CHECKLIST

Active Directory Least Privilege Hardening Checklist

A practical checklist for validating delegated administration, scoped permissions, service account limits, and privilege boundaries before production use in Active Directory.

Active Directory Least Privilege Hardening Checklist

Use this checklist to harden Active Directory with least privilege access controls. Mark each item complete before promoting delegation or privilege changes into production.

1) Define the administrative task and scope

- [ ] Identify the exact business task to be performed.
- [ ] Identify the object types involved: users, groups, computers, OUs, attributes, or GPOs.
- [ ] Define the smallest scope required: specific OU, region, department, or application boundary.
- [ ] Document which tasks are in scope and which are explicitly out of scope.
- [ ] Confirm the task can be completed without broader directory-wide privileges.

2) Separate identities and privileged use

- [ ] Ensure administrators have separate accounts for everyday work and privileged tasks.
- [ ] Confirm privileged accounts are not used for email, web browsing, or general workstation activity.

- [ ] Verify privileged accounts are only used from approved admin workstations or equivalent hardened systems.
- [ ] Remove shared admin accounts unless there is a documented, controlled exception.
- [ ] Review break-glass accounts and confirm they are tightly controlled, monitored, and tested.

3) Design role-based delegation

- [ ] Assign permissions to roles or groups, not directly to individual users.
- [ ] Map each role to a narrow set of tasks.
- [ ] Avoid built-in broad administrative groups unless they are truly required.
- [ ] Use separate roles for help desk, server admins, workstation admins, and directory admins.
- [ ] Review whether any role combines unrelated functions that increase blast radius.

4) Scope permissions to the smallest feasible boundary

- [ ] Delegate rights at the OU or object level instead of the domain level whenever possible.
- [ ] Restrict delegation to the minimum set of objects needed.
- [ ] Limit help desk access to specific OUs or regions if applicable.
- [ ] Restrict rights to the minimum required attributes when attribute editing is needed.
- [ ] Confirm no inherited permissions unintentionally expand access beyond the intended scope.

5) Validate common delegated tasks

- [ ] Test password reset permissions for the intended users only.
- [ ] Test account unlock permissions for the intended users only.

- [ ] Test group membership changes only where explicitly allowed.
- [ ] Test computer account creation only if the business process requires it.
- [ ] Test attribute updates only for approved fields.
- [ ] Test the negative cases: confirm users cannot perform tasks outside their scope.

6) Check for privilege escalation paths

- [ ] Review whether delegated users can modify nested groups that grant broader access.
- [ ] Confirm delegated users cannot alter permissions or ACLs unless explicitly required.
- [ ] Verify users cannot create new objects in locations that could become persistence paths.
- [ ] Check whether delegation enables indirect access through group nesting or inherited permissions.
- [ ] Validate that service accounts cannot be repurposed for interactive administration.

7) Constrain service accounts and automation

- [ ] Remove interactive logon where possible for service accounts.
- [ ] Limit service accounts to the exact systems they must reach.
- [ ] Grant only the object-level permissions the service actually needs.
- [ ] Avoid shared service credentials across unrelated systems or functions.
- [ ] Document credential ownership, rotation, and recovery procedures.
- [ ] Verify automation accounts cannot modify objects outside their defined workflow.

8) Apply administrative tiering or equivalent separation

- [ ] Separate low-trust endpoint administration from directory administration.

- [ ] Ensure the same admin account is not used across mixed-trust management zones.
- [ ] Keep directory-control credentials off standard user workstations.
- [ ] Restrict high-trust administrative activities to approved management systems.
- [ ] Review whether tier boundaries are documented and actually enforced.

9) Control elevation

- [ ] Make elevated access temporary whenever possible.
- [ ] Require approval or justification for high-risk privileges.
- [ ] Confirm elevation expires automatically or is removed after the task is complete.
- [ ] Verify the elevation workflow is logged end to end.
- [ ] Check that emergency elevation paths are documented and tested.

10) Verify logging and monitoring

- [ ] Confirm directory changes are logged at the level needed for review.
- [ ] Ensure delegation changes themselves are auditable.
- [ ] Review alerts for privileged group membership changes.
- [ ] Review alerts for permission changes, ACL edits, and new object creation.
- [ ] Confirm logs can support incident reconstruction for delegated actions.
- [ ] Validate that normal administrative activity is distinguishable from suspicious activity.

11) Review change control and rollback

- [ ] Document the intended permission model before deployment.
- [ ] Capture a baseline of current delegation and privileged group membership.
- [ ] Test rollback steps before broad rollout.

- [ ] Verify permission changes can be reverted cleanly.
- [ ] Include delegation reviews in regular change management.

12) Check for permission drift

- [ ] Review delegated rights after major directory changes.
- [ ] Reconfirm scope after OU restructuring, mergers, or role changes.
- [ ] Look for temporary permissions that have become permanent.
- [ ] Review group nesting for unintended privilege accumulation.
- [ ] Remove redundant access that is no longer required.

13) Validate supportability

- [ ] Confirm the delegated model still allows the team to do its job.
- [ ] Verify help desk and operations teams know the approved process for requesting additional access.
- [ ] Ensure temporary access requests are tracked and expired.
- [ ] Confirm documentation is clear enough for on-call use.
- [ ] Check whether over-restrictive delegation is causing undocumented workarounds.

14) Red flags to fix before production

- [ ] A role can manage objects outside its intended OU.
- [ ] A service account has broader rights than the workflow requires.
- [ ] A temporary elevated account is being used as a long-term workaround.
- [ ] Built-in broad admin roles are used for routine tasks.
- [ ] Delegation has been tested functionally but not with negative cases.
- [ ] No one can clearly explain the scope of a privilege in one sentence.

15) Production readiness checks

- [ ] Every delegated role has an owner.
- [ ] Every privileged path has a documented scope.
- [ ] Every elevated workflow has logging and review.
- [ ] Every service account has a documented purpose and boundary.
- [ ] Every exception has an expiration date.
- [ ] A periodic review cadence is in place for permissions and group membership.

Final go-live question

If you cannot explain the privilege in one sentence that names the object type, the action, and the boundary, the access is probably too broad.

Quick sign-off

- [ ] Scope defined
- [ ] Role-based delegation implemented
- [ ] Negative testing completed
- [ ] Logging verified
- [ ] Rollback confirmed
- [ ] Exception review scheduled

Notes

Use this checklist alongside your standard change management process. Least privilege is not complete until it has been tested, monitored, and re-validated after change.