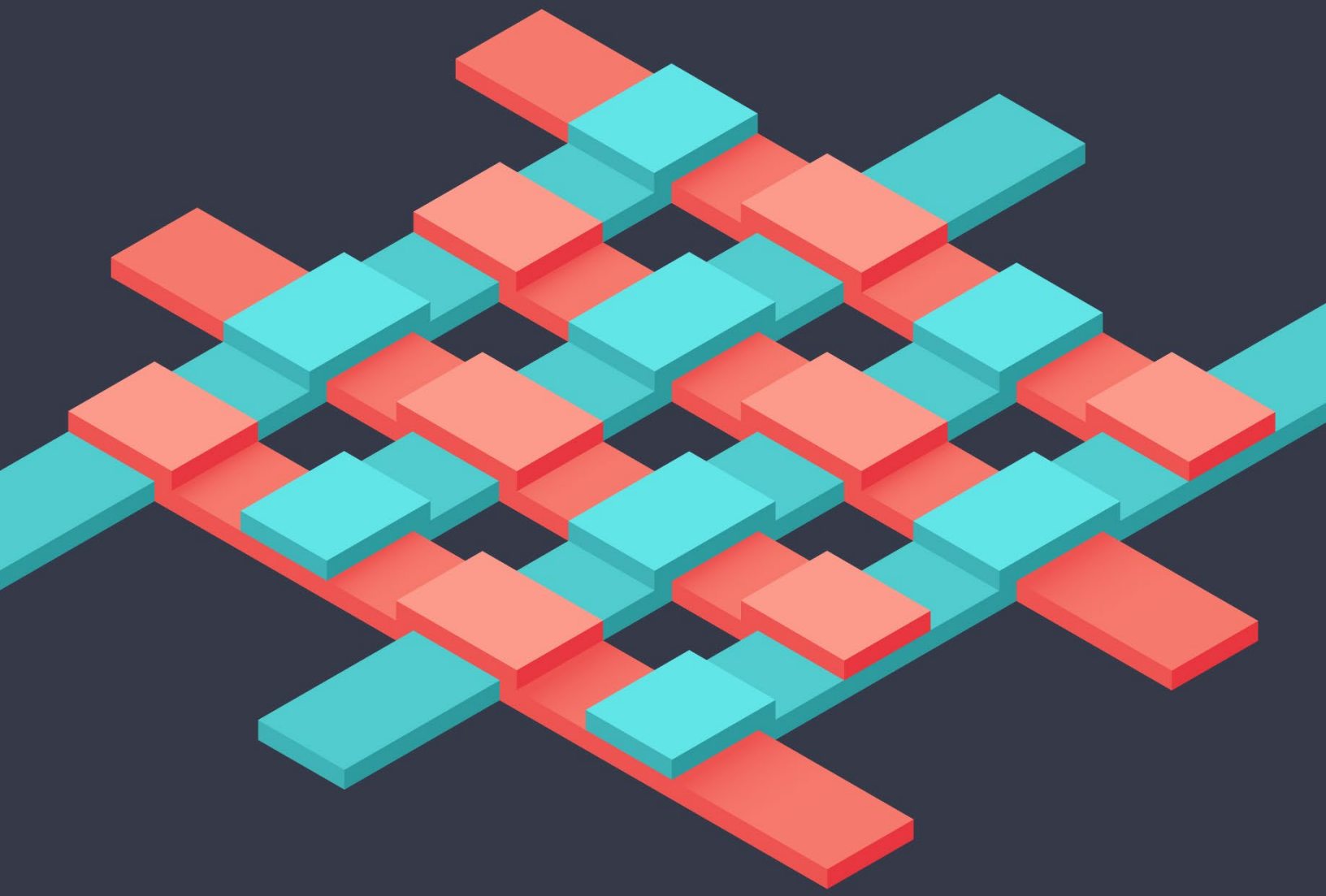
Hyperledger Fabric 2.0 Architecture Security Report



cloud security alliance®

Acknowledgments

Initiative Lead:

Urmila Nagvekar

Authors:

Carlos Dominguez
Urmila Nagvekar

Key Contributors:

John Carpenter
Frederic de Vaulx
Alex Ferraro
Ashish Mehta
Natividad Munoz
Teju Oyewole
Jyoti Ponnapalli
Ramesh Reddi
Michael Theriault
Huili Wang

CSA Staff:

Hillary Baron
Stephen Lumpe (Cover)
AnnMarie Ulskey (Layout)

Reviewers:

Goni Sarakinov
Kurt Seifried

Table of Contents

Executive Summary

As the foundational platform replacing the Internet of Information with the Internet of Value (Carter, 2019), blockchain technology is being rapidly adopted (Global Blockchain Business Council, 2020) (Hoffman et al., 2020) (Gartner, 2020) by enterprises to bring traceability and transparency to external business workflows and to instill trust and efficiency in an untrusted and competitive business environment (IBM, 2020). Considering that many of these external business workflows involve transactions and custody of value in the form of digital assets (European Commission, 2020) or other high-value data, cybersecurity attributes such as privacy, confidentiality, integrity, and availability certainly take center stage in the blockchain space (Birge et al., 2018).[1] A compromise of any of those attributes can result in a high business impact, namely loss of trade, loss of ownership and or loss of trust between the stakeholders (Chia et al., 2019).

In this Hyperledger Fabric 2.0 (Fabric 2.0)[1] Architecture Security Report, targeted for security and risk management leaders and regulators in the financial industry, we have aimed to mitigate the above-mentioned business impact in two ways:

1. We first identify Fabric 2.0's architectural risks to cybersecurity attributes (privacy, confidentiality, integrity, availability) (Angelis et al., 2019) while being implemented as a permissioned blockchain enterprise network for a trade finance business use case in a cloud-based environment.
2. We deliver a fully implementable "Security Controls Checklist" aligned with NIST Cybersecurity Framework's Controls[2] to proactively Prevent, Detect and Respond to the above-identified risks thus mitigating the business impacts downstream to the Trade Finance business workflow caused by Loss of Trade, Loss of Trust and Loss of Ownership.

Since this report is a part of Cloud Security Alliance (CSA)[3] a cloud environment was deliberately selected to house the Fabric network to leverage the CSA's expertise to securely manage the Physical Infrastructure of the Fabric 2.0 permissioned blockchain network.

The scope of the risks identified and the corresponding security countermeasures recommended have been restricted to the design and development stage of the Hyperledger Fabric 2.0 network environment in order to enable security and risk management leaders new to Hyperledger Fabric to quickly come up to speed with the associated organizational risks needed to estimate the operational costs while balancing the security needs with the business priorities.

[1] Hyperledger Fabric 2.0 https://www.hyperledger.org/blog/2020/01/30/welcome-hyperledger-fabric-2-0-enterprise-dlt-for-production

[2] NIST CSF Framework https://www.nist.gov/cyberframework

[3] Cloud Security Alliance https://cloudsecurityalliance.org/about/

Key Findings

The risk identification process comprises a trade finance workflow between a typical importer and exporter (Copigneaux & European Parliament, 2020) running on a Hyperledger Fabric 2.0 permissioned blockchain network within a cloud environment. It was carried across all three layers of Gartner's blockchain security model (Gartner, 2018), namely, that is, business, risk & IAM process and technology/IT layers and included the following:

1. Threat evaluation to trade finance business logic confidentiality and privacy as well as execution and resiliency
2. Threat modeling of the blockchain network and IAM process with the trade finance workflow at runtime

Fabric 2.0 permissioned blockchain network was found to be natively secure by design and default when it came to trade finance business logic and payload confidentiality and privacy.

It was also robust in preventing adversaries from manipulating trade finance's business logic during execution.

Fabric 2.0 architecture threat analysis identified 14 high[4] impact and high likelihood threats with 50% of them stemming from compromised administrative credentials with "elevated privileges."

The above findings demonstrate how a decentralized administration of the fabric system and certificate authority, coupled with lack of robust governance policies to secure administration channels and credentials from compromise, could expand the attack surface considerably, aiding the leap from "establishing foothold" into the trade finance fabric network to potentially compromising the entire fabric network and resulting in a high business impact with loss of trade, loss of ownership and loss of trust between the importer and the exporter in the trade finance workflow.

[4] "High" is as risk methodology definition described in Section "Risk Identification Process"

Introduction

Overview - Fabric Implementation of Trade Finance WorkFlow

A Hyperledger Fabric 2.0 permissioned blockchain network was used to depict a simple transaction within a trade finance workflow[5]: the sale of goods from one party to another — a traditionally complicated transaction between the buyer and the seller from different countries with no common trusted intermediary to ensure that the exporter gets the money it was promised and the importer gets the goods it was promised.

Fabric, with its inherent properties of immutability [permanency of recorded transactions] and distribution [definition and validation of transactions across a multi-participant network] (IBM, 2020), enables both transparency and traceability to this traditional workflow by connecting all the authorized trade finance participants [importer and importer's bank, exporter and exporter's bank, carrier and regulator] together via the Fabric blockchain and synchronizing the transactional state of the distributed ledgers across all the participants respectively.

A software-based smart contract developed for the Fabric network had the trade finance business logic embedded within it; namely, a payment promise is made by the importer's bank to the exporter's bank, though in two installments. The exporter obtains a clearance certificate from the regulatory authority, hands off the goods to the carrier, and obtains a receipt. Production of the receipt triggers the first payment installment from the importer's bank to the exporter's bank. When the shipment reaches the destination port, the second and final payment installment is made, and the process concludes. The details of this workflow are listed below.

1. Importer requests exporter for goods in exchange for money
2. Exporter accepts the trade deal
3. Importer requests a letter of credit (LC) from its bank in favor of the exporter
4. The importer's bank supplies an LC in favor of the exporter, and payable to the latter's bank
5. The exporter's bank accepts the LC on behalf of the exporter
6. The exporter applies for an E/L from the regulatory authority
7. The regulatory authority supplies an E/L to the exporter
8. The exporter prepares a shipment and hands it off to the carrier
9. **(a)** The Carrier accepts the goods after validating the E/L, and **(b)** supplies a B/L to the exporter
10. The exporter's bank claims half the payment from the importer's bank
11. The importer's bank transfers half the amount to the exporter's bank
12. The carrier ships the goods to the destination
13. The importer's bank pays the remaining amount to the exporter's bank

A traditional trade finance workflow with banks as intermediaries and a corresponding Fabric implementation of the same are depicted in Figure 1 and Figure 2 respectively.

[5] Trade finance scenario description and workflow diagram from Hyperledger Fabric GitHub Repository. See https://github.com/HyperledgerHandsOn/trade-finance-logistics.

