Secure Satellite Database Transmission

Atif Farid Mohammad, Pamela Almeida, Yasmin Soliman, Ajay Sadhu, and Keerthi Kata
University of North Carolina at Charlotte
9201 University City Blvd
Charlotte, NC 28223
701-610-6675
amoham19@uncc.edu

Jeremy Straub
Department of Computer Science
North Dakota State University
1320 Albrecht Blvd, Room 258
Fargo, ND 58102
701-231-8562
Jeremy.Straub@ndsu.edu

Abstract—**In order to provide secure transmission over a low-bandwidth, high latency data link, and the need for cyber-security must be properly addressed. This paper focuses on utilizing two ground stations with a satellite in order to transmit data over a wireless medium.**

The key issue of security arises when confidential information is being transferred, which can be breached by outside forces. By syncing two databases with one another, one in the sky collecting data, and two others on ground receiving data, secure connectivity is implemented.

To this end, a process utilizing AES encryption and decryption by using a secret symmetric key which is never shared with outside forces was developed and implemented. The system connects multiple databases. Database 1 generates cipher text which can only be read if it is decrypted. The encrypted message is securely transmitted and then decrypted with the secret decryption key. The original message is then inserted into ground stations: database 2 and database 3. If there is a sync problem or data loss, an error will be detected, and the two databases at ground station communicate with each other and keep the data in sync, reducing retransfer of lost data, and preventing any spoofing, or man-in-the middle attacks.

TABLE OF CONTENTS

1. INTRODUCTION

Satellite data transmission has become an essential, elegant form of finding, retrieving, and sending various types of data all over the world. However, satellite systems undergo some of the harshest environments on earth, and data has increasingly become corruptible due to many impeding factors. Satellite data can become tainted during the encryption process due to Single Event Upsets (SEUs) from radiation emission environments, and during transmission due to noise in the channel. Unmanned Ariel Vehicles (UAVs), and attacks from unauthorized entities attempting to conduct data breaches (Man-in-the-Middle attacks) are also potential threats.

Encryption is the process of turning plain data into cipher data. The reverse process is known as decryption. An encryption algorithm or cipher is used to attain privacy.

Satellites must contain the highest form of cryptography to protect data from any undetected changes during transmission or while in storage. Cryptography and fault detection play critical roles in combating these issues by hiding information, and securely transmitting data. There are various software programs which are designed to target these vulnerabilities, each with its own methods of distributing the keys that scramble and unscramble data.

2. BACKGROUND

RSA, AES, and DES Algorithms

Several encryption methods to use for satellite data transmission system were considered. Many factors were taken into consideration when choosing one of the many different cryptographic algorithms available. The three most popular of these include the Advanced Encryption Standard (AES), the Data Encryption Standard (DES), and the Rivest, Shamir, Aldeman (RSA).

1) DES was mainly adopted by businesses for security products. The algorithm design for encryption and decryption process is done with the same key.

2) AES is a symmetric key algorithm. Both the sender and the receiver use a single key for encryption and decryption. It is great for security and speed, with fast hardware and software implementation.

3) RSA is the most commonly used public key encryption algorithm. RSA is too slow for encrypting large volumes of data, but it is extensively used for to supply keys [1].

Key length, cipher type, block size, development, cryptanalysis resistance, security and time [1] were looked at.

978-1-5090-1613-6/17//$31.00 ©2017 IEEE

Cipher types Block cyphers are a central part of designing a shared-key cryptography. It has two inputs: one being a k-bit string and the other an n-bit string, and returns an n-bit string. The first input is the key. The second might be called the plaintext, and the output might be called a cipher text. The key-length k and the block-length n are parameters associated to the block cipher [6]. It should be designed to provide difficulty when there is an attempt to break the system.

To provide security, the block ciphers depend on the key length and block size. One block cipher is a k-bit string and the other an n-bit string, which returns an n-bit string. The former input is the key. The latter input might be called the plaintext, and the output might be called a cipher text as shown in Figure 1.

