Proceedings of the 8th INDIACom; INDIACom-2014
2014 International Conference on "Computing for Sustainable Global Development", 5th – 7th March, 2014
Bharati Vidyapeeth's Institute of Computer Applications and Management (BVICAM), New Delhi (INDIA)

Advanced Encryption Standard – Cryptanalysis Research

Daniyal M. Alghazzawi[1], **Syed Hamid Hasan**[2] and **Mohamed Salim Trigui**[3]

Abstract – Advanced Encryption Standard (AES) has been the focus of Cryptanalysis since it was released in the 2001, November. The research gained more important when AES as declared as the Type-1 Suite-B Encryption Algorithm, by the NSA in 2003(CNSSP-15). Which makes it deemed suitable for being utilized for encryption of the both Classified & Un Classified security documents and system. The following papers discusses the Cryptanalysis research being carried out on the AES and discusses the different techniques being used establish the advantages of the algorithm being used in Security systems. It would conclude by the trying to assess the duration in which AES can be effectively used in the National Security Applications.

Index Terms – Network Security, Encryption, Cyptanalysis.

I. INTRODUCTION

Since the beginning of times and the advent of encryption algorithm the information related to the algorithms used for encryption of the secret documents was also kept secret. However, in the year 2003 NSA declared AES, a public Domain Encryption, to be suitable for being used for encryption of Classified documents & information. This was a major shift in the way encryption algorithm were looked upon by the world, as the effectiveness of any algorithm cannot be tested by keeping it secret, on the other hand the public domain encryption is available to all for continuous, rigorous and high level of Cryptanalysis. Thus, the advancement or development in the algorithm is known to users and hackers alike, which test the real metal of the algorithm.

If the adversary is at par for a normal consumer application it poses no real danger, however when we change the scenario and the application to a military application for communication things change drastically. We cannot afford to have our enemies identifying vulnerabilities in the algorithm and using them to hack the communication channels. If we have applied vulnerability found then there would be periods of low confidence which would be sustained till the time a different algorithm is applied. Thus it becomes quintessential that the people using or providing the communication equipments are ahead in the cryptanalysis. This paper aims to facilitate the same process. We would discuss the previous and present research efforts in the AES Cryptanalysis domain in Sec II of the paper. The section is sub-divided into 5 sections. Subsection 1 deals with preexistent attacks on AES which were catered to by its design. Subsection 2 deliberates on the Algebric Attack and its developments. No 3 talks about SAT solver's progress and hybrid-attack. In the 4th we discuss the Development of sidechannel cryptanalysis. In the last subsection we discuss the peculiar attack on AES and its related-key vulnerability. Now since AES is deployed in the application apart from encryption of traffic (e.g. Hash(#) Function), it becomes more critical.. Sec III discusses suitability of AES for the national security application. An assessment of the duration in which AES can be effectively used in the National Security Applications in done in this section. Sec IV concludes the paper.

II. PRESENT RESEARCH EFFORTS

A. Pre-Existing Attack

The linear relationship between the input & output of a function block are examined by Linear Cryptanalysis [1]. In cases of block ciphers, linear combination of plaintext pattern & linear combination of cipher-text pattern are equated with linear combination of key bit. The aim is to find relationships that are effective either considerably more-or-less 50 percent of the times. This results in making a "biased" approximation that is in turn useful for determining key bits. A "biased" linear approximation for the algorithm is identified first for the linear attack. plaintext pattern are then applied to retrieve the resultant ciphertext pattern which are combined in a linear fashion (mod-2- sense) as per approximation. This results in a combination of linear key bit. Values of some key bits are guessed through ample number of trials. The attacks are made more successful and accurate by more trial, Comprehensive enumeration is done to discover the rest of the key bits.

Relationship difference between the input & output of the function blocks are exploited by the differential cryptanalysis [2]. For an algorithm with encryption, plaintext pattern that have fixed difference are inspected. The aim is to find out "characteristic". The Characteristic are explicit difference in pair of plaintext pattern that have high likelihood of producing specific difference in the ciphertext pair, for any given key. Pair of plaintexts with fixed difference are applied for differential attack, then difference in the ciphertexts pair are observed and finally probability to various candidate subkey is assigned. The probability is assigned on basis of knowledge of the characteristic of the algorithm by the cryptanalyst. The correct key is obtained by ample number of trials.

At [3] the boomerang attack, developed by Wagner, does not use pairs but data Quartets and can be considered an upgrade to standard differential cryptanalysis. Quartets of plaintext are carefully chosen & observed in reference to the respective

Information Security Research Group, Faculty of Computing and Information Technology, Department of Information Systems, King Abdulaziz University, Kingdom of Saudi Arabia.
E-mail: [2]*shh786@hotmail.com*

Quartets of ciphertext & intermediary states. Wagner demonstrated application of the attack on few of the less known block cipher. It was claimed by Biryukov in 2005 [4] that boomerang attack on 5 & 6 round of AES is much quicker as compared to the comprehensive key search and he also claimed that it was two times faster than the Square attack which was developed by the originators of AES. The latest work on the boomerang attack on AES was the related-key attack in [49] discussed in sec 3.

