



ESCUELA
POLITÉCNICA
NACIONAL



FACULTAD DE INGENIERIA EN SISTEMAS

**DISEÑO Y PROTOTIPO DE UN SISTEMA DE AUDITORIA Y CONTROL
ELECTORAL PARA LOS PROCESO DE ESCRUTINIOS Y
PUBLICACIONES DE RESULTADOS BASADOS EN DLT**

DISEÑO E IMPLEMENTACION DEL PROTOTIPO DEL SISTEMA DE SEGURIDAD

Autor: Emerson Ceracapa

Tutor: Dr. Enrique Mafla

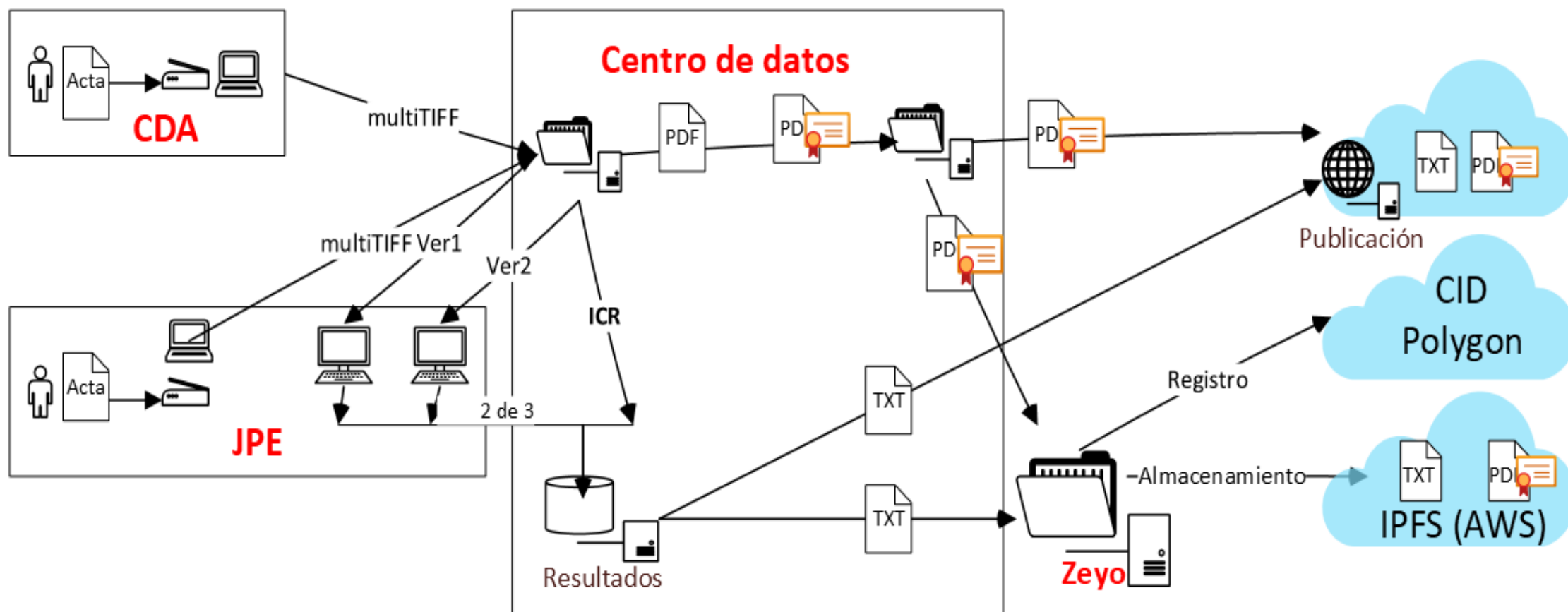
Objetivo General

Garantizar la identificación y autenticación segura y confiable de los diferentes componentes de la red blockchain, así como la integridad y confidencialidad de las comunicaciones entre dichos elementos. Esto se logrará mediante el diseño de una robusta infraestructura de llave pública (PKI).

Objetivos Específicos

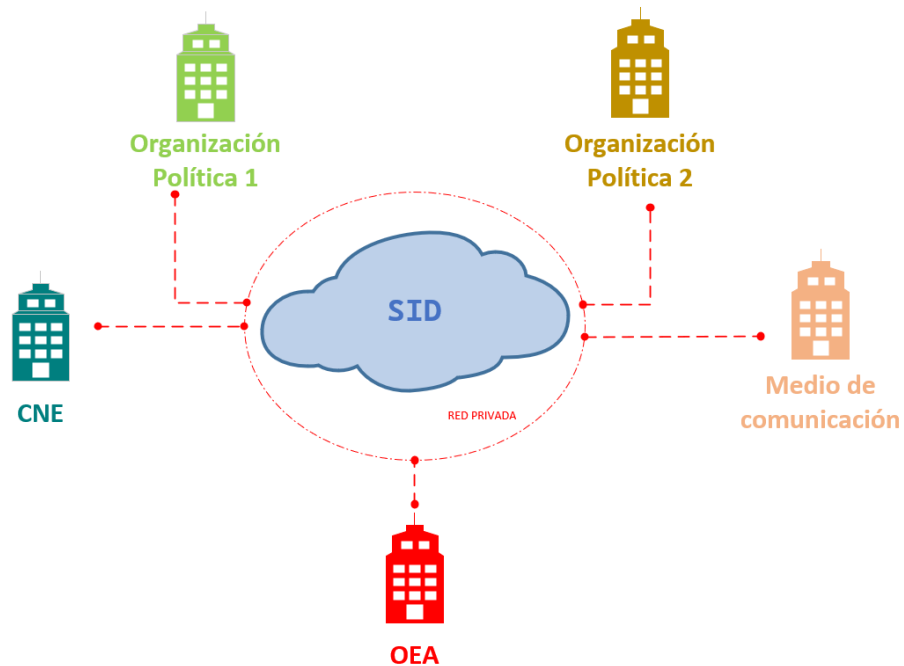
1. Analizar los problemas de seguridad y transparencia en el proceso actual de escrutinios y publicación de resultados.
2. Revisar las soluciones existentes para una PKI.
3. Diseñar un sistema de seguridad basado en una PKI.
4. Implementar un prototipo PKI para validar el diseño.

