Scientific and Practical Cyber Security Journal (SPCSJ) 3(1): 21-28 ISSN 2587-4667 Scientific Cyber Security Association (SCSA)

SYMMETRIC AND HOMOMORPHIC ENCRYPTION ALGORITHMS IN CLOUD DATA SECURITY

Tinatin Mshvidobadze
Associate professor - Gori State University, Georgia

ABSTRACT: Internet and networks applications are growing very fast, so the needs to protect such applications are increased. Encryption algorithms play a main role in information security systems. On the other side, those algorithms consume a significant amount of computing resources such as CPU time, memory, and battery power. This paper provides evaluation of six of the most common encryption algorithms namely: AES, DES, 3DES, RC2, Blowfish, and RC6. A comparison has been conducted for those encryption algorithms at different settings for each algorithm such as different sizes of data blocks, different data types, battery power consumption, different key size and finally encryption/decryption speed. Experimental results are given to demonstrate the effectiveness of each algorithm.

KEYWORDS: 3DES, AES, blowfish, computer security, DES, encryption techniques, RC2, RC6.

In the post-Snowden era, the significance of data security and privacy, as key selection criteria for cloud-infrastructure providers, has risen considerably [1]. To make it easier for organizations to outsource their communication solutions, Ericsson's approach is to push standardization, so that end-to-end protection of content can be combined with hop-by-hop protection of less sensitive metadata [2]. Many cloud-storage providers have adopted client-side encryption to prevent unauthorized access or modification of data, which solves the issues surrounding secure storage and forwarding for cloud data.

Data encryption has other benefits; in many jurisdictions users need to be informed of data breaches unless their information was encrypted. However, encryption does not necessarily mean better compliance with privacy regulations.

Many encryption algorithms are widely available and used in information security [3, 4, 5]. 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. 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 ECC). Public key encryption is based on mathematical functions, computationally intensive and is not very efficient for small mobile devices [6, 7, 8]. 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. Blowfish uses various (32-448); default 128bits while RC6 is used various (128,192,256) bits keys [9, 10, 11, 12, 13].

This paper examines a method for evaluating performance of selected symmetric encryption of various algorithms. Encryption algorithms consume a significant amount of computing resources such as CPU time, memory, and battery power. Battery power is subjected to the problem of energy consumption due to encryption algorithms. Battery technology is increasing at a slower rate than other technologies. This causes a "battery gap" [14,15]. This study evaluates six different encryption algorithms namely; AES, DES, 3DES, RC6, Blowfish, and RC2. The

performance measure of encryption schemes will be conducted in terms of energy, changing data types - such as text or document, Audio data and video data-power consumption, changing packet size and changing key size for the selected cryptographic algorithms.

Identity and attribute-based encryption

Homomorphic encryption is one of the key breakthrough technologies resulting from advances in cryptographic research. In contrast to AES, for example, this approach allows operations to be performed directly on encrypted data without needing to access data in its decrypted form. Unfortunately, fully homomorphic encryption, which includes methods that allow arbitrary computations on encrypted data, have yet to overcome some performance issues. However, a number of specialized methods like partially homomorphic encryption, deterministic encryption, order-preserving encryption, and searchable encryption allow a specific set of computations to be performed on encrypted data, with a sufficient level of performance so that they can be applied to real-life scenarios. By combining these methods, it is possible to cover many types of computations that arise in practice. For example, different proofs of concept have shown that by combining encryption methods, typical SQL operations such as SUM, GROUP BY, and JOIN can be carried out on encrypted databases [16]. Many computations, best outsourced to the cloud, use a restricted set of operations that can be dealt with using these specialized methods with good performance. For example, sums, averages, counts, and threshold checks can be implemented. However, further research is needed to make these methods applicable to real-world use cases. For example, data encryption performance is crucial for use cases with high data throughput. Ericsson's research [17] into the encryption performance of the most popular partially homomorphic cryptosystem (the Paillier system) has shown a performance increase of orders of magnitude, which makes Paillier suitable for high-throughput scenarios.

Specialized methods, like homomorphic encryption, used for carrying out computations on encrypted data, could also be used for preserving confidentiality in cloud computation and analytics-as-a-service. With these methods, clients with large datasets to be analyzed – such as network operators, health care providers, and process/engineering industry players – would be able to outsource both storage and analysis of the data to the cloud service provider. Once outside the client's network, data is encrypted, thereby preserving confidentiality, and allowing the cloud provider to perform analytics directly on the encrypted data.

