Contents

[Design a compute solution 2](#_Toc53187933)

[Virtual Machine 2](#_Toc53187934)

[Resource Locks 2](#_Toc53187935)

[JIT VM Access 2](#_Toc53187936)

[Container Deployment Options: 3](#_Toc53187937)

[Design a network solution 4](#_Toc53187938)

[Design an application architecture 4](#_Toc53187939)

[Design migrations 5](#_Toc53187940)

# Design a compute solution

* recommend a solution for compute provisioning
* determine appropriate compute technologies, including virtual machines, App Services, Service Fabric, Azure Functions, Windows Virtual Desktop, and containers
* recommend a solution for containers o AKS versus ACI and the configuration of each one
* recommend a solution for automating compute management

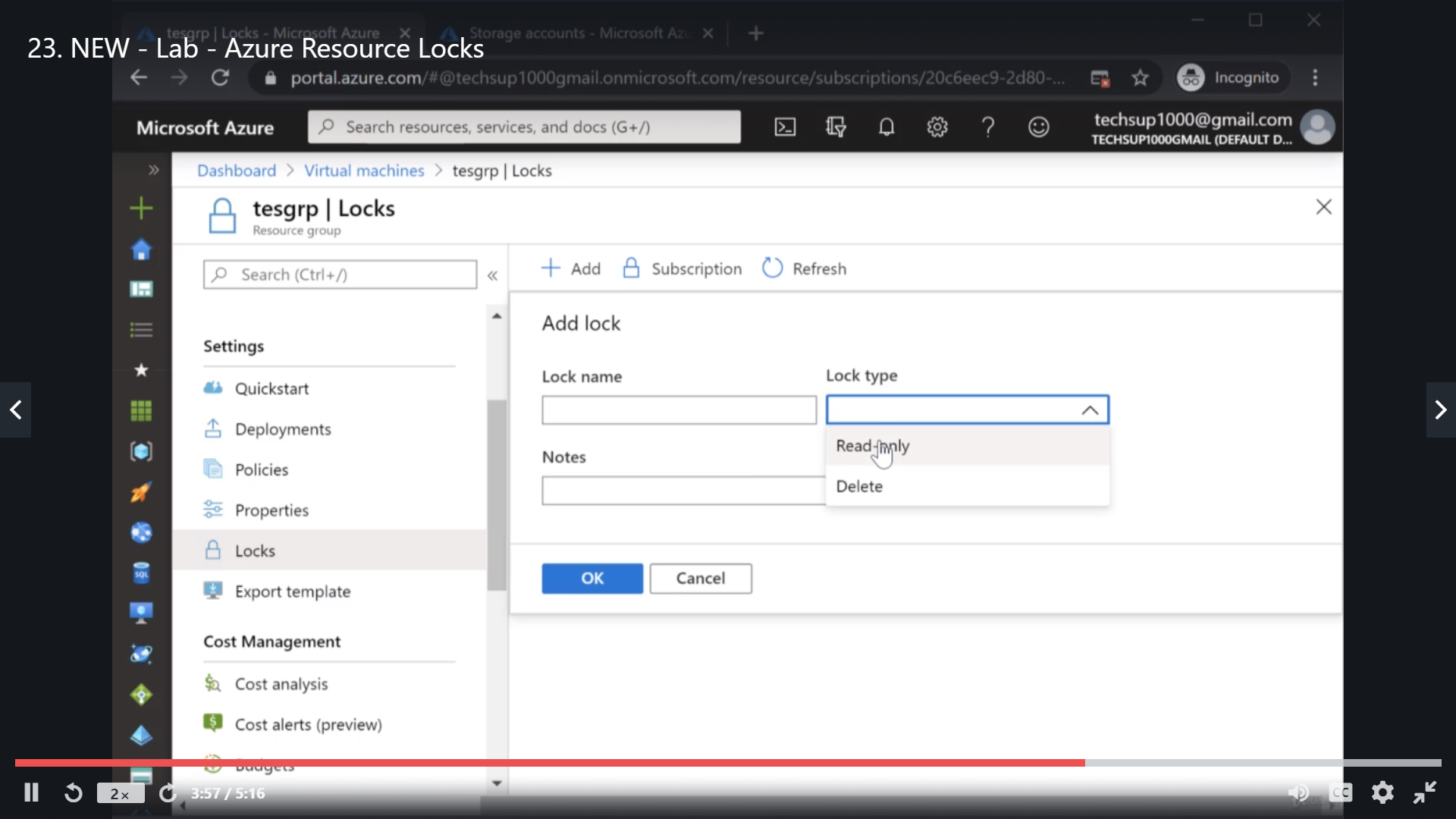
## Virtual Machine

### Resource Locks

Two types of locks:

