웹 서버 구축 보안점검 가이드

2011. 01

학술정보처 정보시스템팀

제•개정 이력

순번	제•개정일	변경내용	발간팀	연락처
1	2011.01	제정	정보시스템팀	054)279-2514

목 차

그림 목차

제 1 장 개요

대부분의 침해사고들은 보안 시스템 부족 등의 1차적인 문제가 아닌, 부정확한 현황 파악과 보안 담당자의 시스템 관리 소홀로 인해 발생하는 경우가 대부분이며, 이는 관리자와 보안 담당자의 보안인식 부족에 기인한다.

최근의 침해사고 유형은 단순히 한 가지의 기법만이 이용되는 것이 아니라 시스템 취약점과 더불어 네트워크의 구성을 활용하는 등의 다양한 기법이 사용되고 있어, 사고 예방을 위해서는 최초 시스템을 구축하는 단계에서부터 보안을 고려하는 것이 점차 중요해지고 있다.

본 가이드에서는 교내에서 운영 중인 웹서버에서 서비스 환경을 구축할 때 적용해야 하는 최소한의 보안 설정과 취약점 점검 항목을 다루고자 하였다. 시스템의 최초 설치 시 필요한 기본적인 절차와 항목들을 체크리스트 형태로 제시하여 교내 웹서버 운영자가 쉽게 적용할 수 있도록 가이드를 구성하고자 노력하였다.

본 가이드에서는 Windows IIS, Linux Apache 서버에 적용하여야 하는 최소한의 서버 설정을 다루고 있으므로 웹사이트 또는 웹어플리케이션의 보안 설정은 취약점 점검을 통하여 강화하기 바란다. 웹사이트의 취약점 점검은 해모수(hemos.postech.ac.kr)의 정보보호 서비스에서 자세하게 안내하고 있으니 참고하기 바란다.

제2장에서는 웹 서버 시스템과 관련하여 OS 자체의 보안 설정과 더불어 웹 서버의 보안 설정 및 점검 방법에 대해 알아보며, 제3장에서는 웹서버 OS별로 적용하여야 하는 최소한의 보안설정에 대해 알아보도록 한다. 그리고 제4장에서는 웹서버 구축 단계에서 점검하여야 하는 보안체크리스트를 제시하여 웹서버 구축 시 필요한 보안항목에 대해 알아보도록 한다.

제 2 장 웹 서버 취약점 점검

제 1 절 호스트 OS 보안

모든 시스템은 어플리케이션 보안에 앞서 호스트 OS의 보안 작업이 선행되어야 한다. 웹 서버의 경우, 아무리 웹 서버를 안전하게 설정 및 운영한다 해도, 웹 서버가 설치될 OS가 안전하지 않다면 결코 웹 서버의 안전을 보장할 수 없다.

1. OS에 대한 최신 패치 적용

OS 벤더사이트나 보안 취약점 정보 사이트를 주기적으로 방문하여 현재 사용하고 있는 OS에 대한 최신 취약점 정보를 얻고, 패치 등 관련된 보안대책을 신속하게 적용하도록 한다.

2. OS 취약점 점검

정기적으로 취약점 점검 도구와 보안 체크리스트를 사용하여 호스트 OS의 보안 취약점을 점검한다. 점검 결과로 발견된 취약점들은 보완조치하고 조치사항은 히스토리 관리를 위해 기록해 둔다.

3. 웹 서버 전용 호스트로 구성

웹 서버의 중요도를 고려하여 가급적이면 웹 서버 전용 호스트로 구성하도록 한다. 웹 서비스 운영에 필요한 최소한의 프로그램들만 남겨두고, 불필요한 서비스들은 반드시 제거 하도록 한다. 시스템 사용을 목적으로 하는 일반 사용자 계정들은 모두 삭제하거나 최소의 권한만 할당한다. 오로지 관리자만이 로그인이 가능하도록 한다.

※ 개발도구 및 백업파일 제거

웹 서버를 구축한 후에는 컴파일러 같은 소프트웨어 개발 도구와 백업파일들을 제거 하도록 한다. 이러한 도구들은 공격자가 서버에 침입한 후에 공격코드를 컴파일하거나, 웹 페이지의 소스확인을 통해 DB 접속 계정정보 등이 유출될 수 있다.

4. 서버에 대한 접근 제어

관리목적의 웹 서버 접근은 콘솔 접근만을 허용하는 것이 가장 좋다. 그것이 불가능하다면 관리자가 사용하는 PC의 IP만 접근이 가능하도록 접근제어를 수행한다.

5. DMZ 영역에 위치

웹 서버를 DMZ 영역에 위치시키도록 한다. 웹 서버를 방화벽에 의해서 보호 받도록 하고, 웹 서버가 침해당하더라도 웹 서버를 경유해서 내부 네트워크로의 침입은 불가능하도록구성한다.

6. 강력한 관리자 계정 패스워드 사용

관리자 계정의 패스워드는 모든 보안의 가장 기본이다. 하지만 이런 기본이 지켜지지 않아 여전히 해킹사고가 많이 발생하고 있다. 패스워드 보안은 모든 보안의 기본이자 가장 중요한 필수 보안 사항이다.

관리자 계정 패스워드는 유추가 불가능하고 패스워드 크랙으로도 쉽게 알아낼 수 없는 강력한 패스워드를 사용하도록 한다. 패스워드는 길이가 최소한 8자 이상이고, 이름이나 계정명으로 유추할 수 없는 것이어야 한다. 또한 사전에 없는 단어를 사용하도록 하고, 기호문자를 최소 한 개 이상 포함시키도록 한다.

관리자 계정 패스워드의 주기적인 변경 또한 중요하다. 관리자 계정을 포함한 주요 계정의 패스워드는 내부적으로 규정한 주기마다 변경하도록 하며, 변경 시에는 일정한 규칙을 가지지 않도록 한다.

7. 파일 접근권한 설정

관리자 계정이 아닌 일반 사용자 계정으로 관리자 계정이 사용하는 파일들을 변경할 수 없도록 해야 한다. 만약 관리자 계정보다 권한이 낮은 일반 계정으로 관리자가 실행하거나 쓰기를 수행하는 파일들을 변경할 수 있다면 관리자 권한 획득이 가능하다.

제 2 절 웹 서버 설치보안

1. 소스코드 형태의 배포본 설치

웹 서버 소프트웨어가 소스코드와 바이너리 형태로 배포되는 경우, 보안상 가장 좋은 것은 소스코드를 다운로드 받아 필요한 기능만 설치하는 것이다. 소스의 다운로드는 해당 프로그램의 공식 사이트를 통해 다운로드 받으며, 다운로드 후 MD5 해쉬값을 비교하도록 한다.

2. 설치 시 네트워크 접속 차단

웹 서버를 설치하기 전부터 보안설정을 안전하게 끝낼 때까지 호스트의 네트워크 접속을 차단하도록 한다. 보안설정이 완전히 끝나지 않은 상태에서 웹 서버가 외부에 노출될 경우 쉽게 해킹 당할 수 있으며, 그 이후에 취해지는 보안 조치들이 의미가 없게 될 수 있다.

3. 웹 프로세스의 권한 제한

시스템 전체적인 관점에서 웹 프로세스가 웹 서비스 운영에 필요한 최소한의 권한만을 갖도록 제한한다. 이렇게 하여 웹사이트 방문자가 웹 서비스의 취약점을 이용해 시스템에 대한 어떤 권한도 획득할 수 없도록 한다. 시스템 운영 시, root 권한으로 웹 데몬이 재구동되고, 웹 취약점을 통해 접속권한을 획득한 경우 root 권한을 획득하게 되므로 웹 서버 관리 시에는 일반적으로 사용되는 nobody 권한으로 웹 프로세스가 동작하도록 한다.

