Phishing in the Wireless: Implementation and Analysis

Ivan Martinovic, Frank A. Zdarsky, Adam Bachorek, Christian Jung,
and Jens B. Schmitt

disco | Distributed Computer Systems Lab
University of Kaiserslautern, Germany
{martinovic,zdarsky,a_bacho,c_jung82,jschmitt}@informatik.uni-kl.de

Abstract. Web-based authentication is a popular mechanism implemented by Wireless Internet Service Providers (WISPs) because it allows a simple registration and authentication of customers, while avoiding high resource requirements of the new IEEE 802.11i security standard and backward compatibility issues of legacy devices. In this work we demonstrate two different and novel attacks against web-based authentication. One attack exploits operational anomalies of low- and middle-priced devices in order to hijack wireless clients, while the other exploits an already known vulnerability within wired networks which, in dynamic wireless environments, turns out to be even harder to detect and protect against.

1 Introduction

Taking into consideration the tremendous growth of public Internet access, one can easily see that IEEE 802.11 [1] networks have played a major role during recent years. High transmission rates, low costs, and simple deployment have all resulted in a high number of *hotspots* that are now offering wireless Internet access in coffee shops, airports, libraries, conferences, hotels, etc. For example, one of the major German WISP states that it operates more than 25,000 domestic and international hotspots that customers may use in order to roam and access the Internet worldwide.

Parallel to the popularity of wireless LAN technology, the topic of its security gained a similar, although rather negative publicity. The tragic end of Wired Equivalent Privacy (WEP) [8, 4] and the simplicity of various DoS attacks on a wireless medium have resulted in giving up security at the logical-link layer and shifting it to upper layers (or in the best case leaving it within virtual private networks (VPNs)).

Although WLAN's new security standard IEEE 802.11i [2], which was ratified in 2004, provides mechanisms for strong mutual authentication, data integrity, and data confidentiality, its deployment and utilization have not followed the same growth. Therefore, IEEE 802.11i is still not widely utilized, its strong security services often require new hardware and extension of the already existing infrastructure, while most of the handhelds have not yet been certified according to the standard. As a result, WISPs incorporate proprietary security solutions that can easily be implemented within their infrastructure and business models, providing a higher usability and lower complexity for customers, but on the other hand customers are expected to take care of the security themselves.

In a popular scenario of public hotspots provided by WISPs most security services are reduced to a simple access control mechanism which is implemented through a web-based authentication. For example, in most of the WISPs that we have analyzed, the only requirement placed upon a customer is to have a "wireless-enabled mobile device, BSSID set to a WISP, and Internet-ready web browser". No additional software is required. Every user can associate himself with the WISP's access point and by launching his Internet-browser he will be redirected to a login page. A customer can use that page to either authenticate or create a new account by paying for wireless access with his credit card. Upon a successful login, his Ethernet address (MAC address) will be authenticated and allowed to access the Internet. This simple solution does not require any knowledge of digital certificates, signatures, or any other security mechanisms. Yet, as we will show in this work, there is a price to pay for this simplicity.

We show and analyze two different attacks on web-based authentication. The goal of both attacks is to impersonate a legal AP and to inject a fake web page asking the user for his credentials. The first attack targets the access points and can in certain cases result in operational anomalies allowing the attacker to "steal" new clients. This attack is described and discussed in Section 2. The second attack focuses on the wireless clients and is based on a well-known vulnerability within wired networks. Unlike in wired networks, however, it shows to be still fully exploitable today and fatal on every wireless client, especially in conjunction with web-based authentication. The second attack as well as various countermeasures are discussed in Section 3. The related work on this subject is presented in Section 4. Section 5 concludes this work.

