Core Networking

Designing and Implementing Microsoft Azure Networking Solutions

KNOWLEDGE CHEATSHEET

CORE NETWORKING DECISIONS

Address Segment → **DNS** Resolve → **Route** Forward → **Monitor** Diagnose

01 Design and implement core networking infrastructure 25–30%

Address spaces and subnet arithmetic

- **VNets are regional** and span availability zones. Subnets also span zones; separate subnets are not required simply because VMs occupy different zones.
- **IPv4 usable addresses = 2^(32 − prefix) − 5**. Azure reserves the first four and last address in every subnet: /29 gives 3 usable, /27 gives 27, /25 gives 123, and /23 gives 507.
- Choose **non-overlapping ranges** for peering and hybrid connectivity. Reserve growth space; expanding a VNet does not automatically resize its subnets or update every peer.
- **Dual-stack** design adds IPv6 address space, /64 IPv6 subnets and IPv6 NIC configurations alongside IPv4. An IPv6 VNet prefix and an IPv6 subnet prefix serve different scopes.
- Set a VM's **static private IP** on its Azure NIC configuration. Keep guest networking consistent with Azure allocation; changing only the guest address can break connectivity.

Subnets for managed services

- **GatewaySubnet** holds virtual network gateways. Plan /27 or larger for gateway growth and VPN/ExpressRoute coexistence; do not deploy ordinary workloads or an NSG there.
- **AzureFirewallSubnet** requires /26 or larger. **AzureBastionSubnet** requires /26 or larger for dedicated Bastion deployments. Forced-tunnel firewall designs also use a management subnet.
- **Application Gateway** needs a dedicated subnet. Allow addresses for maximum instances, any private frontend, Azure reservations and upgrade headroom; /24 is a practical v2 planning size.
- **Subnet delegation** authorizes a service to manage resources in the subnet. App Service integration, SQL Managed Instance and container groups have distinct subnet requirements; do not combine incompatible delegations.

Core Networking

Designing and Implementing Microsoft Azure Networking Solutions

KNOWLEDGE CHEATSHEET

CORE NETWORKING DECISIONS

Address Segment → **DNS** Resolve → **Route** Forward → **Monitor** Diagnose

Public IP addresses and prefixes

- **Public IP SKU**, region, IP version and zone properties must match the consuming resource. Standard public IPs use static allocation; explicitly permit required inbound traffic.
- A **public IP prefix** reserves a contiguous block of Azure public addresses for predictable allocation and allowlists. A **custom IP prefix** brings an owned public range into Azure after ownership and routing validation.
- A public IP on a gateway identifies that gateway; it does not automatically become the outbound address of all VMs in the VNet. Select an explicit workload egress method.

Public and private DNS

- **Public DNS delegation**: create the Azure DNS zone, obtain its assigned name servers, then update delegation at the parent zone or registrar. Hosting a zone does not register the domain.
- **A / AAAA** map names to IPv4 / IPv6. **CNAME** maps an alias to another name; **TXT** commonly proves domain ownership. Use supported alias records for zone-apex targets.
- **Private DNS zones** resolve through linked VNets. Peering alone does not link a zone. A resolution link makes records visible; an autoregistration link also manages eligible VM records.
- A VNet can have **one autoregistration zone** and multiple resolution-only links. Autoregistration manages VM lifecycle records; manually created records require separate lifecycle management.
- **Split-horizon DNS** uses public and private answers for the same namespace. Check the client's resolver and zone links before changing records; a missing private answer can differ from a public answer.

Hybrid DNS and Private Resolver

- **Azure-provided DNS: 168.63.129.16** serves Azure workloads. On-premises clients cannot query it directly; use a reachable DNS forwarder or Azure DNS Private Resolver.
- **Inbound endpoint**: on-premises DNS conditionally forwards Azure private namespaces to its private IP. Link the required private zones to the resolver's VNet.
- **Outbound endpoint + forwarding ruleset**: Azure forwards selected namespaces to on-premises DNS. Link the ruleset to client VNets and ensure network reachability to target DNS servers.
- Resolver endpoints use **separate dedicated delegated subnets**, each /28 or larger within supported limits. A ruleset link alone does not require another resolver in every spoke.
- **Custom DNS settings** redirect VNet clients to chosen servers; refresh client DHCP settings after changes. Avoid forwarding loops. Azure Firewall DNS proxy can centralize queries through its private IP.

