전세계 클라우드를 내 손안에, 멀티 클라우드

클라우드바리스타 커뮤니티 제11차 컨퍼런스

One Secure Network for Multi-Cloud

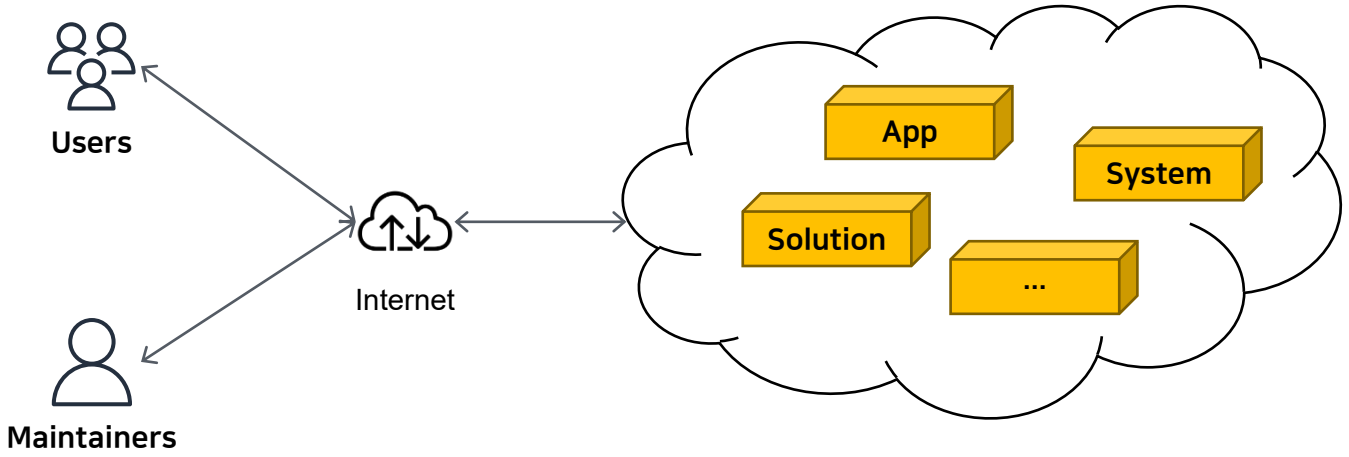
이종 클라우드에 안정적인 네트워크 기반 구축하기

메인테이너@클라우드바리스타

김 윤 곤

얼그레이 (Earl Grey) 한잔 어떠세요 ?

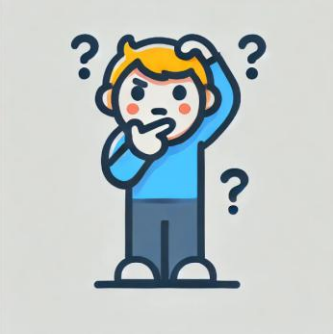
클라우드에 **직접 프로그램을 배치하고 사용하다 보면…**



이런 생각이 듭니다…

괜찮나? 안전한가?

(사용자가 관리해야 하는 영역에서 안전한가 입니다 ^^;;;)



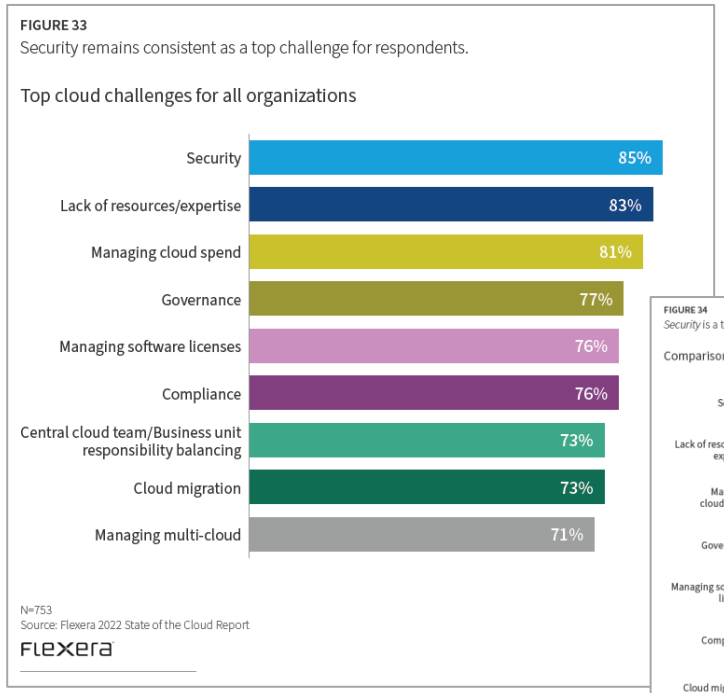
Thanks DALL·E ☺

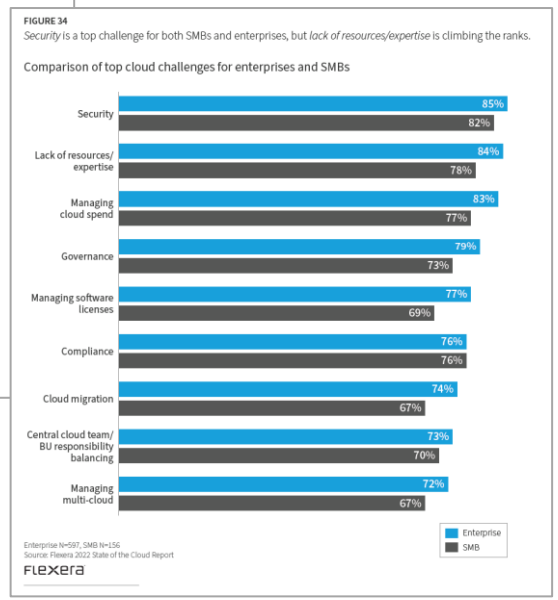
CLOUD BARISTA

목 차

Trends in Top Cloud Challenges

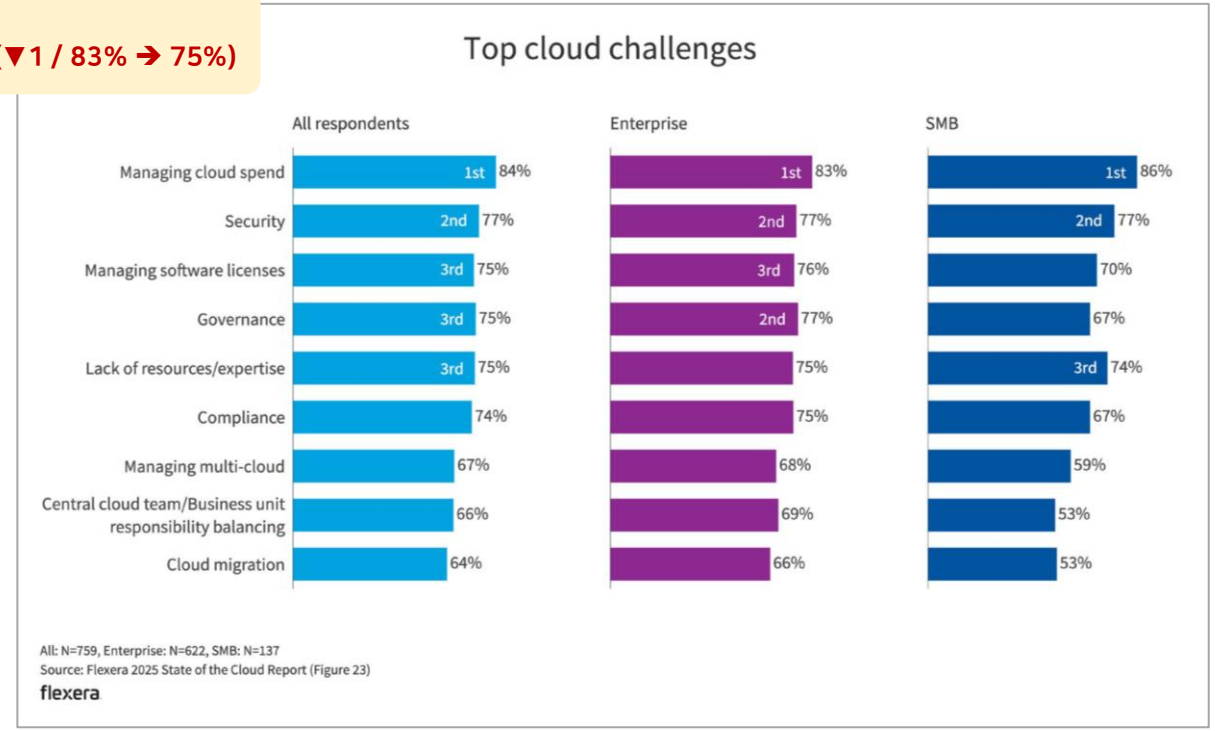
Flexera 2022 State of the Cloud Report





1. Managing cloud spend (▲4 / 81% ➔ 84%)
2. Security (▼1 / 85% ➔ 77%)
3. Managing software licenses (▲2 / 76% ➔ 75%)
4. Governance (▲1 / 77% ➔ 75%)
5. Lack of resources/expertise (▼1 / 83% ➔ 75%)

Flexera 2025 State of the Cloud Report

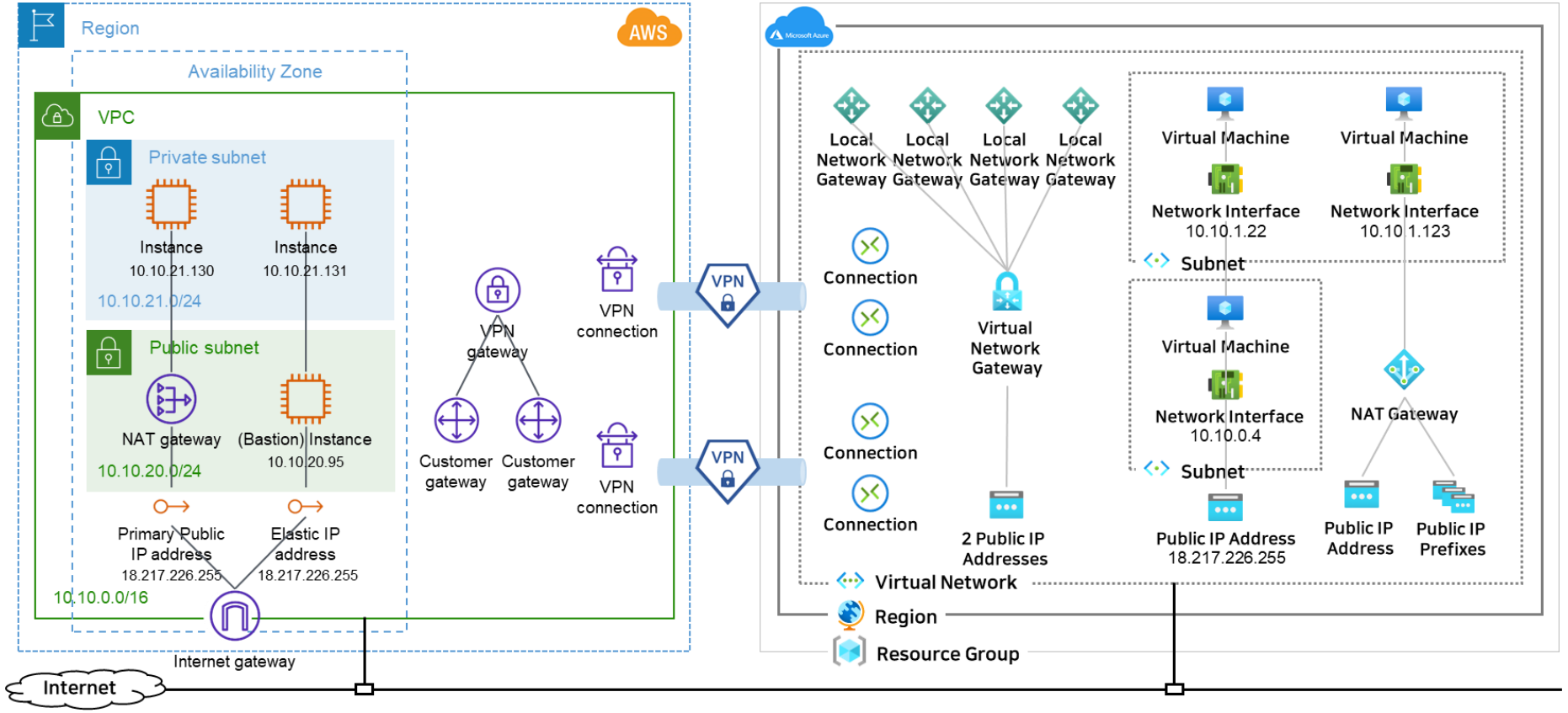


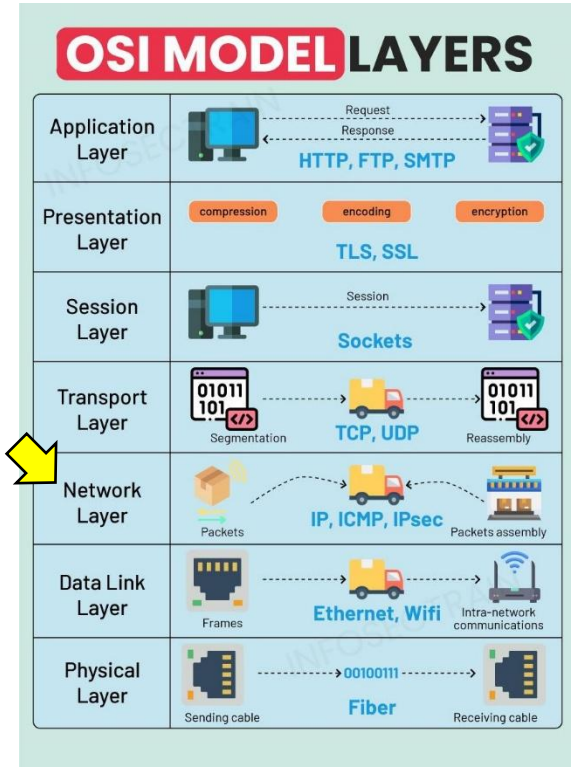
이종 클라우드에 안정적인 네트워크 기반 구축?!

(i.e., Secure: 안전한, 확실한)

Security 및 Lack of resources/expertise 우려 해소에 도움이 되는 네트워크 기술을 소개 드립니다 ☺

✓ CSP의 네트워크 특성/요구사항 사전 검증 ✓ 확실한 멀티 클라우드 네트워크 구축 ✓ CSP간 안전한 연결성 지원 ✓ (번외편) 멀티 클라우드에 안전한 연결 및 운영/관리





https://www.infosectrain.com/blog/osi-model-a-comprehensive-guide-for-exam-and-interview/

https://learn.microsoft.com/en-us/azure/architecture/reference-architectures/n-tier/windows-vm
https://docs.aws.amazon.com/vpc/latest/userguide/vpc-example-private-subnets-nat.html
https://blog.cloud63.fr/multicloud-azure-aws-connectivity-using-vpn-and-bgp/

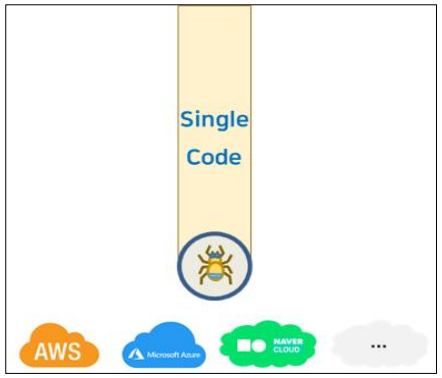
General Purpose APIs

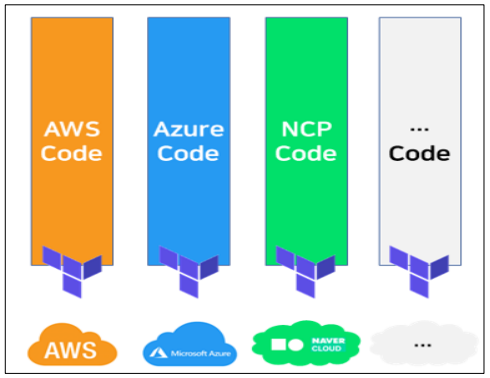
Synergy !!!

Specific Purpose IaCs

Cloud Barista

OpenTofu





* API: Application Programming Interface / IaC: Infrastructure as Code

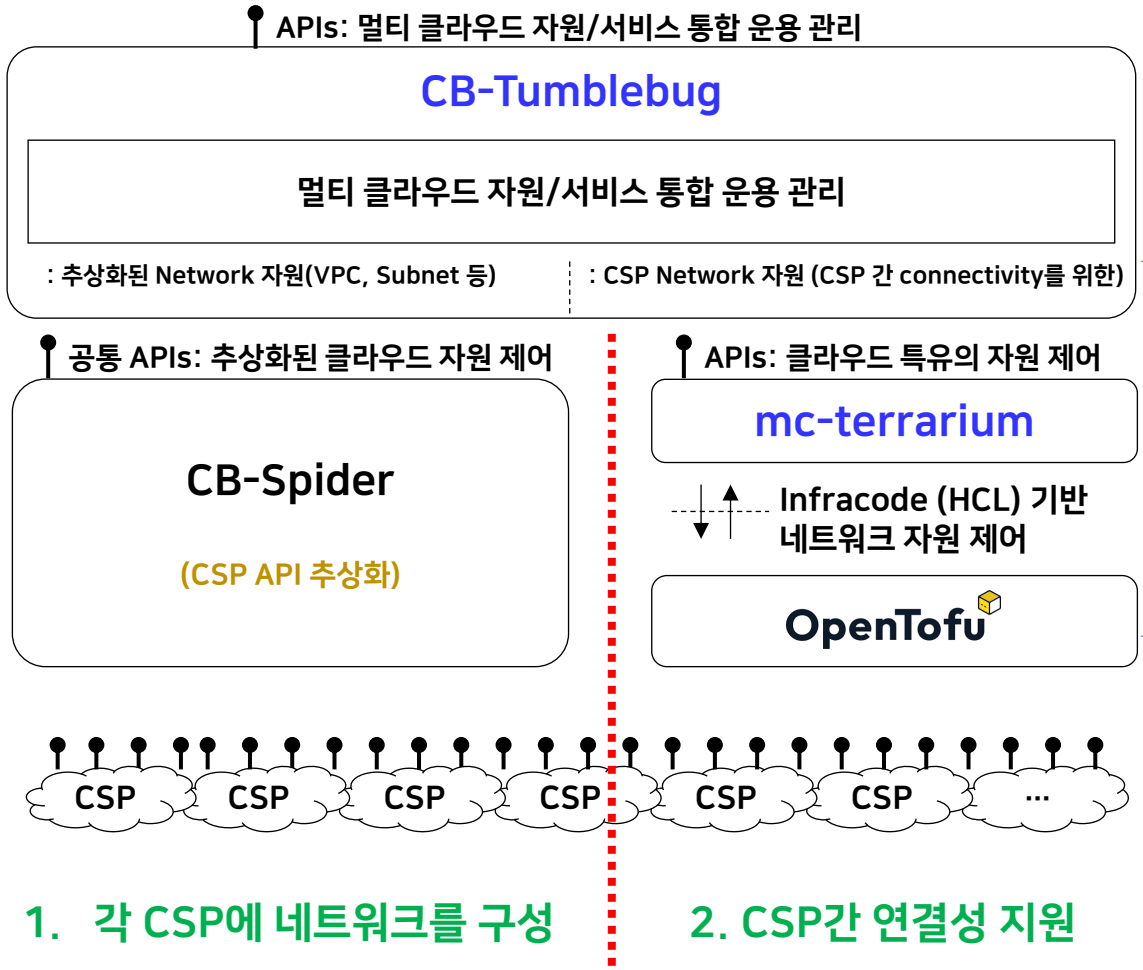
네트워크 자원의 공통 확장과 개별 확장

General Purpose APIs

CLOUD BARISTA

- Supported Computing Infrastructure Resources
  - **Basic Resources**: Public Image, VM Spec, VPC/Subnet, Security Group, VM KeyPair
  - **VM Infrastructures**: VM, NLB(Network Load Balancer), Disk, MyImage
  - **Container Infrastructures**: PMKS(Provider-Managed K8S)
- Supported CloudOS:

Provider, CloudOS	CloudOS Constant	Cloud Driver Lib.	Etc
Amazon Web Services	AWS	aws-driver-v1.0.so	
Microsoft Azure	AZURE	azure-driver-v1.0.so	
Google Cloud Platform	GCP	gcp-driver-v1.0.so	
Alibaba Cloud	ALIBABA	alibaba-driver-v1.0.so	
Tencent Cloud	TENCENT	tencent-driver-v1.0.so	
IBM VPC Cloud	IBM	ibmvpc-driver-v1.0.so	
OpenStack Platform	OPENSTACK	openstack-driver-v1.0.so	
NCP Classic Cloud	NCP	ncp-driver-v1.0.so	
NCP VPC Cloud	NCPVPC	ncpvpc-driver-v1.0.so	
NHN Cloud	NHNCLOUD	nhncloud-driver-v1.0.so	
KT Classic Cloud	KTCLOUD	ktcloud-driver-v1.0.so	
KT VPC Cloud	KTCLOUDVPC	ktcloudvpc-driver-v1.0.so	



Specific Purpose IaCs

(API를 곁들인 ^^;;)

OpenTofu

개별 확장 = 자원 보강

목표: Site-to-site VPN

(As-Is) Supported AWS-to-site VPN
- Hub site: AWS
- Spoke site: Azure, GCP, Alibaba, Tencent, IBM

Source:
- CB-Spider, "Supported Computing Infrastructure Resources", (accessed on 2024-03-14, https://github.com/cloud-barista/cb-spider/wiki/Supported-Compute-Infrastructure-Resources#supported-computing-infrastructure-resources)
- CB-Spider, "Supported CloudOS", (accessed on 2024-03-14, https://github.com/cloud-barista/cb-spider/wiki/Supported-CloudOS#supported-cloudos)

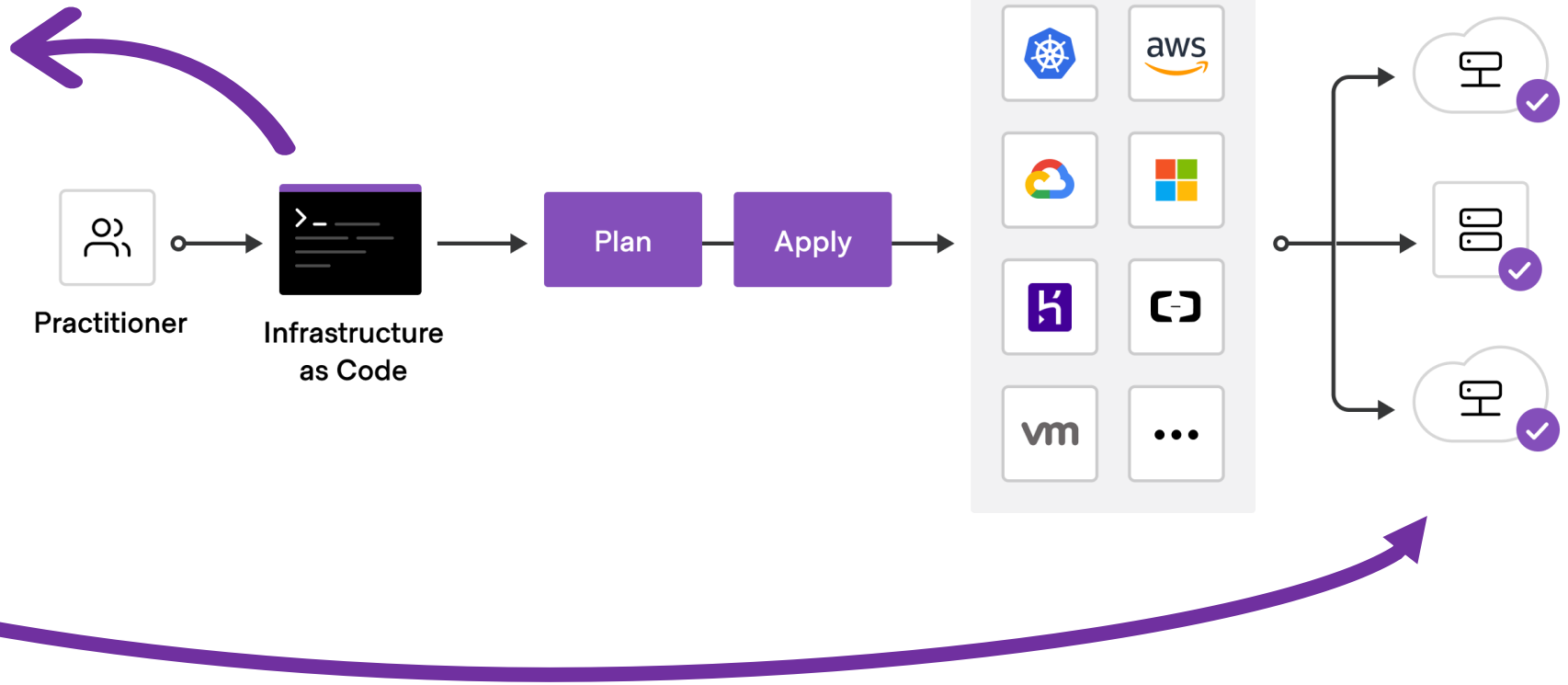
참고

```
terraform {
  required_providers {
    aws = {
      source  = "hashicorp/aws"
      version = "~> 4.16"
    }
  }

  required_version = ">= 1.2.0"
}

provider "aws" {
  region  = "us-west-2"
}

resource "aws_instance" "app_server" {
  ami           = "ami-830c94e3"
  instance_type = "t2.micro"

  tags = {
    Name = "ExampleAppServerInstance"
  }
}
```

HashiCorp Terraform

Terraform is HashiCorp's Infrastructure as Code (IaC) tool.