4. 로그 파일의 보호

로그 파일은 침입 혹은 침입시도 등 보안 문제점을 파악하는데 중요한 정보를 제공한다. 이러한 로그 파일이 노출, 변조 혹은 삭제되지 않도록 불필요한 접근으로부터 보호한다.

5. 웹 서비스 영역의 분리

웹 서비스 영역과 시스템(OS)영역을 분리시켜서 웹 서비스의 침해가 시스템 영역으로 확장될 가능성을 최소화한다. 웹 서버의 루트 디렉토리와 OS의 루트 디렉토리를 다르게 지정한다.

웹 컨텐츠 디렉토리는 OS 시스템 디렉토리는 물론 가급적 다른 웹 서버 디렉토리와도 분리시킨다. 또한 로그 디렉토리와 설정 디렉토리는 웹 서비스를 통해 접근이 불가능한 곳에 위치시키도록 한다.

6. 링크 사용금지

공개 웹 컨텐츠 디렉토리 안에서 서버의 다른 디렉토리나 파일들에 접근할 수 있는 심볼릭 링크, aliases, 바로가기 등을 사용하지 않는다.

7. 자동 디렉토리 리스팅 사용중지

디렉토리 요청 시 디렉토리 내에 존재하는 파일 목록을 보여주지 않도록 설정한다. 디렉토리 내에 존재하는 DB 패스워드 파일이나 웹 어플리케이션 소스 코드 등 중요한 파일들에 대해 직접 접근이 가능하면 보안상 매우 위험하다. 이를 막기 위해 자동 디렉토리 리스팅 기능의 사용을 중지시킨다.

8. 기본 문서 순서 주의

웹 서버에서는 디렉토리 요청시 기본적으로 보여지는 파일들을 지정할 수 있도록 되어있다. 이 파일 목록을 올바로 지정하여 기본 문서가 악의적인 목적의 다른 파일로 변경되지 않도록 한다.

9. 샘플 파일, 매뉴얼 파일, 임시 파일의 제거

웹 서버를 설치하면 기본적으로 설치되는 샘플 파일이나 매뉴얼 파일은 시스템 관련 정보를 노출하거나 해킹에 악용될 수 있다. 따라서 웹 서버 설치 후에 반드시 이러한 파일들을 찾아서 삭제하도록 한다.

만약 관리 등의 이유로 웹을 통해 설명문서에 접근해야 한다면 접근제어를 통해 꼭 필요한 사용자만 접근을 허용하고 그 외의 사용자들은 접근하지 못하도록 설정한다.

또한 웹 서버를 정기적으로 검사하여 임시 파일들을 삭제하도록 한다. 특히 웹 서비스의 업데이트나 유지보수 시 생성되는 백업파일이나 중요한 파일 등은 작업이 끝난 후 반드시 삭제하도록 한다.

정확한 관리를 위해 폴더와 파일의 이름과 위치, 개수 등이 적혀있는 별도의 문서를 관리하는 것이 좋다. 그래서 문서에 등록되지 않은 불필요한 파일들을 점검해서 삭제하도록 한다.

10.웹 서버에 대한 불필요한 정보 노출 방지

웹 서버 종류, 사용 OS, 사용자 계정 이름 등 웹 서버와 관련된 불필요한 정보가 노출되지 않도록 한다. 이러한 정보가 사소한 것처럼 보일 수 있지만, 이러한 정보를 아는 것만으로도 공격에 필요한 나머지 정보를 수집하는데 도움이 될 수 있다.

뉴스그룹이나 메일링 리스트를 통해 웹 서버 운영에 대한 질의를 할 경우에도, 조직의 네트워크와 시스템에 대한 상세정보가 유출되지 않도록 주의한다.

11.업로드 제어

불필요한 파일 업로드는 허용하지 않는다. 파일 업로드를 허용해야 한다면, 대량의 업로드로 인한 서비스 불능상태가 발생하지 않도록 한다. 또한 업로드를 허용해야 하는 파일의 종류를 지정하여 그 외 종류의 파일들은 업로드가 불가능하도록 한다. 업로드된 파일은 웹 서버에 의해 바로 처리되지 못하도록 해야 한다. 처리되기전에 반드시 수동이나 자동으로 파일의 보안성 검토를 수행하도록 한다. 또한, 업로드 되는 폴더의 실행권한을 제거하여 악성 파일이 업로드 되었을 시 실행되지 못하도록 한다. 업로드 폴더를 웹 서비스 폴더와 별도로 사용하는 것도 좋은 방법이다.

12.인증과 접근제어의 사용

웹 서버에서 제공하는 인증 기능과 접근제어 기능을 필요한 곳에 적절하게 활용한다. 웹 서버에서는 사용자 아이디/패스워드 기반의 인증 기능과 특정 IP나 도메인에 대한 접근제어 기능을 제공한다.

13.패스워드 설정 정책 수립

웹 서버의 인증 기능을 이용하는 경우에, 유추가 불가능한 패스워드를 사용하도록 한다. 패스워드 길이와 사용 문자에 대한 최소 복잡도를 설정하도록 하고, 사용자의 개인정보나 회사 공개정보를 이용한 패스워드 사용을 금지하도록 한다. 또한 사용자들에게 웹사이트의 패스워드와 다른 중요한 것들의 패스워드(예를 들어, 은행이나 주식 관련 비밀번호)를 다르게 사용하도록 권장한다. 웹 서버 보안이 100% 완벽할 수 없기 때문에, 이렇게 함으로써 웹 서버 침해로 인한 더 큰 피해를 막을 수 있다.

14.동적 컨텐츠 실행에 대한 보안 대책 수립

동적 컨텐츠 처리 엔진들은 웹 서버의 일부로서 실행되면서 웹 서버와 동일한 권한으로 실행된다. 따라서 각 엔진 사용시 발생할 수 있는 모든 보안 취약점들을 파악하고 이와 관련된 보안 기능들을 설정해야 한다.

동적 컨텐츠와 관련된 기능 중 사용하지 않는 기능들은 제거를 하고 예제 파일들은 반드시 삭제한다. 가능하다면 동적 컨텐츠가 실행될 수 있는 디렉토리를 특정 디렉토리로 제한시키도록 하고, 컨텐츠의 추가 권한은 관리자로 제한하도록 한다.

15.설치 후 패치 수행

웹 서버 기본 설치 후 알려진 취약점을 바로잡기 위해 취약점 정보사이트나 벤더 사이트를 방문해서 웹 서버와 관련된 취약점 정보를 얻고, 패치나 업그레이드를 수행한다.

16.설정 파일 백업

웹 서버를 인터넷에 연결하기 전에 초기 설정 파일을 백업 받아서 보관해 둔다. 또한 변경이 있을 때마다

설정 파일을 백업하도록 한다. 이렇게 하여 해킹이나 실수가 발생해도 빠르게 복구할 수 있도록 한다.

17.SSL/TLS 사용

보안과 기밀성이 요구되는 경우 SSL이나 TLS를 사용하도록 한다. 대부분의 경우에 SSL/TLS는 웹 서버에서 사용할 수 있는 가장 훌륭한 인증 및 패스워드 방법이다.

