mathematics

MDPI

Article

Improved Recurrent Neural Network Schema for Validating Digital Signatures in VANET

Arpit Jain [1], Jaspreet Singh [2,†], Sandeep Kumar [1,†], Ţurcanu Florin-Emilian [3,*], Mihaltan Traian Candin [4,*] and Premkumar Chithaluru [1,†]

[1] Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation (KLEF), Vaddeswaram 522302, Andhra Pradesh, India
[2] Department of Computer Science and Engineering, Chandigarh University, Gharuan, Mohali 140413, Punjab, India
[3] Department of Building Services, Faculty of Civil Engineering and Building Services, Gheorghe Asachi Technical University of Iasi, 700050 Jassy, Romania
[4] Faculty of Building Services Cluj-Napoca, Technical University of Cluj-Napoca, 400114 Cluj-Napoca, Romania
* Correspondence: florin-emilian.turcanu@academic.tuiasi.ro (Ţ.F.-E.); mihaltantraian83@gmail.com (M.T.C.)
† These authors contributed equally to this work.

Abstract: Vehicular ad hoc networks (VANETs) allow communication between stationary or moving vehicles with the assistance of wireless technology. Among various existing issues in smart VANETs, secure communication is the key challenge in VANETs with a 5G network. Smart vehicles must communicate with a broad range of advanced road systems including traffic control and smart payment systems. Many security mechanisms are used in VANETs to ensure safe transmission; one such mechanism is cryptographic digital signatures based on public key infrastructure (PKI). In this mechanism, secret private keys are used for digital signatures to validate the identity of the message along with the sender. However, the validation of the digital signatures in fast-moving vehicles is extremely difficult. Based on an improved perceptron model of an artificial neural network (ANN), this paper proposes an efficient technique for digital signature verification. Still, manual signatures are extensively used for authentication across the world. However, manual signatures are still not employed for security in automotive and mobile networks. The process of converting manual signatures to pseudo-digital-signatures was simulated using the improved Elman backpropagation (I-EBP) model. A digital signature was employed during network connection to authenticate the legitimacy of the sender's communications. Because it contained information about the vehicle on the road, there was scope for improvement in protecting the data from attackers. Compared to existing schemes, the proposed technique achieved significant gains in computational overhead, aggregate verification delay, and aggregate signature size.

Keywords: wireless technologies; security; pseudo-digital-signature; VANET; PKI; ANN; I-EBP

MSC: 68Q06

Citation: Jain, A.; Singh, J.; Kumar, S.; Florin-Emilian, Ţ.; Traian Candin, M.; Chithaluru, P. Improved Recurrent Neural Network Schema for Validating Digital Signatures in VANET. *Mathematics* **2022**, *10*, 3895. https://doi.org/10.3390/math10203895

Academic Editor: Ximeng Liu

Received: 6 September 2022
Accepted: 13 October 2022
Published: 20 October 2022

1. Introduction

A VANET contains moving vehicles with roadside units (RSUs) and these RSUs have small radios mounted on them for proper communication [1,2]. VANETs focus on communication without a centralized unit or controller. They provide communication vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I), and infrastructure-to-vehicle (I2V) [3,4]. The primary need for such a type of communication is that each node should be in each other transmission/communication range. The range of communication in most VANETs is 300 m to 1000 m. The data rate is between 6 Mbps to 27 Mbps. With the unexpected growth of smart vehicles, VANETs continuously provide new challenges to present-day researchers.

Due to gateways and wireless routers, wireless ad hoc networks (WANETs) are decentralized wireless networks that do not possess any preexisting architecture, such as wired networks. WANETs are further subdivided into two types: mobile ad hoc networks (MANETs) and VANETs. MANETs establish connections between different portable devices, whereas VANETs communicate with vehicles and RSUs as shown in Figure 1 [5,6]. VANETs are becoming popular because they provide V2V, V2I, and I2V communication. VANETs are brilliant solutions for accident prevention, traffic control, toll payment, transmitting relevant information for security updates, and weather monitoring notifications. There is no fixed infrastructure in such ad hoc networks, which results in security issues in VANETs [7]. The challenges include safeguarding drivers' personally identifiable information; banking information; the confidentiality of incoming messages; private information; and the safety of keys saved in intelligent transportation systems.

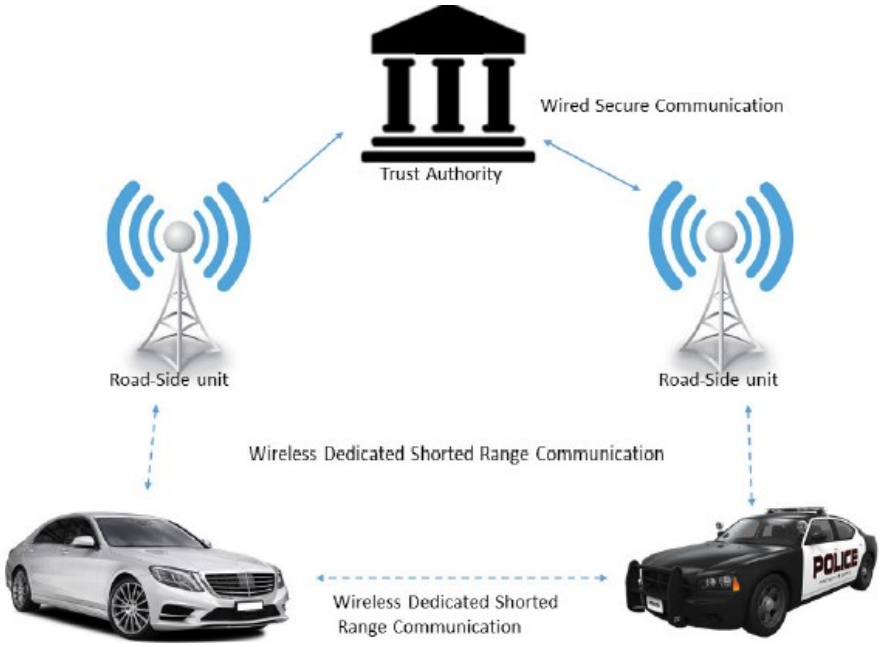


Figure 1. VANET typical structure.

In an ad hoc ecosystem, VANETs offer numerous key features to users, and private details such as users' location tracking, bank details, and so forth are associated with network systems. Any inappropriate behavior in the environment has a broad range of adverse consequences. When a vehicle wishes to transmit a message, it needs to join the ad hoc cluster and, for safety reasons, go through a succession of safety checks. Vehicular verification is one of these safety features, where the vehicle is confined to identity verification. There are numerous secure authentication methods available in VANETs. Some of them are biometric-based, such as fingerprint authentication and customer biometrics.

1.1. Layer Recurrent Neural Network (LRNN)

A layer-recurrent neural network is a kind of training model that is time-dependent, so it is known as a deep network across time. These networks form learning system graphs with the help of connections which are used for attempting to solve complex information learning problems as a result of this functionality. Because a recurrent neural network (RNN) is good for both text and numerical information, the authors converted fingerprint images into a matrix of a double data frame in the proposed model. RNNs have input and output flows, which distinguishes them from other biological systems [8,9]. The reversed flow of data chases forward the information flow to impact the learning experience, hence the term backpropagation over time [10]. VANETs can create major problems on roads such as traffic jams, road accidents, blockage, and other vulnerable problems [11,12]. Therefore,

VANETs must provide a security feature to their node for preserving its privacy. The following are the security issues of biometric protection techniques:

I **Authentication:** Authentication ensures that the message is generated by an authentic user, not by a spammer. One of the attacks that can be avoided by providing authentication is the Sybil attack [13,14].

II **Entity authentication:** It can be done at the receiver end. The receiver will be able to authenticate the identity of the sender and their activeness during communication or in the network [15,16].

III **Access control:** This helps determine that all nodes work according to the job or tasks assigned to them in the network [17].

IV **Privacy:** This is one of the important security requirements as the private information of the user must not be shared or leased to unauthorized parties [18–22].

1.2. Problem Statement

A VANET is an open wireless communication network that allows attackers to easily track, update, and modify transmission information. For instance, suspicious vehicles in VANETs may broadcast fraudulent information to obtain benefits or transmit inaccurate data to misdirect the decisions of the traffic management center (TMC). Digital signatures, as an efficient way of resolving these issues, could provide security-related features such as nonrepudiation, identity verification, and authenticity. To stop the illegal breach, responsive confidential details of drivers, such as their real identities and travel directions, should be secured. Privacy and confidentiality can be used to resolve this issue. When attempting to communicate with other vehicles or RSUs, vehicles can use pseudoidentities. However, if some vehicles deliberately interrupt traffic flow, the TMC ought to be capable of monitoring their actual identity. Thus, there is a need to provide security mechanisms that can secure the data and private information stored in the internal storage of vehicles along with the verification of the vehicle and driver's identity without taking much time and also reduce the computational overhead.

The main objectives of the paper are as follows,

I We propose an I-EBP for validating digital signatures in a VANET, which protects privacy. Pseudo-digital-signatures are used in the proposed scheme.

II The proposed scheme comes up with a comparison of multiple networks based on time and epochs showing that the network with the fewest neurons in the hidden layers takes less time to train and the network with the most neurons takes the fewest epochs to train to achieve the goal.

III The primary focus of the research work is that if an attacker wants to compromise the vehicle's security by forging the pseudo-digital-signature, it must establish a manual signature and then employ various network filters to convert the manual signature into the pseudo-digital-signature.

IV Compared with existing schemes, the proposed enhanced perceptron scheme has the advantage of maintaining a lower computation overhead, reducing the aggregate verification delay, and being effective in aggregate signature size.

The remainder of the paper is organized as follows. Section 2 enumerates the most relevant and significant works. Section 3 describes the proposed system methodology, followed by a security analysis of the proposed system in Section 4. Section 5 discusses the results and observations. Finally, Section 6 provides the conclusion and future scope of the manuscript.

