

4장

Virtual Network 관리

전체 내용

Azure
Networking 종류

Virtual Network
구성하기

NSG와 EndPoint
구성하기

VPN 구현하기

User Defined
Route 구성하기

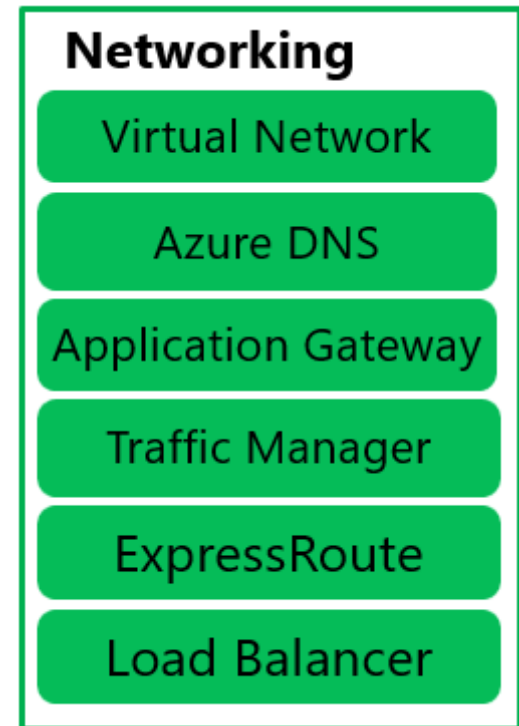
Network
Peering 구성하기

Load Balancer
구성하기

Traffic Manager
구성하기

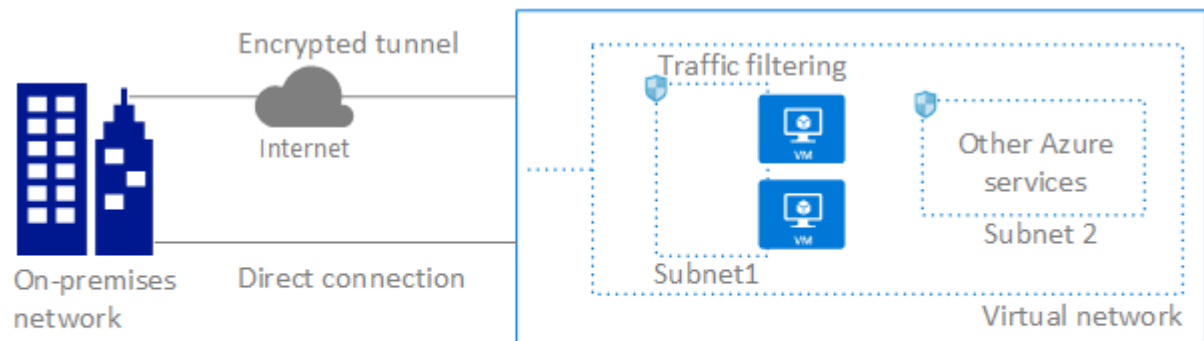
1-Azure Networking 종류

- Virtual Network
 - Network Switch 역할
- Azure DNS
 - DNS 역할
- Traffic Manager
 - Traffic 분산
- Load Balancer
 - VM들에게 몰리는 부하를 분산하기
- Application Gateway
 - Layer 7까지 Load Balance 기능 제공
- ExpressRoute
 - 고속으로 Azure에 접근하기
- Azure에서 제공하는 Load Balance 서비스 구분하기
- Load Balancer와 Application Gateway의 차이점



1-Azure Networking 종류

- Virtual Network(a.k.a. VNet)
 - Virtual Network(VNet)는 Azure에 있는 **Private Network**다
 - 같은 VNet(동일한 Switch)에 속한 VM들간에는 내부적으로 통신 가능
 - 기본적으로 **다른 VNet에 속한 VM들간의 내부 통신은 불가**.
즉, 다른 Switch에 속한 VM들간에는 통신이 안되는 것이 당연하다
 - 하지만 Switch끼리 Cable을 연결하면 통신이 가능한 것처럼 **VNet간에도 VPN 연결**을 하든가, 아니면 **같은 지역에 속한 VNet간에는 Network Peering**을 통하여 내부 통신이 가능하게 할 수도 있다



1-Azure Networking 종류

- Azure DNS

- 기본적으로 Azure Infra를 사용하는 VM들이 요청하는 Host Name에 대하여 IP Address를 제공한다
 - VM들이 Azure DNS를 이용하는 경우에는 Internet에 액세스는 할 수 없고, 다만 Azure Infra에 있는 VM 및 Web site에 대해서만 액세스가 가능하다
- 뿐 만 아니라, Public Domain에 대하여도 DNS Zone을 제공하여 `www.myazure.kr`에 대한 요청에 대하여 Public IP Address를 알려주어 회사 Home Page에 접속하도록 할 수 있다
 - 이를 위해서는 Domain 등록 대행 기관에서 만든 도메인을 Azure로 관리를 이관하도록 작업을 해야 한다

The screenshot shows the Azure portal interface for managing a DNS zone named 'myazure.kr'. The left sidebar contains navigation links for Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, and SETTINGS (Properties, Locks). The main content area displays the 'Essentials' section with the following information:

- Resource group: [korRG](#) (change)
- Subscription name: [MSDN Platforms](#) (change)
- Subscription ID: 72d8cc8e-59cd-4176-905e-6fa0dc7f3999

Below this, the 'Name servers' are listed:

- Name server 1: ns1-09.azure-dns.com.
- Name server 2: ns2-09.azure-dns.net.
- Name server 3: ns3-09.azure-dns.org.
- Name server 4: ns4-09.azure-dns.info.

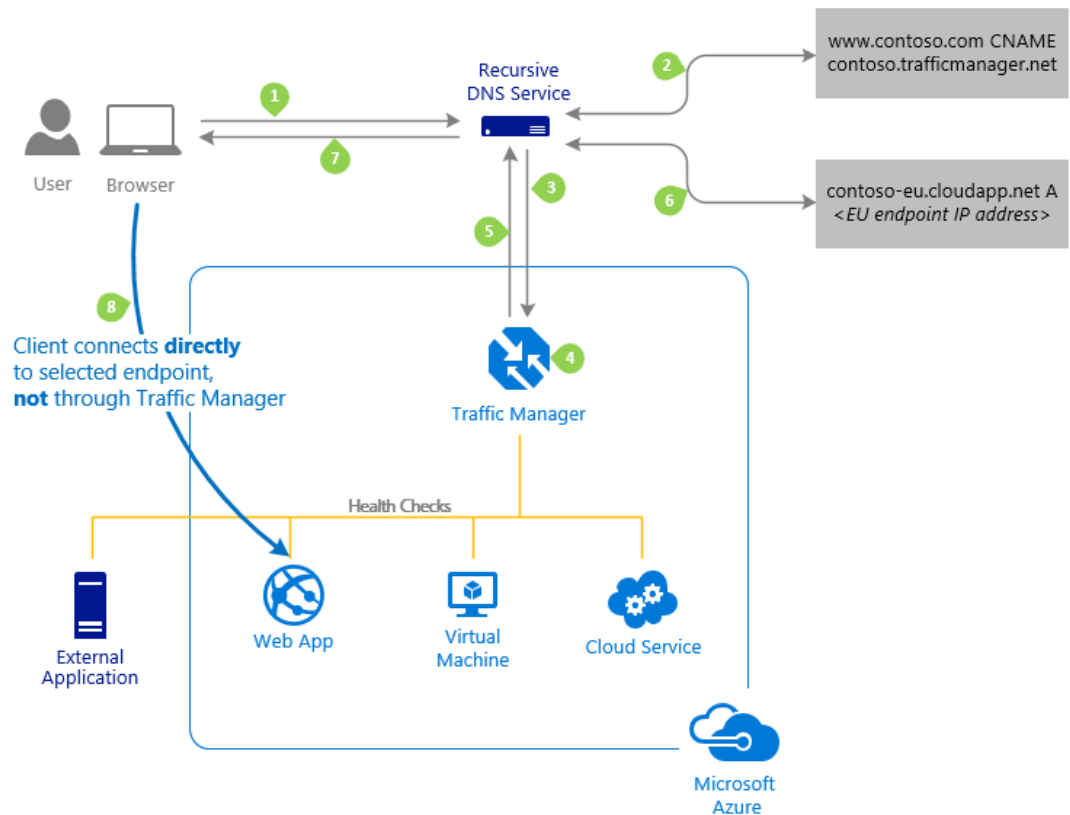
The 'Record sets' section is visible below, with a search bar and a table showing the '@' record set:

NAME	TYPE	TTL	VALUE
@	NS	172800	ns1-09.azure-dns.com. ns2-09.azure-dns.net. ns3-09.azure-dns.org. ns4-09.azure-dns.info.

1-Azure Networking 종류

- Traffic Manager

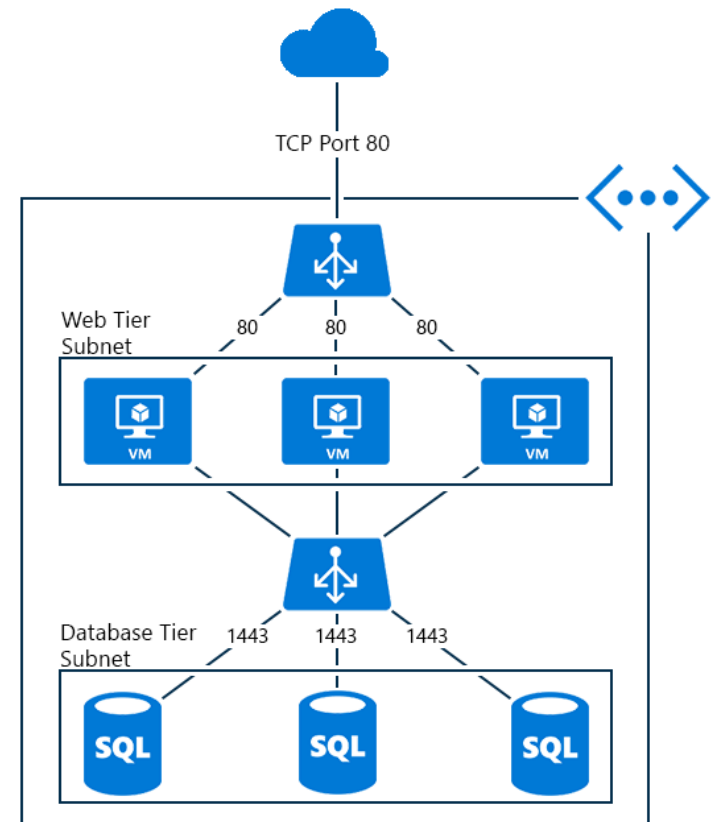
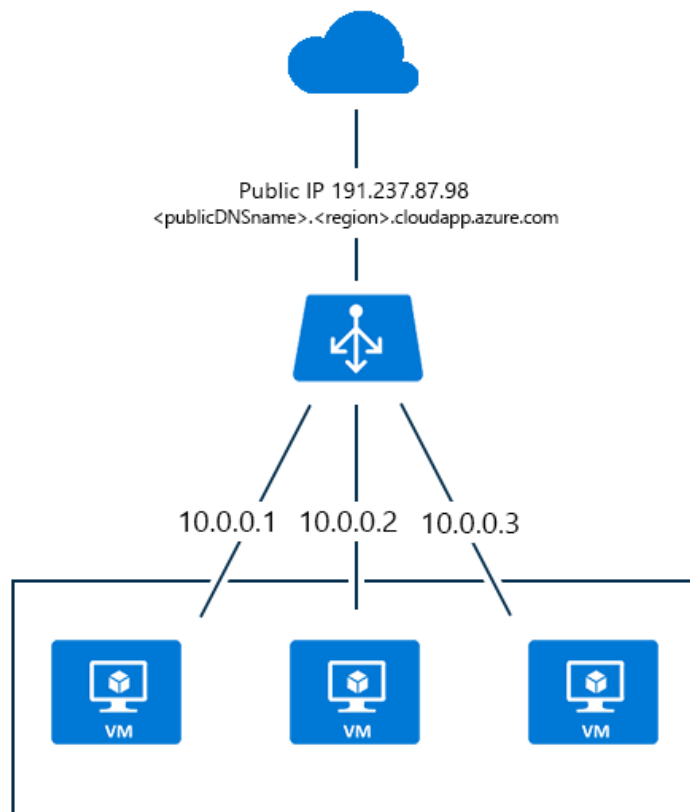
- 사용자가 Web Browser에서 `www.contoso.com`을 입력했을 때 사용자와 가장 가까운 곳에 있는 Web site로 연결해주는 역할을 한다
 - Performance
 - Priority
 - Weighted



1-Azure Networking 종류

- Load Balancer

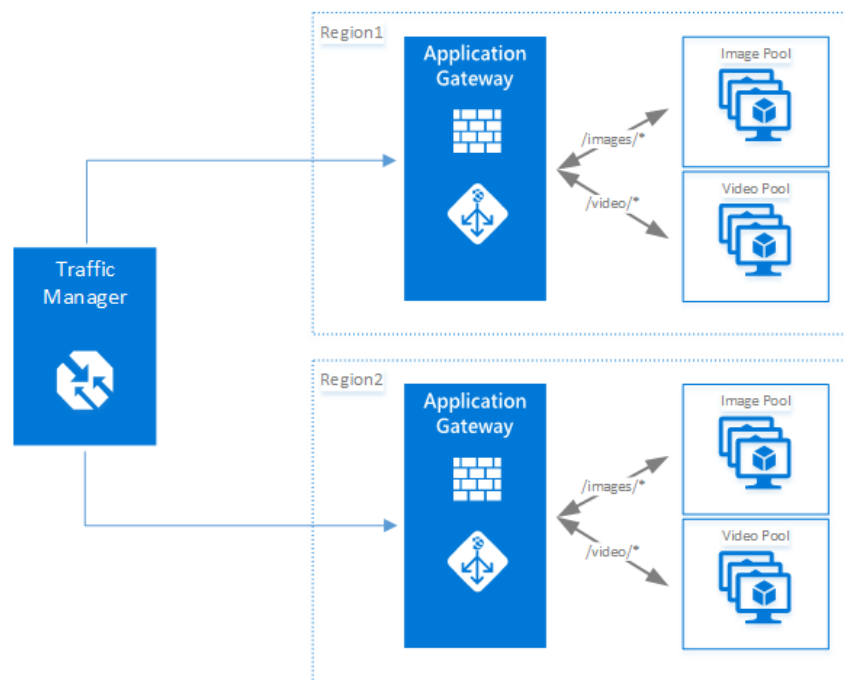
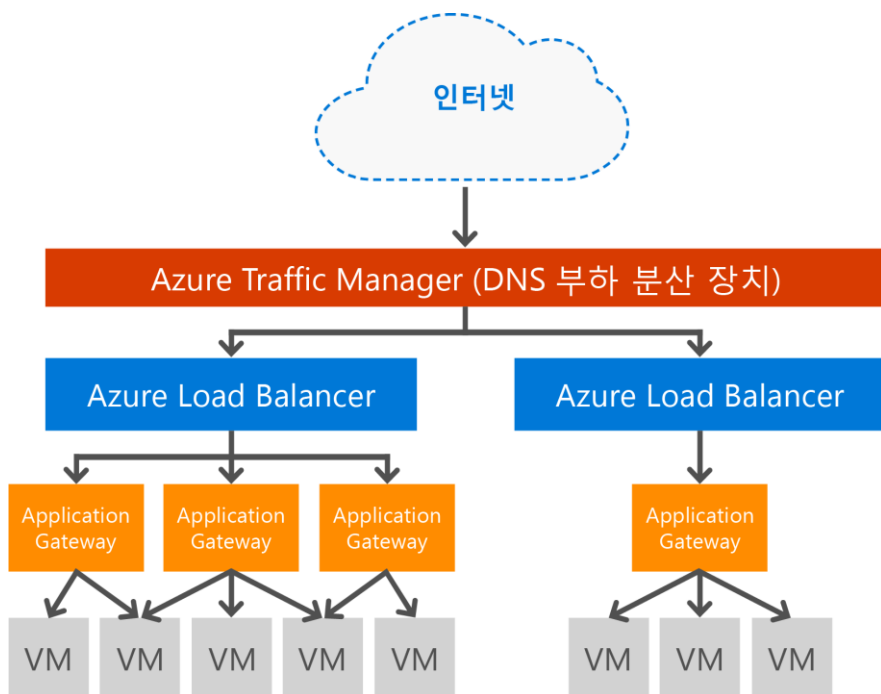
- Internet용 및 Internal용 Load Balancer를 제공하며, 3-Tier 모델에서 사용할 수 있다
- 특정한 포트로 서비스하는 VM들에게 부하를 분산해준다



