Measurements of Multicast Service Discovery in a Campus Wireless Network

Se Gi Hong, Suman Srinivasan and Henning Schulzrinne
Columbia University, New York, NY
{segihong, sumans, hgs}@cs.columbia.edu

***Abstract*—Applications using multicast service discovery protocols, such as iTunes, have become increasingly popular. However, multicast service discovery protocols generate significant network traffic overhead, especially in a wireless network. We measure and analyze the traffic of one of the most widely deployed multicast service discovery protocols, DNS-SD/mDNS, in a campus wireless network where a single multicast domain serves a large number of users. We define three service discovery models which correspond to different service discovery behaviors. We compare these three models in terms of packet overhead and service discovery delay for different network sizes and service lifetime assumptions. The measurement shows that mDNS traffic consumes about 13 percent of the total bandwidth of the network.**

***Index Terms*—Service discovery, multicast, measurements.**

I. Introduction

Large-scale wireless networks have become increasingly common as WiFi-enabled mobile devices gain popularity. Service discovery protocols are needed so that users of wireless devices can find peer services. DNS-based service discovery (DNS-SD) [1] that leverages multicast DNS (mDNS) [2] is among the most widely deployed service discovery protocols. One of the most popular applications that use mDNS is iTunes [3], a multimedia application, which allows users to browse playlists of other iTunes users in the same subnet.

There is a surge in the popularity of applications that use mDNS. mDNS generates significant traffic overhead. Such overhead may especially be seen on college campuses, where wireless networks are pervasive and a large number of wireless users are able to work on the same subnet. Because of the performance impact of mDNS traffic, the Office of Information Technology at Princeton University filters mDNS packets [4].

We are not aware of any previous measurement and analysis of the overhead of multicast service discovery traffic in a campus wireless network. Therefore, we measured and analyzed the performance impact of mDNS traffic in a typical college wireless network to show how much traffic overhead mDNS generates. We show the bandwidth usage of mDNS packets and the effect of multiple APs on multicast packets in Columbia University's wireless network.

We define three service discovery models, which correspond most closely to the DNS-SD/mDNS traffic behavior. We analyze these different service discovery models in terms of traffic overhead and service discovery delay for different network sizes and lifetimes.

The measurement and analysis provides insights in choosing a proper service discovery models that make different trade-offs in different environments.

The remainder of this paper is structured as follows. In Section II, we discuss the overview of multicast service discovery. Section III describes how our measurement was set up. Section IV outlines our findings about mDNS traffic in a campus environment, in terms of the number of packets and their impact on wireless traffic. In Section V, we analyze different service discovery models. Finally, we describe related work in Section VI.

II. Multicast service discovery basics

There are two protocols for host naming on a local network without a central DNS server: mDNS and Link-local Multicast Name Resolution (LLMNR) [5]. LLMNR is currently implemented only in Windows Vista and Windows CE, but mDNS is implemented on Windows, Linux and Mac OS X, and other POSIX platforms, and Java implementation [6] is available as well. Multicast DNS translates between a local host name and IP address without a central DNS server. This local host name selected by each device is meaningful only on the local network. DNS-based service discovery that leverages mDNS allows users to announce their services and discover peer services. The service discovery protocol uses three record types; namely the PTR, SRV and TXT records. The PTR record is used to discover service instances on the local network. The SRV record provides port number and IP addresses of the service providers. The TXT record provides additional information about services as attribute-value pairs.

The most commonly used implementation of DNS-SD/mDNS is an implementation by Apple Inc. called Bonjour [7].

III. Measurement setup

Since Columbia University's wireless network (IEEE 802.11 b/g) is organized as a single subnet and mDNS works within a subnet, all mDNS packets are transmitted to all wireless users who use mDNS in the campus. We use two sniffing tools, mDNSNetMonitor and Wireshark 0.99.6 [8], to measure mDNS traffic in this network. Apple provides mDNSNetMonitor along with the Bonjour source code to record the patterns of mDNS protocol usage. We use Wireshark to analyze the raw mDNS packets and other multicast packets to obtain more details.

Each measurement lasted for two minutes. We selected a measurement time from 2 PM to 5 PM, which is the busiest

978-1-4244-4148-8/09/$25.00 ©2009
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE "GLOBECOM" 2009 proceedings.

