Journal of Knowledge Information Technology and Systems(JKITS), Vol. 9, No. 6, pp. 707~714, December 2014

Journal of Knowledge Information Technology and Systems

ISSN 1975-7700

http://www.kkits.or.kr

Encryption of TCP Sequence Numbers for Session Hijacking Attacks

Jae-Yeong Choi[1], Hyun-Chul Baek[2], Sang-Bok Kim[1], Jong-Chae Sim[2], Jae-Heung Park[*1]

[1]*Department of Computer Science, Gyeongsang National University*

[2]*Department of Intenet Information Technology, Gyeongnam Provincial Namhae College*

ABSTRACT

Session hijacking attacks represent a type of attack that captures existing session information by terminating normal sessions, which are connected through a TCP 3-Way handshaking process, using RST signals. In the case of the TCP 3-Way handshaking process, different sequence signals between servers and clients are generated to carry out a cross certification for each other. In a normally connected session, attackers interrupt sessions for a normal connection between servers and clients through generating RST signals and attempt to access a system by disguising an attacker as a registered user by generating a new sequence number. In precedent studies on preventing session hijacking attacks, a method that encrypts all sequence numbers generated in accessing a system has proposed. However, this method may cause lots of overheads in a cross certification process because it encrypts all sequence numbers required for the cross certification between servers and clients. Thus, in this study the encryption process is performed using a part of the information of the cross certification numbers in order to prevent illegal session hijacking as abnormal RST signals are generated. In addition, this method prevents the recognition of sequence numbers from attackers even though a sequence number is leaked. Moreover, it is possible to perform a defense for session hijacking attacks and to partly solve the overheads, which have been issued in the conventional studies.

KEYWORDS: TCP 3-Way, Session Hijacking, Data Encryption Standard, TCP/IP Sockets, Sniffing

ARTICLE INFO: Received 29 October 2014, Revised 12 December 2014, Accepted 12 December 2014.

1. 서 론

오늘날 계속적으로 확장되고 있는 네트워크 상황은 사용자들에 대한 서비스 측면에서 획기적인

[*]Corresponding author is with the Department of Computer Science, Gyeongsang National University, 501, Jinju-daero, Jinju-si, Gyeongsangnam-do, 660-701, KOREA.

E-mail address: pjh@gnu.ac.kr

발전을 해 오고 있다. 그렇지만 이러한 네트워크의 발전은 네트워크를 이용한 다양한 공격 행위를 유발시키고 있다. 특히 TCP는 그 구조상 불법적인 공격 행위에 매우 취약한 것이 현실이다. 이런 문제로 인하여 TCP 순서번호 암호화 기법[1]이 제안되었지만 이는 세션 연결과정에 필요한 모든 시퀀스 번호를 암호화시키기 때문에 연결과정에 오버헤드가 발생 할 수 있다. 일반적인 네트워크 환경에서 서버와 클라이언트의 연결 과정은 TCP 3-Way 핸드셰이킹 과정을 거치게 된다[2]. 이 과정에서 서버와 클라이언트 간에는 상호 인증을 위한 시퀀스 번호를 발생시킨다. 정상적인 연결 상태에서 RST 신호는 상호 연결 중 ACK 신호 등 정상적인 신호가 접수되지 않았을 경우 발생시켜 새로운 연결에 대한 재설정 과정을 거치게 한다. 그러므로 공격자는 이러한 시퀀스 번호를 획득한 후 RST 신호를 전송하여 기존의 정상적인 접속자로 위장을 하여 접속을 시도하는 것이다. 이렇게 세션하이재킹 공격이 성공하게 되면 공격자는 이를 통하여 정상적인 사용자들의 중요한 정보를 획득하게 되는 것이다. 그러므로 본 논문에서는 새로운 재설정 요구가 발생하게 되면 시퀀스 번호 암호화[3]를 통하여 기존의 정상적인 사용자의 재설정 요구인지, 공격자에 의한 재설정 요구인지를 결정 할 수 있도록 하였다. 본 논문에서는 먼저 TCP 3-Way 핸드셰이킹 과정을 분석하고 이와 관련한 시퀀스 번호 교환 과정을 살펴본다. 그 다음 세션 하이재킹 공격에 대해 분석하고, 시퀀스 번호 암호화 과정과 시뮬레이션을 통하여 그 결과를 보도록 한다.

