(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第6830552号
(P6830552)
(45)発行日　令和3年2月17日(2021.2.17)　(24)登録日　令和3年1月28日(2021.1.28)

(51)Int.Cl.　　F I
G06F 21/62 (2013.01)　G06F 21/62
G06F 21/64 (2013.01)　G06F 21/64
G06F 21/55 (2013.01)　G06F 21/55
H04L 9/32 (2006.01)　H04L 9/00　675B

請求項の数 20　(全 23 頁)

(21)出願番号　特願2019-559277(P2019-559277)
(86)(22)出願日　平成31年4月26日(2019.4.26)
(65)公表番号　特表2020-524320(P2020-524320A)
(43)公表日　令和2年8月13日(2020.8.13)
(86)国際出願番号　PCT/CN2019/084510
(87)国際公開番号　WO2019/137563
(87)国際公開日　令和1年7月18日(2019.7.18)
審査請求日　令和1年12月18日(2019.12.18)

早期審査対象出願

(73)特許権者　520015461
アドバンスド　ニュー　テクノロジーズ　カンパニー　リミテッド
英国領ケイマン諸島　グランド　ケイマン　ケーワイ１－９００８　ジョージ　タウン　ホスピタル　ロード　２７　ケイマン　コーポレート　センター
(74)代理人　100188558
弁理士　飯田　雅人
(74)代理人　100205785
弁理士　▲高▼橋　史生

最終頁に続く

(54)【発明の名称】アンチリプレー攻撃認証プロトコル

(57)【特許請求の範囲】
【請求項１】
ブロックチェーンネットワークの安全性を高めるためのコンピュータ実装方法であって、
ブロックチェーンノードとクライアント間の通信セッション中に前記クライアントからトランザクション要求を前記ブロックチェーンノードが受け取るステップであって、前記トランザクション要求が、ブロックチェーン上に記録されるように要求されるトランザクション、および、前記ブロックチェーンノード用に生成された疑似ランダム値と、前記クライアントに固有であり前記通信セッション中に前記クライアントによって入力されたパスワードと、を用いて前記トランザクションをハッシュすることに基づいて計算されたトランザクションハッシュを含む、受け取るステップと、
前記トランザクションハッシュがキャッシュリソースにも前記ブロックチェーンにも以前に記憶されていないことを前記ブロックチェーンノードが決定するステップと、
前記キャッシュリソースに前記トランザクションハッシュを記憶するステップと、
前記トランザクション要求を実行するステップと
を含む、コンピュータ実装方法。
【請求項２】
前記トランザクション要求が前記トランザクションに基づいて生成されたデジタル署名を含み、前記デジタル署名は、前記トランザクションハッシュの非対称暗号化を含む、請求項1に記載されたコンピュータ実装方法。

【請求項３】
前記トランザクションハッシュが前記キャッシュリソースにも前記ブロックチェーンにも以前に記憶されていないと決定するステップが、
前記トランザクションハッシュを使用して前記キャッシュリソースに照会するステップと、
前記トランザクションハッシュの同一のコピーが前記キャッシュリソース中に記憶されていないと決定するステップと
を含む、請求項1に記載のコンピュータ実装方法。
【請求項４】
前記キャッシュリソースが、前記トランザクション要求を受け取る前に前記ブロックチェーンノードによって受け取られたトランザクションハッシュを記憶するブルームフィルタを実施する、請求項3に記載のコンピュータ実装方法。
【請求項５】
前記トランザクションが第1のトランザクションであり、前記トランザクションハッシュが第1のトランザクションハッシュであり、前記方法が、
前記ブロックチェーンノードが、第2のトランザクションおよび第2のトランザクションハッシュを含む第2のトランザクション要求を受け取るステップと、
前記第2のトランザクションハッシュが前記キャッシュリソースおよび前記ブロックチェーンのうち少なくとも１つに以前に記憶されていることを前記ブロックチェーンノードが決定するステップであって、前記キャッシュリソースおよび前記ブロックチェーンのうち少なくとも１つに前記第２のトランザクションハッシュが以前に記憶されていることは、攻撃者が前記トランザクション要求を傍受したネットワーク攻撃を示す、決定するステップと、
前記第２のトランザクション要求に応答して前記ネットワーク攻撃に対抗するために前記クライアントにトランザクション拒否を前記ブロックチェーンノードが送るステップとをさらに含む、請求項4に記載のコンピュータ実装方法。
【請求項６】
前記トランザクションが第1のトランザクションであり、前記トランザクションハッシュが第1のトランザクションハッシュであり、前記方法が、
前記ブロックチェーンノードが、第2のトランザクションおよび第2のトランザクションハッシュを含む第2のトランザクション要求を受け取るステップと、
前記第2のトランザクションハッシュが前記ブロックチェーンに以前に記憶されていることを前記ブロックチェーンノードが決定するステップであって、前記ブロックチェーンに前記第２のトランザクションハッシュが以前に記憶されていることは、攻撃者が前記トランザクション要求を傍受したネットワーク攻撃を示す、決定するステップと、
前記第２のトランザクション要求に応答して前記ネットワーク攻撃に対抗するために前記クライアントにトランザクション拒否を前記ブロックチェーンノードが送るステップとをさらに含む、請求項1に記載のコンピュータ実装方法。
【請求項７】
前記トランザクションが、ブロックチェーンアドレス、トランザクション量、および前記トランザクションの時間のうちの1つまたは複数に関連する情報を含む、請求項1に記載のコンピュータ実装方法。
【請求項８】
動作を実行するためにコンピュータシステムによって実行可能な１つまたは複数の命令を格納する、非一時的コンピュータ可読記憶媒体であって、前記動作は、
ブロックチェーンノードとクライアント間の通信セッション中に前記クライアントからトランザクション要求を前記ブロックチェーンノードが受け取ることであって、前記トランザクション要求が、ブロックチェーン上に記録されるように要求されるトランザクション、および、前記ブロックチェーンノード用に生成された疑似ランダム値と、前記クライアントに固有であり前記通信セッション中に前記クライアントによって入力されたパスワ

ードと、を用いて前記トランザクションをハッシュすることによって計算されたトランザクションハッシュを含む、受け取ることと、
前記トランザクションハッシュがキャッシュリソースにも前記ブロックチェーンにも以前に記憶されていないことを前記ブロックチェーンノードが決定することと、
前記キャッシュリソースに前記トランザクションハッシュを記憶することと、
前記トランザクション要求を実行することと、
を含む、非一時的コンピュータ可読記憶媒体。
【請求項９】
前記トランザクション要求が前記トランザクションに基づいて生成されたデジタル署名を含む、請求項8に記載の非一時的コンピュータ可読記憶媒体。
【請求項１０】
前記トランザクションハッシュが前記キャッシュリソースにも前記ブロックチェーンにも以前に記憶されていないと決定することが、
前記トランザクションハッシュを使用して前記キャッシュリソースに照会することと、
前記トランザクションハッシュの同一のコピーが前記キャッシュリソース中に記憶されていないと決定することと、
を含む、請求項8に記載の非一時的コンピュータ可読記憶媒体。
【請求項１１】
前記キャッシュリソースが、前記トランザクション要求を受け取る前に前記ブロックチェーンノードによって受け取られたトランザクションハッシュを記憶するブルームフィルタである、請求項10に記載の非一時的コンピュータ可読記憶媒体。
【請求項１２】
前記トランザクションが第1のトランザクションであり、前記トランザクションハッシュが第1のトランザクションハッシュであり、前記動作が、
前記ブロックチェーンノードが、第2のトランザクションおよび第2のトランザクションハッシュを含む第2のトランザクション要求を受け取ることと、
前記第2のトランザクションハッシュが前記キャッシュリソースおよび前記ブロックチェーンに以前に記憶されていることを前記ブロックチェーンノードが決定することと、
前記クライアントにトランザクション拒否を前記ブロックチェーンノードが送ることと、
をさらに含む、請求項11に記載の非一時的コンピュータ可読記憶媒体。
【請求項１３】
前記トランザクションが第1のトランザクションであり、前記トランザクションハッシュが第1のトランザクションハッシュであり、前記動作が
前記ブロックチェーンノードが、第2のトランザクションおよび第2のトランザクションハッシュを含む第2のトランザクション要求を受け取ることと、
前記第2のトランザクションハッシュが前記ブロックチェーンに以前に記憶されていることを前記ブロックチェーンノードが決定することと、
前記クライアントにトランザクション拒否を前記ブロックチェーンノードが送ることと、
をさらに含む、請求項8に記載の非一時的コンピュータ可読記憶媒体。
【請求項１４】
前記トランザクションが、ブロックチェーンアドレス、トランザクション量、および前記トランザクションの時間のうちの1つまたは複数に関連する情報を含む、請求項8に記載の非一時的コンピュータ可読記憶媒体。
【請求項１５】
コンピュータ実装されたシステムであって、前記システムは、
１つまたは複数のコンピュータと、
前記１つまたは複数のコンピュータと相互運用可能に接続され、前記１つまたは複数のコンピュータによって実行されると、１つまたは複数の動作を実行する１つまたは複数の

