airConn: A Framework for Tiered Services in Public Wireless LAN Hot Spots

Arup Acharya, Chatschik Bisdikian, and Archan Misra, IBM Corp.
Young-Bae Ko, Ajou University

ABSTRACT

Access to data services via wireless LANs at private and public hot spot sites is becoming commonplace. The goal of the *airConn* project is to define an architecture and a prototype implementation that enable the provision of premium and non-premium service tiers for both transient and nontransient users of wireless hot spots. airConn provides for dynamic renegotiation of service tiers and facilitates various billing modes. Thus, it enables service providers to increase their revenue opportunities via multiple flexibly manageable service offerings.

INTRODUCTION

With the standardization and product maturity of license-free wireless technologies, we are witnessing a rapid increase in alternative opportunities users have to access Internet services at home, in the office, and anywhere between. Appearing first at business and academic buildings and campuses and then at homes, wireless access solutions based on the IEEE 802.11 family of standards [1] are mushrooming in many different places where people congregate, like airports, hotels, cafés, train stations, and parks. These public congregation areas constitute connectivity islands, or *hot spots*, where broadband access services can be provided wirelessly to one's personal devices, such as notebook computers and PDAs. The providers of access services at the public hotspots, referred to as wireless Internet service providers (WISPs), range from home hobbyists to startups to established communication services providers.

The "home-grown" providers understandably provide a single service offering, basic best effort access to the public Internet. However, even for-profit WISPs also provide the same single service [2, 3]. The provision of only one service makes it hard for service providers to differentiate among themselves. Even though for-profit WISPs provide better and more predictable levels of service to their users than home-grown providers, we believe that richer service offerings and service differentiation are fundamental enablers for successful WISP operation. Otherwise, the availability of basic free-access offerings from home-grown providers, intentional or unintentional, just around the corner creates a compelling connectivity alternative for mobile users.

To facilitate the offering of differentiated, or *tiered*, services to users of public hot spots by (for-profit) WISPs, we have developed an easily replicable and extendable networking framework we have coined *airConn* (for airport connectivity). In this article we highlight this framework as well as a testbed we have built with off-the-shelf components based on this framework. The *airConn* framework supports differentiated access to value-added networked services in a public hot spot setting. It allows mobile users to select a desired service tier impulsively from a list or dynamically managed set of service offerings. The framework is based on an add-on server-side control mechanism that does not require any significant modification to an existing wireless network installation and is able to ensure appropriate access privilege for mobile users in a scalable manner.

The *airConn* framework can be applied to both mobile users with a prior subscription relationship with a WISP and those that do not, referred to as *transient* users, allowing WISPs to offer dynamically changeable extra services. These extra services may be present in certain locales where the WISP operates, or may change dynamically, allowing the WISP to differentiate their offerings in a timely fashion. The multi-tiered service offering capability of the framework allows service provider (e.g., WISPs) and property owners (e.g., airport owners and operators) to derive additional revenue opportunities from the operation of the wireless network. For example, an airport operator could make deals with content providers to offer premium content

0163-6804/04/$20.00 © 2004 IEEE

to the users of a specific service tier at hot spots. The assignment of services to various tiers may be further made at a WISP's or property owner's discretion, permitting them to emphasize or de-emphasize multiple revenue generating alternatives based on various criteria like temporal, spatial, or competitive criteria.

Public access can be provided through wireline LAN technologies (e.g., IEEE 802.3) as well, or other personal wireless technologies like Bluetooth [4]. However, it is the undisputed convenience of untethered communications using the 802.11 technologies that drives this space. Thus, even though the *airConn* framework is applicable to these other access technologies as well, we present our work under the guise of an 802.11-based hot spot setup.

The rest of the article is organized as follows. We describe the architectural framework and design motivation for *airConn*. We describe the information flows experienced during the various *airConn* communication phases. We then describe a reference implementation of our framework. We conclude the article with some closing remarks.

THE AIRCONN FRAMEWORK

The deployment of hot spots is expected to enjoy a healthy upswing over the next few years, up to over 150,000 installations by 2005 from a mere 1200 in 2001 [5].The *airConn* framework aims to exploit the existing and anticipated installed base of these hot spots. Hence, it has been central to the framework's philosophy that *airConn* should be seen as added value that does not seek to replace existing installations, but rather enhance their capabilities to support dynamically selectable service tiers. Thus, judiciously left outside the scope of *airConn* are any methods that specifically aim at defining how users become eligible to access the hot spot network in the first place. The latter process is typically defined by the WISPs themselves. The framework aims to minimize user intervention by taking advantage of open standards, but does not dictate any specific methods that would require providers to fundamentally change the way they accept users to their networks.

Next, we highlight the main design objectives of *airConn*.

THE DESIGN GOALS OF THE AIRCONN FRAMEWORK

The *airConn* framework focuses on a solution that permits providers of public networked services to offer different service tiers. To use these services, users are assumed to use their own devices to select and dynamically adjust a desired service tier on a per-use basis, even during an ongoing *airConn* session. To achieve and maintain an enhanced user experience and also facilitate the addition of the framework to existing (legacy) hot spot installations, this capability should require no *airConn*-specific modifications of user devices.

An additional design goal is an enforcement mechanism, applicable in the communications infrastructure, that supports such public service offerings. The enforcement mechanism should be applicable to either the communication elements, such as routers and wireless access points, or other higher-layer traffic controllers, such as Web proxies. The enforcement mechanism ensures that individual users are allowed to access only those services within the service tier they have most recently selected, and denied access to all services that do not fall within that tier. The enforcement mechanism may further be supplemented by some means to alert users of their attempt to access a particular service that does not fall within their currently selected tier, and by which users may dynamically renegotiate their desired service tiers so that they can access new services whenever desired.

