Journal of The Korea Institute of Information Security & Cryptology ISSN 1598-3986(Print)
VOL.28, NO.1, Feb. 2018 ISSN 2288-2715(Online)
https://doi.org/10.13089/JKIISC.2018.28.1.135

Threat Modeling을 이용한 PS4와 PC간의 Remote Play 상황 속 위험 분석

김 혜 민,† 김 휘 강‡
고려대학교 정보보호대학원

Threat Modeling and Risk Analysis: PS4 Remote Play with PC

Hye Min Kim,† Huy Kang Kim‡
Graduate School of Information Security, Korea University

요 약

최근 소니 사에서 PS4(PlayStation4)와 PC 간의 인터넷 연결을 통한 리모트 플레이 서비스를 런칭하였다. 이 서비스는 외부 네트워크와 PS4가 설치된 환경의 네트워크 연결을 가능하게 하였다. 새로운 서비스로 인해 리모트 환경에서 추가적인 보안 위협이 발생할 수 있으며 이를 분석하고 그에 대한 대안을 마련해야 한다. 본 논문에서는 위협 모델링 기법을 이용해 새로이 나타나는 보안 위협을 파악하고 도출한 위협에 대해 비용대비 분석, 유용성 분석을 진행하여 합리적인 보안 대책을 세울 것이다.

ABSTRACT

Sony has recently launched a remote play service that connects PC and PlayStation4 using the Internet. This service enables the network connection between the external network and PS4 network. After the service released, additional security threats may arise in remote environments with new services. Therefore, those threats should have been analyzed. In this paper, as applying threat modeling to remote play system, threats have been analyzed and identified. After cost-effective and usability analysis, finally, reasonable security measure of each threat has been suggested.

Keywords: Remote Play, PS4, Internet, Threat Modeling

I. 서 론

Sony 사는 전 세계 콘솔 게임 시장에서 가장 큰 점유율을 가지고 있다. 최근 PS4(PlayStation4) slim, pro 버전을 출시하면서 더욱 더 판매에 박차를 가하고 있다. 2016년, 펌웨어 3.50버전을 공개함으로써 PC에서 PS4의 리모트 플레이가 가능해졌다. 리모트 플레이는 PS4와 PC가 인터넷에 연결이 되어있으면 둘의 인터넷 환경이 동일하지 않아도 가능하다. 사용자는 PSN(PlayStation Network)에서 제공하는 리모트 플레이 프로그램을 PC에 다운받아 이를 실행한다. Micro 5 pin USB선을 이용하여 PC에 PS4 컨트롤러를 연결하고 사용자의 PSN 계정으로 로그인하면, PS4의 실행화면이 인터넷을 통해 PC에 전송 된다. 이 과정에서 PS4는 외부 네트워크 환경과 연결이 되어 새로운 보안 위협이 발생할 가능성이 생기게 된다. 본 논문에서는 리모트 플레이 환경에서 발생할 수 있는 PS4로의 무단 접근을 막기 위한 여러 가지 보안대책에 대해 비용 대비 효율성과 유용성 두 가지 관점에서 분석을 진행하여 알맞은 보안대책을 제시해볼 것이다.

새로운 위협을 도출하기 위해서는 기존 보안 위협

Received(10. 11. 2017), Modified(12. 13. 2017), Accepted(12. 15. 2017)
† 주저자, hm941224@korea.ac.kr
‡ 교신저자, cenda@korea.ac.kr(Corresponding author)

모델의 추가적인 보안 위협 모델링이 필요하다. 실제로도 콘솔게임 기기에 대한 공격이 일어나고 있으며 이를 막기 위한 보안 대책 연구가 이루어지고 있다. 본 논문에서는 PC와 PS4간의 리모트 환경에서 일어날 수 있는 위협에 대한 분석을 통해 공격에 대처할 수 있는 체계적인 방법론을 제시할 것이다.

II. 관련 연구

최근 IoT(Internet of Things)와 CPS(Cyber Physical System)와 같은 개념이 일상생활에 도입됨에 따라 보안 목표를 실현하기 위한 확실하고 입증된 위협모델링의 중요성이 부각되고 있으며 다양한 분야에서 위협모델링에 대해 연구가 진행되고 있다. 게임 분야에서도 보안 위협이 발생하고 있으며 그에 대한 사례와 대책에 대한 연구가 이루어지고 있다.

