정 보 보 호 학 회 지
제34권 제4호, 2024. 8

이종 분산원장기술(DLT) 시스템 간 상호운용성을 위한 내부 처리 방안 및 DLT 게이트웨이 보안 요구사항

김 미 경*, 황 정 연**, 김 영 진***

요 약

DLT(Distributed Ledger Technology) 기술의 빠른 발전과 광범위한 채택으로 인해 단일 DLT 시스템만으로는 다양한 분산 응용 프로그램(DApp)의 요구를 충족하기 어렵습니다. 이에 따라 DLT 시스템 상호운용성 솔루션에 대한 수요가 증가하고 있습니다. 이러한 상황에서 DLT 게이트웨이는 표준 프로토콜을 준수하여 다양한 분산 응용 프로그램 및 DLT 시스템 간의 원활한 상호작용을 가능하게 합니다. 본 논문에서는 이러한 DLT 게이트웨이 기반 상호운용성 제공 시에 처리할 내부 처리 방안을 제시합니다. 또한 발생가능한 보안 위협을 분석하고, 안전하고 신뢰할 수 있는 운영을 보장하기 위한 보안 요구사항을 제시합니다.

Ⅰ. 서 론

DLT 기술은 그 잠재력과 함께 다양한 산업 분야에서 광범위하게 채택되고 있으나 단일 DLT 시스템만으로는 여러 분산 응용 프로그램의 요구를 충분히 충족하기 어렵습니다. 이로 인해 DLT 상호운용성 솔루션 필요성이 대두되고 있습니다. DLT 상호운용성은 서로 다른 DLT 네트워크 간의 데이터 및 자산 전송을 가능하게 하며, 이를 통해 더 많은 분산 응용 프로그램이 다양한 DLT 시스템의 기능을 활용할 수 있게 합니다.

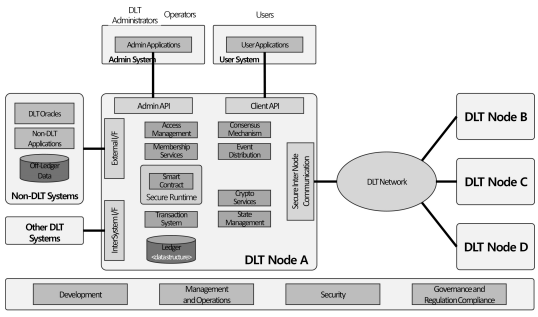
이러한 DLT 시스템 간 상호운용성을 보장하기 위해 DLT 게이트웨이가 중요한 역할을 합니다. DLT 게이트웨이는 표준 프로토콜 준수, 다양한 분산 응용 프로그램 및 DLT 시스템 간 원활한 상호작용을 지원합니다. 이러한 DLT 게이트웨이는 퍼블릭 및 프라이빗 블록체인 연결 지원뿐만 아니라, 데이터 전송, 자산 전송 및 자산 교환 등 다양한 상호운용을 지원합니다.

본 논문에서는 DLT 게이트웨이 기반 상호운용성을 제공하는 DLT 게이트웨이의 내부 처리 프로세스 및 상호연동 시 발생할 수 있는 보안 위협을 분석하고, 안전하고 신뢰할 수 있는 운영 보장을 위한 보안 요구사항을 제시하여, 보다 안전한 DLT 시스템 운용 환경을 구축하는 데 기여하고자 합니다.

Ⅱ. DLT 상호운용성

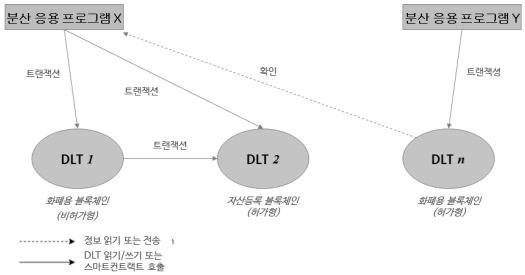
서로 다른 DLT 시스템들은 각기 다른 프로토콜과 트랜잭션 구조를 가지고 DLT 네트워크를 구성하여, 상호 간의 데이터 및 자산 교환이 어렵습니다.

