Article
KEPCO Journal on Electric Power and Energy
Vol. 4, No. 2, December 2018
ISSN 2465-8111(Print), 2466-0124(Online)
DOI : 10.18770/KEPCO.2018.04.02.075

A Security Nonce Generation Algorithm Scheme Research for Improving Data Reliability and Anomaly Pattern Detection of Smart City Platform Data Management

스마트시티 플랫폼 데이터 운영의 이상패턴 탐지 및 데이터 신뢰성 향상을 위한 보안 난수 생성 알고리즘 방안 연구

Jaekwan Lee[1†], Jinho Shin[1], Yongjae Joo[1], Jaekoo Noh[1], Jae Do Kim[1], Yongjoon Kim[1], Namjoon Jung[1]
이재관[1†], 신진호[1], 주용재[1], 노재구[1], 김재도[1], 김영준[1], 정남준[1]

Abstract

The smart city is developing an energy system efficiently through a common management of the city resource for the growth and a low carbon social. However, the smart city doesn't counter a verification effectively about a anomaly pattern detection when existing security technology (authentication, integrity, confidentiality) is used by fixed security key and key deodorization according to generated big data. This paper is proposed the "security nonce generation based on security nonce generation" for anomaly pattern detection of the adversary and a safety of the key is high through the key generation of the KDC (Key Distribution Center; KDC) for improvement. The proposed scheme distributes the generated security nonce and authentication keys to each facilities system by the KDC. This proposed scheme can be enhanced to the security by doing the external pattern detection and changed new security key through distributed security nonce with keys. Therefore, this paper can do improving the security and a responsibility of the smart city platform management data through the anomaly pattern detection and the safety of the keys.

마이크로 그리드 환경에는 변압기, 스위치, 에너지저장장치 등 많은 종류의 전력 설비가 존재하지만, IoT 기술의 발달에 따라 온도, 압력, 습도와 같은 센서 정보를 취득할 수 있는 기회를 제공하고 있다. 기존의 마이크로 그리드 환경에서는 IEC 61850 표준에서 정의하고 있는 MMS 등의 통신 프로토콜을 준용하여 전력 설비와 플랫폼 간 통합 운용되고 있다. 그렇기 때문에 IoT 데이터를 수용하기 위해서는 IEC61850 기반으로 구성된 데이터 수집 장치(FEP)에 IoT 데이터를 연계해 줄 수 있는 게이트웨이 기술이 필요하다. 본 논문에서는 마이크로그리드 운영 시스템 연계를 위한 IEC61850기반 IoT 게이트웨이 플랫폼 프로토타입을 제안하고자 한다. 게이트웨이 플랫폼은 IoT 프로토콜(MQTT, CoAP, AMQP) 인터페이스 모듈과 데이터베이스, IEC61850서버로 구성되어 있다. 데이터베이스의 경우, JSON 데이터를 저장하기 위해 오픈소스 기반의 NoSQL 데이터베이스인 Hbase와 MongoDB를 이용하였다. IoT 프로토콜을 검증하기 위해 라즈베리파이·아두이노·인텔 에디슨 SoC 기반 전력 IoT 디바이스 시뮬레이터를 이용하였고, IEC61850은 Sisco's MMS EASY Lite를 이용하여 IoT 프로토콜과 IEC 61850 프로토콜간의 상호호환성을 검증하였다.

Keywords: Smart City, Security Platform Module, Key Distribution Center, Random Generation Module

Manuscript received August 4, 2017, Accepted December 3, 2018
[1] KEPCO Research Institute, Korea Electric Power Corporation, 105 Munji-ro Yuseong-gu, Daejeon 34056, Korea
† jaekwan.lee@kepco.co.kr

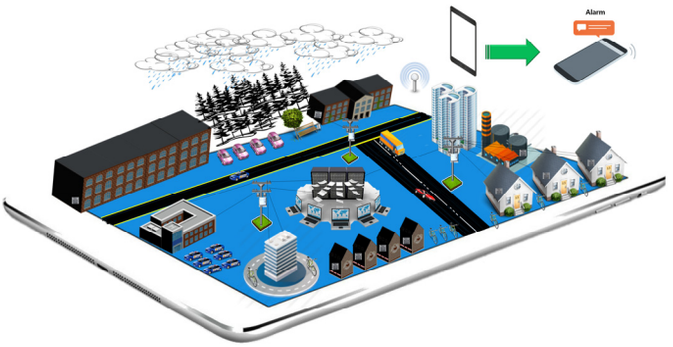

Fig 1. 스마트시티 구성도.

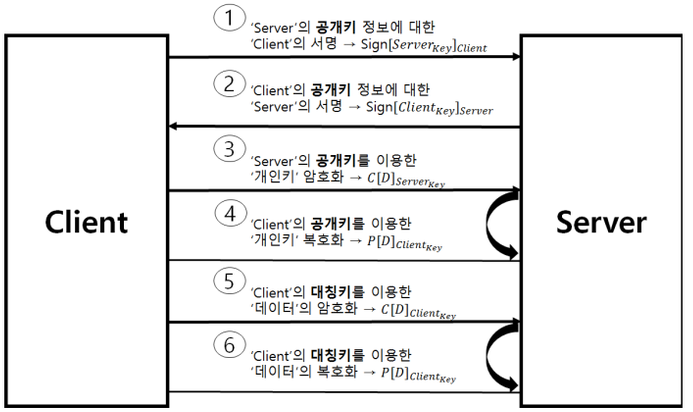

Fig 2. P2P 보안기술.

