PV286 - Secure coding principles and practices

Cloud programming security

Lumir Honus

CRoCS
Centre for Research on Cryptography and Security

Agenda

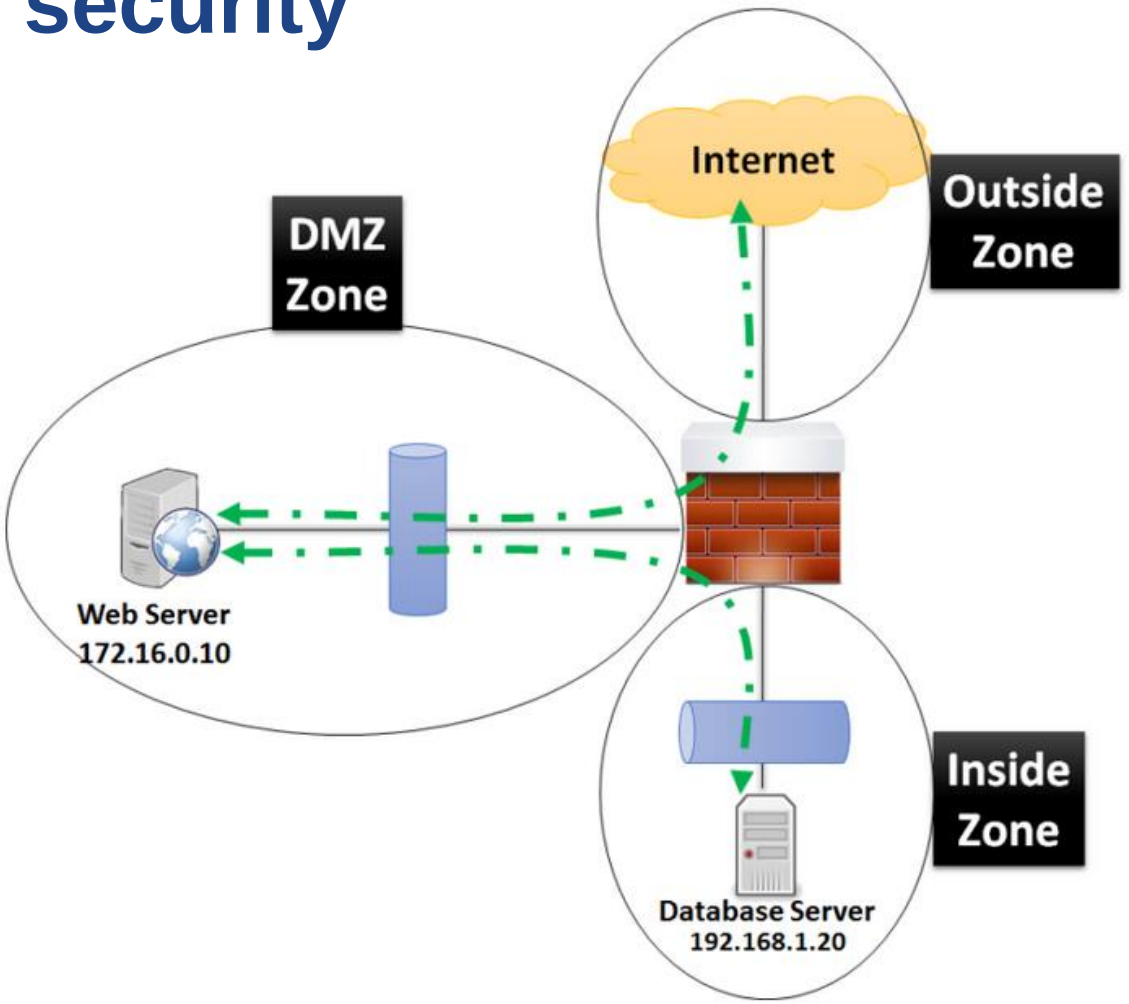
- Introduction
- Evolution of enterprise architecture
- Cloud fundamentals
- Storing secrets in the Keyvault
- Application authorization, how to pass secret to the app

What I did after I graduated:

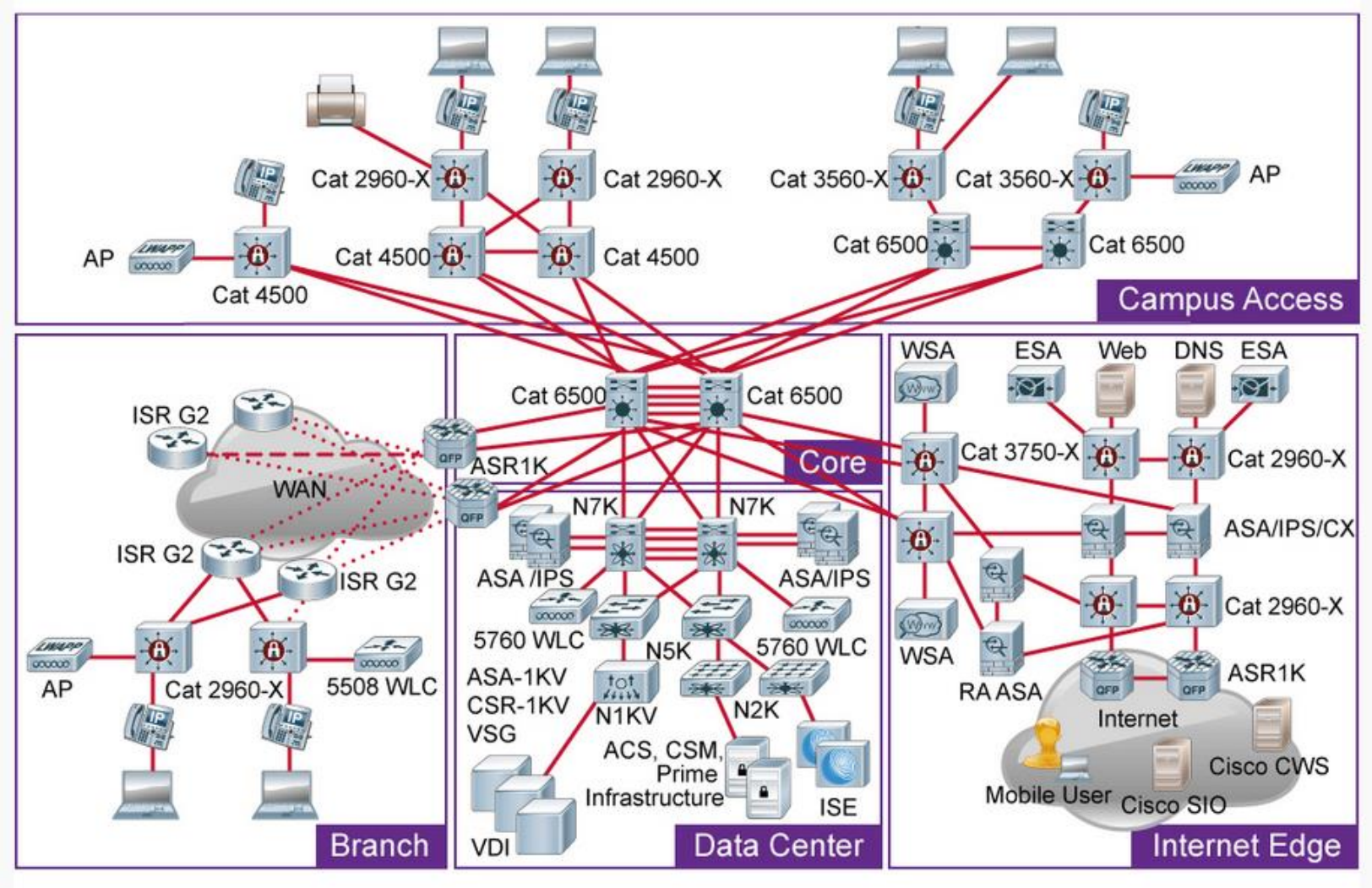
- Founded a software development startup company … and failed.
- Joined AT&T as Tier2 engineer … for a temporary job ☺
- Networking is interesting, passed CCIE certification, became network architect
- Designed complex enterprise datacenters for large financial institutions
- Founded AT&T Software Defined Datacenter offering
- Software Defined Datacenter evolved into Software Defined IT Infrastructure

Classical concept of enterprise security

- N-Tier application design
- Attackers are outside
- Premise based firewalls split security zones
- Workload is relatively static, must have “static” IP address to be defined in firewall policies



Cisco Enterprise reference network design (2015)



Google BeyondCorp (2014)

- Access to services must not be determined by the network from which you connect
- Access to services is granted based on contextual factors from the user and their device
- Access to services must be authenticated, authorized, and encrypted

Covid19 – and home office

- It works just fine, right ?

