

Improving The Security of Wireless LANs By Managing 802.1x Disassociation

Ping Ding, JoAnne Holliday
 Computer Engineering Dept
 Santa Clara University
 Santa Clara, CA 95053, USA
 {pding,jholliday}@scu.edu

Aslihan Celik
 OMIS Department
 Santa Clara University
 Santa Clara, CA 95053, USA
 acelik@scu.edu

Abstract

802.1x is a security protocol based on the frame structure of 802.11. It attempts to provide strong authentication, access control, and WEP key management for Wireless LANs. Unfortunately, 802.1x misses its goals in access control denial-of-service attacks. Currently, there are no IEEE approved ways to solve the security hole. We propose a Central Manager not only to take the responsibility of an authentication server, but also to add functionality to prevent denial of service attacks. We also analyze how the 802.11 MAC layer protocol works with our solution.

Keywords: 802.11, Network Management, Network Security, Protocols, Performance

1. Introduction and Related Work

The security holes in the wireless LAN standard IEEE 802.11 are well known[1]. IEEE has proposed a secure architecture for 802.11 called Robust Security Network, or RSN. RSN uses the recently approved IEEE standard for Port-based Network Access Control, 802.1x. 802.1x takes advantage of an existing authentication protocol known as the Extensible Authentication Protocol[2] to provide centralized authentication of wireless clients. EAP messages are encapsulated in 802.1x messages and referred to as EAPOL. 802.1x authentication for wireless LANs has three main components: The supplicant (the client software); the authenticator (the access point); and the authentication server (AS, usually Remote Authentication Dial-In User Service server).

Based on RSN, IEEE provides an 802.11/802.1x state machine (Figure 1) to provide strong authentication, access control, and WEP key management for Wireless LANs. Unfortunately, 802.1x does not realize all its security goals since it is still vulnerable to disassociation attacks. Mishra and Arbaugh[3] describe these attacks which we summarize in section 2. All these attacks are done with management frames. Since IEEE has stated that all management frames must be unencrypted in 802.11, management frames should be kept clean

in 802.11 or 802.1x. In industry, some companies use keyed integrity check (IC) to prevent rogue disassociation, such as PEAP (Microsoft[4]). Keyed IC uses a key generated from a seed value, source and destination MACs and payload and the key is included to every WEP packet. The method is time consuming and only solves disassociation after the WEP key has been generated. We propose a Central Manager(CM) to dynamically manage access points (APs) and clients for mitigating disassociation attacks. In section 2, we introduce disassociation attacks (also called DOS attacks); in section 3, we introduce how the CM works and explain the algorithms; in section 4, we evaluate the performance of our algorithms and conclude.

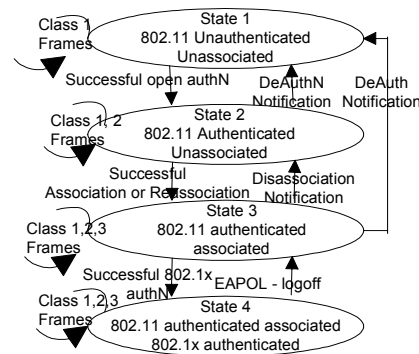


Figure 1. 802.11/802.1x state machine

Table 1. 802.11/802.1x state transition functions

Management Frame\AP state	S 1	S 2	S 3	S 4
EAP-Start	NA	NA	S 1,4	NA
EAP-Failure	NA	NA	S 1	NA
MACdisassoc/EAPOLlogoff	S 1	S 2	S 1	S 1
802.11 Association	S 2	NA	NA	NA

2. Denial of service (DOS) attacks

Based on the state machine, we can write a state transition function as in Table 1. NA means the AP does not accept the management frame in that state. For example, after an AP receives EAP-Start request in State 3, the AP's state could be State 1 or State 4. After the AP in state 1 receives 802.11 association, the AP's new state is State 2. The AP will not know its next state until the authentication succeeds or fails. However, after an AP receives EAP-Failure, MAC disassociation or EAPOL

logoff (State 4), the AP's status goes to State 1. The reason is that the AP cannot identify whether the management frame comes from a client or an attacker till authentication is complete. We call these DOS attacks or disassociation attacks and summarize them here:

Large number of association requests An attacker continuously uses random MAC addresses to do association with an AP, making the AP busy working with the attacker and preventing any other clients from joining the AP. This is called LASO.

EAPOL Logoff In this attack, the attacker spoofs the client's MAC address and sends EAPOL logoff request to the AP, making the AP disassociate with its authenticated client, thus denying service to the client.

