DoS Resilience of Wireless Access Points: An Empirical Study

Christian Jung

Projektarbeit

DoS Resilience
of
Wireless Access Points:

An Empirical Study

vorgelegt von

Christian Jung

08. Juni 2007

Technische Universität Kaiserslautern
Fachbereich Informatik
AG Verteilte Algorithmen

PA-005

Betreuer:	Dipl.-Inform. Ivan Martinovic
Prüfer:	Prof. Dr.-Ing. Jens B. Schmitt

Eidesstattliche Erklärung

Hiermit versichere ich, die vorliegende Projektarbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet zu haben. Alle wörtlich oder sinngemäß übernommenen Zitate sind als solche kenntlich gemacht worden. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Kaiserslautern, den 08. Juni 2007

Christian Jung

An dieser Stelle bedanke ich mich beim Distributed Computer Systems Lab, insbesondere bei Dipl.-Inform. Ivan Martinovic und Dipl.-Wirtsch.-Ing. Frank Zdarsky, die mir bei der Durchführung und Ausarbeitung meines Projektes mit Rat und Tat zur Seite standen. Des Weiteren möchte ich mich bei der Firma Transkom Kommunikationsnetzwerke GmbH bedanken, die es mir durch eine Leihgabe ermöglicht hat ein Gerät der Firma Cisco zu testen. Ein besonderer Dank gilt meinen Freunden und Bekannten, die keine Unanehmlichkeiten gescheut haben, um mir ihre eigenen Geräte zur Verfügung zu stellen. Außerdem danke ich meiner Schwester und meinen Mitstudenten, die mich durch ihre Kritik und Anregungen bei der Arbeit unterstützt haben.

Contents

List of Figures

List of Tables

1 Introduction

1.1 Motivation

The IEEE 802.11 networks have a tremendous growth in the last years, but also now there is a rapid development of the wireless LAN technologies. High transmission rates, simple deployment and especially low costs make this network technology an efficient and cheap way to get access to the Internet.
Fon[1] is the world-wide greatest WIFI community and in January 2007 this community offers more than 11.000[2] access points in Germany and nearly 55.000 all over the world.
However, this technology has also his shady sides. For example, it is possible for everyone to receive data from the wireless medium. So a protection against this open data traffic is a encryption mechanism called Wired Equivalent Privacy (WEP). The tragic end of the Wired Equivalent Privacy (WEP)[Tews 07] and the simplicity of various Denial-of-Service (DoS) attacks on the wireless medium have resulted in giving up the security at the logical-link layer and shifting it to upper layers (or in the best case leaving it within virtual private networks (VPNs)).
Nevertheless, there is an enormous growth in using public access to the Internet via HotSpots in cafés, libraries, schools or at airports, train stops etc. Therefore, it is important for the Wireless Internet Service Provider (WISP) to make sure that anyone with a usual wireless device can connect to their access points. Offering this service to anybody makes giving a sufficient level of security very difficult. On the one hand it should be easy for everyone to use this access, on the other hand there is, in most cases, no security. A businessman is not very pleased about phishing his account data for a great enterprise or for his online office like the KIS at the University of Technology in Kaiserslautern.
In most cases the WISPs use a simple web based authentication mechanism. By connecting to the WISPs services, the user is redirected to a webpage1.1 requesting his login data or credit card information. Therefore the user only needs a wireless LAN device and a webbrowser to authenticate.
An attacker could sniff on the wireless medium to phish delicate data from a legal connected user or use DoS attacks as initial point for various other attacks. In most cases, this can be done with no or only small effort. On the other side, in some cases,

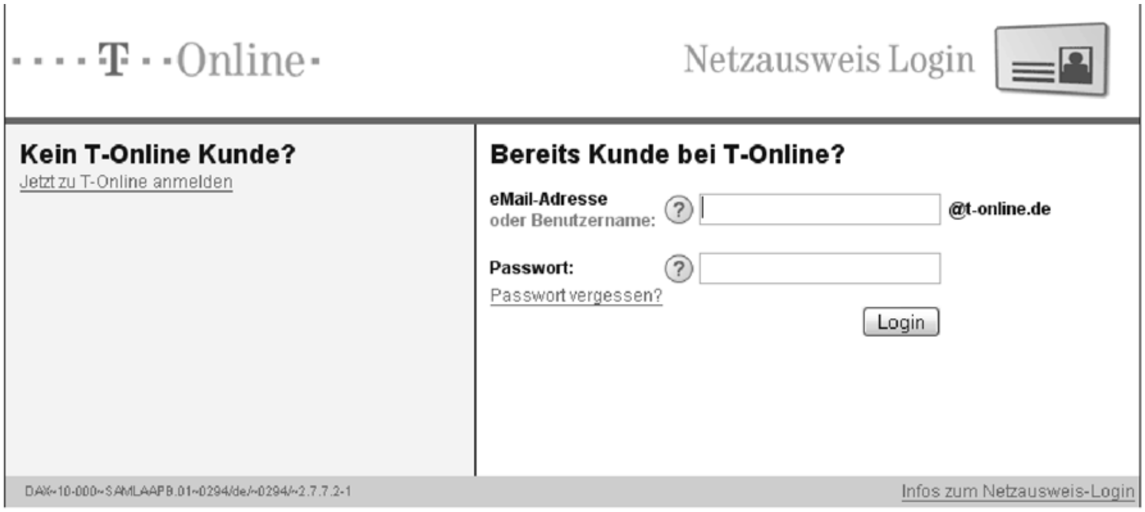
[1] http://www.fon.com

[2] http://www.dslweb.de/dsl-news/FON-zaehlt-11327-WLAN-Hotspots-in-Deutschland-News-2502.htm

the WISP has to do a hard reset on his wireless devices after a DoS attack.
Therefore an analysis of access points is done in this work. So, the first part is to show how "‘new"’ access points react to flooding attacks and what mechanisms are used to protect them. The second part implements an attack using an anomaly of some access points that are discovered in the first part. And the last chapter deals with some information about using an Intrusion Detection System (IDS) to protect the devices against such attacks.

1.2 Concept

This anomaly is shown by using different tools and measurements. In order to have a representative result there are ten different access points from 2002 to 2006 with price classes from 50 Euro to more than 400 Euro.
The main interest is to compare these access points, especially to show their reaction to these flooding attacks. Some figures and graphs help to underline and understand the results.



T-Online

Netzausweis Login

Kein T-Online Kunde?

Jetzt zu T-Online anmelden

Bereits Kunde bei T-Online?

eMail-Adresse
oder Benutzername: @t-online.de

Passwort:

Passwort vergessen?

DAX~10-000~SAMLAAPB.01~0294/de/~0294/~2.7.7.2-1

Infos zum Netzausweis-Login

Figure 1.1: web based authentication

2 Preparings

2.1 Hard and Software Overview

The Access Points

As seen in chapter one we will use seven different access points. This paragraph will give a quick overview on the devices. The price information comes from Geizhals.at[1]

manufacturer	**label**	**firmware**	**price**
Belkin	F5D7130	4.03.03 (10.03.2004)	ca. 45 Euro
Cisco	AIR-AP1231G-E-K9	TODO	ca. 370 Euro
D-Link	DI-624+	2.09 (01.03.2006)	ca. 60 Euro
D-Link	DWL900AP+	TODO	ca. 60 Euro
AVM	FRITZ!Box Fon WLAN 7050	14.04.2004	ca. 130 Euro
Linksys	WRT54G	Alchemy-V1.0 v3.37.6.8sv	ca. 60 Euro
Proxim	ORiNOCO AP-4000	v3.1.0(1052) SN-04UT13570825 v3.1.0	400 Euro
Siemens	T-Sinus 154 DSL	TODO	ca. 25 Euro
T-Com	Speedport W700V	1.06.000	ca. 110 Euro
Lancom	3550	unknown	ca. 400 Euro
Lancom	L54Dual	unknown	ca. 400 Euro

Table 2.1: Access Points

The Wireless LAN Cards

- Netgear MA401
- Proxim ORiNOCO 11a/b/g ComboCard 8480-WD
- D-Link DWL-G650, H/W Ver.: C2, F/W Ver.: 3.1.6
- Linksys WPC11, Ver.: 4

[1] Quelle:http://www.geizhals.at/deutschland/(Stand: 01.11.2006)

Software and Drivers

- hostap[2] - a Linux driver for Prism2/2.5/3
- hostapd[3] - a user space daemon for access point and authentication servers
- void11[4] - a tool with some basic 802.11b attacks
- progtest(libwlan) - a flooding tool that sends authentication and association requests
- file2air - a packet injector for wireless networks
- ethereal[5] - a network protocol analyser (wired and wireless)
- smartspoof - a tool developed at the distributed computer science lab at the University of Technlogy in Kaiserslautern

2.2 Wireless LAN IEEE 802.11

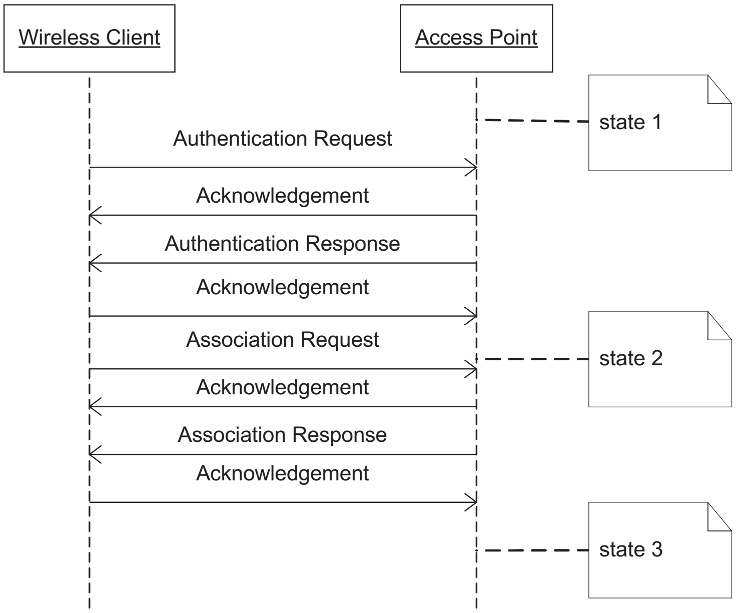


Figure 2.1: from state one to state three

It is necessary to give some general information about the 802.11 standard because of some expressions the author will use in his explanations.

[2] http://hostap.epitest.fi/

[3] http://hostap.epitest.fi/hostapd/

[4] www.wlsec.net/void11/

[5] http://www.ethereal.com/

The used mode of the access point is the infrastructure mode with visible Service Set Identifier (SSID). Therefore, the AP sends every 100ms a beacon frame to show his service disposition to any clients in range and acts as point coordinator to manage all connected stations. A new station has to authenticate and associate to the point coordinator before it can send any data to the network. There are three states a client can have. In the first state the client is not authenticated and not associated. After a successful authentication the client is in state two (authenticated, but not associated). And state three means the station is authenticated and associated. In this state, the client is allowed to send and receive data from the network. Mention, that all access points must allocate resources to save the state of a client. Figure 2.1 shows a sequence diagram from state one to state three.
This mechanism has two major aims. The access point can prevent clients from sending packets to the network when they are in a wrong state. In this case the point coordinator answers with a deauthentication or a deassociation request depending on the state of the station. The other major purpose keeps the access point from being replaced by a fake or called roque access point. If the legal AP detects another AP with the same Basic Service Set Identifier (BSSID) it sends deauthentication frames to all connected clients. So they are forced to reconnect.

3 Flooding

3.1 Access Point Behaviour

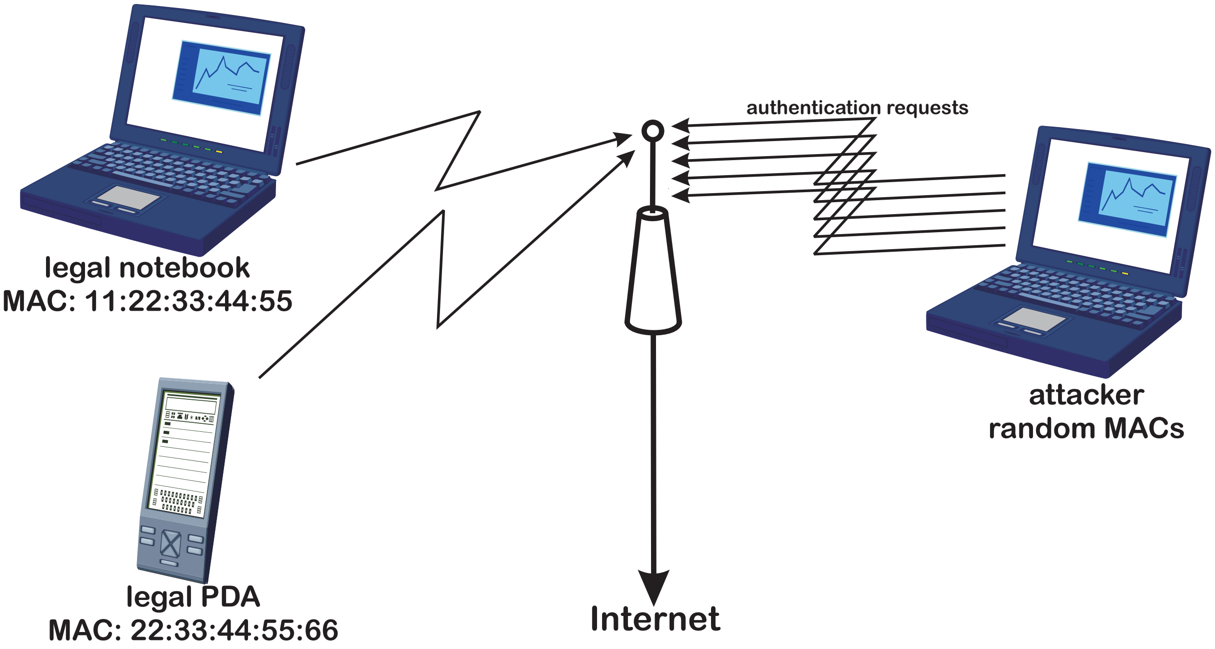


Figure 3.1: Denial of Service attack with flooding

This chapter describes the flooding with *void11* that has a chosen rate of approximated fifty authentication requests per second. The measurement includes sending faked class 3 frames (in this measurement Address Resolution Protocol (ARP) packets are used) from a client station that is not in the third state. Thus, the access point has to send a deauthentication frame to this station. Some access points have also been flooded with progtest(libwlan). This tool floods the AP with authentication and association requests. Using this program, in contrast to void11, points out other results to some access points. If the author does not refer to use progtest, void11 is the standard attacking tool.
The configuration is a legal access point, a notebook with void11/progtest to flood the access point, another notebook in monitor mode to observe the channel and a notebook to sent the faked ARP packets to the network. The faked ARP packets are sent every 500 ms. The flooding starts after a twenty seconds interval.

