# Introdução

- Azure oferece serviços de cloud computing, onde você paga pelo aquilo que você usa (Pay-As-You-Go).

- Azure é orientado a grupo de recursos.

- Apartir do Azure CLI/ssh/power shell/python dá para criar múltiplos recursos ao mesmo tempo

## Certificações

AZ-900 -> fundamentos (Iniciante)

AZ-10X -> IT Professional

AZ-20X -> Developers (AZ-203)

AZ-30X -> Architects (AZ-301) Ganha $$$$$$$

AZ-40X -> DevOps (Developer + Operational)

AZ-500 -> Azure Security

## Tipos de núvem

### Pública

Serviço contratado em grande escala com alto desempenho, baixo custo e escalabilidade.

### Privada

Datacenter privado com alta segurança com alto investimento e desperdício de recursos

## Subscriptions

- Subscription é uma assinatura paga que tem seus limites e restrições. Outra assinatura é necessária quando atingir o limite de recursos por região (MUITO IMPROVÁVEL) ou para atribuir para cada filial da empresa.

- Cada subscription é colocar dentro de um diretório. Para visualizar todas as subscriptions, tem que ir até Switch Directory e selecionar todas as assinaturas.

### Policies

Por default todos os usuários associados a assinatura podem fazer qualquer coisa. As policies são regras para limitar o uso de recursos.

Policies úteis:

* Filtrar Tipos de recursos (SKU) para cada usuário: Exemplo: SKU: Basic B1
* Definir versão do SQL server

## Grupo de recursos

- A partir do grupo de serviços/recursos podemos implementar, gerenciar e monitorar todos os recursos.

**- OS GRUPOS DE RECURSOS SÃO CRIADOS PARA CADA RECURSO DA EMPRESA, EXEMPLO: APP, WEBSITE, CRM... Cada grupo de recurso tem usuário responsáveis e politicas.**

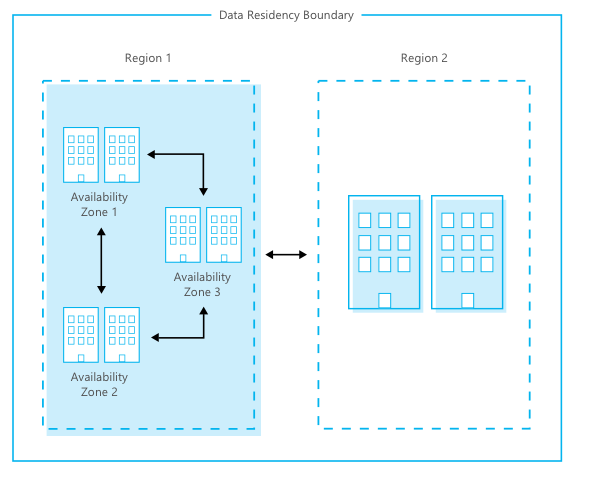
**- Removendo Resource Group remove todos as configurações, scripts, dados das aplicações**

- Permite gerenciar a infraestrutura a partir dos modelos declarativos de invés de scripts

- Podemos definir as dependências entre os recursos para que eles sejam implantados na ordem correta.

- Declarar permissão de acesso para cada grupo de recursos e auditar as atividades de cada usuário.

## Infraestrutura

- Region é um ou mais availability zones (data centers). Exemple (South-Brazil=SP)

- Em cada *Region* sempre azure tenta manter mais do que 1 Availability Zone para ter alta disponibilidade (full resiliency).

- Os datacenter são conectados via rede regional com baixa latência.

- Cada availability zone (data center). Os data centers são geograficamente separados com energia, refrigeração e rede completamente separada.

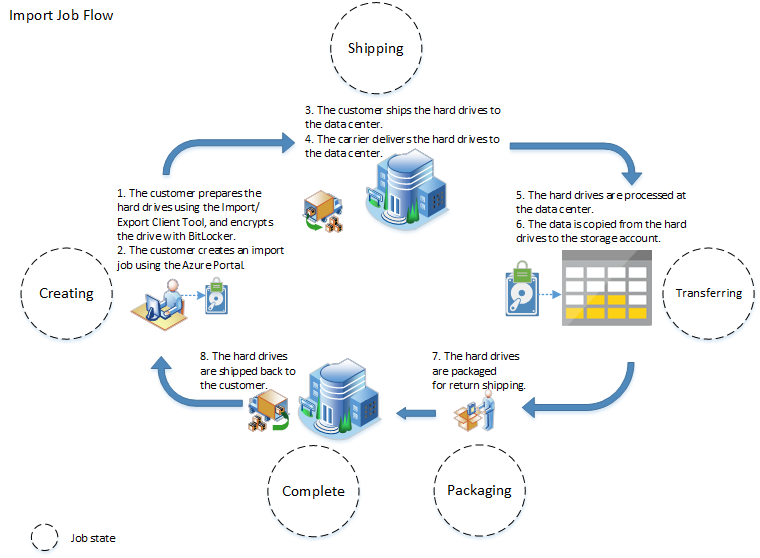
## Procurar disponibilidade do produto nas regiões

<https://azure.microsoft.com/en-us/global-infrastructure/services/>

# Export Grande volume de dados

- Quando é necessário exportar grande volume de dados para storage no Azure (> 1tb). Fica inviável fazer via upload. Por isso Existe Azure Import/Export.

- Azure Import/Export é um serviço onde o usuário envia seus dados em HDs para o datacenter mais perto. Os dados são transferidos para o HD via Export Client Tool e são criptografados via BitLocker.



## Criação de nova conta

- Quando é criado uma nova conta, a Azure estabelece alguns limites, como limite de máquinas virtuais. Este limite aumenta conforme o uso ou abrindo um ticket. Limites:

<https://docs.microsoft.com/en-us/azure/azure-subscription-service-limits>

- A conta free tem $200 dolares para 1 mês e 12 meses de serviços gratuitos.

# Data Redundancy

Data redundancy é redundância do storage em mais que um SSUnit em servidor/datacenter/zona

## Tipos

**Read-access geo-redundant storage (RA-GRS):** Arquivos armazenados e lidos de diversas regiões e availability zones. (Podem existir aplicações de outras regiões lendo os arquivos)

**Geo-redundant storage (GRS):** Arquvios armazenados (backup) em regiões diferentes. **Só consegue ler da região principal. Se a região principal cair, storage redundantes converterá para ativo e será o único principal acessível**

**Locally-redundant storage (LRS):** Arquivos armazenados dentro de um único data center.

**Zone-redundant storage (ZRS):** Arquivos armazenados em todos os datacenters do zone. Os arquivos são replicados em 3 diferentes availability zones dentro da mesma zone region.

# Disk Caching

- É uma técnica de utilização memória RAM e SSD para melhorar a performance em VHD (Virtual Hard disk, como por exemplo os disks dos vms).

- Disk caching é disponível em SSH Standard e Premium.

- Se o disco de caching é read-only a latência e menor e IOPS é maior.

