Optimized Access Control Enforcement Over Encrypted Content in Information-centric Networks

Elisa Mannes
Federal University of Paraná
Curitiba, Brazil
elisam@inf.ufpr.br

Carlos Maziero
Fed. Univ. of Technology - Paraná
Curitiba, Brazil
maziero@utfpr.edu.br

Luiz Lassance
CONFESOL
Florianópolis, Brazil
luiz@confesol.com.br

Fábio Borges
Technische Universität Darmstadt
Darmstadt, Germany
fabio.borges@cased.de

***Abstract*—The Information-centric Network (ICN) paradigm is an important initiative toward an Internet architecture more suitable for content distribution. The change it imposes by naming, routing, and forwarding content directly on the network layer empowers the architecture with several interesting characteristics, such as in-network caching. As contents are meaningful for different users, they can be opportunistically cached and easily accessed by them, which improves content delivery and user experience. However, the fact that users can retrieve content through caches without interacting with the content provider raises security concerns regarding unauthorized access and the enforcement of access control policies. In this context, we propose an access control solution for ICN by adapting and optimizing a proxy re-encryption scheme, reducing up to 33% the processing time. The proposed solution is perfectly aligned with ICN demands, simultaneously ensuring content protection against unauthorized access of contents retrieved from unrestricted in-network caches as well as access control policies enforcement for legitimate users.**

I. Introduction

The Information-centric Network (ICN) [1] paradigm has gained considerable attention from both academia and industry in recent years. It aims to overcome current Internet shortcomings by changing the main network entity from hosts to named content, thus routing and forwarding named content directly on the network layer. This new paradigm favors very special features, such as traffic aggregation, content-location independence, and in-network caching, which evolve the current Internet architecture for a natural environment to content distribution. However, the ICN paradigm also changes many aspects related to the network security. For example, naming content modifies the security paradigm from securing hosts, links, or sessions to securing content, requiring that users could assess authenticity and integrity directly from the named content [2]. Moreover, the deployment of in-network caches results in content being retrieved from anywhere by anyone, arising new challenges concerning ***unauthorized access*** and ***access control policies enforcement***, since it is no longer necessary to connect to a specific server to retrieve contents.

The access control issue is particularly worrisome for protected or copyrighted content distribution, e.g., movies, music, and software, in which access is linked with payments and user's compliance with strict rules such as age, type of subscription, and location. Thus, disclosing such kind of content on the Internet without the ability to enforce access control policies would be damaging for content providers, and it is highly unlikely that the ICN architecture would be adopted under such terms. Thereby, there is an obvious need to enforce access control on content retrieved from caches, while allowing the cache to be broadly used. Most of current solutions on access control for ICN fails on providing an effective way of protecting such content, as they encrypt each content with a distinct secret key and focus on guaranteeing only authorized users have access to such secret keys [3]–[9]. While these solutions allow content in cache to be useful for different users, the content protection is easily compromised if a malicious user discloses the secret key, since the key related to a specific content is the same for all users.

In this paper, we propose an access control enforcement solution for encrypted content in ICN aiming three main properties: (i) content can be cached anywhere and retrieved by anyone; (ii) no entities are added to the network for access control enforcement and policy verification; (iii) users possessing the content cannot decrypt it unless allowed by the content provider. To meet these requirements, we adapt and optimize a proxy re-encryption (PRE) cryptographic scheme and employ it in an access control solution for ICN. In the proposed solution, we remove the proxy entity and place its functions on the user side. Thus, the content is encrypted by the provider and decrypted by any user having a re-encryption key, issued by the content provider. Cache satisfies any request for contents regardless the user, while our solution guarantees access control policies enforcement by the content provider, since it is mandatory for users to request the provider for a re-encryption key to decrypt the content. Although content access control would be desirable for most of contents on the Internet, we tackle the specificities of very popular contents, such as videos, music, and software updates, where caching will work on their full potential and the benefits from ICN emerge in a more substantial way. We go further than [10] by improving the ideas and optimizing the algorithms.

This paper is structured as follows. Section II describes current efforts for access control in ICN and discusses their main shortcomings and challenges. Section III details the PRE scheme, which bases our solution. Section IV describes our assumptions regarding the network, content distribution, and threat models, and details an optimization on the PRE scheme. Section V analyzes the computational suitability of our solution. Section VI discusses the solution regarding performance, security, and suitability for ICN architectures. Section VII provides final remarks and suggests future works.

