Packet Cryptography Technique for Data Transit in Mobile Cloud Computing

Mr. Vikram Patalbansi[1], Dr. Jayshree Jain[2], Dr. G. Prasnna Laxmi[3]

1. Research Scholar, Pacific University, Udaipur, India
(Corresponding author:

2. Professor, Pacific University Udaipur

3. Principal & IT Head,
SDS College of Arts and Applied Science, Shreeramnagar, Vizianagaram District, AP, India

Abstract

High-quality wireless services are made available via pervasive networks like Mobile Cloud Computing (MCC), but their reliability is contingent on the robustness of the underlying wireless communication system's network security. Many studies have been conducted on the topic of developing security algorithms for wireless communication systems built with varying degrees of network reliability. Our proposed thesis paper details the theoretical development of a security system for the MCC Security Layer Protocol, which makes use of the SHA-256 cryptographic hash function to generate private keys for entities, the RC5 encryption and decryption algorithm, the Temporal Key Integrity Protocol (TKIP) to generate a dynamic sequential key, and the CRC-32 checksum to detect errors in our packets. Using the Diffie-Hellman Key sharing algorithm, the MSLP's stored symmetric secret key is used to generate a keystream for use in cryptographic operations. Before deploying on Mobile Cloud Computing, we save a private key to our database on the device, and that key never changes. To foil reply attacks and message integrity codes based on the addresses of sending and receiving devices and the contents of individual frames, these systems make use of a "dynamic initialization vector." Before data is sent across mobile networks, it is encrypted in the form of frames at the MSLP level and translated into its equivalent radio signals at the physical layer, both of which are analyzed in the proposed thesis paper.

Keywords: Mobile Cloud Computing, mobile network security, wireless signal security algorithm, packet cryptography, RC5

I. INTRODUCTION

Mobile Cloud Computing (MCC) connects mobile phones, laptops, and other devices to cloud servers over cellular networks. Mobile devices and cloud servers share resources and information wirelessly. The transmission and reception of secure information necessitate MCC security. Wireless communication is broadcast, so anyone may listen in. Wireless networks have radio communication and mobile endpoint vulnerabilities in addition to the many wired network vulnerabilities. Airlink packets are easy to capture and eavesdrop on them, insert malware, or execute DoS attacks. [1] Cloud communication providers must secure data and information as part of their service. New information computing technologies succeed or fail based on user security. Wireless communication requires secrecy, integrity, and availability from the service provider.[4]Once a synchronization header (preamble) is recognized, most wireless network receivers, including IEEE 802. x devices, start accepting airborne messages. Messages halt after a frame length byte. A collision during header receipt prevents reception. Mobile Cloud Computing needs packet encryption to avoid such issues.

[25]To protect the privacy and authenticity of information sent via a wireless network, a method called "packet cryptography" is employed. Each data packet is encrypted before transmission and decrypted upon arrival. Preventing malicious parties from gaining access to private data in transit is packet cryptography's fundamental goal. If the packets are encrypted, even if an attacker can intercept them, it will be impossible to read the data without the key.

Packet cryptography makes use of a wide range of cryptographic methods and protocols, including both symmetric and asymmetric encryption. In symmetric encryption, a single key serves as both the encryption and decryption for the data packets. The Advanced Encryption Standard (AES) and the Data Encryption Standard (DES) are both examples of symmetric encryption methods. However, in asymmetric encryption, two keys are used: one for encryption and one for decryption. Digital signatures and safe key exchange are two of the extra security features made possible by this technology. Elliptic Curve Cryptography (ECC) and RSA are two popular asymmetric encryption techniques.

Packet cryptography is used to secure wireless network communications and can be implemented at multiple layers of the network stack, including the network layer (through IPsec) and the transport layer (via TLS or SSL). Factors like as the network protocol in use and the intended level of security inform the decision of which encryption method to employ and at which layer to implement it. When it comes to wireless network communication, packet cryptography is crucial since it safeguards sensitive data from being intercepted while in transit and ensures its authenticity.

Data sent over a wireless network can be encrypted using a method called "packet cryptography." A key feature is its ability to encrypt data packets before their transmission over a network. Because of this, it is considerably harder for hackers to intercept the data and decipher it. Numerous methods of packet encryption are now in use. Some of the most typical examples include:

[23]Data Encryption Standard (DES) is a 56-bit key symmetric encryption technique. Although it has a poor reputation for strength, it finds widespread use.

The 128-bit, 192-bit, or 256-bit keys used by AES, the Advanced Encryption Standard, make it a symmetric encryption technique. It is the most used method of packet encryption because of its high level of security and widespread adoption.

[23]To encrypt data, public-key cryptography (PKC) employs

a pair of keys—a public one and a private one—to create an asymmetric encryption method. While anybody with access to the public key can encrypt data, only those with access to the private key can decode it. Given how challenging it is to obtain the private key, PKC is a particularly secure protocol.

Security for wireless networks is not possible without packet cryptography. It aids in preserving data privacy, integrity, and accessibility by blocking unauthorized access to it.

Using a combination of the DES and RSA algorithms, the authors of the study you reference propose a hybrid cryptography solution for usage in MANETs. After simulating the suggested method, we find that it significantly boosts the safety and performance of MANETs.

The use of packet cryptography in wireless network communication has several advantages, including the following:

[5]In terms of data secrecy, packet cryptography is a useful tool for preventing unauthorized access. This is vital for the security of private information like bank records and social security numbers.

Data integrity: packet cryptography can aid in preventing data from being altered in transit. For data accuracy and trustworthiness, this is crucial.

The availability of data is improved through the use of packet cryptography, which helps to prevent data from being corrupted or deleted. The availability of data at any time necessitates this. If you want to keep your wireless network safe, packet cryptography is a must. It aids in keeping data safe from prying eyes, maintaining data integrity, and guaranteeing data availability.

[26]Security for data packets being sent across an unencrypted wireless network is provided through packet cryptography. These methods apply cryptographic algorithms and protocols to ensure that the data in the transmitted packets remains private, legitimate, and undamaged. The following are some examples of packet cryptography that are frequently implemented in wireless networks:

Using an encryption algorithm, plaintext information can be transformed into ciphertext.

1. Encryption is used in wireless networks to safeguard data from prying eyes and ears. Only authorized recipients with the appropriate decryption keys can read the encrypted packets.

2. Authentication: Authentication guarantees the truthfulness and credibility of the communication parties. To ensure that the data being sent over a wireless network is legitimate, authentication procedures like digital signatures or message authentication codes (MAC) are utilized. Because of this, the communication is less likely to be intercepted by malicious parties or modified by unauthorized users.

3. Securely establishing encryption keys between wireless devices is accomplished using key exchange protocols. These methods allow cryptographic key exchange between several parties across a potentially vulnerable wireless channel. Diffie-Hellman key exchange and its variations are widely used as standard protocols for key exchange.

4. Fourth, the data in the packet is hashed using a secure algorithm to produce a fixed-size hash value, such as SHA-2 or SHA-3. The receiver can check the packet's authenticity by comparing the hash value with a known good copy of the packet. The hash value changes if the packet is altered in any way during transmission, which can be used to detect attacks or transmission mistakes.

