# SCADA용 DNP3 프로토콜의 소규모 실험환경 구축*

이 동 수†, 김 광 조‡
카이스트 전산학과

# Building Small-Scale Testbed for DNP3 Protocol in SCADA system

Dongsoo Lee† , Kwangjo Kim‡
Dept. of Computer Science,
Korea Advanced Institute of Science and Technology

요 약

SCADA는 산업 제어 시스템에서 산업 장비를 원격으로 감시하고 제어하는데 쓰이는 시스템이다. 이 시스템은 Smart Grid를 적용하기에 앞서 보안을 강화하기 위해 새로운 방법을 고려할 필요가 있다. 그러나 소규모 연구실에서 SCADA 시스템에 연구용 테스트베드를 구축하는데 어려움이 있다. 이에, 이 논문에서는 전력망 SCADA 시스템에서 가장 널리 이용하고 있는 DNP3 프로토콜에 대해 공격과 방어를 시도할 수 있는 소규모 테스트베드의 구성방법을 제시하고, 실제 동작과 유사하게 동작하며, 공격을 한 뒤 보완할 수 있는 실험 환경을 구축하였다.

ABSTRACT

SCADA is a type of industrial control systems which monitor and control the industrial device. To adjust the smart-grid, a new method is needed which improves security to SCADA system. But a difficulty for small laboratory happens to research SCADA system in a large scale, because of building own testbed. In this paper, a testbed in a lab environment is suggested to attack and defence DNP3 protocol which has the largest market share in industrial control systems. We found that this testbed has low price, simulate like the real SCADA system over TCP/IP protocol and has ability to attack, defence and enhance the security.

**Keywords: SCADA Protocol, DNP3, Testbed**

## I. 서 론

SCADA(supervisory control and data acquisition)는 산업 제어 시스템에서 산업 장비를 감시하고 제어하는데 쓰이는 시스템이다. 최근 Smart Grid를 사용하기 위해 전력 생산 및 전송 체계를 개방해야할 필요성이 생기면서 기존에 사용되던 SCADA 시스템에서 보안 3요소 기밀성, 무결성, 가용성 중 기밀성과 무결성을 추가로 보장할 수 있도록 새로운 방법을 고려해야할 필요성이 있다.

이에 따라 보안 취약점을 찾고 보완할 수 있도록 SCADA 시스템을 직접 구현하거나 테스트베드를 구축하여야 한다.

그러나 대부분의 테스트베드는 국가 주도로 대규모로 구성되어 있고 테스트베드 소프트웨어로 사용되는 Power World와 같은 프로그램들이 매우 고가이기 때문에[1], 소규모로 연구하기에 어려움이 있는 것이 사실이다.

따라서 본 논문에서는 소규모 연구실에서 다룰 수 있는 규모이면서, 특히 전력 시스템에서 널리 쓰이는 SCADA 프로토콜인 DNP3를 사용하여 이 프로토콜을 공격하는 테스트베드를 만들고 이 테스트베드를 이용하여 실제 공격과 방어를 수행할 수 있음을 보이

* This research was supported by the KUSTAR-KAIST Institute, Korea, under the R&D program supervised by the KAIST.
† letrhee@kaist.ac.kr
‡ kkj@kaist.ac.kr

고자 한다.

## II. SCADA 시스템과 테스트베드

### 2.1 DNP3 프로토콜

DNP3 프로토콜은 Power-Grid 시스템을 구축하기 위해 만들어진 SCADA 프로토콜로 북미에서 전력망 제어를 위해 가장 많이 쓰이고 있는 프로토콜이다. 이 프로토콜은 기존에 많이 쓰이던 Modbus에 비해 높은 신뢰성을 가지도록 설계되었으며, 구성 방식으로 하나의 마스터스테이션(통제실)과 한 개 이상의 아웃스테이션(RTU, 장비단)이 기본 방식이며, 마스터스테이션과 아웃스테이션 사이에 서브마스터스테이션(중계기)이 놓이는 구조도 가능하다. DNP3는 TCP/IP 통신과 Serial 연결을 지원할 뿐만 아니라 Link Layer, Transport Layer, Application Layer의 3단계로 분리하여 복잡하면서 많은 데이터를 보낼 수 있는 것도 특징이다.[2] 그러나 가용성을 중시하는 산업 네트워크의 특성상 외부의 공격에 대한 대비가 되어있지 않다는 문제점이 있어, 최근에야 DNP Sec과 같은 보안 프로토콜 등이 제안되고 있다.