It allows you to manage infrastructure with configuration files rather than through a graphical user interface.

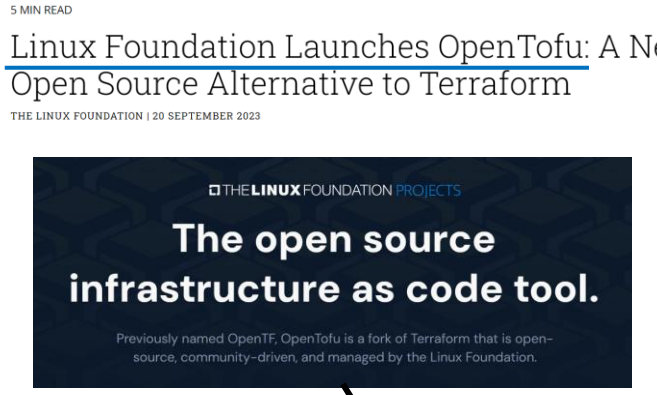


https://developer.hashicorp.com/terraform/tutorials/aws-get-started/infrastructure-as-code

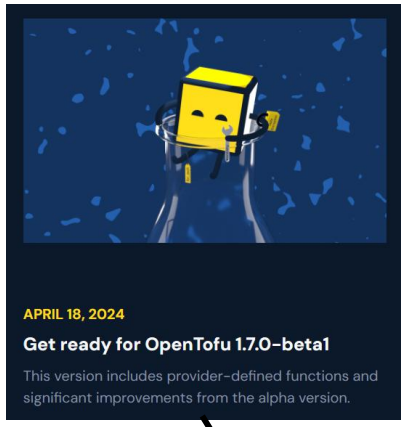
참고

MPL-2.0 license

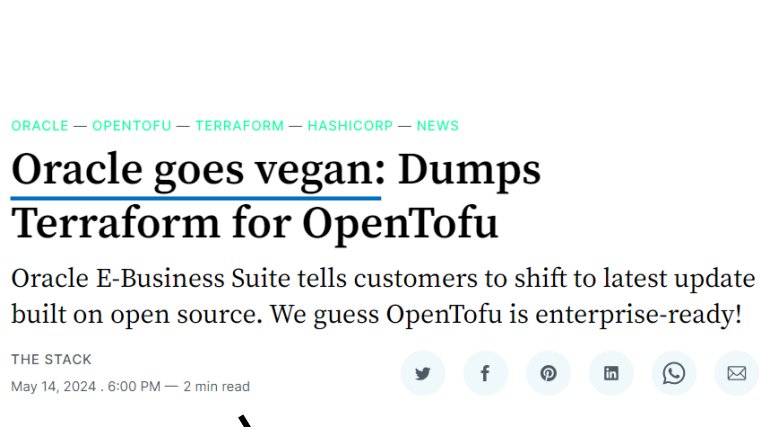
2023.08.10



2023.08.25



2023.09.20

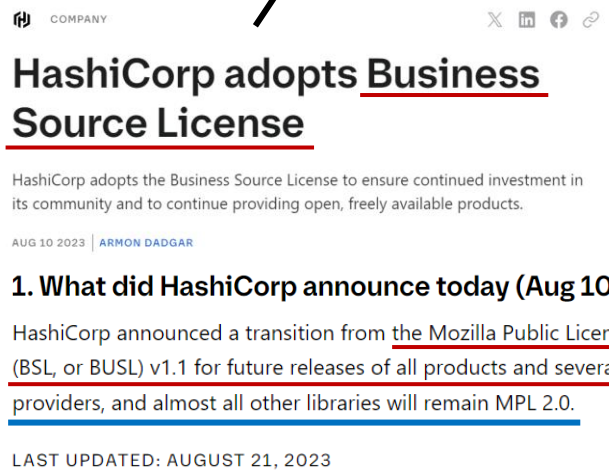


2024.05.14

2024.04.18

2024.04.24

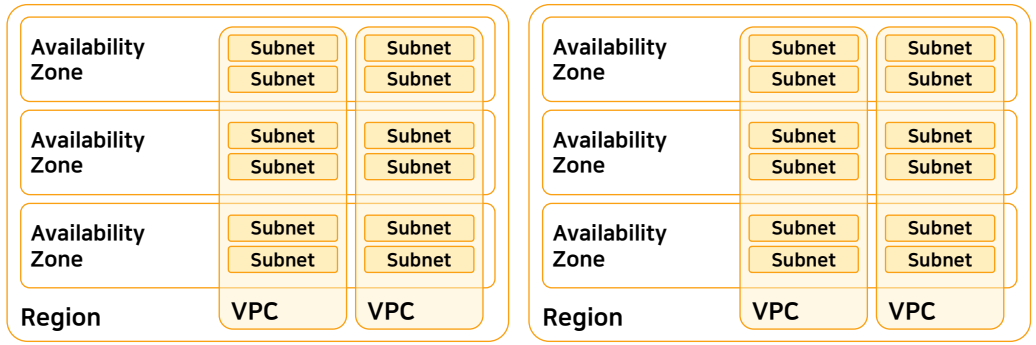
2023.10.27



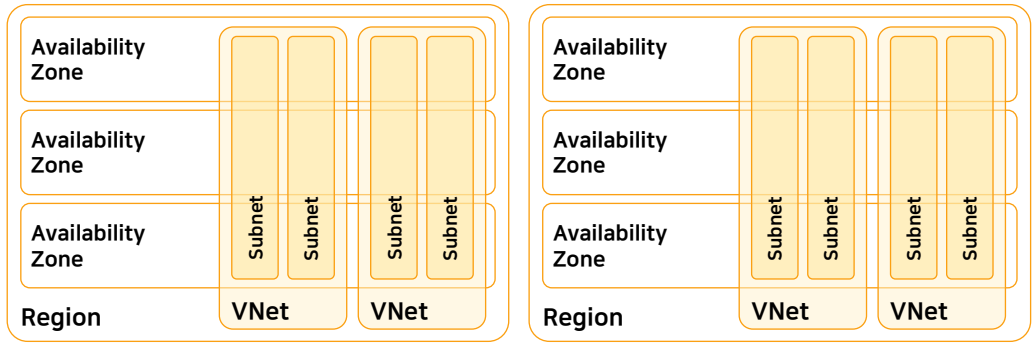
https://www.hashicorp.com/blog/hashicorp-adopts-business-source-license
https://www.hashicorp.com/license-faq
https://opentofu.org/blog/opentofu-announces-fork-of-terraform/
https://github.com/opentofu/opentofu/issues/296
https://www.linuxfoundation.org/press/announcing-opentofu
https://newsroom.ibm.com/2024-04-24-IBM-to-Acquire-HashiCorp-Inc-Creating-a-Comprehensive-End-to-End-Hybrid-Cloud-Platform
https://www.thestack.technology/oracle-dumps-terraform-for-opentofu/

클라우드 사업자의 네트워크 구조적 특성/차이점

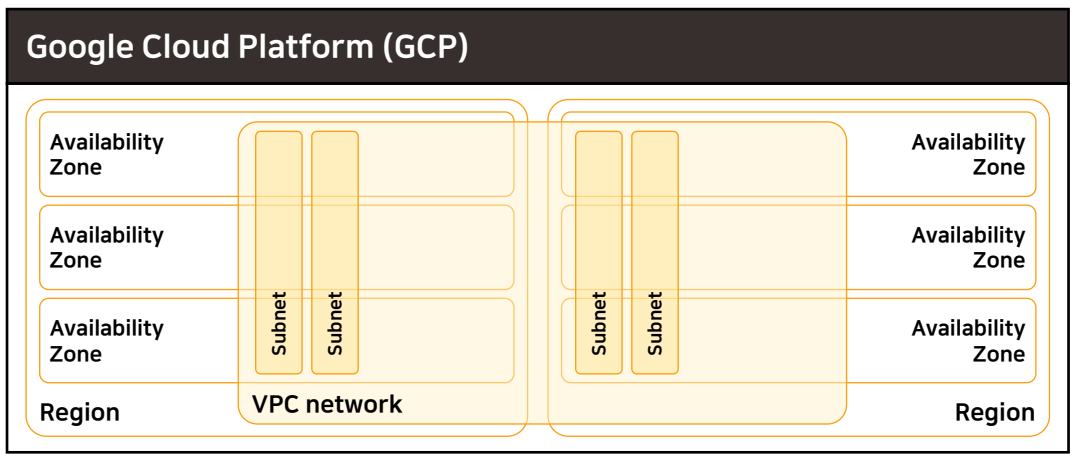
Amazon Web Services (AWS) / Alibaba Cloud / Tencent Cloud



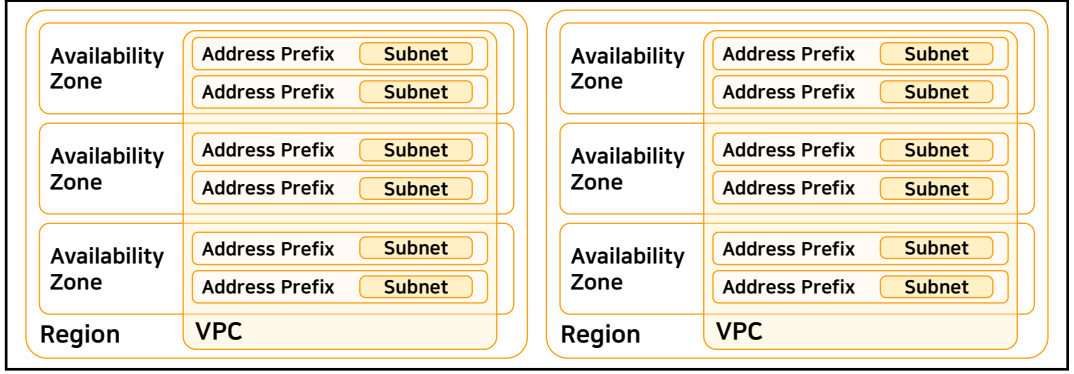
Microsoft Azure (MS Azure)



Google Cloud Platform (GCP)



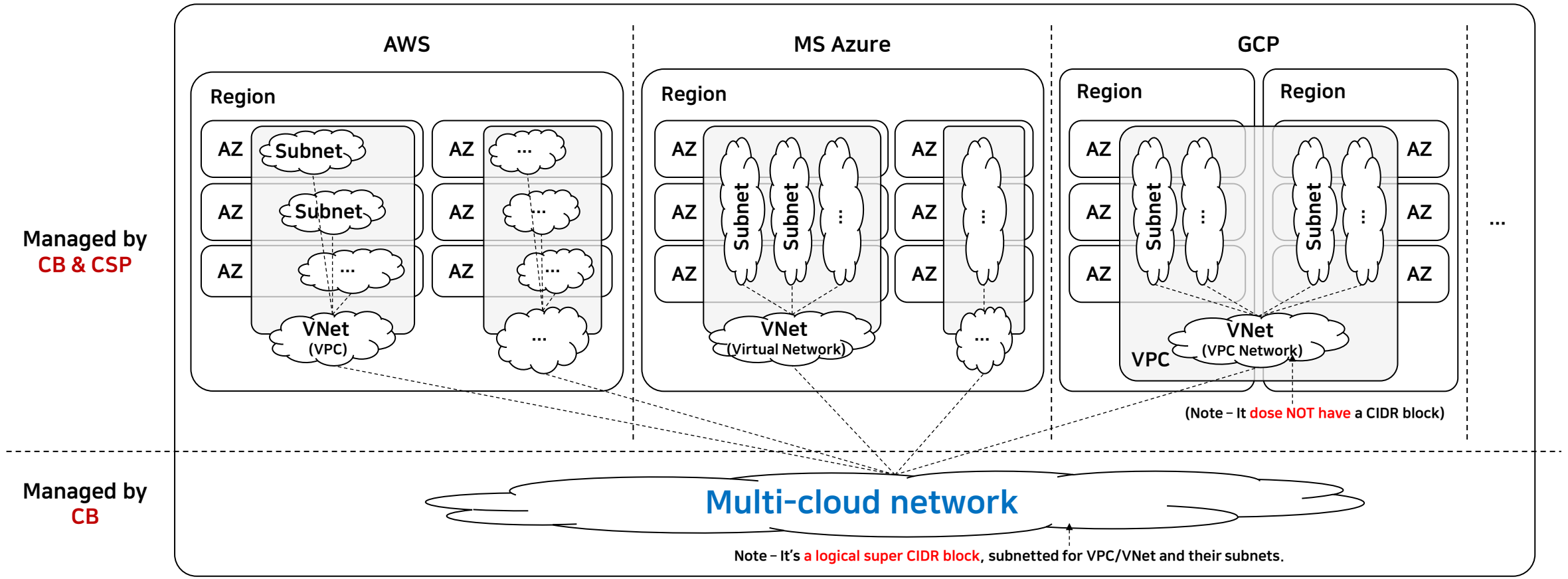
IBM Cloud



출처:

ipSpace.net, "Virtual Networks and Subnets in AWS, Azure, and GCP", 2021 (accessed on 2022-01-25 https://blog.ipspace.net/2021/02/vpc-subnets-aws-azure-gcp.html)
ipSpace.net, "Availability Zones and Regions in AWS, Azure and GCP" 2021 (accessed on 2022-01-25 https://blog.ipspace.net/2021/02/public-cloud-regions-availability-zones.html)
Mate Gulic, "AWS, Azure, GCP: Virtual Networking Concepts overview", 2020 (accessed on 2022-01-25 https://www.linkedin.com/pulse/aws-azure-gcp-virtual-networking-concepts-overview-mate-gulic/)
Alibaba Cloud, "Overview – VPCs and vSwitches", (accessed on 2022-01-26 https://www.alibabacloud.com/help/en/doc-detail/100380.htm)
IBM Cloud, "About networking", 2020 (accessed on 2022-01-26 https://cloud.ibm.com/docs/vpc?topic=vpc-about-networking-for-vpc)
Tencent Cloud, "Virtual Private Cloud Product Introduction Product Documentation", 2020. (accessed on 2022-01-26 https://main.qcloudimg.com/raw/document/intl/product/pdf/215_532_en.pdf)

멀티 클라우드 인프라 측면에서 바라본 네트워크 구조는?



* VNet: Virtual Private Cloud (VPC) / Virtual Network / VPC network
* AZ: Availability Zone

Multi-Cloud Infrastructure (MCI)

CLOUD BARISTA

각 CSP의 네트워크 구성설정적 특성/요구사항이 상이함

(i.e., configuration)

각 CSP에 네트워크를 구축 시 입력할 수 있는 구성설정 정보(2025년 4월 조사됨)

	AWS	Azure	GCP	Alibaba	Tencent	IBM	NCP	NHNCloud	Open Stack	KTCloud
Available CIDR blocks	* 10.0.0.0/8 * 172.16.0.0/12 * 192.168.0.0/16	* 10.0.0.0/8 * 172.16.0.0/12 * 192.168.0.0/16	* 10.0.0.0/8 * 172.16.0.0/12 * 192.168.0.0/16	* 10.0.0.0/8 * 172.16.0.0/12 * 192.168.0.0/16	* 10.0.0.0/8 * 172.16.0.0/12 * 192.168.0.0/16	* 10.0.0.0/8 * 172.16.0.0/12 * 192.168.0.0/16	* 10.0.0.0/8 * 172.16.0.0/12 * 192.168.0.0/16	* 10.0.0.0/8 * 172.16.0.0/12 * 192.168.0.0/16	TBD	정보 불충분...
Reserved CIDR blocks	* 172.17.0.0/16	-	* 172.17.0.0/16	-	-	-	-	-	TBD	정보 불충분...
Prefix range for VNet	/16 ~ /28	/8 ~ /29	-	/16 ~ /24	/12 ~ /28	/9 ~ /28	/16 ~ /28	/8 ~ /24	TBD	정보 불충분...
Prefix range for subnet	/16 ~ /28	/8 ~ /29	/8 ~ /29	/16 ~ /29	/12 ~ /29	/9 ~ /29	/16 ~ /28	/8 ~ /28	TBD	정보 불충분...
The number of reserved IPs	5	5	4	4	3	5	5	5	TBD	정보 불충분...
VPN	-	* GatewaySubnet * ~ /27	-	-	-	-	-	-	TBD	정보 불충분...

➡ 각 CSP의 네트워크 구성설정이 다름, 그래서! 멀티 클라우드 생성시 이를 잘 알고 활용해야 합니다.

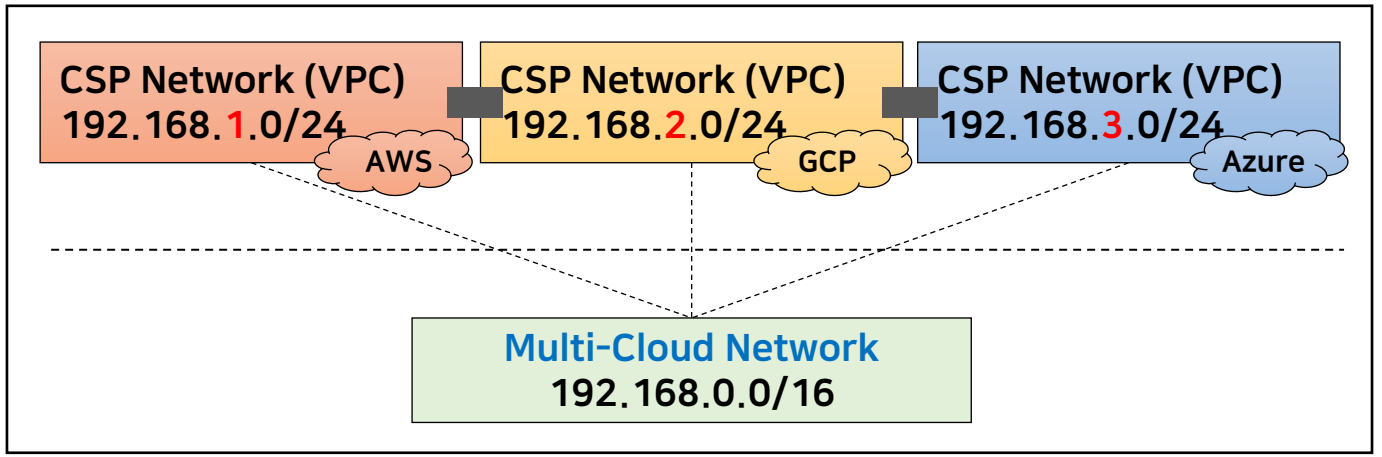
➡ 모두 알고 사용하기는 어렵겠죠? ^^;; 그래서! CB-Tumblebug에서 설계, 검증, 배치할 수 있는 기능을 제공 합니다.

멀티 클라우드 네트워크 설계 및 관리 기능/API 제공

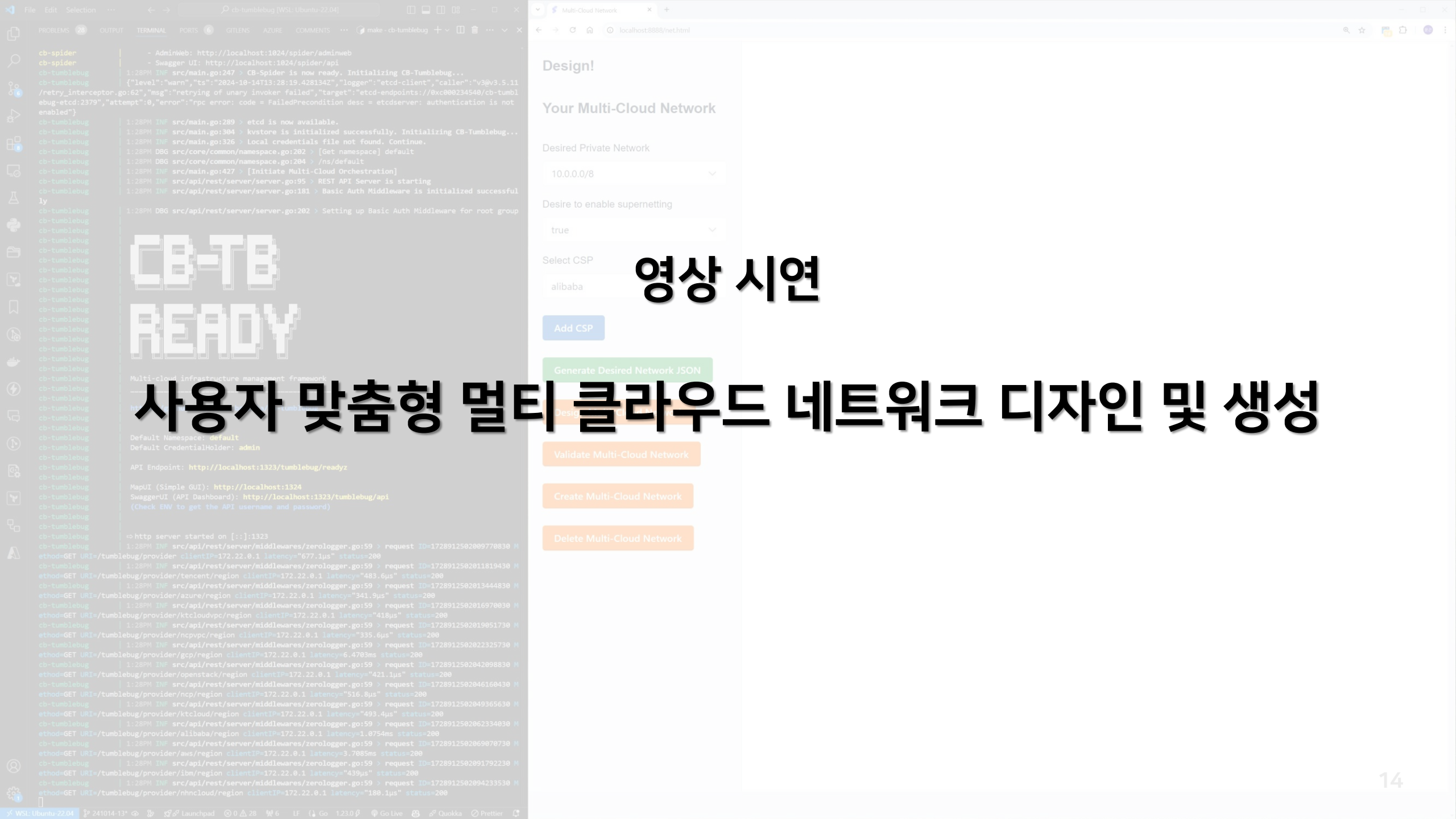
- vNet/Subnet 생성 전, 정보 기반의 사전 검증 수행
  - 예) CIDR blocks 검증, regions/zones 검증,
    vNet 생성 시 최소 1 subnet 필요 조건 검증, 등
- vNet/Subnet CR(U)D 및 관리 기능/API 제공
  - 자원의 CR(U)D 및 정보를 나타내는 Object (metadata) 기반 관리 수행
  - 예) CR(U)D 상태 관리, 실패 시 Object 삭제, 동시성 처리 지원 등
- 네트워크 설계 및 검증 유틸리티 기능/API 제공
  - vNet 설계 기능/API 개선
- (참고) API 사용 방법:
  - https://github.com/cloud-barista/cb-tumblebug/discussions/1783

IMHO: 멀티 클라우드 네트워크, 초기에 확실하게 구축할수록 좋습니다 ☺

MCI Network address space * Consider supernetting



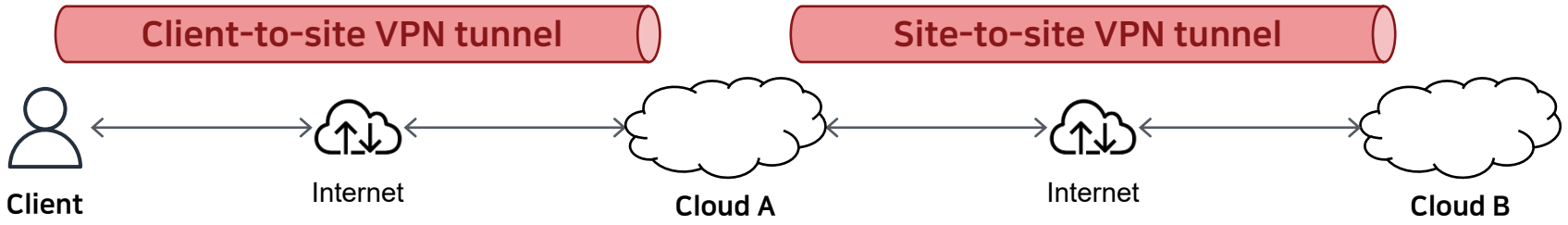
1. 각 CSP에 확실한 네트워크 구성
2. CSP간 안전한 연결성 지원 (이후 설명)


