WebKnight를 이용한 SQL Injection 공격 차단

KISA는 본 문서에서 언급한 WebKnight 및 해당 도구 개발사인 AQTRONIX와 어떠한 관계도 없으며, 국내 웹 해킹 피해 예방을 위해 공개 웹방화벽인 WebKnight를 보안 참고용으로 소개합니다.

2006. 02. 10

KrCERT/CC
인터넷침해사고대응지원센터

KISA 한국정보보호진흥원
Korea Information Security Agency

1. 개요

단순 홈페이지 해킹이 아닌 홈페이지 방문자들의 정보를 빼내 금전적인 이득을 취하고자 하는 홈페이지 해킹이 심각한 수준에 달하고 있다. 이는 해킹당한 업체가 피해기관이 되기도 하지만 해당 웹사이트를 신뢰하고 방문하는 수많은 네티즌들을 감염시키는 공격사이트이기도 하여 조치가 시급하다.

최근 윈도우즈 웹서버를 대상으로 발생되고 있는 해킹은 대부분 SQL Injection 공격이 그 원인이다. SQL Injection 취약점은 게시판, 공지사항 등에서 URL 인자에 대한 입력값을 검증하지 않음으로 해서 공격이 발생되는 웹 개발과정에서의 오류라고 할 수 있다. 대형 포털, 뉴스 사이트 등 수많은 국내 사이트들이 공격을 당해 웹 방문자들을 감염시키고 있지만, 이러한 악성코드 유포지로 이용되고 있는 사이트들은 취약점이 있음을 알고 있지만 제대로 조치를 하지 못해 수차례 다시 해킹을 당하는 경우를 많이 볼 수 있다. 인터넷침해사고대응지원센터의 분석에 의하면 국내 악성코드 경유지 또는 유포지 사이트 중 약 30% 가량이 2회 이상 재차 해킹을 당하고 있는 것으로 나타났다. 이는 SQL Injection 취약점 자체가 웹 프로그램의 소스 코드를 수정해야만 근본적으로 해결될 수 있는 문제이지만 운영 중인 웹 서버의 프로그램 수정이 쉽지 않기 때문이다.

웹 시스템 구축 이후 문제점을 수정하기 보다는 설계·개발 단계에서 보안을 고려하여 개발되는 것이 바람직하다. 인터넷침해사고대응지원센터에서는 홈페이지 개발시 고려하여야 하는 보안 사항과 웹언어별 사례를 제공하고 있으므로 이를 참고하여 개발하기 바란다.

o 홈페이지 개발 보안 가이드 다운로드 :
http://www.kisa.or.kr/news/2005/announce_20050427_submit.html

향후, 인터넷침해사고대응지원센터에서는 SQL Injection 공격, 업로드/다운로드 공격, XSS 공격 등 대표적인 웹공격에 대비할 수 있는 표준 웹애플리케이션 보안 템플릿도 제공할 계획이다.

하지만, 이미 구축되어 있는 웹사이트들은 대부분 보안을 고려하여 개발되지 않았으며, 이를 단기간에 수정하는 것도 쉽지는 않아 많은 국내 웹사이트들이 재차 해킹을 당하고 있다. 따라서, 근본적인 대책인 웹 소스 수정이 어려울 경우, Secure하지 못한 웹서버를 보완할 수 있는 추가적인 방안이 필요하다. MS사에서는 IIS 웹서버의 보안성을 강화시켜 주기 위해 IISLockdown, URLScan 등과 같이 도구를 제공해 주고 있다.

IISLockdown은 웹 서버를 보호하기 위한 과정을 대부분 자동화할 수 있는 도구로 서버의 용도에 따라 유형별로 다양한 보안기능을 해제하거나 보호할 수 있는 사용자 템플릿을 제공해 준다.

http://msdn.microsoft.com/library/default.asp?url=/library/en-us/secmod/html/secmod113.asp

URLScan은 웹 사이트 관리자가 서버에서 처리 가능한 웹 요청을 제한할 수 있는 ISAPI(Internet Server Application Program Interface) 필터로써 특정 웹 요청을 제한하여 잠재적으로 유해한 웹 요청이 서버에 도달하기 이전에 차단함으로써 공격을 예방한다.

http://msdn.microsoft.com/library/default.asp?url=/library/en-us/secmod/html/secmod114.asp
http://www.microsoft.com/technet/security/tools/urlscan.mspx

하지만, 아쉽게도 IISLockdown이나 URLScan도 DB Query 문장을 필터링하지는 못하여 최근 발생되고 있는 SQL Injection 공격을 차단할 수는 없다.

최근 웹 공격이 심각한 수준에 이르러 국내·외 상용 웹방화벽들도 많이 출시되었다. 다양하고 정교한 웹공격을 기존의 네트워크 방화벽이나 침입탐지시스템가 방어하는데 한계가 있다. 웹방화벽은 SQL Injection 등 웹공격에 특화된 보안 솔루션이므로 웹방화벽의 도입도 검토할 필요가 있다. 그러나, 기업에서 경제적인 문제로 인해 상용 웹방화벽 도입이 어려운 경우가 많으므로 본 고에서는 공개 웹방화벽인 WebKnight를 통해 SQL Injection 등 웹공격에 대해 방어하는 방안을 살펴보고자 한다.

WebKnight는 GNU 공개 라이센스 원칙을 따르는 공개 소프트웨어로써 모든 기업이나 개인이 자유로이 사용할 수 있다. 하지만, 대부분의 공개 소프트웨어와 마찬가지로 WebKnight도 상용 웹방화벽에 비해 인터페이스나 메뉴얼 등 사용자 편의성이 부족하고, 지속적인 유지보수도 어렵다는 단점이 있다. WebKnight도 2003년 11월에 v1.3 버전이 릴리즈된 후 업데이트가 없다.