Thought:

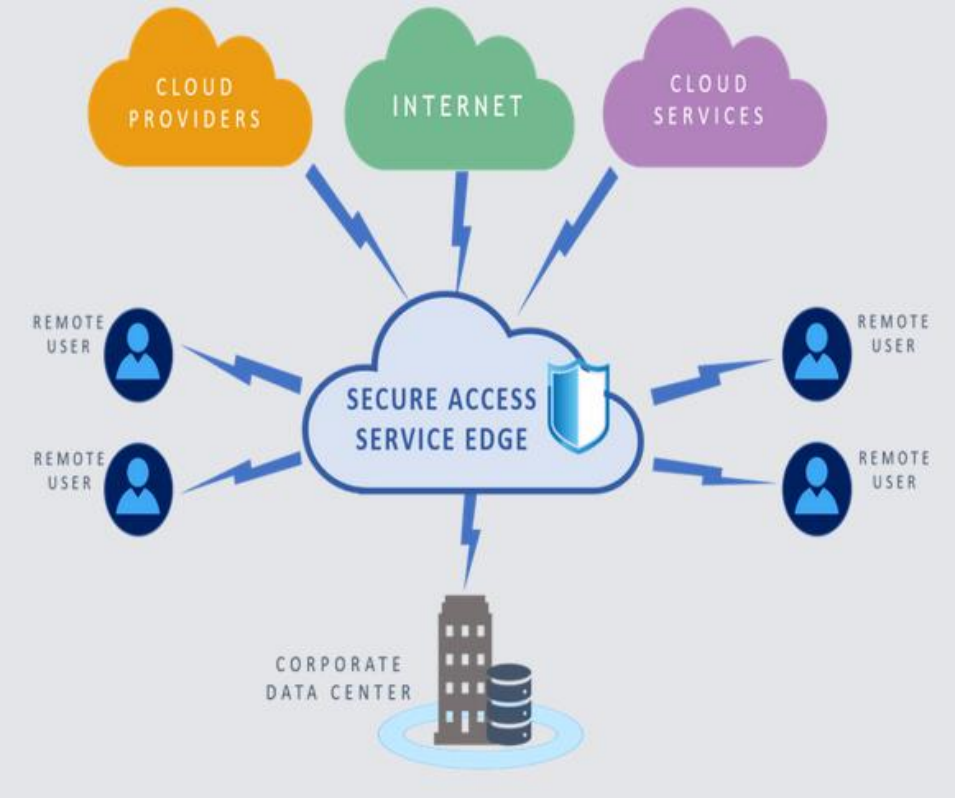
- If half of your workload is in the cloud and majority of people works from home, where is your perimeter ?

SASE – Secure access service edge

First described by Gartner

Convergence of:
- WAN
- Network security services
- Zero Trust environment

Into
- Cloud-delivered service model



Zero trust

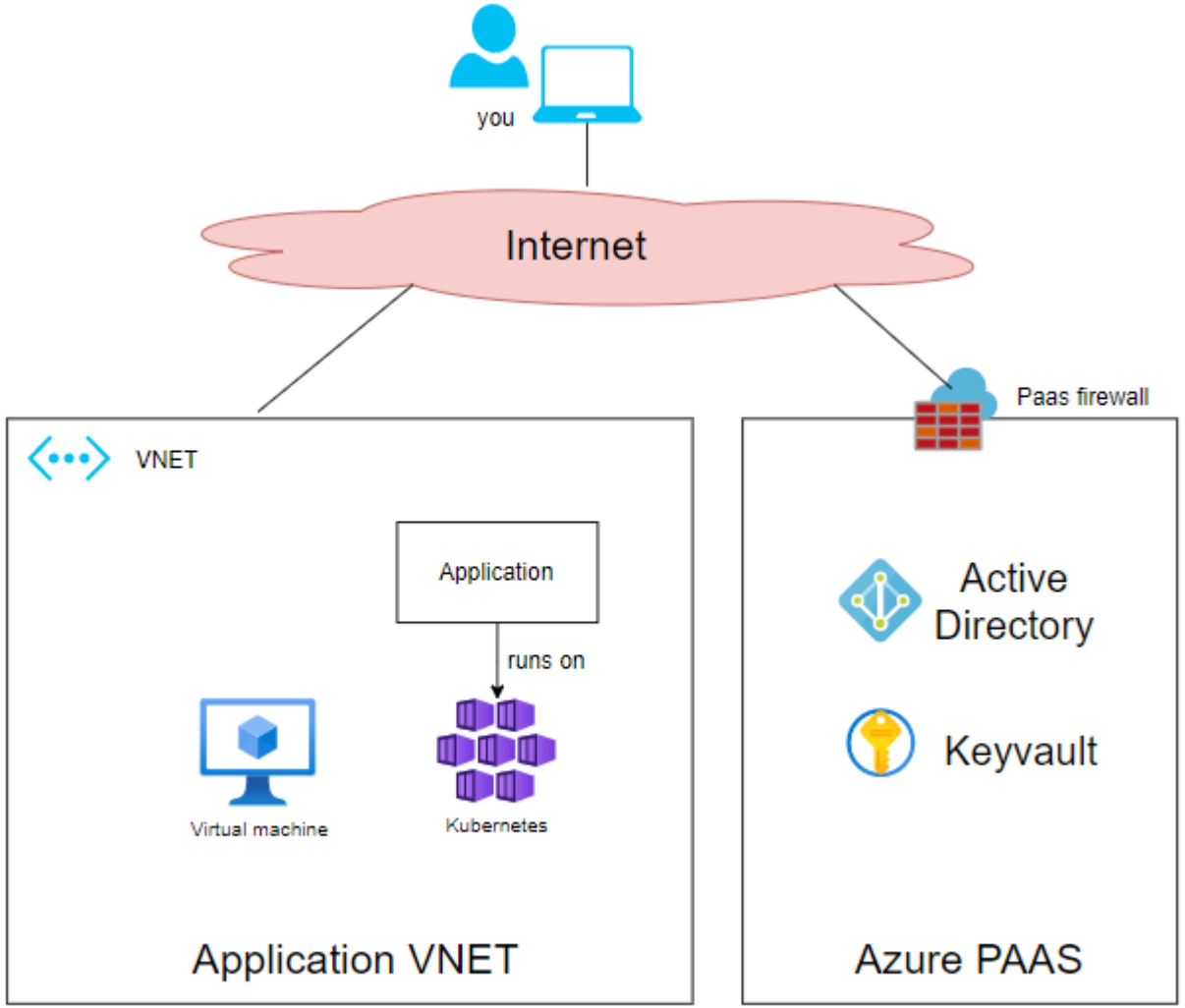
- **Verify explicitly** - Always authenticate and authorize based on all available data points.
- **Use least privilege access** - Limit user access with Just-In-Time and Just-Enough-Access (JIT/JEA), risk-based adaptive policies, and data protection.
- **Assume breach** - Minimize blast radius and segment access. Verify end-to-end encryption and use analytics to get visibility, drive threat detection, and improve defenses.

Everything I knew about network architecture is dead !