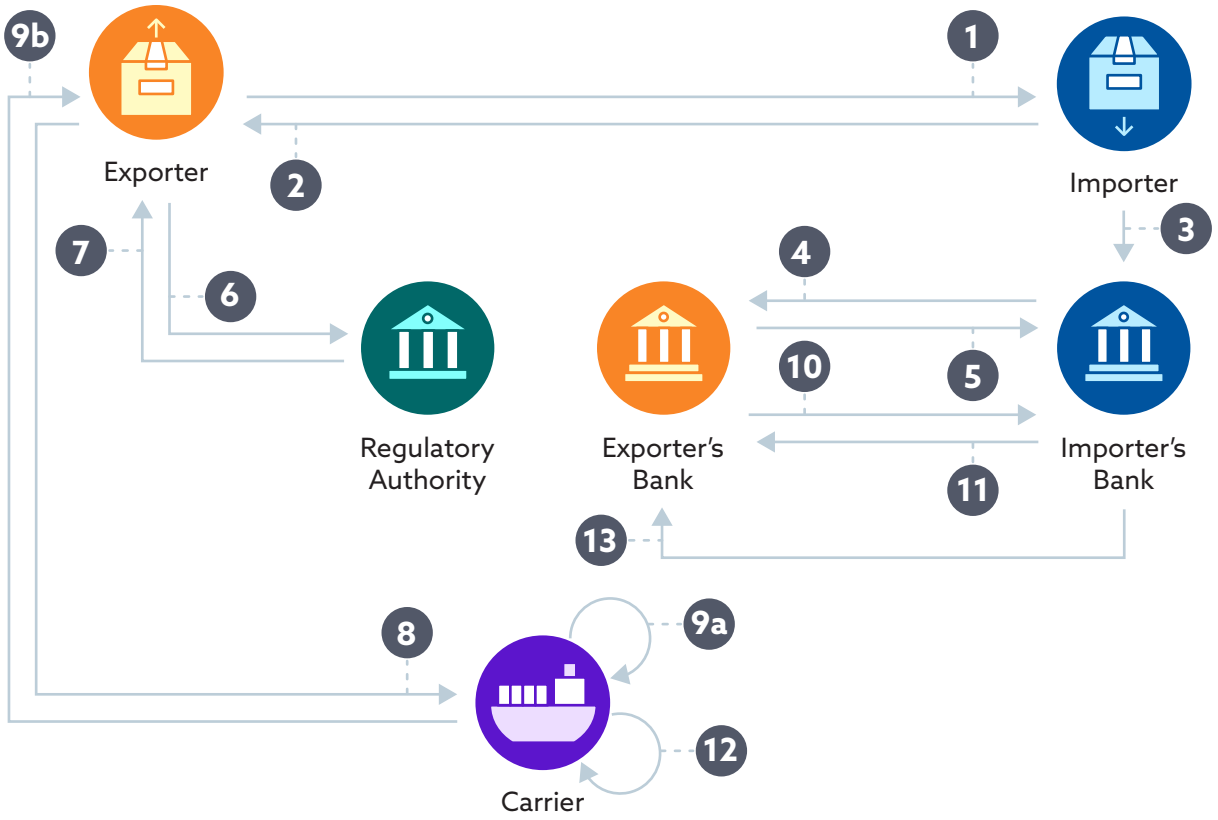


Figure 1: Business WorkFlow Diagram for a Trade Finance Use Case

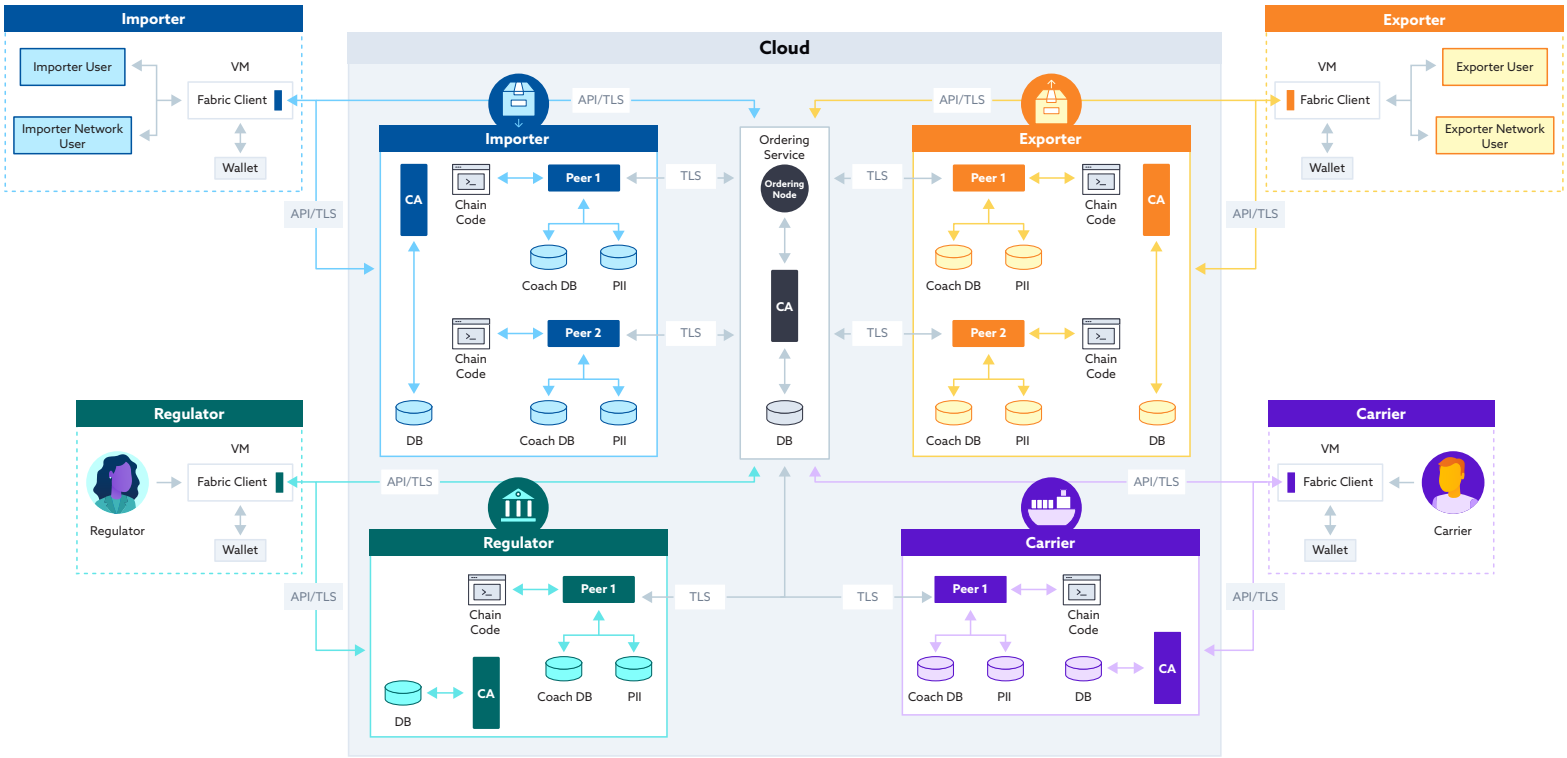


Figure 2: Fabric Implementation of the Trade Finance WorkFlow within a Cloud Environment

Blockchain networks in general have considerable implementation costs. An organization can merge its blockchain implementation with another to reduce these costs. In Figure 2 an importer organization and importer's bank together represent the Fabric nodes designated "importer" in the cloud, while the exporter organization and exporter's bank together represent the Fabric nodes designated "exporter" in the cloud respectively. Access to the Fabric nodes is permitted via valid certificates and keys together known as "identities" in the Fabric network. Each Fabric stakeholder organization hosts its own certificate authority (CA) for issuing "identities" to its users. For these "identities" to be available during login, they are usually stored in repositories called "wallet" and are easily accessible by the "Fabric client" as shown in Figure 2.

The trade finance business workflow is contained within the chaincode (i.e., smart contract) that resides on the peer nodes. The importer organization's user activates this workflow by submitting the transaction request to the exporter for "goods." Linux Foundation's Accord Project[6] was used to enable this software-based trade finance workflow to be legally binding with actual legal clauses and obligations as found in a typical commercial agreement. Accord Project is a nonprofit, collaborative initiative for developing an ecosystem and open source tools for legally enforceable machine-readable agreements called smart legal contracts (European Commission, 2019); their objective is to help reduce friction and transaction costs in creation and management of commercial relationships (Linux Foundation Projects, 2017).

Figure 3 and Figure 4 below depict sections of an executable smart legal supply agreement developed using the Accord Project's Cicero and Ergo tools for the trade finance workflow.

Smart Legal Supply Agreement for Trade Finance

This SUPPLY AGREEMENT (together with all schedules attached hereto, the "Agreement") is entered into as of {{ExecutionDate}} (the "Execution Date") between {{Exporter}}, a Delaware corporation ("Exporter"), having a principal place of business at 12345 Main Street, California 92705, and {{Importer}}, a company organized and existing under the laws of Sweden ("Importer"), with a place of business at 39E, TunaVagen Stockholm - SE114 55.

1.
Supply and Purchase of Products. {{Exporter}} shall supply and {{Importer}} shall purchase {{Product}} (the "Products", {{Shipment}}) in accordance with the terms of this agreement.

2.Creditworthiness Qualification for {{Importer}}. The {{Importer}} shall only be able to import {{Product}} based on their creditworthiness as determined in the sole and exclusive discretion of the {{ImporterLOCBank}}.

Letter Of Issue Date: {{IssueDate}}
{{ImporterCreditworthiness}}
L/C Number: {ImporterLOCNumber}}
{{ImporterLOCBank}} hereby issues this irrevocable documentary Letter of Credit, {{ImporterLOCNumber}} to {{Importer}} for {{ImporterLOCAmount}}.
An initial payment of {{% {{ImporterLOCAmount}}/2 %}} will be made immediately upon sight by a draft drawn against {{ImporterLOCBank}} in accordance with {{ImporterLOCNumber}}.
The draft is to be accompanied by the following documents:
1.{{OrderBillOfLading}}
2.{{PackingList}}
3.{{Invoice}}
The remaining {{% {{ImporterLOCAmount}}/2 %}} will be made upon acceptance of the {{Product}} by the {{Importer}} defined in section 6. below.

1.Orders.

1. Purchase Orders. The {{Importer}} shall submit all orders for {{Product}} to the {{Exporter}} in writing to the {{ExporterAddress}} and include in each {{PurchaseOrder}}
   2.each {{Product}} it is ordering, identified by model or part number,
   3.the {{AmountOfEachProduct}} it is ordering,
   4.the {{UnitPriceOfEachProduct}} it is ordering,
   5.the {{LocationForDelivery}}, and
   6.the {{DeliveryDate}}, allowing reasonable time for {{Exporter}} to receive, review, process the {{PurchaseOrder}}, and ship the {{Product}} by the {{DeliveryDate}}.
   7.the {{Importer}} {{ImporterLOCAmount}}

2.Accepting, Modifying, and Rejecting Purchase Orders

1. By Notice. Within {{TurnaroundTime}} Business Days' of receiving a {{PurchaseOrder}} from {{Importer}}, {{Exporter}} shall accept, reject, or propose a modification to the {{PurchaseOrder}} by sending the {{Importer}} written notice of its acceptance, rejection, or proposed modification.
2. Acceptance. {{Importer}} shall notify {{Exporter}} of its acceptance, rejection, or proposed modification of the {{PurchaseOrder}}.
3. An acceptance notice will include {{ExporterBankAccount}} information.
4. Modification of Purchase Order. {{Exporter}} may propose a modification to a {{PurchaseOrder}} by including in its notice to {{Importer}} a {{ModifiedPurchaseOrder}} for the {{Importer}} to accept or reject according to the acceptance and rejection procedures under paragraphs 3.b.i and 3.b.ii.
5. Canceling Purchase Orders. The {{Importer}} or {{Exporter}} may, at no expense to itself, cancel part or all of a {{PurchaseOrder}} up to {{CancellationDeadline}} Business Days before the {{DeliveryDate}}

Figure 3: Actual Smart Legal Supply Agreement for the Trade Finance Business WorkFlow

[6] Accord Project https://accordproject.org/about

Figure 4: Bill of Lading, Acceptance and Limited Warranty Sections of the Smart Legal Supply Agreement for the Trade Finance Business WorkFlow

Scope for Fabric's Architectural Threat Model

Since the risk identification primarily focused on providing insights into the architectural risks of Fabric 2.0 during the design and development stages of a permissioned blockchain network, both the Fabric and IT operational environments, as well as smart contract software and governance topics are out of scope.