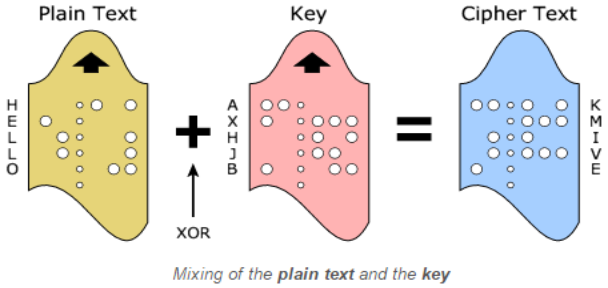

Fig. 1. Cypher Text [8]

According to the data found in a survey on performance analysis of the three encryption types, DES algorithms often have key distribution and key agreement problems, but have less power consumption. Meanwhile RSA consumes a large amount of time to perform encryption and decryption operations [1].

SAES Algorithm

The AES algorithms were found to consume the least amount of time for data encryption, decryption, and buffer usage compared to DES and RSA algorithms. AES also offers flexibility during and after implementation, which the other algorithms do not.

Encryption algorithms may be symmetric or asymmetric. AES uses a symmetric key algorithm for cryptography. This means that the same cryptographic key is used for both encryption of plaintext and decryption of cipher text.

More specifically, the AES algorithm uses a symmetric block cipher to protect information in software and hardware to encrypt sensitive data. The Public Key algorithms are used to perform the authentication and key connection, and then AES uses the symmetric algorithm to encrypt the data [3].

As shown in Figure 2, the sender encrypts the plain data with key and sends it to the receiver through an unsecured channel. The receiver decrypts the data, with the same key into its original form. The key must only be shared only using a secure channel by both the sender and the receiver [3].

Asymmetric key algorithms on the other hand, use a key pair. One key is used for encryption and the other for decryption. Either of the keys in a key pair can be used for encryption and the other for decryption. In this type of methodology, a public key can be made public for anyone to do the encryption, but only the owner of the private key can decrypt and read the message. For encryption, the input is a plain data block and a key, and the output is a cipher data block. For decryption, the input is a cipher data block and a key, and the output is a plain data block [2].

```
static SecretKey  keygen() throws Exception
{
    KeyGenerator KeyGen = KeyGenerator.getInstance("AES");
    KeyGen.init(128);

    SecretKey SecKey = KeyGen.generateKey();
    System.out.println(SecKey);

    return SecKey;
}
```

Fig. 2. Generating an EAS Symmetric Key

Many modes can be used to avoid faulty transmission. However, AES has one specific mode that can be used for ground stations, the CTR (Counter) mode. CTR is referred to as stream cipher mode as they do not require the whole block before encryption, only a partial block. Also, the CTR mode is more suitable for noisy channels because unlike other modes, cypher data bit transmission errors are not expanded in the received plain data. This mode is recommended as the optimum choice for satellite applications [4]. Based on these factors, AES is able to transfer data in the most efficient way [3].

3. Proposed Solution

The goal of this study was to ensure that data is transferred from a UAV satellite to two ground stations using secure and intact methodology. As mentioned before, a potential data breach that can occur commonly is referred to as a Man-in-the-middle attack. This happens if an attacker alters the communication between the sender and the receiver, and the receiver believes the data is being transferred over a secure, private connection, when in fact the communication is controlled by an attacker. Secure data transmission, with following goals, was desired:

1. Data being sent is received by intended user.
2. Detect and prevent Man-in-the middle attacks.

A security framework in which a new security layer encrypts data at the sender side and decrypts data at the receiver side is proposed.