EAP-Start, EAP- Failure spoofing In this attack, an attacker continuously sends EAP-Start request, making an AP busy with the authentication dialog and unable to handle legitimate traffic. Or, an attacker continues to send EAP-Failure, then, the AP disassociates with its legitimate client.

MAC disassociation Attackers continue to send MAC disassociation, making an AP disassociate with an authenticated or authenticating client.

3. Avoiding DOS attacks

We propose to use a CM to dynamically manage a large number of APs and their clients. The CM is a back-end server that takes the place of the AS defined by 802.1x. It not only takes the responsibilities of the AS, but also tracks clients in the authentication process to avoid the DOS attacks described in section 2, and helps in load-balancing of the APs. The authentication service of the CM is unchanged from 802.1x. In this paper, we explain how the CM avoids DOS attacks.

3.1 Terminology

Priority of login(**PLI**): 0 means the priority of login is low. 1 means the priority of login is high.

Priority of logout(**PLO**): 0 means the priority of logout is low. 1 means the priority of logout is higher than login.

Times of login(**TLI**): The number of login requests received by the CM.

Times of logout To CM (**TLOT**): The number of logout requests received by the CM.

Times of logout from CM (**TLOF**): The number of logout requests sent by the CM.

Table for AP (T1): T1 records all the information for the access points. CM generates and manages the table. It contains the MAC address and location of the AP, and the number of authenticated and unauthenticated clients for that AP.

Table 2. Authenticated table for clients (T2)

Clients' MAC address	PLO	TLO
*****	1	0

Authenticated table for clients (T2): It records all the information of authenticated clients. It includes the MAC addresses of the clients and the time the clients are authenticated (login). The default values of T2 are given in table 2.

Unauthenticated table for clients (T3): It records all the information of unauthenticated clients. Pre-Association time (PAT) is the time a client starts association. The default values of T3 are given in table 3.

Table 3. Unauthenticated table for clients (T3)

Client MAC	PLI	TLI	PLO	TLOF	TLOT	AP MAC	PAT
*****	1	0	0	0	0	*****	***

All vulnerable management frames can be separated into two categories. The first is the login part, such as EAP-Start. The second is the logout part, such as: EAPOL logoff, MAC disassociation or EAP-Failure. All management frames are forwarded to the CM by APs. An AP does not respond to management frames until the CM instructs to it. The CM avoids DOS attacks by using the T1, T2 and T3 tables. Management frames are kept clear and unencrypted as defined by the IEEE, however, if the WEP key has already been generated, then the CM will send an encrypted request packet to the client using the WEP key. If the WEP key has not been generated, then the CM will use T3 to decide its next step.

We define a time interval of 30 seconds. This means that a request needs to get response in no more than 30 seconds. It is suggested in IEEE EAP that a request be resent a minimum of 3 times before terminating the authentication[2]. In our design, a client could try three times before the CM sends an EAP-Failure message to the client.

3.2 LASO attacks

We introduce Pre-Association which is the process that starts when a client sends an 802.11 association request. After an AP recognizes the association request, the packet will be forwarded to the CM. The CM gets the MAC addresses of both the client and the AP. After checking in T1, CM sends a success or denial packet to the client. If the reply packet is a success packet, it means the client

can continue the authentication process. The AP will give the client an association id and the 802.11 association is a success. If the packet is a denial packet with the location of a suggested AP, it means the client needs to connect to the suggested AP or another AP or the association failed. The CM will decide if the current connection is reasonable based on the network load. The Pre-Association process is finished when the reply message comes to the AP. For example, assume the CM finds that AP-A already has 10 clients, but its neighbor AP-B does not have any clients. When a new client sends a association packet to AP-A, the CM will suggest that the client associate with AP-B. If the CM sends a success packet to the client, the CM writes the client information to T1 and T3. The client will be an unauthenticated client to AP-A. The client also gets an association id from AP-A. A client is allowed to continue 802.1x authentication if it gets an association id or it is in T3. If the client is already in T3, by default, it can associate with the AP it associated with the last time. However, the client has to get an association id through re-association. In re-association, if the CM finds the client is already in T3, the CM just leaves the client's information as it is and sends a success packet to the AP. Pre-Association effectively avoids a large number of association attacks. An attacker does not get the chance to make an AP busy and deny service to others.