5. Secure Tunneling: Virtual private networks (VPNs) and Secure Sockets Layer/Transport Layer Security (SSL/TLS) are two examples of secure tunneling technologies that provide an encrypted tunnel between mobile devices and the network backbone. [20]The tunnel encrypts all data packets sent via it, making the wireless network more secure for confidential conversations. To protect critical business communications and financial transactions via wireless networks, among other uses, these packet cryptography approaches are important. Encryption cracking, packet sniffing, and assaults on the primary protocol for secure wireless communications, such as WEP, are all mentioned as popular methods of attacking packet cryptography in wireless network communication. Encryption cracking is the process of trying to decipher the code that protects data transmissions over an unsecured wireless network. Attackers can gain access to private data by packet sniffing, which involves the interception of unencrypted transmission. Wireless network security can also be broken if the original protocol for secure communications via wireless media is attacked, as was the case with WEP. Note that more study is required to identify other frequent assaults on packet cryptography used in wireless network communication.

The Pros and Cons of Using Packet Cryptography in Wireless Networks

There is a lack of detail about the benefits and drawbacks of using packet cryptography in wireless network communication in the search results. [25]Cryptography is the practice of encoding and decoding information so that it can be securely transmitted and read by only the intended recipients. Information and communication security, authentication, and restricted access for authorized parties are all bolstered by cryptography's ability to turn source data into a code that can be deciphered only by the intended recipient(s). Despite this, the search results do not detail the benefits and drawbacks of using packet cryptography for wireless network communication. More study is required to fully understand the benefits and drawbacks of using packet cryptography in wireless network communication. [23]Network packets can be encrypted using a variety of methods, including RSA, AES, Advanced Encryption Standards, Transport Layer Security, Internet Protocol Security, and Pretty Good Privacy (PGP). When used to encrypt and decode data, these methods keep network packets safe from prying eyes. Encryption techniques and keys are used to transform plaintext information into ciphertext, which can then be read back into its original form only by those who have been granted access. The Advanced Encryption Standard (AES) technique is especially widespread and is used to secure a great deal of data at rest and in transit. TLS/SSL, IPsec, SSH, and even PGP rely on it for security.

[23]RC5: The RC5 cipher algorithm encrypts data in a symmetric stream, one byte at a time. RC5 algorithm mathematical stages are as follows:

1. First, a 256-byte state vector S, containing elements S to S, is initialized with a key whose length can vary from 1 to 256 bytes. During encryption and decryption, a byte k is created from S by randomly selecting one of the 255 entries and then permuting S once more.

A Key-Generation Algorithm for

2. A key input pseudorandom bit generator that outputs an unpredictable stream of 8-bit numbers without knowledge of the input key. Key-stream is the output of the generator, and it is joined byte-by-byte with the plaintext stream cipher via the X-OR operation.

3. Encryption: RC5 creates a stream of bits (a keystream) that is statistically and cryptographically equivalent to a random

stream of bits, allowing it to encrypt data. These, like any stream cipher, can be used for encryption by bitwise exclusive oring with the plaintext.

Since the exclusive or with provided data is an involution, decryption is conducted in the same way as encryption.

Even though RC5 is noteworthy for its simplicity and speed in software, it has been found to have several weaknesses. As a result of these security issues, its use is discouraged in new projects.

Pros: • Can be implemented in either software or hardware; • Algorithm is simple and fast.

Inadequate for new designs because of its sensitivity to attacks like the Fluhrer-Mannin-Shamir attack.

It is crucial to select an algorithm that meets both security and performance needs. Even while Blowfish encrypts data more quickly than most other algorithms, it may not be a good fit for uses that necessitate frequent key re-entry. In contrast, RC5 is easy to implement and quick to use, but it is vulnerable to certain attacks.

The RC5 algorithm is described in broad strokes here:

Step 1:In the first place, RC5 keys can range in size from 40 bits to 2048 bits. Using the key, we populate an array of permutations labeled "S" with numbers from 0 to 255. The key is used to determine an initial value sequence for the permutation array.

Step 2: Two indexes, "i" and "j," are initialized to 0 during the setup phase. During encryption, these indices are utilized to exchange elements of the permutation array.

Step 3: Scheduling of Keys: RC5 creates a keystream by iteratively altering a permutation array. To generate new permutations, we iteratively swap adjacent elements of the permutation array using the current key and the prior permutation array state. The keystream is created using the values at the newly assigned indexes.

Step 4: Encryption RC5 encrypts a network packet by first generating a keystream of bytes, as explained in step 3. The plaintext packet is then merged with the keystream via a bitwise XOR technique. A packet of ciphertext is then transmitted over the network.

Step 5: Decryption: To recover the original plaintext from a ciphertext packet, we first produce the same keystream using the RC5 technique as we did during encryption.

There are some safety issues and loopholes with RC5, especially when it is used in insecure ways or with weak keys. Therefore, it is often advised to employ more up-to-date and secure encryption methods like AES (Advanced Encryption Standard), which offers more assurance of safety.

II. RELATED WORKS

[1]The authors of this work provide a solution to the problems of privacy, data transmission vulnerabilities, and the required use of public packets that protect users' anonymity. They propose that the present interval after which a device changes its public identification can be exploited by timing-based attacks on the randomization of MAC addresses on any wireless network. When we randomize more often, more devices in the population will likely change their MAC addresses at the same time, increasing the likelihood of conflicts. In general, the higher this value is, the more challenging it is for an attacker to link together multiple MAC addresses from the same device. Because of the public nature of packet transmission, it is necessary to employ fingerprinting and user profiling to ensure packets' safety.[4]In this study, the authors explored the problems that arise in a communication system anytime an unauthorized user can access the network system, thereby posing a threat to the security of the network. A technology called frequency hopping was developed to encrypt packets before they were sent through networks. Frequency hopping is a method used to protect wireless networks from brute force attacks, offline dictionary attacks, and online dictionary attacks by preventing the full capture of the four-way handshake. Using a combination of the WPA2/PSK protocol and some additional security protocol, this solution can be readily implemented to the existing devices with just a firmware update. While the suggested method guarantees that only authentic users can join a wireless network, it does not discuss any security arrangements to be made while the signal is in transit. [5]All Aggregator Nodes (ANs) and Sensor Nodes (SNs) are authenticated making use of blockchain property, and this study proposes an encryption and trust evaluation model based on a blockchain in which their identities are maintained to prevent illegal behaviors by inauthentic nodes. False data is transmitted and extra resources are used because of malicious nodes. Network nodes' trust values are used to accomplish secure routing. Rivest-Shamir-Adleman (RSA) is a crypto scheme used by authors that use asymmetric keys to encrypt data during transmission.

[7]This research proposes a symmetric cryptography technique for protecting physiological signals, using a chaotic map and a multilayer machine learning network (MMLN). To extend the security of private cipher codes, a chaotic pseudorandom number generator beneath certain control parameters can powerfully create unordered arrangement numbers to set the mystery keys for a intermittent mystery key overhaul. The private cipher codes are determined by the starting and specific control parameters at the data emitter and receiver ends, and the chaotic map is rapidly integrated to provide a pseudorandom key stream for real-time applications. To ensure the privacy of physiological signals transmitted over a wireless network, an encryptor and decryptor based on recurrent neural networks (RNNs) were proposed. An electrocardiogram (ECG) signal sequence was converted to a series of digital values between 0 and 255 using a digital signalling procedure, and then the resulting digital data were chaotically permuted using secret keys generated by a logistic map function. [9]This research proposes a secure method of data clustering that can be used to lessen network traffic while also cutting down on energy use. There is now a secure technique for data clustering. This efficient cluster selection procedure is supported by cluster head helpers, centralised management by the base station and the efficient use of Blowfish-EAX-RSA.

