LASA: Lightweight, Auditable and Secure Access Control in ICN with Limitation of Access Times

Peixuan He*, Yinxin Wan†, Qiudong Xia†, Shaohua Li*, Jianan Hong*, Kaiping Xue*‡
*The Department of EEIS, University of Science and Technology of China, Hefei, Anhui 230027 China
†The Department of Information Security, University of Science and Technology of China, Hefei, Anhui 230027 China
‡kpxue@ustc.edu.cn (corresponding author)

Abstract—**Information Centric Networking (ICN), a future network architecture candidate, aims to alleviate the problem of insufficient bandwidth in traditional IP network. In ICN, contents are distributed in the whole network, so access control becomes more intractable. As we know, almost all of existing solutions consider it as a "Yes or No" problem, where a user either has the permission to access the corresponding content or not. However, in many practical situations, a content provider doesn't expect a single authorized user has the ability to access its repertory without times limitation when taking copyright protection into account. In this paper, we propose LASA, a lightweight, auditable and secure solution where legitimate users are limited to access a content provider's data within pre-designate times. In LASA, each content provider sets maximum access times for each legitimate user and edge routers perform authentication and audit based on users' signatures attached to interest packets. Once a legitimate user attempts to exceed his/her limited access times, his/her secret key will be leaked and the dishonest behavior will be detected. Our security analysis shows that LASA can provide signature unforgeability, data confidentiality and other security features. Experiment results show that our scheme LASA brings a little computational cost.**

I. Introduction

Information Centric Networking has recently been proposed as an ideal future network structure to cope with bandwidth insufficiency problem in traditional IP network [1]. It has many excellent properties: in-network cache, support for mobility and multicast by nature, built-in security, etc. However, these properties bring some new challenges on security as well. Due to the decoupling of the contents and content providers (CPs), the contents are cached by cache-enabled routers in ICN and distributed in the whole network. So users can access contents arbitrarily from the routers in ICN without content providers' permission and notice, which will severely damage the content providers' benefits. Hence, access control in ICN is indeed an important research issue needed to be focused on.

Quite a few relevant schemes have been proposed in recent years. Some schemes try encryption-based method [2]–[4], where any user can fetch the contents but only legitimate users can decrypt them. However, because of the indiscriminate service of ICN, it is easy for malicious attackers to deplete the network resources by flooding numerous interest packets during a short time [5]. Some schemes point out the entities where caches hit should authenticate the user before sending data packets back to the user. Obviously, this requires all the routers in ICN have the authentication ability or calls for extra authentication entities [6], which brings much computation overhead [7] or communication overhead [8].

Moreover, these schemes simply consider access control as a "Yes or No" problem, where users are only classified as legitimate users and illegitimate users. But in practice, most content providers achieve profitability by offering content subscription service. In order to protect their copyright and earn more profits, they commonly set a limitation on users' access times during a certain period (e.g., a week or a month). Apparently, none of the aforementioned schemes can satisfy this requirement, which becomes the obstacle to popularize the ICN paradigm.

We propose LASA, a lightweight, auditable and secure access control scheme in ICN to meet the access times limitation requirement. In LASA, we utilize broadcast encryption to guarantee the data confidentiality and design a signature algorithm to limit users' access times, which is motivated by [9]. We put authentication job on edge routers to filter illegitimate requests and intermediary cache-enabled routers only care about forwarding and caching. Besides, we take users' requests continuity into full consideration, by using hash chain to efficiently decrease the authentication overhead. Our contributions can be summarized as follows:

- We propose a lightweight and secure edge-side access control scheme in ICN, where illegitimate requests will be blocked at the very beginning. To make our system more efficient, we present a new authentication way with the help of hash chain.
- We conduct a novel signature algorithm, by which users can only access the specific content with limited times. Once users are dishonest with the maximum access times, their secret keys will be leaked and Internet Service Provider (ISP) can audit based on this.

The rest of this paper is organized as follows. Section II first introduces the related work and Section III presents the system model and the threat assumption. After stating the preliminaries in Section IV, the proposed scheme is given in Section V. Then section VI shows both the security and performance analysis. Finally we conclude this paper in Section VII.