II. Related work

This section presents related works regarding content access control in information-centric network as well as explains

978-1-4673-7194-0/15/$31.00 ©2015 IEEE

the fundamentals of proxy re-encryption schemes.

A. Access control in Information-Centric Networks

The in-network caching infrastructure introduced by the ICN paradigm represents a great challenge for content access control. As the content is cached along the path by unreliable entities such as routers, mobile devices, or third party servers (as in a CDN-like infrastructure), content providers face problems to manage and enforce access control to their content [3]. Restricting the knowledge of content names only to authorized users is not sufficient, because content names can be easily discovered [11]. The basic approach to control the access to content in ICN is through content encryption, ensuring that only users having a valid key can access it [12]. However, the cryptographic scheme should be carefully chosen, as some of them may hinder the caching (like traditional symmetric key cryptographic schemes, in which a content encrypted to a given user may not be useful to any other user [13]).

Different cryptographic approaches have been explored in access control solutions for ICN, such as attribute-based encryption [3], [5], [14] and broadcast encryption [4]. The idea of such solutions is to create groups of users to share keys for decrypting content, consequently optimizing caching over the network. This same concept is used in solutions by [8], [15], but using the traditional public-private key cryptosystem. Although such solutions allow users to share cached contents, they become vulnerable if malicious or compromised users discloses the keys, as any unauthorized user on the network retrieving content from caches may use these keys. Trying to overpass such limitations, [16] uses two layers of symmetric encryption. However, it introduces overhead to the system as well as demands modification on ICN behavior.

Proxy re-encryption schemes have already been applied in the context of access control. The works from [17] and [18] are applied to access control on content in clouds, thus considering that the content providers have control over the content storage and are able to revoke access at any time (which is a difficult assumption in ICN). Parallel to our work, [9] explores the proxy re-encryption scheme for access control in ICN. However, the analysis conducted by the authors lead them to propose the use of symmetric cryptography to encrypt the content and the use of re-encryption to protect symmetric keys, which incurs the same deficiencies previously identified. Thus, it is observed that a solution to provide access control in ICN is not a trivial task and requires alignment of multiple objectives, particularly with respect to data security and content delivery performance through caches.

B. Proxy Re-Encryption

The basic idea of standard proxy re-encryption (PRE) schemes is to transform a message m encrypted with the public key of user A (and thus decryptable using A's private key) into a message decryptable using the private key of another user, B, without exposing the content m nor the corresponding private keys. This transformation takes place on a semi-trusted delegated *proxy*. User A authorizes a proxy to transform her ciphertexts to user B by giving it a *re-encryption key* $rk_{A \to B}$. They assume that the proxy does not learn the plaintext it re-encrypts nor the private keys of users A and B. These characteristics makes proxy re-encryption schemes suitable for many applications, such as encrypted e-mail forwarding, secure distributed file systems, and outsourced encrypted spam filtering. Several proxy re-encryption solutions have been explored in the literature, focusing on very different attractive properties. For our solution, we employ the efficient unidirectional proxy re-encryption scheme proposed by [19], detailed below.

III. The Efficient Unidirectional Proxy Re-encryption Scheme

The Efficient Unidirectional Proxy Re-encryption (EU-PRE) scheme proposed by [19] aims at a simple design, short ciphertexts and computational efficiency. In order to achieve such goals, the authors propose a PRE scheme that does not rely on pairings and is secure against adaptive chosen-ciphertext attack under the computational Diffie-Hellman assumption. The EU-PRE scheme is based on ElGamal encryption and Schnorr signature, and applies the token-controlled encryption technique to "hide" the delegator secret key. Next, we detail the six algorithms used in EU-PRE, from [19].

Setup: choose primes p and q such that $q \mid p-1$, message m of size ℓ_0, security parameter ℓ_1 and a generator g for the group $\mathbb{G}$ of order q. In addition, four hash functions are used: $H_1 : \{0,1\}^{\ell_0} \times \{0,1\}^{\ell_1} \to \mathbb{Z}_q^*$, $H_2 : G \to \{0,1\}^{\ell_0+\ell_1}$, $H_3 : \{0,1\}^* \to \mathbb{Z}_q^*$ and $H_4 : G \to \mathbb{Z}_q^*$.

Key generation: secret keys $kv_{i,1}$ and $kv_{i,2}$ are chosen randomly from $\mathbb{Z}_q^*$ and public keys are set by g^{kv}, thus $kp_{i,1} = g^{kv_{i,1}} \mod p$ and $kp_{i,2} = g^{kv_{i,2}} \mod p$.

