World Academy of Science, Engineering and Technology
International Journal of Electronics and Communication Engineering
Vol:4, No:12, 2010

Weaknesses and Strengths Analysis over Wireless Network Security Standards

Daniel Padilla, *Member, IEEE,* and Edward Guillen, *Member, IEEE*

Abstract—Several wireless networks security standards have been proposed and widely implemented in both business and home environments in order to protect the network from unauthorized access. However, the implementation of such standards is usually achieved by network administrators without even knowing the standards' weaknesses and strengths. The intention of this paper is to evaluate and analyze the impact over the network's security due to the implementation of the wireless networks security standards *WEP*, *WPA* and *WLAN* 802.1X.

Keywords—802.1X; Vulnerabilities Analysis, WEP; Wireless Security; WPA.

I. Introduction

Wireless local area networks –*WLANs*– have been widely implemented and for more than ten years now have proliferated in such an unthinkable way. The *WLANs'* ease of installation, combined with the mobility it provides, have allowed a large number of corporate environments to install and even to completely migrate their networks to wireless technologies. Even tough the mobility is limited to a coverage ratio, it certainly allows the network's users to move within small areas and to easily connect and access the network services. The growing phenomenon of *WLANs* implementation has not only spread within corporate environments, but also has benefited from the expansion of broadband penetration to reach the home users' attention. A large number of residential users initially deployed *WLANs* with the first purpose of sharing the Internet connection, but it has evolved through the simple task of sharing files, to allowing and achieving the wireless interconnection of several entertainment devices and common home appliances.

However, the information sent through wireless communication media can be highly sensible and vulnerable to several types of attacks. Using a public channel such as the air to achieve data transmission then requires the implementation of at least one wireless security standard that protects the information's confidentiality, integrity and availability. The wireless security standards aim to provide the required privacy level by implementing users access control and encrypting the transmitted data. Nevertheless, these standards in some cases fail in fulfilling its primary objectives and allow the unwanted access of users. By using a set of attack techniques and exploiting the standards' vulnerabilities an intruder can gain access to the system and once inside the whole information is susceptible of being sniffed and manipulated [1].

The previously stated situation creates the need to examine, document and analyze the failures and vulnerabilities present in wireless security standards such as Wired Equivalent Privacy –*WEP*–, Wi-Fi Protected Access –*WPA*– and *WLAN* 802.1X.

The second part of the paper provides an overview of the three wireless network security standards *WEP*, *WPA* and *WLAN* 802.1X. Evaluation criteria are described and explained in the third section. Fourth part shows the network scenario used during the test. Finally, the results are presented and analyzed in the fifth section of the paper.

II. Wireless Network Security Standards

WLANs deployments under the 802.11 set of standards, also known with the wireless technology brand Wi-Fi, use a public transmission channel and work in conjunction with security standards, whose main objectives are to implement access controls by authenticating users and to provide encryption techniques in order protect the transmitted information.

WEP, *WPA* and *WLAN* 802.1X are nowadays three of the most important and implemented wireless security standards and hence the main attack objectives when trying to trespass wireless access controls or sniffing sensitive information.

A. Wired Equivalent Privacy

Defined and documented in the *IEEE* 802.11 Standard [2], the Wired Equivalent Privacy –*WEP*– is a data confidentiality algorithm that aims to offer the functionalities of wired local area networks –*LANs*– by providing protection to wireless *LANs* users and to the data they send. The data confidentiality directly depends on the methods used to distribute encryption and decryption keys.

WEP algorithm has characteristics such as being reasonable strong, self-synchronous and efficient. *WEP* works over a 40- or 104-bit key that can be changed frequently. This hinders to find out the key by using brute force attacks. In addition, *WEP* auto-synchronizes in every single message, which is an important feature if we take into account that the data link layer encryption methods assume the best-effort data delivery and the packet loss can be high. *WEP* is also considered to be efficient because its implementation can be achieved by both software and hardware.