2 Flooding-based Client Hijacking

By executing a DoS attack, the aim of the attacker is to exhaust the server's resources which would then result in the server's inability to provide any service at all. On one hand, this can remain the main goal of the attacker and on the other hand, it can serve as a starting point for the execution of even more sophisticated attacks. A similar attack can also be started against web-based authentication provided that the attacker is able to disrupt an original service offered by a WISP's AP and during the attack a fake service is provided to wireless users. IEEE 802.11 networks have been subject to this type of attack from the beginning of their deployment. A common wireless attack based on rouge AP is traditionally executed by installing the rouge AP with a stronger signal and disguising it with the same BSSID as the legal one. The fact that a wireless client automatically connects to the AP with a stronger signal would then be abused so as to hijack the wireless station. Back in 2001/2002 various tools were enabling a DoS attack based on flooding an AP with authentication requests. In most cases the AP would crash or freeze and only a hard reset would help. Today, most access points provide a "DoS protection mechanism" based on a reduced rate of allowed authentication requests and should no longer be vulnerable to this type of attack.

2.1 IEEE 802.11 Association Process

Before going into more detail, we briefly summarize the functionality of IEEE 802.11 networks operating in infrastructure mode.

The infrastructure mode of IEEE 802.11 contains an access point which provides certain control and management functionalities. An access point takes care of accepting only the data traffic of wireless stations that are in a valid connection state. A wireless station can be in three different connection states: *initial* state, *authenticated but not associated* state, and *authenticated and associated* state. In order to send or receive data frames, the wireless client must be in the third state, i.e. in the authenticated and associated state.

A successful authentication is realized by sending an authentication frame in which one of two different authentication algorithms can be chosen: Open-System authentication (meaning no authentication at all) or Shared-Key authentication. Since the introduction of IEEE 802.11i and as a consequence of WEP being completely broken, Open-System authentication is now the only mandatory IEEE 802.11 authentication algorithm. Therefore, an authentication frame can no longer provide any authentication functionality but mainly serves to bring the wireless station into the second state. After a successful authentication, the wireless station proceeds by sending an association frame by which the association procedure is being finalized. From that moment onward, a wireless client is able to receive and send data.

On the other side, if an AP detects frames coming from a wireless client that is not in a valid state (with respect to the frames it is sending), the AP will respond with either a deauthentication frame or a disassociation frame, depending on the state the client managed to reach. This mechanism is important for two reasons; the first is to help a wireless station to re-authenticate itself in case it is in the wrong state and the second is to mitigate the possibility of an AP impersonation. For example, if a fake AP uses the same MAC address of a legal AP to steal wireless clients, the legal AP will then respond with a deauthentication frame to every client that starts to communicate with the fake AP.

2.2 Exploiting Operational Vulnerability

Although no real authentication takes place during the association process, an AP still needs to reserve resources to keep state about every wireless client. Common DoS attacks make use of this fact to fill up authentication table by flooding the AP with fake authentication requests. Eventually, this results in a total crash of the AP after which only a physical reset could help. An attacker may then use its own fake AP to impersonate a legal one. One should assume that this kind of DoS attack should no longer be feasible on modern equipment. With the aim of investigating this matter we have collected 6 different access points dating from 2003 to 2006 that were chosen based on their popularity and price (all of them with the latest firmware upgrade as provided by the manufacturer). For legal purposes, we shell keep the vendor and product names of the selected APs anonymous and therefore only describe price classes:

– Class 1: low-priced access points ($\leq$50 USD). Two APs, produced in 2003, 2004.

– Class 2: middle-priced access points (from 50 USD to 100 USD). Two APs, produced in 2004, 2006.
– Class 3: high-priced access-points (from 350 USD and higher). Two APs, produced in 2004, 2006.

To analyze the AP's behaviour we have flooded each AP with approximately 50 authentication requests per second. Since no significant differences in the operation of APs within the same class were detected, we select one AP from each class to describe it in more detail.

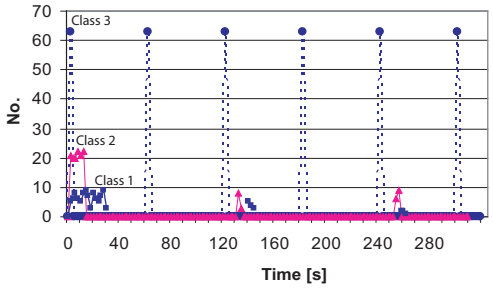

Fig. 1. Time Trace of Successful Authentications

Figure 1 shows the behaviour of one representative of each class under the flooding attack. As it can be seen, the most expensive AP (Class 3) allows 63 new authentication requests every 60 seconds. It is interesting to mention that both APs from Class 3, after allowing a certain number of requests, refuse to send any further response. This violates the IEEE 802.11 standard which mandates replies with appropriate reason codes to notify wireless client of unsuccessful authentications. This minor deviation from the standard introduces a certain performance degradation for wireless clients, because clients wait for the maximal response timeout before trying to authenticate again (we have observed that certain clients wait up to 7 seconds before retrying to authenticate).

