

SYSTEM, METHOD, AND COMPUTER PROGRAM PRODUCT FOR TRACKING FUNDS TRANSFERSCROSS REFERENCE TO RELATED APPLICATION

[0001] This application claims the benefit of U.S. Provisional Patent Application No. 63/591,488, filed on October 19, 2023, the disclosure of which is hereby incorporated by reference in its entirety.

BACKGROUND

1. Technical Field

[0002] This disclosure relates generally to blockchain networks and, in some non-limiting embodiments or aspects, to systems, methods, and computer program products for tracking funds transfers through a blockchain network.

2. Technical Considerations

[0003] Existing systems for tracking funds transfers require a starting point and ending point for each set of funds to detect suspicious activity. This requires a considerable amount of computational resources and lacks transparency for all interested parties. In the banking industry, money trail tracking is a complicated process requiring a detailed investigation to track the starting point and the multiple parties to whom money gets transferred over a period of time.

[0004] According to estimates from the United Nations Office on Drugs and Crime (UNODC), the amount of money laundered annually is between 2% and 5% of global gross domestic product (GDP), which translates to roughly \$800 billion to \$2 trillion. Transfers of money, especially large amounts of money, are often used as a means of facilitating money laundering. Funds transfers can be used at any stage of the money laundering process, but are most commonly used in the layering stage in which the launderer attempts to conceal the origin of the funds by moving the funds through a series of transactions, such as wire transfers, investments, etc. The use of electronic funds transfers, such as wire transfers or online payment systems, has made it easier for money launderers to move funds quickly and across borders, making it more challenging for law enforcement agencies to track and trace illicit funds.

SUMMARY

[0005] Accordingly, provided are improved systems, methods, and computer program products for tracking funds transfers.

[0006] According to non-limiting embodiments or aspects, provided is a system comprising at least one processor of an administrative node in a blockchain network,

the at least one processor programmed or configured to: establish a unified ledger stored by a plurality of nodes of the blockchain network; receive transaction data for a plurality of transactions processed by a plurality of different systems via an interoperable protocol while each transaction of the plurality of transactions is being processed between at least two systems of the plurality of different systems; store, in the unified ledger, the transaction data; and automatically monitor the transaction data in the unified ledger to detect anomalous transactions. In non-limiting embodiments or aspects, the plurality of nodes comprises a plurality of issuer systems in an electronic payment processing network.

[0007] In non-limiting embodiments or aspects, the administrative node comprises a transaction processing system in an electronic payment processing network. In non-limiting embodiments or aspects, each transaction is processed by: creating, with a first node corresponding to a first system, an interledger packet comprising an address of the first node, an address of a second node, a transaction amount, and at least one customer identifier; communicating, from the first node, the interledger packet to a next node via the interoperable protocol until the interledger packet is received by the second node; and communicating, from the second node, the transaction data to the administrative node. In non-limiting embodiments or aspects, the interledger packet is communicated via the interoperable protocol through at least one interledger connector establishing a communicative connection between nodes. In non-limiting embodiments or aspects, the next node is configured to verify the transaction data before communicating the interledger packet to the second node or another next node.

[0008] According to non-limiting embodiments or aspects, provided is a method comprising: establishing, with at least one processor of an administrative node in a blockchain network, a unified ledger stored by a plurality of nodes of the blockchain network; receiving, with the at least one processor of the administrative node, transaction data for a plurality of transactions processed by a plurality of different systems via an interoperable protocol while each transaction of the plurality of transactions is being processed between at least two systems of the plurality of different systems; storing, in the unified ledger, the transaction data; and automatically monitoring, by the at least one processor of the administrative node, the transaction data in the unified ledger to detect anomalous transactions.

[0009] In non-limiting embodiments or aspects, the plurality of nodes comprises a plurality of issuer systems in an electronic payment processing network. In non-limiting

embodiments or aspects, the administrative node comprises a transaction processing system in an electronic payment processing network. In non-limiting embodiments or aspects, each transaction is processed by: creating, with a first node corresponding to a first system, an interledger packet comprising an address of the first node, an address of a second node, a transaction amount, and at least one customer identifier; communicating, from the first node, the interledger packet to a next node via the interoperable protocol until the interledger packet is received by the second node; and communicating, from the second node, the transaction data to the administrative node. In non-limiting embodiments or aspects, the interledger packet is communicated via the interoperable protocol through at least one interledger connector establishing a communicative connection between nodes. In non-limiting embodiments or aspects, the next node is configured to verify the transaction data before communicating the interledger packet to the second node or another next node.

[0010] According to non-limiting embodiments or aspects, provided is a computer program product comprising at least one non-transitory computer-readable medium including program instructions that, when executed by at least one processor of an administrative node in a blockchain network, causes the administrative node to: establish a unified ledger stored by a plurality of nodes of the blockchain network; receive transaction data for a plurality of transactions processed by a plurality of different systems via an interoperable protocol while each transaction of the plurality of transactions is being processed between at least two systems of the plurality of different systems; store, in the unified ledger, the transaction data; and automatically monitor the transaction data in the unified ledger to detect anomalous transactions.

[0011] Further non-limiting embodiments or aspects will be set forth in the following numbered clauses:

[0012] Clause 1: A system comprising at least one processor of an administrative node in a blockchain network, the at least one processor programmed or configured to: establish a unified ledger stored by a plurality of nodes of the blockchain network; receive transaction data for a plurality of transactions processed by a plurality of different systems via an interoperable protocol while each transaction of the plurality of transactions is being processed between at least two systems of the plurality of different systems; store, in the unified ledger, the transaction data; and automatically monitor the transaction data in the unified ledger to detect anomalous transactions.

[0013] Clause 2: The system of clause 1, wherein the plurality of nodes comprises a plurality of issuer systems in an electronic payment processing network.

[0014] Clause 3: The system of clause 1 or 2, wherein the administrative node comprises a transaction processing system in an electronic payment processing network.

