정 보 보 호 학 회 지
제25권 제5호, 2015. 10

PLC 기반의 제어시스템 취약점 분석 방법

김 동 욱*, 민 병 길**, 박 현 동***, 박 상 우****

요 약

PLC(Programmable Logic Controller)는 기반시설 운영 현장의 다양한 환경을 견딜 수 있도록 설계된 소규모 산업용 컴퓨터로써, 많은 제어시스템에서 사용되고 있다. 과거의 PLC가 장치 제어를 위한 목적으로 제한된 필수 프로그램만을 사용하였다. 하지만 최근의 PLC는 ethernet 등의 네트워크를 기반으로 다양한 IT 기술들이 적용되고 있다. PLC 기반의 제어시스템은 운영자 및 관리자에게 편의성을 제공하지만 여러 보안 취약점들이 내재되어 있을 수 있다. 취약점들이 공격자에게 노출되어 사이버 위협을 받을 경우, 심각한 보안 사고를 일으킬 수 있으므로, 이러한 취약점들을 찾고 보안 대책을 마련하여 적용하는 것이 필요하다. 그러나 PLC의 특성 및 운용 환경 등이 IT 장비들과는 상이한 점이 있어 IT 장비들을 위한 취약점 분석 방법을 그대로 PLC에 적용하는 것은 한계가 있다. 따라서, 본 논문에서는 PLC의 특성 및 운용 환경 등을 고려한 PLC 취약점 분석 방법에 대해서 제안하고자 한다.

Ⅰ. 서 론

PLC(Programmable Logic Controller)는 가빈시설 운영 현장의 다양한 환경을 견딜 수 있도록 설계된 소규모 산업용 컴퓨터로써, 요즈음 거의 모든 제어시스템(ICS, Industrial Control System)에서 사용되고 있다. NEMA(National Electric Manufacturers Association) 표준 ICS3-1978의 기술되어 있는 PLC(Programmable Logic Controller) 정의에 따르면 “PLC는 디지털 또는 아날로그 모듈을 통해 다양한 타입의 기기와 프로세스를 제어하는 특수 기능(로직, 시퀀싱, 타이밍, 카운팅, 연산)의 구현을 위한 명령들의 내부 저장을 위해 프로그래밍 메모리를 사용하는 디지털 운영 전자 기기“를 의미한다.

PLC는 직·간접적으로 연결된 입출력장치(예, 밸브, 스위치 등)들의 입출력과 관련한 연산을 수행하기 위해, 제어로직 또는 제어프로그램이라 불리는 소프트웨어를 사용하며, 제어프로그램을 구동하는 VxWorks 등의 임베디드 RTOS를 탑재하고 있다. 또한 PLC는 연결되어 있는 입·출력장치 또는 다른 PLC 및 EWS 등과의 통신을 위해 필드버스 프로토콜을 사용할 수 있다. 그리고, 최신 PLC의 경우 운영자에게 편의성을 제공해 주기 위한 다양한 IT 기술들(예, TCP/IP, FTP, 웹서버, 원격 관리 서비스 등)이 적용되고 있다. 이렇듯, PLC에 다양한 기능들이 탑재되면서 복잡도가 증가하였고, PLC 설계 및 구현 취약점[1-3]이 발생할 가능성 또한 높아졌다. 공격자는 PLC 관련 취약점들을 악용하여, 입·출력장치, PLC 또는 주변 장치들의 고장을 유발, 인명 피해 또는 경제적인 손실이 발생할 수 있다.

스턱스넷(Stuxnet)[4] 원자력발전소의 제어시스템을 겨냥한 제어시스템 대상의 최초의 악성코드이다. 스턱스넷은 이란 나탄즈 원자력시설에서 사용하는 PLC 제어 프로그램을 조작하여, 원심분리기를 1000여기 정지시켰다. 이로 인해 이란의 핵 프로그램은 2년 이상 지연되었다. 또한 스턱스넷 이외에도, 듀크(duqu), 플레임(Flame), 가우스(Gauss), 샤문(Shamoon), 및 스카이와이퍼(SkyWiper) 등의 새로운 악성코드가 지속적으로 발견되어 제어시스템을 위협하고 있는 상황이다.

이제 제어시스템은 사이버 위협으로부터 안전하다고 말할 수는 없다. 제어시스템의 보안사고의 발생가능성을 줄이고, 보안피해가 발생하였을 때 신속하게 대처하는 것이 필요하다. 제어시스템의 보안사고 발생가능

* 국가보안기술연구소 (dwkim1980@nsr.re.kr)
** 국가보안기술연구소 (bgmin@nsr.re.kr)
*** 국가보안기술연구소 (hdpark@nsr.re.kr)
**** 국가보안기술연구소 (psw@nsr.re.kr)

성을 줄이기 위해서는 제어시스템의 알려진 혹은 알려지지 않은 취약점들을 점검하거나 찾고 보안 대책을 마련하여 적용하는 것이 필요하다. 본 논문에서는 제어시스템의 핵심장비인 PLC가 설치되어 운용되는 실 환경(예, 공장 및 발전소)의 취약점 분석 시 활용 가능한 취약점 분석 방안을 제시하고자 한다.

이를 위해 본 논문에서는 다음 내용들에 대한 연구를 수행하였다. 첫째, 일반 PLC의 특성 분석을 수행하였다. PLC에 지식이 없는 보안전문가들은 PLC에 대한 기본 정보를 습득할 수 있다. 둘째, PLC 위협 모델 및 영향 분석을 수행하였다. PLC 운영자는 공격자가 PLC에 어떠한 사이버보안 위협을 가할 수 있는지 파악할 수 있으며, 또한 사고 발생 시 피해 정도를 가늠할 수 있다. 마지막으로 PLC 취약점 점검 대상을 식별하고, 선정된 대상들에 대한 취약점 점검을 위해 '환경설정 취약점 점검 항목'과 '설계 및 구현 취약점 점검항목'을 각각 제시하였다. 그리고 실제 PLC가 운용되고 있는 환경에서 PLC 취약점 점검 및 보안 대책 적용 시 고려해야 할 사항에 대해서 설명하였다.

