한국융합보안학회 2021년 하계학술대회 논문집

신원 기반 네트워크 패킷 접근제어 시스템 개발

조재현[1]*, 김현진[1], 허송이[1]
중부대학교[1]

Development of identity-based network packet access control system

Jaehyeon Cho[1]*, Hyeonjin Kim[1], Songyi Heo[1]

요약 : 재택근무 환경에서 직원들이 내부망으로 접근하기 위해 핸드폰 본인인증 등과 같은 다양한 인증을 거쳐야 접속한다는 불편함이 존재한다. 이러한 문제를 해결하기 위해 본 논문에선 지문 인식 기반 사용자 인증을 안전하게 수행할 프로토콜을 정의하고, 보안 라우터를 통해 기업 내/외부망에 접근하는 네트워크 패킷을 제어하는 시스템을 개발한다. 위협 대응을 위해 관제 웹 서버를 구성해 기업 내/외부망에 접근하는 인증 성공, 실패 관련 로그를 모니터링 및 관리할 수 있도록 한다.

Key Words : Fingerprint authentication, Protocol Design, Packet management, Network Separation

1. 서 론

코로나19 바이러스의 장기화로 기업에서는 비대면 업무가 일상화되고 있다. 이에 따라 RDP, SDP, VDI를 이용해 스마트워크 환경을 구축하고 있다[1]. 이때 VDI는 가상 데스크톱 환경을 지원하는 기술로 공공기관에서 많이 도입하여 쓰고 있다. 이 기술은 SSL VPN을 이용하여 사용 권한 인증을 받고 또한 2-Factor 인증을 요구하기도 한다[1]. 이로 인해 원격 근무를 하는 사용자는 특정 환경에 접근하기 위해 여러 번 인증을 수행해야 하는 등의 불편함을 겪는다. 보안성을 증가시키며 간편한 인증을 할 수 있는 환경을 설계/개발한다면 근무 환경의 생산성을 높일 수 있을 것이다.

따라서 본 연구에서는 지문 인증을 이용하여 간편한 인증을 실현할 것이며, 이를 통해 내/외부망에 접근하는 패킷을 제어할 것이다. 또한, 관리/관제 사이트를 통해 사용자, 포트 관리를 손쉽게 해주며 로그들을 통해 부적절한 접근을 확인할 수 있는 시스템을 설계 및 개발하고자 한다. 특히 Challenge-Response 인증 프로토콜을 직접 설계하여 빠른 속도의 인증과 동시에 보안을 강화하고자 한다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 기술에 대해 알아본다. 3장에서는 애플리케이션 설계 4장에서는 애플리케이션 개발에 관해 설명하고 5장에서는 결론을 내며 본 논문을 마치고자 한다.

2. 관련 기술

2.1 지문 인증

지문 인증은 입 퇴실 관리, PC나 핸드폰의 로그인, 핀테크 서비스 등 다양한 영역에 이용되고 있다. 또한, SE를 이용하여 지문 인증을 설계한 연구[2]와 같이 다양한 기술과 결합하여 보안을 더욱 강화하는 연구가 진행되고 있다.

지문 인증은 또 다른 인증 수단인 공인인증서, ID/PW 입력 등과 비교했을 때 다양한 강점을 지니고 있다. 비밀번호의 경우 사용자들은 기억하기 쉬운 번호를 사용하며 재사용되기도 한다. 따라서 노출되기 쉬원 단점이 있다. 반면 공인인증서의 경우 보안성은 좋지만, 개발 및 운영비용이 증가한다는 단점이 있다. 지문 인증은 생체 정보를 활용하며 비교적 투자 비용도 적기 때문에 두 단점을 보완해주는 장점이 있다[3].

따라서 본 연구에서는 간편하면서도 보안성이 높은 지문 인증을 이용하고자 한다.