Análisis de la Situación Actual



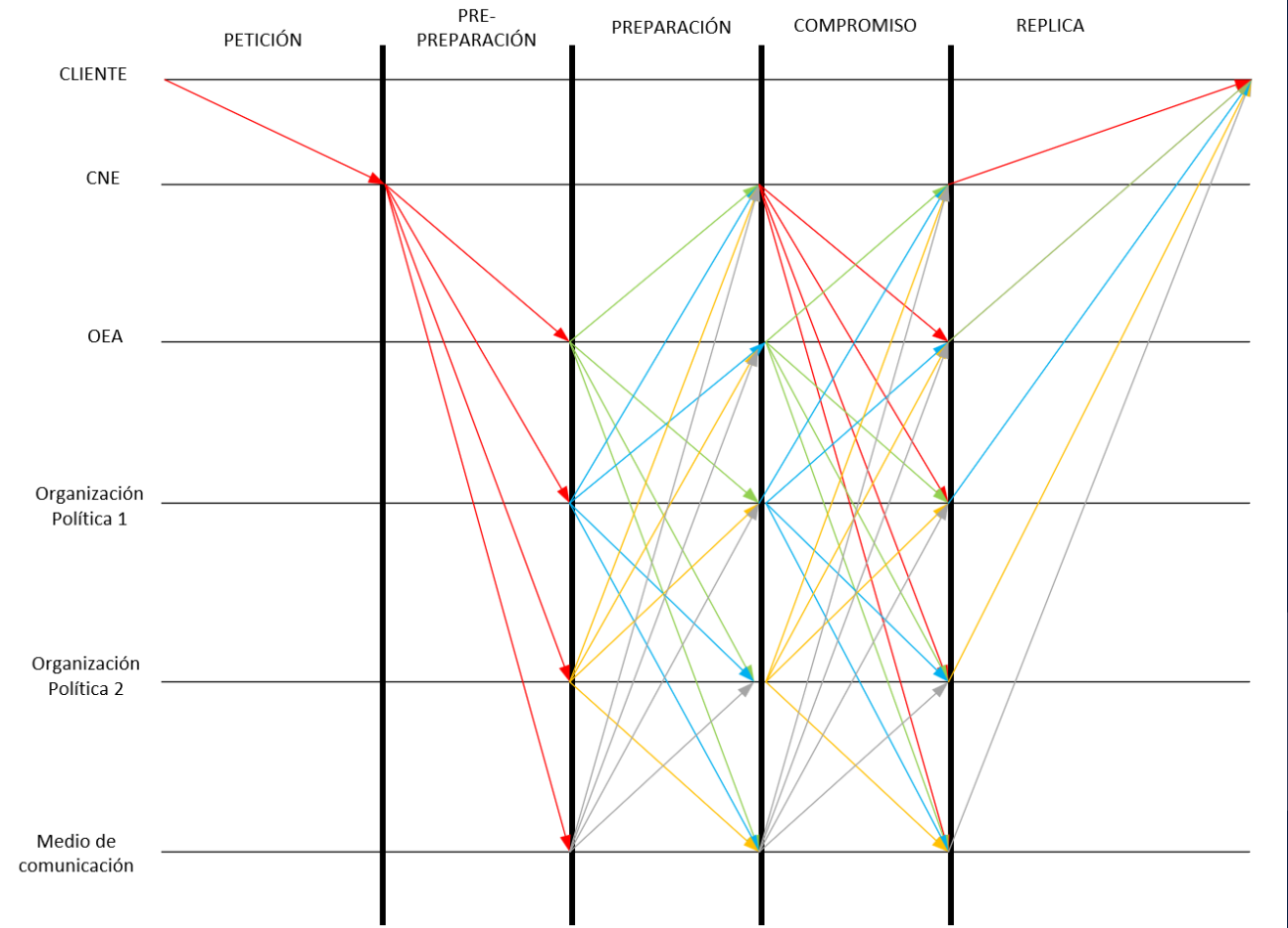
Descentralización y Red

- Para solventar el problema de descentralización, el diseño plantea implementar un SID, en el cuál participen más organizaciones.
- A su vez, también se plantea que este SID se encuentre en una red privada.



