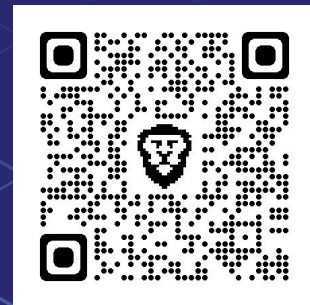



StarkNet 101



Arturo Castañon

 <https://github.com/starknet-edu> | [@starkwareltd](https://twitter.com/starkwareltd)

 <https://github.com/ArturVargas> | [@0xVato](https://twitter.com/0xVato)

 Agosto 2022

Disclaimer:

Si no entiendes algo

- **No es tu culpa, esto es difícil**
- **Casi nadie lo entiende completamente**
- **Aún no soy un experto en el tema**



STARKWARE - Compañía, Fondo por más de \$250M de inversores como Paradigm, Sequoia, Ethereum Foundation, Vitalik Buterin, entre otros, son creadores StarkEx un ZK Rollup Permissionado y de StarkNet un ZK Rollup Descentralizado.

StarkNet - Es un zk Rollup no permissionado y descentralizado que opera como una L2 sobre ethereum permitiendo que cualquier dapp tenga una escalabilidad ilimitada para sus cálculos sin comprometer la seguridad de ethereum.



Qué Son las ZK Proofs

Los Zero Knowledge Proofs son una forma de demostrar que sabes algo sin revelar lo que sabes o como lo sabes, en una configuración criptográfica esto significa que alguien debe probar que sabe algo (**Prover**) y alguien más necesita comprobar que es cierto (**Verifier**)



Pruebas ZK-STARK



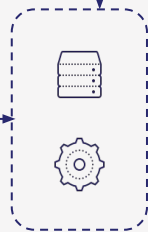
Privacidad (Zero Knowledge, ZK)

Las entradas (inputs) privadas del probador están protegidas



Escalabilidad

Tiempo de ejecución del verificador exponencialmente pequeño*
Tiempo de funcionamiento del probador casi lineal *

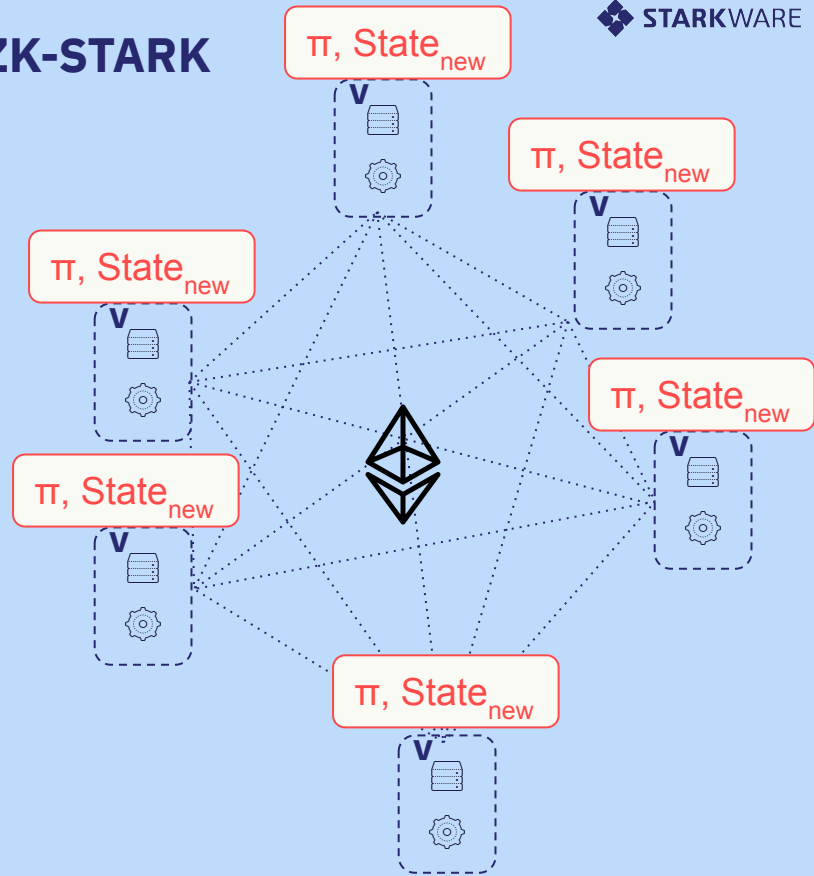


Tirador de pruebas

$\pi, \text{State}_{\text{new}}$

**Con respecto al tamaño del cálculo*

ZK-STARK



Verifique la prueba **STARK**, no confíe

Starknet Components

Prover: Un proceso separado que recibe el resultado de los programas de Cairo y genera las pruebas **STARK** para ser verificadas. El prover envía la prueba al verificador que registra en L1.

Starknet OS: Actualiza el estado de la L2 basado en las transacciones que recibe como inputs.

Starknet State: En Starknet las transacciones no se registran en la blockchain, solo los cambios de estado resultantes de las propias transacciones se registran en Ethereum L1.

Pros

- Transacciones Rápidas y Económicas.
- No necesita un trusted Setup.
- Resistente a la computación cuántica.
- Privacidad.
- Seguridad Compartida con L1.

Cons

- Starknet está centralizado aún.
- Curva de aprendizaje alta.
- Ecosistema y Herramientas aún en construcción.
- Aún no es Open-Source

Anexo

Trusted Setup:

El Trusted Setup es un procedimiento que se realiza una vez para generar un dato que luego debe usarse cada vez que se ejecuta un protocolo criptográfico, para generar esta data se requiere información secreta, la “confianza” proviene del hecho de que alguna persona o grupo de personas tiene que generar estos secretos, usarlos para generar los datos y luego publicar los datos y olvidar los secretos.