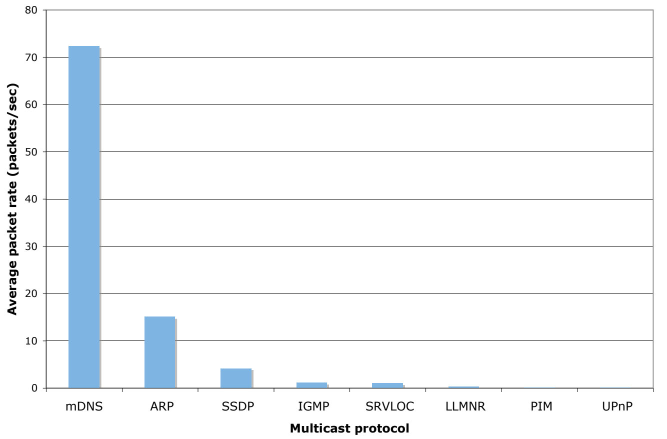


Fig. 1. A comparison of the average rate of multicast and ARP packets on the network that we measured traffic on.

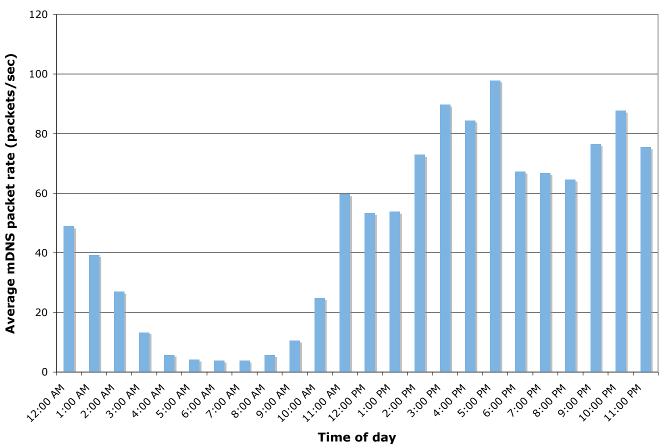


Fig. 2. The average mDNS packet rate on a weekday

time of user activity on campus. The machine we used for sniffing has 1 GB of RAM and a 1.66 GHz Core 2 CPU running Linux. The PCMCIA wireless card used for sniffing is a Orinoco 11a/b/g card with the Atheros chipset. We used the network card interface in monitoring mode to capture all packets over-the-air.

IV. mDNS in a Campus Wireless Network

Fig. 1 shows the average multicast and ARP packet rate categorized by protocols. It shows that the majority of multicast packets in the network are mDNS packets. LLMNR packets make up a very small portion of multicast packets. The figure also shows that there are other service discovery protocols (SRVLOC, SSDP and UPnP), but the number of those service discovery packets is very small. Therefore, we will mainly focus on mDNS. We also measure the number of ARP packets since ARP packets are one of the most popular broadcast packets in the network. As we can see, the number of mDNS packets was much higher than that of ARP packets.

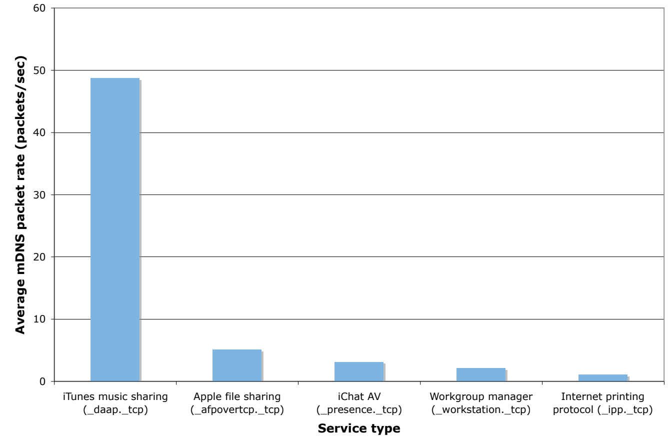


Fig. 3. The average mDNS packet rate by service types

TABLE I
Average number of mDNS users seen in networks during measurement when the total of 944 users are seen in the network

