Fine Tuning the Advanced Encryption Standard (AES)

Behnam Rahnama
European Univ. Lefke
Gemikonagi, Mersin 10
TRNC, Turkey
behnam@brahnama.com

Atilla Elci
Aksaray University
Aksaray, Turkey
atilla.elci@gmail.com

Ibukun Eweoya
European Univ. Lefke
Gemikonagi, Mersin 10
TRNC, Turkey
ewesther2000@yahoo.com

ABSTRACT.

The Advanced Encryption Standard has been playing a prominent role in embedded systems security for a decade after being announced by the National Institute of Standards and Technology (NIST). However, vulnerabilities have emerged, especially timing attacks, that challenges its security. This paper demonstrates the introduction of a unique diffusion and confusion scheme in Rijndael by incorporating ASCII codes manipulations using playfair ciphering into the algorithm; it is not dependent on the key and input thereby making it a constant time module in AES algorithm. The concept counters possible leakages from the S-box lookups; intermediary operations (SubstituteByte, ShiftRows, MixColumns, AddRoundKey) of the AES are still applicable but it becomes impossible for cryptanalysis discovery of enciphering method and ciphertext bits. Success of cracking efforts will be beyond human patience as it avoids statistical precision, thereby curbing timing attacks.

Categories and Subject Descriptors.

E.3 Data Encryption – Private key cryptosystems.
D.4.6 Security and Protection (K.6.5) – Authentication, cryptographic controls, verification.

General Terms.

Algorithms, design, security, reliability.

Keywords.

Rijndael, state, plaintext, ciphertext, s-box, ASCII, MixColumns, ShiftRows, SubBytes, AddRoundkey

1. INTRODUCTION

Surprising growth of Internet and the fiery growth in computer networks have increased the dependence of both organizations and individuals on the information stored and disseminated by these systems. However, at the same time it also brought about a plethora of new issues and concerns, the utmost among them being the need to protect data and resources from unwarranted disclosure, guarantying the authenticity of data and messages, also protecting systems from network based attacks [5].

Previously, encryption algorithms such as Data Encryption Standard (DES) had been efficient to handle most security needs, but apparently DES's time is gone. With the limitations of DES's 56-bit key and the advent of faster computers, DES could no longer be considered a secure algorithm. Subsequently, Triple DES which uses 168-bit key has taken the place of DES in an attempt to enhance security without compromising currently accepted encryption standards.

Triple-DES is simply too slow and with the extinction of DES, NIST requested submissions of stronger encryption algorithms in substitution to outdated DES with the following requirements: It must be a block cipher with longer key length, larger block size, fast in computation and greater flexibility. After several rounds of submissions and eliminations, the AES algorithm also known as Rijndael algorithm was selected.

The Advanced Encryption Standard (AES) was published in 2001 by the National Institute of Standards and Technology (NIST) as the successor of the Data Encryption Standard (DES). The AES is a symmetric key encryption standard used in embedded systems, e-business, telecommunications, and all human endeavors where security is required with considerable speed.

The standard has three block ciphers, AES-128, AES-192, and AES-256, adopted from a larger collection originally published by Rijndael. Each of these ciphers has a 128-bit block size, with key sizes of 128, 192, and 256 bits respectively. The AES ciphers have been analyzed extensively and are now used worldwide as was the case of its predecessors, the Data Encryption Standard (DES) [1].

Joan Daemen and Vincent Rijmen are the developers of Rijndael who submitted it to the AES selection process [2]. The AES is a standard block cipher. It encrypts and decrypts data with a secret key and not two as found in DES, using a combination of primitives such as MixColumns, ShiftRows and SubBytes over many rounds (10 for a 128-bit key). A common optimization technique on 32-bit processors is to precompute series of tables on the basis of the combination of these primitives. AES encryption then becomes a series of table lookups and XOR operations [3].

It has been observed that the index for these AES tables is the XOR of a plaintext byte and a key byte, therefore the indices themselves must remain secret. However, a spy process running on the same system can observe the variable timing of the AES encryption due to cache behavior, narrowing down the possible values for the key [4].

SIN '12, October 22 - 26 2012, Jaipur, India