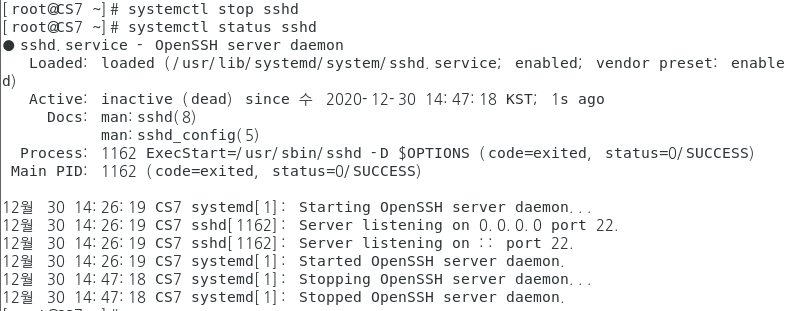
|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | |  | | KITRI 모의해킹 28기 | | | |
|  | | |  | |  | | | |
|  | | |  | |  | | | |
|  | | |  | |  | | | |
|  | | |  | |  | | | |
|  | | |  | |  | | | |
|  | | |  | |  | | | |
|  | | |  | |  | | | |
|  | | |  | 작성:김홍규 | | |  | |
|  | **Scanning** | | | | | | |  |
|  | | **-Tcp open scan-** | | | |  | | |
|  | |  | | | |  | | |
|  | | |  | |  | | | |
|  | | |  | |  | | | |
|  | | |  | |  | | | |

|  |
| --- |
| 1. 개요(공격개념/원리/취약점/공격 시 예상피해 등) |
|  |
| 1. **TCP Open Scan이란?** |
| **피해자**  **공격자**  **공격자**  **피해자**  SYN  SYN  RST+ACK  SYN+ACK  ACK  **Port close**  **Port Open**   * **TCP 스캔은 TCP 연결 시 일어나는 3-way handshaking을 탐지하는 기법**   **만약 정상적으로 완료되었다면 해당 TCP포트의 활성화 여부를 알 수 있다.**  **서비스 거부 공격을 막기 위해 포트 스캐너는 handshaking 직후 연결을 종료한다.**  **3-way handshaking 과정을 온전히 수행하여 정상적인 TCP 연결을 맺는다면 신뢰성 있는 결과를 얻을 수 있지만, 속도가 느리고 로그가 남는다는 단점이 있습니다** |
|  |
|  |
|  |
|  |
|  |

|  |
| --- |
|  |
| 2. 공격과정 (필요시 가정이나 예상 시나리오 포함) |
| |  | | --- | | 1. **실습환경** |  |  |  |  | | --- | --- | --- | |  | OS | IP | | 공격 호스트 | **Kali Linux** | 1.1.1.201/24 | | 타겟 호스트 | **CentOs7** | 1.1.1.202/24 | |
| 1. **Port Open** |
| **[타겟PC]**  **# systemctl start sshd**    **[공격자PC]**  **# nmap ① –sT ② –p 22 ③ 1.1.1.202**   1. **-sT: TCP Open Scan** 2. **-p: 포트번호** 3. **Target IP**     **CentOs(1.1.1.202) 의 22번 포트를 스캔**  **# wireshark 실행 후, Filter: ip.addr == 1.1.1.202**     * **포트가 열려있을 경우, SYN/ACK 패킷 회신 후 ACK 전송** |
| 1. **Port Close** |

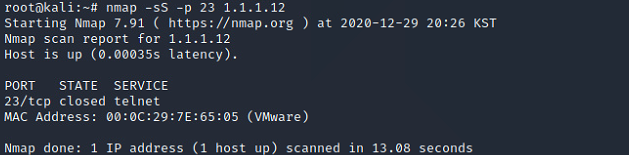
**[타겟PC]**

**# systemctl stop sshd**



**[공격자PC]**

**# nmap –sT –p 22 1.1.1.202**

****

**CentOs(1.1.1.202) 의 22번 포트를 스캔**

**# wireshark 실행 후, Filter: ip.addr == 1.1.1.202**



* **포트가 닫혀있을 경우, RST/ACK 패킷 회신**