II. Related Work

Access Control in ICN: In recent years, more and more schemes have been proposed to cope with the access control problem in ICN, some of which are also qualified with some

978-1-5386-3180-5/18/$31.00 ©2018 IEEE

other security properties, such as privacy preserving [6] and cache poison attack resistance [10], etc.

Several encryption technologies are introduced into access control schemes in ICN. Zheng *el al.* [2] provided a data access control solution in ICN using proxy re-encryption. Edge routers are required to re-encrypt received data before forwarding them to the users and users need to ask CPs for the decryption key for every content. Misra *el al.* [3] designed an access control framework based on broadcast encryption which can efficiently achieve user revocation. Attribute-based encryption (ABE) is also widely used in many network scenarios because of its natural support for fine-grained access control [4], [11], [12]. These schemes aforementioned achieve access control by controlling users' decryption ability, but they are vulnerable to DoS attacks and are often lack of support for user revocation.

There are some other schemes to stipulate that users should be authenticated before obtaining their desired contents [6]–[8]. In [6], all the users need to be authenticated by the access control point. Li *el al.* [7] proposed a lightweight integrity verification and access control scheme, in which only legitimate users can get private tokens from CPs and only the users who have private tokens can verify data integrity successfully and fetch the contents. In [8], each authorized user shares a secret with the CP and routers in ICN and they can distinguish authorized and unauthorized users based on shared secrets. Nevertheless, these schemes either bring much computation and communication overhead or need an additional entity for user authentication.

Access Times Limitation Requirement: Though it is a deeply considered issue in cloud computing and some other areas [13], [14], there is still no scheme to meet this requirement in ICN. Ning *el al.* [13] built a concrete σ-time oursourced CP-ABE system. Decryption is outsourced to cloud server and users can only request cloud to decrypt encrypted data at most σ times by maintaining a counter in cloud server. Yuen *el al.* [14] designed a k-times access control system using attribute-based signature. This scheme provides a mechanism to detect whether the user has exceeded k-times for accessing the system using some defined claim-predicate.

III. System Model and Threat Assumptions

A. System Model

The Information Centric Networking structure is mainly consisted of three kinds of components: Content Providers, Internet Service Provider and users. Our proposed system model can be depicted by Fig. 1.

Content providers deal with user registration, key management and content publication. They also need to specify users' maximum access times and provide user information when queried by edge routers. There are two kinds of routers in ICN provided by ISP: intermediary routers and edge routers. Intermediary routers are responsible for forwarding interest/data packets and caching the contents they forward. Edge routers are the routers with additional tasks that are in charge of authenticating requests before transmitting them to the core network. Users consume contents by sending corresponding interest packets and legitimate users can be categorized as: limited users and unlimited users according to whether they can access the contents without limitation.

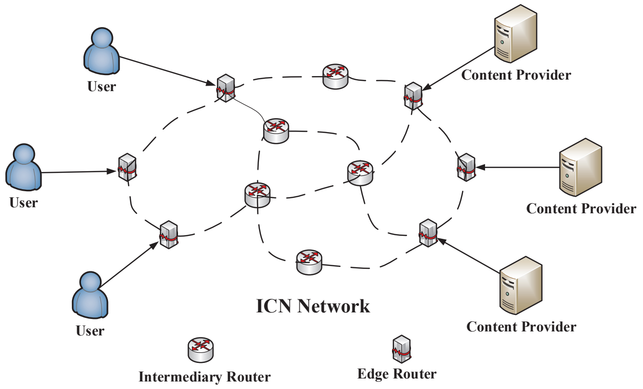


Fig. 1. System model.

B. Threat Assumptions

In our system, we assume that ISP is semi-trusted. First, CPs can not trust ISP on the audit messages it offers entirely. Next, ISP is curious but honest. It honestly carries out the protocol, but it also tries to learn something from the contents they store in the meanwhile.

Users are considered to be malicious. For the unauthorized users, they will try to access the contents which are unavailable for them. For the authorized users, they are also greedy and try to get extra contents as many as they can. Content providers are assumed to be trusted because they own the contents and are responsible for their security.