Strong cryptography alone does not work without proper key management. Specifically, management covers how keys are generated and distributed, and how authorization to use them is granted.

Protecting data exchange between n endpoints using symmetric key cryptography requires the secure generation and distribution of roughly n^2 pair-wise symmetric keys. With the breakthrough invention of public key cryptography in the works of Diffie, Hellman, Rivest, Shamir, and Adleman in the mid-1970s, the use of asymmetric key pairs reduced the quadratic complexity, requiring only n key pairs. However, this reduction in the number of keys is offset by the need to often ensure that the public portion of the key pair can be firmly associated with the owner of its private (secret) portion. For a long time, a Public Key Infrastructure (PKI) was the main way to address this issue. But PKIs require management and additional trust relations for the endpoints and are not an optimal solution.

Identity-Based Encryption (IBE) allows an endpoint to derive the public key of another endpoint from a given identity. For example, by using an e-mail address (name.surname@company.com) as a public key, anyone can send encrypted data to the owner of the e-mail address. The ability to decrypt the content lies with the entity in possession of the corresponding secret/private key – the owner of the e-mail address – as long as the name space is properly managed.

Attribute-Based Encryption (ABE) takes this idea further by encoding attributes, for example, roles or access policies, into a user's secret/private keys. IBE and ABE allow endpoints without network connections to set up secure and authenticated device-to-device communication channels. As such, it is a good match for public safety applications and used in the 3GPP standard for proximity-based services for LTE.

Post-quantum cryptography

Although the construction of quantum computers is still in its infancy, there is a growing concern that in a not too distant future, someone might succeed in building much larger quantum computers than the current experimental constructions. This eventuality may have dramatic consequences for cryptographic algorithms and their ability to maintain the security of information. Attack algorithms have already been invented and are ready for a quantum computer to execute on.

For symmetric key cryptography, Grover's algorithm is able to invert a function using only $\sqrt{N}$ evaluations of the function, where N is the number of possible inputs. For a symmetric 128-bit key algorithm, such as AES-128, Grover's algorithm enables an attacker to find a secret key 200 quintillion times faster, using roughly 2^{64} evaluations instead of 2^{128} – the complexity of an exhaustive search. Quantum computing therefore weakens the effective security of symmetric key cryptography by half. Symmetric key algorithms that use 256-bit keys such as AES -256 are, however, secure even against quantum computers.

The situation for public-key algorithms is worse; for example, Shor's algorithm for integer factorization directly impacts the security of RSA. This algorithm is also effective in dealing with all other standardized public-key crypto systems used today. With Shor's algorithm, today's public-key algorithms lose almost all security and would no longer be secure in the presence of quantum computing. Figure 1 shows the effect of quantum computing on today's algorithms.

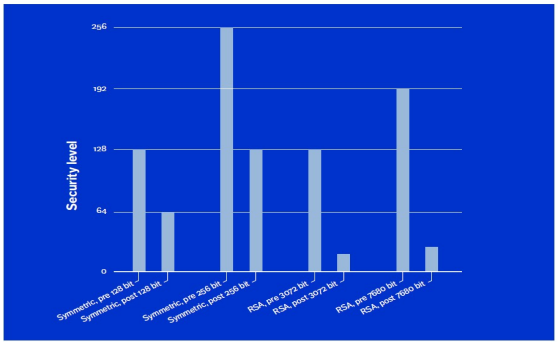

Figure1: Relative complexities for breaking cryptographic algorithms before quantum computers and post-quantum computers.

Although current research is far from the point where quantum computing can address the size of numbers used today in crypto schemes, the ability to perform quantum computing is increasing. The largest number factored by a quantum computer used to be the integer 21 (3 × 7),

but in 2014, a quantum computer factored 56,153 (233 × 241). The term post-quantum cryptography (PQC) is used to describe algorithms that remain strong, despite the fledgling capabilities of quantum computing. In 2014, ETSI organized a workshop on quantum-safe cryptography, and in 2015 the US National Security Agency (NSA) said [18] it would initiate a transition to quantum-resistant algorithms. The potential impact of quantum computing has reached the level of industry awareness.

