic re-routing, if through ISP, should not cause congestion or packet loss at the HARNET(HR2)-ISP link, because the utilization during the changeover is minimal.

C. TEIN3 Migration and New ISP Backup Link Installation

The HR4 router was redundant before migration. First, the hardware of HR4 was upgraded and 10 Gbps modules added. The connection between HR3 and TEIN3 was shut down. The link was then switched to HR3. HR4 was configured with the new TEIN3 connection. HR4 was re-configured to have TEIN3 network, HR$[2|3|IX]$ as neighbours, with TEIN3 has the highest priority, followed by HR2. After that, HR4 started announcing BGP routes for the HARNET network's prefixes. New port module was inserted to HR3. IP address and BGP setting at HR3 was configured for the new connection between HR3 and HR4. Here ends the first part of the tasks solely for TEIN3 Migration.

Similar to HKIX migration above, the temporary disconnection of HARNET-TEIN3 link will result in a temporary re-routing of traffic between HARNET members and remote academic networks utilizing the HARNET-TEIN3 link. RTT fluctuations are expected, but not congestion and packet loss of the alternative routes, such as HKIX and ISP.

The second stage on this day was to add a backup connection on HR3 with new ISP's link. An IP address was set on new interface at HR3, and link from new ISP was connected to HR3. Ping test was done to verify connection. Add BGP neighbor to new ISP with lower local-preference

and longer AS-Path. A test subnet, where the HAR-hku node was connected, was announced to the new ISP instead of the old ISP.

We do not expect change of performance or disruption at all nodes except for the HAR-hku node. Since the HAR-hku's subnet was announced to new ISP instead of the old one, the incoming path was expected to change. HAR-hku should observe fluctuation of RTT and a short period of inaccessibility to remote networks utilizing TEIN3 connection.

D. CERNET Migration and New ISP Primary Link Installation

The CERNET connection at HR1 was to be relocated to HR2 in order to make rooms for connection to the new ISP. The old CERNET connection at HR1 was shut down. The CERNET link was then switched to HR2. HR2 was configured with the new connection CERNET connection. HR2 was re-configured to have CERNET network, HR$[1|4|IX]$ as neighbors, with CERNET has the highest priority, followed by HR4. New port module was inserted to HR1. Similar to the HKIX and TEIN3 migrations above, RTT fluctuations, route flip-flop and short period of inaccessibility to remote networks utilizing CERNET connection are expected.

The second stage on this day was to add the primary connection of new ISP to HR1. A new IP address was added to new interface on HR1, and link from new ISP was connected to the interface. Similar to ISP backup link configuration, BGP neighbor to new ISP was added with lower local-preference and longer AS-Path.

The new ISP primary link installation did not affect all existing connections. The lower routing preference also ensures that the link would not become effective at this stage. We therefore do not expect any special observation from all measurement nodes.

E. ISP Migration

The incoming traffic were switched to the new ISP by announcing IP prefix of HARNET to the new ISP connection. The local preference of new ISP connection was raised, and the AS-Path shortened to route outgoing traffic through the new ISP.

We expect that all measurement nodes would observe outgoing paths switched to new ISP almost simultaneously, together with RTT changes in all destinations served by the ISP link. Except for the HAR-hku node, the incoming paths to all measurement nodes may take more time to switch to the new ISP, depending on the speed of propagating the new BGP announcement to remote networks. Since the ISP link may be the only available network to carry the traffic to and from these destinations, a short period of inaccessibility could be observed.

III. Measurement Platform Setup

We have deployed a measurement prober in each of the eight universities in Hong Kong for over six years. They are connecting immediately to CR$[1|2]$ in the campus network in Figure 1. The universities are connected to HARNET for Internet connectivity, and some of them procure additional upstream providers for load-balancing and backup connection. Two additional measurement nodes, HAR-hku and HAR-cuhk, were set up at HARNET on 1 March 2015 to compare and monitor the network performance of HARNET. HAR-[hku|cuhk] connect to routers HR$[2|4]$ respectively in Figure 1.

We select around 50 web servers in Hong Kong, Asia, Oceania, Europe, and North America as destination of measurements. The measurements are divided into five groups, each of them is scheduled to run for 1 minute for every 10 minutes. Therefore, measurements are run on each node every other minute.

The measurement tool trtcp opens a pre-set number of TCP/IP connections with random client TCP ports to the remote server in each trial. In normal setting, it then sends traceroute-like request with increasing number of TTL value to trigger ICMP responses from routers to reveals the forward path. Finally, it reuses the connections to send packet-pair probes to collect network performance metrics, including RTT, and packet loss and re-ordering events on both forward and reverse paths. However, this procedure causes connection time-out in some remote servers. We use an alternative approach for such servers: performing traceroute and network performance in separate trials. However, the per-destination approach did not prevent the occasional disruption of measurements from nodes at HKIEd and LU. We ignore the results of these two nodes in our analysis.

On 5 March 2015 07:30, we doubled probing rate. We also deployed a traceroute script in 7 Planet-lab nodes on 5 March 2015. For every 3 minutes, they traceroute to 19 hosts in eight universities and HARNET. They include the web servers at the eight universities and the ten probers. The results from all measurement nodes and Planet-lab nodes were collected at a server located in one of the universities.

IV. Monitoring Results

A. General Route Dynamics and Successful Measurement

We collect the forward-path routes from all ten probers during the monitoring period and count the number of distinct IP addresses appeared in all the routes for every AS. Figure 2 plots the time-series of the percentage of counts for an AS (i.e, the number of counts for the AS divided by the total number of IP addresses in the routes). We show only the top 10 ASes in terms of the percentages in the figure.

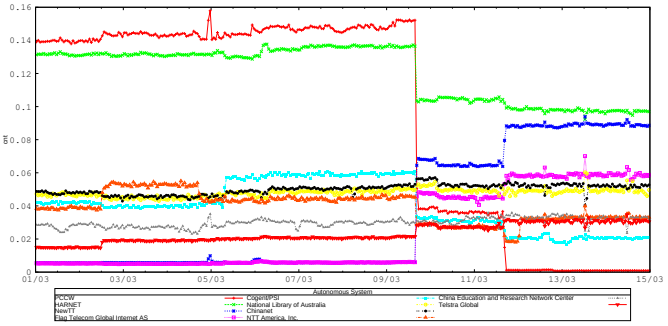


Fig. 2. Percentage of IP addresses in forward-path routes for the top ten ASes.

The most significant change takes place on 9/3/2015. The value for PCCW decreases sharply from 15% to 4%. At the

same time, the values for HARNET decreases from 14% to 10%. In contrast, the value for NEWTT increases from 0.5% to 5%. That of Flag also increases from 0.5% to 4.5%. Another sharp change, following the same trend but in a smaller scale, happens about 48 hours later. The value for PCCW decreases to negligible, and that of NEWTT increases to 9% after the second sharp change. The increases in NEWTT and decrease in PCCW on 9/3/2015 are the result of changing the ISP from PCCW to NEWTT. We subsequently discuss with CUHK's network administrator and find that CUHK switched to the new ISP on 11/3/2015 which was responsible for the second change in the IP address counts for PCCW and NEWTT.

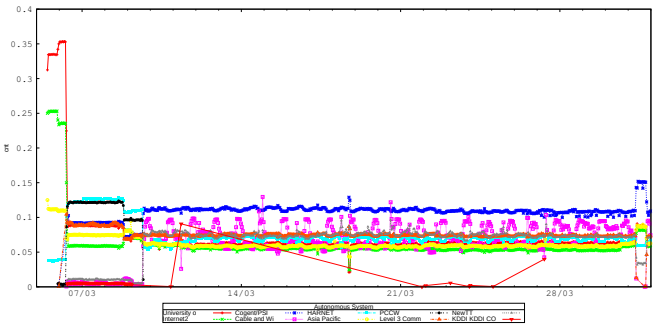


Fig. 3. Percentage of IP addresses in reverse-path routes.

