DcentraLab Diligence

dcentralab.com/diligence

Audit Report

HyperCycle - SwapV2

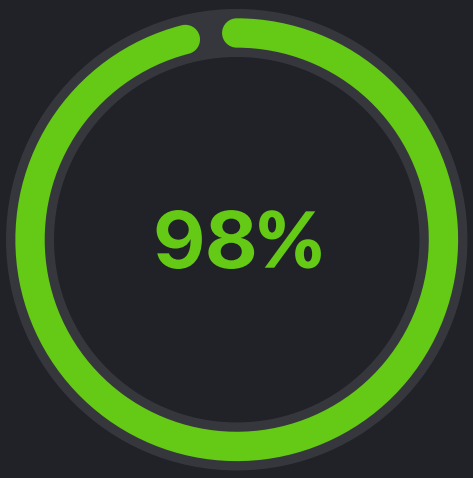
https://www.hypercycle.ai

Provided By DcentraLab Diligence on November 12, 2023

Security Audit Score

Pass

DcentraLab Diligence team has conducted an extensive audit on HyperCycle Contracts and has found the code to be in low risk level given proper deployment and multi-sig permissioning.



- Minimal Risk
- Small Risk
- Medium Risk
- High Risk

Scope

Audited Repository:

https://github.com/hypercycle-development/hypercycle-contracts

Audited Branch:

develop

Audited Commit Hash:

693a115ea048e9f38bf46b4583a79ad7952979a6

Fix Branch:

develop_swapv2

Fixes Commit Hash:

e297df92af545343c7237ab239ed57e9e91d2b85

Audited Contracts:

HyperCycleSwapV2.sol

Reviewed For Context:

- CrowdFundHYPCPoolV2.sol
- CHYPC.sol
- HyperCycleSwap.sol
- HyperCycleLicense.sol

Nomenclature Of Issues:

E - Environmental
A - Contract HyperCycleSwapV2

Contracts Architecture Overview

Detailed description of contract's purpose and interactions can be found in the audited contract file throughout the flow and more specifically in the section at L12-61.

HyperCycleSwapV2 Analysis
HyperCycleSwapV2 is a multi-functional contract made primarily to enable swaps with HYPC and CHYPC (itself). Existing backwards compatibility of this contract is based on the inheritance of the interfaces which the original HyperCycleSwap contract implemented together with the CHYPC contract.

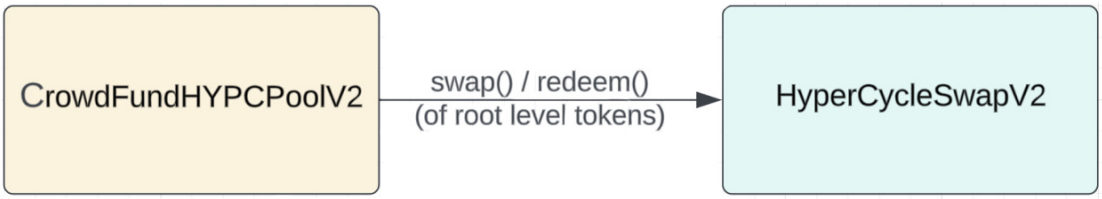
Main feature separating this version from the previous is allowing for lower levels of CHYPC to be swapped. This means that you can gather a lower amount of HYPC (2^level) in order to swap it for a lower level of CHYPC.

E.g. swap 2^17 HYPC for a CHYPC token of level 17
HyperCycleSwapV2 is a synthesis of HyperCycleSwap and CHYPC as in the previous version separate contract instances served to enable the swap of root level tokens. This implies the expansion of functionality that the new version provides aside from the architectural change of merging two entities into one.
CrowdFundHYPCPoolV2 makes the following calls on HyperCycleSwapV2:

- swap()
- redeem()
- assign()
- approve()
- getAssignment()
- nfts(0)

All of the mentioned functions are, even though some are not necessary in order to work with HyperCycleSwapV2, present in the contract in order to enable backwards compatibility with V1 integrators.

So on CrowdFundHYPCPoolV2 (or any of the other integrators of V1 protocol), HyperCycleSwapV2 should be set as both 'hypcNFT' (CHYPC) and 'swapContract'.
These contracts work only with root level CHYPC tokens, they're the ones being utilized with the proposals, and being swapped and redeemed.



