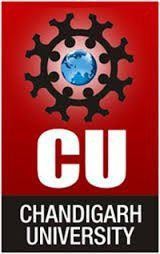
**HYBRID CRYPTOGRAPHY**

Submitted in partial fulfillment of the requirements for the award of degree of

**BACHELOR OF ENGINEERING** **IN**

**COMPUTER SCIENCE & ENGINEERING**

**2nd progress report**



**Submittedto:**

**SHIKHA ATWAL**

**SubmittedBy:**

**YUKTI GOYAL 18BCS1838**

**SHUBHAM SHUKLA 18BCS1848**

**PRADIPTA SARKAR 18BCS1854**

**MIRTUNJAY GUPTA 18BCS1843**

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

**Chandigarh University, Gharuan**

**August 2019**

**2nd progress report**

**Review of Various Algorithms Used in Hybrid Cryptography**

**Introduction:**

Cryptography enables the user to transmit confidential information across any insecure network so that it cannot be used by an intruder. Cryptography is the process that involves encryption and decryption of text using various mechanisms or algorithms. A cryptographic algorithm is a mathematical function that can be used in the process of encryption and decryption. Encryption is the process of converting the plain text into an unreadable form called a cipher text. This unreadable form cannot be easily understood by an intruder and sent across the insecure media. Decryption is the process of converting this unreadable form back into its original form, so that it can be easily understood by the intended recipient. Many algorithms exist for encryption that can be categorized into symmetric and asymmetric encryption.. In symmetric-key cryptography, also called conventional cryptography or secret-key encryption, one key is used both for encryption and decryption. Examples include DES and AES. But symmetric-key cryptography has some limitations. One major limitation is the key distribution problem. If in case key, while sending through the channel, get compromised, whole communication will get vulnerable to attacks.

There are many aspects to security and many applications, ranging from secure commerce and payments to private communications and protecting health care information. One essential aspect for secure communications is that of cryptography. But it is important to note that while cryptography is *necessary* for secure communications, it is not by itself *sufficient*. The reader is advised, then, that the topics covered here only describe the first of many steps necessary for better security in any number of situations.

**DES (Data Encryption Standard)**

The Data Encryption Standard is one of the first commercially developed ciphers. DES is the result of efforts done by IBM (International Business Machines) corporation, NBS (National Bureau of Standards) and NSA (National Security Agency). DES is a block cipher that encrypts 64-bit data blocks and encryption of the data is performed using a 56-bit secret key [4]. DES consists of sixteen rounds and two permutation layers. DES uses a shared key both to encrypt and decrypt the message. The decryption process is the reverse of encryption process. DES possesses strong Avalanche effect and is flexible as it works in CBC, ECB, CFB and OFB modes. DES easily falls pray to Brute Force attack and relatively slow in software.

**AES (ADVANCED ENCRYPTION STANDARD)**

The algorithm was invented by Joan Daemen and Vincent Rijmen. AES can process 128 bit data block and uses key lengths of 128, 192, or 256 bits. For the key length of 128,192 and 256 bits, AES may be referred to as AES-128, AES-192 and AES-256 respectively. Unlike DES, AES is not a fiestel structure. Number of rounds in AES depends on key length i.e. for a key length of 128, number of rounds is 10 and similarly for 192 and 256 bit keys, it is 12 and 14 respectively. AES provides resistance against all known attacks, simple in design and good speed of computation. The problems of key distribution are solved by public key cryptography. Some examples of public-key cryptosystems are: Elgamal, RSA, Diffie-Hellman and DSA.

**RSA (Rivest, Shamir and Adleman)**

A public key encryption algorithm developed by Ronald Rivest, Adi Shamir, and Leonard Adleman in 1977. It was the first algorithm known to be suitable for signing as well as encryption, and one of the first great advances in public key encryption. It is still widely used in electronic commerce protocols, and is believed that its security depends on the difficulty of decomposition of large numbers. RSA is secure because it is able to resist concerted attack.

**Diffie-Hellman**

Whitfield Diffie and Martin Hellman discovered DiffieHellman (DH) algorithm in 1976 was the first public key algorithm ever invented. Diffie–Hellman establishes a shared secret key that can be used for secret communications by exchanging data over a public network. Diffie–Hellman algorithm does not need any known key before communication begins and Discrete Logarithm Problem makes it extremely difficult to crack. Diffie–Hellman algorithm easily falls pray to man-in-the-middle attack.

