JKSCI

한국컴퓨터정보학회논문지
Journal of The Korea Society of Computer and Information
Vol. 29 No. 1, pp. 121-130, January 2024
https://doi.org/10.9708/jksci.2024.29.01.121

Enhancing Installation Security for Naval Combat Management System through Encryption and Validation Research

Byeong-Wan Lee*

*Engineer, Naval R&D Center, Hanwha Systems, Gumi, Korea

[Abstract]

In this paper, we propose an installation approach for Naval Combat Management System(CMS) software that identifies potential data anomalies during installation. With the popularization of wireless communication methods, such as Low Earth Orbit(LEO) satellite communications, various utilization methods using wireless networks are being discussed in CMS. One of these methods includes the use of wireless network communications for installation, which is expected to enhance the real-time performance of the CMS. However, wireless networks are relatively more vulnerable to security threats compared to wired networks, necessitating additional security measures. This paper presents a method where files are transmitted to multiple nodes using encryption, and after the installation of the files, a validity check is performed to determine if there has been any tampering or alteration during transmission, ensuring proper installation. The feasibility of applying the proposed method to Naval Combat Systems is demonstrated by evaluating transmission performance, security, and stability, and based on these evaluations, results sufficient for application to CMS have been derived.

▸ **Key words**: Naval Combat Management System, File Transfer, Integrity Verification, File Security, Hash Algorithm

[요 약]

본 논문에서는 함정 전투체계 소프트웨어 설치 시 발생 가능한 데이터 이상을 확인하는 설치 방안을 제안한다. 최근 저궤도 위성 통신과 같은 무선 통신을 이용한 정보교환 방법이 대중화되며, 함정 전투체계에서도 무선망을 이용한 여러 활용 방안이 논의되고 있다. 활용 방안 중 하나로서 무선망 통신을 이용한 설치를 함정 전투체계에 적용함으로써 실시간으로 전투체계 성능을 향상시킬 수 있는 방법이 가능해질 것으로 기대한다. 하지만 무선망의 경우 유선망보다 상대적으로 보안상 취약하므로 더 많은 보안 대책이 강구된다. 본 논문에서는 암호화 방식을 통해 다수의 노드에 파일을 전송하고 파일 설치 이후 유효성 검사를 수행함으로써, 전송 도중 위/변조 여부를 판단하여 정상적으로 설치됨을 확인한다. 제안한 방법의 함정 전투체계 적용 가능성을 보이기 위하여 전송 성능 및 보안성, 안정성 등을 평가하였으며, 이를 바탕으로 함정 전투체계에 적용하기 충분한 수준의 결과물을 도출하였다.

▸ **주제어**: 함정 전투체계, 파일 전송, 무결성 검증, 파일 보안, 해시 알고리즘

• First Author: Byeong-Wan Lee, Corresponding Author: Byeong-Wan Lee
*Byeong-Wan Lee (bwlee3415@hanwha.com), Naval R&D Center, Hanwha Systems
• Received: 2023. 11. 20, Revised: 2024. 01. 02, Accepted: 2024. 01. 02.

http://www.ksci.re.kr pISSN:1598-849X | eISSN:2383-9945

I. Introduction

함정 전투체계(CMS : Naval Combat Management System)는 함정 내의 각종 센서 및 무장을 통합하며, 표적 관리 및 교전 등을 수행하며 그 외에 함정 작전 수행에 필요한 다양한 기능을 제공하고 있다. 이러한 각종 기능을 수행하기 위해 전투체계는 하드웨어와 소프트웨어가 유기적으로 결합된 복잡한 기능을 가지고 있으며, 취역 이후에도 소프트웨어와 하드웨어를 업데이트하며 성능 개선과 운용정비를 수행하고 있다.[1]

해군에서는 현장 정비를 전문적으로 수행하면서도 예상되는 문제를 선제적으로 대응하기 위해 수명주기지원(LTS : Life Time Support) 업무를 다양한 함정을 대상으로 수행하고 있다.[2] LTS 현장정비센터는 각 군항에 전문 인력들이 상주함으로서 하드웨어 및 소프트웨어의 신속한 고장 복구에는 능통하지만, 함정 해외 파병, 연합훈련과 같이 먼 바다로 갈 때 소프트웨어 문제가 발생하면 조치에 다소 제한이 생긴다.

이러한 단점을 보완할 수 있는 것이 무선 네트워크를 이용한 함정 전투체계 소프트웨어 설치이다. 최근 군에서는 저궤도 위성을 이용한 다양한 활용방안이 논의되고 있다.[3] 함정 전투체계에 저궤도 위성과 같은 무선 네트워크를 이용한 설치가 가능하다면 SW 개선사항이 생길 경우 기존 대비 단시간에 개선한 SW 제공이 가능하며, 시간적/공간적 제약이 없어지므로 많은 시간과 비용 절감이 가능하다.

저궤도 위성과 같은 무선 통신망의 경우 파일 전송 시 여러 요소를 고려해야 한다. 저궤도 위성의 경우 중궤도 위성 및 정지궤도 위성에 비해 상대적으로 전파왕복시간(RTT : Round Trip Time)이 짧으므로 손실도 및 평균 지연시간이 낮지만[4] 유선망에 비해 높은 데이터 손실을 고려해야 한다. 또한 나날이 높아지는 사이버 위협에 대비하여[5] 파일 전송 시 보안도 고려해야 한다.

