

Authentication

BACKGROUND

Field

5 Certain examples of the present disclosure provide one or more techniques for performing authentication between at least a first party and a second party, for example a first device and a second device.

Description of the Related Art

10 A fundamental problem in communication theory is how to transmit information between two parties without a third party being able to acquire and/or alter the information. For example, in the field of electronic financial transactions, it is very important to maintain secrecy in the communication between two parties.

Conventionally, the two parties who wish to exchange information are known respectively as Alice (A) and Bob (B), while an eavesdropper who wishes to gain unauthorised access to the information is known as Eve (E). For example, Alice, Bob and Eve may refer to devices or
15 users of devices.

A fundamental requirement of any secure communication scheme is that Alice and/or Bob must possess some kind of secret information that is unknown to Eve. For example, secret information may be used as the basis of encryption and/or subsequent decryption of information.

20 In various techniques, it is also important for Alice and/or Bob to be able to authenticate the other party before engaging in a secure operation, for example transmission of sensitive information. In the present disclosure, authentication may refer generally to a procedure or protocol to allow one party to verify the identity of another party. Mutual, or bi-directional authentication is a protocol that requires both parties to verify each other. Authentication helps
25 to prevent unauthorised access to sensitive information and other malicious activities that can compromise the security and integrity of systems and networks.

The above information is presented as background information only to assist with an understanding of the present disclosure. No determination has been made, and no assertion is made, as to whether any of the above might be applicable as prior art with regard to the
30 present disclosure.

SUMMARY

It is an aim of certain examples of the present disclosure to address, solve, mitigate or obviate, at least partly, at least one of the problems and/or disadvantages associated with the related art, for example at least one of the problems and/or disadvantages mentioned herein. Certain
5 examples of the present disclosure aim to provide at least one advantage over the related art, for example at least one of the advantages mentioned herein.

The present invention is defined in the independent claims. Advantageous features are defined in the dependent claims.

Certain examples of the present disclosure provide a method for authentication between a first
10 party and a second party, the method comprising: generating a current shared key by exchanging one or more messages between the first and second parties; and authenticating the current shared key based on a previously authenticated shared key.

In certain examples, the current shared key may be generated before the first and second parties are authenticated and/or before the identities of the first and second parties are
15 established.

In certain examples, a shared key may comprise a dynamically generated key (e.g. not a pre-existing key known to both parties in advance).

In certain examples, the current shared key may be authenticated based on both the previously authenticated shared key and the current shared key.

20 In certain examples, the current shared key may be authenticated based on at least one secret value obtained by the first party and/or the second party.

In certain examples, authenticating the current shared key may comprise exchanging one or more messages between the first and second parties, wherein at least one of the messages may be generated based on one or more of: the current shared key; the previously
25 authenticated shared key; and the at least one secret value.

In certain examples, generating the current shared key may comprise generating two or more current shared keys, and wherein the two or more current shared keys may be authenticated based on two or more previously authenticated shared keys.

In certain examples, the current shared key may be generated using one or more of: an
30 asymmetrical method of key exchange (e.g. Elliptic Curve Diffie-Hellman Ephemeral (ECDHE) key exchange); and Quantum Key Distribution (QKD).

In certain examples, the method may further comprise: when the current shared key is determining to be authenticated, storing the current shared key to be used as a previously authenticated shared key when authenticating a new current shared key.

5 In certain examples, the method may further comprise: when the current shared key is not determined to be authenticated, re-authenticating one or more of the previously authenticated shared keys.

In certain examples, the method may further comprise: identifying, based on the re-authentication, a set of one or more of the previously authenticated shared keys that are deemed to be authenticated.

10 In certain examples, the authentication may comprise symmetric authentication, wherein each of the first party and the second party perform substantially the same operations.

In certain examples, authenticating the current shared key may comprise: generating a combined key by combining the current key with one or more previously authenticated shared keys; authenticating the combined key as a shared key by exchanging one or more messages
15 between the first and second parties; and determining that the current shared key is authenticated based on authentication of the combined key.

In certain examples, generating the combined key may comprise an XOR operation between the current key and the one or more previously authenticated shared keys.

In certain examples, authenticating the combined key as a shared key may comprise
20 performing, by the first party, the following operations: generating a first token; encrypting the first token using the combined key; transmitting, to the second party, the encrypted first token; receiving, from the second party, an encrypted second token, wherein the second token has been generated based on the first token; decrypting the second token using the combined key; and authenticating the combined key as a shared key based on the first token and the second
25 token.

In certain examples, authenticating the combined key as a shared key may comprise performing, by the first party, the following operations: receiving, from the second party, an encrypted third token; decrypting the third token using the combined key; generating a fourth token based on the third token; encrypting the fourth token using the combined key; and
30 transmitting, to the second party, the encrypted fourth token.

In certain examples, the first and second parties may mutually authenticate each other based on authentication of the current shared key.

Certain examples of the present disclosure provide a device configured to operate according to a method according to any example, aspect, embodiment and/or claim disclosed herein.

Certain examples of the present disclosure provide a computer program comprising instructions which, when the program is executed by a computer or processor, cause the
5 computer or processor to carry out a method according to any example, aspect, embodiment and/or claim disclosed herein.

Certain examples of the present disclosure provide a computer or processor-readable data carrier having stored thereon a computer program according to any example, aspect, embodiment and/or claim disclosed herein.

10 Embodiments, aspects or examples disclosed in the description and/or figures falling outside the scope of the claims are to be understood as examples useful for understanding the present invention.

Other aspects, advantages, and salient features of the present disclosure will become apparent to those skilled in the art from the following detailed description, which, taken in
15 conjunction with the accompanying drawings, disclose examples of the present disclosure.

