

WEB APPLICATION HOSTING IN THE AWS CLOUD

www.wisen.co.kr



Wisely Combine the Network platforms

WEB APPLICATION HOSTING IN THE AWS CLOUD

Contents

개요/대상	1
전통적인 웹 호스팅의 개요	2
아마존 웹 서비스를 사용하는 클라우드의 웹 어플리케이션 호스팅	3
AWS는 어떻게 일반적인 웹 응용프로그램 호스팅 문제를 해결하는가 웹 호스팅에 대한 AWS 클라우드 아키텍처 AWS 웹 호스팅 아키텍처의 주요 컴포넌트	
AWS 웹 호스팅에 주요 고려 사항	12

개요

가용성과 확장성이 높은 웹 호스팅은 복잡하고 비용이 많이 드는 사업이 될 수 있습니다. 전통적인 웹 확장 아키텍처는 높은 수준의 안정성을 보장하기 위해 복잡한 솔루션으로 구현해야 했고, 또한 높은 수준의 고객 서비스를 제공하기 위해서는 정확한 트래픽 예측이 필요합니다. 변동폭이 크거나 피크 기간이 빈번한 트래픽 패턴은 비싼 하드웨어의 사용율을 저하하며, 대기 상태의 하드웨어를 유지하는데 높은 비용을 감당해야할 뿐만아니라 하드웨어의 성능을 충분히 사용할 수 없게 합니다.

아마존 웹 서비스는 가장 까다로운 웹 어플리케이션을 위한, 신뢰할 수 있는 확장 성과, 보안 성이 우수하며, 고도의 수행 인프라로, 실시간으로 고객의 트래픽 패턴과 IT비용이 일치하는 인프라를 제공합니다.

대상

이 백서는 온 디맨드 컴퓨팅 요구사항을 충족하는데 필요한 확장성을 달성 할 수 있는 클라우드를 찾는 IT관리자와 시스템 아키텍트를 위한 것입니다.

전통적인 웹 호스팅의 개요

외부 방화벽

스탠다드 포트들을 열기 위한 하드웨어 또는 소프트웨어 솔루션입니다. (80, 443)

웹로드 밸런서

웹 서버를 통해 트래픽을 분산하기 위한 하드웨어 또는 소프트웨어 솔루션입니다.

웹 계층

HTTP 요청을 처리하는 장비의 그룹.

백 엔드 방화벽

웹 계층에서 어플리케이션에 접근하는 것을 제한합니다.

어플리케이션의 로드 밸런서

어플리케이션 서버를 통해 트래픽을 분산하기 위한 하드웨어 또는 소프트웨어입니다.

어플리케이션 서버 계층

어플리케이션 워크로드를 처리하는 장비 그룹. 캐시 서버는 이 계층에서 구현될 수 있습니다.

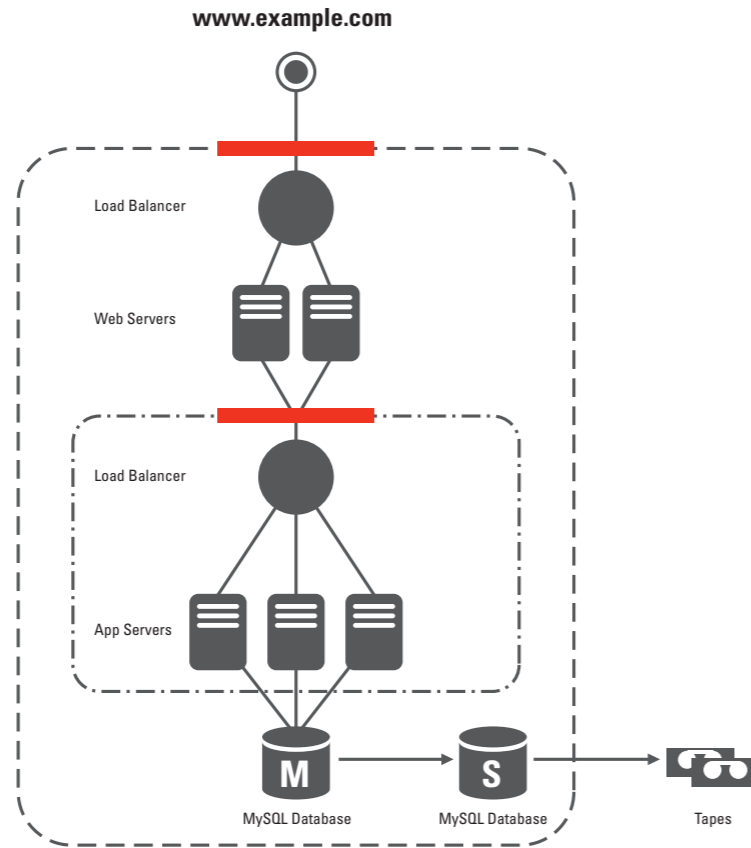
데이터 계층

마스터와 로컬 별도로 실행되는 데이터베이스 서버 시스템, 정적 객체에 대한 네트워크 스토리지입니다.

테이프에 백업

일반적으로 서드파티에 의해 관리되어 테이프에 주기적으로 백업.

전통적인 웹 호스팅 모델을 표현한 다음 그림은 확장성 있는 웹 호스팅의 일반적인 문제들을 포함하고 있습니다. 이 글의 주제인 클라우드에서의 배포에 대해 설명하기 위해 클라우드의 아키텍처와 비슷하게 표현하였습니다.



