

A Study of the TKIP Cryptographic DoS Attack

Stephen Glass¹, Vallipuram Muthukumarasamy²

¹Queensland Research Lab, NICTA
stephen.glass@nicta.com.au

²Queensland Research Lab, NICTA and Griffith University
v.muthu@griffith.edu.au

Abstract—Wireless networks, especially those based on 802.11, have found widespread use in domestic, commercial, educational, military and public-safety environments. The security of these wireless networks is assuming an increasing importance as users come to rely on the availability and correct functioning of wireless network services.

This paper investigates the cryptographic denial-of-service (DoS) attack against the 802.11i TKIP security protocol. We have conducted a laboratory study and show that it takes very little effort to bring TKIP-protected network traffic to a complete halt. This attack maybe used not just to compromise availability but is also an effective means of conducting a security-level rollback to the insecure WEP protocol. We use a testbed network to evaluate a remedial measure that eliminates the vulnerability on which the attack is based.

I. INTRODUCTION

The early years of the 802.11 standard were dogged by the discovery of serious security flaws in the WEP security protocol most of which have been addressed by the TKIP and CCMP security protocols[1]. Nevertheless, there are still no safeguards as to the availability of 802.11 wireless networks. *Denial-of-Service* (DoS) vulnerabilities are present at all levels of the 802.11 stack and many can be exploited easily by a malicious adversary. This paper investigates a potentially serious DoS vulnerability that has been introduced by the TKIP protocol.

A. Cryptographic vulnerability

TKIP has been designed to allow older hardware to be used in a *Robust Security Network* (RSN) with only software or firmware modifications. It makes use of the RC4 stream cipher to protect confidentiality and a new *Message Integrity Code* (MIC) named Michael[2] to ensure authenticity and integrity. Michael is, however, known to be cryptographically weak and suffers from two major security flaws that can lead to key recovery.

The first of these flaws is that Michael is invertible; given a plaintext frame and its corresponding MIC it is a trivial task to recover the MIC key[3]. To protect the key, 802.11 requires that MIC values are never transmitted in the plain, but are instead encrypted along with the data frame's payload as shown in figure 1. The second security flaw is that Michael offers no serious defences against active key-recovery attacks.

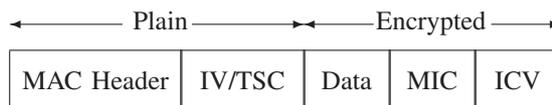


Fig. 1. TKIP Data Frame

To protect the MIC key against this threat TKIP relies on “countermeasures” that are intended to alert the administrator and slow down the attacker. The countermeasures mandated by the standard require that all TKIP operations must cease for one minute followed by re-negotiation of the group and all pairwise session keys[4, pages 49–51]. This represents a potent DoS vulnerability: a hostile adversary can deliberately provoke TKIP countermeasures to bring all TKIP-protected traffic to a complete halt.

B. Exploiting the vulnerability

TKIP countermeasures are triggered by the arrival of two frames with incorrect MICs within one minute of each other. To prevent trivial attacks TKIP requires that both the *Integrity Check Value* (ICV) and *TKIP Sequence Counter* (TSC) must be correct or the frame will be discarded without checking the MIC. The ICV is a CRC32 checksum of the frame's plaintext data and is transmitted as part of the frame's encrypted payload. The TSC is the replay protection mechanism. Every frame is assigned a unique TSC value which contributes to the per-packet encryption key.

Assuming the adversary to be an external attacker and not in possession of any encryption keys the necessary and sufficient conditions for a successful cryptographic DoS attack are:

- Intercepting a TKIP-encrypted frame before delivery.
- Modifying the frame's encrypted payload such that it invalidates the MIC but cannot be detected by the ICV.
- Forwarding the modified frame.
- Ensuring that two such modified frames reach the intended destination within one minute of each other.

These conditions can be met by the use of the message-modification attack[5] and simply re-transmitting the modified frame. The message modification attack exploits a security flaw inherited from WEP that enables encrypted data to be modified and a compensating change made to the encrypted ICV.