Differential cryptanalysis are generalized into truncated differential where partially assessed differential [5]. The partial differential are clustered in the form of a pool that contains the difference pairs. This yields figures which enable reduction of complexity ensuring an effective attack.

The attack which was originally suggested for the Square-Block Cipher has been generalized into the Square-attack [6]. A plaintext "multiset" which has specific property is meticulously chosen for this attack. Then result is examined after the multiset is propagated over a number of rounds along with it being applied to the algorithm. The performance of the algorithm statistically is revealed by the perseverance of the property that in turn is utilized for revealing key bits.

A high-degree polynomial is used for to model the cipher for the interpolation attack[7]. The key dependent coefficient is then obtained by solving the polynomial. If a low degree compact expression that describes the cipher is available the technique proves to be quite effective.

Even before the designing of AES, the doctrines of interpolation attack, Square attack, truncated differential, linear cryptanalysis, and differential cryptanalysis had already matured. In [8], it was established by the AES authors that there should not be any differential trail that has a predictable propagation ratio of more than 21-n for the cipher to termed as secured from differential cryptanalysis and there mustn't be linear trail that has correlation coefficient which is more than 2n/2 for a cipher to be secured from linear cryptanalysis. They also demonstrated that that AES fulfills the condition with 8 or more rounds, thus secured from the attacks. Additionally AES is secured from interpolation attack by design, from Square attack with 7 or more rounds, and from truncated differentials 6 or more rounds.

B. Algebraic Attack

It was in 2002 that the Algebraic attack was introduced for the first time in [9]. This technique treats AES as combination of multivariable polynomial-equations across a singular Galois field, with the aim of recovering the key variables by solving these equations. One of the key features of the algebraic attack is the requirement of just one or very few plaintext or ciphertext pair, where an unknown key was used for encryption. It is completely opposite to the standard linear attack on DES, which can be managed on computational level, however the requirement of the pairs from them is about 240 which is unrealistically high. While the algebraic attack can be any danger only in case of the equations are resolvable with a size a couple of thousand equations & variables. We do not have any convincing proof that is reasonable to do such computations, while we know that working on a much smaller set is notoriously difficult.

Kipnis & Shamir [13], in 1999, were probably the first to draw researchers' attention to a generic strategy where we have a combination of multivariable-polynomials that describe the relationship between different variable, keys and i/o of few cryptographic functions, initially attempt is made to present it in the form of a single univariate-polynomial of an exceptional degree across an extensions field, after that the initial cryptanalytic challenge is reduced into a quadratic equation system, across the extensions field. Relinearization method me be used to attack such system, as they are easy to handle, however a large number of variable are required.

In 2000 Shamir, Patarin, Klimov and Courtois extended it [12] into a technique that can potentially be used to attack AES, it was named the XL algorithm (eXtended-Linearization). This technique utilized linearization for solving combinations of multivariable quadratic-equation. Which was further enhanced in 2002 by Pieprzyk and Courtois [9] to form the eXtended-Sparse-Linearization (XSL algorithm). 2 Properties obtained by the cryptanalysis of the huge system of equation were the target of exploitation; first that the system are quite scant and second that they are over-defined. After that we had quite some work done to improve these algorithms, but we also had work done to suggest that these attack were impractical if applied in accordance to the original aims.

The ability of determining a low degree polynomial depiction of the cipher's output is relied upon by the Cube attacks. Once identified, the expression is resolved, by utilizing a smart iterative approach to discover the key bits. Stream ciphers that have LFSR structures are most susceptible to this kind of attack [10]. However, since any algebraic polynomial which describes a decent block-cipher will be quite high degreed for allowing the attack to be successful in comparison to a simple brute force attack searching for a key space, DES and AES are considered invulnerable to the cube attack[11].

Generally speaking, AES appears to be over-designed in reference to differential and linear cryptanalysis. However, for algebraic attacks it is not so. No one actually has a complete answer as XSL's general design vaguely identifies a particular algorithm to be used, thus leaving a lot for the implementer to decide. As a result we can consider that the special structures of the available quadratic and linear identities were not properly exploited in all failed implementations.

Courtois in 2006, described the Courtois Toy Cipher (CTC) & the upgrade towards CTC2, which were attacked in a nice manner by him [17] utilizing the simple algebraic-method. Along with one more paper [16], he appeared to present a much dangerous and stronger result of attacking block cipher through algebraic-method. Not only that, he delayed the publication of these to reduce possible harm that would result from the quick unforeseen attacks. However, it not turn out to be as apprehended on the other hand it resulted in a notion build up that the algebraic attacks were impractical. Work done recently by Keller and Dunkelman [18] indicate that the algebraic attacks succeeded on a few CTC versions. This does not indicate it to be a powerful attack, but that this technique has special features in terms of cipher design. Thus is interesting to perceive that the Algebraic attack would be successful against the AES, it being algebraically elegant and very structured. Yet, we cannot be very sure that there would be any major improvement in the methods in the recent future.