[14] According to the authors, the key to securing physical-layer communications is in using the characteristics and limitations of the propagation medium. Research conducted by the authors analyses physical layer security for a variety of 5G enabling technologies, including massive MIMO, millimeter wave communication, heterogeneous networks, non-orthogonal multiple access, and a full duplex. PLS technique about new network scenario unmanned aerial vehicles (UAV), enhanced mobile broadband (eMBB) URLLC, massive machine type communications (mMTC), and vehicle-to-everything (V2X) networks must be studied to provide for the expanding service.

[16]In this paper, we assume that encrypted and unencrypted Zigbee networks' packet delivery ratios (PDRs), power consumption, network lives, link quality, latency, and

throughput can be optimized for application deployment over short-range wireless networks. The costs and benefits of multi-hop transmission and encrypted data storage are also highlighted. To sum up, we looked at how changing factors such as transmission power, transmission distance, packet size, baud rate, deployment situation, and data encryption affected performance measures like PDR power consumption, link quality, latency, and throughput. [17]Many technological concerns were revealed by the combination of IoT and WSN use, including data security, multi-sensory, multi-communications capabilities, energy utilization, and the information age. The authors employ the hybrid Vlsekriterijumska Optimizacija I Kompromisno Resenje (VIKOR) approach to information pooling and CH selection. After a cluster has been constructed using a genetic algorithm (GA), the data in it are aggregated. After collating, the data are encrypted and sent via a secure channel using the TIGSO-EDS framework. Using the Cuckoo Search Algorithm (CSA), the best route was determined, and then data were sent to the BS for analysis

[18]The authors of this study put forth a theoretical framework for security solutions based on the Open System InterConnection (OSI) layer, and how they might function in a 5G network setting. The fifth generation of communication networks provides an increased variety of services above their predecessors. Vulnerabilities, threats, security solutions difficulties, and security gaps exist at every layer of the OSI model. The authors addressed all aspects of security at all layers, including the physical layer. Data security and integrity in 5G networks depend on all 5G levels cooperating and contributing their specialized technologies. Although application-based encryption safeguards the apps themselves, it is insufficient to safeguard data traveling through 5G mobile networks due to power leakage in wireless signaling. While cryptographic techniques have been effective in the past, a more practical answer can be found in the realm of physical layer security for 5G networks. [19]The authors of this study suggest the concept of a "wireless sensor network," or WSN, which consists of a collection of interconnected, low-power electronic devices. The authors propose MAC layer security techniques that strike a balance between packet throughput and power consumption. The authors examine, evaluate, and compare the security features of two different MAC protocols (BMAC and LMAC). They used a variety of cryptographic methods, including AES, RSA, and ECC, to protect their networks from attacks that exploit the MAC protocol.

III. PROPOSED THEORY

[21] Security concerns in Mobile Cloud Computing (MCC) can be broken down into three categories: device/terminal security, network/communication security, and cloud infrastructure security. The topic of wireless communication channel security will be discussed here. Through the use of various channels of communication or wireless interfaces, mobile users in Mobile Cloud Computing can interact with cloud server nodes. Traditional encryption and authentication methods have many weak points that can be exploited by malicious actors. Most attackers are familiar with these methods, and with a little bit of research, they can simply break through security measures like access control assaults or confidential attacks.

There are three main types of security breaches: integrity, authentication, and availability.

[24]Our self-defined protocol architecture, the MCC Security Layer Protocol (MSLP), is proposed as a means to realize the foregoing goals. To encrypt and decode data on mobile devices, our MSLP provides a multi-layered security architecture developed during chip fabrication and placed on-chip. The information stored on the chip is immutable due to its unique identifier and the fact that the debug read-back feature has been deactivated, which safeguards against the typical weakness of reverse engineering using ordinary debugging tools.

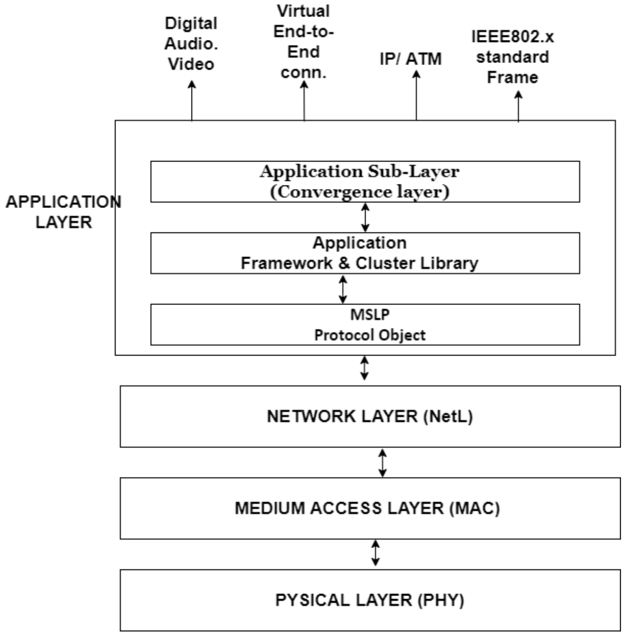

Fig. 1 MCC Security Layer Protocol (MSLP)

In certain ways, the MCC Security Layer Protocol operates like the TCP/IP protocol layer of an internetwork. Here, we employ our Mobile Cloud Computing protocol to retrieve data from cloud servers to our mobile electronic devices via cellular (mobile) networks, and vice versa.[20]The primary goal of this protocol layer is to safeguard data during transmission across a wireless network. The services we expect to access on the go include video, audio, photo, digital content files, etc. The number of bits in a frame dictates the type of security methods we need for these services.

The A) Application Layer is the layer that communicates with the outside world contribution from the user. There are three distinct sections of the application layer. i) Services: [18] Many services, such as the transmission of digital audio and video between mobile devices and cloud servers, are necessary for the widespread use of mobile technology. IP service provides internet service by producing IP datagrams, and ATM service provides logical distribution of frequency spectrum over network slices. MSLP also generates frames compliant with the Wi-Fi and Wi-max networks' standards, the IEEE 802. x protocol suite.

[18] This layer is responsible for ensuring that the services supplied by the mobile network continue to operate at a high standard of quality (QoS). This layer's function is identical to that of the presentation layer in the OSI reference model, hence it is referred to as layer iii (Application Framework and Cluster Library). This layer takes the data entered by the user at the application layer, turns it into a machine-dependent format (binary digits), and then compiles it with the help of a built-in cluster library. The sublayer generates frames or packets after double-checking that they are in the correct format.

The MSLP Protocol Object communicates with the authentication application on Mobile Devices to retrieve the secret key generated by the cellular network by the Mobile Equipment Identity (IMEI), Mobile Equipment Identity (IMEI) (or MAC address in the case of electronic devices other than Smartphones, etc.), fingerprint, virtual smart card, and universal integrated closed circuit (UICC) by hashing with a random number generated by the registration center of the Cellular Network. In our earlier work, we elaborated on this idea in full detail. VIdent represents that private identifier. Using this private key, the MSLP Protocol Object encrypted our frame. The following diagram illustrates the process by which VIdent is produced.

