Computer Science Extended Essay:

Comparative Analysis of Time Complexity in Brute Force Decryption Attempts: Assessing the Security of WPA2-PSK (RC4) with TKIP Encryption versus WPA2-PSK with CCMP (AES) Encryption, in Relation to Authentication Handshake and Password Types

Research Question: How does the security of the WPA2-PSK (Personal) wireless security standard, with the TKIP encryption protocol, compare to WPA2-PSK (Personal) with the CCMP encryption protocol, against brute force decryption attempts targeting the authentication handshake exchanged between the client and the access point, for different types of passwords, in terms of time complexity?

Word Count: 3718 words

Table of Contents:

1. Introduction

WPA2 networks are very commonly used nowadays, making up a staggering 68.02% of network configurations, (Legezo), virtually anyone in a developed country has a router in their house, offices often use WPA2-protected networks, and it is most often accepted as the standard, secure protocol to use for a network which uses the process of authenticating user access with a password. The WPA2-PSK protocol ensures that the process of a client gaining access to the network with a predetermined password is taken care of, the initial connection is established through a four-way handshake, in which the access point ensures that the user has given the correct password, hence they can access the network. The problem is that the handshake, while being transferred, can be captured by a malicious third party. While the data captured is still encrypted, the attacker can now locally attempt to decrypt, or crack, the file locally on their machine. (Lu, He-Jun, and Yang Yu) This means that instead of trying passwords directly, and attempting to log in directly through the router every time, which not only would allow for a few attempts a minute at best, as some routers have systems in place that block users who seem to be trying to guess the password, the attacker can now attempt thousands of passwords per second, without arising any suspicion. This makes for a significant security flaw in the WPA2 protocol, which is often considered to be very secure. Once an attacker gains access to one's network, they can perform a plethora of malicious attacks, from man-in-the-middle, to SSH attacks, to evil portals and twins, or similar. Data shared across a network can also be very sensitive and thus exploited.

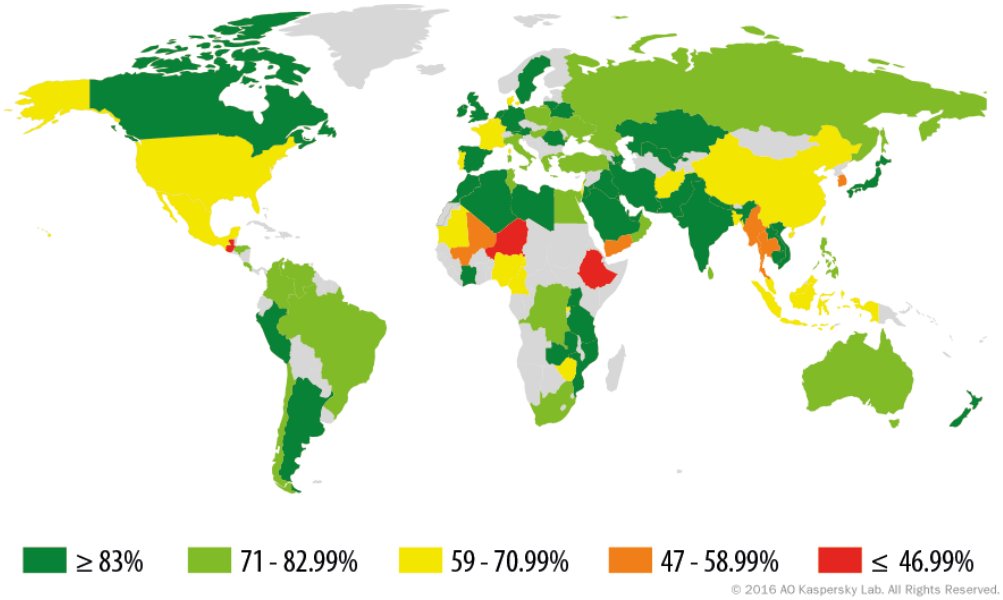


Figure 1, “Share of hotspots that use WPA/WPA2 by country”, (Legezo)

While WPA3 does exist, and offers several security improvements compared to WPA2, WPA2 still remains the dominant form of securing a network. This is also because not many routers support WPA3, realistically, the average customer who receives the router included in their home wifi plan, is likely going to use that router, assuming it has no catastrophic defects, and since these routers are oftentimes cheap, they do not support WPA3. This means that the security of one’s network is limited by the strength of the encryption they use. Strong encryption means a hacker could never realistically crack a network’s password hash, weak encryption means they could in a few weeks.