2.2 Challenge-Response

Challenge-Response란 다음과 같다. 서버가 임의의 난수값을 생성해 사용자에게 보내고 사용자는 이 난수값을 보호할 정보와 암호 알고리즘 혹은 비밀키를 이용하여 암호화한다. 그리고 그 암호 값을 서버에게 전달하여 인증을 수행한다. 해당 방식은 인증 서버가 사용자의 토큰이나 동기를 유지할 필요가 없으며 재사용 공격에서는 안전하다는 특징이 있다.

본 연구에서는 인증 속도를 고려하여 네트워크 영역에서의 인증을 실현하고자 하였다. 이때 인증에 필요한 정보들은 네트워크를 통해 주고받기 때문에 보안은 필수적이다. 따라서 해당 방식을 이용하여 여러 네트워크 공격으로부터 정보를 안전하게 보호하고자 하였다.

2.3 망 분리 환경에서의 보안 솔루션

망 분리 환경은 외부에서는 악성코드 또는 해커, 내부에서는 산업 스파이 등으로 인한 기업 내 정보 유출의 위험성을 가지고 있다. 따라서 이에 대응하기 위한

보안 솔루션들이 여러 가지가 있는데 DB 보안, DRM, DLP 등이 있다. 또한, 보안 로그를 이용하여 이상 징후를 발견하는 방법 등이 있다[4].

본 연구에서는 보안 로그를 이용해 인증 과정에서 발생 가능한 위협을 식별하는 방식을 채택하였다. 주고받는 패킷의 송수신 성공, 실패 여부를 로그로 남길 것이며 더 나아가 IDS 기능을 구현하여 보안 로그를 남길 것이다. 이후 정상인증 로그와 비교하여 이상 징후를 탐지하는 이상 징후 탐지, 패턴화된 공격을 탐지하는 오용탐지 기법[4] 등을 이용해 위협을 발견하고 차단하고자 한다.

3. 애플리케이션 설계

3.1 시스템 구성도

본 논문에서 제안하는 시스템 동작 과정은 다음 [그림 1]과 같다. 해당 구성도에서의 라우터는 스위치 등으로 교체될 수 있으며 내부망에서 외부망으로 나가는 패킷을 통제하는 시나리오로 구성하였다.

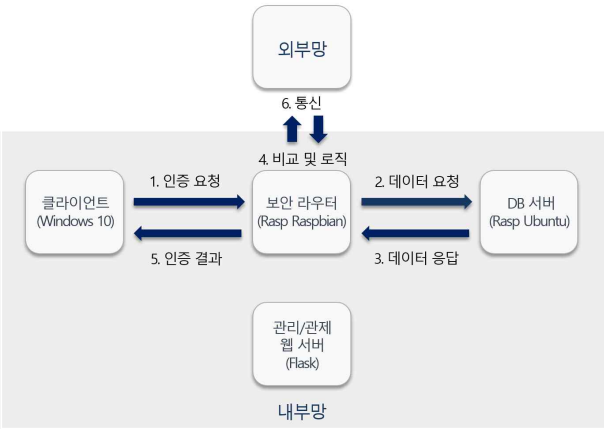

Fig. 1. System Configuration Diagram

3.2 C/R 인증 프로토콜 설계

인증 과정에서의 속도와 보안성을 증가시키기 위해 직접 프로토콜을 설계하였다. 안전한 통신을 위해 OTP 방식 중 하나인 Challenge-Response 방식을 채택하였으며, [그림 2]가 설계한 프로토콜이다.