[0015] Clause 4: The system of any of clauses 1-3, wherein each transaction is processed by: creating, with a first node corresponding to a first system, an interledger packet comprising an address of the first node, an address of a second node, a transaction amount, and at least one customer identifier; communicating, from the first node, the interledger packet to a next node via the interoperable protocol until the interledger packet is received by the second node; and communicating, from the second node, the transaction data to the administrative node.

[0016] Clause 5: The system of any of clauses 1-4, wherein the interledger packet is communicated via the interoperable protocol through at least one interledger connector establishing a communicative connection between nodes.

[0017] Clause 6: The system of any of clauses 1-5, wherein the next node is configured to verify the transaction data before communicating the interledger packet to the second node or another next node.

[0018] Clause 7: A method comprising: establishing, with at least one processor of an administrative node in a blockchain network, a unified ledger stored by a plurality of nodes of the blockchain network; receiving, with the at least one processor of the administrative node, transaction data for a plurality of transactions processed by a plurality of different systems via an interoperable protocol while each transaction of the plurality of transactions is being processed between at least two systems of the plurality of different systems; storing, in the unified ledger, the transaction data; and automatically monitoring, by the at least one processor of the administrative node, the transaction data in the unified ledger to detect anomalous transactions.

[0019] Clause 8: The method of clause 7, wherein the plurality of nodes comprises a plurality of issuer systems in an electronic payment processing network.

[0020] Clause 9: The method of clause 7 or 8, wherein the administrative node comprises a transaction processing system in an electronic payment processing network.

[0021] Clause 10: The method of any of clauses 7-9, wherein each transaction is processed by: creating, with a first node corresponding to a first system, an interledger

packet comprising an address of the first node, an address of a second node, a transaction amount, and at least one customer identifier; communicating, from the first node, the interledger packet to a next node via the interoperable protocol until the interledger packet is received by the second node; and communicating, from the second node, the transaction data to the administrative node.

[0022] Clause 11: The method of any of clauses 7-10, wherein the interledger packet is communicated via the interoperable protocol through at least one interledger connector establishing a communicative connection between nodes.

[0023] Clause 12: The method of any of clauses 7-11, wherein the next node is configured to verify the transaction data before communicating the interledger packet to the second node or another next node.

[0024] Clause 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 of an administrative node in a blockchain network, causes the administrative node to: establish a unified ledger stored by a plurality of nodes of the blockchain network; receive transaction data for a plurality of transactions processed by a plurality of different systems via an interoperable protocol while each transaction of the plurality of transactions is being processed between at least two systems of the plurality of different systems; store, in the unified ledger, the transaction data; and automatically monitor the transaction data in the unified ledger to detect anomalous transactions.

[0025] Clause 14: The computer program product of clause 13, wherein the plurality of nodes comprises a plurality of issuer systems in an electronic payment processing network.

[0026] Clause 15: The computer program product of any of clauses 13-14, wherein the administrative node comprises a transaction processing system in an electronic payment processing network.

[0027] Clause 16: The computer program product of any of clauses 13-15, wherein each transaction is processed by: creating, with a first node corresponding to a first system, an interledger packet comprising an address of the first node, an address of a second node, a transaction amount, and at least one customer identifier; communicating, from the first node, the interledger packet to a next node via the interoperable protocol until the interledger packet is received by the second node; and communicating, from the second node, the transaction data to the administrative node.

[0028] Clause 17: The computer program product of any of clauses 13-16, wherein the interledger packet is communicated via the interoperable protocol through at least one interledger connector establishing a communicative connection between nodes.

[0029] Clause 18: The computer program product of any of clauses 13-17, wherein the next node is configured to verify the transaction data before communicating the interledger packet to the second node or another next node.

[0030] These and other features and characteristics of the present disclosure, as well as the methods of operation and functions of the related elements of structures and the combination of parts and economies of manufacture, will become more apparent upon consideration of the following description and the appended claims with reference to the accompanying drawings, all of which form a part of this specification, wherein like reference numerals designate corresponding parts in the various figures. It is to be expressly understood, however, that the drawings are for the purpose of illustration and description only and are not intended as a definition of the limits of the disclosed subject matter.

BRIEF DESCRIPTION OF THE DRAWINGS

[0031] Additional advantages and details are explained in greater detail below with reference to the non-limiting, exemplary embodiments that are illustrated in the accompanying schematic figures, in which:

[0032] FIG. 1 is a schematic diagram of a system for tracking funds transfers according to some non-limiting embodiments or aspects;

[0033] FIG. 2A is a flow diagram of a method for tracking funds transfers according to some non-limiting embodiments or aspects;

[0034] FIG. 2B is a flow diagram of a method for tracking funds transfers according to some non-limiting embodiments or aspects;

[0035] FIG. 3 is a schematic diagram of a system for tracking funds transfers according to some non-limiting embodiments or aspects;

[0036] FIG. 4 is a schematic diagram of example components of one or more devices according to some non-limiting embodiments or aspects; and

[0037] FIG 5 is a schematic diagram of an electronic payment processing network according to non-limiting embodiments or aspects.

DETAILED DESCRIPTION

[0038] For purposes of the description hereinafter, the terms “end,” “upper,” “lower,” “right,” “left,” “vertical,” “horizontal,” “top,” “bottom,” “lateral,” “longitudinal,” and

derivatives thereof shall relate to the embodiments as they are oriented in the drawing figures. However, it is to be understood that the present disclosure may assume various alternative variations and step sequences, except where expressly specified to the contrary. It is also to be understood that the specific devices and processes illustrated in the attached drawings, and described in the following specification, are simply exemplary and non-limiting embodiments or aspects of the disclosed subject matter. Hence, specific dimensions and other physical characteristics related to the embodiments or aspects disclosed herein are not to be considered as limiting.