영상 시연
사용자 맞춤형 멀티 클라우드 네트워크 디자인 및 생성
CB-TB
READY
Design!
Your Multi-Cloud Network
Desired Private Network
10.0.0.0/8
Desire to enable supernetting
true
Select CSP
alibaba
Add CSP
Generate Desired Network JSON
Validate Multi-Cloud Network
Create Multi-Cloud Network
Delete Multi-Cloud Network

안정적인 네트워크 연결성 지원

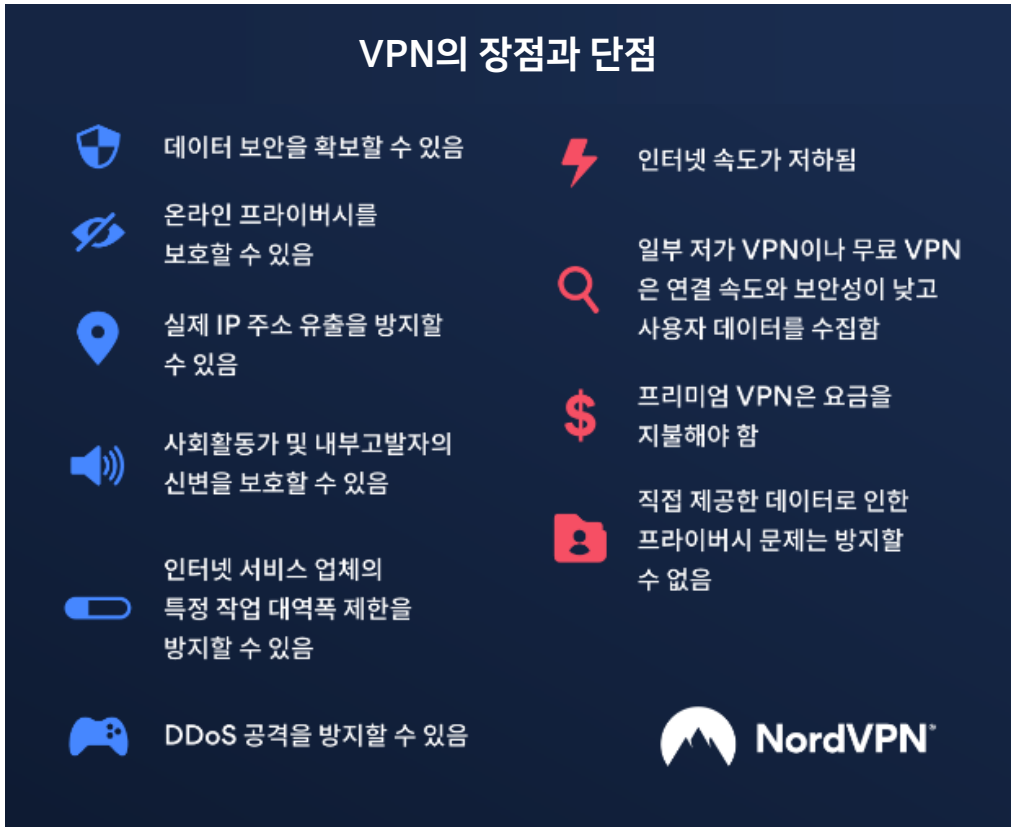
(i.e., Secure: 안전한, 확실한)

2가지 종류의 VPN

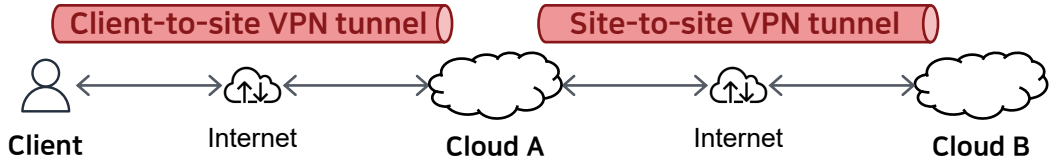


Virtual Private Network (VPN) 요약

#Security
#Application #IPsec #WireGuard
#Layer3 #NetworkLayer

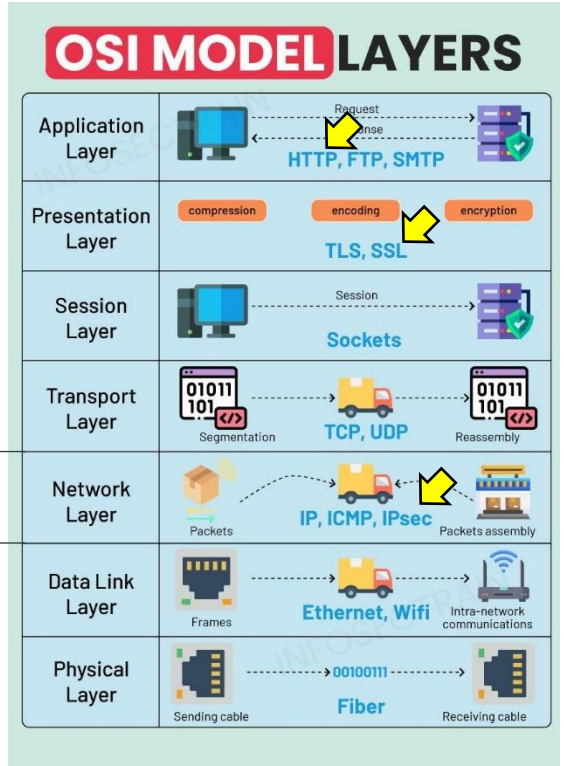


https://nordvpn.com/ko/blog/vpn-pros-and-cons/



WireGuard is a secure network tunnel, operating at layer 3

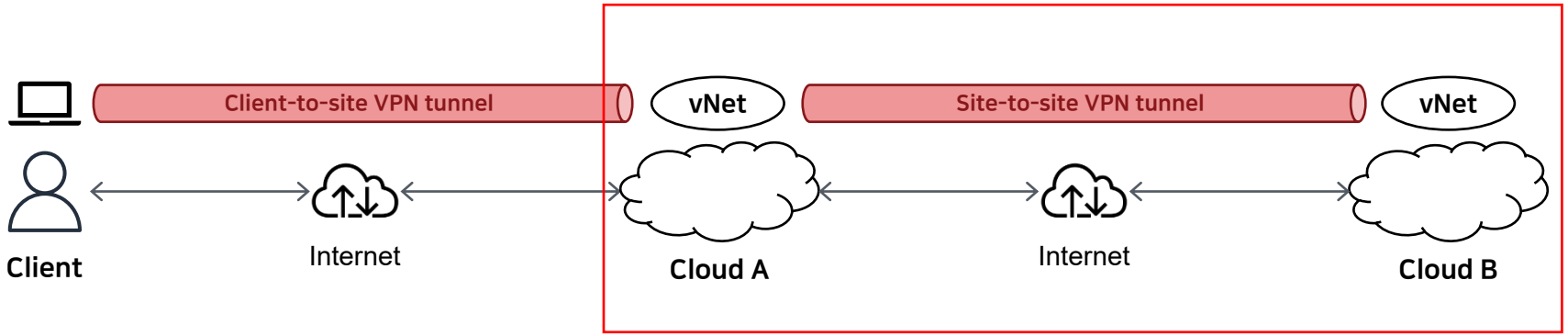
https://www.wireguard.com/papers/wireguard.pdf



https://www.infosectrain.com/blog/osi-model-a-comprehensive-guide-for-exam-and-interview/

CSP간 안전한 연결성 지원

Site-to-site VPN 개요 및 현황 (i.e., IPsec VPN)



(현황) AWS-to-site VPN 기능/API 제공

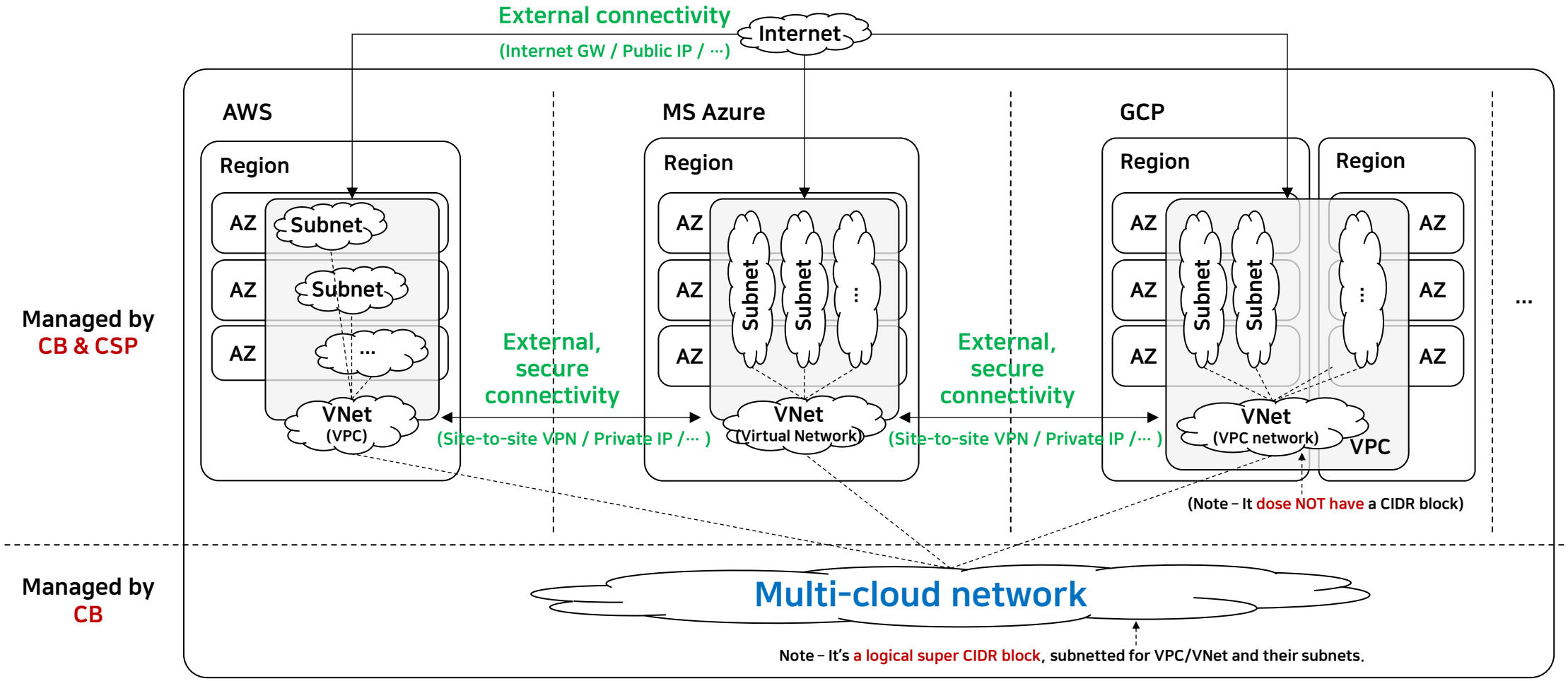
1. AWS에서 하나의 다른 CSP로 1:1 VPN 연결을 지원하기 위함
2. 시장 점유율을 고려하여 AWS-to-site VPN 을 우선적으로 구현

(효과)

1. AWS와 하나의 다른 CSP간 안전한 연결 및 통신 기대
2. 두 CSP간 사설망을 구성하여 하나의 네트워크/인프라 처럼 활용 가능

이종 CSP 인프라간 네트워크 연결성 지원

(e.g., AWS, Azure, GCP, and …)

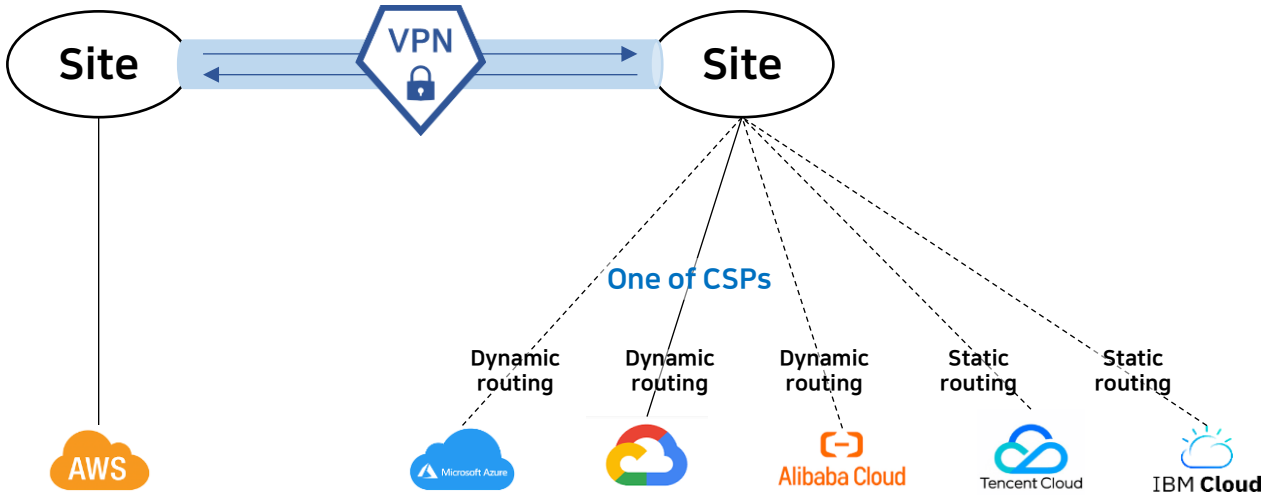


* VNet: Virtual Private Cloud / Virtual Network / VPC network
* AZ: Availability Zone

Multi-Cloud Infrastructure (MCI)

AWS-to-site VPN (on Terrarium)

- AWS-to-site VPN: A feature to configure a VPN between AWS and one of CSPs
  - Supported CSPs: MS Azure, GCP, Alibaba Cloud, Tencent Cloud, IBM Cloud
  - Not available CSPs: NCP, NHN Cloud, KT Cloud (Note - Korean CSPs doesn't provide APIs for VPN…)



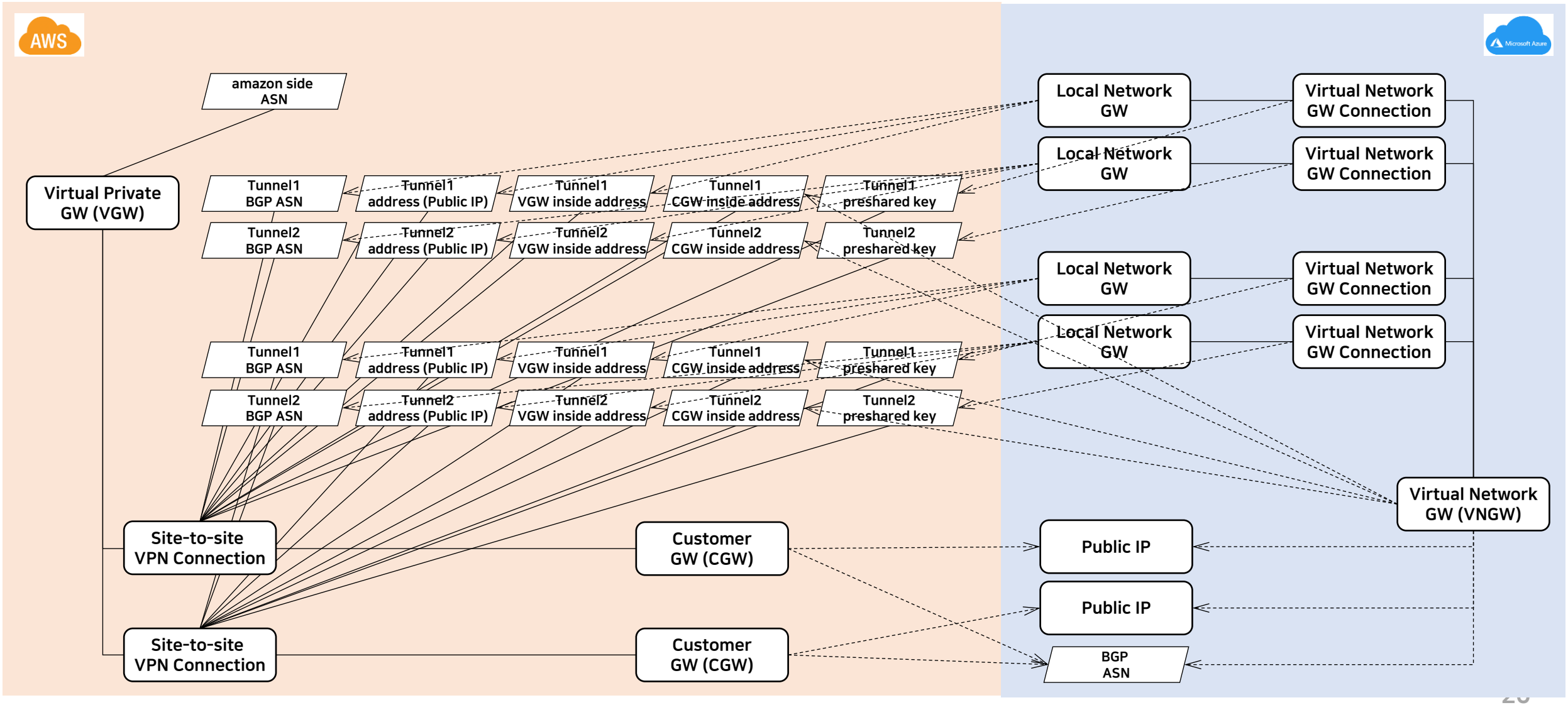
NAVER Cloud NHN Cloud kt cloud openstack

OpenTofu providers of the CSPs does NOT support the resource blocks for VPN

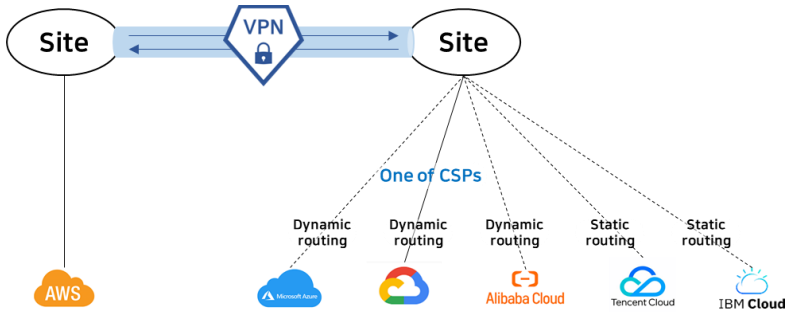
(예시) AWS와 Azure간에 VPN을 구성하려면…

#actually…
#chaos…

(They are all different…)



Objectives of site-to-site VPN in Cloud-Barista



Choose “combo” or “all the way”

— made with care, balanced just right.

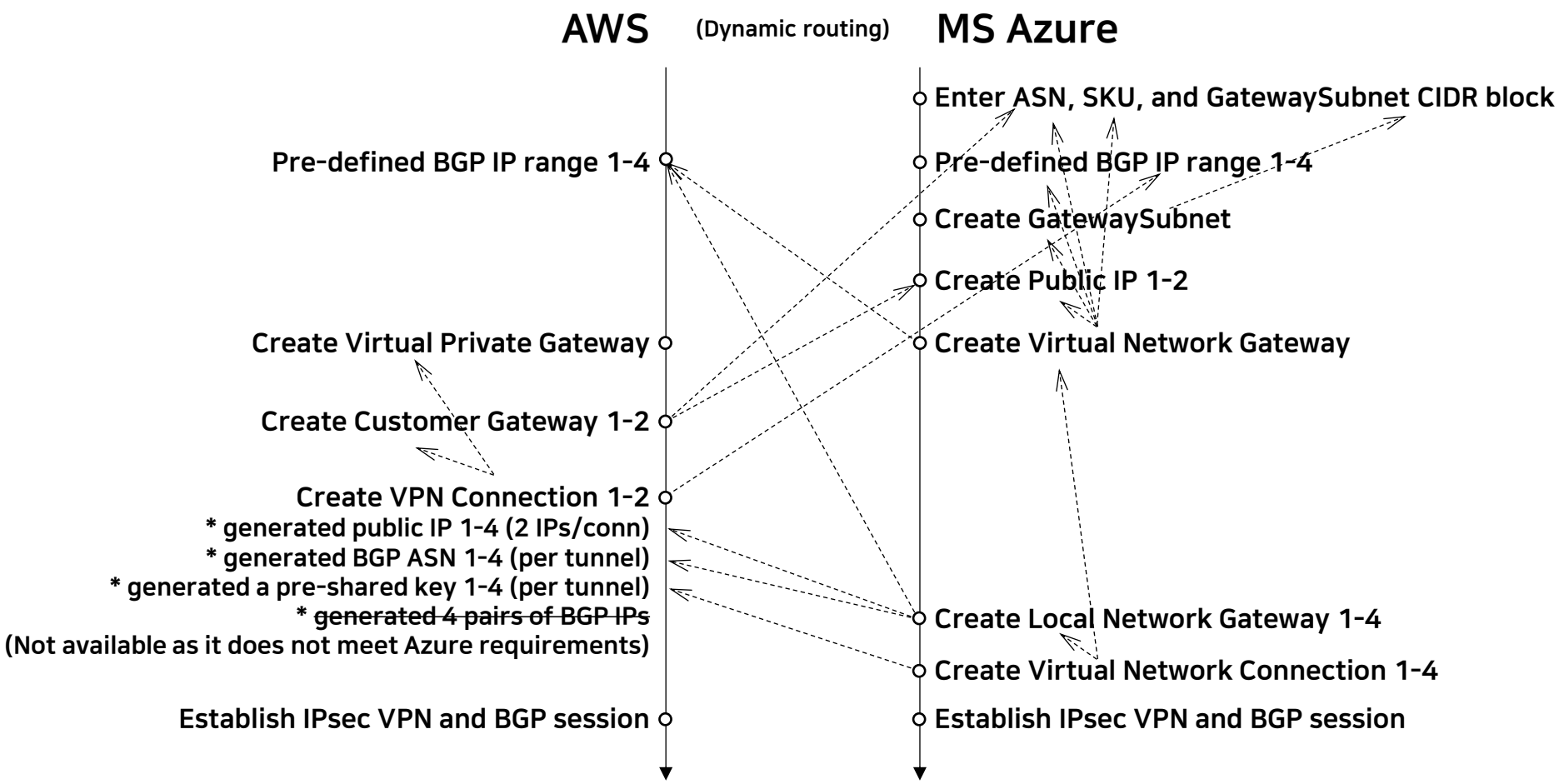
Configure your site-to-site VPN like placing an order

— we’ll handle what’s under the hood to ensure maturity and stability

➡ mc-terrarium makes it possible!

AWS to Azure VPN Configuration Process and Dependency

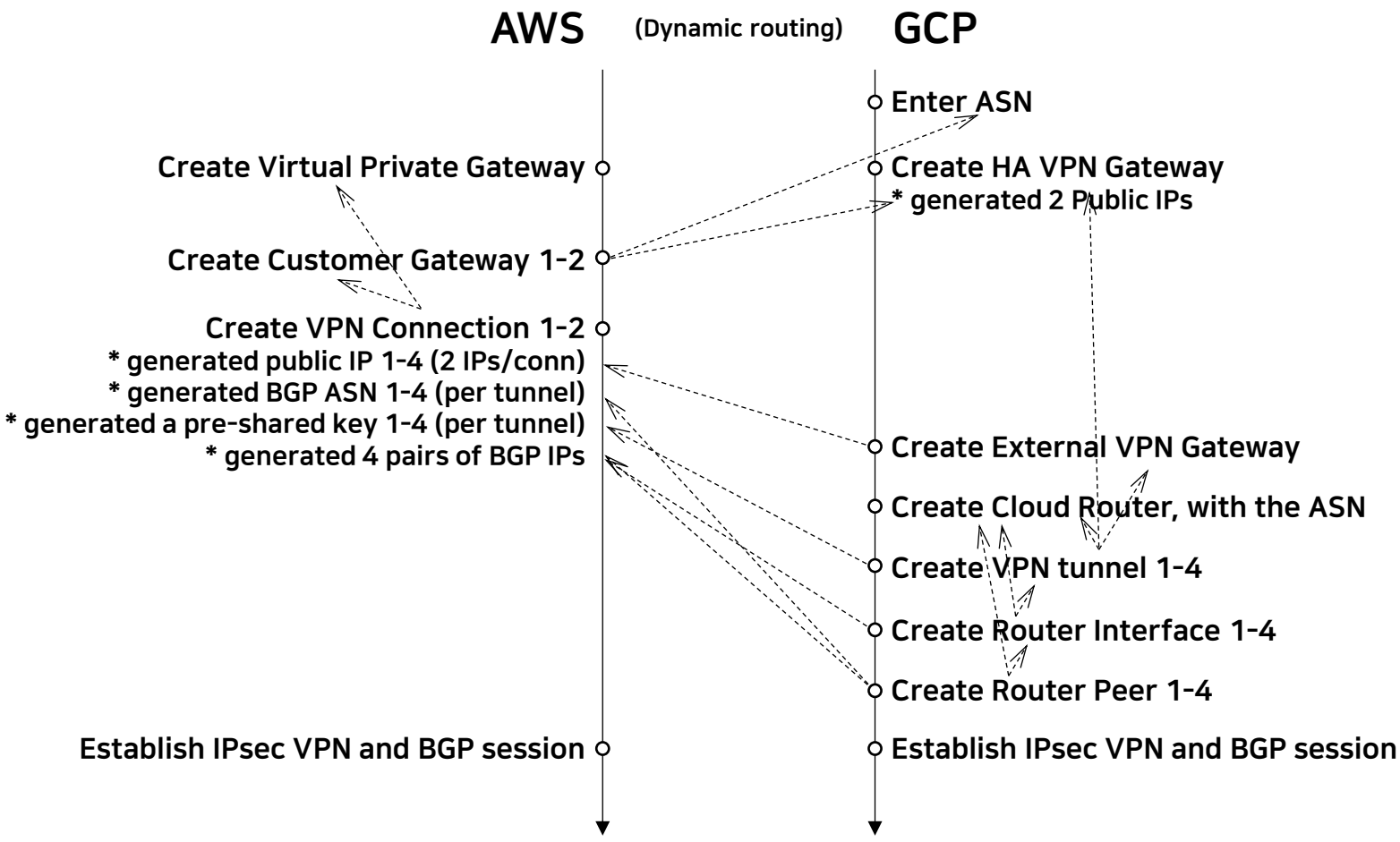
Note - Metadata of existing resources have been excluded as much as possible.



