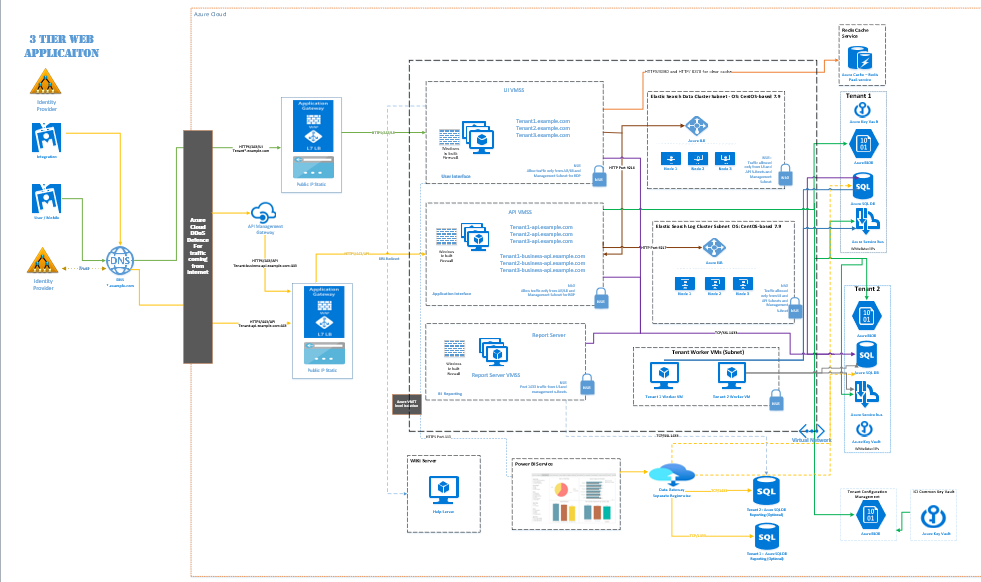
**Challenge #1**

A 3-tier environment is a common setup. Use a tool of your choosing/familiarity create these resources on a cloud environment (Azure/AWS/GCP). Please remember we will not be judged on the outcome but more focusing on the approach, style, and reproducibility.