Figure 1 graphically represents the *WEP* encryption process, which begins with three items: the plaintext, the *WEP* key and an initialization vector –*IV*–. The *IV* is a 24-bit randomly generated sequence that strengths the Ron's Code 4 –*RC4*–

D. Padilla and E. Guillen are with the GISSIC Investigation Group, Department of Telecommunications Engineering, Nueva Granada Military University, Bogotá, Colombia. E-mail: danielpadilla@ieee.org, edward.guillen@unimilitar.edu.co.

Manuscript received October 4, 2010; revised December 25, 2010.

algorithm [2] by granting different inputs and this is why the the system should never use the same *IV* more than once.

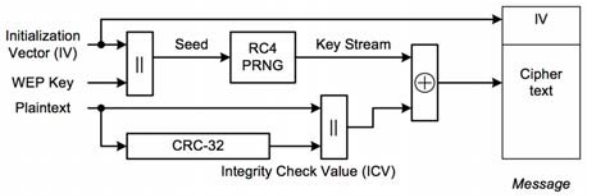

Fig. 1. WEP Encapsulation

The encryption algorithm initially performs an integrity check value –*ICV*–, which consists of a 32-bit cyclic redundancy check –*CRC*– over the message and its concatenation at the end of the plaintext. In addition, the *IV* and the *WEP* key are joined and then inputted into the *RC4* Pseudo Random Number Generator –*PRNG*– [3]. The procedure generates a keystream. The message is finally encrypted by performing an exclusive or –*XOR*– operation between the plaintext with its *CRC* and the keystream. The *IV* is added at the beginning of the encrypted text and included as part of the transmitted data.

The *WEP* decryption consists in exactly the inverse of the encryption process. The receiver reads the message and identifies the *IV*. The keystream is obtained by concatenating the *IV* with the *WEP* key and inputing it into the *PRNG*. The message is decrypted by performing a *XOR* operation between the keystream and the encrypted text. A *CRC-32* is finally performed in order to check the message integrity. The process is shown in Figure 2.

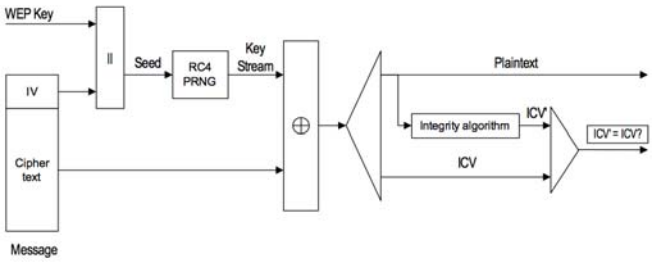

Fig. 2. WEP Decapsulation

B. Wi-Fi Protected Access

In contrast to *WEP*, Wi-Fi Protected Access –*WPA*– was not defined by *IEEE* but by the Wi-Fi alliance in order to improve the *WEP* standard [4]. The Wi-Fi alliance groups the manufacturers of devices based on the *IEEE* 802.11 standards, however, it also aims to establish as an *IEEE* 802.11i subset.

WPA implementation can be achieved in two different modes: enterprise (also known as 802.1X) and pre-shared key. The *WPA* enterprise mode uses authentication servers and works over the set of protocols *RADIUS* [5], [6] in order to grant the keys authentication and distribution. The enterprise mode is further explained during the *WLAN* 802.1X subsection. The *WPA* pre-shared key mode, on the other hand, authenticates users on the access point by verifying a key previously shared throughout secure and reliable channels.

WPA works over an encryption scheme that includes the Temporal Key Integrity Protocol –*TKIP*– [2]. The *TKIP* process initiates with two keys: a temporal key –*TK*– (138-bit encryption key) and a message integrity code –*MIC*– key (64-bit pre-shared key). A phase 1 key (*TTAK*) is obtained by mixing the transmitter's *MAC* address (*TA*) and the *TK*. The phase 1 key is then hashed with a *TKIP* sequence code (*TSC*) in order to produce the phase 2 output, which is further used as a 128-bit *WEP* key (*IV+RC4* key). The remaining process occurs as a traditional *WEP* transaction. The main difference is that due to the phase 1 procedure, the clients do not share the *WEP* key and there is no more correlation between *IVs*. Figure 3 shows the process described above.