[그림1. 전통적인 웹 어플리케이션 아키텍처]

이 전통적인 웹 호스팅 아키텍처는 프리젠테이션, 응용프로그램, 퍼시스턴스 계층으로 구분되는 일반적인 3티어 웹 어플리케이션 모델을 나타냅니다. 확장성은 각 계층에 서버를 추가하여 얻을 수 있습니다. 또한 고성능, 장애복구 및 가용성을 내재해야 합니다. 다음 섹션에서는 왜 그리고 어떻게 이러한 아키텍처가 아마존 웹 서비스 클라우드에서 배포되어야 하는지 살펴보겠습니다.

아마존 웹 서비스를 사용하는 클라우드의 웹 어플리케이션 호스팅

전통적인 웹 호스팅 아키텍처(그림1)는 AWS 제품으로 제공되는 클라우드 서비스에 약간의 수정으로 쉽게 이식이 가능하지만, 기존 웹 어플리케이션 호스팅 솔루션을 AWS 클라우드로 이동하는 것의 가치에 대해서 먼저 생각해봐야 할 것입니다. 클라우드를 사용할 것이라고 결정했다면 그에 적합한 아키텍처가 필요할 것입니다. 이 섹션은 AWS 클라우드 솔루션을 평가하는데 도움이 됩니다. 온-프레미스와 클라우드에서의 배포를 비교하고, 어플리케이션을 호스팅하기 위한 AWS 클라우드 아키텍처를 제시하며, AWS의 주요 컴포넌트에 대해 설명합니다.

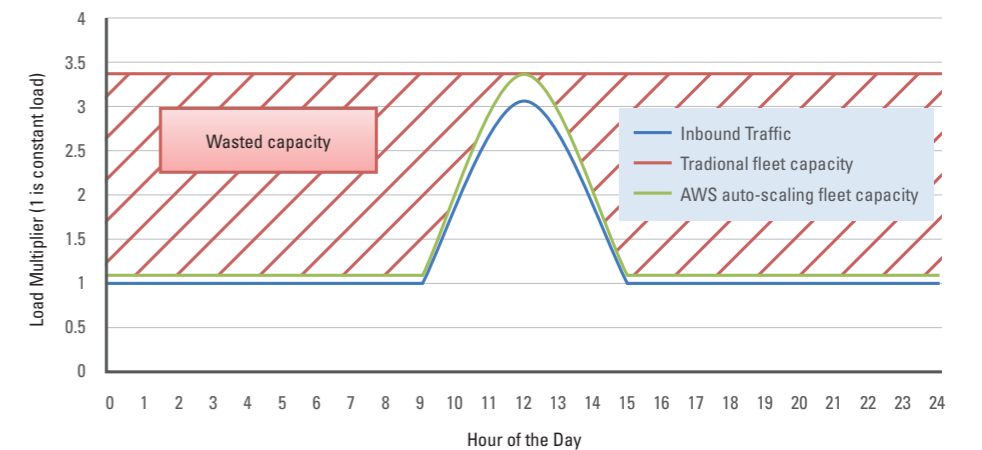
AWS는 어떻게 일반적인 웹 어플리케이션 호스팅 문제를 해결하는가

웹 어플리케이션 관리자라면 인프라와 아키텍처의 다양한 문제에 직면하고 있을 것입니다. AWS는 원활하고 비용 효율적인 솔루션을 제공합니다. 다음은 전통적인 호스팅 모델에 AWS를 사용하는 장점들 중 일부입니다.

피크 트래픽 처리에 필요한 대형 인프라 그룹의 비용 효율적인 대안

전통적인 호스팅 모델에서, 서버는 피크 용량을 처리하기 위해 준비되어 있어야 하며 그 외의 기간에는 낭비되었습니다. AWS에서 호스팅되는 웹 어플리케이션은 추가 서버의 온 디맨드 프로비저닝을 활용할 수 있고, 그러므로 항상 실제 트래픽 패턴에 대한 용량 및 비용을 조정할 수 있습니다.

예를들어, 다음 그래프는 웹 어플리케이션이 오전 9시부터 오후 3시까지 사용량이 많고, 나머지 시간동안은 사용량이 적은 것을 보여줍니다. 실제 트래픽 흐름에 기반하여 자원을 필요시에만 공급하는 Auto-scaling은 낭비를 줄이고 50% 이상의 비용절감 효과를 제공합니다.



[그림2. 일반적인 호스팅 모델에서 낭비되는 용량의 예]