命令を格納する、有形の非一時的な機械可読媒体を有する１つまたは複数のコンピュータメモリデバイスと、を含み、前記１つまたは複数の動作は、
ブロックチェーンノードとクライアント間の通信セッション中に前記クライアントからトランザクション要求を前記ブロックチェーンノードが受け取ることであって、前記トランザクション要求が、ブロックチェーン上に記録されるように要求されるトランザクション、および、前記ブロックチェーンノード用に生成された疑似ランダム値と、前記クライアントに固有であり前記通信セッション中に前記クライアントによって入力されたパスワードと、を用いて前記トランザクションをハッシュすることによって計算されたトランザクションハッシュを含む、受け取ることと、
前記トランザクションハッシュがキャッシュリソースにも前記ブロックチェーンにも以前に記憶されていないことを前記ブロックチェーンノードが決定することと、
前記キャッシュリソースに前記トランザクションハッシュを記憶することと、
前記トランザクション要求を実行することと、
を含む、システム。
【請求項１６】
前記トランザクション要求が前記トランザクションに基づいて生成されたデジタル署名を含む、請求項15に記載のシステム。
【請求項１７】
前記トランザクションハッシュが前記キャッシュリソースにも前記ブロックチェーンにも以前に記憶されていないと決定することが、
前記トランザクションハッシュを使用して前記キャッシュリソースに照会することと、
前記トランザクションハッシュの同一のコピーが前記キャッシュリソース中に記憶されていないと決定することと、
を含む、請求項15に記載のシステム。
【請求項１８】
前記キャッシュリソースが、前記トランザクション要求を受け取る前に前記ブロックチェーンノードによって受け取られたトランザクションハッシュを記憶するブルームフィルタである、請求項17に記載のシステム。
【請求項１９】
前記トランザクションが第1のトランザクションであり、前記トランザクションハッシュが第1のトランザクションハッシュであり、前記動作が、
前記ブロックチェーンノードが、第2のトランザクションおよび第2のトランザクションハッシュを含む第2のトランザクション要求を受け取ることと、
前記第2のトランザクションハッシュが前記キャッシュリソースおよび前記ブロックチェーンに以前に記憶されていることを前記ブロックチェーンノードが決定することと、
前記クライアントにトランザクション拒否を前記ブロックチェーンノードが送ることと、
をさらに含む、請求項18に記載のシステム。
【請求項２０】
前記トランザクションが第1のトランザクションであり、前記トランザクションハッシュが第1のトランザクションハッシュであり、前記動作が
前記ブロックチェーンノードが、第2のトランザクションおよび第2のトランザクションハッシュを含む第2のトランザクション要求を受け取ることと、
前記第2のトランザクションハッシュが前記ブロックチェーンに以前に記憶されていることを前記ブロックチェーンノードが決定することと、
前記クライアントにトランザクション拒否を前記ブロックチェーンノードが送ることと、
をさらに含む、請求項15に記載のシステム。
【発明の詳細な説明】
【背景技術】

【０００１】
コンセンサスネットワークおよび/またはブロックチェーンネットワークと呼ばれることもある分散型台帳システム(DLS)は、参加するエンティティが安全かつ不変にデータを記憶することを可能にする。DLSは、任意の特定の使用事例に言及することのないブロックチェーンネットワークと一般的に呼ばれる。例示的なタイプのブロックチェーンネットワークは、パブリックブロックチェーンネットワーク、プライベートブロックチェーンネットワーク、およびコンソーシアムブロックチェーンネットワークを含み得る。コンソーシアムブロックチェーンネットワークは、コンセンサスプロセスを制御するエンティティの選択グループのために提供され、アクセス制御層を含む。
【０００２】
ネットワークを用いるアプリケーションでは、2つのコンピューティングデバイス間のネットワーク接続にわたって送信されたデータは、リプレー攻撃などといった様々なネットワーク攻撃を受けやすい可能性がある。リプレー攻撃は、攻撃者が、2つのコンピューティングデバイス間で送られた1つまたは複数のメッセージを傍受し、後日にメッセージを(おそらく、いくつかの変更を行って)再送信し、元のメッセージによって促されたものと同じ挙動の実行を促すことを含む。たとえば、攻撃者は、支払い要求を傍受し、要求中の宛先アカウントを攻撃者自身のアカウントで置き換える場合がある。攻撃者は、次いで、変更した支払い要求を送信して、資金を攻撃者自身のアカウントに転送させることを試みる場合がある。
【０００３】
中央サーバと相互作用する複数のクライアントを含む集中型システムでは、アンチリプレー攻撃プロトコルは、たとえば、ただ1回だけ使用できる、各メッセージ中の識別子(たとえば、ナンス)を含むことによって、実施することができる。中央サーバは、どのナンスが使用されたかについて管理して、別のメッセージに既に含まれているナンスすなわち無効ナンスを含むメッセージを拒否することができる。攻撃者は、したがって、同じナンスを有するメッセージを単にリプレーすることはできない。というのは、中央サーバがメッセージを拒否することになるためである。中央サーバのない非集中型のアプリケーションでは、使用されたナンスのリストを維持するのは困難な可能性がある。というのは、メッセージが受け取られると、異なるネットワークデバイスがナンスのリストを更新するのに時間がかかり、同じナンスを有するメッセージを使用するリプレー攻撃が受け入れられ得る時間の窓が残る可能性があるためである。したがって、ブロックチェーンネットワークの安全性を高めるための方法が必要である。
【発明の概要】
【課題を解決するための手段】
【０００４】
本明細書の実施形態は、ブロックチェーンネットワーク上のデータ安全性を高めるためのコンピュータ実施される方法を含む。より具体的には、本明細書の実施形態は、ブロックチェーンネットワークに接続されるクライアントのため、アンチリプレー攻撃認証プロトコルを実施することを対象とする。
【０００５】
本明細書は、1つまたは複数のプロセッサに結合され、命令を記憶した1つまたは複数の非一時的コンピュータ可読記憶媒体も提供し、命令は、1つまたは複数のプロセッサによって実行されると、1つまたは複数のプロセッサに、本明細書に提供される方法の実施形態に従う動作を実施させる。
【０００６】
本明細書は、本明細書に提供される方法を実施するためのシステムをさらに提供する。システムは、1つまたは複数のプロセッサ、および1つまたは複数のプロセッサに結合され、命令を記憶したコンピュータ可読記憶媒体を含み、命令は、1つまたは複数のプロセッサによって実行されると、1つまたは複数のプロセッサに本明細書に提供される方法の実施形態に従う動作を実施させる。

【０００７】
本明細書による方法は、本明細書に記載される態様および特徴の任意の組合せを含み得ることが了解される。すなわち、本明細書による方法は、本明細書に具体的に記載される態様および特徴の組合せに限定されず、提供される態様および特徴の任意の組合せをやはり含む。
【０００８】
本明細書の1つまたは複数の実施形態の詳細は、添付する図面および以下の記載に記述される。本明細書の他の特徴および利点は、記載および図面、ならびに請求項から明らかとなろう。
【図面の簡単な説明】
【０００９】
【図１】本明細書の実施形態を実行するために使用できる環境の例を描く図である。
【図２】本明細書の実施形態による概念的アーキテクチャの例を描く図である。
【図３】本明細書の実施形態による、リプレー攻撃下の分散コンピューティングシステムの例を描く図である。
【図４】本明細書の実施形態による、アンチリプレー攻撃認証プロトコルを実施するためのプロセスの例を描くスイムレーン図である。
【図５】本明細書の実施形態に従って実行できるプロセスの例を描く図である。
【図６】本明細書の実施形態による装置のモジュールの例を描く図である。
【発明を実施するための形態】
【００１０】
様々な図における同様の参照符号は同様の要素を示す。
【００１１】
本明細書の実施形態は、ブロックチェーンネットワーク上のデータ安全性を高めるためのコンピュータ実装方法を含む。より具体的には、本明細書の実施形態は、ブロックチェーンネットワークに接続される各クライアントのため、アンチリプレー安全方式を実施することを対象とする。いくつかの実施形態では、行為は、クライアントからトランザクション要求を受け取るステップと、トランザクションハッシュがキャッシュリソースにもブロックチェーンにも以前に記憶されていないことを決定するステップと、キャッシュリソースにトランザクションハッシュを記憶するステップと、トランザクション要求を実行するステップとを含む。
【００１２】
本明細書の実施形態についてのさらなる状況を提供するため、上で導入されたように、(たとえば、ピアツーピアノードからなる)コンセンサスネットワークおよびブロックチェーンネットワークと呼ばれることもある分散型台帳システム(DLS)は、参加するエンティティが安全かつ不変にトランザクションを実行しデータを記憶することを可能にする。ブロックチェーンという用語は一般的に特定のネットワークおよび/または使用事例と関連するが、本明細書ではブロックチェーンは、任意の特定の使用事例に言及することのないDLSのことを指すために使用される。
【００１３】
ブロックチェーンは、トランザクションが不変である方法でトランザクションを記憶するデータ構造である。したがって、ブロックチェーン上に記録されるトランザクションは、信頼でき信用できる。ブロックチェーンは、1つまたは複数のブロックを含む。チェーン中の各ブロックは、前のブロックの暗号学的ハッシュを含むことによって、チェーンの中のその直前の前のブロックにリンクされる。各ブロックは、タイムスタンプ、それ自体の暗号学的ハッシュ、および1つまたは複数のトランザクションをやはり含む。ブロックチェーンネットワークのノードによって既に検証されているトランザクションは、ハッシュされて、マークル木へと符号化される。マークル木は、木の葉ノードにおけるデータがハッシュされ、木の各枝におけるすべてのハッシュが枝のルートで連結されるデータ構造である。このプロセスは、木全体のルートまで木を進み、ルートは、木のすべてのデータ

を表すハッシュを記憶する。木に記憶されるトランザクションのものである主張するハッシュは、それが木の構造と一致するか決定することによって迅速に検証することができる。

