* Banner Grabbing을 살펴보고 Telnet의 배너를 변경

Banner Grabbing이란 공격 대상 시스템의 운영체제를 탐지하는 방법이다. 배너를 확인함으로서 상대방 시스템의 운영체제를 탐지할 수 있다. Telnet의 경우 원격지 시스템에 로그인하면 자동으로 출력되는데 이를 통해 대상 시스템의 정보를 얻을 수 있다.

측정기, 전화이(가) 표시된 사진

자동 생성된 설명

그림 1. 변경 전의 배너

검은색이(가) 표시된 사진

자동 생성된 설명

그림 2. 변경 후 배너

* Telnet의 로그인 시도를 tcpdump를 이용해 확인하고 아이디와 패스워드를 추출

키보드, 앉아있는, 컴퓨터, 옅은이(가) 표시된 사진

자동 생성된 설명

그림 3. tcpdump login

키보드, 앉아있는, 컴퓨터, 검은색이(가) 표시된 사진

자동 생성된 설명

키보드, 검은색, 앉아있는, 컴퓨터이(가) 표시된 사진

자동 생성된 설명키보드, 앉아있는, 테이블, 컴퓨터이(가) 표시된 사진

자동 생성된 설명**키보드, 앉아있는, 검은색, 컴퓨터이(가) 표시된 사진

자동 생성된 설명**

키보드, 앉아있는, 검은색, 컴퓨터이(가) 표시된 사진

자동 생성된 설명

앉아있는, 테이블, 키보드, 검은색이(가) 표시된 사진

자동 생성된 설명

키보드, 컴퓨터, 앉아있는, 검은색이(가) 표시된 사진

자동 생성된 설명

키보드, 검은색, 앉아있는, 컴퓨터이(가) 표시된 사진

자동 생성된 설명

검은색, 키보드, 앉아있는, 컴퓨터이(가) 표시된 사진

자동 생성된 설명

그림 4. 아이디 및 비밀번호 확인

* ARP Redirect 에 대해 설명하고 ARP Redirect를 이용한 Telnet 스니핑 시연.

ARP Redirect은 라우터의 MAC 주소를 알아내어, 공격대상에게 자신의 MAC주소가 라우터인 것처럼 소인 후, 브로드캐스트를 주기적으로 하여 패킷을 스니핑하는 공격방법이다. 좀 더 정확히 이해해보면 A가 B의 ip주소에 해당하는 MAC주소를 ARP Request로 통해 브로드캐스트하는 특성을 악용해서 reply를 공격자인 자신의 MAC 주소로 위조하여 응답한다. 이를 통해 ARP Table이 조작되어진다.

하지만 공격자가 자신이 받은 패킷을 다시 B에게 전달해 주는 작업을 하지 않으면 수동적 공격이 아닌 능동적 공격이다. 이론은 이해했으니 실습을 통해 결과를 확인해 보도록 하겠다.

먼저 스니핑 전 arp -a 명령어를 통해 맥 주소를 확인하였다.

사진, 닫기, 표지판, 빨간색이(가) 표시된 사진

자동 생성된 설명

**Ub1**

**KALI**

그림 5. arpspoof 전 MAC 주소 확인

키보드, 사진, 컴퓨터, 앉아있는이(가) 표시된 사진

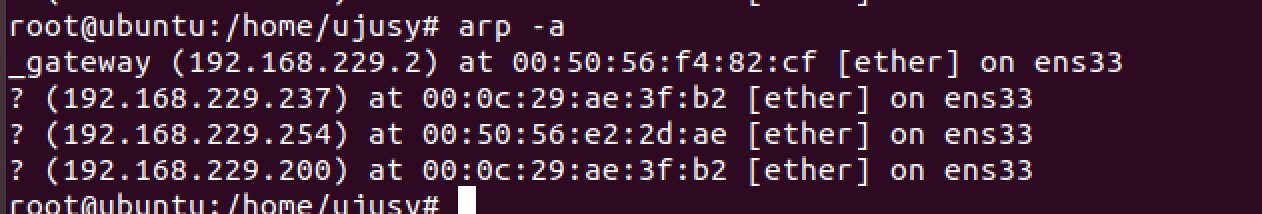
자동 생성된 설명

**Ub2**

**KALI**

그림 6. arpspoof 전 MAC 주소 확인

그림5와 그림 6에서 MAC 주소를 확인해보았다. Arpspoof 후 MAC주소를 확인해보자.



**Ub1**

**KALI**

그림 7. arpspoof 후 MAC 주소 확인

키보드, 컴퓨터, 사진, 앉아있는이(가) 표시된 사진

자동 생성된 설명

**Ub2**

**KALI**

그림 8. arpspoof 후 MAC 주소 확인

그림7과 그림 8에서 확인 할 수 있듯이 arpspoof 후에는 ubuntu 1,2의 mac주소가 kali의 mac주소와 동일하게 변하였다.

그럼 ubuntu2에서 telnet으로 1에 접속해보도록 하겠다. 1에 접속하여 id와 패스워드를 쳐서 접속하였다. 이를 tcp dump를 통해 무엇을 입력하는지 훔쳐 볼 수 있지만 dsniff을 이용했다. Dsniff는 스니핑 중에서 계정과 암호화를 위한 자동화된 도구로 실행한 모든 것들을 정리하여 보여준다.

스크린샷이(가) 표시된 사진

자동 생성된 설명

그림 9. ubuntu2에서 1로의 접속

음식이(가) 표시된 사진

자동 생성된 설명

그림 10. dsniff 를 통한 결과 확인

그림 10에서 확인할 수 있는 것처럼 telnet으로 접속하여 수행한 모든 명령어 및 아이디 패스워드 정보들을 볼 수 있다.

Telnet의 취약점을 통해 sniffing을 살펴보았다.