AP MATRIX: A New Access Point Architecture for Reliable Public Wi-Fi Services

Zhe Fu[1, 2], Xiaohe Hu[1, 2], Xiang Wang[1, 2], Chang Chen[1, 2] and Jun Li[2, 3]
[1]Department of Automation, Tsinghua University, China
[2]Research Institute of Information Technology, Tsinghua University, China
[3]Tsinghua National Lab for Information Science and Technology, China
{fu-z13, hu-xh14, wang-xiang11, chenchang13}@mails.tsinghua.edu.cn, junl@tsinghua.edu.cn

***Abstract*—Public Wi-Fi hotspots are everywhere: in libraries, airports, shopping centers, etc. Public Wi-Fi services allow people to access Internet conveniently and freely, but today's unorganized deployments of wireless access points (APs) make the services unmanageable, unreliable, inefficient, unscalable, and high-cost. Besides, advanced network functionalities such as transparent migrations of TCP/IP sessions are often preferred for offering high-quality network services like remote desktop application in mobile environment. In this paper, we present AP MATRIX, a novel architecture for AP deployments, aiming at providing controllable, reliable, seamless, scalable, and low-cost Wi-Fi services in public places. The AP MATRIX architecture consists of three layers: slave AP, master AP and central controller. A prototype of AP MATRIX has been deployed at a university building and providing public Wi-Fi services with functions including network admission control, seamless handover, high reliability, load balancing and dynamic scalability.**

***Index Terms*—Wireless Network Management, Architecture, Deployment, Mobile, Hierarchical Control**

I. INTRODUCTION

Ubiquitous wireless network access allows people to access an increasing range of Internet services from various endpoint devices, including smartphones, phablets, tablets and laptops. Usually dozens of wireless hotpots could be found in coffee shops, libraries, airports, shopping centers, university buildings or other public places. However, such large quantities of wireless hotspots fail to provide powerful and reliable Wi-Fi services. Due to today's unorganized and unsystematic deployments, many wireless access points (APs) share the same default communication channels which causes interference between each other, leading to dropped connections or slow service, wasted bandwidth and energy consumption, as well as a big loss of economy. Several most pressing needs of public Wi-Fi services are summarized as follows.

Low cost. Taking shopping centers as examples, every shop prefers to own respective APs which are cheaper but also have lower performance compared to enterprise infrastructures. Though each AP doesn't cost much, the overall cost of deploying Wi-Fi services in that public place is staggering. Therefore, a centralized architecture or its deployment is demanded with the intent of lowering the cost of public Wi-Fi services.

High reliability. Home-using APs cannot provide high available and high reliable Wi-Fi services like those for enterprise using. Operation and maintenance personnel are often considered expensive for free services in public places. Besides, the update cycle of those APs will hardly be short in consideration of the cost. How to implement highly available and highly reliable Wi-Fi services over these cheap and thus unreliable devices is the key problem to be solved.

Seamless handover. Considering the inevitable movements of endpoint devices in public areas, a seamless handover technique, in particular for transparent migrations of TCP/IP sessions for applications such as HTTP downloading, FTP service and remote desktop, is needed for providing continuous network services.

Load balancing. Assuming that a sales promotion is held in a shopping center or a big conference is held at a university building, the APs around are carrying much more network traffic load than usual, which would cause the public Wi-Fi services congested or even collapsed. Network workloads should be balanced to optimize resource usage, keeping system stable and maximizing overall throughput.

Safety. Each person's identity varies in public place, e.g.: customers and clerks in a shopping center, guests and university staff at a university building. Public Wi-Fi services should provision security features while keeping the access to Internet convenient.

Apart from the points above, energy-efficiency, flexibility and scalability are also key considerations for public Wi-Fi services.

In this paper, AP MATRIX, a novel architecture is proposed for AP deployments to overcome the limitations of current public Wi-Fi services. The AP MATRIX architecture consists of three layers: slave AP, master AP, and central controller. Master AP takes charge of access authentication, traffic migration, and the management, configuration and dispatching of the slave APs. Data is only transmitted between the connected clients and the slave APs which dynamically build subnets in order to distinguish different services. The central controller has all information of the AP MATRIX architecture and coordinates master APs in different parts of the public place. Detailed design is described in Section II and III.

