

# Measuring Wireless LANs

TRISTAN HENDERSON and DAVID KOTZ

Department of Computer Science, Dartmouth College, Hanover, New Hampshire

## 1.1 INTRODUCTION

Wireless local area networks (WLANs) have appeared in many venues, including academic and corporate campuses, residences, and wireless “hotspots.” It becomes increasingly important to understand how these networks are used, as they continue to appear in more numerous and varied environments. Measuring and collecting data from production WLANs in a usage study is one way of fulfilling this need for understanding.

Wireless usage studies and usage data are valuable for many aspects of wireless network research. Understanding how and where clients use the network, what applications clients are using, and how applications are using the network can help with network provisioning and deciding where to expand or augment coverage in an existing WLAN. Models of wireless application workloads can aid the design of future network protocols. Measurements of client mobility in a WLAN can help with the design of location-aware applications, or for developing and improving mobile handoff algorithms.

Collecting data on a WLAN can be difficult, however. There are many technical and nontechnical logistical hurdles involved in collecting high-quality wireless measurements. We have been continuously monitoring a campus WLAN for over 3 years in the course of conducting two of the largest wireless measurement studies to date [9,13], and we have encountered many of these hurdles. In this chapter we describe some of the tools that the research community has used for measuring WLANs, and provide hints for their effective use obtained from our real-world experiences. We also discuss some of the usage studies that have been conducted using these tools, both on our own campus and elsewhere. In particular we concentrate

on the most common type of wireless LAN, the IEEE 802.11 infrastructure network, as this has seen the highest number of deployments, and thus most usage studies have considered infrastructure networks.

This chapter is laid out as follows. In Section 1.2 we examine some of the tools that are available for measuring a WLAN. Section 1.3 surveys various wireless measurement studies, considering both the tools that were used and the insights that were learned. Section 1.4 concludes the chapter with a checklist of items that a potential wireless usage researcher should consider.

## 1.2 MEASUREMENT TOOLS

The purpose of a wireless usage study is to collect data about the operations of a WLAN. There are several tools available to the researcher for this purpose. The most commonly used tools include syslog, SNMP, network sniffing, authentication logs, and developing client-side applications. Figure 1.1 shows how some of these tools might be deployed in an example WLAN. In this section, we summarize the pros and cons of using each of these tools, and offer some advice from our own experiences.

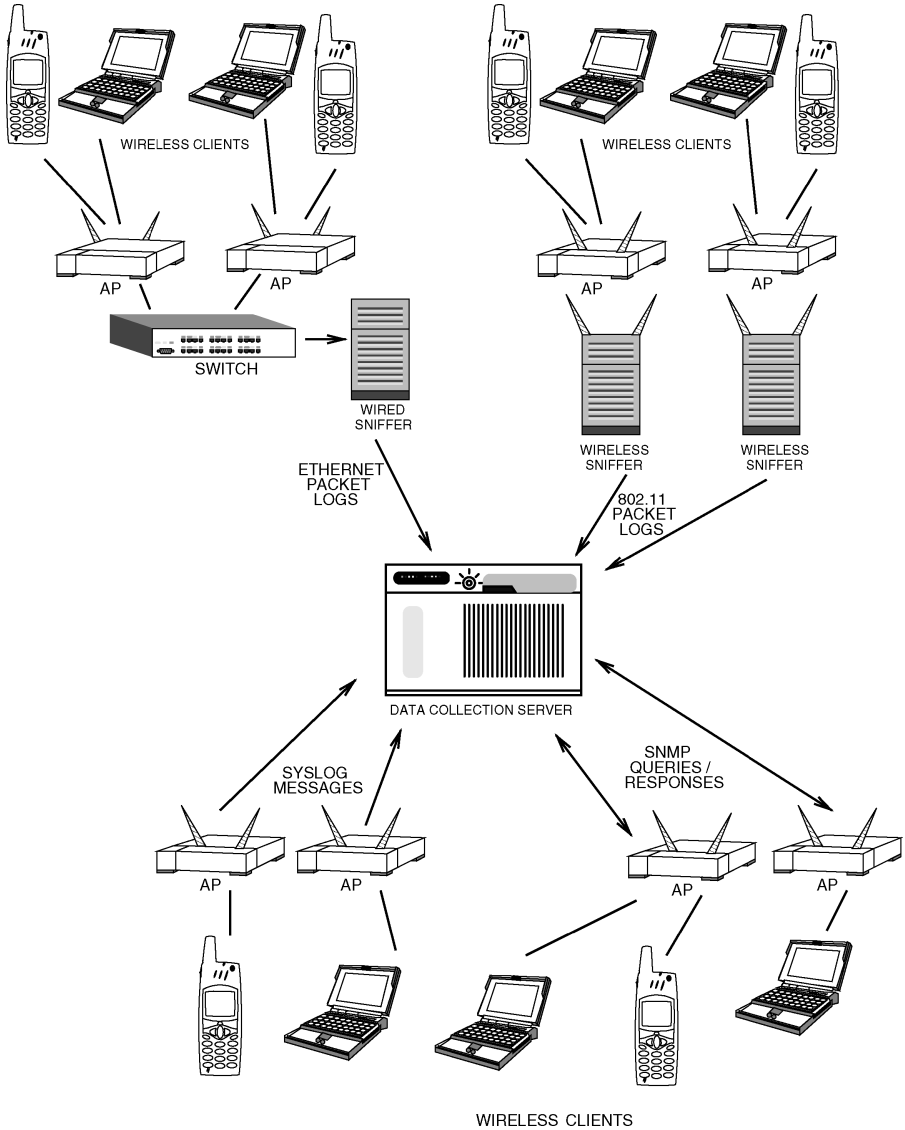
### 1.2.1 Syslog

Syslog is a somewhat loosely specified standard [14] for sending and receiving logging messages. Messages can be stored locally or transmitted across a network to another host.

Many 802.11 access points (APs) can be configured to send syslog messages. By choosing appropriate events to be logged, syslog messages can be used to understand the state of clients on the network. For instance, an AP can send a time-stamped syslog message whenever a client authenticates, deauthenticates, associates, disassociates, or roams to that AP. By collecting these syslog messages from all of the APs in a network, it is possible to determine the state of the clients on the network.

Once an AP has been configured to send syslog messages to a particular host, no further information is required from the receiving host. This makes syslog a simple tool to set up. The receiving host, however, must take care to ensure that messages are being received correctly, as network problems, firmware upgrades, or malfunctioning APs, may lead APs failing to send syslog messages.

