CLAIMS

1. A computer-implemented method, comprising:

pre-storing a user authentication credential in an encrypted form on a companion application running on a second computing device, wherein pre-storing the user authentication credential comprises:

generating a fourth encrypted data by encrypting a third encrypted data having the user authentication credential using a mutual authentication key;

generating a fifth encrypted data by appending a second Message Authentication Code (MAC2) generated using the mutual authentication key to the fourth encrypted data; and

sending the fifth encrypted data to the second computing device via a communication channel;

initiating, by a first computing device of a user, a detection of the second computing device of the user, the detection initiated based at least on an application requiring a user authentication credential to be entered, wherein the application is running on the first computing device and the user authentication credential is pre-stored in an encrypted form on a companion application running on the second computing device;

generating, by the first computing device, the communication channel with the second computing device, the communication channel generated based at least on generating the mutual authentication key by the first computing device and the second computing device;

sending, by the first computing device, a request to receive the user authentication credential to be entered on the companion application on the second computing device via the communication channel;

receiving, by the first computing device, the fifth encrypted data that includes the user authentication credential from the second computing device via the communication channel in the encrypted form;

decrypting, by the first computing device, the user authentication credential received in the fifth encrypted data; and

facilitating, by the first computing device, the entering of the user authentication

credential on the application without user intervention.

2. The method as claimed in claim 1, wherein the first computing device is paired with the second computing device by entering a device identifier by the user on the application running on the first computing device, wherein the device identifier is displayed on the companion application running on the second computing device.

3. The method as claimed in claim 1, wherein the mutual authentication key is generated by the first computing device and the second computing device by applying a Diffie–Hellman key exchange agreement protocol.

4. The method as claimed in claim 1, wherein generating the communication channel with the second computing device, further comprises:

generating a first random value key;

receiving a second random value key generated by the second computing device;

appending the first random value key to the second random value key;

generating a first encrypted data by encrypting the appended first and second random value keys using the mutual authentication key; and

sending the first encrypted data to the second computing device, wherein the second computing device is configured to validate the first encrypted data using the mutual authentication key.

5. The method as claimed in claim 4, further comprising:

sending the first random value key to the second computing device, wherein the second computing device is configured to append the second random value key to the first random value key;

receiving a second encrypted data from the second computing device, the second encrypted data being generated by encrypting the appended first and second random value keys using the mutual authentication key; and

validating the second encrypted data using the mutual authentication key.

6. The method as claimed in claim 1, wherein pre-storing the user authentication

credential in the encrypted form on the companion application running on the second computing device further comprises:

generating an encryption key and a Message Authentication Code (MAC) key, wherein the user authentication credential is entered on the application running on the first computing device by the user;

encrypting the user authentication credential using the encryption key; and

generating the third encrypted data by appending a first Message Authentication Code (MAC1) generated using the MAC key to the encrypted user authentication credential;

wherein the second computing device is configured to:

verify the MAC2 using the mutual authentication key,

decrypt the fourth encrypted data using the mutual authentication key to retrieve the third encrypted data, and

store the third encrypted data.

7. The method as claimed in claim 6,

wherein the second computing device is configured to generate the user authentication credential in the encrypted form by:

generating the fourth encrypted data by encrypting the third encrypted data using the mutual authentication key,

generating the fifth encrypted data by appending the MAC2 generated using the mutual authentication key to the fourth encrypted data, and

sending the fifth encrypted data to the first computing device via the communication channel.

8. The method as claimed in claim 7, further comprising:

verifying the MAC2 using the mutual authentication key;

decrypting the fourth encrypted data using the mutual authentication key to retrieve the third encrypted data;

verifying the MAC1 using the MAC key; and

decrypting the user authentication credential using the encryption key.

9. The method as claimed in claim 1, wherein the application is a payment application and the method further comprises facilitating a payment transaction based at least upon entering the user authentication credential.

10. A computing device, comprising:

a communication interface configured to:

initiate a detection of a pre-paired second computing device of a user, the detection initiated based at least on an application requiring a user authentication credential to be entered, wherein the application is running on the computing device of the user and the user authentication credential is pre-stored in an encrypted form on a companion application running on the second computing device;

send a request to receive the user authentication credential to be entered on the companion application of the second computing device via a communication channel; and

receive the user authentication credential from the second computing device via the communication channel in the encrypted form;

a memory comprising executable instructions; and

a processor communicably coupled to the communication interface, the processor configured to execute the instructions to cause the computing device to at least:

generate the communication channel with the second computing device, the communication channel generated based at least on individually generating a mutual authentication key by the computing device and the second computing device,

generate a fourth encrypted data by encrypting a third encrypted data using the mutual authentication key;

generate a fifth encrypted data by appending a second Message Authentication Code (MAC2) generated using the mutual authentication key to the fourth encrypted data;

send the fifth encrypted data to the second computing device;

decrypt the user authentication credential received in the fifth encrypted data, and

facilitate the entering of the user authentication credential on the application

without user intervention.

