

A Self-contained Software Suite for Post-Disaster ICT Environment Using Linux Live USB

Vaibhav Garg, Kotaro Kataoka, Sahil Sachdeva

Department of Computer Science and Engineering

Indian Institute of Technology Hyderabad, India

Email: {cs12m1014, kotaro, cs11b032}@iith.ac.in

Abstract—This paper proposes a self-contained software suite for post-disaster situations packaged in Linux Live USB flash drive, that enables ad-hoc Information and Communication Technology (ICT) environment. In post-disaster situations, recovery of ICT environment including network and user devices takes certain amount of time. Moreover, resources are limited while online communication is crucial everywhere no matter when and how such recovery happens. However, ICT rescue supply tends to be for heavy duty, expensive, limited in availability, heterogeneous, and large in size (i.e., bigger than a suitcase). Our approach utilizes laptop computers and Linux Live USB flash drive that remain in a disaster affected site and can be easily carried by rescue workers. Our solution is plug-and-play to override the computing and communication resource of a laptop computer without accessing the existing storage drive on it. Our system provides ad-hoc networking, Wi-Fi hotspot, web server and various local online services for immediate use without requiring Internet connection.

Keywords—Post-disaster Networking; Linux Live USB; Wireless Mesh Network; Wi-Fi Hotspot; Local Online Services.

I. INTRODUCTION

One of the main challenges that victims and rescue workers face at disaster affected site is the lack of Information and Communication Technology (ICT) resources that makes organizing and exchanging information difficult. The consequence is that disaster mitigation and recovery might slow down and become inefficient. Victims and their relatives will get mentally stressed if they are not aware of the well-being of each other. Given the critical nature of this information, this paper proposes a self-contained software suite packed in Linux Live USB flash drive to enable ad-hoc ICT environment in a post-disaster situation.

The idea is to pack everything that helps local and, if available, global communication and information exchange in a disaster site into Linux Live USB. The proposed system has following features.

- *Light-weight and Small*: Our software suite is independent of any heavy and big machinery. A laptop computer is enough to boot from Live USB and start serving for other devices.
- *Inexpensive*: USB flash drives are inexpensive and easily available. Laptop computers need not have any special configuration. USB wireless cards are also available at low price.
- *Easy to setup*: Our system is preconfigured and plug-and-play. If any configuration is required on-the-fly, GUI helps users without special skills and knowledge.

- *Everything through Wireless*: Live USB node deploys Wi-Fi hotspot through which many user devices can communicate. Multiple Live USB nodes can establish wireless mesh network to form a larger network coverage. If Internet connection through LAN, 3G, satellite, etc. is available, then, user devices in the domain can also share the Internet connection through hotspot and mesh.
- *Local Online Services*: Our system provides local online services like web, proxy, video/audio communication, etc. within the local network. Once Live USB node is launched, users can start practical communication immediately without waiting for the recovery of Internet connection.

The contribution of this paper is compiling the widely demanded and accepted ICT technologies to be easily deployable, flexible, and available in a feasible manner for immediate use in a challenging situation like post-disaster recovery.

The rest of the paper is organized as follows. Section II presents some challenges and lessons learnt from ICT recovery activity in response to 2011 Tohoku Earthquake and Tsunami in Japan. Section III discusses the solution using Linux Live USB. Sections IV and V depict System Architecture and Implementation, respectively. Section VI deals with System Evaluation and Discussion. Section VII presents Related Work. We conclude the paper in Section VIII.

II. CHALLENGES IN POST-DISASTER RECOVERY OF ICT ENVIRONMENT AND LESSONS FROM 2011 TOHOKU EARTHQUAKE AND TSUNAMI IN JAPAN

We claim that the preparedness to collect, store and exchange information immediately after a disaster occurs is very crucial, whether the mode of communication is local network or Internet. As Utani et al. [1] mentioned, on-the-fly development of counter-disaster systems and crowdsourcing made great contributions. Based on their summary, we focus on the fact that deployment of such systems was very rapid and ready to accept inputs from disaster sites.