C. Message modification attack

The message modification attack of Goldberg, Borisov and Wagner exploits the linear structure of CRCs and stream ciphers to change a WEP or TKIP-encrypted message without detection by the ICV. Unfortunately, the discussion is incorrect in a minor detail that renders it ineffective against 802.11. We believe that this is the first time the problem has been identified[6] and so the correct attack is produced in full here. The object of the attack is a TKIP-protected message that has been intercepted before it can reach the intended destination:

$$A \rightarrow (B) : (TSC, C) \quad (1)$$

The intercepted message contains the TSC along with the ciphertext C .

$$C = RC4(k) \oplus (M || CRC32(M)) \quad (2)$$

TKIP derives the encryption key differently from WEP and in this case the per-packet encryption key k is derived from the TKIP key-mixing function. To compute the new ciphertext C' the attacker constructs a bit string Δ representing the bits to change and computes its checksum $CRC(\Delta)$.

$$C' = C \oplus (\Delta || CRC(\Delta)) \quad (3)$$

Note that it is the $CRC(x)$ function that is used to compute the checksum for Δ . This is the normal CRC algorithm (division by a specified polynomial over $GF(2)$) and not the CRC32 function. This is necessary because $CRC32$ is such that:

$$CRC32(x) \oplus CRC(y) = CRC32(x \oplus y). \quad (4)$$

The modified message can then be forwarded to its intended destination with a spoofed source address.

$$(A) \rightarrow B : (TSC, C') \quad (5)$$

D. Cryptographic DoS attack

The cryptographic DoS attack is a simple extension to the message modification attack. All that is required is for the hostile adversary to send a second copy of the modified message.

$$(A) \rightarrow B : (TSC, C') \quad (6)$$

This evades the TKIP replay protection mechanism because TKIP updates the TSC only after a frame passes the ICV, TSC and MIC tests. On receipt of this second message TKIP countermeasures should begin.

E. Outline of the paper

The rest of this paper discusses our experience of the cryptographic DoS attack gained from a laboratory study. The next section discusses related work. In section III the testbed environment and attack implementation are described. Section IV presents the experiments and results with an analysis in section V. Finally, the conclusions are presented in section VI.

II. RELATED WORK

MAC layer security flaws are responsible for a number of wireless network DoS vulnerabilities. At the most basic level any unfairness in the MAC layer can be exploited by a greedy station to obtain unfair access to the channel. Combined with traffic flooding at the application layer this can be highly effective as a DoS attack. The remedial measure lies in the design and use of fair MAC protocols that detect and punish greedy behaviour[7].

The *virtual jamming* attack exploits the virtual carrier sense mechanism used to coordinate access to the shared radio channel. A malicious adversary can repeatedly inject *request-to-send* (RTS) and *clear-to-send* (CTS) control frames to prevent other stations from transmitting[8]. NAV validation has been suggested by several authors as a means of addressing this threat[9][10]. NAV validation monitors the channel to ensure that the RTS/CTS request is followed by the appropriate transmission or the original request is cancelled.

Identity attacks exploit the implicit trust that 802.11 networks place in the source address of transmitted frames. An adversary can misrepresent the source of a message. In 802.11 management frames are authenticated only by their source address and Bellardo and Savage describe how deauthentication and disassociation management frames can be spoofed to effect a DoS attack. Even in 802.11i these management frames remain unauthenticated and there are several new opportunities to conduct identity attacks to deny availability[11]. 802.11 TGw is pursuing the approach of authenticating management frames although authenticity cannot be guaranteed for all frames throughout an association. An alternative approach to prevent this class of DoS attacks has been to propose the use of a central manager to explicitly manage the association, authentication and disassociation process[12].

An identity vulnerability is also present in the original 802.11i four-way handshake used to establish session keys. He and Mitchell were able to show that if an attacker could spoof the first message in the 4-way handshake then subsequent key negotiation would fail and repeatedly spoofing frames presents a DoS vulnerability[13]. A similar attack prevents successful completion of the 4-way handshake by injecting frames with bad *Information Elements* (IEs).