본 논문에서는 암호화 및 유효성 검사를 이용한 함정 전투체계 소프트웨어 설치 방안을 제안한다. 보안을 중요시하는 전투체계 특성을 고려하여[6] 설치를 위한 파일 전송은 암호화된 전송을 통해 안전하게 수행하며, 데이터 위/변조 시도 및 무선망의 손실도 가능성을 고려하여 파일 전송 후 유효성 검사를 수행하는 기능을 제공한다.

본 논문의 구성은 다음과 같다. 2절에서는 본 논문에 서술되는 기술 동향 및 배경지식에 대해 설명하고 3절에서는 소프트웨어 설치 프로그램에 대한 개발 방법에 관해서 설명한다. 그리고 4절에서는 소프트웨어 설치 프로그램에 대한 기능과 성능을 입증한다. 마지막으로 5절에서는 결론 및 추후 연구과제로 이 논문을 마무리한다.

II. Preliminaries

1. Existing Installation Program

1.1 CMS Installation Program

기존 전투체계 설치 프로그램은 소프트웨어 배포를 주 목적으로 하므로 다음과 같은 프로세스를 거치게 된다.

- Ping을 이용한 가용 Node 확인
- 보안 Shell 프로그램 설치
- 기존 데이터 백업
- 보안 Shell과 FTP를 이용한 전투체계 프로그램 설치

기존 전투체계는 유선 LAN으로 이루어진 별도의 망으로 운영되고 있고, Node의 개수 또한 하드웨어 1개당 하나의 Node로 구성되어 있었기 때문에 전투체계 전체로 봤을 때는 약 40-50개 정도로 전체 설치 시 약 2시간가량 소요되고 있다. 하지만 최근 전투체계의 경우 생존성 강화를 위해 가상화를 채택하고 있으며, 이에 따라 하드웨어당 Node의 개수가 상당수 늘어나게 되었다. 또한, 각각의 프로그램들 또한 성능 향상 및 기능의 다양화로 인해 파일 개수와 용량 또한 증가하게 되었다.

Table 1. Compare number of Node and File

CMS Name	Number of Nodes	Total File Count	Storage Capacity
A	약 50개	약 1,300	약 800MB
B	약 80개	약 6,000	약 4.7GB
-	1.6배	4.6배	5.9배

Table 1은 과거 전투체계(A)와 최근 전투체계(B)의 파일의 용량 및 개수 비교를 보여준다.

전투체계가 발전함에 따라 설치해야하는 Node와 파일 개수, 용량 등이 4배 이상 증가하였으며 이에 따라 기존 설치 프로그램으로는 발전한 최근 전투체계 설치하기에 효율성이 아쉬운 상황이다.

1.2 Related Installation Program

파일 전송은 현대 사회에서 자주 쓰이는 기술 중 하나로, 비교적 간단한 구조지만 속도, 보안, 효율 등 목적에 따라 다양한 분야에서 연구하고 있다. 특히 보안의 경우 중요한 요소 중 하나지만 너무 많은 보안 장비 및 방법론을 적용할 경우 오히려 전송 성능 저하 및 장애를 일으킬

수 있다.[7] 그렇기 때문에 파일 전송 시 성능과 보안 사이에 적절한 타협점을 찾아야 하며, 이를 위한 대안 중 하나가 애플리케이션 수준에서 보안이 강화된 파일 전송 프로세스를 구현하는 것이다.[8] 본 연구에서는 보안성이 적용된 파일전송 프로토콜을 적용하는 것 외에도 해시함수를 적용하여 안정성을 강화하고자 한다. 더 나아가 해당 방법론이 적용된 프로그램을 구현하여 손쉽게 전투체계 설치 및 설치 확인을 할 수 있도록 한다.

2. International Trends

최근 자동차 산업에서는 OTA(Over The Air) 기능을 적용하고 있는 사례가 점진적으로 늘어나고 있다.[9] OTA 기능이란 차량의 전자제어장치(ECU : Electronic Control Unit)를 무선 통신 기술을 이용하여 ECU 내 소프트웨어를 업데이트 하는 기능으로 현재 대부분은 차량용 인포테인먼트(IVI : In-Vehicle Infotainment) 및 텔레매틱스(Telematics)에 대부분 적용되어 있지만, 향후에는 각 ECU에 적용 필요성이 높아지고 있다.[10] 무선 통신을 이용한 소프트웨어 설치 및 배포 기술은 위성통신 및 이동통신 기술이 발전함에 따라 군용 시스템에도 적용되고 있는 추세이다.

미국의 IDT(Innovative Defense Technologies) 사에서는 군수업체인 Lockheed Martin과 파트너 십을 맺고 이지스 엔터프라이즈 PaaS(Platform as a Service)를 제공하고 있다.[11]

실 사례로 하와이의 ARDEL, 뉴멕시코의 WSMR과 같은 Test Site 혹은 실제 선박의 가상화된 AEGIS에 변경된 소프트웨어를 빠르게 배포하고 있다. 또한, 기밀 클라우드 시스템을 통해 함정 내의 모델링, 시뮬레이션, 데이터 분석을 수행함으로써 함정에 더 적합한 소프트웨어 버전을 분석 및 배포하여 미 해군에 적용하고 있다.

