CHECKLIST

AWS Virtualization: Hybrid Network Segmentation Readiness Checklist

A practical, vendor-neutral checklist for validating secure hybrid network segmentation across cloud and on-premises networks before production rollout.

AWS Virtualization: Hybrid Network Segmentation Readiness Checklist

Use this checklist to validate that hybrid network segmentation is intentionally designed, observable, and ready for production traffic.

1) Define the trust boundary

- [ ] Trust zones are documented by purpose, sensitivity, and owner.
- [ ] Cloud, on-premises, partner, management, and shared service zones are identified.
- [ ] Each cross-zone connection has a clear business justification.
- [ ] Every allowed path can be described in one sentence.
- [ ] Unclear or legacy exceptions have an owner and remediation date.

2) Inventory required flows

- [ ] Source, destination, protocol, and port are documented for every approved flow.
- [ ] Inbound and outbound flows are listed separately.

- [ ] East-west flows inside the cloud environment are included.
- [ ] Management traffic is separated from application traffic.
- [ ] Logging, monitoring, patching, identity, and backup flows are documented.
- [ ] No ?temporary? flow is left undocumented.

3) Validate topology and segmentation layout

- [ ] Workloads are separated into distinct network segments or clearly defined tiers.
- [ ] Public-facing, internal, management, and shared service networks are not flattened together.
- [ ] On-premises connectivity does not expose the full cloud address space by default.
- [ ] Segmentation boundaries are aligned with administrative ownership where possible.
- [ ] Shared services are treated as protected tiers, not universally trusted infrastructure.

4) Review routing controls

- [ ] Route tables contain only the destinations needed for each segment.
- [ ] Default or broad routes have been justified and approved.
- [ ] Route propagation is reviewed and restricted where appropriate.
- [ ] Return paths are verified so traffic does not bypass intended inspection or policy points.
- [ ] Cloud-to-on-premises and on-premises-to-cloud routes are both validated.
- [ ] No route exposes a segment to networks that should remain unreachable.

5) Review stateful and subnet-level policy

- [ ] Instance-level rules permit only required sources, destinations, and ports.
- [ ] Subnet-level guardrails are in place where coarse controls are useful.

- [ ] Rules are deny-by-default unless explicitly required.
- [ ] ?Allow any? rules have been removed or tightly justified.
- [ ] Security policy does not rely on a single control layer.
- [ ] Rule intent is documented so operators can distinguish design from exception.

6) Confirm inspection and logging

- [ ] Inspection points are placed where policy enforcement is needed.
- [ ] Logs capture allowed and denied traffic where practical.
- [ ] Log retention meets operational and audit requirements.
- [ ] Flow logs, firewall logs, and routing evidence are centrally reviewable.
- [ ] Logs can be tied back to specific segments and change events.
- [ ] There is a clear process to investigate unexpected cross-boundary traffic.

7) Validate DNS and name resolution behavior

- [ ] Internal names resolve only where they are intended to be used.
- [ ] DNS does not unintentionally broaden access across segments.
- [ ] Split-horizon or equivalent resolution logic is documented if used.
- [ ] Application dependencies do not rely on ambiguous naming or shared suffixes.
- [ ] DNS changes are reviewed as part of segmentation change control.

8) Test reachability from each segment

- [ ] Connectivity tests were run from every major zone.
- [ ] Each required flow works only from the approved source segment.
- [ ] Denied flows were tested and confirmed blocked.
- [ ] Tests include both direct and return-path validation.

- [ ] Results are captured as evidence for review and audit.
- [ ] Test evidence is updated after major network changes.

9) Check hybrid-side containment

- [ ] On-premises networks are segmented with the same discipline as cloud networks.
- [ ] Internal data center networks cannot reach cloud segments broadly by default.
- [ ] Administrative subnets are isolated from application integration paths.
- [ ] Partner or third-party access is constrained separately from internal access.
- [ ] A compromise in one environment does not automatically open the other.

10) Review operational trade-offs

- [ ] Added isolation does not create brittle routing dependencies.
- [ ] The design remains understandable to operators outside the original build team.
- [ ] Centralized controls are not creating unacceptable bottlenecks.
- [ ] Distributed controls do not create policy drift or inconsistent enforcement.
- [ ] Shared services are limited to the smallest practical access surface.
- [ ] The environment balances security goals with maintainability.

11) Confirm change control and drift management

- [ ] Network policy changes require review before deployment.
- [ ] Route tables, security policy, and DNS changes are tracked together.
- [ ] Drift detection or periodic review is in place.
- [ ] Exceptions have expiration dates or scheduled revalidation.
- [ ] Ownership is clear for each segment and each rule set.
- [ ] Emergency changes have a follow-up review process.

12) Production readiness gate

- [ ] The smallest reachable surface has been documented for each segment.
- [ ] Required flows are confirmed and unnecessary flows are blocked.
- [ ] Logging and inspection provide enough evidence to troubleshoot incidents.
- [ ] Routing, policy, and DNS are consistent with the intended boundary.
- [ ] Recovery procedures exist if a rule or route change breaks connectivity.
- [ ] Security, platform, and network owners all sign off on the design.

Quick pass/fail questions

If you can answer yes to all of the following, the design is likely ready for production review:

- [ ] Do we know exactly which flows are allowed across the boundary?
- [ ] Are those flows explicit in routing and policy?
- [ ] Can we observe and prove that the rules are being used as intended?
- [ ] Have we verified that the on-premises side is segmented too?
- [ ] Can a compromised host move only within the blast radius we intended?

Evidence to collect

- [ ] Network diagrams with trust zones labeled.
- [ ] Approved flow matrix.
- [ ] Route table exports or screenshots.
- [ ] Security policy exports or rule summaries.
- [ ] Firewall and flow log samples.
- [ ] Connectivity test results.

- [ ] Change records for recent updates.
- [ ] Exception register with owners and review dates.

Final review

Before exposing mixed environments to real traffic, confirm that every allowed path is intentional, observable, and reviewable. If any segment depends on tribal knowledge, broad routing, or undocumented exceptions, pause the rollout and tighten the boundary first.