IV. Preliminaries

A. Bilinear Map

Let G_1, G_2 be an additive cyclic group and a multiplicative cyclic group of the same order q respectively. A bilinear map can be described as $e : G_1 \times G_1 \rightarrow G_2$ with the following properties: *(1) Bilinearity:* For all $a, b \in Z_q^*$ and $P, Q \in G_1$, we have $e(aP, bQ) = e(P, Q)^{ab}$. *(2) Computability:* There exists an algorithm that can compute $e(P, Q)$ for any $P, Q \in G_1$ efficiently. *(3) Non-degeneracy:* Supposing P is the generator of G_1, we have $e(P, P) \neq 1$.

Besides, from security perspective, the cyclic groups and bilinear map are assumed to hold the following two computational problems:

Definition 1: **Discrete Logarithm Problem (DLP).** Given $P, P^{'} \in G_1$, it is computationally intractable to find an integer n making $P = nP^{'}$.

Definition 2: **Weak Bilinear Diffie-Hellman Exponent Assumption (WBDHE Assumption).** Given a tuple $(P_1, rP_1, r^2P_1, ..., r^nP_1, P_2 \in G_1)$, where $r \in Z_q^*$, it is infeasible to compute $e(P_1, P_2)^{\frac{1}{r}}$.

B. Lagrangian Interpolation Polynomial

Given a set of $n+1$ points $(x_0, f(x_0))$, $(x_1, f(x_1))$, ..., $(x_n, f(x_n))$ in the polynomial $f(x)$ of degree n, the corresponding Lagrangian interpolation polynomial of degree n can be calculated as following:

$$L_n(x) = \sum_{j=0}^{n} (f(x_j) \prod_{k=0, k \neq j}^{n} \frac{x - x_k}{x_j - x_k}).$$

Because the interpolating polynomial of the least degree is unique, so we can reconstruct the polynomial $f(x)$ by calculating the Lagrangian interpolation polynomial of the same degree.

V. Proposed Scheme

A. System Overview

In our proposed scheme, CPs divide their stored contents into different groups with unique GIDs according to their security policy and add GIDs into contents' name. Besides, CPs utilize Privilege Mask [15] [16] to represent users' privileges and each user can access more than one groups' contents. Here Privilege Mask is a bit map and each bit represents whether the user is available to the contents in corresponding group. For example, if the second bit of a user's Privilege Mask is "1", it indicates that the user has right to access the contents in the group whose GIDs are 2. In this paper, we assume a CP divides its contents into m groups.

Intuitively, the basic idea of our system is illustrated in Fig. 2. A user registers to the CP at first and then the user can send out interest packets attached with signatures to fetch the desired contents. The edge router authenticates requests based on the user's information got from the CP. After successful authentication, the interest packets are allowed to enter the core network and bring the corresponding contents to the user. To guarantee the data confidentiality, the CP encrypts the contents by broadcast encryption before publication. If a users is dishonest with his/her maximum access times, ISP will know his/her secret key as shown in section VI and report this dishonest behavior to the CP.

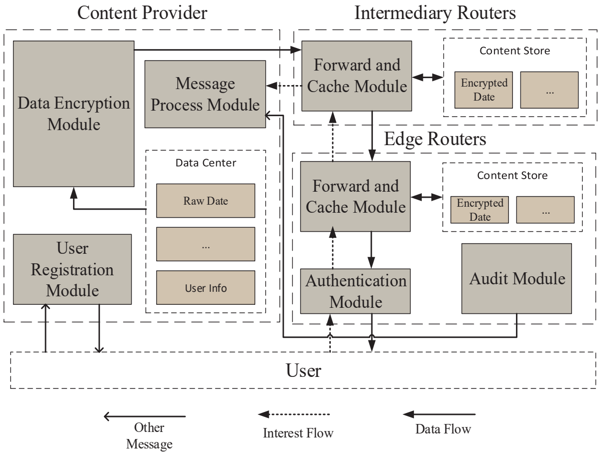


Fig. 2. System overview.

To reduce the burden of edge routers, we make authentication procedure more efficient with the help of hash chain. As we all know, when a user requests a file, he/she will request a serial of chunks continuously. Benefiting from the one-way property of hash chain, edge routers can authenticate users by signature for the first request of a file and utilize hash chain to authenticate subsequent requests of the same file.

