Hyperledger Fabric Gateway Support for Fabric Private Chaincode

By :- Deepjyoti Sarmah
Date :- 19th December 2023
Presented at :- FPC Community Meeting

Fabric Private Chaincode

- Fabric Private Chaincode (FPC) is an extension of Hyperledger Fabric, enhancing chaincode confidentiality via execution in a Trusted Execution Environment (TEE).

- Transactional data is safeguarded while in use, in transit, and stored on the ledger, allowing Fabric peers to only handle encrypted data.

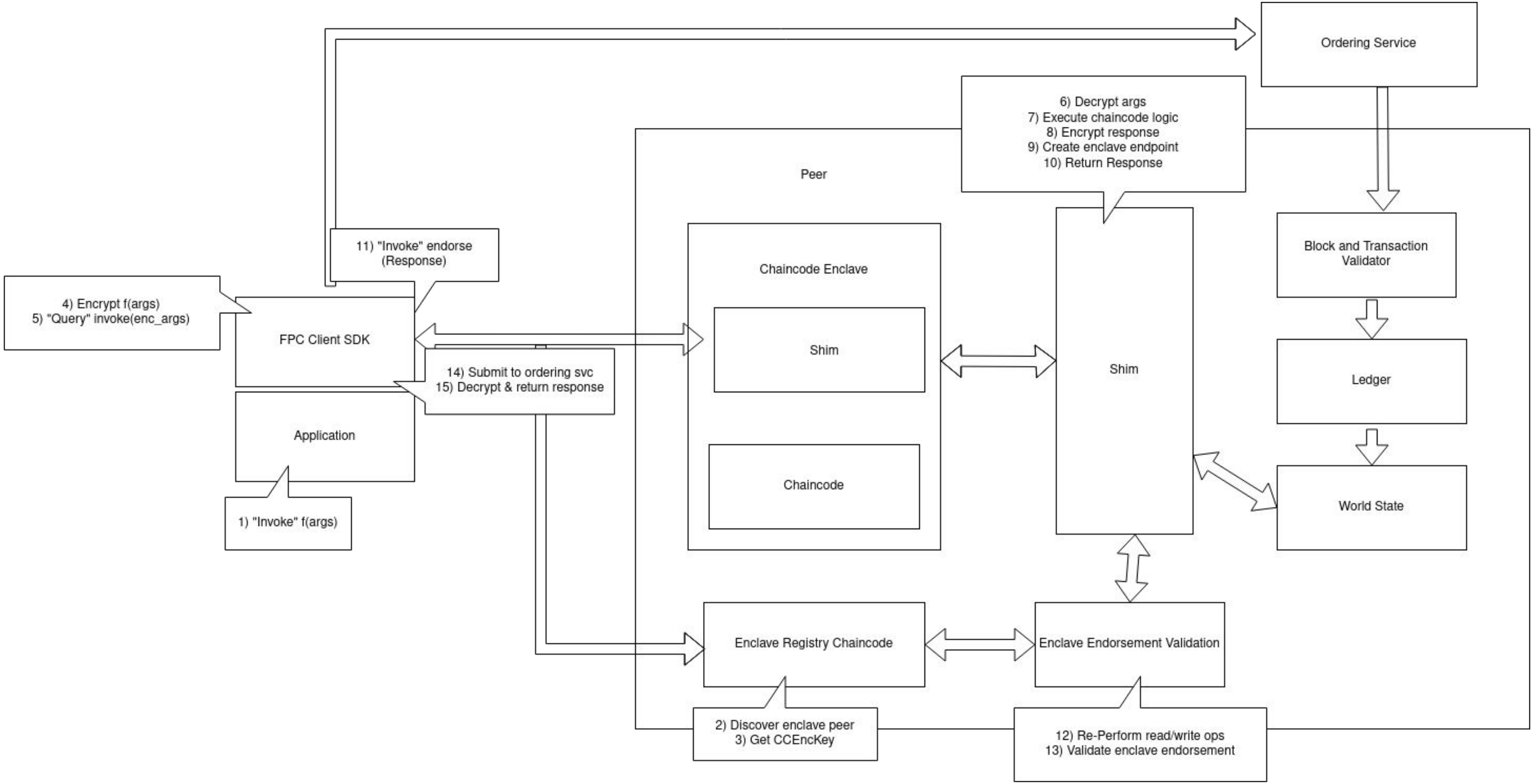
- FPC allows parties to establish trust in an FPC Chaincode through a hardware-based remote attestation, verifying the protection of the chaincode and its data by a genuine TEE.