*It is important to note that WPA2 with CCMP encryption is typically called WPA2-AES, this is technically wrong as AES is the cypher, not the encryption protocol. (Gstefanick) Please keep this in mind as you read this essay. RC4 or AES encryption will

always be discussed within the context of the WPA2 protocol. Furthermore, this essay is only going to be examining the WPA2 Personal protocol. All references to WPA2 are made to the Personal protocol.

2. Scope and significance

The scope of this investigation is to assess the security standard of WPA2 networks. Since they are so commonly used, it is important to know what their vulnerabilities may be, so that better security practices can be advised. Because of how WPA2-PSK works, the four-way handshake exchange, when captured, is the most vulnerable point, as an attacker can keep the file, and decrypt it using various methods, such as brute force, dictionary attacks, rainbow tables, or hybrid methods, depending on the situation. This means that the difference between a secure and an unsecured network is whether the attacker could decrypt the file in a few days, or a few years. The purpose of this essay is to investigate if this is a security risk, if so, how severe it is, and what users could do to mitigate or prevent an attacker from gaining access to their network through this method.

3. Hypothesis and Applied Theory

Due to the fact that WPA2-CCMP uses a more advanced encryption algorithm, (Hoelscher) and should, in theory, have less security flaws than the now outdated WPA2-TKIP encryption algorithm, I hypothesize that any same password that is secured

with WPA2-CCMP will have an exponentially higher time complexity than the exact same password that would be secured with WPA2-TKIP.

4. Methodology

To fully simulate the environment, I will be using a router that uses WPA2-PSK, running DD-WRT, supporting both AES (CCMP-128 bit) and TKIP (128-bit) encryption. a device that will act as the client who connects to the network, and a third device using a packet analyzer, that will attempt to capture the authentication packets being sent across the network. once the file is captured, the cracking process will begin using Hashcat's PMKID attack. Hashcat was the tool selected for this experiment as it supports GPU cracking, is generally a popular tool, and has huge support from leading cybersecurity experts, along with its open-source nature. Different types of passwords will be used in the experiment, from very simple, low-security passwords, to very complex passwords with special characters. The decryption attempts run on the hashes will always be a brute force attack, this is because other forms of attacks, such as dictionary attacks, are more reliant on how common the password used is, if it's been previously leaked, and other factors decided by the authors of said dictionaries. This introduces other variables that could skew the results, hence, brute force will be used, since the only two possible outcomes for a brute force attack are either success, or abandonment of the attack due to high time complexity.

Different WPA2-PSK Security Options & Definitions of Terms:

Four-way handshake:

To allow a client and an access point to prove to each other that they know the PSK, without actually disclosing it, the access point and the client send encrypted messages to each other, that can only be decrypted with the PSK. If decryption of the PSK from both sides is successful, the client can join the network, if it is not, the client is rejected from the network. This exchange of messages encrypted with the PSK shows the vulnerability of this process, as once an external third party captures the packets, they can attempt to decrypt the encrypted messages. If the handshake is decrypted, the attacker knows that whatever string they used to do so is the PSK, hence, they can use that to gain access to the network

WPA2-PSK (TKIP):

By far the most common, it uses the TKIP (Temporal Key Integrity Protocol) encryption method, which utilizes RC4, generating all of its encryption using the Pre Shared Key (PSK) and an SSID (Service Set Identifier), one of the reasons it's still used today is that it's compatible with a wide range of older devices. (Goel) Not a lot of processing power is required from the router to use TKIP encryption either, making it very accessible.

WPA2-PSK (CCMP):

The most secure personal version of WPA2 uses the CCMP encryption method, which uses the AES cypher. Encryption is much stronger, providing a far more secure network framework for the users, however, if a user is utilizing an old, or not particularly powerful router, they may experience slower network speeds, as the CCMP encryption needs more processing power to be implemented. Legacy devices also do not support "WPA2-AES". This is why most networks do not use WPA2-PSK with CCMP encryption, and it is not considered a default option whatsoever.