The following is analysed:

1. the time between the class 3 frame and the deauthentication frame

2. the time between the faked authentication request frame and the response frame from the AP

3. the loss of the ARP packets

4. the loss of the authentication requests

There is a drawing with two graphs for every access points. The left side shows the monitored authentication requests and responses. The values are discretised to draw the sum of collected frames every ten seconds. The right graph shows on the one hand the time between a faked ARP frame and the deauthentication frame, and on the other hand the time slot between a faked authentication request and the deauthentication frame.

3.1.1 Belkin F5D7130

Authentication Request vs. Authentication Response

At the beginning of the flooding attack the Belkin AP responds to about 360 authentication requests in the first thirty seconds. After this time it allows every ten seconds only one client to authenticate and after approximatly two minutes it serves near to twenty stations again. In a second measurement the significant data has been repeated. So the access points protection against the flooding attack is to allow only a certain number of stations in a time slot. It is to assume that this number probably depends on the table size for saving the states. This seems to be a very easy and successful protection system against the flooding attack and the clients who are not allowed receive an authentication response with a failure message.
Keep in mind that a legal client is only accepted in certain time slots. So the drawback is that about 90 to 95% of the authentication requests are rejected. Comparing the sum of monitored authentication requests and responses (successful and unsuccessful responses) points out a packet loss smaller than twenty percent.

DeAuthentication Delay

As shown in the figure 3.2 (right) the Belkin AP has a delay in responding to the faked ARP packets and to the authentications. So the access point is overloaded by responding to the faked authentication requests. The deauthentication delay is about 12 seconds.
A second notice is a significant miss or not deauthenticating of ARP packets. There

is a loss of almost seventy percent. On the other side, 35 to 40 percent of the authentication requests are not deauthenticated. It stands to reason that the authentication responses with failure notice are not deauthenticated.

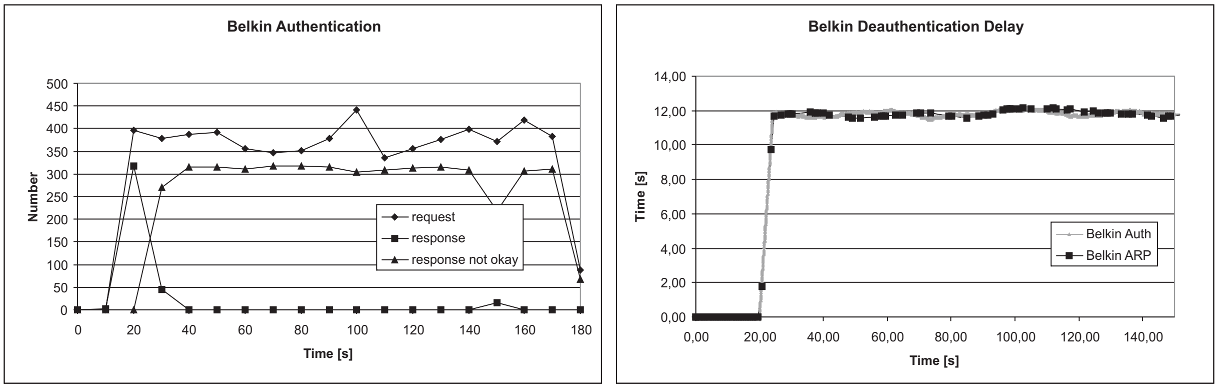


Figure 3.2: Belkin

3.1.2 Cisco AIR-AP1231G-E-K9

Authentication Request vs. Authentication Response

The figure 3.3 (left) shows a good ratio between the authentication request and the authentication response. The AP is able to response to all monitored requests by the attacker client. There are authentication response peaks between 40 and 60 seconds and between 120 and 140 seconds. In measurements with longer duration the peaks are repeating approximately every 80 seconds. Analysing this inconsistency with a network protocol analyser shows that the Cisco AP sends three authentication responses to some clients without setting the retry flag. Nevertheless, the access point has nearly no wastage of any authentication requests.

The loss of autentication requests is about zero percent. This shows a very good performance against the flooding attack and it seems that the resources for saving the client states are endless. Every legal client is able to connect to the Cisco AP.

DeAuthentication Delay

There is no deauthentication delay as seen in figure 3.3. A minimal increase of time can only be realized regarding the deauthentication frames to the authentication requests. However, the delay remains smaller than ten milliseconds excepting some peaks.

As for the packet loss it shows also a very good performance. In all measurements, the loss of the faked ARP packets and also of the attackers authentication responses is smaller than two percent. Recapitulating, the deauthentication frames are not affected by the flooding attack.

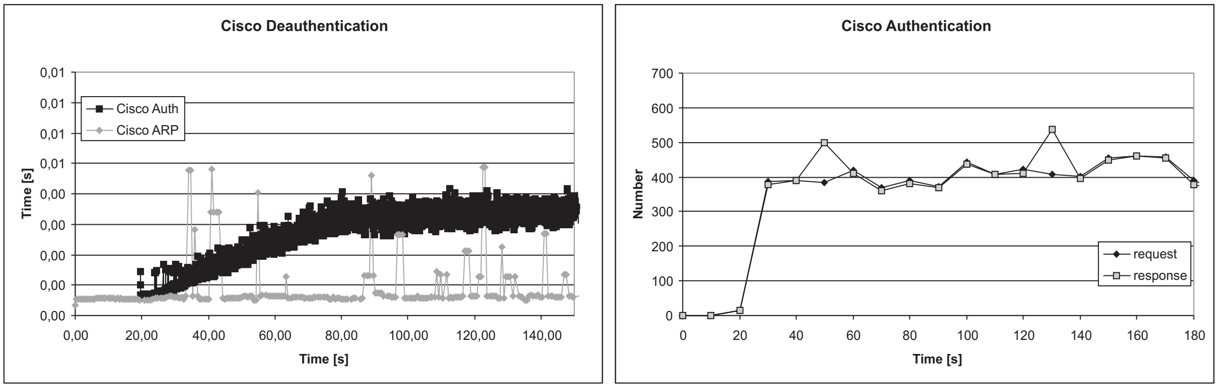


Figure 3.3: Cisco

3.1.3 D-Link DI-624+

Authentication Request vs. Authentication Response

The first D-Link AP, the DI-624+, accepts 64 clients in the beginning of the flooding attack with void11. The table for saving the client states seems to have space for 64 stations. In relation to flooding, this behaviour is similar to the Belkin AP, but the DI-624+ does not accept any more stations after a certain time. On the one hand, the protection is simple but good, on the other hand the loss rises to 100 percent.
Other measurements with progtest as flooding tool highlight a crash alternatively a freeze of the attacked access point. In this case it responds only to the first packet and does not give any service to any clients anymore. However, it sends beacon frames every 100ms. The loss is also assumed to be 100 percent. A legal client can only connect to the network in the beginning of the flooding attack.

DeAuthentication Delay

As seen in figure 3.4 (right) there is only an inconsiderable increase at the beginning of the flooding not exceeding 20 milliseconds. Starting the flooding attack, the access point accepts a certain number of clients. Thus, the authentication managing impacts the reaction time. This increase is not deciding.
Considering the loss it is necessary to differentiate between the faked ARP packets and the attackers authentication frames. The faked ARP packets have a loss below three percent, but the attackers authentications are only deauthenticated in the first seconds of the attack. After this, there is no deauthentication reply to the attacker. This loss is therefore near to 100 percent if the time increases to infinite. It is to mention, that there is a time slot at the beginning of the attack, in which no ARP packet is deauthenticated (see Figure 3.4 (left) from about 7 seconds).

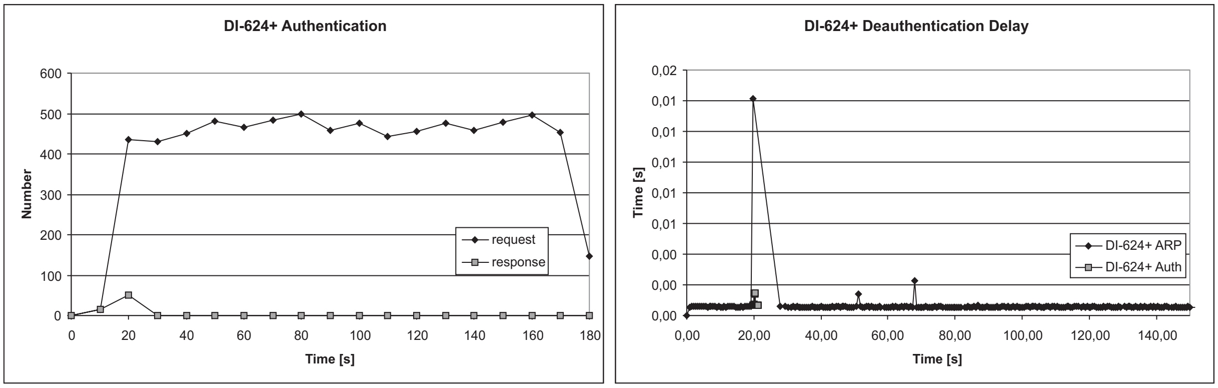


Figure 3.4: DI-624+

3.1.4 D-Link DWL900AP

Authentication Request vs. Authentication Response

Figure 3.5 (left) is in the first 40 seconds similar to the Cisco AP. In this time frame it accepts near to all stations, only a few requests are missed. But after this time, the AP is also freezed like the other D-Link AP. It still sends beacon frames every 100ms. Other measurements show a variance of time, in which the AP is able to give any service to the clients. It starts at about 40 seconds and the best case is near to 60 seconds. Apart from the time in which the flooding has not been started the AP survives / endures about 20 to 40 seconds flooding. In this time a legal client is able to connect to the access point.

The loss rises towards 100 percent. Looking at the authentication request and the authentication response mechanism with a network protocol analyser, the AP sends always three authentication responses without setting the retry flag like the Cisco AP does every 80 seconds.

DeAuthentication Delay

As seen in 3.5 (right) there is an increase in the reaction time at the flooding start, so that the deauthentication delay mounts several times near to 50ms. Different from DI-624+ the DWL900AP does neither response to the faked ARP packets nor to the authentication requests after it crashed by flooding. Serving any legal stations is not possible after the crash, so the robustness is not good.

As for the operation time of the AP, the loss is near to 100 percent.

3.1.5 AVM FRITZ!Box Fon WLAN 7050

Authentication Request vs. Authentication Response

The FRITZ!Box is able to serve 64 stations in every measurement. This means the

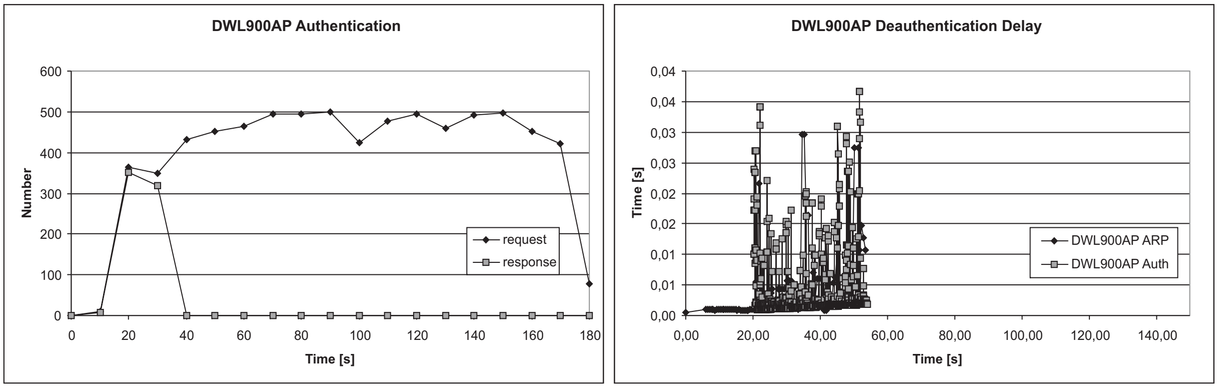


Figure 3.5: DWL900AP

AP has resources to save the state of 64 clients. After accepting this number of clients, it does not allow anymore stations (similar to the DI-624+ AP). The same results can be found on flooding with progtest, in contrast to the DI-624+ that is freezed by flooding with progtest. Like the above-mentioned D-Link access points, a legal client can only connect in the first seconds of the attack. The protection system against the authentication flooding allows only the first 64 stations.
Therefore, the loss rises also to 100 percent.

