ATTRIBUTE-BASED ACCOUNTABLE ACCESS CONTROL FOR MULTIMEDIA CONTENT WITH IN-NETWORK CACHING

Peixuan He[1], Kaiping Xue[1,4], Jie Xu[1], Qiudong Xia[1], Jianqing Liu[2], Hao Yue[3]

[1]Department of EEIS, University of Science and Technology of China, Hefei, Anhui 230027 China
[2]Department of ECE, University of Alabama in Huntsville, Huntsville, AL 35899 USA
[3]Department of Computer Science, San Francisco State University, San Francisco, CA 94132 USA
[4]kpxue@ustc.edu.cn (corresponding author)

ABSTRACT

Nowadays, multimedia content retrieval has become the major service requirement of the Internet and the traffic of these contents has dominated the IP traffic. To reduce the duplicated traffic and improve the performance of distributing massive volumes of multimedia contents, in-network caching has been proposed recently. However, because in-network content caching can be directly utilized to respond users' requests, multimedia content retrieval is beyond content providers' control and makes it hard for them to implement access control and service accounting. In this paper, we propose an attribute-based accountable access control scheme for multimedia content distribution while making the best of in-network caching, in which content providers can be fully offline. In our scheme, the attribute-based encryption at multimedia content provider side and access policy based authentication at the edge router side jointly ensure the secure access control, which is also efficient both in space and time. Besides, secure service accounting is implemented by letting edge routers collect service credentials generated during users' request process. Through the informal security analysis, we prove the security of our scheme. Simulation results demonstrate that our scheme is efficient with acceptable overhead.

Index Terms— Multimedia content security, Access control, In-network caching, Attribute-based encryption, Access policy based authentication.

1. INTRODUCTION

It is reported by Cisco that globally, multimedia traffic will be 82% of all IP traffic by 2022, up from 75% in 2017 [1]. Such massive volumes of multimedia contents bring a huge amount of redundant traffic and cause insufficient bandwidth utilization in traditional IP network architecture. To solve these problems and make the Internet more scalable, various content-centric network architectures have been proposed such as information centric networking [2], most of which are equipped with in-network caching capabilities and route by content name.

In-network caching is an excellent choice for multimedia content providers (MCPs) to enhance users' experience and make their content services more competitive. With in-network caching, when users request contents, the nearby routers that cache one corresponding copy will respond them directly instead of fetching the contents from remote MCPs. In traditional IP architecture, what MCPs concern most is access control and service accounting. On one hand, MCPs do not expect users to access their repertories without permissions. On the other hand, MCPs need to collect feedback information (e.g., the amount of a certain content requested) to let them make targeted marketing plans and improve their services. However, when introducing in-network caching, many requests may not reach MCPs as users can access the copies in cache-enabled routers without MCPs' consents or even awareness. Therefore, how to allow MCPs to conduct access control and service accounting for in-network cached content becomes an interesting but challenging problem.

Many schemes have been proposed to solve the first part of the problem, i.e., access control. Some of them achieve effective access control by restricting users' abilities in decrypting the contents. More specifically, to take full advantage of in-network caching, MCPs utilize some cryptographic techniques to encrypt the published multimedia contents and also make sure the encrypted caching can be obtained by different users with different secret keys. Attribute-based encryption (ABE) [3–5] and broadcast encryption [6, 7] are such cryptographic techniques which are widely used to achieve fine-grained access control over in-network caching. Nevertheless, these schemes are vulnerable to denial-of-service (DoS) attacks and network resources are easily exhausted by malicious requests because of indiscriminate services provided by

This work is supported in part by the National Key R&D Program of China under Grant No. 2017YFB0801702, the National Natural Science Foundation of China under Grant No. 61379129 and No. 61631017, the Key Research Program of of the Chinese Academy of Sciences (CAS) under Grant No. ZDRW-KT-2016-2-5, Youth Innovation Promotion Association of CAS under Grant No. 2016394, and the Fundamental Research Funds for the Central Universities.

cache-enabled routers. In light of it, some of existing schemes let some entities (e.g. MCPs, cache-enabled routers or extra servers) authenticate users' requests before responding them and only authorized users can get the contents [8–10]. However, an increased computation and communication overhead is incurred by these solutions due to the frequent interactions. Moreover, as for now, little attention has been devoted to the second part of the problem, i.e., service accounting requirement of MCPs.