How TKIP and CCMP work, in detail:

TKIP implements a key mixing function and a 64-bit Message Integrity Check (MIC), and re-initializes the sequence number each time a new Temporal Key is used. These low-level processes allow for a better random and unbiased encryption, making TKIP harder to crack. TKIP's RC4 cypher is advantageous for legacy device support. However, RC4 is known for creating weak encryption keys, if the initial key, which is the chosen user password is not random or strong enough, RC4 will simply output a weak encryption key. (Crashtest Security) TKIP ensures that every packet is sent with its own encryption key thanks to a rekeying mechanism.

CCMP (AES) is a symmetric-key block cypher that uses a fixed-length key to encrypt 128-bit data blocks. It also uses a nonce, and a "key seed" from the PSK and SSID to increase randomness. The algorithm used to encrypt the data is Cypher Block Chaining (CBC), in CBC, each plaintext block is XORed with the previous cyphertext

block, forming a very powerful encryption. Due to its more compute-intensive algorithm, CCMP has not seen the adoption that TKIP currently has, despite it slowly becoming more common for modern, high-end routers. CCMP can use 128-bit to 256-bit keys, compared to TKIP's 128 bits, CCMP uses a block cypher (AES), while TKIP uses a stream cypher (RC4). This means that CCMP encrypts data in fixed-size blocks, while TKIP encrypts each bit or byte of data separately.

5. **Password Strength and Authentication Methods:**

Password strength is the measure of the resilience of a password against brute force attacks or guessing. One of the main determining factors in password strength is how many guesses can be attempted. For example, a website that allows one password-guessing attempt every minute is virtually impossible to crack with an 8-digit or higher password. (In this essay, the exploit that is being explored allows for an indefinite number of attempts, limited only by the speed of the computer) In the WPA2 protocol, users are asked to generate their own passwords. This is problematic, as human users are not very good at creating secure passwords. This can be proven by a MySpace phishing scheme which took place in 2006, revealing 34,000 passwords, with only 8.3% of them using mixed case, numbers, or symbols, having overwhelming majority of the passwords being only around 8 characters long. (David Jaeger et al.)

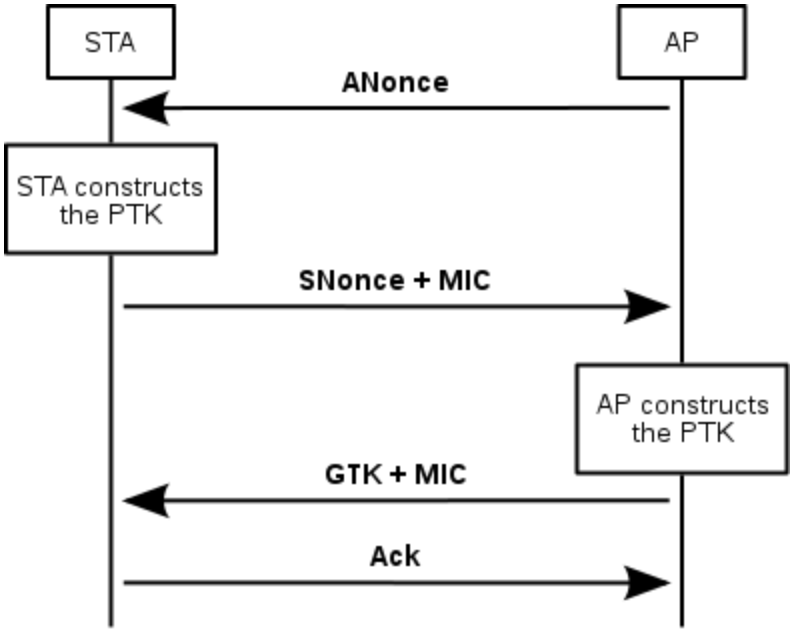


Figure 2, A diagram of the four-way authentication handshake (Sklavos, Nicolas, and Paris Kitsos).

In the authentication protocol I am analyzing, the initial authentication is carried out with a PSK, In PSK authentication, the PMK is the same as the PSK. This typically is derived from the WiFi password, hence making the authentication package capturable, and once an attacker can crack the package, they also have the PSK, which is the authentication password. This creates a very significant problem and security flaw, as both TKIP and CCMP hashes can be captured from any third party “listening in”. This means that the network security is reliant on the strengths of those encryption algorithms.