2. Literature Review

This section discusses the overview of ANNs and related techniques for validating digital signatures, pattern recognition, and pattern classification.

2.1. Overview of ANN

An ANN is based on the neural network structure of a human brain [23]. They are not entirely the same but have some similarities with each other. An ANN is formed by artificial neurons which in turn have the same functionality as biological neurons [24]. Table 1 shows the differences and similarities between ANNs and biological neural networks (BNNs). Figure 2 shows the structure of biological neurons and the basic terminologies attached to them and Figure 3 shows the artificial neuron and its basic functions and parts.

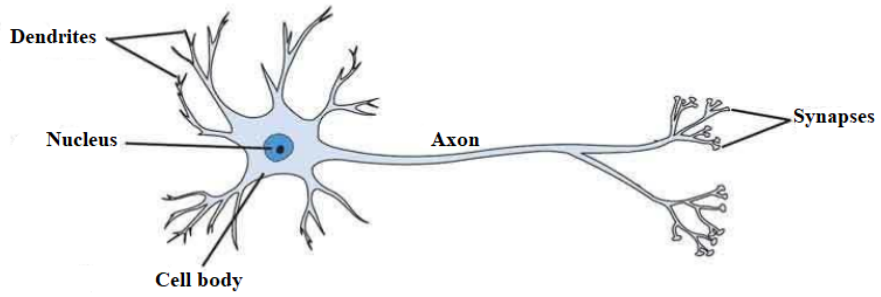


Figure 2. Biological neuron.

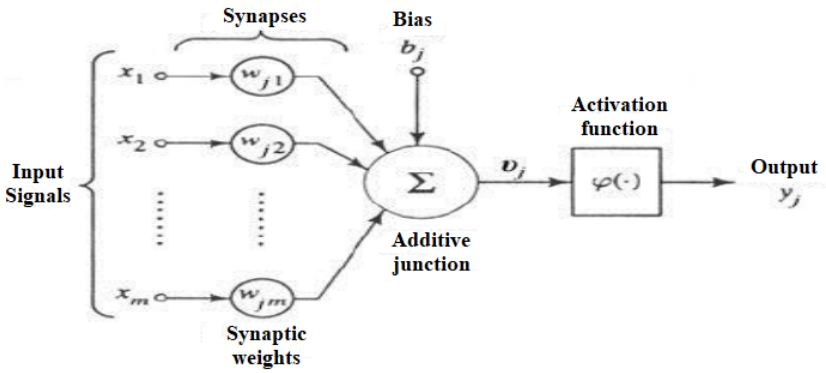


Figure 3. Artificial neuron.

Table 1. Comparison parameters of ANN Vs. BNN.

BNN	ANN
It works via serial processing. Processing of instructions and problem rules take place at one time.	It works via parallel processing, which means various processes work at the same time in parallel.
The functionality of these networks based on a rule-based approach like “if & else” rules.	The functionality depends on learning algorithms.
Dendrites	Weighted inputs
Cell body	Artificial neurons
Axon	Outputs

In ANNs, the information comes into the body through inputs that have some weights associated with them. An ANN is also called a weighted directed graph in which artificial neurons act as nodes. Each input is multiplied by the weight associated with it. Weights are the processed form of data that work as a strength to solve a particular problem in the neural network. The body of the artificial neuron sums the weighted inputs and if it is zero, then bias values are added to make it nonzero; then, it processes the sum with a transfer function. A linear or nonlinear activation function is set as the transfer function to limit the responses for arriving at the desired output point. In the end, the processed information is transferred to the output. The neural network is robust and has fault-tolerance property [25]. It can easily handle fuzzy, noisy, interrupted, and imprecise information.

Many security concerns were present in early wireless ad hoc networks. However, the uniqueness of moving objects communicating with each other in VANETs has a set of new outside attack challenges. Their work has a detailed description of the types of attackers and the type of attacks that these machines in the networks face on daily basis. It has been detailed that the approaches designed for the security of MANETs are no longer valid for VANETs due to high mobility constraints. The parameters of jittering, packet-loss latency, and throughput have interestingly become service parameters. The hierarchy of networks from the wireless networks to the mobile networks, thus to MANETs and VANET, has been presented. The driver–vehicle model, traffic-flow model communication and application models have been covered [24–26].

The works on cryptosystems implementing security by elliptic curve algorithms gave scientific suggestions to improve public-key cryptosystems. These authors started their work from the basic history of the public use of cryptosystems. They tracked the similarity of Rivest–Shamir–Adleman (RSA) and elliptic curves and the academic use of both algorithms. Further, the design of the elliptic curve using the digital signature algorithm (DSA) and encryption curve were the practical test-bed developments [27].

Various applications related to comfort and safety in VANETs were investigated, such as emergency vehicle warnings, warning for violating stop signs, violating traffic signal warnings, intersection collision warning, safety recall notices, just-in-time repair notifications, road condition warnings, etc. Then, in the latter application, Chien carried out various challenges existing in VANETs related to security and privacy and strengthened the VANETs environment by tackling all loopholes in VANETs [28].

Malicious, rational, active and passive, local, extended, and monitoring attacks were the security requirements of VANETs for the integrity, availability, privacy, traceability, revocability, and confidentiality issues. The damage inflicted by Global Positioning System (GPS) spoofing, hidden vehicles, illusion attacks, ID disclosures, and tunnel attacks are the type of attacks where security codes have been violated [29].

The storage in VANET-based clouds and beacons broadcast by different vehicles in VANETs have been collected and stored in a cloud architecture. The data stored in clouds are then used by other vehicles to know about the traffic conditions. An additional benefit is to the authorities and those implementing artificial intelligence in VANETs for route predictions. The cloud-based framework used by VANETs can help the administrators and security agencies in route-tracing mechanisms whereas, on the other hand, the scheme gives identity-less beacons for privacy reasons [30].

Elliptical curve cryptography (ECC) in mobile system security was investigated with other cryptography systems in terms of computation and keys. The authors concluded that ECC keys were the fastest, needed less time for verification, needed less computation, and hence were less costly than other cryptographic keys. Moreover, ECC produces keys from elliptical curves and certificate authorities are responsible for ECC [31].

In an ad hoc environment on the road, when vehicles are moving at a very high speed, there is a need for a strong security mechanism for a communication system. An approach for secure message communication in VANETs was based on the RSA cryptography algorithm. The main goal was to achieve a strong encryption mechanism to safeguard the VANET communication system from threats and attacks [32].

The backpropagation learning algorithm was introduced in 1970 to solve the problem of single-layer perceptrons, i.e., the XOR gate problem. This learning algorithm adjusts the weights of the network in such a way that inputs can be transformed into the desired set (target) of outputs after the learning process. This algorithm can be used in pattern recognition and pattern classification problems. In this learning algorithm, the network is first initialized by some random small-value weight between -1 and 1. Then, the output is calculated after training by applying input patterns. This output is compared with the target, and error values are calculated. Depending on these error values, the weights are adjusted, and then the network is trained again until minimal error values are obtained and the calculated output is close to the target values. A two-layer network with feedback

networking was used to detect and produce time-varying patterns. This algorithm differed from others in that the time delays of recurrent connections from previous time steps were saved and could be used in the current timestamp [33].

Different cryptographic algorithms have been used in various vehicular ad hoc networks to ensure secure communication between nodes. The authors began their work with the appeal of driverless vehicles and considered the information sent and received from roadside units, vehicles moving in the same direction and opposite direction, and outside attackers. All traditional cryptographic schemes' security and reliability were used in MANETs and VANETs to meet a vehicle's privacy and security requirements. Securing message communication was also a contribution of digital signatures and crypto methods based on attributes. Radio interfaces equipped with onboard units can collect private information about vehicles. Digital signatures and crypto methods differ based on hardware used in VANETs, i.e., event data recorders, tamper-proof devices (TPD), and infrastructure. Identity-based systems and attribute-based encryption schemes were used for cipher text policies applied to message authentication [34].

VANETs are an easy target for attackers, and such attacks can lead to network corruption. Many of these security concerns were present earlier in ad hoc wireless networks. However, the uniqueness of moving objects communicating with each other is a new outside attack challenge in VANETs. It was observed that the approaches designed for the security of MANETs were no longer valid for VANETs due to high mobility constraints [35].

VANETs are distributed network systems that have complex communication mechanisms. By studying various research articles, it was observed that VANETs were prone to unauthorized access, eavesdropping [36], hardware tempering [37], denial of service (DoS) [38], surveillance [39], replay the legitimate message, imposture, bogus information [40], etc. Many attacks can affect the confidentially, integrity, and availability of the vehicle and the data attached to them. Therefore, there is a great need to enhance the security of vehicles in a real-time environment. In the future, driverless cars will be on the road everywhere and the potential security mechanisms will protect vehicles and humans. Privacy and impersonality are also hazardous issues that need to be addressed. It is prudent to secure the identity of vehicles and their drivers from intruders who can take advantage of that and can create false identities using it.

2.2. Drawbacks Identified in the Literature

With intelligent transportation systems and the increasing demand for intelligent vehicles on the road, security is becoming a prime concern in VANETs as they are directly related to human lives. VANETs are distributed network systems that have complex communication mechanisms. By studying related works, authors observed that VANETs were prone to unauthorized access, eavesdropping, hardware tempering, denial of service, surveillance, a replay of the legitimate message, imposture, bogus information, etc., as there are many attacks present which can affect the confidentially, integrity, and availability of the vehicle and the data attached to it. As a result, there is a great need for enhancing the security of vehicles in a real-time environment. In the future, driverless cars will be on the road everywhere and the potential security mechanisms will provide security to vehicles and humans.

3. Proposed Method

Biometric identification technologies are used in VANETs to confirm the authenticity of the communicator. The biometric data are kept in the vehicle's computing data registers. If a hacker or intruder succeeds in rooting the device, they will have access to the biometric fingerprints. In the proposed design, first, we encode the fingerprint visuals using morphology, and the system remembers these encrypted images using a recurrent neural network. For the test-bed analysis, we adopted a layer-recurrent neural network technique.