We also performed reverse traceroute from seven Planet-Lab nodes to 19 hosts in HARNET. Figure 3 plots the percentage of IP addresses for each AS. The initial sharp decline for Internet2 (including University of Maryland) is not caused by route change, but by the different start times of the Planet-Lab measurement. At around 16:00 on 9/3/2015, the values were decreased for PCCW and Level3 but increased for Cable and Wireless and NewTT. Clearly this is also the result of the ISP changeover at that time. The other fluctuations are apparently not related to route changes. Some diurnal ones may come from load-balancing paths. The reverse-path route changes due to the changeover are therefore consistent with the forward path's.

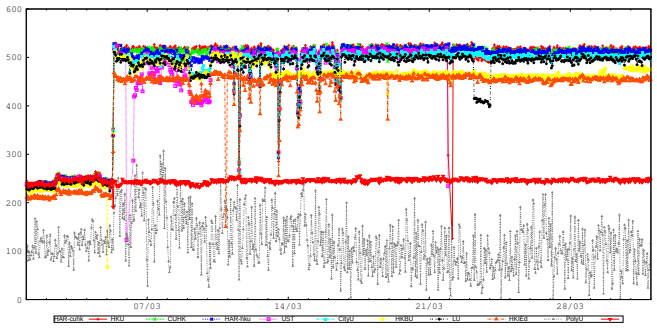


Fig. 4. Hourly Total Number of Successful Measurements for each Prober.

Figure 4 presents the time-series of the total number of successful measurement conducted by each prober. Each point is the number of successful measurement in an hour. The surges on 5/3/2015 were caused by the doubling the rate of measurement from 10 to 20 times per hour. There were two exceptions. The node at PolyU did not receive the updated measurement configurations, and the one at HKIEd failed frequently due to some other reasons.

Moreover, we observe from the one-day period on 9/3/2015 that the number of successful measurement decreases for 8 out of the 10 probers. The number of decreases was around 50 (or 10%). We found that there were a lot of unresponsive hops in the traceroute phase of trtcp, which was not completed before timeout. The performance measurement phase of trtcp thus could not be run. We conjecture that the ICMP messages induced from the routers could not reach to our measurement nodes, because of the route instability at that moment.

B. HKIX Preparation

Our measurement observes none or only short period network disruptions for the HKIX preparation during 01:00-09:30 on 4/3/2015. They are:

- PolyU-MingPao (Hong Kong, via HKIX)
- HAR-hku-TP1RC (Taiwan, via TEIN3)
- HAR-hku-NCU (Taiwan, via TEIN3)

The three failed measurements occurred at around 07:00, 10:30, and 12:20. For each of these paths, only one measurement failed during the period. There was neither forward-path route change nor TTL value change. As a result, we cannot conclude that these short disruptions are due to the HKIX preparation.

C. HKIX Migration

During the planned HKIX migration period during 22:00-22:30, there was no disruption. However, during period (1) 00:10-00:50 and (2) around 03:00 on 5/3/2015, several disruptions were reported by our measurement. They are presented in Table II with 1 and 2 to represent the two time periods.

	PCCW	MPao	TPRC	NCU	KRN	APJP	TWG
HKU			2	2			
CUHK			2	2			
PolyU			2	2			
CityU	1	1	1,2	1,2	1,2	1	
HKBU			2	2			
UST			2		2		2
HAR-hku			2	2			
HAR-cuhk			2	2			2

TABLE II. DISRUPTION OBSERVED IN HKIX MIGRATION DURING (1) 00:10-00:50 AND (2) AROUND 03:00 ON 5/3/2015

In addition, a temporary outgoing route change for several paths was observed from 22:20 to 00:00. The traffic was redirected to other networks during the HKIX connection downtime and routed back through HKIX at around 00:00. These paths can be grouped into several categories.

a) Sources from CUHK: CUHK uses dedicated connection to HKIX because of the short proximity. They were therefore not affected by the HKIX migration.

b) Hong Kong: 10 paths to 2 destinations in Hong Kong were re-routed to the secondary ISPs: PCCW ISP (6), WTT (2), NTT (1), and HGC (1).

c) TEIN3: 9 paths to 3 TEIN3 destinations (TEIN3, URENNES, BERLIN) via HKIX were re-routed to the TEIN3 paths through HR3.

d) CERNET: 7 paths to 2 CERNET destinations (PKU, SHNet) via HKIX were re-routed to the CERNET paths through HR1.

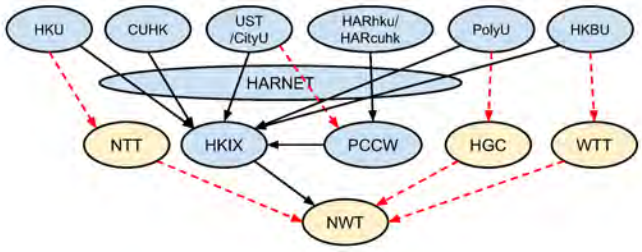
e) Internet2: 12 paths to 3 Internet2 destinations (Internet2, MIT, Stanford) were re-routed to TEIN3 (6), PCCW (5), and WTT (1).

f) Asia R&D Networks: 17 paths to 4 destinations (TP1RC, NCU, KREONET, APANJP) were re-routed to TEIN3 (9), PCCW (7), and WTT (1).

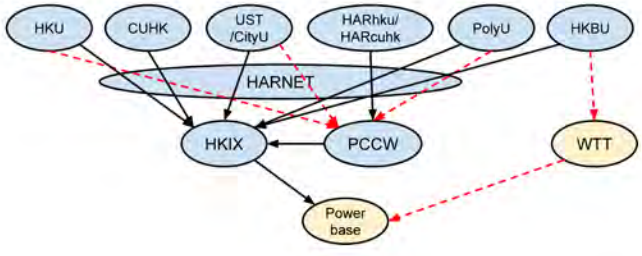
g) China: 5 paths to a destination in China were re-routed to PCCWs during the period.

Figure 5(a) illustrates the forward paths from all nodes to MingPao, one of the destinations in Hong Kong that can be accessed through HKIX. The solid lines represent the primary outgoing paths and dash lines the alternative paths during the period. As mentioned above, CUHK did not observe route change due to its close proximity with HKIX. HAR-hku and HAR-cuhk did not have access to HKIX, and they accessed MingPao through HARNET's old ISP (PCCW) that had no interruption during the period. HKU, PolyU, and HKBU accessed MingPao through their secondary ISPs (NTT, HGC, and WTT), respectively. UST and CityU, on the other hand, opted HARNET's ISP as the second best path and accessed it through the ISP link. Figure 5(b) shows the forward path from all nodes to a PCCW destination in Hong Kong. The original paths are the same as that for MingPao. But for the alternative paths, HKU and PolyU accessed it through PCCW instead of their own secondary ISPs as in MingPao's case.

Figure 6 shows the RTT from CityU to two destinations in Hong Kong. It clearly shows that the RTT increased during the temporary route change between 22:00 and 00:00, and both RTT and forward route reverted back at 00:00. Immediately after that, the measurement failed for 40 minutes since 00:10. This 40-minute outage occurred only between CityU and HKIX destinations. It should be caused by the routing configuration or connection problem between CityU CR1 and the HRIX routers.

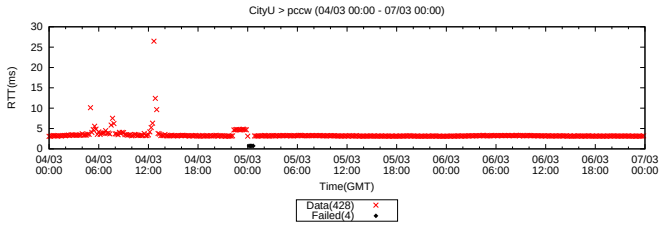


(a) Mingpao

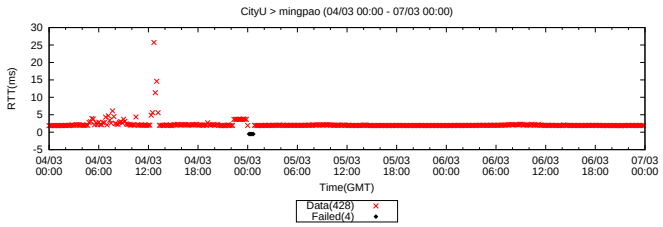


(b) PCCW

Fig. 5. Primary and Secondary Forward Routes to MingPao and PCCW During the HKIX Migration.