HyperCycleSwapV2 example interaction with V1 integrators

This is a representation of a simple interaction between the instances. Both are static (non-upgradeable).
If users want to switch to a new version (in order to use new features), they need to get their HYPC back from previous CHYPC by redeeming on HyperCycleSwap. Then they can interact directly with the HyperCycleSwapV2 in order to get the new version of CHYPC tokens of chosen level.

Risks:

DcentraLab Diligence (DD) has performed all checks and verifications in its capacity to ascertain the safety of the code. However, it should be noted that misuse of the code, bad deployment practices, bad key management, exposing of private keys of the deployer and/or owner address and/or multi-sig signer addresses and/or fee collector address and/or any exposition of the code to malicious actors may result in an exploit of the code and loss of state and/or funds.
Furthermore, there is always a chance that other Smart Contracts code could be written and deployed to cause the provided code by DD to act outside the intended scope by the client, to the point of causing state corruption or loss of funds to the client of the users of the code.

Issues Severity Reference Table

Type

Discussion

The issue severity is dependent on design, centralization, and product specifications of the project.

Informational

This issue is not critical and does not pose an immediate threat to the functionality or security of the smart contract. It is simply an informational item that the auditors have identified and recommends addressing for best practices or to improve the overall performance of the contract.

Low

This issue is relatively minor and does not pose a significant risk to the functionality or security of the smart contract. While it is recommended to address these issues to ensure the highest level of quality and security, they are not likely to cause significant problems if left unaddressed.

Medium

This issue poses a moderate risk to the functionality or security of the smart contract. While it may not be immediately exploitable, it has the potential to cause problems in the future if left unaddressed. It is recommended to address these issues as soon as possible to prevent any potential negative impact on the contract.

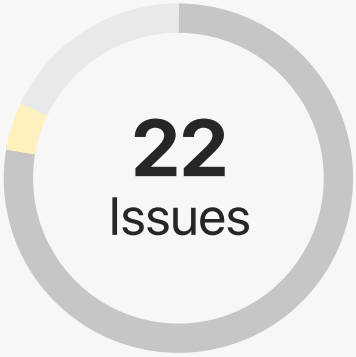
High

This issue poses a significant risk to the functionality or security of the smart contract. Addressing these issues as soon as possible is recommended to prevent any potential negative impact on the contract. Failure to address these issues could result in significant problems and potential loss of funds or other assets.

Critical

This issue poses an immediate and severe risk to the functionality or security of the smart contract. It is recommended to address these issues immediately to prevent any potential negative impact on the contract. Failure to address these issues could result in catastrophic problems and significant loss of funds or other assets.

Findings Summary



- Discussion
- Informational
- Low Risk
- Medium Risk
- High Risk
- Critical Risk

ID	Title	Severity	Status
A.1	storage load optimization	Informational	Resolved
A.2	computation optimization	Informational	Resolved
A.3	computation optimization	Informational	Resolved
A.4	lack of explicit revert messages	Informational	Resolved
A.5	typographical issues	Informational	Resolved
A.6	missing 'skipLevels' value check	Low	Resolved
A.7	index argument of splitHeldToken()	Discussion	Resolved
A.8	misleading error message	Informational	Partially Resolved
A.9	code order	Informational	Resolved
A.10	order of inheritances	Informational	Resolved
A.11	constructor arguments	Informational	Acknowledged

Findings Summary

ID	Title	Severity	Status
A.12	contract upgradeability	Discussion	Acknowledged
A.13	pausability of sensitive functions	Discussion	Acknowledged
A.14	checks-effects-interactions pattern	Informational	Resolved
A.15	missing ‘constant’ keyword	Informational	Resolved
A.16	unused import	Informational	Resolved
A.17	value comparison to boolean constant	Informational	Resolved
A.18	redundant local variable casting	Informational	Resolved
A.19	value assigned zero outside of related if statement	Informational	Resolved
A.20	redundant external call	Informational	Resolved
A.21	nft token numbers and license numbers	Discussion	Acknowledged
A.22	function addNFT()	Informational	Resolved

Complete Analysis

Local Contract Findings:

Contract: HyperCycleSwapV2.Sol

ID A.1: **Status: Resolved**

Informational | storage load optimization

Present at: HyperCycleSwapV2.sol / addRootTokens() @ L257-273

Description: '_rootData.currentRootNumber' is being accessed multiple times throughout the flow. Value is being accessed three times during a single iteration of the 'for' loop (and one time outside of it) which increases the gas spendings of the execution. Also the value of '_rootData.currentRootNumber + i' is being computed 3 times in a single loop.

