Verifying MQV-Based Protocols Using ProVerif

Ernest-YongYi Yap, Ji-Jian Chin, and Alwyn Goh

Abstract ProVerif is an automatic protocol verifier that is usually used to find symbolic attacks in a protocol as described in the Dolev-Yao Security Model [7]. But according to its manual [2], it can also be used to verify some computation attacks such as those described in the Bellare-Rogaway (BR) or Canetti-Krawczyk (CK) Security Model [5]. This cryptographic tool does not recognize the laws of mathematics and the laws needed to be applied manually. This paper shows the security verification of authenticated MQV-based key exchange (AKE) protocols. We show the proof of correctness using this protocol verifier tool as well as some of the known computational attacks done by others such as Unknown-Key-Share attack using it. Included in our results are two MQV-based protocol variants: an identity based key agreement (FG IB-KA) and a certificateless identity authenticated based key agreement (CLAKA).

Keywords ProVerif · Protocol · MQV · IBKA · CLAKA · UKS · KCI

This manuscript was written during the second author's visit to Information Security Lab, MIMOS Berhad. The authors appreciate the financial assistance from the Ministry of Education of Malaysia in supporting this work with the Fundamental Research Grant Scheme (FRGS/1/2019/ICT04/MMU/02/5). The authors would also like to thank Jason Chia for assisting in discussions on cryptography proof techniques.

E.-Y. Yap (✉) · J.-J. Chin
Faculty of Engineering, Multimedia University, Cyberjaya, Malaysia

J.-J. Chin · A. Goh
Information Security Lab, MIMOS Berhad, Cyberjaya, Malaysia

H. Kim and K. J. Kim (eds.), *IT Convergence and Security*, Lecture Notes in Electrical Engineering 782, https://doi.org/10.1007/978-981-16-4118-3_6

1 Introduction

Cryptographic protocols have existed in the computing world for a long time and methods to create one have been proposed throughout the years. To start an encrypted conversation between two parties, they have to first decide on a secret key for encrypting their messages. The Diffie-Hellman (DH) key exchange [6] is one of the methods to allow two parties to share a symmetric key, but since the DH key exchange does not have entity authentication, it can be broken easily with a man-in-the-middle attack. Since then, cryptographers have modified the DH key exchange with additional entity authentication properties using methods such as digital signatures and certificates [3, 4, 13]. More secure and efficient mathematical methods such as the elliptic curves have been proposed as well.

In this paper, the automatic security analyzer tool for cryptographic protocol used is ProVerif [1]. This cryptographic protocol verifier is designed based on the Dolev-Yao model which means that it is primarily used to detect symbolic attacks [15–17]. It supports both symmetric and asymmetric cryptographic protocols, hash functions, digital signatures and key exchanges based on DH mechanics. It can also allow multiple sessions of the protocol to be run at once and providing an unlimited message capacity. Sometimes ProVerif may results in a false positive attack, but if some property is stated to be satisfied, then the property is confirmed to be satisfied. Some of the properties that ProVerif can verify is the secrecy of a message, entity authentication and strong secrecy, which means the adversary can't detect the value change of a secret.

To the best of our knowledge, it seems that there is no existing work that provides a ProVerif verification for MQV-based protocols. Thus, we fill in that research gap with these results. We use ProVerif to demonstrate the security properties of the main MQV protocol. We also demonstrate the Unknown Key Share (UKS) attack on MQV using it. As corollary results, we utilize ProVerif to verify the security of two related protocols: and ID-Based Key Agreement (FG IB-KA) protocol by Fiore and Gennaro [9] which is of an identity-based construction, as well as the Certificateless Authenticated Key Agreement (CLAKA) Protocol by He et al. [11]that uses a certificateless construction.

2 Protocols

2.1 MQV Protocol

Security Feature. The MQV protocol [14] is considered one of the most efficient DH-based protocols that uses public key cryptography to provide entity authentication. This protocol does not require a third-party key provider such as a KGC and it only allows two parties in a session. The protocol designers specifically designed this protocol to resist key compromise impersonation (KCI) attacks, known key attacks