He and Mitchell also discuss the cryptographic vulnerability and TKIP countermeasures in their comprehensive 802.11 security analysis paper[14]. To prevent the adversary triggering countermeasures by simply re-injecting the damaged frame this paper proposes a change to the TSC update policy. Incrementing the TSC after the ICV check would require the adversary to intercept and damage multiple frames although as a defensive measure this is extremely weak. Another alternative countermeasures was considered by the 802.11 TGi working group. In a presentation by Harkins he argues that incurring a sixty second shutdown and key re-negotiation for all stations is unnecessary. Instead, this proposal argues that only the secure association under attack should be subject to a small delay and key re-negotiation[15].

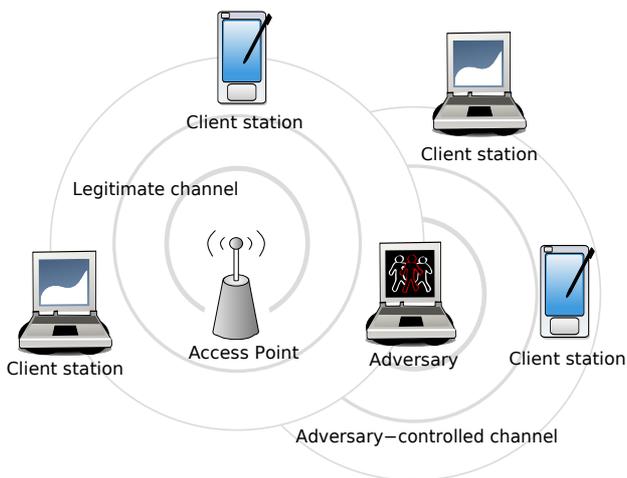


Fig. 2. Middleperson attack

III. EQUIPMENT AND PREPARATION

This section describes the laboratory setup and the actual implementation of the cryptographic DoS attack.

A. Testbed wireless network

An infrastructure IEEE 802.11 wireless network is used for our experiments. Client stations are computers running the Windows XP (Service Pack 2), MacOS X, FreeBSD and GNU/Linux operating systems using a variety of wireless network interfaces and access points. The principal access point used is the HostAP - a software implementation that runs on a conventional GNU/Linux PC equipped a wireless network interface. The use of a general purpose PC for this role allows for easier debugging, for the the TKIP countermeasures to be changed and a packet sniffer used to capture traffic after it has been decrypted by the wireless network interface. The `ping` and `iperf` programs are used to generate data traffic between stations and collect throughput statistics.

B. Attack implementation

The precondition for a successful cryptographic DoS attack is that a message can be intercepted in flight and a modified copy substituted in its place. To meet this requirement the *middleperson* or *man-in-the-middle* attack is implemented as suggested by Edney and Arbaugh[16, pages 334–335].

A GNU/Linux laptop computer equipped with two wireless network interfaces is used to conduct the attack. One interface operates on the radio channel of the legitimate network and the other a channel of the attacker’s choosing and traffic is forwarded between the channels as illustrated in figure 2. Once a station joins the network via the adversary-controlled channel it is possible for the adversary to insert, modify, re-order, delay and delete traffic to/from that station. The advantage of the two-channel approach is that it effectively eliminates the possibility that the captive stations and legitimate BSS communicate with each other without the active involvement of the adversary.

A user-mode C++ program is used to forward frames between interfaces. Not all frames need to be forwarded — control frames such as RTS/CTS and ACK and frames to/from stations not participating in the experimental network should not be forwarded. A Berkley Packey Filter program is installed in the kernel and used to discard these frames without the intervention of the user-mode frame-relay program. When a frame is received it is subjected to a series of rule-based editors that may modify, delete or insert frames into the transmit queues. These editors are responsible for frame forwarding, implementing the channel-change, deauthentication and cryptographic DoS attacks and re-writing beacons and probe responses. The cryptographic DoS editor, for example, relies on identifying TKIP data frames and using the message modification attack against them. TKIP frames containing a MIC are identified simply by consulting the “protected frame”, “extended IV” and “more data” fields of the MAC header.

