SCIENCE CHINA
Information Sciences

• HIGHLIGHT •

September 2017, Vol. 60 098101:1–098101:3
doi: 10.1007/s11432-016-0305-5

A privacy-preserving video subscription scheme with the limitation of expire date

Liehuang ZHU, Mingxin CHEN, Zijian ZHANG*, Ange TONG & Chen XU

School of Computer Science and Technology, Beijing Institute of Technology, Beijing 100081*, China*

Received May 10, 2016; accepted July 8, 2016; published online December 9, 2016

Citation Zhu L H, Chen M X, Zhang Z J, et al. A privacy-preserving video subscription scheme with the limitation of expire date. Sci China Inf Sci, 2017, 60(9): 098101, doi: 10.1007/s11432-016-0305-5

There are many video service websites (e.g., iqiyi and youku) providing VIP services. In these video services, users can register for VIP by paying fare and then they can enjoy some special services in a certain time interval. We just define the video subscription as an aforementioned service. More and more users enjoy this form of video services and the number of the users is increasing all the time. Unfortunately, it is extremely easy for the video service provider to monitor and trace the online activities of users, and as a consequence, the sensitive information about the users' personal habits and interest may be exposed.

Therefore, in order to protect the interests of both the users and the video service provider in a video subscription service, designing a privacy-preserving video subscription scheme is imperative. We start by looking at the scheme of Lee et al. [1]. They proposed that the users login the system in a short period and provided an efficient linking operation for users who do not need unlinkability for the next period. It means when watching a collection of short videos, the users might require a short time period so they can quickly "re-anonymize", when watching a 120-minute movie straight through such re-anonymization may not be necessary. However, it does not consider the limitation of subscription expire date.

Therefore, in this article we present a privacy-preserving video subscription scheme with the limitation of expire date. This scheme proposes that the users subscribing the video service must choose the start subscription date and the end subscription date. Simultaneously, we utilize the efficient zero-knowledge proofs that a committed number lies in an interval [2] to prove the start subscription date and end subscription date to the service provider. Moreover, we ensure that the users' different logins cannot be tracked and linked by generating different anonymous login tokens with CL signature [3]. In addition, we allow the real-name payment and the real-name payment account cannot be linked with any registered information related to the users. In order to protect the interests of the video service provider, a user cannot login the system more than one time in a login period simultaneously.

Methodology. We present a description for the video subscription scheme with limitation expire date. This scheme consists of three participants, the system setting provider, the server and the client. And it contains six phases, including setup, registration, login, link, end and logout. Figure 1 illustrates the overview of the privacy-preserving video subscription scheme. Next we provide a detailed description as follows.

Setup. The setup phase is executed by the setup module of system setting provider and

* Corresponding author (email: zhangzijian@bit.edu.cn)

The authors declare that they have no conflict of interest.

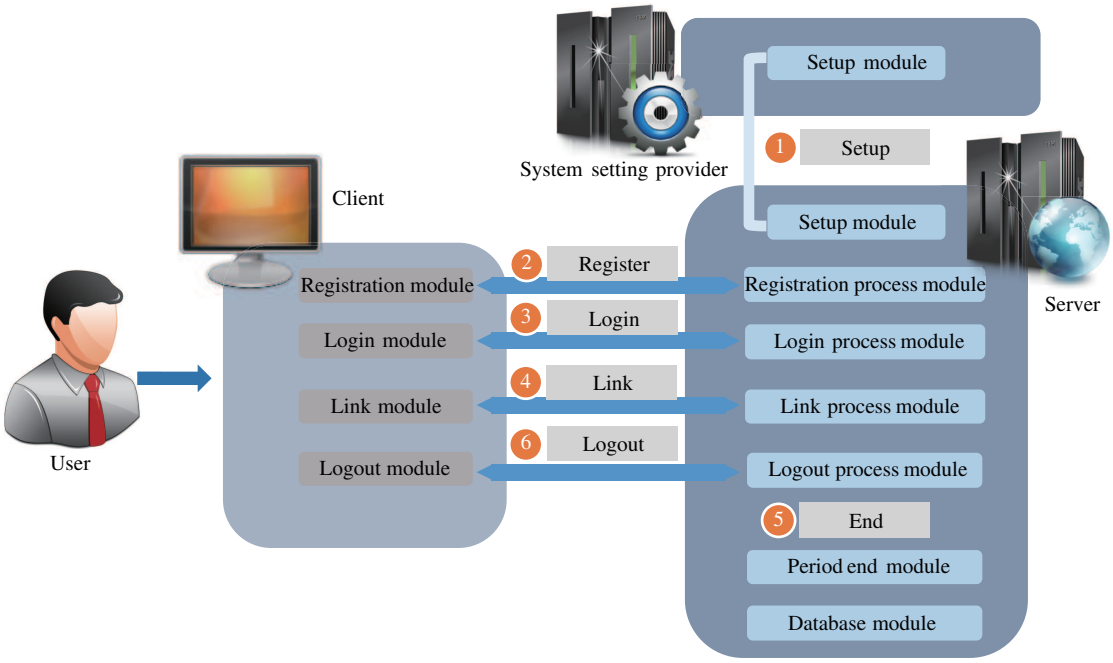

Figure 1 (Color online) Overview of the privacy-preserving video subscription scheme.

server.