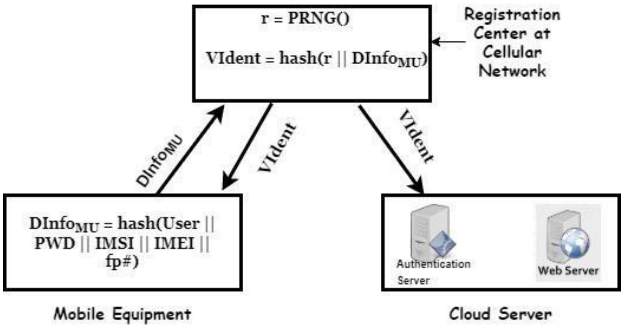
r = PRNG()
VIdent = hash(r || DInfo$_{MU}$)
Registration Center at Cellular Network
DInfoMU
VIdent
VIdent
DInfo$_{MU}$ = hash(User || PWD || IMSI || IMEI || fp#)
Authentication Server
Web Server
Mobile Equipment
Cloud Server

Fig.2 Generation Pair-wise Transient Key VIdent

A) Our protocol MSLP's Network Layer (NetL) facilitates the generation of packets or frames that are suitable for use in a wireless network environment. Frames are created at the network layer based on the network's configuration. Likewise, the network layer keeps track of the frames' routing information by broadcasting a router request message and processing a route reply message. By creating session keys and verifying communication on both ends, it also established a logical link between mobile devices and cloud servers.

[23]Using Internet Protocol Security (IPSec), an IETF-standardized protocol, we can apply the network security process to the frame that has traveled from the Application Layer to this Network Layer. Together with our MSPL network layer protocol, it can also produce a frame. IPSec does not require any special configuration on the part of either network administrators or end users. With IP over cellular network encryption, the procedures are transparent on both ends, and the packets are encrypted locally before transmission.

B) A) MAC Layer: Every device on a network, whether wired or wireless, is required to have a unique identifier called a MAC address, or a physical address, under IEEE standards. MAC addresses are programmed into electrical mobile devices at the time of manufacture. The IEEE is the authoritative registry for all electronic gadgets, and all devices may be checked against each other online in terms of their wireless network settings. According to our MSPL protocol standardization, the MAC layer is in charge of its security processing, but the upper layer Application Layer logical subparts of MSPL protocol objects read the unique authenticated key VIdent with the help of the authentication application, and these authentication keys determine the encrypting key to frame or level of security to use. This VIdent authenticated key is an active network key because the authentication app releases the associated memory as soon as the device is no longer connected to the cellular network or cloud servers. The MAC layer also provides error detection and repair for frames during transmission and reception. The MAC layer incorporates logical circuits for encryption, decryption, error correction, and decryption of frames to improve MSLP's functionality, and the MAC layer also incorporates the WAP2 protocol, a technical standardization for gaining access to data via mobile wireless networks, to generate IP datagrams for use in internet communications. Multiplexing, channel scheduling, header compression, reordering of packets, retransmission of lost packets, and all cryptographic protection to access stratum signals (integrity, confidentiality, etc.) are all handled by the MAC layer.

Layer C) The Physical One: [10] Evolved Universal Terrestrial Radio Access Network (E-UTRAN) is the collective name for the many Radio Access Networks (RANs) that make up cellular networks. The Evolved Node B (eNodeB), which is part of the Evolved Universal Terrestrial Radio Access Network (E-UTRAN) and is responsible for modulating and demodulating radio signals to communicate with mobile devices or user equipment (UE), is an intermediate node in the Mobile Cloud Computing architecture. The cellular network base station, also known as an eNodeB, is responsible for creating and relaying IP packets to and from the internet. The primary function of the Physical layer in our MSLP protocol suite is to link mobile devices to the eNodeB. Through the use of a network adapter, it translates the frame or packets from the higher layers into the radio signals necessary for transmission across RAN. The over-the-air interface, MSPL protocol, primarily serves two purposes through its usage of two logical operational planes: the user plane and the control plane. The user plane is responsible for the transmission of user data such as voice conversation, SMS, and application traffic. As for the connection between mobile devices and the eNodeB or mobile network base station (BS), it is handled by the control plane. Non-Access Stratum (3GPP) and Access Stratum (3GPP) are the two logical layers that make up the working protocol of a cellular network. The Access layer is concerned with the transmission of data through radio frequency (RF) channels by mobile devices. All communications between mobile devices and the cellular network backbone that isn't radio signaling make up the NAS. The Physical Layer of the MSPL protocol suite is responsible for establishing, maintaining, and disabling the air interface connection between mobile devices and mobile (cellular) network entities. This is done through the execution of several control tasks, including the dissemination of system information, the establishment of a connection with eNodeB, the transmission of paging signals to the various base stations, authentication with the registration center of the cellular network, the bearing of bearer traffic, and the termination of the connection.

IV. SECURITY ARCHITECTURE

Step 1) First, the MAC and Application layers are where cryptographic techniques are deployed, making them crucial to the security of wireless communications. The physical layer and the Radio Access Network are both protected by the safeguards we provide. Given the ease with which wireless communications can be eavesdropped, our data must be transmitted in an encrypted format.

Our MCC Security Layer Protocol (MSLP) is a layered protocol based on the IEEE 802.15.x standard that offers cryptographic security between mobile networks and their organizations. The Application Layer is the topmost layer of a mobile application stack, and it is responsible for providing a user interface and transforming data entered by the user into a format that is suitable for the hardware configuration of mobile devices. Information transmission via a wireless communication network is then supported by the Network layer, MAC layer, and Physical layer. When communicating over Mobile Cloud Computing (MCC), we will use dynamic encryption technology to ensure that each session's worth of data is encrypted using a unique set of encrypting keys generated by a secure asymmetric key algorithm. [23]Here, the frame between mobile devices and the cloud server over a cellular network is encrypted using Diffie-Hellman key exchange methods to exchange encrypting keys and Wired Equivalent Encryption (WEP) protocol based on RC5 PRNG (Ron's Code 5 Pseudo-Random Number Generator developed by Ron Rivest. To maintain a consistent data rate during transmission, the frames or packets are the smallest unit of data that can be sent. Each packet's source bit rate is entirely possible.

Step 2) The kreyvium encrypts the data using the sender's public key, as is customary in symmetric key encryption procedures, and the receiver decrypts the encrypted data using the same public key and sends the data and a public encryption key to the recipient. Using a public key that was also sent by the sender, the receiver can now read the encrypted message. Eavesdropping is a major flaw in this approach since it allows a third party or hacker to intercept the transmission session and decrypt all of the data because they have access to the encryption key at once. To get over these issues, we need to employ the Diffie-Hellman key exchange protocol to trade secret information through Mobile Cloud Computing. As a result of using secret keys that are calculated with the aid of the Diffie-Hellman Key Exchange protocol, the information exchanged between the sender and recipient is public, meaning it can be hacked by anybody but cannot be decoded. In this case, we'll employ asymmetric encryption (the public-private key principle) to safely transfer the secret information between users' key VIdent and VIdent are generated by the RC and shared with mobile devices and cloud servers, and then saved in the memory of those devices and the servers. All Mobile Cloud Computing entities around the world are familiar with VIdent.

Step 3) Here, we'll use the [23]cryptographic Secure Hash Algorithm 256 (SHA-256), which generates hash values that are difficult to anticipate from the input, to generate the private key of the mobile devices (UE) randomly. Many high-level programming languages include pre-built libraries that make it easy to put the SHA-256 algorithm to work for us. Using SHA-256, the generated sequence of random integers is correct and unpredictable in light of the input data.