**Elgamal Algorithm**

El Gamal algorithm was invented by Taher El-Gamal which is based on Discrete Logarithm Problem and Diffie-Hellman key exchange [19]. Elgamal can be used for encryption as well as digital signature. Each time when the same plaintext is encrypted, it gives a different ciphertext. Elgamal has the disadvantage of having ciphertext twice the size of the plaintext.

**DSA (Data Signature Algorithm)**

Data Signature Algorithm as an approved signature scheme was invented by David Kravitz. DSA is a variant of the ElGamal and Schnorr algorithms. Digital Signature Standard (DSS) used DSA proposed by National Institute of Standards and Technology (NIST) in 1991. Security of DSA is based on the difficulty to solve discrete logarithms. DSA has been accepted widely. DSA is more efficient and faster than RSA. In practice, the symmetric key algorithms and public key cryptography algorithms are generally combined together. Combining the features of two algorithms for the sake of better efficiency and performance and for combating the problems with the already existing algorithms, the process occurs is known as Hybrid cryptography. Hybridization of algorithms is a useful scheme that provides solutions to some major problems in the communication networks or any other means. Numerous ways have been made available by the study to carry hybridization. Deploying the positive points of an algorithm such as RSA into other less efficient algorithms will result in new hybrid cryptographic algorithm. For this literature survey, the period of interest begins in the year 1993. For the period from 1993 to 2013, 500 IEEE papers were found. Topic filtering reduced this number to 49 which were related to the keyword,

Hybrid Cryptograph.Some papers are also included that are not related to hybrid cryptography but those papers are very beneficial for the researchers to view which algorithm in comparison to others can be used in their studies. The remaining part of the paper is organized as follows. Section II provides the details of the literature survey been performed, results and discussions are available in Section III and Section IV contains the conclusion of the paper.

**Access Control and Encryption of Sensitive Data Using i-Se4GE Algorithm:**

**Advantages**

New scheme presented provides secure data transmission, privacy preservation and reduced processing time for key exchange. It also enhances key exchange security and provided a new method for message digest calculation.

**Introduction**

Transferring of Text/Audio/Video through an application by using cryptography is the software aimed at providing a high level of security to the sensitive data for organizations, since they face issue in using the existing system in which they cannot send all three media files. The aim is all about security and making the sensitive data secured. There is an additional feature in the project of self-liquidation according to which the data gets deleted after a fixed amount of time.

The message application developed can be used for military purposes as the messages will be encrypted which is needed for every organization to communicate. We have added the access control policies which help data delivered only to the desired user. Other than the military, it can be used by any external parties to send confidential messages. It can also be used to share images and videos that are meant to be sensitive.

**Cryptocat3,6**

One of the open source mobile and web application tool which allows secure and encrypted chatting is Cryptocat. Cryptocat uses end-toend encryption and encrypts chats on the client side, only server trust with data that is already encrypted. Cryptocat is existing as an app for Mac OS X or as a browser extension for Google Chrome, Mozilla Firefox, Apple Safari, Opera and as a mobile app for iPhone. Cryptocat’s defined goal is to make encrypted communications more accessible to average users. The chat software aims to strike a balance between security and usability recommending more privacy than services such as Google Talk or Internet Relay Chat, while maintaining a higher level of accessibility than Pidgin.

**TextSecure 5,8**

TextSecure is an cutting-edge end-to-end encryption protocol as well as a free and open-source encrypted instant messaging application for Android which uses that protocol. TextSecure enables the secure transmission of instant messages, group messages, attachments and media messages to other Text Secure users.TextSecure and Signal are developed by Open Whisper Systems and are published under the GPLv3 license. TextSecure allows users to send encrypted text messages, audio messages, photos, videos, contact information, and a wide selection of emotions over a data connection (e.g. Wi-Fi, 3G or 4G) to other TextSecure users with smartphones running Android and to Signal users on iOS. TextSecure also allows users to exchange unencrypted SMS and MMS messages with people who do not have TextSecure or Signal. Messages sent with TextSecure to other TextSecure users and to Signal users are automatically end-to-end encrypted, which means that they can only be read by the intended recipients.

**Se4ge Algorithm**

The aim is to create an application that encrypts all the media files before being sent to the receiver. Since, there has been a great increase in the amount of data available on the internet due to which it is important to encrypt and send the data which is critical to an organization or a company.

Evolved Packet core .

User Equipment.

