CHECKLIST

Hyper-V Replica Disaster Recovery Failover Checklist

A practical pre-production and operational checklist for validating Hyper-V Replica as a disaster recovery failover option. Covers readiness checks, security, testing, failover execution, and post-failover recovery for system and security teams.

Hyper-V Replica Disaster Recovery Failover Checklist

Use this checklist to verify whether Hyper-V Replica is ready for production use and to guide planned or unplanned failover operations.

> How to use: Mark each item complete before production rollout, before a test failover, and again after any real failover event.

---

1) Define the recovery objective

- [ ] Identify the business services protected by replication.
- [ ] Confirm the acceptable RPO for each workload.
- [ ] Confirm the acceptable RTO for each workload.
- [ ] Document whether the environment can tolerate asynchronous replication and a short data-loss window.
- [ ] Identify which VMs are suitable for replica-based recovery and which require a different DR design.
- [ ] Confirm whether failover will be planned, unplanned, or both.

---

2) Validate workload suitability

- [ ] Confirm the VM is a good candidate for VM-level recovery.
- [ ] Review application dependencies such as DNS, Active Directory, certificates, databases, and middleware.
- [ ] Identify workloads that require special handling for transaction consistency.
- [ ] Verify whether guest OS shutdown behavior is clean and predictable.
- [ ] Confirm that recovery of the VM alone is enough to restore the service.
- [ ] Document any services that must be started in a specific order after failover.

---

3) Verify network and host connectivity

- [ ] Confirm the primary host can reach the replica host over the intended replication path.
- [ ] Confirm network latency and stability are acceptable for the chosen replication frequency.
- [ ] Verify firewall rules allow only the required replication traffic.
- [ ] Confirm routing works in both directions if reverse replication will be used later.
- [ ] Test connectivity from the administrative network to both sites.
- [ ] Confirm there is no dependency on an undocumented or fragile network path.

---

4) Review authentication and access control

- [ ] Confirm only approved hosts are allowed to participate in replication.
- [ ] Restrict administrative access to operators who need to manage replication or initiate failover.

- [ ] Verify service accounts, if used, are scoped minimally.
- [ ] Ensure privileged access to the recovery site is audited.
- [ ] Confirm failover actions require the right level of approval in your process.
- [ ] Review local and domain-based permissions for the recovery environment.

---

5) Check storage readiness at the replica site

- [ ] Confirm enough capacity exists for current replicas and growth.
- [ ] Reserve space for recovery points and longer outage scenarios.
- [ ] Verify the replica storage can handle boot-time and post-failover load.
- [ ] Confirm performance is adequate for the protected workload.
- [ ] Document storage ownership and operational responsibility at the recovery site.
- [ ] Verify there is enough capacity to run the recovered VMs simultaneously if needed.

---

6) Establish replication policy

- [ ] Set a replication frequency aligned to the business RPO.
- [ ] Confirm the retention of recovery points matches operational needs.
- [ ] Decide whether application-consistent recovery points are required.
- [ ] Verify the initial replication method is appropriate for the environment.
- [ ] Confirm the replica relationship includes the correct VM disks and settings.
- [ ] Document any exclusions or special settings.

---

7) Validate recovery point quality

- [ ] Confirm the latest recovery point is recent enough for business tolerance.
- [ ] Verify that earlier recovery points are available if the latest point is unusable.
- [ ] Test restore behavior using a non-production failover.
- [ ] Confirm guest shutdown state, memory state, and application state behave as expected.
- [ ] Identify what level of data loss is acceptable in a real incident.
- [ ] Document how to choose between the newest point and an older, safer point.

---

8) Test failover before production

- [ ] Perform a test failover before relying on the replica in production.
- [ ] Confirm the recovered VM boots successfully.
- [ ] Verify network identity, IP addressing, and name resolution.
- [ ] Confirm domain authentication works from the recovery site.
- [ ] Validate application startup order and service readiness.
- [ ] Check that dependent systems can reach the recovered service.
- [ ] Record the test results and any remediation required.

---

9) Prepare the operational runbook

- [ ] Write step-by-step instructions for planned failover.
- [ ] Write step-by-step instructions for unplanned failover.
- [ ] Define who declares a site incident and who approves failover.
- [ ] Identify who communicates with application owners and end users.
- [ ] Define the validation checklist used after the VM starts.
- [ ] Document how to reverse replication or re-protect the VM after recovery.
- [ ] Store the runbook where on-call staff can access it during an outage.