The MadWifi device driver used by the attacker allows for the device MAC address to be changed, for promiscuous reception and the injection of raw frames (allowing source addresses to be spoofed). Nevertheless, a problem arises because of the strict timing constraints for acknowledging frame reception cannot be met by simply relaying acknowledgments between the interfaces. As illustrated in figure 3 it is not possible to relay the ACK within the required time limit even under optimal conditions. Such an approach would incur at least one retransmission for every frame and this could expose the adversary’s presence prior to the launching of an attack.

To meet the timing constraints the attacker must acknowledge the receipt of frames relayed to or from the captive stations. For the interface operating on the attacker-controlled channel simply changing the device MAC address to the BSSID is sufficient to generate these acknowledgments. The other interface is more difficult because it must acknowledge frames sent to any MAC address present on the adversary-controlled channel. To meet this requirement the device driver has been modified to implement a promiscuous acknowledgment mode. In this mode the interface will acknowledge all frames whose MAC addresses match its own when using a bit-masked comparison.

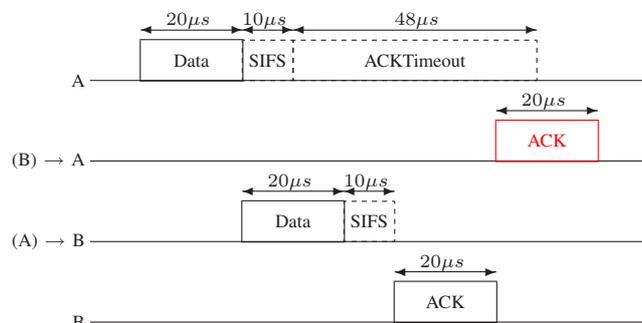


Fig. 3. Timing diagram - relayed ACKs in 802.11b

IV. EXPERIMENTS

A. Experimental Objectives

The laboratory study is intended to demonstrate the cryptographic DoS attack, evaluate its effectiveness and compare with a remedial measure in the form of the Harkins' countermeasures.

1) *Objective one: middleperson establishment:* The precondition for a successful cryptographic DoS attack is that a message can be intercepted in flight and a modified copy substituted in its place. It is, therefore, essential that a middleperson be able to inject itself between legitimate parties.

2) *Objective two: test the message modification attack:* The message modification attack should ensure that messages are received with a valid ICV but invalid MIC. Experimental validation is required to validate the attack.

3) *Objective three: invoke TKIP countermeasures:* The cryptographic DoS attack depends on the activation of TKIP countermeasures for its effect. The objective here is to conduct the cryptographic DoS attack described above and gather measurements as the basis for comparison.

4) *Objective four: invoke Harkins' countermeasures:* The Harkins countermeasures address the threat of a cryptographic DoS attack by tearing down and re-negotiating only those secure associations that come under attack. These countermeasures are implemented and compared to those of the standard TKIP countermeasures.

B. Middleperson establishment

The 802.11 specification offers little guidance for finding the BSS but allows for passive and active discovery. Most implementations scan for the desired BSS by listening for beacons and/or issuing probe requests across all of the permissible radio channels. Therefore a number of different approaches have been tried to insert the middleperson between legitimate stations:

- Simply forwarding and re-writing beacon/probe traffic in the hope that the adversary-controlled channel will be found and used by the client.
- Actively attempting to capture stations already associated with the legitimate BSS using a de-authentication attack to provoke a station into re-association (and hopefully scanning).
- Actively attempting to capture stations already associated with the legitimate BSS by injecting spoofed channel change announcements to force stations to use the attacker's chosen channel.

Prior to running the experiment the only result that could be predicted with confidence is that the channel change attack should have a high chance of success. This attack exploits the dynamic frequency selection requirements of 802.11h and should cause clients to immediately change the operating channel. The actual results are summarised in table I where a tick mark indicates that a station could be captured at least some of the time and that this behaviour is repeatable and a cross indicates that clients could not be captured repeatedly.

