Available online www.ejaet.com

European Journal of Advances in Engineering and Technology, 2023, 10(2):84-95

Research Article ISSN: 2394 - 658X

PROXY ARP IMPLEMENTATION-BASED SECURE DATA TRANSMISSION IN EVPN VXLAN UNDERLAY NETWORK USING FM2A2C TECHNIQUES

Amaresan Venkatesan

v.amaresan@gmail.com

ABSTRACT

Inter-subnet communication among data is enhanced by the Proxy Address Resolution Protocol (Proxy-ARP) implementation in Ethernet Virtual Private Network (EVPN) Virtual Extensible Local Area Network (VXLAN) underlay networks. Nevertheless, none of the prevailing works utilized Proxy ARP as a primary tool in EVPN VXLAN underlay networks, which resulted in unnecessary ARP broadcasts and inefficient bandwidth usage. Hence, in the proposed work, Feistel Message Autokey Authentication Cipher Code (FM2A2C) and Renyi Gibbs Entropy-based Advanced Polygram Hill Encryption Standard (RGE-APHES) techniques with Proxy ARP implementation for secured Data Transmission (DT) in EVPN VXLAN underlay network are utilized. Here, ARP requests are sent to the Proxy ARP phase by the sourced devices from the VXLAN network. A table of Message Authentication Code (MAC) and Internet Protocol (IP) addresses is maintained in the Proxy ARP phase grounded on the EVPN control plane. Later, ARP requests are verified by FM2A2C; in addition, the data from the verified MAC is then encrypted and transmitted via authenticated tunnels. Lastly, the destination is reached by the data utilizing an optimal routing path, which is chosen within 1366ms by employing the Gibbs Guided Coati Entropy Optimization Algorithm (2GCEOA). Therefore, the proposed work performed better than the conventional methodologies.

Keywords: Proxy ARP, EVPN VXLAN underlay network, Advanced Encryption Standard (AES), Quantum Cryptography (QC), Hadamard Gate Normalized-Quantum Cryptography (HGN-QC), Coati Optimization Algorithm (COA), Secured Data Transmission (SDT).

INTRODUCTION

Local Area Network (LAN) connects devices within a limited area, namely a building (George & George, 2021), whereas the Virtual Private Network (VPN) extends a private network across a public network for rendering secure remote access (Gaur et al., 2021). Therefore, scalable and efficient data center connectivity is enabled by the concatenation of EVPN VXLAN underlay networks through ARP requests. ARP requests map IP addresses to MAC addresses (Scazzariello et al., 2020, Pradhan & Mathew, 2020). In the meantime, Proxy ARP responds to ARP requests on behalf of other devices, thereby rendering seamless DT in EVPN VXLAN underlay networks (Sun et al., 2020).

Internet Protocol Security (IPsec), MAC security, and Border Gateway Protocol (BGP) are included in the prevailing techniques for secured DT in EVPN VXLAN underlay networks (Sun et al., 2019, Karamichailidis et al., 2020). For secured routing, these techniques encrypt data betwixt endpoints (Kellermann et al., 2021). Nevertheless, the prevailing techniques had limitations like elevated latency, delays, and diminished throughput (Zhu et al., 2022). Moreover, these techniques failed to handle ARP requests, causing inefficient bandwidth usage (Giatsios et al., 2019). Hence, this work leveraged various techniques for securing DT in the EVPN VXLAN underlay network to overcome such issues.

Problem Statement

The limitations in prevailing works are detailed below,

- None of the works implemented Proxy ARP in the EVPN VXLAN underlay network, resulting in unnecessary ARP broadcast and inefficient bandwidth use.

- Non-verification of MAC addresses in (Li et al., 2019) allowed malicious data to reach the destination, thereby degrading the entire performance.
- The data was transmitted via non-secured tunnels in many works, leading to security vulnerabilities.
- Unencrypted data transmission via a suboptimal path (Morsy & Nashat, 2022) resulted in delays and potential threats.

The proposed work's objectives are described further,

- The proposed work implemented Proxy ARP in the EVPN VXLAN underlay network for handling ARP requests effectively.
- MAC addresses are verified using the FM2A2C technique for effective DT.
- To avoid security vulnerabilities, tunnel authentication is done utilizing the HGN-QC technique.
- Data is encrypted and the optimal path for DT is chosen utilizing the RGE-APHES and 2GCEOA techniques for secured and faster DT.

The remaining paper is structured as: section 2 discusses the related works, section 3 describes the proposed methodology, section 4 presents the results and discussion, and finally, section 5 concludes the proposed work with future development.

LITERATURE SURVEY

(Li et al., 2019) established a BGP-centered IP VPN solution for DT in Software Defined (SD)-cloud. Extensible Messaging and Presence Protocol based South-Bound Interface for the Open Networking Operating System were used in this framework. In addition, the DT process is enhanced by the usage of Mininet and Open vSwitch techniques. Nevertheless, owing to heavy network loads, this framework had limited scalability.

(Morsy & Nashat, 2022) introduced the D-ARP scheme to detect and prevent ARP spoofing for effective DT. Here, the ARP packets signed with a key were sent betwixt ARP requests and replies for SDT. However, the entire work process was degraded by the ineffectiveness of handling ARP requests.