DeAuthentication Delay

Comparing figure 3.6(right) with the DI-624+ figure 3.4 (right), they are looking very similar. So the FRITZ!Box has also an increase in the deauthentication delay at the beginning of the flooding attack. Similar to the DI-624+, it is required to differentiate between the faked ARP packets and the attackers authentication frames.
On the one hand, the FRITZ!Box has a loss of authentication requests nearly to 100 percent, because of the above-mentioned blocking mechanism. On the other hand, there is only a minimal loss of faked ARP frames. It does only miss to deauthenitcate less than three percent. Using progtest, the loss is slightly higher but still under ten percent.

3.1.6 Linksys WRT54G

Authentication Request vs. Authentication Response

This AP behaviour is comparable with the behaviour of the Belkin AP. As seen in figure 3.7 it also replies to authentication requests not being able to be served with a failure notice in its authentication response. In the first 40 seconds, that means 20 seconds flooding, the AP responses to about 360 authentication requests and after approximately two minutes it allows roughly another 25 stations to connect. This behaviour is similar to that of the Belkin AP.

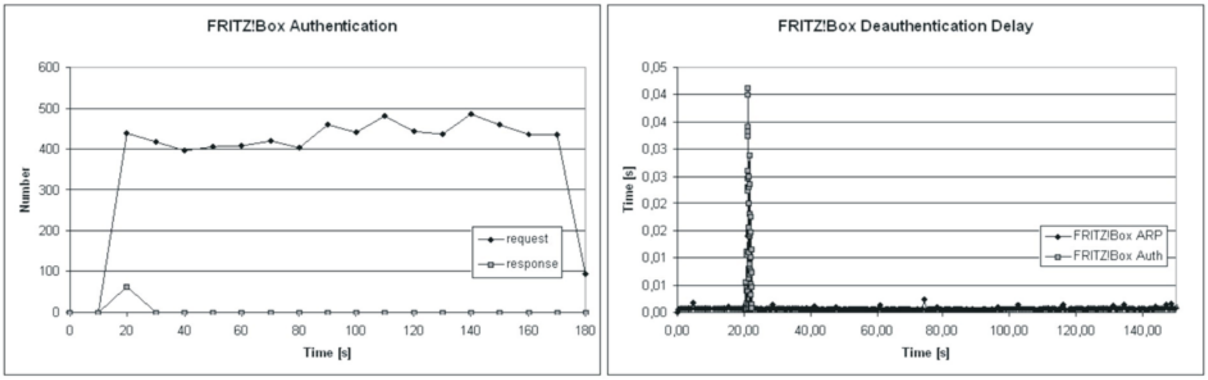


Figure 3.6: FRITZ!Box

Looking at the monitored network traffic, the Linksys AP responses to about 75 percent authentication requests. A legal client has a chance of about five percent to connect to the network.

DeAuthentication Delay

Recalling the behaviour of the Belkin AP, the Linksys AP has the same authentication delay in both testings. That means a faked ARP packet and also an attacker authentication will be delayed deauthenticated. This delay is about 12 seconds after the access point is flooded (similar to the Belkin AP).
The loss of faked ARP packets and attackers authentications is different. The deauthentications to the ARP packets are missing about 60 to 65 percent and the false authentication frames have a loss of about 40 percent. Comparing this data with the Belkin F5D7130, the measurements of the Linksys AP discover the same results.

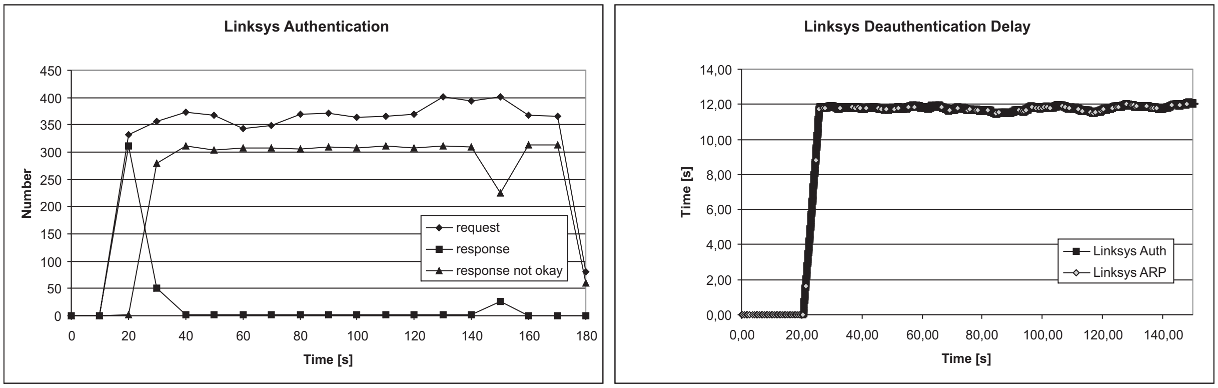


Figure 3.7: Linksys

3.1.7 Proxim ORiNOCO AP-4000

Authentication Request vs. Authentication Response

The graph 3.8 (left) shows that the Proxim AP acts like some other tested access points. At the beginning of the attack, it allows 60 clients and after every minute the AP accepts again the same number of clients. Within this interval of 60 seconds, there are no clients allowed to connect to the access point. In some measurments this access point allows a few station in this interval. It is to assume, that the AP clears his table with the saved states every 60 seconds. Therefore it is possible to refill the client state table with 60 new authentication requests.

As seen before, this is a moderate protection against the flooding attack, although the AP is only able to serve about two percent of the authentication requests. A legal client has a minimal chance to be accepted.

DeAuthentication Delay

This access point has nearly no deauthentication delay. Accepting some new stations increases the deauthentication delay in worst case about ten milliseconds. This increase can be disregarded, so that this access point is assumed to have no deauthentication delay.

Comparing the loss of faked ARP packets and attackers authentications, the ORiNOCO AP-4000 replies only to the false authentication requests when allowing new clients (see Figure 3.8 (left)). Therefore the loss of authentication frames is very high in contrast to the loss of faked ARP frames. The access point misses about 97 percent of the attackers requests to authenticate, but nearly all faked ARPs are deauthenticated. The loss is smaller than one percent.

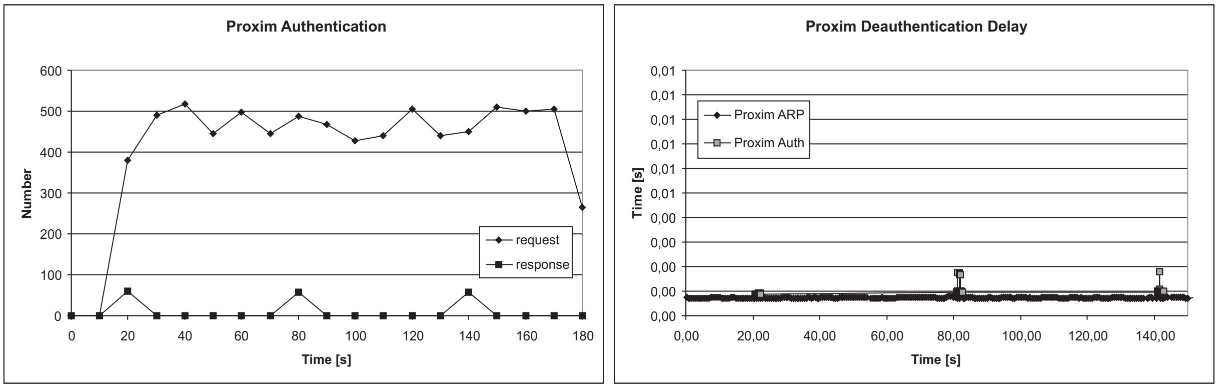


Figure 3.8: Proxim

3.1.8 Siemens T-SINUS 154 DSL

Authentication Request vs. Authentication Response

At the beginning of the attack, the T-Sinus AP allows only about 30 stations and after this period, one station is accepted every ten seconds. Similar to the Proxim AP, the T-Sinus is able to serve about 50 new clients every 60 seconds. As mentioned before, this behaviour depends on clearing the table for saving the client states.
Regarding the loss of authentication requests, the AP is able to accept about two percent of the clients. So it is difficult for a legal station to connect to the network. Therefore, the protection seems to be a reasonably good solution with the disadvantage of a high loss.

DeAuthentication Delay

The figure 3.9 (right) shows a slight increase in time of the deauthentication responses, but always under 30 milliseconds. Comparing this behaviour with the Proxim AP, a similarity can be found. The T-Sinus AP has a delay when it allows some new stations to authenticate to the network. The time of the delay can be disregarded. Therefore, the deauthentication delay is not a crucial factor.
Starting the attack, the access point has nearly no loss of faked ARP packets and attackers authentication requests. But after a certain time, it only replies to theses packets every 60 seconds. That signifies a freezing of the AP in the intermediate time slots. The T-Sinus 154 DSL misses about 90 percent of the sent faked ARP packets and about 98 percent of the attackers authentication requests.

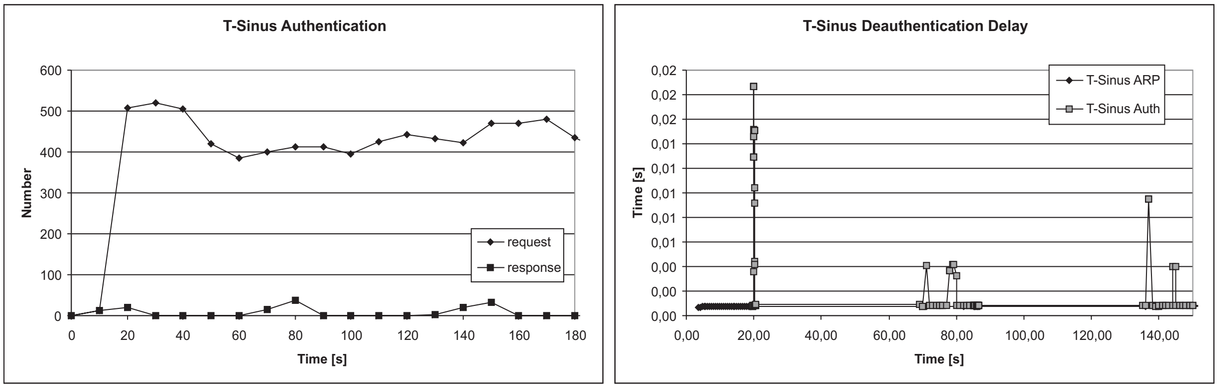


Figure 3.9: T-Sinus

3.1.9 T-Com Speedport W700V

Authentication Request vs. Authentication Response

Starting the flooding attack, the T-Com access point allows about 60 stations to au-

thenticate. As seen before, this seems to be a moderate mechanism to protect this access point against the flooding attack (compare the FRITZ!Box or DI-624+ behaviour). It is to mention, that this access point allows only at the beginning of this flooding attack a certain number of stations to authenticate. A legal client has nearly no chance to connect to the network after the flooding attack has started.
The loss of authentication requests rises to 100 percent, because of the behaviour explained above. As mentioned before, this behaviour is a simple way to protect the access point against the authentication flooding, but with a drawback in reachability of the network for new legal clients.

DeAuthentication Delay

The graph 3.10 shows a little delay that can be disregarded. Starting the flooding attack does not change this delay. The delay for the authentication responses is a little bit higher than the time to response to the faked ARP packets. There are only replies for the first authentication requests. So the access points sends only packets to the stations that are allowed to authenticate. All other authentication requests remain without an answer or notification.
Taking a look at the loss for the authentication requests and the faked ARP packets shows a different behaviour. On the one hand there is a miss about twenty percent of the manipulated ARP packets, on the other hand there are nearly all authentication requests missed or as the case may be they voluntarily receive no consideration by the access point.