예기치 않은 트래픽 피크를 처리하는 확장 가능한 솔루션

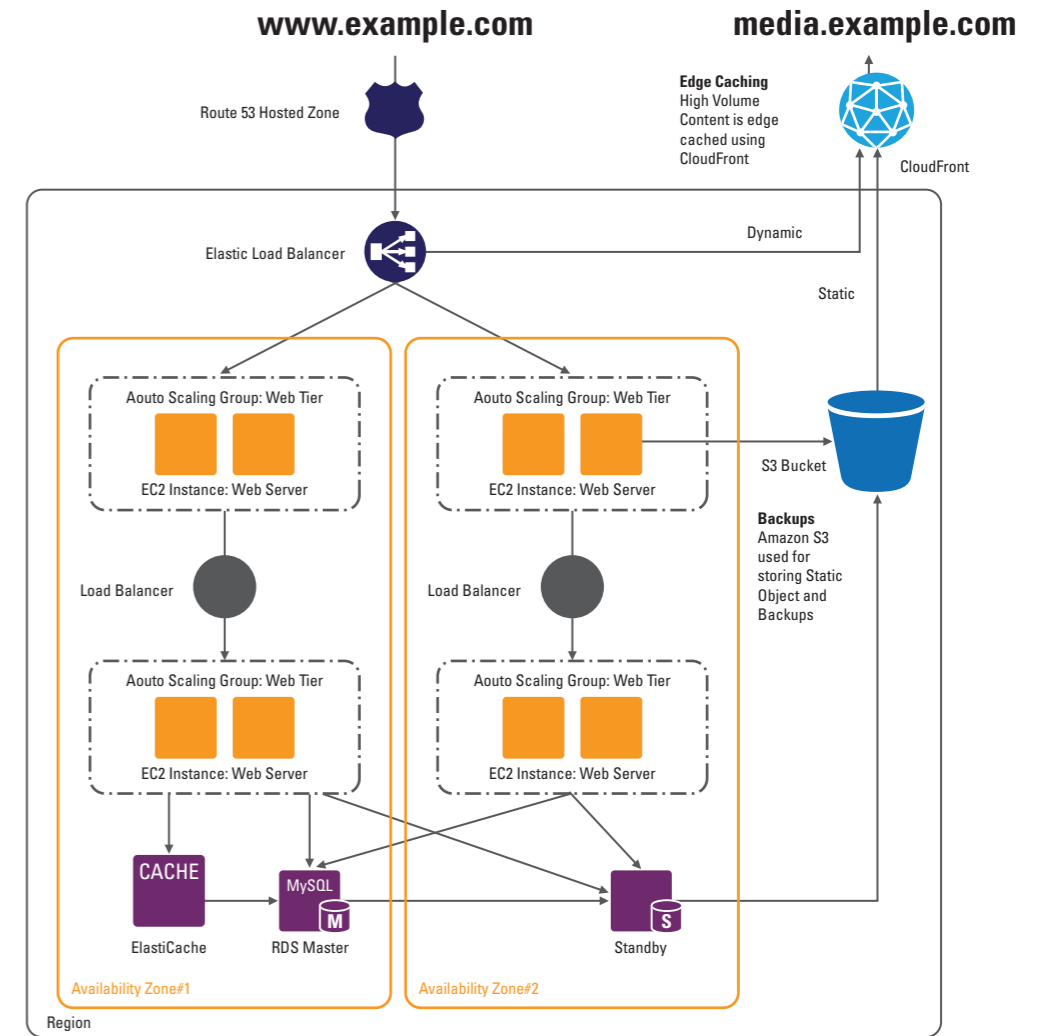
전통적인 호스팅 모델에서 느린 프로비저닝의 심각한 문제는 예상치 못한 트래픽 스파이크때 제 시간에 대응하지 못하는 것입니다. 사이트가 인기있는 매체에서 언급된 후 예기치 않은 스파이크의 트래픽 때문에 웹 어플리케이션이 문제되는 것에 대한 많은 사례가 있습니다. 이와 같은 웹 어플리케이션의 스케일링하는 데 도움을주는 온 디맨드 기능은 예기치 않은 로드를 처리 할 수 있습니다. 새로운 호스트들은 시작과 준비가 단시간내에 가능하고 트래픽이 정상으로 돌아올 때 인프라를 반환하여 재빠르게 오프라인으로 만들 수 있습니다.

테스트, 부하, 베타 및 사전 제작 환경을 위한 온 디맨드 솔루션

제품 웹 어플리케이션에 대한 전통적인 호스팅 환경을 구축하는 하드웨어 비용은 생산 그룹에 국한되지 않습니다. 대개 사전제작, 베타 및 테스트 그룹 또한 개발 주기의 각 단계에서 웹 어플리케이션의 품질을 보장하기 위해 만들어질 필요가 있습니다. 다양한 최적화된 이 하드웨어의 테스트는 높은 이용 가능성을 보장 할 수 있지만, 이러한 병렬화된 그룹들은 항상 최적으로 이용되지는 않습니다. 수많은 고가의 하드웨어들이 장시간동안 사용되지 않은 채로 있습니다. AWS 클라우드에서는 테스트 제공 그룹은 필요 할 때 공급됩니다. 그리고, 부하 테스트 동안 유저 트래픽을 AWS 클라우드에서 시뮬레이션 할 수 있습니다. 또한 신제품 릴리즈를 위한 이러한 병렬 그룹으로 현재 제품에서 새로운 버전으로 거의 무중단으로 전환 가능합니다.

웹 호스팅에 대한 AWS 클라우드 아키텍처

다음은 고전적인 웹 어플리케이션 아키텍처의 또 다른 모습이며, AWS 클라우드 컴퓨팅 인프라를 활용할 수있는 방법을 보여 줍니다.



[그림3. AWS에서 웹 호스팅 아키텍처의 예]

[Route 53 : 간편한 도메인 관리와 zone APEX 지원을 제공하는 DNS 서비스]

Elastic Load Balancer

ELB는 웹 서버에 트래픽을 분산합니다.

외부 방화벽은 보안 그룹을 통해 모든 웹 서버 인스턴스로 이동합니다.

오토 스케일링 기반 웹 계층

HTTP 요청을 처리하는 EC2 인스턴스의 그룹입니다.

백엔드 방화벽은 모든 백엔드 인스턴스로 이동되었습니다.