- FPC supports secure implementation of use cases needing privacy, including privacy-preserving analytics on sensitive data and confidential supply chain agreements.

- FPC deployment follows the standard Fabric process with an added step of creating and registering a chaincode enclave.

- The FPC Client SDK is currently built on the deprecated Fabric Client SDK in Go. This results in the lack of support for the new Fabric Gateway API, which simplifies the transaction flow for clients.

Fabric Private Chaincode Architecture



Ordering Service
6) Decrypt args
7) Execute chaincode logic
8) Encrypt response
9) Create enclave endpoint
10) Return Response
Peer
11) "Invoke" endorse
(Response)
Chaincode Enclave
Shim
Chaincode
Shim
Block and Transaction Validator
Ledger
World State
4) Encrypt f(args)
5) "Query" invoke(enc_args)
FPC Client SDK
14) Submit to ordering svc
15) Decrypt & return response
Application
1) "Invoke" f(args)
Enclave Registry Chaincode
Enclave Endorsement Validation
2) Discover enclave peer
3) Get CCEncKey
12) Re-Perform read/write ops
13) Validate enclave endorsement

Current Problem

- The existing FPC client SDK hides the complex transaction flow but is based on the deprecated Fabric Client SDK. Switching to the Gateway Client would make the client application more lightweight.

- The Gateway Client performs most client-specific transaction steps transparently, reducing the client's burden of handling these steps.

- However, there are cases where specific peers need to be targeted, such as during chaincode installation. In these cases, using gRPC client stubs directly could be a suitable approach.

Fabric Gateway

- Fabric Gateway simplifies the transaction flow for clients by moving high-level programming logic to its own process, reducing client SDK complexity and network administration.

- It can run as a standalone process or embedded with a peer, interacting with the network on behalf of the client, eliminating the need for direct client-peer connections.

- The Gateway handles scaling and load balancing for client applications, aiming to be stateless and allowing clients to be routed through an appropriate load balancer.

How Fabric Gateway works ?

- Fabric Gateway simplifies transaction submission by handling transaction steps like evaluation, endorsement, submission, and commit status on behalf of the client .

- It operates as a client process associated with an organization, requiring an identity to interact with the discovery service and register for events. It can run as a standalone process or embedded within a peer, exposing its functionality to clients via a gRPC interface .

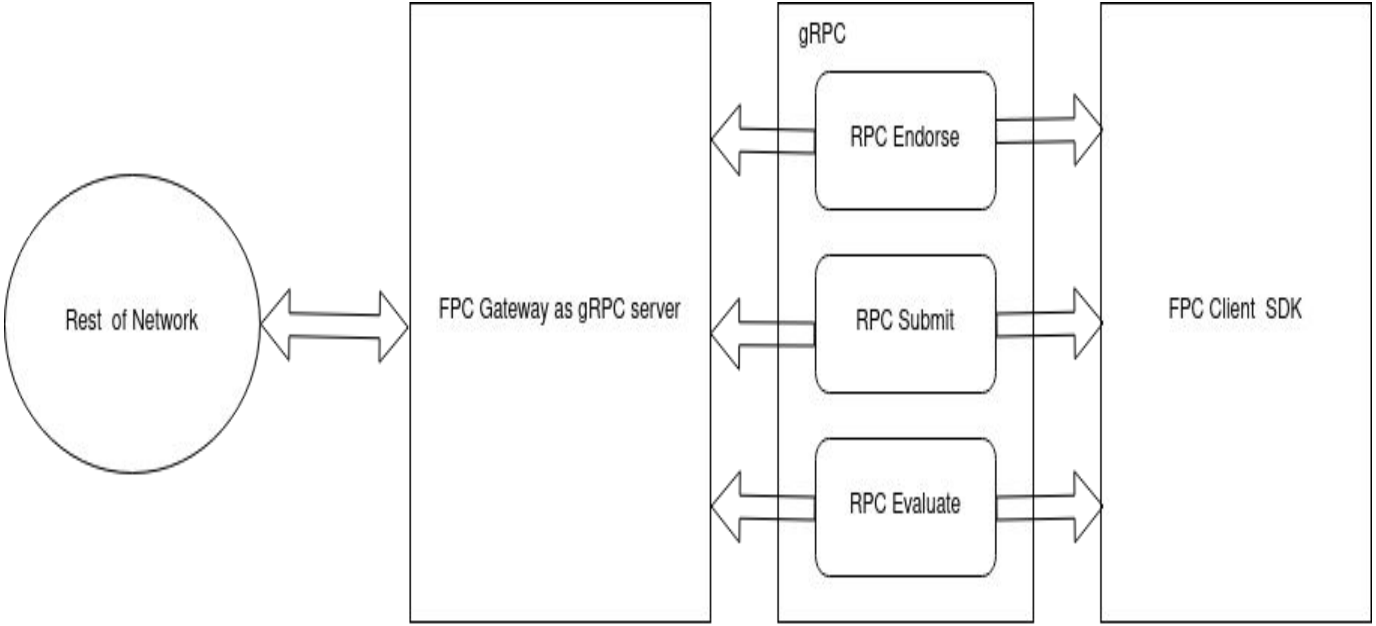
- The Gateway uses a discovery service to keep information synchronized across peers, including the endorsement policies and which peers have the chaincode installed.