Core Networking

Designing and Implementing Microsoft Azure Networking Solutions

KNOWLEDGE CHEATSHEET

CORE NETWORKING DECISIONS

Address Segment → **DNS** Resolve → **Route** Forward → **Monitor** Diagnose

Route selection and inspection paths

- **Longest prefix wins** before route-source priority. For equal prefixes, the general order is **UDR** → **BGP** → **system**; service-endpoint and certain platform routes have special preference.
- Associate a **route table with a subnet**. A route selects a destination prefix and next hop, such as Virtual appliance, Virtual network gateway, Internet or None; it does not itself permit traffic.
- For NVA inspection, send relevant prefixes to the appliance's **private IP**, enable Azure NIC IP forwarding and guest forwarding, and configure an allowed return path.
- **0.0.0.0/0** changes the default path but does not override more-specific routes. Forced tunneling can send internet egress to an on-premises firewall; check service dependencies and SNAT.
- **Symmetric routing** keeps both directions through a stateful inspection device. Inspect effective routes at both ends; do not disable gateway route propagation on GatewaySubnet.

Peering, transit and Route Server

- **VNet peering** connects non-overlapping networks over Microsoft's backbone, including across regions and subscriptions. Configure both directions and verify a Connected state.
- **Peering is non-transitive**: two spokes peered only to a hub do not automatically reach one another. Add direct connectivity or an explicitly routed transit design.
- **Gateway transit**: allow gateway transit on the hub peering and use the remote gateway on the spoke. A spoke using a remote gateway cannot also use its own VNet gateway.
- **Allow forwarded traffic** permits eligible traffic forwarded from a peer; it neither creates routes nor turns the peer into a router. Pair it with the intended appliance and UDRs.
- **Azure Route Server** exchanges BGP routes with NVAs and supported gateways. It is a control-plane service; packets traverse the chosen data path, not Route Server. Gateway/NVA transit needs the appropriate route exchange configuration.

Core Networking

Designing and Implementing Microsoft Azure Networking Solutions

KNOWLEDGE CHEATSHEET

CORE NETWORKING DECISIONS

Address Segment → **DNS** Resolve → **Route** Forward → **Monitor** Diagnose

Outbound connectivity and SNAT

- **NAT Gateway** provides subnet-level outbound connectivity with stable public addresses. One gateway can serve several subnets in one VNet; it does not accept unsolicited inbound connections.
- For new internet-bound connections, an associated **NAT gateway takes precedence** over instance public IPs and load-balancer outbound rules. A UDR to an appliance or gateway changes the path before internet SNAT.
- **Standard NAT Gateway** uses Standard static IPv4 public addresses or prefixes. **StandardV2** supports zone redundancy and IPv6 and requires matching StandardV2 public addresses.
- A **zonal Standard NAT gateway** can serve VMs from other zones, but egress depends on its deployment zone. Align zone-specific subnets and gateways, or use supported zone-redundant capacity.
- **SNAT exhaustion** reflects insufficient concurrent translations, often to the same destination. Reuse connections, monitor failures and add public IP capacity; adding another subnet to the same gateway does not add ports.
- Provide **explicit outbound access** for private workloads. NAT gateways and service endpoints do not consume private addresses from the workload subnet; private endpoints and internal load-balancer frontends do.

Network diagnostics and monitoring

- **IP flow verify** evaluates an NSG allow/deny decision for a VM flow. **Next hop** identifies the selected route. Neither test proves the guest application is listening.
- **Connection troubleshoot** checks reachability to a destination. **Connection Monitor** repeatedly tests endpoints for reachability, loss and latency and supports Azure and on-premises sources with the required agents.
- **Packet capture** records packets with optional filters and size/time limits. Install the required Network Watcher extension; write captures to supported Storage or a local VM path.
- **Diagnostic settings** export supported resource logs and metrics to Storage, Log Analytics or Event Hubs. **KQL** queries collected logs; Activity Log alerts detect control-plane changes.
- **Azure Monitor Network Insights** summarizes network health and topology. **Defender for Cloud** exposes network recommendations and attack paths; use resource queries to find internet exposure and missing server protection.