* ⟶: Time
* VPN: Virtual Private Network
* ASN: Autonomous System Number
* ⇢: Dependency
* SKU: Stock Keeping Unit (리소스/서비스의 성능 또는 기능별 상품 옵션을 구분하는 단위)
* BGP: Border Gateway Protocol

AWS to GCP VPN Configuration Process and Dependency

Note - Metadata of existing resources have been minimized as much as possible.

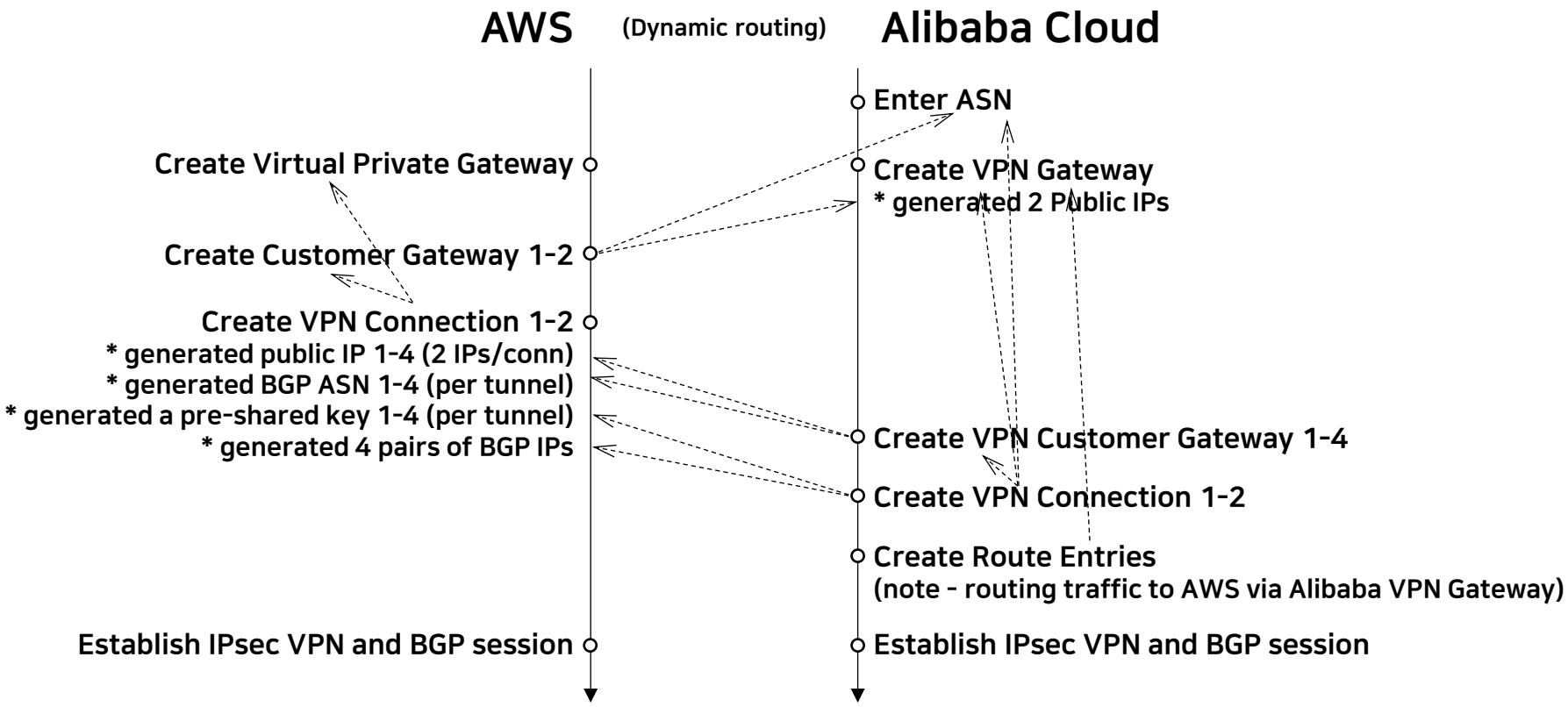


* ⟶: Time
* VPN: Virtual Private Network
* ASN: Autonomous System Number

* ⇢: Dependency
* BGP: Border Gateway Protocol

AWS to Alibaba VPN Configuration Process and Dependency

Note - Metadata of existing resources have been excluded as much as possible.

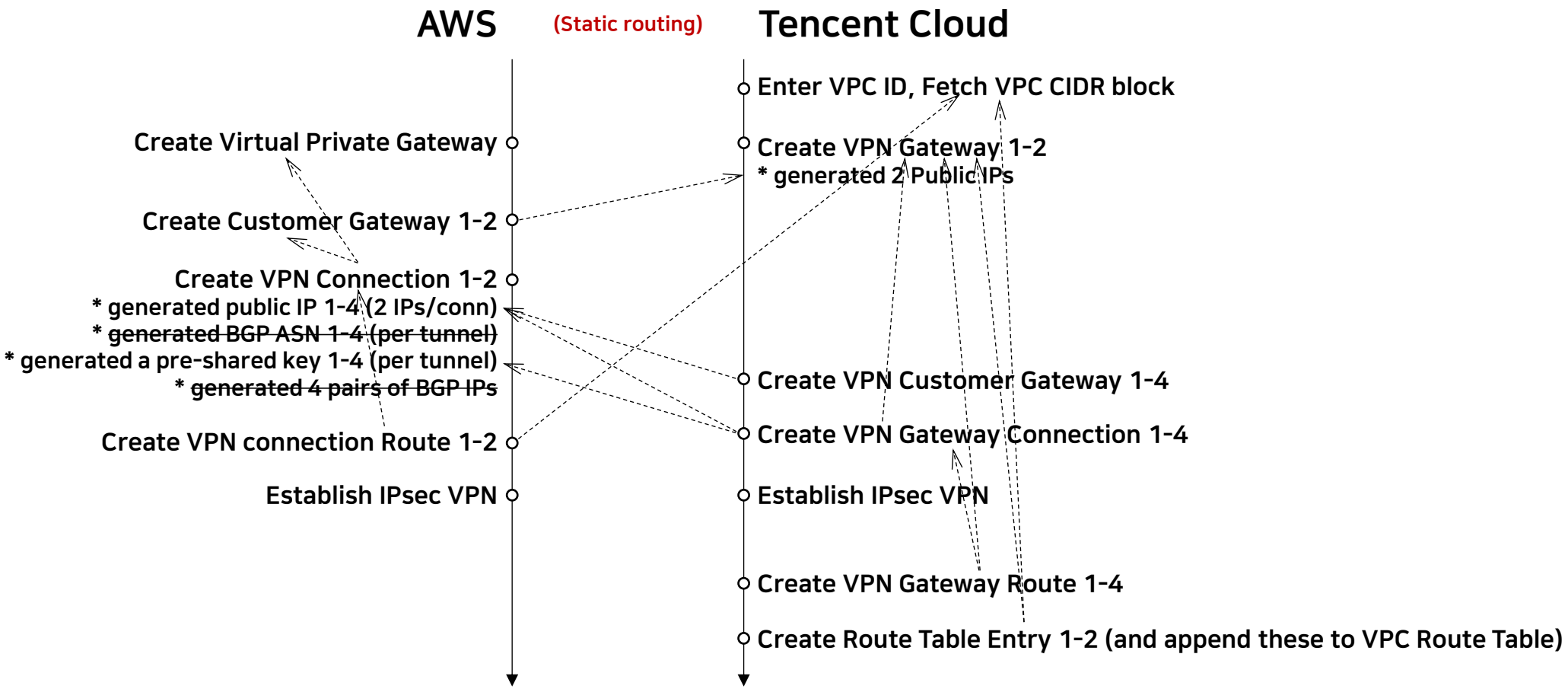


* ⟶: Time
* VPN: Virtual Private Network
* ASN: Autonomous System Number

* ⇢: Dependency
* BGP: Border Gateway Protocol

AWS to Tencent VPN Configuration Process and Dependency

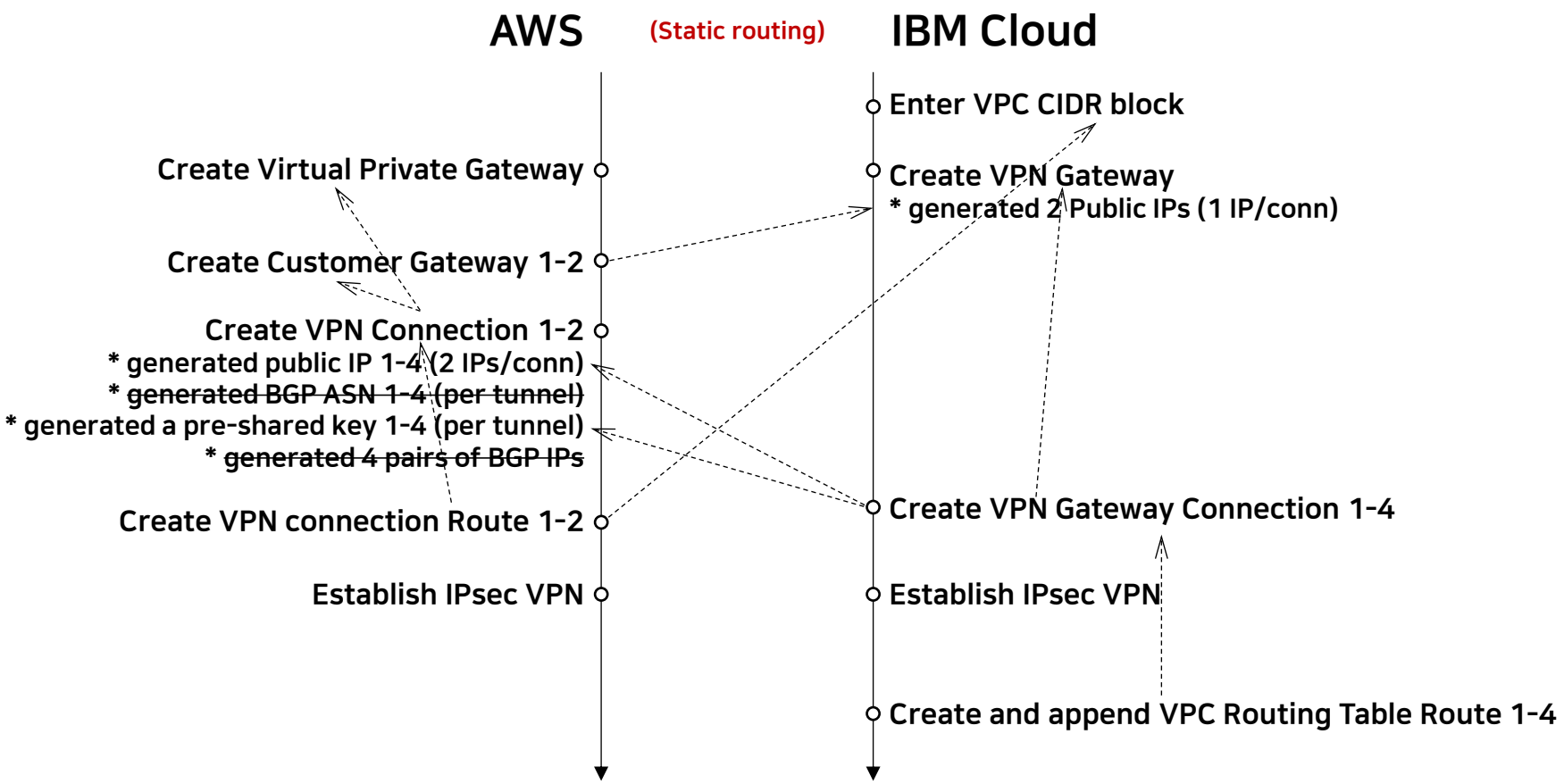
Note - Metadata of existing resources have been excluded as much as possible.



* ⟶: Time
* ⇢: Dependency
* VPN: Virtual Private Network

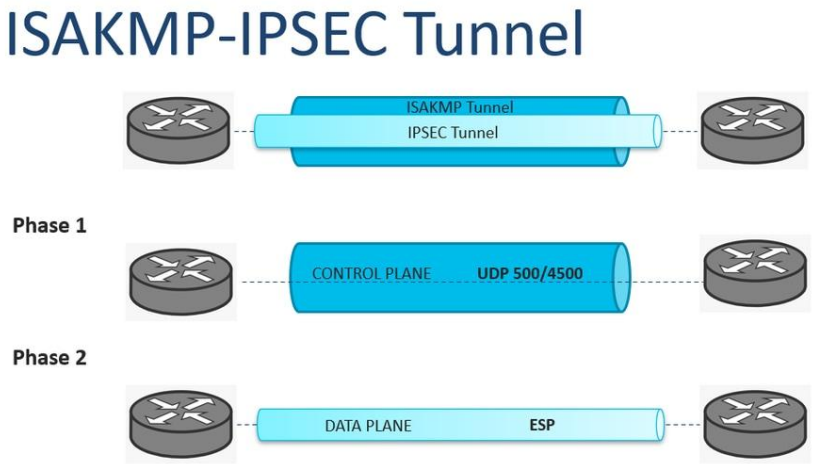
AWS to IBM VPN Configuration Process and Dependency

Note - Metadata of existing resources have been excluded as much as possible.



* ⟶: Time
* ⇢: Dependency
* VPN: Virtual Private Network

IPsec negotiation phases and algorithms



Note: Phase 1 (ISAKMP) Tunnel protects the Control Plante VPN traffic between the two gateways. Control Plane traffic can be Negotiation packets, information packages, DPD, keepalives, rekey, and so on. ISAKMP negotiation uses the UDP 500 and 4500 ports to establish a secure channel.

Note: Phase 2 (IPsec) Tunnel protects the Data Plane traffic that passes through the VPN between the two gateways. The algorithms used to protect the data are configured in Phase 2 and are independent of those specified in Phase 1.
The protocol used to encapsulate and encrypt these packets is the Encapsulation Security Payload (ESP).

IPsec Tunnel Establishment Phases:

IKE Phase 1
=ISAKMP SA
ISAKMP SA를 협상하여 안전한 터널을 생성하고 상대방을 인증

IKE Phase 2
=IPsec SA
Phase 1에서 생성된 안전한 터널을 기반으로 패킷을 암호화/인증할 실질적인 SA를 협의하고 패킷을 암호화하여 전송할 터널을 생성

* IPsec: Internet Protocol Security / IKE: Internet Key Exchange / ISAKMP: Internet Security Association and Key Management Protocol / SA: Security Association

https://www.cisco.com/c/en/us/support/docs/security-vpn/ipsec-negotiation-ike-protocols/217432-understand-ipsec-ikev1-protocol.html

참고

IPsec negotiation phases and algorithms in CSP VPNs (Cont'd)

* ISAKMP: Internet Security Association and Key Management Protocol + Security Association
* IPsec SA: Internet Protocol Security + Security Association

Representative Name Candidate	Mutually Supported Algorithms
Note	Note for tunnels * Link-local addresses: 169.254.0.0/16 * Inside IPv4 CIDR for tunnels: A size /30 IPv4 CIDR block from the 169.254.0.0/16 range. * Pre-shared key for tunnel: generated by some CSP or entered by a user * IKE version: ikev2
IPsec negotiation phase 1 = IKE phase 1 = ISAKMP SA	* IKE phase 1(ISAKMP SA) - Encryption Algorithm: AES128 | AES256 — 2종 - Integrity Algorithm: SHA2-256 — 1종 - Diffie-Hellman Group Number: 14 — 1종 - Lifetime (seconds): 28800 (note: 8h)
IPsec negotiation phase 2 = IKE phase 2 = IPsec SA	* IKE phase 2(IPsec SA) - Encryption Algorithm: AES128 | AES256 — 2종 - Integrity Algorithm: SHA2-256 — 1종 - Diffie-Hellman Group Number: 14 — 1종 - lifetime (seconds): 3600 (note: 1h)
ETC	
References	

Representative Name Candidate	AWS	Azure
Note	Be set in the VPN Connection (Tunnel options) * Inside IPv4 CIDR for tunnel x: Generated by Amazon, A size /30 IPv4 CIDR block from the 169.254.0.0/16 range. * Pre-shared key for tunnel x: Generated by Amazon, The pre-shared key must have 8-64 characters. Valid characters: A-Z, a-z, 0-9, _ and . The key cannot begin with a zero. * IKE version: ikev1 | ikev2	Be set in Virutal Network Gateway Connection
IPsec negotiation phase 1 = IKE phase 1 = ISAKMP SA	* IKE phase 1(ISAKMP SA) - Encryption Algorithm: AES128 | AES256 | AES128-GCM-16 | AES256-GCM-16 — 4종 - Integrity Algorithm: SHA1 | SHA2-256 | SHA2-384 | SHA2-512 — 4종 - Diffie-Hellman Group Number: 2 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 — 12종 - Lifetime (seconds): (Default 28800) 900 ~ 28800	(IPsec/IKE connection policies) * Cryptographic algorithms & key strengths - IKEv2 encryption : GCMAES256, GCMAES128, AES256, AES192, AES128 — 5종 - IKEv2 integrity: SHA384, SHA256, SHA1, MD5 — 4종 - DH group: DHGroup24, ECP384(DHGroup20), ECP256(DHGroup19), DHGroup14, DHGroup2048(DHGroup14), DHGroup2, DHGroup1, None — 8종
IPsec negotiation phase 2 = IKE phase 2 = IPsec SA	* IKE phase 2(IPsec SA) - Encryption Algorithm: AES128 | AES256 | AES128-GCM-16 | AES256-GCM-16 — 4종 - Integrity Algorithm: SHA1 | SHA2-256 | SHA2-384 | SHA2-512 — 4종 - Diffie-Hellman Group Number: 2 | 5 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 — 13종 - lifetime (seconds): (Default 360) 900 ~ 3600	(IPsec/IKE connection policies) * Cryptographic algorithms & key strengths - IPsec encryption: GCMAES256, GCMAES192, GCMAES128, AES256, AES192, AES128, DES3, DES, None — 9종 - IPsec integrity: GCMAES256, GCMAES192, GCMAES128, SHA256, SHA1, MD5 — 6종 - PFS group: PFS24(DHGroup24), ECP384(DHGroup20), ECP256(DHGroup19), PFS2048(DHGroup14), PFS2(DHGroup2), PFS1(DHGroup1), None — 7종 - Quick Mode SA lifetime: (Optional; default values if not specified) Seconds (integer; minimum 300, default 27,000), Kilobytes (integer; minimum 1,024, default 102400000) - Traffic selector: `UsePolicyBasedTrafficSelectors` (`$True` or `$False`, b[illegible]; default `$False` if not specified) - DPD timeout: Seconds (integer; minimum 9, maximum 3,600, default 45)
ETC	* Rekey margin time (seconds): (Default 540) 60 ~ half of phase 2 lifetime * Rekey fuzz (percentage): (Default 100) 0 and 100 * Replay window size (packets): (Default 1024) 64 and 2048 * DPD timeout (seconds): (Default 30) 30 or higher * DPD timout action: (Default clear) clear | none | restart * Startup action: (Default add) add | start	
References	* https://docs.aws.amazon.com/vpn/latest/s2svpn/VPNTunnels.html	* https://learn.microsoft.com/en-us/azure/vpn-gateway/ipsec-ike-policy-howto

Source: https://docs.google.com/spreadsheets/d/1MH2ctI7NbauPIIK1HP_awDsCYnCihClhyj3tjb8zsko/edit?gid=672109706#gid=672109706