2. 관련연구

2.1 TCP 3-Way 핸드셰이킹 과정

TCP 3-Way 핸드셰이킹 과정은 다음 <그림 1>과 같다.

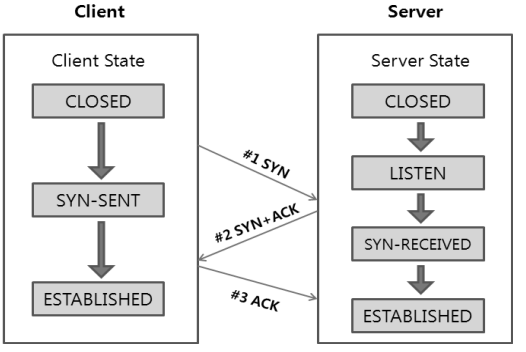


그림 1. TCP 3-Way 연결 설정 과정
Figure 1. TCP 3-Way Connection Establishment Process

<그림 1>에서 보는 바와 같이 클라이언트는 서버와의 세션 확보를 위하여 SYN 신호를 서버로 전송한다. 해당 SYN 신호를 수신한 서버는 다시 이에 대한 응답으로 클라이언트에서 전송되어 온 SYN 신호에 자신의 ACK 신호를 더하여 클라이언트로 응답 신호를 전송한다. 그 다음 이 신호를 수신한 클라이언트는 이에 대한 응답으로 자신의 ACK 신호를 서버로 전송하게 되고, 서버와 클라이언트는 세션 연결 상태를 확보하게 되는 것이다. 즉 클라이언트와 서버 간의 통신을 위하여 3 단계의 메시지 송/수신 과정을 거치게 되는 것을 TCP 3-Way 핸드셰이킹이라고 한다.

2.2 세션 하이재킹 공격

인터넷에 연결된 서버들은 공격자들에게 항상 노출되어 있다. 공격자들의 다양한 공격 기법들 중 정상적인 사용자를 가장하여 내부 보안망을 무력화 시킬 수 있는 공격 기법에 세션 하이재킹 공격[4]이 존재한다. 세션 하이재킹(Session Hijacking)이란 문자 그대로 '세션 가로채기'를 의미한다. 여

기서 세션이란 사용자와 컴퓨터, 또는 두 대의 컴퓨터간의 연결 활성화 상태를 말한다.

TCP 세션 하이재킹 공격은 정상적인 사용자로 위장을 한 후 공격을 시도하는 부분에서 IP 스푸핑과 비슷하다고 할 수 있다[5]. 하지만 IP 스푸핑의 경우는 상호 트러스트 정보를 이용한 공격을 시도하고, TCP 세션 하이재킹 공격은 활성화 되어 있는 세션을 RST 신호를 이용하여 강제로 빼앗아 가는 부분에 그 차이가 있다.

아울러 TCP 3-Way 핸드셰이킹 과정에서 발생하는 신호들은 시퀀스 번호를 통하여 상호 인증을 하고 있다. 즉, TCP 세션 하이재킹 공격은 서버와 클라이언트가 상호 세션 연결을 시도할 때 발생하는 시퀀스 번호를 가로챈 다음 그 정보를 가지고 공격자 자신이 정상적인 클라이언트로 위장하여 연결을 시도하는 것이다. 다음은 일반적인 세션 하이재킹 공격 과정이다.

1. 공격자는 스니핑을 통하여 상호 연결에 필요한 시퀀스 번호를 획득한다.

2. 공격자는 RST 신호를 서버로 전송하여 세션 연결 상태를 단절시킨다. 즉, 서버는 잠깐 동안 Closed 상태가 되고, 클라이언트는 그대로 Established 상태로 남는다.

3. 공격자는 재생성 된 시퀀스 번호를 서버로 전송한다.

4. 서버는 공격자가 재생성하여 전송한 시퀀스 번호를 수신한 후 연결 재설정을 시도한다.

5. 공격자는 정상적인 연결처럼 서버와 시퀀스 번호를 교환하고, 공격자와 서버 모두가 Established 상태가 된다.

