Supporting "One Big AP" illusion in Enterprise WLAN: an SDN-based Solution

Dong Zhao*, Ming Zhu*, Ming Xu*
* College of Computer, National University of Defense Technology, Changsha, China
Email:{dongzhao, zhuming, xuming}@nudt.edu.cn

***Abstract*—IEEE 802.11 standard leaves client to determine AP association selection and initiate AP handoff. This often leads to sub-optimal results in enterprise WLAN with a large number of users. Therefore, network operators expects the ability to dominate AP association selection and AP handoff initiation so that better AP/client association can be achieved. In addition, AP handoff procedure involves time-consuming packets exchange. Although many methods have been proposed to simplify AP handoff procedures, there is at least hundreds of milliseconds during which client's data transmission is blocked. In this paper, we proposed the abstraction of "*One Big AP*" in which unmodified 802.11-compliant client is tricked into believing that there is *one* access point with very large coverage, while the AP that sends and receives its packets has changed. To support the"*One Big AP*" illusion, we leveraged the emerging idea of *Software-Defined Networking (SDN)* to reform the architecture of enterprise WLAN. In the SDN-based architecture, we can easily realize network-controlled AP association and client-transparent AP handoff. We demonstrated the feasibility and efficiency of our solution through simulation.**

***Keywords*—SDN, enterprise WLAN, AP handoff**

I. Introduction

Today we can find an increasing number of access points (APs) deployed in public areas. In enterprise environment, These APs are interconnected through wired backbone network, constituting the scaffold of enterprise WLAN. According 802.11 standard [1], each client must associate one and only AP to be attached to the backbone and enjoy network services. 802.11 standard lets client to make decision about AP association and AP handoff. That is to say, client determines by itself which AP to associate with, which AP to handoff to and when to handoff.

However, such "client-controlled" mechanism often leads to sub-optimal AP/client association results in enterprise WLAN due to the myopic view of local decision. For example, in Fig. 1, suppose client chooses to associate with the AP with highest signal strength, then all the four clients would simultaneously associate with the right AP. This leads to imbalanced load distribution between the two APs. Apparently, we could have better choice: two APs associate with the left AP regardless of the fact that the signal of the left AP perceived by client is slightly weaker.

To achieve better overall performance, some methods have been proposed to let network involved in the determination of AP association and handoff. These methods either rely on AP providing additional information to aid in client's decision [2], [3], or assume that there exists a central server that makes

Fig. 1. Client-controlled AP association leads to imbalanced AP load distribution.

decision based on the global view [4]–[6] and can send a specific instruction to clients to enforce its decision. However, all these methods inevitably need to modify clients, which is infeasible and impractical for real applications [7].

Another issue in our concern about enterprise WLAN is the latency of AP handoff. AP handoff is time-consuming procedure as it needs many complicated signaling among client, old AP, new AP, as well as switches and routers in wired backbone. There are many methods proposed to reduce the latency of every procedure of AP handoff from various aspects, including fast scanning scheme [8], [9], fast re-authentication [10], [11], and fast route update in wired backbone [12]. However, AP handoff latency still exists and is at least hundreds of milliseconds in spite of these optimization.

In this paper, we proposed the concept of *"One Big AP"*: an abstraction for clients. We envision that all the APs in enterprise WLAN collaboratively provide an illusion for every client, making them believe that there is only one AP with large coverage area. When client move in the service area, we leverage the broadcast nature of wireless signal to realize *network-controlled client-transparent* AP handoff. Fig. 2 shows the illusion.

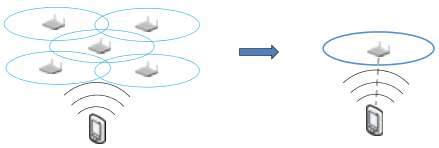

Fig. 2. "One Big AP" illusion: client is tricked into believing that there is only one AP with very large coverage area and it is always interacting with this AP, while its associated AP has changed.