- The world has changed
- Technologies changes, but basic principles remains
- Cryptography is critical -- yet surprisingly few people understands it

Scenario

what we will be focusing on during this session and lab



(Azure) cloud fundamentals

Virtual Network (VNET)
- “Virtual routing domain”
- A logical isolation of the cloud dedicated to your subscription
- Contains one or more subnets

Subnet
- IP subnet - range of IP addresses in the virtual network
- Has routing table + network security group

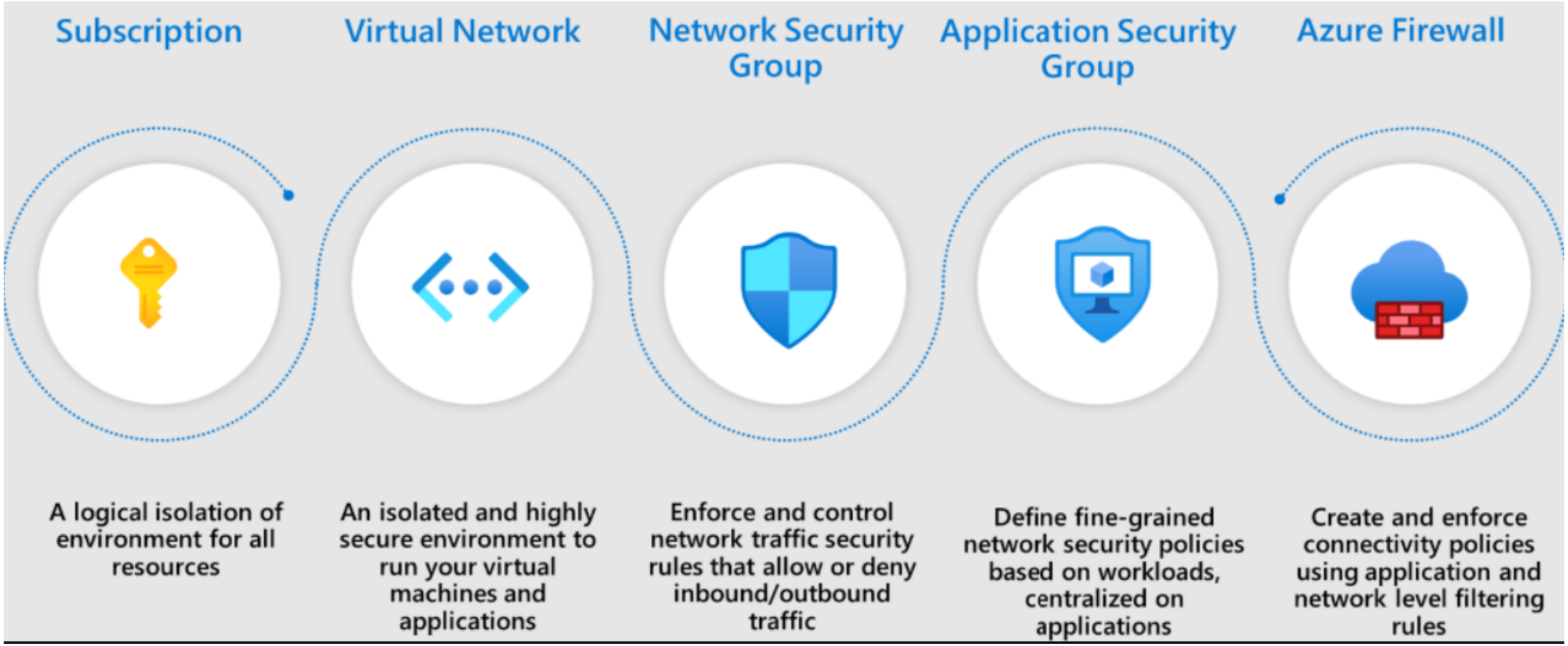
Network security group
- Applied typically on VNIC / VM level
- Works on layer4 only

Azure Firewall
- Works on Layer4 + Layer7
- Centralized policy, usually used as “north-south” security perimeter

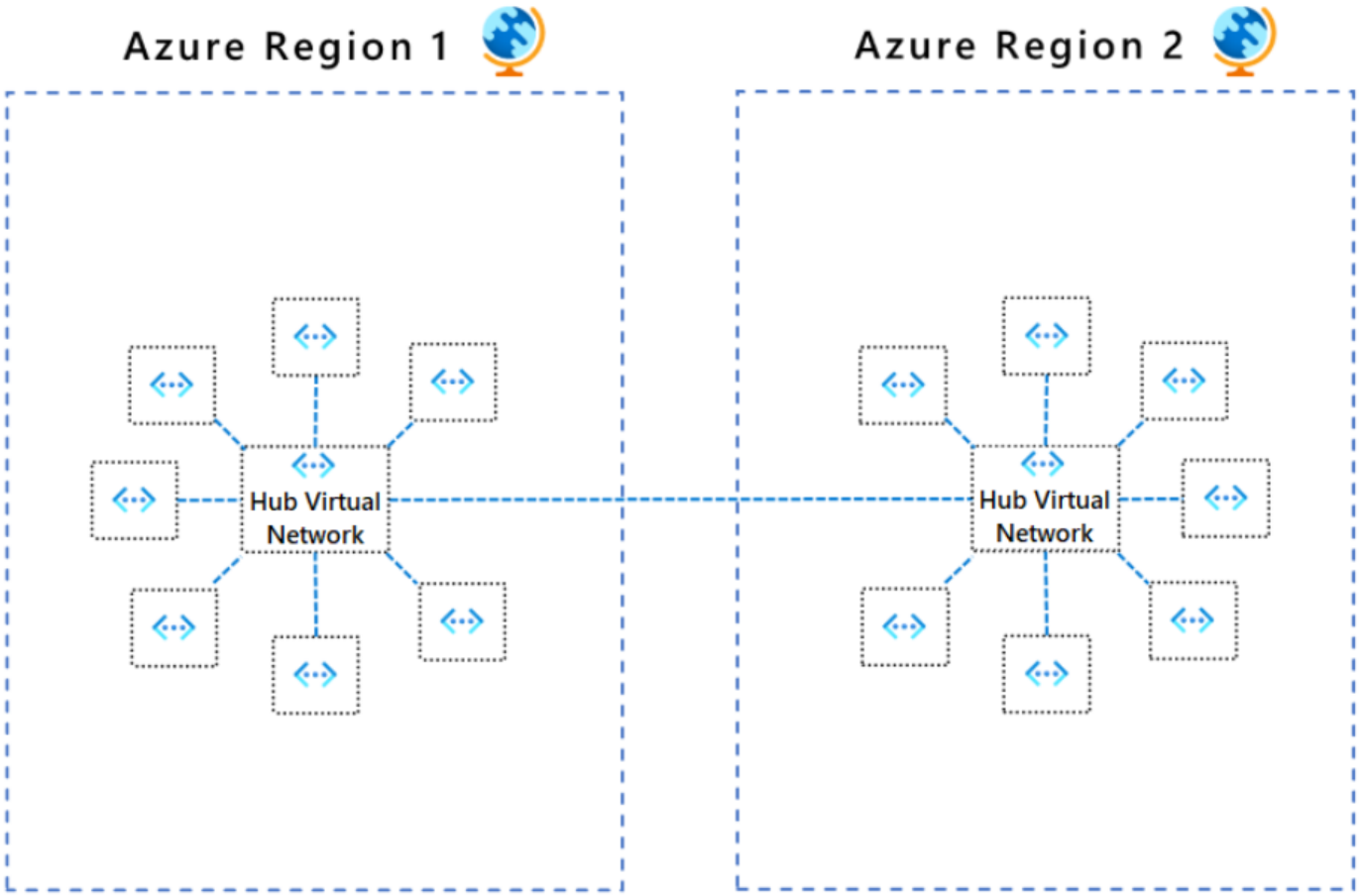
Cloud brings new security challenges

- Cloud has powerful tools to enforce security
- But you need to know what you are doing – you are one “wrong click” from being exposed on Internet
- Build automated patterns / playbooks
- Use defense in depth principle
- Use audit and compliance tools to validate that deployment is compliant with intended architecture

Azure Segmentation patterns



Hub and spoke pattern



Cloud comparison

Fabric Service	MS Azure	AWS
Data Centre HA	Region Pairs	Availability Zone
Tenant	Resource Group	EC Instance
Subnets	Subnets	Private/Public Net
VLAN	Subnet	Private/Public Net
Routing Domain	Virtual Network	VPC
Load Balancing	Network Load Balancer	Elastic LB
Firewall	Azure Firewall	Web App FW
DNS	DNS	Amazon Route 53
Cloud Connection	Azure ExpressRoute	AWS Direct Connect
Private VLAN	ASG	Security Group
Access Control List	NSG	Security Group Rule
Outbound ACL	Outbound Rule	Outbound Rule
Inbound ACL	Inbound Rule	Inbound Rule