A further design goal is to permit users a degree of mobility and roaming within the hot spot area. In particular, users should be allowed to recover their communication sessions following temporary communication interruptions. The interruptions may result from various reasons like momentary loss of connectivity due to a move outside the coverage range of an access point, a device reboot, or a transition to a different point of attachment, either wireless or wireline, within the same hot spot area.

It is an additional design goal for the *airConn* system to facilitate payment policies that charge users relative to the service they have selected and accessed. These payment policies are based on various criteria, including the degree of user activity in terms of the amount of traffic transferred to and/or from the user, and the duration for which a selected service tier is provided (i.e., the session time).

The CHOICE architecture [6] shares a number of the goals of the *airConn* framework. However, *airConn* and CHOICE have fundamentally different design philosophies. With CHOICE, access enablement (e.g., authentication process) and traffic control are integrated. All communications in a CHOICE network require a CHOICE-specific protocol. Operation in a CHOICE-enabled network requires, at a minimum, firmware and software modifications to both user devices and traffic control elements. The *airConn* framework separates access enablement and service access control. By focusing only on the latter, it can be retrofitted to current and future hot spot installations independent of how the former is addressed by the WISP.

THE FUNCTIONAL ELEMENTS OF THE AIRCONN FRAMEWORK

During an *airConn* communication session at a hot spot area, the following phases are experienced:

- *Personal device configuration*: During this phase, user devices are properly configured and permitted access to the hot spot site's intranet.
- *User registration*: During this phase, users provide identification credentials to the system. These could be provided explicitly, where a user explicitly provides personal information, or implicitly, where a pointer to a stored log-on profile is provided by the users or their devices.

The airConn framework focuses on a solution that permits providers of public networked services to offer different service tiers. To use these services, users are assumed to use their own devices to select and adjust dynamically a desired service tier on a per-use basis, even during an ongoing airConn session.

The use of the identification metadata enables a user to easily rejoin a recently interrupted airConn session. More interestingly, it may enable a user to establish a transient temporal and spatial relationship with participating airConn providers.

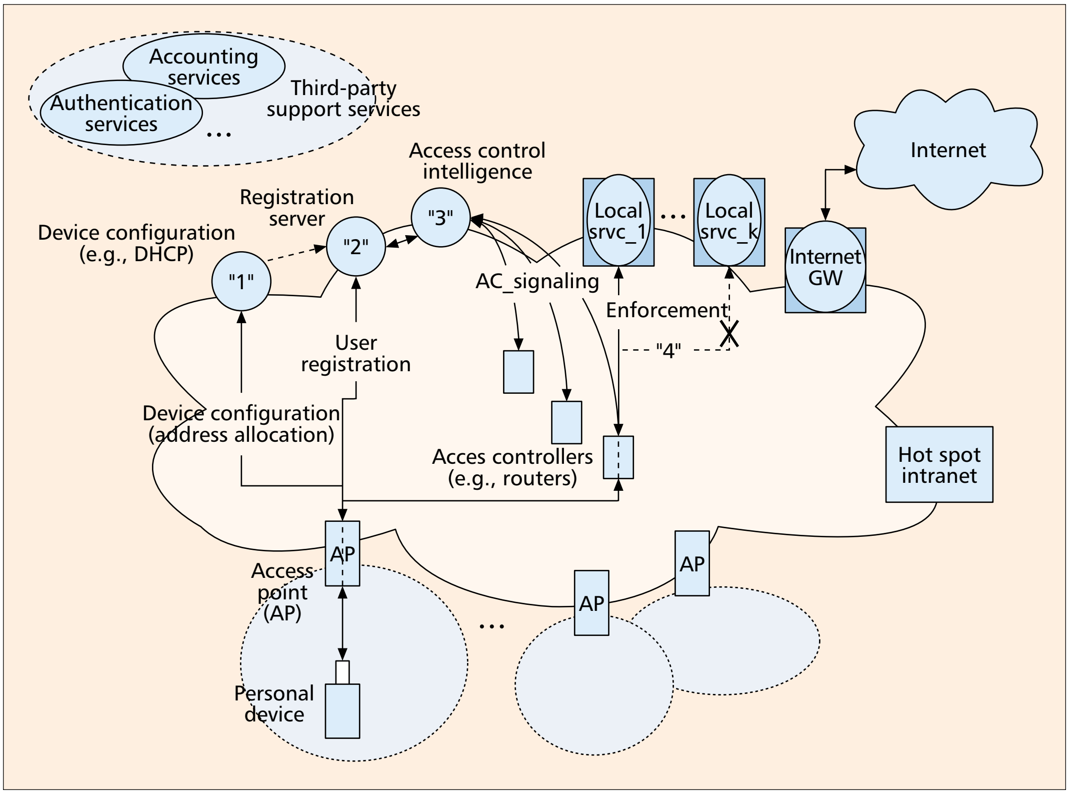

Figure 1. *The phases of the airConn framework.*

- *Service access control*: During this phase, the *airConn* intelligence configures the hot spot site's intranet to allow users to access only services within the service tier they have selected.
- *Session management*: This is a supervisory activity that keeps track of user sessions including their service tier selections, the duration of accessing services at a selected tier, or traffic statistics.

