CLAIMS:

1. A system for computing a private key sk for a device participating in a self-certified identity based signature system comprising:

a secure server configured to:

compute parameters for the device based on a second random number r_{i2} generated by the secure server, a first set of components received from the device, a master secret key *x* and parameters associated with a master public key *mpk*, wherein the first set of components comprises a first random number r_{i1} generated by the device,

transmit the computed parameters to the device; and

the device configured to compute the private key sk based on the received computed parameters and the random number r_{i1}.

2. The system according to claim 1 wherein the step of computing the parameters for the device comprises:

retrieving an arbitrary first value R_{i1}, an identity of the device id_i and a homomorphic encryption value *c* from the first set of components, wherein the arbitrary first value R_{i1} is generated based on the first random number r_{i1} and the homomorphic encryption value *c* is generated by providing the first random number r_{i1} and a prime number q obtained from the parameters associated with the master public key *mpk* to an additive homomorphic encryption function *HEnc( )*;

setting an arbitrary value of the device R_i and a first integer s_{i1} as the parameters for the device,

whereby the arbitrary value of the device R_i is computed based on the arbitrary first value R_{i1} and the second random number r_{i2}; and

whereby the first integer s_{i1} is computed by providing the second random number r_{i2}, the homomorphic encryption value *c*, the master secret key *x*, the arbitrary value of the device R_i, the identity of the device id_i and the prime number q to the additive homomorphic encryption function *HEnc( )*.

3. The system according to claim 2 wherein the step of computing the private key sk comprises:

computing an integer s_i by applying a complementary homomorphic decryption function to the first integer s_{i1} as retrieved from the computed parameters transmitted from the secure server; and

forming the private key sk based on the arbitrary value of the device R_i and the integer s_i.

4. The system according to claims 2 or 3 wherein the arbitrary value of the device R_i is defined by $R_i = (R_{i1})^{r_{i2}}$, the arbitrary first value R_{i1} is defined by $R_{i1} = g^{r_{i1}}$, the homomorphic encryption value c is defined by $c = HEnc({r_{i1}}^{-1} \, mod \, q)$ and the first integer s_{i1} is defined by $s_{i1} = HEnc(r_{i2}) \cdot c^{xH(R_i, id_i) \, mod \, q}$ where HEnc( ) is the additive homomorphic encryption function.

5. The system according to claims 2 or 3 wherein the arbitrary value of the device R_i is defined by $R_i = (R_{i1})^{{r_{i2}}^{-1}}$, the arbitrary first value R_{i1} is defined by $R_{i1} = g^{r_{i1}}$ the homomorphic encryption value c is defined by $c = HEnc({r_{i1}}^{-1})$ and the first integer s_{i1} is defined by $s_{i1} = HEnc({r_{i2}}^{-1}) \cdot c^{xH(R_i, id_i) \, mod \, q}$ where HEnc( ) is the additive homomorphic encryption function.

6. The system according to claims 4 or 5 wherein the integer s_i is defined by $s_i = r_{i1} HDec(s_{i1})$; where HDec( ) is the complementary homomorphic decryption function.

7. The system according to claims 2 or 3 wherein the arbitrary value of the device R_i is defined by $R_i = (R_{i1})^{r_{i2}}$, the arbitrary first value R_{i1} is defined by $R_{i1} = g^{{r_{i1}}^{-1}}$, the homomorphic encryption value c is defined by $c = HEnc(r_{i1})$ and the first integer s_{i1} is defined by $s_{i1} = HEnc(r_{i2}) \cdot c^{xH(R_i, id_i) \, mod \, q}$ where HEnc( ) is the additive homomorphic encryption function.

8. The system according to claim 7 wherein the integer s_i is defined by $s_i = {r_{i1}}^{-1} HDec(s_{i1})$ where HDec( ) is the complementary homomorphic decryption function.