I. INTRODUCTION

스마트시티 플랫폼은 전력 데이터뿐만 아니라 수도, 가스, 난방 비전력 에너지 데이터, 기상, 환경 및 교통 등 도시 운영에 필요한 다양한 정보들을 송·수신하는 기반 인프라로서 높은 수준의 정보보안 정책과 보안위협에 대한 방어 대책이 요구된다.

기존의 보안기술을 스마트시티 플랫폼 데이터 운영에 그대로 적용할 경우 인증, 기밀성, 무결성 및 부인방지 등 관련 보안기술에서 악의적으로 행동을 취하는 공격자가 사이버 공격(스니핑, DoS, 악성코드, 중간자 공격, 데이터 위/변조 등)을 시도할 경우 보안 위협에 취약 할 수 있다 [1]-[3]. 현재의 보안기술(인증, 암호화, 무결성, 부인 방지 등)에서 사용되는 키는 고정된 키(공개키, 대칭키 등)를 사용하는데 이 방식은 제3자로부터 키가 탈취될 경우 위/변조 데이터 및 DoS 공격 등의 보안위협을 받을 수 있다. 이와같은 활용 방식은 방대한 데이터가 발생되는 스마트시티 플랫폼 데이터 운영에서는 현재의 보안기술만 사용하면 데이터 신뢰성과 운영관리 측면에서 비효율적일 수 있다. 따라서 본 논문에서는 스마트시티 플랫폼 데이터 운영 환경에서 키의 안전성 향상을 위해 키생성 및 분배역할을 하는 KDC (Key Distribution Center)모듈과 외부자 침입 탐지를 위한 보안 난수값 생성 기반 이상패턴 탐지 기법을 제안하였다. 제안된 이 방식은 앞서 설명한 기존 보안기술의 취약점인 사용되는 키가 탈취될 경우 이상행동 탐지가 불가하기 때문에 키의 안전성 향상과 외부자 이상행동 탐지 불가를 해결하기 위한 수단으로 미래의 스마트시티 플랫폼 운영 데이터 운영에 대한 보안성 강화를 위해 향상된

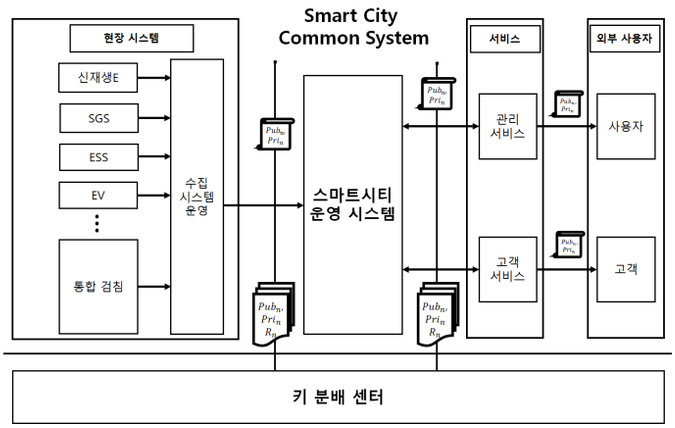

Fig 3. 스마트시티 통합 운영 보안 시스템.

보안기술을 제공할 수 있다. 본 논문에서 제안하는 방식은 다음과 같이 네 가지 절차로 운영된다. 첫 번째는 KDC를 통한 각 EndPoint별 공개키, 대칭키, 세션키 생성이며, 두 번째는 KDC로 인한 랜덤값 생성 및 할당이다. 세 번째로는 운영 과정에서의 대칭키 전달이며, 마지막으로는 정해진 규칙에 따른 이상행동 탐지과정이다. 본 논문은 2장 배경, 3장 보안 난수 알고리즘과 관련한 제안기법에 대해 서술하며, 마지막 4장에서는 본 논문에서 제안된 알고리즘에 대한 결론 및 향후 계획에 대한 내용으로 구성하였다.

II. BACKGROUND

이번 장에서는 기존에 통신 방식에서 사용되는 보안기술 종류에 대해 서술하였다.

A. 기존 보안 보안기술

Fig. 2는 P2P 보안 적용 시 이용되는 보안기술들에 대한 프로세스 절차를 보여준다. ①과②에서는 서로간의 신원확인을 위해 공개키 기반구조의 서명값을 서로 교환하여 검증을 통해 서로의 신원확인 후 세션 연결을 한다. ③과 ④는 서명을 통해 서로 신원이 확인됐을 때 클라이언트에서는 데이터 교환 시 기밀성 유지를 위한 개인키를 Server의 공개키로 암호화하여 전달한다. 이후 전달된 대칭키로 데이터를 암호화하여 서로간의 신뢰된 디바이스끼리 목적에 맞는 데이터를 안전하게 전달할 수 있다. 그러나 앞서 설명한 기존의 보안기술 단점을 개선하기 위해 보안 난수값 생성 기반 이상패턴 탐지 기법을 제안한다.