VAULTs

What is Azure Keyvault

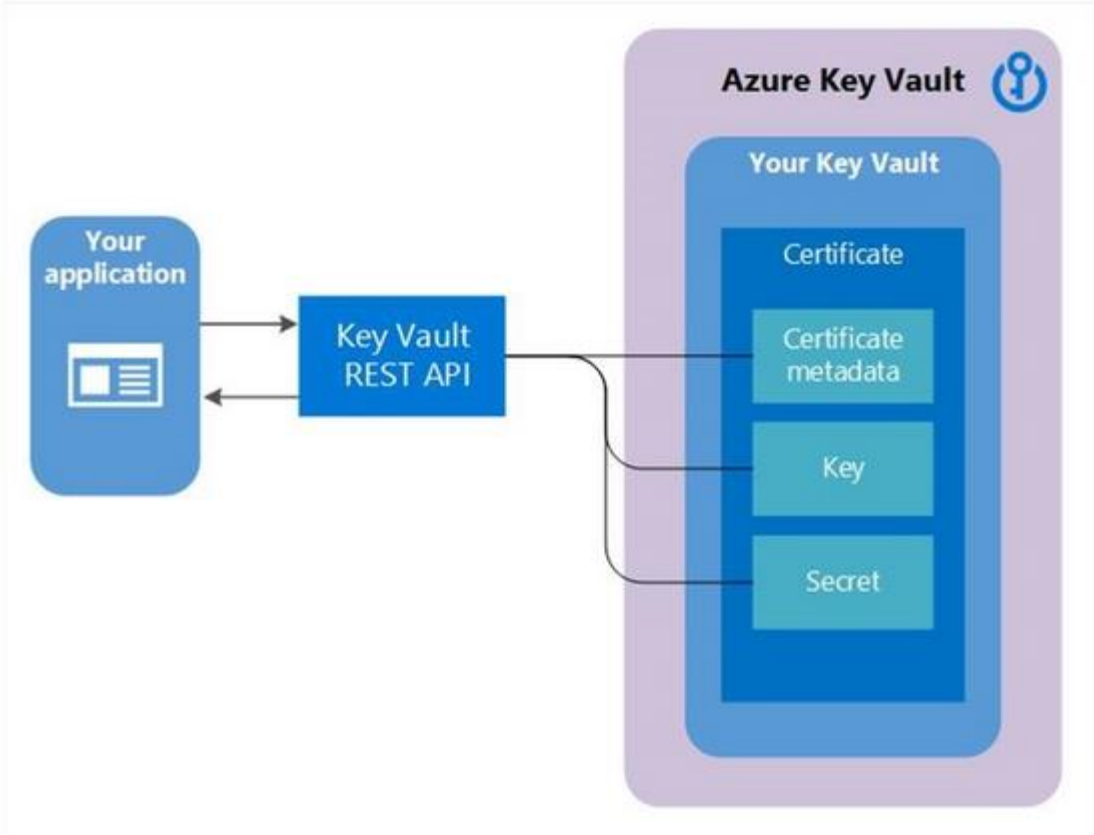
Cloud service that provides a secure store for secrets.

Securely store keys, passwords, certificates, and other secrets.

Provides API accessible via REST + OAUTH2

Granular Role Based Access control

Strong focus on audit – who can define and see the password



Microsoft vault types – same API:

- Azure Key Vault (Standard Tier): A FIPS 140-2 Level 1
- Azure Key Vault (Premium Tier): A FIPS 140-2 Level 2
- Azure Managed HSM: A FIPS 140-2 Level 3
- Azure Dedicated HSM: A FIPS 140-2 Level 3
- Azure Payments HSM: A FIPS 140-2 Level 3, PCI HSM v3

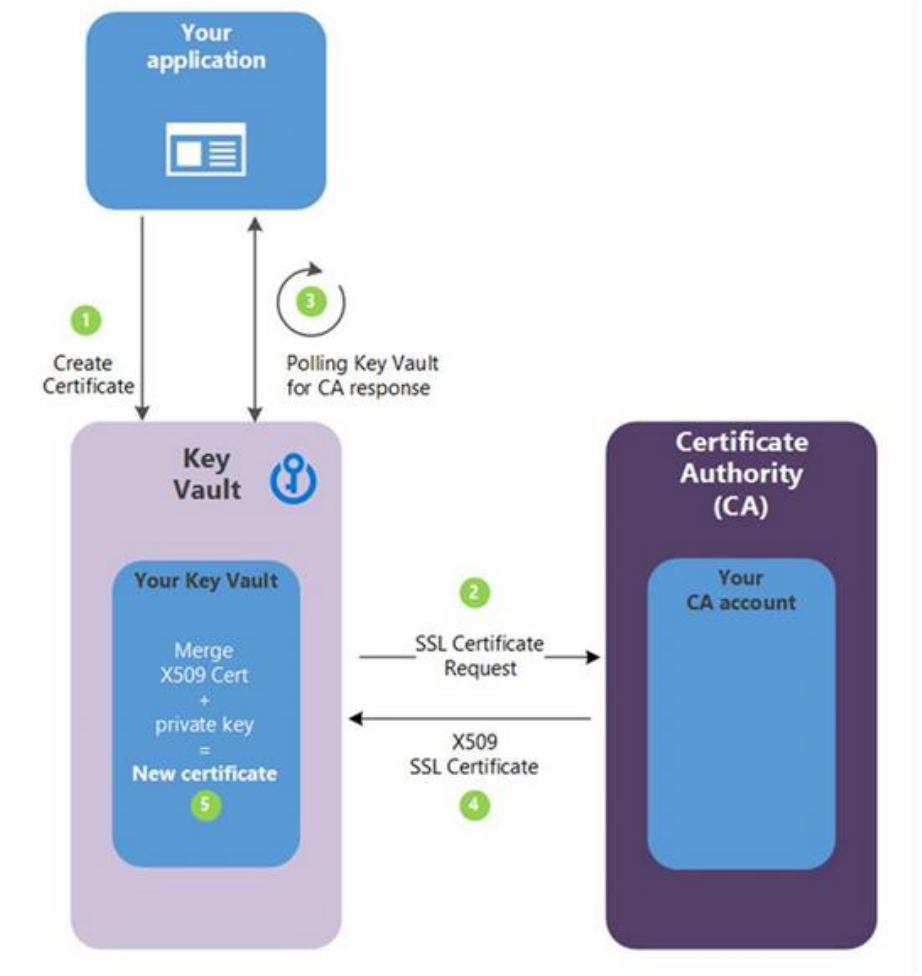
Azure Keyvault roles

Apply principle of least privillege

Built-in role	Description
Key Vault Administrator	Perform all data plane operations on a key vault and all objects in it, including certificates, keys, and secrets. Cannot manage key vault resources or manage role assignments. Only works for key vaults that use the 'Azure role-based access control' permission model.
Key Vault Certificates Officer	Perform any action on the certificates of a key vault, except manage permissions. Only works for key vaults that use the 'Azure role-based access control' permission model.
Key Vault Crypto Officer	Perform any action on the keys of a key vault, except manage permissions. Only works for key vaults that use the 'Azure role-based access control' permission model.
Key Vault Crypto Service Encryption User	Read metadata of keys and perform wrap/unwrap operations. Only works for key vaults that use the 'Azure role-based access control' permission model.
Key Vault Crypto User	Perform cryptographic operations using keys. Only works for key vaults that use the 'Azure role-based access control' permission model.
Key Vault Reader	Read metadata of key vaults and its certificates, keys, and secrets. Cannot read sensitive values such as secret contents or key material. Only works for key vaults that use the 'Azure role-based access control' permission model.
Key Vault Secrets Officer	Perform any action on the secrets of a key vault, except manage permissions. Only works for key vaults that use the 'Azure role-based access control' permission model.
Key Vault Secrets User	Read secret contents. Only works for key vaults that use the 'Azure role-based access control' permission model.