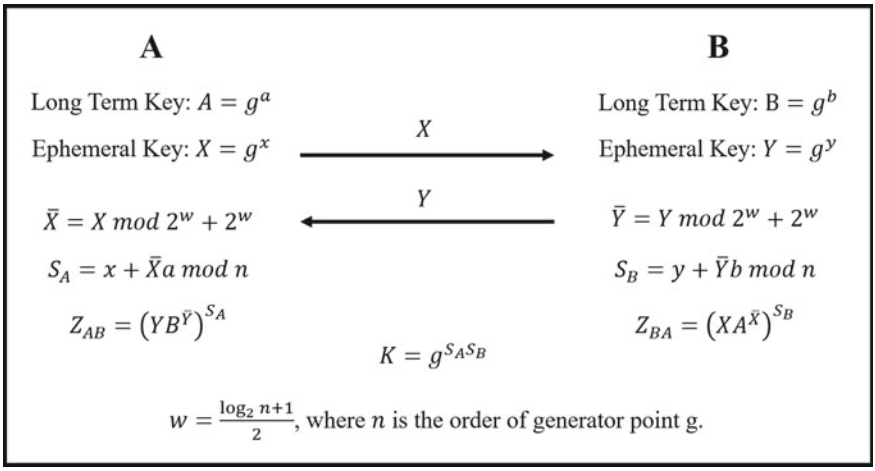

Fig. 1 MQV protocol

and provide perfect forward secrecy which means the adversary can't obtain every used session key despite having the long-term key. This protocol is proven to be secure in the BR model. However, an UKS attack was discovered by Kaliski [12] when the entity authentication is done implicitly.

Protocol Outline. The first MQV protocol proposed only uses implicit entity authentication, which mean both users only exchange the ephemeral key. Both users create their own long-term keys and derive a partial private key S_i as shown in Fig. 1, The shared key is then calculated using each other's long-term public key and self partial private key forming K.

2.2 FG IB-KA Protocol

Security Feature. The Fiore-Gennaro ID-Based Key Agreement (FG IB-KA) protocol [9] is an identity-based protocol derived from the MQV protocol. Identity-based cryptography is a method to remove certification of public keys by allowing principals to compute the public key of another principal based on the identity's information. The FG IB-KA protocol is modelled under the CK model which shows that the adversary can't distinguish between an actual session key and a random generated key with the same length. This protocol also provides forward secrecy, but it is considered weak forward secrecy which means the past used session keys are all safe but not the future ones [10]. Besides that, it resists most symbolic and computational attacks such as reflection attacks, KCI attacks and impersonation attacks.

Protocol Outline. Unlike the original MQV protocol, the FG IB-KA uses explicit entity authentication which means that the user identity is also sent by each of the

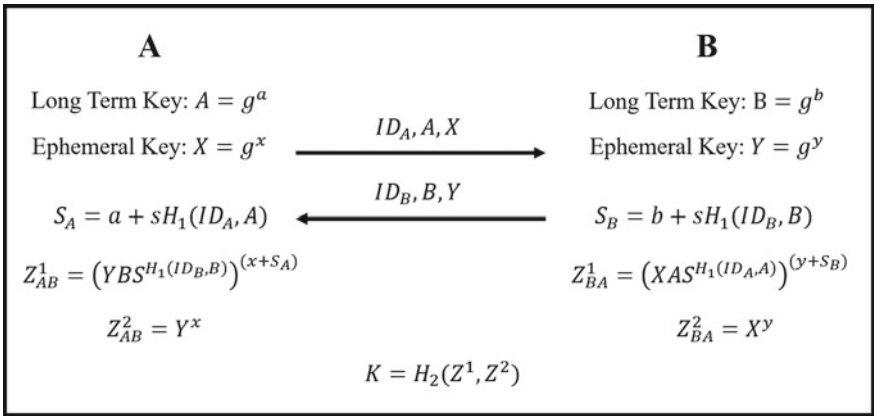

Fig. 2 FG IB-KA protocol

user in a session with it's long-term and ephemeral key. A Key Generation Center (KGC) takes in an user identity and derive a private key using Schnorr's Signature and also provide longterm public keys to both users. Using those KGC keys, Z^1 and Z^2 is calculated and the share key K is formed as shown in Fig. 2.

2.3 He et al. CLAKA Protocol

Security Feature. The Certificateless Authenticated Key Agreement (CLAKA) Protocol also known as the He-Padhye-Chen Protocol [11] is the certificateless version of the MQV-based protocol, where there is no key escrow. The CLAKA protocol is proved to be secure in the eCK model under the Gap-Diffie-Hellman (GDH) assumption. If the GDH problem has been broken with negligible probability, the advantage of the adversary in this protocol is said to be still negligible. In the security analysis done by Farouk [8], it is proven that this protocol is secure against an eCK model adversary, which means it resist known key attacks, KCI, UKS and provides forward secrecy.