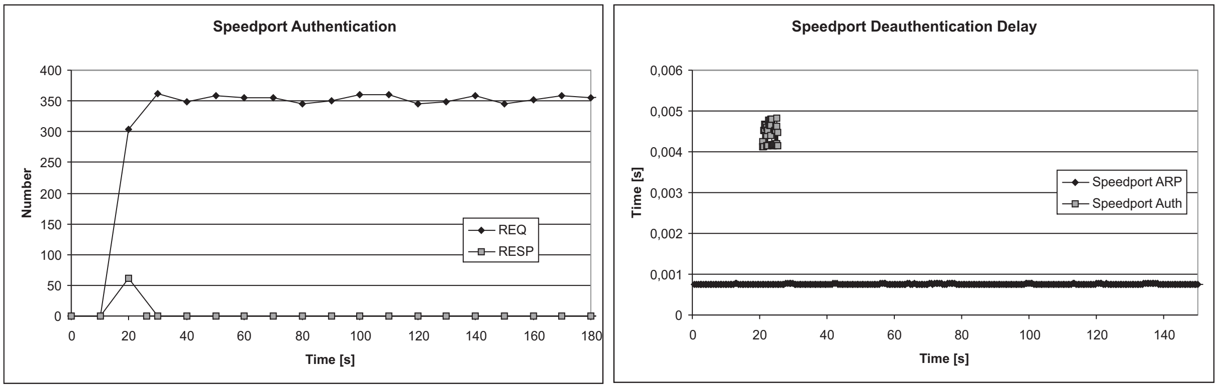


Figure 3.10: Speedport W700V

3.1.10 Lancom 3550

Authentication Request vs. Authentication Response

This access point is able to response to nearly all authentication requests. This device is able to serve nearly all legal stations, that want to connect to the network. A

closer look on the results of the measurments shows, that this access point sends two authentication responses to the attacker without setting the retry flag. The number of responses are divided by two for the figure 3.11. Another notice points out, that the Lancom 3550 is not able to respond to all authentication requests every two minutes. This behaviour can only be declared as an internal working of this device which repeats every two minutes.
There is nearly no loss of authentication requests excepting the behaviour every two minutes. This access point is similar to the Cisco device. The loss of authentication requests is smaller than ten percent.

DeAuthentication Delay

Taking a look at the graph 3.11 points out, that there is no delay. So the faked ARP packets are nearly directly deauthenticated. Starting the flooding attack does not increase the time slot between the ARP packet and the deauthentication. There is also no visible increase every two minutes as perhaps assumed because of the described behaviour above. If the access point is stressed every two minutes by some internal work, this effort does not stress the response time but the device is not able to process all authentication requests. It is to suggest that this device misses some authentications to give a better service to the accepted requests.
The loss for the faked ARP packets is higher than the loss of the authentication requests. The access points misses about eighteen percent of the ARP packets and ten percent of the authentication requests.

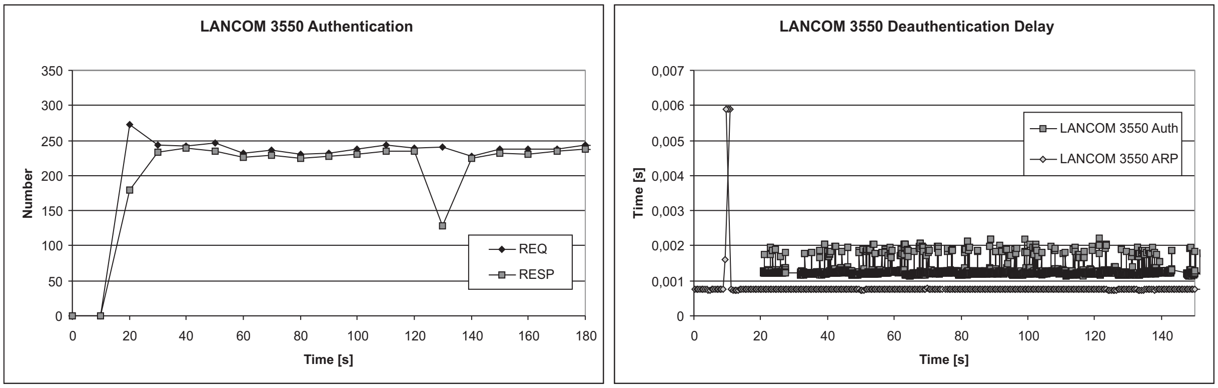


Figure 3.11: LANCOM 3550

3.1.11 Lancom L54Dual

Authentication Request vs. Authentication Response

Comparing this access point with the other Lancom device shows a similar behaviour. The Lancom L54Dual is also able to serve nearly all authenticating stations.

Regarding the authentication responses with a network protocol analyser shows the same behaviour as the Lancom 3550. It is only neccessary to take half of the measured responses, because of sending two replies without setting the retry flag. As seen above, this device is also not able to handle all authentication requests after two minutes. But this anomaly does not repeat in contrast to the other Lancom device.
The loss is smaller than four percent and is therefore a little bit better than the other Lancom access point.

DeAuthentication Delay

The access point has no deauthentication delay, so it is not stressed by the authentication flooding. After two minutes flooding, the access point does not response to all authentication requests, but the deauthentications for the ARP packets are not affected.
The loss of the faked ARP packets is smaller than four percent and is therefore also better than the other Lancom.

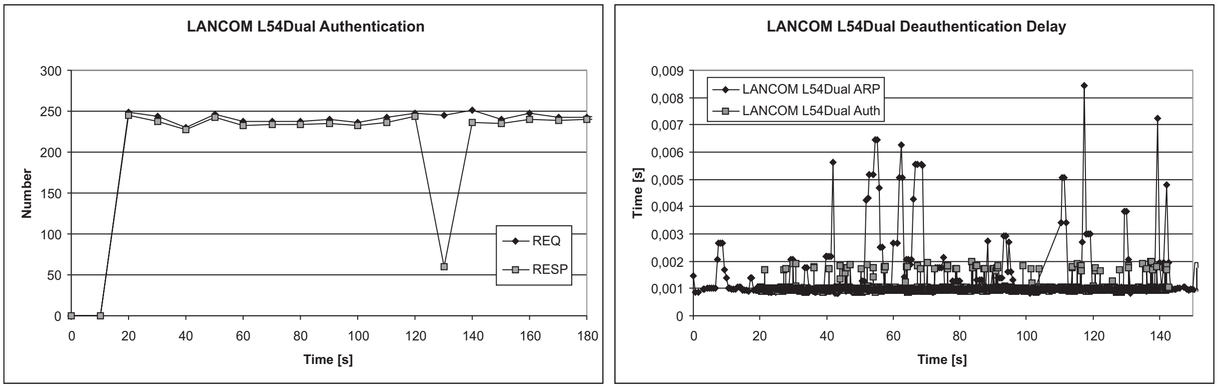


Figure 3.12: LANCOM L54Dual

3.1.12 Summary

Summing up the measured results, it is possible to divide the access points in different classes depending on their behaviour. The Belkin and the Linksys access points are the only measured access points replying with an authentication response that includes a failure notice as soon as no more stations are accepted. However, they have a deauthentication delay about twelve seconds depending on the load of the device. It suggests itself that the delay depends on the additional effort to send the failure notice to the authenticating clients.
The flooding attack freezes both D-Link access points, so that they are not able to give any service to any legal stations - taking into consideration that the DI-624+ crashes only with progtest. Without using this tool, this AP can be compared with

the FRITZ!Box and the Speedport.

The three devices allow a certain number of clients at the beginning of the flooding attack, depending on their free resources to save the state of the authenticating stations. The Proxim ORiNOCO AP-4000 is similar to these access points, but it allows authentication requests from new stations every 60 seconds. Also the T-Sinus DSL 154 accepts new stations every 60 seconds, but during this interval it is not able to serve any operations to new stations, except one successful authentication every ten seconds. Therefore, the device can be seen as temporarily freezed. The three last named devices have no deauthentication delay.

The Cisco AIR-AP1231G-E-K9 seems to have unlimited resources to save the client states. There is also no deauthentication delay. Both Lancom access points have a similar behaviour, but a higher loss than the Cisco.

The research shows different approaches to protect the access point against the flooding attack. The first mechanism is no real protection, but has enough resources to serve all clients (see Cisco and Lancom AP). Another possibility is to allow a certain number of clients at the beginning of the attack. After this time, no more clients are allowed to connect (FRITZ!Box, Speedport and DI-624+). A related behaviour is to allow some clients every few seconds or minutes. The Belkin, Linksys, Proxim and T-Sinus APs show this behaviour. Only the DWL900AP crashes by using an authentication flooding tool. Another important fact to remember is that only the Belkin and the Linksys APs are able to send an authentication response with a failure notice. If a WISP had to choose between these ten APs, it would probably prefer the Cisco AIR-AP1231G-E-K9 or one of the Lancom access points. All other devices are not able to handle all authentication requests or are overloaded by the flooding attack.

The table 3.1 gives an overview on the tested devices.

access point	delay	auth succ	failure notice	Loss ARP	Loss Auth	especialness
Belkin F5D7130	12s	some, every 120s	yes	<70%	35 - 40%	sends failure notice
Cisco AIR-AP1231G-E-K9	no delay	nearly all	no	<2%	<2%	replies at some time with three authentication responses
D-Link DI-624+	no delay	some at startup	no	<3%	nearly 100%	crashes with progtest after the first packet
D-Link DWL900AP+	no delay	all, till crash	no	crashed	crashed	very good behaviour until crashed
AVM FRITZ!Box Fon WLAN 7050	no delay	some at startup	no	< 10%	nearly 100%	-
Linksys WRT54G	12s	some, every 120s	yes	<65%	35 - 45%	sends failure notice
Proxim ORiNOCO AP-4000	no delay	some, every 60s	no	<2%	nearly 100%	-
Siemens T-Sinus 154 DSL	no delay	some, every 60s	no	<90%	nearly 100 %	seems to be freezed for 60s
T-Com Speedport W700V	no delay	some at startup	no	<20%	nearly 100%	-
Lancom 3550	no delay	nearly all	no	<20%	<10%	seems to do internal work every 60s, replies every time with two authentication responses
Lancom L54Dual	no delay	nearly all	no	<4%	<4%	replies every time with two authentication responses

Table 3.1: Summary Flooding

The below classification distinguishes between these access points:

- class 1 - accepting nearly all stations, no deauthentication delay, no crash
  Cisco AIR-AP1231G-E-K9, Lancom 3550, Lancom L54Dual
- class 2 - accepting some but not all stations, no deauthentication delay, no crash
  AVM FRITZ!Box Fon WLAN 7050, Proxim ORiNOCO AP-4000, D-Link DI-624+, T-Com Speedport W700V
- class 3 - accepting some but not all stations, deauthentication delay, no crash
  Linksys WRT54G, Belkin F5D7130
- class 4 - accepting some but not all stations, no deauthentication delay, temporarily freezed
  Siemens T-Sinus 154 DSL
- class 5 - freezed or crashed
  D-Link DWL900AP, D-Link DI-624+

The following measurements analyse the data throughput with a given flooding rate. It is not necessary to analyse all seven access points, only the disrupted access points are considered. Especially the two access points with the deauthentication delay are interesting, because of the second part of this research. This deauthentication delay is used to implement a novel attack on the affected access points.

3.2 Throughput vs. Flooding Rate

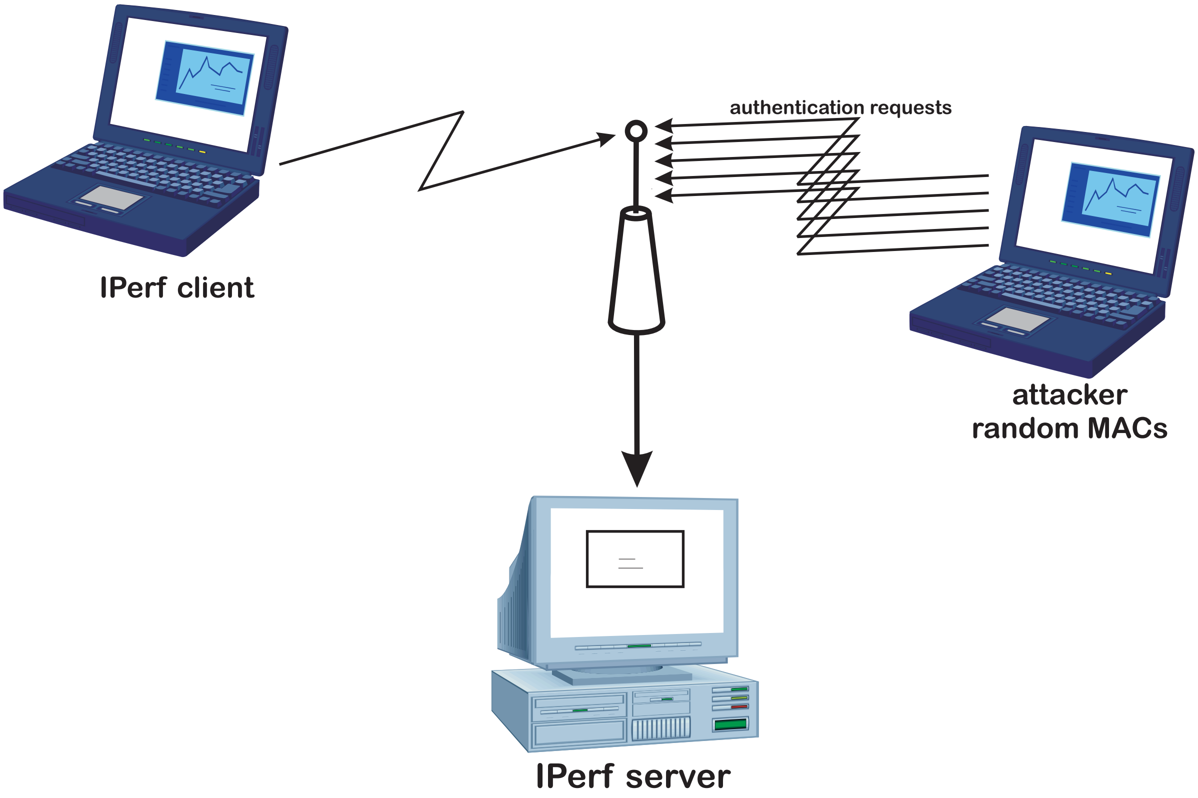


Figure 3.13: Throughput vs. Flooding Rate

This section deals with some measurements regarding the data throughput of a legal client that is connected to the access point. The AP serves one client over the wireless network and a wired connection is established to a computer. This workstation provides an IPerf server to allow connections from the notebook and saves the current output of the datarate every 500 milliseconds. Another notebook operates as attacker and floods the wireless network with authentication requests. The flooding starts after 20 seconds measurement. The construction can be seen in figure 3.13. The results are shown in two figures. The left graph shows the mean datarate and the theoretical number of authentications per second. The real number of requests is smaller than in the void11 settings. The right graph is used to show the access points behaviour by flooding with a delay of 150ms, 100ms and 50ms.