위에서 서술하고 있는 비정상적인 접속자들의 공격으로 부터 정상적인 세션 연결 상태 확보를 위하여 본 논문에서는 암호화 기법을 제안하였다. 즉, 공격자가 스니핑을 통하여 세션 연결에 필요한 시퀀스 번호를 획득 하더라도 해당 시퀀스 번호를 알지 못하면 RST 신호를 이용한 연결 재설정을 할 수 없도록 하였다[6].

3. 시퀀스 번호 암호화 과정

본 논문은 RAW Socket을 사용하여 TCP 세션 확립에 필요한 플래그를 가진 신호를 생성하여 프로그램을 리눅스 운영체제 상에서 직접적으로 구현하였다[7][8][9].

3-Way 핸드셰이킹 과정은 서버와 클라이언트간 상호 인증을 위하여 자신들의 시퀀스 번호를 발생시켜 상호 인증 과정을 거친다. 그러므로 공격자는 이 시퀀스 번호를 획득하여 자신이 정상적인 클라이언트로 위장하여 연결 재설정을 시도하는 것이다. 본 논문에서는 초기 연결 설정 과정에서 발생하는 시퀀스 번호를 클라이언트 자신의 암호화 버퍼에 저장하고, 서버 또한 초기 연결 설정 과정에서 클라이언트로부터 전송되어온 시퀀스 번호를 자신의 암호화 버퍼에 저장한다. 그 다음 연결 재설정 요구 신호가 수신되면 각각의 암호화 버퍼에 있는 시퀀스 번호에 클라이언트와 서버간 미리 약속 된 난수를 이용하여 암호화된 새로운 시퀀스 번호를 생성한다. 암호화된 시퀀스 번호는 연결 재설정 신호에 대한 상호 인증을 위하여 사용한다. 이에 대한 구체적인 과정은 그림 2와 같다. 먼저 Client_My_Seq에 미리 클라이언트와 서버가 약속한 정상적인 시퀀스 번호에 대한 변화를 주기 위하여 임의의 값을 곱한다. 그 다음 생성된 값에서 임의 의 자리수를 정한 후 암호화키 CKey로 사용하였으며, 본 논문에서는 3번째부터 7번째 자리 값을 추출하여 암호화키로 이용하였다. 복화키는 서버측에서도 동일한 수행 과정을 거친다[10][11][12].

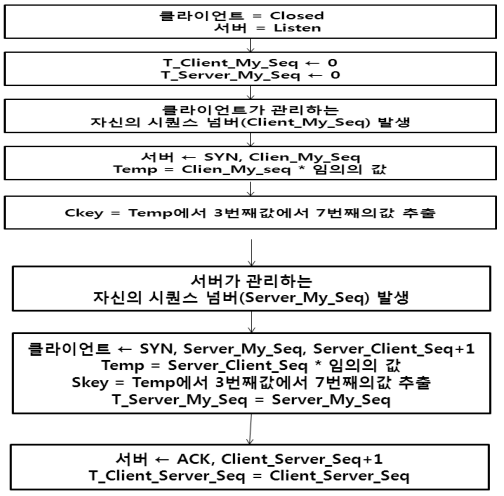


그림 2. 제안 암호화 키 생성 과정
Figure 2. Proposed Cryptographic Key Generation Process

서버에서 생성한 첫 시퀀스 번호 (Server_My_Seq)는 서버의 T_Server_My_Seq, 클라이언트에는 T_Client_Server_Seq에 각각 보관하여 RST 신호가 발생했을 때 상호 인증을 위한 암/복호화 정보로 이용한다.