Keyvault for certificate management

(1) - Application is creating a certificate which internally begins by creating a key in your key vault.
(2) - Key Vault sends an TLS/SSL Certificate Request to the CA.
(3) - Your application polls, in a loop and wait process, for your Key Vault for certificate completion. The certificate creation is complete when Key Vault receives the CA's response with x509 certificate.
(4) - The CA responds to Key Vault's TLS/SSL Certificate Request with an X509 TLS/SSL Certificate.
(5) - Your new certificate creation completes with the merger of the X509 Certificate for the CA.



Many Vault offerings:

- Venafi
- Azure
- AWS
- Hashicorp
- Spring vault

What are the architecture decisions ?

Selecting Vault solution – architecture decisions:

- What are the regulatory requirements ? Do you need HSM ?
- How many vaults do you need ? One per company, per environment, per application ?
- What do you need to integrate with ?
- Buy or build (Cloud service or self-managed) ?
- What are the cost implications ?

Azure cost implications:

Vaults

Vaults are offered in two service tiers—standard and premium.

	Standard	Premium
Secrets operations	**$0.03**/10,000 transactions	**$0.03**/10,000 transactions
Certificate operations[1]	Renewals—**$3** per renewal request. All other operations—**$0.03**/10,000 transactions	Renewals—**$3** per renewal request. All other operations—**$0.03**/10,000 transactions
Managed Azure Storage account key rotation (in preview)	Free during preview. General availability price — **$1** per renewal[2]	Free during preview. General availability price — **$1** per renewal[2]

[1]Key Vault does not issue certificates or resell certificates from CAs. Key Vault provides the ability to simplify and automate certain tasks on certificates that you purchase from Public CAs, such as enroll and renew.

[2]Storage account keys are stored as 'secrets' in Key Vault, and therefore operations charges (see 'Secrets Operations' row above) will apply on any operation performed on these keys, including a renewal. See FAQs below for more details on how operations are defined.

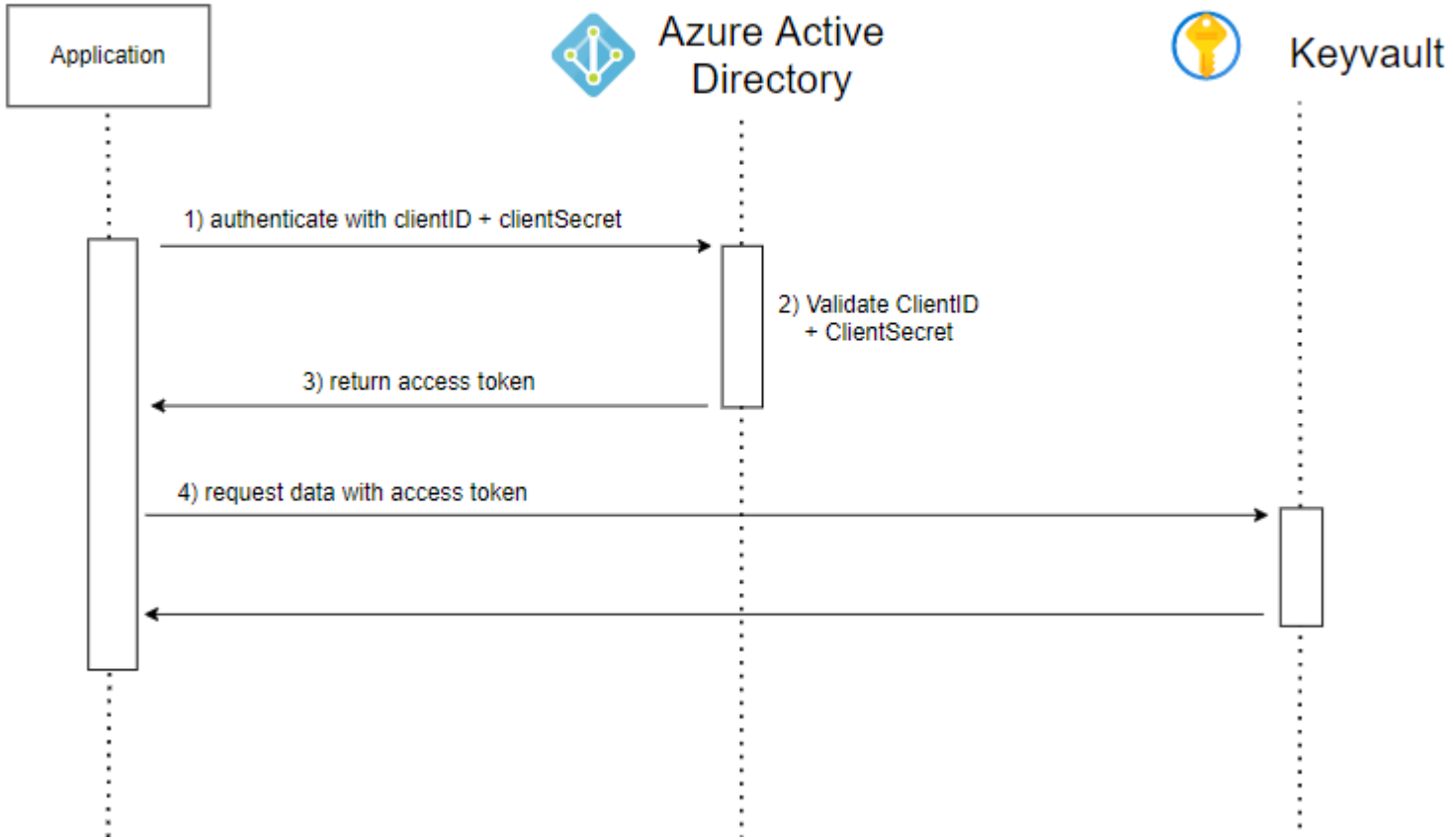
AAA
Authentication + Authorization + Accounting

Application

- Need to authenticate and authorize clients/users (not our primary focus in this seminar)
- Need to authenticate itself to against various backend APIs and resources (Keyvault, DB, etc…)
- How ?

Client credentials grant

- ClientID + Client secret
- Client ID + certificate



OAUTH2 + OpenID Connect

- Adopted by all major cloud service providers
- OpenID Connect extends OAuth 2.0 by providing user authentication and single sign-on
- Study:
  - https://oauth.net/2/

How to pass secrets to the application ?

1) Hardcode it in the code / image ☺
2) Hardcode it in the code / image, but encrypt by some “secret algorithm” ☺
3) Store it on the disk and inject into the application runtime
   - Is your disk encrypted in all your runtime environments ? Can you pass PCI DSS compliance audit ?
4) Pass it via ENV variables
   - Better – especially if you can limit it to the application process
5) Use “managed identity” solution

Never store secrets in GIT repositories

- Seems obvious but
- No matter how many times I say it, people still do it
- Update your .gitignore files to prevent secrets in the GIT repository
- Use automated SAST scanning to detect it
- Bounty hunters

Store secrets in docker image

Dockerfile

ENV secret=pass1435

COPY app.jar /

CMD [“java app.jar”]

Problem

Docker info….

