보안 클라우드 기반 NFC 모바일 지불 프로토콜 정형화 검증

박현성

순천향대학교 정보보호학과

Formulation Verification of Secure Cloud-Based NFC Mobile Payment Protocol

Hyeon-Seong Park

Division of information security, Soonchunhayng University.

요 약

근거리 통신(NFC)은 스마트폰을 통한 어플리케이션 개발 및 서비스 제공 분야의 최신 기술 중 하나이다. NFC는 스마트폰을 신원 확인 및 사용자의 신용카드로 작동할 수 있다. 보안 NFC 거래를 보장하는 데 사용할 수 있는 기술 중 하나는 NFC 지원 스마트폰에서 단일 요소로 보안 요소를 사용하는 것과 비교하여 광범위한 이점을 제공하는 클라우드 컴퓨팅이다. 해당 기술을 응용하여 해외 학술지에서 제안한 클라우드 컴퓨팅을 응용한 보안 클라우드 기반 NFC 모바일 지불 프로토콜에 대한 안전성을 확인하기 위해 정형화 검증 도구를 사용하여 안전성을 도출한다.

I. 서론

서비스 환경에서 쉬운 협력 관계를 구축하기 위해 NFC 기술을 사용하는 산업 분야에서 기술 표준과 상호 운용성이 필수적이다. 실제로 서비스 환경의 복잡한 응용프로그램 수준에서 상호 운용성이 부족하여 NFC 기술은 느리게 발전되었다. 또한, 현재의 서비스 어플리케이션은 지불 환경에 적합한 솔루션을 제공하지 못하므로 적합한 기술이 아니다. 이에 따라 현재 NFC 생태계 모델을 확장하여 비즈니스 영역의 개발을 가속화하기 위해 해당 프로토콜을 제안하였다.

프로토콜 보안의 중요성은 점차 증가하고 있으며 인터넷에 연결되어 통신하는 기기가 증가할수록 기기 간 상호 인증과 메시지에 대한 기밀성 요구가 높아지고 있다. 이에 따라 본 논문에서는 해외 학회지에서 제안되었던 '보안 클라우드 기반 NFC 모바일 지불 프로토콜'의 정형화 검증을 통해 프로토콜의 안전성을 증명하여 프로토콜의 상용화에 일조할 수 있을 것으로 사료된다.

본 논문의 구성은 다음과 같다. 2장에서는 프로토콜을 설명하고, 3장에서 BAN Logic을 이용하여 정형화검증을 수행하며 4장에서 AVISPA를 이용한 정형화 검증을 수행한다. 마지막으로 5장에서는 결론으로 논문을 마무리한다.

II. 프로토콜 분석

보안 클라우드 기반 NFC 모바일 지불 프로토콜(이하 프로토콜)에서 참여자는 이동통신사와 모바일기기, 상점NFC로 구성된다. 이동통신사와 모바일기기는 GSM Link로 연결되고, 모바일기기와 NFC단말기는 NFC Link로 연결된다. 이동통신사와 모바일기기가 상호인증을 통해 신용 거래 메시지를 생성하고 모바일기기와 상점에게 보냄으로써 신용 거래메시지와 거래

내용의 유효성을 확인하는 과정의 프로토콜이다.

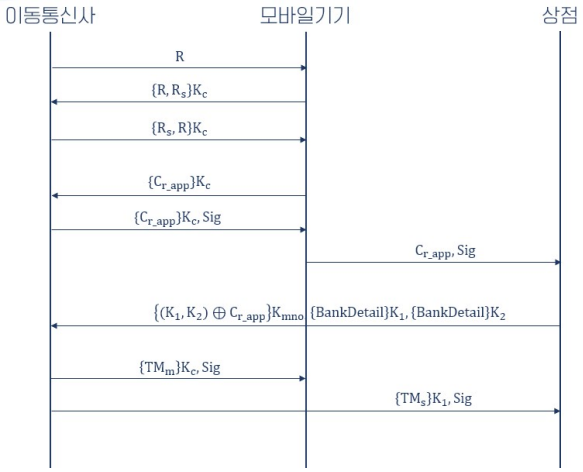
프로토콜의 정형화 검증에 앞서, 몇 가지의 가정이 필요하다. 첫 번째, 이동통신사, 모바일기기에는 서로의 128bits의 난수로 대칭키K_c를 발급할 수 있는 암호화 모듈이 포함되어있다. 두 번째, 보이지 않는 참여자(금융 기관)이 존재한다. 세 번째, 모바일기기와 상점은 근거리 통신인 NFC를 사용하기 때문에 서로를 신뢰할 수 있다고 가정한다.

