

## Claims

1. A computer-implemented method of establishing a key between a first device and a second device in a network, the network comprising a first key node and a second key node, the first key node and second key node with access to a same set of keys, the method

5 comprising:

receiving, at the first device, data representative of first key establishment data from at least the first key node and the second device for use in calculating a Bilocation Key, wherein the Bilocation key is based on a selected key from the set of keys; and

10 receiving, at the second device, data representative of second key establishment data from at least the first device for use in requesting the Bilocation Key from the second key node, wherein the Bilocation Key is calculated based on the selected key from the set of keys;

wherein the first and second devices use their corresponding Bilocation Keys to each generate a Final Key based on an agreed portion of the first key establishment data and second key establishment data.

15

2. The computer-implemented method as claimed in claim 1, wherein the first device and the second device are using a quantum-secure network, the quantum-secure network comprising the first key node and the second key node, the set of keys of the first key node and second key node being a set of quantum distributed keys that are the same, the method

20 further comprising:

receiving, at the first device, data representative of first key establishment data from at least the first key node and the second device for use in calculating the Bilocation Key, wherein the Bilocation key is based on a selected quantum distributed key from the set of quantum distributed keys; and

25

receiving, at the second device, data representative of second key establishment data from at least the first device for use in requesting the Bilocation Key from the second key node, wherein the Bilocation Key is calculated based on the selected quantum distributed key from the set of quantum distributed keys;

30

wherein the first and second devices use their corresponding Bilocation Keys to each generate the Final Key based on an agreed portion of the first key establishment data and second key establishment data.

3. The computer-implemented method according to claims 1 or 2, wherein the Final Key is calculated by each of the first and second devices based on combining their corresponding

Bilocation Keys with an agreed combination of one or more random nonces received in the first or second session establishment data therebetween.

4. The computer-implemented method according to any of claims 1 to 3, wherein the Final Key is calculated by each of the first and second devices based on their corresponding Bilocation Keys and a challenge response protocol the results of which are used to agree the Final Key.

5. The computer-implemented method according to any preceding claim, wherein receiving, at the first device, data representative of first session establishment data further comprising:

performing a first handshake with the second device to receive data representative of identity data associated with the second device and second key node, and a first nonce generated by the second device;

requesting an Intermediate Bilocation Key from the first key node, wherein the request includes data representative of the retrieved identity data and a third nonce computed from the received first nonce and a second nonce generated by the first device, wherein the first key node uses the request data and a selected key from the set of keys to establish the Intermediate Bilocation Key and a first anti-replay nonce;

receiving the Intermediate Bilocation Key and the first anti-replay nonce from the first key node; and

performing a further handshake with the second device for receiving a second anti-replay nonce from the second device after the second device receives the Bilocation Key and the second anti-replay nonce from the second key node.

6. The computer-implemented method according to claim 5, wherein receiving, at the second device, data representative of second session establishment data further comprising:

performing a second handshake with the first device to receive data representative of identity data associated with the first device and first key node, and the first anti-replay nonce, and the second nonce;

requesting a Bilocation Key from the second key node, wherein the request includes data representative of the retrieved identity data, a third nonce computed from the first nonce and second nonce, and the first anti-replay nonce, wherein the second key node uses the data in the request and a selected key from the set of keys to generate the Bilocation Key;

receiving the Bilocation Key from the second key node and a second anti-replay nonce generated by the second key node.

7. The computer-implemented method according to any preceding claim, wherein receiving, at the first device, data representative of first session establishment data further comprising:

5 sending, by the first device, a first handshake request for the second device to provide data representative of: the identity of the second device; the identity of the second key node; and a first nonce generated by the second device;

receiving, from the second device, data representative of: the generated first nonce; the identity of the second device; and the identity of the second key node;

generating, by the first device, a second random nonce;

10 sending, from the first device, an Intermediate Bilocation Key request to the first key node, wherein the Intermediate Bilocation Key request includes data representative of: the identity of the second device, the identity of the second node, a third nonce based on combining the first and second nonces;

15 receiving, at the first device, the Intermediate Bilocation Key established by the first key node and a first anti-replay nonce generated by the first key node;

sending, by the first device, a second handshake message to the second device including data representative of: the second nonce and first anti-replay nonce; the identity of the first device; and the identity of the first key node;

