안전한 P2P 컴퓨팅을 위한 UDP 홀펀칭과 Wireguard를 결합한 VPN 망의 구현

이재석, 박재형*

전남대학교 컴퓨터정보통신공학과

wotjr6573@jnu.ac.kr, *hyeoung@jnu.ac.kr

Implementation of a Virtual Private Network using UDP Hole Punching and Wireguard for Secure P2P Computings

Jae Seok Lee, Jaehyung Park

Dept. of Computer Engineering, Chonnam National University

요 약

P2P 망은 중앙 서버 없이 서로 통신을 수행하는 컴퓨터들의 집합으로 디스크 공간이나 처리 능력을 공유하는 응용에 많이 사용된다. P2P 망에 공유된 자료와 자원을 특정 사용자 그룹에서만 접근할 수 있고 안전하게 송수신할 수 있는 지원 기법이 필요하다. 그러므로, 본 논문은 STUN Protocol을 응용하여 동적 제약 조건을 갖는 P2P 망에서 Wireguard 기반의 VPN을 구축하였다. 구축된 VPN 망에서 P2P 통신이 암호화되고 주로 로컬 네트워크(내부망)에서 사용되는 파일공유 시스템인 SMB/NFS가 안정적으로 실행됨을 확인하였다.

Ⅰ. 서 론

P2P(Peer-to-Peer) 통신 기술은 중앙 서버의 부하를 줄이고, 각 노드가 직접 통신함으로써 네트워크 효율성을 향상시킨다. 그러나 현재 IPv4 네트워크 환경의 IP 주소 공간 부족 문제로 인해 NAT 기술[1]이 필수적이 되었고, 이는 하나의 공인 IP 주소 아래 여러 개의 사설 네트워크 기기가 인터넷에 접속하도록 한다. 이러한 구조는 P2P 통신의 복잡성을 증가시키며, 특히 NAT 환경에서는 STUN 프로토콜[2,3]을 활용한 홀펀칭 기법[4]이 필요하게 된다. 또한, 이러한 P2P 통신을 원활하게 하기 위해 UPnP와 같은 기술이 도입되었으나 보안취약성 문제가 대두되었다.

본 논문에서는 P2P 네트워킹의 보안 취약성을 해소하고, 유연하게 네트워크를 구성하며 랜덤 포트를 이용하여 노드 간 데이터를 직접 교환할 수 있는 통신 방법을 제안한다. 이를 위해 STUN 프로토콜을 응용한 UDP 홀펀칭 기법과 UDP 기반의 WireGuard 프로그램[5]을 통합하여, NAT 환경의 제약을 극복하는 유연한 가상 사설 네트워크(VPN) 구축 방법을 개발하였다. 본 연구는 복잡한 네트워크 환경에서도 효과적으로 작동할 수 있는 새로운 VPN 기술의 활용 가능성을 제시하고자 한다.

Ⅱ. 제안한 P2P VPN 구조

본 논문에서는 STUN 프로토콜을 활용하여 사용자의 공인 IP 주소뿐만 아니라 대기 중인 사용자들의 IP 주소와 가상 사설망을 위한 목표 IP 주소(TIP)를 할당하고 전달한다. 또한, 대기자들의 UDP VPN(WireGuard) 연결을 위해 PublicKey를 교환한다. 이 과정을 통해 STUN 서버를 제외한 모든 사용자들은 서로의 hole punching된 UDP 포트 번호와 IP 주소 정보를 획득하게 되며, STUN 서버를 통해 사전에 할당된 TIP를 이용해 사설망을 구성하기로 합의하게 된다. 네트워크는 hole punching된 UDP 포트를 통해 VPN 패킷의 handshake가 시작되고, 키 교환 후에는 네트워크 구성이 완료되어 그림 1과 같이 가상 사설망이 구축된다.

이러한 P2P 구조에 따라 고정된 IP 주소가 없는 NAT 환경에서도 VPN 터널을 구현하고, 가상의 사설망을 성공적으로 구축할 수 있다.