BRIEF DESCRIPTION OF THE FIGURES

Figure 1 illustrates an authentication technique involving an initial authentication event and one or more subsequent authentication events;

20 Figure 2 illustrates an authentication technique for performing subsequent authentication after an initial authentication;

Figure 3 is a flow diagram of a first authentication technique between Alice and Bob based on the technique of Figure 2;

Figure 4 is a flow diagram of a second authentication technique between Alice and Bob based
25 on the technique of Figure 2;

Figure 5 is a flow diagram of a third authentication technique between Alice and Bob based on the technique of Figure 2; and

Figure 6 is a block diagram of an exemplary device for deriving secret information shared with another device.

DETAILED DESCRIPTION

The following description of examples of the present disclosure, with reference to the accompanying drawings, is provided to assist in a comprehensive understanding of the present invention, as defined by the claims. The description includes various specific details to assist
5 in that understanding but these are to be regarded as merely exemplary. Accordingly, those of ordinary skill in the art will recognize that various changes and modifications of the examples described herein can be made.

Certain examples of the present disclosure provide one or more techniques for performing authentication between at least a first party and a second party. In the present disclosure, the
10 term “party” may refer to any suitable type of entity, for example a device, a user of a device, a network node, and the like. Different parties may be distinguished using any suitable label, for example “Alice” and “Bob”. In certain examples, authentication between parties (e.g. mutual authentication) may be achieved by authenticating shared secret information between the parties, and vice versa.

15 A device capable of implanting one or more techniques described herein may be of any suitable type, for example a mobile device (such as a mobile telephone), a computer terminal, a relay device, a server, a node in a network (such as the Internet or a private network) or any other suitable type of device for communicating. Furthermore, such a device may be a manually operated device (e.g. one operated by a user), or may be a device that is partially or fully
20 automated. In certain examples, one or more more of the techniques described herein may be applied to communication between internal components of one or more devices. Accordingly, references herein to a ‘device’ that communicates with another device may also include references to an internal component of a device that communicates with another internal component of either the same device or a different device.

25 The techniques described herein may be used in a wide variety of different applications, including, but not limited to, financial transactions, Police, Armed Forces, Government, mobile data, mobile voice, navigation and location information (e.g. GPS), financial services, banking, shipping communications, subscriber services, mobile security services, distributed networking, remote access, Internet communications, virtual private networks, satellite
30 communications, remote command and control systems, aircraft (e.g. drone aircraft), remote control, data storage and archiving, and identity management and security.

The skilled person will appreciate that parties authenticated according to one or more of the techniques described herein may engage in any suitable type of secure operation, including, but not limited to, secure communication.

In certain examples, authentication may be separated into different types of authentication and the different types of authentication may be handled in different ways. In particular, a distinction may be made between two different types of authentication:

1. Initial authentication, in which the identity of a party is verified or established a first time.
- 5 2. Subsequent authentication, in which the identity of a party whose identity has been verified previously is verified or established again.

As an example of the first type of authentication, a bank may wish to establish the identity of a new customer, and likewise the new customer may wish to establish that the bank is a genuine and trusted entity. As an example of the second type of authentication, a bank may wish to
10 verify that a party engaging with the bank is a genuine existing customer, and likewise an existing customer may wish to verify that the party they are engaging with is their bank. Similar interactions involving these two types of authentication may occur in relation to other entities, for example between a hotel and its guests.

Authentication events of the first type are typically carried out once, or at least relatively
15 infrequently. On the other hand, authentication events of the second type are typically carried out multiple times, possibly relatively frequently, after authentication of the first type has been successfully completed. Both types of authentication typically involve the exchange of information between the parties. However, the first type of authentication typically involves a more complex procedure than the second type of authentication. This is because
20 authentication of the second type typically relies, at least partly, on previously established shared secret information (for example a customer password in online banking). In contrast, authentication of the first type does not assume any pre-established shared secret information. Consequently, additional steps are typically required to establish shared secret information for the first time in such a way that the information does not become available to unauthorised
25 third parties.

Figure 1 illustrates an authentication technique involving an initial authentication event (comprising a first type of authentication) and one or more subsequent authentication events (each comprising a second type of authentication).

Referring to Figure 1, in a first operation 101, a first party (Alice) and a second party (Bob)
30 mutually authenticate each other. As part of the authentication process, Alice and Bob each acquire shared secret information. This is information known to Alice and Bob but not known to any unauthorised third party (Eve). The acquired shared secret information may be denoted $K(0)$. Any suitable procedure may be used to perform the authentication in operation 101. In this example, the authentication performed in operation 101 comprises the first type of
35 authentication, and so a relatively complex authentication procedure may be required.

In a second operation 102, Alice and Bob mutually authenticate each other again. As part of the authentication process, Alice and Bob each acquire new shared secret information, which may be denoted $K(1)$. Any suitable procedure may be used to perform the authentication in operation 102. In this example, the authentication performed in operation 102 comprises the second type of authentication, and so a relatively simple authentication procedure may be used. In particular, the authentication procedure may be based, at least partly, on the previously established shared secret information $K(0)$ allowing a more simple procedure to be used.

As shown in Figure 1, operation 102 may be repeated any number of times in an iterative manner. New shared secret information may be acquired in each iteration of operation 102. The shared secret information acquired in the n -th iteration may be denoted $K(n)$. The authentication performed in the n -th iteration may be based, at least partly, on the previously established shared secret information acquired in one or more previous iterations, for example $K(n-1)$.

Certain examples of the present disclosure provide one or more techniques for performing the second type of authentication. As noted above, the second type of authentication may be regarded as an authentication procedure in which two parties have already acquired some shared secret information, for example as a result of successful completion of a previous authentication procedure. For example, the previous authentication procedure may comprise a second type of authentication or a first type of authentication. As noted above, the first type of authentication may be regarded as an authentication procedure that does not assume any pre-established shared secret information between the parties.

Figure 2 illustrates an authentication technique for performing subsequent authentication (i.e. authentication of the second type) after an initial authentication (i.e. authentication of the first type). For example, the technique of Figure 2 may correspond to operation 102 of Figure 1.