3.2.1 Belkin F5D7130

The left figure 3.14 indicates a decrease of the mean datarate by increasing the flooding rate. Disturbing the access point produces this behaviour. A breakdown is seen from a flooding delay smaller than 80 milliseconds, so the client is not able to deliver

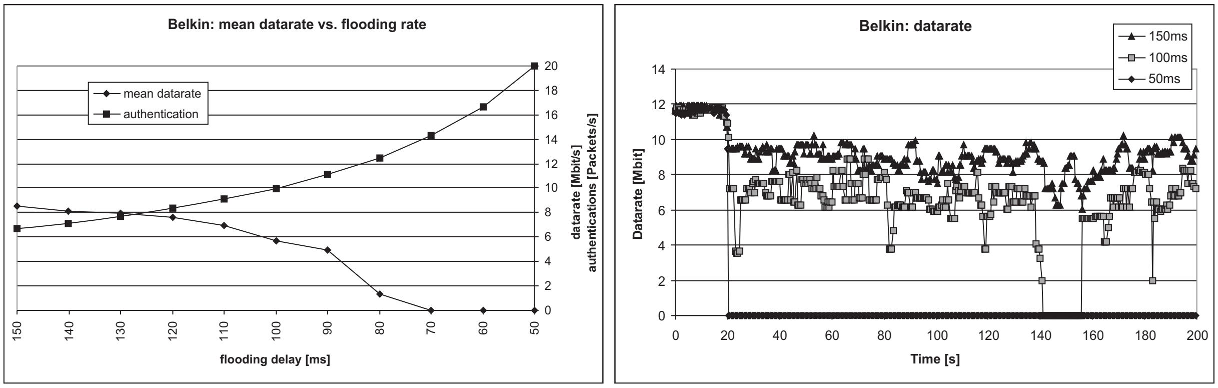


Figure 3.14: Belkin: IPerf datarate

any data to the IPerf server. The figure 3.14 (right) demonstrates a constant datarate in the first 20 seconds (no flooding). After this time, the Belkin AP has an alternating datarate for a flooding delay exceeding 100ms. The client is still able to communicate with the server. Regarding the curve for 50ms, the communication is interrupted briefly after starting the flooding attack. Although the access point does not crash or freeze by the flooding attack, the legal and connected clients are not able to interact with the network.

3.2.2 Linksys WRT54G

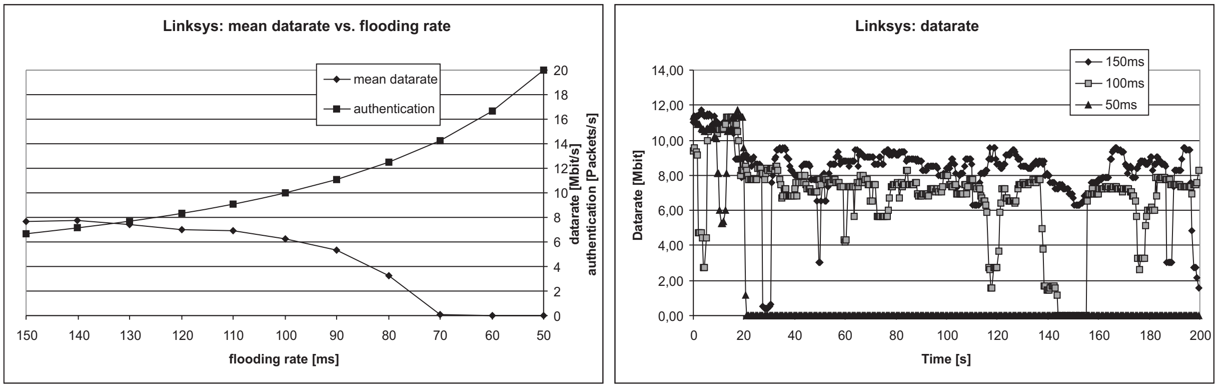


Figure 3.15: Linksys: IPerf datarate

As expected, the Linksys AP has the same behaviour as the Belkin. There is also a breakdown of the device when the flooding delay is smaller than 80 milliseconds, but the Linksys is able to survive this attack longer. Table 3.2 lists the mean datarate and the flooding delay. This shows the better robustness of the Linksys WRT54G.

3.2.3 Other

The five other access points are able to communicate with the legal connected access points. The T-Sinus 154 DSL is also able to interact with the legal connected notebook, although the AP reacts to no faked ARP packets and to no attackers authentications (only every 60 seconds, when the device accepts a certain number of authentication requests).

3.2.4 Summary

Both tested access points are not able to deliver data to the wired network. As seen in table 3.2 the mean datarate decreases by increasing the flooding rate. A flooding delay of 80ms or smaller is enough to stress both access points in a way that they are not able to serve the legal connected clients. Summing up, a legal connected client has hardly no bearing on flooding these other devices. In a real scenario, the attacked access point is additionally stressed by other legal clients and especially from their data transfers.

flooding rate (interarrival rate) [ms]	150	140	130	120	110	100	90	80	70	60	50
Belkin datarate [Mbit/s]	8,54	8,13	7,92	7,59	6,94	5,66	4,95	1,35	0,04	0,03	0,03
Belkin std. dev. [Mbit/s]	2,13	2,37	2,42	2,65	3,11	3,67	3,75	4,10	3,78	3,72	3,72
Linksys datarate [Mbit/s]	7,70	7,76	7,39	7,04	6,92	6,21	5,35	3,27	0,04	0,03	0,04
Linksys std. dev. [Mbit/s]	2,71	2,39	2,48	2,74	2,53	2,97	3,34	3,85	3,03	2,95	3,19

Table 3.2: Belkin: IPerf mean datarate

3.3 Discussion

Taking a look at the summaries, a potential demand on other techniques to protect access points against a conventional and well known flooding attack with authentication requests is to be noted. Most devices are able to give a pretty good service to the connected clients, although new clients, connecting to the network, are rejected or simply ignored by the access point. This is perhaps an applicable mechanism to protect a wireless network for home users, but a wireless internet service provider would not be pleased about his devices being unreachable for their customers. Therefore, another research especially in this area is needed.

Regarding some new attempts and related work at the University of Technology in Kaiserslautern, the author will give a short overview about two methods, based on the idea, that a legal station has to work for its connection to the network.

The first method is based on known results from the cryptography. The access points send a puzzle to be solved by the client. This puzzle can be placed into the beacon frames or it is sent to the clients after getting a probe request packet. After this, the authentication request must contain the answer to the problem. In case of failure, the access point does not reserve any resources to save the client state. The factorisation into primes is a possible puzzle for a station. For the access point it is easy to riddle. But on the client side it is an intensive effort to factorise, because of non existing efficient algorithms (not yet). This method offers two main advantages. Firstly, a flooding tool has to wait for the result of the given puzzle. Assuming that it takes about 100ms to solve the problem, the flooding rate can only be about ten packets per second. This low rate cannot be seen as flooding. And secondly, there is no wasting of system resources to save the attackers client states.

The second method uses the ability to locate or get the distance to other stations. Legal connected stations have to operate together with the AP. That means a new station must listen to the channel, in order to collect the next located stations. This information is sent to the access point at the beginning of the interaction. This is confirmed to be done by asking the named clients if they are able to hear the new station. This has two major aims. On the one side, the AP is able to check the information and on the other side, it is able to locate the station, at least an approximate position. Analysing this information, the point coordinator is able to detect a flooding attack, because it is impossible to send hundreds of authentication requests from nearly the same position.

There is also a great disadvantage looking at the mechanisms implementation. It is necessary to modifiy the drivers for the stations and the access points in order to be able to handle the new requirements.

4 A Novel Attack

This chapter analyses a novel attack[Martinovic 07b] in which a rogue access point is set up to replace the legal one. In combination with the flooding attack, the fake AP is able to steal new stations that want to authenticate. Previous attacks used the ability that a client tries to connect to the point coordinator with the strongest signal strength. Therefore the attacker has to use a device with more signal power than the legal access point to steal legal connecting clients. Another aspect is the different mac address of the rogue access point. So the attacker's mac address is not the same as the mac address of the legal device.
The attack in this work uses the same mac address and tries to stress the WISPs hotspot, so that it is not possible to react to the faked service. In the best case, it is possible to crash or freeze the legal device by a flooding attack.
The following is analysed:

1. quality of the rogue access points connection
2. connection times
3. a faked web server with SSL

4.1 Connection Quality

This section describes some measurements to analyse the quality of the faked connection. If the legal access point observes another access point using the same BSSID (Basic Service Set IDentifier), it has to send deauthentication requests to all connected clients. It is interesting to analyse the connection qualitiy between a legal client and the roque access point when the legal access point is flooded.
The configuration to analyse the qualitiy is the following. An attacker station is set up with two wireless lan interfaces. One interface uses void11 to flood the legal access point with authentication requests. The other interface sets up a rogue access point with the madwifi wlanconfig tool. The false access points starts an IPerf server, so clients are allowed to connect. The legal access point is in infrastructure mode and has one special setting, that is described in the following section. A client wants to connect after a certain (((ramdomly chosen)) time to the network and after this it starts an IPerf client. The program IPerf is used to analyse the throughput from the legal client to the faked access point. It is to differentiate between TCP and UDP traffic. The configuration uses TCP traffic with 11Mbit/s, so that the congestion control[RFC2581 99]

(as described in [Kurose 05] 254-264) has to manage the datarate. The UDP troughput measured with 1Mbit/s, 2Mbit/s and 5MBit/s. The legal access point and the rogue one are both running without resetting them for more than one measurement. This is done to have a realistic measurement like it would also be in real life. That means both access points are running and a legal client tries to connect to the network without having a definite arrival time.

The first measurements show, that the rogue access point is able to communicate with the legal client over a long time period without monitoring any deauthentication frame. Taking a closer look at the results points out, that the legal client is able to authenticate and associate with the legal and the rogue access point. In this situation, the legal access point sends no or only sporadic deauthentication frames. Therefore the attacker is able to hold a stable channel to the legal client over a long time. It is necessary to activate the access points mac filter, to prevent the client to connect to the legal device. Setting this option helps to force the legal access point to send deauthentication requests to this client.

In the following, the two devices with the deauthentication delay are analysed and the results are described. The T-Sinus is also mentioned, because of a special behaviour when a client tries connecting to the network for the first time. That means this client has not authenticated or associated to this access point since the last reset respectively restart of the device.

4.1.1 Belkin F5D7130

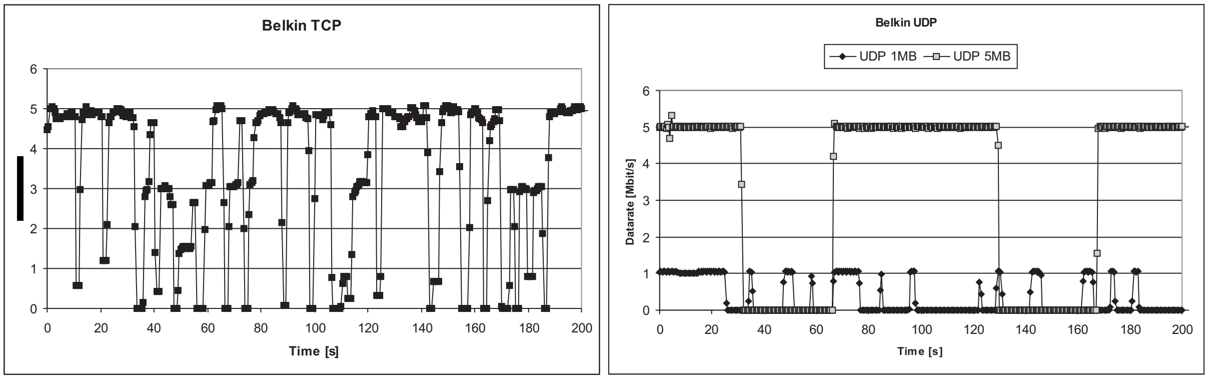


Figure 4.1: Belkin: rogue access point

TCP

The figure 4.1 (left) shows the expected behaviour of the datarate between the faked access point and the legal client. There are time slots in which a data transfer is possible, but also smaller time slots in which the client is disconnected. The datarate is

sometimes only at about the half as in the time slot before. This behaviour is characterisitc for the congestion control (see for more details [Kurose 05]) and shows the size of the congestion window. And in some points, it is possible to realize the increasing of the window size.
Regarding the time slots points out a arithmetic mean of approximatly 9s for slots in which traffic is possible and about 1,5s for the time, in which the client is not able to send any data. It is to mention, that there is a standard deviation of 7,82s for the time in which data transfer is possible, but only a standard deviation of 0,71s, in which no traffic is measured.