프로토콜의 주요 약어는 [표 2]와 같다.

[표 2] 약어

AuC	인증 센터(MNO 하위 시스템)
App_{ID}	승인 ID. 신용 승인 후 생성
Acc_{ID}	고객 계정 ID
C_{r-req}	신용 요청 메시지
C_{r-app}	신용 승인 메시지
$IMSI$	인터넷 모바일 가입자 ID
K_i	특정 SIM키. SIM 및 AuC의 안전한 위치에 보관됨
K_c	A8 알고리즘을 사용하는 $E_{ki}(R)$
K_1	상점에 의해 생성된 암호화키
K_2	상점에 의해 생성된 MAC 값
K_{pub}	이동 통신 사업자의 공개키
K_{pr}	이동 통신 사업자의 비밀키
K_{sign}	이동 통신 사업자의 전자서명키
K_{ver}	이동 통신 사업자의 검증키
LAI	로컬 영역 식별자
MNO	이동 통신 사업자
R	이동 통신 사업자에 의해 생성된 128bits 난수 값
R_s	SIM에 의해 생성된 128bits 난수 값
SE	보안 요소
TM_m	모바일 거래 메시지
TM_s	상점 거래 메시지
$TMSI$	임시 이동 가입자 신원
TP	거래 총 가격
T_{SID}	임시 상점 ID
TS_s	상점 타임스탬프
TS_t	거래 타임스탬프

제시되었던 프로토콜 다이어그램에서 정형화 검증에 필요하지 않은 과정은 제거한 다이어그램은 [그림 1]과 같다. 정형화 검증에서 디지털 서명 과정에 대한 부분은 난이도가 높으므로 본 논문에서 다루는 정형화 검증 과정 중 디지털 서명 과정은 제외하기로 한다.



[그림 1] 프로토콜 다이어그램

Step1. 트리플렛 난수 생성

이동통신사는 임의의 난수를 생성하여 트리플렛으로 대칭키K_c를 생성한다. 그 후 모바일기기에서 같은 작업을 수행할 수 있도록 난수 R을 전송한다.

Step2. 인증 난수 생성

모바일기기는 전송받은 R을 이용하여 K_c를 생성하고 상호 인증을 위한 R_s를 생성하여 이동통신사에게 전송한다. 이 때, 두 난수를 생성한 대칭키로 암호화하여 전송한다.

Step3. 상호인증

이동통신사는 Step2에서 전송받은 데이터를 통해 모바일기기를 인증하고 두 난수의 자리를 바꾸어 다시 모바일기기로 전송한다. 모바일기기는 R_s를 검사하여 이동통신사를 인증한다. 상호 인증이 완료되면 K_c를 이용한 암호화로 메시지를 주고받는다.

Step4. 신용 요청 메시지 생성

모바일기기는 신용요청 메시지를 생성하여 이동통신사에 전송한다. 신용요청 메시지에는

구매금액, 상점ID, 타임스탬프, 임시 소비자 신원정보, 소비자 ID가 있다. 이동통신사는 메시지를 해독하여 신용검증을 하게 되는데, 신용검증은 금융기관을 통해 확인하게 된다. 신용이 확인되면 신용 승인 ID가 발급된다.

Step5. 신용 승인(1)

이동통신사는 신용 요청 메시지에 담긴 데이터와 신용 승인 ID가 담긴 신용 승인 메시지와 서명을 암호화하여 모바일기기에 전송한다. 모바일기기는 해당 메시지의 전자서명을 검증한다.

Step6. 신용 승인(2)

모바일기기가 검증한 신용 승인 메시지는 상점 NFC단말기로 전송한다. 이 때, 근거리 통신을 이용하므로 암호화되지 않은 메시지로 전달된다. NFC단말기는 모바일기기와 동일하게 신용 승인 메시지를 검증한다.