Now in mobile devices, an authentication application gets installed which is provided by the cloud service providers to get cloud service from them. First mobile user, login with the authentication application by inserting a valid username and password. The mobile devices also have unique parameters such as IMSI, IMEI (in the case of laptop&i-Pod MAC address according to IEEE standardization),

Using the biometric module already present in most smartphones, we can randomly generate the UICC's corresponding string parameter (fp#) based on the user's valid fingerprint (fp), and then use the cloud service provider's Fuzzy extractor function to assign that fingerprint to the UICC. This UICC is like a hardware chip embedded into the phone's extensible hardware slots.

User, password, IMSI, IMEI, UICC, and fp# are all assumed to be the seed values used by the hash function to generate the necessary value for the mobile device's user in that session. When starting a new session, we need to come up with fresh seed values.

Consider

DInfo_{MU} = hash(user || pwd || IMSI || IMEI || fp# ||UICC)

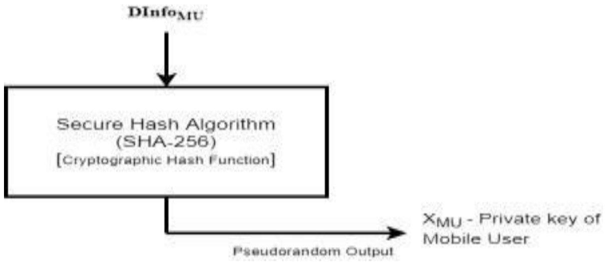

Fig.3 Private Key generation of Mobile user

$$X_{MU} = \text{SHA-256}(\text{DInfo}_{MU});$$

To generate the necessary private key for the user, a pseudorandom number generator (PRNG) needs two things: a seed value and a deterministic technique for generating a stream of pseudorandom functions (PRFs). We employ the SHA-256 and DInfo_{MU} algorithms in this case.

Similarly, a Cloud Server (CS) private key can be made. In addition to a username (userCS) and password (pwdCS) for a valid cloud server administrator, an authentication application is installed on a cloud server to take the IP address of the cloud machine (IP) to access the internet, the MAC address of the cloud server(approved by the IEEE registration authority and it can be cross-checked online on IEEE), and the port number (PORT) over which the application(web) server is running in the cloud.

DInfo_{CS} = hash(user_{CS} || pwd_{CS} || MAC ||PORT || IP)

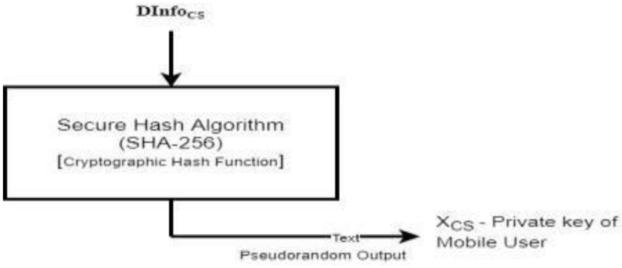

Fig.4 Private Key Generation of Cloud Server

$$X_{CS} = \text{SHA-256}(\text{DInfo}_{CS});$$

Y_{MU} and Y_{CS} are the public keys of mobile devices and cloud servers, respectively, and are therefore known to all parties involved in Mobile Cloud Computing.

1. In this case, we'll suppose that q is prime and that is the primitive root of q2. The values " and "q" are agreed upon by both terminals and are widely known.
2. The public key for both the mobile device and the cloud server can be calculated using the Diffie-Hellman Key Exchange formulae.
3. $Y_{MU} = \alpha^{XMU} \bmod q$………for Mobile Device And
4. $Y_{CS} = \alpha^{XCS} \bmod q$… ....for Cloud Server.

Public keys Y_{MU} and Y_{CS} are shared across mobile devices and cloud servers, but private key values X_{MU} and X_{CS} are guarded by each.

Next, we use the Diffie-Hellman Key Exchange formula to determine the packet's secret key (SKM) when mobile devices are the packet's originator.

$SK_{MU} = (Y_{CS})^{XMU}$ mod q ….Secret key for packets at mobile devices during sending of packets.

$SK_{CS} = (Y_{MU})^{XCS}$ mod q …. Secret key for packets at cloud server during sending of packets.

The Diffie-Hellman Key Exchange protocol specifies that the secret keys at both ends must be identical. Our proposed methodology relies heavily on this premise. so, $SK_{MU} = SK_{CS}$ (through the Diffie-Hellman formula) RC5 Encryption and Decryption protocol is factory-fabricated and programmed into the Medium Access Layer(MAC) of MSPL protocol to encrypt every packet and perform error detection on every packet using the secret values SK_{MU} and SK_{CS}.

The following protocols and algorithms are used in encrypting and decrypting the packets coming from the upper layer of the MSPL protocol.

i. First, the Temporal Key Integrity Protocol (TKIP) can improve the security of wireless communications between mobile devices without requiring any changes to the underlying hardware. TKIP is a protocol used as a preprocessor for the RC5 encryption technology. To encrypt each data packet with unique encryption using robust values, the new TKIP protocol uses a TKIP Sequence Counter (TSC) to generate a Dynamic Initialization Vector (DIV), which is then used as a seed for the RC5 encryption technique. TKIP additionally supports the subsequent algorithms to boost the key strength values.

ii. Improved packet integrity is provided by using Message Integrity Check (MIC) in addition to CRC-32. The MICHAEL MIC depends on more than simply the information included in the packet. Additionally, the PTK(pair-wise transient keys), MAC addresses of both the sender and the receiver, packet priority, and other factors come into play. Here in our protocol VIdent - pairwise transient keys are already calculated after mobile device registration in the cellular network. MICHAEL is built to forego these recursive measures of guesswork and bit-flipping

iii. Increased cryptographic strength is achieved by executing the key mixing function for each packet.

iv. Hashing function cryptography makes use of the TKIP Sequence Counter (TSC), which allows the Dynamic Initialization Vector (IV) and key ID fields to grow to 8 bytes. Each packet's sequence counter is unique, making this a handy tool for thwarting replay attempts.

v. A re-keying mechanism to provide new key generation after every 1000 packets.

When computing the MIC (Message Integrity Check), the MICHAEL only employs shifting and adding operations to generate a short check word, eliminating the need for multiplication. All of the bytes in a message frame are added together by MICHEAL, and a check value of 8 bytes is then generated and appended to the frame at the end, along with the original message, before being sent to the receiving devices. This check value is known as the MIC (Message Integrity Check). To calculate the MICHEAL MIC, a special on the reversible method is combined with the private key of the mobile devices. Most mobile devices and cloud servers are already familiar with the PTK (pair-wise transient keys), also known as the VIdent. To counteract packet re-routing attacks and strengthen the source validation process, the MIC calculation now takes into account both the destination and source MAC addresses of the packet. It is attached to the back of the frame to prevent any tampering. Only the one true receiver has access to the paired temporary key Ident, and hence the true check value MIC may be recalculated. Thus, MIC is a form of message authentication code employed in the detection of packet forgeries.

vi) To prevent the same sequence number from being used twice in a single session, the TKIP Sequence Counter (TSC) allocates a packet sequence number to the MIC and dynamically increments to a new sequence value for subsequent packets using the Dynamic Initialization Vector (DIV). All packets with a sequence number beyond the common range between the sender and the receiver are rejected.

vii) Ron Rivest created the RC5 Encryption Algorithm, which uses a symmetric key block encryption technique. Compared to RC5, it is easier to use, requires less memory, and performs computational operations like XOR, shift, etc. quickly.