Consenso

- Se hará uso de algoritmos contra fallas bizantinas (BFT), ,en específico, el algoritmo PBFT (Practical Byzantine Fault Tolerance).
- Para que una transacción sea realizada dentro del sistema todos los participantes dentro del SID deben aprobarla.



Autoridades Certificadoras (CAs)

Para TLS

- CA utilizada para asegurar las comunicaciones entre las organizaciones.
- Las comunicaciones se realizan con el uso de certificados de TLS.

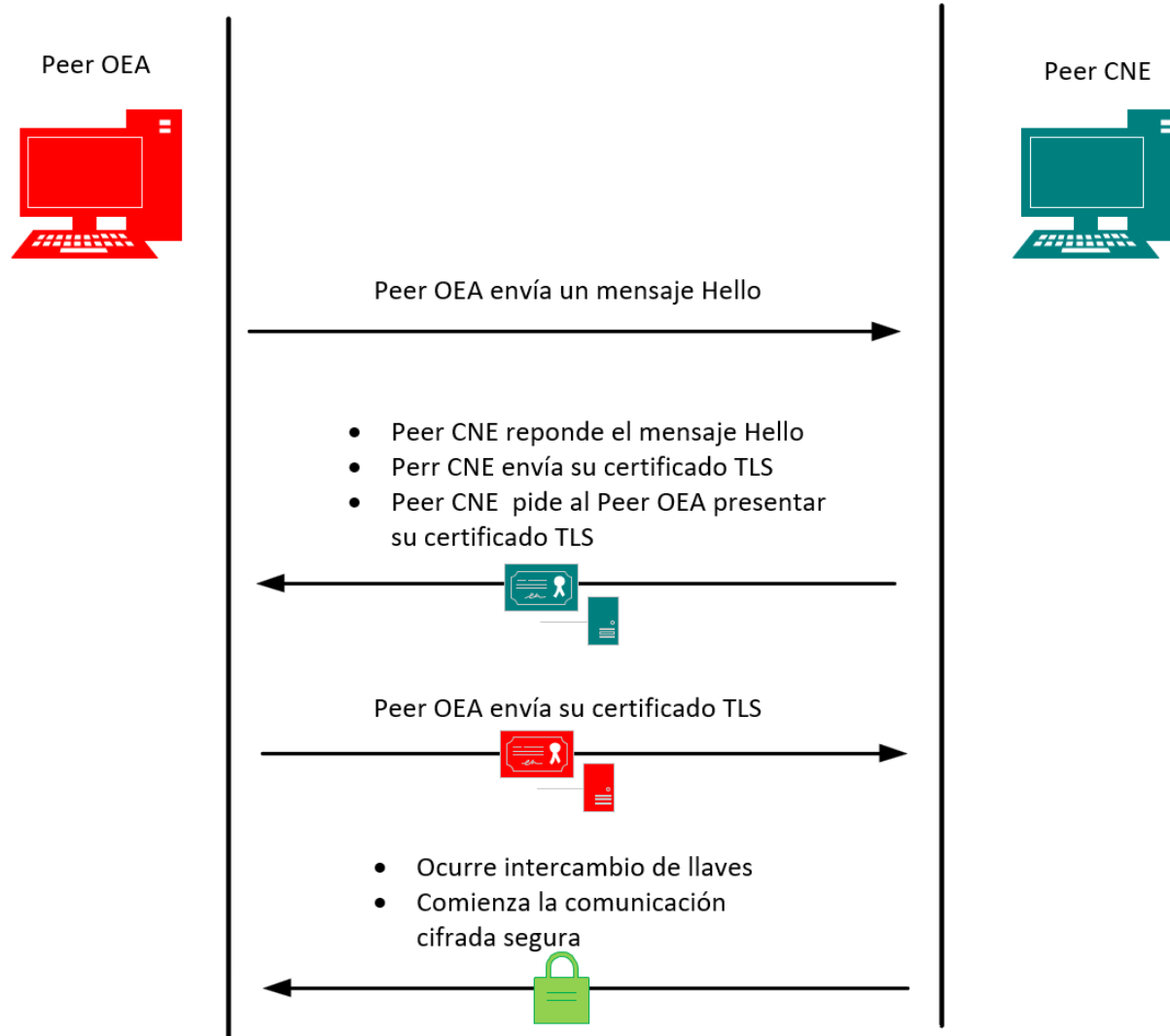


Para Enrollment

- CA encargada de la emisión de certificados de identidad de los usuarios, administradores y nodos participantes de la red.
- Estos certificados son usados para autenticar las identidades y autorizar operaciones dentro del SID.



Funcionamiento TLS



Proveedor de servicios de membresía (MSP)

- Cada organización tendrá su propio MSP.
- MSP específica que certificados de las respectivas cas de las organizaciones serán válidos para actuar en nombre de dicha organización.