본 논문의 남은 장에서 기술될 내용은 다음과 같다. 2장은 일반 PLC의 특성을 분석하고, 3장에서는 PLC 취약점 분석 방안에 대해서 기술한다. 본 장에서는 위협모델 및 영향 분석, 취약점 점검 대상 및 점검 항목 식별, 취약점 점검 및 보안 대책 적용 시 고려사항에 대해 기술한다. 마지막으로 4장에서 결론을 맺는다.

Ⅱ. PLC 특성 분석

2.1. PLC 특징

PLC를 사용하는 제어시스템을 기존의 산업시스템과 비교했을 때 가장 큰 특징은 바로 소프트배선(softwiring)을 사용한다는 점이다. 기존 제어시스템의 경우, 제어장치 또는 입력장치(예, 스위치)와 운용장치 또는 출력장치(예, 조명기구) 간에는 물리적인 직접 배선(wiring)이 이루어졌다. 하지만, PLC를 사용하는 제어시스템에서는 입력장치 및 출력장치들이 PLC에 연결되며, PLC내 소프트웨어인 제어로직 또는 제어프로그램을 통해 입력장치와 출력장치 간에 연결이 설립한다. 이를 소프트배선이라 부르며, 제어프로그램 변경을 통해 장치 간 연결 또는 동작 등을 비교적 손쉽게 변경할 수 있는 장점을 지닌다.

〔표 1〕 PLC 특징

특징	설명
실시간성	정해진 시간 제약 안에 제어프로그램의 실행 결과 값을 도출해야 함
고안정성	제어시스템의 경우 가용성이 매우 중요하므로, PLC는 높은 안정성이 필요함
내환경성	온도, 진동, 습도 등 다양한 환경적 요인에 강한 내성이 필요함
취급의 용이성	PLC 설치, 구성, 공정 변경에 있어 취급이 용이해야함
경제성	기존 릴레이제어반 대비 높은 경제성을 지님

이 외에도 PLC는 [표 1]의 특징들을 지닌다.

2.2. PLC 주요 구성요소

PLC의 주요 구성요소는 크게 운영체제, 제어프로그램, 런타임 시스템, 필드버스 통신, 관리 인터페이스로 나눌 수 있으며 각 구성요소에 대한 설명은 다음과 같다.

PLC의 운영체제는 메모리 관리, 파일 시스템, 디스크 액세스, 타임 틱, 부트로더, 통신인터페이스, 표준 코드 라이브러리, 네트워킹과 같은 기능들을 제공한다. PLC에서 사용되는 운영체제는 임베디드 RTOS(Real Time Operating System)로써, 주로 VxWorks, microware OS-9등이 사용된다. 예를 들어, Allen-Bradley Controllogix PLC는 VxWorks 운영체제가 탑재되어 운용되고 있으며, Alle-Bradley PLC5는 Microware OS-9가 탑재되어 있다.

제어프로그램 혹은 제어로직은 PLC에서 실행되는 프로그램으로서 PLC와 직·간접적으로 연결된 입·출력 장치들의 입력 및 출력과 관련한 연산을 수행하기 위해 사용된다. 예를 들어, 수조에 담긴 액체가 일정 수준에 도달하면 밸브를 여는 것과 같은 동작을 구현한다고 가정하자. 이 때, 입력장치는 수조의 액체가 담긴 정도와 관련한 값을 주기적으로 PLC로 전달해주는 역할을 한다. PLC 제어 프로그램은 이 값이 일정 수준 이상인지를 확인하여, 물이 일정 수준 도달 했다고 판단하면, 밸브 출력장치로 밸브를 열라는 신호를 전달할 수 있다. 입력장치로부터 입력 신호를 받고, 필요한 제어 연산 후에, 출력 장치로 출력신호를 전달하는 일련의 과정을 제어프로그램으로 구현할 수 있다. 이러한 제어프로그램은 PLC 프로그램 또는 제어로직으로도 불린다. 이러

한 제어프로그램은 보통 IEC 61131-3(PLC 프로그래밍 언어)표준을 준수하여 구현된다.

런타임 시스템은 일반 임베디드 장치를 PLC로 만드는 다양한 기능(PLC와 IDE간 통신, 제어프로그램 호출, 디버깅, 입/출력)을 장치에 제공하는 소프트웨어 패키지이다.

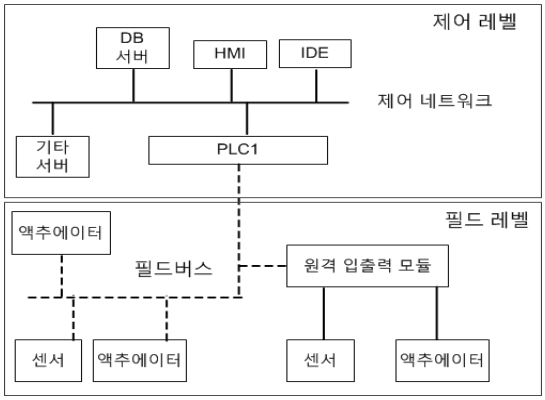
필드버스는 PLC와 입·출력장치 및 PLC 간의 통신을 위해 사용되는 산업용 컴퓨터 네트워크 프로토콜 집합을 통칭하는 이름으로, 현재는 IEC 61158으로 표준화되었다. 널리 사용되는 필드버스 프로토콜은 Modbus, Profibus, CAN(controller area network), AS-I(Actuator Sensor Interface) 등이 있으며, 최근 대부분의 PLC는 두 개 이상의 필드버스 프로토콜을 지원한다. 그리고, 스마트센서라고도 불리는 최신 입·출력 장치에는 필드버스 통신 지원을 위한 통신 모듈이 장착되어 있다.

관리 인터페이스는 웹 관리, 텔넷/SSH, SNMP, FTP 서비스 및 이와 유사한 원격 관리 서비스를 제공하는 인터페이스이다. 현재 여러 PLC에서 웹서버, SNMP, 텔넷, FTP 서버 등의 관리 편의를 위해 서비스를 제공하고 있어, 운영자는 이러한 외부 인터페이스를 통한 장치 관리 서비스를 활용하여 PLC 설정을 변경할 수 있다. 이러한 기능들은 운영자에게 PLC를 쉽게 모니터링하고 관리하도록 도와준다.

2.3. 제어시스템 네트워크 구조

[그림 1]는 PLC를 포함하는 제어시스템 네트워크 구조의 일부로, 크게 제어레벨과 필드레벨로 나뉜다. 일반적인 제어시스템은 제어레벨의 상위 레벨인 관리레벨을 포함하고 있으나, 본 논문에서는 제외하였다.