1-Azure Networking 종류

- Application Gateway

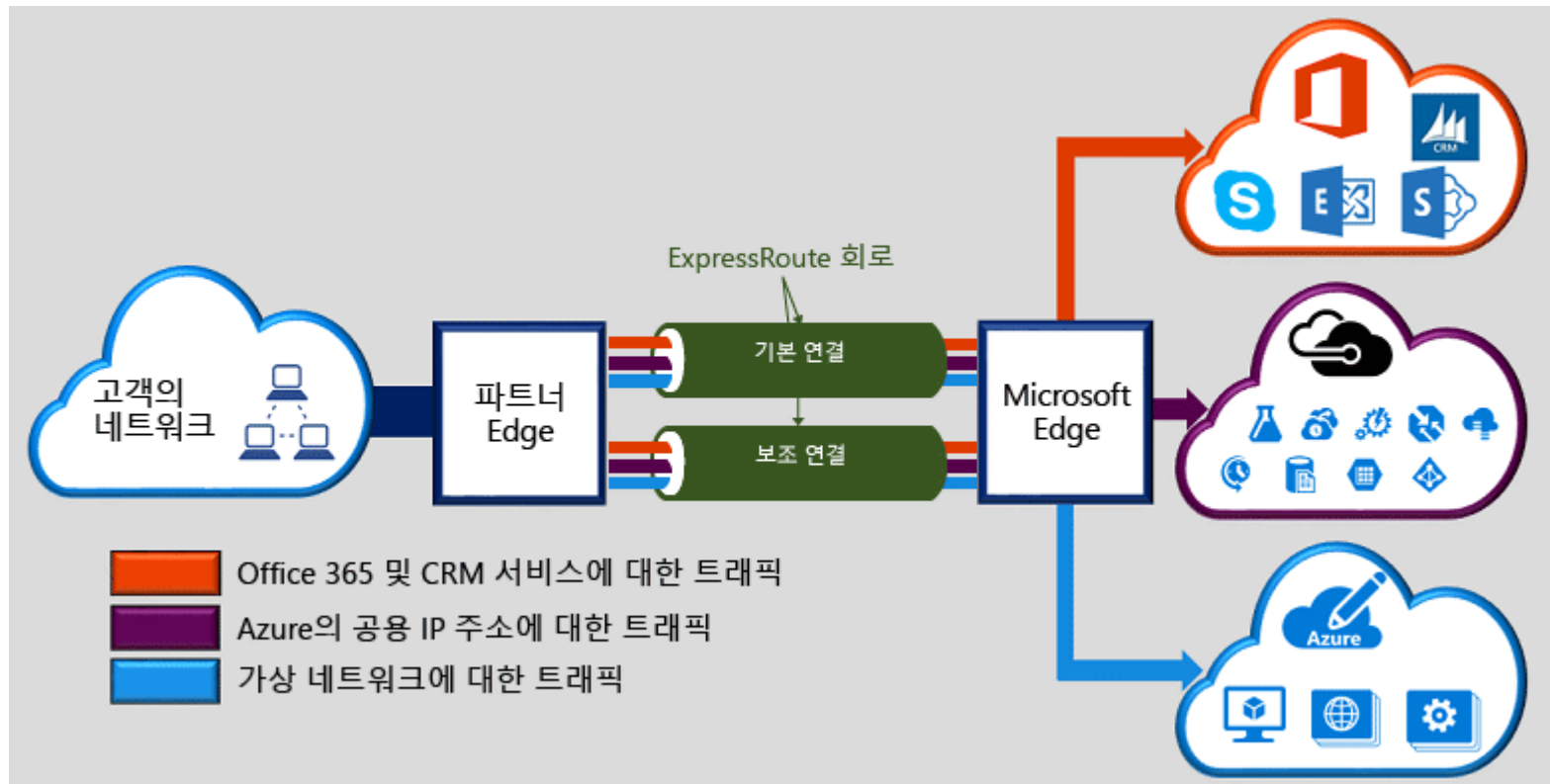
- Application을 위해 Layer 7까지 Load Balance를 제공한다
 - CPU 집약적인 SSL 종료를 Offload하여 Web Farm 생산성을 최적화
 - Inbound Traffic의 round-robin load distribution, 쿠키 기반 세션 선호도, URL 경로 기반 라우팅 및 단일 Application Gateway 뒤에 여러 웹 사이트를 호스트할 수 있다



1-Azure Networking 종류

- ExpressRoute

- 사내 네트워크가 국내 ISP 업체를 통하여 전용 네트워크를 사용하여 고속으로 Azure, Office 365, CRM Online 등등의 서비스를 이용할 수 있도록 한다



1-Azure Networking 종류

- Azure에서 제공하는 Load Balance 서비스 구분하기
 - Azure Load Balancer
 - Layer 4까지 담당하고, 동일한 Azure 데이터 센터에서 실행되는 응용 프로그램 인스턴스 간에 네트워크 수준 트래픽 분산을 제공
 - Application Gateway
 - Layer 7까지 담당하고, Reverse Proxy Service로서 클라이언트 연결을 종료하고 Backend Endpoint로 요청을 전달
 - Traffic Manager
 - DNS 응답을 사용하여, 요청한 Client에게 전역으로 분산된 Endpoint(App service, Cloud Service, Public IP)에 대한 IP Address를 제공하여, 최종적으로 Client가 직접 Endpoint에 연결하게 된다

1-Azure Networking 종류

- Azure에서 제공하는 Load Balance 서비스 구분하기

부여	Azure Load Balancer	응용 프로그램 게이트웨이	트래픽 관리자
기술	전송 수준(계층 4)	응용 프로그램 수준(계층 7)	DNS 수준
지원되는 응용 프로그램 프로토콜	모두	HTTP, HTTPS 및 WebSockets	모두(HTTP 끝점은 끝점 모니터링에 필요함)
끝점	Azure VM 및 클라우드 서비스 역할 인스턴스	모든 Azure 내부 IP 주소, 공용 인터넷 IP 주소, Azure VM 또는 Azure 클라우드 서비스	Azure VM, 클라우드 서비스, Azure 웹앱 및 외부 끝점
Vnet 지원	인터넷 연결 및 내부(Vnet) 응용 프로그램 모두에 사용할 수 있습니다.	인터넷 연결 및 내부(Vnet) 응용 프로그램 모두에 사용할 수 있습니다.	인터넷 연결 응용 프로그램만 지원
끝점 모니터링	프로브를 통해 지원됨	프로브를 통해 지원됨	HTTP/HTTPS GET을 통해 지원됨

1-Azure Networking 종류

- Load Balancer와 Application Gateway의 차이점

형식	Azure Load Balancer	응용 프로그램 게이트웨이
프로토콜	UDP/TCP	HTTP, HTTPS 및 WebSockets
IP 예약	지원됨	지원되지 않음
부하 분산 모드	5 튜플(원본 IP, 원본 포트, 대상 IP, 대상 포트, 프로토콜 유형)	라운드 로빈 URL 기반 라우팅
부하 분산 모드(원본 IP/고정 세션)	2 튜플(원본 IP 및 대상 IP), 3 튜플(원본 IP, 대상 IP 및 포트) 가상 컴퓨터의 수에 따라 확장 또는 축소할 수 있습니다.	쿠키 기반 선호도 URL 기반 라우팅
상태 프로브	기본값: 프로브 간격 15초 회전 중단: 2번의 연속 실패 사용자 정의 프로브 지원	유휴 프로브 간격 30초 5번의 연속 라이브 트래픽 실패 또는 유휴 모드의 단일 프로브 실패 후 중단됩니다. 사용자 정의 프로브 지원
SSL 오프로딩	지원되지 않음	지원됨
Url 기반 라우팅	지원되지 않음	지원됨
SSL 정책	지원되지 않음	지원됨

2-Virtual Network 구성하기

- Virtual Network(VNet) 개요
- Azure **Classic** Portal에서의 Virtual Network 개요
- **ASM** 자원 관리 모델에서의 Virtual Network 이해
- Azure **Classic** Portal에서 VM을 생성할 때 주의 사항
- Azure Portal에서의 Virtual Network 개요
- **ARM** 자원 관리 모델에서의 Virtual Network 이해
- VNet에 연결되지 않은 VM들
- Virtual Network 생성하기

2-Virtual Network 구성하기

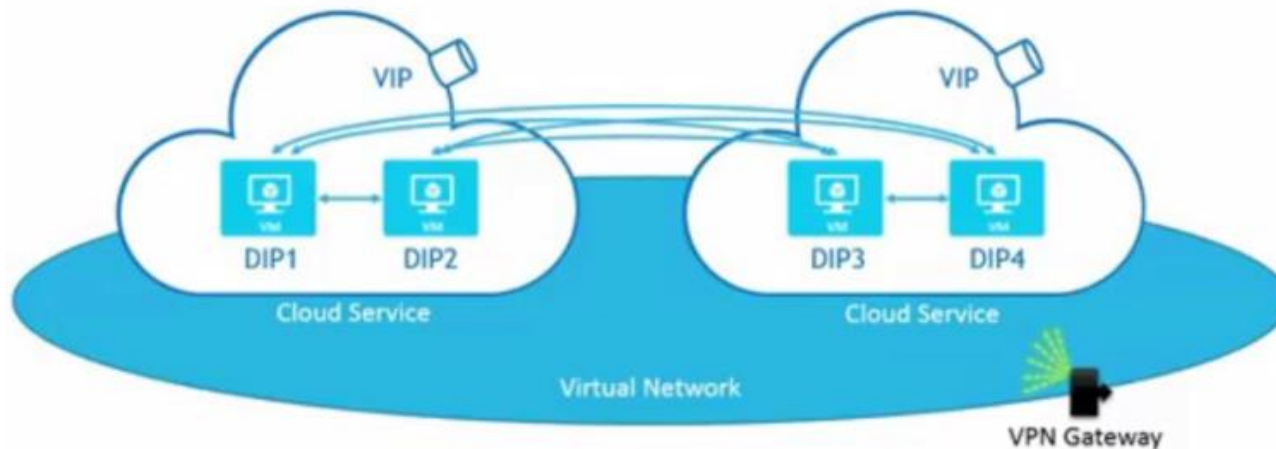
- Virtual Network(VNet) 개요

- Virtual Network(VNet)는 VM간의 연결 및 PaaS Cloud Service간의 연결에 사용된다
- VNet에서는 IP Address 할당, Security Policy, Route Table 및 DNS Setting을 한다
- Azure VM을 생성하기 전에 반드시 Virtual Network을 먼저 만들어 놓아야 편리하다-필수
- VM들이 동일한 VNet에 연결되어 있으면 서로 통신 가능하다
 - 이 때 VM들은 Subnet에서 IP Address를 받게 된다
- 사내 네트워크(컴퓨터)를 Azure VNet에 연결할 때는 VPN을 사용한다
- ExpressRoute를 사용하여 전용 사설 망을 통해 Internet에 접속하지 않는 Azure VNet에 연결하여 사내 컴퓨터와 Azure에 있는 VM간의 통신을 가능하게 한다

2-Virtual Network 구성하기

- Virtual Network(VNet) 개요
 - 지역 기반(Region-based)이다
 - Azure Classic Portal에서 VM을 생성할 때 새로운 Cloud Service(cs100)를 만드는 경우, Region/Virtual Network을 선택하게 된다
 - 선택한 지역 또는 Virtual Network에 VM을 생성하겠다는 뜻
 - 그 이후로, 새로운 VM을 생성할 때 기존 Cloud Service(cs100)를 선택하였다면 그 Cloud Service와 연결된 Region 및 Virtual Network을 강제로 이용할 수 밖에 없다
 - VM 생성할 때 Region과 Virtual Network은 동일한 항목에 있다

2-Virtual Network 구성하기



- Azure Classic Portal에서의 Virtual Network 개요
 - VNet 생성
 - DHCP로 IP Address를 VM에 할당함
 - 여러 개의 Subnet이 존재해도 동일한 VNet을 사용하는 VM들간에는 통신 가능
 - Cloud Service 생성
 - fqdn 생성 및 공인 IP(VIP) 할당됨
 - VM 생성시 기존 VNet과 Cloud Service 사용
 - 동일한 VNet에 할당되기 때문에 VM들 간의 내부 통신 가능
 - Cloud Service를 이용하므로 EndPoint로 제어하고, Load Balance 구성 가능
 - 사설 IP 할당되어 VM들간의 통신 가능

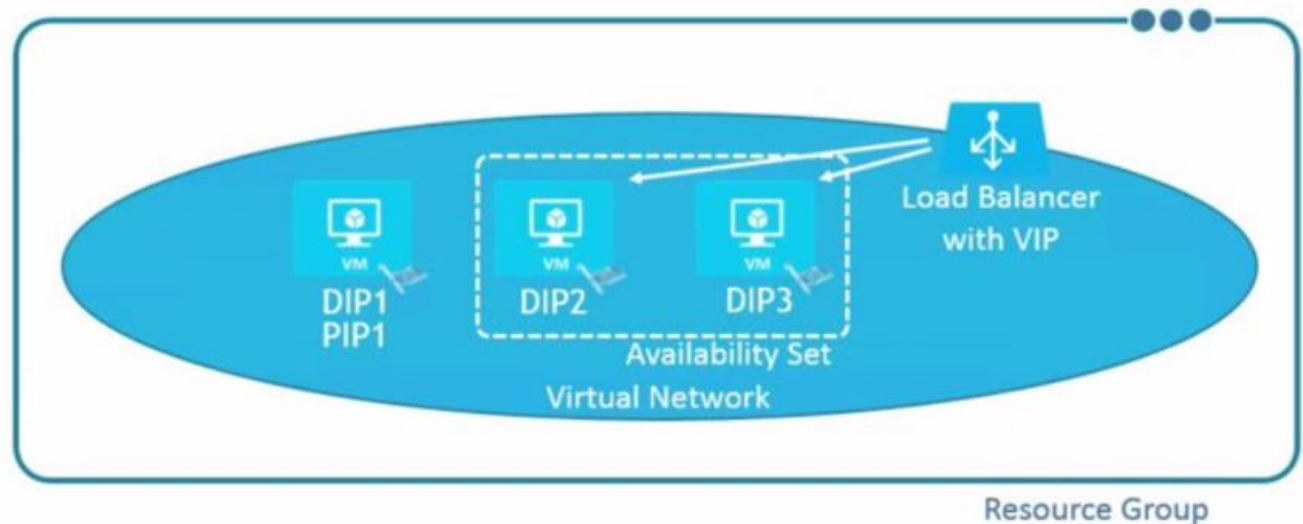
2-Virtual Network 구성하기

- **ASM** 자원 관리 모델에서의 Virtual Network 이해
 - ASM 자원 관리 모델을 사용하는 Cloud Service의 장점
 - Azure Classic Portal에서 기본적으로 VM을 생성하면 자동으로 독립된 VNet과 Cloud Service가 할당된다
 - Cloud Service가 자동으로 VM에 할당되면 인터넷으로부터의 VM을 보호하는 Proxy Server를 구축하는 효과가 난다 (공짜)
 - 이것은 바로 Endpoint로 허용된 Port로만 통신이 가능하다는 것이다
 - 동일한 Cloud Service에 연결된 VM들에 동일한 서비스(web service)를 Load Balancing을 하기 위해 조금의 설정만으로도 가능하다는 것이다(쉬움)
 - VM 없이 Cloud Service 단독으로 Web Service를 할 수도 있다

2-Virtual Network 구성하기

- Azure **Classic** Portal에서 VM을 생성할 때 주의 사항
 - Azure Classic Portal에서는, 기본적으로 VM을 생성하면 각 VM당 고유의 VNet 및 Cloud Service가 하나씩 만들어진다
 - 그 결과 VM들은 **각각 다른 VNet에 할당되어** 서로 간의 직접 통신이 불가능하고, 단지 EndPoint를 사용한 Internet으로만 통신이 가능하다
 - 각 VM들이 일반적인 EndPoint를 가지므로 인터넷으로부터의 접속 및 관리가 편리하다
 - 각 VM을 생성할 때 기존 VNet 및 Cloud Service에 연결하여 생성한다
 - 그 결과 VM들간의 직접 통신이 가능하다
 - 하지만 이 때는 VM들이 사용중인 동일한 서비스에 대한 EndPoint가 모두 다르므로 인터넷 서비스를 하는데 불편함이 있다
 - 동일한 VNet에 속한 VM들 간에는 직접 통신이 가능하다
 - 동일한 Cloud Service에 연결된 VM들은 동일한 VNet을 사용하기 때문에 직접 통신이 가능하다
 - 동일한 Cloud Service를 사용하는 VM들은 EndPoint로 관리된다

2-Virtual Network 구성하기



- Azure Portal에서의 Virtual Network 개요
- VNet 생성
 - DHCP로 IP Address를 VM에 할당함
 - 여러 개의 Subnet이 존재해도 동일한 VNet을 사용하는 VM들간에는 통신 가능
- VM 생성시 기존 VNet과 Cloud Service 사용
 - 공인 IP(VIP) 할당 및 수동으로 fqdn 생성
 - 사설 IP 할당되어 VM들간의 통신 가능
 - 동일한 VNet에 할당되기 때문에 VM들 간의 내부 통신 가능
 - Load Balancer를 붙여서 부하 분산 가능하고, 이것을 위해서는 사전에 VM을 만들 때 Availability Set에 할당해야 함

2-Virtual Network 구성하기

- **ARM** 자원 관리 모델에서의 Virtual Network 이해
 - Azure Portal에서는 기본적으로 ARM 자원 관리 모델을 사용한다
 - Virtual Network 관점에서 ARM 자원 모델의 장점
 - 일반 사내에서 Network을 구축하여 운영하는 것과 동일한 개념이어서 이해하기 쉽다
 - 사내 네트워크에서 Network을 구축하여 운영하려면 Router, Switch, VPN 장비, Proxy Server등을 구매하여 각각 설정을 해야 하지만, Azure Portal에서는 이것들을 매우 저렴한 비용으로 빠르게 구축할 수 있다
 - 하물며 Availability Set을 사용하여 Host 컴퓨터의 고장 및 RACK의 고장에서 VM이 정상적으로 운영하도록 하고 있다
 - Load Balancer로 붙여서 운영할 수 있다