There is no standard format for a syslog message, and there is also no standard format for an 802.11 syslog message. The messages that APs send can vary in format, and in the amount of information that is contained. Figures 1.2 and 1.3 show two sets of syslog messages. These messages are both taken from the same Cisco Aironet 350 802.11b AP. Figure 1.2 shows messages from the AP when it was running the VxWorks operating system, whereas Figure 1.3 is a set of messages from the AP after it had been upgraded to the Cisco Internetworking Operating System



**Figure 1.1** Tools for measuring a wireless LAN.

```

Jan 1 04:54:27 example1-ap example1-ap (Info): Station 1234567890ab Reassociated
Jan 1 04:54:27 example2-ap example2-ap (Info): Station 1234567890ab roamed
Jan 1 04:55:22 example3-ap example3-ap (Info): Station 0987654321ef Reassociated
Jan 1 04:55:26 example4-ap example4-ap (Info): Station 0987654321ef Reassociated
Jan 1 04:57:23 example5-ap example5-ap (Info): Deauthenticating abcdef123456, reason "Inactivity"
    
```

**Figure 1.2** Example of Cisco VxWorks AP syslog.

```

Jan 1 04:57:58 example1-ap 382: example1-ap:Jan 1 08:57:57: %DOT11-6-DISASSOC: Interface \
Dot11Radio0, Deauthenticating Station 1234.5678.90ab Reason: Disassociated because \
sending station is leaving (or has left) BSS
Jan 1 04:58:01 example2-ap 36723: example2-ap:Jan 1 08:58:00: %DOT11-6-DISASSOC: Interface \
Dot11Radio0, Deauthenticating Station abcd.ef12.3456 Reason: Previous authentication \
no longer valid
Jan 1 04:58:01 example3-ap 13031: example3-ap:Jan 1 08:58:00: %DOT11-6-DISASSOC: Interface \
Dot11Radio0, Deauthenticating Station 0987.6543.12fe Reason: Disassociated because \
sending station is leaving (or has left) BSS
Jan 1 04:58:08 example2-ap 36724: example2-ap:Jan 1 08:58:07: %DOT11-6-ASSOC: Interface \
Dot11Radio0, Station abcd.ef12.3456 Associated KEY_MGMT [NONE]
Jan 1 04:58:10 example4-ap 6882: example4-ap:Jan 1 08:58:09: %DOT11-6-DISASSOC: Interface \
Dot11Radio0, Deauthenticating Station 0004.2356.5b74 Reason: Previous authentication \
no longer valid

```

**Figure 1.3** Example of Cisco IOS AP syslog.

(IOS). Both sets of messages contain the same basic information: client 802.11 events. They differ, however, in the way that this information is presented; in Figure 1.3 there are multiple timestamps (from the syslog daemon and the AP itself), and the client MAC addresses are formatted differently. Parsing syslog messages can therefore be a tedious process, as the format can change between different AP firmware versions. A long-term measurement study should monitor syslog messages for format changes, and also monitor changes in firmware, either through close communication with network administrators, or by using SNMP (see Section 1.2.2).

A further consideration when parsing AP syslog messages is that not all messages may accurately correspond to 802.11 events. Figure 1.4 shows a set of syslog messages from a “wireless switch.” This switch is representative of the newest type of 802.11 infrastructure network, where “dumb” APs are deployed across the area to be covered, and a centralized switch handles authentication, association, and access control. In this setup is the switch that sends syslog messages, not the APs. Rather than sending an individual message for each authenticate, associate, roam, disassociate, and deauthenticate event, the switch sends only two types of message: “station up” and “station down.” The types of message available from the APs in the WLAN to be measured may impact the suitability of syslog as a measuring tool, depending on the type of data required for the study.

```

Jan 1 03:11:48 wireless-switch.example.com 2004 [1874327] auth[30927]: <INFO> \
station up <01:23:45:67:89:0a> bssid 00:11:22:33:44:55, essid Example_ESSID, vlan 12, \
ingress 4226, u_encr 1, m_encr 1, loc 156.1.1 slotport 4035
Jan 1 03:14:04 wireless-switch.example.com 2004 [1874341] auth[30927]: <INFO> \
station up <09:87:65:43:21:fe> bssid 00:11:22:44:55:66, essid Example_ESSID, vlan 12, \
ingress 4258, u_encr 1, m_encr 1, loc 2.2.1 slotport 4035
Jan 1 03:14:07 wireless-switch.example.com 2004 [1874345] auth[30927]: <INFO> \
station up <09:87:65:43:21:fe> bssid 00:11:22:44:55:66, essid Example_ESSID, vlan 12, \
ingress 4258, u_encr 1, m_encr 1, loc 2.2.1 slotport 4035
Jan 1 03:14:40 wireless-switch.example.com 2004 [1874359] auth[30927]: <INFO> \
station up <12:34:56:78:90:ab> bssid 00:11:22:55:66:77, essid Example_ESSID, vlan 12, \
ingress 4262, u_encr 1, m_encr 1, loc 156.4.1 slotport 4035
Jan 1 03:14:47 wireless-switch.example.com 2004 [1874369] auth[30927]: <INFO> \
station up <12:34:56:78:90:ab> bssid 00:11:22:66:77:88, essid Example_ESSID, vlan 12, \
ingress 4296, u_encr 1, m_encr 1, loc 156.4.1 slotport 4035

```

**Figure 1.4** Example of wireless switch syslog.

```

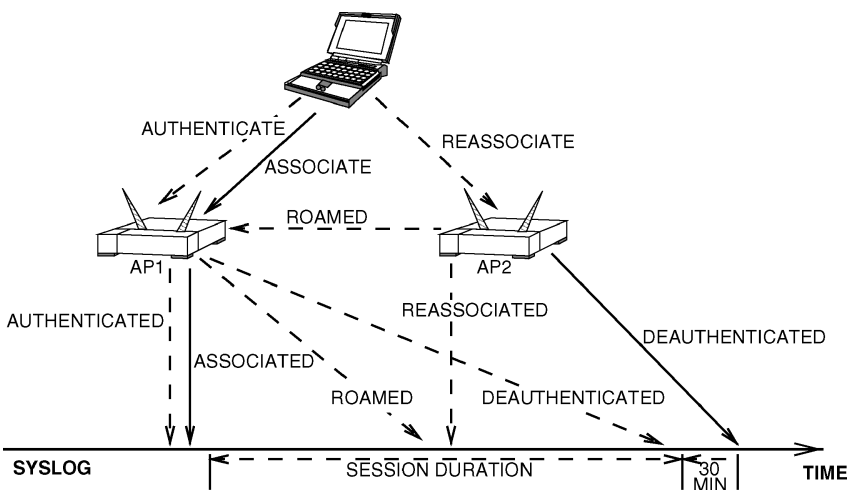
1072933205 0123456789ab roamed example1-ap
1072933214 0123456789ab disassociated example1-ap
1072933215 0123456789ab reassociated example1-ap
1072933241 09876543e1ef deauthenticated example2-ap
1072933244 09876543e1ef authenticated example2-ap
1072933244 09876543e1ef reassociated example2-ap
1072933265 0123456789ab roamed example1-ap
1072933269 0123456789ab disassociated example1-ap
1072933270 0123456789ab reassociated example1-ap
1072933307 abcdef123456 reassociated example3-ap

```

**Figure 1.5** Parsed syslog messages.

In a mixed AP environment such as ours, with multiple types of AP and thus multiple types of syslog messages, we have found it useful to translate syslog messages into an intermediate format prior to data analysis. Figure 1.5 shows this intermediate format. The time, client MAC address, event, and AP hostname are extracted from the syslog messages. The year is added to the time, as syslog messages do not contain a year, and the time is replaced with a Unix timestamp. Some syslog messages contain only the MAC address of an AP and not the hostname, as in Figure 1.4 (e.g., `bssid 00:11:22:33:44:55`). For these APs, we keep a separate mapping of AP names to AP MAC addresses, and refer to this when translating syslog messages.

