DDoS 대응을 위한 AWS 모범사례

2015 년 6 월

Notices

This document is provided for informational purposes only. It represents AWS's current product offerings and practices as of the date of issue of this document, which are subject to change without notice. Customers are responsible for making their own independent assessment of the information in this document and any use of AWS's products or services, each of which is provided "as is" without warranty of any kind, whether express or implied. This document does not create any warranties, representations, contractual commitments, conditions or assurances from AWS, its affiliates, suppliers or licensors. The responsibilities and liabilities of AWS to its customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.

공지사항

이 문서는 정보제공만을 목적으로 제공됩니다. 문서에 설명된 제품들과 방법들은 이 문서가 공개된 날짜를 기준으로 유효한 내용이며 추후 사전 공지 없이 변경될 수 있습니다. 이 문서에 나와있는 서비스나 설명 및 방법들에 대한 최종 판단은 고객 여러분의 개별적인 판단에 달려있으며, 이를 어떤 방식으로든 보장하지는 않습니다. 이 문서에 대한 내용에 대해서 AWS 에서는 어떠한 보장도 제공하지 않습니다. AWS 와 고객 여러분간의 법적인 책임은 AWS 이용계약(Agreement)에 따르며 이 문서는 AWS 와 고객간의 맺는 이용계약(Agreement)에 해당하지 않으며 어떠한 영향도 주지 않습니다.

원본문서 : AWS Best Practices for DDoS Resilience

https://d0.awsstatic.com/whitepapers/DDoS_White_Paper_June2015.pdf

한글번역 : 김용우 매니저 (Kevkim@Amazon.com) – Solutions Architect, AWS Partner and Alliance Team

Contents

개 요

이 문서는 AWS 를 사용하시는 고객 중 DDoS (Denial of Service)공격에 대비하여 자사 어플리케이션의 가용성을 높이려는 분들을 위해 쓰여졌습니다. 이 문서에서는 DDoS 공격에 대한 일반적인 패턴들을 설명하고, 가용성을 높이는데 도움을 줄 수 있는 기술들과 함께 공격에 대응할 수 있는 레퍼런스 아키텍처를 제공함으로써 보다 잘 준비된 아키텍처를 구성할 수 있도록 도와드립니다.

이 문서는 네트워킹, 보안 및 AWS 에 대한 기본적인 지식과 경험이 있는 IT 실무자, 보안 담당자 들을 대상으로 쓰여졌습니다. 각각의 섹션에 연결된 링크를 따라가면 해당 작업을 어떻게 설정하고 구성해야 하는지에 대한 정보들을 얻으실 수 있으며, AWS 의 Re:Invent 세션 SEC 305 와 SEC307 영상을 통해서도 더 많은 정보들을 얻으실 수 있습니다.

소 개

DDoS 공격의 목적은 최종 사용자(End User)들이 여러분의 웹 사이트나 어플리케이션을 이용할 수 없도록 만드는 것입니다. 이러한 목적을 달성하기 위해 공격자는 다양한 기술들을 사용하는데, 대표적으로 네트워크나 다른 자원들을 고갈시켜 정당한 사용자의 요청을 처리할 수 없게끔 만들어 버립니다. 가장 간단한 형태로는 아래 그림과 같이 단일 공격자가 하나의 호스트만으로 대상에 DoS 공격을 날리는 방식이 있습니다.

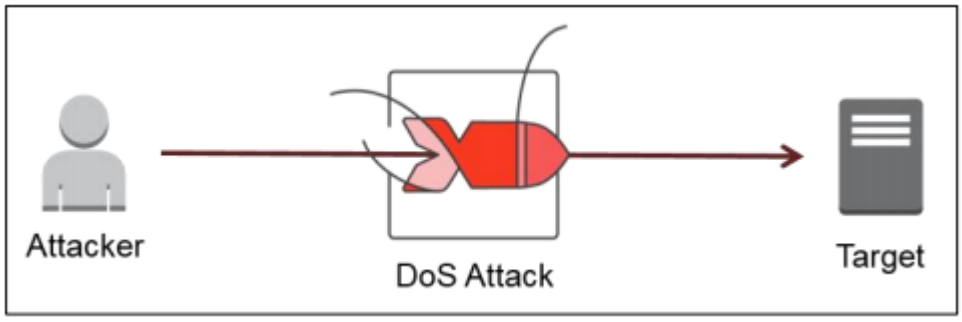

Figure 1. 일반적인 DoS 공격 방식

DDoS 공격 (Distributed Denial of Service Attacks)

DDoS 공격의 경우 공격자들은 여러 대의 호스트를 사용하는데, 이때 해당 호스트들은 공격자를 도와 함께 공격을 수행하는 호스트일 수도 있고 특정 경로를 통해 악성코드에 감염되어 공격자에게 조종당하는 좀비 호스트일 수도 있습니다. 아래 그림은 일반적인 DDoS 공격을 도식화 한 것으로, 각각의 공범 혹은 좀비 호스트들이 목표 호스트를 상대로 대규모의 패킷 전송 혹은 연결 요청을 보냄으로써 해당 호스트의 자원을 고갈시키려는 시도를 합니다.

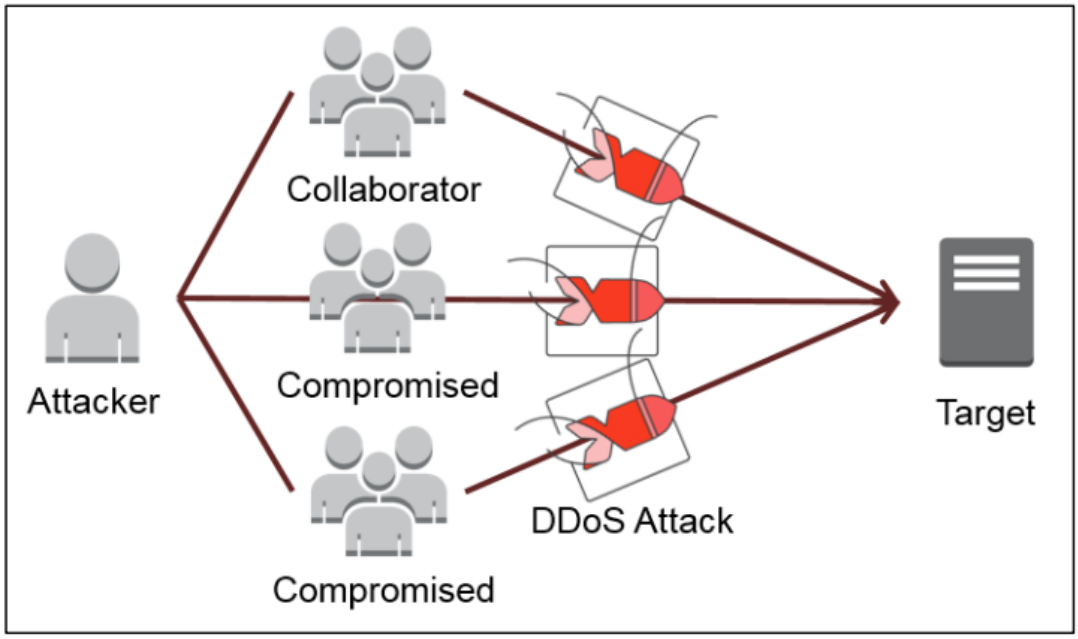

Figure 2. 일반적인 DDoS 공격 방식

공격자들이 대상 호스트의 자원을 고갈시키는 방법은 여러가지가 있습니다. 예를 들어, 공격자가 리플렉션(Reflection) 이나 증폭(Amplification) 공격 방법을 병행하거나 다수의 봇넷(Botnet)을 사용함으로써 다량의 패킷을 사용해 공격에 이용하는 방법이 있습니다. 리플렉션(Reflection)공격은 패킷의 출발지(소스) IP 를 대상 Host 의 IP 로 변조해 좀비 호스트들에게 요청 패킷을 전송하는 방식으로 해당 요청에 대한 응답이 대상 서버로 몰리도록 하는 방법입니다.