Recommendation: Consider reducing the number of repeating storage loads and computations.

ID A.2: **Status: Resolved**

Informational | computation optimization

Present at: HyperCycleSwapV2.sol / splitHeldToken() @ L282-319

Description: 'level - skipLevels' computation is performed multiple times throughout the flow. As neither of 2 variables change during the flow execution, performing more than a single computation is not necessary.

Recommendation: Consider executing computation only once and storing the result in a local variable which can be used throughout the flow.

Complete Analysis

ID A.3: **Status: Resolved**

Informational | computation optimization

Present at: HyperCycleSwapV2.sol / splitHeldToken() @ L309-312

Description: 'tokenNumber * tokensToCreate' computation is performed two times throughout the single 'for' loop iteration. As neither of 2 variables undergo changes, performing more than a single computation (outside the loop) is not necessary.

Recommendation: Consider executing computation only once, outside of the loop, and store the result in a local variable which can be used throughout the flow.

ID A.4: **Status: Resolved**

Informational | lack of explicit revert messages

Present at: HyperCycleSwapV2.sol

Description: There are several sections where an error may occur but an explicit revert message is not returned. This applies to the computations where a possibility for an overflow or underflow exists.

Recommendation: Consider providing explicit revert messages in these scenarios.

Complete Analysis

ID A.5: **Status: Resolved**

Informational | typographical issues

Present at: HyperCycleSwapV2.sol

Description: There are several places with typographical issues:

- L25:97 'srings' instead of 'strings'
- L41:41 'unspliting' instead of 'unsplitting'
- L46:75 'any' instead of 'many'
- L86:9 'Errof' instead of 'Error'
- L100:54 'create' instead of 'created'
- L255:32 'transfered' instead of 'transferred'
- L350:63 'transfered' instead of 'transferred'
- L350:83 'ths' instead of 'this'
- L427:17 'wraper' instead of 'wrapper'
- L454:11 'retreve' instead of 'retrieve'
- L488:31 'compatiblity' instead of 'compatibility'

Recommendation: Consider fixing the typographical issues.

ID A.6: **Status: Resolved**

Low | missing 'skipLevels' value check

Present at: HyperCycleSwapV2.sol / splitHeldToken

Description: Based on the 'LEVEL_LIMIT' check present at line 292, 'skipLevels' argument value should never be greater than 4, as that would imply that there are more than 'LEVEL_LIMIT' (16) tokens to be created. While value of 4 may work if the level is empty, there is no scenario where greater values can be utilized.

Recommendation: Consider implementing the check.

Complete Analysis

ID A.7: **Status: Resolved**

Discussion | index argument of splitHeldToken()

Present at: HyperCycleSwapV2.sol @ L282-319

Description: Function takes token 'index' as an argument. The index value enables users to choose a specific token they want to split. Since all of the CHYPC tokens present in the same level have the same value, providing an index (having a preference) seems to be unnecessary. On the other hand if the user is supposed to have a preference of which token he wants to split, providing an index is not helpful as another user can do a split on that index before him, which results in the user splitting an undesired token. If having a preference over tokens should be established please consider letting users choose tokens by a 'tokenNumber' instead of index to ensure precision (and ensure revert if the token is not available). If having a preference is not desired, consider removing the index parameter from the flow and using the first available token of chosen level (zero index).

ID A.8: **Status: Partially Resolved**

Informational | misleading error message

Present at: HyperCycleSwapV2.sol / validHeldToken @ L205-210

Description: Modifier contains a check which reverts if the index is out of bounds with error message 'InvalidTokenLevelIndex'. This check can also revert in case that level is outside of scope (19 - 10) and when there are no available tokens on an existing level, in both cases no index would be valid.

Recommendation: Consider segregating the check and reverting with different error messages based on each case in order to simplify the debugging process and improve UX.

Complete Analysis

ID A.9: **Status: Resolved**

Informational | code order

Present at: HyperCycleSwapV2.sol

Description: Among the declared variables there are mixed constants, structs and immutable variables. As immutable variables and constants are not taking place in the contract's storage, it is inconvenient to mix them, as well as structs, with ordinary variables.

Recommendation: Consider separating the variables, immutable variables, constants and structs in order to increase the code readability and ease the view over storage layout.