(Scazzariello et al., 2021) accomplished an open-source, scalable, and distributed architecture named VXLAN and EVPN BGP protocols for effective DT. Here, Docker containers and Kubernetes were utilized for SDT. However, malicious data were allowed by the non-verification of MAC addresses in this framework to reach the destination, thereby hindering the total performance.

(Subratie et al., 2023) presented a bounded flood, a technique for creating EVPN across edge and cloud resources. This model combined tunnels with SDN and utilized peer-to-peer (P2P) overlay, Symphony, and Kademlia for resilient routing. Nevertheless, this work had reduced accuracy in DT owing to P2P vulnerabilities and delays.

(Zaballa et al., 2021) amalgamated the In-band Network Telemetry (INT) protocol with programmable switches in a hybrid SDN network for enhancing DT monitoring. This framework configured data plane applications to interact with legacy Multiprotocol Label Switching devices and deployed INT headers for detailed traffic analysis. However, the DT monitoring efficiency was reduced by the high error rate in this system.

PROPOSED METHODOLOGY FOR SECURED DATA TRANSMISSION USING PROXY ARP IN EVPN VXLAN UNDERLAY NETWORK

Figure 1 depicts the structural diagram of the proposed work utilizing FM2A2C and RGE-APHES techniques with Proxy ARP implementation,

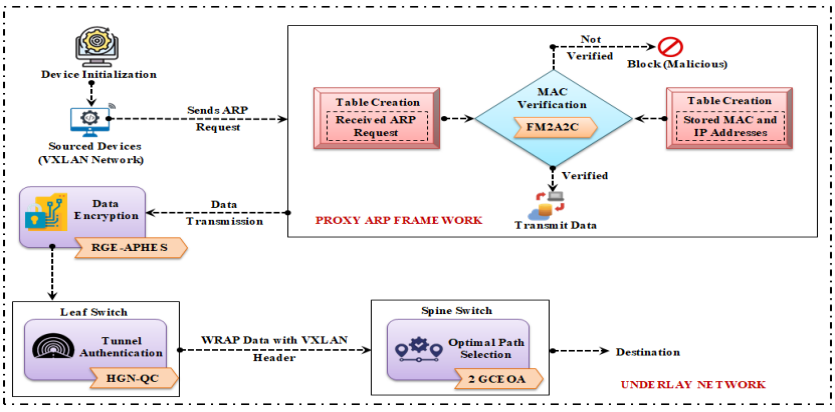


Figure 1: Structural Diagram of the Proposed Work

Device Initialization

The proposed work begins with device initialization in the VXLAN network, and the initialized devices (V^x) are signified as,

$$V^x = V^1, V^2, \ldots\ldots, V^{x''} \ \ where\ x = 1\ to\ x'' \quad (1)$$

Here, the total number of (V^x) is represented as (x'').

Firstly, the sourced devices (S^d) from the VXLAN network send ARP Requests to the Proxy ARP Framework for MAC verification, which is explained in further sections.

Proxy-ARP Framework

In this, for effective DT, the MAC addresses from (S^d) are verified with stored MAC addresses (MAC and IP addresses of devices stored during device initialization).

Table Creation

The MAC and IP addresses of initialized devices are stored during (V^x) centered on the EVPN control plane. Here, a table (T^1) is created for MAC and IP address storage as,

$$T^1 = V^x(MAC \& IP) \quad (2)$$

Where, the MAC and IP addresses of initialized devices are represented as $(MAC\ \&\ IP)$.

In the meantime, (S^d) sends ARP requests (A^r) to the Proxy-ARP framework. It is stored in table format as (T^2) to handle requests for effective bandwidth usage. This is equated as,

$$T^2 = S^d(A^r) \quad (3)$$

Here, the table that stores (A^r) of (S^d) is represented as (T^2).

MAC Verification

To ensure SDT, the MAC addresses for (V^x) are then verified from both the tables (T^1, T^2) by utilizing MAC protocol. However, the verification process might be hindered by the usage of weak keys in MAC. Thus, Feistel Autokey Cipher (FAC) is introduced in MAC, which generates dynamic keys grounded on message content to make hackers harder to guess or attack. Figure 2 ilustrates the structural diagram of FM2A2C,

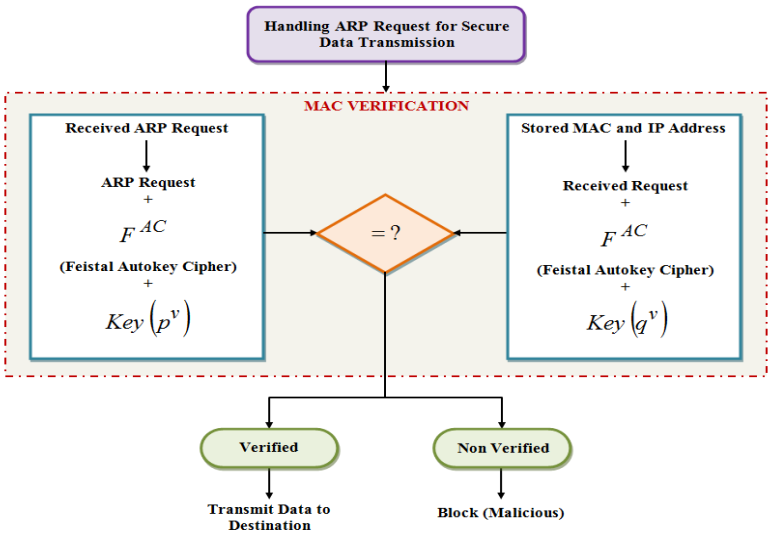


Figure 2: Handling ARP Request using FM2A2C

i. Primarily, the MAC addresses to be verified (T^1, T^2) are signified as (M^{AC}), which is split into 2 halves as equated below,

$$M^{AC} = L^H, R^H \quad (4)$$

Here, the left and right halves are represented as (L^H, R^H).