Step7. 지불 정보 생성

NFC단말기는 신용 승인 메시지의 검증이 확인되면 대칭키K_1과 K_2를 생성한다. 생성된 키는 각각 자신의 계좌 정보를 암호화하고 MAC계산을 위해 사용된다. K_1과 K_2는 신용 승인 메시지에 담긴 신용 승인 ID(다이어그램에서는 신용 승인 메시지로 통일함)와 XOR연산을 마친 후, 이동통신사의 공개키로 암호화하여 이동통신사에게 전송한다. 3개의 메시지를 전송받은 이동통신사는 받은 정보를 해독하고 금융기관에 전달하여 최종 거래 메시지를 생성한다.

Step8. 최종 거래 메시지 검증

이동통신사는 최종 거래 메시지를 암호화하여 전자서명과 함께 모바일기기와 NFC단말기에 전송한다. 이때, 모바일기기에서 전자서명 검증이 확인되어야만 NFC단말기로 최종 거래 메시지가 전달된다.

III.BAN Logic 정형화 검증

이동통신사는 MNO(Mobile Network Operator), 모바일기기는 MD(Mobile Device), 상점은 SH(Shop)으로 나타내었다.

Idealization은 [그림 2]와 같다. 정형화 검증을 보기 쉽게 나타내기 위해 [그림 3]과 같이 변형하였다. MNO는 A, MD는 B 그리고 SH는 C로 표현하였다. Idealization과정 중 3번과 4번, 6번, 7번의 메시지에는 타임스탬프가 담겨있다.

1) MD → MNO : $\{R, R_s\}K_c$
2) MNO → MD : $\{R_s, R\}K_c$
3) MD → MNO : $\{C_{r\_req}\}K_c$
4) MNO → MD : $\{C_{r\_app}\}K_c$
5) SH → MNO : $\{(K_1, K_2) \oplus C_{r\_app}\}K_{mno}$
6) MNO → MD : $\{TM_m\}K_c$
7) MNO → SH : $\{TM_s\}K_1$

[그림 2] Idealization 원형

1) B → A : $\{R, R_s\}K_c$
2) A → B : $\{R_s, R\}K_c$
3) B → A : $\{C_{r\_req}\}K_c$
4) A → B : $\{C_{r\_app}\}K_c$
5) C → A : $\{(K_1, K_2) \oplus C_{r\_app}\}K_{mno}$
6) A → B : $\{TM_m\}K_c$
7) A → C : $\{TM_s\}K_1$

[그림 3] Idealization 변형

Assumption은 [그림 4]와 같다. 이동통신사와 모바일기기는 난수와 내장 암호화 모듈을 통해 대칭키를 생성할 수 있으므로 <A1>과 <A3>이 성립한다. 이동통신사와 모바일기기가 각각 생성한 난수는 스스로가 fresh하다고 할 수 있으므로 <A2>, <A4>이 성립한다. 이동통신사는 자신의 비대칭키를 믿을 수 있으므로 <A5>가 성립한다. 상점은 자신이 생성한 2개의 암호화키는 스스로가 fresh하다고 할 수 있

고, 믿을 수 있으므로 <A6>, <A7>이 성립한다. 이동통신사, 모바일기기, 상점은 타임스탬프(*TS*)를 항상 fresh하다고 할 수 있으므로 <A8>, <A9> 그리고 <A10>이 성립한다.

<A1> A believes A $\overset{K_c}{\leftrightarrow}$ B

<A2> A believes fresh(R)

<A3> B believes A $\overset{K_c}{\leftrightarrow}$ B

<A4> B believes fresh(R_s)

<A5> A believes $\overset{K}{\rightarrow}$ A

<A6> C believes fresh(K_1, K_2)

<A7> C believes believes (K_1, K_2)

<A8> A believes fresh(TS)

<A9> B believes fresh(TS)

<A10> C believes fresh(TS)

[그림 4] Assumption

Goal은 [그림 5]와 같다. 이동통신사와 모바일기기 간 난수가 올바르게 교환했는지 확인해야한다. 이동통신사와 모바일기기 간 교환한 신용 요청 메시지나 신용 승인 메시지가 올바른지 확인해야한다. 이동통신사는 상점에서 발급한 2개의 키가 상점에게서 발급된 사실을 믿을 수 있어야한다. 모바일기기와 상점은 이동통신사에게 받은 거래메시지가 올바른 메시지임을 확인해야한다.