6. Why brute forcing?

Brute forcing is a very primitive way of attempting to decrypt a target file or message, however, in theory, brute forcing will always eventually succeed. (Tsekleves, Lampoudis and Tsitroulis) The only limiting factor is time and processing power. This means that across a variety of different passwords, their safety will be evaluated by how long it would theoretically take to decrypt them using brute force methods. While far more sophisticated methods exist, such as rainbow tables, wordlists, or perhaps, a combination of wordlists and brute forcing, these are much less time-limited, and much more limited by chance. This means that while a very large wordlist might only take 2 days to go through, there are only two possible results; the target password was in the wordlist, or it was not. This would not be able to provide generalized results on the safety of passwords, as a 24-digit password that happened to appear in the wordlist would be considered unsecure, wether an 8-digit password that didn't, would be considered secure. Since the nature of wordlists is controlled by those who make them, mostly sourcing their passwords from data dumps and leaks of large website and company databases, the security of a password becomes probabilistic and binary, rather than a matter of degrees, making it hard to extrapolate and draw conclusions from.

7. The Mathematics of a brute force attack:

A traditional brute force attack has a keyspace (all possible characters) of 95, this means that all the possible combinations to guess a password are 95 to the power of the password length. This begins to show one of the key points of this essay; A long

password is the easiest way to ensure that it remains theoretically impossible to brute-force decrypt within a reasonable amount of time. Meaning that if a person were to make a complex password, they would be safe.

However, humans don't typically use the entire keyspace.

Hashcat's keyspace-specific brute force attack allows the user to specify the keyspace, this means that, as previously shown, humans are very unlikely to use special characters in their passwords. This already reduces the possible combinations from 95^x to 62^x. This can be taken even further, for example, if we assume the person only uses lowercase letters and numbers for their passwords, this reduces the possible combinations to 36^x.

This means that if the brute force attack is tailored to generic, (or more common) password practices, the potential possible combinations for an 8-digit password can go from (95^8) 6.6 quadrillion possible combinations (6,634,204,312,890,625) to (36^8) 2.8 trillion possible combinations (2,821,109,907,456). This amounts to a decrease of ~99.96%.

In theory, the amount of attempts needed to successfully brute force a password is equivalent to half all the possible combinations. So, the average amount of attempts needed to brute force an 8-character alphanumeric password (without capital letters)

would be around 1.4 trillion. Power like this can be bought, rented, or acquired with methods such as FPGAs. (Kammerstetter et al.)The desktop RTX 2070 used in this experiment has a Hashrate of 400kH/s on Hashcat's WPA-PBKDF2-PMKID+EAPOL benchmark, meaning that it could be able to decrypt that password in around 41 Days.

8. Running the Experiments:

The experiment was run with a router dedicated for this purpose, which was flashed with dd-wrt, making it much easier to fine-tune and modify for this experiment, especially as most routers don't allow their users to make technical changes. The router would have the passwords changed, and a computer would log in to the router with the password, while a second computer would be running Hcxdumptool, collecting the transmitted handshakes with its packet analyzer. This process was repeated until all 12 hashes were collected (6 TKIP + 6 CCMP).

For the experiment, there were 6 different types of passwords, which were repeated twice, making 12 passwords, which would be encrypted either with TKIP or CCMP.

Table of Types of Passwords and Passwords Used in Experiment:

Types of Passwords:	Password Rules:	Password 1 (TKIP):	Password 2 (CCMP):

Type 1	8 characters, numbers only	65852739	16179488
Type 2	8 characters, lowercase alphanumeric	3209cweo	crtu4581
Type 3	8 characters, lowercase alphanumeric with only the first letter capitalized	Giea3908	Kfhu0183
Type 4	8 characters, full alphanumeric	f73Ov2Ep	2309ZMpw
Type 5	8 characters, full alphanumeric with special characters	lP$(w3L1	*@wkVI82
Type 6	10 characters, lowercase alphanumeric	mkibq19876	78962wfgjq

2 Passwords for each rule were used in this experiment, making a total of 12 passwords

Justification for types of passwords:

All passwords except for Type 6 passwords are 8 characters long, this is because a very large majority of passwords used are 8 characters long. However, Type 6 passwords are 10 characters long, to see how much security it would add. Type 1 and 2 passwords are statistically the most common, and presumably the least secure. Type 3 passwords represent passwords that use capitalized letters, but make the common mistake of only capitalizing the first letter (this is surprisingly common), Type 4 passwords represent a password that implements capital letters more randomly. Type 5 passwords represent

those who make an effort to have a very secure password, and Type 6 simply represents a longer, yet somewhat common type of password.

