A Hybrid Encryption Algorithm based on RSA and Diffie-Hellman

CS 573 A - Introduction to Cyber Security

Spring 2022

By,

Omkar Sinha

CWID: 10468301

Dept. Computer Science

Stevens Institute of Technology

Under Guidance of

Prof. Dr. Edward G. Amoroso

Introduction:

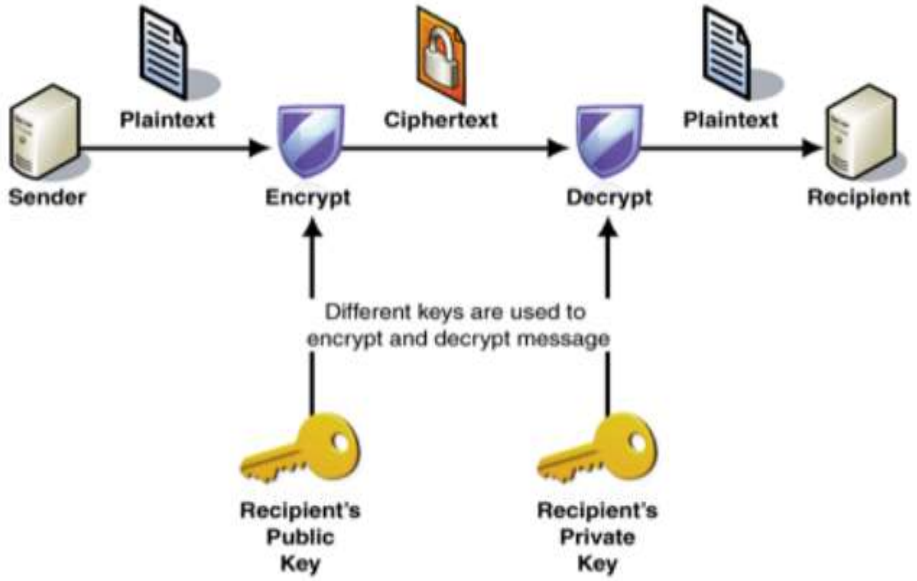
The subject of our discussion is the published IEEE paper: "*A Hybrid Encryption Algorithm based on RSA and Diffie-Hellman*" by Shilpi Gupta and Jaya Sharma in 2012[1]. (The paper is attached with submission). This paper's objective is to elucidate a hybrid model to integrate the application of the Rivest–Shamir–Adleman (RSA) public key model with the Diffie-Hellman secret key exchange system. The hybrid is meant to be greater than the sum of its parts by eliminating the drawbacks of each technique using a novel integration approach to provide greater security.

The paper starts with concise explanation of asymmetric cryptography and its advantages, followed by introduction of RSA and Diffie-Hellman models before explaining the proposed hybrid model. Our discussion follows a roughly similar order as we explain the main points of the paper before presenting our own conclusion and observations regarding this proposal.

Asymmetric Cryptography:

Asymmetric cryptography has 2 keys per user – a private key and a public key. Its main features are:

- The sender of a message uses the receiver's public key (which is visible to everyone, hence public) to encrypt the message.
- The receivers, on receiving the encrypted message use their private key to decrypt the encrypted message.
- The public and private keys have an inverse effect on each other, i.e., undoing the effect of the other.
- Certification Authority (CA) provides the users with key-pair (the public keys of these CAs are embedded into browser code to facilitate secure e-commerce).



(images are taken from the paper or the internet for illustrative purpose)

[1] A hybrid encryption algorithm based on RSA and Diffie-Hellman: Downloaded from IEEE Explore (*https://ieeexplore.ieee.org/document/6510190*); Published in: 2012 IEEE International Conference on Computational Intelligence and Computing Research; Date of Conference: 18-20 Dec. 2012

Advantages of asymmetric cryptography:

- There is a better key distribution infrastructure. A Centralized key distribution center - which can be compromised - is not needed
- Scalability at a higher level can be achieved because there is no need for a unique key for every pair of transacting user entities. Each user needs just a pair of public and private key.