【００１４】

ブロックチェーンが、トランザクションを記憶するための、非集中的な、または少なくとも部分的に非集中的なデータ構造である一方、ブロックチェーンネットワークは、トランザクションをブロードキャストすること、検証すること、および有効にすることなどによって1つまたは複数のブロックチェーンを管理、更新、および維持するコンピューティングノードのネットワークである。上で導入したように、ブロックチェーンネットワークは、パブリックブロックチェーンネットワーク、プライベートブロックチェーンネットワーク、またはコンソーシアムブロックチェーンネットワークとして提供することができる。本明細書の実施形態は、コンソーシアムブロックチェーンネットワークを参照して、本明細書でさらに詳細に記載される。しかし、本明細書の実施形態は、任意の適切なタイプのブロックチェーンネットワークで具体化できることが意図される。

【００１５】

一般的に、コンソーシアムブロックチェーンネットワークは、参加するエンティティ間の私的なものである。コンソーシアムブロックチェーンネットワークでは、コンセンサスプロセスは、コンセンサスノードと呼びことができる許可されたノードの組によって制御され、1つまたは複数のコンセンサスノードは、それぞれのエンティティ(たとえば、金融機関、保険会社)によって運営される。たとえば、10個のエンティティ(たとえば、金融機関、保険会社)のコンソーシアムは、コンソーシアムブロックチェーンネットワークを運営することができ、その各々が、コンソーシアムブロックチェーンネットワーク中で少なくとも1つのノードを運営する。

【００１６】

いくつかの例では、コンソーシアムブロックチェーンネットワーク内で、すべてのノードにわたって複製されるブロックチェーンとして、グローバルブロックチェーンが提供される。すなわち、すべてのコンセンサスノードは、グローバルブロックチェーンに関して完全状態一致となっている。コンセンサス(たとえば、ブロックチェーンに対するブロックの追加の合意)に到達するために、コンソーシアムブロックチェーンネットワーク内でコンセンサスプロトコルが実施される。たとえば、コンソーシアムブロックチェーンネットワークは、下でさらに詳細に記載される、実用的ビザンチンフォールトトレランス(PBFT)コンセンサスを実施することができる。

【００１７】

本明細書の実施形態は、上の状況の観点から、本明細書でさらに詳細に記載される。より具体的には、上で導入したように、本明細書の実施形態は、ブロックチェーンネットワークに接続されるクライアントのため、アンチリプレー攻撃認証プロトコルを実施することを対象とする。

【００１８】

いくつかの実施形態では、開示されるアンチリプレー攻撃認証プロトコルは、各提案されたトランザクションに、固有のトランザクションハッシュをタグ付けし、攻撃者が盗まれたクライアント情報をリプレーすることを防止する。

【００１９】

図1は、本明細書の実施形態を実行するために使用できる環境100の例を描く図である。いくつかの例では、例示的な環境100は、エンティティがコンソーシアムブロックチェーンネットワーク102に参加することを可能にする。例示的な環境100は、コンピューティングデバイス106、108、およびネットワーク110を含む。いくつかの例では、ネットワーク110は、ローカルエリアネットワーク(LAN)、ワイドエリアネットワーク(WAN)、インターネット、またはそれらの組合せを含み、ウェブサイト、ユーザデバイス(たとえば、コンピューティングデバイス)、およびバックエンドシステムを接続する。いくつかの例では、ネットワーク110は、有線および/またはワイヤレス通信リンクを介してアクセスすること

ができる。

【００２０】

描かれた例では、コンピューティングシステム106、108は、各々が、コンソーシアムブロックチェーンネットワーク102中のノードとしての参加を可能にする任意の適切なコンピューティングシステムを含むことができる。例示的なコンピューティングデバイスは、限定しないが、サーバ、デスクトップコンピュータ、ラップトップコンピュータ、タブレットコンピューティングデバイス、およびスマートフォンを含む。いくつかの例では、コンピューティングシステム106、108は、コンソーシアムブロックチェーンネットワーク102と相互作用するために、1つまたは複数のコンピュータ実施されるサービスをホストする。たとえば、1つまたは複数の他のエンティティ(たとえば、他のユーザ)との第1のエンティティのトランザクションを管理するために第1のエンティティが使用するトランザクション管理システムなどといった、第1のエンティティ(たとえば、ユーザA)のコンピュータ実施されるサービスをコンピューティングシステム106がホストすることができる。コンピューティングシステム108は、1つまたは複数の他のエンティティ(たとえば、他のユーザ)との第2のエンティティのトランザクションを管理するために第2のエンティティが使用するトランザクション管理システムなどといった、第2のエンティティ(たとえば、ユーザB)のコンピュータ実施されるサービスをホストすることができる。図1の例では、コンソーシアムブロックチェーンネットワーク102は、ノードのピアツーピアネットワークとして表され、コンピューティングシステム106、108は、それぞれ、コンソーシアムブロックチェーンネットワーク102に参加する第1のエンティティのノードおよび第2のエンティティのノードをそれぞれ提供する。

【００２１】

図2は、本明細書の実施形態による例示的な概念的アーキテクチャ200を描く。例示的な概念的アーキテクチャ200は、エンティティ層202、ホストされるサービス層204、およびブロックチェーンネットワーク層206を含む。描かれた例では、エンティティ層202は、3つの参加者、すなわち参加者A、参加者B、および参加者Cを含み、各々の参加者は、それぞれのトランザクション管理システム208を有する。

【００２２】

描かれた例では、ホストされるサービス層204は、各トランザクション管理システム208のためのインターフェース210を含む。いくつかの例では、それぞれのトランザクション管理システム208は、プロトコル(たとえば、ハイパーテキスト転送プロトコルセキュア(HTTPS))を使用してネットワーク(たとえば、図1のネットワーク110)にわたってそれぞれのインターフェース210で通信する。いくつかの例では、各インターフェース210は、それぞれのトランザクション管理システム208とブロックチェーンネットワーク層206の間の通信接続を提供する。より具体的には、インターフェース210は、ブロックチェーンネットワーク層206のブロックチェーンネットワーク212と通信する。いくつかの例では、インターフェース210とブロックチェーンネットワーク層206の間の通信は、遠隔手続呼び出し(RPC)を使用して行われる。いくつかの例では、インターフェース210は、それぞれのトランザクション管理システム208のためにブロックチェーンネットワークノードを「ホスト」する。たとえば、インターフェース210は、ブロックチェーンネットワーク212にアクセスするためのアプリケーションプログラミングインターフェース(API)を提供する。

【００２３】

本明細書に記載されるように、ブロックチェーンネットワーク212は、ブロックチェーン216に情報を変わらずに記録する複数のノード214を含む、ピアツーピアネットワークとして設けられる。単一のブロックチェーン216が概略的に描かれるが、ブロックチェーン216の複数のコピーが設けられ、ブロックチェーンネットワーク212にわたって維持される。たとえば、各ノード214がブロックチェーンのコピーを記憶する。いくつかの実施形態では、ブロックチェーン215は、コンソーシアムブロックチェーンネットワークに参加している2つ以上のエンティティ間で実施されるトランザクションに関する情報を記憶する。

【００２４】
ブロックチェーン(たとえば、図2のブロックチェーン216)は、ブロックの鎖からできており、各ブロックがデータを記憶する。例示のデータは、2つ以上の参加者間のトランザクションを表すトランザクションデータを含む。本明細書では、トランザクションが非限定の例として使用されるが、ブロックチェーン中に任意の適切なデータ(たとえば、文書、画像、動画、音声)を記憶できることが意図される。例示的なトランザクションは、限定しないが、何かの値(たとえば、資産、製品、サービス、通貨)の交換を含むことができる。トランザクションデータは、ブロックチェーン内に変わらずに記憶される。すなわち、トランザクションデータは、変えることができない。
【００２５】
ブロック中に記憶する前に、トランザクションデータはハッシュされる。ハッシュすることとは、(文字列データとして提供される)トランザクションデータを固定長ハッシュ値に変換する(文字列データとしても提供される)プロセスである。トランザクションデータを得るためにハッシュ値をアンハッシュするのは不可能である。トランザクションデータ中のわずかな変化でさえ、完全に異なるハッシュ値をもたらすことがハッシュすることによって確保される。さらに、上述したように、ハッシュ値は、固定長のものである。すなわち、トランザクションデータのサイズがどうであれ、ハッシュ値の長さは固定される。ハッシュすることは、ハッシュ値を生成するために、トランザクションデータをハッシュ関数を通して処理することを含む。例示的なハッシュ関数は、限定しないが、256ビットハッシュ値を出力する、セキュアハッシュアルゴリズム(SHA)-256を含む。
【００２６】
複数のトランザクションのトランザクションデータがハッシュされ、ブロック中に記憶される。たとえば、2つのトランザクションのハッシュ値が提供され、それら自体がハッシュされて、別のハッシュを提供する。このプロセスが、ブロック中に記憶されるべきすべてのトランザクションについて、単一のハッシュ値が提供されるまで繰り返される。このハッシュ値は、マークルルートハッシュと呼ばれ、ブロックのヘッダに記憶される。トランザクションの何らかの変更は、そのハッシュ値の変更をもたらし、最終的に、マークルルートハッシュの変更をもたらすことになる。
【００２７】
コンセンサスプロトコルを通してブロックチェーンにブロックが追加される。ブロックチェーンネットワーク内の複数のノードがコンセンサスプロトコルに参加し、ブロックをブロックチェーンに追加させる仕事を実施する。そのようなノードはコンセンサスノードと呼ばれる。上で導入されたPBFTは、コンセンサスプロトコルの、非限定の例として使用される。コンセンサスノードは、コンセンサスプロトコルを実行して、トランザクションをブロックチェーンに追加し、ブロックチェーンネットワークの全体の状態を更新する。
【００２８】
さらに具体的には、コンセンサスノードがブロックヘッダを生成し、ブロック中のトランザクションのすべてをハッシュし、ハッシュ値を対に組み合わせて、ブロック中のすべてのトランザクションについて単一のハッシュ値が提供されるまでさらなるハッシュ値を生成する(マークルルートハッシュ)。このハッシュがブロックヘッダに追加される。コンセンサスノードは、ブロックチェーン中の、最も最近のブロック(すなわち、ブロックチェーンに加えられた最後のブロック)のハッシュ値をやはり決定する。コンセンサスノードは、ブロックヘッダに、ナンス値およびタイムスタンプをやはり追加する。
【００２９】
一般的に、PBFTは、ビザンチンフォールト(たとえば、誤動作しているノード、悪意のあるノード)に耐性がある実用的ビザンチンステートマシンの複製を提供する。これは、フォールトは発生すると仮定すること(たとえば、独立したノード故障、および/またはコンセンサスノードによって送信される操作されたメッセージの存在を仮定すること)によって、PBFTで達成される。PBFTでは、コンセンサスノードは、プライマリコンセンサスノードおよびバックアップコンセンサスノードを含むシーケンスで提供される。プライマリ

