CHECKLIST

Azure VM Network Isolation Checklist: NSGs and Private Endpoints

A practical checklist for validating Azure virtual machine network isolation using NSGs, private endpoints, DNS, routing, and management access controls before production.

Azure VM Network Isolation Checklist

Use this checklist to validate whether an Azure virtual machine is isolated the way you expect when using NSGs and private endpoints. It is written for operational review before production use.

---

1) Define the isolation boundary

- [ ] Identify the VM?s purpose and data sensitivity.
- [ ] Document whether the VM is internet-facing, internal-only, or management-only.
- [ ] List the exact management methods required:
- [ ] SSH
- [ ] RDP
- [ ] Bastion/jump host
- [ ] Remote agent or other admin tool
- [ ] Identify every service the VM must reach:
- [ ] Storage

- [ ] Database
- [ ] Messaging/queue service
- [ ] Monitoring/telemetry
- [ ] Key management or secrets service
- [ ] Confirm which of those services support private endpoints.
- [ ] Record the minimum ports, source ranges, and destinations required.

2) Remove unnecessary public exposure

- [ ] Verify the VM does not have a public IP unless there is a documented exception.
- [ ] If a public IP is required, document the business need and compensating controls.
- [ ] Ensure inbound access is not left open to broad internet ranges.
- [ ] Confirm no temporary deployment rule was left in place after provisioning.
- [ ] Review whether any public load balancer, NAT rule, or DNS entry still exposes the VM.

3) Review NSG design for the VM and subnet

- [ ] Confirm which NSG applies to the subnet.
- [ ] Confirm whether a NIC-level NSG also exists and how it interacts with the subnet NSG.
- [ ] Verify inbound rules allow only required sources and ports.
- [ ] Verify outbound rules allow only required destinations and ports.
- [ ] Ensure administrative ports are restricted to management subnets or approved jump hosts.
- [ ] Confirm application ports are not open to unnecessary address ranges.
- [ ] Check for overly broad rules such as:
- [ ] Any source
- [ ] Any destination
- [ ] Large address-space allow rules without justification
- [ ] Wide port ranges without a documented need

- [ ] Validate rule priority order so deny/allow behavior matches intent.
- [ ] Confirm east-west traffic between subnets is intentionally allowed or denied.
- [ ] Document any rule that must remain broad and why.

4) Validate private endpoint usage

- [ ] Confirm each supported dependent service has a private endpoint.
- [ ] Verify the private endpoint is placed in the correct virtual network/subnet.
- [ ] Confirm access to the service no longer depends on a public endpoint.
- [ ] Review whether public network access on the service should be disabled.
- [ ] Confirm the VM uses the private endpoint path for service access.
- [ ] Check that application components do not hardcode public service URLs.
- [ ] Verify service permissions remain scoped to the VM?s identity or managed identity.

5) Confirm DNS resolves to private addresses

- [ ] Validate that the service name resolves to the private IP where private access is intended.
- [ ] Confirm the correct private DNS zone or equivalent internal name resolution is in place.
- [ ] Check for conflicting DNS records that could send traffic to a public endpoint.
- [ ] Verify DNS forwarding or resolver configuration works across all relevant networks.
- [ ] Test resolution from:
- [ ] The VM itself
- [ ] The management subnet
- [ ] Any application subnet that must access the service
- [ ] Confirm fallback to public DNS is not silently reintroducing exposure.

6) Validate routing and path selection

- [ ] Confirm route tables do not send private service traffic through an unintended next hop.
- [ ] Verify traffic to private endpoints stays inside the intended private path.
- [ ] Check for forced tunneling or security appliances that may break service access.
- [ ] Validate peered networks cannot bypass the intended isolation boundary.
- [ ] Confirm VPN, ExpressRoute, or other hybrid paths are accounted for in the design.

7) Test required traffic flows

- [ ] Test inbound admin access only from the approved management source.
- [ ] Test application access from intended internal sources only.
- [ ] Test service access to each private endpoint dependency.
- [ ] Test any required outbound update, telemetry, or identity traffic.
- [ ] Verify all required flows succeed without opening extra ports.
- [ ] Capture evidence of successful validation.

8) Prove denied paths stay denied

- [ ] Attempt access from an unauthorized source and confirm it fails.
- [ ] Attempt access to blocked ports and confirm it fails.
- [ ] Attempt direct access to the VM from the internet and confirm it fails.
- [ ] Attempt access to services through their public endpoint and confirm it fails if public access should be disabled.
- [ ] Confirm denied behavior remains consistent after redeployment or restart.

9) Check identity and platform permissions

- [ ] Confirm the VM uses least-privilege identity permissions.

- [ ] Review whether the VM has access to secrets, storage, or data it does not need.
- [ ] Validate that network isolation is not being relied on as the only security control.
- [ ] Confirm privileged operations require separate approval or role-based access.

10) Review operational readiness

- [ ] Document how support staff will access the VM during incidents.
- [ ] Document how patching, backup, and monitoring traffic will work.
- [ ] Confirm logging exists for NSG changes, route changes, and DNS updates.
- [ ] Define a rollback plan for incorrect network changes.
- [ ] Record the owner for each control: NSG, DNS, routing, identity, and private endpoints.
- [ ] Validate the design after any significant application or network change.

11) Production readiness sign-off

Before go-live, confirm all of the following:

- [ ] The VM has no unnecessary public exposure.
- [ ] NSGs allow only approved inbound and outbound paths.
- [ ] Private endpoints exist for all supported dependent services.
- [ ] DNS resolves private names to private IPs as intended.
- [ ] Routing preserves the required private path.
- [ ] Denied traffic has been tested and confirmed blocked.
- [ ] Identity permissions are least privilege.
- [ ] Monitoring and support procedures are documented.
- [ ] Exceptions are documented, approved, and time-bound.

Quick rule of thumb

- NSGs control who can reach the VM.
- Private endpoints control how the VM reaches supported services.
- DNS and routing determine whether private access actually works.
- Identity and permissions determine what the VM can do after it connects.

If any one of those layers is missing, the design is not fully isolated.