The following scope determinations are as per the selected use case, its architecture, and choice of infrastructure (Cloud Managed Services Provider). The scope narrowed the number of potential threats under consideration from the pool of Hyperledger Fabric threats already reported, some of which have been superseded by Fabric 2.0. (Baset et al., 2018) (Dabholkar & Saraswat, 2019)

In Scope

Fabric Specific:

- Detailed threat model of Fabric 2.0's Identity & Access Management (IAM) and technology architecture
- Detailed threat evaluation of Fabric 2.0 for business logic privacy, confidentiality, execution and resiliency
- Fabric trust boundaries

- Fabric subsystem components
- Fabric system data flow
- Pluggable consensus mechanism variants (RAFT)[7]
- Pluggable cryptographic algorithms
- Fabric nodes housed across regions within a single cloud service provider

General:

- Privacy regulation requirements - secure by design and default
- Threat model evaluation using a real-life financial use case

Out of Scope

Fabric Specific:

- Chaincode software threat model
- Fabric nodes housed across multiple cloud service providers
- Fabric / IT operational environment
- Fabric network governance
- Decentralized smart contract governance
- Fabric network integration considerations with downstream enterprise financial systems
- In-depth analyses of cryptographic algorithms for consensus
- IT components of Fabric certificate authority (PKI, wallets or other key storage options)

General:

- Data management considerations
- IT processes/components not unique to the fabric's functionality
- (change and vulnerability management for node or client platform; web servers; IT communication network, etc.)
- Cloud service provider configuration modules for IAAS implementation

Risk Identification Process

Methodology

The risk identification process comprises a trade finance workflow between a typical importer and exporter running on a Hyperledger Fabric 2.0 permissioned blockchain network within a cloud environment. It was carried across all three layers of Gartner's blockchain security model namely, business, risk and IAM process and technology/IT layers and included the following (Gartner, 2018):

- Threat evaluation of trade finance business logic confidentiality and privacy as well as execution and resiliency

[7] RAFT is the consensus mechanism in use by Hyperledger Fabric 2.0. See https://hyperledger-fabric.readthedocs.io/en/release-2.2/glossary.html?highlight=RAFT#raft.

- Threat model analysis of Fabric 2.0 permissioned blockchain network and IAM process with the trade finance workflow at runtime

The Threat model analysis was executed using STRIDE methodology (Shostack, 2014), which included the identification of trust boundaries in the architecture as well as a review of the information flows, relevant data, actors, potential threats and their actions.

Threat Evaluation of Trade Finance Business Logic

Fabric 2.0 architecture was evaluated for compromise to the confidentiality and privacy of both the trade finance business logic as well as the transaction and its payload.

The architecture was also evaluated for weaknesses in its operational semantics[8], to ensure trade finance business logic embedded within smart contracts cannot be manipulated by adversaries during execution to gain financial advantage.

Fabric 2.0 was specifically evaluated for vulnerabilities that have been the root cause of prior business execution compromises in non-Fabric blockchain environment (Dika & Nowostawski, 2018; Dingman et al., 2019; Perez & Livshits, 2020; Albreiki et al., 2020; Praitheeshan et al., 2020).

These vulnerabilities include the following:

1. Non-deterministic transactions within smart contracts: Executing non-deterministic transactions can cause inconsistencies in peer states causing them to diverge. Since blockchains operate on the main premise that the state of all peers must be the same after executing a transaction, a non-deterministic transaction can cause a ledger to "fork."
2. Transaction duplication: Also known as "Double Spending" in which a digital asset state is included in multiple illegitimate transactions, effectively creating new copies of the asset.
3. Timestamp dependency: conditions triggered by introducing logic that depends on blocks timestamps which is not a valid source of timing for smart contract logic. An example is using the block timestamp as a random number generator.
   - In Hyperledger Fabric transaction latency can be computed as the delay between transaction timestamp and the block timestamp of the transaction's block. This metric is thus computed for each transaction upon block inclusion. By subscribing to Hyperledger Fabric's channel Events, this metric can be computed for each transaction in each block signed by the ordering service.
   - Outgoing transactions signed by a Hyperledger Fabric SDK contain the transaction hash and timestamp. These can be sent to and tracked by a monitoring service (push).
4. Transaction ordering dependency: Two dependent transactions invoke the same contract and are part of the same block. In such a case there is a discrepancy between the contract state which the caller wishes to invoke and the actual state when the execution happens.
5. Third-party trusted services (Oracles): Third-party trusted services, commonly known as Oracles, are one of the mechanisms for extending smart contracts by implementing off-chain logic that maintains trust, visibility, and transparency as qualities of service for a blockchain network (IBM, 2019).

[8] The term "Operational Semantics" is used to indicate how the operational logic of the architecture was evaluated

Details of the evaluation results are in Section titled "Findings"

Threat Analysis as per STRIDE Model

A detailed threat model of the Hyperledger Fabric 2.0 permissioned blockchain network and IAM process was evaluated for the trade finance business transaction at run time within a cloud IAAS deployment. The analysis was conducted in several steps as shown below:

1. Identify Fabric 2.0 permissioned network's subsystems
2. Delineate/decompose Fabric 2.0 permissioned network's trust boundaries (physical and logical)
3. Detail the trade finance workflow on the Fabric 2.0 permissioned network at runtime
4. Identify vulnerabilities in the trade finance workflow at runtime using STRIDE
5. Determine the risk by rating the likelihood and impact of the vulnerabilities
6. Group the vulnerabilities by cybersecurity functional areas

Step 1 - Identify Fabric 2.0 Permissioned Network's Subsystems

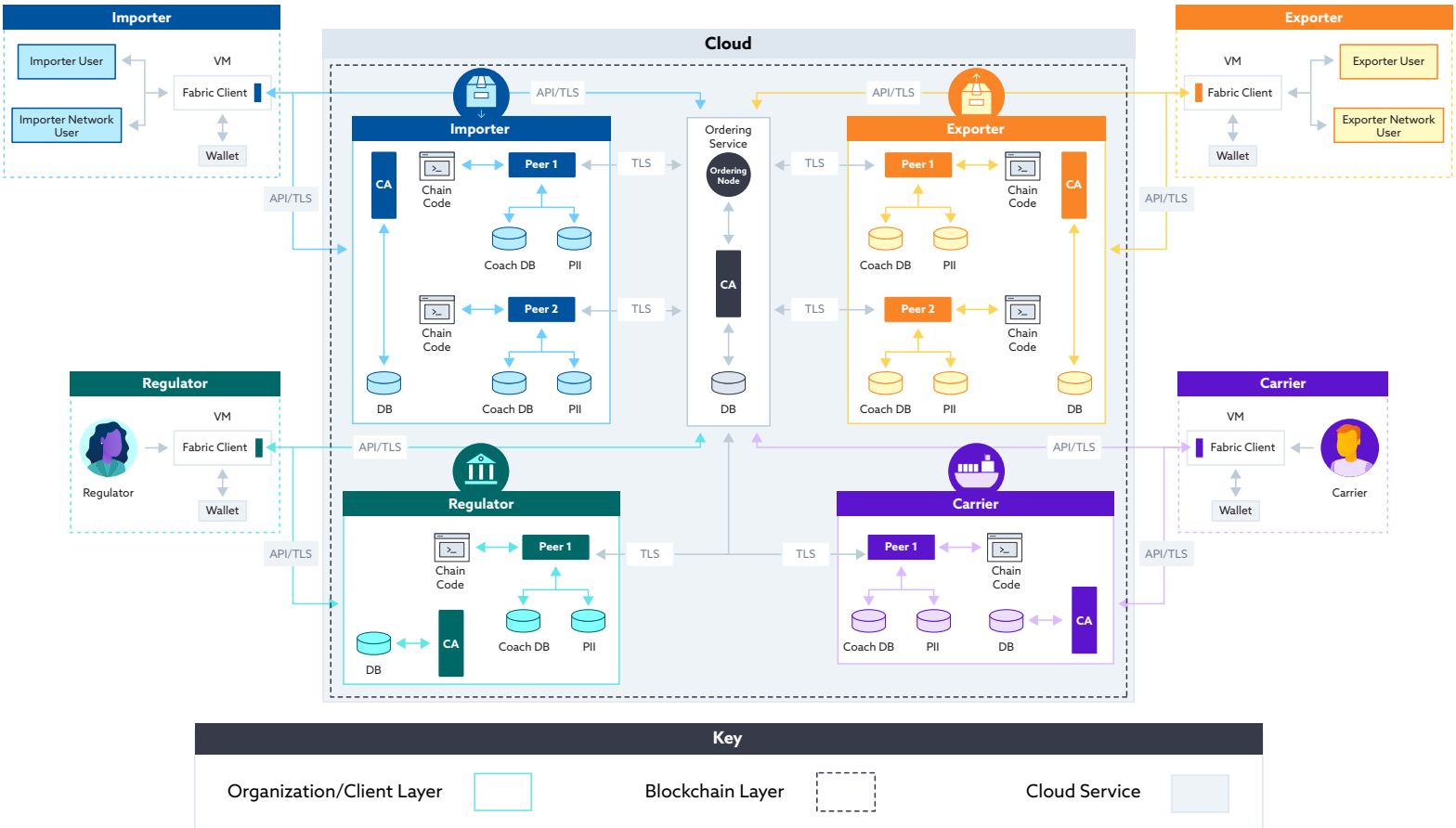


Figure 5: Three layers that comprise the Fabric 2.0 Subsystems

A typical Hyperledger Fabric 2.0 implementation in an IAAS configuration within a cloud environment will have three distinct layers, as shown in Figure 5 above, that together comprise the system:

Layers	Description
Fabric Network @ cloud service provider's data center *(denoted by "cloud" in the Figure 5 above)*	This is the principal layer where the Fabric resides mostly as a network of nodes on virtual machines (VMs) within a cloud IAAS configuration. Its main components are: • Fabric-Certificate Authority (Fabric-CA) – authentication and authorization services for Fabric clients, peer & orderer nodes. There is a Fabric-CA per organization. • Peer and orderer nodes – transaction processing center • Chaincode on the peer nodes – transaction processing trigger The communication protocol between peer & orderer; peer & Fabric-CA is via TLS. The communication protocol between any two peers (P2P) is Gossip. The communication protocol between the Fabric network and client is TLS. The chaincode on the peer resides within Docker containers.
Client Network @ client premise *(denoted by "importer," "exporter," "carrier," and "regulator" in the Figure 5 above)*	Trade finance clients constituting importer, exporter, carrier, and regulator reside on this layer. They use their host machines to log in to the Fabric APIs through the Fabric SDK client. The Fabric SDK client controls the "client" access using a combination of the role and attribute-based control. Authentication of users is via the Fabric's certificate authority provider while authorization of the users to access the fabric network is via its membership service provider functionality. Importer on the client network invokes the chaincode (smart contract) as part of the transaction proposal request to the exporter. this initiates the execution of the chaincode on the endorsing peers in turn triggering the processing of the business transaction. The chaincode resides within a Docker container.

