

APACHE HTTP Server





INDEX

1 APACHE SERVER란?

2 Windows 에서 구축법

3 Linux 에서 구축법

4 Mac 에서 구축법

5 구성 설정



INDEX

▲ 6 보안강화

▲ 7 커뮤니티

▲ 8 활용 사례

▲ 9 국내,해외 기술지원

▲ 10 기타

1. APACHE SERVER란?

▶ 1.1 APACHE SEVER 연혁

- Apache HTTP Server는 오픈 소스 소프트웨어 그룹인 아파치 소프트웨어 재단에서 만드는 웹 서버 프로그램이다. 팀 버너스 리가 만든 최초의 웹 서버 프로그램인 "NCSA HTTPd"를 기반으로 만들어졌다.

버전		공개일
Apache	1.3	1998년 6월 6일
Apache2	2.0	2002년 4월 6일
	2.2	2005년 12월 1일
	2.4 (현재 2.4.41)	2012년 2월 21일

1. APACHE SERVER란?

- 팀 버너스 리의 NCSA HTTPd는 유닉스 기반으로 만들어졌기에, 아파치 HTTP 서버는 NCSA HTTPd를 리눅스에서도 돌리는 것을 목표로 만들어진 프로그램이다. 그 이후 리눅스와 함께 퍼져나갔고, 리눅스가 서버 OS의 최다 점유율을 차지하자 아파치도 자연스럽게 최다 점유율을 차지하게 되었다. 현재 거의 모든 리눅스 배포판이 이 아파치를 지원한다.
- 윈도우 7에서는 1.3에서 2.2 버전까지 지원한다.
- 윈도우 10은 현재 모든버전을 지원한다.

1. APACHE SERVER란?

▶ 1.2 APACHE SEVER 특징

- 현재 시장 점유율은
2019년 6월 기준으로 Nginx가 9%차이로 점차 점유율을 높이는 중이다.
아파치(29.39%), Nginx(20.55%), Google(8.36%), Microsoft(5.97%)
- 라이선스는 독자적인 라이선스로 자유로운 수정 및 재배포가 가능한 아파치 라이선스 2.0을 사용한다.
- 아파치 웹서버는 무료 오픈소스이며 거의 모든 운영체제에서 사용가능하다
- 아파치를 포함한 APM(Apache + PHP + MySQL)은 웹서버구성의 기본으로 받아들여질 정도이다.

1. APACHE SERVER란?

- 아파치 서버는 호환성, 이식성, 확장성, 안정성이 높아서 웹서버에서 많이 사용한다.
- 아파치는 요청 당 프로세스 또는 스레드가 처리하는 구조이며, 요청이 많아지면 CPU와 메모리 사용량이 높아져 성능이 저하될수있다.
- 이 문제들은 Keep Alive를 활성화함으로 해결할 수 있지만 대량 접속시에 효율이 급격히 떨어지는 문제점이 발생한다.
- 이를 개선하기 위해 아파치2.4버전부터는 리소스 사용량이 더 적은 Event MPM을 적용하게되었다.

1. APACHE SERVER란?

- 2.4에서 Event MPM을 적용시켰음에도 Event Driven을 사용하여 리소스 사용량이 적은Nginx보다 떨어지는 성능과, 대량접속에도 적은 리소스를 사용하며 빠르게 서비스를 요구하는 웹사이트가 늘어남으로 인해서 아파치는 하락세를 타고있으며, 경량화서버인 Nginx가 시장 점유율을 점차 늘려가고있다.
- 그럼에도 안정성,확장성,호환성 면에서는 더 우세한 측면이 있기에 현재 시장 점유율에서도 Nginx를 앞서고 있다.

2. Windows 에서의 구축법

▶ 2.1 Windows에 대한 요구사항 (2.4 기준)

아파치 2.4을 실행하기위한 기본 Windows 플랫폼은 Windows NT이다. 바이너리 설치프로그램은 Intel과 AMD와 같은 x86 개열 프로세서에서만 동작한다. 아파치는 Windows 9x에서 충분히 검사하지 않았기때문에 절대로 실제 서비스에 사용하지 않길 바란다.

운영체제에 설치한 TCP/IP 네트워크가 동작해야 한다. Windows 10 이하 제품군은 이제 지원하지 않는다.

Windows NT 4.0을 사용한다면 서비스팩 4의 TCP/IP 문제와 Winsock 문제가 다음 서비스팩에서 해결되었기때문에, 서비스팩 6을 설치하길 적극 권한다.

2. Windows 에서의 구축법

▶ 2.1.1 Windows용 아파치의 주된 차이점(2.4 기준)

- Windows용 아파치는 다중 쓰레드 방식을 사용하기때문에, 유닉스와 달리 요청마다 다른 프로세스를 사용하지 않는다. 대신 아파치 프로세스는 항상, 부모 프로세스와 요청을 처리하는 자식 프로세스, 2개이다. 자식 프로세스에 있는 여러 쓰레드들이 요청들을 처리한다.
- 프로세스 관리 지시어도 다르다:
-
- 파일명을 아규먼트로 받는 지시어는 유닉스 파일명이 아닌 Windows 파일명을 사용해야 한다. 그러나 아파치 내부에서 유닉스식 이름을 사용하기때문에 '₩'가 아닌 '/'를 사용해야 한다. 드라이브 문자를 사용할 수 있다. 드라이브를 지정하지 않으면 아파치 실행파일이 있는 드라이브를 사용한다.

2. Windows 에서의 구축법

▶ 2.1.1 Windows용 아파치의 주된 차이점(2.4 기준)

- Windows NT라면 아파치 시작시 발생한 오류를 Windows 이벤트 로그에 기록한다. 그래서 아파치가 보통 사용하는 error.log 파일을 사용할 수 없는 경우에 대비한다. Windows 이벤트 로그는 Windows NT 4.0에서는 이벤트 뷰어 프로그램으로, 최신 Windows 버전에서는 이벤트 뷰어 MMC 스냅인에서 볼 수 있다.

2. Windows 에서의 구축법

▶ 2.2 Windows용 Apache 다운로드 (2.4 기준)

아파치 웹서버 <http://httpd.apache.org/download.cgi>에서 아파치 최신 버전에 대한 정보를 얻을 수 있다. 여기에는 최신 발표판과 알파 혹은 베타 테스트 버전과, 아파치 웹서버를 다운로드할 수 있는 HTTP 미러와 FTP 미러 목록이 있다. 빠르고 안정하게 다운받으려면 가까운 미러를 사용하면 된다.

Windows에 설치하려면 확장자가 .msi인 Windows용 아파치 버전을 다운받아야 한다. 이 파일은 다운받은 즉시 실행할 수 있는 아파치를 저장한 Microsoft 설치파일이다. 따로 소스코드만 묶어둔 .zip파일이 있다. Microsoft Visual C++ (Visual Studio)을 사용하여 직접 아파치를 컴파일할 수도 있다.

2. Windows 에서의 구축법

▶ 2.2 Windows용 Apache 설치 (2.4 기준)

Microsoft Installer 1.2 이상 버전이 필요하다. Windows 9x를 사용한다면 여기에서 Microsoft Installer를 2.0 버전으로 업그레이드할 수 있고, Windows NT 4.0과 2000을 사용한다면 여기에서 2.0 버전 업데이트를 구할 수 있다. Windows 2000 이하 제품군은 2.2 부터는 지원을 하지 않는다.