E-UTRAN Node Bs .

Mobility Management Entity .

Home Subscriber Server .

**The i-Se4GE Functions**

The Logical functions used in Algorithms are as follows.

1. U +2 V : where +2 indicates Logical OR operation which ignores most significant carry.

2) X⊕Y : where ⊕ represents Logical Ex-OR function.

3) Dafun(a,b,c) = (a ⊕ b) + c : is data authencation function.

4) IDafun(a,b,c) : inverse of Dafun() is defined as a = IDafun(a,b,c)= (x - c) ⊕ b, if xe” c (x +c+ 1) ⊕ b, if x < c Where x= Dafun(a,b,c). The IDafun() enables retrieval of a if and only if the user has b and c. This way it makes an intelligent protection key chain and guarantees that user will get a only if he has b and c.

5) RSA\_En(mi ) = (mi ) eimod Ni : encrypts plain text message expressed as BigInteger.

6) RSA\_De(ci ) = (ci ) di mod Ni : decrypts cipher text to plain text expresses as BigInteger.

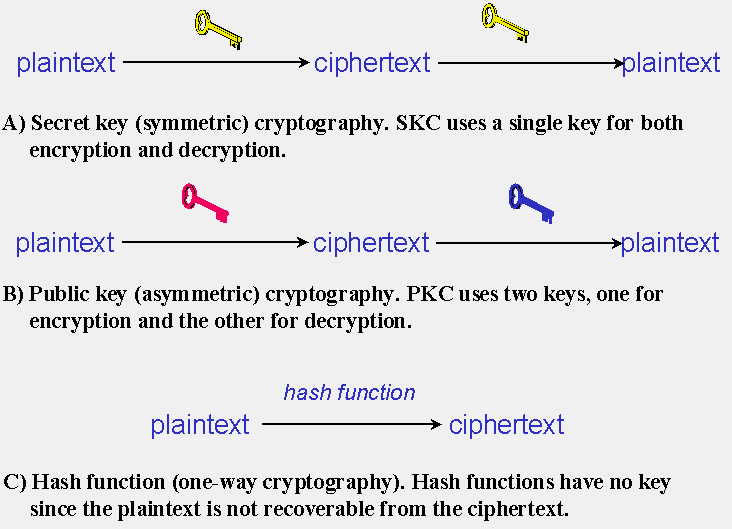
7) HMAC(k) : Hash based Message Authentication Code. Ensures the integrity of sending and receiving message.

**TYPES OF CRYPTOGRAPHIC ALGORITHMS**

**Secret Key Cryptography (SKC):** Uses a single key for both encryption and decryption; also called symmetric encryption. Primarily used for privacy and confidentiality.

**Public Key Cryptography (PKC):** Uses one key for encryption and another for decryption; also called asymmetric encryption. Primarily used for authentication, non-repudiation, and key exchange.

**Hash Functions:** Uses a mathematical transformation to irreversibly "encrypt" information, providing a digital fingerprint. Primarily used for message integrity.

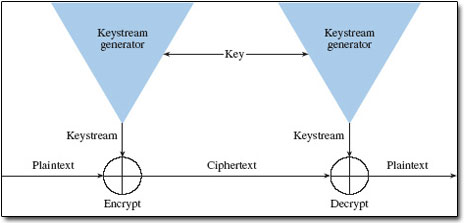


### Secret Key Cryptography

Secret key cryptography methods employ a single key for both encryption and decryption. As shown in Figure 1A, the sender uses the key to encrypt the plaintext and sends the ciphertext to the receiver. The receiver applies the same key to decrypt the message and recover the plaintext. Because a single key is used for both functions, secret key cryptography is also called *symmetric encryption*.

With this form of cryptography, it is obvious that the key must be known to both the sender and the receiver; that, in fact, is the secret. The biggest difficulty with this approach, of course, is the distribution of the key (more on that later in the discussion of public key cryptography).

Secret key cryptography schemes are generally categorized as being either *stream ciphers* or *block ciphers*.