B. Scheme Description

Step 1 **System Initialization:** Before all of other operations, the content provider conducts system initialization as follows:

- Generating a bilinear map group system $S = (q, G_1, G_2, e(.,.))$.
- Randomly choosing two elements $P, Q \in G_1$ and computing $Z = e(P, Q)$. Selecting m random numbers $\alpha_1, \alpha_2, ..., \alpha_m \in Z_q^*$ and setting $v_i = \alpha_i \cdot Q$, where $i = 1, 2, ..., m$. Denoting V as $(v_1, v_2, ..., v_m)$.
- Publishing the system public parameters as: $(S, P, V, Z, H_1, H_2, Enc(\cdot))$. Here H_1 is a one-way hash function: $\{0,1\}^* \rightarrow Z_q^*$; H_2 is a standard hash algorithm like SHA-256; and $Enc_k(\cdot)$ is a secure symmetric encryption algorithm with symmetric secret key k.

Step 2 **User Registration:** When user i who has limited access times registers to a CP, the CP generates the user's Privilege Mask msk_i and sets his/her maximum access times n_i according to user's identity ID_i. Then CP randomly selects a number $s_i \in Z_q^*$ and other few numbers $x_i^k \in Z_q^*$, where $k = 1, 2, ..., m$, and computes

$$b_i^k = \begin{cases} 0 & msk_i[k-1] = 0 \\ \dfrac{1}{\alpha_k + x_i^k} P & msk_i[k-1] = 1 \end{cases},$$

$$c_i^k = \begin{cases} 0 & msk_i[k-1] = 0 \\ \dfrac{x_i^k}{\alpha_k + x_i^k} Q & msk_i[k-1] = 1 \end{cases}.$$

After that, the CP needs to select n_i random numbers $(a_1^i, a_2^i, ..., a_{n_i}^i)$ and computes $y_l^i = a_l^i \cdot P$, where $l = 1, 2, ..., n_i$. Denote A_i, B_i, C_i, Y_i as $(a_1^i, a_2^i, ..., a_{n_i}^i)$ and the set of b_i^k, c_i^k, y_l^i, respectively. After the registration, the user gets his/her own secret key (A_i, B_i, C_i, s_i) and maximum access times n_i, where B_i, C_i are used for content decryption and A_i, s_i are used in signature generation.

If user i is an unlimited user, the CP sets his/her maximum access times n_i as a negative value and sets A_i,Y_i empty. The rest of user registration is the same as limited users.

At last, the CP stores user i's information uif_i as $(ID_i, msk_i, etime_i, n_i, Y_i, \Gamma_i = s_i \cdot P)$ for further use, where $etime_i$ represents the expired time of user i's privilege.

Step 3 **Content Publication:** To preserve the data confidentiality, the CP must encrypt the contents. For a content M whose GID is d, the CP needs to choose a random number $\beta \in Z_q^*$ and compute

$$C_1 = \beta \cdot v_d, C_2 = \beta \cdot P, K = Z^\beta, C = Enc_K(M).$$

Finally, the CP publishes the content M as (C_1, C_2, C).

Step 4 **Content Request and Authentiaction:**

(1) Request phase

Limited users: After registration, the first thing user i needs to do is to construct a polynomial $f(x)$ of degree n_i: $f(x) = s_i + a_1^i x + a_2^i x^2 + ... + a_{n_i}^i x^{n_i}$ $(mod\ q)$ using his/her secret key s_i, A_i. Every time user i wants to get a new file, he/she needs to generate a hash chain with proper length. Denote the last element in the hash chain as H_{final}. When user i arrives at a new place and connects to a new edge router, he/she is required to send an authentication request with his/her identity ID_i. Once the edge router receives this, it checks whether the user's information uif_i is in its cache. If not, the edge router will ask corresponding CP for uif_i. Then it selects a random number $\zeta \in Z_q^*$ and sends it to user i.

After user i generates signature σ according to his/her maximum access times n_i, ζ and his/her secret keys, user i sends an interest packet with σ and the last element in the hash chain H_{final} piggybacked. The details in signature generation are shown in Algorithm 1. This signature generation algorithm can only be used n_i times at most during a period because of the polynomial in signature, otherwise the user will leave his/her secret key s_i to the risk of leakage. When the period passes, the user can ask CP for a new A_i and then he/she can use this algorithm n_i times again in the next period.