Service type	number of users
iTunes music sharing (_daap._tcp)	188
Apple file sharing(_afpovertcp._tcp)	83
iChat AV (_presence._tcp)	25
Workgroup manager (_workstation._tcp)	49
Internet printing protocol (_ipp._tcp)	16

A. *Number of mDNS packets*

Fig. 2 shows the average mDNS packet rate on a weekday. The times of 2 PM-6 PM and 9 PM-12 PM are the busiest times of the day when users use mDNS-based networking applications. Fig. 3 shows the average mDNS packet rate by service type. We show the five major applications based on the number of mDNS packets. It shows that the majority of mDNS packets are generated by the iTunes (_daap._tcp) application. Table I shows the average number of users seen on the networks during the measurement. As we can see from the table, the number of iTunes users is the largest. In the service resolution process of iTunes, iTunes sends SRV and TXT query records to all iTunes users, and responds with SRV and TXT records. The records are periodically sent with a delay based on an exponential backoff algorithm mentioned in the mDNS specification. The period starts from one second and goes up to one hour. When the period reaches one hour, the exponential backoff halts, and messages are sent every hour after this.

This service resolution process generates many mDNS packets. In other applications, even though the process of querying and responding for SRV and TXT record processing exists, the process is rarely performed since users do not actively use the application.

B. *Channel utilization of mDNS packets on wireless networks*

One multicast packet occupies the channel twice. During the uplink transmission of a mDNS packet to an AP, the transmission to an AP is unicast. However, on the downlink transmission of mDNS packets from an AP to many multicast

978-1-4244-4148-8/09/$25.00 ©2009
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE "GLOBECOM" 2009 proceedings.

users, the transmission is multicast. Therefore, all the uplink transmission rates depend on the users' network interfaces. Since most users use laptop computers with IEEE 802.11 g wireless cards, most of the uplink transmission follows IEEE 802.11 g. However, the downlink transmission rate is fixed to 11 Mb/s since in Columbia University's wireless network, an AP has to match the transmission rate with the lowest maximum transmission rate between IEEE 802.11 g and b even though most users use a IEEE 802.11 g wireless card. The maximum transmission rates are 54 Mb/s and 11 Mb/s for IEEE 802.11 g and b, respectively.

We calculate the bandwidth usage of mDNS packets by the channel utilization, ρ. The utilization, ρ, is the ratio of the sum of all the busy periods of mDNS packets to a unit time, U.

$$\rho = \frac{T_D + T_U}{U},$$

where T_D and T_U are the total downlink and uplink transmission time of mDNS packets per unit time including all overhead. The receiver can synchronize the incoming signal by the physical layer convergence protocol (PLCP) preamble before receiving actual data, and the header provides information of the frame [9]. Therefore, we include the PLCP for performance measurement. The downlink transmission time, T_D, is

$$T_D = (T_{DIFS} + T_d) \times N_d$$

$$T_d = \frac{L_m}{R_d} + T_{PLCP},$$

where T_{DIFS} is the time length of DIFS, T_d is the transmission time of a mDNS downlink packet, and N_d is the average number of mDNS downlink packet in unit time. L_m is the average mDNS packet size including MAC layer header, R_d is the downlink transmission rate, and T_{PLCP} is the time length of the PLCP preamble and header. Table II shows the parameters of IEEE 802.11 b/g.