바이너리 설치파일로는 같은 컴퓨터에 서로 다른 아파치 2.0 버전을 설치할 수 없음을 주의하라. 그러나 1.3 버전과 2.0 버전은 같은 컴퓨터에 아무 문제없이 설치할 수 있다. 같은 컴퓨터에 두가지 다른 2.0 버전을 설치하려면 소스를 컴파일하여 아파치를 설치해야 한다.

.msi 파일은 <http://httpd.apache.org/download.cgi>에서 설치한다.

2. Windows 에서의 구축법

▶ 2.3 아파치 .msi 파일 설치

1. **네트워크 도메인 (Network Domain).** 등록된 서버의 DNS 도메인을 입력한다. 예를 들어, 서버의 전체 DNS 이름이 server.mydomain.net이라면 여기에 mydomain.net을 입력한다.
2. **서버명 (Server Name).** 서버의 전체 DNS 이름. 위의 경우 여기에 server.mydomain.net을 입력한다
3. **관리자 전자우편 주소 (Administrator's Email Address).** 여기에 서버 관리자나 웹마스터의 전자우편 주소를 입력한다. 기본적으로 클라이언트에게 보내는 오류문에 이 주소를 기록한다.

2. Windows 에서의 구축법

▶ 2.3 Windows용 Apache 설치 (2.4 기준)

4. **사용자 대상 (For whom to install Apache)** 새로 설치하는 아파치가 80번 포트에서 요청을 기다리게 하려면 (모든 사용자, 80번 포트, service로 – 추천)를 선택한다. 아파치 service로 실행한다. (즉, 아파치는 서버에 로그인한 사람이 없어도 실행 된다.) 개인적으로 테스트해보거나 이미 80번 포트를 사용하는 다른 웹서버가 있다면 (현재 사용자만, 8080번 포트, 직접시작)를 선택한다.

5. **설치 종류 (The installation type).** 모듈 개발에 필요한 소스코드와 라이브러리를 제외한 모든 것을 설치하려면 Typical을 선택한다. Custom을 선택하면 설치할 내용을 지정할 수 있다. 전체 설치시 디스크에 빈 공간이 약 13 메가바이트 정도 필요하다. 이 수치는 웹사이트 크기를 제외한 것이다.

6. **설치 장소 (Where to install).** 기본 경로는 C:\ProgramFiles\Apache Group이고, 이곳에 Apache2라는 디렉토리를 만든다.

2. Windows 에서의 구축법

▶ 2.3 Windows용 Apache 설치 (2.4 기준)

설치할 동안 아파치는 conf 하위디렉토리에 있는 파일들을 선택한 설치 디렉토리에 맞게 구성한다. 그러나 이 디렉토리에 설정파일이 이미 있다면 그대로 둔다. 대신, 해당 파일의 새로운 복사본에 확장자 .default를 붙인다.

2. Windows 에서의 구축법

▶ 2.4 아파치를 Service로 실행하기

1. 아파치 bin 하위디렉토리에서 명령행 프롬프트에 다음과 같이 입력하면 아파치를 Windows NT service로 설치한다:

```
apache -k install
```

설치할 service 이름을 지정하고 싶다면 다음 명령어를 사용한다. 컴퓨터에 아파치가 여러개 설치되었다면 이름을 다르게 주어야 한다.

```
apache -k install -n "MyServiceName"
```

service가 사용할 설정파일을 직접 지정하려면 다음과 같이 한다:

```
apache -k install -n "MyServiceName" -f "c:\files\my.conf"
```

2. 아파치 service를 제거하기 쉽다. 간단히:

```
apache -k uninstall
```

다음과 같이 제거할 아파치 service를 지정할 수 있다:

```
apache -k uninstall -n "MyServiceName"
```

2. Windows 에서의 구축법

▶ 2.4 아파치를 Service로 실행하기

3. 보통 아파치 service 시작, 재시작, 종료는 Apache Service Monitor나 NET START Apache2, NET, STOP, Apache2 같은 명령어 혹은 Windows 서비스 관리 창에서 한다. 어떤 방법을 사용하든지 아파치 service를 시작하기 전에 설정파일을 검사해봐야 한다:

```
apache -n "MyServiceName" -t
```

명령행 옵션으로도 아파치 service를 조정할 수 있다. 설치한 아파치 service를 시작하려면:

```
apache -k start
```

service 종료하려면:

```
apache -k stop or apache -k shutdown
```

3. Linux(CentOS7)에서의 구축법

▶ 3.1 설치

1. 아파치 웹서버 설치 확인
 - 1) 터미널 에서 다음 명령어를 사용해서 httpd가 설치 되어 있는지 확인 한다. (설치가 되어 있지 않으면 아무것도 나오지 않는다.)
 - # rpm -qa httpd
2. 아파치 웹서버 설치 하기
 - 1) 슈퍼유저 권한으로 위 명령어를 터미널에서 실행한다.
 - # yum install -y httpd
3. 설치가 되었는지 확인 하기
 - 1) 터미널에서 다음 명령어를 실행한다.
 - # rpm -qa httpd

3. Linux(CentOS7)에서의 구축법

▶ 3.2 설치 확인

1. Apache 포트 변경(변경을 원할 경우에 수행하며 필수 항목이 아님 기본포트는 80번 포트이다.)

- 1) 다음 명령을 터미널에서 실행

- # vi/etc/httpd/conf/httpd.conf

- 2) Listen 80(기본값)을 원하는 값으로 바꾼다 ex) 8080

```
#  
# Listen: Allows you to bind Apache to specific IP addresses and/or  
# ports, instead of the default. See also the <VirtualHost>  
# directive.  
#  
# Change this to Listen on specific IP addresses as shown below to  
# prevent Apache from glomming onto all bound IP addresses.  
#  
#Listen 12.34.56.78:80  
Listen 80
```

- 3) 다음 명령으로 apache를 재구동한다.

- # service httpd restart

3. Linux(CentOS7)에서의 구축법

▶ 3.2 설치 확인

1. 아파치 실행

- # service httpd start or \$ systemctl enable httpd.service

3. Linux(CentOS7)에서의 구축법

▶ 3.2 설치 확인

2. 정상적으로 실행되었는지 상태를 확인한다.

- # service httpd status or systemctl status httpd.service

```
[root@localhost network-scripts]# systemctl status httpd.service
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor preset: disabled)
   Active: active (running) since 화 2020-02-18 21:10:08 KST; 1 months 26 days ago
     Docs: man: httpd(8)
           man: apachectl(8)
  Main PID: 1511 (httpd)
    Status: "Total requests: 0; Current requests/sec: 0; Current traffic: 0 B/sec"
    CGroup: /system.slice/httpd.service
            └─ 1511 /usr/sbin/httpd -DFOREGROUND
               40173 /usr/sbin/httpd -DFOREGROUND
               40174 /usr/sbin/httpd -DFOREGROUND
               40175 /usr/sbin/httpd -DFOREGROUND
               40176 /usr/sbin/httpd -DFOREGROUND
               40177 /usr/sbin/httpd -DFOREGROUND
```

3. Linux(CentOS7)에서의 구축법

▶ 3.2 설치 확인

3. 웹 브라우저 주소창에 <http://localhost> 또는 <http://아이피>를 입력하여 다음과 같이 잘 구동 되는지 확인한다.



