GCP Infra 구성

3-Tier 구축

Choi Seong Gi, 3-Tier Infra I linux1547@hanmail.net

목차

GCP Infra 구성

3-Tier 구성내용

개요

GCP Infra 구축 조건
동물병원에 대한 웹사이트 구성

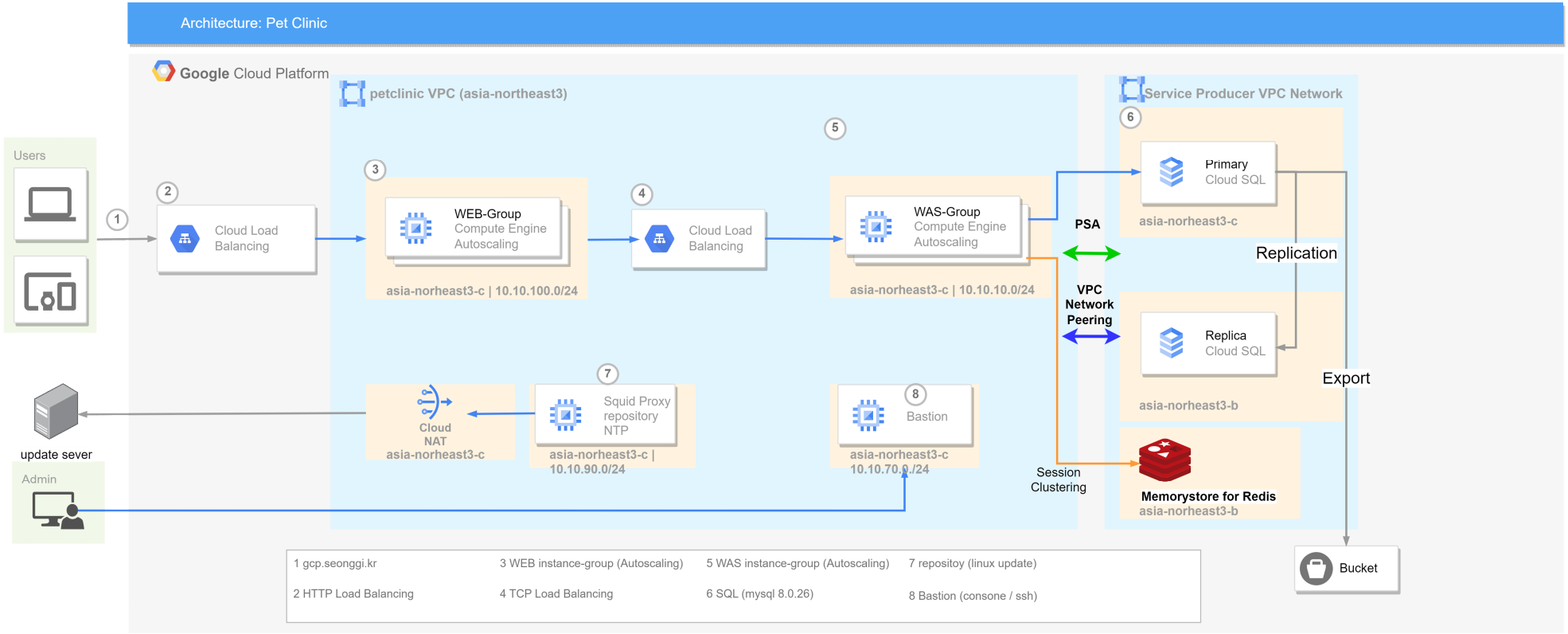
서비스 개요
동물병원은 사용자 정보와 매칭되는 동물의 정보를 저장하고 보여주는 웹사이트

서비스 구성에 대한 조건
- **3-Tier로 구성**
- 접속은 IP가 아닌 도메인으로 하며 SSL을 사용 : **gcp.seonggi.kr**
- 외부에서의 접속은 지정된 장소(IP), 아니면 그에 상응되는 보안을 가진 프로세스로 접근할 것
- 공인IP는 되도록 부여하지 않을 것
- 해당 서비스는 한국에서만 제공하고 있음
- DR을 구성하되 4시간 이내로 동작 되어야 한다.
  DR의 구성은 원본 서비스와 동일하지 않아도 된다.
- 백업은 매일 사용량이 적은 시간대에 실행되며, 별도 파일로 최종 보관되어야 한다.
- DB데이터는 온프레미스에도 복제 되어야 한다.