### Public Key Cryptography

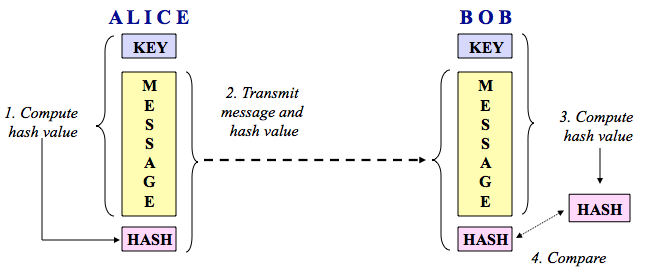
Public key cryptography has been said to be the most significant new development in cryptography in the last 300-400 years. Modern PKC was first described publicly by Stanford University professor Martin Hellman and graduate student Whitfield Diffie in 1976. Their paper described a two-key crypto system in which two parties could engage in a secure communication over a non-secure communications channel without having to share a secret key.

PKC depends upon the existence of so-called *one-way functions*, or mathematical functions that are easy to compute whereas their inverse function is relatively difficult to compute. Let me give you two simple examples:

1. **Multiplication vs. factorization:** Suppose you have two prime numbers, 3 and 7, and you need to calculate the product; it should take almost no time to calculate that value, which is 21. Now suppose, instead, that you have a number that is a product of two primes, 21, and you need to determine those prime factors. You will eventually come up with the solution but whereas calculating the product took milliseconds, factoring will take longer. The problem becomes much harder if we start with primes that have, say, 400 digits or so, because the product will have ~800 digits.
2. **Exponentiation vs. logarithms:** Suppose you take the number 3 to the 6th power; again, it is relatively easy to calculate 36 = 729. But if you start with the number 729 and need to determine the two integers, *x* and *y* so that logx 729 = y, it will take longer to find the two values.

### Hash Functions

Hash functions, also called *message digests* and *one-way encryption*, are algorithms that, in essence, use no key Instead, a fixed-length hash value is computed based upon the plaintext that makes it impossible for either the contents or length of the plaintext to be recovered. Hash algorithms are typically used to provide a *digital fingerprint* of a file's contents, often used to ensure that the file has not been altered by an intruder or virus. Hash functions are also commonly employed by many operating systems to encrypt passwords. Hash functions, then, provide a mechanism to ensure the integrity of a file.



Let me reiterate that hashes are **one-way** encryption. You cannot take a hash and "decrypt" it to find the original string that created it, despite the many web sites that claim or suggest otherwise, such as [CrackStation](https://crackstation.net/" \t "https://www.garykessler.net/library/_blank), [HashKiller.co.uk](https://hashkiller.co.uk/md5-decrypter.aspx" \t "https://www.garykessler.net/library/_blank), [MD5 Online](http://www.md5online.org/" \t "https://www.garykessler.net/library/_blank), **[md5this](http://www.md5this.com/" \t "https://www.garykessler.net/library/_blank)**[cracker](http://www.md5this.com/" \t "https://www.garykessler.net/library/_blank), [OnlineHashCrack](http://www.onlinehashcrack.com/" \t "https://www.garykessler.net/library/_blank), and [RainbowCrack](http://project-rainbowcrack.com/" \t "https://www.garykessler.net/library/_blank).

Note that these sites search databases and/or use [rainbow tables](https://en.wikipedia.org/wiki/Rainbow_table" \t "https://www.garykessler.net/library/_blank) to find a suitable string that produces the hash in question but one can't definitively guarantee what string originally produced the hash. This is an important distinction. Suppose that you want to crack someone's password, where the hash of the password is stored on the server. Indeed, all you then need is a string that produces the correct hash and you're in! However, you cannot prove that you have discovered the user's password, only a "duplicate key."

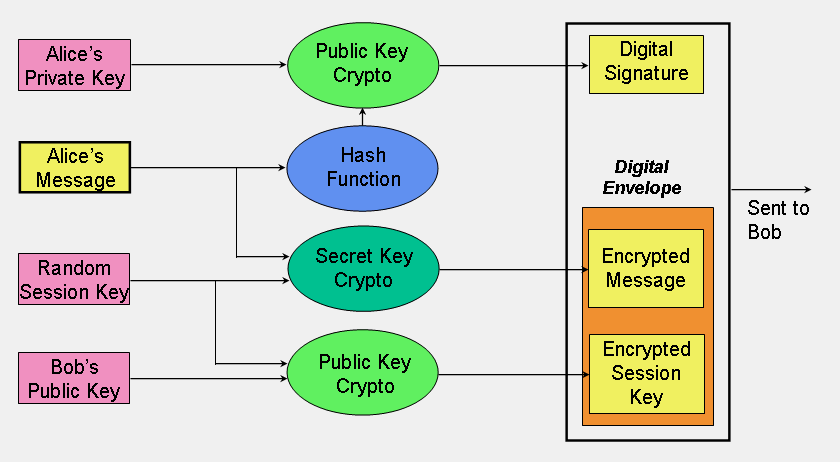
### Why Three Encryption Techniques?