<G1> A believes B believes {R, R_s}

<G2> B believes A believes {R, R_s}

<G3> A believes B believes C_{r_req}

<G4> B believes A believes C_{r_app}

<G5> A believes C believes K_1, K_2

<G6> B believes A believes TM_m

<G7> C believes A believes TM_s

[그림 5] Goal

Derivation은 [그림 6, 7, 8]과 같다.

<D1> A sees {R, R_s}K_c

<D2> A believes B said {R, R_s} / MM, D1, A1

<D3> A believes B believes {R, R_s} / NV, D2, A2

<D4> B sees {R, R_s}K_c

<D5> B believes A said {R, R_s} / MM, D4, A3

<D6> B believes A believes {R, R_s} / NV, D5, A4

<D7> A sees {C_{r_req}}K_c

<D8> A believes B said C_{r_req} / MM ,D7, A1

<D9> A believes B believes C_{r_req} / NV, D8, A8

[그림 6] Derivation(1)

<D10> B sees {C_{r_app}}K_c

<D11> B believes A said C_{r_app} / MM, D10, A3

<D12> B believes A believes C_{r_app} / NV, D11, A9

<D13> A sees {(K_1, K_2) ⊕ C_{r_app}}K_{mno}

<D14> A believes B said {(K_1, K_2) ⊕ C_{r_app}} / MM, D13, A5

<D15> A believes B believes (K_1, K_2)/ NV, DCO, D14, D12, A8

[그림 7] Derivation(2)

<D16> B sees {TM_m}K_c

<D17> B believes A said T_m / MM, D16, A3

<D18> B believes A believes TM_m / NV, D17, A8

<D19> C sees {TM_s}K_1

<D20> C believes A said T_s / MM, D19, A7

<D21> C believes A believes TM_s / NV, D20, A6

[그림 8] Derivation(3)

IV. AVISPA 정형화 검증

AVISPA에서 가장 중요한 부분인 goal은 다음과 같이 설정하였다. K_c, K_1, K_2의 기밀성 유지와 이동통신사, 모바일기기 간 상호 인증을 목표로 하였다. 공격자 사전지식은 이동통신사 ID, 모바일기기ID, 상점ID, 이동통신사 공개키로 설정하였다.

AVISPA를 이용한 정형화 검증에 사용한 HLPSL코드는 다음 그림을 참조한다. 이동통신사는 MNO(Mobile Network Operator), 모바일기기는 MD(Mobile Device), 상점은 SH(Shop)으로 나타내었다.

```
role environment() def=

    const md, mno, sh : agent,
          kc, k1, k2 : symmetric_key,
          k_mno : public_key,
          mk_md, mk_mno : protocol_id

    intruder_knowledge = {md, mno, sh, k_mno}

    composition
        session(md, mno, sh, kc, k1, k2, k_mno)

end role

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

goal
    secrecy_of kc, k1, k2
    authentication_on mk_md
    authentication_on mk_mno
end goal
```

[그림 9] Environment Role과 Goal

```
role session(MD, MNO, SH: agent,
             Kc , K1, K2 : symmetric_key,
             K_MNO : public_key) def=

    local SMD,RMD,SMN,RMN,SSH,RSH: channel(dy)

    composition
        mobile_device(MD,MNO,Kc,SMD,RMD,K_MNO)/\
        mobile_network_operator(MD,MNO,SH,Kc,
        K1,K2,SMN,RMN,K_MNO)/\
        shop(MNO,SH,Kc,K1,K2,SSH,RSH,K_MNO)

end role
```

[그림 10] Composition Role

State 0. 이동통신사 - 모바일기기

이동통신사는 start 메시지를 받으면 128bits의 난수R을 생성하여 모바일기기로 전송한다.