[0039] Some non-limiting embodiments or aspects are described herein in connection with thresholds. As used herein, satisfying a threshold may refer to a value being greater than the threshold, more than the threshold, higher than the threshold, greater than or equal to the threshold, less than the threshold, fewer than the threshold, lower than the threshold, less than or equal to the threshold, equal to the threshold, etc.

[0040] No aspect, component, element, structure, act, step, function, instruction, and/or the like used herein should be construed as critical or essential unless explicitly described as such. Also, as used herein, the articles “a” and “an” are intended to include one or more items and may be used interchangeably with “one or more” and “at least one.” Furthermore, as used herein, the term “set” is intended to include one or more items (e.g., related items, unrelated items, a combination of related and unrelated items, and/or the like) and may be used interchangeably with “one or more” or “at least one.” Where only one item is intended, the term “one” or similar language is used. Also, as used herein, the terms “has,” “have,” “having,” or the like are intended to be open-ended terms. Further, the phrase “based on” is intended to mean “based at least partially on” unless explicitly stated otherwise. In addition, reference to an action being “based on” a condition may refer to the action being “in response to” the condition. For example, the phrases “based on” and “in response to” may, in some non-limiting embodiments or aspects, refer to a condition for automatically triggering an action (e.g., a specific operation of an electronic device, such as a computing device, a processor, and/or the like).

[0041] As used herein, the term “communication” may refer to the reception, receipt, transmission, transfer, provision, and/or the like of data (e.g., information, signals, messages, instructions, commands, and/or the like). For one unit (e.g., a device, a system, a component of a device or system, combinations thereof, and/or

the like) to be in communication with another unit means that the one unit is able to directly or indirectly receive information from and/or transmit information to the other unit. This may refer to a direct or indirect connection (e.g., a direct communication connection, an indirect communication connection, and/or the like) that is wired and/or wireless in nature. Additionally, two units may be in communication with each other even though the information transmitted may be modified, processed, relayed, and/or routed between the first and second unit. For example, a first unit may be in communication with a second unit even though the first unit passively receives information and does not actively transmit information to the second unit. As another example, a first unit may be in communication with a second unit if at least one intermediary unit processes information received from the first unit and communicates the processed information to the second unit. In some non-limiting embodiments or aspects, a message may refer to a network packet (e.g., a data packet and/or the like) that includes data. It will be appreciated that numerous other arrangements are possible.

[0042] As used herein, the term “computing device” may refer to one or more electronic devices configured to process data. A computing device may, in some examples, include the necessary components to receive, process, and output data, such as a processor, a display, a memory, an input device, a network interface, and/or the like. A computing device may be a mobile device. As an example, a mobile device may include a cellular phone (e.g., a smartphone or standard cellular phone), a portable computer, a wearable device (e.g., watches, glasses, lenses, clothing, and/or the like), a personal digital assistant (PDA), and/or other like devices. A computing device may also be a desktop computer or other form of non-mobile computer.

[0043] As used herein, the term “server” may refer to or include one or more computing devices that are operated by or facilitate communication and processing for multiple parties in a network environment, such as the Internet, although it will be appreciated that communication may be facilitated over one or more public or private network environments and that various other arrangements are possible. Further, multiple computing devices (e.g., servers, point-of-sale (POS) devices, mobile devices, etc.) directly or indirectly communicating in the network environment may constitute a “system.”

[0044] As used herein, the term “system” may refer to one or more computing devices or combinations of computing devices (e.g., processors, servers, client

devices, software applications, components of such, and/or the like). Reference to “a device,” “a server,” “a processor,” and/or the like, as used herein, may refer to a previously-recited device, server, or processor that is recited as performing a previous step or function, a different device, server, or processor, and/or a combination of devices, servers, and/or processors. For example, as used in the specification and the claims, a first device, a first server, or a first processor that is recited as performing a first step or a first function may refer to the same or different device, server, or processor recited as performing a second step or a second function.

[0045] An “application program interface” (API) refers to computer code or other data stored on a computer-readable medium that may be executed by a processor to facilitate the interaction between software components, such as a client-side front-end and/or server-side back-end for receiving data from the client. An “interface” refers to a generated display, such as one or more graphical user interfaces (GUIs) with which a user may interact, either directly or indirectly (e.g., through a keyboard, mouse, etc.).

[0046] As used herein, the term “issuer institution” may refer to one or more entities, such as a bank, that provide accounts to customers for conducting transactions (e.g., payment transactions), such as initiating credit and/or debit payments. For example, an issuer institution may provide an account identifier, such as a primary account number (PAN), to a customer that uniquely identifies one or more accounts associated with that customer. The account identifier may be embodied on a portable financial device, such as a physical financial instrument, e.g., a payment card, and/or may be electronic and used for electronic payments. The term “issuer system” refers to one or more computer devices operated by or on behalf of an issuer institution, such as a server computer executing one or more software applications. For example, an issuer system may include one or more authorization servers for authorizing a transaction.

[0047] As used herein, the term “transaction service provider” may refer to an entity that receives transaction authorization requests from merchants or other entities and provides guarantees of payment, in some cases through an agreement between the transaction service provider and an issuer institution. For example, a transaction service provider may include a payment network such as Visa® or any other entity that processes transactions. The term “transaction processing system” may refer to one or more computer systems operated by or on behalf of a transaction service provider, such as a transaction processing server executing one or more software applications. A transaction processing server may include one or more processors

and, in some non-limiting embodiments or aspects, may be operated by or on behalf of a transaction service provider.

[0048] Non-limiting embodiments described herein allow for funds transfers to be tracked to, for example, detect money laundering and/or other occurrences. Non-limiting embodiments include a blockchain-based architecture and an interoperable protocol to enable recordation of funds transfers on a shared (e.g., distributed) ledger. By connecting multiple entities, such as issuer institutions, through an interoperable protocol, funds transfers between two issuer institutions can be tracked by other issuer institutions, regulators, third parties, and/or others that are not a party to that particular funds transfer. Through implementations of the methods and systems described herein, permissioned access and/or automated monitoring may be performed. This improves upon existing funds transfers tracking methods that require a starting point and ending point to track each funds transfer for suspicious activity. Non-limiting embodiments reduce the use of computational resources necessary to track funds transfers through different systems, protocols, and data structures. Non-limiting embodiments provide other improvements as described herein and as will be appreciated by those skilled in the art.