증폭(Amplification) 공격은 작은 패킷이나 요청들을 보내고 큰 규모의 응답을 유도하는 방식입니다. 이 공격에 많이 사용되는 프로토콜로는 DNS, NTP 및 SSDP 등이 있는데, 프로토콜에 따라서 요청 패킷의

크기 대비 최대 수십 배 크기의 응답이 전달될 수 있습니다. DNS 를 예로 들면, 64byte 크기의 요청을 통해 3,456byte 의 불필요한 트래픽이 생성되도록 할 수 있습니다.

리플렉션과 증폭 공격이 함께 사용되면 좀비 호스트들에 작은 규모의 패킷 전송으로 해당 호스트들에서 대규모의 요청/응답을 대상 호스트로 전송하도록 만들 수 있습니다.

다음 그림은 공격자가 작은 양의 트래픽으로 큰 규모의 트래픽을 대상호스트로 보내는 방법을 설명하고 있습니다.

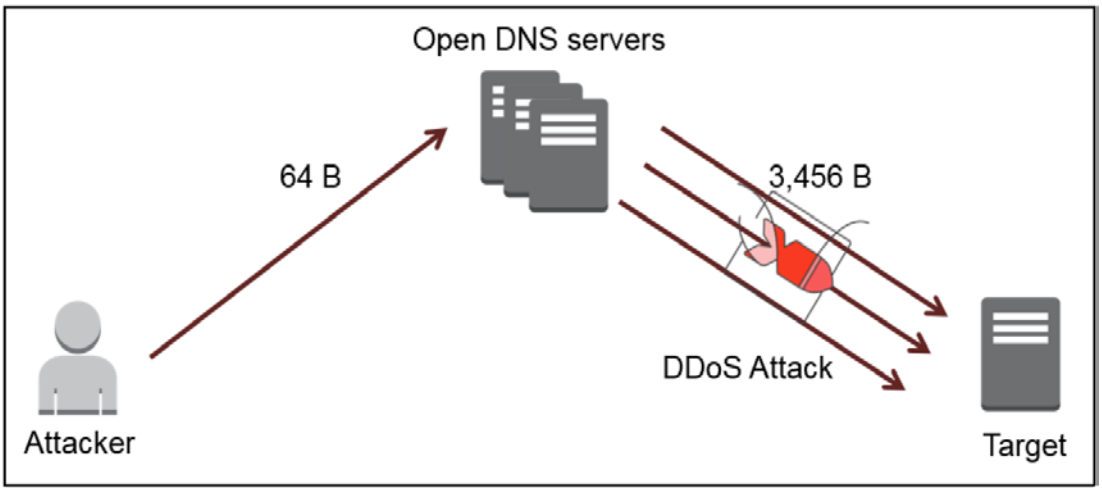

Figure 3. DDoS 플러딩, 리플리케이션 및 증폭(Amplification) 공격

원치 않는 트래픽이 대상 호스트에 도달하는 과정에서 해당 트래픽은 다양한 계층의 네트워크 장비들을 거쳐 호스트의 운영체제를 지나 어플리케이션 레벨까지 도달하게 되며, 이 과정을 거치는 중 어느 계층 에서든 자원의 고갈로 인해 트래픽을 처리하지 못하게 되면 서비스의 장애가 발생하게 됩니다. 소모되는 자원에 따라서, 해당 공격은 대역폭 공격으로 분류되기도 하고(예. UDP Floods), 프로토콜 공격(예, SYN Flood) 또는 어플리케이션 공격으로 구분되기도 합니다. (예. HTTP GET/POST floods).

어떤 유형의 공격이든 DDoS 공격의 목적은 서비스의 가용성 문제를 발생시켜 정당한 사용자들이 해당 서비스를 사용하지 못하도록 만드는 것입니다. AWS 에서는 DDoS 공격이 야기하는 가용성 문제에 대해 사전에 대응할 수 있는 여러 서비스 및 기능들을 제공하고 있습니다.

DDoS 공격에 대응할 수 있는 각각의 서비스들이 제공하는 안정성 및 가용성의 정도는 해당 서비스를 어떻게 구성하고 사용했는지에 따라 다를 수 있기 때문에, 이 문서에서 설명되는 기술 및 방법들이 특정

수준의 가용성을 보장하지는 않습니다. 이 문서가 설명하는 기술 및 방법을 적용할 때는 AWS 비용이 사용량에 기반한다는 것을 고려하시길 권합니다.

DDoS 공격에 대한 완화 방안들

전통적 IT 인프라 환경에서 DDoS 에 대한 보안은 트래픽 필터링이나 특정 장애물(기타 보안장비)을 설치하여 공격자로 하여금 해당 공격이 성공하지 못하도록 막는 데 집중되었습니다. AWS 기반 환경에서는 이러한 필터링 및 블록킹 기술들에 더해 변화하는 상황에 따라 유연하게 확장될 수 있는 아키텍처를 추가하실 수 있습니다.

이 문서에서는 DDoS 공격에 대한 취약점을 최대한 줄일 수 있는 다섯 가지 기술들을 설명합니다.

- **공격 받을 수 있는 포인트의 최소화**
- **공격을 흡수할 수 있는 확장 가능한 아키텍처 설계**
- **노출된 자원들에 대한 보안 강화**
- **정상적인 트래픽(사용) 패턴을 숙지**
- **공격에 대한 대응 계획 수립**

이 문서에서는 아래의 레퍼런스 아키텍처를 통해 위에 열거된 기술들을 설명합니다. 아래의 아키텍처는 AWS 를 활용해 고가용성 아키텍처를 구성하는 방법을 나타내고 있습니다. AWS 아키텍처 센터에서 제공하는 가이드라인을 따라 이와 유사한 아키텍처를 쉽게 구성하실 수 있습니다.

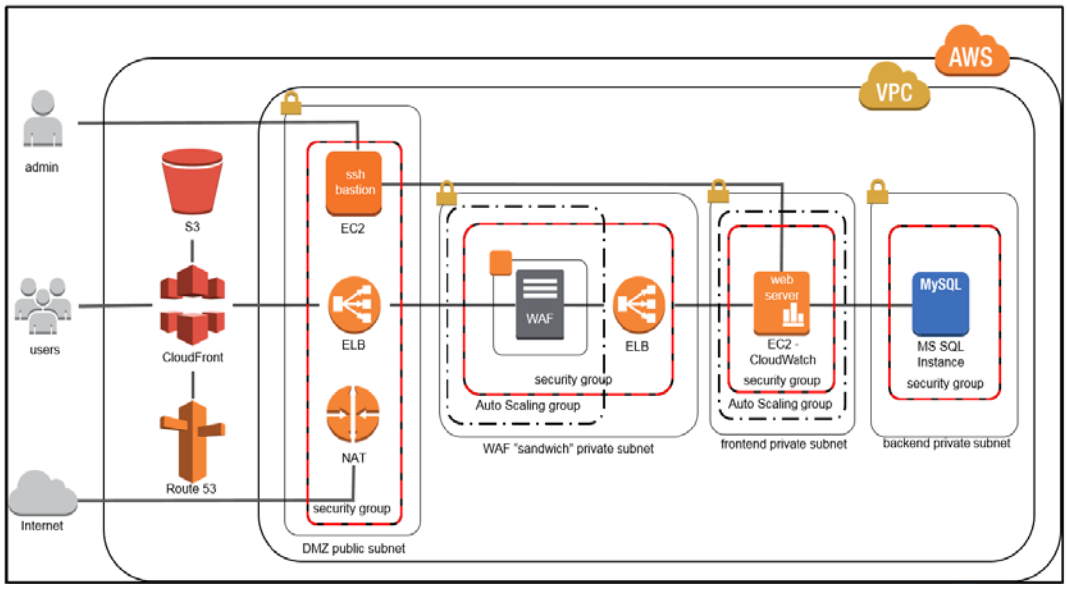

Figure 4. DDoS 에 대응하는 레퍼런스 아키텍처

공격받을 수 있는 포인트의 최소화 (Minimize the Attack Surface Area)

공격을 받을 수 있는 포인트를 최소화한다는 것은 공격자로 하여금 해당 서비스 또는 어플리케이션에 접근할 수 있는 인터넷의 접근 포인트를 최소화하는 것을 뜻합니다. 이를 위한 방법으로 **a)** 어플리케이션에 접근할 수 있는 인터넷 접근 포인트 최소화 **b)** 서비스에 직접 관련이 없는 인터넷 포인트 제거 **c)** 사용자 트래픽과 관리 트래픽에 대한 분리 **d)** 어플리케이션 혹은 서비스의 인터넷 접속 포인트를 정당하지 않은 사용자/트래픽이 쉽게 접근할 수 없도록 추상화(모호하게 만들어 줌) **e)** 인터넷 접속 포인트를 분리하여 공격에 대한 영향을 최소화 등을 고려할 수 있습니다. 이와 같은 방법들은 Amazon VPC(Virtual Private Cloud)를 통해 구현이 가능합니다.