コンセンサスノードは、周期的に変更され、ブロックチェーンネットワークの世界状態についての合意に達しているブロックチェーンネットワーク内のすべてのコンセンサスノードによって、トランザクションがブロックチェーンに追加される。このプロセスでは、メッセージがコンセンサスノード間で送信され、各コンセンサスノードは、メッセージが指定されたピアノードから受信されたことを証明し、メッセージが送信の間に変更されてないことを検証する。

【００３０】

PBFTでは、コンセンサスプロトコルは、複数のフェーズで提供され、すべてのコンセンサスノードが同じ状態で始まる。最初に、クライアントがプライマリコンセンサスノードに要求を送り、サービス動作を起動する(たとえば、ブロックチェーンネットワーク内でトランザクションを実行する)。要求を受け取ったことに応答して、プライマリコンセンサスノードは、要求をバックアップコンセンサスノードにマルチキャストする。バックアップコンセンサスノードは、要求を実行し、各々がクライアントに返答を送る。クライアントは、閾値の数の返答を受け取るまで待機する。いくつかの実施形態では、クライアントは、f+1個の返答を受け取るのを待ち、ここでfは、ブロックチェーンネットワーク内で耐性を有することができる障害のあるコンセンサスノードの最大数である。最終的な結果は、ブロックチェーンに追加されようとする記録の指令について、十分な数のコンセンサスノードが合意に至り、記録が受け入れられる、または拒否されることとなる。

【００３１】

いくつかのブロックチェーンネットワークでは、トランザクションのプライバシーを維持するために、暗号が実施される。たとえば、ブロックチェーンネットワーク中の他のノードがトランザクションの詳細を識別することができないように、2つのノードがトランザクションをプライベートに保ちたい場合、ノードは、トランザクションデータを暗号化することができる。例示的な暗号法は、限定しないが、対称暗号化および非対称暗号化を含む。対称暗号化は、暗号化(平文から暗号文を生成すること)および復号化(暗号文から平文を生成すること)の両方のために単一の鍵を使用する暗号化プロセスのことを指す。対称暗号化では、複数のノードに対し同じ鍵が利用可能であり、そのため各ノードがトランザクションデータを暗号化/復号化することができる。

【００３２】

非対称暗号化は、各々がプライベート鍵および公開鍵を含むキーの対を使用し、プライベート鍵は、それぞれのノードに対してのみ知られており、公開鍵は、ブロックチェーンネットワーク中の任意のまたはすべての他のノードに知られている。ノードは、データを暗号化するため別のノードの公開鍵を使用することができ、暗号化したデータは、他のノードのプライベート鍵を使用して復号化することができる。たとえば、図2を再び参照して、参加者Aは、参加者Bの公開鍵を使用してデータを暗号化し、暗号化したデータを参加者Bに送ることができる。参加者Bは、参加者Bのプライベート鍵を使用して、暗号化したデータ(暗号文)を復号し、元のデータ(平文)を抽出することができる。あるノードの公開鍵で暗号化したメッセージは、そのノードのプライベート鍵を使用してのみ復号化することができる。

【００３３】

非対称暗号化は、デジタル署名を提供するために使用され、デジタル署名は、トランザクション中の参加者が、トランザクション中の他の参加者ならびにトランザクションの有効性を確認することを可能にする。たとえば、あるノードがメッセージにデジタルで署名をすることができ、参加者Aのデジタル署名に基づいて、そのメッセージがそのノードによって送られたことを別のノードが確認することができる。デジタル署名は、メッセージが移送中に改ざんされていないことを確かめるために使用することもできる。たとえば、図2を再び参照して、参加者Aは、参加者Bにメッセージを送ることになっている。参加者Aは、メッセージのハッシュを生成し、次いで参加者Aのプライベート鍵を使用して、ハッシュを暗号化し、暗号化したハッシュとしてデジタル署名を提供する。参加者Aは、メッセージにデジタル署名を添付し、デジタル署名を付けたメッセージを参加者Bに送る。参

加者Bは、参加者Aの公開鍵を使用してデジタル署名を復号化してハッシュを抽出する。参加者Bはメッセージをハッシュして、ハッシュを比較する。ハッシュが同じ場合、参加者Bは、メッセージが本当に参加者Aからであって、改ざんされなかったことを確認することができる。

【００３４】

図3は、リプレー攻撃下の分散コンピューティングシステム300の例を描く。分散コンピューティングシステム300は、コンピュータネットワークを使用して互いに通信可能に結合されたクライアントとサーバを含む。分散コンピューティングシステム300は、図2に記載されたものと同様のアーキテクチャを有し、トランザクション管理システム208がクライアントであり、インターフェース210とノード214が一緒になってサーバであってよい。一例では、トランザクション管理システム208は、ユーザデバイス上で動作するデジタル財布アプリケーションであってよい。デジタル財布アプリケーションは、ユーザアカウントの金融トランザクションを管理することができ、ノード214と通信して、ブロックチェーン216上に新しいトランザクションを登録する。例示的な金融トランザクションには、デジタル通貨を送受信すること、スマート契約を実行すること、新しいユーザアカウントを開くことなどが含まれる。

【００３５】

いくつかの実施形態では、トランザクション管理システム208(クライアント)とインターフェース210(サーバ)との間の通信チャネルは、1つまたは複数の認証プロトコルを使用して、データの完全性およびデータの安全性を確かめる。たとえば、トランザクション管理システム208は、各トランザクションメッセージ304にメッセージ認証コード(MAC)306をタグ付けすることができる。トランザクションメッセージ304は、送り手のブロックチェーンアドレス、受け手のブロックチェーンアドレス、トランザクションの時間、デジタル通貨の量などといった、トランザクション管理システム208とノード214の間のトランザクションの内容を指定する。下でさらに詳細に議論されるように、MAC306は、トランザクションを認証してリプレー攻撃と戦うため、トランザクションメッセージ304用に一意に生成される。いくつかの例では、トランザクション管理システム208は、トランザクションメッセージ304をハッシュすることによって、MAC306を生成することができる。いくつかの例では、MAC306は、トランザクションメッセージおよびトランザクションメッセージに関係するパスワードをハッシュすることによって生成することができる。

【００３６】

いくつかの実施形態では、攻撃者308が、分散コンピューティングシステム300上でリプレー攻撃を試みる可能性がある。たとえば、攻撃者308は、最初に、トランザクション管理システム208からインターフェース210に送られたデータを傍受し、次いで、傍受したデータを使用してインターフェース210との認証を試みる可能性がある。攻撃者308は、傍受したデータを異なる方法で使用する可能性がある。たとえば、攻撃者308は、インターフェース210またはトランザクション管理システム208との新しい通信セッションで傍受したデータを逐語的に再送信する可能性がある。攻撃者308は、やはり傍受したデータを使用して、MAC306を復号化するのを試みる可能性がある。そのようなリプレー攻撃と戦うために、インターフェース210および/またはトランザクション管理システム208は、上で議論したMAC306を使用して受信したメッセージに認証プロトコルを実施することができる。

【００３７】

認証プロトコルの有効性は、MAC306を生成するため使用された技法に依存し得る。たとえば、MAC306がトランザクション管理システム208のパスワードの正確なコピーである場合、攻撃者308は、サーバにMAC306をリプレーして、トランザクション管理システム208のアカウントにアクセスできるようになる可能性がある。

【００３８】

別の例では、MAC306は、トランザクション管理システム208のアカウントパスワードpと、インターフェース210により発行される課題cの組合せを使用して生成される。たとえば、トランザクション管理システム208がインターフェース210と通信セッションを開始する

と、インターフェース210は、トランザクション管理システム208にランダムに生成した課題cを送ることになる。トランザクション管理システム208は、課題cをアカウントパスワードpに連結し、ハッシュ関数を使用して、ハッシュ出力h(c||p)を生成することができる。例示的なハッシュ関数は、SHA-256、MD-5などを含む。トランザクション管理システム208は、次いで、課題cおよびハッシュ出力h(c||p)を、認証のためにインターフェース210に送る。結果として、攻撃者308は、cおよびh(c||p)を見るが、サーバ(インターフェース210)が異なる課題を発行するため、攻撃者308は、異なる通信セッションでh(c||p)を再使用することが不可能になる。