- Se o disk é Read Write a latência é maior e o IOPS é menor

**- NÃO PODE ALTERAR O TIPO DE DISK CACHING PARA L-SERIES E B-SERIES VM.**

# Availability Set

Quando cria vários discos ou VMs, não temos garantia se serão criados no mesmo rack de servidores ou não. Por isso existe a opção de availability set que define isto.

Fault Domain: 0 até 4, distribuição em racks (fault domain para evitar se um desastre acontecer um um dos racks)

Update Domain: 0 até 20, distribuição em servidores (quando atualizam ou aplicam um patch de segurança em um servidor a VM será movida para outro servidor e depois da atualização e colocado de volta)

# Virtual Machine – VM

- Quando está sendo criada a máquina virtual, há campo para inserir a licencia de sistema operacional.

- Podemos Criar um template com uma imagem para subir na máquina virtual

## Tier

A: Básicas para testes e desenvolvimento

B: Burstable para usar capacidade máxima de CPU **(se não usar toda capacidade máxima retorna créditos para conta)**

D: General Purpose (para empresas e storage)

E: Memoria

F: CPU

G: Instâncias grandes

H: High performance compute – Cientistas de dados

L: Storage Velocidade I/O

M: Large Memory RAM (TB)

N: GPU

SAP Hana

## Azure Compute Units

- Unidade criada pela Microsoft para definir a velocidade de processamento.

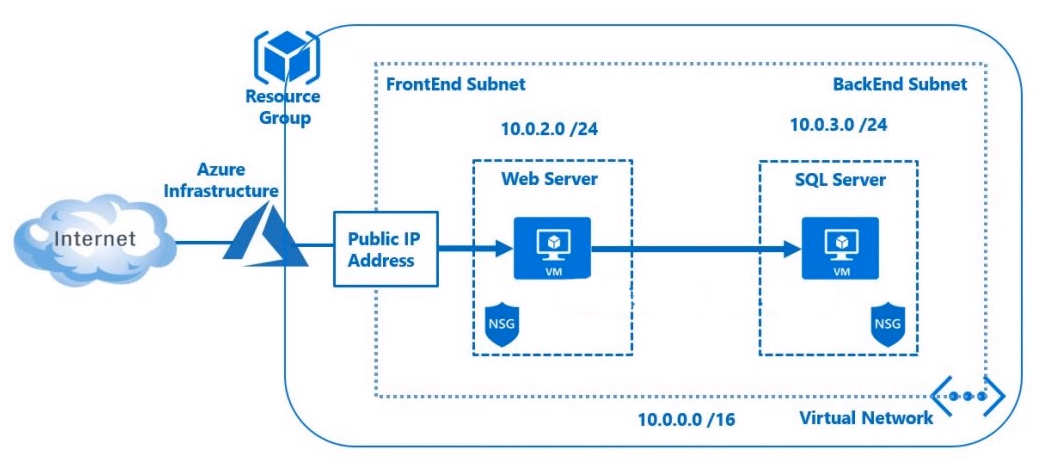
**- Cada ACU é definido para cada núcleo de CPU**

Exemplo máquina virtual de 2CPU de 100ACU, então o total será 2x100 = 200ACU

# Networking

- Os endereços IP públicos são aqueles são não estão dentro da faixa de IPs da Classe A, B, C, D e E de endereços IP privados.

**EM TODA SUBREDE TEM 4 ENDEREÇOS RESERVADOS: REDE .0, BROADCAST .255 E OS 4 PRIMEIROS ENDEREÇOS DE CADA FAIXA SÃO RESERVADOS PARA DNS E GATEWAY (.1 .2 .3)**



# \_\_\_\_\_\_\_\_\_\_\_\_\_\_ PRÁTICA \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Resouce Groups

**SEMPRE COLOCAR TAGS PARA GERAR RELATÓRIOS:**

* Name
* developementTeam

## Exportar configuração de resource group

Selecionar Group -> Export template -> selecionar Template, CLI ou PowerShell -> Download

# Billing

## Alertas

Cost Management + billing -> Selecionar a Assinatura -> Budgets -> Add

# PowerShell

**É RECOMENDÁVEL UTILIZAR POWERSHELL 6.2.3 VIA GITHUB**

PowerShell é um console instalável em: Windows, Linux, macOS

## Instalar Azure no PowerShell

## Install-Module -Name Az -AllowClobber

## CloudShell

<https://shell.azure.com/powershell>

Cloud Shell é um console web. Para usar é necessário ter Storage montado.

## Comandos

**Conectar:** Connect-AzAccount

**Listar Vms:** Get-AzVM

**Listar Subscriptions:** Get-AzSubscription

**Listar Resource Groups:** Get-AzResource

## Criar Storage

New-AzStorageAccount -ResourceGroupName backend -Name storagenamewww -Location eastus -SkuName Standard\_LRS -kind StorageV2 -AccessTier Hot

# Cloud Shell

## Criar Storage

az storage account create \

--name <account-name> \

--resource-group storage-resource-group \

--location westus \

--sku Standard\_RAGRS \

--kind StorageV2

# Storage

## Blob

**NÃO EXISTE PERMISSÃO PARA ARQUIVOS. É O MELHOR PARA COLOCAR VÍDEOS**

**Blob/Big Bucket:** É um storage para armazenamento gigante de dados não estruturados (textos e arquivos binário). Alguns usos:

* Hospedagem de arquivos públicos que são servidos para o navegador via streaming (imagens e vídeos, docs..)
* Backups

## File Storage

* Recomendado para empresas parece dropbox
* Armazenar e compartilhar arquivos TXT, XLS, DOC...etc.
* Unidade montável
* Azure file Sync (Parece Dropbox)

## Table Storage

* Tabelas NoSQL.
* Tabelas podem ser estruturadas ou não
* Usado para protocolos OData e LinQ Queries

## Queue Storage

**Storage para alto índice de trafego e alto desempenho**, onde é acessível via REST API

Exemplo: App para enviar mensagens para clientes (via SMS, email...etc).

# Storage Accounts

<https://docs.microsoft.com/en-us/azure/storage/common/storage-account-overview?toc=%2fazure%2fstorage%2fblobs%2ftoc.json>

<https://docs.microsoft.com/en-us/azure/storage/common/storage-scalability-targets?toc=%2fazure%2fstorage%2fblobs%2ftoc.json>

## Account Kind

**Storage V1 (GPV1)**: ANTIGA Blobs, files, tables. Não suporta hot & cold **(NÃO USAR)**

**Storage V2 (GPV2)**: New para blobs, files, tables

**BlobStorage**: Blobs

## Account Tier

**Hot**: Acesso com muita frequência – Acesso diário (mais caro):

* Documentos, vídeos, Planilhas

**Cold**: Acesso com menos frequência (latência para espalhamento nos datacenters):

* Backups
* **Se acessar cold storage frequentemente vai ter extra charge (cobrança extra)**

**Archive**: Dados acessados raramente e armazenados por pelo menos 180 dia (latência de horas para espalhamento nos datacenters).

