k – Means Clustering Algorithms for Vehicular Ad Hoc Networks using Certificate Revocation List Validation Scheme

D. Srinivas Reddy[1] **Prof. A. Govardhan[2]** **Dr.V.Bapuji[3]**

[1,3]Dept of CSE, Vaageswari College of Engineering, Karimnagar
[2]Dept of CSE, JNTUH

1. INTRODUCTION

Vehicular ad hoc networks (VANETs) play an important role in wireless communications among vehicles, which raises the popularity of safety and drivers assistance applications [1,2]. In order to establish a reliable vehicular communication environment, the guarantee of nodes credibility is required. Security in vehicular networks is critical and indispensable. The figure 1 shows the secured structure of vehicular communication system.

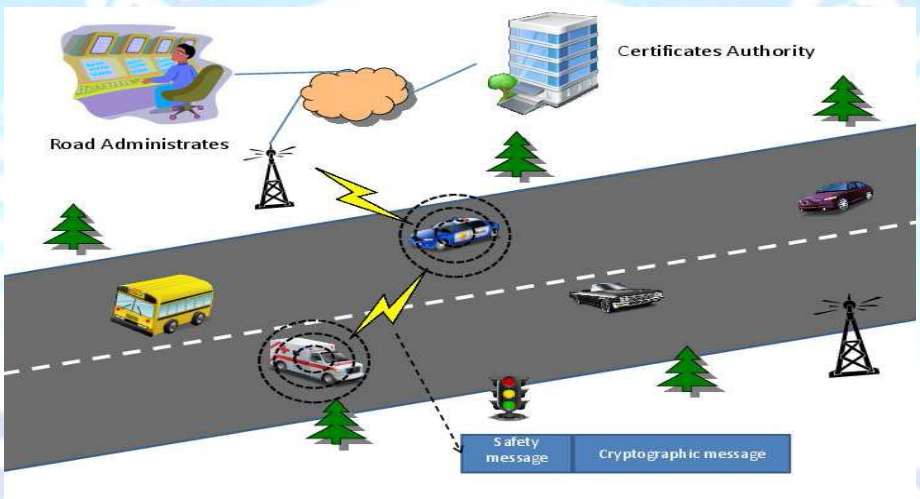

Figure 1. Overview of the secured structure of VANETs.

Usually authentication and digital certificates act as the major tools used to validate the identification of each communicating entity. The entity's certificate can be validated by checking its digital certificates. However, the promptness of validation would be much more important for VANETs when compared to conventional networks, because it is not unusual that every vehicle receives a large number of messages in a short time.

Moreover keeping connections live between different entities could be extremely hard to achieve, because of the high speed of moving vehicles as well as the increasing distance between these vehicles since they may move in different directions. Hence it is necessary to find an efficient scheme to expedite the certificate validation process.

In this work, a novel certificate validation scheme is proposed to adopt the concept of clustering from data mining technique.

1.2. K-Means Clustering Based Scheme for Certificate Authentication

In this work, to propose an accelerating certificate revocation status validating scheme for authentication in VANETs. The acceleration is caused by two aspects.

a. Introducing new elements in CRL and
b. Adopting k-means clustering algorithm with enhanced centroids selection

In virtue of the acceleration procedure, a successful validation could be achieved.

1.3. *K*-MEANS CLUSTERING

To ensure smooth transition of proposed centroids selection approach of *k*-means is unsupervised knowledge learning and partitioning algorithm for clustering *n* data points into *k* discrete clusters *C*, with the cluster C_j contains n_j data points [11]. Each cluster has a centroid, which represents a central vector used to assign different entities to that specific cluster. *k*-means picks an initial centroid randomly and the equation 1 determine the next cluster centroids.

$$L = \sum_{j=1}^{k} \sum_{i=1}^{n} \left\| x_i - \mu_j \right\|^2 \quad - (1)$$

Where:

x_i is a vector denoting the x_i th data point
μ_j is the centroid of data points in C_j
L is the distance for each data points to all centroids

The k-Means clustering algorithm [11] Algorithm 1:

Algorithm 1: *K* – Means Clustering Algorithm

Require: Input the number *k* of cluster centroids.
Ensure: Output *k* cluster
1: Get *k* = number of clusters
2: Get $X = (x_1, x_2, \ldots, x_3)$, x_i € R^d
3: **for** *j* = 1 to *k* **do**
4: select $\mu_1, \mu_2, \ldots, \mu_k$ randomly
5: **end for**
6: **for** *j* = 1 to *k* **do**
7: **for** *i* = 1 to *n* **do**
8: determine $\mu_j = \{\mu_j / max \sum_{j=1}^{k} || x_i - \mu_j ||^2\}$
9: **end for**
10: **end for**
11: Assign x_i to μ_j
12: After all data points have been assigned, recalculate the position of the centroids.
13: Repeat step 6 to 10 until all centroids are convergent

The centroids are considered as converged if their positions do not change any more after a number of iterations.

1.4. CENTROIDS SELECTION PROCEDURE

To choose the initial centroids and the reason behind it to use two new attributes in the CRL file. The algorithm is tailored and optimizes its performance on the two-dimensional vector space. The improvement is based on two aspects.

- The distance between newly discovered initial centroids and existing ones
- The distribution density of data points in some certain zones.

For the problem concerning the distance between current and previous centroids, the original *k*-means clustering algorithm selects the initial centroids randomly without considering their spread out placement. Therefore, this deficiency has an enormous potential to result in some centroids being too close to each other which may risk the clustering results. The situation is illustrated in Figure 2.

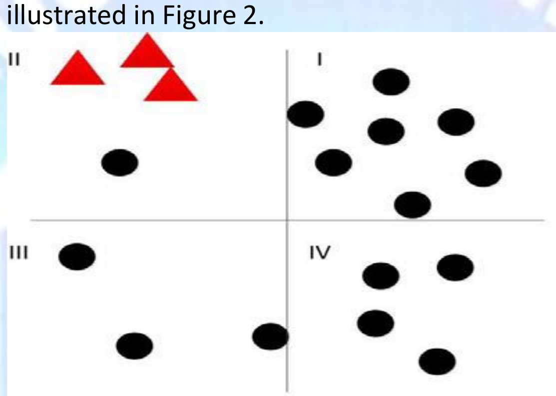

Figure 2(a). The three initial centroids (triangles) trapped in a local small zone

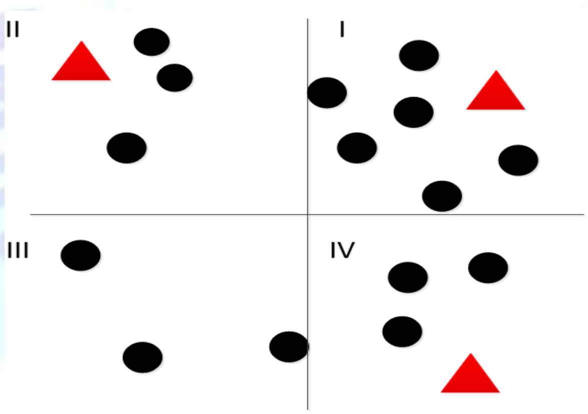

Figure 2(b). A better initial centroids selection and from that each centroid has a further distance between others.

The other issue is the density ("frequency") of data point distribution. In the vector space that contains data points, there are plenty of areas with varied density. Commonly, the probability that an area will contain the initial centroid is directly proportional to the density of that area. For instance, as illustrated in Figure 2 (a) and 2(b).

2. FORMAL DESCRIPTION OF THE ALGORITHM

In order to better understand the operation of the proposed algorithm, the following description is as follows:

Let X = (x_1, x_2, . . . x_n), $x_i \in R^d$ be the set of data points, *k* be number of clusters, and *w* be the group width metric.

$S = \{S_i | i = 1, 2, . . ., k^2\}$ is the set of segments that have been partitioned in a two-dimensional vector space. $F = \{fi | i = 1, 2, . . . , k^2\}$ be the frequency in S, $G = \{G_i | i = 1, 2, . . ., k\}$ be the section where the initial centroids should generated within, $\mu = \{\mu_i | i = 1, 2, . . ., k\}$ be initial centroids, and $P = \{p_i | i = 1, 2, . . ., k\}$ is the set of potential initial centroids.