20 generating at the second device the third nonce based on the received second nonce and its generated first nonce;

sending, by the second device, a Bilocation Key request to the second key node including data representative of: the third and first anti-replay nonces; the identity of the first device; and the identity of the first key node;

25 receiving, at the second device, the Bilocation Key established by the second key node and a second anti-replay nonce generated by the second key node, wherein the second key node establishes the Bilocation Key based on generating the Intermediate Bilocation Key based on the received data from the second device and combining the Intermediate Bilocation Key with the second anti-replay nonce;

30 sending, from the second device, in a further handshake message the received second anti-replay nonce to the first device;

generating, at the first device, the Bilocation key based on combining the received second anti-replay nonce with the received Intermediate Bilocation Key;

wherein the first device and second device each establish a Final Key therebetween based at least on the Bilocation key.

35 8. The computer-implemented method according to any of claims 5 to 7, further comprising establishing, at the first key node, an Intermediate Bilocation Key for use by the first device in calculating the Bilocation Key, establishing the Intermediate Bilocation Key comprising:

receiving, at the first key node from the first device, a Intermediate Bilocation Key request to generate the Intermediate Bilocation Key, wherein the request includes data representative of: the identity of the second device, the identity of the second key node, and a third nonce which is derived from a first random nonce generated by the second device and a second random nonce generated by the first device;

selecting, at the first key node, a key from the set of keys accessible by the first key node using a key selection function with inputs based at least on data representative of the identity of the second device, the identity of the second key node, the identity of the first device, the identity of the first key node, and the third nonce, wherein the set of keys accessible by the first key node are the same as the set of keys accessible by the second key node;

generating, at the first key node, a first random anti-replay nonce;

generating, at the first key node, the Intermediate Bilocation Key based on the selected key, the first anti-replay nonce, the identity of the second device, the identity of the second key node, the identity of the first device, the identity of the first key node, and the third nonce;

sending, from the first key node to the first device, the Intermediate Bilocation Key and the first anti-replay nonce.

9. The computer-implemented method according to any of claims 5 to 8, further comprising establishing, at the second key node, the Bilocation Key for use by the second device in calculating the Key, wherein establishing the Bilocation Key comprising:

receiving, at the second key node from the second device, a request for a Bilocation Key, wherein the request includes data representative of: the identity of the first device, the identity of the first key node, a third nonce derived from a first random nonce generated by the second device and a second random nonce generated by the first device, and a first anti-replay nonce received at the second device from the first device;

selecting a key from the set of keys accessible by the second key node using a key selection algorithm based at least on the identity of the second device, the identity of the second key node, the identity of the first device, the identity of the first key node, and the third nonce, wherein the set of keys accessible by the first key node are the same as the set of keys accessible by the second key node;

generating, at the second key node, the Intermediate Bilocation Key based on the selected key, the first anti-replay nonce, the identity of the second device, the identity of the

second key node, the identity of the first device, the identity of the first key node, and the third nonce;

generating, at the second key node, a second random anti-replay nonce;

generating, at the second key node, the Bilocation Key based on combining the

5 Intermediate Bilocation Key with the second anti-replay nonce; and

sending, from the second key node to the second device, data representative of the Bilocation Key and the second anti-replay nonce.

10. The computer-implemented method according to claims 5 to 9, wherein sending, from the first device, the Intermediate Bilocation Key request further comprising:

10 generating, by the first device, a second random nonce,  $N_I$ ;

generating, by the first device, the third nonce,  $N_{IR}$ , based on the second random nonce,  $N_I$ , and the first random nonce,  $N_R$ ;

creating, by the first device, a first incomplete link-info data packet,  $L_I$ , based on data representative of at least: the identity of the second device,  $ID_R$ ; the identity of the second key node,  $KID_R$ ; and the third nonce,  $N_{IR}$ ;

15

sending, from the first device to the first key node, a Intermediate Bilocation Key requests, wherein the request includes data representative of the first incomplete link-info data packet,  $L_I$ .