11. The computing device as claimed in claim 10, wherein the mutual authentication key is generated by the computing device and the second computing device by applying a Diffie–Hellman key exchange agreement protocol.

12. The computing device as claimed in claim 10, wherein for generating the communication channel with the second computing device, the computing device is further caused to:

generate a first random value key;

receive a second random value key generated by the second computing device;

append the first random value key to the second random value key;

generate a first encrypted data by encrypting the appended first and second random value keys using the mutual authentication key; and

send the first encrypted data to the second computing device, wherein the second computing device is configured to validate the first encrypted data using the mutual authentication key.

13. The computing device as claimed in claim 12, wherein the computing device is further caused to:

send the first random value key to the second computing device, wherein the second computing device is configured to append the second random value key to the first random value key;

receive a second encrypted data from the second computing device, wherein the second encrypted data is generated by encrypting the appended random value keys using the mutual authentication key; and

validate the second encrypted data using the mutual authentication key.

14. The computing device as claimed in claim 10, wherein for pre-storing the user authentication credential in the encrypted form on the companion application running on the second computing device, the computing device is further caused to:

generate an encryption key and a Message Authentication Code (MAC) key, wherein

the user authentication credential is entered on the application running on the computing device by the user;

encrypt the user authentication credential using the encryption key; and

generate a third encrypted data by appending a first Message Authentication Code (MAC1) generated using the MAC key to the encrypted user authentication credential;

wherein the second computing device is configured to:

verify the MAC2 using the mutual authentication key,

decrypt the fourth encrypted data using the mutual authentication key to retrieve the third encrypted data, and

store the third encrypted data.

15. The computing device as claimed in claim 14, wherein the second computing device is configured to:

generate the user authentication credential in the encrypted form by:

generating the fourth encrypted data by encrypting the third encrypted data using the mutual authentication key,

generating the fifth encrypted data by appending the MAC2 generated using the mutual authentication key to the fourth encrypted data, and

send the fifth encrypted data to the computing device via the communication channel.

16. The computing device as claimed in claim 15, wherein the computing device is further caused to:

verify the MAC2 using the mutual authentication key;

decrypt the fourth encrypted data using the mutual authentication key to retrieve the third encrypted data;

verify the MAC1 using the MAC key; and

decrypt the user authentication credential using the encryption key.

17. A computing device, comprising:

a communication module configured to:

receive a request for a user authentication credential to be entered on an application running on a first computing device of a user via a communication channel wherein the first computing device is pre-paired with the computing device of the user and the user authentication credential is pre-stored in an encrypted form on a companion application running on the computing device;

a storage module comprising executable instructions; and

a processing module communicably coupled to the communication module, the processing module configured to execute the instructions to cause the computing device to at least:

generate a communication channel with the first computing device, the communication channel generated based at least on generating a mutual authentication key by the computing device and the first computing device,

generate a fourth encrypted data by encrypting a third encrypted data using a mutual authentication key wherein the third encrypted data is pre-stored,

generate a fifth encrypted data by appending a second Message Authentication Code (MAC2) generated using the mutual authentication key to the fourth encrypted data, and

send the fifth encrypted data using the communication module to the first computing device via the communication channel,

wherein the first computing device is configured to decrypt the fifth encrypted data to retrieve the user authentication credential and facilitate the entering of the user authentication credential on the application without user intervention.

18. The computing device as claimed in claim 17, wherein for generating the communication channel with the first computing device, the computing device is further caused to:

receive a first random value key from the first computing device;

generate a second random value key;

append the second random value key to the first random value key;

generate a second encrypted data by encrypting the appended first and second random value keys using the mutual authentication key; and

send the second encrypted data to the first computing device, wherein the first computing device is configured to validate the second encrypted data using the mutual authentication key.

19. The computing device as claimed in claim 18, wherein the computing device is further caused to:

receive a first encrypted data from the first computing device, wherein the first encrypted data is generated by encrypting the appended first and second random value keys using the mutual authentication key; and

validate the first encrypted data using the mutual authentication key.

20. The computing device as claimed in claim 17, wherein for pre-storing the user authentication credential in the encrypted form on the companion application running on the computing device, the computing device is further caused to:

receive the fifth encrypted data from the first computing device;

verify the MAC2 using the mutual authentication key;

decrypt the fourth encrypted data using the mutual authentication key to retrieve the third encrypted data; and

store the third encrypted data, wherein the third encrypted data is generated by the first computing device by appending a first Message Authentication Code (MAC1) generated using a MAC key to the user authentication credential encrypted using an encryption key.