- It interacts with the discovery service to determine which organizations are required to endorse a transaction proposal.

- The Gateway manages node connectivity, retry attempts, errors, and timeouts.

- It also provides a simplified API for client applications to receive chaincode events, allowing them to respond to events that are emitted by a chaincode function when a transaction is committed to the ledger.

Fabric Gateway Architecture



Gateway Services

- RPC Endorse: Kickstarts approval for a proposed transaction by crafting a Proposed Transaction with a Signed Proposal, triggering the Endorse service on the Gateway, and getting back a Prepared Transaction that holds an Envelope.

- RPC Submit: Wraps up sending a transaction by signing the Envelope hash with the user's private key, using the Submit service on the Gateway, and getting responses (success/errors) via a stream after sending to the ordering service.

- RPC Evaluate: Fetches the result of a chaincode function without starting a transaction by using the Evaluate service, sending a Signed Proposal, and getting the output of the chaincode function from the Gateway.

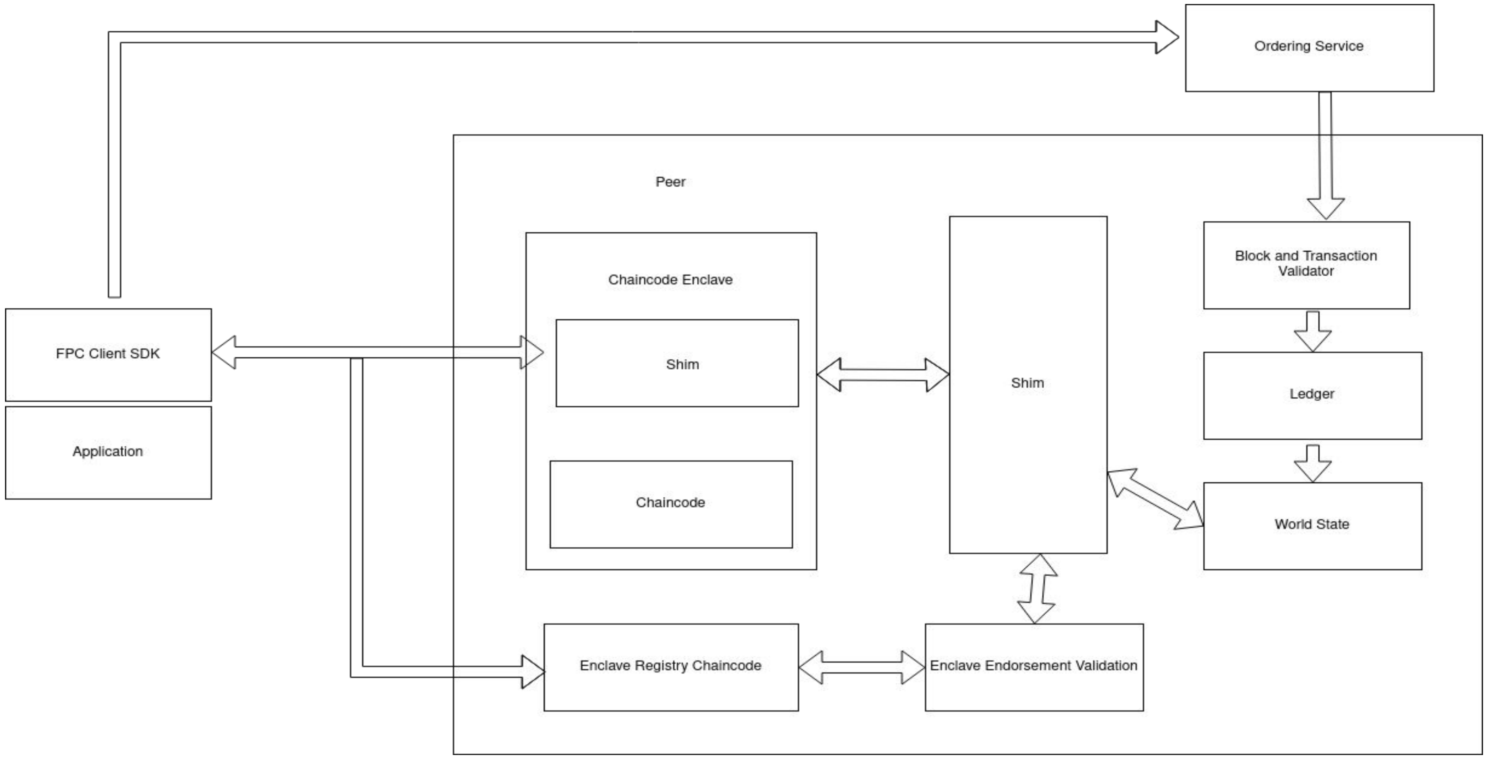
Link to Gateway Documentations and Repository