State 1. 모바일기기 - 이동통신사

R을 전송받은 모바일 기기는 128bits의 난수 R_s를 생성하고 R로부터 파생된 대칭키K_c로 두 난수를 암호화하여 이동통신사로 전송한다. 이때 witness를 이용하여 이동통신사에게 인증대기를 알린다.

State 2. 이동통신사 - 모바일기기

이동통신사는 두 난수의 자리를 바꾸어 모바일기기로 전송한다. 이동통신사는 request를 통하여 모바일기기의 witness를 인증하고, 모바일기기와 마찬가지로 witness를 사용하여 모바일기기에게 인증대기를 알린다.

State 3. 모바일기기 - 이동통신사

자리가 바뀐 두 난수를 받은 모바일기기는 신용 요청 메시지를 생성하고, 대칭키로 암호화하여 이동통신사에게 전송한다. 모바일기기와 마찬가지로 request를 통해 이동통신사의 witness를 인증한다.

State 4. 이동통신사 - 모바일기기

신용 요청 메시지를 받은 이동통신사는 금융기관의 신용 승인을 받았다는 가정 하에 신용 승인 메시지를 생성하여 암호화한다. 그 다음, 모바일기기로 전송한다.

State 5. 모바일기기 - 상점

신용 승인 메시지를 받은 모바일기기는 전자서명을 통해 메시지를 검증했다고 가정한다. 복호화한 메시지를 상점 NFC단말기로 전송한다.

State 6(State' 8). 상점 - 이동통신사

신용 승인 메시지를 받은 NFC단말기는 모바일기기와 마찬가지로 전자서명을 통해 메시지를 검증했다고 가정한다. 대칭키K_1, K_2를 생성하고 신용 승인 메시지와 XOR연산 후 이동통신사의 공개키로 암호화하여 전송한다. 상점

의 계좌 정보는 K_1과 K_2로 암호화되어 전송된다.

State 6(State' 9). 이동통신사 - 모바일기기, 상점

전달 받은 계좌 정보의 무결성을 검사하고 금융기관을 통해 최종 거래 메시지를 생성한다. 각각의 거래 메시지는 이동통신사로 전송할 때 K_c로, 상점에게 전송할 때 K_1으로 암호화하여 각각 보낸다.

State 7, 8 모바일기기, 상점

이동통신사로부터 전달받은 메시지를 복호화하여 전자서명을 검증한다. 검증이 완료되면 지불이 완료된다.

```
role mobile_device(MD, MNO : agent,
                   Kc : symmetric_key,
                   SND,RCV : channel(dy),
                   K_MNO: public_key)
played_by MD
def=

    local State: nat,
        R: text,
        Rs: text,
        C_req: text,
        C_app: text,
        Tm: text,
        MK_MD, MK_MNO: message

    init State := 1

    transition

        1. State = 1 /\ RCV(R') =|>
           State':= 3 /\ Rs':= new()
                      /\ MK_MD':= R'.Rs'
                      /\ MK_MNO':= Rs'.R'
                      /\ SND({R'.Rs'}_Kc)
                      /\ witness(MD,MNO,mk_mno,MK_MNO')

        2. State = 3 /\ RCV({Rs.R}_Kc) =|>
           State':= 5 /\ C_req' := new()
                      /\ SND({C_req'}_Kc)
                      /\ request(MD,MNO,mk_md,MK_MD)

        3. State = 5 /\ RCV({C_app'}_Kc) =|>
           State':= 7 /\ SND(C_app')

        4. State = 7 /\ RCV({Tm'}_Kc) =|>
           State':= 10

end role
```