4. Security Framework and Methodology

For secure data transfer, Application A was run at Database1 (UAV Satellite, sender), which is the only way to communicate with the database. The database is not open to

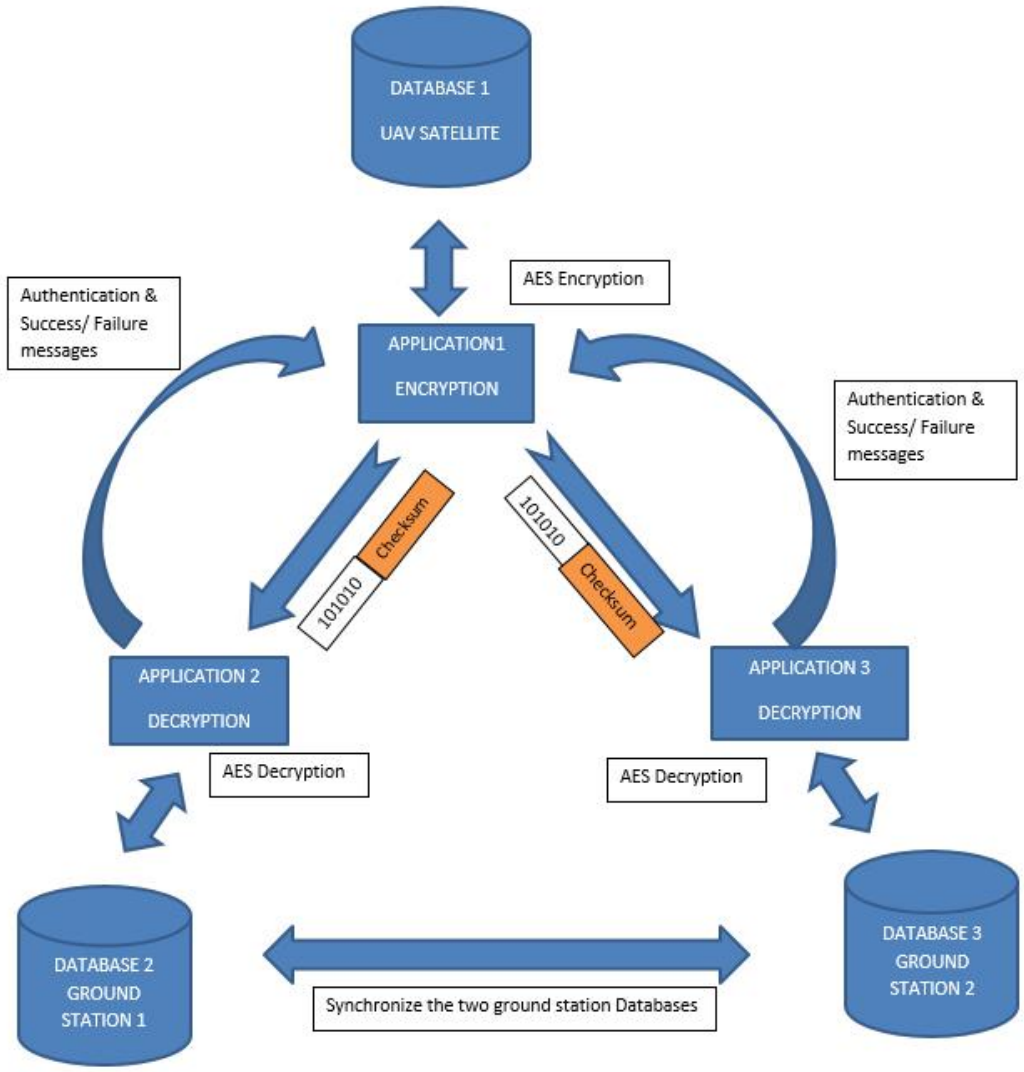

Fig. 3. Architecture Diagram

access by any other machine or application. This security policy shields the database access from outside forces. The application at the sender side will support features on the database limited to data extraction. The role of the Application A is to extract information from the Database1, and then encrypt. The data is then attached with a checksum and encrypted using AES secret key and read for transmission.