Amazon Virtual Private Cloud

Amazon VPC(Virtual Private Cloud)는 여러분의 AWS 계정에서 직접 정의하고 생성하실 수 있는 논리적으로 완전히 분리된 가상의 네트워크로써, 라우팅 테이블 정의, 네트워크 게이트웨이(인터넷/VPN) 생성 및 보안 설정 등을 통해 EC2 인스턴스와 같은 자원들을 안전하게 구성 및 운영할 수 있는 환경을 제공합니다.

VPC 사용으로 얻게 되는 더 큰 장점은 여러분의 EC2 인스턴스가 인터넷으로부터 완전히 감춰져 오직 여러분이 정의한 퍼블릭 서브넷 상에서만 접근이 가능하도록 설정 하실 수 있는 것입니다. 또한 프라이빗 서브넷은 내부 네트워크로써 내부의 사설 IP 통신만 가능하도록 구성하실 수도 있습니다. 보안 그룹(Security Group)과 네트워크 접근제어(NACL) 기능을 통해 이런 환경을 설정할 수 있습니다. 보안 그룹은 EC2 인스턴스 레벨에서 설정하실 수 있는 IP/Port 기반의 방화벽 기능이고, 네트워크 ACL 은 서브넷과 연결되는 방화벽 기능입니다. 여러분은 보안 그룹(Security Group)을 EC2 인스턴스에 적용하여 보안의 첫 번째 계층을 구성하실 수 있고 추가로 서브넷에 Network ACL 을 적용하여 두 번째 계층에 대한 보안까지 적용하실 수 있습니다. 보안 그룹과 네트워크 ACL 에 대한 보다 자세한 설명은 다음을 참하시기 바랍니다. (VPC 에서의 보안)

다음 링크들을 통해 Amazon 관리 콘솔에서 VPC 를 구성하는 법을 확인하실 수 있습니다. 만약 Step 3 에서 어떤 인스턴스를 골라야 할지 잘 모르시겠다면 다음 단계로 진행하시기 전에 EC2 에 대한 기본적 내용을 먼저 학습하시기 바랍니다.

Step 1: VPC 및 인터넷 게이트웨이 설정

Step 2: VPC 에 보안 그룹 설정

Step 3: VPC 에서 인스턴스 시작

Step 4: 인스턴스에 엘라스틱 IP 주소 할당

Step 5: 네트워크 ACL 구성

또한, 아래와 같이 사전에 준비된 시나리오들을 활용하여 원하는 VPC 환경을 손쉽게 구성할 수 있습니다.

Scenario 1: 퍼블릭 서브넷 한 개를 가진 VPC

Scenario 2: 퍼블릭 서브넷과 프라이빗 서브넷을 가진 VPC

Scenario 3: 퍼블릭 서브넷, 프라이빗 서브넷에 H/W VPN 연결을 가진 VPC

Scenario 4: 프라이빗 서브넷에 H/W VPN 연결을 가진 VPC

위의 단계들을 잘 수행하셨다면 이제 VPC 내에 생성된 EC2 인스턴스에 접속 하실 수 있습니다. Linux 인스턴스에 접속하는 방법은 다음을 참고하시고, Windows 인스턴스에 접속하는 방법은 다음을 참고하시기 바랍니다.

공격을 흡수할 수 있는 확장성 있는 아키텍처 설계 (Be Ready to Scale and Absorb the Attack)

DDoS 공격에서 가장 중요한 것은 규모(볼륨)입니다. 대부분의 공격자들은 그들의 목적을 달성하기 위해 어플리케이션이 수용할 수 없는 레벨의 트래픽을 전송합니다. 해당 공격들의 규모를 넘어서는 확장성 있는 서비스 아키텍처를 구성해서 공격자로 하여금 더 많은 시간과 노력을 필요로 하게 만든다면 효과적인 방어 수단이 될 수 있습니다.

AWS 에서 사용할 수 있는 확장 방법은 두 가지로, 하나는 수평적 확장(Scale-Out)이고 다른 하나는 수직적 확장(Scale-Up)입니다. 수평확장은 여러분의 인프라에 인스턴스 및 서비스를 추가하는 것을 의미하며, 수직확장은 인스턴스 선택 시 CPU, 메모리 등의 용량이 높은 타입을 선택하는 것을 말합니다. 이 두 가지 확장 방법들을 사용하면 DDoS 공격에 대비해 다음의 4 가지 이점을 얻으실 수 있습니다.

- **공격이 더 넓은 지역으로 확대되어 단일 포인트에 과부화가 발생하는 것을 방지함**
- **공격자들이 확장되는 서비스 인프라에 맞춰 더 많은 자원과 시간을 소비해야 함**
- **확장을 통해 시간을 벌고 DDoS 공격의 패턴을 분석해 적절한 방어조치를 취할 수 있음**
- **확장성 있는 아키텍처를 통해 다른 장애 시나리오에도 선제적 대응가능**

DDoS 의 관점에서 AWS 의 확장 기능(Scaling)의 이점을 얻는 데는 다음의 3 가지 방법이 있습니다. **(1)** 여러분의 어플리케이션에 가장 적합한 인스턴스를 선택하세요. **(2)** ELB 를 서비스에 활용하시고 Auto scaling 그룹을 설정해서 자동으로 확장되도록 구성하세요. **(3)** Amazon CloudFront 나 Route 53 처럼 확장을 염두에 두고 설계된 글로벌 서비스를 활용하세요.

Amazon Elastic Compute Cloud

Instance Types

Amazon EC2 (Elastic Compute Cloud)는 클라우드에서 확장성있는 컴퓨팅 용량을 제공하고, 하드웨어에 대한 초기 투자비용을 없애드립니다. EC2 를 사용하면 인스턴스라고 불리는 가상 서버를 구동할 수 있으며, 트래픽에 따라 해당 인스턴스를 스케일 업 혹은 다운 할 수 있습니다. 인스턴스를 구동할 때 여러분이 선택하는 인스턴스 타입에 따라 컴퓨팅 CPU, 메모리, 디스크 공간 등이 결정되며, 어플리케이션에 맞는 가장 적절한 타입을 선택하실 수 있습니다. 어플리케이션의 가용성을 최대한으로 높이기 위해서는 해당 어플리케이션의 컴퓨팅 요구사양 뿐만 아니라 갑작스런 트래픽 폭주에도 대응할 수 있는 인스턴스의 선택이 필요합니다. 일부 EC2 인스턴스의 경우 비용 최적화에 초점이 맞추어져 있어 해당 인스턴스에 할당된 네트워크 대역폭에 제한이 있는 경우도 있으며, 이 경우 DDoS 와 같은 공격에 좀더 큰 영향을 받을 수 있습니다. 여러분의 서비스 인프라 구성에서 중요한 어플리케이션의 경우에는 EBS 최적화 인스턴스 또는 10G 네트워크 대역폭을 제공하는 인스턴스 타입을 선택하시는 것이 좋습니다. 더 자세한 정보는 이 링크를 통해 확인하실 수 있습니다.

Enhanced Networking

C3, C4, R3, D2 그리고 I2 인스턴스 타입을 선택하시면 향상된 네트워크 성능(PPS)을 지원하는 Enhanced Networking 기능을 사용하실 수 있습니다. 이 기능은 네트워크 가상화 스택을 사용해 낮은 CPU 점유율로 높은 I/O 성능을 제공해 드리는 것으로, 이러한 기능을 통해 높은 어플리케이션 처리성능, 낮은 지연의 네트워킹 등으로 DDoS 공격에 대응하여 가용성을 높일 수 있습니다.