Sequencer:

El secuenciador es un nodo que toma un lote de transacciones y genera una lista de cambios causados por todas las transacciones en el lote y una prueba de que si todas las transacciones incluidas en el lote se ejecutan con éxito contra el estado anterior de la red, el resultado será la lista de cambios numerada anteriormente.

La Magia de Cairo

Other blockchains developers:



Rust is better



**NOOOOOOOO! Solidity
is far better**

Starknet developers:



Cairo is hell



Yes

Cairo

Es un lenguaje usado por StarkNet que tiene como objetivo validar los cálculos computacionales e incluir los roles como el prover y el verifier.

En Solidity podemos escribir una función para extraer el balance de una wallet, en Cairo escribimos una función para validar que las partes involucradas y la suma de los saldos no han cambiado.

Puntos Generales

- Cairo es un lenguaje Turing Complete.
- Es un lenguaje de bajo nivel, soporta memoria no determinista de solo lectura.
- Tiene Syntactic Sugar lo que lo hace más amigable.

Tool Box

1. [Wallets](#)
2. [Voyager](#)
3. [Test Ether](#)
4. [Cairo Playground](#)
5. [Protostar](#) / [Nile](#)
6. [StarknetJS](#)



Ejercicios

1. Instalar Wallet.
2. Conseguir ETH testnet.
3. Starknet 101 - Ex 1.
4. Playground.

Abstracción de cuenta

- Las transacciones en StarkNet son diferentes a las de Ethereum - no tienen un originador (“From”)
- Se envían a un “punto de entrada” (“entry point”) - un “contrato inteligente de cuenta” a cargo de:
 - Autenticando el usuario
 - Protección de reproducción
 - Protección de fondos
 - Permite construir muchos UX diferentes
- Esto no se aplica actualmente
 - las transacciones se pueden enviar a cualquier contrato, mientras que el remitente es visto como 0 por el receptor

Next Steps

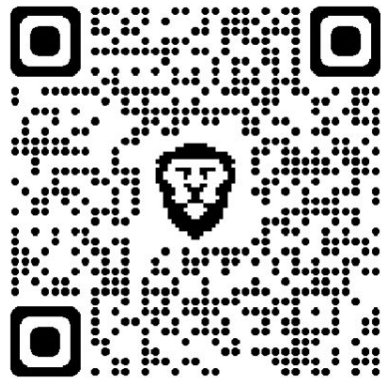
Desplegando su propio token

Implementando un ERC20 en unos sencillos pasos

Sigue estos pasos para acumular todos los puntos

1. **Lee** el ReadMe para una comprensión de alto nivel
2. **Lee** el código de evaluator.cairo para una comprensión de bajo nivel
3. **Usa** el código ERC20 y personalízalo para hacer una ICO
4. **Despliega** tu contrato
5. **Envíe** la dirección de su contrato al evaluador
6. **Pídale** al evaluador que corrija su contrato
7. **Consulta** por puntos
8. Repetir

Ir a Github:



Crear un sick NFT

Implementa un ERC721 en StarkNet

Sigue estos pasos para acumular todos los puntos

1. **Lee** el ReadMe para una comprensión de alto nivel
2. **Lee** el código de evaluator.cairo para una comprensión de bajo nivel
3. **Usa** el código ERC721 y personalízalo para rastrear animales
4. **Despliega** tu contrato
5. **Envíe** la dirección de su contrato al evaluador
6. **Pídale** al evaluador que corrija su contrato
7. **Consulta** por puntos
8. Repetir

Ir a Github:



Magia de capa cruzada

Las cosas buenas.

Sigue estos pasos para acumular todos los puntos

1. **Lee** el ReadMe para una comprensión de alto nivel
2. **Lee** el código de evaluator.cairo para una comprensión de bajo nivel
3. **Interactuar** con L1 y L2 para enviar / transmitir mensajes
4. **Escribir** un contrato de transmisor en L1 y L2
5. **Escribir** un contrato de receptor en L1 y L2
6. **Consulta** por puntos
7. Repetir

Ir a Github:



Depuración de sus contratos

Puntos de ruptura en tu contrato, ¿qué tan genial es eso?

Sigue estos pasos para acumular todos los puntos

1. **Lee** el ReadMe para una comprensión de alto nivel
2. **Ejecute pruebas usando Hardhat, Ape o StarkNet CLI**
3. Repetir

Ir a Github:



Redefine la experiencia de usuario

Personaliza el contrato de tu cuenta

Sigue estos pasos para acumular todos los puntos

1. **Lee** el ReadMe para una comprensión de alto nivel
2. **Lee** el código de evaluator.cairo para una comprensión de bajo nivel
3. **Ejecute el asistente de python para obtener más instrucciones**
4. **Consulta** por puntos
5. Repeat

Ir a Github:



Preguntanos


¡El equipo educativo de StarkNet está aquí!



Gracias!

Equipo educativo de StarkNet

 [@starkwareltd](https://twitter.com/starkwareltd)

 Agosto 2022



Para llegar más lejos

Recursos en StarkNet

- [Documentación oficial de Cairo](#)
- [Links de la Comunidad](#) - lista de ejemplos seleccionados por el equipo de StarkWare
- [Awesome StarkNet](#) - una lista curada de recursos por Georgios Konstantopoulos

Cosas geniales construidas en StarkNet

- [Physics simulation by @guiltygyoza](#)
- [brinq V1 contracts](#)
- [Qasr, an ETH <> StarkNet NFT bridge](#)
- [Tictactoe by @guiltygyoza](#)