제 3 장 OS별 필수 설정

이 장에서는 안전한 웹서버 관리를 위한 OS별 최소한의 설정을 다루려고 한다. 따라서 신규로 웹서버를 구성하려는 사용자뿐만 아니라 운영중인 웹서버의 모든 관리자는 해당 서버의 설정을 반드시 조치하기 바랍니다. 이 장에서 다루는 설정은 안전한 웹서버의 관리를 위한 최소한의 서버 설정이므로 운영중인 웹사이트 및 신규 구축 웹사이트는 웹사이트의 취약점 점검을 수행하기 바란다. 학술정보처 정보시스템팀에서 취약점 점검 서비스를 지원하고 있으니 자세한 정보는 해모수(hemos.postech.ac.kr) – 정보보호서비스에서 확인하기 바랍니다.

제 1 절 Windows IIS

1. 관리자 페이지 접근통제

■ [설정] ➔ [제어판] ➔ [관리도구] ➔ [인터넷 서비스 관리자] 선택

- 해당 관리자 페이지 폴더를 선택하고 [등록정보] ➔ [디렉토리 보안] ➔ [IP 주소 및 도메인 이름 제한] ➔ [편집] 선택
- 액세스 거부를 선택하고 추가 버튼을 선택하여 관리자 호스트 IP 또는 서브넷을 등록

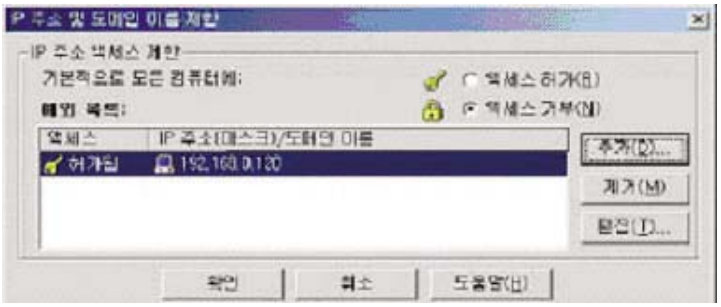


그림 1 관리자 페이지 IP 접근 제한 설정

2. 디렉토리 나열(Directory Listing) 금지

■ [제어판] ➔ [관리도구]의[인터넷 서비스 관리자](혹은 [인터넷 정보 서비스]) 메뉴에서[기본 웹 사이트]의 마우스 오른쪽 클릭,' 속성'부분을 보면'기본 웹 사이트 등록 정보'가 나온다.

■ '기본 웹 사이트 등록 정보'에서'홈 디렉토리'부분을 클릭하면 [그림 2]와 같은 화면이 나타난다.

■ [그림 2]에서 중간에'디렉토리 검색(B)'이란 옵션이 있는데 바로 이 부분이 웹브라우저를 통해 디렉토리 리스팅을 가능케 하는 부분이며, [그림 2]와 같이 이 부분의 체크를 해지한다.

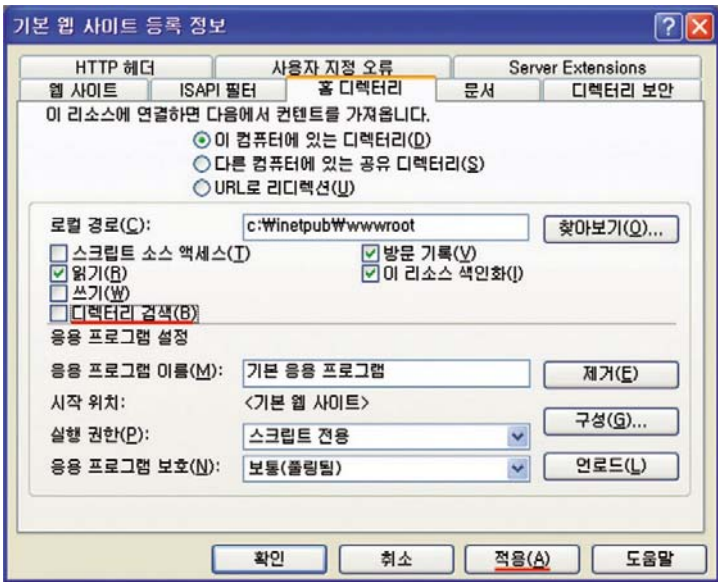


그림 2 기본 웹 사이트 등록 정보에서 디렉토리 검색 불가 설정 화면

3. 불필요한 Method 및 WebDAV 취약점 제거

■ 불 필요한 메소드 제거(IIS)

아래 그림과 같이 웹 홈 디렉토리의 쓰기 권한을 제거 하고, PUT, DELETE 등의 불필요한 메소드를 제거 하기 위해서는 원격지에서 웹 서버의 컨텐츠를 추가 하고 관리 하는 WebDAV 를 중지 시켜야 한다.

실행 regedit 를 이용하여 아래 경로의 레지스트리 값을 새로 추가한 후 시스템을 재 부팅한다.

레지스트리 경로 : HKEY_LOCAL_MACHINE₩SYSTEM₩CurrentControSet₩Services ₩W3SVC₩Parameter 레지스티리 값 : DisableWebDAV=1(DWORD)(대소문자 주의)

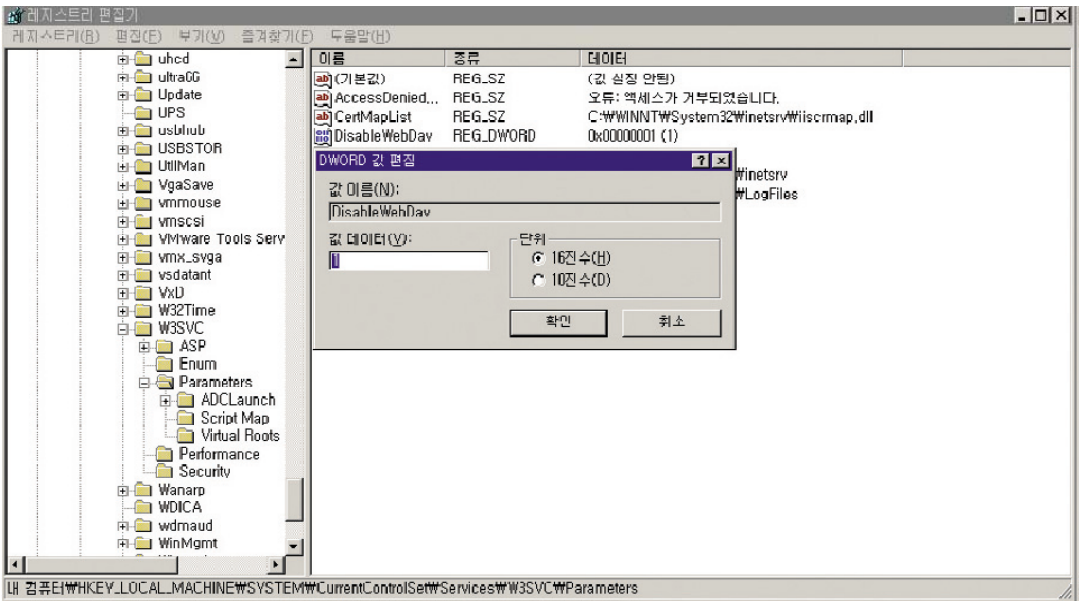


그림 3 레지스트리 추가

WebDAV 를 반드시 사용해야 하는 경우라면, 다음 소개되는 2 개의 조치 사항을 모두 수행해야 한다.

■ **httpext.dll 파일의 Everyone 권한 삭제**

① Httpext.dll 파일은 ₩winnt₩system32₩inetsrv₩httpext.dll 에 존재 하거나 ₩windows₩system32₩inetserv₩httpext.dll 에 존재한다. 우선 웹 서버의 해당 디렉토리로 이동한다. [그림 4]와 같이 파일을 선택한 후에 마우스 오른쪽 클릭을 하여'속성'메뉴를 클릭한다.