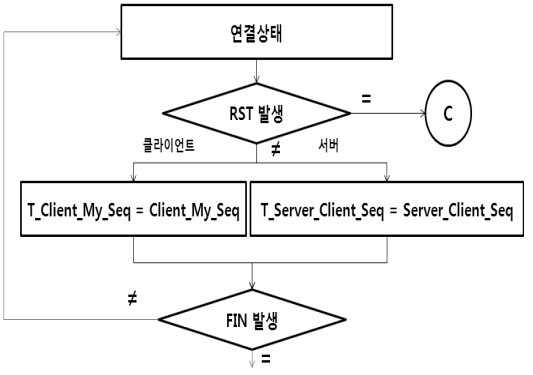


그림 3. 암호화 키 보관 단계
Figure 3. Encryption Key Storage Step

다음으로 3-Way 핸드셰이킹 과정을 거쳐 정상적인 세션 연결 상태에서는 <그림 3>과 같은 동작을 한다. 클라이언트는 자신이 관리하는 시퀀스 번호(Client_My_Seq)를 T_Client_My_Seq에 보관하고, 서버는 자신이 알고 있는 클라이언트의 시퀀스 번호(Server_Client_Seq)를T_Server_Client_Seq에 보관한다.

<그림 4>는 RST 신호가 발생했을 경우 클라이언트와 서버의 대응 과정을 설명하고 있다.

RST 신호가 발생하면 클라이언트와 서버는 TCP 3-Way 핸드셰이킹 과정을 다시 수행하게 된다. 이때 클라이언트는 이전의 연결 설정 과정에서 처음 생성되었던 T_Client_Server_Seq 값과 RST 신호가 발생하기 직전의 Client_My_Seq 번호의 합을 구한 후 DES 암호화 기법을 이용하여 해당 시퀀스 번호를 암호화 시킨다. 이와 함께 서버에서는 T_Server_My_Seq 값과 T_Server_Client_Seq 값에 대한 합을 구한 후 Temp에 보관한다. 이 과정에서 클라이언트는 암호화 된 결과를 서버로 전송하고, 서버는 SKey를 이용하여 해당 암호문을 복호화 시킨다. 복호화 된 값은 Temp의 값과 상호 비교하여 같은 경우에만 연결을 유지하고 상이할 경우 접속을 끊는다.

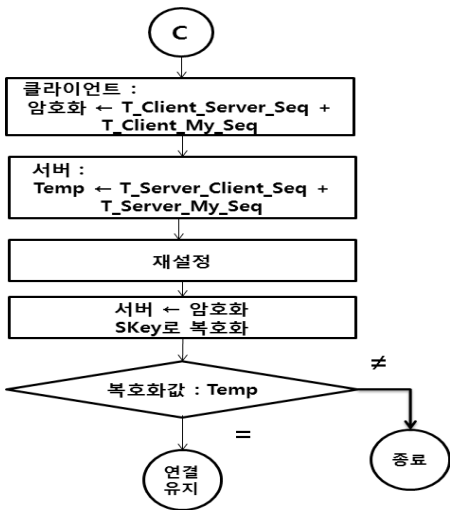


그림 4. RST 신호 발생 후 재설정 및 암호화 단계
Figure 4. Reset And Encryption Step after RST Signal Generation

<그림 5>는 FIN 신호가 발생했을 경우 정상적으로 종료되는 과정을 보여주고 있다.

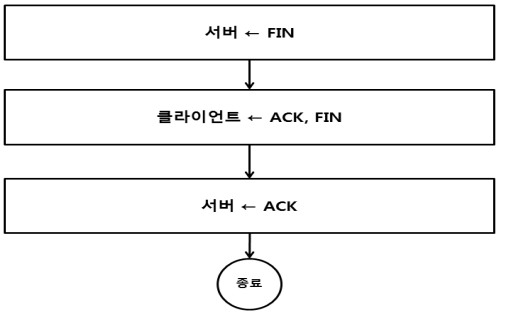


그림 5. 정상적인 종료 단계
Figure 5. Normal Termination Step

4. 시뮬레이션 및 결과

본 논문은 다음과 같은 실험 과정을 거쳤다. 먼저 네트워크 환경에서 NIC(Network Interface Card) 정보 확인과 선택에 필요한 패킷 수집을 위하여 와이어샤크 프로그램을 사용하였다. 다음 <그림 6>은 정상적인 TCP 3-Way 핸드셰이킹 연결 과정과 자료 전달 과정 및 정상적인 종료 단계를 나타내는 그림이다.