The challenge for new schemes is to find solutions that have the same properties, such as non-repudiation, that digital signatures have today or provide data integrity with public verification. From this perspective, the blockchain construction used in Bitcoin is interesting. Although Bitcoin itself is not quantum immune, there is an interesting ingredient in its construction: when the chain has grown long enough, the integrity of hash value does not rely on verification against a digital signature but by having it endorsed by many users. By creating a public ledger, any tampering of a hash value is revealed by comparing it with the public value. The idea of a public ledger is significant in the KSI solution [19] for data integrity available in Ericsson's cloud portfolio. Yet the search for PQC schemes that can provide digital signatures with non-repudiation continues.

Today's systems that use or introduce symmetric schemes, should be designed with sufficient margin in key size, so they can cope with the potential capability of quantum computers. However, just as advances have been made in the fields of computer engineering and algorithm design over the past half-century, developers may well bring us new cryptographic schemes that will change the security landscape dramatically.

Symmetric Encryption Algorithms

This study evaluates six different encryption algorithms namely; AES, DES, 3DES, RC6, Blowfish, and RC2. The performance measure of encryption schemes will be conducted in terms of energy, changing data types - such as text or document, Audio data and video data-power consumption, changing packet size and changing key size for the selected cryptographic algorithms.

It is discusses the results obtained from other resources. It was shown in [20] that energy consumption of different common symmetric key encryptions on hand held devices. It is found that after only 600 encryptions of a 5 MB file using Triple-DES the remaining battery power is 45% and subsequent encryptions are not possible as the battery dies rapidly.

It was concluded in [21] that AES is faster and more efficient than other encryption algorithms. When the trans-mission of data is considered there is insignificant difference in performance of different symmetric key schemes. Even under the scenario of data transfer it would be advisable to use AES scheme in case the encrypted data is stored at the other end and decrypted multiple times. A study in [22] is conducted for different popular secret key algorithms such as DES, 3DES, AES, and Blowfish.

They were implemented, and their performance was com-pared by encrypting input files of varying contents and sizes. The algorithms were tested on two different hard-ware platforms, to compare their performance. They had conducted it on two different machines: P-II 266 MHz and P-4 2.4 GHz. The results showed that Blowfish had a very good performance compared to other algorithms.

Also it showed that AES had a better performance than 3DES and DES. It also shows that 3DES has almost 1/3 throughput of DES, or in other words it needs 3 times than DES to process the same amount of data [23].

In a study of security measure level has been pro-posed for a web programming language to analyze four Web browsers. This study consider of measuring the performances of encryption

process at the programming language's script with the Web browsers. This is followed by conducting tests Experimental in order to obtain the best encryption algorithm versus Web browser.

Experimental Design and results

In this experiment, was used a laptop IV 2.4 GHz CPU, in which performance data is collected. In the experiments, the laptop encrypts a different file size ranges from 321 K byte to 7.139Mega Byte139MegaBytes for text data, from 33 Kbytes to 8262 Kbytes for audio data, and from 4006 Kbytes to 5073 Kbytes for video files.

Several performance metrics are collected: 1) Encryption time; 2) CPU process time; and 3) CPU clock cycles and battery power.

The encryption time is considered the time that an encryption algorithm takes to produce a cipher text from a plaintext. Encryption time is used to calculate the throughput of an encryption scheme. It indicates the speed of encryption. The throughput of the encryption scheme is calculated as the total plaintext in bytes encrypted divided by the encryption time [24].

The following tasks that will be performed are shown as follows:

- A comparison is conducted between the results of the selected different encryption and decryption schemes in terms of the encryption time at two different encoding bases namely; hexadecimal base encoding and in base 64 encoding.
- A study is performed on the effect of changing packet size at power consumption during throughput for each selected cryptography algorithm.
- A study is performed on the effect of changing data types - such as text or document, audio file, and video file - for each cryptography selected algorithm on power consumption.
- A study is performed on the effect of changing key size for cryptography selected algorithm on power consumption. each algorithm in. As the throughput value is increased, the power consumption of this encryption technique is decreased.

Encryption of Different Packet Size

Encryption time is used to calculate the throughput of an encryption scheme. The throughput of the encryption scheme is calculated by dividing the total plaintext in Megabytes encrypted on the total encryption time for each algorithm in. As the throughput value is increased, the power consumption of this encryption technique is decreased.

Experimental results for this compassion point are shown Figure 2 at encryption stage. The results show the superiority of Blowfish algorithm over other algorithms in terms of the processing time. Another point can be noticed here; that RC6 requires less time than all algorithms except Blowfish. A third point can be noticed here; that AES has an advantage over other 3DES, DES and RC2 in terms of time consumption and throughput. A fourth point can be noticed here; that 3DES has low performance in terms of power consumption and throughput when compared with DES. It always requires more time than DES because of its triple phase encryption characteristics. Finally, it is found that RC2 has low performance and low throughput when compared with other five algorithms in spite of the small key size used.