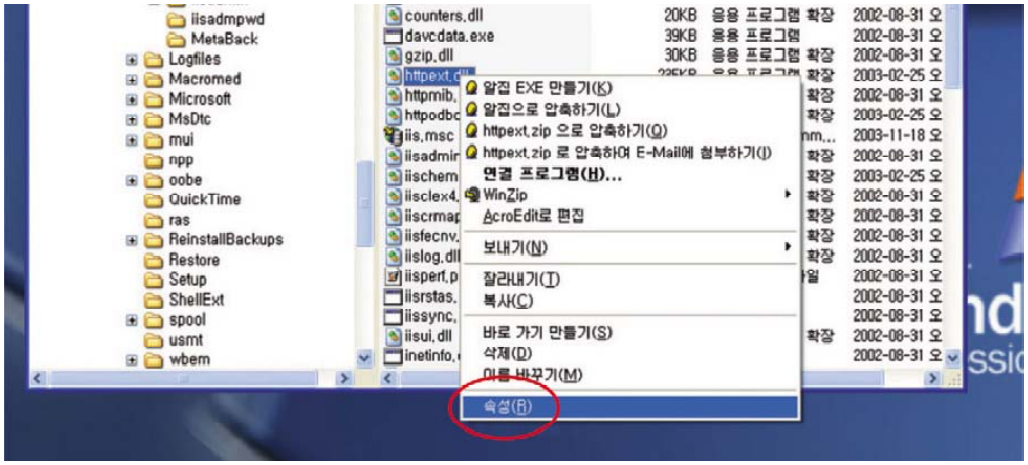


그림 4 httpext.dll 위치 및 속성 확인

② [그림 5]와 같이 'httpext.dll 등록정보' 창의 '보안' 탭에 들어간다. '그룹 또는 사용자 이름' 항목의 Everyone 을 선택, 그리고 '제거' 버튼을 눌러 Everyone 을 제거한다.

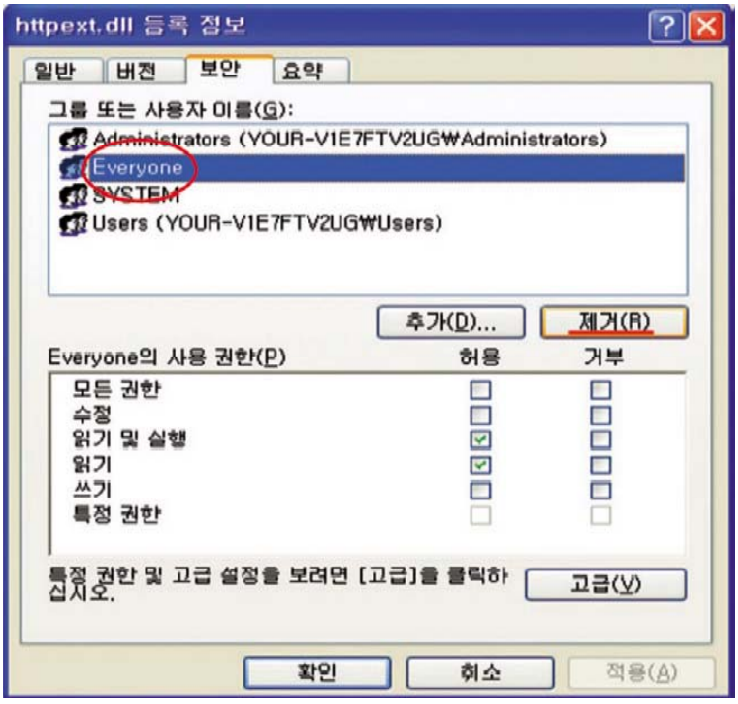


그림 5 httpext.dll 등록 정보 창

■ **홈 디렉토리 메뉴의‘쓰기’권한 삭제**

① [제어판] ➔ [관리도구]의 [인터넷 서비스 관리자](혹은 [인터넷 정보 서비스]) 메뉴에서 [기본 웹사이트]의 마우스 오른쪽 클릭, ‘속성’부분을 보면 ‘기본 웹사이트 등록 정보’가 나온다.

② ‘기본 웹사이트 등록 정보’에서 ‘홈 디렉토리’ 부분을 클릭, ‘로컬 경로’항목의 ‘쓰기’부분을 해제해 주시고 ‘확인’ 및 ‘적용’ 버튼을 클릭한다. ([그림 6] 참조).

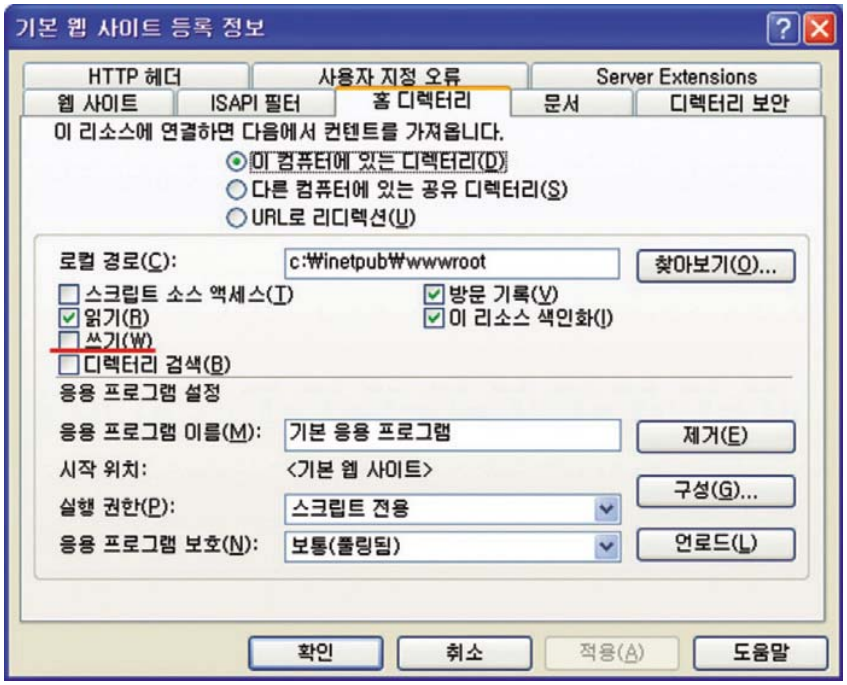


그림 6 기본 웹 사이트 등록 정보 창

4. 전송 시 개인정보 노출 취약점

■ 웹 상에서 스니핑이 가능해서 로그인, 회원가입, 전자상거래의 이용 시 입력 정보를 안전하게 주고받기 위하여, SSL(Secure Socket Layer)이라는 암호화 전송규약을 정하고 인터넷 브라우저와 웹 서버 사이에 암호화 통신이 가능함. SSL 서버는 대부분의 웹사이트에서 선택사항이 아니라 필수 항목이 되었으며 SSL 서버를 통해 개인정보 및 상거래 정보를 보호할 수 있다.

※ 보안서버(SSL 인증서) 구축 가이드 참조

[해모수(hemos.postech.ac.kr)] – 사용자가이드 – 17 번 보안서버(SSL 인증서) 구축 가이드

■ 중요 정보를 보여주는 페이지는 캐쉬를 사용하지 못하도록 설정한다.

