Evaluating the performance of Symmetric Key Algorithms: AES (Advanced Encryption Standard) and DES (Data Encryption Standard)

Shanta¹, Jyoti Vashishtha²
¹²Guru Jambheshwar University of Science & Technology, Hisar,(Haryana) India
shanta2010.engg@gmail.com, jyoti.vst@gmail.com

Abstract
Encryption algorithms are known to be computational intensive. Internet and networks applications growing very fast, so the needs to protect. Encryption plays the very important role in information security system. On the other side, the symmetric key algorithms consume a significant amount of computed resources such as speed, time, memory and cost. The symmetrical algorithms AES and DES as a security enhancement. Current AES and DES standards are applied to encrypt and to protect. This paper provides evaluation of the most common encryption algorithms namely: AES (Rijndael) (Advanced Encryption Standard) and DES (Data Encryption Algorithm). A comparison has been conducted for those encryption algorithms at different setting speed, time and cost. Our results show that AES is more suitable than DES.

Keywords: Encryption techniques, DES, AES, computer security

1. INTRODUCTION

The encryption algorithms are widely available and used in information security. They can be categorized into Symmetric (private) and Asymmetric (public) keys encryption. In Symmetric keys encryption or secret key encryption, only one key is used to encrypt and decrypt data. The key should be distributed before transmission between entities. Keys play an important role. If weak key is used in algorithm then every one may decrypt the data. Strength of Symmetric key encryption depends on the size of key used. There are many examples of strong and weak keys of cryptography algorithms like RC2, DES, 3DES, RC6, Blowfish, and AES. RC2 uses one 64-bit key. DES uses one 64-bits key. Triple DES (3DES) uses three 64-bits keys while AES uses various (128,192,256) bits keys.

Asymmetric key encryption or public key encryption is used to solve the problem of key distribution. In Asymmetric keys, two keys are used; private and public keys. Public key is used for encryption and private key is used for decryption (E.g. RSA and Digital Signatures). Because users tend to use two keys: public key, which is known to the public and private key which is known only to the user. There is no need for distributing them prior to transmission efficient for small mobile devices [1]. Asymmetric encryption techniques are almost 1000 times slower than Symmetric techniques, because they require more computational processing power [2].

This paper examines a method for evaluating performance of selected symmetric encryption of various algorithms. This study evaluates two different encryption algorithms namely: AES, DES.

2. Implementation Algorithms:

The following secret key encryption algorithms were chosen for implementation, i.e
- DES
- AES
A. Data Encryption Standard (DES)

DES (Data encryption standard) is currently the most widely used block cipher in the world. In May 1973, NIST (then NBS) called for possible encryption algorithms for use in unclassified adoption. The algorithm is in many standards around the world (e.g. Australian standard AS2805.5-1985). One of the largest users of the DES is in the banking industry. DES is (64 bits key size with 64 bits block size). Since that time, many attacks and methods recorded the weaknesses of DES, which made it an insecure block cipher [3], [4].

DES HISTORY

DES is the Data Encryption standard. It is a block cipher defined and endorsed by the US government in 1977 as an official standard. The DES has undergone general public scrutiny since its publication and is the most well-known and widely used as a commercial and a secret key cryptosystem in the world. DES algorithm has also been adopted by the American National Standards Institute (ANSI) as US national standard. The DES is working into 16-round Feistel-ladder cipher with a 64-bit block size. It encrypts a 64-bit input plaintext into a 64-bit output cipher text using a 64-bit key. The 64-bit key contains 56 independent key bits, which determine the exact cryptographic transformation, and 8 bits that may be used as parity bits for error detection. As well as The Data Encryption Standard (DES) is an example of a conventional cryptography is widely employed by the Federal Government.

DES Algorithm

Actually the DES is a 16-iteration (round) cipher, with 64-bit block size. It encrypts a 64-bit input plaintext into a 64-bit output cipher text using a 64-bit key. The 64-bit key contains 56 independent key bits, which determine the exact cryptography transformation, and 8 bits that may be used as parity bits for error detection. After while will see how DES dose work, as well as explain these iteration (rounds). The DES algorithm enciphers and deciphers data in 64-bit blocks under the control of a 56-bit key. Some of the most important requirement was:

- The algorithm had to be available to all users on the royalty free basis.
- The algorithm had to be adaptable for use in diverse applications.
- The algorithm had to be economically implementable in electronic devices

![Fig 2. Bound level steps in DES](image)

B. Advanced encryption standard (AES)

In 1997 the National Institute of Standards and Technology (NIST), a branch of the US government, started a process to identify a replacement for the Data Encryption Standard (DES). It was generally recognized that DES was not secure because of advances in computer processing power. The goal of NIST was to define a replacement for DES that could be used for non-military information security applications by US government agencies. Of course, it was recognized that commercial and other nongovernmental users would benefit from the work of NIST. NIST has released all submissions and unclassified analyses. The AES candidates are the latest generation of block ciphers, and have a significant increase in the block size - from the old standard of 64-bits up to 128-bits; and keys from 128 to 256-bits. In part this has been driven by the public demonstrations of exhaustive key searches of DES (64-bits)
The rules which is use in AES

The algorithm must be a symmetric block cipher.