Encrypt cipher

- Create library “encrypt.jar”
- Cipher.decrypt(…)

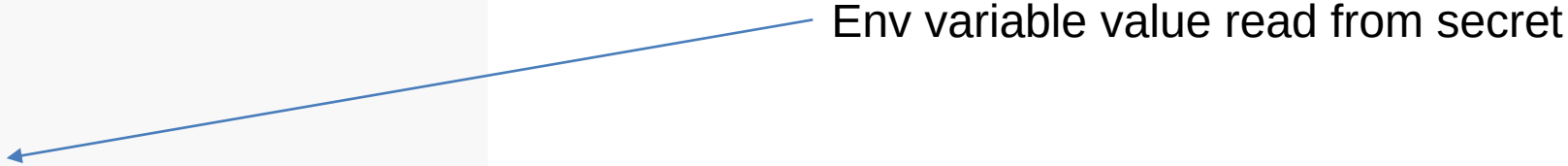
Problem:
-- library can be reverted
-- library is published in MAVEN

Mount secrets as a volume

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: mypod
spec:
  containers:
  - name: mypod
    image: redis
    volumeMounts:
    - name: foo
      mountPath: "/etc/foo"
      readOnly: true
  volumes:
  - name: foo
    secret:
      secretName: mysecret
      optional: false # default setting; "mysecret" must exist
```

Pass secrets as ENV variables

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: secret-env-pod
spec:
  containers:
  - name: mycontainer
    image: redis
    env:
      - name: SECRET_USERNAME
        valueFrom:
          secretKeyRef:
            name: mysecret
            key: username
            optional: false # same as default; "mysecret" must exist
                            # and include a key named "username"
      - name: SECRET_PASSWORD
        valueFrom:
          secretKeyRef:
            name: mysecret
            key: password
            optional: false # same as default; "mysecret" must exist
                            # and include a key named "password"
  restartPolicy: Never
```



Secrets encryption:

- Kubernetes Secrets are by default stored unencrypted in ETCD (Kubernetes API server database)

PCI DSS:

“The requirement to protect keys from disclosure and misuse applies to both data-encrypting keys and key-encrypting keys. Because one key-encrypting key may grant access to many data-encrypting keys, the key-encrypting keys require strong protection measures.”

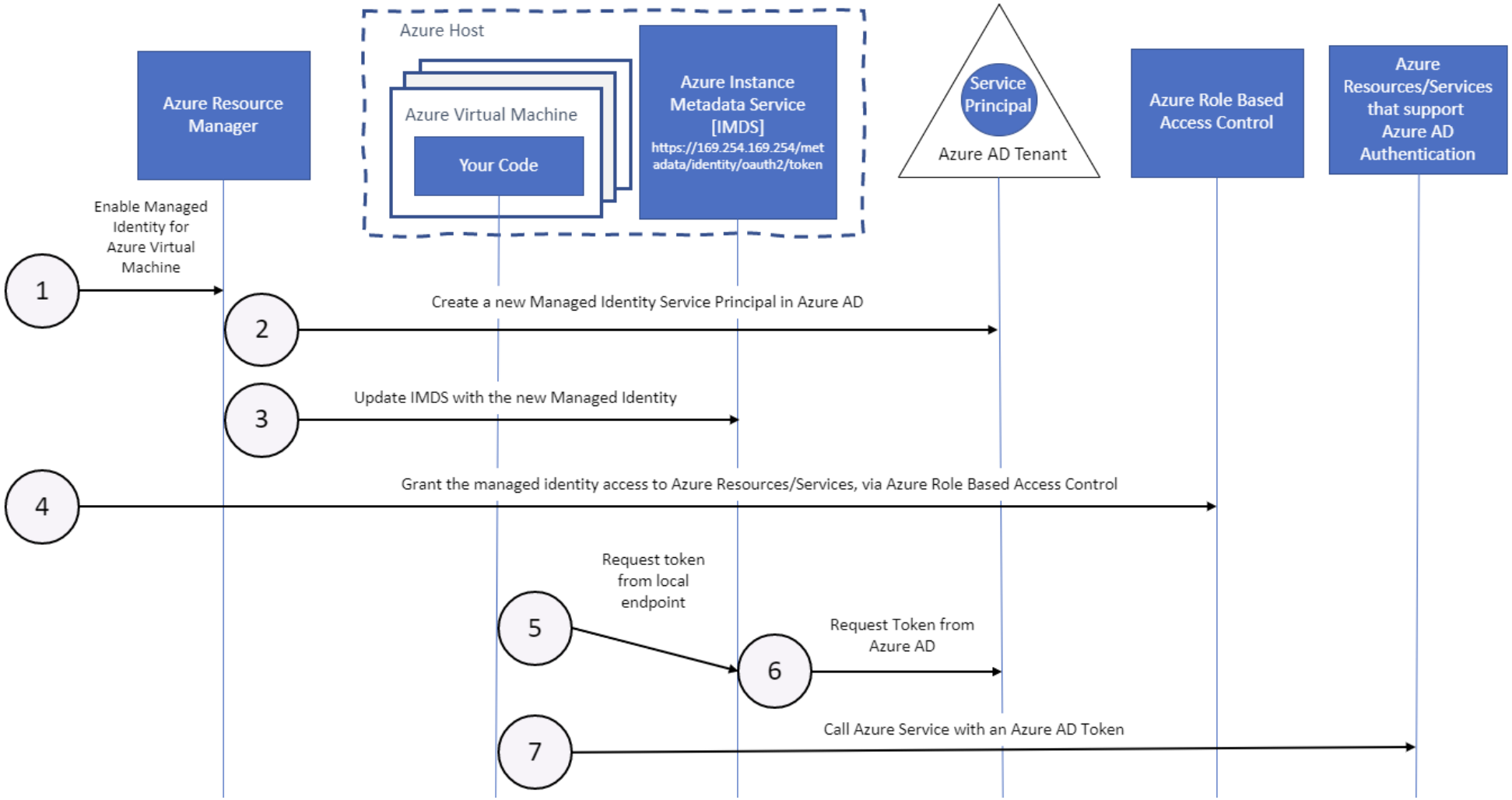
Azure managed identity

- Developers don’t need to manage credentials (credentials are not even accessible by users)

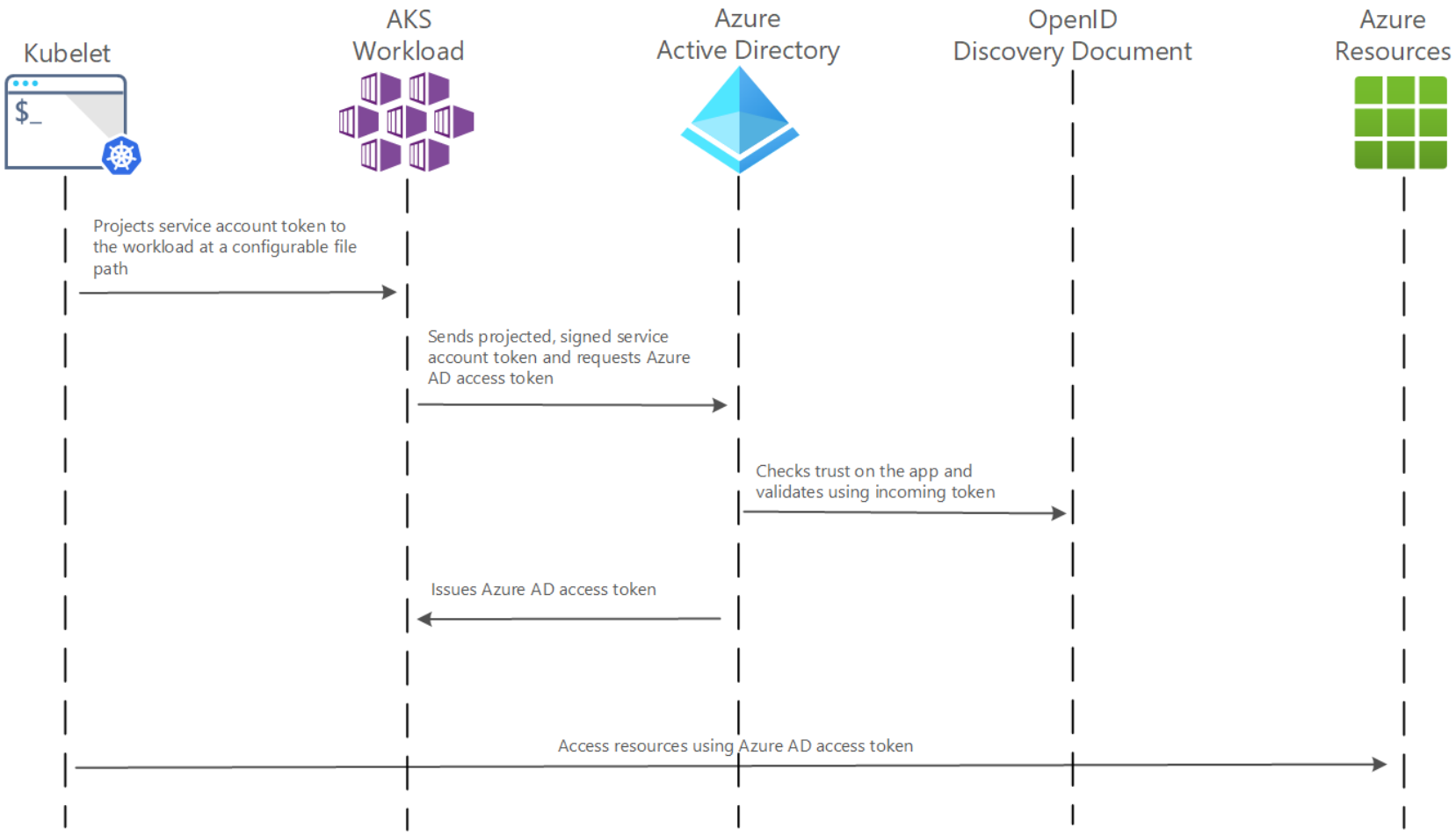
- **System-assigned**
  - For example Kubernetes (AKS)