Similarly, two receivers at the Ground Station will have Application B, and C which will decrypt the data fetched from Application A. First, a checksum is cross-checked with the value attached with the transmitted data. If an issue arises, the data is considered incomplete and is then discarded. A failure message is then replied to Application A. In case of success, the data is inserted into databases and a success message is sent and inserted along with the index number. The index number helps Sender to send the next set of data and records to the receiver.

In the meantime, Application B and Application C communicate with each other. If there is a possibility that Application B gets the intact data while Application C gets corrupted data, Application B sends the data to Application C to synchronize both the databases. This will avoid resending of data in case of data loss or data corruption. If Application A does not receive the success message from at least one ground station, the data is resent. If continual data loss is detected, the secret keys, transmission ports, and frequencies are changed.

A Master Application which resides in a physically secure location on the ground is proposed. This Master application can control the database schema and structure of all the databases considered thus far. Only the Master application can create and change the structure of schema. It will not have privileges to insert or extract.

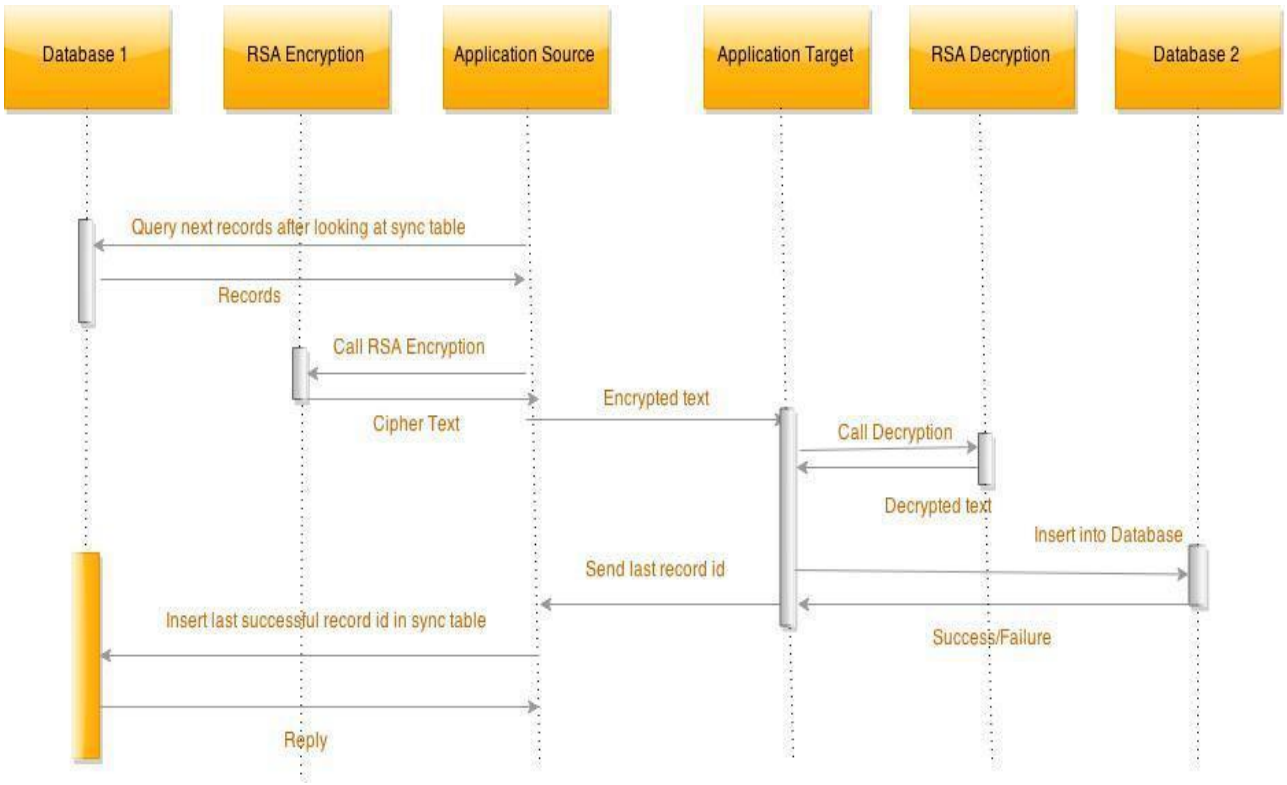

Fig. 4. Sequence Diagram

Applications A, B, and C act as a gateway for communication between the UAV satellite and ground stations as depicted in Figure 3. The purpose of these applications is not only to encrypt and decrypt data for transmission, but also to make sure that the data is sent to the intended user, and to detect any Man-in-the-middle attacks. Since Application A has to authenticate with Application B to make sure the application is asking for access to a trusted source, many options such as Single Factor Authentication, Multi-factor Authentication, and Cryptographic Authentication were considered. Any one of these authentication processes can be used, based depending on feasibility and further study.

