2018.12

**How to use the Nginx**

Revision History

|  |  |
| --- | --- |
| 문 서 번 호 |  |
| 문 서 명 |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 개정번호 | 개정일자 | 개정 내용 요약 | 작성자 | 확인자 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Table of Contents

[1. Nginx의 개요 1](#_Toc532992328)

[1.1. Nginx(웹서버)의 역할 1](#_Toc532992329)

[1.1.1. 정적 파일을 처리하는 HTTP 서버로서의 역할 1](#_Toc532992330)

[1.1.2. 응용프로그램 서버에 요청을 보내는 리버스 프록시로서의 역할 2](#_Toc532992331)

[1.2. AWS에 nginx 설치하기 3](#_Toc532992332)

[1.2.1. 설치 및 제거 3](#_Toc532992333)

[1.2.2. Nginx 경로 4](#_Toc532992334)

[1.3. Nginx 디렉토리 구조 살펴보기 4](#_Toc532992335)

[1.4. nginx.conf 기본 환경설정 튜닝하기 5](#_Toc532992336)

[1.4.1. 최상단 (Core 모듈) 6](#_Toc532992337)

[1.4.2. events 블락 7](#_Toc532992338)

[1.4.3. http 블락 7](#_Toc532992339)

[1.4.4. 기타 8](#_Toc532992340)

[1.4.5. 설정 파일의 반영 8](#_Toc532992341)

[1.4.6. 참고 사이트 8](#_Toc532992342)

List of Tables

**그림 목차 항목을 찾을 수 없습니다.**

List of Figures

**그림 목차 항목을 찾을 수 없습니다.**

# Nginx의 개요

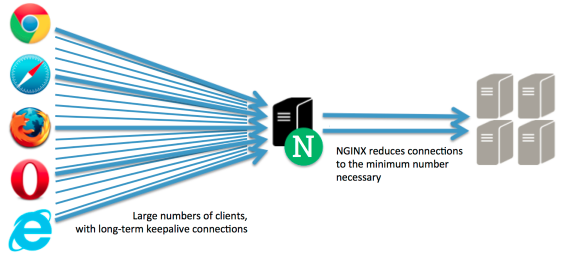
엔진엑스(Nginx)는 lgor Sysoev라는 러시아 개발자가 동시접속 처리에 특화된 웹서버 프로그램이다. Apache보다 동작이 단순하고, 전달자 역할만 하기 때문에 동시접속 처리에 특화되어 있다.

동시접속자(약 700명) 이상이라면 서버를 증설하거나 Nginx 환경을 권장한다고 한다. 지금은 아파치가 시장 점유율이 압도적이지만, 아마존웹서비스(AWS)상에서는 시장 점유율 44%에 달할정도로 가볍고, 성능이 좋은 엔진이라고 한다.

본 문서에는 AWS 인스턴스 상에서 Nginx를 설치하고, 기본적인 설정파일들을 알아보는 시간을 가질 것이다. .

참조 URL : http://whatisthenext.tistory.com/123

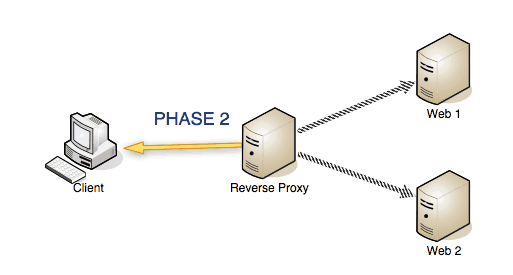
## Nginx(웹서버)의 역할



### 정적 파일을 처리하는 HTTP 서버로서의 역할

웹서버의 역할은 HTML, CSS, Javascript, 이미지와 같은 정보를 웹 브라우저(Chrome, lexplore, Opera, Firefox 등)에 전송하는 역할을 한다. (HTTP 프로토콜 준수)

