|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | |  | | KITRI 모의해킹 28기 | | | |
|  | | |  | |  | | | |
|  | | |  | |  | | | |
|  | | |  | |  | | | |
|  | | |  | |  | | | |
|  | | |  | |  | | | |
|  | | |  | |  | | | |
|  | | |  | |  | | | |
|  | | |  |  | | |  | |
|  | 모의해킹 보고서 | | | | | | |  |
|  | | -보안팀- | | | |  | | |
|  | |  | | | |  | | |
|  | | |  | |  | | | |
|  | | |  | |  | | | |
|  | | |  | |  | | | |

|  |
| --- |
| 1. 보안팀 모의해킹 프로세스 (시나리오 기반) |
|  |
| 1. 대상 선정 (Targeting) |
| 1. 진단 대상 규모, 용도, 주요 기능 목록, 예상 취약점 목록화 2. 주요 공격 포인트 및 목표 설정, 모의침투 테스트 자산 선정 |
| 1. 정보 수집 (Scanning) |
| 1. 각종 Scan기법 등을 활용하여 대상 진단 및 구체화 2. 취약점 검색 3. 목록화 4. 우선 순위 선정 및 예상 침투 시나리오 작성 |
|  |
| 1. 모의 해킹 (PenTest) |
| 1. 식별된 취약점 별 침투 및 공격 시도  * 웹 : OWASP Top 10 (2020) 기반 웹 취약점을 중심으로 한 웹 모의해킹 * 시스템 : 버퍼 오버플로우 등의 전통적 기법부터 랜섬웨어 등의 최신 기법 등을 활용한 모의해킹 * 네트워크 : 네트워크 토폴로지를 고려하여 sniffing, dos 공격 등 다양한 해킹기법을 이용한 침투, 해킹  1. 공격 성공 시 가능한 피해의 종류 및 강도 판단 2. 공격 및 우회 실패 시 원인 분석 및 결과 정리 |
| 1. 분석 (Evaluation) |
| 1. 결과 분석 2. 원인 분석 3. 영향도 분석 |
| 1. 피드백 (Feedback) |
| 1. 식별된 취약점 및 예상되는 피해 제시 2. 보안대책 제안 |
|  |

|  |
| --- |
| 4. 해킹 시나리오 |
|  |
| 1. 웹 | |
| 해커 C는 당사의 내부망에 침입하기 위해 웹을 이용하여 정보를 수집중에 있다.   1. Injection : OWASP Top 1/10 (2020)   해커 C는 정의되지 않은 쿼리문을 삽입한 뒤 전송하고, 이후 악의적인 SQL문 삽입을 통하여 의도적인 오류를 도출한다. 반환되는 값을 통해 DB 구조 및 시스템의 취약점을 파악한다. 이후 쿠키 및 세션 등으로 타 사용자의 권한획득을 시도하며 동시에 DB서버의 사내 정보를 탈취한다.   1. Broken Authentication : OWASP Top 2/10 (2020)   해커 C는 웹 서버 관련 정보를 탐색하던 도중, 유저 인증 및 세션 관리와 관련된 기능에서 보안정책이 제대로 구현되지 않은 것을 발견하였다. 이를 활용하여 타 유저의 암호, PW 또는 세션 토큰을 의도적으로 노출시키기 위하여 BruteFroce 공격 등을 시도하여 탈취한다.   1. Sensitive Data Exposure : OWASP Top 3/10 (2020)   해커 C는 서버 정보를 수집하는 도중에 당사 웹사이트가 모든 페이지들에 대해서 TLS를 적용하지 않고 약한 암호화만을 지원하는 것을 눈치챘다. 이를 활용하여 해커C는 네트워크 트래픽을 모니터링하고 스니핑하여 사용자들의 세션 쿠키를 훔친 뒤, 타 유저들의 개인정보를 탈취하고자 한다.   1. XXE : OWASP Top 4/10 (2020)   해커 C는 XML 프로세서 관련 취약점을 활용, 서버의 Passwd파일을 탈취하거나 사설망 환경정보를 유출하고자 한다.   1. Broken Access Control : OWASP Top 5/10 (2020)   인증된 사용자들의 권한과 페이지에 대한 제한이 명확하게 구현되어 있는지 여부를 확인한 해커 C는 이부분이 취약하다는 점을 이용하여 관리자 권한에 접근 및 DB 접근을 시도한다.   1. Security Misconfiguration : OWASP Top 6/10 (2020)   보안헤더 구성, 오류 메시지 처리 등 시큐어 코딩과 같은 보안 구성이 제대로 되어 있는지 확인한 뒤 발생한 취약점을 통해 내부 정보를 접근 및 탈취한다.   1. XSS : OWASP Top 7/10 (2020)   악의적인 코드를 삽입한 스크립트를 당사 웹 페이지에 삽입한다. 