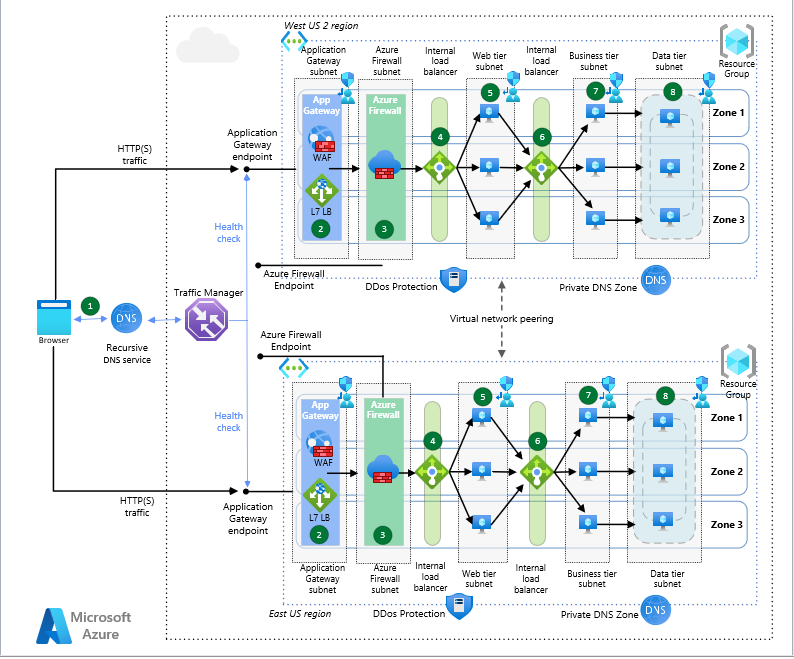
**Example – 1**



**Components**

* [Azure Application Gateway](https://azure.microsoft.com/services/application-gateway/) is a layer-7 load balancer with optional Web Application Firewall (WAF) functionality. The v2 SKU of Application Gateway supports availability zone redundancy and it is recommended for most scenarios. The Application Gateway includes configurable horizontal autoscaling so that it can react automatically to application demand changes.
* [Azure Load Balancer](https://azure.microsoft.com/services/load-balancer/) is a layer-4 load balancer. A zone-redundant load balancer will still distribute traffic with an availability zone failure to the remaining zones.
* [Azure DDoS Protection](https://azure.microsoft.com/services/ddos-protection/) has enhanced features to protect against distributed denial of service (DDoS) attacks.
* [Azure DNS](https://azure.microsoft.com/services/dns/) is a hosting service for DNS domains. It provides name resolution using Microsoft Azure infrastructure. By hosting your domains in Azure, you can manage your DNS records using the same credentials, APIs, tools, and billing as your other Azure services.
* [Azure Private DNS zones](https://learn.microsoft.com/en-us/azure/dns/private-dns-overview) are a feature of Azure DNS. Azure DNS Private Zones provide name resolution within a virtual network, and between virtual networks. The records contained in a private DNS zone aren't resolvable from the Internet. DNS resolution against a private DNS zone works only from virtual networks linked to it.
* [Azure Virtual Machines](https://azure.microsoft.com/services/virtual-machines/) are on-demand, scalable computing resources that give you the flexibility of virtualization but eliminate the maintenance demands of physical hardware. The operating system choices include Windows and Linux. Certain components of the applications can be replaced with platform-as-a-service Azure resources
* [Azure Virtual Machine Scale Sets](https://azure.microsoft.com/services/virtual-machine-scale-sets/) is automated and load-balanced virtual machine scaling that simplifies management of your applications and increases availability.
* [Azure SQL Elastic Pools](https://learn.microsoft.com/en-us/azure/sql-database/sql-database-elastic-pool): Provides a solution for managing a set of databases flexibly with a pool of resources. The service allocates resources on demand to the databases. It gives the developer of a multitenant SaaS architecture the power to deliver database resources to clients as they need it. The service also reduces the budget and overhead of maintaining multiple SQL Servers with large chunks of unused compute resources.
* [Azure Virtual Network](https://azure.microsoft.com/services/virtual-network/) is a secure private network in the cloud. It connects virtual machines to one another, to the Internet, and to cross-premises networks.
* [Azure Cache for Redis](https://azure.microsoft.com/services/cache) is a fully managed, in-memory cache that enables high-performance and scalable architectures. You can use it to create cloud or hybrid deployments that handle millions of requests per second at submillisecond latency—all with the configuration, security, and availability benefits of a managed service. Although Azure Cache for Redis is often used as a data cache to improve application performance, you can also use it as a message broker.