$D = \{D_i | i = 1, 2, . . ., n\}$ denotes the distance metric for each iteration, and *dis(x,y)* is the function that calculates the distance between data point *x* and data point *y*.

t represents the current iteration step, *d* is the index number used to find the segment in the next iteration that requires an initial centroid to be chosen in, and $f(G_i)$ denotes the function that computes the frequency of G_i.

Finally, *m(X)* and *m(G)* are the grand mean of *X* and *G* respectively. The execution steps of the proposed algorithm is described as in algorithm 2.

Require: Input the number *k* of cluster centroids
Ensure: Output k cluster centroids locations
1: $S = \emptyset$, $F = \emptyset$, $G = \emptyset$, $\mu = \emptyset$, $P = \emptyset$, $D = \emptyset$
2: Calculate $w \leftarrow \{(\max(X) - \min(X))/k\}$
3: Divide the Vector space into k^2 group with w
4: Assign $S \leftarrow$ segments of vector space
5: $F \leftarrow f(S)$
6: Find $S_i = \{S_i \in S \& F_i = max(F)\}$
7: $G_1 \leftarrow S_i$
8: $\mu_1 \leftarrow m(S_i)$
9: $\mu = \mu \cup \mu 1$
10: Calculate $D \leftarrow dis(\mu_1, m(X))$
11: Set t = 2
12: **while** $t \leq k$ **do**
13: **if** $f(G_{t-1} = 0)$ **then**
14: Exit
15: **else**
16: Set $d = 0$
17: { comment : if q = 0 }
18: Select $S_j = \{S_j \in S \& f(S_j) = f(G_{t-1}) - d\}$
19: **if** $\neg\exists S_j € S, f(S_j) = f(G_{t-1}) - d$ **then**
20: $d = d + 1$
21: Go to step 18
22: **else**
23: { comment : if q = 1 }
24: **if** $\neg S_j \cdot (Sj \in S \cup (\forall S_i(S_i \in S \rightarrow S_i \neq S_j)))$
then
25: $G_t = S_j$
26: Assign $\mu_t \leftarrow m(G_t)$
27: $\mu = \mu \cup \mu_t$
28: Calculate $D \leftarrow dis(\mu_t, \mu_{t-1})$
29: **else**

{ comment : if $q > 1$ }
$\forall S_j = \{S_j \in S \& f(S_j) = f(G_t) - d\}$
Calculate $m(S_j)$
$P = P \cup m(S_j)$
Assign $D \leftarrow \max(dis(P, \mu_{t-1}))$
Select $p_i = \{pi \in P \& dis(p_i, \mu_{t-1}) = D\}$
Set $\mu_t = p_i$
$\mu = \mu \cup \mu_t$
end if
end if
end if
$t = t + 1$
end while
Exit

2.1. CERTIFICATE REVOCATION LIST PARTITIONING

Before vehicles and RSUs initialize a conversation with each other the four phases are needed to be performed during the revocation validation.

a. **Clustering**: In this phase, vehicles and RSU pre-process the latest CRL file using the two newly added attributes, issued date and credibility, combined with both *k*-Means clustering algorithm and the enhanced initial centroids selection scheme to efficiently cluster the revocation certificates entries. A sample illustration of the clustering result is shown in Figure 3.

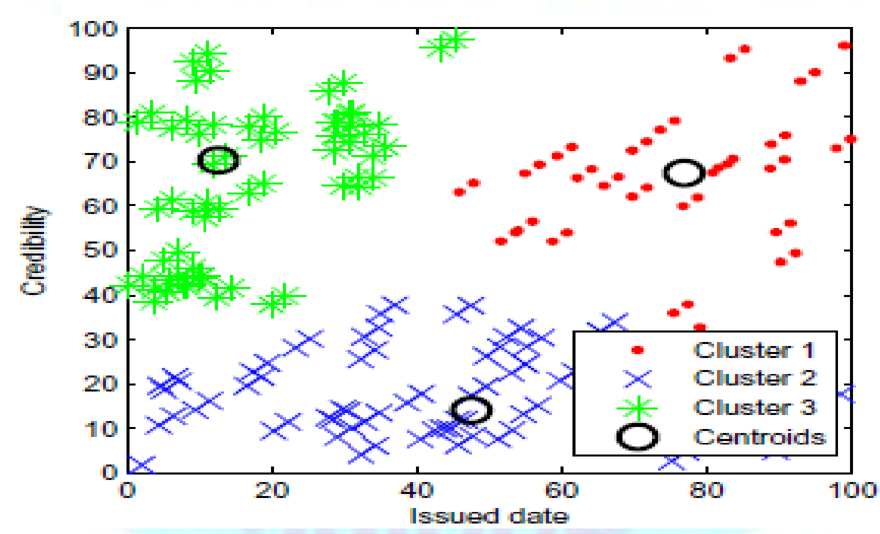

Figure 3. Clustering results using all entries in a CRL, where $N = 100$, $K = 3$.

