RSS를 활용한 SCADA 시스템 보안 향상에 관한 연구

정성모*, 김석수*, 송재구*, 김태훈*
*한남대학교 멀티미디어공학과
e-mail:SungmoJ@Gmail.com

A Study on SCADA system Security Improvement using RSS

Sungmo Jung*, Seoksoo Kim*, Jae-gu Song*, Taihoon Kim*
*Dept. of Multimedia Engineering, Hannam University

요 약

SCADA 시스템은 대개 생산 공정을 감시하고 제어하는데 사용되는 소프트웨어 패키지로써, 대부분 대규모 플랜트 상태를 감시하고 제어하기 위해 사용된다. 특히, 전력, 댐 철도, 원자력 등과 같은 주요 핵심기반시설에서 이를 활용한다. 기존 SCADA 시스템은 일반적으로 분리된 독자적 네트워크상에서 존재했기 때문에 보안에 소홀할 수밖에 없었다. 그러나 최근 기업정보시스템과의 연동 필요성으로 인해 아주 적게나마 원격에서 접속가능한 지점이 존재하고 이로 인한 취약성이 드러나고 있다. 이처럼 외부 공격에서의 취약성 분석을 통한 연구는 현재 진행 중에 있지만, 물리적인 접속을 통한 RTU Master와 Slave의 데이터를 직접적인 변조에 대한 연구는 이루어지지 않고 있다.

Modbus RS485통신을 사용하는 SCADA 시스템의 특성상 RTU Master와 Slave는 RJ11 케이블을 통해 1km까지도 연결될 수 있는 상황이므로, 이러한 케이블에 물리적인 접속을 통하여 데이터를 Sniffing하고 Spoofing하는 것이 가능하다.

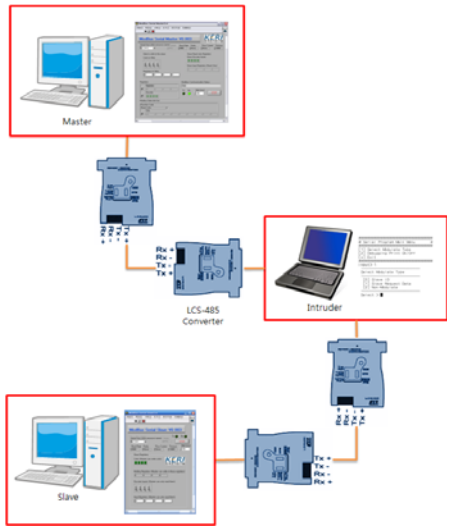
따라서 본 논문에서는 이러한 물리적인 접속을 통한 데이터 변조 공격에 대비하기 위하여 RSS를 활용한 보안 향상 방안에 대하여 연구하였고, 이러한 데이터 변조 공격을 검출해 낼 수 있는 모니터링 시스템에 대하여 제안하였다.

1. 서론

정보통신기술이 급격히 발전하면서 대규모 산업 시설들의 제어시스템도 이러한 기술발전의 흐름에 따라 점차 정보화되고 있는 추세이다. 이러한 대규모 산업 시설들에서 사용하는 제어 시스템은 민감한 물리적 기능을 제어하고 감시하기 위하여 대다수 산업 분야에서 사용하고 있는 컴퓨터 기반의 시스템이다. 이러한 제어 시스템은 필드로부터 운영된 데이터와 센서측정 결과를 모아 정보를 표시하며, 순차적 제어 명령을 수행한다. 이러한 제어시스템 기반의 대규모 플랜트 네트워크는 국가에서 중점관리하고 있으며 중요핵심기반시설로 지정되어 운영되고 있다. 이런 시설들의 공통점은 장치마다 상호 또는 외부 기기와 연결하여 각각의 장치에 대한 원격 접근과 제어가 가능하고, 명령 및 조작을 할 수 있도록 양방향 통신서비스 환경을 구축하고 있다. 이러한 환경을 제어시스템의 일종으로 일반적인 분산제어시스템을 SCADA(Supervisory Control and Data Acquisition)라고 한다[1].