---

10) Confirm security controls at the recovery site

- [ ] Verify the recovery site uses approved hardening standards.
- [ ] Confirm recovery VMs inherit the expected baseline security settings.
- [ ] Review whether the recovered guest OS is protected by the same controls as production.
- [ ] Confirm logging, monitoring, and alerting work at the replica site.
- [ ] Validate antivirus, EDR, and patch management assumptions for recovered systems.
- [ ] Ensure credentials and secrets required by the workload are available and protected.

---

11) Plan for application validation after failover

- [ ] Verify the application can start without manual recovery steps beyond the runbook.
- [ ] Confirm database consistency checks are available if needed.
- [ ] Validate user login, service-to-service authentication, and API connectivity.
- [ ] Check certificates, trusts, and endpoint registrations.
- [ ] Confirm scheduled jobs, integrations, and background services are functioning.
- [ ] Define who signs off that the service is fully restored.

---

12) Verify user and service redirection

- [ ] Confirm how users will be redirected to the recovery site.
- [ ] Update DNS, load balancers, routing, or service endpoints as required.
- [ ] Confirm TTL values support timely cutover.
- [ ] Validate external and internal access paths.

- [ ] Check that dependent services have switched to the recovered workload.
- [ ] Ensure redirection is reversible when the primary site returns.

---

13) Prepare for reverse replication and failback

- [ ] Confirm the primary site can be rebuilt or repaired.
- [ ] Verify reverse replication procedures are documented.
- [ ] Decide whether failback will be immediate or scheduled.
- [ ] Check that synchronization back to the primary site will not disrupt the restored service.
- [ ] Validate there is enough bandwidth to return changes to the primary location.
- [ ] Confirm the failback window is approved by stakeholders.

---

14) Set monitoring and alerting

- [ ] Monitor replication health continuously.
- [ ] Alert on replication failures, excessive lag, or stalled updates.
- [ ] Monitor storage capacity at both sites.
- [ ] Alert on authentication, firewall, or connectivity issues affecting replication.
- [ ] Track failover test results and remediation items.
- [ ] Confirm on-call staff know which alerts indicate loss of recovery readiness.

---

15) Make the final production decision

- [ ] The business accepts asynchronous recovery.

- [ ] The workloads fit VM-level replication.
- [ ] The replica site has sufficient capacity and operational ownership.
- [ ] Connectivity, access control, and security controls are validated.
- [ ] Test failover succeeded and application validation passed.
- [ ] The runbook has been reviewed by system and security teams.
- [ ] The environment is ready for production use.

Failover execution checklist

Planned failover

- [ ] Notify stakeholders and obtain approval.
- [ ] Pause or drain application traffic if needed.
- [ ] Confirm the primary VM is in a safe state for transition.
- [ ] Verify the latest replica recovery point.
- [ ] Execute planned failover.
- [ ] Validate boot, network, identity, and application readiness.
- [ ] Redirect users only after validation completes.
- [ ] Record the time of failover and any data-loss window.

Unplanned failover

- [ ] Confirm the primary site is unavailable.
- [ ] Declare incident status and assign an incident owner.
- [ ] Select the most appropriate available recovery point.
- [ ] Execute unplanned failover.

- [ ] Validate the recovered VM and dependent services.
- [ ] Redirect traffic only when the service is confirmed usable.
- [ ] Communicate recovery status to stakeholders.
- [ ] Capture lessons learned after the incident.

Post-failover validation checklist

- [ ] The VM boots successfully.
- [ ] The guest OS is reachable.
- [ ] Network settings are correct.
- [ ] Name resolution works.
- [ ] Authentication succeeds.
- [ ] Application services are online.
- [ ] Data appears consistent enough for the selected recovery point.
- [ ] Users can access the service.
- [ ] Logging and monitoring are active.
- [ ] The environment is ready for reverse replication or failback.

Quick decision summary

Hyper-V Replica is a strong fit when you need:

- [ ] A lower-cost disaster recovery option.
- [ ] Asynchronous replication for a limited number of VMs.
- [ ] Operator-driven failover.
- [ ] A practical recovery process that can be tested and validated.

It is a weaker fit when you need:

- [ ] Zero-data-loss assumptions.
- [ ] Immediate service continuity.
- [ ] Automatic site-wide recovery.
- [ ] Storage-level orchestration across many workloads.

Final note

Replication is not the same as readiness. A replica is only useful if the failover path, security controls, storage, and application validation all work together when the primary site is unavailable.