TABLE I
SUCCESS OF MIDDLEPERSON ESTABLISHMENT

	Beacon/Probe	Deauthentication	Channel Change
FreeBSD 6.0	✓	×	×
GNU/Linux	×	×	×
Mac OSX	✓	×	✓
Windows XP	✓	×	×

1) *Beacon/probe:* The beacon/probe response mechanism proved to be the least repeatable of the establishment techniques. Simply relaying beacons and relaying probe requests/responses to the adversary-controlled radio channel does not guarantee that the channel will be picked. Factors such as signal strength, scan order, beacon frequency and so on might all have an effect on which channel is chosen. Our assumption had been that cards often select a channel based on the received signal strength of beacons and probe responses. Only when a directional antenna is substituted for the omnidirectional antenna were clients captured repeatedly.

The scanning strategies of the operating system appears to play a big part. For example, Mac OSX starts scanning on the channel last used to communicate with a given BSS. This makes the client much more likely to find the legitimate network and establishment of the middleperson much more difficult. Only when the adversary-controlled channel had been used previously for a BSS (i.e. as the result of the channel change attack) would the Macintosh associate via the adversary controlled channel.

2) *Deauthentication:* In contrast the deauthentication method proved to be extremely repeatable with no stations ever being captured by this approach. This approach has, however, worked for other middleperson attacks described in the literature[17]. Mac OSX responds singularly badly to the deauthentication attack as can be seen in figure 4. The access point tears down the secure association in response to the deauthentication but the client refuses to do so despite every subsequent frame being answered by a deauthentication frame. In itself this is a very effective single-frame DoS attack against Macintosh stations.

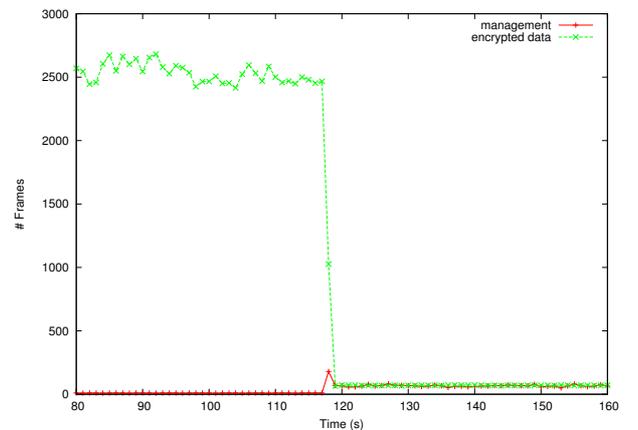


Fig. 4. Mac OSX response to deauthentication attack

3) *Channel change*: The channel change attack also proved to be extremely repeatable either working every time or never working at all. Two mechanisms have been used for this attack:

- injecting false beacons onto the legitimate channel to which a “channel change with immediate effect” announcement has been appended and
- Sending channel change action messages to individual stations.

The majority of the tested operating system/driver combinations do not appear to implement the 802.11h dynamic frequency selection behaviours and do not change channel. This has been confirmed for the free software device drivers by inspecting their source codes¹.

C. Message-modification

To validate the message modification attack the GNU/Linux `iwevent` program is used. This program monitors the RT-Netlink socket to receive notifications from kernel space of important wireless events. The monitored event is a “MICFailure” indicating that a message has been received for which the ICV is correct but the MIC is wrong. The message modification attack as outlined in section I-C causes this message to occur for all damaged frames and is completely repeatable.

D. Invoking TKIP countermeasures

The attack is conducted against the HostAP access point. To validate the message modification attack the GNU/Linux `iwevent` program is used. When the attack is conducted `iwevent` reports the MIC failures but HostAP did not enforce a one minute blackout period and did not tear down the existing secure associations. Instead an error message is issued by HostAP. Inspection of the source code uncovered the reason for the attack failure against the HostAP. In this case the bug was present not in HostAP but in the `madwifi-ng` device driver which reports the MIC failure but does not send the correct MAC address. In this case it sends the 802.11 receiver address when it should send the transmitter address. HostAP simply ignores the MIC failure and TKIP countermeasures are not invoked.