어플리케이션 서버로드 밸런서

EC2 인스턴스의 소프트웨어 LB (예 HAProxy)는 어플리케이션 서버 클러스터 전체로 트래픽을 분산합니다.

Auto Scaling App Tier

실제 어플리케이션을 실행중인 EC2 인스턴스의 그룹입니다. 인스턴스는 Auto Scaling 그룹에 속합니다.

ElastiCache

데이터베이스를 부하를 줄이는 인 메모리 캐싱 서비스를 제공합니다.

DB 계층

MySQL의 RDS DB 다중 AZ 배포와 고 가용성 아키텍처를 만듭니다. 읽기 전용 복제본도 많이 사용하는 어플리케이션을 읽어 확장하는 데 사용할 수 있습니다.

에지 캐싱

높은 볼륨의 콘텐츠는 CloudFront를 사용하여 에지 캐시됩니다.

백업

아마존 S3는 정적 개체 및 백업을 저장하는 데 사용됩니다.

AWS 웹 호스팅 아키텍처의 주요 컴포넌트

다음 섹션에서는 AWS 클라우드에 배포된 웹 호스팅 아키텍처의 주요 구성 요소 중 일부를 간략히 설명하고, 그들이 어떻게 전통적인 웹 호스팅 아키텍처와 다른지에 대해 설명합니다.

콘텐츠 딜리버리

아마존 웹 서비스 클라우드에서 엣지 캐싱도 할 수 있습니다. 웹 어플리케이션의 기존 솔루션도 AWS 클라우드에서 잘 작동할 것입니다. AWS로 엣지 캐싱 기능을 사용하기 위해서는 아마존 CloudFront 서비스를 이용하면 됩니다.

아마존 CloudFront는 글로벌 네트워크에 위치한 엣지들을 사용하여 동적, 정적 및 스트리밍 콘텐츠를 포함한 웹 사이트를 제공하기 위해 사용될 수 있습니다. 콘텐츠에 대한 요청은 자동으로 가장 가까운 엣지로 라우팅되어 최상의 성능을 제공합니다. 아마존 CloudFront는 아마존 Simple Storage Service(아마존 S3)나 아마존 Elastic Compute Cloud(아마존 EC2)와 같은 다른 아마존 웹 서비스와 함께 작동하도록 최적화되어 있습니다. 또한, 아마존 CloudFront는 파일의 최종 버전을 저장하고 있는 비 AWS 원본 서버와도 원활하게 작동합니다.

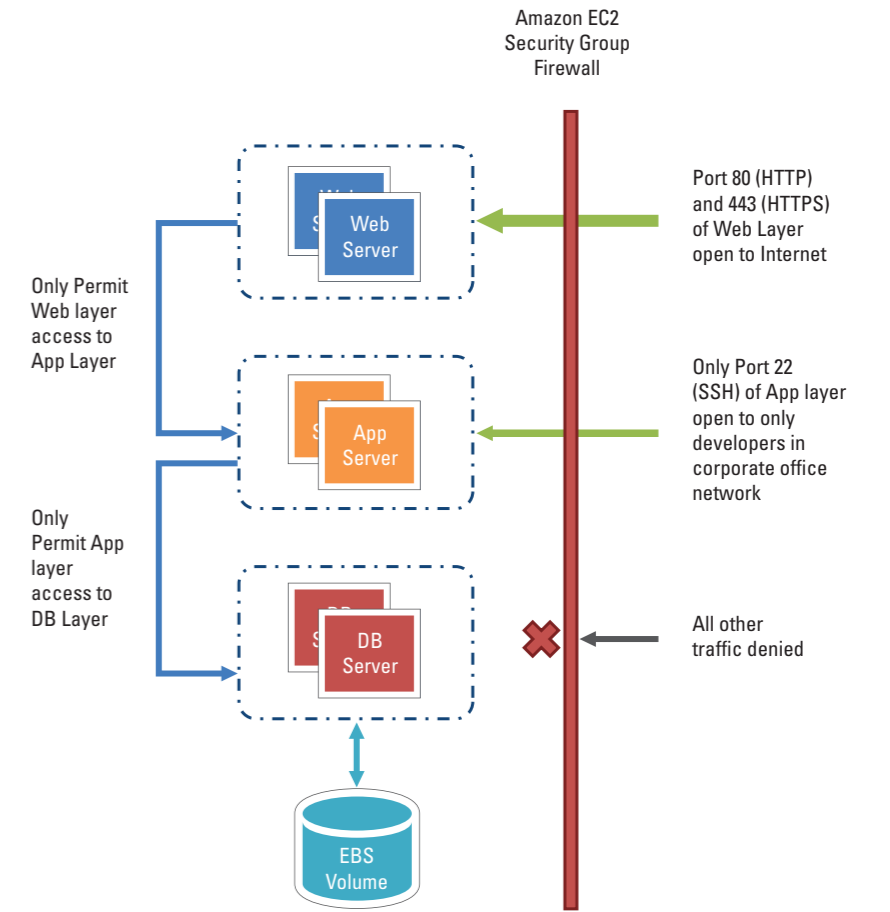
아마존 CloudFront는 다른 아마존 웹 서비스와 마찬가지로 사용에 대한 계약 혹은 약정이 없습니다. 실제로 서비스를 제공한 만큼의 비용만 지불하면 됩니다.

