Exposing WPA2 security protocol vulnerabilities

Achilleas Tsitroulis

Brunel University,
School of Engineering and Design,
Kingston Lane, Uxbridge, UB8 3PH, UK
Fax: +44 (0)1895-265580
E-mail: Achilleas.Tsitroulis2@brunel.ac.uk

Dimitris Lampoudis

Department of Information Systems,
University of Macedonia (U.O.M.),
Greece
Fax: +30-6983754201
E-mail: lampoudis.d@gmail.com

Emmanuel Tsekleves*

Lancaster University,
Imagination Lancaster,
LICA Building, Bailrigg,
Lancaster, LA1 4YW, UK
Fax: +44-(0)-1524-510794
E-mail: e.tsekleves@lancaster.ac.uk
*Corresponding author

Abstract: Wi-Fi protected access 2 (WPA2) is considered one of the most secure protocols employed in wireless local area networks (WLANs). This is despite of having significant security vulnerabilities. The aim of this paper is two-fold. Firstly it analyses the WPA2 security protocol and presents its weaknesses in detail. Secondly it presents a methodology that demonstrates how the WPA2 security protocol can be fully exposed by malicious attacks. Importantly, proposals on how to enhance its security are offered.

Keywords: wireless security; 802.11 security; Wi-Fi protected access 2; WPA2; Wi-Fi protected access; WPA; WPA2 vulnerabilities; privacy; authentication; wireless security enhancement.

Reference to this paper should be made as follows: Tsitroulis, A., Lampoudis, D. and Tsekleves, E. (2014) 'Exposing WPA2 security protocol vulnerabilities', *Int. J. Information and Computer Security*, Vol. 6, No. 1, pp.93–107.

Biographical notes: Achilleas Tsitroulis is currently working as a Webmaster, Digital Marketing and IT Systems Configuration, Infrastructure and Technician at Brunel University, London. He is an active researcher and his research interests are: IT security, networks, web platforms and cloud applications. He holds an MSc in Wireless Communication Systems and a BSc in Information Technology (IT).

Dimitris Lampoudis is working as a System Administrator and Web Developer at the Aristotle University of Thessaloniki. He holds an MSc in Information Systems and a BSc in Information Technology (IT). He is keen on researching and implementing innovative applications and programs that will effectively evolve the web and the IT security.

Emmanuel Tsekleves is a Senior Lecturer in Design Interactions at Lancaster University. He holds a PhD in Electronic and Computing Engineering. He is interested on digital economy and digital interactions.

1 Introduction

Today a plethora of digital devices are connected to wireless fidelity (Wi-Fi) networks allowing people to perform several tasks (ranging from browsing the internet to performing financial transactions). This often attracts individuals who try to steal information from users by attempting to bypass and break the Wi-Fi network security.

In contrast to wired networks, wireless networks, such as the Wi-Fi network lack in terms of security and suffer from several vulnerability issues. Within the context of computer security, the term 'vulnerability' refers to a security weakness that allows a malicious user to reduce the system's information assurance. Several wireless security protocols have been created in order to deal with the aforementioned issue. The wired equivalent privacy (WEP), Wi-Fi protected access (WPA) and Wi-Fi protected access 2 (WPA2) are popular security protocols used in Wi-Fi today. These do solve several security issues, however they are not free from security vulnerabilities. The best security protocol, so far is considered to be the WPA2 (Secure Wireless Access Point Configuration, 2006). In this paper, we test and analyse the WPA2 security protocol. WPA2 weaknesses are revealed and a smart and efficient way on how to expose several WPA2 security issues is presented.

The paper is organised into nine sections. After a brief introduction, an overview of the WPA and WPA2 security protocols is provided. This is followed by a discussion into related work from the literature. Section 4 presents the proposed methodology for exposing the WPA2 security protocol and Section 5 presents the results, which are then discussed in Section 6. Section 7 introduces a proposed scheme for enhancing network security and preventing malicious attacks. Section 8 presents the future work. Lastly, concluding comments are made in Section 9.

2 WPA and WPA2 security analysis

The WPA protocol uses either the temporal key integrity protocol (TKIP) or the more secure counter mode with cipher block chaining message authentication code protocol (CCMP). In contrast to WEP, WPA uses alphanumeric strings. WPA is implemented in two versions. The first edition (personal) uses a pre-shared key (PSK) and the second one uses an extensible authentication protocol (EAP) for authentication without the use of PSK. The WPA2 protocol has been developed for better future interoperability. Its biggest advantage is the encryption algorithm. In October 2000, the National Institute of Standards and Technology (NIST) adopted the Advanced Encryption Standard (AES) as a robust successor to the Data Encryption Standard (DES) (Shao et al., 2010; Ferguson et al., 2001). Similarly to the WPA and WEP protocols, WPA2 has two modes of security: 'home user/personal' and 'corporate/enterprise'. A pre-shared secret key is used in the 'home user' mode, which protects unauthorised network access by utilising a set-up password. The 'corporate' mode relies on 802.1X, EAP and a secure key distribution (Lashkari et al., 2009) and verifies network users through a server (Wi-Fi Alliance, 2005). WPA resolves the problem related with weak WEP headers, which are referred to as initialisation vectors (IV). It includes a method that ensures the integrity of messages passing through a message integrity check (MIC) (Glass and Muthukkumarasamy, 2005; Hytnen and Garcia, 2006) using TKIP to elevate data encryption. WPA has a special edition-mode, which does not include an enterprise authentication server and provides the same powerful encryption protection. WPA-PSK is a very powerful and strong encryption algorithm, in which the keys that are used for the encryption are automatically changed (known as rekeying) and are authenticated between devices after a specified period of time, or after a certain number of packets that have been transmitted (WPA Wireless Security for Home Network, 2006). WPA-PSK offers better security than WEP for two main reasons. Firstly, the process used to generate an encryption key is very robust and the change of the key is performed quickly. This can prevent even the most determined hacker from collecting enough data to break the encryption. Secondly, the WPA-PSK protocol implements a simple method for securing one's network. It employs a passphrase that must be filled in both the wireless access points (APs)/ modems-routers and the WPA clients. The passphrase can be between 8 to 63 characters including special characters and spaces. The WPA2 protocol implements the obligatory elements of 802.11i. More precisely, it introduces a new AES-based algorithm, the CCMP, which is considered very secure (Cam-Winget et al., 2003; Gin and Hunt, 2008). Government grade security is being provided through WPA2, by implementing the NIST, and Federal Information Processing Standards (FIPS) 140-2, both of which are compliant with the AES encryption algorithm. WPA2 is backward compatible with WPA. WPA usually employs TKIP and Rivest Cipher 4 (RC4), in contrast to WPA2 that uses CCMP and AES (Committee on National Security Systems, 2003; Biryukov and Khovratovich, 2009; Ferguson et al., 2001). In light of this, researchers are investigating the use of AES in Field Programmable Gate Array (FPGAs) (Deshpande et al., 2009; Granado-Criado et al., 2010) to enhance its performance and optimise it (Edney and Arbaugh, 2003; Gast, 2005; Gin and Hunt, 2008; Gold, 2011; Imai et al., 2005).