The idea is to generate manual signatures into pseudo-digital-signatures and then analyze them for validation using an ANN's I-EBP model. The simulation was carried out in MATLAB using the Neural Network Toolbox. The driver's manual signatures were collected on white paper, and these manual signatures were then translated into a soft form using two techniques. The first step involved merely clicking the image and saving it to a separate directory on the vehicle's microcontroller. Second, this application required a smartphone camera for real-time capturing by utilizing the Image Acquisition Toolbox. The captured images by this framework were also saved in a specific location on the storage device. Different morphological approaches were then used to transform these image data into a new format. The classified format images were unrecognizable and were utilized as a pseudo-digital-signature in VANETs to verify the communicator's authenticity. Following that, the network was trained using the I-EBP method of the ANN to shorten the identity verification. The system learned quickly and validated the pseudo-digital-signatures. Figure 4 depicts the process of converting a manual signature to a pseudo-digital-signature.

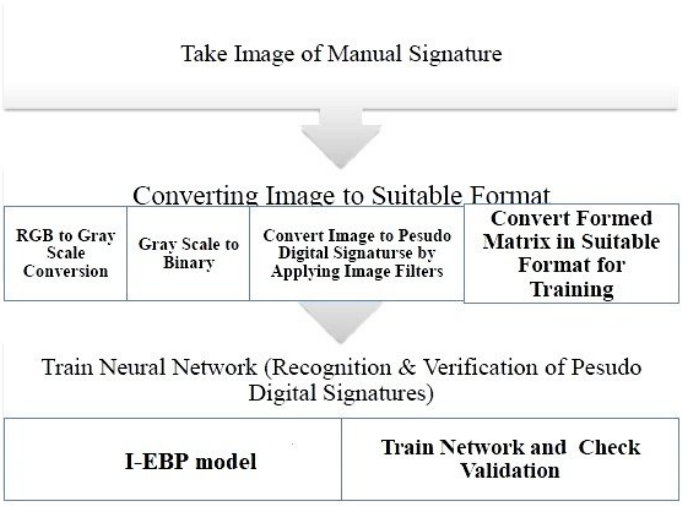


Figure 4. Converting a manual signature to a pseudo-digital-signature.

3.1. System Model

The system model is made up of three units: TA, RSU, and OBU, as shown in Figure 5. The three units are explored below.

- **Trusted authority (TA):** The TA is equipped to maintain a trustable estimate and has storage requirements for the evaluation of RSUs and OBUs under its authority. If the system contains fraudulent or malicious information, the TA can determine the actual source of the information. All entities recognize the TA as an absolute source of confidence in the VANET environment, and incorporating a TA is not an option. To avoid a single failure point or bottleneck induced by traffic problems, TAs should be superfluous.
- **RSU:** An RSU is a component of vehicles and infrastructure that is permanent. The RSU can share information with the vehicle's OBU and the TA via the I-EBP protocol and ensure secure wired connectivity, respectively. The RSU can inform the driver about vehicular traffic conditions such as heavy traffic and collisions. Notifications about traffic from the signatory, i.e., the driver, may also be authenticated and conveyed to the TA or applied on their own.
- **OBU:** The vehicle receives an OBU that supports the I-EBP method. The OBU sends a vehicle-related notification to the nearest OBU or RSU regularly, informing them about traffic status updates such as speed, location, and threat warnings.

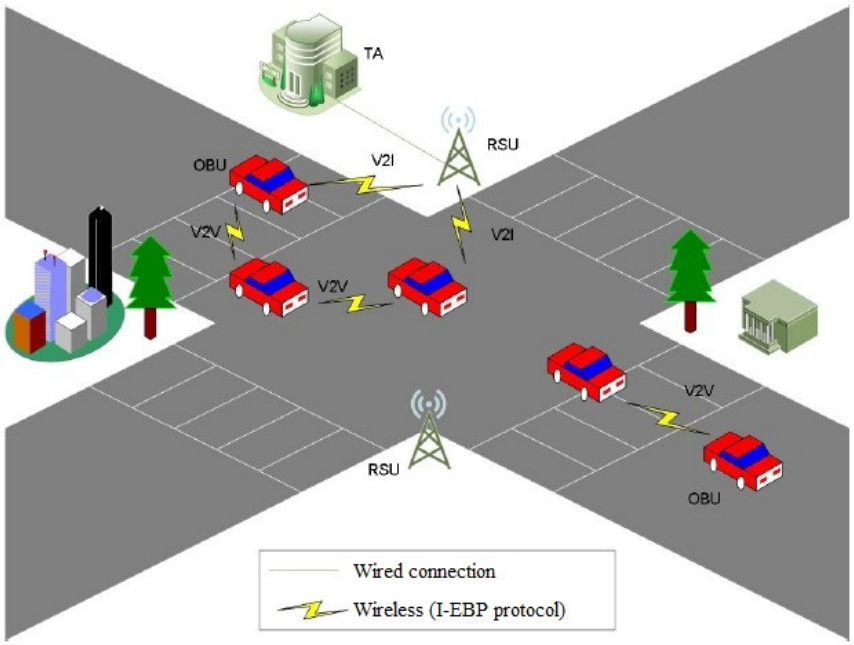


Figure 5. System model of proposed method.

3.2. Design Goals

This article's focus is on the following protective measures, and the security goal should be achieved:

- **Identification and data protection:** RSUs, vehicles, and third-party respondents could infer the true vehicle authenticity from any traffic-related information.
- **Tracking and tracing:** If required, the TA is the one signatory able to obtain the true vehicular identity.
- **Enhancing the security:** The adversary vehicle or RSUs cannot effectively detect the anonymized object by connecting some of the sending messages and signature verification.
- **Unobservability:** A vehicle is allowed to access a source of information or facility without being recognized by others—especially third parties—when using the assistance or facility.
- **Message integrity and validity:** RSUs and OBUs must inspect each vehicular notification, and endpoints should be able to recognize any improvements or falsifications in the packet transmitted.

3.3. Changing Keys in Binary Form

For training a network, keys cannot be used in hexadecimal form, so, first, we converted them into binary form. This task was performed using coding done in MATLAB. We present the pseudocode in MATLAB for converting hexadecimal to binary form as per Algorithm 1. Each hexadecimal character was converted to a 16-bit binary value. Afterward, the conversion keys were reshaped to form a matrix.

3.4. Network Training

In this stage, the private keys in the binary form obtained from the previous stage were used as input and target values. The improved feed-forward backpropagation model was then used to train the network. The "Trainlm" function was used for training. As activation functions, the "Tansig" and "purelin" functions were used. Initially, the weights and bias values were taken as 0 and 1. The weights, bias values, and network parameters were the outputs of this training process.

Algorithm 1 Pseudocode for hexadecimal to binary conversion.

```
Start procedure
input=("Private Key in Hexadecimal form")
n = length(input)
hex_str = reshape(input,n,1)
dec_str = hex2dec(hex_str);
binary = dec2bin(dec_str,32);
for i=1: size(binary,1) do
    for j=1: size(binary,2) do
        newin(1,(i-1) × size(binary,2)+j) = str2num(binary(i,j))
    end for
end for
input_final = reshape(newin,32,n);
target=input_final;
End procedure
```

3.5. Using Network Parameters to Replace Keys

After the network training, the weights and bias values were obtained. Now, these network parameters could be saved in hardware and the file containing the keys could then be deleted from the storage. The weight and bias values were in the form of matrices and by only viewing them, it was impossible to find out the size of the network that had been used for training. Once trained, the network was never showed the same training dataset again. The weight values differed each time we trained the network with the same input values. An encryption mechanism was proposed based on the improved perceptron model for converting the message to a key matrix form.

3.6. Creating a Matrix from Private Keys

Private keys were kept in the user profile, located in the root directory. After logging in, an authenticated user could readily obtain it from the root directory. In the proposed work, the generated key could be in alphanumerical form, thus, the first step was to convert it to binary form, following which it was turned into an appropriate matrix that was used as an input in the training stage. The private key was used for sample training as follows,

01B703C327477634349CA686C57949014B2E8AD2C862B2C9D748896Aw8B91F636F275D6E8
CD19906027315735644D95GD6763CEM49F56AC2F376E1CEE0EBF282DF439906F34D96E08
5BD5656KL931F313D72D395EFE33CBFF29E4030B3D05A28FB7F18EA27637B07957D32F2B
DE8706227D04665EC91BAF8B1AC3EC9144AB7F21.

It is a 256 bits private key that is used for producing digital signatures in VANETs. The key was converted into 256 × 8 matrices and used for validating the digital signatures, as shown in Figure 6.

3.7. Learning Process

During the learning stage, the enhanced perceptron model was used to attain the private key. The 256 × 8 matrix key form was used as an input value as well as the network's target value. The output of an enhanced perceptron model could be 0 or 1. If the input transfer function was equal to or greater than 0, the enhanced perceptron neuron output was 1, otherwise, it was 0.

Let w = weight, b = bias, x = input.
Then, $f(x) = w \cdot x + b$.
$f(x) = 0$ if $w \cdot x + b < 0$ otherwise $f(x) = 1$.

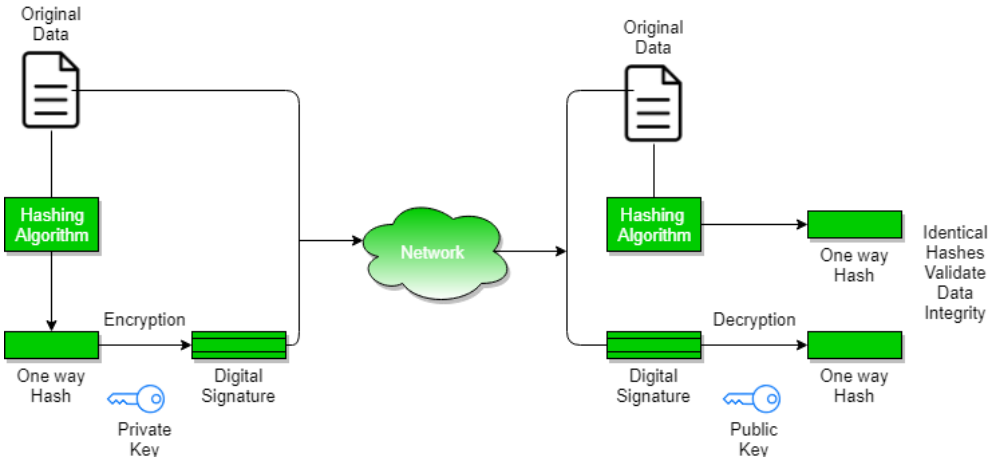


Figure 6. I-EBP proposed model for validating the digital signatures.