2-Virtual Network 구성하기

- VNet에 연결되지 않은 VM들
 - VM들간의 직접 통신을 하려면 각 VM들이 동일한 VNet에 속해 있어야 한다
 - VNet에는 Subnet이 있고, 각 Subnet에서 Private IP를 할당한다
 - 동일한 VNet에 속한 Subnet간에는 통신이 된다(Routing 시켜준다)
 - 각 VM들이 인터넷 서비스를 원할하게 하기 위해서는(인터넷으로부터 접속 허용) 각각 고유한 Cloud Service에 연결되면 된다
 - Cloud Service 당 하나의 Public IP가 할당된다
 - Cloud Service가 동일한 VM1, VM2는 각각의 EndPoint가 다르므로 인터넷 서비스하는데는 불편함이 있다
 - VNet을 생성했다면 새로운 VM과 Cloud Service를 새로운 VNet으로 넣을 수 있다
 - 동일한 VNet에 존재하는 VM과 Cloud Service는 서로 직접 통신이 가능하다

2-Virtual Network 구성하기

- Virtual Network 생성하기
 - **New - Networking - Virtual Network - Resource Manager** 확인 - **Create** 클릭
 - VNet1_HQ라는 VNet 생성
 - VNet1_HQ에 **Second** 서브넷 생성
 - VNet2_Factory라는 VNet 추가 생성

Create virtual network

* Name
VNet1_HQ ✓

* Address space ⓘ
10.1.0.0/16
10.1.0.0 - 10.1.255.255 (65536 addresses)

* Subnet name
First ✓

* Subnet address range ⓘ
10.1.1.0/24 ✓
10.1.1.0 - 10.1.1.255 (256 addresses)

* Subscription
무료 체험 ▼

* Resource group ⓘ
☐ Create new ☒ Use existing
RGyslee ▼

* Location
Japan East ▼

3-Network Security Group과 EndPoint 구성하기

- Network security group(NSG) 구성하기
- Access Control List와 Endpoint
- NSG와 FrontEnd ACL 구분하기

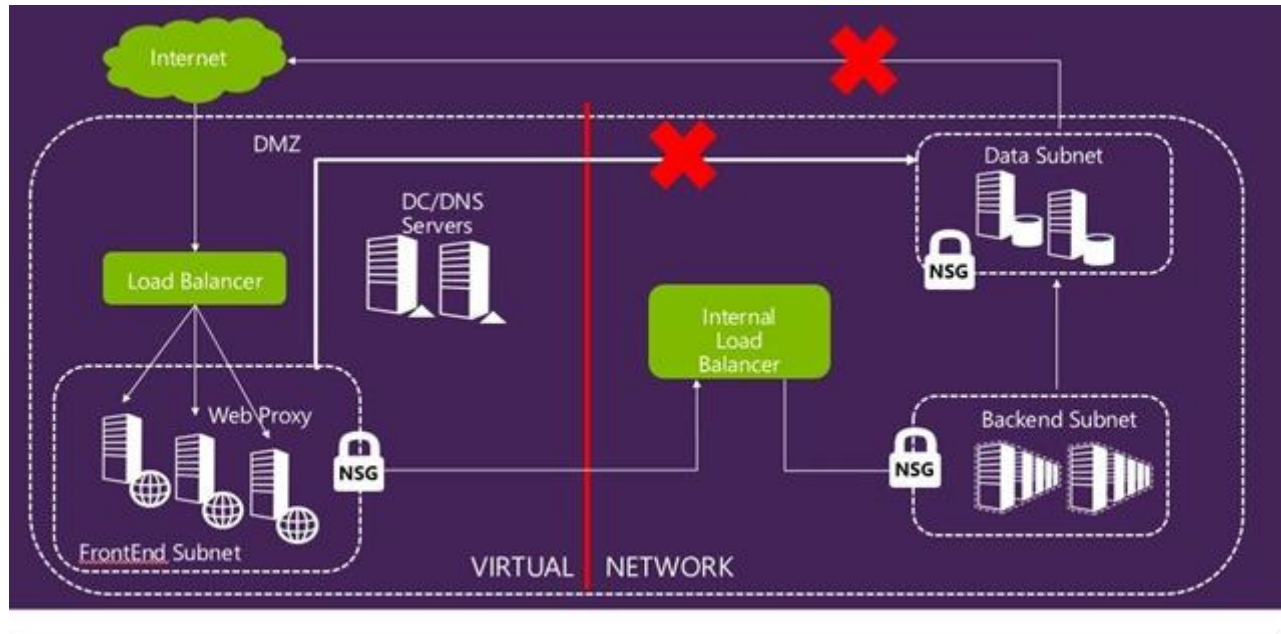
3-Network Security Group과 EndPoint 구성하기

- Network security group(NSG) 구성하기
 - VM들간에 직접 통신이 가능한 경우
 - 동일한 Virtual Network(vnyslee1)에 소속되어 있는 경우
 - 하나의 Virtual Network에 2개의 Subnet이 있다면 각 Subnet에 소속된 VM들 간에도 통신이 가능하다
 - First Subnet : vmpeace1
 - Second Subnet: vmpeace2
 - VM이 자신의 네트워크를 벗어나서 다른 곳으로 통신(Inbound, Outbound)하는 것을 통제하는 것이 Network Security Group
 - Subnet에 NSG가 적용되므로 동일한 Subnet에 있는 VM들간의 통신 가능 여부는 해당 VM의 Firewall을 통해서 설정하면 된다
 - 자신이 속한 Subnet 밖으로의 통신 통제는 NSG에서 구현하면 된다

3-Network Security Group과 EndPoint 구성하기

- Network security group(NSG) 구성하기

- 동일한 VNet에 속해 있으면서 Subnet이 다른 VM들 간의 통신을 통제하고자 할 때 Network Security Group을 사용한다
 - vnyslee1에 속한 vmpeace1는 First Subnet에 속해 있다
 - vnyslee2에 속한 vmpeace2는 Second Subnet에 속해 있다
 - 서로 간의 통신은 www만 가능하고 그 외 모든 것을 차단하고자 할 때 Network Security Group을 사용한다
 - NSG를 생성하여 특정한 Subnet 에 설정한다



3-Network Security Group과 EndPoint 구성하기

- Network security group(NSG) 구성하기
 - NSG 적용하는 곳(자원)
 - NSG는 Subnet, NIC, VM에 모두 적용할 수 있지만, 자원(VM)의 Deployment Model에 따라서 다르다.
 - **Subnet:** Both **Classic** and **Resource Manager**
 - **NIC:** **Resource Manager** deployment only
 - **VM:** **Classic** deployment only

3-Network Security Group과 EndPoint 구성하기

- Network security group(NSG) 구성하기

- NSG 구성하기

- **New - Networking - Network Security Group - Resource Manager 선택**

* Name
NSG_yslee ✓

* Subscription
무료 체험 ▼

* Resource group ⓘ
☐ Create new ☒ Use existing

RGyslee ▼

* Location
Japan East ▼

- **NSG_yslee - Outbound Security rules - 139, 445 차단**

- **NSG_yslee - Subnets - Associate 클릭 - Virtual Network - vnyslee1 - Subnet - first**

- 즉, Virtual Network가 vnyslee1을 사용하고 subnet을 first를 사용하는 VM들은 외부로 나갈 때는 139, 445번을 네트워크단에서 차단한 것이다

- **vmpeace1에 NSG 적용하기**

- Virtual Machines - vmpeace1 - Network Interfaces - 해당 NIC 선택 - Network Security Group - Edit 클릭 - 기존 NSG 클릭 - **NSG_yslee** 선택 - Save 클릭하여 적용

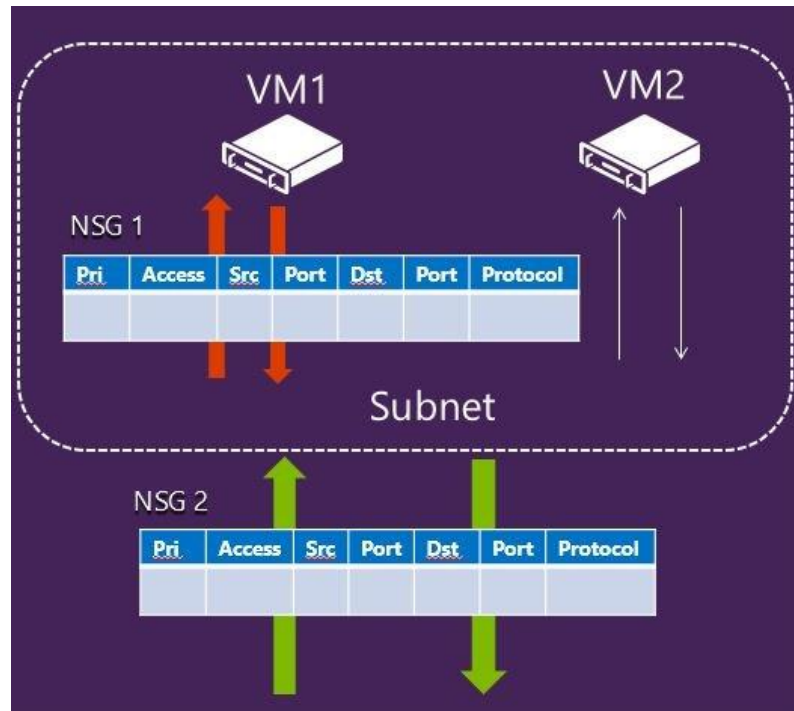
3-Network Security Group과 EndPoint 구성하기

- Access Control List와 Endpoint
 - **NSG**는 **Subnet**으로 들어오고 나가는 트래픽을 통제한다
 - VM이 공통의 VNet에 소속되어 있을 때는 NSG로 통제하면 된다
 - 한 편, **ACL**은 개별 **VM의 Endpoint**로 들어오고 나가는 트래픽을 통제한다
 - VM이 공통의 VNet을 사용하지 않을 때 유용하게 사용할 수 있다
 - VM이 생성되면 기본적으로 **모든 Incoming Traffic**를 차단하고, 관리용으로 사용되는 포트 2~3개만 허용한다
 - Endpoint는 Incoming traffic에 대해서만 관리하는 것이다
 - 접속을 허용한 Endpoint에 ACL를 적용하여 특정한 subnet에서 들어오는 것은 관리도 못하게 할 수 있다
 - EndPoint에서 ACL을 수정한다

3-Network Security Group과 EndPoint 구성하기

- NSG와 FrontEnd ACL 구분하기

- ACL은 각 VM의 EndPoint에 설정하여 접속 허용 및 거부하는 것
- NSG는 특정한 VNet에 소속된 Subnet들, 해당 Subnet의 VM들, 각 VM에 있는 여러 개의 NIC에 적용하여 네트워크 접속 허용 및 거부한다
- FrontEnd ACL이 적용된 VM에 NSG를 구현하고자 하면 실패한다. 먼저 FrontEnd를 제거한 후 NSG를 구성해야 한다



4-VPN 구현하기

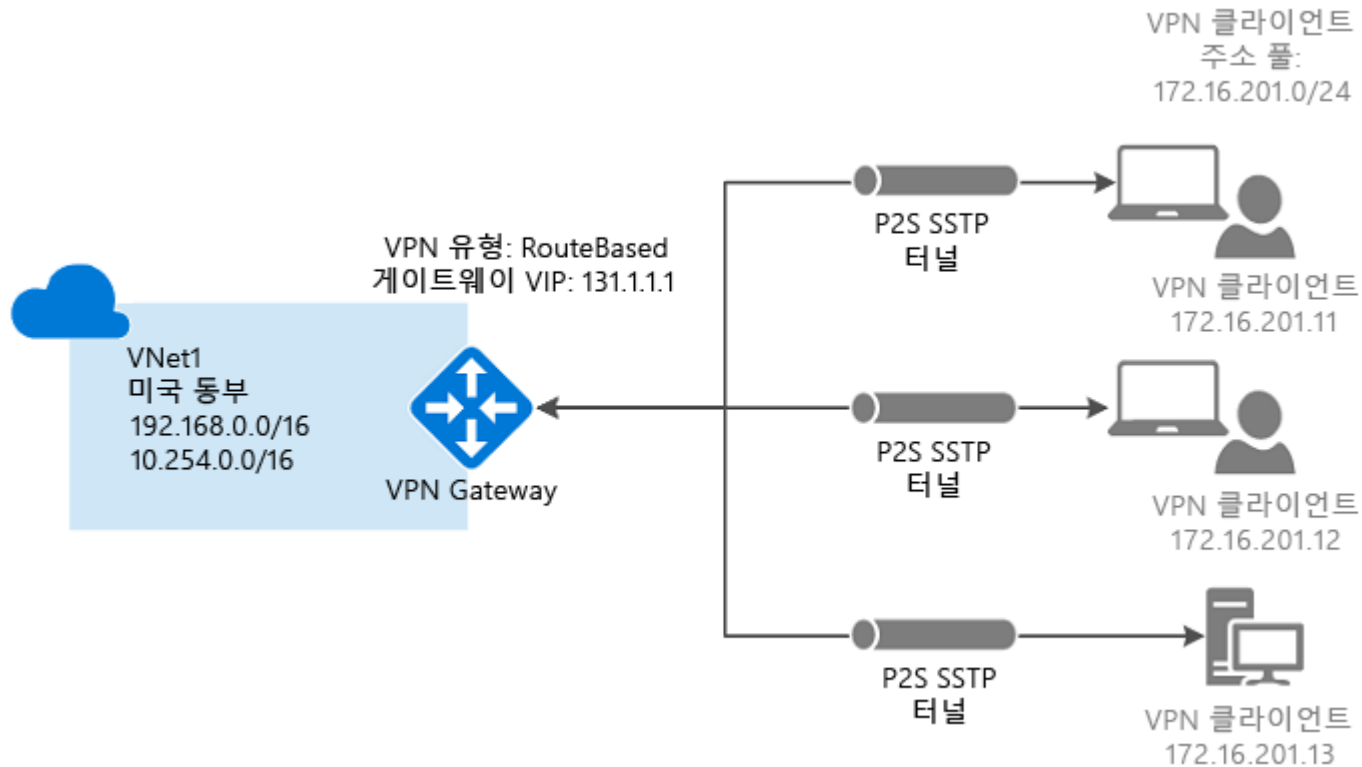
- Azure VNet과 연결하는 다양한 VPN 연결 방법
- Point-to-Site VPN 구성하기
- VNet-to-VNet VPN 구성하기

Virtual Network 개요

- Azure VNet과 연결하는 다양한 VPN 연결 방법
 - **A Point-to-Site VPN**
 - 한 대의 컴퓨터가 직접 VNet에 연결하는 것이다
 - 이렇게 하면 Azure에 있는 VM을 안전하게 관리할 수 있다
 - **A Site-to-Site VPN**
 - 사내 네트워크가 Azure의 VNet과 연결하는 것이다
 - 각 사내 컴퓨터에 Azure VM에 연결하기 위한 작업을 할 필요가 없다
 - **ExpressRoute**
 - 전용망을 통하여 Azure VNet에 연결하는 것이다
 - ExpressRoute를 사용하면 보안 및 신뢰성을 높이고, 높은 대역폭 보장한다
 - **VNet-to-VNet**
 - Azure 내의 두 개의 VNet을 연결하는 것이다
 - **Virtual gateway**
 - VPN을 VNet에 연결할 때마다 VNet에 Virtual Gateway가 필요하다

4-VPN 구현하기

- Point-to-Site(P2S) VPN 구성하기
 - Azure를 관리하는 관리자 컴퓨터에 VPN Client를 구성하여 Azure VNet에 VPN 연결하여 VNet에 있는 VM들을 손쉽게 관리하기



4-VPN 구현하기

- Point-to-Site(P2S) VPN 구성하기

- 첫 번째 Virtual Network 생성하기

- **Virtual networks -> +Add**

- Name: **VNet1**
 - Address space: **192.168.0.0/16**
 - Subnet name: **FrontEnd**
 - Subnet address range: **192.168.1.0/24**
 - Subnet name: **BackEnd**
 - Subnet address range: **192.168.2.0/24**
 - Resource Group: **TestRG**
 - Location: **East US**

- 참고:<Virtual Network Gateway에서 할 작업>
 - Connection type: **Point-to-site**
 - Client address pool: **172.16.201.0/24**

Create virtual network

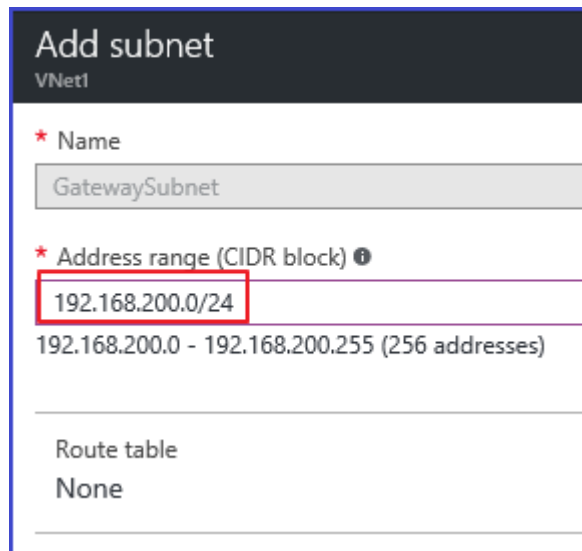
- * Name: VNet1 ✓
- * Address space ⓘ: 192.168.0.0/16 ✓
192.168.0.0 - 192.168.255.255 (65536 addresses)
- * Subnet name: FrontEnd ✓
- * Subnet address range ⓘ: 192.168.1.0/24 ✓
192.168.1.0 - 192.168.1.255 (256 addresses)
- * Subscription: MSDN Platforms ▼
- * Resource group ⓘ: ☒ Create new ☐ Use existing
TestRG ✓
- * Location: East US ▼

4-VPN 구현하기

- Point-to-Site(P2S) VPN 구성하기

- Gateway Subnet 추가하기

- VNet1을 Gateway에 연결하기 전에, 먼저 연결하려는 VNet1에 대한 Gateway Subnet을 만들어야 한다
 - **Virtual networks** -> **VNet1** 선택 -> **Subnets** -> **+Gateway subnet** 클릭
 - Address range: **192.168.200.0/24**