To support the proposed "One Big AP" illusion in enterprise WLAN, we draw on the emerging idea of "Software-Defined Networking" to reform the architecture of enterprise WLAN. SDN [13] is a revolutionary networking paradigm that

978-1-4799-7339-2/14/$31.00 ©2014 IEEE

makes control and forwarding of packets more flexible and programmable. The core of SDN is to decouple control logic from network devices and perform centralized control over the forwarding behavior of network switches. SDN, along with its supporting technology OpenFlow [14], offers a promising solution to address the aforementioned issues. We proposed an SDN-based architecture for enterprise WLAN: *SDWLAN*, in which we can easily realize "One Big AP" in enterprise WLAN.

More specifically, SDWLAN supports "One Big AP" from the following aspects:

- It supports network-controlled AP association and AP handoff, which is most important motivation of "One Big AP" and makes it possible to achieve better overall network performance.
- It supports client-transparent AP handoff without triggering any explicit AP handoff procedure. This eliminate the latency of AP handoff.
- Our solution needs no modification to 802.11 standard and 802.11-compliant client, which makes our solution practical and deployable.

II. Overview of "One Big AP"

Similar with the 'Virtual Cell" proposed by Meru [15], we aim to let APs operate on the same channel and form a very large coverage area. In "One big AP" illusion, every client can see only *one* AP in service area, so that it has no other choice, and no explicit AP handoff procedure won't be triggered. From client's view, it believes that it is always interacting with one AP, while it may actually communicating with many different APs (As shown in Fig. 2). Meru's solution is proprietary and the details are not available.

In enterprise WLAN, client's associated AP plays two roles: "receiver and forwarder", and "sender". In uplink direction, AP is "receiver and forwarder" that receives the client's packets, responds ACKs to clients and forward packets to wired backbone that delivers packets to their destination. In downlink direction, AP is the sender in that every packets destined to client are first delivered to client's associated AP and then sent to the client.

Wireless signal is broadcast in nature which can be heard by all the nearby nodes. Therefore, client's packets can be received by several APs simultaneously. The key to achieve "One Big AP" illusion is client-transparent fast AP handoff mechanism. AP handoff should happened without interruption to client's on-going traffic. We can achieve this from two direction. In uplink direction, we let old AP stop receiving client's packets and responding ACK and let the new AP receives packets and respond ACK on behalf of the old AP. In this way, client's "receiver and forwarder" changed without noticing client. In downlink direction, we direct all the client's packets to the new AP, and let the new AP send packets to the client. To trick client to believe that packets is still from its associated AP, the new AP send packets with old's BSSID.

Apparently, we can achieve "silent" change of client's "receiver and forwarder" and "sender" as long as ensure that at any time there is one and only one AP providing these "services" for the client. This entails a high degree of cooperation among APs and wired backbone.

III. SDN-based Architecture

The proposed SDN-based architecture of enterprise WLAN is called SDWLAN. Its goal is to support flexible and fine-grained control over enterprise WLAN to realize network-controlled, client-transparent AP handoff without requiring any change to clients. This is the key to support "One Big AP". The reason why we introduce SDN is the requirement of high degree of cooperation among APs and wired backbone. In this section, we first introduce basic components in SDWLAN, and then explain how to reorganize 802.11 AP MAC-layer functions to realize the decoupling MAC-layer management policy and packet processing. And then we detail the new device, "*Wireless Access Switch (WAS)*", and our extension to OpenFlow standard to support our architecture.

A. Overview

SDWLAN includes a central controller, a set of *wireless access switches (WASes)*, a OpenFlow-based wired backbone network which is composed of several OpenFlow-enabled *wired backbone switches (WBSes)*, and some optional appliances (shown in Fig. 3,). The controller is responsible for taking over all the WASes and WBSes .

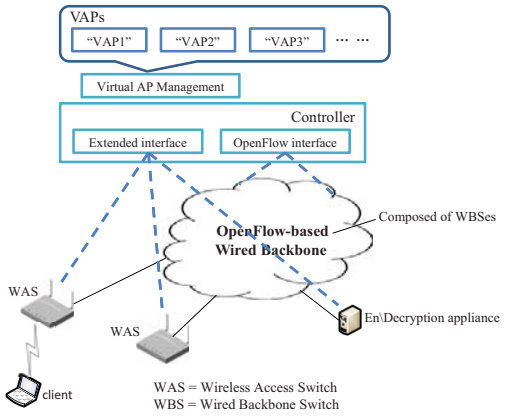

Fig. 3. SDWLAN includes a centralized controller, a number of WASes, an OpenFlow-based backbone network composed of several OpenFlow-enabled WBSes, and some optional appliances.