So, why are there so many different types of cryptographic schemes? Why can't we do everything we need with just one?

The answer is that each scheme is optimized for some specific cryptographic application(s). Hash functions, for example, are well-suited for ensuring data integrity because any change made to the contents of a message will result in the receiver calculating a different hash value than the one placed in the transmission by the sender. Since it is highly unlikely that two different messages will yield the same hash value, data integrity is ensured to a high degree of confidence.

Secret key cryptography, on the other hand, is ideally suited to encrypting messages, thus providing privacy and confidentiality. The sender can generate a *session key* on a per-message basis to encrypt the message; the receiver, of course, needs the same session key in order to decrypt the message.

Key exchange, of course, is a key application of public key cryptography (no pun intended). Asymmetric schemes can also be used for non-repudiation and user authentication; if the receiver can obtain the session key encrypted with the sender's private key, then only this sender could have sent the message. Public key cryptography could, theoretically, also be used to encrypt messages although this is rarely done because secret key cryptography values can generally be computed about 1000 times faster than public key cryptography values.



**Significance of Key Length**

In a 1998 article in the industry literature, a writer made the claim that 56-bit keys did not provide as adequate protection for DES at that time as they did in 1975 because computers were 1000 times faster in 1998 than in 1975. Therefore, the writer went on, we needed 56,000-bit keys in 1998 instead of 56-bit keys to provide adequate protection. The conclusion was then drawn that because 56,000-bit keys are infeasible (*true*), we should accept the fact that we have to live with weak cryptography (*false!*). The major error here is that the writer did not take into account that the number of possible key values double whenever a single bit is added to the key length; thus, a 57-bit key has twice as many values as a 56-bit key (because 257 is two times 256). In fact, a 66-bit key would have 1024 times more values than a 56-bit key.

**Summary**

The paragraphs above describe three very different trust models. It is hard to say that any one is better than the others; it depends upon your application. One of the biggest and fastest growing applications of cryptography today, though, is electronic commerce (e-commerce), a term that itself begs for a formal definition.

PGP's web of trust is easy to maintain and very much based on the reality of users as people. The model, however, is limited; just how many public keys can a single user reliably store and maintain? And what if you are using the "wrong" computer when you want to send a message and can't access your keyring? How easy it is to revoke a key if it is compromised? PGP may also not scale well to an e-commerce scenario of secure communication between total strangers on short-notice.

**. Pretty Good Privacy (PGP)**

Pretty Good Privacy (PGP) is one of today's most widely used public key cryptography programs. Developed by Philip Zimmermann in the early 1990s and long the subject of controversy, PGP is available as a plug-in for many e-mail clients, such as Apple Mail (with GPG), Eudora Email, Microsoft Outlook/Outlook Express, and Mozilla Thunderbird (with Enigmail).

PGP can be used to sign or encrypt e-mail messages with the mere click of the mouse. Depending upon the version of PGP, the software uses SHA or MD5 for calculating the message hash; CAST, Triple-DES, or IDEA for encryption; and RSA or DSS/Diffie-Hellman for key exchange and digital signatures.