제어 레벨의 주요 구성요소로는 HMI(Human Machine Interface), IDE(Integrated Development Environment), DB 서버 등이 있다. HMI는 PLC 공정과 연관된 데이터를 사람이 인지할 수 있는 형태로 나타내고, PLC 공정을 제어할 수 있도록 지원한다. IDE는 제어프로그램을 구현하기 위한 개발환경(구현, 컴파일, 디버깅)을 제공하는 툴이며 EWS(Engineering WorkStation)이라고도 불린다. DB서버는 PLC 공정 데이터 및 로그 데이터 등을 저장한다. HMI, DB서버, IDE는 모두 같은 컴퓨터에서 구동될 수 있다. 그리고 제어 네트워크로 이더넷이 사용될 경우, 해당 네트워크



〔그림 1〕 제어시스템 구조(일부)

에는 라우터/스위치 등의 네트워크 장비가 사용될 수 있다.

필드레벨은 입·출력장치와 원격 입출력 모듈로 구성된다. 입력장치는 보통 센싱을 하는 센서 또는 스위치이며, 출력장치는 장비를 구동하는 엑추에이터이다. [그림 1]에서 보듯이 입·출력장치는 PLC의 입출력 모듈 또는 원격 입출력 모듈에 연결된다. 입력장치는 그 장치에 따라 PLC의 통신모듈 또는 입력모듈에 연결되어 자신의 입력신호를 전달할 수 있다. 통신모듈이 없는 구형의 입력 장치는 PLC의 입력모듈에 연결되어 아날로그 신호를 송신하며, 이 때 PLC는 아날로그 신호를 디지털 신호로 변환하여 사용한다. 통신 모듈이 있는 입력 장치는 디지털로 변환된 신호를 필드 버스 통신을 이용하여 PLC의 통신 모듈로 송신한다. 출력장치의 경우도 마찬가지이다. 그리고 원격 입출력 모듈은 원거리의 입·출력장치를 상위 PLC에 효율적으로 연결하기 위한 모듈로서, PLC의 제어프로그램을 수행하지 않는점을 제외하고는 PLC와 유사하다. 즉, 입력장치로부터 신호를 수신하여 상위 PLC로 전달하고, 상위 PLC의 출력장치 제어 신호를 받아 자신에게 연결된 출력장치에게 전달하는 역할을 한다. 이 때, 입력장치는 원격 입출력 모듈의 통신모듈 또는 입력모듈에 연결될 수 있다.

2.4. PLC 스캔

PLC는 메모리에 기록되어 있는 입력장치의 입력신호를 읽고, 제어프로그램을 수행하여, 출력장치가 수행해야 할 신호를 메모리에 기록한다. 이 후, 운영체제는

오류 발생 등을 확인하기 위한 진단 기능을 실행하고, 와치독 타이머 값 등의 타이머를 갱신한다. 이 동작과정을 일컬어 PLC 스캔이라고 하며, PLC는 이 과정을 지속적으로 반복 실행한다.

[표 1]에서 PLC의 특징으로 실시간성을 언급했었는데, 실시간성을 보장하기 위해서는 제어프로그램을 정시에 수행하는 것이 중요하다.

2.5. PLC 표준

IEC(International Electrotechnical Com ‐ mision)에서는 PLC를 구성하는 통신 네트워크와 프로그래밍 언어가 서로 달라 발생하는 불편함을 해소하기 위해 PLC 국제 표준화 규격(IEC 61131)을 8파트([표 2])로 나누어 제정하였다. 이 중 다음 Part 3과 Part 5에 대해서 살펴본다.

〔표 2〕 PLC 국제표준화 규격

구분	설명
Part 1	PLC의 기본 기능 및 용어 정의
Part 2	PLC 하드웨어 정의와 요구사항
Part 3	PLC 프로그래밍 언어에서 의미와 문장 구조 정의
Part 4	사용자 지침
Part 5	PLC 내·외부와의 통신 및 네트워크
Part 6	기능 안전
Part 7	퍼지 제어 프로그래밍
Part 8	프로그래밍 언어의 적용 및 구현을 위한 지침

2.5.1. PLC 프로그래밍 언어(IEC 61131-3)

과거 PLC 제어프로그램을 작성하기 위한 프로그래밍 언어는 PLC 제조사 별로 달라서 호환성 문제 등 많은 문제들이 발생하였다. 이러한 문제를 해결하기 위해 1993년 IEC에서 IEC 61131-3을 표준으로 제정하였으며, 해당 표준에는 프로그래밍 언어로 2가지 문자기반(Instruction List, Structured List) 언어 및 3가지 도형기반(Ladder Diagram, Function Block Diagram, Sequential Function Chart)의 언어를 정의하였다. 어떤 언어를 사용하더라도 동일한 기능을 하는 프로그램을 구현할 수 있기에, 사용자는 편의에 따라 선정하여 사용할 수 있다.

IEC 61131-3에서는 다양한 표준 데이터 타입, 표준 기능, 표준 기능 블록들을 제공하며, 사용자는 이러한 요소들을 활용하여 유도 데이터 타입, 기능, 기능 블록들을 생성할 수 있으며, 이들을 활용하여 제어프로그램을 구현할 수 있다.

2.5.2. PLC 네트워크 통신 (IEC 61131-5)

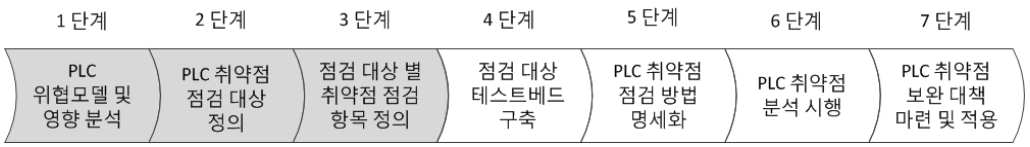
IEC 61131-5의 파트 5는 PLC의 통신 측면에 대해 명시하고 있다. PLC는 제어시스템의 나머지 시스템에게 여러 정보를 제공(서버)하거나 다른 PLC들에게 정보를 요청(클라이언트)할 수도 있다. 통신 시스템을 통해 PLC의 여러 하위 시스템들(예, I/O 서브시스템, 프로세싱 유닛, 파워 서플라이 서브시스템, 메모리 서브시스템 통신 서브시스템 등)에 대한 상태 정보들을 제공할 수 있다. 이러한 정보들은 필드버스 프로토콜을 사용하여 전달된다. 예를 들어, PLC들은 이더넷 기반으로 연결될 수도 있으며, 이 경우 필드버스 프로토콜은 응용계층에 구현될 수 있다.