(a) PCCW



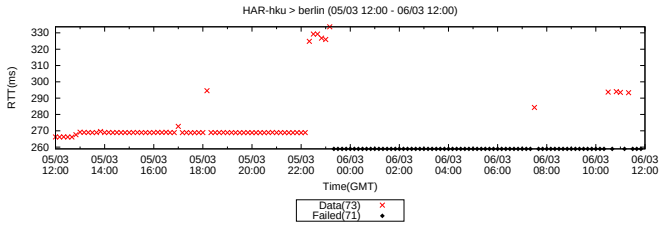
(b) MingPao

Fig. 6. RTT measurements from CityU to two Hong Kong Destinations Through HKIX.

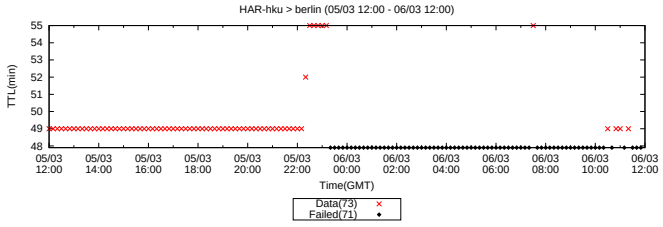
D. TEIN3 Migration

Our measurement observes temporary re-routing during the TEIN3 connection migration on 5/3/2015. Traffic going through TEIN3 link via HR3 was routed to other links at around 22:25 on 5/3/2015. After around 4 hours 50 minutes, the traffic was routed back to the TEIN3 link at around 03:15 on 6/3/2015, via HR3 and HR4.

Despite some measurement failures, we still observe 12 paths subjected to short disruptions during the transition period, and 13 paths subjected to longer disruptions. The 13 paths were all sourced from HAR-hku and to destinations through the ISP and TEIN3 networks. The disruption period was around 6 hours 50 minutes, between 23:20 on 5/3/2015 and 07:10 on 6/3/2015. Figure 7 presents the RTT and TTL values for one of the 13 paths, HAR-hku-BERLIN. The initial change of TTL value in Figure 7(b) coincides with the re-routing of forward path at around 22:25 on 5/3/2015. It signals the disconnection of the TEIN3 network, and re-routing of outgoing and incoming traffic through ISP via backup links. The disruption at around 23:20 on 5/3/2015 was probably caused by the disconnection of the backup link to ISP network. The backup link was briefly resumed at around 07:30 on 6/3/2015.



(a) RTT



(b) TTL

Fig. 7. RTT and TTL of the HARhku-BERLIN Measurement During TEIN3 Migration.

E. CERNET Migration

The CERNET migration was done similar to the TEIN3 migration but with less paths affected. Table III presents the route change observed between 22:30 on 6/3/2015 and 00:50 on 7/3/2015, which lasted for 4 hours 20 minutes.

	PKU		SHNet	
HKU	HKIX	-	HKIX	-
CUHK	CERN	-	CERN	-
PolyU	HKIX	-	HKIX	-
CityU	CERN	HKIX	CERN	HKIX
HKBU	-	-	CERN	TEIN3
UST	HKIX	-	HKIX	-
HAR-hku	-	-	TEIN3	-
HAR-cuhk	CERN	ISP	CERN	ISP

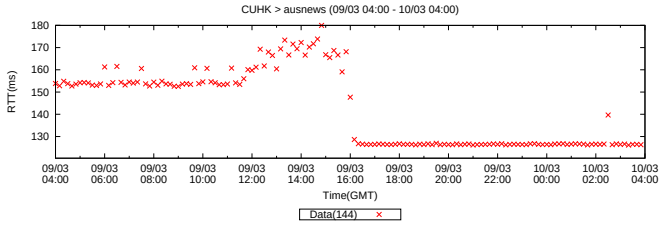
TABLE III. ROUTE CHANGE OBSERVED IN CERNET MIGRATION BETWEEN 22:20 ON 6/3/2015 & 00:00 ON 7/3/2015.

Except for paths experiencing measurement disruptions before the transition, only short disruptions were observed in four paths. Similar to the HAR-hku node for TEIN3 paths, the paths sourced from HAR-cuhk node to destinations through CERNET were re-routed to ISP during the transition period. Figure 8 presents the RTT and TTL values of HAR-cuhk-PKU path. The temporary change of RTT and TTL between 22:30 on 6/3/2015 and 00:50 on 7/3/2015 coincides with the forward-route change.

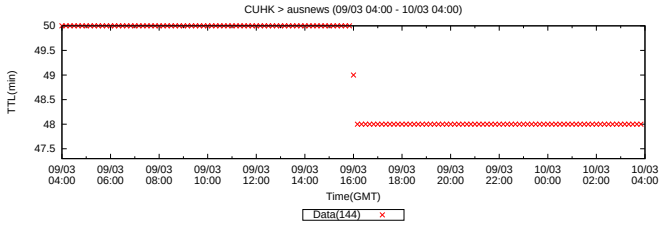
F. ISP Link Installation and Migration

The ISP link installations were done along with the TEIN3 and CERNET migrations. They were not utilized by the HARNET members until the migration. We did not observe relevant disruption or performance change.

The ISP migration, which was planned to take place at 16:00 on 9/3/2015, were actually performed at around 16:10 according to the forward route change observed by us. No disruption were observed around the transition time. Figure 8 presents the RTT and TTL of the CUHK-ausnews path. The permanent change of RTT and TTL at 16:10 on 9/3/2015 coincides with the forward route change observed.



(a) RTT



(b) TTL

Fig. 8. RTT and TTL of CUHK-ausnews Measurement During the ISP Migration.

V. RELATED WORK

Although there is no previous work on studying the impact of ISP changeover, Wang et al. conducted active measurement on controlled route changes and end-to-end network performance [6]. They found that routing changes contribute to end-to-end packet loss significantly. Huang et al. used routing information to detect and identify network disruptions [5]. They proposed a multivariate analysis technique on dynamic routing information to detect node and link disruptions.

VI. CONCLUSION

In this paper we presented the detailed ISP changeover plan and the measurement result during the migration. We have set up a measurement platform to monitor route changes and end-to-end performance change during the entire changeover period. We reported the impact of the changeover in each major step of the changeover. Except for short intervals of instability and disruptions, the ISP changeover is considered quite smooth.

REFERENCES

[1] Avoid Pitfalls When Changing Your Business' ISP. http://www.3dcorp.us/blog/articles/avoid-pitfalls-when-changing-your-business-isp/.

[2] HARNET - The Hong Kong Academic and Research NETwork. http://www.jucc.edu.hk/jucc/harnet.html.

[3] E. Chan, X. Luo, R. Chang, W. Fok, and W. Li. Non-cooperative diagnosis of submarine cable faults. In *Proc. PAM*, 2011.

[4] W. Fok, X. Luo, R. Mok, W. Li, Y. Liu, E. Chan, and R. Chang. Monoscope: Automated network faults diagnosis based on active measurements. In *Proc. IFIP/IEEE IM*, May 2013.

[5] Y. Huang, N. Feamster, A. Lakhina, and J. J. Xu. Diagnosing network disruptions with network-wide analysis. In *Proc. ACM SIGMETRICS*, 2007.

[6] F. Wang, Z. M. Mao, J. Wang, L. Gao, and R. Bush. A measurement study on the impact of routing events on end-to-end internet path performance. In *Proc. ACM SIGCOMM*, 2006.

Netflow realtime query and ELK based analyzer on TWAREN

Ming-Chang Liang, Jiunn-Jye Chen, and Li-Chi Ku.

***Abstract*—This paper aims at presenting the TWAREN NOC team's work on developing a realtime Netflow processor and an ELK-based Netflow analyzer. Written in C language, the realtime Netflow processor is able to collect and calculate the IP and link statistics from the huge amount of the realtime TWAREN Netflow data by using only one server. Its performance has been verified to be able to handle the peak data volume during heavy malicious attacks. Furthermore, its ability to sort the results and respond in seconds makes it a perfect tool to deal with frequent queries and serves as a reliable monitor tool to periodically detect incidents of network abuse.**

Meanwhile, an ELK based Netflow analyzer has been developed to discover deeper insight out of the Netflow data. Yet powerful, the ELK based solution is susceptible to certain performance bottlenecks, such as large index imports and the poor garbage collection induced cluster chain crash. This paper introduces the details of these problems and the solutions we have found.