Algorithm 1: Signature Generation

```
Input: User's maximum access times $n_i$, user's identity $ID_i$,
       user's secret keys $s_i$, polynomial $f(x)$, random number
       $\zeta$, system parameters $(S, P)$.
Output: A valid signature.
Choose a random number $r_i \in Z_q^*$;
Generate a timestamp $ts$;
$R_i = r_i \cdot P$, $M_i = ID_i || ts$;
if $n_i > 0$ then
    $f(\zeta) = s_i + a_1^i \zeta + a_2^i \zeta^2 + ... + a_{n_i}^i \zeta^{n_i}$ $(mod\ q)$;
    $\sigma_1 = r_i + f(\zeta) H_1(M_i) s_i$;
    $\zeta = H_1(\zeta)$;
    return $\sigma = (\sigma_1, f(\zeta), R_i, M_i)$;
else
    $\sigma_1 = r_i + H_1(M_i) s_i$;
    return $\sigma = (\sigma_1, R_i, M_i)$;
end
```

Unlimited users: If user i has unlimited access times, he/she doesn't need to construct the polynomial and just needs to generate a hash chain for continuous authentication. The rest of request phase is similar with limited users but there is no polynomial participating in signature generation process.

(2) Authentication phase

After receiving user i's interest packet, the edge router first checks the timestamp ts in signature σ and then finds out whether the user has right to access the chunk according to msk_i and $etime_i$ in uif_i. Finally, it verifies signature σ by using Algorithm 2.

If all of them successfully verified, the edge router has adequate reasons to believe that this request is legitimate and then transmits this interest packet to the core network. Otherwise, the edge router drops this interest packet. In order to make it convenient to authenticate subsequent chunk requests, the edge router maintains a table to store the file name f, and the latest received hash value H_{latest}, which is set as H_{final} at first.

When user i requests subsequent chunks of the same file, he/she needs to send an interest packet with the previous element H_{pre} of the latest sent hash value in the hash chain piggypacked. The edge router just needs to check whether there exists the tuple $(f, H_2(H_{pre}))$ in the table. Owing to the one-way property of hash chain, the user who can provide the right hash element can only be the user i authenticated before.

When the edge router doesn't receive requests from user i for a while, it deletes the corresponding item in the table and store $(\zeta, f(\zeta))$, ID_i and n_i, for further audit.

Algorithm 2: Signature Verification

```
Input: User's signature $\sigma$, user's information $uif_i$, random
       number $\zeta$.
Output: Valid or Invalid.
$X_1 = \sigma_1 \cdot P$;
if $n_i > 0$ then
    $X_2 = R_i + f(\zeta) H_1(M_i) \Gamma_i$;
    $X_3 = \Gamma_i + \zeta \cdot Y_i[0] + \zeta^2 \cdot Y_i[1] + ... + \zeta^{n_i} \cdot Y_i[n_i - 1]$;
    $X_4 = f(\zeta) \cdot P$, $\zeta = H_1(\zeta)$;
    if $X_1 == X_2$ && $X_3 == X_4$ then
        return Valid;
    else
        return Invalid;
    end
else
    $X_2 = R_i + H_1(M_i) \Gamma_i$;
    if $X_1 == X_2$ then
        return Valid;
    else
        return Invalid;
    end
end
```

(3) Audit phase

ISP aggregates signatures stored and checks whether there are dishonest users regularly. If there are, ISP computes their secret keys and deliveries them to the CP. At last, the CP punishes dishonest users, e.g. reduces their subsequent service time or even forbids them to get service forever. In order to enhance ISP's positivity in auditing, CPs can offer ISP extra financial reward.

Step 5 **Content Decryption:** After user i obtains the encrypted content (C_1, C_2, C), he/she needs to choose the right decryption key according to the content's GID. If the GID equals d, user i recovers corresponding symmetric secret key $\widetilde{K}$ as following:

$$\begin{aligned} \widetilde{K} &= e(C_1, b_i^d) e(C_2, c_i^d) \\ &= e(\beta \cdot v_d, \frac{1}{\alpha_d + x_i^d} P) e(\beta \cdot P, \frac{x_i^d}{\alpha_d + x_i^d} Q) \\ &= e(P, Q)^\beta = Z^\beta. \end{aligned}$$

Finally, user i can decrypt the received content using $Enc(\cdot)$ and recovered secret key $\widetilde{K}$.