[0049] Referring to FIG. 1, a system 1000 for tracking funds transfers is shown according to some non-limiting embodiments or aspects. The system 1000 may include several computing devices 102, 104, 106, 108. In non-limiting embodiments, each of the computing devices 102, 104, 106, 108 operate as nodes in a blockchain network. However it will be appreciated that not all of the computing devices may be in direct communication with a blockchain network. It will also be appreciated that any number of nodes may be used in non-limiting embodiments and that the four (4) nodes shown in FIG. 1 are for purposes of illustration only. In non-limiting implementations there may be several, hundreds, and/or thousands of nodes in the blockchain network.

[0050] In non-limiting embodiments, the first node 102, second node 104, and third node 106 may each be issuer systems associated with issuer institutions. In non-limiting embodiments, first node 102, second node 104, and third node 106 may each be a separate payment system. There may be any number of nodes. The nodes 102, 104, 106 may be in communication through an interoperable protocol 100, which may include one or more APIs with standardized parameters. For example, the interoperable protocol may include the Interledger Protocol (ILP). The interoperable protocol 100 may be utilized by hosts (e.g., nodes) in an interledger environment

involving protocol modules that call on local ledger protocols to carry a packet 114 of data to the next connector (e.g., the next node) until it reaches the recipient (e.g., destination) system or account. Each node 102, 104, 106 may be associated with a respective ledger 103, 105, 107 that stores data (e.g., transaction data) for that node. The interoperable protocol 100 may be any communication protocol that allows for the real-time (e.g., during a payment transaction, with minimal or imperceptible delay, and/or the like) transfer of funds and settlement across different ledgers and/or different payment networks. In non-limiting embodiments, the interoperable protocol may provide for atomic transactions, where the transfer is either fully completed or fully cancelled, thereby eliminating the risk of intermediary failure or fraud.

[0051] The interoperable protocol 100 is depicted in FIG. 1 as a box connected to each of the nodes 102, 104, 106 for purposes of illustration. It will be appreciated that the interoperable protocol 100 may not be an individual system, but rather a set of software functions and/or network protocols that enable the nodes 102, 104, 106 to communicate among themselves without using a central system or intermediary. For example, the interoperable protocol 100 may include one or more APIs for connecting different ledgers of different banks associated with the nodes 102, 104, 106, one or more protocols for how to generate, route, and/or reassemble data packets, one or more software functions for generating, routing, and/or reassembling data packets, and/or the like. The interoperable protocol 100 may also include connectors 111 to communicatively interface with and between the nodes 102, 104, 106, 108. The interoperable protocol may, in non-limiting embodiments, use one or more communication protocols, such as HTTP, HTTPS, WebSocket, and/or the like, for communicating among nodes. The interoperable protocol may, in non-limiting embodiments, be used with a public blockchain network, private blockchain network, electronic payment processing network, and/or the like to provide data communication that is agnostic to a specific ledger (e.g., for a specific bank or financial institution).

[0052] As used herein, a “connector” for an interoperable protocol may refer to software executing on a computing device, such as a node 102, 104, 106, configured to route, exchange, and/or transfer units of value between different ledgers (e.g., different issuers, different payment networks, and/or the like). A connector may process packets in accordance with rules of the interoperable protocol 100. In non-limiting embodiments, a connector may act as middleware between nodes to facilitate packet transfers.

[0053] With continued reference to FIG. 1, when one system (e.g., first node 102) makes a funds transfer to another system (e.g., third node 106), the first node 102 may create (e.g., generate) an interledger packet 114. The interledger packet 114 may include one or more data structures including funds transfer data, such as but not limited to an address of the first node (e.g., a starting network address, a starting blockchain address, and/or the like), an address of the third node (e.g., a destination network address, a destination blockchain address, and/or the like), a transfer amount (e.g., an amount of funds or units), and/or at least one customer identifier (e.g., a customer account identifier, a user identifier, and/or the like that identifies a customer and/or account sending or receiving funds through a respective issuer system). The packet may include a header and various fields. The packet 114 is communicated via the interoperable protocol 100 such that it first is communicated from the first node 102 to the second node 104 (e.g., the next connecting node) and then from the second node 104 to the third node 106 (or a different destination node). Each connecting node between the starting node and the destination node (e.g., the second node 104 in the above example) may verify the funds transfer data (e.g., transaction data) before transferring the packet to the next connecting node. The nodes may verify the funds through respective ledgers 103, 105, 107 local to each node, as an example.

[0054] The interoperable protocol 100 and connectors 111 of the interoperable protocol 100 may provide direct network connections (e.g., without a central server) between the nodes 102, 104, 106. When the destination node (e.g., third node 106) receives the interledger packet 114, the destination node may then communicate to an administrative node 108 in the blockchain network. This communication may occur via the interoperable protocol 100 or may occur outside the interoperable protocol 100 (e.g., through the blockchain network without regard to an interoperable protocol 100). In non-limiting embodiments, each node may standardize transaction data to be compatible with a ledger 110. For example, the node 102 may standardize the transaction data in the ledger 103 according to the interoperable protocol 100 for communication in the blockchain network and storage in the ledger 110.

[0055] With continued reference to FIG. 1, the administrative node 108 may include or be in communication with a ledger 110. The ledger 110 may be stored on one or more data storage devices in local or remote communication with the administrative node 108. In non-limiting embodiments, the administrative node 108 may be a computing device operated by a transaction processing system, payment gateway,

issuer system, third-party system, and/or the like. There may be a plurality of administrative nodes 108 in some non-limiting embodiments. In non-limiting embodiments, the ledger 110 may be a distributed ledger that is hosted by a plurality of nodes of the blockchain network (e.g., nodes 102, 104, 106, 108) such that each node can view and/or audit the ledger 110.