2.1 위협모델링 연구

시스템이 안전하다는 결론을 도출하기 위한 가장 중요한 사항은 '모델링한 보안대책들이 보안목표를 충족시키는 것'이다. Myagmar 외 2인은 위협모델링을 위해 복잡한 시스템에 대한 높은 이해도와 시스템을 목표로 한 모든 가능한 취약점에 대한 체계적인 과정 속에서 분석 방법을 연구하였다[1]. 이 때, 취약점 도출을 위해 공격자가 목표로 하는 자산에 대한 식별이 매우 중요하다.

본 논문에서는 Microsoft 사에서 개발한 STRIDE를 이용하여 위협모델링을 진행한다. STRIDE는 보안 목표를 공격할 수 있는 방법에 대해 그룹화를 한 것이다. STRIDE에 대한 자세한 사항은 Table 1.에서 확인할 수 있다. Spoofing은 목표 자산에 접근 권한을 얻기 위해서 공격자가 관리자나 사용자인 체 하는 것을 말하며 Tampering은 데이터저장소의 데이터를 바꾸거나 데이터 흐름을 갈취하여 원하는 데이터를 수신자에게 전송하거나 데이터를 위조하는 것을 말한다. Repudiation은 해당 프로세스가 실행되었을 때의 책임이나 자신이 했다는 증거를 부인하는 것을 말한다. Information disclosure는 공격자가 원하는 정보에 대한 열람 권한이 없어도 열람이 가능한 것을 말한다. 공격자의 접근 권한이 허락되지 않는 곳에서 접근이 허락되는 곳으로 데이터가 전달이 되었을 때, 공격자는 해당 데이터를 열람할 수 있다. Denial of service는 모든 리소스를 공격자가 차지하여 정상 유저가 다른 프로그램을 사용하지 못하게 하는 것이다. Elevation of privilege는 권한이 없는 공격자가 프로세스를 실행할 수 있는 권한을 얻을 수 있게 하는 것이다. 이를 통해 권한이 없는 프로세스를 실행시키거나 정보를 열람할 수 있는 권한을 획득할 수 있다.

Table 1. STRIDE

attack	description
Spoofing	pretending someone to get access to target assets
Tampering	changing data to deceive the user for attack
Repudiation	happen when a user denies performing an action, but there is no way to prove
Information disclosure	someone that has no permission to data can see important information
Denial of service	take resources to interrupt valid user's request
Elevation of privilege	happen when an unprivileged user gain privileged status

위협모델링 과정 중에서 공격 시나리오에 대한 이해도를 높이고 각 노드에 대한 위험도 계산을 위한 도구로써 attack tree 기법을 사용한다. attack tree는 다양한 공격 시나리오에 대해 정형화되고 수학적인 방법으로 묘사해주기 때문에 정성적인 접근이 가능하다. Saini 외 3인이 연구한 attack tree 기법의 순서는 다음과 같다[2]. 먼저, 전반적인 공격 목표를 나열하고 각 목표에 대한 하위 공격 목표들을 OR 연산과 AND 연산으로 이어주며 연결한다. 각 노드별 위험도를 알게 되면, 공격자가 공격을 성공했을 때 얻는 이익과 공격을 할 때 드는 비용을 비교하여 가지를 치는 과정을 거쳐야 한다. 가지치기가 끝난 attack tree를 통해 위협에 대한 보안대책을 도출한다.

Johansson 외 1인의 연구에서는 네트워크 환경에서 위협을 분석하였고 대책을 세우기 위해서는 네트워크 환경과 기본 네트워크 설정, 프로토콜 설정에 대한 이해의 필요성을 주장하였다[3]. 이 논문에서는 네트워크의 시스템 구조를 이해하고 이에 대한 DFD를 통해 네트워크 환경 내 위협을 도출한다.

Dekker 외 2인은 스마트폰에 다운 받을 수 있는 앱을 배포하는 앱스토어의 위협모델링에 대한 연구를 진행하였다. 앱 시스템의 DFD(Data Flow Diagr

am)를 통해 악성 앱을 이용한 위협에 대해 공격 시나리오들을 도출하고 분석하였다[4].

2.2 게임에 대한 보안 이슈 연구

Davies 외 2인은 PS4에 대해서 범죄 수사적인 관점에서 연구를 하였다. PS4의 하드 디스크, 에러 히스토리, 메시지, 저장 관리 시스템 등 디지털 범죄 수사관에게 분석해야 할 가이드라인을 제공한다. 공격할 수 있는 공격 벡터에 대한 상세한 실험과 분석이 진행되었고 PS4의 펌웨어 버전에 따라 얻을 수 있는 정보에 대한 분석이 진행되었다[5]. 본 논문에서는 해당 논문과 다른 관점인 보안공학의 관점에서 위협모델링을 통하여 그에 대한 분석을 진행하려 한다.