III. PROPOSED SCHEME

Fig. 3의 데이터 흐름은 다음과 같다. 현장 시스템에 해당되는 신재생E, SGS (Smart Grid System), ESS (Energy Storage System), EV (Electric Vehicle)와 통합검침(AMI, 수도, 가스, 난방 등)에서 발생되는 데이터들은 수집 시스템 운영을 통해 스마트시티 운영 시스템으로 해당 데이터들을 스마트시티 플랫폼에서의 정해진 정책으로 인해 일괄적 또는 실시간으로 전송한다. “스마트시티 운영 시스템”은 전송받은 데이터를 “서

비스"영역에 전송한다. 전국 각 지역에서 수집한 데이터는 관리 및 고객 보안기술에 제공되거나 외부 사용자에 제공된다. 본 논문의 특징은 4개의 동작 시스템(수집 시스템 운영, 스마트시티 운영 시스템, 서비스, 외부 사용자)이 있으며, 4개의 시스템은 기기인증, 무결성 검증, 기밀성, 이상 탐지 등에 대한 검증 및 운영을 진행한다. 진행 단계는 다음과 같다. 키 분배 센터에서는 각 시스템별 공개키, 대칭키와 랜덤값을 생성하고 각 시스템에 전송한다. 제안된 보안 모듈 장치는 기존의 보안 기술과는 달리 크게 2가지의 기능을 가진다. 첫 번째는 정책에 의한 주기적인 키 분배 방식이며, 두 번째는 랜덤값 쉬프트 연산을 통한 이상행위 탐지 방식이다. 쉬프트 연산 횟수에 대한 초기 랜덤값과 쉬프트 방향은 설립 당시에 생성하며, 이러한 프로세스는 운영 도중에 외부 공격자의 이상행위 탐지를 위한 하나의 수단이 될 수 있다. 이상 행동 탐지는 처음에 이상행위 감지를 탐지한 "스마트시티 운영 시스템"에서 초기에 설정된 규칙을 통해 수상한 행위를 다음과 같은 체크를 통해 감지한다. 공격자 이상행동 의심 설비 디바이스들은 1) 데이터 수집 시간 체크, 2) 세션키 사이즈 요청, 3) 응답 횟수 체크, 4) 쉬프트 연산값 체크를 통해 출력된 데이터를 암호화하여 운영 시스템에 보낸다. 이후에 운영 시스템은 이 데이터에 대한 비교를 하여 이상행위 탐지에 대한 검증을 수행한다. 기존의 보안기술은 이와같이 키 재분배와 설비 또는 키 탈취 후 공격자 이상행동에 대한 탐지 기능이 없기 때문에 보안성이 떨어지며, DoS (Denial of Servece)와 위/변조 데이터를 통한 불필요한 자원 낭비와 전체 시스템에 대한 운영 효율을 떨어트린다. 따라서 제안된 보안 모듈 장치는 DoS, 위/변조 공격, 비효율적인 자원 낭비와 운영 효율을 떨어트리게 되는 행동을 사전에 방지 및 대응하여 보안성과 시스템 운영 효율성을 향상시킬 수 있다. 제안된 방식은 "Step-by-Step 인증"이며, 본 논문에 대한 상세한 기술 내용은 Fig. 4부터 Fig. 12까지 관련 알고리즘을 제시하고 그에 따른 내용을 서술하였다.

A. Step by Step 인증

Fig. 4는 본 논문에서 제안하는 개선방식 "First-Step 인증"이다. 이 인증은 "수집 시스템 운영"과 "스마트시티 운영 시스템" 사이에서 전달되어지는 데이터를 보낼 시 인증 방법을 나타내며, 기존의 평문 전송 방식의 단점을 해결하고 이상행위 탐지를 위한 신규 방식을 제안하였다. 제안된 방식의 인증 프로세스는 다음과 같다. "수집 시스템 운영"에서 인증에 필요한 공개키, 대칭키 쌍(Pub_m, Pri_m)을 생성한 다음에 "스마트시티 운영 시스템"으로부터 Pub_{dc}를 전달 받는다. 이후에 검침된 계량 데이터 D와 세션키(S_k)를 생성하고 생성된 D를 S_k를 통해 MAC값을 산출한다. 그리고 그 값을 변수 HM에 저장한다. MAC값을 저장한 후 세션키를 안전하게 전달하기 위해 "수집 시스템 운영"의 Pri_m과 "스마트시티 운영 시스템"의 Pub_{dc}를 이용하여 S_k를 암호화하고 PS 변수에 저장한다. 마지막 단계로 암호코드를 생성하기 위해 Pub_{dc}+S_k 키로 D를 암호화하여 C_1 변수에 저장한다. 이렇게 생성된 Pub_m, HM, PS와 C_1을 메시지 M에 첨부하여 "스마트시티 운영 시스템"에 전송한다. 메시지 M에서 Pub_m을 첨부하여 보내는 이유는 복호화 용도와 사전에 키분배 센터로부터 랜덤한 비트를 쪼개서 값이 달라졌기 때문에 같이 첨부하여 보내는 것이다. 다음 "스마트시티 운영 시스템"의 인증 모듈에서는 메시지 M을 "스마트시티 운영 시스템"으로부터 받기전에 키쌍(Pub_{dc}, Pri_{dc})을 생성하여 Pub_{dc}를 "수집 시스템 운영"에 전달한다. 메시지 M을 전송받은 "스마트시티 운영 시스템"은 "수집 시스템 운영"의 공개키인 Pub_m을 메모리에 저장하고 암호화된 세션키 PS를 복호화 이후에 DS_1 변수에 저장하고 암호화된 데이터 C_1을 Pub_{dc}+DS_2를 통해 복호화하여 D_1에 저장한다. 마지막으로 "수집 시스템 운영"으로부터 전달된 데이터가 네트워크상에서의 위/변조 여부를 판단하기 위해 MAC(D_1)을 생성하여 메시지 M에 첨부된 HM과 비교한다. 만약 다를 경우, "스마트시티 운영 시스템"에 통보를 하고 같을 경우에는 검증에 성공한 것이다.