[0056] In non-limiting embodiments, the ledger 110 may be a shared, unified ledger that includes transaction data for funds transfers between participating banks and/or issuers. For example, the ledger 110 may include Know-Your-Customer (KYC) data to trace entities or parties to a transaction, account information for an origin account (e.g., bank information including location), account information for a destination account (e.g., receiving bank information including location), transaction details (e.g., value, currency, date, etc.), and/or the like. In non-limiting embodiments, the ledger 110 may be made accessible to authorized entities through the blockchain network, such as but not limited to one or more governmental entities, authentication services, investigators, regulators, and/or the like.

[0057] Still referring to FIG. 1, the administrative node 108 and/or other system (e.g., a transaction processing system, a third party, nodes 102, 104, 106, and/or the like) may continually, continuously, and/or periodically monitor the ledger 110 to detect anomalous transactions. A transaction may be determined to be anomalous based on one or more transaction parameters satisfying a threshold, based on a machine-learning model, based on one or more rules, and/or the like. In non-limiting embodiments, the ledger 110 may be monitored by identifying features and classifying each funds transfer (e.g., transaction) with a model. For example, one or more clustering algorithms may be used to cluster discrete transactions into clusters of transactions for identifying patterns, outliers, and anomalous transactions that are more than a threshold distance from any cluster. The transactions may be converted into features (e.g., such as feature vectors) based on the transaction parameters for each transaction. The distance between features may be determined as a Euclidean distance between feature vectors, as an example. In non-limiting embodiments, the magnitude of a transaction may be used to classify a transaction. In some non-limiting embodiments, one or more alerts may be generated by the administrative node 108 in response to detecting an anomalous transaction, in response to detecting a suspicious pattern of activity, in response to a transaction satisfying one or more thresholds,

and/or the like. Alerts may be communicated to regulators, third parties, customers, merchants, and/or the like.

[0058] In non-limiting embodiments, end-to-end encryption may be implemented to secure the funds transfer data. The end-to-end encryption may begin at the starting node (e.g., first node 102) for a funds transfer and end at the ledger 110, where the funds transfer information may be stored in encrypted form or decrypted form. For example, the interoperable protocol 100 may include one or more rules for encryption that are used by the sending node (e.g., first node 102) to encrypt the transaction data in the interledger packet 114 and/or the interledger packet 114 itself. The interoperable protocol 100 may also include one or more rules for decryption that are used by the receiving node (e.g., third node 106) and/or administrative node 108 to decrypt the transaction data in the interledger packet 114 and/or the interledger packet 114 to store the transaction data in the ledger 110.

[0059] In non-limiting embodiments, a compliance monitoring system (not shown in FIG. 1) may monitor the transaction data stored in the ledger 110, generate compliance reports, identify suspicious transactions, and/or the like. In some examples, a compliance monitoring system may be node 108 or another node in the blockchain network. In non-limiting embodiments, an access authorization system (not shown in FIG. 1) may grant permissioned access to one or more other entities (e.g., such as government or regulatory authorities). In some examples, an access authorization system may be node 108 or another node in the blockchain network.

[0060] The number and arrangement of systems and devices shown in FIG. 1 are provided as an example. There may be additional systems and/or devices, fewer systems and/or devices, different systems and/or devices, and/or differently arranged systems and/or devices than those shown in FIG. 1.

[0061] Referring now to FIG. 2A, shown is a flow diagram for a method for tracking funds transfers according to some non-limiting embodiments or aspects. The steps shown in FIG. 2A are for example purposes only. It will be appreciated that additional, fewer, different, and/or a different order of steps may be used in some non-limiting embodiments or aspects. In some non-limiting embodiments or aspects, a step may be automatically performed in response to performance and/or completion of a prior step. The steps shown in FIG. 2A may be performed to detect anomalous payment transactions, such as fraudulent transactions, and may be performed by one or more nodes (e.g., nodes 102, 104, 106, 108 in FIG. 1) and/or any other computing system.

[0062] At step 201, a sending node (e.g., such as node 102 in FIG. 1) may generate a packet (e.g., an interledger packet) including the sending node address (e.g., a network address, a blockchain address, and/or the like), the destination (receiving) node address, a transaction value (e.g., an amount to be transferred), a sending customer and/or account identifier (e.g., an entity or account making a payment), a receiving customer and/or account identifier (e.g., an entity or account receiving a payment), and/or other transaction data. The packet may be generated at step 201 in accordance with an interoperable protocol, such as but not limited to the Interledger Protocol (e.g., ILPv4). In non-limiting embodiments, the sending node may modify and/or process the transaction data to be in a standardized format of the interoperable protocol, and such a format may differ from a data format used by the node to store locally. For example, with reference to the example of FIG. 1, the format used by the node 102 to store in the ledger 103 may be different than the format used for the packet 114.

[0063] At step 203 the packet generated at step 201 is communicated to a next node in the blockchain network. For example, the nodes may be ordered and/or sequential such that the destination node is one or more hops away from the sending node. It will be appreciated that a next node may be determined in any number of ways, and may be predetermined or dynamically determined in non-limiting embodiments. At step 205 it is determined whether the packet has been received by the destination node (e.g., by comparing the node address to the destination node address in the packet). If the destination node has not yet been reached, the method continues sending the packet on to the next node. Once the destination node is reached at step 205, the method may proceed to step 207 and the transaction may be settled on a ledger of the destination node. For example, in the non-limiting embodiment shown in FIG. 1, if node 106 is the destination node, the transaction data may be stored on ledger 107.