A trivial change to the device driver to report the transmitter address was made and the experiment repeated. This time the attack was successful and TKIP countermeasures are engaged. The graph in figure 5 shows the effect of a cryptographic DoS attack conducted against a steady stream of traffic from two stations. During the one minute blackout period there are several association attempts which are answered by deauthentication frames. Once the blackout has ended one of the stations re-authenticates automatically and resumes whereas the second station required manual intervention to re-establish communications and resume communicating.

¹At the time of writing an 802.11h patch for the MadWifi-ng device driver was being discussed on the developers mailing list but was not publicly available.

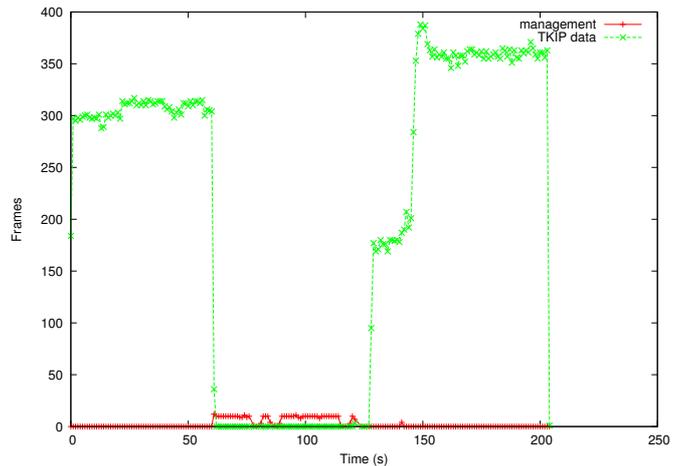


Fig. 5. TKIP countermeasures response to cryptographic DoS attack

E. Invoking the Harkins countermeasures

The Harkins countermeasures necessitated changing the HostAP implementation. HostAP is a C program and the necessary changes are restricted to the `handle_auth`, `ieee80211_michael_mic_failure`, `ieee80211_tkip_countermeasures_stop` and `ieee80211_tkip_countermeasures_start` functions. The latter function is modified to accept an additional parameter specifying the MAC address of the secure association to be invalidated. This parameter is used to end only the association which has been attacked instead of tearing down all the secure associations.

The original proposal allowed for a variable delay of at least 30ms before the secure association is allowed to be re-established. This delay is intended to slow down the adversary but still take at least one year on average for a key recovery attack to succeed. Harkins also allowed for this figure to be tuned by the administrator to allow a trade-off between key security and DoS risk. For ease of implementation our implementation fixes this timeout at 1s. In order to restrict this per-station blackout it has also been necessary to implement additional record keeping.

Testing the Harkins countermeasures follows the same procedure as for TKIP countermeasures except that the modified HostAP program is used. Over a three minute-period several stations are used to generate a stream of traffic and after one minute has elapsed the adversary conducts a cryptographic DoS attack. The results of this are summarised in figure 6 and show that there is no appreciable drop in traffic. The station affected by the attack is forced to delay for one second and then reauthenticate and associate but the total data throughput is not appreciably affected. Over several runs the same behaviour is repeated. The Macintosh, which normally requires manual intervention to re-authenticate when using P countermeasures is able to automatically re-authenticate.