b. ***Retrieving:*** Upon receiving a connection set up request message from other vehicles, receivers will check the certificates contained in that messages and extract all relevant information included in that certificate i.e. serial number, issue time, and credibility.

c. ***Localizing:*** Using the credibility and issued date, we can calculate the Euclidean Distance between the data point (i.e., new certificate) and all centroids to locate the closest cluster to join.

d. ***Verifying:*** In this phase, the new data points that join will check all neighbouring data points in the recently joined cluster for a match in terms of credibility and issue date. If a match is found, this means that its certificate has been revoked. Otherwise, this data point is not in the CRL and can therefore be trusted.

3. ANALYSIS OF SECURITY

In this work mainly focused on the attacks existing in vehicular communication systems and the major concern is the perpetration against messages during communications. The typical attack is illustrated in Figure 4.

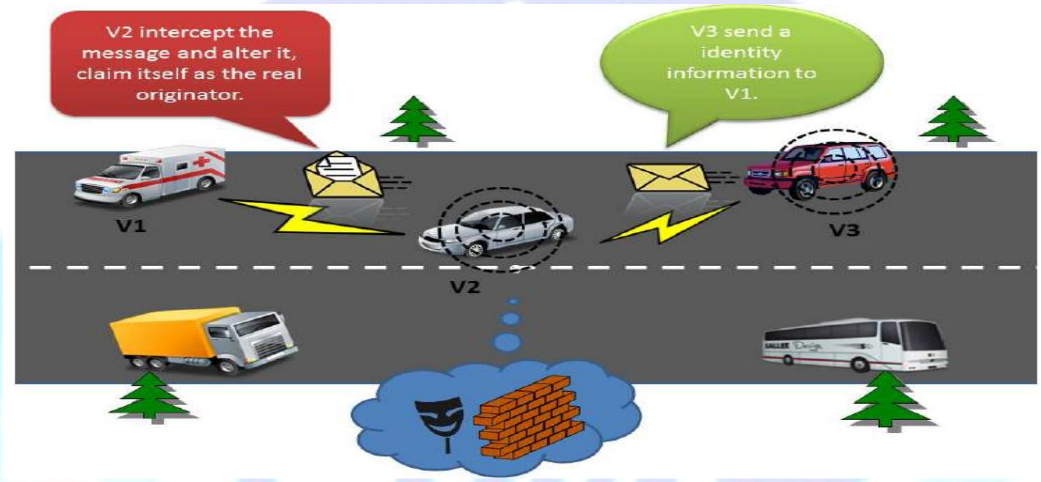

Figure 4. A typical attacks in VANETs

A . Correctness Proof

The correctness of the certificate revocation validation scheme is based on [12] and in this work, it is further improved and simplified. The symbols in table 2 depict the notations are used.

Table 2 . Notations

Symbol	*Notation*
$Cert_{A,B,*}$	*Certificate issued to B by A*
$CRL_{A,B,*}$	*CRL issued to B by A*
R_i	*The i-th RSU*
$PK_{A,*}, SK_{A,*}$	*The Public Key and the Secret Key of entitiy*
$\partial_{B,*}$	*A digital Signature signed by the entity B*
$Sign(SK_A, M)$	*Signature of message M by Digital Signature Algorithm (DSA) with secret key* SK_A
$Verify(PK_A, M, \partial_{A}, M)$	*Verifying of message M signed by signature algorithm with public Key* PK_A
V_j	*The j-th vehicle*
$RK_{A,B}$	*The returned re-signed key that issued to B by A*

1. Message Signature and Verification: Initially transmitting every message *M*, vehicle V_a signs *M* with the signature algorithm $\partial_{Vi,M} = Sign(SK_{va,M})$ to the destination. After receiving the message, vehicle V_b verifies the message $(M, \partial_{Va,M}, Cert_{CA,Va})$ by checking the revocation status of the certificate $(Cert_{CA,} V_a)$.
If $verify\ (PK_{Va}, M, \partial_{Va,M})$ is TRUE then vehicle (V_a) is ACCEPTED *(i.e.* $Cert_{CA,Va}$) is not revoked; otherwise message *'M'* is rejected.