These four phases of the framework are shown in Fig. 1, with the logical/physical entities (e.g., servers and networking elements) responsible for them numbered accordingly. The figure also shows additional third-party services, like accounting and authentication, that may interact with the framework but are not part of it.

Following user registration, users will be able to access the services they have selected. How these services are further organized into various tiers may be based on:
- The locality of the services (services specific for users of an airport, hotel, conference center, shopping mall, train station, etc.)
- How a service provider desires to organize them, which may depend, for example, on how the providers make deals with local or remote content providers
- The time of day or various promotional incentives

The Information Flow in the airConn System

In this section we highlight the information flows anticipated by the *airConn* framework that were summarized in Fig. 1. Note that some of the information exchanges that take place during phase 1 may be proprietary. These exchanges are outside the scope of the *airConn* framework. The framework is only concerned with the fact that by the end of phase 1, a user's device will have obtained, among other things, all the necessary network configuration parameters (e.g., using standard DHCP procedures). We will not discuss this phase further.

In the figures that follow, activities and processes in dashed lines have not been implemented in the testbed system discussed later.

Initial Access to airConn

Figure 2 shows the interaction between a user and the *airConn registration server*. This interaction takes place typically during the initial contact of a user with the *airConn*-enabled system. However, it could happen at a later time as well (e.g., whenever the user renegotiates its service tier or reconnects to the system following a device reboot).

The server inspects the incoming request for log-on identification metadata, such as a Web cookie set up during a previous visit of the user to an affiliated *airConn* installation. The use of the identification metadata enables a user to easily rejoin a recently interrupted *airConn* session. More interesting, it may enable a user to establish a transient temporal and spatial *relationship* with participating *airConn* providers. Such will be the case, for example, when a user enters an *airConn*-enabled establishment. Based on the activity of the user, the user may be offered promotional and personalization opportunities at additional participating *airConn*-enabled establishments she visits afterward.

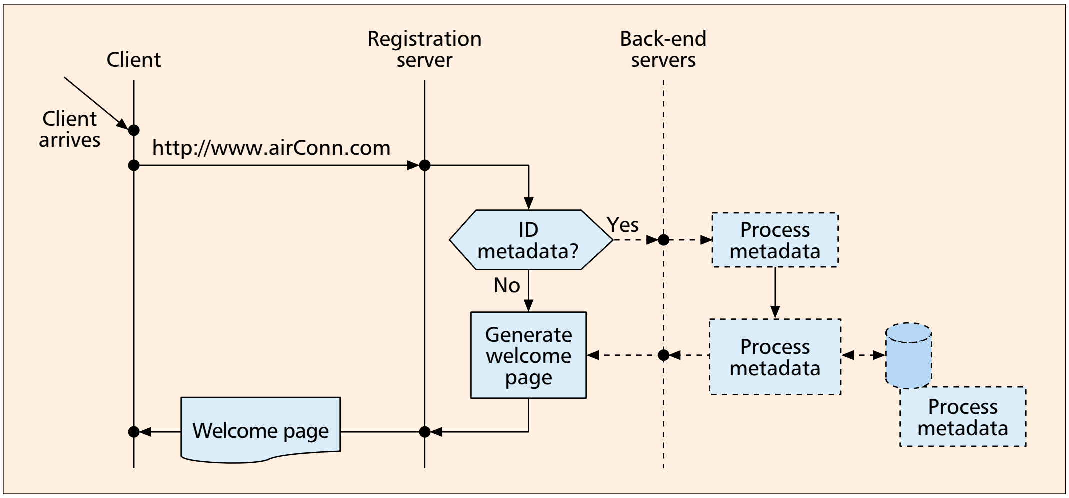

Figure 2. *Initial access to airConn.*

When identification metadata are available, the registration server extracts them and sends them for additional processing. The objective of the additional processing will be to associate any available personalization information (stored in a user profile repository) that could be tied to these metadata. The user profile repository may contain transient information created during recent visits of the user to participating *airConn* locations. This facilitates providing a better user experience on subsequent user visits at a hot spot, like a particular airport. It is worth noting here that users at airports are not everyday users. They travel occasionally through the airport, so the notion of transient user profiles may be quite applicable. For example, when a business traveler passes through an airport en route to a destination, she may be expected to pass through the same airport on the way back a few days later. Thus, maintaining user session statistics for a few days (or weeks) may be a reasonable way to maintain a good user experience without overburdening the storage resources of the service provider.

When identification metadata are available, the registration server extracts them and sends them for additional processing. The objective of the additional processing will be to associate any available personalization information (stored in a user profile repository) that could be tied to these metadata.

Finally, the registration server returns a welcome page to the user, which could contain a combination of payment, service tiers choices, and other information possibly personalized to each individual user.

Setting Up the Infrastructure

Figure 3 shows the information flows after a user selects a service tier. When a profile repository is maintained, the user's profile is updated to reflect the user's tier choice. Then the registration server sends configuration commands to the various controllable network elements (e.g., access points, routers, or proxies) in the hot spot site's intranet. These commands will cause an update of their network traffic flow control parameters (e.g., routing tables), which will permit traffic from a user to access only those services in the tier the user has selected.