This paper makes the following contributions. First, we summarize the limitations of public Wi-Fi architecture nowadays. Second, AP MATRIX, a novel access point deployment architecture for provisioning reliable public Wi-Fi

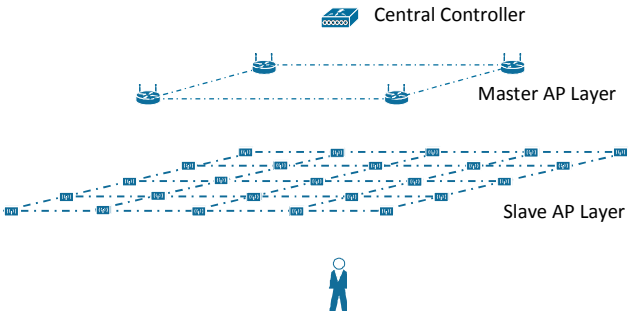

Figure. 1. Overview of the architecture of AP MATRIX

services is introduced. Third, a prototype of AP MATRIX is implemented in a real campus network and explored to demonstrate its efficiency and powerful network functionalities such as network admission control (NAC), load balancing, seamless handover, etc.

The rest of the paper is organized as follows. Section II presents an overview to show the design of AP MATRIX and its basic workflow. Section III describes the major functions of AP MATRIX. Section IV shows a prototype implementation deployed at Tsinghua University and its evaluation. Section V presents related work, and Section VI concludes with a summary and future research possibilities.

II. ARCHITECTURE

Figure 1 shows an AP MATRIX architecture, which consists of three layers: slave AP, master AP and central controller.

A. Overview

Slave APs are cheap, and sometimes old and outdated wireless access points which could hardly guarantee high performance and high reliable Wi-Fi services. In this architecture, multiple slave AP nodes are treated as an integrated data plane device, and a virtual tunnel migration method across this layer is designed and implemented, so that the probability of Wi-Fi connection failure is significantly reduced compare to that of a single node case. Besides, different slave APs could be dynamic connected to disparate subnets for the purpose of providing flexible and fine-grained control. Slave APs are densely-distributed, and usually invisible to clients which intend to join the network.

Master APs are more powerful and more reliable wireless devices compared to slave APs. They have the following duties: 1) Master AP should have control over all the slave APs in its zone, and monitor network traffic and system load of each slave APs using SNMP, to perform dynamic scheduling and load balancing. 2) As the only visible access entries to the endpoint clients before enrolled in the network, master APs take the responsibilities of user or endpoint device authentication and slave APs dispatching. To be more specific, every master AP maintains several continuously updated slave AP lists, which will be pushed into client after the client identity is authenticated. Then the client tries to connect to the slave APs according to the dispatched list. 3) Master APs communicate to the central controller to realize functions such as global network admission control and cross-zone migration.

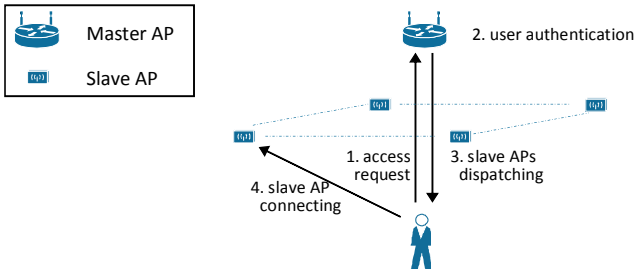

Figure. 2. Simplified workflow of AP MATRIX

Compared to slave APs, the deployment of master APs can be much sparser, but it should be guaranteed that at least one master AP would be found in any part of the public place.

Central controller handles the information of the whole AP MATRIX architecture, by which master APs could exchange information between each other. Only one central controller is needed for one public place; however, central controller is still an important role in this architecture for the sake of advanced network functionalities including global admission control and cross-zone migration of mobile endpoint devices.

B. Workflow

Figure 2 shows a simplified workflow of access to AP MATRIX, where the central controller is not mentioned. The basic steps are summarized as follows:

Step 1: When a client enters an area covered by AP MATRIX in which place only master AP can be seen, it sends an access request with its identity information to the master AP and waits for authentication.

Step 2: The master AP authenticates the client's identity (or forwards it to the central controller for more complex authentication and/or authorization procedures).

Step 3: Upon successful authentication, the master AP dispatches a list of several available slave APs in accordance with the client's identity.

Step 4: The client tries to connect to the slave APs according to the list dispatched from the master AP.

It must be noted that when it comes to specific use cases, more technical details need to be elaborated, which will be described in next section.

III. USE CASES

In this section, several typically use cases are introduced.