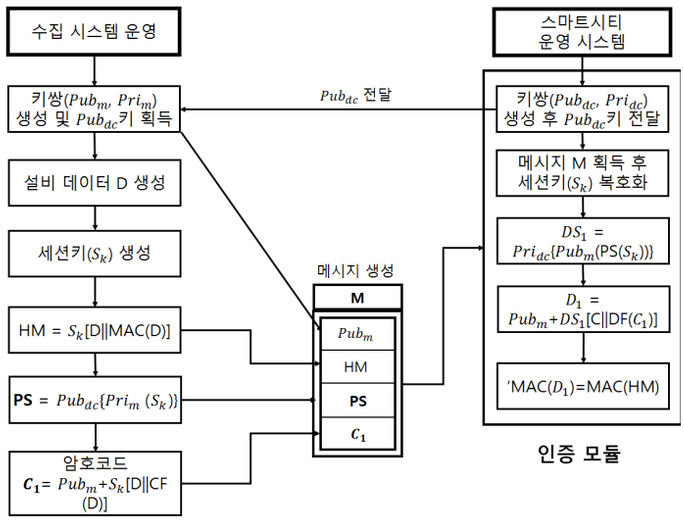

Fig 4. First-Step 인증.

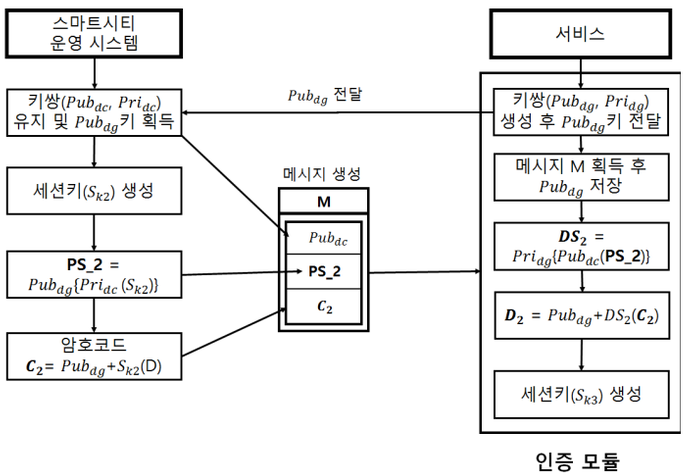

Fig 5. Second-Step 인증.

Fig. 5는 본 논문에서 제안하는 개선방식 "Second-Step 인증" 이다. 이 인증 방식은 "스마트시티 운영 시스템"과 "서비스"에서의 전달되는 데이터를 보낼 때 인증 방식을 나타낸 것이다. 기존의 데이터 전달 방식을 다음과 같이 제안하였다. "스마트시티 운영 시스템"에서 인증에 필요한 공개키와 대칭키 쌍(Pub_{dc}, Pri_{dc})은 기존 키쌍을 유지하고 "서비스"의 Pub_{dc}를 전달 받는다. 그리고 세션키 S_{k2}를 생성하고 이 키를 Pri_{dc}와 Pub_{dc} 순으로 암호화하여 그 값을 PS_2 변수에 저장한다. 마지막으로 암호코드 생성을 위해 계량데이터 정보 D를 Pub_{dc}+S_{k2}를 통해 암호화한다. 이렇게 만들어진 결과값{Pub_{dc} ,

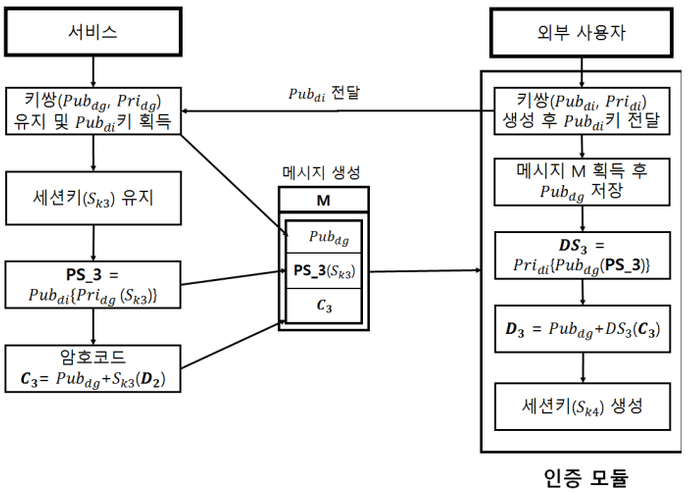

Fig 6. Third-Step 인증.

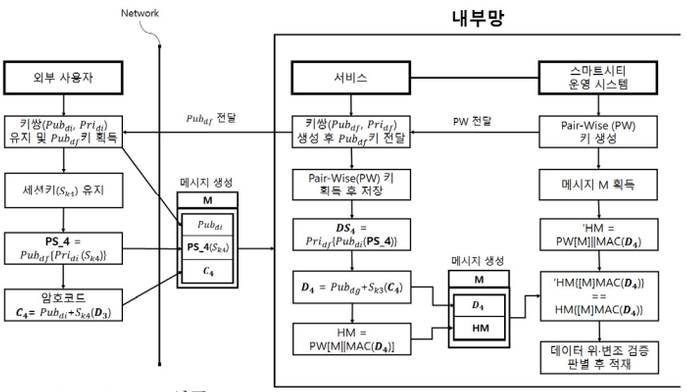

Fig 7. Fourth-Step 인증.