Protocol Outline. The He-Padhye-Chen CLAKA uses elliptic curve cryptography, but it is converted to Diffie-Hellman notation for easier understanding. Since this is a certificate-less protocol, it does not rely completely on KGC but it has some similarity to FG IB-KA. Different from FG IB-KA, the KGC in CLAKA provides a random key and a similar Schnorr's Signature as private key to both principal. Both users generate their own long-term key and will exchange their identity, random key and ephemeral key. The long-term key is used to calculate Z^2, Z^1 and Z^3 is calculated in a similar way with FG IB-KA. At last, the share key is K as shown in Fig. 3.

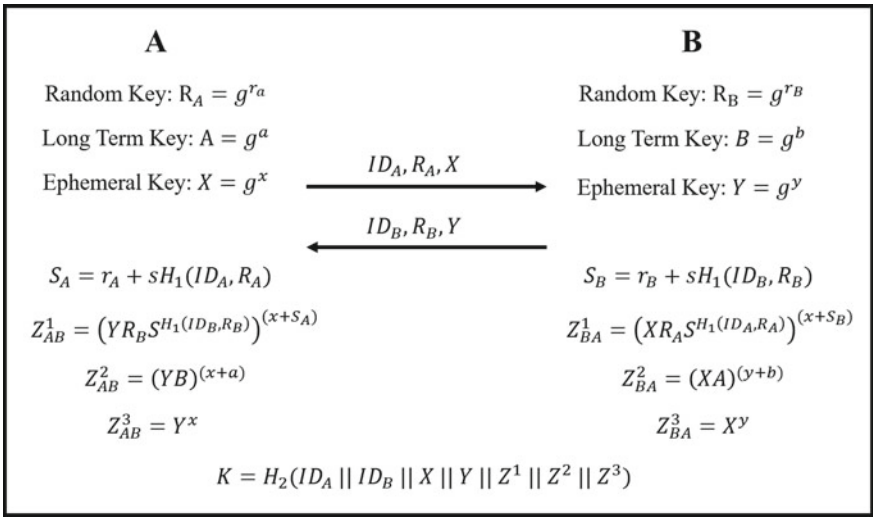

Fig. 3 CLAKA protocol

3 ProVerif

Since ProVerif does not recognize mathematical properties, the programmer needs to define the properties such as commutative, associative and distributive using the equation and reduc syntax. ProVerif is mostly used in protocols that do not have complicated mathematical expressions such as pairings, for example the Needham-Schroeder Public Key protocol [2]. The ProVerif uses queries to detect the vulnerabilities of the protocol. Each query can be programmed differently to simulate different types of attacks. A false query usually means that an attack is detected.

Proof of Correctness. To prove that the protocol works as intended, a proof of correctness is needed. Alice and Bob start a session with each other and exchange the secret key, Alice and Bob will then send each other a message encrypted using the secret key they exchanged. When Alice or Bob received the message, they decrypt it with their own key and check whether the message holds. If the message holds, the event will be executed, and the query will be false as the adversary only acts as a wire and attacks passively.

Secrecy of Messages. The secrecy of the message is defined as the adversary can't obtain the secret message that Alice sends to Bob or vice versa. The secrecy of a message is the most fundamental security property of a protocol: if an adversary can obtain the secret message of a session easily without the compromise of any keys, it means that the protocol is vulnerable to other attacks such as man-in-the-middle attacks or replay attacks. ProVerif can easily validate this security property using $query\ attacker(secretMessage)$.

KCI. A KCI attack is when the adversary possesses the long-term key of Alice or Bob, it can impersonate as the intended principal of Alice or Bob. To detect KCI attack in ProVerif, the long-term private key of Alice or Bob will be leaked out to the adversary via a public channel. The initiator will try to send out a secret message then execute the $eventSend(A, B, M)$. When the responder receives the message she will execute the event $Recv(A, B, M)$. If the responder executed $Recv(A, B, M)$ without the event $Send(A, B, M)$ executed, KCI attack is successful. The declared query for KCI attack is $event(Recv(A, B, M)) ==> event(Send(A, B, M))$.