Enhanced Networking 기능을 활용하려면 인스턴스 선택 시 해당 기능에 필요한 HVM(Hardware Assisted Virtual Machine) 및 SR-IOV 드라이버를 제공하는 타입을 선택하셔야 합니다. Amazon HVM Linux AMI 는 기본적으로 SR-IOV 드라이버를 포함하고 있으며, SR-IOV 를 기본적으로 제공하지 않는 AMI 의 경우 아래의 링크에서 필요한 드라이버를 다운로드해 설치하실 수 있습니다.

Amazon 리눅스에서 향상된 네트워킹 기능 사용

Ubuntu 에서 향상된 네트워킹 기능 사용

다른 리눅스 배포판에서 향상된 네트워킹 기능 사용

Windows 인스턴스에서 향상된 네트워킹 기능 사용

Elastic Load Balancing

Amazon Elastic Load Balancing 을 사용하면 트래픽이 여러 곳의 가용 영역(AZ)에 걸쳐있는 여러 대의 EC2 인스턴스로 분산되도록 구성하실 수 있으며, 이를 통해 단일 인스턴스로만 구성되었을 경우 발생할 수 있는 트래픽 과부하를 최소화하실 수 있습니다. ELB 를 사용하면 기존 서비스의 중단 없이 필요에 따라 손쉽게 EC2 인스턴스를 제거하거나 추가하실 수 있습니다. 예를 들어 만약 하나의 EC2 인스턴스에 장애가 발생하면, ELB 는 자동으로 나머지 EC2 인스턴스들로 트래픽의 경로를 변경합니다. 이후 해당 EC2 인스턴스가 복구되면 ELB 는 해당 인스턴스로 다시 트래픽을 보냅니다.

ELB 는 또한 단일 관리 포인트를 제공하며, 네트워크로 들어오는 공격을 방어할 수 있는 첫 번째 관문으로써의 역할을 수행합니다. 여러분은 모든 EC2 인스턴스를 ELB 뒷단에 위치시키고 ELB 만을 인터넷에 노출시켜 표면적으로 드러나는 공격 포인트를 최소화하실 수 있습니다. 또한 ELB 에서 직접 인스턴스를 지정해 확장 또는 축소에 대한 정책을 적용하실 수 있으며, 이를 통해 개별 인트턴스를 관리하는 수고를 줄일 수 있습니다. VPC 와 함께 ELB 를 병용할 경우 보안 그룹(Security Group)이나 Network ACL 등의 적용으로 추가적인 보안 계층을 구성하실 수 있습니다. ELB 를 사용함으로써 얻게 되는 가장 큰 장점 중 하나는, ELB 는 오직 유효한 TCP 요청만을 지원하기 때문에 UDP 공격이나 SYN Flood 공격 등은 여러분의 EC2 인스턴스에 도달할 수 없다는 것입니다.

다음 섹션에서는 두 가지 기본적인 로드 밸런서 타입인 외부(퍼블릭) 로드 밸런서와 내부 네트워크용 로드 밸런서를 설정하는 방법에 대해 설명합니다. 이 두 가지 로드 밸런서들은 Layer 7 계층 보호를 위해 필요하며 이후에 설명되는 Web Application Firewall 섹션에서 좀더 자세히 설명될 예정입니다.

Step 1: Create a Basic Load Balancer in Default VPC

Step 2: Create a Basic Internal Load Balancer in Amazon VPC

Auto Scaling

오토 스케일링은 사용자가 설정해 놓은 조건에 따라 EC2 인스턴스의 용량을 높이거나 줄일 수 있도록 구성할 수 있는 기능으로, 어플리케이션의 가용성을 높이는데 도움을 줍니다. 예를 들어 네트워크 트래픽의 증가에 따라 인스턴스들이 점진적으로 추가되도록 오토 스케일링 그룹에 설정하시면 실제 트래픽의 증가에 따라 미리 설정된 인스턴스들이 추가로 생성되어 해당 트래픽의 증가에 대응할 수 있습니다. 이와 반대로 (예를 들어 DDoS 공격이 종료되며) 트래픽이 점진적으로 감소하면 추가되었던 인스턴스들이 종료되도록 설정하실 수도 있습니다. Amazon CloudWatch 를 활용해 이러한 확장/축소 작업을 동작시킬 수 있으며, ELB 를 활용해 오토 스케일링 그룹 내의 인스턴스들에게 트래픽을 배분할 수 있습니다.

오토 스케일링 그룹을 처음으로 실제 프로덕션 환경에 추가하기 전에 아래와 같은 사항들을 고려해야 합니다. 시작하시기 전에 AWS Cloud 에서 운용되는 여러분의 어플리케이션에 대한 특성을 잘 확인하시고 다음 사항들에 대해 준비하시기 바랍니다.

- 여러분의 서버를 구성하고 실제 서비스가 시작될 수 있는 준비 상태까지 얼마나 시간이 소요됩니까? 만약 5 분 이상 시간이 걸릴 경우 서비스 시작 시점부터 여러 대의 EC2 인스턴스를 사용하시거나 스케일링 조건의 임계치를 충분히 낮추실 것을 권합니다.
- 어떤 메트릭이 여러분의 어플리케이션 성능에 가장 높은 영향을 미칩니까? DDoS 공격과 연관된 메트릭으로는 CPUUtilization, NetworkIn 그리고 StatusCheckFailed 등이 있습니다.
- 현재 가지고 있는 자원 중 어떤 것을 오토 스케일링 그룹을 통해 사용하고 싶으신가요? 공격을 받을 경우 아마 현재 서비스중인 인스턴스와 같은 종류, 또는 그 이상의 용량을 제공하는 인스턴스를 오토 스케일링 그룹에 사용하시고자 할 것입니다.
- 여러분의 오토 스케일링 그룹이 몇 개의 가용 영역(AZ)에 걸쳐 확장되시기를 원하시나요? AWS 는 최소 두 개 이상을 권고합니다.
- 얼마나 빠르게 인프라가 스케일 업 또는 다운되어야 하나요? DDoS 공격은 연속으로 시행될 수 있다는 것을 명심하시기 바랍니다. 아마도 첫 번째 공격이 끝나자마자 스케일 다운 후 두 번째 공격에 바로 다시 스케일 업을 수행하시고 싶진 않으실 거라 생각합니다.
- 오토 스케일링 그룹을 사용하여 확장 시 최대 얼마까지를 예산범위로 책정하고 계시는지요?

인스턴스를 추가한다는 것은 추가적인 과금이 발생한다는 것을 의미합니다. 오토 스케일링 그룹을 만들 때 해당 그룹이 최대 몇 대까지의 인스턴스를 가질 수 있는지를 정하실 수 있습니다. 또한 CloudWatch 를 통해 해당 그룹의 인스턴스 수가 최대 규모에 도달했을 경우 알림을 발생시키도록 설정 하실 수 있습니다. 알림을 설정하는 방법은 Amazon CloudWatch 부분에서 보실 수 있습니다.

자신의 인프라와 어플리케이션에 대한 이해도가 높으면 높을수록 더 효율적으로 오토 스케일링 그룹을 구성하고 활용하실 수 있습니다. 충분한 정보를 모으셨다고 생각하신다면 아래의 단계에 따라 오토 스케일링 그룹을 만드실 수 있습니다.

Step 1: 시작 구성 생성 (Launch Configuration)

Step 2: 오토 스케일링 그룹 생성

Step 3: 오토 스케일링 그룹 확인

Amazon CloudFront

Amazon CloudFront 는 컨텐츠 전송 네트워크(CDN)로써 다른 Amazon 서비스들과 통합되어 최종 사용자에게 더욱 쉽고 빠르게 컨텐츠를 전달해 주는 역할을 수행합니다. CDN 은 여러분의 원본(Origin)과 최종 사용자 사이에 위치하며 프록시 계층과 같은 역할을 합니다. CDN 은 컨텐츠를 캐싱하고 여러 개의 PoP(Point of Presence)에서 최종 사이트까지의 접속 경로들을 최적화해 전체적인 성능을 향상시켜 줍니다. CloudFront 를 사용하면 개별 트래픽 요청들이 여러분의 원본(Orgin)으로 가는 대신 복수의 PoP 으로 분산되므로 최종 사용자에게 보다 빠른 응답을 제공할 수 있습니다.