Motivated by above observations, in this paper, we propose an attribute-based accountable access control scheme over in-network multimedia content caching. In our scheme, MCPs encrypt original content using ciphertext policy-attribute based encryption (CP-ABE) before content publication. To release MCPs from being always online and thwart DoS attacks, edge routers are empowered to authenticate users' requests so that only legitimate requests are allowed to access the cached contents. Note that adopting the existing content-based authentication in edge routers is not feasible since it brings significant storage overhead to edge routers. Therefore, we propose an access policy-based challenge response authentication mechanism and edge routers can only store the outsourced keys every MCP generates for every access policy. To reduce the computation overhead in edge routers, hash chain is utilized to authenticate users' continuous requests. Besides, users' responses in challenge response phase and hash chains are collected by edge routers as service credentials to make it possible for MCPs to do service accounting. In the meanwhile, MCPs will pay Internet Service Provider (ISP) according to the amount of requests served, considering ISP helps MCPs cache content copies in cache-enabled routers and collect feedback information. Our contributions can be summarized as follows:

- We propose an efficient, secure and accountable access control scheme for multimedia content with in-network caching by combining CP-ABE with authentication on edge routers. The edge-side authentication is based on access policy, which largely decreases the storage overhead on edge routers, and MCPs can be offline during the authentication.
- We further design a solution to enable MCPs to implement secure service accounting. MCPs extracts useful information from service credentials collected by edge routers and the embedded users' signatures are able to prevent edge routers from forging credentials for more profits.

2. SYSTEM MODEL, SECURITY ASSUMPTION AND PRELIMINARIES

2.1. System Model

Our system model mainly consists of three components: Multimedia Content Providers, Internet Service Provider and Users. Fig. 1 illustrates the construction of our system.

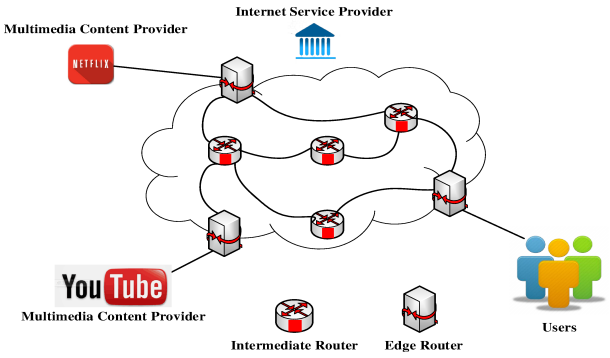

Fig. 1. System Model

MCPs provide users with multimedia content subscription service and assign secret keys for registered users. ISP manages the core network that is capable in in-network caching and routing by content name like ICN. ISP provides network access service to both MCPs and users. The routers in the network can be categorized as intermediate routers and edge routers. Intermediate routers are cache-enabled and only focused on forwarding and caching whereas edge routers do not cache the contents they transmit but they need to store the outsourced keys received from different MCPs and authenticate users' requests at the very beginning. Users consume multimedia contents by sending requests with corresponding content names.

2.2. Security Assumption

In our system, we assume MCPs are trusted and they are responsible for the security of the multimedia contents they own. ISP is assumed to be semi-trusted. On one hand, it is curious about the contents routers stored and wants to learn something from them. On the other hand, ISP is greedy but rational because it wants to increase its profit by lying about the amount of served requests. However, in practice, usually ISP is generally a prestigious company which concerns about its own credibility so if the detection possibility of misbehavior is non-negligible (e.g. 1%), we assume ISP will not take the risk of trying to cheat MCPs but will otherwise. Users are considered to be untrusted and they always want to obtain unaccessible contents.