Referring to Figure 2, in a first operation 201, a first party (Alice) and a second party (Bob) perform a procedure to acquire shared secret information. For example, Alice and Bob may perform a key exchange procedure to acquire a shared secret key. The acquired key may be denoted $K(n)$. In certain examples, the overall method of Figure 2 may be performed iteratively, and the value n may denote the n -th iteration. Any suitable procedure may be used to generate $K(n)$, for example: an asymmetrical method of key exchange (e.g. Elliptic Curve Diffie-Hellman Ephemeral (ECDHE) key exchange); Quantum cryptographic Key Distribution (QKD); one or more techniques as disclosed in International patent application publication number WO 2013/175224 A1 and equivalents; and/or one or more techniques disclosed in co-pending patent application GB 2211041.5 and equivalents. In certain examples, the shared secret information (e.g. a shared secret key) may be acquired dynamically. For example, the shared

secret information may be generated, derived etc. at the time of performing operation 201.

Operation 201 may be regarded as establishing “end points” of a communication sequence. This may be done to ensure that all subsequent communications will be between the key holders.

- 5 In a second operation 202, Alice and Bob perform an authentication procedure to enable the parties to mutually authenticate each other (e.g. establish and/or verify each others' identities). Any suitable authentication procedure may be used in operation 202. Various examples are described further below in relation to Figures 3-5. The authentication procedure is performed based on previously acquired shared secret information. For example, the authentication
10 procedure may be performed based on a shared secret key, $K(n-1)$, generated in a previous iteration of the method of Figure 2. The skilled person will appreciate that any combination of one or more secret keys generated in one or more previous iterations may be used.

- In certain examples, the authentication procedure in operation 202 may be performed based on the shared secret information acquired in operation 201, as well as the previously acquired
15 shared secret information. For example, the authentication procedure may be performed based on the key $K(n)$ generated in operation 201, as well as $K(n-1)$ and/or any other key(s) generated in any of the previous iterations, for example $K(n-2)$, $K(n-3)$, etc.

In certain examples, the authentication procedure in operation 202 may be performed based on one or more pieces of secret information obtained by Alice and/or Bob.

- 20 In certain examples, when the authentication procedure in operation 202 is successful (e.g. when Alice and Bob are successfully authenticated and/or when the identities of Alice and Bob are successfully established or verified), the key $K(n)$ may be stored. Accordingly, $K(n)$ may be used as a previously authenticated key when authenticating a new key in one or more subsequent iterations. This may ensure continuous and dynamic key management based on
25 previously authenticated key(s), rather than relying for example on pre-existing keys.

Accordingly, successful authentication in a previous iteration provides a basis for successful authentication in a current iteration. Such a chain of authentication may be initiated with a first type of authentication that does not rely on any pre-established shared secret information and which may be used to generate initial shared secret information, for example a key $K(0)$.

- 30 The technique illustrated in Figure 2 contrasts with certain conventional techniques. In particular, in the technique of Figure 2, the key exchange procedure of operation 201 is performed before the authentication procedure of operation 202. On the other hand, in certain conventional techniques, the authentication procedure is performed before the key exchange procedure.

There is a risk that Alice or Bob might actually be an imposter and not a legitimate party. In the technique of Figure 2, since the key exchange procedure occurs before the authentication procedure, this means that the illegitimacy of the imposter would not be identified through the authentication procedure until the shared key $K(n)$ has already been acquired by the parties, including the imposter. However, the authentication procedure is based on a previously generated key $K(n-1)$ (and possibly also on the key $K(n)$ generated in the first operation 201 and/or one or more other pieces of secret information). Accordingly, on the assumption that the previously generated key $K(n-1)$ is not known to the imposter then the imposter would fail the authentication procedure. In various techniques disclosed herein, $K(n)$ allows $K(n-1)$ to be compared without revealing $K(n-1)$.

In certain examples, the technique of Figure 2 may ensure security by continuously updating and authenticating shared secret information (e.g. keys) based on a shared history, which is not dependent for example on a set of fixed or pre-distributed keys known in advance to both parties. Successful current authentication is only possible if both parties share the same history (e.g. a history of previously successful authentication of one or more previously shared keys).

In certain examples, authentication is performed (e.g. identities are established) over an encrypted channel using the acquired shared secret information (e.g. post-key exchange).

In certain examples, authentication codes are never explicitly revealed. Instead, they change for every transaction, ensuring that the secrecy of the involved parties is maintained while establishing continuity in identity verification. This is particularly advantageous for example in scenarios involving impersonation attempts.

Figures 3 to 5 illustrate various exemplary methods based on the general method of Figure 2. In some methods (e.g. the methods of Figures 1 and 2) each party (Alice and Bob) sends a initial message to the other party and receives a response message in return, where each party generates the response message based at least partly on information contained in the received initial message. Each party then forms an opinion of the trustworthiness of the other party based on the response message they receive. In other methods (e.g. the method of Figure 5) each party sends a message to the other party without receiving a response message. Each party then forms an opinion of the trustworthiness of the other party based at least partly on information contained in the message they receive. In each case, the trustworthiness of the other party may be determined based on information already in their possession, for example information obtained during key exchange and/or pre-stored information, for example a key already authenticated in a previous iteration.

Figure 3 is a flow diagram of a first authentication technique between Alice and Bob based on the technique of Figure 2. Operation 301 of Figure 3 corresponds to operation 201 of Figure 2,

and operations 302-307 of Figure 3 correspond to operation 202 of Figure 2.

Referring to Figure 3, in a first operation 301a/301b, Alice and Bob perform a procedure to acquire shared secret information. In this example, the shared secret information comprises a shared secret key, denoted $K(n)$, although the skilled person will appreciate that the present disclosure is not limited to this example.