## Performance

**Discos standard:** mais barata, spinning disks (HD normais)

**Premium:** mais cara, SSD Disk (alta velocidade de R/W)

# Storage – Criação & Utilização

## Configuração Padrão

**Name**: Único

**Location**: South Brazil

**Performance**: Standard

**Account** **kind**: Storage V2

**Replication**: LRS

**Security transfer**: enabled

**Allow access from**: All networks

**Blob soft delete**: disabled (será cobrado se enabled)

**Data Lake Storage Gen2**: Disabled (Para bigdata, Milhões ou bilhões de documentos)

**Tags**: Name:Storage01

## Utilização

1. Via Storage Explorer (no Storage account -> selecionar storage -> Storage Explorer)
2. Instalar Azure Storage

- Blobs Container, file Share = pasta (private).

- Blobs e file shares files são acessíveis publicamente.

- Tables são noSQL

## Editar Configurações

Storage Accounts -> Selecionar o Storage -> Configuration

## Permissões

### Access Policy

- São regras gerais aplicados em todo o container. Sem filtros de arquivo, IP ou outra coisa

#### Níveis de acesso

* Private (Não permitir acesso anônimo)
* Blob (Acesso anônimo somete para Blobs)
* Container (Acesso anônimo para blobs e containers)

#### Alterar nível de acesso

* Na data de criação
* Storage -> Blob service -> Blobs -> selecionar container -> Change access level

### Access Keys

- Os access Keys são usados para autenticar as aplicações para usar Azure storage. A autenticação é necessária para que as regras SAS – Share Access Signature sejam implementadas.

- Existem duas chaves para poder fazer a renovação

### SAS – Share Access Signature

SAS são regras de permissões mais especificas para atribuir nos containers ou nos arquivos.

- Podem existir mais do que uma regra de permissão, que funcionarão ao mesmo tempo.

- Existem 2 chaves de acesso para pode recriar a mesma regra com outra chave e renovar a chave antiga (para não interromper a aplicação). Isso é necessário se a chave for comprometida.

**- Quando regenerar o key1 ou key2, tudo que estiver usando vai parar de funcionar.**

#### Níveis de acesso

Selecionar Data/Hora de acesso

Permissões: CRUD

IP

Https only

#### Criar Regras e tokens

Storage -> Selecionar Storage -> Shared access signature -> Definir as regras -> Generate SAS and connection string -> Copiar SAS token para colocar no final de URL de cada arquivo ou usar nos apps

#### Criar Modelo de Shared Access Signature para usar

Software Azure Storage Explorer -> Selecionar Container -> Right Click -> Manage Access Policies -> Da nome no Id e definir as regras

#### Utilizar Model já criado para gerar tokens

Selecionar container -> Right click -> Get Shared access signature -> selecionar policy -> Create

## Upload files from pc to Azure

1. Download azCopy.exe
2. Azure -> Storage Accounts -> Selecionar Storage -> Storage Explorer -> Selecionar Container -> Botão direito -> Get Shared Access Signature -> Selecionar todas as permissões -> Create
3. Copiar URL
4. PowerShell -> .\azcopy.exe copy "C:\imagens\\*" urlCopiado –recursive

# Dominio Apontando para Azure Storage

1. Storage Accounts -> selecionar Storage -> Custom domain -> Copiar URL
2. Abrir página do domínio -> Criar CName -> Colar URL -> Create
3. Storage Accounts -> selecionar Storage -> Custom domain -> Colocar nosso domínio que está apontando

# Hospedagem de Websites estáticos

A hospedagem estática de websites estáticos é feita dentro de um blob storage. Quando é habilitado o recurso é gerado um blob container automaticamente:

1. Storage Accounts -> Selecionar storage -> Static website -> Enabled
2. Escrever o nome do arquivo .html/.php do site e da página de erro
3. Acessar o blob $web

# Máquinas Virtuais

- Quandro cria um VM, é criado um resource group “NetworkWatcherRG” para poder monitorar e fazer logging das vms

## Criação

BASICS

* Authentication Type: Password (Windows) / SSH Public key (Linux)
* Availability Options: Configurar availabity set caso precisa distribuir os recursos em servidores e racks diferentes
* Select inbound ports: Allow selected ports
* Select inbound ports: RDP

DISKS

* OS disk type: Premium SSD ou Standard SSD (Depende de IOPS e Budget)
* Data disk: Criar somente se precisar mais do que 1 partição

NETWORKING

* Virtual Network: Selecionar se já estiver VNet
* Subnet: Selecionar se já estiver subnet
* Public IP: Usar existente ou Criar Novo
* NIC network security group: basic
* Public inbound ports: Allow selected ports
* Select inbound ports: SSH(Linux)/RDP(Windows)
* Load Balancing: NO

MANAGEMENT

* Boot diagnostics: on
* Os guest diagnostics: off
* Identity: off
* Auto-shutdown: on
* Notification before shutdown: on
* Enable Backup: Somente para máquina de produção

## Acesso

Via Remote Desktop (windows)

Via SSH (linux)

## Máquina virtual Linux

1. Generate RSA public and private key no puttygen e salvar em arquivos os keys
2. Colocar o SSH Public key no formulário de criação da VM (Seguir as etapas de cima)
3. Colocar o SSH Private key no putty para acesso remoto (Putty -> SSH -> Auth -> Browse)
4. Azure -> VM Selecionar VM -> Connect -> Copiar comando para conectar: Usuario@IP
5. Colar dentro do Putty -> Session -> Host Name

## Adicionar HD/SSD Extra no VM

**- Sempre colocar os discos extra dentro da mesma região da VM (para evitar latência)**

Azure -> Virtual Machines -> Selecionar VM -> Disks -> Add data disk -> (Create New)Disk -> Save

## Alterar Disk Caching

- O Disk caching utilizar SSD e memória RAM para aumentar a performance e IOPS do disco virtual (VHD).

**- NÃO PODE ALTERAR O TIPO DE DISK CACHING PARA L-SERIES E B-SERIES VM.**

- Para Alterar: Virtual Machines -> selecionar VM -> Disks -> Edit -> Host caching -> selecionar

## Networking Interface

- Quando é criado uma máquina virtual é criado também uma NIC (Network interface Card) com IP privado e IP público.

- Os IPs privados são para conversa entre VM dentro do mesmo VNet.

- O Network Interface Card pode ser movida para outra VM. 1 VM pode ter mais de uma Network Interface Card.

No Network Interface Card Podemos:

* Associar para uma máquina virtual
* Mover para outra máquina virtual
* Atribuir DNS
* Definir IP público dinâmico ou estático
* Alterar IP público ou dinâmico
* Criar e reservar no IP público
* Alterar o subnet/Virtual network
* Ativar/desativar o IP público

## Conectar a um file storage

1. Criar File storage com RA-GRS
2. Account Storage -> Selecionar Storage Criado -> File share -> +File Share -> Definir Name e tamanho de armazenamento -> Create
3. Selecionar o file share criado -> Connect -> Escolher letra/nome -> copiar comando
4. (Linux) conectar no VM via SSH -> Colar comando

(Windows) conectar no Windows via RDP -> PowerShell -> Colar comando

## Availability Set

**- Não há custo para availability Set**

- É configuração de distribuição dos recursos em racks por preferência ou caso haja erros ou atualização no rack atual.