- Hyperledger Fabric Gateway
- Fabric Gateway Wiki
- Fabric Gateway Go File

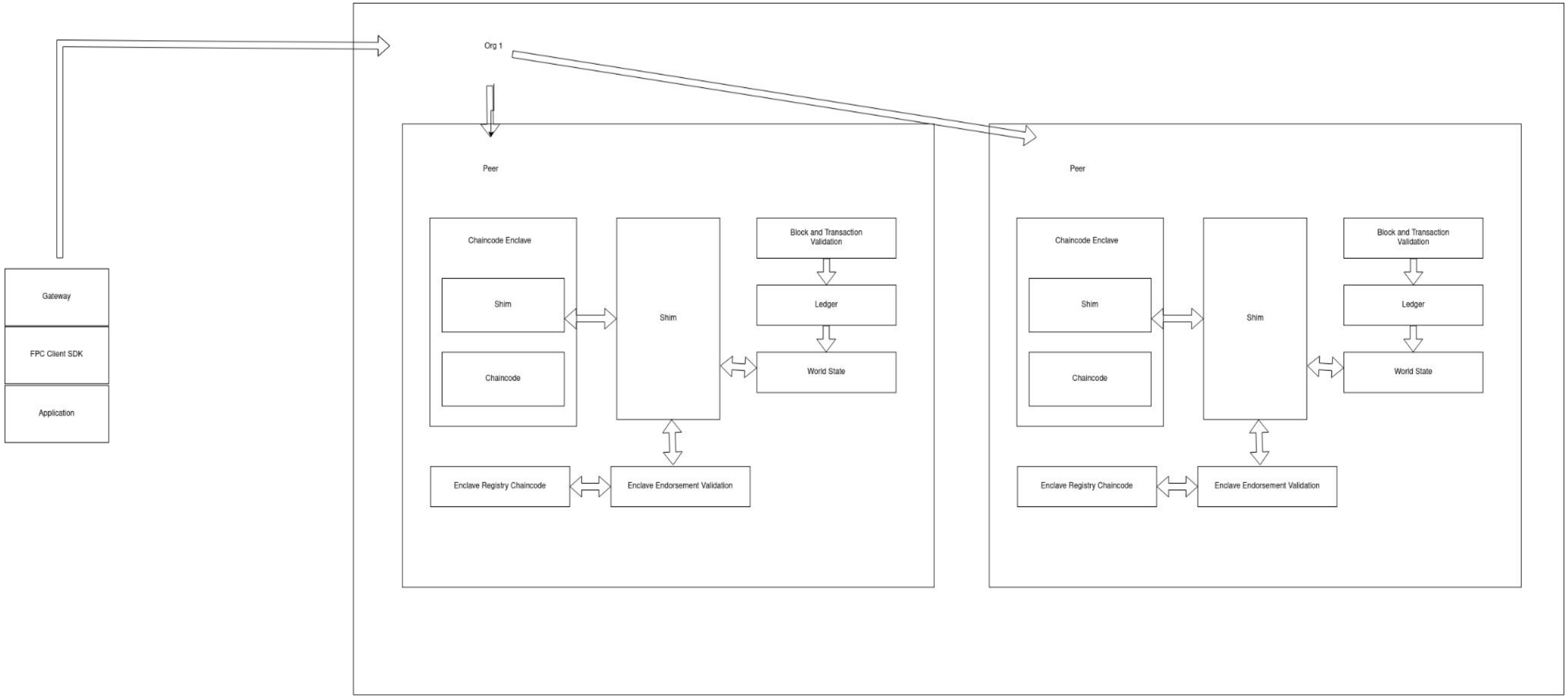
Potential Solutions

- Solution 1:
  - The Fabric Gateway API does not support directly targeting a specific peer with invocations.
  - It manages the intricacies of sending transaction proposals to endorsing peers based on the chaincode endorsement policy.