그러나, 이 도구는 SQL Injection을 포함한 다양한 웹공격에 대해 차단할 수 있는 프레임을 제공해 주고 있고, 현재 많이 사용되고 있는 IIS5, IIS6에서도 아무런 문제없이 운영이 가능하여 최근의 웹공격 차단에 상당히 많은 도움을 줄 수 있다. 물론, WebKnight의 잘못된 설정은 정상적인 웹 요청까지 차단할 수 있으므로 충분한 커스트마이징 과정은 웹서버 관리자의 몫임을 명심하여야 할 것이다.

본 고에서는 WebKnight의 주요 기능을 살펴보고 설치 및 커스트마이징 방법을 소개하고, 실제 공격을 얼마나 잘 차단하는지 테스트한 결과를 소개한다.

2. WebKnight 개요

WebKnight는 AQTRONIX사(http://www.aqtronix.com/)에서 개발한 IIS 웹서버에 설치할 수 있는 공개용 웹 방화벽이다. WebKnight는 ISAPI 필터 형태로 동작하며, IIS 서버 앞단에 위치하여 웹서버로 전달되기 이전에 IIS 웹서버로 들어온 모든 웹 요청에 대해 웹서버 관리자가 설정한 필터 룰에 따라 검증을 하고 SQL Injection 공격 등 특정 웹 요청을 사전에 차단함으로써 웹서버를 안전하게 지켜준다. 이러한 룰은 정기적인 업데이트가 필요한 공격 패턴 DB에 의존하지 않고 SQL Injection, 디렉토리 traversal, 문자 인코딩 공격 등과 같이 각 공격의 특징적인 키워드를 이용한 보안필터 사용으로 패턴 업데이트를 최소화하고 있다. 이러한 방법은 알려진 공격 뿐만 아니라 알려지지 않은 공격으로부터도 웹서버를 보호할 수 있다.

또한, WebKnight는 ISAPI 필터이기 때문에 다른 방화벽이나 IDS에 비해 웹서버와 밀접하게 동작할 수 있어 많은 이점이 있다. MS의 URLScan과 마찬가지로 ISAPI 필터로써 inetinfo.exe 안에서 동

작하므로 오버헤드가 심하지 않다. 해킹당한 한 웹사이트에 WebKnight를 적용하여 테스트한 결과 안정적인 웹서버 운영으로 인해 웹서버 속도가 오히려 빨라진 것을 느낄 수 있었다. 하지만 다량의 웹 트래픽이 발생되는 사이트에서는 사전에 충분한 검증을 거친 후에 적용할 필요는 있다.

다음은 WebKnight의 주요 특징이다(http://www.aqtronix.com/?PageID=99 참조).

o 낮은 보유 비용(Total Cost of Ownership)
WebKnight는 윈도우즈 인스톨러 패키지와 원격 설치 스크립트로 설치가능해 사내에서 쉽게 WebKnight를 채택할 수 있다. 또한 WebKnight 설정을 바꾸기 위해 그래픽 사용자 인턴페이스를 제공한다.

o 운영 중 업데이트 가능
일부 설정의 변경을 제외하고 대부분의 설정 변경은 웹서버의 재가동을 요구하지 않아, 웹 사용자들에 대한 어떠한 서비스 장애 없이 설정을 변경할 수 있다. 성능상의 이유로 매 1분마다 이러한 변경을 탐지하여 적용한다.

o SSL 보호(protection)
다른 전통적인 방화벽과는 달리 WebKnight는 ISAPI 형태로 IIS의 일부로 동작하므로 HTTPS 상의 암호화된 세션들도 모니터링 및 차단할 수 있다.

o Logging
기본적으로 차단된 모든 요청에 대해 로그를 남기고, 로깅 전용 모드로 운영할 경우 추가적으로 모든 허용된 요청에 대해서도 로그를 남길 수 있다. 로깅 전용 모드는 공격을 차단하지는 않고 로그 파일에서 공격 사실을 조사하는데 도움을 줄 수 있다.

o HTTP Error Logging
WebKnight는 웹서버로부터 HTTP 에러들을 로그할 수 있도록 설정할 수 있다. 이 방법으로 '404 Not Found'와 같은 일반적인 에러나 '500 Server Error'와 같이 보다 심각한 로그들도 기록할 수 있다. 에러 로그를 이용하여 공격을 탐지하거나 깨진 링크를 발견하거나 잘못된 설정도 쉽게 발견할 수도 있다.

o 웹기반 애플리케이션과의 호환성
WebKnight는 Frontpage Extensions, WebDAV, Flash, Cold Fusion, Outlook Web Access, SharePoint 등과도 호환이 잘 이루어진다.

3. WebKnight 설치 및 제거

가. WebKnight 설치

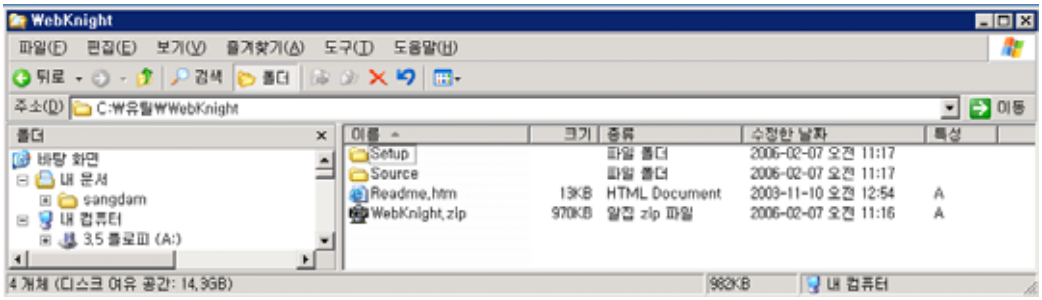
WebKnight는 윈도우즈 인스톨러를 이용한 설치, install.vbs 스크립트를 이용한 설치, 수동 설치 등 3가지 방법으로 설치할 수 있다.