- SÓ PODEMOS DEFINIR AVAILABILITY SET QUANDO O VM É CRIADO, NÃO PODEMOS ALTERAR.

- Fault domains (distribuição em racks)

- Update domains (distribuição em servidores)

- Criar Availability Set: Azure -> Availability Set -> Add

VM -> Add -> Availability -> Availability set

- Consultar rack e servidor do vm ou qualquer outro recurso:

Azure -> Availability Set -> Overview -> Consultar Fault domain (Rack) e Update domain (servidor)

## VM Scale set (Autoscale)

O scaling é necessário para aumentar ou diminuir os recursos ou número maquinas VM conforme o uso. É definido um set de regras para escalar, exemplo se uso médio de CPU em todos os servidores for maior de 75% sobre mais 1 máquina.

- Quando a porcentagem de CPU subir mais do que especificado de maneira bem rápida, o azure viola as regras e sobe o número de VMs drasticamente (+3 ou + 4)

- Criação: Azure -> virtual machine scale set

* Zones: 1,2,3
* Instances count: número inicial de VMs (1)
* Instance size: Standard B1ls
* Autoscale: Enabled
* Minimum Number
* Maximum Number
* Scale out (aumento) -> CPU Threshold: 75% (AVG do CPU de todas as instâncias)
* Scale In (diminuição) -> CPU Threshold: 25% (Médio do CPU de todas as instâncias)
* Choose Load balacing options: Load Balancer
* Public IP Address Name
* Domain name label
* Subnet: Create new -> Selecionar todos os IPs -> Create
* Public ip address per instance: Off

- Após de criar o scale set:

* Alterar “durantion” a duração que a CPU ficará dentro dos rules para poder fazer o scale: Autoscale -> Scaling -> Selecionar when (Avarage Percentage CPU > 75)
* Ativar Notification por email: Azure -> Autoscaling -> Scaling -> Notify -> Emails Coadministrators

- Teste:

1. Abrir 2 sessões SSH
2. Na primeira sessão executar htop para monitorar o uso de CPU
3. Em outras 2 sessões executar o comando: yes

# Networking

## Criar VNet – Virtual Network (Privada)

**- EM TODA SUBREDE TEM 4 ENDEREÇOS RESERVADOS: REDE .0, BROADCAST .255 E OS 4 PRIMEIROS ENDEREÇOS DE CADA FAIXA SÃO RESERVADOS PARA DNS E GATEWAY (.1 .2 .3)**

**- SEMPRE DEVEMOS SEGUEMENTAR A REDE PARA SUBREDES**

- Por default é permitido a conversa entre os Subnets que estão dentro da mesma VNet. No caso dos VMs tem 2 regras AllowVnetOutBound e AllowVnetInBound

Criar VNet: Virtual Networks -> add -> preencher:

* Name: Nome do recurso a ser criado
* Address space: Rede 172.16.0.0/16 ou 10.0.0.0/16 ou 192.168.0.0/16
* Subnet Name: nome da subnet (SubA-Web)
* Address range: SubRede 172.16.1.0/24 ou 10.0.1.0/24 ou 192.168.1.0/24
* DDOS protection: Basic (Para evitar extra charge)
* Service endpoints: Disabled
* Firewall: Disabled

## Criar Subnet

Azure -> Virtual Networks -> Selecionar VNet -> Subnets -> +Subnet

* Name: Nome da subnet
* Address range CIDR block: subnet 172.16.2.0/24
* Network security group: None
* Route table: None
* Services: 0 selected
* Delegrate subnet to a service: None

## Visualização Diagrama da Rede, Subredes e Hosts

Azure -> Virtual Networks -> Selecionar VNet -> Diagram

## Routing

- A tabela de roteamento dentro da VNet tem pelo menos:

* 1 Entrada para as redes internas
* 1 Entrada para Internet Gateway
* 1 Entrada para VNet Gateway (VPN, S2S...)

### Roteamento Subnet para Firewall

1. Criar um Firewall e configurar
2. Azure -> Route Tables -> +Add -> Criar route no mesmo region
3. Azure -> Route Tables -> Selecionar Route Criada -> Routes -> +Add -> Configurar a saída de qualquer IP via firewall:

* Route name: Internet-via-firewall
* Address prefix: 0.0.0.0/0 (prefixo que quero chegar = Qualquer IP)
* Next hop type: Virtual appliance (Dispositivo interno com IP)
* Next hop address: 172.16.10.1 (Endereço do Firewall)

1. Azure -> Route Tables -> Selecionar Route -> Subnets -> +Associate -> Selecionar Virtual network e Subnet que o roteamento será aplicado.

## Conexão entre 2 VNets diferentes – Peers

- Por default 2 subnets dentro da mesma VNet se conversam. Enquanto subnets dentro de VNets diferentes não se conversam. Por isso Utilizamos Peering/Neighbor

Peering:

1. Virtual Networks -> Selecionar primeira VNet -> Peering -> +Add

* Name: Vnet1ToVnet2
* Virtual network deployment model: Resource manager
* Virtual network: Selecionar a outra Virtual Network
* Name: Vnet2ToVnet1
* Allow virtual network access Vnet1 to Vnet2: Enabled
* Allow virtual network access Vnet2 to Vnet1: Enabled
* Configure forwarded traffic Vnet1 to Vnet2: Disabled (Enable só se pode direcionar os pacotes de internet de Vnet1 até Vnet2 – Vnet1 via como intermediário)
* Allow gateway transit: Marcar somente se outra Vnet pode usar o gateway na Vnet atual

## Network Security Group

É um filtro de rede que funciona como se fosse firewall.

**- Toda Vm e subnet tem NSG, então o trafego começa com NSG da VMOri, NSG da SubnetOri, NSG da SubnetDes até NSG da VMDest.**

**- As configurações Inbound security rules e Outbound security rules não são aplicadas de imediato, demora +- 1 minuto.**

- Filtros permitir/negar:

* Protocolos
* Número de portas
* Source/destination IP.

- Definir regras Inbound/Outbound para aplicar os filtros na entrada ou na saída.

- Regras do NSG podem ser aplicados no NIC (Network Interface Card) ou no Subnet.

### Criar Network Security Group para Subnet

1. Network Security Groups -> +Add -> escolher o mesmo Location do subnet -> create
2. Network Security Groups -> Selecionar NSG criado -> Subnets -> Associate -> Escolher Vnet e Subnet
3. Settings -> configurar Inbound security rules e Outbound security rules (ficar atento o ip de origem e destino no inbound e outbound é inverso). Definir prioridade com número menor para ter mais prioridade (não ser sobreescrita)

## Network Watcher

É um grupo de recursos para monitorar e visualizar a topologia da nossa rede

**- Deve ser habilitado na região aonde estamos.**

### Connection Monitor

- Monitorar acessibilidade do trafego, latência e mudança da topologia entre um ponto e outro (VM para VM ou VM para URI). É útil para verificar se a conexão caiu em algum momento em algum dos VMs.