C. SAT-Solver & Hybrid Attack

The AES or DES block cipher could be presented as an highly complex Boolean expression that involves many variable. The variables are: cipher-text output bit, key-input bits and plain-text input bits. This expression would result in a true value only when the cipher-text bit are equal to the plain-text bit encrypted by the key-bits. A possible way of attacking a block cipher is setting cipher-text and plain-text variable in the expression to their corresponding values of a known plain-text and cipher-text pair, thus discovering values of the key variable that result in the expression to be true. The above is a classic Boolean-satisfiability problem (SAT). For finding solutions to SAT problems computer programs known as “SAT solver” are used. Some of open-source SAT solver of the modern times SAT4J, MiniSat and zChaff. The SAT solvers does not apply a brute force search by utilizing multiple combinations of values of the variables, it rather uses a tentative approach of assigning value to the variable one by one till there is conflict resulting in the Boolean expression to yield a false value. Different values are then assigned to the variables by the SAT solver in order avoid conflict in a backtracking fashion. The approach of backtracking based on the conflict eliminates a large area of the search space and yields results in a lot less time than the use of brute force by the SAT solver.

There were attempts to attack DES, during some studies by use the above mentioned technique. Massaci & Marraro [19] and Massaci [20] found the keys after 3-rounds and 2-rounds DES, respectively. Even though, theoretically, the key can be discovered by the SAT solver after any count of round, which may evenbe to the extent of the complete 16 round of DES, still it takes too long to get the results. One possible; explanation for this is that for the 16 round of DES expression does not result in conflict till values are assigned to most of the unknown key variable. Which results in SAT solver not able to eliminate the search space or most of it and taking almost the same amount of time as the brute force. Even though we do not know of any work that just dumped the AES Boolean expression in the SAT solver, it is not probable that the AES can successfully attacked in this manner.

There can also be a Hybrid attack by combining the SAT solver and other techniques for better results. At [21], Potlapally et al. informed of a combination of SAT Solver and side-channel attack on the AES, 3DES and DES. It was mentioned that if values of the input & output bits for any of the 10 round on AES was provided by the side channel attack then the SAT-solver can discover the entire 128 bit key. Yet, the actual attack was never carried out nor the assessment, of difficulty that would be faced to find all input & output obtained by using side-channel technique, done, thus it is not known if this hybrid attack is indeed practical.

Bard and Courtois [22] described one more hybrid-attack on DES, which was the combination of SAT-Solver with the algebraic attack. The DES S-boxes were represented by them as a nearly-linear, sparse, and large system of equation in GF(2). ("Nearly-linear" depicts maximum of 1 nonlinear term in all equations.) They were utilized to make equations hat described the entire cipher for a few round. Then the equation was transformed into a Boolean expression. A subsection of 36 key-bit was then found using the SAT solver, while the rest of the 20 key-bit were fixed. (The key-bit not discovered by the SAT solver could alternatively be discovered through brute force.) Using this technique only in 6 rounds the Key for DES was found.

We again do not have any knowledge for any research that tried the hybrid-attack of SAT-solver & algebraic on AES, but based on the present level of AES advancement it is likely that if done it would be unsuccessful. Even then, we need to keep a watch for this technique, despite being new. As we know that algebraic attack may not be able independently break AES today, yet, there will be improvement in algebraic technique; and the SAT-solver program would also improve; thus a combination of these two technique may finally be a threat for AES. Additionally, as Bard and Courtois pointed out, that the SAT-solver/algebraic attack finds the key from only single known cipher-text & plain-text pair. While the differential and linear cryptanalysis require numerous ciphertext-plaintext pair. As a result it is more likely that the SAT-solver/ algebraic attack is successful, as it is not practically possible for an attacker to gather enough ciphertext/plaintext pair for launching a differential or linear attacks.

D. Side Channel Attack

The side channel attack utilizes info that is leaked out of the cryptosystem because of the loopholes in the system‘s physical implementation, instead of the cryptographic weakness in the algorithm. Any information extracted from noticeable parameters e.g. variations in acoustic emanations, thermal emanations, electromagnetic radiation, power consumption or timing can form the basis of leaking of the critical data such as key variable or plaintext bits.