3.3 EAPOL logoff attack

After an AP receives an EAPOL logoff packet, the AP will forward the packet to the CM. Once the client is authenticated, it is considered to be in the TLS channel. If the client is already in the TLS channel, the CM will send an encrypted request packet to the client using the WEP key. In the request packet, the CM will ask the client if it wants to logoff. After the client receives the request packet, it needs to give a response, either confirmation or denial. If the CM receives a confirmation message, it will send a logoff-continue message to the AP, then the AP disassociates with the client. If the CM receives a denial message or does not get a message from the client, the CM will send a logoff-ignore message to the AP. The AP will ignore the logoff request and keep the client's current status.

The variables and tables defined in section 3.1 are used by the CM to avoid the EAPOL logoff attack. If the CM receives confirmation message from the client, the CM deletes the authenticated client from T1 and T2. If the CM receives a denial message from the client, the CM adds 1 to TLO.

This repeats until TLO equals 3, at which time the CM will update PLO to be 0. If the PLO is 0, the CM will ignore EAPOL logoff request for the rest of the time interval. After the current time interval is over, the CM will set all the client information in T2 to be its default value. Figure 2 shows how the CM manages EAPOL-logoff request. Figure 3 shows the detail of the UpdateT2 process.

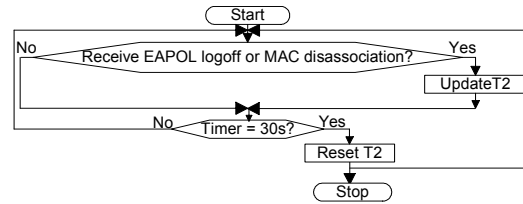


Figure 2. CM manages T2

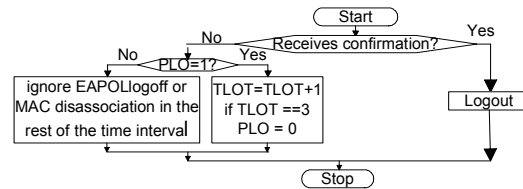


Figure 3. UpdateT2

3.4 EAP-Start & EAP-Failure spoofing and MAC disassociation attacks

3.4.1 Mitigating EAP-Start spoofing attack

When an attacker sends an EAP-Start, the AP responds to the attacker and makes the attacker start an 802.1x authentication process with the AS. The process stops when the AS cannot identify the attacker. The AS will send an EAP-Failure to the AP. The AP disassociates with the attacker. In this scenario, two things could happen. Before the AS sends EAP-Failure, the attacker sends several EAP-Start; or the AS could identify and send an EAP-Failure to the AP immediately after an EAP-Start request. In addition, there can be a combination of these two cases. The CM can successfully mitigate this attack using PLI and PLO. The algorithm is shown in Figure 4.

After a client finishes pre-association, the CM adds the client information to T3. When the CM receives an EAP-Start request, it will also add one to the TLI. T3 is updated by setting PLI=1, TLI=1, PLO=0, TLOF=0, and TLOT=0. The CM and the client now do mutual authentication based on 802.1x. If 802.1x authentication is successful, then the client information will be deleted from T3 and added to T1 and T2. If the 802.1x authentication failed, the CM just keeps the client's information in T3. Before the CM identifies and

sends an EAP-Failure to the AP, the attacker may continue to send an EAP-Start request. Every time when the CM receives an EAP-Start request message, TLI increases by one. The CM will only allow AP responses to the EAP-Start message three times. After the CM finds TLI equals three and TLOF equals zero, the CM will update PLO to one and ignore any EAP-Start request in the current time interval. If the AP receives an EAP-Failure message from the CM, the AP will disassociate with the client. In the second case, the CM sends EAP-Failure message to the AP immediately after receiving an EAP-Start request. When the CM sends an EAP-Failure to the AP, the CM increases TLOF by one. The CM will allow the process to repeat till TLOF to three. At that time, the CM will set PLI to zero and PLO to one. If the CM receives EAP-Start again, the CM will let the AP ignore EAP-Start request in the rest of the time interval.

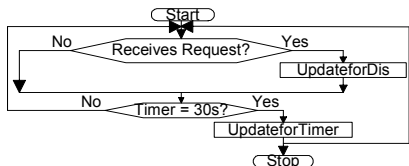


Figure 4. The CM manages T3

T3 is updated regularly in the time interval (see UpdateforTimer in figure 5). If a client's PLO equals one or the client stays in T3 more than 30 seconds (current time (CT) minus Pre-Association time (PAT) longer than 30 seconds), the client's information will be deleted from T3. Otherwise, the client's information is kept in T3. The timer in the CM does not interfere with the time stamps contained in the beacon frames of the AP.