In the following, corresponding values obtained (e.g. calculated or derived) respectively by Alice and Bob may be distinguished with subscripts A and B. For example, the shared secret key obtained by Alice may be denoted $K_A(n)$ and the shared secret key obtained by Bob may be denoted $K_B(n)$. When Alice and Bob are legitimate parties and the authentication procedure is performed correctly, corresponding values (i.e. values whose denotation differ only by the subscripts $A \leftrightarrow B$) should be equal. For example, $K_A(n)$ should be equal to $K_B(n)$. For convenience, a value indicated without any subscript may be used to denote the values associated with Alice and/or Bob according to context.

Any suitable procedure may be used in operation 301a/301b to acquire the shared secret information. For example, to acquire the shared secret key $K(n)$, Alice and Bob may use any of: an asymmetrical method of key exchange (e.g. ECDHE key exchange); QKD; and/or techniques disclosed in WO 2013/175224 A1 and/or GB 2211041.5.

Alice and Bob each store their key, $K_A(n)$ and $K_B(n)$.

In operations 302 – 307, Alice and Bob perform a procedure to mutually authenticate each other, in order to determine whether each other are legitimate parties. If the mutual authentication is successful then the key $K(n)$ can be regarded as secure. As will become apparent, the authentication is performed based not only on newly generated key $K(n)$ but also a previously generated key $K(n-1)$ that has already been involved with successful authentication. This means that successful authentication involving the new key $K(n)$ depends on knowledge of $K(n-1)$ as well as $K(n)$. Therefore, even if an imposter gains knowledge of $K(n)$ in the key exchange procedure of operation 301, the subsequent authentication in operations 302-307 would fail based on the assumption that the imposter does not have knowledge of $K(n-1)$. This assumption is justified due to the previous successful authentication involving $K(n-1)$. Successful authentication of $K(n)$ then provides confidence in the secrecy of $K(n)$, which can then be used as the basis for further authentication in the next iteration when a new key $K(n+1)$ is generated.

In a second operation 302a, Alice computes a value $T_A(n)$ based on $K_A(n)$ and $K_A(n-1)$. The value $T_A(n)$ is computed such that neither the values of $K(n)$ nor $K(n-1)$ are derivable from $T_A(n)$. For example, if $K(n)$ and $K(n-1)$ are expressed as binary strings, then $T_A(n)$ may be computed as $T_A(n) = K(n) \otimes K(n-1)$, where $\otimes$ denotes bitwise XOR. Bob performs the same operation to

compute $T_B(n) = K_B(n) \otimes K_B(n-1)$ in operation 302b. The skilled person will appreciate that any other suitable technique may be used to generate $T_A(n)$ based on $K_A(n)$ and $K_A(n-1)$.

In a third operation 303a, Alice obtains a secret value RA. For example, RA may be a random value generated by Alice using any suitable method, for example a quasi random number generator or a method that exploits an inherent random characteristic of a physical system (e.g. sampling a quantum system or a signal subject to noise).

Alice then transmits a first secure message $M^{(1AB)}(T_A(n); RA)$ to Bob, where the message $M^{(1AB)}$ is generated based on $T_A(n)$ and RA. For example, Alice may encrypt RA using $T_A(n)$ as an encryption key and transmit the encrypted message to Bob. Any suitable encryption scheme may be used. For example, in the illustrated example a symmetric encryption scheme may be used such that the same key may be used to both encrypt and decrypt the message.

In a fourth operation 304b, Bob receives the message $M^{(1AB)}(T_A(n); RA)$ and processes it to recover the value RA. For example, in the illustrated example Bob decrypts the message using $T_B(n)$ as a decryption key to recover RA. In view of the symmetric encryption algorithm used, it can be seen that RA will be successfully recovered by the second party only if $T_A(n) = T_B(n)$. In turn, it can be seen that, in general, $T_A(n) = T_B(n)$ only if both $K_A(n) = K_B(n)$ and $K_A(n-1) = K_B(n-1)$. The skilled person will appreciate that this is why $K(n)$ is important to the security of the exchange.

In a fifth operation 305b, Bob generates a value RA' based on RA using any suitable operation that is known to both Alice and Bob. For example, RA' may be calculated as $RA' = RA + 1$.

Bob then transmits a second secure message $M^{(2BA)}(T_B(n); RA')$ to Alice, where the message $M^{(2BA)}$ is generated based on $T_B(n)$ and RA'. For example, Bob may encrypt RA' using $T_B(n)$ as an encryption key and transmit the encrypted message to Alice. Any suitable encryption scheme may be used, which may be the same as, or different from, the encryption algorithm used to generate the message $M^{(1AB)}$. In the illustrated example, a symmetric encryption scheme may be used.

In operation 305b, the reason a value RA' based on RA is used, instead of RA, is the following. The following considers a situation in which (i) RA is used instead of RA' in operation 305b, and (ii) $T_A(n) \neq T_B(n)$, possibly indicating that Alice or Bob is not legitimate, or that an error has occurred in the authentication procedure. In this scenario, Bob decrypts $M^{(1AB)}$ using $T_B(n)$ and then re-encrypts the resulting value using $T_B(n)$. In view of the symmetric property of the encryption/decryption algorithm, even though the decrypted value would be incorrect (due to incorrect $T_B(n)$), the re-encrypted value may still be correct. Essentially, the error introduced by decrypting using the incorrect key would be cancelled out by re-encrypting using the same (incorrect) key. Using a value RA' based on RA, instead of RA, avoids this cancellation effect.

In a sixth operation 306a, Alice receives the message $M^{(2BA)}(T_B(n); RA')$ and processes it to recover the value RA' . For example, in the illustrated example Alice decrypts the message using $T_A(n)$ as a decryption key to recover RA' . In view of the symmetric encryption algorithm used, it can be seen that RA' will be successfully recovered by the first party only if $T_A(n) =$
 5 $T_B(n)$.