Wired backbone switches (WBSes) are just OpenFlow-enabled switches that can be communicated with and controlled through standard OpenFlow protocol. They constitute the wired backbone network, interconnecting WASes and external network. The wired backbone just acts as a uniform routing fabric that capitalizes on IP and Ethernet technology to deliver packets between WAS and external network (Internet). Any packet destined for a client should be delivered to the client's associated WAS and then sent to the client.

It is noted that we use the term "*wireless access switches (WAS)*" instead of access point (AP). In SDWLAN, traditional AP MAC-layer have been reorganized. We decompose 802.11 AP MAC-layer into several modules, and extracts most of AP MAC-layer function modules from original AP and place

them onto controller, leaving a simplified device. Except for standard OpenFlow protocol, the controller can manipulate WASes through an extended OpenFlow protocol (mentioned in section III-C) to realize the reorganization of 802.11 AP MAC-layer function.

We in effect merge two controllers: the WAS controller and the WBS controller. The outcome is a unified control platform of wireless APs (more exactly, WAS in SDWLAN) and wired backbone. Using the interface provided by controller, we can develop various applications to coordinate the behavior of WASes and WBSes to improve network performance more efficiently and implement some functions which are difficult to realize in traditional architecture (e.g., network-controlled client-transparent fast AP handoff).

Operators can deploy some additional appliances at any location as required to aid the completion of some specific functions. For example, We will introduce en/decryption appliances later to aid network-controlled, client-transparent fast WAS handoff (seen in section IV). We can define our proprietary protocol between the application and corresponding appliances to exchange necessary message between them.

B. The Reorganization of 802.11 AP's MAC-layer

According to 802.11 standard [1], we decompose 802.11 AP's MAC-layer function into the following functional modules (as shown in Fig. 4 (a)):

1) **Beaconing & Probe Response**: AP should constantly generate beacon packets. Some clients may broadcast probe request and AP is supposed to respond to them with probe respond packet. Both beacon and probe response are to claim the AP's existence and covey information about the AP's configuration and features.

2) **Association & Re-association**: Client should associate with an AP by sending association request. If the client's identity is verified, AP will respond to the client by sending associate response. Reassociation is used when client determine to associate with a new AP. In effect, association is a procedure to register client to wired backbone, so that the client's traffic can be correctly routed to client's AP.

3) **Authentication & Re-authentication**: when a client wants to associate with an AP, the AP should initiate authentication challenge and verify client's identity. In enterprise environment, AP plays an intermediate role to forward authentication request and response packets between clients and remote authentication (RADIUS) server.

4) **En/Decryption**: In a secure enterprise WLAN, user's traffic are encrypted to be securely transmission in open air to keep confidentiality. So AP should encrypt packet data frame being sent to clients and decrypt data frame received from clients.

5) **ACK & RTS/CTS**: ACK and RTS/CTS mechanism is the feature of 802.11 wireless MAC that wired MAC don't have. Every 802.11 unicast frame (data frame or management frame) should be responded with an ACK frame. A transmitted data frame without receiving ACK is believed to be failure and should be retransmitted. RTS/CTS handshake is optional mechanism used to clear wireless channel when sending long data frame.