The strongest WPA2 encryption thus far is the 256 bit version. The length of the secret key can range from 8 to 63 printable ASCII characters (passphrase), or 64 hexadecimal digits.

3 Related work

In the past, several researchers have attempted to analyse the vulnerability issues related to WPA2 security. The WPA2 security protocol is vulnerable in Denial of Service (DoS) attacks (Bicakci and Tavli, 2009; Odhiambo et al., 2009). Re-authentication and re-association is an upcoming fact, making the protocol vulnerable in DoS attacks. Odhiambo et al. (2009) proposed an integrated security model (ISM), which incorporates a drop policy, aiming to enhance security against DoS attacks. In that model, a drop policy is used at the media access control (MAC) layer for the frames and dynamic virtual local area networks (VLAN), providing backward compatibility with the model of Robust Security Network (RSN) capable devices (Odhiambo et al., 2009). Also, Zhang and Sampalli (2010) have presented a novel client-based scheme aiming at preventing DoS intrusions. With the use of a MAC filtering mechanism, the proposed 'smart' client is able to differentiate between legitimate and forged management frames. This is a non-cryptographic mechanism and has low computational overheads (Zhang and Sampalli, 2010). Furthermore, in order to protect management frames, Malekzadeh et al. (2007) have proposed the use of a HMAC-SHA 1 algorithm. Provided the management frames are properly authenticated, the procedures of de-association, re-association, de-authentication and authentication flooding attacks can be mitigated (Sankar et al., 2006). In addition, a tool aiming at preventing DoS flooding has been proposed by Chibiao and Jame (2007). That tool uses a traffic pattern filtering procedure. Moreover, Changhua and Mitchel (2006) proposed that IEEE 802.1x/EAP authentication should be performed before association. This is an alteration to the IEEE 802.11i protocol execution (Odhiambo et al., 2009).

Dongsheng and Kai (2011) have proposed the concept of building separate and independent hash libraries (including possible passwords) based on the subject of the capable passphrases. They also suggested the use of cloud computing technology in order to build up hash libraries. Mavridis et al. (2011) offered an interesting approach on how to expose WPA2 security. However after the capture of the preshared key, they considered that the secret and unrevealed part of the password is a four digit number. This works, because they examined just one very specific infrastructure of their AP. In that infrastructure the password is: the AP's MAC address (which can be found easily through the sniffing process) followed by a '–' and then just a four digit number code (not even including lower/upper case letters, special characters or extended printable ASCII code characters). For that specific purpose it works well, however that method could not be extended to work for longer and more complex passwords.

Several different attempts have been made aiming to expose WPA2 security. Few of them succeeded in specific circumstances and infrastructures, but no one can guarantee the breaking of the WPA2 password in every different case. An alternative way is to generate a complete dictionary and use the appropriate method for checking and matching the password, in order to ensure that the password will be found. This paper employs and presents the aforementioned method.

4 Methodology

Several methods have been proposed in order to expose WPA2 security. Brute force attack is a popular method employed to find the network password key. This works by attempting to match the caught-instance of the WPA2 PSK with every record in a dictionary until the password key is found or the dictionary comes to an end. One popular program for this kind of attack is 'Aircrack'. Another way to attack WPA2 is by using the 'DoS' method, which aims at reducing the network performance or making it inaccessible. A common way to achieve that is to create traffic on the target network. In addition to this, the 'Man-in-the-middle attack-Evil Twin' forms another popular method. In this method, the attacker stands between two parties but (s)he is invisible or it is considered as a legal part of the network. In order to perform this type of attack special requirements and equipment are required (Liu et al., 2010; Maple, 2006).

In order to expose WPA2 security, deauthentication and Brute Force attacks have been selected for our proposed methodology. These kinds of attacks have been performed in Backtrack 4 operating system (OS) – Linux by using the Aircrack software suite and a custom program of ours creating the appropriate dictionary. Details can be found below.

4.1 Experimental network setup

The experimental network comprised of three computers using the 802.11g wireless protocol. 802.11g operates on a 2.4 GHz frequency. Station's 1 (STA1) and STA3's distance from the AP was approximately 10 metres (m) and STA2's distance was approximately 15m. All STAs were not on a straight angle with the AP, and walls were 'cut-in' between STAs and the AP.

