

Secure Wireless Internet Access in Public Places

Paramvir Bahl, Srinivasan Venkatachary

Microsoft Research
One Microsoft Way
Redmond, WA 98052-6399

Anand Balachandran

University of California, San Diego
9500 Gilman Drive
La Jolla, CA 92093-0114

Abstract – We have built a network, called the CHOICE network, which globally authenticates users and then securely connects them to the Internet via a high-speed local area wireless network. Our network provides easy-to-use, individual-centric, service-oriented wireless Internet access in places other than the traditional corporate offices and homes. Our architecture is hardware and protocol agnostic and is built on an easily deployable software module called the Protocol for Authorization and Negotiation of Services or PANS. PANS provides authorization, access, privacy, security, policy enforcement, quality of service (QoS) and accounting. In this paper, we describe PANS in detail. We discuss our system design and operation, implementation, and performance. We evaluate PANS and show that it is scalable and secure. Our network has been deployed and is operational at a local mall in Bellevue, Washington.

I. INTRODUCTION

We live in an increasingly fast-paced mobile society where the computing needs have crossed the boundaries of offices and homes and moved to a new paradigm: *anywhere, anytime* computation on *any device*. Consequently, there is much enthusiasm around the impending deployment and availability of the “third-generation” (3G) wide-area cellular networks. These networks are touted as the wave of the future because of their ability to support data networking at speeds of up to 384 Kb/sec per cell for a non-roaming user [1], where a typical cell is between 1 to 2 miles in radius. This means that the maximum achievable *raw data rate* that an individual can get with 3G networks is around 384 Kb/sec when he or she is the **only** user of the network in a 1-2 mile radius! However, in reality, there may be many users in the cell, in which case the average throughput that any one individual gets is substantially below the 384 Kb/sec advertised speed.

Today, wireless local area networks (LANs) can provide data connectivity at up to 11 Mb/sec per access point [2] [3], within 1 to 3 years they will provide access speeds of up to 54 Mb/sec [4]. We believe that as the Internet becomes increasingly multimedia-centric, users will need an infrastructure to connect to the Internet at speeds much higher than what the 3G systems can provide. However, such an infrastructure is not available today. An individual’s access to the Internet is restricted to the availability of an Internet Service Provider (ISP) or to a network that is provided and maintained by her employer. We believe that it is desirable to eliminate the dependence of Internet access on either or both of these elements.

We have built and deployed a network, called the CHOICE network, which provides a *choice* to individuals in how they access the Internet from “almost anywhere”. Almost anywhere includes places of congregation or public places such as shopping malls, airports, restaurants, libraries, and hotels. Using widely available standards-based wireless LAN technology [2] the CHOICE network provides painless Internet access to individuals who present the proper identification. We have deployed the CHOICE network in a local mall to offer Internet access to visitors and are using this pilot to carry out research on connectivity and computing in public places. Readers are referred to [5] for details about the deployment.

CHOICE is built on the *Protocol for Authorization and Negotiation of Services*, or PANS. PANS facilitates authentication, authorizes access, enforces policy and last-hop QoS, and provides privacy to network users and accounting to network operators. The user can be anywhere in the world and PANS can securely authenticate her credentials using a globally available database. Last-hop privacy and security is based on per-user dynamically generated varying-length keys, which are valid for a varying amount of time. In this paper, we describe the design, implementation, operation, and performance of PANS. Because of space limitations, we keep our description concise. We refer our readers to [6] for more details.

The rest of this paper is organized as follows: In Section II we describe the problems and difficulties in using current wireless LAN technologies in public spaces. In Section III, we describe the CHOICE network. We outline the system components, system operation, and implementation. In Section IV, we look at scalability issues and PANS performance. In Section V, we survey related work in the field. We describe on-going and future work in Section VI.

II. ISSUES IN SECURING WIRELESS LANS

For several years now in events such as the IETF and IEEE meetings, temporary wireless LANs are set up for attendees to connect to the Internet. Unfortunately, expanding this initiative to a wider setting in a public place has not been easy. This is because these networks are non-trivial to configure, do not guarantee adequate user privacy and security and are not impervious to malicious attackers.