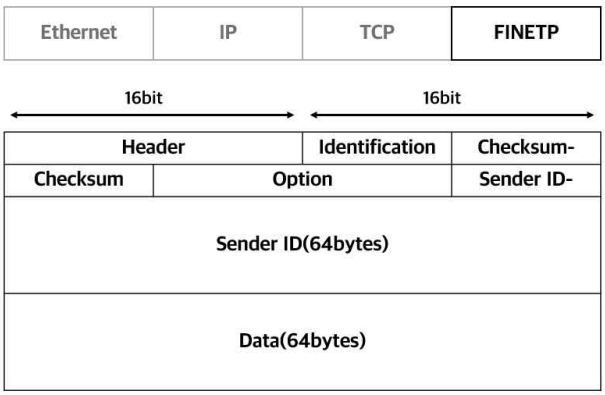

Fig. 2. FINET Protocol

Header 필드는 2 Bytes로 해당 프로토콜을 식별하기 위한 값이다. Identification 필드는 2 Bytes이며 인증 과정의 단계를 식별하기 위한 Flow 값을 나타낸다. Checksum 필드 역시 2 Bytes로 해당 프로토콜의 무결성을 검사하기 위해 사용된다. IP Header의 Checksum 계산 방식을 채택하였으며 계산 과정은 다음과 같다. Checksum 필드를 제외한 Header 필드부터 Data 필드까지 전부 더한 뒤 1의 보수 형태를 취한다. 이후 뒤 2 Bytes를 Checksum 값으로 사용한다. 검증 과정 역시 위와 같다. Option 필드는 2 Bytes로 Success 혹은 Fail을 의미하며, Challenge 값을 보내는 Flow에서는 Challenge 값이 담긴다. Sender ID 필드는 64 Bytes이며 평소엔 패딩 상태이다. Response 과정에서 FIDO 방식일 경우 사용자의 해시화 된 MAC 정보, ID/PW 방식을 경우 사용자의 ID 정보를 포함해 인증 시 사용자를 식별하기 위한 정보로 사용된다. Data 필드도 64 Bytes로 평소엔 패딩 상태이다. Response 과정에서 FIDO 방식은 GUID와 Challenge 난수가 더해져 해시화 된 값, ID/PW 방식은 PW와 Challenge 난수가 더해져 해시화 된 값을 포함해 사용자 인증에 사용된다. 다음은 [그림 3] 해당 프로토콜을 이용한 인증 수행 과정이다.

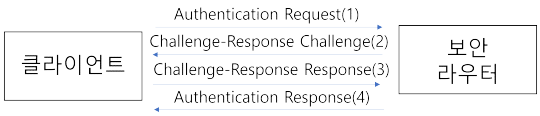

Fig. 3. C/R Authentication process

클라이언트에서 서버로 Authentication Request 패킷을 보내 인증을 요청한다. 그럼 서버는 난수 값을 담은 Challenge 패킷을 클라이언트에게 보낸다. 클라이언트는 받은 Challenge 값으로 Data를 암호화시키고 그 정보가 담긴 Response 패킷을 보낸다. 서버는 암호화된 인증 정보를 DB에 저장된 인증 정보와 비교를 수행하고, 최종적으로 인증 성공, 실패 여부를 포함한 Authentication Response 패킷을 클라이언트에게 보낸다.

4. 애플리케이션 개발

4.1 개발 환경

개발 환경은 개발 환경은 [표 1]과 같다. 클라이언트는 Windows 10 환경, 보안 라우터는 라즈베리파이 Ubuntu 환경에서 개발을 진행하였다. 지문 인증을 위한 Framework는 Windows Biometric을 이용했다. 클라이언트와 서버 통신을 위한 라이브러리는 Socket, 방화벽 기능을 위한 라이브러리는 Netfilter를 사용해 C++ 언어로 개발하였다. 관리/관제 사이트는 Python 언어로 Flask 라이브러리를 이용하여 구현하였다.

Table 1. Development Environment

Development Tool	Visual Studio, Qt
Programming	C++, Python
Library	Socket, Netfilter, MFC
Framework	Windows Biometric
Web	Flask, Mysql
OS	Windows 10(Client),

	Ubuntu(Server)
IOT	Rasberry PI 3, 4

4.2 Database 설계

DB에는 인가된 사용자의 정보들이 담긴다. Client의 기기 및 지문정보인 MAC 주소와 WinBioDatabase ID 그리고 대체 수단 제공을 위한 User ID, User Password가 DB에 저장된다. 모든 정보는 SHA-256 암호화 알고리즘을 통해 일 방향 암호화하며, [표 2]와 같이 구성한다.