2.3. Preliminaries

Definition 1: Bilinear Map. Let $\mathbb{G}, \mathbb{G}_T$ be two multiplicative cyclic groups of the same order q. A bilinear map can be described as $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$, which has the several properties: 1) *Computability*: there exists an efficient algorithm to compute $e(u, v)$ for any $u, v \in \mathbb{G}$. 2) *Bilinearity*: for all $a, b \in \mathbb{Z}_q$ and $u, v \in \mathbb{G}$, we have $e(u^a, v^b) = e(u, v)^{ab}$. 3) *Non-degeneracy*: $e(g, g) \neq 1$, where g is the generator of $\mathbb{G}$.

Definition 2: Discrete Logarithm Problem (DLP). The DLP is given $g, g' \in \mathbb{G}$ to compute $x \in \mathbb{Z}_q$ satisfying $g' = g^x$.

3. CONSTRUCTION OF OUR SCHEME

3.1. Overview

In our proposed scheme, MCPs encrypt the contents using CP-ABE before publication to make sure unauthorized users cannot get the original contents from encrypted caching. However, encryption by itself is not sufficient to keep the system robust. Because intermediate routers suppose to respond all users' requests, illegitimate requests can easily exhaust resources of the core network. In light of it, edge routers should authenticate users' requests at the very beginning through an access policy based challenge response mechanism. Specifically, MCP generates an outsourced key for every access policy used. According to the access policy of contents users request, edge routers utilize corresponding outsourced key to generate challenges. Only users who are accessible to the contents can return correct responses and only their requests will be routed to the intermediate routers. In such way, authentication is run by edge routers so that MCP can be offline. Besides, network resources are protected from DoS attacks. Moreover, we utilize hash chain technology to associate continuous requests with their former ones and replace the complex challenge response based authentication to a lightweight hash authentication for subsequent requests to reduce the authentication overhead.

For simplicity, we only consider the scenario where MCP transcodes multimedia contents into a multimedia stream at a constant bit rate before encryption, but our scheme is inherently compatible with existing adaptive streaming protocols such as DASH [11].

3.2. System Initialization

First MCP generates a bilinear map group system $S = (q, \mathbb{G}, \mathbb{G}_T, e(\cdot,\cdot))$, where g is a generator of $\mathbb{G}$. Then it randomly chooses two numbers $\alpha, a \in \mathbb{Z}_q$ and computes $w = g^\alpha, y = e(g,g)^\alpha, z = g^a$. For each attribute $Att_i \in \mathcal{U}$, MCP selects a random element $h_i \in \mathbb{G}$ as its public key, where $\mathcal{U}$ is the universal set of attributes MCP manages and $i = 1, \ldots, |\mathcal{U}|$. Finally, MCP publishes the public parameters as: $PK = \{S, g, y, z, h_1, \ldots, h_{|\mathcal{U}|}, H, H_1, Enc(\cdot)\}$. Here, H is a standard hash function like SHA-256, H_1 is a one-way hash function:$\{0,1\}^* \to \mathbb{G}$ and $Enc_k(\cdot)$ is a secure symmetric encryption algorithm with the secret key k. In the meanwhile, MCP stores the master key (w, a) securely.

3.3. User Registration

When user i with the identity ID_i registers to MCP, MCP assigns corresponding attributes according to his/her behavior. For example, if the user logins the system, MCP assigns attribute "Member" to him/her. Denote the set of attributes the user assigned as S_i. Then MCP selects a random number $t \in \mathbb{Z}_q$ and computes the secret key SK_i for user i as follows:

$$K = wz^t, L = g^t; \forall x \in S_i, K_x = h_x^t.$$

Besides, MCP chooses a random number $r_i \in \mathbb{Z}_q$ as a private key and computes its corresponding public $PK_i = g^{r_i}$, which are used to generate signatures for further usage. Finally, MCP sends the keys (SK_i, r_i) to the user after successful registration.