PS_2(S_{k2}), C_2}들을 메시지 M에 첨부하여 "서비스"에 전송한다. "서비스"는 키쌍(Pub_{dg}, Pri_{dg})을 생성하고 Pub_{dg}를 "스마트시티 운영 시스템"에 전달한다("스마트시티 운영 시스템" 세션 연결 후 자동으로 전달). "서비스"는 메시지 M을 획득한 이후에 Pub_{dc}를 메모리에 저장하고 세션키 S_{k2}를 얻기 위해 PS_2를 Pri_{dg}와 Pub_{dc}를 통해 복호화하여 DS_2에 저장한다. 그리고 암호화된 데이터(C_2)를 복호화하기 위해 Pub_{dc}+DS_2 값을 통해 복호화하고 D_2에 저장하고 다음 전송을 위해 새로운 세션키 S_{k3}을 생성한다.

Fig. 6은 본 논문에서 제안하는 개선방식 "Third-Step 인증" 이다. 이 인증 또한 앞의 방식과 동일하다. "서비스"와 "외부 사용자"에서의 전달되는 데이터를 보낼 때 인증 방식을 보여준다. 두 엔티티(서비스, 외부 사용자) 사이에서도 기존의 데이터 전달 방식을 Fig. 6과 같이 인증을 통한 방식으로 제안하였다. "스마트시티 운영 시스템"에서도Fig. 3-5와 동일하게 인증에 필요한 키쌍을 기존에 생성된 값을 유지하고 "외부 사용자"의 Pub_{di}를 전달 받는다. 그리고 세션키(S_{k3})을 이전과 동일하게 유지한다. 세션키 S_{k3}를 안전하게 "외부 사용자"로 전달하기 위해 Pub_{di}와 Pri_{dg}로 암호화하여 그값을 PS_3에 저장한다. 마지막으로 계량데이터 D_2 데이터를 Pub_{dg}+ S_{k3}으로 암호화하여 C_3에 저장하여 업무 수행을 종료한다. 그 이후에 메시지 M에 Pub_{dg}, PS(S_{k3})와 C_3을 첨부하고 "외부 사용자"에 M을 전송한다. "외부 사용자"는 키쌍(Pub_{di}, Pri_{di})을 생성하고 Pub_{di}를 "서비스"에 전달한다. "외부 사용자" 세션 연결 후 자동으로 전달(Fig. 5와 동일). "외부 사용자"는 메시지 M을 획득한 이후에 Pub_{dg}를 메모리에 저장하고 세션키 S_{k3}를 얻기 위해 Pri_{di}와 Pub_{dg}를 통해 PS_3을 복호화하여 DS_3에 저장한다. 그리고 데이터 암호화를 위한 새로운 세션키 S_{k4}를 생성한다.

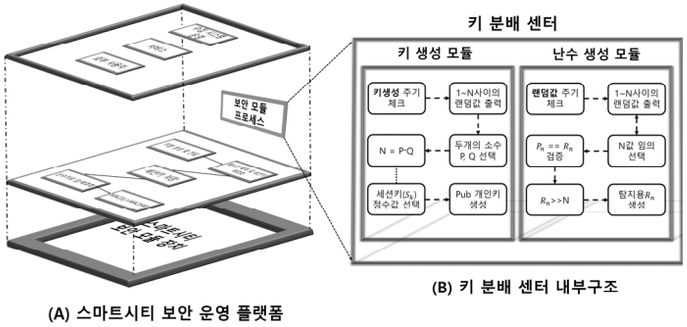

Fig 8. 스마트시티 분배장치 보안 플랫폼.

Fig. 7은 본 논문에서 제안하는 개선방식 "Fourth-Step 인증" 이다. Fig. 7은 "외부 사용자", "서비스"와 "스마트시티 운영 시스템"까지 전송되는 데이터를 보낼 때 인증 방식을 보여준다. "외부 사용자"에서는 인증에 필요한 기존에 생성된 키쌍(Pub_{di}, Pri_{di})과 세션키(S_{k4})를 유지한 상태에서 "서비스"의 Pub_{df}를 전달받는다. 그리고 세션키 S_{k4}를 안전하게 "서비스"로 전달하기 위해 Pub_{df}와 Pri_{di}로 암호화하여 그 값을 PS_4 변수에 저장한다. 마지막으로 데이터 D_3을 Pub_{di}+S_{k4}를 통해 암호화하여 C_4에 저장한다. 그 이후에 Pub_{di}, PS_4(S_{k4})와 를 첨부한 메시지 M을 "서비스"에 전송한다. "서비스"는 키쌍(Pub_{di}, Pri_{di})을 생성하고 Pub_{di}를 "외부 사용자"에 전달한다(Fig들과 동일한 프로세스). "서비스"는 "스마트시티 운영 시스템"과 세션 연결을 하고 "스마트 시티 운영 시스템"의 내부 키 생성 모듈에서(안전한 내부망에서 생성된 키를 통해 MAC 검증 시도) 생성된 PW키를 전달받아 메모리에 저장한다. 그리고 "외부 사용자"로부터 받은 메시지 M에 있는 PS_4를 복호화하기 위해 Pri_{di}와 Pub_{di}를 통해 데이터 복호를 하고 DS_4 변수에 그 값을 저장한다. 또한 전달된 데이터를 얻기위해 C_3을 Pub_{dg}+S_{k3}을 통해 복호화하여 D_3 변수에 저장한다. 그리고 전달된 데이터의 무결성 검증을 위해 D_3을 사전에 전달받은 PW키를 통해 MAC값을 생성[PW[M||MAC(D_3)]하고 HM에 저장한다. 마지막으로 D_3과 HM을 메시지 M에 첨부하여 "스마트시티 운영 시스템"에 전송한다. "스마트시티 운영 시스템"은 메시지 M을 획득하고 무결성 검증 진행 작업을 시도한다. D_3을 PW키를 통해 MAC값을 검출하고 "서비스"로부터 받은 HM과 비교한다. 비교한 검출값이 일치하면 데이터 위·변조 검증 판별에 성공하면 저장소에 데이터 적재를 하고 실패하면 폐기한다.