### Ip flow verify

É uma verificação (ping) de um endereço para outro (Este recurso utiliza VM). **Este recurso mostra também a regra que liberou/bloqueou o trafego.**

### Next Hop

Verificar os hops para o pacote chegar no destino. **Este recurso é útil para debugar o funcionamento da tabela de roteamento.**

### Packet Capture (Wireshark)

É um recurso para capturar todos os pacotes que trafegam in/out de uma máquina virtual. Este recurso aceita filtros de pacotes, baixar caption para abrir no wireshard.

### Connection Troubeshoot

Verificar a conexão entre uma VM para outra. Obtendo informações de latência e portas abertas

## VPN

Serve para conectar uma máquina/host em uma rede virtual. Isso possibilita colocar a nossa máquina local na rede virtual da azure e se conectar nos VMs com os IPs privados e usando um canal criptografado.

### P2S (Point to Site)

- É uma conexão ao VPN Gateway utilizando uma aplicação. Após da autenticação na aplicação é aberto um SSTP tunnel para encriptar os dados.

- Este tipo de VPN é o mais interessante por que pode usar em qualquer computador ou dispositivo móvel.

1- Criar VNet: Virtual Networks -> add -> preencher:

* Name: Nome do recurso a ser criado
* Address space: Rede 172.16.0.0/16 ou 10.0.0.0/16 ou 192.168.0.0/16
* Subnet Name: nome da subnet (SubA-Web)
* Address range: SubRede 172.16.1.0/24 ou 10.0.1.0/24 ou 192.168.1.0/24
* DDOS protection: Basic (Para evitar extra charge)
* Service endpoints: Disabled
* Firewall: Disabled

2- Criar Virtual Network Gateway: **Pode Demorar +- 40 minutos**

Azure -> Virutal Network Gateway -> +Add

* Selecionar Subscription
* Name: vpn-gateway
* Region: A mesma do Virtual Network
* Gateway Type: VPN
* VPN Type: Route-based
* SKU: Basic
* Selecionar Virtual Network criada
* Public IP Address: Create New
* Public IP address name: vpn-ip
* Enable active-active mode: Disabled
* Configure BGP ASN: Disabled

3- Gerar Certificados:

Powershell:

Comando 1

$cert = New-SelfSignedCertificate -Type Custom -KeySpec Signature `

-Subject "CN=P2SRootCert" -KeyExportPolicy Exportable `

-HashAlgorithm sha256 -KeyLength 2048 `

-CertStoreLocation "Cert:\CurrentUser\My" -KeyUsageProperty Sign -KeyUsage CertSign

Comando 2

New-SelfSignedCertificate -Type Custom -DnsName P2SChildCert -KeySpec Signature `

-Subject "CN=P2SChildCert" -KeyExportPolicy Exportable `

-HashAlgorithm sha256 -KeyLength 2048 `

-CertStoreLocation "Cert:\CurrentUser\My" `

-Signer $cert -TextExtension @("2.5.29.37={text}1.3.6.1.5.5.7.3.2")

4- Exportar Certificados

Windows 10 -> start -> procurar por certificado -> Gerenciar Certificados dos usuários -> Pessoal -> Exportar P2SChildCert & P2SRoot Cert da data atual:

* Right Click -> Todas as Tarefas -> Exportar
* No, do not export the private key
* Base-64 encoded X.509 (.CER)

5- Configurar Point-to-site: Pode **Demorar uns 10 minutos**

Azure -> Virtual Network Gatways -> Selecionar gateway criado -> Point-to-site configuration -> configure now

* IP Address: 172.16.201.0/24
* Root Certificates Name: P2SRootCert
* Public Certificate Date: Copiar do Root Cert exportado sem -----Start of cerificate --- e end of ceritifcate
* Save -> Download VPN Client

6- Conectar no VPN:

* Executar Instalador -> VPN -> virtualnetwork -> Conectar

### S2S (Site to Site VPN)

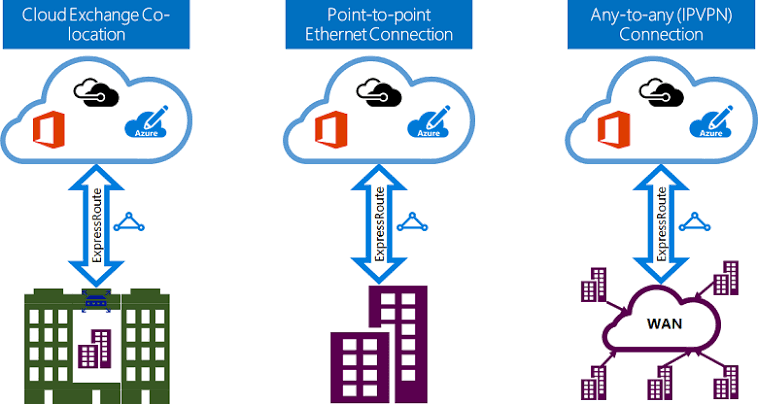
- Firewall do router faz conexão com VPN gateway do Azure Cloud utilizando IPSec para encriptar o trafego.

- Tem segurança mas sem garantia de banda (no horário de pico reduz muito).

### Express Route (Office + Cloud)

Este Tipo de VPN necessita de um ISP fornecendo um link dedicado para criar Express Route Circuit, onde os dados encriptados trafegam.

- Com isso temos tempo CiR (Committed information rate) – garantia da largura de banda.



# Load Balancer

- Trabalha na camada 4

<https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-overview#skus>

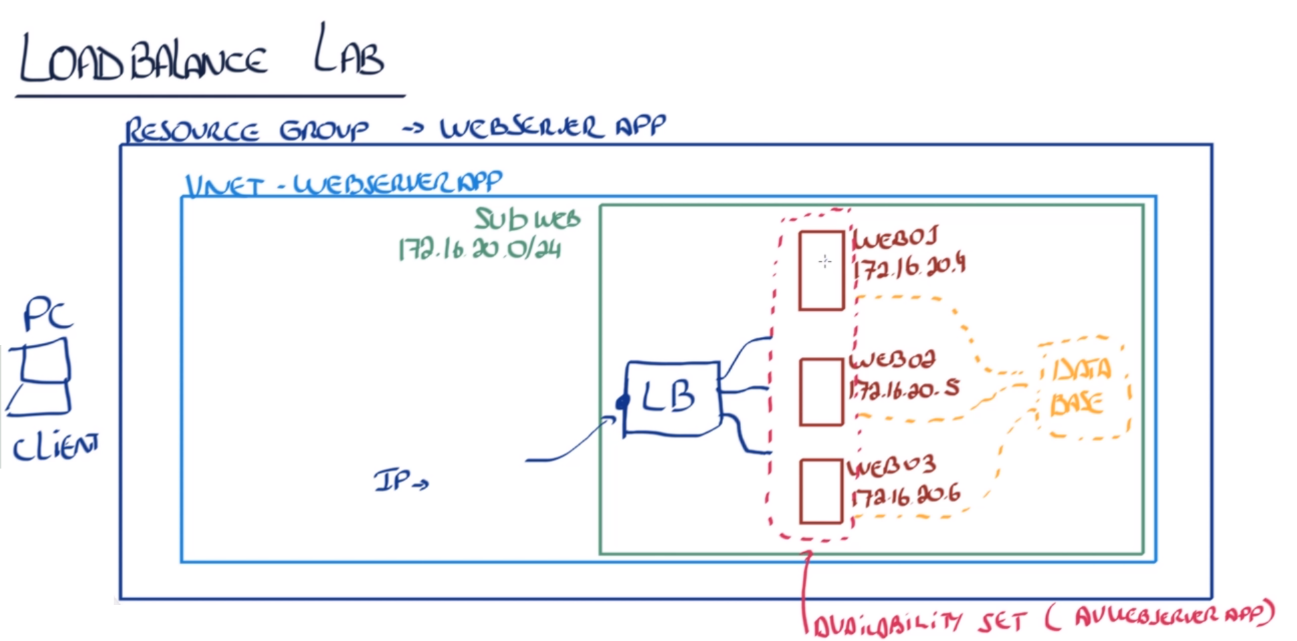
- Probes: monitorar cada VM e é configura timeout de cada VM para detectar quando ela cai.