On the other hand, we also faced several challenges while post-disaster network recovery activity in 2011 Tohoku Earthquake and Tsunami in Japan [2]. Following is a summary of lessons learnt from the experience.

- An expensive system is not sustainable to support multiple distributed sites.

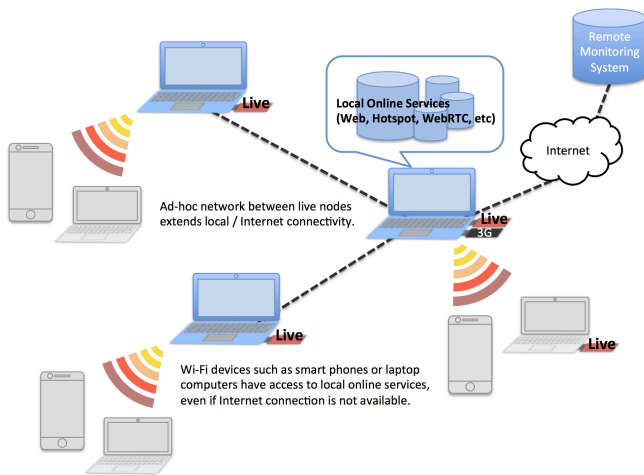


Figure 1. Laptop computers booting from Live USB and connecting to each other forming mesh topology.

- A big or heavy system introduces difficulty of logistics including packaging, dispatching, transportation, storage, planning and management.
- There might not be enough human resource to setup a technically complicated system. Again, handling multiple distributed sites will also be difficult.
- System should be up and running for a longer duration to reduce maintenance cost. Remote access is also important for health check and maintenance of the system.
- Some softwares/programs are written such that they need Internet connectivity to work even when local online communication is sufficient.
- It is not guaranteed that the Internet connection will be available soon.

We also made the following observations.

- Due to widespread use of smartphones, serving Wi-Fi hotspot helped people a lot.
- Thanks to the good preparedness for the disaster, power supply was sufficient to support additional devices. However, disaster sites faced power cuts.
- Laptop computers were available as popular rescue supply. However their maintenance cost is high. Who will conduct the update of operating systems and necessary software installation for each of them?
- Some desktop and laptop computers were there but not for public use. They were, of course, password protected and the data was untouchable. Hence, no one could use them.
- Mode of information dissemination is basically paper/notice board based in disaster sites. Digitizing such information and making it available online drastically improves the effectiveness of its dissemination.

III. SOLUTION USING LIVE USB

Our system is packed in Live USB flash drive. Live USB allows users to boot guest Operating System (OS) without touching the existing one in a host computer. This means that

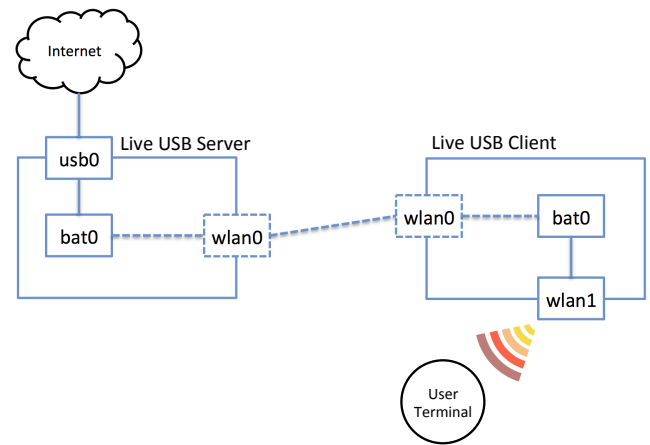


Figure 2. Internal connection of Live USB server, client and other devices.

our system makes use of the computing resource, and enables everything people want in a disaster site. As a USB flash drive is small and lightweight with enough storage capacity, it will solve most of the problems that we discussed in Sections I and II.