B. 보안 플랫폼

Fig. 8은 보안 모듈 프로세스 계층을 가진 운영 플랫폼과 해당 플랫폼 내에서의 키생성 및 난수 생성 모듈 장치 내부 구조를 보여준다. (A) 스마트시티 보안 운영 플랫폼은 "수집 시스템 운영", "서비스", "외부 사용자" 순의 계층을 가지며, 보안 모듈 프로세스는 디바이스의 검증 과정들을 나타낸다. (B) 키 분배 센터에서는 키 생성 모듈과 난수 생성 모듈이 있다. 각 프로세스 동작 과정은 다음과 같다. 키 생성 모듈에서는

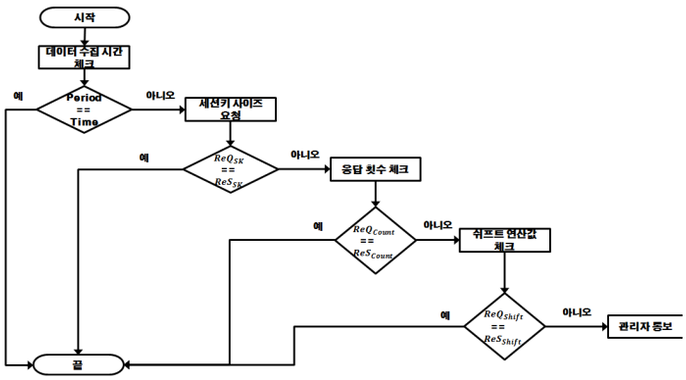

Fig 9. 공격자 이상행동 탐지 알고리즘.

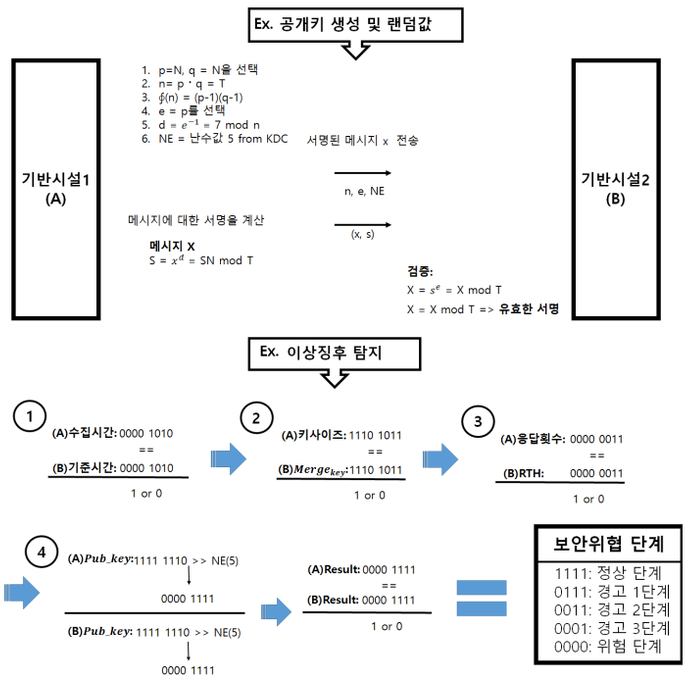

Fig 10. 공개키/난수 및 이상징후 탐지 내부 프로세스 절차.

관리자 보안 정책 기준으로 설정된 키생성 주기를 체크하여 임계치값이 확인되면 바로 키생성 동작 과정을 수행한다. 다음과 같이 공개키 쌍, 세션키(S_k), 개인키(Pub)를 생성하는 순으로 프로세스가 진행되며, 키들이 생성되면 트리거가 바로 난수생성 모듈을 동작 시킨다. 난수생성 모듈 또한 랜덤값 주기 체크를 확인하고 탐지용 랜덤값 생성을 Fig. 8처럼 진행한다. 난수생성 모듈에서 생성된 값은 Fig. 9에서처럼 공격 탐지용에 사용된다.