고려사항
- 되도록 GCP의 서비스를 사용하여 진행

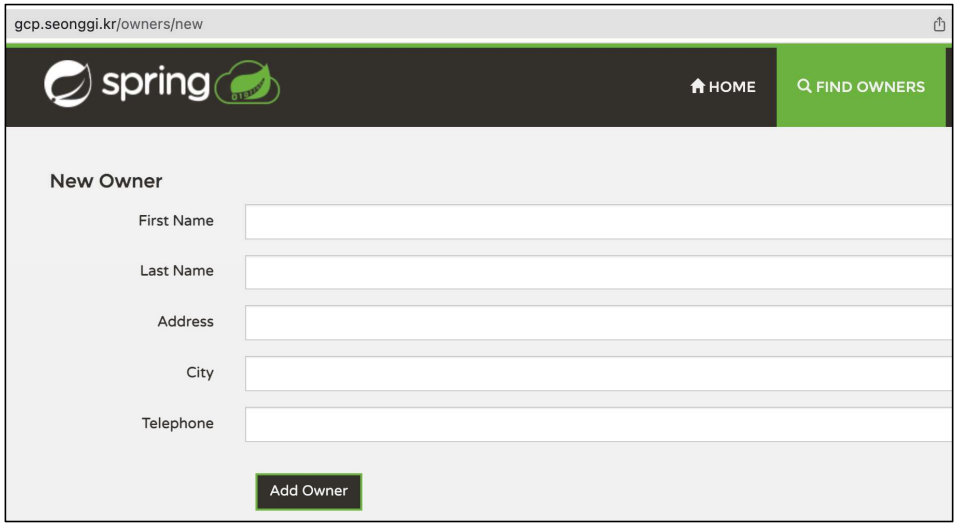
구성도

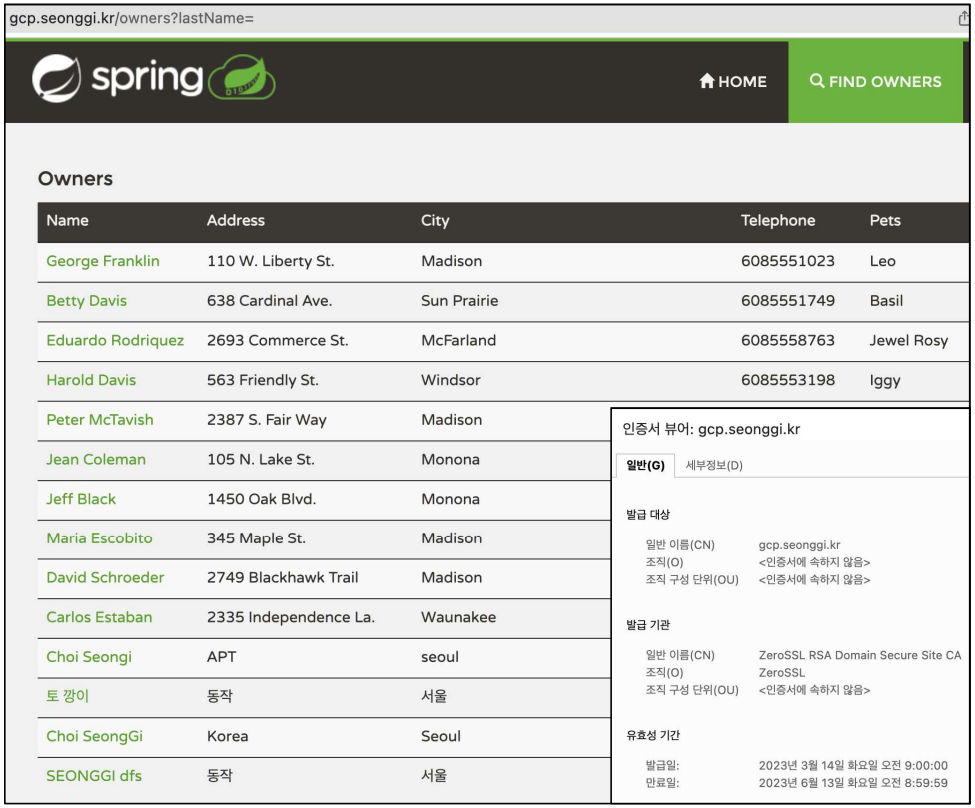
Pet Clinic



구축된 Web Site

gcp.seonggi.kr





WEB

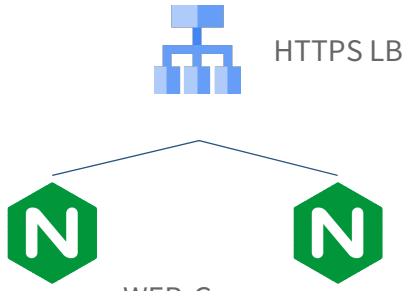
WEB Server 구축
- Nginx을 사용
- 해당 웹사이트는 동물병원의 사용자, 펫의 정보를 보여주는 것으로 정적 컨텐츠가 많습니다.

구성내역
- Reverse Proxy로 구성
- Load balancing에서 보낸 http(80)패킷을 받아 WAS의 8080포트로 전달 합니다.
- Nginx -> Load Balancing(internal / TCP) -> WAS Server

Nginx를 선택한 사유
- Tomcat으로 충분히 동작 가능한 적은 웹사이트이므로, 가벼운 Nginx를 선택
- Reverse proxy로 사용이므로, Apache보다 더 간단한 설정
- Apache와 달리 구성 시스템이 없어 빠른 속도

비고
- AutoScaling 구성

WEB 구성도



WEB Server
OS : Ubuntu, 20.04 LTS
CPU : vCPU 1Core
Memory : 3.75GB
DISK : 10GB

WAS

WAS Server 구축내용
- SW : Tomcat
- JDK패키지 설치

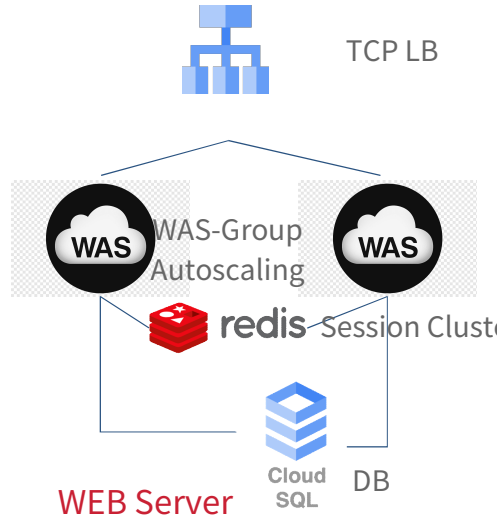
구성내역
- 앞 단의 TCP LB에서 각 WAS로 패킷을 보냄
- Session Clustering은 Redis로 구성

고려사항
Tomcat에서 Session Clustering을 사용하려면 멀티캐스트 (45564)를 사용하나
GCP VPC에서 지원하지 않습니다. (유니캐스트만 지원)

다른 방안은?
별도의 세션 서버를 구성하던가, Redis를 통해 Session Clustering하는 방법

```
linux1547@was-group-258q:~$ netstat -nr
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
0.0.0.0         10.10.10.1      0.0.0.0         UG        0 0          0 ens4
10.10.10.1      0.0.0.0         255.255.255.255 UH        0 0          0 ens4
224.0.0.0       0.0.0.0         240.0.0.0       U         0 0          0 ens4
linux1547@was-group-258q:~$ sudo tcpdump -i ens4 -s0 -vv net 228.0.0.4
tcpdump: listening on ens4, link-type EN10MB (Ethernet), capture size 262144 bytes
```

WAS 구성도



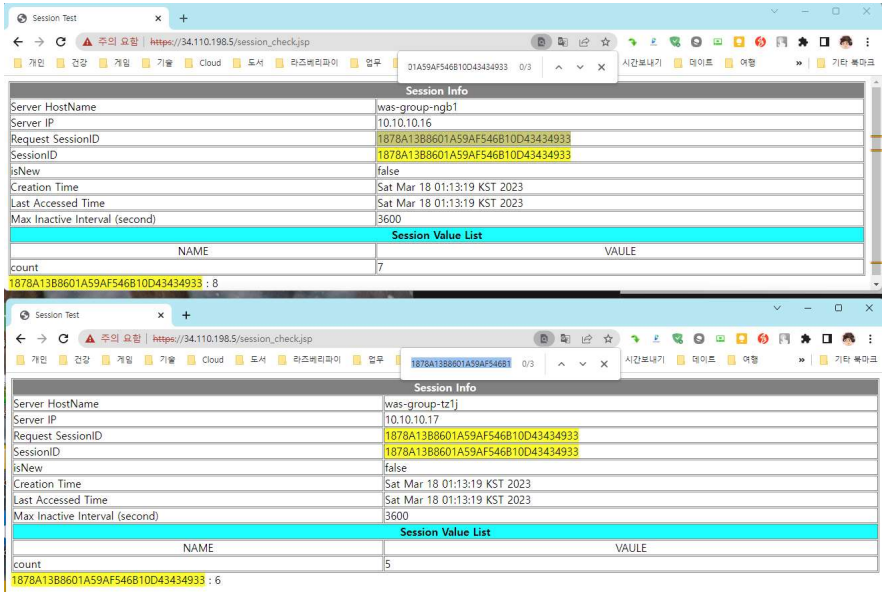
WEB Server
OS : Ubuntu, 20.04 LTS
CPU : vCPU 1Core
Memory : 3.75GB
DISK : 10GB

WAS

Redis로 Sessions Cluster구성

GCP Memorystore Redis 서비스를 사용
tomcat-cluster-redis-session-manager를 구성하여
구성도와 같이 Sessions Cluster를 구성

- 테스트 결과 (Session ID공유 확인)



Session Test

주의 요함 | https://34.110.198.5/session_check.jsp

Session Info	
Server HostName	was-group-ngb1
Server IP	10.10.10.16
Request SessionID	1878A13B8601A59AF546B10D43434933
SessionID	1878A13B8601A59AF546B10D43434933
isNew	false
Creation Time	Sat Mar 18 01:13:19 KST 2023
Last Accessed Time	Sat Mar 18 01:13:19 KST 2023
Max Inactive Interval (second)	3600