The enhanced perceptron model was processed as per the stages listed below:

1. **Scaling input up and down:** The corresponding weight values were multiplied by the input values. The weights were initially random and during the learning phase, these random weights were modified based on the error values.
2. **Activation:** The result was fed into an activation function (transfer function), which converted input values to output values. The enhanced perceptron model could be trained using 2 types of training and learning functions. We could not use the symmetric hard-limit transfer function as its output was 1, if the net threshold was achieved, otherwise, the output would be -1. We were working on binary values and bound to produce 0 and 1, that was why we could not use this transfer function. We trained the proposed network with all discussed possibilities. Table 2 shows the various functions used during the training. Figure 7 shows the hard-line (hard-limit transfer) function.

Table 2. Transfer and learning functions in the networks.

Network	Size	Transfer Function	Learning Function
Network1	256 × 8	Hardlim	Learnp
Network1	256 × 8	Hardlim	Learnpn

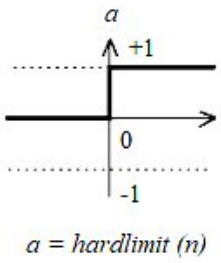


Figure 7. Hard-limit transfer function.

3.8. Replacing Keys with Network Parameters

After the training process, the network simulation parameters were retrieved from the weight and matrix values. The original image records were removed from memory storage and replaced with the latest network configuration.

Algorithm 2 contains the code for converting manual signatures to pseudo-digital-signatures. Here, the main objective of this procedure was to process the manual signatures, to convert into suitable formats, including a conversion from RGB to gray scale and gray scale to binary, the image to pseudo-digital-signatures by applying image filters using the I-EBP technique, and to form a matrix in a suitable format for training

purpose. Finally, it tested the train network and checked the validation of the digital signature. After that, all pseudo-digital-signatures were transformed into a 100 × 100 matrix for use in the Neural Network Toolbox. The matrix's attributes were all in a binary representation. The network's input and output were identical.

Algorithm 2 Pseudocode to convert manual signature to pseudo-digital-signature.

```
Start procedure
Input: Sample image
Output: Convert manual signature to pseudo-digital-signature.
Clear all
sig = imread(fig4.png);
figure.imshow(sig);
msig = imresize(sig, [100,100]);
figure.imshow(msig);
manualsig = rgb2gray(msig);
manualsig = im2double(manualsig);
manualsig = im2bw(manualsig);
manualsig = bwmorph(manualsig, 'thin', inf);
manualsig = bwmorph(manualsig, 'thick', inf);
manualsig = manualsig;
figure.imshow(manualsig);
input = double(manualsig);
input = reshape(input[], 100);
target=input;
End procedure
```

4. Security Analysis

4.1. Reading Keys from Temper-Proof/Resistant Devices

In VANETs, cryptographic keys are stored in the temper-proof modules of temper-proof devices and any authorized user after login can read keys from these devices. A security number, username, and password are used to retrieve these keys. The keys are stored in encrypted form in these devices. Table 3 presents some private keys produced during cryptographic operations with different specifications of ECC. All these private keys are in Privacy-Enhanced Mail (PEM) (base 64) form, which is a standard format for open SSL, and we converted those PEM format keys into a hexadecimal format using online software.

Table 3. Training of network using ECC's private keys.

Specification	Standard Private Key of ECC 14	Hexadecimal Key Form
Secp112r1	!DY|$x(-;s^38;&590	0D8C6CDFCE7D
Secp128r1	!i7I:*H*103*Y*VgGRaS@%	8BB207D74DD856019169
Secp160r1	31H35IUF-ctDgWc<?=5%,#t{_	DF51F7E4850572D0E059
Secp192r1	!_&>%%2hHO@]#@C$4ONgT MEI^^,a7%	DA11CE0B838D81330421AE
Secp224r1	&{N}lg5SEpkeLQkD#W&y3Y *UV>4vT2SRi<f	365839484A6478B4240D 6CB7614578BD3D924627
Secp256r1	!G,C}NGZq%nRhNCV]T<C0p Arfc17aUiKl9-1q%e^	18234666A9D184D0954 C2D2902B7DCD7B694 88A97DD6A7
Secp384r1	#pmDn!Ps/gY=vbK+JR_- !H7vye,|K3S$J6E5)b%)&zz_qIc 2[rFU@iQZ&I7*	A660E73ECFE062F6 CAF89447EEFC9E2B7489 E84E5BCF3A88736 AC552241923

The proposed technique provided a unique identifier based on the encryption key, public key, and message using the PKI approach. This approach is well known for building confidence. Security messages generated by RSUs for all neighboring vehicles do not have to be secured, but the recipient's authenticity must be verified because a falsified entity could also use this type of communication to induce incompetence. To verify its authenticity, the message's recipient digitally signs it with a set of private or public digital certificates designated for it. These signatures are combined with the message, certificate, and timestamp and sent to the other end. Following that, the transceiver end node validates the signer's identification using a signature, digital certificate, and public key. Anyone with access to the encryption key can generate a digital certificate that seems to be authorized by the actual owner of the credential. Refer to Table 4 for acronyms and variables.

Table 4. Acronyms.

Symbols	Meaning
α	Vehicle's transmitted message
κ	Recipient
β	Communication message
τ	Timestamp to verify the message authenticity
p_k	Private key
γ	Recipient's digital certificate
M	Thinning of an image
I	Structuring element
RSU_{sc}	RSU service capacity
T_{auth}	Time required to authenticate a single signature
V_d	Density of the vehicles
V_s	Average speed of vehicle
T_r	Transmission range of the RSU network
S_{acc}	Accurate signature

Let α be the vehicle that transmits a message, κ be the recipient, β be the communication message, τ be the timestamp to verify the message authenticity, p_k be the private key, and γ be the recipient's digital certificate, then the technique functions as per Equation (1).

$$\alpha \rightarrow \kappa : (\beta, DSign''p_k''\{\beta \mid \tau\}, \gamma''\alpha'') \quad (1)$$

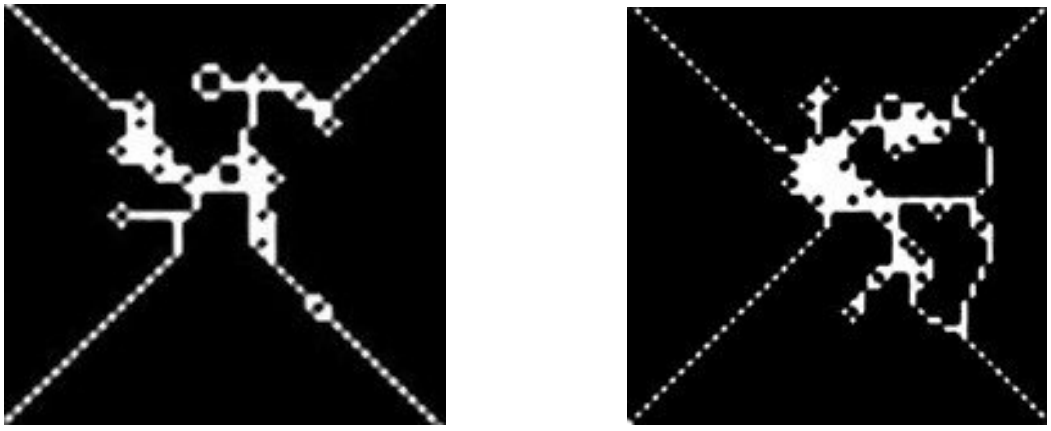
To protect privacy, a vehicle must securely store keys and change them regularly. Every key is stored in a TPD to improve security. However, due to the expensive nature of this equipment, we attempted to propose an approach that differed from existing techniques for ensuring the confidentiality and safety of records in vehicles. To transform the way private keys were stored, we attempted to re-encrypt the keys used during the generation of digital signatures. The keys in the proposed method were stored as a set of system parameters derived during the ANN process of learning. These weight values were fully safe because it was difficult to generate the original input value without understanding all of the network's information.

Morphology is a wide set of image processing operators which are used to process various images based on structure and, as a result, produce an image of the same size. The input image must be in binary or grayscale form for morphology. For binary images, usually, black pixels are used for the background and white pixels are used for the foreground region, but the reverse can also be done in special cases. The coordinate sets of binary images for foreground pixels are represented in two-dimensional Euclidean coordinates with the origin at one of the corners. The origin is required to be at the corner to make all elements positive.

The actual fingerprints used as input in the network are presented in Figure 8a,b and encrypted simulation outcomes after employing the morphology algorithms spur and skel are shown in Figure 9a,b.

(**a**) Actual fingerprint (**b**) Extracted fingerprint

Figure 8. Training in different file formats from actual fingerprints.



(**a**) Morphology fingerprint-1 (**b**) Morphology fingerprint-2

Figure 9. Applying morphology to actual fingerprints.

For grayscale images, to define heights (above the base plane), the intensity values are used and the images are represented in three-dimensional Euclidean space. We need one structural element, also known as a kernel, of size 3×3 with an origin at the center pixel for morphological operations. It is necessary to place the origin at the center pixel because some coordinate elements have negative values. The structure element is small in size in comparison to the input image. The images in morphology are processed using simple mathematical operations from set theory.