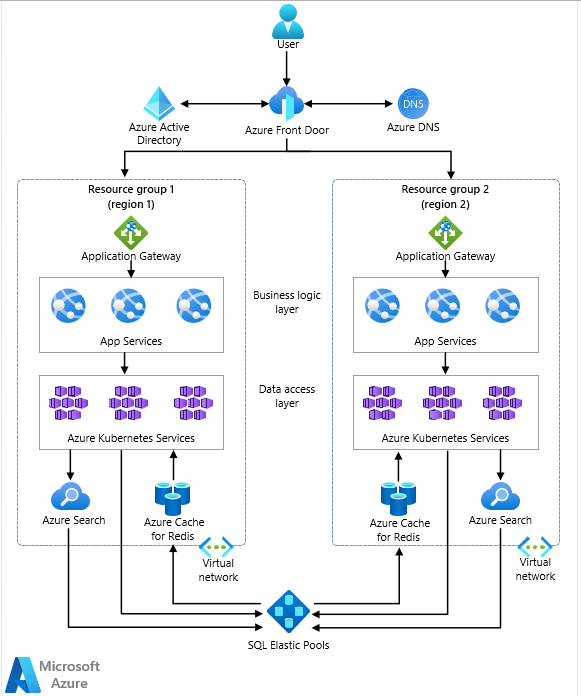
**Example 2**



**Components**

* [Azure Firewall](https://learn.microsoft.com/en-us/azure/private-link/private-link-overview) is a cloud-based, Microsoft-managed next-generation firewall that provides deep packet inspection for both North/South and East/West traffic flows. It can be spread across Availability Zones and it offers automatic autoscaling to cope with application demand changes.
* [Azure Application Gateway](https://azure.microsoft.com/services/application-gateway/) is a layer-7 load balancer with optional Web Application Firewall (WAF) functionality. The v2 SKU of Application Gateway supports availability zone redundancy and it is recommended for most scenarios. The Application Gateway includes configurable horizontal autoscaling so that it can react automatically to application demand changes.
* [Azure Traffic Manager](https://azure.microsoft.com/services/traffic-manager/) is a DNS-based global traffic load balancer that distributes traffic to services across global Azure regions while providing high availability and responsiveness. For more information, see the section [Traffic Manager configuration](https://learn.microsoft.com/en-us/azure/architecture/reference-architectures/n-tier/multi-region-sql-server#traffic-manager-configuration).
* [Azure Load Balancer](https://azure.microsoft.com/services/load-balancer/) is a layer-4 load balancer. A zone-redundant load balancer will still distribute traffic with an availability zone failure to the remaining zones.
* [Azure DDoS Protection](https://azure.microsoft.com/services/ddos-protection/) has enhanced features to protect against distributed denial of service (DDoS) attacks.
* [Azure DNS](https://azure.microsoft.com/services/dns/) is a hosting service for DNS domains. It provides name resolution using Microsoft Azure infrastructure. By hosting your domains in Azure, you can manage your DNS records using the same credentials, APIs, tools, and billing as your other Azure services.
* [Azure Private DNS zones](https://learn.microsoft.com/en-us/azure/dns/private-dns-overview) are a feature of Azure DNS. Azure DNS Private Zones provide name resolution within a virtual network, and between virtual networks. The records contained in a private DNS zone aren't resolvable from the Internet. DNS resolution against a private DNS zone works only from virtual networks linked to it.
* [Azure Virtual Machines](https://azure.microsoft.com/services/virtual-machines/) are on-demand, scalable computing resources that give you the flexibility of virtualization but eliminate the maintenance demands of physical hardware. The operating system choices include Windows and Linux. Certain components of the applications can be replaced with platform-as-a-service Azure resources (for example the database and the frontend tier), but the architecture wouldn't change significantly if using [Private Link](https://learn.microsoft.com/en-us/azure/private-link/private-link-overview) and [App Service VNet Integration](https://learn.microsoft.com/en-us/azure/app-service/overview-vnet-integration) to bring those PaaS services into the virtual network.
* [Azure Virtual Machine Scale Sets](https://azure.microsoft.com/services/virtual-machine-scale-sets/) is automated and load-balanced virtual machine scaling that simplifies management of your applications and increases availability.
* [SQL Server on VMs](https://azure.microsoft.com/services/virtual-machines/sql-server/#overview) lets you use full versions of SQL Server in the cloud without having to manage any on-premises hardware.
* [Azure Virtual Network](https://azure.microsoft.com/services/virtual-network/) is a secure private network in the cloud. It connects virtual machines to one another, to the Internet, and to cross-premises networks.
* [User-Defined Routes](https://learn.microsoft.com/en-us/azure/virtual-network/virtual-networks-udr-overview) are a mechanism to override the default routing in virtual networks. In this scenario they are used to force traffic inbound and outbound traffic flows to traverse the Azure Firewall.