중요 정보를 보여주는 화면에 no-cache 설정을 하지 않을 경우, 로그아웃을 한 이후에도 [뒤로가기] 버튼을 사용해서 해당 내용을 볼 수 있는 위험이 존재한다.

no-cache 설정을 위해서 HTML HEAD 부분에 아래 내용을 추가한다.

```
<meta HTTP-EQUIV="Pragma" CONTENT="no-cache">
```

5. 파일 업로드 취약점

■ 설정 ➔ 제어판 ➔ 관리도구 ➔ 인터넷 서비스 관리자 선택
해당 업로드 폴더에 오른쪽 클릭을 하고 등록정보 디렉토리 실행권한을"없음"으로 설정

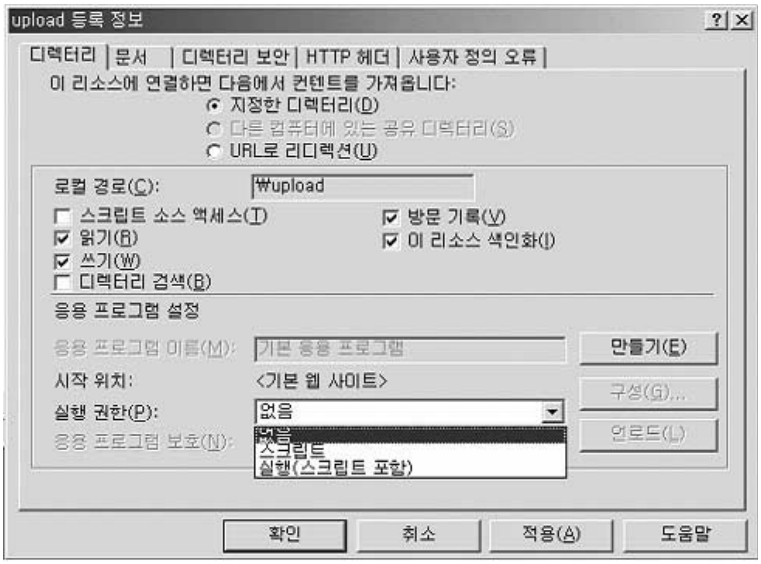


그림 7 IIS 보안 설정

6. 에러처리 취약점

■ 모든 에러(404 error,403 error,500 error, ...)발생 시 에러 메시지를 외부에 제공하지 않도록 함. 에러가 발생한 경우 에러 발생 메시지를 사용자 클라이언트의 브라우저에 표시하지 않게 하고 메인 페이지 또는 별도로 만든 에러 페이지로 Redirect 시키도록 함.

■ 에러 메시지는 공격자에게 무엇이 틀렸는지 알려주는 표시를 해주며 이로 인해 공격자는 각각의 지시에 대해 다양한 공격 방법을 시도할 수 있게 된다.
▪ 제어판 ➔ 관리도구 ➔ 인터넷 정보 서비스(IIS) ➔ 웹사이트 속성 ➔ 사용자 지정 오류 탭 에서 에러 메시지에 대한 사용자 지정 페이지를 지정해 준다.

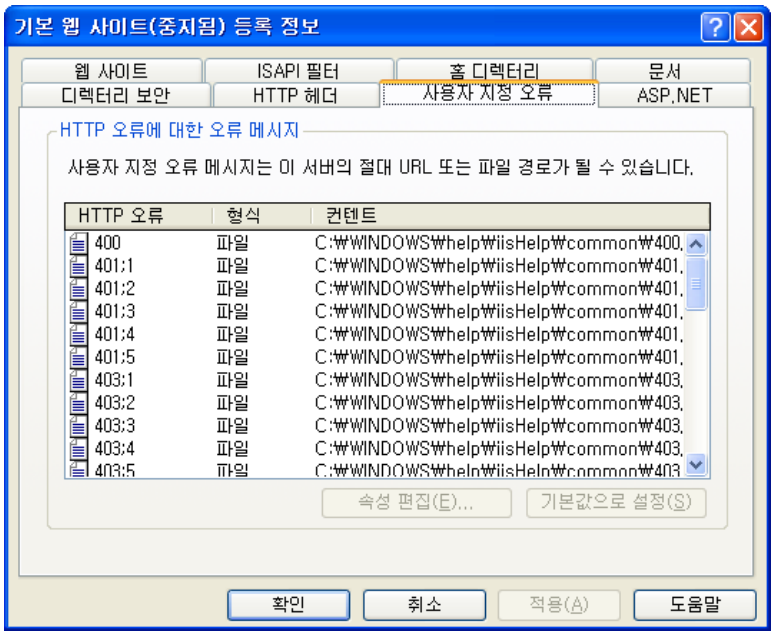


그림 8 IIS 사용자 지정 오류 설정

제 2 절 Linux - Apache

1. 관리자 페이지 노출 취약점

■ **관리자 페이지 접근 통제**

관리자페이지는 가장 보호되어야 할 페이지로 매우 제한적으로 접근을 허용하여야 한다. 기본적으로 특정 IP Address 에서만 접근을 허용하고 그 외의 모든 접근은 차단하는 것이 바람직하다. 이를 위해 httpd.conf 파일에서 다음과 같이 설정한다.

```
...
<Directory "/usr/local/www/admin/">
order deny,allow
deny from all
allow from 100.100.1000.100 ← 100.100.1000.100 에서만 접근 허용
</Directory>
...
```

2. 디렉토리 나열 취약점

■ 아파치 서버의 모든 설정은 httpd.conf 라는 파일을 통해 가능하다.

httpd.conf 파일은 운영체제 및 아파치 버전에 따라서 다양하나 주로 다음 디렉토리 중에 존재한다.

```
/etc/httpd/conf/httpd.conf
/etc/apache/httpd.conf
/usr/apache/conf/httpd.conf
/usr/local/apache/conf/httpd.conf
/usr/lib/apache/conf/httpd.conf
```

■ 파일 위치가 조금 다를 수 있으므로 위 5 가지 디렉토리 주변 디렉토리들을 찾아보면 apache 의 httpd.conf 파일을 찾을 수 있다. 초기 설치 시 옵션을 주어 사용자 임의의 디렉토리에 설치될 수도 있으므로 경우에 따라 설치한 사람에게 문의해 보는 것이 필요할 수도 있다.

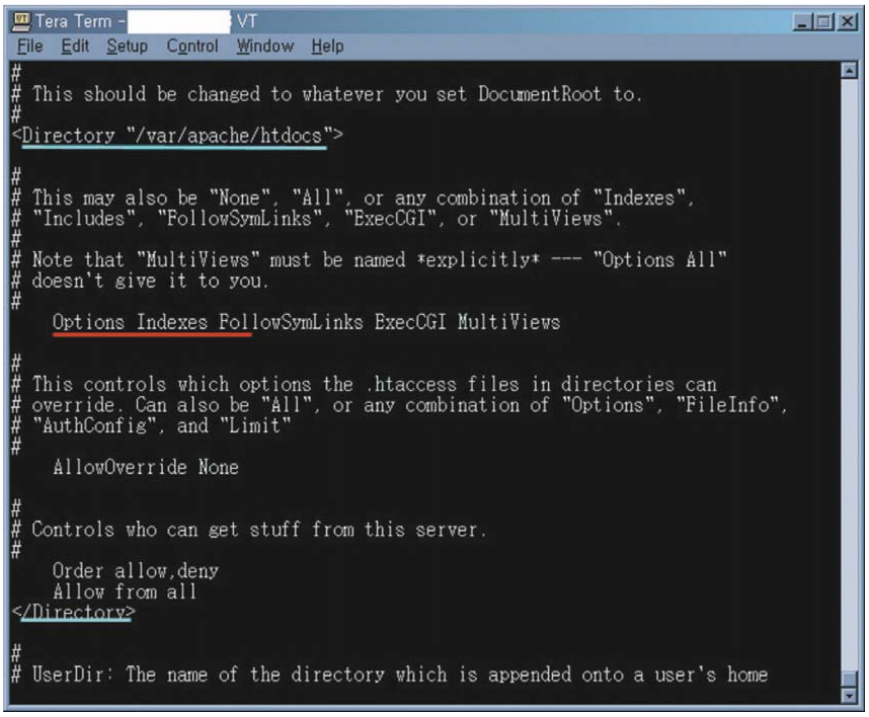


그림 9 httpd.conf 파일 내용

■ [그림 9]의 중간에 보면 Options 항목 뒤에 Indexes 옵션이 바로 디렉토리 리스팅을 가능하게 하는 옵션이다. Indexes 설정을 끄기 위해 Indexes 단어를 지워주고 파일을 저장하고 설정을 적용하기 위해 웹 서버 데몬을 다시 띄워주어야 한다.