Session Value List	
NAME	VAULE
count	7

1878A13B8601A59AF546B10D43434933 : 8

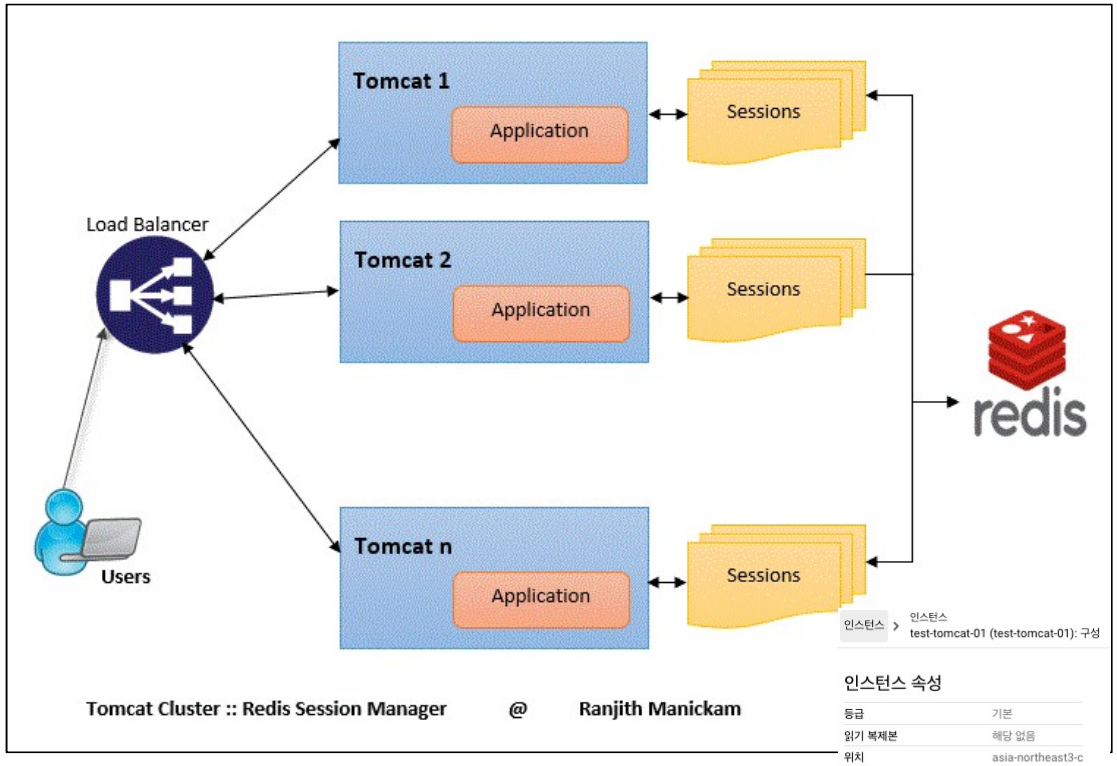
Session Test

주의 요함 | https://34.110.198.5/session_check.jsp

Session Info	
Server HostName	was-group-tz1j
Server IP	10.10.10.17
Request SessionID	1878A13B8601A59AF546B10D43434933
SessionID	1878A13B8601A59AF546B10D43434933
isNew	false
Creation Time	Sat Mar 18 01:13:19 KST 2023
Last Accessed Time	Sat Mar 18 01:13:19 KST 2023
Max Inactive Interval (second)	3600

Session Value List	
NAME	VAULE
count	5

1878A13B8601A59AF546B10D43434933 : 6



인스턴스 > 인스턴스
test-tomcat-01 (test-tomcat-01): 구성

인스턴스 속성

등급	기본
읽기 복제본	해당 없음
위치	asia-northeast3-c
기본 위치	asia-northeast3-c
용량	1 GB
최대 메모리	1 GB
RDB 스냅샷	사용 안함
최대 네트워크 처리량	500 MB/s
버전	6.x
예상 비용	US$47.45/월

DB

DBMS 구축내용

- GCP의 PaaS서비스 Cloud SQL(MySQL)로 구성
- Public IP가 필요 없는 환경이므로, 할당제외
- 비공개 IP로 별도의 VPC로 설정 (10.10.90.0/24)
- 원본, 복제본 생성

인스턴스 ID	유형	공개 IP 주소	비공개 IP 주소	인스턴스 연결 이름	고가용성	위치	사용된 저장용량
petclinic	MySQL 8.0		10.10.90.3	linux1547-int-20...	추가	asia-northeast3-c	1GB/100GB
petclinic-replica	MySQL 읽기 복제본		10.10.90.12	linux1547-int-20...	추가	asia-northeast3-c	1GB/100GB

- Petclinic Table생성

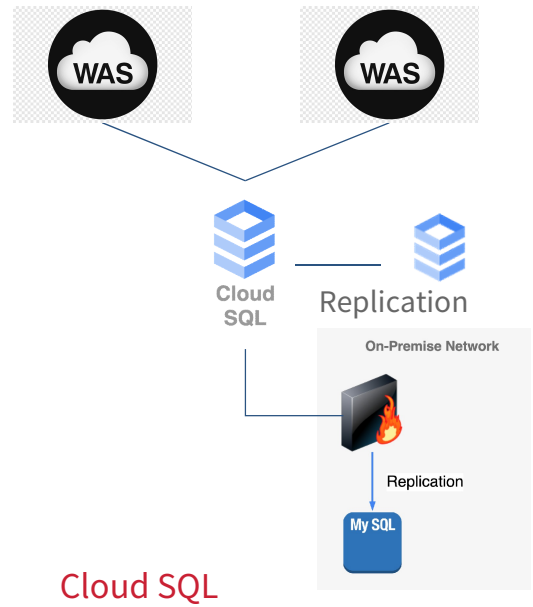
```
mysql> show tables;
+--------------------+
| Tables_in_petclinic |
+--------------------+
| owners             |
| pets               |
| specialties        |
| types              |
| vet_specialties    |
| vets               |
| visits             |
+--------------------+
7 rows in set (0.01 sec)
```

- 복제설정

On-premise에 위치하고 있는 Mysql 에 VPN을 통하여 복제를 진행

```
                Master_Host: 10.10.90.3
                Master_User: dbto
                Master_Port: 3306
              Connect_Retry: 60
            Master_Log_File:
        Read_Master_Log_Pos: 4
             Relay_Log_File: ubuntu-relay-bin.000001
              Relay_Log_Pos: 4
      Relay_Master_Log_File:
           Slave_IO_Running: No
          Slave_SQL_Running: Yes
            Replicate_Do_DB:
        Replicate_Ignore_DB: petclinic
```