Once the syslog messages have been collected and translated into a parsable format, it is possible to create a state machine that can calculate a session for each MAC address observed in the syslog trace. Figure 1.6 shows the session state machine that we have used in our campus wireless traces [9,13]. A session consists of an association, followed by zero or more roam events, and ends with a disassociate or deauthenticate event.



**Figure 1.6** The structure of a 802.11 session.

This session structure assumes that a MAC address corresponds to a unique user. This may not be the case in some network environments, for instance, where 802.11 network interface cards (NICs) are shared among several users, or where users tend to alter their MAC addresses. If this is likely to be the case, and the purpose of the study is to track individual's usage, combining syslog data with other data such as authentication logs (Section 1.2.3) may be required.

Our final hint for dealing with syslog messages is to be conscious of holes in the data. As most syslog daemons use a UDP transport, some messages may be lost or misordered in the network. Additional messages may be lost as a result of changes in network configuration or malfunctions. These holes can lead to errors in the estimation of a session length. For instance, if a disassociate message is lost, a simple parser may assume that a client has never disassociated from their last observed AP, and so overestimate the session length. In our studies, we have attempted to alleviate this problem by looking for sessions that are still active at the end of our trace. We assume that these sessions are missing a disassociate message, and we manually terminate the sessions 30 min after the last syslog message recorded for this MAC address. We chose a 30-min window since this is the usual period that an AP uses to time out inactive clients. Advantages and disadvantages of syslog are as follows:

*Pros* — somewhat passive (no additional traffic sent to APs); one-second granularity

*Cons* — no common data format; UDP transport means that messages can be lost; may need to manually configure every AP to send syslog messages

### 1.2.2 SNMP

As its name implies, the Simple Network Management Protocol (SNMP [15]) is a means for managing network devices, or more generally, network objects. A network administrator runs a tool known as a manager, which communicates with SNMP agents. Agents run on network devices, and provide an interface between the device and the manager. A network device can contain several managed objects, such as statistics or configuration items, arranged in a database known as a Management Information Base (MIB).

For the purposes of measuring a wireless LAN, SNMP provides a mechanism for extracting more detailed information out of an access point than syslog provides. The level of data depends on the extent of the particular AP's SNMP support. The IEEE 802.11 standard includes a MIB [11], but this is sparse, and concentrates on client-side variables. In keeping with the intent of RFC 1812 [2], which requires "the ability to do anything on the router through SNMP that can be done through a console," many AP vendors have written their own vendor-specific MIBs. These MIBs contain many variables that are useful for measuring a wireless LAN. These may be client-specific variables, such as the MAC address, signal strength, or power saving mode of each client associated with the AP. Or they may be AP-specific variables, such as the number of clients currently associated with the AP, or the number of clients that have recently roamed away from the AP.

Even if an AP lacks a vendor-specific wireless MIB, there remain many useful data that can be obtained from general MIBs. Most APs support the standard network interface MIBs [8]. By querying these MIBs, it is possible to determine some interface-specific variables, for instance, the number of inbound and outbound bytes and packets that have passed the AP's wired interface. This can be used as an indicator of the amount of wireless traffic, although it may not include traffic between two wireless hosts on the same AP, whose traffic may not traverse the wired interface.

As with syslog, SNMP data collection can be impacted by different WLAN setups. If a centralized wireless switch is deployed, it may be necessary to query this switch in addition to, or instead of, individual APs. Some networks may prevent SNMP for security reasons, or allow SNMP queries only from particular subnets.

Once the variables to be queried have been determined, a script is required to query these variables on a periodic basis. If querying a large number of APs, a tool that can perform simultaneous asynchronous queries, without having to wait for previous queries to complete, is highly recommended. In our studies, we have had success using the open-source net-snmp suite of SNMP tools [18] and the related Perl modules.

By collecting the MAC addresses of the associated clients at each AP over time, SNMP can also be used for the identification of client sessions. The accuracy of these sessions, however, will depend on the chosen poll interval, that is, the period between queries. If the poll interval is too high, then the SNMP queries may fail to observe those clients who associate and disassociate with an AP between two polls. On the other hand, if polls are too frequent, the resulting additional traffic to and from the APs may impact the performance of the network by overloading the APs or links. Previous studies (see Section 1.3) have used poll intervals ranging from 1 to 15 min. In our studies, where SNMP was used to query over 500 APs, we found that a 5-min poll interval was required to prevent overloading the network with SNMP traffic. Advantages and disadvantages of SNMP are as follows:

*Pros* — detailed information easily retrievable from many APs; data can include link, network, and transport layers

*Cons* — coarse temporal granularity (a poll interval below 5 min may saturate a LAN); vendor-specific MIBs means additional effort required to measure different types of AP

### 1.2.3 Authentication Logs

Wireless LANs are popular because of the ease with which a client can connect. This presents new security vulnerabilities, however, and as such, many deployed WLANs require some form of authentication before a client is permitted to access the network. Analysis of the logs from an authentication server is another mechanism for determining user behavior; user sessions can be calculated by recording login and logout times. Since an individual user will always use the same login

name, irrespective of the host being used to access the network, these sessions may be more accurate for studies where individual usage patterns are of interest. On the other hand, these sessions may not necessarily correspond to actual WLAN behavior; they may lack details of the APs that a user visits, or the timestamps may differ from actual 802.11 authentication and deauthentication times. Nonetheless, in a network that uses authentication, authentication logs are a source of data that are easy to collect, as they will typically be stored in a single central authentication server. Advantages and disadvantages of authentication are as follows:

*Pros* — accurate session-level information for each individual user; easy to collect from a single source

*Cons* — not all networks use authentication; authentication sessions may not necessarily correspond to wireless sessions

### 1.2.4 Network Sniffing

Network or packet “sniffing” refers to the act of capturing network traffic. By placing a network interface into promiscuous mode, the interface will ignore its assigned address and accept all frames. It is then possible to observe any packets that pass this interface. A program such as tcpdump [25] can capture these packets to disk, and a protocol analyzer such as ethereal [6] can dissect these packets to determine such useful data as the source and destination, the protocol, and in many cases the application being used.

By placing a network sniffer near a router or switch that connects a WLAN’s APs to the wired network, it is possible to record the traffic that is traversing the wireless portion of the network. If MAC addresses are being used to represent individual users, then care must be taken to place the sniffer before the first router, so that the original wireless client MAC addresses are preserved. Some switches offer a “port mirroring” mode, which can bounce the traffic seen on some ports to another port. This can be useful for sniffing, as a sniffer could be connected to a mirrored port and thus monitor any number of ports on that switch. This requires a sniffer with two Ethernet interfaces: one interface connected to the mirrored port, and another to the wired LAN for remote access. Tcpdump can then be run on the interface that is connected to the mirrored port. If port mirroring is not used, and a sniffer with only one interface is used, then it is necessary to remove any traffic to and from the sniffer (e.g., remote logins) from the packet traces. Furthermore, we have found that a sniffer intended to monitor a wireless subnet may sometimes end up seeing traffic from wired hosts on that subnet because the switches have been misconfigured or are malfunctioning. It is useful to correlate sniffer data with data from other sources, and we use a list of the MAC addresses observed through syslog to remove any nonwireless data.