9. A system for generating a common session key SK for encoding digital communications between a first device *i* and a second device *j* that are participating in an extended Transport Layer Security (TLS) protocol or an extended Datagram Transport Layer Security protocol (DTLS), the system comprising:

a secure server configured to instruct:

the first device to compute a private key sk_i based on a first set of parameters received from the secure server, and a random number r_{i1} generated by the first device, wherein the first set of parameters is generated by the secure server based on a second random number r_{i2} generated by the secure server, a first set of components comprising

the first random number r_{i1}, a master secret key *x* and parameters associated with a master public key *mpk*, wherein the first set of components is generated by the first device and transmitted to the secure server, and

the second device to compute a private key sk_j based on a second set of parameters received from the secure server, and a random number r_{j1} generated by the second device, wherein the second set of parameters is generated by the secure server based on a second random number r_{j2} generated by the secure server, a second set of components comprising the first random number r_{j1}, the master secret key *x* and the parameters associated with a master public key *mpk*, wherein the second set of components is generated by the second device and transmitted to the secure server;

the first and second devices configured to use a self-certified Identity Based Signature Scheme to generate the common session key SK for the TLS protocol or the DTLS protocol between the first device and the second device, wherein an identity of the first device id_i is included in the first device's Certificate message and the identity of the second device id_j is included in the second device's Certificate message; wherein the self-certified Identity Based Signature Scheme are dictated by the sk_i and sk_j.

10. The system according to claim 9 wherein the generating the first set of parameters for the first device comprises:

the secure server being configured to:

retrieve an arbitrary first value R_{i1} and the identity of the first device id_i from the first set of components, wherein the arbitrary first value R_{i1} is generated based on the first random number r_{i1};

compute an arbitrary value of the first device R_i and a first integer s_{i1}; and

set the computed arbitrary value of the first device R_i and the first integer s_{i1} as the parameters for the first device,

whereby the arbitrary value of the first device R_i is computed based on the arbitrary first value R_{i1} and the second random number r_{i2}; and

whereby the first integer s_{i1} is computed based on the second random number r_{i2}, the master secret key *x*, the arbitrary value of the first device R_i, the identity of the first device id_i and a prime number q obtained from the parameters associated with the master public key *mpk*.

11. The system according to claim 10 wherein the computing the private key sk_i comprises:

the first device being configured to:

compute an integer s_i based on the first integer s_{i1} as retrieved from the first set of parameters, the first random number r_{i1}, and a prime number q as obtained from the parameters associated with the master public key *mpk*; and

form the private key sk_i based on the arbitrary value of the first device R_i and the integer s_i.

12. The system according to claim 9 wherein the generating the second set of parameters for the second device comprises:

the secure server being configured to:

retrieve an arbitrary first value R_{j1} and the identity of the second device id_j from the second set of components, wherein the arbitrary first value R_{j1} is generated based on the first random number r_{j1};

compute an arbitrary value of the second device R_j and a first integer s_{j1}; and

set the computed arbitrary value of the second device R_j and the first integer s_{j1} as the parameters for the second device,

whereby the arbitrary value of the second device R_j is computed based on the arbitrary first value R_{j1} and the second random number r_{j2}; and

whereby the first integer s_{j1} is computed based on the second random number r_{j2}, the master secret key *x*, the arbitrary value of the second device R_j, the identity of the second device id_j and a prime number q obtained from the parameters associated with the master public key *mpk*.

13. The system according to claim 12 wherein the computing the private key sk_j comprises:

the second device being configured to:

compute an integer s_j based on the first integer s_{j1} as retrieved from the second set of parameters, the first random number r_{j1}, and a prime number q as obtained from the parameters associated with the master public key *mpk*.; and

form the private key sk_j based on the arbitrary value of the second device R_j and the integer s_j.

14. A system for generating a common session key SK for encoding digital communications between a first device *i* and a second device *j* that are participating in an extended Transport Layer Security (TLS) protocol or an extended Datagram Transport Layer Security protocol (DTLS), the system comprising:

a first secure server configured to instruct:

the first device to compute a private key sk_i based on a first set of parameters received from the first secure server, and a random number r_{i1} generated by the first device, wherein the first set of parameters is generated by the first secure server based on a second random number r_{i2} generated by the first secure server, a first set of components comprising the first random number r_{i1}, a master secret key x_i and parameters associated with a master public key mpk_i, wherein the first set of components is generated by the first device and transmitted to the first secure server,