DB 구성도



Cloud SQL

DB : MySQL 8.0
복제1 : GCP내 읽기복제본 생성
복제2 : on-primise mysql

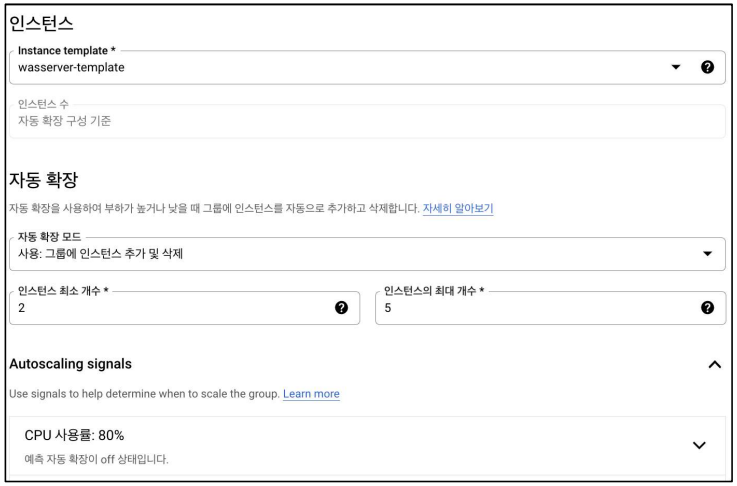
AutoScale

AutoScale 설정

- GCP 인스턴스 그룹에서 관리형으로 설정시 사용가능
- WAS에서 Sessions Cluster을 사용하기에 세션정보를 저장하지 않는 stateless로 진행
- AutoScale설정을 위해 아래와 같이 작업하였습니다.
- 모든 설정이 완료된 Template VM생성 -> 스냅샷 생성 -> 이미지 생성 -> 인스턴스 템플릿 생성 -> 인스턴스 그룹생성 -> 그룹별에 맞게 상태체크 생성

WEB, WAS Server Group

- **Signal은 CPU사용률 80%로 설정**
- HTTP부하 분산 사용률은 외부 부하 분산만 가능하여 WAS 적용불가



네트워크

VPC - Subnet

- 서버별 영역 생성

이름	리전	내부 IP 범위	게이트웨이	비고
bastion	asia-northeast3	10.10.70.0/24	10.10.70.1	모든서버에 접속 가능한 영역
Forward-proxy	asia-northeast3	10.10.20.0/24	10.10.20.1	Forward-proxy, NTP, Repository
petclinic-was	asia-northeast3	10.10.10.0/24	10.10.10.1	WAS 영역
petclinic-web	asia-northeast3	10.10.100.0/24	10.10.100.1	WEB 영역
petclinic-db	asia-northeast3	10.10.90.0/24		PSA 영역

Loadbalance설정

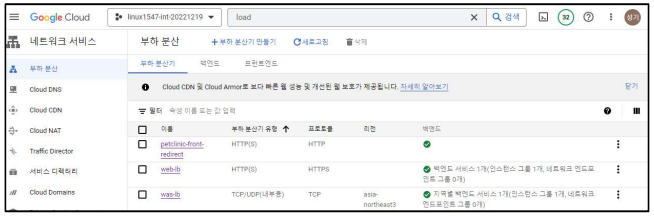
- 프론트엔드 구성
- 백엔드 구성
- 상태체크 : Service Port만 오픈 되어 있는 것이 아닌 WEB Response 내용 체크
- 라우팅 규칙

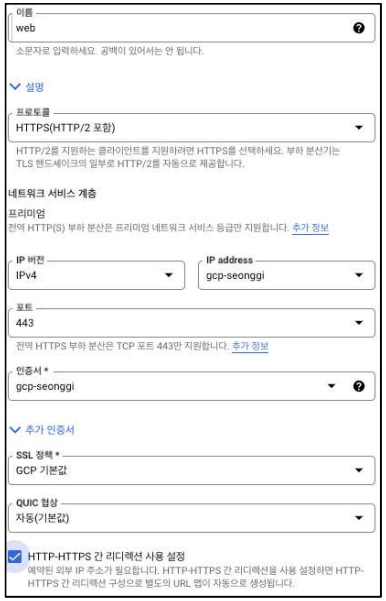
네트워크

Load Balancer설정

- 프론트엔드 구성

HTTPS설정
HTTP -> HTTPS간 리다이렉션 설정
인증서는 자체 인증서를 발급받아 사용
SSL offload를 사용하여 보안성 강화





- 백엔드 구성

백엔드 유형 : 인스턴스 그룹
백엔드 세부 설정은 생성한 인스턴스 그룹 선택 1개의 백엔드를 사용하므로, 그 외는 기본 값으로 설정

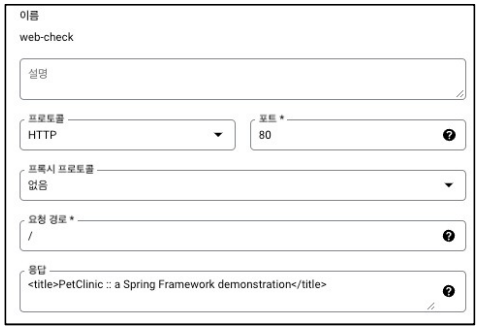
고급설정 -> 트래픽 정책 -> 최소 요청 (Least request)

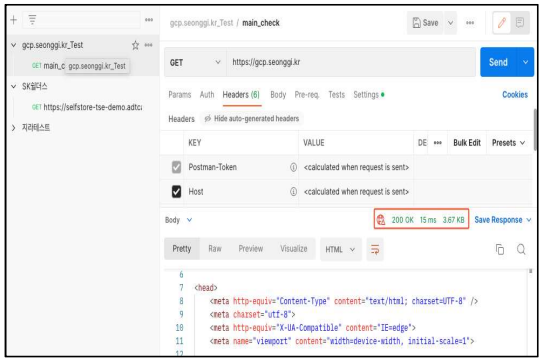
네트워크

Load Balancer설정

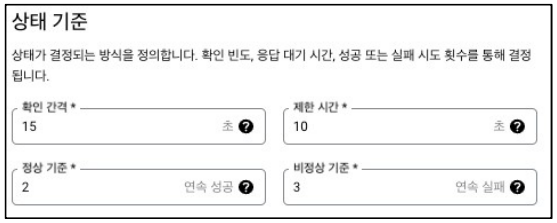
- 상태체크

프로토콜 : HTTP
요청경로 : index.jsp 이므로 기본인 ”/” 선택
응답 : <title>PetClinic :: a Spring Framework demonstration</title>