이를 통해 웹 페이지를 활용하는 당사 직원들의 정보를 탈취하고, 탈취한 정보를 바탕으로 사내망 내부로 진입하고자 한다.   1. Insecure Deserialization : OWASP Top 8/10 (2020)   해커C는 사전에 작업한 악성코드를 저장한뒤, 데이터를 역직렬화하는 과정에서 발생할 수 있는 원격코드 실행, 권한 상승 취약점을 이용하여 자신의 악성코드를 실행시킨다. 이를 통하여 서버장악 및 차후 공격을 진행한다.   1. Using Components with known vulnerabilities : OWASP Top 9/10 (2020)   취약한 요소를 갖고 있는 라이브러리 및 프레임워크 사용여부를 확인하여 이를 통해 컴포넌트를 악용하여 보안수준을 약화시킨 뒤 데이터 손실을 유발하고 서버를 장악한다.   1. Insufficient logging and monitoring : OWASP Top 10/10 (2020)   위와 같은 공격을 통하여 서버를 침투하고 공격, 다양한 데이터를 탈취하였음에도, 보안관제를 통한 실시간 방어가 진행되지 않는다면, 추가적이고 지속적인 공격으로 더욱 큰 2차피해를 야기시킬 수 있다. | |
| 1. 시스템 | |
| 1. 시스템 해킹 시나리오 PLAN A  해커 A는 당사의 내부망에 침입하기 위하여 회사 홈페이지에 있는 문의접수용 회사 이메일, 문의사항 게시판 등 다양한 루트를 활용하여 지속적인 APT 공격을 시도하고 있다.   1. Macro Malware   정보를 수집하던 해커 A는 당사에서 업무 간 MS WORD와 EXCEL 파일을 업무에 주로 활용하는 것을 감안하여 사원들이 사용하는 것으로 추정되는 무작위 메일주소에 업무보고서로 위장한 Macro Malware를 삽입하여 송신하였다.   1. Trojan Horse Attack   해커 A는 정상적으로 잘 작동하는 XXX 프로그램에 트로이 목마 바이러스를 심은 후, 정보공유를 가장한 글을 작성하여 웹하드 및 일반 블로그, 당사 웹사이트 게시판에 게재하였다.   1. Ransom Ware   해커 A는 git hub에서 공개된 hidden-tear 랜섬웨어 소스를 목적에 맞게 수정하여 일반적인 pdf파일로 위장한다. 이후 당사에서 본인의 저작권법을 침해했다며 수정을 요청하는 글을 허위로 작성한 뒤 및 위반사항들을 정리한 알집파일을 첨부하여 당사에서 활용하는 이메일로 송신하였다.  내부 사원 B는 1)2)3)를 인터넷용 PC에서 다운받은 후 실행하였고, 이로 인해 해커A와의 페이로드가 활성화되어 해커의 통제 하에 놓이게 되었다. 이를 모른채로 사원 B는 업무상의 목적으로 내부망용 PC와 접점연계 프로그램을 통해 연결하였고, 해커A는 로그를 통해 접점경로를 확인한 뒤, 내부망으로 침투에 성공한다. 그 후 내부망 PC를 감염시켜 유저계정을 확보한 뒤 관리자 권한 확보를 시도한다.   1. OS Password Cracking   확보한 유저계정을 통하여 /etc/passwd 파일을 확보하였고, 관리자 id와 타 유저 계정을 확보한다. 그 후 password cracking을 통하여 관리자계정 로그인을 시도한다.   1. Race Condition Attack   내부망에 침투한 해커A는 정보를 수집한 뒤, 내부망에서 활용중인 서버나 프로그램 중 임시파일을 활용하는 취약점을 발견한다. 이를 이용하여 Race Condition Attack을 시도하여 관리자계정을 가진 유저를 생성해낸다.   1. Got Overwrite   시스템권한을 다루는 함수를 생성하여 기존 라이브러리에 overwrite한 후, 이것이 일반 유저들이나 관리자에게 호출되었을 경우 관리자 계정을 탈취해낸다.   