웹호스팅 서버 등 하나의 웹서버 내에 다수개의 사이트가 운영되는 경우 웹서버 전체에 필터를 적용(글로벌 필터)할 수도 있으며, 개별 웹사이트별로 서로 다른 룰에 의해 필터를 적용(사이트 필터)할 수도 있다.

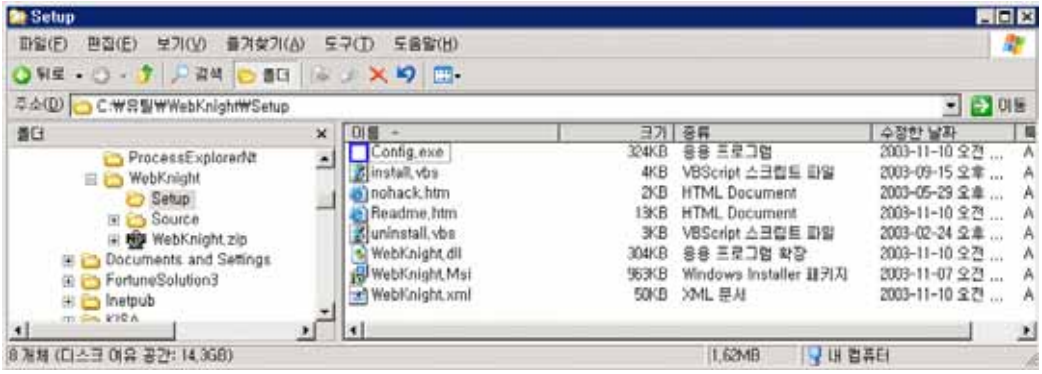
윈도우즈 인스톨러와 install.vbs 스크립트를 이용한 설치시에는 기본적으로 글로벌 필터가 적용되는데 설치 과정은 다음과 같다.

① 아래 URL에서 WebKnight 1.3(2003.11.10 릴리즈)을 다운로드 받는다.
http://www.aqtronix.com/downloads/WebKnight/2004.02.01/WebKnight.zip

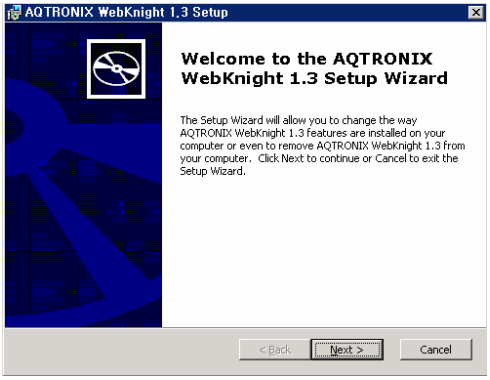
② 압축을 해제하면 아래와 같은 폴더와 파일이 생성된다.



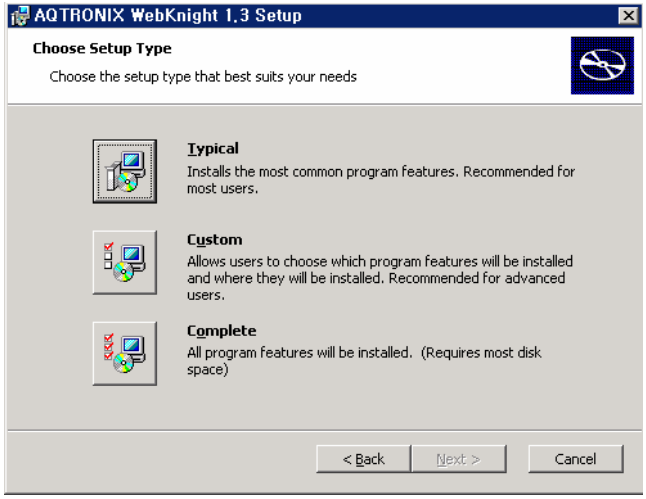
③ Setup 폴더로 이동하여 설치 방법을 선택할 수 있다.



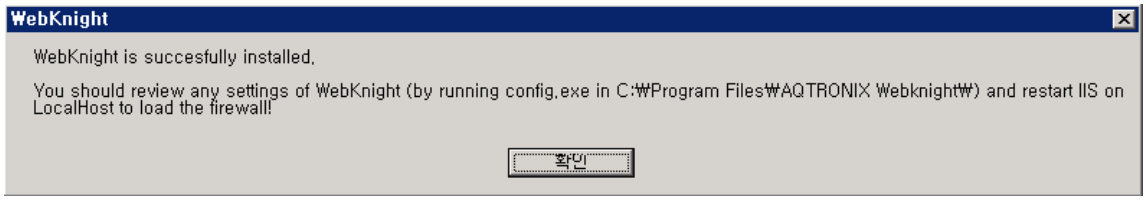
④ WebKnight.msi 파일을 더블클릭하여 윈도우즈 인스톨러를 동작시켜 WebKnight를 설치할 수도 있고, install.vbs 스크립트를 더블클릭하여 설치할 수도 있다.(삭제시에는 uninstall.vbs 파일 실행) 다음 그림은 IIS 6.0에서 WebKnight.msi 파일 실행하여 윈도우즈 인스톨러에 의해 설치되는 화면이다.