Since sniffers need to be located before the first router to capture client MAC addresses, they may need to be physically located near the APs being sniffed. For our studies, we have deployed 18 sniffers among 11 buildings around our campus.

These sniffers are located in locked switchrooms, and physical access requires contacting a network sysadmin. Management of these sniffers is therefore more challenging than for a syslog collecting machine or a SNMP poller, both of which have no restrictions on physical location. To minimize the need for physical access, our sniffers are connected to an uninterruptible power supply (UPS) and configured to automatically reboot after a power failure. Our central data collection server periodically runs a script to check that all the sniffers are reachable via the network, and that they are correctly collecting packet traces. As well as making sure that the sniffers are alive and running, they need to be kept secure. While there exist fully automated mechanisms for keeping the software on a machine up-to-date and patched (e.g., “Windows Update” or “RedHat Up2Date”), we have found that automatically applying updates may interfere with the sniffing process. Instead, our scripts signal the presence of updated software, which are then tested on a sniffer in our laboratory before being manually applied to the deployed sniffers.

One important consideration with network sniffing is that the amounts of data involved are much larger than with syslog and SNMP. Monitoring an 11-Mbps (megabits per second) 802.11b WLAN can quickly create hundreds of gigabytes of packet traces, and even more storage space is required to sniff a higher-throughput 802.11a or 802.11g WLAN. It is vital to ensure that sufficient disk space is available for a trace, and it is useful to perform test sniffing to estimate space requirements before the actual start of the measurement study. Even then, some studies have seen machines run out of disk space because of unexpectedly high levels of traffic [27]. Our sniffers collect packet traces 24 hours a day, and then compress and transfer these to a central data collection server in the middle of the night, when network activity is low. We use a feature in tcpdump to ensure that when a trace file reaches a particular size, the file is closed and a new file is created, to prevent large files from exceeding filesystem limits. In addition, we periodically run scripts to monitor free disk space on both the sniffers and the central collection server.

A further consideration is privacy. The packets that are captured through sniffing may contain sensitive data, especially if the LAN being monitored does not use encryption. Most academic institutions will require a study to be approved by their Institutional Review Board for human-subjects research. Some privacy concerns may be alleviated by only capturing packet headers, which may be sufficient for a study that is only concerned with header-level data (packet sizes, interarrival times, and so forth). Advantages and disadvantages of sniffing are as follows:

*Pros* — detailed packet capture information, including Ethernet headers and data; microsecond temporal granularity

*Cons* — easiest to capture traffic only on the wired side of the AP, which misses some wireless traffic; ease of sniffing depends on network topology; lots of disk space is required; potential privacy concerns; if a sniffer is monitoring several APs, it can be difficult to determine which AP delivered a particular packet in a trace

### 1.2.5 Wireless Sniffing

SNMP, syslog, and network sniffing are useful tools for measuring the wired side of the wireless LAN, that is, the wireless traffic that APs bridge on to the wired network. In most wireless LANs, this might be preferred, since the wireless side of the network is likely to be more bandwidth-constrained, and so any active measurement should take place in the less utilized wired network. The disadvantage of only looking at the wired side of the WLAN is that not all wireless data are observable on the wired network. Wireless hosts who are communicating with each other, while both associated with the same AP will not send their traffic via the wired network. IEEE 802.11 management frames and beacons, retransmissions, and collisions are not sent on the wired network, as they are specific to the wireless side. Users that fail to associate with an AP, for instance, rogue wireless clients attempting to gain access to a closed WLAN through MAC address spoofing, or clients that have been misconfigured, will also not be seen on the wired network.

To measure all of this additional traffic and observe the 802.11 PHY/MAC layer, it is necessary to “sniff” the wireless side of the network, that is, to scan the RF spectrum. Fortunately this can be accomplished using relatively simple hardware. Certain 802.11 NICs are capable of being placed into “monitor” mode. With a card in this mode, a packet sniffer will capture 802.11 headers and management frames as well as data packets. These stored frames can be analyzed in a fashion similar to those for wired sniffing. Not all NICs support this mode; popular chipsets with monitor support include the Intersil Prism, Orinoco, and Atheros.

Another measurement option is to use dedicated wireless monitoring hardware, such as a “wireless intrusion protection system” [19]. These typically involve small low-powered wireless devices, designed to be placed in monitor mode and monitor the RF spectrum for specific behavior, such as rogue clients. These devices are similar to APs, and with one of the many APs that run Linux, such as the Linksys WRT54G, it is possible to flash a new firmware on to the AP to turn it into a wireless sniffer [22]. Using these systems can be cheaper than using PCs as sniffers. They lack dedicated storage, however, and a measurement study that intends to store 802.11 frames would require frames to be transmitted from these devices to a central server. To transfer frames from these devices, the 802.11 frames need to be encapsulated into an Ethernet packet for transmission across the wired network. There are several different formats for this encapsulation, depending on the tool being used [1,12,24]; to facilitate data analysis, it is useful to ensure that all the measuring devices use the same format.

Wireless sniffing has several challenges that are not present in wired sniffing. Yeo et al. [26] define three instances where a wireless sniffer might not capture all the traffic on the network. *Generic loss* is where frames are lost because of lack of signal strength, for instance, if a sniffer is too far away from the AP or the client being sniffed. *Type loss* is where frames are not captured as a result of device driver failure, or the inability of a particular card to be placed in monitor mode. The third type of loss, *AP loss*, occurs when firmware incompatibilities

cause a particular 802.11 NIC to be incapable of capturing all the packets from a particular type of AP. Some of these losses can be minimized by using multiple sniffers, or sniffers with different 802.11 chipsets. Experimentation in the area to be measured with various antennas and sniffer positioning may also help.

In addition to inadvertently missed frames from type, generic, and AP losses, a wireless sniffer may also miss frames if it is on the incorrect channel. Most wireless NICs can monitor only one channel at a time. With three nonoverlapping channels in the 2.4 GHz band, and 12 non-overlapping channels in the 5 GHz band, monitoring just one channel may potentially miss a large amount of traffic. Mishra et al. [17] find that it is possible to sniff three adjacent channels simultaneously, although 12% of the frames are lost. To resolve this problem, one could choose to either (1) monitor only the channels on which the WLAN's APs are operating, thereby missing any misconfigured client traffic; (2) cycle the sniffer's NICs through all the available channels, which may miss traffic on the channels not currently being monitored; or (3) install one sniffer for each 802.11 channel, at a greater expense.