We raise three major benefits of using Live USB. First, the maintenance of software suite such as installing, deleting, upgrading and testing applications and the guest operating system can be done at a single place. So, users can save a huge amount of time and network bandwidth for setting up the disaster rescue mode of Operating Systems. Second, users need not worry about what Operating System is installed on the laptop computer. They need not have IT skills to handle available features as User Interface can be designed to be beginner-friendly. There is no need to know the password, if any, of the host Operating System installed on laptop computer. Almost any laptop computer can be used unless that system's Basic Input/Output System (BIOS) is password protected. Third, as the behavior of Operating System can be controlled, a Live USB node can be configured to connect to remote system, also known as Virtual Private Network (VPN) server, for remote login. This VPN server is present in global Internet, and Live USB node connects to it if and only when it can access Internet. VPN connection will enable human resources such as IT engineers to stay outside of disaster sites without needing to travel too much. If the remote login does not help in trouble shooting, the simple solutions are system reboot or replacement of the USB flash drive, that are not beyond IT skills of people in a disaster site.

A. Expected Deployment Scenario

Figure 1 presents the expected deployment scenario of our system. Laptop computers boot using Live USB flash drives that are carried to or prepared in a disaster affected site. The guest OS is pre-configured to support wireless mesh networking. Once the guest OS is booted, Wi-Fi devices of laptops try to connect to neighbor Live USB nodes. On some of the laptops, USB wireless cards can be used to create Wi-Fi hotspot, so that smartphones, laptops, sensors and any other Wi-Fi devices can also connect to the network. As an initial expectation, Live USB nodes will not move around. They are most probably deployed in a public space like evacuation

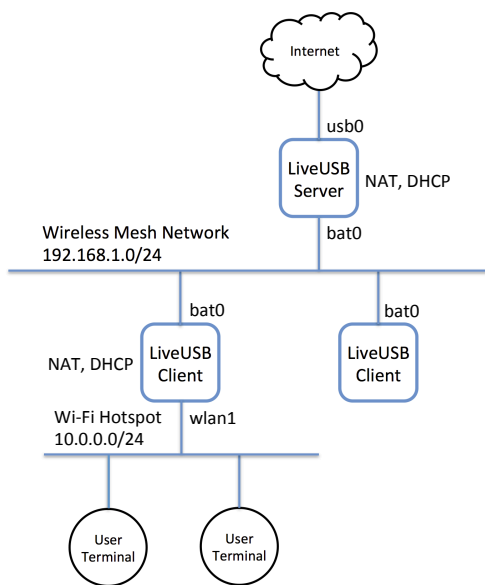


Figure 3. Network Diagram of Live USBs and other devices.

buildings and shared by users in turn, while providing various online services including Wi-Fi hotspot.

IV. SYSTEM ARCHITECTURE

A. Live USB Server and Client

A Live USB node acts either as a server or a client, depending on what packages it contains. The server provides local online services such as web, online storage, video/audio/text communication, etc. Only one Live USB server is active at any time in one disaster site. Live USB server can have Internet connectivity if it is available at disaster site. The server also creates a wireless mesh network to let other Live USB clients (or simply client) access the local online services and share the Internet connection. Thus, server acts as the gateway to Internet. Live USB server (or simply server) connects to VPN server so that it can be monitored remotely.

A Live USB client avails the functionalities provided by Live USB server. Depending on the needs at a disaster site, client connects to the server through wireless mesh network and accesses the local online services. It can also share the Internet connection, if it is available to Live USB server.

B. Network Design

Figure 2 depicts the internal connection of Live USB server, client and other devices. Our system works without an existing wireless or wired LAN infrastructure. Live USB nodes enable wireless mesh network to communicate with each other and share the local and Internet connection. As wireless mesh network forms a single layer 2 domain, it is important to reduce the amount of unwanted traffic.