VI. Security and Performance analysis

A. Security analysis

1) Signature Unforgeability: Given some signatures σ and the corresponding user information uif_i, it is impossible for any attacker to forge a valid signature in our system.

If an adversary wants to forge an unlimited user's signature, he must obtain the user's secret keys s_i. There are only two ways to get it. Computing $s_i = (\sigma_i - r_i)H_1(M_i)^{-1}$ based on signature σ is one possible choice. This means he/she must compute r_i through R_i, which contradicts **DLP**. Moreover, guessing $s_i \in Z_q^*$ is the other method. When q is large enough, the probability of success can be negligible. Besides, there is no reason for an attacker to forge a limited user's signature. The polynomial value $f(\zeta)$ included in the signature of a limited user brings many obstacles to the attacker. To forge a signature, he/she must compute s_i and all the elements in A_i through σ and Y_i in uif_i, which is much harder than forging a signature of an unlimited user. Thus, the signature unforgeability can be ensured in our system.

2) Data Confidentiality: The routers in ICN can not decrypt the ciphertext they store in our system.

The contents are stored as (C_1, C_2, C) in the routers in ICN, where $C_1 = \beta \cdot v_d, C_2 = \beta \cdot P, C = Enc_K(M)$. Supposing the routers can get the plaintext of the content they store, they have to try to compute the symmetric key K through C_1, C_2. In other word, given C_1, C_2, P, they can get the symmetric key $K = Z^\beta$ by computing $e(C_1, P) = e(\beta\alpha_d \cdot Q, P)^{\frac{1}{\alpha_d}} = e(P,Q)^\beta = K$, where β is unknown to the routers. It entirely contradicts **WBHE Assumption**. Hence, the data confidentiality can be guaranteed.

3) Auditability: Any user who is dishonest with his/her maximum access times can be audited in our system.

Supposing user i tries to cheat on access times, which means edge routers receive at least $n_i + 1$ signatures. In other word, ISP owns $(\zeta_1, f(\zeta_1)), (\zeta_2, f(\zeta_2)), ..., (\zeta_{n_i+1}, f(\zeta_{n_i+1}))$. Then ISP reconstructs the polynomial by calculating the Lagrangian Interpolation Polynomial $L_{n_i}(x)$ if degree n_i. ISP can get user i's secret key s_i by computing $L_{n_i}(0)$ and send (ID_i, s_i) to corresponding CP. The CP verifies the correctness of the audit message by checking whether user i's secret key is s_i. So our system is auditable.

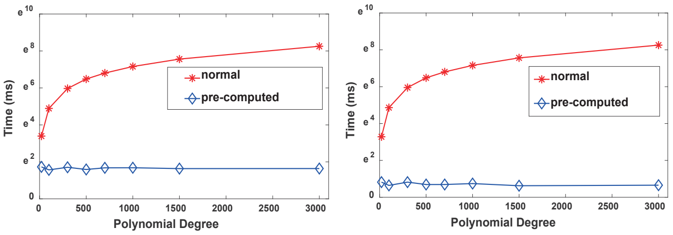
4) DoS attacks resistance: Our scheme can effectively resist DoS attacks launched by legitimate users or illegitimate users.

Each edge router needs to check whether requests are legitimate before forwarding them to the core network. Thus, a large number of requests sent by illegitimate users will be dropped by edger routers and are unable to get into the core network to consume the resource in ICN. Though DoS attacks launched by legitimate users cannot be perceived at the first time, ICN can quickly find the edge routers where attacks take place [5] and ease the damage by decreasing the interest packets sending rate of them. Besides, corresponding CPs can finally get attackers' information from ISP and give them severe punishment.

B. Performance analysis

1) Algorithm implementation: We implement the algorithm in our scheme by using GNU Multi-Precision Arithmetic library, Pairing-Based Cryptography library and OpenSSL library. All the experiments are conducted on the Ubuntu 16.04 LTS with a 3.6GHz Intel Core i7 processor and 20G RAM.