Whereas wired sniffing can use a relatively small number of sniffers to measure several APs by placing a wired sniffer near an appropriately located router, a wireless sniffer needs to be physically collocated with the APs that it is monitoring, as it needs to be able to "hear" the same frames as the AP. This means that the number of sniffers is proportional to the number of APs, and so a wireless measurement study of a large WLAN could prove expensive. Advantages and disadvantages of wireless sniffing are as follows:

*Pros* — can capture *all* wireless traffic, including management frames, as opposed to just the traffic that traverses the wired side of an AP

*Cons* — capturing every packet can be difficult, and is highly dependent on antennas, 802.11 card firmware, and the positioning of sniffers; not all cards support monitoring; no common data format; privacy issues

### 1.2.6 Client-Side Tools

The previously discussed tools are all designed to monitor from the network perspective. Another measurement method is to directly measure what a wireless client is doing, by installing software on the client. This offers many advantages. A client-side tool can accurately determine exactly what the client is seeing. While syslog provides the AP at which a client is associated, a client-side tool could list all the additional APs that a client can see, which can be useful for mobility tracing. A client-side tool can list all the applications that a wireless device is using, rather than just those applications that are generating network traffic.

Writing a client-side tool can be challenging, however, if it is to run on a variety of client devices, with different operating systems and different device drivers. In addition, a tool will need to be installed on end devices. Some users may find this intrusive, and choose to disable the tool, and there may be privacy implications to consider. Advantages and disadvantages of client-side tools are as follows:

*Pros* — the best way to accurately capture exactly what the client is seeing

*Cons* — can be difficult to write a tool that supports multiple platforms, device types, and device drivers; difficult to deploy and maintain tool on a large number of devices; privacy issues

### 1.2.7 Other Considerations

As well as the software and hardware required for a wireless LAN measurement study, there are some data that require manual nonautomated collection. Much of this requires collection before a study should commence.

The first item that needs to be manually obtained is a list of the APs to be measured. The AP MAC addresses will be needed to make sense of syslog data. If the APs have been assigned IP addresses or hostnames, then these should also be collected. If the APs have dynamically assigned IP addresses, then access to a DHCP server may be required to collect SNMP data, as the AP's IP address needs to be known to query it via SNMP. If syslog or SNMP are to be used, then all the APs to be measured will need to be configured for syslog and/or SNMP, and the SNMP community string will need to be ascertained.

Collecting mobility traces requires knowledge of the physical location of the APs. This can be done with the aid of GPS units, but in most WLAN installations, the APs are indoors, where GPS is of little use. Maps of buildings are generally the best way of plotting the location of APs.

Long-term measurement studies must also keep track of changes in the network. In the course of monitoring our campus WLAN for over 3 years, we have found that APs will be moved to improve coverage, additional APs are introduced over time, or new security measures are introduced that interfere with data collection. In some scenarios it may be possible to automatically determine these changes, for instance, through a wireless sniffer detecting frames from new APs. In most cases, however, close interaction with network administrators will be needed to track these changes. Moreover, if syslog and SNMP are being used, new APs will need to be configured appropriately as they are added to the network.

## 1.3 MEASUREMENT STUDIES

Having described some of the techniques that can be used to measure a wireless LAN, we now discuss some of the measurement studies that have been conducted, and the methods that these studies have employed.

### 1.3.1 Campus WLANs

Most measurement studies have taken place in a university campus setting. This is not surprising, as for an academic researcher, it is typically easier to get permission to measure one's own network.

One of the first wireless LANs to be measured was at Stanford University. Tang and Baker measured 74 users on the Stanford Computer Science departmental WLAN for 12 weeks in 2000 [23]. They used network sniffers, authentication logs, and SNMP with a 2-min poll period. With only 12 APs in the wireless subnet, a 2-min poll period was feasible, as each poll generated only approximately 50 kB (kilobytes) of traffic. Moreover, the SNMP polls were small, as they only queried one specific variable: the list of MAC addresses associated with a particular AP.

Tang and Baker's study looked at user behavior, mobility, and traffic. They found that usage peaks in the middle of the day. Users were not highly mobile, and on average only 3.2 users visited more than one AP in a day. The sniffer analysis indicated that the most popular applications were WWW browsing and ssh or telnet sessions. The latter is unsurprising given the computer scientist population. Half of the users used interactive chat applications such as ICQ and IRC.

Hutchins and Zegura [10] traced a subset of the Georgia Tech campus WLAN, comprising 109 APs in 18 buildings, for a 2-month period in 2001. The methods used included network sniffers, SNMP, and Kerberos authentication logs. Their SNMP polls had a relatively large interval of 15 min. The authentication logs provided a basis for calculating user sessions, from the time that a user logged in, until the network's firewall timed out an idle user.

This study was again concerned with user behavior, mobility, and traffic patterns. Strong diurnal usage patterns were found, and there was a peak in usage around 4 P.M. which they suggest was due to the end of the workday. The number of users each day grew almost linearly over time, falling only during university holidays. From the sniffer traces, they examined flow counts and flow lengths, rather than the absolute amounts of traffic. Short flows (less than 5 min) dominated, although some long flows of almost 9 hours were observed. The longest flows were ssh or telnet, but the largest number of flows were HTTP. Over the course of the study, 228 out of 444 users were seen in more than one building. They calculated mobility on an aggregate basis, rather than a per user basis, and users who "ping-ponged" between nearby APs may skew these data.

Chinchilla et al. [5] conducted a WLAN measurement study focusing on WWW users at the University of North Carolina. They used syslog and network sniffers to trace 222 APs over an 11-week period. Rather than collect every single wireless packet, this study chose to collect only HTTP requests. The authors used tcpdump to look for TCP traffic on any port, and recorded any packet where the payload began with the ASCII string GET.

This study was interested in the locality of WWW behavior and mobility; 13% of the unique URLs being requested accounted for 70% of the HTTP requests, and 8% of requests were for WWW objects that a nearby client had requested within the last hour. This suggests that caching at APs might have some benefit, and they estimate that a cache at each AP would have been useful for 55% of the requests over the entire trace. Student residences were found to have the most wireless associations, and most clients were nonmobile, which may be due to students leaving their laptops connected to the WLAN in their dorms. A Markov chain was used

to develop an algorithm to predict the next AP that a user will visit; this was capable of predicting the correct AP 87% of the time over the trace.

Schwab and Bunt measured the WLAN at the University of Saskatchewan in 2003 [20]. They used a network sniffer and Cisco LEAP authentication logs to trace 18 APs over a one-week period. Unlike most other measurement studies, this study did not use the tcpdump sniffer, but an alternative program called EtherPeek [7].

The Saskatchewan study examined user behavior, mobility, and traffic. This is a nonresidential WLAN, and so again the diurnal patterns mirrored the workday. Web traffic accounted for  $\approx 30\%$  of the traffic, but there was little ssh or telnet usage, which may be due to most WLAN users being law students, as opposed to computer scientists. Users were nonmobile, and the APs in the law school saw significantly more use than did other APs. This led the authors to conclude that APs should be deployed with a view to providing network access in a specific location, rather than providing ubiquitous mobile access.

McNett and Voelker [16] used a client-side tool to measure mobility on the University of California San Diego WLAN. A tool was installed on 272 PDAs, which were equipped with a 802.11b CompactFlash adapter. The tool periodically recorded the client's signal strength for each visible AP, the AP at which the client was associated, the device type, and whether the PDA was using AC or battery power. As the PDAs lacked large storage capabilities, the PDAs would contact a central server to upload collected data.