***Index Terms*—TWAREN, Netflow, ELK, ElasticSearch, LogStash, Kibana**

I. Introduction

Due to the increasing concern of the network security, TWAREN[1] NOC has played an more important role on mitigating network abuses and malicious attacks. Without the rights to touch the payload of the packets, the Netflow data and flow statistics become the main tool to detect network malicious behaviors. Although we already have daily and monthly digests from the Netflow data, they are never realtime enough to respond to ongoing threats in time. Thus realtime Netflow processor and analyzer become the goal of our development.

Because of the lack of computational resources, in the beginning, the rule of thumb of our development is to make it efficient enough to process the Netflow data from the whole TWAREN backbone with minimal computational costs. To achieve such a high efficiency, we used C language to implement the Netflow processor. It turned out to be capable of extracting the usage data and correlating them with IP and link information much faster than the speed of the Netflow data generation. Thus it is able to handle the burst of Netflow data during the network attacks. Because it can complete the query and result sorting in a very computational efficient manner and respond in seconds, it is quite suitable for periodically and automatically detecting network abnormal behavoir. The design and implement of this program will be introduced in this paper.

To further provide more useful features, we have developed a Netflow analyzer based on the open source Elastic Stack (ELK)[2]. ELK has a very rich feature set, including online realtime indexing and query, distributed storage, a nice graphic user interface and dashboard. These features save the developers a lot of time when compared to building applications on Hadoop. This paper will introduce the design concept of this application, the problems we have faced and the solution we have found.

2016/05/27.

Ming-Chang Liang is with the National Center for High-Performance Computing, No. 28, Nan-Ke 3rd Rd., Hsin-Shi Dist., Tainan City, Taiwan (e-mail: liangmc@narlabs.org.tw).

Jiunn-Jye Chen is with the National Center for High-Performance Computing, No. 28, Nan-Ke 3rd Rd., Hsin-Shi Dist., Tainan City, Taiwan (e-mail: jjchen@nchc.narl.org.tw).

Li-Chi Ku is with the National Center for High-Performance Computing, No. 7, R&D 6th Rd., Hsinchu Science Park, Hsinchu City, Taiwan(e-mail: lku@narlabs.org.tw).

II. Design and Implementation

This chapter describes the architecture, the implementation and the result of the development of the realtime Netflow processor and the ELK based Netflow analyzer cluster.

A. The realtime Netflow processor

When developing a high performance application, the choosing of programming language is oftem a dillema of pursuing high speed or easy maintenance. Therefore we chose to divide the application into multiple programs and modules, each being programmed in a different language that fits best, as illustrated in Fig. 1. For the FLMd and FLQd modules which need to be extremely fast, C language has been used. For user interface and other modules, the languages were chosen by each programmer in charge. It works well as long as the interface and protocol between modules have been clearly defined. This application runs on a Intel Xeon E5-2630 dual CPU machine with 128GB Ram.

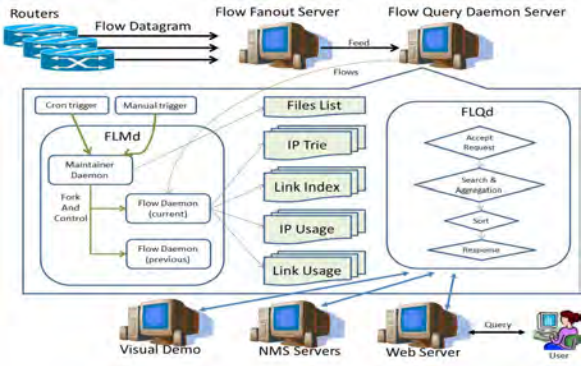


Fig. 1. The realtime Netflow processor architecture

The maintainer module of the FLMd program is responsible for managing the flow-daemon. Once triggered by the SigUsr1 external signal, the maintainer module spawns new flow-daemons to take over the job of receiving Netflow datagrams. Meanwhile, it notifys the old flow-daemons to stop receiving, finish processing its queue and write the resulting statistics of IP trie, link index, IP usage table and link usage table to a disk file. Once completed, the old flow-daemons gracefully terminate. The disk file is named by the maintainer module and handed over to the new flow-daemons. An overall file list is kept in a files-list and maintained by the maintainer module. When periodically triggered by a cron job, it consistently delivers the most recent Netflow statistics of the duration of wish.

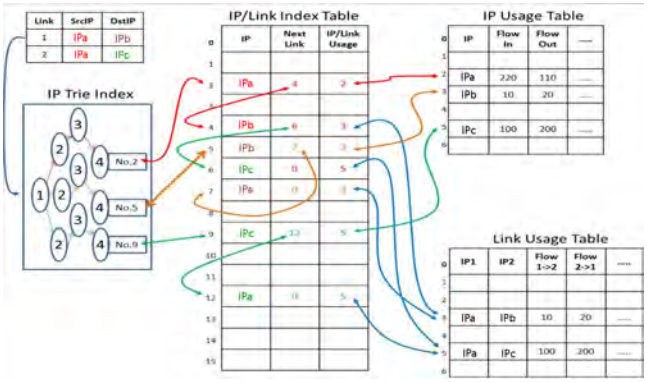


Fig. 2. The building of the IP/Link index table

The flow-daemon extracts the necessary informtaion from the Netflow datagrams and builds the IP/Link index table, as shown in the figure 2. The first step is to build the SrcIP index. The SrcIP of each flow is checked against the IP Trie to lookup the corresponding index of this particular SrcIP in the IP/Link index table. If no matching is found, a new row is appended to the table. For example, through the IP Trie, the IPA in the figure 2 would find a corresponding entry of no. 2 in the IP/Link index table, which further points to the corresponding entry in the IP usage table. A new row will be added to the IP usage table and linked in the IP/Link index table if it doesn't exist. The IP usage table counts the flows, packets and octets of every presented IP.

The second step is to build the link chain of the given SrcIP. If the corresponding "Next Link" field of the SrcIP in the IP/Link index table is 0, it indicates that this link is not in the record and needs to be added. In this case, a new row of the DstIP is appended to the IP/Link index table and becomes the "Next Link" of the SrcIP row. A corresponding row is also appended to the Link usage table and linked in the "IP/Link Usage" field of the row in the IP/Link index table. Similar to the IP usage table, the Link usage table counts the flows, packets and octets of the given link (SrcIP-DstIP pair). The two directions of the link are counted in separate fields.

The third step is to add new row for the DstIP in the IP usage table in the same way as described above. Since each row in the Link usage table counts the information for both directions, the corresponding row is reused and linked in the newly added DstIP row in the IP/Link index table.

In our observation, when processing the Netflow data from all the network equipments on the TWAREN backbone, the size of the aforementioned three tables seldom grows over 1 GB in 10 minutes interval. Several DDos attacks have been witnessed on TWAREN recently. Even during such extreme events, the table size only reaches 64 to 128 GB in 5 minutes interval. Therefore with moden machines, all the working files can be easily put in RAM file systems, such as the tmpfs file system, or SSDs to ensure a high performance operation.

As explained above, every time when the maintainer module receives a SigUsr1, all the aforementioned data get flushed into the disk. However, in order to make the future query operation efficient, the IP/Link index table will be sorted first before getting write to the disk. During the sorting, the entries of this table is sorted according to the IP sequence of the IP usage table, with all the linking rows pooled in consecutive order and number of link counted and written to the IP usage table. Thus when the information of certain IP is queried in the future, the beginning position and the number of the records can be easily obtained and all of them can be loaded into RAM as an array in a single read.

