



(주) 000

WWQ 진단 결과보고서



작성 : 취약성분석팀

2008. 04. 01

Confidentiality Agreements

1. 본 문서는 [주]O O O] 의 의뢰를 받아 실시된 취약점 진단 보안 컨설팅의 결과 산출된 보고서로써, 현재 www.000.co.kr 웹 페이지의 정보보안의 취약점 현황 부분을 설명하고 있습니다.
2. 본 문서에 포함된 취약점에 대한 어떠한 내용도, 어떠한 기본적인 개념도 제3자에게 공개 되어서는 안됩니다.
3. 본 문서의 열람은 본사의 컨설팅 연구원 및 책임자와 관계자가 문서의 열람을 허락한 최소의 인원으로 제한하여 주시기 바랍니다.

목 차

I. WWQ 진단 개요.....	4
1. 진단 목적	4
2. 진단 과정 및 방법.....	4
3. 주요 진단 항목.....	6
II. WWQ 진단 정보.....	7
1. 진단 대상 및 일정.....	7
2. 수행인력	7
III. WWQ 진단 결과 요약.....	8
1. WWQ 진단 결과 총평	8
2. WWQ 진단 결과 요약	9
3. 위험도별 보안문제점	10
4. 보안 취약점 원인 분포	10
IV. 상세 진단 결과.....	11
1. Cross Site Scripting(XSS)	11
2. SQL Injection 취약점	13
3. File Upload 취약점	15
4. File Download 취약점	17
5. CSRF.....	19
6. 부적절한 에러페이지	21
7. Cookie 취약점	23
8. 취약한 ID/PASSWORD	25
9. 주요정보 노출	26
10. Failure to Restrict URL Access.....	27
V. 결 론.....	29
APPENDIX A. NSHC의 정보보호 서비스	30
APPENDIX B. 참고 문헌 및 사이트	30

I. WWQ 진단 개요

1. 진단 목적

NSHC의 WWQ¹ 진단은 (주)000 웹사이트의 웹 취약성 정도를 파악하여 어떠한 정보를 불법적으로 획득할 수 있는지, 내부 시스템에 어느 정도까지 침투할 수 있는지, 각 진단 항목에서 발견된 취약점들이 어떻게 해킹에 이용되는지를 점검하는데 목적이 있으며, 불법 침입의 가능성이 발견될 경우, 취약한 부분에 대한 대응책 및 개선 방안을 마련하여 고객사의 보안사고를 미연에 방지하는 것을 그 목표로 하고 있습니다.

2. 진단 과정 및 방법



[그림 1] 진단 과정

NSHC의 WWQ 진단은,

1st. 신청한 URL을 자동화 툴로 점검

2nd. 자동화툴의 결과값을 참조하여,

WWQ 진단 인력의 수동점검(외부자 관점 및 내부자 관점)

3rd. 점검된 결과값을 분석하여 보고서 작성

이후 보고서를 토대로 한 보안 이행작업² 작업 후 재 점검을 통해 모든 취약점을 제거한 후 최종 산출물이 발송됩니다.

¹ NSHC에서 최초로 도입한 새로운 개념의 차별화된 취약성 진단 서비스인 WWQ 는 Web Well-being Quotient의 약자로 “웹건강지수”를 의미하며 귀사 웹사이트의 건강상태를 객관적으로 알아볼 수 있습니다.

² WWQ 진단시 도출된 취약점을 제거하는 작업으로 이는 WWQ 서비스에 포함되지 않으며 요청시 별도 과금됩니다.

2.1 외부자 관점

외부자 관점 취약성 진단은 점검 대상 사이트의 URL 및 IP 정보만을 알고 있다는 가정하에 외부 침입자의 관점에서 진단을 수행합니다.

시스템용도	IP Address	외부 URL
Web Server	xxx.xxx.xxx.xxx	http://www.000.co.kr

[표 1] 외부자 관점 대상정보

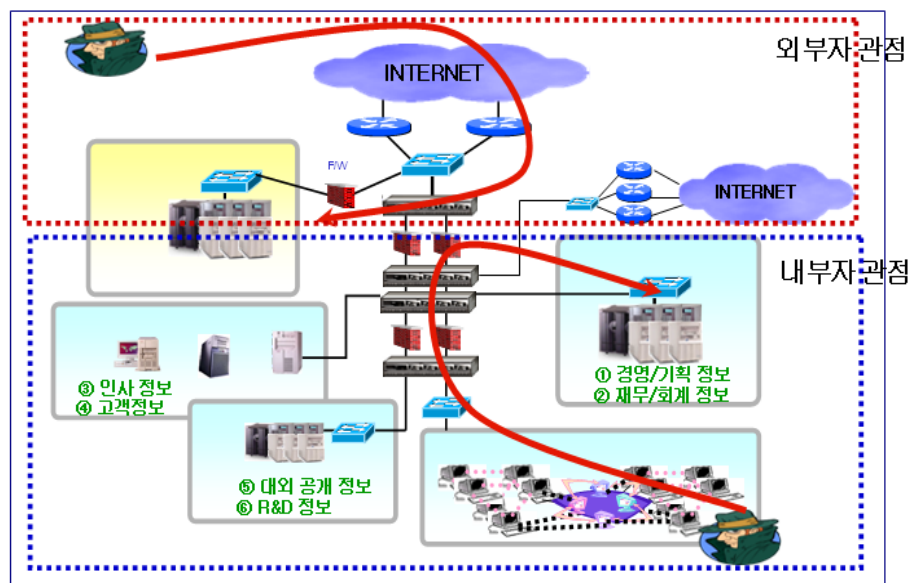
2.2 내부자 관점

내부자 관점 취약성 진단은 점검 대상 사이트의 직원 또는 관리자 ID 정보를 알고 있다는 가정하에 내부 직원 및 관리자의 관점에서 진단을 수행합니다.

내부자 ID	IP Address	내부 URL
진단용 내부자 ID	xxx.xxx.xxx.xxx	http://xx.xx.xx.xx

[표 2] 내부자 관점 대상정보

2.3 외부자 / 내부자 관점 구성



[그림 2] 외부자/내부자 관점 구성

3. 주요 진단 항목

WWQ 진단항목은 2007년 OWASP(Open Web Application Security Project)에서 지정한 10대 웹어플리케이션 취약점과 국정원 8대 취약점을 이용한 공격을 병행하여 수행하며, 각 항목별로 점검 수행할 내역들은 다음과 같습니다.