Vaughan 외 1인은 PS4와 비슷한 콘솔게임 기기인 Xbox에서 Linux 운영체제를 실행하는 데에 성공했다[6]. 이로써 PC와 비슷한 CPU와 GPU를 가지고 있는 Xbox는 Linux 운영체제 기반의 PC와 같은 기능을 할 수 있게 되어 큰 보안 위협을 초래했다. 공격 비용보다 공격 성공 후 이익이 더 크기 때문에 많은 사람들이 공격을 시도하게 되었다. 공격에 성공한 공격자들은 Xbox 하드 디스크 안에 불법 소프트웨어를 다운 받을 수 있게 되었다. 근래 PS4에도 Linux 운영체제를 실행시킨 동영상이 인터넷에 존재한다. 이와 같이 PC와 같은 기능을 하는 콘솔게임 기기는 큰 위험을 초래할 수 있다.

Jeff 외 2인은 온라인 게임 내에서 발생할 수 있는 위협들과 일어난 위협에 대한 보안 대책에 대해서 연구를 진행하였다[7]. 그들은 네트워크 연결이 필요한 온라인 상태 때문에 다양한 보안 이슈들이 생겨난 것이라 결론을 맺었다.

Cone 외 3인의 연구에서는 비디오 게임에서도 보안 이슈에 대한 보안 훈련과 주의의 필요성을 강조하였다. 주로 유저는 콘솔게임 기기를 인터넷에 연결한 상태로 집에 놓아두는 상황이 많다. 사용자가 집에서 나와 장시간 기기와 떨어져 있으면 공격자에게 공격 기회가 많아진다. 보안 문제에 대해 많은 주의가 필요함에도 불구하고 그에 대한 예방법이 부족하다. 따라서 사용자와 개발자 모두에게 도움이 될 수 있는 보안 훈련을 진행하여 사용자의 프라이버시(privacy)나 콘솔게임 기기의 안전을 지킬 수 있도록 장려한다[8].

III. 위협 모델링 분석 및 결과

본 논문에서는 PC와 PS4와의 리모트 플레이 서비스 내 보안상의 요구사항에 대해 분석한다. 이 모든 과정은 보안공학 관점의 보안위협모델링을 거쳐 진행한다.

보안위협모델링의 순서는 다음과 같은 순서로 진행한다. 위협모델링을 진행할 시스템 경계와 범위를 정한 후, 중요한 정보 자산을 식별한다. 이 때, 자산 식별과 위협 분석을 명확하게 명세하기 위해서 DFD(Data Flow Diagram)를 그린다. DFD를 참고하여 도출한 위협들은 STRIDE를 이용하여 그룹화 한다. 여러 위협을 결합시켜 공격자 관점에서 가능한 공격 시나리오를 모두 기술한다. 이 때, 공격 시나리오는 attack tree를 이용하여 시각화 한다. 도출된 위협에 대한 비용대비 분석, 유용성 분석을 통하여 합리적인 보안대책을 제시하고 보안대책의 안전성을 증명한다.

3.1 DFD기반 위협 분석

리모트 플레이 중 위협이 일어날 수 있는 상황을 판단하기 위해서는 시스템에 대한 이해도가 높아야한다. 따라서 데이터의 흐름을 통해 시스템의 구조를 이해할 수 있는 DFD를 작성한다. 본 논문에서 작성한 DFD는 리모트 플레이 연결 과정과 유저가 리모트 플레이를 이용하고 있는 과정을 범위로 한하여 작성한 DFD는 Fig. 1.과 같다. trust boundary를 기준으로 두 부분으로 나누어 시스템 분석을 진행한다. Fig. 1.의 좌측은 유저가 컨트롤러를 사용할 때

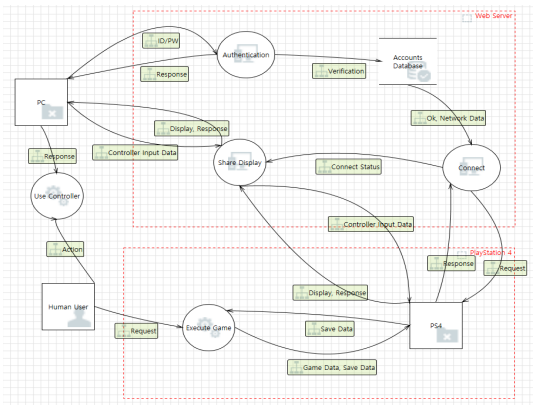

Fig. 1. DFD

발생하는 데이터 흐름이다. 유저는 컨트롤러를 이용하여 PC와 데이터를 주고받는다. 유저가 컨트롤러의 버튼을 누르면 입력 데이터는 PC에게 보내진다.

