Cache Pollution Prevention Mechanism Based on Cache Partition in V-NDN

1st Jie Zhou
School of Communication and Information Engineering
Chongqing University of Posts and Telecommunications
Chongqing 400065, China
S180101160@stu.cqupt.edu.cn

2nd Jiangtao Luo
Electronic Information and Networking Research Institute
Chongqing University of Posts and Telecommunications
Chongqing 400065, China
Luojt@cqupt.edu.cn

3rd Lianglang Deng
School of Communication and Information Engineering
Chongqing University of Posts and Telecommunications
Chongqing 400065, China
S180101147@stu.cqupt.edu.cn

4th Junxia Wang
School of Communication and Information Engineering
Chongqing University of Posts and Telecommunications
Chongqing 400065, China
D170101010@stu.cqupt.edu.cn

***Abstract*—The information-centric networking, which aims to solve the demand for distributing a large amount of content on the Internet, has proved to be a promising example for various network solutions, such as the Vehicular ad-hoc network (VANET). However, some problems are introduced when the named data networking is combined with V-NDN, such as the cache pollution. In order to solve the cache pollution attack, we propose a mechanism based on cache partition, which divides the cache of nodes into two parts and stores the content of different popularity respectively. We monitor the interest packets received by each node and get the corresponding popularity of each content. According to the popularity of the content, the content is stored in the corresponding cache. In addition, when the popularity of the content changes, we add the name of the content to the monitoring list to determine whether it is an attack content. This paper simulates the cache partition mechanism under different request frequencies and different forwarding strategies. The experimental results show that the average hit rate of node cache can be increased by 14% and the user request delay can be reduced by 30% when the node is attacked. At the same time, the number of Interest packets requested by normal users in the whole network has also been greatly reduced, which greatly reduces the traffic within the network. Experiments show that the cache partition mechanism can effectively resist the attack of cache pollution.**

***Index Terms*—Vehicular ad-hoc network, information-centric networking, cache partition, cache pollution**

I. Introduction

Due to the large scale of the vehicle network and the extremely uneven network density, the network state is more easily affected by time, space and other factors. The rapid movement of nodes will cause the network topology to change more frequently [1]. For the Internet of Vehicles, the location of the terminal is constantly changing, which makes it very difficult to manage the mobility of the Internet of Vehicles under the IP network. Moreover, the constant connectivity of the link cannot be guaranteed in the vehicle network, resulting in the recalculation of routes and remote connections frequent re-establishment prevents the IP network architecture from functioning well on the Internet of Vehicles.

Information-centric networking (ICN) is a promising network solution for the Internet of Vehicles scenario [2]. It proposes different semantics for data packets at the network layer, and extensively uses intra-network caching, which is mainly used to improve the efficiency of large-scale content distribution. In particular, ICN uses content names instead of IP addresses for routing and can obtain copies of content from any source. It inherently supports multicast and mobile communications. This makes it suitable for VANET. Compared with the IP network, the information-centric networking is more concerned about the data itself than it's physical storage location, which has improved network security, mobility, and scalability [3]. The communication of the information-centric networking is driven by the receiving end, and consumers send interest packets to request the corresponding data packets. This communication model driven by the receiving end can better meet the needs of VANET. As one of the information-centric networking examples, named data networking (NDN) mainly has the following characteristics:

- Taking named content as the center, routing based on content names rather than ip addresses, and focusing on the content itself, not the location of the content [4].
- Introduceing a cache mechanism in the network, the router can cache the content, and the content can be replied from multiple places [5].

NDN contains two types of packet: interest packet and data packet. Consumers request the required content by sending interest packet. The interest packet contains the name of the requested content. The content requested by the consumer is returned through the data packet. The NDN router mainly maintains three tables: content store(CS), pending interest table(PIT), forwarding information base(FIB).

- CS: The cache component in the NDN, which caches

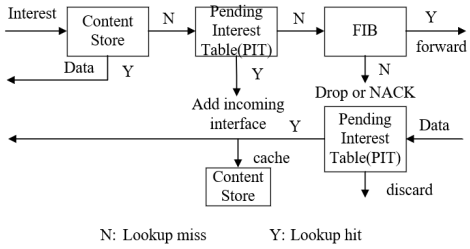

Fig. 1. Forwarding process of nodes in named data networking

the data packet passing through the node to satisfy subsequent requests.