Read Only : Will not allow any changes in the resource

Delete : Will not allow deleting the resource

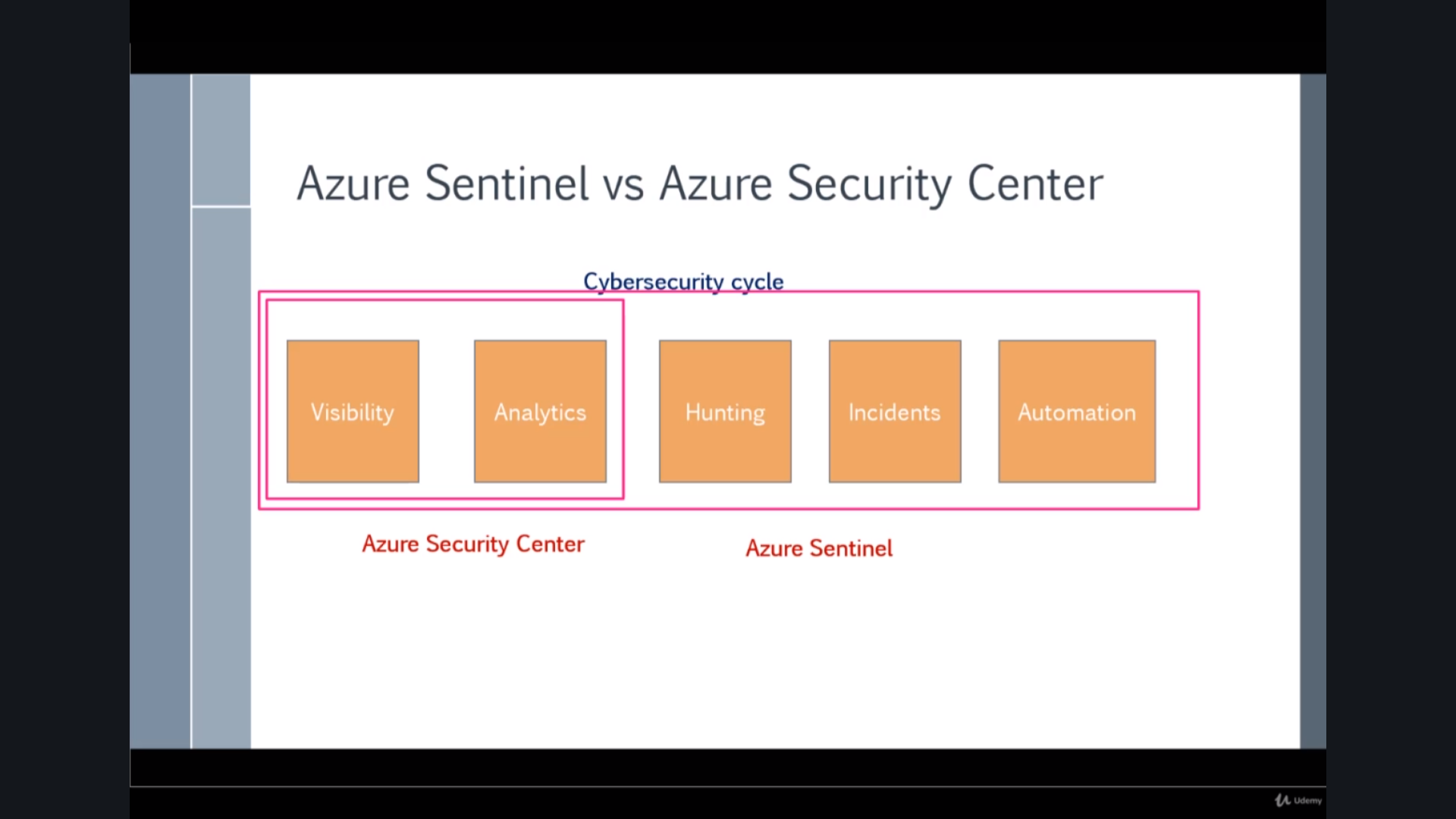


