A Probability Verification Strategy Based on Credibility of Router in Content Centric Networking

Yi Zhu, Haohao Kang, Jia Shi and Jianlong Yang

***Abstract*—As the representative architecture of the next generation network, Content Centric Networking (CCN) adopts a content-oriented security design by using the signature and verification mechanism, thus binding security to content itself. However, because of the huge computation overhead this basic design may cause, it cannot be exploited effectively. As a result, the content pollution has been a serious security threat in CCN. Aiming at this problem, this paper evaluates the validity of received data traffic and defines it as the credibility of router, then proposes a probability verification strategy based on router's credibility (named as PVSRC in short). The verification probability of PVSRC is co-determined by the credibility and the maximum verification capacity of router. To assess the strategy performance, the theoretical approximation expressions are derived for valid content hit probability,verification overhead and packet drop rate of user. The numerical results show that PVSRC can guarantee high percentage of valid contents in network while reducing the workload of router. Compared with the strategy of verification on hit, it provides better robust protection against content pollution attack.**

***Index Terms*—Content centric networking, Verification strategy, Credibility of router, Probability verification, Content pollution.**

I. Introduction

IN the big data age, contents and content services have gradually become the body of network applications. The traditional host-to-host IP architecture cannot meet current users' requirements due to its inherent flaws, such as traffic redundancy, bandwidth competition, poor mobility and etc. To solve these problems, several future network architectures like DONA [1] and PSIRP [2] and the most competitive Content Centric Networking (CCN) [3] are proposed. Caching technology is the key feature of CCN, each CCN router owns a large-capacity Content Store (CS) to cache data packets passed by. Combining with naming routing, users can obtain data from nearby routers without visiting source server. This novel architecture provides powerful content delivery service, effectively reduces data fetching time, alleviates traffic congestion and avoids the negative impact of poor link connection.

Manuscript received February 07, 2018; revised June 10, 2018.

Yi Zhu (corresponding author) is with the school of Computer Science and Communication Engineering, Jiangsu University, Zhen Jiang, China, e-mail: zhuyi@ujs.edu.cn

Haohao Kang is with the school of Computer Science and Communication Engineering, Jiangsu University, Zhen Jiang, China, e-mail: 2211608006@stmail.ujs.edu.cn

Jia Shi is with the Department of Student Affairs, Jiangsu University, Zhen Jiang, China, e-mail: 253442502@qq.com

Jianlong Yang is with Jingjiang College of Jiangsu University, Zhen Jiang, China, e-mail:ArcYang@protonmail.ch

In the aspect of security, CCN also proposes a brand-new design. Instead of link protection, the security system of CCN is built based on the signature and verification mechanism. In CCN, a data packet contains not only content but also some necessary information required for content verification, including producer ID, public key or the public key locater [4]. Using these information, the validity of data packet can be checked. Theoretically, this mechanism can provide an effective way to avoid content pollution attack. However, as mentioned in the paper [5], current verification mechanism of CCN cannot normally operate in practice, it faces two challenges. The first one is signature verification overhead, which is also mentioned in the paper [6]. Whatever encrypt algorithm the router employs, it will confront huge calculation pressure if data traffic is considerable. Gasti and Paolo [5] points out that despite using optimized RSA algorithm, the verification limitation for a router with Intel Core 2 Duo 2.53GHz CPU is about 150Mbps when the size of signed packet is 1.5KB. The second challenge is trust management. Though each data packet contains a filed indicating public key or public key locator, CCN needs an extra trust management to check whether a specific key deserves trust. In the original design of CCN, applications can adopt arbitrary trust models for their contents. It also creates a tension between flexibility and security for content verification.

Due to the failure of current verification mechanism, content pollution has become a serious security threat in CCN [7], [8], [9], [10]. By hijacking source server, attackers can inject invalid contents into network and then disseminate them to users. When users receive the invalid contents, they only discard them and request again. Because frequent re-requests will lead to traffic congestion and seriously degrade the network performance, how to defense content pollution attack has been an urgent research topic in CCN.

Focusing on this problem, in this paper, we analyze the causes of content pollution, define the credibility of router by evaluating the validity of received data traffic, and then propose a probability verification strategy based on routers credibility (named as PVSRC in short). The theoretical and numerical analysis results show that our strategy can effectively defense content pollution while reducing unnecessary workload.

Compared with previous works, the contributions of this paper lie in three aspects. First, we introduce the concept of router credibility to evaluate whether the router is trustworthy. This indicator is calculated based on the credibility of upstream content sources, including upstream

routers or source servers. Second, we propose a probabilistic verification in which the verification probability is dynamically adjusted according to routers credibility and its actual handling capacity. This design fully considers the relationship between verification performer and its network environment. If the network is relative reliable, our strategy can significantly reduce the workload of router. Third, we derive the theoretical approximation expressions of valid content hit probability, verification overhead and packet drop rate of user. The theoretical analysis method we adopt can offer valuable reference for CCN researchers.

The remainder of this paper is organized as follows. Section 2 introduces content pollution attack and related research works. Section 3 gives our strategy in detail. Section 4 focuses on the theoretical analysis about PVSRC. In section 5, performance of PVSRC is proved through numerical calculation. Section 6 summarizes this paper and makes future research consideration.

II. Content Pollution and its Related Works

A. Content pollution

Content pollution attack is a kind of malicious attack aiming at ubiquitous caching mechanism of CCN. As shown in Figure 1, let's assume there are N devices in network, including N-n source servers and n routers. Server v_{n+1} is hijacked by attacker and injected into some invalid contents, e.g. an invalid content named Content-A-Fake is the imitation of valid content Content-A. Now user U_1, as the first requester, sends interest packet for Content-A. This interest packet will be forwarded to source server v_{n+1}, and be responded with the invalid data packet Content-A-Fake. If no effect verification mechanism performed in network, the invalid content will be cached into each router along the reverse path, then routers from v_{i-1} ... v_2 to v_1 will be all polluted. Next, when the subsequent requester U_2 requests Content-A again, he/she only retrieves the invalid data packet Content-A-Fake from the polluted router v_{i-1}. As a result, attacker separates users from valid content successfully. Whats worse, if the popular contents are polluted, the invalid data packets will be spread throughout the network and fatally damage the running of CCN.