- PIT: Recorded the name of the forwarded Interest packet and the interface of interest packet arrives. In this way, the subsequent data packet can be correctly forwarded to the consumer.
- FIB: Used to find the appropriate forwarding interface for Interest packets. Similar to the forwarding table in the IP router, the request is sent to the destination node. The difference is that FIB in NDN can forward requests to multiple nodes.

The NDN forwarding process is shown in Fig. 1.

When the router receives the interest packet, it checks the CS. If there is the requested content in the CS, it directly returns the corresponding data packet and discards the interest packet. If it is not in the CS, it checks the PIT. If the corresponding name in the PIT is found, the interface of the interest packet is added to the entry, and the interest packet will discard. If not, a new PIT entry is created. Finally, according to the longest matching principle, the FIB forward the interest packet to the next hop.

When the router receives the data packet, it checks whether there is a corresponding entry in the PIT. If not, it indicates that it is an unsolicited data packet and discards it. If it exists, it forwards the data packet to all interfaces corresponding to the entry, and according to the cache policy determines whether to cache the packet.

Although the combination of the information-centric networking and the Internet of Vehicles has achieved encouraging initial results, the application of NDN to VANET still needs more in-depth research. Current Internet security is based on creating a secure channel between end-users to protect content transmission between end-users. However, this principle may have some disadvantages. For example, if the session expires, the exchanged security information cannot be used again. In contrast, NDN provides security for the content itself by embedding the signature and some auxiliary related information in each data packet [6]. By using content-based security, NDN solves many problems in IP-based networks [7]. For example, distributed denial of service (DDoS) attacks can benefit from knowing IP addresses, but NDN reduces the efficiency of DoS attacks because nodes are not directly addressable. However, when met with VANET, there are still other security and privacy challenges in NDN [8]. For example, the proliferation of unnecessary content requests, the widespread of forged content, and the low-popular content flooding the node cache space all pose a serious threat to VANET. A lot of malicious requests may deplete content producers and infrastructure resources. The spread of forged content may pollute cache in the network, resulting in legal content requests cannot be processed and effective content is difficult to obtain. Low-popular content flooding the cache space will reduce the cache hits of normal user, resulting in requests cannot be satisfied in intermediate nodes, so that the named data VANET loses the advantage of universal caching in the network. The main contributions of this article are as follows.

- A defense mechanism based on cache partition is proposed for cache pollution attacks on the Internet of Vehicles.
- A popularity monitoring algorithm is proposed to monitor the change of popularity. Adding the content with abnormal popularity to the monitoring list to check whether it is an attack content.

According to the changes of content requests in the network, part of the content is monitored. When the node caches the content, it will process the data according to the monitoring situation, so as to achieve the purpose of preventing cache pollution.

The rest of this article is organized as follows. In the second part, we reviewed related work. The third part introduces our design in detail. Next, the simulation results are in the fourth part. The conclusion is shown in the fifth part.

II. Related Work

In order to solve the cache pollution attacks, Lin et al. [9] clustered interest packets to detect and prevent cache pollution. The scheme can distinguish whether interest packet requests follow a Zipf-like distribution for accurate detection. Once an attack is detected, the attack table will be updated to record abnormal requests. Although such requests are still being forwarded, the corresponding content blocks are not cached, thereby reducing cache pollution. However, this solution does not consider the low popularity of normal user requests, so some low popularity content requests are considered as attack requests, resulting in false judgments. Also, when the number of attackers is sufficient, the attacker does not need to request a large amount of data, so as not to destroy the Zipf distribution. Aiming at the detection of cache contamination attacks, Guo et al. [10] proposed an anti-pollution algorithm (APA) based on path diversity. The algorithm believes that a genuinely popular content interest request packet should arrive through multiple path. If no such feature is observed at the node, it is considered that the interest packet may come from the attacker and should be discarded directly. Conti et al. [11] proposed a cache replacement strategy, CacheShield, to resist cache pollution. The algorithm consists of two parts: a probability function that calculates the cache probability and a container for storing vectors. When the NDN intermediate router receives a data packet, the algorithm uses a probability function to determine the probability of caching the content. This caching strategy

scheme can be used to avoid caching less popular content, which has a certain defense against cache pollution, but when there are too many attackers, the effect will decrease.

III. Defense mechanism of cache partition

In this section, we introduce the basic problem to be solved. After analyzing the problem, a defense mechanism based on cache partition is proposed. Finally, we discussed how to implement the algorithm.