**Example 3 –**



**Components**

* [Azure Front Door](https://azure.microsoft.com/services/frontdoor/): A regional load balancer that routes client traffic to the correct region. It can fail over to the second region if a regional failure happens, and it can secure the internet-facing entry point via [Azure Web Application Firewall](https://learn.microsoft.com/en-us/azure/web-application-firewall/ag/ag-overview).
* [Azure Active Directory (Azure AD)](https://azure.microsoft.com/services/active-directory/): Acts as the identity provider for the entire application, enforcing authentication and end-to-end authorization of the request in the application.
* [Azure DNS](https://azure.microsoft.com/services/dns/): A hosting service in Azure for domain name resolution. In a multitenant solution, multiple clients access the solution via their own individual domains. Use Azure DNS to configure and resolve client requests to their correct application stack.
* [Application Gateway](https://azure.microsoft.com/services/application-gateway/): Routes and load-balances traffic internally in the application to the various services that satisfy client business needs. While Azure Front Door balances load across high-level regions, it's Application Gateway that has awareness of the load on individual services within a group. Azure Front Door and Application Gateway combine to provide complex load-balancing at all levels in a multitenant solution. For more information on load-balancing options in Azure, visit this [overview on Azure load-balancing](https://learn.microsoft.com/en-us/azure/architecture/guide/technology-choices/load-balancing-overview).
* [App Service](https://azure.microsoft.com/services/app-service/): Azure's premier service for web applications and web-based APIs. Security integrates with services like Azure AD and [Azure Key Vault](https://azure.microsoft.com/services/key-vault/). You can configure automatic scaling. Also, the amount of resources available to scale to is flexible between the various App Service plans on which the app can run. App Service can also leverage integrated DevOps capabilities for continuous integration and deployment to multiple environments. These and other supporting features of the Azure platform allow for developers to focus on the development of their applications.
* [Azure Kubernetes Service (AKS)](https://azure.microsoft.com/services/kubernetes-service/): Orchestrates instances of container images deployed to a cluster. Managing multiple clients' data often involves implementing a suite of components to manage:
  + Data modeling
  + Data source connectivity
  + Extract, transform, load (ETL)
  + Import/export activities

Developing these many smaller components as container-based microservices creates an ideal scenario for the deployment to an AKS cluster. Tools for autoscaling, load balancing, and upgradeability are built into the framework. AKS integrates well with a continuous integration/continuous delivery (CI/CD) strategy using the available DevOps features and Azure Container Registry.

* [Azure SQL Elastic Pools](https://learn.microsoft.com/en-us/azure/sql-database/sql-database-elastic-pool): Provides a solution for managing a set of databases flexibly with a pool of resources. The service allocates resources on demand to the databases. It gives the developer of a multitenant SaaS architecture the power to deliver database resources to clients as they need it. The service also reduces the budget and overhead of maintaining multiple SQL Servers with large chunks of unused compute resources.
* [Azure Cognitive Search](https://azure.microsoft.com/services/search/) (formerly known as Azure Search): A service that adds a powerful indexing and query engine to your application. It gives clients access to strong query functionality. They can also use Azure's AI capabilities to enrich and enhance the query functionality. Azure Cognitive Search can account for multitenancy using an index-per-tenant or service-per-tenant strategy.
* [Azure Cache for Redis](https://azure.microsoft.com/services/cache/): Applies a caching layer as a service to the solution, providing an in-memory managed cache to reduce latency and increase performance for the clients. High throughput allows for a high volume of requests to handle multiple tenants accessing the system. You can flexibly scale up the service as application loads increase. It also supports encryption at rest to protect and isolate cached tenant data.

**Challenge #2**

We need to write code that will query the meta data of an instance within AWS or Azure or GCP and provide a Json formatted output. The choice of language and implementation is up to you.