Amazon CloudFront 는 여러 개의 PoP 을 사용함으로써 DDoS 공격중 인프라 계층에 대한 공격과 일부 어플리케이션 계층의 공격을 여러 지역으로 분산시킴으로써 공격으로 인한 영향을 줄일 수 있습니다. AWS 는 각각의 CloudFront PoP 이 위치한 지역에서 높은 용량과 가용성 확보를 위하여 여러 개의 인터넷 연결을 제공하며 이로 인해 CloudFront 는 공격 트래픽에 대응하면서도 정상적인 사용자들의 요청까지 처리할 수 있습니다.

Amazon CloudFront 는 필터링 기능도 가지고 있어서 유효한 TCP 연결과 HTTP 요청만을 허용하고 그렇지 않은 요청들은 거부해 버립니다. 이 기능을 통해 원본(Origin)에서 유효하지 않은 트래픽 처리(UDP Flood, SYN Flood 및 Slow Read 등)에 따르는 부담을 줄일 수 있습니다.

Amazon CloudFront 를 사용하기 위해서는 배포(Distribution)를 생성하고 여러분의 원본(Origin)을 지정해 주어야 하며, 이는 EC2, S3 Bucket, ELB 또는 웹서버 등이 될 수 있습니다. 배포(Distribution)를 설정해주고 나면 CloudFront 는 사용자의 요청에 응답을 시작하고 성능을 향상하기 위해서 컨텐츠를 캐싱할 것입니다.

다음의 단계들을 통해 Amazon CloudFront 를 설정하시는 법을 익히실 수 있습니다. 또한 CloudFront 를 활용해 다수의 배포(Distribution)를 생성하기 위해서는 다음의 문서를 참고하시기 바랍니다.

Step 1: Create a Web Distribution

Step 2: Test Your Web Distribution

Amazon Route 53

DNS 서버는 DDoS 공격에 있어서 가장 대표적인 타겟 중 하나입니다. DNS 서버는 도메인 이름을 IP 로 변환해주는 역할을 수행하는데 공격자들이 이러한 DNS 서버를 단일 장애 포인트(SPOF)로 인식하기 때문입니다. 예를 들어 여러분의 어플리케이션이 실제로는 정상 동작하고 있다고 해도 DNS 에 장애가 생기면 사용자들이 해당 어플리케이션을 제대로 사용하기 어려워집니다. 이러한 점 때문에 DNS 를 공격하면 여러분의 어플리케이션을 효과적으로 이용 불능 상태로 만들 수 있습니다. DNS 는 UDP 를 사용하고, 쿼리에 대한 응답을 주는 데 사용되기 때문에 리플리케이션(Replication) 공격과 증폭(Amplification)공격에 특히 취약하며, 이 때문에 DNS 의 가용성을 높이기 위해서는 많은 추가 자원들이 필요하게 됩니다.

Amazon Rotue53 은 고가용성 및 확장성을 제공하는 DNS 서비스로써 사용자들을 AWS 내부 혹은 외부의 인프라로 라우팅 해주는 역할을 수행합니다. Route 53 은 지연 기반 라우팅(Latency Based Routing), 위치기반 DNS(Geo DNS), 가중치 기반 트래픽 분배(Weighted Round Robin)등의 기능을 통해

글로벌 차원에서 서비스의 트래픽을 관리해 줍니다. 이러한 라우팅 방식들은 Route 53 의 페일오버 기능과 함께 낮은 지연 및 장애에 잘 대응할 수 있는 아키텍처를 구성할 수 있도록 해 줍니다.

Amazon Route 53 에는 여러분의 어플리케이션이 DDoS 의 공격을 받는 도중에도 여전히 사용자들에게 서비스를 수행할 수 있도록 해 주는 두 가지 기능이 있는데, 그 중 하나는 셔플 샤딩(Shuffle Sharding) 이고 다른 하나는 애니캐스트 라우팅(Anycast Routing)입니다.

Shuffle Sharding

셔플 샤딩(Shuffle Sharding)은 데이터 베이스에서 사용되는 샤딩과 비슷한 개념으로, 샤딩은 수평적인 데이터의 파티션들이 여러 대의 데이터베이스 서버로 분산되어 처리됨으로써 이중화 기능을 제공하는 것을 뜻합니다. 이와 유사하게 Amazon Route 53 은 셔플 샤딩을 활용해 여러 곳의 PoP 으로 DNS 요청을 나누어 보냄으로써 여러분의 어플리케이션까지 도달하는 복수의 경로를 제공해 줍니다.

Anycast Routing

애니캐스트 라우팅(Anycast Routing)은 여러 곳의 PoP 에서 같은 IP 주소를 사용함으로써 서비스의 가용성을 높여줍니다. 만약 DDoS 공격이 어느 하나의 PoP 을 공격하여 무력화 시킬지라도, 셔플 샤딩을 통해 해당 장애가 한 곳으로만 국한되며 나머지 경로들을 통해 여러분의 인프라까지 이상없이 서비스할 수 있습니다.

Amazon Route 53 은 또한 헬스체크를 통한 DNS 페일오버 기능을 제공함으로써 DNS 쿼리를 통해 정상적인 자원으로만 라우팅 되도록 해 줍니다. 예를 들어 Example.com 이라는 사이트가 10 개의 인스턴스들에서 호스팅되고 이 중 두개는 다른 가용 영역(AZ)에서 서비스된다고 가정해 보겠습니다. 이때 여러분은 Route 53 의 헬스체크 기능을 설정하여 모든 인스턴스들의 동작상태를 모니터링하고 DNS 를 통해 들어오는 쿼리가 오직 정상적인 인스턴스로만 라우팅 되도록 구성할 수 있습니다. 또한 Route 53 의 Alias, 가중치 기반, 지연율 또는 위치기반 등의 다양한 방법을 통해 페일오버(Fail-over)를 설정하실 수 있습니다.

Amazon Route 53 DNS 서비스를 사용하는 방법으로는 여러분이 보유하고 있는 기존 도메인을 Route 53 으로 이전하시는 방법과 Route53 를 통해 신규 도메인 이름을 등록하시는 방법이 있습니다. 이번

섹션에서는 Amazon CloudFront 의 배포(Distribution)를 Route 53 의 ALIAS 레코드 셋으로 사용할 것입니다. AWS 리소스로 쿼리를 라우팅하는 법에 나열된 단계에 따라 Route 53 에 들어오는 트래픽이 다른 서비스들로 라우팅되도록 설정하실 수 있습니다.

Step 1: 도메인 이름을 등록하고 Route 53 을 DNS 서비스로 설정하기

Step 2: CloudFront 배포(Distribution)로 쿼리 라우팅하기 (Public Hosted Zone 에만 적용 가능)

Step 3: 헬스체크 및 DNS 페일오버 설정

Route 53 에 대한 추가적인 정보를 얻으시려면 다음 링크들을 참고하시기 바랍니다.

· 기존 DNS 서비스에서 Amazon Route 53 으로 마이그레이션하는 방법

· 루트 도메인은 그대로 둔채 Route 53 을 사용하는 서브 도메인을 생성하는 방법

· 루트 도메인은 그대로 둔채 기존의 서브 도메인을 Rotue 53 으로 마이그레이션하는방법

노출된 자원들에 대한 보안강화 (Safeguard Exposed Resources)

여러분의 어플리케이션으로 연결되는 인터넷 접속 지점들을 줄일 수 없는 상황이라면 그 지점들에 대해 추가적인 조치들을 취해 정상적인 트래픽 이외의 모든 접근들을 차단함으로써 해당 지점들을 보호할 필요가 있습니다. Amazon CloudFront, Route 53 그리고 웹 어플리케이션 방화벽(WAF)을 사용하면 이러한 조치를 취하는 데 도움이 됩니다.

Amazon CloudFront

Amazon CloudFront 는 컨텐츠로의 접근을 제한하는 두 가지 방법을 제공하는데, 그 중 한 가지는 위치 기반 접근제한(Geo Restriction)이고 나머지 하나는 OAI(Origin Access Identity)입니다.

