

(HTTP 완벽 가이드) V. 콘텐츠 발행 및 배포

18 웹 호스팅

CodeDiary18(codediary18@gmail.com)

2022.03.29

목차

- ✓ 18.01 호스팅 서비스
- ✓ 18.02 가상 호스팅
- ✓ 18.03 안정적인 웹 사이트 만들기
- ✓ 18.04 웹 사이트 빠르게 만들기

웹 호스팅이란?

콘텐츠 리소스를 **저장, 중개, 관리**하는 일

18.01 호스팅 서비스

✓ 월드 와이드 웹 초기

- 각 회사가 자체 컴퓨터 하드웨어를 구매하고
- 자체 컴퓨터 망을 구축하며
- 자체 네트워크 연결을 확보하고
- 자체 웹 서버 소프트웨어를 관리



18.01 호스팅 서비스

✓ 웹이 빠르게 대세가 되면서...

- 냉난방 장치가 있는 서버실을 짓고, 도메인 이름을 등록하고, 네트워크 대여폭을 구매할 기술과 시간을 가진 사람은 드물었음

✓ 시간을 절약하기 위해...

- 전문적으로 관리하는 웹 호스팅 서비스를 제공하는 여러 신사업이 만들어짐
 - 물리적인 장비관리(공간, 냉난방, 연결)부터 고객이 직접 콘텐츠를 제공할 수 있는 총체적인 웹 호스팅까지 다양한 종류의 서비스

18.02 가상 호스팅

✓ 가상 호스팅이란?

- 많은 웹 호스팅 업자는 **컴퓨터 한 대를 여러 고객이 공유**하게 해서 저렴한 웹 호스팅 서비스를 제공
- 이를 **공유 호스팅** 혹은 **가상 호스팅**이라고 함
- 각 웹 사이트는 다른 서버에서 호스팅하는 것처럼 보이겠지만, 사실은 **물리적으로 같은 서버에서 호스팅 됨**
 - 최종 사용자의 관점에서 가상 호스팅에 있는 웹사이트는, 물리적으로 분리된 전용 서버에서 호스팅하는 사이트와 구분할 수 없어야 함

18.02 가상 호스팅

✓ 가상 호스팅이란?

- 가상 호스팅은 비용, 공간, 관리에 이점을 가짐
 - 호스팅 업자는 복제 서버 더미(서버 팜)를 만들고 서버 팜에 부하를 분산할 수 있음
 - 팜에 있는 각 서버는 다른 서버를 복제한 것이며,
 - 수많은 가상 웹 사이트를 호스팅하고 있기 때문에 관리자는 훨씬 편해짐