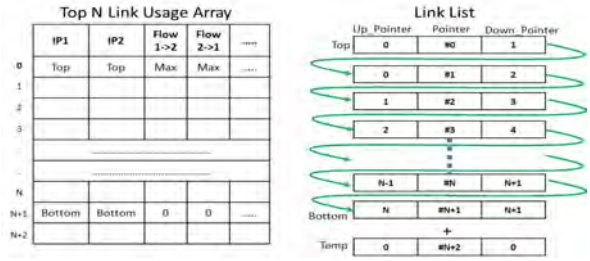


Fig. 3. The building of the IP/Link index table

FLQd is the daemon providing the query service to the user interface. NOC staff often needs to know the usage and the links related to a certain IP. In such query, through the IP Trie, the comparison only needs to be performed over four nodes for IPv4, or sixteen nodes for IPv6, to lookup the position of the starting row of this IP in the IP/Link index table. From this table we know which rows to go in the IP usage table and the Link usage table. From the latter two tables, all the necessary information is available.

The result of the query must be sorted to be useful. A top-N query implys to find the first N records in the sorted list. A trivial sorting over the 10 minute interval data, which often

contains millions of records, is obviously impractical. We used a modified insertion sort as depicted in the Fig 3.

Since only the top-N are relavent, the link list for sorting is of the size of N+1, with each item pointing to the corresponding row of the Link usage array. Each record of the usage table being queried will firstly be compared with the N+1 item, which is called the temp item. If it is smaller, it is discarded. Once it is larger, the comparison continues with item N and so on. The comparison stops when it is smaller or equal to the next item, or when it reaches the top of the list. Then the new record settles at its current position, pushing all the items below it one position down. The item N becomes the temp item and the original temp item is discarded.

After a few iterations, the temp item will be large enough to defeat most incoming records. Since the value of N is usually far smaller than the number of the query result, which is often some millions, the computational complexity to do the insertion sort of the N items is small enough to be ignored. Thus the overall complexity of finding the top-N is nealy O(n), with n being the size of the query result. This method is extremely memory efficient since it eliminates the need to allocate a huge memory space enough to hold the whole dataset. Instead, only a tiny amount of memory enough for the top-N table and link list is necessary.

In practice, the query and sorting complete in one to two seconds, and most of it is spent on the disk I/O. To further reduce the I/O time, using memory mapped technique is recommended. Fig. 4 shows an example of the query result through the web interface.

Fig. 4. Query result of the realtime Netflow processor

B. The development of the ELK Cluster

Although the realtime Netflow processor is very fast and resource efficient, it is not flexible enough to support complex and combined query, which could be useful in some occasion. Eventually the Elastic Stack (ELK) came into our horizon. It's Kibana Dashboard, as an example, provides a feature rich UI that it can save us a lot of time to develop something our own.

As the ELK officially suggested, the heap size of each JVM should be kept under 32 GB. In addition, half of the system memory should be reserved for the Lucene search engine as I/O buffers[3]. Following these guidelines, 64 DataNodes have been created, each with a hard disk and a 31 GB heap, as shown in Fig. 5..

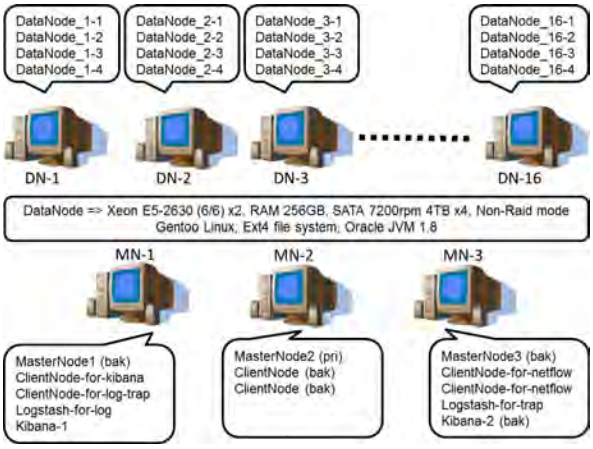


Fig. 5. The architecture of the ELK Cluster

3 additional machines are used to each hosts a MasterNode and several ClinetNodes. Several LogStash and Kibana programs are also run on these machines. All of the TWAREN Netflow data has been sent to the Logstash Input::Netflow codec, and then outputs to the Elasticsearch cluster (ES-Cluster) by the Output::Elasticsearch.

C. The ELK online query and visualizationr

Through the Kibana interface, the Netflow types of interests are selected and then an aggregated query is sent to the Elasticsearch engine to obtain the sorted statistics as a pie chart. Fig. 6-9 show the pie charts of the top 100 usage of Netflow dstaddr, srcaddr, dstport and src port (as highlighted in the ellipse region), respectively, over 15 minutes interval on TWAREN. In this way, we can easily inspect the top IP and port usage on the backbone, in any time span of interests.

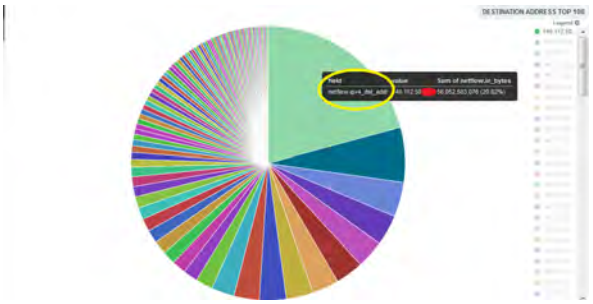


Fig. 6. Top 100 pie chart of destination IP

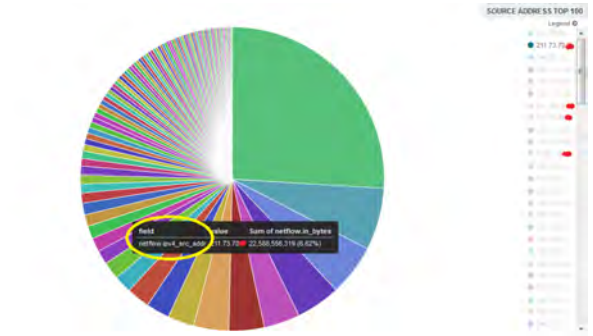


Fig. 7. Top 100 pie chart of source IP

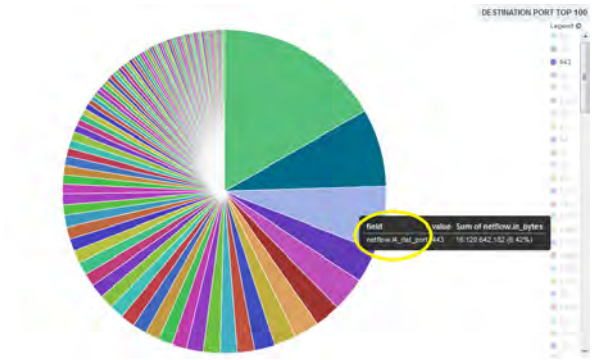


Fig. 8. Top 100 pie chart of destination port

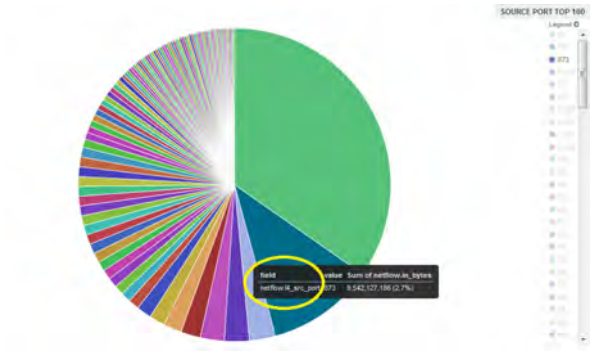


Fig. 9. Top 100 pie chart of source port

Kibana is even capable of producing multiple layers concentric pie charts when making multiple factors querys. Fig. 10. shows the top 10 dstaddr IPs in the inner pie chart, with top 10 dstports of each IP shown in the outer pie chart, directly above their respective IPs. Fig. 11. presents the same data in opposite order, with top 10 dstports in the inner pie chart and top 10 dstaddr in the outer pie chart. Likewise, if we are interested in who generates the most traffic, a query of dstaddr+srcaddr delivers just that, as shown in Fig. 12. This outstanding capability of Kibana provides deeper insights over the network.