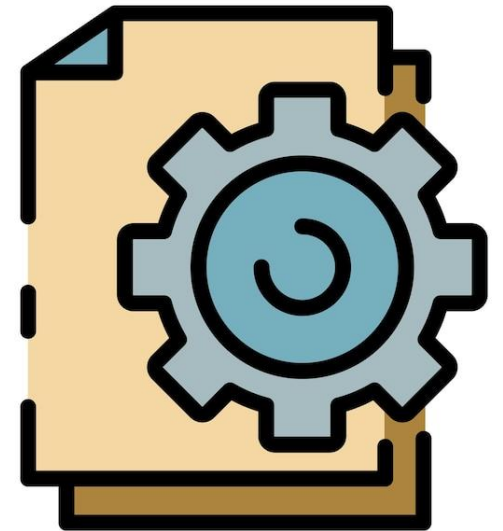
Archivos de configuración

fabric-ca-server-config

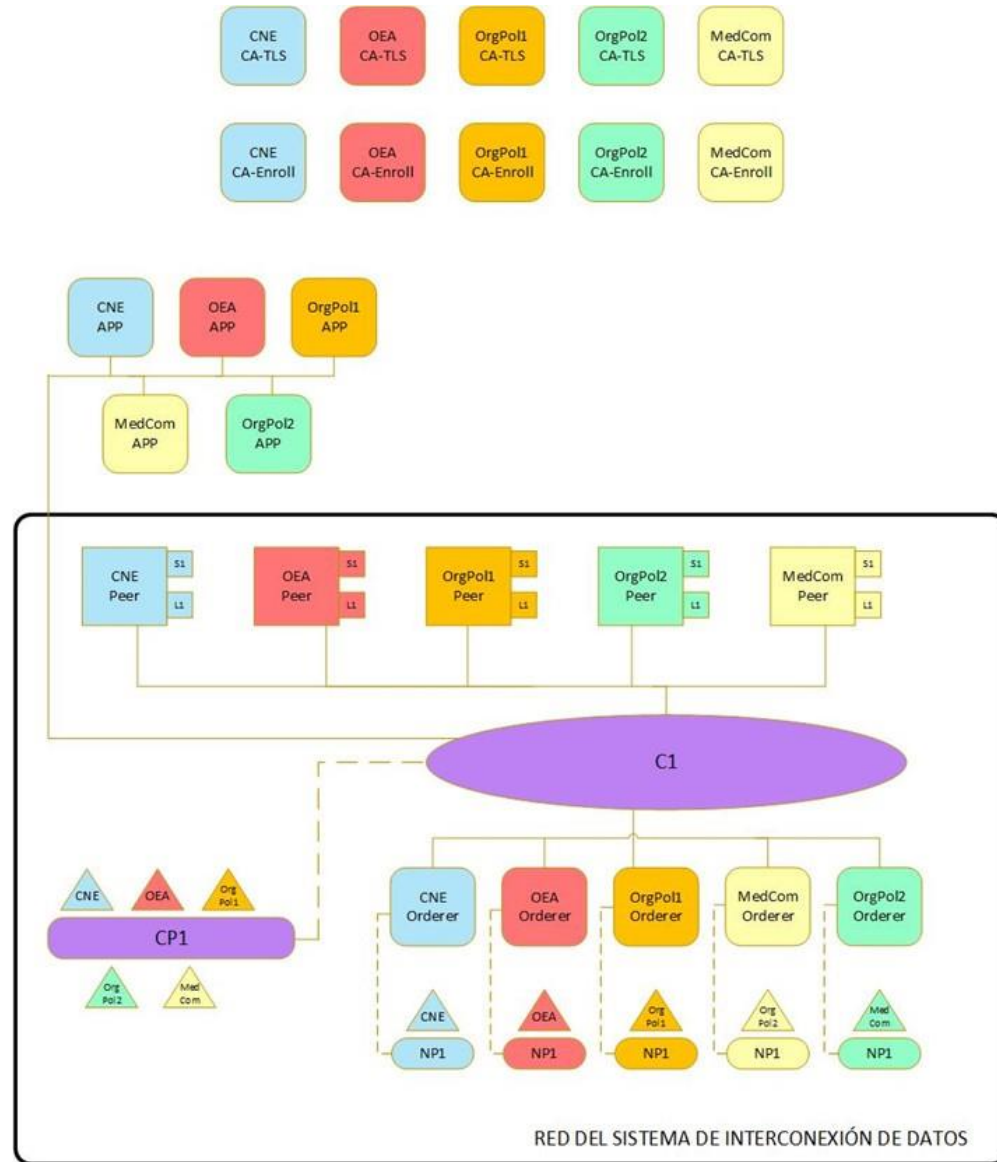
- Define los parámetros de funcionamiento de una CA.
- Todas las organizaciones participantes deberán acordar las configuraciones de sus CAs.