Key expansion, encryption, and decryption are the three methods that makeup RC5. There are essentially three basic phases to every algorithm.

(a) The "+" symbol indicates a two-to-a-word addition. That's addition modulo 2w.

Words can be bit-wise ORed () together in (b).

A word turn to the left, or a left spin.

x y represents a rotation to the left of x bits in the word x. For y to be interpreted modulo w, just the lowest lg(w) bits are utilized to calculate the amount of rotation.

The RC5 is a block cipher that can encrypt two words simultaneously. Word size (w), rounds (r), and key size (b) all contribute to the definition and expression of RC5, which may be written as RC-w/r/b. Both the key size and the maximum number of rounds can be anything between zero and 255 bytes. There will only be three possible sizes for RC5 blocks or words: 16 bits, 32 bits, and 64 bits. Because of this, the maximum size for a plaintext block is only 32, 64, or 128 bits, while RC5 employs a 2-word block.

V . THE PROPOSED METHODOLOGY

Our first premise is that the mobile device secret key (SK_{MU}) and the cloud server secret key

(SK_{CS}) are equivalent to the Diffie-Hellman key exchange scheme. Therefore, $SK_{CS} == SK_{MU}$.

None of the entities involved in Mobile Cloud Computing (MCC) ever have access to either of the secret keys. Registration Centre (RC) computes VIdent based on parameters sent by mobile devices; RC then communicates this value to the mobile device and the cloud server, which each keeps a copy of it in their separate memory file systems.

In this case, we can define VIdent as a pair-wise temporary key (PTK) between mobile devices and cloud servers. When a connection is made, the two devices exchange their respective MAC (physical) addresses. For a mobile phone, the IMEI number can be used in place of the MAC address. MAC_{MU} and MAC_{CS} are both addresses. When a mobile device and cloud server are successfully connected, the TKIP Sequence Counter (TSC) creates a dynamic initialization number (DIV) that is incremented with each MAC Service Data Unit (MSDU) packet. DIV's interval is completely arbitrary, beginning with a number chosen at random and ending with the same. r represents this DIV interval.

As a first step, the unit doing the sending—here, mobile phones—encrypts the range and sends it to the device doing the receiving—a cloud server. Take the following algorithm as given. In the following algorithm, we employ the 3DES technique to encrypt{ r}.

```
Algorithm 1: Sending Range of Dynamic Initialization Vector.
Requirement :
  isAvailable(mobile device , mobile network,cloud server,TKIP_Sequence_Counter)
  hasNetworkAccess(mobile device, cloud server)
Procedure :
                  Role_Of_Mobile_Device
  const $SK_{MU}$, VIdent, $Y_{MU}$
  var {r}
  {r} = $E_{YMU}(D_{VIdent}(E_{SKMU}(\{r\})))$
  CS □ $Y_{MU}$,{r}
                  Role_of_Cloud_Server
  recv($DInfo_{MU}$)
  const $SK_{CS}$, VIdent, $Y_{MU}$
  var {r}
  {r} = $D_{SKCS}(E_{VIdent}(D_{YMU}(\{r\})))$
  #database □ pointer { store ({r})}
```

After executing these steps, a predefined range ofdynamic initialization vector (DIV) is stored in the database memory of the Cloud Server for further reference.

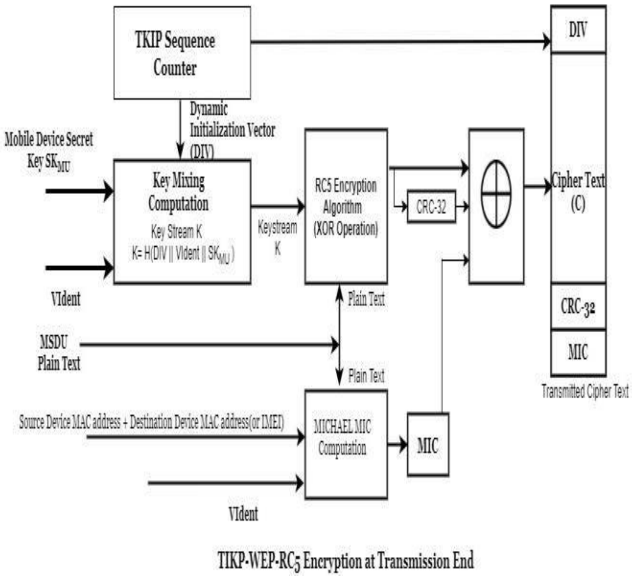
TKIP Sequence Counter
Dynamic Initialization Vector (DIV)
Mobile Device Secret Key SK_{MU}
Key Mixing Computation
Key Stream K
K= H(DIV || VIdent || SK_{MU})
VIdent
Keystream K
RC5 Encryption Algorithm (XOR Operation)
CRC-32
Plain Text
MSDU Plain Text
Plain Text
Source Device MAC address + Destination Device MAC address(or IMEI)
MICHAEL MIC Computation
MIC
VIdent
DIV
Cipher Text (C)
CRC-32
MIC
Transmitted Cipher Text
TIKP-WEP-RC5 Encryption at Transmission End

Fig.5 TKIP-WEP-RC5 Encryption at Transmission End

ENCRYPTION OF THE PACKETS AT SENDING END

First, the TKIP's TSC will create that it a simple and symmetric encryption algorithm with modest memory requirements. The enlarged key array S is initialized with data taken from the keystream K so that it looks like the array S[0,1,.........(t-1)], where t = 2(r+1) random binary words chosen by K.

In our algorithm, the ciphertext and plaintext are both inputs to the encryption and decryption functions, and the rotation shift or the number of rounds is variable depending on the block of data.

```
Algorithm: Encryption of packets usingRC5
Input: Plaintext in two 32-bit variables A & B
       Number of rounds (R)
       Expanded key table S[]
Output: Ciphertext stored in variables A and B
Procedure :
        A = A + S[0]
        B = B + S[1]
     var I;
     for I = 1 to R
        A = ((A $\oplus$ B) <<< B) + S[2 * I ]
        B = ((B $\oplus$ A ) <<< A) + S[2 * (I+1)]
```

Fourth, once packets have been encrypted, the CRC-32 checksum will be produced using the bits of the ciphertext and appended to the end of the MPDU as a means of error detection. Finally, to make the MAC Protocol Data Unit (MPDU) header size compatible with the maximum transmission unit (MTU) of wireless protocols, the three variables CRC-32 checksum, ciphertext, and MIC code from MICHAEL MIC methods are seeded into a cryptographic hash function and compressed into packets. This is what the compressed and cryptographically hashed packets will look like.

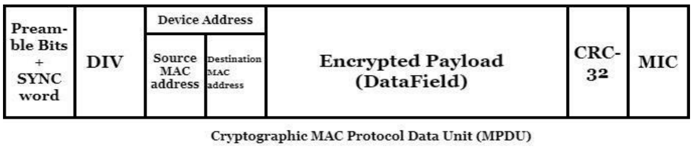
Pream-ble Bits + SYNC word
DIV
Device Address
Source MAC address
Destination MAC address
Encrypted Payload (DataField)
CRC-32
MIC
Cryptographic MAC Protocol Data Unit (MPDU)

Fig.6 Cryptographic Packets

The MIC and CRC-32 values were added at the rearposition MPDU and DIV at the front position of MPDU.

[14] The packets are processed at the MSPL physical layer before being sent through wireless networks. All of these complete packets are converted by the network adapter at the physical layer into their e
equivalent wireless signals, and then broadcast through cellular networks.