Encryption: choose random u from $\mathbb{Z}_q^*$, random w of size l_1 and calculate $r = H_1(m, \omega)$ and D, E and F as follows

$$D = (kp_{P,1}^{H_4(kp_{P,2})} kp_{P,2})^u \mod p \tag{1}$$

$$E = (kp_{P,1}^{H_4(kp_{P,2})} kp_{P,2})^r \mod p \tag{2}$$

$$F = H_2(g^r \mod p) \oplus (m||\omega) \tag{3}$$

and $s = u + r \cdot H_3(D, E, F) \mod q$. Outputs (D, E, F, s).

Re-encryption key generation: choose a random h of size l_0 and π of size l_1 and calculate $v = H_1(h \times \pi)$. Calculate $V = kp_{U,2}^v \mod p$ and $W = H_2(g^v \mod p) \oplus (h||\pi)$. The re-encryption key is

$$rk_{P \to U} = h\,(X) \mod p, \tag{4}$$

where

$$X = (kv_{P,1} H_4(kp_{P,2}) + kv_{P,2})^{-1} \mod p-1. \tag{5}$$

The output is $(rk_{P \to U}, V, W)$.

Re-encryption: validate $((kp_{P,1}^{H_4(kp_{P,2})} \mod p) kp_{P,2} \mod p)^s \mod p = D \cdot (E^{H_3(D,E,F)} \mod p) \mod p$. If the equality holds, compute

$$E' = E^{rk_{P \to U}} \mod p \tag{6}$$

and outputs (E', F, V, W).

Decryption: recover $(h||\pi)$ and $(m||\omega)$ by calculating

$$(h||\pi) = W \oplus H_2(V^{kv_{U,2}^{-1} \mod p-1} \mod p) \tag{7}$$

$$(m||\omega) = F \oplus H_2(E'^{h^{-1} \mod p-1} \mod p) \tag{8}$$

Outputs m if $V = kp_{U,2}^{H_1(h,\pi)} \mod p$ and $E' = g^{H_1(m,\omega) \cdot h} \mod p$.

IV. An optimized access control solution for ICN

Different from previous approaches, we aim to propose an access control solution completely aligned with ICN purposes, thus ensuring different users could benefit from content on any cache over the network while protecting it against unauthorized access. In addition, it would be desirable neither to introduce extra entities for the access control procedure nor to modify the core functions of any ICN architecture, keeping the process simple while following the ICN specifications. In light of such goals, we adapt the EU-PRE scheme in two distinct aspects: (i) we eliminate the proxy entity, by allocating proxy functions directly on the user side, and (ii) we optimize re-encryption and decryption algorithms for better performance. In our model, the content provider encrypts each content with a distinct public key. Either the content provider or in-network caches can satisfy requests for these contents, as usual. However, in order to decrypt the content, users must request a corresponding re-encryption key to the content provider, thus allowing the content provider to enforce access control policies on users. Next subsections introduce the system model and detail the proposed access control solution.

A. System model

Our assumptions are divided into a three-layer model composed by the network model, content distribution model, and threat model. These models are discussed below.

Network model: we assume the particularities of Named Data Network architecture (NDN) [12], however, as the proposed solution does not imply any change in the ICN paradigm, it can be applied to any ICN architecture. In the NDN architecture, each content is composed by a set of 4Kb *chunks* [20], which are individually named, and the name-content binding is cryptographically signed, as proposed in [2]. To request a content, users send an `Interest` packet and the network returns a `Data` packet. Content names from the same content provider may share common prefixes, which are aggregated in routing table for performance. Routers are in charge of routing and forwarding content requests, as well as optionally storing content chunks in internal caches according to adopted caching policies. After receiving an `Interest` packet, the router checks its cache and promptly replies, in case the content is stored on its cache. Otherwise, the router checks its pending interest table (PIT). If an interface is waiting for the same content, the router aggregates the requests by appending the incoming interface to the entry. If no identical request is found on PIT, the router adds a new entry and consults its forwarding information base to send the request toward the content provider. The `Data` packet with content follows the request path back to the user, consuming PIT entries on routers.

Content distribution model: content providers offer contents to users upon subscription of a service, and demand legitimate users to be properly registered and logged into the application to have access to controlled content catalog. Such application can be previously loaded on devices or available for installation upon request. Content chunks are stored in content providers' servers or in a third-party CDN infrastructure; to access the content, the user should use the content provider application with her credentials. The request follows the NDN routing specification and can be satisfied by the content provider itself or by any cache in the network. The content provider should validate the user (check her identity and public key) to certify she is a legitimate user of the service and to grant access to contents based on its own policies.