[0064] At step 209 of FIG. 2A, the packet and/or transaction data from the packet may be stored in a unified shared ledger accessible to all of the nodes in the blockchain network. It will be appreciated that the transaction data on the shared ledger may be stored as it is received by the destination node and/or may be stored after being processed. For example, the transaction data may be processed to identify one or more features to represent the transaction data, and the features may be stored in the shared ledger with or without the original transaction data. At step 211, the transaction

data may be monitored by one or more entities (e.g., other nodes, government entities, regulators, investigators, and/or the like). The monitoring may be performed automatically on a continual, continuous, and/or periodic basis. The transaction data may be monitored to identify anomalies with funds transfers, to track suspicious transfers, to determine compliance and/or generate compliance reports, and/or the like.

[0065] Referring now to FIG. 2B, shown is a flow diagram for a method for tracking funds transfers according to some non-limiting embodiments or aspects. The steps shown in FIG. 2B are for example purposes only. It will be appreciated that additional, fewer, different, and/or a different order of steps may be used in some non-limiting embodiments or aspects. In some non-limiting embodiments or aspects, a step may be automatically performed in response to performance and/or completion of a prior step. The steps shown in FIG. 2B may be performed to detect anomalous payment transactions, such as fraudulent transactions, and may be performed by a node (e.g., nodes 102, 104, 106, 108 in FIG. 1) and/or any other computing system. Transaction data stored on a unified shared ledger 202 on a blockchain network may represent transactions between different banks, issuers, and/or payment systems that have been made through an interoperable protocol. The unified shared ledger 202 may, in some non-limiting embodiments, be the ledger 110 of FIG. 1. The unified shared ledger 202 may be continuously, continually, and/or periodically updated with transaction data from multiple different issuers and/or banks.

[0066] At step 204, the transaction data in the unified shared ledger 202 may be processed to generate one or more features (e.g., such as feature vectors). Features may be generated by identifying parameters of the transaction data and generating a numerical representation and/or data structure of those parameters. The features may represent transaction values (e.g., amounts), transaction times (e.g., timestamps), transaction locations (e.g., geographic information), customer profiles (e.g., flagged, suspended, etc.), and/or the like.

[0067] At step 206, one or more clustering algorithms may be used to process the features generated at step 204. For example a k-means clustering algorithm may be used to cluster transactions such that similar transactions may be clustered together and that some transactions may be outliers. At step 210, one or more pattern detection algorithms and/or models may be applied to the features. For example, transactions in one or more clusters or transactions that are outliers to clusters may be processed

according to a pattern detection algorithm. The pattern detection algorithm may be based on transaction magnitude (e.g., large sums of money transfers satisfying a predetermined threshold), transfers (e.g., transfers to predetermined tax havens or other locations), round-trip transactions (e.g., transfers that match incoming and outgoing), transfers at predetermined locations (e.g., high-risk jurisdictions), transfers at predetermined times or circumstances (e.g., predetermined situations or events), shell company transfers, unusual activity, and/or the like. It will be appreciated that various parameters may be used to determine one or more patterns based on one or more models, sets of rules, and/or the like.

[0068] At step 208, one or more outlier detection algorithms and/or models may be applied to the features from step 204. For example, a z-score statistical method may be applied to identify outliers in each feature and/or combination of features for a transaction or multiple transactions. At step 212, the outliers (e.g., outlying features) may be processed to determine if there are any anomalous transactions. For example, one or more models may be applied to transactions having outlying features to determine if those transactions are anomalous. At step 214, the patterns detected at step 210 and the anomalies detected at step 212 may be streamed to an application to be viewed and/or used to perform one or more additional steps. For example, a distributed streaming platform such as Kafka may be used to stream analytic data from steps 210 and 212 to generate an alert at step 216. The alert may be communicated to a bank, issuer system, account holder, merchant, transaction processing system, third party (e.g., such as a government entity), and/or the like. It will be appreciated that other automated actions may be alternatively or additionally performed, such as displaying a graphical user interface, flagging transactions in a database, and/or the like.

[0069] FIG. 3 illustrates data in a system for tracking funds transfers according to some non-limiting embodiments. FIG. 3 shows how funds transfers can be tracked for a loan after the loan amount is sanctioned. For example, a loan may be sanctioned for the purchase of flights, but the money may be transferred elsewhere for a different intent. In non-limiting embodiments, smart contracts may be defined to monitor various scenarios to track the trail and generate alerts. FIG. 3 shows interledger packets through various transfers. An initial interledger packet 302 includes transaction data for a loan. The Tracker_ID parameter may be an identifier of the node that created and/or received the packet 302. For example, the node that generates packet 302 may

be a bank or issuer system that is making loans to other entities. Funds are transferred, using an interledger packet, from that node to additional nodes as part of the loan.

[0070] With continued reference to FIG. 3, packets 304, 306, 310 each are associated with respective nodes that receive funds downstream of the original loan. There may be logic applied for each subsequent transfer that includes, for example, a time parameter from the original loan release date or transfer. If a transfer qualifies (e.g., is within a time period or other parameter), the packet may be modified to populate the Tracker_ID-Origin parameter field with an identifier of the original node that initiated the loan. For example, packet 304 may include a Tracker_ID-Origin parameter that is populated with an identifier of the original node that initiated the loan and generated packet 302 (i.e., where the trail of the loan amount began). When the node that generates packet 304 after receiving a transfer from the loan originating node makes a subsequent transfer, the node receiving that subsequent transfer may also generate a packet 308 with modified transaction data to indicate where the packet and funds were received from (e.g., the node associated with packet 304), the originating node (e.g., the node associated with packet 302), and other parameters. Each packet 302, 304, 306, 308, 310, 312, 314 may be stored on the blockchain ledger, such as in a unified shared ledger.