  - It has the option to indicate specific organizations for endorsement, indirectly directing the request towards the peers associated with those organizations.

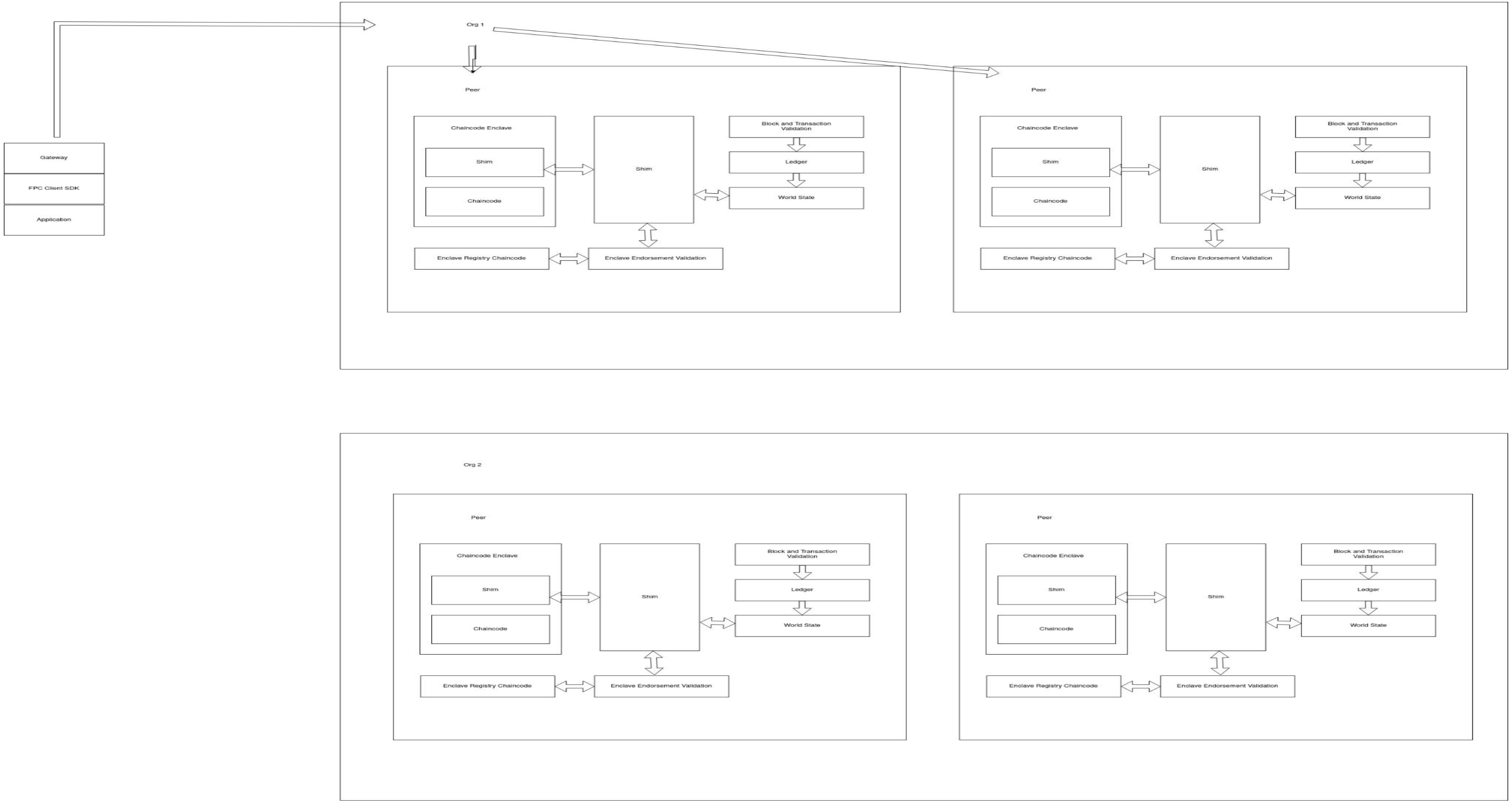
Current FPC Architecture



Solution Architecture (single organisation, multiple peer)