Organization Node Network @ cloud service provider's data center *("importer" denoted by the blue shaded peers and the Fabric-CA; "exporter" denoted by green shaded peers; "carrier" by orange shaded peers, "regulator" by purple shaded peers and "ordering Service" by brown shaded peers in the Figure 5 above)*	Each Organization (Org) has its own set of peer nodes and optionally orderer nodes (if consensus policy is decentralized). There is at least one endorsing peer node and one anchor peer node from each organization. The Transaction's world state resides within a database (Couch DB) at each of the peer nodes. Optionally, to comply with GDPR "Secure by Design"[9] principle, any private or confidential data within a transaction must be stored securely within a separate database at each of the peer nodes while its hash is carried in the transaction's world state.

Step 2 - Decompose / Delineate Fabric 2.0 Permissioned Network's Trust Boundaries (Physical and Logical)

Physical Trust Boundaries

Hyperledger Fabric Network implemented in a cloud environment as an IaaS configuration comprises the following physical trust boundaries as shown in Table 1.0 and Figure 6 respectively.

Area	Description
Fabric Client Trust Boundary	This trust boundary segregates the fabric environment residing in the cloud from the various client environments. The importer org business user, exporter org business user, regulator org user, and carrier org user can all access the fabric network via a Fabric Client API.
Cloud Service Provider Trust Boundary	This trust boundary houses the fabric network that is compromised of the identity service, the ordering service, and the peers along with their operational data stores.
Ordering Service Trust Boundary	This boundary segregates the consensus trust model, critical to the integrity of the fabric network from the potentially Byzantine peers and clients.
Membership Service Provider Trust Boundary	This trust boundary is internal to the cloud, authorizing authenticated fabric client's access to the fabric peer and orderer nodes.
Peer Trust Boundary	This trust boundary houses peers of a single organization. Peers within an organization trust each other but do not trust peers of another organization.

Table1.0: Fabric 2.0 Trust Boundaries

[9] Local government regulations and laws supersede all recommendations made in this document.

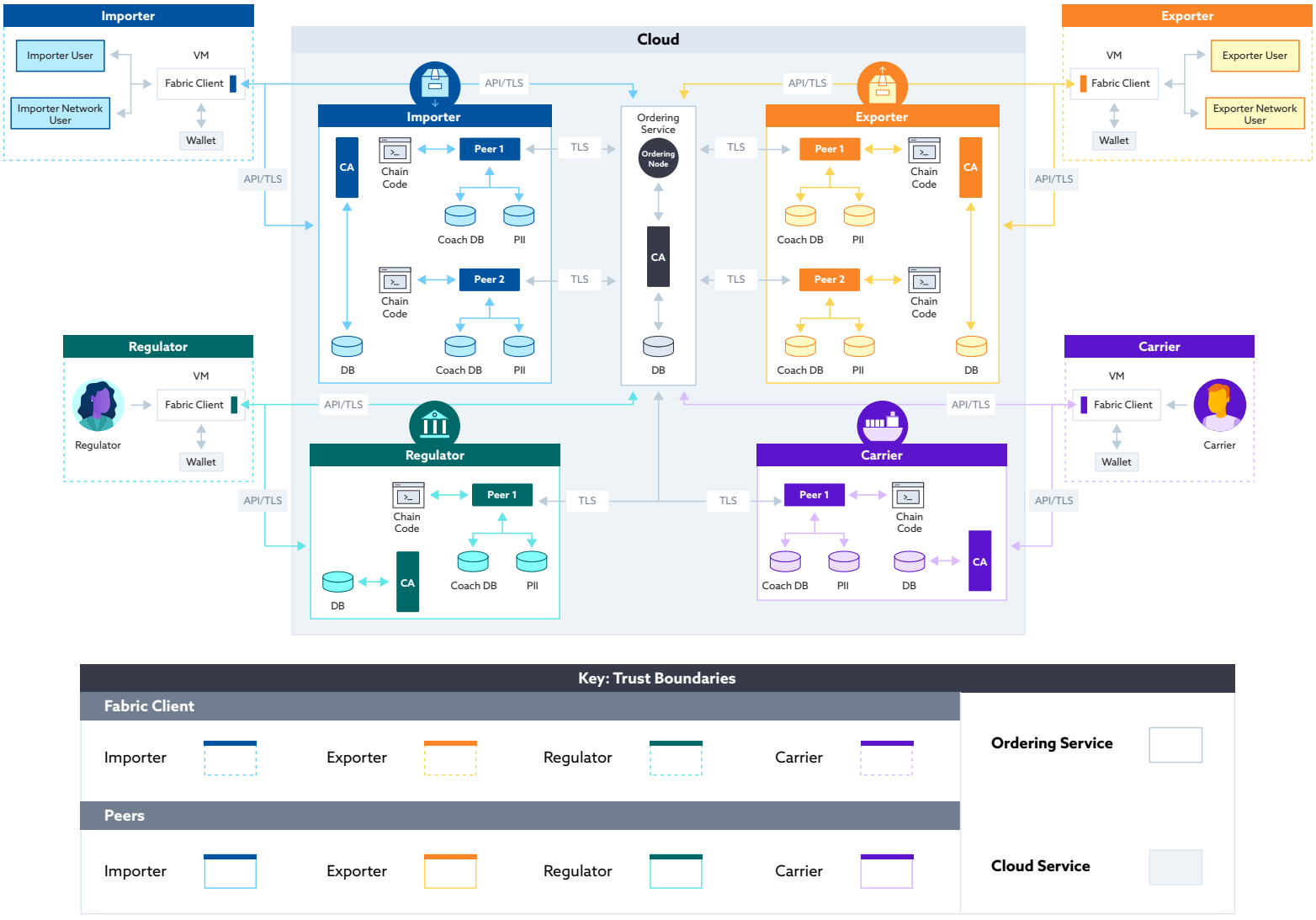


Figure 6: Fabric 2.0 Physical Trust Boundaries

Logical Trust Boundaries

Channels in Figure 7 below comprise the logical boundaries depicting how transactions between two parties can be carried out privately and confidentially. Fabric's chaincode instantiates the "Application Channel," allowing only organization peers that have a "need to know" or a "need to transact" with each other. The "System Channel," as the name indicates, is used for all communication between the peer nodes and between the peer and the orderer nodes. The membership service provider is a logical representation of the services provided by Fabric to authorize "client" access to the peer and orderer nodes.

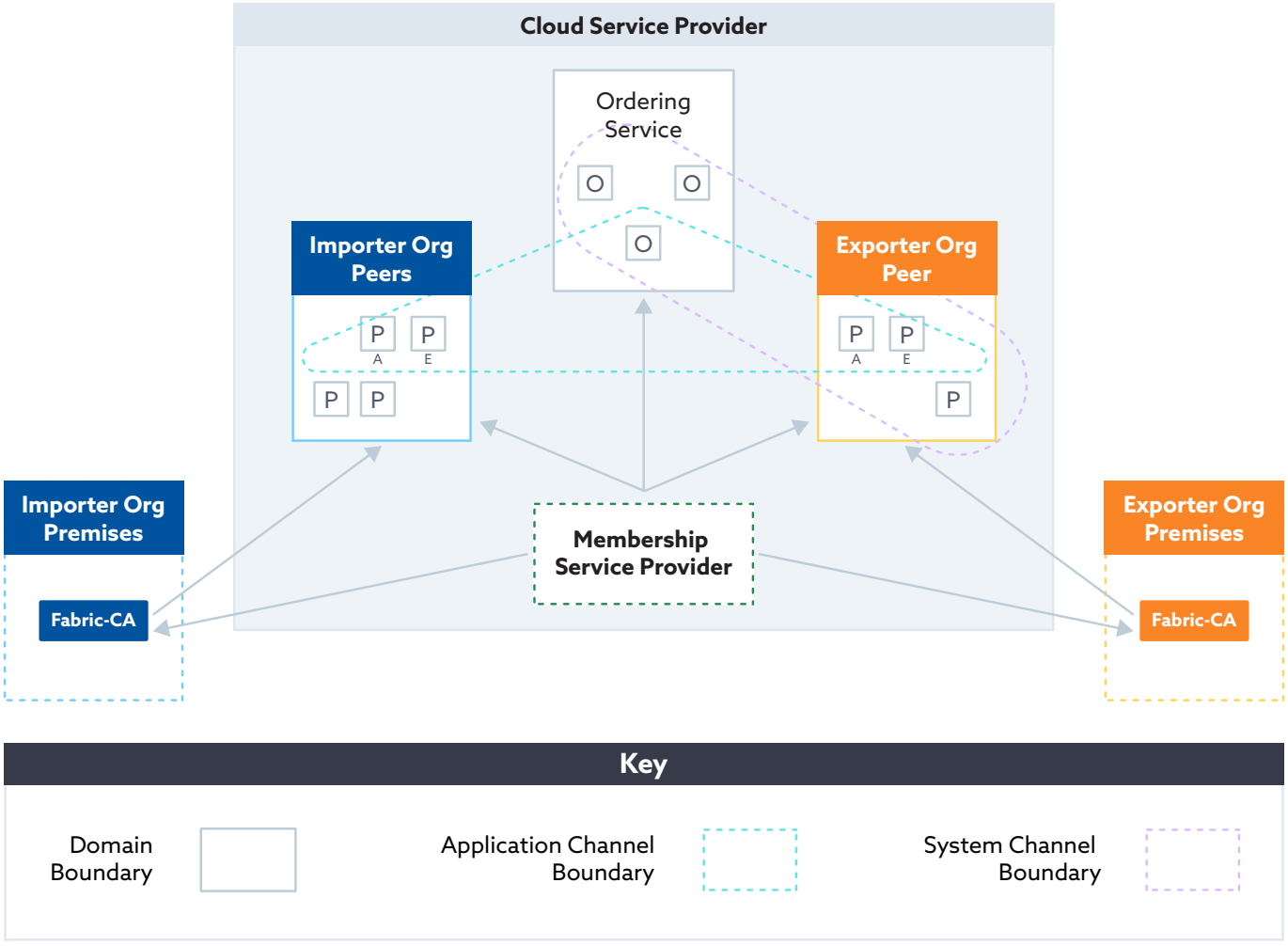


Figure 7: Fabric 2.0 Logical Trust Boundaries

Step 3 - Detail the Trade Finance WorkFlow on the Fabric 2.0 Permissioned Network at Runtime