Currently available wireless LANs limit themselves to the problem of user authentication, privacy and security via either (1) MAC level filtering and/or via (2) shared key

authentication and privacy, both of which are layer-2 mechanisms. In MAC level filtering the access point (AP) maintains a list of valid MAC addresses. For each incoming packet on the wireless segment, the AP checks the source MAC address against the list of valid addresses in the table. If there is a match, the packet is forwarded to the wired segment otherwise it is dropped. However, MAC level filtering is inadequate to secure the network from unauthorized users who may masquerade as a valid user through hardware address spoofing. Furthermore, this option does not scale with the potentially large number of users that may visit the public place. This then motivates the second method that is available today, which depends on using shared keys.

Wireless LAN standards such as the IEEE 802.11 [2] include an optional provision for authentication and privacy called the Wired Equivalent Privacy (WEP) function. WEP operates by encrypting the data before it is sent wirelessly using a 40-bit encryption algorithm known as RC4. The same key is used for both authentication and encryption/decryption of data; thus only wireless clients with the exact shared key can correctly decipher the data. Problems emerge when such networks are deployed in a public place because keys have to be deployed on a per-user basis to maintain security. Currently there are no simple mechanisms for generating and distributing keys for last-hop privacy. Further, even per-user keys have to be changed frequently since the algorithm can be broken in time [7]. Changing keys frequently is not convenient with current products.

A popular layer-3 mechanism for secure communication on the Internet that can also be used in wireless LANs is IP Security (IPSec) [8]. However, a system based on IPSec did not fully satisfy all of our requirements for the following reasons. First, we found that IPSec is not as widely available on all platforms as we would like it to be. Second, IPSec couples user keys and security association very tightly with IP level information. This directly impacts our goal of supporting roaming users whose IP address changes frequently. In the CHOICE network, as we will describe in subsequent sections, we identify users by a specific (key, token) pair that is completely de-coupled from IP level information and consequently, support for mobility with fast hand-offs is an integral part of the network. Fourth, we explored the possibility of IPSec tunnel mode between the mobile client and AP by treating it as an IPSec gateway. However we did not find a single vendor who currently supports IPSec functionality in the AP. Finally, we wanted a mechanism that was protocol agnostic, so we could support both IP and WAP [9] devices at the same time; IPSec is tightly linked to the IP protocol.

In summary, while current layer-2 wireless LAN technologies do not provide adequate levels of security and privacy, layer-3 technologies like IPSec are not widely available, do not support mobility and are not protocol agnostic. We were motivated by a desire to empower the mobile individual by providing her with choices when she

accesses the Internet wirelessly in a public place. To do this expeditiously, we concluded that a lightweight mechanism that provides authorization, access control, privacy, security, local mobility, accounting, and last-hop QoS was needed. Furthermore, we wanted a mechanism to be both protocol and hardware agnostic so we could support both IP and WAP devices and to run over legacy wireless hardware if needed. The CHOICE network that we describe below contains such a mechanism.

III. THE CHOICE NETWORK

We now describe the architecture, system components, system implementation, and system operation of the CHOICE network.

A. System Components

The CHOICE network has several components that manage address allocation, authentication, authorization, security, accounting, and last-hop QoS. These components, illustrated in Fig. 1, enable the user to gain secure access to the network as follows. The DHCP server [10] leases out an IP address to the client wishing to connect to the Internet. The client is authenticated by the global authenticator, procures a key from the PANS authorizer, and gains access to the network. We describe each component in detail below.

1) Global Authenticator:

We use MS Passport [11] as our global authenticator. Several factors motivated our choice of Passport. First, its wide availability enables us to establish a large user base. Second, all transactions with Passport are web-based thereby greatly enhancing the usability of the system for the layperson. Third, and most importantly, all these transactions are carried out over HTTPS, which is encrypted using the Secure Socket layer (SSL) [12]. This means that there is an