Fig. 9는 앞서 설명한 공격자 이상행동 탐지 알고리즘에 대한 순서도를 나타내었다. 운영 시스템에서 수집 시스템에 대한 데이터 수집 시간을 항상 체크하다가 운영시 수집되는 데이터의 양 또는 수집되는 시간의 주기가 달라질 경우 기존의 평균 데이터양과 시간값을 체크하고 정책에 의해 결정된 값과 이상이 없으면 별도 테이블에 기록하고 알고리즘을 종료한다. 만약 다를 경우 기존의 세션키 사이즈를 요청하여 세션키 사이즈를 비교해보고 동일하면 알고리즘을 종료하고 그렇지 않다면 응답 횟수 체크를 해본다. 응답 횟수 체크는 관리자에 의해서 결정된 횟수만큼 수집 시스템쪽에 요청을 하

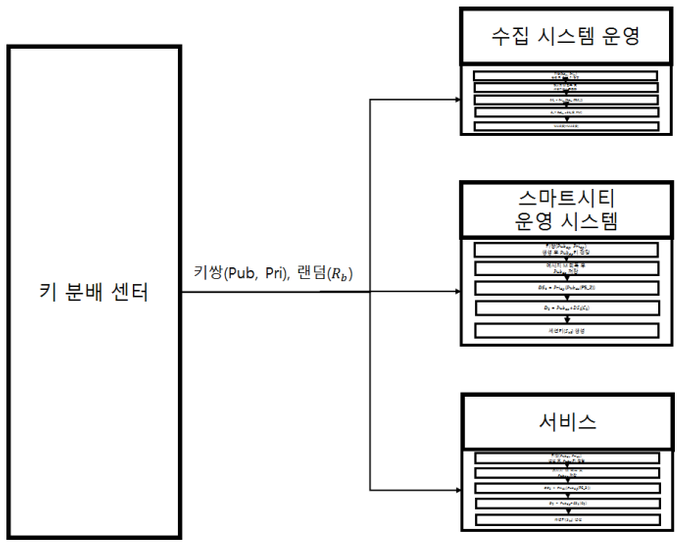

Fig 11. 키쌍 및 랜덤값 분배 과정.

는데 결정된 횟수에 응답을 하면 해당 프로세스를 종료하고 그 횟수에 응답하지 않을 경우 마지막 검증인 쉬프트 연산값 체크 함수가 실행된다. 쉬프트 연산은 최초 설립 당시에 메모리에 기록된 데이터와 연산값을 통해 데이터를 변경하는데 쉬프트된 데이터를 운영 시스템에 전달한다. 운영 시스템은 이를 비교해보고 맞으면 기록 후 알고리즘을 종료하고 맞지 않을 경우 시스템 관리자에게 통보한다. 마지막으로 위 알고리즘의 프로세스는 이상행위 탐지를 통해 보안성을 향상시킬 수 있는 가능성은 충분히 있지만 검증 순서에 따라 보안강도가 계속적으로 높아지는 것은 아니기 때문에 검증 순서와는 무관하다.

Fig. 10은 공개키 생성 및 난수값(랜덤) 전달과 난수값을 통한 이상징후 탐지 내부 처리 절차 과정을 표현하였다. 우선 "공개키 생성 및 랜덤값" 과정에서의 1~5에 해당하는 수식들은 기존에 오픈된 공개키 생성 산출 과정을 나타낸다 [4]. 6은 본 논문에서 사용될 난수값을 추가하였다. 본 논문에서는 위 과정에서 보안점검에 사용되는 난수값 NE를 KDC로부터 할당받아 상호인증 할 때 사용하는 방식으로 운영된다. 해당 난수값을 KDC로부터 전달 받은 기반 시설 B는 메모리에 난수값을 저장한다. 그 이후에 이상징후 탐지 프로세스가 동작될 때 앞서 설명한 Fig. 9와 같은 순서로 프로세스가 진행된다. ①에서는 예를들어 데이터가 수집되는 정상 기준시간이 10 초(시, 분 초 등)라고 가정했을 때 A의 수집 시간과 B의 기준시간이 동일해야 True(1)값이 생성된다. True일 경우, ②는 기존에 세션키에 Merge를 위한 11(1011)이라는 내장된 정수값(2진 비트)을 병합하여 동일한 값인지 체크하고 True(1)을 생성한다. ③에서는 응답횟수 체크하는 부분인데 각 기반시설에서 공유한 카운트 Threshold인데 B가 A에게 Threshold만큼 응답을 했을 때 Respond를 수신해야한다. 만약 그렇지 않다면 False(0)이고 정확한 Threshold일 때 온다면 True(1)를 생성한다. 마지막으로 ④에서는 기존에 시설 설립 후 KDC로부터 할당받은 랜덤값 즉 보안검증에 사용되는 보안 난수값인데 이 값으로 마지막 이상징후 여부를 판단한다. B는 A의 공개키로 A는 자신의 공개키로 할당받은 NE(5)를 기준으로 그만큼 시프트 연산(왼쪽 또는 오른쪽 등)하여 동일한 값이 나오는지

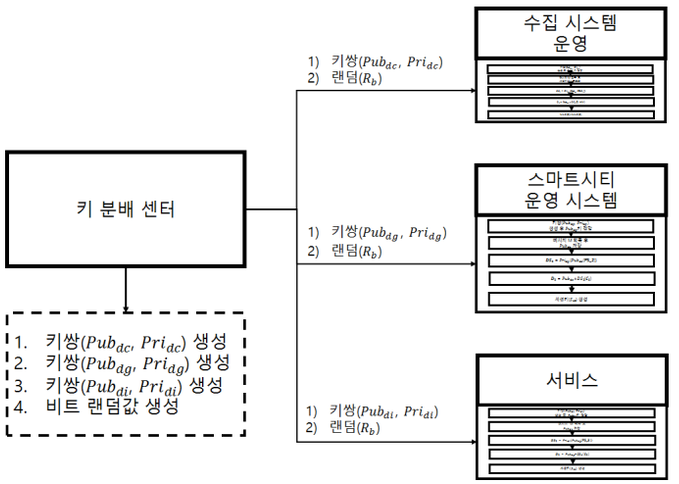

Fig 12. 키쌍 및 랜덤값 생성 과정.