A. Cache pollution in the VANET

In-network caching in named data networks can reduce traffic in the network. Mobile nodes can bring content closer to potential consumers, thereby reducing average data retrieval time and overall network traffic, and accelerating the spread of key information on the Internet of Vehicles [12]. In fact, in-network caching can bring more advantages than just download time and network traffic. For example, the contents of the vehicle cache can be moved to different areas to improve the spread of special event notifications (such as car accidents or traffic jams).

The purpose of cache pollution is to make the cache in the network almost invalid, and let network nodes cache some rarely used content. In the end, most of the requests of normal users cannot be satisfied at the intermediate node. Finally, these network requests are always transmitted to the content source. network traffic also increased [13]. Cache pollution is different from content poison. The latter's attack method is to allow forged or invalid content to spread in the network, thereby poisoning the cache. These contents are illegal, and can be detected by performing signature verification on each data packet. Cache pollution use legitimate content to attack, these data packets can not be identified by signature verification. In cache pollution, an attacker requests content with extremely low popularity through a large amount, so that the content will be stuck in the cache of the intermediate node. Due to the limited capacity of the node, the really popular content will be removed from the cache. When normal user request content, the request can only be forwarded to other nodes, thus achieving the purpose of the attack.

B. Cache partition

In response to cache pollution attacks, Lin et al. [9] use clustering techniques to detect and defend against cache pollution. When an attack is detected, the network request is classified by clustering technology. However, this method does not consider the situation where normal users request low-prevalence content, and some normal users will be mistaken as attackers, damaging the interests of legitimate users.

This article considers the situation of normal users requesting low popularity content, and designs a defense mechanism for cache partitioning. This method adds a popularity list to each node to record the content of high popularity stored in node and monitor list to record the name of abnormal popularity content when the content of the node monitoring changes in popularity. We divide the CS of NDN node into two CS, which save the content of high popularity and low popularity respectively. Finally,we set up a cache replacement strategy for these two cache spaces to perform cache replacement. The capacity of the two different caches Can be calculated by the following formula.

Algorithm 1 The process of inserting data into CS

Input: $data, InterestStatistics$
$popularity \leftarrow InterestStatistics$
$averagePopularity \leftarrow popularity$
$dataName \leftarrow data$
if $popularity.dataName < averagePopularity$ **then**
  $insert\ it\ into\ unpopCS$
  if $popularityList.find(dataName)$ **then**
    $popularityList.erase(dataName)$
  end if
else
  if $popularityList.find(dataName)$ **then**
    $insert\ it\ into\ popCS$
    if $popCS.size \geq popCapacity$ **then**
      $According\ to\ the\ corresponding\ cache\ policy, remove\ the\ data$
    end if
  else
    $generate\ p_i$
    if $p_i < P$ **then**
      $drop\ and\ return$
    else
      $insert\ it\ into\ unpopCS$
      if $unpopCS.size \geq unpopCapacity$ **then**
        $According\ to\ the\ corresponding\ cache\ policy, remove\ the\ data$
      end if
      $monitorName \leftarrow dataName$
      $call\ monitor\ function$
    end if
  end if
end if

$$popCapacity = maxCapacity * \alpha \tag{1}$$

$$unpopCapacity = maxCapacity * (1 - \alpha) \tag{2}$$

Among them, $maxCapacity$ is the maximum capacity of the node cache, $popCapacity$ is the capacity to store the cache of high popularity, and $unpopCapacity$ is the capacity to store the cache of low popularity. The cache division is shown in Algorithm 1.

There are two ways to express the popularity of content objects.One of them is the number of times each content object is accessed in a period of time, the other is based on the probability of the content object being accessed [14]. According to Borel's theorem of large numbers, when the statistical time is long enough, the probability that a certain content is accessed can be expressed by the frequency with

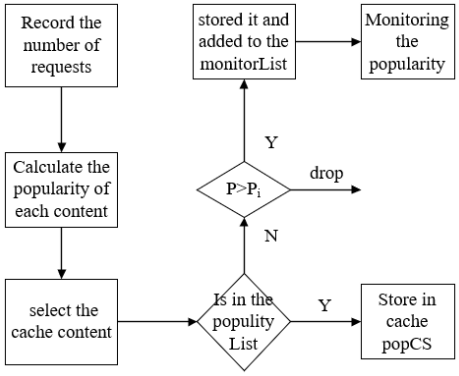

Fig. 2. The process of a node caches data

which it is accessed. It can be seen that the two definitions have the same effect, and this article uses the latter definition.

Definition of content popularity: set within the time interval t, the router receives N packets of requests, a total of M different requests. Among them, the number of occurrences of the i-th request is f_i. Then the popularity of the i-th request is:

$$pop_i = f_i/N \tag{3}$$

The data flow of network node cache is shown in Fig. 2.