After ownership of the valid certificate is shown by the sender and the verification of the revocation status is finished. This message can be accepted and guaranteed by using a signature algorithm when the originality of message is authenticated. Hence this prevents the Sybil and vulnerable attacks.

2. CRLs Issuing: The certificate authority (CA) issues a list to road side equipment $CRL_{CA,RSUa}$. This process of CA issuing to vehicle V_a as follows.

1. A hash number H=(h(m)) is calculated by *SHA-1*cryptographic hash function with the key that is the MAC address of the receiver RSU_a network interface controller.
2. CA is assigned a secret key $(SK_{CA} = h\ (m))$ and generate public key PK_{RSUa}
3. The digital signature of $\partial_{CA,RSUa}$ is generated by certificate authority with Digital Signature Algorithm (DSA) $\partial_{CA,RSUa} = Sign(SKCA, PK_{RSUa})$.
4. Certificate authority successfully delivers PK_{CA}, PK_{RSUa} and $CRL_{CA,RSUa} = (PK_{RSUa}, \partial_{CA,RSUa})$ to road side unit. The mapping among the RSU_a and $CRL_{CA,\ RSUa}$ then the certificate is verified then the certificate is storing by RSU_a.
5. RSU_a verifiy its own *CRL*. $CRL_{CA,RSUa}$ using $verify(PK_{CA}, PK_{RSUa}, \partial_{CA,RSUa})$.

Since digital signature algorithm applied, the road side unit's $CRL_{CA,RSUa}$ are ensured to be original. The certificate authority associates during the issue of digital certificate, MAC address of the receiver such as RSU_a or V_a. It restricts the any specified digital signature. This prevents masquerading from pretending to be other legitimate nodes since the MAC address of masqueraders from pretending to be other legitimate nodes since the MAC address of masqueraders cannot be identical to the RSU_a or V_a.

3. *Certificate Re-signing:* Vechile V_a passes by an RSU_a and sends its own certificate $Cert_{CA,Va,T}$ to RSU_a designed periodically by their certificates. The re-signed signature has time stamps. Only the Valid certificates can get re-signed simply by sending a request to RSU when passing by it. If the certificate is not revoked, RSU_a timestamps the certificate to denote it is valid, and returns it to the V_a. Otherwise the RSU rejects to re-sign it. The signature of the validity of certificate process within a given timestamp 'T' as follows:

1. V_a sends certificate to RSU_a, then RSU_a generates a re-signature key corresponding to the signal certificate $Cert_{CA\ ,\ Va\ ,\ T'\ *}$, *Where* $T' > T$.
2. RSU_a broadcasts $(M,\ Cert_{CA,RSUa})$ remain periodically every incoming vehicle when entering the area covered by RSU_a.
3. When the V_a receives $(Cert_{CA,\ RSUa})$ then it sends the request message $t_{stamp}(Cert_{CA,Va,T})$ to RSU_a.
4. When the RSU_a receives the message, if t_{stamp} is 'fresh' (valid period) and $Cert_{CA,Va,T}$ is not revoked. RSU_a sends the re-signature key $RK_{RSUa,Va}$, and $t_{stamp}\ Cert_{CA,Va,}\ T$ back to V_a. After the RSU_a records current time T' and certificate $<T'\ t_{stamp}, Cert_{CA,Va,} T'>$ is created.
5. The V_a checks the resigning key $RK_{RSUa,} V_a$ for the presence of the $t_{stamp}\ Cert_{CA}, V_{a,} T'$ with it.

A malicious vehicle may attempt to generate a certificate with invalid identity to prevent itself from being tracked by the certificate authority. Since the RSU_a signed the message via $Sign(SK_{Va}, M)$, the vehicle cannot forge the certificate due to other vehicles SK_{va} being confidential.