Fig. 1.의 하단의 PlayStation 4 파트를 보면, PS4는 유저에 의해 게임이 실행되고 PS4는 게임에 필요한 데이터와 게임의 저장 데이터를 프로세스를 통해 주고 받는다.

다음은 Fig. 1.의 상단의 Web Server 파트를 보면, 데이터 흐름이 제일 복잡하며 리모트 플레이의 핵심 프로세스가 속해있어 위협의 위험도가 제일 클 것이라 생각된다. 외부 환경으로부터 유저의 PS4에 대한 무단 접근과 무단 사용을 막아야하며 유저가 실행하고 있는 게임 서비스에 대한 공격을 막아야 한다. 따라서 이 부분을 중점으로 범위를 정해 위협 모델링을 진행할 것이다. 유저가 PC를 이용해 아이디와 패스워드를 입력하여 로그인을 끝내고 나면, 웹서버는 PS4가 연결되어 있는 네트워크 환경을 전달받아 PS4에게 연결을 요청하고 기기의 전원을 켜고 기기가 출력하는 화면을 PC로 공유한다.

3.2 위협 분석

본 섹션에서는 발생할 수 있는 위협에 대해서 분석해보고자 한다. 위협 모델링은 체계적인 순서를 따라야하고 항상 동일한 결과가 나와야하기 때문에 Microsoft 사의 threat modeling 도구인 STRIDE를 이용하여 객관적인 분석을 행하고자한다. PS4는 PC와 비슷하게 CPU, 인터넷 네트워크 환경, GPU, 하드 디스크 등의 시스템으로 구성되어 있기 때문에 소프트웨어나 컴퓨터 기기를 사용할 때의 자산을 보호하는 것과 PS4의 정보 자산을 보호하는 목표가 같다. 따라서 컴퓨터 기기의 자산 보호를 목적으로 한 위협 모델링 분석 기법인 STRIDE를 PS4의 위협모델링에 적용하는 것이 타당하다.

먼저, 분석을 행하기 전에 DFD를 구성하는 요소들은 Table 3.과 같이 기호로 정의한다. 이를 참고하여 일어날 수 있는 모든 위협에 대해 분석하고 STRIDE 유형에 따라 위협을 분류한다. 분류한 위협은 다음 Table 2.와 같다. Table 2.에 나타난 Type

Table 2. Threats

element	type	threat description
E1	S	PC may be spoofed by an attacker to access to Share Display
	R	PC may deny receiving the data from any processes
E2	S	User may be spoofed to access to Download or Execute Game
E3	S	PS4 may be spoofed by an attacker to give wrong display to PC
	R	PS4 may deny receiving the controller input data
P1	S	P1 may be spoofed to get ID/PW of user or permission from accounts DB
	T	P1 may give other user data to get the game the attacker wants or permission to Connect process
	R	P1 may deny receiving data from any processes
	I	ID/PW may be exposed by an attacker
	D	P1 may stop or become unable to do anything
	E	An attacker may give the higher privilege to someone who do not have it
P2	S	P2 may be spoofed to get connect status or display data
	T	P2 may give wrong input to PS4 or give wrong display to PC
	R	P2 may deny receiving the data from any external entities
	I	An attacker can see what user plays
P2	D	P2 may stop or shutdown connection of sharing display
	E	An attacker may give the upper privilege to someone who do not have it
(Ellipsis)		
P5->E1	T	The modified input data may be transmitted
	I	An attacker can see how user plays game
	D	The flow of input data can be jammed
E2->P5	D	The action of user cannot be transmitted

Table 3. symbol settings

elements	name	symbol
external entity	PC	E1
	Human User	E2
	PS4	E3
process	Authentication	P1
	Share Display	P2
	Connect	P3
	Execute Game	P4
	Use Controller	P5
data store	Accounts Database	D1

는 Table 3.의 공격 유형을 의미한다. 예를 들어 S는 Spoofing, R은 Repudiation과 같다. 이 때, STRIDE를 이용하여 나온 위협 중 중요도가 낮은 것은 무시한다.

3.3 Attack Tree 기반 공격 시나리오 분석

STRIDE를 통해 도출된 위협을 이용하여 발생할 수 있는 공격 시나리오를 도출하였다. attack tree는 다음 Fig. 2.와 같으며 주로 리모트 플레이 중에 생길 수 있는 공격 시나리오는 본 유저의 PS4 사용을 막거나 방해하는 데에 있다.