In contrast to Class 3, both other classes accept various numbers of requests approximately every 2 minutes. Furthermore, they notify wireless clients if the authentication request has not been accepted by sending an "unsuccessful authentication" response. On the other hand, both classes have a high decrease in the number of accepted requests after initial admission, thus it seems that the flooding attack still impacts their resource management. Especially interesting is the longer period of time during which both other classes accept authentication request (e.g. one of the Class 2 APs accepts 126 authentication requests within the first 12 seconds and one of the Class 1 APs accepts 95 new requests within the first 30 seconds).

To analyze these phenomena in more detail, we have measured the delay between authentication requests and responses before and during the flooding attack. The attack rate remains the same with approximately 50 authentication requests per second, which

implies an attacker throughput of about 1.5 KByte/s. The flooding attack started after 20 seconds of normal operation. We have found that both Class 1 and Class 2 APs suffer from an operational anomaly that causes an exceptionally high delay between the authentication request and the authentication response (see Figure 2). After only 8 seconds of flooding, the response delay increases to 12 seconds. This is in contrast to all Class 3 APs where the authentication response delay remains stable with a mean of 1.6 ms and a standard deviation of 3 ms.

From a security perspective all three classes of APs have a potential vulnerability. Class 3 APs only respond to accepted authentication requests, leaving all other wireless clients to wait for authentication responses for a client-dependent period of time. This fact can be exploited by an attacker who uses a fake AP with the same MAC address to answer authentication and association requests as successfully. As a result, wireless clients associate with the fake AP instead of the legal one.

The two other classes, although answering all requests, suffer from a high delay during the flooding attack by which only after 12 seconds an authentication response reaches a wireless client. Similarly, the attacker can also answer those requests before the legal AP.

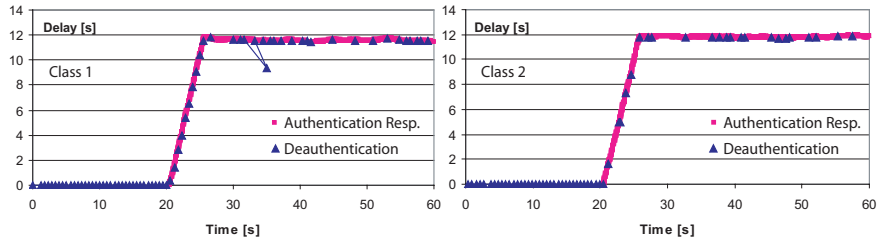

Fig. 2. Authentication and Deauthentication Delay

One last barrier for the attacker is the Deauthentication frame which is sent every time a data frame from an unauthenticated client is detected by the AP (as explained in subsection 2.1). We have measured the delay for this frame under the same experimental settings and only Class 3 APs are able to send the Deauthentication frame on time, meaning without any significant delay (mean response time for Deauthentication frame is 1.04 ms with a std.dev. of 1.7 ms). Both other classes have an increased Deauthentication delay which follows the Authentication delay as depicted in Figure 2. These results show that Class 3 APs, although deviating from the IEEE 802.11 standard, do not appear to have a security vulnerability due to their prompt response with a Deauthentication frame in case of their impersonation. This, regrettably cannot be said for their cheaper relatives.

2.3 Attack Implementation

In this subsection we are particularly interested in exploiting the aforementioned anomalies in order to implement an AP impersonation attack. The scenario remains the same as described in the motivation. The attacker's objective is to impersonate a WISP's access point and to inject a fake web page to a wireless client.