퍼블릭 DNS 관리

웹 어플리케이션을 AWS로 이전할 때 AWS에서 제공하는 여러 가용 영역의 장점을 활용하기 위해서 DNS 변경이 필요합니다. 가용성과 확장성이 높은 DNS 서비스인 아마존 Route53을 제공합니다. 도메인에 대한 쿼리는 자동으로 가장 가까운 DNS 서버로 전달되므로 최상의 성능으로 응답됩니다. Route53은 ELB 도메인(예, www.example.com)이나 zone apex 레코드(example.com)에 대한 요청에 응답합니다.

호스트 보안

전통적인 웹 호스팅 모델과 달리 인바운드 네트워크 트래픽 필터링은 엣지에 국한하지 않고, 호스트 수준에도 적용되어야 합니다. 아마존 Elastic Compute Cloud(EC2)는 보안 그룹이라는 기능을 제공합니다. 보안 그룹은 EC2 인스턴스에 도달 할 수 있는 프로토콜, 포트 및 소스 IP의 범위를 지정하는 인바운드 네트워크 방화벽과 유사합니다. 각 EC2 인스턴스는 적절한 트래픽을 라우팅하는 하나 이상의 보안 그룹에 할당될 수 있습니다. 보안 그룹은 특정 서브넷이나 IP 주소에서만 EC2 인스턴스에 접근할 수 있도록 구성하거나 다른 보안 그룹을 참조하여 EC2 인스턴스에 접근 제한을 할 수 있습니다.



[그림4. 웹 어플리케이션의 보안 그룹]

위의 예에서 웹 서버 클러스터를 위한 보안 그룹은 TCP 80 및 443 포트(HTTP, HTTPS)로 접근 가능하게 하고, 어플리케이션 서버 보안 그룹은 서버 관리 목적으로 22(SSH) 포트를 허용할 수 있습니다. 다른 한편으로 어플리케이션 서버 클러스터의 보안 그룹은 웹 요청 처리와 사내 서브넷에서 서버 관리 목적으로 TCP 22포트를 통한 접근을 허용할 수 있습니다. 위 모델에서 관리자는 사내망에서 어플리케이션 서버에 직접 로그인한 다음 다른 서버 그룹에 접근할 수 있습니다.

클러스터들간의 로드 밸런싱

하드웨어 로드 밸런서는 기존의 웹 응용 프로그램 아키텍처에서 사용되는 일반적인 네트워크 장비입니다. AWS는 Elastic Load Balancing 서비스를 통해 이러한 기능을 제공합니다. ELB는 서버의 헬스 체크, 여러 가용 영역에 있는 EC2 인스턴스에 트래픽 분산 그리고 EC2 인스턴스의 동적인 추가, 삭제 기능을 지원합니다. ELB는 또한 고정된 CNAME을 사용하여 예측가능한 엔트리 포인트를 제공하여 동적으로 성장하거나 줄어드는 트래픽에 적절하게 로드 밸런싱할 수 있습니다. 고급 라우팅 요구사항으로 고정 세션 기능도 지원합니다. 좀 더 고급 로드밸런싱 기능이 필요하면 Zeus, HAProxy 와 nginx 같은 소프트웨어 로드 밸런싱 패키지를 EC2에서 실행할 수 있습니다. 그런 다음 DNS 변경을 최소화하기 위해 이러한 로드 밸런싱 EC2 인스턴스에 엘라스틱 IP 주소를 할당 할 수 있습니다.

다른 호스트 및 서비스 찾기

기존의 웹 호스팅 아키텍처에서는 사용자의 호스트의 대부분은 고정 IP 주소를 가지고 있습니다. 클라우드에서는, 호스트들 대부분은 동적 IP 주소를 가지게 될 것입니다. 모든 EC2 인스턴스가 공인 및 사설 DNS 항목을 모두 가질 수 있고, 인터넷을 통해 알려질 수 있지만 인스턴스를 시작할 때 DNS 항목과 IP 주소는 동적으로 지정됩니다. 그리고 그들은 수동으로 지정할 수 없습니다. 고정 IP 주소 (엘라스틱 IP는 AWS 용어)가 실행 된 후 실행중인 인스턴스에 할당 될 수 있습니다. 마스터 데이터베이스, 중앙 파일 서버와 EC2 기반 로드 밸런스와 같은 고정적인 엔드포인트가 필요한 인스턴스나 서비스에는 Elastic IP 주소를 사용해야 합니다.

웹 서버처럼 확장과 축소가 빈번한 서버들은 IP 주소들을 중앙 저장소에 등록하여 동적인 엔드포인트가 검색되도록 합니다. 대부분의 웹 어플리케이션 아키텍처는 항상 동작하는 데이터베이스가 있기 때문에 데이터베이스 서버는 검색 정보를 위한 일반적인 저장소로 사용됩니다. 고정 주소가 필요한 인스턴스는 인스턴스가 시작될 때 부트스트랩 스크립트에 의해 주소 풀에서 Elastic IP를 할당 받을 수 있습니다.

이 모델을 이용하여, 새로 추가 된 호스트는 부트 스트래핑 단계의 일부로서 데이터베이스에서 필요한 통신을 위한 엔드 포인트의 리스트를 요청할 수 있습니다. 데이터베이스의 위치는 실행되는 각 인스턴스에 전달되는 사용자 데이터로 제공 될 수 있습니다. 또는, 구성 정보를 저장하고 유지하기 위해 아마존 SimpleDB 서비스를 사용할 수 있습니다. 아마존 SimpleDB는 잘 알려진 엔드 포인트에서 사용할 수 있는 고가용성 서비스입니다.