a second secure server configured to instruct:

the second device to compute a private key sk_j based on a second set of parameters received from the second secure server, and a random number r_{j1} generated by the second device, wherein the second set of parameters is generated by the second secure server based on a second random number r_{j2} generated by the second secure server, a second set of components comprising the first random number r_{j1}, the master secret key x_j and the parameters associated with a master public key mpk_j, wherein the second set of components is generated by the second device and transmitted to the second secure server,

wherein the first secure server is located in a different domain from the second secure server;

the first and second devices configured to use a self-certified Identity Based Signature Scheme to generate the common session key SK_{ij} for the TLS protocol between the first device and the second device, wherein an identity of the first device id_i is included in the first device's Certificate message and the identity of the second device id_j is included in the second device's Certificate message; wherein the self-certified Identity Based Signature Scheme are dictated by the sk_i and sk_j.

15. The system according to claim 14 wherein the generating the first set of parameters for the first device comprises:

the first secure server being configured to:

retrieve an arbitrary first value R_{i1} and the identity of the first device id_i from the first set of components, wherein the arbitrary first value R_{i1} is generated based on the first random number r_{i1};

compute an arbitrary value of the first device R_i and a first integer s_{i1}; and

set the computed arbitrary value of the first device R_i and the first integer s_{i1} as the parameters for the first device,

whereby the arbitrary value of the first device R_i is computed based on the arbitrary first value R_{i1} and the second random number r_{i2}; and

whereby the first integer s_{i1} is computed based on the second random number r_{i2}, the master secret key x_i, the arbitrary value of the first device R_i, the identity of the first device id_i and a prime number q obtained from the parameters associated with the master public key mpk_i.

16. The system according to claim 15 wherein the computing the private key sk_i comprises:

the first device being configured to:

compute an integer s_i based on the first integer s_{i1} as retrieved from the first set of parameters, the first random number r_{i1}, and a prime number q as obtained from the parameters associated with the master public key mpk_i.; and

form the private key sk_i based on the arbitrary value of the first device R_i and the integer s_i.

17. The system according to claim 14 wherein the generating the second set of parameters for the second device comprises:

the second secure server being configured to:

retrieve an arbitrary first value R_{j1} and the identity of the second device id_j from the second set of components, wherein the arbitrary first value R_{j1} is generated based on the first random number r_{j1};

compute an arbitrary value of the second device R_j and a first integer s_{j1}; and

set the computed arbitrary value of the second device R_j and the first integer s_{j1} as the parameters for the second device,

whereby the arbitrary value of the second device R_j is computed based on the arbitrary first value R_{j1} and the second random number r_{j2}; and

whereby the first integer s_{j1} is computed based on the second random number r_{j2}, the master secret key x_j, the arbitrary value of the second device R_j, the identity of the second device id_j and a prime number q obtained from the parameters associated with the master public key mpk_j.

18. The system according to claim 17 wherein the computing the private key sk_j comprises:

the second device being configured to:

compute an integer s_j based on the first integer s_{j1} as retrieved from the second set of parameters, the first random number r_{j1}, and a prime number q as obtained from the parameters associated with the master public key mpk_j; and

form the private key sk_j based on the arbitrary value of the second device R_j and the integer s_j.

19. The system according to claim 14 wherein before the secure server is configured to compute parameters for the device based on the second random number r_{i2} generated by the secure server, the secure server is configured to:

receive a zero-knowledge proof result from the device, whereby the zero-knowledge proof result is generated by the device using the first random number r_{i1} and a system parameter λ;

determine from the zero-knowledge proof result if the first random number r_{i1} is equal or less than the system parameter λ; and

compute parameters for the device based on the second random number r_{i2} generated by the secure server, when the first random number r_{i1} is equal or less than the system parameter λ.