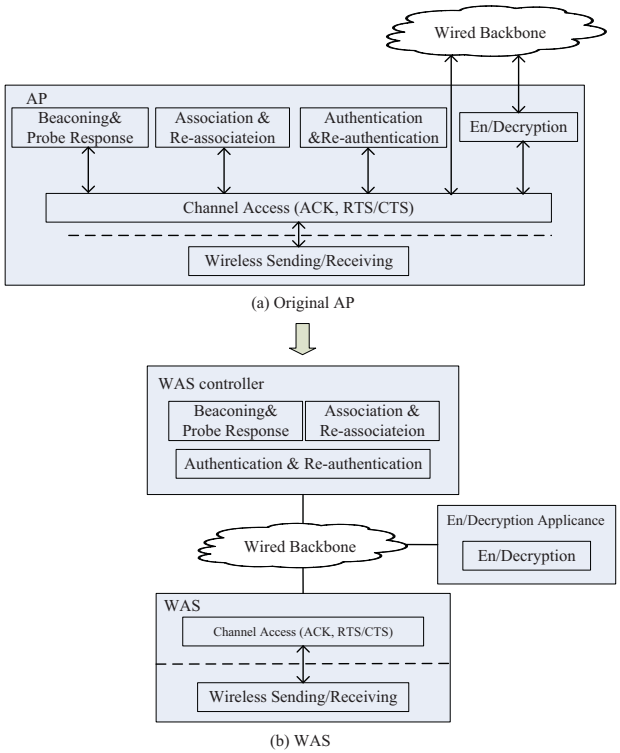

Fig. 4. The reorganization of 802.11 AP's MAC-layer. Most of original AP's functional modules are lifted to controller, while "ACK & RTS/CTS" module, together with PHY-layer 802.11 wireless sending and receiving function, remains in WAS.

We reorganize 802.11 AP's MAC-layer. As shown in Fig. 4 (b), the following functional modules are moved to the controller: *"Beaconing & Probe response", "Association & Re-association", "Authentication & Re-authentication", and "En/Decryption"*. In contrast, *"ACK & RTS/CTS"* module, together with PHY-layer 802.11 wireless signal sending/receiving function, remains in WAS. As a result, most 802.11 AP MAC-layer intelligence has been removed from original AP. That is why we replace original term "AP" with the coined term "WAS".

The controller takes over these WASes and instructs them how to deal with 802.11 packets through extended OpenFlow interface. Specifically, the controller guides WASes whether or not to respond ACK for a packet received from client. We will detail the extension of OpenFlow in the following section.

The synergy of controller, WASes, and some optional appliances realizes the complete 802.11 AP MAC-layer function. We implemented a *"Virtual AP management application"* that coordinates the configuration and packets processing of WASes, WBSes and related appliances. We will detail the VAP management application in the following section.

C. Wireless Access Switch and Extended OpenFlow

As shown in Fig. 5, a WAS is equipped with a wireless NIC and a wired Ethernet NIC. Except for the wireless NIC, WAS has no difference with wired switch. Naturally, WAS

supports standard OpenFlow protocol just as a OpenFlow-enabled switch. So the WAS controller can communicate with WASes through standard OpenFlow protocol.

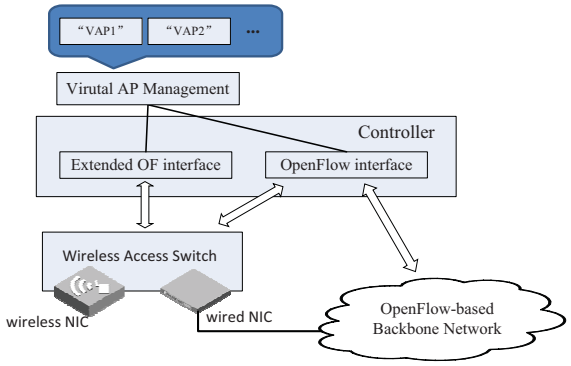

Fig. 5. Wireless Access Switch (WAS) has two NICs: one wireless NIC and one wired NIC. The controller uses extended OpenFlow interface to communicate with WAS.

Apart from standard OpenFlow, we proposed an extra interface used between WAS and controller, i.e., extensions of OpenFlow protocol. The extensions are critical to implement the proposed reorganization of AP function. Our extensions are based on the latest OpenFlow specification version 1.4.0 [16].

1) **New matching field: BSSID.** 802.11 MAC-layer packet has a specific field "BSSID" that is used to identify AP. MAC-layer packet from Ethernet to wireless client should be converted into 802.11-specific format by adding the field and filing in it with the AP's BSSID before forwarding to the client. Conversely, packet from wireless client to Ethernet will be checked if the value of "BSSID" field is exactly AP's BSSID and then converted into Ethernet format. We extend OpenFlow specification by adding this new field so that controller can instruct WAS to match BSSID field of received packet. An fringe benefit is that, in conjunction with and rewrite BSSID of output packet as needed.