In a seventh operation 307a, Alice performs verification based on the value of RA generated by Alice and the value of RA' recovered from the message $M^{(2BA)}$. For example, Alice may generate a value of RA' based on RA and compare this value with the value of RA' recovered from the message $M^{(2BA)}$. If the values match then the verification is deemed to be successful.
 10 Any other suitable equivalent comparison may be used instead.

It can be seen that the operations 302-307 comprise a procedure in which a value RA is passed from Alice to Bob and a related value RA' is passed back from Bob to Alice. These values are encrypted and decrypted on Alice's side using $T_A(n)$ and decrypted and encrypted on Bob's side using $T_B(n)$. The entire procedure will be successful if $T_A(n) = T_B(n)$. Therefore, the
 15 verification in operation 307a allows Alice to verify that $T_A(n) = T_B(n)$. As noted above, in general, $T_A(n) = T_B(n)$ only if both $K_A(n) = K_B(n)$ and $K_A(n-1) = K_B(n-1)$. Therefore, the verification in operation 307a allows Alice to verify that $K_A(n) = K_B(n)$ under the assumption that $K_A(n-1) = K_B(n-1)$ has been previously verified.

In order to allow Bob to also verify that $T_A(n) = T_B(n)$ and hence that $K_A(n) = K_B(n)$, the
 20 operations 302-307 may be carried out with the roles of Alice and Bob exchanged. These operations are briefly described below. The skilled person will appreciate that these operations may be carried out at the same time as, before, or after the operations described above.

In a third operation 303b, Bob obtains a secret value RB (e.g. a random value), generates a first secure message $M^{(1BA)}(T_B(n); RB)$, for example by encrypting RB using $T_B(n)$ as an encryption key, and transmits the first secure message to Alice.
 25

In a fourth operation 304a, Alice receives and processed the message $M^{(1AB)}(T_A(n); RB)$ to recover the value RB , for example by decrypting the message using $T_A(n)$ as a decryption key.

In a fifth operation 305a, Alice generates a value RB' based on RB using any suitable operation, for example $RB' = RB + 1$. Alice then generates a second secure message $M^{(2AB)}(T_A(n); RB')$, for
 30 example by encrypting RB' using $T_A(n)$ as an encryption key, and transmits the second secure message to Bob.

In a sixth operation 306b, Bob receives and processes the message $M^{(2AB)}(T_A(n); RB')$ to recover the value RB' , for example by decrypting the message using $T_B(n)$ as a decryption key.

In a seventh operation 307b, Bob performs verification based on the value of RB generated by Bob and the value of RB' recovered from the message $M^{(2AB)}$, for example by generating a value of RB' based on RB and comparing this with the value of RB' recovered from the message $M^{(2AB)}$.

- 5 Alice and Bob may exchange one or messages to confirm to each other whether or not the verification on each side was successful. For example, Alice may transmit a one bit flag to Bob where, for example, "1" indicates successful verification by Alice and "0" indicates unsuccessful verification by Alice. Likewise, Bob may transmit a one bit flag to Alice to indicate successful or unsuccessful verification by Bob. Any other suitable indication and/or encoding may be used
10 in other examples.

- If verification was successful for both Alice and Bob then Alice and Bob retain their stored values of $K(n)$. The retained values may then be used as an already-verified value in a next iteration of the method described above. For example, when repeating the above method, the retained values may be used as $K(n-1)$, and a newly generated value may be used as $K(n)$.
15 Accordingly, in each iteration, a newly generated value may be verified based on a value that was verified in the previous iteration. In certain examples, Alice and Bob may use a key derived from $K(n)$ or exchanged at the same time.

In certain examples, the values of RA and RB may be different for each iteration. That is, Alice and Bob may generate new values for each iteration.

- 20 If the verification by either Alice or Bob fails in operation 307 then both Alice and Bob discard their stored values of $K(n)$. In this case, any suitable procedure may be performed.

In certain examples, the procedure described above may be repeated to authenticate based on a newly generated $K(n)$.

- In certain examples, a procedure may be carried out to re-authenticate one or more previously stored values $K(n-1)$, $K(n-2)$, ..., $K(0)$. For example, the above procedure may be carried out
25 by Alice and Bob (omitting the key generation operation 301) using previously stored values $K(n-a)$ and $K(n-a-1)$ in place of $K(n)$ and $K(n-1)$, where $a=1, 2, 3, \dots$, to re-authenticate $K(n-a)$.

- In certain examples, Alice and Bob may each store the values of RA and RB for each iteration (denoted $RA(1)$, $RA(2)$, ... and $RB(1)$, $RB(2)$, ...) and these stored values may be used,
30 together with the stored values of K_A and K_B to perform re-authentication. For example, Alice may reauthenticate $K_A(n-a)$ by computing $K_A(n-a) \otimes K_A(n-a-1)$, $a=0, 1, 2, \dots$ based on previous stored values and comparing the result with $RA(n-a)$ and/or $RB(n-a)$. If the values match then $K_A(n-a)$ may be deemed reauthenticated.

In certain examples, the smallest value of 'a' for which re-authentication described above is successful may be regarded as indicative of the iteration before the iteration in which the authentication procedure began to fail. Alice and Bob may exchange one or more messages to inform each other of the determined value of 'a'. This procedure may be referred to as "drift
5 detection". The authentication procedure described above in relation to Figure 3 may then be performed starting from this iteration. This may be referred to as "sync correction".

Using the above techniques, Alice and Bob are able to verify shared secret information generated in a current iteration based on the previous verification of shared secret information generated in a previous iteration. Alice and Bob can detect errors in a predictable way. Alice
10 and Bob may also verify a part or all of the chain of secret information at any time.

For a device with a relatively small memory, only some previous values of K may be stored. In this case, when a new value of K is generated, verified and stored, the oldest value of K may be discarded. For example, if only two values of K are stored, when $K(n)$ is generated, verified and stored, the value of $K(n-2)$ is discarded so that only $K(n)$ and $K(n-1)$ are stored. Then, in
15 the next iteration, when $K(n+1)$ is generated, verified and stored, the value of $K(n-1)$ is discarded so that only $K(n+1)$ and $K(n)$ are stored.