During the experiments, one computer took the role of the adversary executing all the various attacks in a shell script environment. The others were the legal users. All the methods, experiments and attacks were made in a system that consisted of a wireless router (AP) and three wireless stations. STA1 used an AirPort Extreme Wireless Card, STA2 used an Intel PRO/Wireless 3945ABG wireless card and STA3 used an Atheros AR9285 wireless network adapter. The wireless local area network's (WLAN's) service set identifier (SSID) was 'AXILLEAS'. The network's internet protocol (IP) address-version 4- was: 192.168.178.0/24:

Table 1 Network addresses

	IP address	*MAC address*
AP	192.168.178.1/ 24	00:1C:4A:A3:55:4F
STA1	192.168.178.23/ 24	00:1C:B3:7C:7D:0F
STA2	192.168.178.29/ 24	00:1B:77:3C:56:E0
STA3	192.168.178.50/ 24	00:25:D3:67:79:4A

In the WPA and WPA2 configuration, the key was encrypted with TKIP and CCMP (using AES) respectively.

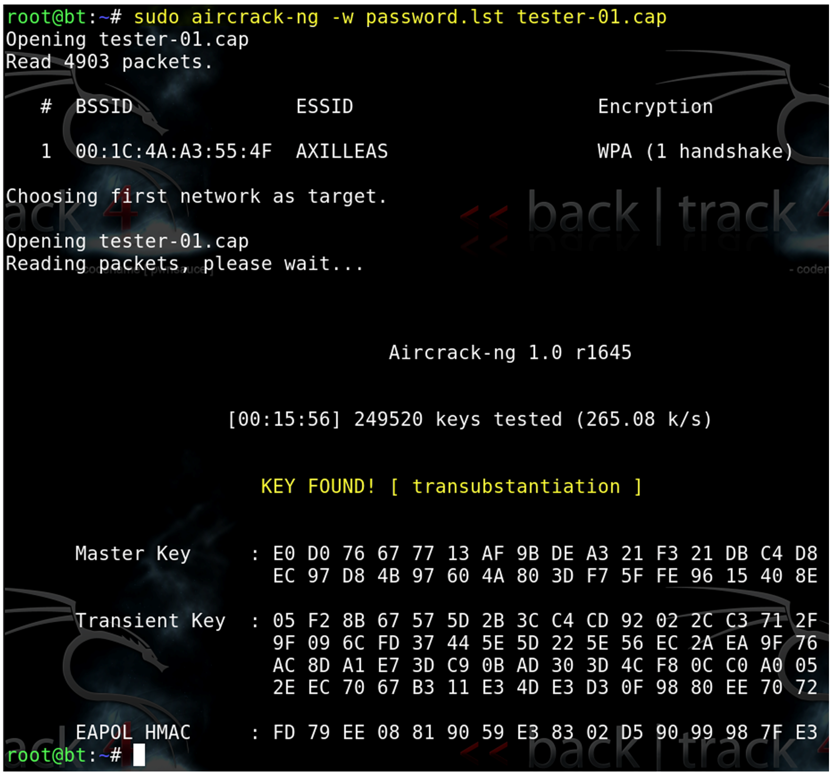
4.2 Exposing WPA2 security

Ten different scenarios have been examined. The main difference between them is the password that it was entered as it is shown at Table 2. The procedure followed in order to break the passwords of the ten different scenarios was the deauthentication and Brute Force attack. At the beginning, the area was scanned-sniffed with 'Airodump' and then a deauthentication attack was made with 'Aireplay'. Through that, an instance of the PSK was caught. Finally, 'Aircrack' was attempting to reveal the secret password by using the instance of the PSK and matching it with every record of the dictionary. For these experiments we used a very big dictionary that consisted of 666,696 standard printable ASCII character records of various lengths. 'Airodump' and 'Aireplay' are commands of the 'Aircrack' suite, responsible for sniffing and deauthentication respectively.

Table 2 Experimental cases

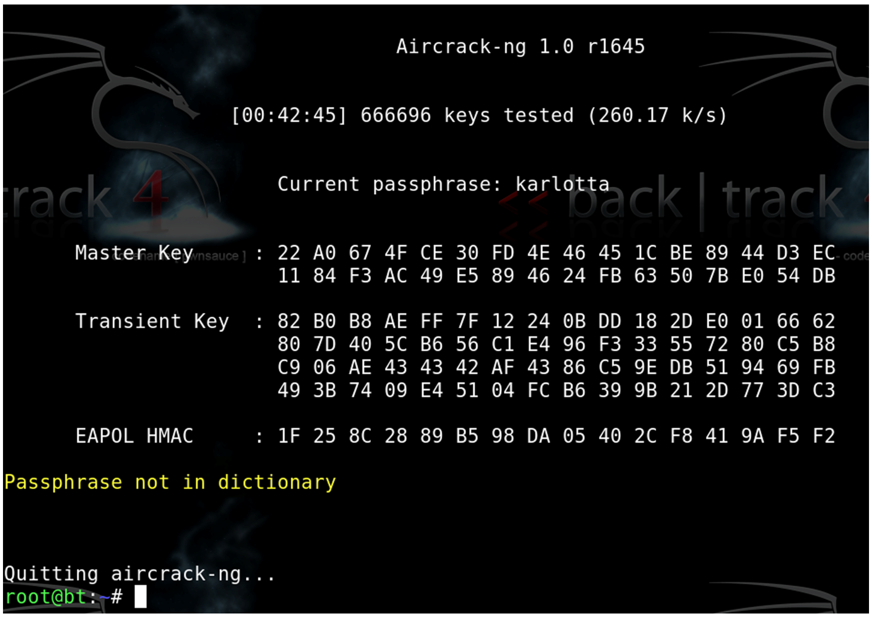
Case	*WPA2 key*	*Adversary*	*Legal user*
1	Icecream	STA2 operating on Backtrack 4 (BT 4)	STA1 operating on MAC OSX 10.6.2
2	transubstantiation	STA3 on BT 4	STA2 on Windows 7
3	SKy$kr@p3r!newy0rkc1ty%	STA3 on BT 4	STA2 on Windows 7
4	513ndA$*312n35Q#7Jjp3r5	STA3 on BT 4	STA2 on Windows 7
5	M0n601i4ni5m	STA3 on BT 4	STA2 on Windows 7
6	ArlEneseb@st!an	STA3 on BT 4	STA2 on Windows 7
7	arlI1IngtonHEIGHTS$9	STA3 on BT 4	STA2 on Windows 7
8	b01773121770m4n	STA3 on BT 4	STA2 on Windows 7
9	WwWbontokk@@@anka1290nayY%	STA3 on BT 4	STA2 on Windows 7
10	012bi70z960m47ic	STA3 on BT 4	STA2 on Windows 7