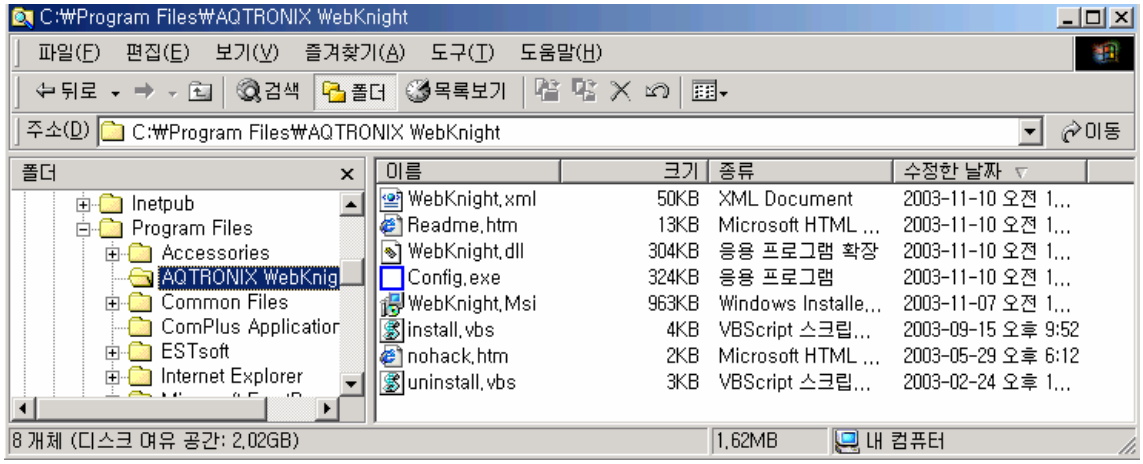
⑤ 라이센스 동의 후 설치 타입 선택화면이 나타나는데, 일반적으로 “Typical"을 선택한다.



⑥ 이후 자동 설치과정이 진행되며 설치가 완료되면 다음과 같은 메시지가 나타난다.

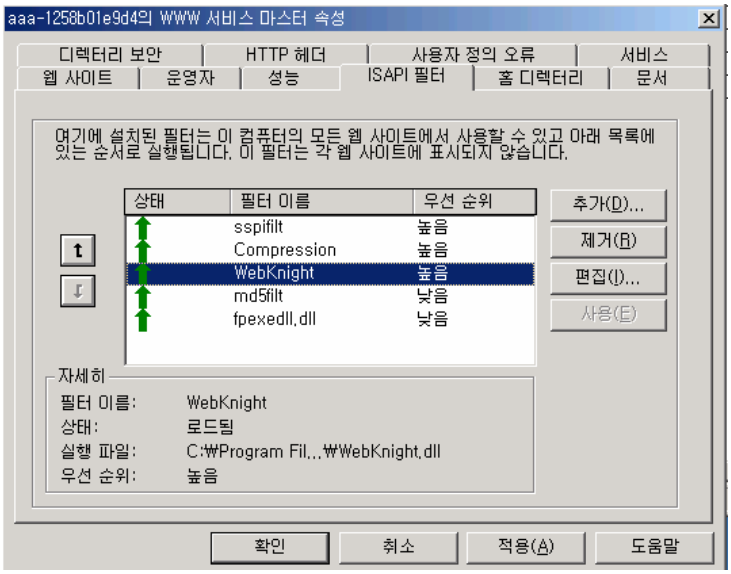


⑦ 기본적으로 C:\\Program Files\AQTRONIX\WebKnight 폴더에 설치가 완료된다.
이 폴더 내에 필터 역할을 하는 DLL파일(WebKnight.dll)과 향후 커스트마이징을 위해 필요한 설정실행 파일(Config.exe), 로그파일(IIS 재가동 후 생성) 등이 위치하고 있으므로 이 폴더의 위치를 기억할 필요가 있다.



⑧ IIS 웹서버를 재가동 한다.

⑨ IIS 웹서버를 재가동 후에 정상적으로 설치가 완료되었을 경우 다음과 같이 웹사이트 동록 정보의 “ISAPI 필터”에 다음과 같이 WebKnight 필터가 정상적으로 적용이 된 것을 확인할 수 있다.



위 과정을 통해 WebKnight의 설치는 간단히 수행할 수 있다. 만일 다수개의 웹사이트가 운영되어 각 사이트마다 필터링 룰을 달리 적용하거나 자동 설치가 어려운 경우 다음과 같이 수동으로 설치할 수 있다.

■ 글로벌 필터로 수동 설치

① 압축 해제 후 생성되는 Setup 폴더를 C:\Program Files\AQTRONIX WebKnight와 같은 서버 내의 로컬 폴더를 생성하고 여기에 복사한다.
② 인터넷 정보 서비스를 연다.
③ 서버 이름(사이트 이름이 아님)에서 우측 마우스를 클릭하여 "등록정보"를 선택한다.
④ 마스터 속성 리스트에서 "WWW 서비스"를 선택하고, "편집" 버튼을 누른다.
⑤ "ISAPI 필터" 탭을 선택하고 "추가" 버튼을 클릭한다.
⑥ "필터 등록 정보"가 나타나면 필터 이름과 실행 파일 경로를 입력한다.
(예를들어, 필터 이름 : WebKnight, 실행 파일 경로 : C:\Program Files\AQTRONIX WebKnight\WebKnight.dll)
⑦ "OK" 버튼을 누르고 대화상자를 빠져 나간다.
⑧ IIS를 재 가동한다.