4. Mac에서의 구축법

▶ 4.1 설치

1. 아파치를 구동 하기 위해서 아파치를 설치 해야 하지만, 맥에는 기본적으로 아파치가 설치 되어 있기 때문에 따로 설치를 하지 않아도 됩니다.

4. Mac에서의 구축법

▶ 4.2 설치 확인

1. Apache 설치 버전 확인 하기
 - 1) 터미널에 다음 명령어를 실행합니다.
 - \$ httpd -v
2. 아파치 실행하기
 - 1) 터미널에 다음 명령어를 실행합니다.
 - \$ sudo apachectl start
3. Apache 실행 확인
 - 1) 웹 브라우저 주소창에 <http://localhost> 또는 <http://아이피를> 입력하여 다음과 같이 잘 구동 되는지 확인한다.

5. 구성 파일 설정

▶ 5.1 각 파일의 기능

※파일의 경로는 cent OS 기준으로 설명한다.

1. 기본 설정 파일(httpd.conf)

- 경로: /etc/httpd/conf/httpd.conf
- 기능: 수신 포트 설정 Apache 설치 경로등 대부분의 핵심 설정을 이 파일 안에서 설정한다.

2. 추가 어플리케이션 설정 디렉터리

- 경로: /etc/httpd/conf.d/ (php.conf 등과 같은 파일 저장)
- 기능: httpd.conf의 설정을 분리 해서 저장하고 httpd.conf에서 해당 기능을 불러와 사용할 수 있도록 한다.

5. 구성 파일 설정

▶ 5.1 각 파일의 기능

※파일의 경로는 cent OS 기준으로 설명한다.

3. 로그 파일 디렉터리

- 경로: /etc/httpd/logs -> /var/log/httpd (화살표는 소프트 링크를 의미함)
- 기능: access.log, error.log 등과 같은 시스템 운영 시 발생하는 로그들을 저장한다.

4. 모듈 파일 디렉터리(64비트 기준)

- 경로: /etc/httpd/modules -> /usr/lib64/httpd/modules
- 기능: 아파치에서 사용할 모듈 파일을 저장 한다. 추가 및 설정은 httpd.conf에서 한다.

5. 구성 파일 설정

▶ 5.1 각 파일의 기능

※파일의 경로는 cent OS 기준으로 설명한다.

5. 프로세스 PID 저장 디렉터리

- 경로: /etc/httpd/run -> /var/run/httpd
- 기능: 현재 실행중인 아파치의 PID가 저장 되어있다.

6. 기본 홈페이지 디렉터리

- 경로: /var/www/html
- 기능: 아파치의 페이지(html)를 저장할 하는 디렉터리 이다. 아파치 웹 서버에 접속하는 것은 이 디렉터리에 들어가는 것이라고 볼 수 있다

5. 구성 파일 설정

▶ 5.1 각 파일의 기능

※파일의 경로는 cent OS 기준으로 설명한다.

7. CGI 파일 저장 디렉터리

- 경로: /var/www/cgi-bin
- 기능: CGI 파일들이 실행되는 위치 이다.

8. 에러 파일 위치 디렉터리

- 경로: /var/www/error
- 기능: 에러코드별 에러파일 위치가 저장 되는 위치

5. 구성 파일 설정

▶ 5.1 각 파일의 기능

※파일의 경로는 cent OS 기준으로 설명한다.

9. 이미지아이콘 파일 디렉터리

- 경로: /var/www/icons
- 기능: 이미지 아이콘 파일들이 저장되는 위치이다.

5. 구성 파일 설정

▶ 5.2 기본 설정 파일 구성(Cent OS 기준)

- 기본 설정 파일(httpd.conf) 열기
 - \$ vi /etc/httpd/conf/httpd.conf 명령어 사용

```
#
# This is the main Apache HTTP server configuration file. It contains the
# configuration directives that give the server its instructions.
# See <URL:http://httpd.apache.org/docs/2.4/> for detailed information.
# In particular, see
# <URL:http://httpd.apache.org/docs/2.4/mod/directives.html>
# for a discussion of each configuration directive.
#
# Do NOT simply read the instructions in here without understanding
# what they do. They're here only as hints or reminders. If you are unsure
# consult the online docs. You have been warned.
#
# Configuration and logfile names: If the filenames you specify for many
# of the server's control files begin with "/" (or "drive:/" for Win32), the
# server will use that explicit path. If the filenames do *not* begin
# with "/", the value of ServerRoot is prepended -- so 'log/access_log'
# with ServerRoot set to '/www' will be interpreted by the
# server as '/www/log/access_log', whereas as '/log/access_log' will be
# interpreted as '/log/access_log'.
#
#
# ServerRoot: The top of the directory tree under which the server's
# configuration, error, and log files are kept.
#
# Do not add a slash at the end of the directory path. If you point
# ServerRoot at a non-local disk, be sure to specify a local disk on the
# Mutex directive, if file-based mutexes are used. If you wish to share the
# same ServerRoot for multiple httpd daemons, you will need to change at
# least PidFile.
#
ServerRoot "/etc/httpd"
```

5. 구성 파일 설정

▶ 5.2 기본 설정 파일 구성(Cent OS 기준)

◆ 지시자별 기능

- **ServerRoot "/etc/httpd"**
 - Apache의 홈 디렉터리를 절대 경로로 지정 한다. 이후에 지시자에서 사용하는 경로는 이 경로의 상대 경로로 나타낸다.
- **PidFile run/httpd.pid**
 - 서버 시동 및 작동시 자신의 프로세스 고유 아이디(PID)를 기록하는 파일 위치이다.

5. 구성 파일 설정

▶ 5.2 기본 설정 파일 구성(Cent OS 기준)

- **Timeout 300**
 - 클라이언트 요청에 의해 서버와 연결이 될 때, 서로간에 아무런 메시지가 발생하지 않을 때 연결을 끊는 시간.
 - 300초가 기본이다.
- **KeepAlive On/OFF**
 - Apache의 한 프로세스가 사용자의 지속적인 요청작업을 계속 처리할 것인지 대한 여부이다.
 - 켜져있는게 기본값이다.
- **KeepAlive Timeout 5**
 - KeepAlive가 On으로 설정 되어 있으면 유효한 값이며 설정 시간(초)동안 요청이 없으면 타임아웃 시킨다.

5. 구성 파일 설정

▶ 5.2 기본 설정 파일 구성(Cent OS 기준)

- **MaxKeepAliveRequests**
 - KeepAlive가 On인 경우 유효한 값으로, 하나의 프로세스가 방문자들의 요청을 처리할 횟수를 지정한다.
 - 0으로 설정하면 무제한 요청이 허용됨. 서버 성능을 최대화하려면 설정 값을 높은 값으로 유지하는 것이 좋음
- **Listen 80**
 - 시스템의 기본값 이외에 다른 IP 주소와 포트에 대해서도 연결할 수 있도록 해준다. 주로 가상호스트(Virtual Host)부분에서 기본 80번 이외에 다른 포트를 이용하여 가상호스트를 설정하고자 할 때에 필요하다.

5. 구성 파일 설정

▶ 5.3 CGI를 이용한 동적 페이지 설치 (Cent OS 기준)

- **CGI란**
 - 웹 서버와 클라이언트 간에 필요한 정보 교환을 가능하게 해주는 일종의 웹 인터페이스이다.
 - 동적 페이지 서비스를 제공하기 위해 만들어졌다.