Step 1: The node counts the request packets received within the time interval t.

Step 2: According to the definition of popularity above, calculate the popularity of each request packet. According to the average value of all data popularity, the data lower than the average value of popularity is regarded as low popularity content; otherwise, it is high popularity content.

$$averagePopularity = \Sigma_i^N pop_i/N \tag{4}$$

Step 3: Select the data to be cached according to the overall cache policy.

Step 4: Judge whether the data selected is high popularity content in step 3. If it is not, check whether it is in the popularity list. If it is, remove it. If it is high popularity content, check whether its name is in the popularity list. If it is, the data will be directly stored in the cache that stores high popularity content; if it is not, the probability will be used to choose whether to cache the data. If we choose to cache the data, add the name of the content to the monitoring list, and enter step 5.

Step 5: Monitor the contents in the monitoring list for a period of time. If the name is still popular content after time t, it will be removed from the monitoring list and added to the popularity list; otherwise, it will be removed from the monitoring list. The monitoring process is shown in Algorithm 2.

IV. EVALUATION

We implement our design in ndnSIM [15] and compare the mechanism based on cache partitioning with the original NDN mechanism. For different forwarding strategies and different request frequencies of legitimate users, simulation comparisons are made to measure the cache hit ratio and request delay of user requests in different situations. The relevant parameters are shown in Table 1.

Algorithm 2 Monitor the changing names of popularity

Input: $popularity, montiorNames, popularityList$

$curTime \leftarrow Currentsimulationtime$
$maxMonitorTime \leftarrow Maximum\ monitoring\ time$
$minMonitorTime \leftarrow Mimimum\ monitoring\ time$
$averagePopularity \leftarrow popularity$
for each $name \in montiorNames$ **do**
  $hasMonitorTime = curTime - name.recordTime$
  if $hasMonitorTime \geq maxMonitorTime$ **then**
    $montiorNames.erase(name)$
  end if
  if $hasMonitorTime \geq minMonitorTime$ **then**
    if $popularity.name > averagePopularity$ **then**
      $popularityList.insert(name)$
      $montiorNames.erase(name)$
    end if
  end if
end for

TABLE I
SIMULATION PARAMETERS

Number of normal users	1
Number of attack users	2
Total number of nodes	7
CS size	40
Cache policy	LRU
P_i	0.2
α	0.8
Moving speed	10m/s-15m/s
Attack frequency	16 interest per second
Monitoring time	10s
Simulation time	40s
Attack time	10s-40s

The simulation network topology is shown in Fig. 3. The initial position of all vehicles are randomly generated within a certain range. After the start of the simulation, the vehicle moves at a certain speed. The simulation time lasts 40 seconds. Before 10 seconds, only normal users requested content. After 10 seconds, the attacker attacks until the end of the simulation.

A. Results and Discussion with Different forwarding strategies

In the VANET environment, the mobility of vehicle nodes is large, and the intermittent connection between nodes is prone

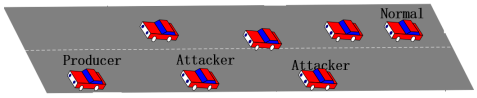

Fig. 3. Network topology

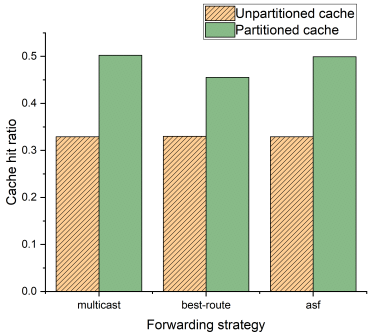

Fig. 4. Cache hit ratio under different forwarding strategies

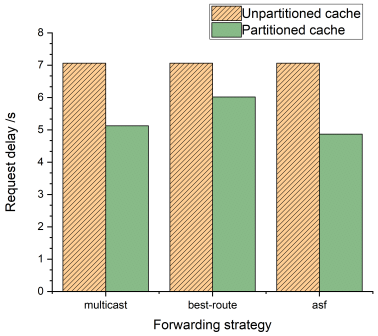

Fig. 5. Average request delay under different forwarding strategies

to occur. The packet loss in the network is more serious, and overtime retransmissions occur frequently. Therefore, the cache hit ratio of the nodes are generally low, and the user request has a large delay.