취약점 진단항목		위험지수	세부 진단 내용
국 정 원 8 대 취 약 점	WebDav 취약점	낮음	자동화 점검툴로 WebDAV 사용여부 스캔
	테크노트 취약점	중간	테크노트 프로그램 버전확인
	제로보드 취약점	중간	제로보드 프로그램 버전확인
	XSS 취약점	높음	글쓰기가 가능한 게시판에 간단한 스크립트 명령을 삽입하여 점검
	SQL Injection 취약점	높음	로그인 페이지에 SQL Injection 공격으로 권한 획득 또는 SQL 구문 에러
	파일업로드 취약점	높음	파일업로드 페이지에 로그인 없이 파일 업로드 시도
	파일다운로드 취약점	중간	외부자 관점에서 파일 다운로드 여부 확인
	디렉터리 노출	높음	URL 강제 접속을 통해 디렉터리 리스팅 시도
	부적절한 에러페이지	낮음	에러페이지를 유도하여 어플리케이션 정보가 노출되는지 점검
	CSRF 취약점	높음	Cross Site Script 공격을 통한 정보노출
O W A S P T O P 10	Cookie 취약점	높음	쿠키정보가 노출되는지 확인, 쿠키값을 조작하여 로그인시도
	취약한 ID/PW	높음	취약한 ID/Password 무차별 대입 공격
	데이터 암호화	높음	사용자 패스워드 등의 민감한 정보를 암호화 하여 저장하는지 점검
	주요정보 노출	중간	데이터 송수신시 주요 정보가 암호화가 되었는지 평문으로 전달되는지 점검
	URL 변조 및 강제접속	중간	URL 파라메타 값을 변조하여 다른 페이지로 직접적인 접근이 가능한지 점검

[표 3] 주요 진단 항목

II. WWQ 진단 정보

1. 진단 대상 및 일정

- WWQ 진단 일정 : 2008년 3월 10일(월) ~ 2008년 3월 14일(목)
- 3월 10일 ~ 3월 10일 (1Day) : 진단대상 분석 및 계획 수립
- 3월 11일 ~ 3월 14일 (4Days) : 취약성 분석 수행 및 보고서 작성

(WWQ 프로세스 완료)

- 3월 17일 ~ 3월 18일 (2Days) : 이행작업 단계 (요청시 별도 과금)
- 3월 19일 : 최종 산출물 제출

단계	세부작업	3 월							
		10	11	12	13	14	17	18	19
분석계획	대상선정 및 계획수립								
	정보 수집								
분석수행	취약성 진단								
	취약성 분석								
	분석 보고서 작성								
이행단계	보안 이행 작업								
	최종 산출물 제출						평가결과협의		

[표 4] 수행 일정

2. 수행인력

본 WWQ 진단 수행 업무는 NSHC사의 모의해킹 전담인원에 의해 수행되며 진단 시 투입 인력은 다음과 같습니다.

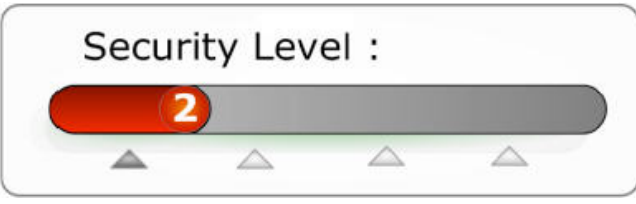
성명	소속	직급	담당업무
권혁진	취약성분석팀	주임연구원	WWQ 진단 총책임
김경수	취약성분석팀	연구원	취약성진단, 보고서작성
박용운	취약성분석팀	연구원	취약성진단, 보고서작성

[표 5] 수행 인력

III. WWQ 진단 결과 요약

1. WWQ 진단 결과 총평

귀사 정보보호 시스템의 웹 건강지수는 **24.45점**으로 기업 웹 건강지수의 평균 수치(51.24점) 보다 매우 낮아 위험한 상태에 놓여 있는 것으로 판단되었습니다.

점검 결과	Score	그래프
웹 건강지수 (WWQ)	24.45점	
기업평균 건강지수	51.24점	

[표 6] <http://www.ooo.co.kr>의 WWQ 점수

➤ 취약성 평가 점수 산정 방법

위험 평가 점수 산정 방법 $WWQ\ Point = (취약성\ 항목 * 위험\ 등급) / 페이지\ 수$

시스템의 위험 평가 등급은 각 대상 서버를 진단한 이후 결과를 분석 하여, 위험도 (Risk Level)에 따른 취약점 지수(Vulnerability value)를 WWQ 단계 구별에 따라 구분하여 연산한다. 이후 대상 시스템의 전체 웹 Page수와 연산하여 최종 WWQ 점수를 산정한다.

WWQ의 상세 점수 산정 방법론은 저작권 관련 문제로 공개하지 않는다.

[표 7] 취약성 평가 점수 산정 방법

2. WWQ 진단 결과 요약

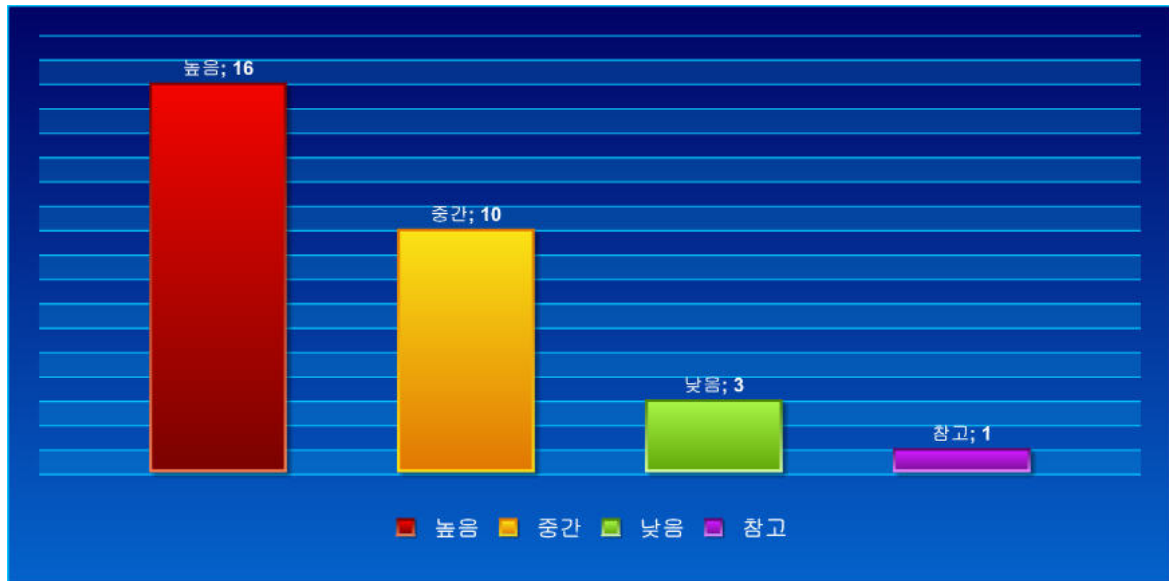
총 31개의 취약한 URL이 탐색되었고, 이 중 위험도 높은 취약점이 17개, 위험도 중간 취약점이 10개, 위험도 낮은 취약점이 3개, 참고 위험도 취약점이 1개 발견되었습니다.