【００３９】

攻撃者308は、それにもかかわらず、ハッシュ出力h(c||p)をリバースしてパスワードpを取得するのを試みる可能性がある。たとえば、攻撃者308は、レインボーテーブルを使用してハッシュ出力をリバースすることができる。レインボーテーブルは、特定のハッシュ関数について、異なるハッシュ出力をハッシュ入力にマッピングする予め計算した表である。これを行うため、攻撃者308は、正当なサーバを装って、トランザクション管理システム208に偽の課題c'を送る可能性がある。攻撃者308は、使用される特定のハッシュ関数について、c'で以前に計算したレインボーテーブルを有することができる。攻撃者308が、無防備なトランザクション管理システム208からh(c'||p)を受信する場合、攻撃者308は、場合によってハッシュ関数をリバースしてパスワードpを得ることができる。結果として、この認証プロトコルは以前のものよりも安全であるが、攻撃者308がフィッシングの課題c'を能動的に送出できる場合、安全上の欠陥にさらされる。

【００４０】

別の例では、MAC306は、サーバが発行した課題c、クライアントのパスワードp、およびナンスnの組合せを使用して生成される。ナンスnは、サーバおよびクライアントによって選ばれる任意の数である。この認証プロトコルの下で、トランザクション管理システム208は、ハッシュ出力h(n||c||p)を計算し、このハッシュ出力を課題Cおよびナンスnとともにインターフェース210に送る。この認証プロトコルは、攻撃者308が、新しい通信セッション中で傍受したデータを単にリプレーするのを防止することができ、フィッシングc'でハッシュ値をリバースするのを実行不可能にもする。というのは、異なるトランザクションではナンスnが異なるためである。

【００４１】

上の例では、トランザクション管理システム208とインターフェース210の間のトランザクションは、単調増加する数によってインデックス付けされる。インデックスは、安全目的のナンスとして使用することができる。たとえば、トランザクション管理システム208は、トランザクションインデックスをローカルで管理すること、またはトランザクションインデックスを得るためノード214にピングを打つことができる。しかし、単一のアカウントについて複数のトランザクション管理システムが存在する場合、現在のトランザクション数を登録するためのクライアント間の調整は複雑となる可能性がある。さらに、トランザクションは、並列に行われる代わりに直列に行われなければならない。というのは、各トランザクションは、以前のトランザクションからのカウンタに依拠するためである。1つのトランザクションが不適切なナンスを使用した場合、数が違うために、すべての後続の待ち状態のトランザクションは、再スタートするように強いられることになる。いくつかの場合には、トランザクション管理システム208は、並列なトランザクションを可能にするための複数のスロットを管理する。たとえば、各スロットがそれ自体のインデックス数を維持することができる。しかし、多数のスロットを有するのは、計算費用を増加させる。

【００４２】

別の例では、トランザクションは、タイムスタンプを使用することによって、リプレー攻撃からさらに保護される。クライアントは、サーバにピングを打って、各トランザクションについてのタイムスタンプを受け取らなければならず、トランザクションは、指定された時間窓内に終了する必要がある。結果として、攻撃者がクライアントから送られた情

報を首尾よく傍受した場合でさえ、攻撃者は、時間窓が閉じたときには情報を使用することが可能でないことになる。しかし、最新のブロックのタイムスタンプを得ることは、計算費用を追加する可能性があり、サーバとクライアントが同期していないとき、正当な要求を拒否する可能性がある。

【００４３】

図4は、本明細書の実施形態に従って実行できる認証プロトコルを実施するためのプロセス400の例のスイムレーン図を描く。認証プロトコルは、ネットワーク中のリプレー攻撃と戦うことができ、図3に記載されたようなナンスの組を維持する義務からクライアントを解放することができる。いくつかの実施形態では、プロセス400は、1つまたは複数のコンピューティングデバイスを使用して実行される、1つまたは複数のコンピュータ実行可能プログラムを使用して実施することができる。認証プロトコルは、クライアント401とサーバ403の間で実行される。いくつかの例では、クライアント401は、ユーザが動作可能なコンピューティングデバイスであってよい。サーバは、ブロックチェーンネットワークの1つまたは複数のコンセンサスノードであってよい。

【００４４】

最初のステップとして、クライアント401は、トランザクションメッセージmをローカルで生成すること(404)によって、トランザクションを開始する(402)。たとえば、トランザクションは、クライアント401によって制御されるアカウントからブロックチェーンネットワーク中の別のアカウントに、指定された量のデジタル通貨を転送することを表すことができる。クライアント401は、ブロックチェーンネットワークのプロトコルによって指定されたデータ形式を使用してトランザクションメッセージmを生成する。たとえば、トランザクションメッセージmは、送り手のブロックチェーンアドレス、受け手のブロックチェーンアドレス、交換されるデジタル通貨の量、マイニング報酬、タイムスタンプなどを含むことができる。そのため、トランザクションメッセージmは、開始したトランザクションと一意に関連する。

【００４５】

クライアント401は、次いで、トランザクションメッセージmのトランザクションハッシュh(m)を計算する(406)。トランザクションハッシュh(m)を計算する際に使用されるハッシュ関数は、ブロックを生成するためブロックチェーンネットワークによって使用されるハッシュ関数と同じであってよい。

【００４６】

クライアント401は、次いで、ブロックチェーン上でトランザクションを実施して記録するためのトランザクション要求を生成することができる(408)。トランザクション要求は、トランザクションメッセージmおよびトランザクションハッシュ値h(m)を含むことができる。いくつかの場合に、クライアント401は、そのプライベート鍵でトランザクション要求にデジタル署名をすることができる。デジタル署名は、トランザクション要求が変更された場合、無効になる。

【００４７】

クライアント401は、次に、サーバ403と通信セッションを確立して(410)、サーバにトランザクション要求を送る(412)。

【００４８】

トランザクション要求を受け取ったら、サーバ403は、トランザクションハッシュh(m)を使用してブロックチェーン上の過去のトランザクションを検索する。ブロックチェーン上のあらゆるトランザクションは固有のハッシュ値によってインデックス付けされ、トランザクションハッシュh(m)とブロックチェーン上のトランザクションハッシュの間の一致は、トランザクション情報が複製され、リプレー攻撃からのものである可能性があることを示すことになる。いくつかの実施形態では、各ブロックチェーンがいくつかのブロックを含むことができ、各ブロックは、いくつかのトランザクションをさらに含むことができる。サーバ403がブロックチェーンのコピーを記憶するために、サーバ403が特定のトランザクションハッシュを検索するのは、計算コストが高い場合がある。より効率的な検索戦

略を実施するため、サーバ403が既存のトランザクションハッシュにインデックスを付け、それらをキャッシュリソース中に記憶することができる。キャッシュリソースは、通常のメモリまたはデータベースよりも速いアクセス速度を有することができる。いくつかの場合に、キャッシュリソースは、トランザクションハッシュを記憶するための専用とすることができる。

【００４９】

いくつかの場合に、トランザクションハッシュは、トランザクションがブロックチェーン上に記録された後で、キャッシュリソースから除去することができる。そのような場合に、サーバ403は、トランザクションハッシュh(m)が以前に受け取られたかを決定するため、キャッシュリソースおよびブロックチェーンを検索することができる。

【００５０】

いくつかの場合に、受け取ったトランザクションハッシュは、トランザクションがブロックチェーン上で記録されているかにかかわらず、キャッシュハッシュ中に維持することができる。そのような場合に、サーバ403は、最初に、ブルームフィルタなどの効率的なデータ構造に基づいて、キャッシュリソースを検索することができる。

【００５１】

ブルームフィルタは、ある要素がある組のメンバーであるかを決定するため使用される確率的データ構造である。ブルームフィルタに対するクエリーは、フォールスポジティブを戻す場合があるが、決してフォールスネガティブを戻さない。言い換えると、クエリーは、「おそらく組の中」または「確実に組の中にない」のいずれかを戻す。結果として、ブルームフィルタに対するクエリーが、あるトランザクションハッシュが存在しないことを示す場合、トランザクションハッシュがブロックチェーン上に存在しないことが確かである。一方、ブルームフィルタに対するクエリーが、あるトランザクションハッシュが存在することを示す場合、トランザクションハッシュが実際に存在するかを確かめるために、全ブロックチェーン上でさらなる検索を行うことができる。

【００５２】

トランザクション情報を受け取ったら、サーバ403は、ブルームフィルタに関連するキャッシュリソース中でトランザクションハッシュh(m)を検索する(414)。キャッシュリソースは、以前に受け取ったトランザクションハッシュを記憶することができる。検索がネガティブを戻し、トランザクションハッシュがブロックチェーン上に存在しないことを示す場合、関連するトランザクションは、正当なトランザクションであって、サーバ403は、そのトランザクションで進める(416)。トランザクションハッシュは、次いで、ブルームフィルタに関連するキャッシュリソースに記憶される(418)。

【００５３】

検索がポジティブを戻す場合、トランザクションハッシュh(m)がブロックチェーン上に存在する可能性も存在しない可能性もある。ブロックチェーン上のh(m)の存在をさらに決定するため、サーバ403は、トランザクションハッシュh(m)の第2の検索を実施する(420)。このとき、サーバ403は、サーバ403に関連する全ブロックチェーンでトランザクションハッシュh(m)を検索することになる。検索がネガティブを戻し、トランザクションが正当であることを示す場合、サーバ403は、そのトランザクションで再び進め(416)、ブロックチェーンネットワーク上でトランザクションをブロードキャストする。トランザクションが、たとえば、プルーフオブワークプロセスを通して有効化され、クライアント401が十分な残高を有することが確かな場合、トランザクションはブロックチェーン上に記録されることになる。