11. The computer-implemented method according to claim 10, wherein the first

20 incomplete link-info packet takes the form  $L_I = \{ID_R, KID_R, N_{IR}\}$ .

12. The computer-implemented method according to claim 10, wherein the data representative of the third nonce,  $N_{IR}$ , comprises data representative of a hash of the third nonce,  $N_{IR}'$ .

13. The computer-implemented method according to claim 12, wherein the first

25 incomplete link-info packet takes the form  $L_I = \{ID_R, KID_R, N_{IR}'\}$ .

14. The computer-implemented method according to any of claims 10 to 13, wherein establishing, by the first key node, the Intermediate Bilocation Key further comprising:
- receiving, at the first key node from the first device, the Intermediate Bilocation Key request, wherein the request includes data representative of the incomplete link-info data packet,  $L_I$ ;
- generating, by the first key node, a first complete link-info data packet,  $L'$ , using data representative of at least: the identity of the first device,  $ID_I$ ; the identity of the first key node,  $KID_I$ ; data representative of the received first incomplete link-info packet,  $L_I$ ;
- calculating, by the first key node, the hash of  $L'$ , denoted  $\#(L')$  and using the  $\#(L')$  as an index for selecting the set of keys associated with the first key node and the second key node;
- selecting, by the first key node, a key,  $K_Q$ , from the set of keys accessible by the first key node using a key selection function applied to the complete link-info  $L'$ , wherein the key selection function used by the first key node is the same as that used by the second key node;
- generating, by the first key node, the first random anti-replay nonce,  $N_{REPI}$ ;
- performing, by the first key node, a look-up of the identity key associated with the second key node,  $K_R$  based on the identity of the second key node,  $ID_R$ , wherein the first key node and second key node maintain a mapping or lookup table of identity keys and corresponding key node identifiers of key nodes within the network;
- performing, by the first key node, a look-up of the identity key associated with the first key node,  $K_I$ ;
- creating, by the first key node, the Intermediate Bilocation Key,  $K_{IBK}$ , based on combining data representative of: the first complete link-information; the selected key,  $K_Q$ ; the identify key of the first key node,  $K_I$ ; the identity key of the second key node,  $K_R$ ; and the first anti-replay nonce,  $N_{REPI}$ ; and
- sending, from the first key node to the first device, the Intermediate Bilocation Key,  $K_{IBK}$ , and the first anti-replay nonce,  $N_{REPI}$ .
15. The computer-implemented method according to claim 14 as dependent on claim 10 or 11, wherein the first complete link-info packet takes the form  $L' = (ID_I || KID_I || L_I.ID_R || L_I.KID_R || L_I.N_{IR})$  where  $||$  denotes concatenation.
16. The computer-implemented method according to claim 14 as dependent on claim 12 or 13, wherein the first complete link-info packet takes the form  $L' = (ID_I || KID_I || L_I.ID_R || L_I.KID_R || L_I.N_{IR'})$  where  $||$  denotes concatenation.
17. The computer-implemented method according to any of claims 14 to 16, wherein the Intermediate Bilocation Key,  $K_{IS}$ , is created based on  $K_{IS} = \#(K_Q, \#(L'), N_{REPI}) XOR$

$K_I \text{ XOR } K_R$ .

18. The computer-implemented method according to any of claims 14 to 17, wherein the key selection function comprises one or more of: selecting the least significant  $N$  bits of  $\#(L')$ , where  $N \geq 1$ ; selecting the most significant  $N$  bits of  $\#(L')$ , where  $N \geq 1$ ; selecting a specific  
5 portion of  $N$  bits of  $\#(L')$ , where  $N \geq 1$ ; selecting the bottom  $N$  bits of  $\#(L')$ , where  $N \geq 1$ ; any other selection function suitable for generating a unique index for selecting a key from a set of keys; or any other function configured to select from any number of elements in an array representing  $\#(L')$ .