※ *주의 : Options 항목은 특정 디렉토리 별로 설정할 수 있게 되어 있기 때문에 항상 그 설정이 적용될 디렉토리를 명시해 주어야 한다. 현재[그림 9]에서는 Directory" /var/apache/htdocs"같이 웹 루트 디렉토리 필드 안에 Options 항목이 위치되어 있다. 다른 특정 디렉토리 설정에도 Options 항목을 통해 Indexes 가 설정되어 있을지도 모르므로 그러한 부분들을 찾아 확인하고 지워주어야 한다.*

3. 시스템 관리 취약점

■ 불필요한 파일 관리

홈페이지 서버에 테스트 파일과 같은 불필요한 파일을 삭제하고 홈페이지 서비스와 관련 없는 디렉토리(백업디렉토리 등)는 일반사용자가 접근이 불가능하도록 적절한 권한(디렉토리 또는 파일 접근권한)을 설정한다.

```
<Files ~"₩.bak$">
Order allow,deny
Deny from all
</Files>
```

■ 심볼릭 링크 사용 설정 제거

유닉스 계열의 시스템인 경우 심볼릭 링크 기능(ln -s / system.html)을 사용하여 다른 위치의 파일이나 디렉토리와 연결하여 사용할 수 있는데 이런 경우 웹에서 허용하는 디렉토리 외에 다른 디렉토리를 참조하는 링크가 존재하는 경우 해당 링크를 엑세스할 수 있는 위험성이 존재한다. 이 위험성을 제거하기 위해서는 디렉토리 리스트와 마찬가지로 httpd.conf 파일에서 DocumentRoot 항목을 아래와 같이 수정한다.

```
...
<Directory "/usr/local/www">
Options FollowSymLinks ← 제거한다
</Directory>
...
```

■ 헤더 정보 숨기거나 최소화

클라이언트와 Apache 가 통신을 할 경우 웹 서버에서는 응답 메시지의 헤더에 웹서버 버전이나 응용 프로그램 버전 등을 전송한다.

```
[root@ conf]# telnet xxx.xxx.xxx.xxx 80
Trying xxx.xxx.xxx.xxx...
Connected to xxx.xxx.xxx.xxx.
Escape character is ']'
GET / HTTP/1.1
HTTP/1.1 400 Bad Request
Date : Tue, 15 Oct 2002 11:25 : 10 GMT
Server : Apache/1.3.19 (Unix) PHP/4.0.4pl1 ← 서버 버전 정보
```

이 정보는 공격자에 의해 Apache 웹 서버 버전별 또는 구동되고 있는 응용프로그램에 잘 알려진 취약점을 공격하는데 유용하게 이용되기 때문에 공격자에게 웹 서버의 버전과 같은 banner 정보를 숨기는 것이 안전하다.

Apache 웹 서버에서는 httpd.conf 내용에 ServerTokens 지시자를 삽입하여 헤더에 의해 전송되는 정보를 최소화 할 수 있다.

```
ServerTokens Minimal|ProductOnly|OS|Full
```

표 1 설정별 제공되는 헤더 정보

키워드	제공하는 정보	예시
Prod[uctOnly]	웹 서버 종류	Server : Apache
Min[imal]	Prod 키워드 제공 정보 + 웹 서버 버전	Server : Apache/1.3.0
OS	Min 키워드 제공 정보 + 운영체제	Server : Apache/1.3.0 (Unix)

Full	OS 키워드 제공 정보 + 설치된 모듈(응용프로그램) 정보	Server : Apache/1.3.0 (Unix) PHP/3.0 MyMod/1.2

※ 위 내용은 httpd.conf 에서 기본적으로 설정되어 있지 않으며 내부적인 기본 설정은 Full 이다.

4. 불필요한 Method 허용 취약점

■ Web 서버의 DocumentRoot 로 사용되는 디렉토리에 일반 사용자에 대한 접근권한을 제거 한다.
Httpd.conf 설정 파일에서 아래와 같은 내용을 추가 한다.

```
...
<Directory "/home/*/public_html">
<Limit POST PUT DELETE>
Require valid-user
</Limit>
</Directory>
...
```

5. 전송 시 개인정보 노출 취약점

■ 웹 상에서 스니핑이 가능해서 로그인, 회원가입, 전자상거래의 이용 시 입력 정보를 안전하게 주고 받기 위하여, SSL(Secure Socket Layer)이라는 암호화 전송규약을 정하고 인터넷 브라우저와 웹 서버 사이에 암호화 통신이 가능함. SSL 서버는 대부분의 웹사이트에서 선택사항이 아니라 필수 항목이 되었으며 SSL 서버를 통해 개인정보 및 상거래 정보를 보호할 수 있다.
※ 보안서버(SSL 인증서) 구축 가이드 참조
[해모수(hemos.postech.ac.kr)] – 사용자가이드 – 17 번 보안서버(SSL 인증서) 구축 가이드

■ 중요 정보를 보여주는 페이지는 캐쉬를 사용하지 못하도록 설정한다.
중요 정보를 보여주는 화면에 no-cache 설정을 하지 않을 경우, 로그아웃을 한 이후에도 [뒤로가기] 버튼을 사용해서 해당 내용을 볼 수 있는 위험이 존재한다.
no-cache 설정을 위해서 HTML HEAD 부분에 아래 내용을 추가한다.

```
<meta HTTP-EQUIV="Pragma" CONTENT="no-cache">
```

6. 파일 업로드 취약점

■ Apache 설정 파일인 httpd.conf 에 해당 디렉토리에 대한 문서 타입을 컨트롤하기 위해 Directory 섹션의 AllowOverride 지시자에서 FileInfo 또는 All 추가

```
<Directory "/usr/local/apache">
AllowOverride FileInfo (또는 All) .....
.....
.....
</Directory>
```

파일 업로드 디렉토리에 .htaccess 파일을 만들고 다음과 같이 AddType 지시자를 이용 현재 서버에서

운영되는 Server Side Script 확장자를 text/html 로 MIME Type 을 재조정하여 업로드 된 Server Side Script 가 실행되지 않도록 설정한다.

또는 FileMatch 지시자를 이용하여 *.ph, *.inc, *lib 등의 Server Side Script 파일에 대해서 직접 URL 호출을 금지시킨다.

```
<.htaccess>
<FilesMatch "₩.(ph|inc|lib)">
Order allow, deny
Deny from all
</FilesMatch>
AddType text/html .html .htm .php .php3 .php4 .phtml .phps .in .cgi .pl .shtml .jsp
```

※ 주의사항

1. Apache 서버의 경우 AllowOverride 지시자를 변경 시 apache restart 가 필요하다.

2. 파일 업로드 되는 디렉토리에 운영에 필요한 Server Side Script 가 존재하는지 확인한다. 파일 다운로드 프로그램이 아닌 직접 URL 호출을 통해 파일을 다운받는 경우 FileMatch 지시자를 사용하면 차단 설정한 확장자의 파일 다운로드는 거부된다.