All the Domain Name System (DNS) and Dynamic Host Configuration Protocol (DHCP) requests from Live USB clients are addressed by Live USB server. Server also acts as the Network Address Translation (NAT) router for the clients. On the other hand, Live USB client addresses DNS and DHCP requests from user terminals that are connected to it using Wi-Fi hotspot. Live USB server can also serve Wi-Fi hotspot, if

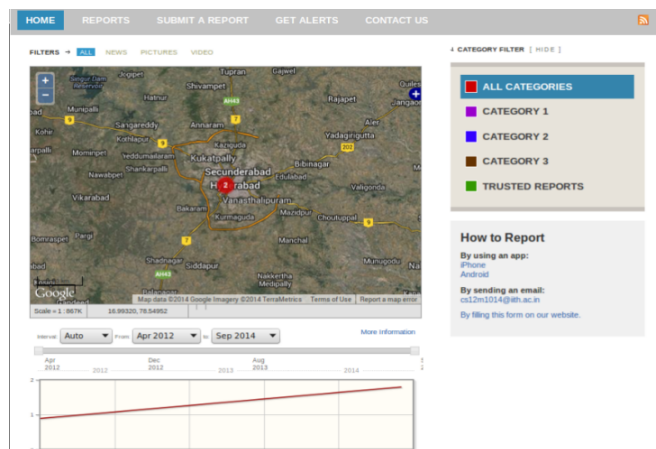


Figure 4. Location Based Report Submission in Ushahidi.

required. bat0 (Figure 3) interface of Live USB server has static IP address so that all the clients and user terminals can access services provided by the server.

C. Immediate Deployment of Local Online Services

To immediately enable ad-hoc communication and information exchange among users, Live USB server provides local online services that are pre-configured and ready to be deployed in a disaster affected site. Instead of installing variety of applications on each Live USB individually, our system integrates most of the applications on Live USB server that can be accessed by clients and other devices. The following two features introduce a lot of benefits that make ICT service and environment to be prepared for instantaneous information exchange.

- It reduces the maintenance cost of individual software client.
- By locally hosting services at Live USB server, these applications are “Internet-free” or offline at the moment of deployment. It will also save bandwidth.

Once the Internet connection becomes available, Live USB nodes and other user terminals can switch to “Internet-dependent” which is online mode.

V. IMPLEMENTATION

Server provides various services that are not required to be present on client. In order to reduce the size of client OS image, we prepare separate guest OS images for implementing server and client mode of Live USB. Table I summarizes the implementation details of the system.

A. Web Based Information Exchange

Web-based content management systems like MediaWiki [7], WordPress [8] and Ushahidi [9] (Figure 4) help to facilitate local information exchange. All these systems can be accessed via web browser and require single click operation. Ushahidi works only in online mode as it requires the exact GPS location of the sender of the report. Currently, work is being done to make it work in offline mode as well.

TABLE I. IMPLEMENTATION DETAILS OF LIVE USB SERVER AND CLIENT

Feature	Live USB	
Hardware	SanDisk Extreme USB 3.0 32 GB USB Flash Drive	
Guest OS	Linux Ubuntu 14.04.1 LTS	
Mesh Networking	Better Approach To Mobile Adhoc Networking (B.A.T.M.A.N.) Advanced [3]	
Wi-Fi Hotspot	hostapd [4] and dnsmasq [5]	
	Live USB Client	Live USB Server
Persistent Storage	4 GB	8 GB
Web Proxy	-	Squid [6]
Monitoring and Health Check	Live USB clients that are connected to the server can be monitored at the server.	Live USB servers can be monitored at Remote VPN server when server can access Internet.

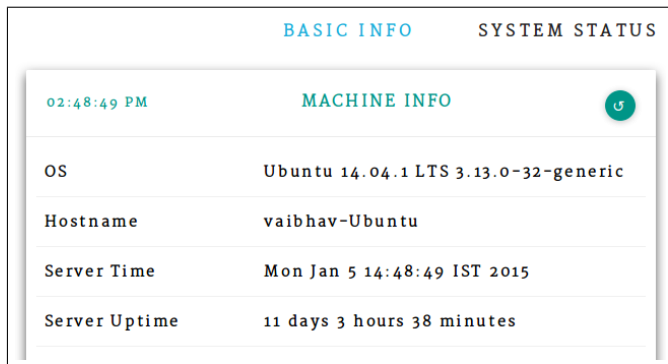


Figure 5. Linux Dash showing various statistics of Live USB Servers.