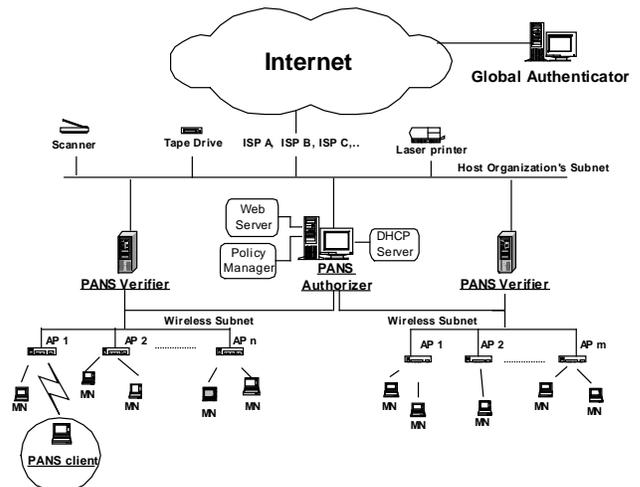


Fig. 1. The CHOICE Network Architecture (note: ideally the PANS Verifier should be resident at every AP)

end-to-end secure channel between the user and the authentication service. Even if PANS were to be set up by an un-trusted third party, this party cannot decrypt the user's name and password while it is being supplied to Passport.

2) Address Allocation and Naming:

The CHOICE network uses a standard DHCP server to lease out IP addresses to potential clients on the public sub-network. The IP address scope and the lease period are configured by the host organization at setup time. One drawback of DHCP is the limited scope within which the server can lease IP addresses. This problem can be overcome by using a Network Address Translator (NAT) [13].

The Web server is the user's entry point into the CHOICE network. It is through the Web interface that the user begins the authentication process. The CHOICE Web server is based on Active Server Pages (ASP) [14] and guides the user through the authentication process. Once the user obtains an IP address, she can access the Web server and easily download the network access software (See subsection 5).

3) PANS Authorizer:

The PANS Authorizer authorizes the client's access to the network, upon successful completion of authentication. In addition, it handles the task of determining service policies, generating keys, and communicating service levels and keys to the clients and to the PANS Verifiers (described below). The PANS Authorizer performs IP-level filtering based on the destination IP-address of each packet; any packet with a destination address other than the DHCP server, the Web server, or Passport is dropped.

Upon authentication, the PANS Authorizer does a look-up in its Policy Table to determine the users' service level L_n , generates a $(key, token)$ pair, and then communicates the service level and the $(key, token)$ to the PANS Client module residing on the users mobile host and to the PANS Verifier. In addition, the client and the verifier also get a key_id , which is an index into an array of valid $(key, token)$ pairs that have been given out to the clients. Each $(key, token)$ pair is valid for a finite amount of time after which the user may renew her identity and obtain a new pair. Alternatively, the user can request the authorizer to extend the validity of her existing key. The key is used for encryption/decryption while the token is the value that is tagged to every packet before encrypting it with the key. Once the user has been authenticated, all her communication is directed through the PANS Verifier, which has knowledge of all valid keys.

4) PANS Verifier:

The PANS Verifier handles verification, accounting, and policy enforcement on a per-packet basis for authorized users. The Verifier actively processes each packet that is sent out of the mobile host and runs on a much smaller time scale as compared to the Authorizer. The task of the PANS Verifier

includes checking if each packet from a client (identified by a unique key_id) contains the right $(key, token)$ combination that the PANS Verifier has in its table entries. In addition, the Verifier keeps an account of the number of packets per user it has serviced, for purposes of accounting.

5) PANS Client:

The PANS Client is resident on the user's mobile host and tags all outgoing packets with the $(key, token)$ pair obtained from the Authorizer after successful authentication. The downloadable PANS Client enables users to access CHOICE from anywhere without requiring any additional support on their mobile device or any modifications to the protocol stack.

B. Implementation and Operation

Having described the components that make up the CHOICE Network, we now describe the implementation details of the PANS protocol.

1) The PANS Intermediate Driver and User-level Module:

The implementation of PANS can be divided into an OS-dependent part and an OS-independent part. We implemented the OS-dependent part as an Intermediate Miniport driver within the Windows Network Driver Interface Specification (NDIS) protocol stack [15]. The modular design of NDIS allows us to write PANS as an NDIS Intermediate driver that plugs in to the stack seamlessly. We programmed PANS to manipulate packets delivered by NDIS from the protocol driver above, down to the NIC.