From this example, we can find that the hijacked source server is the core cause of content pollution in CCN. As is illustrated in Fig.1, if all contents stored in source servers are valid, content pollution will not occur. On the other side, if no server is compromised except v_{n+1}, only the downstream routers of v_{n+1}, such as v_1 to v_i, are suffered pollution. Hence, we can draw a conclusion that under Pull-based communication mode of CCN, whether a router will suffer content pollution totally depends on its upstream servers or routers are safe or not. That is to say, in an unreliable network, the pollution level of a router rests on: (i) polluted percentage of its upstream routers or source servers; (ii) the traffic between the upstream content sources and itself. The more heavier traffic is, the more serious pollution it will suffer.

B. Related works

Because the verification mechanism is the key way to solve content pollution in CCN, researchers have carried

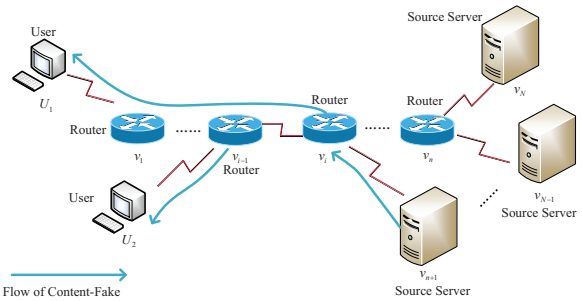

Fig. 1. Illustration of content pollution attack

out some meaningful explorations in this field recently. Bianchi et al. [11] suggested a scheme called check before storing. In this scheme, content is verified and cached with a certain probability p. Although probability verification is advisable, the fixed probability is not good design. If network is relative safe, high verification probability is completely unnecessary, only results in more computation overhead. But if network environment is poor, low verification probability cannot defend content pollution attack effectively.

Another important strategy is Verification on Hit (VOH) [12]. This strategy deploys SLRU replacement policy which divides CS into two parts, one is the protected area and another is the unprotected area. For new data packet received, it will be stored in the unprotected area first. When a hit event occurs, the target data packet will be verified. If credible, it will be moved to the protected area. Otherwise, it will be discarded. This design can reduce router's verification overhead dramatically, but its capacity of against polluted content is weak.

DiBenedetto et al. [13] point out that the underlying cause of poor data credibility is unreliable content producers. Based on this analysis, they propose a forwarding strategy that tries to forward interest packets to reliable producers as much as possible. This is a novel solution for CCN content pollution problem, but there are still some problems. In some terrible scenarios where most source servers are hijacked, CCN is unable to avoid attack if only depending on the improvement of forwarding strategy. So the improvement of verification mechanism is more important to defend against content pollution attack in CCN.

III. Description of PVSRC

The aforementioned works show the verification workload and proportion of invalid contents is a pair of contradiction for CCN router. To reduce workload blindly only leads to high content pollution risk. But more check works to filter invalid contents means considerable computing cost. Supposing that if routers can make a fair evaluation about the authenticity of data traffic received, the repeated verification will be cut down by a big margin. Due to the polluted producer is the fundamental cause of content pollution, the security state of each router is totally depended on its upstream source servers. In view of this, it is reasonable to assess the credibility of router based on its upstream routers' credibility and the traffic size from them.

Definition 1 (Credibility of CCN router): This value indicates the percentage of valid contents in the data traffic received by router. It is co-determined by the pollution status of upstream routers and the traffic from each upstream router. When network state is steady, it equals roughly to the percentage of valid contents stored in CS.

Based on the credibility evaluation, router can implement the probability verification strategy (PVSRC) and then reduce the redundant verification workload. Because credibility evaluation is the foundation of PVSRC, we should establish an evaluation mechanism at first.

A. Packet design for credibility evaluation

To obtain the statistics for credibility evaluation, now we modify the format of data packet, add two different interest packets to trace information of invalid content, and then set a management server(MS) to collect all invalid data reports from routers and users.

1) Data packet modification: A new field indicating producers' ID is added in original data packet. New format is "/Name/Meta-Info/Content/Signature/Source-ID". The ID field helps users and routers know the source of invalid content.

2) Credibility notification interest packet: This kind of interest is used for upstream routers to report their credibility to downstream routers. The format is "/Credit-Report/Upstream-ID/Credit". The meanings of each field are as follow, "Upstream-ID" is used to describe the identity of router which generates this interest packet, "Credit-Report" denotes that it is only used to notify credibility, "Credit" field is the credibility information of the reporter router.

3) Invalid content report: [13] This kind of interest is sent from router or user to inform management server when they detect invalid content through verification. Its format is "/Fake-Content-Report/Data-Name/Source-ID/Self-ID/Nonce/PK", where "Fake-Content-Report" indicates that this interest packet is used to report invalid content, "Data-Name" is the name of invalid content detected, "Source-ID" denotes which source server produces the invalid content, "Self-ID" is a unique identity of router in network. To avoid repeated reports for same content from different routers, a "Nonce" field filled by random number is included. The final field 'PK' is the public key to verify the signed report interest.

B. Deployment of PVSRC

The deployment of PVSRC includes two successive phases, initialization and running. In former phase, the credibility of source servers will be evaluated. In later phase, every router evaluates its own credibility respectively, and then sets verification probability according its credibility. The detail of PVSRC as follows.