### 2.2 기존의 소규모 SCADA 테스트베드

대학 등에서 연구를 위하여 이미 소규모 SCADA 테스트베드 사례도 있다.

Queiroz 등[3]은 SCADA 네트워크 구성을 위해 OMNET++에서 INet Framework를 이용하였다. 이 네트워크는 수력 발전소 시스템을 시뮬레이션 하는데 RTU에는 수위를 확인하는 센서와 물을 퍼올리는 펌프를 사용하여 2가지 장치가 연결되어있으며, HMI Server와 HMI Client를 사용하는 단순한 구조이다. 주목할 점으로는 RTU Device로 Lego NXT를 이용하여 직접 물리 장치로 표현해 내었다는 점이 있다. 위 테스트베드를 테스트를 위해 TCP Syn Flood Attack을 이용하였으며, HMI Server와 아웃스테이션 사이에 동기화가 제대로 되지 않음을 보였다. 이 테스트베드에서 Lego NXT를 사용한 것은 시각화 측면에서 강력한 장점이 될 수 있으나 아웃스테이션을 다수 구성하기 위해서는 Lego NXT를 다수 구입해야하며, 이 장비들을 유기적으로 동작하도록 하는 모델을 구상하는 것이 매우 어렵다는 것을 볼 때 한계점이 존재한다고 볼 수 있다.

기존 논문으로써 Mallouhi 등[4]과 Giani 등[5]은 OMNET++을 이용하여 네트워크 시뮬레이션을 구축한 후 OMNET++용 Modbus 플러그인을 사용하여 SCADA System을 구현할 수 있으며, 다양한 공격 방식을 수행 가능함도 보였으나, 비교적 구조가 단순하여 테스트가 쉬운 Modbus 프로토콜에 머물러있어 실제 연구가 필요한 DNP3 프로토콜에 대해선 언급이 부족하다.

국내의 DNP3 관련 연구로는 김의형 등[6]이 시리얼 연결을 통해 단말A, 단말B, 공격자 총 3대의 컴퓨터를 이용하여 마스터스테이션-아웃스테이션 구분이 없는 간이 시스템을 구축하고 패킷 변조가 가능함을 보였다.

### 2.3 무료 DNP3 프로토콜 소프트웨어 자료들

Axongroup에서는 SCADA 프로토콜로 쓰이는 IEC 60870-5-104와 DNP3 프로토콜에 대한 시뮬레이터 Axon Test를 제공하고 있다.[7] 이 시뮬레이터는 DNP3 프로토콜을 지원하며 Data Object를 XML파일의 형식을 사용하여 지정 가능한 장점이 있으나, 아웃스테이션에 해당하는 기능을 지원하지 않기 때문에 테스트를 위해선 SCADA 장비를 따로 구하거나 아웃스테이션 기능을 제공하는 다른 소프트웨어를 구해야하는 단점이 있다.

Tgscada사에서는 Modbus, Modbus+, L&G, DNP3를 지원하는 시뮬레이터 xMasterSlave를 제공하고 있다.[8] 이 시뮬레이터는 마스터스테이션와 아웃스테이션 기능을 지원할 뿐만 아니라 실제 SCADA 시스템에 사용할 수 있도록 HMI기능 또한 제공하지만, 아웃스테이션에서 PLC 장비를 연결하는 것이 전제되어 있기 때문에 연구용으로는 적절하지 않으며, 다양한 프로토콜을 지원하기위해 DNP3 전용 기능을 제공하지 않는 문제가 있다.