Threat model: we assume that the content provider behaves correctly, i.e., does not distribute private content or decryption rights to unauthorized users. On the other hand, routers follow a *honest but curious* adversary model, in which they correctly perform their functions, but may be curious and try to access content passing through it. Malicious entities may be illegitimate users without access to the content provider service or legitimate users trying to access content for which they have no authorization. Their intention is to gain access to content without the burden assigned to authorized users, such as payment, personal data checking, or different types of accounts. They can exploit protected content learning the content name by either eavesdropping communication paths from nearby users or by snooping, or yet by probing nearby caches. Furthermore, malicious entities can retrieve re-encryption keys from caches as well. Moreover, as users have access to content through specific applications, there is no need to discover the name of the content beforehand or by any untrusted method.

B. The proposed solution

Our solution is structured in three domains: *content provider*, *network*, and *user* domains. The content provider domain covers content encryption and re-encryption key generation. The network domain comprises content distribution on the ICN paradigm. The user domain is composed by re-encryption and decryption. Fig. 1 illustrates operations in each domain. Next, we detail these operations. From now on, we refer to EU-PRE as EU-RE (Efficient Unidirectional *Re-encryption*), to emphasize the absence of proxies in our solution.

Key pair generation: Either content providers or a third party PKI generate and distribute public-private key pairs for content providers and users. The Key Generation algorithm generates two key sets of public-private keys. This feature is introduced by [19] to guarantee the content provider private key is not disclosed in case of proxy and user collusion and is extremely important in our case, since we deliberately allocate the proxy function in the user domain.

Content encryption: The content provider has a set $\mathcal{C}$ of contents that it wishes to make available to users. Each content $c_i \in \mathcal{C}$ has a distinct public key kp_{c_i}. Chunks from the same content are individually encrypted (step 1) with the content public key. The corresponding private key, kv_{c_i}, is kept secret by the content provider. The encrypted content is distributed as users request for it, and is properly cached in the network according to caching policies. Notice that the content can be everywhere in the network, but as it is encrypted with a public key whose corresponding private key is only known by the content provider, no one is able to access it other than the content provider. Thus, both legitimate users and malicious entities that retrieve the content from cache or intercept a communication cannot access the content at this stage.

Re-encryption key generation: Before the application can access the content, it must be decrypted. Thus, a legitimate user U wanting to decrypt content c_i must request the re-encryption key $rk_{c_i \to U}$. Then, U sends an `Interest` packet requesting the re-encryption key for content c_i to the content

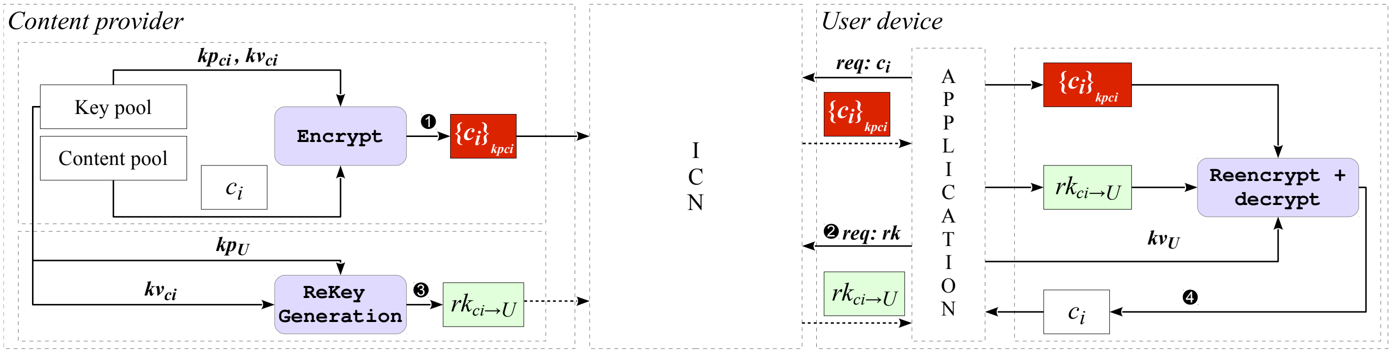

Fig. 1. Access control policy enforcement for ICN: operation overview

provider (step 2). The content provider checks if the user is allowed to access c_i, identifies the private key related to the content, calculates $rk_{c_i\to U}$ (step 3) and sends a `Data` packet containing the re-encryption key. This means that the user U is the only one able to use $rk_{c_i\to U}$ to decrypt content c_i, since it requires user U private key kv_U (unless user U also discloses his own public-private key pair). It is useless to a potential malicious entity to gather the content and the re-encryption key by eavesdropping or by sniffing caches: it may be able to re-encrypt the content, but the resulting ciphertext can only be decrypted with the intended user private key.