For secured data transmission, socket programming is used. The data and records from Database 1 are encapsulated by a class (termed as Data Encapsulation in Object Oriented Programming) and encrypted using AES 128, which is a symmetric key encryption/ decryption algorithm by Application A. Then encrypted data is transferred to a receiver at the ground station. Application B at the receiving station ends the decrypted data with the AES decryption key, which is already shared between the applications.

In case of Man-in-the-middle attack:

If the data being transmitted is lost and never reaches the receiver, or the data sent is not received intact, and it is treated as spoofing by an attacker. The next steps would be to change the authentication, encryption decryption keys, as well as the sockets for data transmission. Now, the attacker cannot spoof any end user, since all the network access authorization and authentication processes are changed.

5. DATA MODEL

Satellites generate huge chunks of data every second, it can be atmospheric weather relay data, Geo Spatial images, and pictures and videos of distant stars and galaxies. The Database at the Satellite should have a structure in place for storing data from various activities. An Entity-relationship diagram for the UAV Satellite is shown in Figure 5, which is a structure for different the developed system. The ER diagram consist of an Orbital Status table, which includes Satellite_id as a Primary key, Orbital Status and Designation as other column schemas. Geography Dimension is another table which relays information such as Country_Code which is the primary key, Launching Country and Tenant Country.

The Satellite Info table consists of Satellite_id as primary key, geo id, Satellite Name, Country Operating and COSPAR number. The OperatorInfo table consists of Satellite_Id, Operator, Purpose, wavelength, frequency, and temperature. These tables map out various activities of satellite. The proposed application needs to be robust enough to provide the right set of data spanned across various tables.

In Figure 5 mentioned above, a separate entity is defined to store the geography information. This includes attributes like the country code, launching country of the satellite and the tenant country of the satellite. In addition, it captures the mission plan data for each satellite. This displays some attributes such as the name of the satellite, the operator, and