The uplink transmission time, T_U, is

$$T_U = (T_{DIFS} + T_{SIFS} + T_u + T_{ACK}) \times N_u$$

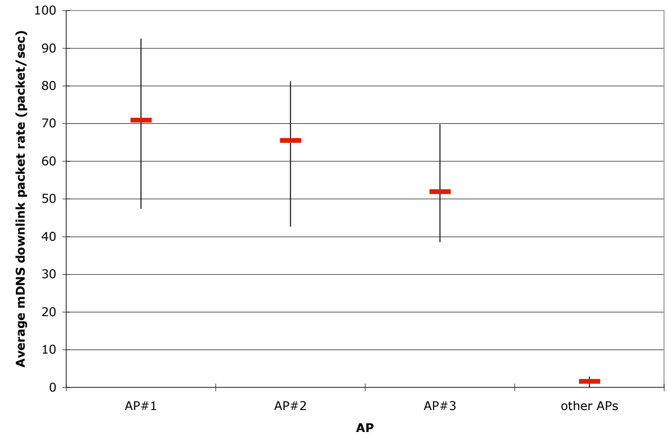
$$T_u = \frac{L_m}{R_u} + T_{PLCP}$$

$$T_{ACK} = \frac{L_c}{R_c} + T_{PLCP},$$

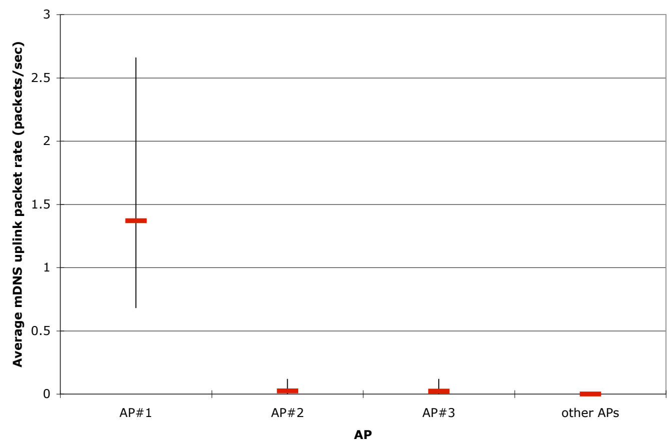
TABLE II
PARAMETERS IN IEEE 802.11 B/G

Parameters	Size (bits)	Tx rate (Mb/s)	Tx time (μs)
PLCP Preamble	144	1	144
PLCP Header	48	1	48
SIFS	-	-	10
DIFS	-	-	50
ACK	112	R_c	$112/R_c$

where T_{SIFS} is the time length of SIFS, T_u is the transmission time of the mDNS uplink packets, and N_u is the average number of mDNS uplink packet in unit time. R_u is the uplink



(a) Average rate of mDNS downlink packets associated with co-channel APs



(b) Average rate of mDNS uplink packets associated with co-channel APs

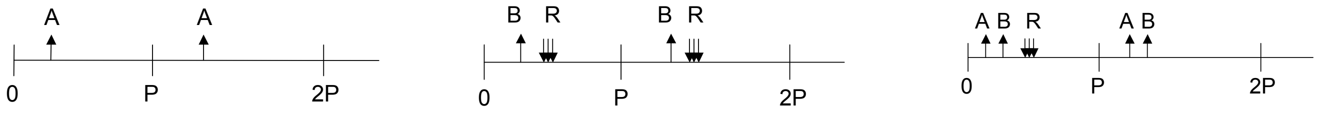
Fig. 4. Average rate of mDNS packets associated with co-channel APs

transmission rate. T_{ACK} is the transmission time of control frames, ACK. L_c is the length of control frames, ACK. R_c is the transmission rate of ACK. Our measurements show that the transmission rates of ACK from APs are the same as data transmission rates from users.

C. The effect of multiple APs on the same channel

A station can typically see multiple APs on the same channel, which are called co-channel APs. However, the station associates with only one of the APs. Each of these APs sends out the same multicast packet to the stations associated with them. Therefore, even though a station receives packets from an associated AP, packets from all co-channel APs consume bandwidth at the station. Fig. 4(a) and Fig. 4(b) shows the measured maximum and minimum rates as well as the average rate of the captured downlink and uplink mDNS packets at several APs. As we can see, the same multicast packets from several APs are captured by a station. When we calculate the utilization, ρ, we should consider this co-channel effect. Therefore, the average rate of mDNS downlink and uplink packets is the sum of all downlink and uplink packets from the co-channel APs. The values of N_d and N_u are 190 packets/sec

978-1-4244-4148-8/09/$25.00 ©2009
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE "GLOBECOM" 2009 proceedings.



(a) Periodic announcements (b) Periodic browsing (c) Periodic announcements and browsing

Fig. 5. The service announcements and browsing models (A: service announcements, B: service browsing, R: responding, P: period)

and 1.4 packets/sec, respectively. The reason for the difference between the two values is that on the downlinks, the packets come from all users in the campus wireless network, but on the uplinks, the packets come only from the users in the small local area. In our measurement, L_m is 578 bytes. Therefore, the utilization, ρ is 0.13. Since other packets cannot be transmitted during the transmission of mDNS packets, we can say that mDNS traffic consumes 13 percent of the total bandwidth.