구분	(주) 0 0 0 웹 사이트			
URL	http://www.ooo.co.kr	IP Address	xxx.xxx.xxx.xxx	
용도	쇼핑몰 웹서비스	운영체제	Linux Kernel 2.6	
웹서버	Apache 2.0	개발언어	PHP	
항목별 취약점 진단 결과				
진단 항목	위험도	진단결과	취약점 수치	대책
XSS 취약점	높음	취약	4	보안지침 1
SQL Injection 취약점	높음	취약	2	보안지침 2
파일업로드 취약점	높음	취약	3	보안지침 3
파일다운로드 취약점	중간	취약	10	보안지침 4
디렉터리 노출	높음	취약	3	보안지침 5
CSRF 취약점	높음	취약	2	보안지침 6
Cookie 취약점	높음	취약	1	보안지침 7
취약한 ID/PW	높음	취약	1	보안지침 8
정보유출 및 부적절한 오류처리	낮음	취약	3	보안지침 9
WebDav 취약점	낮음	N/A	0	보안지침 10
테크노트 취약점	중간	N/A	0	보안지침 11
제로보드 취약점	중간	N/A	0	보안지침 12

[표 8] 진단 결과 요약

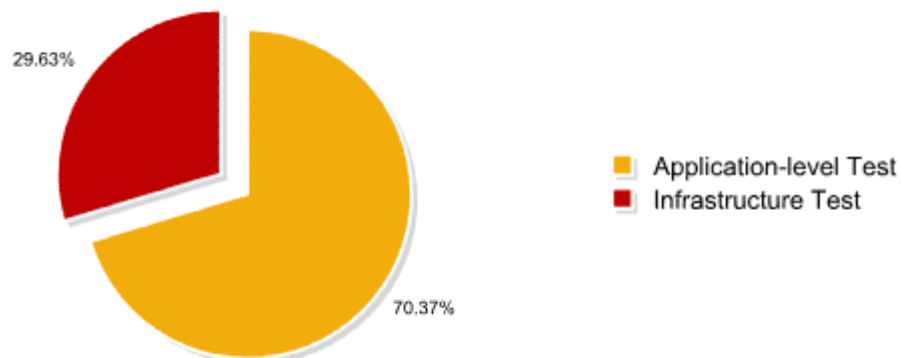


3. 위험도별 보안문제점



[그림 3] 위험도별 보안문제점

4. 보안 취약점 원인 분포



[그림 4] 보안취약점 원인분포

- 어플리케이션 관련 보안 취약점 - 개발자 수정
- 인프라스트럭처 및 플랫폼 보안 취약점 - 시스템 및 네트워크 관리자 수정

IV. 상세 진단 결과

1. Cross Site Scripting(XSS)

1.1 진단 개요

발생원인	사용자의 입력을 받아 처리하는 웹 응용프로그램에서 입력 내용에 대해 실행코드인 스크립트의 태크를 적절히 필터링 하지 않아 발생함
위험도	높음
분류	개발자 수정
취약점 점검방법	- 웹페이지 게시판에 일반사용자들이 글을 게시할 수 있는지 점검한다. - 웹페이지 글쓰기 기능이 존재하는 경우 간단한 스크립트를 게시했을 때 스크립트문이 적용되었을 때 취약점이 성립된다. - 게시물 스크립트 삽입 <pre><script>alert("XSS TEST")</script></pre> <pre><iframe>삽입</pre>
영향	- 악의적인 스크립트가 포함된 게시물을 등록할 수 있어 해당 게시물을 열람하는 일반 사용자의 PC로부터 개인정보이 쿠키를 유출할 수 있는 등의 피해를 초래할 수 있음 - Cookie sniffing, session hijacking 가능 - 악의적인 콘텐츠 삽입 가능 - 웹 사이트 변조 가능 - 피싱 침투 행위 가능 - CSRF 위험
영향받는 URL	/board/ - http://192.168.0.100/board/board.php?board=service(고객의소리)
조치방법	- 글쓰기가 가능한 게시판 페이지에 사용자들의 input 스크립트를 모두 필터링 한다. - 스크립트 문장에 사용되는 메타캐릭터가 포함되어 있을 때 이를 변환시키는 필터링을 수행한다.

1.2 진단 결과

XSS 취약점

■ 진단 결과 : 취약

고객게시판 페이지에 <iframe> 스크립트를 삽입하여 게시판 변조가 가능하였으며, <META> 태그를 이용하여 불특정 다수의 사용자를 공격자의 홈페이지로 강제 이동시켜 제2의 피해를 줄 수 있었습니다.

Step 1) 고객게시판 제안마당 페이지 게시판에 iframe 공격으로 페이지 변조를 시도

The screenshot shows a web application security tool interface. The 'Payload' field contains the following HTML code: `<META http-equiv='refresh' content='3; url=http://www.phrack.org/issues.html'>`. The tool also shows fields for 'URL', 'E-Mail', and 'Homepage'.