Add subnet
VNet1

* Name
GatewaySubnet

* Address range (CIDR block) ⓘ
192.168.200.0/24
192.168.200.0 - 192.168.200.255 (256 addresses)

Route table
None

4-VPN 구현하기

- Point-to-Site(P2S) VPN 구성하기
 - Virtual network Gateway 생성하기
 - **Virtual network Gateway -> +Add**
 - Virtual network gateway name: **VNet1GW**
 - Gateway type: **VPN**
 - VPN type: **Route-based**
 - Public IP address: **VNet1GWpip**
 - ****여기서 시간이 많이 소요된다(약20분)**

Create virtual network gate... [X]

* Name
VNet1GW ✓

Gateway type ⓘ
VPN ExpressRoute

VPN type ⓘ
Route-based Policy-based

* SKU ⓘ
Standard ▼

* Virtual network ⓘ
VNet1 >

* Public IP address ⓘ
(new) VNet1GWpip >

* Subscription
MSDN Platforms ▼

Resource group ⓘ
TestRG

* Location ⓘ
East US ▼

4-VPN 구현하기

- Point-to-Site(P2S) VPN 구성하기

- Certificate 생성한 후에 내보내기

- Self-signed Root certificate용 .cer 파일 생성하기

- PowerShell 콘솔을 관리자 권한으로 실행하여 다음과 같이 입력한다

- **New-SelfSignedCertificate** -Type Custom -KeySpec Signature -Subject "CN=**P2SRootCert**" -KeyExportPolicy Exportable -HashAlgorithm sha256 -KeyLength 2048 -CertStoreLocation "Cert:\CurrentUser\My" -KeyUsageProperty Sign -KeyUsage CertSign

```
PSParentPath: Microsoft.PowerShell.Security\Certificate::CurrentUser\My
```

```
Thumbprint
```

```
Subject
```

```
-----  
6B86D6A8D8C8F665E58F857540BBC887E3009460
```

```
-----  
CN=P2SRootCert
```

4-VPN 구현하기

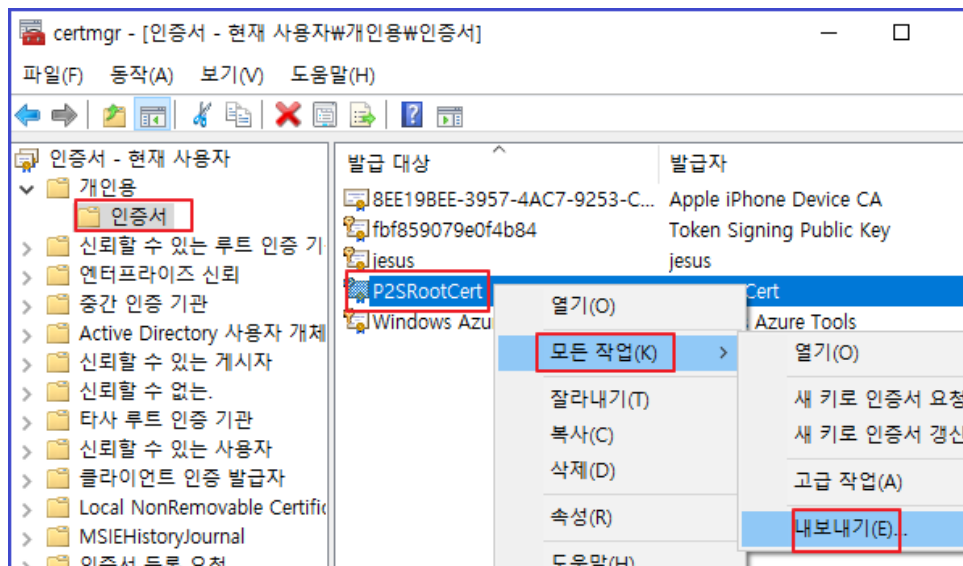
• Point-to-Site(P2S) VPN 구성하기

• Certificate 생성한 후에 내보내기

• Self-signed Root certificate용 .cer 파일 생성하기

- Azure로 업로드할 Public key(.cer)가 필요하기 때문에 Self-signed Public key를 파일로 내보내기 한다

• WIN +R -> certmgr.msc



인증서와 함께 개인 키를 내보내시겠습니까?

☐ 예, 개인 키를 내보냅니다(Y).

☒ 아니요, 개인 키를 내보내지 않습니다(O)

사용할 형식을 선택하십시오.

☐ DER로 인코딩된 바이너리 X.509(.CER)(D)

☒ Base 64로 인코딩된 X.509(.CER)(S)

☐ 암호화 메시지 구문 표준 - PKCS #7 인증서(.P7B)(C)

내보낼 파일

내보낼 파일 이름을 지정하십시오.

파일 이름(F):

c:\down\WP2SRootCert.cer

4-VPN 구현하기

- Point-to-Site(P2S) VPN 구성하기

- Client Certificate 생성한 후에 보내기

- P2S로 VNet에 연결하는 컴퓨터는 반드시 client certificate가 설치되어 있어야 한다
 - Self-signed root certificate(.cer)을 사용하여 client certificate를 생성한다
 - 위의 명령어 결과에서 Thumbprint를 복사하여 아래의 Get-ChildItem에 사용한다
 - **New-SelfSignedCertificate** -Type Custom -KeySpec Signature -Subject "CN=**P2SChildCert**" -KeyExportPolicy Exportable -HashAlgorithm sha256 -KeyLength 2048 -CertStoreLocation "Cert:\CurrentUser\My" -Signer (**Get-ChildItem -Path "Cert:\CurrentUser\My\6B86D6A8D8C8F665E58F857540BBC887E3009460"**) -TextExtension @("2.5.29.37={text}1.3.6.1.5.5.7.3.2")

```
PSParentPath: Microsoft.PowerShell.Security\Certificate::CurrentUser\My
Thumbprint                               Subject
-----
D9CE0170F8AA5FF11D070A37FD89770B1FA17544 CN=P2SChildCert
```

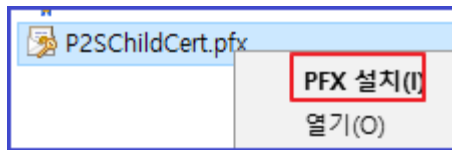
C:\down\P2SChildCert.pfx

4-VPN 구현하기

- Point-to-Site(P2S) VPN 구성하기

- 내보내기한 Client Certificate 설치하기

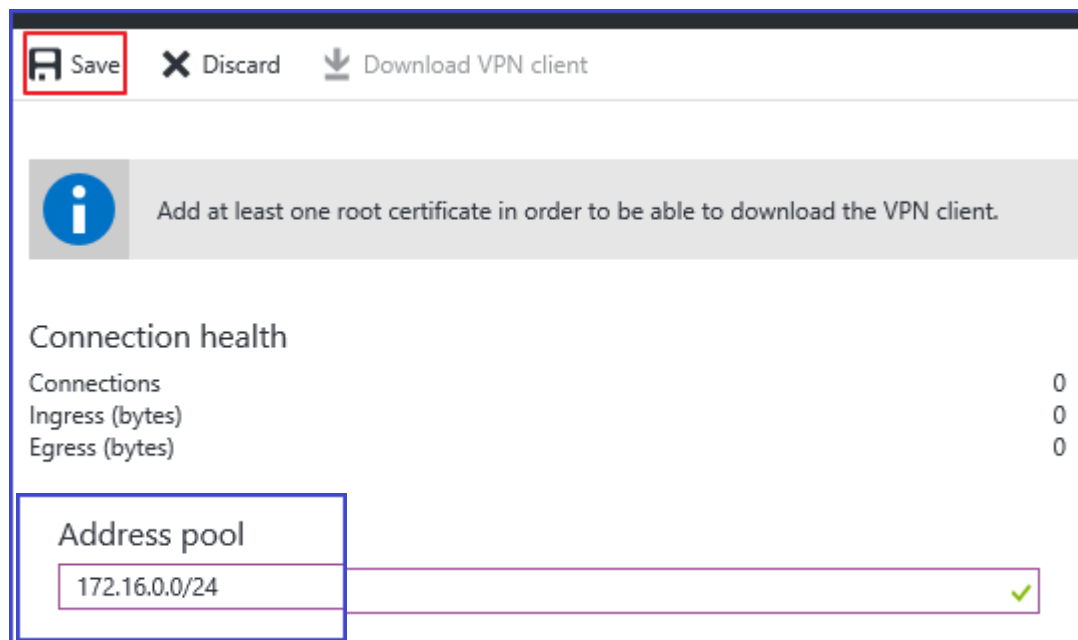
- Client computer에서 P2S 연결을 생성하려면 client certificate를 설치해야 한다
 - C:\down\P2SChildCert.pfx 파일을 실행하여 설치한다



- 설치하는 과정에서 암호만 입력하고 Default 설정대로 Next 만 클릭하여 설치한다

4-VPN 구현하기

- Point-to-Site(P2S) VPN 구성하기
 - Client Address Pool 추가하기
 - **Virtual network Gateways -> VNet1GW 선택 -> Point-to-Site configuration** 클릭



4-VPN 구현하기

- Point-to-Site(P2S) VPN 구성하기

- Root certificate (.cer)을 Azure로 Upload하기

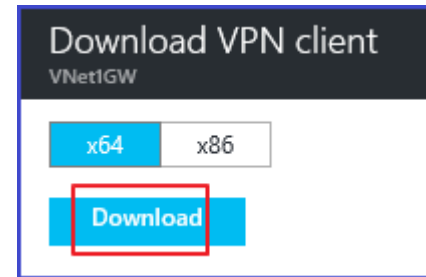
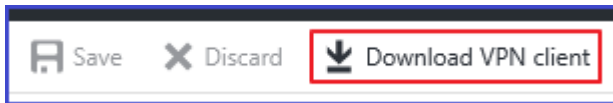
- Gateway를 생성한 후에는 trusted root certificate용 파일인 .cer 파일을 Azure로 업로드한다
 - 최대 20개까지 가능
 - Private key는 업로드하지 않는다
 - .cer 파일이 Upload 되면, Azure는 VNet에 연결하는 Client를 인증하기 위해 .cer 파일을 사용하게 된다
 - 메모장으로 c:\down\P2SRootCert.cer 파일을 열어서 본문 만 복사하여 **Virtual network Gateways -> VNet1GW 선택 -> Point-to-Site configuration** 클릭하여 붙여 넣기 한다
 - 반드시 본문 내용을 한 줄로 만들어서 붙여 넣기 해야 한다

Root certificates

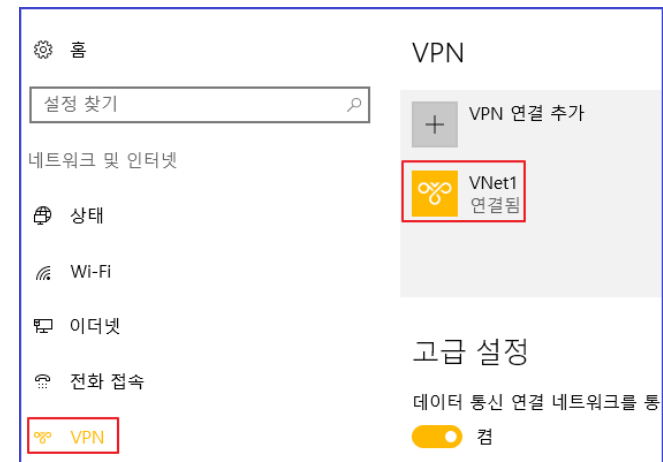
NAME	PUBLIC CERTIFICATE DATA
P2SRootCert	MIIC5zCCAc+gAwIBAgIQV53IIRZVxaJINBdiFKqGGDANBgkqhkiG9w0BAQsFADAWMRQwEgYDVQQDDAtQMINSb290Q2VydDAeFw0xNzA0MDgxNjA4MjVaFw0xODA0MDgxNjA4MjVaMBYxf...

4-VPN 구현하기

- Point-to-Site(P2S) VPN 구성하기
 - VPN client configuration package를 다운로드하여 설치하기
 - **Virtual network Gateways -> VNet1GW 선택 -> Point-to-Site configuration -> Download VpN client 클릭**



- 다운로드한 파일을 실행하여 설치한다
- VPN 연결을 시도한다
- ipconfig를 통하여 IP 정보 확인





```
PPP 어댑터 VNet1:
연결별 DNS 접미사. . . . . :
IPv4 주소. . . . . : 172.16.0.2
서브넷 마스크. . . . . : 255.255.255.255
기본 게이트웨이. . . . . :
```

4-VPN 구현하기

- Point-to-Site(P2S) VPN 구성하기

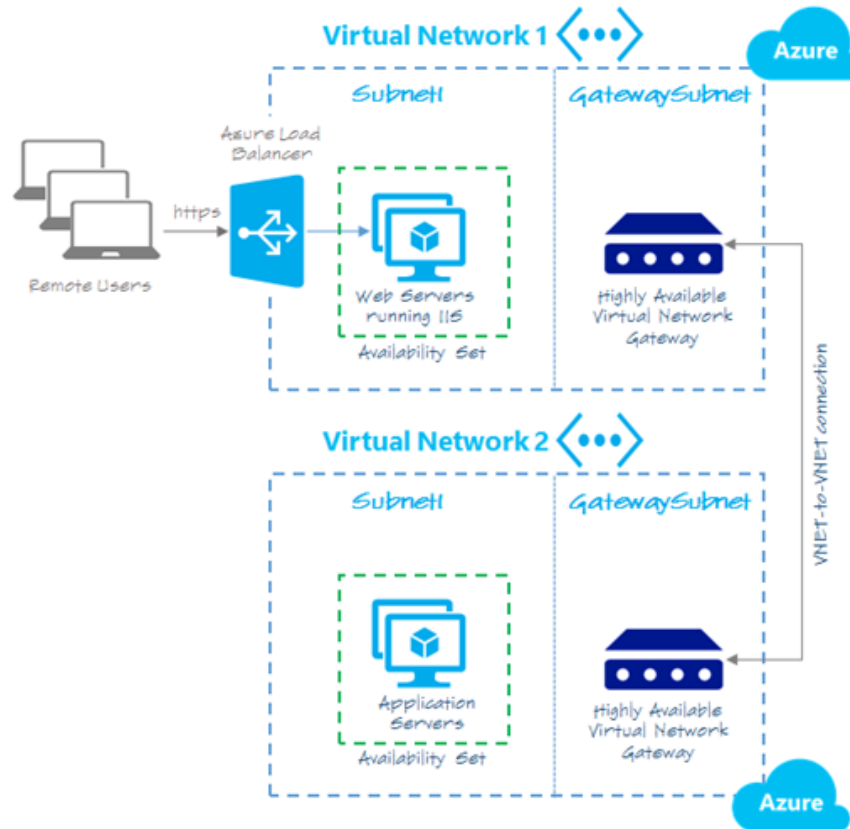
- VNet1을 사용하여 VM을 생성하여 Client 컴퓨터랑 내부 통신이 되는지 확인하기
 - Windows 2016 VM을 생성하되, TestRG Resource Group을 이용하고, East US 지역에 managed disk로 하고, VNet1에 소속되도록 한다
 - VM이 설치되었다면 **mstsc.exe /v:192.168.1.4**를 실행하여 접속을 해본다

Virtual machines					
기본 디렉터리 (jesuswithmehotmail.onmicrosoft.com)					
+ Add Columns Refresh					
Subscriptions: MSDN Platforms – Don't see a subscription? Switch directories					
Filter by name...					
2 items					
NAME	STATUS	RESOURCE GROUP	LOCATION	PUBLIC DNS NAME	PRIVATE IP ADDRESS
 linux	Running	korRG	Korea Central	-	10.0.0.4
 win2016	Running	TestRG	East US	-	192.168.1.4

- 접속 성공!!