ii. Then, to enhance the verification process, the dynamic keys $\left(d^{keys}\right)$ are generated for each round (iteration) utilizing the FAC technique and hash functions $\left(h\right)$ as,

$$d^{keys} = h\left[L^H \oplus \lambda\left(R^H, \varphi\right)\right] \tag{5}$$

Here, the initial key is notated as $\left(\varphi\right)$, the cipher or non-linear function based on the FAC technique is symbolized as $\left(\lambda\right)$, and the Exclusive-OR operation is represented as $\left(\oplus\right)$.

iii. After iteration, to form a ciphertext $\left(S^{mac}\right)$, $\left(L^H, R^H\right)$ are concatenated utilizing private and public keys $\left(p^v, q^v\right)$ as,

$$S^{mac} = p^v\left(L^H, R^H\right) \times q^v\left(L^H, R^H\right) \tag{6}$$

iv. Now, the MAC addresses from $\left(T^1, T^2\right)$ are effectively verified $\left(\Psi\right)$ using the below conditions,

$$\Psi = \begin{cases} if\ \ M^{T^1} \in M^{T^2}\ \ then\ \Re \\ if\ \ M^{T^1} \notin M^{T^2}\ \ then\ \tilde{\Re} \end{cases} \tag{7}$$

Here, the condition states that if the MAC addresses $\left(M^{T^1}, M^{T^2}\right)$ from $\left(T^1\right)$are verified $\left(\in\right)$ with $\left(T^2\right)$, then the data is transferred to reach the destination $\left(\Re\right)$; or else, the data is deemed as malicious $\left(\tilde{\Re}\right)$ and is blocked for further transmission. The data from verified MAC is therefore notated as $\left(\Im\right)$.

Pseudo code of FM2A2C

Input: MAC addresses from $\left(T^1, T^2\right)$

Output: Verified MAC, $\left(\Im\right)$

Initialize iterations $\left(k, k^{\max}\right)$

While $\left(k < k^{\max}\right)$

Initialize $\left(T^1, T^2\right)$

Split $M^{AC} = L^H, R^H$,

Generate $\left(d^{keys}\right)$ using FAC technique,

$$d^{keys} = h\left[L^H \oplus \lambda\left(R^H, \varphi\right)\right]$$

Concatenate $\left(L^H, R^H\right)$ using $\left(p^v, q^v\right)$

For MAC verification

If $\left(M^{T^1} \in M^{T^2}\right)$,

MAC is verified

Else

Not-verified

End if

End for

End while

Return $\rightarrow \left(\Im\right)$

End

After that, $\left(\Im\right)$ is encrypted utilizing the AES technique as detailed in the section below.

Data Encryption

AES is employed for encrypting data by resisting differential and linear cryptanalysis attacks. Nevertheless, AES's effectiveness relies on proper key management; therefore, compromised keys can cause data breaches. Thus, Renyi Gibbs Entropy (RGE) and Polygram Hill (PH) cipher that enhance AES key generation by ensuring higher entropy and randomness for SDT are used. The RGE-APHES's algorithmic steps are detailed below,

♣ Primarily, by using RGE and PH technique, the key $\left(\Omega\right)$ is generated as,

$$\Omega = \frac{1}{1+\alpha}\log\sum\rho(\Im)^{\alpha}\cdot\mathrm{mod}(\Im) \quad (8)$$

Here, the logarithmic and modulus functions are represented as $(\log and \mathrm{mod})$, the probability of $(\Im)$ is symbolized as (ρ), and the order of entropy is signified as (α). Later, by utilizing the base key (χ) of AES and $(\alpha),(a,b)$ is generated hierarchically for encrypting $(\Im)$ as,

$$a = \chi \oplus \alpha \quad (9)$$

$$b = a \oplus \alpha \quad (10)$$

- Later, for each key, the matrix (a,b) is created, and it is shifted in row $(\tilde{r})$ and column $(\tilde{c})$ wise manner as illustrated below,

$$S^{out} = (\tilde{r}[a,b]\cdot m)+(\tilde{c}[a,b]\cdot n) \quad (11)$$

Here, the shifted output is proffered as (S^{out}) and the shifting factors are signified as (m,n). Grounded on this, the data is encrypted as,

$$\Phi = (S^{out}\times a,b)+\Im \quad (12)$$

Here, the encrypted data is represented as (Φ).

Pseudo-code of RGE-APHES

Input: MAC verified data, $(\Im)$

Output: Encrypted data, (Φ)

Begin

Initialize $(\Im)$

For data encryption,

Evaluate $\Omega = \frac{1}{1+\alpha}\log\sum\rho(\Im)^{\alpha}\cdot\mathrm{mod}(\Im)$

Generate (a,b)

Shift rows and columns

$S^{out} = (\tilde{r}[a,b]\cdot m)+(\tilde{c}[a,b]\cdot n)$

Encrypt data,

$\Phi = (S^{out}\times a,b)+\Im$

End for

Return (Φ)

End

Later, by using leaf and spine switches, (Φ) is transmitted via tunnels of VXLAN for secured and faster DT.