Accessing Destinations

Figure 4 shows the information flow following a user selecting a destination. Based on the service tier a user has previously made (Fig. 3), traffic control elements in the hot spot site's intranet will inspect whether the destination chosen by the user belongs to a service tier the user has selected. If yes, access to the particular destination is allowed. If not, the course of action depends on the application used to access the destination and whether that application can provide feedback to the user. For example, consider a case where a standalone MP3 player application on a notebook computer is used to access (for a fee) an online music repository. If this repository does not belong to the service tier selected by the user, the connection may simply be dropped. Alternatively, a prerecorded MP3 file may be downloaded and audibly inform the user that the selected destination is not available for the selected service tier. Alternatively, a third medium may be used for feedback information; for example, a short message service (SMS) message may be used, if a user has registered its SMS service number with *airConn*.

If the application used is a Web browser, any attempt to access a destination not included in the service tier selected by the user may result in a Web page returned with an appropriate message. Furthermore, the user may be redirected automatically or manually to the tier selection Web page on the registration server, where the user may be given the option to modify her selection of service tier. Note that Web page redirection techniques can also be used for newcomers to an *airConn*-enabled establishment to be directed to the welcome page after they start their Web browsers.

Ending a Session

Sooner or later the user will terminate its use of *airConn* and leave. Figure 5 shows the information flows when this happens. The figure contains two alternatives that may exist: one in which the user explicitly clicks the "I'm leaving!" button on her browser, and one where the user simply leaves the premises. No matter which event triggers the termination of an *airConn* session, what happens after is the same. We first discuss the common operations that occur after a session termination has been declared.

Following a session termination, the registra-

Late termination of the user session may be desirable to give the opportunity to the user to "return" back to its session following a temporary interruption, for example, due to temporary hibernation of the user's computer.

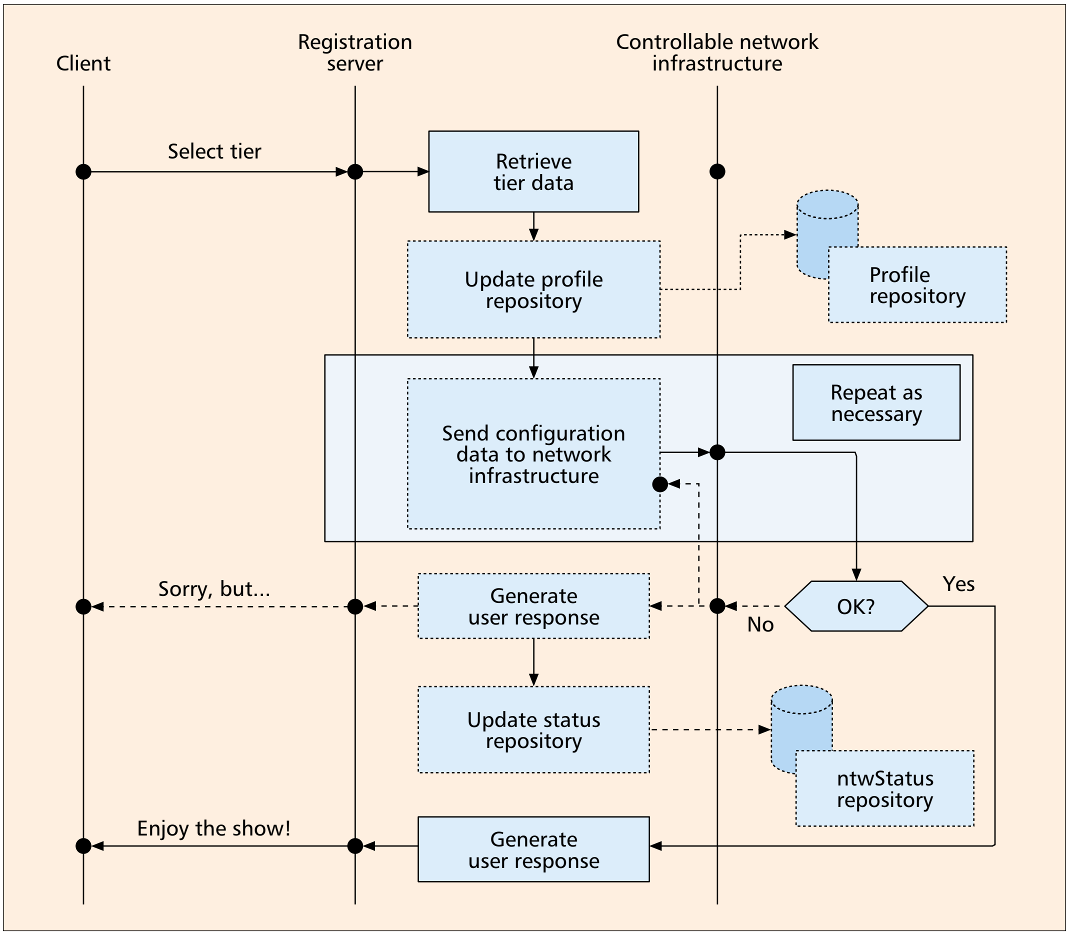

Figure 3. *Setting up the infrastructure.*

tion server will first collect session statistics. Assuming that the user is not charged with a prefixed price, these pertinent session statistics will include the session duration and possibly the bandwidth used during the session. If the user renegotiated a new service tier during her session, the session statistics will include the time and bandwidth pertinent ti each of the service tiers she used during her session.

After the session statistics have been collected from the various network elements involved, the registration server can now instruct these network elements to clear their state pertinent to this user. The *airConn* system may now pass the session statistics information to a third party for proper billing procedures. Finally, if any profiles are legitimately maintained, they can also be updated at this time; for example: user *A* visited *airConn* at location *B*, and selected tiers *C* and *D* for *x* and *y* amounts of time, respectively.