4-VPN 구현하기

- VNet-to-VNet VPN 구성하기
 - **VNet들이 서로 다른 지역에 있을 때** 해당 VNet에 소속 VM들간에 내부적으로 통신하기 위해서는 VNet-to-VNet VPN을 구성하면 된다

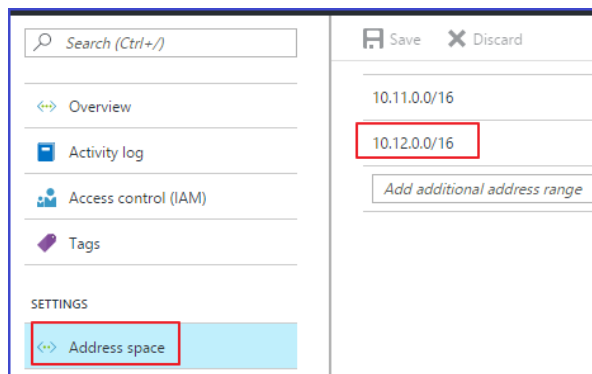


4-VPN 구현하기

- VNet-to-VNet VPN 구성하기
 - 첫 번째 Virtual Network 생성하기
 - **Virtual networks -> +Add**
 - Name: **TestVNet1**
 - Address space: **10.11.0.0/16**
 - Subnet name: **FrontEnd**
 - Subnet address range: **10.11.0.0/24**
 - Resource Group: **TestRG1**
 - Location: **East US**
 - Address space: **10.12.0.0/16**
 - Subnet name: **BackEnd**
 - Subnet address range: **10.12.0.0/24**

Create virtual network

- * Name: TestVNet1 ✓
- * Address space ①: 10.11.0.0/16 ✓
10.11.0.0 - 10.11.255.255 (65536 addresses)
- * Subnet name: FrontEnd ✓
- * Subnet address range ①: 10.11.0.0/24 ✓
10.11.0.0 - 10.11.0.255 (256 addresses)
- * Subscription: Azure Pass
- * Resource group ①: ☒ Create new ☐ Use existing
TestRG1 ✓
- * Location: East US



NAME	ADDRESS RANGE
FrontEnd	10.11.0.0/24
BackEnd	10.12.0.0/24

4-VPN 구현하기

- VNet-to-VNet VPN 구성하기
 - 첫 번째 Virtual Network 생성하기
 - **Virtual networks** -> **+Add**
 - Name: **TestVNet4**
 - Address space: **10.41.0.0/16**
 - Subnet name: **FrontEnd**
 - Subnet address range: **10.41.0.0/24**
 - Resource Group: **TestRG1**
 - Location: **West US**
 - Address space: **10.42.0.0/16**
 - Subnet name: **BackEnd**
 - Subnet address range: **10.42.0.0/24**

Create virtual network

* Name
TestVNet4 ✓

* Address space ⓘ
10.41.0.0/16 ✓
10.41.0.0 - 10.41.255.255 (65536 addresses)

* Subnet name
FrontEnd ✓

* Subnet address range ⓘ
10.41.0.0/24 ✓
10.41.0.0 - 10.41.0.255 (256 addresses)

* Subscription
Azure Pass ▼

* Resource group ⓘ
☐ Create new ☒ Use existing
TestRG1 ▼

* Location
West US ▼

Save Discard

10.41.0.0/16

10.42.0.0/16

Add additional address range

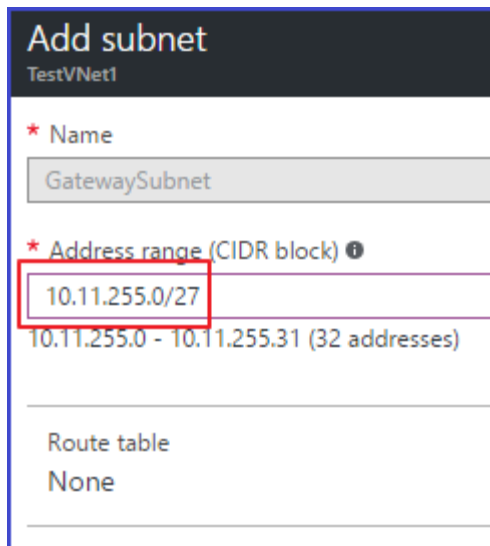
+ Subnet + Gateway subnet

Search subnets

NAME	ADDRESS RANGE	AVAILABLE ADDRESSES
FrontEnd	10.41.0.0/24	251
BackEnd	10.42.0.0/24	251

4-VPN 구현하기

- VNet-to-VNet VPN 구성하기
 - Gateway Subnet 추가하기
 - TestVNet1을 Gateway에 연결하기 전에, 먼저 연결하려는 TestVNet1에 대한 Gateway Subnet을 만들어야 한다
 - **Virtual networks** -> **TestVNet1** 선택 -> **Subnets** -> **+Gateway subnet** 클릭
 - Gateway Subnet name: **GatewaySubnet**
 - Gateway Subnet address range: **10.11.255.0/27**



Add subnet
TestVNet1

* Name
GatewaySubnet

* Address range (CIDR block) ⓘ
10.11.255.0/27
10.11.255.0 - 10.11.255.31 (32 addresses)

Route table
None

4-VPN 구현하기

- VNet-to-VNet VPN 구성하기
 - Gateway Subnet 추가하기
 - TestVNet4를 Gateway에 연결하기 전에, 먼저 연결하려는 TestVNet4에 대한 Gateway Subnet을 만들어야 한다
 - **Virtual networks** -> **TestVNet4** 선택 -> **Subnets** -> **+Gateway subnet** 클릭
 - Gateway Subnet name: **GatewaySubnet**
 - Gateway Subnet address range: **10.41.255.0/27**

Add subnet
TestVNet4

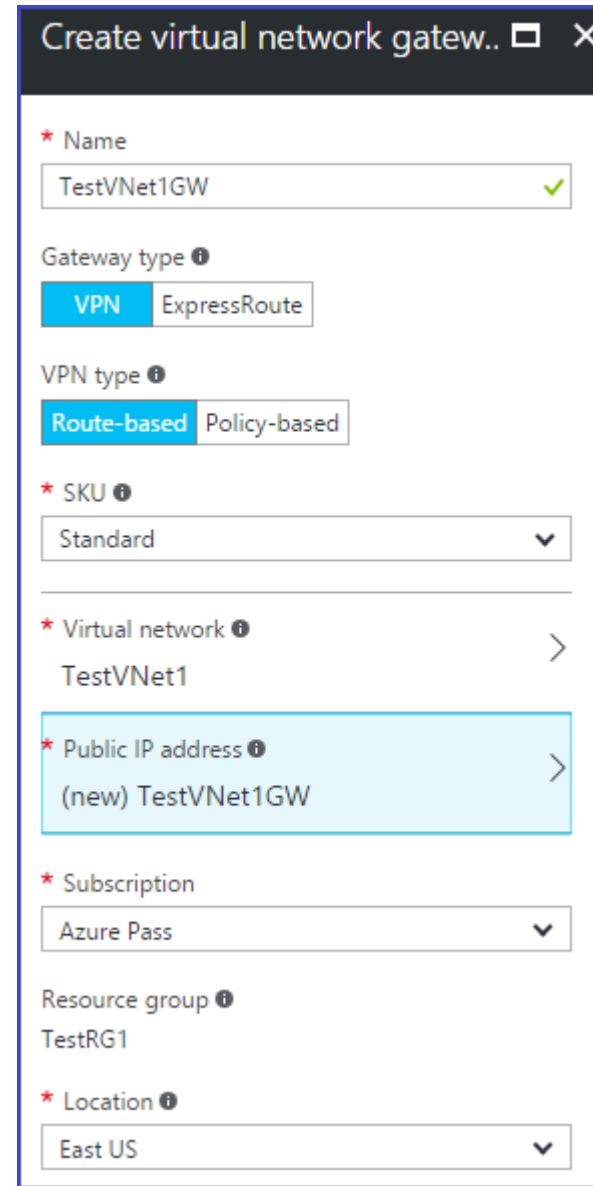
* Name
GatewaySubnet

* Address range (CIDR block) ⓘ
10.41.255.0/27
10.41.255.0 - 10.41.255.31 (32 addresses)

Route table
None

4-VPN 구현하기

- VNet-to-VNet VPN 구성하기
 - Virtual network Gateway 생성하기
 - **Virtual network Gateway -> +Add**
 - Virtual network gateway name: **TestVNet1GW**
 - Gateway type: **VPN**
 - VPN type: **Route-based**
 - Virtual network: **TestVNet1**
 - Public IP address: **TestVNet1GW**
 - Location: **East US**
- TestVNet1GW가 생성되는데 약 20분 소요된다

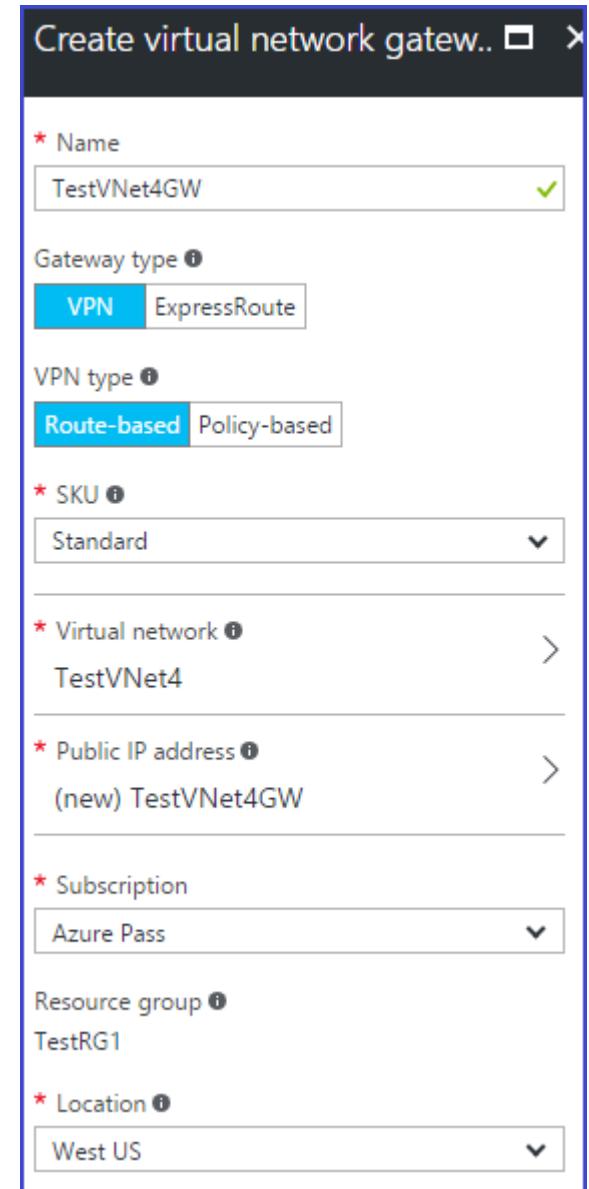


The screenshot shows the 'Create virtual network gateway' form in the Azure portal. The form is titled 'Create virtual network gateway..' and contains the following fields and options:

- Name:** TestVNet1GW (with a green checkmark icon)
- Gateway type:** VPN (selected) and ExpressRoute
- VPN type:** Route-based (selected) and Policy-based
- SKU:** Standard (with a dropdown arrow)
- Virtual network:** TestVNet1 (with a right arrow icon)
- Public IP address:** (new) TestVNet1GW (with a right arrow icon)
- Subscription:** Azure Pass (with a dropdown arrow)
- Resource group:** TestRG1
- Location:** East US (with a dropdown arrow)

4-VPN 구현하기

- VNet-to-VNet VPN 구성하기
 - Virtual network Gateway 생성하기
 - **Virtual network Gateway -> +Add**
 - Virtual network gateway name: **TestVNet4GW**
 - Gateway type: **VPN**
 - VPN type: **Route-based**
 - Virtual network: **TestVNet4**
 - Public IP address: **TestVNet4GW**
 - Location: **West US**
- TestVNet1GW가 생성되는데 약 20분 소요된다



The screenshot shows the 'Create virtual network gateway' form in the Azure portal. The form is titled 'Create virtual network gateway..' and contains the following fields and options:

- Name:** TestVNet4GW (with a green checkmark icon)
- Gateway type:** VPN (selected) and ExpressRoute
- VPN type:** Route-based (selected) and Policy-based
- SKU:** Standard (with a dropdown arrow)
- Virtual network:** TestVNet4 (with a right arrow icon)
- Public IP address:** (new) TestVNet4GW (with a right arrow icon)
- Subscription:** Azure Pass (with a dropdown arrow)
- Resource group:** TestRG1
- Location:** West US (with a dropdown arrow)

4-VPN 구현하기

- VNet-to-VNet VPN 구성하기
 - 지역적으로 떨어진 VNet에 속한 VM을 하나씩 생성하여 VPN 통신 여부 확인하기
 - TestVNet1에 **VNet1VM** 생성한다
 - Location: **East US**
 - TestVNet4에 **VNet4VM** 생성한다
 - Location: **West US**
 - 각 VM에 ICMP에 대하여 응답하도록 각각 Firewall 설정을 한다
 - VNet4VM(10.41.0.4)에서 VNet1VM(10.11.0.4)으로 **tracert**를 한다
 - 결과는 **실패!!**

4-VPN 구현하기

- VNet-to-VNet VPN 구성하기
 - **TestVNet1** connection 구성하기
 - **Virtual network Gateway -> TestVNet1GW -> Connections -> +Add**
 - Name: **TestVNet1toTestVNet4**
 - Connection type: **VNet-to-VNet**
 - Second virtual network gateway:
TestVNet4GW
 - 만약 TestVNet4GW를 선택할 수 없으면 좀 더 기다린다
 - Shared key (PSK): **myazure9191**

The screenshot shows the 'Add connection' dialog for 'TestVNet1GW'. The fields are as follows:

- Name:** TestVNet1toTestVNet4 (checked)
- Connection type:** VNet-to-VNet (dropdown)
- First virtual network gateway:** TestVNet1GW (locked)
- Second virtual network gateway:** TestVNet4GW (dropdown)
- Shared key (PSK):** myazure9191 (checked)
- Subscription:** Azure Pass (dropdown)
- Resource group:** TestRG1 (locked)
- Location:** East US (dropdown)

4-VPN 구현하기

- VNet-to-VNet VPN 구성하기
 - **TestVNet4** connection 구성하기
 - **Virtual network Gateway -> TestVNet4GW -> Connections -> +Add**
 - Name: **TestVNet4toTestVNet1**
 - Connection type: **VNet-to-VNet**
 - Second virtual network gateway:
TestVNet1GW
 - 만약 TestVNet1GW를 선택할 수 없으면 좀 더 기다린다
 - Shared key (PSK): **myazure9191**

Add connection
TestVNet4GW

* Name
TestVNet4toTestVNet1 ✓

Connection type ⓘ
VNet-to-VNet ▼

* First virtual network gateway ⓘ
TestVNet4GW 🔒

* Second virtual network gateway ⓘ
TestVNet1GW >

* Shared key (PSK) ⓘ
myazure9191 ✓

Subscription ⓘ
Azure Pass ▼

Resource group ⓘ
TestRG1 🔒
Create new

Location ⓘ
West US ▼

4-VPN 구현하기

- VNet-to-VNet VPN 구성하기
 - 지역적으로 떨어진 VNet간의 연결성 확인하기

NAME	STATUS	CONNECTION TYPE	PEER
TestVNet1toTestVNet4	Connected	VNet-to-VNet	TestVNet1GW
TestVNet4toTestVNet1	Connected	VNet-to-VNet	TestVNet1GW

- 원하는 항목을 더블 클릭하여 자세하게 관찰하기

The screenshot shows the Azure portal interface for a VNet-to-VNet connection. The connection name 'TestVNet4toTestVNet1' is highlighted in the top header. The left sidebar contains navigation options: Overview (selected), Activity log, Access control (IAM), Tags, and SETTINGS. The main content area displays the connection's status as 'Connected' and its location as 'West US'. The 'Essentials' section shows the resource group 'TestRG1' and the subscription name 'Azure Pass'. The 'Virtual network' section lists the connected VNet as 'TestVNet1, TestVNet4' and the virtual network gateways as 'TestVNet4GW (52.160.105.56)' and 'TestVNet1GW (52.168.80.172)'. The 'Data in' and 'Data out' sections show traffic volumes of 2.5 KiB and 2.38 KiB respectively.

Resource group (change)	Data in
TestRG1	2.5 KiB
Status	Data out
Connected	2.38 KiB
Location	Virtual network
West US	TestVNet1, TestVNet4
Subscription name (change)	Virtual network gateway 1
Azure Pass	TestVNet4GW (52.160.105.56)
Subscription ID	Virtual network gateway 2
8f1de46e-aebc-4a24-96ec-8831f371f9a6	TestVNet1GW (52.168.80.172)

4-VPN 구현하기

- VNet-to-VNet VPN 구성하기
 - 지역적으로 떨어진 VNet에 속한 VM에서 다시 VPN 통신 여부 확인하기
 - VNet4VM(10.41.0.4)에서 VNet1VM(10.11.0.4)으로 **tracert**를 한다
 - 결과는 **성공!!**

```
PS C:\Users\adminuser> tracert 10.11.0.4
Tracing route to 10.11.0.4 over a maximum of 30 hops
  1    <1 ms    <1 ms    <1 ms    10.41.255.5
  2     *        *        *        Request timed out.
  3    75 ms    75 ms    75 ms    10.11.0.4
```


4-VPN 구현하기

- VNet-to-VNet VPN 구성하기

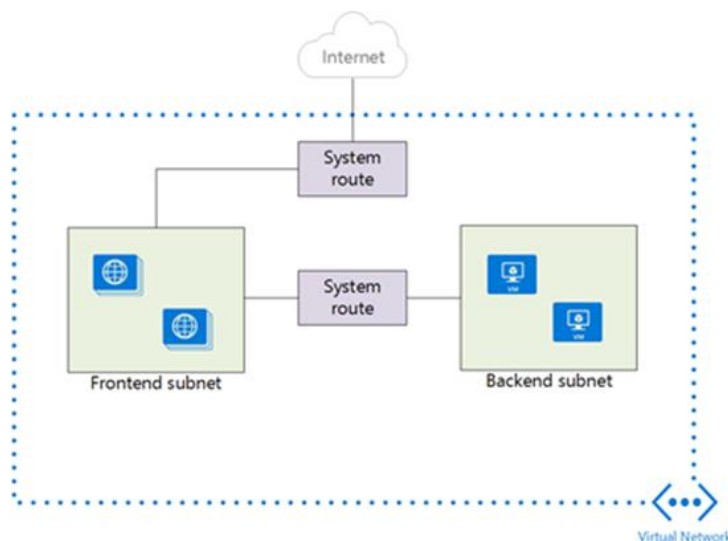
- VNet 간 FAQ

- Azure는 VNet 간 트래픽에 요금을 청구하나요?
 - 동일한 지역 내의 VNet 간 트래픽은 양방향 모두에 대해 무료입니다.
 - 지역 전체 VNet 간 송신 트래픽은 원본 지역을 기반으로 아웃바운드 VNet 간 데이터 전송 요금으로 청구됩니다
 - VNet 간 트래픽은 인터넷을 거쳐서 이동하나요?
 - 아니요. VNet 간 트래픽은 인터넷이 아닌 Microsoft Azure 백본을 거쳐서 이동합니다
 - VNet 간 트래픽은 안전한가요?
 - 예, IPsec/IKE 암호화로 보호됩니다
 - VNet이 동일한 지역에 있어야 하나요?
 - 아니요. 가상 네트워크는 같은 Azure 지역(위치)에 있을 수도 있고 다른 Azure 지역(위치)에 있을 수도 있습니다
 - VNet간 연결을 멀티 사이트 연결과 함께 사용할 수 있나요?
 - 예. 가상 네트워크 연결을 다중 사이트 VPN과 동시에 사용할 수 있습니다