Tunnel Authentication

The tunnels are authenticated using QC in the leaf switch. QC ensures unconditional security by leveraging quantum mechanics. Since inaccurate generation may cause inefficiencies and high resource consumption, the accurate generation of quantum gates is crucial. Thus, the Hadamard Gate (HG) is employed in QC. HG creates superposition states by transforming qubits into equal probability for accurate quantum gate operations.

- Initially, by using HGN, a Quantum Key $(\tilde{q}^{key})$ is generated. Here, $(\tilde{q}^{key})$ analyzes (p^{v},q^{v}) to create a unique key, which is rendered as,

$$\tilde{q}^{key} = \frac{1}{\sqrt{2}}(p^{v}+q^{v}) \quad (13)$$

$$p^{v} = (q^{v}\times\wp) \quad (14)$$

Here, the coefficient (based on the Proxy ARP router) of (q^{v}) is represented as $(\wp)$.

- Later, by using $(\tilde{q}^{key})$ and $(\wp)$, the tunnels are authenticated (A^{tunnel}) as,

$$A^{tunnel} = \tilde{q}^{key}\times\delta(\wp) \quad (15)$$

Here, the polarization factor of $(\wp)$ for an effective authentication process is symbolized as (δ).

→ Lastly, by employing $(\tilde{q}^{key})$, the eavesdropping (E^D) is checked for SDT concerning (A^{tunnel}) as,

$$E^D = \begin{cases} \mathrm{E} & when \quad \left[\tilde{q}^{key}\left(A^{tunnel}\right)=1\right] \\ \mathrm{E}^* & when \quad \left[\tilde{q}^{key}\left(A^{tunnel}\right)=0\right] \end{cases} \tag{16}$$

When the value is 1, (E) represents the absence of eavesdropping, and when the value is 0, (E^*) represents the presence of eavesdropping. Therefore, the data with no eavesdropping is articulated as,

$$A^{tunnel}(\mathrm{E}) \rightarrow \left[A^{tunnel} * \tilde{q}^{key}\left(E^D\right)=1\right] \tag{17}$$

Therefore, in order to securely reach the final destination, (Φ) is transmitted via authenticated tunnels (A^{tunnel}). In the course of transmission, the data is wrapped (D^{wrap}) with a VXLAN header, which has additional information (I^{add}) about the destination VTEP's IP address. It is equated as,

$$D^{wrap} = \omega\left(\Phi \times I^{add}\right) \tag{18}$$

Here, the process of wrapping (Φ) is symbolized as (ω).

Spine Switch-Optimal Path Selection

The optimal path for DT is selected in the spine switch using COA by mimicking coati foraging behavior. Nevertheless, challenges in balancing exploration and exploitation are faced by COA, causing premature or slow convergence. Thus, the Gibbs Entropy-based Guided Random Walk (GE-GRW) technique is introduced to dynamically guide and adjust the search behavior for balancing the exploration-exploitation phase.

Population Initialization

❋ Primarily, the (l) numbers of paths (P^{DT}) for DT are deemed as coatis, and the population initialization of coatis (C^{ε}) with (z) number of dimensions are given as,

$$C^{\varepsilon} = \begin{bmatrix} C^{\varepsilon}_{(1,1)} & \cdots & C^{\varepsilon}_{(1,f)} & \cdots & C^{\varepsilon}_{(1,z)} \\ \vdots & \ddots & \vdots & \iddots & \vdots \\ C^{\varepsilon}_{(z,1)} & \cdots & C^{\varepsilon}_{(y,f)} & \cdots & C^{\varepsilon}_{(y,z)} \\ \vdots & \iddots & \vdots & \ddots & \vdots \\ C^{\varepsilon}_{(l,1)} & \cdots & C^{\varepsilon}_{(l,f)} & \cdots & C^{\varepsilon}_{(l,z)} \end{bmatrix} \tag{19}$$

Here, the value of the (y^{th}) variable for the (f^{th}) coati position is notated as $(C^{\varepsilon}_{(y,f)})$.

Fitness Calculation

Grounded on minimum duration (M^{dur}), the fitness value (F^{VAL}) for Optimal Path Selection (OPS) is computed as,

$$F^{VAL} = \min\left(M^{dur}\right) \tag{20}$$

Two strategies, namely exploration and exploitation are used by the coatis for OPS.

Exploration

By using GE-GRW, the exploration phase broadly guides and navigates the path for faster DT as shown below,

$$C^{\varepsilon+1} = -\Xi \sum \rho \times \ln\left(C^{\varepsilon} + \tilde{\alpha} \cdot G^{id} \cdot \left(C^{\varepsilon}_{best} - C^{\varepsilon}\right) + \tilde{\beta} \cdot R^{id}\right) \tag{21}$$

Where, the updated positions are represented as $(C^{\varepsilon+1})$, the scaling factors are notated as $(\tilde{\alpha}, \tilde{\beta})$, the guided direction and random component are signified as $(G^{id}\ and\ R^{id})$, the candidate best solution are symbolized as (C^{ε}_{best}), the Boltzmann constant based on GE is proffered as (Ξ), and the logarithmic functions and probability of search space are represented as $(\ln\ and\ \rho)$.