configtx

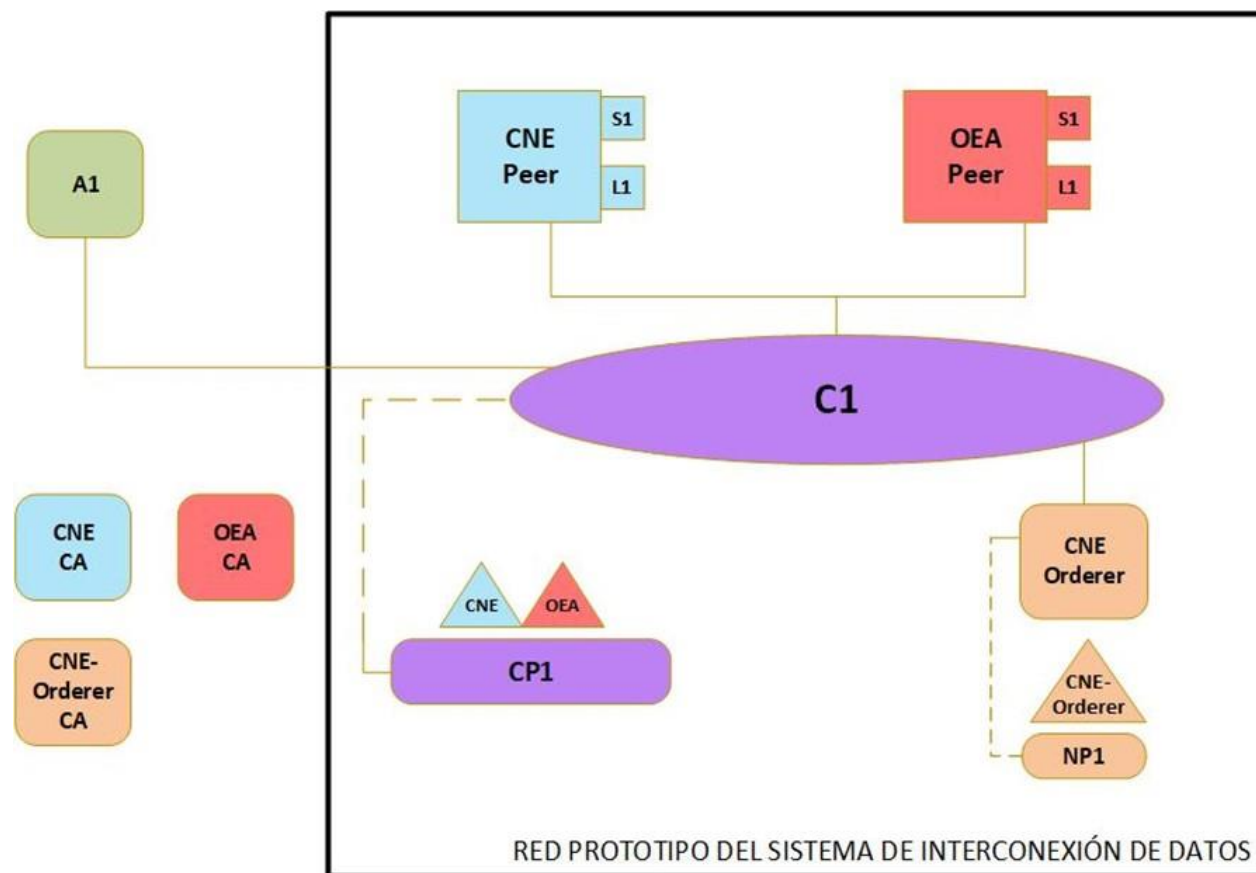
- Define aspectos claves de la red.
- Asegura que las operaciones dentro del SID sigan las mismas reglas y condiciones.
- Todas las organizaciones deben tener el mismo “configtx”.