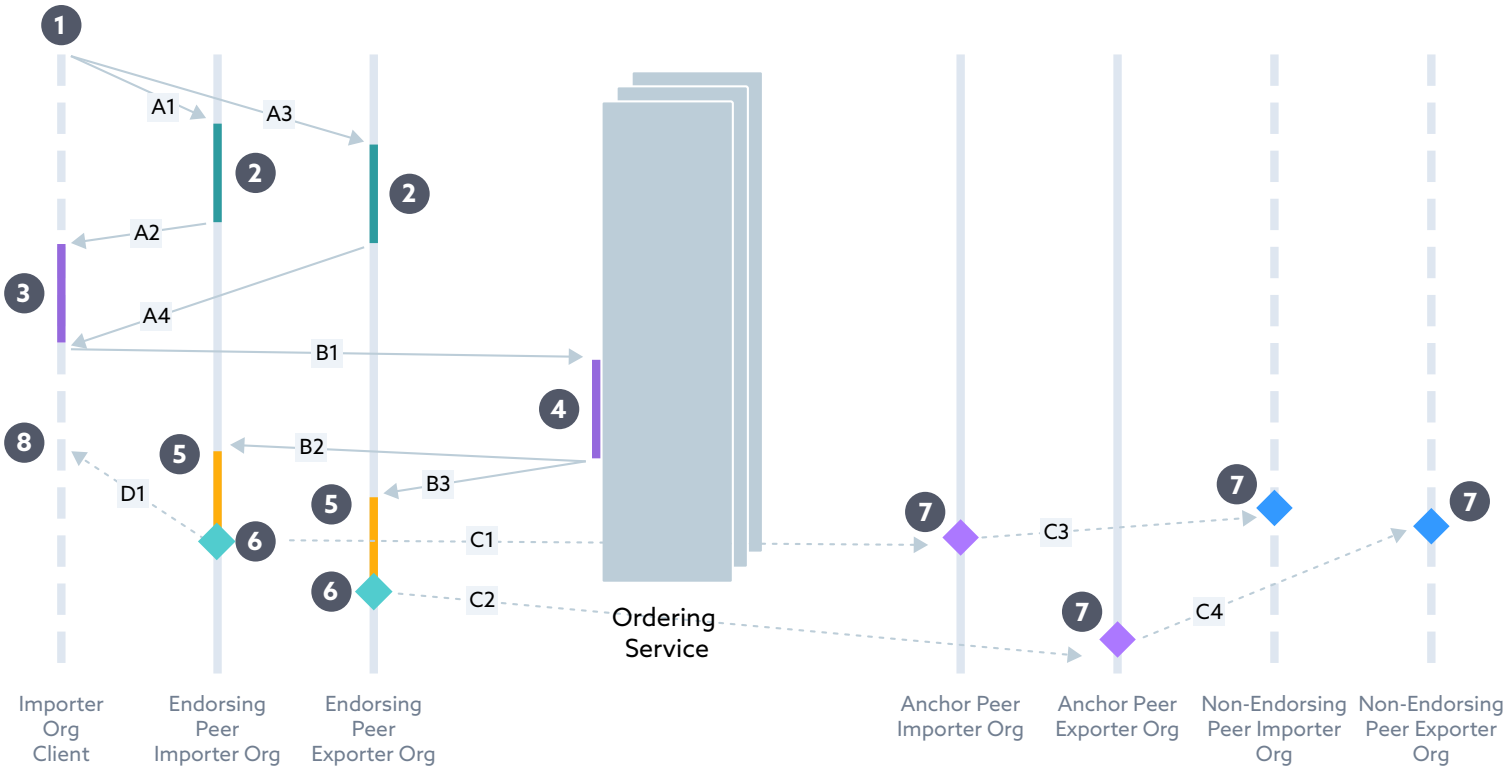


Figure 8: Anatomy of a Private Trade Finance Transaction Flow within Fabric 2.0

Figure 8. above details the flow of a Transaction containing Personally Identifiable Information (PII) from start to finish within the Fabric 2.0 Network.

- **Step 1:** (A1,A3) - The Importer Org Client invokes the Smart Contract containing the signed Private Transaction Proposal on Endorsing Peers in the Channel. The Endorsing Peers are pre-specified by the Channel members using the Endorsement Policy.
- **Step 2:** (A2, A4) - The Endorsing Peers execute the locally installed Smart Contract against their local ledgers resulting in a proposal response to the Importer Org Client. The actual PII within the Private transaction is stored separately in databases at the endorsing peers of the Importer & Exporter Orgs while only hashes are carried forward in the proposal response to prevent unauthorized access to the PII by the Regulator and Carrier Orgs
- **Step 3:** (B1) - The Importer Org Client collects the proposal responses and when a defined number satisfying the Endorsement Policy is reached sends them over to the Ordering Service.
- **Step 4:** (B2,B3) (B4,B5) - The Ordering Service running RAFT orders this transaction along with those received from other clients within the same Channel, groups them into a hash chained sequence of Blocks, and delivers the Blocks to Endorsing Peers.
- **Step 5:** The Endorsing Peers within the Channel validate the Transaction Block
- **Step 6:** The Endorsing Peers commit the Transaction to the Ledger. The Blockchain World state is updated with the new Transaction Block.

- **Step 7:** (C1, C2) (C3, C4) - The committed Transaction is forwarded to the Anchor Peers who further deliver it to the Non-Endorsing Peers.
- **Step 8:** (D1) - The Importer Org Client is notified of the results of the committed Transaction via the Smart Contract.

Step 4 - Identify Vulnerabilities in the Trade Finance Workflow at RunTime using STRIDE

A detailed threat analysis of the Trade Finance Workflow while being executed on the Fabric 2.0 Permissioned Network is carried out using Microsoft's "STRIDE" Threat Modeling Methodology *(Shostack, 2014)*. Figure 9. below enumerates the Threats that "STRIDE" stands for.

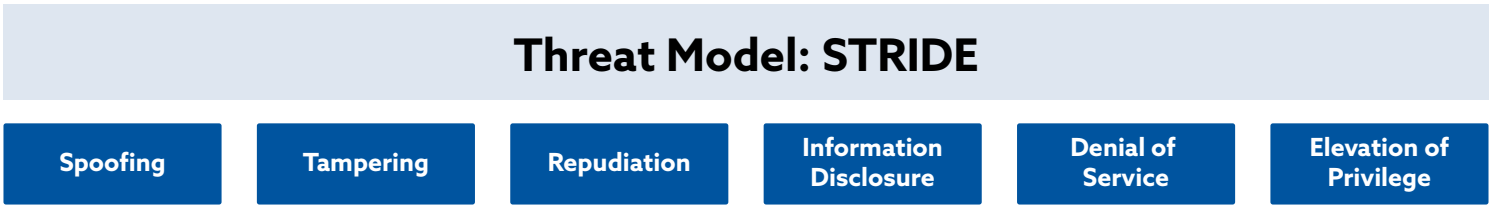


Figure 9: STRIDE mnemonic expanded

The "STRIDE" analysis consists of identifying the vulnerabilities in the Fabric System that could be exploited by the enumerated Threats for each of steps of the Trade Finance WorkFlow across the various Fabric Trust Boundaries viz., Fabric Client, Ordering Service, Membership Service Provider and Peer Trust Boundaries as listed in Table 1.0 in Step 2. above. The identified vulnerabilities are collectively rated for their risk as shown in Figure 13. in Step 5. below.

This effort being part of the Cloud Service Alliance (CSA) its well known Cloud Control Matrix (CCM) has been used to secure the Cloud Service Provider Trust Boundary (listed in Table 1.0) as well as the physical infrastructure layer of the Fabric Network Implementation.

A sample application of STRIDE to the transmission of the Transaction Proposal as it flows from the Importer's Org to the Endorsing Peers across the Trade Finance Client Trust Boundary is shown in Figure 10. below.

Note: The goal of this effort was to provide an insight into how the Trade Finance Business WorkFlow could be impacted by Fabric's architectural design vulnerabilities . Hence the source of common IT threats like compromise of Fabric Administration Accounts with "Elevated Privileges'' have not been deep dived into, focusing instead on the consequences of such a compromise to the Trade Finance Business WorkFlow.

Trade Finance Client Trust Boundary						
Action: Invoke Chain Code	**Importer Client @ Importer Or**		**Data Flow >>>> <<<<**		**Endorsing Peers**	
Stride Methodology	**Vulnerabilities**	**Mitigations**	**Vulnerabilities**	**Mitigations**	**Vulnerabilities**	**Mitigations**
Spoofing	Client Host machine compromised; Transaction Proposal hijacked	End Device Security needs to be in place	-	-	-	-
	Client's Unrevoked expired digital credentials Spoofed by malicious actor to extend authorization to send transaction proposal	Policy to REVOKE expired certificates needs to be in place Fabric has the ability to generate Certificate Revocation Lists	-	Malicious node poses as valid client	Fabric natively allows for signature validation at the node level	-
Tampering	Chaincode Access Control Policy tampered to include unregistered User	Ensure access to ChainCode Access Control Policy is restricted to Fabric Admin ONLY	Unencrypted Transaction Proposal intercepted as it flows over internet from Client to Endorsing Peers	Ensure TLS for Data Transmission	Compromised Peer tampers with local ledger	-
Repudiation	API logs missing tracebility of system operations done by Admins [in production] and developers in dev	-	-	-	-	-
Information Disclosure	-	-	Proprietary Transaction Data gets leaked with Transaction proposal intercept	Encrypt Transaction proposal if confidentiality needed	Confidentiality of Proprietary Transactions is disclosed to all endorsing peers of Orgs in channel; MSP Admin of these Orgs & to Orderer Admin	Ensure Transaction proposal is encrypted if transaction data is to remain confidential

Denial of Service	-	-	-	-	Untrusted chaincode issues DoS attack against Endorsing Peer	Run Untrusted Chaincode within Docker containers
Elevation of Privilege	With Fabric Client directly calling Blockchain Network's API there is a HIGH likelihood that Web Vulnerabilities or End device could compromise Fabric Admin or MW Admin accounts	Segregate Client API from the Blockchain Network API using a middleware layer				

Figure 10: Application of STRIDE at the Client Trust Boundary during transmission of Importer's Transaction Proposal

Step 5 - Determine the Risk by rating the Likelihood and Impact of the Vulnerabilities

The Vulnerabilities identified using STRIDE are then rated for their likelihood and impact and the Risk of Compromise to the Fabric 2.0 Network is determined.

As shown in the Figure 11[10] below Likelihood involves rating Attack Vector, Weakness Prevalence[11] and Weakness Detectability[12], while Impact is determined by rating the Technical Impact of the Vulnerability being exploited by the Attack Vector. The ratings scores are assigned as per Subject Matter Experts judgment on the vulnerability specifics.

The Risk Rating Methodology is not a Quantitative Risk calculation but a Qualitative one. The use of Qualitative methods to support refining the results of Threat Models, such as the ones produced by STRIDE, is an industry practice for deriving insights as per technical factors present during design (Jones, 2019). The actors behind the attack vectors are considered to be Advanced Persistent Threats[13] (APT) as per industry reports *(Allianz, 2021)(Crowdstrike, 2021)(Verizon, 2020)*. APT-type threats are assumed to have a relatively high contact frequency which simplifies the risk calculation.

[10] Figure 11 is a simplified version of OWASP Risk Rating Methodology. See https://owasp.org/www-community/OWASP_Risk_Rating_Methodology

[11] As defined by MITRE "How frequently this type of weakness appears in software" See: https://cwe.mitre.org/cwss/cwss_v1.0.1.html

[12] As per OWASP Risk Factors defined as "How easily is to be detected by an attacker". See: https://owasp.org/www-project-top-ten/2017/Details_About_Risk_Factors

[13] Refer to APT definition: https://en.wikipedia.org/wiki/Advanced_persistent_threat

Score	Attack Vector	Weakness Prevalence	Weakness Detectability	Technical Impact
3	Easy	Widespread	Easy	Severe
2	Average	Common	Average	Moderate
1	Difficult	Uncommon	Difficult	Minor

Likelihood = Average (Attack Vector, Weakness Prevalence, Weakness Detectability)
Risk Rating = Likelihood * Technical Impact

Figure 11: Calculating Risk by estimating Likelihood and Impact

The Risk Rating[14] is calculated by multiplying the Technical Impact rating and the average of the Vulnerability ratings (Attack Vector, Weakness Prevalence, Weakness Detectability). As an example the following Vulnerability has a Risk Rating of 7 as per Average (3, 2, 2) * 3 as shown in the Figure 12. below. Note Risk Ratings are rounded to the nearest integer.

