

한국정보보호학회 동계학술대회

CISC-W'18

Conference on Information Security and Cryptography-Winter 2018

일자 2018년 12월 8일(토)

장소 세종대학교(군자관, 총무관)

주최  **한국정보보호학회**
Korea Institute of Information Security & Cryptology

주관  **세종대학교**
SEJONG UNIVERSITY

후원  **국가정보원**
National Intelligence Service Korea

 **행정안전부**

 **한국인터넷진흥원**

 **ETRI** **한국전자통신연구원**
Electronics and Telecommunications Research Institute

 **NSR**
국가보안기술연구소

 **LIG 넥스원**

 **보안프로젝트**
WWW.BSAPROJECT.COM

 **kakao**

Proceedings

포스터발표 트랙 E

10:20 - 11:20	17	크롬환경에서의 웹 로그정보를 활용한 개인 성향분석 방법 연구 김지영, 박예원, 안지은, 양지수, 임규희, 김형중 (서울여자대학교)
	18	OpenSSL에서 제공하는 의사난수 API 구조 분석 연구 김현기, 송행권, 이옥연 (국민대학교)
	19	다크웹 악성유저 추적방안에 대한 연구 안상현, 유동민, 정주현, 김만수, 이유진, 김영용, 이해민 (KITRI BoB)
	20	안드로이드 환경에서의 O2O 서비스 현황 및 보안이슈 문현지, 주완빈, 신지호, 현지산, 정성조, 조민재, 박세한 (KITRI BoB), 이성권 (고려대학교)
	21	소프트웨어 개발 생명주기에서 위협 인텔리전스 적용을 통한 소프트웨어 보안강화 방안 김예준, 류기백, 이유정, 염홍열 (순천향대학교)
	22	DLMS/COSEM 프로토콜을 사용하는 스마트 미터의 취약점에 관한 연구 박준영, 한창현, 피대권, 송상현, 조수민 (한국정보기술연구원)
	23	HastFuzz: 선행 분석 결과 기반의 실행 흐름 유도를 통한 퍼징 방법론 조승현, 이광진, 최광준, 진호용, 나윤종, 박기웅 (세종대학교)
	24	효율적인 악성코드 분석을 위한 에뮬레이션 기반 동적 분석 도구 구현 및 검증 김민재, 임명은, 주하중, 염홍열 (순천향대학교)
	25	민간 클라우드 도입 장애요인 분석을 통한 국방 클라우드 도입 전략 도출 박준규, 전우진 (세종대학교), 이상훈 (국방과학연구소), 박기웅 (세종대학교)
	26	PDF 악성코드 유형분류를 위한 API 시퀀스 정제 방안 연구 김경한, 최보민, 강홍구, 박준형 (한국인터넷진흥원)
	27	탈중앙화 항공 체크인 시스템을 위한 스마트 계약 김혜빈, 박지선, 신상욱 (부경대학교)
	28	자바스크립트 난독화 도구 분석 및 평가 신동순, 김형식 (성균관대학교)
	29	LDA를 이용한 사회공학 공격 산출방안 임승호, 김지원 (육군사관학교)
	30	음성인식 기반 사용자 인증 기술 동향 조사 김베드로, 김형식 (성균관대학교)
	31	An Investigation and Analysis of Blockchain Use Cases for Digital Forensics Sandi Rahmadika, Kyung-Hyune Rhee (부경대학교)
	32	식품 추적 및 제어를 위한 블록체인 솔루션 쵸노에진랏, 이경현 (부경대학교)
	33	머신러닝에 기반한 PE패턴과 악성코드 시각화 분석을 활용하는 분석 방법에 대한 연구 염성규, 신동일, 신동규 (세종대학교)

클라우드 환경에서의 스마트카드를 활용한 개체 인증 개선 방안

조정석*, 꺾진**

*아주대학교 컴퓨터공학과 정보보호응용및보증연구실

**아주대학교 사이버보안학과

Improvement of Entity Authentication Using Smartcard in Cloud Environment

Jeong-Seok Jo*, Jin Kwak**

*ISAA Lab., Department of Computer Engineering, Ajou University.

**Department of Cyber Security, Ajou University.

요 약

클라우드 서비스는 다양한 서비스 제공이 목적인데, 기존의 네트워크와 달리 대량의 자원이 클라우드에 존재하게 된다. 이에 따라 하나의 위협으로 인해 전체의 클라우드 자원에 영향을 끼칠 수 있으므로, 높은 보안성이 요구된다. 이를 위해 클라우드 서버에서의 인증 방안 중 스마트카드를 활용한 다중 요소 인증 방식이 많이 사용되고 있으며, Rohan등은 스마트카드를 활용한 인증을 제시하였다. 하지만 스마트카드를 대상으로 한 공격이 다수 발생하고 있으며, 감염된 스마트카드를 활용하여 제 3자가 권한이 없는 클라우드 자원에 접근하는 문제들이 발생하고 있다.