ID A.10: **Status: Resolved**

Informational | order of inheritances

Present at: HyperCycleSwapV2.sol / validHeldToken @ L205-210

Description: Among the inheritances, 3 of them are affecting storage layout and 1 is not.
Three inheritances are abstract contracts affecting the layout are ERC721Enumerable, Ownable and ReentrancyGuard. One inheritance is an interface IHYPCSwapV2.
Interface is inherited in between the abstract contracts, while it is more convenient to inherit it either before or after the storage layout altering section of inheritances.

Recommendation: Consider sorting the inheritances in a more convenient order.

Complete Analysis

ID A.11: **Status:** Acknowledged

Informational | constructor arguments

Present at: HyperCycleSwapV2.sol / constructor @ L247-248

Description: Since 'startRootNumber' and 'endRootNumber' attributes of '_rootData' are known beforehand and there is seemingly no intention of changing them or initializing the contract with different values, hardcoding the values can decrease chance for a mistake during initialization of the contract.

Recommendation: Consider hardcoding the values to ensure secure initialization.

ID A.12: **Status:** Acknowledged

Discussion | contract upgradeability

Present at: HyperCycleSwapV2.sol

Description: Consider making the contract upgradeable in order to increase the flexibility of the architecture. If upgradeability flow is properly secured (ex. via providing ownership over it to a multi signature wallet) it can also provide great security benefits if the production error occurs.

ID A.13: **Status:** Acknowledged

Discussion | pausability of sensitive functions

Present at: HyperCycleSwapV2.sol

Description: Consider making the contract's most sensitive functions pausable by inheriting the pausable contract from OZ. In a case of danger, the production environment will be able to get paused until the issues have been Resolved.

Complete Analysis

ID A.14: **Status: Resolved**

Informational | checks-effects-interactions pattern

Present at: HyperCycleSwapV2.sol

Description: Make sure that logic inside of each function is adhered to the CEI pattern.

Resource: https://fravoll.github.io/solidity-patterns/checks_effects_interactions.html

ID A.15: **Status: Resolved**

Informational | missing 'constant' keyword

Present at: HyperCycleSwapV2.sol @ L144

Description: Variable 'SIX_DECIMALS' should be a constant.

Recommendation: Consider making the mentioned variable a constant by adding a missing keyword.

ID A.16: **Status: Resolved**

Informational | unused import

Present at: HyperCycleSwapV2.sol @ L9

Description: Though 'ICHYPC' interface should be inherited by the contract as it represents the synergy of the CHYPC and HyperCycleSwap, we've noticed that functions contained in it are also contained in IHyperCycleSwapV2. This makes the import unnecessary.

Recommendation: Consider removing an unused import.

Complete Analysis

ID A.17: **Status: Resolved**

Informational | value comparison to boolean constant

Present at: HyperCycleSwapV2.sol @ L215 && L224

Description: Boolean value is compared against a constant which is impractical as boolean itself can be used as a condition.

Recommendation: Consider using boolean itself as a condition (add '!' to invert it).

ID A.18: **Status: Resolved**

Informational | redundant local variable casting

Present at: HyperCycleSwapV2.sol @ L471

Description: Variable is declared in order to be used only once.

Recommendation: Consider either declaring the variable in previous lines to let it be used twice, or using the value directly.

ID A.19: **Status: Resolved**

Informational | value assigned zero outside of related if statement

Present at: HyperCycleSwapV2.sol @ L474

Description: '_targetToken.assignedNumber' is assigned zero outside of the previous 'if' statement which makes new value assignments if value is greater than zero. This means that even if the value is already zero, additional gas will be spent in order to set it again.

Recommendation: Consider moving the action inside the mentioned 'if' statement.

Complete Analysis

ID A.20: **Status: Resolved**

Informational | redundant external call

Present at: HyperCycleSwapV2.sol @ L502

Description: Function 'getAssignment' calls function 'getAssignmentString' via external call. This implies no gas spendings if the 'view' function was called by an EOA, but otherwise if called by a contract it will increase gas spendings unnecessarily.

Recommendation: Consider calling 'getAssignmentString' internally.

ID A.21: **Status: Acknowledged**

Discussion | nft token numbers and license numbers

Present at: HyperCycleSwapV2.sol @ L247-248

Description: In the constructor, values for 'startRootNumber' and 'endRootNumber' are being specified in comments. License contract documentation lets us know that CHYPC numbers should be present only on the left side of the tree (which is respected by the commented numbers) and that license numbers should persist on the right side of the tree. Though we've noticed that mentioned numbers persist in the different levels of the tree, please confirm if this is desired behavior. Other than that, both licenses and CHYPC numbers are stored in levels where more than 4096 slots are available per side of the three, so introducing flexibility with 'endRootNumber' (adding a setter) might be helpful if the supply increase becomes desired.