DECRYPTION OF PACKETS AT RECEIVING END

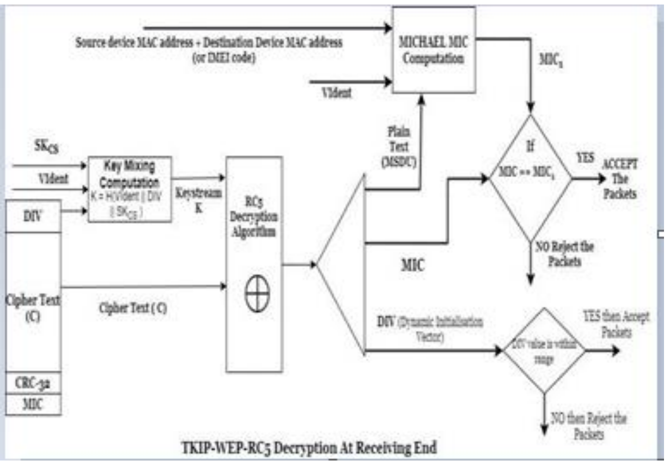
Source device MAC address + Destination Device MAC address (or IMEI code)
MICHAEL MIC Computation
MIC1
VIdent
Plain Text (MSDU)
If MIC == MIC1
YES
ACCEPT The Packets
NO Reject the Packets
SKCS
VIdent
Key Mixing Computation K = H(VIdent || DIV || SKCS)
Keystream K
RC5 Decryption Algorithm
DIV
Cipher Text (C)
Cipher Text (C)
CRC-32
MIC
MIC
DIV (Dynamic Initialisation Vector)
DIV value is within range
YES then Accept Packets
NO then Reject the Packets
TKIP-WEP-RC5 Decryption At Receiving End

FIG.7 TKIP-WEP-RC5 DECRYPTION AT RECEIVING END

Step 1) When an endpoint receives encrypted data, its network adapter performs the necessary encryption at the physical layer of the Media Separation Physical Layer (MSPL) before passing the resulting ciphertext frame onto the MAC layer for further processing. Deciphering, validating, and checking for errors in packets are all handled here.

Step 2) First, the DIV, the ciphertext, the seed, the Pair-wise temporary key VIdent, and the Cloud Server Secret key SKCS are all extracted and placed into the Key Mixing Computation function. Because the Diffie-Hellman Key Exchange formula dictates that the SKMU and SKCS must be the same, the keystream K is formed after the computation of the Key mixing function and is the same value at the sender or mobile devices.

Step 3) Then by using the RC5 Decryption algorithm this ciphertext is decrypted as follows

```
Algorithm: Decryption of packets usingRC5
Input: Ciphertext in two 32-bit variables A &B
        Number of rounds (R)
        Expanded key table S[]
Output: plaintext stored in variables A and B
Procedure :
          var I;
          for I = R to 1
      B = ((B – S[2*(I+1)]) >>> A ) ⊕ A
      A = ((A – S[2*I]) >>> B) ⊕ B
      B = B – S[1];
      A = A – S[0];
```

Then all the values in Cipher text are separated into four parts DIV, plaintext, CRC-32, and MIC. All these four values will be validated then packets are accepted.

Step 4) Validation of DIV.

```
Algorithm: Validation of DIV
Input: DIV from a ciphertext
        Range of DIV {r}
Output: DIV valid or invalid
Procedure :
        {r} □ #database (cloud server)
    var i;
    for i := o to Length({r} -1)
        If (DIV == {r}[i])
          Accept the packets
        else
          Reject the packet
      endif
  endfor
```

Step 5) If the Value of DIV exists into the range of DIV {r} then accepts the packets or otherwise rejects the packets.

```
Algorithm: Validation of DIV
Input: DIV from a ciphertext
        Range of DIV {r}
Output: DIV valid or invalid
Procedure :
        {r} □ #database (cloud server)
    var i;
    for i := o to Length({r} -1)
        If (DIV == {r}[i])
          Accept the packets
        else
          Reject the packet
      endif
  endfor
```

Step 6) Then validate the Message Integrity Code (MIC) with the newly generated Message Integrity Code (MIC1) using the MICHAEL MIC method with the input of source and destination device MACaddress and Paired wise transient key VIdent.

```
Algorithm: Validation of MIC
Input: MIC from a ciphertext
        MACMU, MACCS, VIdent, plaintext
Output: MIC valid or invalid
Procedure :
          var I, MIC1
MACMU,  MACCS, VIdent,  plaintext  □ MICHAEL MIC()
    MICHAEL MIC() □ MIC1
        If (MIC == MIC1)
          Accept the packets
        else
          Reject the packet
      endif
```

If the source device or mobile device's MAC$_{MU}$, MAC$_{CS}$, VIdent, and plaintext match the values entered, then the MICHAEL MIC technique will produce identical results in the cloud server.

Step 7) Lastly, the 32-bit CRC-32 error detection technique is used on the 32-bit plaintext. The basic idea of CRC-32 is the well-known Detection of Errors. In this case, we may say that the fault can be seen in plaintext when using CRC-32. The negative acknowledgment is sent to the sender and a request for retransmission of packets is made if an error is found in the plain text; otherwise, the packet is accepted.

Our suggested theory allows us to implement a secure technique for encrypting data in transit using mobile cloud services.

VI. CONCLUSION

Our current thesis article discusses four different approaches to the problem of securing data during wireless transmission. In our MCC Secure Protocol Layer (MSPL), we have merged several different methods of encryption and decoding. It is possible to fabricate or hack the MAC address, IMSI, user name, password, encryption, and decryption keys. To ensure the safety of transmitted data via wireless networks, our protocol MSPL combines the security benefits of several different cryptographic methods into a single, more efficient one. These algorithms include the RC5 technique, the TKIP methodology, the MICHAEL MIC method, and the CRC-32 error detection algorithm. To protect sensitive data, we never use a shared key or a public key for encryption or decryption within MCC. Our proposed MSPL protocol is superior to previous versions in several significant ways, including the inclusion of a cryptographic message integrity code (MIC), updated DIV sequencing values in every packet, and a key mixing function with a secret key generated using the Diffie-Hellman key exchange scheme.

There are several advantages of using packet cryptography for wireless networks:

For one, packet cryptography ensures that the data being transmitted is secure from prying eyes by encrypting its contents. Even if a hacker manages to intercept the packets, they will only be able to see the encrypted data without the key, making decryption extremely difficult. Packet cryptography uses cryptographic techniques like message authentication codes (MACs) or digital signatures to guarantee that no tampering has occurred with the sent data. Using these methods, a receiver can check to see if a packet has been altered after it was sent. Thirdly, authentication can be provided through packet cryptography, in addition to the previously mentioned confidentiality and integrity. This allows the receiver to confirm the sender's identity, adding confidence that the data packets come from a reliable source. Digitized certificates and other forms of authentication can do this. Finally, we can say that our protocol ensures safe communication over wireless networks by employing multifactor cryptography for data transmission. The following study will focus on the wireless signal protection technique in Mobile Cloud Computing.