Figure 4 is a flow diagram of a second authentication technique between Alice and Bob based on the technique of Figure 2. Operation 401 of Figure 4 corresponds to operation 201 of Figure 2, and operations 402-407 of Figure 4 correspond to operation 202 of Figure 2.

20 Referring to Figure 4, in a first operation 401a/401b, Alice and Bob perform a procedure to acquire shared secret information. In this example, the shared secret information comprises a shared secret key, denoted $K(n)$, although the skilled person will appreciate that the present disclosure is not limited to this example. Operation 301a/301b may be the same as, or similar to, operation 301a/301b of Figure 3.

25 Alice and Bob each store their key, $K_A(n)$ and $K_B(n)$.

In operations 402-407, Alice and Bob perform a procedure to mutually authenticate each other, in order to determine whether each other are legitimate parties. If the mutual authentication is successful then the key $K(n)$ can be regarded as secure. On the other hand, if the mutual authentication is not successful then the key $K(n)$ can be discarded. In this example, the
30 authentication is performed based on a previously generated key $K(n-1)$ that has already been involved with successful authentication. This means that successful authentication depends on knowledge of $K(n-1)$, and hence on the previous successful authentication. Operations 402-407 correspond to operations 302-307 of Figure 3.

In a second operation 402a, Alice obtains a secret value RA, for example a random value.

Operation 402a may be the same as, or similar to operation 303a of Figure 3.

In a third operation 403a, Alice generates a first secure message $M^{(1AB)}(K_A(n-1); RA)$ based on $K_A(n-1)$ and RA. For example, Alice may encrypt RA using $K_A(n-1)$ as an encryption key. Any suitable encryption algorithm may be used, for example a symmetric encryption scheme, such as AES₂₅₆. Alice then transmits the first secure message $M^{(1AB)}$ to Bob.

In a fourth operation 404b, Bob receives the first secure message $M^{(1AB)}(K_A(n-1); RA)$ and processes it to recover the value RA. For example, in the illustrated example Bob decrypts the first secure message using $K_B(n-1)$ as a decryption key to recover RA.

In a fifth operation 405b, Bob generates a value RA' based on RA using any suitable operation that is known to both Alice and Bob. For example, RA' may be calculated as $RA' = RA + 1$.

Bob then generates a second secure message $M^{(2BA)}(K_B(n-1); RA')$ based on $K_B(n-1)$ and RA'. For example, Bob may encrypt RA' using $K_B(n-1)$ as an encryption key. Any suitable encryption scheme may be used, which may be the same as, or different from, the encryption algorithm used to generate the first secure message $M^{(1AB)}$. In the illustrated example, a symmetric encryption scheme may be used. Bob then transmits the second secure message to Alice.

In a sixth operation 406a, Alice receives the message $M^{(2BA)}(K_B(n-1); RA')$ and processes it to recover the value RA'. For example, in the illustrated example Alice decrypts the message using $K_A(n-1)$ as a decryption key to recover RA'.

In a seventh operation 407a, Alice performs verification based on the value of RA generated by Alice and the value of RA' recovered from the message $M^{(2BA)}$. For example, Alice may generate a value of RA' based on RA and compare this value with the value of RA' recovered from the message $M^{(2BA)}$. If the values match then the verification is deemed to be successful. Any other suitable equivalent comparison may be used instead.

Operations 402-407 described above allow Alice to determine that Bob has knowledge of $K_B(n-1)$. Under the assumption that only a legitimate or trustworthy party would have knowledge of $K_B(n-1)$, this allows Alice to determine that Bob is a legitimate party. This in turn allows Alice to have confidence that only a legitimate party, namely Bob, was involved in the key exchange of operation 401, and hence only Bob as a legitimate party has knowledge of $K_B(n)$. This process thus provides justification for the above assumption in the next iteration.

In order to allow Bob to perform a corresponding verification, the operations 402-407 may be carried out with the roles of Alice and Bob exchanged. These operations are briefly described below. The skilled person will appreciate that these operations may be carried out at the same time as, before, or after the operations described above.

In a second operation 402b, Bob obtains a secret value RB (e.g. a random value).

In a third operation 403b, Bob generates a first secure message $M^{(1BA)}(K_B(n-1); RB)$, for example by encrypting RB using $K_B(n-1)$ as an encryption key, and transmits the first secure message $M^{(1BA)}$ to Alice.

- 5 In a fourth operation 404a, Alice receives and processes the first secure message $M^{(1BA)}(K_B(n-1); RB)$, for example using decryption, to recover the value RB.

In a fifth operation 405a, Alice generates a value RB' based on RB. Alice then generates a second secure message $M^{(2AB)}(K_A(n-1); RB')$, for example by encrypting RB' using $K_A(n-1)$ as an encryption key, and transmits the second secure message to Bob.

- 10 In a sixth operation 406b, Bob receives and processes the second secure message $M^{(2AB)}(K_A(n-1); RB')$, for example using decryption, to recover the value RB'.

In a seventh operation 407b, Bob performs verification based on the value of RB generated by Bob and the value of RB' recovered from the message $M^{(2AB)}$.

- 15 Alice and Bob may exchange one or messages (e.g. one bit flags) to confirm to each other whether or not the verification on each side was successful. If verification was successful for both Alice and Bob in operation 407 then Alice and Bob retain their stored values of $K(n)$, which may then be used in a next iteration of the method described above. On the other hand, if the verification by either Alice or Bob fails in operation 407 then both Alice and Bob discard their stored values of $K(n)$. Any suitable technique described above in relation to the method of Figure 3, for example re-authentication, drift detection, sync correction and/or storing a limited number of previous values of K , may be applied in the method of Figure 4.
- 20

Figure 5 is a flow diagram of a third authentication technique between Alice and Bob based on the technique of Figure 2. Operation 501 of Figure 5 corresponds to operation 201 of Figure 2, and operations 502-504 of Figure 5 correspond to operation 202 of Figure 2.