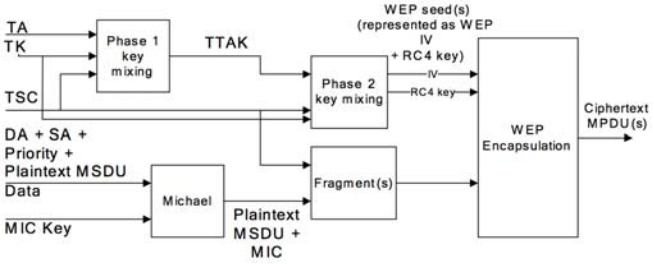

Fig. 3. TKIP Encapsulation

Another important issue of the *WPA* standard is the association process. When a station –*STA*– aims to associate with a *WPA* access point –*AP*–, the 4-way handshake depicted in Figure 4 must be accomplished. The pairwise master key is a key derived from the pre-shared key by using the RSA's Password-Based Key Derivation Function –*PBKDF2*–.

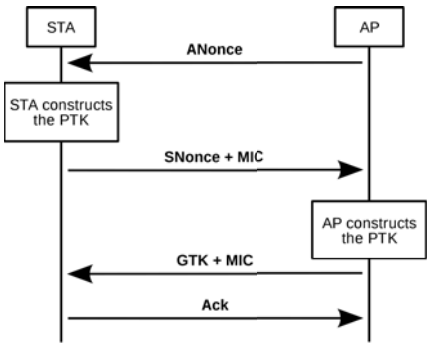

Fig. 4. WPA Association

The *AP* generates a nonce-value and sends it to the *STA* (ANonce). The client uses the received message to construct a pairwise transient key –*PTK*–, which is a value derived from the pre-shared key and the client address, the ANonce and the SNonce. The *STA* then sends its own nonce-value (SNonce) to the *AP* together with a *MIC*. The *AP* sends the group temporary key –*GTK*– and a sequence number together with another *MIC*. The *STA* finally replies by sending a confirmation to the *AP*.

C. WLAN 802.1X

802.1X protocol allows the network administrators to implement port-based authentication. Even though it was not initially defined for wireless networks, 802.1X can be implemented over a *WLAN* in order to improve its security. On wired *LANs*, 802.1X works by blocking the network's *RJ-45* ports and unblocking them after success authentications. On *WLANs* environments, 802.1X controls the network's access by considering clients as *RJ-45* virtual ports.

802.1X has its roots in the point-to-point protocol –*PPTP*– [7], which was originally designed for telephone lines and further used in Digital Subscriber Line –*DSL*– connections. The 802.1X use of *PPTP* then evolved to the Extensible Authentication Protocol –*EAP*– [8], which in comparison with *PPTP* allows the implementation of any authentication standard (biometric, certificates, intelligent cards, etc.) even if it is not yet developed.

802.1X can be, in fact, implemented over wireless networks as a *WPA* extension and improves it by solving *WPA* security issues such as key management and brute-force attacks.

A 802.1X network consists of three elements: the supplicant, the authenticator and the authentication server. The 802.1X supplicant is the network user or client that requests access to the wireless network, the authenticator (usually the access point –*AP*–) blocks or permits the traffic flow and the authentication server –*AS*– manages the authentication information and over 802.11 environments is a *RADIUS* server.

The 802.1X standard aims to implement a flow control over the *MAC* Service Data Unit –*MSDU*– between the distribution system and the end devices. The process is performed by using the protocol 802.1X's port control model, which uses the additional authentication entity.

Previous to the 802.1X authentication procedure, an association must be performed, which is graphically explained by Figure 5. Each association between the *STA* and the *AP STA* creates a single pair of 802.1X ports and the association is relative only to those ports [9].