DLT 상호운용성은 서로 다른 DLT 네트워크 간의 데이터와 자산의 원활한 교환 및 상호작용을 가능하게



〔그림 1〕 DLT 시스템 기능 요소

이 논문은 2024년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구 내용임(No.RS-2024-00438796, 이종 블록체인 시스템 간 표준 오퍼레이션을 통한 서비스 데이터 융합 및 상호 운용 기술 개발).
* ㈜드림시큐리티 정보보안연구소 (책임, mg.kim@dreamsecurity.com)
** 성신여자대학교 수리통계데이터사이언스학부 (교수, jyhwang@sungshin.ac.kr)
*** ㈜드림시큐리티 정보보안연구소 (상무, yjkim@dreamsecurity.com)



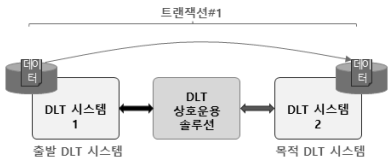
〔그림 2〕 분산 응용 프로그램을 위한 DLT 상호운용 개념

하는 능력입니다. 이는 다양한 DLT 시스템이 서로 자산을 공유하며, 거래를 할 수 있도록 합니다. 상호운용성은 DLT 기술의 확장성과 실용성을 높이며, 여러 DLT 네트워크가 통합된 생태계를 형성하는 데 중요한 역할을 합니다.

2.1. DLT 상호운용성 유형

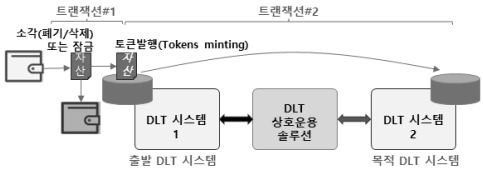
DLT 상호운용성 유형은 데이터 전송, 자산 전송 및 교환 등 3가지 유형으로 구분됩니다.

1) 데이터 전송(Data Transfer): 한 DLT 네트워크에서 다른 DLT 네트워크로 데이터를 전송하는 능력입니다. 예를 들어, 한 DLT에서 생성된 트랜잭션 데이터를 다른 DLT에서 참조할 수 있습니다.



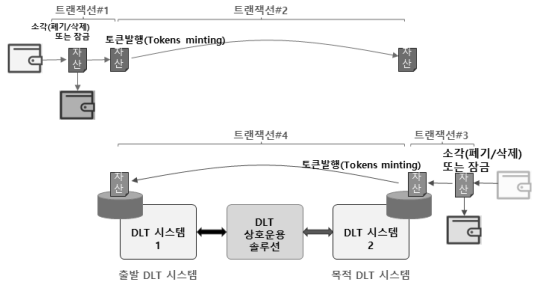
〔그림 3〕 “데이터 전송” 상호운용성 유형

2) 자산 전송(Asset Transfer): 한 DLT 네트워크에서 다른 DLT 네트워크로 자산을 이동하는 능력입니다. 예를 들어, 한 DLT에서 발행된 토큰을 다른 DLT로 이동할 수 있습니다.



〔그림 4〕 “자산 전송” 상호운용성 유형

3) 자산 교환(Asset Exchange): 서로 다른 DLT 네트워크 간에 자산을 교환하는 능력입니다. 이는 두 DLT 간의 자산을 상호 교환하는 거래를 의미합니다.



〔그림 5〕 “자산 교환” 상호운용성 유형

2.2. DLT 상호운용성 구현

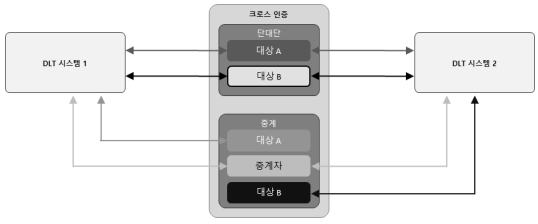
DLT 상호운용성 구현은 DLT 오라클, 크로스 인증 등 2가지 방식으로 이뤄집니다.

1) DLT 오라클 (DLT Oracles): 오라클은 외부 데이터를 DLT 시스템에 가져오는 서비스로, 데이터 전송 및 자산 전송 모드에서 사용됩니다. 오라클은 푸시 기반 또는 풀 기반으로 동작할 수 있습니다.

- 푸시 기반 오라클: 외부 시스템의 데이터를 명시적인 요청 없이 DLT 시스템으로 전달합니다.
- 풀 기반 오라클: DLT 시스템의 요청에 따라 외부 시스템의 데이터를 가져옵니다.

〔그림 6〕 DLT 오라클 상호운용성 구현

2) 크로스 인증 (Cross Authentication): 자산 교환 모드에서 사용되며, 서로 다른 DLT 시스템 간 자산



〔그림 7〕 크로스 인증 상호운용성 구현

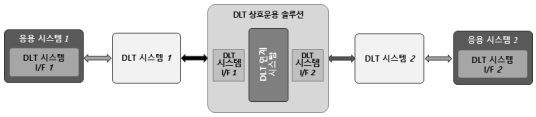
교환 위해 양쪽에서 사용자 인증을 수행합니다. 이는 P2P 방식과 중개자를 통한 방식으로 나뉩니다.

