WHAT IS CLAIMED IS:

1. A computer-implemented method comprising:

loading, with at least one processor, at a merchant system, a software development kit (SDK) wrapper from a payment gateway system via a merchant webpage associated with the merchant system;

executing, with at least one processor, the SDK wrapper, wherein the SDK wrapper loads an SDK core when executed;

determining, with at least one processor, an integrity of the SDK core;

in response to determining the integrity of the SDK core, performing, with at least one processor, a handshake between the SDK wrapper and the SDK core and overloading a real function exported by the SDK wrapper; and

providing, with at least one processor, from the merchant system via the SDK core, a secure payment container request to a payment gateway system.

2. The computer-implemented method of claim 1, further comprising:

receiving, with at least one processor, at the merchant system via the merchant webpage, account data associated with an account of a user, wherein the SDK wrapper is loaded in response to receiving the account data via the merchant webpage, and wherein the secure payment container request includes the account data.

3. The computer-implemented method of claim 1, further comprising:

receiving, with at least one processor, from the payment gateway system, a token associated with the account data in response to providing the secure payment container request.

4. The computer-implemented method of claim 1, wherein determining the integrity of the SDK core includes performing a W3C Subresource iIntegrity check hash on the SDK core.

5. The computer-implemented method of claim 1, wherein the secure payment container request is provided in response to a successful handshake between the SDK wrapper and the SDK core.

6. The computer-implemented method of claim 1, wherein executing the SDK wrapper exposes the real function as a public method, and wherein the real function throws an error in response to being called by the merchant system.

7. The computer-implemented method of claim 1, wherein the SDK wrapper includes an Accept.js library, and wherein the SDK core includes an AcceptCore.js library.

8. A computing system comprising:

one or more processors programmed and/or configured to:

load, at a merchant system, a software development kit (SDK) wrapper from a payment gateway system via a merchant webpage associated with the merchant system;

execute the SDK wrapper, wherein the SDK wrapper loads an SDK core when executed;

determine an integrity of the SDK core;

in response to determining the integrity of the SDK core, perform a handshake between the SDK wrapper and the SDK core and overloading a real function exported by the SDK wrapper; and

provide, from the merchant system via the SDK core, a secure payment container request to a payment gateway system.

9. The computing system of claim 8, wherein the one or more processors are further programmed and/or configured to:

receive, at the merchant system via the merchant webpage, account data associated with an account of a user, wherein the SDK wrapper is loaded in response to receiving the account data via the merchant webpage, and wherein the secure payment container request includes the account data.

10. The computing system of claim 8, wherein the one or more processors are further programmed and/or configured to:

receive, from the payment gateway system, a token associated with the account data in response to providing the secure payment container request.

11. The computing system of claim 8, wherein determining the integrity of the SDK core includes performing a W3C Subresource Integrity check hash on the SDK core.

12. The computing system of claim 8, wherein the secure payment container request is provided in response to a successful handshake between the SDK wrapper and the SDK core.

13. The computing system of claim 8, wherein executing the SDK wrapper exposes the real function as a public method, and wherein the real function throws an error in response to being called by the merchant system.

14. The computing system of claim 8, wherein the SDK wrapper includes an Accept.js library, and wherein the SDK core includes an AcceptCore.js library.

15. A computer program product comprising at least one non-transitory computer-readable medium including program instructions that, when executed by at least one processor, cause the at least one processor to:

load, at a merchant system, a software development kit (SDK) wrapper from a payment gateway system via a merchant webpage associated with the merchant system;

execute the SDK wrapper, wherein the SDK wrapper loads an SDK core when executed;

determine an integrity of the SDK core;

in response to determining the integrity of the SDK core, perform a handshake between the SDK wrapper and the SDK core and overloading a real function exported by the SDK wrapper; and

provide, from the merchant system via the SDK core, a secure payment container request to a payment gateway system.

16. The computer program product of claim 15, wherein the instructions further cause the at least one processor to:

receive, at the merchant system via the merchant webpage, account data associated with an account of a user, wherein the SDK wrapper is loaded in response to receiving the account data via the merchant webpage, and wherein the secure payment container request includes the account data.

17. The computer program product of claim 15, wherein the instructions further cause the at least one processor to:

receive, from the payment gateway system, a token associated with the

~~account data in response to providing the secure payment container request.~~

~~18. The computer program product of claim 15, wherein determining the integrity of the SDK core includes performing a W3C Subresource Integrity check hash on the SDK core.~~

~~19. The computer program product of claim 15, wherein the secure payment container request is provided in response to a successful handshake between the SDK wrapper and the SDK core.~~

~~20. The computer program product of claim 15, wherein executing the SDK wrapper exposes the real function as a public method, and wherein the real function throws an error in response to being called by the merchant system.~~

1. A computer-implemented method, comprising:

receiving, with at least one processor, at a merchant system via a merchant webpage associated with the merchant system, account data associated with an account of a user:

loading, with at least one processor, at [[a]] the merchant system, a software development kit (SDK) wrapper from a payment gateway system via [[a]] the merchant webpage associated with the merchant system, wherein the SDK wrapper is loaded in response to receiving the account data via the merchant webpage;

executing, with at least one processor, the SDK wrapper, wherein the SDK wrapper loads an SDK core when executed;

determining, with at least one processor, an integrity of the SDK core;

in response to determining the integrity of the SDK core, performing, with at least one processor, a handshake between the SDK wrapper and the SDK core and overloading a real function exported by the SDK wrapper; and

providing, with at least one processor, from the merchant system via the SDK core, a secure payment container request to [[a]] the payment gateway system, wherein the secure payment container request includes the account data.