Implicit Entity Authentication. Entity authentication is where one principal knows the identity of another principal that is in the same session, and is used to avoid MITM attacks or impersonation attacks. In ProVerif, Alice will execute $acceptsA(A, B, M, K)$ where A is her own public key, B is Bob's public key, M is the message she sent out and K is the shared key. When Bob received Alice's message, he will also compute the shared key and end the protocol by executing $termB(A, B, M, K)$. The declare query is $event(termB(A, B, M, K)) ==> event(acceptsA(A, B, M, K))$.

Key Indistinguishability. To fit in the model of BR or CK, an adversary must not have the ability to distinguish a real session key from a random key with negligible probability. ProVerif shows an example for key indistinguishability query in it's manual. This query shows the secrecy of the keys established by Alice when it starts a session with an honest principal Bob, with the sense that these keys are indistinguishable from independent random numbers.

UKS. UKS attack is an attack that allows an adversary to cause one principal to believe it is sharing a key with the adversary, but the principal actually shares it with another different principal that is not the adversary. In ProVerif, the query is $event(termB(A, B, M, K)\&\& event(acceptsA(A, B', M', K')) ==> K = K'$. This query allows the adversary to send different messages to Alice and Bob whereby the message can be different, but the key must the same.

4 Results and Discussions

Since these are quite complicated protocols, ProVerif requires some time to validate all queries. The more secure the protocol is, the longer time it takes for ProVerif to verify it. The time used to verify all the queries in every protocol is shown in Fig. 4. The specification of the computer used to verify these computer is i5-4460 core processor, 16GB RAM and operating system is Windows 10 Home with a solid state drive. All the queries in MQV finish processing under 1.5 h; FG IB-KA took under 10 h; and CLAKA completed under 47 h because of the long hash function in the end.

Although all the protocols have the same output from ProVerif, the trace graphs are different for every protocol. The adversary is passive in $event(Asuccess)$ and

Verification summary:
Query not attacker(secretA[]) is true.
Query not attacker(secretB[]) is true.
Query not event(Asuccess) is false.
Query not event(Bsuccess) is false.
Query event(Recv(x_1,exp(g,b[]),z)) ==> event(Send(x_1,exp(g,b[]),z)) is true.
Query event(termA(exp(g,a[]),y_1,z)) ==> event(acceptsB(exp(g,a[]),y_1,z)) is true.
Query event(termB(x_1,exp(g,b[]),z,k)) ==> event(acceptsA(x_1,exp(g,b[]),z,k)) is true.
Query event(termB(x_1,exp(g,b[]),z,k)) && event(acceptsA(x_1,exp(g,b[]),z,k')) ==> k = k' is true.
Query event(termB(x_1,y_1,z,k)) && event(acceptsA(x_1,y',z',k')) ==> k = k' is false.

Fig. 4 ProVerif result

$event(Bsuccess)$ as it is acting as a wire giving the proof of correctness in the protocols. Secrecy, implicit entity authentication, KCI resistance and key indistinguishability are all output true for all three protocols.

However, the UKS query is outputted as false on every protocol. Since ProVerif has the possibility of giving a false attack, the trace graph of the attack can be checked to verify the attack. The trace graph will be shown below in Fig. 5. The trace graph indeed shows the proof of correctness and the adversary is simply acting as a wire. For Fig. 5, an UKS attack is found on MQV as Alice thinks that she shares her key with the adversary, but Bob thinks that he is sharing his key with Alice, hence this is a positive attack. The UKS attack shown in FG IB-KA and CLAKA protocol is a false attack because both Alice and Bob is sharing the key with each other instead of the adversary, hence the UKS attack does not hold. This paper only shows one of the graph for explanation purpose, the rest of the trace graphs and codes can be found on GitHub on https://github.com/ernestyyy0306/ProVerif-MQV-Based.

5 Conclusion

In this work, the security properties of MQV, FG IB-KA and CLAKA protocol is verified using an automatic cryptographic verifier tools called ProVerif. This tool successfully detected an UKS attack on MQV protocol but gives a false UKS attacks on FG IB-KA and CLAKA protocol. These three protocols are proven to be secure in term of secrecy, key indistinguishability, KCI resistance and most symbolic attacks such as replay attacks and MITM attacks. This work shows that ProVerif is not only able to show symbolic attack as described in Dolev-Yao Model, but is also able to verify computational attacks such as KCI and UKS as described in BR or CK model. Besides, it also proves that complicated protocols that uses complicated mathematical properties such as point addition and point multiplication can be verified in ProVerif.