2) **New actions: MAC_ACK and CTS_ACK.** Due to the unreliability of wireless channel, the sending of an 802.11 unicast packet should be followed by an ACK. By adding new action "MAC_ACK", the controller can instruct WAS whether to respond to a received packet with ACK or not. Similarly, RTS/CTS is a specific mechanism to 802.11 MAC-layer transmission, the action "CTS_ACK" enable controller to determine whether a WAS should respond a specified RTS handshake request.

IV. Network-Controlled, Client-Transparent Fast AP handoff in SDWLAN

As mentioned above, the key to support "One Big AP" is Network-controlled, Client-transparent fast AP handoff mechanism. We developed an application "Virtual AP Management", which takes advantage of SDWLAN to realize such "One Big AP" illusion. Specially, "Virtual AP Management" coordinates packets processing behavior of WASes, WBSes, and some appliances involved to support network-controlled client-transparent fast AP (WAS) handoff. "Virtual AP Management" application creates several virtual AP (VAP), and maintain some basic information of virtual AP, including SSID, MAC address, supported authentication methods, the address of AAA server, and so on. The application also stores every client's association state information.

We deployed several appliances (called en/decryption appliances) around wired back bone. These appliances is to encrypt and decrypt 802.11 MAC-layer packets on behalf of virtual APs that are stored in controller. By installing rules on WASes and WBSes involved along specified path, encrypted wireless packets are encapsulated by WASes and directed to these appliances. The appliances are responsible for decrypting these packets and send to their destination. Similarly, packets destined for clients are also routed to these appliances by wired backbone, and appliances encrypt them and then send them to their corresponding WASes. As you can see, packets from/to the clients should travel along a well-designed path.

In SDWLAN client's packets can be categorized into three flows as shown in Fig. 6. Red dotted line denotes 802.11 MAC-layer management packets that are sent to controller. Green dashed line denotes encrypted packets that are sent to specified en/decyption appliance, and the appliance keeps the key, decrypts the packets and sends them back to the wired backbone. The green solid line represents the decrypted packets. Blue solid line is the Unencrypted packets that are directly routed their destination. In SDWLAN, controller can easily achieve this by installing appropriate rules in WBSes through OpenFlow interface.

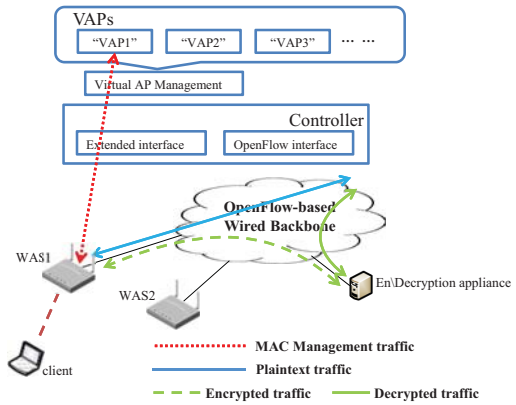

Fig. 6. The flow of a client. Red dotted line represents flow of 802.11 MAC-layer management traffic, green dashed line represents flow of encrypted packets traffic, green solid line represents the decrypted packets traffic, and blue solid line represents unencrypted packets traffic.

Reducing AP handoff delay is the critical point to improve users' experience in enterprise WLAN. Traditional AP handoff process takes considerable time of one second or so. In this section we attempt to take advantage of SDWLAN to design a network-controlled client-transparent fast AP[1] handoff method.

In SDWLAN, AP Handoff involves two critical steps: first, changing client's "serving" WAS; second, steering the client's

[1]As explained in prior section, the term "WAS" replaces "AP" in SDWLAN, but we still use "WAS" and "AP" interchangeably for easy of exposition. In this section, "AP handoff in SDWLAN" is equivalent to "WAS handoff in SDWLAN".

traffic from old WAS to new WAS. A client's "serving" WAS means the client's "receiver and forwarder" and "sender", as we mentioned above.