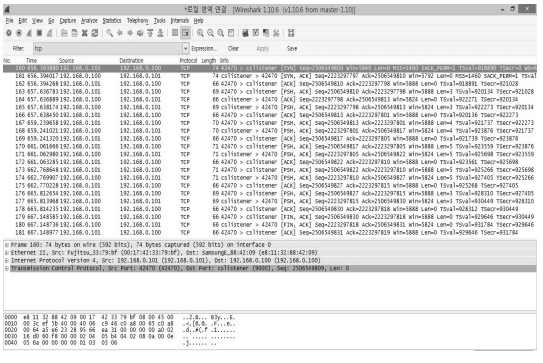


그림 6. 정상적인 세션 연결 과정 및 종료 단계
Figure 6. Normal Session Connection process and Termination Step

<그림 7>은 클라이언트에서 RST 신호가 발생하여 기존의 정상적인 접속자일 경우 상호 인증을 거친 다음 재설정이 완료된 상태를 보여주는 그림이다.

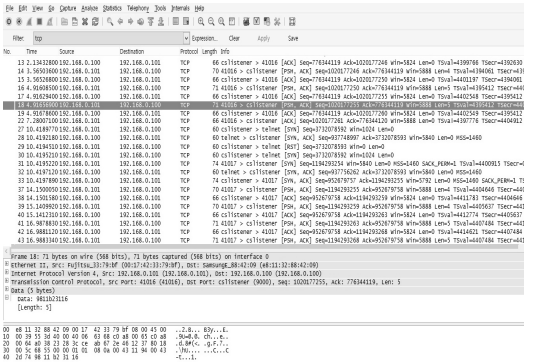


그림 7. 정상적인 RST 신호 발생 후 암/복호화 키를 이용한 재설정 완료 상태
Figure 7. The State of Reset Completion Using Encryption Process/Decoding Process after Normal RST Signal Generation

<그림 8>은 공격자가 RST 신호를 발생시켜 세션 하이재킹 공격을 시도하였지만, 상호 인증 단계를 거치면서 정상적인 접속자가 아닌 것으로 판정받아 연결 종료가 일어난 상태를 보여준다.

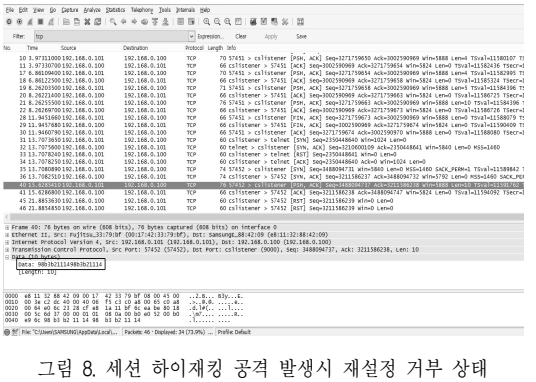


그림 8. 세션 하이재킹 공격 발생시 재설정 거부 상태
Figure 8. Reset Occurs, The Session Hijacking Attack Denied State

<그림 9>는 RST 신호가 발생한 후 클라이언트에서 암호화를 시킨 값을 2진수로 보여주고 있다. 이

암호화된 값은 서버에 전송하여 서버에서 <그림 10>과 같이 복호화 했을 경우 원래의 값으로 출력된 결과이다.

```
Encryption :
10111001
10011011
00011011
00011010
00110001
10010110
00011100
00011011
```

그림 9. 클라이언트 암호화 과정 결과
Figure 9. Client Encryption Process Result

```
Decryption :
00110010
00110110
00110011
00110100
00110011
00111001
00111000
00110110

26343986
```