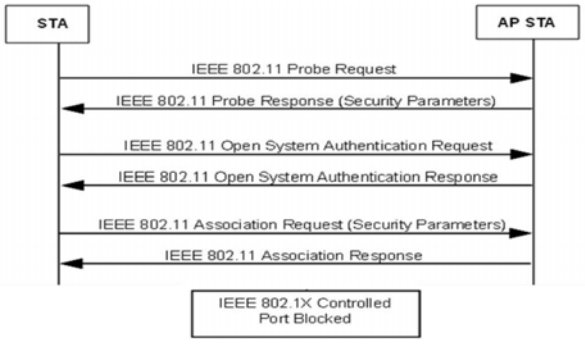

Fig. 5. WLAN 802.1X Association

The authenticator works as a firewall. It will only allow 802.1X traffic until a user authentication is achieved. Once the user is authenticated, its whole traffic is permitted. This function is accomplished by using two virtual ports. The authentication is performed via the 4-way handshake over the *EAP*. The process is shown in Figure 6 [9] and carried out between the supplicant, the authenticator and the *AS*.

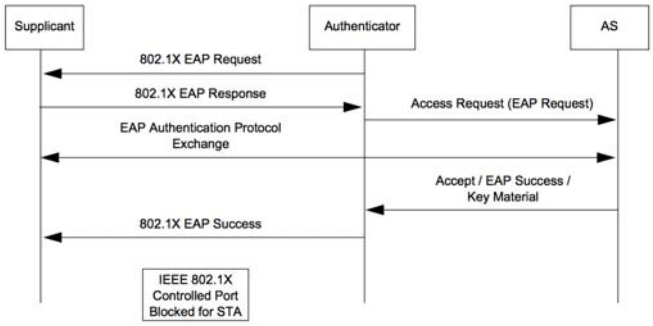

Fig. 6. 802.1X Authentication 4-Way Handshake

The 802.1X authentication frames are sent to the authenticator within 802.11 data frames throughout an 802.1X uncontrolled port. The 802.1X controlled port blocks the whole traffic between devices until an authentication process is successfully accomplished through the uncontrolled port. The port blocking is, in fact, responsibility of both the supplicant and the authenticator entity.

III. Evaluation Criteria

Wireless network security standards were assessed by analyzing parameters such as standard's vulnerabilities, risk value, attack and penetration susceptibilities, and required knowledge level for implementation procedures. The results obtained from the tests provide an overview of the security failures and network behaviour due to the implementation of *WEP*, *WPA* or *WLAN* 802.1X standards.

A. Standard's Vulnerabilities

Standard's vulnerabilities –*SV*– evaluation variable corresponds to the vulnerabilities found for the wireless security standards. This variable represents the start point of the research project.

B. Weighted Risk Value

Risk value –*RV*– is a variable that assesses the vulnerabilities risk by taking into account the impact and the damage produced to the wireless network. *RV* takes a value between 1 and 5, where 5 represents the highest *RV*. The risk is indeed given and inverse proportional to the ease with which an attacker can exploit vulnerabilities in order to gain access to the network. The 1 to 5 score is hence determined in accordance with the following parameters: 1–Low Risk, 2–Potential Risk, 3–Moderate Risk, 4–Significant Risk and 5–High Risk.

The weighted risk value –*WRV*– is in fact obtained by applying the equation (1), where *n* represents the number of vulnerabilities found for a specific standard and *m* defines the totality of vulnerabilities found for the three standards.

$$WRV = \frac{\sum_{i=1}^{n} RV_i}{\sum_{j=1}^{m} RV_j} * 5 \quad (1)$$

C. Attack Susceptibility

Attack Susceptibility –*AS*– is a variable that assesses how susceptible is a vulnerability to be exploited and how complex is to develop a certain attack against the wireless security standard. *AS* takes a value between 0 and 5, where 0 means that the vulnerability is not susceptible to being exploited. An *AS* value equal to 1 corresponds to a low-susceptible vulnerability and therefore a very difficult to develop attack. An *AS* value equal to 4 represents, on the contrary, a highly susceptible vulnerability and means that the attack development is very easy.