5. 구성 파일 설정

▶ 5.3 CGI를 이용한 동적 페이지 설치 (Cent OS 기준)

- **CGI 설정 예시**

- httpd.conf 설정

- # vi /etc/httpd/conf/httpd.conf
 - 다음 구문을 ScriptAlias 아래에 추가한다.

```
246 #
247 ScriptAlias /cgi-bin/ "/var/www/cgi-bin/"
248 <Directory "/var/www/cgi-bin">
249     AllowOverride None
250     Options +ExecCGI
251     Require all granted
252 </Directory>
253 AddHandler cgi-script .cgi .pl .py
254 </IfModule>
255
```

- 추가한 뒤 # systemctl restart httpd

5. 구성 파일 설정

▶ 5.3 CGI를 이용한 동적 페이지 설치 (Cent OS 기준)

- python으로 cgi 생성
 - # vi /var/www/cgi-bin/first.py
 - 다음 예시를 작성한다.

```
#!/bin/env python

print "Content-type: text/html\n\n";
print "<html>\n<body>\n";
print "<div style =\nwidth: 100% font-size: 80px; font-weight: bold; text-align: center; \n>\n";
print "Python Test Page in www.chul.com";
print "\n</div>\n";
print "</body>\n</html>\n";
```

- 생성 후 다음 명령어로 권한을 부여한다.
- # chmod 705 /var/www/cgi-bin/first.py

5. 구성 파일 설정

▶ 5.3 CGI를 이용한 동적 페이지 설치 (Cent OS 기준)

- ifconfig로 로컬 주소 확인

```
[root@localhost cgi-bin]# ifconfig
ens33: flags=4163<UP, BROADCAST, RUNNING, MULTICAST> mtu 1500
    inet 192.168.42.136 netmask 255.255.255.0 broadcast 192.168.42.255
    inet6 fe80::c7fb:c29a:6517:a911 prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:88:54:e3 txqueuelen 1000 (Ethernet)
    RX packets 2003 bytes 599595 (585.5 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 1245 bytes 111606 (108.9 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

5. 구성 파일 설정

▶ 5.3 CGI를 이용한 동적 페이지 설치 (Cent OS 기준)

- ifconfig에서 확인한 로컬 주소를 firefox로 이동
- `www.로컬주소/cgi-bin/first.py`



6. 보안강화

▶ 6.1 보안취약점

Apache 웹 서버에 대한 보안 문제에 대해 언급한다.
크게 3가지로 분류된다.

1. 웹 서버/클라이언트/애플리케이션 자체의 버그
2. 웹 서버/클라이언트/애플리케이션 설정의 오류
3. 침입차단시스템의 웹 서비스 오픈

SANS TOP20 Vulnerabilities 에서
윈도우는 1위 웹서버&서비스 , 6위 웹 브라우저를 가지며
유닉스는 2위 웹서버를 가지고 있다.

보안취약점 CVE에서는 다음과 같이 보안취약점이 발생했다.

6. 보안강화

▶ 6.1 보안취약점

Vulnerability Trends Over Time

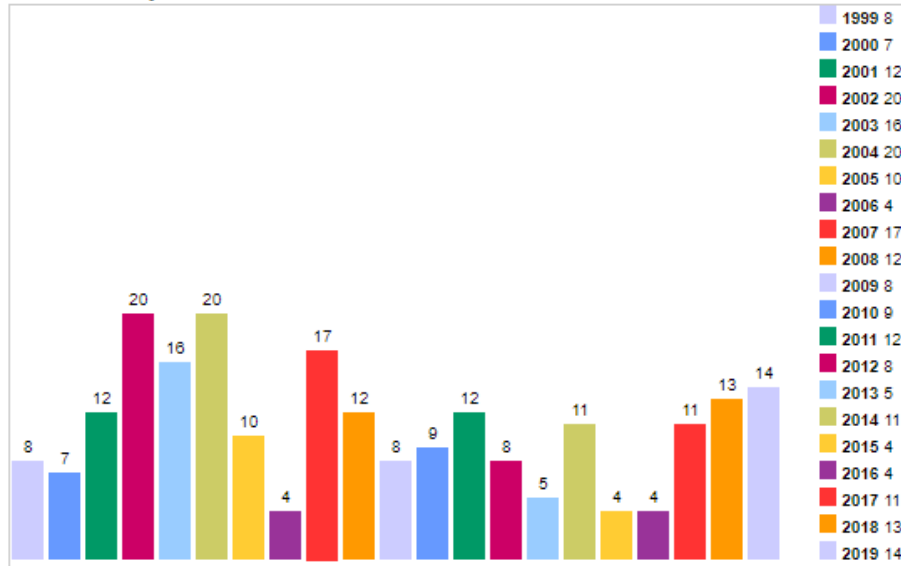
Year	# of Vulnerabilities	DoS	Code Execution	Overflow	Memory Corruption	Sql Injection	XSS	Directory Traversal	Http Response Splitting	Bypass something	Gain Information	Gain Privileges	CSRF	File Inclusion	# of exploits
1999	8	3	2	1											
2000	7		1				1								
2001	12	1								5	1				
2002	20	6	5	3			2	1			2				
2003	16	9	3	1							1				
2004	20	8	2	4				1		3	1	1			
2005	10	5	2	3			3			2					
2006	4	1	2				1			1					
2007	17	5	3				4	2		1	2	1			
2008	12	2			1		6		1			1	1		
2009	8	5								1		1			
2010	9	3	2	1			1				3				1
2011	12	8		1								1			2
2012	8	4		1			1				2	1			
2013	5	1	1				2								
2014	11	9	1	2						2	1				1
2015	4	2								1					
2016	4	2								1					
2017	11	1		1					1	1	1				
2018	13	3		1					1						
2019	14	1	1	2			1			2					
Total	225	79	25	21	1		22	4	3	20	14	6	1		4
% Of All		35.1	11.1	9.3	0.4	0.0	9.8	1.8	1.3	8.9	6.2	2.7	0.4	0.0	

출처: CVEDETAILS (apache http)

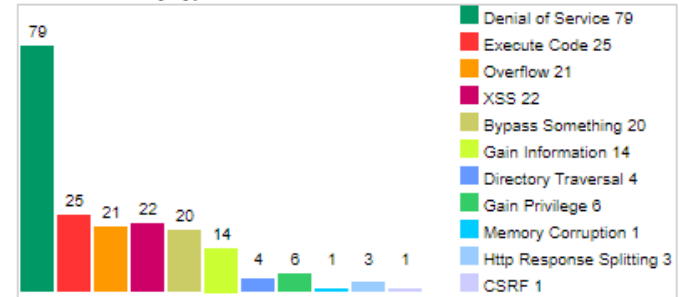
6. 보안강화

▶ 6.1 보안취약점

Vulnerabilities By Year



Vulnerabilities By Type



출처: CVEDETAILS (apache http)

6. 보안강화

▶ 6.1 보안취약점

상세 페이지에서 각 cve 에 대한 정보를 볼 수 있으며 이에대한 심각도가 설정되어 있다. 버전업시 해당 스코어가 줄어들거나 해당 취약점에 대한 보안을 완료하고 있다.

[Copy Results](#) [Download Results](#)

#	CVE ID	CWE ID	# of Exploits	Vulnerability Type(s)	Publish Date	Update Date	Score	Gained Access Level	Access	Complexity	Authentication	Conf.	Integ.	Avail.
1	CVE-2019-0217	362		Bypass	2019-04-08	2019-05-13	6.0	None	Remote	Medium	Single system	Partial	Partial	Partial
In Apache HTTP Server 2.4 release 2.4.38 and prior, a race condition in mod_auth_digest when running in a threaded server could allow a user with valid credentials to authenticate using another username, bypassing configured access control restrictions.														
2	CVE-2019-0215	284		Bypass	2019-04-08	2019-05-13	6.0	None	Remote	Medium	Single system	Partial	Partial	Partial
In Apache HTTP Server 2.4 releases 2.4.37 and 2.4.38, a bug in mod_ssl when using per-location client certificate verification with TLSv1.3 allowed a client to bypass configured access control restrictions.														
Total number of vulnerabilities : 2 Page : 1 (This Page)														

출처: CVEDETAILS (apache http)

6. 보안강화

▶ 6.2 버전유지&권한설정

예 : 취약점 ID : CVE-2018-8011 에서는 2018년 cvss 7.5 -> 2019년 cvss 5.0
으로 버전업데이트 하면서 하향되었다.

mod_md 챌린지 처리기에서 null 포인터 참조를 작동시켜 대상 자식
프로세스가 중단 할 수 있던 문제점이 발견되었던 부분을 수정하였으며
V2.4.34에서부터 해결하였다.

Apache 웹 서버 구축시 버전유지를 통해 보안에대한 1차 관리가 가능하다.

따라서 <http://httpd.apache.org/lists.html#http-announce> 와 같은
최신 아파치에 대한 정보를 구독하여 버전에 대한 최신화가 필요하다.

6. 보안강화

▶ 6.2 버전유지&권한설정

Root 사용자가 아파치를 시작한 후, 요청에대한 서비스를 위해 User 지시어로 지정한 사용자로 변환한다. Root 가 실행하는 명령어가 있을시 root 이외의 사용자가 수정하는 경우가 없도록 권한설정을 해야한다.
(디렉토리 및 모든 상위디렉토리에 대해)

예) ServerRoot로 /usr/local/apache를 사용한다면 root 사용자가 다음과 같이 디렉토리를 만들어야 한다.