- P2P 자산 교환: 중개자 없이 직접 자산을 교환합니다.
- 중개자 기반 자산 교환: 신뢰할 수 있는 중개자를 통해 자산을 교환합니다.

2.3. DLT 상호운용성 구조

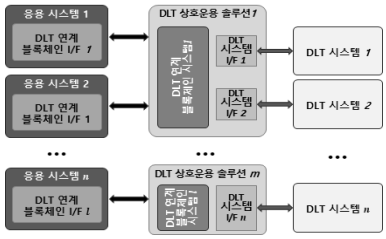
DLT 상호운용성을 지원하는 구조는 다음과 같은 방식으로 운용될 수 있습니다.

1) 직접 DLT 상호연결 (Direct Blockchain Interconnection): DLT 간의 직접 연결을 통해 상호운용성을 제공합니다.



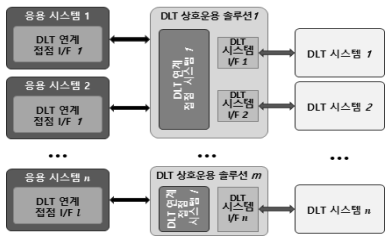
〔그림 8〕 직접 DLT 상호연결에 의한 상호운용성 구조

2) 상호연결 DLT 기반 접근 (Interconnected Blockchain-Based Approaches): 상위 DLT를 통해 하위 DLT 간의 상호운용성을 제공합니다.



〔그림 9〕 상호연결 DLT 기반 접근에 의한 상호운용성 구조

3) 응용 계층 상호연결 접근 (Application-Layer Interconnected Approaches): 표준화된 데이터 형식, API 인터페이스 및 프로토콜을 사용하여 응용 계층에서 상호운용성을 제공합니다.



〔그림 10〕 응용 계층 상호연결 접근에 의한 상호운용성 구조

Ⅲ. DLT 게이트웨이

DLT 게이트웨이는 응용 계층 상호연결 접근 상호운용성 구조에서 이종 DLT 시스템 간의 원활한 상호작용을 가능하게 하는 중심 허브 역할을 합니다. DLT 게이트웨이는 표준화된 인터페이스와 프로토콜을 준수하여 분산 응용 프로그램이 DLT 시스템 간 데이터 및 디지털 자산의 원활한 전송 및 교환을 가능하도록 지원합니다.

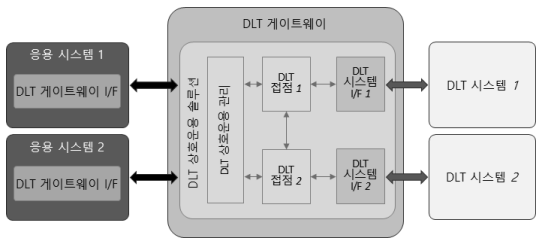
3.1. DLT 게이트웨이의 기능 아키텍처

DLT 게이트웨이는 다음과 같은 주요 구성 요소로 이루어져 있습니다.

1) DLT 접점(DLT Contact): DLT 게이트웨이와 연결된 DLT 시스템 간의 통신 채널을 설정하여, 상호운용성을 지원합니다. DLT 접점은 각 DLT 시스템의 인터페이스를 통해 DLT 시스템 간의 원활한 상호작용을 가능하게 합니다.

2) DLT 상호운용 관리(DIM): 다양한 DLT 시스템 간의 데이터 및 거래를 조정하고 관리합니다. DIM은 DLT 시스템의 이름, 합의 알고리즘, 암호화폐 이름 및 DLT 접점의 IP 주소와 같은 중요한 정보를 관리하며, 표준화된 공통 운영 절차(COP)를 DLT 접점에 배포하여 상호운용성을 지원합니다.

3) DLT 게이트웨이 인터페이스(DLT Gateway I/F): 응용 프로그램이 DLT 게이트웨이에 접근할 수 있도록 하는 진입점입니다. 이를 통해 분산 응용 프로그램은 여러 DLT 시스템과 상호작용하여 다양한 기능을 활용할 수 있습니다.



〔그림 11〕 DLT 게이트웨이 기능 아키텍처

4) DLT 시스템 인터페이스(DLT System I/F): 특정 DLT 시스템과의 직접 연결을 보장합니다. 이를 통해 데이터 및 자산의 교환이 가능하며, 각 DLT 시스템의 거래 프로토콜을 준수합니다.

3.2. DLT 게이트웨이의 운영 절차

DLT 게이트웨이를 통한 원장 데이터 공유 과정은 다음과 같이 이루어집니다.