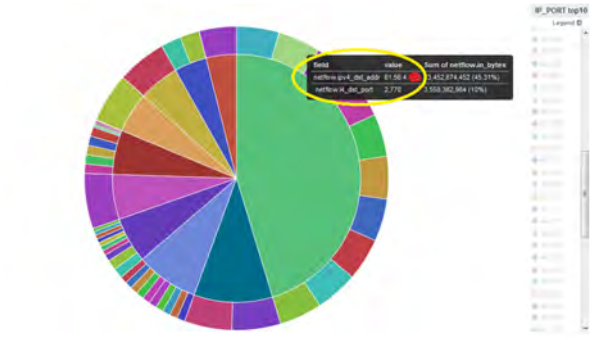


Fig. 10. Top 10 pie chart of IP + port

Fig. 13. shows the top 5 source IPs of packet counts of each 30 seconds intervals. This type of chart draws the statistics against the time axis, which is very useful to detect the suspicious IPs which transmit numerous small packets in short time bursts, a typical trait of DDoS attacks.

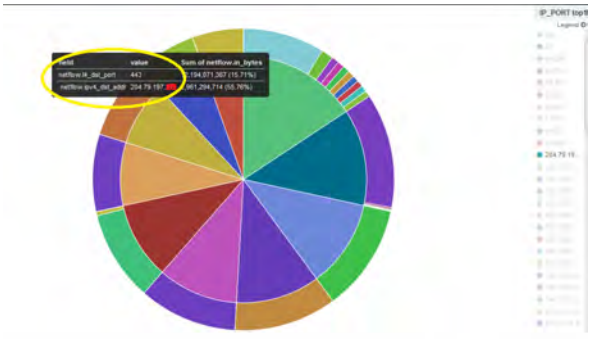


Fig. 11. Top 10 pie chart of port + IP

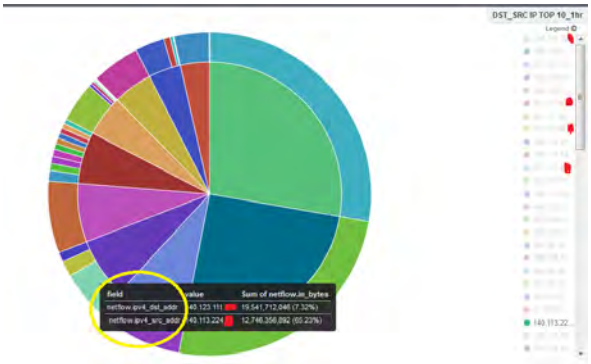


Fig. 12. Top 10 pie chart of Dst IP + Src IP

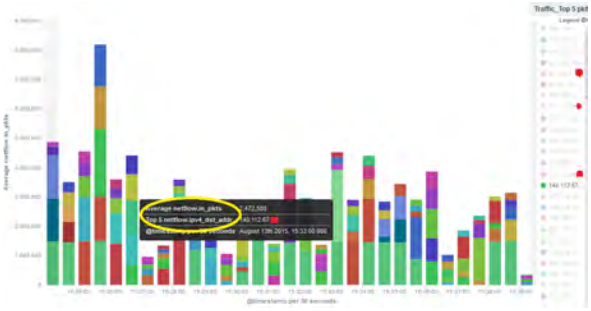


Fig. 13. Top 5 IPs of packet counts in each time interval

The query performance comparison against the number and type of sorting requests is summarized in the Table I. Even with two kinds of average computation and a top 10 IP sorting, the response time to the query remains consistently low. When an additional sorting request of top 5 ports is added, the response time becomes 60 times higher. This indicates that doing two independent sorts over two different fields of value increases the loading significantly. However in practice, this level of performance over the typical 40 million flow records is still considered useful.

TABLE I.
PERFORMANCE COMPARISON UNDER DIFFERENT SORTING AGGREGATION

Sorting Aggregation	Response Time (ms)	Number of Data
Bits	597	40297487
Bits + Packet #	1253	40298536
Bits + Packet # + Top 10 IP	1370	40269181
Bits + Packet # + Top 10 IP + Top 5 Port	72551	40828719

III. THE PROBLEMS WHEN IMPLEMENTING ELK

A. The performance bottleneck on Logstash

The TWAREN backbone generates Netflow data in the speed of 10,000-30,000 flows per second (fps) during regular hours, 40,000-50,000 fps during rush hours and over 100,000 fps when under DDoS attacks. Initially we used 5 Logstash and 5 ClientNodes to receive the Netflow data, with the Logstash configured as Input::UDP::Netflow codec (24 threads) and Output::Elasticsearch (24 threads). To our surprise, only 6,000-7,000 fps got through. All the rest were discarded due to insufficient performance. All our trials to adjust the output flush_size, idle_flush_time and the Elasticsearch (ES) index.refresh_interval resulted in vain. In order not to get records discarded, we changed it into using TCP (Input::TCP+Filter::Grok&Date), it turned out that the 5 minutes Netflow data (5,338,416 records in total, 17,795 fps) took 58 minutes to complete to output to ES. Obviously it is not realtime competent.

After installing Logstash onto all 19 machines, the performance barely increased to 40,000 fps, which is still far from enough. Meanwhile, those Logstash generated a very high CPU loading on every machine, severely hindering the Elasticsearch from working properly. Compared to our own FLMd program which can handle more than 100,000 fps with single thread on a single machine, the performance of Logstash is disappointing. Therefore we have to abandom it eventually in favor of our own program. The FLMd has been modified to convert the Netflow into json format. Every 100,000 records or 10 seconds, whichever reaches first, will trigger FLMd to fork a new thread to deliver the data to the ES-Cluster via the HTTP Bulk API. The resulting performance is 185,856 fps, much more than necessary for TWAREN.

B. The JVM HEAP Garbage Collection Problem

The DataNode (DN) of Elasticsearch compiles the data of a single day into an index. In our case, each index is roughly 768 GB in size. Since we only have 28 DNs and each DN holds a shard, the index can only be splitted into 28 shards of 14 GB each.

Every time when Elasticsearch processes an aggregation query, such as a top-N query, it needs to build very large field-data arrays for each field involved, and big request-data arrays for metadata and the results. Elasticsearch will try to keep them in the memory for reuse to avoid rebuilding them every time. That usually takes 60-75% of the heap. For a 31 GB heap, this means there is only 9 GB free. Obviously, when a new aggregated query is issued, which involves other 14 GB shards and more arrays, the memory will be insufficient. This triggers a large amount of garbage collection (GC). If the memory is still not enough after purging the young generation data, it triggers the old generation GC, which causes the "world stop" of JVM -- JVM stops execution and solely focuses on GC. During this time, the DNs involved will be freezed, unable to respond to the alive-check from the MasterNode and finally being kicked out of the cluster.

By utilizing the doc_value attribute of the number-type field, Elasticsearch will build an additional array for the number-type field in the shards. This array can be read from the shards on demand, eliminating the need to build and retain those arrays in the memory, thus greatly reducing the pressure on the heap. By carefully tuning the field attribute to exclude those fields that never need to be query and sort, the index and the memory consumption can be also reduced. By using these technique, the GC problem has been greatly minimized.

C. Elasticsearch cluster crash

When the aforementioned freezed DN comes back alive and re-join, MasterNode (MN) will assign the same task to it. Facing the same problem, it gets freezed again. The same situation repeats until the re-join retry count is exhaustet and the DN is forced to restart. This phenomenon eventually drives most of the DN offline, making the cluster useless.

Elasticsearch includes a circuit-braker mechanism for this purpose. When enabled, DN will utilize the metadata in the shards to estimate the necessary array space before actually loading and building the field-data array in Ram. If the estimated space exceeds a certain threshold (in our case, 40% heap space), the query will be refused. This mechanism helps mitigating the GC problem to cause the cluster to crash, but it also introduces a problem that some complex aggregated query may be interrupted and fail.

IV. CONCLUSION

Although ELK stack is really powerful and flexible in making complex query, it easily suffers from the memory problem. Its nature of retaining old metadata arrays in the heap and the fast incoming of the TWAREN Netflow data somehow exaggerate the problem. To make it stable and usable under complex aggregated query, the cost of RAM investment will be propotional to the overall data size. In our experience, our 14 data servers with 1,792 GB Ram in total are still not enough to perform a top-N query over a whole month of Netflow data.