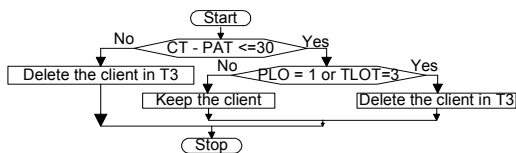


Figure 5. Update for Timer

3.4.2 Mitigating EAP-Failure spoofing attack

The EAP-Failure attack works when a client is in the process of authentication. The attacker sends an EAP-Failure message to the AP and the AP disassociates with the real client.

The CM uses TLOF to mitigate this attack. The algorithm is shown in Figure 4. When the CM receives an EAP-Failure from the client, The CM will add one to TLOF. The CM lets the AP disassociate with the client and keeps the client's information in T3. The process repeats until TLOF

equals three. Then, the CM will ignore any EAP-Failure messages for the rest of the time interval. The real client gets a chance to continue its authentication process and further attacks are prevented. T3 is updated regularly during the time interval (see UpdateforTimer in Figure 5). If a client's TLOF equals three or the client stays in T3 more than 30 seconds (current time (CT) minus PAT longer than 30 seconds), the client's information will be deleted from T3. If the TLOF is less than three and CT minus PAT is less than 30 seconds, the client's information is kept in T3.

3.4.3 Mitigating MAC disassociation attack

In this attack, an attacker sends a MAC disassociation to an AP before or after a client is authenticated making an AP disassociate with the real client. In our algorithm, T2 and T3 are used to mitigate this attack.

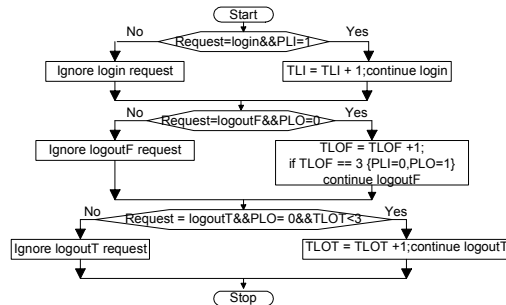


Figure 6. Update for Dis

After the CM receives a MAC disassociation request, the CM checks in T2 and T3. If the CM finds the client is in T2, it means the client is currently in state 4. The process will be the same as when the CM receives an EAPOL logoff request. If the CM finds the client is in T3, it means the client is in state 1, state 2 or state 3. The process is the same as when the CM receives an EAP-Failure message. Figure 4 shows how the CM manages T3. Figure 5 shows the UpdateforTimer process. Figure 6 shows the UpdateforDis process. In Figure 6, Request equals login means the CM receives an EAP-Start request. Request equals logoutT means the CM receives an EAP-Failure message. Request equals logoutF means the CM sends an EAP-Failure message to an AP.

4. Performance Evaluation

To evaluate our algorithms, we define two metrics: **Throughput** (the number of packets that can pass through in a fixed time) and **Response time** (the interval from the time a user requests a service to the time the service is granted).

4.1 802.11 MAC layer protocols

The MAC architecture is composed of two basic coordination functions: PCF[5] and DCF. Both of these functions define an operation mode for the clients that want to access the wireless medium. We address the PCF model in this paper. In PCF model, time is divided into super-frames. A super-frame is composed of a contention free period (CFP) where PCF works and a contention period (CP) where DCF works.

CP The basic DCF is CSMA/CA. Carrier sensing is done through physical and virtual mechanism. A client senses the medium to check if another client is transmitting in the physical medium. The virtual carrier-sense is done by distributing reservation information with RTS/CTS exchanges in MACAW.

CFP PCF uses a centralized, contention-free polling. It is done by software, which is installed in the AP and polls clients. Before a PCF polling cycle, the AP contends with other clients in DCF, so, the CFP may vary from super-frame to super-frame. However, the AP only needs to wait a PIFS period, which is shorter than a DIFS period. Because of the priority, DCF will not interrupt CFP. To prevent starvation of clients that are not allowed to send during the CFP, the CP is at least long enough to transmit one max length frame. The AP polls the clients in a round-robin fashion. After a client finishes association with an AP, the AP gives an AID to it and puts the client in its polling list based on the client's AID. A polled client always responds to a poll. If there is no pending transmission, the response is a null frame with no payload. If the CFP terminates before all clients have been polled, the polling list is resumed at the next client in the following CFP cycle.