Org 1
Gateway
FPC Client SDK
Application
Peer
Chaincode Enclave
Shim
Chaincode
Shim
Block and Transaction Validation
Ledger
World State
Enclave Registry Chaincode
Enclave Endorsement Validation
Peer
Chaincode Enclave
Shim
Chaincode
Shim
Block and Transaction Validation
Ledger
World State
Enclave Registry Chaincode
Enclave Endorsement Validation

Solution Architecture (multiple organisation, multiple peer):


Gateway
FPC Client SDK
Application
Org 1
Peer
Chaincode Enclave
Shim
Chaincode
Shim
Block and Transaction Validation
Ledger
World State
Enclave Registry Chaincode
Enclave Endorsement Validation
Peer
Chaincode Enclave
Shim
Chaincode
Shim
Block and Transaction Validation
Ledger
World State
Enclave Registry Chaincode
Enclave Endorsement Validation
Org 2
Peer
Chaincode Enclave
Shim
Chaincode
Shim
Block and Transaction Validation
Ledger
World State
Enclave Registry Chaincode
Enclave Endorsement Validation
Peer
Chaincode Enclave
Shim
Chaincode
Shim
Block and Transaction Validation
Ledger
World State
Enclave Registry Chaincode
Enclave Endorsement Validation

Problems in Solution 1

- Fabric Gateway client API allows users to specify endorsement organizations for transactions.

- The API does not directly support targeting specific peers. The Gateway autonomously handles proposal dispatch to endorsing peers based on the chaincode endorsement policy.

- Specific instances may require explicit targeting of particular peers. Users, in such cases, can direct transaction proposals to specific predefined peers within specified organizations to meet unique requirements.

Solution 2

- Fabric-Admin-SDK Approach:
    - Fabric-Admin-SDK leverages the Fabric Gateway client API for transaction evaluation and submission.
    - For tasks like chaincode install, it directly uses gRPC client stubs when specific peers need to be targeted.
    - This approach can be extended to other scenarios requiring the explicit targeting of specific peers in similar fashion.

- Transaction Proposal Handling:
    - For these specific cases, a transaction proposal is sent to the peer but not submitted to the orderer.

    - If submission to the orderer is needed, using the Gateway might be beneficial as it handles failover between orderers and doesn't require the client to manage.

- Gateway gRPC Service Stubs:
    - The Fabric Gateway client API doesn't expose the capability to submit previously endorsed transactions.

    - However, the Gateway gRPC service stubs could be used similarly to how the fabric-admin-sdk code uses the peer Endorser service .