A few of these techniques are: fault injection based attack, simple power analysis attack, differential power analysis attack, and timing attacks. The timing analysis capitalizes on the relationship of function’s run-time inside the cryptographic device with the sensitive data element under process. Model of the system along with the deviations in the time of execution of the function is used to ascertain sensitive data-bits. Even though, being limited by requirement of precise measurement, yet the timing attack can be very dangerous as can be launched remotely and are non-invasive [23]. DPA (Differential Power Analysis) compromises the the security of cryptographic device by enabling the analysis of the devices’ power consumption. SPA (Simple Power Analysis) on the other hand does not need statistical analyzing and hence a simple attack [24] [25].

Cryptographic keys are found by Fault injection based attack through exploiting computational errors [26] [27]. The attacker introduce Computational errors into the cryptographic device through the device being exposed to certain physical effect e.g. providing input that are beyond its specifications(input timing , input levels, clock rate,etc.) excessive temperature, or electromagnetic radiation. A fault model along with the miscomputed result, is used to obtain sensitive data. Some of the other example of side channel attack are electromagnetic emanation-analysis and acoustic attack [28] [29].

Architectural features on the micro level of the AES Software implementation are the subject of the timing attacks in most of the recent development. The cache usage and the secret key correlation is taken advantage of by the Cache based attacks. For a single thread implementation it is done through a direct timing analysis [30] and for a multiprocess environment the attack is done through a dependent coresident process. The "cache collision" attack was described at [31] and it claimed to have recovered the complete key by just utilizing 213 timing sample. The "cache usage" attacks utilized a process with no privileges and is reported at [32] it can retrieve 45.7 bit of the key with just 1 min of timing-data. An alternate method capitalizes on the timing dependency amongst the branch prediction ability which is present in most of the high-performance-micros and secret-keybits [33], [34]. A successful attack was shown on the RSA algorithm and it was suggested that the symmetric cipher be generalized in future research work at [35]. The attacks can also be applied to any hardware implementation that use the same elements for processing.

Intrusion to the physical system is protected by the Military. So we may assume that they are protected from the power analysis attack. Yet, any lacuna in the design of the equipment can lead to remote monitoring of parameters correlating with the power consumption (transmission envelope power and electromagnetic leakage). A monitoring device could be hidden in the machine either during the design or later by someone to monitor the power consumption of the device. Like the timing analysis attack even the power analysis attack exploit the weakness in the AES implementation. Cache usage can be utilized to obtain information related to the secret key as this information is leaked though the power consumption profile. The AES final round is subjected to the "cache trace attack" at [36]. The method of optimized constraints is used to discover the complete key through the power trace of encryption in-between Five to Fifty. We have a lesser complex cache attack was reported as [37] which required 256 traces for deriving the complete key. Any architecture hardened to DPA through 480 traces can be yield the key to a version of these attacks. Breveglieri and Boracchi investigated DPA application against AES's hardware implementation of S-Boxes at [38]. They indicated that the DPA is successful even against the AES's hardened hardware-implementation. We come across one more interesting research that combined the power-analysis technique with the analytical methods. SPA is utilized for detecting the cryptographic collision at [39]. This attack required a small sample base and bypassed the complex statistical tactic. A 128 bit key was revealed by just 40 power measurement through a plaintext attack, as reported by the authors.

Fault injection analysis is another main area of research for the side channel attack on AES implementation[26]. Even though AES is known to be susceptible against fault analysis, yet the cryptosystem should be physically possessed by the attacker and even access to the encrypting device may be required[40] for the attack to be successful. Additionally, the device's "fault model" is required along with reliable means to introduce fault in the target machine without damaging it permanently.

The availability of the fault model is required well in advance before the attack and also detailed knowledge regarding the system's design is required. Even if, military tactical communication is not really threatened by the current level of fault-injection analysis, still the research being done in the field is quick and we can see emergence of practical application outside of the tactical-environment. An AES-based smart card is under powered for inducing setup time violation, as a demonstration of Predictable fault injection at [41]. It demonstrated that ability of reliably inducing faults, which are in compliance with the ARC fault model, without the unit being damaged permanently. The concept presented at [26 and [44] is practically applied at [42], it demonstrated retrieval of the complete AES 128 key through analysis of 50 cipher-text or less, by fault injection analysis method.

At [32], we find a summary of quite a few methods of protection against timing attacks. The secret-key data is in strong correlation with the access of memory for look-up tables, for AES. It is sought to eliminate or atleast minimize this correlation by a number of implementations. It is suggested that look-up tables should completely be avoided and the logical implementation of AES be used instead. Alternatively, to eliminate the access of memory and the timing associated with it, the look-up table many be stored not in the memory but only in the registers. A small set or multi-copied table approach is also available, this enables changing access statistics and makes it harder to predict timings, for AES implementations.

Some of the other recommendation for "obfuscating" the memory access timing are :

1. Access of memory is implemented in such a manner that every entry from the concerned tables are read, in a pre-decided order, and only the one required is used.
2. Reading of 1 symbolic element from all of the memory blocks.
3. Shuffling of the memory content after being accessed or sometimes permute the memory and keep the cache locked while permuting.
4. Spurious accesses being added to the pattern of access of memory for adding noise.
5. Random latency or delays added to normalized access timings and thus hiding it.
6. Leakage being prevented by disabling of the simultaneous threads & interrupts during access of memory.
7. The cache capability being disabled.