유저의 사용을 방해하는 공격 시나리오에는 세 가지 방법이 있다. 첫 번째는 PC와 PS4 간의 연결을 방해하는 방법이 있다(Interrupt Connection). PC와 PS4 간의 연결 환경과 네트워크 데이터, 연결 상황에 대한 정보를 획득하여 리모트 플레이의 핵심 기능을 방해할 수 있다. 두 번째는 모든 프로세스에서 발생하는 입출력 프로세스를 방해하는 방법이다(Interrupt Input). 세 번째 방법은 게임 데이터와 저장 데이터를 관리하는 프로세스를 공격하는 방법이다(Interrupt Data Management). 데이터가 위조되어 손상되면 유저는 게임 진행이 불가하다.

다음으로 유저의 사용을 막는 공격 시나리오에는 유저의 주도권을 뺏는 방법이 있다(Take the Initiative). 유저가 PS4를 사용하고 있을 때, 유저의 계정을 도용하여 리모트 플레이 서비스에 접속하게 되면 유저의 기존 컨트롤러의 주도권이 리모트 플레이 서비스를 이용하고 있는 공격자에게로 넘어오게 된다. 이와 같이 컨트롤러 주도권을 갖게 된 공격자는 유저의 PS4를 사용중지 시킬 수 있다. 이 과정이 담긴 동영상을 인터넷에 게시하였다[9].

IV. 보안 대책 제안 및 보안성 분석

4.1 보안 대책 제시

4.1.1 비용 대비 분석

도출한 공격 시나리오 중 PS4로의 무단 접근 권한을 얻는 것을 중점으로 보안 대책을 세워볼 것이다. 본 공격이 성공하면 공격자는 큰 권한을 얻게 되고 유저인 척 가장할 수 있게 된다. 이 권한을 얻게 되면 사용자의 플레이를 방해하는 Disturbance 공격보다 더 큰 피해를 입게 된다. 예를 들어 Disturbance 공격은 사용자의 입력을 막거나 데이터를 변조시키는 등 외부에서 사용자의 PS4에 간접적인 공격만을 진행할 수 있지만 사용자의 권한을 빼앗는 Take the initiative 공격은 계정을 도용하고 사용자의 기기 사용을 직접적으로 차단할 수 있기 때문이다. 따라서 본 논문에서는 Disturbance 공격 보다는 Take the initiative 공격에 초점을 두고 이에 대한 보안 대책을 제시할 것이다. 공격자는 Tak

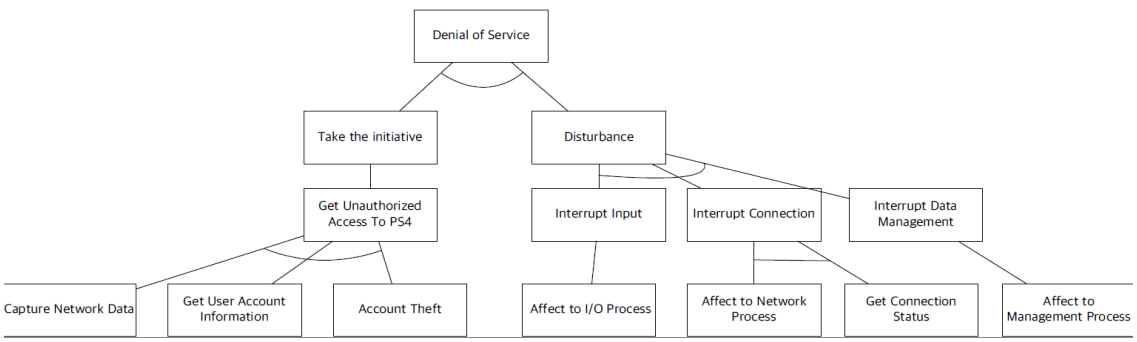

Fig. 2. attack tree

e the initiative 공격을 실행하기 위해 무단 접근 권한을 얻는 것을 목표로 한다. 이를 위해 이용되는 공격방법 중 계정을 도용하는 방법이 있다. 실제 환경에서 많이 일어나고 있는 계정도용 방법에는 많은 공격벡터가 존재한다. 하지만 비용대비 측면에서 이 공격 시나리오를 막기 위해 본 논문에서는 계정도용 유무에 관계없이 무단 접근에 대한 보안 대책을 제시할 것이다. 많은 공격벡터들을 모두 막으려면 비용이 많이 들기 때문에 많은 공격을 막을 수 있고 대부분의 공격 가능성을 성공적으로 줄일 수 있는 하나의 보안 대책을 제시하는 게 비용대비 측면에서 효율적이다. 따라서 본 논문에서는 계정도용이 되더라도 공격자의 무단 접근을 막을 수 있는 보안 대책을 제시한다. 이에 많은 공격을 막기 위해 대책을 세우기보다는 공격이 성공하더라도 공격의 목적을 막는 방법 하나만을 제시하여 공격을 무효화시킬 수 있기 때문에 비용 대비 효율적이다.