참고

IPsec negotiation phases and algorithms in CSP VPNs

Representative Name Candidate	GCP	Alibaba	Tencent	IBM
Note	Be set in VPN Tunnel (Supported tunnel options) * IKEv2 ciphers that use the authenticated encryption with associated data (AEAD) * IKE version: (Default: 2) 1, 2	Be set in Connection (tf), IPsec Connection (console) (Tunnel options) * Tunnel settings - Customer Gateway - Pre-Shared Key	Be set in VPN Connection (tf) / Tunnel (console) (Tunnel options) * IKE Version: (Default: IKEV1) IKEV1, IKEV2	Be set in VPN Gateway Connection (tf) / VPN connection for VPC (console), IKE policy, IPsec policy * Peer gateway address: IP address or FQDN * Establish mode: Bidirectional or peer only * Preshared key: (Policies of IKE negotiation) * IKE version: (Default: IKEv2) - Note: IBM Cloud auto-negotiation uses IKEv2. Use a customized IKE policy if your on-premises device does not support IKEv2.
IPsec negotiation phase 1 = IKE phase 1 = ISAKMP SA	* IKEv2 ciphers that don't use AEAD - phase 1 - Encryption: AES-CBC-128, AES-CBC-192, AES-CBC-256, 3DES-CBC - Integrity: AES-XCBC-96, AES-CMAC-96, HMAC-SHA1-96, HMAC-MD5-96, HMAC-SHA2-256-128, HMAC-SHA2-384-192, HMAC-SHA2-512-256 - Pseudo-Random Function (PRF): (Many devices don't require an explicit PRF setting.) PRF-AES-128-XCBC, PRF-AES-128-CMAC, PRF-HMAC-SHA1, PRF-HMAC-MD5, PRF-HMAC-SHA2-256, PRF-HMAC-SHA2-384, PRF-HMAC-SHA2-512 - Diffie-Hellman (HD): modp_2048 (Group 14), modp_2048_224 (modp_2048s224), modp_2048_256 (modp_2048s256), modp_1536 (Group 5), modp_3072 (Group 15), modp_4096 (Group 16), modp_8192 (Group 18) *, modp_1024 (Group 2), modp_1024_160 (modp_1024s160), ecp_256 (Group 19), ecp_384 (Group 20), ecp_521 (Group 21), curve_25519 (Group 31) - Phase 1 lifetime: 36,000 seconds (10 hours) - Pseudo-Random Function (PRF): PRF-AES128-XCBC, PRF-AES128-CMAC, PRF-HMAC-SHA1, PRF-HMAC-MD5, PRF-HMAC-SHA2-256, PRF-HMAC-SHA2-384, PRF-HMAC-SHA2-512 * IKEv2 ciphers that use AEAD - phase 1 - Encryption & Integrity: AES-GCM-16-128, AES-GCM-16-192, AES-GCM-16-256 - - Pseudo-Random Function (PRF): (Many devices don't require an explicit PRF setting.) PRF-AES128-XCBC, PRF-AES128-CMAC, PRF-HMAC-SHA1, PRF-HMAC-MD5, PRF-HMAC-SHA2-256, PRF-HMAC-SHA2-384, PRF-HMAC-SHA2-512 - Diffie-Hellman (HD): modp_2048 (Group 14), modp_2048_224 (modp_2048s224), modp_2048_256 (modp_2048s256), modp_1536 (Group 5), modp_3072 (Group 15), modp_4096 (Group 16), modp_8192 (Group 18) *, modp_1024 (Group 2), modp_1024_160 (modp_1024s160), ecp_256 (Group 19), ecp_384 (Group 20), ecp_521 (Group 21), curve_25519 (Group 31) - Phase 1 lifetime: 36,000 seconds (10 hours)	* Encryption configurations: IKE configurations - IKE Version: (Default: ikev2) ikev1, ikev2 - Negotiation Mode: (Default: main) main, aggressive - Encryption Algorithm: (Default: aes) aes (=aes128), aes192, aes256, des, and 3des - Authentication Algorithm: (Default: sha1) sha1, md5, sha256, sha384, and sha512 - DH Group (Perfect Forward Secrecy): (Default: group2) group1, group2, group5, group14 - SA Life Cycle (seconds): (Default: 86400) 0 ~ 86400 - LocalId: (Default: the IP address of the tunnel) IP address or a fully qualified domain name (FQDN) - RemoteId: (Default: the IP address of the customer gateway) IP address or an FQDN - Note: Recommened to set aggresive mode if FQDN is used.	* IKE configuration - Encryption algorithm: (Default: 3DES-CBC) 3DES-CBC, AES-CBC-128, AES-CBC-192, AES-CBC-256, DES-CBC, SM4 + (tf) AES128GCM128, AES192GCM128, AES256GCM128,AES128GCM128, AES192GCM128, AES256GCM128 - Authentication algorithm: (Default: MD5) MD5, SHA, SHA-256 + (console) SHA-384, SHA-512, SM3 - DH group: (Default: GROUP2) GROUP1, GROUP2, GROUP5, GROUP14, and GROUP24 - IKE SA lifetime: (Default: 86400) 60 ~ 604800 - Negotiation model: (Default: MAIN) MAIN, AGGRESSIVE - Local identifier: (Default: ADDRESS) ADDRESS, FQDN - Local address: xxx - Remote identifier: (Default: ADDRESS) ADDRESS, FQDN - Remote address: xxx	* IKE policy (Phase 1) - IKE version: 1 or 2 - Encryption Algorithm: aes128, aes192, aes256 - Authentication Algorithm: sha256, sha512, sha384 - Diffie-Hellman (DH) Group: 14, 19, 15, 16 ,17 ,18 ,20 ,21 ,22 ,23 ,24 ,31 - Key lifetime: (Default: 28800) 1800 ~ 84600
IPsec negotiation phase 2 = IKE phase 2 = IPsec SA	* IKEv2 ciphers that don't use AEAD - phase 2 - Encryption: AES-CBC-128, AES-CBC-256, AES-CBC-192, - Integrity: HMAC-SHA2-256-128, HMAC-SHA2-512-256, HMAC-SHA1-96 - Diffie-Hellman (DH): Refer to Phase 1. - Phase 2 lifetime: 10,800 seconds (3 hours) - PFS Algorithm (required): modp_2048 (Group 14), modp_2048_224 (modp_2048s224), modp_2048_256 (modp_2048s256), modp_1536 (Group 5), modp_3072 (Group 15), modp_4096 (Group 16), modp_8192 (Group 18) *, modp_1024 (Group 2), modp_1024_160 (modp_1024s160), ecp_256 (Group 19), ecp_384 (Group 20), ecp_521 (Group 21), curve_25519 (Group 31) * IKEv2 ciphers that use AEAD - phase 2 - Encryption & Integrity: AES-GCM-16-128, AES-GCM-16-256, AES-GCM-16-192 - - PFS Algorithm (required): modp_2048 (Group 14), modp_2048_224 (modp_2048s224), modp_2048_256 (modp_2048s256), modp_1536 (Group 5), modp_3072 (Group 15), modp_4096 (Group 16), modp_8192 (Group 18) *, modp_1024 (Group 2), modp_1024_160 (modp_1024s160), ecp_256 (Group 19), ecp_384 (Group 20), ecp_521 (Group 21), curve_25519 (Group 31) - Diffie-Hellman (DH): Refer to Phase 1. - Phase 2 lifetime: 10,800 seconds (3 hours)	* Encryption configurations: IPsec configurations - Encryption Algorithm: (Default: aes) aes (=aes128), aes192, aes256, des, and 3des - Authentication Algorithm: (Default: sha1) sha1, md5, sha256, sha384, and sha512 - DH Group (Perfect Forward Secrecy): (Default: group2) disabled, group1, group2, group5, group14 - SA Life Cycle (seconds): (Default: 86400) 0 ~ 86400 - Dead peer detection (DPD): (Default: enabled) - NAT Traversal: (Default: enabled)	* IPsec information - Encryption algorithm: (Default: 3DES-CBC) 3DES-CBC, AES-CBC-128, AES-CBC-192, AES-CBC-256, DES-CBC, SM4, + (tf) NULL, AES128GCM128, AES192GCM128, AES256GCM128 - Authentication algorithm: (Default: MD5) MD5, SHA1, SHA-256 + (console) SHA-384, SHA-512, SM3 - PFS: (Default: NULL) DH-GROUP1, DH-GROUP2, DH-GROUP5, DH-GROUP14, DH-GROUP24, NULL - IPsec SA Lifetime (seconds): (Default: 3600) 180 ~ 604800 - IPsec SA lifetime (traffic): (Default: 1843200) 2560 ~ - Packet encapsulation mode: "Tunnel" (고정값) - Security protocol: "ESP" (고정값)	* IPsec policy (Phase 2) - Encryption Algorithm: aes128, aes192, aes256, aes128gcm16, aes192gcm16, aes256gcm16 - Authentication Algorithm: sha256, sha512, sha384, disabled - Perfect Forward Secrecy (PFS): disabled, group_2, group_5, and group_14 - Key lifetime: (Default: 3600) 300 ~ 86400
ETC	* IKEv2 ciphers that use AEAD - phase 1 & 2 - see link * IKEv1 ciphers - phase 1 & 2 - see link	* BGP configuration - Tunnel CIDR Block: The CIDR block must fall into 169.254.0.0/16. The mask of the CIDR block must be 30 bits in length. The CIDR block cannot be 169.254.0.0/30, 169.254.1.0/30, 169.254.2.0/30, 169.254.3.0/30, 169.254.4.0/30, 169.254.5.0/30, or 169.254.169.252/30. - Local BGP IP address: The BGP IP address of the tunnel. This IP address must fall within the CIDR block of the tunnel.		
References	* https://cloud.google.com/network-connectivity/docs/vpn/concepts/supported-ike-ciphers * https://en.wikipedia.org/wiki/Authenticated_encryption	* https://www.alibabacloud.com/help/en/vpn/sub-product-ipsec-vpn/user-guide/create-and-manage-an-ipsec-vpn-connection-in-dual-tunnel-mode?spm=a2c63.p38356.help-menu-2786991.d_2_1_1.243e699ehUmMmi	* https://www.tencentcloud.com/document/product/1037/39635	* https://cloud.ibm.com/docs/vpc?topic=vpc-using-vpn

Source: https://docs.google.com/spreadsheets/d/1MH2ctl7NbauPIIK1HP_awDsCYnCihClhyj3tjb8zsko/edit?gid=672109706#gid=672109706

Static routing and Dynamic routing

특징	Static Routing	Dynamic Routing
설정 방식	수동	자동 (라우팅 프로토콜 사용)
유지보수	변경 시 수동 수정 필요	자동 업데이트
확장성	낮음	높음
사용 환경	단순한 네트워크	복잡한 네트워크, 멀티클라우드, 클라우드 환경
예제	직접 Route 추가	BGP, OSPF 등 사용

BGP

- Dynamic Routing
  - 라우팅 프로토콜(BGP, Boarder Gateway Protocol)을 사용하여 자동으로 경로를 학습 및 업데이트
  - 네트워크 토폴로지가 변경되면 자동으로 적절한 경로를 선택
  - 설정이 복잡할 수 있으나, 대규모 네트워크에서 자동화 및 유연성을 제공
  - 클라우드 환경, 멀티 클라우드 VPN, 고가용성 네트워크에서 주로 사용
- Static Routing
  - 관리자가 수동으로 라우팅 테이블에 경로를 설정
  - 네트워크 변경이 발생하면 직접 수정 필요
  - 설정이 간단하고 예측이 가능하지만, 수동 변경이 필요하므로 확장성이 낮음
  - 소규모 네트워크 또는 변경이 적은 환경에 적합

- The type of routing that you select can depend on the make and model of your customer gateway device.
  - If your customer gateway device supports Border Gateway Protocol (BGP), specify dynamic routing when you configure your Site-to-site VPN connection.
  - If your customer gateway device does not support BGP, specify static routing.

Source:
https://docs.aws.amazon.com/vpn/latest/s2svpn/vpn-static-dynamic.html

참고

BGP support in CSP VPNs

Note: It indicates whether BGP is supported or not at the VPN Gateway level configuration.

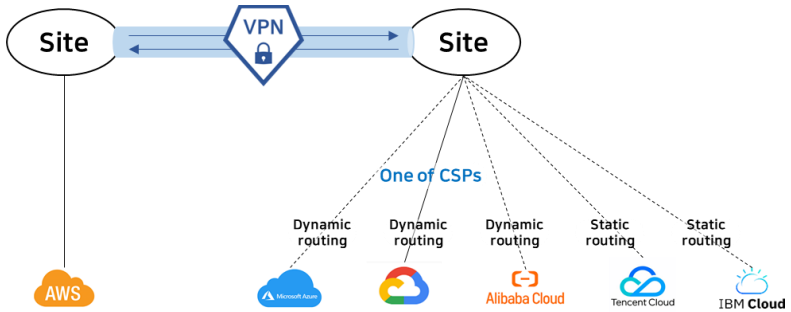
O O O O X X

Representative Name Candidate	AWS	Azure	GCP	Alibaba	Tencent	IBM
BGP support	Supported - Customer Gateway	Supported - Virtual Network Gateway - Local Network Gateway - Virtual Network Gateway Connection	Supported - Cloud Router - Router Interface - Router Peer	Supported - Customer Gateway - Connection	Not supported	Not supported
References	* https://docs.aws.amazon.com/vpn/latest/s2svpn/VPNTunnels.html	* https://learn.microsoft.com/en-us/azure/vpn-gateway/ipsec-ike-policy-howto	* https://cloud.google.com/network-connectivity/docs/vpn/concepts/supported-ike-ciphers * https://en.wikipedia.org/wiki/Authenticated_encryption	* https://www.alibabacloud.com/help/en/vpn/sub-product-ipsec-vpn/user-guide/create-and-manage-an-ipsec-vpn-connection-in-dual-tunnel-mode?spm=a2c63.p38356.help-menu-2786991.d_2_1_1.243e699ehUmMmi	* https://www.tencentcloud.com/document/product/1037/39635	* https://cloud.ibm.com/docs/vpc?topic=vpc-using-vpn

VPN을 구축하는데 필요한 서로 다른 종류의 자원, 활용 방식, 복잡한 종속성을 보여드렸습니다.

관련된 모든 자원들을 추상화 후 공통 API로 제공하는 것. 쉽지 않고 비효율적이겠죠?

(Remind) Objectives of site-to-site VPN in Cloud-Barista



Choose “combo” or “all the way”

— made with care, balanced just right.

Configure your site-to-site VPN like placing an order

— we’ll handle what’s under the hood to ensure maturity and stability

➡ mc-terrarium makes it possible!

Introduction to mc-terrarium

Multi-Cloud Terrarium (mc-terrarium) is an open-source project designed to provide an environment —an infrastructure terrarium—that **enhances** multi-cloud infrastructure management.

Repository: https://github.com/cloud-barista/mc-terrarium

🚀 Powered by OpenTofu

🤝 A Collaboration Between Cloud-Barista & OpenTofu

네트워크 자원의 공통 확장과 개별 확장

General Purpose APIs

CLOUD BARISTA

- Supported Computing Infrastructure Resources
  - **Basic Resources**: Public Image, VM Spec, VPC/Subnet, Security Group, VM KeyPair
  - **VM Infrastructures**: VM, NLB(Network Load Balancer), Disk, MyImage
  - **Container Infrastructures**: PMKS(Provider-Managed K8S)
- Supported CloudOS:

Provider, CloudOS	CloudOS Constant	Cloud Driver Lib.	Etc
Amazon Web Services	AWS	aws-driver-v1.0.so	
Microsoft Azure	AZURE	azure-driver-v1.0.so	
Google Cloud Platform	GCP	gcp-driver-v1.0.so	
Alibaba Cloud	ALIBABA	alibaba-driver-v1.0.so	
Tencent Cloud	TENCENT	tencent-driver-v1.0.so	
IBM VPC Cloud	IBM	ibmvpc-driver-v1.0.so	
OpenStack Platform	OPENSTACK	openstack-driver-v1.0.so	
NCP Classic Cloud	NCP	ncp-driver-v1.0.so	
NCP VPC Cloud	NCPVPC	ncpvpc-driver-v1.0.so	
NHN Cloud	NHNCLOUD	nhncloud-driver-v1.0.so	
KT Classic Cloud	KTCLOUD	ktcloud-driver-v1.0.so	
KT VPC Cloud	KTCLOUDVPC	ktcloudvpc-driver-v1.0.so	

APIs: 멀티 클라우드 자원/서비스 통합 운용 관리

CB-Tumblebug

멀티 클라우드 자원/서비스 통합 운용 관리

: 추상화된 Network 자원(VPC, Subnet 등) | : CSP Network 자원 (CSP 간 connectivity를 위한)

공통 APIs: 추상화된 클라우드 자원 제어

CB-Spider

(CSP API 추상화)

APIs: 클라우드 특유의 자원 제어

mc-terrarium

Infracode (HCL) 기반
네트워크 자원 제어

OpenTofu

CSP CSP CSP CSP CSP CSP ...

1. 각 CSP에 네트워크를 구성

2. CSP간 연결성 지원

Specific Purpose IaCs

(API를 곁들인 ^^;;)

OpenTofu

개별 확장 = 자원 보강

목표: Site-to-site VPN

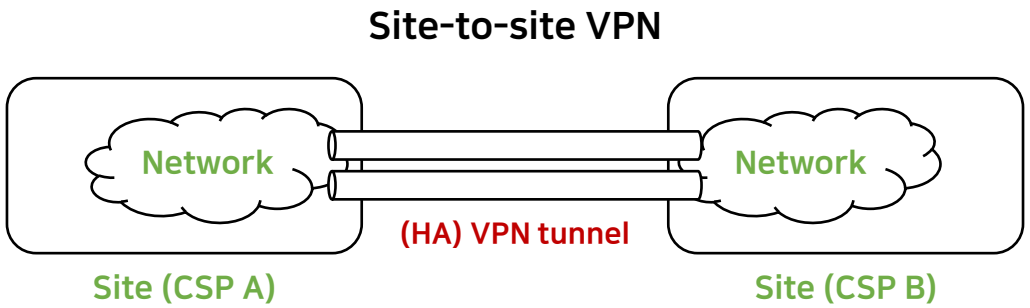
(As-Is) Supported AWS-to-site VPN
- Hub site: AWS
- Spoke site: Azure, GCP, Alibaba, Tencent, IBM

Source:
- CB-Spider, "Supported Computing Infrastructure Resources", (accessed on 2024-03-14, https://github.com/cloud-barista/cb-spider/wiki/Supported-Compute-Infrastructure-Resources#supported-computing-infrastructure-resources)
- CB-Spider, "Supported CloudOS", (accessed on 2024-03-14, https://github.com/cloud-barista/cb-spider/wiki/Supported-CloudOS#supported-cloudos)

(개념설명) OpenTofu 기반의 Site-to-site VPN 보강(또는 생성)

mc-terrarium: OpenTofu 기반의 클라우드 자원/서비스 보강 시스템(환경)

+ 멀티 클라우드 인프라 구축
+ Site-to-site VPN 구축



General Purpose APIs

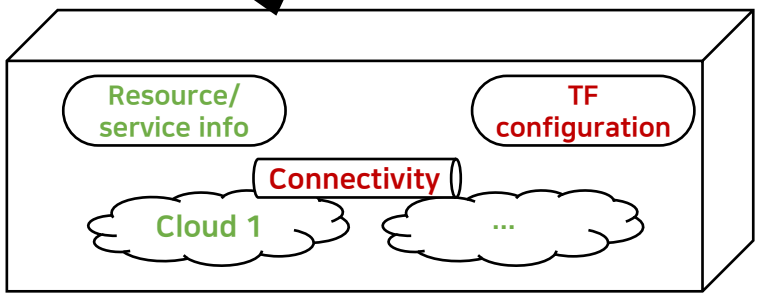
Cloud Barista

생성된 MCI 정보
(SP 및 TB를 통해 생성)

보강할 인프라 자원 코드
(Terrarium에서 VPN 보강)

Cloud Barista

OpenTofu client



* 선언형 (Declarative)

Specific Purpose IaCs
(API를 곁들인 ^^;;)

OpenTofu

mc-terrarium: AWS-to-site VPN 보강 기능/APIs

AWS-to-site VPN APIs (지속 확장 예정)

[Terrarium] An environment to enrich the multi-cloud infrastructure

- GET /tr Read all terrarium
- POST /tr Issue/create a terrarium
- GET /tr/{trId} Read a terrarium
- DELETE /tr/{trId} Erase the entire terrarium including directories and configuration files

[AWS to site VPN] Resource Operations

- GET /tr/{trId}/vpn/aws-to-site Get AWS to site VPN
- POST /tr/{trId}/vpn/aws-to-site Create AWS to site VPN
- DELETE /tr/{trId}/vpn/aws-to-site Delete AWS to site VPN

[AWS to site VPN] OpenTofu Actions (for fine-grained contorl)

- POST /tr/{trId}/vpn/aws-to-site/actions/apply Apply AWS to site VPN
- DELETE /tr/{trId}/vpn/aws-to-site/actions/destroy Destroy AWS to site VPN
- DELETE /tr/{trId}/vpn/aws-to-site/actions/emptyout EmptyOut AWS to site VPN
- POST /tr/{trId}/vpn/aws-to-site/actions/init Init AWS to site VPN
- GET /tr/{trId}/vpn/aws-to-site/actions/output Output AWS to site VPN
- POST /tr/{trId}/vpn/aws-to-site/actions/plan Plan AWS to site VPN

API 입력 값 (Request Body)

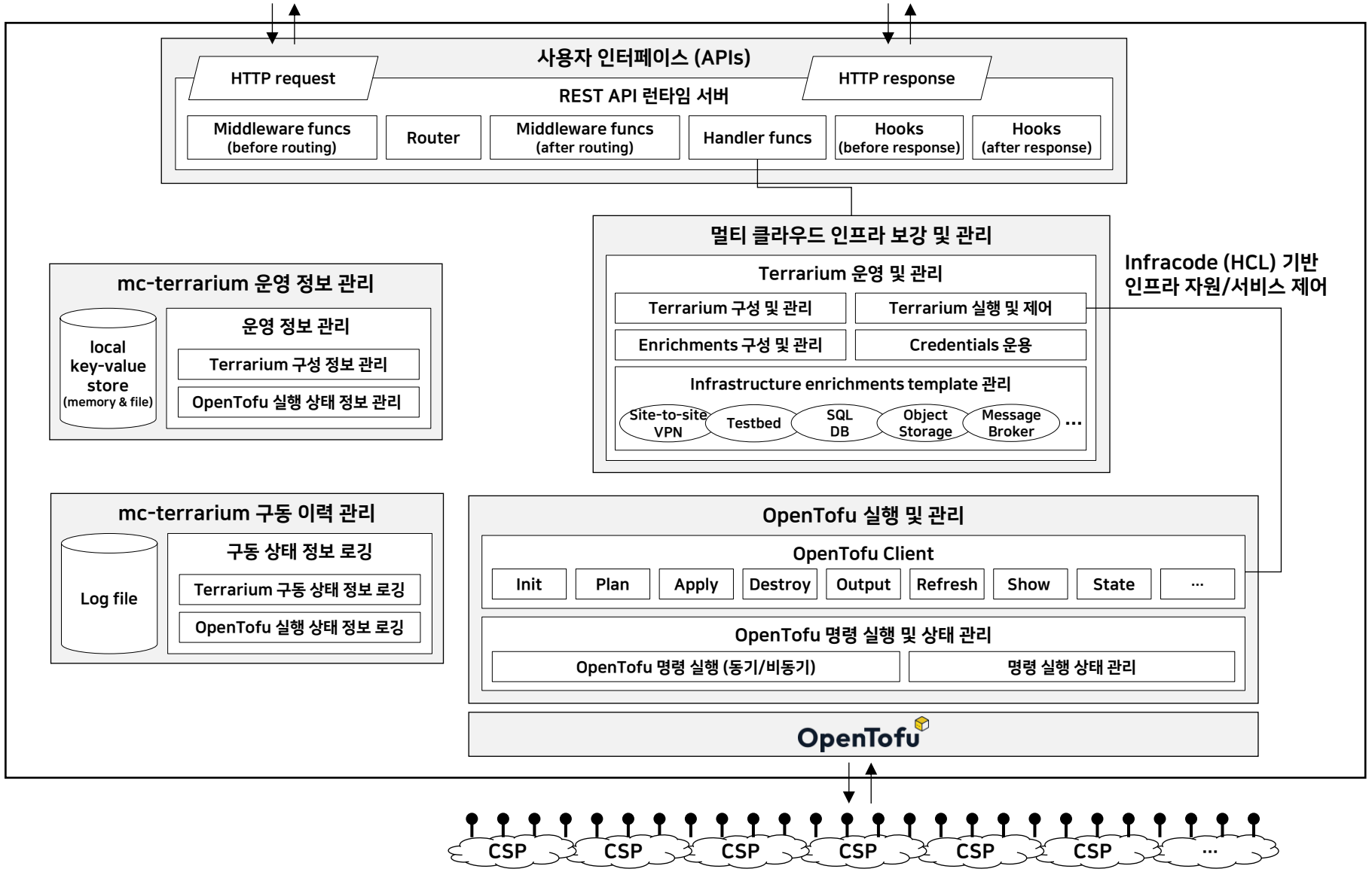
```
{
 "vpn_config": {
  "aws": {
   "bgp_asn": "64512",
   "region": "ap-northeast-2",
   "subnet_id": "string",
   "vpc_id": "string"
  },
  "target_csp": {
   "alibaba": {
    "bgp_asn": "65532",
    "region": "ap-northeast-2",
    "vpc_id": "string",
    "vswitch_id_1": "string",
    "vswitch_id_2": "string"
   },
   "azure": {
    "bgp_asn": "65531",
    "gateway_subnet_cidr": "string",
    "region": "string",
    "resource_group_name": "string",
    "virtual_network_name": "string",
    "vpn_sku": "VpnGw1AZ"
   },
   "gcp": {
    "bgp_asn": "65530",
    "region": "asia-northeast3",
    "vpc_network_name": "string"
   },
   "ibm": {
    "region": "au-syd",
    "subnet_id": "string",
    "vpc_cidr": "string",
    "vpc_id": "string"
   },
   "tencent": {
    "region": "ap-seoul",
    "subnet_id": "string",
    "vpc_id": "string"
   },
   "type": "string"
  },
  "terrarium_id": "string"
 }
}
```

(참고) API groups

- [System] Utility
- [Terrarium] An environment to enrich the multi-cloud infrastructure ✓
- [Message Broker] Operations
- [Object Storage] Operations
- [SQL Database] Operations
- [Testbed] Resource Operations
- [Testbed] OpenTofu Actions (for fine-grained contorl)
- [AWS to site VPN] Resource Operations ✓
- [AWS to site VPN] OpenTofu Actions (for fine-grained contorl) ✓
- [VPN] GCP to AWS VPN tunnel configuration ✓
- [VPN] GCP to Azure VPN tunnel configuration (under development) ✓

참고

mc-terrarium: 시스템 구조



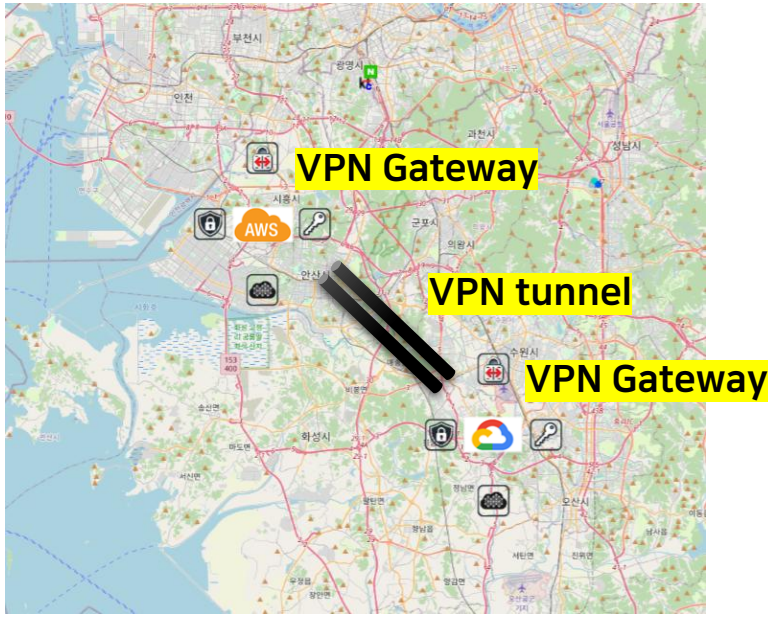
멀티 클라우드 네트워크 관리 기능/API 제공: Site-to-site VPN

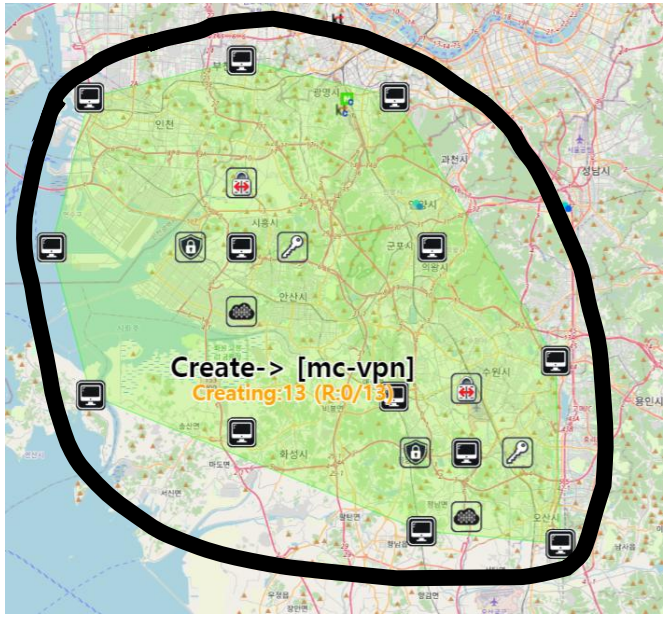
- CB-Tumblebug: Site-to-site VPN 기능/APIs 제공
  - mc-terrarium API 연동
  - Site-to-site VPN objects (metadata) 관리 기능 추가
    - VPN GW 자원의 CRUD 상태 및 정보를 나타내는 Object의 관리 기능 개선
  - 지원 현황: AWS-to-site VPN 지원
    - AWS hub와 타 CSP spoke간 1:1 VPN 연결
    - 지원 CSP spoke: Azure, GCP, Alibaba, Tencent, IBM
- (참고) API 사용 방법:
  - https://github.com/cloud-barista/mc-terrarium/discussions/81

Tumblebug APIs

[Infra Resource] Site-to-site VPN Management (under development)

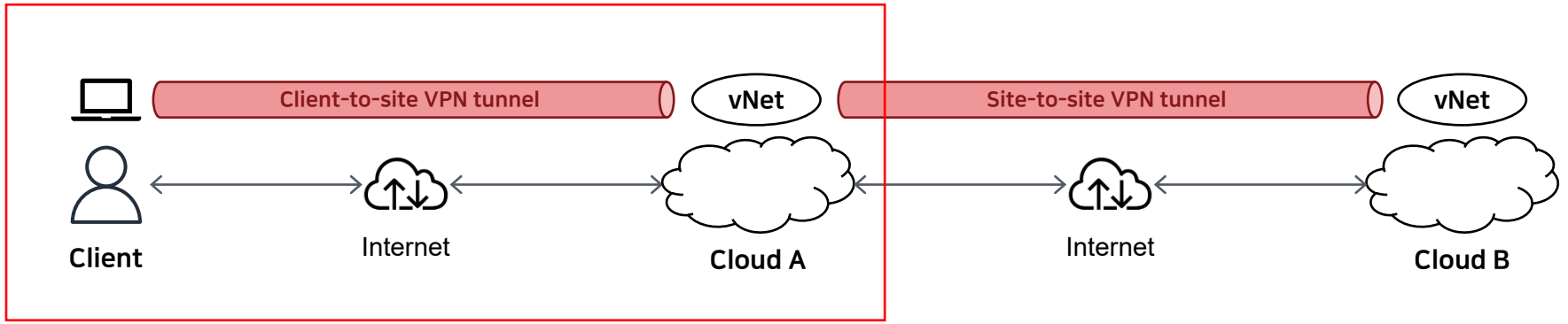
GET	/ns/{nsId}/mci/{mciId}/site	Get sites in MCI
GET	/ns/{nsId}/mci/{mciId}/vpn	Get all site-to-site VPNs
POST	/ns/{nsId}/mci/{mciId}/vpn	Create a site-to-site VPN
GET	/ns/{nsId}/mci/{mciId}/vpn/{vpnId}	Get resource info of a site-to-site VPN
DELETE	/ns/{nsId}/mci/{mciId}/vpn/{vpnId}	Delete a site-to-site VPN
GET	/ns/{nsId}/mci/{mciId}/vpn/{vpnId}/request/{requestId}	Check the status of a specific request by its ID





Multi-Cloud와 안전한 연결성 지원 ※ 연구개발용

Client-to-site VPN 개요 및 현황 (i.e., SSL VPN)



(현황) WireGuard 기반 Client-to-site VPN 구성 가이드 및 Bootstrapper 스크립트 제공

1. 사용자가 중요 정보를 수정 후 WireGuard Easy VPN Setup을 배포할 수 있도록 가이드 제공
2. 간단한 테스트 용도로 Bootstrapper 스크립트 제공

중요! 라이선스를 확인 후 사용하시기 바랍니다.