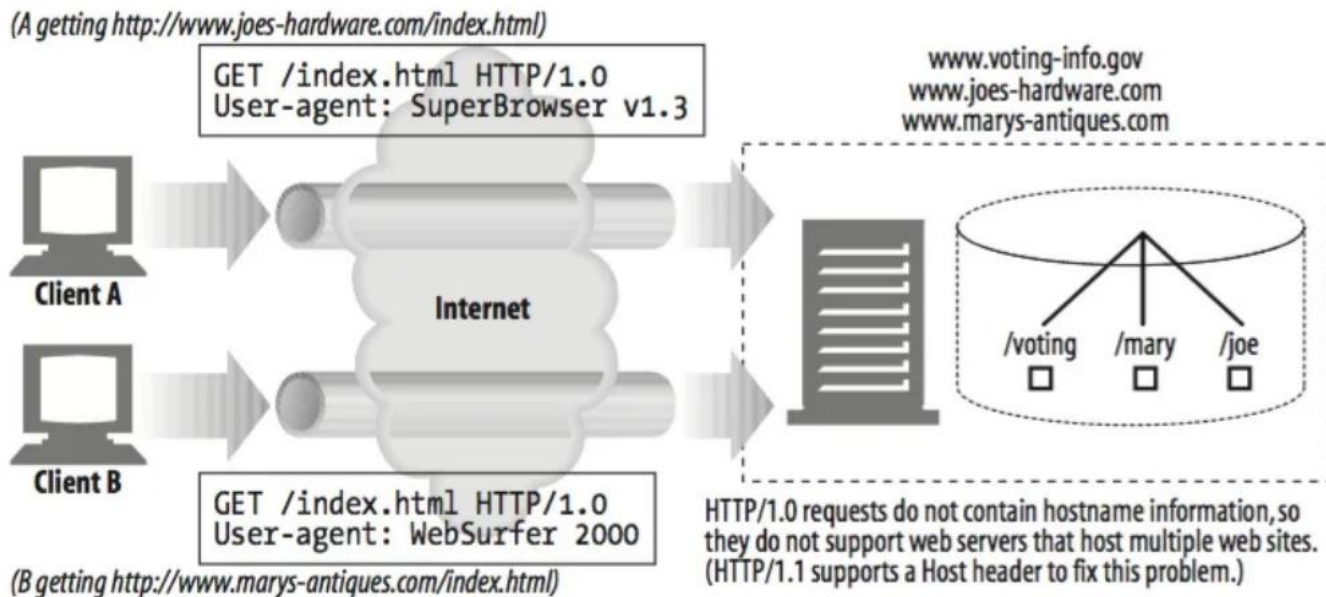
18.02 가상 호스팅

✓ 호스트 정보가 없는 가상 서버 요청

- HTTP/1.0 명세는 공용 웹 서버가 호스팅하고 있는 가상 웹 사이트에 누가 접근하고 있는지 식별하는 기능을 제공하지 않음
- 만일, `http://www.joes.com/index.html`을 요청한다면
- HTTP 1.0 요청은 호스트명에 대한 별다른 언급 없이 “GET /index.html”이라는 요청을 제공
 - 서버가 여러 개의 사이트를 가상 호스팅하고 있으면, 사용자가 어떤 가상 웹 사이트로 접근하려고 하는 것인지 아는 데 필요한 정보가 충분하지 않음

18.02 가상 호스팅

- ✓ 호스트 정보가 없는 가상 서버 요청



18.02 가상 호스팅

- ✓ 호스트 정보가 없는 가상 서버 요청

- HTTP 대리 서버(리버스 프락시)와 인터셉트 프락시는 어떤 사이트를 요청하는지에 관한 정보가 필요

18.02 가상 호스팅

✓ 가상 호스팅 동작하게 하기

- HTTP 설계자들은 공유 서버인 가상 호스팅을 고려하지 않음
- 이를 해결하기 위해, HTTP 요청 메시지에 완전한 URL도 포함해서 보냄

18.02 가상 호스팅

✓ 가상 호스팅 동작하게 하기

○ URL 경로를 통한 가상 호스팅

- 서버가 어떤 사이트를 요청하는 것인지 알 수 있게 URL에 특별한 경로 컴포넌트를 추가
 - 조의 컴퓨터 가게 : `http://www.joes-hardware.com/joe/index.html`
요청 - GET /joe/index.html
 - 메리의 골동품 가게 : `http://www.marys-antiques.com/mary/index.html`
요청 - GET /mary/index.html
- 좋지 않은 방법이라 거의 사용하지 않음

18.02 가상 호스팅

✓ 가상 호스팅 동작하게 하기

○ 포트번호를 통한 가상 호스팅

- 각 사이트에 **다른 포트번호를 할당**하여, 분리된 웹 서버의 인스턴스가 요청을 처리
 - 조의 컴퓨터 가게 : 82번 포트 사용
 - 메리의 골동품 가게 : 84번 포트 사용
- URL 경로를 통한 가상 호스팅과 유사한 문제점을 가지고 있음
 - 사용자가 URL에 비표준 포트를 쓰지 않고 리소스를 찾길 원하기 때문