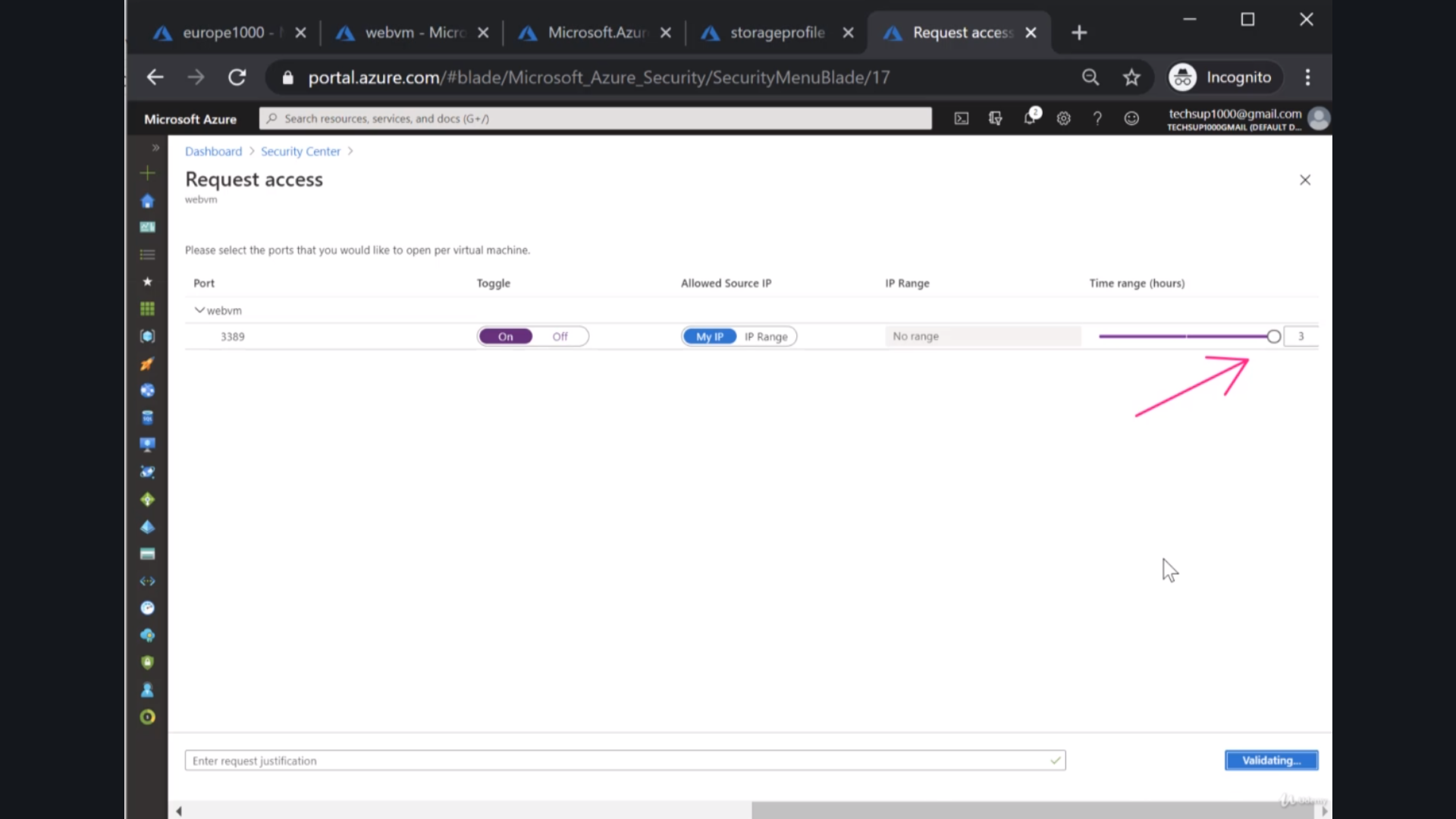
Even a **global administrator** can’t overwrite locks. It is only after deleting the lock itself one can make changes in the lock.