Thinning and thickening operations are nearly the same as erosion and dilation. Thinning is the same as erosion and is used to delete some pixels from the foreground. This is related to the hit-and-miss transformation operation in which both foreground and background pixels are taken to carry out particular patterns. The hit-and-miss operation is the basic operation of morphology, and all other operations can be derived from this operation. The behavior in a thinning operation is calculated by the structuring element, and this structuring element is a hit-and-miss operation. The thinning of an image M by a structuring element I is computed as per Equation (2):

$$thin(M, I) = M - hit - miss(M, I) \tag{2}$$

Thickening is nearly the same as dilation or closing. It is normally applied to binary images, and a binary image is produced as the output. This operation is also related to the hit-and-miss transform as the structuring element used to determine the behavior of

the thickening operation is described by the hit-and-miss transform. The thickening of an image M by a structuring element I is computed as per Equation (3):

$$thicken(M, I) = M \cup hit - and - miss(M, I) \tag{3}$$

From this Equation (3), the thickened image is a result of a combination of the original image plus the addition of foreground pixels, which are switched on because of the hit-and-miss transformation. Now, the thickening operation is calculated by translating the structure elements' origin for each possible pixel position in the binary image. Each position's origin is compared with the pixel at that position. If both the foreground and background pixels of the structuring element and the binary image exactly match, then the pixel of the image underneath the origin of the structuring element is set to one, or we can say foreground. In the opposite case, it will remain unchanged. Thus, here, we can say that thickening is the dual of thinning. The thickening of the background pixels is the same as the thinning of foreground pixels.

4.2. Signature Validation

The proposed system used RSA logic to validate the authenticity of both OBU and RSU. RSA logic is a widely used platform for achieving specific security requirements for collaborative identification and key agreement. The following are the main abbreviations and definitions for RSA logic:

- S, R: Primary attendees.
- X_m: Communication notifications.
- SK: Common key.
- $S \mid\equiv R$: S has confidence and trust in R.
- $S \mid \lhd X_m$: S notices X_m.
- $S \mid\sim X_m$: S transmitted X_m.
- $\#(X_m)$: X_m communication messages are new.
- $S \overset{SK}{\longleftrightarrow} R$: S and R share information using SK.
- $\mid \overset{Pub}{\longrightarrow} R$: R has a public key (Pub) that corresponds to a private key (Pri).
- SR: R can be controlled by S.
- $(X_m)_{SK}$: SK is hashing the message X_m.

The following are the fundamental rules of the RSA logic procedure:
The communication messages are derived as per Equations (4) and (5).

$$\frac{S \mid\equiv S \overset{SK}{\longleftrightarrow} R, S \lhd (X_m)_{SK}}{S \mid\equiv R \mid\sim X_m} \tag{4}$$

$$\frac{S \mid\equiv S \rightarrow \text{Pub}\, R, S \lhd (X_m)_{Pub^-}}{S \mid\equiv R \mid\sim X_m} \tag{5}$$

The freshness function is computed as per Equation (6).

$$\frac{S \mid\equiv \#(X_m)}{S \mid\equiv \#(X_m, Y_m)} \tag{6}$$

The pseudo-digital-signature validation is based on Equation (7).

$$\frac{S \mid\equiv \#(X_m), S \mid\equiv R \mid\sim X_m}{S \mid\equiv R \mid\equiv X_m} \tag{7}$$

The authentication is checked as per Equation (8).

$$\frac{S \mid\equiv R \Rightarrow (X_m), S \mid\equiv R \equiv X_m}{S \mid\equiv X_m} \tag{8}$$

4.3. Validation of Random Oracle Model (RoM)

The proposed scheme's protection is demonstrated using the RoM. A_1 and A_2 are two types of opponents to the certificate-less cumulative authentication scheme. Opponent A_1 does not have access to the game's master key, but it can substitute the valid participant's public key. Opponent A_2 is aware of the game's master key but lacks the capability to substitute the valid participant's digital certificate.

Theorem 1. *When there is an opponent A_1 who can break the iterative process that involves the pseudo-digital-signature with non-negligible benefits after making various communication attacks and authenticity attack requests in quadratic instances, then there is a distinguisher B who can hold for a quadratic duration to fix a density-based location privacy (DLP) issue with non-negligible benefits $Adv[B] \geq \frac{1}{ne(q_s+n)}(1-\frac{q_{ppk}}{2})$ under the RoM (In which q_{ppk} and q_s are the total of incomplete key distribution queries and single signature queries, respectively, and n represents the number of users participating in the cumulative authorization.)*

Proof. A_1 is an intruder to the DLP challenges, whereas B is an adversary. B's objective is to resolve the DLP challenge that used A_1, so as to assess x, considering a random DLP problem P, xP. □

B executes the network configuration's automated system and produces the public key parameters $q, H_1, H_2, H_3, H_4, p_{pub}$ and communicates them to A_1. B sets $p_{pub} = xP$ and x is the game's authenticated key. B contains the lists $L_2, L_3, L_4, L_{ppk}, L_{sk}$, and L_s to monitor the H_2 oracle, query for private value generation, and query for signature, respectively. The possibility of choosing pseudoidentification (PID^*) is $\theta \in [\frac{1}{q_s+n}, \frac{1}{q_s+1}]$ (The query of a single signature is q_s, and the total number of cumulative participants in the forgery stage is n.)

The following query is posed by opponent A_1.

H_2 oracle: B maintains the initially empty list $L_2 = \{m_i, PID_i, P_i, R_i, t_i, h_i\}$. Once B receives the H_2 query from A_1, when there is a correlating permutation in the list L_2, the significance h_i is immediately restored to A_1. Instead, B chooses at irregular intervals $h_i \in PID_q^*$, adds $\{m_i, PID_i, P_i, R_i, t_i, h_i\}$ to list L_2, and forwards back h_i to A_1.

Query for private value generation: B maintains the list $L_{ppk} = \{PID_i, d_i\}$, which is initially empty. Once B receives a private key generation query, when there is a correlating permutation in the list L_{ppk}, B gets back d_i to A_1 immediately; alternatively, B needs to check if PID_i and PID^* are equivalent or not.

Query for signature: Once B has received a signature query from A_1 for the authentication key pair PID_i, m_i, P_i, B executes the following:

- If $PID_i \neq PID^*$, B selects a number at irregular intervals $r_i \in PID_q^*$.
- Alternatively, B renounces and the experiment is revoked.

The chances of B's achievements are assessed. To begin, establish the following occurrences:

- F_1 is the failure of at least one authenticity PID_i, $1 \leq i \leq n$ to perform a private key generation query;
- F_2 is the failure of B to disconnect during the signature query.

5. Results and Observations

One of the cryptographic approaches utilized in VANETs is the usage of digital signatures. Protecting private keys from outsiders is a complicated process in digital signatures. Furthermore, verifying these keys in real-time scenarios of fast-moving vehicles is a time-consuming and challenging task. As a result, a very effective hardware device is utilized, requiring more work and money. Therefore, the authors attempted to validate private keys using one of the single-layer networks. After the learning process, we stored the keys as network parameters to make them more secure in hardware resources. Figure 10

displays the proposed training model, which contained 256 neurons as input and output. Figure 11a,b present the performance graphs of the training process of both networks. Table 5 presents the comparison of the epochs, time taken, and best performance of the networks to train the desired inputs to meet the final goal, which shows that if we train our network with the learning function, it takes less time in comparison to the learning function for the same network parameters and same input values. Table 5 shows the comparison of network 1 and network 2 based on time and epoch.

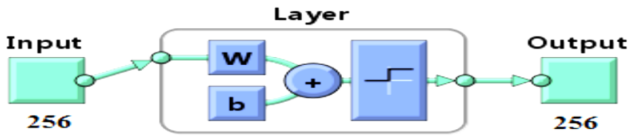
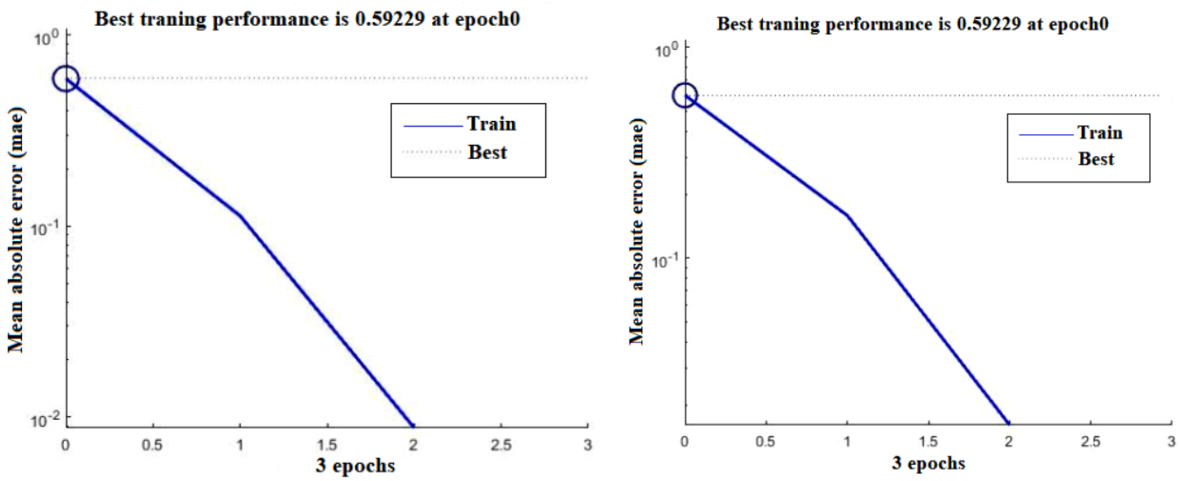


Figure 10. Enhanced perceptron model.



(**a**) Network-1 performance graph (**b**) Network-2 performance graph

Figure 11. Networks performance graph.

Table 5. Performance of network 1 and network 2.