4.1.2 무단 접근에 대한 보안 대책 제시

공격자의 무단 접근을 막기 위해 컨트롤러 접근 제어 방법과 OTP(One Time Password) 사용과 같이 두 가지 방법을 보안 대책으로 제시하였다. 컨트롤러 접근 제어 방법은 유저가 사용할 컨트롤러만을 서버에 등록하는 방식으로써 등록이 되지 않은 컨트롤러를 이용한 리모트 플레이 서비스에서 일어날 수 있는 공격을 막을 수 있다. 이 때, 접근 제어 테이블 관리 시스템은 *trustworthy system*이라고 가정한다. OTP 사용방법은 컨트롤러를 사용할 때마다 본인인증이 된 휴대폰에 깔려있는 어플리케이션을 이용하여 전송된 OTP를 알맞게 입력하여 리모트 플레이 접근 권한을 얻는 방법이다. 이 때, 본인인증 서비스는 trustworthy라 가정한다.

4.1.3 유용성 분석

유용성 측면에서 제시된 두 가지 방법을 분석하여 유저의 편의성을 높여줄 뿐만 아니라 보안성 또한 높여줄 수 있는 보안대책에 대해 생각해본다.

첫 번째, 컨트롤러 접근 제어 방법은 유저가 컨트롤러를 구입하여 자신의 계정이 로그인되어 있는 PS 4에 컨트롤러를 연결하면 자동으로 컨트롤러가 등록이 된다. 이 방법을 이용하면 유저는 한 번 기기 등록을 하고 나면 추가적인 인증이 필요 없게 되어 리모트 플레이 서비스 사용이 편리할 것이다.

두 번째 방법은 리모트 플레이 서비스를 이용하기 위해서 매번 OTP를 입력하기 위해 어플리케이션을 이용하는 것은 매우 번거로운 방법이다. 하지만 보안성 측면에서는 효율적인 방법이다.

따라서 본 논문에서는 유용성이 높은 컨트롤러 접근 제어 방법을 쓰되 PC에 연결하여 기기를 사용하기 원할 시에는 OTP를 입력하여 리모트 플레이 권한을 얻을 수 있는 방법을 도입하여 통합적인 보안 대책을 제시하고 이에 대한 안전성을 증명한다.

4.2 보안 대책의 수학적 정형화

임의의 컨트롤러 C_t는 모든 컨트롤러 제품을 뜻하며(수식 1) 접근 제어 테이블에 등록된 컨트롤러는 $Controller\,List$에 속한다(수식 2).

$Playable$ 상태는 컨트롤러를 이용하여 리모트 플레이 서비스를 이용할 수 있음을 의미한다. 모든 컨트롤러의 입력 값은 다음 (수식 3)과 같다. 컨트롤러가 연결할 수 있는 장치들은 $Device\,List$에 속해있다(수식 4).

$$C_t \in All\ Controller \tag{1}$$

$$Controller\,List \in \{C_1, C_2\} \tag{2}$$

$$input = \begin{Bmatrix} \bigcirc, \square, \triangle, \times, \uparrow, \downarrow, \\ \leftarrow, \rightarrow, R1, L1, R2, L2 \end{Bmatrix} \tag{3}$$

$$Device\,List \in \{PS, PC\} \tag{4}$$

4.2.1 접근 제어

유저가 리모트 플레이 서비스를 이용하기 위해 컨트롤러를 PC나 PS4에 연결을 하였을 때, 그 컨트롤러가 $Controller\,List$에 속하는 컨트롤러라면 유저가 리모트 플레이 서비스를 이용가능토록 한다(수식 5).

$$if\ \ C_t \in Controller\,List \rightarrow Playable \tag{5}$$

4.2.2 OTP(One Time Password)

OTP의 동기화는 trustworthy로 가정된 PSN을 통해 시간 동기화 방식으로 진행된다. PSN에서 입력 값으로 OTP를 생성해 서버로 전송하고, 유저는 OTP를 입력한다. 서버는 유저가 입력한 값의 유효성을 검사한다. PSN가 생성한 OTP 값은 A 벡터라 하고 유저가 입력한 OTP 값을 B 벡터라 한다. 각 벡터는 입력의 순서가 존재하며 $input$에 속한다. A와 B 벡터가 같으면 등록되어 있지 않은 임의의 컨트롤러 C_t는 $Controller\,List$에 속하게 된다.

$$if\ (a_1, a_2, a_3, a_4) = (b_1, b_2, b_3, b_4) \rightarrow \{C_t\} \cup Controller\,List \quad (6)$$

$$a_i,\ b_i \in input \mid\ i = 1, 2, 3, 4 \quad (7)$$

4.2.3 프로토콜

제시한 보안대책을 위해 컨트롤러의 정보를 교환하는 프로토콜을 구현해야한다. 본 프로토콜은 Needham-Schroeder 프로토콜을 참고하여 정의하였다. 본 프로토콜에서는 각자의 비밀키는 유출되지 않으며 키 등록 테이블은 $trustworthy\ system$이라 가정한다.