Geo Restriction

Amazon CloudFront 는 위치기반 접근 제한을 지원하는데, 이는 여러분의 컨텐츠로 접근하는 사용자의 지리적 위치를 확인해 접근을 제한하는 기능입니다. 사용자가 여러분의 컨텐츠를 요청하면 일반적으로 CloudFront 는 아무 제한 없이 해당 사용자에게 관련 컨텐츠를 전송해 줍니다. 하지만 해당 사용자가 여러분이 통상적으로 서비스를 제공하지 않는 일부 국가들에 속해 있다면 해당 국가들로부터 들어오는 접근을 제어할 수 있습니다. 위치기반 접근제한(Geo Restriction)이 지원하는 시나리오는 다음과 같습니다.

- 인가된 국가들로부터 들어오는 컨텐츠에 대한 요청은 모두 허가한다.
- 인가되지 않은(Blacklisted) 국가에서 들어오는 컨텐츠에 대한 요청은 모두 거부한다.

위치기반 접근제한(Geo Restriction)을 사용하기 위해서는 CloudFront 에서 기본적으로 제공하는 서비스를 사용하실 수도 있고, 일반적인 국가 단위보다 더 세밀한 제한을 설정할 수 있는 다른 3rd Party 솔루션을 사용하실 수도 있습니다. CloudFront 에서 제공하는 위치기반 접근제한(Geo Restriction)을 사용하시려면 콘텐츠에 위치기반 접근제한 거는 법을 참고하시어 설정하시기 바랍니다. 다른 솔루션 사용에 관해서는 3rd Party 위치기반 서비스 사용하는 법을 참고해 주세요.

OAI (Origin Access Identity)

통상적으로 S3 버킷을 CloudFront 의 오리진으로 사용할 때는 해당 버킷의 모든 컨텐츠에 대해서 모두에게 읽을 수 있는 권한을 부여합니다. 이렇게 하면 누구든지 S3 의 URL 혹은 CloudFront 의 URL 을 통해 해당 컨텐츠에 마음껏 접근할 수 있습니다. CloudFront 는 기본적으로 S3 의 URL 을 드러내진 않지만 최종 어플리케이션에서 S3 버킷의 컨텐츠를 직접 사용할 경우 공격자가 해당 URL 을 쉽게 알아낼 수 있습니다.

S3 에 대한 직접적인 접근을 제어할 수 있는 방법으로는 OAI(Origin Access Identity)가 있는데, 이는 여러분이 생성하는 CloudFront 특별 사용자라고 생각하시면 됩니다. OAI 에게 CloudFront 를 통해 S3 에 접근할 수 있는 권한을 주고 S3 에 설정된 나머지 권한들은 모두 삭제합니다. 이후 사용자들이 CloudFront 를 통해 S3 에 있는 컨텐츠를 요청하면 OAI 가 대신해서 해당 컨텐츠를 가져옵니다. 만약 사용자들이 S3 URL 을 통해 직접 컨텐츠에 접근하려고 하면 해당 요청들은 모두 거부됩니다.

Step 1: CloudFront OAI 를 생성하고 배포(Distribution)에 추가하기

Step 2: OAI 에게 S3 버킷의 객체들을 읽을 수 있는 권한 주기

Step 3: S3 버킷의 권한 수정

Amazon Route53

Amazon Route 53 은 DDoS 공격에 대응하여 여러분의 인프라가 손쉽게 확장할 수 있도록 도와주는 두 가지 기능들을 제공하고 있는데, 하나는 Alias Record Set 이고 다른 하나는 Private DNS 입니다.

Alias Record Sets

일반적인 Route53 의 레코드 세트와는 다르게, Alias 레코드 세트는 Route 53 에 특화된 확장자를 지원해 DNS 기능을 제공합니다. IP 주소나 도메인 이름대신 Alias 레코드 세트가 CloudFront 배포(Distribution), ELB 로드밸런서, S3 버킷 또는 같은 Hosted Zone 내의 다른 Route 53 레코드 세트를 가리킬 수 있습니다.

Alias 레코드 세트는 여러분이 공격받고 있을 때 필요한 조치를 취하는데 필요한 시간을 절약해 줄 수 있을 뿐만 아니라 이러한 조치에 유용한 방법들도 제공합니다. 예를 들어, example.com 의 Alias 레코드 세트가 몇 대의 EC2 인스턴스가 연결된 ELB 로드밸런서를 가리키고 있다고 가정해 보겠습니다. 만약 여러분의 어플리케이션이 공격받는 상황이 생기면 재빠르게 Alias 레코드 세트를 CloudFront 배포(Distribution)로 변경하거나 또는 웹 방화벽을 비롯한 각종 보안 솔루션들이 구동중인 더 높은 사양의 EC2 인스턴스들이 연결된 ELB 로드 밸런서로 손쉽게 변경해 주실 수 있습니다. 이후 Route 53 가 자동으로 수정된 설정을 반영해 example.com 에 대한 DNS 쿼리에 응답하며, example.com 의 Alias 레코드 세트가 있는 Hosted Zone 를 별도로 변경하지 않아도 됩니다.

이렇듯 Alias 레코드 세트를 만들어 사용하면 공격을 받는 등 긴급상황 발생시 트래픽을 손쉽게 추가적 자원으로 라우팅할 수 있는 유연성을 확보할 수 있습니다. Alias 레코드 세트에 대한 자세한 사항은 다음의 링크를 참고하시기 바랍니다. Alias 레코드 세트를 생성하시려면 다음의 링크를 참고하시기 바랍니다.

Private DNS

Private DNS 기능은 내부 어플리케이션 자원들(웹서버, 어플리케이션 서버, DB 서버 등)에 대해서 외부 노출 없이 내부 DNS 이름을 통해 관리할 수 있도록 해 줍니다. 예를 들어 어떤 내부 자원이 있고 해당 자원을 CNAME 을 통해 라우팅 하고 싶다면, VPC 내부에서 Private DNS 를 사용하시면 됩니다.

Route 53 을 사용해 스플릿 호라이즌 DNS(Split Horizon DNS)를 구성할 수도 있습니다. 만약 여러분이 하나의 웹사이트 도메인을 가지고 고객용 외부버전과 직원용 내부버전으로 나누어 운용하고 싶으시다면, 쿼리를 보내는 사용자의 위치에 따라 각각 퍼블릭 존과 프라이빗 존을 통해 다른 사이트로 라우팅해 주도록 설정하실 수 있습니다. 이 경우 단순히 퍼블릭과 프라이빗 존에 같은 도메인 이름 및 서브도메인을 갖도록 생성하시면 됩니다.

Step 1. 프라이빗 호스트 존 생성하기(Create a Private Hosted Zone)

Step 2: 프라이빗 호스트 존 리스팅하기(List Private Hosted Zone)

Step 3: 프라이빗 호스트 존과 Amazon VPC 연동하기 (Associate Amazon VPCs with a Private Hosted Zone)

Web Application Firewall

어플리케이션 계층의 DDoS 공격은 일반적으로 인프라를 대상으로 하는 공격에 비해 적은 볼륨의 트래픽으로 웹 어플리케이션을 공격합니다. 이런 공격들에 대응하여 여러분의 인프라에 웹 어플리케이션 방화벽(WAF)을 고려해 보실 수 있습니다.

WAF 는 일정한 보안 규칙 세트를 가지고 웹 트래픽에 대한 필터로 동작합니다. 일반적으로 이러한 보안 규칙들은 크로스 사이트 스크립팅(XSS) 또는 SQL 인젝션(SQLi)과 같은 공격을 탐지하는데 사용되지만, DDoS 공격으로 인한 HTTP Get 또는 POST 플러딩을 완화하여 가용성 높은 인프라를 구성하는데 도움을 줍니다.

HTTP 는 사용자와 어플리케이션간 요청-응답의 과정을 거쳐 동작하는데 이때 사용자는 데이터를 요청(GET)하거나, 처리가 되어야 할 데이터를 전송(POST)합니다. GET 플러딩은 동일한 URL 을 매우 빠른 주기로 요청하거나 여러분의 어플리케이션이 가리키는 모든 개체들을 요청하면서 발생하며, POST

플러딩은 여러분의 어플리케이션 서비스 중 가장 많은 시간과 자원이 필요한 것을 반복적으로 요청하여 전체 서비스를 무력화시키는 방법입니다.