The first step needs to coordinate the old WAS and new WAS to ensure there is one and only one WAS serving a client at any time.

The second step is to ensure packets destined for a client can be correctly routed to the client's serving WAS. In WAS handoff, client won't be conscious of the change of its actual serving WAS and any MAC-layer re-authentication and re-association or network-layer update will never be triggered. So the client can preserve the MAC-layer connection status and IP address. Any on-going session won't cause significant interruption as long as the background operation in WAS, WBS, and controller can be done in a short time.

Change of serving WAS. We use "*serving*" WAS instead of "*associated*" here. In SDWLAN, client is essentially associated with virtual AP that is stored on controller side. When a client joins in the network, the "virtual AP management application" takes over association management and control, and client's association state information is stored on controller side after authentication. WAS acts as an agent of virtual AP and provides service on behalf of the virtual AP. So, we call it "*serving*" WAS in this section. As a result, change of serving WAS in SDWLAN is equivalent to change of associated AP in traditional architecture.

The first time a client joins in the network, virtual AP will assign the client a unique BSSID, and the serving WAS will communicate with the client using the assigned BSSID. Suppose client A is assigned $BSSID_A$, we only need to install two rules (shown in Fig. 7) on a WAS to let that WAS act as the serving WAS of client A. The first rule means that any packet, which are from wireless NIC port, whose BSSID field is set to be $BSSID_A$, and whose MAC source address is client A's MAC address should be responded with ACK and then forwarded to the wired NIC. The second rules means that any packet, which are from wired NIC port, and whose MAC destination address is client A's MAC address should be forwarded to wireless NIC with its "BSSID" field set to be $BSSID_A$. By deleting the two rules from old WAS and installing them on new WAS, we can realize the change of "serving" WAS.

Ingress Port	BSSID	MAC Dst	MAC Src	…	Actions
Wireless NIC	$BSSID_A$	…	MAC_A	…	ACK, Forward to wired NIC, …
Wired NIC	…	MAC_A	…	…	Set "BSSID" to ***BSSID***$_A$, Forward to wireless NIC, …

Fig. 7. Rules installed on client A's serving WAS.

Routing update in wired backbone. Directing a client's packets from old WAS to new WAS involves routing update update route in wired backbone. Since the controller control WBSes in wired backbone through OpenFlow interface, we just need to install appropriate forwarding rules in WBSes involved to achieve correct routing. We have realized network-controlled WAS handoff in SDWLAN, therefore we can proactively install rules before updating rules on old WAS and new WAS. In this way, we can eliminate the latency of routing update in wired backbone.

V. Evaluation

In this section, we evaluated the performance of fast AP handoff in SDWLAN. Since our contribution focus on fast AP handoff mechanism, rather than handoff policy, we artificially executed handoff procedure in SDWLAN and traditional WLAN. We setup UDP and TCP connection between client and remote server node, and measured the instant throughput overtime. We calculate the instant throughput every 0.2 seconds, and the results are shown in Fig. 8 and Fig. 9.

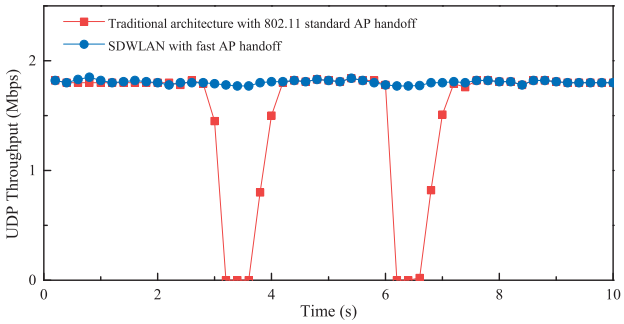

Fig. 8. Effect of AP handoff procedure on UDP instant throughput over time (calculate every 0.2 second). Handoff occurred 3 seconds and 6 seconds after the beginning the experiment respectively.

