



KYC Hook

Hami and Eyal



Overview

- Why do we need KYC
- What is KYC
- The Three lines of defense
- KYC Hook Architecture
- KYC Scenarios
- Live on Sepolia

Why KYC?

Section 326 of the USA PATRIOT Act - OCT. 26, 2001

“(I) IDENTIFICATION AND VERIFICATION OF ACCOUNTHOLDERS.

“(1) IN GENERAL.—Subject to the requirements of this subsection, the Secretary of the Treasury shall prescribe regulations setting forth the **minimum standards** for financial institutions and their customers regarding the identity of the customer that shall apply in connection with the opening of an account at a financial institution.

“(2) MINIMUM REQUIREMENTS.—The regulations shall, at a minimum, require **financial institutions** to implement, and customers (after being given adequate notice) to comply with, reasonable procedures for

“(A) **verifying the identity of any person seeking to open an account** to the extent reasonable and practicable;

“(B) **maintaining records of the information used to verify a person’s identity**, including name, address, and other identifying information; and

“(C) **consulting lists of known or suspected terrorists or terrorist organizations** provided to the financial institution by any government agency to determine whether a person seeking to open an account appears on any such list.

+

•

○

What is KYC ?

KYC – Know Your Customer



collection and analysis of basic identity information



creating an expectation of customers transactional behavior



Assessing propensity to commit **money laundering, terrorist finance** or other **illicit activity**



Monitor customer's transactions against the expected behavior

+

•

◦

The Three Lines of Defense

KYC – Know Your Customer



Operational
Management



Compliance Oversight
and Risk Management



Auditing

Business Entities



Swapper – an individual, fiduciary or institution trading on Uniswap



Liquidity Provider - an individual, fiduciary or institution providing liquidity to pools on Uniswap



Pool owner – an individual fiduciary or institution responsible for initializing a pool and providing initial liquidity



Front-End and Dapp providers – organization owning and supporting front ends and Dapps offering access to Uniswap pools



Notary - a person authorized to identify individuals and confirm the authenticity of certain documents and