Hybrid Connectivity

Designing and Implementing Microsoft Azure Networking Solutions

KNOWLEDGE CHEATSHEET

HYBRID CONNECTION CHOICES

- **S2S VPN** — Sites
- **P2S VPN** — Users
- **ExpressRoute** — Private circuits
- **Virtual WAN** — Managed transit

02 Design, implement, and manage connectivity services 20–25%

Site-to-site VPN building blocks

- **S2S VPN** encrypts traffic with IPsec/IKE. Deploy a VPN virtual network gateway, a **local network gateway** describing the remote VPN endpoint and prefixes, and a connection with matching shared secrets.
- **Route-based VPN** uses routing to choose tunnels and supports richer multi-site and BGP designs. For compatible policy-based devices, use supported custom IPsec/IKE policies and policy-based traffic selectors.
- An **IPsec/IKE policy** defines algorithms, integrity, Diffie-Hellman/PFS groups and lifetimes. Both peers must agree; creating a policy object alone does not apply it to a connection.
- **BGP** needs peer IPs and ASNs on the Azure and on-premises sides. It exchanges reachable prefixes and supports dynamic failover; static local-network address spaces serve non-BGP designs.
- **AZ gateway SKUs** support zonal resilience. Active-active designs use both gateway instances with redundant remote tunnels; size the SKU for aggregate throughput, tunnel count and authentication needs.
- For a failed tunnel, check endpoint reachability, shared keys, selectors and policies. **IKEDiagnosticLog** helps with negotiation; tunnel, route and gateway logs expose different stages. Prefer a connection-scoped reset when only one connection is affected.

Point-to-site access and authentication

- **P2S VPN** connects individual clients from a separate address pool. Plan pool capacity and avoid overlaps with Azure and remote networks; P2S is not a replacement for a branch S2S gateway.
- **Microsoft Entra authentication** uses OpenVPN and Azure VPN Client. Configure the gateway's tenant, issuer and audience to match the application registration and client profile.
- **Certificates** require a trusted root uploaded to the gateway and an eligible client certificate with its private key. **RADIUS** integrates supported authentication servers, including AD DS-backed authentication.
- Match **tunnel protocol, OS, client and authentication** together. SSTP, IKEv2 and OpenVPN have different support; Entra authentication is not enabled merely by installing an IKEv2 client.
- **Conditional Access**, authentication strengths and risk policies can govern Entra VPN sign-in where supported. Use application assignments and appropriately separated audiences/apps when gateway access must differ by group.
- After topology or gateway changes, **regenerate and import client profiles** and verify installed routes and DNS. Gateway transit does not guarantee that every client has learned every reachable prefix.

Hybrid Connectivity

Designing and Implementing Microsoft Azure Networking Solutions

KNOWLEDGE CHEATSHEET

HYBRID CONNECTION CHOICES

- **S2S VPN** — Sites
- **P2S VPN** — Users
- **ExpressRoute** — Private circuits
- **Virtual WAN** — Managed transit

Hybrid connectivity edge cases

- **Overlapping prefixes** prevent ordinary routed connectivity. Prefer address redesign; supported VPN Gateway NAT can translate address spaces, with SKU, connection and BGP limitations.
- **Azure Extended Network** stretches a subnet for migration scenarios using paired appliances and Windows Admin Center. Treat it as a specific migration design rather than normal VNet peering.
- **Azure Network Adapter** configures a server's P2S connectivity through Windows Admin Center. An existing compatible Azure VPN gateway reduces setup work.
- **ExpressRoute + VPN coexistence** uses distinct gateway resources in a sufficiently sized GatewaySubnet. Plan BGP route preference and test failover; a backup VPN must carry the required prefixes and capacity.

ExpressRoute circuits and peerings

- **ExpressRoute** provides private provider connectivity to Microsoft's network. Provision a circuit, pass its **service key** to the provider, configure peering, then connect the VNet gateway.
- **Azure private peering** reaches VNet private addresses. **Microsoft peering** reaches supported Microsoft public services; **route filters** select service BGP communities for advertised prefixes.
- **Local** limits access to regions associated with its peering location and includes data transfer. **Standard** reaches supported regions in the geopolitical area. **Premium** expands geographic reach and connection/route limits.
- Separate **circuit bandwidth**, gateway throughput and the data billing plan. The peering location is not the Azure deployment region; verify reach and supported plan combinations before choosing the lowest-cost circuit.
- One circuit can connect multiple VNets, including authorized VNets in other subscriptions. The circuit owner creates a **connection authorization**; the VNet owner redeems it when creating the gateway connection.