【００５４】

第2の検索がポジティブを戻す場合、トランザクションハッシュh(m)は既にブロックチェーン中に存在する。これは、クライアント401が以前に使用された情報をサーバに送ることを試みている-リプレー攻撃の可能性を示す。結果として、クライアント401は、失敗メッセージを受け取り、トランザクションを中断することができる(422)。

【００５５】

図5は、本明細書の実施形態による、アンチリプレー攻撃認証プロトコルを実施するためのプロセス500の例のフローチャートを描く。プロセス500は、たとえば、図4のインターフェース210およびノード214を備えるサーバ403といった、サーバの観点から記載される。サーバ403は、図2に記載されたような、ブロックチェーンネットワーク212のサーバであってよい。

【００５６】

最初のステップとして、サーバ403は、たとえば、図4のクライアント401といった、クライアントからの通信セッションを確立する(502)。通信セッションは、クライアント401とサーバ403の間の双方向のデータ交換を可能にする。たとえば、トランザクション要求は、クライアント401によって制御されるデジタル資産の転送を含むことができ、クライアント401は、たとえば、ブロックチェーン216といったブロックチェーン上にトランザクションを記録するようにサーバ403に要求することができる。サーバ403とクライアント401は、認証目的で、ある種の秘密情報を共有する。たとえば、サーバ403は、クライアント401のパスワードのコピーを記憶することができる。

【００５７】

通信セッションを確立したことに応答して、サーバ403は、クライアント401に課題を発行する(504)。課題は、通信セッションのため特に生成されたランダムまたは擬似ランダムな値である。異なる通信セッションは、異なる課題を使用することになる。

【００５８】

課題を発行したことに応答して、サーバ403は、クライアント401からトランザクション要求を受け取る(506)。たとえば、トランザクション要求は、発行された課題から計算されるハッシュ値、クライアントに関連するパスワード、およびサーバ403によって維持されるブロックチェーン中に記憶するため要求されたトランザクションのハッシュを含むことができる。要求されたトランザクションのハッシュを計算するためクライアント401に使用されるハッシュ関数は、ブロックチェーン216上でトランザクションをハッシュしてインデックス付けするためブロックチェーンネットワーク212によって使用される同じハッシュ関数である。トランザクション要求は、トランザクションメッセージハッシュを含む(508)。

【００５９】

サーバ403は、ここで、要求されたトランザクションがブロックチェーン216に含まれるかを決定する(508)。たとえば、サーバ403は、ブロックチェーンに以前に記憶したすべてのトランザクションのハッシュ値を記憶するキャッシュリソースに照会することができる。クエリー性能を改善するため、サーバ403は、ブルームフィルタを使用して、ブロックチェーン216が要求されたトランザクションのハッシュを既に含むかを決定することができる。

【００６０】

要求されたトランザクションのハッシュがブロックチェーン216に含まれないとサーバ403が決定した場合、サーバ403は、そのトランザクションで進める(512)。たとえば、サーバ403は、クライアント301からのパスワードを検証するために進むことができる。検証が成功である場合、サーバ403は、有効化するためにブロックチェーンネットワーク212に現在のトランザクションをブロードキャストすることができる。

【００６１】

一方、要求されたトランザクションのハッシュがブロックチェーン216に既に含まれるとサーバ403が決定した場合、サーバ403は、クライアント301にトランザクション拒否を送ることになる(510)。複製トランザクションの存在は、クライアント301は悪意があり、サーバアクセスを得るために、盗まれた情報をリプレーしていることを示す可能性がある。

【００６２】

図6は、本明細書の実施形態による装置600のモジュールの例を描く図である。装置600がブロックチェーンノードの例示的な実施形態であってよい。装置600は、上で記載した

実施形態に対応することができ、装置600は、以下、すなわち、クライアントからのトランザクション要求を受け取るための受信モジュール602であって、トランザクション要求が、ブロックチェーン上に記録されるように要求されたトランザクションと、トランザクションをハッシュすることに基づいて計算されたハッシュとを含む、受信モジュール602と、トランザクションハッシュがキャッシュリソースまたはブロックチェーンの中に以前記憶されていないかを決定するための決定モジュール604と、トランザクションハッシュをキャッシュリソースに記憶するための記憶モジュール606と、トランザクション要求を実行するための実行モジュール608とを含む。

【００６３】

本明細書に記載される技法は、1つまたは複数の技術的な効果をもたらす。いくつかの実施形態では、本技法によって、繰り返される要求がブロックチェーンネットワークによって処理されてコンセンサスプロセスに提出される前に、同じトランザクション要求を複数回提出する試み(すなわち、リプレー攻撃)をブロックチェーンネットワークが検出することが可能になる。このことによって、ブロックチェーンネットワークがこれらの無効なトランザクションを処理するのを回避して、より高いトランザクションスループットをもたらすことが可能になる。いくつかの実施形態では、本技法は、複数のクライアント間で協調しなければならないナンスまたは他の値の使用をせず、それによって、より簡単なクライアント実施形態および意図しない複製要求の可能性の低下がもたらされる。

【００６４】

本主題の記載される実施形態は、1つまたは複数の特徴を、単独でまたは組み合わせて含むことができる。たとえば、第1の実施形態では、ブロックチェーンネットワークの安全性を高めるための方法は、クライアントからトランザクション要求を受け取るステップであって、トランザクション要求が、ブロックチェーン上に記録されるように要求されるトランザクションと、トランザクションをハッシュすることに基づいて計算されたトランザクションハッシュとを含む、ステップと、トランザクションハッシュがキャッシュリソースにもブロックチェーンにも以前に記憶されていないことを決定するステップと、キャッシュリソースにトランザクションハッシュを記憶するステップと、トランザクション要求を実行するステップとを含む。

【００６５】

上記および他の記載された実施形態は、各々が任意選択で以下の特徴のうちの1つまたは複数を含むことができる。

【００６６】

以降の特徴のいずれかと組合せ可能な第1の特徴としては、トランザクション要求が、トランザクションに基づいて生成されたデジタル署名を含むことが規定される。

【００６７】

以前または以降の特徴のいずれかと組合せ可能な第2の特徴としては、トランザクションハッシュがキャッシュリソースにもブロックチェーンにも以前に記憶されていないと決定するステップが、トランザクションハッシュを使用してキャッシュリソースに照会するステップおよびトランザクションハッシュの同一のコピーがキャッシュリソース中に記憶されていないと決定するステップを含むことが規定される。

【００６８】

以前または以降の特徴のいずれかと組合せ可能な第3の特徴としては、キャッシュリソースは、トランザクション要求を受け取る前にブロックチェーンノードによって受け取られたトランザクションハッシュを記憶するブルームフィルタであることが規定される。

【００６９】

以前または以降の特徴のいずれかと組合せ可能な第4の特徴としては、トランザクションが第1のトランザクションであり、トランザクションハッシュが第1のトランザクションハッシュであり、ブロックチェーンは、第2のトランザクションおよび第2のトランザクションハッシュを含む第2のトランザクション要求をさらに受け取り、第2のトランザクションハッシュがキャッシュリソースおよびブロックチェーンに以前に記憶されていることを

決定し、クライアントにトランザクション拒否を送ることが規定される。

【００７０】

以前または以降の特徴のいずれかと組合せ可能な第5の特徴としては、トランザクションが第1のトランザクションであり、トランザクションハッシュが第1のトランザクションハッシュであり、ブロックチェーンノードは、第2のトランザクションおよび第2のトランザクションハッシュを含む第2のトランザクション要求をさらに受け取り、第2のトランザクションハッシュがブロックチェーンに以前に記憶されていることを決定し、クライアントにトランザクション拒否を送ることが規定される。

【００７１】

以前または以降の特徴のいずれかと組合せ可能な第6の特徴としては、トランザクションが、ブロックチェーンアドレス、トランザクション量、およびトランザクションの時間のうちの1つまたは複数に関連する情報を含むことが規定される。

【００７２】

本主題の実施形態および本明細書に記載された行為および動作は、本明細書で開示された構成およびそれらの構成上の等価物を含む、デジタル電子回路、有形に具体化されたコンピュータソフトウェアまたはファームウェア、コンピュータハードウェア、またはそれらの1つまたは複数の組合せで実施することができる。本明細書に記載される主題の実施形態は、データ処理装置が実行するため、またはデータ処理装置の動作を制御するため、コンピュータプログラム担体上に符号化された、たとえば、コンピュータプログラム命令の1つまたは複数のモジュールといった、1つまたは複数のコンピュータプログラムとして実施することができる。たとえば、コンピュータプログラム担体は、その上に命令を符号化または記憶した1つまたは複数のコンピュータ可読記憶媒体を含むことができる。担体は、磁気、光磁気、もしくは光ディスク、固体ドライブ、ランダムアクセスメモリ(RAM)、読取り専用メモリ(ROM)、または他のタイプの媒体などの、有形の非一時的コンピュータ可読媒体であってよい。代替または追加として、担体は、データ処理装置が実行するため、好適な受信器装置に送信する情報を符号化するために生成される、たとえば、機械生成した電気、光、または電磁信号といった人工的に生成した伝播信号であってよい。コンピュータ記憶媒体は、機械可読記憶デバイス、機械可読記憶基板、ランダムもしくはシリアルアクセスメモリデバイス、もしくはそれらの1つもしくは複数の組合せであってよく、または部分であってよい。コンピュータ記憶媒体は、伝播信号でない。