Complete Analysis

ID A.22: **Status: Resolved**

Informational | **function addNFT()**

Present at: HyperCycleSwapV2.sol @ L496

Description: Function 'addNFT()' stays in the HyperCycleSwapV2 as it should provide value in backwards compatibility. Since in the previous implementation of the swap contract this function was meant to be called solely by the CHYPC.sol which is a non-upgradeable singleton contract, it seemingly serves no purpose to keep this function on the CHYPC.sol in current conditions. Think about whether the mint of old CHYPC.sol should stay enabled in the new version, as this would be purposeful only in specific conditions of keeping some of the previous contracts in use.

Recommendation: Consider removing the function.

Disclaimer:

DcentraLab Diligence (DD) has provided the code to the client as is and assumes no responsibility nor legal liability for any use client may do with the code. Any and all usage and/or deployment of the code provided by DcentraLab Diligence will be done solely by the client, at the sole discretion, responsibility, risk, and legal liability of the Client, and DD will not be held accountable or liable for any loss of funds, security exploits or incidents, or any other unintended or negative outcome that may occur in relation to the code provided by DD.

This report is not, nor should be considered, an "endorsement" or "disapproval" of any particular project or team. This report is not, nor should be considered, an indication of the economics or value of any "product" or "asset" created by any team or project that contracts DD to perform a security assessment. This report does not provide any warranty or guarantee regarding the absolute bug-free nature of the technology analyzed, nor do they provide any indication of the technologies proprietors, business, business model, or legal compliance.

This report and the provided code or services as part of the SOW pertaining to this report should not be used in any way to make decisions around investment or involvement with any particular project. This report in no way provides investment advice, nor should it be leveraged as investment advice of any sort. This report represents an extensive assessing process intending to help our customers increase the quality of their code while reducing the high level of risk presented by cryptographic tokens and blockchain technology.

Blockchain technology and cryptographic assets present a high level of ongoing risk. DD's position is that each company and individual are responsible for their own due diligence and continuous security. DD's goal is to help reduce the attack vectors and the high level of variance associated with utilizing new and consistently changing technologies, and in no way claims any guarantee of security or functionality of the technology we agree to analyze.

The assessment services provided by DD are subject to dependencies and are under continuing development. You agree that your access and/or use, including but not limited to any services, code, reports, and materials, will be at your sole risk on an as-is, where-is, and as-available basis. Cryptographic tokens are emergent technologies and carry with them high levels of technical risk and uncertainty. The assessment reports could include false positives, false negatives, and other unpredictable results. The services may access, and depend upon, multiple layers of third parties.

ALL SERVICES, THE LABELS, THE ASSESSMENT REPORT, WORK PRODUCT, OR OTHER MATERIALS, OR ANY PRODUCTS OR RESULTS OF THE USE THEREOF ARE PROVIDED "AS IS" AND "AS AVAILABLE" AND WITH ALL FAULTS AND DEFECTS WITHOUT WARRANTY OF ANY KIND. TO THE MAXIMUM EXTENT PERMITTED UNDER APPLICABLE LAW, DcentraLab Diligence (DD) HEREBY DISCLAIMS ALL WARRANTIES, WHETHER EXPRESS, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE SERVICES, ASSESSMENT REPORT, OR OTHER MATERIALS. WITHOUT LIMITING THE FOREGOING, DD SPECIFICALLY DISCLAIMS

ALL IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT, AND ALL WARRANTIES ARISING FROM THE COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

WITHOUT LIMITING THE FOREGOING, DD MAKES NO WARRANTY OF ANY KIND THAT THE SERVICES, THE LABELS, THE ASSESSMENT / VERIFICATION REPORT, WORK PRODUCT, CODE OR OTHER MATERIALS, OR ANY PRODUCTS OR RESULTS OF THE USE THEREOF, WILL MEET THE CUSTOMER'S OR ANY OTHER PERSON'S REQUIREMENTS, ACHIEVE ANY INTENDED RESULT, BE COMPATIBLE OR WORK WITH ANY SOFTWARE, SYSTEM, OR OTHER SERVICES, OR BE SECURE, ACCURATE, COMPLETE, FREE OF HARMFUL CODE, OR ERROR-FREE.