```
mkdir /usr/local/apache
cd /usr/local/apache
mkdir bin conf logs
chown 0 . bin conf logs
chgrp 0 . bin conf logs
chmod 755 . bin conf logs
```

6. 보안강화

▶ 6.3 SSI & CGI ScriptAlias

SSI는 서버의 부하를 늘리고 일반적인 CGI 스크립트와 동일한 위험을 가짐.

이에 다음과 같은 대응책이 있다.

1. SSI 파일이 가져올 수 있는 피해를 격리하기위해 서버관리자는 일반적인 CGI 절에서 설명하는 방법으로 suexec를 사용한다.
2. SSI 파일은 일반적으로 많이 사용하는 .shtml 같은 별도의 확장자를 가져야 한다
3. SSI 페이지가 스크립트나 프로그램을 실행하지 못하도록 만든다.

6. 보안강화

▶ 6.3 SSI & CGI ScriptAlias

일반적인 CGI 에서는 항상 CGI 스크립트/프로그램의 저자를 신뢰 할 수 있어야 하며 고의건 실수이건 CGI의 잠재적 보안상 허점을 발견할 수 있어야 한다.

- 기본적으로 CGI 스크립트는 웹서버 사용자 권한으로 시스템에서 어떠한 명령도 실행할 수 있기때문

또한, 충돌 가능성이 있다(고의 또는 실수) 이에 대해서는 관리 차원에서의 해결이 필요하며 이는 기타 CGI 보안 취약점에서 더 상세히 다룬다.

예) .htaccess 에서 보안기능변경 가능성을 방지하기 위한 처리