19. The computer-implemented method according to claims 5 to 18, wherein sending,  
10 from the second device, the Bilocation Key request further comprising:  
receiving, from the first device at the second device, the second random nonce,  $N_I$ , and the first anti-replay nonce,  $N_{REPI}$ ;  
generating, by the second device, the third nonce,  $N_{IR}$ , based on the second random nonce,  $N_I$ , and the first random nonce,  $N_R$ ;  
15 creating, by the second device, a second incomplete link-info data packet,  $L_R$ , based on data representative of at least: the identity of the first device,  $ID_I$ ; the identity of the first key node,  $KID_I$ ; and the third nonce,  $N_{IR}$ ;  
sending, from the second device to the second key node, a Bilocation Key request, wherein the request includes data representative of the second incomplete link-info data  
20 packet,  $L_R$ , and the first anti-replay nonce,  $N_{REPI}$ .

20. The computer-implemented method according to claim 19, wherein the second incomplete link-info packet takes the form  $L_R = \{ID_I, KID_I, N_{IR}\}$ .

21. The computer-implemented method according to claim 19, wherein the data  
representative of the third nonce,  $N_{IR}$ , comprises data representative of a hash of the third  
25 nonce,  $N_{IR}'$ .

22. The computer-implemented method according to claim 21, wherein the first incomplete link-info packet takes the form  $L_I = \{ID_R, KID_R, N_{IR}'\}$ .

23. The computer-implemented method according to any of claims 19 to 22, wherein establishing, by the second key node, the Bilocation Key further comprising:  
30 receiving, at the second key node from the second device, the Bilocation Key request, wherein the request includes data representative of the second incomplete link-info data packet,  $L_R$ , and the first anti-replay nonce,  $N_{REPI}$ ;  
generating, by the second key node, the second complete link-info data packet,  $L'$ , using data representative of at least: the identity of the second device,  $ID_R$ ; the identity of

the second key node,  $KID\_R$ ; and data representative of the received second incomplete link-info packet,  $L\_R$ ;

calculating, by the second key node, the hash of  $L'$ , denoted  $\#(L')$  and using the  $\#(L')$  as an index for selecting the set of keys associated with the second key node and the first key node;

selecting, by the second key node, the key,  $K\_Q$ , from the set of keys of the second key node using the key selection function applied to the second complete link-info  $L'$ , wherein the key selection function used by the second key node is the same as that used by the first key node;

generating, by the second key node, the second random anti-replay nonce,  $N\_REPR$ ;  
performing, by the second key node, a look-up of the identity key associated with the first key node,  $K\_I$  based on the identity of the first key node,  $KID\_I$ , wherein the first key node and second key node maintain a mapping or lookup table of identity keys and corresponding key node identifiers of key nodes within the network;

performing, by the second key node, a look-up of the identity key associated with the second key node,  $K\_R$ ;

creating, by the second key node, the Intermediate Bilocation Key,  $K\_IBK$ , based on combining data representative of: the second complete link-information; the selected key,  $K\_Q$ ; the identity key of the first key node,  $K\_I$ ; the identity key of the second key node,  $K\_R$ ; and the first anti-replay nonce,  $N\_REPI$ ;

creating, by the second key node, the Bilocation Key,  $K\_BK$ , based on combining data representative of: the Intermediate Bilocation Key,  $K\_IBK$ , and the second anti-replay nonce,  $N\_REPR$ ; and

sending, from the second key node to the second device, the Bilocation Key,  $K\_BK$ , and the second anti-replay nonce,  $N\_REPR$ .

24. The computer-implemented method according to claim 23 as dependent on claim 19 or 20, wherein the second complete link-info packet takes the form  $L' = (L\_R.ID\_I || L\_R.KID\_I || ID\_R || KID\_R || L\_R.N\_IR)$  where  $||$  denotes concatenation.

25. The computer-implemented method according to claim 23 as dependent on claim 21 or 22, wherein the second complete link-info packet takes the form  $L' = (L\_R.ID\_I || L\_R.KID\_I || ID\_R || KID\_R || L\_R.N\_IR')$  where  $||$  denotes concatenation.

26. The computer-implemented method according to any of claims 23 to 25, wherein the Intermediate Bilocation Key,  $K\_IBK$ , is created based on  $K\_IBK = \#(K\_Q, \#(L'), N\_REPI) XOR K\_I XOR K\_R$ .

27. The computer-implemented method according to any of claims 23 to 26, wherein the

Bilocation Key,  $K_{BK}$ , is created based on  $K_{BK} = \#(K_{IBK}, N_{REPR})$ .