Techniques for masking algorithms are also present (see Figure 1). Modification of the AES algorithm done in such a manner that mixing of a “mask” with the plain-text data, is allowed, before encryption or decryption which can be removed at a later stage to give the right results. Through this method the correlation amongst sensitive data and timing measurements can be removed. The mask may be a fixed value, a calculated value or any random value, whereas mixing may be done multiplicatively, or additively, or in a combined manner.

However there would be an impact on performance in all of the methods, even for hardware implementation. For high-performance application, we can minimize the impact by application of counter-measures to only the rounds that may be attacked.

Finally, induction of 7 new AES instruction has been announced by Intel in its newly launched “WestmereTM” processor line. They are designed to calculate the AES key-expansion-function, decryption and encryption round function in the constant time hardware-implementation. “As instruction don’t use look-up tables and are run in data independent time, they eliminate the main cache-based and timing attacks which threaten AES’s implementations based on tables.” [48]

Eliminating of correlation amongst sensitive data and power fluctuations is attempted by countermeasures for power analysis. One powerful scheme is DPA. It is unlikely that a method can provide complete guarantee particularly with software implementation. We have suggestions of cryptographic device being introduced with noise generating circuit and they appear to be a sound countermeasure, yet in real world DPA captures only a few more power traces to overcome the measures.

Research has been done and special standard ASIC cell libraries have been developed that don’t display power consumption based on data [46]. An alternate research area is applying the masking scheme based on hardware(Figure 1). Advantages of this technique is that if it is properly designed then it can lead to security against timing attacks as well as DAP, simultaneously. At [45], the AES’s hardware implementation was successfully masked with the performance dipping by just 40-50%.

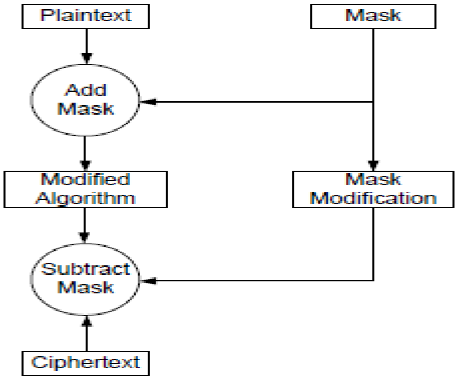

Figure 1: General Masking Mechanisms

There are quite a few countermeasures against “fault injection analysis” and they make an attempt at increasing the security of the cryptographic device security through minimizing or eliminating error detection. Most of the successful techniques employ a mechanism to detect errors, with intention of deactivating the cryptographic device in case a specific number errors are detected. This prevents the attacker from gathering enough data for launching a successful attack. We have Research focusing on achieving a balance between the assurance level and performance level while developing the error detection mechanism. Like, [43] and [47] focuses on applying the error detection code to the non-linear and linear algorithmic elements in order to achieve maximum possible coverage. The method provide high assurance levels but as a result performance is impacted considerably. Simple comparison between redundant implementations for detection of faults is suggested at [44]. The basis is that it’s not practically possible for the fault injection to induce both of the elements in an identical manner and while comparing the function’s output the differences can be detected. It is an effective method. The hardware implementation can be brought about with little or no impact on the performance and the only thing affected here is the cost of the hardware. There are also suggestions from other research works about how to prevent the attacker’s attempt to bypass its error detection mechanism.

E. Distinguishing and Related-Keys attack

The chosen plain-text differential attacks have been modified into a related-key attack. Multiple pair of plaintext is chosen by the attacker, with specified differences between each plaintext pair. Now the cipher is used as a black box oracle, and each plain-text is encrypted by 2 keys, with specified difference them (though the keys remain un-known); the attack is named for these related keys. The attacker discovers the unknown keys on the basis of information derived. Even though it is not likely to find related key, if the block cipher was being utilized for the purpose of encryption, however it is likely to find related key when the block cipher was being utilized as fragment of the cryptographic hash functions. The hash functions may then be broken by the related-key attacks.

Biryukov et al., in 2009 [49] circulated attack on on fullstrength AES-256 and AES-192 through related-key. Keys were recovered by the attacks with 2119 work for AES256 and 2176 work for AES-192. As the time taken by these attack is less compared to brute force, theoretically AES-256 and AES-192 are broken; but practically the time taken is too much. Yet, in [50] Biryukov et al. presented related-key attack upon lower round variant of AES-256 which can be termed as practical.

The attacker can technically detect absence of randomness in a block cipher under a distinguishing attack; the difference between the behavior of a typical random cipher and the block cipher is distinguished by the attacker. As cryptographic constructions’ security, especially for the hash function, is build upon the assumption that that the block ciphers is a typical random cipher, the distinguishing attack raises questions on the construction security.