[그림] iframe 스크립트 삽입

Step 2) Iframe 공격을 통해 고객게시판을 변조시켰으며, <META>태그를 통해 스파이웨어, 봇넷 등 악성스크립트에 감염될 위험이 있는 공격자의 홈페이지에 강제 접속도 가능



[그림] 고객게시판 변조

시연동영상

■ XSS 시연 동영상

<mms://stream2.nshc.net/nshc/01.xss.wmv>

2. SQL Injection 취약점

2.1 진단 개요

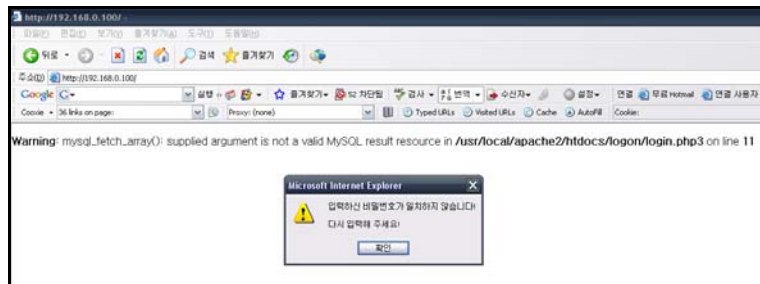
발생원인	웹브라우저 주소(URL)창 또는 사용자 ID 및 패스워드 입력화면 등에서 데이터베이스 SQL문에 사용되는 문자기호(' 및 ")의 입력을 적절히 필터링하지 않아 발생함
위험도	높음
분류	개발자 수정
취약점 점검방법	로그인 페이지에 SQL 쿼리를 삽입하여 테스트 한다. ' or '1'='1 "or 1=1-- "or 1=1-- or 1=1-- 'or 'a'='a "or "a"="a ') or '(a'='a sq'l or 1=1— sq"l or 1=1-- +or 1=1-- ';--
영향	- SQL문으로 해석될 수 있도록 조작한 입력으로 데이터 베이스를 인증 절차없이 접근, 자료를 무단 유출하거나 변조할 수 있다. - 악성스크립트 실행 가능 - 외부프로그램 사용가능 - DB정보 열람 및 조작 가능 - 프로시저를 통한 OS명령 실행 가능 - 웹 어플리케이션을 통해 다른 시스템 공격 가능
영향받는 URL	/logon/ - http://192.168.0.100/logon/login.php3 (로그인 페이지)
조치방법	ID 및 패스워드란에 특수문자를 쓸 수 없도록 소스코드 수정을 수정한다.

2.2 진단 결과

진단 결과 : 취약

사용자 로그인 입력폼(<http://192.168.0.100/login/login.php3>)에서 SQL 쿼리문을 삽입시 SQL 에러문 노출로 SQL Injeciton 공격의 가능성을 확인하였으며, 간단한 SQL 쿼리문으로 사용자 계정획득에 성공하였습니다

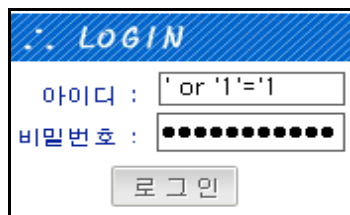
Step 1) /login/login.php3 로그인폼에 SQL 싱글쿼터(') 입력시 SQL 에러가 발생하였으며, SQL 구문 에러를 통해 SQL Injection 공격 가능성을 확인함



[그림] SQL 에러페이지 노출

SQL Injection

Step 2) SQL Injection 패턴 공격으로 로그인 시도



[그림] SQL Injection 공격시도

Step 3) SQL Injection 패턴 공격성공으로 일반계정 획득 성공



[그림] 고객계시판 변조

시연동영상

SQL Injection 시연 동영상

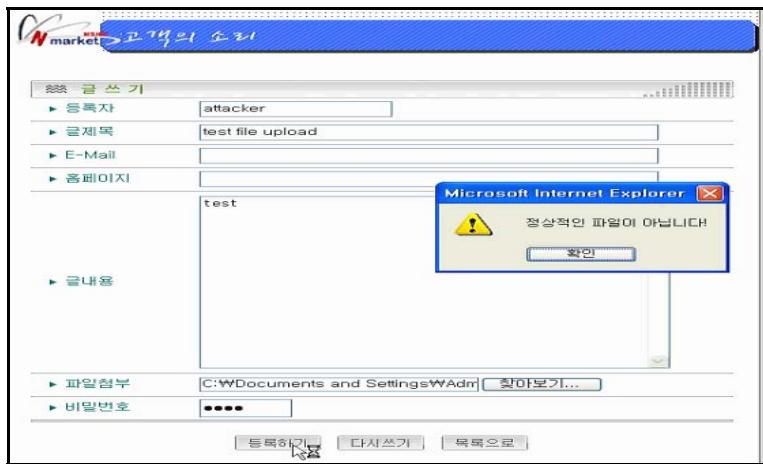
- mms://stream2.nshc.net/nshc/02.SQL_Injection.wmv

3. File Upload 취약점

3.1 진단 개요

발생원인	플랫폼 상에서 프레임워크는 URL 이나 파일 시스템 참조와 같은 외부 객체 참조사용을 허용하여 발생함
위험도	높음
분류	개발자 수정
취약점 점검방법	Php, asp, jsp 등의 확장자를 가진 파일들을 업로드한다.
영향	홈페이지 게시판에 .php, .jsp등의 확장자 이름의 스크립트 파일의 업로드를 허용할 경우 악성 웹쉘을 통해 시스템 권한이 가능하다.
영향받는 URL	/board/ - http://192.168.0.100/board/board.php?board=service(고객의소리)
조치방법	게시판 첨부파일 업로드를 처리하는 웹소스코드에서 첨부파일의 확장자를 보고 필터링 한다.