3. SSH and SFTP

SSH(Secure Shell)는 네트워크 상 연결된 타 컴퓨터와 통신하거나, 명령을 전송하기 위해 사용되는 네트워크 프로토콜이다. 과거에는 Telnet, FTP와 같은 네트워크 프로토콜을 사용하여 파일 전송, 명령 전송을 수행하였으나, 이들은 암호화가 적용되어 있지 않아서 보안상 취약점이 많았다. 이와 달리 SSH는 암호화된 쉘을 제공함으로써, 다양한 네트워크 위협에 대응하며 안전한 통신을 수행한다.[12]

파일 전송 프로토콜로는 SFTP, SCP, FTPS, NFS 등이 있다. 파일 전송 프로토콜은 사용 환경 및 요구사항에 따라 선택되는데, 본 논문에서는 데이터 전송 시 보안이 우선시되므로 SFTP(SSH File Transfer Protocol)를 선택하였다. SFTP는 SSH 프로토콜을 기반으로 하는 파일 전송 프로토콜로서, 파일 전송이 SSH 내에서 이루어지므로 모든 데이터 전송이 암호화되어 전송된다. 다른 파일 전송 프로토콜에 비해 보안성이 높고, 다수의 사례로 검증받은 방법이므로 신뢰성이 높아 SFTP를 사용하게 되었다.

위 사유로 본 연구에서 제안할 명령 전송 및 파일 전송 방식은 SSH와 SFTP로 결정하였다. SSH는 보안성이 유지된 채로 명령을 전달할 수 있으며, SFTP는 유선 및 무선 환경에서도 암호화와 보안성이 유지되어 파일 설치에 가장 적합하다.

4. Hash Function

해시 함수는 데이터를 입력으로 받아서 고정된 길이의 문자열을 출력하는 함수로서 크게 암호학적 해시 함수와 비암호학적 해시 함수로 나뉜다.

이중 암호학적 해시 함수의 경우 주요한 특징으로 눈사태 효과, 충돌 저항성, 역상 저항성이 있다.

눈사태 효과(Avalanche Effect)는 원문(PlainText)의 한 비트 변화가 최종 암호문(CipherText)에 큰 변화를 주는 암호학적 특성이다. 본 연구에서 제안할 유효성 검사에 가장 중요한 특성으로, 각각의 파일에 대해 약간이라도 위변조가 생기면 해시 검사를 통해 비정상적인 접근이 있었는지 확인할 수 있다. Fig. 1은 SHA-256 해싱을 통해 유효성 검사를 수행한 결과를 보여준다.

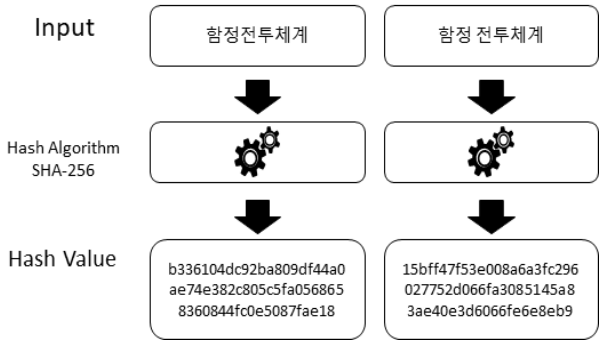

Fig. 1. SHA-256 Hash Value Example

충돌 저항성(Collision resistance)은 같은 결괏값을 갖는 다른 입력 값을 계산하기 어려워야한다는 특성이다. 전투체계는 수천 개 이상의 파일이 존재하며, 각각에 대해 고유한 해시 값이 존재해야한다. 충돌 저항성의 특성으로 각 파일의 해시 값이 충돌이 생기는 경우는 거의 없다고 할 수 있다.

역상 저항성(preimage resistance)은 해시 값이 주어졌을 때, 입력 값을 예상하기 힘들어야 한다는 특성이다. 이를 통해 checksum이 드러나도 외부에서는 해당 파일이 어떤 파일인지 알기 힘들다.

일반적으로 널리 쓰이는 해시 알고리즘에는 SHA, MD5, Blake, Whirlpool 등이 있다. 이 중 본 연구에서는 SHA-256 방식을 채택하였다.

SHA-256(Secure Hash Algorithm 256-bit)는 암호화 해시 알고리즘 중 한 종류이다. 빠른 연산을 통해 256비트 길이의 해시 값을 추출 할 수 있으며, 여러 보안 어플리케이션 및 프로토콜에서 안정성 및 무결성 보장을 위해 사용된다. 충돌 저항성에서도 큰 이점을 가지기 때문에 체크썸 검사에 탁월한 성능을 보인다.[13] 보안성이 더 높은 해시 함수로는 SHA-512, SHA-3 등이 있으나, 상대적으로 연산속도가 느리며 본 연구에서는 무결성 확인을 목표로 하므로 연산 속도와 충돌 저항성이 낮은 SHA-256을 선택하였다.

III. The Proposed Scheme

이 절에서는 무선망을 활용한 소프트웨어 설치 프로그램 개발 방법에 관해서 설명한다.

함정 전투체계는 국내에 도입된 이래로 계속 발전하여 소프트웨어 규모가 점점 대형화되고 있고, 생존성 증대 및 가상화 등을 만족시키기 위해서 Node 개수가 기존 체계와 비교해 상당수 증가하게 되었다. 함정 전투체계가 발전할수록 Node 개수가 더욱 증가하는 경향이 있으므로, 소프트웨어 설치 시 효율성을 위해 다수의 Node에 동시에 배포하는 것 또한 고려하여 개발한다.

1. Purpose of Program

제안한 함정 전투체계 설치 프로그램은 Fig. 2의 절차와 기능을 가진다.