웹 응용 프로그램 캐싱

인-메모리 응용 프로그램 캐시는 서비스에 부하를 줄일 수 있으며 자주 사용하는 정보를 캐싱하여 데이터베이스 계층에 대한 성능과 확장성을 향상시킬 수 있습니다. 아마존 ElastiCache는 클라우드에서 배치, 동작 그리고 메모리 캐시의 확장을 쉽게 해주는 웹 서비스입니다. 인-메모리 캐시는 자동으로 부하를 확장하고 자동으로 실패 노드를 교체하도록 구성 할 수 있습니다. 아마존 ElastiCache는 온-프레미스 솔루션에서 쉽게 이행할 수 있는 Memcached와 호환됩니다.

데이터베이스 구성, 백업 및 장애 복구

대부분의 웹 애플리케이션은 일반적으로 관계형 또는 NoSQL에 데이터베이스의 형태인 퍼시스턴스 형태를 포함합니다. AWS는 관계형 및 NoSQL의 데이터 베이스 인프라를 제공합니다. 다른 방법으로는 아마존 EC2 인스턴스에 자신의 데이터베이스 소프트웨어를 배포 할 수 있습니다. 여기 보시는 표는 더 자세히 설명하기 위해 이러한 옵션을 요약 한 것입니다.

	관계형 데이터 베이스 솔루션	NoSQL 솔루션
관리된 데이터 베이스 서비스	아마존 관계형 데이터 베이스 서비스 (RDS)-MySQL, 오라클, SQL 서버	아마존 DynamoDB
셀프 관리	아마존 EC2 인스턴스에 관계형 DBMS 호스팅	아마존 EC2 인스턴스에 NoSQL 솔루션을 호스팅

아마존 관계형 데이터베이스 서비스 (RDS)

아마존 RDS는 익숙한 MySQL, 오라클, 또는 Microsoft SQL Server 데이터베이스 엔진에 액세스 기능을 제공합니다. 이미 사용하고있는 코드, 어플리케이션 및 도구는 아마존 RDS와 함께 사용할 수 있습니다. 아마존 RDS 는 자동으로 데이터베이스 소프트웨어를 패치하고 데이터베이스를 백업하고, 사용자가 지정한 보유 기간동안 백업을 저장합니다. 또한 특정 시점에서의 복구를 지원합니다. 하나의 API 호출을 하는 것으로 관계형 데이터베이스 인스턴스와 관련된 컴퓨팅 리소스 또는 스토리지 용량을 확장 할 수 있는 유연성 혜택을 누릴 수 있습니다.

또한, 아마존 RDS 다중 AZ 배포는 데이터베이스의 가용성을 높이고 계획되지 않은 서비스 중단으로 부터 데이터베이스를 보호 할 수 있습니다. 단일 데이터베이스에 읽기 작업이 많을 경우 Read Replicas는 읽기 전용 복제본 으로 용량을 확장할 수 있습니다. 또한 모든 아마존 웹 서비스와 같이 선택된 투자자가 없습니다. 단지 사용하는 리소스에 대한 비용만을 지불합니다.

아마존 EC2 인스턴스에서의 관계형 데이터베이스 (RDBMS) 호스팅

관리되는 아마존 RDS 이외에 EC2 인스턴스에 직접 선택한 RDBMS(예: MySQL, Oracle, SQL Server, DB2)를 설치하고 관리할 수 있습니다. 아마존 EC2에서 데이터베이스를 호스팅하는 고객들은 읽기 전용 복제본과 로그 전송 슬레이브를 포함한 마스터/ 슬레이브 와 복제 모델을 다양하게 활용하고 있습니다.

아마존 EC2에서 직접 데이터베이스 소프트웨어를 관리할 경우 내결함성 및 지속성입니다. 이를 위해 네트워크 연결 스토리지 형태의 아마존 Elastic Block Storage (아마존 EBS) 볼륨을 활용하는 것을 권장합니다. 데이터베이스를 실행중인 EC2 인스턴스에 대해서는, 모든 데이터베이스의 데이터와 로그는 아마존 EBS 볼륨에 배치해서 데이터베이스 호스트가 실패하는 경우에도 사용 가능한 상태로 유지되어야 합니다. 이는 호스트의 장애 발생시 새로운 EC2 인스턴스를 생성하고, 기존 아마존 EBS를 새로운 인스턴스에 장착하는 간단한 장애 복구 시나리오로 구성할 수 있습니다. 그리하여 데이터베이스는 원래 있던 지점에서 복구됩니다. 아마존 EBS 볼륨은 단순 디스크 보다 가용성이 높은 가용 영역에 자동으로 중복됩니다. 만일 하나의 EBS 볼륨으로 성능이 충분하지 않다면 스트라이프 볼륨 구성으로 IOPS 성능을 높일 수 있습니다. 부하가 큰 경우, 필요한 IOPS를 명시하여 프로비저닝된 IOPS를 사용할 수 있습니다. 아마존 RDS 서비스는 데이터 관리에만 집중할 수 있도록 스토리지를 대신 관리하여 줍니다.