ExpressRoute performance and resilience

- **FastPath** bypasses the ExpressRoute gateway in the supported data path to reduce latency and improve throughput. The gateway remains required; verify gateway SKU and feature-path support.
- **Global Reach** joins eligible ExpressRoute circuits so on-premises sites communicate through Microsoft's backbone. It solves site-to-site transit, a different requirement from FastPath.
- **ExpressRoute Direct** supplies dedicated physical port pairs on Microsoft's edge. Circuits are logical connections on connectivity infrastructure; Direct is not the circuit reach SKU.
- **BFD** detects eligible link failures faster than normal BGP timers. Redundant links within one circuit do not protect against every peering-location failure; use diverse circuits and locations for broader resilience.

Hybrid Connectivity / Application Delivery

Designing and Implementing Microsoft Azure Networking Solutions

KNOWLEDGE CHEATSHEET

HYBRID CONNECTION CHOICES

- **S2S VPN** — Sites
- **P2S VPN** — Users
- **ExpressRoute** — Private circuits
- **Virtual WAN** — Managed transit

- ExpressRoute is **not encrypted by default**. Use an IPsec overlay when end-to-end traffic encryption is required; MACsec on supported Direct links protects that link, with a different scope.
- Monitor **ARP/BGP availability**, circuit/provider state and gateway metrics. A zone-redundant gateway addresses Azure zone failure; it does not remove provider or on-premises router dependencies.

Virtual WAN architecture and routing

- **Virtual WAN** coordinates regional virtual hubs, VNet connections and branch/user connectivity. **Standard** supports the broader transit, ExpressRoute and user VPN feature set; Basic focuses on S2S.
- Represent a branch with a **VPN site** and its links, then create a hub VPN connection. A **VNet connection** attaches a spoke; a user VPN configuration supplies P2S settings.
- Scale **S2S, P2S and ExpressRoute gateways independently**. Gateway scale units and virtual hub router capacity are separate controls; size both for aggregate traffic and connection demand.
- **Association** chooses the route table used by traffic entering from a connection. **Propagation** chooses which tables learn that connection's prefixes. Labels can propagate to a group of tables.
- Use separate **route tables and propagation scopes** for isolation. Prevent unintended branch transit between isolated groups; validate routes in both directions rather than assuming a full mesh remains isolated.
- **Routing intent** steers private or internet traffic through supported security providers. Use connection/static routes for selective paths and supported route maps for route filtering or transformation.
- The hub's **routing preference** influences eligible ExpressRoute/VPN path selection. Longest prefix, AS path and configured preference all matter; an ExpressRoute link is not an unconditional winner.
- Integrate supported **NVAs/SD-WAN** with virtual hubs. During hub-and-spoke migration, replace old hub peerings with VNet connections and remove obsolete transit UDRs; keep application gateways in suitable connected VNets.

03 Design and implement application delivery services 15–20%

Choose the traffic delivery service

- **Load Balancer** distributes TCP/UDP flows at layer 4. **Application Gateway** proxies regional HTTP(S) at layer 7. **Front Door** proxies global HTTP(S) at Microsoft's edge.
- **Traffic Manager** returns DNS answers; client traffic goes directly to the chosen endpoint. It does not terminate TLS or inspect URL paths, and failover depends partly on DNS caching.

Application Delivery

Designing and Implementing Microsoft Azure Networking Solutions

KNOWLEDGE CHEATSHEET

TRAFFIC DELIVERY LAYERS

- **Load Balancer** — TCP / UDP
- **App Gateway** — Regional web
- **Front Door** — Global web
- **Traffic Manager** — DNS

- A **public frontend** receives internet traffic; an **internal frontend** has a private IP. Combine regional backends with a supported global service when regional failure must be tolerated.
- **Gateway Load Balancer** inserts compatible NVAs transparently through service chaining and tunnel interfaces. It preserves flow symmetry for inspection of supported public traffic.