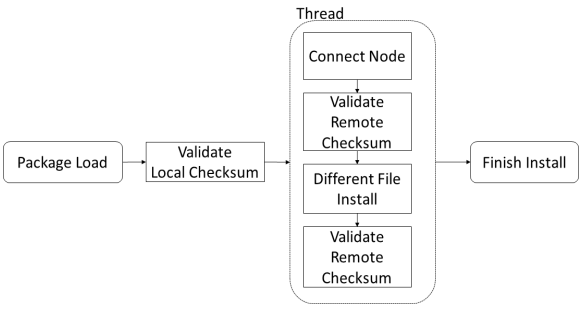

Fig. 2. Proposed Installation Program Function

SSH를 이용하여 각 Node에 접근한다. 특정 데이터 파일에 암호화 된 ID/Password를 읽고, Shell 객체를 생성한다. 또한 해당 Shell의 SFTP를 생성하여 파일 전송 및 명령어 처리를 한다.

Local 컴퓨터의 설치할 파일 체크썸(CheckSum)과 Remote 컴퓨터에 설치된 파일의 체크썸 비교를 통해 설치 직후 파일 정상 설치 여부를 식별한다. 체크썸 비교는 SHA-256 알고리즘을 이용하여 각 파일에 대해 신속히 해싱하며 결과를 리턴 하여 화면에 전시한다.

설치를 진행하는 Local 컴퓨터의 파일과 설치 대상인 Remote 컴퓨터의 체크썸 결괏값을 비교함으로써 파일 전송 후 위/변조 여부 및 정상 설치 여부를 판단할 수 있다.

유효성 검사 결과 Remote 컴퓨터에 대해 체크썸이 다른 파일에 대해서만 설치를 진행한다. 전투체계의 경우 성능 개선을 위한 일부 파일만 변경되는 경우가 다수이며, 변경되는 파일 비중이 최대 35% 미만이다. 설치 전 유효성 검사를 진행하는 시간을 고려하더라도 변경된 파일만 설치하는 것이 시간 측면에서 유리하다. 또한 다수의 Node에 설치하기 때문에 병렬 작업을 수행하기 위해 멀티 쓰레딩을 활용하여 설치한다.

기존에는 빌드 한 프로그램 및 설치 정보가 담겨있는 데이터 파일을 이용하여 설치를 수행하였으나, 이러한 형태는 기능 구성이나 UI 변경 등에 시간과 노력이 다소 들게 된다. 이를 개선하기 위해 HTML과 JavaScript 기반의 Web으로 화면을 구성하여 기능 확장 및 UI 변경이 유연하게 이루어질 수 있도록 한다. 기능 실행은 Python으로 수행함으로써 코드를 간결하게 구성한다.

2. Structure of Program

제안한 설치 프로그램에서는 위에서 제안한 설치 프로그램의 목적성에 맞게 기능을 구성하였으며, 프로그램의 전체적인 구성은 Fig. 3과 같다.

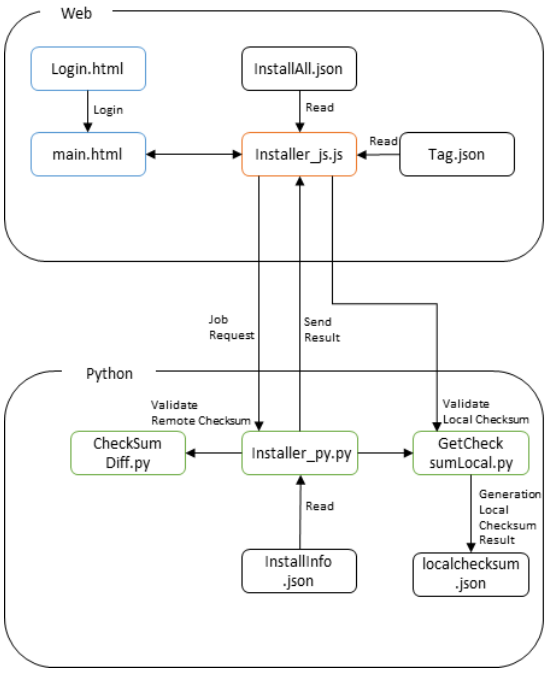

Fig. 3. Configuration of the Proposed Installation Program

각 컴포넌트의 대략적인 기능은 다음 Table 2와 같다.

Table 2. Function of Component

File Name	Kind	Description
Login	Web (HTML)	Perform User Login
main		Display Web UI
installer_js	Web (Java Script)	Execute Web Function/Events
installer_py	python	Establish Node Connection and Perform File Transfer
CheckSumDiff		Perform Remote File Checksum Comparison
GetChecksum Local		Perform Local File Checksum Comparison
InstallAll	Data (json)	Remote Installation Information

2.1 Login

Login 컴포넌트는 설치를 위해 인가자만 접근 가능하게 만든 Web 화면이다. 웹 서버 구동 시 해당 화면부터 접근되며, 인가된 ID 및 Password를 입력해야만 main 컴포넌트로 넘어가게 된다.

2.2 main

main 컴포넌트는 설치 및 기타 기능을 수행하도록 설계된 Web 화면이다. 해당 화면은 기능을 수행하기 위한 버튼 부와 각 Node 들이 전시되는 리스트 화면, 각 Node 별로 설치할 파일이 전시되는 리스트 화면 그리고 설치 로그를 보여주는 텍스트 화면으로 나뉜다.

전시되는 화면은 Fig. 4와 같다.

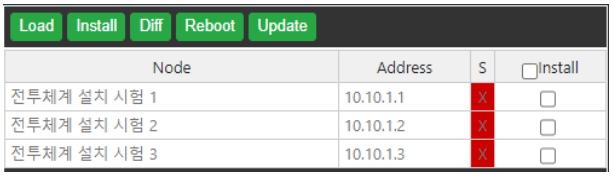

Fig. 4. Proposed Installation Program Display Example

설치를 위한 대표 기능은 다음과 같다.