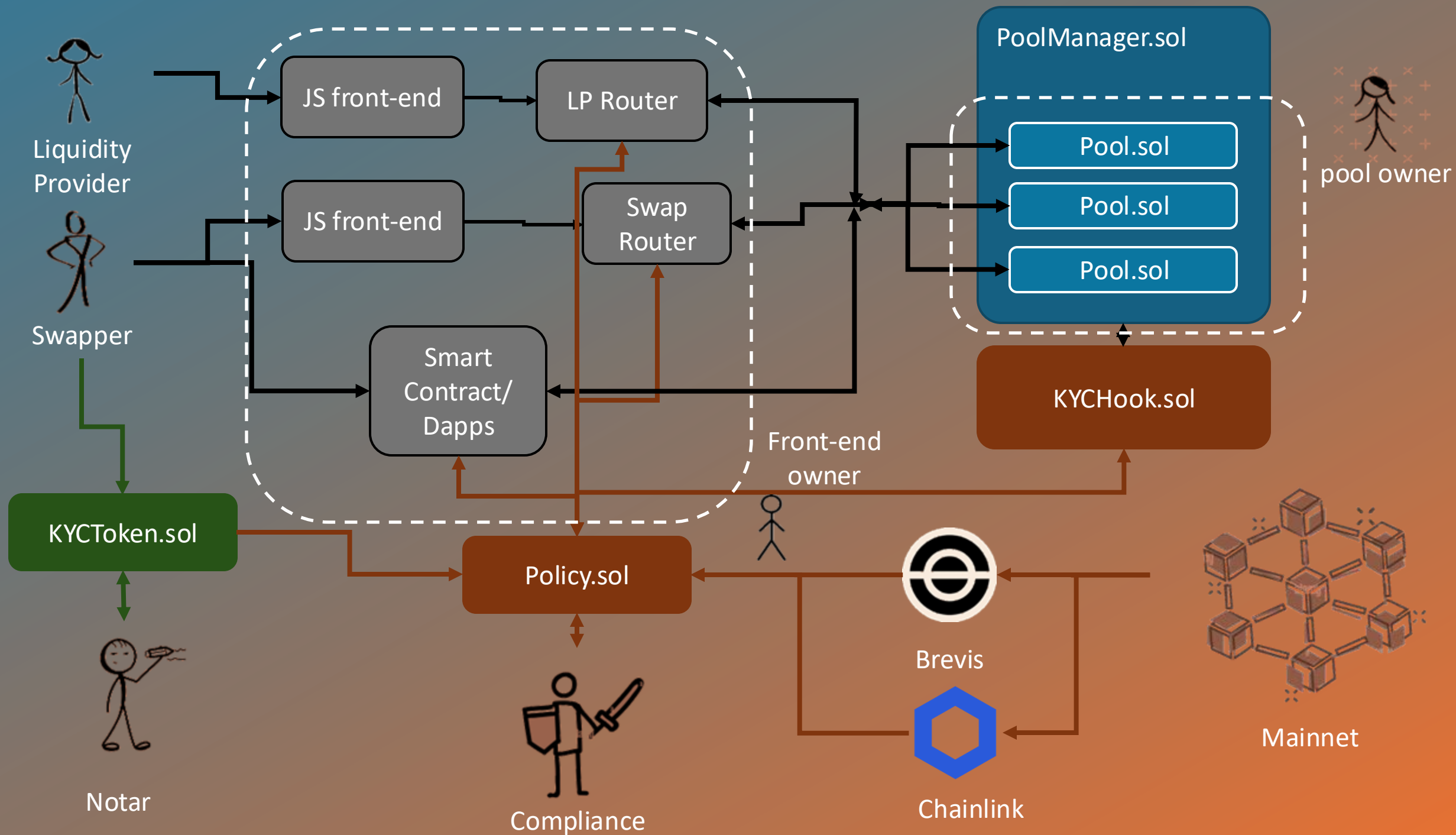
Compliance – an individual or group responsible for setting the KYC standard, compliance oversight and risk management



Chainlink – A third part firm providing oracle services



Brevis – A third part firm providing ZK co-processor services



Why Use a Hook?

```
function swap(...
    delta = abi.decode(
        manager.unlock(abi.encode(CallbackData(msg.sender,
            setting, key, params, hookData))), (BalanceDelta)
    );

function unlockCallback(...

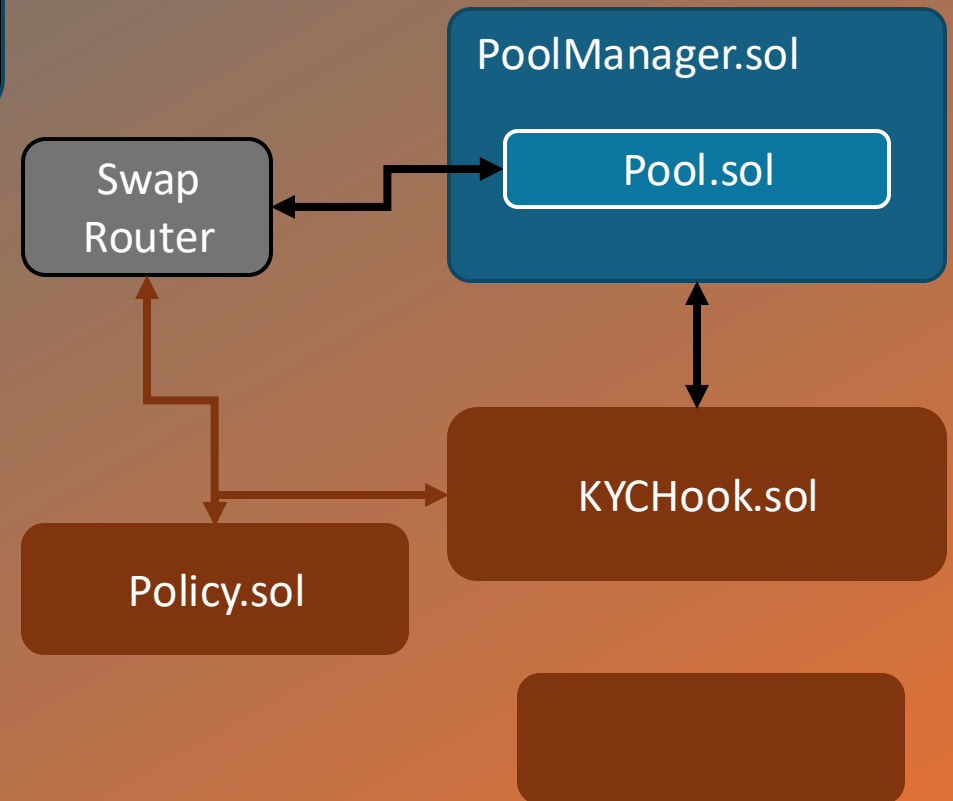
data.key.currency0.take(manager, data.sender,
uint256(deltaAfter0), data.testSettings.takeClaims);
```