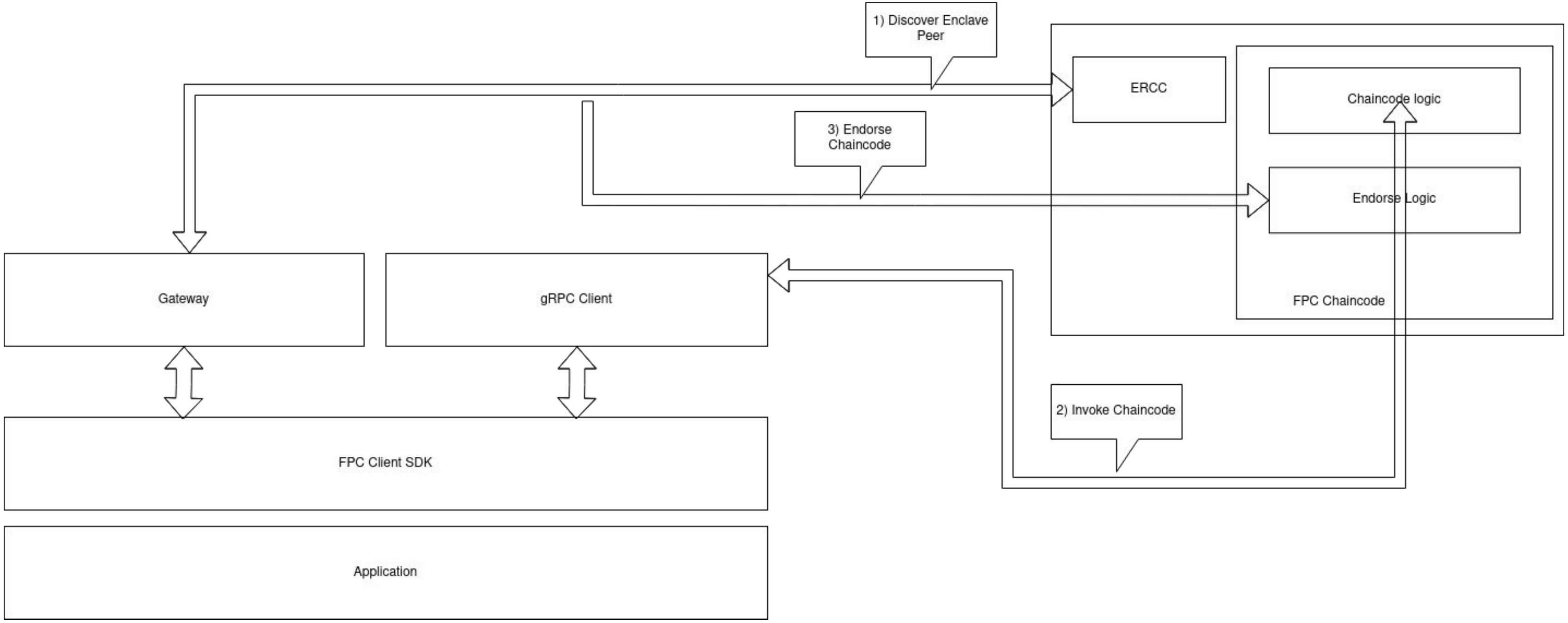
Enclave Integration Steps for FPC Chaincode

- The main function needs to be adapted to connect to the Fabric Gateway and call the enclave chaincode to get the enclave peer endpoint.

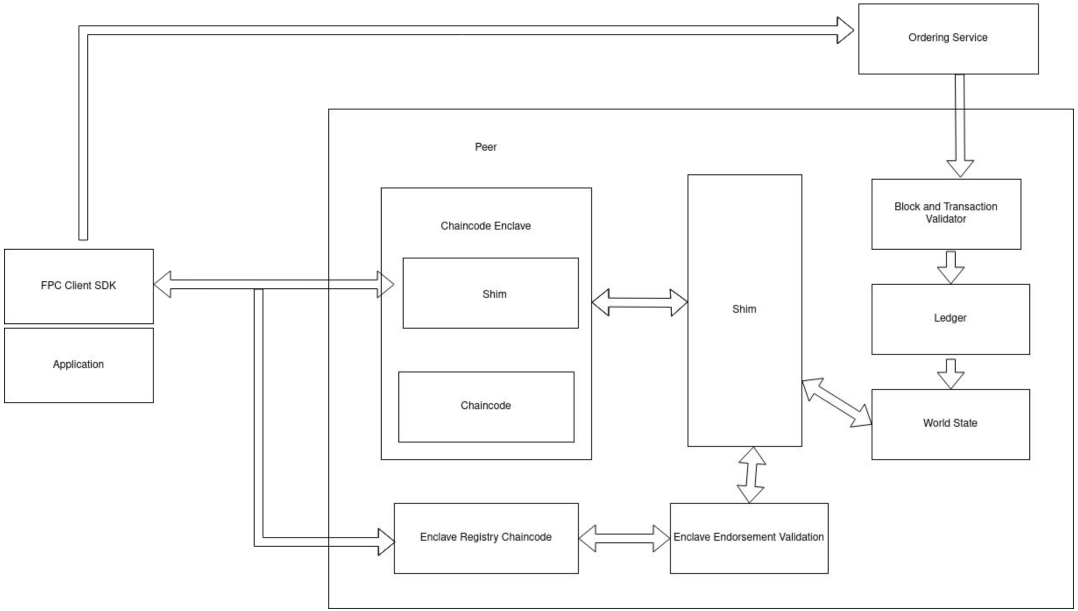
- A new gRPC connection is needed to establish a connection to the enclave peer. This can be done by copying the newGrpcConnection function.

- The FPC chaincode can be invoked via the direct gRPC connection established in previous step. This can be done following the example of the Fabric Admin SDK.