Example for collected handshake of the password "65852739"

WPA*02*caadd2b9a562a436ae7f1d2be49ee23a*c8d71987d243*04cf4b1fb576*64642d777274*d861c101b6d9ac2e26993f4f447f2ed27488a187893277eba790ccb
9fe9f39b3*0103007502010a0000000000000000f98e7142c4866a7ea157a3efeda357c3083b4d8b00d5d4df65f47aa7106ba858164800000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000001630140100000fac020100000fac020100000fac028000*10

MIC AP_MAC Client MAC ESSID NONCE_AP EAPOL_CLIENT MESSAGEPAIR

Table with Experiment Results

Results:	TKIP (Time to Crack)	CCMP (Time to Crack)
Type 1:	Cracked in 113 seconds	Cracked in 127 seconds
Type 2: (After 12 hours)	74 days, 8 hours estimated	76 days, 12 estimated
Type 3: (After 12 hours)	55 days, 19 estimated	56 days, 21 estimated
Type 4: (After 12 hours)	15 years, 294 estimated	16 years, 109 estimated
Type 5: (After 12 hours)	483 years, 141 estimated	493 years, 52 estimated
Type 6: (After 12 hours)	267 years, 13 estimated	272 years, 137 estimated

Example of cracked results:

7a122c34b16ddf718fe0333b92443bc1:c8d71987d243:4aefff1224e0:dd-wrt:**16179488**

caadd2b9a562a436ae7f1d2be49ee23a:c8d71987d243:04cf4b1fb576:dd-wrt:**65852739**

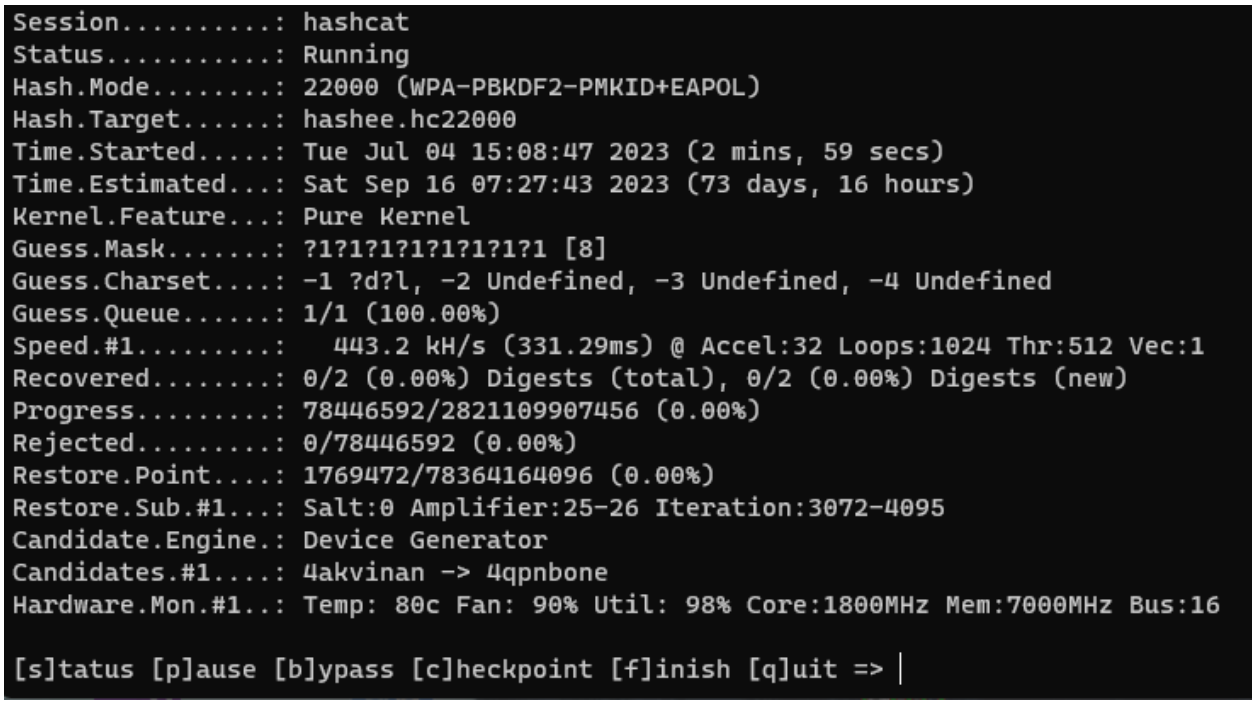


Figure 3, a Screenshot of Hashcat Running

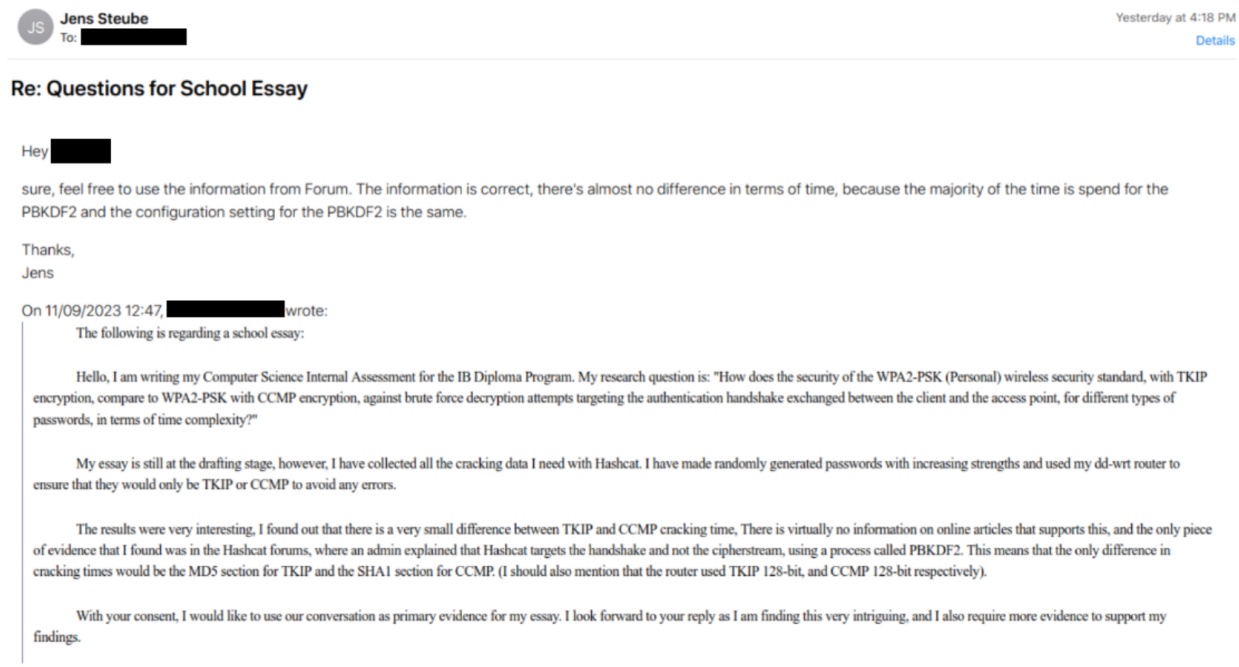
Hashcat can extrapolate the GPU's performance throughout the process and average it out to a longer timeframe to make its predictions, this makes the predictions quite accurate. As seen, there are not 73 days available for this experiment to run, hence the experiments were run for a minimum of 12 hours if not yet cracked so that Hashcat could establish an accurate prediction over that timeframe.

9. Why do the passwords take similar amounts of time to crack?

Through the experiment, I was very surprised to find that the cracking time for TKIP versus that of CCMP was much more similar than what I expected, after looking

into it, I found out that it's because Hashcat's cracking process targets the handshake, not the cypherstream, hence the time difference is very small, as the process that's used, PBKDF2, a cryptographic key derivation function, is the same for both TKIP and CCMP encryption. After that, Hashcat uses MD5 for TKIP and SHA1 for CCMP, but these parts don't take significant amounts of time, as the PBKDF2 makes up for the majority of time complexity. This shows that the difference between TKIP and CCMP security is far less than expected, and using the correct techniques, the encryption strength of WPA2 CCMP can be strongly degraded.

Because of these peculiar results and their conflict with the majority of information online, I decided to contact the lead developer of Hashcat, Jens Steube, who responded, confirming my test results



Jens Steube
To: [redacted]

Yesterday at 4:18 PM
Details

Re: Questions for School Essay

Hey [redacted]

sure, feel free to use the information from Forum. The information is correct, there's almost no difference in terms of time, because the majority of the time is spend for the PBKDF2 and the configuration setting for the PBKDF2 is the same.