In praxis, Liquidity Providers and Swappers interact with PoolManager through a Router. For verification, the hook needs to know the address of the original sender, not the Router -> Router encodes in hookData

A Malicious Router could report a KYC compliant address, while acting on behalf of a non compliant address. Therefore, Trusted Routers are Whitelisted by the Hook, hook functions gate swapping and liquidity provision to the Pools

Compliance rules are defined in **Policy** Smart contracts and can be flexibly designed. Our specific implementation is a **KYCToken Policy** verifying ownership of a **KYC Token** with stores the sender's information for compliance verification.

Our KYCHook can handle different policies for different pools. Policies are upgradable



Auditing

As a third line of defense, transactions on the blockchain are audited and it is continuously assessed whether the addresses reported by the routers as the source of the swap, are the same addresses to which currencies were eventually returned. Mismatch results in the removal of a router from the whitelist of the hook. Auditing can be done based on offchain-indexing.

A more advanced approach is to automate the auditing process:



- Chainlink Automation allows automatic triggering of De-Whitelisting Routers by the Hook. The trigger comes from Processing events and transactions between Router and Swapper



- With Brevis, the automation is trustless. Our approach relies on Proof of Compliant behavior by the Router and De-Whitelisting as a sanction for missing proofs
- The inputs come from purely on-chain data, therefore ideal for Brevis

Scenario Protagonists



Lakshman -
Notary



Sara - LP

KYC Token	
Passport	False
SSCard	True
DriverLicense	True
DoDId	False
BirthCertificate	False
MailedBills	5



Eyal – Swapper

KYC Token	
Passport	False
SSCard	True
DriverLicense	True
DoDId	False
BirthCertificate	False
MailedBills	5



Puja – Pool
Owner



Hami - Swapper

KYC Token	
Passport	False
SSCard	True
DriverLicense	True
DoDId	False
BirthCertificate	False
MailedBills	0



Alice -
Compliance

KYC Policy	
Photo Id	3
Name	3
Address	2
DoB	4
SS#	4
DL#	1
DoD#	2

Scenario 1

KYC Token

Passport	True
SSCard	True
DriverLicense	True
DoDid	False
BirthCertificate	False
MailedBills	3



modifyLiquidityRouter.modifyLiquidity

LiquidityRouter

Clear Pool

kycRouter.swap (fail)

KYC Router

Clear Pool

kycToken.mintTo (success)

kycToken.UpdateIDDocumentBundle (success)

kycRouter.swap (success)

KYC Router

Clear Pool

kycRouter.swap (success)