1) DLT 접점-1과 DLT 접점-2는 DLT 시스템-1과 DLT 시스템-2의 정보를 DIM에 등록합니다. 여기에는 DLT 이름, 합의 알고리즘 이름, 암호화폐 이름 및 DLT 접점의 IP 주소가 포함됩니다.

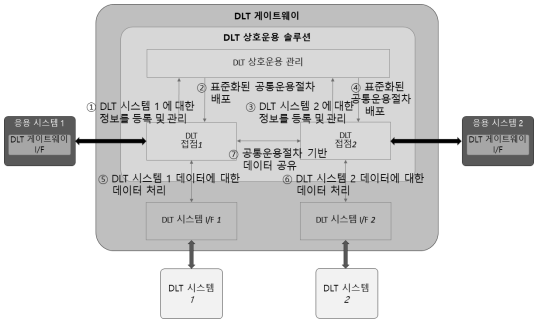
2) DIM은 공통 운영 절차(COP)를 DLT 접점-1과 DLT 접점-2에 배포하여 상호운용성을 보장합니다.

3) DLT 접점-1과 DLT 접점-2는 상호 인증 및 검증을 수행하여 신뢰할 수 있는 연결을 설정합니다.

4) DLT 접점-1은 DLT 시스템-2의 원장 구조에 대한 요청을 시작하고, DLT 접점-2는 요청된 원장 데이터를 DLT 시스템-2에서 DLT 접점-1로 전달합니다.

5) DLT 접점-1은 수신된 원장 데이터를 변환하여 DLT 시스템-1의 원장 구조에 맞게 조정한 후, 이를 DLT 시스템-1에 저장합니다.

6) 마지막으로 DLT 접점-1은 변환된 원장 데이터와 원본 데이터를 삭제하여 데이터의 무결성과 일관성을 유지합니다.



〔그림 12〕 DLT 게이트웨이 운영 절차

이와 같이 운용되는 DLT 게이트웨이에서의 암호화폐 송금은 다음과 같이 처리됩니다.

1) 자산 전송 요청 초기화 : 송신자의 지갑이 포함된 DLT 시스템 1에서 자산 전송 요청을 생성합니다. 이 요청은 송신자의 지갑과 이체 대리인의 지갑을 통해 이루어집니다.

2) DLT 접점 시스템 1에서의 처리 : DLT 시스템 1에서 전송 요청을 수신한 DLT 접점 시스템 1은 송신자의 자산을 잠금 상태로 설정합니다. 이후, 자산을 DLT 시스템 2로 전송할 준비를 합니다.

3) DLT 상호운용 관리 시스템(DIMS)에서의 조정 : DLT 상호운용 관리 시스템(DIMS)은 DLT 시스템 1과 DLT 시스템 2 간의 전송을 조정합니다. DIMS는 DLT 시스템 1에서 받은 자산 전송 요청을 DLT 시스템 2로 전달합니다.

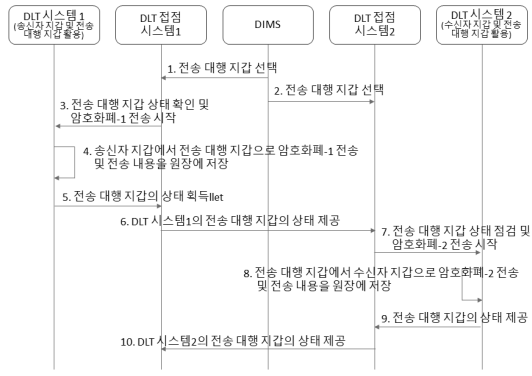
4) DLT 접점 시스템 2에서의 수신 준비 : DLT 접점 시스템 2는 DIMS로부터 전송 요청을 수신합니다. 이후, DLT 시스템 2에서 자산을 수신할 준비를 합니다.

5) 자산 수신 확인 : 수신자의 지갑이 포함된 DLT 시스템 2는 자산 수신을 확인합니다. 이 과정에서 수신자의 지갑과 이체 대리인의 지갑을 통해 자산이 성공적으로 전송되고 수신자의 지갑에 반영됩니다.

6) 원장 데이터 변환 및 저장 : DLT 접점 시스템 1은 수신된 원장 데이터를 변환하여 DLT 시스템 1의 원장 구조에 맞게 조정한 후, 이를 DLT 시스템 1에 저장합니다.

7) 데이터 무결성 및 일관성 유지 : 마지막으로 DLT 접점 시스템 1은 변환된 원장 데이터와 원본 데이터를 삭제하여 데이터의 무결성과 일관성을 유지합니다.