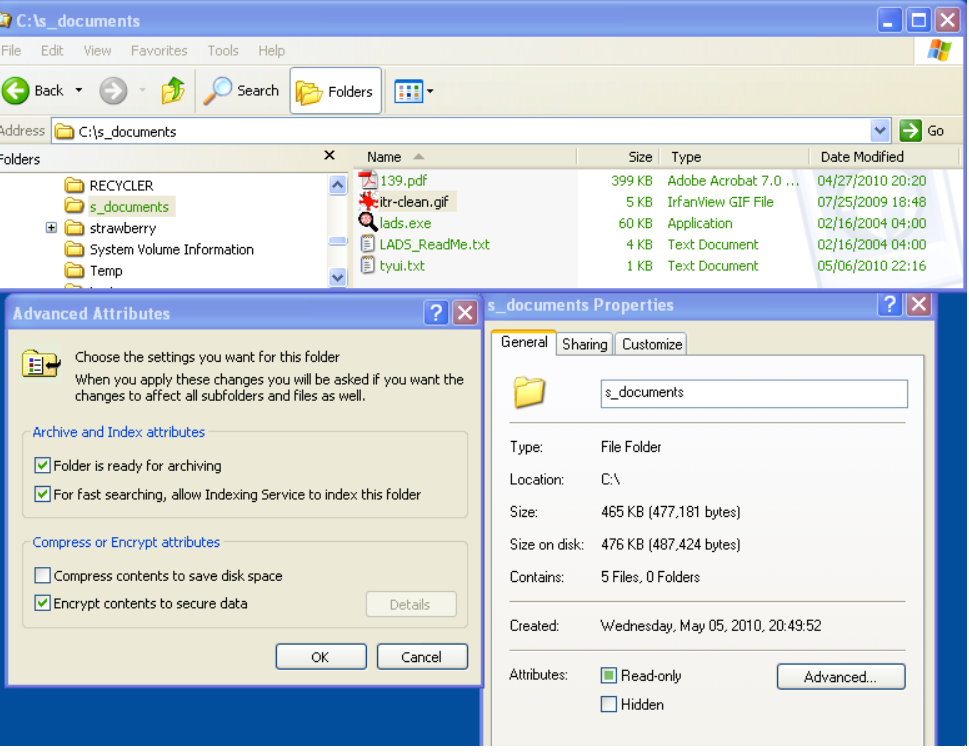
When PGP is first installed, the user has to create a key-pair. One key, the public key, can be advertised and widely circulated. The private key is protected by use of a passphrase. The passphrase has to be entered every time the user accesses their private key.

|  |
| --- |
| -----BEGIN PGP MESSAGE-----  Comment: GPGTools - https://gpgtools.org  hQEMA02ePRsA5fMjAQf/fJIFvXKggtfaSAXJzRRZ3sXhKJN+0kH5Kj7GdqbhBd81  n0TZ31BAoXksEQ0Q3HbN8PbJ4qTwLE+glAqVqaGfGmieEirD74/6jX/jffuA028+  X110IX4gJTpkhHzYeabrxPy9yerjI2EQL0XI4313K7w4vKZ8kAEVU86+DCHKm3br  B/UrYlYDPg2xPjgqIx8Zyga2fSnb4TvqYs2+6k9O7RHKu72wKY0H/xx+rqhEmEAk  L/sIxrlCufVM22zEnlbhO5BEqRhBN6CkRS7HkvxOWHetw4dOIbCjc1WI2CMGzHoK  Lx2a3wOskjFbS+0PnDU/CKJV002fO+MCxvYhsF3B7dLAFAG80+08XFBKMll99/Wj  bttk96NLkaz45SG9KNJyqj7KaRFYscnUbEjHunvFIJCnl0CblJb3J/gBaa/ZbSLq  0GGDr9oc0tKoR97mabkyAhohcpiIn7f2y2aCBx5qTRT2uMiedmu4XtyktgB4LUo+  /MspTWlbcyKk9+Gx+9a7QaRkvBskwDn1wL/EIfNhXvtga+qGCV3rZwEX7f46jdzi  NcptK7urvqiWgYtKS1z/0VrppbHBoTux3TQcFAEzKAGCYnS2YCyjNJRqTC/ODPE8  BIIYcYNq  =+eqR  -----END PGP MESSAGE----- A PGP encrypted message. The receiver's e-mail address is the pointer to the public key in the sender's keyring. At the destination side, the receiver uses their own private key. |

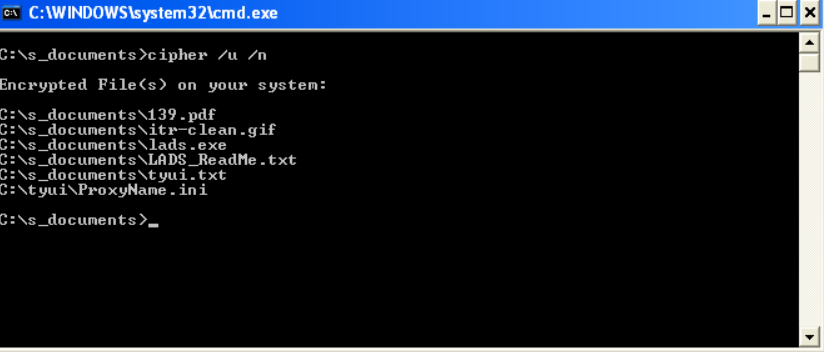
|  |
| --- |
| Hi Gary,  "Outside of a dog, a book is man's best friend.  Inside of a dog, it's too dark to read."  Carol FIGURE 11: The decrypted message. |