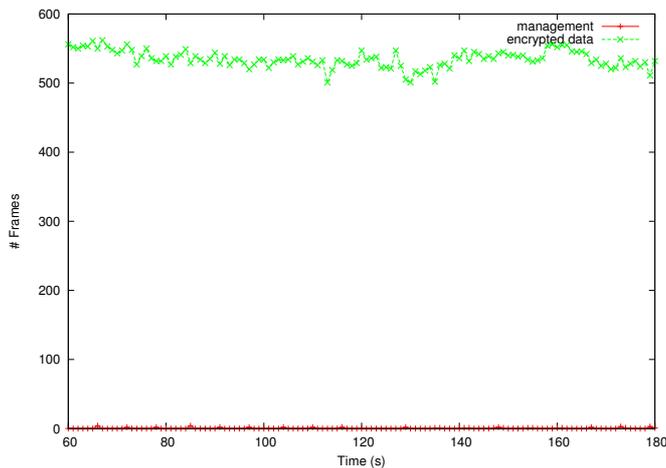


Fig. 6. Harkins countermeasures

V. ANALYSIS

The key threat posed by the cryptographic DoS attack is not as might be supposed the threat to availability. Although the attack requires very little effort and is devastatingly effective it only targets TKIP-protected traffic — one of only three available security protocols. Associations using CCMP and WEP will continue unaffected and new associations can be made using these protocols. The real threat is that this might be used as part of a security level rollback attack. The WEP cipher suite, although completely compromised, is still widely supported and used. An adversary that can deny service to TKIP users maybe able to force clients to re-associate using WEP with the attendant risks to confidentiality and integrity.

Surveys have shown that users are now more conscious of the need to use the security protocols and that the percentage of networks employing encryption has increased markedly in the period 2004–2006[18]. Although there has been a substantial increase in the use of encryption little information has been collected on the security protocols in use. There are grounds to suspect that the insecure WEP protocol comprises the majority of the surveyed sites. “Advanced security was detected in a surprisingly small number of access points across all three cities and constituted less than 1% of all encrypted traffic.”[19]

To address this question we conducted a wireless security survey of a local business district in July 2006. A total of 512 separate 802.11 wireless networks were identified with supported protocols as shown in figure II. Although this

TABLE II
JULY 2006 WIRELESS SECURITY SURVEY RESULTS

Protocol	No. of networks	Percentage
Unencrypted	196	38.3%
WEP	247	48.2%
TKIP/WEP	63	12.3%
CCMP/TKIP/WEP	6	1.2%

shows that over 13% of surveyed sites to be using advanced cryptogtaphy (TKIP and CCMP) all of the sites using these

protocols were also allowing WEP associations. In this survey, the number of networks supporting TKI/WEP is an order of magnitude greater than those supporting CCMP/TKIP/WEP. In this case for 90% of networks the cryptographic DoS attack would represent potent threat to confidentiality and integrity as users fall back to using WEP. In view of the many serious flaws in WEP the only sensible approach is for users is to abandon the use of WEP altogether. The use of TKIP should be retained as a transitional protocol and users should be encouraged to employ CCMP.

VI. CONCLUSIONS

The contribution of this paper is to have studied the cryptographic DoS attack in a testbed environment. We have shown the correct mechanism for an 802.11 message modification attack, described the implementation of this attack using a middleperson approach and compared the TKIP and Harkins countermeasures in a controlled environment.

The cryptographic DoS attack is demonstrated to be practical and can be mounted by a single adversary with limited resources. This attack requires very little work on behalf of a hostile adversary and will bring TKIP-protected traffic to a complete halt. The threat of this attack being used to accomplish a security-level rollback should not be underestimated. The Harkins countermeasures address this threat by reacting in a more measured manner to active key recovery attacks.

VII. ACKNOWLEDGMENTS

National ICT Australia is funded by the Australian Government’s Department of Communications, Information Technology and the Arts and the Australian Research Council though *Backing Australia’s Ability* and the ICT Research Centre of Excellence Programs.