Automatak에서 오픈소스로 공개하고 있는 DNP3 라이브러리도 있다.[9] 이 라이브러리는 파일 전송을 제외한 기본 기능을 모두 지원하고 필요에 따라 마스터스테이션, 서브마스터스테이션, 아웃스테이션에 제한 없이 구성가능 하다. ECC DNP3나 DNP Sec과 같은 최신 전송 모드를 지원하고 있지는 않으나, 소스가 공개되어 있어 얼마든지 보안 요소를 직접 추가할 수 있는 것이 큰 장점이다.

## III. 테스트베드 제안

### 3.1 요구 사항

이 테스트베드를 구성하는데 있어서 필요한 사항들을 기술한다.

#### 3.1.1 공격 환경

Smart Grid를 고려한 테스트베드이므로 TCP/IP 네트워크상에서 DNP3 프로토콜을 이용하여 모델을 구축한 뒤, DNP3 프로토콜을 공격 및 방어할 수 있는 환경이 준비되어야 한다.

#### 3.1.2 동작 모델

테스트베드에서 공격자가 공격 시나리오를 세울 수 있도록 마스터스테이션과 아웃스테이션이 데이터를 주기적으로 주고받으며, 장비의 값이 시간의 흐름에 따라 유기적으로 바뀌어야한다.

#### 3.1.3 구현 용이성

소규모로 구성되는 테스트베드이므로 가능한 낮은 비용으로 구축 가능해야 한다. 해결방안으로써 테스트베드의 소프트웨어로 Open Source 또는 Freeware의 형태로 제공되는 것을 사용하며, 하드웨어 장비의 경우 다른 목적의 연구로도 사용할 수 있도록 노트북, 네트워크 스위치와 같은 범용 장비를 이용하여 구성할 필요가 있다.

#### 3.1.4 확장 가능성

DNP3 프로토콜에 대해 유효한 공격을 찾았을 때 방어 시스템 제안을 후속 연구로 진행할 수 있도록 테스트베드를 구성해야 한다. 이를 위해 스크립트 기반으로 동작하여 사용자 정의 명령을 수행할 수 있는 소프트웨어를 사용하거나, Open Source 프로그램을 사용하는 방법이 효과적이다.

### 3.2 테스트베드 모델

위의 요구 사항에 따라 테스트베드에서 수행할 모델을 고려해 보았다. 스마트그리드 시스템이 동작되는 각종 발전소를 모델로 삼는 것이 적절하다고 보았다. 이중에서 원자력 발전소를 모델로 삼는 것이 가장 실제 상황과 비슷한 것은 사실이나 테스트베드 모델로 사용하기에 추상화가 어려우므로 이해하기 쉬운 수력 발전소(댐)를 우리의 테스트 모델로 정하였다.

이 발전소는 저수지의 수위를 관측하며, 갑문이 열린 정도, 발전소 갑문이 열린 정도, 갑문으로 흐르는 물의 속도, 발전소 갑문으로 흐르는 물의 속도, 발전량을 측정할 수 있고, 관리자는 갑문, 발전소 갑문의 열린 정도를 변경할 수 있다.

마스터스테이션은 위에 언급한 측정값들을 표시하고, 수위에 따라 갑문의 개폐정도를 조절할 수 있는 UI를 제공한다.

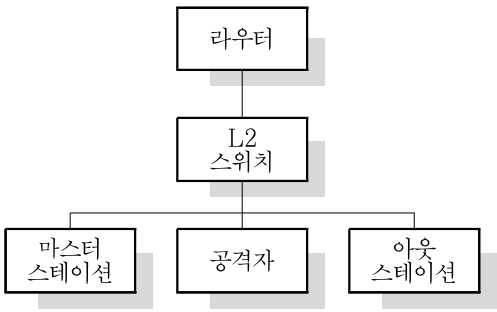
아웃스테이션은 센서를 측정하고 마스터스테이션의 지시를 받아 갑문의 개폐정도를 결정한다. 센서 값을 측정할 장비가 실제로 존재하지 않기 때문에 임의로 정해진 수식에 따라 움직이게 된다. 댐에 유입되는 물은 아무 일도 없을 때 유입되는 최소 물의 양과, 계절에 따라 평균 강우량이 바뀐다고 가정하고 사인파에 가깝게 변하는 유입량, 가끔씩 랜덤하게 폭우나 태풍이 발생하여 순식간에 수위가 올라가는 유입이 있을 수 있다고 가정한다.

