

Public-Key-based Secure Internet Access

Daniel B. Faria
dbfaria@cs.stanford.edu

David R. Cheriton
cheriton@cs.stanford.edu

Computer Science Department
Stanford University
Stanford, CA 94305-9040

1. INTRODUCTION

Internet access is an important and expected amenity in many settings. For example, a typical professional employee has a laptop with an WiFi LAN card, allowing this laptop to access the Internet at work, at home and in other locations such as airports, cafes and companies the employee may visit. However, the traffic to and from his or her laptop should be secure from others even if the user is just accessing an airline web site to reschedule a next flight.

Various approaches have been proposed without seriously addressing the general secure Internet access problem. The IEEE 802.11 standard defined the WEP protocol, which aimed to provide privacy and authentication for wireless users, but lacked any means of key distribution. Consequently, to date, most installations are insecure because they either use no encryption or use a common key statically configured across many hosts. To make matters worse, recent work as pointed out many flaws in the WEP encryption scheme [6, 3, 2].

There has been some effort to provide secure wireless access, as WiFi installations are being rapidly deployed. However, much of the work has been placed around the 802.1X [1] specification, which apparently encourages a diversity of solutions at the higher-level, essentially creating incompatibility between different networks.

Current practice is to physically secure the enterprise network by hiding the wires inside walls and securing the switches and routers in locked wiring closets. We view this portion of the network as the true *intranet*. The only points of physical exposure are the RJ-45 ports in the wall outlets and the wireless access antennae distributed around the campus. We focus on providing secure Internet access at these points of physical exposure of the intranet, which we call the *public access network*.

This work describes a protocol architecture providing secure Internet access that solves the security vulnerabilities while providing ease of use, transparency, flexibility, and interoperability between networks with different address prefixes. A distinctive aspect to our approach is basing the

request for access on names and public-key-level identification of the requesting host. Authentication is performed by the SIAP protocol, while the SLAP protocol provides message confidentiality, integrity, and authentication. A prototype implementation and measurements thereof indicate it is feasible to implement with acceptable performance even without hardware support.

Compared to the solution composed by 802.1X and WEP, our approach extends the services provided in many significant ways. First, the specification of the SIAP protocol permits interoperability between domains, mutual authentication between the mobile client and the access points in the network, and user-transparent mobility between networks with different IP prefixes. Second, by coalescing authentication and IP address assignment, SIAP enables the implementation of different network views based on the IP address given to the client and also avoids the DoS attacks that can be performed against DHCP. As the client's IP address becomes tied to its session key, the APs can identify and block any kind of address spoofing. Third, SIAP defines a client-driven state propagation mechanism that eliminates the need for an inter-AP protocol and prevents the propagation of state to access points not reachable to the client. Finally, SLAP services extend WEP¹ services by making its services link-layer independent and providing distributed replay detection.

2. PUBLIC-KEY-BASED SECURE INTERNET ACCESS

SLAP, the *Secure Link Access Protocol*, is a protocol located just above the link layer, intercepting and processing all incoming and outgoing frames and providing a secure tunnel between the wireless host and the access point (AP). Given a per-client state consisting of MAC address, IP address, and session keys, SLAP performs its services over all outgoing frames and reconstruct frames sent by its peer entity. SLAP services include encryption, per-packet authentication, and replay detection. In order to set up this per-client state in both the client's laptop and in the neighboring access points, an application-layer authentication protocol is used, called *Secure Internet Access Protocol* (SIAP). The protocol stack is shown in figure 1.

SIAP is responsible for providing the authentication service using RSA public keys. The SIAP client present in a laptop performs a three-message handshake with the SIAP

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MOBICOM'02, September 23–28, 2002, Atlanta, Georgia, USA.
Copyright 2002 ACM 1-58113-486-X/02/0009 ...\$5.00.

¹consider a version of WEP with none of the published vulnerabilities.

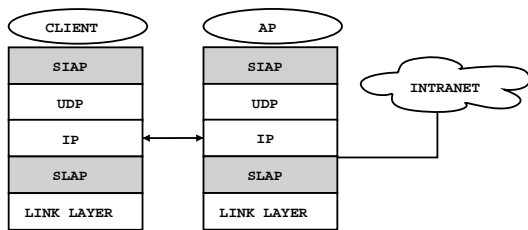


Figure 1: Protocol stack.

server in the access point. This handshake provides mutual authentication and provides the client with the IP address to be used and the session keys associated with it. From this point on, the client's MAC and IP addresses and the session keys are tied together, and its correct use is enforced by the SLAP entity in the access point.