- **User-assigned**
  - Standalone Azure resource

Managed identity -- How it works



Workload identity

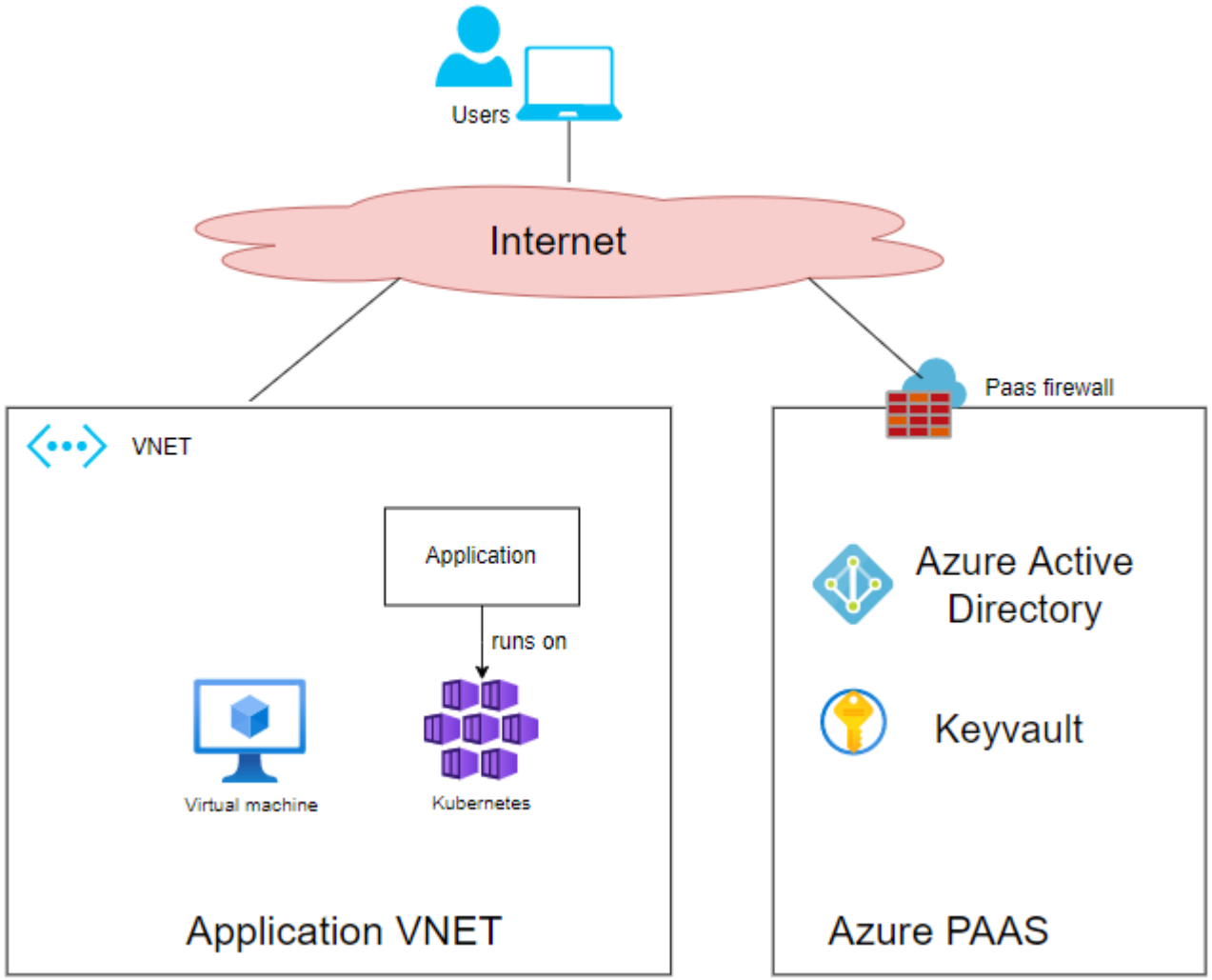


Infrastructure security in cloud

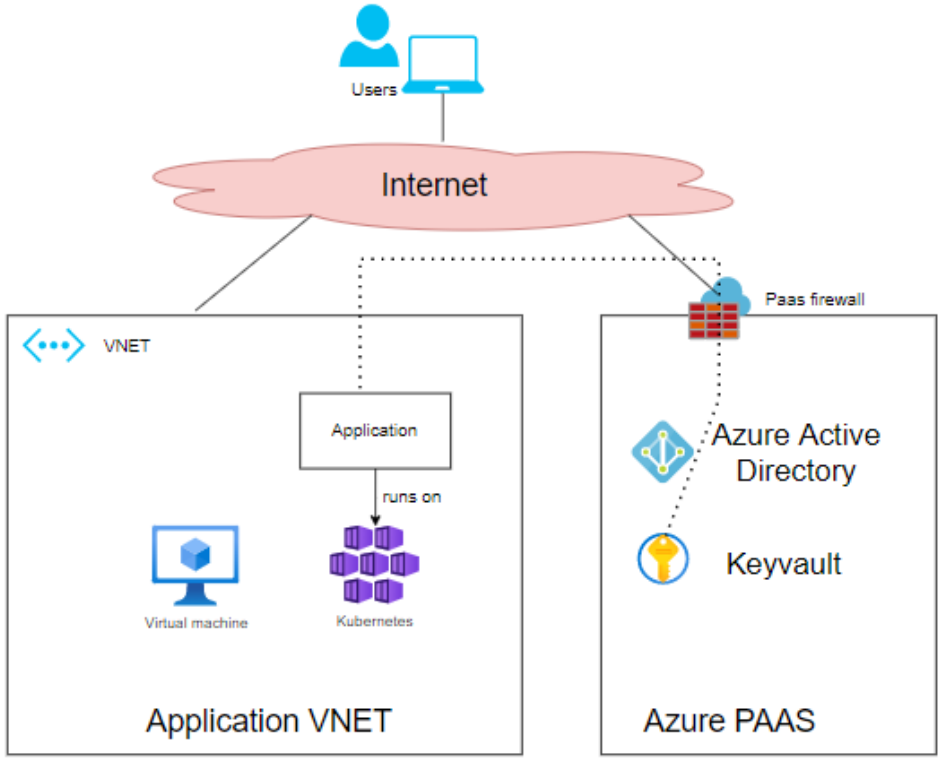
Design exercise

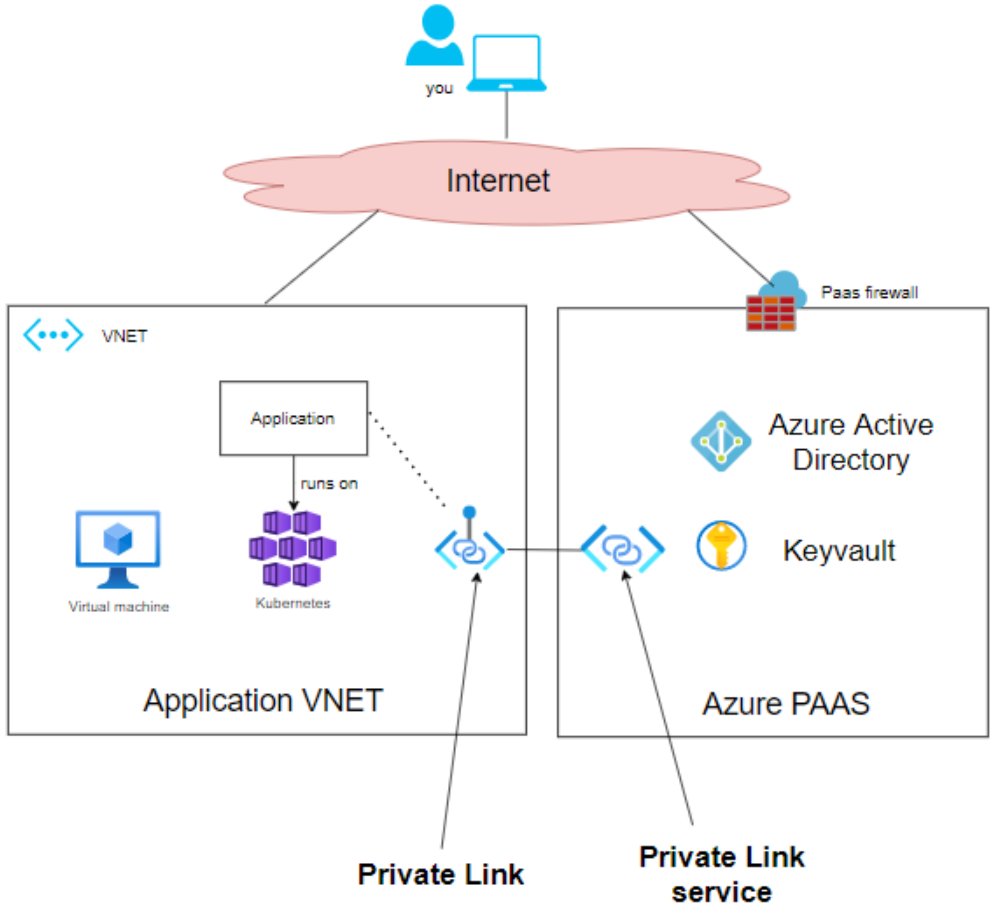
Let’s assume that application is an internal HR system

What can be improved

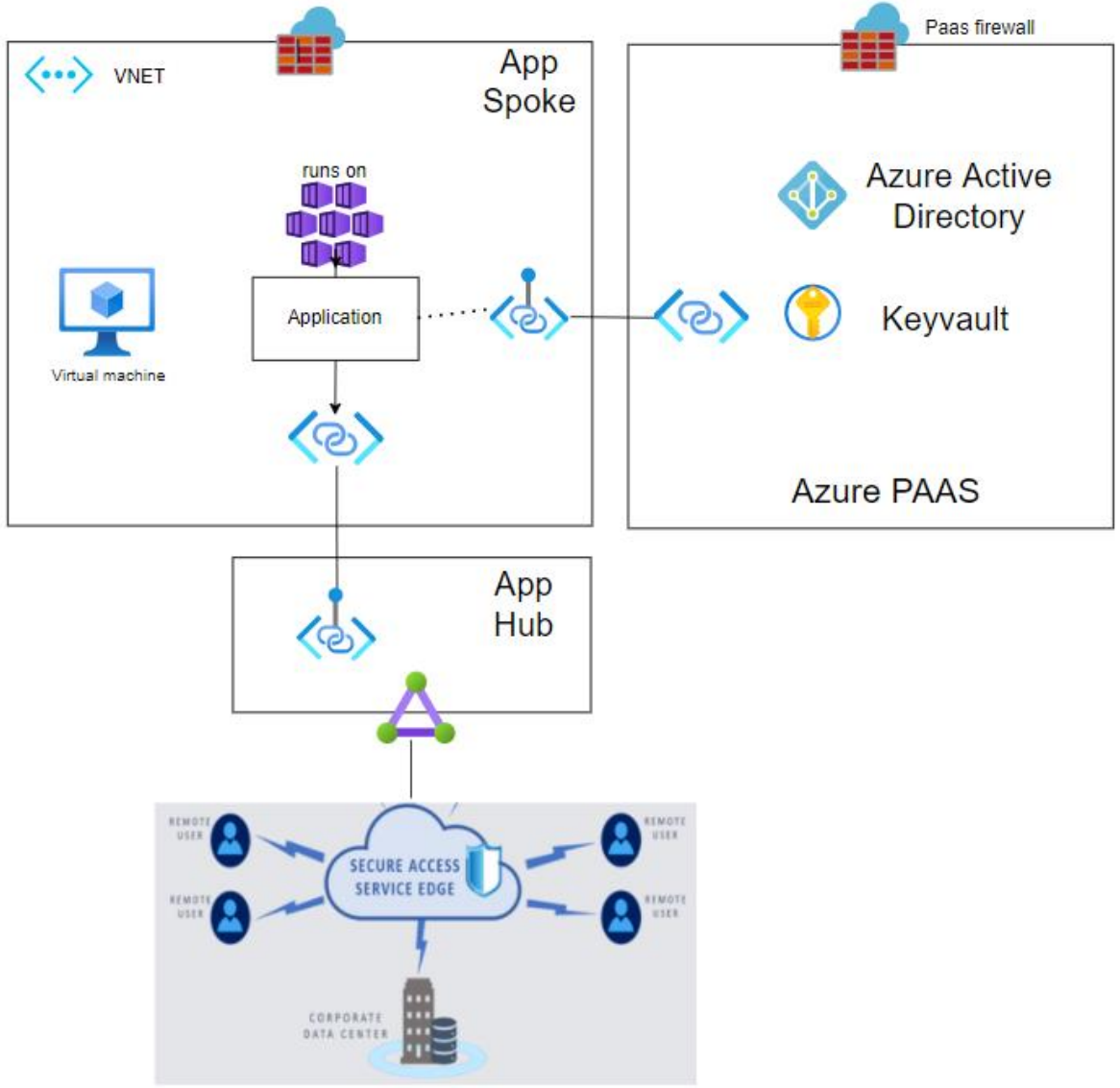


Private link





Typical design



Challenges with Kubernetes

- Kubernetes workload is often short-lived and has dynamic IP addresses
- By default, Kubernetes has open policy – any POD can communicate with any other PODs (even across namespaces)
- Many approaches how to solve it:
  - Ingress
  - Network policy (CNI)
  - Service Mesh

Cloud adoption in enterprises

If cloud is so cool, why not everybody is using it ?

Cloud transformation

92%

Of organization’s total IT environment is at least somewhat in the cloud today

54%

Mostly on-premise with some cloud

9%

Cloud only

Source: IDC Cloud Computing Study

Cloud Repatriation (analysis by Andreessen Horowitz)

The Cost of Cloud, a Trillion Dollar Paradox

https://a16z.com/2021/05/27/cost-of-cloud-paradox-market-cap-cloud-lifecycle-scale-growth-repatriation-optimization/?utm_source=thenewstack&utm_medium=website&utm_campaign=platform

- Dropbox detailed in its S-1 a whopping $75M in cumulative savings over the two years prior to IPO due to their infrastructure optimization overhaul, the majority of which entailed repatriating workloads from public cloud.
- Thomas Dullien, former Google engineer and co-founder of cloud computing optimization company Optimyze, estimates that repatriating $100M of annual public cloud spend can translate to roughly less than half that amount in all-in annual total cost of ownership (TCO) — from server racks, real estate, and cooling to network and engineering costs.
- Extending this analysis to the broader universe of scale public companies that stands to benefit from related savings, we estimate that the total impact is potentially greater than $500B.
- If you're operating at scale, the cost of cloud can at least double your infrastructure bill.

LABS

Make sure that you have personal account in the Azure

Use your education license or register via https://azure.microsoft.com/en-us/free (200$ free credit)

During preparation for the course, I spent 0.26 Eur for compute ☺