Figure 1 WPA2 key search results: key exposure (see online version for colours)



```
root@bt:~# sudo aircrack-ng -w password.lst tester-01.cap
Opening tester-01.cap
Read 4903 packets.

   #  BSSID              ESSID                     Encryption

   1  00:1C:4A:A3:55:4F  AXILLEAS                  WPA (1 handshake)

Choosing first network as target.

Opening tester-01.cap
Reading packets, please wait...


                                 Aircrack-ng 1.0 r1645


                   [00:15:56] 249520 keys tested (265.08 k/s)


                      KEY FOUND! [ transubstantiation ]


      Master Key     : E0 D0 76 67 77 13 AF 9B DE A3 21 F3 21 DB C4 D8
                       EC 97 D8 4B 97 60 4A 80 3D F7 5F FE 96 15 40 8E

      Transient Key  : 05 F2 8B 67 57 5D 2B 3C C4 CD 92 02 2C C3 71 2F
                       9F 09 6C FD 37 44 5E 5D 22 5E 56 EC 2A EA 9F 76
                       AC 8D A1 E7 3D C9 0B AD 30 3D 4C F8 0C C0 A0 05
                       2E EC 70 67 B3 11 E3 4D E3 D3 0F 98 80 EE 70 72

      EAPOL HMAC     : FD 79 EE 08 81 90 59 E3 83 02 D5 90 99 98 7F E3
root@bt:~# 
```

Figure 2 WPA2 key search results: key not found (see online version for colours)



```
                              Aircrack-ng 1.0 r1645


                   [00:42:45] 666696 keys tested (260.17 k/s)


                       Current passphrase: karlotta


      Master Key     : 22 A0 67 4F CE 30 FD 4E 46 45 1C BE 89 44 D3 EC
                       11 84 F3 AC 49 E5 89 46 24 FB 63 50 7B E0 54 DB

      Transient Key  : 82 B0 B8 AE FF 7F 12 24 0B DD 18 2D E0 01 66 62
                       80 7D 40 5C B6 56 C1 E4 96 F3 33 55 72 80 C5 B8
                       C9 06 AE 43 43 42 AF 43 86 C5 9E DB 51 94 69 FB
                       49 3B 74 09 E4 51 04 FC B6 39 9B 21 2D 77 3D C3

      EAPOL HMAC     : 1F 25 8C 28 89 B5 98 DA 05 40 2C F8 41 9A F5 F2

Passphrase not in dictionary



Quitting aircrack-ng...
root@bt:~#
```

5 Results

In some of the cases the key was very simple (case 1, 2), whereas in the other ones the key was too complex (case 3–10). As demonstrated, the key was found easily in all cases, with the exception of case 3. The results followed by measurements regarding the 'deauthentication' and 'finding the key' processes. Furthermore, there are two screenshots showing the success and the failure of finding the WPA2 key respectively.

Table 3 WPA2 deauthentication measurements

Case	*Data collected*	*Time needed for a WPA2 handshake*
1	22794	120 seconds
2	32561	180 seconds
3	14761	180 seconds
4	227	36 seconds
5	19832	60 seconds
6	31310	40 seconds
7	11839	20 seconds
8	17149	28 seconds
9	4329	120 seconds
10	7122	20 seconds

Table 4 WPA2 measurements on finding the key

Case	*Keys tested*	*Time needed, in order to find the key*
1	156	0 seconds
2	249,520	956 seconds
3	666,696	Failed in 2,565 seconds
4	77,772	52 seconds
5	193,332	129 seconds
6	317,556	153 seconds
7	317,612	215 seconds
8	335,372	225 seconds
9	367,152	249 seconds
10	704	0 seconds

6 Discussion

WPA/ WPA2 are considered amongst the most secure protocols. This is due to the fact, that even having an instance of the preshared key, it requires a dictionary attack to break it, which can last from a few minutes to several weeks, depending on the complexity of the key and the pluralism in words- records of the correspondent dictionary. The more complex the password is, the safer the network security will be. More precisely, words like: icecream, computer, clouds, wireless, mynet, airhouse, etc are commonly used, increasing the probability of finding the key in a short period of time. On the other hand, if the key consists of different types of characters (a combination of lower case, upper case, special characters and numbers) the complexity would be increased. Hence, the adversary must have a dictionary consisting of all the different combinations of all the printable ASCII characters of all the possible lengths, in order to ensure that (s)he will be able to find the secret key. In order to have a complete dictionary with all the different combinations of all the standard printable ASCII characters, the length (records) of the dictionary will be:

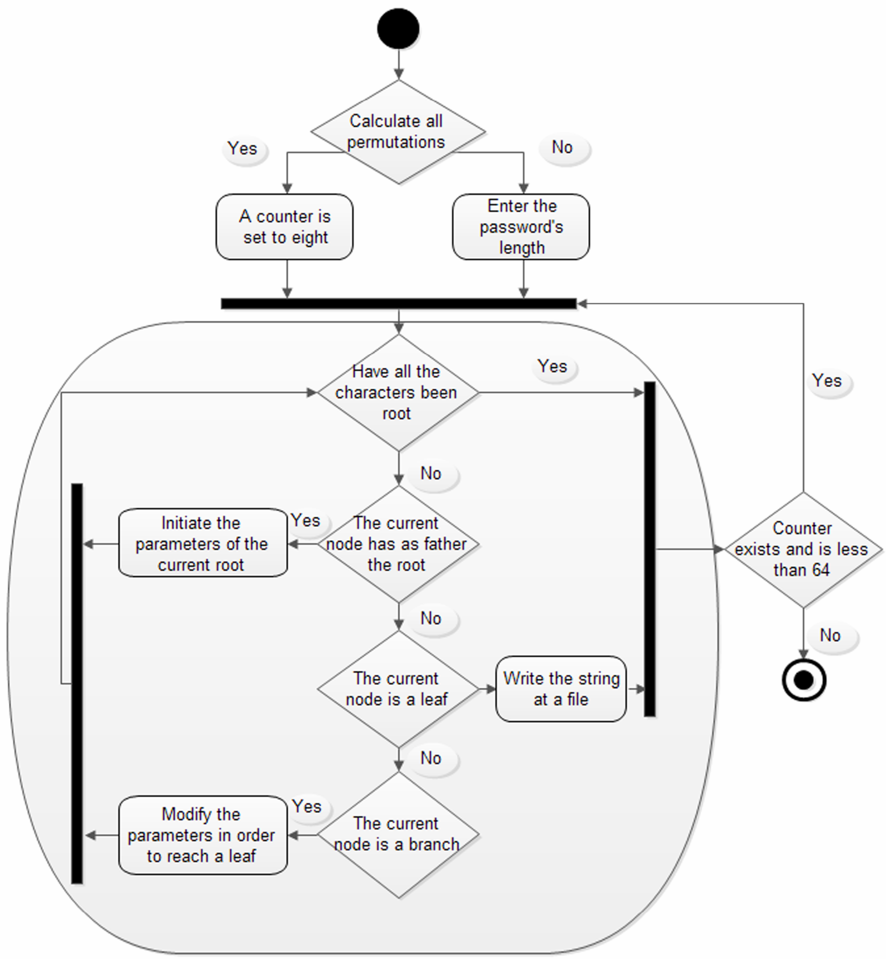
$$f(r) = \sum_{r=8}^{63} 95^r \tag{1}$$

In our case, we have '*permutations with repetitions*':

$$PR(n,r) = n^r \tag{2}$$

where 'n' is the number of objects to choose from and '*r*' is selected from them. Regarding WPA2, our objects (*n*) are 95, which are all the standard printable ASCII characters (ASCII Table and Description-Extended ASCII codes, 2012) and '*r*' represents the current length of the password. By performing the calculations the complete dictionary would consist of 3.991929703310227E124 records. Thus, this procedure (that creates and searches the dictionary) will last several weeks using a simple computer, due to the required time which will be extremely high. At this point, it is obvious that a lot of time will be required to create a dictionary which will include all the different printable ASCII character sets of all the possible lengths. To achieve this in a shorter time, it is advisable to use a supercomputer or a computer cluster. An alternative to that would be to use an FPGA. In our experiment, case 3 failed due to the great length of time taken to find the secret key, resulting in the process being aborted in the end. The reason behind this was that password that was used was too complex and the correspondent dictionary did not have that word as a record. In case 9, which was as complex as case 3, the password existed as a record at the correspondent dictionary, resulting in the successful finding of the key. In light of this, it is clear that the successful finding of the key relies on the dictionary records. If the key is part of the dictionary then the procedure will be successful.

Figure 3 Activity diagram of the dictionary program


Calculate all permutations
Yes
No
A counter is set to eight
Enter the password's length
Have all the characters been root
Yes
Yes
No
Initiate the parameters of the current root
Yes
The current node has as father the root
Counter exists and is less than 64
No
No
The current node is a leaf
Write the string at a file
No
Modify the parameters in order to reach a leaf
Yes
The current node is a branch

In order to find a WPA/WPA2 password with absolute success, we have created a Java program, which creates a dictionary comprised of all the different combinations of all the standard printable ASCII characters (95) of all the different and possible password lengths. Thus, the program creates all the possible passwords of all the possible lengths.

In addition, the program provides the extra option of creating a dictionary comprised of all the different combinations of all the standard printable ASCII characters (95) for a fixed password length, given by the adversary. In that option, the adversary defines (by assuming) the length of the secret password and the program creates all the possible passwords based on the length given by the adversary. That is very useful in terms of speed, if the adversary can guess or ascertain the password length. Furthermore, the program user can incorporate the desirable extended printable ASCII characters (ASCII Table) to the program in order to create letters/ words from other languages like Greek, Japanese etc. A unified modelling language (UML) diagram is illustrated in Figure 3 in order to explain how that program works and how to create a similar program. Thus, if the adversary can create a dictionary of all the possible printable ASCII characters of all possible lengths, (s)he will be able to expose the network security.