KYC Router

Clear Pool

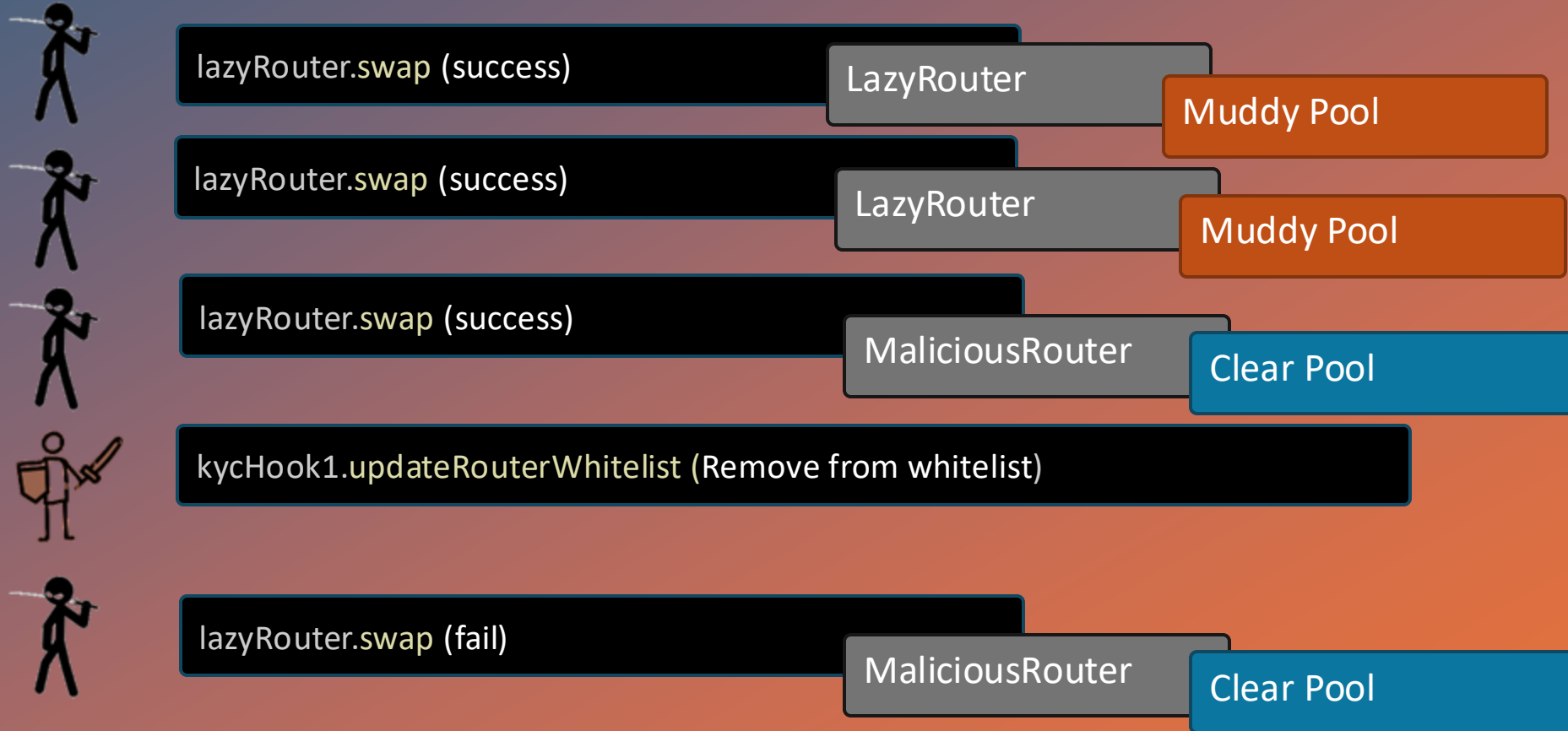
s_kycTokenPolicy.updatePolicyStandard

kycRouter.swap (fail)


KYC Router

Clear Pool

Scenario 2



Deployment on Sepolia

 Etherscan

Transaction Details < >




Overview

Logs (1)