[0071] Non-limiting embodiments of a system for tracking funds transfers may be implemented with the following pseudocode written for an Interledger Protocol implementation:

1. Define the Interledger Connectors:

```
InterledgerConnectors {
    ConnectorList
    ForwardPacket(Packet) {
        // Forwards packet to the next connector in the chain
    }
}
```

2. Define the Shared Ledger:

```
SharedLedger {
    TransactionList
    RecordTransaction(Packet) {
        // Records transaction details on the shared ledger
    }
}
```

```
}
```

```
}
```

3. Define the Compliance Reporting and Monitoring:

```
ComplianceReporting {  
    GenerateReports() {  
        // Generates compliance reports for each bank  
    }  
    AutomatedMonitoring() {  
        // Implements automated monitoring systems to detect suspicious  
        transactions  
    }  
}
```

4. Define the Authorized Access:

```
AuthorizedAccess {  
    GrantAccess(Authority) {  
        // Grants permissioned access to the shared ledger for tracking  
        and monitoring funds movement  
    }  
}
```

5. Define the Transaction:

```
Transaction {  
    Packet  
    SenderLedger  
    ReceiverLedger  
    Settlement() {  
        // Settles the transaction on the receiver's ledger  
    }  
}
```

6. Define the Bank:

```

Bank {
    Ledger
    SendTransaction(Transaction) {
        // Sends the transaction to the next connector in the chain
    }
}

```

[0072] Referring now to FIG. 4, shown is a diagram of example components of a device 400 according to non-limiting embodiments or aspects. Device 400 may correspond to at least one of the computing devices (e.g., nodes 102, 104, 106, 108) in FIG. 1. In some non-limiting embodiments or aspects, such systems or devices may include at least one device 400 and/or at least one component of device 400. The number and arrangement of components shown in FIG. 4 are provided as an example. In some non-limiting embodiments or aspects, device 400 may include additional components, fewer components, different components, or differently arranged components than those shown in FIG. 4. Additionally, or alternatively, a set of components (e.g., one or more components) of device 400 may perform one or more functions described as being performed by another set of components of device 400.

[0073] As shown in FIG. 4, device 400 may include bus 402, processor 404, memory 406, storage component 408, input component 410, output component 412, and communication interface 414. Bus 402 may include a component that permits communication among the components of device 400. In some non-limiting embodiments or aspects, processor 404 may be implemented in hardware, firmware, or a combination of hardware and software. For example, processor 404 may include a processor (e.g., a central processing unit (CPU), a graphics processing unit (GPU), an accelerated processing unit (APU), etc.), a microprocessor, a digital signal processor (DSP), and/or any processing component (e.g., a field-programmable gate array (FPGA), an application-specific integrated circuit (ASIC), etc.) that can be programmed to perform a function. Memory 406 may include random access memory (RAM), read only memory (ROM), and/or another type of dynamic or static storage device (e.g., flash memory, magnetic memory, optical memory, etc.) that stores information and/or instructions for use by processor 404.

[0074] With continued reference to FIG. 4, storage component 408 may store information and/or software related to the operation and use of device 400. For example, storage component 408 may include a hard disk (e.g., a magnetic disk, an optical disk, a magneto-optic disk, a solid state disk, etc.) and/or another type of computer-readable medium. Input component 410 may include a component that permits device 400 to receive information, such as via user input (e.g., a touch screen display, a keyboard, a keypad, a mouse, a button, a switch, a microphone, etc.). Additionally, or alternatively, input component 410 may include a sensor for sensing information (e.g., a global positioning system (GPS) component, an accelerometer, a gyroscope, an actuator, etc.). Output component 412 may include a component that provides output information from device 400 (e.g., a display, a speaker, one or more light-emitting diodes (LEDs), etc.). Communication interface 414 may include a transceiver-like component (e.g., a transceiver, a separate receiver and transmitter, etc.) that enables device 400 to communicate with other devices, such as via a wired connection, a wireless connection, or a combination of wired and wireless connections. Communication interface 414 may permit device 400 to receive information from another device and/or provide information to another device. For example, communication interface 414 may include an Ethernet interface, an optical interface, a coaxial interface, an infrared interface, a radio frequency (RF) interface, a universal serial bus (USB) interface, a Wi-Fi® interface, a cellular network interface, and/or the like.

[0075] Device 400 may perform one or more processes described herein. Device 400 may perform these processes based on processor 404 executing software instructions stored by a computer-readable medium, such as memory 406 and/or storage component 408. A computer-readable medium may include any non-transitory memory device. A memory device includes memory space located inside of a single physical storage device or memory space spread across multiple physical storage devices. Software instructions may be read into memory 406 and/or storage component 408 from another computer-readable medium or from another device via communication interface 414. When executed, software instructions stored in memory 406 and/or storage component 408 may cause processor 404 to perform one or more processes described herein. Additionally, or alternatively, hardwired circuitry may be used in place of or in combination with software instructions to perform one or more processes described herein. Thus, embodiments described herein are not limited to

any specific combination of hardware circuitry and software. The term “configured to,” as used herein, may refer to an arrangement of software, device(s), and/or hardware for performing and/or enabling one or more functions (e.g., actions, processes, steps of a process, and/or the like). For example, “a processor configured to” may refer to a processor that executes software instructions (e.g., program code) that cause the processor to perform one or more functions.

[0076] FIG. 5 shows an electronic payment processing network 1100 according to non-limiting embodiments or aspects. The payment processing network may be used in conjunction with the systems and methods described herein. It will be appreciated that the particular arrangement of the electronic payment processing network 1100 shown is for example purposes only, and that various arrangements are possible. The transaction processing system 1101 (e.g., a transaction handler) is shown to be in communication with one or more issuer systems (e.g., such as an issuer system 1106) and one or more acquirer systems (e.g., such as an acquirer system 1108). Although only a single issuer system 1106 and single acquirer system 1108 are shown, it will be appreciated that the transaction processing system 1101 may be in communication with a plurality of issuer systems and/or acquirer systems. In some embodiments, the transaction processing system 1101 may also operate as an issuer system such that both the transaction processing system 1101 and issuer system 1106 are a single system and/or controlled by a single entity.