REFERENCES

- [1] Nancy Cam-Winget, Russ Housley, David Wagner, and Jesse Walker. Security flaws in 802.11 data link protocols. *Communications of the ACM*, 46(5):35–39, May 2003.
- [2] Niels Ferguson. Michael: an improved MIC for 802.11 WEP. IEEE 802.11 Working Group Document 02/020r0, Institution of Electrical and Electronics Engineers, January 2002. Available from <http://grouper.ieee.org/groups/802/11/Documents/DocumentHolder/2-020.zip>.
- [3] Avishai Wool. A note on the fragility of the “Michael” message integrity code. *IEEE Transactions on Wireless Communications*, 3(5):1459–1462, September 2004.
- [4] LAN/MAN Standard Committee of the IEEE Computer Society. *IEEE Std 802.11i, Part 11: Medium Access Control (MAC) security enhancements*. Institution of Electrical and Electronics Engineers, July 2004. Available from <http://www.ieee.org/getIEEE>.
- [5] Nikita Borisov, Ian Goldberg, and David Wagner. Intercepting mobile communications: the insecurity of 802.11. In *Proceedings of the 7th Annual International Mobile Computing and Networking Conference*, pages 180–189, New York, NY, USA, 2001. ACM SIGMOBILE, ACM Press.
- [6] Nikita Borisov. Personal communication with author. email, May 2007.
- [7] Pradeep Kyasanur and Nitin H. Vaidya. Detection and handling of MAC layer misbehavior in wireless networks. In *International Conference on Dependable Systems and Networks (DSN’03)*, pages 173–, San Francisco, CA, June 2003. Institution of Electrical and Electronics Engineers.
- [8] Stephen Glass and Vallipuram Muthukkumarasamy. Denial of service vulnerabilities in the IEEE802.11 DCF. In *3rd Australian Computer, Information and Network Forensics Conference*, Mount Lawley, Western Australia, September 2005.

- [9] Dazhi Chen, Jing Deng, and Pramod K. Varshney. Protecting wireless networks against a Denial of Service attack based on virtual jamming (poster session). In David B. Johnson, Anthony D. Joseph, and Nitin H. Vaidya, editors, *The Ninth ACM Annual International Inproceedings on Mobile Computing and Networking (MobiCom 2003)*, San Diego, CA, USA, September 2003. SIGMobile, ACM.
- [10] John Bellardo and Stefan Savage. 802.11 denial-of-service attacks: real vulnerabilities and practical solutions. In *Proceedings of the 12th USENIX Security Symposium*, Washington D.C., August 2003. Available from <http://www.usenix.org/publications/library/proceedings/sec03/tech/bellardo.html>.
- [11] Bernard Aboba. Issues in pre-standard IEEE 802.11i implementations. Webpage, 2004. Available from <http://www.drizzle.com/~aboba/IEEE/prestand.html>.
- [12] Ping Q. Ding, JoAnne Holliday, and Aslihan Celik. Improving the security of wireless LANs by managing 802.1X disassociation. In *Proceedings of the First IEEE Consumer Communications and Networking Conference*, pages 53–58. Institution of Electrical and Electronics Engineers, January 2004.
- [13] Changhua He and John C. Mitchell. Analysis of the 802.11i 4-way handshake. In *WiSe '04: Proceedings of the 2004 ACM Workshop on Wireless Security*, pages 43–50, New York, NY, USA, 2004. ACM Press.
- [14] Changhua He and John C. Mitchell. Security analysis and improvements for IEEE, 802.11i. In *Network and Distributed System Security Symposium Proceedings: 2005*, pages 90–110. The Internet Society, February 2005. Available from <http://www.isoc.org/isoc/conferences/ndss/05/proceedings/index.htm>.
- [15] Dan Harkins. Attacks against Michael and their countermeasures. IEEE 802.11 Working Group Document 03/211r0, Institution of Electrical and Electronics Engineers, March 2003. Available from <http://grouper.ieee.org/groups/802/11/Documents/DocumentHolder/3-211.zip>.
- [16] Jon Edney and William A. Arbaugh. *Real 802.11 Security: WiFi Protected Access and 802.11i*. Pearson Education, 2004.
- [17] Tim Schmoyer, Yu-Xi Lim, and Henry L. Owen. Wireless intrusion detection and response: a case study using the classic man-in-the-middle attack. In *2004 IEEE Wireless Communications and Networking Conference*, volume 2, pages 883–888, March 2004.
- [18] Price Waterhouse Coopers. Information security breaches survey. Technical report, DTI, April 2006. Available from <http://www.pwc.com/Extweb/pwcpublishings.nsf/docid/F9843CD3C8E0FB828025715A0058C63B>.
- [19] Phil Cracknell. Wireless security survey of London 2006. A report commissioned by RSA Security 5th edition, RSA Security, Bracknell, UK, May 2006. Available only on request.