A Method to Simulate A Large Number of AP Upline

Lianfen HUANG*, Fujia YAN*, Baosheng HUANG**, Liang XIAO*

*Department of Communication Engineering, Xiamen University, China

**Fujian Radio and Television University Sanming Campus, China

lfhuang@xmu.edu.cn, yanfujia911@163.com, smhbs99@163.com, lxiao@xmu.edu.cn

***Abstract*—As such a convenient data transmission system, Wireless Local Area Networks(WLANs) using radio frequency technology, replacing the old architecture of twisted copper pairs, shunting a 2G/3G data flow and helping to build a 2G comprehensive coverage and 3G hotspot coverage with three characteristics of "high bandwidth", "IP-based" and "mobility", has played an increasingly important role in our life. Wireless Access Controller (AC) is an essential part in WLANs based on fit AP(Access Point) network mode. However, long term stable operation of AC is always a big problem haunting us. And how much pressure AC can bear is considered to be a key point of its stability performance. In this paper, a method simulating a large number of virtual AP upline instead of physical AP is put forward to provide sufficient pressure to AC, in order to guarantee the stability of AC and improve testing efficiency.**

***Keywords*—WLAN, AC, multi thread, CAPWAP, raw socket**

I. INTRODUCTION

In recent years, we have witnessed increasing penetration of wireless broadband Internet to our everyday life, mainly due to the availability of affordable Wi-Fi capable consumer products such as laptops, PDAs and gaming devices. In order to allow these devices to exchange data through the Internet, more and more people are setting up WLANs in their homes, common areas or wherever wireless connection is needed[1]. WLANs use an entirely different network protocol and are deployed in different topologies. The purpose of WLANs is primarily to provide LAN connectivity to portable and mobile stations (laptop computers, voice handsets, bar-code readers, etc.), though fixed-station use is becoming more popular as the technology becomes widely adopted[2]. WLANs using 2.4GHz and 5GHz band radio frequency technology, through the IEEE 802.11 standard[3], make network construction and terminal mobility more flexible.

In the WLANs, we can use the fat or the fit AP network. This paper based on the fit AP network mode makes the AC play a very important role in WLANs. Undertaking unified configuration management and data forwarding, the AC greatly simplify initial installation and later maintenance inputs, and it's easy to manage and suitable for operation level large scale network. So the stable working of AC is an important index of AC performance and a large number of APs on-line is considered to be an important aspect of AC stability. This paper using a computer to simulate a large number of APs on-line instead of the physical environment of a large number of real APs, greatly improve the test efficiency of AC stability.

II. WLAN NETWORK ARCHITECTURE

Traditional WLANs use fat AP network mode. Fat AP refers to completely independent management, data forwarding and security control. Because of the independent work between APs, the lack of a unified management tool, roaming users support and effective access and security control strategies, it is suitable for household or small WLANs. But fit AP network mode, which is carried out based on the construction of large scale WLANs, mainly moves the function of fat APs to AC for unified management. In the fit AP network mode, AP is only an antenna and radio frequency module and its network configuration should be downloaded from AC. The fit AP mode via the centralized management of a large number of fit APs by AC and constituting an organic whole between the AC and every fit AP, can be a better solution to solve the user roaming, user access and security control. The WLAN network system structure is shown in Figure 1. In the figure, AC mainly complete WLAN user access control, accounting information collection and wireless business management.

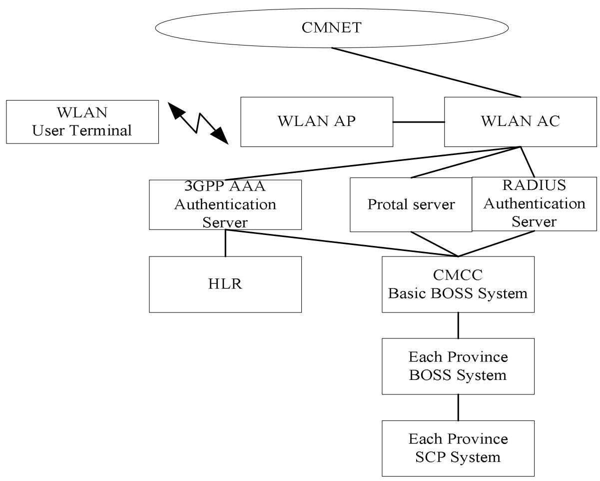


Figure 1. CMCC WLAN network system structure

III. AP UPLINE PROCEDURE

By means of automatic discovery, security authentication, obtaining software image and initial and dynamic configuration from AC, the CAPWAP(Control And Provisioning of Wireless Access Points Protocol), which is defined in RFC5415[4] and RFC5416[5], is used to communicate between AP and AC. The CAPWAP based on C/S architecture uses UDPv4 and UDPv6 as carrying protocol.