■ 사이트 필터로 수동 설치

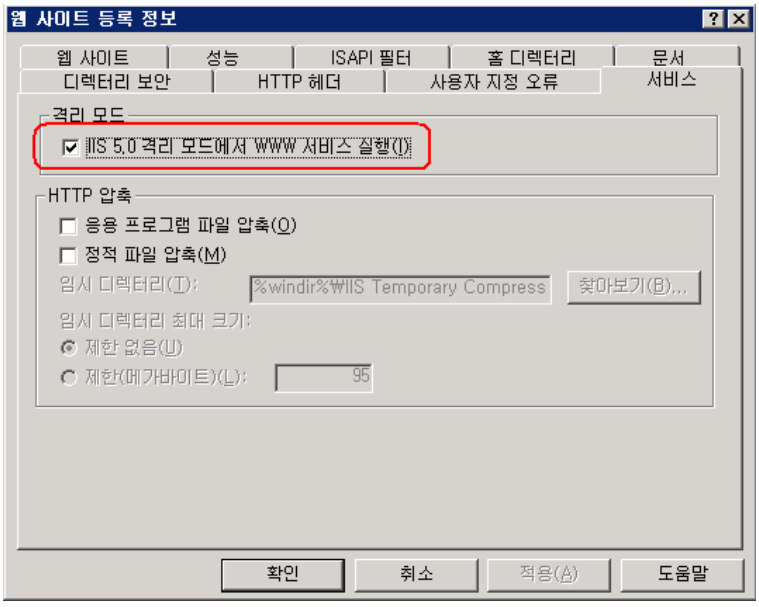
① 압축 해제 후 생성되는 Setup 폴더를 C:\Program Files\AQTRONIX WebKnight\W3SVC1 과 같은 서버내의 로컬 폴더를 생성하여 여기에 복사한다.(단, 각 WebKnight 설치를 위한 unique한 폴더를 가져야 한다.)
② 인터넷 정보 서비스를 연다.
③ 사이트 이름(서버 이름이 아님)에서 우측 마우스를 클릭하여 "등록정보"를 선택한다.
④ "ISAPI 필터" 탭을 선택하고 "추가" 버튼을 클릭한다.
⑤ "필터 등록 정보"가 나타나면 필터 이름과 실행 파일 경로를 입력한다.
(예를들어, 필터 이름 : WebKnight, 실행 파일 경로 : C:\Program Files\AQTRONIX WebKnight\W3SVC1\WebKnight.dll)
⑥ "OK" 버튼을 누르고 대화상자를 빠져 나간다.

⑦ Setup 폴더 아래의 config.exe 파일을 실행해서 “Global Filter Capabilities" 섹션에서 ”Is Installed As Global Filter“의 체크를 해제한다.

⑧ IIS를 재 가동한다.

■ IIS 6.0에서 WebKnight 구동시 주의사항

① 글로벌 필터로 구동하기 위해서는 “IIS 5.0 격리 모드”로 IIS를 구동하여야 한다.
IIS 6.0은 기본적으로 “작업자 프로세스 모드”로 구동이 되고 있는데 “IIS 5.0 격리 모드”로 변경할 경우에는 동일 환경의 테스트 서버에서 사전에 검증을 거치는 것이 바람직하다. 왜냐하면 만약 해당 사이트가 IIS 5.0에는 없는 IIS 6.0의 새로운 기능을 사용하고 있을 경우 해당 기능과 관련된 프로그램들이 정상적으로 동작하지 않거나 오류를 발생시킬 수 있기 때문이다.
또한, “IIS 5.0 격리 모드”로의 변경은 IIS의 재시작을 필요로 한다.



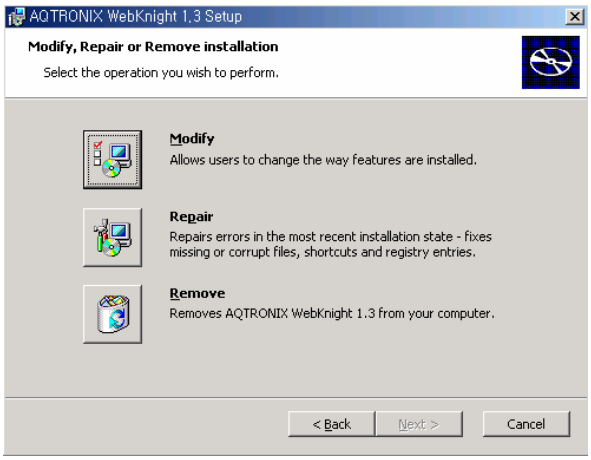
② IIS 6.0에서 “작업자 프로세스 모드"로 계속 구동하고 각 프로세스별로 unique한 로그가 필요한 경우에는 config.exe를 실행하여 다음과 같이 설정한다.

- “Global Filter Capabilities“ 섹션에서 ”Is Installed As Global Filter“을 체크하지 않는다.
- “Logging" 섹션에서 ”Per Process Logging“을 체크한다.
- “NETWORK SERVICE” 계정이 WebKnight 폴더와 하위 폴더들의 퍼미션을 바꾸었는지 확인한다.

나. WebKnight 제거

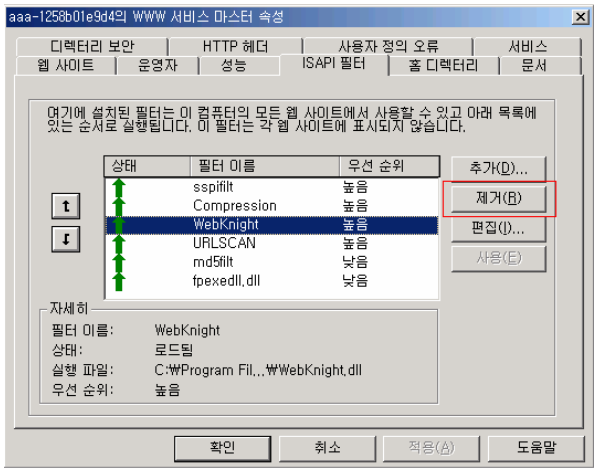
만일 WebKnight를 제거하고자 할 경우 다음과 같은 3가지 방법 중 하나를 선택하여 WebKnight를 제거한 후 IIS를 재가동한다.

① 윈도우즈 인스톨러를 이용한 자동 제거
(WebKnight.msi 실행)
디폴트 경로에 설치되어 있는 경우 C:\Program Files\AQTRONIX WebKnight\WebKnight.msi를 실행하면 오른쪽 그림과 같이 “Modify", "Repair", "Remove" 화면이 나타나는데, 이 중 ”Remove"를 선택하면 자동으로 WebKnight가 제거된다.



② 스크립트를 이용한 자동 제거 (uninstall.vbs 실행)
디폴트 경로에 설치되어 있는 경우 C:\Program Files\AQTRONIX WebKnight\uninstall.vbs를 실행하면 자동으로 WebKnight가 제거된다.