- **Initialization phase**
  In initialization phase, each user and router must verify data packet as much as possible. If verification is fail, the router will generate **"Invalid content report packet"** and forward it to management server. When receiving this report, the management server checks the legality of this packet by matching public key in its PK database. If check passes, management server collects it for statistics. After collecting enough **"Invalid content reports"**, management server periodically evaluates the credibility of each source server using the following formula.

$$\mathrm{Credit}_i = 1 - \frac{\mathrm{reports}_i}{\mathrm{all\_reports}}, n+1 \le i \le N \quad (1)$$

Where Credit_i is the credibility of source server i ($0 \le \mathrm{Credit}_i \le 1$), $\mathrm{reports}_i$ denotes the amount of invalid content reports pointing to source server i, $\mathrm{all\_reports}$ means the total amount of invalid content reports received in statistics period. Obviously, for source server i, low credibility means it has fell into serious pollution and credibility approaching one means it is in clean state. Although router is forced to check the data packet passed by as much as possible, a considerable proportion of data packets still have no chance for verification due to limitation computation capacity of router. In this phase, the data packets without verification will be directly cached into CS.

Algorithm 1 Initialization of PVSRC

Input: Input Network Device

switch (Network Device)
  case 0 : User or Router
    The user or router verifies the signature of received data packet
    if (Signature verification fails) **then**
      Generates an invalid content report packet and forwards it to the MS
    else
      Router: Accept it and cache into CS
      User: Accept it
    end if
  case 1 : Management server(MS)
    while (MS collects report packets)
      for $(i = n+1 \to N)$ **do**
        if (the public key of report packets exist in the PK database) **then**
          $\mathrm{report}_i \leftarrow \mathrm{report}_i + 1$
        end if
      end for
      $\mathrm{all\_reports} \leftarrow \mathrm{all\_reports} + 1$
      for $(i = n+1 \to N)$ **do**
        $\mathrm{Credit}_i \leftarrow 1 - \frac{\mathrm{report}_i}{\mathrm{all\_reports}}$
      end for
end switch

- **Running phase**
  1) Credibility self-evaluation
  After initialization, the management server begins to send **"Credibility notification interest packet"** to the routers within one hop scope of source servers. When router receives credibility notification reports, it can calculate its own credibility according to expression (2) and then produces a new **"Credibility notification interest packet"** to its downstream routers. After a period of time, all routers in network

will estimate their own credibility.

$$\mathrm{Credit}_i = \sum_{j=1, j\neq i}^{N} f_{i,j}\mathrm{Credit}_j, 1 \leq i \leq n \quad (2)$$

In above expression, $f_{i,j}$ is the forwarding coefficient of v_i. It denotes the proportion of unsatisfied interest packets forwarded from v_i to v_j. Credit_i is the credibility of router v_i, Credit_j represents the credibility of upstream content source v_j (router or source server) which v_i directly connects to. Under the "Pull-based" mode, forwarding traffic of interest packets should be equal to the received traffic of data packets. So, for v_i, $f_{i,j}$ can also represent the influence from upstream router v_j.

2) Probability Verification

After the credibility estimation stage, router can perform probability verification strategy according to its credibility. In our design, the verification probability of a router should be determined by its credibility and its limitation verification capacity. Assuming $P_{\mathrm{verify}}(i)$ denotes verification probability of router v_i, it is given by

$$P_{\mathrm{verify}} = \min\{\frac{\mathrm{Limit}}{r_i}, 1 - \mathrm{Credit}_i\}, 1 \leq i \leq n \quad (3)$$

Where r_i means the received data traffic of v_i, Limit denotes the limitation verification capacity of v_i (it equals to the maximum number of data packets that can handle with per second). Considering that the credibility of router reflects the security self-assessment for area it is located at, if the network is relative safe, router can provide authentic content delivery service just only checking few contents. But if the network is unreliable, router must try its best for verification. This idea is reflected in expression (3), if data traffic received beyond router's capacity, it has no choice but to perform maximum verification; otherwise, it can perform verification with probability "$1 - \mathrm{Credit}_i$".

After verification, router will send **"Invalid content report packet"** to management server if detecting invalid data packet. For these data packets escaping from verification by probability $1 - P_{\mathrm{verify}}(i)$, they will be cached into CS as well as those passing verification.

Algorithm 2 Running of PVSRC

Input: Input Network Device

switch (Network Device)
  case 0 : Router
    The router estimates its own credibility Credit_i
    $\mathrm{Credit}_i \leftarrow \sum_{j=1, j\neq i}^{N} f_{i,j}\mathrm{Credit}_j, 1 \leq i \leq n$
    The router sends Credit_i o its downstream routers using "Credibility notification interest packet"
    if ($r_i >$ Limit) **then**
      The router calculates the verification probability $P_{\mathrm{verify}}(i)$
      $P_{\mathrm{verify}} \leftarrow \min\{\frac{\mathrm{Limit}}{r_i}, 1 - \mathrm{Credit}_i\}, 1 \leq i \leq n$
      The router performs probability verification according to $P_{\mathrm{verify}}(i)$
    else
      The router tries to verify all received data packets as much as possible
    end if
  case 1 : Management server (MS)
    MS sends the credibility of source servers to the routers that connected directly with source servers
    {MS repeats the steps 11-21 in Algorithm 1}
end switch

IV. Theoretical Analysis

In this section, we focus on the theoretical analysis for PVSRC from two aspects, computation overhead and verification efficiency. We adopt verification workload and valid content hit ratio as two main indicators and then derive their theoretical approximation expressions. The subsequent research works are based on the following assumptions.