If we do not consider the co-channel effect (i.e., consider only the packets associated with AP#1), the value of N_d and N_u are 70.9 packets/sec and 1.3 packets/sec, respectively. Therefore, the utilization, ρ is 0.05. Therefore, the effect of mDNS packets for bandwidth usage varies depending on the number of APs on a channel.

These measurement results could be taken to be specific to the wireless network in our campus network. But we argue that if the measurement is performed at other places and college campuses and if the network size (number of users and number of APs) is similar, the results will be similar. Furthermore, since only three channels (channel 1, 6, and 11) are used in IEEE 802.11 b/g, there will always be multiple APs on the same channel and interference will occur between the APs. If the network size is different, it might be possible to extrapolate our results by scaling our results according to network size.

V. Comparison of service discovery models

To see the trade-offs between different service discovery models in terms of packet overhead and service discovery delay under different network sizes and lifetimes, we define and analyze three different service discovery models: periodic announcements, periodic browsing, periodic announcements and browsing. We do not consider co-channel effects since the number of co-channel APs and the signal strength of the APs could vary in different places. Therefore, we only consider the traffic that belongs to one AP. We do not consider the mechanisms used in the mDNS protocol to reduce packets, such as aggregation of several answers into one packet. Therefore, the models in this analysis are similar but not exactly the same as multicast DNS service discovery protocol. The objective of this analysis is to compare different service discovery models under different network sizes and lifetimes. This analysis provides insights in choosing proper models that make different trade-offs in different environments.

We assume that users join with Poisson distribution with rate λ. We assume that the service announce, browse and response packets are transmitted by multicast, and the system is in steady state. The average number of users in the network is N and the average number of users associated with each AP is N_a. The service announce and browse period is P.

A. Model A: Periodic announcements

Fig. 5(a) shows the periodic announcements model. In this model, users discover services only by the periodic announcements (A) of other users. Therefore, there are no browse and response packets. On the uplinks, N_a users associated with an AP send announce packets. Thus, the average uplink rate is N_a/P. On the downlinks, the AP transmits announcement packets from all N users to users associated with that AP. Thus, the average downlink rate is N/P.

Since the service discovery is accomplished only through announcements, the maximum service discovery delay is the period, P. When we assume that distribution of the service announcements follows uniform distribution, the mean service discovery delay to find one particular service is $P/2$. The mean service discovery delay to find all services is $P-P/(2(N-1))$. Thus, as N increases, the mean service discovery delay to find all services goes to the finite period, P.

B. Model B: Periodic browsing

Fig. 5(b) shows the periodic browsing model. In this model, users discover services by periodic browsing (B) and the response to the browsing requests of other users. Therefore, there are no announce packets. On the uplinks, N_a users browse a service. Thus, the rate of uplink browse packets is N_a/P. Each user responds to the browse of other users (that are in the same AP) and other users (that are in the different APs). Thus, the average rate of uplink response packets is $(N_a-1)N_a/P + N_a(N-N_a)/P$. On the downlinks, the AP transmits the service browse packets from all N users. Thus, the rate of downlink browse packets is N/P. Since users response to the browse packets from all other users, the rate of downlink response packets is $(N-1)N/P$.

The service discovery delay in this model is only one network round trip time because when a newly joined user browses for a service, others respond immediately. Therefore, this delay is very small.

C. Model C: Periodic announcements and browsing

Fig. 5(c) shows the periodic announcements and browsing model. In this model, users discover services both by periodic announcements (A) and periodic browsing (B) of services. The response packets are transmitted only when a new node joins

978-1-4244-4148-8/09/$25.00 ©2009

TABLE III
THE AVERAGE RATE OF SERVICE DISCOVERY PACKETS WITH DIFFERENT MODELS

Models	Types of uplink packets			Types of downlink packets		
	Announce	Browse	Response	Announce	Browse	Response
Model A	$N_a\frac{1}{P}$	0	0	$N\frac{1}{P}$	0	0
Model B	0	$N_a\frac{1}{P}$	$N_a(N_a-1)\frac{1}{P}+N_a(N-N_a)\frac{1}{P}$	0	$N\frac{1}{P}$	$(N-1)N\frac{1}{P}$
Model C	$N_a\frac{1}{P}$	$N_a\frac{1}{P}$	$(N_a-1)\frac{N_a}{N}\lambda+N_a\frac{N-N_a}{N}\lambda$	$N\frac{1}{P}$	$N\frac{1}{P}$	$(N-1)\lambda$