상태 기준은 상황에 맞게 설정
이번 과제에서는 1개의 백엔드를 사용하므로, 여유롭게 설정



- 라우팅규칙

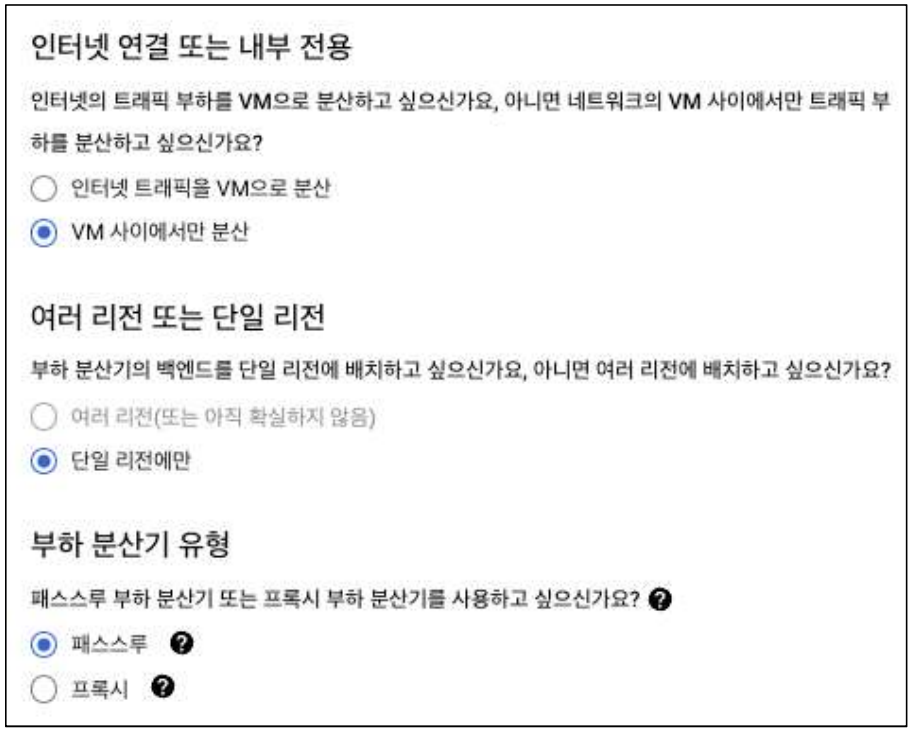
HTTP 요청의 호스트명에 따라 백엔드 서비스로 라우팅하는 규칙 설정
HTTP URI를 기반으로 라우팅 규칙을 설정
이번 과제에서는 모든 트래픽을 한 개의 백엔드 서비스로 라우팅 하기 때문에 별도의 규칙 설정 없이 기본 서비스로 사용

네트워크

Load Balancer설정

- 내부 TCP/UDP 부하분산

앞 단이 VM이므로, VM 사이에서 분산, 단일 리전에 패스스루로 내부분산으로 설정하였습니다.



- 백엔드 구성

인스턴스 그룹설정, Was Server체크 상태 확인, 세션 어피니티 설정

- 프론트엔드 구성

단일포트, 내부 고정IP생성

네트워크 보안

방화벽 규칙

규칙명	유형	소스 (IP 범위)	대상	프로토콜/포트	적용	순위
petclinic-allow-http	수신	34.110.198.5/32	webserver	tcp:80	허용	100
LB -> Web Service 허용						
web-was	수신	10.10.10.70/32	wasserver	tcp:8080	허용	110
Web(LB)-> Was간 통신						
allow-apt-update	수신	10.10.70.0/24, 10.10.10.0/24, 10.10.100.0/24	forward-proxy	tcp:9090, udp:9090	허용	120
forward-proxy						
allow-health-check	수신	130.211.0.0/22, 35.191.0.0/16	wasserver, webserver	tcp:80, 8080	허용	200
GCP 에서 상태체크를 위한 정책						
ntp-service	수신	10.10.0.0/16	forward-proxy	udp:123	허용	300
NTP서비스를 위한 정책						
onpremise-allow	수신	192.168.1.0/24	전체 적용	all	허용	400
Onpremise와 VPN으로 통신 정책						
vpn-to-aws	수신	192.168.50.0/24	전체 적용	all	허용	410
Aws(DR)과 VPN으로 통신 정책						
iap-to-ssh	수신	35.235.240.0/20	전체 적용	tcp:22	허용	800
GCP의 iap를 이용한 SSH접속 허용						
bastion-ssh	수신	211.208.159.146/32	bastion	tcp:51547	허용	1000
사무실, 재택에서의 bestion접속						
ssh-allow	수신	10.10.70.200/32	webserver, wasserver, forward-proxy	tcp:22	허용	1000
Bestion -> 서버별 ssh접속 허용						
all-deny	수신	0.0.0.0/0	전체 적용	all	거부	1001
내부로 들어오는 패킷 차단						

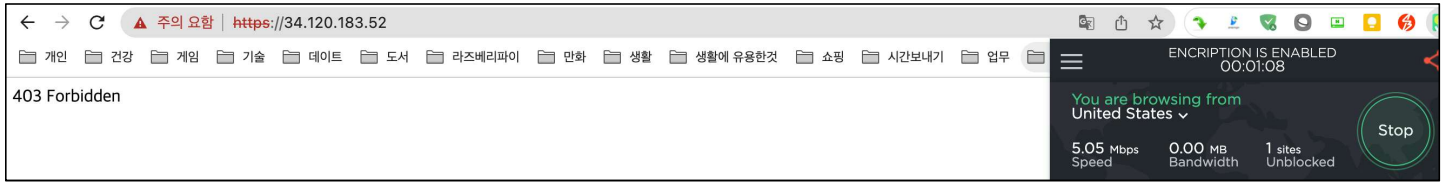
네트워크 보안

Cloud Armor

조건 : 한국을 제외하고, 모든 국가 차단
WAF기능으로는 사전 구성된 WAF 규칙으로 진행
DDOS 기능 활성화

- 한국을 제외하고, 모든 국가 차단
Global Load balancer (HTTP/HTTPS)를 사용시에만 가능 (내부 부하분산은 불가)
Apache라면 GeoIP database를 사용할 수 있으나
Apache에 클라이언트가 접속 후 차단, 허용된다는 점이 다름

국가 코드를 사용하여, 한국 외에는 거부 처리 origin.region_code == "KR"
VPN으로 미국으로 우회하여 접속시 차단됨을 확인하였고, 설정대로 403 메시지 출력 확인



네트워크 보안

Cloud Armor

WAF기능 사용
사전 구성된 WAF 규칙은 GCP Docs를 참조하여 아래와 같은 형식으로 규칙 추가
evaluatePreconfiguredExpr('sqli-v33-stable', ['owasp-crs-v030301-id942421-sqli’])
evaluatePreconfiguredExpr(' WAF규칙이름’, [‘규칙ID’])

유의사항
WAF보안 정책, 규칙당 요금부과

네트워크 보안

Cloud Armor

DDOS방어

L3, L4 계층의 DDOS 방어
기본적으로 HTTP, TCP, SSL Load balancing 사용시 자동으로 설정됨

ICMP flooding, SYN flooding같은 공격들 방어

L7 계층의 DDOS 방어
정책을 생성하거나, Adaptive Protection 사용설정을 하면 사용가능
대량의 트래픽으로 공격 되는 것은 아니기에 머신러닝을 이용한 학습 후 적용하는 과정
공격을 감지, 알림 -> 정책 생성 -> 규칙생성 후 적용(신뢰도) 제안

보안

IAM

ID 및 액세스 관리 정책을 사용하여, GCP의 각 리소스를 역활별로 제어
온프레미스에서는 접근할 수 있는 IP나 ID로 구분하였으나 GCP에서는 정책적으로 구분 및 사용

서버 담당자 : compute Instance Admin

- Compute Engine (VM) 생성, 삭제, 복제
- 스냅샷 생성, 삭제
- 머신 이미지 생성, 삭제
- 인스턴스 그룹 , 인스턴스 템플릿

네트워크 담당자 :
compute network Admin, compute Load Balancer Admin

- VPC 설정, 편집
- Load Balancer 생성, 수정, 인증서 등록

보안 담당자 : compute security Admin

- 방화벽 규칙 설정
- SSL인증서 생성가능

컴퓨팅 네트워크 관리자
()`roles/compute.networkAdmin`

방화벽 규칙 및 SSL 인증서를 제외한 네트워킹 리소스를 생성, 수정 및 삭제할 수 있는 권한입니다. 네트워크 관리자 역할은 방화벽 규칙, SSL 인증서 및 인스턴스에 대한 읽기 전용 액세스를 허용합니다(임시 IP 주소 보기). 네트워크 관리자 역할은 사용자가 인스턴스를 생성, 시작, 중지 또는 삭제할 수 없습니다.