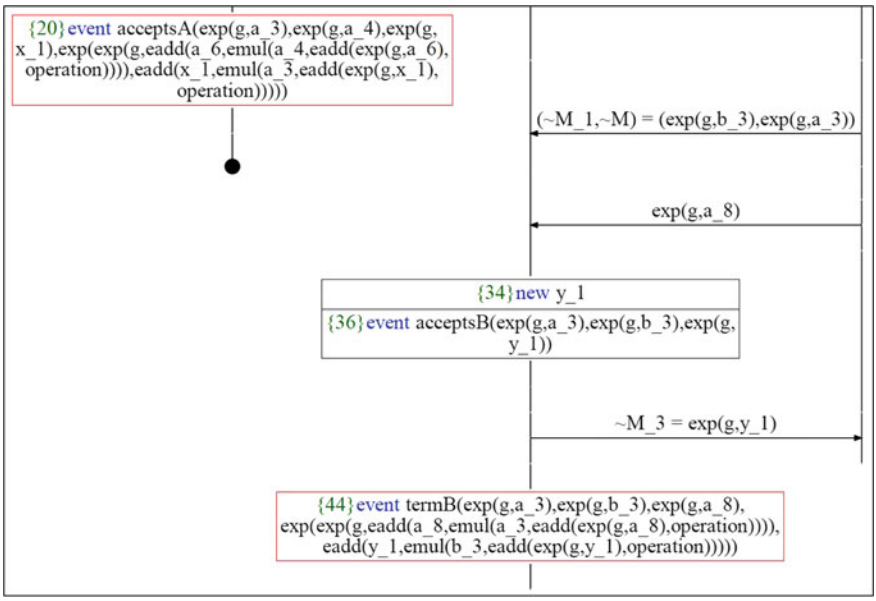

Fig. 5 MQV UKS attack trace graph

References

1. Blanchet B (2016) Modeling and verifying security protocols with the applied pi calculus and proverif
2. Blanchet B, Smyth B, Cheval V, Sylvestre M (2020) Proverif 2.02 pl1: automatic cryptographic protocol verifier, user manual and tutorial
3. Boyd C, Mathuria A, Stebila D (2003) Protocols for authentication and key establishment, vol 1. Springer
4. Choo KKR (2006) Key establishment: proofs and refutations. PhD thesis, Queensland University of Technology
5. Choo KKR, Boyd C, Hitchcock Y (2005) Examining indistinguishability-based proof models for key establishment protocols. In: International conference on the theory and application of cryptology and information security, pp 585–604. Springer
6. Diffie W, Hellman M (1976) New directions in cryptography. IEEE Trans Inform Theory 22(6), 644–654
7. Dolev D, Yao A (1983) On the security of public key protocols. IEEE Trans Inform Theory 29(2), 198–208
8. Farouk A, Miri A, Fouad MM, Abdelhafez AA (2014) Efficient pairing-free, certificateless two-party authenticated key agreement protocol for grid computing. In: 2014 fourth international conference on digital information and communication technology and its applications (DICTAP), pp 279–284. IEEE
9. Fiore D, Gennaro R (2010) Making the diffie-hellman protocol identity-based. In: Cryptographers' track at the RSA conference, pp 165–178. Springer
10. Fiore D, Gennaro R, Smart NP (2010) Constructing certificateless encryption and id-based encryption from id-based key agreement. In: International conference on pairing-based cryptography, pp 167–186. Springer

11. He D, Padhye S, Chen J (2012) An efficient certificateless two-party authenticated key agreement protocol. Comput Math Appl 64(6), 1914–1926
12. Kaliski BS Jr (2001) An unknown key-share attack on the mqv key agreement protocol. ACM Trans Inform Syst Secur (TISSEC) 4(3):275–288
13. Katz J, Lindell Y (2020) Introduction to modern cryptography. CRC Press
14. Menezes A (1997) Some new key agreement protocols providing implicit authentication. In: Workshop on selected areas in cryptography. CRC Press
15. Shashidhara R, Nayak SK, Das AK, Park Y (2021) On the design of lightweight and secure mutual authentication system for global roaming in resource-limited mobility networks. IEEE Access 9:12879–12895
16. Wu TY, Yang L, Lee Z, Chen CM, Pan JS, Islam S (2021) Improved ecc-based three-factor multiserver authentication scheme. Secur Commun Netw
17. Zhang J, Yang L, Gao X, Tang G, Zhang J, Wang Q (2021) Formal analysis of quic handshake protocol using symbolic model checking. IEEE Access (2021)