the network and browses services. After initially responding to the service browsing of new nodes, users do not send response packets for service browsing requests until other new nodes join the network. The number of response packets depends on the arrival rate of new nodes, λ. Therefore, the rate of announce and browse packets is the sum of model A and model B. On the downlinks, AP transmits all responses from $N-1$ users. Thus, the rate of downlink response packets is $(N-1)\lambda$. On the uplinks, there are two cases: the newly joined user is associated with the AP and the newly joined node is associated with one of the other APs. The probability that a newly joined node is associated with the current AP is N_a/N when we assume that selecting an AP which a new node is associated with follows uniform distribution. Therefore, the rate of uplink response packets is $\lambda(N_a-1)N_a/N+\lambda N_a(N-N_a)/N$.

The service discovery delay in this model is also very small. The reason is the same as the one for the model B.

Table III shows the average rate of service discovery packets with different models by types of packets.

D. Comparison of models

From Little's law, $N = \lambda T$ where T is the average lifetime users spend in the network, we can calculate the arrival rate, λ, given the average number of users (N) in the network and their average lifetime (T) in the network. We vary the lifetime of users from 10 minutes to 1 hour. We vary the average number of users in the network (N) from 100 to 500. We assume that the average number of users associated with an AP is 10 ($N_a = 10$). Fig. 6 shows the average service discovery packet rate of three different models when the period (P) is 30 minutes. As we can see in this figure, model A generates the lowest number of service discovery packets. In model A, the average rate of service discovery packets increases slightly as N increases. The service discovery delay of model A is the highest since the service discovery only depends on the announcements from users. The worst case of the service delay of model A is P (30 minutes). The mean service discovery delay to find a particular service is 15 minutes and the mean service discovery delay to find all services is 29.95 minutes (when $N = 300$). The delay of the two other models is very small since existing nodes immediately respond to the service browsing of nodes that have just joined. The number of service discovery packets in model C depends on the lifetime of users. When the lifetime is 10 minutes, the number of service discovery packets is the largest. When we compare model B and model C (which has a lifetime of 30 minutes), the result is almost the same. In model C, a new node joins the network every 30 minutes, and in model B, the browsing period is 30 minutes. Therefore, the result is the same.

Choosing the proper model is based on the network environments and characteristics. In terms of low service discovery delay, models B and C are the best choice. In terms of low traffic overhead, model A is the best choice. If the average lifetime of nodes is high, model C gives low traffic overhead and low service discovery delay. If the average lifetime of nodes is low, model C generates high traffic overhead.

VI. RELATED WORK

Many papers present the measurement and analysis of network packets in large area wireless networks [10] [11] [12]. However, their main considerations are network traffic pattern and user behavior in networks. They do not analyze mDNS and multicast packets. Handerson et. al [13] present the characteristic of wireless network traffic by applications, including iTunes, but this paper does not show the details of iTunes, which uses the mDNS protocol.

Our previous work [14] presents the delay and service loss probability of service discovery in ad-hoc Zero Configuration Networking, but it does not show the measurement and analysis of service discovery packet overhead.

Furthermore, there are no papers at present which perform the measurement and analysis of mDNS packet overhead in a campus wireless network.

VII. FUTURE WORK

We believe that the findings from our measurement might seem surprising since they have not been reported before. Hence, measuring traffic in various other network environments would produce interesting results. Therefore, taking measurements at various other environments, such as at conferences where there is usually network congestion, might be interesting, and could be basis for future work for this project.

VIII. CONCLUSION

mDNS uses multicast, so it generates significant traffic overhead and consumes network resources, especially in a campus network. However, there has been no formal measurement of mDNS in large wireless networks, so we do not really know how much traffic overhead mDNS generates.

We define and analyze three different service discovery models which correspond most closely to DNS-SD/mDNS:

978-1-4244-4148-8/09/$25.00 ©2009
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE "GLOBECOM" 2009 proceedings.