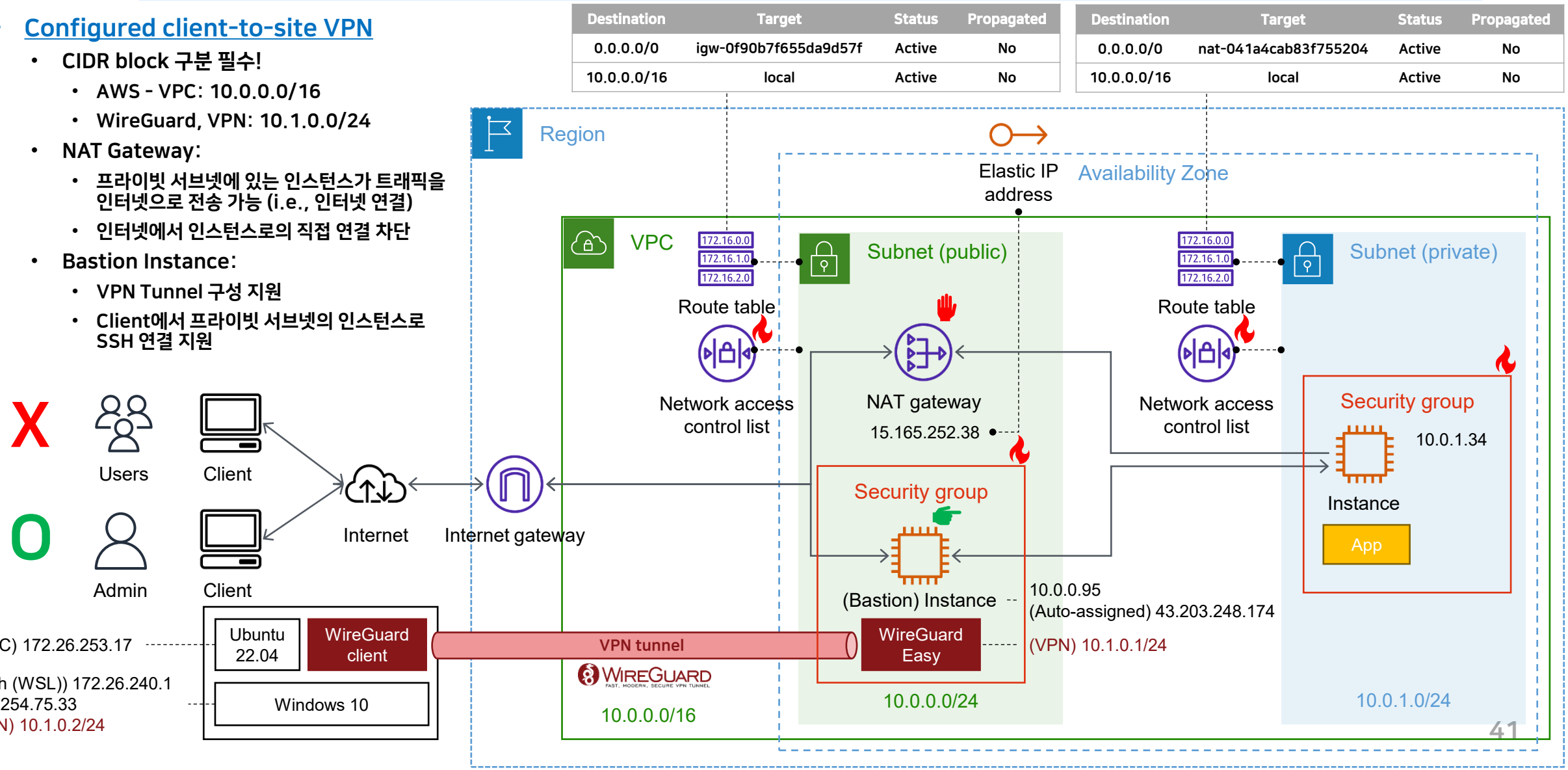
* WireGuard Easy license: AGPL-3.0 license / WireGuard for the linux kernel license: GPLv2 / WireGuard Tools: GPLv2 / Windows client of WireGuard: MIT

(효과)

1. 상용 클라우드를 안전하게 운영, 관리, 활용할 수 있을 것으로 기대 (예, 안전한 시연환경/테스트베드 구성)
2. 마이그레이션 수행 시, 소스 컴퓨팅 환경의 서버와 클라우드의 VPC/vNet 간 VPN 구성 지원 기대(필요시)

안전한 Cloud-Barista Testbed 구축을 위한 Architecture

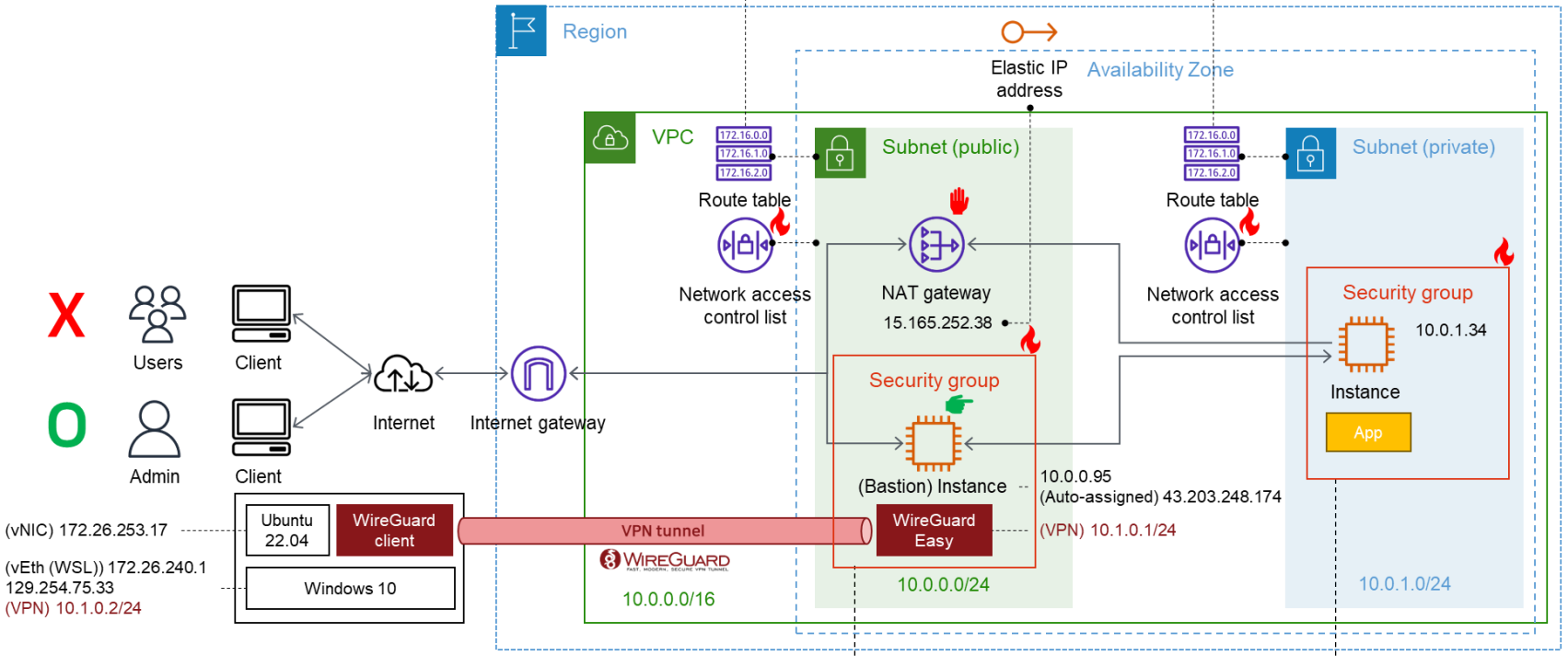
- Configured client-to-site VPN
  - CIDR block 구분 필수!
    - AWS - VPC: 10.0.0.0/16
    - WireGuard, VPN: 10.1.0.0/24
  - NAT Gateway:
    - 프라이빗 서브넷에 있는 인스턴스가 트래픽을 인터넷으로 전송 가능 (i.e., 인터넷 연결)
    - 인터넷에서 인스턴스로의 직접 연결 차단
  - Bastion Instance:
    - VPN Tunnel 구성 지원
    - Client에서 프라이빗 서브넷의 인스턴스로 SSH 연결 지원



안전한 Cloud-Barista Testbed 구축을 위한 Security Groups

Destination	Target	Status	Propagated
0.0.0.0/0	igw-0f90b7f655da9d57f	Active	No
10.0.0.0/16	local	Active	No

Destination	Target	Status	Propagated
0.0.0.0/0	nat-041a4cab83f755204	Active	No
10.0.0.0/16	local	Active	No



in/out-bound	Name	Security group rule ID	IP version	Type	Protocol	Port range	Source	Desc
inbound	-	sgr-0f1377a0473a06153	IPv4	All ICMP - IPv4	ICMP	All	10.0.0.0/16	Allow ···
inbound	-	sgr-068dfd2a7c2e6b6ef	IPv4	Custom UDP	UDP	51820	ooo.ooo.0.0/16	Allow ···
inbound	-	sgr-002e003597fe28b15	IPv4	Custom TCP	TCP	51821	ooo.ooo.0.0/16	Allow ···
inbound	-	sgr-0489578751d2058b9	IPv4	SSH	TCP	22	ooo.ooo.0.0/16	Allow ···
outbound	-	sgr-0c822b132c99bf7d5	IPv6	All traffic	All	All	::/0	Allow ···
outbound	-	sgr-0014933eb623ac67a	IPv4	All traffic	All	All	0.0.0.0/0	Allow ···

in/out-bound	Name	Security group rule ID	IP version	Type	Protocol	Port range	Source	Desc
inbound	-	sgr-0e23ec67782ea3657	IPv4	SSH	TCP	22	10.0.0.0/16	Allow ···
inbound	-	sgr-05ed02116c4cef1b6	IPv4	All ICM - IPv4	ICMP	All	10.0.0.0/16	Allow ···
Inbound	-	sgr-0ek59123dkfj4k2kf4	IPv4	Custom TCP	TCP	1323	10.0.0.0/16	Allow ···
Inbound	-	sgr-021ektisad47941j23	IPv4	Custom TCP	TCP	1324	10.0.0.0/16	Allow ···
outbound	-	sgr-0aeb8b219cbbbff53	IPv4	All traffic	All	All	0.0.0.0/0	-

WireGuard Easy 기반의 Client-to-site VPN 구성 방법

1. (Optional) SSH를 통해 Bastion 인스턴스에 접속 또는 원격 명령(Remote Command) 활용 중 선호하는 방식 활용
   - 참고: SSH 연결 또는 원격명령을 위한 SSH Key pair 필요

2. 기 개발 WireGuard Easy bootstrapper script (wg-easy-bootstrapper.sh)를 활용하여 배포 및 구동
   - https://github.com/cloud-barista/mc-terrarium/tree/main/examples/aws/client-to-site-vpn/wireguard-easy
   - Included README, docker-compose.yaml, wg-easy-bootstrapper.sh
     - 참고: Default password: multicloud123!
     - 참고: the bcrypt hash of 'multicloud123!' ➔ $2a$12$iSCQRRM8cJxXnCNbWWM.1.4rHSEXloWPVy6XXei0TXfMWhDsSsTVq

3. WireGuard Easy Web UI 접속 및 Client 설정 정보 생성
   - URL: http://Bastion_Instance_Public_IP:51821
   - Password: foobar123
   - Client 설정 정보 생성 및 다운로드

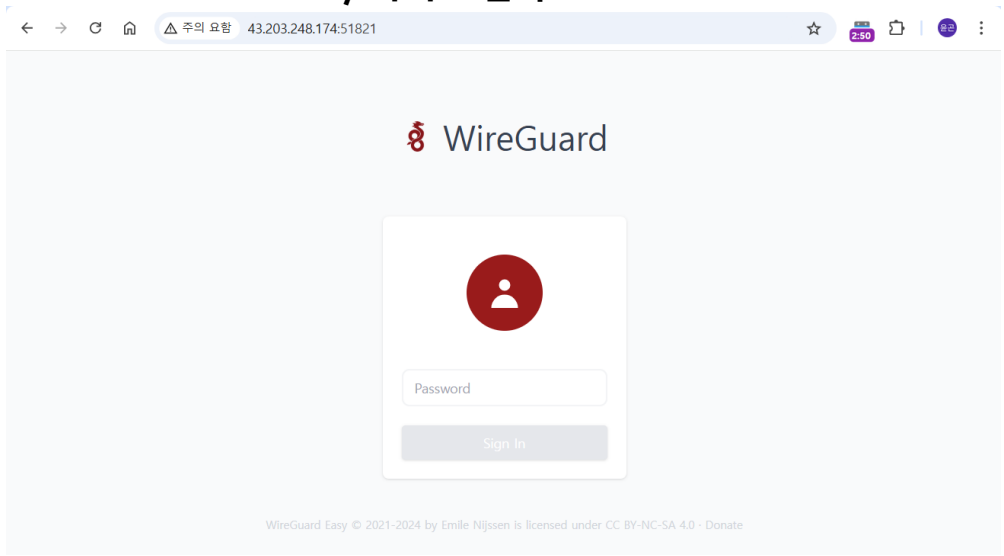
4. WireGuard-Windows client 설치, 구동 및 설정
   - Client 설정 정보 등록
   - 인터페이스 활성화(Activate)

(대체 프로젝트?) wireguard-ui
https://github.com/ngoduykhanh/wireguard-ui
LICENSE: MIT
Last commit: 2024. 01. 13.

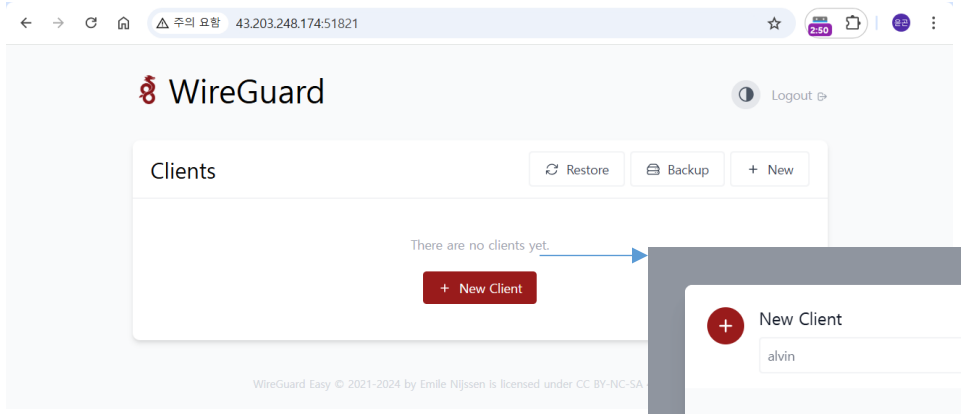
VPN server: WireGuard Easy (WireGuard + WebUI)

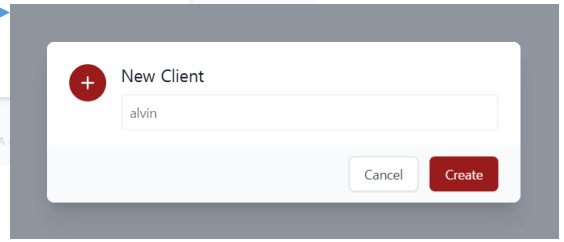
License: AGPL-3.0 license (Changed on Mar. 5th, 2025)

1. WireGuard Easy 사이트 접속

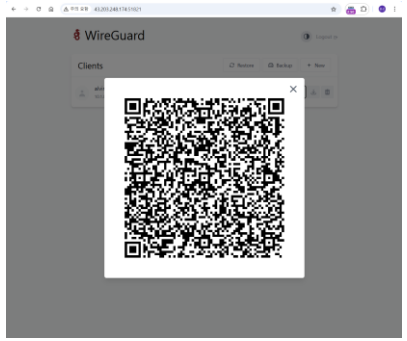


2. Client 생성





3. Client 활성화 및 VPN 연결 정보 획득



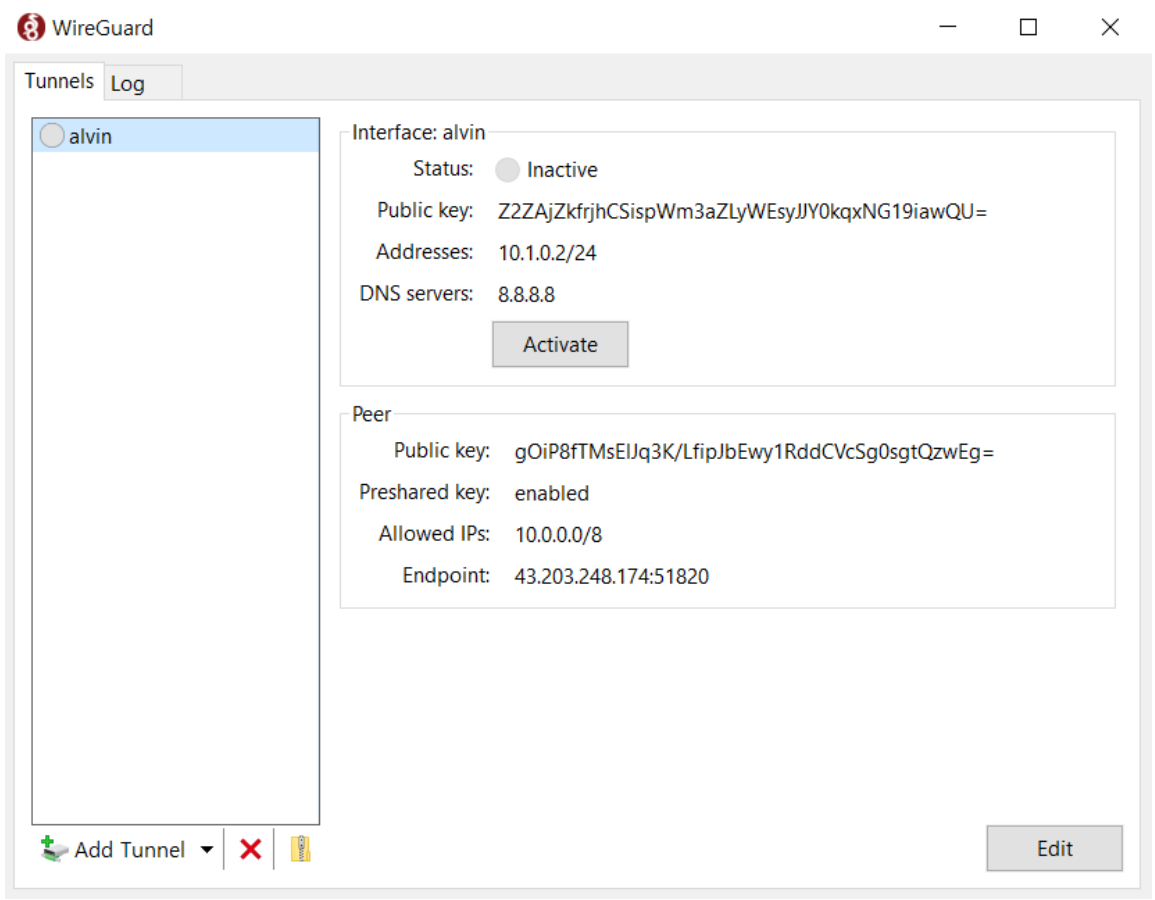

```
[Interface]
PrivateKey = gJvZt8WZt7creZfDNCVH2s1tVyfloryr0HJzHPW21UM=
Address = 10.1.0.2/24
DNS = 8.8.8.8

[Peer]
PublicKey = gOiP8fTMsElJq3K/LfipJbEwylRddCVcSg0sgtQzwEg=
PresharedKey = FawDb5Q2yBmLtD032dOuEmn3JAinoAs2YERvDjte43M=
AllowedIPs = 10.0.0.0/8
PersistentKeepalive = 0
Endpoint = 43.203.248.174:51820
```

Source: https://github.com/wg-easy/wg-easy

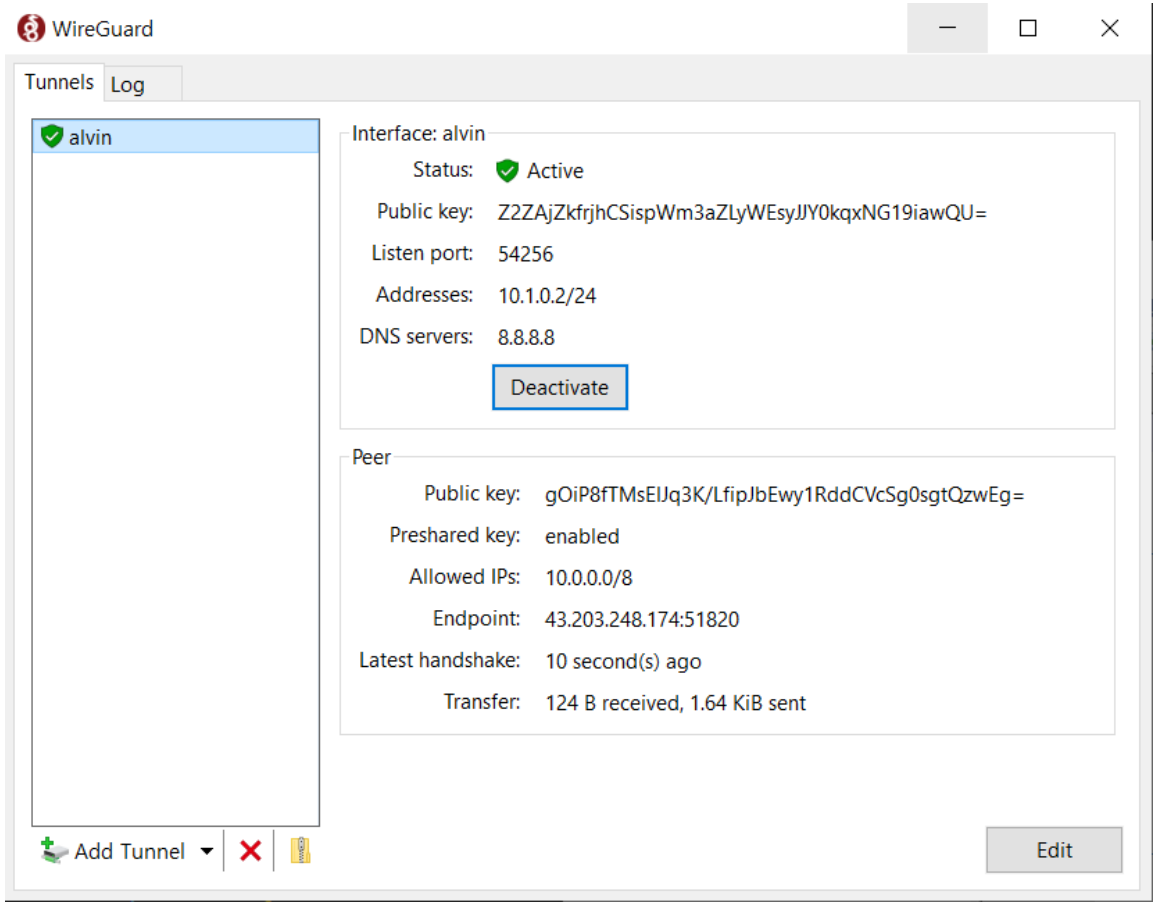
VPN client: WireGuard-Windows

Windows (LICENSE: MIT)