### JIT VM Access

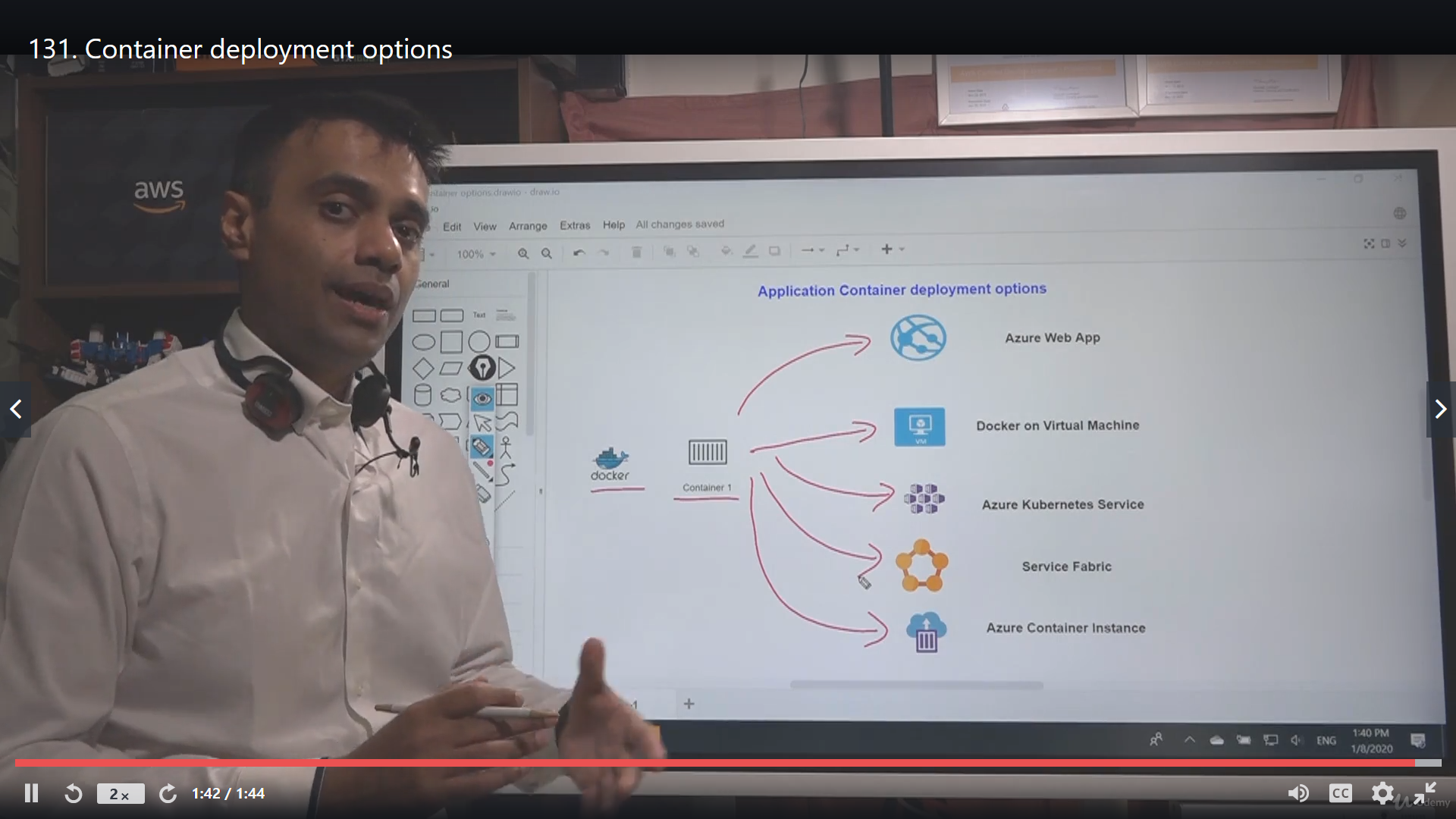
<https://docs.microsoft.com/en-us/azure/security-center/just-in-time-explained>

Its a **Security Centre** feature





## Container Deployment Options:



## Service Discovery:

an Azure Container Service with Kubernetes (AKS) environment can handle service instance registration and deregistration. It also runs a proxy on each cluster host that plays the role of server-side discovery router.

