Research Inventy: International Journal Of Engineering And Science
Vol.3, Issue 2 (May 2013), PP 28-30
Issn(e): 2278-4721, Issn(p):2319-6483, Www.Researchinventy.Com

A Fast Handoff Scheme of Ieee 802.11 Wireless Networks

[1]Jyoti Sachan, [2]Anant Kr. Jaiswal
[1]*(Amity University)*
[2]*(Amity University)*

ABSTRACT *- The Handoff delay is one of the major problems in Wireless Network (WLN) that needs to be solved in order to allow time-critical and real-time applications run continuously during handoff. We have developed a fast handoff scheme called NodeScan (or MobileScan) to provide a novel use of channel scanning latency by employing open system authentication. This scheme comprises two steps: firstly a client device takes advantage of the WMN architecture to maintain a list of active mesh nodes. Secondly when handoff is required, a client transmits Authentication Request frames to all mobile nodes (MNs) from the list instead of broadcasting Probe Request frames as in an active scan to discover the available MNs. This fast handoff scheme is feasible by upgrading the software only on the client side. This paper compares the theoretical handoff latency of Mesh Scan with other approaches and we demonstrate the effectiveness of our through experiment.*

KEYWORDS *– Handoff, Wireless LAN, IEEE 802.11, NodeScan, Mobile Node*

I. INTRODUCTION

A traditional wireless network deployment involves Access Points (AP) with overlapping coverage zones where each AP has a wired network connection. WMNs only require a few of the MNs to have wired network connections and allowing all others to forward packets over multiple wireless hops. In this paper, we are concerned with an IEEE 802.11a network which operates in the 5.3 GHz frequency band. A practical problem with WMNs occurs when a connection transition (handoff) from one MN to another MN is required for mobile client to maintain network connectivity when a mobile client moves away from one MN and closer to another one. Ideally, handoff should be completely transparent to mobile client to support real-time traffic such as interactive VoIP or video conferencing. The handoff procedure aims to reduce this time as much as possible so that the upper layers do not notice the connectivity interruption. However, Under the IEEE 802.11 WLAN standard, there are three steps involved in the handoff process: Discovery, Authentication and Re-association. Previous works reported that the standard handoff incurs latency of the order of hundreds of milliseconds to several seconds. Moreover, the discovery step accounts for more than 90% of this latency [1]. In this paper, we present an experimental fast handoff scheme called NodeScan to reduce the latency associated with node by using open system authentication where no key exchange is involved. The fast handoff scheme uses a client-side control mechanism which requires a client software upgrade. The experiments are performed on an wireless mesh testbed which uses open system key authentication. All measurements are taken from the system kernel layer to ensure accuracy. The basic idea behind NodeScan is to take advantage of the WMN architecture where all the MNs are required to cache a list of mesh node at client side, and exploit multiple Authentication Request frames to find the next mesh node within same mesh network. In this paper, we compare the theoretical handoff time required by NodeScan with other approaches and demonstrate the effectiveness of our system through experiment.

The rest of this paper is organized as follows. In Section II, we describe the background (link-layer handoff procedure in IEEE 802.11 wireless networks) of the handoff and present a discovery latency analysis. Section III introduces the experimental testbed wireless interface driver. In section IV, we describe our experimental method to improve the efficiency of discovery. Section V presents the details of our implementation on Linux box and experimental results. Section VI concludes the paper and outlines our future work in this area.

.

II. BACKGROUND

The Link-layer handoff refers to the change of the mesh node to which a station is connected in a WMN. In the case of IEEE 802.11a WLANs it implies an interruption of data frame transmission. The duration of this interruption is called handoff latency. For the purposes of this work we divide handoff into two phases: Scanning and Execution. The scanning phase is used to acquire information about the available APs in each channel. In the IEEE 802.11 standard, there are two methods used: passive scanning and active scanning. In passive scanning, a mobile client listens for beacon frames on one channel at a time.

Beacon frames are normally broadcast by MN every 100ms. In active scanning, the mobile client broadcasts probe request frames and waits for probe response frames on each channel. The execution phase is the phase when the mobile client exchanges information and establishes a physical connection with the MN. It involves authentication and re-association. Authentication verifies the identity between client and MN. The standard defines two algorithms: open system authentication and shared key authentication. The time required for authentication takes one round trip time (RTT) for open system or twice RTT for shared key. Re-association follows after successful authentication where the client is assigned a proper association identity and required resources by new MN. The re-association delay takes one RTT. RTT is the time corresponding to the transmission time of a probe request frame and an ACK response frame between two nodes. Four timestamps are required to calculate RTT using equation (1):

RTT= (T 21– T11) + (T 22– T12) (1)

In this paper we assume as shown in Figure 1. (T11 is the timestamp of the probe request frame that is transmitted from Node A, T21 is the times that the request frame from Node A is received by Node B, T22 and T21 are similar to T11 and T21) RTT depends on a number of factors that includes the network load, interference and contention.

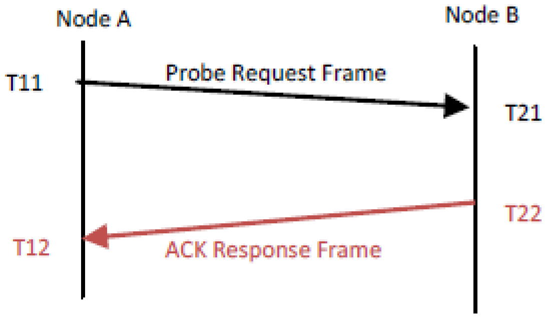

Fig. 1: Round Trip Time

III. SYSTEM DESCRIPTION

All experiments have been carried out using the CNRI wireless mesh testbed [13]. This testbed is a multi-purpose networking experimental platform which consists of 17 IEEE 802.11abg based mesh nodes, located around the Focus building in DIT. Each Mesh Node uses Soekris net 4521 boards as hardware platform and NETGEAR WAG511 wireless adapter cards. It runs under the Pebble Linux distribution as software platform and uses madwifi version 0.9.4 as the wireless network interface driver. Further information about the CNRI mesh testbed can be obtained from http://mesh.cnri.dit.ie.