Exploitation

The path selection is refined in the exploitation phase by using GE-GBW as,

$$C^{\varepsilon^*} = \Xi \times \left(C^{\varepsilon+1} + \gamma \cdot G^{id} \cdot \left(C^{\varepsilon+1}_{best} - C^{\varepsilon+1}\right) + \eta \cdot R^{id}\right) \tag{22}$$

Where, the updated position after $(\varepsilon+1)$ iterations is notated as $\left(C^{\varepsilon^*}\right)$ and the smaller scaling factors for fine-tuning are symbolized as (γ,η). The optimal path is then chosen centered on the below conditions,

$$\breve{C}^{OP}=\begin{cases}C^{\varepsilon^*} & \forall\left(C^{\varepsilon^*}\le C^{\varepsilon}\right)\\ C^{\varepsilon} & else\end{cases} \tag{23}$$

Thus, the optimal path $\left(\breve{C}^{OP}\right)$ is chosen by updating the coati's position. Lastly, the encrypted data reaches its destination in minimum duration with the help of $\left(\breve{C}^{OP}\right)$. Thus, the proposed work's performance assessment is further described.

RESULTS AND DISCUSSION

The performance analysis is done to showcase the proposed model's reliability. The proposed work is implemented on the PYTHON platform.

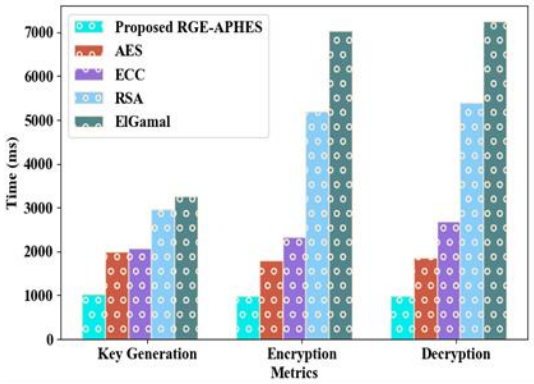
Performance validation

The proposed mechanism's performance is assessed by comparing it with conventional frameworks.

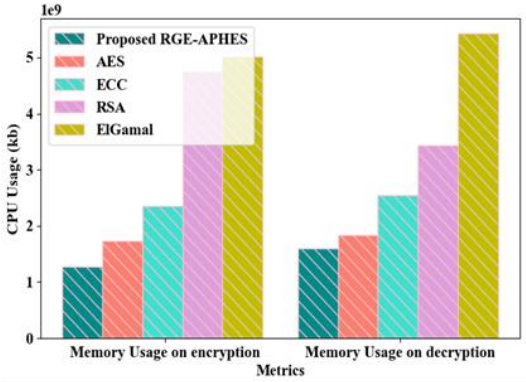
Table 1: MAC verification time analysis

MAC verification	MAC Verification Time (ms)
Proposed FM2A2C	1024
MAC	2115
HMAC	2746
KMAC	3362
CMAC	3781

The presence of FAC aids in upgrading the security level. In Table 1, the MAC verification time of the proposed FM2A2C and existing algorithms like MAC, Hash Message Authentication Code (HMAC), Keccak Message Authentication Code (KMAC), and Cipher-based Message Authentication Code (CMAC) are compared. A MAC verification time of 1024ms was obtained by the proposed FM2A2C, while the traditional algorithms achieved an average MAC verification time of 3001ms. Thus, the proposed scheme had low time complexity in MAC verification.



(a)



(b)

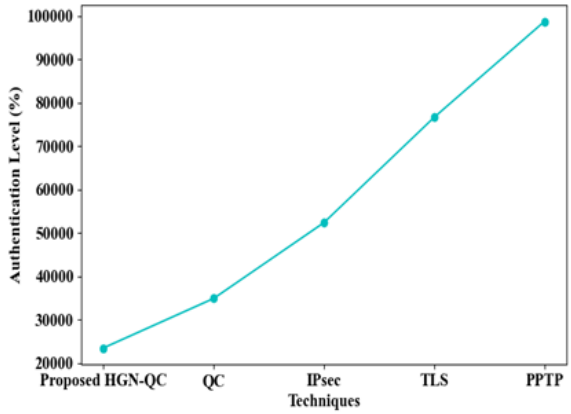
Figure 3: *Performance analysis of the proposed RGE-APHES regarding (a) ET, DT, and KGT and (b) memory usage on ET and memory usage on DT*

Owing to the RGE, the proposed work had higher significance in data security. The performance evaluation of the proposed RGE-APHES and traditional algorithms like AES, Elliptic Curve Cryptography (ECC), Rivest-Shamir-Adleman (RSA), and Elgammal is exhibited in Figure 3. The proposed RGE-APHES acquired Encryption Time (ET) of 985ms, Decryption Time (DT) of 994ms, Key Generation Time (KGT) of 1021ms, memory usage on ET of 1268925873kB, and memory usage on DT of 1599723645kB. Yet, due to improper key management, the conventional frameworks had limited results. Thus, the proposed work had higher reliability.