$outputPath = "C:\test"

New-Item -Path $outputPath -ItemType Directory -ErrorAction Ignore

$vmname= Get-ComputerInfo | select csname

$Proxy=New-object System.Net.WebProxy

$WebSession=new-object Microsoft.PowerShell.Commands.WebRequestSession

$WebSession.Proxy=$Proxy

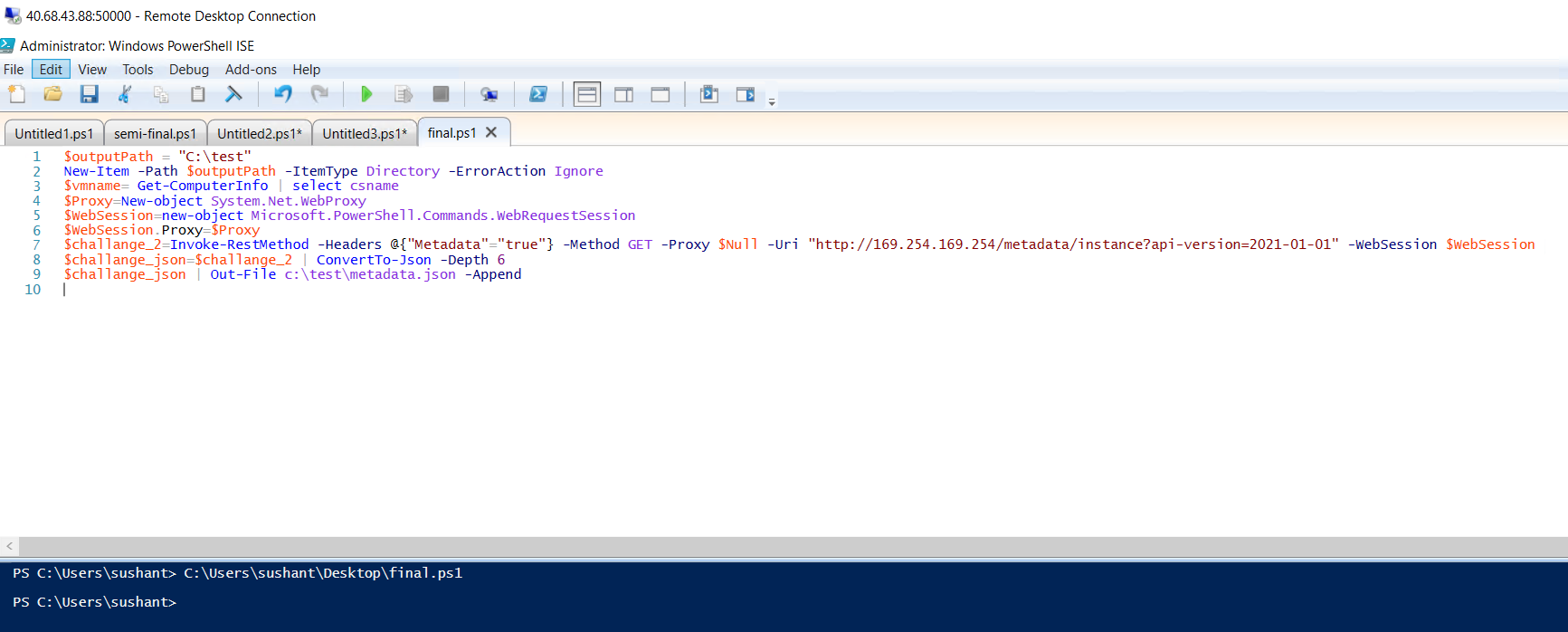
$challange\_2=Invoke-RestMethod -Headers @{"Metadata"="true"} -Method GET -Proxy $Null `