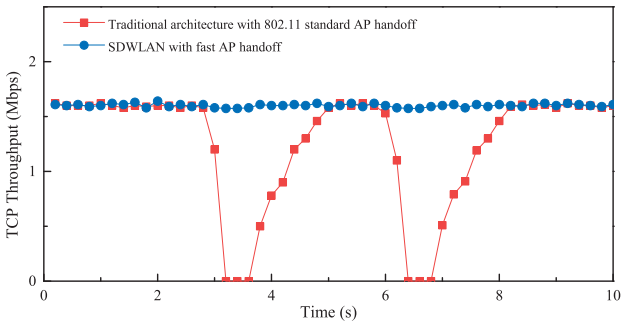

Fig. 9. Effect of AP handoff procedure on TCP instant throughput over time(calculate every 0.2 second). Handoff occurred 3 seconds and 6 seconds after the beginning of the experiment.

By comparing the effect of AP handoff mechanism on UDP and TCP throughput in traditional WLAN and SDWLAN, we found that 802.11 standard AP handoff mechanism in traditional architecture lead to a relatively long time interruption of TCP/UDP session. It takes nearly one second to recover from such interruption caused by AP handoff (For TCP, it takes over one second due to TCP's "Slow Start" mechanism). In Traditional WLAN, 802.11 standard AP handoff involves MAC-layer re-authentication and re-association, MAC address relearning (for layer-2 handoff), and DHCP procedure or mobile IP register signaling (for layer-3 handoff). On the contrary, the proposed fast WAS handoff mechanism eliminates these procedures. So we found that in SDWLAN with fast WAS handoff mechanism, both UDP and TCP throughput

are slightly affected by handoff operation. It indicates that AP handoff operation performed by controller can be done very quickly, incurring very short duration of communication interruption.

VI. Related Work

Attempts to apply SDN to WLAN can be found in [17], [18] and [19]. [17] and [18] proposed to use OpenFlow to monitor traffic flows and provide a GUI for administrators to control traffic flows. [18] aimed to realize network virtualization by OpenFlow, and slice network according to user's requirements or application characteristics. But [18] does not allow to control the IEEE 802.11 MAC layer.

In Odin [19], users' association states are kept on a central controller and AP is responsible for authentication and beaconing. Odin introduce LVAP, which records a user's association context. With a user's LVAP, AP can communicate with the user. AP handoff in odin is realized by removing LVAP from old AP and spawning it in new AP. This inevitably takes quite some time, while AP handoff in SDWLAN have no such overhead. Moreover, since LVAP contains client's key, client's mobility may lead to key scattering throughout several APs. This increases security risk. There is no such issue in SDWLAN because client's key is stored in only one en/decryption appliance.

CloudMAC [20] is the most similar work with SDWLAN in that it also lifts MAC-layer management function onto central controller. However, CloudMAC don't mention how to unified the wired controller and wireless controller. Moreover, due to the lack of OpenFlow extension proposed in SDWLAN, CloudMAC only supports switching all the associated clients from one AP to a new AP at the same time, which don't satisfy the requirement of per-client AP handoff.

VII. Conclusion

In this paper, we proposed "One Big AP" illusion to achieve network-controlled AP association and AP handoff in enterprise WLAN. In "One Big AP" illusion, client can only see one AP with large coverage area so that we can achieve client-transparent AP handoff. To support the illusion, we leveraged SDN to reform the architecture of enterprise WLAN so that cooperation between APs and wired backbone can be achieved. we demonstrated by experiment that SDWLAN can significantly reduce the adverse effect of AP handoffon client's throughput. In addition, our solution requires no modification to 802.11 standard and 802.11-compliant clients, which make it practical and deployable.

Acknowledgment

This work was partially supported by the National Science Foundation of China under Grant No. 61379144 (2014-2017).

References

[1] *802.11-2007: IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, IEEE Std. 802.11-2007, 2007.

[2] S. Makhlouf, Y. Chen, S. Emeott, and M. Baker, "A network-assisted association scheme for 802.11-based mesh networks," in *Proceedings of IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 1339–1343, 2008.

[3] A. Fujiwara, Y. Sagara, and M. Nakamura, "Access point selection algorithms for maximizing throughputs in wireless lan environment," in *Proceedings of International Conference on Parallel and Distributed Systems (ICPADS)*, vol. 2, pp. 1–8, Dec 2007.