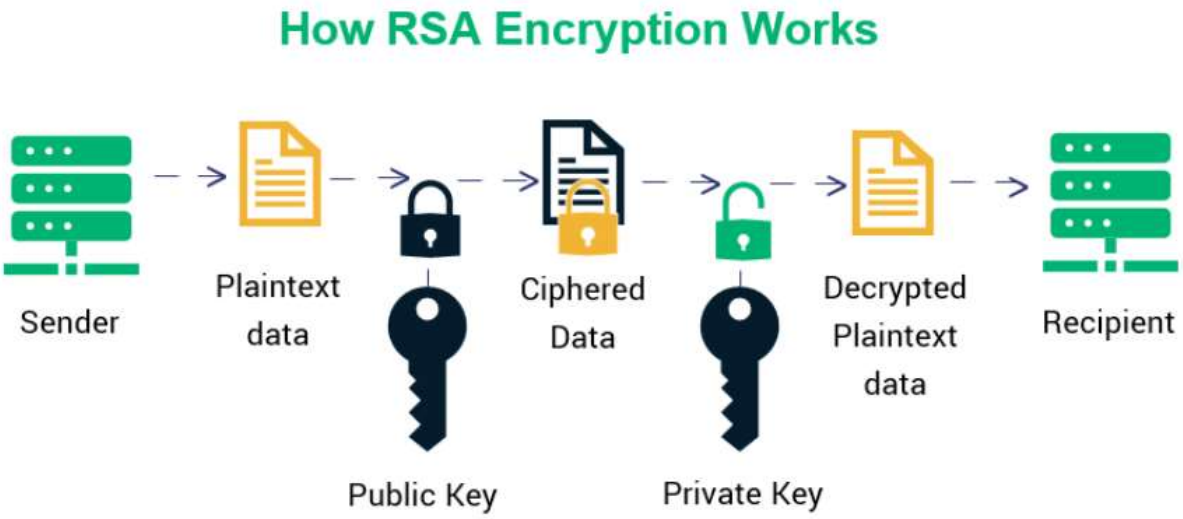
The examples of asymmetric encryption are RSA, Diffie-Hellman, ElGamal, DSA, etc. The two most popular – RSA and Diffie-Hellman – are discussed here.

RSA Algorithm:

The RSA is an asymmetric public key encryption algorithm devised by Ronald Rivest, Adi Shamir, and Leonard Adleman in 1978. Its security is based on the difficulty of factoring large prime numbers. The underlying idea is that it is easy to multiple 2 huge prime numbers (100 digits or more) but incredibly hard to factorize their product.

RSA Steps:

1. Choose 2 very large prime numbers – *p* and *q* (of about 1024 bits)
2. Calculate *N = P x Q* (*n* is used as mod for both the public and private keys)
3. Randomly select encryption(public) key *E*, such that *E* is not factor of *(P-1)(Q-1)*
4. Obtain the decryption(private) key *D*, such that the equation holds: *(DxE) mod (P-1)(Q-1) = 1*
5. Encrypt the plain-text using the public key of the intended recipient, *E*.
6. Send the cypher-text over to the recipient.
7. Decrypt the cypher-text using the recipient's private key, *D*.



Advantages of RSA:

1. Allows digital signatures (encryption with private key) to authenticate senders
2. Is highly secure due to public key encryption system

Drawbacks of RSA:

1. If any one of P, Q, E, W is hacked or leaked, then all other values can be calculated. Hence confidentiality is very critical.
2. Is slower algorithm because of the public key system.
3. It is important to make sure that message length should be less then bit length N (= P x Q)
4. Each time RSA initialization process needs to generate 2 large prime numbers. This is both expensive and time-consuming.
5. Key-generation systems are not entirely random but rely on algorithms; which can be used to predict the prime numbers.