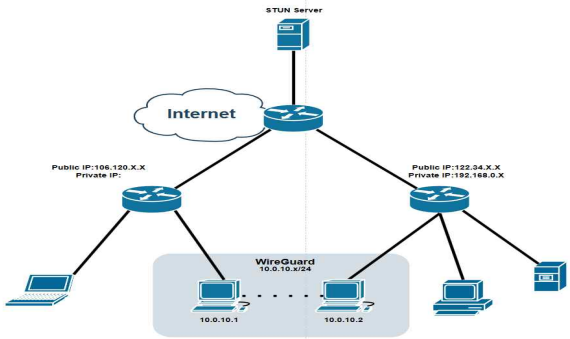
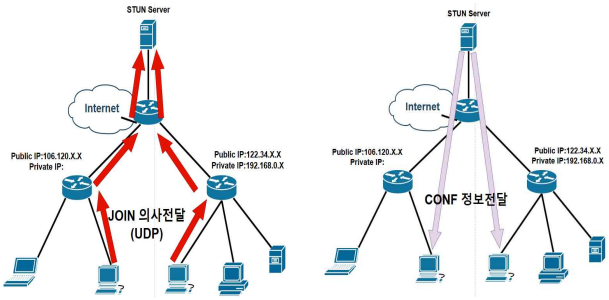


그림 1. P2P VPN 구성도

본 논문에서 설계된 VPN의 동작 과정은 다음과 같다:

1. JOIN 패킷 전송: 사용자는 UDP를 통해 STUN 서버에 JOIN 패킷을 전송한다. 이 패킷에는 사용자의 PublicKey가 포함되어 있다.



(a) JOIN 패킷 전송 (b) CONF 정보 전달

그림 2. P2P VPN 설정(Config) 전달 과정

2. 대기 목록 관리: 서버는 연결을 요청하는 사용자들의 목록을 Wait_list에 저장한다. UDP 홀 펀칭 테이블이 유지되도록, 마지막 요청 이후 15초가 경과하면 해당 요청은 대기 목록에서 자동으로 삭제된다.

3. 네트워크 구성 정보 전송: 대기 목록의 인원이 충족되면, STUN 서버는 TIP를 할당하고, RTC_[IP주소]TIP_[TIP]-pub[PublicKey] 형식으로 네트워크 구성 정보를 대기 중인 사용자들에게 전달한다. [그림3]

4. WG_UP 명령: 모든 정보가 전송된 후, 서버는 WG_UP 명령을 보내고 Wait_list를 삭제한다. 이후 15초의 쿨타임을 가진다.
5. 구성 파일 자동 생성: 모든 사용자는 WG_UP 명령을 받은 후 전달된 정보를 바탕으로 구성 파일을 자동으로 생성한다.
6. 사설망 구성: 생성된 구성 파일을 바탕으로 사설망을 구축한다.

Ⅲ. 구현 및 실험 결과

본 논문에서 제안한 P2P VPN 시스템은 STUN 프로토콜을 활용하여 NAT 환경에서도 효과적으로 동작하도록 설계되었다. 주요 실험은 동적 IP 환경과 NAT 구성을 사용하여 수행되었으며, WireGuard 기반의 UDP 터널링을 통해 제한된 IP주소와 많은 수의 기기가 연결된 상태에서 P2P VPN의 동작 과정과 SMB/NFS 서비스 실행과정을 실험하였다.

그림 3. VPN IP 10.0.10.x인 호스트 환경에서 받은 config 메시지

그림 3은 서버에서 구성값을 대기 중인 노드들에게 전송한 메시지를 나타낸다. 여기서 'RTC_'는 실제 공인 IP 주소와 포트 번호를, 'TIP_'는 가상으로 할당된 IP 주소를 나타낸다. 또한, '-pub' 접미사는 할당된 장치의 공개 키(public key)를 의미한다.

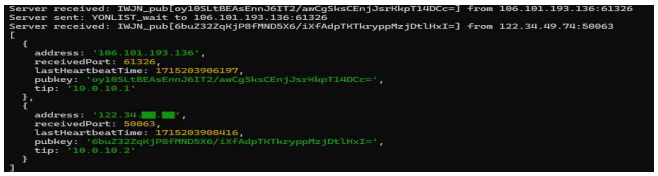


그림 4. STUN Server

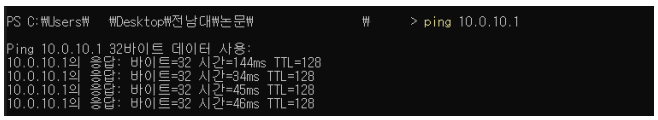


그림 5. LTE 환경에서 통신 테스트 결과

그림 4는 STUN 서버의 구성값을 보여주며, 그림 5는 서버에서 보낸 설정값을 바탕으로 연결된 VPN이 동작함을 보이는 그림이다.