Load Balancer rules and probes

- A load-balancing design needs a **frontend**, **backend pool**, **health probe** and **rule** linking the ports and protocols. Backend membership alone does not create a traffic path.
- **Health probes** remove unhealthy instances from new-flow selection. Match the probe to real application health and allow probe traffic through NSGs and the guest firewall.
- **Inbound NAT rules** map frontend ports to individual backend targets. A **load-balancing rule** distributes traffic among healthy backends; **outbound rules** allocate explicit SNAT capacity.
- **HA ports** on supported internal Standard Load Balancer configurations handle all TCP/UDP ports with one rule. **Floating IP** changes address mapping for scenarios such as direct server return; it is not an all-port rule.
- **Session persistence** changes the hash used for client affinity. It does not replicate application session state between backends.
- **Cross-region Load Balancer** uses eligible regional public Standard load balancers as backends. Keep frontend and rule compatibility aligned. Older Basic designs have narrower reach and availability capabilities.

Traffic Manager routing methods

- **Performance** selects by network latency data. **Priority** selects the highest-priority available endpoint. **Weighted** distributes DNS responses according to configured weights.
- **Geographic** maps DNS query origins to regions. **Subnet** maps client resolver address ranges. **Multivalue** can return multiple healthy IPv4/IPv6 endpoints for clients that select a target.
- **Nested profiles** combine routing methods, such as geographic selection followed by weighted rollout. A nested endpoint's minimum healthy child count influences its parent health.
- Map a host name to **profile.trafficmanager.net** using the appropriate DNS record. Configure endpoint probe protocol, port, path and expected responses; include firewall access for monitoring.
- **DNS TTL** limits caching duration, not existing connection lifetime. Weighted routing gives a distribution over DNS responses, not an exact percentage of every HTTP request.

Application Delivery

Designing and Implementing Microsoft Azure Networking Solutions

KNOWLEDGE CHEATSHEET

TRAFFIC DELIVERY LAYERS

- **Load Balancer** — TCP / UDP
- **App Gateway** — Regional web
- **Front Door** — Global web
- **Traffic Manager** — DNS

Application Gateway configuration

- A **listener** identifies frontend IP, port, protocol and optionally hostname. A **routing rule** connects the listener to backend settings and a backend pool.
- **Multi-site listeners** distinguish hostnames sharing a frontend. **Path-based rules** use URL path maps to select backend pools. Define rule priorities and avoid a catch-all intercepting specific hosts.
- **Backend settings** define backend protocol, port, host handling and probe association. A pool can reference supported IPs, FQDNs or scale-set backends; one scale-set target can represent many workers.
- **Custom health probes** can test a specific host, path, port and response range. Wrong host headers, TLS names or blocked probes can mark a functioning server unhealthy.
- **Rewrite rules** alter headers, paths or query strings using conditions, server variables and captures. A **redirect** returns a response telling the client to request another URL.
- **v2 autoscaling** adjusts gateway instance capacity within configured bounds. Capacity units and instance counts describe different measurements; leave subnet headroom for scaling and maintenance.

Application Gateway TLS and certificates

- **TLS termination** decrypts at the gateway; HTTP to the backend leaves that hop unencrypted. **End-to-end TLS** establishes a separate HTTPS session from gateway to backend.
- An HTTPS listener needs a **server certificate with private key**. Backend TLS validation checks the certificate chain and hostname; private CA backends require the appropriate trusted root configuration.
- **Key Vault integration** uses a **user-assigned managed identity** with secret-read permission and network access. Use a versionless secret reference for automatic certificate rotation.
- Application Gateway needs an **exportable software-backed PFX certificate** for TLS termination. HSM-backed, non-exportable certificates do not meet this integration requirement.
- **Mutual TLS** additionally authenticates clients. Configure the trusted client CA chain in an SSL profile, associate it with the HTTPS listener, and provision valid client certificates.