이에 따라 본 논문에서는 클라우드 환경에서의 스마트카드를 활용한 인증의 개선 방안을 제시하고자 한다.

I. 서론

클라우드 서비스가 등장하면서, 클라우드 환경에서 인증을 위한 다양한 방안들이 등장하였다. Rohan등은 Sarang V. Hartwar의 Multi-factor 인증을 활용하여[1] 클라우드 환경에서의 Multi-factor 방식 스마트카드 인증 프로토콜을 제시하였다[2]. 그러나 Rohan등이 제시한 프로토콜에서 사용자의 PC에 맬웨어를 감염시키고 백도어를 통해 공격자가 원하는 서버에 접근하는 취약점이 존재한다. 이에 따라, 해당 취약점을 분석하고 이에 대한 대응 방안을 제시하고자 한다.

본 논문에서는 2장에서 sykipot 맬웨어 분석 및 Rohan등이 제시한 스마트카드를 활용한 인

증 프로토콜과 해당 프로토콜의 취약점을 분석하였으며, 3장에서 해당 취약점의 대응하는 개선된 인증 프로토콜을 제시하고, 4장에서 Rohan등이 제시한 프로토콜과 개선된 프로토콜을 비교 분석하고, 5장에서 결론을 맺는다.

II. 관련연구

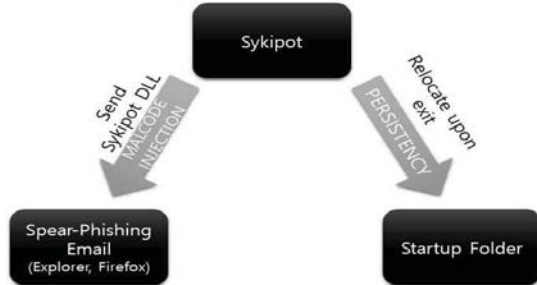
2.1 sykipot 맬웨어

sykipot 맬웨어는 스마트카드를 대상으로 동작하는 맬웨어이다. 해당 맬웨어는 sykipot EXE, sykipot DLL로 구성되어있다. sykipot EXE는 Sykipot DLL이 암호화되지 않은 상태로 내장된 실행파일이며, 동작방식은 사용자가 스피어 피싱 전자 메일에서 악성 링크 혹은 첨부 파일을 열면 Sykipot EXE가 저장되고 실행되는 방식이다. Sykipot의 동작방식은 (그림 1)과 같다.

해당 맬웨어는 키 로깅을 이용해 정보를 얻

이 논문은 2018년도 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(No. NRF-2017R1E1A1A01075110).

고, 백도어를 활용해 공격자가 원하는 서버로 연결한다[3]. 본 논문에서는 백도어를 이용하여 발생 가능한 취약점을 제시하고, 그에 대한 개선 방안을 제시하고자 한다.

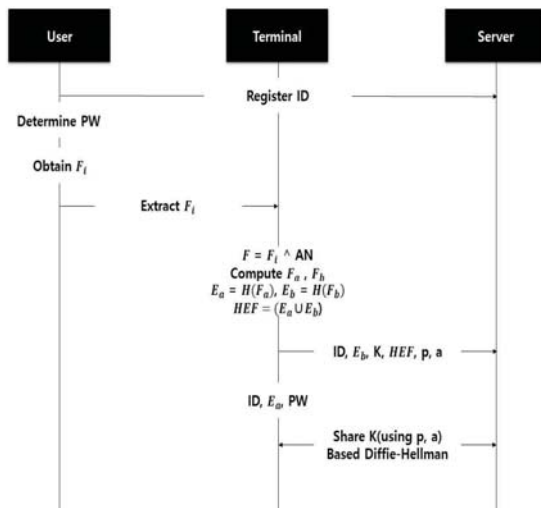


(그림 1) Sykipot 동작방식

2.2 Rohan's Proposed Protocol

Rohan등은 클라우드 환경에서의 스마트카드를 활용한 인증 프로토콜을 새롭게 제안하였다[2]. 제안한 프로토콜의 동작방식은 (그림 2)와 같다.

본 절에서는 Rohan등이 제안한 프로토콜을 분석하고, sykipot 맬웨어를 활용한 취약점을 분석한다.