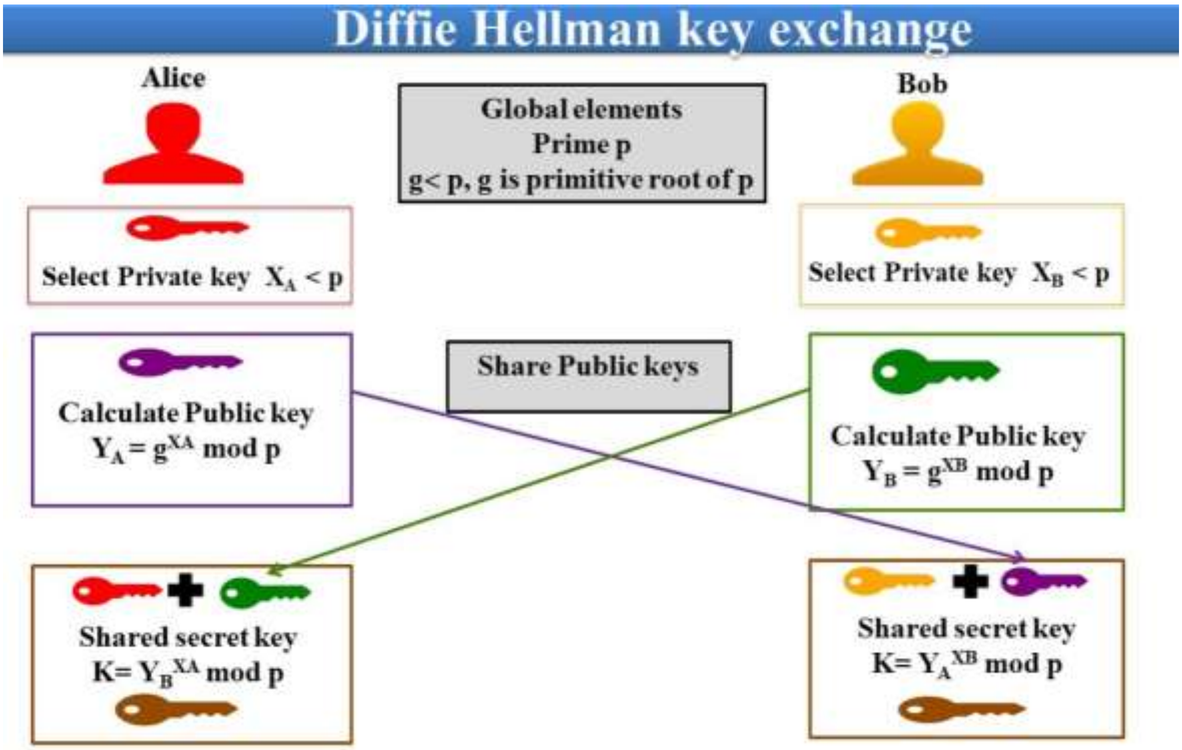
Diffie-Hellman Algorithm:

Whitfield Diffie and Martin Hellman discovered the Diffie-Hellman (DH) algorithm in 1976, and it's been common ever since. It is commonly used in various internet connectivity protocols such as Secure Sockets Layer (SSL), Secure Shell (SSH), and Internet Protocol Security(IPSec).

The strength of Diffie-Hellman security is based on difficulty of discrete logarithms. It has an elegant way of creating a shared secret key between two users with no prior communication. This secret key is used to encrypt and decrypt messages. This technique resembles symmetric cryptography but without the need of a Key Distribution Center.

Diffie-Hellman Steps:

1. Take 2 public numbers – a very large prime number, N (2000-4000 bits long), and a small prime number *G*(often called base).
2. Pick two secret numbers A and B (for sender and receiver respectively, between 1 and N).
3. Compute the 2 public keys for the users: *X, Y* as follows:
   $X = G^A \bmod N$
   $Y = G^B \bmod N$
4. The users exchange their public keys
5. First user knows *P, G, A, X, Y* and second knows *P, G, B, X, Y.*
6. First calculates the secret key; $K_A : Y^A = (G^B \bmod N)^A = G^{ab} \bmod N$
   Second calculates the secret key; $K_B : X^B = (G^A \bmod N)^B = G^{ab} \bmod N$
   Which are both the same, $K_A = K_B = K$ (secret session key). This is our 'Shared Secret key'.
7. Encrypt message using *K* and send. Receiver decrypts using the same key.