- Auto-Reconfiguration: O load balancer autoconfigura caso for adicionar mais uma VM para distribuir o trafego.

- Hash based distribution: o trafego é distribuído e as sessões são mantida via 5 Tuple hash.

* Source IP
* destination IP
* Source Port
* Destination Port
* Protocol

- Load balancer tem public e private IP



1. Criar Resource Group
2. Criar VNet
3. Criar Máquina virtual com Availability zone e Cloud init script
4. Criar mais outras 2 Máquinas virtuais com o mesmo availability zone (para criar Load balancer)
5. Load Balancer -> Add

* Region: Igual das Vms
* Type: Public (Internal se precisa balancar a carga interna, ex: DBs servers)
* SKU: Basic (free, Até 100 instâncias, TCP & HTTP Only, Vms in single availability zone)
* Public IP address: Create New
* Assignment: Dynamic para teste

1. Frontend IP configuration -> Verificar se o IP público foi aplicado
2. Load Balancers -> Selecionar LB -> backend pool (Vms)-> Add ->

* Associar ao availability set (ou virtual machine scale)
* Add a target network IP Configuration -> Adicionar os VMs com seus IPS

1. Health Probs (Monitorar os VMs e tirar de backend pools se estiver offline) -> Add

* Name: HP-webserverapp
* Protocol: HTTP
* Port: 80
* Path: /
* Interval: 5 seconds
* Unhealthy threshold: 2

1. Load balancing rules -> Add

* Name: LBR-webserveapp
* IP Version: IPV4
* Frontend Ip address: Selecionar IP do LoadBalancer
* Protocol: TCP
* Port 80
* Backend port: 80
* Backend pool: Selecionar Pool do LB criada
* Health probe: Selecionar Health probe LB criada
* Session persistence: Client IP
* Timeout: manter 4
* Floating IP: Disabled (Enable só se estiver mysql e precisa direcionar)

# Active Directory

**Há versão free e versão paga**

- Normalmente a empresa tem 2 servidores. O primeiro é servidor de arquivos, o segundo é um servidor web, onde tem proxy e sistemas de gerenciamento da empresa.

## Azure AD (AAD)

* Active directory 100% Cloud.
* Ele é integrado com Office365,
* pode syncronizar AD Cloud com AD server da empresa (on premise).

## Active Directory Domain Service (ADDS)

* Servidor Active directory local da empresa. (Usa Kerberol e LDAP)

## Azure Active Directory Domain Service (AADDS)

* Serviço Active Directory gerenciado pela Microsoft (ready to use)
* A Microsoft configura as políticas do Azure Active Directory dependendo dos tickets que abriram

## Criar e Configurar Active Directory

1. Criação AD Free: Azure Active Directory -> Create a Directory -> Definir Name. Domain name e region -> Create
2. Azure Active Directory -> Switch directory -> All Directories -> Selecionar Directory Criado

### Adicionar Groups e Usuários

1- Criar Grupo: AD -> Groups -> +New Group:

* Group type: Security (Office 365 = AD + Email + Office365)
* Groupname: Sales
* Group descriptions: users of sales
* Create

2- Criar Usuário: AD -> Users -> +New User:

* Create User
* User name: João.silva
* Name, FirstName, LastName
* Groups: Selecionar Group
* Block sign in: No
* Usage Location: Brazil
* Create

3- Visualizar Usuário Clicando no nome dele onde podemos verificar login de login, dispositivos, aplicações, licenças para usuário e Login method

### Multi Factor Authentication (MFA)

- Aplicado a usuários para autenticar usando Senha e code SMS ou App para gerar código (Melhor).

<https://play.google.com/store/apps/details?id=com.azure.authenticator&hl=pt_BR>

AD -> Users -> Selecionar Usuário -> Multifactor authentication -> Selecionar usuário Enabled Multifactor Authentication

Enforce somente se precisa da senha para acessar Apps da Office 365 e Lync

### AD Self Service Password (Gerador de links)

**Serviço Pago – Premuim 2**

- Quando o usuário esquecer a sua senha é feito reset apartir SMS, App, Email... (não há necessidade de administrador resetar)

AD -> Password Reset -> Self service password -> All (Ou Alguns grupos) -> Authentication Methods-> Email, MobileSMS e Mobile app Code -> Number of Dates: 180 -> Notify users: yes -> Notify admin: yes

Link para reset: [https://passwordreset.microsoftonline.com/](https://passwordreset.microsoftonline.com/n/passwordreset?experience=WIN10_TB#!/)

## PIM (Privilaged Indentity Management)

**Requere Licencia P2 Licence**

- Configurar Privilegios dos usuários cadastrados, Liberar/Bloquear Acesso a:

* Recursos
* AD
* Acesso Administrativo por período de horas
* Visualizar log de atividade de cada usuário
* Criar Alertas quando um usuário ou grupo fazer uma ação
* Configurar Workflow para administrador aprovar uma requisição por exemplo de criar VM para um administrador aprovar

- Tipos de Roles:

* Directory Roles

Colocar usuário como administrador local ou admin de um recurso

* Resource Roles

Roles associando o usuário a uma subscription, limitar o uso de recursos como VM até N2, RBAC..