Ⅲ. 제어시스템 취약점 분석 방안

[그림 2]은 제안하는 제어시스템 취약점 분석 모델로서, 일반적인 취약점 분석 모델과 유사하다. 취약점 분석 모델은 제어시스템 위협 모델 및 영향 분석, 제어시스템 취약점 점검 대상 식별 , 점검 대상 별 취약점 점검 항목 식별, 점검 대상 테스트베드 구축, 취약점 점검 방법 명세화, 취약점 분석 시행, 및 보완 대책 마련 및 적용 등의 7 단계로 구성되어 있다. 이 중 1~3단계는 대부분의 PLC 기반의 제어시스템에 공통으로 적용될 수 있는 일반 제어시스템에 대한 보안위협 및 영향, 점검 대상, 점검 항목들에 대한 내용이다. 4~7단계는 점검 대상이 되는 PLC가 설치되어 운용되고 있는 제어시스템 환경(예, 원자력발전소 등)에서 취약점 분석 및 보완대책 적용 시에 수행해야 할 단계들이다. 본 논문에서는 1~3단계에 대해 중점적으로 다룰 예정이다.

3.1. 제어시스템 보안위협 및 영향분석

본 절에서는 PLC 기반의 제어시스템 취약점 점검 단계([그림 2])의 첫 번째 단계로 PLC에 발생할 수 있는 PLC 보안위협을 식별하고 해당 위협들이 PLC 및



〔그림 2〕 PLC 기반 제어시스템 취약점 분석 단계

이를 적용하는 제어시스템에 끼칠 수 있는 영향에 대해서 살펴본다. 이는 PLC 취약점 점검 및 대응방안 수립의 필요성을 인지하는데 그 목적을 두고 있다.

3.1.1. 제어시스템 보안위협

PLC 위협모델은 PLC 및 주변 디지털 장치([그림 1]의 제어레벨에 위치하는 장치)에서 고려해야할 사안 및 발생 가능한 공격 또는 위협들의 집합을 정의한 것으로서, 공격자의 PLC 및 주변 디지털 장치에 대한 세부공격대상 및 목표를 식별하기 위해 필요하다.

Morten 등[8]은 마이크로소프트사의 STRIDE 위협분류[9]를 기반으로 PLC의 위협모델을 정의하였다. 그러나 [8]에서는 주변 디지털 장치에 대한 위협모델을 고려하지 않은 한계점이 있다. 공격자는 주변 디지털 장치를 PLC 공격의 중간 도구로 활용할 수 있기 때문에 주변 디지털 장치에 대한 위협모델 식별도 필요하다. 2장에서 기술한 PLC의 특성 분석을 통해 습득한 여러 정보들을 기반으로 STRIDE 모델 각 분류 항목에 대해 PLC에 영향을 일으킬 수 있는 보안위협들을 세분화하였다. [표 3]은 가능한 PLC 보안위협을 나타낸다.

3.1.2. 제어시스템 보안위협 영향

본 절에서는 [표 3]의 제어시스템 보안위협 중 '변조' 및 'DoS'분류에 해당하는 보안위협들에 대해, PLC 및 제어시스템에 미치는 영향에 대해 살펴본다.

〔표 3〕 PLC 보안위협

STRIDE 분류	위협
스푸핑	인증을 우회할 수 있는가? Credential을 가로챌 수 있는가? PLC와 IDE/OPC/HMI간 세션을 제어할 수 있는가? PLC를 중단/재시작하는 것이 가능한가? 제어프로그램을 제거하는 것이 가능한가?
변조	다음을 변조할 수 있는가? - PLC 환경설정(와치독 타이머 설정 값 등) - PLC 입출력 모듈의 I/O 값 - 메모리 값(PLC, 원격 입·출력 모듈, 입·출력 장치) - 제어프로그램, 런타임 시스템 - 제어프로그램 삽입 - PLC 펌웨어 설정 변경 - 필드버스 통신 펌웨어 (PLC, 원격 입·출력 모듈, 입·출력 장치) 설정 변경 - HTML 파일, 운영 체제, - 네트워크(필드버스 등) 통신 트래픽 - 로그 파일 - 패스워드
부인 방지 (로그)	다음 작업에 대한 로그가 남는가? - 파일 시스템 액세스 로그 - 환경설정 파일 읽고 쓰는 작업 로그 - 제어프로그램 업/다운로드 로그 - 운영 체제 시작/중단/재시작 로그 - 제어프로그램 시작/중단/재시작 로그 - 관리서비스 시작/중단/재시작 로그 - I/O 값을 읽거나 쓰는 작업 로그 - 기타 프로그램 설치/삭제/사용 로그 - 명령어 히스토리 로그
정보 획득	다음에 대한 정보 획득이 가능한가? - 패스워드, 환경설정 파일, 파일시스템, - 제어프로그램, 펌웨어, 로그파일, - I/O값 획득, I/O 장치 정보, 메모리 값 - 네트워크 트래픽(필드버스, 이더넷 등)
DoS	DoS 공격을 통해 다음의 수행이 가능한가? - 원격 관리 서비스 중단, 제어프로그램 중단 - 통신 중단, 워치독 타이머 중단 - PLC와 IDE/HMI/OPC간 세션 중단 또는 통신 장애 - PLC 장애 - PLC와 입·출력 장치 간 통신 장애
권한 상승	다음의 공격을 수행하는 것이 가능한가? - PLC 또는 주변 디지털 장비의 루트권한 획득 - 인증된 사용자에 대한 세션정보 획득 - 원격 관리 서비스 시작/중단 및 이용 - 제어프로그램 시작/중단 - 제어프로그램 업/다운로드 - 와치독 타이머 시작/중단 - (통신)펌웨어 임의 설치 - 임의 프로그램 설치

□ 와치독 타이머 설정값 변조

와치독 타이머는 일반적으로 하드웨어 타이머로서, 제어프로그램이 적시에 수행이 완료되는지를 확인하기 위해 사용된다. 즉, 와치독 타이머는 제어프로그램 스캔 시간이 설정된 값(예, 100ms)을 초과할 경우 와치독 이벤트를 발생시켜 런타임 시스템으로 전달한다. 이를 수신한 런타임 시스템은 제어프로그램을 중단하고 에러를 리포팅한 후 제어프로그램을 재 시작한다. 예를 들어 해당 이벤트에 대한 로그를 남기거나 혹은 HMI(Human Machine Interface)를 통해 알림 메시지를 전송함으로써 이벤트 발생을 운영자에게 알릴 수 있다.