**Encrypting File System (EFS)**

Microsoft introduced the Encrypting File System (EFS) into the NTFS v7.0 file system and has supported EFS since Windows 10 and XP (although EFS is not supported in all variations of all Windows OSes). EFS can be used to encrypt individual files, directories, or entire volumes. While off by default, EFS encryption can be easily enabled via File Explorer (aka Windows Explorer) by right-clicking on the file, directory, or volume to be encrypted, selecting Properties, Advanced, and Encrypt contents to secure data . Note that encrypted files and directories are displayed in green in Windows Explorer.

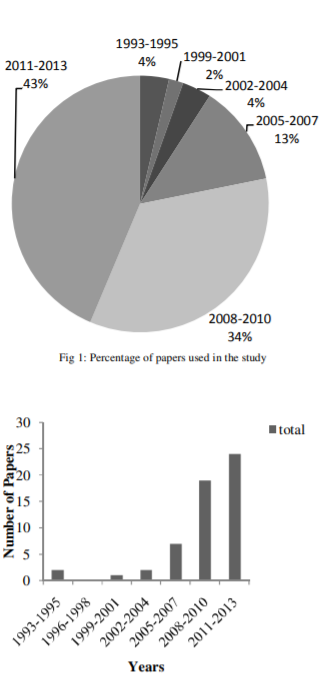


The Windows command prompt provides an easy tool with which to detect EFS-encrypted files on a disk. The cipher command has a number of options, but the /u/n switches can be used to list all encrypted files on a drive



**Results and Discussion**

An overview of different cryptographic algorithms is presented in this paper. The IEEE database search has been done to obtain the papers dealing with hybrid cryptography. The search is mostly done on the basis of IEEE papers survey. According to all search results Tables 2, 3 and 4 have been prepared. Table 2 summarizes the Figs. 1 and 2, and gives the number of papers available for study on a yearly basis. Fig.-1 shows the percentage of papers used in the study from a particular year. By analyzing Fig.-1, it can be noticed that more efforts have been put to improve the performance of different algorithms during the years 2011-2013 i.e. 43%. Fig.-2 depicts the comparison of the number of papers published between the years 1993- 2013 that has been used in this review paper. It can be clearly visible that the use of hybrid cryptography is going on increasing day by day.

summarizes the number of papers that usescryptographic algorithms solely as well as algorithms that have been used along with some other algorithms like in case of hybrid cryptography. It can be noticed that hybrid cryptography is a demanding approach for today. Hybrid cryptography is gaining its strength as the naïve researchers laid more emphasis on combination of different cryptographic algorithms for better results

**Conclusion**

With the results of increased efficiency, speed and throughput of various algorithms by the combination of various algorithms and techniques, hybrid cryptography has a great scope in the near future. Hybrid cryptography has been creating various opportunities for the naïve researchers and allows them to work upon various challenging limitations of algorithms in their original forms. Hybrid cryptography is easy to work upon and a great number of chances for improvement are there. A number of different useful techniques and algorithms have been prescribed in this paper that can be used for providing security in the insecure media. This paper has been providing the study of past 20 years in the search for hybrid cryptographic algorithms that may help researchers to orientate their study areas and to choose various cryptographic algorithms for their studies. The study indicates the maximum use of RSA in the hybridization of various algorithms because of its Integer Factorization Problem. Diffie-Hellman being very secure is the prior choice for eliminating various limitations of cryptographic algorithms. AES and DES have limited scope of use because of the problem of key management. No doubt, the number of cryptographic algorithms presented here is neither complete nor exhaustive but a sample of papers that demonstrates the advantages and limitations of used cryptographic algorithms.

This paper has briefly (!?) described how digital cryptography works. The reader must beware, however, that there are a number of ways to attack every one of these systems; cryptanalysis and attacks on cryptosystems, however, are well beyond the scope of this paper. In the words of Sherlock Holmes (ok, Arthur Conan Doyle, really), "What one man can invent, another can discover" ("The Adventure of the Dancing Men," in: The Return of Sherlock Holmes, 1903).