At [51], [52] Biryukov et al. released distinguishing related-key attacks on AES-256 that required 2120 time. The distinguishing attacks were developed into key recovery attacks that required 2131 time and 265 memory. And like previously in theory these attacks also break fullstrength AES-256, but not in practice. At [53] Peyrin and Gilbert have reported a distinguishing attack with a known-key on AES-128 which was reduced to 8 round from the previous 10 round; this attack required 248 time and 232 memory and thus seems practical, and yet breaking the almost-full-strength variants of AES.

III. TACTICAL MILITARY APPLICATION- A SECURITY SUMMARY

Any development in cryptanalysis is a threat to the Government/Military organization from the encryption algorithm perspective. Especially if the research is made public shortly after being conducted. The concern is the advancement of the opponent and under the government /military threat model, intelligence agencies of other countries are the opponent. They have availability of state of the art resources, expertise and no lack of funds. Any piece of information that can give the opponent a political or military advantage is targeted. The standard cost tradeoff cannot be applied as this information is invaluable. We can safely assume that the opponent would not put a cap on the expense limit if the security system of targeted country can be compromised [54].

Thus, the government/military encryption solution should be able to withstand all possible cryptanalysis methods. The aim of AES design was to make it secured against linear and differential cryptanalysis along with the variations. Thus they do not pose much threats. Even the algebraic attack have not turned out to be practical threats in spite of having good theoretical results. There is a possibility of the Hybrid SAT solver /algebraic attack yielding results, but they still need a lot of research to be done. Even though we do not a clear and imminent threat, yet the approach should be of caution. As the related-keys attack is known to be successful against the AES when it is utilized in a hash function setup and thus not suggested for use in the military applications.

But the case of the Side channel attack is different as the recent research in this filed has made it come out a noteworthy threat and it should be kept in mind of the security applications implementers of the government/ military domains. Also, it is not advisable to have an AES software implementation done with low end processors in the case of applications in military communication. Even though there are measures proposed to counter the side channel attack, yet we are not sure if the low end processors would ever be able to constantly execute and distribute power uniformly [21]. The use of a hardware implementation with uniform power function/constant execution is highly suggested in all possible situations. The designer of the system should take measures to control the inessential information leakage from not just the encryption perspective but all other possible points as well. The physical access to the fielded system and its accessories like headsets and batteries must be limited. As any of them could be utilized by the attacker for gaining access to the system and monitor the parameters for launching an attack.

IV. CONCLUSION

The paper discussed the studies on the advancement of cryptanalysis research on AES. It aimed at identifying specific vulnerabilities and threats against the communication application in the military domain. The threat model of the military presents quite highly equipped opponent and lot more critical conditions faced against the opponent in comparison to any commercial domain.

We demonstrated that that progress is being in the field of cryptanalysis research against AES and it requires a great deal of caution as the major work is being carried out in the public domain. Vulnerabilities of AES against the different side channel attack were also discussed. Yet, if the available countermeasures are applied properly then the weaknesses can be negated at the hardware level. Steady progress is also seen in the alternate techniques like hybrid attack and algebraic attack etc., yet no reported breakthroughs are available. On the basis of the above we can say that the AES will not have the standard life span, which is expected out of an algorithmic suit that has obtained approval for applications of the classified domain (50 years [54]). As a result it can be stated that AES is not appropriate for being used in the strategic applications that have been classified. But Programmed cryptography is employed at the hardware in the strategic communication equipments. If there is a breakthrough in the public domain, a secure algorithm can be developed relatively faster and the length of vulnerability timeframe would be more dependent on logistic aspects rather than the technical aspects. However, the plan to handle such an inevitable situation is required.

REFERENCES

[1]. M. Matsui, “Linear Cryptanalysis Method for DES Cipher”, EUROCRYPT, LNCS 765, pp.386-397, Springer, 1994.

[2]. Ben-Aroya, E. Biham, "Differential Cryptanalysis of Lucifer", CRYPTO, Journal of Cryptology, pp.187-199, Springer, 1994.

[3]. D. Wagner, “The Boomerang Attack, Fast Software Encryption”, 6th International Workshop on Fast Software Encryption, LNCS 1636, Springer, 1999.

[4]. Biryukov, “The Boomerang Attack on 5 and 6-Round Reduced AES”, LNCS 3373, pp.11-15, Springer, 2005.

[5]. L. Knudsen, "Truncated and Higher Order Differentials", 2nd International Workshop on Fast Software Encryption, LNCS 1008, pp.196–211, Springer, 1994.

[6]. J. Daemen, L. Knudsen, V. Rijmen, "The Block Cipher Square", 4th International Workshop on Fast Software Encryption, LNCS 1267, pp. 149–165, Springer, 1997.