- Load(Button) : 전투체계 패키지를 불러오기 위한 동작으로 최초 1회 자동으로 수행되며, Tag/InstallAll json 파일을 읽어 들여 각 Node 이름 전시 및 Node 별 파일을 전시한다.
- Update(Button) : 설치를 수행하는 Local 컴퓨터 내의 전투체계 패키지 파일들의 체크썸을 확인한다. 체크썸은 SHA-256 알고리즘을 통해 생성되며, 체크썸을 확인한 후 'localchecksum.json' 파일을 생성한다.
- Diff(Button) : 설치되는 Remote 컴퓨터에 SSH 접속한 후 해당 Node 설치 파일 목록을 읽어 들인다. 해당 파일 목록은 localchecksum 상의 파일 목록 및 체크썸과 Relation 되며, CheckSumDiff.py 파일이 업데이트된다.

업데이트된 CheckSumDiff.py는 SFTP를 통해 Remote 컴퓨터로 파일 전송된 후, CheckSumDiff.py 내용이 수행된다. 수행 결과로 전달된 결괏값은 최종적으로 파일 목록에 배경 음영으로 전시된다.

각 배경 음영이 나타내는 결과는 다음과 같다.

1) 파란색 : 동일 파일(체크썸 결과가 같음)
2) 노란색 : 다른 파일(Remote 컴퓨터에 파일은 존재하나 체크썸 결과가 다름)
3) 빨간색 : 없는 파일(Remote 컴퓨터에 파일이 없음)

만약 Node상 모든 파일 목록이 파란색인 경우 Node 목록에서 해당 목록 또한 파란색이라도 표시되며, 노란색이나 빨간색이 하나라도 있을 경우 해당 색상으로 표시된다.

위의 과정을 간략화한 것은 Fig. 5와 같다. 해당 체크썸 비교 프로세스는 두 가지 경우에 사용된다. 첫 번째로 설치 전에 수행하게 되는데, 설치 전 동일 파일 여부를 판별함으로써 실제로 설치가 필요한 파일들만 골라서 설치를 진행함으로써 운용자의 시간 및 네트워크 부하가 줄어든

다. 두 번째로는 설치 후에 수행하게 되는데, 이 경우 설치 간 위/변조가 있었는지 확인이 가능하며, 위/변조가 아니더라도 설치 간 누락되거나, 네트워크 문제로 설치가 제대로 되지 않은 경우 판별이 가능하다. 이로써 운용자가 의도한 대로 정상 설치가 되었는지 확인이 가능하다.

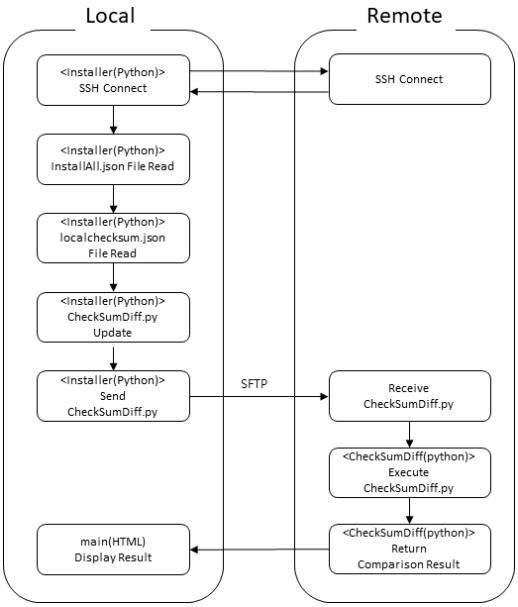

Fig. 5. Proposed Installation Program Process

- Install(Button) : 전투체계 설치를 수행하는 기능으로서 설치하고자 하는 Node와 Node 별 파일 목록에 'Install' 체크박스에 체크가 되어 있어야 한다.

파일 설치를 시도하는 경우는 다음과 같다.

1) 해당 Node에 Ping이 가능해야 한다. 설치 프로그램에서는 패키지가 Load 된 이후 모든 Node에 대해 Ping을 시도하며 해당 Node가 가용할 경우 Node 목록의 'S' 항목에 'O'(배경 : 초록색)이 표시된다.
2) 체크썸 비교(Diff)를 수행한 경우 노란색 혹은 빨간색 음영으로 표시된 파일만 설치를 수행한다. 해당 Node가 파란색으로, 모든 파일이 동일한 경우 해당 Node는 설치를 진행하지 않는다. 이후 운용자가 설치를 수행할 때, 파란색으로 표시된 파일 목록만 Installer(python)으로 전달된다.

2.3 Installer_js

Installer_js 컴포넌트는 main 컴포넌트와 연결되어 동적인 동작을 수행한다. main의 각 버튼과 동작이 연결되어 있으며, 동작을 수행할 대상 정보를 인자로 받아 Installer_py로 동작을 지시한다.

최초 실행 시 InstallAll, tag json 파일을 읽어서 Node 목록 정보와 Node 별 파일 목록을 내부적으로 처리하고, main으로 전달하여 전시한다.

또한 특정시간 마다 각 Node에 Ping 체크를 하며, 가용여부를 기록한다.

2.4 Installer_py

설치 프로그램이 동작할 때마다 실행되는 컴포넌트로서 각 Node와 SSH 연결을 시도하고 설치 및 체크썸 비교를 수행하는 메인 처리 파일이다.

SSH 연결은 paramiko 모듈을 사용하여 SSH Client 객체와 SFTP 객체를 만들게 된다. Installer_js로부터 기능을 수행할 IP와 파일 내용을 수신하게 되면, InstallInfo json 파일을 읽은 후 해당 IP에 대한 ID/Password를 입력하여 SSH로 연결한다.