The *AS* value is determined by the following parameters: 1–Development of a structured attack that requires prior planning and the use of both free and licensed software tools, 2–Tracking of commands or procedures that require licensed tools, 3–Tracking of commands or procedures with open source tools, whose availability is limited, 4–Tracking of commands or procedures with open source tools, whose availability is free, and 5–Do not require any tool external to the used operating system.

The open source tools' availability is defined free when the tool is downloadable directly at the developers' official webpage. An availability is defined limited, on the other hand, when the tool does not have an official download site and must be obtained by using alternative methods such as peer-to-peer transferences or visiting hackers communities.

D. Penetration Susceptibility

Penetration Susceptibility –*PS*– is a variable that evaluates and quantifies how susceptible is the network to be penetrated by assessing the time taken by an attacker during the exploiting of a certain vulnerability and penetrating the network. *PS* takes a value between 0 and 5, where 0 means that the network could not be penetrated. *PS* values are determined within an inversely proportional relation with the penetration time, as follows: 1–Greater than 1 day, 2–Greater than 12 hours and less than or equal to 1 day, 3–Greater than 1 hour and less than or equal to 12 hours, 4–Greater than 10 minutes and less than or equal to 1 hour, and 5–Less than or equal to 10 minutes.

E. Knowledge Level

The knowledge level –*KL*– variable assesses the required network administrator's knowledge and expertise level to successfully achieve the implementation of the wireless security standards under consideration. *KL* is evaluated using a 1 to 5 scale and imply cumulative expertise, for example, a *KL* value equal to 5 requires the expertise of *KL* values 1, 2, 3 and 4.

KL is indeed determined by the administrator's knowledge and expertise on the following areas: 1–Use of Internet, 2–Windows Server operating system, 3–Linux operating system, 4–Servers configuration and 5–Database management.

F. Standard's Final Value

From the variables explained above an overall value is calculated. The standard's final value –*SFV*– directly depends on the variables *WRV*, *AS*, *PS* and *KL* and is obtained by solving equation (2).

$$SFV = \sqrt{\frac{WRV^2 + AS^2 + PS^2 + KL^2}{4}} \quad (2)$$

The *SFV* quantitatively evidences the standard's failures, weaknesses and implementation difficulties. Hence, the low the *SFV* is, the best the standard performs.

IV. Network Scenario

The network implementation for test procedures was achieved in a single scenario. The tested network scenario is shown in Figure 7 and represents a traditional wireless *LAN* where users share the communication channel through an *AP*.

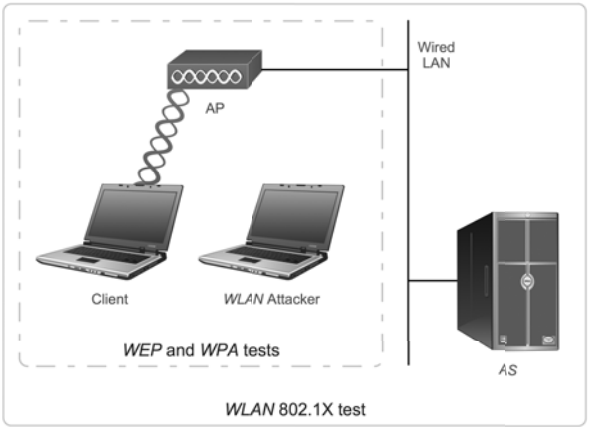

Fig. 7. Network Scenario

For *WEP* and *WPA* vulnerabilities tests, the scenario consists of three elements: two hosts and an *AP*. One of the hosts acts as the client device that has the legitimate access to the wireless network and the other host is the attacker, whose main objective is to penetrate the standard. Since the implementation of *WLAN* 802.1X requires additional authentication, the security tests scenario includes an *AS*.

V. Results

A. Standard's Vulnerabilities

Seven different vulnerabilities were found for the analyzed wireless network security standards. The vulnerabilities and risks are described and analyzed below.

Table I summarizes the five vulnerabilities found for the *WEP* security standard.

Table II describes the two vulnerabilities found for the *WPA* standard.