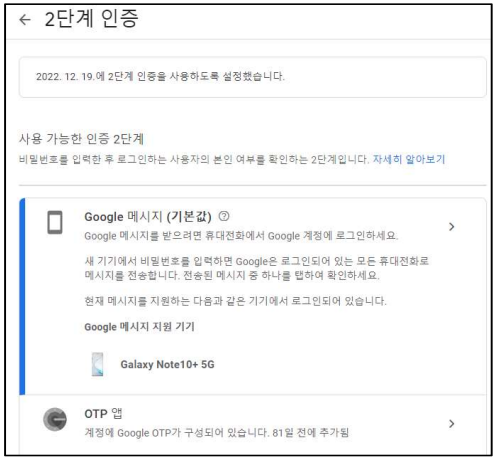
예를 들어 회사에 방화벽 및 SSL 인증서를 관리하는 보안 팀과 나머지 네트워킹 리소스를 관리하는 네트워킹 팀이 있는 경우 이 역할을 네트워킹 팀의 그룹에 부여합니다. 또는 보안과 네트워킹을 모두 관리하는 결합된 팀이 있는 경우 이 역할과 함께 결합된 팀의 그룹에 역할을 부여하십시오.
`roles/compute.securityAdmin`

보안

공통

- MFA 설정
- Bestion Server의 접속은 사무실, 허가 받은 재택IP만 허용처리
- 기본적으로 Bestion을 통해 접속하지만, GCP Console을 통해 IAP로 접속하는 것도 허용
- NTP, Forward-proxy 서버만 직접적으로 Cloud NAT를 통해 외부의 NTP, repository정보에 접근 가능하며
- 그 외 서버들은 꼭 필요한 경우에만 Forward proxy 를 통해 외부의 접근이 가능하도록 했습니다.

1. IAP를 통한 서버 접속
2. MFA(2단계)설정



NTP, Repository, Forward Proxy

NTP

- NTP를 내부에서 사용하는 목적은 모든 리소스들의 시간을 동일하게 하여, 로그의 신뢰성을 가지는 것이 목적
- 내부의 모든 서버들은 NTP서비스를 바라보도록 설정
- 서버 : NTP daemon설치
- Client들은 NTP서버를 바라보도록 /etc/ntpq.conf설정
- NTP서버 내부에서 접속이 가능하도록 UDP123, TCP123 허용
- GCP의 Network, CloudSQL등 서비스들은 custom ntp서비스를 제공하지 않음
  time.google.com을 사용하는 것을 권고

```
linux1547@wasserver-template-1:~$ ntpq -p
     remote           refid      st t when poll reach   delay   offset  jitter
==============================================================================
*forward-proxy.c 169.254.169.254  3 u   28   64  377    0.155   21.421   4.797
```

```
linux1547@wasserver-template-1:~$ cat /etc/apt/apt.conf.d/90proxy
Acquire::http::proxy "http://10.10.20.100:9090/";
linux1547@wasserver-template-1:~$ sudo apt-get update
Hit:1 http://asia-northeast3.gce.archive.ubuntu.com/ubuntu focal InRelease
Hit:2 http://asia-northeast3.gce.archive.ubuntu.com/ubuntu focal-updates InRelease
Get:3 http://asia-northeast3.gce.archive.ubuntu.com/ubuntu focal-backports InRelease [108 kB]
Hit:4 https://packages.cloud.google.com/apt google-cloud-ops-agent-focal-all InRelease
Get:5 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Fetched 222 kB in 1s (306 kB/s)
Reading package lists... Done
```

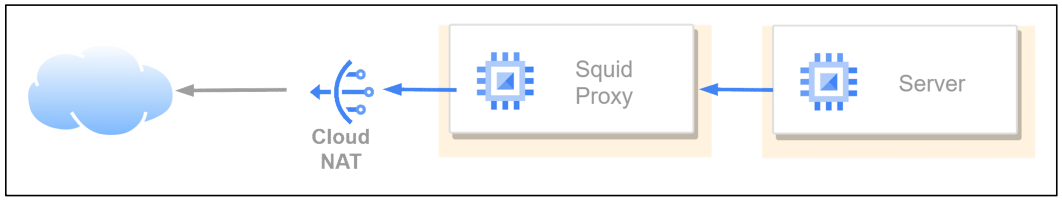
```
root@wasserver-template-1:~# curl -v google.co.kr
* Uses proxy env variable http_proxy == 'http://10.10.20.100:9090'
*   Trying 10.10.20.100:9090...
* TCP_NODELAY set
* Connected to 10.10.20.100 (10.10.20.100) port 9090 (#0)
> GET http://google.co.kr/ HTTP/1.1
> Host: google.co.kr
> User-Agent: curl/7.68.0
> Accept: */*
> Proxy-Connection: Keep-Alive
>
* Mark bundle as not supporting multiuse
< HTTP/1.1 301 Moved Permanently
```

Repository – Forward Proxy

- 내부에서 apt update로 실행시 Forward Proxy를 통해 업데이트 받는 구조

forward proxy

- squid proxy 사용 (9090Port사용)

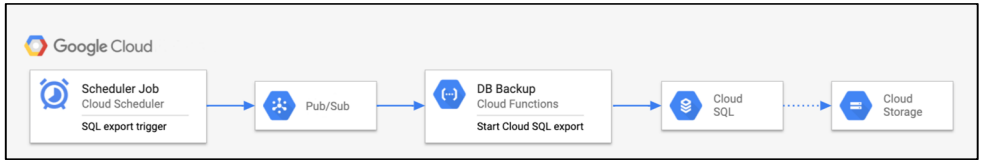


1. ntp데몬 – 내부 ntp서버(10.10.20.100) 정보를 캐싱
2. apt update설정
3. 서버에서 curl 로 체크시 10.10.20.100 서버로 요청

백업

GCP 백업

- VM 인스턴스
  스냅샷 일정을 통해 주기적으로 VM 인스턴스에 대한 백업 진행
- DB 백업은 자동으로 진행
- 다른 곳으로 이전하기 위해서는 파일로 Export가 필요
  자동화 방법은 아래와 같으며, 별도로 보내야할 특별한 시나리오에서 사용
  Cloud Scheduler -> pub/sub -> Cloud Functions -> Cloud SQL
  Cloud Storage -> rsync -> VM



- 파일 백업
- gsutil 사용하여 crontab에 등록하여 사용
- gsutil cp –r /원본_절대경로 gs://bucker명
- sync가 필요하다면 gs util의 rsync 사용
- 그 이후에는 buckets에서 백업설정에 의해 진행