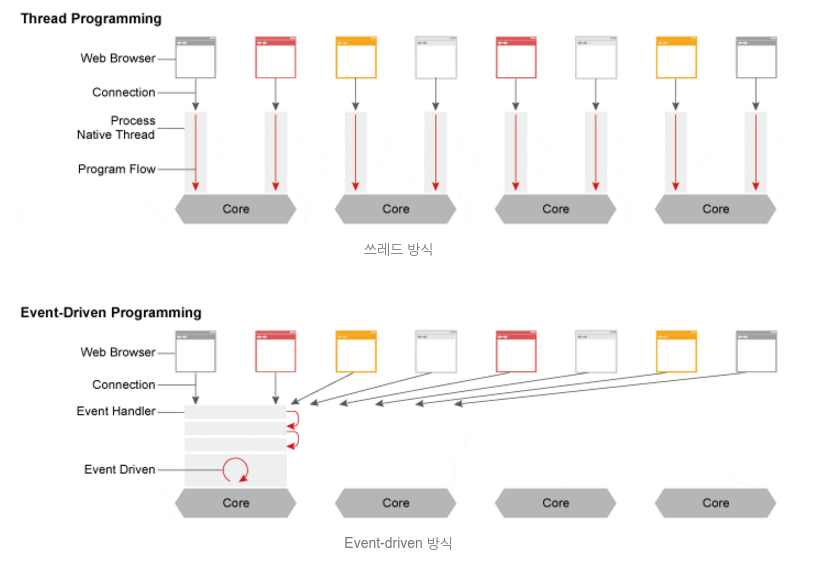
### 응용프로그램 서버에 요청을 보내는 리버스 프록시로서의 역할



두번째 역할은 리버스 프록시(reverse proxy) 인테, 한마디로 말하면 클라이언트는 가짜 서버에 요청(request)하면, 프록시 서버가 배후 서버(reverse server)로부터 데이터를 가져오는 역할을 한다. 여기서 프록시 서버가 Nginx, 리버스 서버가 응용프로그램 서버를 의미한다.

웹 응용프로그램 서버에 리버스 프록시(Nginx)를 두는 이유는 요청(request)에 대한 버퍼링이 있기 때문이다. 클라이언트가 직접 App 서버에 직접 요청하는 경우, 프로세스 1개가 응답 대기 상태가 되어야만 한다. 따라서 프록시 서버를 둠으로써 요청을 배분하는 역할을 한다.

Nginx.conf파일에서 location 지시어를 사용하여 요청을 배분한다.



Nginx는 비동기 처리방식(Event-Drive) 방식을 채택하고 있다.

## AWS에 nginx 설치하기

OS : Ubuntu 16.04 LTS

본 문서는 AWS 인스턴스(Ubuntu 16.04 LTS)에서 설치를 진행한다.

### 설치 및 제거

|  |
| --- |
| sudo apt-get install nginx # 설치  sudo apt-get remove nginx # 제거 |

|  |
| --- |
| $> nginx –v  nginx version : nginx/1.13.2 |

nginx version : nginx/1.13.2

nginx –v 했을 때 위와 같은 메시지가 출력할 것이다.

### Nginx 경로

* apt-get을 이용한 패키지 설치 방법을 이용하게 되면 기본적으로 /etc/nginx 폴더에 설치된다. 본 문서는 이 방식을 사용한다.
* 직접 compile한 경우에는 /usr/local/nginx/conf 또는 /usr/local/etc/nginx 에 설치된다.

|  |
| --- |
| sudo find / -name nginx.conf |

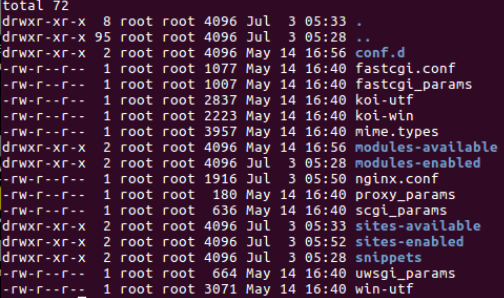
위 명령어를 이용해 nginx가 설치된 경로를 찾아본다.

## Nginx 디렉토리 구조 살펴보기

sudo find / -name nginx.conf

명령어를 통해 nginx.conf 가 위치한 경로를 이동한다.

|  |
| --- |
| cd /etc/nginx  ls –al |



|  |
| --- |
| |-- conf.d # (디렉토리) nginx.conf에서 불러들일 수 있는 파일을 저장  |-- fastcgi.conf # (파일) FASTCGI 환경설정 파일  |-- fastcgi\_params  |-- koi-utf  |-- koi-win  |-- mime.types  |-- nginx.conf # 접속자 수, 동작 프로세스 수 등 퍼포먼스에 관한 설정들  |-- proxy\_params  |-- scgi-params  |-- sites-available # 비활성화된 사이트들의 설정 파일들이 위치한다.  |-- default  |-- sites-enabled # 활성화된 사이트들의 설정 파일들이 위치한다. 존재하지 않은 경우에는 디렉토리를 직접 만들 수도 있다.  |-- default -> /etc/nginx/sites-avilable/default  |-- snippets  |-- fastcgi-php.conf  |-- snakeoil.conf  |-- uwsgin\_params  |-- win-utf |

트리명령을 사용하려면 sudo apt-get install tree를 실시한다. 설치가 완료되면 tree-d 또는 tree .로 확인 가능하다.