이와 같은 절차를 통해 DLT 게이트웨이는 다양한 DLT 시스템 간의 자산 전송을 안전하고 효율적으로 수행할 수 있습니다. 이 과정에서 각 시스템 간의 상호 인증 및 검증이 이루어지며, 데이터와 자산의 기밀성과



〔그림 13〕 DLT 게이트웨이 기반 암호화폐 송금 처리 흐름

무결성이 보장됩니다.

3.3. DLT 게이트웨이 내부 처리 서비스

DLT 게이트웨이는 서로 다른 DLT 시스템 간의 상호운용성을 보장하기 위해 복잡한 내부 처리 서비스를 수행합니다. DLT 게이트웨이의 내부 처리 서비스는 [표 1]과 같은 일련의 단계를 처리합니다.

DLT 게이트웨이의 내부 처리 서비스는 이종 DLT 시스템 간의 트랜잭션을 완결하는 데 필수적입니다. 다만 위의 단계는 DLT 유형 및 상호연동 대상 Asset에 따라 생략하는 절차도 존재합니다. 각 단계는 트랜잭션의 일관성, 원자성, 보안성, 무결성을 보장하기 위해 신중하게 설계되어야 합니다.

〔표 1〕 DLT 게이트웨이 서비스 내부 처리 단계

번호	처리 단계	처리 설명
1	연계 DLT 정보 확인	상호 연동될 대상 DLT의 정보(DLT 유형, 합의알고리즘, DLT 운영 속성 정보 등)를 확인합니다.
2	Asset Type 체크	처리될 자산의 유형을 확인합니다. 코인, 토큰, 메시지 데이터 등 자산 유형에 따라 처리 방식을 결정합니다.
3	DLT 연동 방향성 체크	DLT 간의 연동 방향성을 확인합니다. 퍼블릭 DLT에서 프라이빗 DLT로 상호연동 등 다양한 연동 방향을 검토합니다.
4	정책 체크 (강제적 처리 여부)	트랜잭션 처리 시 적용될 정책을 확인합니다. 합리적 기술적 우선순위가 아닌 강제적 처리가 필요한 지 여부를 검토합니다.
5	DLT 합의 알고리즘 우선순위 체크	DLT 간의 합의 알고리즘을 비교하여 우선순위를 결정합니다. 합의알고리즘의 탈중앙성, 보안 수준 등으로 우선순위 결정 합니다.
6	트랜잭션 액션 타입 확인	트랜잭션의 액션 타입을 확인합니다. 전송(Transfer)과 교환(Exchange) 등의 액션 타입을 검토합니다.
7	상호작용 패턴 체크	DLT 간의 상호작용 패턴을 확인합니다. 읽기-쓰기(RW)와 쓰기-쓰기(WW) 등의 패턴을 검토합니다.
8	트랜잭션 처리	모든 검토를 마친 후 트랜잭션을 처리합니다. 필요한 경우 Lock 처리를 통해 트랜잭션의 완결성을 보장합니다.

Ⅳ. DLT 게이트웨이 보안 위협

이종 DLT 시스템의 상호운용성을 제공하는 DLT 게이트웨이는 복잡한 운영 구조로 인해 다양한 보안 위협에 직면할 수 있습니다. DLT 게이트웨이 운영 구조의 주요 취약 영역은 다음과 같이 분류할 수 있습니다.

- DLT 게이트웨이 인터페이스의 취약점
- DLT 시스템 인터페이스의 취약점
- DLT 접점의 취약점
- DLT 게이트웨이 내부 처리 위협

4.1. DLT 게이트웨이 인터페이스의 취약점

DLT 게이트웨이 인터페이스(I/F)에서의 취약점은 애플리케이션과 DLT 게이트웨이 간의 자산 교환에서 발생하는 보안 격차를 포함합니다. 이러한 취약점은 a 자산의 진입 및 출구 지점으로서 주요 보안 침해 대상이 됩니다. 이러한 취약점을 해결하기 위해서는 보안 프로토콜 부족, 강력한 인증 메커니즘의 부족, 인터페이스 설계 자체의 잠재적 약점을 식별하는 것이 중요합니다. 구체적인 취약점 예시는 [표 2]와 같습니다.

〔표 2〕 DLT 게이트웨이 인터페이스 취약점

번호	명칭	설명
1	무단 접근	인증 메커니즘이 부족해 공격자가 시스템에 접근할 수 있습니다.
2	데이터 인터셉션	데이터 전송 중 가로채기 당할 위험이 있습니다.
3	API 악용	보안이 취약한 API를 통해 시스템이 악용될 수 있습니다.
4	서비스 거부(DoS) 공격	대량의 요청으로 시스템 가용성을 저하시킬 수 있습니다.
5	입력 유효성 검사 부족	적절한 검증 없으면 주입 공격(Injection,XSS 등) 가능합니다.