In contrast, our realtime Netflow analyzer is extremely resource efficient but lacks the ability to make complex aggregated query. Thus strategically it would be most beneficial to use our own program in the automatic system to detect possible incidents. Once a suspicious incident is detected, the operators can further investigate the problem in depth by using the ELK based Netflow analyzer.

REFERENCES

[1] ***TaiWan Advanced Research and Education Network (TWAREN)***. Available: http://www.twaren.net/

[2] ***Elastic Stack (ELK): Elasticsearch, Logstash, Kibana, and Beats***. Available: https://www.elastic.co/

[3] Limiting Memory Usage, ***Elasticsearch Guide***. Available: https://www.elastic.co/guide/en/elasticsearch/guide/current/_limiting_memory_usage.html

A Study about Web Application Inter-Cloud Auto-Scaling

Yuko KAMIYA, Toshihiko SHIMOKAWA

***Abstract*— Cloud Computing System is a basic infrastructure of current ICT systems. We propose "Meta-Cloud Computing". It makes it possible to use multiple Cloud Computing System as a single virtual Cloud Computing System. We develop a prototype system of Meta-Cloud Computing System. In this paper, we discuss about Meta-Cloud friendly server system configurations.**

***Index Terms*— IaaS, Cloud Computing, Selection Policy**

I. INTRODUCTION

CLOUD Computing System, especially, IaaS, is a basic infrastructure of current ICT systems. Large number of systems are constructed on Cloud Computing System.

One of important feature of Cloud Computing System is Auto-scaling. It automatically launches or terminates virtual machines inside a Cloud Computing System based on pre-defined users' policies. It scales computational capability of servers depending upon actual workload.

There are large number of Cloud Computing Systems on the Internet. For example, EC2 of Amazon Web Services, Softlayer by IBM, Google Cloud Platform, and so on. These are so called "public cloud". On the other hand, users can build their own cloud so called "private cloud" by Open Source Software, ex. OpenStack, CloudStack, and so on.

There are some problems to construct system on Cloud Computing System. One of them is users have to select a base Cloud Computing System for their system, because there are very many Cloud Computing System as we mentioned above. The other one is network bandwidth for their system. Auto-Scaling can increase their computational capacity. However, it is difficult to increase network bandwidth. Auto-Scaling can increase virtual machine inside a Cloud Computing System. Even in that case, external bandwidth is not increased, because all of their virtual machines are inside same Cloud Computing System.

As We mentioned above, users have to select Cloud Computing System. To solve this, we propose "Meta-Cloud Computing" [1,2]. It makes it possible to use multiple Cloud Computing Systems as a single virtual Cloud Computing System. And we developed a prototype system of Meta-Cloud Computing System to assess the feasibility of Inter-Cloud Auto-Scaling. We call it "Soarin".

Many services are provided based on Web System. One of typical system architecture is Web 3-tier model. In this paper, we discuss about Meta-Cloud friendly system configurations.

This work was supported by JSPS KAKENHI Grant Number JP26330124.

Yuko KAMIYA and Toshihiko SHIMOKAWA are with the Kyushu Sangyo University, Fukuoka, JAPAN (e-mail: kamiya@is.kyusan-u.ac.jp, toshi@is.kyusan-u.ac.jp)

II. META-CLOUD COMPUTING

One of key feature of Meta-Cloud Computing System is "Inter-Cloud Auto-Scaling". This is natural extension of Auto-Scaling. Usual Auto-Scaling launches new virtual machines inside a Cloud Computing System. "Inter-Cloud Auto-Scaling" extends it beyond multiple Cloud Computing Systems. It launches new virtual outside of origin Cloud Computing System. Therefore, users who construct their system on Cloud Computing System does not have to select Cloud Computing System. "Meta-Cloud Computing System" automatically select a suitable Cloud Computing System for new virtual machine. In this paper, we call Auto-Scaling inside a single Cloud "Single Cloud Auto-Scaling" to distinguish with "Inter-Cloud Auto-Scaling".

"Inter-Cloud Auto-Scaling" has the other advantage. It can increase external bandwidth. As we mentioned above, it can increase virtual machine on the other Cloud Computing System. Therefore, it can use another external network between Cloud Computing System and the Internet. And also, it may decrease Round Trip Time between servers and clients.

A. Soarin: Prototype System of Meta-Cloud Computing

As we mentioned above, we developed a prototype system of Meta-Cloud Computing System "Soarin".

To realize Inter-Cloud Auto-Scaling, Soarin has to be able to launch same virtual machines on multiple Cloud Computing Systems. If there are shared storage system, it is easy. Because, new virtual machine is able to launch from same HDD image as original virtual machine. However, this assumption is not correct on Inter-Cloud Auto-Scaling. To solve this problem, Soarin use configuration management tool to launch same virtual machine on different Cloud Computing System.

Many of servers are based on existing major Linux distribution, ex. Red Hat Linux, CentOS, Ubuntu, and so on. And, the servers use major server applications, ex. Apache, nginx, PHP, MySQL, and so on. Finally, their own application programs are installed on the server. On the other hand, almost

all Cloud Computing System has HDD image of unmodified major Linux distributions. Soarin launches virtual machine using these unmodified Linux distribution image. After that, Soarin uses configuration management tool to build up server on the virtual machine.

Soarin uses fog[3] to control multiple different Cloud Computing Systems. Fog is called "cloud services library" based on Ruby. Fog provides generalized abstraction to Cloud Computing Systems. Current fog support very many Cloud Computing Systems, ex. Amazon AWS, IBM Softlayer, OpenStack and so on. Therefore, Soarin can use these Cloud Computing Systems.

B. Cloud Selection Policies

Both of "Single Cloud Auto-Scaling" and "Inter-Cloud Auto-Scaling" launch or terminate virtual machines based on pre-defined users' policies. These policies contain metric and threshold to determine WHEN it launches or terminates virtual machines. In addition, policies of "Inter-Cloud Auto-Scaling" contains metric and threshold to determine WHICH CLOUD it launches or terminates virtual machines.

We call the policy "Cloud Selection Policy". We implemented some basic Cloud Selection Policy as follows.

- Neighborhood of users
  RTT between Servers and Clients are important on interactive applications. This policy collects IP addresses of clients. It selects a Cloud Computing System depending on number of clients of each networks. Current implementation uses NetFlow [3] to collects clients' information to reduce workload.
- Minimum resource usage
  Many of public Cloud Computing System provide very large computing resources. However, most of private cloud provide restricted computing resources. As we mentioned above, "Meta-Cloud Computing"
- Neighborhood of origin cloud
  Contents synchronization is one of important issue of distributed server. There are large number of solutions. This policy uses simple solution to reduce synchronization cost. It uses network distance between origin Cloud Computing System and each Cloud Computing Systems.
- Round Robin
  This is very simple selection policy. It selects a Cloud Computing System by round robin algorithm.

III. CONSIDERATION OF INTER-CLOUD AUTO-SCALING

In this section, we discuss about Inter-Cloud Auto-Scaling. First, we show some typical web based system architecture.

A. System configuration of some services

1) General web system architecture (Web 3-tier model)

Fig. 1. shows general web system architecture (Web 3-tier model) including web server, application server, and database server. Auto-Scaling feature may launch new web server and/or application server to serve users' requests. Local load balancer spreads these requests. In some cases, for example, the service many clients access database servers frequently or disasters happens, new database server is launched.