- 25 Referring to Figure 5, in a first operation 501a/501b, Alice and Bob perform a procedure to acquire two items of shared secret information. In this example, the shared secret information comprises a pair of shared secret keys, denoted $K1(n)$ and $K2(n)$, although the skilled person will appreciate that the present disclosure is not limited to this example. The shared secret information may be obtained in the same or a similar manner to operation 301a/301b of Figure 3, where two items of shared secret information may be obtained by performing the key exchange procedure twice.
- 30

Alice and Bob each store the keys, $K1(n)$ and $K2(n)$.

In a second operation 502a, Alice generates a value $T1_A(K1_A(n), K1_A(n-1))$ based on $K1_A(n)$ and $K1_A(n-1)$. For example, $T1_A$ may be generated using any suitable hash function taking $K1_A(n)$ and $K1_A(n-1)$ as inputs. Alice then transmits $T1_A$ to Bob. Alice then transmits the value $T1_A$ to Bob.

- 5 In a third operation 503b, Bob generates a value $T1_B(K1_B(n), K1_B(n-1))$ based on $K1_B(n)$ and $K1_B(n-1)$ in the same manner that Alice generated the value $T1_A(K1_A(n), K1_A(n-1))$ based on $K1_A(n)$ and $K1_A(n-1)$.

- 10 In a fourth operation 504b, Bob compares the value $T1_A$ received from Alice and the value $T1_B$ generated by Bob. If the values are equal then Bob to determine that Alice has knowledge of $K1_A(n-1)$. Under the assumption that only a legitimate or trustworthy party would have knowledge of $K1_A(n-1)$, this allows Bob to determine that Alice is a legitimate party. This in turn allows Bob to have confidence that only a legitimate party, namely Alice, was involved in the key exchange of operation 501, and hence only Alice as a legitimate party has knowledge of $K1_A(n)$. This process thus provides justification for the above assumption in the next iteration.

- 15 In order to allow Alice to perform a corresponding verification, the operations 502-404 may be carried out with the roles of Alice and Bob exchanged. These operations are briefly described below. The skilled person will appreciate that these operations may be carried out at the same time as, before, or after the operations described above.

- 20 In a second operation 502b, Bob generates a value $T2_B(K2_B(n), K2_B(n-1))$ based on $K2_B(n)$ and $K2_B(n-1)$, for example using a hash function, and transmits the value $T2_B$ to Alice.

In a third operation 503a, Alice generates a value $T2_A(K2_A(n), K2_A(n-1))$ based on $K2_A(n)$ and $K2_A(n-1)$ in the same manner that Bob generated the value $T2_B$.

In a fourth operation 504a, Alice compares the value $T2_B$ received from Bob and the value $T2_A$ generated by Alice.

- 25 Alice and Bob may exchange one or messages (e.g. one bit flags) to confirm to each other whether or not the verification on each side was successful. If verification was successful for both Alice and Bob in operation 504 then Alice and Bob retain their stored values of $K1(n)$ and $K2(n)$, which may then be used in a next iteration of the method described above. On the other hand, if the verification by either Alice or Bob fails in operation 504 then both Alice and Bob
30 discard their stored values of $K1(n)$ and $K2(n)$. Any suitable technique described above in relation to the method of Figure 3, for example re-authentication, drift detection, sync correction and/or storing a limited number of previous values of $K1$ and $K2$, may be applied in the method of Figure 5.

Figure 6 illustrates an exemplary device (or apparatus) for deriving shared secret information and/or for communicating with another device. For example, the techniques disclosed in relation to Figures 1-5 may be implemented using a device as disclosed in relation to Figure 6. For example, Alice and Bob may each comprise a device as disclosed in relation to Figure 6. The device 600 comprises a processor (or controller) 601 for controlling the overall operation of the device 600. For example, the processor 601 may be configured for performing operations as described above for key exchange and authentication. The device 600 also comprises a memory 603 for storing information and data required for the aforementioned operations. The device 600 also comprises an external interface 605 for communicating with another device via any suitable communication link (e.g. wired or wireless). For example, under the control of the processor 601, the external interface 605 may be configured to transmit and receive messages as described above.

In certain examples, the device 600 may also comprise a user input/output (I/O) unit 607 for allowing a user to interact with the device 600. For example, the user I/O unit 607 may comprise one or more input devices (e.g. a keyboard, touch screen, etc.) for inputting commands to the device 600. The user I/O unit 607 may comprise one or more output devices (e.g. display, LEDs, speaker, etc.) for outputting information (e.g. status information) for a user. In certain examples, if the device 600 is configured to operate autonomously, then the user I/O unit 607 may be omitted. In some examples, the device 600 may be configured to interface with another device in close proximity. In this case, the interface between the device 600 and the other device may be a wired link or a relatively short-range communication link such as a Bluetooth or NFC link. In other examples, the device 600 may be configured to interface with another device located remotely. In this case, the device 600 may communicate with the other device via a network, for example the Internet.

The terms and words used in this specification are not limited to the bibliographical meanings, but are merely used to enable a clear and consistent understanding of the present disclosure.

The same or similar components may be designated by the same or similar reference numerals, although they may be illustrated in different drawings.

Detailed descriptions of elements, features, components, structures, constructions, functions, operations, processes, characteristics, properties, integers and steps known in the art may be omitted for clarity and conciseness, and to avoid obscuring the subject matter of the present disclosure.

Throughout this specification, the words "comprises", "includes", "contains" and "has", and variations of these words, for example "comprise" and "comprising", means "including but not limited to", and is not intended to (and does not) exclude other elements, features, components,

structures, constructions, functions, operations, processes, characteristics, properties, integers, steps and/or groups thereof.