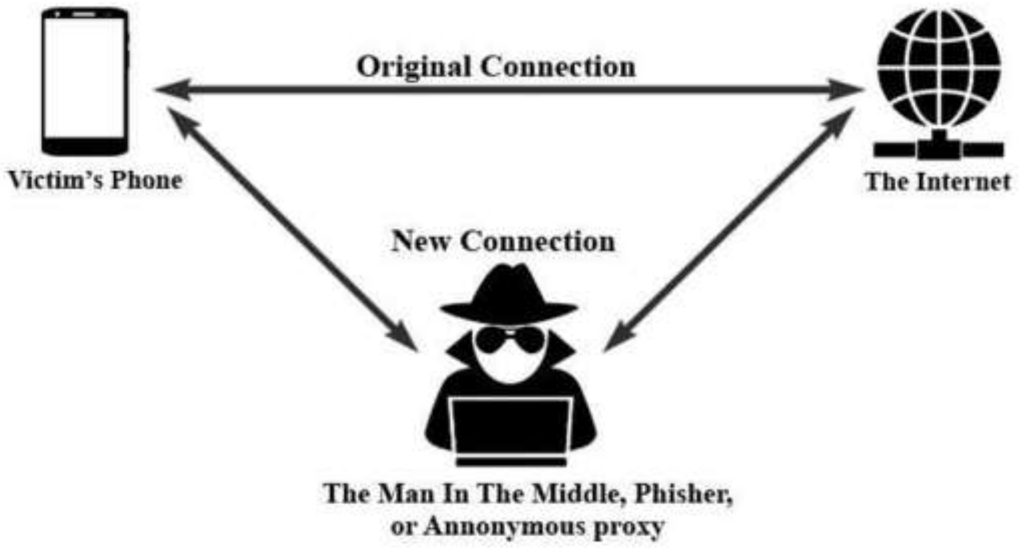


Advantages of Diffie-Hellman:

1. Generates a “shared secret”-an identical cryptographic key shared by each side, who need not have communicated before
2. Is relatively faster because of shared secret key.

Drawbacks of Diffie-Hellman:

1. Inability to implement authentication. Diffie-Hellman does not have digital signature and hence is susceptible to Man-in-the-middle(MITM) attacks, where malicious users spoof the system.
2. It’s not an encryption method by itself. Encryption/decryption is carried out in symmetric manner by the secret key which is held by both sides.
3. Vulnerable to denial-of-service(DDoS) attacks.



Man-in-the-middle attack

Proposed Hybrid model:

Both RSA and Diffie-Hellman are commonly used but have certain drawbacks. Our proposed model combines them to get security advantage of public key system and speed advantage of secret key system.

We basically use the RSA to generate public and private keys for both users separately, and use them to accomplish both digital signatures and secret key exchange.

Proposed model steps:

1. Choose 2 large prime numbers, P and Q. Do following on both sides.
   a. Calculate $N = P \times Q$.
   b. Select public key(encryption key) E, such that it's not a factor of $(P - 1)$ and $(Q - 1)$.
   c. Select the private key(decryption key) D such that: $(D \times E) \bmod (P - 1) \times (Q - 1) = 1$
   d. Take R and G is automatic generated prime constants.
   e. And put the value of E and D from above as secret number such that $A=E$ and $B=D$.