7. 에러처리 취약점

■ 모든 에러(404 error,403 error,500 error, ...)발생 시 에러 메시지를 외부에 제공하지 않도록 함. 에러가 발생한 경우 에러 발생 메시지를 사용자 클라이언트의 브라우저에 표시하지 않게 하고 메인 페이지 또는 별도로 만든 에러 페이지로 Redirect 시키도록 함.

■ 에러 메시지는 공격자에게 무엇이 틀렸는지 알려주는 표시를 해주며 이로 인해 공격자는 각각의 지시에 대해 다양한 공격 방법을 시도할 수 있게 된다. Apache 에서 에러 메시지를 처리하는 방법은 4 가지로 나누어진다.

- 간단한 시스템에서 작성된 에러 메시지 출력
- 사용자가 수정한 메시지 출력
- 문제나 에러를 해결하기 위한 로컬 URL 을 Redirection
- 문제나 에러를 해결하기 위한 외부 URL 을 Redirection

■ 이중 가장 보편적인 방법은 별도의 에러 페이지를 제작하여 각각의 에러코드에 대해 에러 페이지로 Redirection 시키는 방법이다.

httpd.conf 파일에서 아래와 같이 설정한다.

```
ErrorDocument 404 /error_page.html
```

제 4 장 보안 체크리스트

제 1 절 IIS Web Server 보안 체크리스트

항목	상세설명	체크
보안패치	1. 웹서버가 운영되고 있는 운영체제는 최신 보안 업데이트를 하였는가? 2. 현재 사용하는 웹서버 버전에 대한 최신 보안 업데이트를 하였는가? (흔히 최신 보안 업데이트는 사용 중인 Application과의 호환을 이유로 미루는 경향이 있는데, 보안패치를 하지 않아 Critical한 문제가 발생하는 경우가 종종 있으므로 개발장비에서 반드시 미리 테스트 후 적용하도록 함) 3. ASP.NET Framework을 사용시 최신 버전으로 업데이트하였는가?	
주요 Tool	1. IISLockDown (MS 제공 IIS 운영시 보안향상용 툴) 2. URLScan (MS 제공 IIS 운영시 HTTP 요청 제한 툴) 3. WebKnight (KISA 제공 IIS용 무료 웹방화벽)	
로깅	1. 웹서버 로깅을 하고 있는가? 2. 로그파일에 적절한 ACL을 설정하였는가? (관리자만 Read-Only) 3. 로그파일은 주기적으로 백업하고 있는가?	
불필요 요소 제거	1. 샘플 디렉토리와 기본 컨텐츠를 제거하였는가? - %systemdirectory%₩inetsrv₩iisadmin - systemdirectory%₩inetsrv₩iisadmpwd - inetpub₩wwwroot (or ₩ftproot or ₩smtproot) - inetpub₩scripts - inetpub₩iissamples - inetpub₩adminscripts - %systemroot%₩help₩iishelp₩iis - %systemroot%₩web₩printers - %systemdrive%₩program files₩common files₩system₩msadc 2. 불필요한 COM 구성요소를 제거하였는가? - regsvr32 scrrun.dll /u 3. 불필요한 스크립트 매핑(Script Mapping)은 제거하였는가? - 홈 디렉토리 탭 → 응용 프로그램 구성 → 응용 프로그램 매핑 .asa, .asp, .bat, .cdx, .cer, .htr, .htw, .ida, .idc, .idq, .printer, .shtm, .shtml, .stm 4. 불필요한 ISAPI 필터는 제거하였는가? [Frontpage ISAPI 제거] c:₩program files₩common files₩microsoft shared₩web server extensions₩40₩bin₩fpsrvadm –o uninstall –p all [Digest 인증], [HTTP 압축] 5. 사용하지 않는 모든 스크립트와 실행 파일은 제거하였는가?	

	6. 불필요한 SubSystem은 제거하였는가? - HKKM/System/CurrentControlSet/Control/Session Manager/SubSystems 삭제 - %systemroot%/system32 폴더에서 os2*, posix*, psx* 파일 삭제	
안전한 설정	1. 디렉토리 인덱싱이 가능한 디렉토리가 있는가? 2. 실행권한이 필요한 경우 특정 디렉토리로 한정하였는가? (사용자가 임의로 업로드하는 디렉토리의 경우 실행권한 제거) 3. 불필요한 HTTP Method를 제거하였는가? 4. 사용자 지정 에러 페이지를 사용하고 있는가? (웹서버의 에러코드는 RFC2616의 상태코드 정의부분에 명시되어 있음) 5. 상위 디렉토리로 갈 수 없도록 설정하였는가? - 홈 디렉토리 탭 → 구성 버튼 → 옵션 → 상위 디렉토리 Disabled 6. SYN Attack을 방어하기 위한 설정을 하였는가? - HKEY_LOCAL_MACHINE₩System₩CurrentControlSet₩Services₩Tcpip₩Parameters - SynAttackProtect : 2 [REG_DWORD] 3 way-Handshaking이 완료될 때까지 자원의 할당을 지연 - TcpMaxHalfOpen : 100 (Server), 500 (Advanced Server) [REG_DWORD] SYN-ATTACK 공격이 동작하기 전 SYN-RCVD 상태 연결 개수 제한 - TcpMaxHalfOpenRetried : 80 (Server), 400 (Advanced Server) [REG_DWORD] SYN-RCVD 상태에서 연결되는 개수 제어	
운영	1. 웹서버는 제한적인 계정으로 최소한의 권한을 가지고 운영되는가? 2. 웹서버 설정파일과 주요데이터는 정기적으로 백업하고 있는가? 3. 백업된 데이터는 정상적으로 복구하여 검증하는 절차가 있는가? 4. cmd.exe, runas.exe에 대한 권한은 admin을 제외하고 모두 삭제하였는가? 5. 터미널 서비스를 사용하고 있다면, 적절한 ACL을 설정하였는가? 6. 주요 디렉터리(백업, 문서, 임시파일 등)는 webroot 외부에 생성하는가? 7. 가상 디렉터리에 적절한 ACL을 설정하였는가?	
주요정보 암호화	1. 민감한 정보는 인증서를 이용하여 SSL/TLS로 암호화하고 있는가? 2. IIS 서버에 RootCA 인증서를 업데이트하였는가?	