SCADA 시스템은 대개 생산 공정을 감시하고 제어하는데 사용되는 소프트웨어 패키지로써, 대부분 대규모 플랜트 상태를 감시하고 제어하기 위해 사용된다. 특히, 전력, 댐 철도, 원자력 등과 같은 주요핵심기반시설에서 이를 활용한다[2]. 기존 SCADA 시스템은 일반적으로 분리된 독자적 네트워크상에서 존재했기 때문에 보안에 소홀할 수밖에 없었다. 그러나 최근 기업정보시스템과의 연동 필요성으로 인

해 아주 적게나마 원격에서 접속가능한 지점이 존재하고 이로 인한 취약성이 드러나고 있다. 이처럼 외부 공격에서의 취약성 분석을 통한 연구는 현재 진행 중에 있지만, 물리적인 접속을 통한 RTU Master와 Slave의 데이터를 직접적인 변조에 대한 연구는 이루어지지 않고 있다.



[그림 1] 물리적 연결을 통한 데이터 변조 구조[3]

Modbus RS485통신을 사용하는 SCADA 시스템의 특성상 RTU Master와 Slave는 RJ11 케이블을 통해 1km까지도 연결될 수 있는 상황이므로, 이러한 케이블에 물리적인 접속을 통하여 데이터를 Sniffing하고 Spoofing하는 것이 가능하다[3].

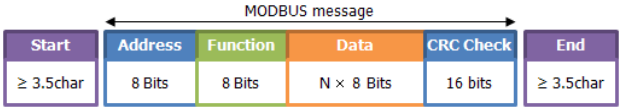
따라서 본 논문에서는 이러한 물리적인 접속을 통한 데이터 변조 공격에 대비하기 위하여 RSS를 활용한 보안 향상 방안에 대하여 연구하였고, 이러한 데이터 변조 공격을 검출해 낼 수 있는 모니터링 시스템에 대하여 제안하였다.

2. 관련연구

2.1. Modbus 프로토콜

Modbus는 전세계적으로 널리 보급되어 있는 자동화 프로토콜의 하나이며, 기존의 RS-232/422/ 485 디바이스를 지원한다. PLC, DCS, HMI, 계측기, 미터 등의 수많은 공업 기기는 Modbus를 통신표준으로 사용하고 있다.

MODBUS 통신의 종류는 MODBUS serial, Modbus plus and Modbus TCP/IP이다. 본 연구에서는 Modbus serial을 이용하여 진행하였으며, RS232(EIA/TIA-232), RS422, RS485(EIA/TIA-485) 세 종류가 있으며, 메시지 구조는 다음 그림과 같다[4].



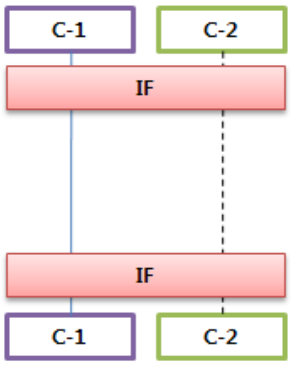
[그림 2] Modbus 메시지 구조

전송 모드는 네트워크에 따라 전송된 메시지의 비트 바이트의 내용을 정의하며, 메시지를 스트림으로 패키지화하는 방법 및 메시지 정보를 해독한다. 표준 Modbus 네트워크는 ASCII나 RTU 모드 전송방식 중에서 한 가지를 사용하며, 본 연구에서는 RTU 모드 기반의 모니터링 시스템을 제안하였다.

2.2. RSS

RSS(Route Standby System)은 경로 예비 방식으로써 통신 회선의 신뢰도를 높이기 위해 미리 현용 회선 이외의 예비 회선을 두고 그 회선을 항상 동작시켜서, 장애가 발생하면 그 중계기를 포함하는 현용 회선을 어느 단위 구간에서 경로마다 예비 회선으로 전환하는 방식을 말한다. 보통 회선을 몇 개의 중계소를 포함하는 구간으로 분할하고 이를 단위로 하여 전환한다. 따라서 전환 구간마다 여러 개의 현용 회선에 대해 하나의 예비 경로를 갖는다[5]. RSS는 예비 전력 및 예비 네트워크를 위하여 전력 공급 및 네트워크 연결성이 중요한 산업 전반에서 사용되고 있다.

다음 그림은 RSS의 기본 구조이다.



[그림 3] RSS 기본 구조

본 연구에서는 이러한 RSS 기본 구조를 활용하여 물리적인 접속을 통한 데이터 변조 공격에 대비하기 위한 보안 향상 방안에 대하여 연구하였고, 이러한

데이터 변조 공격을 검출해 낼 수 있는 모니터링 시스템에 대하여 제안하였다.