This PDA study looked at user session behavior and mobility. There were regular diurnal patterns, and less usage at the weekends. Usage was bursty, which may be due to the difficulty of using a PDA for long periods of time. Interestingly, there was a steady decline in the number of users over the trace period. This may be due to the user population (students) becoming bored with the devices. They defined two types of session: (1) the *AP session*, the amount of time that a given PDA spends associated with an AP, and (2) the *user session*, the contiguous time period in which a PDA is switched on and connected to the WLAN. AP sessions were significantly shorter than user sessions, indicating that roaming was taking place while the PDA is in use. Despite the difficulty of using a PDA, there were some long sessions, with 20% of user sessions over 41 min long. Over the course of the trace, 50% of the users visited more than 21 APs. As in the Saskatchewan study, AP load was uneven, and 50% of the APs only saw 5 users or less, and 10% of the APs saw 84 users or more. The mobility traces were used to develop a *campus waypoint* mobility model, which incorporated knowledge of specific geographic locations on campus. Comparing the trace-based mobility model to traditional synthetic mobility models indicated three significant differences. In the trace-based model (1) only a small number of users (11%) were actually mobile at any given time, compared to most nodes in a synthetic model; (2) users were walking at lower speeds (1 m/s) than synthetic models (0–20 m/s); and (3) users appeared and disappeared from the network, which is not considered in most synthetic models.

At Dartmouth College, we have conducted the largest studies of an academic WLAN. We have collected syslog messages from most of the APs on campus since their installation in 2001. We have also used SNMP and tcpdump wired sniffers for

two extensive studies covering 476 APs for 11 weeks in 2001/02 [13] and 566 APs for 17 weeks in 2003/04 [9].

In our 2001/02 study we examined user behavior and traffic patterns. The Dartmouth campus differs from those in other studies as it covers a wide range of locations: academic areas, sporting grounds including a ski slope, residential dormitories and houses, communal eating and social areas, and parts of the town in which the college is based, including some shopping areas, a hotel, and restaurants. In terms of the amount of traffic, the residential areas dominated all other areas. The diurnal usage patterns observed elsewhere were also present at Dartmouth, although the residential nature of the campus meant that usage did not stop at the end of the workday, with many students using the WLAN late at night. User sessions were short, with a median of 16.6 minutes, and 71% of sessions were shorter than one hour. As in other studies, WWW traffic was the most popular application, although some clients also used backup programs over the WLAN, which contributed to a large proportion of the overall traffic.

In our 2003/04 study, we chose to examine changes in user behavior on the WLAN. After 3 years of deployment, the WLAN could be considered a mature network, and an integral part of college life. The college had also begun to replace the analog telephone system with a Voice over IP (VoIP) telephone system, and some students were issued with VoIP clients, which could be used over the WLAN. We found that the types of application used on the WLAN changed dramatically between 2001/02 and 2003/04; while HTTP was still the most popular application in terms of the amount of traffic, peer-to-peer file sharing and streaming media saw significant increases in usage. Wireless VoIP did not appear to be a popular application, with most VoIP calls being made on the wired network. As a result of the increase in file-sharing, local (on-campus) traffic exceeded off-campus traffic, a reversal of the 2001/02 situation. Residences still continued to generate the most traffic, and usage remained diurnal, between our two studies.

Our 2003/04 study also examined mobility. The syslog data indicated that many users “ping-ponged” between APs in range, and so when examining the mobility of a session, we considered the *session diameter*, that is, the maximum distance between any two APs visited in a session. Sessions with a diameter below 50 m were considered to be nonmobile, as they were assumed to consist of ping-ponging clients. From the tcpdump logs, we used a tool for analyzing TCP flows to estimate the operating system being used by a device (by looking for differences in window sizes, ACK values and so on). We used this information to classify the device by type: Mac or Windows laptop, VoIP phone, PDA, and so forth. This information was used to characterize mobility among different device types. Devices such as VoIP phones, which are always switched on, were found to visit significantly higher numbers of APs and have longer session durations than laptops, which are typically powered down before a user moves between locations. Overall, users were found to be nonmobile, with 50% of users spending 98% of their time in a *home location*, that is, a group of one or more APs within a 50 m<sup>2</sup> area with which a user is most often associated. In separate work, we have also used the association and disassociation times in our 3 years of syslog traces to create a mobility history for each

user, which were then used to develop and evaluate mobility prediction models [21]. For user histories containing less than 1000 movements, most predictors performed badly. For histories longer than this, however, the best predictors had accuracies of around 65–72% for the median user; that is, they were able to correctly predict the next AP with which a user would associate 65–72% of the time. Interestingly, simple Markov-based predictors performed just as well as more complex compression-based predictors. In particular, an order 2 Markov predictor, with a “fallback” to a shorter order 1 predictor when encountering a new context not seen in the user’s history, performed the best overall.

### 1.3.2 Nonacademic WLANs

One of the few WLAN measurement studies to take place outside an academic campus setting was conducted at a corporate research facility by Balazinska and Castro in 2002 [4]. They used one method, SNMP with a polling interval of 5 min, to query 117 APs over 4 weeks.

This corporate study concentrated on AP loads and user mobility. As seen in other studies, some APs were little used, with 10% of the APs seeing less than 10 simultaneous users. The most highly utilized APs in terms of the number of simultaneous users were in communal locations such as cafeterias and auditoriums. In terms of traffic levels, however, the most highly utilized APs were in laboratories and conference rooms.

Users were found to be predominantly nonmobile, with 50% of the users visiting less than three APs in a given day. Two metrics were introduced to characterize mobility: *prevalence*, the amount of time that a user spends at a given AP over the course of a user’s trace; and *persistence* which measures the amount of time that a user stays associated with a given AP before moving to the next AP. Using the prevalence data, users were categorized by varying degrees of mobility, from “stationary” to “highly mobile.” Stationary users had a high maximum prevalence, as they spent most of their time associated to a single AP, while highly mobile users had low maximum and median prevalences, spending their time at different APs. The persistence metric complements prevalence by accounting for the amount of time spent at each AP, and unsurprisingly, persistence was lower at guest locations.

Also outside the academic setting, Balachandran et al. [3] used SNMP and sniffers to analyze 195 wireless users at the 2001 ACM SIGCOMM conference. They chose a polling interval of one minute. Such a short polling interval was possible because of the small number of APs involved in the study—there were only four APs used at the conference.

This study examined user behavior, traffic patterns, and AP loads. Given the conference setting, usage closely followed the conference schedule, and the number of users rose when sessions were taking place, and fell during meals and breaks. Arrival times were modeled using a Markov modulated Poisson process, where arrivals vary randomly during an ON period (the conference sessions). Session durations were Pareto distributed, with most sessions under 5 minutes in length, and many of the longest sessions idle and transferring little data. The most popular application

was again WWW browsing, and since SIGCOMM participants are predominantly computer scientists, ssh was the second most popular application. Unlike most studies, the majority of users (over 80%) were seen at more than one AP in a day, although this may be conference-specific, where an attendee does not have a designated seat, and so they would associate with a different AP depending on where they are sitting in a given conference session. AP loads were found to vary not with the number of users, but rather with the applications that individual users are using.