Fig. 4 shows the cache hit ratio of user request interest packet under different forwarding strategies. It can be seen from the figure that the cache hit ratio is only about 0.33. After using the cache partition mechanism that we proposed, the user's request cache hit ratio has increased by at least about 13%. Among the three forwarding strategies, the cache hit ratio of Multicast and ASF forwarding strategies is better than that of best-route. The reason is that the Best Route is based on the route cost. It forwards the interest packet to upstream with the lowest routing cost, rather than some upstream nodes. Due to the greater mobility of nodes, a node may no longer be within its communication range when a user makes a request, resulting in packet loss and a decrease in cache hit ratio.

Fig. 5 shows the average delay of user requests under different forwarding strategies. Under the three forwarding strategies, the average delay of user requests is lower than the original NDN caching mechanism. Among them, the request delay with the Multicast and ASF forwarding strategies is the lowest, and the average request delay is reduced by about 2s compared with the case where the cache division mechanism is not used. The Best Route forwarding strategy is next, and the average request delay is reduced by about 1s. Due to the high mobility of nodes on the Internet of Vehicles and the fast change of the network topology, forwarding requests to a single node at the lowest routing cost becomes unreliable and may cause timeout retransmissions due to packet loss.

B. Results and Discussion with Different request frequencies

Fig. 6 shows the request cache hit ratio of normal users under different request frequencies. It can be seen from Fig. 6 that under different request frequencies, the cache hit ratio based on the cache partitioning mechanism is always higher than that without the cache partitioning. With the increase of the frequency of normal user requests, the cache hit ratio of nodes increases gradually. The main reason is that the attacker's attack frequency is fixed. As the normal user's request frequency increases, the attack effect will gradually

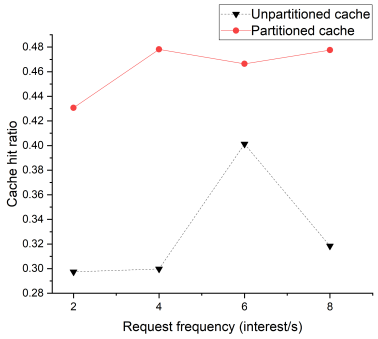

Fig. 6. Cache hit ratio under different request frequencies

decrease, so the user's cache hit ratio will gradually increase.

Fig. 7 compares the average request delay of normal users at different request frequencies. Similarly, since the attacker's attack frequency is fixed, as the frequency of normal user requests increases, the attack effect decreases. The user's request delay is declining. It can be seen from the figure that when the difference between the frequency of normal user requests and the frequency of attacks is greater, the effect of cache partitioning is more obvious. As the frequency of user requests increases, the average request delay of normal users gradually decreases. But the mechanism based on cache division is always better than the case without division.

C. Results and Discussion with the number of interest packet

Finally, under different forwarding strategies and different request frequencies of normal users, the number of interest packets sent and forwarded by normal users is counted, as shown in Fig. 8 and Fig. 9. It can be seen from the figure that, without using the cache partition mechanism, the interest packets that normal users need to send are far larger than the cache partitioning mechanism. This is because, in the absence of cache partitioning mechanism, the cache hit ratio of the user is low, most of the user's requests will not be satisfied by the intermediate node, resulting in the user's requests will be forwarded to the content provider. Also, the mobility of the nodes on the Internet of vehicles, intermittent connection

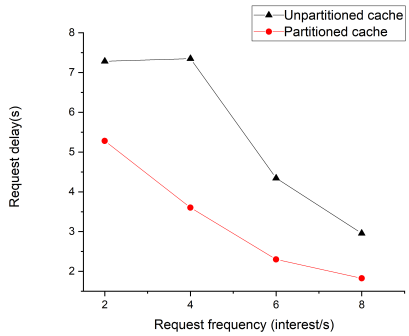

Fig. 7. Average request delay under different request frequencies

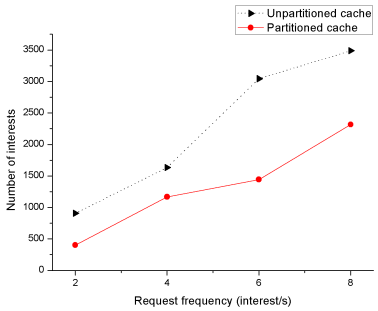

Fig. 8. Number of interest packets under different request frequencies

between nodes and other characteristics, exacerbated the case of packet loss, retransmission, greatly increased the number of interest packet requests and forwarding.