As mentioned earlier, session termination is closely related to customer billing and network resetting. If a user has purchased service for a specific time block (e.g., 24 h), billing is straightforward and will be based on this fact. If, however, the user has not specified a time block, and instead has chosen to be charged based on usage (e.g., total time spent or bandwidth consumed), some means must be provided to monitor the continuous presence of the user in the system. This could happen actively, by having an *airConn* system monitor (ping) devices periodically to see whether they are still available, or passively. In our testbed implementation, discussed in the next section, we have chosen the latter as it is more robust and enjoys the scalability benefits of a distributed system.

The session management entity in the registration server may use information received from the DHCP server to terminate the user session immediately or at a later time. Late termination of the user session may be desirable to give the user an opportunity to return to its session following a temporary interruption (e.g., due to temporary hibernation of the user's computer).

An airConn Reference Implementation Based on Network Layer Enforcement

In this section we present our testbed implementation of the core functions of the *airConn* framework. We focus on the networking aspects of the framework, including device configuration, tier selection, access control and enforcement, and collection of session statistics. The design choices made for this implementation are as follows:

- User devices should require no extra client code to exploit the value-added features of *airConn*. Thus, a server-based solution is adopted.

For user devices, we assume they are capable of IP-based communications and running a Web browser application.

•Access control could be applied in principle at any level of the protocol stack: medium access control (MAC), IP, application/proxy level, and so on. However, MAC layer access control is technology-dependent and binary (allow/disallow a device). It is intended for authentication/connectivity control, not service access control. It will be handled by emerging standards such as IEEE 802.1X [7] and IEEE 802.11i. Access control via Web proxies applies only to HTTP transported traffic and is not applicable to general traffic streams; it can also lead to bottlenecks as all Web traffic needs to pass through them. Applying access control directly on individual servers and services is not scalable or practical, since not all services are under the direct control of the WISP itself. Therefore, our implementation focuses on access control at the IP layer using the routing elements in the hot spot site's intranet. This gives us a high degree of flexibility in controlling traffic from users, as identified by their IP address, by imposing controls on the destination IP address and/or the port number of their transmissions, thus enabling access control at various levels, including per application class.

•Session maintenance and user mobility in the hot spot site is managed using cookies. As each readjustment of the service tier by a user will require a visit to the welcome page" of the WISP, a session cookie may be used to reestablish user credentials quickly.

•To collect usage information per user session, we opted to use adjustable DHCP leases, which enable us to detect when a user moves.

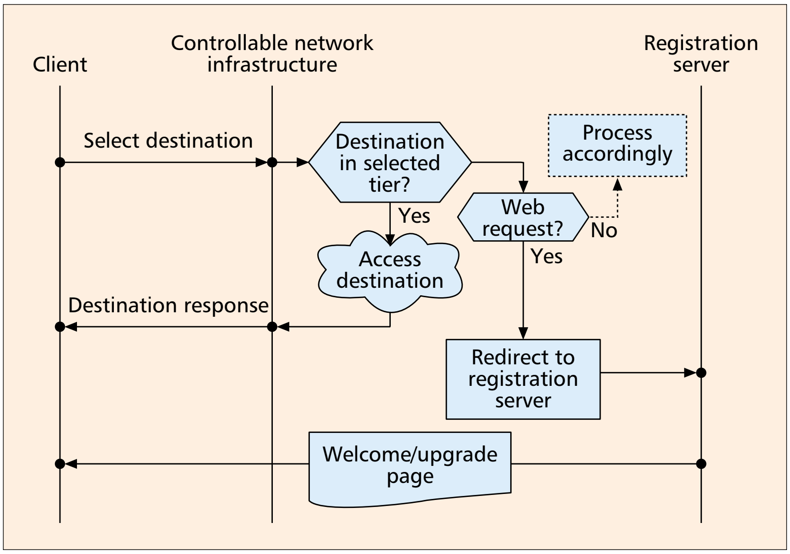

Figure 4. *Accessing destinations.*

Furthermore, we collect traffic statistics by interrogating the routing elements about the traffic statistics they maintain.

THE TESTBED IMPLEMENTATION

As mentioned earlier, this implementation is based on filtering packets at the routers interconnecting the wireless access points and services. When a device first enters the network (i.e., is within range of a wireless access point), it executes the DHCP protocol to obtain an IP address and configure network parameters such as the address of the DNS server or next-hop router. Having obtained an IP address, a user