NoSQL 솔루션

관계형 데이터베이스에 대해 지원하는 것 뿐만 아니라, AWS는 원활하고 확장성을 빠르게, 예측 가능한 성능을 제공하는 완벽하게 관리되는 NoSQL 데이터베이스 서비스인 아마존 DynamoDB를 제공합니다. AWS 관리 콘솔 또는 아마존 DynamoDB의 API를 사용하면 중단 또는 성능 저하없이 위 또는 아래로 용량을 확장 할 수 있습니다. 아마존 DynamoDB가 운영 및 AWS에 분산 된 데이터베이스를 확장 시에 관리 부담을 덜어주기 때문에, 하드웨어 프로비저닝, 설정 및 구성, 복제, 소프트웨어 패치, 또는 클러스터 스케일링에 대해 걱정할 필요가 없습니다.

아마존의 SimpleDB는 고정된 schema 요구 사항없이 쿼리 및 데이터의 색인을 제공하는 경량의 고가용성 및 내결함성 코어 비 관계형 데이터베이스 서비스를 제공합니다. SimpleDB는 크고 높게 인덱싱되며, 유연성이 있는 schema 테이블이 필요한 데이터 액세스 시나리오에서 데이터베이스에 대한 매우 효과적인 대체 될 수 있습니다.

추가적으로, 아마존 EC2는 카산드라, CouchDB와 같은 NoSQL 움직임의 다른 많은 새로운 기술을 호스팅 할 수 있습니다.

스토리지와 데이터 및 자산의 백업

AWS 클라우드에서 저장, 액세스 그리고 웹 애플리케이션 데이터와 자산의 백업에는 많은 옵션이 있습니다. 아마존 Simple Storage Service(아마존 S3)는 고가용성과 객체의 복제 저장을 제공합니다. 아마존 S3는 이미지, 비디오 및 기타 정적 미디어와 같이 정적이거나 변화가 적은 객체들에 좋은 스토리지 솔루션입니다. 아마존 S3는 아마존 CloudFront 서비스와 상호 작용에 의해 엣지 캐싱 및 스트리밍을 지원합니다.

EC2 인스턴스에는 실행중인 EC2 인스턴스에 마운트 가능한 디스크처럼 동작하는 장착형 파일시스템 스토리지인 아마존 EBS 볼륨을 장착할 수 있습니다. 아마존 EBS는 블록 스토리지로 액세스할 필요가 있고, 데이터베이스 파티션 및 응용 프로그램 로그와 같은 실행중인 인스턴스의 수명보다 지속성을 필요로 하는 데이터에 좋습니다.

아마존 EBS 볼륨은 EC2 인스턴스에 독립적이며, 스냅샷을 만들 수 있고, 아마존 S3에 저장될 수 있습니다. EBS 스냅샷은 이전 스냅샷에서 변경된 내용만 백업하므로 스냅샷을 자주할 경우 시간을 줄일 수 있습니다. 또한 여러 EBS 볼륨에 데이터를 복제해 실행중인 다른 인스턴스에 부착하기 위한 기준으로 스냅샷을 사용할 수 있습니다.

단일 EBS 볼륨의 최대 크기는 1TB이며, 여러 EBS 볼륨은 더 큰 볼륨이나 IO 성능 향상을 위해 스트라이핑될 수 있습니다. IO에 의존적인 애플리케이션의 성능을 최대화하려면 프로비저닝된 IOPS 볼륨을 사용할 수 있습니다. 프로비저닝된 IOPS 볼륨은 I/O 집약적인 워크로드, 특히 랜덤 액세스에 I/O 성능이 균일하고 스토리지 성능에 민감한 데이터베이스 워크로드를 만족하도록 설계되었습니다. 볼륨을 만들 때 IOPS 속도를 지정하고 볼륨의 수명을 조절시켜 주는 아마존 EBS 프로비전을 지정합니다. 아마존 EBS는 현재 볼륨 당 1,000 IOPS를 지원합니다. 애플리케이션에서 인스턴스당 수천 IOPS를 지원하기 위해 스트라이프 볼륨을 구성할 수 있습니다.

자동 그룹 확장

AWS의 클라우드 아키텍처와 기존의 호스팅 모델의 주요 차이점 중 하나는 AWS가 트래픽의 동적 변화를 처리하기 위해 필요에 따라 웹 애플리케이션 그룹을 확장 할 수 있다는 것입니다. 기존의 호스팅 모델에서는 일반적으로 예상트래픽에 따라 미리 서버들을 준비해야만 했습니다. AWS에서 인스턴스들은 그룹을 늘리고 줄이기 위해, 일종의 트리거 방식을 통해 프로비전 될 수 있습니다. 아마존 Auto Scaling은 필요에 따라 확장 또는 축소 할 수 있는 서버의 용량 그룹을 만들 수 있습니다. Auto Scaling은 메트릭 데이터를 위해 CloudWatch와 부하 분산을 위한 호스트 추가, 삭제를 ELB와 직접 동작합니다. 예를 들어, 웹 서버가 어떤 기간에 80% 이상의 CPU 사용률을 보고하는 경우 추가적인 웹 서버를 빠르게 배포하고 부하 분산을 위해 ELB에 추가됩니다.

AWS 웹 호스팅 아키텍처 모델에 표현한 바와 같이 각 계층이 독립적으로 확장될 수 있도록 복수의 Auto Scaling 그룹들이 아키텍처의 각 계층에 맞게 생성될 수 있습니다. 예를 들어, CPU 사용률에 따라서 애플리케이션 서버 그룹이 확장될 수 있으며, 네트워크 I/O 변화에 따라 웹 서버 그룹을 확장할 수 있습니다. 최소 및 최대값을 설정하여 24/7 가용성을 보장하고 그룹내 사용량을 제한할 수 있습니다.