7 Proposed scheme

From the above discussion, it is clear that WLAN is vulnerable and it can be exposed to malicious attacks. These attacks could be prevented by adopting the following countermeasures. Firstly, network security can be increased by firstly hiding the SSID, so that the procedure of gathering information regarding the network becomes more difficult. Furthermore, in some APs the Telnet/SSH services are enabled by default. It is advisable to disabling those services in order to protect unauthorised network access, by providing password checks. Not following the above actions, increases the risk of unauthorised network access that can lead to various malicious actions, such as having the AP reconfigured by the adversary. Moreover, specific brands and models of APs provide the administrator with the possibility to add Shell Script code. Using shell script code can prevent some attacks. For instance, a set of maximum tries can be defined, in which a supplicant can try to connect. If that number is exceeded, the AP will ban the adversary from retrying, for a certain period of time. In addition, a Proxy server should be established on an independent computer. That proxy can monitor almost all networks. For instance, it can prevent MAC substitution attacks by checking the multiplicity of the same MAC addresses. Furthermore, protection from malicious software and denial of undesirable connections from the 'outside' can be provided. Also, the DHCP mode must be disabled and the static mode must be enabled. In that case, MAC filtering attacks can be limited. Lastly, this can be protected by limiting the available set of IPs and by managing the subnet mask.

Another good idea would be the use of a 'honeypot'. Honeypot is a network trap, or a special constructed computer designed to attract and detect malicious attacks. In this case an additional computer can act like a honeypot. Its job is to get deliberately infected and enter the botnet. As long as, the honeypot is not detected it can collect information regarding the adversary in order to inform the authorised system and network administrator of the upcoming danger. Enhancing that idea, a distributed honeypot network that can accurately emulate the network traffic coming in, from the internet should be established. Thus, security defenders can take proactive actions in directing these kinds of attacks (Wang et al., 2010; Valli, 2007). The above method can be easily combined with a modified, for an AP network infrastructure, reputation-based clustering algorithm (RECA) that can be used for security management (Elhdhili et al., 2009). Furthermore, the 'weighted trust evaluation (WTE)' algorithm that Hu et al. (2009)

proposed can be used. WTE detects compromised nodes by monitoring the data that the nodes reports. The algorithm basis is a hierarchical topology network for reducing the communication overhead among the network. WTE algorithm models a cluster of sensor nodes (SN) under the control of a forwarding node (FN) and it detects malicious nodes by examining their weights. Those weights are being assigned to SNs, representing the reliabilities of SNs. In terms of both scalability and robustness tests, the misdetection ratios could be largely reduced by introducing the weight recover mechanism (Hu et al., 2009).

An innovative ontology-based approach can be also employed in order to react to network attacks. The Reaction after Detection (ReD) enhances the detection and reaction process and improves the overall resilience of IP networks to attacks. It also helps telecommunication and service providers to maintain sufficient quality of service and respect service level agreements. That technology provides a way to map alerts into an attack context, which can be used to identify the policies to be applied in the network to solve the various threats. Ontologies which are describing alerts and policies can thus be defined. These ontologies use inference rules to perform such mappings. The ReD architecture contains the following elements: the alert correlation engine (ACE), policy instantiation engine (PIE), policy decision point (PDP), reaction decision point (RDP) and policy enforcement point/reaction enforcement point (PEP/REP) (Cuppens-Boulahia et al., 2009). Also, the first trial protocol (FTR) that Sodiya et al. (2011) introduced, can be modified and used for the WLAN case. FTR was created to prevent dictionary and brute force attacks. The FTR protocol uses a rule-based reasoning and then splits the authentication procedure into the first and second protocol layers. The first layer undertakes the validation of the login password against set of recorded invalid passwords in the first layer repository. The second level of authentication, which is the second layer, is located on another host, different from the first layer, containing the protocol and its penalties (Sodiya et al., 2011). Finally, in order to increase the security against DoS/distributed DoS (DDoS) attacks, it is advisable to use the identity-based privacy-protected access control filter (IPACF) protocol that Wu et al. (2009) proposed. In the case, the user and the responder must always authenticate each other. In every frame, the value and the identity for authentication are being changed. Following this ensures that privacy is being protected for both the user and server (Wu et al., 2009).

8 Further work

Further work can be done on eliminating or preventing deauthentication attacks. A proposed area of research investigation is to develop a hardware chip embedded on wireless cards to prevent the reception of deauthentication packages. In addition, an automated software that would dynamically change the beacon interval time depending on the network's demands could be researched. Increasing the beacon interval time, when network conditions allow that, will have as a result the increase of difficulty on finding the WLAN, during the sniffing process by an adversary. Research in that area should also focus on the creation of a method that will decrease the power consumption of the legal clients constituting the network, when they 'wake up' from sleep mode and search the correspondent AP. Further work, could be done with regards to the section range. Software can be created in order to, dynamically change the transmission range of the

AP, depending on the true distances of the clients constituting the WLAN. This will aim at preventing long distance adversaries from finding the network. Also, further work could be performed in resolving the 'WPA2 hole 196 vulnerability', as it was presented at the 'black hat (Arsenal)' technical security conference, on the 29th of July, 2010 (AirTight networks). The 'Hole 196' exposes WPA2-secured 802.11i networks on attacks made from the 'inside'. In that case a multi-layered wireless security approach must be performed. In addition, research investment should be made in programming and configuring a Wireless Intrusion Prevention System (WIPS), protecting the WLAN from wireless threads such as Rogue APs, inappropriate behaviour of 802.11i clients, inappropriate configurations regarding the WLAN infrastructure, vulnerabilities and security leaks according to 802.11i security protocols (AirTight networks; Bradbury, 2011). Regarding the dictionary program, this can be expanded by creating passwords according to specific categories such as nature, cinema, music, sports etc., so, the dictionary is filled only with words of the selected categories. This option can be useful, time wise, if the adversary thinks the WPA2 password will be part of a specific category. Lastly, the procedure of creating dictionary words, based on a fixed length, category etc and matching those words with the instance of the PSK can be implemented in a cloud system in order to accelerate the whole procedure and make it accessible to everyone. Such cloud system will be comprised of FPGAs and distributed-computers.