[0077] In some non-limiting embodiments or aspects, the transaction processing system 1101 may communicate with a merchant system 1104 directly through a public or private network connection. Additionally or alternatively, the transaction processing system 1101 may communicate with the merchant system 1104 through the payment gateway 1102 and/or acquirer system 1108. In some non-limiting embodiments or aspects, an acquirer system 1108 associated with the merchant system 1104 may operate as a payment gateway 1102 to facilitate the communication of transaction requests from the merchant system 1104 to the transaction processing system 1101. The merchant system 1104 may communicate with the payment gateway 1102 through a public or private network connection. For example, a merchant system 1104 that includes a physical POS device may communicate with the payment gateway 1102 through a public or private network to conduct card-present transactions. As another example, a merchant system 1104 that includes a server (e.g., a web server)

may communicate with the payment gateway 1102 through a public or private network, such as a public Internet connection, to conduct card-not-present transactions.

[0078] In some non-limiting embodiments or aspects, the transaction processing system 1101, after receiving a transaction request from the merchant system 1104 that identifies an account identifier of a payor (e.g., such as an account holder) associated with an issued payment device 1110, may generate an authorization request message to be communicated to the issuer system 1106 that issued the payment device 1110 and/or account identifier. The issuer system 1106 may then approve or decline the authorization request and, based on the approval or denial, generate an authorization response message that is communicated to the transaction processing system 1101. The transaction processing system 1101 may communicate an approval or denial to the merchant system 1104. When the issuer system 1106 approves the authorization request message, it may then clear and settle the payment transaction between the issuer system 1106 and acquirer system 1108.

[0079] Although embodiments have been described in detail for the purpose of illustration, it is to be understood that such detail is solely for that purpose and that the disclosure is not limited to the disclosed embodiments or aspects, but, on the contrary, is intended to cover modifications and equivalent arrangements that are within the spirit and scope of the appended claims. For example, it is to be understood that the present disclosure contemplates that, to the extent possible, one or more features of any embodiment or aspect can be combined with one or more features of any other embodiment or aspect.

WHAT IS CLAIMED IS:

1. A system comprising at least one processor of an administrative node in a blockchain network, the at least one processor configured to:
 - establish a unified ledger stored by a plurality of nodes of the blockchain network;
 - receive transaction data for a plurality of transactions processed by a plurality of different systems via an interoperable protocol while each transaction of the plurality of transactions is being processed between at least two systems of the plurality of different systems;
 - store, in the unified ledger, the transaction data; and
 - automatically monitor the transaction data in the unified ledger to detect anomalous transactions.
2. The system of claim 1, wherein the plurality of nodes comprises a plurality of issuer systems in an electronic payment processing network.
3. The system of claim 1, wherein the administrative node comprises a transaction processing system in an electronic payment processing network.
4. The system of claim 1, wherein each transaction is processed by:
 - creating, with a first node corresponding to a first system, an interledger packet comprising an address of the first node, an address of a second node, a transaction amount, and at least one customer identifier;
 - communicating, from the first node, the interledger packet to a next node via the interoperable protocol until the interledger packet is received by the second node; and
 - communicating, from the second node, the transaction data to the administrative node.
5. The system of claim 4, wherein the interledger packet is communicated via the interoperable protocol through at least one interledger connector establishing a communicative connection between nodes.

6. The system of claim 5, wherein the next node is configured to verify the transaction data before communicating the interledger packet to the second node or another next node.

7. A method comprising:
establishing, with at least one processor of an administrative node in a blockchain network, a unified ledger stored by a plurality of nodes of the blockchain network;

receiving, with the at least one processor of the administrative node, transaction data for a plurality of transactions processed by a plurality of different systems via an interoperable protocol while each transaction of the plurality of transactions is being processed between at least two systems of the plurality of different systems;

storing, in the unified ledger, the transaction data; and

automatically monitoring, by the at least one processor of the administrative node, the transaction data in the unified ledger to detect anomalous transactions.

8. The method of claim 7, wherein the plurality of nodes comprises a plurality of issuer systems in an electronic payment processing network.

9. The method of claim 7, wherein the administrative node comprises a transaction processing system in an electronic payment processing network.

10. The method of claim 7, wherein each transaction is processed by:
creating, with a first node corresponding to a first system, an interledger packet comprising an address of the first node, an address of a second node, a transaction amount, and at least one customer identifier;

communicating, from the first node, the interledger packet to a next node via the interoperable protocol until the interledger packet is received by the second node; and

communicating, from the second node, the transaction data to the administrative node.

11. The method of claim 10, wherein the interledger packet is communicated via the interoperable protocol through at least one interledger connector establishing a communicative connection between nodes.

12. The method of claim 11, wherein the next node is configured to verify the transaction data before communicating the interledger packet to the second node or another next node.

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 of an administrative node in a blockchain network, causes the administrative node to:

establish a unified ledger stored by a plurality of nodes of the blockchain network;

receive transaction data for a plurality of transactions processed by a plurality of different systems via an interoperable protocol while each transaction of the plurality of transactions is being processed between at least two systems of the plurality of different systems;

store, in the unified ledger, the transaction data; and

automatically monitor the transaction data in the unified ledger to detect anomalous transactions.

14. The computer program product of claim 13, wherein the plurality of nodes comprises a plurality of issuer systems in an electronic payment processing network.

15. The computer program product of claim 13, wherein the administrative node comprises a transaction processing system in an electronic payment processing network.

16. The computer program product of claim 13, wherein each transaction is processed by:

creating, with a first node corresponding to a first system, an interledger packet comprising an address of the first node, an address of a second node, a transaction amount, and at least one customer identifier;

communicating, from the first node, the interledger packet to a next node via the interoperable protocol until the interledger packet is received by the second node; and

communicating, from the second node, the transaction data to the administrative node.

17. The computer program product of claim 16, wherein the interledger packet is communicated via the interoperable protocol through at least one interledger connector establishing a communicative connection between nodes.

18. The computer program product of claim 17, wherein the next node is configured to verify the transaction data before communicating the interledger packet to the second node or another next node.