Table 2. Database scheme

컬럼명	타입	옵션	정보
member_idx	int(10)	PRI, NOT NULL, AUTO_INCREMENT	기본키, 인덱스
member_id	varchar(64)	UNIQ, NOT NULL	Client MAC Addr (SHA-256)
member_guid	varchar(64)	UNIQ, NOT NULL	Client GUID (SHA-256)
member_subid	varchar(20)	UNIQ, NOT NULL	Client ID (SHA-256)
member_subpw	varchar(64)	UNIQ, NOT NULL	Client PW (SHA-256)

4.3 클라이언트

클라이언트 프로그램에서는 두 가지 기능을 제공해야 한다. 첫째, 지문 인식 또는 ID/PW 방식으로 인증을 요청한다. 둘째, 편리성을 위해 사용자 등록에 필요한 정보(MAC 주소, WinBioDatabas ID)를 보여준다. 다음 [그림 3]은 개발된 모습이다.

Fig. 3. Client Program

4.4 서버

서버에서는 네 가지 기능을 제공한다. 첫째, 클라이언트로부터 받은 인증 정보와 DB에 저장된 인증 정보를 비교하여 인증을 수행한다. 둘째, 인가된 사용자에게 일정 시간 동안 포트를 열어준다. 셋째, 통신 과정에서 발생하는 모든 에러 로그들을 파일에 저장한다. 넷째, 인증에 3번 이상 실패 시 계정을 잠금 가능해야 한다.

4.5 관리/관제 사이트

관리/관제 사이트에서는 사용자와 포트의 등록, 수정, 삭제를 편리하게 해주며 에러 로그들을 확인할 수 있어 부적절한 접근을 식별할 수 있다. [그림 4]는 관리/관제 사이트의 모습이다.

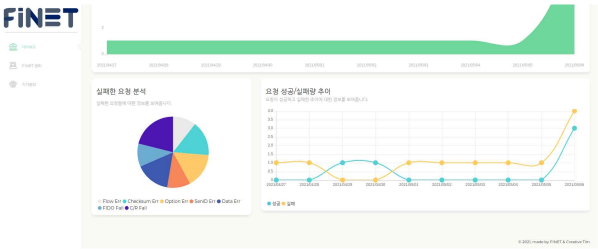

Fig. 4. Admin Control site

3. 결론

망 분리 환경에서 내/외부망 접속을 시도할 때 인증에 대한 불편함이 존재하였다. 본 논문은 이러한 문제점을 해결하기 위해 지문 인증을 인증 수단으로 선택하고 인증 속도를 증가시키고 보안을 강화하기 위해 Challenge-Response 방식의 프로토콜 및 패킷을 설계하고 패킷의 이상 징후를 확인할 수 있는 관리/관제 사이트를 구축하였다. 이를 통해 지문 인식을 통한 패킷 제어 시스템 구현을 실현하였으며, 보안성과 편리성을 개선 시킬 수 있었다. 이와 같은 연구를 지속하여 원격 근무 환경에서도 편리하게 내부자 인증을 수행하고, 보안성을 증가시킬 방안을 연구해 나가다 보면 기업은 보안은 물론 기존보다 더 나은 생산성을 기대할 수 있을 것으로 기대한다.

참고문헌

[1] 공공기관 물리적 망분리 환경에서의 비대면 스마트워크 근무 환경구축을 위한 보안 모델 연구, Journal of the Korea Convergence Society, Vol. 11, No. 10, 2020, pp. 37-44.

[2] SE를 이용한 FIDO Authenticator 설계, 한국정보과학회 학술발표논문집, 2019.12, 1654-1656(3 pages)

[3] OpenSource를 이용한 FIDO 인증 시스템에 관한 연구, Journal of the Korea Convergence Society, Vol. 11, No. 5, 2020, pp. 19-25.

[4] A Study on the Improvement of Security Monitoring in the Separate Network Environment, Journal of Knowledge Information Technology and Systems(JKITS), Vol. 9, No. 6, December 2014, pp. 805~819.