컨트롤러를 서버의 등록 테이블에 등록하기 위해서 Fig. 3.의 과정을 거친다. 등록과정에서 유저는 먼저 공개된 PSN의 공개키를 이용하여 자신의 인증정보와 계정정보 그리고 연결한 컨트롤러의 고유번호를 암호화하여 보낸다. 서버는 자신의 비밀키로 받은 정보를 복호화하여 A를 얻은 후, 랜덤값을 정하고 유저의 공개키를 이용해 서버의 인증정보, A와 랜덤값을 암호화하여 유저에게 보낸다. 이는 서버가 A를 성공적으로 받았다는 걸 뜻한다. 유저는 받은 정보를

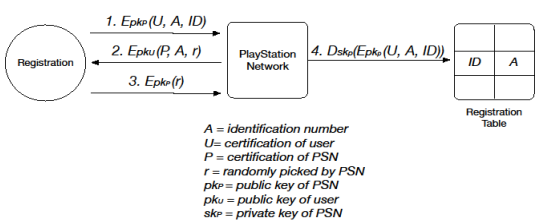

Fig. 3. registration process

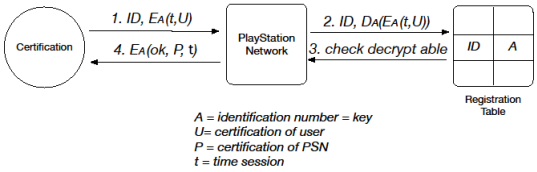

Fig. 4. certification process

복호화하여 랜덤값만을 추출해 암호화하여 서버에게 보낸다. 이로써 서버는 유저가 자신과 통신하던 유저인지를 확인할 수 있다. 성공적으로 통신을 끝마치면 서버는 유저의 계정정보와 컨트롤러의 고유번호를 매칭하여 등록 테이블에 저장한다.

Fig. 4.를 참고하여 유저는 등록과정을 마친 후, 컨트롤러를 사용하기 위해 인증과정을 거쳐야한다. 서버와 유저는 등록과정에서 교환한 대칭키(A)를 가지고 있다. 유저는 A를 이용하여 타임스탬프와 인증정보를 암호화하여 보낸다. 이 때, ID는 공개된 값으로 보낸다. 서버가 어떤 대칭키를 사용해야 할지 알 수 있어야 하기 때문이다. 서버는 이를 받아 복호화하여 등록 테이블에서 ID와 매칭되어 있는 컨트롤러 고유번호를 이용하여 복호화 가능 여부를 확인한 뒤, 성공적으로 복호화가 된다면 유저의 컨트롤러 사용을 허가한다.

4.3 검증

본 섹션에서는 제시한 프로토콜의 안전성과 보안성을 검증 자동화 도구인 scyther를 이용하여 증명한다[10]. 등록과정과 인증과정을 합쳐서 scyther code에 구현하였으며 A와 랜덤값인 ni 그리고 유저의 사용을 허가하는 $success$의 안전성을 증명한다. 안전성 증명을 위한 scyther code는 Fig. 6.과 같다. Fig. 5.를 보면 scyther의 결과에서 모든 공격자가 없다고 검증되었기 때문에 해당 프로토콜은 안전하다.