3. VPN 연결 입력

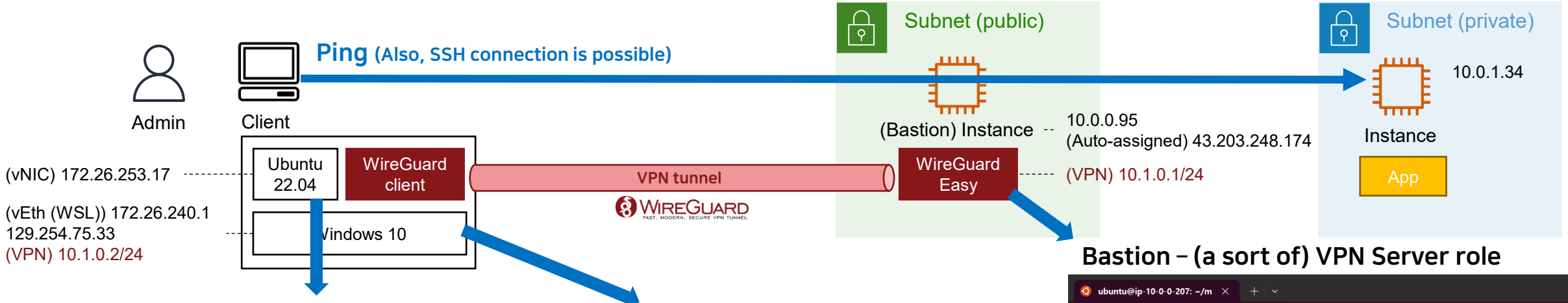


4. VPN Tunnel 활성화



Source:
https://www.wireguard.com/install/
https://github.com/WireGuard/wireguard-windows

Ping 테스트



Guest – (WLS2) Ubuntu 22.04

```
ubuntu@DESKTOP-L3ETGKI:~$ cat /etc/os-release
PRETTY_NAME="Ubuntu 22.04.5 LTS"
NAME="Ubuntu"
VERSION_ID="22.04"
VERSION="22.04.5 LTS (Jammy Jellyfish)"
VERSION_CODENAME=jammy
ID=ubuntu
ID_LIKE=debian
HOME_URL="https://www.ubuntu.com/"
SUPPORT_URL="https://help.ubuntu.com/"
BUG_REPORT_URL="https://bugs.launchpad.net/ubuntu/"
PRIVACY_POLICY_URL="https://www.ubuntu.com/legal/terms-and-policies/privacy-policy"
UBUNTU_CODENAME=jammy
ubuntu@DESKTOP-L3ETGKI:~$ ping 10.0.1.34 -c 3
PING 10.0.1.34 (10.0.1.34) 56(84) bytes of data.
64 bytes from 10.0.1.34: icmp_seq=1 ttl=61 time=6.95 ms
64 bytes from 10.0.1.34: icmp_seq=2 ttl=61 time=7.10 ms
64 bytes from 10.0.1.34: icmp_seq=3 ttl=61 time=9.15 ms

--- 10.0.1.34 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 1998ms
rtt min/avg/max/mdev = 6.953/7.734/9.149/1.002 ms
```

Host - Windows 10

```
선택 Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

새로운 크로스 플랫폼 PowerShell 사용 https://aka.ms/pscore6

PS C:₩Users₩USER> ipconfig

Windows IP 구성


알 수 없는 어댑터 alvin:

   연결별 DNS 접미사. . . . :
   IPv4 주소 . . . . . . . . . . . : 10.1.0.2
   서브넷 마스크 . . . . . . . . . : 255.255.255.0
   기본 게이트웨이 . . . . . . . . :

이더넷 어댑터 이더넷:

   연결별 DNS 접미사. . . . :
   링크-로컬 IPv6 주소 . . . . . . . . : fe80::2866:fda8:d02:715a%7
   IPv4 주소 . . . . . . . . . . . : 129.254.75.33
   서브넷 마스크 . . . . . . . . . : 255.255.255.0
   기본 게이트웨이 . . . . . . . . : fe80::2a87:baff:fe4d:2324%7
                                     129.254.75.1
```

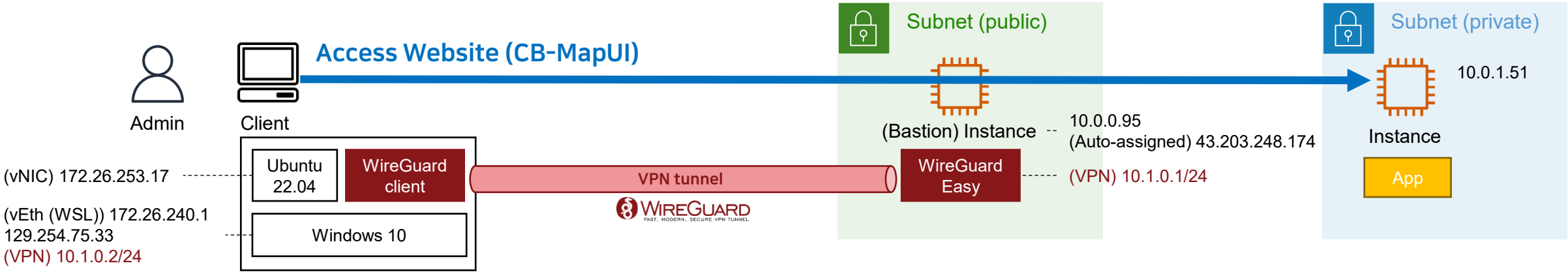
Bastion – (a sort of) VPN Server role

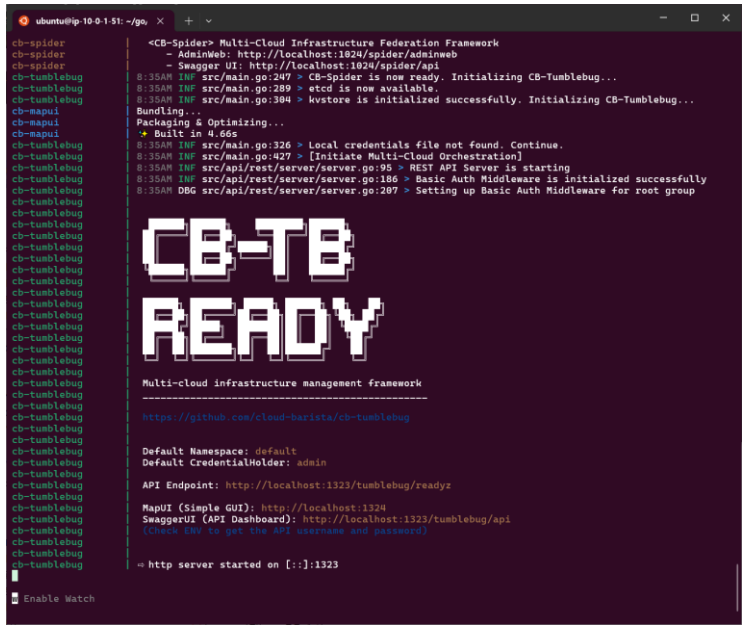
```
ubuntu@ip-10-0-0-207:~/mywg$ sudo docker exec -it fc53436714a2 /bin/bash
fc53436714a2:/app# wg
interface: wg0
  public key: gOiP8fTMsElJq3K/LfipJbEwy1RddCVcSg0sgtQzwEg=
  private key: (hidden)
  listening port: 51820

peer: Z2ZAjZkfrjhCSispWm3aZLyWEsyJJY0kqxNG19iawQU=
  preshared key: (hidden)
  endpoint: 129.254.75.33:61443
  allowed ips: 10.1.0.2/32
  latest handshake: 1 minute, 1 second ago
  transfer: 14.95 KiB received, 2.34 KiB sent
fc53436714a2:/app# ifconfig
eth0      Link encap:Ethernet  HWaddr 02:42:AC:12:00:02
          inet addr:172.18.0.2  Bcast:172.18.255.255  Mask:255.255.0.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:2087 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1785 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:471953 (460.8 KiB)  TX bytes:1292034 (1.2 MiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

wg0       Link encap:UNSPEC  HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
          inet addr:10.1.0.1  P-t-P:10.1.0.1  Mask:255.255.255.0
          UP POINTOPOINT RUNNING NOARP  MTU:1420  Metric:1
          RX packets:104 errors:0 dropped:0 overruns:0 frame:0
          TX packets:41 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:15512 (15.1 KiB)  TX bytes:2392 (2.3 KiB)
```

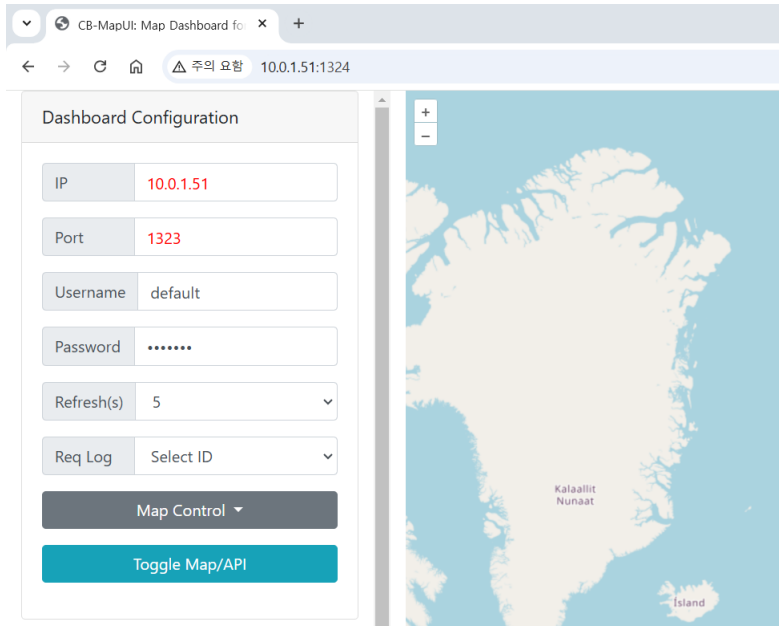
App 구동 및 접속 가능 여부 테스트

(Tumblebug 및 MapUI 구동)

- 10.0.1.51:1324로 CB-MapUI 접속



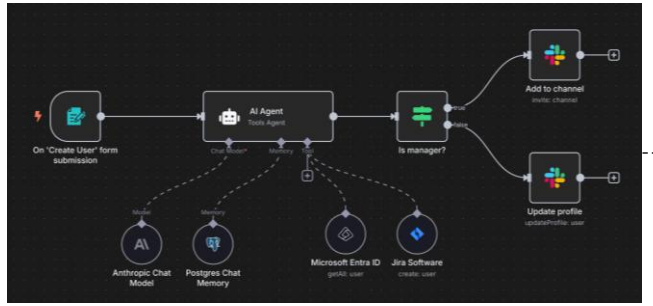




시연: 안정적인 네트워크 기반을 구축하면 무엇을 할 수 있을까요?!

안전한 SW 활용 #AI-tool #GPU #Confidentiality

#Secure #Access #Management **안전한 인프라 운영/관리**

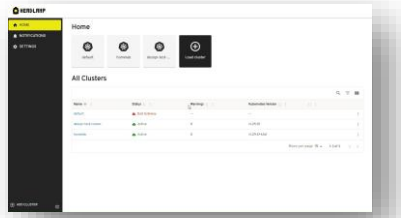


Self-hosting
AI workflow automation tool

n8n

Llama 4

Headlamp
(a user-friendly Kubernetes UI)



Browser (Web Interface) Terminal (SSH/Kubectl)

User (Admin) Client Internet

Client-to-site VPN tunnel

WireGuard Easy

Self-hosting AI workflow automation tool

VM VM VM VM

Subnet1 Subnet2 ...

VNet (VPC)

Hub site: AWS (Seoul)

Site-to-site VPN tunnel (HA VPN tunnel)

Internet

Headlamp

Node Node

Internal Load Balancer

Subnet1 Subnet2 ...

VNet (VPC network)

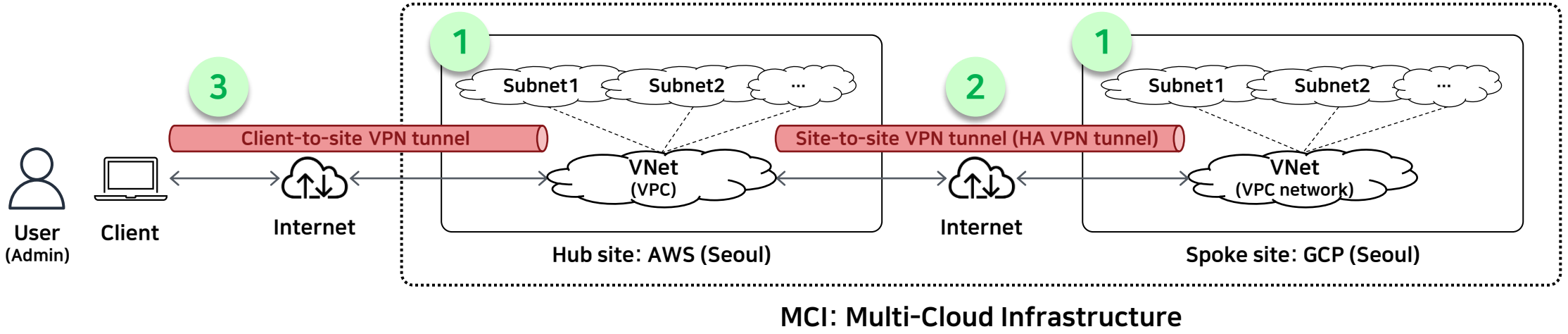
Spoke site: Azure (Seoul)

MCI: Multi-Cloud Infrastructure

안정적인 네트워크 기반을 구축하면
우리는 무엇을 할 수 있을까요?!

Summary

1. 각 CSP에 확실한 네트워크 구성
2. CSP간 안전한 연결성 지원
3. Multi-Cloud와 안전한 연결성 지원



MCI: Multi-Cloud Infrastructure

공개SW 저장소 가볍게 둘러보시기 바랍니다 ☺

- CB-Tumblebug: 멀티 클라우드 네트워크 설계, 검증, 구축
  - https://github.com/cloud-barista/cb-tumblebug

- mc-terrarium: AWS-to-site VPN 구축 + @
  - https://github.com/cloud-barista/mc-terrarium

- Client-to-site VPN 생성 (for research and development purposes)
  - WARNING! Please, check if the license is proper before using WireGuard Easy.
  - https://github.com/cloud-barista/mc-terrarium/tree/main/examples/aws/client-to-site-vpn/wireguard-easy

---

- CM-Beetle: 컴퓨팅 인프라 마이그레이션
  - https://github.com/cloud-barista/cm-beetle

- cb-coffeehouse: 다양한 분야의 정보 공유 (상용 클라우드, 멀티 클라우드, Credentials, IT, 스크립트, GitHub Actions, 등)
  - https://github.com/cloud-barista/cb-coffeehouse

멀티 클라우드에 진심인 사람들의 이야기

전세계 클라우드를 내 손안에, 멀티 클라우드

Cloud-Barista Community 11th Conference

감사합니다.

https://github.com/cloud-barista
https://cloud-barista.github.io

김 윤 곤 / yunkon.kim@etri.re.kr


Cloud
Barista
클라우드바리스타 커뮤니티 제11차 컨퍼런스
부 록
얼그레이 (Earl Grey) 한잔 어떠세요 ?

CSP의 네트워크 서비스/자원 분석

CSP의 네트워크 서비스/자원 분석 및 매칭 테이블

- 링크 참고: https://docs.google.com/spreadsheets/d/1O5tnmDo42aal9VnupuAye-tWpybj01rooxUgD8PLywo/edit?usp=sharing

네트워크 자원/서비스 연관성 분석

Modified on 2023-08-25

AWS	MS Azure	GCP	Alibaba Cloud	Tencent Cloud	IBM Cloud	NHN Cloud	NCP	KT	OpenStack	On-premise
Virutal Private Cloud (VPC)	Virtual network (VNet)	VPC Network	VPC	VPC	VPC 인프라	VPC	VPC	????		Data Center Network
Subnet	Subnet	Subnet	vSwitch	Subnet	Subnet	Subnet	Subnet	Tier (for Server-D1), Private Subnet (for Server G1, G2)		Subnet
						Network Interface				
Internet Gateway (IGW)			IPv4 Gateway		퍼블릭 게이트웨이	Internet Gateway	Internet Gateway			
Elastic IP	Public IP Address	External IP address	Elastic IP	Public IP/EIP	유동IP	Floating IP	Public IP	공인 IP		
Security Group	Network Security Group (NSG)	Firewall	ECS Security Group	Security Group	보안 그룹	Security Group				ACL, Segmentation, and Microsegemtation
Network ACL (Access Control List) or NACL			Network ACL	Network ACL	엑세스 제어 목록	Network ACL	NACL, Access Control Group (ACG):단위 적용			
NAT Gateway	NAT Gateway	Cloud NAT	NAT Gateway	NAT Gateway		NAT Gateway	NAT Gateway	Static NAT		
Route table	Route Table	Route	Route Tables	Route Tables	라우팅 테이블	Routing Table	Route Table			Virtual Router
Network Load Balancer (NLB)	Load Balancer	Load Balancing	Server Load Balancer (SLB)	Cloud Load Balancer (CLB)		Load Balancer	Load Balancer	Load Balancer		L4 Load Balancer
VPC peering	VNet peering	VPC network peering	VPC peering	Peering		Peering Gateway	VPC Peering	Cloud LINK (zone to zone)		
Transit Gateway (TGW)	Virtual WAN		Cloud Enterprise Network		Transit Gateway	(예정, Hub)				
Transit Gateway Route Table										
Attachment										
VPN gateway / VIrtual private gateway	VPN Gateway	Cloud VPN Gateway	VPN Gateway		VPN Gateway		Virtual Private Gateway (IPsec VPN / SSL VPN)			
Customer gateway	VPN Device		Smart Access Gateway (SAG)							
Elastic network interface (ENI)	Network Interface?			Elastic Network Interface (ENI)						
AWS Direct Connect	Azure ExpressRoute	Cloud Interconnect	Express Connect	Direct Connect	Direct Link	Direct Connect	Cloud Connect			
Direct Connect Gateway	ExpressRoute Gateway	Cloud Routers		Direct Connect Gateway		Colocation Gateway (?)				MPLS or Private Circuit
	Bastion									
Route 53	Traffic Manager, Azure Front Door	External Cloud Load Balancing - TCP Proxy, SSL Proxy, HTTP(S)						GSLB		Global Load Balancing and GSLB
Private Link Service	Private Link	Private Service Connect	Private Link	Private Link						
Endpoint	Service Endpoint		Endpoint							
Application Load Balancer	Application Gateway	HTTP(S) Load Balancing						Hybrid Cloud-VPN		L7 Load Balancer
AWS Web Application Firewall (WAF)	Azure WAF	Cloud Armor								WAF

Overview of IPsec phases and algorithms

Internet Key Exchange (IKE) phase 1 and 2

- To establish an IPsec tunnel, we use a protocol called IKE (Internet Key Exchange).
- There are two phases to build an IPsec tunnel:
  - IKE phase 1
  - IKE phase 2

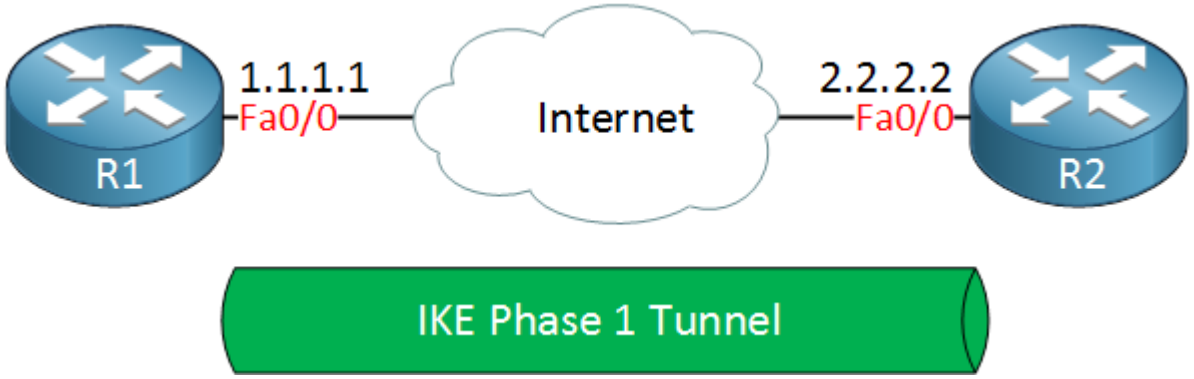
- IKE phase 1 = ISAKMP SA
  - 목표: ISAKMP SA를 협상하여 안전한 터널을 생성하고 상대방을 인증하는 것
  - 안전한 터널: 패킷의 암호화된 교환에 직접적인 영향을 미치는 수단(IKE Phase 2 SA)을 암호화 하여 전달할 수 있는 터널
  - ISAKMP: Internet Security Association and Key Management Protocol
  - SA: Security Association

- IKE phase 2 = IPsec SA
  - 목표: Phase 1에서 생성된 안전한 터널을 기반으로 패킷을 암호화/인증할 실질적인 SA를 협의하고 패킷을 암호화하여 전송할 터널을 생성하는 것
  - IPsec: Internet Protocol Security
  - SA: Security Association

Source:
https://networklessons.com/security/ipsec-internet-protocol-security
https://aws-hyoh.tistory.com/165
https://blog.naver.com/wnrjsxo/221077780557
https://www.watchguard.com/help/docs/help-center/en-us/Content/en-US/Fireware/mvpn/general/ipsec_vpn_negotiations_c.html

IKE phase 1 tunnel (i.e., ISAKMP tunnel)

- The collection of parameters that the two devices will use is called a SA (Security Association).
- Here's an example of two routers that have established the IKE phase 1 tunnel:

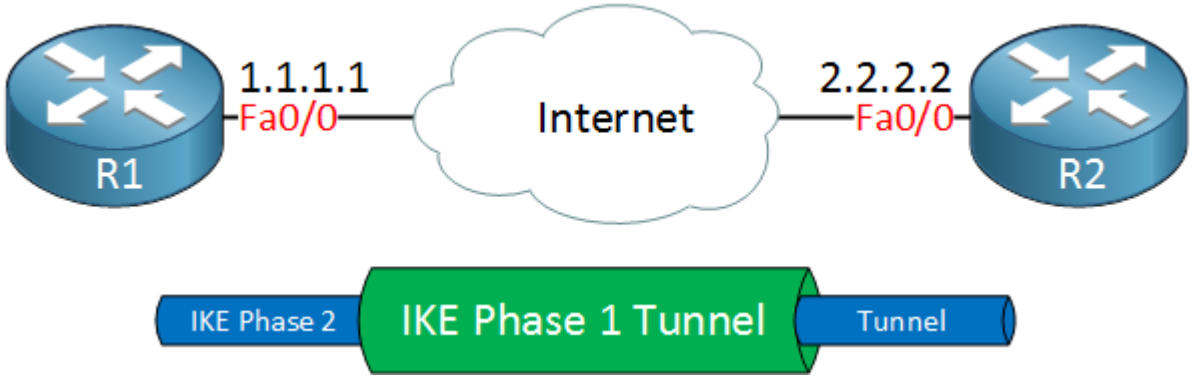


- The IKE phase 1 tunnel is only used for management traffic. We use this tunnel as a secure method to establish the second tunnel called the IKE phase 2 tunnel or IPsec tunnel and for management traffic like keepalives.