만약 와치독 타이머의 설정 값을 공격자가 임의로 큰 값(예, 10초)으로 변경할 수 있을 경우, PLC 고장 등의 장애가 발생하여 PLC 스캔을 적시에 완료하지 못한 문제가 발생했을 시, 운영자는 이를 알아채기 어렵다. 따라서 PLC는 더 이상 실시간성을 보장할 수 없게 된다. 또한 PLC를 사용하는 제어시스템의 문제발생 등 운영 전반에 문제를 유발할 수도 있다.

□ 런타임 시스템 변조

공격자가 런타임 시스템을 변조할 경우 발생하는 문제에 대해서 살펴보자. 3.1.1절에서 설명했던 바와같이 런타임 시스템은 와치독 이벤트를 받고, 에러 리포팅을 한 뒤 제어프로그램을 재 시작한다. 에러 리포팅 정보는 운영자가 PLC 운영상 문제 발생 여부를 판단할 때의 근거자료로 사용할 수 있다.

그러나, 공격자가 와치독 이벤트를 수신한 런타임 시스템이 아무런 처리 작업을 수행하지 않도록 변조할 수 있을 경우, '와치독 타이머 설정값 변조'에서 설명한 것과 동일한문제가 발생할 수 있다.

□ 제어프로그램 변조

공격자가 제어프로그램을 변조할 수 있는 경우 발생하는 문제에 대해서 살펴보자. 제어프로그램은 PLC에 연결된 입・출력 장비들을 연결하고 제어하는 프로그램으로써, 소프트배선을 가능하게 하는 핵심 프로그램이다. 이러한 프로그램이 공격자에 의해 변조 가능할 경우, 장치들을 공격자가 원하는 임의 동작을 수행하도록 변경할 수 있다. 따라서, PLC와 연결된 장치들이 의도치 않게 동작하여, 생산에 차질을 일으키거나 고장 등의 문제를 야기할 수 있다. 예를 들어, Stuxnet[4]은 제어프로그램 변조로 인해 발생한 대표적인 사고이다. PLC에 연결된 원심분리기의 모터 회전수를 증가시키도록 제어하여 과부하를 일으킴으로써 약 1,000여개에 달하는 원심분리기를 파괴하였다.

□ 메모리값 변조

공격자가 PLC의 메모리 값을 변조할 수 있을 때 발생하는 문제에 대해서 살펴보자. PLC의 메모리는 제어프로그램의 저장을 위해서 뿐만이 아니라 입·출력장치의 데이터 값을 저장하기 위해서도 사용된다. 여기서 입력장치의 데이터 값은 입력장치로부터 받은 신호를 PLC가 이해할 수 있는 데이터로 변환한 값을 의미하며, 출력장치의 데이터 값은 출력장치가 이해할 수 있는 신호로 변환하기 이전의 값을 의미한다.

PLC가 초기화 될 때, 입출력장치의 데이터 값 저장을 위한 메모리 영역이 할당되며, 이를 입력 상태 파일 또는 출력 상태 파일이라고 부른다. PLC 제어프로그램은 입력 상태 파일의 값을 읽고 프로그램을 실행하며, 그 결과 값을 출력 상태 파일에 기록한다. 만약, 공격자가 작성한 임의의 악성 제어프로그램을 설치하거나 원격으로 입/출력 상태 파일을 변경할 수 있을 경우, 공격자의 의도대로 PLC와 연결된 입·출력장치들을 동작하도록 만들 수 있다. 물론 메모리 값 변조를 위해서는 입출력 상태 파일과 입출력 장치간의 매핑이 어떻게 이루어져 있는지 등에 대한 내용을 파악해야 한다.

□ 필드버스 트래픽 변조 및 정보획득

제어시스템에는 다양한 종류의 필드버스 프로토콜(2.2절)들이 사용되고 있다. 그러나 대부분의 프로토콜은 일반적으로 보안을 고려하지 않고 설계되어 여러 취약점을 지니고 있다.

Modbus 프로토콜을 예로 들어보자. Modbus는 마스터-슬레이브 구조를 가지는 응용 계층 프로토콜로써 많은 제어시스템에서 사용되고 있는 사실상의 표준이다. 이 때 물리계층은 시리얼이나 이더넷을 사용할 수 있다. Modbus는 설계 시 보안을 고려하지 않은 프로토콜 중 하나로, 모든 메시지들이 암호화되지 않고 전송되어 기밀성 문제를 야기할 수 있으며, 마스터와 슬레이브 장치 간 인증 없이 통신이 수행되므로 중간자 공격(MITM, Man In The Middle)에 취약하다[11]. 즉, 공

격자가 네트워크에 액세스할 수 있다면, modbus의 트래픽 정보 획득 및 변조 모두 가능하며, 이를 통해 공격자는 메모리 값 변조(3.1.2절) 및 PLC의 장애를 일으키는 DoS 공격(3.1.2절)을 유발할 수 있다.

□ PLC 펌웨어 변조

네트워크를 이용한 원격 펌웨어 업데이트는 널리 분산되어 있는 여러 PLC들에 대한 펌웨어 업데이트를 용이하게 하는 기능이다. 하지만, 원격 펌웨어 업데이트 시에 무결성 검증 및 인증이 적절히 사용되지 않을 경우, 펌웨어 변조 공격[8]을 수행할 수도 있다. 펌웨어 변조 공격의 성공은 공격자가 PLC를 완전히 제어할 수 있음을 의미하며, 이를 통해 백도어 또는 루트킷 등의 운영자가 알지 못하는 악성코드를 설치하거나, 제어프로그램 또는 관리서비스를 종료시킬 수도 있다.