Decryption of Different Packet Size

Experimental results for this compassion point are shown Figure 3 decryption stage. It is Possible find in decryption that Blowfish is the better than other algorithms in throughput and power consumption. The second point should be noticed here that RC6 requires less time than all algorithms except Blowfish. A third point that can be noticed that AES has an advantage over

other 3DES, DES, RC2.The fourth point that can be considered is that RC2 still has low performance of these algorithm. Finally, Triple DES (3DES) still requires more time than DES.

The Effect of Changing Key Size of AES, And RC6 on Power Consumption The last performance comparison point is changing different key sizes for AES and RC6 algorithm. In case of AES, the three different key sizes possible i.e., 128-bit, 192-bit and 256-bit keys. In case of AES it can be seen that higher key size leads to clear change in the battery and time consumption. It can be seen that going from 128-bit key to 192-bit causes increase in power and time consumption about 8% and to 256-bit key causes an increase of 16% [25].

Also in case of RC6, the three different key sizes possible i.e., 128-bit, 192-bit and 256-bit keys. In case of RC6 higher key size leads to clear change in the battery and time consumption.

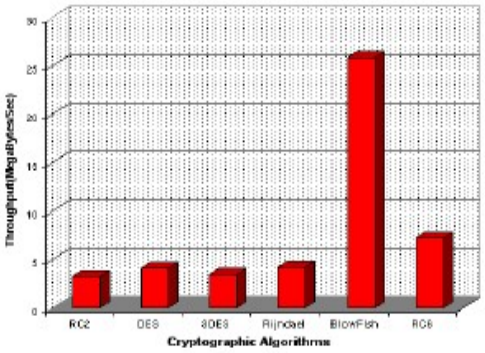

Figure 2: Throughput of each encryption algorithm (Megabyte/Sec)

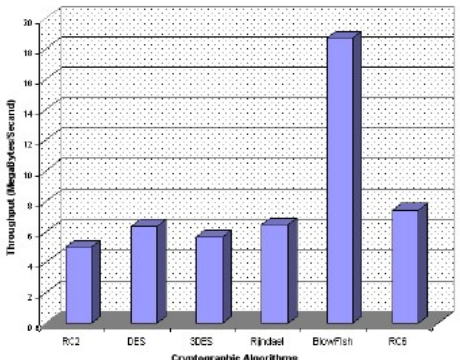

Figure 3: Throughput of each decryption algorithm (Megabyte/Sec

Conclusions

Concerns about security and privacy now rank among the ICT industry's top priorities. For Ericsson, overcoming these concerns is a non-negotiable element of the Networked Society. The world is heading in the direction of comprehensive protection of data, where encryption techniques are not just reserved for access networks, but are applied across the entire communication system. This, together with new, more complex communication services places new demands on cryptography technology.

New cryptographic algorithms such as AEAD and ECC overcome the performance and bandwidth limits of their predecessors, in several cases offering improvements of several orders of magnitude. On the protocol side, TLS 1.3 and QUIC significantly reduce latency, as they require fewer round trips to set up secure communications.

Homomorphic encryption may create new business opportunities for cloud-storage providers. Should quantum computers become a reality, the future challenge will be to replace many established algorithms and cryptosystems. Ericsson has a deep understanding of applied cryptography, its implications, and the opportunities it presents for the ICT industry. We actively use this knowledge to develop better security solutions in standardization, services, and products, well in advance of their need in the world.

This paper presents a performance evaluation of selected symmetric encryption algorithms. The selected algorithms are AES, DES, 3DES, RC6, Blowfish and RC2.

Several points can be concluded from the Experimental results. Firstly; there is no significant difference when the results are displayed either in hexadecimal base encoding or in base 64 encoding. Secondly; in the case of changing packet size, it was concluded that Blowfish has

better performance than other common encryption algorithms used, followed by RC6. Thirdly; It is found that 3DES still has low performance compared to algorithm DES. Fourthly: It is found RC2, has disadvantage over all other algorithms in terms of time consumption. Fifthly: It is found AES has better performance than RC2, DES, and 3DES. In the case of audio and video files It is found the result as the same as in text and document. Finally, in the case of changing key size, it can be seen that higher key size leads to clear change in the battery and time consumption.