2. Now calculate public keys:

   $X= G^A \bmod R$

   $Y= G^B \bmod R$

3. We calculate the session key:

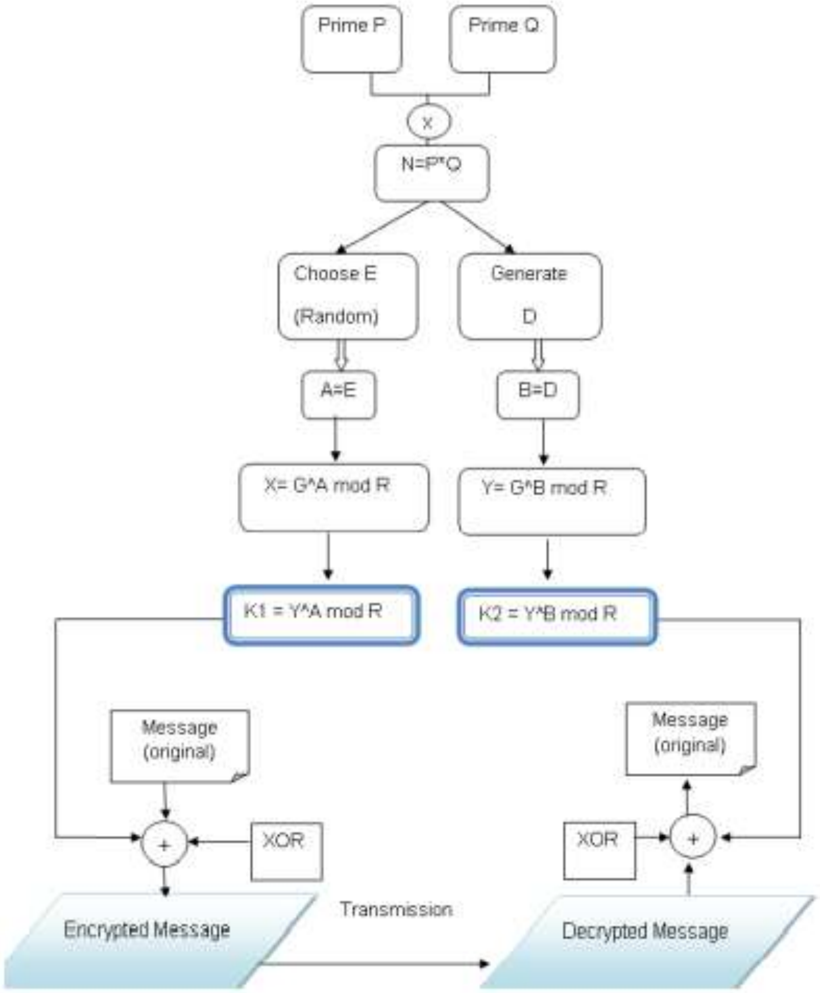
   $K_A = Y^A$

   $K_B = X^B$

   As we saw in Diffie-Hellman, $K = K_A = K_B$ (which is shared secret key)

4. We use K to encrypt/decrypt messages using XOR.

Workflow of Proposed model:



Hybrid RSA and Diffie-Hellman

Our observations and Conclusion

We observe in the above proposed model that keys generated in RSA are used to implement both secret key exchange(Diffie-Hellman) and digital signatures(RSA). Our main goals were to:

- Obtain security advantage of RSA
- Obtain speed advantage of Diffie-Hellman
- Ensure authentication (as in RSA)
- Mitigate drawbacks of both models

Regarding the security advantage we have surely enhanced security by ensuring authentication, thereby, precluding possibility of MITM attacks. This is a major advantage over pure Diffie-Hellman. Also, use of RSA-generated keys makes Diffie-Hellman more secure.

We use shared secret keys for encryption - avoiding the RSA’s need for generating prime numbers on every initialization – enabling faster transaction.
We also ensure private and public key pair of each user will be used to sign digital signature for authentication.

The hybrid model mitigates many drawbacks of both individual algorithms. Its use of secrets keys means it has no single-point-of-failure as in RSA if either (P,Q,E and D) are known. Likewise, it eliminates need for generating large primes frequently. It ensures digital signature, making Diffie-Hellman more secure, reducing vulnerabilities to DDoS and MITM attacks.

However, this novel approach will apply only to specific area of use-cases, depending on specific requirements, rather than universally. Scenarios needing a secure and authenticated channel alongside rapid and seamless communication, will be most benefitted from this model. Other scenarios like those less concerned about insecure channel may continue to opt for Diffie-Hellman and those requiring extreme security may choose pure RSA.

Further enhancements like reducing key size and improving time complexity can make it preferrable to an even larger set of use cases.