3.4. Key Outsourcing

We let edge routers authenticate users' requests using secret keys granted from MCP so that it allows MCP to be offline during the authentication. Specifically, before multimedia content service, for each access policy $\mathbb{A}_i$ used, MCP chooses a random element $\delta_i \in \mathbb{G}_T$ and generates the outsourced key $H(\delta_i)$. Then it encrypts the random element δ_i using corresponding access policy $\mathbb{A}_i$ as follows:

According to the method proposed in [12], MCP can turn the access policy into an LSSS access structure $(\mathbb{M}, \rho)$, where ρ is a function which maps the rows of $\mathbb{M}$ to attributes. Let $\mathbb{M}$ be an $l \times n$ matrix. It randomly generates a vector $\vec{v} = \{s, y_2, \ldots, y_n\} \in \mathbb{Z}_q^n$, where the values will be used to share the encryption exponent s secretly. MCP computes $v_i = \vec{v} \cdot \mathbb{M}_i$ for $i = 1, \ldots, l$, where $\mathbb{M}_i$ is the i-th row of the matrix $\mathbb{M}$. Then it selects l random numbers $\gamma_1, \ldots, \gamma_l$ and computes the ciphertext of δ_i as follows:

$$C = \delta_i \cdot y^s, C' = g^s, C_i = z^{v_i} h_{\rho(i)}^{-\gamma_i}, D_i = g^{\gamma_i}, \forall i \in [1, l].$$

We denote the ciphertext of δ_i under the access policy $\mathbb{A}_i$ as $ABE_{\mathbb{A}_i}(\delta_i)$. Finally, MCP broadcasts the outsourced key related information $\Lambda = \{H(\mathbb{A}_i), ABE_{\mathbb{A}_i}(\delta_i), H(\delta_i)\}_{i=1}^m$ through the broadcast channel, supposing there exits m different access policies. The edge routers store Λ in the outsourced key list as Table 1 for further authentication.

Table 1. Outsourced Key List

Access Policy	Ciphertext	Outsourced Key
$H(\mathbb{A}_1)$	$ABE_{\mathbb{A}_1}(\delta_1)$	$H(\delta_1)$
$H(\mathbb{A}_2)$	$ABE_{\mathbb{A}_2}(\delta_2)$	$H(\delta_2)$
...	...	...

3.5. Content Generation

To make it convenient for edge routers to identify the access policy of a multimedia content, MCP needs to add the related information in the name of the content. For example, if the original name of a multimedia content is */youtube/moive/xxx/chunk_1* and its access policy is *Member and No Less Than 18 Years Old*, MCP is required to computes the hash value $hv = H(\text{“}Member\ and\ No\ Less\ Than\ 18\ Years\ Old\text{''})$ and modifies the name of the content as */youtube/moive/xxx/hv/chunk_1*.

Before MCP responds a request, it transcodes the original multimedia content into a media stream at the predesigned bit rate and encrypts the stream M using symmetric encryption by random chosen symmetric key k: $CT = Enc_k(M)$. Then it encrypts the symmetric key following the process mentioned in Section 3.4 under an access policy $\mathbb{A}_i$ to obtain the ciphertext $CT' = ABE_{\mathbb{A}_i}(k)$. To this end, the multimedia stream M is stored as (CT, CT') in the intermediary routers.