- We model the network topology as a weighted diagraph $D = (V, F)$ [14]. Here, $V = \{v_i | 1 \leq i \leq N\}$ denotes the set of vertices of diagraph, each vertex representing a device. In previous section, we have assumed there are N devices in network including $N - n$ source servers and n routers, so we further define the subset $\{v_i | 1 \leq i \leq n\}$ as routers and $\{v_i | N - n + 1 \leq i \leq N\}$ as source servers.
- $F = \{f_{i,j} | 1 \leq i \leq N, 1 \leq j \leq N, i \neq j\}$, the set of edges of diagraph, it represents the upstream forwarding relationship between two arbitrary devices (router or source sever). In this set, $f_{i,j}$ is the forwarded proportion of unsatisfied interest packets from v_i to its upstream source v_j, $0 \leq f_{i,j} \leq 1$ (if v_i and v_j are not connected to each other, $f_{i,j} = 0$)
- Source servers can provide M different content items which are equally categorized into K classes, with $m = M \times K^{-1}$ content items in each class. We assume that every content item corresponds to just one data packet.
- Content items of class k are requested with probability q_i^k at v_j, $k \geq 1$. According to the paper [15], we assume the popularity of arrival requests at the first level follows Zipf distribution with $q_i^k = c \times k^{-\alpha}, c > 0$, where parameter α represents the content popularity. Larger α means the requests will be more concentrated in the first several classes.
- For v_i, we define $P_{\mathrm{v-hit}}(i)$ as valid content hit probability, $r_{\mathrm{v-cs}}(i)$ as valid content proportion in CS, $V_{\mathrm{Load}}(i)$ as verification workload (data packet/s), $P_{\mathrm{hit}}(i)$ as average hit probability and $P_{\mathrm{hit}}(i, k)$ as average hit probability of class k.
- We assume that v_i generates the interest packets according to the Poisson process of intensity λ_i, then we further define λ_i^k as arrival rate of requests for class k at v_i. Considering that the unsatisfied interest packets will be forwarded upstream, so $\lambda_j^k = \sum_{i=1, i\neq j}^{N} \lambda_i^k (1 - P_{\mathrm{hit}}(i, k)), 1 \leq j \leq N$.
- The cache size and limitation verification capacity

of every router are same and we assume cache size is equal to C content items and the maximum verification capacity is Limit (data packet/s).

A. Valid content hit probability

Definition 2 (Valid content hit probability): For received requests, the probability of hitting on the valid contents stored in CS is valid content hit probability. It is used to assess the reliability of verification strategy. The higher it is, the better protection against content pollution attack the strategy can offer. Let $P_{\mathrm{v-hit}}(i)$ denotes this indicator at v_i, it can be expressed as

$$P_{\mathrm{v-hit}}(i)=r_{\mathrm{v-cs}}(i) \times P_{\mathrm{hit}}(i) \tag{4}$$

Where $r_{\mathrm{v-cs}}(i)$ is valid content proportion in CS of v_i and $P_{\mathrm{hit}}(i)$ is the average hit probability at v_i. Because $P_{\mathrm{v-hit}}(i)$ depends on both $P_{\mathrm{hit}}(i)$ and $r_{\mathrm{v-cs}}(i)$, next, we will give detailed analysis for these two factors respectively.

1) Analysis of average hit probability

From the working mechanism of PVSRC, this strategy can be regarded as a probabilistic in-network caching policy [16], [17]. When a router receives a data packet, the packet will be probabilistically cached into CS under two situations: (i) the packet isn't chosen for verification with probability $1 - P_{\mathrm{verify}}(i)$; (ii) the packet is chosen for verification and it is authentic. Let β_i denotes the probability for a packet to be stored into CS, it can be expressed as follow.

$$\begin{aligned}\beta_i =&(1 - P_{\mathrm{verify}}(i)) \\ &+ P_{\mathrm{verify}}(i) \times \sum_{j=1,i\neq j}^{N} (f_{i,j} \times r_{\mathrm{v-cs}}(i))\end{aligned} \tag{5}$$

Here, $\sum_{j=1,i\neq j}^{N} (f_{i,j} \times r_{\mathrm{v-cs}}(i))$ represents the average percentage of valid content in received data traffic of v_i. According to the references [18], [19], for hit probability analysis, we usually analyze the hit events of two successive requests. In this sense, a hit event of PVSRC will happen in following cases.

- Target content already exists in CS and there is a hit for the first request. If the interval between two successive requests is smaller than the average caching time (it is also called as characteristic time of cache), a hit event will occur again when the second request arrives.
- Unfortunately, the target content doesn't exist in CS and a miss event happens for the first request. Then the router fetches target content from network and caches it into CS with probabilityβ_i. If we ignore the transmission delay, a hit event will occur for the second request on the same condition that the interval between two successive requests is smaller than characteristic time.

When network works in steady state, we can regard the stationary hit probability for two successive requests as same, now we use $P_{\mathrm{hit}}(i,k)$ to denote the average hit ratio of class k at v_i. If τ_i^k indicates the interval between two successive requests for content item in class k at v_i and t_i denotes the characteristic time of v_i, then $P_{\mathrm{hit}}(i,k)$ can be derived by

$$\begin{aligned}P_{\mathrm{hit}}(i,k) =&P_{\mathrm{hit}}(i,k) \times P\{\tau_i^k \leq t_i\} \\ &+ (1 - P_{\mathrm{hit}}(i,k)) \times \beta_i \times P\{\tau_i^k \leq t_i\}\end{aligned} \tag{6}$$

Assuming that requests for class k at v_i obey Poisson distribution with parameter λ_i^k and the arrival rate for each content item in class k is λ_i^k/m (m is the total number of files in each class), then $P_{\mathrm{hit}}(i,k)$ can be simplified as

$$P_{\mathrm{hit}}(i,k) = \frac{\beta_i \times (1 - e^{-\lambda_i^k \times t_i/m})}{1 + (\beta_i - 1) \times (1 - e^{-\lambda_i^k \times t_i/m})} \tag{7}$$

The average hit ratio $P_{\mathrm{hit}}(i)$ for all classes at v_i should be

$$P_{\mathrm{hit}}(i) = \sum_{k=1}^{K} q_i^k P_{\mathrm{hit}}(i,k), 1 \leq i \leq n \tag{8}$$

In above analysis of $P_{\mathrm{hit}}(i)$, the characteristic time t_i is a key parameter, so we should discuss it in detail.