20. A system for generating a common session key SK for encoding digital communications between a first device *i* and a second device *j* that are participating in an extended Transport Layer Security (TLS) protocol or an extended Datagram Transport Layer Security protocol (DTLS), the system comprising:

a secure server configured to:

generate a private key sk_i for the first device based on an identity of the first device id_i and generate a private key sk_j for the second device based on an identity of the second device id_j, using a self-certified Identity Based Signature Scheme; and

transmit the private key sk_i to the first device and transmit the private key sk_j to the second device;

the first and second devices configured to use the same self-certified Identity Based Signature Scheme to generate the common session key SK for the TLS protocol or the DTLS protocol between the first device and the second device, wherein the identity of the first device id_i is included in the Certificate message of the first device and the identity of the second device id_j is included in the Certificate message of the second device; wherein the self-certified Identity Based Signature Scheme are dictated by the sk_i and sk_j .

21. A system for generating a common session key SK for encoding digital communications between a first device *i* and a second device *j* that are participating in an extended Transport Layer Security (TLS) protocol or an extended Datagram Transport Layer Security protocol (DTLS), the system comprising:

a first secure server configured to:

generate a private key sk_i for the first device based on an identity of the first device id_i using a self-certified Identity Based Signature Scheme, and transmit the private key sk_i to the first device;

a second secure server configured to:

generate a private key sk_j for the second device based on an identity of the second device id_j, using a self-certified Identity Based Signature Scheme, and transmit the private key sk_j to the second device;

wherein the first secure server is located in a different domain from the second secure server,

the first and second devices configured to use the same self-certified Identity Based Signature Scheme with different parameters or different self-certified Identity Based Signature Schemes to generate the common session key SK for the TLS or DTLS protocol between the first device and the second device, wherein the identity of the first device id_i is included in the Certificate message of the first device and the identity of the second device id_j is included in the Certificate message of the second device; wherein the self-certified Identity Based Signature Scheme are dictated by sk_i and sk_j.

22. A secure server configured to compute a private key sk for a device participating in a self-certified identity based signature system comprising:

a processor; and

a non-transitory media readable by the processor, the media storing instructions that when executed by the processor, cause the processor to:

compute parameters for the device based on a second random number r_{i2} generated by the secure server, a first set of components received from the device, a master secret key *x* and parameters associated with a master public key *mpk*, wherein the first set of components comprises a first random number r_{i1} generated by the device,

transmit the computed parameters to the device such that the device is configured to compute the private key sk based on the received computed parameters and the random number r_{i1}.

23. The secure server according to claim 22 wherein the instructions to compute the parameters for the device comprises:

instructions for directing the processor to:

retrieve an arbitrary first value R_{i1}, an identity of the device id_i and a homomorphic encryption value *c* from the first set of components, wherein the arbitrary first value R_{i1} is generated based on the first random number r_{i1} and the homomorphic encryption value *c* is generated by providing the first random number r_{i1} and a prime number q obtained

from the parameters associated with the master public key *mpk* to an additive homomorphic encryption function *HEnc( )*;

set an arbitrary value of the device R_i and a first integer s_{i1} as the parameters for the device,

whereby the arbitrary value of the device R_i is computed based on the arbitrary first value R_{i1} and the second random number r_{i2}; and

whereby the first integer s_{i1} is computed by providing the second random number r_{i2}, the homomorphic encryption value *c*, the master secret key *x*, the arbitrary value of the first device R_i, the identity of the device id_i and the prime number q to the additive homomorphic encryption function *HEnc( )*.

24. The secure server according to anyone of claims 22-23 wherein before the instructions to compute parameters for the device based on the second random number r_{i2} generated by the secure server, the secure server comprises:

instructions for directing the processor to:

receive a zero-knowledge proof result from the device, whereby the zero-knowledge proof result is generated by the device using the first random number r_{i1} and a system parameter λ;

determine from the zero-knowledge proof result if the first random number r_{i1} is equal or less than the system parameter λ; and

compute parameters for the device based on the second random number r_{i2} generated by the secure server, when the first random number r_{i1} is equal or less than the system parameter λ.

25. A method for computing a private key sk for a device participating in a self-certified identity based signature system comprising:

computing, by a secure server, parameters for the device based on a second random number r_{i2} generated by the secure server, a first set of components received from the device, a master secret key *x* and parameters associated with a master public key *mpk*, wherein the first set of components comprises a first random number r_{i1} generated by the device,

transmitting the computed parameters to the device such that the device computes the private key sk based on the received computed parameters and the random number r_{i1}.