그림 10. 서버 복호화 과정 결과
Figure 10. Server Decoding Process Result

5. 결 론

본 논문에서 제안하고 있는 시퀀스 번호 암호화는 TCP 3-way 핸드셰이킹 과정에서 발생하는 시퀀스 번호 중 최초 발생한 클라이언트의 시퀀스 번호(Client_My_Seq)와 서버에 저장되어 있는 클라이언트의 시퀀스 번호(Server_Client_Seq)를 암/복호화 키로 이용하고 있다. 아울러 서버 자신의 시퀀스 번호(Server_My_Seq), 클라이언트에서 가지는 서버의 시퀀스 번호(Client_Server_Seq)도 각각 보관한다. 이를 통하여 연결 설정 후 RST가 발생하면 임시 보관된 시퀀스 넘버를 암/복호화 키로 이용하여 상호 인증이 가능하도록 하였다. 이러한 시퀀스 번호에 대한 암호화는 세션하이재킹 공격에 무방비 상태로 노출되어 있는 시퀀스 번호의 안정적인 관리를 가져 올 수 있다. 아울러 기존 논문에서 제안하고 있는 세션 연결을 위하여 생성되는 모든 시퀀스 번호에 대한 암호화 과정의 오버헤드 부분도 본 논문에서는 초기의 시퀀스 번호만 이용하기 때문에 감소시킬 수 있다고 본다. 그리고 이를 기반으로 세션하이재킹 공격에 대해서도 안정적인 네트워크 서비스를 보장 할 수 있다고 본다. 향후 연구과제로는 시퀀스 번호에 대한 암/복호화 과정을 통한 상호 인증 부분이 전체 연결 성능에 어느 정도의 영향을 주는지에 대한 연구가 함께 필요할 것으로 본다.

References

[1] D.-H. Seo, H.-J. Kang, *Suggest of TCP sequence number Encryption*, Journal of Korea Multimedia Society, Autumn Annual Conference, pp. 498~501, 2000.

[2] B.-K. Kim, I.-K. Kim, D.-W. Kim, and J.-T. Oh, J.-S. Jang, T.-M. Chung, *Implementation of high performance TCP proxy logic against TCP flooding attack on network interface card*, Journal of the Korea Institute of Information Security and Crypto logy, Vol. 21, No. 2. pp. 119~129, 2011.

[3] Yadav, Dinesh, and Anjali Sardana. *Enhanced 3-way handshake protocol for key exchange in IEEE 802. 11i,* Electronics Computer Technology, Vol. 6, pp. 132~135, 2011.

[4] Nishanth, N., J. Zareena, and S. Suresh Babu. *Pseudo random alteration of sequence*

numbers (PRAS): A novel method for defending sessiion hijacking attack in mobile adhoc network, Communication Technology, 2013 15th IEEE International Conference on. IEEE, pp. 20~25, 2013.

[5] P.-H. Jo, J.-I. Lim, and H.-K. Kim, *A study on the improvement of security vulnerabilities in intelligent transport systems*, Journal of The Korea Institute of Information Security & Cryptology, Vol. 23, No. 3, pp. 531~543, 2013.

[6] J.-J. Lee, S.-J. Kim, and D.-J. Kang, *A development of cipher device based on embedded linux for serial communication in SCADA*, Journal of the Korean Institute of IIIuminating and Electrical Installation Engineers, Vol. 24, No. 4, pp. 25～32, 2010.

[7] W.-J. Lee, S.-J. Jeong, and Y.-W. Kim, S.-Y. Heo, *Design and implementation of network proxy protocols for energy saving of network terminal*, Journal of KIIT, Vol. 12, No. 2, pp. 111~121, 2014.

[8] D.-J. Kang, H.-J. Park, *A design and implementation of transmit/receive model to speed up the transmission of large string-data sets in TCP/IP socket communication*, Journal of the Korea Institute of Information and Communication Engineering, Vol. 17, No. 4. pp. 885~892, 2013.

[9] S.-K. Kim, K.-S. Chung, *A performance improvement of linux TCP networking by* d*ata structure reuse*, KIPS Tr. Comp. and Comm. Sys, Vol. 3, No. 8. pp. 261~270, 2014.

[10] D.-S. Choi, D.-H. Oh, and J.-S. Park, J.-C. Ha, *An improved round reduction attack on triple DES using fault injection in loop statement*, Journal of The Korea Institute of Information Security & Crypto logy, Vol. 22, No. 4. pp. 709~717, 2012.

[11] Y.-H. Chung, *A spread random interleaver based efficient DES algorithm for personal* c*loud computing environments*, Journal of the Korea Institute of Information and Communication Engineering, Vol. 17, No. 1. pp 41~48, 2013.

[12] K.-H. Song, H.-C. Kang, and J.-C. Sung, *An efficient new format-preserving encryption algorithm to encrypt the personal information*, Journal of The Korea Institute of Information Security & Crypto logy, Vol. 24, No. 4. pp. 753~763, 2014.