[7]. T. Jakobsen, L. Knudsen "The Interpolation Attack on Block Ciphers", 4th International Workshop on Fast Software Encryption, LNCS 1267, pp.28–40, Springer, 1997.

[8]. J. Daemen, V. Rijmen, "AES Proposal: Rijndael, Version 2", http://www.esat.kuleuven..ac.be/vijmen/rijndael, 1999.

[9]. N. Courtois, J. Pieprzyk, "Cryptanalysis of Block Ciphers with Overdefined Systems of Equations", ASIACRYPT, LNCS 2501, pp.267- 287, Springer, 2002.

[10]. Dinur, A. Shamir, "Cube Attacks on Tweakable Black Box Polynomials", EUROCRYPT, LNCS 5479, pp. 278-299, Springer, 2009.

[11]. B. Schneier, "Adi Shamir's Cube Attacks". http://www.schneier.com/blog/archives/2008/08/adi_shamirs_cub.html, August 19, 2008.

[12]. N. Courtois, A. Klimov, J. Patarin, A. Shamir, "Efficient Algorithms for Solving Overdefined Systems of Multivariate Polynomial Equations", EUROCRYPT, LNCS 1807, pp.392-407, Springer, 2000.

[13]. Kipnis, A. Shamir, "Cryptanalysis of the HFE Public Key Cryptosystem by Relinearization", CRYPTO, LNCS 1666, pp.19-30, Springer, 1999.

[14]. E. Filiol, "Plaintext-dependent Repetition Codes Cryptanalysis of Block Ciphers - The AES Case", http://eprint.iacr.org/2003/003, 2003.

[15]. N. Courtois, R. Johnson, P. Junod, T. Pornin, M. Scott, "Did Filiol Break AES?", http://eprint.iacr.org/2003/022, 2003.

[16]. N. Courtois, "How Fast can be Algebraic Attacks on Block Ciphers?", http://eprint.iacr.org/2006/168, 2006.

[17]. N. Courtois, "CTC2 and Fast Algebraic Attacks on Block Ciphers Revisited", http://eprint.iacr.org/2007/152, 2007.

[18]. O. Dunkelman, N. Keller, "Cryptanalysis of CTC2", CT-RSA, LNCS 5473, pp.226-239, Springer, 2009.

[19]. F. Massacci, L. Marraro, "Logical Cryptanalysis as a SAT-Problem: the Encoding of the Data Encryption Standard", Journal of Automated Reasoning, 24, pp.165-203, 2000.

[20]. F. Massacci, "Using Walk-SAT and Rel-SAT for Cryptographic Key Search", International Joint Conference on Artificial Intelligence, pp.290-295, Kaufmann, 1999.

[21]. N. Potlapally, A. Raghunathan, S. Ravi, N. Jha,, R. Lee, "Aiding Side-Channel Attacks on Cryptographic Software with Satisfiability-Based Analysis", IEEE Transactions on VLSI Systems, 15(4), pp.465-470, April 2007.

[22]. N. Courtois, G. Bard, "Algebraic Cryptanalysis of the Data Encryption Standard", IMA Int. Conf. Proceedings, LNCS 4887, pp.152-169, Springer, 2007.

[23]. P. Kocher, "Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems", CRYPTO, LNCS 1109, pp.104-113, Springer, 1996.

[24]. P. Kocher, J. Jaffe, B. Jun, "Introduction to Differential Power Analysis and Related Attacks", Tech. Rep., Cryptography Research Inc, 1998.

[25]. P. Kocher, J. Jaffe, B. Jun, "Differential Power Analysis", CRYPTO, LNCS 1666, pp.388-397, Springer, 1999.

[26]. D. Boneh, R. A. DeMillo, R. J. Lipton, "On the Importance of Checking Computations", EUROCRYPT, LNCS 1233, pp.37-51, Springer, 1997.

[27]. E. Biham, A.Shamir, "Differential Fault Analysis of Secret Key Cryptosystems", CS 0910, CRYPTO, LNCS 1294, pp. 513 – 525, Springer, 1997.

[28]. D. Asonov, R. Agrawal, "Keyboard Acoustic Emanations", IEEE Symposium on Security and Privacy, Oakland, CA, pp.3-11, 2004.

[29]. J.J. Quisquater, D. Samyde, "ElectroMagnetic Analysis (EMA): Measures and Counter-Measures for Smart Cards", Proceedings of the International Conference on Research in Smart Cards: Smart Card Programming and Security, pp.200–210, 2001.

[30]. D. J. Bernstein, "Cache-Timing Attacks on AES", http://cr.yp.to/antiforgery/cachetiming-20050414.pdf, April 2005.

[31]. J Bonneau, "Cache-Collision Timing Attacks Against AES", CHES 2006, 6th International Workshop, Yokohama, Japan, Oct. 2006.