The SLAP module waits for the SIAP entity to perform the authentication process and inform SLAP about the security state to use. After that, all the frames sent between client and AP receive the SLAP services with the session keys just established. By placing SLAP over the link layer, we make it technology independent. SLAP uses AES in CTR mode and HMAC-MD5 to provide confidentiality and message authentication services and also implements a replay detection mechanism.

3. PRELIMINARY RESULTS

Both SLAP and SIAP have been implemented in Linux 2.4. SLAP was implemented as a loadable kernel module while SIAP was implemented as an UDP-based application, using OpenSSL to provide the public key encryption operations. A SIAP server running in an access point listens to a well known port, which is known by the clients. Clients also listen to a predefined port, used by servers when sending advertising messages.

Our testbed is composed by a laptop that works as the client and a desktop computer that plays the role of the access point. The laptop is a 333-MHz Pentium II computer, with 64 MB of main memory, and a FastEthernet 100Mbps card. The desktop computer runs with a 900-MHz Duron processor, 256 MB of memory, and contains two FastEthernet network interfaces. The FastEthernet cards were used to connect the client to the access point in order to provide higher available bandwidth in the last hop.

The measurements show that the operations involving an RSA private key are very demanding, incurring overheads in the order of tens of milliseconds. The first consequence of these high costs is that the authentication handshake takes in the order of 400-600ms to finish. This delay may affect the throughput on the SIAP server, but may have no impact on how smoothly the SIAP client switches from one network to another, as it can authenticate with the second network and get a second IP address before it performs the handoff and possibly deletes its state in the previously used network.

The overhead incurred by SLAP in both directions varies between $50\mu\text{s}$ and $460\mu\text{s}$ in our current test bed. This means that the round-trip time (RTT) between the laptop and a server in the Internet can be increased by up to almost 1 millisecond for large frames. As small packets are predominant in local wireless networks [4] and measurements performed

in Internet backbones show that 175- and 400-byte average packets are common [5], we expect this 1 millisecond increase to be rare.

To quantify the impact of this RTT increase over real applications, we performed several long (50 MB) file transfers using FTP. When using a server with a RTT of 1 ms from the wireless host, the total download time was increased by 17%. This increase drops to 7% when using a 40ms-away server, which we believe to be a more representative scenario. We expect SLAP services to incur an even smaller overhead as code optimizations are performed and no noticeable overhead as hardware implementations are used.

4. CONCLUSION

Secure Internet access is an important facility to provide as the use of mobile Internet devices increases. SIAP provides a simple protocol solution that is efficient, secure, flexible and convenient for the end user. It avoids the denial-of-service and security openings that are problematic with DHCP. SIAP and SLAP allow relatively simple layer 2 devices while ensuring security of access. The name basis for identification allows a site to assign IP addresses to newly arrived hosts to classify them as visitor or employee and then tunnel packets accordingly.

SIAP and SLAP provide an attractive alternative to the approaches to secure access than have been attempted with 802.11b, including WEP and 802.1X. They are either insecure or inflexible and both seem to require a comparable amount of mechanism in the access points to the architecture described here. Moreover, SIAP/SLAP use AES-based encryption, proven PKE technology and higher-level protocol design, avoiding the security weaknesses that have plagued link-level efforts. The performance results presented show that a software implementation is viable to be used as a temporary solution, obtaining performance suitable for current 11Mbps wireless networks.

5. REFERENCES

- [1] LAN MAN Standards Committee of the IEEE Computer Society. Standard for Port based Network Access Control. Technical Report Draft P802.1X/D11, IEEE Computer Society, Mar. 2001.
- [2] W. A. Arbaugh, N. Shankar, and Y. C. J. Wan. Your 802.11 wireless network has no clothes. March 2001.
- [3] N. Borisov, I. Goldberg, and D. Wagner. Intercepting mobile communications: The insecurity of 802.11. In *Proceedings of the Seventh Annual ACM/IEEE International Conference on Mobile Computing and Networking - Mobicom'01*, pages 180-189, July 2001.
- [4] D. Tang and M. Baker. Analysis of a local-area wireless network. In *Proceedings of the Sixth Annual ACM/IEEE International Conference on Mobile Computing and Networking - Mobicom'00*, pages 1-10, Boston, MA, USA, Aug. 2000.
- [5] K. Thompson, G. Miller, and R. Wilder. Wide-area internet traffic patterns and characteristics. *IEEE Network*, 11(6):10-23, Nov. 1997.
- [6] J. Walker. Unsafe at any key size: An analysis of the WEP encapsulation. Technical Report 03628E, IEEE Standards 802.11 Committee, March 2000.