Network	Epochs	Time	Best Performance
Network 1	3	0.01	0.59229
Network 2	3	0	0.59229

5.1. Experimental Design

The complete setup was designed to convert the manual signatures to pseudo-digital-signatures and test for further verification using the I-EBP. The test-bed analysis was based on the Neural Network Toolbox of MATLAB. The driver's manual signatures were taken on paper, and these manual signatures were subsequently transformed into a soft form by using two methods. The first method was to click on the image and save it to a specific folder on the vehicle's computational device. In the second method, the program leveraged the Image Acquisition Toolbox to capture live images from the device's camera.

Network 2, network 3, and network 4 were also fed with a similar type of 100 × 100 matrices obtained from converting images into binary form after applying morphological operations. In this phase, the private keys in binary form were obtained from the previous phase's trained values. The "Trainlm" function was used as a training function. "Tansig" and "purelin" functions were used as activation functions. Initially, the weights and bias values were taken as zero and one. The weights, bias values, and network parameters were the outputs of this training process. Figure 12 depicts the network model of I-EBP.

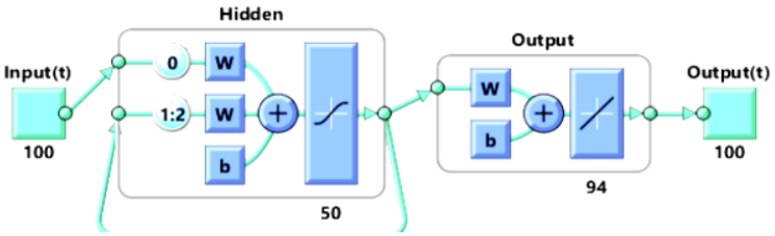
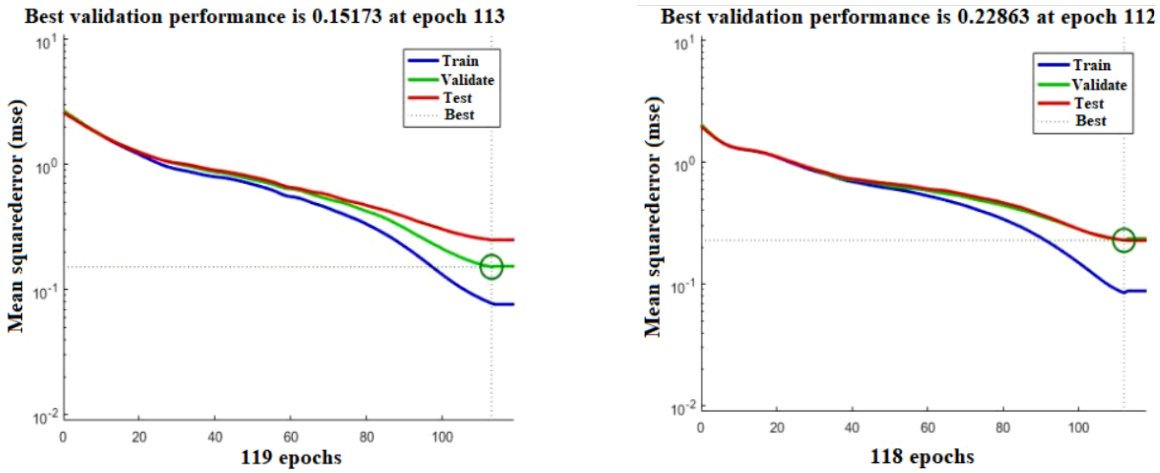


Figure 12. Improved Elman neural network for network 4.

After training the network, the weights and bias values were obtained. At this point, these network parameters could be saved in hardware and the file containing keys could then be deleted from the storage. The weight and bias values were in the form of matrices and by only viewing them, it was impossible to determine the size of the network taken for training. Once the network was trained, it would not show the same training dataset again. The weight values differed whenever we trained the network with the same input values.

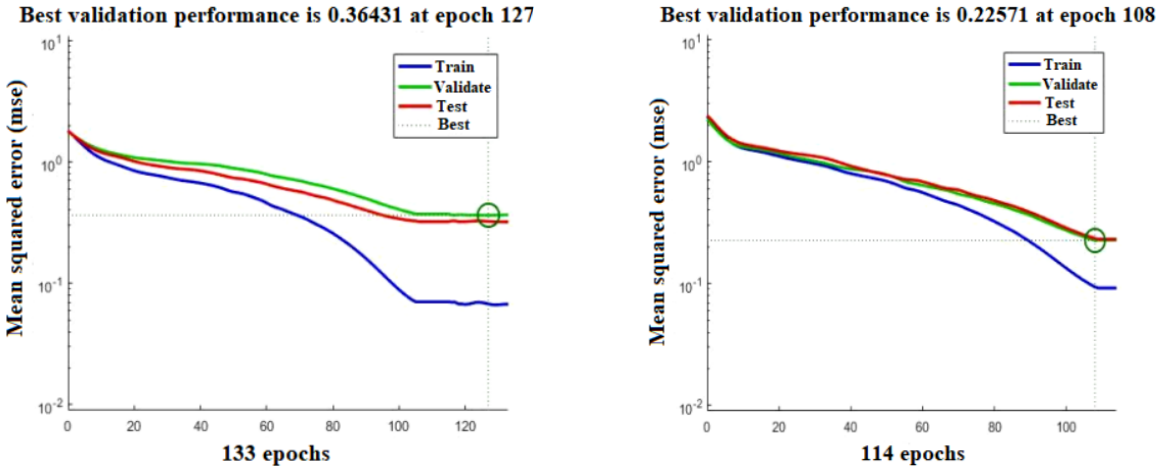
Figures 13a and 14b are related to the training of the first network with 20 neurons in the hidden layer. The training of this network took place for the fourth time. The test-bed analysis was based on several network metrics, such as the number of epochs taken by each training, the amount of time consumed by each training, and the mean squared error (MSE) of various pseudo-digital-signatures. We compared the performance of different networks based on time, epochs, and neurons considered for the hidden layers.



(**a**) Performance of training 1.1 (**b**) Performance of training 1.2

Figure 13. Performance graph for network 1 training.

A comparison of multiple networks based on time and epochs showed that the network with the fewest neurons in the hidden layers took less time to train and the network with the most neurons took the fewest epochs to train to achieve the goal. The primary focus of the research was that if an attacker wanted to compromise the vehicle's security by forging the pseudo-digital-signature, they had to establish a manual signature and then employ various network filters to convert the manual signature into the pseudo-digital-signature. However, this process was impossible since a trained network would never provide the same parameters for the same image or data. The only information an intruder could access was the weights and bias values generated during network training. Consequently, the attacker was unable to create the manual signature from the memory of a vehicle.



(**a**) Performance of training 1.3 (**b**) Performance of training 1.4

Figure 14. Performance graph for network 2 training.

The private key of the second specification, i.e., secp128r1, took 17 epochs for learning and we adjusted the values of the weights three times to get the desired output. The private key of the third specification, i.e., secp160r1 took nine epochs in two training rounds, but secp192r1/nistp192, secp224r1/nistp224, secp256r1/nistp256 took 11, 8, 8 epochs, respectively, and learned in the first training. The seventh specification, i.e., secp384r1, took 14 epochs in two rounds of training. The time taken by all specifications also varied with the number of epochs. Secp192r1, secp224r1, and secp256r1 took 0.01 seconds but the number of epochs was high in the case of secp192r1. Thus, this led us to conclude that secp224r1 and secp256r1 were comparatively good because they took less time and a smaller number of epochs during training. As a result, the secp224r1 and secp256r1 are best to use in practice. Table 6 illustrates the epochs and times required by various standards.

Table 6. Epochs and times required by various standards.

Specification	Total No. of Bits in the Private Key	Epochs	Time
secp112r1	191	4 + 4 + 5 + 5 = 18	0.04 s
secp128r1	319	5 + 6 + 6 = 17	0.03 s
secp160r1	372	4 + 5 = 9	0.02 s
secp192r1/nistp192	478	11	0.01 s
secp224r1/nistp224	641	8	0.01 s
secp256r1/nistp256	743	8	0.01 s
secp384r1/nistp384	981	7 + 7 = 14	0.02 s

To assess the calculation of computational overhead of the proposed and existing VANET methods [24–30], Figure 15 depicts the computational overhead of signing and validating one message using one of the currently available techniques. As shown in Figure 15, the proposed approach had a reduced computing cost between the signature creation and single signature verification compared with the other existing VANET techniques. Figure 16 depicts the overall verification delays of the existing schemes considered in this work. The aggregate verification latency appeared to rise linearly with the number of signatures. Figure 17 shows the aggregate signature size of the proposed and existing schemes.

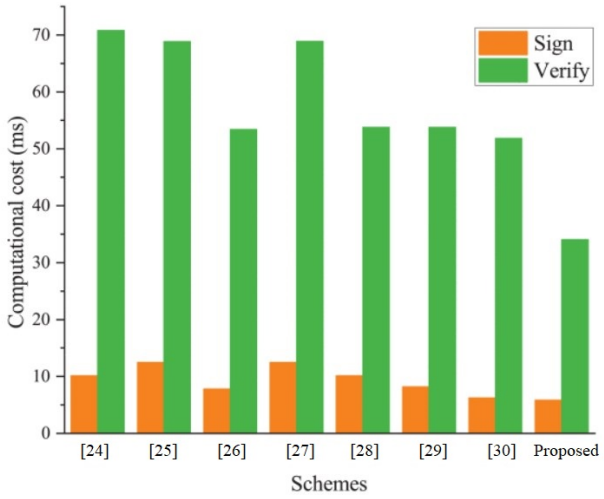


Figure 15. Computational overhead of one message over proposed vs. existing schemes.