State

Transaction Receipt Event Logs

103

Address 0x8b9c277cebf7290edf10c3151a956cfb42d031f   

Name OwnershipTransferred (index_topic_1 address previousOwner, index_t

Topics

0 0x8be0079c531659141344cd1fd0a4f28419497f9722a3daafe3b4186f6b6457e0

1: previousOwner Dec ▾ → 0x00000000000000000000000000000000

2: newOwner Dec ▾ → 0xFDc61d52721c5eBA3e2fc39190fd9a603256E5a2

Data

0x

Deploying contracts for chainid: 11155111

Link token deployed at 0xA77214ddA874A3d01E7d803173Cc70DCFEe45B7f

USDC token deployed at 0xd6415a6F404F2433df69Fa23809527aCB3A65128

Pool token 0 deployed at 0x2CEbC29106112a2C59e20818DaACDcc1322ACB3e

Pool token 1 deployed at 0xd76Cde845063Ab7925Be67D73537B1D1F045207A

PoolManager deployed at 0xe4C3f51fB478DD18852111d77aa1160083ca6304

KYCRouter deployed at 0x8B9c277cEbF7290EDF10c3151a956FCFb42D031F

MaliciousRouter deployed at 0x3c1749B8435b46B4Ab9DEFE0c3b32AD93Ef04c17

CarelessRouter deployed at 0x852b9e444823253Dfa5402d95766795637c68663

SwapRouter deployed at 0x81913C096c94eB5fD3634219BcDc557D9DA46C74

ModifyLiquidityRouter deployed at 0x74d6eFc23e63D9B5AE79F207598e8E65911cf549

KYCTokenPolicy deployed at 0x967293ADcC54b2A3982d76D26eD9E694cc6d8377

KYCToken deployed at 0x8198c6877F26D1d683A588AF0c5f72BdBa0242e5

Running HookMiner for KYCHook on chainid: 11155111

Mined hook address: 0x92aA8E722d0f801f682f33387dFbC9521ed1b880

Salt: 31061

Deployed hook address: 0x92aA8E722d0f801f682f33387dFbC9521ed1b880

KYCHook deployed at 0x92aA8E722d0f801f682f33387dFbC9521ed1b880

Expected flags: 14464

Actual flags: 837316188020718658314053778152126133795386669184

Hook address matches expected flags

BrevisRequest deployed at 0x00

BrevisProof deployed at 0x00

Minted tokens to swapper, liquidity provider, and rogue user

Approved routers to spend tokens on behalf of swapper hamiha70, 17 hours a

Approved routers to spend tokens on behalf of liquidity provider

Approved routers to spend tokens on behalf of rogue user

Initializing pool with KYC

Comprehensive unit and E2E-Testing

```
[*] Compiling...
No files changed, compilation skipped

Ran 3 tests for test/unit/KYCDummyHook.t.sol:TestKYCBaseHook
[PASS] test_beforeAddLiquidity() (gas: 145697)
[PASS] test_beforeSwap() (gas: 58752)
[PASS] test_validateSwapAuthorization() (gas: 17760)
Suite result: ok. 3 passed; 0 failed; 0 skipped; finished in 23.75ms (761.70µs CPU time)

Ran 8 tests for test/unit/KYCTokenPolicy.t.sol:KYCTokenPolicyTest
[PASS] test_BundleTranslation() (gas: 15980)
[PASS] test_ERC721FunctionsDisabled() (gas: 133267)
[PASS] test_EndToEndKYCTokenPolicy() (gas: 169431)
[PASS] test_OnlyOwnerCanUpdatePolicy() (gas: 36029)
[PASS] test_RevokeToken() (gas: 124269)
[PASS] test_isNotStricterRetailKYCInformation() (gas: 13352)
[PASS] test_setup() (gas: 35072)
[PASS] test_validateRetailKYCInformationAgainstRequirements() (gas: 24420)
Suite result: ok. 8 passed; 0 failed; 0 skipped; finished in 23.52ms (4.07ms CPU time)

Ran 3 tests for test/unit/KYCHook.t.sol:TestKYCHook
[PASS] test_eventEmission() (gas: 569540)
[PASS] test_hookNormalExecution() (gas: 619812)