세션 하이재킹 공격에 대한 TCP Sequence Number 암호화

최재영[1], 백현철[2], 김상복[1], 심종채[2], 박재홍[1]

[1]*경상대학교 컴퓨터과학과*

[2]*경남도립남해대학 인터넷정보기술과*

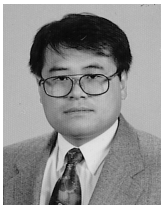
요 약

세션 하이재킹 공격이란 TCP 3-Way 핸드셰이킹을 통하여 연결된 세션을 RST 신호를 이용하여 정상적인 세션 연결을 끊고 기존 세션 정보를 탈취해 가는 공격을 의미한다. 즉 TCP 3-Way 핸드셰이킹 수행과정에서 서버와 클라이언트 간에는 상호 인증을 위한 시퀀스 번호가 발생하게 된다. 이렇게 정상적인 세션 연결 상황에서 공격자는 RST 신호를 발생시켜 서버와 클라이언트 간의 정상적인 연결에 대한 세션을 끊고 새로운 시퀀스 번호를 생성하여 공격자 자신이 기존 접속자로 위장하여 접속을 시도하는 것이다. 이러한 세션 하이재킹 공격에 대비한 기존 연구로는 연결과정 중에 발생하는 모든 시퀀스 번호를 암호화 시키는 기법이 제안되었다. 그렇지만 이는 서버와 클라이언트 간의 상호인증 과정에 필요한 모든 시퀀스 번호를 암호화 시키고 있기 때문에 상호인증 단계에서 많은 오버헤드를 초래할 수 있다. 그러므로 본 논문에서는 비정상적인 RST 신호가 발생했을 때, 불법적인 세션 가로채기를 방지하기 위하여 상호인증 시퀀스 번호 중 일부 정보만을 이용하여 암호화 과정을 수행시켰다.

그리고 이를 이용하여 해당 시퀀스 번호가 유출되었다 하더라도 공격자가 알아 볼 수 없도록 하였다. 아울러 세션 하이재킹 공격에 대한 방어와 기존 연구에서 문제가 되고 있는 오버헤드 부분도 일부 해결할 수 있었다.

Jae Yeong Choi received the Master´s degree in the Department of Computer Science from Gyeongsang National University in 2014. His current research interests include network architecture, network security.

E-mail address: jyoungc67@naver.com



Hyun Chul Baek received the Ph.D. degree in the Department of Computer Science from Gyeongsang National University in 2003. He was a chairman in the Committee of Computer System technology at The Korea Association of Regional Public Hospital in 2007. He has been a professor in the Department of Intenet Information Technology, Gyeongnam Provincial Namhae College since 2013. His current research interests include network, network security, encryption, bigdata security, cloud computing. He is a member of the KKITS.

E-mail address: dosi_gas@lycos.co.kr

Sang Bok Kim received the Ph.D. degree in the Department of Electronics Engineering from Chung-ang University in 1989. He was a director in the Department of Education Information Computer Center at The Gyeongsang National University from 2007 to 2010. He has been a professor in the Department of Computer Science at Gyeongsang National University since 1984. He has been a researcher in the Computer Data Communication Research Institute at The Gyeongsang National University since 1984. His current research interests include computer network and security, computer system architecture. He is a member of the KKITS.

E-mail address: sbkim@gnu.kr

Jong-Chae Sim received the Ph.D. degree in the Department of Computer Science from Gyeongsang National University in 2003. He has been a professor in the Department of Intenet Information Technology, Gyeongnam Provincial Namhae College since 1998. His current research interests include Information system and dp.

E-mail address: simjc@namhae.ac.kr

Jae Heung Park received the Ph.D. degree in the Department of Computer Engineering from Chung-ang University in 1989. He has been a professor in the Department of Computer Science at Gyeongsang National University since 1983. He has been a researcher in the Software Engineering Research Institute at The Gyeongsang National University since 1984. His current research interests include computer network and security, S/W Reliability. He is a member of the KKITS.

E-mail address: pjh@gnu.ac.kr