```
<Directory />  
AllowOverride None  
</Directory>
```

6. 보안강화

▶ 6.3 SSI & CGI ScriptAlias

ScriptAlias하지 않은 CGI 에서는 다음 조건을 만족할때만 CGI 스크립트를 실행하도록 허용할 수 있다.

1. 고의건 실수이건 사용자가 시스템을 공격에 노출시키는 스크립트를 작성하지 않는다고 믿는다.
2. 시스템의 다른 부분의 보안이 약해서, 잠재적인 허점을 하나 더 만들어도 나빠질 것이 없다고 생각하는 경우.
3. 사용자가 없고, 아마 아무도 서버를 방문하지않는 경우

6. 보안강화

▶ 6.3 SSI & CGI ScriptAlias

특정 디렉토리에서만 CGI를 실행할 수 있도록 제한하면 관리자는 이들 디렉토리를 통제할 수 있다.

이 경우는 scriptalias하지 않은 CGI보다 확실히 안전하다.

단, 신뢰하는 사용자만 디렉토리에 접근할 수 있고, 관리자가 새로운 CGI 스크립트/프로그램의 잠재적인 보안상 허점을 검사할 용이가 있다면. 대부분의 사이트는 scriptalias하지 않은 CGI 방식 대신 이 방식을 사용한다.

6. 보안강화

▶ 6.4 mod_security2

Apache 웹 서버의 보안 관리에서 중요하게 다루는 것은 디렉토리 관리, 계정관리, 설정파일 관리, 인증관리, 보안패치, SSL인증서 또는 웹 암호화 솔루션 적용 관리, 데이터 백업, 로그설정 및 분석 등이 있다.

위에서 언급한 내용들은 취약점에 대한 인지와 대표적인 처리방법에 대해서 인지 할 수 있도록 돕는 내용이었으며 전문가의 위의 모든 경우에 대해 보안취약성을 검출하고 대처 할 수 있도록 해야한다.

이중 로그 관리 및 방화벽 설정을 할 수 있는 mod_security2 에 대해 설명한다.

6. 보안강화

▶ 6.4 mod_security2

일반적으로 특정한 요청에 대한 로그진행에 대한 헬프와 방화벽 설정을 위해 사용된다.

다음과 같은 대표적인 공격 패턴에 대해 대응할 수 있다.

XSS, PHP Injection, SQL Injection, Command Execute

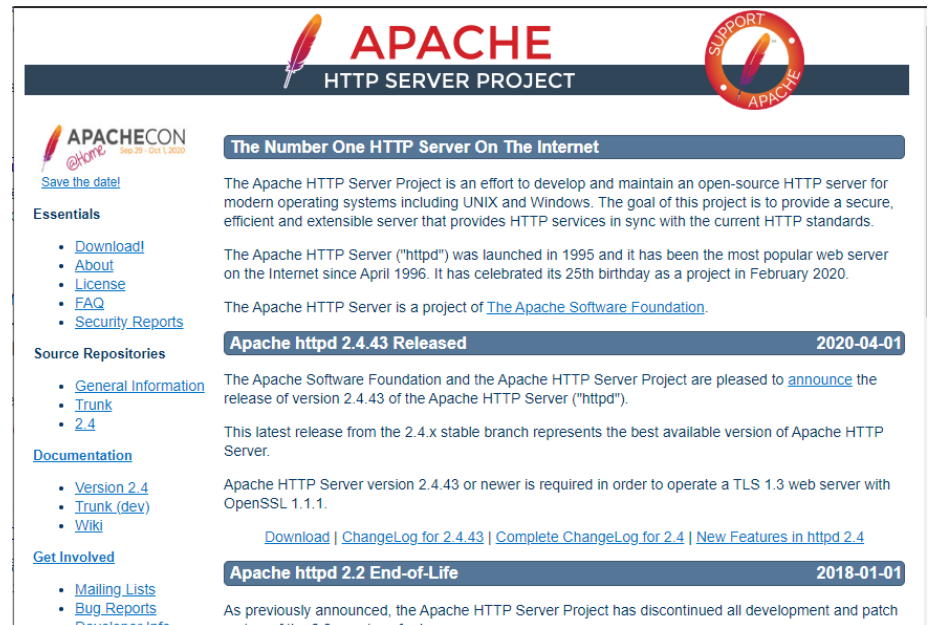
이에 대한 설치법은 기타. Mod_security2 설치 및 적용에 대해서 다룬다.

또한, 기타 구성파일 설정 및 사용자 설정과 인증에 대해서도 기타. 보안설정 파트에서 다룬다.

7. 커뮤니티

▶ 7.1 http server project

http server는 기술을 개발하는 사람들의 커뮤니티가 있고 여기에서 개발자 리스트나 사용자에게 대한 기술 지원을 받을 수 있다. 사이트 주소는 <http://httpd.apache.org/> 이다.



APACHE
HTTP SERVER PROJECT

APACHECON
@home
[Save the date!](#)

Essentials

- [Download!](#)
- [About](#)
- [License](#)
- [FAQ](#)
- [Security Reports](#)

Source Repositories

- [General Information](#)
- [Trunk](#)
- [2.4](#)

Documentation

- [Version 2.4](#)
- [Trunk \(dev\)](#)
- [Wiki](#)

Get Involved

- [Mailing Lists](#)
- [Bug Reports](#)
- [Developer Info](#)

The Number One HTTP Server On The Internet

The Apache HTTP Server Project is an effort to develop and maintain an open-source HTTP server for modern operating systems including UNIX and Windows. The goal of this project is to provide a secure, efficient and extensible server that provides HTTP services in sync with the current HTTP standards.

The Apache HTTP Server ("httpd") was launched in 1995 and it has been the most popular web server on the Internet since April 1996. It has celebrated its 25th birthday as a project in February 2020.

The Apache HTTP Server is a project of [The Apache Software Foundation](#).

Apache httpd 2.4.43 Released 2020-04-01

The Apache Software Foundation and the Apache HTTP Server Project are pleased to [announce](#) the release of version 2.4.43 of the Apache HTTP Server ("httpd").

This latest release from the 2.4.x stable branch represents the best available version of Apache HTTP Server.

Apache HTTP Server version 2.4.43 or newer is required in order to operate a TLS 1.3 web server with OpenSSL 1.1.1.

[Download](#) | [ChangeLog for 2.4.43](#) | [Complete ChangeLog for 2.4](#) | [New Features in httpd 2.4](#)

Apache httpd 2.2 End-of-Life 2018-01-01

As previously announced, the Apache HTTP Server Project has discontinued all development and patch review of the 2.2.x series of releases.

7. 커뮤니티

▶ 7.2 커뮤니티 메인 페이지

메인 페이지에는 현재 httpd의 릴리즈 정보와 진행중인 프로젝트의 정보에 대해 확인할 수 있습니다. 진행중인 프로젝트에 참여해 컨트리뷰터가 되는것도 한 가능하다.

Apache httpd 2.4.43 Released

2020-04-01

The Apache Software Foundation and the Apache HTTP Server Project are pleased to [announce](#) the release of version 2.4.43 of the Apache HTTP Server ("httpd").

This latest release from the 2.4.x stable branch represents the best available version of Apache HTTP Server.

Apache HTTP Server version 2.4.43 or newer is required in order to operate a TLS 1.3 web server with OpenSSL 1.1.1.

Want to contribute to the Apache HTTP Server?

Awesome! Have a look at our current 'Help Wanted' listings then:

Tasks the Apache HTTP Server Project would like help with:

Title	Languages	Difficulty	Created
Rewrite mod_ldap cache handling code	c	Intermediate	2018. 5. 29. 오전 2:23:20
Learn and work on httpd's internals	c	Advanced	2018. 1. 3. 오전 2:19:40
Websocket documentation	english,xml,lua,c	Intermediate	2016. 2. 20. 오전 2:31:35
Improve the Request Processing guide	xml,html,c	Advanced	2016. 2. 14. 오후 8:16:11
Traffic shaping with mod_lua	lua,c	Advanced	2016. 2. 14. 오후 6:09:50
Make a better guide on request hooks	xml,html,c	Advanced	2016. 2. 8. 오후 6:46:07
Write a better cache guide	xml,html	Expert	2016. 2. 7. 오전 6:30:59

Powered by [Help Wanted](#) - a task directory for Apache projects

7. 커뮤니티

▶ 7.3 http server 문서

http server 관련 사용자 지침서 튜토리얼 등을 버전별로 확인 할 수 있다.

Apache HTTP Server Version 2.4 문서

가능한 언어: [da](#) | [de](#) | [en](#) | [es](#) | [fr](#) | [ja](#) | [ko](#) | [pt-br](#) | [tr](#) | [zh-cn](#)