28. The computer-implemented method according to claim 27, wherein generating the Bilocation Key,  $K_{BK}$ , includes generating a final Bilocation Key,  $K_{BK}'$ .

5 29. The computer-implemented method according to claim 28, wherein the final Bilocation Key,  $K_{BK}'$ , is created based on  $K_{BK}' = \#(N_{IR}, K_{BK})$ .

30. The computer-implemented method according to any preceding claim, wherein the Final Key is calculated by each of the first and second devices based on combining their corresponding Bilocation Key with an agreed nonce.

10 31. The computer-implemented method according to claim 28 or 29, wherein the Final Key is calculated by each of the first and second devices based on combining their corresponding final Bilocation Key with an agreed nonce.

15 32. The computer-implemented method according to any preceding claim, the method further comprising generating, by the second device, a further nonce,  $N_T$ , and sending, by the second device, the further nonce,  $N_T$ , to the first device, wherein the further nonce is used for calculating the Final Key.

20 33. The computer-implemented method according to claim 32, wherein the Final Key is calculated based on one from the group of:  $K_S = K_{BK} \text{ XOR } N_T$ ;  $K_S = K_{BK} \text{ XOR } K_{EN} \text{ XOR } N_T$ ;  $K_{BK}' \text{ XOR } N_T$ ;  $K_S = K_{BK}' \text{ XOR } K_{EN} \text{ XOR } N_T$ , wherein  $K_{EN}$  is an encryption key used between the first and second devices;  $K_S = K_{BK} \text{ XOR } K_{TLS} \text{ XOR } N_T$ ;  $K_S = K_{BK}' \text{ XOR } K_{TLS} \text{ XOR } N_T$ , wherein  $K_{TLS}$  is a TLS session key used between the first and second devices.

34. The computer-implemented method according to any preceding claim, wherein the Final Key is calculated by each of the first and second devices based on an intermediate key value,  $K_X$ , derived from their corresponding Bilocation Key,  $K_{BK}$ , and a challenge response

protocol the results of which are used to agree the Final Key, the challenge response protocol further comprising:

- creating, at the first device, a first challenge nonce,  $C_I$ ;
- creating, at the second device, a second challenge nonce,  $C_R$ ;
- 5 sending, from the first device to the second device, the first challenge nonce,  $C_I$ ;
- sending, from the second device to the first device, the second challenge nonce,  $C_R$ ;
- calculating, at the first device, the first signed responses using  $C_I$  and the received  $C_R$  based on the following values of:  $X_I = \#(K_X, C_I, 0)$ ;  $X_R = \#(K_X, C_R, 1)$ ;  $R_I = \#(X_I, C_R)$ ;  $R_R = \#(X_R, C_I)$ , where # is a hash function;
- 10 calculating, at the second device, the second signed responses using  $C_R$  and the received  $C_I$  based on the following values of:  $X_I = \#(K_X, C_I, 0)$ ;  $X_R = \#(K_X, C_R, 1)$ ;  $R_I = \#(X_I, C_R)$ ;  $R_R = \#(X_R, C_I)$ , where # is the same hash function used by the first device;
- 15 sending, from the first device to the second device, the first signed response  $R_I$ ;
- sending, from the second device to the first device, the second signed response  $R_R$ ;
- in response to determining, at the first device, that the  $R_R$  received from the second device matches the value of  $R_R$  calculated by the first device, calculating the Final Key as  $K_S = X_I \text{ XOR } X_R$ ;
- 20 in response to determining, at the second device, the  $R_I$  received from the first device matches the value of  $R_I$  calculated by the second device, calculating the Final Key as  $K_S = X_I \text{ XOR } X_R$ ;
- in response to, either the first device and/or second device, determining the received signed response does not match the corresponding calculated signed response, abandoning the connection.
- 25

35. The computer-implemented method according to claim 34, wherein the intermediate key value,  $K_X$ , is calculated at both the first and second device based on at least one from the group of:

- $K_X = K_{BK}$ ;
- 30  $K_X = \#(K_{BK}, K_{TLS})$ , where  $K_{TLS}$  is a TLS session key between first and second devices;
- $K_X = K_{BK} \text{ XOR } K_{TLS}$ , where  $K_{TLS}$  is a TLS session key between first and second devices;
- $K_X = \#(K_{BK}, K_{EN})$ , where  $K_{EN}$  is a key shared between first and second
- 35 devices;
- $K_X = K_{BK} \text{ XOR } K_{EN}$ , where  $K_{EN}$  is a key shared between first and second devices; and
- any other function for modifying  $K_{BK}$  with one or more agreed values known to both

the first and second devices.