[4] C. Yue, G. Xue, H. Zhu, J. Yu, and M. Li, "S3: Characterizing sociality for user-friendly steady load balancing in enterprise WLANs," in *Proceedings of IEEE International Conference on Distributed Computing Systems (ICDCS)*, pp. 491–499, 2013.

[5] R. Murty, J. Padhye, A. Wolman, and M. Welsh, "Dyson: An architecture for extensible wireless LANs," in *Proceedings of USENIX Annual Technical Conference (USENIX ATC)*, pp. 10–15, 2010.

[6] Y. Bejerano, S.-J. Han, and L. Li, "Fairness and load balancing in wireless LANs using association control," *IEEE/ACM Transactions on Networking (TON)*, vol. 15, no. 3, pp. 560–573, Jun. 2007.

[7] P. Lv, X. Wang, M. Xu, and Y. Chen, "Network-leading association scheme in IEEE 802.11 wireless mesh networks," in *Proceedings of IEEE International Conference on Communications (ICC)*, pp. 1–5, 2011.

[8] I. Ramani and S. Savage, "SyncScan: Practical fast handoff for 802.11 infrastructure network," in *Proceedings of IEEE INFOCOM*, pp. 675–684, 2005.

[9] H. Wu, K. Tan, Y. Zhang, and Q. Zhang, "Proactive Scan: fast handoff with smart triggers for 802.11 Wireless LAN," in *Proceedings of IEEE INFOCOM*, pp. 749–757, 2007.

[10] *802.11r: IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 2: Fast Basic Service Set (BSS)*, IEEE Std. 802.11r, 2008.

[11] S. Pack, H. Jung, T. Kwon, and Y. Choi, "SNC: A selective neighbor caching scheme for fast handoff in IEEE 802.11 wireless networks," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 9, no. 4, pp. 39–49, 2005.

[12] R. Koodli. (2008, June) Mobile IPv6 fast handovers. RFC 5268. . IETF RFC 5268. [Online]. Available: http://tools.ietf.org/html/rfc5268.

[13] "Software-Defined Networking: The new norm for networks," White Paper, Open Networking Foundation (ONF), Open Networking Foundation (ONF), Tech. Rep., April 2012.

[14] 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 (CCR)*, vol. 38, no. 2, pp. 69–74, Mar. 2008.

[15] Meru, "Virtual Cells: The only scalable multi-channel deployment," Meru Networks White Paper, Meru, Tech. Rep., Aug 2009. [Online]. Available: http://me.westcon.com/documents/36960/Virtual_Cell_White_Paper.pdf.

[16] (2013, October) OpenFlow switch specification version 1.4.0. Open Networking Foundation (ONF). . Open Networking Foundation (ONF). [Online]. Available: https://www.opennetworking.org/images/stories/downloads/sdn-resources/onf-specifications/openflow/openflow-spec-v1.4.0.pdf.

[17] R. Mortier, T. Rodden, T. Lodge, D. McAuley, C. Rotsos, A. Moore, A. Koliousis, and J. Sventek, "Control and understanding: Owning your home network," in *Proceedings of International Conference on COMmunication Systems and NETworkS (COMSNETS)*, pp. 1–10, 2012.

[18] Y. Yiakoumis, K.-K. Yap, S. Katti, G. Parulkar, and N. McKeown, "Slicing home networks," in *Proceedings of ACM SIGCOMM Workshop on Home networks (HomeNets)*, pp. 1–6, 2011.

[19] L. Suresh, J. Schulz-Zander, R. Merz, A. Feldmann, and T. Vazao, "Towards programmable enterprise WLANs with Odin," in *Proceedings of ACM SIGCOMM Workshop on Hot Topics in Software Defined Networks (HotSDN)*, pp. 49–54, 2012.

[20] P. Dely, A. Kassler, J. Vestin, N. Bayer, H. Einsiedler, and C. Peylo, "CloudMAC: An OpenFlow-based architecture for 802.11 MAC Layer processing in the cloud," in *Proceedings of IEEE Broadband Wireless Access Workshop*, 2012.