The OS-independent part of PANS is a user-level module, that handles the key and service-level exchanges with the PANS Authorizer after authentication. We have implemented the transfer of keys and service-levels using ASP scripts running on the Web server. Once it has received the key , the user-level module communicates it (along with the $token$ and key_id) to the intermediate PANS driver in the kernel using a simple `ioctl` call. The process of key exchange with the PANS Authorizer is identical for both the PANS Client and the PANS Verifiers. While the client module uses the information for encryption, the verifier module uses the same key for decryption once the packet is received at the PANS Verifier.

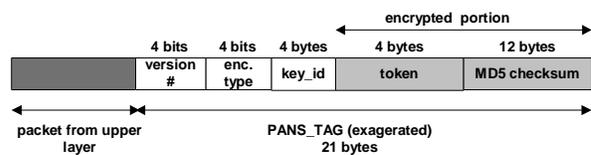


Fig. 2. The PANS_TAG showing the different fields. The version number, encryption type and key_id form the unencrypted portion, while the token and MD5 checksum are encrypted using the encryption algorithm specified under the encryption type.

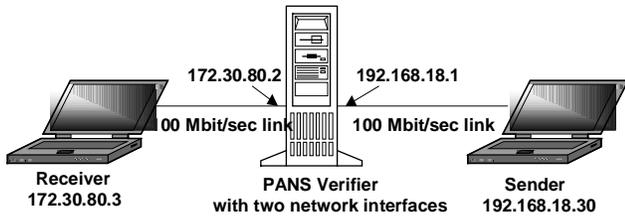


Fig. 3. Experimental setup used for performance studies on the PANS Verifier

2) The PANS Protocol:

PANS is deployed as a software module in the client and the verifiers. The user-level module handles the web-based exchange of keys and tokens, which in turn are handed down to the PANS driver.

Each successfully authenticated user is given a $(key, token)$ pair and a key_id : a combination that is used in all future network transactions. This unique combination forms part of the tag that is appended to every packet that goes out from the client's mobile host. The typical structure of the PANS protocol tag is shown in Fig. 2. The PANS_TAG is composed of two parts, an unencrypted part and an encrypted part. The first unencrypted part contains the version number, the key_id , and encryption type. The encrypted part contains the token (given by the PANS Authorizer) and an MD5 checksum of the data and the PANS_TAG. The checksum field is filled with zeros prior to its computation. The purpose of the MD5 checksum is to ensure authenticity of the data origin and to prevent the packet from being modified in transit. This information is encrypted using the secret key that is part of the $(key, token)$ pair obtained from the authorizer. We use the triple-DES algorithm [16] for encryption. However, our implementation is modular enough to accommodate any other encryption algorithm as indicated in the *encryption-type* field.

IV. PANS PERFORMANCE

The task of per-packet verification by decrypting a packet and subsequently checking for the client's valid signature in the PANS_TAG, adds value to the CHOICE network and makes it intrusion-proof. However, intercepting a packet that is in transit through the network could increase the latency incurred in its transmission and also degrade the throughput of the system. To measure the overhead of the system, we ran benchmark tests on the PANS Verifier to measure its throughput, CPU utilization and the effect of PANS on the packet round trip time.

A. Experimental Setup

We used a common simplified setup for all our experiments. This setup is shown in Fig. 3. Specifically, we used a wired 100 Mb/sec connection between each of the

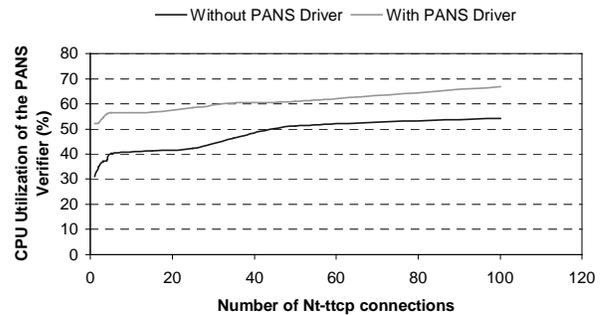


Fig. 4. The CPU Utilization of the PANS Verifier as a function of the number of `nttcp` connections.

systems. We did this to push more data on to the network and potentially stress the PANS Verifier to its limit. We assigned static IP addresses to the client machines. The two programs in our benchmark were `nttcp`, a port of `tcp` [17] to Windows, and `tcpperf`.