CONTENT DECRYPTION: Upon receiving the content c_i (or content chunks) and the re-encryption key $rk_{c_i\to U}$, user U is able to decrypt c_i. Using the re-encryption key $rk_{c_i\to U}$, the content c_i is decrypted with the user's private key kv_U and consumed by the application (step 4). The re-encryption key is exclusive for user U and the corresponding content c_i. This implies that every user receives its own re-encryption key from the content provider for each content on demand, thus the content provider can deny the re-encryption key in case of abuse or non-compliance with stated requirements.

RE-ENCRYPTION KEY REVOCATION: Once the user U has the re-encryption key $rk_{c_i\to U}$ for content c_i, she is able to decrypt c_i whenever she wants. Although keys are stored inside the proprietary application and, in normal conditions, not accessible to users, it is still important to deal with re-encryption key revocation. Furthermore, any content signed with the public key used to encrypt the content c_i could be opened by U with $rk_{c_i\to U}$, and is the main reason why it is mandatory for each content to have a distinct public-private key pair, making it difficult to revoke a re-encryption key once the user has access to it. Thus, a natural way to revoke and invalidate a re-encryption key is by renewing content encryption with a different public-private key pair in a time basis. This would generate different re-encryption keys for each content and demand users to ask for new re-encryption keys whenever they want to access the content.

C. EU-RE optimization

The allocation of proxy functions in the user domain allows the optimization of re-encryption and decryption operations without introducing new information to the system. Given Eqs. (6) (re-encryption) and (8) (decryption), we can simplify the re-encryption operation directly on the decryption phase. Notice that originally E' is computed in Eq. (6) and used in Eq. (8) to recover message m. However, it is possible for the users to apply E directly on Eq. (8). Substituting E' for E and applying $rk_{P\to U}$ directly on Eq. (8), we have

$$(m||\omega) = F \oplus H_2\left(E^{\frac{rk_{P\to U}}{h}} \mod p\right) \mod p, \quad (9)$$

which returns m. Thus, we eliminate Eq. (6) computation and remove h from Eq. 8, as the optimized exponent is

$$N = \frac{rk_{P\to U}}{h} = \frac{1}{kv_{P,1}H_4(kp_{P,2}) + kv_{P,2}} \mod p-1. \quad (10)$$

As each modular exponentiation has exponent close to p, their time complexity is $O(\log p)$, i.e., each modular exponentiation needs $\log_2(p)$ interactions. Re-encryption and decryption functions originally compute three modular exponentiations and two inversions, while their optimized version computes two modular exponentiations and two modular inversions, which may be computed by modular exponentiation. Hence, these functions require $5\log_2(p)$ and $4\log_2(p)$ interactions for original and optimized versions, respectively. Consequently, the optimized version is $1/5$, i.e., 20% faster than the original. In addition, as it does not require the introduction of new assumptions into the system, it does not introduce vulnerabilities.

V. EVALUATION

We aim to validate the computational suitability of original and optimized EU-RE versions for our access control solution. We implemented[1] the six algorithms from EU-RE: SETUP, KEYGEN, ENCRYPT, DECRYPT, REKEYGEN, and REENCRYPT. Table I lists the parameters used in the validation[2].

TABLE I
PARAMETERS USED ON EU-RE VALIDATION

Parameter	Value
Key size (k)	1024, 2048, and 3072 bits
Message size (ℓ_0)	0.5, 1, 2, 4, 8, 16, 32, and 64 KB
Security parameter (ℓ_1)	160 *bits*
Hash functions H_1, H_3, H_4	mod q
Hash function H_2	mod $2^{(\ell_0+\ell_1)}$

[1] Codes are available at *http://www.inf.ufpr.br/elisam/proxy*.

[2] We implemented simple hash functions only for validation purposes, thus, we do not assess their security here.