9 Conclusions

Thus far, WPA2 is considered to be amongst the most secure protocols. However it has several security vulnerabilities. Until now there has not been a complete and fully successful methodology capable of exposing the WPA2 security. This paper provides a novel way of successfully exposing WPA2 security issues by using a complete dictionary that generates all the possible printable ASCII characters of all possible lengths.

The 802.11i deauthentication process presents a 'flaw' on its security system. During that process, clients (STAs) are forced to reconnect and re-authenticate to the correspondent AP, having as a result the capture of an instance of the preshared key. In the case of WPA/WPA2, during the deauthentication process the four way authentication handshake is revealed. The most secure protocols so far are the WPA/WPA2 ones. This is due to the fact that, even having an instance of the preshared key, one would require a dictionary attack, which would last from a few minutes to several weeks depending on the complexity of the key and the pluralism of words- records in the dictionary. The biggest advantage of WPA/WPA2 security protocols is security reliance on dictionary pluralism in words. As mentioned above, it is very difficult to expose the WPA/WPA2 security protocol, but not impossible. Even though, a considerable amount of time would be required. In order to accomplish that, in a relatively short period of time, the adversary should have a FPGA (instead of a computer), performing the whole procedure.

The most efficient way to protect an 802.11i network is the use of WPA2, as the wireless security protocol, with a combination of MAC filtering. 256 bit is the strongest encryption so far. In addition, there is the possibility of changing the key periodically, which will increase the difficulty in adversaries' attempts. It is indeed evident, that the aforementioned suggestions are not sufficient on their own, in achieving a desirable security standard. Hence, the adoption of the specific countermeasures-proposed in the ‘proposed scheme’ chapter, can improve the security standard of the network and make adversary hacking attempts more difficult.

References

AirTight networks [online] http://www.airtightnetworks.com/WPA2-Hole196 (accessed 13 October 2012).

ASCII Table and Description, Extended ASCII codes [online] http://www.asciitable.com/ (accessed 13 October 2012).

Bicakci, K. and Tavli, B. (2009) ‘Denial-Of-Service attacks and countermeasures in IEEE 802.11 wireless networks’, *Computer Standards & Interfaces*, Vol. 31, No. 5, pp.931–941.

Biryukov, A. and Khovratovich, D. (2009) ‘Related-key cryptanalysis of the full AES-192 and AES-256’, *Proceedings of the 15th International Conference on the Theory and Application of Cryptology and Information Security: Advances in Cryptology*, pp.1–18, Springer, Berlin.

Bradbury, D. (2011) ‘Hacking Wi-Fi the easy way’, *Network Security* [online] http://www.sciencedirect.com/science/article/pii/S1353485811700149 (accessed 13 October 2012).

Cam-Winget, N., Housley, R., Wagner, D. and Walker J. (2003) ‘Security flaws in 802.11 data link protocols’, *Communications of the ACM – Wireless Networking Security*, May, pp.35–39.

Changhua, H. and Mitchel, J. (2006) *Security Analysis and Improvements for IEEE 802.lli* [online] http://theory.stanford.edu/~jcm/papers/NDSS05.pdf (accessed 13 October 2012).

Chibiao, L. and Jame, Y. (2007) ‘A solution to WLAN Authentication and Association Attacks’, *International Journal of Computer Science*, Vol. 34, No. 1, pp.1–6.

Committee on National Security Systems (2003) *National Policy on the Use of the Advanced Encryption Standard (AES) to Protect National Security Systems and National Security Information* [offline] http://www.cnss.gov/Assets/pdf/cnssp_15_fs.pdf (accessed 13 October 2012).

Cuppens-Boulahia, N., Cuppens, F., Autrel, F. and Debar, H. (2009) ‘An ontology-based approach to react to network attacks’, *Int. J. of Information and Computer Security*, Vol. 3, Nos. 3/4, pp.280–305.

Deshpande, A.M., Deshpande, M.S. and Kayatanavar, D.N. (2009) ‘FPGA implementation of AES encryption and decryption’, in *INCACEC 2009: Control, Automation, Communication and Energy Conservation*, pp.1–6, IEEE Computer Society Washington, DC, USA.

Dongsheng, Y. and Kai, C. (2011) ‘A research into the latent danger of WLAN’, *ICCSE 2011: Proceedings of the 6th International Conference on Computer Science & Education*, pp.1085–1090, IEEE Computer Society Washington, DC, USA.

Edney, J. and Arbaugh, W.A. (2003) *Real 802.11 Security Wi-Fi Protected Access and 802.11i*, Addison-Wesley Professional, Boston, USA.

Elhdhili, M.E., Azzouz, L.B. and Kamoun, F. (2009) ‘REputation based clustering algorithm for security management in ad hoc networks with liars’, *Int. J. of Information and Computer Security*, Vol. 3, Nos. 3/4, pp.228–244.

Ferguson, N., Schroeppel, R. and Whiting, D. (2001) ‘A simple algebraic representation of Rijndael’, *8th Annual International Workshop, Selected Areas in Cryptography*, pp.103–111, Springer, Toronto.