Discovered delays of Class 1 and Class 2 APs enable us to fully disguise the fake AP as a legal one. In the following steps we describe our implementation:

1. An attacker consists of a laptop running a web server and two wireless interfaces. One of the interfaces is set to a master mode in order to enable the access point's functionality (called a fake AP) while another one is used to start the flooding attack. The web server responds to all HTTP requests sent by a user and contains the same web page as the one of the WISP.
2. The MAC address of the fake AP is set to correspond to the MAC address of the legal AP using the same BSSID. The attacker starts flooding the legal AP.
3. After the legal AP has increased its authentication and deauthentication delay, the fake AP starts answering every request sent by wireless clients.
4. The attacker captures HTTP requests and responds with a fake web page (it can also choose to respond to any other control packet like ARP, DNS, DHCP,...).

As a result of this attack, we were able to authenticate and associate every wireless client with the fake AP. As assumed, all Class 1 and 2 APs did not detect the impersonation and the wireless clients successfully established a connection with a fake AP before first Deauthentication frames from a legal AP arrived and deauthenticated the client. In order to analyse the quality of the connection between a wireless client and the fake AP, we have measured both UDP and TCP throughputs (shown in Figure 3). The UDP sender rate was set to 5 Mb/s.

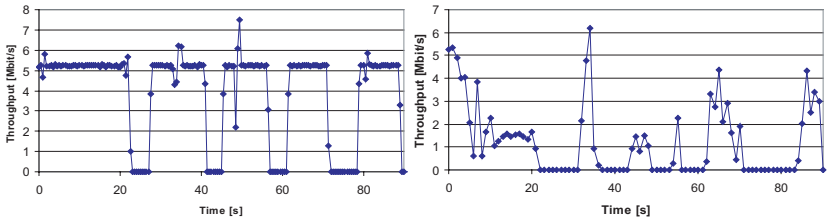

Fig. 3. UDP (left) and TCP (right) Throughput

Figure 3 shows that during the first 22 seconds the communication between the fake AP and the wireless client is undisturbed by the legal AP. We were able to intercept all requests and successfully redirect the wireless client to the fake web page without noticing any quality loss or other indication of the attack. Following that, the channel was influenced by Deauthentications frames sent from a legal AP. By receiving a Deauthentication frame, the wireless client would disconnect and immediately

try to reconnect. Again, the fake AP was the first to react and the connection was re-established. This can clearly be seen in the UDP traffic where the connection is disrupted by slots where the client is not connected. Although this frequent re-connection disturbs the link-layer connection, the transport layer still provides a connection. The TCP throughput on the right shows a trace of SSL traffic between the wireless client and the fake web-server which was used to present a fake authentication login page similar to those from WISPs.

Table 1. Various UDP Rates and Measured delay

UDP Throughput [Mbit/s]	1	2	3	5
delay (mean) [s]	13.23	14.40	20.17	32.97
std. dev.	1.75	1.96	7.07	12.20

Another interesting question that occurs is why the delay presented in Figure 3 is higher then the one initially measured between authentication request and response (12 seconds). The reason is that the delay strongly depends on the traffic sent to the AP. In Table 1, we have used different UDP sender rates and 15 repetitions for each level of UDP throughput. It turned out that increased traffic highly incresed AP's response delay and delay variance.

2.4 Discussion

In this attack we have shown how simple it is for an attacker to fully impersonate a legal AP. It also shows that web-based authentication is highly vulnerable, meaning that the users but also providers should be more careful in using and providing such an authentication method. This attack was possible on all low-priced and middle-priced access points. Only the most expensive class of access points was immune to this kind of attack. In our opinion, this is an important fact because a low price of IEEE 802.11 technology is often considered to be one of its most mentioned advantages.

Another question that arises is how realistic this attack can be? On one hand, an attacker is able to spoof a web page, but on the other hand he still cannot fake an original WISP's digital certificate. This is a well known issue and although most of today's attacks, from fake emails to phishing web-sites, are technically solved, it is also well-known that the most effective and successful attacks are the ones based on abusing human naivety [13].