V. Conclusion

In this paper, a defense mechanism based on cache partition is proposed for cache pollution attacks in the Internet of Vehicles. The network node cache is divided into two parts to cache content with different popularity. According to the changes of content requests in the network, part of the content is monitored. When the node caches the content, it will process the data according to the monitoring situation, so as to achieve the purpose of preventing cache pollution. Through simulation experiments on different forwarding strategies and different user request frequencies, the results show that the mechanism can effectively resist cache pollution attacks, improve the request cache hit ratio of normal users, and reduce the request delay.The cache replacement strategy of each part needs to be further explored in the future work. In addition, the definition of content popularity needs to be optimized to improve the resistance to cache pollution attacks.

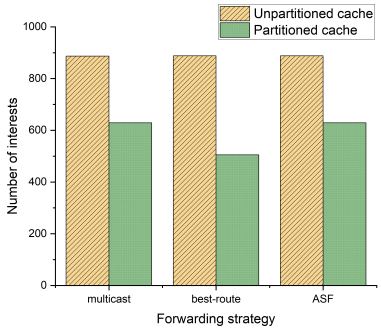

Fig. 9. Number of interest packets under different forwarding strategies

References

[1] Vijayalakshmi, V, Sathya, M, Saranya, S, & Selvaroopini, C. (2014). Survey on various mechanisms for secure and efficient VANET communication. International Conference on Information Communication & Embedded Systems. IEEE.

[2] Ahlgren, B. , Dannewitz, C. , Imbrenda, C. , Kutscher, D. , & Ohlman, B. . (2012). A survey of information-centric networking. IEEE Communications Magazine, 50(7), 26—36.

[3] Vasilakos, A. V. , Li, Z. , Simon, G. , & You, W. . (2015). Information centric network: research challenges and opportunities. Journal of Network & Computer Applications, 52(jun.), 1-10.

[4] Grigoreva, E. , Machuca, C. M. , & Kellerer, W. . (2016). Optical backhaul network planning for DSRC-based Public Intelligent Transportation System: A case study. 2016 18th International Conference on Transparent Optical Networks (ICTON). IEEE.

[5] Jiang, X. , Bi, J. , Nan, G. , & Li, Z. . (2015). A survey on information-centric networking: rationales, designs and debates. China Communications, 12(7), 1-12.

[6] Tourani, R. , Mick, T. , Misra, S. , & Panwar, G. . (2016). Security, privacy, and access control in information-centric networking: a survey. IEEE Communications Surveys & Tutorials, PP(99), 1-1.

[7] Khelifi, H. , Luo, S. , Nour, B. , & Shah, S. C. . (2018). Security and privacy issues in vehicular named data networks: an overview. Mobile Information Systems, 2018(PT.3), 5672154.1-5672154.11.

[8] Bernardini, C. , Asghar, M. R. , & Crispo, B. . (2017). Security and privacy in vehicular communications: challenges and opportunities. Vehicular Communications, S2214209617300803.

[9] Yao, L. , Fan, Z. , Deng, J. , Fan, X. , & Wu, G. . (2018). Detection and defense of cache pollution attacks using clustering in named data networks. IEEE Transactions on Dependable & Secure Computing, 1-1.

[10] Guo, H. , Wang, X. , Chang, K. , & Tian, Y. . (2016). Exploiting path diversity for thwarting pollution attacks in named data networking. IEEE Transactions on Information Forensics & Security, 11(9), 2077-2090.

[11] Conti, M. , Gasti, P. , & Teoli, M. . (2013). A lightweight mechanism for detection of cache pollution attacks in named data networking. Computer networks, 57(16), 3178-3191.

[12] H. Al-Omaisi, E. A. Sundararajan and N. F. Abdullah, "Towards VANET-NDN: A Framework for an Efficient Data Dissemination Design Scheme," 2019 International Conference on Electrical Engineering and Informatics (ICEEI), Bandung, Indonesia, 2019, pp. 412-417, doi: 10.1109/ICEEI47359.2019.8988843.

[13] D. Kim, J. Bi, A. V. Vasilakos and I. Yeom, "Security of Cached Content in NDN," in IEEE Transactions on Information Forensics and Security, vol. 12, no. 12, pp. 2933-2944, Dec. 2017, doi: 10.1109/TIFS.2017.2725229.

[14] Fliam, R. , Flanagan, K. C. , Broome, G. A. , Burgess, J. , & Commeau, G. . (0). Content distribution network supporting popularity-based caching. US.

[15] Afanasyev, Alexander & Moiseenko, Ilya & Zhang, Lixia. (2012). ndnSIM: ndn simulator for NS-3.