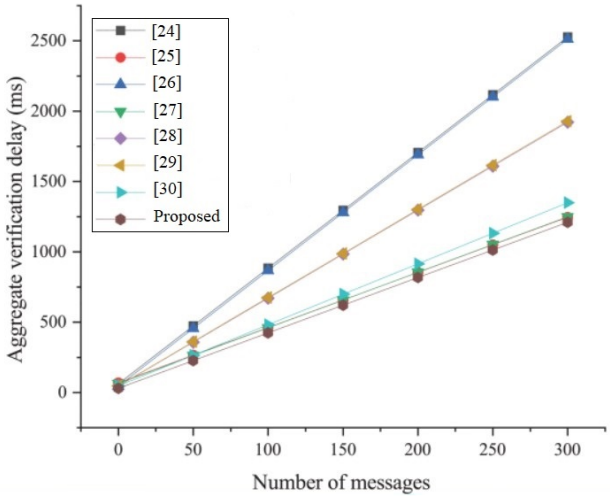


Figure 16. Aggregate verification delay of proposed vs. existing schemes.

Figure 17 shows that the computation overhead yielded by the proposed scheme was lower than that generated by schemes [24,26,27,29,30], and significantly higher than that yielded by schemes [25,28]. However, when compared to schemes [27,30], the proposed system had a lower computation overhead.

Figure 18 depicts the communication cost. For the proposed system, the cumulative communication overhead was comparatively lower. Drivers' primary concern is the retention of their privacy. As a result, we contend that VANET communication systems should be addressed by meeting all individual privacy needs. In comparison to existing works [24–30], the proposed system fulfilled privacy and security concerns. Table 7 compares VANET schemes [24–30] to the suggested method. Noticeably, these methods placed a strong emphasis on information privacy. Despite its significance from a VANET perspective, the experimental data protection demand was not fully met. Only such initiatives ensure the confidentiality and anonymization of the recipient and sender. Due to the overhead, unobservability was overlooked.

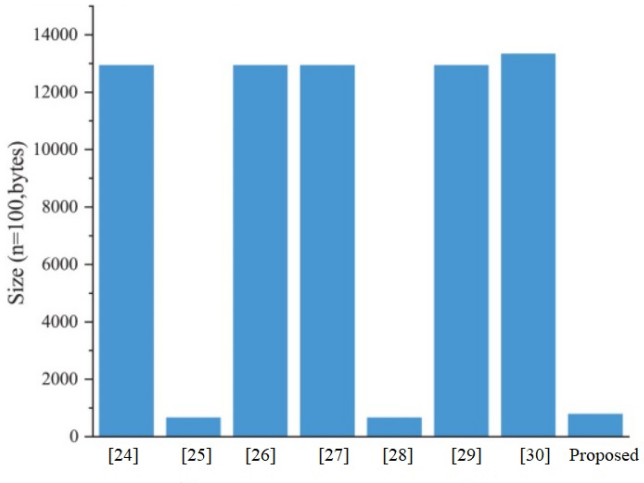


Figure 17. Aggregate signature size of proposed vs. existing schemes.

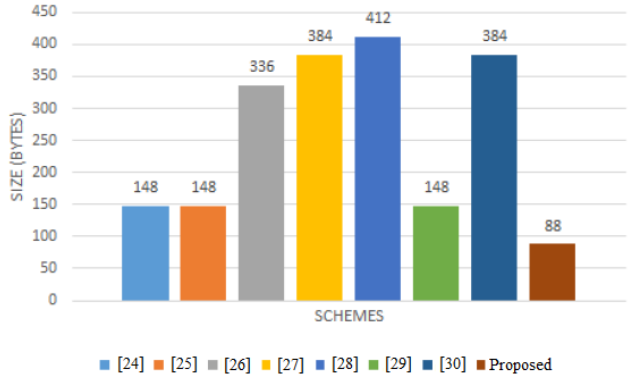


Figure 18. Communication costs of proposed vs. existing schemes.

Table 7. Privacy characteristics based on safety analysis.

Properties	[24]	[25]	[26]	[27]	[28]	[29]	[30]	Proposed
Identity Privacy Preservation	✓	✓	✓	✓	✓	✓	✓	✓
Un-linkability	✓	✓	✓	✗	✗	✗	✓	✓
Un-observability	✗	✗	✗	✗	✗	✗	✗	✓
Traceability	✓	✓	✗	✓	✓	✓	✓	✓
Message Integrity and Authenticity	✓	✓	✗	✓	✓	✓	✓	✓
Resistance to Replay Attacks	✓	✓	✓	✗	✓	✓	✗	✓
Resistance to Impersonation Attacks	✓	✓	✓	✓	✓	✓	✓	✓
Resistance to Modification Attacks	✓	✓	✗	✓	✓	✓	✓	✓
Resistance to Man-in-the-Middle Attacks	✓	✓	✓	✓	✓	✓	✓	✓

5.2. Analysis of Practicality

The RSU service capacity (RSU_{sc}) was introduced for assessing the RSU's computational power, and it was determined as per Equation (9).

$$RSU_{sc} = \frac{S_{acc} \times T_r}{T_{auth} \times V_d \times V_s} \tag{9}$$

T_{auth} is the time required to authenticate a single signature. T_{auth} was 34:0828 ms here. Let V_d represent the density of the vehicles in the RSU's communication range, which ranged from 600 to 800 m ; V_s represents the average speed of a vehicle, which ranged from 5 m/s to 20 m/s; S_{acc} represents the likelihood that the signature is accurate; and T_r represents the transmission range of the RSU's network coverage, which was assumed to be 1000 m.

Figure 19 shows the RSU_{sc} with different vehicular speeds and relative density; the RSU_{sc} slowly decreased as the density of the vehicles and the speed increased. Furthermore, the RSU could authenticate eight signatures in 300 ms. As a result, it was possible to indicate that the density of vehicles should be reduced to achieve a higher RSU_{sc} in this mechanism.

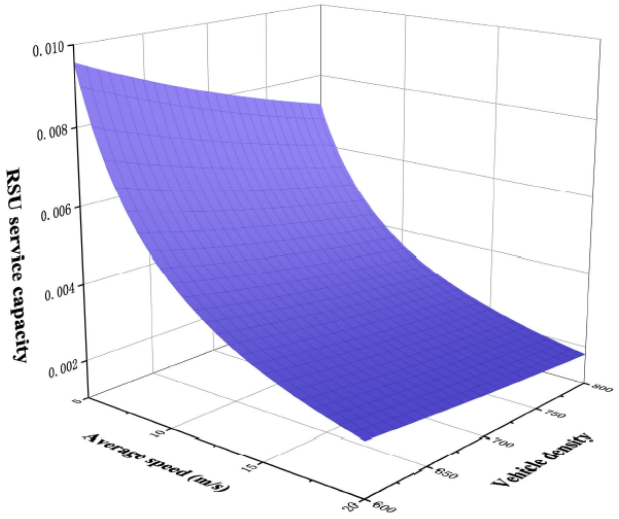


Figure 19. RSU_{sc} with different vehicle's speeds and densities.

6. Conclusions and Future Scope

In VANETs, digital signatures are used for transmission and processing records with security. However, it is difficult to verify digital signatures in real-time scenarios. Private keys are used to validate digital signatures and it is hard to manage private keys in a real-time ad hoc environment. In this paper, an enhanced perceptron model and an I-EBP model were used for the speedy and secure storage of keys in the VANET data registers. Higher computational systems are deployed in VANETs to secure the cryptographic key. The proposed scheme avoided the requirement for additional hardware. The weight matrix obtained during the training phase could be used to substitute keys because it was essential to disentangle keys from the weight matrices without accessing the real network, which led to a lower computational overhead, a reduced aggregate verification delay, and an effective aggregate signature size. Furthermore, the system validated the keys in milliseconds using network parameters obtained during the training process. The future scope of this work can be extended by introducing deep learning (DL) and convolution neural network (CNN) techniques that can be used in real-time streaming for data identification and verification. These can be applied to moving vehicles in a real-time environment to train large data sets.

Author Contributions: Conceptualization, A.J., P.C. and S.K.; methodology, A.J., P.C. and J.S.; software, A.J., P.C. and Ț.F.-E.; validation, M.T.C.; analysis, A.J., P.C. and Ț.F.-E.; investigation, M.T.C.; resources, M.T.C. and P.C.; writing—original draft preparation, A.J. and P.C.; writing—review and editing, A.J., P.C., S.K. and J.S.; supervision, J.S. and M.T.C.; project administration, M.T.C. All authors have read and agreed to the published version of the manuscript.

Funding: This paper was partially supported by UEFISCDI Romania and MCI through projects SOLID-B5G, 5G-SAFE+, IMMINENCE, EREMI, PREVENTION, DAFCC, RECICLARM, MULTI-AI, F4itech, UPSIM, SmartDelta, USWA, STACK, ENTA and by European Union's Horizon 2020 research and innovation program under grant agreements no. 883522 (S4ALLCITIES) and no. 101016567 (VITAL-5G).

Data Availability Statement: After signing a nondisclosure agreement, the data utilized in the study were obtained from UEFISCDI Romania and MCI through projects SOLID-B5G, 5G-SAFE+, IMMINENCE, EREMI, PREVENTION, DAFCC, RECICLARM, MULTI-AI, F4itech, UPSIM, SmartDelta, USWA, STACK, and ENTA. As a result, the data from the resource model cannot be shared.

The solutions, however, can be shared. Please contact the primary and corresponding authors for clarification.

