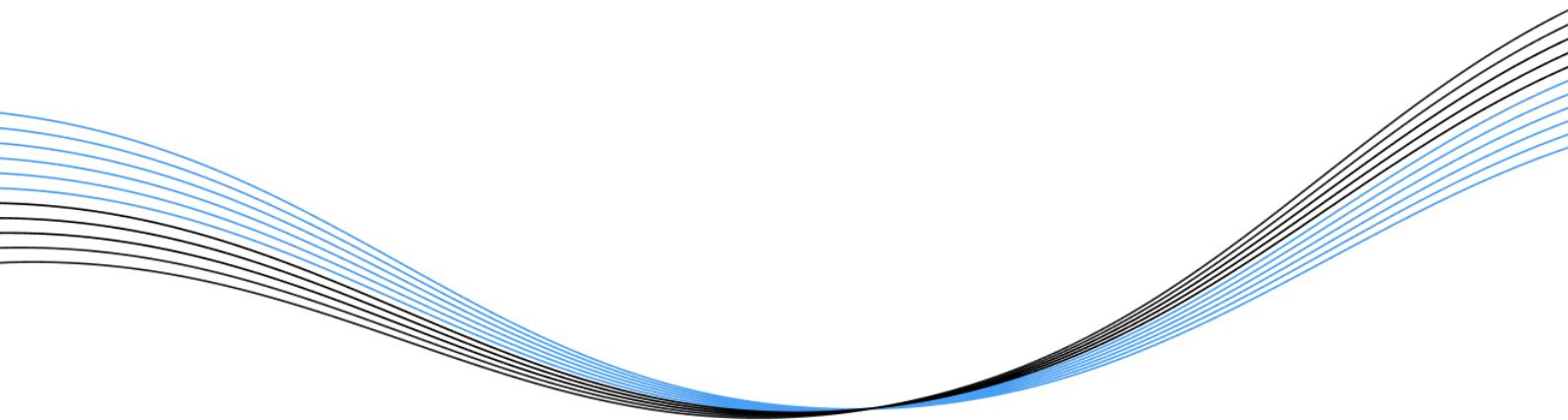


Hexlant.

© Hexlant Inc.
340 Gangnam-daero
Seoul, Republic of Korea
06242
Hexlant.com

KLAYBANK-PROTOCOL SMART CONTRACT AUDIT REPORT



Audit Date
17 Mar 2022

Category
De-fi

Auditor
Hexlant Audit Team

This audit report specifies that the Hexlant Technical Team validated and notified that it has no technical defects.

AUDIT

OVERVIEW

PUBLISHED INFORMATION

REPORT NUMBER	HEXLANT20220302_01
DATE	2022/03/02
AUDIT SCOPE	LendingPoolAddressesProvider.sol LendingPoolAddressesProviderRegistry.sol DefaultReserveInterestRateStrategy.sol LendingPool.sol LendingPoolConfigurator.sol BToken.sol StableDebtToken.sol VariableDebtToken.sol WETHGateway.sol KlaybankProtocolDataProvider.sol

PROJECT INFORMATION

TITLE	KlayBank Protocol
TYPE	DEFI
PLATFORM	KLAYTN
CONTRACT ADDRESS	
REPOSITORY	https://github.com/klaybank/klaybank-audit-hexlant
AUDIT COMMIT HASH	LATEST 14641c475fdaf23879beb0560d4fab9c2d254b54

VULNERABILITY ANALYSIS

CRITICAL	0	No relevant provision
HIGH	0	No relevant provision
MEDIUM	0	No relevant provision
LOW	0	No relevant provision

CENTRALIZED FUNCTION

freezeReserve, unfreezeReserve	Prevent specific reserve's actions in an emergency
setPoolPause	Pause Pool in an emergency

activateReserve, deactivateReserve	Activate or deactivate a reserve
setLendingPoolImpl	Set Lending Pool implement contract of proxy contract
enableBorrowingOnReserve, disableBorrowingOnReserve	Set reserves enable to be borrowed
setReserveFactor	Function to set reserveFactor
setReserveInterestRateStrat egyAddress	Function to set reserve's interestRateStrategy
setFlashLoanPremiumTotal	Function to set flashloan premium

COMPANY PROPOSAL

Hexlant는 2018년에 설립한 블록체인 기술 기업입니다. 삼성전자 출신의 보안·네트워크·소프트웨어 전문가가 스마트 컨트랙트와 블록체인 프로토콜의 보안 결함을 발견하고 블록체인 생태계의 기술 안정성을 입증하기 위해 설립하였습니다.