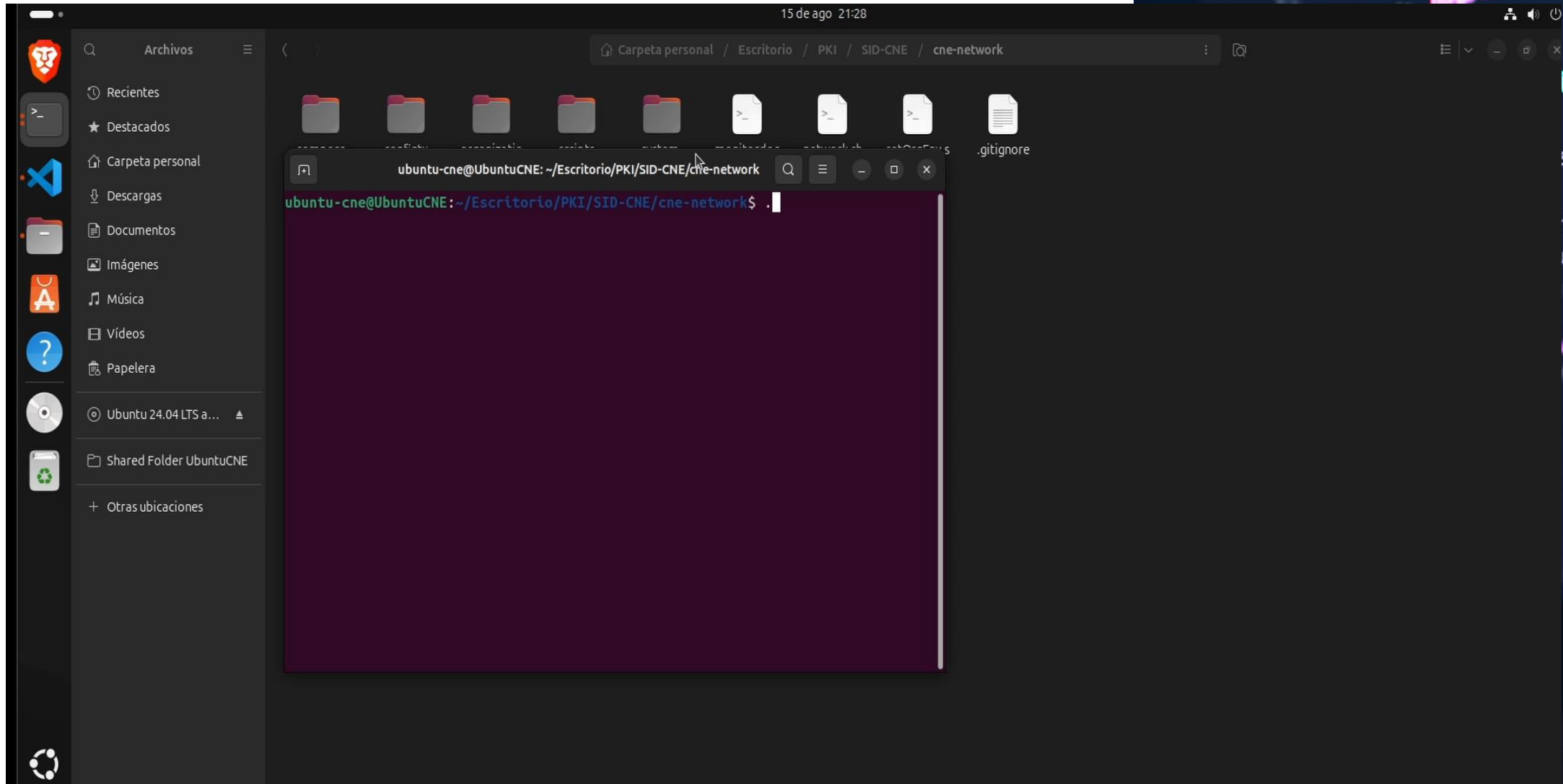
Arquitectura de la red del SID



Arquitectura de la red del SID

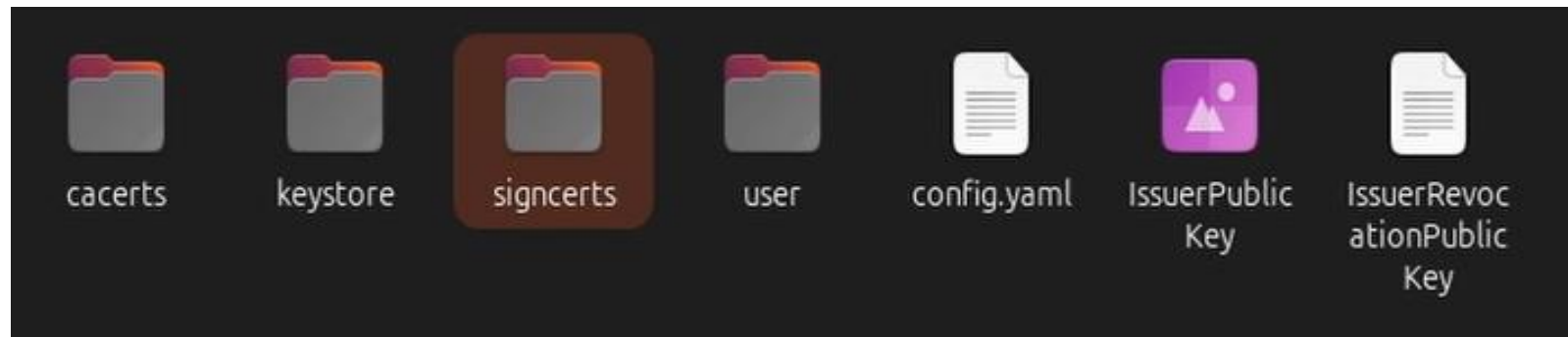


Inicio Contenedores

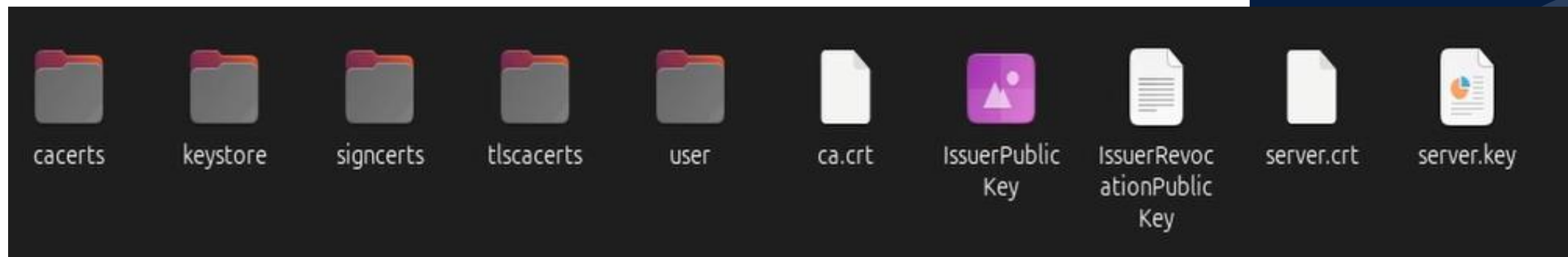


Certificados

MSP



TLS



Creación Canal

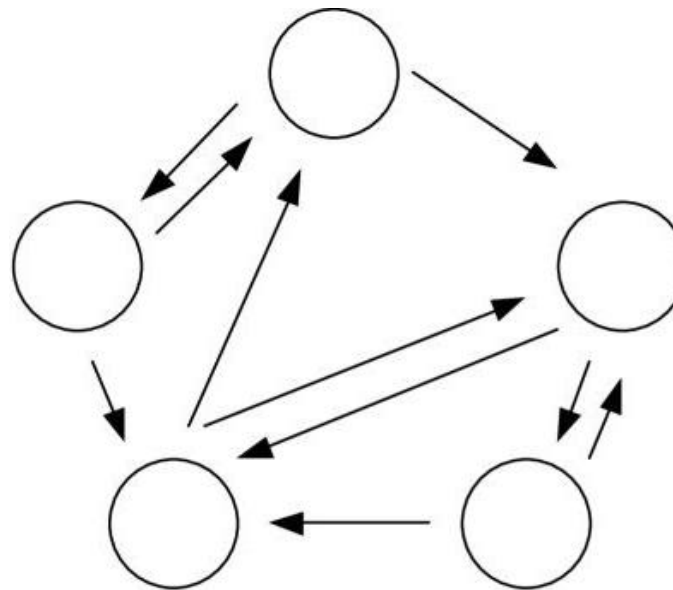
15 de ago 21:33

ubuntu-cne@UbuntuCNE: ~/Escritorio/PKI/SID-CNE/cne-network

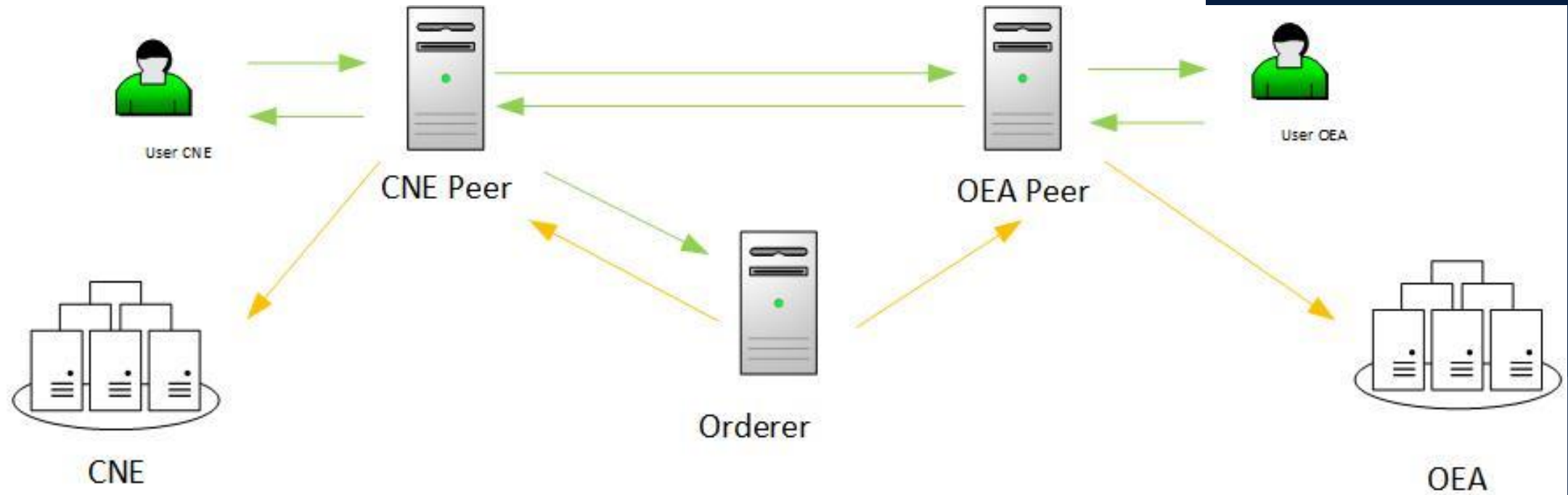
```
2024/08/15 21:28:15 [INFO] encoded CSR
2024/08/15 21:28:15 [INFO] Stored client certificate at /home/ubuntu-cne/Escritorio/PKI/SID-CNE/cne-network/organizations/ordererOrganizations/example.com/users/Admin@example.com/msp/signcerts/cert.pem
2024/08/15 21:28:15 [INFO] Stored root CA certificate at /home/ubuntu-cne/Escritorio/PKI/SID-CNE/cne-network/organizations/ordererOrganizations/example.com/users/Admin@example.com/msp/cacerts/localhost-9054-ca-orderer-cne.pem
2024/08/15 21:28:15 [INFO] Stored Issuer public key at /home/ubuntu-cne/Escritorio/PKI/SID-CNE/cne-network/organizations/ordererOrganizations/example.com/users/Admin@example.com/msp/IssuerPublicKey
2024/08/15 21:28:15 [INFO] Stored Issuer revocation public key at /home/ubuntu-cne/Escritorio/PKI/SID-CNE/cne-network/organizations/ordererOrganizations/example.com/users/Admin@example.com/msp/IssuerRevocationPublicKey
Generating CCP files for Org1 and Org2
Creating volume "compose_orderer.example.com" with default driver
Creating volume "compose_peer0.cne.com" with default driver
Creating volume "compose_peer0.oea.com" with default driver
WARNING: Found orphan containers (ca_cne, ca_oea, ca_orderer_cne) for this project. If you removed or renamed this service in your compose file, you can run this command with the --re
move-orphans flag to clean it up.
Creating peer0.oea.com ... done
Creating peer0.cne.com ... done
Creating orderer.cne.com ... done
Creating cli ... done