Thanks,
Jens

On 11/09/2023 12:47, [redacted] wrote:

> The following is regarding a school essay:
>
> Hello, I am writing my Computer Science Internal Assessment for the IB Diploma Program. My research question is: "How does the security of the WPA2-PSK (Personal) wireless security standard, with TKIP encryption, compare to WPA2-PSK with CCMP encryption, against brute force decryption attempts targeting the authentication handshake exchanged between the client and the access point, for different types of passwords, in terms of time complexity?"
>
> My essay is still at the drafting stage, however, I have collected all the cracking data I need with Hashcat. I have made randomly generated passwords with increasing strengths and used my dd-wrt router to ensure that they would only be TKIP or CCMP to avoid any errors.
>
> The results were very interesting, I found out that there is a very small difference between TKIP and CCMP cracking time, There is virtually no information on online articles that supports this, and the only piece of evidence that I found was in the Hashcat forums, where an admin explained that Hashcat targets the handshake and not the cipherstream, using a process called PBKDF2. This means that the only difference in cracking times would be the MD5 section for TKIP and the SHA1 section for CCMP. (I should also mention that the router used TKIP 128-bit, and CCMP 128-bit respectively).
>
> With your consent, I would like to use our conversation as primary evidence for my essay. I look forward to your reply as I am finding this very intriguing, and I also require more evidence to support my findings.

Figure 4, Email Sent to Hashcat Lead Developer

PBKDF2 is an iterative cryptographic key derivation function (Kaliski), meaning that a hash can be made through as many iterations as the user wants. With every iteration, the hash gets harder to crack. This means that PBKDF2 can be used to create very powerful hashes, but can also be integrated in simple circuits, with low-power CPUS, and very little RAM. This is exactly the hardware configuration of the average home or office router. Because of the weak hardware, PBKDF2 does not go through many iterations, meaning that the GPU cracking method used in this essay worked wonderfully because of this. (Percival) As previously mentioned, ASICs or FPGAs could also efficiently crack this. This means that a much better solution would be changing to the WPA3 protocol, as in theory, it should be much safer than either WPA2-TKIP or "WPA2-AES". Another reason was that CCMP supports 128-bit, 192-bit, and 256-bit encryption, however, the router only supported 128-bit encryption, as any higher would take even more processing power than it already does, thus slowing down the network and adding latency. Hence, it is very uncommon to find a router that uses "WPA2-AES" with 256-bit CCMP and functions acceptably. This is why CCMP 256-bit is often used for non-latency-critical applications, such as encrypting government files. However, no matter how strong the encryption, a hash can always be susceptible to a key attack, such as a dictionary attack if it uses a common or leaked password.

10. Interpreting the results:

A leak from 2006 implies that 91.7% of passwords fall into the lowercase alphabetic-only category, whereas a leak from 2021 claims that 70% of WPA2 passwords collected in the real world, according to the researcher, were cracked with “relative ease” (Nichols) While not a lot of information can be found on how long it took, the researcher said that he used wordlists for his attack, and also noted that a surprising amount of passwords were only 10 numerical digits.

While it is hard to extrapolate these data points to see where we stand today, as asking people for their passwords to gather real-world data is not a particularly good idea, it is clear that even if 50% of wifi passwords only used lowercase letters, or only used numbers, they could be cracked in minutes, as shown by the testing. As further discovered, CCMP does not provide that much of a significant security advantage compared to TKIP when being approached properly, as shown by the sophistication of Hashcat. Overall, the best defence against faulty encryption, as shown, is a strong password. Using a combination of alphanumeric characters and special characters will ensure that the password remains almost impossible to crack, either with WPA2-TKIP or “WPA2-AES”

Furthermore, referring to the previous calculations, a hacker could easily buy a more powerful GPU than the one used in this experiment, which could reduce the cracking time to half or a third of what was shown in the experiment, or, a hacker could

rent cloud GPUS, and with a few hundred dollars, could take a tenth of the time that was shown in the experiment. Depending on how valuable a target can be and how much effort a malicious third party is willing to put in, the results from the experiment show that their security can be compromised in around a month, if they have average passwords.