③ 수동 제거
수동 설치과정과 마찬가지로 인터넷 정보 서비스를 열고 글로벌 필터 또는 사이트 필터에 따라 서버 이름 또는 사이트 이름을 선택한 후 우측 마우스를 클릭하여 “등록정보”→“편집”→“ISAPI 필터” 탭을 선택하여 WebKnight 항목을 선택한 후 “제거” 버튼을 누른다.



상기와 같이 WebKnight를 제거한 후 변경사항을 반영하기 위해서는 IIS를 재가동하여야 한다.

4. 설정 커스트마이징

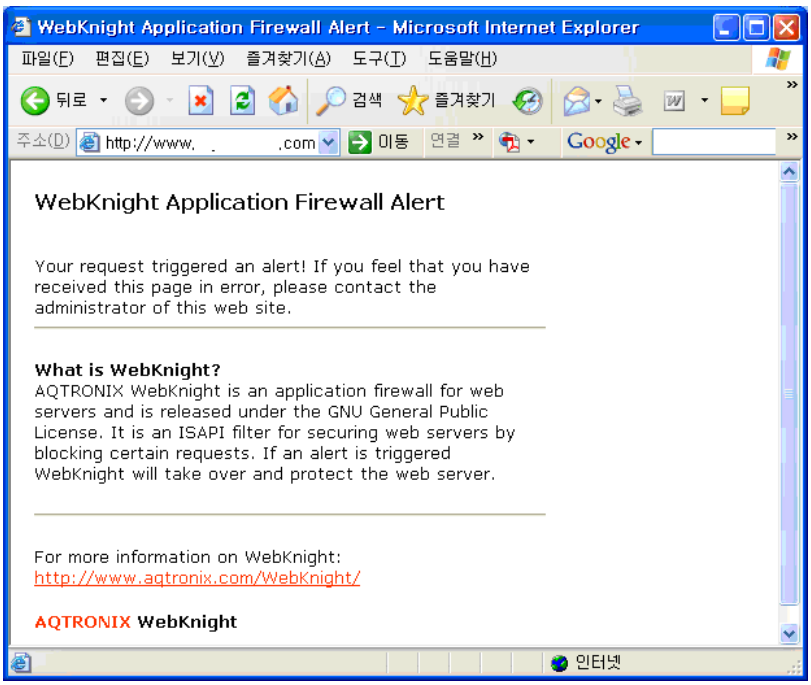
WebKnight는 SQL Injection 공격차단, 허용하지 않는 파일 또는 확장자에 대한 접속 차단 등 웹 공격에 대해 대단히 다양한 차단기능을 제공해 주고 있다. 또한 기본적으로 이러한 차단기능이 설정되어 설치와 동시에 적용이 되는데 이 차단기능이 정상적인 웹 접속을 차단할 수도 있다. 따라서 설치이후 자신의 웹사이트 환경에 맞게 적절하게 커스트마이징하는 과정을 반드시 거쳐야 한다. 실제 설치보다는 커스트마이징에 많은 노력과 시간을 들여야만 한다. 설정과정을 통해 오히려 웹 공격의 다양한 패턴을 익힐 수 있는 기회도 될 수 있을 것이다.

먼저, WebKnight 설치 이후 해당 웹사이트에 방문해서 정상적으로 웹요청 및 응답이 이루어지는지 확인을 하고, 접속이 차단될 경우 WebKnight의 로그를 참조하여 어떠한 룰에 의해 요청이 차단되었는지 찾아 이 룰을 수정하여야 한다.

디폴트 설치시 로그파일의 위치와 설정프로그램은 다음과 같다.

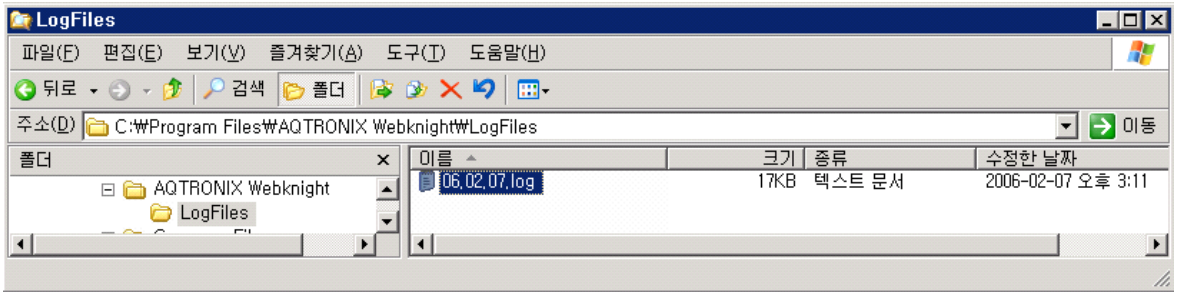
o 로그파일 : C:\Program Files\AQTRONIX WebKnight\LogFiles\YYMMDD.log
o 설정프로그램 : C:\Program Files\AQTRONIX WebKnight\config.exe

WebKnight 설치 후 웹 접속시 다음과 같은 경고 화면이 뜰 수 있다.



이 화면은 WebKnight에서 필터 룰에 의해 차단을 시킨 후 웹접속자에게 보내는 기본 경고화면이다.
정상적인 웹 요청을 했는데도 불구하고 이와같이 차단된다면 로그파일을 열어 "BLOCKED" 메시지를 확인하고 어느 룰에서 차단되었는지 찾아 설정파일에서 이를 해제하여야 한다. 디폴트 설치의 경우

WebKnight의 로그파일은 설치 후 IIS 웹서버를 재가동하게 되면 "C:\Program Files\AQTRONIX WebKnight\LogFiles" 폴더가 생성되고 그 하위에 일자별로 로그파일이 생성된다.