WAF 는 여러 가지 기능들을 가지고 있는데, 이를 통해어플리케이션의 가용성에 영향을 미칠 수 있는 공격들을 차단하는데 도움을 줄 수 있습니다. 그 중 한가지가 HTTP Rate 에 제한을 거는 것인데, 이는 한 명의 사용자당 지정된 시간 동안 얼만큼의 HTTP 연결을 사용할 수 있는지 최대 제한을 설정하는 방법입니다. 이를 통해 만약 특정 사용자가 제한된 만큼의 사용량을 넘어선다면, WAF 에서는 해당 사용자의 추가적인 연결 요청을 거부하거나 기존 커넥션이 종료될 때까지 버퍼에 저장해 둘 수 있습니다.

WAF 는 또한 HTTP 요청을 검사해 해당 요청들이 정상적인 패턴인지의 여부를 확인할 수 있습니다. 예를 들어 로그인 문자수가 최대 개수를 넘어가거나 사이트 내의 모든 항목들을 쿼리하는 구문인 경우 등을 차단할 수 있습니다. 그 밖에 어플리케이션 레벨의 DDoS 를 방어하는데 도움이 되는 기능들로는 해당 요청의 주체가 사람인지 컴퓨터인지를 체크하는 CAPTCHA 테스트 또는 특정 IP 에 대한 Reputation 을 체크하는 방법 등이 있습니다.

WAF 를 여러분의 VPC 에 설치하기 위해 제일 처음 하셔야 할 일은 AWS Marketplace 에서 원하시는 WAF 솔루션을 선택하시는 것입니다. AWS Marketplace 는 필요한 소프트웨어 및 각종 솔루션을 손쉽게 찾고 구매해서 EC2 위에서 구동할 수 있는 온라인 상점입니다.

WAF 를 찾는 방법으로는 검색 창에서 Web Application Firewall 이라고 타이핑하거나 Marketplace 내 보안 카테고리를 확인는 방법이 있습니다.

설치하고자 하는 WAF 솔루션을 찾으셨다면 해당 소프트웨어를 설치하시고 여러분의 환경에 맞게 설정을 하실수 있습니다. 관련된 설정을 하시기 전에 Marketplace 에서 선택하신 WAF 의 용량 확장을 위한 추가적인 요구사항들을 잠시 짚고 넘어가도록 하겠습니다.

모든 HTTP 요청들을 검사하기 위하기 위해서는 WAF 를 트래픽의 경로에 인 라인(In-Line)형태로 설치해야 합니다. 안타깝게도 이러한 인라인 방식을 사용하게 되면 해당 솔루션 자체가 트래픽의 병목현상을 야기시킬 수 있습니다. 이러한 문제를 완화시키기 위해서는 트래픽의 급격한 유입 시 해당 WAF 솔루션도 여러 대로 확장될 수 있는 방법들이 필요할 것입니다. 이런 식의 확장 방법으로 "WAF 샌드위치" 가 있습니다.

"WAF 샌드위치"구성에서는 WAF 를 구동하는 EC2 인스턴스들이 오토 스케일링 그룹에 포함되어 ELB 로드밸런서 사이(외부 ELB 와 내부 ELB)에 위치하게 됩니다.

이 구성에서는 가장 앞 단에 위치하는 외부 ELB 가 모든 유입 트래픽을 받아 퍼블릭 서브넷에 있는 WAF EC2 인스턴스로 분배해 줍니다. WAF EC2 인스턴스를 ELB 뒷 단의 오토 스케일링 그룹에 등록해 두면, 트래픽의 갑작스런 증가에 대응해 해당 WAF 의 용량을 스케일 아웃으로 자동 확장할 수 있습니다.

유입 트래픽에 대한 검사가 끝나면 WAF EC2 인스턴스는 내부 ELB 로드 밸런서로 해당 트래픽을 넘겨주게 되고, 이 트래픽은 내부에서 구동되는 EC2 인스턴스들로 분배됩니다. 이러한 아키텍처(아래 그림 참조)를 사용하면 어플리케이션을 구동중인 EC2 인스턴스에 영향을 끼치지 않으면서 WAF EC2 인스턴스의 확장성에 대한 요구사항을 충족시킬 수 있습니다.

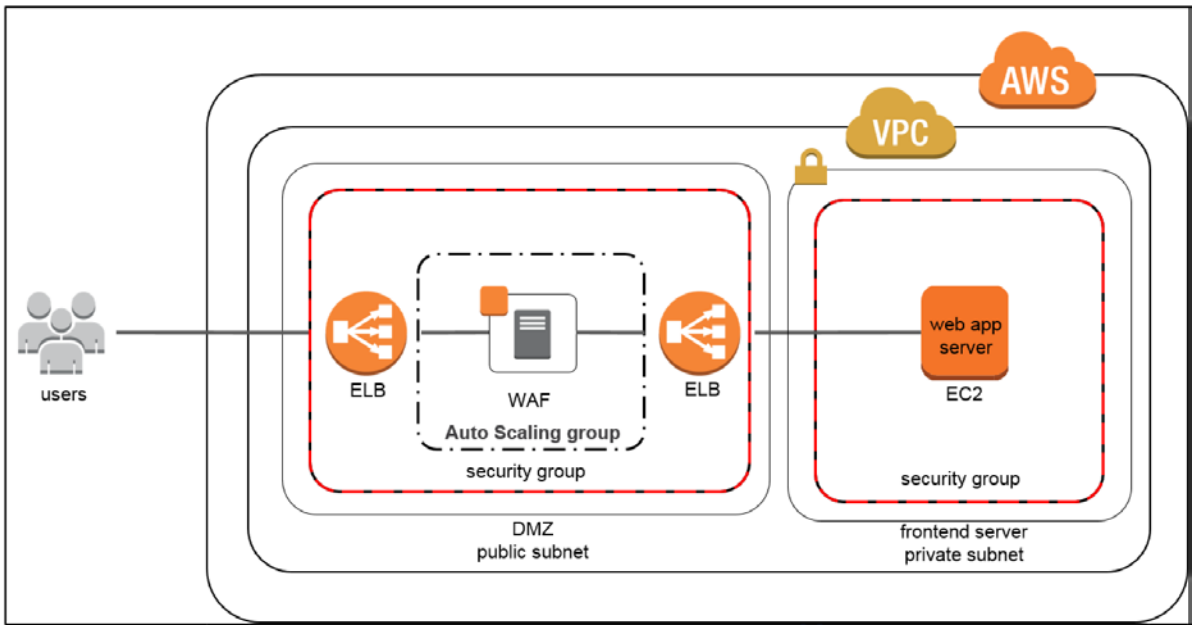

Figure 5. WAF 샌드위치

위와 같이 외부 ELB 와 내부 ELB 가 설정되었다면, 다음 단계는 두 계층의 ELB 사이에 EC2 를 배치하고 WAF 를 설치하는 것입니다. AWS Marketplace 에 등록되어 있는 솔루션들은 "WAF 샌드위치"의 구성을 위한 각기 다른 방법들을 가지고 있을 수 있습니다. Marketplace 에서 특정 벤더의 솔루션을 선택하셨다면 해당 솔루션의 확장기능 설정을 위해 해당 벤더 혹은 파트너 측에 관련 문서를 요청하시기 바랍니다. AWS 프로페셔널 서비스 팀에 문의하셔도 많은 도움을 받으실 수 있습니다.

정상적인 트래픽(사용) 패턴을 숙지 (Learn normal behavior)

안정적인 아키텍처를 보장하는 가장 중요한 방법 중 하나는 언제 여러분의 어플리케이션이나 인프라가 공격을 당하는지 파악하는 것입니다. 흔히 고객 분들은 공격 상황을 서비스의 장애여부에 따라 판단합니다. 이보다 더 좋은 방법은 어플리케이션이 정상 동작할 때의 트래픽 패턴 등을 잘 파악해 그 때의 상태를 서비스의 이상 유무를 판별할 수 있는 기준으로 삼는 것입니다.

일반적으로 DDoS 공격자들은 실제 공격을 수행하기 전에 서비스나 어플리케이션의 한계치를 확인하기 위해 테스트를 수행합니다. 이러한 테스트를 몇 차례 수행하는 동안 공격자들은 어느 정도의 트래픽을 통해 공격을 해야 서비스 및 인프라의 가용성에 영향을 줄 수 있는지를 파악합니다. 이런 상황에서 어플리케이션 또는 서비스에 대한 이상 유무를 빠르게 판단하실 수 있는 근거를 가지고 있다면 보다 효과적으로 대응할 수 있습니다.