4.2 Throughput and response time

For convenience, we calculate the throughput (S_{SUPER}) and response time (T_{SUPER}) by considering the valid packets in each super-frame. The S_{SUPER} and T_{SUPER} are calculated for each of the two parts of a super-frame, that is, CFP and CP. We assume a super-frame includes at most N clients which could be polled in CFP and the CP is exactly long enough to transmit a max length frame. We don't separate management and data frames, because they have to wait the same time to be processed.

4.2.1 CP Contribution to S_{SUPER} and T_{SUPER}

Throughput (S_{CP}) Because DCF functionality is based on random techniques and is used by

asynchronous traffic, we assume: 1) An event is a packet arrival with Poisson distribution. 2) Arrivals occur at random over a time interval. The inter-arrival rate is λ and the length of the average idle period is $I=1/\lambda$. 3) All of the packets have the same propagation delay, τ , and it is much smaller than packet duration P (including RTS, CTS, SIFS and ACK period).

The probability of K packet arrivals in $[0, \tau]$ is:
 $P_k = P \{k \text{ arrivals in } (0, \tau]\} = (\lambda\tau)^k e^{-\lambda\tau} / k!$

$P_0 = P \{0 \text{ arrivals in } (0, \tau]\} = e^{-\lambda\tau}$ and also $P \{one \text{ or more arrivals in } (0, \tau]\} = 1 - P_0 = 1 - e^{-\lambda\tau}$

The probability of successful transmission is P_s (P_0) and the probability of collision is $1 - P_s$. MACAW does not guarantee to prevent collisions[6]. The collision duration will be the maximum length of the frames, because of no collision detection. Two mutually exclusive events could happen in a busy period time (B); a frame successfully transmits, the transmission time (T_s) is $P+\tau$; or, a collision occurs, the transmission time (T_C) is also $P+\tau$. So; $B = P_s * T_s + (1 - P_s) * T_C = e^{-\lambda\tau} * (P + \tau) + (1 - e^{-\lambda\tau}) * (P + \tau) = P + \tau$

The utilization period (U) is the period of successful transmission that has no overhead.

It is $U = P e^{-\lambda\tau}$. The Throughput of CP is:
 $S_{CP} = U / (T_{DIFS} + I + B) = P e^{-\lambda\tau} / (T_{DIFS} + 1/\lambda + P + \tau)$

$G = \lambda P$, $a = \tau/P$

$S_{CP} = G e^{-aG} / (1 + \lambda T_{DIFS} + G + aG)$

Figure 7 shows the throughput of CP in a packet time. $S_{SUPER-CP}$ reduces as G or parameter- a grows. This means the S_{CP} gets less when the probability of collision grows.

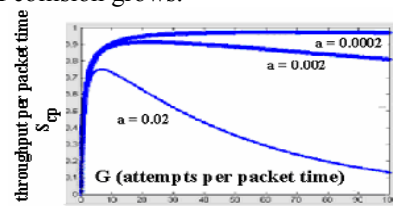


Figure 7 Throughput of S_{cp}

Response Time (T_{CP}) When a client wants to access the medium, it senses the channel. If the medium is idle, the clients can access the medium immediately after waiting DIFS. The shortest T_{CP} equals $T + 1/\lambda + T_{DIFS}$. If the medium is busy, a client has to wait until it is free. The client waits for a time given by the exponential backoff algorithm. T_{CP} is $T + \sum_{i=0}^n (1/\lambda + T_{DIFS} + K_{CPi} * 50)$. If i equals 0, K_{CPi} equals 0; otherwise, K_{CPi} is randomly chosen from 0 to 2^{i-1} ; and n is the times of checking medium (max 12). T is the time used by CFP.

4.2.2 CFP Contribution to S_{SUPER} and T_{SUPER}

Throughput (S_{CFP}) The AP polls the client using round-robin algorithm. The clients are polled in the order of their Association id. The throughput of a CFP is: $S_{CFP} = N$.

Response Time(T_{CFP}) The AP needs to contend in CP to start the polling cycle. The shortest is $1/\lambda + T_{PIFS}$. If the medium is busy, the AP has to wait $\sum_{i=0}^n (1/\lambda + T_{PIFS} + K_{CFPi} * 50)$. If i equals 0, K_{CFPi} equals 0; otherwise, K_{CFPi} is randomly chosen from 0 to 2^{i-1} ; and n is the times of checking medium (max 12). The response time for a client is T_{CFP} and dependent on the polling sequence during the CFP. Therefore, T_{CFP} can be expressed as $\sum_{i=0}^n (1/\lambda + T_{PIFS} + K_{CFPi} * 50) + K_{client} * T_{SIFS}$, where K_{client} is the sequence number of the client in the polling ready queue.