REFERENCES:

1. Gigaom Research, 2014, Data privacy and security in the post-snowden era PERC, 2015, Secure Real-time Transport Protocol (SRTP) for Cloud Services.
2. PERC, 2015, Secure Real-time Transport Protocol (SRTP) for Cloud Services, available at: https://tools.ietf.org/html/draft-mattsson-perc-srtp-cloud
3. M. S. Hwang and C. Y. Liu, \Authenticated encryp-tion schemes: current status and key issues," *Inter-national Journal of Network Security*, vol. 1, no. 2, pp. 61-73, 2005.
4. M. H. Ibrahim, \A method for obtaining deni-able public-key encryption," *International Journal of Network Security*, vol. 8, no. 1, pp. 1-9, 2009.
5. M. H. Ibrahim, \Receiver-deniable public-key en-cryption," *International Journal of Network Secu-rity*, vol. 8, no. 2, pp. 159-165, 2009.
6. P. Ding, \Central manager: A solution to avoid de-nial of service attacks for wreless LANs," *Interna-tional Journal of Network Security*, vol. 4, no. 1, pp.35-44, 2007.
7. Hardjono, *Security In Wireless LANS And MANS*, Artech House Publishers, 2005.
8. P. Ruangchaijatupon, and P. Krishnamurthy, \En-cryption and power consumption in wireless LANs-N," *The Third IEEE Workshop on Wireless LANs*,pp. 148-152, Newton, Massachusetts, Sep. 27-28,2001.
9. D. Coppersmith, \The data encryption standard (DES) and its strength against attacks," *IBM Jour-nal of Research and Development*, pp. 243 -250, May 1994.
10. J. Daemen, and V. Rijmen, \Rijndael: The advanced encryption standard," *Dr. Dobb's Journal*, pp. 137-139, Mar. 2001.
11. N. E. Fishawy, \Quality of encryption measurement of bitmap images with RC6, MRC6, and rijndael block cipher algorithms," *International Journal of Network Security*, pp. 241-251, Nov. 2007.
12. B. Schneier, *The Blow¯sh Encryption Algo-rithm*, Retrieved Oct. 25, 2008. (http://www.schneier.com/blow¯sh.html)
13. W. Stallings, *Cryptography and Network Security*,Prentice Hall, pp. 58-309, 4th Ed, 2005.
14. R. Chandramouli, \Battery power-aware encryption," *ACM Transactions on Information and System Security (TISSEC)*, vol. 9, no. 2, pp. 162-180,May 2006.
15. K. McKay, *Trade-o®s between Energy and Security in Wireless Networks Thesis*, Worcester Polytechnic Institute, Apr. 2005

16. Proceedings of the 23rd ACM,2011, CryptDB: Protecting confidentiality with encrypted query processing, abstract available at: http://dl.acm.org/citation.cfm?id=2043566
17. Ericsson, 2015, Encryption Performance Improvements of the Paillier Cryptosystem, available at: https://eprint.iacr.org/2015/864.pdf
18. National Security Agency, 2009, Cryptography Today, available at: https://www.nsa.gov/ia/programs/suiteb_cryptography/
19. IACR, Keyless Signatures' Infrastructure: How to Build Global Distributed Hash-Trees, available at: https://eprint.iacr.org/2013/834.pdf
20. P. Ruangchaijatupon, and P. Krishnamurthy, \Encryption and power consumption in wireless LANs-N," *The Third IEEE Workshop on Wireless LANs*,pp. 148-152, Newton, Massachusetts, Sep. 27-28, 2001.
21. S. Hirani, *Energy Consumption of Encryption Schemes in Wireless Devices Thesis*, University of Pittsburgh, Apr. 9,2003, Retrieved Oct. 1, 2008. (http://portal.acm.org/citation.cfm?id=383768)
22. A. Nadeem, \A performance comparison of data encryption algorithms," IEEE Information and Communication Technologies, pp. 84-89, 2006.
23. Results of Comparing Tens of Encryption Algorithms Using Di®erent Settings-Crypto++Benchmark, Retrieved Oct. 1, 2008. (http://www.eskimo.com/ weidai/ benchmarks.html)
24. A. A. Tamimi, Performance Analysis of Data Encryption Algorithms, Retrieved Oct. 1, 2008. (http://www.cs.wustl.edu/»jain/cse567-06/ftp/encryption perf/index.html)
25. K. McKay, Trade-o®s between Energy and Security in Wireless Networks Thesis, Worcester Polytechnic Institute, Apr. 2005.