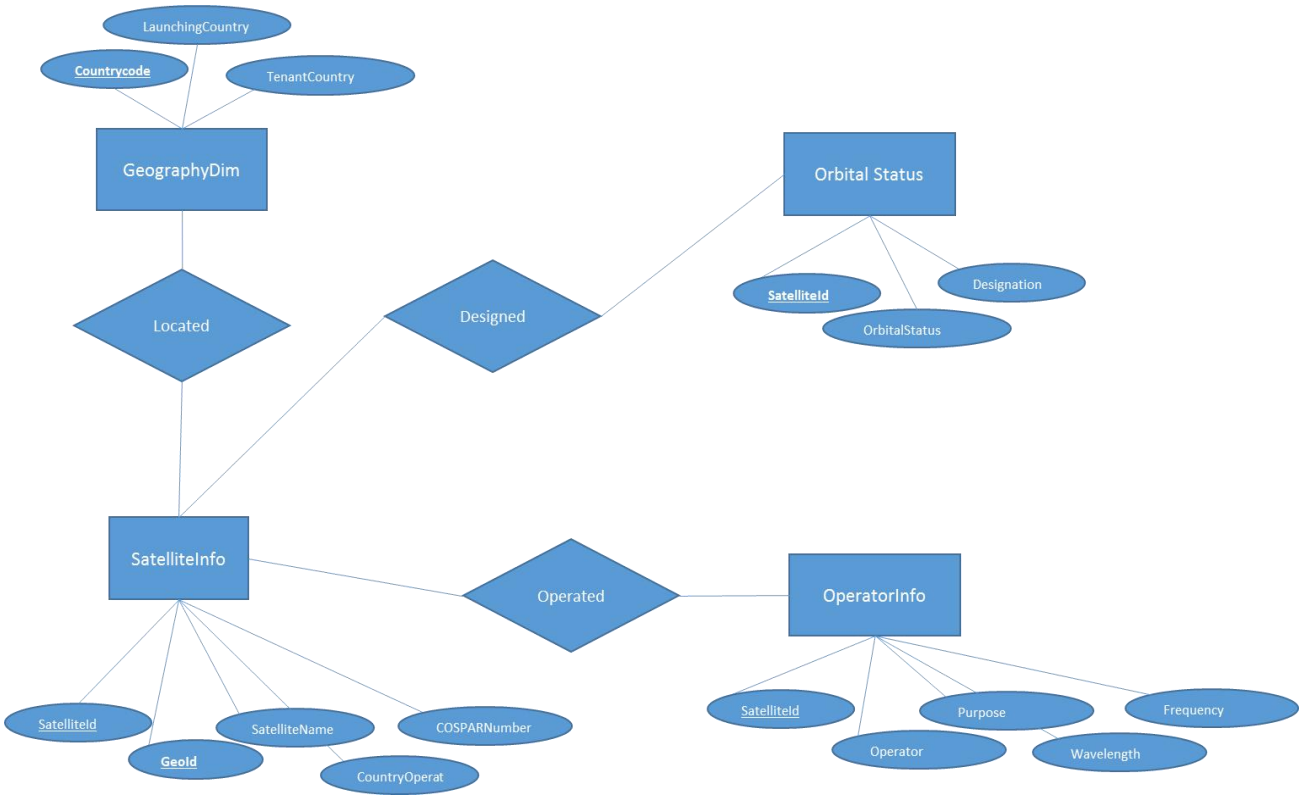

Fig. 5. Entity Relationship Diagram

a separate entity for Operator info. Finally, the ER diagram provides some attributes like the purpose of the launch, wavelength and frequency in which the information is transmitted; and stores the actual information that needs to be transmitted.

6. IMPLEMENTATION

As previously mentioned, the proposed application for the interface should interact with the Databases. The Application at the UAV satellite has to perform following functions:

1. Data Access from Database
2. Data Encryption
3. Checksum calculation
4. Data Transmission

Pseudo-code:

```
Function UAV_satellite ():
While (data exists & ackg_num received)
   {
      If (ackg_num not received from both ground
   stations)
      {
            Retransmit (data)
      }
      Else
      {
      /*       Get next set of n records from Database */
List_records = query_database (ackg_num,
   ackg_num+n)
Cipher = AESEncrypt (List_records, secret_key)
Checksum = md5 (cipher)
Transmit (cipher+checksum)
}
```

Listed below is an overview of functions for applications at Ground Stations:

1. Data Receiver
2. Checksum Verification
3. Data Decryption
4. Insert to Database
5. Transmit Acknowledgement

```
Function ground_station ():
While (incoming data exists && data received is not
   noise)
   {
            cipher = getcipher (data)
            checksum = getchecksum (data)
            If (md5 (cipher) == checksum)
```

```
                    then            "Data is intact"
                    data-records =AESDecypt
(cipher, secret_key)
                    rowid = database_insert (data-
records)
                    Transmit (rowid)
              else
                    /* Data is corrupted */
/* connect to another ground station and synchronize
data */
   records = Get_data_from_groundstation2 ()
   insert_into_ground_station1 (records)
}


Function database_insert (data-records):
    {
    /* Insert data-records into database if it's a new
data */
    db_insert (data-records);
    ackg_num = max (row_id);
    return ackg_num;
    }
```

8. Conclusion

The work presented helped finding suitable data encryption method for data transfer by high latency low bandwidth communication link (i.e., AES). A solution architecture for securing the data transmitted between satellites and ground stations with minimal retransmissions for data corruption and data loss was proposed. Mitigation steps in case of spoofing and man-in-the middle attacks were also discussed.