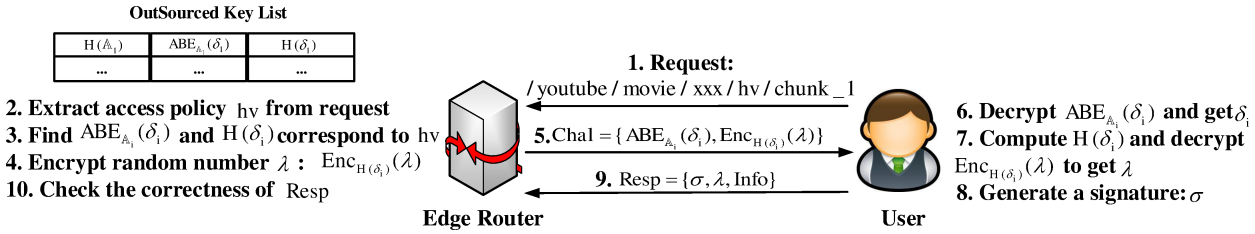

Fig. 2. Example of Request Authentication Process

3.6. Request Authentication

Unlike other existing schemes, which route users' requests to the core network without proper access control, in our scheme only the requests authenticated by edge routers are allowed to enter into the core network.

Here, edge routers conduct an access policy based challenge response to authenticate users' requests, which is illustrated by Fig. 2. Specifically, when an edge router receives the request for a specific chunk in a multimedia file from user j for the first time, it firstly extracts the hash value of access policy hv from the content name in the request. Then edge router finds the corresponding item $< H(\mathbb{A}_i), ABE_{\mathbb{A}_i}(\delta_i), H(\delta_i) >$ satisfying $hv = H(\mathbb{A}_i)$ in the outsourced key list. Moreover, it selects a random number λ and encrypts it using the outsourced key $H(\delta_i)$ as $CS = Enc_{k'}(\lambda)$, where $k' = H(\delta_i)$. Afterwards, the edge router sends the challenge $chal = \{ABE_{\mathbb{A}_i}(\delta_i), CS\}$ to the user.

After receiving the challenge, the user whose attribute set satisfies the LSSS access structure $(\mathbb{M}, \rho)$ corresponding to access policy $\mathbb{A}_i$ is able to decrypt $ABE_{\mathbb{A}_i}(\delta_i)$ to get δ_i. Suppose user j is authorized, let $\mathbb{M}_{ID_j}$ be a sub-matrix of $\mathbb{M}$, where each row of $\mathbb{M}_{ID_j}$ represents a specific attribute in user j's attribute set S_j. Denote I as $I = \{i : \rho(i) \in S_j\}$, where $I \subset \{1, 2, \cdots, l\}$. It is easy for the user to find a set of constants $\{w_i \in \mathbb{Z}_q\}_{i \in I}$ such that $\Sigma_{i \in I} w_i v_i = s$, where $\{v_i\}$ is the set of valid shares of secret s, by computing $\vec{w} = (1, 0, \ldots, 0) \cdot M_{ID_j}^{-1}$. Through the parameters $\{w_i \in \mathbb{Z}_q\}_{i \in I}$, the user further computes:

$$\frac{e(C', K)}{\Pi_{i \in I}(e(C_i, L)e(D_i, K_{\rho(i)}))^{w_i}} = \frac{y^s e(g, g)^{ast}}{\Pi_{i \in I} e(g, g)^{tav_i w_i}} = y^s.$$

Thus, user j can obtain δ_i by computing $\delta_i = C/y^s$. With the value δ_i, the user can recover the outsourced key $k' = H(\delta_i)$ and decrypt CS to obtain the plaintext λ. Finally, user j generates a hash chain with proper length, of which the first and last element we denote as H_f, H_l, respectively. In the meanwhile, a BLS signature is generated $\sigma = H_1(\lambda || Info)^{r_j}$. Here $Info$ is useful feedback information which contains timestamp TS, multimedia content name CN, user identity ID_j and the last element of hash chain H_l. Then the user sends the response $resp = \{\sigma, \lambda, Info\}$ to the edge router.