CONTAINER ID        IMAGE                                     COMMAND                  NAMES                  CREATED              STATUS              PORTS
0da0a2a830df        hyperledger/fabric-tools:latest         "/bin/bash"             cli                    1 second ago        Up Less than a second
c4a4ffb8157         hyperledger/fabric-peer:latest          "peer node start"       peer0.oea.com          2 seconds ago       Up Less than a second 0.0.0.0:9051->9051/tcp, :::9051->9051/tcp, 7051/tcp, 0.0.0.0:9445->
9445/tcp, :::9445->9445/tcp
3ef30f6ca571        hyperledger/fabric-orderer:latest       "orderer"               orderer.cne.com        2 seconds ago       Up Less than a second 0.0.0.0:7050->7050/tcp, :::7050->7050/tcp, 0.0.0.0:7053->7053/tcp,
:::7053->7053/tcp, 0.0.0.0:9443->9443/tcp, :::9443->9443/tcp
8ad44147ad5d        hyperledger/fabric-peer:latest          "peer node start"       peer0.cne.com          2 seconds ago       Up 1 second         0.0.0.0:7051->7051/tcp, :::7051->7051/tcp, 0.0.0.0:9444->9444/tcp,
:::9444->9444/tcp
a95261ae16d0        hyperledger/fabric-ca:latest            "sh -c 'fabric-ca-se..." ca_oea                 9 seconds ago       Up 7 seconds        0.0.0.0:8054->8054/tcp, :::8054->8054/tcp, 7054/tcp, 0.0.0.0:18054->
18054/tcp, :::18054->18054/tcp
44069269b196        hyperledger/fabric-ca:latest            "sh -c 'fabric-ca-se..." ca_orderer_cne        9 seconds ago       Up 8 seconds        0.0.0.0:9054->9054/tcp, :::9054->9054/tcp, 7054/tcp, 0.0.0.0:19054->
19054/tcp, :::19054->19054/tcp
aac21dd2be81        hyperledger/fabric-ca:latest            "sh -c 'fabric-ca-se..." ca_cne                 9 seconds ago       Up 7 seconds        0.0.0.0:7054->7054/tcp, :::7054->7054/tcp, 0.0.0.0:17054->17054/tcp,
:::17054->17054/tcp
ubuntu-cne@UbuntuCNE:~/Escritorio/PKI/SID-CNE/cne-network$ c
```