Amazon CloudWatch

Amazon CloudWatch 를 사용하면 AWS 에서 구동하는 어플리케이션과 인프라를 모니터링할 수 있습니다. CloudWatch 를 통해 다양한 메트릭 값과 로그 파일을 수집하고, 이러한 메트릭 값들이 기준치를 넘어서면 알림을 발생시키도록 설정할 수 있습니다. 또한 CloudWatch 를 사용해서 시스템 전반에 걸쳐 자원 사용률, 어플리케이션 성능 및 운영 상태 등에 대한 가시성을 확보해 평소 공격이 없을 때의 정상 상태가 어떤 모습인지 확인할 수 있습니다. 이와 같은 정보를 활용하면 DDoS 공격이 발생했을 때 단계별로 어떻게 대응해야 하는지 좀더 쉽게 파악할 수 있습니다.

CloudWatch 경보를 생성하기 전에 여러분의 어플리케이션이 정상 동작하는 환경 및 패턴을 잘 이해하시는 것이 좋습니다. 이를 위해 CloudWatch 에서 제공하는 다양한 메트릭들을 확인하시면 많은 도움이 될 것입니다.

Step 1. 확인 가능한 메트릭 확인

Step 2. 가용한 메트릭 찾기

Step 3. 메트릭 선택

Step 4. 메트릭에 대한 통계 확인

Step 5. 메트릭을 그래프로 확인

Amazon CloudFront 리포트나 Amazon Route 53 헬스체크는 별도의 메트릭을 제공하고 있으며 이를 통해 여러분의 어플리케이션에 들어오는 정상 트래픽에 대해 보다 잘 이해하실 수 있을 것입니다.

여러분의 어플리케이션에 대한 기본적인 패턴 및 환경을 잘 이해하셨다면 CloudWath 를 사용해서 정상적이지 않은 패턴들에 대한 알람을 생성해 보시기 바랍니다. 다음의 테이블은 DDoS 공격에 대한 모니터링을 위해서 권고하는 메트릭들입니다.

토픽	메트릭	설명
Auto Scaling	GroupMaxSize	오토 스케일링 그룹의 최대 크기
AWS Billing	EstimatedCharges	AWS 예상 사용비용
Amazon CloudFront	Requests	초당 HTTP 요청 횟수
Amazon CloudFront	TotalErrorRate	HTTP 상태 코드가 4xx, 5xx 인 비율
Amazon EC2	CPUUtilization	할당된 EC2 컴퓨팅 자원에서 실제 사용중인 비율
Amazon EC2	NetworkIn	인스턴스 당 각 네트워크 인터페이스에서 수신된 Byte 양
Amazon EC2	StatusCheckFailed	인스턴스에 또는 호스트에 대한 상태 장애여부 확인
ELB	UnHealthyHostCount	개별 가용영역에 있는 비정상적인 인스턴스 숫자
ELB	RequestCount	등록된 인스턴스로 수신된 정상적인 요청들의 수
ELB	Latency	요청이 ELB 를 떠난 후 응답이 수신되는데 걸리는 시간을 초로 나타냄
ELB	HTTPCode_ELB_4xx HTTPCode_ELB_5xx	로드 밸런서에서 생성된 HTTP 4xx 또는 HTTP 5xx 코드의 수
ELB	BackednConnectionsErrors	제대로 연결되지 않은 커넥션 수

ELB	SpilloverCount	큐가 꽉 차서 거부된 요청들의 수
Amazon Route 53	HealthCheckStatus	엔드 포인트 헬스체크 상태

위에 열거된 기본 메트릭들에 더해 CloudWatch Log 를 통해 여러분의 어플리케이션에서 생성되는 로그를 모니터링 하실 수 있습니다. Amazon CloudWatch Log 는 Amazon Linux, Ubuntu 그리고 Windows 에 설치할 수 있는 별도의 에이전트로, CloudWatch 로 로그를 전송하는 역할을 합니다. CloudWatch Log 를 사용하면 여러분의 어플리케이션에서 얼마만큼의 에러가 발생하는지를 파악하실 수 있고, 해당 에러의 총량이 미리 설정해 놓은 임계치를 넘어가면 알림을 발생하도록 설정하실 수도 있습니다. 또한 어플리케이션 로그를 모니터링해 "NullReferenceException" 등 특정 텍스트를 찾도록 할 수도 있고, 로그 데이터의 특정 위치에서 보이는 키워드(예를 들어 Apache 의 접근 로그에서 보이는 "404" 상태 코드 등)의 발생 빈도를 확인하도록 할 수도 있습니다. 원하시는 키워드가 있다면 이를 CloudWatch Log 를 통해 CloudWatch 메트릭으로 등록하실 수도 있습니다.

이번 섹션에서는 Amazon Linux 또는 Ubunutu 에 CloudWatch Log 에이전트를 설치하는 단계를 설명합니다. Windows 기반의 Amazon EC2 에 CloudWatch Log 에이전트를 설정하시려면 Amazon EC2 Microsoft Windows 인스턴스 사용 가이드의 해당 부분을 확인하시기 바랍니다.

Step 1: 기존 EC2 인스턴스에 CludWatch Logs 에이전트 설치 및 설정

Step 2: Amazon SNS 설정

Step 3: 알람 생성

사용 가능한 알람에 대한 더 자세한 정보를 얻으시려면 이 링크를 확인하시기 바랍니다.

공격에 대한 대응계획 수립 (Create a Plan for Attacks)

공격을 받는 도중에 대응 전략을 세우는 것은 굉장히 비효율적인 방법입니다. 사전에 이런 공격들에 대한 대응계획을 수립하는 것이 중요하며, 그 이유는 다음과 같습니다.

- 여러분의 아키텍처를 사전에 검증하고, 공격 방지를 위해 선정한 기술과 솔루션들이 여러분의 어플리케이션 및 인프라 환경에 적합한지를 미리 파악할 수 있습니다.
- 공격에 대한 대응 시 확장되는 아키텍처로 인해 증가하는 사용 비용을 파악할 수 있습니다.
- 공격이 발생하면 누구에게 연락해야 하는지를 미리 알 수 있습니다.

이러한 계획을 수립하는 과정에서 어떤 등급의 AWS 기술지원을 받을 것인가도 고려해 보셔야 합니다. AWS 는 각 고객별로 특화된 레벨의 기술지원을 제공합니다. 그러나 DDoS 공격이 발생할 때 보다 높은 레벨의 기술지원을 받고자 하실 경우 다음과 같은 이유들로 AWS 의 엔터프라이즈 등급의 기술지원을 고려하시는 것이 좋습니다.

- **Technical Account Manager (TAM):** TAM 은 여러분의 아키텍처에 대한 세부적인 이해를 바탕으로 AWS 의 모든 서비스에 대한 전문성 있는 기술지원을 제공합니다. TAM 들은 Amazon 의 Solution Architect (SA)와 함께 여러분에게 Best Practice 를 제안해 드리고 기술지원이 필요할 시 바로 연락할 수 있는 컨택 포인트의 역할을 수행합니다.
- **기술지원 케이스 특별 처리:** 엔터프라이즈 고객의 케이스는 긴급한 이슈가 발생할 시 보다 정확하고 빠른 해결을 위해 특별히 훈련된 엔지니어 팀으로 라우팅됩니다.

AWS 의 각 기술지원 등급별 차이 및 특징을 좀더 자세히 알고 싶으시면 이 링크를 참고하시기 바랍니다.

결 론

AWS 는 여러분이 DDoS 공격에 대응하여 안전하고 신뢰성 있는 인프라를 구축하시는 데 도움이 되는 여러 가지 서비스와 기능들을 제공해 드리고 있습니다. 이러한 서비스 및 기능들을 활용해서 어플리케이션 및 인프라를 대응하는 것은 여러분의 몫입니다. 이 문서에 설명된 Best Practice 를 적극 활용하시어 DDoS 공격에 대응할 수 있는 안정적인 서비스를 구성하시기 바랍니다.