Attack Vector	Weakness Prevalence	Weakness Detectability	Technical Impact
3 - Easy	2 - Common	2 - Average	2 - Severe
Likelihood = Average (3, 2, 2) = 2.3			Risk Rating = 7

Figure 12: Sample Risk Rating

Vulnerability Values for the example above:

- Attack Vector: 3
- Weakness Prevalence: 2
- Weakness Detectability: 2
- Technical Impact: 3

The results of the Risk Rating are grouped into High, Medium and Low Ranges as follows:

- Low for Risk Factors less or equal than 3
- Medium for Risk Factors between 4 and 6
- High for Risk Factors higher than 7 to a maximum of 9

Figure 13. shows a sample list of Vulnerabilities and the corresponding Risk Ratings

[14] The Risk Rating calculation is also a simplified version of OWASP Risk Rating Methodology

Vulnerability	Attacker Profile	Attack Vector	Weakness Prevalence	Weakness Detectability	Technical Impact	RISK Low: <=3; Med: 4<=6; High:7<=9
Malicious actor compromises client; injects unauthorized list of Peer Nodes as endorsing peers in Endorsing Policy via chaincode	Hacker/ Criminal Groups	1	1	1	1	**1**
Malicious Actor compromises Fabric Admin Account; gains access to Fabric Admin Credentials of Orderer Org from Endorsement Policy	Hacker/ Criminal Groups	3	3	3	3	**9**
Compromised Fabric Admin Account used to instantiate untrusted chaincodes	Hacker/ Criminal Groups	3	3	3	3	**9**
Compromised Fabric Admin deletes all logs detailing malicious activity	Hacker/ Criminal Groups	3	3	3	3	**9**
Orgs Fabric Admin; Orderer Fabric Admins digital credentials compromised while being transmitted out of band to Client Org Fabric Admin	Hacker/ Criminal Groups	3	3	3	3	**9**
Service Provider for Fabric's Consensus Mechanism could manipulate RAFT'S Leader Election Process by modifying the randomness interval in turn affecting the Consistency and Availability of the Consensus (Ordering) Service	Insider Group	1	3	1	3	**4**
Service Provider for Fabric's Consensus Mechanism could manipulate RAFT's Leader Election Process by modifying the randomness interval in turn affecting the Consistency and Availability of the Consensus (Ordering) Service	Hacker/ Criminal Groups	2	3	1	3	**6**
Orderer Org Fabric Admin Account is compromised to gain unauthorized access to the Replication Logs of the Orderer Leader node running RAFT Consensus mechanism causing a Confidentiality Breach	Hacker/ Criminal Groups	2	3	2	3	**9**
Offchain Data Store hosting archived Replication Logs of the Orderer Leader node running RAFT Consensus mechanism is compromised causing Confidentiality Breach	Hacker/ Criminal Groups	2	3	2	3	**7**

Figure 13: Vulnerabilities and the Corresponding Risk Ratings

Step 6 - Group the Vulnerabilities by Cybersecurity Functional Areas

For the "Controls Checklist" deliverable to be Enterprise ready the identified vulnerabilities in the Fabric Network were organized into Cybersecurity Functional Areas that could seamlessly integrate with an enterprise's existing cybersecurity skill sets and capabilities allowing for clear lines of Roles, Responsibility and Accountability in turn making the tracking, managing and reporting of critical vulnerabilities easier.

These Cybersecurity Functional Areas map easily[15] to the "Domains or Families" of the various cybersecurity frameworks such as ISO 27001/27002 Ver 2013[16] or NIST 800-53 Rev4[17] respectively thus allowing for crosswalk against external frameworks to comply with Financial Industry Regulations.

Figure 14. identifies the Cybersecurity Functional Areas included in the report, while Figure 15. groups the identified vulnerabilities and their Risk Scores with the corresponding Cybersecurity Functional Areas.

Cybersecurity Functional Areas		
Application Security	Consensus Security	Data Protection and Cryptography
End Device & Server Security	Identity and Access Management	Incident Response
Peer Security	Systems Administration	

Figure 14: Cybersecurity Functional Areas

[15] The mapping this report Functional Areas to Cybersecurity frameworks is not included in this report
[16] ISO 27001/27002 Ver 2013 https://www.iso.org/standard/54533.html
[17] NIST 800-53 Rev4 https://csrc.nist.gov/publications/detail/sp/800-53/rev-4/final

Cybersecurity Functional Area	Vulnerability	Attacker Profile	Attack Vector	Weakness Prevalence	Weakness Detectability	Technical Impact	RISK Low: <=3; Med: 4<=6; High:7<=9
End Device and Server Security	Malicious actor compromises client; injects unauthorized list of Peer Nodes as endorsing peers in Endorsing Policy via chaincode	Hacker/ Criminal Groups	1	1	1	1	**1**
Identity and Access Management	Malicious Actor compromises Fabric Admin Account; gains access to Fabric Admin Credentials of Orderer Org from Endorsement Policy	Hacker/ Criminal Groups	3	3	3	3	**9**
Identity and Access Management	Compromised Fabric Admin Account used to instantiate untrusted chaincodes	Hacker/ Criminal Groups	3	3	3	3	**9**
Identity and Access Management	Compromised Fabric Admin deletes all logs detailing malicious activity	Hacker/ Criminal Groups	3	3	3	3	**9**
Identity and Access Management	Orgs Fabric Admin; Orderer Fabric Admins digital credentials compromised while being transmitted out of band to Client Org Fabric Admin	Hacker/ Criminal Groups	3	3	3	3	**9**
Consensus Security	Service Provider for Fabric's Consensus Mechanism could manipulate RAFT'S Leader Election Process by modifying the randomness interval in turn affecting the Consistency and Availability of the Consensus (Ordering) Service	Insider Group	1	3	1	3	**4**
Consensus Security	Service Provider for Fabric's Consensus Mechanism could manipulate RAFT's Leader Election Process by modifying the randomness interval in turn affecting the Consistency and Availability of the Consensus (Ordering) Service	Hacker/ Criminal Groups	2	3	1	3	**6**
Identity and Access Management	Orderer Org Fabric Admin Account is compromised to gain unauthorized access to the Replication Logs of the Orderer Leader node running RAFT Consensus mechanism causing a Confidentiality Breach	Hacker/ Criminal Groups	2	3	2	3	**9**
Data Protection and Cryptography	Offchain Data Store hosting archived Replication Logs of the Orderer Leader node running RAFT Consensus mechanism is compromised causing Confidentiality Breach	Hacker/ Criminal Groups	2	3	2	3	**7**

Figure 15: Vulnerabilities Grouped by Cybersecurity Functional Areas

Findings

In this section we report the findings of the Threat Evaluation and the Threat Analysis (as described in Section "Risk Identification Process") that were carried out on the Trade Finance Workflow at run time thus covering all three layers of Gartner's Blockchain Security Model viz, Business, Risk & IAM Process and Technology/IT Layers (Gartner, 2018).

Business Layer (Gartner's Blockchain Security Model)

Threat Evaluation to Trade Finance Business Logic Confidentiality and Privacy

- Hyperledger Fabric 2.0's Architecture was evaluated for Business Logic and Transaction/Payload Confidentiality and Privacy and was found to be natively Secure by Design and Default.
- Business Logic Confidentiality and Privacy: Fabric 2.0 allows for Smart Contracts to be installed on Client selected Peer nodes instead of all Peer nodes as is the case in non-Fabric Blockchains thus ensuring confidentiality and privacy of the Business Logic
- Transaction / Payload Confidentiality: Fabric 2.0 minimizes the exposure to highly confidential transactions via its "channels" feature that allows for entities with a "Need to Know" to be designated as members of a channel.
- Channels also enable "Separation of Ledgers" when transacting parties have a need to keep the transactions and the accompanying data confidential or simply want the entire interaction to be kept private. Fabric 2.0 also offers transacting parties the option of using Private Data Collections where proprietary or Personal Identifiable Information (PII) is separated from the rest of the transaction and exchanged only between authorized peers using Peer to Peer(P2P) Gossip Protocol while storing hashes of the Private Data on the Ledgers to eliminate any potential unauthorized exposure to Personal Identifiable Information (PII) or any other classified or proprietary information.
- End to End TLS between the Client and the Blockchain Nodes and between the Nodes ensures the "Data in Transit" is encrypted. Native Fabric Encryption is also available at the host level to encrypt "Data at Rest"

Note: Unauthorized exposure to Private Data stored in databases at the Peer Nodes or unauthorized access to Transaction logs stored off-chain or on-chain is an IT related Vulnerability and is not specific to Fabric Network Design.

Threat Evaluation to Trade Finance Business Logic Execution and Resiliency

Fabric 2.0's impact was specifically evaluated on the following vulnerabilities known to have been the primary cause for compromise of Business Logic Execution and Resiliency in non-Fabric Blockchain Networks. Fabric 2.0 architecture proved to be robust in preventing compromise of Trade Finance's Business Logic during run time.

1. Non-Deterministic Transactions within Smart Contracts: In Fabric 2.0 the impact from a nondeterministic transaction is only to the transaction on hand which may be rejected if sufficient number of peers cannot endorse it as per endorsement policy
2. Transaction Duplication:This vulnerability does not apply to Fabric 2.0 since duplicate transactions get filtered by the endorsing peers during the Validation Stage so they never get updated to World State
3. Timestamp Dependency: Fabric intentionally does not use the timestamp from the submitting application for anything, therefore there are no impacts with respect to Fabric processing.The timestamp is not intended to be a reflection of "network time", it can only be as trusted so far as the submitting application is trusted. Increasing block heights are the only trusted indication of time passage on a blockchain. If applications do reference the timestamp for additional informational context, they should consider it relative to block heights, e.g. are the timestamps increasing as block heights increase.
4. Transaction Ordering Dependency: This vulnerability applies to Fabric 2.0 where the leader of the ordering service orders transactions to favor specific organizations. Timestamps are critical to detecting the Transaction Reordering Attack. If an organization in the network is reliant on timing-critical contracts, it should track client application outgoing transactions. When the transaction is included in a block, reordering can be detected by comparing timestamps.
5. Third Party Trusted Services (Oracles) : Fabric 2.0 addresses the vulnerability arising from extending smart contracts by leveraging Oracles using three different architectural patterns to access them.
    f. In the first approach, the trusted party service has a membership in the blockchain network and utilizes a channel to make its data available to all members of the network (IBM,2019).
    g. In the second approach, all members of the blockchain network agree to trust and leverage a third-party service by making invocations to it from within the smart contracts. In this context, data inputs to the third-party service are used to correlate invocations that occur within the same transaction and, thus, guarantee determinism(IBM,2019).
    h. In the third and final approach, a claims issuer serves as the oracle by issuing verifiable credentials to entities, which are then corroborated during the execution of smart contracts. Client applications provide the necessary claims as inputs to smart contracts, and these then validate the authenticity of such claims by verifying the signatures (IBM,2019).