3.2 진단 결과

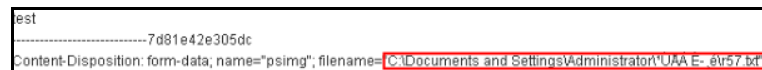
파일업로드	진단 결과 : 취약 고객의소리 게시판에 웹쉘파일을 업로드하여, 시스템권한 획득에 성공하였습니다 Step 1) 고객의소리 게시판에 웹쉘 파일을 업로드를 시도하였으나, 확장자 검증을 하고있어 파일업로드에 실패  [그림] 웹쉘 업로드
-------	---

Step 2) 확장자를 우회하여 파일업로드를 시도하였으며, php->txt파일로 변환하여 웹shell을 업로드하는데 성공

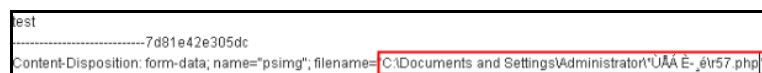


[그림] 웹shell파일 업로드 성공

Step 3) 해당 확장자 검사가 클라이언트 단에서 이루어 졌으며, 프록시 툴을 통해 확장자 검사를 간단히 우회하여 파일업로드 시도

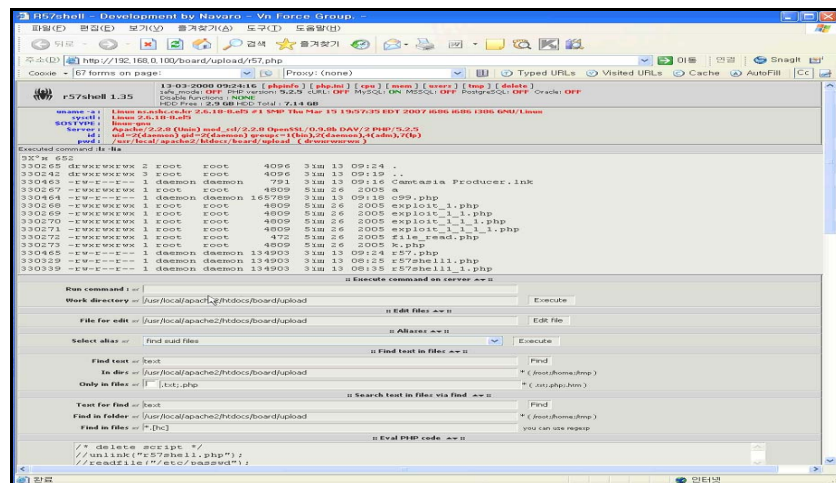


[그림] 확장자 검증 우회1



[그림] 확장자 검증 우회2

Step 4) 업로드된 파일을 웹shell파일을 실행하여 시스템권한을 획득



[그림] 웹shell을 이용한 시스템 권한 획득

시연 동영상

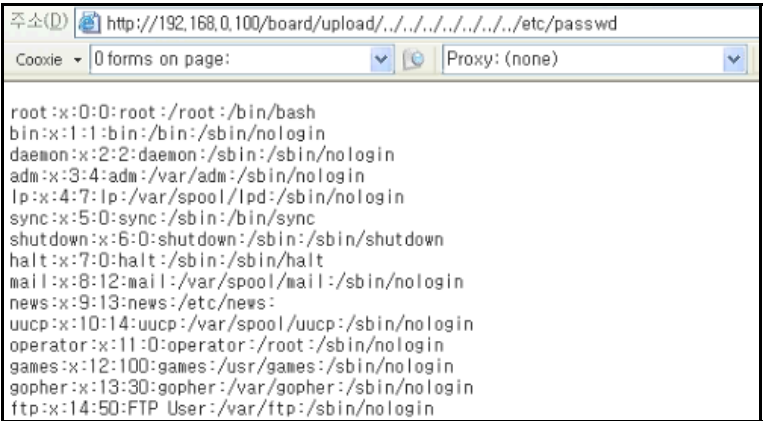
- 파일업로드 공격 시연 동영상
- [mms://stream2.nshc.net/nshc/03.fileupload.wmv](https://stream2.nshc.net/nshc/03.fileupload.wmv)

4. File Download 취약점

4.1 진단 개요

발생원인	직접적인 객체 참조는 개발자가 파일, 디렉토리, 데이터베이스, 레코드나 키와 같이 내부적으로 구현된 객체를 URL 이나 폼 파라미터 형식으로 노출시킬 때 발생
위험도	중간
분류	시스템 및 네트워크 관리자 수정
취약점 점검방법	- 파일 다운로드 스크립트 이용 여부 확인 - 다운로드 스크립트의 매개변수를 변경하면서 주요파일 다운로드 시도
영향	- 다운로드 스크립트를 이용하여 시스템 주요 정보 노출 /etc/passwd - 웹응용프로그램 소스코드 노출
영향받는 URL	/board/ - http://192.168.0.100/board/upload(자료실) 생략
조치방법	다운로드 스크립트 “..”, “/”, “\”와 같은 문자열 필터링

4.2 진단 결과

파일다운로드	진단 결과 : 취약 자료실에 첨부된 파일을 다운로드 받을 때 파일 명에 대하여 필터링이 되어 있지 않을 경우 다음 그림과 같이 파일 명 부분에 './' 와 같은 문자열을 입력하여 상위 디렉터리로 이동할 수 있으며, 이를 통해 시스템 상에 존재하는 /etc/passwd를 다운로드 받을 수 있었습니다. - /etc/passwd/ 파일을 열람하였으며, 노출된 정보로 burte force공격도 가능  [그림] /etc/passwd 파일 다운로드 생략
시연동영상	- 파일다운로드 시연 동영상 mms://stream2.nshc.net/nshc/04.filedownload.wmv

5. CSRF

5.1 진단 개요

발생원인	CSRF 는 XSS 취약점이 존재하는 하에, 근본적으로 정상적인 request 와 비정상적인 request 를 구분하지 못해 발생함
위험도	높음
분류	개발자 수정
취약점 점검방법	- 웹페이지 게시판에 일반사용자들이 글을 게시할 수 있는지 점검한다 - 웹페이지 글쓰기 기능이 존재하는 경우 간단한 스크립트를 게시했을 때 스크립트문이 적용되었을 때 취약점이 성립된다 - 게시물 스크립트 삽입 <pre><script>alert("XSS TEST")</script></pre> <pre><iframe>삽입</pre>
영향	- 악의적인 스크립트가 포함된 게시물을 등록할 수 있어 해당 게시물을 열람하는 일반 사용자의 PC로부터 개인정보 및 쿠키를 유출할 수 있는 등의 피해를 초래할 수 있음 - Cookie sniffing, session hijacking 가능 - 악의적인 콘텐츠 삽입 가능 - 웹 사이트 변조 가능 - 피싱 침투 행위 가능
영향받는 URL	/board/ - http://192.168.0.100/board/board.php?board=service(고객의소리) . . . - 생략 . .
조치방법	- 글쓰기가 가능한 게시판 페이지에 사용자들의 input 스크립트를 모두 필터링 한다 - 스크립트 문장에 사용되는 메타캐릭터가 포함되어 있을 때 이를 변환시키는 필터링을 수행한다

5.2 진단 결과

CSRF

진단 결과 : 취약

CSRF 취약점이 존재하는 경우 취약게시판에 악성 스크립트를 삽입하여 해당 쇼핑몰의 물품을 인증없이 구매할 수 있었습니다.