18.02 가상 호스팅

✓ 가상 호스팅 동작하게 하기

○ IP 주소를 통한 가상 호스팅

- 각 가상 사이트에 **별도의 IP 주소를 할당**하고, 모든 IP 주소를 장비 하나에 연결
- 웹 서버는 IP 주소로 사이트 이름을 식별
 - 조의 컴퓨터 가게 : 209.172.34.3 주소 할당
 - 메리의 골동품 가게 : 209.172.34.4 주소 할당

18.02 가상 호스팅

✓ 가상 호스팅 동작하게 하기

○ IP 주소를 통한 가상 호스팅

■ 문제점

- 컴퓨터 시스템이 연결할 수 있는 장비의 IP의 개수에 제한이 있음
- 가상 사이트를 많이 가지고 있는 호스팅 업자는 호스팅하는 모든 웹 사이트에 할당할 가상 IP 주소를 충분히 얻지 못할 수 있음
- 부하 균형의 구조상, 용량을 늘리기 위해 서버를 복제하면서, 각 복제된 서버에 IP 주소를 부여해야 하므로 IP주소는 복제 서버의 개수만큼 더 필요

18.02 가상 호스팅

✓ 가상 호스팅 동작하게 하기

- Host 헤더를 통한 가상 호스팅

- 모든 요청에 호스트명을 Host 확장 헤더에 기술해서 전달

18.02 가상 호스팅

✓ HTTP/1.1 Host 헤더

- 가상 서버는 매우 흔하기 때문에 대부분의 HTTP 클라이언트가 HTTP1.1과 호환되지 않더라도, Host헤더는 구현

■ 사용 방법

Host = "Host" ":" 호스트[":" 포트]

▼ Request Headers [View source](#)

```
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/ebp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Accept-Encoding: gzip, deflate, br
Accept-Language: ko-KR,ko;q=0.9
Cache-Control: max-age=0
Connection: keep-alive
Cookie: wcs_bt=ef4f35a9c22c50:1648510774; __T_=1; adfit_sdk_id=a3d9-a569-ecaac264e8e3; _T_ANO=HaeIqyPXMHVc19wjaUU6T0cLtgn+wHIohdasMUGLK2Brwgp9d1Gf+ksqnYGERNQb2IfC8Xyggffg2C60Auz9YD7QtpgZq9+GrOPX16zw1RwSVB1+pOJAJoCB4otm55+7XiAkeastow05v0VNQec330EvL34NtnbHObNmbrf6a2Mxx2H+oesJl6qaT5/U3oi0c75fwsJG7f15L1GcDAGuKK11TjMj+jagi1EqDBir0/yLQe4ntk5SajRfJQ9vrXeNPYlw521GJ/9ZthcKM7uHuU1wv9JMqEcNkQ==
Host: computer-science-student.tistory.com
```