4. ***Certificate Revocation:*** When the vehicle V_a is compromised, its certificate is added as an entry in the CRL. In order to perform the following steps are followed by the vehicle.

1. The certificate authority sends the information of the revoked vehicle certificate $Cert_{CA,Va,T}$ to all roadside units.
2. When received $Cert_{CA, Va,T}$ each road side unit the RSU_a adds the related information to its local CRL. Consequently, the revoked $Cert_{CA,Va,T}$ would no longer able to request re-signing the certificate from RSU's. All RSU_a will sending back a confirmation message $(M, SK_{RSUa}, \partial_{CA,Va})$ to the certificate authority. The RSU_a again then determine which cluster of CRL the new revoked certificate $Cert_{CA,Va,T}$ will add based on the revoked certificates credibility and issued date.
3. After receiving the conformation from all RSUs, the certificate authority adds the information of the revoked certificate to $CRL_{CA,V}$, this is shared to all vehicles later on. At the same time, each road side unit RSU_a will broadcast the $(M,\partial_{RSUa,Vb})$ *to all vehicles* V_b, $V_b \neq V_a$ that are within the covered area.
4. When vehicle V_b receives $Cert_{CA,}V_a,T$ it is added the revoked vehicle certificate to local current $CRL_{CA,}V_b$ and determine which clusters the new revoked certificate will be added and it's time to update the next CRL.

The revoked certificate privacy could be preserved. As any anonymous channel that is secure from other vehicles can be used to communicate private information between RSUs to the CAs, and each vehicle could have the latest certificate revocation update from RSUs as long as they are within the RSU coverage area.

The RSU_a could distribute revoked certificate message $Cert_{CA,Va,T}$ using moving vehicles that are ongoing within its covered area in an epidemic manner. At first, road side units broadcast revoked certificate message $(M,Cert_{CA,Va,T})$ and any V_a receiving $(M,Cert_{CA,Va,T})$ is considered as infected. Afterwards, each vehicle continuously infects all vehicles it passes by. Using the steps mentioned above, revoked certificate message $(Cert_{CA,Va,T})$ distribution can be achieved.

REFERENCES

[1]. A. Boukerche, Ed., *Algorithms and Protocols for Wireless, Mobile Ad Hoc Networks*, 1st ed. Wiley-IEEE Press, Nov. 2008.

[2]. A.Boukerche, *Algorithms and protocols for wireless sensor networks*.Wiley, 2009.

[3]. M. Raya, P. Papadimitratos, and J. Hubaux, "SECURING VEHICULAR COMMUNICATIONS," *Wireless Communications, IEEE*, vol. 13, no. 5, pp. 8–15, Oct. 2006.

[4]. Florian Doetzer, Timo Kosch, and Markus Strassberger, "Classification for Traffic Related Inter-vehicle messaging", in proceedings of the 5th IEEE International Conference on ITS Telecommunications France 2005.

[5]. http://en.wikipedia.org/wiki/intelligent_transport_systems

[6]. Vehicle Safety communication Project Task 3 Final Report
http://www.its.dot.gov

[7] J. Kenney, "Dedicated Short-Range communications (DSRC) standards in the united states," *Proceedings of the IEEE*, vol. 99, no. 7, pp. 1162–1182, Jul. 2011.

[8] J. J. Haas, Y. Hu, and K. P. Laberteaux, "Design and analysis of a lightweight certificate revocation mechanism for VANET," in *Proceedings of the sixth ACM international workshop on VehiculAr InterNETworking*, ser. VANET '09. New York, NY, USA: ACM, 2009, pp. 89–98.

[9]. ITU-T,Series X: Data Networks, Open System Communications and Security. http://www.itu.int/ITU-T/ipr/

[10]. V2V Communications NPRM Fact Sheet, US Department of Transportation, www.safercar.gov/v2v/

[11]. T.Kanungo, D.M. Mount, N.S. Netanyahu, C.Piatko, R.Silverman, and A.Y.Wu, "The Analysis of a simple *K*-means Clustering Algorithm", Technical Report CAR-TR-937,Center for Automation Research, Univ. of Maryland, college Park, Jan. 2000.

[12]. Vehicle Safety Communication Project (VSC), "Task 3 Final Report", identify Intelligent Vehicle Safety Applications", US Dept. of Transportation, Technical Report, 2005.