Definition 3 (Characteristic time) [20]: Characteristic time t_i is the average caching time of v_i for arbitrary data packet. Take the typical replacement policy - LRU as example, t_i is the average time for a data packet to move from the head of cache queue to the tail. According to the references [18], [21], in PVSRC, t_i can be obtained by

$$\begin{aligned}C =&m \sum_{k=1}^{K} P_{\mathrm{hit}}(i,k) \times P\{\tau_i^k \leq t_i\} \\ &+ m \sum_{k=1}^{K} (1 - P_{\mathrm{hit}}(i,k)) \times \beta_i \times P\{\tau_i^k \leq t_i\}\end{aligned} \tag{9}$$

Where K is the total number of classes and C is the size of CS. Equation(9) indicates that if the amount of inserted content items to cache head during t_i is equal to the size of CS, a content item originally stored in cache head will be removed to the tail of cache. So, t_i is the average caching time for data packet residing in CS. In equation (9), $P_{\mathrm{hit}}(i,k) \times P\{\tau_i^k \leq t_i\}$ represents the insert probability for cache head when a hit event occurs, and $(1 - P_{\mathrm{hit}}(i,k)) \times \beta_i \times P\{\tau_i^k \leq t_i\}$ represents the insert probability for cache head when a miss event occurs.

2) Analysis of valid content percentage

Definition 4 (Valid content percentage): Valid content percentage is the ratio of the amount of cached valid data packets to the total amount of cached data packets within the unit time.

To clarify this definition, we should distinguish the following concepts, received data traffic, cached data traffic, and received valid data traffic.

- According to aforementioned analysis, the cached probability for data packet (β_i)means some invalid contents are discarded. Therefore, if let r_i denotes received data traffic, the amount of cached data traffic is $r_i \cdot \beta_i$.
- Under 'Pull-based' communication mode, received data traffic depends on the unsatisfied interest packets. If we ignore the transmission loss in 'Pull-and-push' process, the percentage of received data traffic

from v_j to v_i in the total received data traffic of v_i should be same as the proportion of unsatisfied interest packets forwarded from v_i to v_j. That means this proportion can be approximately regarded as $f_{i,j}$. So, the amount of received valid data traffic from v_j should be $r_i \cdot f_{i,j} \cdot P_{\rm hit}(j)$, where $P_{\rm hit}(j)$ is the percentage of valid content of v_j.

- If v_i has several upstream content sources, received valid data traffic of v_i can be expressed as $\sum_{j=1,j\neq i}^{N} f_{i,j} \cdot r_i \cdot P_{\rm hit}(j)$.

Based on above analysis, it is obvious that valid content percentage of v_i is given by

$$r_{\rm v-cs}(i) = \frac{\sum_{j=1,j\neq i}^{N} f_{i,j} \cdot r_i \cdot r_{\rm v-cs}(i)}{r_i \beta_i}, 1 \leq i \leq n \tag{10}$$

Simplified as

$$r_{\rm v-cs}(i) = \frac{\sum_{j=1,j\neq i}^{N} f_{i,j} \cdot r_{\rm v-cs}(i)}{\beta_i}, 1 \leq i \leq n \tag{11}$$

From expression (11) we can see clearly that valid content percentage of v_i depends on verification probability and all content sources connecting. The reasonable selection of verification probability can guarantee a relative high valid content percentage in CS.

B. Verification workload

Definition 5 (Verification workload): It is the amount of verified data packets within the unit time. For v_i, we use $V_{\rm Load}(i)$ to express verification workload. Verification workload represents the overhead of verification strategies. If a router suffers a huge verification workload, its computation resource will be exhausted soon, then network become unsafe due to the router cannot work normally. For PVSRC, the verification workload of v_i is determined by verification probability $P_{\rm verify}(i)$ and received data traffic r_i directly. So $V_{\rm Load}(i)$ can be expressed as

$$V_{\rm Load}(i) = r_i \times P_{\rm verify}(i), 1 \leq i \leq n \tag{12}$$

C. Packet Drop Rate of User

Considering that the data packet will be compulsorily verified when it is received by user, the invalid data packet will be discarded in user side, we can use the "packet drop rate of user" to further describe the reliability of network.

Definition 6 (Packet Drop Rate of User): The miss probability of interest packets sent by user during the observation time.

Now we assume the request hit probability of user is $P_{\rm u-hit}$, packet drop rate of user is $P_{\rm u-drop}$, then $P_{\rm u-drop}$ can be expressed as

$$P_{\rm u-drop} = 1 - P_{\rm u-hit} \tag{13}$$

For user, $P_{\rm u-hit}$ essentially points to the hit probability of valid data packet received. It is equal to the sum probability of valid content hit in network and source server, as shown in expression (14).

$$P_{\rm u-hit} = \left(1 - \sum_{i=1}^{n} P_{\rm hit}(i)\right) \times \frac{\sum_{i=n+1}^{N} r_{\rm v-hit}(i)}{N-n} + \sum_{i=1}^{n} P_{\rm hit}(i) \tag{14}$$

According to expression (4), $\sum_{i=1}^{n} P_{\rm v-hit}(i)$ indicates the valid content hit probability in network. According to expression (8), the valid content hit probability in source server is the product of "hit probability in source server" and "average valid content percentage of source server". The former can be calculated as $1 - \sum_{i=1}^{n} P_{\rm hit}(i)$, the latter can be expressed as $\frac{\sum_{i=n+1}^{N} r_{\rm v-cs}(i)}{N-n}$, where $r_{\rm v-cs}(i)$ is the valid content percentage of source server v_{n+1}–v_N.

V. Numerical results

In this section, we use numerical analysis to evaluate the performance of PVSRC from three aspects, valid content hit probability $P_{\rm v-hit}(i)$, verification workload $V_{\rm Load}(i)$, and the packet miss rate of user $P_{\rm u-drop}$. The numerical analysis tool is Matlab and the results are calculated according to the expressions (4), (12) and (14). In order to illustrate the advantages and shortcomings of our strategy, we select VOH [12] verification strategy as comparison. Network topology of this analysis is a tandem architecture with two layer, as shown in Fig.2. Where v_1, v_2 and v_3 are three CCN routers, v_4 and v_5 are two polluted source servers and U stands for a user. In addition, as mentioned earlier, there is also a management server in network.

The parameters for numerical analysis are set as follows [19], [22]. CCN provides a population of M items by source servers, categorized in $K = 50$ classes, each one with $m = 100$ items. The ratio of CS size to the total content population is $C/M = 0.05$. For the first level, the popularity of arrival requests at v_1 follows the Zipf distribution [23] with $\alpha = 1.2$. This is a typical value of VOD (Video on Demand) service [24]. According to reference [5], the limitation verification capacity is set as 1.25×10^4 data packet/s.