Step 1) 고객게시판에 인증없이 물품을 구입할 수 있는 스크립트를 삽입하여 CSRF공격을 유도

The screenshot shows the '고객의 소리' (Customer Voice) board on the N market website. The '내용' (Content) field contains a malicious script designed to perform a CSRF attack on the order modification page. The script uses a form action to the order modify page with parameters for order number, page, and a malicious payload.

[그림] CSRF 스크립트 삽입

Step 2) 관리자가 악성스크립트 게시물을 열람하면서 인증 절차를 거치지 않고 구매 완료

The screenshot shows the '주문관리' (Order Management) page in the N market admin interface. It displays a list of orders, including one placed by the user 'attacker' for 43,000 won. The order status is '주문완료' (Order Completed), indicating the successful execution of the CSRF attack.

[그림] 공격자 물품구입 성공

시연동영상

CSRF 시연 동영상

- <mms://stream2.nshc.net/nshc/05.CSRF.wmv>

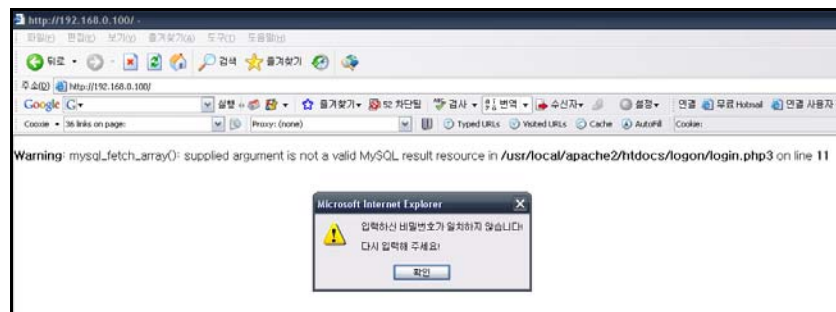
6.2 진단 결과

부적절한
에러페이지

■ 진단 결과 : 취약

공격자는 SQL 삽입 공격의 한 방법으로 비정상적인 입력을 통한 오류 메시지를 통해 해당서버의 데이터베이스 정보를 획득할 수 있습니다. 에러유도를 통한 메시지 반환은 취약한 서버의 경우 공격자로 하여금 해당 서버의 대부분 정보(절대경로, 시스템코드, 처리되지 않은 예외 구문)를 획득할 수 있도록 합니다.

- SQL 쿼리 오류를 유도하여 에러페이지 노출



[그림] SQL 쿼리오류발생

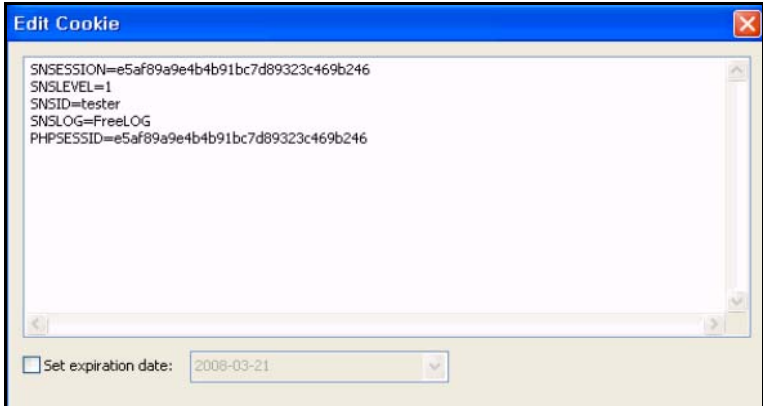
생략

7. Cookie 취약점

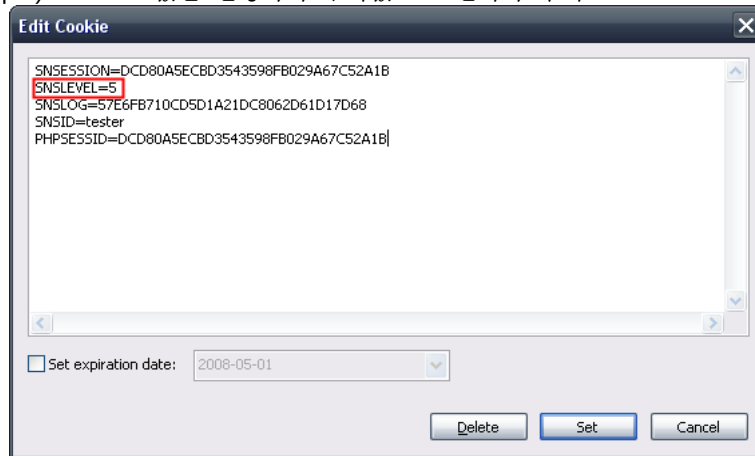
7.1 진단 개요

발생원인	세션 및 토큰 보호 실패에 의한 발생함
위험도	높음
분류	개발자 수정
취약점 점검방법	코드 점검 및 테스트를 복합적으로 사용하는 것을 인증, 세션관리, 부수적인 기능들이 모두 적절하게 구현되어 있는지 점검함
영향	- 사용자나 관리자 계정을 가로챌수 있음 - 권한 및 책임 추구성 관리를 약화시킴 - 개인정보 침해를 유발함
영향받는 URL	/main/ - http://192.168.0.100/main/main.html(메인페이지)
조치방법	웹인증시 사용되는 쿠키 저장시 주요정보는 암호화하여 저장