1. Stack Overflow & RTL Attack   시스템 오버플로우를 유발하여 관리자 권한을 탈취한다. 만약 OS 버전 등으로 인하여 막혔을 경우, 이를 우회하는 RTL 공격을 시도하여 권한을 탈취해낸다.  관리자 권한을 확보한 해커 A는 추후 지속적인 공격을 위하여 백도어 작업을 진행한다.   1. Backdoor   백도어를 시스템 여러곳에 설치한다.   1. RootKit   3세대 루트킷을 생성하여 커널 기반으로 백도어를 동적메모리위치에 은폐시켜 차후 침투를 용이하게 작업한다. | |
| 2. 시스템 해킹 시나리오 PLAN B   1. Stack Overflow & RTL Attack   보안대책으로 인하여 사내 유저 계정 확보에 실패한 해커A는 협력사 직원으로 위장하여 직접 사내로 침입한 뒤 업무로 잠시 자리를 비운 사원 A의 모니터에 붙혀져있는 포스트잇에서 내부망 유저 ID와 PW를 확보하였다.  이하 PLAN A와 동일 | |
| 1. 네트워크 | |
| 해커 B는 당사 네트워크에 침입을 목적으로 정보 수집 중에 있다. 사전에 확보한 여러 희생 PC들을 경유하여 본인의 IP를 은폐한 뒤 당사 네트워크 정보 수집을 시작한다.   1. Scanning (ARP / Decoy / FIN / ICMP / NULL / TCP ACK / TCP\_HALF / TCP\_OPEN / UDP\_OPEN / X-MAS SCAN)   해커 B는 공격시도 전 당사 네트워크를 특정하기 위하여 여러 Stealth Scan 기법을 활용하여 본인의 ip를 숨긴채로 패킷들을 송신하여 당사의 네트워크 및 포트 정보를 수집한다.   1. Sniffing ( ICMP Redirect / ARP Redirect ), Session-Hijacking   해커 B는 당사의 네트워크 정보 획득을 위해 패킷 Sniffing을 시도한다. ARP 패킷은 통신을 위하여 차단할 수 없다는 점을 이용하여 ATP Redirect를 통해 목표 네트워크 망 pc들이 해커 B를 게이트웨이처럼 인식하게 하여 본인을 반드시 거쳐서 통신할 수 있도록 유도한다. 동시에 ICMP 패킷을 조작하여 송신함으로써 당사 네트워크 상에 존재하는 라우터인 것처럼 하여 패킷들의 흐름을 유도하여 내부정보를 수집한다.  해커 B는 Sniffing을 통해 얻은 정보를 분석하고 TCP Session을 확보한 뒤, 세션을 하이재킹하여 정상적인 연결이 된 일반적인 유저처럼 행동하여 추가적인 정보를 수집하고 차후 침입을 위한 여건을 조성한다.   1. Spoofing ( ARP / DHCP / IP / DNS )  * Scan과 Sniffing을 통해 당사 네트워크에서 정적설정이 없다는 것을 확인한 해커 B는 ARP와 ip 패킷을 조작하여 대상으로 하는 사내 네트워크의 패킷을 중간에서 가로채어 정보를 수집한다. * 당사의 DHCP 서버를 겨냥하여 DHCP 프로토콜자체가 UDP로 동작하는 것을 감안, 당사 네트워크의 대상 호스트가 해커 B를 DHCP 서버인 것처럼 간주하도록 ARP Spoofing으로 조작한다. 이를 통해 대상 pc를 활용하여 내부망 네트워크 정보를 수집한다. * 당사가 IT기업인 것을 감안하여, 업무 간 주로 활용될 수 있는 GOOGLE, GIT 홈페이지를 대상으로 DNS Spoofing을 통하여 대상 PC들이 해당 사이트로 접근하려고 하면 해커B가 의도한 위치로 이동되도록 유도하여 웹 공격(XSS 등) 기반을 조성한다.   해커 B는 충분한 정보를 수집한 뒤, 본격적인 공격에 앞서 서버 및 방화벽 등을 무력화하고 서비스 불가 상태를 유도하기 위하여 DoS 공격을 준비한다.   1. DoS ( Ping Of Death / Land / Smurf / Teardrop / Inconsistent-Fragmentation / ICMP Flood / UDP Flood /TCP Syn Flood / Fraggle )   준비한 모든 DoS 공격을 활용하여 사내 서비스를 중단시키고, 서버 및 방화벽의 불능상태를 유발한다. 이를 통해 해커 A와 본인의 내부 침입 조건을 달성시킨다. | |