WLAN 802.1X, on the other hand, showed that its correct implementation protects the network from being unauthorized accessed. It is also important to state that the key management vulnerability is totally controlled if the 802.1X implementation uses the Transport Layer Security –*TLS*– [12], [13] as the *EAP* method. By doing this, the need of passwords is obviated and the vulnerability is indeed removed.

Figure 8 depicts the risk values of every single vulnerability found and explained above. The standards' *WRV* is then obtained by solving equation (1).

TABLE I
WEP VULNERABILITIES.

Vulnerability	Description
Key Management	*WEP* uses a symmetric encryption algorithm and indeed the encryption and decryption keys are the same. The key must be shared within the users group, however, the 802.11 protocol does not specify the way in which the key can be distributed. This situation does not represent a problem if the users group is limited to 3, but what if the number of users raises to hundreds? Just one case of inadequate key management and the entire network is breached.
IV Collisions	Corresponds to the event that occurs when an *IV* is reused during the encryption process. The cause of the vulnerability is that *WEP* does not specify the way of implementing *IV* generators and it is unclear if the *IVs* must be chosen in a random way, if they may begin with a value equal to 0 and increment by 1, or decrement from 16,777,215. When the system generates an *IV* Collision, the concatenation of the key and the *IV* produces a keystream that was used before. Since the *IV* is sent in plain text in conjunction with the encrypted message, an attacker can easily identify the collision by keeping and storing a register of the network's traffic. When two packets are product of the same *IV*, the *XOR* operation between the two encrypted texts generates the same result as the *XOR* operation between the two plain texts. This situation evidences the case that if the attacker have one plaintext, can easily obtain the other. There are two techniques that can be used in order to obtain an original plaintext. The first technique can be launched only if the victim is visible on the Internet, and consists in sending a packet, whose payload is known. In this case the attacker only have to wait until the packet's *IV* is reused by the system. The second technique consists in trying to guess the content of packets generated by protocols such as *DHCP* or *ARP*, whose structure is always the same and clearly documented.
Messages Injection	When a keystream is known, a new encrypted message can be achieved by simply performing a *XOR* operation between the keystream and a plaintext. Since the 802.11 protocol does not require the *IV* to be different, the devices accept packets with reused *IVs*.
Authentication Deception	During the *WEP* authentication process, the client sends an authentication request to the *AP*, which responds with a 128-bit plaintext challenge. The client encrypts the text by using the *WEP* key and sends it back. The *AP* decrypts the packet, compares it with the plaintext and sends to the client the positive or negative response. If an attacker is able to capture the negotiation procedure explained above, he will also know both the plaintext challenge and the encrypted challenge, and by using the messages injection method can obtain the keystream, request the authentication to the *AP* and use the same keystream to generate a valid encrypted challenge. The attacker is finally authenticated without even knowing the *WEP* key.
Key Generators	Key generators allow the users to use a simple *ASCII* password in order to obtain a *WEP* key, instead of directly inputting the hexadecimal numbers key. Even tough the use of such generators is completely proprietary and makes no part of the standard, several manufacturers use the same generator algorithm and it has been proved that these key generators present several security failures [10]. One of them, for instance, evidenced that the process of generating 40-bit keys included a 32-bit input into a *PRNG*, but since the first bit of every ASCII character is always 0, the input reduced from 32 to 21 bits. This situation decreased the attack time (at a rate of 60,000 attempts per second) from 210 days to just 35 seconds [11].