18.03 안정적인 웹 사이트 만들기

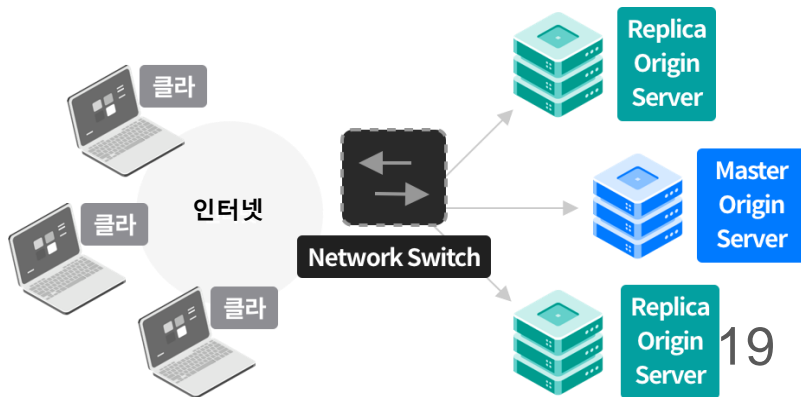
✓ 웹 사이트에 장애가 생기는 상황

- 서버 다운
- 트래픽 폭증
 - 트래픽 폭증으로 인해 웹 서버에 과부하를 주어 느려지게 하거나 완전히 멈춤
- 네트워크 장애나 손실

18.03 안정적인 웹 사이트 만들기

✓ 미러링 된 서버 팜

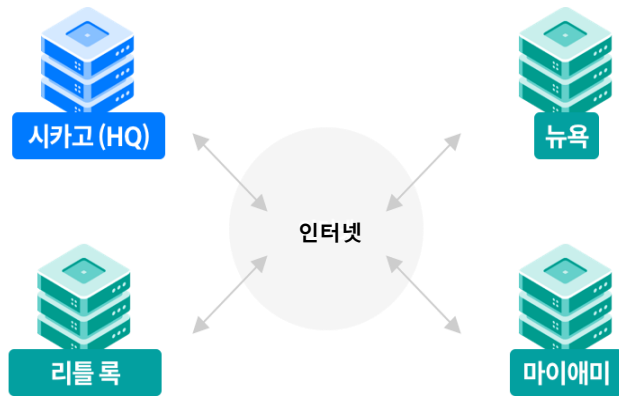
- 서버 팜은 서로 대신할 수 있고 식별할 수 있게 설정된 웹 서버들의 집합
- 서버 팜의 서버에 있는 콘텐츠들은 한 곳에 문제가 생기면 다른 한 곳에서 대신 전달할 수 있게 미러링 할 수 있음
- 미러링 된 서버는 계층적인 관계에 있음
 - **마스터 원 서버(Master Origin Server)**
한 서버(원본 콘텐츠를 가지고 있는)는 '콘텐츠의 원본 제작자'와 같이 행동
 - **복제 원 서버(Replica Origin Server)**
마스터 원 서버로부터 콘텐츠를 받은 미러링된 서버



18.03 안정적인 웹 사이트 만들기

✓ 미러링 된 서버 팜

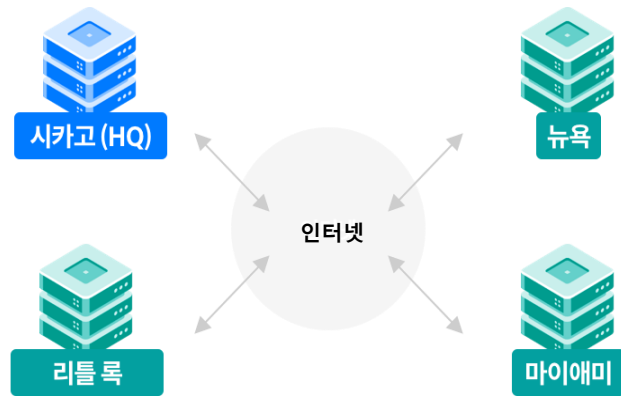
- 서버 팜에 배포하는 간단한 방법
네트워크 스위치를 사용해서 서버에 분산 요청 보내는 것
 - 서버에 호스팅 되고 있는 각 웹 사이트의 IP 주소는 스위치의 IP 주소가 됨
- 마스터 원 서버는 복제 원 서버에 콘텐츠를 보낼 책임이 있음
 - 외부에서 볼 때, 이 콘텐츠를 가리키는 IP 주소는 스위치의 IP 주소
스위치는 서버에게 요청을 전송해야 하는 책임이 있음
- 미러링 된 웹 서버에는 다른 위치에 있는 콘텐츠와 정확히 같은 복제본이 있음