3 Wireless ARP Attack

In contrast to the attack described in the previous section which is based on attacking APs, in this section we describe an attack which focuses directly on wireless clients. It is based on the well-known idea of ARP spoofing, which although considered to

be solved within wired networks, can be fully exploited within wireless networks. We show that by tweaking certain IEEE 802.11 frame parameters, a novel wireless ARP spoofing attack can be mounted which is hard to detect. Moreover, even a well-administrated infrastructure with ARP spoofing protection based on packet analysis cannot help in securing the wireless part of the network. As a result, the simplest solution against this attack is to abandon web-based authentication and to use the logical-link layer protection provided, e.g., by the IEEE 802.11i security standard.

3.1 Good Old ARP Poisoning

The Address Resolution Protocol (ARP) is used to resolve an IP address to a 48 bit Ethernet address (MAC address). It is a simple protocol consisting of the sender's IP address and sender's MAC address as well as the target's IP address (which is known to the sender) and MAC address (which is unknown). Since the ARP request is being sent as broadcast, it will therefore be received by every host on the same network. The host with the target IP address will then respond with an ARP reply containing his Ethernet address as a target MAC address.

By replying to ARP requests with an ARP replay containing a fake target MAC address, an attacker can simply redirect client's traffic to itself. This is why the ARP protocol has served as the basis for many different Man-In-The-Middle and DoS attacks mostly focused on switched (wired) networks.

The simplicity and frequency of ARP spoofing attacks in wired networks has resulted in a wide-spectrum of solutions that can detect and avoid the problem of fake ARP replies (existing solutions against ARP spoofing will be discussed later in subsection 3.3). Nevertheless, in contrast to a wired infrastructure, wireless environments are considerably different in their nature. Most importantly, public hotspots are characterized by clients which dynamically join and leave the wireless network.

3.2 Attack Implementation

At first sight, in order for the attacker to mount an ARP spoofing attack, he can simply choose to impersonate either the already associated stations or the AP itself by using their MAC addresses as sender's address. Although still effective, both of these approaches can be successfully detected. For every frame that the attacker sends (using either the address of an associated client or of the AP) the receiving station will send an acknowledgment. As a result, by receiving many acknowledgment frames, the legal station can identify that someone is using the same MAC address to send frames. Another problem that arises from impersonation of an already existing wireless client is that any frame received by the AP can be forwarded to a wired network in which traffic monitoring tools or intrusion detection systems can easily detect this kind of attack.

Therefore, to avoid being monitored and analyzed by more sophisticated systems, an attacker prefers to attack wireless clients only. Hence his goal is to keep fake ARP packets only within the wireless network. Furthermore, in order to avoid being detected by acknowledgements sent to existing clients, the attacker requires the possibility of using unknown MAC addresses as the source address for his attack.

In the following section we show that only by tweaking certain 802.11 frame characteristics an attacker can successfully send fake ARP packets with fully unknown MAC addresses, keep them undetectable by the AP and thus limit their propagation to wireless participants only.

Figure 4 shows a generic frame control field which is a part of every 802.11 frame.

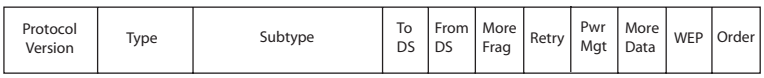

Fig. 4. 802.11 Frame Control Field

The two one-bit flags `ToDS` and `FromDS` are used to indicate whether the frame is sent from the distribution system to a wireless station or the other way around. In infrastructure mode, any frame sent from a wireless client will have `ToDS` bit set and `FromDS` bit cleared. Those frames are checked by AP to assure that a sender is an authenticated and associated station (as described in subsection 2.1).

On the other side, if the frame has the `FromDS` bit set, the AP believes that the frame was sent from a different AP (one distribution system can contain several APs). The AP does not know all the stations within the distribution system and cannot check if the sender's MAC belongs to a known and associated station. As a result, by setting `FromDS` bit an attacker can send arbitrary frames, even with an unknown MAC address without the frame being intercepted by the AP. These fake frames will therefore not be forwarded to the distribution system which renders protection mechanisms inside the networks.