# Design a network solution

* recommend a solution for network addressing and name resolution
* recommend a solution for network provisioning
* recommend a solution for network security o private endpoints
  + Firewalls
  + Gateways
* recommend a solution for network connectivity to the Internet, on-premises networks, and other Azure virtual networks
* recommend a solution for automating network management
* recommend a solution for load balancing and traffic routing

# Design an application architecture

* recommend a microservices architecture including Event Grid, Event Hubs, Service Bus, Storage Queues, Logic Apps, Azure Functions, and webhooks
* recommend an orchestration solution for deployment of applications including ARM templates, Logic Apps, or Azure Functions
  + select an automation method
  + choose which resources or lifecycle steps will be automated
  + design integration with other sources such as an ITSM solution
  + recommend a solution for monitoring automation
* recommend a solution for API integration
  + design an API gateway strategy
  + determine policies for internal and external consumption of APIs
  + recommend a hosting structure for API management
  + recommend when and how to use API Keys

## Azure API Management

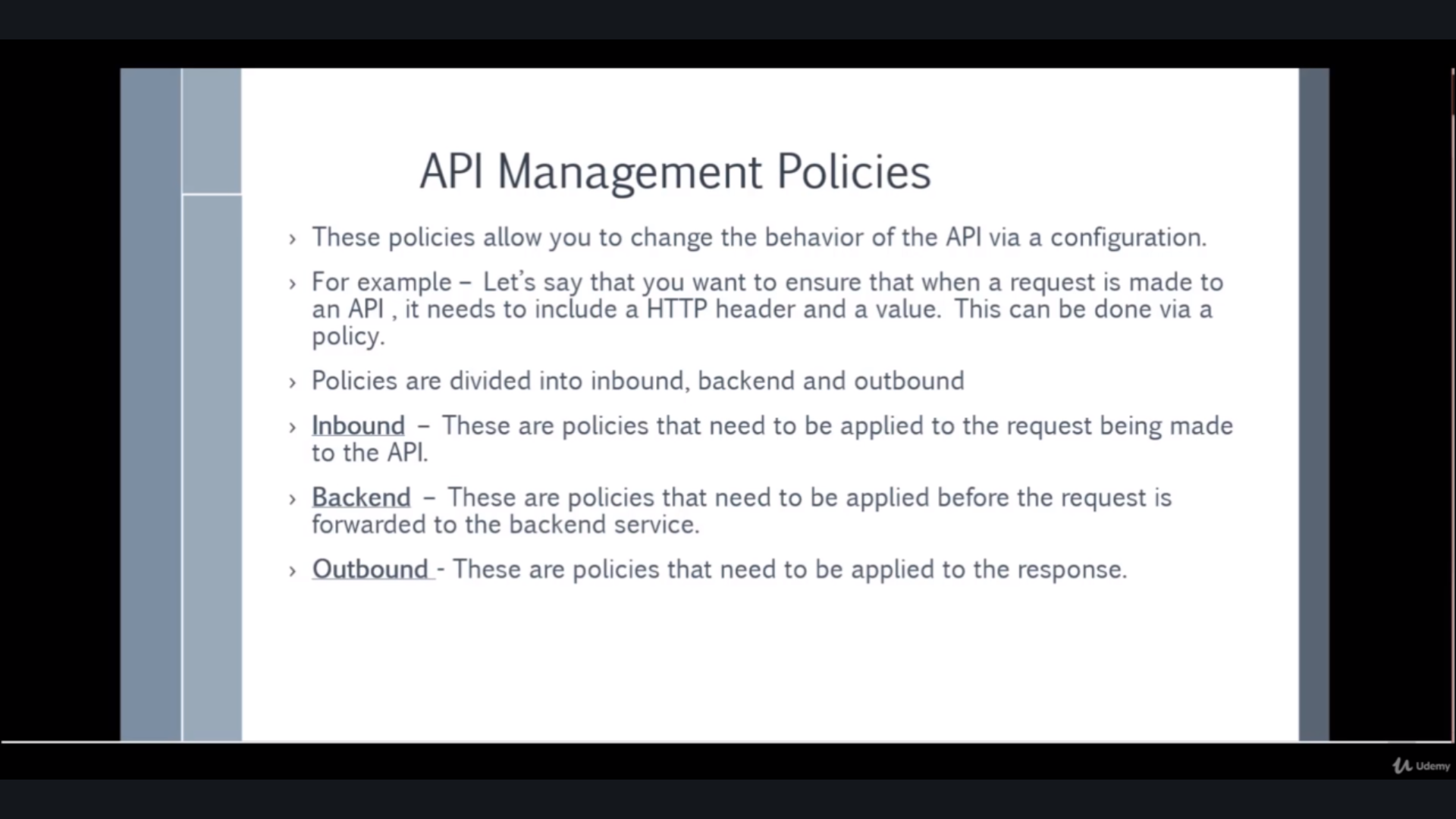
Its helpful in :