기본적인 로그파일의 각 필드는 다음과 같다.

Time ; Site Instance ; Event ; Client IP ; Username ; Additional info about request(event specific)

정상적인 웹 접속이 차단되어 로그파일을 분석해 보니 다음과 같은 로그가 남았다.

05:57:42 ; W3SVC31 ; OnPreprocHeaders ; xxx.xxx.207.85 ; ; GET ; /admin/img/deffortune.jpg ; BLOCKED: '/admin' not allowed in URL ; HTTP/1.1 ; ASPSESSIONIDAQDBDDAD=NACAJJBAPJACHHPHNIPGDKCH

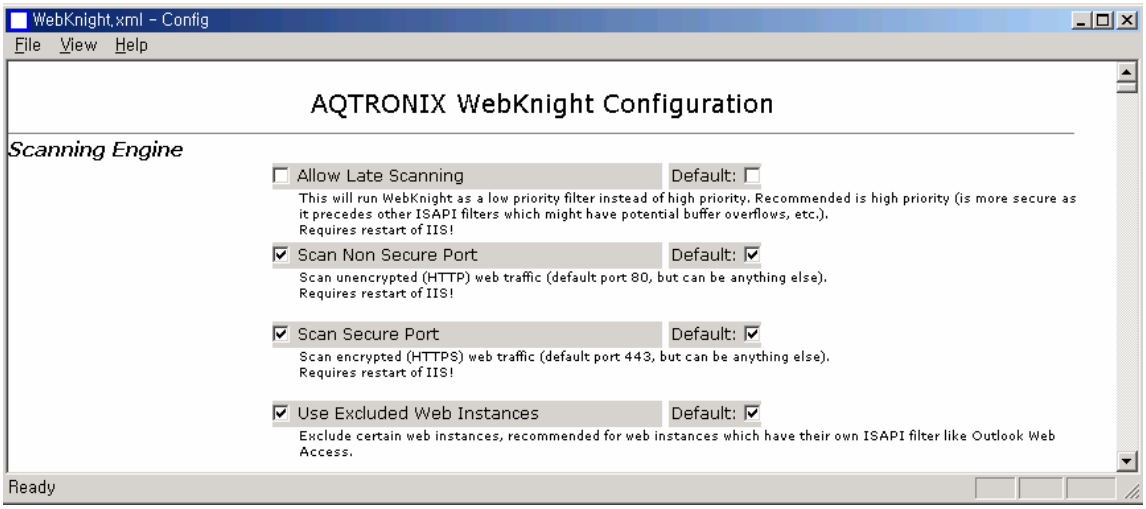
위의 로그를 보면 "/admin" 폴드에 대한 접속을 허용하지 않도록 WebKnight 룰이 설정되어 있는데, /admin/img/deffortune.jpg 파일에 접속하고자 하여 웹접속이 차단된 것을 알 수 있다. 이는 홈페이지 초기화면 구성시 /admin 폴드에서 그림파일을 가져오도록 설계되어 있어 발생되었는데, 일반사용자들이 접근하는 화면에 /admin 폴드의 컨텐츠를 포함하지 않도록 변경할 필요가 있다. 또는, WebKnight의 설정을 변경하여 /admin 폴더에의 접속을 허용할 경우 접속이 차단되는 상황을 막을 수 있다. 이처럼 로그파일을 통한 커스트마이징 과정은 현재의 웹서버 설계상의 문제도 파악하여 개선하도록 도와줄 수도 있다.

다음 FAQ에는 WebKnight의 설치와 환경설정, 로그파일 분석시 자주 발생될 수 있는 문제와 궁금증에 대해 질의·응답식으로 정리되어 있으므로 참고하기 바란다.

http://www.aqtronix.com/?PageID=114

로그파일 해석시 기본 설정의 로그 시간대는 GMT/UTC로 한국 시간대인 GMT+09 보다 9시간 늦으므로 로그 분석시 이를 감안하여야 한다.(설정에서 "USE GMT"를 체크하지 않음으로 시스템 시간과 동기화시킬 수 있다.)

설정 변경은 config.exe 파일을 실행하여 GUI 인터페이스를 통해 설정할 수 있다.



config.exe를 통해 WebKnight의 다양한 필터링 기능을 설정할 수 있는데 다음과 같은 설정을 할 수 있다. 설치경험을 통해 환경설정과정에서 웹서버 관리자가 유심히 확인해야 되는 부분에 대해 "확인 사항"에 의견을 넣었으니 참고하기 바란다.

구 분	기 능	확인 사항
Scanning Engine	암호화 포트(HTTPS), 비암호화 포트(HTTP)에 대한 모니터링 기능 설정	
Incident Response Handling	공격 발생시 WebKnight가 어떻게 행동할지를 결정하며, 기본적으로 경고화면인 nohack.htm으로 redirect하고 웹 요청을 차단하지만, 차단하지 않고 로그만 남기게 할 수도 있음	
Logging	로깅 여부, 로그 시간대, 로그 항목(클라이언트 IP, 사용자 명 등) 등을 설정	"USE GMT" 항목 disable 권고(시스템 시간 사용)
Request Limits	컨텐츠 길이, URL 길이, 쿼리스트링 길이 등을 제한	
URL Scanning	URL Encoding 공격 차단, 상위 패스(..) 차단, URL 백슬래쉬(\) 차단, URL 인코딩(%) 차단, 특정 URL 스트링 차단 등 URL 입력 모니터링 및 차단	"URL Denied Sequences" 항목 확인 필요
Mapped Path	경로에 상위 패스, 백슬래쉬(\) 등 차단 및 로컬 파일시스템의 허용하는 경로 정의	"Allowed Paths"에서 웹 컨텐츠가 있는 위치 확인 및 지정 필요
Requested File	차단시킬 파일 목록과 차단·허용할 파일 확장자 정의	정상적인 요청이 차단될 수 있으므로 반드시 확인필요
Headers	서버 헤더 정보 변경, 특정 헤더 차단 등 설정	
Methods	허용 또는 차단할 Method를 결정(예 : GET, HEAD, POST은 허용하고 DELETE, PUT 등은 차단)	