36. The computer-implemented method according to claim 28 or any of claim 29 to 33 as dependent on claim 28, wherein the Final Key is calculated by each of the first and second devices based on an intermediate key value,  $K_X$ , derived from their corresponding final Bilocation Key,  $K_{BK'}$ , and a challenge response protocol the results of which are used to agree the Final Key, the challenge response protocol further comprising:

creating, at the first device, a first challenge nonce,  $C_I$ ;

creating, at the second device, a second challenge nonce,  $C_R$ ;

sending, from the first device to the second device, the first challenge nonce,  $C_I$ ;

10 sending, from the second device to the first device, the second challenge nonce,  $C_R$ ;

calculating, at the first device, the first signed responses using  $C_I$  and the received  $C_R$  based on the following values of:  $X_I = \#(K_X, C_I, 0)$ ;  $X_R = \#(K_X, C_R, 1)$ ;  $R_I = \#(X_I, C_R)$ ;  $R_R = \#(X_R, C_I)$ , where  $\#$  is a hash function;

15 calculating, at the second device, the second signed responses using  $C_R$  and the received  $C_I$  based on the following values of:  $X_I = \#(K_X, C_I, 0)$ ;  $X_R = \#(K_X, C_R, 1)$ ;  $R_I = \#(X_I, C_R)$ ;  $R_R = \#(X_R, C_I)$ , where  $\#$  is the same hash function used by the first device;

sending, from the first device to the second device, the first signed response  $R_I$ ;

20 sending, from the second device to the first device, the second signed response  $R_R$ ;

in response to determining, at the first device, that the  $R_R$  received from the second device matches the value of  $R_R$  calculated by the first device, calculating the Final Key as  $K_S = X_I \text{ XOR } X_R$  or  $K_S = \#(X_I, X_R)$ ;

25 in response to determining, at the second device, the  $R_I$  received from the first device matches the value of  $R_I$  calculated by the second device, calculating the Final Key as  $K_S = X_I \text{ XOR } X_R$  or  $K_S = \#(X_I, X_R)$ ;

in response to, either the first device and/or second device, determining the received signed response does not match the corresponding calculated signed response, abandoning the connection.

30 37. The computer-implemented claim according to claim 36, wherein the intermediate key value,  $K_X$ , is calculated at both the first and second device based on at least one from the group of:

$K_X = K_{BK'}$ ;

35  $K_X = \#(K_{BK'}, K_{TLS})$ , where  $K_{TLS}$  is a TLS session key between first and second devices;

$K_X = K_{BK'} \text{ XOR } K_{TLS}$ , where  $K_{TLS}$  is a TLS session key between first and second devices;

$K_X = \#(K_{BK'}, K_{EN})$ , where  $K_{EN}$  is a key shared between first and second devices;

$K_X = K_{BK'} \text{ XOR } K_{EN}$ , where  $K_{EN}$  is a key shared between first and second devices; and

5           any other function for modifying  $K_{BK}$  with one or more agreed values known to both the first and second devices.

38.       The computer-implemented method according to any preceding claim, wherein the third nonce is calculated based on an exclusive or, XOR, operation between the first random nonce and the second random nonce.

10       39.       The computer-implemented method as claimed in any preceding claim, wherein each of the key nodes are part of a quantum-secure network, and each set of keys are a set of quantum distributed keys delivered using a quantum key distribution system.

15       40.       The computer-implemented method as claimed in any of claims 1 to 38, wherein each of the key nodes are part of a classically-secure network, and each set of keys or distributed keys are delivered using a classical key distribution system.