□ PLC 장애

최근 PLC는 웹서버, FTP 서버, SNMP등 IT에서 널리 사용 중인 여러 서비스들을 운용 및 관리상의 목적으로 이용하고 있다. 이러한 프로그램 또는 프로토콜은 이미 다수의 취약점을 가지고 있으며, 그 중에는 DoS를 유발하는 취약점들도 있다. 공격자는 DoS 공격을 악용하여 PLC 장애를 일으킬 수 있다. 예를 들어, 최근 원격에서 DoS 공격이 가능한 아파치 웹서버 취약점(CVE-2013-1896)이 발견되었는데, 공격자는 IDE 등에서 원격으로 아파치 웹서버를 설치한 PLC 시스템의 자원(CPU, 메모리 등)을 소모시켜, 와치독 타이머가 만료될 때까지 PLC 스캔을 완료하지 못해 PLC의 제어프로그램이 종료되는 문제가 발생할 수 있다. 이는 출력장치를 제대로 구동하지 못하는 문제로 이어져 재정상의 손실을 끼칠 수 있다.

3.2. PLC 취약점 점검 대상 및 항목

3.2.1. PLC 취약점 점검 대상 식별

이제 PLC 기반의 제어시스템 취약점 점검 단계([그림 2])의 두 번째 단계인 취약점 점검 대상에 대해서 살펴보자. 일반적으로 하나의 제어시스템에는 수많은 디지털 장치들이 산재해 있다. 모든 장치들에 대해서 점검하는 것은 많은 시간이 소요될 뿐만 아니라 효율성도 떨어진다. 즉, 점검을 위해 주어진 시간이 매우 많다면, 모든 장치들에 대해서 점검하는 것이 보안을 위해서 좋은 방안이나, 시간이 한정적인 경우, 제어시스템에 중대한 보안사고를 일으킬 수 있는 장치들을 우선적으로 식별하고, 식별된 장치를 중점적으로 점검을 수행하는 것이 효율적이다. 따라서, 취약점 점검을 수행하기 전, 점검하려는 제어시스템의 취약점 점검 대상을 식별하는 것은 중요한 작업 중 하나이다.

[표 4]는 식별된 취약점 점검 대상을 나타낸다. 점검 대상은 PLC 및 PLC 주변 장치, 그리고 PLC와 주변장치 간 통신에 사용되는 통신 프로토콜들로 분류하였다. 취약점을 악용한 사이버공격 발생 시, PLC 및 필드장치에 직·간접적인 영향을 줄 수 있는 대상인지를 점검 대상의 식별 기준으로 하였다.

취약점 점검 대상을 PLC 자체만으로 국한하지 않은 이유는 PLC 외부에서 시작되어 연쇄적인 효과로 PLC 및 입·출력장치에 직접적인 영향을 줄 수 있기 때문이다. 예를 들어, HMI에 취약점이 존재할 경우 공격자는 HMI를 운용하는 PC(PC1)에 영향을 줄 수 있다. 그리고, HMI와 PLC-HMI 통신 프로토콜 모두에 취약점이 존재할 경우, 공격자는 PC1을 PLC 공격을 위한 중간 경로로 이용할 수 있다. 또한 PLC를 장악한 공격자는 필드버스의 취약점을 찾아 입·출력모듈을 조작하여 입·출력장치에 장애를 일으킬 수도 있다. 이처럼 피해는 PLC를 구성하고 있는 여러 대상들의 취약점들을 악용하여 연쇄적으로 발생할 수 있기 때문에 점검 대상

〔표 4〕 취약점 점검 대상

분류	점검 대상
PLC	운영체제
	런타임 시스템
	관리서비스(웹 서버,FTP 서버 등)
	제어프로그램
PLC 주변장치	PC(HMI, IDE, OPC 등의 소프트웨어 운용 장비)
	라우터/스위치
	입·출력모듈
통신 프로토콜	PLC-HMI/IDE/OPC 통신 프로토콜 (응용계층)
	필드버스 프로토콜(PLC, 입·출력 모듈, 입·출력장치 간 통신 프로토콜)

을 [표 4]와 같이 구성하였다. 각각의 점검 대상에 대해 3.2.2절에서 기술할 취약점 점검 항목을 기준으로 취약점 점검을 수행한다.

3.2.2. PLC 취약점 점검항목 식별

본 절에서는PLC 기반의 제어시스템 취약점 점검 단계([그림 2])의 세 번째 단계인 식별된 취약점 점검 대상에 대한 점검 항목을 정의한다. 일반적으로 취약점 점검 항목은 크게 관리, 물리, 기술적 분류로 나뉠 수 있는데, 본 논문에서는 기술적 항목에 초점을 맞춘다. 제안하는 취약점 점검 항목은 크게 1)‘환경설정 취약점 점검 항목’과 2)‘설계·구현 취약점 점검항목’으로 나뉜다. 고수준의 PLC 취약점 분석 결과를 도출하기 위해, 환경 설정에 관한 취약점 점검과 구현상의 취약점 점검 모두 수행할 필요가 있다.

환경설정 취약점 점검 항목은 PLC 및 주변장치의 환경설정과 관련한 항목들을 의미하며, 해당 취약점 점검 항목에 대한 분석 및 보완을 통해 이미 알려진 보안 취약점들에 의해 발생하는 위험을 최소화할 수 있다. 예를 들어, 최근 PLC를 운용하는 망이 인터넷 망과 연계되어 있는 것[5]이 발견되었는데 이는 설정상의 취약점으로 볼 수 있다. 환경설정 취약점 점검 항목은 주요정보통신기반시설 취약점 분석·평가의 점검항목[10]을 참고하여 도출하였다. [10]에는 제어시스템 및 다양한 IT기술(웹 서버, SNMP, FTP 서버, 텔넷, 데이터베이스, 윈도우즈 등)에 대한 취약점 점검 항목들이 정의되어 있다. 본 논문에서는 이러한 취약점 점검 항목들 중에서 PLC 취약점 점검 대상의 취약점 점검 시 적용할 수 있는 항목들을 추출하였으며, [표 5]는 이를 나타낸다.