판단하고 맞으면 True(1), 틀리면 False(0)를 생성한다. 또한 서로의 공개키를 쉬프트하여 비교해도 무관하다(ex. A는 B의 공개키, B는 A의 공개키를 사용) 따라서 위와 같이 4단계 까지 이상징후 탐지가 됐을 때 Fig. 10처럼 5단계의 보안 위협 단계를 통해 담당자에게 이상 결과를 통보한다.

Fig. 11은 “키분배 센터(KDC)”에서 데이터 “수집부, 서비스, 외부 사용자”에 전달할 키쌍(*Pub*, *Pri*)과 공개키 Pub의 비트를 쪼개기 위한 랜덤값을 전달하는 중요한 역할을 한다(“수집부, 서비스, 외부 사용자”; 이하 “수신부”). “키 분배 센터”는 “수신부”에 전달하기 위한 키쌍(*Pub*, *Pri*)과 랜덤값(R_b)을 생성하는 절차는 Fig. 12와 같다.

Fig. 12는 3개의 “수신부”에 생성한 각 키쌍과 비트 랜덤값을 전달한다. Fig. 12처럼 키 분배 센터는 초기에 키쌍(Pub_{dc}, Pri_{dc}, Pub_{dg}, Pri_{dg}, Pub_{di}, Pri_{di})와 비트 랜덤값을 생성하여 각 “수신부”에 전달한다. 그 이후에 키 분배 센터는 키의 안전성을 제공하기 위해 일정 주기(시간: 초, 분, 시, 횟수 등)를 통해 각각의 키쌍들과 랜덤값을 재생성한다. 그리고 다시 각 “수신부”에 전달한다. 키 분배 센터와 3개의 “수신부”는 장비를 설비하기전에 하드웨어 자체에 내장된 대칭키가 있다고 가정한다. 이 대칭키는 각 “수신부”에 키쌍과 비트 랜덤값을 외부로부터 안전하게 전달하기 위해 ARIA(국내표준)와 AES(국제표준) 등과 같은 암호 알고리즘으로 암호화 한 이후에 전달한다. 그 이후에 3개의 각 “수신부”들은 저장된 대칭키를 통해 키 분배 센터로부터 전달된 키쌍과 비트 랜덤값을 복호화 하여 그 값들을 메모리에 저장한다.

V. 결론 및 향후 연구

본 논문에서는 지속가능 발전, 저탄소사회 및 통합검침 시스템과 도시자원 통합관리를 구현하는 스마트시티 플랫폼 데이터 운영에 적용할 보안기술의 취약점을 개선하기 위해 제안된 기법을 통해서 향후 중간자 공격, DoS, 위/변조 데이터, 수집 영역 장비 탈취 등에 대한 보안 위협에 대한 대응 방법을 살펴보았다. 또한 이와같은 보안 모듈 장치를 적용할 경우, 기존의 보안기술과는 달리 정책 규칙 등에 의한 KDC의 키 분배를 통해 고정키 사용의 단점을 보완하여 보안키의 안전성을 향상시켰고 또한 외부자의 이상 행동 탐지 검증으로 보안위협 (DoS, 데이터 위/변조, 보안키 탈취 등)에 대응할 수 있기 때문에 한층 더 높은 개선된 보안기술 품질을 제공할 수 있으며, 스마트시티 플랫폼 데이터 운영에서 발생되는 개인정보 및 설비 데이터 등에 대한 유출 방지와 데이터 신뢰성을 높일 수 있다. 따라서 위와 같이 제안된 보안 시스템 적용으로 인해 기존의 보안기술보다 향상된 보안성을 제공할 수 있고 향후 발생될 수 있는 보안위협에 대응할 수 있기 때문에 스마트시티 플랫폼 데이터 운영의 인프라 설비 데이터, 통신에서의 보안성, 자원 낭비 등에 대한 효율성을 높일 수 있다. 향후 연구 주제로는 해당 알고리즘을 시뮬레이션 수행을 통하여 검증 성능 및 최적화 방안에 대한 연구를 진행할 예정이다.

REFERENCES

[1] Gharaibeh, A., A. Salahuddin, M., J. Hussini, S., Khreishah, A., Khalil, I., Guizani, M., Al-fuqaha, A., “Smart Cities: A Survey on Data Management, Security and Enabling Technologies,” IEEE, August, 2017, pp. 1-1.
[2] S. Shravya, K., Deepak, A., Chandrasekaran, K., “Smart key generation for smart cities,” IEEE, Feb, 2017, pp. 9-10.
[3] Wang, P., Ali, A., Kelly, W., “Data security and threat modeling for smart city infrastructure,” IEEE, Aug, 2015, pp. 5-7.
[4] Rivest, Shamir, Adleman, “RSA Algorithm,” MIT, 1997.