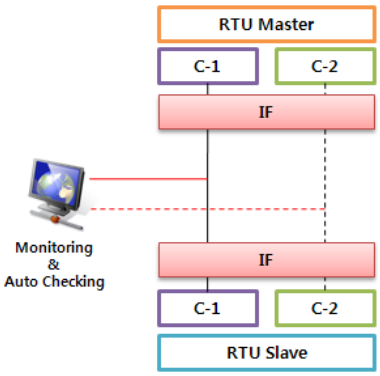
3. RSS를 활용한 SCADA 보안 향상 시스템

본 논문에서 제안하는 RSS를 활용한 모니터링 시스템을 위해 다음과 같은 가정을 하였다.

· 가정 : 공격자는 시스템의 예비 연결 경로를 알 수 없어야 한다.

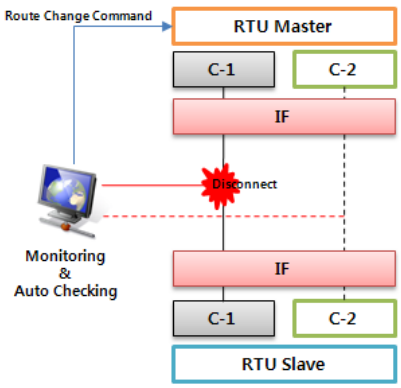
3.1. 제안하는 시스템 구조

본 논문에서 제안하는 모니터링 시스템의 구조는 RTU Master/Slave, C-1/C-2 데이터 전송 경로 그리고 모니터링 및 자동 체킹을 위한 시스템으로 구성되어 있으며, 구조는 다음과 같다.



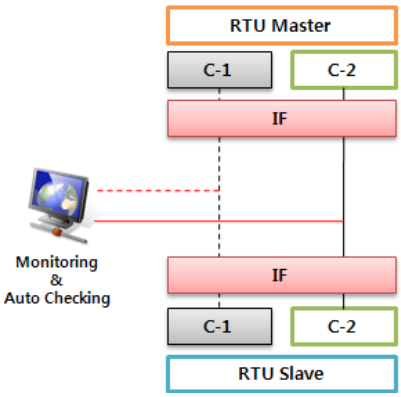
[그림 4] RSS를 활용한 모니터링 시스템

본 논문에서 제안하는 시스템은 기본적으로 RSS의 기능을 포함하고 있다. 하지만 기존 RSS 방식과는 달리, 경로에 모니터링 시스템을 포함하고 있는 것이 특징이다. 본 모니터링 시스템은 C-1이라는 현재 접속 중인 경로와 C-2라는 예비 경로를 모니터링 하고 있다. 만약 C-1 경로에서 데이터 변조 등의 공격이 감지되면 이를 자동으로 체크하여 C-1의 연결을 차단하라는 명령을 내리고, 예비 경로인 C-2를 활용하라는 명령을 내리며 그 구조는 다음 그림과 같다.



[그림 5] 공격 검출시 명령전달 구조

모니터링 시스템으로부터 명령을 전달받은 RTU Master는 C-1의 데이터 전송을 차단하고 C-2로 데이터를 전송하여 물리적인 공격에 대한 예비 경로 활용을 통하여 데이터 변조 공격에 대응하게 된다.



[그림 6] RSS가 적용된 상태 구조

3.2 제안하는 모니터링 방법

본 연구를 위하여 간단한 RTU Master/Slave를 구성하여 테스트를 한 결과 주로 송/수신 되는 데이터 형식은 다음과 같이 한정된 코드를 사용하는 것을 확인할 수 있다.

[표 1] RTU 송/수신 데이터

ID	FC	Data	CRC
01	02	00 00 00 00	79 3F
01	02	00 00 00 04	79 3F
01	02	00 00 00 00	79 C9
01	02	00 00 00 04	79 C9
01	04	00 00 00 00	79 3F
01	04	00 00 00 04	79 3F
01	04	00 00 00 00	F1 C9
01	04	00 00 00 04	F1 C9

