CHECKLIST

# Azure VM Backup and Recovery Best Practices Checklist

A practical checklist for validating Azure VM backup scope, retention, restore testing, and recovery controls before relying on backups in production.

## Azure VM Backup and Recovery Best Practices Checklist

Use this checklist to validate whether an Azure VM backup design is actually recoverable under real failure conditions.

---

## 1) Define the recovery target

- [ ] Identify the VM?s business purpose and owner.
- [ ] Classify the workload as critical, important, or noncritical.
- [ ] Define the acceptable recovery point objective (RPO).
- [ ] Define the acceptable recovery time objective (RTO).
- [ ] Document the main failure modes the backup must support.
- [ ] Accidental deletion
- [ ] Disk corruption
- [ ] OS failure
- [ ] Application or database corruption
- [ ] Security incident / ransomware
- [ ] Administrative access loss
- [ ] Confirm whether the VM must be restored as a full machine, as disks, or at file level.

---

## 2) Validate backup scope

- [ ] Confirm that all required disks are included in the backup policy.
- [ ] Verify whether any attached data disks are excluded.
- [ ] Check whether the guest OS state is captured in a way that supports recovery.
- [ ] Confirm whether the workload requires application-consistent recovery points.
- [ ] Confirm whether crash-consistent recovery is acceptable for this workload.
- [ ] Verify that the backup scope matches the actual restore scope needed during an incident.
- [ ] Document any components that are not protected by VM backup.

---

## 3) Set retention to match recovery needs

- [ ] Confirm short-term retention covers likely operational mistakes.
- [ ] Confirm long-term retention supports delayed-detection incidents.
- [ ] Check whether retention meets internal policy or compliance needs.
- [ ] Review the cost impact of the selected retention window.
- [ ] Verify backup retention is not just a default setting.
- [ ] Ensure retention decisions are tied to business recovery requirements.

---

## 4) Test restore paths

- [ ] Perform at least one controlled restore test for each critical VM class.
- [ ] Test full VM recovery.

- [ ] Test disk-level recovery where applicable.
- [ ] Test file-level recovery where applicable.
- [ ] Restore into an isolated or non-production network.
- [ ] Confirm the restored VM boots successfully.
- [ ] Confirm the guest OS is usable after restore.
- [ ] Confirm the application starts normally.
- [ ] Confirm application dependencies are reachable.
- [ ] Validate data integrity after recovery.
- [ ] Record restore time and compare it to the target RTO.
- [ ] Record the recovery point used and compare it to the target RPO.

---

## 5) Check network and identity dependencies

- [ ] Identify required DNS, domain, and identity dependencies.
- [ ] Verify the restored VM can authenticate if required.
- [ ] Confirm private endpoints, firewall rules, and routing dependencies.
- [ ] Verify network settings needed for the application are documented.
- [ ] Confirm the recovery network will not collide with production addressing.
- [ ] Ensure restored systems can be isolated for testing.
- [ ] Document any dependencies that must exist before recovery begins.

---

## 6) Protect the backup control plane

- [ ] Review who can manage backup policies and vault settings.
- [ ] Review who can delete backups or recovery points.
- [ ] Confirm least-privilege access for backup administrators.

- [ ] Verify deletion protections are enabled where available.
- [ ] Confirm retention changes require appropriate approval.
- [ ] Review whether the backup account is separate from day-to-day admin access.
- [ ] Assume compromised admin credentials may not be trustworthy during an incident.
- [ ] Verify backup access is protected against unauthorized changes.

---

## 7) Confirm recovery documentation exists

- [ ] Document the recovery sequence for the VM or VM class.
- [ ] List the first systems to restore during a larger incident.
- [ ] Document the order of dependency restoration.
- [ ] Identify the owner for each step in the runbook.
- [ ] Record where evidence of successful recovery will be stored.
- [ ] Note any manual steps required after restoration.
- [ ] Include rollback or re-test steps if recovery fails.

---

## 8) Distinguish backup from broader resilience controls

- [ ] Confirm backups are not being used as a substitute for hardening.
- [ ] Confirm the VM is hardened appropriately for its role.
- [ ] Review segmentation and network isolation controls.
- [ ] Review identity protections and privileged access practices.
- [ ] Determine whether automation or infrastructure-as-code should be used for rebuilds.
- [ ] Determine whether replication or application-level failover is required in addition to backup.
- [ ] Identify workloads where backup alone is not enough.

---

## 9) Decide whether the design is production-ready

Answer yes to all that apply before considering the design ready:

- [ ] Backups cover the right disks and the right recovery scope.
- [ ] Retention matches the workload?s recovery needs.
- [ ] Restore tests have been completed successfully.
- [ ] Recovery was tested in an isolated environment.
- [ ] Network and identity dependencies were verified.
- [ ] Access controls protect backup data and recovery points.
- [ ] Recovery runbooks exist and are owned.
- [ ] The team knows how to recover under pressure.

If any answer is no, revise the design and retest.

---

## Quick operational review

Before each production review, confirm the following:

- [ ] The VM still matches its original criticality classification.
- [ ] The workload has not changed in a way that affects recovery.
- [ ] New disks or dependencies have not been added without backup review.
- [ ] Restore tests are current.
- [ ] Access control reviews are current.
- [ ] Retention is still appropriate.
- [ ] The runbook reflects the current architecture.

---

## Incident-day recovery checklist

When recovery is needed, verify:

- [ ] The incident scope is understood.
- [ ] The correct recovery point is selected.
- [ ] The restore target is isolated or safe.
- [ ] The restore result is validated before production use.
- [ ] The application is functioning, not just the VM.
- [ ] The recovery is documented with time, owner, and outcome.

---

## Final note

A backup policy is only useful if the restore works when the workload actually fails. Treat backup as part of the recovery design, test it regularly, and verify the result under realistic conditions.