SSH 연결이 성공한 후 파일 체크썸 비교와 파일 설치를 수행한다. 파일 체크썸 비교의 경우 상술했듯이, SSH 연결 이후 체크썸 비교 수행 시 해당 Node에 대한 파일 리스트와 체크썸 리스트를 읽은 후 CheckSumDiff를 업데이트하여 Remote 컴퓨터에서 실행 후 결괏값을 받아온다. 파일 설치의 경우 Installer_js에서 넘어온 파일 리스트를 기반으로 Remote 컴퓨터로 송신하게 된다.

2.5 CheckSumDiff

CheckSumDiff 컴포넌트는 Installer_py에서 로컬 파일 체크썸 리스트가 업데이트 되며, Remote 컴퓨터에서 실행되어 로컬 파일 체크썸 리스트와 Remote 컴퓨터에 있는 파일 리스트들의 체크썸과 비교하는 역할을 수행한다.

각 파일의 체크썸은 SHA-256 알고리즘에 의해 처리 되며 메모리 효율성을 위해 파일 별로 1,024byte 씩 끊어서 체크썸 업데이트를 수행한다.

체크썸 비교 결과 파일이 없을 경우 0, 체크썸이 다른 경우 1, 체크썸이 동일한 경우 2를 리턴하며 최종 결괏값은 파일 개수만큼 main 컴포넌트로 전달된다.

Fig. 6은 위 과정을 도식화하였다.

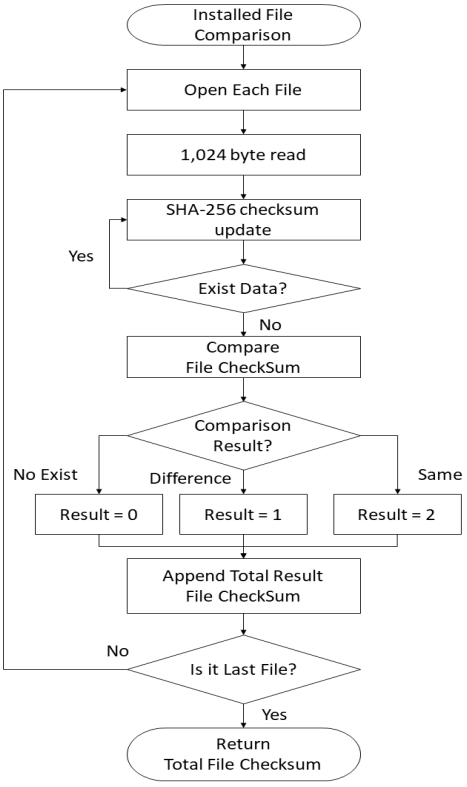

Fig. 6. Proposed File Comparison Process

2.6 GetChecksumLocal

GetChecksumLocal 컴포넌트는 'Update' 기능 수행 시 호출되며, 전투체계 패키지 내에 있는 모든 파일에 대해 체크썸을 확인한다. 체크 알고리즘은 CheckSumDiff와 동일하며 결과파일로 localchecksum.json 파일이 생성된다.

2.7 InstallAll

InstallAll 컴포넌트는 설치되는 파일과 설치되는 위치를 저장하기 위한 데이터 파일로서, [IP주소, Local 파일 위치, Remote 설치 위치]로 구성된다.

IntallAll json 파일은 설치 프로그램 초기 전시 시 노드별 파일 목록을 전시하는 데 사용되며, 체크썸 비교 및 설치 시 설치되는 파일 위치를 확인하는 데 사용된다.

IV. Evaluation

본 논문에서 제안한 전투체계 설치 프로그램 성능 평가를 위해 유효성 확인, 파일 전송, 파일 위/변조 확인을 수행하였다. 비교 시험 환경은 아래 Fig. 7 및 Table 3과 같다.

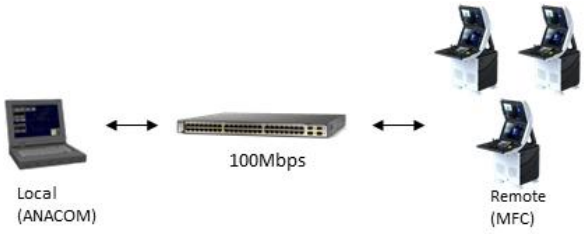

Fig. 7. Test Environment

Table 3. Test Environment

Equipment Name	CPU	MEM	OS
ANA COM (Local)	Intel Core(TM) i7-4600M 3.6GHz	8GB	Windows10
MFC (Remote)	Intel Xeon Gold 6142 2.6GHz	128GB	Windows10

테스트 환경은 실제 설치에 사용되는 스펙과 동일한 분석 컴퓨터(ANACOM : Analysis Computer) 및 다기능콘솔(MFC : Multi Function Computer) 3대를 사용하였다. 테스트 환경에서 실제 무선망 구성은 제한되어 유선망으로 구성하였다. 현재 상용화되어있는 SpaceX 사의 StarLink와 같은 위성통신 서비스는 약 50~150Mbps의 대역폭으로 제공하고 있다.[14] 이와 유사한 저속도 환경에서도 원활히 동작함을 확인하기 위해 랜 대역폭은 100Mbps에서 시험하였다.

1. Validity Test

본 논문에서 제안한 설치 프로그램은 유효성 검사를 통해 설치하는 원격 컴퓨터의 파일 동일 여부를 확인한 뒤, 불일치 파일만 확인하여 중복 설치를 막음으로써 네트워크 부하 및 설치 시간 감소를 목표로 한다. 이를 위해 파일 비교에 큰 시간이 소요되지 않음을 확인하기 위해 크기 및 개수 별 체크썸 확인 시간을 도출하였다.