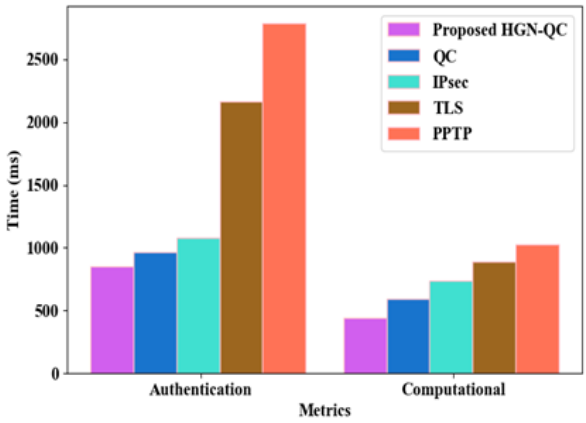
Table 2: Security level

Methods	Security Level (%)
Proposed RGE-APHES	98.99
AES	97.54
ECC	95.31
RSA	93.87
ElGamal	90.42

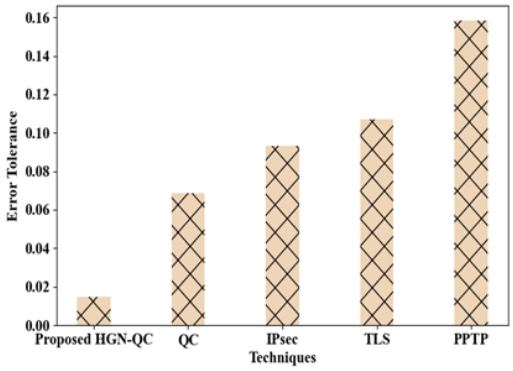
In Table 2, the Security Level (SL) of the proposed RGE-APHES and prevailing methods are analyzed. The proposed RGE-APHES achieved 98.99% SL, while the conventional ElGammal obtained 90.42% SL. Therefore, the proposed work was more effective compared to the existing algorithms.



(a)



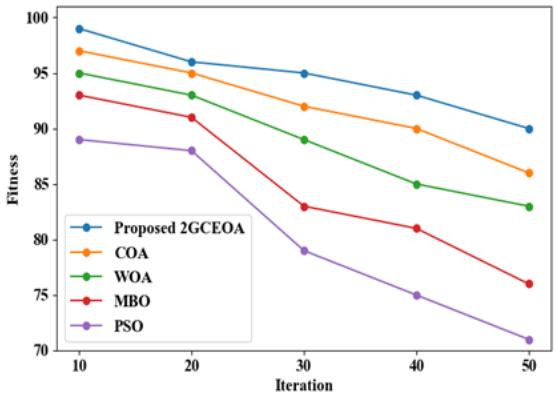
(b)



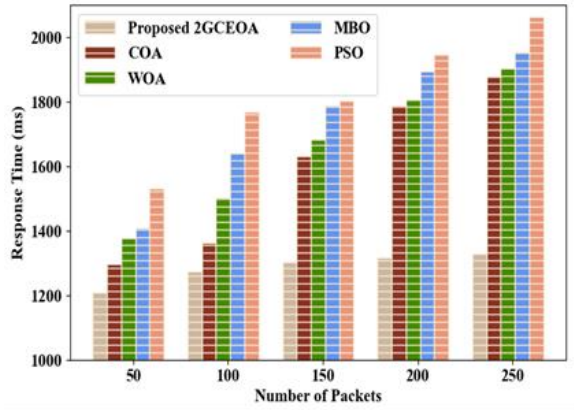
(c)

Figure 4: *Performance validation for tunnel authorization based on (a) AL, (b) AT and CT, and (c) ET*

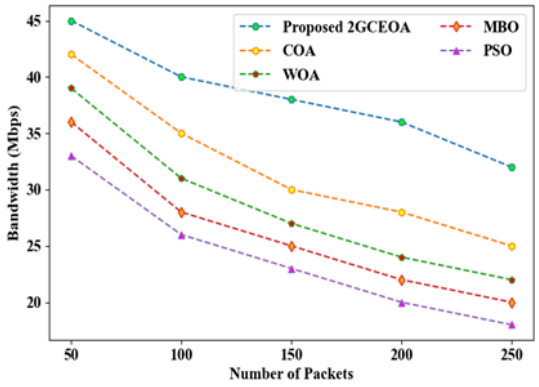
To improve the authorization process, the proposed work establishes the HG. Figure 4 evaluates the performance of the proposed HGN-QC by comparing it with traditional algorithms, such as Internet Protocol Security (IPsec), Transport Layer Security (TLS), and Point-to-Point Tunneling Protocol (PPTP). The proposed HGN-QC attained an Authentication Level (AL) of 98.85%, Authentication Time (AT) of 854ms, Computation Time (CT) of 438ms, and Error Tolerance (ET) of 0.0147. However, the prevailing mechanisms obtained a mean AL, AT, CT, and ET of 91.94%, 1746ms, 809ms, and 0.1067, respectively. Hence, as per the analysis outcomes, the proposed method had better outcomes in tunnel authorization.