7.2 진단 결과

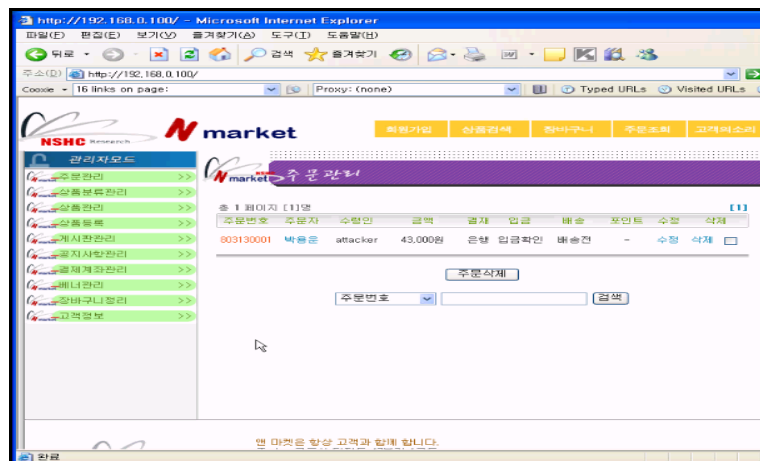
Cookie 취약점	진단 결과 : 취약 쿠키정보가 노출되고 있으며, 노출된 쿠키정보로 타 사용자 정보 노출 및 관리자 권한으로 변조가 가능합니다. Step 1) 주요쿠키정보가 노출되고 있으며, 의심되는 SNSLEVEL값 변경 시도  [그림] 사용자 쿠키정보 노출
------------	---

Step 2) SNSLEVEL값을 변경하여 쿠키값으로 관리자 우회



[그림] 사용자 쿠키정보 노출

Step 3) 사용자 쿠키정보 변조로 관리자 권한 획득



[그림] 관리자 페이지

시연동영상

- Cookie노출 시연 동영상
 - [mms://stream2.nshc.net/nshc/06_07.Cookie sniffing spoofing.wmv](mms://stream2.nshc.net/nshc/06_07.Cookie_sniffing_spoofing.wmv)

8. 취약한 ID/PASSWORD

8.1 진단 개요

발생원인	- 민감한 데이터를 암호화 하지 않음 - 자체 제작 알고리즘 사용 - 강력한 알고리즘의 불안정한 사용 - 취약한 알고리즘의 지속적인 사용(MD5,SHA-1,RC3,RC4 등...) - 하드 코딩된 키와 안전하지 못한 영역에 키 저장
위험도	높음
분류	개발자 수정
취약점 점검방법	- 패스워드나 주민등록 번호등 중요한 데이터가 암호화 되어 저장되어 있는지 점검 - DB 파일을 대상 사이트에서 취득하여 정보를 조회. - 취약한 ID/패스워드 점검
영향	- 패스워드나 신용카드 정보와 같이 민감한 정보를 암호화 하여 저장하는지 확인 - 부적절한 암호화 알고리즘을 사용하는지 점검 - 암호화에 사용되는 키나, 인증서, 비밀번호를 안전하지 않은 장소에 저장하는지 확인
영향받는 URL	/login/ - http://192.168.0.100/login/login.php(로그인폼)
조치방법	강력한 패스워드 정책 수립

8.2 진단 결과

취약한 ID/Password	- 진단 결과 : 취약 - 취약한 ID/패스워드가 존재하며, 취약한 계정의 개인정보 노출이 가능합니다 - 취약한 관리자 계정이 존재, brute force 공격으로 시스템 권한 획득이 가능 취약한 웹어플리케이션 관리자 아이디 : admin / admin1234
시연동영상	- 취약한 ID/PW 시연 동영상 mms://stream2.nshc.net/nshc/08.ID_Password.wmv

9. 주요정보 노출

9.1 진단 개요

발생원인	클라이언트와 애플리케이션 간의 네트워크 통신 과정에서 암호화되지 않은 방식으로 통신을 하기 때문에 발생
위험도	중간
분류	개발자 수정
취약점 점검방법	HTTP 패킷이 암호화되어 통신하는지 여부를 확인
영향	개인 정보 획득 HTTP 패킷 이 암호화 되지 않은 상태로 통신이 이루어 진다면 계정 혹은 패스워드가 평문으로 전송되므로 침투자는 스니핑을 통해 손쉽게 개인 정보 획득 가능
영향받는 URL	/main/ http://192.168.0.100/main/main.html(메인페이지)
조치방법	암호화 알고리즘을 이용

9.2 진단 결과

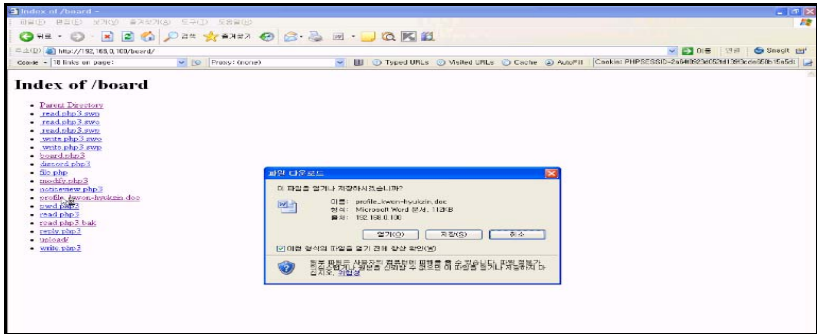
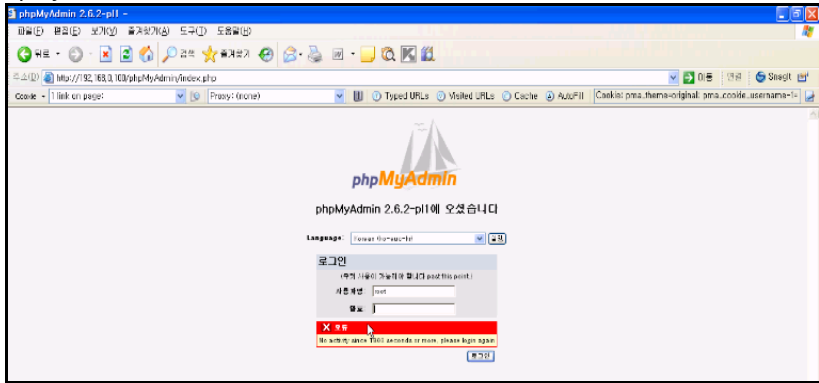
주요정보 노출	진단 결과 : 취약 암호화 되지 않은 전송방식으로 소스코드상에 사용자 패스워드가 노출 되는 취약점이 발견되었습니다 - 소스코드상에 사용자 패스워드가 노출 <pre> <td width="470"></td> <td> <table height="65" cellSpacing="0" cellPadding="0" border="0"> <form name="theForm" action="" method="post"> <tr> <td align="right" width="56" height="32"></td> <td align="absMiddle" width="122"><input class="login" style="WIDTH: 111px; HEIGHT: 23px" tabIndex="1" size="15" name="user_id" value="onsadmingj"></td> <td rowspan="2" width="2" valign="bottom"></td> </tr> <tr> <td align="right" height="20"></td> <td align="absMiddle" size="15" name="passwd" onkeydown="if (event.keyCode==13) login();" value="1111"></td> </tr> </table></td> </tr> </form> </table></td> </pre> [그림] 노출된 패스워드
시연동영상	- 주요정보노출 시연 동영상 - mms://stream2.nshc.net/nshc/09.주요정보노출.wmv