Comunicaciones

Dentro de Hyperledger Fabric, las comunicaciones se realizan mediante el protocolo Gossip.



Comunicaciones



The image is a composite. The background is a dark blue space-themed graphic with white stars and a faint URL `https://d` in the top right. Overlaid on this is a terminal window with a dark background and light-colored text. The terminal shows a command being executed, its output, and a file path. The command is `channel --tls --cafile /opt/gopath/src/github.com/hyperl`. The output is a JSON object: `{"host": "peer0.oea.com", "port": 9051}], "version": "0`. Below the output is the file path `_update.pb`. The terminal window also shows a portion of a function definition: `function_` and `up theme`. The overall image has a futuristic, tech-oriented aesthetic.

```
channel --tls --cafile /opt/gopath/src/github.com/hyperl
{"host": "peer0.oea.com", "port": 9051}], "version": "0
_update.pb
```

The image is a composite. The background is a dark blue gradient with blurred elements: a URL `https://d` in pink, a variable `@link` in pink, a variable `@package _s` in pink, and various code snippets in green and yellow, including `function_`, `up theme`, `e that th`, `is before`, `indicati`, `tion incor`, `* Make the`, `* Transla`, `* If you`, `* to char`, `void, struc`, and `use struc`.

In the foreground, a terminal window is open with a dark background. The command entered is:

```
channel --tls --cafile /opt/gopath/src/github.com/hyperl
```

The output of the command is a JSON array of objects, with the first object being:

```
{"host": "peer0.oea.com", "port": 9051}], "version": "0
```

Below the output, the file `_update.pb` is mentioned.



Conclusiones

- Se demostró que la implementación de una infraestructura de llave pública (PKI) en la red blockchain fue efectiva para garantizar la identificación y autenticación segura de los diferentes componentes.
 - El análisis de los problemas de seguridad y transparencia en el proceso actual de escrutinios y publicación de resultados reveló varias vulnerabilidades, como la centralización del sistema o la posibilidad de acceso no autorizado y la manipulación de datos.
 - La revisión de soluciones existentes para una PKI permitió seleccionar tecnologías adecuadas para el proyecto, como Hyperledger Fabric, que ofreció una infraestructura modular y segura para la gestión de identidades y comunicaciones.
- 

Conclusiones

- El diseño de un sistema de seguridad basado en una PKI incluyó la configuración de autoridades certificadoras (CAs) para la emisión de certificados de TLS y Enrollment. Se diseñó un proceso de autenticación y autorización que permitió controlar el acceso y asegurar la integridad de los datos dentro de la red.
- La implementación del prototipo de una PKI en Hyperledger Fabric validó el diseño propuesto, mostrando que es posible asegurar la integridad y autenticidad de las transacciones dentro de la red blockchain.