## nginx.conf 기본 환경설정 튜닝하기

가장 핵심이 되는 nginx.conf 파일을 먼저 살펴보자. nginx.conf 파일은 Nginx가 동작해야 할 방식을 설정 값을 통해 지정한다.

이 파일은 root 계정만 수정이 가능하기 때문에, 수정이 필요하다면 sudo vim nginx.conf를 통해 파일을 열어야 한다. 명령어를 통해 nginx.conf 파일을 열어보면 아래와 같은 형식을 갖는다. 얼핏보면 json 같기도 하다.



### 최상단 (Core 모듈)

|  |
| --- |
| user nginx; # (디폴트값 : www-data)  worker\_processes 1;  error\_log /var/log/nginx/error.log warn;  pid /var/run/nginx.pid |

1. user : NGINX 프로세스가 실행되는 권한

* nginx는 마스터(master)와 워커(worker) 프로세스로 나눈다.
* 워커 프로세스가 실질적인 웹서버 역할을 수행한다.
* user 지시어는 워커 프로세스의 권한을 지정한다.
* 워커프로세스를 악의적 사용자가 제어하면 해당 머신을 최고 사용자의 권한으로 원격제어하는 것이기 때문에 위험하다.
* 본 문서는 nginx로 변경하였다.

1. work\_processes : NGINX 프로세스 실행 가능 수

* 위에서 언급한 워커프로세스이다. 실질적인 웹서버 역할을 한다.
* auto도 무방하지만, 명시적으로 서버에 장착되어 있는 코어 수 만큼 할당하는 것이 보통이며, 더 높게도 설정 가능하다.

1. pid : NGINX 마스터 프로세스 ID 정보가 저장된다.

### events 블락

|  |
| --- |
| events {  worker\_connections 1024;  # multi\_accept on; (디폴트값 : off)  } |

* NGINX의 특징인 비동기 이벤트 처리 방식에 대한 옵션을 설정한다.
* Worker\_connections는 하나의 프로세스가 처리할 수 있는 커넥션의 수를 의미한다.
* 최대 접속자수 = worker\_processes X worket\_connections가 된다.

### http 블락

|  |
| --- |
| http {  include /etc/nginx/mime.types;  default\_type application/octet-stream;  log\_format main ‘$remote\_addr - $remote\_user [$time\_local] “$request” ‘  ‘$status $body\_bytes\_sent “$http\_referer” ‘  ‘”$http\_user\_agent” “$http\_x\_forwarded\_for”’;  access\_log /var/log/nginx/access.log main;  send\_file on;  #tcp\_nopush on;  Keepalive\_timeout 65;  #gzip on;  Include /etc/nginx/conf.d/\*.conf;  } |

* keepalive\_timeout : 접속시 커넥션을 몇 초동안 유지할지에 대한 설정값. 이 값이 높으면 불필요한 커넥션(접속)을 유지하기 때문에 낮은값 또는 0을 권장한다. (default = 10)
* servers token : NGINX의 버전을 숨길것인가에 대한 옵션이다. 보안상 주석을 제거하여 설정하는 것이 좋다.
* Types\_hash\_max\_size, server\_names\_hash\_bucket\_size 호스트의 도메인 이름에 대한 공간을 설정하는 것으로 이 값이 낮을 경우 많은 가상 호스트 도메인을 등록한다거나, 도메인 이름이 길 경우 bucket 공간이 모자라 에러가 생길 수 있으므로 넉넉하게 설정한다.

### 기타

include옵션 : 가상 호스트 설정이나 반복되는 옵션 항목을 include를 통해 불러올 수 있다. Ex) 리버스 프록시를 각 도메인에 설정한다고 했을 때 헤더 처리 옵션등을 conf.d 디렉토리에 넣어 두고 include 명령을 통해 불러 올 수 있다.

### 설정 파일의 반영

설정 파일의 내용을 변경한 후에는 NGINX에 반영해야 하는데, reload 명령을 이용한다.

|  |
| --- |
| sudo service nginx reload; |

### 참고 사이트

#### 생활 코딩 <https://opentutorials.org/module/384/3462> (2013.05.06)

* 초심자 수준에서 가장 쉽게 설명한 사이트이다. NGINX 뿐만 아니라, 네트워크 환경(웹, 웹서버, 서버와 클라이언트)에 대해 자세히 설명하였다.
* 훌륭한 강의의지만, 4년이 지났기 때문에 초반부터 살펴보고 좀더 최신의 포스팅을 볼 수 있다.

#### NGINX 기본 환경 설정 튜닝 및 설명 <https://extrememanual.net/9976> (2017.03.08)

* nginx 디렉토리 및 nginx.conf 파일 설정값에 대한 도움을 많이 받을 수 있음
* 현재로서 가장 최신 글이므로 참고할 만함.