5-User Defined Route 구성하기

- User Defined Route

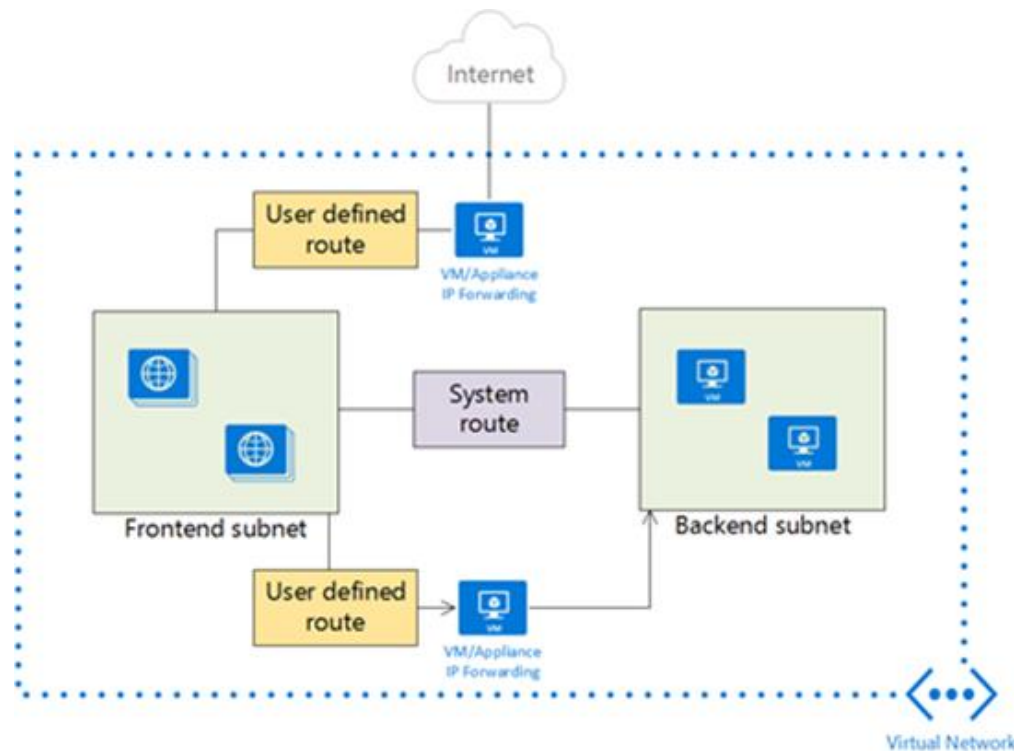
- 동일한 Virtual Network에 있는 서로 다른 Subnet에 존재하는 VM들의 통신을 할 때 반드시 Virtual Appliance를 통하여 통신하도록 사용자가 경로를 수동으로 설정하는 것이다
- Default system rule with a multi-tier web application in Azure
 - Frontend Subnet에는 Web Server가 있고, Backend Subnet에는 Database Server가 있다
 - 기본적으로 인터넷 접속을 Backend Subnet으로 못하게 하는 것은 Network Security Group을 사용하여 기본적인 Port로 차단한다



5-User Defined Route 구성하기

- User Defined Route

- User defined route는 강제로 Azure Virtual Appliance를 통하여 트래픽을 보내도록 하는 것
 - Virtual Appliance는 Windows 2012 R2로 RRAS를 설치하여 구성한 것
 - FrontendSubnet의 VM0이 BackendSubnet의 VM2으로 통신을 할 때는 반드시 Virtual Appliance를 거쳐서 가도록 설정한다



5-User Defined Route 구성하기

- User Defined Route

- 다음과 같이 구성하기 위해서 ARM Template를 사용한다

- ARM Template:

- <https://github.com/Azure/azure-quickstart-templates/tree/master/201-userdefined-routes-appliance>

TEMPLATE

201-userdefined-routes-appliance
8 resources

Edit Learn more

BASICS

* Subscription Azure Pass

* Resource group ① ☒ Create new ☐ Use existing UDR ✓

* Location East Asia

SETTINGS

* Admin Username ① adminuser ✓

* Admin Password ① ✓

* Unique Dns Prefix For VM ① udr9191 ✓

Vm Name Prefix ① Dynamic

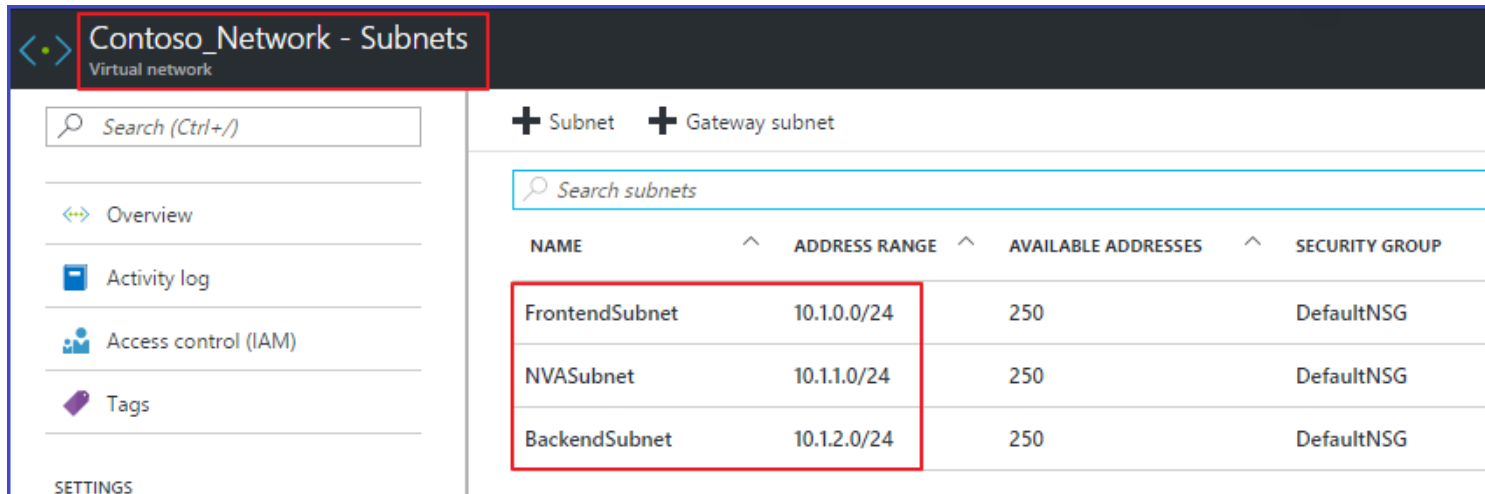
Public IP Address Type ① Dynamic

Windows OS Version ① 2012-R2-Datacenter

- Resource Group: UDR
- VM 3개: Dynamic0, Dynamic1(NIC에서 IP Forwarding이 설정됨), Dynamic2
- Subnet 3개: FrontendSubnet, NVASubnet, BackendSubnet
- User defined route: BasicNVA
- Virtual Network: Contoso_Network

5-User Defined Route 구성하기

- User Defined Route
 - 3개의 Subnet 생성



Contoso_Network - Subnets
Virtual network

Search (Ctrl+ /)

Overview

Activity log

Access control (IAM)

Tags

SETTINGS

+ Subnet + Gateway subnet

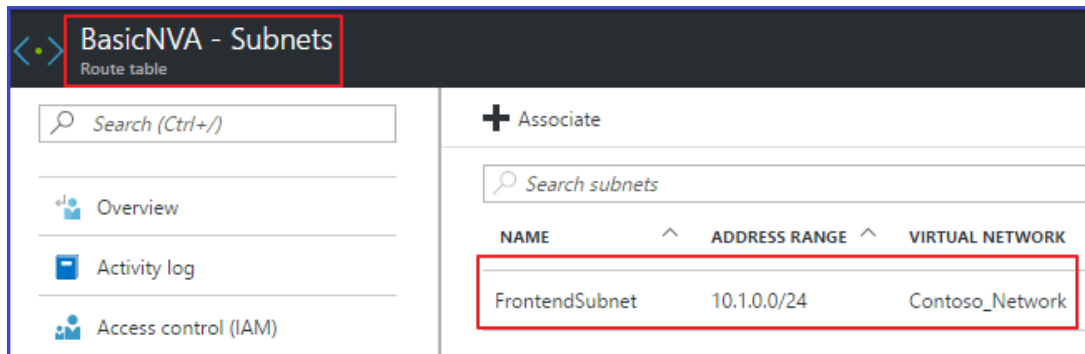
Search subnets

NAME	ADDRESS RANGE	AVAILABLE ADDRESSES	SECURITY GROUP
FrontendSubnet	10.1.0.0/24	250	DefaultNSG
NVASubnet	10.1.1.0/24	250	DefaultNSG
BackendSubnet	10.1.2.0/24	250	DefaultNSG

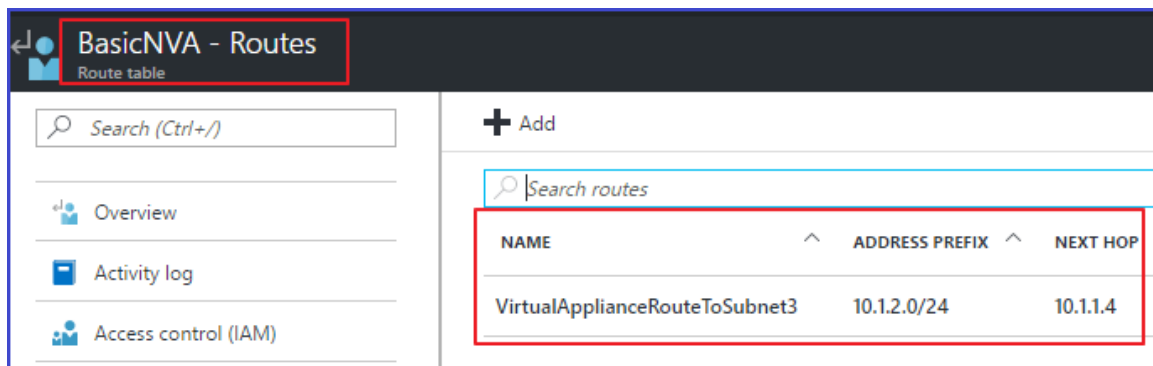
5-User Defined Route 구성하기

- User Defined Route

- BasicNVA라는 Route table을 생성하여 FrontendSubnet이라는 Subnet에 적용함



- BasicNVA의 설정 내용은 10.1.2.0/24으로 가려면 10.1.1.4(RRAS 운영 중)라는 Virtual Appliance를 통하여 가도록 함



5-User Defined Route 구성하기

- User Defined Route

- Traffic 흐름 검사하기

- **Dynamic1**에서 services.msc를 실행하여 Routing and Remoting Access를 Automatic으로 변경한 후 Start한다
 - Dynamic1 컴퓨터의 NIC 설정에서 IP Forwarding 기능이 Enabled됨
 - 다른 컴퓨터가 Dynamic1으로 패킷을 보내면 그것을 다른 네트워크로 전달하는 기능
 - **Dynamic1과 Dynamic2**에서 Ping을 허용하도록 설정한다
 - wf.msc를 실행하여 **File and Printer Sharing (Echo Request - ICMPv4-In)** 을 Enable시킨다
 - **Dynamic0**에서 **tracert dynamic2** 를 하여 트래픽이 Virtual Appliance 역할을 하는 Dynamic1을 통해서 가는지 확인한다

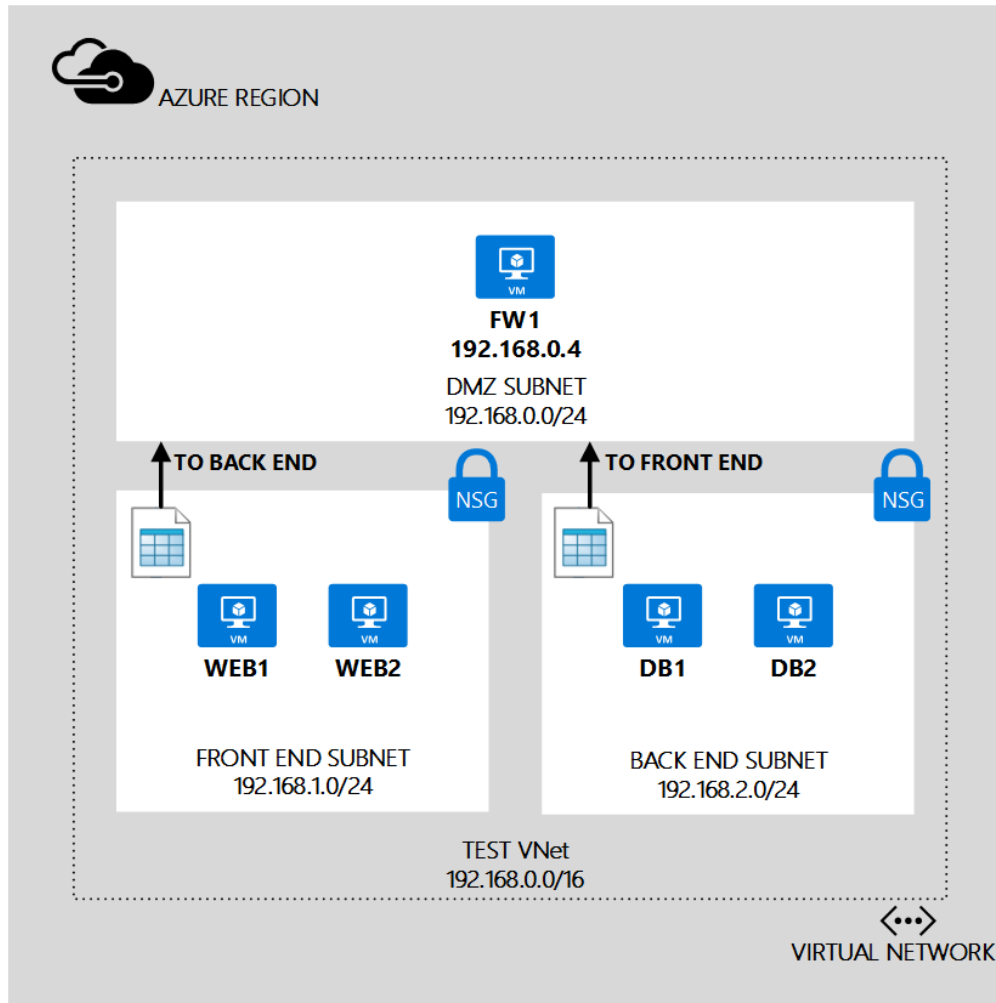
```
PS C:\> hostname
Dynamic0
PS C:\> tracert dynamic2

Tracing route to dynamic2.2qr55jxjivgelkohdralo3mjff.hx.internal.cloudapp.net [10.1.2.4]
over a maximum of 30 hops:

  1     1 ms      *         <1 ms  10.1.1.4
  2     1 ms      <1 ms    <1 ms  10.1.2.4
```

5-User Defined Route 구성하기

- User Defined Route
 - Web Server와 DB Server간의 통신 시나리오



5-User Defined Route 구성하기

- User Defined Route

- Web Server와 DB Server간의 통신 시나리오

- TestVNet 가상 네트워크의 각 서브넷에 대한 **NSG**를 만든다

- **NSG-FrontEnd.** 프론트 엔드 NSG는 FrontEnd 서브넷에 적용되며 다음 두 개의 규칙을 포함한다

- **rdp-rule.** 이 규칙은 FrontEnd 서브넷에 대한 RDP 트래픽을 허용한다

- **web-rule.** 이 규칙은 FrontEnd 서브넷에 대한 HTTP 트래픽을 허용한다

- **NSG-BackEnd.** 백 엔드 NSG는 BackEnd 서브넷에 적용되며 다음 두 개의 규칙을 포함한다

- **sql-rule.** 이 규칙은 FrontEnd 서브넷의 SQL 트래픽만 허용한다

- **web-rule.** 이 규칙은 BackEnd 서브넷의 모든 인터넷 바인딩된 트래픽을 거부한다

- 일부 Virtual Appliance를 사용하여 다음과 같은 작업을 할 수 있다.