The very last problem that the attacker has to overcome in order to successfully disguise his attack is the fact that although pretending to come from a wired network, the replies are received by an AP over the wireless medium. Although this could be used by AP to detect a fake frame, the mentioned situation is valid within IEEE 802.11 networks in case of a wireless bridge (valid ToDS and FromDS configurations and their meaning are shown in Table 2). Thus, the AP will neither see it as security vulnerability nor will react to it.

Table 2. ToDS and FromDS Flags and Their Meaning

	`ToDS = 0`	`ToDS = 1`
`FromDS = 0`	IBSS (ad hoc mode)	Frames from stations to DS
`FromDS = 1`	Frames exiting the DS	Frames from AP to AP (wireless bridge)

As a proof of concept we have created *smartspoof* tool which implements and executes the attack described above (the tool can be made available upon request). It

is an event-based tool that monitors the wireless medium for ARP requests and then immediately answers with a fake ARP inside a manipulated IEEE 802.11 frame. From our experiments in which we have used various wireless clients (both Linux-based and Windows-based) we can state that this attack was successfully executed on every tested client and after only a few minutes, all wireless clients had a poisoned ARP cache and the traffic was diverted to the attacker.

3.3 Discussion

Among the most common protections against ARP spoofing is a static ARP where the MAC-to-IP mapping can only be changed manually. Although very efficient within small infrastructures, this solution is not suitable for more dynamic environments. Especially in wireless environments where joining clients are new and initially do not know the network configuration, this solution cannot be implemented without introducing additional complexity. Furthermore, we have seen that different monitoring and traffic analyzing tools that are used inside the wired network to check if ARP replies provide valid MAC addresses are not effective. These mechanisms focus on networks in which traffic can be physically controlled. In contrast, a wireless environment with its broadcast nature makes neither of these solutions practical.

A more successful approach would be to monitor and analyze the wireless traffic. The difficulty in this approach lies in the operating mode of an access point. To be able to capture all traffic and still provide management and control functions, an AP must operate simultaneously in both, monitoring mode and master mode. However, this still imposes certain operational problems because in this case all the traffic should be analyzed by the AP itself. A more simple protection based on this approach would be to have additional access points or wireless stations for monitoring the traffic which consequently increases operational costs.

However, in contrast to a wireless enterprise network where all clients are known in advance and where the network is centrally administrated, the implementation of 802.11i within public, easily accessible wireless networks seem still to present a problem (although according to our measurements the performance tradeoff of introducing IEEE 802.11i does not represent a significant performance decline [11]). As a matter of fact, the usability-related problems of enforcing such security policies within public WLAN hotspots have already resulted in abandoning PKI-based solutions in favour of more light-weight proprietary solutions like web-based authentication which are the aim of the attacks as motivated at the beginning of this work.

4 Related Work

In 2003 the WLAN's security was a centre of various attacks against all security objectives. The unprotected management and control frames allowed fast and effective attacks on availability [3]. The poor security of WEP allowed attacks on confidentiality and integrity [5, 8, 4]. Various tools enabled simple flooding attacks, wireless client impersonation and injections of different frames directly on a wireless medium. In [3]

the authors showed how simple it is to mount different DoS attacks on IEEE 802.11 networks. There were several research activities coping with that problem and proposing cryptography based solution [6, 12]. Furthermore, in 2005 the IEEE 802.11 Task Group w (TGw) was established with the aim of creating a standard for authentication of management and control frames with an expected draft due in 2008.

The ratification of IEEE 802.11i standard helped to gain more trust into providing confidentiality, but due to still unprotected management frames, attacks on availability of IEEE 802.11i were fast to follow [9, 10].

In contrast to previous research, in this work we have introduced a novel attack based on performance decrease of a certain APs. This attack does not focuses on any of vulnerabilities based on IEEE 802.11 itself but shows that low- and middle-priced access points feature an operational anomaly that although intended to protect against DoS attacks can be abused to implement a new attack. This, in contrary to a reputation of WLAN as a low-cost technology shows that to provide a secure and reliable service more attention should be made on a choice of a hardware.

On the other hand, the second attack introduced in this work has its roots within a well known ARP cache poisoning attack [14] but it is used in a novel way within a wireless network. Probably the most similar work describing ARP poisoning attack within wireless networks is described in [7]. The author shows how a wireless network can be used to attack the wired infrastructure of an enterprise. This attack does not concentrate on the wireless network itself but uses it to attack the wired network.