B. Real-Time Multimedia Communication

Our system enables audio, video and text chat using WebRTC [10]. WebRTC works by enabling Peer-to-Peer (P2P) connection between clients. In case, P2P connection is not successful (mainly due to NATs and/or firewalls), it falls back to Server-Client model by relaying data through some server in global Internet [11]. Ongoing work includes installing STUN server on Live USB server so that basic text, audio and video communication can take place without requiring Internet connection.

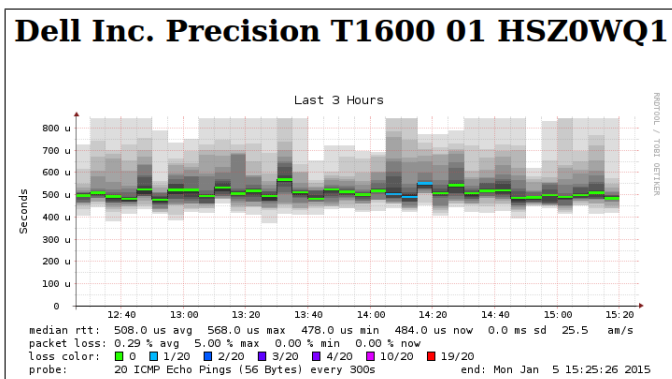


Figure 6. Once Live USB server successfully connects to VPN server, its entry gets automatically added as a monitoring target of SmokePing.



Figure 7. Server LiveUSB GUI for starting/stopping various services like Mesh Networking, Ushahidi and Hotspot.

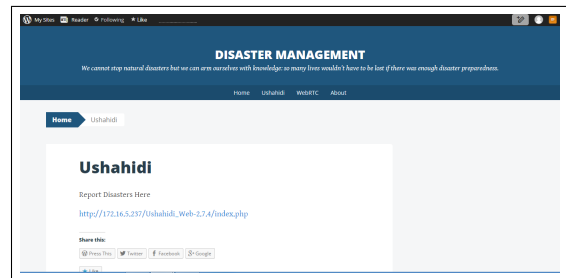


Figure 8. Client LiveUSB WordPress interface for WebRTC and Ushahidi.

C. System Monitor and Health Check

Each Live USB node keeps track of latest information such as memory, CPU, disk, I/O, swap, logged in accounts, common applications, ping speed etc. Linux Dash [12] is a simple web based dashboard that provides these statistics (Figure 5). Live USB server keeps track of all the Live USB clients that are connected to it. It provides a link to each of the client so that each client’s latest statistics can be monitored individually. On the other hand, Live USB server keeps attempting to connect to Remote VPN server, until it is successful in doing so. The script at the VPN server automatically adds an entry for that particular Live USB server in SmokePing [13] (Figure 6). VPN server keeps track of all the Live USB servers that are connected to it. Thus, Live USB servers can be monitored individually by VPN server. We send only minimal data to the VPN server to avoid unnecessary consumption of Internet bandwidth. Some engineering was done to manage the battery life of Live USB server node. When server’s battery goes critically low, it sends a message to remote VPN server and goes in hibernation mode. VPN server then sends intimation to the email id and/or contact no. provided by user. User can then charge the battery of laptop to which Live USB server is connected or use some other laptop computer as server.

D. GUI for Live USB Management

System GUI (Figure 7) contains some controllable features which help in setting up of the Live USB services like Mesh Network, Ushahidi and WordPress (Figure 8).

VI. EVALUATION AND DISCUSSION

A. Tested Hardware

We have confirmed that hardware summarized in Table II supports wireless mesh networking and Wi-Fi hotspot configured on the guest OS. All the testing was done using 5.18GHz frequency band (channel 36).



Figure 9. Multi-hop Scenario in IIT Hyderabad Hostel Premises.

TABLE II. VERIFIED HARDWARE AND DRIVERS

Laptop Model	Internal Wi-Fi Model
Acer Aspire 5750G	Atheros AR5B97 802.11b/g/n
Sony VAIO E Series	Atheros AR9285 VPCEB14EN 802.11b/g/n
HP Pavilion G6 1200tx	Broadcom 4313GN 802.11b/g/n
Wireless LAN Adapter Model	Drivers Used
Logitech LAN-W150N/U2	rt2800usb
Buffalo WLI-UC-AG300N	rt2800usb

B. One-hop Scenario

In one-hop scenario, two laptop computers were kept at a distance of around 50m from each other. It was made sure that they both were in line-of-sight of each other and there was no interference with neighbor devices. The test was performed in an open field in IIT Hyderabad campus. TCP throughput in this case was observed to be 17.7 Mbps and UDP throughput was 28 Mbps. As there is no interference in this scenario, these throughput values act as the benchmark for multi-hop scenario.