이 문서는 최신판 번역이 아닙니다. 최근에 변경된 내용은 영어 문서를 참고하세요.

Google 검색

발표문

[버전 2.0의 새로운 기능](#)
[1.3에서 2.0 버전으로 업그레이드](#)
[아파치 라이선스](#)

참조 설명서

[컴파일과 설치](#)
[시작](#)
[종단과 재시작](#)
[설정 지시어](#)
[지시어 빠른참조](#)
[모듈](#)
[다중처리모듈 \(MPM\)](#)
[필터](#)
[핸들러](#)
[서버와 지원 프로그램](#)
[용어](#)

사용자 지침서

[주소와 포트 지정](#)
[설정파일](#)
[색션 설정](#)
[내용협상 \(content negotiation\)](#)
[동적공유객체 \(DSO\)](#)
[환경변수](#)
[로그파일](#)
[URL을 파일시스템에 대응](#)
[성능향상](#)
[보안 팁](#)
[서버 지역 설정](#)
[SSL/TLS 암호화](#)
[CGI의 SuEXEC 실행](#)
[URL 재작성 \(rewriting\) 지침서](#)
[가상호스트](#)

How-To / 튜토리얼

[인증 권한부여 접근제어](#)
[CGI: 동적 페이지 생성](#)
[.htaccess 파일](#)
[Server Side Includes \(SSI\)](#)
[사용자별 워드락토리 \(public_html\)](#)

플랫폼별 설명

[Microsoft Windows](#)
[Novell NetWare](#)
[EBCDIC 포팅](#)

다른 주제

[자주 물어보는 질문 \(FAQ\)](#)
[사이트맵](#)
[개발자를 위한 문서](#)
[기타](#)

가능한 언어: [da](#) | [de](#) | [en](#) | [es](#) | [fr](#) | [ja](#) | [ko](#) | [pt-br](#) | [tr](#) | [zh-cn](#)

[모듈](#) | [지시어](#) | [FAQ](#) | [용어](#) | [사이트맵](#)

Copyright 2020 The Apache Software Foundation.

7. 커뮤니티

▶ 7.4 버그 리포트

http server에 버그가 발생할 경우 대처법에 대해 적혀 있는 페이지 이다.

해당 매뉴얼 대로 진행 후 해결 되지 않는다면 아파치의 버그 데이터 베이스에 보고서 형식으로 제출한다.

Bug Reporting

Reports of security issues should not be made here. Please see the [security report page](#) if you have concerns or think you have discovered a security hole in the Apache Web server software. If you have identified a bug in the Apache HTTP Server, please fill out a problem report form and submit it. Before you do that, though, **make sure you have done all of the following.**

1. Verified that the bug exists in [the most recent version](#) of Apache.
2. Followed the instructions in the `INSTALL` file correctly
3. Tried compiling with only the distributed set of modules and with no other patches (so that we can make sure it's an Apache bug and not a bug in a module or patch provided by someone else). It is frustrating to take time and effort to track down a problem only to figure out that it is caused by a broken third party module.
4. Looked at the error log. Please include any related messages in your report, or state that there are none.
5. Checked the [FAQ](#).
6. Searched the [bug report](#) database. When you are doing this, you should be sure to search closed problem reports in addition to open ones. The search form on the main bugzilla page by default applies only to open reports; to search all bug reports, begin the search with the keyword "ALL". For example, the following search will find all reported proxy bugs in Apache 2.0: "ALL httpd-2.0 mod_proxy". Some helpful queries:
7. [All bugs fixed in the last 60 days](#)
8. [Frequently reported bugs](#) (bugs with duplicates changed in the last 30 days)
9. [All bugs changed in the last 10 days](#)

7. 커뮤니티

▶ 7.5 기타

- 컨트리뷰터 : 이 프로젝트에 참여한 컨트리뷰터 들을 나열 했다.
- 스폰서 : 이 프로젝트에 기부한 스폰서 들을 나열 했다.

Current Apache HTTP Server Project Members

This page exists to recognize the efforts and contributions of the core individuals in the Apache HTTP Server Project.

Contributor	Activities
Erik Abele	Mostly documentation, various tweaks here and there
Aaron Bannert	Some MPM work, recent APR contributions, "that API guy".
Brian Behlendorf	Various, focusing on infrastructure for development.
Rich Bowen	Documentation, Concom.
Ken Coar	HTML pedant, FAQ editor, UI perfectionist, bugdb script hacker
Eric Covener	Bug fixes, LDAP, Documentation, IRC and ML support
Davi Arnaut	Bug fixes and others things
Mark Cox	Cookie module, Status module, DBM code.
Chris Darroch	
Lars Eilebrecht	Bug hunter
Stefan Eissing	mod_h2
Ralf S. Engelschall	Dr. Cosmetics, mod_rewrite, DSO support, Configure, APACI, apxs, etc.
Justin Erenkrantz	
Roy T. Fielding	Standards Cop.
Tony Finch	Mass virtual hosting
Stefan Fritsch	Bug fixes
Daniel Gruno	Documentation, bug fixes, syntax highlighting
Rob Hartill	general trouble maker, comments to offend all.

PLATINUM SPONSORS:



8. 활용사례

▶ 8.1 국내활용사례

1. 카카오 - S2Graph

- 공식적으로 아파치 HTTP프로젝트에 채택된 사례이다.
아파치 HTTP는 이전 특징에서도 설명하였듯 활용가치가 높아 현재 가장 많이 사용되는 웹 호스팅 OSS이며, ASF에서는 몇 백개의 아파치 프로젝트를 지원하고 있다.
- 카카오의 S2Graph는 아파치 HBase 기반의 온라인 트랜잭션 프로세싱 (OLTP) 그래프 데이터베이스로, 다양한 카카오 서비스들에서 생성되는 방대한 이용자 데이터를 빠르고 효율적으로 관리하기 위해 개발됐다.
- 그래프 데이터베이스란 데이터들의 상관 관계를 그래프 구조로 표현해 저장하는 것이다. 해외에서는 트위터, 페이스북, 인스타그램 등 소셜네트워크 분야에 선도적인 기업들이 자사 서비스 이용자 대상 피드(feed) 추천 등을 위해 자체 개발해 활용하고 있다.
- <https://tech.kakao.com/2016/01/29/opensource-1-s2graph/>

8. 활용사례

▶ 8.1 국내 활용사례

2. 비트나인 – AGE

- 다음은 그래프 데이터베이스 전문기업 비트나인의 AGE를 소개한다. 이 또한, ASF에서 공식적으로 지원받는 프로젝트 중 하나이다.