- 침입 감지 시스템(IDS)을 사용하여 트래픽 모니터링

- 방화벽을 사용하여 트래픽 제어

5-User Defined Route 구성하기

- User Defined Route

- Web Server와 DB Server간의 통신 시나리오

- Frontend Subnet 대해 하나의 UDR(UDR-FrontEnd)을 만들고 Backend Subnet에 대해 다른 UDR(UDR-BackEnd)을 만든다

- **UDR-FrontEnd:** 프론트 엔드 UDR은 FrontEnd 서브넷에 적용되며 다음 한 개의 경로를 포함한다

- **RouteToBackend.** 이 경로는 모든 트래픽을 백 엔드 서브넷으로, 다시 FW1 가상 컴퓨터로 보낸다

- **UDR-BackEnd:** 백 엔드 UDR은 BackEnd 서브넷에 적용되며 다음 한 개의 경로를 포함한다

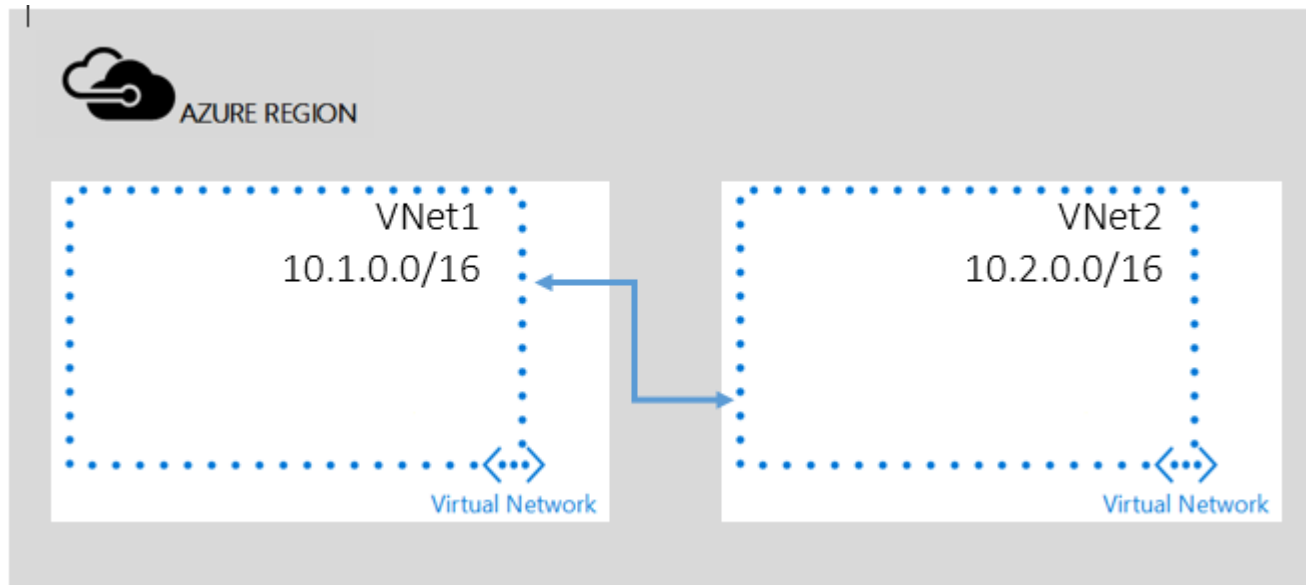
- **RouteToFrontend.** 이 경로는 모든 트래픽을 프론트 엔드 서브넷으로, 다시 FW1 가상 컴퓨터로 보낸다

- 이러한 경로의 조합은 한 Subnet에서 다른 Subnet으로 보내는 모든 트래픽이 Virtual Appliance로 사용되는 FW1 가상 컴퓨터로 경로 지정되도록 한다
- 또한 해당 VM에 대한 IP 전달을 설정하여 다른 VM으로 보내는 트래픽을 수신할 수 있도록 한다

6-Network Peering 구성하기

- Virtual Network Peering

- 동일한 지역에 존재하는 두 개의 VNet간의 내부 통신하기
 - 참고: <https://docs.microsoft.com/ko-kr/azure/virtual-network/virtual-networks-create-vnetpeering-arm-portal>
 - 기본적으로 다른 VNet에 속한 VM들간에는 내부 통신이 안되지만, 해당 VNet이 동일 Region에 있고, Peering 설정을 하면 VM간의 통신이 가능하다



6-Network Peering 구성하기

- Virtual Network Peering
 - Resource Group 생성하기
 - Name: **RG_Peering**
 - Location: **East Asia**
 - East Asia 지역에 Virtual Network를 2개 생성하기
 - Name: **VNet1**
 - Address space: **10.0.0.0/22**
 - Subnet name: **Subnet1**
 - Subnet address range: **10.0.0.0/24**
 - Resource Group: **RG_Peering**
 - Name: **VNet4**
 - Address space: **10.0.4.0/22**
 - Subnet name: **Subnet4**
 - Subnet address range: **10.0.4.0/24**
 - Resource Group: **RG_Peering**

6-Network Peering 구성하기

- Virtual Network Peering
 - VM을 2개 생성하되, 각각 다른 VNet에 할당하기
 - Windows Server 2016 Datacenter - Resource Manager
 - Name: **VM1**
 - Resource Group: **RG_Peering**
 - Virtual Network: **VNet1**
 - Windows Server 2016 Datacenter - Resource Manager
 - Name: **VM4**
 - Resource Group: **RG_Peering**
 - Virtual Network: **VNet4**
-

6-Network Peering 구성하기

- Virtual Network Peering
 - Virtual Network의 Peering 구성하기
 - VNet간의 Peering은 반드시 동일 지역에 있는 VNet간에만 적용된다
 - Virtual Networks - **VNet1** 선택 - **Peerings** 선택 - **+Add**
 - Name: **ToVNet4_Peerings**
 - Virtual Network: **VNet4**
 - Allow virtual network access: **Enabled**
- Virtual Networks - **VNet4** 선택 - **Peerings** 선택 - **+Add**
- Name: **ToVNet1_Peerings**
- Virtual Network: **VNet1**
- Allow virtual network access: **Enabled**

6-Network Peering 구성하기

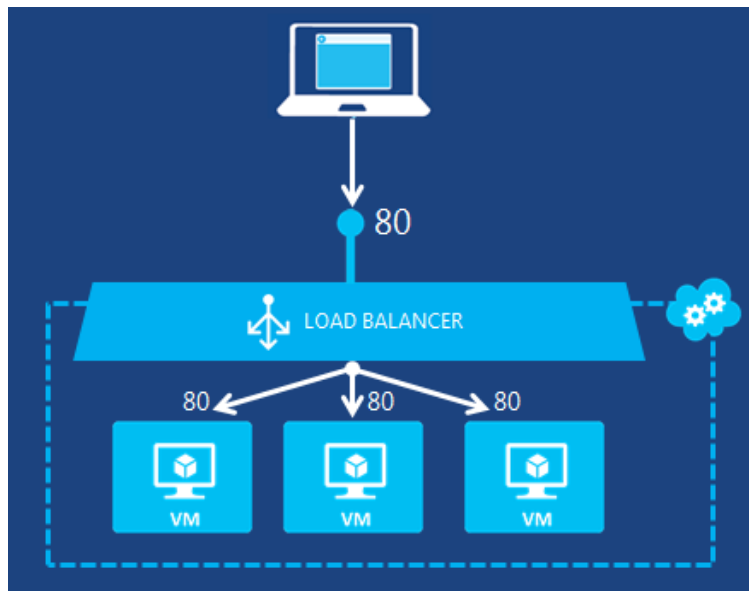
- Virtual Network Peering
 - VM1에서 VM2로 통신이 되는지 확인하기
 - VM1에 연결하여 로그인한다
 - VM1에서 VM4로 원격 데스크톱 연결을 해본다
 - **WIN + R**
 - **mstsc /v:10.0.4.4**

7-Load Balancer 구성하기

- Azure Load Balancer란?
- Azure Load Balancer의 Settings 이해하기
- Azure Portal에서 Load Balancer를 구성하기

7-Load Balancer 구성하기

- Azure Load Balancer란?
 - **Azure Load Balancer**는 동일한 Web Service를 하는 VM들에게 접속 트래픽을 자동 및 무작위 순서로 분산하여 처리하여서, 과중한 접속 부하로 서비스를 중단없이 제공하는 역할을 제공하는 기능이다
 - Azure Classic Portal과 Azure Portal에서 모두 Load Balancer를 사용할 수 있다
 - Azure Classic Portal에서는 Cloud Service에서 구성한다
 - Azure Portal에서는 별도의 Load Balancer를 붙여서 구성한다



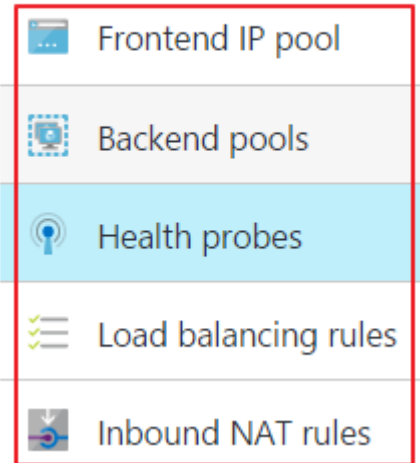
7-Load Balancer 구성하기

- Azure Load Balancer란?
 - **Azure Load Balancer**를 구성하기 전에 사전에 반드시 Public IP Address를 생성해야 한다
 - Public IP Address를 생성할 때 반드시 DNS Name을 설정해 두어야 한다
 - 이렇게 하면 Load Balancer에 접속할 때 FQDN으로 접속 가능하고, Traffic Manager를 통하여 Load Balancer에 접속할 때는 Public IP에 FQDN이 없으면 구성이 되지 않는다(매우 중요)

7-Load Balancer 구성하기

- Azure Load Balancer의 Settings 이해하기
 - **Frontend IP pool**
 - Load balancer의 공인 IP들로서, 사용자는 이 IP로 접속한다
 - **Backend pools**
 - Availability Set에 포함된 VM들의 IP 주소
 - 사용자가 Frontend IP로 접속할 때 사실은 Backend pool에 있는 VM에 접속하게 된다
 - **Health probes**
 - VM들이 정상적으로 운영되고 있는지 검사하기
 - **Load balancing rules**
 - 80 포트로 접속하면 내부의 80 포트로 넘겨주는 규칙 정하기
 - **Inbound NAT rules**
 - Load Balance 대신 내부에 있는 VM을 관리하기 위해서 NAT를 해주는 것
 - Frontend IP의 3389로 접속할 때 내부의 특정한 VM으로 연결해주기

SETTINGS



7-Load Balancer 구성하기

- Azure Portal에서 Load Balancer를 구성하기
 - Availability Set 생성
 - Managed Disk 사용

Create availability set

* Name
AVSET-WEB ✓

* Subscription
MSDN Platforms (34de01eb-88a0-4743-9c1 ▼)

* Resource group ⓘ
☐ Create new ☒ Use existing
myazureRG ▼

* Location
East Asia ▼

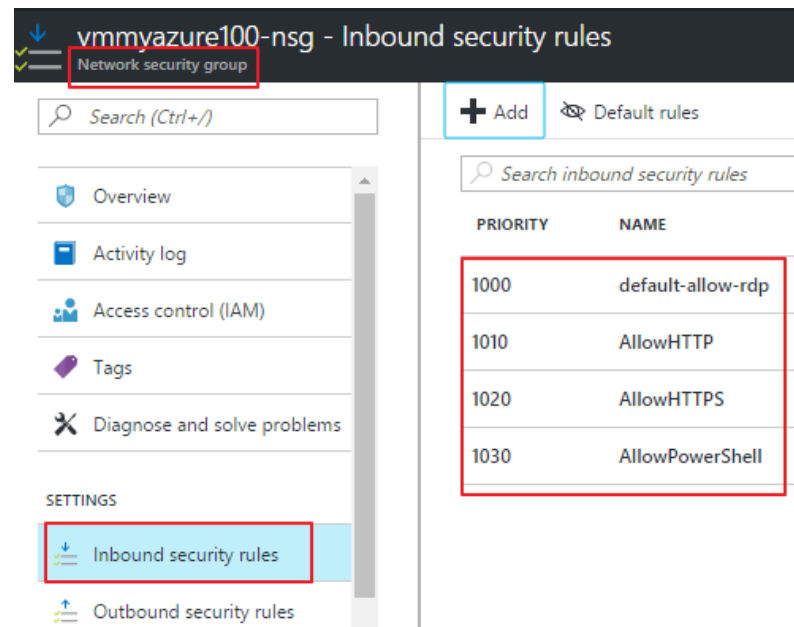
Fault domains ⓘ
2

Update domains ⓘ
5

Use managed disks ⓘ
☐ No (Classic) ☒ Yes (Aligned) ✓

7-Load Balancer 구성하기

- Azure Portal에서 Load Balancer를 구성하기
 - Windows VM 또는 Linux VM을 Azure Portal에서 2개를 생성한다
 - **VM 생성시 Managed Disk를 사용하도록 설정한다**
 - **각 VM 생성시** Network Security Group에 **Inbound Rule**에 **http, https**를 추가하여 RDP, ssh, http, https 접속을 허용한다 (**필수 작업**)
 - 각 VM은 동일한 Network Security Group을 사용한다
 - 공인 IP는 꼭 **수동**으로 설정할 것
 - **각 VM 생성시** 기존의 Availability Set인 **AVSET-Web**에 포함시킨다 (**필수 작업**)



7-Load Balancer 구성하기

- Azure Portal에서 Load Balancer를 구성하기
 - 각 VM에 Windows IIS를 설치하고 구성한다
 - `Install-WindowsFeature Web-Server -IncludeManagementTools`
 - `cd c:\inetpub\wwwroot`
 - default.htm 파일 생성
 - 이러한 작업을 나머지 VM에도 동일하게 한다
 - 작업 후에 Local Computer에서 Web Server에 접속해본다
 - 이제는 Internet으로 Web Server에 접속해본다

7-Load Balancer 구성하기

- Azure Portal에서 Load Balancer를 구성하기
 - Public IP 생성

Create public IP address

* Name
myazure-pip ✓

* IP address assignment
Dynamic Static

* Idle timeout (minutes) ⓘ
4

DNS name label ⓘ
myazurepip ✓
.eastasia.cloudapp.azure.com

* Subscription
MSDN Platforms (34de01eb-88a0-4743-9c1) ▼

* Resource group ⓘ
☐ Create new ☒ Use existing
myazureRG ▼

* Location
East Asia ▼

7-Load Balancer 구성하기

- Azure Portal에서 Load Balancer를 구성하기
 - Azure Load Balancer 생성
 - New - Networking - Load Balancer
 - 이름: **LB-web**
 - 기존에 생성한 Public IP Address를 선택한다
 - Load balancer 생성 후 세부 설정하기
 - 동작중이지 않는 VM들을 탐지하기 위해 **Probes** 설정하기
 - **LB-Probe**

7-Load Balancer 구성하기

- Azure Portal에서 Load Balancer를 구성하기

- Load balancer 생성 후 세부 설정하기

- VM들을 **Backend Pool**에 추가하기

- **LB-Backend**
- Availability set에 AVSET-WEB 추가
- VM 2개를 추가

- 포트를 위해 **Load Balancer rule** 생성하기

- **LB-Rule**

lb-backend
lb-web

Save Discard Delete

Name
lb-backend

IP version
IPv4 IPv6

Associated to
avset-web (availability set)

Target network IP configurations
Only VMs within the current availability set can be chosen. Once a VM is chosen, only network IP configurations related to it can be chosen.

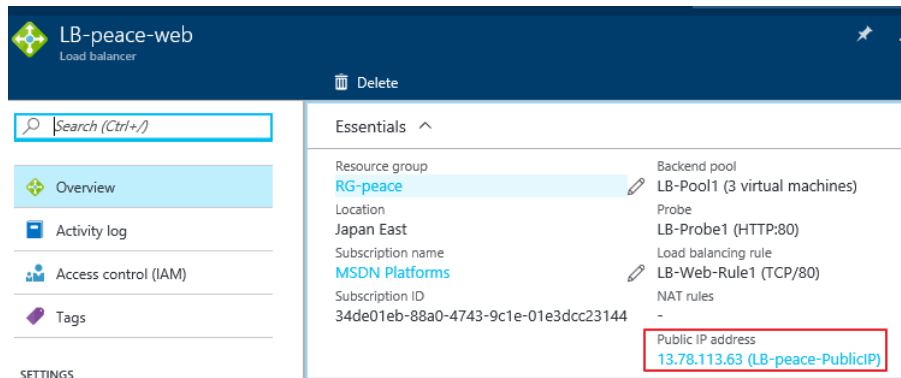
Virtual machine: vmwebtime101 Network IP configuration: vmwebtime101864/ipconfig1 (192.168.1.9)
Virtual machine: vmwebtime100 Network IP configuration: vmwebtime100351/ipconfig1 (192.168.1.8)

+ Add a target network IP configuration

Associated load balancing rules
lb-rule

7-Load Balancer 구성하기

- Azure Portal에서 Load Balancer를 구성하기
 - 자신의 Desktop 컴퓨터에서 Load Balancer의 공인 IP로 접속하기



- <http://13.78.113.63>
- 여러 개의 Web Browser를 실행하여 접속해 본다
- 그러면 어떤 때는 Web1 for yslee, 어떤 때는 Web2 for yslee 홈페이지에 접속한다

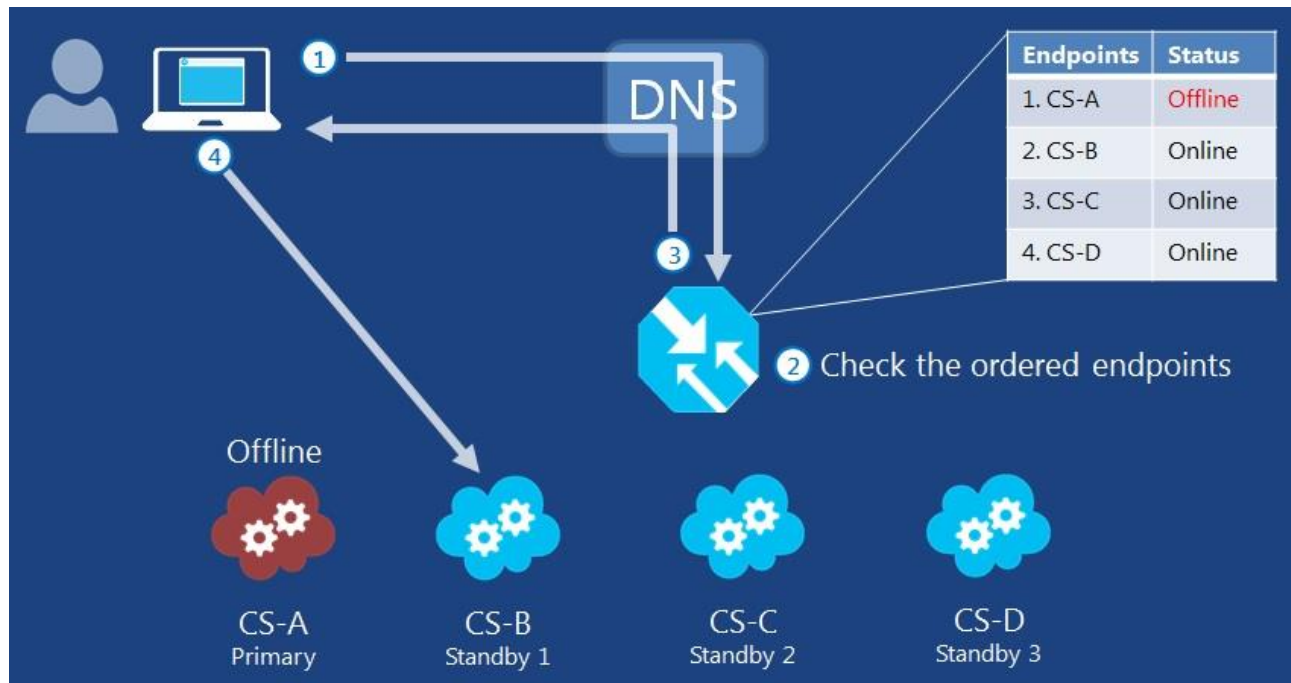
8-Traffic Manager 구성하기

- Traffic Manager란?
- Traffic Manager의 3가지 Routing 방법 이해하기
- **Azure Portal**에서 Traffic Manager 구성하기

8-Traffic Manager 구성하기

- Traffic Manager란?