C. Multi-hop Scenario

In order to verify the performance of mesh in multi-hop scenario, we created a mesh of 4 laptop computers. The transmission power of Wi-Fi NIC was manually managed to make sure that only adjacent laptops (Figure 9) are in direct line-of-sight of each other. Since this test was conducted in hostel premises, there was interference as a lot of people were moving in between laptop computers. Iperf [14] server was enabled on Laptop 1 and Iperf client was used to test the throughput from each of the subsequent laptop to Laptop 1. TCP throughput was averaged over 3 runs, while in case of UDP, we consider the minimum, maximum and average throughput by changing the rate of data transfer. Figure 10 presents the results of the setup.

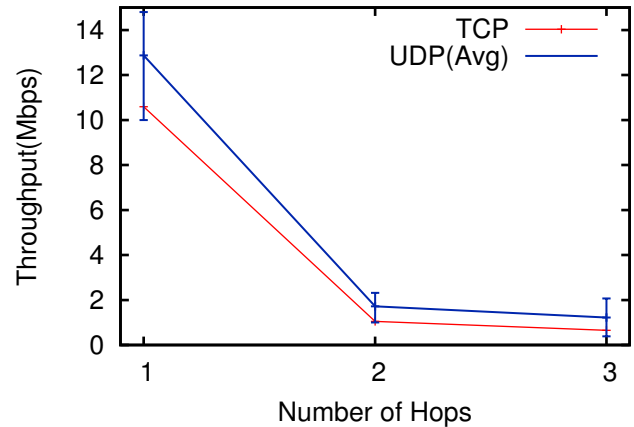


Figure 10. Results of Multi-hop Scenario.

D. What is a good SSID for Wi-Fi hotspot?

In order to well deliver our network connectivity and services to users, currently, we are using the unified SSID "00000INDIA". As in most cases SSIDs are sorted in alphabetical order, five zeros will contribute to place our SSID at the top of list. The choice of this SSID is inspired by discussions and guidelines across telecom networks [15] after 2011 Tohoku Earthquake and Tsunami in Japan. The discussion was about how to enable free and public Wi-Fi hotspots in disaster situation where a lot of commercial ISPs may provide their own hotspot services under different configurations. The discussion also covered the responsibility and other operational guidelines for operating public Wi-Fi hotspots.

E. Does our system replace existing solutions?

Our system does not aim to challenge existing post-disaster ICT environment and technologies. But it provides an alternative when the existing ones are not available. Some components of Live USB, such as local online services, might be easily integrated with the other systems as add-on.

F. Security in Open Network

Our network is basically open for everyone. However, such a network often faces security issues. In our deployment scenario, Live USB server and client should have mechanisms for security inspection and access control before letting user terminals start communication through Wi-Fi hotspot.

G. How do we address the wireless quality problem?

Our system is always subject to the quality of wireless communication as it works by creating single large mesh network. Some optimization can be done to enable multi-channel support using multiple USB Wi-Fi NICs on each Live USB node. This approach will require some engineering so that both the Wi-Fi NICs on same Live USB node are not configured in same channel. Also, giving cognitive feature to Live USB nodes will ease the performance degradation if a lot of Wi-Fi hotspots are deployed in a single place [16].

VII. RELATED WORK

Mobile ad-hoc networking and wireless mesh networks have been actively proposed and examined for post-disaster recovery. [17][18][19][20] discussed communication network using MANET, where highly demanded protocols and applications, like flooding-based communication protocol, push to talk and telemedicine, are made available in post disaster situation. [21] examined Multicast in MANET to efficiently disseminate information. [22] introduced Wireless Mesh Network that can be used in both daily life and emergency situations. [23] proposed to make use of survival time of wireless sensor networks and evacuate critical data to safe zone. [24] examined combination of Overlay Network and MANET to provide redundancy and continuity of services. Their solution expects the availability of high speed network to appropriately place mirror services through overlay network. [25] puts forward a session-based mobility management in MANET which meets the requirements of rapid deployment of the network with auto-configuration.