- 관계형 데이터베이스인 PostgreSQL의 확장 형식을 채택해 성능 저하 없이 기존 솔루션을 사용할 수 있게 하는 기능을 가지고 있으며, 언어는 표준 SQL 및 그래프 쿼리 언어인 오픈 싸이퍼(OpenCypher)를 사용한다. 싸이퍼(Cypher) 쿼리로 구문 분석 후, PG 백엔드와 밀접하게 연결된 쿼리 트리에 맞춰 변환할 수 있도록 정교하게 설계됐다. 그래프 쿼리를 관계형 쿼리로도 작동할 수 있도록 변환하기 때문에 SQL 및 싸이퍼(Cypher) 쿼리를 용이하게 사용할 수 있다.
- 사용자가 데이터를 보다 효율적으로 모델링해 관리하는 것을 목적으로 하며, PostGIS, Prometheus, Timescale 등 기존 확장 프로그램과 통합가능하고 여러 개의 그래프에 접속하고 통합해 단일 쿼리를 실행할 수 있는 '멀티 그래프 쿼리' 기능을 제공한다.

8. 활용사례

▶ 8.2 해외활용사례

1. APACHE HTTP SERVER PROJECT

- 아파치 HTTP 서버 프로젝트 1995년 부터 이어져 온 ASF 지원 프로젝트다.
- 현 튜토리얼에서 사용된 OSS도 해당 프로젝트에서 파생되었다.
이를 사례로 언급한 이유는 개념의 확보를 위해서이며, ASF 에서 지원 되는 것은 아파치 HTTP ,하둡을 비롯한 수많은 프로젝트라 볼 수 있다.
아파치 HTTP OSS를 사용하는 곳은 많지만 구체적으로 사용한 방법에 대해서는 언급하지 않는다.
- COOKBOOK 및 여러 매뉴얼에서 아파치 HTTP를 소개받을 수 있고, 해당 프로젝트의 '회원'으로 참여하기 위해서는 해당 프로젝트 기여 일 수가 6개월이 되어야 한다 (소스 및 버그리포팅 등)

8. 활용사례

▶ 8.2 해외활용사례

2. SiteGround

- 사이트 그라운드스는 2004년에 설립된 웹 호스팅 회사이며 200만개 이상의 도메인을 호스팅한다.
- 공유 호스팅, 클라우드 호스팅 및 엔터프라이즈솔루션 뿐만 아니라 이메일 호스팅 및 도메인 등록을 제공하고 있다.
사이트 그라운드스는 아파치 뿐 아니라 여러 웹 서버 호스팅 소프트웨어를 지원하고 있고 이 중 아파치의 경우 사용자가 웹 호스팅을 직접 설계하는 것이 아닌 이에 대한 기술지원을 받으려 할 때 컨택되는 기업이다.
- 아파치 HTTP를 비롯한 여러 기술지원을 해주는 것을 사업 아이템으로 가지며, 아차피 HTTP 호스팅을 사업아이템으로 갖는 기업중 대표적인 곳으로 지목된다.

9. 국내,해외 기술지원

▶ 9.1 기술 지원이란

- 공개SW 기술지원 : 공개SW를 최적의 상태로 활용 · 유지하기 위한 기술지원 서비스 이다.
- 제공하는 서비스 :
 - 1. 유지관리(설치, 장애복구, 예방점검 등)
 - 2. 제품 지원(기능향상, 제품 수정 및 각종 업데이트)
 - 3. 컨설팅 등이 있다.

9. 국내,해외 기술지원

▶ 9.2.1 국내 rockPLACE(락플레이스)

- 제공 되는 서비스

- 웹 어플리케이션 배포 이슈 지원
- 경량화(Slimming) 및 성능 이슈 지원
- 운영 전환 및 오픈 현장 대기
- 상시 장애지원과 점검지원
- Nagios XI를 통한 모니터링 서비스 제공
- 월간 기술지원 보고서 제공
- Red Hat GSS(Global Support Service)를 통한 Unknown 이슈 대행 지원



9. 국내,해외 기술지원

▶ 9.2.1 국내 rockPLACE(락플레이스)



- **고객의 이점**

- 어플리케이션의 성능 향상
- 효율적인 관리 프로세스 도입
- 저비용 운영 시스템 구축
- 장애 예방 및 장애 기간의 최소화(80여명의 엔지니어보유)
- 교육 및 지원을 통한 고객 사 자체 기술 축적
- 한국어 지원

- **운영 및 기술지원 단계**

- 1. 고객과 인터뷰를 통해 현안 문제점 파악하여 적절한 클라우드 운영 방안 혹은 시스템 검토
- 2. 고객 요구사항에 맞추어 안전하고 원활한 운영을 위한 체크리스트와 함께 모든 세부사항 변경 및 설정 실행
- 3. 구성 완료 및 완료 보고

9. 국내,해외 기술지원

▶ 9.2.2 국내 리원에이스

- 제공 되는 서비스

- 운영 사업자와 기술지원 연계체제 구축
- 운영지원에 필요한 paper work 공유
- 정기점검 및 장애분석 지원
- 실시간 자원상태 모니터링 지원
- 운영 담당자와 업무 협업이 필요할 경우 담장자 지정



9. 국내,해외 기술지원

▶ 9.2.2 국내 리원에이스

- **고객의 이점**

- 안정적인 운영 환경 조성
- 강력한 보안정책 수립
- S/W 버전 이력관리 가능(패치, 업데이트 기준)
- 이력내역 산출물 제공
- 효율적인 이력관리
- 장애 처리 히스토리 관리 가능



- **운영 및 기술지원 단계**

- 1. 해당 사업 계획/분석/설계를 바탕으로 최적의 담당자 투입
- 2. 기술지원 담당자와 실 운영자와의 협업을 통한 기술지원 체계확립
- 3. 사업초기 기술지원 담당자 배정으로 안정화 지원

9. 국내,해외 기술지원

▶ 9.2.3 국내 KTds



- 제공 되는 서비스
 - 오픈소스 전환 서비스
 - K-Watch SMS(Server Monitoring Solution)
 - 데이터 시각화
 - 서버의 성능 정보를 분석하고 수집해,해당 서버 장애 예측 및 예방
 - E-mail,SMS등 장애 발생시 알림 기능
 - K-Watch APM(Application Performance Management)
 - APM 대시보드에서 각종 성능과 서비스 현황에 대한 신속한 파악 기능 제공
 - 실시간 서비스 분석
 - 서비스 통계
 - K-COMPASS
 - 오픈소스 라이선스 관리
 - 보안 취약점 관리

9. 국내,해외 기술지원

▶ 9.2.3 국내 KTds

- **고객의 이점**
 - 제품이 세분화 되어 있어 상황에 맞게 선택가능
 - 서비스 및 데이터 시각화 가능
 - 오픈소스 커뮤니티 리더 보유
 - 다양한 방법으로 장애 발생 정보 알림



9. 국내,해외 기술지원

▶ 9.3.1 해외 managecat



- **제공되는 서비스**
 - 아파치 톰캣 지원
 - 서버 구성관리
 - 버그 추적 및 수정
 - 24시간 서비스
 - 제품 업데이트
 - 보안 패치
 - 긴급 핫픽스
- **고객의 이점**
 - 연중 무휴 서비스
 - 심각도에 따른 응답시간을 정해둠
 - 웹기반 지원
 - 메신저 알림 기능

9. 국내,해외 기술지원

▶ 9.4 기타 국내 http server 기술 지원 기업

그루기술 (070-4325-7011, www.grutech.co.kr)
락플레이스 (02-6251-7788, <http://www.rockplace.co.kr>)
리눅스데이타시스템 (02-6207-1160, <http://www.linuxdata.co.kr>)
범일정보 (053-422-4005, www.bumil.co.kr)
www.bonc.co.kr)
세림티에스지 (042-488-7900, www.selim.co.kr)
www.cywellssystem.com)
에스유리눅스 (031-717-1312, <https://www.sulinux.net/web/>)
오뉴이노베이션 (02-866-2179, www.ohnew.co.kr)
www.etechsystem.co.kr)
케이엘정보통신 (042-824-1077, www.klic.co.kr)

대신정보통신 (02-2107-5000, www.dsic.co.kr)
래빗소프트 (02-2088-7270, www.rabbitsoft.co.kr)
리원에이스 (070-7209-0444, liwonace.co.kr)
비온시이노베이터 (062-371-9339,
싸이웰시스템 (02-783-6972,
엠티데이타 (031-628-8180, www.mtdata.co.kr)
이테크시스템 (02-3449-6705,
케이티디에스 (070-4168-2900, www.ktds.com)

10. 기타

▶ 본문에 사용된 사이트 및 본문내용을 제외한 것을 기재

<https://www.apache.org/> - 아파치 재단 (아파치 커뮤니티, 프로젝트 등 관련 정보를 모두 제공한다

https://httpd.apache.org/docs/2.2/ko/misc/security_tips.html - 아파치 보안 정보

[https://www.oss.kr/storage/app/public/oss/7e/71/\[Apache\]%20Solution%20Guide%20V0.95.pdf](https://www.oss.kr/storage/app/public/oss/7e/71/[Apache]%20Solution%20Guide%20V0.95.pdf) - 아파치 가이드북

<https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2020-9490> - 아파치 포안 취약점 CVE 2020 9490

http://kiise.or.kr/e_journal/2018/4/KTCP/pdf/05.pdf - 아파치 로그 분석 + 보안 공격 분석

https://www.oss.kr/info_sp/show/ea45ab2a-f8e9-43dc-b967-a92de76c2b1a - OSS 사이트 아파치 정보

<https://cwiki.apache.org/confluence/display/HTTPD> - 아파치 HTTPD 정보

+

<https://www.openlogic.com/solutions/support> - 오픈로직 아파치 지원기업

<http://community.apache.org> - Apache community

<http://bytes.com/topic/apache> - 아파치 웹서버 포럼



충북대학교
SW중심대학사업단