The throughput of a superframe (S_{SUPER}) is: $S_{SUPER} = S_{CP} + S_{CFP} = P e^{-\lambda \tau} / (T_{DIFS} + 1/\lambda + P + \tau) + N$. The response time of a client is: $T_{CFP} = \sum_{i=0}^n (1/\lambda + T_{PIFS} + K_{CFPi} * 50) + K_{client} * T_{SIFS}$ and $T_{CP} = T + \sum_{i=0}^n (1/\lambda + T_{PIFS} + K_{CFPi} * 50)$. All of parameters are as defined previously.

4.3 Analysis and conclusion

T_{SUPER} To start CFP, AP must sense an idle medium. If not, it has to wait and try again, making T_{CFP} longer. If an attacker is put in the ready queue, some clients must be put behind the attacker making T_{CFP} longer. In CP, the increased contention from the attacker makes T_{CP} longer.

S_{SUPER} In CFP, when T_{CFP} is longer, the working time for CFP becomes shorter in a super-frame time period; then the ready queue in CFP cannot be polled entirely, so S_{CFP} reduces. In CP, when the network load becomes heavy, S_{CP} reduces and thus, S_{SUPER} reduces.

From the analysis, we can easily conclude in CFP both S and T are mostly determined by wasted polling and delay; in CP, both S and T are mostly determined by collision and delay. We already know the chance of success for a small number of clients is much higher than for a large number of clients. The probability of success will drop close to its asymptotic value of $1/e$, as soon as the number of clients reaches 5. If attackers continue to send management packets, this affects the polling order in CFP and aggravates the contention for the medium in CP. This is because when the client is disassociated as a result of the attack, it joins the contention in the CP to re-associate. So, T_{CFP} and T_{CP} become longer; S_{CFP} and S_{CP} reduce. For

example, if attacks continue to send MAC disassociation, clients will be kicked out of the polling ready queue, then, these clients join the contention in CP. We summarize the affect of these attacks on the performance of S and T in table 5.

Table 4 Effect of DOS to S_{SUPER} and T_{SUPER}

	EAPOL-logoff	EAP-Start	EAP-Failure	MAC Disasso	LASO
S_{super}	reduce	reduce	reduce	reduce	reduce
T_{super}	increase	increase	increase	increase	increase

We cannot prevent an attacker from sending messages, but we can manage the AP so it does not respond to an attackers request or put them in its polling list. This reduces the probability of attackers contending with other clients for the medium. Attackers will not cause DOS attacks and affect the performance of the WLANs. The time an AP waits for instructions from the CM should be much less than the wasted polling time or added collision time. The CM successfully manages the APs and the clients and improves security and performance of WLANs.

In this paper, we reviewed the denial of service attacks in 802.1x. We proposed a Central Manager to dynamically manage APs and clients and avoid denial of service attacks. We analyzed 802.11 MAC protocol to demonstrate the effect of our algorithms on the performance of Wireless LANs. Our solutions not only improve the security of WLANs, but also improve the performance of WLANs. We plan to implement a prototype next.

REFERENCES

- [1] N.Borisov, L.Goldberg, D.Wagner, "Intercepting Mobile Communications: The Insecurity of 802.11". Proc., Seventh Annual International Conference on Mobile Computing and Networking, July, 2001, pages 180-188.
- [2] RFC 2284 March 1998, www.armware.dk/RFC/rfc/rfc2484.html
- [3] A.Mishra, W.A. Arbaugh An Initial Security Analysis of the IEEE 802.1x Standard Feb, 2002, www.ieee802.org/1/files/public/docs2000/ieee_plenary.PDF
- [4] Protected EAP Protocol (PEAP) Feb, 2002, Microsoft, internet -draft <http://www.globecom.net/ietf/draft/draft-josefsson-pppext-eap-tls-eap-02.html>
- [5] Moustafa A. Youssef, Arunchandar Vasan, Raymond E. Miller: Specification and Analysis of the DCF and PCF Protocols in the 802.11 Standard Using Systems of Communicating Machines 10th IEEE International Conference on Network Protocols (ICNP'02) Nov. 2002
- [6] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang, "MACAW: A Media Access Protocol For Wireless LAN's" ACM SIGCOMM, 1994. <http://citeseer.nj.nec.com/bharghavan94macaw.html>