4.2. DLT 시스템 인터페이스의 취약점

DLT 시스템 인터페이스(I/F)는 타겟 DLT 시스템과의 직접 연결을 보장합니다. 이 인터페이스의 특정 취약점은 [표 3]과 같습니다.

〔표 3〕 DLT 시스템 인터페이스 취약점

번호	명칭	설명
1	프로토콜 불일치	서로 다른 프로토콜 사용으로 인한 통신 문제가 있을 수 있습니다.
2	중간자 공격	통신 중간에서 데이터가 탈취될 수 있습니다.
3	무단 DLT 시스템 접근	인증이 취약해 시스템 접근이 가능합니다.
4	데이터 형식 불일치	서로 다른 데이터 형식으로 인한 변환 문제가 있을 수 있습니다.
5	버전 호환성 문제	서로 다른 버전의 DLT 시스템 간 호환성 문제가 있을 수 있습니다.

4.3. DLT 접점의 취약점

DLT 접점은 이종 DLT 시스템 간의 상호운용성을 보장하는 중요한 역할을 하며, 통신 및 데이터 교환을 촉진합니다. 이러한 접점에서 발생할 수 있는 주요 취약점은 [표 4]와 같습니다.

〔표 4〕 DLT 시스템 인터페이스 취약점

번호	명칭	설명
1	구성 설정 조작	설정이 변경되어 시스템이 오작동할 수 있습니다.
2	신원 스푸핑	공격자가 신원을 위조해 접근할 수 있습니다.
3	운영 매개변수 조작	매개변수가 변경되어 시스템이 오작동할 수 있습니다.
4	데이터 변환 취약점	데이터 변환 과정에서 발생하는 보안 문제가 있을 수 있습니다.
5	통신 채널 공격	통신 채널이 공격당해 데이터가 노출될 수 있습니다.
6	공통 운영의 무단 접근	공통 운영에 무단 접근해 시스템을 조작할 수 있습니다.
7	원장 데이터 노출	원장 데이터가 노출될 수 있습니다.
8	동기화 공격	시스템 동기화 과정에서 발생하는 보안 문제가 있을 수 있습니다.
9	API 취약점	보안이 취약한 API를 통한 공격을 수행할 수 있습니다.
10	인증 메커니즘 부족	인증 메커니즘이 복잡성으로 인해 무단 접근이 가능할 수 있습니다.

4.4. DLT 게이트웨이 내부 처리 위협

DLT 게이트웨이 내부 처리 위협은 DLT 게이트웨이 내에서 데이터가 처리되고 관리되는 방식에서 발생하는 취약점을 포함합니다. 이러한 위협은 데이터 변환 절차, 여러 DLT 시스템의 통합, 상호운용 관리 구성 요소의 잠재적 결함 등으로부터 발생할 수 있습니다. 주요 취약점은 [표 5]와 같습니다.

〔표 5〕 DLT 게이트웨이 내부 처리 위협

번호	명칭	설명
1	데이터 손상	데이터 처리 중 손상될 수 있습니다.
2	무단 데이터 조작	데이터가 무단으로 조작될 수 있습니다.
3	DIM 구성 요소의 위협	DIM 구성 요소가 공격당할 수 있습니다.
4	공통 운영의 무결성	공통 운영 데이터의 무결성이 훼손될 수 있습니다.
5	구성 및 업데이트 취약점	시스템 구성 및 업데이트 과정에서 발생하는 보안 문제.
6	중앙 집중화 위험	시스템이 중앙 집중화되어 단일 실패 지점이 될 수 있습니다.
7	DLT 정보 관리 취약점	DLT 정보 관리 과정에서 발생하는 보안 문제.
8	감사 기록 조작	감사 기록이 조작될 수 있습니다.
9	자원 소모 공격	시스템 자원을 소모시켜 서비스 가용성을 저하시킬 수 있습니다.
10	권한 상승	권한이 상승되어 시스템 접근이 가능합니다.

Ⅴ. DLT 게이트웨이 보안 요구 사항

DLT 게이트웨이의 보안 요구사항은 DLT 게이트웨이 인터페이스 취약점, DLT 시스템 인터페이스 취약점, DLT 접점 취약점, 내부 처리 위협 등 네 가지 주요 영역에서 발생하는 취약점을 완화하는 데 중점을 둡니다. 이러한 보안 조치는 이종 DLT 시스템과 상호운용성 솔루션 간의 안전한 데이터 교환과 무결성을 보장하기 위해 필수적입니다. 이러한 보안 요구사항을 구현함으로써 조직은 DLT 게이트웨이의 보안 상태를 강화하고 이종 DLT 시스템 간의 안전하고 신뢰할 수 있는 상호운용성을 보장할 수 있습니다.