Risk/IAM Process & Technology/IT Layer (Gartner's Blockchain Security Model)

Threat Model Analysis of the Trade Finance WorkFlow at RunTime

The Threat Model identified **14 potential Threats** with a **Likelihood and Impact rating of HIGH**. The Attacker Profile for these Threats is the "Hacker/Criminal Groups" category. With the Trade Finance workflow having assets of high value to these attackers one can conclude these threats to be advanced and persistent[APT]. These APT are distributed across the Gartner Blockchain Security Model (Gartner, 2018) as follows:

- 50% of these Threats belong to the Risk and IAM Process Layer stemming from potential compromise of Fabric and Certificate Authority System's Administrative credentials with "Elevated Privileges"
- The remaining 50% belong to the Technology/IT Layer, directly related to unauthorized exposure of Personally Identifiable Information (PII) or Proprietary Transactions, Untrusted Fabric Client SDK or Smart Contract and compromised Peer Node

APTs belonging to Technology/IT Layer were also found to originate at the 'Client Trust Boundary' where the various Client participants of the Trade Finance workflow (viz., Importer & Importer's Bank, Exporter & Exporter's Bank, Carrier and the Regulator) interface with the Fabric Network.

The details of the 14 HIGH RISK APTs are shown in Figure 16. below:

Cybersecurity Functional Area	Vulnerability	Count	Attacker Profile	Likelihood & Impact
Identity and Access Management	Compromise of all types of Fabric System & Certificate Authority Administration Accounts	7	Hacker/Criminal Groups	High
Application Security	Untrusted Fabric Client SDK, Unvalidated Smart Contract	2	Hacker/Criminal Groups	High
Peer Security	Compromise of Peer Node	1	Hacker/Criminal Groups	High
Data Privacy & Cryptography	Unauthorized access to Confidential Transactions and or PII	4	Hacker/Criminal Groups	High

Figure 16: Details of the 14 High Risk APTs

Impact of Findings on Trade Finance Fabric Network

The Threat Model Analysis demonstrated that Decentralized Administration of the Fabric System & Certificate Authority, coupled with lack of robust Governance Policies to secure Administration Channels and Credentials from compromise could expand the attack surface considerably aiding the leap from "Establishing Foothold" into the Trade Finance Fabric Network to potentially compromising the entire Fabric Network and resulting in a **High Business Impact with Loss of Trade, Loss of Ownership and Loss of Trust between the Importer and the Exporter in the Trade Finance WorkFlow.**

Threat Mitigation Strategy Recommendations

The two main vulnerabilities can be mitigated as follows:

- Decentralized Fabric Administration Vulnerability: Selection of a Single Service Provider (preferably a neutral party) for administering the Fabric Network along with a Federated Certificate Authority in a Cloud environment will go a long way in reducing the attack surface due to decentralized Fabric Administration.
- Missing Governance Policies for securing Administration Channels & Credentials: Fabric Network stakeholders will need to enforce Governance Policies to:
  - secure Administrator Identity Credentials at all times (at rest, in transit and in use)
  - enforce "Separation of Duties" or "Layered Admin Privilege Role" restricting Admin's direct CLI access to the task on hand
  - restrict Admin Logins via a selected set of Standardized Tools example: Bastion Hosts, Out of Band/Dedicated Channels, Network Isolation etc)

For the High/Medium/Low Risk APT a risk based Mitigation Strategy is recommended and is as follows:

- **High Risk Threats:** Mitigation Controls are required to follow "Defense in Depth Strategy" spreading over Audit, Forensic, Detective & Preventive Control Categories. Minimum of 3 Control Categories need to be covered with both Audit and Detective being among them. This enables the defenders to buy time for an effective incident response while a High Risk attack is underway. Figure 17. shows a sample "Defense in Depth" Strategy for a High Risk Threat
- **Medium Risk Threats:** Mitigation Controls are recommended to follow " Defense in Depth Strategy" spreading over Audit, Forensic, Detective & Preventive Control Categories. Minimum of 2 Control Categories need to be covered and Detective being one among them. Figure 18. shows a sample "Defense in Depth" Strategy for a Medium Risk Threat
- **Low Risks Threats:** Mitigation Controls are required to have Forensic Controls.

Cybersecurity Functional Area	Vulnerability	RISK Low: <=3; Med: 4<=6; High:7<=9	Audit	Forensic	Detective	Preventive
Identity & Access Management	Compromised Org Fabric Admin Account used to tamper with Endorsement Policy	9	Audit the following: Review process for Compliance with "Digital Rights Management" for file containing Endorsement Policy Review process for compliance with "Split Access" to file containing Endorsement Policy Review process for violation alerts of the above	Establish automated review process of alerts for: Violation of "Digital Rights Management" to File containing Endorsement Policy	Configure Alerts when "Digital Rights Management" for File containing Endorsement Policy are violated Send alerts to Fabric Admins and MSP Admins	Configure File containing Endorsement Policy using Digital Rights Management Configure "Split Access" to File containing Endorsement Policy (Require login by both Fabric Admin & MSP Admin for access to endorsement policy) Configure real-time detection of violation of "Split Access"

Figure 17: Sample "Defense in Depth" Mitigation Strategy for a High Risk Threat

Cybersecurity Functional Area	Vulnerability	RISK Low: <=3; Med: 4<=6; High:7<=9	Audit	Forensic	Detective	Preventive
Application Security	Chaincode Access Control Policy tampered to include unregistered User	4	Audit Evidence of "Split Access" to File containing Chaincode Access Policy Audit Evidence of Review Process for Alerts & Log detection regarding Chaincode Access Control Policy Audit Evidence of Review Process for Validation of Blockchain Network Users	Review logs for unauthorized updates to Chaincode Access Control Policy	Configure alerts when logs detect non-whitelisted accounts and hosts updating Chaincode Access Control Policy Restrict ownership for event processing (SOC/NOC runbook updates) to Business Owner	Whitelist Host machines and or Admin accounts that can create and or update Chaincode Access Control Policy Configure "Split Access" to File containing Chaincode Access Control Policy (Require login by both Fabric Admin & MSP Admin for access to File) Review Access Control policies on a periodic basis. Perform periodic reviews of Users and their access

Figure 18: Sample "Defense in Depth" Mitigation Strategy for a Medium Risk Threat

Incident Response Readiness Strategy Recommendations

A risk based incident response strategy as explained below, that goes hand in hand with the Threat Mitigation Strategy described in Section 7 above could go a long way in aiding defenders to mount a well-formed defensive response while an attack is underway. Accordingly:

- All identified vulnerabilities that are **High Risk** are required to be Incident Response Ready with a documented Incident Response Strategy.
- All identified vulnerabilities that are **Medium Risk** are required to have an Incident Tracking Process in place.
- For all identified vulnerabilities that are **Low Risk** a process for monitoring events that have the potential of turning into Incidents needs to be in place.

Cryptography Module Recommendations for Fabric 2.0 Permissioned Network

As any blockchain, Hyperledger Fabric 2.0 leverages cryptographic primitives at its core to operate. These cryptographic artifacts are used to sign transactions, create hashes of data like block headers and merkle trees. The field of cryptography is constantly evolving researching and developing new algorithms for artifacts like hash functions and digital signatures and what was considered secure once may be obsolete now (Vlad et al., 2017) (Kelly et al., 2018).

Organizations can constrain the types of algorithms, modes and parameters that can be used to perform certain actions to limit vulnerabilities.

For example the U.S. Federal Government[18] through the FISMA Act requires federal information systems to be assessed and authorized before they can operate with federal data. FIPS standards have been developed at NIST to establish among other things what cryptographic algorithms are approved for use by the Federal Government. NIST FIPS 140-2 requires that the implementation of cryptographic algorithms that are used to protect sensitive government information must be validated.

Fabric 2.0 supports several digital signatures and hashing algorithms like the ECDSA P256 curve and the SHA256 hashing function (both are approved by NIST FIPS standards).

These approved cryptographic algorithms need to be implemented in a crypto module that has been validated by NIST in order to be used in Federal information system processing sensitive unclassified information. An example of such a module is Google's BoringCrypto that is FIPS 140-2 (Certificate #3318[19]). This module needs to be configured in 140-2 mode when installed and only validated and

[18] Local Government cryptographic rules/conditions supersede all recommendations made in this document

[19] See https://csrc.nist.gov/projects/cryptographic-module-validation-program/certificate/3318

allowed algorithms should be used. The module can then replace the current implementation of the cryptographic functions in Fabric 2.0.

Glossary

Anchor Peer	A peer node on a channel that all other peers can discover and communicate with. Each Member on a channel has an anchor peer (or multiple anchor peers to prevent single point of failure), allowing for peers belonging to different Members to discover all existing peers on a channel.
(Risk) Likelihood	This refers to the likelihood or frequency of a threat event occurring. The infrequent occurrence of an event represents a lower risk to the company. Conversely, a significant history of threat events occurring in this or similar environments would indicate a higher risk.
Blockchain Network	A blockchain network is a technical infrastructure that provides ledger and smart contract (chaincode) services to applications.
ChainCode	Chaincode is a program that implements a prescribed interface to handle the business logic agreed to by members of the network. It initializes and manages the state of the ledger through transactions. The program is written in Go, Node.JS, or Java and is run on a docker container. It can also be considered a smart contract.
Channel	A mechanism that allows a specific set of peers and applications to communicate with each other within a blockchain network. It permits data isolation and confidentiality as only those allowed to participate in the channel can see the data. In Hyperledger Fabric a channel also refers to a channel specific ledger where only specific peers are allowed to interact with it.
Consensus	This refers to a Majority of participants of a network agreeing on the validity of a transaction. In the context of Hyperledger Fabric Consensus is the process by which a network of nodes provides a guaranteed ordering of transactions and validates the block of transactions.
Consensus Security	An application of security protocols, such as encryption and hashing, to protect data integrity and safeguard Consensus Algorithm against proof of work, proof of Stake etc.

Control	A process, check, or barrier implemented to mitigate risk or to detect realization of a threat. Controls are described as their action regarding timing within a security event **Preventive:** acting before a security event takes place, and with the purpose of preventing the security event from manifesting **Detective:** controls that enable the detection and characterization of an event already in progress **Corrective:** measures designed for limiting the extent of damage and restoring the application to its baseline performance and configuration **Forensic:** measures put in place to support post event investigation. Also includes any controls designed to support the integrity of the investigative process and its underlying data (event data and system configuration)
CVE	Short for Common Vulnerabilities and Exposures, is a list of publicly disclosed computer security flaws.
DLT	Distributed Ledger Technology (DLT) is technological infrastructure and protocols that operate a decentralized network allowing simultaneous secure access, validation and record updating using cryptographic signatures and with no central authority.
Endorsement policy	Defines the peer nodes on a channel that must execute transactions attached to a specific chaincode application and the required combination of responses. For example, a policy can require a minimum number of peers to endorse it.
Endorsing Peer	A peer (node) with a specific role, in the context of a specific chaincode, for endorsing a transaction.
Fabric Admin	A user with "Elevated Privileges" within an organization that administers the Fabric Network.
Fabric Client	The means by which an application interacts with a blockchain network, usually a Peer node.
Follower	Nodes which will replicate entries that are sent to them by the Leader.
HSM	Short for Hardware Security Module. A component, usually a piece of infrastructure, that provides cryptographic key management services.
Hyperledger	This is an umbrella project of open source blockchains and community focused on developing a suite of stable frameworks, tools and libraries for enterprise-grade blockchain (DLT) deployments.
Hyperledger Fabric	Distributed ledger software that can be used as a foundation for developing blockchain based solutions or applications.
Incident Response	A process of addressing the situation of an event that could lead to loss of, or disruption of services of a business.