Currently CAPWAP data type may generally be divided into two categories. One is control packet travelling in control tunnel, and the other one is data packet travelling in data tunnel. The implementation of CAPWAP is based on three-layer network. It means that all the CAPWAP packets are encapsulated as UDP packets format and transferred in IP network. And the CAPWAP tunnel is maintained by the AC interface IP address and AP IP address.

In this paper, the method to simulate AP upline is based on CAPWAP control tunnel, the source port and destination port are respectively 2000 and 5246. CAPWAP identify information interactive process by state machine transfer. The state machine transfer is: Discovery→Join→（Image Data）→Configuration→ Data Check→ Run. In the Join state, AC establish control tunnel with AP.

Before entering CAPWAP process, AP obtain an IP address through DHCP protocol. As the client of DHCP, AP use port 68 and AC built-in DHCP server process use port 67 as the server of DHCP. The DHCP flowchart between AP and AC is shown in Figure 2. AP obtain IP address from AC via field option43 in DHCP OFFER.

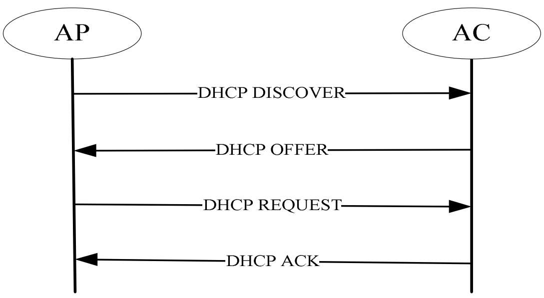


Figure 2. DHCP flowchart between AP and AC

AP enter CAPWAP process after obtaining AC interface IP address via field option43 in DHCP OFFER packet. With the transfer of CAPWAP state machine, AC and AP will take appropriate actions to make AP online. The steps of AP upline are usually divided into 5 processes as follows:

1. Discovery process: if AP is configured with static IP address, AP will send Discovery Request packet to broadcast discover available AC in the network. And if AP is configured with DHCP discovery way, it will enter Join process instead of this step.
2. Join process: in this state, AP will send Join Request to AC for service. And AC answer Join Response to indicate its ability and willingness to provide service to the AP. In this paper, we suppose the AP's software has updated and needs no upgrading.
3. Configuration process: AP send Configuration Status Request, and report its current configuration. And AC answer Configuration Status Response to update configuration of AP.
4. Data check process: AP send Change State Event Request to confirm or report error status. AC answer Change State Event Response packet.
5. Run process: when entering run state, AP is online. In the Run state, AP will send Echo Request periodically, which is used to keep the link between AP and AC alive. When AP 's EchoInterval timer expires, AP will send Echo Request packet and AC answer Echo Response, and reconfigure the EchoInterval timer. If the timer expires and AC has not received Echo Request or any other control packets, AC must have detection ability to make AP offline. Figure 3 shows the process of AP upline.

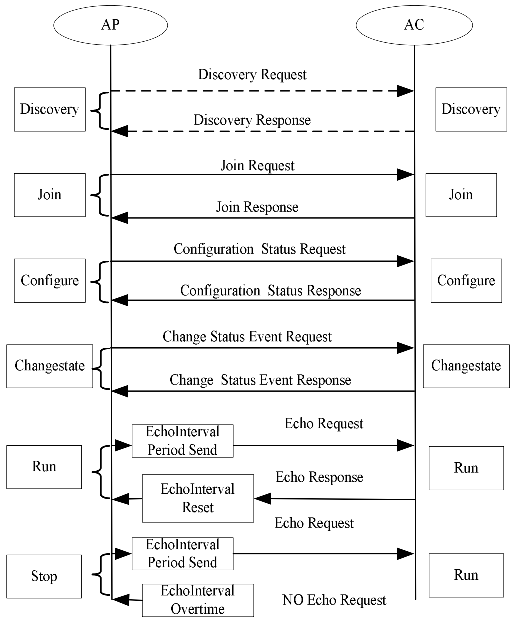


Figure 3. AP upline interactive process

IV. AP UPLINE IMPLEMENTATION

CAPWAP and DHCP both use UDP as carrying protocol. In this paper, we use a computer to simulate many virtual APs. So the source MAC address, source IP address of packet sent from AP to AC must be different. We encapsulate packets from Ethernet layer, and the ETH header, IP header and UDP header should be contained in the packet. The form of packet is shown in Figure 4. The simulation software which is designed to send and receive packet by raw socket, is running on the Linux platform. To prevent discarding packets, the network card must be configured to hybrid mode because actually the network card will discard packets whose destination MAC address is not local or broadcast. The virtual AP MAC address must be different from each other, and is also different from local physical MAC address.