Further, the upstream forwarding coefficients in Fig.2 are configured as $f_{1,2} = f_{1,3} = 0.5$, $f_{2,4} = 0.7$, $f_{2,5} = 0.3$, $f_{3,4} = 0.6$, $f_{3,5} = 0.4$. Obviously, valid content hit probability of arbitrary router is directly impacted by valid content percentage of source servers, while verification workload is directly impacted by received request traffic at edge routers. Based on this consideration, in our numerical experiments, we fix the valid content percentage of v_4, set $r_{\rm v-hit}(4)$=0.8, and then (i) change invalid content percentage $1 - r_{\rm v-hit}(5)$ of v_5; (ii) change the request traffic λ_1 at v_1. These experiments will disclose the reliability and effectiveness of VOH and PVSRC when network suffers different security threat and different access traffic.

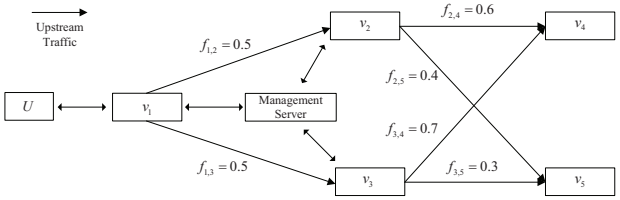

Fig. 2. Two-layer tandem topology

A. Impact of invalid content percentage of source server

Now we fix the arrival rate of received interest traffic at v_1 as interest packet/s, then increase the invalid content percentage of v_5 from 0.1 to 0.9. Fig. 3 and Fig. 4 show the impact of content pollution status of v_5 under VOH and PVSRC respectively.

As shown in Fig 3, for PVRSC, its valid content hit probability is obviously superior than VOH which will lead to a higher proportion of invalid content in CS. The reason is that parts of invalid contents are filtered before being stored into CS by probability verification. Another important point can be observed, for PVSRC, the deterioration speed of edge router v_1 is slower than its upstream routers v_2 and v_3 when v_5 provides more invalid contents. Because v_2 and v_3 are close to v_5, they are sensitive with the polluted status of v_5. For v_1, benefitted from more invalid data packets filtered by its upstream routers, the valid content percentage of its CS varies relative smoothly. Therefore, upstream routers can play a buffering role against degradation of source server under PVSRC.

Fig. 4 reports the relationship between verification workload and polluted level of v_5. Due to v_1 has already reached its maximum verification capacity when its access request traffic is greater than 1.25×10^4 interest packet/s, for both two strategies, the verification workload of v_1 never changes under different invalid content percentage of v_5. But for v_2 and v_3, although they don't face the embarrassment of verification overload, their verification workloads also keep unchanged under VOH. Because the verification event of VOH strategy only occurs after hit event, it is entirely different from the valid content hit event, so the verification workload under VOH doesn't change with the variation of received invalid contents. Under PVSRC, with the increasing of invalid contents provided by v_5, verification workload of v_2 and v_3 will increase accordingly. When the network is relative safe (just as $1 - r_{\mathrm{v-hit}}(5)$ ranging from 0.1 to 0.5), v_2 and v_3 will adopt a relative low verification probability. So, their verification workload is lower than under VOH. Furthermore, when network suffers from aggressive pollution, PVSRC downgrades the credibility of v_2 and v_3. As a result, the verification probability is dynamically lifted and therefore the verification workload of v_2 and v_3 are heavier than under VOH.

Note that, the verification workload under PVSRC will be a bit heavier than under VOH when source server swamp in serious content pollution (e.g. $1 - r_{\mathrm{v-hit}}(5)$ is greater than 0.5). But just in these cases, verification workload under PVSRC only reaches the 1/3 of limitation verification capacity. Considering that PVSRC owns better valid content hit probability, it can provide more excellent performance against content pollution attack.

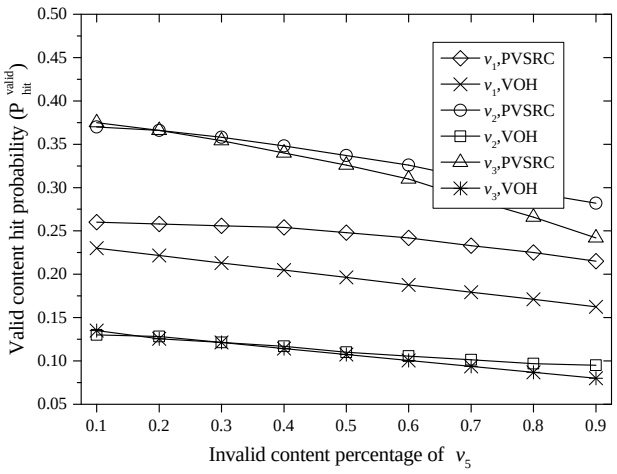

Fig. 3. Pollution level of v_5 vs. $P_{\mathrm{v-hit}}$

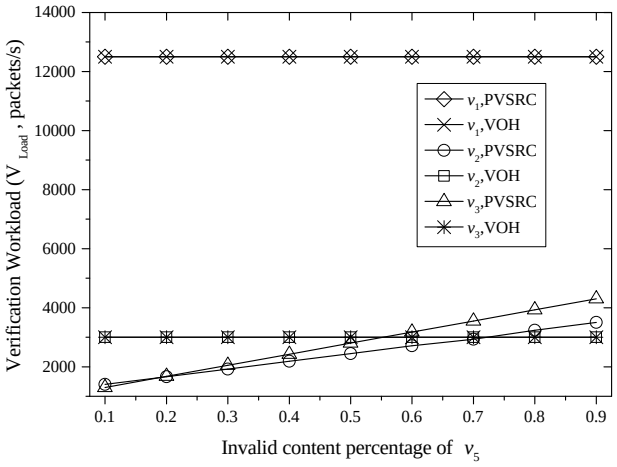

Fig. 4. Pollution level of v_5 vs. V_{Load}

B. Access request traffic of v_1

Fig. 5 and Fig. 6 show the impact of access request traffic for valid content hit probability and verification workload. In this part, $r_{\mathrm{v-hit}}(5)$ is fixed at 0.5 and request traffic arriving at v_1 increases from 3×10^4 to 5×10^4 interest packet/s. Other parameters are same as above setting.