〔표 5〕 환경설정 취약점 점검 항목

점검 대상	점검항목
PLC 관리 서비스 점검	ssh 원격접속 허용 여부
	ftp 디렉토리 접근권한 설정
	ftp 계정 shell 제한
	ftp users 파일 설정
	SNMP 서비스 구동 점검
	apache 웹서비스 정보 숨김
	HTTP/FTP 배너차단
	불필요한 서비스(telnet 등) 사용 여부
	telnet 보안 설정(타임아웃 값 등)
	Anonymous FTP 비활성화
	최신 SNMP 버전 사용(구버전은 취약)
PLC 운영체제	취약한 비밀번호 사용
	불필요한 계정 사용
	원격접속 설정
	인증 및 인증 정보 관리
	환경설정파일 관리
	로그 여부 및 로그 파일 관리
	접근 제어 설정 여부
	파일 및 디렉토리 관리
	최신 패치 적용
라우터/ 스위치	강력 패스워드 사용
	세션 타임아웃 값 설정
	최신 보안 패치 적용
	SNMP
	미사용 인터페이스의 셧다운 설정
	TFTP 서비스 차단
	Spoofing 방지 필터링 적용
PC	패스워드 주기적 변경(강력 패스워드 사용)
	불필요한 서비스/프로그램 제거
	공유폴더 제거
	바이러스 백신 프로그램 설치 및 주기적 업데이트
	원격 모드 비활성화
	CD/DVD/USB의 자동실행 방지
통신 프로토콜	암호화 통신 적용 여부(응용프로그램 자체 암호화, IPSec 또는 SSL 등의 일반 암호화, 또는 PLC 용 특수 암호화), 취약한 암호 알고리즘 사용 여부
	인증 여부
	최신 펌웨어/패치 적용 여부
	주요 로그 기록 여부(로그 암호화)
	인터넷 연결여부

이제 설계·구현 취약점 점검 항목에 대해서 살펴보자. 구현·설계 취약점 점검 항목은 PLC 전용 프로그램(런타임 시스템, HMI, IDE, OPC 등) 및 통신 프로토콜(modbus, profibus, deviceNet 등)들에 대한 아직 알려지지 않은 설계 및 구현상의 취약점 점검을 위한 항목들이다. PLC 전용 소프트웨어 프로그램 및 통신 프로토콜의 경우 보안에 대해 고려하지 않고 설계 및 구현되었기에 많은 취약점들이 존재할 수 있다. 예를 들어, ABB사의 AC500 PLC 웹서버 응용프로그램에서 발견

〔표 6〕 설계·구현 취약점 점검항목

점검 대상	점검항목
전용 프로그램 (IDE, OPC, HMI, FTP 서버 등), 런타임 시스템	부적절한 입력 값 검증 - 버퍼오버플로우 - 경계 체크 부재 - 경로 접근 공격, - OS 또는 SQL 명령 주입
	부실한 코드 품질 - 잠재적 위험 함수 사용 - Null pointer 역참조
	허가 및 접근 통제 - 설치된 PLC 구조에 적합한 접근 제어의 설정 여부
	부적절한 인증 - 인증 우회 가능 여부 - 주요 기능에 대한 인증 여부
	인증 정보 관리 - 하드 코딩된 인증 정보 존재 여부
	무결성 검증 - 다운로드 코드의 무결성 체크
웹서버 (PLC 관리서비스)	[10]의 [붙임2] '취약점 분석·평가 기본항목'의 '기술적 분야'에서 '아. 웹' 의 취약점 점검항목들
필드버스 프로토콜, PLC-HMI/ IDE/OPC 통신 프로토콜	MITM 공격 가능 여부 및 테스트 재전송 공격 가능 여부 및 테스트
	무결성 검증 - 수신 패킷의 무결성 체크

된 DoS를 유발할 수 있는 스택버퍼오버플로우 취약점[3]이 있다. 즉, PLC 보안성 강화를 위해서는 이러한 설계 및 구현상의 취약점들을 찾고, 취약점들을 보완한 패치를 개발하여 적용하는 것도 매우 중요하다. [표 6]은 PLC의 설계·구현 취약점 점검항목을 나타내며, 주요정보통신기반시설 취약점 분석·평가의 점검항목[10] 및 제어시스템 공통 취약점[11]을 참고하여 도출하였다.

3.3. PLC 취약점 점검 및 보완대책 적용 시 고려사항

지금까지 PLC 취약점 점검을 위해 필요한 여러 내용([그림2]의 1~3단계)들에 대해서 살펴보았다. 이제 PLC를 설치하여 운용하는 기반시설(예, 발전소 또는 공장 등)을 대상으로 실제의 취약점 점검 및 보완대책 적용 시 고려해야 할 몇 가지 사항들에 대해 살펴본다.

취약점 점검은 PLC가 설치되어 운용되는 실 환경에서 수행하는 것이 가장 정확한 점검 결과를 도출할 수 있다. 하지만, 실 환경에서 정해지지 않은 시기에 점검을 수행할 경우 가용성의 문제를 일으킬 수도 있다. 또한 실제로 사용되는 운영체제/프로그램 등에 따라 취약점 점검 방법이 상이할 수 있다. 따라서, 취약점 점검 수행에 앞서 점검 항목별 점검 환경의 선정과 적절한 점검 방법의 명세화 및 점검 시기 결정이 필요하다.

먼저, '환경설정 취약점 점검항목'의 경우 실 환경에서의 취약점 점검을 수행하면, 정확한 취약점을 식별할 수 있다. 그러나, '설계·구현 취약점 점검항목'을 점검하기 위해 실 환경에서 모의해킹 등을 수행하는 것은 제어시스템 가용성에 심각한 문제를 일으킬 수 있으므로, 실 환경 점검에서 '설계·구현 취약점 점검'을 수행하는 것은 어려우며, 대안으로 테스트베드를 꾸미고, 점검한다([그림 2]의 4단계). 이 경우, 실제 운영환경과의 유사도 정도에 따라 테스트베드를 기반으로 한 취약점 점검 결과의 신뢰도가 결정될 것이다.

그리고, 각 점검 항목별로 점검 방법을 명세화 할 필요가 있다([그림 2]의 5단계). 특히 '환경설정 취약점 점검항목'의 경우에는 잘못된 명령어 입력 등으로 인해 발생하는 문제를 최소화하기 위해서도 필요하다. 또한 명세화 된 점검 방법은 안전성이 검증 또는 입증된 방법이어야 한다. 취약점 점검에서의 안전성이 검증되지 않은 명령어 사용은 제어시스템에 문제를 일으킬 수도 있기 때문이다.