Suppose the current time is TS', the edge router first checks whether $TS' - TS > \Delta T$ for allowed time interval ΔT after receiving the response. Then it checks the correctness of λ and verifies signature by checking whether $e(PK_j, H_1(\lambda || Info))$ equals $e(g, \sigma)$. If true, the edge router forwards the request to core network; otherwise, it aborts the request. Then it maintains a table to store multimedia file of name f, latest received hash value H_{lt} and a counter len, where f is a prefix of the content name, H_{lt} and len are set as H_l and "1" respectively at first.

With the help of hash chain, it makes convenient for user j to request subsequent chunks of the same multimedia file without the complex challenge response mechanism. For these subsequent chunks, the user just needs to use the hash chain in reverse order and sends the request with the new element H_{new} of the hash chain piggypacked. Owing to the one-way property of hash chain, if the edge router can finds the item satisfying $H_{lt} = H(H_{new})$ with the same multimedia file name f, it can assure that user j is an authorized user and forwards the request to core network. In the meanwhile, the edge router updates the H_{lt} in the table as H_{new} and increases the corresponding counter.

After user terminates the request, the edge router stores $< \sigma, \lambda, Info, H_{lt} >$ as a service credential for further service accounting.

3.7. Content Decryption

When user j obtains the encrypted multimedia stream (CT, CT'), he/she recovers the symmetric key k using the secret key SK_j by following decryption method mentioned in Section 3.6. With the recovered symmetric key, user j can finally decrypt CT' and get the final plaintext M. The original multimedia content is obtained by decoding M.

3.8. Service Accounting

To help MCP learn the exact amount of requests that are served and other useful feedback information, edge routers should send service credentials to MCP regularly and MCP stores them in the ascending order of timestamp to check whether there exits duplicate credentials. For every credential MCP receives, it needs to check whether $H^{len-1}(H_{lt}) = H_f$. Then it verifies the validity of σ. To reduce the large overhead caused by the massive number of service credentials, MCP only chooses a small proportion of received cre-

dentials to verify. In the meanwhile, MCP can further reduce the overhead by conducting batch verification. Suppose $\{\sigma_i, \lambda_i, Info_i, H_{lt,i}\}_{i=1}^n$ is the set of credentials MCP chooses to verify, it can verify the validity of signatures in them by checking: $\Pi_{i=1}^n e(PK_i, H_1(\lambda_i||Info_i)) \stackrel{?}{=} e(g, \Pi_{i=1}^n \sigma_i)$.

If the verification passes, MCP pays ISP according the sum of len in all the credentials, i.e., the amount of served requests. Then MCP can analyze the feedback information $Info_i$ to get more information like users' preference and improve its content service.

4. SECURITY AND PERFORMANCE ANALYSIS

4.1. Security Analysis

1) Data Confidentiality: The data confidentiality of our scheme relies on CP-ABE, which can withstand the attack launched by malicious users and ISP either individually or in collusion.

2) Security against DoS Attack: To resist DoS attack launched for exhausting core network resources, edge routers enforce an access policy based challenge response authentication. If a user wants to request a content whose access policy is $\mathbb{A}$, he/she must compute a correct response from the challenge received from edge routers by following these steps: $\delta \leftarrow ABE_{\mathbb{A}}(\delta), k' \leftarrow H(\delta), \lambda \leftarrow Enc_{k'}(\lambda)$. Edge routers only propagate the requests from the users who compute right λ, because only the authorized users who are granted access to contents have the λ. Thus, edge routers can easily distinguish legitimate and illegitimate requests at the very beginning of the service and at the edge of the core network to prevent the system from DoS attack.

3) Security against Replay Attack: Since in the authentication process a timestamp TS is included in the user's response, replay attacks can be easily detected by checking whether the time interval $TS - TS'$ is allowed, where TS' is the time edge routers receive the response.