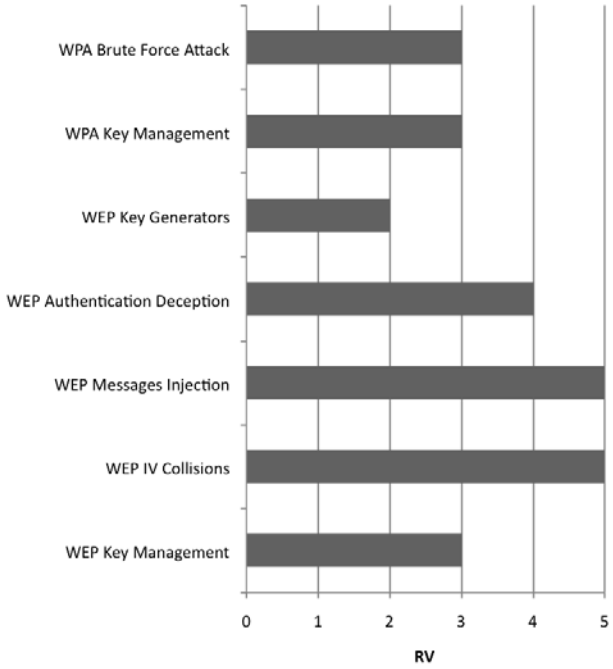

Fig. 8. Vulnerabilities Risk Value

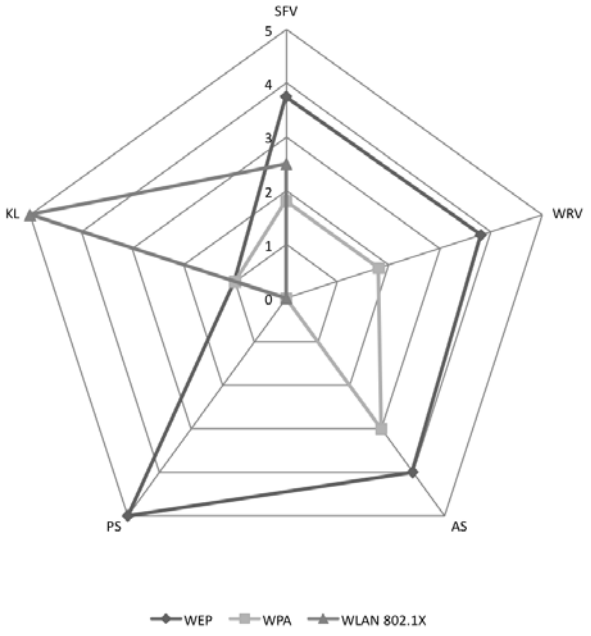

Fig. 9. Standard's Final Value

B. Standard's Final Value

Figure 9 shows the results obtained for the *SFV* variable in conjunction with its sub-variables *WRV*, *AS*, *PS* and *KL*.

By analyzing the number of vulnerabilities found for each standard, the vulnerabilities' *RV* and the *WRV*, it can be

TABLE II
WPA VULNERABILITIES.

Vulnerability	Description
Key Management	As it has been stated, *WPA* standard works over a pre-shared and unique key. This situation implies the use of an external, secure and reliable communication channel for achieving the key distribution. Therefore, the *WEP* key management vulnerability is also present in the *WPA* standard.
Brute Force Attack	The 4-way handshake accomplished during the *WPA* association procedure is susceptible to being brute-force attacked. If the *WPA* 4-way handshake is captured, an attacker can perform a brute-force attack against the *PBKDF2* algorithm by using a huge key database. The attacker accesses the network when finding the pre-shared key, which is contained in the *PTK*. The success on the vulnerability's exploiting directly depends on the quality of the dictionaries and the weakness of the *WPA* key. Therefore, the use of secure and strong *WPA* passwords substantially reduces the penetration risks.

initially drawn that *WEP* is the standard that worst performs. Although *WEP* has one of the easiest implementation methods, using *WEP* as the wireless network security standard is not recommended since it has been here proved to be weak, easily violable, susceptible to several attacks and penetrable in short periods of time.

WPA presented a relatively low *WRV* and its implementation is as easy as *WEP's*. The standard was also proved to be susceptible to being attacked. However, its security is not susceptible to being penetrated whenever a strong *WPA* key is used. The *WPA's WRV* is, in fact reducible to 0 if the key length exceeds the 8 characters and uses a combination of alphanumerics and symbols.

WLAN 802.1X, on the other hand, was proved to be the most secure wireless network security standard since it presented no vulnerabilities, nor attack/penetration susceptibilities. However, the implementation complexity, represented in the highest *KL*, lead to think about the reasons that justify the huge costs on human resources. In fact, *WLAN* 802.1X may be implemented after accomplishing a security requirements analysis.