Hexlant는 블록체인 동작 환경을 파악하기 위해 20개 이상의 블록체인 메인넷을 직접 구축하고 있습니다. 나아가 키 보안 알고리즘 및 메인넷 모니터링 기술을 개발했습니다. 이 방식은 비트코인, 이더리움, 폴카닷, 애이다 등 헥슬란트가 보유한 모든 메인넷 플랫폼에서 적용되고 운영됩니다.

Hexlant는 위와 같은 기술 운영 경험을 바탕으로 스마트 컨트랙트 기술을 검증합니다. 스마트 컨트랙트 내 버그를 발견하는 오류 테스트 뿐만 아니라 메인넷 상황에서의 문제점을 탐지하며 서비스 관점에서 지속적으로 운영할 수 있는 블록체인 기술 가이드를 제공합니다.

Hexlant의 고객사는 컨트랙트에 대한 취약성 감사부터 오너 키 관리, 블록체인 지갑 시스템 구축 등 블록체인 기술 전반의 서비스를 제공받을 수 있습니다. 현재 200여개의 고객사가 Hexlant의 서비스를 바탕으로 블록체인 사업을 시작, 운영했으며 누적으로 관리하는 자산은 12조를 달성했습니다.

Initials for identification purposes:



For identification purposes
Hexlant.

CONTENTS

- 1. Analysis Purpose**
- 2. Vulnerability Classification**
- 3. Test Result**
- 4. Test Case**
- 5. Vulnerability Analysis**
- 6. Conclusion**

ANALYSIS PURPOSE

본 리포트는 발행된 컨트랙트 코드가 요구사항을 충분히 만족하는지, 그리고 보안의 취약점과 실제 운영하면서 발생 할 수 있는 문제들을 파악하고 해결방안을 찾기위해 분석을 수행하고 그 결과를 정리하였습니다. 이번 코드 분석은 다음과 같은 요소들을 검증하기위해 진행하였습니다.

- 구현된 기능의 정상 작동 여부
- 기능 수행 중 보안 위험성
- Off Chain에서 발생하는 문제에 대한 대비
- 컨트랙트 코드의 가독성 및 코드 완성도

VULNERABILITY CLASSIFICATION

본 취약성 검증은 오류 위험도를 아래와 같이 분류, 평가합니다.

• Critical Severity

심각성 치명적 단계는 큰 보안 결함을 뜻하며 자산 탈취 및 동결, 추가 발행 등 치명적인 문제를 야기합니다. 본 결함은 반드시 수정되어 합니다.

• High Severity

심각성 높은 단계는 특수 조건에 의해 보안 결함이 발생할 수 있는 항목이며 수정을 강력하게 권고합니다.

• Medium Severity

심각성 중간 단계는 보안 결함은 아니나 비효율적인 컨트랙트 동작을 야기합니다. 컨트랙트를 효율적으로 동작하도록 수정을 권유하는 항목입니다.

• Low Severity

심각성 낮음 단계는 보안에는 문제가 없으나 컨트랙트 구조 개선을 위해 수정을 권유하는 항목입니다.

Severity	Issue	Status (Found)	Status (Resolve)
• CRITICAL (0)	-		
• HIGH (0)	-		
• MEDIUM (0)	-		
• LOW (0)	-		

TEST RESULT

Code Coverage

코드 커버리지는 작성한 테스트가 얼마만큼 컨트랙트 코드의 기능을 테스트 했는지 알 수 있는 정량적인 지표입니다.

라이브러리와 일부 컨트랙트에 구현된 기능에 대해 추가적인 훈련이 진행되지 않은 경우가 존재합니다.

아래의 Coverage 지표는 위 사항을 반영한 결과입니다.

File Name	Statements	Functions	Lines
LendingPoolAddressesProvider.sol	87.5%	90.91%	87.5%
LendingPoolAddressesProviderRegistry.sol	90%	100%	90%
DefaultReserveInterestRateStrategy.sol	84.62%	40%	84.21%
LendingPool.sol	88%	91.43%	88.14%
LendingPoolConfigurator.sol	86.41%	89.66%	86.67%
BToken.sol	75%	63.64%	75.38%
StableDebtToken.sol	83.7%	69.57%	83.7%
VariableDebtToken.sol	88.89%	73.33%	88.89%
WETHGateway.sol	87.1%	72.73%	87.1%
KlaybankProtocolDataProvider.sol	72.97%	85.71%	69.23%

TEST CASE

실제 적용한 테스트케이스 목록입니다. 총 73개 테스트 시나리오를 적용하여 검수했습니다.