Source:
https://networklessons.com/security/ipsec-internet-protocol-security

IKE phase 2 tunnel (i.e., IPsec tunnel)

- Here's a picture of our two routers that completed IKE phase 2:

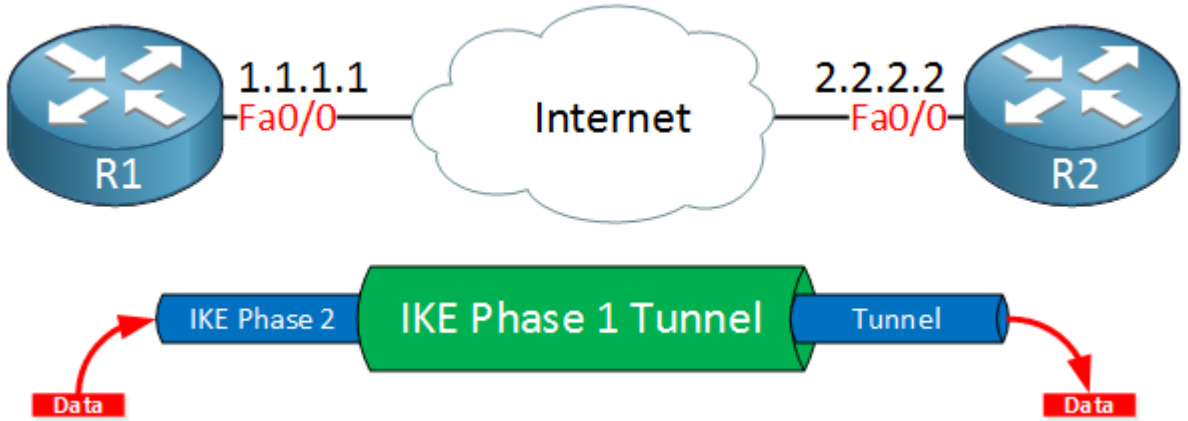


- Once IKE phase 2 is completed, we have an IKE phase 2 tunnel (or IPsec tunnel) that we can use to protect our user data.

Source:
https://networklessons.com/security/ipsec-internet-protocol-security

IKE phase 2 tunnel (i.e., IPsec tunnel)

- This user data will be sent through the IKE phase 2 tunnel:

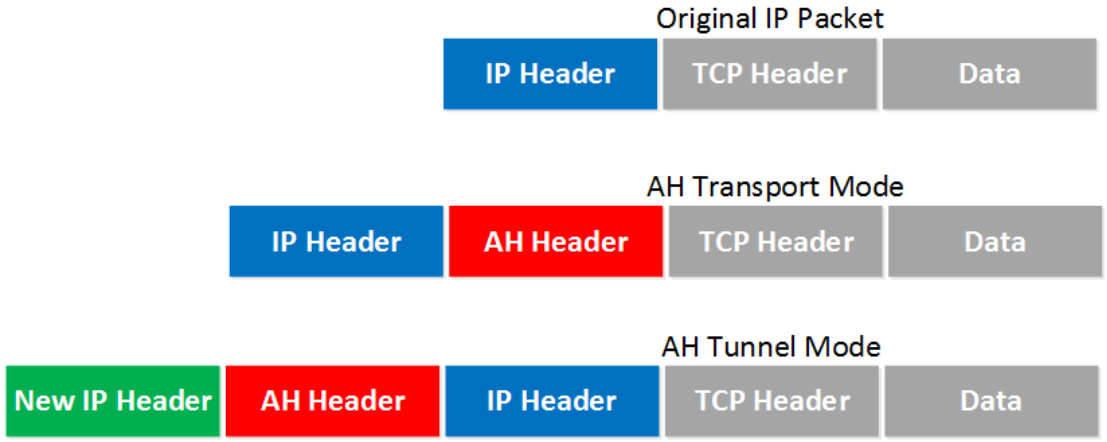


Source:
https://networklessons.com/security/ipsec-internet-protocol-security

Two other protocols for authentication and encryption

- IKE builds the tunnels for us but it doesn't authenticate or encrypt user data.
- We use two other protocols for this:
  - AH (Authentication Header)
  - ESP (Encapsulating Security Payload)
- AH and ESP both offer authentication and integrity but only ESP supports encryption. Because of this, ESP is the most popular choice nowadays.

- Both protocols support two different modes: Transport mode, Tunnel mode
- The main difference between the two is that with transport mode we will use the original IP header while in tunnel mode, we use a new IP header. Here's an example to help you visualize this:



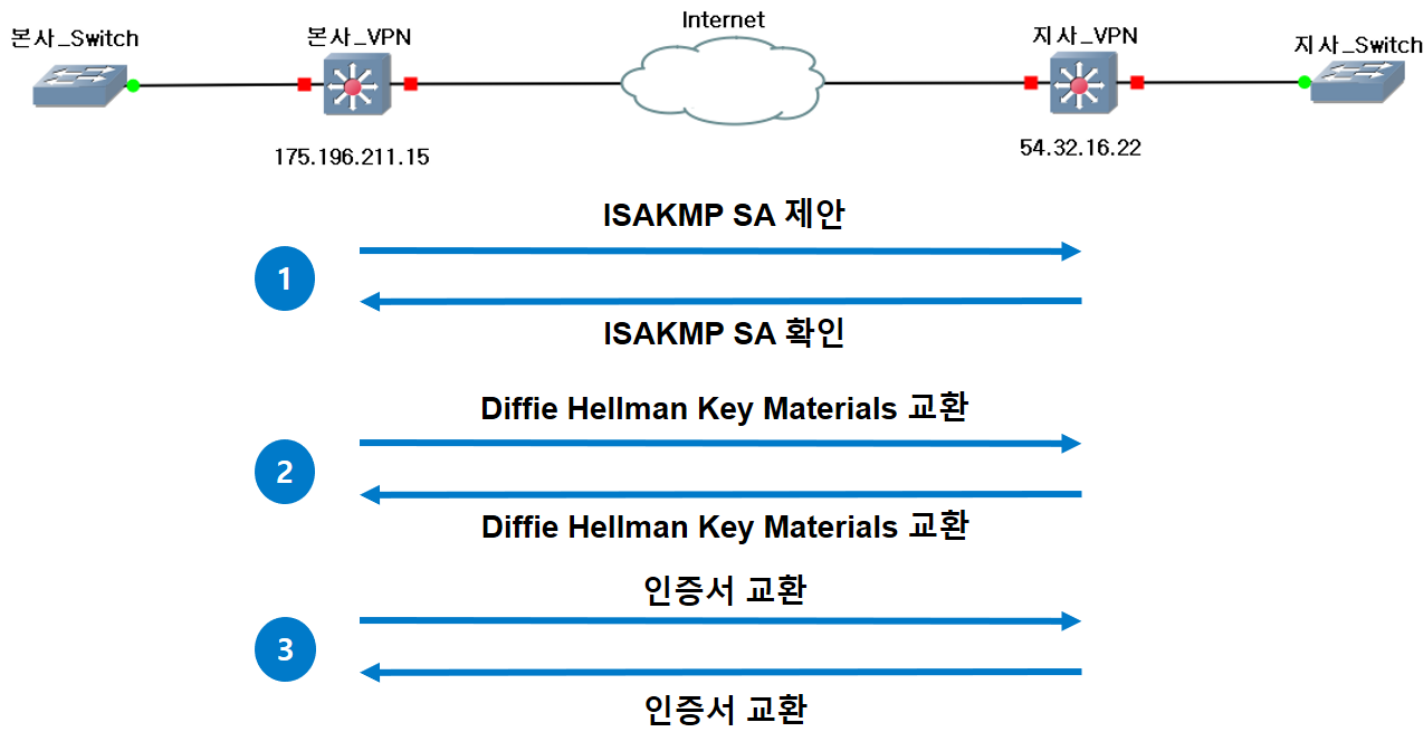
Source:
https://networklessons.com/security/ipsec-internet-protocol-security

IKE (internet Key Exchange)

- IKE (Internet Key Exchange) is one of the primary protocols for IPsec since it establishes the security association between two peers. There are two versions of IKE:
  - IKEv1
  - IKEv2

- IKEv1 was introduced around 1998 and superseded by IKEv2 in 2005. There are some differences between the two versions:
  - IKEv2 requires less bandwidth than IKEv1.
  - IKEv2 supports EAP (Extensible Authentication Protocol) authentication (next to pre-shared keys and digital certificates).
  - IKEv2 has built-in support for NAT traversal (required when your IPsec peer is behind a NAT router).
  - IKEv2 has a built-in keepalive mechanism for tunnels.

IKE phase 1 (i.e., ISAKMP SA) 교환 과정



① VPN 간 사용 가능한 SA를 협의합니다. 한쪽 VPN이 먼저 ISAKMP 세트(다음 Page에서 언급하는 5개의 SA)를 제안하면 상대방 VPN이 이를 확인한 후, 자신이 사용 가능한 ISAKMP 세트 중 일치하는 것을 확인하여 답변을 보냅니다.
② ①에서 결정된 Diffie-Hellman(DH) 키 교환 알고리즘을 사용, 키 재료(Key Materials)를 교환하여 VPN 간 공통적인 대칭키를 생성합니다. 이는 인증 정보 암호화 및 Phase 2의 대칭키 생성에 사용됩니다.
③ ①에서 결정된 해쉬 알고리즘 및 암호화 알고리즘, ②에서 생성된 대칭키를 이용하여 인증 정보를 암호화하여 서로에게 전송함으로써 인증 과정을 마치고 서로를 인증하며 터널을 생성합니다.

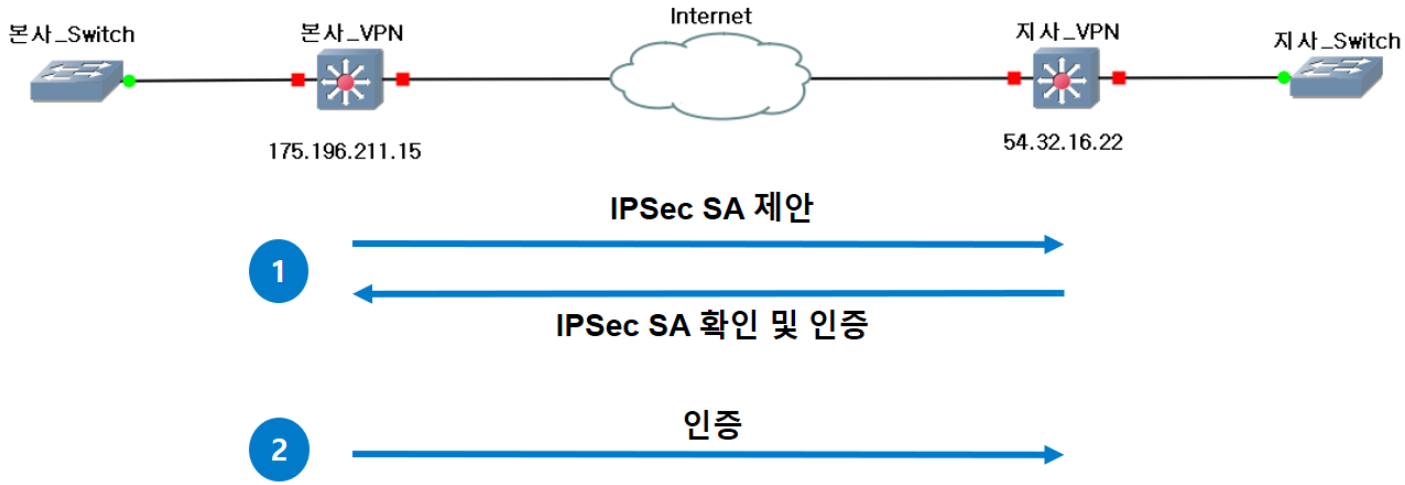
Source: https://aws-hyoh.tistory.com/165

참고, IKE phase 1 (i.e., ISAKMP SA) 알고리즘

관련 알고리즘

- Hash: 인증 정보 교환 시 인증 정보가 변질되지 않았음을 증명하기 위한 해쉬 코드 첨부에 사용되는 해쉬 알고리즘(MD5, SHA)
- Authentication : 상대방 VPN을 인증하기 위한 방법(Pre-shared Key, RSA Encryption, RSA Signature)
- DH(Diffie-Hellman) Group : 인증 정보를 암호화할 키를 생성하는 대칭키 교환 알고리즘으로 Phase 2에서 사용(DH Group 1, DH Group 2, DH Group 5)
- Lifetime : Phase 1 Tunnel이 유지되는 시간, 다시 말해 새로운 키를 생성하는 주기를 의미(통상적으로 86400초)
- Encryption : 키 교환 알고리즘과 함께 인증 정보를 암호화할 암호화 알고리즘(AES, DES, 3DES)

IKE phase 2 (i.e., IPsec SA) 교환 과정



① 한쪽 VPN이 먼저 IPsec 세트(위에서 언급한 6가지 SA)를 제안하면 상대방 VPN이 이를 확인한 후, 자신이 사용 가능한 IPsec 세트 중 일치하는 것을 확인하여 답변을 보냅니다. 여기에 패킷(데이터)에 실질적인 영향을 주는 AH / ESP, Transport Mode / Tunnel Mode, 암호화 알고리즘, 해쉬 알고리즘, PFS 등이 포함됩니다. 말 그대로 IPsec VPN의 사용 의의에 해당하는 SA를 협의하는 과정이라고 할 수 있죠.

(참고) 랜덤 난수 값인 'Nonce'를 교환합니다. 상대방 VPN이 IPsec SA 확인 답변과 함께 'Nonce'를 반대편 VPN에 전송합니다. 이 'Nonce'는 새로운 대칭키를 생성하기 위한 키 재료로 사용될 뿐만 아니라 재전송 공격(Replay Attack)을 방지하기 위한 수단으로 사용됩니다.

② 한쪽 VPN이 마찬가지로 'Nonce'를 상대방 VPN에게 전송합니다. 그리하여 새로운 대칭키가 탄생하고 이 대칭키는 패킷을 암호화하는 데 사용됩니다.

Source: https://aws-hyoh.tistory.com/165

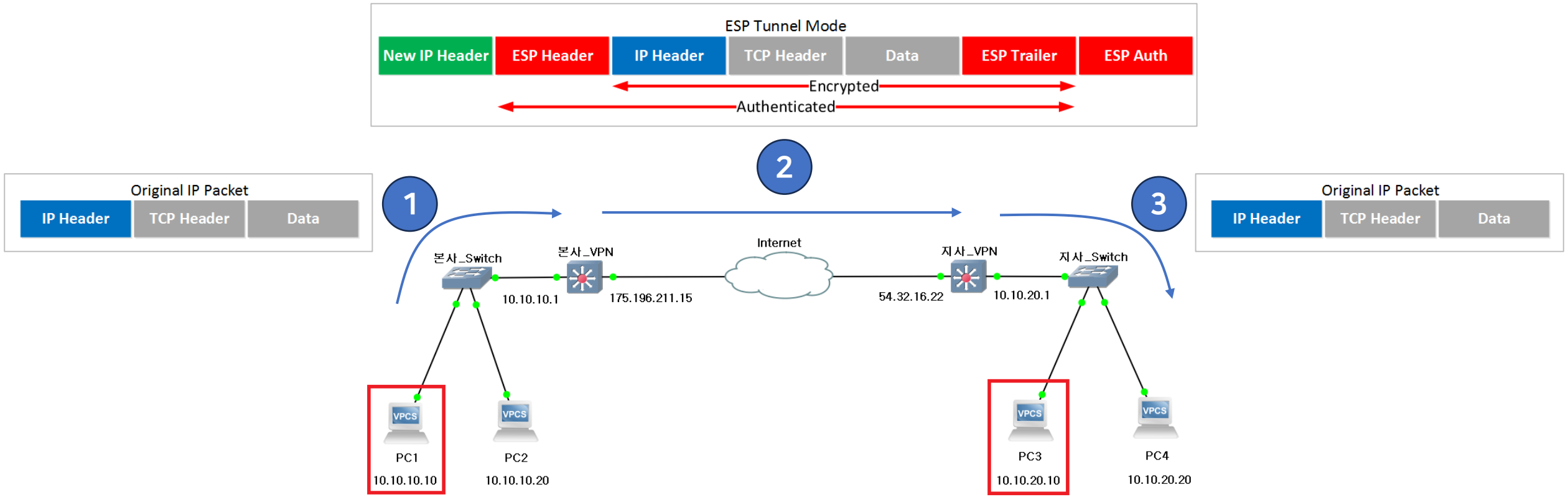
참고, IKE phase 2 (i.e., IPsec SA) 알고리즘

관련 알고리즘

- IPsec Protocol: 패킷 인증/암호화를 위한 프로토콜 헤더 선택(AH / ESP)
- Encapsulation Mode: IPsec 터널의 운용 모드 선택(Transport / Tunnel)
- Encryption : 패킷을 암호화할 암호화 알고리즘 선택(AES, DES, 3DES)
- Authentication : 패킷을 인증할 해쉬 알고리즘 선택(MD5, SHA)
- Lifetime : Phase 2 Tunnel이 유지되는 시간, 다시 말해 Phase 1의 대칭키를 기반으로 키를 재생성하는 주기를 의미(Phase 1보다 낮게 책정, < 86400초)
- Perfect Forward Secrecy(PFS, Option) : 키를 주기적으로 교환하도록 강제하는 기능 설정, 옵션 사항

Packet Forwarding Process in IPsec VPN

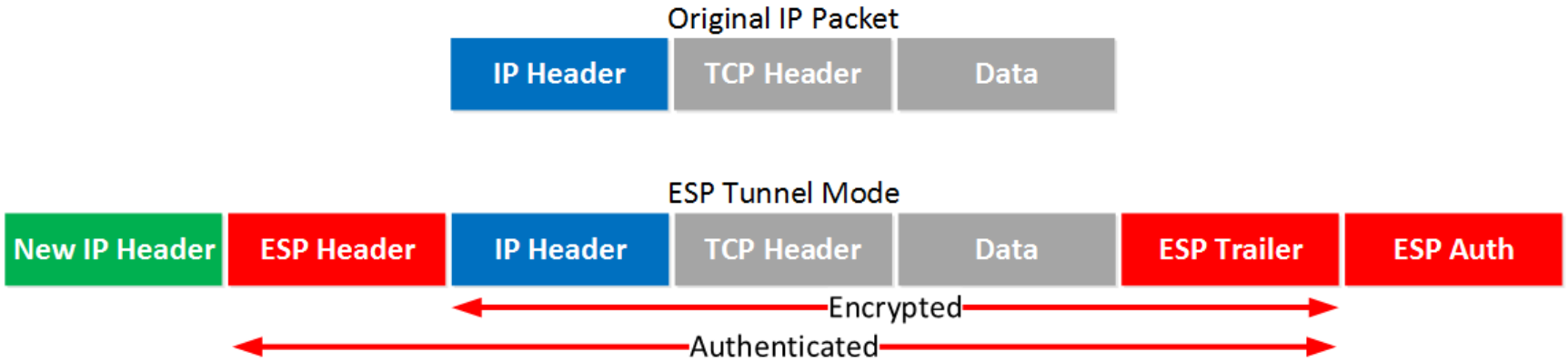
IP Header의 변화



Source: https://aws-hyoh.tistory.com/165

ESP in Tunnel Mode

- This is where we use a new IP header which is useful for site-to-site VPNs



- It's similar to transport mode but we add a new header. The original IP header is now also encrypted.
- Here's what it looks like in wireshark:

```
Frame 2: 182 bytes on wire (1456 bits), 182 bytes captured (1456 bits) on interface 0
Ethernet II, Src: Cisco_8b:36:d0 (00:1d:a1:8b:36:d0), Dst: Cisco_ed:7a:f0 (00:17:5a:ed:7a:f0)
Internet Protocol Version 4, Src: 192.168.12.1 (192.168.12.1), Dst: 192.168.12.2 (192.168.12.2)
    Version: 4
    Header Length: 20 bytes
    Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))
    Total Length: 168
    Identification: 0x023e (574)
    Flags: 0x00
    Fragment offset: 0
    Time to live: 255
    Protocol: Encap Security Payload (50)
    Header checksum: 0x1f92 [validation disabled]
    Source: 192.168.12.1 (192.168.12.1)
    Destination: 192.168.12.2 (192.168.12.2)
    [Source GeoIP: Unknown]
    [Destination GeoIP: Unknown]
Encapsulating Security Payload
    ESP SPI: 0x8bb181a7 (2343666087)
    ESP Sequence: 5
```

Source:
https://networklessons.com/security/ipsec-internet-protocol-security

Inside tunnel IPv4 CIDR

- The Inside Tunnel IPv4 CIDR is used to assign logical IP addresses to each end of the VPN tunnel.
- These IPs are essential for establishing BGP sessions and enabling communication across the tunnel, while also serving as identifiers for managing and monitoring the tunnel interface.

- You can specify a size /30 CIDR block from the 169.254.0.0/16 range.
- The CIDR block must be unique across all Site-to-site VPN connections that use the same virtual private gateway.
  - Note: The CIDR block does not need to be unique across all connections on a transit gateway. However, if they are not unique, it can create a conflict on your customer gateway. Proceed carefully when re-using the same CIDR block on multiple Site-to-site VPN connections on a transit gateway.

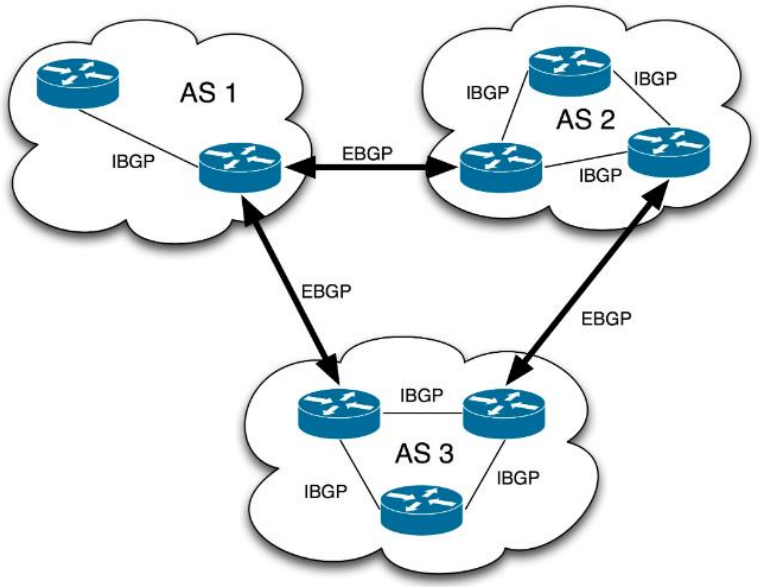
- The following CIDR blocks are reserved and cannot be used:
  - 169.254.0.0/30, 169.254.1.0/30, 169.254.2.0/30, 169.254.3.0/30, 169.254.4.0/30, 169.254.5.0/30, 169.254.169.252/30

Source:
https://docs.aws.amazon.com/ko_kr/vpn/latest/s2svpn/VPNTunnels.html

An Overview of BGP

- BGP (Border Gateway Protocol) is a widely-used routing protocol that enables the exchange of routing information between different networks on the Internet.
- BGP is designed to provide highly scalable and reliable routing for large-scale networks.
- BGP is the de-facto standard for inter-domain routing.

- RFC 1771에 정의됨
- Autonomous System (AS) 간 네트워크 정보 교환을 위한 정책 기반의 Protocol
- AS 사이에서 Looping 없는 Interdomain Routing을 가능하게 만드는 Protocol
- eBGP: 서로 다른 AS간 정보 교환
- iBGP: 동일한 AS간 정보 교환



Source:
https://repost.aws/articles/ARIKYhXEYyQQqtO2ulKERrbw/bgp-negotiation-over-aws-site-to-site-vpn-and-direct-connect-troubleshooting-strategies-for-efficient-networking
https://louis-j.tistory.com/entry/BGP-BGPBorder-Gateway-Protocol%EC%9D%98-%EA%B0%9C%EB%85%90%EA%B3%BC-%EB%8F%99%EC%9E%91-%EC%9B%90%EB%A6%AC

Static routing and Dynamic routing

특징	Static Routing	Dynamic Routing
설정 방식	수동	자동 (라우팅 프로토콜 사용)
유지보수	변경 시 수동 수정 필요	자동 업데이트
확장성	낮음	높음
사용 환경	단순한 네트워크	복잡한 네트워크, 멀티클라우드, 클라우드 환경
예제	직접 Route 추가	BGP, OSPF 등 사용

- Dynamic Routing
  - 라우팅 프로토콜(BGP, Boarder Gateway Protocol)을 사용하여 자동으로 경로를 학습 및 업데이트
  - 네트워크 토폴로지가 변경되면 자동으로 적절한 경로를 선택
  - 설정이 복잡할 수 있으나, 대규모 네트워크에서 자동화 및 유연성을 제공
  - 클라우드 환경, 멀티 클라우드 VPN, 고가용성 네트워크에서 주로 사용
- Static Routing
  - 관리자가 수동으로 라우팅 테이블에 경로를 설정
  - 네트워크 변경이 발생하면 직접 수정 필요
  - 설정이 간단하고 예측이 가능하지만, 수동 변경이 필요하므로 확장성이 낮음
  - 소규모 네트워크 또는 변경이 적은 환경에 적합

- The type of routing that you select can depend on the make and model of your customer gateway device.
  - If your customer gateway device supports Border Gateway Protocol (BGP), specify dynamic routing when you configure your Site-to-site VPN connection.
  - If your customer gateway device does not support BGP, specify static routing.

Source:
https://docs.aws.amazon.com/vpn/latest/s2svpn/vpn-static-dynamic.html

멀티 클라우드에 진심인 사람들의 이야기

전세계 클라우드를 내 손안에, 멀티 클라우드

Cloud-Barista Community 11th Conference

감사합니다.

https://github.com/cloud-barista
https://cloud-barista.github.io

김 윤 곤 / yunkon.kim@etri.re.kr