따라서 마스터스테이션과 아웃스테이션에서 확인할 수 있는 값과 사용자가 직접 변경할 수 있는 값은 〔표 1〕과 같고, ○는 상시 가능, △는 초기 구동 시에만 가능, ×는 불가능함을 의미한다.

〔표 1〕 마스터스테이션, 아웃스테이션의 접근 가능 값

| | 마스터스테이션 | | 아웃스테이션 | |
|---|---|---|---|---|
| | 읽기 가능 | 변경 가능 | 읽기 가능 | 변경 가능 |
| 가상의 날짜 | ○ | × | ○ | ○ |
| 현재 수위 | ○ | × | ○ | △ |
| 강수량 | ○ | × | ○ | △ |
| 갑문 개방 정도 | ○ | ○ | ○ | × |
| 환경 값 | × | × | ○ | ○ |

시스템 요소는 일반적인 TCP/IP 기반 SCADA 네트워크와 같이 외부에서는 방화벽 등으로 인해 외부와 통신이 단절되어 있지만 이미 공격자의 프로그램이 네트워크 내로 침투하는데 성공하였다고 가정하였다.




〔그림 1〕 테스트베드의 연결 구성

## 3.3 테스트 베드 구성

제안한 테스트베드의 연결 형태는 〔그림 1〕과 같으며, 각 요소의 상세 구성에 대해 서술한다.

### 3.3.1 마스터스테이션

SCADA 네트워크에서 마스터스테이션에 해당하는 기기는 아웃스테이션의 값을 수집하며, 그 값에 따라 관리자가 산업 시스템을 제어하도록 되어있다. 시스템의 시각화를 위해선 SCADA에서는 HMI를 이용하여 표현하지만, HMI를 이용한 방법은 소규모 테스트베드에 사용하기에는 필요한 설정이 많아 사용하지 않았다. 대신 수위와 각 수문의 상태를 효과적으로 나타내도록 Windows Application에서 Progress Bar, Slider 등을 사용하였다. DNP3 통신은 앞서 언급한 바와 같이 DNP3 Library를 이용하며, 주기적으로 아웃스테이션에서 갱신된 값을 가져오도록 하였다.

### 3.3.2 아웃스테이션

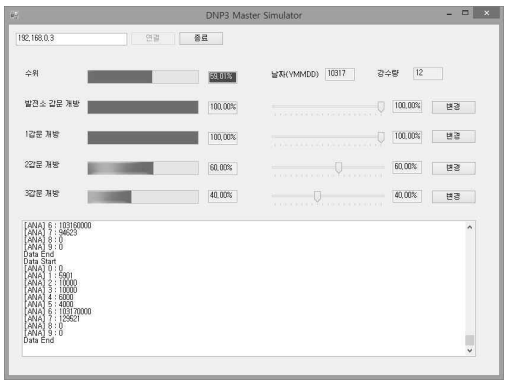
SCADA 네트워크의 아웃스테이션에는 장치가 연결되어있어야 하나, 소규모 테스트베드에서 그것을 모두 구성하는 데에는 어려움이 따른다. 이 테스트베드에서는 외부 장치를 연결하여 사용하기보다는, 각 값들의 상관관계를 식으로 나타내어, 한 값이 바뀔 경우 그 값에 따라 시간차를 두고 다른 값들도 변해가는 구조를 사용하여, 좀 더 사실적인 표현이 가능하도록 하였다.