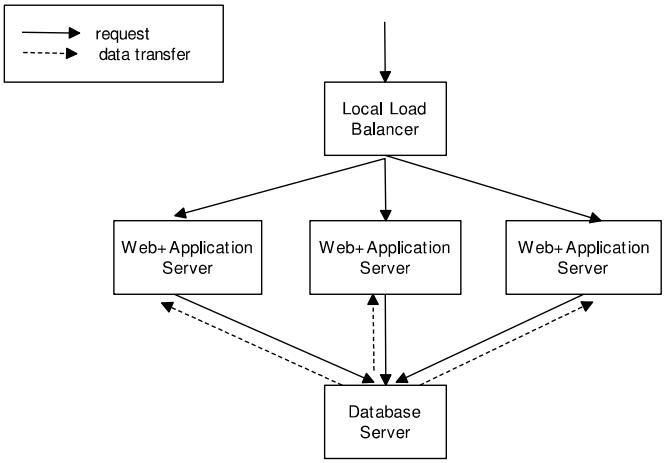


Fig. 1. General Web System Architecture (Web 3-tier model)

2) Video Streaming

Fig. 2. and Fig. 3. shows general video streaming architectures. There are two types of video streaming. Fig. 2. shows live streaming system. To deliver live streaming, encode server (Encoder) encodes a movie from video camera in real time. Streaming Server gets video data from encoder and delivers for users. Fig. 3. shows video on demand system. In this case, video data are prepared in advance. Therefore, real time encoding is not necessary.

Auto-Scaling feature may launch new streaming servers. In the case of live video streaming, contents synchronization is easy. On the other hand, in the case of video on demand, contents synchronization is big problem.

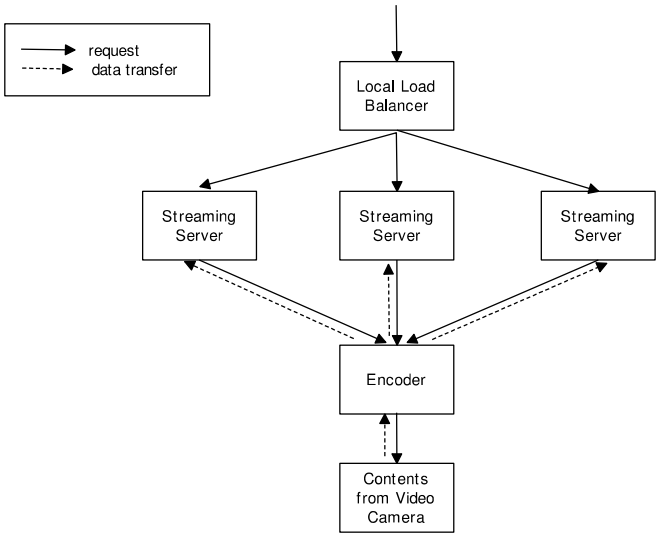


Fig. 2. Live Video Streaming System Architecture

B. Inter-Cloud Auto-Scaling issue

1) General web system architecture (Web 3-tier model)

Inter-Cloud Auto-Scaling in the case of general web system architecture is useful. It can deploy new web servers in the Origin Cloud or in the different Cloud Computing System. If the different Cloud Computing System is chosen, the RTT between web servers and clients may become short compared with in the Origin Cloud. However, the RTT between servers and database servers should become long compared with in the Origin Cloud. To reduce the RTT, Inter-Cloud Auto-Scaling can deploy new database on the new Cloud Computing System. However, new problem: wide area database synchronization is arisen. RTT between clients and servers and database synchronization is tradeoff. It depends on services on the servers.

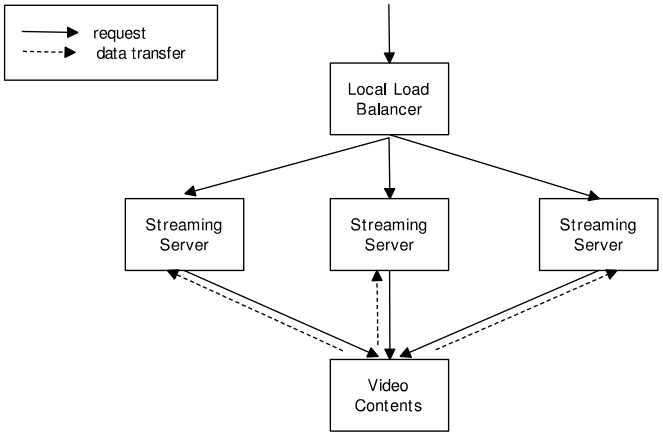


Fig. 3. Video on Demand System Architecture

Session of web services also becomes problem. We will discuss it later at section C. Load Balancer.

2) Video Streaming

Inter-Cloud Auto-Scaling in the case of the live video streaming system is also useful. It can deploy new streaming servers in the Origin Cloud or in the different Cloud Computing System. In this case, contents synchronization is easy as we mentioned above. Therefore, it is easy to launch new streaming server on the other Cloud Computing System. We consider that Inter-Cloud Auto-Scaling is very useful for this service.

On the other hand, Inter-Cloud Auto-Scaling in the case of the Video on Demand have to consider about contents synchronization. Generally speaking, contents of video on demand is large. Therefore, it needs some contents synchronization mechanism.

C. Load Balancer

As we mentioned above, Meta Cloud Computing System is a natural extension of Cloud Computing System. Single Cloud Auto-Scaling increase servers inside a cloud. Inter-Cloud Auto-Scaling system increase servers across clouds. Auto-Scaling increases virtual computers automatically to scale out computing power. We faced new problem. We have to select computer to send request.

LLB: Local Load Balancer is an answer of this problem. Some of LLBs act as proxy server. It hides computers increased by auto scaling. Users do not need to know how many computers behind LLB. Users only need to send request to the LLB. LLB distributes request to suitable computers.

On the other hand, GLB: Global Load Balancer was used on widely distributed systems, ex. CDN: Contents Delivery Networks. CDN deploy their servers on the Internet widely. GLB also hides these servers as LLB does.

Load balancing mechanisms are difficult between LLB and GLB. Servers behind LLB are located nearly. Therefore, proxy server architecture is suitable. However, Servers behind GLB are located distributed. Therefore, most of GLBs used DNS mechanism and/or routing mechanism.

It is easy in principle to inspect request packet in LLB. However, it is no easy in GLB. DNS server cannot inspect request packet from client to server. It can inspect DNS request only. Routing mechanism cannot inspect any request packet. In addition, new servers are launched dynamically under Inter-Cloud Auto-Scaling. Therefore, result of Global Load Balancing may change dynamically. It is difficult to connect client to same servers to keep session. It is open issue.

IV. Conclusion

This paper shows concept of "Meta-Cloud Computing" and "Inter-Cloud Auto-Scaling". It becomes able to use multiple Cloud Computing System as a single virtual Cloud Computing System. It is natural extension of Cloud Computing System and Auto-Scaling. We discuss issue about using Meta-Cloud Computing for existing internet based systems.

We have to implement solution of these issue into our prototype system Soarin, and evaluate it in the future.

References

[1] Hiroshi MAENO, Yuko KAMIYA, Toshihiko SHIMOKAWA, "Study about High-Performance Virtual Machine Deployment on Inter-Cloud Autoscale," Proceedings of Internet Conference 2015, 79-84 (October, 2015) in Japanese

[2] Yuko KAMIYA, Toshihiko SHIMOKAWA, "A Study about Dynamic VM Image Deployment for Autoscaling across Multiple Cloud Systems," IEICE technical report (Internet Architecture), IA2013-33 41-44, (October, 2013)

[3] "fog - The Ruby cloud services library", http://fog.io/

[4] B. Claise, Ed., "Cisco Systems NetFlow Services Export Version 9," RFC3954 (October, 2004)

Yuko KAMIYA received the B.S., M.S. and Ph.D. degree in computer science from Kyushu Sangyo University in 2006, 2008 and 2011. She is a assistant professor of information science in the Department of Information Science at Kyushu Sangyo University since 2014. Her research interests include wide area distributed computation, cloud computing, and e-education. She is a member of The Institute of Electronics, Information and Communication Engineers.

Toshihiko SHIMOKAWA received received the B.E. and M.E. degree in computer science and communication engineering and the Ph.D. degree in computer science from Kyushu University in 1990, 1992 and 2001. He is a professor of information science in the Department of Information Science at Kyushu Sangyo University since 2007. His research interests include parallel and distributed computation, and wide area networking. He is a member of Information Processing Society of Japan and The Institute of Electronics, Information and Communication Engineers.

ISBN 978-4-9905448-6-7

All enquiries should be forwarded to APAN Secretariat (sec@apan.net)