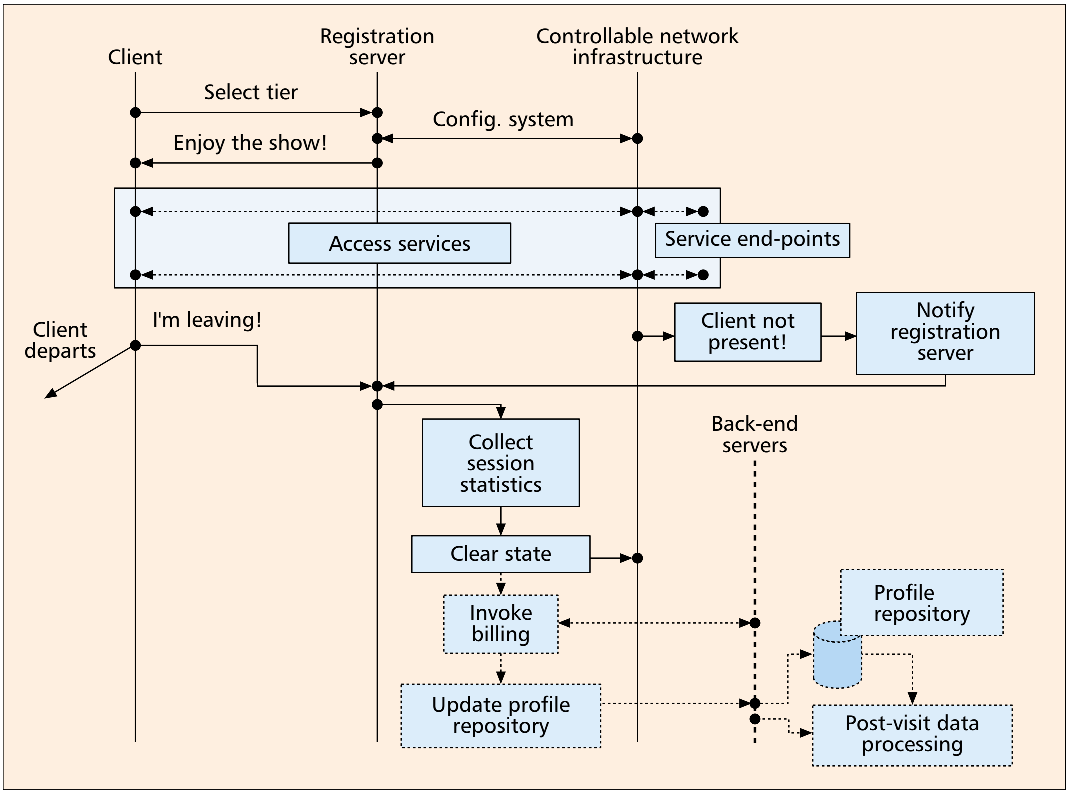

Figure 5. *Ending a session.*

> *When a device first enters the network, i.e., is within range of a wireless access point, it executes the DHCP protocol to obtain an IP address and configure network parameters such as the address of the DNS server or the next hop router.*

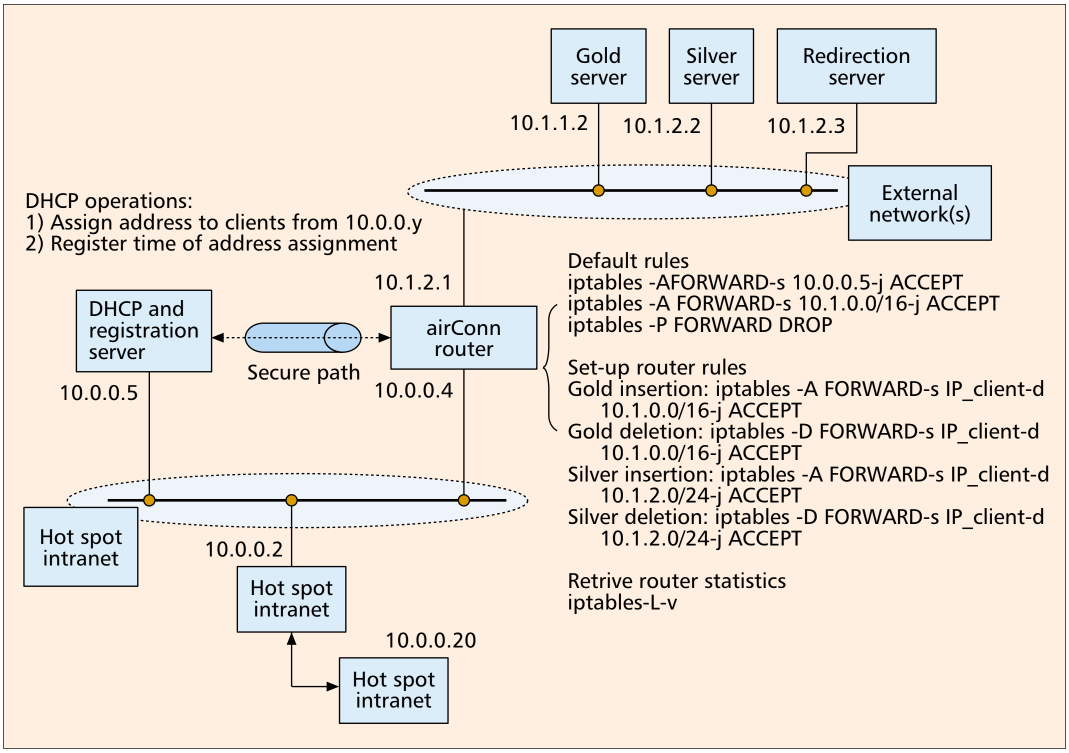

Figure 6. *The airConn testbed.*

device is initially allowed to communicate only with the registration server.

The next step consists of the user starting a Web browser application on her device and communicating with a well-known URL. This URL is advertised out of band. This requirement stems from the fact that we do not modify the client software in any way. If we are allowed to install a logon client utility on the user's device, it may then be possible to send this URL as an option parameter within a DHCP message. We may then extract this URL from the DHCP message and automatically start the browser pointing to this URL, thereby eliminating any user participation. We have indeed developed such a client utility. Since *airConn* connectivity does not depend on the presence of this utility, its existence does not violate our design goal of no extra client code. In either case, the client machine sets up a Web connection with the registration server, and the welcome page from it displays the available service tiers. The user then selects a desired service tier and provides some payment information (e.g., credit card or frequent flyer number) to be charged for the service. This transaction can be secured using typical secure HTTP procedures.