Test Case 1. addressProvider.sol / addressProviderRegistry.sol (6개의 테스트)

SETTERS

1.1. addressProvider	Result
owner 주소만이 setter 함수들을 호출할 수 있는가	PASS FAIL
setter 함수 호출 시 이벤트를 발생시키는가	PASS FAIL

1.2. addressProviderRegistry	Result
owner 주소만이 addressProvider를 등록할 수 있는가	PASS FAIL
owner 주소만이 addressProvider를 제거할 수 있는가	PASS FAIL
등록되지 않은 addressProvider 제거 시 예외처리 되는가	PASS FAIL
등록하려는 addressProvider의 ID가 0일 경우 예외처리 되는가	

Test Case 2. Btoken.sol / StableDebtToken.sol / VariableDebtToken.sol (13개의 테스트)

MODIFIER

2.1. onlyLendingPool	Result
LendingPool이 아닌 주소에서 BToken, StableDebtToken, VariableDebtToken을 발행(mint) 시 예외처리 되는가	PASS FAIL
LendingPool이 아닌 주소에서 BToken, StableDebtToken, VariableDebtToken을 소각(burn) 시 예외처리 되는가	PASS FAIL
LendingPool이 아닌 주소에서 BToken.transferOnLiquidation() 실행 시 예외처리 되는가	PASS FAIL
LendingPool이 아닌 주소에서 BToken.transferUnderlyingTo() 실행 시 예외처리 되는가	PASS FAIL

PERMIT

2.2. permit	Result
유효하지 않은 expiration와 호출 시 예외처리 되는가	PASS FAIL
유효하지 않은 서명과 호출 시 예외처리 되는가	PASS FAIL
permit() 실행 시 approve를 업데이트 하는가	PASS FAIL

새로운 permit 제출 시 이전 permit을 취소하는가	PASS	FAIL
----------------------------------	------	------

TRANSFER

2.3. setStaking	Result	
BToken을 zero address로 전송 시 예외처리 되는가	PASS	FAIL
BToken 전송 시 수량이 0일 경우 예외처리 되는가	PASS	FAIL
BToken 전송 시 토큰 수량이 정상적으로 업데이트 되는가	PASS	FAIL
BToken 전송 시 이벤트가 발생하는가	PASS	FAIL
BToken이 담보로 설정되어 있을 때, 전송 가능 수량보다 많은 토큰 전송 시 예외처리 되는가	PASS	FAIL

Test Case 3. LendingPoolConfigurator.sol (13개의 테스트)

CONFIGURATOR

3.1. batchInitReserve	Result	
admin 주소만이 reserve를 초기화 할 수 있는가	PASS	FAIL
이미 초기화 된 reserve를 초기화 할 경우 예외처리 되는가	PASS	FAIL
유저의 BondInfo 를 정확하게 업데이트 하는가?	PASS	FAIL
reserve 초기화 시 ReserveData, ReserveConfigurationMap를 정확히 업데이트 하는가	PASS	FAIL
이벤트를 발생시키는가	PASS	FAIL

3.2. activate / deactivate rserve	Result	
admin 주소만이 reserve를 활성화/비활성화 할 수 있는가	PASS	FAIL
활성화/비활성화 시 ReserveConfigurationData를 정확히 업데이트 하는가	PASS	FAIL

3.3. setReserveFactor	Result	
admin 주소만이 reserveFactor를 설정할 수 있는가	PASS	FAIL
유효하지 않은 reserveFactor 설정 시 예외처리 되는가	PASS	FAIL

3.4. update Implement	Result	
admin 주소만이 updateBToken(), updateStableDebtToken(), updateVariableDebtToken()를 호출할 수 있는가	PASS	FAIL
BToken, StableDebtToken, VariableDebtToken의 implement 컨트랙트를 업데이트 하는가	PASS	FAIL

3.5. pause	Result
emergencyAdmin 주소만이 setPoolPause()를 호출할 수 있는가	PASS FAIL
pause시, lendingpool의 함수가 실행되지 않는가?	PASS FAIL

Test Case 4. LendingPool.sol (35개의 테스트)