### 1.3.3 Wireless-Side Measurement Studies

As we have described in Section 1.2, wireless sniffing is complicated, and as such, there have been few large measurement studies of the wireless side of a WLAN.

In two studies, Yeo et al. [26,27] looked at the difficulties of conducting wireless-side measurement. To estimate the amount of loss incurred in wireless measurement, three wireless sniffers were compared to a wired sniffer and SNMP polls with an interval of one minute. A packet generator was used to send UDP packets, marked with sequence numbers, between hosts, all on the same channel. The three sniffers were found to have different viewpoints of the wireless medium. All the sniffers were more successful at capturing traffic from the AP, rather than from the clients, as APs tend to have larger and more powerful antennas, and clients may move around and end up out of sight of a sniffer. On average, the sniffers saw 99.4% of the packets from the AP, but only 80.1% of the packets from clients. By merging the traces from the three wireless sniffers, this capture rate was improved, to 99.34% of the traffic that the wired sniffer observed. One recommendation from this study is that one sniffer should be placed near to the AP being monitored, with any other sniffers placed as near as possible to the predicted location of clients.

In a subsequent experiment, Yeo et al. considered seven APs in the University of Maryland's Computer Science department. Three wireless sniffers were used, equipped with Orinoco 802.11b NICs placed into monitor mode, locked to one channel (6), and configured to capture 802.11 frames using the Prism2 file format. This enabled the monitoring of the three APs that were using channel 6. The study took place over 2 weeks, although there was one hole because the sniffers ran out of disk space.

This study concentrated on the PHY/MAC layer, as this can be examined only using wireless sniffers. The maximum throughput seen on a single AP was only 1.5 Mbps, due to contention on the channel that was shared between the three APs. The level of transmission errors, that is, the number of retransmitted frames divided by the total number of frames, varied by day, but there were more transmission errors in the data being sent to an AP, rather than from an AP. Examining the types of frames, they found that dataframes made up 50.7% of the frames sniffed, and beacon frames made up 46.5%. Association and reassociation response frames tended to be sent at the highest data rate, 11 Mbps, whereas the corresponding request frames were sent at 1 Mbps. The 802.11 standard does not specify a behavior for response frames, and by sending responses at a high data rate, many response

frames did not reach the client and needed to be retransmitted. Other management frames, including probe response and power-save polls, were also often retransmitted. For data frames, multiple data rates were common, and the average data rate was 5.1 Mbps.

Mishra et al. [17] used wireless sniffing to look solely at the 802.11 MAC layer handoff process. Eight machines were used as wireless sniffers, with a total of 14 802.11b NICs installed across the machines. Each NIC was set in monitor mode and locked to one individual channel, which allowed the monitoring of all 11 2.4 GHz channels. One user with a laptop then walked around the University of Maryland Computer Science department, which had three WLANs comprising some 60 APs using Cisco, Lucent, and Prism2 chipsets. As this study concentrated on handoffs, the only frames that the sniffer recorded were probe requests and probe responses, reassociations, and authentication frames. To examine variations in handoffs between device drivers, three different 802.11b NICs were used in the client laptop: Lucent Orinoco, Cisco 340, and a ZoomAir Prism2.5.

Across all the devices, probe delay (probe request and probe response frames) was found to account for over 90% of the overall handoff latency. There was a large variation in the handoff latency between devices, with a Lucent client and Cisco AP taking an average of 53.3 ms, and a Cisco client and Cisco AP taking 420.8 ms. With the same device and AP configuration, there was a large variation in handoff latency, and the higher the latency, the higher the standard deviation. Some of the differences in latency between devices could be explained by the different behaviors between devices. The Lucent and Prism NICs would send a reassociate request prior to authenticating with a new AP, and a second reassociate request after authentication. There were also large differences between each device's probe wait time (the amount of time that a scanning client waits before moving on to scanning the next channel). The Cisco client sent 11 probes on each channel, and spent 17 ms on channels with traffic, and 38 ms on channels with no traffic; the Lucent sent three probes on channels 1, 6, and 11, and spent almost the same amount of time on channels irrespective of traffic; the ZoomAir sent only three probes on channels 1, 6, and 11, and spent an additional 10 ms after the three probes on selecting the AP with which to associate. Using the empirical data from the sniffer logs, the authors suggest that device manufacturers could choose to lower these probe wait times.

### 1.3.4 Discussion

Table 1.1 lists the methods used in the studies that we have discussed above.

In summary, Table 1.1 shows that there have been several studies of academic campus WLANs, and fewer studies of nonacademic WLANs. Common methods include syslog, SNMP, and sniffing. The results of these studies show that the most common applications used on a WLAN are not necessarily mobile applications, with HTTP accounting for most traffic, and telnet and ssh used in computer science environments. Short flows and sessions are common, and this should be kept in mind when choosing poll intervals for a measurement study. Users tend

**TABLE 1.1 Wireless Studies and Methods Used**

Study	Location	Duration	APs	Syslog	SNMP [Poll Interval (min)]	Sniffers	Methods Used		
							Authentication Logs	Client Tools	Wireless Sniffing
Balachandran et al. [3]	Conference	52 hours	4		1	✓			
Balazinska and Castro [4]	Corporate	4 weeks	117		5				
Chinchilla et al. [5]	Academic	11 weeks	222	✓	5	✓			
Henderson et al. [9]	Academic	17 weeks	566	✓	5	✓			
Hutchins and Zegura [10]	Academic	2 months	109		15	✓	✓		
Kotz and Essien [13]	Academic	11 weeks	476	✓	5	✓			
McNett and Voelker [16]	Academic	11 weeks	>400					✓	
Mishra et al. [17]	Lab	30 min	60						✓
Schwab and Bunt [20]	Academic	1 week	18			✓		✓	
Tang and Baker [23]	Academic	12 weeks	12		2	✓		✓	
Yeo et al. [26,27]	Lab	2 weeks	3		1	✓			✓

to be nonmobile, although the introduction of new always-on devices is leading to increased mobility. APs tend to be unevenly used across a WLAN, with certain locations accounting for high levels of traffic. Trace-based mobility models and predictors have been developed, and it will be interesting to see how these perform with traces of newer, more mobile, clients.

Wireless sniffing is still a new area, and one that presents many challenges. The studies that have used wireless sniffing are much smaller than those that have used wired sniffing, syslog and SNMP, and have concentrated on specific channels in specific locations. Although small, these studies have yielded insights into 802.11 MAC behavior, and highlighted the differences between chipsets and devices. For instance, 46.5% of the frames observed in one study were 802.11 beacons, which indicates the large amount of data that can be missed in a wired sniffing study. Larger-scale wireless sniffing, with a variety of chipsets and device types, could prove useful for future wireless protocol development.

## 1.4 CONCLUSIONS

Wireless LANs are becoming increasingly popular, and it is useful to be able to measure various characteristics of these WLANs. In this chapter we have discussed the tools available for measurement, and the studies that have already been conducted using these tools. To conclude, we present a checklist that we hope will be useful for those intending to carry out a wireless measurement study.