From Fig. 5, we can find valid content hit probability under PVSRC is always higher than that under VOH when request traffic varies. For v_2 and v_3 with PVSRC, with the increasing of request traffic, their verification probabilities don't change but more invalid contents are detected and discarded. As a result, the cache update time (characteristic time) will be slowed down comparing with the original characteristic time. So, the hit probability of v_2 and v_3 will relatively increase, then lead to the improvement of valid content hit probability. But for v_1, the detected and discarded invalid contents only change a little because the buffering effect of v_2 and v_3, so the characteristic time and valid content hit probability of v_1 are almost same. By the way, under VOH, the valid content percentage and hit probability of router doesn't change with request traffic

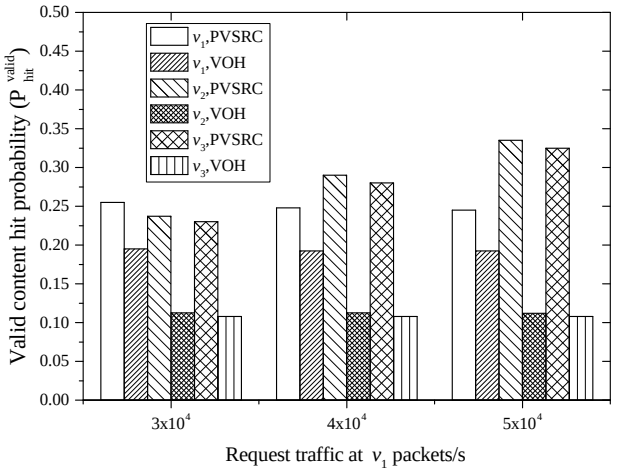

Fig. 5. Pollution level of v_5 vs. $P_{\mathrm{v-hit}}$

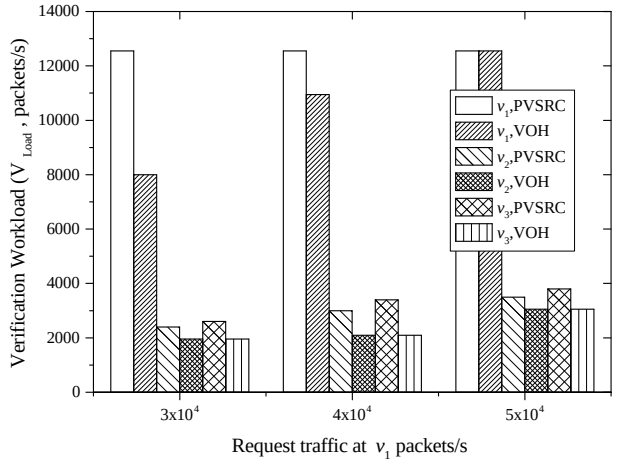

Fig. 6. Pollution level of v_5 vs. V_{Load}

variation, so the valid content hit probability is fixed.

Fig. 6 shows the strategy workload when network traffic is heavy. For v_2 and v_3 under PVSRC, the setting of $r_{\mathrm{v-hit}}(4)$=0.8, $r_{\mathrm{v-hit}}(5)$=0.5 indicate the network is not safe, they must adopt high verification probability. No doubt, their verification workload will increase when request traffic rises. Under VOH, note that the verification workload is totally determined by the request traffic. For parts of requests have already satisfied at v_2, the request traffic arrived at v_2 and v_3 grow slowly, so their verification workload also have a slight growth. Although verification workload of all three routers under PVSRC are heavier than under VOH, PVSRC can provide more effective defense for content pollution attack. Its reliability and feasibility are far superior than VOH.

C. Packet drop Rate of User

Fig. 7 discloses the influence of average pollution level of source server under the conditions of 2×10^4 and 5×10^4. From this figure, when average pollution level of source server is less than 0.5, the packet drop rates of both two strategies are all in low level. This phenomenon shows that user can work normally if network exists in a relative light polluted status. But if the pollutions of source servers worsen, the packet drop rate under both two strategies

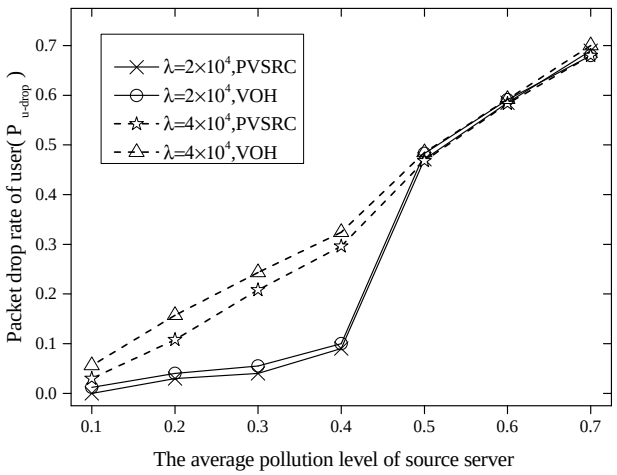

Fig. 7. Pollution level of v_5 vs. V_{Load}

increase obviously. Totally, the performance of PVSRC is superior than VOH, although the superiority is only a little when network falls into deep pollution. For the serious pollution situation, only depending on the improvement of verification mechanism, the content pollution attack cannot be defended. To locate and insulate the pollution source is the fundamental solution.

VI. Conclusion

This paper makes helpful exploration about content verification design in unsecure CCN and proposes a probability verification strategy named PVSRC. PVSRC designs a pertinent assessment about the credibility of router and sets a proper probability to perform verification based on it. As the numerical results show, this kind of novel design can reduce the workload of routers and improve feasibility of verification strategy significantly. Compared with VOH, PVSRC may cause a little heavier workload under terrible network security environment, but can ensure rather high valid content hit probability, and there is no doubt that offers stronger protection against content pollution attack.