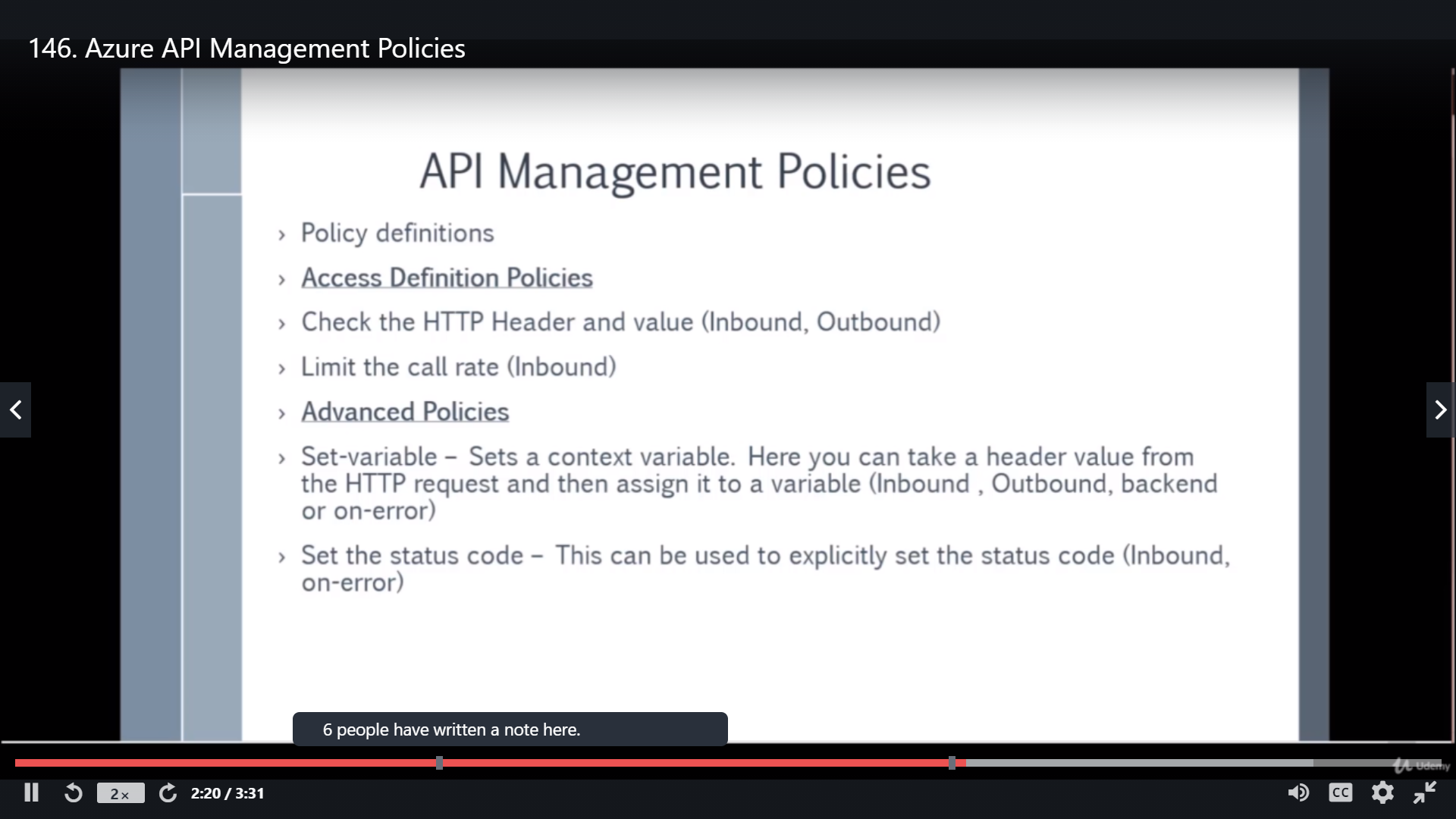
* Making request secure
* Cache results
* Rate Limits

### API Management within Virtual Network

### API Management and Azure Functions

### API Management Policies





# Design migrations

* assess and interpret on-premises servers, data, and applications for migration
* recommend a solution for migrating applications and VMs
* recommend a solution for migration of databases
  + determine migration scope, including redundant, related, trivial, and outdated data