Application Delivery / Private Access

Designing and Implementing Microsoft Azure Networking Solutions

KNOWLEDGE CHEATSHEET

TRAFFIC DELIVERY LAYERS

- **Load Balancer** — TCP / UDP
- **App Gateway** — Regional web
- **Front Door** — Global web
- **Traffic Manager** — DNS

Front Door routes, origins and TLS

- **Endpoint + domain + route** selects incoming traffic; an **origin group** contains backends and load-balancing settings. An **origin** represents a backend endpoint, not each App Service worker.
- Routes match **protocol, domain and path pattern**. More-specific paths distinguish apps; rule sets can rewrite URLs, change headers, redirect or override eligible route behavior.
- Origin selection considers **health, priority, latency sensitivity and weight**. Use a smaller latency sensitivity to narrow eligible nearby origins; equal weights distribute within the eligible set.
- **Health probes** must reach an accurate readiness endpoint. **HEAD** reduces response-body traffic compared with GET when the origin supports it.
- **Custom domains** need ownership validation and DNS mapping to the Front Door endpoint. Configure managed TLS or a supported Key Vault certificate, and separately secure the origin TLS hop.
- **Caching** serves cacheable responses at the edge. Choose query-string behavior and purge content deliberately; cache settings affect content freshness, while routing controls backend selection.
- **Premium** supports Private Link origins and managed WAF capabilities. For public origins, restrict to **AzureFrontDoor.Backend** and validate **X-Azure-FDID** for the intended profile; a DNS CNAME alone does not prevent bypass.

04 Design and implement private access to Azure services 10–15%

Private endpoints and network controls

- A **private endpoint** places a private IP/NIC in the consumer VNet for a specific supported service resource and subresource. Clients can reach it through local, peered or hybrid private connectivity.
- **Connection approval** authorizes the consumer attachment. Network reachability, DNS and service data permissions are still required; a private endpoint does not grant storage or database credentials.
- Plan endpoints for the required **subresources**, such as Blob versus File. Creating an endpoint does not automatically disable the service's public endpoint; configure public network access separately.
- Enable the required **private endpoint network policies** on the endpoint subnet before applying supported NSG or UDR controls. Account for endpoint-specific routes when redirecting traffic to inspection.
- Apply restrictions to the **private destination address** and actual client path. Public-service tags do not necessarily match private-endpoint traffic; test both authorized and unauthorized source networks.

Private Access

Designing and Implementing Microsoft Azure Networking Solutions

KNOWLEDGE CHEATSHEET

PRIVATE SERVICE ACCESS

Client Resolve name → **Private DNS** Private IP → **Endpoint** Approved link → **Service** Authorize data

Private Link name resolution

- Clients normally keep using the **service's normal FQDN**. Public DNS can alias it into a privatelink namespace; the linked private zone supplies an A record for the private IP.
- Common zones include **privatelink.blob.core.windows.net**, **privatelink.database.windows.net** and **privatelink.azurewebsites.net**. Use a private zone, not a public zone exposing private addresses.
- **Private DNS zone groups** integrate endpoint records with the right zones. Link the zones to client VNets or central resolver infrastructure; do not assume VM autoregistration creates PaaS endpoint records.
- For on-premises access, use **conditional forwarding** toward the Azure inbound resolver or DNS forwarder. Validate the complete CNAME chain and final private address before disabling public access.
- A **custom App Service hostname** also needs an application hostname binding and matching TLS certificate. Creating a private CNAME alone does not make the app accept that hostname.

Publish a service with Private Link

- For a custom application, deploy a **Standard Load Balancer**, attach a **Private Link service** to its supported frontend, and let consumers create approved private endpoints to that service.
- Private Link exposes the **selected service**, without requiring full VNet peering or broad network access. It can cross subscriptions and tenants with the appropriate visibility and approval settings.
- **Provider NAT IPs** translate consumer connections and provide connection capacity. Add NAT IP capacity for translation pressure; distinguish NAT capacity from private-endpoint connection quotas.
- **Proxy Protocol v2** can convey consumer connection metadata to a compatible backend. Enabling it requires application support; configure backend ports and rules for each published application.
- For native PaaS Private Link, create an endpoint to the **PaaS resource itself**. A custom Private Link service is for publishing your own supported load-balanced service.