### 3.3.3 공격자

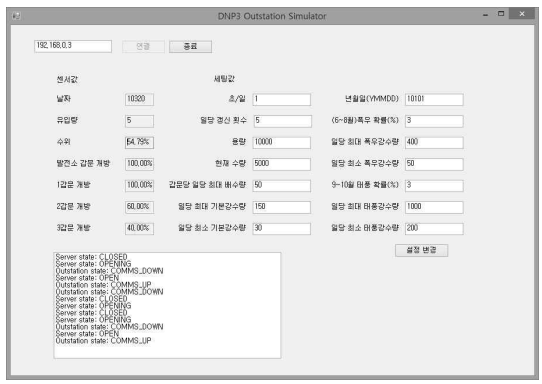
공격자는 네트워크상에서 Man-in-the-Middle 공격을 수행하여 SCADA Network를 공격한다고 가정하였다. Attack Taxonomy〔10〕에 따른 공격을 수행하려면 마스터스테이션과 아웃스테이션 사이의 패킷을 훔칠 수 있어야 하므로 ettercap을 이용하여 ARP spoofing 및 TCP packet sniff을 수행한다.

### 3.3.4 라우터 및 L2스위치

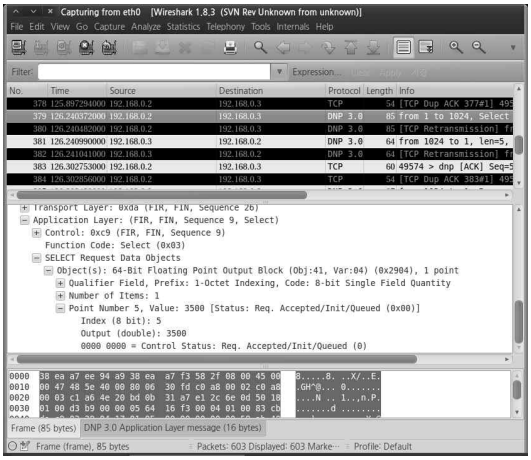
라우터와 L2스위치는 테스트베드에 TCP/IP 네트워크를 제공한다. 실험에 필요한 파일을 전송할 때에만 외부로 연결되며 실제 실험 시에는 외부 연결을 차단해 폐쇄망으로 운영된다.



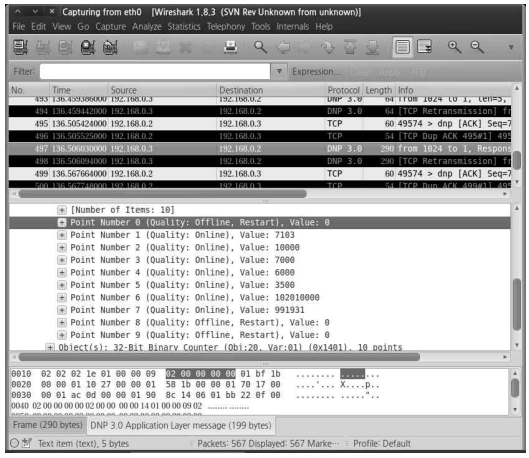

〔그림 2〕 마스터스테이션 애플리케이션




〔그림 3〕 아웃스테이션 애플리케이션




〔그림 4〕 Wireshark로 확인한 마스터스테이션 송신코드




〔그림 5〕 Wireshark로 확인한 아웃스테이션 응답 코드

## IV. 테스트베드 검증 및 결과

### 4.1 동작 모델 점검

테스트베드가 정상 동작하는지 확인한다. 마스터스테이션과 아웃스테이션이 정상적으로 연결되는지 확인하고, 아웃스테이션에서 정해진 환경 변수로 적절한 센서 값 변화를 일으키는지 확인하였다. 그리고 이 값이 마스터스테이션으로 전달되는지 확인하고, 마스터스테이션 측에서 갑문을 여는 정도에 따라 수위가 적절히 변하는가를 확인하였다.

〔그림 2〕은 마스터스테이션의 화면이며, 좌측은 아웃스테이션에서 받아온 센서 값, 우측은 갑문의 상태를 변경하는 부분이다. 〔그림 3〕은 슬레이트 파트의 화면이며, 좌측은 센서 값을 나타내는 곳이며 우측은 강수량(유입량)을 결정하는 환경 변수를 설정하는 곳이다. 시간이 지남에 따라 마스터스테이션에서 갑문의 상태를 변경해야할 정도로 수위가 변하는 것을 확인하였으므로 정상적으로 값이 변하는 것을 알 수 있다.