[그림 11] Mobile Device Basic Role

```
role mobile_network_operator(MD,MNO,SH : agent,
                             Kc, K1, K2 : symmetric_key,
                             SND, RCV : channel(dy),
                             K_MNO: public_key)
played_by MNO
def=

    local
        State: nat,
        R: text,
        Rs: text,
        C_req: text,
        C_app: text,
        Tm: text,
        Ts: text,
        MK_MD, MK_MNO: message

    init State := 0

    transition

        1. State = 0 /\ RCV(start) =|>
           State':= 2 /\ R' := new() /\ SND(R')

        2. State = 2  /\ RCV({R.Rs'}_Kc) =|>
           State':= 4 /\ MK_MNO':= Rs'.R
                      /\ MK_MD':= R.Rs'
                      /\ SND({Rs'.R}_Kc)
                      /\ request(MNO,MD,mk_mno,MK_MNO')
                      /\ witness(MNO,MD,mk_md,MK_MD')

        3. State = 4 /\ RCV({C_req'}_Kc) =|>
           State':= 6 /\ C_app':= new()
                      /\ SND({C_app'}_Kc)

        4. State = 6 /\ RCV({xor(K1.K2,C_app)}_K_MNO) =|>
           State':= 9 /\ Tm':= new()
                      /\ Ts':= new()
                      /\ SND({Tm'}_Kc)
                      /\ SND({Ts'}_K1)

end role
```

[그림 12] Mobile Network Operator Basic Role

```
role shop(MNO, SH:agent,
          Kc, K1, K2: symmetric_key,
          SND, RCV: channel(dy),
          K_MNO: public_key)
played_by SH
def=

    local
        State: nat,
        C_req: text,
        C_app: text,
        Ts: text

    init State := 6

    transition

        1. State = 6 /\ RCV(C_app') =|>
           State':= 8 /\ SND({xor(K1.K2,C_app')}_K_MNO)

        2. State = 8 /\ RCV({Ts'}_K1) =|>
           State':= 10

end role
```

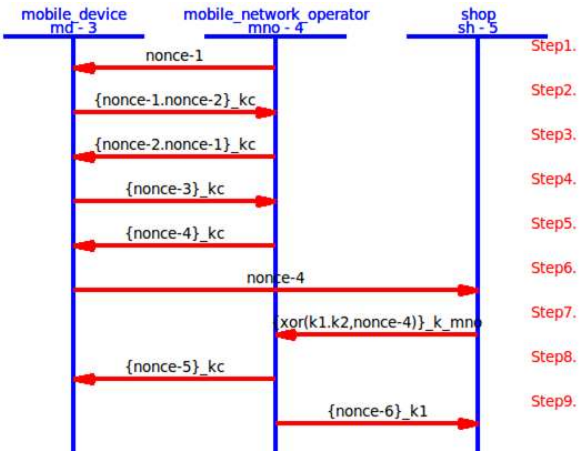
[그림 13] Shop Basic Role

```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  /home/span/span/testsuite/results/project.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.00s
  searchTime: 0.09s
  visitedNodes: 38 nodes
  depth: 10 plies
```

[그림 14] AVISPA 정형화 검증 결과(OFMC)

```
SUMMARY
  SAFE

DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
  TYPED_MODEL

PROTOCOL
  /home/span/span/testsuite/results/project.if

GOAL
  As Specified

BACKEND
  CL-AtSe

STATISTICS

  Analysed   : 85 states
  Reachable  : 33 states
  Translation: 0.02 seconds
  Computation: 0.00 seconds
```

[그림 15] AVISPA 정형화 검증 결과(ATSE)



[그림 16] AVISPA 프로토콜 시뮬레이터

V. 결론

본 논문에서는 보안 클라우드 기반 NFC 모바일 지불 프로토콜에 대한 정형화 검증을 수행하였다. 해당 프로토콜은 BAN Logic과 AVISPA를 이용하여 정형화 검증을 수행했으며, 모두 공격자로부터 안전함을 도출하였다. 하지만 이동통신사와 모바일기기에 키 생성 모듈이 내장되어 있다는 점과 모바일기기와 NFC 단말 간 근거리 통신을 신뢰할 수 없는 상황이 발생할 수 있다는 점 등 다양한 상황에서의 문제점이 나타날 수 있다. 정형화 검증을 통한 안전성과 별개의 연구를 통해 해당 프로토콜의 신뢰성을 향상시켜야할 것으로 보인다.

[참고문헌]

[1] Pardis Pourghomi, Muhammad Qasim Saeed and Gheorghita Ghinea, “A Secure Cloud-Based NFC Mobile Payment Protocol” International Journal of Advanced Computer Science and Applications(IJACSA), 5(10), 2014.