A. Network Admission Control

Network Admission Control (NAC) is an approach to secure access to network by devices when they initially attempt to entry into the network, aiming to improve the security of a proprietary network such as enterprise network by restricting the availability of network resources to endpoint devices. The AP MATRIX architecture allows easy implementation of generalized network admission control.

Since only master APs are visible to the clients, an access request is first sent from the client to the master AP. If the client is entering the network under AP MATRIX for the first time (which means no master APs or central controller contains the active record of this client), an unregistered client report will be

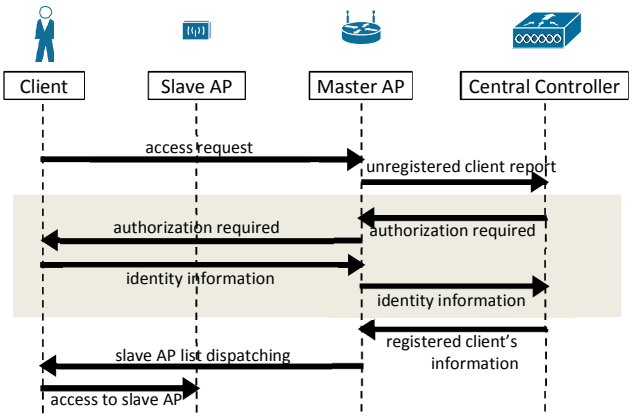

Figure. 3. Network admission control by AP MATRIX

sent from the master AP to the central controller. Because this client is new to AP MATRIX, the central controller has no active record either. As a consequence, the central controller sends an authentication-required message to the master AP, and then to the client. After the central controller receives the identity information of the client and finishes authentication, the authorization information is sent back to the master AP. The master AP dispatches several slave APs' SSIDs to the client according to the categories of identities. For example, a customer's client and a clerk's client may get entirely different list of slave APs' SSIDs according to predefined policy. Section III.C will see more details on services differentiation.

If an authenticated client just moves to another zone, the procedure of NAC can be much simpler (shaded part in Fig. 3 can be clipped), with the purpose of avoiding repeated authentication while providing safe and convenient access service. Reconnecting to different slave APs in the same zone doesn't require re-authentication either.

B. Seamless Handover

Seamless handover or migration is an important and indispensable function in communication networks due to the mobility of endpoint devices. Many works on seamless handover techniques have been proposed before, including handover within the same network type and handover across heterogeneous network types (between 802.11 networks and cellular networks). In traditional wireless network, it would require endpoint devices to re-associate to a new AP in order to realize handover, which may lead to a considerable service interruption, especially for TCP/IP-session-sensitive services. Under AP MATRIX architecture, seamless handover can be easily achieved without network reconfiguration of clients or APs, meanwhile TCP/IP sessions are kept alive so that applications such as remote desktop remain running during the handover procedure.

When a client successfully connects to a slave AP, a virtual tunnel is built between the client and the slave AP. Each client connected to the slave AP has its particular virtual tunnel, which is only determined by the account identity of the client. The slave APs can be considered as a shared data plane, which only acts the data transmission role through the virtual tunnel. Therefore, moving traffic from one slave AP to another only

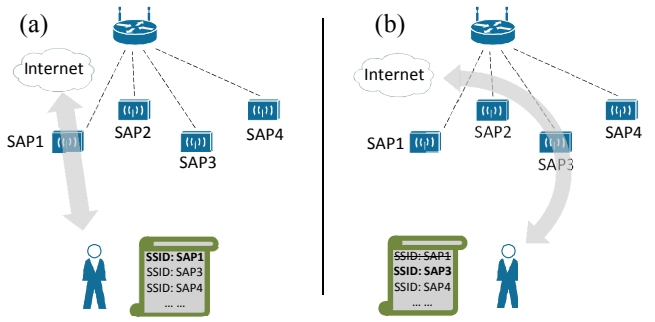

Figure. 4. Realization of high reliability before (a) and after (b) slave AP 1 breaks down

requires the migration of virtual tunnel the client is transmitting data through. No IP address, gateway address and other network configurations need to be changed, so all ongoing communications are kept active, and TCP/IP sessions are not lost. From the client perspective, it will be totally unware of the handover from one to another, except for some time loss which is so small that it is considered sustainable in practice.

Seamless handover is the base of advanced network functionalities including high reliability and load balancing. In Section IV.A, experiments of four different applications are conducted to evaluate the seamless handover implementation in AP MATRIX.

C. High Reliability

Slave APs are composed of old and outdated wireless access points which are unreliable and have poor performance. Thanks to the seamless handover technique, high reliability can be achieved over a group of multiple unreliable APs.