(그림 2) Rohan's Proposed Protocol

2.2.1 Rohan's Proposed Protocol Parameter

Rohan등이 제시한 프로토콜에서 사용되는 파라미터는 (표 1)과 같다.

대부분의 연산은 Terminal에서 이루어진다.

(표 1) Rohan's Proposed Protocol Parameter

Parameter	Contents
ID, PW	- 사용자의 ID/PW
F_i	- 사용자의 지문
AN	- 고유한 비밀 값
F	- F_i, AN 의 XOR 값
F_a, F_b	- F 의 짝수, 홀수 비트열
E_a, E_b	- F_a, F_b 의 Hash 연산 값
K	- AES 키
HEF	- $E_a \cup E_b$ 의 Hash 연산 값
p	- 임의의 큰 소수
a	- Root A

2.2.2 프로토콜 동작 순서

Rohan등이 제안한 프로토콜 동작 순서는 아래와 같다.

Step 1. 비밀 값 생성

프로토콜에 사용될 파라미터(a, p, K, AN)를 생성한다.

Step 2. ID 등록(User→Server)

User는 Server에 본인의 ID를 등록한다.

Step 3. PW 결정(User)

User는 인증에 사용할 PW를 설정하고 스마트카드에 저장한다.

Step 4. 지문 정보 확보 및 전송(User→Terminal)

User는 본인의 지문 정보를 확보하고 Terminal은 Orientation Maps(OM)을 활용하여, 확보된 지문정보로부터 값을 추출한다.

Step 5. 파라미터 형성(Terminal)

Terminal은 User로부터 받은 지문정보를 활용하여 인증에 활용될 파라미터들을 확보한다. 이 때, (표 1)의 $F, F_a, F_b, E_a, E_b, HEF$ 가 생성된다.

Step 6. 파라미터 전송(Terminal→Server)

Terminal은 확보된 파라미터 중 ID, E_b, K, HEF, p, a 값들을 전송한다.

Step 7. 파라미터 저장(Terminal)

Terminal은 ID, PW, E_a 값들을 저장한다.

Step 8. 키 공유(Terminal $\leftrightarrow$ Server)

Sever와 Terminal은 p, a 기반 Diffie-Hellman 키 교환 방식을 활용하여 AES 키 K 를 공유한다.

Step 9. 인증(User $\leftrightarrow$ Server)

User에게 받은 실시간 지문 정보를 이용하여 F 를 추출하고, 그에 대한 E_a, E_b, HEF 값 정당성을 비교하여 인증을 수행한다.

2.2.3 sykipot 맬웨어 기반 프로토콜 취약점

Rohan등이 제안한 프로토콜은 Terminal에 대한 인증을 진행하지 않고, 지문 정보 및 스마트카드에 등록된 정보를 보낸다.

이에 따라, 공격자가 2.1절에서 분석한 sykipot 맬웨어를 활용하여, 스마트카드에 접근하고 백도어를 이용해 공격자가 원하는 서버와 연결하여, 정당한 Terminal인 척 위장하는 공격이 가능하다. 공격 진행 과정은 아래와 같다.

Step 1. 맬웨어 전송(Attacker $\rightarrow$ User)

Attacker는 User에게 sykipot 맬웨어를 삽입한 전자 메일을 보낸다.

Step 2. 맬웨어 삽입(User)

User는 수신한 전자 메일의 첨부 파일을 다운로드 받게 되고 그에 따라 sykipot 맬웨어에 감염되게 된다.

Step 3. 백도어 연결(Attacker $\rightarrow$ User)

sykipot 맬웨어 감염으로 인해 User의 인증을 위한 Terminal은 Attacker가 원하는 서버가 된다.

Step 4. Rohan 프로토콜 수행(User $\rightarrow$ Attacker)

User는 Terminal에 대한 인증 작업을 수행하지 않고 Rohan등이 제시한 프로토콜을 수행한다.

Step 5. 인증 파라미터 확보(Attacker)

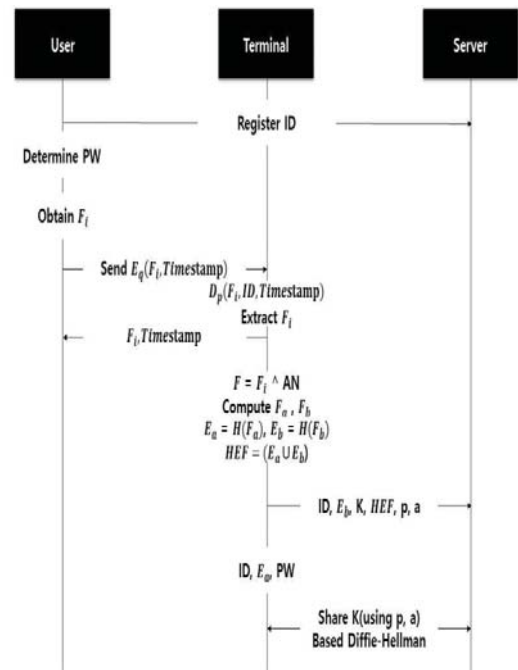
Attacker는 Step 1.~Step 4.과정에서 확보한 파라미터를 이용하여 서버와의 인증을 수행한다. 그 결과, 클라우드 자원에 접근 가능하다.