There has been significant amount of work done on recovering the ICT environment at a disaster site. [26] presented a fully-functional prototype of long-distance multimedia wireless mesh network during the time of large-scale disaster, but their solution is aimed at networking in open space, where mesh nodes may move around. [27] discussed the use of cognitive agents for bringing back telecommunication network in post-disaster aftermath. The cognitive agents run periodically to detect emergency situation, if exists. In case of emergency, the agents try to disseminate information to disaster information center. [28] proposed a quickly-deployable package for post-disaster communication, and [2] reported how such systems were used in 2011 Tohoku Earthquake and Tsunami in Japan. Their approach requires technically trained people for the setup.

VIII. CONCLUSION AND FUTURE WORK

This paper considered the challenges of having easy access to computers and communication infrastructures in post-disaster settings. It proposes a solution using Linux Live USB flash drive, where the guest OS and a set of preloaded software are ready to serve. One can attach the USB flash drive to an available computer, and boot her own OS to get access to the computing resources. A Live USB node can also act as a Wi-Fi hotspot in a wireless mesh setting. Our system is intended to fill the gap between the timings when people in a disaster site start to demand the ICT infrastructure and when such infrastructure is actually recovered and becomes ready to use. The result of field trial proved that our system can facilitate the online services and enable user terminals to access them through Wi-Fi hotspot and mesh network with or without the Internet connection.

So far, we have tested a very simple deployment scenario. The traffic engineering and optimization of backhaul connection under complex network topology, like supporting multiple Internet connections, will be future work to directly improve the network capacity. Also, some simulations can be conducted to explore the appropriate size of network for larger-scale deployment. As “crowdsourcing” provides a scaling out solution for data processing, data can be exported from the disaster site and made available for third-party software developers or data analysts. As part of software management

of Live USB, we need to examine the mode of information exchange which may vary by emergence of new technology or change of trend.