[32]. D. Osvik, A. Shamir, E. Tromer, "Cache Attacks and Countermeasures: the Case of AES", CT-RSA, LNCS 3860, pp.1-20, Springer, 2006.

[33]. O. Acicmez, S. Gueron, J. Seifert, "New Branch Prediction Vulnerabilities in OpenSSL and Necessary Software Countermeasures", IMA Int. Conf. Proceedings, pp.185-203, 2007.

[34]. O. Acicmez, C. K. Koc, J. Seifert, "On the Power of Simple Branch Prediction Analysis", ASIACCS 2007, Singapore, pp.312-320, 2007.

[35]. O. Acicmez, C. Koc, J. Seifert, "Predicting Secret Keys via Branch Prediction", CT-RSA, LNCS 4377, pp.225-242, Springer, 2007.

[36]. J. Bonneau, "Robust Final-Round Cache-Trace Attacks Against AES", IACR Cryptology ePrint Archive, Report # 374, 2006.

[37]. J. Fournier, M. Tunstall, "Cache Based Power Analysis Attacks on AES", LNCS 4058, pp.17-28, Springer, 2006.

[38]. G. Boracchi, L. Breveglieri, "A Study on the Efficiency of Differential Power Analysis on AES S-Box", Technical Report 2007-17, DEI Politecnico di Milano, 2007.

[39]. K. Schramm, G. Leander, P. Felke, C. Paar, “A Collision-Attack on AES Combining Side Channel and Differential-Attack”, CHES 2004, 6th International Workshop, Cambridge, MA, USA, 2004.

[40]. O. Faurax, T. Muntean, “Security Analysis and Fault Injection Experiment on AES”, Proceedings of SAR-SSI 2007, 2007.

[41]. N. Selmane, S. Guilley, J-L Danger, “Practical Setup Time Violation Attacks on AES”, Dependable Computing Conference, pp.91-96, 2008.

[42]. P. Dusart, G. Letourneux, O. Vivolo, “Differential Fault Analysis on AES”, LNCS 2846, pp.293-306, Springer, 2003.

[43]. M. Medwed, “A Continuous Fault Countermeasure for AES Providing a Constant Error Detection Rate”, Cryptology ePrint Archive, Report 2009/119, http://eprint.iacr.org, 2009.

[44]. M. Joye, P. Manet, J. Rigaud, “Strengthening Hardware AES Implementations Against Fault Attacks”, IET Info Security 1(3), pp.106– 110, 2007.

[45]. N. Pramstaller, F. Gurkaynak, S. Haene, H. Kaeslin, N. Felber, W. Fichtner, “Towards an AES Crypto-Chip Resistant to Differential Power Analysis”, ESSCIRC, Leuven, Belgium, 2004.

[46]. S. Mangard, “Hardware Countermeasures Against DPA, A Statistical Analysis of Their Effectiveness”, CT-RSA, San Francisco, USA, 2004.

[47]. M. Karpovsky, K. Kulikowski, A. Taubin, “Differential Fault Analysis Attack Resistant Architectures for the Advanced Encryption Standard”, CARDIS ’04, Toulouse, France, Kluwer, pp.177–192, 2004.

[48]. S. Gueron, “Intel®’s Advanced Encryption Standard (AES) Instructions Set”, Intel Corporation, White Paper, http://software.intel.com/enus/articles/advanced-encryption-standard-aes-instructions-set, 2009.

[49]. Biryukov, D. Khovratovich, “Related-key Cryptanalysis of the Full AES-192 and AES-256”, ASIACRYPT, LNCS 5912, pp.1-18, Springer, 2009.

[50]. A. Biryukov, O. Dunkelman, N. Keller, D. Khovratovich, A. Shamir, “Key Recovery Attacks of Practical Complexity on AES Variants with up to 10 Rounds”, EUROCRYPT, Springer, 2010. https://www.cryptolux.org/mediawiki/uploads/3/38/Fast_attack_on_redu ced_AES-256.pdf, 2009.

[51]. Biryukov, D. Khovratovich, I. Nikolić, “Distinguisher and Related-Key Attack on the Full AES-256”, CRYPTO, LNCS 5677, pp.231-249, Springer, 2009.

[52]. Biryukov, D. Khovratovich, I. Nikolić, “Examples of Differential Multicollisions for 13 and 14 Rounds of AES-256”, https://www.cryptolux.org/mediawiki/uploads/f/f2/AES-256_nonrandomness_examples.pdf, 2009.

[53]. H. Gilbert, T. Peyrin, “Super-Sbox Cryptanalysis, Improved Attacks for AES-like Permutations”, Cryptology ePrint Archive Report 2009/531, November 2, 2009. http://eprint.iacr.org/2009/531.pdf.

[54]. M. Kurdziel, J. Fitton, “Baseline Requirements for Government & Military Encryption Algorithms”, Proc. IEEE, Mil. C