- 지역적으로 떨어져서(다른 Azure Region에서) 운영 중인 Web site들을 위한 부하 분산하는 기능이다
- 또한 사용자들이 접속을 하면 가장 가까운 VM으로 트래픽을 보내주어 응답 속도가 빠르게 된다
- **Web App 및 VM에서 구성한 Web Site에 접속할 때** Traffic Manager의 도움을 받게 된다



8-Traffic Manager 구성하기

- Traffic Manager의 3가지 Routing 방법 이해하기
 - **Performance** (Performance)
 - 접속하는 컴퓨터와 가장 가까운 곳으로 연결하기
 - ** 괄호 안의 용어는 Azure Classic Portal에서 사용하던 것
 - **Priority** (Failover)
 - 접속하는 컴퓨터 Primary endpoint로 연결해 준다
 - 만약 Primary endpoint가 고장 나면 Secondary endpoint로 연결한다
 - **Weighted** (Round robin)
 - 설정한 Weight 값을 기준으로 Round robin으로 연결해준다

Routing method ⓘ

Weighted
Performance
Weighted
Priority

8-Traffic Manager 구성하기

- **Azure Portal**에서 Traffic Manager 구성하기
 - WebApp을 지역적으로 떨어진 **Japan East**와 **US Central**에 생성하기
 - **WebApp-JP1**
 - **WebApp-US1**

App Service Plan을
Traffic Manager가
지원되는 S1으로
설정해야 함
(D1은 안됨)

Web App

* App name
wa-peace-je ✓
.azurewebsites.net

* Subscription
MSDN Platforms (34de01eb-88a0-4743-9c ▼)

* Resource Group ⓘ
☐ Create new ☒ Use existing
RG-peace ▼

* App Service plan/Location
appserviceplan-je(Japan East) >

App Insights ⓘ

Web App

* App name
wa-peace-usc ✓
.azurewebsites.net

* Subscription
MSDN Platforms (34de01eb-88a0-4743-9c ▼)

* Resource Group ⓘ
☐ Create new ☒ Use existing
RG-peace ▼

* App Service plan/Location
appserviceplan-usc(Central US) >

App Insights ⓘ

8-Traffic Manager 구성하기

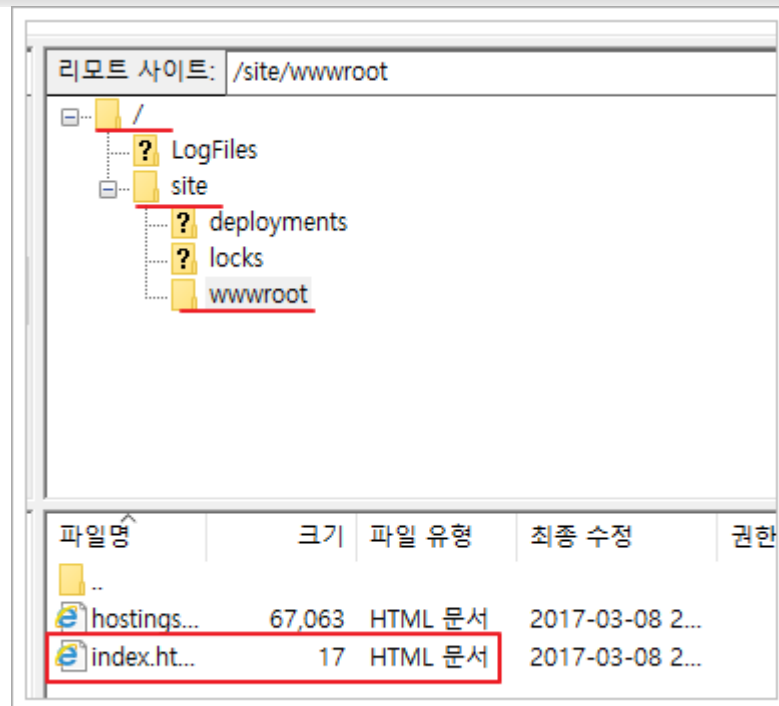
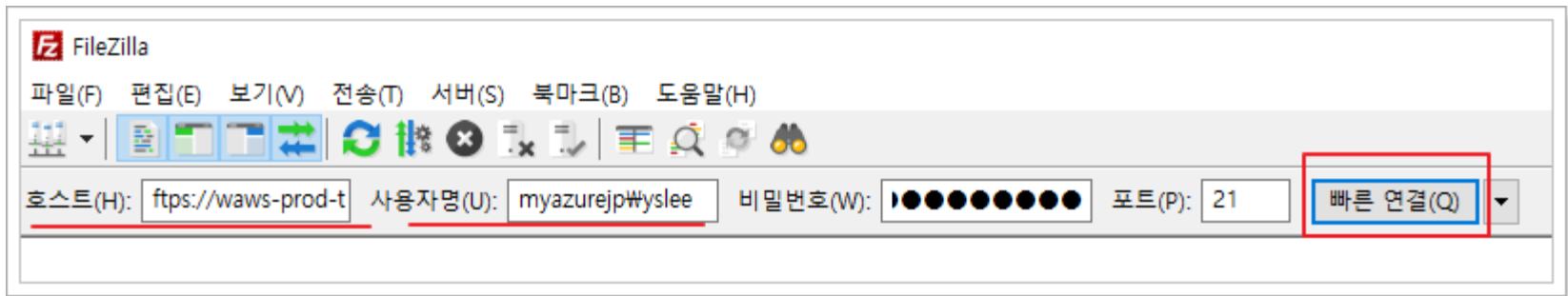
- **Azure Portal**에서 Traffic Manager 구성하기
 - WebApp의 FTP 주소와 사용자 이름을 확인한다

The screenshot displays the Azure Portal interface for an App Service named 'myazurejp'. The left sidebar contains navigation options, with '개요' (Overview) highlighted. The main content area shows the '필수' (Essentials) section, which includes various configuration details. The '리소스 그룹' (Resource Group) is 'EMSTest', the '상태' (Status) is 'Running', and the '위치' (Location) is 'Japan East'. The '구독 이름' (Subscription Name) is '무료 체험' (Free Trial), and the '구독 ID' (Subscription ID) is 'ea190aa4-76a4-456e-bf4d-f68b4091294c'. The 'URL' is 'http://myazurejp.azurewebsites.net'. The 'App Service 계획/가격 책정 계층' (App Service Plan/Pricing Tier) is 'myazurejp (공유)' (Shared). The 'FTP/배포 사용자 이름' (FTP/Deployment User Name) is 'myazurejpWyslee'. The 'FTP 호스트 이름' (FTP Host Name) is 'ftp://waws-prod-ty1-007.ftp.azurewebsites.windows.net'. The 'FTPS 호스트 이름' (FTPS Host Name) is 'ftps://waws-prod-ty1-007.ftp.azurewebsites.windows.net'.

필수	URL
리소스 그룹 (변경)	http://myazurejp.azurewebsites.net
EMSTest	App Service 계획/가격 책정 계층
상태	myazurejp (공유)
Running	FTP/배포 사용자 이름
위치	myazurejpWyslee
Japan East	FTP 호스트 이름
구독 이름 (변경)	ftp://waws-prod-ty1-007.ftp.azurewebsites.windows.net
무료 체험	FTPS 호스트 이름
구독 ID	ftps://waws-prod-ty1-007.ftp.azurewebsites.windows.net
ea190aa4-76a4-456e-bf4d-f68b4091294c	

8-Traffic Manager 구성하기

- **Azure Portal**에서 Traffic Manager 구성하기
 - FTP Server에 연결하여 파일을 upload한다



8-Traffic Manager 구성하기

- **Azure Portal**에서 Traffic Manager 구성하기
 - 홈페이지 기본 문서 파일을 추가하는 방법은 Console을 이용한다
 - 각 App Service의 웹사이트의 Console에서 default.htm 파일을 생성한다
 - echo "Welcome to Japan Web site" > default.htm
 - echo "Hello, Everyone," > default.htm

8-Traffic Manager 구성하기

- Azure Portal에서 Traffic Manager 구성하기
 - **Traffic Manager Profiles** 생성하기
 - **More Services - Traffic Manager** 검색 - **Add**
 - 세부 구성하기: Endpoints 구성하기
 - Target Resource Type을 **App Service**로 선택
 - Target Resource는 사전에 만든 WebApp-US1, WebApp-JP1을 선택

* Name
yslee ✓
.trafficmanager.net

Routing method
Performance ▼

* Subscription
MSDN Platforms (34de01eb-88a0-4743-9c ▼

* Resource group ⓘ
☐ Create new ☒ Use existing

RG-peace ▼

Add endpoint
yslee

Type ⓘ
Azure endpoint ▼

* Name
yslee-us ✓

Target resource type
App Service ▼

* Target resource
WebApp-US1 >

Add endpoint
yslee

Type ⓘ
Azure endpoint ▼

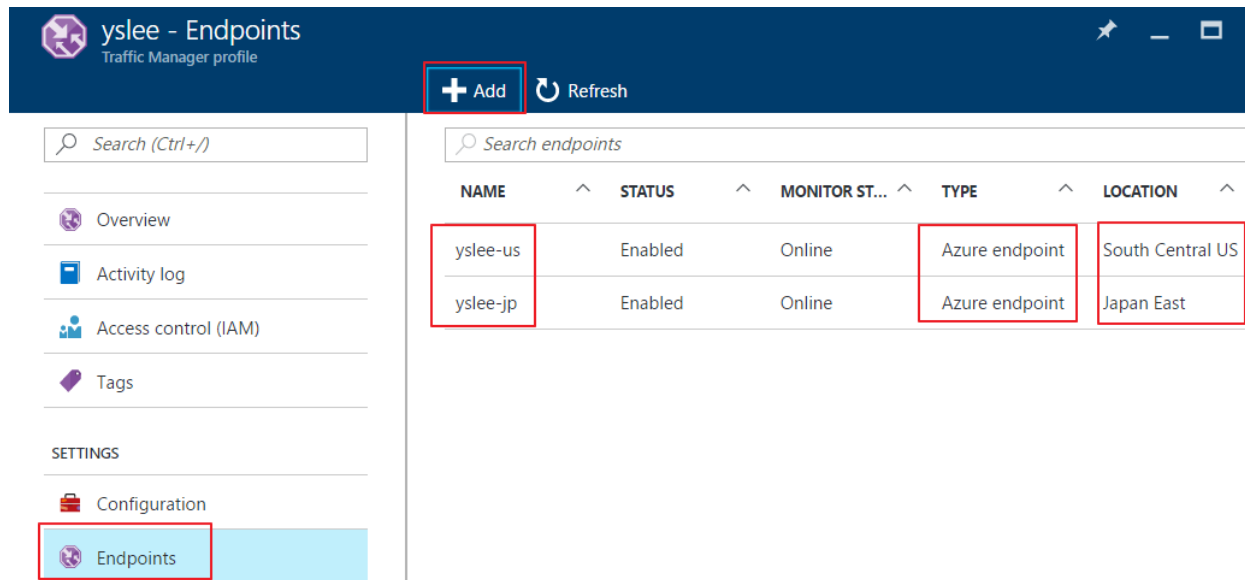
* Name
yslee-jp ✓

Target resource type
App Service ▼

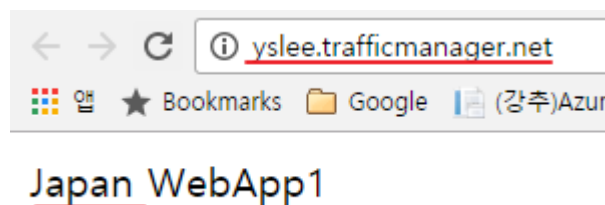
* Target resource
WebApp-JP1 >

8-Traffic Manager 구성하기

- Azure Portal에서 Traffic Manager 구성하기
 - **Traffic Manager Profiles** 생성하기



- Traffic Manager 주소록 접속하기



8-Traffic Manager 구성하기

- Azure Portal에서 Traffic Manager 구성하기
 - **Traffic Manager Profiles** 생성하기
 - Japan East의 WebApp을 중지한 후 다시 접속하기

App Services

기본 디렉터리 (jesuswithmehot...)

+ Add Columns Refresh

Subscriptions: MSDN Platforms – Don't see a subscription? [Switch directories](#)

Filter items...

NAME

- WebApp-JP1
- WebApp-US1

Search (Ctrl+/)

- Overview
- Activity log
- Access control (IAM)
- Tags
- Diagnose and solve problems

WebApp-JP1

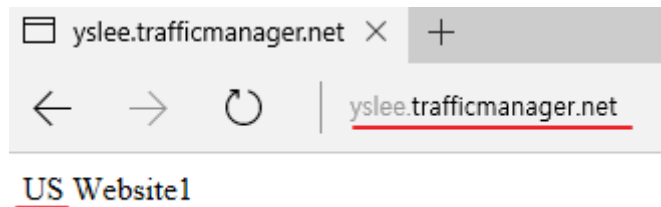
App Service

Browse Start Swap Restart Delete

⚠ Your app is stopped. App Service plan charges still apply.

Essentials ^

Resource group	URL
rgYSLee	http://webapp-jp1.azurewebsites.net
Status	App Service plan/pricing tier
Stopped	WebAppServicePlan2 (Standard: 1 Small)
Location	FTP/deployment username
Japan East	No FTP/deployment user set
Subscription name	FTP hostname
MSDN Platforms	ftp://waws-prod-ty1-005.ftp.azurewebsites.net

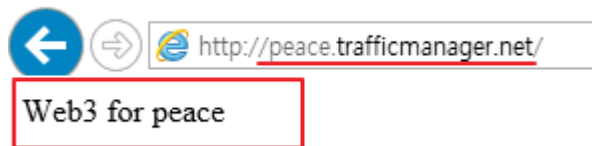


8-Traffic Manager 구성하기

- Azure Portal에서 Traffic Manager 구성하기

- **Traffic Manager Profiles** 생성하기

- 현재는 Japan East의 WebApp이 중지 되어 있고, 다시 Japan East의 VM에서 운영 중인 Web site를 추가한 상태에서 다시 Traffic Manager 주소로 접속하기



- http://peace.trafficmanager.net으로 접속을 하니 가까이에 있는 Japan East에서 실행 중인 Web site인 "Web3 for peace"에 접속이 되었다
 - 더 이상 US Central의 Web site에 접속하지 않는다
 - Traffic Manager로 Web site 운영할 때의 결론
 - Traffic Manager Profiles의 Endpoints에는 **Cloud service, App service, Public IP address**가 모두 가능하다
 - Traffic Manager를 생성한 후 그 주소로 접속하면 사용자의 위치와 가까이에 있는 Web site에 접속하게 된다

8-Traffic Manager 구성하기

- Azure Portal에서 Traffic Manager 구성하기
 - Traffic Manager로 Web site 운영할 때의 결론
 - Traffic Manager Profiles의 Endpoints에는 **Cloud service, App service, Public IP address**가 모두 가능하다
 - Traffic Manager를 생성한 후 그 주소로 접속하면 라우팅 방법(성능, 가중, 우선순위)에 따라 Web site에 접속하게 된다
 - 어느 쪽으로 접속이 되는지 확인하려면 **nslookup peace.trafficmanager.net**을 실행해보면 알 수 있다.
- 공인 DNS Server에 CNAME 생성하여 운영
 - peace.trafficmanager.net으로 고객이 접속하면 어느 회사 것인지 모른다. 그래서 회사 공인 DNS server(Azure DNS도 좋음)에서 www.myazure.kr을 만들어 CNAME 레코드를 peace.trafficmanager.net으로 생성하면, 고객이 앞으로는 www.myazure.kr로 접속하면 peace.trafficmanager.net으로 연결되고, 최종적으로 Azure Cloud에서 운영중인 Web site에 접속하게 된다.
 - 이것을 위해서는 첫 번째 Web App에서 **Custom domain을 생성하여 www.myazure.kr를 사용하도록** 해야 한다. 물론 두 번째 Web App에서도 동일하게 작업을 반드시 해주어야 된다.