본 논문에서 제안하는 시스템을 실제 적용하기 위해 미리 정의된 CRC 값을 모니터링 하는 방법은 다음과 같다.

```
    // Frame 수신
    switch(Recv_Buff[1]) {    // Fc
      case 0x01 : // Read Coil
      case 0x02 : // Read Input Status
      case 0x03 : // Read Holding Register
      case 0x04 : // Read Input Register
      case 0x05 : // Force Single Coil
      case 0x06 : // Preset Single Register
        RecvLen += ReadN(Ser_FD, &Recv_Buff[RecvLen], 6);
        if(RecvLen!=8) return -1;
        break;

      case 0x0F : // Force Multiple Coil

        RecvLen += ReadN(Ser_FD, &Recv_Buff[RecvLen], 5);
        if(RecvLen!=7) return -1;
        RecvLen += ReadN(Ser_FD, &Recv_Buff[RecvLen], (int)Recv_Buff[6]+2);
        if(RecvLen!=((int)Recv_Buff[6]+2+7)) return -1;
        break;

      case 0x10 :
        RecvLen += ReadN(Ser_FD, &Recv_Buff[RecvLen], 5);
        if(RecvLen!=7) return -1;
        RecvLen += ReadN(Ser_FD, &Recv_Buff[RecvLen], (int)Recv_Buff[6]+2);
        if(RecvLen!=((int)Recv_Buff[6]+2+7)) return -1;
        break;
      default :
        printf("[Server->Modulator] Invalid Function Code.. \n");
        break;
    }

// CRC 확인
CalcCRC = CRC16(Recv_Buff, RecvLen-2);
memcpy(&RecvCRC, &Recv_Buff[RecvLen-2], 2);
RecvCRC = User_htons(RecvCRC);
if(CalcCRC != RecvCRC) {
   printf("[Client->Modulator] CRC Error(RecvCRC:%d / CalcCRC:%d) \n", RecvCRC, CalcCRC);
     // Debugging Print
     printf("\n");
     printf("[Client->Modulator] : ");
     for(Loop=0; Loop<RecvLen; Loop++) printf("[0x%02x]", Recv_Buff[Loop]);
     printf("\n");
   return -2;
}
```

[그림 7] CRC 체크를 통한 데이터 변조 확인

RTU Master와 Slave가 데이터를 송/수신하는 내용은 한정적이며, 그에 따른 CRC 계산 결과 값 또한 한정적이다. 따라서 이러한 한정적인 CRC 계산 결과 값을 모니터링 시스템에 미리 정의하여 정의되지 않은 CRC 값이 검출될 경우, 이를 공격으로 간주하고 RTU Master에 예비 경로 사용 명령을 내리도록 한다.

4. 결론

최근 해킹 기술의 보편화로 인하여 공개된 프로그램을 이용하면 누구나 해킹 시도를 할 수 있고, 산업 시설 전반에 피해를 입힐 수 있는 계기가 되었다. 이로 인해 보안의 중요성이 높아지면서 정보통신 환경에서 중요핵심기반시설인 SCADA 시스템에 대한 위협 요소 분석 및 보안관리가 이슈화 되고 있다. 불특정 시스템에 대한 공격으로 인하여 SCADA 시스템에 피해가 발생하게 되면 피해대상은 해당 산업 시설에 그치는 것이 아니고, 일반인들에게도 그 피해가 올 수 있다.

본 논문에서는 SCADA 시스템의 보안 중요성을 고려하여 RSS를 활용한 보안 향상에 관한 연구를 하였다. 이를 통하여 물리적 공격에 대한 하나의 대안을 마련하였으며, 이를 통하여 보안 향상을 기대할 수 있다.

그러나 제안하는 RSS를 활용한 모니터링 시스템은 RTU Master와 Slave의 예비 연결 경로를 공격자가 알 수 없어야 한다는 가정이 필요하며, 공격자가 Sniffing을 통한 패킷 분석이 선행된다면 본 논문에서 제안하는 시스템은 효과를 기대하기 힘들 수밖에 없다.

향후 연구에서는 본 논문에서의 취약점을 바탕으로 데이터 전송 속도 측정을 통하여 보다 효율적인 모니터링 시스템을 구현할 것이다.

참고문헌

[1] National Intelligence Service, 2004 The White Paper of National Information Security, http://www.nis.go.kr, 2004.

[2] GAO, Critical Infrastructure Protection: Challenge and Efforts to Secure Control System, http://www.gao.gov, Mar. 2004.

[3] Modbus 기반 SCADA 해킹 테스트 시스템 설계, 송재구, 정성모, 김석수, 김태훈, 강동주, 김석주, 한국정보기술학회, 8권, 2009.

[4] Introduction to MODBUS, Technical Tutorial, Dec. 2002.

[5] http://terms.naver.com/item.nhn?dirId=213&docId=773, Naver, 2009.