18.03 안정적인 웹 사이트 만들기

✓ 클라이언트의 요청이 특정 서버로 가는 방법

- HTTP 리다이렉션
콘텐츠에 대한 URL은 마스터 서버의 IP를 가리키고,
마스터 서버는 요청을 받는 즉시 복제 서버로 리다이렉트 시킴
- DNS 리다이렉션
콘텐츠의 URL은 4개의 IP 주소를 가리킬 수 있고,
DNS 서버는 클라이언트에게 전송할 IP 주소를 선택할 수 있음



18.03 안정적인 웹 사이트 만들기

- ✓ 콘텐츠 분산 네트워크(CDN, Contents Delivery Network)
 - CDN은 특정 콘텐츠의 분산을 목적으로 하는 단순한 네트워크
 - 네트워크의 노드는 서버, 대리 서버, 혹은 프락시 서버가 될 수 있음

18.03 안정적인 웹 사이트 만들기

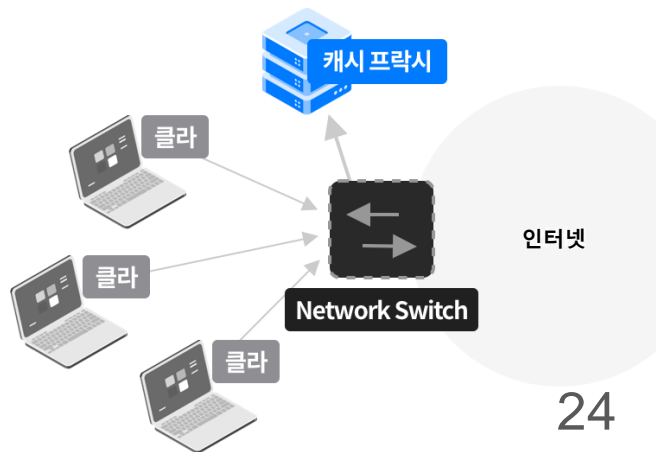
✓ CDN 의 대리 캐시(==리버스 프락시)

- 대리 캐시는 복제 원 서버를 대신해 사용될 수 있음
- 대리 서버는 미러링 된 웹 서버처럼 콘텐츠에 대한 요청을 받음
- 특정 원 서버 집합을 대신해 요청을 받음
- 대리 서버와 미러링 된 서버의 차이점
 - 대리 서버는 보통 수요에 따라서 동작
 - 대리 서버는 원 서버의 전체 콘텐츠를 복사하지 않음
 - 클라이언트가 요청하는 콘텐츠만 저장

18.03 안정적인 웹 사이트 만들기

✓ CDN의 프락시 캐시

- 대리 서버를 사용하면, 프락시 캐시의 콘텐츠는 요청이 있을 때만 저장되고 원본 서버 콘텐츠를 정확히 복제한다는 보장이 없음
- 요청이 있을 때만 저장하는 프락시 캐시는 스위치 혹은 라우터가 중간에서 웹 트래픽을 가로채 처리함



18.04 웹 사이트 빠르게 만들기

- ✓ 서버 팜이나 분산 프락시 캐시나 대리 서버는 혼잡을 조절하고 네트워크 트래픽을 분산
- ✓ 콘텐츠를 분산시키면,
 - 그 콘텐츠를 사용자에게 더 가깝게 만들어 줌
→ 콘텐츠를 서버에서 클라이언트로 전송하는 시간이 단축
- ✓ 리소스의 로딩속도를 좌우하는 요소
 - 어떻게 요청과 응답이 클라이언트와 서버 사이에서 연결을 맺고 인터넷을 가로질러 데이터를 전송하는 지 → 리다이렉트
 - 콘텐츠 인코딩

참고 및 사진 출처

- HTTP 완벽 가이드
- <https://bebiangel.github.io/2020/02/03/http-guide-chap18/>
- https://feel5ny.github.io/2020/02/02/HTTP_018/
- https://kr.freepik.com/free-vector/people-holding-world-wide-web-icons_3425163.htm
- <https://www.breakingviews.com/wp-content/uploads/2017/10/datacenter.jpg>

QnA