### Atribuir Permissão Azure para um usuário AD

Azure -> Azure Active Directory -> Users -> Selecionar usuário -> Directory Role -> Global Administrator

### Atribuir PIM para um usuário AD

Azure -> Azure Active Directory -> Users -> Selecionar usuário -> Directory Role -> Global Administrator

Azure -> Azure Active Directory -> Users -> Selecionar usuário -> Licenses -> +Assign -> Selecionar Licencia P2 Premium

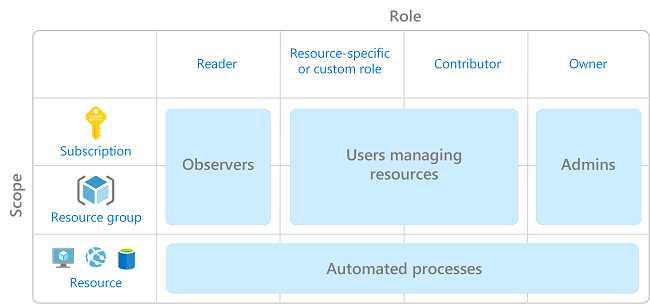
Azure -> AD PIM -> Consent to PIM -> Consent -> Esperar 2 minutos

AD PIM -> My Roles -> Sign up AD Roles -> Sign up

## RBAC (Role Based Access Control)

- Atribuir Permissões e recursos a usuários e grupos. **O usuário pode ser administrador, gerenciador de recursos ou observador (Read-only).**

**- Custom roles pode limitar regiões para recursos e etc..**



1- Azure -> Groups -> +New Group:

* Type: Security
* Name: WebAppMonitoring
* Membership type: Assigned
* Create

2- Atribuir o grupo a um resource Group: Selecionar Resource Group -> Access Control (IAM) -> Add -> Role Assignment:

* Role: Reader
* Assign access to: Azure AD user, group
* Selecionar Grupo criado

### Policies

Policies servem para filtrar as ações do usuário. Como por exemplo região onde os usuários podem criar a VM:

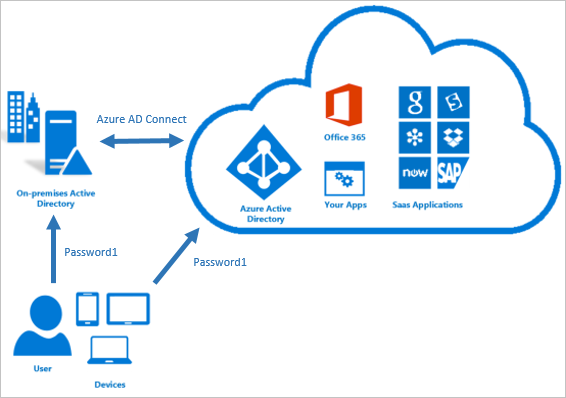
1. Resouce Group -> Selecionar RG -> Policies -> Definitions -> Search: Location -> Slecionar: Allowed Locations -> Assign

* Scope: Selecionar Subscription e RG
* Exclusions: (Se precisa excluir recurso da policy)
* Assignment name: Allowed locations East US
* Description: Somente East US
* Assigned By: Motasem El Malat
* Allowed Locations: Selecionar locais (EastUs e South America BR)
* Assign

Agora quando o usuário tenta criar recurso dentro do RG não vai conseguir

## Migração e Sincronização

### AD Connect Sync

Quando há um Ative Directory Local e necessita migrar tudo para Azure Active Directory Cloud e fazer espelho um a outro. É um sistema que copiar todas as configurações policies, password hash sincronization, password writepack, Device writeback, prevent acidental delete (se deletar em um lado não deletar em outro, tem que deletar nos 2 lados)

**- COM INTEGRAÇÃO AD CONNECT POSSIBILITA CRIAR USUÁRIO NO CLOUD OU NO LOCAL QUE SERÁ REGISTRADO LOCALMENTE E CLOUD DENRTO DO CICLO DE SINCRONIZAÇÃO (+- 5 MINUTOS)**

- Com AD Connect Sync podemos fazer migração ou deixar AD local e Cloud funcionando

#### 1- Criar Servidor AD Local

**IMPORTANTE: SEMPRE CRIAR UM NOVO SERVIDOR AD PARA NÃO OCORRER O RISCO DE SOBRE ESCREVER AS INFORMAÇÕES DO AD LOCAL E PERDER TUDO.**

1. Server Manager -> Add Roles and Features -> Role based or feature based -> Marcar: Active directory domain service, DNS Server -> Install
2. Promote this server to a domain controller -> add new forest -> mathbox.local -> Level (Windows Server 2016) -> Definir Senha de AD -> NetBOIS domain name: MATHBOX -> Install

#### 2- Sincronizar o servidor local com o Cloud AD

1. No Servidor local Abrir portal do Azure -> Switch Directory para diretório de AD -> Azure Active Directory-> Azure AD Connect -> Download Azure AD Connect -> Download
2. Use express Settings > Inserir login de Admin no AD Cloud -> Inserir Login de Admin do AD Local (Dominio\usuário e senha) -> Install

#### 3- Verificar a Sincronização

Demora uns 5 minutos para aparece o status

1. Azure -> Azure Active Directory -> Azure AD Connect -> Verificar o status

### Azure Connect Health

Serve para monitorar a sincronização entre o servidor cloud e local. Também podemos configurar a notificação em caso de erros ou conflitos durante a sincronização.

- Para ativar o recurso devemos instalar no servidor local o AD Connect Heath:

Azure -> Azure Active Directory -> Azure AD Connect -> Health and Analytics -> Download Azure Ad Connect for AD DS

Abrir Instalador -> Verificar as regras do instalador:

* Inbound e Outbound do firewall permite o trafego e as portas do AD
* Server Manager -> Local Server -> IE Enhanced Security Configuration -> Mudar para OFF

Configure Now -> Adicionar Credenciais de usuário admin no AD Cloud -> Autenticação via SMS -> Configurar Autenticação do Email do Azure

Azure Active Directory -> AD Connect -> Azure AD Connect Health -> AD DS Services -> Clicar no Serviço -> Visualizar informações

# Web App Service

- O App Service serve para ter infraestrutura automática onde todo o gerenciamento é feito pela Azure. Como por exemplo: configuração do domínio, VM set, DBs, Load Balancers, Availability zones....etc

- Usos:

* Web Apps: hospedagem de websites
* Mobile Apps: Backends
* Logic Apps: Automação (Tarefas automática orientadas a eventos)
* API Apps: Interfaces para integração de sistemas

- Criação:

* Name: definir nome do domínio
* Publish: Code
* Runetime stack: selecionar linguagem e versão
* Operating System: Linux (mais barato)
* Region: South Brasil (Mais perto)
* SKU and size: B1

## Deployment Slot

É usado para fazer testes em slot, cada slot gera um URL para testar o website antes de subir para produção.

**- Este recurso só é disponível em para planos de produção.**

- Criação: App service -> Deployment slots -> Add slot -> Clone settings from production

### Swap

É a troca de conteúdo entre a versão de teste e a versão de produção. Isso é feito depois que verificar a versão de testes e querer lançar para versão de produção.

Swap: App Service -> Deployment Slots -> Swap -> Confirmar source e target -> Swap

# Serverless

- Subir serviços e integrar sistemas sem configurar servidores. Onde sobe Funções, logic apps e eventos.

## Vantagens

* O uso de recursos é bem dinâmico até mais do que autoscale, onde é monitorado o número de acesso
* Pagamento por transação/acesso (Tempo de execução).

## Function App Nodejs

Hospedagem de website e backend: C, Java, Javascript. Pagamento por transação/acesso (Tempo de execução).