Private Access

Designing and Implementing Microsoft Azure Networking Solutions

KNOWLEDGE CHEATSHEET

PRIVATE SERVICE ACCESS

Client Resolve name → **Private DNS** Private IP → **Endpoint** Approved link → **Service** Authorize data

Service endpoints and endpoint policies

- **Service endpoints** identify a subnet to a supported Azure service and keep traffic on the backbone while retaining the service's public endpoint. They do not allocate a private endpoint IP.
- Enable the **service type on each client subnet**, then allow that subnet in the target service's network firewall. An endpoint alone does not restrict the service to that subnet.
- **Service endpoint policies** restrict eligible Storage destinations for subnet egress. A storage firewall restricts who can reach that account; it does not stop those clients reaching other accounts.
- **Service tags** group Azure address prefixes for NSG rules. **Endpoint policies** select allowed supported resources. A Storage service tag cannot distinguish one storage account from another.
- Storage service-endpoint rules cover supported **paired-region replicas**. Plan regional versus global Storage endpoint scope, service compatibility and policy dependencies; managed-service aliases can support eligible service-owned storage.
- On-premises clients cannot acquire a subnet's **service endpoint identity** through VPN or ExpressRoute. Use private endpoints for private hybrid access, or supported public-endpoint controls for a public design.

App Service inbound and outbound paths

- **VNet integration** gives an App Service app outbound access to private resources. An **App Service private endpoint** gives clients private inbound access. Many solutions require both.
- Regional integration uses a dedicated subnet delegated to **Microsoft.Web/serverFarms**. Plan addresses for all worker instances, scaling and deployment operations; apps in one plan share its workers.
- **Multi plan subnet join** can share an eligible integration subnet across plans in the same subscription; it requires at least /26. Do not assume one subnet per app or universal sharing across plans.
- **Outbound routing settings** determine which app traffic enters the VNet. Pair integration with correct DNS, peering, UDRs and NAT when private dependencies or controlled internet egress are required.
- **Access restrictions** protect the public app endpoint. Private-endpoint access uses its private path and network controls; verify both paths when allowing administrators privately and users through Front Door.

Network Security

Designing and Implementing Microsoft Azure Networking Solutions

KNOWLEDGE CHEATSHEET

NETWORK PROTECTION ROLES

- **NSG** — Flow control
- **Firewall** — Inspection
- **WAF** — Web protection
- **DDoS** — Attack mitigation

05 Design and implement Azure network security services 15–20%

NSG evaluation and application groups

- **NSGs are stateful** layer-3/4 filters. Custom rule priorities run from 100 to 4096; lower numbers run first and the first match wins. Return traffic for an allowed flow is permitted.
- **Subnet and NIC NSGs both apply**. Inbound evaluation crosses the subnet then NIC; outbound crosses the NIC then subnet. Either layer can deny the connection.
- Default rules allow **VirtualNetwork** traffic and outbound internet, with a final inbound deny. Add a narrow allow followed by a broader deny when only one application port should remain open.
- **VirtualNetwork** is a service tag whose effective scope can include connected prefixes. Use explicit subnet prefixes or suitable application groups for tighter boundaries.
- **ASGs** group NIC IP configurations by role, such as web or database. Add new role instances to the group to avoid rewriting address rules; ASG membership is scoped to a VNet.
- When both source and destination use **ASGs**, their NICs must belong to the same VNet. For different VNets, use supported IP-prefix rules and avoid assuming peering merges ASG scope.
- **IMDS: 169.254.169.254** is a link-local platform endpoint. Use guest firewall/process controls for restrictions; ordinary NSG filtering does not block this endpoint.

Azure Firewall policy and traffic types

- **Azure Firewall** is a managed stateful inspection service. Deploy it in the hub's AzureFirewallSubnet and route client traffic to its private IP; firewall rules alone do not attract traffic.
- **Network rules** match source/destination addresses, protocols and ports; supported FQDN network filtering relies on DNS. **Application rules** filter supported application protocols using FQDNs.
- **DNAT rules** publish selected internal services through the firewall's public IP and port. Restrict source ranges; use application rules for outbound HTTPS hostnames rather than an inbound translation.
- **Threat intelligence** can alert or deny known malicious addresses and domains. **Premium** adds TLS inspection, IDPS and deeper web filtering; threat intelligence is also available in Standard.

Network Security

Designing and Implementing Microsoft Azure Networking Solutions

KNOWLEDGE CHEATSHEET

NETWORK PROTECTION ROLES

- **NSG** — Flow control
- **Firewall** — Inspection
- **WAF** — Web protection
- **DDoS** — Attack mitigation

- Rule processing follows service stages: **threat-intelligence filtering**, then **DNAT**, **network** and **application** rules. Lower priorities run first within their applicable hierarchy; application rules cannot override an earlier matching network allow.
- **Firewall Policy** centralizes rule collection groups, rule collections and settings. Parent-policy inheritance and rule-type order matter alongside numeric priority.