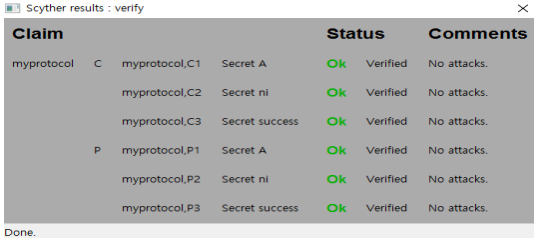

Fig. 5. scyther result

```
/* myprotocol */

usertype identinum;
usertype account;
usertype timestamp;
usertype ok;

const pk: Function;
secret sk: Function;
inversekeys(pk, sk);

protocol myprotocol(C, P)
{
        role C
        {
                const ID: account;
                const A: identinum;
                var ni: Nonce;
                const t: timestamp;
                var success: ok;

                send_1(C, P, {C, A, ID}pk(P));
                read_2(P, C, {P, A, ni}pk(C));
                send_3(C, P, {ni}pk(P));
                send_4(C, P, ID, {C, t}A);
                read_5(P, C, {P, success, t}A);

                claim_C1(C, Secret, A);
                claim_C2(C, Secret, ni);
                claim_C3(C, Secret, success);

        }

        role P
        {
                var ID: account;
                var A: identinum;
                const ni: Nonce;
                var t: timestamp;
                const success: ok;

                read_1(C, P, {C, A, ID}pk(P));
                send_2(P, C, {P, A, ni}pk(C));
                read_3(C, P, {ni}pk(P));
                read_4(C, P, ID, {C, t}A);
                send_5(P, C, {P, success, t}A);

                claim_P1(P, Secret, A);
                claim_P2(P, Secret, ni);
                claim_P3(P, Secret, success);
        }
}
```

Fig. 6. scyther code

V. 결 론

본 논문에서는 PS4와 PC 간의 리모트 플레이 환경에서 발생할 수 있는 위협을 threat modeling을 통해 도출하였다. 각 도출된 위협을 이용하여 가능한 공격 시나리오를 정의하여 attack tree로 나타내었다. 많은 취약점 중 최종 목표 달성을 저지하고 또 하나의 공격 방법이 될 수 있는 무단 접근을 하나의 방법으로 막는 것이 비용대비 관점으로 보았을 때, 의미 있는 결정이라고 보았다. 또한, 각 공격시나리오 별로 위험도를 따졌을 때, 위험도가 높은 위협을 막는 것이 효율적이다. 따라서 PS4의 무단 접근에 대한 보안대책을 정의하였으며 그에 대한 정형화 명세를 통해 안전성을 증명하였다.

References

[1] Suvda Myagmar, Adam J. Lee, and William Yurcik, "Threat modeling as a basis for security requirements," *Symposium on requirements engineering for information security (SREIS)* vol. 2005, pp.1-8, Aug. 2005.

[2] Saini Vinnet, Qiang Duan, and Vamsi Paruchuri, "Threat modeling using attack trees," *Journal of Computing Sciences in Colleges* vol. 23, no. 4, pp.124-131, Apr. 2008.

[3] Johansson, Jesper M, "Network threat modeling," *Enabling Technologies: Infrastructure for Collaborative Enterprises,. WET ICE 2003. Proceedings. Twelfth IEEE international Workshops on.* IEEE, Jun. 2003.

[4] Dekker Marnix and Giles Hogben, "Appstore security: 5 lines of defence against malware," *ENISA Report*, Sep. 2011.

[5] Davies, M., Read, H., Xynos, K. and Iain S.. "Forensic analysis of a Sony PS4: A first look," *Digital Investigation* vol. 12: pp.81-89, Mar. 2015

[6] Vaughan, Chris. "Xbox security issues and forensic recovery methodology (utilising Linux)," *Digital Investigation* vol. 1, no. 3: pp.165-172, Sep. 2004

[7] Jeff Yan, Jianxin and Hyun-Jin Choi. "Security issues in online games," *The Electronic Library* vol. 20, no. 2, pp. 125-133, 2002

[8] Cone, Benjamin D., Irvine, C.E., Thompson, M.F. and Thuy D.N.. "A video game for cyber security training and awareness," *computers & security* vol. 26, no. 1: pp. 63-72, Feb. 2007

[9] https://youtu.be/-BjaaRGxxGE, the video of PS4 threat situation.

[10] Cremers, Cas J.F. "The scyther tool:

Verification, falsification, and analysis of security protocols," *International Conference on Computer Aided Verification*. Springer Berlin Heidelberg, pp.414-418, Jul. 2008.

〈저자소개〉

김 혜 민 (Hye Min Kim) 학생회원
2017년 2월: 고려대학교 컴퓨터학과 졸업
2017년 3월~현재: 고려대학교 정보보호학과 석사과정
〈관심분야〉 데이터마이닝, 데이터 분석, 온라인게임 보안

김 휘 강 (Huy Kang Kim) 종신회원
1998년 2월: KAIST 산업경영학과 학사
2000년 2월: KAIST 산업공학과 석사
2009년 2월: KAIST 산업및시스템공학과 박사
2004년 5월~2010년 2월: 엔씨소프트 정보보안실장, Technical Director
2010년 3월~2014년 12월: 고려대학교 정보보호대학원 조교수
2015년 1월~현재: 고려대학교 정보보호대학원 부교수
〈관심분야〉 온라인게임 보안, 네트워크 보안, 네트워크 포렌식