WITHOUT LIMITATION TO THE DISCLAIMER HyperCycle Contracts FOREGOING, DD PROVIDES NO WARRANTY OR UNDERTAKING, AND MAKES NO REPRESENTATION OF ANY KIND THAT THE SERVICE WILL MEET THE CUSTOMER'S REQUIREMENTS, ACHIEVE ANY INTENDED RESULTS, BE COMPATIBLE OR WORK WITH ANY OTHER SOFTWARE, APPLICATIONS, SYSTEMS OR SERVICES, OPERATE WITHOUT INTERRUPTION, MEET ANY PERFORMANCE OR RELIABILITY STANDARDS OR BE ERROR-FREE OR THAT ANY ERRORS OR DEFECTS CAN OR WILL BE CORRECTED.

WITHOUT LIMITING THE FOREGOING, NEITHER DD NOR ANY OF DD'S AGENTS MAKES ANY REPRESENTATION OR WARRANTY OF ANY KIND, EXPRESS OR IMPLIED AS TO THE ACCURACY, RELIABILITY, OR CURRENCY OF ANY INFORMATION, CODE OR CONTENT PROVIDED THROUGH THE SERVICE. DD WILL ASSUME NO LIABILITY OR RESPONSIBILITY FOR (I) ANY ERRORS, MISTAKES, OR INACCURACIES OF CONTENT AND MATERIALS OR FOR ANY LOSS OR DAMAGE OF ANY KIND INCURRED AS A RESULT OF THE USE OF ANY CONTENT OR CODE, OR (II) ANY PERSONAL INJURY OR PROPERTY DAMAGE, OF ANY NATURE WHATSOEVER, RESULTING FROM CUSTOMER'S ACCESS TO OR USE OF THE SERVICES, CODE, ASSESSMENT REPORT, OR OTHER MATERIALS.

ALL THIRD-PARTY MATERIALS ARE PROVIDED "AS IS," AND ANY REPRESENTATION OR WARRANTY OF OR CONCERNING ANY THIRD-PARTY MATERIALS IS STRICTLY BETWEEN THE CUSTOMER AND THE THIRD-PARTY
OWNER OR DISTRIBUTOR OF THE THIRD-PARTY MATERIALS. THE SERVICES, ASSESSMENT REPORT, AND ANY OTHER MATERIALS HEREUNDER ARE SOLELY PROVIDED TO THE CUSTOMER AND MAY NOT BE RELIED ON BY ANY OTHER PERSON OR FOR ANY PURPOSE NOT SPECIFICALLY IDENTIFIED IN THIS AGREEMENT, NOR MAY COPIES BE DELIVERED TO, ANY OTHER PERSON WITHOUT DD'S PRIOR WRITTEN CONSENT IN EACH INSTANCE.

NO THIRD PARTY OR ANYONE ACTING ON BEHALF OF ANY THEREOF, SHALL BE A THIRD PARTY OR OTHER BENEFICIARY OF SUCH SERVICES, ASSESSMENT REPORT, AND ANY ACCOMPANYING MATERIALS AND NO SUCH THIRD PARTY SHALL HAVE ANY RIGHTS OF CONTRIBUTION AGAINST DD WITH RESPECT TO SUCH SERVICES, ASSESSMENT REPORT, AND ANY ACCOMPANYING MATERIALS. THE REPRESENTATIONS AND WARRANTIES OF DD CONTAINED IN THIS AGREEMENT ARE SOLELY FOR THE BENEFIT OF THE CUSTOMER. ACCORDINGLY, NO THIRD PARTY OR ANYONE ACTING ON BEHALF OF ANY THEREOF, SHALL BE A THIRD PARTY OR OTHER BENEFICIARY OF SUCH REPRESENTATIONS AND WARRANTIES AND NO SUCH THIRD PARTY SHALL HAVE ANY RIGHTS OF CONTRIBUTION AGAINST DD WITH RESPECT TO SUCH
REPRESENTATIONS OR WARRANTIES OR ANY MATTER SUBJECT TO OR RESULTING IN INDEMNIFICATION UNDER THIS AGREEMENT OR OTHERWISE.
FOR AVOIDANCE OF DOUBT, THE SERVICES, INCLUDING ANY ASSOCIATED ASSESSMENT REPORTS, CODE, OR MATERIALS, SHALL NOT BE CONSIDERED OR RELIED UPON AS ANY FORM OF FINANCIAL, TAX, LEGAL,
REGULATORY, OR OTHER ADVICE.

dcentralab.com/diligence

DcentraLab Diligence

Provided By DcentraLab Diligence on November 12, 2023