【００７３】

プログラム、ソフトウェア、ソフトウェアアプリケーション、アプリ、モジュール、ソフトウェアモジュール、エンジン、スクリプト、またはコードと呼ばれ、または記載されることもあるコンピュータプログラムは、コンパイル型もしくはインタープリタ型言語、または宣言型もしくは手続型言語を含む任意の形のプログラミング言語で書くことができ、スタンドアロンプログラム、または、1つまたは複数の場所においてデータ通信ネットワークによって相互接続される1つまたは複数のコンピュータを含むことができるコンピューティング環境で実行するのに好適な、モジュール、構成要素、エンジン、サブルーチン、もしくは他のユニットを含む任意の形で展開することができる。

【００７４】

コンピュータプログラムは、ファイルシステム中のファイルに対応することができるが、必ずしも対応する必要はない。コンピュータプログラムは、マークアップ言語文書、対象のプログラム専用の単一のファイル、またはたとえば、1つもしくは複数のモジュール、サブプログラム、もしくはコードの部分を記憶するファイルといった複数の協調したファイルに記憶される、たとえば、1つもしくは複数のスクリプトといった他のプログラムまたはデータを保持するファイルの一部に記憶することができる。

【００７５】

コンピュータプログラムの実行のためのプロセッサは、例として、汎用マイクロプロセッサと専用マイクロプロセッサの両方、および任意の種類のデジタルコンピュータの任意の1つまたは複数のプロセッサを含む。一般的に、プロセッサは、プロセッサに結合され

た非一時的コンピュータ可読媒体から、実行のためのコンピュータプログラムの命令ならびにデータを受け取る。

【００７６】

「データ処理装置」という用語は、例として、プログラム可能プロセッサ、コンピュータ、または複数のプロセッサもしくはコンピュータを含む、データを処理するためのすべての種類の装置、デバイス、および機械を包含する。データ処理装置は、たとえば、FPGA(フィールドプログラム可能ゲートアレイ)、ASIC(特定用途向け集積回路)、またはGPU(グラフィックス処理ユニット)といった専用論理回路を含むことができる。装置は、ハードウェアに加えて、たとえば、プロセッサファームウェア、プロトコルスタック、データベース管理システム、オペレーティングシステム、またはそれらのうちの1つまたは複数の組合せを構成するコードといった、コンピュータプログラムのための実行環境を作るコードをやはり含むことができる。

【００７７】

本明細書で記載されるプロセスおよび論理フローは、1つまたは複数のコンピュータプログラムを実行する1つまたは複数のコンピュータまたはプロセッサが実施して、入力データに作用して出力を生成することによって動作を実施することができる。プロセスおよび論理フローは、たとえば、FPGA、ASIC、もしくはGPUといった専用論理回路によって、または専用論理回路と1つまたは複数のプログラムしたコンピュータの組合せによって実施することもできる。

【００７８】

コンピュータプログラムの実行に好適なコンピュータは、汎用マイクロプロセッサもしくは専用マイクロプロセッサもしくはそれらの両方、または任意の種類の中央処理装置に基づくことができる。一般的に、中央処理装置は、読取り専用メモリまたはランダムアクセスメモリまたはそれらの両方から命令およびデータを受け取る。コンピュータの要素は、命令を実行するための中央処理装置、および命令およびデータを記憶するための1つまたは複数のメモリデバイスを含み得る。中央処理装置およびメモリは、専用論理回路によって補助すること、または専用論理回路中に組み込むことができる。

【００７９】

一般的に、コンピュータは、また、1つもしくは複数の記憶デバイスを含むこと、または1つもしくは複数の記憶デバイスからデータを受け取るため、もしくは1つもしくは複数の記憶デバイスにデータを転送するために動作可能に結合することとなる。記憶デバイスは、たとえば、磁気、光磁気、もしくは光ディスク、固体ドライブ、または任意の他のタイプの非一時的コンピュータ可読記憶媒体であってよい。しかし、コンピュータがそのようなデバイスを持つ必要はない。したがって、コンピュータは、ローカルおよび/またはリモートである、1つまたは複数のメモリなどの1つまたは複数の記憶デバイスに結合することができる。たとえば、コンピュータは、コンピュータと一体となった構成要素である1つもしくは複数のローカルメモリを含むことができ、またはコンピュータは、クラウドネットワークにある1つもしくは複数のリモートメモリに結合することができる。さらに、コンピュータは、ほんのいくつか例を挙げれば、たとえば、モバイル電話、携帯情報端末(PDA)、モバイルオーディオもしくはビデオプレイヤー、ゲームコンソール、全地球測位システム(GPS)受信器、またはたとえば、ユニバーサルシリアルバス(USB)フラッシュドライブといった携帯型記憶デバイスといった別のデバイスに埋め込むことができる。

【００８０】

構成要素は、直接的または1つもしくは複数の中間の構成要素を介してのいずれかで、互いに電気的または光学的に接続するなど、可換であることによって互いに「結合」することができる。構成要素は、構成要素のうちの1つが他のものに組み込まれる場合、互いに「結合」することがやはりできる。たとえば、プロセッサへと組み込まれた記憶構成要素(たとえば、L2キャッシュ構成要素)は、プロセッサに「結合」される。

【００８１】

ユーザとの相互作用を実現するため、本明細書に記載される主題の実施形態は、ユーザ

に情報を表示するための、たとえば、LCD(液晶ディスプレイ)モニターといったディスプレイデバイス、および、たとえば、キーボードおよびたとえば、マウス、トラックボール、またはタッチパッドといったポインティングデバイスといった、ユーザがコンピュータに入力を提供することができる入力デバイスを有するコンピュータ上に実装すること、またはコンピュータと通信するように構成することができる。同様に、他の種類のデバイスを使用してユーザとの相互作用を実現することができる。たとえば、ユーザへ提供されるフィードバックは、たとえば視覚フィードバック、音声フィードバック、または触覚フィードバックといった任意の形の感覚フィードバックであってよく、ユーザからの入力は、音響、音声、または触覚入力を含む任意の形で受け取ることができる。加えて、コンピュータは、ユーザに使用されるデバイスとの間で文書を送受信することによって、たとえば、ウェブブラウザから受け取った要求に応答してユーザのデバイス上のウェブブラウザにウェブページを送ることによって、またはたとえば、スマートフォンもしくは電子タブレットといったユーザデバイス上で動作するアプリと相互作用することによって、ユーザと相互作用することができる。また、コンピュータは、たとえば、メッセージングアプリケーションが動作しているスマートフォンといった、個人用デバイスにテキストメッセージまたは他の形のメッセージを送ること、および返事としてユーザから応答メッセージを受け取ることによって、ユーザと相互作用することができる。

【００８２】

本明細書は、システム、装置、およびコンピュータプログラム構成要素に関して、「構成される(configured to)」という用語を使用する。特定の動作または行為を実施するように構成される1つまたは複数コンピュータのシステムとは、そのシステムが、動作中にシステムに動作または行為を実施させるソフトウェア、ファームウェア、ハードウェア、またはそれらの組合せを、システム自体にインストールしていることを意味する。特定の動作または行為を実施するように構成される1つまたは複数コンピュータプログラムとは、データ処理装置が実行すると、装置に動作または行為を実施させる命令を1つまたは複数のプログラムが含むことを意味する。特定の動作または行為を実施するように構成される専用論理回路とは、動作または行為を実施する電子論理を回路が有することを意味する。

【００８３】

本明細書が多くの具体的な実施形態の詳細を含んでいる一方で、これらは、請求項自体によって規定される、特許請求されているものの範囲についての制限と考えるべきでなく、むしろ、特定の実施形態に固有であってよい特徴の記載と考えるべきである。別個の実施形態の文脈で本明細書で記載されるある種の特徴は、単一の実施形態で組み合わせて具体化することもできる。逆に、単一の実施形態の文脈で記載される様々な特徴は、複数の実施形態で別個に、または任意の好適な下位の組合せで具体化することもできる。さらに、特徴は、上で、ある種の組合せで機能すると記載される場合があり、最初にそのように特許請求されさえするが、特許請求される組合せからの1つまたは複数の特徴は、いくつかの場合に、組合せから切り取られる可能性があり、特許請求は、下位の組合せまたは下位の組合せの変形形態を対象とする場合がある。

【００８４】

同様に、動作は、特定の順番で、図に描かれ、請求項に言及される一方、これを、所望の結果に到達するために、そのような動作が示される特定の順番でもしくは連続的な順番で実施されること、またはすべての図示された動作が実施されることを必要とすると理解するべきではない。ある種の状況では、マルチタスクおよび並列処理が有利となる場合がある。さらに、上に記載された実施形態における様々なシステムモジュールおよび構成要素の区切りは、すべての実施形態でそのような区切りを必要とすると理解するべきでなく、記載されたプログラム構成要素およびシステムは、単一のソフトウェア製品に全体的に一緒に組み込むことができ、または複数のソフトウェア製品にパッケージすることができると理解するべきである。

【００８５】

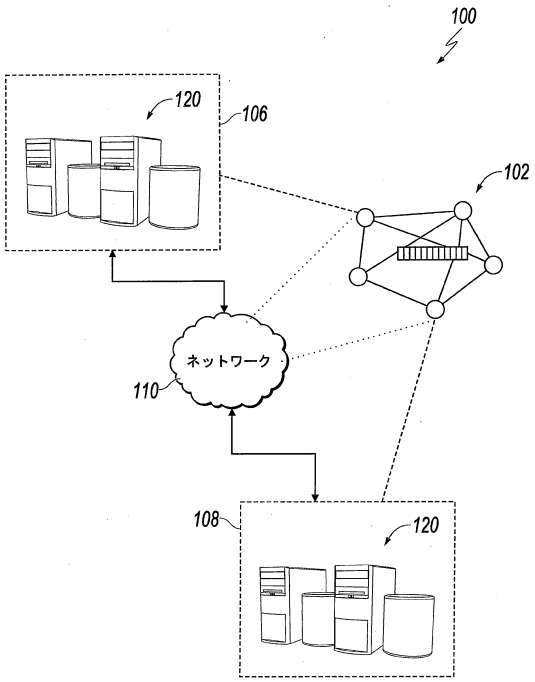
本主題の特定の実施形態が記載されてきた。他の実施形態は、以下の請求項の範囲内である。たとえば、請求項中で言及される行為を、異なる順番で実施して、依然として望ましい結果を達成することができる。一例として、添付図面に描かれたプロセスは、望ましい結果を達成するために、必ずしも示される特定の順番または連続的な順番を必要としない。いくつかの場合に、マルチタスクおよび並列処理が有利となる場合がある。