41.       The computer-implemented method as claimed in claim 40, wherein the quantum key distribution system may include at least one from the group of:

Satellite quantum key distribution system;

Terrestrial-optical-quantum key distribution system; or

20       any type of quantum key distribution system that is capable of delivering a set of quantum keys to a first and a second key node in a quantum-safe or quantum-secure fashion.

42.       The computer-implemented method as claimed in any preceding claim, wherein: communications between the first device and the first key node are performed over a first quantum-secure channel;

25       communications between the second device and the second key node are performed over a second quantum-secure channel; and

communications between the first device and second device are performed, during establishment of the Final Key over a classical communications channel.

30       43.       The computer-implemented method as claimed in any preceding claim, wherein the set of keys is a set of one time pad keys.

44.       The computer-implemented method as claimed in claims 42 or 43, wherein the classical communications channel is a standard encrypted communications channel.

45. The computer-implemented method as claimed in any preceding claim, wherein the first key node and second key node are located in different geographical locations, and the first key node is different to the second key node.
- 5 46. The computer-implemented method as claimed in any of claims 1 to 44, wherein the first key node and second key node are located in the same geographical location.
47. The computer-implemented method as claimed in any of claims 1 to 44 or 46, wherein the first key node and second key node are located in the same logical location.
- 10 48. The computer-implemented method as claimed in any one of claims 1 to 44, 46 or 47, wherein the first key node and second key node are the same key node, and the set of keys is only known by the first key node and the second key node.
- 15 49. The computer-implemented method as claimed in any one of claims 1 to 44, 46, 47 or 48, wherein the first key node and second key node are the same logical key node, the set of keys of the logical key node are is only shared with itself or that only the logical key node owns, and the selected key is selected from the set of keys known only to the logical key node.
- 20 50. A system comprising a first device and a second device and a secure network, the secure network comprising a first key node and a second key node, the first key node including or accessing a set of keys and the second key node including or accessing the same set of keys, wherein each of the first device, the second device, the first key node and second key node include a processor unit, a memory unit, and a communication interface, the processor unit connected to the memory unit and the communication interface, wherein the processor unit, memory unit and communication interface are adapted to implement the corresponding steps of the computer-implemented method as claimed in any of claims 1 to 49.
- 25 51. A system comprising:  
an first device;  
a second device;  
an first key; and  
30 a second key node, wherein the first device, the second device, the first key node, and the second key node each comprise a respective processor unit, memory unit, and communication interface, the processor unit connected to the memory unit and the communication interface, wherein the processor unit, memory unit and communication interface are adapted to implement the corresponding steps of the computer-implemented method as claimed in any of claims 1 to 49; and

wherein the first device and second device are configured to communicate with each other over standard communication channel and with the first key node and second key node, respectively, over corresponding quantum-secure channels for establishing a Final Key for a communication session between the first and second devices.

5     52.     The system as claimed in claim 51, wherein the system is a satellite quantum key distribution system comprising a plurality of satellites, each satellite including the functionality of quantum key distribution, each satellite in communication with one or more ground receiving stations, and each ground receiving station coupled or connected over a quantum-secure channel to one of the first and second key nodes.

10    53.     The system as claimed in claim 51, wherein the system is a terrestrial quantum key distribution system comprising a plurality of terrestrial quantum transceivers, each transceiver including the functionality of quantum key distribution and coupled or connected over a quantum-secure channel to one of the first and second key nodes.

15    54.     A system comprising:  
an first device;  
a second device;  
an first key node; and  
a second key node, wherein the first device, the second device, the first key node,  
and the second key node each comprise a respective processor unit, memory unit,  
20    and communication interface, the processor unit connected to the memory unit and  
the communication interface, wherein the processor unit, memory unit and  
communication interface are adapted to implement the corresponding steps of the  
computer-implemented method as claimed in any of claims 1 to 49; and  
wherein the first device and second device are configured to communicate with each  
25    other over standard communication channel and with the first key node and second key node,  
respectively, over corresponding classically-secure channels for establishing a Final Key for a  
communication session between the first and second devices.

30    55.     The system as claimed in claim 54, wherein the system includes a classical key distribution system including the functionality of secure key distribution over classically-secure channels to each of the first and second key nodes.