B. Throughput and CPU utilization of the PANS Verifier

In our first experiment, we used `nttcp` to measure the throughput and CPU utilization of the PANS Verifier. To measure throughput, we used 20K buffers of size 64K each. These parameters were sufficient to drive the system at the link bottleneck bandwidth of 100 Mb/sec. We repeatedly executed `nttcp` by varying the number of connections between sender and receiver and measured the total throughput as recorded by the receiver machine. In the first run, the client and Verifier machines did not have the PANS driver installed. During the second run, both machines had the PANS driver installed. Comparing the two runs, there was no measurable difference in throughput.

We used a performance monitor on the PANS Verifier to record the average CPU utilization for the duration of each run of `nttcp`. Our results are shown in Fig. 4. From this figure we see a 40% difference increase in the average CPU utilization in the presence of the PANS intermediate driver.

C. Effect of PANS Verifier on packet RTT

For round-trip time measurements we used `tcpperf`. We flooded the network with 100,000 buffers, varying the packet size during each run. The plot in Fig. 5 shows the variation of per-packet RTT with the buffer size sent. The RTT values were compared in the presence and absence of the PANS intermediate driver. From the plot it can be seen that the per-packet RTT difference is in the order of tens of microseconds, which is not very significant.

Our experiments were conducted using the default buffer size, which simulates bulk data transfers. We noticed that the 100 Mb/sec backbone link gets saturated before the Verifier does. We measured the throughput of the 11 Mb/sec IEEE 802.11 wireless LAN AP from a couple of different vendors and found that the average throughput for these networks is

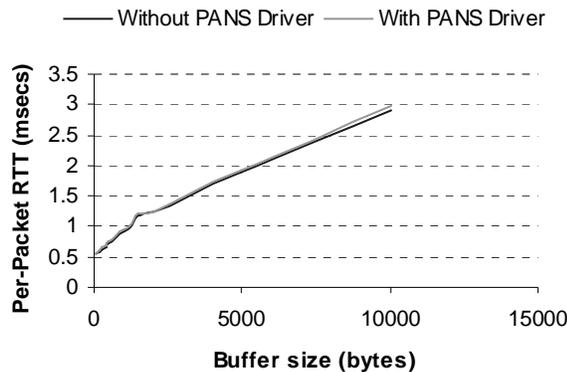


Fig. 5. Effect of PANS Verifier on packet RTT.

around 6 Mb/sec. This means that at least for the case of bulk transfers where the wired network is the bottleneck instead of the CPU, one single Verifier can handle at least 10 APs.

V. RELATED WORK

We are aware of a four other systems that address some of the problems that CHOICE tackles.

The only fully deployed system that we are aware of is the SPINACH system developed at Stanford [18]. SPINACH enforces last-hop security by keeping a log of the (IP, MAC) address pair of every user who has been successfully authenticated and filtering incoming packets based on this tuple. Unfortunately, this model does not protect against MAC-address spoofing. The architects of SPINACH note that a way to avoid address spoofing attacks is to use a stronger mechanism like IPsec for authentication

A system proposed at UC Berkeley, is built around an “*authenticated DHCP*” server by dynamically enabling and disabling access to network ports [19]. In addition to requiring specialized hardware, this solution is not a viable option for authenticating wireless users whose presence or absence in the network cannot be detected. Like SPINACH, this design does protect against MAC-address spoofing.

One more recent and promising approach is the IEEE 802.1X standard’s port-based network access control, which does authentication by carrying the *Extensible Authentication Protocol* (EAP) frame within the Ethernet frame [20]. Implementing EAP-based authentication requires specialized hardware in the access points. Further, an attacker can replace the AP with its own rogue AP and perpetrate a dictionary attack to recover the user’s password.