IV. EXPERIMENTAL SETUP

NodeScan comprises two steps: First the mobile client is given a list of available mesh node information called a SmartList. Secondly, when handoff is required, the mobile client performs a unicast scan by transmitting Authentication Request frames to the each of the MNs on the list to discover the next MN for handoff to. SmartList is where the MN information stores and manages the MNs. The list is ordered where a MN's position on the list depends on its Received Signal Strength Indications (RSSI) value. The MN with the highest RSSI value will be put at the top of the list in order to provide fast handoff to the best available MN in real-time. The MN information can be easily added to and stored onto mobile client when a mobile client joins a particular WMN for the first time. The MN's RSSI is provided in real-time by listening to beacon frame from all Mesh Nodes. NodeScan does not generate any overhead during handoff and so does not produce any communication performance degradation, nor require modifications to the protocols.

V. HANDOFF PROCEDURE

In our system, the handoff procedure is performed with the following steps. When handoff is required the mobile client transmits an Authentication Request to each of the MNs on the SmartList. This is in accordance with the 802.11 standard which allows for authentication with multiple MNs. When the first Authentication Response frame is received, mobile client stops transmitting Authentication Request to the rest of the MNs on the SmartList and re-associates with the MN which sends the first Authentication Response.

In the case where no Authentication Response is received after Authentication Requests have been transmitted to all MNs in the SmartList, the mobile client will perform active scanning to try to discover any available wireless networks. The NodeScan algorithm is shown in Figure 2. In terms of the NodeScan algorithmic delay and assuming at least one MN is available, we have equation (2) where M is the number of Authentication Request frames transmitted. In the best case scenario the first MN from the SmartList is the next MN to re-associate with, so the delay is 2 * RTT. The worst case will be there is no available MN and the mobile client carries out an active scanning.

VI. IMPLEMENTATION DETAILS

We implemented the SmartList within the kernel driver (madwifi 0.9.4) [14]. The changes in madwifi driver are the minimum required to support NodeScan by using SmartList. All processes are carried out in kernel layer to provide stable and fast handoff in this prototype implementation. In the madwifi driver, a new structure is implemented to create single linked list as SmartList. A command line input method (ioctls) is also implemented to provide a flexible way to add Mesh Node information. We created a new information state to trigger SmartList to reorder the MN position to make the best MN on top of SmartList. The management frame retry and management frame backoff count are minimized to provide fast Authentication Request transmission. In the original madwifi driver, the same Authentication Request will be retransmitted 11 times if an Authentication Request does not get response from MN where it waits for 1 second before transmitting the next Authentication Request. They can cause significant delay so the management frame retry is set to zero and management frame backoff count is set to one millisecond when a fast handoff is triggered.

VII. CONCLUSION

We have developed a fast handoff scheme, called NodeScan to improve handoff within WMN, by using a novel usage of the open system authentication scan to reduce channel scanning latency. NodeScan maintains a list of MNs in SmartList, and performs unicast scanning by transmitting authentication request frame to discover available MN and perform handoff instead of broadcasting probe request frame. In this paper we have shown a significant reduction in handoff latency of our approach through implementation and experiments. Our future work will be concerned with implementing real-time list of MNs at the MN side, and the MN provides accurate mesh node list.

REFERENCES

[1] A. Mishra, et al.: "An Empirical Analysis of the IEEE 802.11 MAC Layer Handoff Process, " ACM SIGCOMM Computer Communication Review.

[2] Diane Tang and Mary Baker, "Analysis of a Metropolitan- Area Wireless Network," ACM/Kluwer Wireless Networks. Special issue: Selected Papers from Mobicom'99, vol. 8, no. 2/3, pp. 107-120, 2002.

[3] B. Chambers, “The grid roofnet: a rooftop ad hocwireless network," 2002. h

[4] John C. Bicket, Daniel Aguayo, Sanjit Biswas, and Robert Morris, "Architecture and evaluation of an unplanned 802.11b mesh network.," in MOBICOM, 2005, pp. 31-42.

[5] Locusworld," http://locustworld.com

[6] Tropos networks," http://www.tropos.com

[7] Extricom," http://www.extricom.com

[8] Cisco," http://www.cisco.com/en/

[9] Arunesh Mishra, Minho Shin, and William Arbaugh, "An empirical analysis of the IEEE 802.11 MAC layer hando_ process," SIGCOMM Comput. Commun. Rev., vol. 33, no. 2, pp. 93-102, 2003.

[10] Ishwar Ramani and Stefan Savage, "Syncscan: Practical Fast Hando_ for 802.11 Infrastructure Networks," in Proc. Of IEEE INFOCOM, march 2005.M.

[11] V. Brik, et al.: "Eliminating handoff latencies in 802.11 WLANs using multiple radios: Applications, experience, and evaluation, ACM/USENIX IMC 2005, USA, October 2005.

[12] M. Jeong, et al.: "Fast active scan for measurement and handoff, "Technical report, DoCoMo USA Labs, May 2003.

[13] CNRI Wireless Mesh Testbed. Available at http://www.cnri.dit.ie/research.mesh.testbed.html

[14] MadWifi - a Linux kernel device driver for Wireless LAN