제 2 절 Apache Web Server 보안 체크리스트

항목	상세설명	체크
보안패치	1. 웹서버가 운영되고 있는 운영체제의 최신 보안 업데이트 하였는가? 2. 현재 사용하는 웹서버 버전에 대한 최신 보안 업데이트하였는가? (흔히 최신 보안 업데이트는 사용 중인 Application과의 호환을 이유로 미루는 경향이 있는데, 보안패치를 하지 않아 Critical한 문제가 발생하는 경우가 종종 있으므로 개발장비에서 미리 테스트 후 반드시 적용하도록 함)	
운영	1. 웹서버 데몬이 최소권한(nobody 등)으로 운영되고 있는가? 2. 웹서버 Root 외부 디렉토리 접근권한통제를 하고 있는가? 3. 웹서버 설정파일과 주요데이터는 정기적으로 백업하고 있는가? 4. 백업된 데이터는 정상적으로 복구하여 검증하는 절차가 있는가? 5. Rewrite 룰을 이용한 침입탐지를 하고 있는가? 6. 보안강화를 위한 보안 모듈(mod_Security 등)을 사용하고 있는가?	
로깅	1. 웹서버 로깅(access_log, error_log)을 하고 있는가? 2. 로그파일에 적절한 ACL을 설정하였는가? 3. 로그파일은 주기적으로 백업하고 있는가?	
불필요 요소 제거	1. 설치 기본 디렉토리를 제거하였는가? (htdoc) 2. 불필요한 CGI 스크립트를 제거하였는가? 3. 불필요한 HTTP Method를 제거하였는가? (일반적으로 GET,POST,HEAD만 허용)	
안전한 설정 (httpd.conf)	1. 디렉토리 인덱싱이 불가능하도록 설정되었는가? - Options → ""Indexes"" 제거 2. 심볼릭 링크 기능을 해제하였는가? - Options → ""FollowSymLinks"" 제거 3. SSI를 이용한 실행권한을 제거하였는가? - Options → ""IncludeNoExec"" 추가 4. 실행권한이 필요한 경우 특정 디렉토리로 한정하였는가? (사용자가 임의로 업로드하는 디렉토리의 경우 실행권한 제거) - ScriptAlias /cgi-bin/ ""/usr/local/apache/cgi-bin/"" 5. 응답헤더에서 웹서버의 버전정보(배너)를 숨기는 설정을 하고 있는가? - ServerTokens ProductOnly 6. 사용자 지정 에러 페이지를 사용하고 있는가? (웹서버의 에러코드는 RFC2616의상태코드 정의부분에 명시되어 있음) (예) ErrorDocument 404 /errorpagedefined/usr404page.htm 7. 시스템 설정을 보호하고 있는가? <Directory /> AllowOverride None </Directory>	

주요정보 암호화	1. 민감한 정보는 인증서를 이용하여 SSL/TLS로 암호화하고 있는가? 2. Apache 서버에 RootCA 인증서를 업데이트하였는가?	

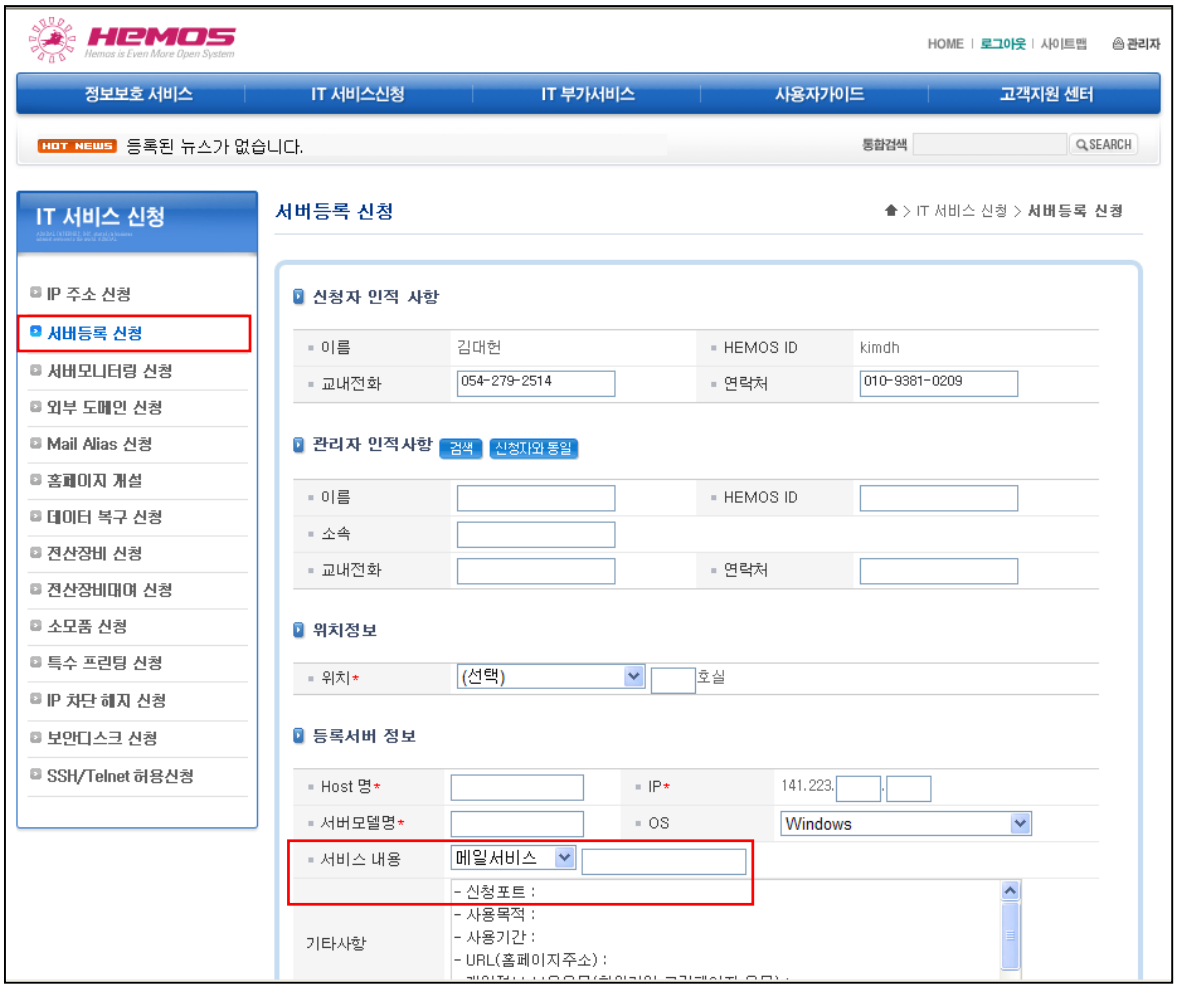
Appendix

웹서버 등록 절차 안내

웹서비스를 위한 독립 웹서버의 등록은 교내 정보보호 및 전산자원 보호를 위해 공용 웹서버(연구실, 센터 등의 웹사이트)에 한해서만 지원해드리며 아래 신청 절차를 통해 접수하고 있다.

※ 웹서버 등록은 신청 전에 반드시 웹사이트를 구축 완료하여 주시기 바랍니다.

1. 해모수(hemos.postech.ac.kr) – IT 서비스신청 – 서버등록 신청에서 신청
   - 서비스 내용에서 웹서비스 선택
   - 등록서버 정보 기타사항 반드시 기입(웹서비스 기본 포트 : 80)



2. 취약점 점검
   - 웹 취약점 점검을 통해 보안 취약점을 최소화하여 웹서버를 등록하고 있으며 취약점 조치가 안되면 등록이 불가함

3. 보안서버(SSL 인증서) 구축
   - 개인정보(회원가입, 로그인, 회원정보수정 등)를 포함하거나 수집하는 웹사이트는 반드시 보안서버를 구축
4. 서버 등록 및 DNS 등록
   - 취약점 점검 및 보안서버(SSL 인증서) 구축 완료 후 서버 등록
   - DNS 등록을 위해 HEMOS → IT 서비스신청 → IP 주소 신청 후 등록

서버등록 완료 시, 서버모니터링 서비스가 제공되며 10 일간 DOWN 시 자동 등록대상에서 제거 됩니다.
(사전 안내메일 발송됨)
서버모니터링 안내메일은 서버등록 시 입력한 관리자 정보를 기반으로 하므로 정보를 정확히 기재해주시기 바랍니다.

[참고자료]

[1] 홈페이지 개발 보안 가이드 - KISA
[2] 웹서버 보안취약점 대응가이드 – 교육사이버안전센터
[3] 공공기관 홈페이지 개인정보 노출방지 가이드라인 – 행정안전부
[4] 과학기술분야 정보시스템 보안 가이드 Part 2(윈도우) – 한국과학기술정보연구원
[5] 침해사고대응팀(CERT) 구축/운영 안내서 – 한국인터넷진흥원