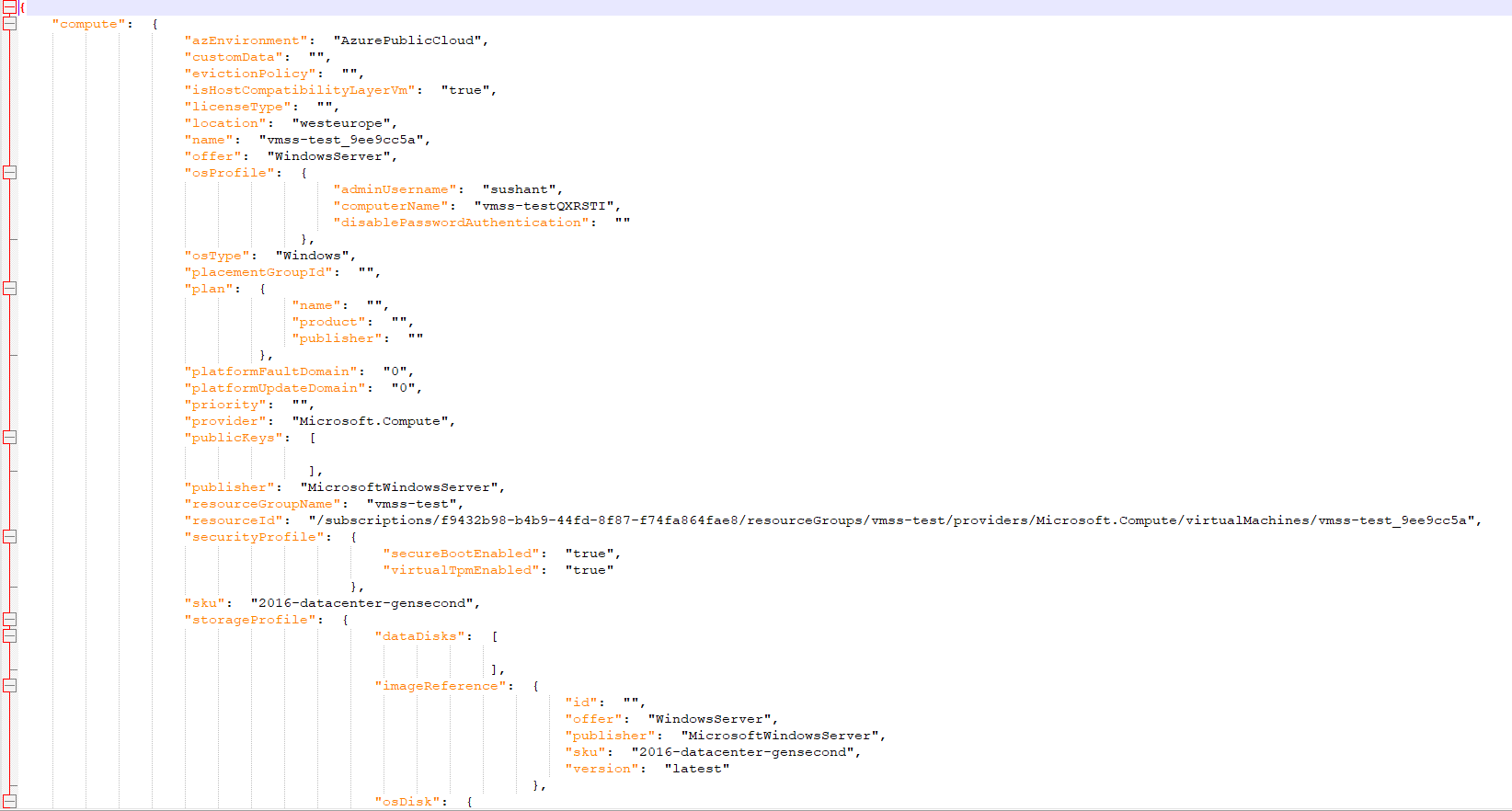
-Uri "http://169.254.169.254/metadata/instance?api-version=2021-01-01" `

-WebSession $WebSession

$challange\_json=$challange\_2 | ConvertTo-Json -Depth 6

$challange\_json | Out-File c:\test\metadata.json -Append





**Challenge #3**

We have a nested object. We would like a function where you pass in the object and a key and get back the value. The choice of language and implementation is up to you.

Example Inputs

object = {“a”:{“b”:{“c”:”d”}}}

key = a/b/c

object = {“x”:{“y”:{“z”:”a”}}}

key = x/y/z

value = a