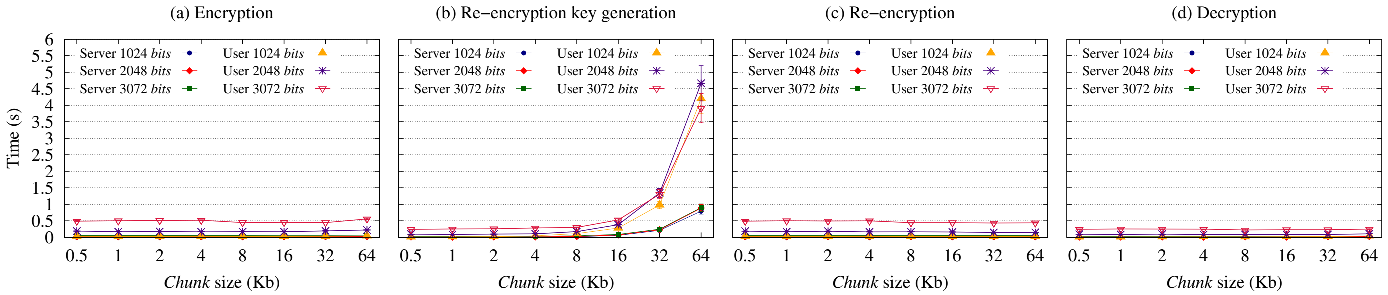

Fig. 2. Results from the validation of original EU-RE on server (scenario A) and user device (scenario B)

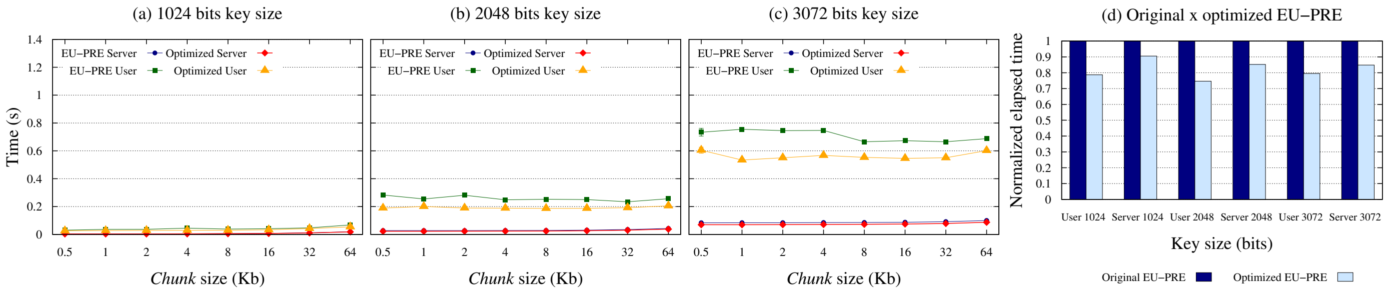

Fig. 3. Original *versus* optimized EU-RE performance validation for re-encryption and decryption operations on scenarios A and B

The results were benchmarked on a Linux Mint 17 Qiana 64 bits server, AMD Opteron Processor 6136 2.4GHz, 86GB RAM, representing a content provider (scenario A), and on an Ubuntu 13.10 32 bits Sony Vaio laptop, Core 2 Duo 1.66GHz and 2GB RAM, representing a user device (scenario B). We measured the time to perform encryption, re-encryption key generation, re-encryption, and decryption operations. Although we are interested in 4Kb chunk sizes [20], we used different chunk sizes to better assess the behavior of EU-RE, as well as different key sizes. The results are the average of 100 executions, with confidence interval of 95%. As SETUP and KEYGEN algorithms can optionally be executed in a public key infrastructure, we do not account key generation and distribution costs in this stage.

Fig. 2 shows results for the original EU-RE operations. From the results, we conclude that EU-RE operations are suitable for employment in an access control solution for ICN. For example, content provider's operations of encryption and re-encryption key generation perform below 100ms on server scenario, as well as user's operations of re-encryption and decryption on the user scenario. We notice that re-encryption key generation time is high due to the pseudo-random number generator for h, which has the size of the chunk. This may be optimized by adopting symmetric cryptography for pseudo-random number generation as in [21]. We clarify that although Fig. 2(b) costs are bigger than 4 seconds, such results are related to the user scenario, while these operations are exclusively processed by content providers (results from both scenarios for all operations are plotted for completeness). Fig. 3(a-c)[3] shows a comparison of processing time between original and optimized versions of EU-RE for decryption plus re-encryption operations. The proposed optimization results in a reduction up to 33% of processing time with average of 18%, which is very close to the theoretical value found on complexity analysis (20%). The average processing time reduction is better observed on Fig. 3(d). As these operations are executed in the user side, the reduction represents an expressive gain for the solution, valuable in resource-constrained devices.