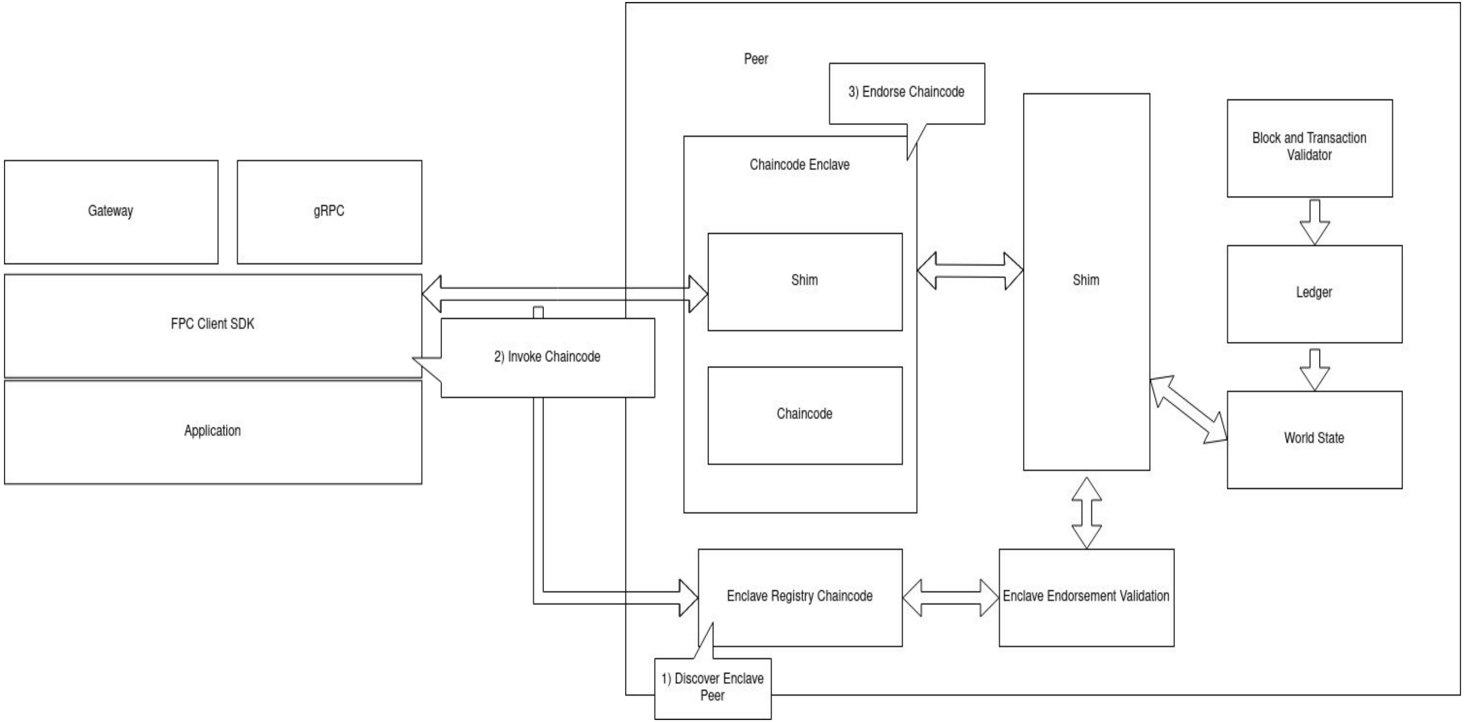
Solution Architecture



FPC Architecture without Gateway API



FPC Architecture Excluding Gateway API



Problems in current Solution 2

- The Fabric Gateway API call to the enclave chaincode, attempting to obtain the enclave peer endpoint, is unsuccessful or,

- The issue lies in the unsuccessful creation of the certificate pool and the subsequent failure to establish transport credentials using the TLS root certificate, preventing secure communication.

Implementation Resources

- assetTransfer.go file
- gateway.go file
- contract.go file
- Marcus Brandenburger's fpc-sample.go file
- Deepjyoti-Sarmah fpc-test.go file

Summary and Future Work

- Explore integrating the Fabric Gateway API with Fabric Chaincode for lightweight client applications. The API operates independently or embedded in a peer, interacting with the network on behalf of clients and eliminating the need for direct client-peer connections.

- Future Work :
    - Identify and address existing issues in the solution.
    - Conduct comprehensive testing using unit-tests and integration-tests.
    - Develop sample applications showcasing the practical use of the new FPC Client SDK.