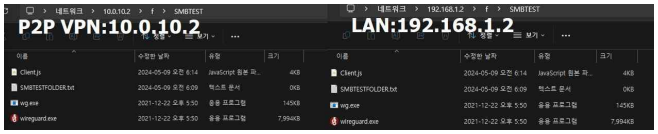


그림 6. 구현된 VPN 상에서 SMB/NFS 동작 테스트

그림 6은 로컬 네트워크에서 사용되는 파일 공유 시스템인 SMB/NFS의 작동을 확인하기 위해 설정된 매개변수와 동일한 폴더에 대한 접근을 서로 다른 주소를 통해 접근이 가능함 시각적으로 보여준다. 그림 7의 상단이미지는 기존 평문 UDP 통신을 하단 이미지는 P2P VPN을 통해 구성된 네트워크에서 패킷이 암호화되어 전송되는 패킷을 보여준다

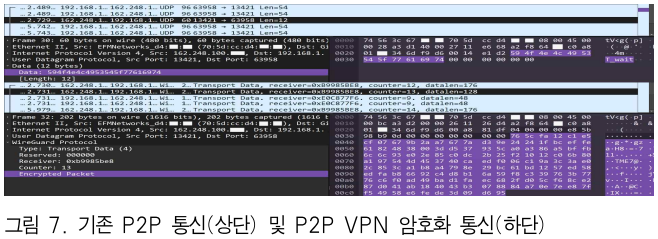


그림 7. 기존 P2P 통신(상단) 및 P2P VPN 암호화 통신(하단)

표 1. P2P VPN 연결측정

연결여부	평균 연결속도	지연시간(VPN)
50/50 연결성공	200~300 ms	=< 7 ms

표2. 인터넷과 VPN프로그램간의 통신속도 차이 비교

\	최소	최대	평균
인터넷	229 ms	253 ms	238 ms
P2P VPN	229 ms	260 ms	240 ms

실험 결과, 개발된 시스템은 NAT 환경에서 무작위로 배정된 포트를 사용하여 VPN 네트워크를 성공적으로 구성할 수 있음을 입증하였다. 이 시스템은 VPN 프로그램을 통해 UDP 통신 중 발생할 수 있는 데이터 손실을 방지하고, 전송되는 모든 데이터를 암호화하여 보안성을 강화하였다. 총 50회의 연결 테스트를 실시한 결과, 모든 시도에서 연결이 성공적으로 완료되었다. 표1과 2의 결과와 같이, 첫 번째 연결을 위한 핸드셰이크 단계는 평균 220밀리초(ms)의 시간이 소요되었으며, 이후 데이터 전송 시 패킷당 평균 7밀리초 이하의 지연 시간을 기록하였다. 또한, 구축된 가상 네트워크는 기존의 VPN 프로그램들과 높은 호환성을 보이며 기능적으로 동등한 성능을 제공하였다.

Ⅳ. 결론

본 연구를 통해 랜덤하게 배정된 포트와 NAT 환경에서 UDP 홀펀칭 기술을 이용하여 가상의 사설망을 구축하고 안정적으로 동작함을 보였다. 구현된 사설망은 일반적인 내부망과 동일하게 동작하며, P2P 통신 중에도 안전하게 암호화된 VPN 터널을 통해 데이터를 전송할 수 있다. 이러한 내부망을 이용함으로써, 공인 IP의 부족으로 인해 이용에 제약이 있었던 프로그램들과 내부망 전용으로 설계된 프로그램 간의 호환성을 보장할 수 있다.

추가적인 연구를 통해 Symmetric NAT 환경에서의 동작 가능성을 검토하고, 한 개의 공인 IP에서 여러 포트를 통해 서로 다른 디바이스로 인식될 수 있도록 구현하는 방안을 모색한다면, 기존의 내부망 설계 프로그램과 P2P 기반의 프로그램을 결합하여 네트워크 트래픽을 줄이고 보안성 향상에 도움이 될 것으로 예상된다.

ACKNOWLEDGMENT

이 논문은 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 지역지능화혁신인재양성사업임. (IITP-2024-00156287)

참 고 문 헌

[1] P. Srisuresh and M. Holdrege, “IP Network Address Translator (NAT) Terminology and Considerations,” RFC 2663, 1999.

[2] T.H. Tran, J. Park, Y. Won, and J. Kim, “Combining STUN protocol and UDP hole punching technique for peer-to-peer communication across network address translation,” International Conference on IT Convergence and Security, 2014.

[3] J. Rosenberg, J. Weinberger, C. Huitema, R. Mahy, “STUN-Simple Traversal of User Datagram Protocol Through Network Address Translators,” RFC 3489, 2003.

[4] P. Srisuresh, B. Ford, D.K egel, “State of Peer-to-Peer (P2P) Communication across Network Address Translators (NATs),” RFC 5128, 2008.

[5] J. A. Donenfeld, "WireGuard: Next Generation Kernel Network Tunnel," Technical White Paper, 2018. https://www.wireguard.com/papers/wireguard.pdf