By the end of registration, the registration server will have activated additional packet filtering rules on the router in accordance with the selected service tier and user's IP address.

The *airConn* testbed implementation is shown in Fig. 6. The service tiers are represented by two Web servers, the Gold and Silver servers. The hot spot site's intranet is represented by the network with addresses in the 10.0.0.*y* range. The default packet filtering rules at the router disallow any packets from a user device to cross the router. When a user device enters the network, it receives an address from the 10.0.0.*y* subnet via DHCP. Initially, only the registration/DHCP server is accessible to any user device via the access point. Once network connectivity to the registration server is established, the user starts the Web browser application and accesses the welcome page of the registration server. Users then select the desired service tier (Gold or Silver in this case) and provide a credit card number to enable the service. Gold service allows access to both servers, while Silver service allows access only to the Silver server. If the user selects Gold service, the registration service installs a packet filtering rule at the router to allow packets from the client IP address to IP addresses of both Gold and Silver services: this is represented as access to 10.1.0.0/16 in the router rules in Fig. 6. However, if Silver service is selected, the filter rule installed allows access only to 10.1.2.0/24 (i.e., the Silver server).

For the *airConn* router, we use a notebook computer running Linux (configured with routing capabilities). We use the *iptables* module [8] that allows per-host routing of the traffic passing through the notebook computer. Whenever a new user registers with the registration server, the registration server issues remote configuration commands (using the *rsh* (remote shell) commands) to the *iptables* daemon on the router, thereby setting up the appropriate routing tables. In Fig. 6 we highlight some of the *iptables* configuration commands we have used for setting up routing rules on the router during network initialization, router configuration, and routing rule reset. The communication path between the registration server and the router can be secured through a combination of means including IPSec, using secure shell (*srsh*) to send commands from the registration server and the router, and adding a MAC-level *iptables* filter that may restrict any traffic destined to the router itself to have as its origin the registration server. Since users access the wireline portion of the network over a wireless link, they can see no traffic that occurs between the registration server and the router because the access point can easily double as a bridge-level filter for this traffic as well. Likewise, no wireless *airConn* user can see the MAC address of the registration server, further enhancing the security level of the core *airConn* network.

Upon termination of a user's access session, the registration server collects the usage statistics from the router regarding bandwidth usage (in terms of packet and byte counts). It also collects session duration times using information it obtains from the DHCP server, which we have modified to signal the registration server whenever the lease time of an IP address expires without renewal. By assigning small enough leases, we control the granularity to determine the exact departure time of a user from the hot spot. This approach takes advantage of the standard DHCP procedures where a device will automatically request renewal of its lease as the lease termination approaches. In the absence of a request to renew a particular IP address, the DHCP server can conclude that the user is not available anymore and then notify the registration server of this event.

Our *airConn* testbed has been built from off-the-shelf components. The servers and router used were Linux-based laptop computers. Specifically, we have used Red Hat Linux v. 7.1 on a 366 MHz Pentium II machine with 228 Mbytes to implement the *airConn* router. Even with such a small system, we have shown in [9] that the router is capable of handling the processing of as many as 2000 routing conditions without discernable reduction in system performance.

Concluding Remarks

In this article we have introduced a framework, called *airConn*, for supporting dynamically selectable service tiers in public wireless hot spot sites. The *airConn* framework has been designed to leverage rather than replace existing and future hot spot installations, at the same time requiring no modification of end-user devices.

Although necessary, to achieve the design goals for *airConn*, we found that simply using open standards is not sufficient. Instead, we found it necessary to consider the connection of a user device to the wireless network and the eventual access to services as two distinct procedures. The former procedure is more relevant to a network provider's responsibilities, while the latter procedure is more relevant to the actual service (e.g., content) provider. Both of these procedures are necessary to provide a complete end-to-end service enablement; typically they are both provided by the same entity, the WISP. However, to provide higher flexibility and differentiation of service offerings, this may not always be desirable. In a hot spot, it is entirely possible that a property owner will want to exercise close control on the services offered on their premises. Viewing network access and content access as two separate (although not necessarily independent) procedures permits a lot of flexibility as to how services can be packaged and offered in dynamically managed tiers. With this separation in mind, the *airConn* objectives were realized by empowering end users, when they dynamically select a desired service tier, to control in an immediate yet indirect and controlled manner the networking elements of an *airConn* installation.

Research in the areas of hot spot deployment and service offering and management is expected to continue to mature in coming years. By building our *airConn* testbed using off-the-shelf components, we came to the realization that it would take a reasonable amount of effort to set up a simple yet functional research environment. Given such an environment, research experimentation in this area can easily proceed. We envision research activities covering several areas in this space can be performed using and strengthening the *airConn* framework. These activities can span diverse areas, including access to the network like zero configuration, mobility, seamless roaming, and security; user management like transient users and user profiling; and content and service management like remote and local storage, service level agreements between property owners and content providers, and service brokering.

By assigning small enough leases, we control the "granularity" of determining the exact departure time of a user from the hot-spot. This approach takes advantage of the standard DHCP procedures where a device will automatically request renewal of its lease as the lease termination approaches.

References

[1] ISO/IEC 8802-11:1999, "IEEE Standard for Information Technology — LAN/MAN — Specific Requirements — Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," 1999.

[2] T-Mobile Wireless Broadband Network, http://www.tmobilebroadband.com/

[3] Wayport, Inc., http://www.wayport.com

[4] B. A Miller and C. Bisdikian, *Bluetooth Revealed*, 2nd ed., Prentice Hall, 2002.