For unlimited users, the signature generation and verification time are about 1.874ms and 2.469ms on average respectively. Fig. 3 represents the signature generation and verification time of limited users for different polynomial degree. When there is no pre-computation, the time cost increases as polynomial degree grows linearly. But the signature generation and verification time will decrease largely with the help of pre-computation and they are about 2.042ms and 5.231ms on average.



(a) Signature Generation Time (b) Signature Verification Time

Fig. 3. Signing and Verification Time for Different Polynomial Degrees.

Fig. 4 illustrates the encryption and decryption time for different chunk sizes. The CP needs to encrypt symmetric key through broadcast encryption and then encrypts the contents using symmetric encryption. The users are required to decrypt symmetric key before getting the contents they want. The time cost of broadcast encryption and decryption has no relationship with chunk size and are approximately 2.855ms and 1.356ms on average. The encryption and decryption time of AES256 increases as chunk size grows, which are 4.098ms per MB and 4.222ms per MB respectively.

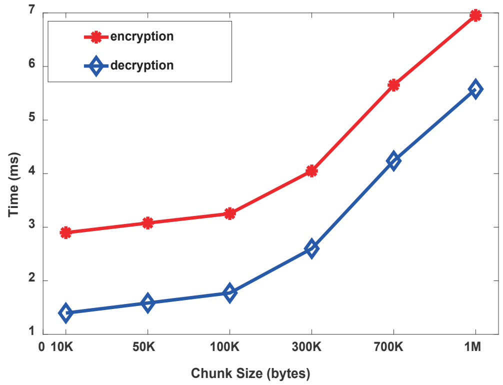


Fig. 4. Symmetric Encryption and Decryption Time.

2) Network simulation: We simulate complete LASA and incomplete LASA, where edge routers are required to authenticate each request by signature, to compare with basic NDN by using ndn-SIM 2.3. The topologies are created by BRITE using two-layer Top-Down hierarchical model. The topology for our simulation has 1300 nodes and 2377 edges. There are 4 edge routers in the topology and each edge router serves 5 users through a link with 20Mbps bandwidth and 5ms delay. The links between intermediary routers in the core network have 1Gbps bandwidth and 10ms delay.

CP in ICN is randomly selected and it publishes files with different sizes: 10MB, 100MB, 300MB, 700MB and 1GB. The chunk size we use in the simulation is 1MB. The users request contents with right prefix and always request successive chunks of the same file.

Fig. 5 shows the file retrieval delay for different file sizes. In our simulation, basic NDN brings the lowest delay. When the network is equipped with LASA, it also performs well if the file size is not so big. As the file size grows, the difference between LASA and basic NDN becomes clear because more chunks means authentication and decryption will conducted more times. Benefiting from the use of hash chain, complete LASA shows its advantage in such situation compared with incomplete LASA. When the file size is 1GB, incomplete LASA introduce about 10s more delay than basic NDN, while complete LASA just increases about 4s delay. Overall, the gap between the basic NDN and LASA is acceptable, especially when we run the complete LASA.

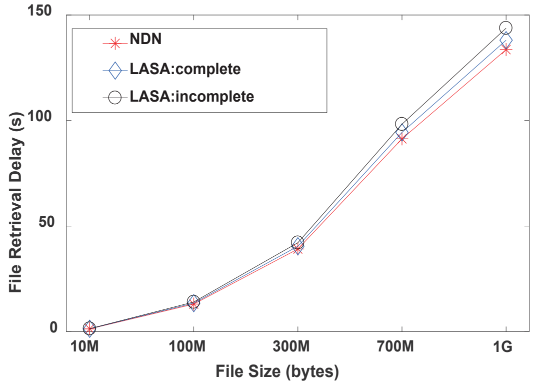


Fig. 5. File Retrieval Delay for Different File Sizes

VII. Conclusion

We have proposed a lightweight, auditable and secure access control scheme in ICN to meet access times limitation requirement. Our scheme not only achieves basic access control on edge-side but also provides a solution for CPs to limit users' access ability. In our scheme, content providers can set maximum access times for each user, and ISP can easily detect whether users have exceeded access times limit with the audit mechanism. Besides, we let edge routers take charge of authentication and largely decrease the authentication overhead through hash chain. The experimental evaluation proves that our scheme performs efficiently, which only introduces acceptable delay in file retrieval.