- The full design must be public.
- The key length of 128, 192 and 256 bit must be support.
- Both software and hardware implementation must be possible.
- The algorithm must be public on nondiscriminatory terms.

CHARACTERISTICS OF AES AND DES

AES algorithm is an iterative block cipher performing encryption and decryption in fixed size blocks. The incoming data and key are stored in a matrix, called state matrix, and all operations are performed over the state matrix [8][9]. There are three different input lengths for data and key length, which are 128, 192, 256 bits. Each iteration is called a round and the round number is changed depending on the data and key length. Byte Substitution, Shift Row, Mix Column and Add Round Key transformations are the four main transformations in one round [10]. In Byte Substitution the State byte is replaced with a substitution table element, which is calculated with nonlinear transformations in GF (28). In Shift Row, rows of the State matrix are shifted to the right cyclically. For each data length and for each State matrix row, there is a different shift offset value. Mix Column transformation acts independently on every column of the state. Each column of the State matrix is considered as a four-term polynomial over GF (28) and multiplied with a fixed polynomial. Add Round Key is applied to the State by a simple bitwise EXOR operation. Decryption process is the inverse operation of the encryption process and the transformations in the encryption round are also reversed in the mean of the sequence.

DES is a symmetric crypto algorithm, which operates on 64-bit block size within 16 rounds. The input plain text and the output ciphered text are 64-bit. The input key data length is also 64-bit, but only the 56 bits of the whole key data is effective [11]. The remaining 8 bits have no effect on the encryption/decryption process of the DES. The main operations are bit permutations and substitution in one round of DES. There are six different permutation operations, which are used both in Key Expansion part and cipher part. The main operations in DES algorithm, like key related operations and SBox operations are performed in the Cipher part. Decryption of DES algorithm is similar like encryption, but only the round keys are applied in reverse order.

Microsoft visual studio:

Microsoft Visual Studio 2010 is an integrated development environment (IDE) from Microsoft. It is used to develop console and graphical user interface applications along with Windows Forms applications, web sites, web applications, and web services in both native code together with managed code for all platforms supported by Microsoft Windows, Windows Mobile, Windows CE, .NET Framework, .NET Compact Framework and Microsoft Silver light.

Visual Studio includes a code editor supporting IntellISense as well as code refractory. The integrated debugger works both as a source-level debugger and a machine-level debugger. Other built-in tools include a forms designer for building GUI applications, web designer, class designer, and database schema designer. It accepts plug-ins that enhance the functionality at almost every level—including adding support for source-control systems (like Subversion and Visual SourceSafe) and adding new toolsets like editors and visual designers for domain-specific languages or toolsets for other aspects of the software development lifecycle (like the Team Foundation Server client: Team Explorer).

Visual Studio supports different programming languages by means of language services, which allow the code editor and debugger to support (to varying degrees) nearly any programming language, provided a language-specific service exists. Built-in languages include C/C++ (via Visual C++), VB.NET (via Visual Basic .NET), C# (via Visual C#), and F# (as of Visual Studio 2010). Support for other languages such as M, Python, and Ruby among others is available via language services installed separately. It also supports XML/XSLT, HTML/XHTML, JavaScript and CSS. Individual language-specific versions
of Visual Studio also exist which provide more limited language services to the user: Microsoft Visual Basic, Visual J#, Visual C#, and Visual C++. [12], [13].

**CryptoTools**

CryptoTools is a Macintosh application program that enables you to encrypt and decrypt text files using a variety of simple and historically interesting ciphers. Encrypting operations require that the user create an appropriate key, and decrypting operations require the user supply the correct key. The following ciphers are implemented. Clicking on their links gets you additional information about the cipher method itself. CryptoTool provides a dialog boxes to assist with creating and using keys. CryptoTool also provides a number of Helpers -- cryptanalysis tools that may be helpful in "cracking" encrypted files.

CryptoTools is a multi-language suite of encryption components and libraries. CryptoTools provides DES and Triple DES (3DES) encryption, as well as Base64 encoding and MD5 hashing capabilities. CryptoTools supports the following major languages C (win32), C++, ActiveX/COM, .Net, and Java. Components for all languages are available in one single package.

Wodcrypt is the component that provides strong encryption for your applications. It provides purpose for analysis and solution of the problem. It provides the encryption and decryption messages to users who use to PGP, OpenPGP, SECEXMail, and many more. For this purpose it may be use other tools like wodcrypto2.1.4. It provides supports for most common crypto algorithms, such as AES, DES, TripleDES, BLOWFISH, CAST, RC2, RC4, RC5, for Symmetric Encryption and decryption RSA, DSA For making and verification signatures, and SHA1.