(1) The system setting provider generates three security parameters t, l, s. Then it lets n be a large composite number whose factorization is unknown by the server and client. And then it lets g_n be an element of large order in $\mathbb{Z}_n^*$ and $h_n \in \langle g_n \rangle$.

(2) The server chooses $x, y, z_1, z_2 \leftarrow \mathbb{Z}_q$ and sets $X = g^x, Y = g^y, Z_1 = g^{z_1}$ and $Z_2 = g^{z_2}$. The public key of the server is $\mathrm{spk} = (q, \mathbb{G}, \mathbb{G}_T, g, g_T, X, Y, Z_1, Z_2)$ and the secret key is $\mathrm{ssk} = (x, y, z_1, z_2)$. Here $\mathbb{G} = \langle g \rangle$ is a bilinear group of prime order q with target group $\mathbb{G}_T$. $e(\cdot, \cdot)$ is the bilinear map and we let $g_T = e(g, g)$.

(3) The server states $\sigma = (\sigma.\mathrm{cur}, \sigma.\mathrm{next})$ and σ is a pair of sets. $\sigma.\mathrm{cur}$ is used to store the tokens of the current login period and $\sigma.\mathrm{next}$ is used to store the tokens of the next login period. Then the server must store σ into the database module.

(4) The server sets the time of login period T, the minimum and maximum of date $t_{-\infty}$, t_{∞}.

Registration. The registration phase is the interaction of registration module and registration process module.

(1) The user controls the client, chooses $d \leftarrow \mathbb{Z}_q$, and determines the subscription start date $\exp_{\mathrm{s}}$ and the subscription end date $\exp_{\mathrm{e}}$. It constructs $M = g^d {Z_1}^{\exp_{\mathrm{s}}} {Z_2}^{\exp_{\mathrm{e}}}$ and $\exp = \exp_{\mathrm{e}} - \exp_{\mathrm{s}}$. Then the client sends M and exp to the server.

(2) The client acts as the prover and the server acts as the verifier in a zero-knowledge proof of knowledge [4] to prove the $d, \exp_{\mathrm{s}}, \exp_{\mathrm{e}}$. If the proof fails, the registration operation fails.

(3) The server computes the fare of this subscription according to exp, and sends the fare to the client. Then the client uses its real-name payment account to pay the fare for this subscription.

(4) The server chooses $\alpha \leftarrow \mathbb{Z}_q^*$ and sets $a = g^\alpha$. Then it forms CL signature [3] $s = (a, A_1 = a^{z_1}, A_2 = a^{z_2}, b = a^y, B_1 = {A_1}^y, B_2 = {A_2}^y, c = a^x M^{xy\alpha})$, and sends s to the client.

(5) The client checks the signature. If the client checks successfully, the client sets the anonymous access credential as $\mathrm{sk} = (s, d, \exp_{\mathrm{s}}, \exp_{\mathrm{e}})$.

Login. The login phase is the interaction of the login module and the login process module. If a user wants to watch a video in the video service, he or she must login the service after the registration.

(1) The user inputs the sk. Then the cilent uses sk to create a blinded signature. It chooses $r_1, r_2 \leftarrow \mathbb{Z}_q^*$ and computes blinded signature $\tilde{s} = (\tilde{a} = a^{r_1}, \tilde{A_1} = {A_1}^{r_1}, \tilde{A_2} = {A_2}^{r_1}, \tilde{b} = b^{r_1}, \tilde{B_1} = {B_1}^{r_1}, \tilde{B_2} = {B_2}^{r_1}, \widehat{c} = c^{r_1 r_2})$.

(2) The client creates login token $Y_d(t) = {g_T}^{1/(d+\exp_{\mathrm{s}}+\exp_{\mathrm{e}}+t)}$. t is the start time of current login period and it is unique in a login period.

(3) The client sends $\tilde{s}, Y_d(t)$ to the server. If $Y_d(t) \in \sigma.\mathrm{cur}$, login fails. Otherwise, the server checks the blind signature $\tilde{s}$.

(4) The client acts as the prover, and the server acts as the verifier in a zero-knowledge proof of knowledge to prove $d, \exp_{\mathrm{s}}, \exp_{\mathrm{e}}, 1/r_2$ and $t_{\mathrm{cur}} \in [\exp_{\mathrm{s}}, \exp_{\mathrm{e}}]$. To get more information of the proof, we refer the reader to [2,3]. If the proof fails, then the login operation fails.