However, we think that there is still much space for improvement in PVSRC. Researches in the future will focus on the following aspects.

- Verification overload at the edge routers. The huge traffic edge routers have to deal with be a potential invitation for more content pollution if no optimization is made.
- Trust management. In PVSRC, upstream routers are ruled to broadcast their creditability to downstream router. Such information exchange needs extra trust management otherwise attackers will exploit this weakness and launch invalid creditability attacks.

References

[1] Koponen Teemu, Chawla Mohit, Chun Byung-Gon, et al., "A data-oriented (and beyond) network architecture," *ACM SIGCOMM Computer Communication Review*, vol.37, no.4, pp.181-192, 2007.

[2] Lagutin Dmitrij, Visala Kari, Tarkoma Sasu., "Publish/Subscribe for Internet: PSIRP Perspective," *Future internet assembly*, vol.84, pp.75-84, 2010.

[3] Jacobson Van, Smetters Diana K, Thornton James D, et al., "Networking named content," *Proceedings of the 5th international conference on Emerging networking experiments and technologies, ACM*, pp.1-12, 2009.

[4] Yu Yingdi, Afanasyev Alexander, Clark David, Jacobson Van, Zhang Lixia,et al. "Schematizing trust in named data networking," *Proceedings of the 2nd International Conference on Information-Centric Networking, ACM*, pp. 177-186, 2015.

[5] Gasti Paolo, Tsudik Gene, Uzun Ersin, Zhang Lixia. "DoS and DDoS in Named Data Networking," *International Conference on Computer Communications and Networks,IEEE*, pp. 1-7, 2012.

[6] Kim DaeYoub. "Efficient and secure NDN," *Ubiquitous and Future Networks (ICUFN), 2015 Seventh International Conference on IEEE*, pp.118-120, 2015.

[7] Ghali Cesar, Tsudik Gene, Uzun Ersin. "Needle in a haystack: Mitigating content poisoning in named-data networking," *Proceedings of NDSS Workshop on Security of Emerging Networking Technologies (SENT)*, pp. 68-73, 2014.

[8] Xie Mengjun, Widjaja Indra, Wang Haining. "Enhancing cache robustness for content-centric networking," *INFOCOM, 2012 Proceedings IEEE*, pp. 2426-2434, 2012.

[9] Li Qi, Zhang Xinwen, Zheng Qingji, Sandhu Ravi, Fu Xiaoming. "LIVE: lightweight integrity verification and content access control for named data networking," *IEEE Transactions on Information Forensics and Security*, vol.10, no.2, pp.308-320, 2015.

[10] Aamir Muhammad, Zaidi Syed Mustafa Ali. "Denial-of-service in content centric (named data) networking: a tutorial and state-of-the-art survey," *Security and Communication Networks*, vol.8, no.11, pp.2037-2059, 2015.

[11] Bianchi Giuseppe, Detti Andrea, Caponi Alberto, Blefari Melazzi Nicola. "Check before storing: What is the performance price of content integrity verification in LRU caching?" *ACM SIGCOMM Computer Communication Review*, vol.43, no.3, pp.59-67, 2013.

[12] Kim Dohyung, Nam Sunwook, Bi Jun, Yeom Ikjun. "Content verification in Named Data Networking," *International Conference on Information NETWORKING, IEEE*, pp.109-116, 2015.

[13] DiBenedetto Stephanie, Papadopoulos Christos. "Mitigating poisoned content with forwarding strategy," *Computer Communications Workshops (INFOCOM WKSHPS) on 2016 IEEE*, pp.164-169, 2016.

[14] Diestel Reinhard. "Graph theory.2005.Grad," *Texts in Math*. 2005.

[15] Ioannou Andriana, Weber Stefan. "Exploring content popularity in information-centric networks," *China Communications*, vol.12, no.7, pp.13-22, 2015.

[16] Psaras Ioannis, Chai Wei Koong, Pavlou George. "Probabilistic in-network caching for information-centric networks," *Proceedings of the second edition of the ICN workshop on Information-centric networking,ACM*, pp. 55-60, 2012.

[17] Zhang Guoqiang, Li Yang, Lin Tao. "Caching in information centric networking: A survey," *Computer Networks*, vol.57, no.16, pp.3128-3141, 2013.

[18] Che Hao, Tung Ye, Wang Zhijun. "Hierarchical web caching systems: Modeling, design and experimental results," *IEEE Journal on Selected Areas in Communications*, vol.20, no.7, pp.1305-1314, 2002.

[19] Carofiglio Giovanna, Gallo Massimo, Muscariello Luca, Perino Diego. "Modeling data transfer in content-centric networking," *Proceedings of the 23rd international teletraffic congress*, pp.111-118, 2011.

[20] Laoutaris Nikolaos, Che Hao, Stavrakakis Ioannis. "The LCD interconnection of LRU caches and its analysis," *Performance Evaluation*, vol.63, no.7, pp.609-634, 2006.

[21] Zhu Yi, Mi Zheng-Kun, Wang Wen-Nai. "A Probability Caching Decision Policy with Evicted Copy Up in Content Centric Networking," *Journal of Internet Technology*, vol.99, no.99, pp.1-11, 2015.

[22] WANG Guo-qing, HUANG Tao, Jiang LIU, CHEN Jian-ya, LIU Yun-jie. "Modeling in-network caching and bandwidth sharing performance in information-centric networking," *The Journal of China Universities of Posts and Telecommunications*, vol.20, no.2, pp.99-105, 2013.

[23] Adamic Lada A, Huberman Bernardo A. "Zipf's law and the Internet," *Glottometrics*, vol.3, no.1, pp.143-150, 2002.

[24] Fricker Christine, Robert Philippe, Roberts James, Sbihi Nada. "Impact of traffic mix on caching performance in a content centric network," *Computer Communications Workshops (INFOCOM WKSHPS)on 2012 IEEE* , pp.310-315, 2012.