Acknowledgments: We are thankful for the support of UEFISCDI Romania and MCI through projects SOLID-B5G, 5G-SAFE+, IMMINENCE, EREMI, PREVENTION, DAFCC, RECICLARM, MULTI-AI, F4itech, UPSIM, SmartDelta, USWA, STACK, ENTA and European Union's Horizon 2020 research and innovation program under grant agreements no. 883522 (S4ALLCITIES). This work was supported by a grant from the Gheorghe Asachi Technical University of Iași: postdoctoral research—2022 and POCU—InoHubDoc.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Abbasi, I.A.; Shahid Khan, A. A review of vehicle to vehicle communication protocols for VANETs in the urban environment. *Future Internet* **2018**, *10*, 14. [CrossRef]
2. Kumar, A.; Jain, A. Image smog restoration using oblique gradient profile prior and energy minimization. *Front. Comput. Sci.* **2021**, *15*, 1–7. [CrossRef] [PubMed]
3. Zhang, L.; Kang, B.; Dai, F.; Zhang, Y.; Liu, H. Hybrid and Hierarchical Aggregation-Verification Scheme for VANET. *IEEE Trans. Veh. Technol.* **2022**, *71*, 11189–11200. [CrossRef]
4. Jain, A.; Dwivedi, R.K.; Alshazly, H.; Kumar, A.; Bourouis, S.; Kaur, M. Design and simulation of ring network-on-chip for different configured nodes. *Comput. Mater. Contin.* **2022**, *71*, 4085–4100. [CrossRef]
5. Chithaluru, P.K.; Khan, M.S.; Kumar, M.; Stephan, T. ETH-LEACH: An energy enhanced threshold routing protocol for WSNs. *Int. J. Commun. Syst.* **2021**, *34*, e4881. [CrossRef]
6. Jain, A.; Kumar, A. Desmogging of still smoggy images using a novel channel prior. *J. Ambient. Intell. Humaniz. Comput.* **2021**, *12*, 1161–1177. [CrossRef]
7. Zheng, K.; Zheng, Q.; Chatzimisios, P.; Xiang, W.; Zhou, Y. Heterogeneous vehicular networking: A survey on architecture, challenges, and solutions. *IEEE Commun. Surv. Tutor.* **2015**, *17*, 2377–2396. [CrossRef]
8. Rajkumar, M.N.; Nithya, M.; HemaLatha, P. Overview of VANETs with its features and security attacks. *Int. Res. J. Eng. Technol.* **2016**, *3*, 137–142.
9. Agarwal, A.K.; Jain, A. Synthesis of 2D and 3D NoC mesh router architecture in HDL environment. *J. Adv. Res. Dyn. Control. Syst.* **2019**, *11*, 2573–2581.
10. Jiang, X.; Yu, F.R.; Song, T.; Leung, V.C. Resource allocation of video streaming over vehicular networks: A survey, some research issues and challenges. *IEEE Trans. Intell. Transp. Syst.* **2021**, *23*, 5955–5975. [CrossRef]
11. Wang, H.; Wang, L.; Zhang, K.; Li, J.; Luo, Y. A conditional privacy-preserving certificateless aggregate signature scheme in the standard model for VANETs. *IEEE Access* **2022**, *10*, 15605–15618. [CrossRef]
12. Tyagi, S.; Dwivedi, R.K.; Saxena, A.K. A High Capacity PDF Text Steganography Technique Based on Hashing Using Quadratic Probing. *Int. J. Intell. Eng. Syst.* **2019**, *12*, 192–202. [CrossRef]
13. Maria, A.; Rajasekaran, A.S.; Al-Turjman, F.; Altrjman, C.; Mostarda, L. Baiv: An efficient blockchain-based anonymous authentication and Integrity Preservation Scheme for secure communication in VANETs. *Electronics* **2022**, *11*, 488. [CrossRef]
14. Mei, Q.; Xiong, H.; Chen, J.; Yang, M.; Kumari, S.; Khan, M.K. Efficient certificateless aggregate signature with conditional privacy preservation in IoV. *IEEE Syst. J.* **2020**, *15*, 245–256. [CrossRef]
15. Denny, D.; Kumar, K.P. Secure Authenticated Communication Via Digital Signature and Clear List in VANETs. *Ecs Trans.* **2022**, *107*, 20065. [CrossRef]
16. Thumbur, G.; Rao, G.S.; Reddy, P.V.; Gayathri, N.B.; Reddy, D.K.; Padmavathamma, M. Efficient and secure certificateless aggregate signature-based authentication scheme for vehicular ad hoc networks. *IEEE Internet Things J.* **2020**, *8*, 1908–1920. [CrossRef]
17. Chithaluru, P.; Prakash, R. Organization Security Policies and Their After Effects. In *Information Security and Optimization*; Chapman and Hall/CRC: Boca Raton, FL, USA, 2020; pp. 43–60.
18. Ye, X.; Xu, G.; Cheng, X.; Li, Y.; Qin, Z. Certificateless-based anonymous authentication and aggregate signature scheme for vehicular ad hoc networks. *Wirel. Commun. Mob. Comput.* **2021**. [CrossRef]
19. Al-Mutiri, R.; Al-Rodhaan, M.; Tian, Y. Improving vehicular authentication in VANET using cryptography. *Int. J. Commun. Netw. Inf. Secur.* **2018**, *10*, 248–255.
20. Akhter, A.F.M.S.; Ahmed, M.; Shah, A.F.M.S.; Anwar, A.; Zengin, A. A Secured Privacy-Preserving Multi-Level Blockchain Framework for Cluster Based VANET. *Sustainability* **2021**, *13*, 400. [CrossRef]
21. Prashar, D.; Rashid, M.; Siddiqui, S.T.; Kumar, D.; Nagpal, A.; AlGhamdi, A.S.; Alshamrani, S.S. SDSWSN—A Secure Approach for a Hop-Based Localization Algorithm Using a Digital Signature in the Wireless Sensor Network. *Electronics* **2021**, *10*, 3074. [CrossRef]
22. Tihanyi, V.; Rövid, A.; Remeli, V.; Vincze, Z.; Csonthó, M.; Pethő, Z.; Szalai, M.; Varga, B.; Khalil, A.; Szalay, Z. Towards Cooperative Perception Services for ITS: Digital Twin in the Automotive Edge Cloud. *Energies* **2021**, *14*, 5930. [CrossRef]

23. ElGhanam, E.; Ahmed, I.; Hassan, M.; Osman, A. Authentication and Billing for Dynamic Wireless EV Charging in an Internet of Electric Vehicles. *Future Internet* **2021**, *13*, 257. [CrossRef]
24. Chithaluru, P.; Singh, K.; Sharma, M.K. Cryptocurrency and Blockchain. In *Information Security and Optimization*; Chapman and Hall/CRC: Boca Raton, FL, USA, 2020; pp. 143–158.
25. Ramakuri, S.K.; Chithaluru, P.; Kumar, S. Eyeblink robot control using brain–computer interface for healthcare applications. *Int. J. Mob. Devices, Wearable Technol. Flex. Electron. (IJMDWTFE)* **2019**, *10*, 38–50. [CrossRef]
26. Hu, X.; Tan, W.; Ma, C.; Xu, H. Certificateless aggregate signature scheme with high efficencicy in vehicular ad hoc network. In Proceedings of the 2020 4th International Conference on Electronic Information Technology and Computer Engineering, Xiamen, China, 6–8 November 2020; pp. 1008–1012.
27. Robshaw, M.J.B.; Yin, Y.L. *Elliptic Curve Cryptosystems*; An RSA Laboratories Technical Note: Bracknell, UK, 1997; p. 10.
28. Chien, H.Y. SASI: A new ultralightweight RFID authentication protocol providing strong authentication and strong integrity. *IEEE Trans. Dependable Secur. Comput.* **2007**, *4*, 337–340. [CrossRef]
29. Kohli, S.; Dhiman, R. Secure Message Communication using Digital Signatures and Attribute Based Cryptographic Method in VANET. *Int. J. Inf. Technol.* **2010**, *2*, 591–594.
30. Tanwar, R.; Balamurugan, S.; Saini, R.K.; Bharti, V.; Chithaluru, P. (Eds.) *Advanced Healthcare Systems: Empowering Physicians with IoT-Enabled Technologies*; John Wiley & Sons: New York, NY, USA, 2022.
31. Yayik, A.; Kutlu, Y. Neural network based cryptography. *Neural Netw. World* **2014**, *24*, 177. [CrossRef]
32. Hasrouny, H.; Samhat, A.E.; Bassil, C.; Laouiti, A. VANet security challenges and solutions: A survey. *Veh. Commun.* **2017**, *7*, 7–20. [CrossRef]
33. He, L.; Zhu, W.T. Mitigating DoS attacks against signature-based authentication in VANETs. In Proceedings of the 2012 IEEE International Conference on Computer Science and Automation Engineering (CSAE), Zhangjiajie, China, 25–27 May 2012; Volume 3, pp. 261–265.
34. Jena, L.; Ammoun, L.; Chithaluru, P. Supervised Intelligent Clinical Approach for Breast Cancer Tumor Categorization. In *Augmented Intelligence in Healthcare: A Pragmatic and Integrated Analysis*; Springer: Singapore, 2022; pp. 15–40.
35. Chithaluru, P.; Stephan, T.; Kumar, M.; Nayyar, A. An enhanced energy-efficient fuzzy-based cognitive radio scheme for IoT. *Neural Comput. Appl.* **2022**, 1–23. [CrossRef]
36. Wang, X.; Li, S.; Zhao, S.; Xia, Z.; Bai, L. A vehicular ad hoc network privacy protection scheme without a trusted third party. *Int. J. Distrib. Sens. Netw.* **2017**, *13*, 1550147717743696. [CrossRef]
37. Chithaluru, P.; Al-Turjman, F.; Kumar, M.; Stephan, T. MTCEE-LLN: Multilayer Threshold Cluster-Based Energy-Efficient Low-Power and Lossy Networks for Industrial Internet of Things. *IEEE Internet Things J.* **2021**, *9*, 4940–4948. [CrossRef]
38. Chithaluru, P.; Kumar, S.; Singh, A.; Benslimane, A.; Jangir, S.K. An Energy-Efficient Routing Scheduling Based on Fuzzy Ranking Scheme for Internet of Things. *IEEE Internet Things J.* **2021**, *9*, 7251–7260. [CrossRef]
39. Zhang, D.G.; Zhang, T.; Zhang, J.; Dong, Y.; Zhang, X.D. A kind of effective data aggregating method based on compressive sensing for wireless sensor network. *EURASIP J. Wirel. Commun. Netw.* **2018**, *2018*, 1–15. [CrossRef]
40. Lamaazi, H.; Benamar, N. OF-EC: A novel energy consumption aware objective function for RPL based on fuzzy logic. *J. Netw. Comput. Appl.* **2018**, *117*, 42–58. [CrossRef]