VIII. Acknowledgment

The authors sincerely thank the anonymous referees for their invaluable suggestions that have led to the present improved version of the original manuscript. This work is supported in part by the National Key R&D Program of China under Grant No. 2017YFB0801702 and No. 2016YFB0800301, the National Natural Science Foundation of China under Grant No. 61671420, Youth Innovation Promotion Association of CAS under Grant No. 2016394, and the Fundamental Research Funds for the Central Universities.

References

[1] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard, "Networking named content," in *Proceedings of the 5th international conference on Emerging networking experiments and technologies (CoNEXT)*. ACM, 2009, pp. 1–12.

[2] Q. Zheng, G. Wang, R. Ravindran, and A. Azgin, "Achieving secure and scalable data access control in information-centric networking," in *Proceedings of 2015 IEEE International Conference on Communications (ICC)*. IEEE, 2015, pp. 5367–5373.

[3] S. Misra, R. Tourani, F. Natividad, T. Mick, N. E. Majd, and H. Huang, "AccConF: An access control framework for leveraging in-network cached data in the ICN-enabled wireless edge," *IEEE Transactions on Dependable and Secure Computing, online*, 2017.

[4] B. Li, D. Huang, Z. Wang, and Y. Zhu, "Attribute-based access control for ICN naming scheme," *IEEE Transactions on Dependable and Secure Computing, online*, 2016.

[5] Y. Xin, Y. Li, W. Wang, W. Li, and X. Chen, "A novel interest flooding attacks detection and countermeasure scheme in NDN," in *Proceedings of 2016 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2016, pp. 1–7.

[6] N. Fotiou, G. F. Marias, and G. C. Polyzos, "Access control enforcement delegation for information-centric networking architectures," in *Proceedings of the second edition of the ICN workshop on Information-centric networking (ICN)*. ACM, 2012, pp. 85–90.

[7] Q. Li, X. Zhang, Q. Zheng, R. Sandhu, and X. Fu, "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.

[8] E. G. AbdAllah, M. Zulkernine, and H. S. Hassanein, "DACPI: A decentralized access control protocol for information centric networking," in *Proceedings of 2016 IEEE International Conference on Communications (ICC)*. IEEE, 2016, pp. 1–6.

[9] Z. Chen, S. Li, Q. Huang, Y. Wang, and S. Zhou, "A restricted proxy re-encryption with keyword search for fine-grained data access control in cloud storage," *Concurrency and Computation: Practice and Experience*, vol. 28, no. 10, pp. 2858–2876, 2016.

[10] Q. Li, P. P. Lee, P. Zhang, P. Su, L. He, and K. Ren, "Capability-based security enforcement in named data networking," *IEEE/ACM Transactions on Networking*, vol. 25, no. 5, pp. 2719–2730, 2017.

[11] K. Xue, J. Hong, Y. Xue, and et al., "CABE: A new comparable attribute-based encryption construction with 0-encoding and 1-encoding," *IEEE Transactions on Computers*, vol. 66, no. 9, pp. 1491–1503, 2017.

[12] W. Li, K. Xue, Y. Xue, and J. Hong, "TMACS: A robust and verifiable threshold multi-authority access control system in public cloud storage," *IEEE Transactions on parallel and distributed systems*, vol. 27, no. 5, pp. 1484–1496, 2016.

[13] J. Ning, Z. Cao, X. Dong, K. Liang, H. Ma, and L. Wei, "Auditable σ-time outsourced attribute-based encryption for access control in cloud computing," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 1, pp. 94–105, 2018.

[14] T. H. Yuen, J. K. Liu, M. H. Au, X. Huang, W. Susilo, and J. Zhou, "k-times attribute-based anonymous access control for cloud computing," *IEEE Transactions on Computers*, vol. 64, no. 9, pp. 2595–2608, 2015.

[15] D. He, J. Bu, S. Zhu, S. Chan, and C. Chen, "Distributed access control with privacy support in wireless sensor networks," *IEEE Transactions on wireless communications*, vol. 10, no. 10, pp. 3472–3481, 2011.

[16] H. Wang and Q. Li, "Distributed user access control in sensor networks," in *Proceedings of 2016 International Conference on Distributed Computing in Sensor Systems (DCOSS)*. Springer, 2006, pp. 305–320.