Indicator of Compromise (IoC)	Data elements, usually found in system log entries or files, that identify potentially malicious activity on a system or network.
Leader	The node responsible for ingesting new log entries, passing them to follower nodes, and managing the entries which are committed to the ledger.
MSP	Short for Membership Service Provider. A component of the network that validates credentials of clients and peers so these can participate in the Hyperledger Fabric network. Credentials are used to authenticate transactions. It obscures the cryptographic mechanisms for issuing and validating certificates as well as user authentication. A Fabric network can have more than one MSP.
Orderer	A peer (node) participating in Ordering Services (see Ordering Service)
Ordering Service	A collective of nodes that orders transactions into a block and then distributes blocks to connected peers for validation and commitment. The service is independent of the peers process. Transactions are ordered on a first come first serve basis.
Org	Short for Organization. Refers to the businesses with membership in a permissioned blockchain network. Also known as members.
Org MSP Admin	User with "Elevated Privileges" within an Organization that administers the Membership Service of the Fabric Network
Peer	A node in a Blockchain Network. Peers are associated with Orgs (short for Organizations, as in participating Organizations). In Hyperledger Fabric, a peer runs chaincode containers and performs read/write operations on the ledger. Peers are owned and maintained by members.
Policy	Expressions which are used to restrict access to resources on the blockchain network. Examples include who can read or write to a channel or who can use a chaincode API.
RAFT	Consensus algorithm used by Hyperledger Fabric, which uses a "leader and follower" model where a leader node is elected and the decisions of the leader are filtered down to the followers.
Risk	The probability or threat of damage, injury, liability, loss, or any other negative occurrence caused by external or internal vulnerabilities may be avoided through preemptive control actions. A reduction in either the threat or vulnerability reduces the risk.
Risk Rating	A process of assessing risk activities and classifying them as: 1. Low 2. Medium 3. High The rating is classified based on a combination of estimations for the likelihood of an event and its impact.

SIEM	Security Incident and Event Management. Usually in reference to a platform or system.
Smart Contract	Code invoked by a client application that is external to the blockchain network which manages access and modifications.
State	Also Ledger State. The aggregate state of assets in the network as per all completed transactions within a channel.
StateDB	Fabric component storing key-value pairs included in transactions. This is where World state data is stored.
VSCC	Validation System Chain Code. Used to validate the transaction against the endorsement policy. The transaction is marked invalid if it does not satisfy the policy.
Vulnerability	A flaw or weakness in the procedures, hardware, software, or internal controls that could be triggered accidentally or intentionally exploited to cause harm in the DLT.
Web API	An application programming interface for a web server or web browser.
World State	Fabric component which represents the latest values for all keys in the chain transaction log. The world state will change every time the value of the key changes.

Bibliography

Adhav, P. (2020, August 25). *System Chaincodes in Hyperledger Fabric — VSCC, ESCC, LSCC, ESCC, QSCC*. Medium. https://medium.com/coinmonks/system-chaincodes-in-hyperledger-fabric-vscc-escc-lscc-cscc-a48db4d24dc3

ALBREIKI, H., HABIB UR REHMAN, M., SALAH, K., & SVETINOVIC, D. (2020). Trustworthy Blockchain Oracles: Review,Comparison, and Open ResearchChallenges. *IEEE Access*. 10.1109/ACCESS.2020.2992698

Allianz. (2021). Allianz Risk Barometer. https://www.agcs.allianz.com/news-and-insights/reports/allianz-risk-barometer.html

Androulaki, E., Barger, A., Bortnikov, V., Christian Cachin, Christidis, K., De Caro, A., Enyeart, D., Ferris, C., Laventman, G., Manevich, Y., Muralidharan, S., Murthy, C., Nguyen, B., Sethi, M., SIngh, G., Smith, K., Sorniotti, A., Stathakopoulou, C., VukolićJuvoli, M., ... Yellick, J. (2018). Hyperledger fabric: a distributed operating system for permissioned blockchains. *Proceedings of the Thirteenth EuroSys Conference*, 1-15. 10.1145/3190508.3190538

Angelis, S. D., Zanfino, G., Aniello, L., Lombardi, F., & Sassone, V. (2019). Blockchain and Cybersecurity: A Taxonomic Approach. *University of Southampton*. https://www.eublockchainforum.eu/sites/default/files/research-paper/wrks-main_1.pdf

Baset, S., Desrosiers, L., Gaur, N., Novotny, P., Ramakrishna, V., & O'Dowd, A. (2018). *Hands-On Blockchain with Hyperledger* (ISBN: 9781788994521 ed.). Packt Publishing. https://www.packtpub.com/product/hands-on-blockchain-with-hyperledger/9781788994521

Birge, C., Craig, A., Dadoun, D., Glaros, M., Cristin, C., & Chamber of Digital Commerce. (2018). Advancing Blockchain Cybersecurity: Technical and Policy Considerations for the Financial Services Industry. https://query.prod.cms.rt.microsoft.com/cms/api/am/binary/RE1TH5G

Carter, H. (2019). Journey to Blockchain: A Non-Technologist's Guide to the Internet of Value. *BRI*.

Chia, V., Hartel, P., Hum, Q., Ma, S., Piliouras, G., Reijsbergen, D., Staalduinen, M. v., & Szalachowski, P. (2019). Rethinking Blockchain Security: Position Paper. *ArXiv:1806.04358*. http://arxiv.org/abs/1806.04358.

Copigneaux, B., & European Parliament. (2020). Blockchain for Supply Chains and International Trade: Report on Key Features, Impacts and Policy Options Study. *European Parliamentary Research Service, and Scientific Foresight Unit*. http://www.europarl.europa.eu/RegData/etudes/STUD/2020/641544/EPRS_STU(2020)641544_EN.pdf.

Crowdstrike. (2021). 2021 Global Threat Report. https://www.crowdstrike.com/resources/reports/global-threat-report/

Dabholkar, A., & Saraswat, V. (2019). Ripping the Fabric: Attacks and Mitigations on Hyperledger Fabric. *Applications and Techniques in Information Security, 10th International Conference, ATIS 2019, Thanjavur, India, November 22–24, 2019, Proceedings*, (pp.300-311). 10.1007/978-981-15-0871-4_24

Dika, A., & Nowostawski, M. (2018). Security Vulnerabilities in Ethereum Smart Contracts. *Conference: 2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*. 10.1109/Cybermatics_2018.2018.00182

Dingman, W., Cohen, A., Ferrara, N., Lynch, A., Jasinski, P., Black, P. E., & Deng, L. (2019). Defects and Vulnerabilities in Smart Contracts, a Classification using the NIST Bugs Framework. *Atlantis Press*. 10.2991/ijndc.k.190710.003

European Commission. (2019). Legal and Regulatory Framework of Blockchains and Smart Contracts. *The European Union Blockchain Observatory and Forum*. https://www.eublockchainforum.eu/sites/default/files/reports/report_legal_v1.0.pdf

European Commission. (2020). 2018-2020 CONCLUSIONS AND REFLECTIONS. E*U BLOCKCHAIN OBSERVATORY AND FORUM*. https://www.eublockchainforum.eu/sites/default/files/reports/report_conclusion_book_v1.0.pdf

Gartner. (2017). Blockchain Technology: How Security Relates to Use Cases. (ID:G00317396).

Gartner. (2018). Evaluating the Security risks to Blockchain Ecosystems. (ID:G003247104).

Gartner. (2020). Blockchain Trials Show Business Executives Drive Focused Solutions to Production. (ID G00733890).

Global Blockchain Business Council,. (2020). Chain Reaction: Blockchain Enters the Mainstream. *Annual Report 2020*. https://www.lw.com/thoughtLeadership/gbbc-report-blockchain-enters-mainstream

Hoffman, C., Ignatova, P., Fong, F., Bischof, D., & Defeo, J. (2020). Blockchain & DLT in Trade: A reality check. *WTO*. https://www.wto.org/english/res_e/booksp_e/blockchainanddlt_e.pdf

Homoliak, I., Venugopalan, S., Hum, Q., & Szalachowski, P. (n.d.). A Security Reference Architecture for Blockchains. *ArXiv:1904.06898*. Retrieved 2019, from http://arxiv.org/abs/1904.06898

IBM. (2020). Advancing global trade with blockchain. *Institute for Business Value*. https://www.ibm.com/downloads/cas/WVDE0MXG

International Finance Corporation. (2019). Blockchain. Opportunities for Private Enterprises in Emerging Markets. *World Bank Group Report*. https://www.ifc.org/wps/wcm/connect/2106d1c6-5361-41cd-86c2-f7d16c510e9f/201901-IFC-EMCompass-Blockchain-Report.pdf

Jones, J. (2019). *Understanding Cyber Risk Quantification*. FAIR Institute. https://www.fairinstitute.org/blog/download-understanding-cyber-risk-quantification-the-buyers-guide-by-jack-jones

Kelly, J., Lauer, M., Prinster, R., & Zhang, S. (2018). Investigation of Blockchain Network Security Exploration of Consensus Mechanisms and Quantum Vulnerabilities. https://courses.csail.mit.edu/6.857/2018/project/Kelly-Laurer-Prinster-Zhang-BlockchainNetSec.pdf

Koens, T. (2019, October 29). *Atomic Swaps for Distributed Ledgers*. Medium. https://medium.com/ing-blog/atomic-swaps-for-distributed-ledgers-cacfb7e1d90

Perez, D., & Livshits, B. (2020). Smart Contract Vulnerabilities:Vulnerable Does Not Imply Exploited. https://arxiv.org/pdf/1902.06710.pdf

Praitheeshan, P., Pan, L., Yu, J., Liu, J., & Doss, R. (2020). Security Analysis Methods on Ethereum SmartContract Vulnerabilities — A Survey. https://arxiv.org/pdf/1908.08605.pdf

Putz, B., & Pernul, G. (2020). Detecting Blockchain Security Threats. *2020 IEEE International Conference on Blockchain (Blockchain)*, 313–320. https://doi.org/10.1109/Blockchain50366.2020.00046

Rilee, K. (2018, February 14). *Understanding Hyperledger Fabric — Endorsing Transactions*. Medium. https://medium.com/kokster/hyperledger-fabric-endorsing-transactions-3c1b7251a709

Shostack, A. (2014). *Threat Modeling: Designing for Security* (1st ed.). Wiley Publishing.

Verizon. (2020). 2020 Data Breach Investigations Report. https://enterprise.verizon.com/resources/reports/2020-data-breach-investigations-report.pdf

Vlad, G., Gorbunov, S., Mosca, M., & Munson, B. (2017). QUANTUM-PROOFING THE BLOCKCHAIN. *Blockchain Research Institute*. https://evolutionq.com/quantum-safe-publications/mosca_quantum-proofing-the-blockchain_blockchain-research-institute.pdf