DEPOSIT / WITHDRAW

4.1. deposit	Result
deposit시 유저가 전달한 reserve 토큰과 같은 양의 BToken을 발행하는가	PASS FAIL
deposit시 이벤트를 발생시키는가	PASS FAIL
onBehalfOf 주소 지정시, BToken을 onBehalfOf 주소에 발행하는가	PASS FAIL
deposit 수량이 0일 경우 예외처리 되는가	PASS FAIL
등록되지 않은 토큰을 deposit 시 예외처리 되는가	PASS FAIL
freeze된 reserve 토큰을 deposit 시 예외처리 되는가	PASS FAIL

4.2. withdraw	Result
withdraw시 유저가 소각한 BToken 토큰과 같은 양의 reserve 토큰을 전달하는가	PASS FAIL
withdraw시 이벤트를 발생시키는가	PASS FAIL
to주소 지정시, reserve 토큰을 to주소에 전달하는가	PASS FAIL
withdraw 수량이 0일 경우 예외처리 되는가	PASS FAIL
withdraw 시 담보가 부족해질 경우 예외처리 되는가	PASS FAIL

BORROW / REPAY

4.3. borrow	Result
LendingPoolConfigurator.enableBorrowingOnReserve() 함수로 등록된 reserve만을 빌릴 수 있는가	PASS FAIL
LendingPoolConfigurator.configureReserveAsCollateral() 함수로 등록된 reserve만을 담보로 설정할 수 있는가	PASS FAIL
borrow 시 유저에게 정확한 수량의 reserve 토큰을 전달하는가	PASS FAIL
CollateralBalanceETH 보다 큰 가치의 토큰을 borrow 시 예외처리 되는가	PASS FAIL
borrow 수량이 0일 경우 예외처리 되는가	PASS FAIL
borrow 시 이벤트를 발생시키는가	PASS FAIL

잘못된 borrow mode를 설정할 경우 예외처리 되는가	PASS	FAIL
----------------------------------	------	------

4.4. repay	Result	
repay 시 정확한 수량의 debtToken을 소각하는가	PASS	FAIL
repay 수량이 0일 경우 예외처리 되는가	PASS	FAIL
repay 시 이벤트를 발생시키는가	PASS	FAIL
onBehalfOf 주소 지정시, onBehalfOf 주소의 debtToken을 소각하는가	PASS	FAIL

4.5. swapBorrowRateMode	Result	
swapBorrowRateMode() 실행 시 해당 reserve의 borrow mode를 변경하는가	PASS	FAIL

FLASHLOAN

4.6. flashloan	Result	
receiverAddress.executeOperation()을 실행하는가	PASS	FAIL
premium amount를 treasury 주소로 전달하는가	PASS	FAIL
유효하지 않은 flashloan mode와 호출 시 예외처리 되는가	PASS	FAIL
receiverAddress가 reserve를 반환하지 않을 경우 예외처리 되는가	PASS	FAIL
receiverAddress.executeOperation() 실패 시 예외처리 되는가	PASS	FAIL
이벤트를 발생시키는가	PASS	FAIL

LIQUIDATION

4.7. liquidation	Result	
healthfactor > 1 일때 liquidationCall 호출 시 예외처리 되는가	PASS	FAIL
청산 대상과 다른 reserve와 호출 시 예외처리 되는가	PASS	FAIL
청산 대상과 다른 담보와 호출 시 예외처리 되는가	PASS	FAIL
liquidationCall 호출 시 정확한 수량의 담보 토큰을 전달하는가	PASS	FAIL
청산 대상의 정보를 업데이트 하는가	PASS	FAIL
이벤트를 발생시키는가	PASS	FAIL

Test Case 5. DefaultReserveInterestRateStrategy.sol (2개의 테스트)

5.1. DefaultReserveInterestRateStrategy	Result	
configurator.setReserveInterestRateStrategyAddress() reserve의 InterestRateStrategy를 업데이트 하는가	PASS	FAIL
utilizatioin rate에 따라 정확한 borrow rate를 계산하는가	PASS	FAIL

Test Case 6. WETHGate.sol (4개의 테스트)