2. The computer-implemented method of claim 1, further comprising:

receiving, with at least one processor, from the payment gateway system, a token associated with the account data in response to providing the secure payment container request.

3. The computer-implemented method of claim 1, wherein determining the integrity of the SDK core includes performing a W3 subresource integrity check hash on the SDK core.

4. The computer-implemented method of claim 1, wherein the secure payment container request is provided in response to a successful handshake between the SDK wrapper and the SDK core.

5. A computer-implemented method, comprising:

loading, with at least one processor, at a merchant system, a software development kit (SDK) wrapper from a payment gateway system via a merchant webpage associated with the merchant system;

executing, with at least one processor, the SDK wrapper, wherein the SDK wrapper loads an SDK core when executed;

determining, with at least one processor, an integrity of the SDK core;

in response to determining the integrity of the SDK core, performing, with at least one processor, a handshake between the SDK wrapper and the SDK core and overloading a real function exported by the SDK wrapper; and

providing, with at least one processor, from the merchant system via the SDK core, a secure payment container request to the payment gateway system,

wherein executing the SDK wrapper exposes the real function as a public method, and wherein the real function throws an error in response to being called by the merchant system.

6. The computer-implemented method of claim 1, wherein the SDK wrapper includes an Accept.js library, and wherein the SDK core includes an AcceptCore.js library.

7. A computing system, comprising:

one or more processors programmed and/or configured to:

receive, at a merchant system via a merchant webpage associated with the merchant system, account data associated with an account of a user;

load, at [[a]] the merchant system, a software development kit (SDK) wrapper from a payment gateway system via [[a]] the merchant webpage associated with the merchant system, wherein the SDK wrapper is loaded in response to receiving the account data via the merchant webpage;

execute the SDK wrapper, wherein the SDK wrapper loads an SDK core when executed;

determine an integrity of the SDK core;

in response to determining the integrity of the SDK core, perform a handshake between the SDK wrapper and the SDK core and overloading a real function exported by the SDK wrapper; and

provide, from the merchant system via the SDK core, a secure payment container request to [[a]] the payment gateway system, wherein the secure payment container request includes the account data.

8. The computing system of claim 7, wherein the one or more processors are further programmed and/or configured to:

receive, from the payment gateway system, a token associated with the account data in response to providing the secure payment container request.

9. The computing system of claim 7, wherein determining the integrity of the SDK core includes performing a W3 subresource integrity check hash on the SDK core.

10. The computing system of claim 7, wherein the secure payment container request is provided in response to a successful handshake between the SDK wrapper and the SDK core.

11. A computing system, comprising: one or more processors programmed and/or configured to:

load, at a merchant system, a software development kit (SDK) wrapper from a payment gateway system via a merchant webpage associated with the merchant system;

execute the SDK wrapper, wherein the SDK wrapper loads an SDK core when executed;

determine an integrity of the SDK core;

in response to determining the integrity of the SDK core, perform a handshake between the SDK wrapper and the SDK core and overloading a real function exported by the SDK wrapper; and

provide, from the merchant system via the SDK core, a secure

payment container request to the payment gateway system,

wherein executing the SDK wrapper exposes the real function as a public method, and wherein the real function throws an error in response to being called by the merchant system.

12. The computing system of claim 7, wherein the SDK wrapper includes an Accept.js library, and wherein the SDK core includes an AcceptCore.js library.

13. A computer program product comprising at least one non-transitory computer-readable medium including program instructions that, when executed by at least one processor, cause the at least one processor to:

receive, at a merchant system via a merchant webpage associated with the merchant system, account data associated with an account of a user;

load, at [[a]] the merchant system, a software development kit (SDK) wrapper from a payment gateway system via [[a]] the merchant webpage associated with the merchant system, wherein the SDK wrapper is loaded in response to receiving the account data via the merchant webpage;

execute the SDK wrapper, wherein the SDK wrapper loads an SDK core when executed;

determine an integrity of the SDK core;

in response to determining the integrity of the SDK core, perform a handshake between the SDK wrapper and the SDK core and overloading a real function exported by the SDK wrapper; and

provide, from the merchant system via the SDK core, a secure payment container request to [[a]] the payment gateway system, wherein the secure payment container request includes the account data.

14. The computer program product of claim 13, wherein the instructions further cause the at least one processor to:

receive, from the payment gateway system, a token associated with the account data in response to providing the secure payment container request.

15. The computer program product of claim 13, wherein determining the integrity of the SDK core includes performing a W3 subresource integrity check hash on the SDK core.

16. The computer program product of claim 13, wherein the secure payment container request is provided in response to a successful handshake between the SDK wrapper and the SDK core.

17. A computer program product comprising at least one non-transitory computer-readable medium including program instructions that, when executed by at least one processor, cause the at least one processor to:

load, at a merchant system, a software development kit (SDK) wrapper from a payment gateway system via a merchant webpage associated with the merchant system;

execute the SDK wrapper, wherein the SDK wrapper loads an SDK core when executed;

determine an integrity of the SDK core;

in response to determining the integrity of the SDK core, perform a handshake between the SDK wrapper and the SDK core and overloading a real function exported by the SDK wrapper; and

provide, from the merchant system via the SDK core, a secure payment container request to the payment gateway system,

wherein executing the SDK wrapper exposes the real function as a public method, and wherein the real function throws an error in response to being called by the merchant system