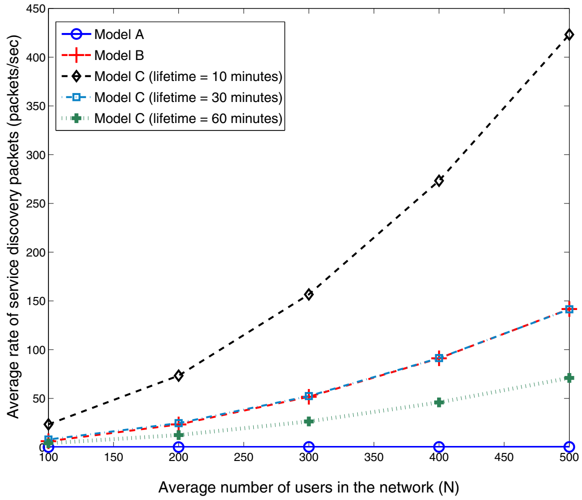


Fig. 6. The average service discovery packet rate of model A, B and C when period (P) is 30 minutes. The average rates are obtained from the Table III.

periodic announcements, periodic browsing, and periodic announcements and browsing models. The analysis shows the average rate of service discovery packets of different models under different network sizes and lifetimes. We found that model A generates lowest traffic overhead. Models B and C have less service discovery delay.

We evaluated the network traffic overhead (bandwidth usage) of mDNS packets by measurements. Our measurement shows that the current overhead of mDNS packets is not severe as mDNS traffic consumes 13 percent of the total bandwidth. In congested networks, however, such as those at IETF meetings or conferences, this overhead can have adverse effect on the performance of other network services. Furthermore, this consumption of bandwidth is mostly due to one popular application, iTunes, as about 69 percent of mDNS packets are iTunes packets. This means that if there are other popular applications which work in a manner similar to iTunes, or if the number of users using iTunes increases, the overhead of mDNS traffic will increase even more.

Our measurement and analysis of different service discovery models can provide insights for making design choice with different trade-offs in terms of traffic overhead and service discovery delay for different network sizes and lifetimes.

References

[1] S. Cheshire and M. Krochmal, "DNS-Based Service Discovery," Internet Engineering Task Force, Internet Draft, September 2008, work in progress.

[2] S. Cheshire and M. Krochmal, "Multicast DNS," Internet Engineering Task Force, Internet Draft, September 2008, work in progress.

[3] Available: http://www.apple.com/itunes/, iTunes.

[4] Available: http://www.net.princeton.edu/filters/mdns.html/, office of Information Technology at Princeton University.

[5] B. Aboba, D. Thaler, and L. Esibov, "Link-Local Multicast Name Resolution (LLMNR)," Internet Engineering Task Force: RFC4795, January 2007.

[6] Available: http://jmdns.sourceforge.net/, JmDNS.

[7] Available: http://developer.apple.com/networking/bonjour/, Bonjour.

[8] Available: http://www.wireshark.org/, Wireshark.

[9] IEEE Computer Society LAN MAN Standards Committee, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," *IEEE Std. 802.11*, pp. 123–151, 1997.

[10] R. Hutchins and E. Zegura, "Measurements from a campus wireless network," in *Proceedings of IEEE International Conference on Communications (ICC)*, New York, NY, 2002.

[11] M. Ploumidis, M. Papadopouli, and T. Karagiannis, "Multi-level application-based traffic characterization in a large-scale wireless network," in *Proceedings of IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, New Port Beach, California, 18-21 June 2007.

[12] D. Tang and M. Baker, "Analysis of a metropolitan-area wireless network," in *Proceedings of Mobile computing and networking (MobiCom)*, Seattle, Washington, 1999.

[13] T. Henderson, D. Kotz, and I. Abyzov, "The changing usage of a mature campus-wide wireless network," in *Proceedings of Mobile computing and networking (MobiCom)*, Philadelphia, PA, 2004.

[14] S. Hong, S. Srinivasan, and H. Schulzrinne, "Accelerating Service Discovery in Ad-hoc Zero Configuration Networking," in *Proceedings of IEEE Global Communications Conference (GLOBECOM)*, Washington D.C., November 2007.

978-1-4244-4148-8/09/$25.00 ©2009
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE "GLOBECOM" 2009 proceedings.