The CMU NetBar system is yet another proposal in which a NetBar port remains isolated on a “non-connected” VLAN until the user at that port is authenticated [21]. This system relies on an expensive specialized hardware switch, and is not secure from MAC-address spoofing attacks.

VI. CONCLUSIONS AND FUTURE WORK

In this paper we argue that high-speed wireless LANs deployed in public places are ideal for wireless connectivity to the wide-area Internet, and we describe how the CHOICE network provides such a service. CHOICE, and its underlying protocol PANS, is a comprehensive system that is secure, protocol agnostic, hardware agnostic, and self-contained. We evaluate PANS and show that it is secure and scalable. We compare our design to other documented alternatives and make the case for why the CHOICE network should be adopted and deployed. We have deployed CHOICE in a local mall in Bellevue, Washington, to offer Internet access to visitors. As part of ongoing work we are extending CHOICE to incorporate dynamic mobility management of hosts between public and private networks [22].

REFERENCES

- [1] ITU-R Rec. M. 1225, “Guidelines for evaluation of radio transmission technologies for IMT-2000.”
- [2] IEEE 802.11b/D3.0, “Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specification: High Speed Physical Layer (PHY) extensions in the 2.4 GHz band,” 1999.
- [3] Aironet Wireless Communications Inc, “Developer’s reference manual: PC4500/PC4800 PC card Wireless LAN adapter,” 1999.
- [4] R. V. Nee, G. Awater, M. Morikura, H. Takanashi, M. Webster, and K. W. Halford, “New high-rate wireless LAN standards,” *IEEE Communications Magazine*, Vol. 37, no. 12, pp. 82-88, Dec. 1999.
- [5] MSCHOICE: <http://www.mschoice.com>.
- [6] The CHOICE Network – Broadband wireless Internet access in public places, MSR-TR-2000-21, Feb. 2000.
- [7] “Single computer breaks 40-bit RC4,” Jan. 1996, <http://www2.ecst.csuchico.edu/~atman/Crypto/misc/netcape-cebreaker.html>
- [8] R. Atkinson, “Security architecture for the Internet Protocol”, *IETF RFC 2401*, Nov. 1998.
- [9] The Wireless Application Protocol (WAP) White Paper, <http://www.wapforum.org/what/whitepapers.htm>.
- [10] R. Droms, “Dynamic Host Configuration Protocol,” *IETF RFC 2131*, Mar. 1997.
- [11] MSPassport : <http://www.passport.com>.
- [12] T. Elgamal, S. Cotter, and the Netscape security team, “Netscape Security: Open-standard solutions for the enterprise, 1998”, <http://developer.netscape.com/docs/manuals/security/scwp/index.htm>.
- [13] Y. Rekhter, B. Moskowitz, D. Karrenberg, G. de Groot, “Address Allocation for Private Internets,” *IETF RFC 1597*, Mar. 1994.
- [14] MSDN Active Server Pages tutorial, Dec. 2000.
- [15] P G. Viscarola and W. A. Mason, “Windows NT device driver development,” *OSR Open System Resources*, 1999.
- [16] National Bureau of Standards, “Data Encryption Standard” *FIPS PUB 46*, January 1977, and “DES modes of operation,” *FIPS PUB 81*, Dec. 1980.
- [17] R. Stine, “FYI on a network management tool: Catalog tools for monitoring and debugging TCP/IP Internets and interconnected devices,” *IETF RFC 1147*, Apr. 1990.
- [18] G. Appenzeller, M. Roussopoulos, and M. Baker, “User-friendly access control for public network ports,” *Proceedings of INFOCOM '99*, Mar. 1999.
- [19] D. L. Wasley, “Authenticating aperiodic connections to the campus network,” Jun. 1996.
- [20] *IEEE draft*, “Port-based network access control,” Sep. 1999.
- [21] E. A. Napjus, “NetBar - Carnegie Mellon’s solution to authenticated access for mobile machines,” CMU White Paper, Dec. 1996.
- [22] Allen Miu and Paramvir Bahl, “Dynamic host configuration for managing mobility between public and private networks,” Third Usenix Symposium on Internet Technologies and Systems, USITS’01, in press.