4) Service Accountability: To make it capable for MCPs to know the condition of contents requested (e.g., the amount and distribution of requests) and improve the service quality MCPs provide, edge routers send service credentials $< \sigma, \lambda, Info, H_{lt}, len >$ to MCPs. Because MCPs pay ISP according to the amount of requests served, edge routers may risk forging some credentials. Suppose edge routers are able to forge credentials, they must be able to compute r_i given $\sigma, \lambda, Info$, which contradicts the intractability of DLP. Nevertheless, though edge routers are unable to forge legitimate credentials, they can also submit some faked credentials which contain invalid signatures. In our scheme, MCPs conduct probabilistic verification on σ in credentials. Suppose MCPs' check rate is β and edge routers only submit one faked credential every time when MCPs receive n credentials, then the detection possibility of misbehavior is:

$$p = 1 - \binom{n-1}{n\beta} / \binom{n}{n\beta} = \beta,$$

which is a non-negligible possibility for a reputed company. Besides, edge routers dare not modify the length of hash chain and submit duplicate credentials for the detection possibility of these behavior is 100%. Finally, secure service accountability can be protected.

4.2. Performance Analysis

To evaluate the performance of our scheme, we implement our scheme in named data network (NDN) simulated by ndnSIM, which is a typical network architecture with in-network caching and name-based routing. All the simulations are conducted on the Ubuntu 16.04 LTS with a 3.6GHZ Inter Core i7 processor and 20G RAM by using Pairing-Based Cryptography library with type-A curve and OpenSSL library.

The network topology we use has 600 nodes. The links between any two routers have bandwidth selected randomly from 1 to 5 Gbps and delay selected randomly from 1 to 5 ms. The number of user nodes is 20% of the number of routers. Each user node connects an edge router through a link with 100 Mbps bandwidth and 1 ms delay, distributed uniformly in the ASs. The intermediate routers are equipped with cache space for 200 chunks and LRU cache policy. MCP is located centrally in the network and able to respond to every request containing its prefix. Each user requests the chunks from the same multimedia file continuously and does not request the next chunk until the current request is satisfied.

First of all, we measure the average file retrieval delay increment (i.e., multimedia file retrieval delay divides the number of chunks of the file) of our scheme and clumsy scheme (i.e., edge routers authenticate every request using challenge response authentication mechanism proposed) with standard NDN which is not equipped with our access control scheme.

Fig. 3 illustrates the average chunk retrieval delay increment under different number of attributes. In this simulation, we set the file size of contents users request to be 100 MB and chunk size is 1 MB. As shown in Fig. 3, our scheme only brings a little overhead. Even when the number of attributes is 9, our scheme only increases no more than 1% file retrieval delay, which indicates that our scheme is efficient enough. However, for clumsy scheme, due to the huge overhead brought by challenge response authentication, the increased delay is linear to the number of attributes. Therefore, lightweight hash authentication is helpful to reduce the overhead of our system.

Fig. 4 depicts the result of average chunk retrieval delay increment with different file sizes. The attributes and chunk size in this simulation is 6 MB and 1 MB respectively. From Fig. 4, we can see that the performance of clumsy scheme has no relation with file size and the increased delay maintains a high level about 25.7%. However, with larger file size, users need to send more requests and more hash authentication can be used to balance the overhead of challenge response. Thus, the file retrieval delay increment of our scheme decreas-