VI. DISCUSSION

The main goal of the proposed solution is to provide a way for content providers to enforce access control policies on protected content under the ICN paradigm. The main obstacles to achieve such property came from two intrinsic characteristics of ICN: (i) the implementation of named content and (ii) ubiquitous cache. While named content makes it easier for malicious users to identify available content on the network, the cache decentralizes content distribution, thus users no longer have to retrieve content directly from the content provider; instead, the request may be satisfied by a nearby cache. Our solution tackles the latter issue by distributing encrypted content and using a modified version of a proxy re-encryption scheme to allow only authorized users to decrypt it. Next, we present considerations about performance, security, and suitability of our solution in light of the ICN paradigm.

Performance: from the perspective of the content provider domain, there is the onus of encrypting content and computing re-encryption keys. Results obtained with EU-RE validation show that the extra load imposed by re-encryption key creation does not seem to impact on the content provider performance, as the re-encryption keys are supposed to be created on demand. However, the amount of users and contents may influence the content provider performance, as it is closely related to the amount of re-encryption keys that it manages. Besides, we assume that content providers can overcome performance issues by employing load-balancing strategies. At the user domain, the issue is the extra task of re-encrypting the content

[3] Notice that we used a different Y-axis scale from Fig. 2 for a better visualization of EU-RE optimization behavior.

before decrypting it. The proposed optimization involving re-encryption and decryption operations on user domain provides a substantial improvement in user side processing times. Another important issue relates to the periodical change of content encryption with different keys to provide re-encryption keys revocation. While this mechanism may be useful to avoid brute force attacks on contents due to a substantially large sampling set of contents encrypted with the same key, the key management overhead of our access control scheme is being considered for further investigation.

Security: using a public-private key encryption scheme instead of a symmetric encryption one, as used in previous solutions for access control in ICN [3]–[9], improves overall security in the sense that it makes more difficult for unauthorized users to access protected content. For example, in symmetric key solutions it is sufficient for malicious entities to intercept the secret key to have access to the content. In our solution, it is necessary to disclose both private and re-encryption keys from the same authorized user. Even so, only content related to the disclosed re-encryption key would be accessed. Nevertheless, the content provider can simultaneously implement measures that restrict the number of players accessing a given content set under the same credentials, making it even less likely that users disclose their keys, at risk of being penalized by the content provider. Although the original EU-PRE scheme assumes that proxy are semi-trusted entities, proxy functions are placed at the user side into the application, thus the user has no incentives to behave maliciously on proxy functions.

Suitability for ICN architectures: our solution does not imply any change in the ICN architecture, since only content providers and users are involved in encryption and decryption actions. As the network is not loaded with specific requirements, it is free to route and forward packets to whoever requests it, in the best possible way. As the re-encryption key has size p, it fits in one chunk and does not introduce overhead to the network. In addition, no security functions are transferred to network elements: routers do not have to check keys nor enforce access policies. However, the process of key revocation implies some drawbacks: for a time window, the old content can be accessed by users holding the old corresponding re-encryption key, if they retrieve the content from cache. An idea to solve this issue is to incorporate a timestamp into the content name and to make the application aware of new names, ensuring that the application asks for the current timestamp. This issue is being considered for further investigation.

VII. Conclusion

In this work, we proposed an access control enforcement solution for ICN architectures. Each content is encrypted with a distinct public key, maximizing cache use as the content satisfies requests from any user. In order to decrypt the content, the user has to request the content provider for a re-encryption key, tied with the user's public key. Thus, the content provider has an active access control over the content. Even in case a malicious user retrieves the protected content and the re-encryption key, it still cannot access the content, as the private key associated with the re-encryption key is required to decrypt the content. Simulation shows that the proposed solution is suitable for ICN and that the proposed optimization improves processing times up to 33%. Future work consists on refining and investigating the re-encryption key revocation phase.

References

[1] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher, and B. Ohlman, "A survey of information-centric networking," *IEEE Communications Magazine*, vol. 50, no. 7, pp. 26–36, 2012.

[2] D. Smetters and V. Jacobson, "Securing network content," PARC TR-2009-1, Tech. Rep., 2009.

[3] M. Ion, J. Zhang, and E. Schooler, "Toward content-centric privacy in ICN: attribute-based encryption and routing," in *3rd ACM SIGCOMM Works. on Information-centric networking (ICN '13)*, 2013, pp. 39–40.