각 파일은 임의로 생성된 텍스트 파일을 기준으로 하였다. 파일 개수에 따른 비교를 수행할 때는 1MB 크기의 파일을 사용하였으며, 파일 크기에 따른 비교를 수행할 때는 파일 10개를 기준으로 하였다. 시험은 3회의 평균값을 작성하였다.

Table 4. Processing Time by File Count

Number of Files	Time(s)
100	0.17
1,000	1.82
3,000	5.21
5,000	8.65

Table 5. Processing Time by File Size

File Size	Time(s)
10MB	0.15
100MB	1.37
500MB	7.01
1GB	13.71

Table 4와 Table 5는 각각 파일 개수와 파일 크기를 변경시키며 비교한 내용이다. 파일 수와 크기에 따라서 수행시간은 정비례하여 증가하였으나, 소요 시간은 길지 않음을 확인하였다.

2. Data Alteration Test

데이터 위/변조 확인을 위해 임의로 생성된 1000개의 텍스트 파일에 대해 원격 컴퓨터에 설치를 진행한 후 체크썸 검증을 진행하여 정상적으로 설치됨을 검증하였다. 그리고 설치된 파일 한 개에 대해 문자 하나를 변경하여 재검증을 시행하였다.

Fig. 8과 같이 검증 결과 변경된 파일에 대해서 체크썸 변경이 이루어졌음을 확인할 수 있었으며, 이를 통해 사용자는 제안한 설치 프로그램을 이용해서 정상적으로 설치되었는지를 알 수 있음을 확인하였다.

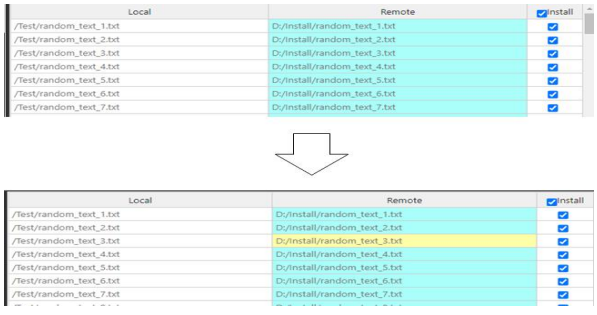

Fig. 8. Result of Data Alteration Detection

새로운 전투체계 설치 시 기존 파일과 동일 여부를 검사 후 설치가 필요한 파일만 선택적으로 전송하므로 데이터 전송량을 효율적으로 할 수 있다. 또한, 설치 후 데이터 위/변조 확인을 통해 통신 중에 이루어 질 수 있는 보안 위협 및 Data Loss로 인한 파일 전송 오류를 확인 할 수 있으므로 기존 설치 패키지 대비 보안성, 안정성이 높아졌음을 확인할 수 있다.

3. Installation Evaluation

본 논문에서 제안한 설치 프로그램과 FTP를 이용한 파일 전송 프로그램을 이용한 시간을 비교하였다.

전송에 사용되는 데이터는 전투체계 내 파일 구성과 유사하게 다음과 같이 구성하였다.

- 50KB : 5,000개
- 1MB : 1,000개
- 5MB : 500개
- 10MB : 15개
- 100MB : 10개

OOO 사업 기준 설치 시 전체 파일 대비 최대 35% 가량 수정되므로, 해당 수치를 기준으로 산정하여 시간 비교를 수행하였다.

수행 한 결과는 Table 6과 같다. FTP를 통해 전체 파일을 전송했을 때 579.27초가 소요되었고 제안된 설치프로그램을 이용하여 35% 가량 수정되는 일반적인 설치를 수행할 경우 245.58초가 소요된다. 이는 기존 대비 57.6%가량 줄어든 것이다.

Table 6. Processing Time by Installation Program

Program	Action	Time(s)
FTP Install Program	Install	579.27
Proposed Install Program	Validate Local CheckSum	11.57
	Validate Remote CheckSum	11.98
	Install	210.02
	Validate Remote CheckSum	12.01
	Total	245.58

다음은 다수 Node에 대해 설치를 시도했을 때의 설치 시간을 비교하였다. 양 프로그램은 멀티쓰레드를 통해 파일 전송을 수행하고 있으며, 설치는 단일 Node 시험과 같이 진행하였다.

Fig. 9는 다수 Node 설치 시 Node 개수에 따른 시간 기록을 보여준다. Node 개수가 증가할수록 시간 차이가 커지는 양상을 보여주는데, 제안한 설치 프로그램의 경우 Local 컴퓨터 체크썸 검사는 1번만 수행되며, Remote 체크썸 검사는 각 Remote 컴퓨터에서 동시에 수행되므로 실질적으로 35% 설치 시간만 추가된다.

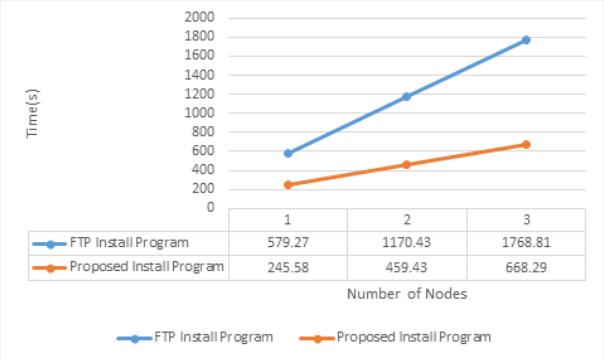

Fig. 9. Processing Time by Installation Program for Multi-Node System

이를 통해 본 논문에서 제안한 설치 방안 적용 시 함정 전투체계와 같이 다수의 Node를 가진 시스템일수록 효용성이 커짐을 확인하였다.