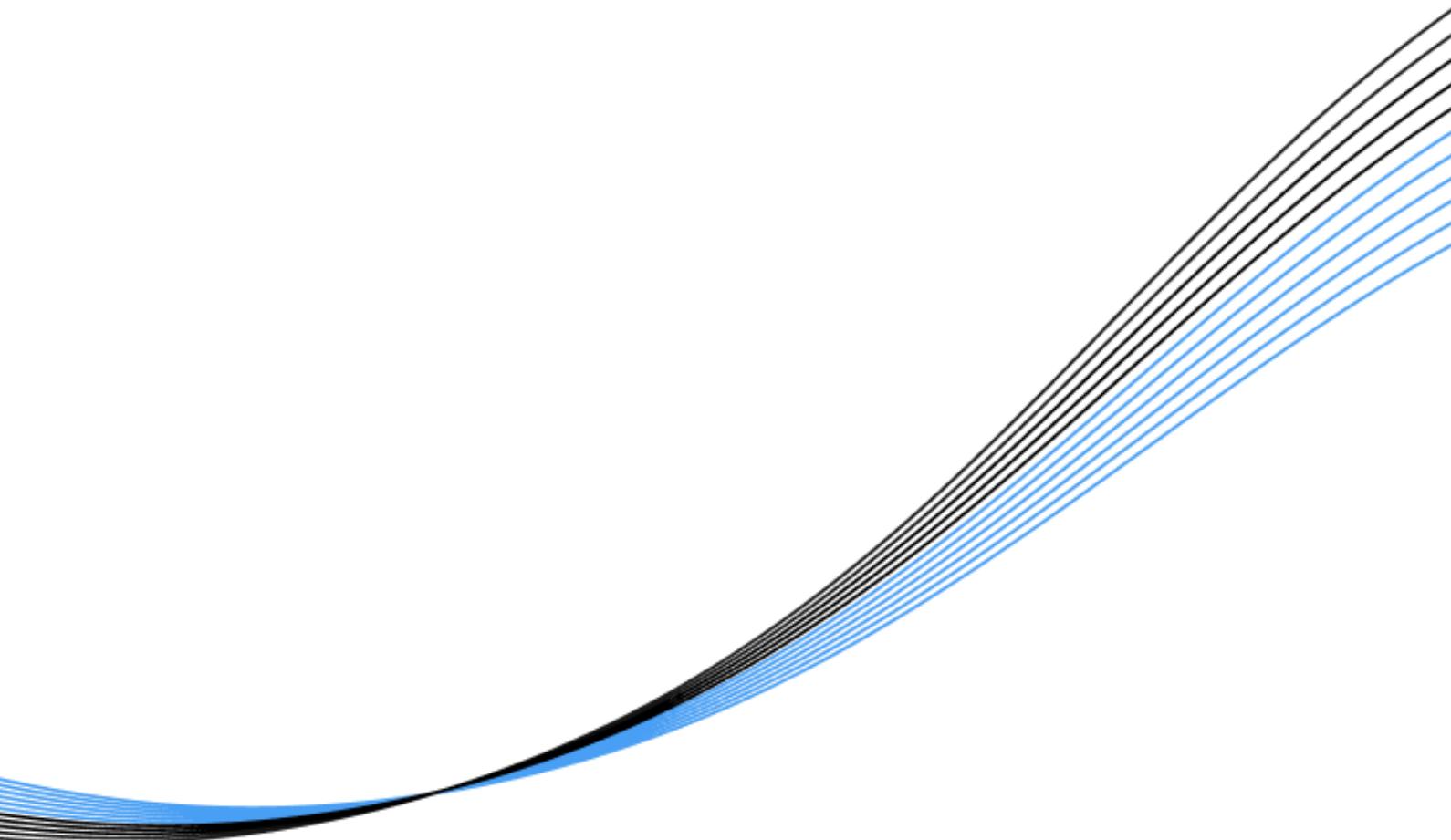
6.1. WETHGate	Result	
owner 주소만이 authorizeSendingPool를 호출할 수 있는가	PASS	FAIL
deposit, borrow, repay, withdraw가 정상적으로 작동하는가	PASS	FAIL
WKlay 컨트랙트 주소가 아닌 다른 주소로부터 Klay를 받았을 경우 예외처리 되는가	PASS	FAIL
fallback 함수 실행 시 예외처리 되는가	PASS	FAIL

VULNERABILITY ANALYSIS

해당 컨트랙트에서는 취약점이 발견되지 않았습니다.

Declare

해당 리포트는 Hexlant의 스마트 컨트랙트 보안 감사 결과를 바탕으로 작성되었습니다. 해당 리포트는 비즈니스 모델의 적합성과 법적 규제, 투자에 대한 의견을 보증하지 않습니다. 리포트에 기술한 문제점 이외에 메인넷 기술 또는 가상머신을 비롯하여 발견되지 않은 문제점이 있을 수 있습니다. 해당 리포트는 논의 목적으로만 사용됩니다.



Hexlant.

-
contact@hexlant.com
www.hexlant.com