유의사항

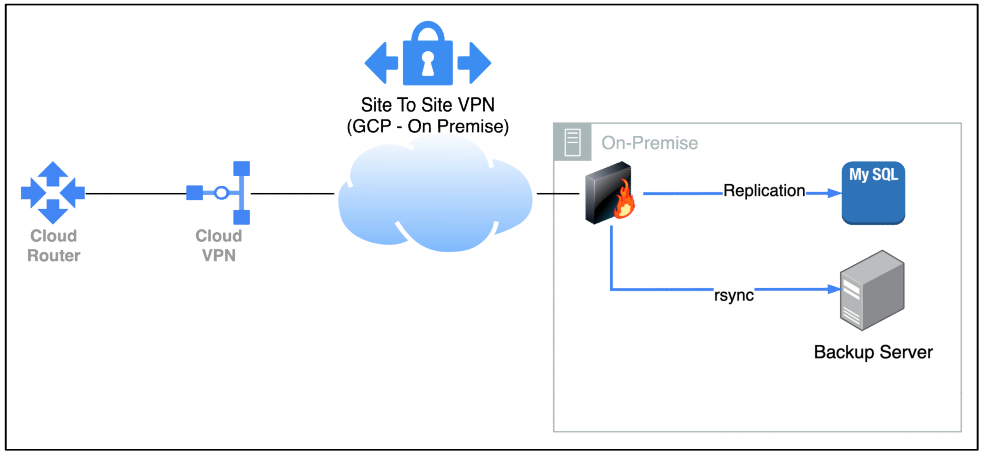
bucket 업로드시 서버의 API 엑세스 허용처리, /.gsutil 디렉토리 삭제하여 권한 재로드 필요

백업

On-Premise (Site to Site VPN)

- 정책기반의 VPN선택
- Site to Site VPN
- On-Premise 쪽은 오픈소스인 StrongSwan 사용

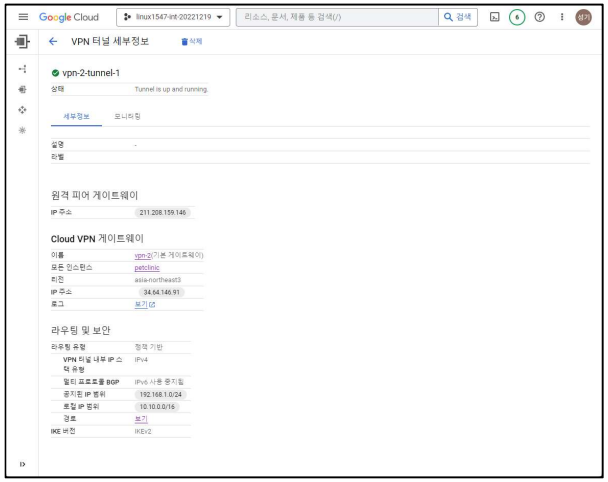
고려사항 : VPC network와 On-Premise network간
네트워크 IP 중복되지 않도록 설계 필요



GCP to On-Premise 백업

- DB파트에서 작성한 Mysql Replication
- 매일 파일백업을 rsync로 복제
- rsync –hrave “ssh –I 키파일”/소스(원본) user@타켓서버:디렉토리
- sudo rsync -hrave "ssh -i /home/linux1547/id_rsa_4096" /home/snap/ bastion@ip:/home/backup/

```
linux1547@bastion: $ sudo rsync -rave "ssh -i /home/linux1547/id_rsa_4096" /home/linux1547/snap bastion@10.10.10.18:/home/backup/
sending incremental file list
snap/
snap/google-cloud-cli/
snap/google-cloud-cli/current -> 111
snap/google-cloud-cli/111/
snap/google-cloud-cli/115/
snap/google-cloud-cli/common/
```



DR

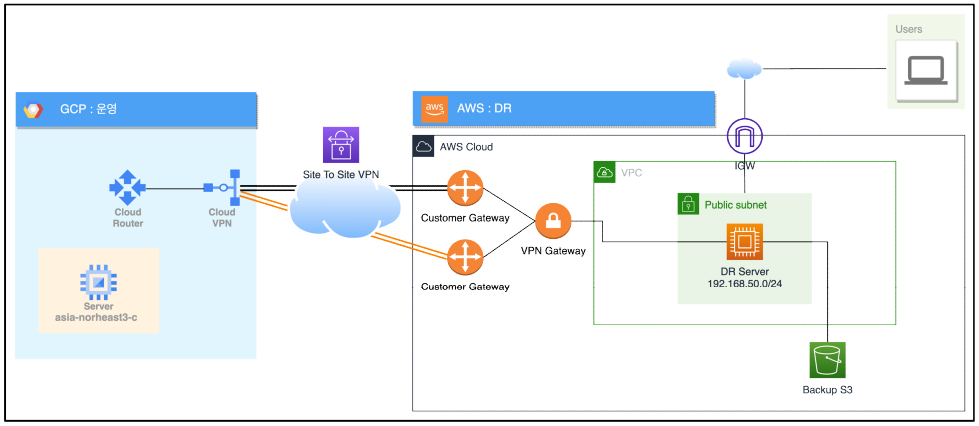
AWS DR구성

- DR구성의 조건
- DR의 구성은 원본 서비스와 동일하지 않아도 된다.
- **DR을 구성하되 4시간 이내로 동작 되어야 한다.**

- 구성내역
- Site to Site VPN으로 연결하여 외부가 아닌 내부통신으로 백업을 진행한다.
- HA VPN으로 설정
- 기본 DNS TTL을 2시간으로 두고 DR변환시 TTL을 1초로 변경하여 전환하는 방법으로 진행

DR 변경 절차

1. 장애상황인지
2. DR서비스 이전 결정
3. Backup된 WAS, DB데이터로 DR서버 재기동
4. gcp.seonggi.kr A레코드 값을 AWS의 public IP / TTL값을 최소(1~10초)로 변경
5. 서비스 접속 확인



모니터링

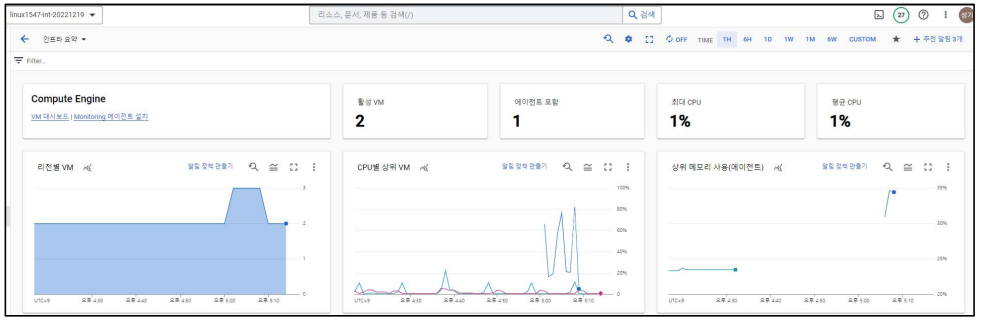
GCP리소스에 대한 다양한 정보 수집, 시각화

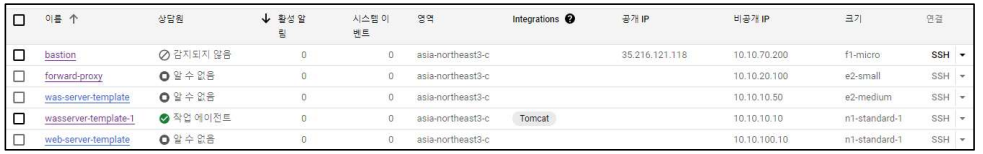
- GCP Monitoring
- 기본적인 CPU사용률, 네트워크 트래픽, IOPS등은 제공
- VM 작업 에이전트 설치시 메모리, 디스크 사용률, 프로세스등 상세한 모니터링 가능