### 4.2 DNP3 공격 시도

수동 공격 방법인 Packet Sniffing을 이용하여 위 테스트베드의 공격 가능성을 검증한다. ettercap을 이용한 ARP Spoofing을 통해 공격자가 모든 Packet을 알아볼 수 있는지를 확인하고, Wireshark를 통해 현재 시스템의 값들이 어떤 값인지 확인하고, 마스터스테이션에서 어떤 명령을 내렸는지 추적해 보았다. 〔그림 4〕와 〔그림 5〕는 각각 마스터스테이션에서 아웃스테이션으로 수문의 상태 값을 변경하는 명령과, 아웃스테이션이 마스터스테이션으로 센서 값들을 전달해주는 패킷을 잡아낸 것이다.

## V. 결론

우리는 DNP3 프로토콜에 대해 공격을 시도할 수 있는 소규모 테스트베드 환경을 구축하였다. 이 테스트베드는 가능한 저렴하게 구성이 가능하며, 실제로 동작하는 모델을 가지고 있으며, 공격자가 실제로 공격을 할 수 있고, 필요에 따라 프로토콜을 보완할 수 있어야 함이 보장되어야 한다. 이를 위해 간단한 수력 발전소를 본 따 날짜에 따라 강수량이 변하고, 수위를 균등하게 유지하기 위해 갑문을 열고 닫도록 하는 모델을 제시하고 구현하였다. 또한 공격자가 Packet Sniffing을 통해 테스트베드에서 주고받는 값을 훔쳐낼 수 있다는 사실을 보여, 이 테스트베드를 이용하여 DNP3 프로토콜 공격을 확인해 볼 수 있다는 결론을 얻었다.

이번 연구에서는 테스트베드를 공격시도가 Packet Sniffing에 머무르고 있으나, 이를 보완하여 능동 공격을 통해 DNP3 패킷을 위조 및 변조하여 시스템에 영향을 끼칠 수 있음을 보이고, 이러한 공격에 대하여 기존에 연구된 다른 DNP3 보안 프로토콜이나 새로 제안한 프로토콜을 이용하여 각종 공격에 대한 방어가 가능한 지 실증하는 추후 연구가 필요하다.

## 참 고 문 헌

[1] 강동주, 김휘강, "스마트그리드에서의 CPS (cyber-physical system) 시뮬레이션 구현을 위한 제반 연구이슈 및 방법론 검토," 한국정보보호학회지, 22(2), pp. 15-22, 2012년 8월.
[2] DNP User Group, "Overview of the DNP3 Protocol," http://www.dnp.org/pages/aboutdefault.aspx
[3] Carlos Queiroz, Abdun Mahmood, Jiankun Hu, Zahir Tari, and Xinghuo Yu, "Building a SCADA security testbed," Network and System Security, 2009. NSS'09. Third International Conference on. IEEE, 2009.
[4] Malaz Mallouhi, Youssif Al-Nashif, Don Cox, Tejaswini Chadaga, and Salim Hariri, "A testbed for analyzing security of SCADA control systems (TASSCS)." Innovative Smart Grid Technologies (ISGT), 2011 IEEE PES, IEEE, 2011.
[5] Annarita Giani, Gabor Karsai, Tanya Roosta, Aakash Shah, and Bruno Sinopoli, "A testbed for secure and robust SCADA systems," SIGBED Review, Vol 5.2, 2008.
[6] 김의형, 김형식, 임광혁, 임을규, "전력 SCADA 시스템의 RS-232C 시리얼 통신구간에서의 Man-in-the-Middle Attack 가능성 연구," 보안공학연구논문지, 7(4), pp. 295-309. 2010년.
[7] Axon Group, "Axon Test", http://www.axongroup.com.co/pages/atsim
[8] Tgscada, "xMasterSlave." http://xmasterslave.tgscada.com/
[9] Automatak, "DNP3 Open Source Software." http://dnp3.github.io/
[10] Samuel East, Jonathan Butts, Mauricio Papa. and Sujeet Shenoi, "Critical Infrastructure Protection III. Chapter5. A Taxonomy of Attacks on the DNP3 Protocol," Springer, pp.67-81, Mar, 2009.