V. Conclusions

본 논문에서는 암호화 및 유효성을 적용한 전투체계 응용 소프트웨어 설치 방안을 제안하였으며, 제안한 방법으로 구현 후 시험을 통해 다수 Node에 암호화 및 유효성 검사를 적용하여 설치할 수 있음을 검증하였다. 설치 대상과 설치 파일 체크썸 비교를 수행함으로써 전투체계 내 다수 Node에 소프트웨어 설치 시 설치 시간 단축 및 네트워크 부하 감소를 기대할 수 있으며, 시험을 통해 기존 설치 방법 대비 50% 이상 시간 단축됨을 확인하였다. 위성통신 기술이 발전할수록 군에서도 위성통신을 활용한 다양한 방법을 마련할 것이다. 특히 해군에서는 먼바다로 갈수록 위성 통신 활용성이 높을 것이고, 향후에는 상황에 따른 맞춤형 전투체계를 적시 적소에 활용할 수 있을 것이다. 본 논문에서 제안한 설치 방안을 적용한다면 보안성과 유효성 검사를 통해 안전하게 원하는 함정 전투체계 소프트웨어 설치가 가능할 것이고, 무선망이 아니더라도 발전하며 거대해지는 함정 전투체계 설치에 도움이 될 것이다.

아쉬운 점으로는 현재 전투체계와 설치 시험이 가능한 무선망이 구축되어 있지 않기 때문에 본 논문에서는 제한된 환경에서 설치된 데이터에 대한 변조 시험으로 보안성 및 안정성을 확인하였다. 추후 연구에서는 유선망과 다르게 무선망에서 더 위협적인 신호 간섭 및 중간자 공격과 같은 다양한 보안 위협에도 대응이 가능한지 연구를 수행할 예정이다.

REFERENCES

[1] Jeong-Woo Son, "A Study on the Design Plan of Naval Combat System Software to Reduce Cost of Hardware Discontinuation Replacement," Journal of the Korea Society of Computer and Information , Vol. 28, No. 1, pp. 71-78, 2023. DOI : 10.9708/jksci.2023.28.01.071

[2] Im Jin Guk, Choi Yong Seok, Kim Soo Beom, “Performance and Development of the warship Combat System Life Time Support”,Defence and Technology, 497(1), pp.84-95, Jul 2020.

[3] Sunghwan Cho, Hochan Lee, Sangjun Park, and Yongchul Kim, "Military Applications of Commercial Low Earth Orbit Satellite Communications," Proceedings of Symposium of the Korean Institute of communications and Information Sciences, pp. 886-887, 2023.

[4] Yeon Jaesung, Han Youngsik, Yoon Jongtaek, "Technological Trends and Research Directions for Next-Generation Military Communication Networks Based on Low Earth Orbit (LEO) Satellites," Defense & Technology, No. 537, pp. 108-121, 2023.

[5] Song, Jae-Ik, "A Study on Enhancing Joint Cyber Operations of the Republic of Korea(ROK) Forces - Focused on linkage with Joint Operations -," Korean Journal of Military Affairs, No. 2, pp. 147-186, 2017. DOI: 10.33528/kjma.2017.12.2.147

[6] Cheol-Gyu Yi, and Young-Gab Kim, "A Study on Software Security Test of Naval Ship Combat System," The Journal of Korean Institute of Communications and Information Sciences, Vol. 45, No. 3, pp. 628-637, 2020. DOI: 10.7840/kics.2020.45.3.628

[7] Woojin Seok, Jeonghoon Moon, and Sanggeun Lee, "A Study on Big Data Transfer over Security-centric Network" Proceedings of Symposium of the Korean Institute of communications and Information Sciences, pp. 794-795, 2021.

[8] Seokhwan Kang, "Proposal of data transfer software process with enhanced security," Journal of KIISE, pp. 883-885, 2021.

[9] WAN- KI KIM, "Secure Vehicle OTA Software Update Design," Korea Computer Congress, pp. 197-199, 2021.

[10] Hyunjoo LIM, Jeehun PARK, Kyeonghwa Jeong, Youngsu KIM, Kyungho LEE, Changwon KIM, and Jincheol KIM, "Design Automotive Update System Architecture Using OTA(Over The Air)," The Korean Society of Automotive Engineers, pp. 662-665, 2018.

[11] Lockheedmartin, https://www.lockheedmartin.com/en-us/news/features/2020/devsecops-digital-engineering-aegis-of-tomorrow.html

[12] Ahn, Jae-Won, Choi, Beom-Jin, Ok, Sung-Jin, Kang, Jung-Ha, Kim, Jae-Young, and Kim Eun-Gi, "Design and implementation of file transfer protocol supporting security functionalities" Journal of the Korea Academia-Industrial cooperation Society, Vol. 15, No. 5, pp. 3086-3092, 2014. DOI : 10.5762/KAIS.2014.15.5.3086

[13] LAMBERGER, MENDEL, Florian, “Higher-order differential attack on reduced SHA-256”, Cryptology ePrint Archive, 2011.
[14] StarLink, https://www.starlink.com/

Authors

Byeong-Wan Lee received the B.S. degrees in Information and Communication Engineering from Chungbuk National University, Korea, in 2016. He is currently working in Hanwha Systems Co. from 2016

He is interested in Naval Combat Management Systems, Hash Algorithm, File Security and so on.