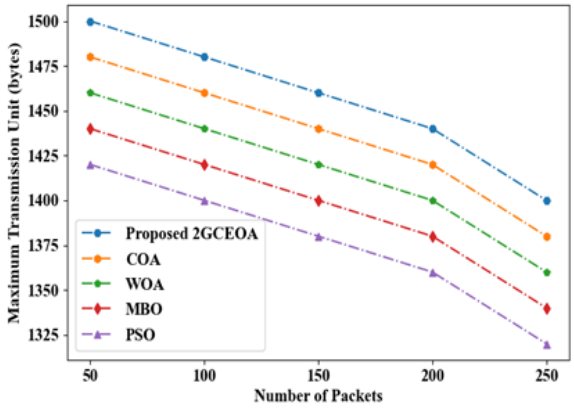
(a)



(b)



(c)



(d)

Figure 5: *Performance assessment for optimal path selection with respect to (a) fitness vs iteration, (b) response time, (c) bandwidth, and (d) MTU*

In Figure 5, the performance analysis of the proposed 2GCEOA and existing techniques like COA, Whale Optimization Algorithm (WOA), Monarch Butterfly Optimization (MBO), and Particle Swarm Optimization (PSO) are shown. Due to the presence of GB-GRW, the proposed work had superior outcomes. At the 10th and 50th iterations, the proposed 2GCEOA achieved a fitness value of 99 and 90, respectively. However, at the 10th iteration, the existing algorithms attained an average fitness value of 93.5. Similarly, the proposed 2GCEOA attained Response Time (RT), bandwidth, and Maximum Transmission Unit (MTU) of 1208ms, 45Mbps, and 1500bytes, respectively. However, the traditional COA obtained an average RT of 1404ms, bandwidth of 37.5Mbps, and MTU of 1450bytes. At last, it is concluded that the proposed approach had higher superiority when analogized to traditional algorithms.

Comparative analysis

The research methodology's comparative assessment is shown further,

Table 3: Comparative evaluation

Author's name	Aim	Techniques	Advantage	Drawback
Proposed model	Proxy ARP tool-powered data transmission in an EVPN VXLAN underlay network	FM2A2C	The proposed work had a high security level.	Nevertheless, this work only concentrated on data security.
(Munther et al., 2022)	Proxy ARP-based secure mechanism for Software Defined Networks (SDN)	Dynamic Host Configuration Protocol (DHCP)	This work had lesser memory usage.	However, it had a single point of failure issue owing to the fixed tree topology.
(Sadio et al., 2020)	SDN-based WIFI direct group formation using Proxy ARP	Group formation strategies	It had higher feasibility.	But, it had maximal network overhead.
(Bruschi et al., 2022)	Advance Arp-ON-based formal verification in Ethernet networks	ArpON algorithm	This framework had a high significance.	Yet, it failed to identify the optimal path.
(Tchendji et al., 2021)	Secure protocol-based ARP attack mitigation in SDN	Efficient Bayes-Based Security Protocol (E2BaSeP)	This model had superior efficiency.	However, it had a higher time complexity.
(Girdler & Vassilakis, 2021)	Defending mechanism for SDN against ARP attack and blacklisted MAC addresses	ARP-MAC and machine learning	This approach had a lower mitigation time.	Yet, it was ineffective due to the insufficient traffic features.

In Table 3, the comparative assessment of the proposed model and related approaches is shown. The MAC address is proficiently verified by the proposed FM2A2C, thus enhancing network security. Nevertheless, due to inappropriate key management and network functionalities, the existing models had poorer performance than the proposed work. Collectively, the proposed model outperformed the prevailing works.

CONCLUSION

A proxy ARP implementation-based secure data transmission in EVPN VXLAN underlay network using FM2A2C is proposed in this paper. The MAC verification and tunnel authorization were effectively performed by the proposed FM2A2C and HGN-QC, thus augmenting the security rate. Moreover, to elevate the system's consistency, the processes like data encryption and optimal route selection were done. Therefore, as per the experimental results, the proposed approach attained 98.99% SL, 438ms CT, and 1024ms MAC verification time. The proposed methodology had better outcomes for all the quality metrics. Thus, the proposed approach had lower complexity as well as better efficiency. Nevertheless, data security in the EVPN-VXLAN network was only focused on in this work.

Future scope: The efforts of this paper will be extended in the future by concentrating on ARP attack-type prediction for upgrading the network's trustworthiness.

REFERENCES

[1]. Bruschi, D., Di Pasquale, A., Ghilardi, S., Lanzi, A., & Pagani, E. (2022). A Formal Verification of ArpON - A Tool for Avoiding Man-in-the-Middle Attacks in Ethernet Networks. IEEE Transactions on Dependable and Secure Computing, 19(6), 4082–4098. https://doi.org/10.1109/TDSC.2021.3118448

[2]. Gaur, K., Kalla, A., Grover, J., Borhani, M., Gurtov, A., & Liyanage, M. (2021). A Survey of Virtual Private LAN Services (VPLS): Past, Present and Future. Computer Networks, 196, 1–25. https://doi.org/10.1016/j.comnet.2021.108245

[3]. George, A. S., & George, A. S. H. (2021). A Brief Overview of VXLAN EVPN. International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering, 9(7), 1–12. https://doi.org/10.5281/zenodo.7027361