UDP

Taking a look at the UDP traffic points out, that the time slots are extremly depending on the bandwidth used to send the data. The small bandwidth, about 1Mbit per second has a lot of small time slots. Sending the data with 5MBit in contrast shows very long time slots.
Differentiating between the datarates, it is also neccessary to calculate the mean time for the 1Mbit UDP traffic and the 5Mbit one. The small bandwidth has a mean time for the tunnel about 9s and the mean of the no connection time is about 11,5s. The standard deviation for 1Mbit is nearly the same (about 8,5s). In contrast the 5Mbit UDP traffic has much bigger time slots. A rogue access point is able to serve the legal client about 36s (mean value). But also the slots in which no connection is possible is very high (about 26s mean). But the standard deviation is very different. For the connection time it is near to 27s. The standard deviation for the mean value of the no connection time is about 11s.

4.1.2 Linksys WRT54G

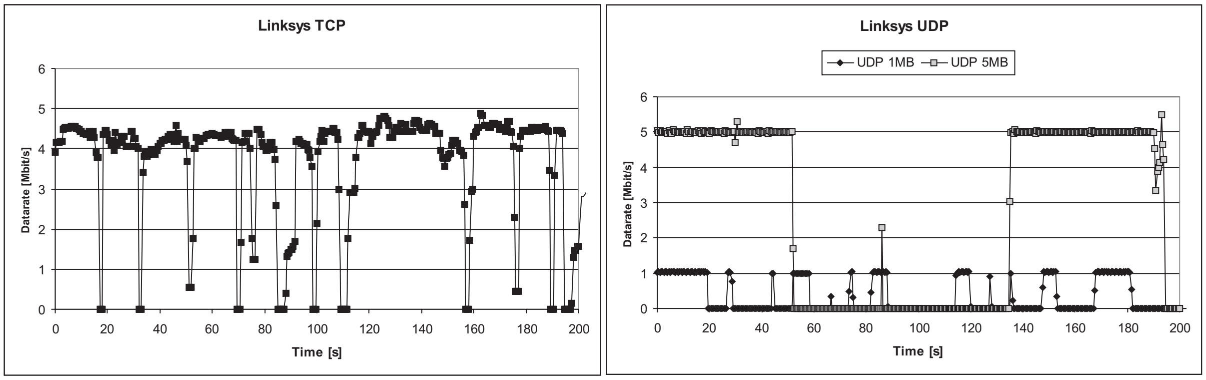


Figure 4.2: Linksys: rogue access point

TCP

The graph 4.2 on the left side shows the TCP traffic. As seen before, the channel between the legal client and the roque access point is continously disturbed by the Linksys WRT54G. There is a great difference between the connection time and the time in which no connection is possible. There is also a big variance of the measured time periods in which the rogue access point is able to serve the legal client.

Taking a closer look on the measured values, it is possible to calculate a mean value for the faked channel of about 16,5 seconds with a standard deviation of nearly 14 seconds. The time in which no traffic is possible is smaller than two seconds with a standard deviation smaller than one second. These results are similar to the one of the Belkin access point and shows the chracteristic behaviour of this attack using TCP traffic.

UDP

Examining the UDP graph 4.2 (right) shows the data throughput with a bandwidth of 1MBit per second and 5Mbit per second. As mentioned above by describing the behaviour of the Belkin F5D7130, UDP with 1MBit per second generates small time slots in which traffic is possible and not possible. The UDP traffic with 5MBit in contrast, has very big time slots in which the IPerf server is able to communicate with the IPerf client, but also very long time slots in which no communication respectively no connection is established.

Summing this up with the captured results, the mean of the time slots in which a connection is possible for the 1MBit UDP traffic is nearly 5 seconds with a standard deviation of about 5 seconds. The time in which no traffic is measured is in this case about 11 seconds (standard deviation of 6 seconds). Regarding the 5MBit UDP traffic points out a mean value for the connection time about 26,5 secondes (standard deviation of 25 seconds). But also the time in which no data is measured has a mean about 35 seconds with a standard deviation about 13 seconds.

4.1.3 Siemens T-SINUS 154 DSL

This device has no deauthentication delay as seen in chapter three. Making some measurements with a rogue access point in combination with the flooding attack points out some special behaviour.

Using IPerf with TCP traffic, the legal access point does not disturb the connection about 60 seconds. After this time, there is no new connection between the IPerf server and the IPerf client possible. It is neccessary to restart the T-SINUS 154 DSL to repeat this characteristic behaviour.

Another special behaviour is shown by using IPerf in UDP mode. In this case, the access point does not disturb the connection over the hole measurement.

4.1.4 Summary

Comparing the UDP and the TCP graphs and especially the captured results, it is possible to detect a characteristic behaviour at the beginning. Every measurement shows at the beginning a good connection in the first 11 seconds. This time slot is as long as the deauthentication delay. An attacker is able to serve a client with the rogue access point 11 seconds + x. x depends on the loss of the access point. In some measurements the attacker was able to have a connection near to 30 seconds for the TCP traffic and nearly one minute for the UDP traffic using a bandwidth of 5Mbit per second at the beginning of the communication.
An attacker is able to have a good connection about 11 seconds + x at the beginning and in mean smaller time slots after this time. These are not really long time slots, but regarding the login mechanism of some major WISPs, they start their service by asking the customer for their login data over a webserver with SSL (Secure Sockets Layer). A new station, that want to have access to the internet is redirected to this login mechanism. So, the attacker could fake this login page on his own webserver with SSL. In combination with his rogue access point, he is able to collect the user data from a customer, who wants to connect to the network. This seems to be a simple and efficient way to phish the users login data for the WISP.
Taking a look at the behaviour of a user who is browsing through the internet shows, that there are time slots in which the user does not need any data from the internet. For example, if he is reading something on a website or reading a mail over a webmailer. In this worst case for the user and with some luck, the user is not able to realize a disconnecting and reconnecting to the network. So the attacker is able to act as a man-in-the-middle and has the possibility to phish all data from the user.

4.2 Connection Times

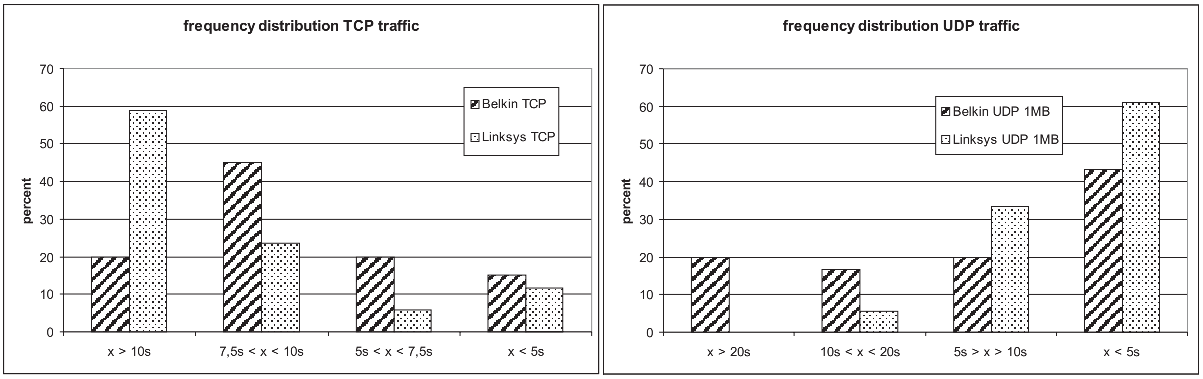


Figure 4.3: Frequency Distribution

This section gives a short overview about the measured time slots in which a connection is possible. The graphs 4.3 shows the frequency distribution for the faked connection. On the left side is the chracteristic behaviour for the TCP traffic and the right figure deals with the results for the UDP traffic for 1Mbit.
As seen in the left graph, the TCP traffic has a lot of time slots greater than 7,5 seconds. Especially the Linksys access point allows the rogue access point to have many time slots greater than ten seconds. Regarding the time in which no data traffic is possible points out, that about 70 to 80 percent of the time slots is smaller than two seconds. This is a small and perhaps sometimes not visible disconnecting from the network.
Taking a look on the right graph shows a very different beast. The majority of time slots are smaller than 5 seconds. And on the other side, the connection is disturbed more than five seconds in over 90 percent.
Summing up the results from the measurements once again, it is necessary to differentiate between the UDP and TCP traffic because of the different behaviour. In a realistic scenario, as shortly analysed in the following section, TCP is predominantly used for data transport in the internet.

4.3 A Faked Webserver with SSL

This section deals with two major aims. The first one is the attempt to redirect a user to a faked webserver with SSL[Schäfer 03][RFC4346 06] although the legal one is still running. And the second one is to disturb at best to destroy a SSL session in order to force the legal client to create a new session with our faked webserver.
In this scenario, there is no filtering or other special settings done. In real life, it would also be very hard for an attacker to manipulate the WISPs access points. The configuration is shown in figure 4.4. It is to mention, that without setting up a mac filter, the client is able to authenticate and associate with both access points. The tests are done with and without flooding the legal access point.

4.3.1 Results

Trying to redirect a connecting client to our faked website is not really successful. It is not possible to give a clear statement. On the one hand it is possible to redirect the client to the phishing website, on the other hand there are a lot of trials in which the user is able to visit the legal webpage. Flooding the legal device points out, that the attacker has a better chance to redirect the user to his phishing site. It stands to reason, that the legal client accepts the data from the first responding webserver. This seems to be like a first come first serve mechanism. The assumption can be confirmed by

the HyperText Transfering Protocol (HTTP). HTTP uses TCP as a reliable data transfer service. This implies that each HTTP packet emitted by the server arrives intact at the client side. A HTTP response which is already received will not be considered. Regarding SSL, placed between the TCP layer and the HTTP layer, uses also TCP because of its reliability and confirms also this assumption.
Disturbing the existing SSL session is only possible by flooding the legal access point. And in this way, the browser seems to be frozen for a long time before the it is redirected to the faked website. In many other trials the browser shows an error page like "'The page can not be found."' Another bad notice is the request to accept the SSL certificate, but this behaviour is expected and can be neglected for this test. In real life, many users are not really reading the SSL certificate and they will trust the visiting website without questioning.

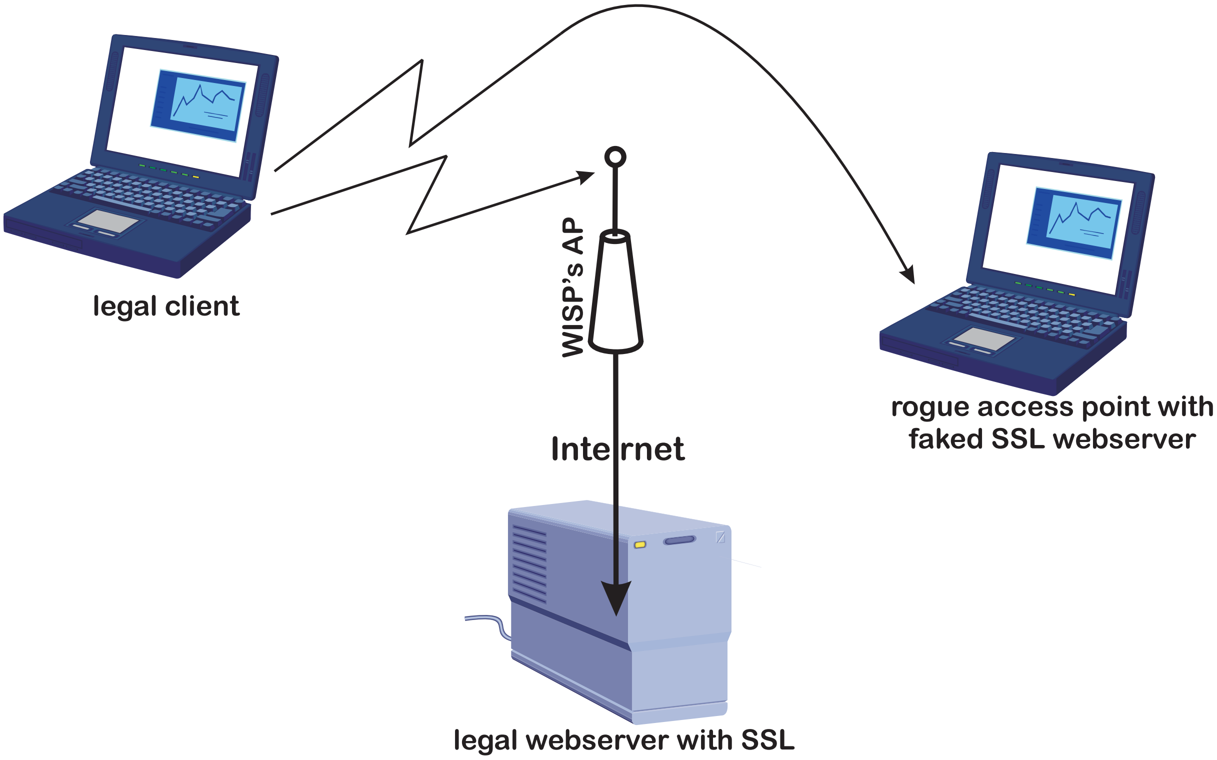


Figure 4.4: A Faked Webserver with SSL

4.4 Discussion

This chapter presents the possibility of this novel attack and gives a overview about the quality of the faked connection. As seen in the first two sections of this chapter, describing the quality of the connection and the connection times, there is a very high probability to hold a stable connection for over ten seconds. In combination with the results from the last section, an attacker has a good chance to redirect the connecting user and is able to phish his data. Conluding this results show that this attack can be done with little effort.