【符号の説明】

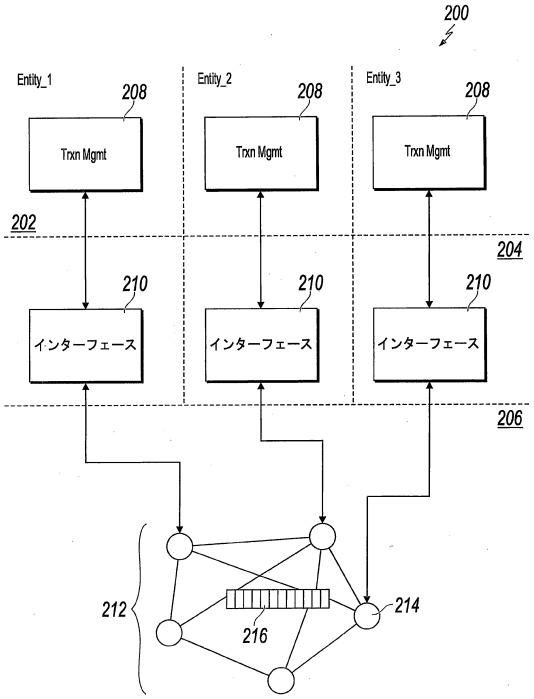
【００８６】

100 環境
102 コンソーシアムブロックチェーンネットワーク
106 コンピューティングデバイス
108 コンピューティングデバイス
110 ネットワーク
200 概念的アーキテクチャ
202 エンティティ層
204 ホストされるサービス層
206 ブロックチェーンネットワーク層
208 トランザクション管理システム、クライアント
210 インターフェース、サーバ
212 ブロックチェーンネットワーク
214 ノード
215 ブロックチェーン
216 ブロックチェーン
300 分散コンピューティングシステム
304 トランザクションメッセージ
306 メッセージ認証コード(MAC)
308 攻撃者
400 プロセス
401 クライアント
403 サーバ
500 プロセス
600 装置
602 受信モジュール
604 決定モジュール
606 記憶モジュール
608 実行モジュール

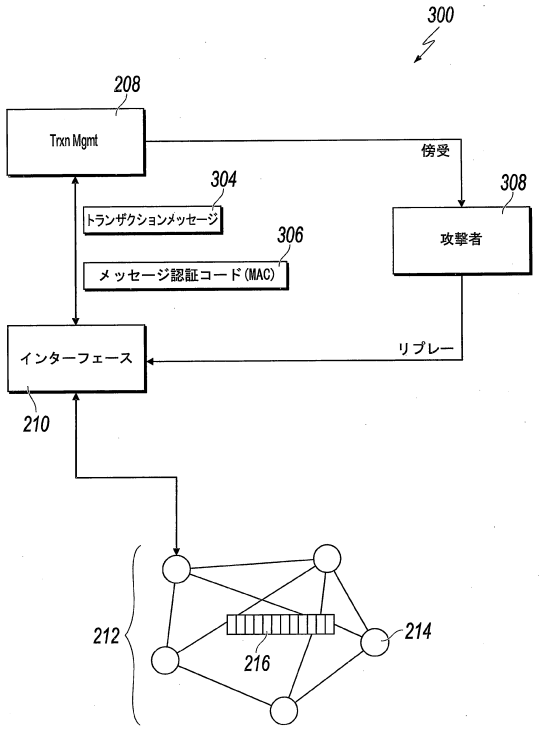
【図１】


100
120
106
102
ネットワーク
110
108
120

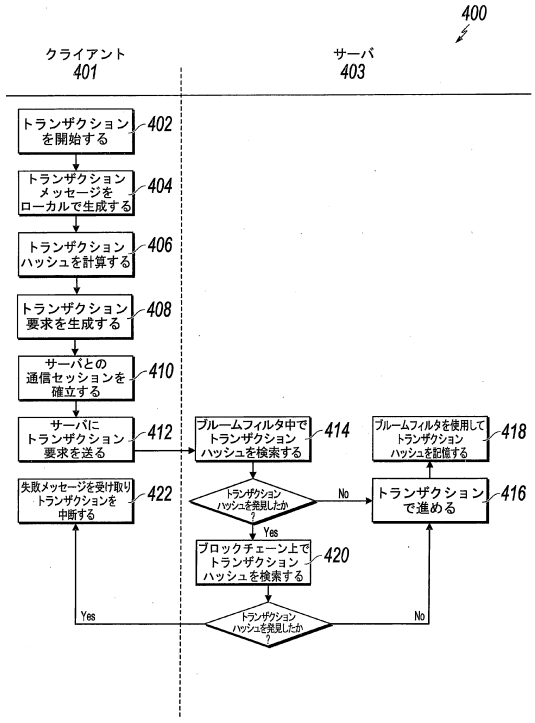
【図２】


200
Entity_1
208
Trxn Mgmt
Entity_2
208
Trxn Mgmt
Entity_3
208
Trxn Mgmt
202
204
210
インターフェース
210
インターフェース
210
インターフェース
206
212
214
216

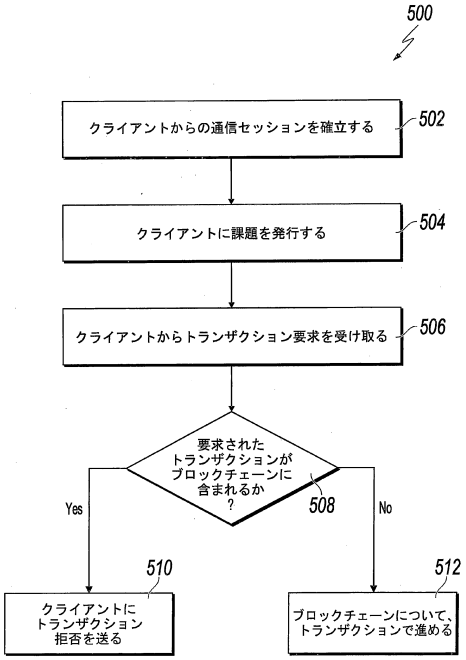
【図３】


300
208
Trxn Mgmt
傍受
304
トランザクションメッセージ
308
攻撃者
306
メッセージ認証コード(MAC)
リプレー
インターフェース
210
212
214
216

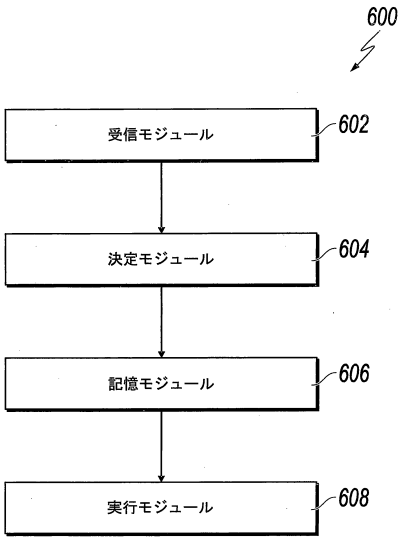
【図４】


400
クライアント
401
サーバ
403
トランザクションを開始する
402
トランザクションメッセージをローカルで生成する
404
トランザクションハッシュを計算する
406
トランザクション要求を生成する
408
サーバとの通信セッションを確立する
410
サーバにトランザクション要求を送る
412
ブルームフィルタ中でトランザクションハッシュを検索する
414
ブルームフィルタを使用してトランザクションハッシュを記憶する
418
失敗メッセージを受け取りトランザクションを中断する
422
トランザクションハッシュを発見したか？
No
トランザクションで進める
416
Yes
ブロックチェーン上でトランザクションハッシュを検索する
420
Yes
トランザクションハッシュを発見したか？
No

【図５】


500
クライアントからの通信セッションを確立する
502
クライアントに課題を発行する
504
クライアントからトランザクション要求を受け取る
506
要求された
トランザクションが
ブロックチェーンに
含まれるか
？
508
Yes
No
510
クライアントに
トランザクション
拒否を送る
512
ブロックチェーンについて、
トランザクションで進める

【図６】


600
受信モジュール
602
決定モジュール
604
記憶モジュール
606
実行モジュール
608

フロントページの続き

(72)発明者　ホン・ル
中華人民共和国・３１１１２１・ハンジョウ・ユ・ハン・ディストリクト・ウェスト・ウェン・イ・ロード・ナンバー・９６９・ビルディング・３・５／エフ・アリババ・グループ・リーガル・デパートメント

審査官　平井　誠

(56)参考文献　国際公開第２０１９／０７２３１２（WO，Ａ２）
特表２０２０－５０５７９９（ＪＰ，Ａ）
国際公開第２０１８／２０５９７１（WO，Ａ１）
特表２０２０－５１６０８９（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.，ＤＢ名)
Ｇ０６Ｆ　２１／００－８８
Ｈ０４Ｌ　９／００－３８