Throughout this specification, the singular forms “a”, “an” and “the” include plural referents unless the context dictates otherwise. For example, reference to “an object” includes reference to one or more of such objects.

By the term “substantially” it is meant that the recited characteristic, parameter or value need not be achieved exactly, but that deviations or variations, including for example, tolerances, measurement errors, measurement accuracy limitations and other factors known to those of skill in the art, may occur in amounts that do not preclude the effect the characteristic, parameter or value was intended to provide.

Throughout this specification, language in the general form of “X for Y” (where Y is some action, process, function, activity, operation or step and X is some means for carrying out that action, process, function, activity, operation or step) encompasses means X adapted, configured or arranged specifically, but not exclusively, to do Y.

Elements, features, components, structures, constructions, functions, operations, processes, characteristics, properties, integers, steps and/or groups thereof described herein in conjunction with a particular aspect, embodiment, example or claim are to be understood to be applicable to any other aspect, embodiment, example or claim disclosed herein unless incompatible therewith.

It will be appreciated that examples of the present disclosure can be realized in the form of hardware, software or any combination of hardware and software. Any such software may be stored in any suitable form of volatile or non-volatile storage device or medium, for example a ROM, RAM, memory chip, integrated circuit, or an optically or magnetically readable medium (e.g. CD, DVD, magnetic disk or magnetic tape).

Certain examples of the present disclosure provide a computer program comprising instructions which, when the program is executed by a computer or processor, cause the computer or processor to carry out a method according to any example, embodiment, aspect and/or claim disclosed herein. Certain examples of the present disclosure provide a computer or processor-readable data carrier having stored thereon such a computer program.

The techniques described herein may be implemented using any suitably configured apparatus and/or system. Such an apparatus and/or system may be configured to perform a method according to any aspect, embodiment, example or claim disclosed herein. Such an apparatus may comprise one or more elements, for example one or more of receivers, transmitters, transceivers, processors, controllers, modules, units, and the like, each element configured to

perform one or more corresponding processes, operations and/or method steps for implementing the techniques described herein. For example, an operation/function of X may be performed by a module configured to perform X (or an X-module). An apparatus and/or one or more elements thereof may be implemented in the form of hardware, software, a virtualised
5 function instantiated on an appropriate platform (e.g. on a cloud infrastructure), or any combination of these.

While the invention has been shown and described with reference to certain examples, it will be understood by those skilled in the art that various changes in form and detail may be made therein without departing from the scope of the invention, as defined by the appended claims.

Claims

1. A method for authentication between a first party and a second party, the method comprising:
generating a current shared key by exchanging one or more messages between the first
and second parties; and
5 authenticating the current shared key based on a previously authenticated shared key.
2. A method according to claim 1, wherein the current shared key is authenticated based on both
the previously authenticated shared key and the current shared key.
- 10 3. A method according to claim 1 or 2, wherein the current shared key is authenticated based on
at least one secret value obtained by the first party and/or the second party.
4. A method according to claim 1, 2 or 3, wherein authenticating the current shared key comprises
exchanging one or more messages between the first and second parties, wherein at least one of
15 the messages is generated based on one or more of: the current shared key; the previously
authenticated shared key; and the at least one secret value.
5. A method according to any preceding claim, wherein generating the current shared key
comprises generating two or more current shared keys, and wherein the two or more current shared
20 keys are authenticated based on two or more previously authenticated shared keys.
6. A method according to any preceding claim, wherein the current shared key is generated using
one or more of: an asymmetrical method of key exchange (e.g. Elliptic Curve Diffie-Hellman
Ephemeral (ECDHE) key exchange); and Quantum Key Distribution (QKD).
25
7. A method according to any preceding claim, further comprising: when the current shared key
is determining to be authenticated, storing the current shared key to be used as a previously
authenticated shared key when authenticating a new current shared key.
- 30 8. A method according to any preceding claim, further comprising: when the current shared key
is not determined to be authenticated, re-authenticating one or more of the previously authenticated
shared keys.
9. A method according to claim 8, further comprising: identifying, based on the re-authentication,
35 a set of one or more of the previously authenticated shared keys that are deemed to be
authenticated.

10. A method according to any preceding claim, wherein the authentication comprises symmetric authentication, wherein each of the first party and the second party perform substantially the same operations.

5 11. A method according to any preceding claim, wherein authenticating the current shared key comprises:

generating a combined key by combining the current key with one or more previously authenticated shared keys;

10 authenticating the combined key as a shared key by exchanging one or more messages between the first and second parties; and

determining that the current shared key is authenticated based on authentication of the combined key.

12. A method according to claim 11, wherein generating the combined key comprises an XOR
15 operation between the current key and the one or more previously authenticated shared keys.

13. A method according to claim 11 or 12, wherein authenticating the combined key as a shared key comprises performing, by the first party, the following operations:

20 generating a first token;
encrypting the first token using the combined key;
transmitting, to the second party, the encrypted first token;
receiving, from the second party, an encrypted second token, wherein the second token has been generated based on the first token;

25 decrypting the second token using the combined key; and
authenticating the combined key as a shared key based on the first token and the second token.

14. A method according to claim 13, wherein authenticating the combined key as a shared key comprises performing, by the first party, the following operations:

30 receiving, from the second party, an encrypted third token;
decrypting the third token using the combined key;
generating a fourth token based on the third token;
encrypting the fourth token using the combined key; and
transmitting, to the second party, the encrypted fourth token.

35 15. A method according to any preceding claim, wherein the first and second parties mutually authenticate each other based on authentication of the current shared key.

16. A device configured to operate according to a method of any preceding claim.
17. A computer program comprising instructions which, when the program is executed by a computer or processor, cause the computer or processor to carry out a method according to any of
5 claims 1 to 15.
18. A computer or processor-readable data carrier having stored thereon a computer program according to claim 17.