TLS inspection, DNS and secured hubs

- **Premium TLS inspection** decrypts supported outbound/east-west TLS for inspection and re-encrypts it. Use an intermediate CA certificate in Key Vault, managed identity access and a root trusted by clients.
- **IDPS** detects known attack signatures; encrypted payload inspection requires the applicable decryption path. **URL filtering** and **web categories** offer finer web controls than a host-level allowlist.
- **DNS proxy** makes clients and firewall use a consistent resolution path. Configure custom DNS when required and forward to the firewall's private IP; verify TCP/UDP DNS reachability.
- **Forced tunneling** separates firewall management connectivity from controlled data egress where required. Verify platform dependencies, including Windows activation/KMS, rather than blindly blocking every outbound path.
- **Firewall Manager** applies centralized policies to eligible deployments. A **secured Virtual WAN hub** uses Azure Firewall for inspection; deploy routing intent or the appropriate routing configuration to send traffic through it.

Web Application Firewall policies

- **WAF** protects HTTP(S) applications from attacks such as SQL injection and cross-site scripting. **Managed rule sets** provide known-attack coverage; **custom rules** express application-specific conditions.
- **Detection mode** logs matches. **Prevention mode** enforces blocking actions. Associate the policy with the intended Front Door domains or Application Gateway scope; an unattached policy does not protect traffic.
- **Custom match rules** can evaluate IP ranges, geographic location, headers, cookies and other supported fields. Conditions within one rule combine logically; use separate rules for independently different actions.
- **Rate-limit rules** are custom rules with a threshold and time window. Choose match conditions and aggregation scope deliberately; a managed attack rule is not a per-client request counter.

Network Security

Designing and Implementing Microsoft Azure Networking Solutions

KNOWLEDGE CHEATSHEET

NETWORK PROTECTION ROLES

- **NSG** — Flow control
- **Firewall** — Inspection
- **WAF** — Web protection
- **DDoS** — Attack mitigation

- **Front Door Premium** supports managed attack and bot rule sets. Bot protection identifies bot categories; rate limiting addresses excessive request volume. Configure priorities so broad allows do not bypass intended checks.
- For **false positives**, inspect the matched rule, field and request in WAF logs. Apply a narrow exclusion or targeted rule adjustment; a 403 response alone does not identify the correct exception.

Administration, DDoS and central control

- **Azure Bastion** provides managed RDP/SSH access to VM private IPs. For dedicated public deployments, allow client HTTPS to Bastion and RDP/SSH from Bastion to targets, plus required platform rules.
- **DDoS protection** mitigates volumetric/protocol attacks; **WAF** handles web-layer attacks. Network Protection covers eligible public resources in associated VNets; IP Protection scopes protection to eligible individual public IPs.
- A **DDoS Network Protection plan** can cover multiple eligible VNets across subscriptions in the same tenant. Plan membership, protected-IP count and service eligibility drive cost; do not assume one plan per VNet.
- **Virtual Network Manager** uses network groups for central connectivity and security admin configurations. **Deny** blocks before NSGs, **Always allow** bypasses NSG evaluation, and **Allow** leaves NSG evaluation in place.
- **RBAC** controls management actions; NSGs, firewalls and WAF control traffic. Assign the narrowest role and scope; Network Contributor does not itself grant permission to assign roles or manage every resource lock.

Flow logs and Traffic Analytics

- **VNet flow logs** record network flows for new monitoring designs. **Traffic Analytics** processes eligible flow data from Storage into Log Analytics to expose top talkers, unknown traffic and external communication.
- Existing **NSG flow logs** identify rule decisions and five-tuples. Version 2 adds **B / C / E** flow states and directional packet/byte counters; sum interval counters without treating repeated state records as new sessions.
- **Activity logs** record administrative operations, not packet flows. Resource diagnostic logs capture service behavior; packet captures expose individual packets. Select the source that answers the operational question.
- Use **metrics and alerts** for tunnel health, probe health, throughput and SNAT failures. Route supported diagnostic logs to Log Analytics for KQL and to Storage for retention; configure alert action groups separately.