Gast, M.S. (2005) *802.11 Wireless Networks: The Definitive Guide*, 2nd ed., O’Reilly Media, California.

Gin, A. and Hunt, R. (2008) ‘Performance analysis of evolving wireless IEEE 802.11 security architectures’, *Mobility ‘08 Proceedings of the International Conference on Mobile Technology, Applications, and Systems*, ACM New York, NY, USA.

Glass, S. and Muthukkumarasamy, V. (2005) 'A study of the TKIP cryptographic DoS attack', *ICON 2007: Proceedings of the 15th IEEE International Conference on Networks*, pp.59–65, IEEE Computer Society Washington, DC, USA.

Gold, S. (2011) 'Cracking wireless networks', *Network Security*, Vol. 2011, No. 11, pp.14–18.

Granado-Criado, J.M., Vega-Rodriguez, M.A., Sanchez-Perez, J.M. and Gomez-Pulido J.A. (2010) 'A new methodology to implement the AES algorithm using partial and dynamic reconfiguration', *Integration, the VLSI Journal*, Vol. 43, No. 1, pp.72–80.

Hu, H., Chen, Y., Ku, W., Su, Z. and Chen, C. (2009) 'Weighted trust evaluation-based malicious node detection for wireless sensor networks', *Int. J. of Information and Computer Security*, Vol. 3, No. 2, pp.132–149.

Hytnen, R. and Garcia, M. (2006) 'An analysis of wireless security', *Journal of Computing Sciences in College*, Vol. 21, No. 4, pp.210–216.

Imai, H., Mohammad, G.R. and Kazukuni, K. (2005) *Wireless Communications Security*, Artech House, London, UK.

Lashkari, A.H., Samadi, B. and Danesh, M.M.S. (2009) 'Wireless security protocols (WEP, WPA and WPA2/802.11i)', *ICCSN '10: Proceedings of the 2nd IEEE International Conference on Computer Science and Information Technology*, pp.48–52, IEEE Computer Society Washington, DC, USA.

Liu, Y., Jin, Z. and Wang, Y. (2010) 'Survey on security scheme and attacking methods of WPA/WPA2', *WiCOM 2010: Proceedings of the 6th International Conference on Wireless Communications Networking and Mobile Computing*, pp.1–4, IEEE Computer Society Washington, DC, USA.

Malekzadeh, M., Azim, A., Ghani, A., Zulkarnain, Z.A. and Muda, Z. (2007) 'Security Improvement for Management frames in IEEE 802.11 wireless networks', *International Journal of Computer Science and Network Security*, Vol. 7, No. 6, pp.276–284.

Maple, C. (2006) 'Choosing the right wireless LAN security protocol for the home and business user', *ARES '06: Proceedings of the First International Conference on Availability, Reliability and Security*, pp.1025–1032, IEEE Computer Society Washington, DC, USA.

Mavridis, I.P., Androulakis, A.I.E., Halkias, A.B. and Mylonas, P. (2011) 'Real-life paradigms of wireless network security attacks', *PCI 2011: Proceedings of the 15th Panhellenic Conference on Informatics*, pp.112–116, IEEE Computer Society Washington, DC, USA.

Odhiambo, O.N., Biermann, E. and Noel, G. (2009) 'An integrated security model for WLAN', *AFRICON, 2009: AFRICON '09*, pp.1–6, IEEE Computer Society Washington, DC, USA.

Sankar, K., Sri, S., Balinsky, A. and Miller, D. (2006) *Cisco Wireless LAN Security*, Cisco Press, Indiana.

Secure Wireless Access Point Configuration (2006) [online] http://technet.microsoft.com/en-us/library/cc875845.aspx (accessed 13 October 2012).

Shao, F., Chang, Z. and Zhang, Y. (2010) 'AES encryption algorithm based on the high performance computing of GPU', *ICCSN '10: Proceedings of the Second International Conference Communication on Software and Networks*, pp.588–590, IEEE Computer Society Washington, DC, USA.

Sodiya, A., Afolorunso, A.A. and Ogunderu, O. (2011) 'A countermeasure algorithm for password guessing attacks', *Int. J. of Information and Computer Security*, Vol. 4, No. 4, pp.345–364.

Valli, C. (2007) 'Honeypot technologies and their applicability as a strategic internal countermeasure', *Int. J. of Information and Computer Security*, Vol. 1, No. 4, pp.430–436.

Wang, P., Wu, L., Cunningham, R. and Zou, C. (2010) 'Honeypot detection in advanced botnet attacks', *Int. J. of Information and Computer Security*, Vol. 4, No. 1, pp.30–51.

Wi-Fi Alliance (2005) *Deploying Wi-Fi Protected Access (WPATM) and WPA2TM in the Enterprise* [online] http://www.wi-fi.org/files/wp_9_WPA-WPA2%20Implementation_2-27-05.pdf (accessed 13 October 2012).

WPA Wireless Security for Home Network (2006) [online] http://www.microsoft.com/windowsxp/using/networking/expert/bowman_03july28.mspx (accessed 25 October 2012).

Wu, C., Liu, T., Huang, C. and Irwin, J.D. (2009) 'Modelling and simulations for identity-based privacy-protected access control filter (IPACF) capability to resist massive denial of service attacks', *Int. J. of Information and Computer Security*, Vol. 3, No. 2, pp.195–223.

Zhang, Y. and Sampalli, S. (2010) 'Client-based intrusion prevention system for 802.11 wireless LANs', *WiMob2010: Proceedings of the 6th International Conference IEEE 2010 on Wireless and Mobile Computing, Networking and Communications*, pp.100–107, IEEE Computer Society Washington, DC, USA.