III. 제안 사항

3.1 개선된 인증 방안

2.2.3절에서 제시한 취약점에 대응하기 위한 개선된 인증 프로토콜은 (그림 3)과 같다.

본 절에서 제안하는 프로토콜은 기존에 생성된 난수 p 를 활용한 공개키 암호화를 이용해 Terminal 인증을 수행하고 지문 정보 재전송 공격 방지를 위한 Timestamp를 활용한다.



(그림 3) Proposed Protocol

3.1.1 제안 방안에서 사용되는 파라미터

본 절에서 제안하는 프로토콜에서 추가로 사용되는 파라미터는 (표 2)와 같다.

(표 2) Proposed Protocol Parameter

Parameter	Contents
q	- p 과 함께 사용할 공개키
Timestamp	- 재전송 공격을 방지하기 위한 값

3.1.2 제안 프로토콜 동작 순서

Step 1. 비밀 값 생성

프로토콜에 사용될 파라미터(a, p, K, AN)를 생성한다.

Step 2. ID 등록(User→Server)

User는 Server에게 본인의 ID를 등록한다.

Step 3. PW 결정(User)

User는 인증에 사용할 PW를 설정하고 스마트카드에 저장한다.

Step 4. 지문 정보 확보 및 암호화 전송

(User→Terminal)

User는 본인의 지문 정보를 확보하여 확보한 지문 정보를 User의 Timestamp와 함께 가지고 있는 난수 q 를 이용해 암호화하여 보낸다.

Step 5. Terminal 인증(Terminal→User)

Terminal은 Step 4.에서 수신한 값을 난수 p 를 이용해 복호화하여, User에게 보낸다. 또한 확보한 지문 정보로부터 Orientation Maps(OM)을 활용하여, 지문 값을 추출한다.

Step 6. Terminal 검증(User)

User는 수신한 값들을 이용해 Terminal이 정당한지 검증한다.

이후의 동작 순서는 2.2.2절의 Step 5.~Step 8.와 동일하다.

IV. 안전성 분석

본 논문에서 제시한 프로토콜에 대하여 sykipot 맬웨어 기반 프로토콜 취약점 공격을 수행한 과정은 아래와 같다.

Step 1.~Step 4.과정은 3.1.2절과 동일하다.

Step 5. Terminal 인증(Terminal→User)

이 때 기존의 Terminal은 복호화가 불가능하기 때문에 User의 Terminal 인증을 수행할 수 없다. 따라서, sykipot 맬웨어 취약점을 이용한 공격을 방지할 수 있다.

또한, Terminal 인증을 통과하더라도 공격자는 등록 당시의 지문 정보를 가지고 있기 때문에 Timestamp와 함께 전송함으로써 지문 정보의 재전송을 방지할 수 있다.

V. 결론

본 논문을 통해 Rohan등이 제시한 클라우드 환경에서의 스마트카드를 이용한 인증 프로토콜을 분석하고, 스마트카드 대상으로 동작하는 맬웨어를 분석하였다. 또한, 해당 맬웨어를 이용한 공격 시나리오를 제시하고 기존 프로토콜의 취약점 개선을 위한 인증 프로토콜을 제시하였다.

개선된 인증 프로토콜은 Terminal에 대한 인증 없이 인증 프로토콜을 수행하는 Rohan등의 인증 프로토콜에 추가적으로 난수 q 와 Timestamp를 활용하여 sykipot 맬웨어와 재전송 공격을 방지하고, Terminal에 대한 인증 과정을 추가하였다.

향후 클라우드 환경에서의 경량화 인증 프로토콜에 대해 연구를 수행할 예정이다.

[참고문헌]

- [1] Sarang V. Hatwar, "Cloud Computing Security Aspects, Vulnerabilities and Countermeasures", Jun. 2015
- [2] Rohan H. Shah, Prof. D. P. Salapurkar, "A multifactor authentication system using secret splitting in the perspective of Cloud of Things", Feb. 2017
- [3] SANS Institute InfoSec Reading Room, "Detailed Analysis Of Sykipot"