- 운영 에이전트 설치 방법
- gcloud, Ansible, Terraform, Agent다운로드 후 설치
  개별설치로 agent 스크립트를 다운받아, 설치 진행 가능
- 설치시 다양한 정보수집을 위한 agent들이 같이 설치
- 각각의 서비스로 이루어짐

```
linux1547@bastion:~$ service google-
google-cloud-ops-agent                          google-osconfig-agent
google-cloud-ops-agent-diagnostics              google-oslogin-cache
google-cloud-ops-agent-fluent-bit               google-shutdown-scripts
google-cloud-ops-agent-opentelemetry-collector  google-startup-scripts
google-guest-agent
```

```
linux1547@bastion:~$ service google-cloud-ops-agent status
● google-cloud-ops-agent.service - Google Cloud Ops Agent
     Loaded: loaded (/lib/systemd/system/google-cloud-ops-agent.service; enabled; vendor preset: enabled)
     Active: active (exited) since Thu 2023-03-16 00:23:07 KST; 1min 54s ago
    Process: 493 ExecStartPre=/opt/google-cloud-ops-agent/libexec/google_cloud_ops_agent_engine -in /etc/google-cloud-o
    Process: 700 ExecStart=/bin/true (code=exited, status=0/SUCCESS)
   Main PID: 700 (code=exited, status=0/SUCCESS)
```



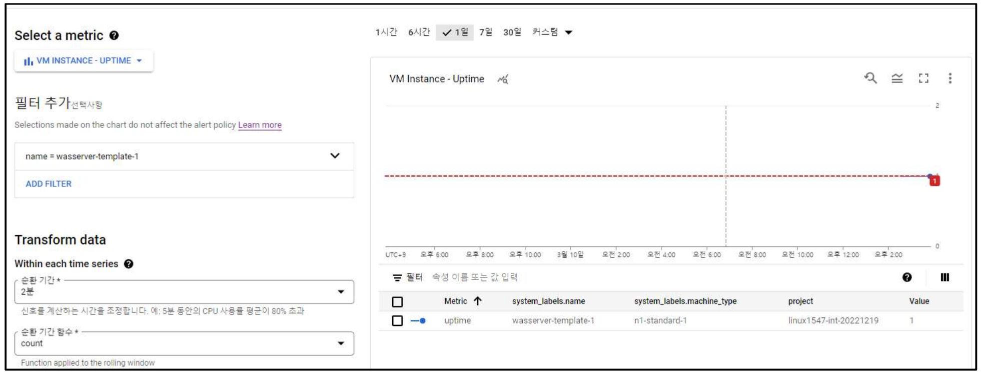


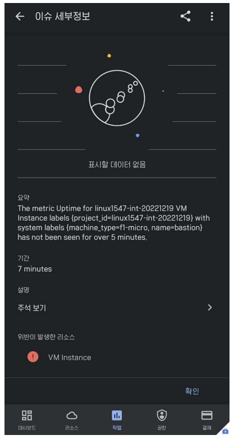
모니터링

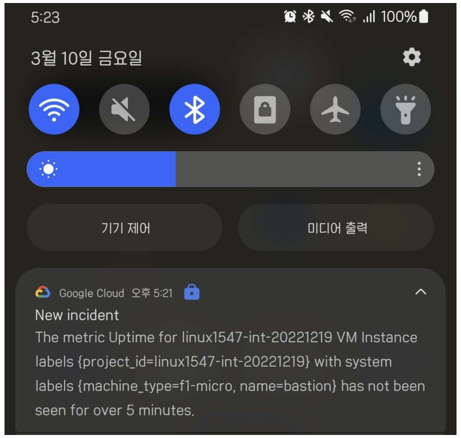
GCP리소스에 대한 다양한 정보 수집, 시각화

- 애플리케이션 모니터링
  Nginx, tomcat, IIS등 여러 제품에 대해 모니터링 지원
  각 어플리케이션에
  config.yaml 적용

- 알림설정
  Slack, Email, SMS, Mobile App등 다양하게 지원

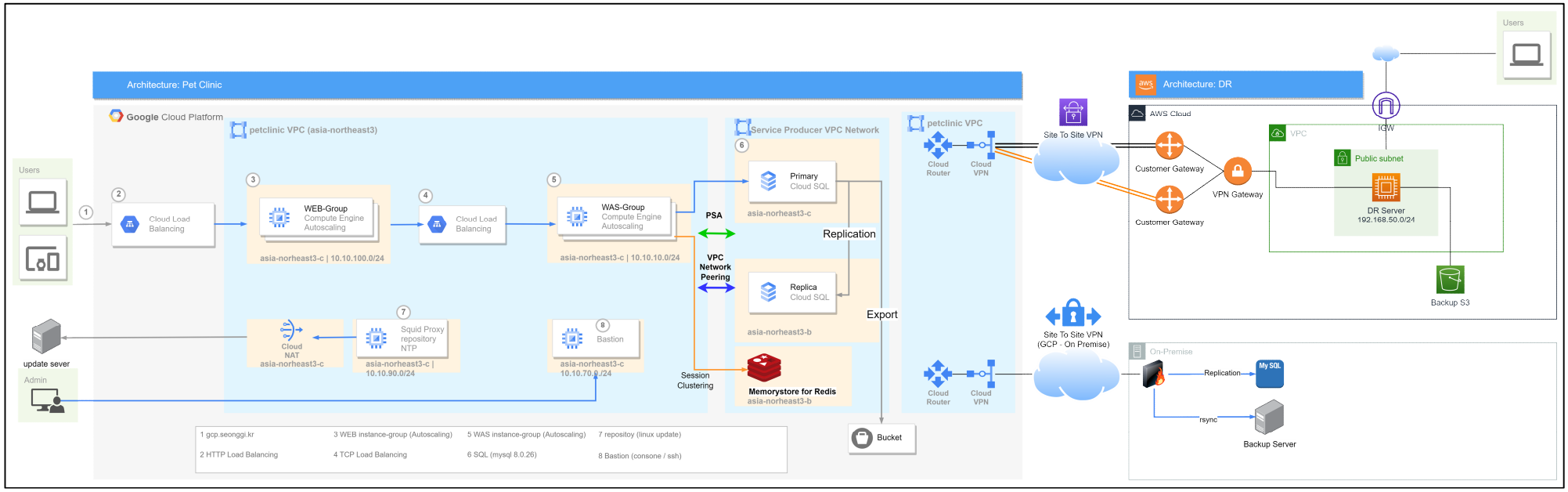






전체 구성도

DR구성을 포함한 전체 구성도



Thank you