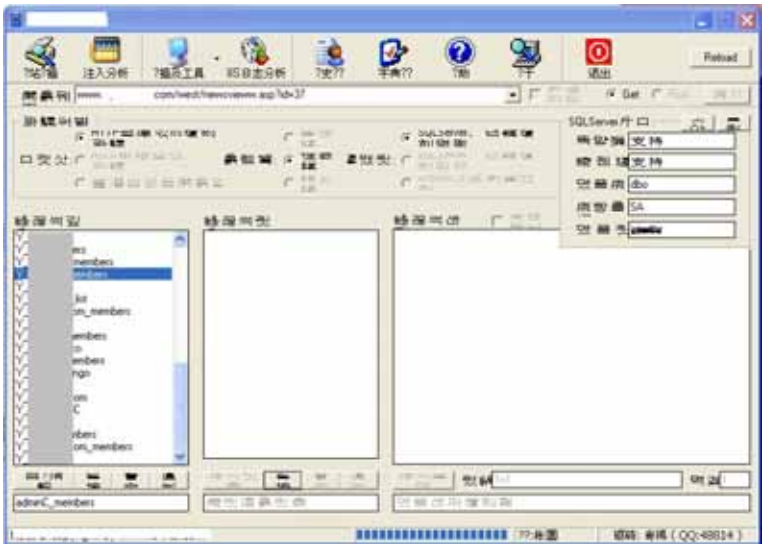
Querystring	특정 query 스트링(xp_cmdshell, cmd.exe 등) 차단, query 스트링에서 SQL Injection 차단 등 설정	
Global Filter Capabilities	글로벌 필터 적용 여부, 특정 헤더 스트링(xp_cmdshell, cmd.exe 등) 차단 등 결정	
SQL Injection	SQL Injection 공격에 이용되는 키워드 정의(' ; select insert xp_ 등)	공격에 이용될 수 있는 수십개의 키워드가 정의되어 있으나 확장저장프로시져의 사용 유무 등을 고려하여 추가/삭제 필요
Web Applications	WebDAV, IISADMPWD 등 웹애플리케이션의 허용유무 결정	기본적으로 모두 사용하지 않는 것으로 설정되어 있음

위 대부분의 설정 변경 사항은 IIS의 재가동 없이 바로 적용이 되지만 일부 항목은 재가동을 하여야만 적용이 되는 것도 있으므로 IIS의 재가동 여부를 확인할 필요가 있다. 만일 설정을 잘못 변경하여 다시 디폴트 설정으로 바꾸기 위해서는 WebKnight.xml 파일을 삭제한 후에 웹서버를 재시작하면 된다(디폴트 상태의 WebKnight.xml 파일이 새로 생성됨).

5. 모의 공격 및 공격차단 확인

WebKnight의 커스트마이징으로 정상적으로 설치가 완료가 되었을 경우 웹 공격이 정상적으로 차단되고 있는지 확인해 볼 필요가 있다.

다음 그림은 WebKnight 설치 이전에 해당 웹서버가 SQL Injection 공격에 취약하여 공격툴에 의해 DB 접근이 가능하고 DB 계정 및 테이블이 노출되고 있는 화면이다.



하지만, WebKnight의 설치 이후 동일한 공격툴을 이용하여 테스트한 결과 공격은 실패하였으며, 웹서버의 WebKnight 로그파일에 공격 차단 로그가 남았다.

06:13:40 ; W3SVC31 ; OnPreprocHeaders ; xxx.xxx.151.24 ; ; GET ; /west/newsvieww.asp ; id=37'%20and%20user%2Bchar(124)=0%20and%20''=' ; **BLOCKED: possible SQL injection in querystring** ; HTTP/1.1 ; ASPSESSIONIDAQDBDDAD=EDIAJJBAFOHJCEKKEMBNCEJD
06:13:40 ; W3SVC31 ; OnPreprocHeaders ;xxx.xxx.151.24 ; ; GET ; /west/newsvieww.asp ; id=37%25'%20and%20user%2Bchar(124)=0%20and%20'%25'=' ; **BLOCKED: possible SQL injection in querystring** ; HTTP/1.1 ; ASPSESSIONIDAQDBDDAD=EDIAJJBAFOHJCEKKEMBNCEJD

SQL Injection 공격이외에도 취약한 CGI 공격, 디렉토리 traversal 공격 등 다양한 웹 공격이 차단되는 것을 확인할 수 있었다. WebKnight가 정상적으로 공격을 차단하고 있음을 확인하면 운영에 들어가는데, 운영하는 과정에서도 주기적인 WebKnight의 로그 확인이 필요하다. 로그 확인을 통해 어떠한 공격시도가 일어나고 있는지 확인하고 적절한 대응을 하여야 한다.

지금까지 공개 웹방화벽인 WebKnight를 이용한 SQL Injection 차단 방안에 대해 소개하였다.
WebKnight를 실제 운용되고 있는 취약한 웹서버에 적용시켜 본 결과 훌륭한 공격 차단효과를 확인할 수 있었는데, 상용 웹 보안도구의 도입이 여의치 않은 중소규모의 웹사이트에서 적용하기에 적절할 것으로 보여진다.

웹 보안의 기본은 안전하게 코딩된 웹 프로그램에 있음을 명심하여야 할 것이다.
홈페이지 개발보안 가이드, 표준 웹애플리케이션 보안 템플릿 등을 참고하여 웹 애플리케이션 설계단계에서부터 안전하게 개발하는 것이 가장 우선시 되어야 할 것이고, 부가적인 보안 조치로 WebKnight를 활용하기 바란다.