5.1. DLT 게이트웨이 인터페이스 보안 요구 사항

DLT 게이트웨이 인터페이스와 관련된 취약점을 완화하기 위한 보안 요구사항은 안전한 통신 및 데이터 처리를 보장하는 데 중점을 둡니다. [표 6]은 DLT 게이트웨이 인터페이스 취약점에 대한 보안 요구사항을 자세히 설명한 것입니다.

〔표 6〕 DLT 게이트웨이 인터페이스 보안 요구 사항

번호	보안 요구사항	설명
1	인증 및 접근 통제	사용자 및 응용 프로그램의 신원을 확인하기 위해 강력한 인증 메커니즘과 접근 통제 정책을 구현
2	데이터 기밀성 및 무결성	전송 및 저장 중 암호화 및 암호화 기술을 사용하여 데이터의 기밀성과 무결성을 보장
3	안전한 통신 채널	DLT 게이트웨이와 참여 엔티티 간의 데이터 교환을 보호하기 위해 전송 계층 보안(TLS)과 같은 안전한 통신 프로토콜을 설정
4	감사 및 로깅	활동을 모니터링하고 추적 가능성을 촉진하며 무단 또는 악의적인 행동을 감지하기 위한 포괄적인 감사 및 로깅 메커니즘을 구현
5	입력 검증	주입 공격을 방지하고 데이터 무결성을 유지하기 위해 적절한 입력 검증을 보장

5.2. DLT 시스템 인터페이스 보안 요구 사항

DLT 시스템 인터페이스의 취약점을 완화하기 위한 보안 요구사항은 [표 7]과 같습니다.

〔표 7〕 DLT 시스템 인터페이스 보안 요구 사항

번호	보안 요구사항	설명
1	프로토콜 검증 및 호환성	데이터 손상 및 시스템 오류를 방지하기 위해 프로토콜이 호환되고 적절히 검증되었는지 확인
2	안전한 통신 채널	데이터 무결성을 보호하고 중간자 공격을 방지하기 위해 안전한 통신 프로토콜을 설정
3	접근 통제 및 인증	무단 접근을 방지하기 위해 엄격한 접근 통제 및 인증 메커니즘을 구현
4	데이터 형식 검증	다른 데이터 형식을 적절히 처리하고 검증하여 데이터 무결성을 유지
5	버전 제어 및 호환성 관리	호환성을 보장하고 보안 격차를 방지하기 위해 버전 제어를 관리

5.3. DLT 접점 보안 요구 사항

DLT 접점에서 발생할 수 있는 보안 위협을 해결하기 위해 [표 8]과 같은 보안 요구사항을 적용해야 합니다.

〔표 8〕 DLT 접점 보안 요구 사항

번호	보안 요구사항	설명
1	구성 관리	안전한 업데이트 및 검증 메커니즘을 통해 무단 구성 변경 방지
2	강력한 인증 및 신원 확인	신원 스푸핑을 방지하기 위해 다중 인증 및 디지털 인증서 사용
3	매개변수 검증 및 모니터링	무단 조작을 방지하기 위해 운영 매개변수의 적절한 검증 및 모니터링 보장
4	안전한 데이터 변환	다양한 DLT 시스템 간의 변환 과정에서 데이터 무결성 보호
5	통신 채널 보안	DLT 접점 간의 통신 채널을 보호하여 데이터 인터셉션 및 변조 방지
6	공통 운영에 대한 무단 접근 방지	공통 운영을 안전하게 관리하고 무단 접근을 방지
7	원장 데이터 노출 방지	원장 데이터 접근 및 저장 과정에서 민감한 정보 노출 방지
8	동기화 공격 방지	DLT 접점 간의 동기화 과정에서 데이터 무결성 보장
9	API 취약점 보안	DLT 시스템과의 상호작용을 위한 API의 보안 강화
10	인증 메커니즘 강화	DLT 접점과 DLT 시스템 간의 인증 메커니즘 강화

5.4. DLT 게이트웨이 내부 처리 보안 요구 사항

DLT 게이트웨이의 내부 처리 과정에서 발생할 수 있는 보안 위협을 해결하기 위해 [표 9]와 같은 보안 요구사항을 적용해야 합니다.