26. The method according to claim 25 wherein the computing the parameters for the device comprises:

retrieving, by the secure server, an arbitrary first value R_{i1}, an identity of the device id_i and a homomorphic encryption value *c* from the first set of components, wherein the arbitrary first value R_{i1} is generated based on the first random number r_{i1} and the homomorphic encryption value *c* is generated by providing the first random number r_{i1} and a prime number q obtained from the parameters associated with the master public key *mpk* to an additive homomorphic encryption function *HEnc( )*;

setting an arbitrary value of the device R_i and a first integer s_{i1} as the parameters for the device,

whereby the arbitrary value of the device R_i is computed based on the arbitrary first value R_{i1} and the second random number r_{i2}; and

whereby the first integer s_{i1} is computed by providing the second random number r_{i2}, the homomorphic encryption value *c*, the master secret key *x*, the arbitrary value of the first device R_i, the identity of the device id_i and the prime number q to the additive homomorphic encryption function *HEnc( )*.

27. A first device for generating a common session key SK for encoding digital communications between the first device *i* and a second device *j* that are participating in an extended Transport Layer Security (TLS) protocol or an extended Datagram Transport Layer Security protocol (DTLS), the first device comprising:

a processor; and

a non-transitory media readable by the processor, the media storing instructions that when executed by the processor, cause the processor to:

receive instructions from a secure server to compute a private key sk_i based on a first set of parameters received from the secure server, and a random number r_{i1} generated by the first device, wherein the first set of parameters is generated by the secure server based on a second random number r_{i2} generated by the secure server, a first set of components comprising the first random number r_{i1}, a master secret key *x* and parameters associated with a master public key *mpk*, wherein the first set of components is generated by the first device and transmitted to the secure server; and

communicate with the second device to generate the common session key SK for the TLS protocol or the DTLS protocol by using a self-certified Identity Based Signature Scheme, wherein an identity of the first device id_i is included in the first device's Certificate message and the identity of the second device id_j is included in the second device's Certificate message; wherein the self-certified Identity Based Signature Scheme

are dictated by a sk_j and the sk_i; wherein the sk_j is a private key computed by the second device.

28. A method for generating a common session key SK for encoding digital communications between a first device *i* and a second device *j* that are participating in an extended Transport Layer Security (TLS) protocol or an extended Datagram Transport Layer Security protocol (DTLS), the method comprising:

generating, by a secure server, a private key sk_i for the first device based on an identity of the first device id_i and generate a private key sk_j for the second device based on an identity of the second device id_j, using a self-certified Identity Based Signature Scheme; and

transmitting the private key sk_i to the first device and transmit the private key sk_j to the second device such that the first and second devices are configured to use the same self-certified Identity Based Signature Scheme to generate the common session key SK for the TLS protocol or the DTLS protocol between the first device and the second device, wherein the identity of the first device id_i is included in the Certificate message of the first device and the identity of the second device id_j is included in the Certificate message of the second device; wherein the self-certified Identity Based Signature Scheme are dictated by the sk_j and the sk_i.

29. A secure server for generating a common session key SK for encoding digital communications between a first device *i* and a second device *j* that are participating in an extended Transport Layer Security (TLS) protocol or an extended Datagram Transport Layer Security protocol (DTLS), the secure server comprising:

a processor; and

a non-transitory media readable by the processor, the media storing instructions that when executed by the processor, cause the processor to:

generate a private key sk_i for the first device based on an identity of the first device id_i and generate a private key sk_j for the second device based on an identity of the second device id_j, using a self-certified Identity Based Signature Scheme; and

transmit the private key sk_i to the first device and transmit the private key sk_j to the second device such that the first and second devices are configured to use the same self-certified Identity Based Signature Scheme to generate the common session key SK for the TLS protocol or the DTLS protocol between the first device and the second device, wherein the identity of the first device id_i is included in the Certificate message of the first device and the identity of the second device id_j is included in the Certificate message of

the second device; wherein the self-certified Identity Based Signature Scheme are dictated by the sk_j and the sk_i.