VII. REFERENCES

[1] Abhishek Kumar Mishra, Aline Carneiro Viana, Nadjib Achir, Catuscia Palamidessi, "Public Wireless Packets Anonymously Hurt You", IEEE LCN 2021 (Doctoral-track - Promising ideas), Oct 2021, Edmonton / Virtual, Canada. ffhal-03298339v "

[2] Abir Mchergui, Rejab Hajlaoui, Tarek Moulahi, Abdulatif Alabdulatif, Pascal Lorenz, "Steam computing paradigm: Cross-layer solutions over cloud, fog, and edge computing ", 2023 The Authors. IET Wireless Sensor Systems published by John Wiley & Sons Ltd on behalf of The Institution of Engineering and Technology

[3] Ali Kamil Abed, Angesh Anupam, "Review of security issues in the Internet of Things and artificial intelligence-driven solutions", Security Privacy. 2023;6:e285. wileyonlinelibrary.com/journal/spy2, DOI: 10.1002/spy2.285

[4] Arikumar K.S.; Deepak Kumar A,; Sahaya Beni Prathiba; Tamilarasi K; , Rajalakshmi Shenbaga Moorthyd;Mohamed Iqbal M; "Enhancing the Security of WPA2/PSK Authentication Protocol in Wi-Fi Networks" Procedia Computer Science 215 (2022) 413-421, DOI : 10.1016/j.procs.2022.12

[5] Awan, S.; Javaid, N.; Ullah, S.; Khan, A.U.; Qamar, A.M.; Choi, J.-G. Blockchain-Based Secure Routing and Trust Management in Wireless Sensor Networks. Sensors 2022, 22, 411. https://doi.org/ 10.3390/s22020411

[6] Cao, Y., et al.: Anomaly detection with ensemble empirical mode decomposition and approximate entropy for quick user datagram protocol internet connection-based distributed Blockchain systems. IET Soft. 1–13 (2023). https://doi. org/10.1049/sfw2.12096

[7] CHIA-HUNG LIN, JIAN-XING WU, PI-YUN CHEN, CHIEN-MING LI, NENG-SHENG PAI, AND CHAO-LIN KUO "Symmetric Cryptography With a Chaotic Map and a Multilayer Machine Learning Network for Physiological Signal Infosecurity: Case Study in Electrocardiogram ", IEEE Access, Open Access Journal, DOI: 10.1109/ACCESS.2021.3057586

[8] Custura A, Jones T, Secchi R, Fairhurst G. Reducing the acknowledgment frequency in IETF QUIC. Int J Satell Commun Network. 2022;1-16. doi:10.1002/sat.1466

[9] Dener, M. SDA-RDOS: A New Secure Data Aggregation Protocol for Wireless Sensor Networks in IoT Resistant to DOS Attacks. Electronics 2022, 11, 4194. https://doi.org/10.3390/ electronics11244194

[10] ET da Silva, Costa ALD, JMH de Macedo. On the realization of VANET using named data networking: On improvement of VANET using NDN-based routing, caching, and security. Int J Commun Syst. 2022;35(18):e5348. doi:10.1002/dac.5348

[11] Goel, S., Basha, S.M., Carvalho dos Reis, M., de Albuquerque, V.H.C., Lathar, P., Alkhayyat, A.: Improved Malicious Node Detection method for detecting a bait in an extensive network for getting the maximum throughput. IET Commun. 1–8 (2022). https://doi.org/10.1049/cmu2.12498

[12] Hamid Mirvaziri, Rahim Hosseini "A Novel Method for Key Establishment Based on Symmetric Cryptography in Hierarchical Wireless Sensor Networks" Wireless Personal Communications, Springer Science+Business Media, LLC, part of Springer Nature 2020 https://doi.org/10.1007/s11277-020-07155-y

[13] Hongbo Zhang, "Application of Information Encryption Technology in Computer Network Communication Security", Hindawi Wireless Communications and Mobile Computing Volume 2022, Article ID 9354441, https://doi.org/10.1155/2022/9354441

[14] Jose David Vega S anchez; Luis Urquiza-Aguiar; Martha Cecilia Paredes Paredes; Diana Pamela Moya Osorio; "Survey on physical layer security for 5G wireless networks"; Annals of Telecommunications https://doi.org/10.1007/s12243-020-00799-8

[15] K. Suriyakrishnaan, D. Sridharan, "Reliable Packet Delivery in Wireless Body Area Networks", Springer Science+Business Media, LLC, part of Springer Nature 2018 Using TCDMA Algorithm for e-Health Monitoring System",Wireless Personal Communications https://doi.org/10.1007/s11277-018-5998-5

[16] Khandaker Foysal Haque, Ahmed Abdelgawad, and Kumar Yelamarthi," Comprehensive Performance Analysis of Zigbee Communication: An Experimental Approach with XBee S2C Module", Sensors 2022, 22, 3245. https:// doi.org/10.3390/s22093245

[17] Selvaraj, P.; Burugari, V.K.; Gopikrishnan, S.; Baza, M.; Srivastava, G.

An Enhanced and Secure Trust-Aware Improved GSO for Encrypted Data Sharing in the Internet of Things. Appl. Sci. 2022, 13, 831. https://doi.org/10.3390/ app13020831

[18] Sullivan S. et. Al. "5G Security challenges and solutions: A Review by OSI Layers" (2021) Electrical Engineering & Computer Science Faculty Publications. 490 https://engagedscholarship.csuohio.edu/enece_facpub/495 DOI: 10.1109/ACCESS.2021.3105396

[19] Tropea, M.; Spina, M.G.; De Rango, F.; Gentile, A.F. Security in Wireless Sensor Networks: A Cryptography Performance Analysis at MAC Layer. Future Internet 2022, 14, 145. https://doi.org/10.3390/ fi14050145

[20] Yichao Zhao, and Wenjun Ouyang; "Wireless Communication Network Security System Based on Big Data Information Transmission Technology "; Hindawi Wireless Communications and Mobile Computing Volume 2022, Article ID 1066331, 6 pages https://doi.org/10.1155/2022/1066331

[21] BOWEN ZHOU and RAJKUMAR BUYYA "Augmentation Techniques for Mobile Cloud Computing: A Taxonomy,Survey, and Future Directions" , The University of Melbourne,Australia, *ACM Comput. Surv.* 51, 1, Article 13 (January 2018), 38 pages. https://doi.org/10.1145/3152397

[22] Pham Phuoc Hung, Mohammad Aazam, Tien-Dung Nguyen and Eui-Nam Huh; "A Novel Approach for optimal Multimedia Data Distribution in MobileCloud Computing. Published in: Hindawi Publishing Corporation, Advances in Multimedia, Volume 2014, Article ID 137296. http://dx.doi.org/10.1155/2014/137296

[23] Wikipedia

[24] Dinesh Goyal, S. Balamurugan, Sheng-Lung Peng and O.P.Verma;" Security Protocol for Cloud Based Communication" *Publisher :* (eds.) Design and Analysis of Security Protocol for Communication, (247–254) © 2020 ScrivenerPublishing LLC

[25] Ajay D M1 & Umamaheswari; "Packet Encryption for Securing Real-Time Mobile Cloud Applications" Published: Mobile Networks and Applications (2019)24:1249–1254 https://doi.org/10.1007/s11036-019-01263-1

[26] Peilin Zhang, Xiaoyuan Ma, Oliver Theel, and Jianming Wei;"Packet-in-Packet: Concatenation with Concurrent Transmission for Data Collection in Low-PowerWireless Sensor Networks" *Published* 2018 IEEE 24th International Conference on Parallel andDistributed Systems (ICPADS) 978-1-5386-7308-9/18/$31.00 ©2018 IEEE DOI 10.1109/ICPADS.2018.00117 DOI 10.1109/AINA.2018.00065