(5) The server updates the token storage sets, i.e., adds $Y_d(t)$ into σ.cur.

After login phase, the user can watch videos as long as the subscription date is not expired.

Link. The link phase is the interaction of the link module and the link process module. If a user does not need unlinkability with the next login period, he or she can execute this phase.

(1) The client uses sk to compute $Y_d(t)$ and $Y_d(t+T)$. Then the client sends $Y_d(t)$ and $Y_d(t+T)$ to the server.

(2) The server checks $Y_d(t) \in \sigma$.cur and $Y_d(t+T) \notin \sigma$.next. If not, the link phase fails.

(3) The client acts as the prover and the server acts as the verifier in a zero-knowledge proof of knowledge to prove $d, \exp_{\mathrm{s}}, \exp_{\mathrm{e}}$ and $t_{\mathrm{cur}} + T \in [\exp_{\mathrm{s}}, \exp_{\mathrm{e}}]$. If the proof fails, the link operation fails. Otherwise, the server adds $Y_d(t+T)$ into σ.next.

End. The period end module communicates with the database module, and updates the token storage sets, i.e., empties σ.cur and puts all the tokens from σ.next into σ.cur. Then the system comes into the next login period. The client who does not execute the link operation must login again in the next login period.

Logout. The logout phase is the interaction of the logout module and the logout process module. If a user wants to logout the video service, he or she can execute this phase.

(1) The client sends $Y_d(t)$ and $Y_d(t+T)$ to the server. Then the server checks $Y_d(t) \in \sigma$.cur. If not, the logout operation fails.

(2) Otherwise, The client acts as the prover and the server acts as the verifier in a zero-knowledge proof of knowledge to prove $d, \exp_{\mathrm{s}}, \exp_{\mathrm{e}}$. If the proof fails, the logout operation fails.

(3) Otherwise, the server deletes the $Y_d(t)$ from σ.cur. In addition, the server deletes the $Y_d(t+T)$ from σ.next if it exists in σ.next.

Experiments. We set the elliptic pairing group $\mathbb{G}$ is $y^2 = x^3 + 1 \bmod p$ and set a 160-bit group order (i.e., $|q| = 160$) and 512-bit base field (i.e., $|p| = 512$). We set the three security parameters $t = 80, l = 40, s = 40$ and $|n| = 512$.

We conduct the experiments on a computer with a 2.50 GHz Intel Core i5-2450M 4 CPU, 6 GB of RAM and Windows 7 64 bit operation system.

We compare two prior subscription protocol with our scheme. In Marina Blanton's protocol [5], there are two main phases, subscribe and access corresponding to registration and login in our scheme. In Anon-pass [1] there are three phases (i.e., registration, login and link) corresponding to our scheme. The costs of registration, login and link are respectively about 883, 535 and 355 ms in our scheme. However, in Marina Blanton's protocol the costs of subscribe and access phases are 1069 and 1944 ms. We can see that our scheme is faster than Marina Blanton's protocol. Moreover, the costs of registration, login and link in the Anon-pass protocol are respectively 296, 184 and 101 ms. To the best of our knowledge, the Anon-pass protocol does not consider the expire date. Although Anon-pass is more efficient than our scheme, our scheme is more secure than it.

Conclusion. In this article, we proposed a privacy-preserving video subscription scheme with the limitation of expire date, which enables the users to subscribe the service, and to access the service anonymously. This scheme can achieve the limitation of subscription expire date. At the same time, it can ensure that the users' different logins cannot be tracked and linked in the video service. Moreover, our scheme allows real-name payment to pay for the subscription service, different from other anonymous subscription protocols or systems which only provide anonymous payment.

Acknowledgements This work was supported by National Natural Science Foundation of China (Grant Nos. 61272512, 61300177) and Beijing Natural Science Foundation (Grant No. 4132054).

References

1 Lee M Z, Dunn A M, Waters B, et al. Anon-pass: practical anonymous subscriptions. In: Proceedings of IEEE Symposium on Security and Privacy (SP), Berkeley, 2013. 319–333

2 Boudot F. Efficient proofs that a committed number lies in an interval. In: Advances in Cryptology-EUROCRYPT 2000. Berlin: Springer, 2000. 431–444

3 Camenisch J, Lysyanskaya A. Signature schemes and anonymous credentials from bilinear maps. In: Advances in Cryptology-CRYPTO 2004. Berlin: Springer, 2004. 56–72

4 Camenisch J, Stadler M. Efficient group signature schemes for large groups. In: Advances in Cryptology-CRYPTO'97. Berlin: Springer, 1997. 410–424

5 Blanton M. Online subscriptions with anonymous access. In: Proceedings of the ACM Symposium on Information, Computer and Communications Security. New York: ACM, 2008. 217–227