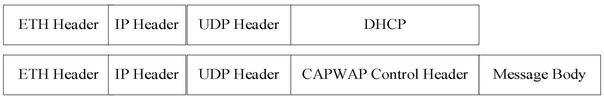


Figure 4. DHCP and CAPWAP packet format

The simulation software sends and receives packets by raw socket. We create and process the packet of different mock NIC(Network Interface Card) based on multi-thread programming. Each layer processes its own business. Data packet flow is shown in Figure 5. CAPWAP and DHCP flow are processed by application layer. At first, mock NIC which have not been configured with static IP address will obtain IP address by DHCP and be online by CAPWAP flow. But if mock NIC is configured with static IP, it will skip DHCP and enter CAPWAP flow directly.

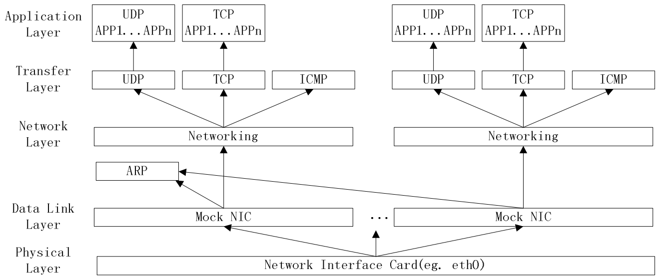


Figure 5. Data packet flow

V. Experimental Results and Analysis

The experiment tests AC stability performance by adding the number of virtual AP constantly and continues to give AC pressure. In this experiment, AC system software is running on ATCA 7220 board whose theoretical processing capacity is up to 14G and can handle 1024 APs. The software architecture is shown in Figure 6. Left is Linux program and right is SE(Simple Execute) program. In the Linux program, up is application layer, among which apm is charge to process all the services of AP and STA(station), include CAPWAP, AP manage, authentication and accounting, and dhcp is charge to process DHCP. The middle is middle layer and bottom is operating system layer and hardware layer. The SE program includes APP: the application module of SE; UDP/TCP: the analysis module of layer 4; IP: IP layer module, process IP route, data packing and unpacking; Eth/Vlan: Ethernet layer and VLAN layer module; HAL: simple execute layer of Cavium; Manager: layer manage. Linux and SE program communicate with each other by shared memory.

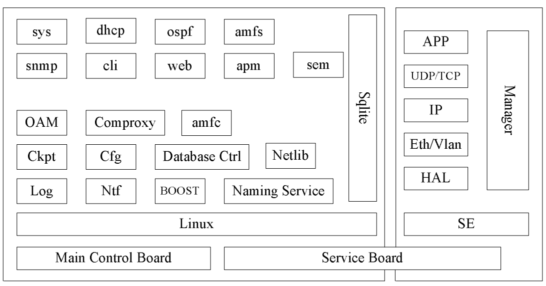


Figure 6. AC software architecture

Computers simulating more than 500 APs connect each other by layer 2 switch just as Figure 7 shows. PC1, PC2 and PC3 run simulation software and PC4 is used to capture packets between AC and virtual AP.

Simulation software simulates a virtual AP upline who is configured with DHCP discovery way. The packet captured in PC4 with Wireshark sniffer software is shown in Figure 8. We can see in the figure that virtual AP succeed to obtain IP address by DHCP and be online by CAPWAP. We can get the results shown in Table 1 through adding the number of virtual AP and running time is 30 minutes.

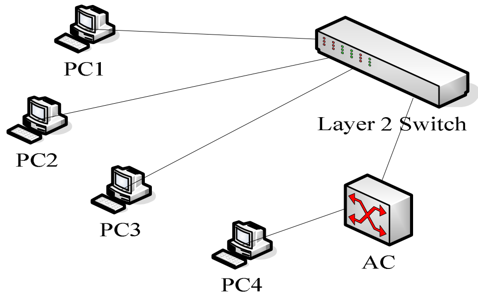
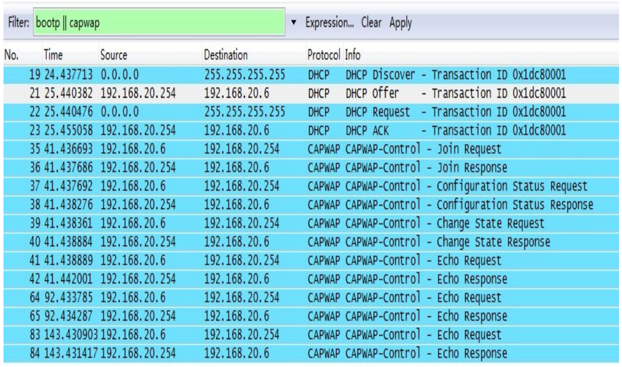