There are a lot of topics that have been discussed above that will be big issues going forward in cryptography. As compute power increases, attackers can go after bigger keys and local devices can process more complex algorithms. Some of these issues include the size of public keys, the ability to forge public key certificates, which hash function(s) to use, and the trust that we will have in random number generators. Interested readers should check out "Recent Parables in Cryptography" (Orman, H., January/February 2014, IEEE Internet Computing, 18(1), 82-86).

Cryptography is a particularly interesting field because of the amount of work that is, by necessity, done in secret. The irony is that secrecy is not the key to the goodness of a cryptographic algorithm. Regardless of the mathematical theory behind an algorithm, the best algorithms are those that are well-known and well-documented because they are also well-tested and well-studied! In fact, time is the only true test of good cryptography; any cryptographic scheme that stays in use year after year is most likely a good one. The strength of cryptography lies in the choice (and management) of the keys; longer keys will resist attack better than shorter keys.

TEAM MEMBER WISE DISTRIBUTION OF WORK

**Yukti Goyal (Team Leader):**

Yukti will be to do research on various cryptographic algorithms of both symmetric and asymmetric encryption and how they both can be used to make more optimized algorithm on the basis of speed and security which is hybrid encryption of public and private key cryptography.

**Shubham Shukla:**

My main field will be to do research on various shortcomings of symmetric and asymmetric algorithms like DES(Data Encryption Standard), Triple DES, Diffie-Hellman and many more and also how these shortcomings can be overcome by using hybrid encryption.

**Pradipta Sarkar:**

Pradipta will look after the performance evaluation of symmetric and asymmetric algorithms like DES(Data Encryption Standard), Triple DES, Diffie-Hellman and many more and compare them and thus deduce an optimized algorithm for hybrid encryption.

**Mirtunjay Gupta:**

He will be doing research on future scope of cryptography and modern cryptography techniques which cannot be exploited using brute force attacks.

**References**

[1] S. Mohanty, B. Majhi, and V. Iyer, “A Strong Designated Verifiable Group Signature”, Automation, Computing, Communication, Control and Compressed Sensing (iMac4s), 2013 International Multi-Conference on 22-23 March 2013, page(s): 518-523.

[2] H. WANG, Z. SONG, X. NIU, and Q. DING, “Key Generation Research of RSA Public Cryptosystem and MATLAB Implement”, Sensor Network Security Technology and Privacy Communication System (SNS & PCS), 2013 International Conference on 18-19 May 2013, page(s): 125-129.

[3] Y. L. Huang, F. Y. Leu, Y. K. Sun, C. C. Chu, and C. T. Yang, “A Secure Wireless Communication System by Integrating RSA and Diffie-Hellman PKDS in 4G Environments and an Intelligent Protection-key Chain with a Data Connection Core”, Industrial Electronics (ISIE), 2013 IEEE International Symposium on 28-31 May 2013, page(s): 1-6.

[4] A. K. Mandal, C. Parakash, and Mrs. A. Tiwari, “Akash Kumar Mandal, Chandra Parakash, Mrs. Archana Tiwari”, 2012 IEEE Students’ Conference on Electrical, Electronics and Computer Science, 2012.

[5] J. Zhang, and X. Jin, “Encryption System Design Based on DES and SHA-1”, Distributed Computing and Applications to Business, Engineering & Science (DCABES), 2012 11th International Symposium on 19- 22 Oct. 2012, page(s):317-320.

Bamford, J. (1983). The Puzzle Palace: Inside the National Security Agency, America's most secret intelligence organization. New York: Penguin Books.

Bamford, J. (2001). Body of Secrets : Anatomy of the Ultra-Secret National Security Agency from the Cold War Through the Dawn of a New Century. New York: Doubleday.

Barr, T.H. (2002). Invitation to Cryptology. Upper Saddle River, NJ: Prentice Hall.

Basin, D., Cremers, C., Miyazaki, K., Radomirovic, S., & Watanabe, D. (2015, May/June). Improving the Security of Cryptographic Protocol Standards. IEEE Security & Privacy, 13(3), 24:31.

Bauer, F.L. (2002). Decrypted Secrets: Methods and Maxims of Cryptology, 2nd ed. New York: Springer Verlag.

Belfield, R. (2007). The Six Unsolved Ciphers: Inside the Mysterious Codes That Have Confounded the World's Greatest Cryptographers. Berkeley, CA: Ulysses Press.

Denning, D.E. (1982). Cryptography and Data Security. Reading, MA: Addison-Wesley.

Diffie, W., & Landau, S. (1998). Privacy on the Line. Boston: MIT Press.