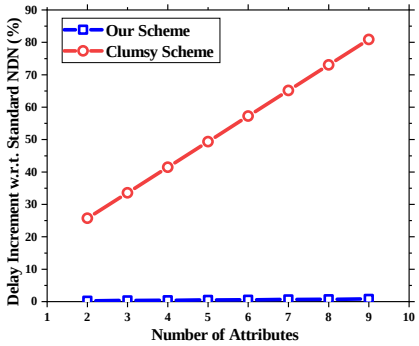

Fig. 3. Average Chunk Retrieval Delay Increment vs. Attribute Number

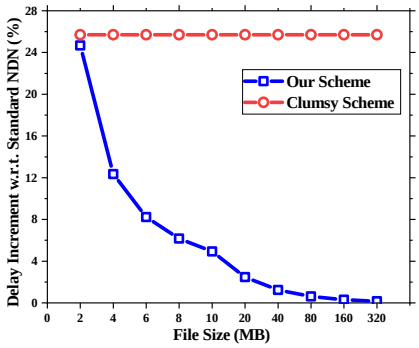

Fig. 4. Average Chunk Retrieval Delay Increment vs. File Size

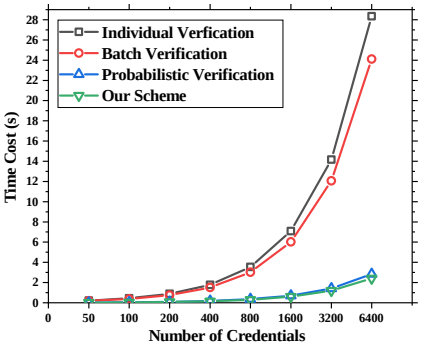

Fig. 5. Verification Time Cost vs. Number of Credentials

es rapidly as the file size becomes large. When the file size is 40 MB, the increased delay reduces to about 1%, which is an acceptable overhead.

Then we evaluate the performance of the method used to verify the validity of received service credentials. To reduce the overhead of verifying service credentials in MCPs, in our scheme, MCPs choose a small portion of the received service credentials to conduct batch verification. In the simulation, the check probability is set as 10%. Compared with individual verification, batch verification and probabilistic verification, our scheme achieves the best performance. Specifically, when MCPs receive 6,400 service credentials, they can finish verification within 2.5s, which is efficient enough for MCPs.

5. CONCLUSION

In this paper, we propose a solution to provide secure access control and service accounting to multimedia content providers in an in-network caching context. Specifically, we achieve secure access control by combing CP-ABE and edge-side authentication, which makes our system more robust and efficient. The computation overhead of authentication is largely reduced with the design of hash chain while access policy based authentication helps cut down the storage overhead in edge routers. The service credentials consisting of users' signatures and hash chain are collected by edge routers to enable MCPs to implement secure service accounting. Our security and performance analysis shows that our scheme has several security features and is efficient enough.

6. REFERENCES

[1] "VNI," https://www.cisco.com/c/m/en_us/solutions/service-provider/vni-forecast-highlights.html.

[2] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher, and B. Ohlman, "A survey of information-centric networking," *IEEE Communications Magazine*, vol. 50, no. 7, 2012.

[3] C. Ma and C. Chen, "Secure media sharing in the cloud: Two-dimensional-scalable access control and comprehensive key management," in *IEEE ICME*, 2014.

[4] C. Ma, Z. Yan, and C. Chen, "Scalable access control for privacy-aware media sharing," *IEEE Transactions on Multimedia*, 2018.

[5] B. Li, D. Huang, Z. Wang, and Y. Zhu, "Attribute-based access control for ICN naming scheme," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 2, pp. 194–206, 2018.

[6] 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*, 2017.

[7] K. Xue, X. Zhang, Q. Xia, D. S. Wei, H. Yue, and F. Wu, "SEAF: A secure, efficient and accountable access control framework for information centric networking," in *IEEE INFOCOM*, 2018, pp. 2213–2221.

[8] 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.

[9] 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.

[10] E. G. AbdAllah, M. Zulkernine, and H. S. Hassanein, "DACPI: A decentralized access control protocol for information centric networking," in *IEEE ICC*, 2016.

[11] K. T. Bagci, K. E. Sahin, and A. M. Tekalp, "Compete or collaborate: Architectures for collaborative DASH video over future networks," *IEEE Transactions on Multimedia*, vol. 19, no. 10, pp. 2152–2165, 2017.

[12] J. Li, K. Ren, and K. Kim, "A2BE: Accountable attribute-based encryption for abuse free access control," *IACR Cryptology ePrint Archive*, vol. 2009, pp. 118, 2009.