11. Conclusion:

Overall, from all the data collected, processed, and interpreted, it is safe to say that WPA2-TKIP and WPA2-CCMP have very similar security, and in real-life applications, with real-life hardware and with advanced hacking software that anyone can download for free, WPA2-CCMP is not as safe in practice as it originally seemed, although it supposedly being much more secure in theory. This unexpected finding is in part due to Hashcat's chosen approach to target PBDFK2, which has proven very effective, and makes the cracking times between TKIP and CCMP almost identical. Hence the conclusion to this experiment is that any router using any form of WPA2 security should be deemed not secure. The easiest way to fix this while still using a WPA2 network would be to choose a long, random and varied password, which in turn creates a strong encryption key, and as shown in this experiment, becomes very challenging to crack. The small effort of adding extra keyspace characters to the password can save a network from a cybersecurity attack, which is much more common and feasible than people think.

12. Bibliography:

"Crashtest Security, Rivest Cipher 4 (RC4) - Definition, Impact and Prevention." *Crashtest Security*, 16 Nov. 2022, www.crashtest-security.com/disable-ssl-rc4/#:~:text=Biased%20outputs%3A%20RC4%20produces%20keystreams,the%20doors%20for%20WEP%20attacks.

David Jaeger, Chris Pelchen, Hendrik Graupner, Feng Cheng, and Christoph Meinel. "Analysis of Publicly Leaked Credentials and the Long Story of Password (Re-)use" Mykayem, 2016, http://mykayem.org/pdfs/Jaeger2016.pdf. Accessed 14 September, 2023

Emmanuel Tsekleves, Dimitris Lampoudis, Achilleas Tsitroulis. "Exposing WPA2 Security Protocol Vulnerabilities - Researchgate." Research Gate, 2014, https://www.researchgate.net/profile/Emmanuel-Tsekleves/publication/262282304_Exposing_WPA2_security_protocol_vulnerabilities/links/590dda384585159781859ce0/Exposing-WPA2-security-protocol-vulnerabilities.pdf. Accessed 14 September, 2023

Goel, Shorya. "Data Security - WPA-2 PSK Vulnerabilities." Encryption Consulting, 16 Aug. 2022,

https://www.encryptionconsulting.com/is-wpa2-psk-vulnerable/. Accessed 14 September, 2023

Gstefanick, "A closer look at WiFi Security IE (Information Elements)" *Airheads Community*, https://community.arubanetworks.com/browse/articles/blogviewer?blogkey=6748b650-12a4-466a-8aca-f3cb352e8277. Accessed 20 July 2023.

Hoelscher, Penny. "WPA2: What Is the Difference between AES and TKIP?" *Comparitech*, 18 Mar. 2021, www.comparitech.com/blog/information-security/wpa2-aes-tkip/. Accessed 14 September, 2023

Kaliski, Burt. "RFC 2898: PKCS #5: Password-Based Cryptography Specification Version 2.0." *IETF Datatracker*, 1 Sept. 2000, www.datatracker.ietf.org/doc/html/rfc2898#section-5.2.

Kammerstetter, Markus, et al. "Efficient High-Speed WPA2 Brute Force Attacks Using Scalable Low-Cost FPGA Clustering." SpringerLink, Springer Berlin Heidelberg,

https://link.springer.com/chapter/10.1007/978-3-662-53140-2_27 Accessed 14 September, 2023

Legezo, Denis "Research on Unsecured Wi-Fi Networks across the World." *Securelist English Global Securelistcom*, 13 May 2021, https://securelist.com/research-on-unsecured-wi-fi-networks-across-the-world/76733/. Accessed 14 September, 2023

Lu, He-Jun, and Yang Yu. "Research on WIFI Penetration Testing with Kali Linux." Complexity, Hindawi, 27 Feb. 2021, https://www.hindawi.com/journals/complexity/2021/5570001/. Accessed 14 September, 2023

Nichols, Shaun. "Researcher Cracks 70% of Neighborhood Wi-Fi Passwords: TechTarget." *Security*, 26 Oct. 2021, www.techtarget.com/searchsecurity/news/252508625/Researcher-cracks-70-of-neighborhood-Wi-Fi-passwords. Accessed 14 September, 2023

Percival, Colin. *Stronger Key Derivation via Sequential Memory-Hard Functions - Tarsnap*, www.tarsnap.com/scrypt/scrypt.pdf. Accessed 10 Oct. 2023.

Sklavos, Nicolas, and Paris Kitsos. “The RSNA 4-Way Handshake” *The RSNA 4-Way Handshake | Download Scientific Diagram - Researchgate*, www.researchgate.net/figure/The-RSNA-4-Way-Handshake_fig2_235760646. Accessed 12 Sept. 2023.