As shown in Fig. 4, a client receives a list of available slave APs' SSIDs from the master AP after it passes identity authentication. The client selects the first slave AP to connect. If the connection fails, the next slave AP is then selected in order. In case that none of the slave APs in the list can be connected successfully, the client will send a request message to the master AP to ask for another group of available slave APs. On the other side, the master AP polls each slave AP it manages periodically to check each AP's availability and resource utilization, so when it receives a request for slave APs, it can send response message back immediately.

D. Load Balancing

In public Wi-Fi services, load balancing distributes workloads across multiple network resources, aiming at optimizing resource usage and maximize overall throughput. In AP MATRIX architecture, each slave AP is taken as the finest unit to balance network traffic load under the control of the master APs.

Whether the load should be balanced is determined by the overall load of a slave AP, including CPU usage and network traffic throughput. For a certain slave AP, we define its standard valve of CPU usage as SV_{CPU} and traffic throughput as $SV_{traffic}$. Under these standard valves, a slave AP could work pretty well. Assuming that the overall load is negatively exponential distributed, the sampling time by master AP is t, and at n_{th} sampling point, the value of CPU usage and traffic

throughput are V_{CPU} and $V_{traffic}$. So the average overall load L of the slave AP over time T can be calculated as:

$$L_n = L_{n-1}e^{-\frac{t}{T}} + \frac{1}{2}\left(\frac{V_{CPU}}{SV_{CPU}} + \frac{V_{traffic}}{SV_{traffic}}\right)\left(1 - e^{-\frac{t}{T}}\right) \quad (*)$$

In the case of $V_{CPU} = SV_{CPU}$ and $V_{traffic} = SV_{traffic}$, the average overall load L tends to be 1, which means the slave AP works in its most appropriate condition. If the value of CPU usage or traffic throughput is far larger than its standard value, the average overall load L will increase sharply. On the contrary, the average overall load L will decrease according to the rule of negative exponent. Since slave APs are in different conditions, the standard valves of CPU usage SV_{CPU} and traffic throughput $SV_{traffic}$ of each slave AP vary from each other. The master AP gathers load information from each slave AP and sends back the m slave APs with the smallest load to the client when there is a request.

In general, we divide load balancing into two categories: proactive load balancing and reactive load balancing.

1) Proactive load balancing

As we described before, when a client first attempts to access the network, a list of available slave APs is dispatched from the master AP to the client. To implement proactive load balancing, appropriate slave APs and the order of each slave APs should be selected according to formula (*). Therefore, the same client may receive different lists of slave APs in different conditions, which balances network traffic and system load proactively.

2) Reactive load balancing

When several clients have already associated to a slave AP, if one of the clients generates burst traffic, a reactive mechanism to balance traffic to other slave APs is needed. When a slave AP is found under excessive load, the master AP would send a message with `destroy one virtual tunnel` command to the slave AP. As a response, the slave AP closes the virtual tunnel with the maximum volume of traffic, causing that the client which is transmitting data through that virtual tunnel loses connection (just like the slave AP is down), so it attempts to connect to the next slave AP. By taking these steps, the heavy traffic is migrated to another slave AP. Section IV.B will show some experimental results about reactive load balancing by AP MATRIX.

E. Dynamic Scalability

The AP Matrix architecture ensures elastic Wi-Fi services by dynamically provisioning network resources in real-time adjusting on demand, which brings the benefits of low energy consumption and high flexibility.

In Fig. 5(a), nine slave APs are managed by a master AP. Initially, four of the nine slave APs (blue ones) are linked to the Internet, while two of the slave APs (yellow ones) are connected to an internal network (for example, a specific network for the clerks in a shopping center). The remaining three slave APs (gray ones) are turned off to save energy. So in this situation, customers can only connect to the slave APs linked to the Internet because of the customers' identities, but clerks have the access to slave APs linking to the internal network.

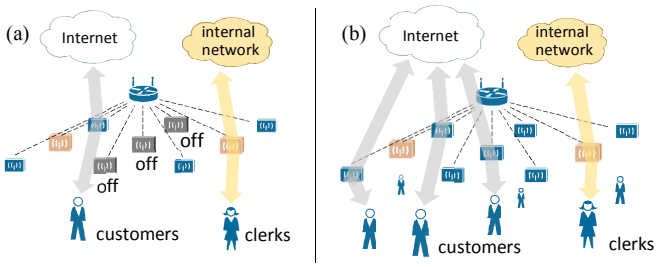

Figure. 5. Dynamic scalability of AP MATRIX

Thinking about a situation where the number of customers increases rapidly due to a sales promotion, only four slave APs for customers can hardly carry such a huge load. In response, the three unoccupied APs are configured as slave APs linked to the Internet (in Fig. 5(b), three slave APs are turned from gray to blue), so there are seven slave APs in total to provide Wi-Fi services for customers. In an even worse scenario, the slave APs for clerks can also be reconfigured to provide public Wi-Fi services for customers.