[4] S. Misra, R. Tourani, and N. E. Majd, "Secure content delivery in information-centric networks: design, implementation, and analyses," in *3rd ACM SIGCOMM workshop on Information-centric networking (ICN '13)*, 2013, pp. 73–78.

[5] J. P. Papanis, S. I. Papapanagiotou, A. S. Mousas, G. V. Lioudakis, D. I. Kaklamani, and I. S. Venieris, "On the use of attribute-based encryption for multimedia content protection over information-centric networks," *Trans. on Emerging Telecommunications Technologies*, pp. 1–14, 2013.

[6] N. Fotiou, G. F. Marias, and G. C. Polyzos, "Access control enforcement delegation for information-centric networking architectures," in *2nd Works. on Information-centric networking (ICN '12)*, 2012, pp. 85–90.

[7] S. Singh, A. Puri, S. S. Singh, A. Vaish, and S. Venkatesan, "A trust based approach for secure access control in information centric network," *International Journal of Information and Network Security (IJINS)*, vol. 1, no. 2, pp. 97–104, 2012.

[8] B. Hamdane, M. Msahli, A. Serhrouchni, and S. El Fatmi, "Data-based access control in named data networking," in *9th International Collaborative Computing: Networking, Applications and Worksharing (Collaboratecom '13)*, Oct 2013, pp. 531–536.

[9] C. Wood and E. Uzun, "Flexible end-to-end content security in CCN," in *IEEE Consumer Communications and Networking Conference*, ser. CCNC '14, 2014, pp. 1–8.

[10] E. Mannes, C. Maziero, L. C. Lassance, and F. Borges, "Controle de acesso baseado em reencriptação por proxy em redes centradas em informação," in *XIV Brazilian Symposium on Information and Computing Systems Security (SBSeg)*, 2014, pp. 2–15, (in portuguese).

[11] A. Chaabane, E. De Cristofaro, M. A. Kaafar, and E. Uzun, "Privacy in content-oriented networking: threats and countermeasures," *SIGCOMM Computer Communications Review*, vol. 43, no. 3, pp. 25–33, 2013.

[12] V. Jacobson, D. K. Smetters, J. D. Thornton, M. Plass, N. Briggs, and R. Braynard, "Networking named content," *Communications of the ACM*, vol. 55, no. 1, pp. 117–124, jan 2012.

[13] Éric Renault, A. Ahmad, and M. Abid, "Toward a security model for the future network of information," in *4th Intl Conference on Ubiquitous Information Technologies Applications (ICUT '09)*, 2009, pp. 1–6.

[14] B. Li, A. P. Verleker, D. Huang, Z. Wang, and Y. Zhu, "Attribute-based access control for ICN naming scheme," in *Proc. of the IEEE Conf. on Communications and Network Security (CNS '14)*, 2014, pp. 391–399.

[15] J. Zhang, Q. Li, and E. Schooler, "iHEMS: An information-centric approach to secure home energy management," in *3rd Intl Conference on Smart Grid Communications*, 2012, pp. 217–222.

[16] M. Mangili, F. Martignon, and S. Paraboschi, "A cache-aware mechanism to enforce confidentiality, trackability and access policy evolution in content-centric networks," *Comp. Netw.*, vol. 76, pp. 126–145, 2015.

[17] H. Xiong, X. Zhang, W. Zhu, and D. Yao, "CloudSeal: End-to-end content protection in cloud-based storage and delivery services," in *Security and Privacy in Comm. Netw.*, 2012, vol. 96, pp. 491–500.

[18] Z. Kissel and J. Wang, "Access control for untrusted content distribution clouds using unidirectional re-encryption," in *2013 Intl Conf. on High Performance Computing and Simulation*, July 2013, pp. 49–56.

[19] S. Chow, J. Weng, Y. Yang, and R. Deng, "Efficient unidirectional proxy re-encryption," in *Progress in Cryptology (AFRICACRYPT)*, ser. Lecture Notes in Computer Science, D. J. Bernstein and T. Lange, Eds., 2010, vol. 6055, pp. 316–332.

[20] S. Salsano, A. Detti, M. Cancellieri, M. Pomposini, and N. Blefari-Melazzi, "Transport-layer issues in information centric networks," in *2nd ACM Works. on Information-centric Networking*, 2012, pp. 19–24.

[21] F. Borges, A. Petzoldt, and R. Portugal, "Small private keys for systems of multivariate quadratic equations using symmetric cryptography," in *XXXIV CNMAC*, 2012, pp. 1085–1091.