Taking a look on some more realistic scenarios. Firstly, a WISP would prefer a high-end access point like the Cisco or Lancom devices. They have no deauthentication delay and setting up a stable second access point would be very difficult. But it is to mention, that a client can be authenticated and associated with the legal and the rogue access point. By tweaking the settings of the rogue access point, perhaps by switching off the beacon frames, it could be possible to hide the faked device from the legal access point. Another approach is placing the attackers access point out of the legal access points sending range, but still visible for the legal client. Using this configuaration makes this attack again practical.

Regarding the results from the measurements and the realisitc reflections above deals with the possibility and feasibility of this attack.

5 Intrusion Detection System

This chapter describes the usage of a wireless intrusion detection system and the possible avoidance of the attacks described in chapter three. The used system is Snort[1] as a de facto standard for intrusion detection in combination with Snort-Wireless[2]. The first section explains the principle of an intrusion detection system.

5.1 IDS - a small overview

An Intrusion Detection System (IDS) detects anomalies in a computer network and helps the administrators to keep their system clean from any attackers. The system is used to analyse all malicious network traffic and computer usage that a conventional firewall is not able to detect. Especially attacks on applications, unauthorized access and malware like viruses, trojan horses and worms should be avoided by intrusion systems.

An IDS needs some components to monitor the network traffic (sensors) and a central unit to collect information and events, but also to create alerts by separating this data. In a wired network, the sensors must be placed to monitor all traffic. In wireless networks, the monitoring sensors must be able to collect every wireless packet sent to the network, that means there are as many IDS monitoring stations as access points in the network. This is the minimal number for the system. It is to differentiate between passive systems and reactive systems. A passive IDS detects a possible insecurity, logs the information and creates an alert signal. The reactive system, in contrast, is able to operate in the network for example to change the firewall settings or something like that. This system is also known as Intrusion Prevention System (IPS). It is to mention, that an IDS is not the same as a firewall. A firewall has to manage the network by limiting the access outwardly and block every access from outside, that is not allowed. But a firewall is not analysing the traffic, so that an attacker within the system cannot be detected. The IDS examines/scans the network communication and uses signatures like identifying heuristics and patterns of common computer attacks.

An example for an intrusion is the well known ARP spoofing. An attacker tries to change the ARP tables by sending faked ARP packets to the network. The system

[1] http://www.snort.org/

[2] http://www.snort-wireless.org/

has to detect these changes and must react to this disruption. In a wired network this is not really a problem, because the IP address and the belonging MAC address do not change very often. A wireless network is a little bit more difficult because of the changing infrastructure.

5.2 Snort - the de facto standard for intrusion detection/prevention

Snort[3] is an Open Source Project with more than 3 Million downloads and 150.000 active users. This intrusion detection system is most commonly used all over the world.

5.2.1 Snort Modes

The software can be run in four different modes[SnortManual 06]:

- sniffer mode: This mode monitors on the network and displays the sniffed packets in a continuous stream on the console
- packet logger mode: This mode is similar to the sniffer mode, but in this case all sniffed packets are stored to the disk.
- network intrusion detection system (NIDS) mode: This mode is the most complex and configurable configuration, which uses user-defined rules to analyse the network traffic and reacts based upon what is monitored.
- inline mode: Using packets from iptables instead of from libpcap makes this mode able to cause iptables to drop or pass packets based on Snort rules making use of inline-specific rule types

Unless otherwise noted, in the following only the network intrusion detection system mode is used.
The next part describes the defining of the rules, that Snort uses to perform several actions based on these guidelines.

5.2.2 Defining Rules

Using Snort as an IDS, it is necessary to define a file with several rules. The software uses a simple, lightweight rules description language that is flexible and quite powerful. It is possible to write most of the rules in a single line, but it is also possible to

[3] http://de.wikipedia.org/wiki/Snort

span the rules in multiple lines by adding a backslash to the end of the line.
Firstly, this work gives a quick overview of defining rules for Snort and secondly in more details, an introduction in writing rules for Snort Wireless.

5.2.3 Snort Rules

Snort rules are divided into rule header and rule options (figure 5.1). The header includes static definitions and has to be included in every rule. The rule options containing various definitions are not always necessary and more than fifty options are available.

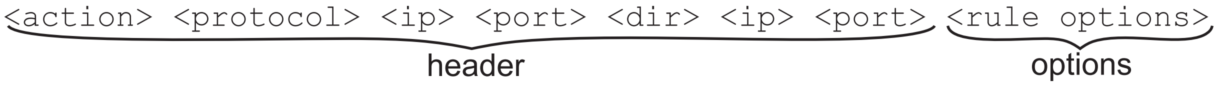

<action> <protocol> <ip> <port> <dir> <ip> <port> <rule options>
header
options

Figure 5.1: Snort Rules Format

action
The action command tells Snort what to do when it finds a packet matching this rule. The following commands are possible, depending on the running mode (drop, reject and sdrop can only be used in inline mode):

- alert - generates an alert, depending on the selected alert method
- log - logs the packet
- pass - ignores the packet and lets it pass
- activate - alerts and turns on a dynamic rule
- dynamic - is idle until activated by an activate rule
- drop - drops and logs the packet
- reject - drops and logs the packet, then sends a TCP reset if the protocol is TCP or an ICMP port unrechable if the protocol is UDP
- sdrop - similar to drop, but does not log the packet

There is a possibility to define own rule types that can be used as rule actions.

protocol
Snort currently uses four protocols: TCP, UDP, ICMP and IP.

ip

The IP addresses are given by a straight numeric IP address and a CIDR (Classless Inter-Domain Routing) block. For example *192.168.0.0/24* stands for the addresses between 192.168.0.1 to 192.168.0.255 (class C network) or 192.168.0.1/32 specifies a single machine address. Using the keyword *any* indicates to look after any IP addresses. It is also possible to use a list of IP addresses, for example *[192.168.0.0/24, 10.1.1.0/24]*. Another facility is to use a *!* as negation. If Snort should react on any traffic from outside the network, the rule contains something like *!192.168.0.0/24* (if the network is in the range 192.168.0.1 to 192.168.0.255).

port

The port numbers are defined as a single port like *80* or a range of ports *4999:8000* with the range operator *:*. If there is no number before or after the colon, Snort looks from this port number upward or downward. Example: *5000:* defines ports greater or equal 5000, *:1024* defines ports less or equal 1024. Port numbers can also be used with the negation operator *!*. *!80* looks after all ports excepting port 80 for http.

dir

The direction is given by ->,<- and <>. Using -> as direction operator means, that the left side is considered to be the traffic coming from the source host and the right side is the destination. <> tells Snort to analyze both sides of conversation (such as telnet or POP3 sessions).

rule options

The rule options are very expensive, so this work represents only a small part of all possibilities.
Snort has four major categories of rule options:

- meta-data - these options provide information about the rule but do not have any effect during detection
- payload - these options are looking for data in the payload of the packet
- non-payload - these options are looking for non-payload data
- post-detection - these options are rule specific triggers that happen after a rule has fired

meta-data options are for example *msg* or *priority*. *msg* tells the logging or alerting engine to print a message to a packet dump or to an alert. The *priority* keyword is used to assign rules a priority level to specifiy the severity of the rule.
The most important **payload** option is the *content* option. Using this keyword allows the user to set rules searching for specific content in the packet payload. The search

string can contain mixed text and binary data. This keyword can be used with the *!* as negation operator.
non-payload options are for example *ttl*,*flags* or *seq*. Detecting the attempt of a traceroute, the IP time-to-live value is very important. For checking this value the *ttl* keyword is used. Checking if specific TCP flag bits are present, the rule writer can use the *flags* keyword. Using *seq* as keyword checks the packet for a TCP sequence number.
post-detection can be used with keywords like *logto* or *react*. Logging packets to a specific file, the user can use the *logto* keyword to specifiy a filename. In order to block some interesting sites a user wants to visit, the rule file can include the *react* keyword with the mode *block* to close an offending connection and send a visible notice to the browser.

5.2.4 Snort-Wireless Rules

Defining rules for Snort-Wireless (chapter 2 in [SnortWirelessUserGuide 03]) is similar to the normal Snort rules excepting the IP address and the port number, they are replaced by MAC addresses. Wireless Snort extends the Snort rules with a new protocol called *wifi*. Therefore the format 5.2 is similiar to the above shown Snort Rules format 5.1(excepting the MAC addresses and the protocol).
In the following the format is explained and subsequently the rule options are described in more details.

$$\underbrace{\texttt{<action> wifi <mac> <dir> <mac>}}_{\text{header}}\ \underbrace{\texttt{<rule options>}}_{\text{options}}$$

Figure 5.2: Snort-Wireless Rules Format

action

The rule actions are the same as described for the Snort rules.

protocol

As mentioned, the rule writer has to use *wifi* to define rules for Snort-Wireless.

mac

The MAC address can be a single one (for example: *00:DE:AD:BE:EF:00*) or a list of addresses (for example: *[00:DE:AD:BE:EF:00, 00:DE:AD:C0:DE:00, ...]*).

dir

The direction is similar to the normal Snort rules, excepting the direction <-. In order to define rules, it is possible to use -> as unidirectional operator (source to destination) or the bidirectional operator <> (source to destination or destination to source).

rule options

Snort-Wireless extends the Snort rule options with a lot of 802.11 specific rules. Therefore many Snort standard rules can be used with the *wifi* protocol.
The following table gives an overview of the Snort-Wireless specific keywords:

keyword	**description**
frame_control	checks the frame control field
type	checks the 802.11 frame type
subtype	checks the 802.11 frame subtype
from_ds	checks the from distribution system frame control flag
to_ds	checks the to distribution system frame control flag
more_frags	checks the more fragments frame control flag
retry	checks the retry frame control flag
pwr_mgmt	checks the power management frame control flag
more_data	checks the more data frame control flag
wep	checks the wep frame control flag
order	checks the order frame control flag
duration_id	checks the frame duration/id field
bssid	checks the frame BSSID
seqnum	checks the frame sequence number
fragnum	checks the frame fragment number
addr4	checks the frame 4th address field
ssid	checks the frame SSID

Table 5.1: Snort-Wireless Keywords

The keywords are detailed below:

frame_control

The frame_control option checks if the entire field contains a specific value. The 802.11 control field has 16 Bits, so the format allows a value in hexadecimal or decimal notation between 0 and 65535. *!* can be used as negation.
Usage: *frame_control: [!]<number>;*

type

This option checks the 802.11 frame types. As above, a *!* can be used as logical NOT. The following types (wifi types) are valid:

- STYPE_MANAGMENT
- STYPE_CONTROL
- STYPE_DATA

Usage: *type:[!]<wifi type>;*

stype

The subtype option is similar to the type option and the following subtypes (wifi subtypes) are valid (using this option implies checking the frame type):
management frame subtypes:

- STYPE_ASSOCREQ
- STYPE_ASSOCRESP
- STYPE_REASSOC_REQ
- STYPE_REASSOC_RESP
- STYPE_PROBEREQ
- STYPE_PROBERESP
- STYPE_BEACON
- STYPE_ATIM
- STYPE_DISASSOC
- STYPE_AUTH
- STYPE_DEAUTH

control frame subtypes:

- STYPE_PS
- STYPE_RTS
- STYPE_CTS
- STYPE_ACK

- STYPE_CFEND
- STYPE_CFEND_CFACK

data frame subtypes:

- STYPE_DATA
- STYPE_CFACK
- STYPE_CFPOLL
- STYPE_CFACK_CFPOLL
- STYPE_NULL
- STYPE_CFACK_NULL
- STYPE_CFPOLL_NULL
- STYPE_CFACK_CFPOLL_NULL

Usage: *stype:[!]<wifi subtype>;*

from_ds

This option checks the from_ds flag. Valid arguments are ON, OFF, TRUE, or FALSE. *!* can be used as logical NOT.
Usage: *from_ds:[!] TRUE | FALSE | ON | OFF ;*

to_ds

Similar to the from_ds option this option checks the to_ds flag.
Usage: *to_ds:[!] TRUE | FALSE | ON | OFF ;*

more_frags

The more_frags option allows the rule writer to check the more_frags flag. Valid arguments are ON, OFF, TRUE or FALSE. *!* is used as negation.
Usage: *more_frags:[!] TRUE | FALSE | ON | OFF ;*

retry

This option is used to check the retry flag. Valid arguments are ON, OFF, TRUE, or FALSE. As negation the *!* operator can be used.
Usage: *retry:[!] TRUE | FALSE | ON | OFF ;*

pwr_mgmt

Checking this flag can be used to detect if the transmitting device is in power-save mode. Valid arguments are ON, OFF, TRUE, or FALSE. For consistency, *!* may proceed the argument to perform a logical NOT operation on the comparison.