REFERENCES

- [1] A. Utani, T. Mizumoto, and T. Okumura, “How geeks responded to a catastrophic disaster of a high-tech country: Rapid development of counter-disaster systems for the great east japan earthquake of march 2011,” in Proceedings of the Special Workshop on Internet and Disasters, ser. SWID '11. New York, NY, USA: ACM, 2011, pp. 9:1–9:8. [Online]. Available: <http://doi.acm.org/10.1145/2079360.2079369>
- [2] K. Kataoka, K. Uehara, O. Masafumi, and J. Murai, “Design and deployment of post-disaster recovery internet in 2011 tohoku earthquake,” IEICE transactions on communications, vol. 95, no. 7, 2012, pp. 2200–2209.
- [3] Open-mesh, “B.A.T.M.A.N. Advanced Documentation Overview,” <http://www.open-mesh.org/projects/batman-adv/wiki/Doc-overview>, [Accessed: 2015-03-19].
- [4] Linux Wireless, “Hostapd,” <http://wireless.kernel.org/en/users/Documentation/hostapd>, [Accessed: 2015-03-19].
- [5] “Dnsmasq,” <http://www.thekelleys.org.uk/dnsmasq/doc.html>, [Accessed: 2015-03-19].
- [6] “Squid: Optimising Web Delivery,” <http://www.squid-cache.org/>, [Accessed: 2015-03-19].
- [7] “MediaWiki,” <https://www.mediawiki.org/wiki/MediaWiki>, [Accessed: 2015-03-19].
- [8] “WordPress,” <https://wordpress.org/>, [Accessed: 2015-03-19].
- [9] “Ushahidi,” <http://www.ushahidi.com/>, [Accessed: 2015-03-19].
- [10] “WebRTC,” <http://www.webrtc.org>, [Accessed: 2015-03-19].
- [11] Mozilla Corporation, “WebRTC Connectivity,” https://developer.mozilla.org/en-US/docs/Web/API/WebRTC_API/Architecture/Connectivity, [Accessed: 2015-03-19].
- [12] “Linux Dash,” <http://linuxdash.afaqtariq.com/>, [Accessed: 2015-03-19].
- [13] “SmokePing,” <http://oss.oetiker.ch/smokeping/>, [Accessed: 2015-03-19].
- [14] “Iperf,” <https://iperf.fr/>, [Accessed: 2015-03-19].
- [15] “Guideline About Large-scale Deployment of Free Open Public Radio LAN In Disasters Occurrence,” http://www.wlan-business.org/info/pdf/Wi-Fi_Free_Guideline_v1.01_20140527.pdf, [Accessed: 2015-03-19].
- [16] B. Tamma, B. S. Manoj, and R. Rao, “An autonomous cognitive access point for wi-fi hotspots,” in Proceedings of the Global Telecommunications Conference (GLOBECOM), Nov 2009, pp. 1–6.
- [17] T. Umedu, H. Urabe, J. Tsukamoto, K. Sato, and T. Higashinoz, “A manet protocol for information gathering from disaster victims,” in Proceedings of the Fourth Annual IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom), March 2006.
- [18] Y.-N. Lien, H.-C. Jang, and T.-C. Tsai, “A manet based emergency communication and information system for catastrophic natural disasters,” in Proceedings of the 29th IEEE International Conference on Distributed Computing Systems Workshops (ICDCS), June 2009, pp. 412–417.
- [19] Y.-N. Lien, L.-C. Chi, and C.-C. Huang, “A multi-hop walkie-talkie-like emergency communication system for catastrophic natural disasters,” in Proceedings of the 39th International Conference on Parallel Processing Workshops (ICPPW), Sept 2010, pp. 527–532.
- [20] J. K. et al., “Development of mobile ad hoc network for emergency telemedicine service in disaster areas,” in Proceedings of the International Conference on New Trends in Information and Service Science (NISS), June 2009, pp. 1291–1296.
- [21] M. Iqbal, X. Wang, D. Wertheim, and X. Zhou, “Swanmesh: a multicast enabled dual-radio wireless mesh network for emergency and disaster recovery services,” Journal of Communications, vol. 4, no. 5, 209, pp. 298–306.
- [22] Y. Takahashi, Y. Owada, H. Okada, and K. Mase, “A wireless mesh network testbed in rural mountain areas,” in Proceedings of the second ACM international workshop on Wireless network testbeds, experimental evaluation and characterization. ACM, 2007, pp. 91–92.

- [23] C. Debata and R. Roy, "Efficacy of power of two choices technique for data evacuation process in sensor networks for post-disaster relief operations," in Proceedings of the IEEE International Conference on Signal Processing, Computing and Control (ISPCC), Sept 2013, pp. 1–6.
- [24] Y. Shibata, H. Yuze, T. Hoshikawa, K. Takahata, and N. Sawano, "Large scale distributed disaster information system based on manet and overlay network," in Proceedings of the 27th International Conference on Distributed Computing Systems Workshops (ICDCSW), June 2007, pp. 33–33.
- [25] R. Li, R.-L. Yang, and D.-W. Hu, "A designing of mobility management mechanism in manet in disaster-rescue situations," in Proceedings of the 11th IEEE International Conference on Communication Technology (ICCT), Nov 2008, pp. 596–599.
- [26] K. Kanchanasut, A. Tunpan, M. Awal, T. Wongsardsakul, D. Das, and Y. Tsuchimoto, "Building a long-distance multimedia wireless mesh network for collaborative disaster emergency responses," Internet Education and Research Laboratory, Asian Institute of Technology, Thailand, 2007.
- [27] S. Majid and K. Ahmed, "Post-disaster communications: A cognitive agent approach," in Proceedings of the Seventh International Conference on Networking (ICN), April 2008, pp. 645–650.
- [28] K. Kataoka, K. Uehara, and J. Murai, "Lifeline station: A quickly deployable package for post disaster communications," in Proceedings of the Internet Conference, 2009, pp. 41–47.