Figure 7. Experiment continuous graph



No.	Time	Source	Destination	Protocol	Info
19	24.437713	0.0.0.0	255.255.255.255	DHCP	DHCP Discover - Transaction ID 0x1dc80001
21	25.440382	192.168.20.254	192.168.20.6	DHCP	DHCP Offer - Transaction ID 0x1dc80001
22	25.440476	0.0.0.0	255.255.255.255	DHCP	DHCP Request - Transaction ID 0x1dc80001
23	25.455058	192.168.20.254	192.168.20.6	DHCP	DHCP ACK - Transaction ID 0x1dc80001
35	41.436693	192.168.20.6	192.168.20.254	CAPWAP	CAPWAP-Control - Join Request
36	41.437686	192.168.20.254	192.168.20.6	CAPWAP	CAPWAP-Control - Join Response
37	41.437692	192.168.20.6	192.168.20.254	CAPWAP	CAPWAP-Control - Configuration Status Request
38	41.438276	192.168.20.254	192.168.20.6	CAPWAP	CAPWAP-Control - Configuration Status Response
39	41.438361	192.168.20.6	192.168.20.254	CAPWAP	CAPWAP-Control - Change State Request
40	41.438884	192.168.20.254	192.168.20.6	CAPWAP	CAPWAP-Control - Change State Response
41	41.438889	192.168.20.6	192.168.20.254	CAPWAP	CAPWAP-Control - Echo Request
42	41.442001	192.168.20.254	192.168.20.6	CAPWAP	CAPWAP-Control - Echo Response
64	92.433785	192.168.20.6	192.168.20.254	CAPWAP	CAPWAP-Control - Echo Request
65	92.434287	192.168.20.254	192.168.20.6	CAPWAP	CAPWAP-Control - Echo Response
83	143.430903	192.168.20.6	192.168.20.254	CAPWAP	CAPWAP-Control - Echo Request
84	143.431417	192.168.20.254	192.168.20.6	CAPWAP	CAPWAP-Control - Echo Response

Figure 8. DHCP and CAPWAP process packet

TABLE 1. Statistical Number of Virtual AP On-line

Virtual AP numbers	Offline AP numbers	On-line success rate
500	0	100%
800	38	95.25%
900	42	95.33%
1000	92	90.80%
1024	347	66.11%
1100	411	62.64%
1200	619	48.42%
1300	832	36.00%
1400	910	35.00%
1500	1073	28.47%

From Table 1 we can see that the on-line success rate when the number of virtual AP is less than 1024 is obviously greater than when it is more than 1024. In the process of upline, some AP cannot be online because they have not obtained IP address at the beginning and some drop because of the timer EchoInterval timeout. So we must make sure the number of AP is less than 1000 when the AC is in its practical application.

VI. CONCLUSIONS

As a access method at hotspots, WLAN is playing an increasingly important role in our daily life. At the same time AC, which is the essential key equipment in the WLAN network, plays a very important role. But long term stable operation of AC is always a big problem troubling us. Usually we give AC enough pressure by adding a large number of real APs to test its stability. But in this way, it will cost many manpower and material resources. This paper presents a method to test the stability of AC by making use of a computer to simulate a large number of AP upline, and greatly improves the efficiency of testing stability of AC.

ACKNOWLEDGMENT

This work is supported by the National Natural Science Foundation of China (no.61172097), the Natural Science Foundation of Fujian Province, China (no.2012J01424) and is supported in part by NCETFJ, Fundamental Research Funds for the Central Universities (2012121028), and NSFC(61271242).

References

[1] Xiaonan Yue, Chi-Fai Wong, and S.-H. Gary Chan, "A Distributed Channel Assignment Algorithm for Uncoordinated WLANs", Annual IEEE Consumer Communications & Network Conference, 2010.

[2] Tom Alexander, "Optimizing and Testing WLANs", Elsevier's Science &Technology, 2007, vol.1, pp.1-2.

[3] IEEE 802.11, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, IEEE standard for information technology telecommunication and information exchange between systems-local and metropolitan area network-specific requirements, 1999.

[4] [RFC5415] Calhoun, P., Montemurro, M., and D.Stanley, "Control And Provisioning of Wireless Access Points (CAPWAP) Protocol Specification", March 2009.

[5] [RFC5416] Calhoun, P., Montemurro, M., and D.Stanley, "Control and Provisioning of Wireless Access Points (CAPWAP) Protocol Binding for IEEE 802.11", RFC 5416, March 2009.