9. Future Work

Satellite communications are data intensive. Compression methods can be introduced before encryption. This can significantly improve amount of data transferred per second.

References

[1] B. Padmavathi, and S. Ranjikha, "A Survey on Performance Analysis of DES, AES, and RSA Algorithm along with LSB Substitution Technique," International Journal of Science and Research, vol. 2, Online ISSN: 2319-7064.

[2] A. Masoomi, and R. Hamzehiyan, "A New Approach for Detecting and Correcting Errors in Satellite Communications Based on Hamming Error Correcting Code," International Journal of Computer Theory and Engineering, vol 5, 2013.

[3] P. Sayeda, and R. Banv, "Satellite On-Board Encryption," Guilford,Surrey, UK Oct. 2007 University of Surrey School of Electronics and Physical Sciences.

[4] R. Banu, and T. Vladimirova, "Investigation of Fault Propagation in Encryption of Satellite Images Using the AES Algorithm," Proceedings of 25th IEEE Military Communications Conference (MILCOM 2006), 23-25 October 2006, Washington D.C., USA, pp. 1 – 6.

[5] O. Guerel, M. Cakir, U., "XMPP based applications under low bandwidth and high latency conditions," Lecture Notes on Software Engineering, 3(4), 2015, pp. 314-317, doi:http://dx.doi.org/10.7763/LNSE.2015.V3.211

[6] M. Bellare and P. Rogaway, Introduction to Modern Cryptography, May 2005, pp. 39 -56,

[7] S. Spinsante, and E. Gambi, "Selective Encryption for Efficient and Secure Transmission of Compressed Space Images," Department of Biomedical Engineering, Electronics and Telecommunications Universit`a Politecnica delle Marche, Ancora, Italy.

[8] Crypto Museum, The Vernam Cipher, Crypto Museum Website [Available Online] http://www.cryptomuseum.com/crypto/vernam.htm

Biography

Atif Farid Mohammad *is an MIT certified Systems Architect and currently an Adjunct Computer Science and Data Science Professor at the University of North Carolina at Charlotte. Atif holds Ph.D., in Information Technology from University of Quebec at Chicoutimi, Canada. Atif has more than twenty one years of experience in software engineering, professional business systems analysis, design, application development and staff management for diversified business and educational organizations. He is a member of IEEE, ACM and AST.*

Pamela Almeida *is a student at the University of North Carolina at Charlotte in the Data Science and Business Analytic program.*

Yasmin Soliman *is a student at the University of North Carolina at Charlotte in the Health Informatics Department at the University of North Carolina, at Charlotte*

Ajay Sadhu *is a student at the University of North Carolina at Charlotte in the Data Science and Business Analytic program.*

Keerthi Kata *is a student at the University of North Carolina at Charlotte in the Data Science and Business Analytic program.*

Jeremy Straub *is an Assistant Professor in the Department of Computer Science at the North Dakota State University. He holds a Ph.D. in Scientific Computing, an M.S. and an M.B.A. and two B.S degrees. He has published over 40 journal articles and over 120 full conference papers, in addition to making numerous other conference presentations. Straub's research spans the gauntlet between technology, commercialization and technology policy. In particular, his research has recently focused on robotic command and control, aerospace command and 3D printing quality assurance. Straub is a member of Sigma Xi, the AAAS, the AIAA, SPIE and several other technical societies, he has also served as a track or session chair for numerous conferences.*