**Implementation Result**

For our experiment, we use a laptop 2GB Memory, in which performance data is collected. In the experiments, the laptop encrypts different algorithms of AES and DES. We use the Microsoft Visual Studio and VC++ that support wodcrypto- tool in which performance data is collected. In the experiments, AES and DES algorithms are encrypted in .NET and wodcrypto- tool. The wodcrypto-tool use the 4.30MB size, 2.1.4 version, 128MB RAM Windows 7/Vista/XP/2000 system

Several performance metrics are collected:

1. Speed
2. Time
3. Cost
4. Memory

1) **Speed**

When evaluating a system, it is often important to know how much performance gain is achieved by parallelizing a given application over a sequential implementation. Speed is defined as the ratio of the time taken to solve a problem on a single processor to the time required to solve the same problem on a parallel computer with p identical processors. The speed is denoted by the symbol S. Therefore,

\[ S = \frac{1}{T} \]

Formally, the speed S is defined as the ratio of the serial run time. The speed is decrease when the key length of the string increase

2) **Time**

The encryption time is the time that an encryption algorithm takes to produce a plaintext to ciphertext The time to be considered that the time of the AES algorithm in c# is better than that the AES algorithm of c++ in VC++. The time is considered in the millisecond. Like this the time of DES algorithm in c# is better than that the DES algorithm of c++.

3) **Cost**

The cost of the encryption/ decryption algorithm of AES and DES is depending on the two things:

1. Time
2. Memory

4) **Memory**

When evaluating the memory in the system it is important to check current Memory usage on a Windows 7/Vista computer, right click on Task Bar (Task Bar is the Grey/Blue bar at the bottom of the screen) then left click on “Start Task Manager”. You will see “Windows Task Manager” window. Task Manager shows applications currently running on your computer. We click on “Performance” tab within “Windows Task Manager”. This screen shows Memory and CPU usage statistics of your computer.

To keep this guide simple, we will note:

- “Total” under Physical Memory; which in this case is 2047MB or 2GB.
- "Memory" Section shows the amount of Memory computer is currently using; which in this case is 1.13GB.
"Commit (MB)" value in right side column shows the total of Physical RAM and Virtual Memory in MB (Mega Bytes) currently in use; which in this case is 1619MB or 1.58GB.

Results

Encryption Throughput

In figure (4, 5, 6), we show the encryption throughput of DES in c# and c++ on the basis of time, speed, memory. The encryption is performed with different data. The time is increase in c# and decrease on c++ when using the different packet size. However, with increasing the packet size, the throughput of DES in c# is far better than that DES in c++. This shows that DES algorithm on c# is more efficient than DES algorithm on c++. With the different key size, Time is increase and speed, memory is decrease in c#. The performance of DES in c# is giving the better and more efficient result.

Results of DES in c++

In figure (7, 8, 9), we show the performance of DES on the basis of Time, Speed, Memory. If the packet size is increase then the Time is decrease, and the Speed and Memory is increase when we increase the packet size in c++.
Results of the AES algorithms in c#

In figure (10, 11, 12), we show the performance of AES with c# and c++ in terms of cost, time, speed, memory. The encryption time is the encryption algorithms takes to produce a cipher text from the plaintext. Encryption time is used to calculate the throughput of an encryption scheme that indicates the speed of encryption. The throughput of the plaintext in bytes encrypted divided by the encryption time. When we are using the large packet size, the Time is increase and Speed, Memory are decrease in c#. A longer key size would provide stronger security of data.

Result of the AES algorithms in c++

In figure (13, 14, 15), we show the performance of AES on the basis of Time, Speed, Memory. If the packet size is increase then the Time is increase and the Speed and Memory is increase when we increase the packet size in c++. The AES algorithm in c++ takes more time as comparison the AES algorithm in c#. It takes the less memory as comparison the AES algorithm in c++ for encryption and decryption.
3. Conclusion:

This paper presents a performance evaluation of AES and DES symmetric encryption algorithms. The performance metrics of the encryption throughput, speed, time and cost. The AES encryption/decryption algorithm in c# in Microsoft visual studio is give the better results. Like the AES algorithm, DES algorithm in c# in Microsoft visual studio is giving the better results.

The performance of the selected algorithms of AES/DES in c++ which run in VC++ .net give the better result in required keys like speed, time and cost. It found that AES/DES algorithms in c++ has disadvantage over other algorithms in terms of time consumption. Also, we find that AES/DES algorithms still has low performance compared to the AES/DES algorithms in c#. Finally-in the case of speed, time and cost the AES/DES algorithms in c# which run in Microsoft visual studio give the better results.

References