Usage: *pwr_mgmt:[!] TRUE | FALSE | ON | OFF ;*

more_data

This option checks if the more data control flag is set. Valid arguments are ON, OFF, TRUE, or FALSE. *!* is used as negation.

Usage: *more_data:[!] TRUE | FALSE | ON | OFF ;*

wep

The arguments ON, OFF, TRUE, or FALSE check if the frames have been processed by the WEP algorithm. *!* is used as logical NOT.

Usage: *wep:[!] TRUE | FALSE | ON | OFF ;*

order

The order option allows Wireless-Snort to check if the 802.11 frames, that are being transmitted, are using the strictly-ordered service class. Valid arguments are ON, OFF, TRUE, or FALSE. *!* makes a logical negation.

Usage: *order:[!] TRUE | FALSE | ON | OFF ;*

duration_id

This option checks if the duration/ID field has a specific value. The 802.11 duration/ID field has 16 Bits, so the format allows a value in hexadecimal or decimal notation between 0 and 65535. *!* can be used as negation.

Usage: *duration_id:[!]<number>;*

bssid

Using this option allows the rule writer to check the BSSID of a 802.11 frame. The BSSID is a 48-Bit hexadecimal number, so valid arguments are between 0x000000000000 and 0xFFFFFFFFFFFF. *!* is used as a logical NOT operation on the comparison.

Usage: *bssid:[!]0x00DEADBEEF00;*

5.2.5 Using Snort as Protection System

The idea is to use Snort Wireless as protection system. There are two main aims to be discussed in this work: The first is the detection of a flooding attack and the second aim is to realize a faked access point.

It is to mention, that it is necessary to have at least one device in monitoring mode running Wireless Snort to analyse the wireless network traffic. This device must be placed in such a way, that it is possible to sniff on all frames sent to this infrastructure.

Authentication Flooding

Detecting this attack is as simple as the attack itself. The attempt is to collect all authentication frames and to have a possible limit per second. If this limit is reached or exceeded, the access point has to block these requests. A Snort rule to log the authentication frames is the following:
log wifi any -> MAC_AP (stype: STYPE_AUTH; msg: "authentication frame";)
Depending on the logging mode, Snort writes a simple format with a timestamp and log message to a file. This file can be parsed by another process to count the requests per second.
As predescribed, the solution to detect this attack is simple but not really a new approach. Regarding some access points in chapter three, there is still a similar protection provided by the access points themselves. As discussed, the protection is an easy way to protect the device against the flooding attack, with the drawback of opening the eventuality of another Denial-of-Service attack. Blocking all requests, avoids legal stations connecting to the network.

A Rogue Access Point

Regarding the used configuration of the rogue access point in this work (cloning the legal one), it is nearly impossible to locate the faked service. There are predefined rules to solve this problem, but they are all based on finding the rogue device by detecting a not allowed mac address in the same BSSID. Using this mac address for the faked device makes the predefined rules not applicable for detecting the cloned access point. Other solutions based on analysing the traffic in the wired area are only possible, if the faked services use the legal infrastructure for their client communication to the legal network or the Internet.

ARP Spoofing

An ARP spoofing attack as described in Phishing in the Wireless: Implementation and Analysis[Martinovic 07b] uses a special configuration. Setting the FromDS and ToDS flag to true, the legal AP cannot check whether this frame is a legal or illegal one (the AP cannot know all other APs in a distribution system). The following rule can be used to detect this ARP poisoning:
alert wifi any <> any (from_ds: TRUE; to_ds: TRUE; msg: "possible ARP spoofing"’;)
If there is a wireless brigde in the network, this rule cannot be used because of alerting legal and faked ARP frames.

5.2.6 Related Work

In Specification-Based Intrusion Detection in WLANs[Gill 06], the system uses a wireless sensor to monitor the radio frequency and constructs a state transition model for each station and associated access point that the device is able to sense. The approach is to detect anomalies by using the state transition model, so all stations should strictly transition through the sequence of this model. This intrusion detection system requires the Extensible Authentication Protocol (EAP) and therefore has nine states. In order to present only the principle of this mechanism, this work will not explain the EAP protocol.
There are three possible state shifts:

- a negative state shift, which occurs when the station transitions from a higher state to a lower state
- a positive state shift, which occurs when the station bypasses an incremental state
- a zero state shift, which occurs when the station does not change its current state

A negative state shift can be a Denial-of-Service attack, but does not have to. So the system uses an index of suspicion for every station. This index is incremented for every negative shift. If the index exceeds a threshold value, the monitoring system has to send an alert.
A positive state shift is normal, because of the sequentially moving through the states. If the state shift is greater than one, this is a possible indication for a spoofing, session hijacking or man-in-the-middle attack. It is to mention that there could also be a packet loss. Therefore another index of suspicion is used with a predefined threshold for every station. If the number of postive shifts rise above this threshold, the monitoring system alerts this condition as an excessive frame loss.
There is only one state, in which a legal station remains. If a station stays repeatedly in another state, a misconfiguration or a DoS attack is supposed. Retransmissions of traffic could also results in remaining in one state. In order to counteract this behaviour, there is also an index of suspicion similar to them above.
Depending on the current state of the station, there are only a few accepted frames as expected. All other frames between the access point and the client are assumed to be unexpected. Therefore, another index is used to count all unexpected frames.

Authentication Flooding

Taking a look at the authentication flooding attack from chapter three, this system is not able to detect this simple attack. Every authentication request is send with a new random mac address. Therefore, this detection mechanism generates an index of suspicon for every request and has to save the state. This behaviour results in wasting resources for saving all necessary information.

A Rogue Access Point

In order to detect a faked service in the distribution system, the intrusion detection system has to interchange information with the access points. However, the system as noted above is not developed to use any information from the legal point coordinators. Detecting whether a station is connected to a legal or a faked service is not really possible with this mechanism.

ARP Spoofing

The mechanism does not implement any analysis of class three frames, because of the assumption that the data traffic is encrypted. An ARP spoofing attack as described in [Martinovic 07b] for an encrypted 802.11 network is not subject of this work.

5.3 Discussion

There are possibilities to set up an Intrusion Detection System in order to discover the above mentioned attacks, some of them can only be detected with an high implementation effort, e.g. rogue access point. However, realizing such an attack does not automatically protect the devices in the distribution system. For example, a faked deauthentication frame cannot be stopped in the medium air and as a result, the attacked device is forced to disconnect from the network.
Summing up, such an detection system, as the name already says, is able to detect these attacks, but there is no prevention.

6 Conclusion

The first part of this work deals with the analysis of various access points. As seen in chapter three, a simple authentication flooding attack is still able to stress or even to crash some of the tested devices. This flooding can be used as a starting point for various other attacks. It is to mention that a flooding attack is only one possibility to stress an access point. An example for another possibility is a vulnerability in Cisco Aironet Wireless Access Points that has been released in January 2006[1]. Surely, there are a lot of other approaches to crash or stress an access point still being unknown or at least not official.

As shown in chapter three and four, an attacker is still able to use a well known and maybe obsolete flooding attack. The advantage to stress the legal point coordinator and to fake its mac address opens new facilities for the attacker. Earlier attacks on a wireless infrastructure based on a better signal strength of the rogue access point. A connecting station would choose the access point with the best signal qualitiy. But this attack can be easily detected by using a wireless device in monitoring mode. For example, the Cisco AIR-AP1231G-E-K9 can be used as a Lightweight Access Point (LAP) in different modes like the above mentioned monitoring mode for detecting rogue access points or as an intrusion detection system (see also [2] especially the question: What are the different modes in which a lightweight access point (LAP) can operate?). However, using the same mac address for the rogue access points makes it very difficult for any detection system to differentiate between the legal and the rogue service.

Another aspect to be noted is the low price of this attack. An evil user needs a mobile device and two wireless interfaces. One for the flooding of the access point and the other has to act as rogue access point. There are no costs for powerful antennas with high gain or similar tools.

There are approaches to solve these flooding problems. Another thesis at the distributed computer science lab deals with the problem of multiple associations or authentications of a station to an access point. Therefore a regional-based WLAN access control[Wu 07][Martinovic 07a] is developed based on a listen-before-talk procedure. This procedure generates neighbourhood lists depending on the signal strength. Gathering this information by the access point, it is able to differentiate between a legal connecting client and a flooding attack. In theory this is a suitable solution to solve the flooding problem. Taking a practical look on this method shows some serious disadvantages. There is a high number of false positives, but also approaches to solve

[1] http://www.cisco.com/en/US/products/products_security_advisory09186a00805e465b.shtml

[2] http://www.cisco.com/warp/public/102/lap_faq.pdf

this problem are explained[Martinovic 07a]. Although, future research is necessary in this area.

Protecting management frames (Management Frame Protection MFP) is an alternative. The leading manufacturers like Cisco but also the distributed computer science lab at the University of Technology in Kaiserslautern are working on this subject. In order to present the state-of-the-art of this development, the author tries to contact various producers of wireless equipment. Unfortunately, only Cisco and D-Link answered this information request. D-Link does not give any official information about their current progress in this area. Cisco on the other hand, answers with a lot of stuff but without giving any detailed information. Even today, there is a possibility to apply management frame protection by using supported Cisco devices as described in this document[3]. It is to mention, that Cisco protects only the access points, the clients are still vulnerable. The principle of the protection mechanism is to add a message integrety check information element (MIC IE) to each frame. Attempting to copy, alter or replay the frame invalidates the message integrity check, which causes any receiving access point that is configured to detect MFP frames to report the discrepancy. Another feature is the management frame validation which can be enabled by the administrator of the access point. Using this validation ensures that the MIC IE is present and matches the content of the management frame. If the receiving frame has no valid MIC IE but the BSSID belongs to a MFP frames transmitting AP, the detecting device reports the discrepancy to the network management system.

The Cisco access points are secured by this mechanism, but there is no protection for the station. An attacker is still able to send a disassociation frame to disturb a connected client. The network management system is surely informed by an access point with the management frame validation check enabled, nevertheless the station will be disassociated. Another disadvantage is the constraint to use Cisco products. Other wireless equipment is not compatible with this proprietary protection mechanism.

An Intrusion Detecting System offers the possibility to discover all mentioned attacks, but in some cases only with a high effort. At best the access points in the distribution system can be protected, but not the clients. Providing the above noted detection and protection, an information exchange must be ensured between the point coordinators and the detection system.

Therefore, attackers are still able to run a Denial-of-Service (DoS) on the WLAN or to attempt a man-in-the-middle attack on the client when it reconnects (as seen in chapter four). In 2005, the IEEE 802.11 Task Groug w (TGw) was established with the aim of creating a standard for authentication of management and control frames with an expected draft due 2008 (status of the project[4]). There will be no standard until 2008 for protecting the management and control frames and also then there might be a lot of devices not being able to handle this new standard. Furthermore, the WISPs are interested in providing their service to anybody with the drawback that the customers

[3] http://www.cisco.com/en/US/tech/tk722/tk809/technologies_configuration_example
09186a008080dc8c.shtml

[4] http://grouper.ieee.org/groups/802/11/Reports/tgw_update.htm

have to take care of security themselves.

Bibliography

[Gill 06] Rupinder Gill, Jason Smith, Andrew Clark. Specification-Based Intrusion Detection in WLANs. In *Proceedings of the 22nd Annual Computer Security Applications Conference on Annual Computer Security Applications Conference*. IEEE Computer Society, 2006.

[Kurose 05] James F. Kurose, Keith W. Ross. *Computer Networking - A Top-Down Approach Featuring the Internet*. Pearson Education, Inc., 2005.

[Martinovic 07a] Ivan Martinovic, Frank Zdarsky, Jens Schmitt. Regional-based Authentication against DoS Attacks in Wireless Networks. -, Juni 2007.

[Martinovic 07b] Ivan Martinovic, Frank A. Zdarsky, Adam Bachorek, Christian Jung, Jens B. Schmitt. Phishing in the Wireless: Implementation and Analysis. In *Proceedings of the 22nd IFIP International Information Security Conference (SEC 2007), Sandton Johannesburg, Gauteng, South Africa*. Springer, Mai 2007.

[RFC2581 99] TCP Congestion Control. http://tools.ietf.org/rfc/rfc2581.txt, April 1999.

[RFC4346 06] TLS Version 1.1. http://www.ietf.org/rfc/rfc4346.txt, April 2006.

[Schäfer 03] Günter Schäfer. *Netzsicherheit - Algorithmische Grundlagen*. dpunkt.verlag GmbH, 2003.

[SnortManual 06] Snort Users Manual. http://snort.org/docs/, Dez. 2006.

[SnortWirelessUserGuide 03] Snort Wireless User Guide. http://www.snort-wireless.org/docs/usersguide/, Juli 2003.

[Tews 07] Erik Tews, Ralf-Philipp Weinmann, Andrei Pyshkin. Breaking 104 bit WEP in less than 60 seconds. -, April 2007.

[Wu 07] Shujun Wu. Regional-based WLAN Access Control. -, April 2007.