The roles the slave APs are acting is under the control of the master AP. If the master AP monitors that the average overall load L of all slave APs which play one same role is greater than a threshold (which is set to 3 in our implementation), it sends messages to wake the unoccupied slave APs up and gives them the role aforementioned. If all the slave APs have been occupied, other types of slave APs with the lightest loads would be selected to change roles. Nevertheless, this situation is very rare and should occur infrequently.

IV. Prototype Evaluation

We test our prototype implementation of AP MATRIX at a building of Tsinghua University. OpenWrt [13] is installed on each AP and OpenVPN [14] is introduced to build virtual tunnels. Two of the most important performance measures of the proposed architecture are evaluated: seamless handover and reactive load balancing.

A. Evaluation of Handover

In Fig. 6, a client is connecting to a slave AP and running a network application. Four applications are tested, including ping, TFTP uploading, HTTP downloading and file transferring using `scp` command. The vertical axis in Fig. 6 stands for received packet size or transferred file size ratio, and the horizontal axis stands for time. At time t_1 the slave AP is turned off, so the client has to connect to the next slave AP in the dispatched list. Suppose that at time t_2 the client finishes the handover procedure, at which time the applications resume running, the handover time $(t_2 - t_1)$ is shown in Table I. The average handover time is about 6.3s. Considering the benefit it brings to us, the time loss is acceptable.

TABLE I
HANDOVER TIME OF FOUR APPLICATIONS

Applications	PING	TFTP	HTTP	SCP
Protocol type	ICMP	UDP	TCP	TCP
Handover time	4.4s	7.9s	6.2s	6.6s

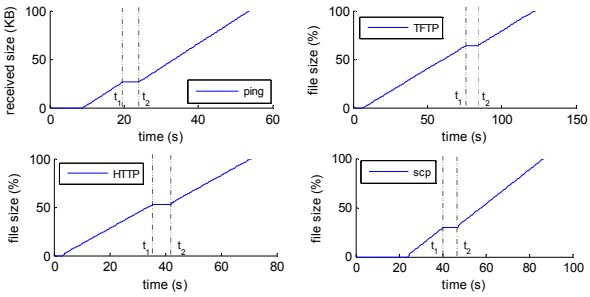

Figure. 6. Handover time of four applications

B. *Evaluation of Reactive Load Balancing*

Figure 7 shows a reactive load balancing process by AP MATRIX. First, only one client connects to slave AP 1, and the network traffic speed is about 700KB/s. At time t_1, slave AP 1 is connected by another client which generates burst network traffic subsequently. At time t_2 the master AP detects that slave AP 1 is seriously overloaded, so it sends a message to tell slave AP 1 to destroy the virtual tunnel that the second client is using. Then the second client attempts to connect to slave AP 2, which has little load at that moment. At time t_3 the second client begins to transmit data through slave AP 2, indicating that the load has been successfully balanced from slave AP 1 to slave AP 2 through seamless handover.

V. Related Work

Management of Wi-Fi networks has been widely studied in the past. Some researches [1, 2] focus on power control and rate adaptation to minimize interference among neighboring APs, ensuring robust end-client performance. LiveSec [3] suggests a flexible security management architecture for large scale production networks. DenseAP [4] is proposed to argue that dense deployment of APs can improve performance significantly for enterprise network. Unlike DenseAP, AP MATRIX provides reliable network services over unreliable AP nodes, especially for Wi-Fi services in public places.

Running an OS hypervisor on APs [5, 6] has been proposed to provide virtual APs. However, such full AP virtualization requires powerful APs (usually x86-based APs), which are not practical for public services. In AP MATRIX architecture, old and legacy APs can be used as slave APs in order to save cost. JMB [7] scales wireless capacity with user demands, but it is achieved by joint multi-user beamforming technique which is a lower-layer implementation compared to AP MATRIX. FMC [8] allows transparent migration of services in TCP/IP networks with dynamic configuration of a set of coordinated OpenFlow switches. OpenRoad [9], OpenRadio [10], CloudMAC [11] and meSDN [12] extends SDN control to wireless APs to support easy migration for mobile users across different types of network, but they focus more on physical and data link layers.