[5] Gartner press release, http://www4.gartner.com/5_about/press_releases/pr30june2003a.jsp, June 30, 2003.

[6] V. Bahl *et al.*, "PAWNs: Satisfying the Need for Ubiquitous Secure Connectivity and Location Services," *IEEE Wireless Commun.*, Feb. 2002.

[7] IEEE Std 802.1X-2001, "IEEE Standard for Local and Metropolitan Area Networks: Port-Based Network Access Control," June 2001.

[8] netfilter/iptables Linux code, http://www.iptables.org

[9] A. Acharya *et al.*, "ts-PWLAN: A Value-Add System for Providing Tiered Wireless Services in Public Hot Spots, *ICC '03*, Anchorage, AK, May 11–15, 2003.

Biographies

Arup Acharya (arup@us.ibm.com) is a research staff member in the Autonomic Computing and Networking Services Department at IBM T.J. Watson Research Center. His current work includes VoIP, novel usage of SIP, and 802.11 based wireless mesh networks. His interests include emerging networking architectures in both wireline and mobile wireless networks, such as MPLS, IPv6, and 3G/4G networks. Before joining IBM, he was with NEC C&C Research Laboratories, Princeton, New Jersey from May 1995 to November 1999. He received his B.Tech degree in computer science and engineering from the Indian Institute of Technology, Kharagpur, and a Ph.D. in computer science from Rutgers University in 1995. He has published extensively in the areas of mobile, wireless, and wireline networking, participated in program and organizing committees of leading conferences as well as in standards bodies such as the IETF and ATM Forum. He has been awarded five patents with several others pending. Further information is available at http://www.research.ibm.com/people/a/arup/

Chatschik Bisdikian [F'04] (bisdik@us.ibm.com) holds a Ph.D. degree in electrical engineering from the University of Connecticut, Storrs. He is currently a research staff member at IBM T. J. Watson Research Center, Hawthorne,

By building our airConn testbed using off-the-self components, we came to the realization that it would take a reasonable amount of effort to set up a simple yet functional research environment. Given such an environment, research in this area can easily proceed.

New York. He has worked on the development and analysis of communications protocols for personal, local, and metropolitan area networks; he is currently focusing on the areas of pervasive communications and computing. During his career, he has authored over 90 peer-reviewed papers in the above mentioned areas, had eight patents issued, co-authored the book *Bluetooth Revealed* (Prentice Hall), and served on several conference program committees. He has been involved with development of the Bluetooth specification from its early stages and has served as vice-chair of the IEEE 802.15.1 task group that developed a standard for personal area networks adapted from the Bluetooth specification. He has guest edited various special issues in technical journals, including an issue on Wireless Personal Area Networks (*IEEE Network*, September/October 2001) and an issue on Ubiquitous Computing and Communications (*IEEE Personal Communications*, February 2000). He is currently serving as Editor-in-Chief of *IEEE Network*. He received the 2002 best tutorial award from IEEE Communications Society for his paper titled "An Overview of the Bluetooth Wireless Technology." He is a 1995 finalist of the Eta Kappa Nu Honor Society's Outstanding Young Electrical Engineer Award Program; and is a 2004 inductee of the Academy of Distinguished Engineers of the School of Engineering of the University of Connecticut, and an IEEE Fellow for contributions to the development, modeling, and analysis of communication protocols and wireless personal area networks. He has appeared in several listings, including *Who's Who in Science and Technology*, and is a lifelong member of the Eta Kappa Nu and Phi Kappa Phi Honor Societies.

YOUNG-BAE KO [M](youngko@ajou.ac.kr) is currently an assistant professor at the School of Information and Computer Engineering, Ajou University, Korea. Prior to joining Ajou University he worked at IBM T.J. Watson Research Center as a research staff member in the Department of Ubiquitous Networking and Security. He also worked as an intern at Hughes Research Laboratory, California, in 1999. He received his Ph.D. in computer science from Texas A&M University. His Ph.D. dissertation was on utilization of location information for routing and MAC problems in mobile ad hoc networks. His research interests are in the areas of mobile computing and wireless networking. In particular, he is actively working on mobile ad hoc networks, wireless sensor networks, wireless LAN with smart antennas, and various ubiquitous networking issues. He has served as a technical program committee and organizing committee member of various networking related conferences including ACM MobiCom, ACM MobiHoc, and IEEE ICC. He is a member of ACM, ACM SIGMOBILE, and IEEE Computer Society. See http://dmc.ajou.ac.kr/~youngko for further details.

ARCHAN MISRA [M] (archan@us.ibm.com) is a research staff member with the Pervasive Security and Networking Department at the IBM T. J. Watson Research Center. His current research efforts are related to middleware infrastructure for Internet-scale distributed pervasive computing, information-theoretic mobility management in smart wireless environments, and MAC protocols for wireless mesh networks. Prior to joining IBM in 2001, he spent three and a half years at Telcordia Technologies (formerly Bellcore), where he worked in the areas of IP-based mobility management, congestion control, and Internet QoS. He serves on the technical program committee of several conferences, and was a co-author on papers that received the Best Paper awards in ACM WOWMOM 2002 and IEEE MILCOM 2001. He received his Ph.D. in electrical and computer engineering from the University of Maryland at College Park in May 2000, and his B.Tech in electronics and communication engineering from IIT Kharagpur, India in July 1993. He is the current Untethered Network Technologies chair of the IEEE Computer Society's Technical Committee on Computer Communications (TCCC).