5 Conclusion

In this work we have presented two different attacks within IEEE 802.11 wireless environments. Although both attacks have known ancestors, we have developed novel way to show that even a new generation of APs is prone to such kind of attacks. The first attack, based on extensive measurements of various APs abuses an operational anomaly of low- and middle-priced APs to hijack wireless clients and intercept their traffic. Although we cannot state that this attack can be applicable on every AP, our measurements let us assume that cheaper devices do introduce certain performance degradation which can also represent a security vulnerability. While the first attack can be avoided by using more expensive equipment, in our second attack we showed simple and yet effective client-based attack applicable in every scenario which sacrifices link-layer security. More importantly, we showed that one of the frequently used authentication methods within WISPs perfectly assists the attacker in hijacking the wireless clients.

6 Acknowledgement

We would like to thank Steffen Reithermann from Transkom Ltd. for many fruitful discussions and information from a WISPs perspective and for making various APs available to analyze.

References

1. IEEE 802.11. IEEE Standard for Local and Metropolitan Area Networks - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. IEEE Standard, July 1999.
2. IEEE 802.11i/D10.0. Security Enhancements, Amendment 6 to IEEE Standard for Information Technology. IEEE Standard, April 2004.
3. J. Bellardo and S. Savage. 802.11 Denial-of-Service attacks: real vulnerabilities and practical solutions. In *Proceedings of the USENIX Security Symposium*, pages 15–28, August 2003.
4. A. Bittau, M. Handley, and J. Lackey. The Final Nail in WEP's Coffin. In *SP '06: Proceedings of the 2006 IEEE Symposium on Security and Privacy (S&P'06)*, pages 386–400, Washington, DC, USA, 2006. IEEE Computer Society.
5. N. Borisov, I. Goldberg, and D. Wagner. Intercepting Mobile Communications: The Insecurity of 802.11. In *MobiCom '01: Proceedings of the 7th Annual International Conference on Mobile Computing and Networking*, pages 180–189, July 2001.
6. D. Faria and D. Cheriton. DoS and authentication in wireless public access networks. In *Proceedings of the 2004 ACM Workshop on Wireless Security*, pages 47–56, September 2002.
7. B. Fleck and Dimov J. Wireless Access Points and ARP Poisoning: Wireless vulnerabilities that expose the wired network. `www.packetnexus.com/docs/arppoison.pdf` (last access: 2006-10-30).
8. S. Fluhrer, I. Mantin, and A. Shamir. Weaknesses in the key scheduling algorithm of RC4. In *SAC '01: Revised Papers from the 8th Annual International Workshop on Selected Areas in Cryptography*, pages 1–24, August 2001.
9. C. He and J. C. Mitchell. Analysis of the 802.11i 4-way handshake. In *Proceedings of the 2004 ACM Workshop on Wireless Security*, pages 43–50, October 2004.
10. C. He and J. C. Mitchell. Security analysis and improvements for IEEE 802.11i. In *Proceedings of the 12th Annual Network and Distributed System Security Symposium (NDSS'05)*, pages 90–110, February 2005.
11. I. Martinovic, F. A. Zdarsky, A. Bachorek, and J. B. Schmitt. Introduction of IEEE 802.11i and Measuring its Security vs. Performance Tradeoff. In *Proceedings of the 13th European Wireless Conference, Paris, France*. accepted for publication, April 2007.
12. I. Martinovic, F. A. Zdarsky, and J. B. Schmitt. On the Way to IEEE 802.11 DoS Resilience. In *Proceedings of IFIP Networking 2006, Workshop on Security and Privacy in Mobile and Wireless Networking, Coimbra, Portugal*. Springer LNCS, May 2006.
13. B. Schneier. *Secrets & Lies: Digital Security in a Networked World*. John Wiley & Sons, Inc., New York, NY, USA, 2000.
14. S. Whalen. Introduction to ARP Spoofing. `http://www.node99.org/projects/arpspoof/arpspoof.pdf` (last access: 2006-10-24).