VI. Conclusion and Future Work

In this paper, we present AP MATRIX, a new architecture for providing reliable public Wi-Fi services. By abstracting the architecture into slave AP layer, master AP layer and a central

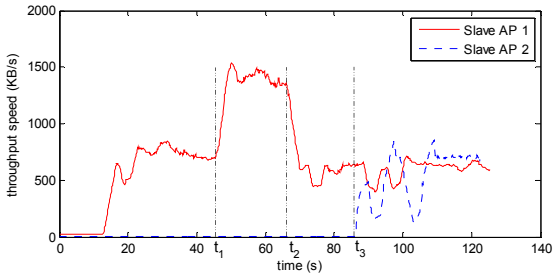

Figure. 7. Reactive load balancing by AP MATRIX

controller, AP MATRIX realizes high reliable services over a set of unreliable wireless access points, along with advanced functionalities including network admission control, seamless handover, load balancing and dynamic scalability. As a proof of our design, a prototype of AP MATRIX has been implemented at a university building, and it demonstrated the unique advantages of the new architecture.

We are confident that AP MATRIX has a great potential to enable more functionalities, meanwhile the application scenario will not be limited to Wi-Fi services for public places. Our future work will focus on reducing the handover transaction time by optimizing the logic in physical and data link layer, and further expanding the scope of the prototype.

References

[1] A. Akella, G. Judd, S. Seshan, and P. Steenkiste. Self-Management in Chaotic Wireless Deployments. Wireless Networks, 2007, 13(6): 737-755.

[2] I. Broustis, K. Papagiannaki, S. V. Krishnamurthy, M. Faloutsos, and V. Mhatre. MDG: Measurement-driven Guidelines for 802.11 WLAN Design. In Proc. of ACM international conference on Mobile computing and networking (MobiCom), 2007: 254-265.

[3] K. Wang, Y. Qi, B. Yang, Y. Xue, and J. Li. LiveSec: towards effective security management in large-scale production networks. In Proc. of International Conference on Distributed Computing Systems Workshops (ICDCSW), 2012: 451-460.

[4] R. Murty, J. Padhye, R. Chandra, A. Wolman, and B. Zill. Designing High Performance Enterprise Wi-Fi Networks. In Proc. of USENIX Symposium on Networked Systems Design and Implementation (NSDI), 2008, 8: 73-88.

[5] T. Hamaguchi, T. Komata, T. Nagai, and H. Shigeno. A framework of better deployment for WLAN access point using virtualization technique. In Proc. of International Conference on Advanced Information Networking and Applications Workshops (WAINA), 2010: 968-973.

[6] O. Braham, and G. Pujolle. Virtual wireless network urbanization. In Proc. of International Conference on the Network of the Future (NOF), 2011: 31-34.

[7] H. Rahul, S. Kumar, and D. Katabi. JMB: Scaling Wireless Capacity with User Demands. Communications of the ACM, 2012, 57(4): 235-246.

[8] R. Bifulco, M. Brunner, R. Canonico, P. Hasselmeyer, and F. Mir. Scalability of a mobile cloud management system. In Proc. of the first edition of the workshop on Mobile cloud computing (MCC), 2012: 17-22.

[9] K. K. Yap, M. Kobayashi, R. Sherwood, T. Y. Huang, M. Chan, N. Handigol, and N. Mckeown. OpenRoads: Empowering research in mobile networks. ACM SIGCOMM Compu. Commun. Rev., 2010, 40(1): 125-126.

[10] M. Bansa, J. Mehlman, S. Katti, and P. Levis. Openradio: a programmable wireless dataplane. In Proc. of the first workshop on Hot topics in software defined networks (HotSDN), 2012: 109-114.

[11] J. Vestin, P. Dely, A. Kassler, N. Bayers, H. Einsiedler, and C. Peylo. CloudMAC: towards software defined WLANs. ACM SIGCOMM Mobile Compu. Commun. Rev., 2013, 16(4): 42-45.

[12] J. Lee, M. Uddin, J. Tourrilhes, S. Sen, S. Banerjee, M Arndt, K. H. Kim, and T. Nadeem. meSDN: mobile extension of SDN. In Proc. of International workshop on Mobile cloud computing & services (MCS), 2014: 7-14.

[13] OpenWrt. https://openwrt.org/

[14] OpenVPN. https://openvpn.net/