취약점 점검 시기는 기계나 시스템의 정비를 위해 예정된 정비 기간을 이용해야 한다. 예를 들어, 원자력 발전소의 경우 원전 가동을 중단하고 점검을 수행하는 계획예방정비 기간이 지정되어 있으며, 이 기간 동안 PLC의 '환경설정 취약점 점검'을 수행할 수 있다. '설계·구현 취약점 점검'은 테스트베드가 구축되어 있다면, 언제든지 수행할 수 있다.

이제, 취약점 점검을 실시하고, 점검 결과 발견한 취약점들을 해결할 수 있는 보완대책을 수립한 뒤, 정비기간을 이용하여 실 환경에 일괄 적용한다([그림 2]의 6, 7단계). 이 때 취약점에 대한 보완대책의 적용은 제어시스템의 정비기간에 수행해야 한다. 취약점 점검 방법 상세는 논문의 범주를 벗어나므로 생략한다. 발견된 취약점들 중, 환경설정과 관련한 일부 취약점들에 대한 보완대책은 즉시 적용할 수가 있다. 예를 들어, [표 5]의 'PLC 관리서비스' 분류에서 'Anonymous FTP 비활성화' 항목이 취약할 경우, Anonymous FTP가 비 활성화되도록 설정을 바꾸어준다. 즉, 이와 같은 대책은 취

약점 점검을 수행하는 정비 기간 동안에 보완이 가능하다. 그러나, 새로이 발견된 구현 및 설계상의 취약점들에 대한 보완대책은 즉각적인 적용이 어렵다. 예를 들어, 특정 PLC의 웹 서버 응용프로그램에서 버퍼오버플로우 취약점이 발견되었을 때, 개발자가 코드를 수정하여 디버깅 및 테스트 후 적용하기까지는 수일 또는 수개월의 시간이 소요될 수도 있다.

[표 7]은 본 장에서 설명한 취약점 점검 및 적용 시 고려해야 할 사항을 나타낸다.

〔표 7〕 취약점 점검 및 보완대책 적용 시 고려사항

고려사항 / 점검항목	점검 환경	점검 시기	보완대책 적용시기
환경설정	실 환경	정비 기간	정비 기간
설계·구현	테스트베드	모든 시기	

Ⅳ. 결 론

본 논문에서는 제어시스템 전반에서 사용되는 PLC에 대한 취약점 분석 방법에 대해 연구하였다. 구체적으로는 먼저 일반 PLC 특성을 살펴보았다. PLC 특성 분석을 기반으로 일반 PLC에서 발생할 수 있는 보안위협들과 이 위협들이 PLC에 끼칠 수 있는 영향에 대해서 분석하였다. 또한 PLC에 직·간접적인 피해를 입힐 수 있는 대상들로 PLC 취약점 점검 대상을 식별하였고, 대상 별 환경설정 및 설계·구현상의 취약점 점검 항목을 각각 정의하였다. 마지막으로 실 환경에서 취약점 점검 수행 및 보안 대책 적용 시 필요한 고려사항들에 대해 살펴보았다. 이는 PLC가 운용되고 있는 제어시스템 실 환경에서의 취약점 분석 및 보완대책 적용을 위한 자료로 활용될 수 있을 것으로 기대한다.

참 고 문 헌

[1] ICS-CERT, http://ics-cert.us-cert.gov/alerts/ICS-ALERT-12-020-02A,

[2] ICS-CERT, http://ics-cert.us-cert.gov/alerts/ICS-ALERT-12-019-01A

[3] ICS-CERT, http://ics-cert.us-cert.gov/advisories/ICSA-12-320-01

[4] Nicolas Falliere, Liam O Murchu, Eric Chien, "W32.Stuxnet Dossier v1.3", Symantec Security Response, November, 2010

[5] SHINE, http://www.icscybersecurityconference.com/session/project-shine-update/

[6] ICS-CERT, http://ics-cert.us-cert.gov

[7] BaseCamp Project, http://www.digitalbond.com/tools/basecamp/

[8] G. Morten, Creating a Weapon of Mass Disruption: Attacking Programmable Logic Controllers, Master Thesis, 2013

[9] Frank Swiderski and Window Snnyder. Threat modeling. O'Reilly Media, Inc., 2009

[10] 미래창조과학부 , 주요정보통신기반시설 취약점 분석.평가 기준, 미래창조과학부고시 제2013-37호 주요정보통신기반시설 취약점 분석.평가 기준

[11] Department of Homeland Security, Common Cybersecurity Vulnerabilities in Industrial Control Systems", May 2011

〈저 자 소 개〉

사 진

김 동 욱 (Dongwook Kim)
정회원

2005년 2월 : 경북대학교 컴퓨터공학과 학사
2012년 2월 : 포항공과대학교 컴퓨터공학과 박사
2012년 4월~현재 : 국가보안기술연구소 선임연구원

관심분야 : 통신공학, 제어시스템 보안, 정보보호

사 진

민 병 길 (Byunggil Min)
정회원

2002년 2월 : 충북대학교 컴퓨터공학과 학사
2004년 2월 : 포항공과대학교 컴퓨터공학과 석사
2004년 3월~현재 : 국가보안기술연구소 선임연구원

관심분야 : 제어시스템 보안, 침입탐지 시스템, 취약성 분석

박 현 동 (Hyun-Dong Park)
정회원

1995년 2월 : 충남대학교 컴퓨터과학과 학사
1997년 2월 : 충남대학교 컴퓨터과학과 석사
2000년 2월 : 충남대학교 컴퓨터과학과 박사
2000년~2002년 : 국가보안기술연구소 선임연구원
2002년~2004년 : 대덕대학 전임강사
2004년~2004년 11월 : 충남대학교 전기정보통신공학부 연구교수
2004년 12월~현재 : 국가보안기술연구소 책임연구원
관심분야 : 제어시스템 보안, 통신공학, 정보보호

박 상 우 (Sangwoo Park)
정회원

1989년 2월 : 고려대학교 수학교육과 졸업
1991년 8월 : 고려대학교 수학과 석사
2003년 2월 : 고려대학교 수학과 박사
1991년 8월~1999년 12월 : 한국전자통신연구원 선임연구원
2000년 1월~현재 : 국가보안기술연구소 책임연구원
관심분야 : 제어시스템 보안, 암호론, 정보보호