10. Failure to Restrict URL Access

10.1 진단 개요

발생원인	어떤 권한이 부여된 자원을 접근 또는 사용하기 위해서 인증이란 절차를 거쳐야 함. 보안적 관점에서 매우 중요한 것으로 잘못된 어플리케이션 설계로 인해서 이러한 과정이 생략되는 경우가 종종 있기 때문에 발생
위험도	중간
분류	개발자 수정
취약점 점검방법	침투자에게 권한이 없는 URL 주소를 직접 입력해서 데이터 조작이 가능한지 여부를 확인
영향	- 운 좋은 침투자는 권한 없는 페이지를 찾아 접근하여 기능을 이용하거나 데이터를 볼 수 있음 - 관리자 페이지를 침투 성공했다면 관리자 페이지를 관리자의 권한으로 보고 데이터를 조작 가능
영향받는 URL	/board/ - http://192.168.0.100/board/upload(자료실) /admin/ - http://192.168.0.100/admin/admin.php(메인페이지)
조치방법	- 사용자의 민감한 기능들에 대한 요청을 승인하기 전에 접근 제어를 반드시 수행하여 사용자가 그 기능을 접근하기 위해서는 지속적인 접근 통제를 받도록 해야 함 - 각각의 페이지에 대한 접근 통제를 하도록 해야함 - Include/libraty 파일을 주의 해야 하고 .asp, .exe 와 같이 실행 가능한 확장자를 가진 경우 웹 루트가 아닌 외부에 파일을 두어야 함 - 어플리케이션이 제공하지 않는 모든 파일 형태에 대한 접근을 차단하고 대상 어플리케이션에서 제공하는 html, asp 등 서비스 하려는 파일 형태만 허용해야함

10.2 진단 결과

디렉토리 노출	진단 결과 : 취약 URL 강제접속을 통해 디렉토리 구조가 노출되며, 디렉토리 노출을 통해 주요파일 다운로드 및 절대경로가 노출되는 취약점이 발견되었습니다 - URL 강제접속으로 디렉토리 리스팅 취약점 발견  [그림] 디렉토리 리스팅
불필요 페이지 존재	진단 결과 : 취약 URL 강제접속을 통해 php관리자 페이지가 노출되었으며, 관리자 권한 획득을 시도 할 수 있는 또다른 공격에 노출되어 있습니다. - Phpmyadmin 페이지 존재  [그림] phpmyadmin노출
시연동영상	- 디렉토리 노출 시연 동영상 mms://stream2.nshc.net/nshc/10.디렉토리노출.wmv - URL 강제 변조 시연 동영상 mms://stream2.nshc.net/nshc/11.URL변조.wmv - 불필요페이지 존재에 따른 해킹 시연 동영상 mms://stream2.nshc.net/nshc/12.불필요페이지.wmv

V. 결 론

NSHC는 귀사 웹사이트의 웹건강지수를 알아보고 이에 따른 보안성과 안전성을 확보 하기 위하여 보안전문가가 솔루션과 모의 해킹을 통하여 종합적이고 체계적인 보안 진단을 수행하였습니다.

WWQ 진단을 통한 취약성 분석 결과, 외부 망에서 (주)000의 공인IP에 대한 모의침투시 NAT를 통한 보안 설정 및 불필요한 포트 취약점에 대한 대응이 비교적 잘 되어 있었으나, 다음과 같은 취약점을 찾아볼 수 있었습니다.

- 1) 방문자 웹 페이지의 로그인 및 패스워드 입력 창을 통한 특수문자 입력의 필터링이 이루어지지 않아, XSS(Cross-Site Scripting)에 대한 취약점이 발견되었으며, 관리자 페이지가 노출
- 2) 관리자 페이지는 로그인 페이지보다 XSS 공격에 훨씬 취약한 것으로 분석 되었으며 특수 문자를 통한 관리자 권한 획득 가능
- 3) 일부 게시판 업로드 파일에 대한 적절한 필터링 규칙이 존재하지 않음.
- 4) 페이지 전반적으로 사용자의 입력 값에 대한 검증을 자바스크립트(Client Side Script) 를 사용하여 공격자가 얼마든지 페이지를 및 입력 값을 변조 가능
- 5) 내부 망을 통한 서버모의침투 결과 불필요한 SMB 및 FTP 서버가 존재하며, 패스워드가 취약한 administrator 계정 사용
- 6) 전송 과정에서 평문으로 전달 되어 스니핑을 통한 ID 및 패스워드 획득이 가능하여 계정 획득 및 회사 정보, 고객 정보들에 대한 접근 가능

크게 위와 같은 취약점을 발견할 수 있으며, 안전한 정보 보호 시스템 구축 방안으로 우선, 결과보고서와 함께 제공되는 WWQ 보안이행지침서를 참고하여 신속하게 취약성을 제거 할 것을 권고합니다. 또한 웹의 특성상 웹상의 데이터들은 계속적으로 변화하고 증가하고 있기 때문에, 안전 상태 유지하여 신뢰성 있고 안정적인 웹 서버를 운영하기 위해서는 주기적인 점검을 통한 지속적인 관리가 필요하다고 판단됩니다.

APPENDIX A. NSHC의 정보보호 서비스



APPENDIX B. 참고 문헌 및 사이트

- OWASP and OWASP Top 10(2007 Update)
- 황순일 김광진, “웹 해킹 패턴과 대응”, SciTech 미디어
- OWASP Testing Guide
- OWASP Web Application Penetration Test List
- XSS Cheat Sheet : <http://ha.ckers.org/>
- Web Programming Tutorials : <http://www.w3schools.com/>
- OWASP : http://www.owasp.org/index.php/Main_Page