VI. CONCLUSION

The analysis over *WEP*, *WPA* and *WLAN* 802.1X provided an overview of the wireless network security standard's weaknesses and strengths.

WEP, for instance, initially aimed to provide a security level equivalent to wired channels and its implementation procedure is very easy to achieve. However, *WEP* was also proved to be weak, susceptible to several types of attacks and easily penetrable. Therefore, whenever the standard's main objective is to grant the wireless network information's integrity, confidentiality and availability, the *WEP* standard implementation is not recommendable.

The vulnerabilities analysis showed that *WPA* implementation complies with the security requirements that protects the data transmitted through the wireless network, whenever a strong key is used. *WPA* implementation is indeed highly recommendable as wireless security standard over home and small business *WLANs*.

The *WLAN* 802.1X standard presented the highest security level. Although its implementation can result complex and requires the network administrator's advanced expertise on servers installation and configuration, 802.1X provides the ideal architecture that satisfy the robust security requirements of medium and large business' wireless network. It also provides integral management since it works over an authentication, authorization and accounting –*AAA*– scheme.

REFERENCES

[1] E. Guillen, S. Loaiza, "Análisis de Vulnerabilidades de Tres Estándares de Seguridad para Redes de Área Local Inalámbricas," GISSIC Investigation Group, Telecommunications Engineering Department, Nueva Granada Military University, Bogotá-Colombia, 2008.
[2] Institute of Electrical and Electronics Engineers, "IEEE Standard for Information technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – Specific requirements – Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, 1999 Edition (R2003).
[3] C. Grogans, J. Bethea, I. Hamdan, "RC4 Encryption Algorithm," North Carolina Agricultural and Technical State University , March 5, 2000.
[4] Wi-Fi Alliance, "Wi-Fi Protected Access: Strong, standards-based, interoperable security for todays Wi-Fi networks," 2003.
[5] C. Rigney, S. Willens, Livingston, A. Rubens, Merit, W. Simpson, Daydreamer, "Remote Authentication Dial In User Service (RADIUS)," IETF RFC 2865, June 2000.
[6] J. Hassell, "RADIUS - Securing Public Access to Private Resources," O'Reilly & Associates, ISBN: 0596003226.
[7] K. Hamzeh, G. Pall, W. Verthein, J. Taaru, W. Little, G. Zorn, "Point-to-Point Tunneling Protocol (PPTP)," IETF RFC 2637, July 1999.
[8] D. Stanley, J. Walker, B. Aboba, "Extensible Authentication Protocol (EAP) Method Requirements for Wireless LANs," IETF RFC 4017, March 2005.
[9] Institute of Electrical and Electronics Engineers, "IEEE Standard for Information technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – Specific requirements – Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. Amendment 6: Medium Access Control (MAC) Security Enhancements", 2003.
[10] L, Barken, "How Secure Is Your Wireless Network? Safeguarding Your Wi-Fi LAN," Prentice Hall, ISBN: 0-13-140206-42003, pp. 224.
[11] T. Newsham, "Cracking WEP Keys. Applying known techniques to WEP keys," @Stake, 2001, pp. 35.
[12] T. Dierks, E. Rescorla, "The Transport Layer Security (TLS) Protocol: Version 1.2," IETF RFC 5246, August 2008.
[13] E. Rescorla, M. Ray, S. Dispensa, N. Oskov, "Transport Layer Security (TLS) Renegotiation Indication Extension," IETF RFC 5746, February 2010.
[14] E. Guillen, D. Padilla, Y. Colorado, "Weaknesses and Strengths Analysis over Network-based Intrusion Detection and Prevention Systems," Proceedings of the IEEE Latin-American Conference on Communications 2009, LATINCOM '09, pp. 1-5.
[15] E. Guillen, D. Padilla, K. Martinez, "Vulnerabilities and Performance Analysis over Fingerprint Recognition Systems," Proceedings of The 2010 World Congress in Computer Science, Computer Engineering, and Applied Computing, WORLDCOMP10.