### 1.4.1 Wireless Measurement Checklist

- Determine which tools are most appropriate for the purposes of the study. Syslog is useful for mobility, while SNMP is an easy method for extracting traffic statistics. Client tools and wireless sniffing provide the most detail, but incur the greatest costs in terms of setup time and equipment. It is also useful to use multiple tools and correlate the data, such as using MAC addresses observed in syslog messages to verify that sniffer logs are accurately capturing wireless client traffic.
- Gain approval from the appropriate Institutional Review Board for human-subjects research. Wireless data collection can involve potentially sensitive information, such as the location of wireless users or the data that they are transferring.
- Decide how much of the WLAN will be monitored. Different tools may be able to monitor different parts; for instance, it may be easy to use SNMP to monitor every AP, but sniff only a subset of the WLAN.
- Draw up a list of all the APs to be measured. If required, determine the physical location of these APs, using a building plan and/or GPS.

- Ensure that all the APs that are to be measured are configured correctly, for example, that they are configured to send syslog messages, or to allow SNMP queries, and that the network security policies (if any) allow this syslog and SNMP data to be transmitted to the host that is storing the data. Do not rely on a sysadmin to do this, but confirm it for yourself.
- Test the data collection and analysis software in a “dry run” before the actual measurement study begins. Checking that the analysis software works will help to determine whether sufficient data are being collected and if the appropriate tools are being used.
- Closely monitor the data collection. Keep track of changes in output, such as syslog messages changing as a result of AP firmware changes. Measuring devices may malfunction or run out of disk space, which also requires careful monitoring.
- Keep in touch with the WLAN’s sysadmins. It is important to know when new APs are installed, or when existing APs are moved or decommissioned.
- Minimize disruption on the network being measured. Most of the tools described here are *active* measurement tools, in that they generate additional network traffic. It is vital not to impact the network being monitored. For instance, on one particular type of AP, we have found that frequent SNMP queries could cause the AP to stop forwarding packets.
- Expect the unexpected! Measurement of a live network, with large numbers of real wireless network users, may encounter many surprising events. We have had our measurement studies impacted by viruses, worms, misconfigured wireless clients, firewalls, changes in network subnetting and VLANs and more. With a comprehensive monitoring system as discussed above, however, we have been able to detect most of these problems and reconfigure the measurement infrastructure where required.

Readers who are interested in conducting a wireless measurement study, or who would like access to data from some of the studies discussed here, are directed to our Websites at <http://www.cs.dartmouth.edu/~campus> and <http://crawdad.cs.dartmouth.edu/>.

## REFERENCES

1. Apware project, <http://nms.csail.mit.edu/projects/apware/software/>.
2. F. Baker, *Requirements for IP Version 4 Routers*, IETF RFC 1812, June 1995.
3. A. Balachandran, G. M. Voelker, P. Bahl, and P. Venkat Rangan, Characterizing user behavior and network performance in a public wireless LAN, *Proc. Int. Conf. Measurements and Modeling of Computer Systems (SIGMETRICS)*, Marina Del Rey, CA, June 2002, ACM Press, pp. 195–205.

4. M. Balazinska and P. Castro, Characterizing mobility and network usage in a corporate wireless local-area network, *Proc 2003 Int Conf Mobile Systems, Applications, and Services (MobiSys)*, San Francisco, May 2003, USENIX Assoc., pp. 303–316.
5. F. Chinchilla, M. Lindsey, and M. Papadopouli, Analysis of wireless information locality and association patterns in a campus, *Proc. 23rd Annual Joint Conf. IEEE Computer and Communications Societies (InfoCom)*, Hong Kong, March 2004, IEEE.
6. Ethereal protocol analyzer, <http://www.ethereal.com>.
7. EtherPeek protocol analyzer, <http://www.wildpackets.com>.
8. J. Flick and J. Johnson, *Definitions of Managed Objects for the Ethernet-like Interface Types*, IETF RFC 2665, Aug. 1999.
9. T. Henderson, D. Kotz, and I. Abyzov, The changing usage of a mature campus-wide wireless network, *Proc. 10th Annual ACM Int. Conf. Mobile Computing and Networking (MobiCom)*, Philadelphia, Sept. 2004, ACM Press.
10. R. Hutchins and E. W. Zegura, Measurements from a campus wireless network, *Proc. IEEE Int. Conf. Communications (ICC)*, New York, April 2002, IEEE Computer Society Press, Vol. 5, pp. 3161–3167.
11. IEEE 802.11 MIB, <http://standards.ieee.org/getieee802/download/MIB-D6.2.txt>.
12. Kismet wireless sniffing software, <http://www.kismetwireless.net>.
13. D. Kotz and K. Essien, Analysis of a campus-wide wireless network, *Wireless Networks* **11**: 115–133 (2005).
14. C. Lonvick, *The BSD Syslog Protocol*, IETF RFC 3164, Aug. 2001.
15. K. McCloghrie, D. Perkins, and J. Schoenwaelder, *Structure of Management Information Version 2 (SMIv2)*, IETF RFC 2578, April 1999.
16. M. McNett and G. M. Voelker, *Access and Mobility of Wireless PDA Users*, Technical Report CS2004-0780, Dept. Computer Science and Engineering, Univ. California, San Diego, Feb. 2004.
17. A. Mishra, M. Shin, and W. A. Arbaugh, An empirical analysis of the IEEE 802.11 MAC layer handoff process, *ACM SigComm Comput. Commun. Rev.* **33**(2):93–102 (April 2003).
18. Net-snmp SNMP tools, <http://net-snmp.sourceforge.net>.
19. Network Chemistry RFProtect wireless intrusion protection system, <http://www.networkchemistry.com>.
20. D. Schwab and R. Bunt, Characterising the use of a campus wireless network, *Proc. 23rd Annual Joint Conf. IEEE Computer and Communications Societies (InfoCom)*, Hong Kong, March 2004, IEEE.
21. L. Song, D. Kotz, R. Jain, and X. He, Evaluating location predictors with extensive Wi-Fi mobility data, *Proc. 23rd Annual Joint Conf. IEEE Computer and Communications Societies (InfoCom)*, Hong Kong, March 2004, IEEE.
22. Sveasoft alternative Linksys WRT54G firmware, <http://docs.sveasoft.com/>.
23. D. Tang and M. Baker, Analysis of a local-area wireless network, *Proc. 6th Annual ACM Int. Conf. Mobile Computing and Networking (MobiCom)*, Boston, Aug. 2000, ACM Press, pp. 1–10.
24. Tazmen Sniffer Protocol, [http://www.networkchemistry.com/support/appnotes/an001\\_tzsp.html](http://www.networkchemistry.com/support/appnotes/an001_tzsp.html).

25. Tcpdump packet capture software, <http://www.tcpdump.org>.
26. J. Yeo, S. Banerjee, and A. Agrawala, *Measuring Traffic on the Wireless Medium: Experience and Pitfalls*, Technical Report CS-TR 4421, Dept. Computer Science, Univ. Maryland, Dec. 2002.
27. J. Yeo, M. Youssef, and A. Agrawala, *Characterizing the IEEE 802.11 Traffic: The Wireless Side*, Technical Report CS-TR 4570, Dept. Computer Science, Univ. Maryland, March 2004.