〔표 9〕 DLT 게이트웨이 내부 처리 보안 요구 사항

번호	보안 요구사항	설명
1	데이터 손상 방지	데이터 처리 파이프라인의 오류 또는 취약점으로 인해 데이터 손상을 방지
2	무단 데이터 조작 방지	약한 접근 통제 또는 보안 격차로 인해 무단 데이터 조작 방지
3	DIM 구성 요소 보안	DIM의 취약점으로 인해 발생할 수 있는 무단 접근 또는 조작 방지

번호	보안 요구사항	설명
4	공통 운영 무결성 보장	공통 운영에 대한 무결성 위협 방지
5	구성 및 업데이트 보안	DIM의 구성 또는 업데이트 과정에서 발생할 수 있는 취약점 방지
6	중앙 집중화 위험 완화	DIM의 중앙 집중화로 인한 단일 실패 지점의 위험 완화
7	DLT 정보 관리 보안	DLT 시스템 정보의 관리가 약하여 발생할 수 있는 무단 공개 방지
8	감사 기록 무결성 보장	DIM이 유지하는 감사 로그의 무결성 위협 방지
9	자원 소모 공격 방지	악의적인 행위자가 DIM을 과부하시켜 서비스 거부를 초래할 수 있는 위협 방지
10	권한 상승 방지	DIM의 취약점을 악용하여 무단 접근 또는 권한 상승 방지

Ⅵ. 결 론

DLT 기술의 빠른 발전과 다양한 산업 분야에서의 광범위한 채택으로 인해 단일 DLT 시스템만으로는 다양한 분산 응용 프로그램의 요구를 충족하기 어려워졌습니다. 이러한 상황에서 DLT 게이트웨이는 표준 프로토콜을 준수하여 다양한 분산 응용 프로그램 및 DLT 시스템 간의 원활한 상호작용을 가능하게 하며, 퍼블릭 및 프라이빗 DLT 간의 데이터와 자산 전송, 자산 교환 등의 상호운용성을 지원합니다.

그러나, DLT 게이트웨이를 통한 상호운용성 제공 시 다양한 보안 위협이 존재합니다. 무단 접근, 데이터 인터셉션, API 악용, 서비스 거부(DoS) 공격 및 입력 유효성 검사 부족 등의 보안 위협은 DLT 게이트웨이의 신뢰성과 안정성을 저하시킬 수 있습니다. 본 논문에서는 이러한 보안 위협을 분석하고, 각 기능 요소별 보안 요구사항을 제시하였습니다. 이를 통해 DLT 게이트웨이가 안전하고 신뢰할 수 있는 운영을 보장할 수 있도록 하였습니다.

앞으로 DLT 게이트웨이의 보안 요구사항을 충족시키기 위해서는 지속적인 연구와 보완이 필요합니다. 특히, 새로운 보안 위협이 나타날 수 있는 복잡한 운영 환경에서 DLT 게이트웨이의 보안성을 유지하기 위해 정기적인 보안 점검과 업데이트가 중요합니다. 본 논문에서 제시한 보안 요구사항을 바탕으로 다양한 DLT 시스템 간의 안전한 상호운용성을 구축하고, 보다 발전된 DLT 생태계를 형성하는 데 기여할 수 있기를 기대합니다.

참 고 문 헌

[1] ISO/TC 307/WG7, “Interoperability Framework Draft v0.81”, *ISO*, 2023

[2] ITU-T SG17/WG14, “ITU-T X.DLT-dgi: Security requirements of DLT gateway for interoperability”, *ITU-T*, 2024

〈저 자 소 개〉

김 미 경 (Kim Mi Gyeong)

2019년 8월 : 건국대학교 정보통신대학원 (공학석사)
2022년 3월 : 고려대학교 정보보호대학원 (박사수료, 재학중)
2022년 11월 ~ 현재 : ㈜드림시큐리티 책임
<관심분야> 정보보호, 블록체인, 보안 프레임워크

황 정 연 (Jung Yeon Hwang)

증신회원

1999년 2월 : 고려대학교 수학과 졸업
2013년 2월 : 고려대 정보보호대학원 석사
2016년 8월 : 고려대 정보보호대학원 박사
2020년 3월~현재 : 성신여대 수리통계데이터사이언스학부 조교수
<관심분야> 암호이론 및 응용, 프라이버시, 블록체인, 핀테크 암호

김 영 진 (Kim Young Jin)

1989년 2월 : 중앙대학교 컴퓨터공학과 (공학사)
2000년 8월 : 충남대학교 컴퓨터과학과 (이학석사)
2003년 8월 : 충남대학교 컴퓨터과학과 (이학박사 수료)
1990년 3월~2000년 1월 : 국방과학연구소 연구원
2000년 2월~2001년 1월 : 국가보안기술연구소 선임연구원
2004년 7월~현재 : ㈜드림시큐리티 상무
<관심분야> 인증프레임워크, 암호프로토콜, 전자서명, 보안토큰, 바이오인식, 블록체인