The Security of Advanced Encryption Standard

Joni Mönttinen
Abstract

UNIVERSITY OF EASTERN FINLAND

Faculty
Faculty of Science and Forestry

Department
Computer Science

Author
Joni Mönttinen

Title
Security of advanced encryption standard

Main subject
Computer Science
Level
Master
Date
16.2.2015
Number of pages
7

Abstract

Advanced encryption standard is a block cipher with 128, 192 or 256-bit key and 128-bit block size. U.S National institute of Standards and technology selected it as a successor to DES in 2001 because it was not considered secure enough anymore. This paper is an overview of what AES is and how secure it is by todays standards. The security analysis includes descriptions of three different ways to recover AES encryption key and how difficult it is to execute these attacks. This paper is a literature review and it is written to be easy to understand even for those who are not familiar with the subject. The conclusion is that there are no significant security threats AES but it is possible that a poor implementation may allow side-channel attacks.

Key words
Advance encryption standard, brute-force attack, cryptanalysis, biclique attack, side-channel attack
1. Introduction

Data security is increasingly important in these days since people have a lot of personal data and network communications that need to be kept safe. This is why data encryption is needed and it is important to make sure that commonly used encryption methods are secure. Unfortunately there is no way to provide absolutely perfect data security but it is possible to make sure that it is computationally infeasible to decrypt an encrypted message without correct key.

One of the great challenges in data security is the increasing computing power of computers. Old encryption methods become computationally insecure when computers can easily calculate the secret encryption keys. This is why DES (data encryption standard) were replaced by AES (Advanced encryption standard). DES had only 64 bit key which made it vulnerable even to the brute force attacks where computer checks all possible keys until it finds the correct one.

The paper is structured as follows. The second section describes briefly how the AES works without getting too much into technical details. Third section describes various cryptanalysis methods of AES and whether they cause significant security threat. The fourth section concludes.

2. Basics of AES

Advanced encryption standard is a symmetric-key cipher which means that encrypting and decrypting is done by the same key. It is also a block cipher and thus it operates on fixed-length groups of bits. All implementations of AES use fixed block size of 128 bits but the key size can be either 128, 192 or 256-bits. (Stallings, 2014) The AES is based on Rijndael cipher which was designed by Joan Daemen and Vincent Rijmen. The Rijndael cipher supports other key and block sizes such as 160 and 224 bits but AES does not include these. (Daemen & Rijmen, 2003)

AES encrypts each block of data in multiple rounds. The number of rounds depends on key length. If the key length is 128-bits then the AES encrypts all blocks in 10 rounds. 192 and 256-bit keys encrypt the blocks of data in 12 and 14 rounds. One of the problems of DES is that it only encrypts 32-bits each round while the block size is 64-bits. AES encrypts the entire 128-bit block of data in every round which is why AES has smaller number of rounds compared to DES which has 16 rounds. (Paar & Pelzl, 2009)
In each round of AES encryption the cipher makes four different transformations to the block of data. The transformations are: SubBytes, ShiftRows, MixColumns and AddRoundKey. Only exception is the final round which contains only three transformations since it does not have MixColumns operation. During these rounds each block of data is depicted as 4 x 4 byte matrix and also the key is divided to 4 x 4 byte subkeys. Each round gets these matrices as an input and produces 4 x 4 byte state matrix as an output. The last round produces 128-bit block of cipher text as output. (Stallings, 2014)

![Encryption process diagram](image)

**Figure 1. Encryption process**

### 3. Brute-force attack and cryptanalysis of AES

The primary goal of the attack against encryption system is to recover the key that was used to encrypt the data. With this key the attacker can decrypt all of the encrypted data. The attacker can either use brute-force attack or various cryptanalysis techniques to calculate the encryption key.
Brute-force attacks try to guess the key by trying all possible keys until it finds the correct one. Cryptanalysis attacks try to find weaknesses in the encryption algorithm in order to reduce the computational complexity of the key recovery. (Stallings, 2014) There are also side-channel attacks where attacker tries to find weaknesses from the physical implementation of the cryptosystem.

### 3.1 Brute-force attack

Brute-force attacks are very simple attacks and they can be used against every single cipher. With these attacks the goal is to go through every possible key permutation until it finds the key that can decipher the data into plaintext. (Paar & Pelzl, 2009) In worst case scenario the brute-force algorithm must try every single key permutation before correct one is found. In average case the algorithm must try atleast half of all possible key permutations. (Stallings, 2014)

The feasibility of the brute-force attack depends mostly on the size of the key space. This means that brute-force attacks can only work if cipher uses short keys since with longer keys the size of the key space is exponentially larger. A cipher with a key length of N bits has a key space of $2^N$. AES has a minimum key length of 128-bits so this means that it has atleast $2^{128} = 3.4 \times 10^{38}$ alternative keys. If we have a powerful computer that can calculate $10^6$ decryptions/μs then in worst case scenario it will take $5.4 \times 10^{18}$ years to find the correct key. (Stallings, 2014)

Due to the large key space and high computational complexity the brute-force attacks are not threatening to the security of AES. This cryptanalytic attack method highlights the importance of sufficient key length and the reason why DES is not sufficiently secure anymore. DES has only $2^{56} = 7.2 \times 10^{16}$ alternative keys and if we use the same computer as in previous example then it would take only 10.01 hours to find the correct key. (Stallings, 2014)

### 3.2 Biclique attack

In the meet-in-the-middle (MITM) attack the attacker needs pairs of plaintext and corresponding ciphertext. The attacker divides the cipher into two subciphers. One of the subciphers encrypts the plaintext and the other decrypts the corresponding cyphertext. The idea is make these subciphers to ”meet in the middle” by finding a correct key-pair. This method is ineffective against AES because it has a nonlinear key schedule. (Bogdanov, et al. 2011)
MITM attack can be enhanced by utilizing a biclique structure. Biclique or complete bipartite graph consists of two groups of vertices where every vertex of the first group is connected to all vertices in the second group. (ibid.)

Biclique attack is the most efficient cryptanalysis of AES but it is still only a marginal improvement from brute-force attack. It has a computational complexity of $2^{126.18}$ for 128-bit key while brute-force has complexity of $2^{128}$. If we use computer that can calculate $10^6$ decryptions/μs then biclique attack will take $1.5 \times 10^{18}$ years to calculate which is still infeasible to do. It would also require to store $2^{88}$ bits of data which equals to 38 trillion terabytes. (ibid.)

### 3.3 Side-channel attacks

Side-channel attacks does not target the vulnerabilities of encryption algorithms but instead they attempt to exploit the information leaks from the physical implementation of the cryptographic system. For example in timing attacks the attacker can gather timing information from target computer. This information tells the attacker the exactly how many clock cycles the encryption process has taken. With this information it is possible get the encryption key. Solution for this problem is to make the implementation of the AES to run in constant time. (Bernstein, 2005)

### 4. Conclusion

While there are some theoretical attacks against AES they are all infeasible to execute. Even though the computing power of computers doubles every one and half years it would still take decades before AES becomes computationally insecure. It seems like the side-channel attacks are the most
significant security threat. It is very important that software developers take these types security issues into account when implementing AES encryption. While there is no way to achieve perfectly secure data encryption it is safe to say that AES secure enough for all data security needs.

References


Christof Paar, Jan Pelzl. ”Understanding Cryptography, A Textbook for Students and Practitioners”. Springer 2009

Andrey Bogdanov, Dmitry Khovratovich, Christian Rechberger. ”Biclique Cryptanalysis of the Full AES”. Microsoft Research Redmond 2011

Daniel J. Bernstein. ”Cache-timing attacks on AES”. 2005