Auto Scaling 트리거는 실제 리소스 사용량에 맞추어 특정 계층의 서버 그룹을 늘이거나 줄이도록 설정할 수 있습니다. 그리고 Auto Scaling 서비스는 EC2 API를 이용하여 EC2 인스턴스 그룹을 확장하거나 실행, 종료, 검사할 수 있습니다.

AWS와 장애 조치

기존의 웹 호스팅에 비해 AWS의 또 다른 주요 장점은 다른 배포 지역에 쉽게 액세스할 수 있는 가용 영역입니다. 가용 영역은 다른 가용 영역에서 장애를 극복할 수 있도록 설계되어있는 물리적인 분리된 위치입니다. 같은 지역의 다른 가용 영역에 저렴하고 대기 시간이 짧은 네트워크 연결을 제공합니다. 이 백서의 AWS 웹 호스팅 아키텍처 다이어그램에서 보여주는 데로, 웹 애플리케이션이 내결함성을 만들기 위해 여러 가용 영역을 통해 EC2 호스트를 배포하는 것을 권장합니다. 장애 발생시 다른 가용 영역에 액세스 할 수 있는 단일 포인트를 사용할 수 있는지 확인하는 것도 중요합니다. 예를 들어, 아무리 장애 가능성이 없더라도 데이터의 일관성 및 고 가용성 유지가 되도록, 데이터베이스 슬레이브는 두번째 가용 영역에서 설정해야 합니다.

AWS 클라우드로 기존의 웹 애플리케이션을 이행하는 동안 몇 가지 아키텍처 변경이 요구되지만, AWS 클라우드 사용하는데에는 확장성, 안정성 및 비용 효율성에 상당한 개선이 있기에 가치가 있습니다. 다음 섹션에서, 우리는 이러한 개선들을 논의 할 것입니다.

AWS

웹 호스팅에 주요 고려 사항

AWS 클라우드와 전통적인 웹 애플리케이션 호스팅 모델 사이에는 몇 가지 중요한 차이가 있습니다. 이전 섹션에서는 클라우드에 웹 응용 프로그램을 배포할 때 고려해야 할 주요 영역을 많이 강조했습니다. 이 섹션에서는 클라우드로 응용 프로그램을 이행할 때 고려해야 할 주요 아키텍처의 변화들 일부를 확인합니다.

더 이상 필요없는 물리적 네트워크 장비

AWS에서는 물리적 네트워크 장비를 배포 할 수 없습니다. 예를 들어, 방화벽이나, AWS 애플리케이션을 위한 라우터 및로드 밸런서는 더 이상 물리적 장치에 의존할 수 없으며, 소프트웨어 솔루션으로 교체해야 합니다. 로드 밸런싱을 위해서든 (예. Zeus, HAProxy, nginx, Pound) VPN 연결을 위해서든 (예. OpenVPN, OpenSwan, Vyatta) 다양한 엔터프라이즈 품질의 소프트웨어 솔루션들이 존재하고 있습니다. 이것은 AWS의 제약사항이 아니라 아키텍처의 변화입니다.

모든 곳에 방화벽

예전에 우리가 전통적인 호스팅 모델에 호스트들 사이에서 DMZ를 가지고 통신을 했다면, AWS는 모든 호스트가 잠겨있는 상태에서 안전한 모델을 적용합니다. 호스트 사이의 트래픽 분석은 AWS 배포의 계획 단계 중 하나입니다. 이 분석은 어떤 포트가 개방 될 필요가 있는지에 정확한 결정을 가이드할 것입니다. 아마존 EC2 내에서 보안 그룹은 아키텍처의 각 유형의 호스트에 대해 생성할 수 있으며, 간단하고 계층화 된 보안 모델들이 큰 다양한 아키텍처 내에서 최소량의 액세스를 호스트들 사이에서 되도록 할 수 있습니다.

멀티 데이터 센터

AWS 리전 내 가용 영역은 여러 데이터센터처럼 생각해야 합니다. 다른 가용 영역에서 EC2 인스턴스는 모두 논리적 및 물리적으로 분리하고, 높은 가용성과 안정성 둘 다를 데이터 센터에서 애플리케이션을 배포하기 위해 사용하기 쉬운 모델들로 제공합니다.

호스트를 임시 또는 동적인 것으로 가정

AWS 응용 프로그램을 설계하는데에 가장 중요한 고려 사항은 아마존 EC2 호스트는 임시 및 동적으로 고려되어야 한다는 점일 것입니다. AWS 클라우드에 내장된 모든 응용 프로그램은 호스트가 항상 대기하고 있다고 가정해서는 안되며 EC2 인스턴스가 실패할 경우 아마존 EBS 볼륨에 없는 모든 데이터가 손실된다라는 점을 인식하고 디자인이 되어야합니다. 새로운 호스트가 제기될 때, 호스트의 가용 영역 내에서 IP 주소 또는 위치에 대해서도 어떠한 가정도 만들어져서는 안됩니다. 구성 모델은 유연성이 있어야 하고, 호스트를 부트스트래핑하는 접근점이 클라우드의 동적 특성을 같이 가져가야 합니다. 이러한 기술들은 확장성이 뛰어난 내결함성 응용 프로그램을 구축하고 실행하기 위해 중요합니다.

www.wisen.co.kr



Wisely Combine the Network platforms

WEB APPLICATION HOSTING IN THE AWS CLOUD



서울특별시 구로구 경인로 576 (구로동) [TEL] 02-2630-5795 [FAX] 02-2630-5255