Information Security

Indiana University Health, Inc. Smart Contract Security Requirements

These are minimum requirements required by IU Health's Information Security Program for technologies used on behalf of IU Health to provide intelligent contracts, colloquially known as Smart Contracts, on behalf of IU Health. We recognize that this is a technology that can provide significant benefits to IU Health through their use to provide both contracts and automated responses to changes. The purposes of these requirements are to ensure that the underlying technologies utilized on behalf of IU Health are properly assessed and monitored for vulnerabilities that may compromise their integrity, that the contracts work as intended and designed, that IU Health has verification and validation that they are able to work in their intended environment, and that IU Health has reasonable and appropriate controls and measures to prevent intentional or unintentional misuse. For the purposes of below, (i) each reference to "Agreement" shall be defined to include the BAA and Service Agreement, (ii) each reference to "Provider" shall be defined to include Business Associate, and (iii) each reference to "IU Health" shall be defined to include Covered Entity.

i. Each Smart Contract needs to have a defined written use case, preferably in Unified Modeling Language (UML) format or a similar format that defines:
    a. Sender(s)
    b. Recipient(s)
    c. Input(s)
    d. Output(s)
    e. Actor(s)
        i. Blockchain/Distributed Ledger Technology (DLT) system that it will execute on
        ii. Other Contract(s)
        iii. Oracles, which are external resources that can trigger contract execution
        iv. Interfacing systems and methods
        v. Externally accessible media or files.
    f. Execution Conditions
    g. Preconditions
    h. Postconditions
    i. Programming Language Used
ii. All Smart Contracts need to be validated by a third-party Smart Contracts Validation Service which performs security and integrity testing on contracts to ensure that the contracts perform as intended. Of note, EY is specifically excluded due to their conducting financial services audits for IU Health.
    a. Contracts need to be tested by a neutral third party that does not have an interest in the contract or its outcomes or performs audit services for IU Health.
    b. Contracts need to be tested in a separate environment from the production environment.
        i. Both Business Associate and IU Health will provide non-production systems to the third-party validator to test use cases.
    c. As part of the validation testing, contracts need to be tested for the following:
        i. Race Conditions. This is when multiple concurrently executing contracts achieve different results based on timing and execution, and execution becomes dependent upon the timing, not the instructions within of the contract.
        ii. Reentrancy Attacks. This is when a smart contract can be interrupted in the middle of its execution and instructions to be used for the purpose of exploiting code vulnerabilities to transfer assets or resources outside the bounds of the contract to another party.
        iii. Concurrency Testing. Contracts need to be tested to ensure that multiple simultaneous running copies do not cause race conditions, reentrancy attacks, or behavior outside intended conditions.

iv. Timestamp Dependencies. Contracts need to be tested to ensure they are not dependent upon timestamps for conditions of execution, as this can potentially cause a Race Condition, Reentrancy Attack, or Concurrency issue leading to execution outside of defined bounds and exploitable vulnerabilities.
v. Resource Usage. Contracts will be tested to ensure that they execute instructions consistently and use a defined range of resources.
vi. Access to external resources and Oracles must be tested as part of the validation process.
vii. Upstream and downstream data interchange that occurs as part of the contract must be validated.
viii. Contracts must use validated Application Program Interfaces (APIs) and methods to communicate with upstream and downstream systems.

d. Media or files associated with Smart Contracts need to be stored on globally accessible media using InterPlanetary File System (IPFS).
    i. Pinning, which is the permanent storage of resources on IPFS, needs to be enabled for the lifetime of the contract for all associated media or files.
    ii. DNSLink, which uses Domain Name Services to map a domain name to an IPFS resources, needs to be configured and enabled for resources utilized in Smart Contracts.
    iii. When the hashing algorithm used on IPFS is changed, which changes the resource name, the DNSLink name must be updated.
    iv. Sensitive files must be encrypted using the public key(s) of the recipient(s).
e. Audit logging of all activities must occur, preferably using a Blockchain-based system to ensure their integrity.
    i. The audit log system utilized must meet the requirements in the Indiana University Health Verification and Validation using Distributed Computing Requirements appendix.
f. Zero-knowledge proofs must be utilized for Smart Contracts that contain sensitive or regulated information.
g. Blockchain systems and networks that host and execute these contracts must follow the security requirements in the Indiana University Health Verification and Validation using Distributed Computing Requirements appendix.