[4]. Giatsios, Di., Choumas, K., Flegkas, P., Korakis, T., Cruelles, J. J. A., & Mur, D. C. (2019). Design and Evaluation of a Hierarchical SDN Control Plane for 5G Transport Networks. ICC 2019-2019 IEEE International Conference on Communications (ICC), 1–6. https://doi.org/10.1109/ICC.2019.8761263

[5]. Girdler, T., & Vassilakis, V. G. (2021). Implementing an intrusion detection and prevention system using Software-Defined Networking: Defending against ARP spoofing attacks and Blacklisted MAC Addresses. Computers and Electrical Engineering, 90, 1–12. https://doi.org/10.1016/j.compeleceng.2021.106990

[6]. Karamichailidis, P., Choumas, K., & Korakis, T. (2020). OpenFlow enabled Integrated Routing and Bridging over Ethernet Virtual Private Networks. NOMS 2020-2020 IEEE/IFIP Network Operations and Management Symposium, 1–5. https://doi.org/10.1109/NOMS47738.2020.9110339

[7]. Kellermann, T., Canellas, F., Gonzalez, R., & Camps-Mur, D. (2021). VL2-WIM: Flexible virtual layer 2 connectivity services in distributed 5G MANO domains. Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit), 413–418. https://doi.org/10.1109/EuCNC/6GSummit51104.2021.9482422

[8]. Li, Y., Osinski, T., & Dandoush, A. (2019). Enabling BGP/MPLS VPN as a connectivity and multi-tenants isolation technology for SD-Cloud Networks. Proceedings - 2019 International Conference on Future Internet of Things and Cloud, FiCloud 2019, 1–8. https://doi.org/10.1109/FiCloud.2019.00018

[9]. Morsy, S. M., & Nashat, D. (2022). D-ARP: An Efficient Scheme to Detect and Prevent ARP Spoofing. IEEE Access, 10, 49142–49153. https://doi.org/10.1109/ACCESS.2022.3172329

[10]. Munther, M. N., Hashim, F., Latiff, N. A. A., Alezabi, K. A., & Liew, J. T. (2022). Scalable and secure SDN based ethernet architecture by suppressing broadcast traffic. Egyptian Informatics Journal, 23(1), 113–126. https://doi.org/10.1016/j.eij.2021.08.001

[11]. Pradhan, A., & Mathew, R. (2020). Solutions to Vulnerabilities and Threats in Software Defined Networking (SDN). Procedia Computer Science, 171, 2581–2589. https://doi.org/10.1016/j.procs.2020.04.280

[12]. Sadio, O., Ngom, I., & Lishou, C. (2020). Controlling WiFi Direct Group Formation for Non-Critical Applications in C-V2X Network. IEEE Access, 8, 79947–79957. https://doi.org/10.1109/ACCESS.2020.2990671

[13]. Scazzariello, M., Ariemma, L., Battista, G. Di, & Patrignani, M. (2020). Megalos: A Scalable Architecture for the Virtualization of Network Scenarios. NOMS 2020-2020 IEEE/IFIP Network Operations and Management Symposium, 1–7. https://doi.org/10.1109/NOMS47738.2020.9110288

[14]. Scazzariello, M., Ariemma, L., Battista, G. Di, & Patrignani, M. (2021). Megalos: A scalable architecture for the virtualization of large network scenarios. Future Internet, 13(9), 1–17. https://doi.org/10.3390/fi13090227

[15]. Subratie, K., Aditya, S., & Figueiredo, R. J. (2023). EdgeVPN: Self-organizing layer-2 virtual edge networks. Future Generation Computer Systems, 140, 104–116. https://doi.org/10.1016/j.future.2022.10.007

[16]. Sun, Y., Esaki, H., & Ochiai, H. (2019). Detection and Classification of Network Events in LAN Using CNN. 4th International Conference on Information Technology (InCIT), 203–207. https://doi.org/10.1109/INCIT.2019.8911924

[17]. Sun, Y., Esaki, H., & Ochiai, H. (2020). Visual Analytics for Anomaly Classification in LAN Based on Deep Convolutional Neural Network. Joint 9th International Conference on Informatics, Electronics & Vision (ICIEV) and 2020 4th International Conference on Imaging, Vision & Pattern Recognition, 1–6. https://doi.org/10.1109/ICIEVicIVPR48672.2020.9306641

[18]. Tchendji, V. K., Mvah, F., Djamegni, C. T., & Yankam, Y. F. (2021). E2BaSeP: Efficient Bayes Based Security Protocol Against ARP Spoofing Attacks in SDN Architectures. Journal of Hardware and Systems Security, 5(1), 1–17. https://doi.org/10.1007/s41635-020-00105-x

[19]. Zaballa, E. O., Franco, D., Thomsen, S. E., Higuero, M., Wessing, H., & Berger, M. S. (2021). Towards monitoring hybrid next-generation software-defined and service provider MPLS networks. Computer Networks, 191, 1=30. https://doi.org/10.1016/j.comnet.2021.107960

[20]. Zhu, S., Lu, J., Lyu, B., Pan, T., Jia, C., Cheng, X., Kang, D., Lv, Y., Yang, F., Xue, X., Wang, Z., & Yang, J. (2022). Zoonet: A Proactive Telemetry System for Large-Scale Cloud Networks. Proceedings of the 18th International Conference on Emerging Networking EXperiments and Technologies, 321–336. https://doi.org/10.1145/3555050.3569116