- Features:

* **URL para acessar website/backend**
* **SSL**
* Custom domains
* Scaling (App Service plan)
* Logic Apps
* **Console**
* **Advanced tools (Kudu)**
* **Push notification**
* **Autenticação para entrar na página web (Facebook, Microsoft Google, Twitter, AD)**
* CORS
* Process Explorer
* Log steaming

### Criação

* Novo Resource group
* OS: Windows (mais recursos)
* Hosting Plan: Consumption Plan
* Location: South America (Mais perto)
* Runtime stack: Nodejs
* Storage: Create new
* Nome Storage
* Application insights(monitorar performance e logs): disabled (pago)

### Http Trigger Functions

- requisição http GET/POST

1. Selecionar Function App -> Functions -> +Add -> Http trigger -> Athorization level: Function
2. Selecionar httpTrigger criada -> Editar Arquivo index.js
3. Selecionar httpTrigger criada -> Integrate Selecionar HTTP Methods

## Logic Apps - Automation

**Verificar o price primeiro antes de usar (diferencia de preço entre lugar e outro)**

Automation workflow (listener de eventos para executar ação)

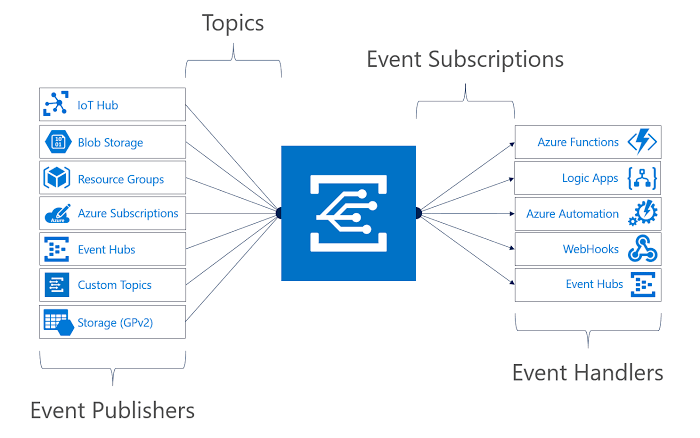
* Http Requests e Http Response
* Email received outlook
* Tweet Posted
* Arquivo adicionado no FTP Server
* Arquivo adicionado no One drive
* Receber IDOCs de SAP
* Postar mensagens no Slack
* Compartilhar Tweets no Facebook
* Quando novo arquivo adicionado no dropbox salvar no OneDrive
* Quando Receber twitter com uma palavra chave mandar no email

1. Criação: Azure -> Logic App -> +Add -> Log Analytics: off
2. Selecionar Logic App -> Verificar se o que precisamos tem template, se não +blank logic app
3. Search BlobStorage -> Azure Blob Storage -> When a blob is added or modified (properities)
4. Connection Name: logicBlobAccount -> Seleciona storage -> create
5. Container: Selecionar -> number of blobs to return from the trigger: 1 -> check cada 1 minuto -> next step
6. Outlook ou Gmail -> Send email -> To: digital ou obter lista dinâmica -> preencher subject e body
7. Save and Run
8. Visualizar o design no Logica pp designer

## Event grid - Logging

Monitoramento de registros, disparar mensagens e demonstração log dos eventos. As aplicações monitoradas são os subscribers.

- Evento Subscriptions: WebHooks é um App Service que tem um página com logs e mensagens



1. Criar Event Grid Topics: Azure -> Event Grid Topics -> +Add

* Name: logusers
* Event Schema: Event Grid Schema -> Create

1. Criar Event Viewer (Gratuito):

* Abrir: <https://azure.microsoft.com/en-gb/resources/samples/azure-event-grid-viewer/>
* Deploy to Azure
* Selecionar Resouce Group
* Selecionar Site Name
* Host Plan Name: definir nome
* SKU: F1 (Free)

1. Copiar Url do Event Viewer (App Service)
2. Selecionar Event Topic Criado -> Create one:

* Name: log-users-app001
* Event Schema: Event Grid Schema
* Endpoint: Select an endpoint -> Colar URL do app Service/api/updates -> Confirm Selection
* Create

1. Logging: Powershell 6

<https://docs.microsoft.com/en-us/azure/event-grid/custom-event-quickstart-portal>

$resourceGroupName = <resource group name>

$topicName = <topic name>

$endpoint = (Get-AzEventGridTopic -ResourceGroupName $resourceGroupName -Name $topicName).Endpoint

$keys = Get-AzEventGridTopicKey -ResourceGroupName $resourceGroupName -Name $topicName

1. Copiar abaixo e colocar no powerShell

$eventID = Get-Random 99999

#Date format should be SortableDateTimePattern (ISO 8601)

$eventDate = Get-Date -Format s

#Construct body using Hashtable

$htbody = @{

id= $eventID

eventType="recordInserted"

subject="myapp/vehicles/motorcycles"

eventTime= $eventDate

data= @{

make="Ducati"

model="Monster"

}

dataVersion="1.0"

}

#Use ConvertTo-Json to convert event body from Hashtable to JSON Object

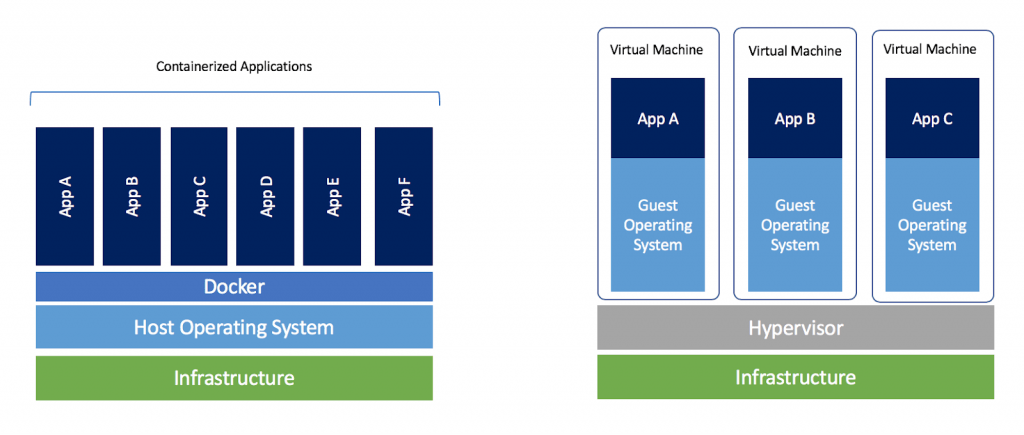
#Append square brackets to the converted JSON payload since they are expected in the event's JSON payload syntax

$body = "["+(ConvertTo-Json $htbody)+"]"

1. Enviar

Invoke-WebRequest -Uri $endpoint -Method POST -Body $body -Headers @{"aeg-sas-key" = $keys.Key1}

# Docker

- Virtualização é muito custosa, além do sistema operacional do servidor é instalado outros sistemas operacionais virtual que desperdiça muito processo. O docker utiliza o mesmo sistema operacional para criar containers e colocar a aplicação.

## Vantagens

* Portátil (Podemos Exportar imagem de todo sistema)
* Isolação
* Otimização (Muito rápido)
* Não hypervisor (diferente do VM)
* Container que tem ambiente separado que funciona em qualquer máquina