SSH Protocol

* **SSH Protocol**

**SSH(secure shell) 프로토콜**은 네트워크 상에서 데이터를 안전하게 주고받기 위해 설계된 **암호화된 통신 프로토콜**입니다. 주로 원격 서버에 로그인하거나, 명령어를 실행하고, 파일을 전송하는 데 사용됩니다. SSH는 공개 키 암호화 방식을 사용하여 통신을 보호하고, 비밀번호나 다른 인증 정보를 안전하게 전송할 수 있게 합니다.

**주요 기능**

1. **원격 접속**: SSH는 원격 서버에 안전하게 로그인하고 명령어를 실행할 수 있는 가장 일반적인 방법입니다.
2. **파일 전송**: SSH 프로토콜을 사용하여 **SCP(Secure Copy Protocol)** 또는 \*\*SFTP(Secure File Transfer Protocol)\*\*을 통해 안전하게 파일을 전송할 수 있습니다.
3. **포트 포워딩 및 터널링**: SSH는 네트워크 보안을 강화하기 위해 포트 포워딩 및 터널링 기능을 지원합니다. 이를 통해 로컬 네트워크에서 원격 네트워크에 안전하게 연결할 수 있습니다.

* **SSH의 동작 방식**

SSH는 **클라이언트-서버 구조**로 동작하며, 세션을 시작할 때 클라이언트는 서버에 연결 요청을 보냅니다. 이 과정에서 **공개 키 암호화 방식**을 사용하여 클라이언트와 서버 간의 연결을 암호화하고 보호합니다.

1. **클라이언트가 서버에 연결**: SSH 클라이언트는 서버에 연결하고, 서버는 자신을 증명하기 위해 **공개 키**를 클라이언트에게 보냅니다.
2. **인증 과정**: 클라이언트는 서버에 대한 인증을 확인한 후, \*\*개인 키(Private Key)\*\*를 사용하여 서버에서 요구하는 인증을 수행합니다.
3. **암호화된 통신 채널**: 인증이 완료되면 클라이언트와 서버 간에 암호화된 통신 채널이 설정됩니다. 이 통신 채널을 통해 안전하게 데이터를 주고받을 수 있습니다.

* **SSH 프로토콜의 구성**

SSH 프로토콜은 세 가지 주요 구성 요소로 나뉩니다:

1. **Transport Layer Protocol**: 데이터의 암호화 및 무결성 검증을 담당합니다.
2. **User Authentication Protocol**: 사용자 인증을 처리하며, 비밀번호, 공개 키, 기타 인증 방식을 사용할 수 있습니다.
3. **Connection Protocol**: 원격 명령어 실행, 파일 전송, 포트 포워딩 등 SSH 세션에서 여러 채널을 처리하는 프로토콜입니다.

**주요 포트**

**SSH는 기본적으로 TCP 포트 22번**을 사용하여 통신합니다. 네트워크 방화벽을 설정할 때 이 포트를 열어야 SSH 연결을 허용할 수 있습니다.

**SSH의 보안 이점**

**암호화**: 데이터가 네트워크를 통해 전송될 때 암호화되어 도청을 방지합니다.

**인증**: 사용자와 서버의 신원을 확인하여 신뢰할 수 있는 통신을 보장합니다.

**무결성**: 데이터의 무결성을 보장하여 중간에서 데이터를 수정할 수 없도록 합니다.

* SSH URL 형식

ssh://[user]@[host]:[port]/[path]

각 구성 요소에 대한 설명은 다음과 같습니다:

* user: 원격 서버에서 사용자의 계정 이름.
* host: 서버의 도메인 이름 또는 IP 주소.
* port: SSH 연결에 사용되는 포트 번호(기본적으로 22번). 생략 가능.
* path: 원격 서버에서 접근하려는 파일 경로나 리포지토리 위치
* SSH의 **공개 키 인증 (Public Key Authentication)** 과정
  1. 클라이언트가 SSH 연결을 요청하면, 서버는 클라이언트에게 난수를 보냅니다.
  2. 클라이언트는 자신의 **개인 키**로 이 난수를 암호화한 후 서버에 돌려줍니다.
  3. 서버는 클라이언트의 **공개 키**를 사용하여 이 암호화된 데이터를 복호화하고, 올바르게 복호화된다면 클라이언트가 올바른 **개인 키**를 가지고 있다고 판단하여 인증을 허용합니다.
* /.ssh 폴더 내부의 id\_rsa (Private Key)와 id\_rsa.pub (Public Key)파일 설명

SSH 클라이언트는 기본적으로 **사용자의 홈 디렉토리**에 위치한 **.ssh 폴더**에서 키 파일을 관리합니다. 이 폴더 안에는 SSH 키와 관련된 여러 파일이 포함될 수 있으며, 특히 id\_rsa와 id\_rsa.pub 파일은 **개인 키**와 **공개 키**를 나타냅니다.

* **id\_rsa 파일 (Private Key, 개인 키)**

**내용: 이 파일에는 클라이언트의 개인 키가 저장되어 있습니다. 개인 키는 암호화된 데이터의 복호화 또는 서명을 생성할 때 사용됩니다.**

**역할: 이 키는 외부에 절대 노출되어서는 안 됩니다. 개인 키는 서버로부터 받은 난수 데이터를 암호화하여 서버에 보낼 때 사용됩니다. 서버는 이 데이터를 클라이언트의 공개 키로 복호화하여 클라이언트가 올바른 개인 키를 가지고 있는지 확인합니다.**

**형식: 개인 키는 일반적으로 PEM 형식으로 인코딩된 텍스트입니다. 파일 내용은 다음과 같은 형태입니다:**

-----BEGIN OPENSSH PRIVATE KEY-----

(여러 줄의 암호화된 데이터)

-----END OPENSSH PRIVATE KEY-----

* **id\_rsa.pub 파일 (Public Key, 공개 키)**

**내용**: 이 파일에는 클라이언트의 **공개 키**가 저장되어 있습니다. 공개 키는 암호화된 데이터를 복호화하거나, 클라이언트의 서명을 검증하는 데 사용됩니다.

**역할**: 공개 키는 서버에 저장되며, 서버는 이 공개 키를 사용하여 클라이언트가 올바른 개인 키를 소유하고 있는지 검증합니다. 공개 키는 누구에게나 공개할 수 있으며, 이를 통해 클라이언트를 인증할 수 있습니다.

**형식**: 공개 키는 일반적으로 한 줄의 텍스트로 저장됩니다. 파일 내용은 다음과 같은 형식입니다:

ssh-rsa AAAAB3Nza...base64로 인코딩된 데이터... 사용자@호스트

ssh-rsa는 RSA 방식의 SSH 키임을 나타냅니다.

AAAAB3Nza...는 base64로 인코딩된 공개 키의 실제 값입니다.

사용자@호스트는 키를 생성한 사용자와 호스트에 대한 메타데이터입니다.