[PASS] test_setup() (gas: 24424)
Suite result: ok. 3 passed; 0 failed; 0 skipped; finished in 26.12ms (3.79ms CPU time)

Ran 7 tests for test/unit/LazyRouter.t.sol:LazyRouterTest
[PASS] test_LazySwapAttemptThroughRouterEvent() (gas: 642400)
[PASS] test_setup() (gas: 183545)
[PASS] test_swapFailsWithBadPolicy() (gas: 94366)
[PASS] test_swap_lazyRouter() (gas: 775516)
[PASS] test_swap_lazyRouter_withBlacklistedUser() (gas: 605325)
[PASS] test_swap_lazyRouter_withMalformedHookData() (gas: 656224)
[PASS] test_swap_standardRouters() (gas: 883821)
Suite result: ok. 7 passed; 0 failed; 0 skipped; finished in 31.16ms (12.09ms CPU time)

Ran 7 tests for test/unit/KYCHookMultiplePolicies.t.sol:TestKYCHookMultiplePolicies
[PASS] test_BlackwhitelistPolicy() (gas: 327596)
[PASS] test_BlackwhitelistPolicyHook() (gas: 507146)
[PASS] test_BlacklistPolicy() (gas: 165952)
[PASS] test_BlacklistPolicyHook() (gas: 479223)
[PASS] test_WhitelistPolicy() (gas: 180637)
[PASS] test_WhitelistPolicyHook() (gas: 492750)
[PASS] test_setup() (gas: 71241)
Suite result: ok. 7 passed; 0 failed; 0 skipped; finished in 31.76ms (9.20ms CPU time)
```

```
Ran 6 tests for test/unit/Blackwhitelist.t.sol:TestBlackwhitelistHook
[PASS] test_BlackwhitelistPolicy() (gas: 339762)
[PASS] test_BlackwhitelistPolicyHook() (gas: 666427)
[PASS] test_BlacklistPolicy() (gas: 175573)
[PASS] test_BlacklistPolicyHook() (gas: 640672)
[PASS] test_WhitelistPolicy() (gas: 190240)
[PASS] test_WhitelistPolicyHook() (gas: 647524)
Suite result: ok. 6 passed; 0 failed; 0 skipped; finished in 31.80ms (9.36ms CPU time)

Ran 5 tests for test/unit/MaliciousRouter.t.sol:MaliciousRouterTest
[PASS] test_setup() (gas: 193763)
[PASS] test_swapCorrectSwapper_maliciousRouter_shouldSucceedAndEmitEventIfNotBlacklisted() (gas: 803454)
[PASS] test_swapFailsWithBadPolicy() (gas: 93150)
[PASS] test_swapFakedSwapper_maliciousRouter_withBlacklistedUser_shouldSucceedAndEmitEvent() (gas: 7081)
[PASS] test_swap_standardRouters() (gas: 883697)
Suite result: ok. 5 passed; 0 failed; 0 skipped; finished in 31.90ms (9.79ms CPU time)

Ran 9 tests for test/unit/KYCRouter.t.sol:KYCRouterTest
[PASS] test_KYCRouter_errorHandling() (gas: 734912)
[PASS] test_KYCRouter_nativeCurrency() (gas: 3855)
[PASS] test_KYCRouter_swapParameters() (gas: 1229866)
[PASS] test_KYCRouter_swapSettings() (gas: 1268834)
[PASS] test_eventEmission() (gas: 759151)
[PASS] test_setup() (gas: 183561)
[PASS] test_swapFailsWithBadPolicy() (gas: 96515)
[PASS] test_swap_KYCRouter() (gas: 783916)
[PASS] test_swap_standardRouters() (gas: 883921)
Suite result: ok. 9 passed; 0 failed; 0 skipped; finished in 32.61ms (20.04ms CPU time)

Ran 8 test suites in 109.50ms (232.62ms CPU time): 48 tests passed, 0 failed, 0 skipped (48 total tests)
```


+

•

○

Takeaway

- KYC controls are essential to enable mainstream finance access to Uniswap markets
- We presented a design for liquidity pool with access restricted to KYC verified swappers and LPs using the UniswapV4 Hook framework
- We demoed an implementation of a KYC pool and demonstrated its resilience to various attack vectors