AWS Cost Waste Checklist

for Small Teams

Find waste, understand the bill, and reduce spend safely without breaking production.

Use this as a review queue, not a delete list. Every row is a possible waste signal that needs ownership and verification before action.

Verify first, then act.

Confirm owner, usage, dependencies, backups, retention needs, DNS references, and rollback options before changing resources.

What is inside

- 35 practical review checks across 14 AWS service areas
- 7 categories with risk, priority, owner, status, savings, and next-action fields
- Usage workflow, safety rules, best first review path, and quick wins
- CSV-friendly structure for Google Sheets or spreadsheet review

Educational resource

Not affiliated with AWS. Not production remediation guidance. Use your team's normal review, approval, change window, and rollback process before changing resources.

How to use this checklist

Start with spend, then move toward ownership and verification. The goal is lower spend without reckless cleanup.

1. Start with the highest-cost services.
2. Identify possible waste signals.
3. Assign an owner.
4. Verify usage, dependencies, backups, retention needs, and business purpose.
5. Estimate monthly savings.
6. Decide whether to keep, monitor, resize, schedule cleanup, or investigate further.
7. Make changes only through the team's normal change process.
8. Keep notes so the same waste does not return.

Find waste -> Verify -> Act safely

01

Find waste

Identify possible cost waste signals using Cost Explorer, tags, usage metrics, and service-specific checks.

02

Verify

Confirm owner, usage, dependencies, backups, retention requirements, DNS references, and rollback options.

03

Act safely

Choose the safest next step: keep, monitor, resize, schedule cleanup, or investigate through the normal change process.

Suggested review status values

Not reviewed	Needs owner	Needs usage review	Verified waste
Keep	Cleanup scheduled	Completed	

Best first review path

If this is your first AWS cost review, start with visibility and ownership before cleanup.

1. Check AWS Budgets and forecast alerts.
2. Review Cost Explorer for the highest-cost services.
3. Look for unattached EBS volumes.
4. Check idle Elastic IPs.
5. Review CloudWatch log groups with no retention.
6. Review old snapshots with no clear owner.
7. Check NAT Gateway usage types and data processing charges.
8. Find untagged high-cost resources.

Start with visibility and ownership before cleanup. The goal is to identify likely waste, assign owners, and verify safely before making changes.

Quick wins to review first

- Idle Elastic IPs
- Unattached EBS volumes
- CloudWatch log groups with no retention
- Old snapshots with no owner
- Missing budgets or forecast alerts
- Untagged high-cost resources
- Old ECR images
- Dev/test resources running full-time

Safety rules

Every row is a possible waste signal. Confirm ownership, dependencies, and business purpose before deleting, resizing, or changing anything.

- This is a review queue, not a delete list.
- Unused does not always mean safe to delete.
- High cost does not automatically mean waste.
- Verify ownership before changing anything.
- Check dependencies, backups, DNS, retention requirements, and rollback options.
- Use change windows for risky changes.
- When unsure, mark as Needs owner or Needs usage review.

CSV workflow fields

Use the CSV as the working tracker. Keep Owner, Review status, Estimated monthly savings, Next action, and Notes current during review.

- Category
- AWS service
- What to check
- Why it matters
- How to verify
- Risk level
- Priority
- Safe to delete or change?
- Do not change until verified
- Owner
- Review status
- Estimated monthly savings
- Next action
- Notes

Service-by-service checklist

Use the CSV for owner, status, savings, and notes. Use this PDF for review prompts.

CMP **1. Compute / EC2** — Risk: High | Priority: High

Idle or underused instances

Why: Instances with low CPU, network, or connection activity may be overprovisioned or forgotten.
Verify: Review Cost Explorer, CloudWatch metrics, instance tags, schedules, and application ownership.
Next: Assign owner and collect 14-30 days of usage data

CMP **2. Compute / EC2** — Risk: Medium | Priority: High

Stopped instances with attached EBS volumes

Why: Stopped EC2 instances do not run compute charges, but attached EBS volumes can keep billing.
Verify: List stopped instances, attached volumes, volume size and type, snapshots, tags, and start history.
Next: Review attached volumes and decide keep, snapshot, resize, or remove

CMP **3. Compute / EC2 AMIs** — Risk: Medium | Priority: Medium

Old AMIs with unclear ownership

Why: Private AMIs can keep related snapshots and storage around after the source system is gone.
Verify: Review AMI age, launch history, descriptions, linked snapshots, image users, and owner tags.
Next: Map AMI users before deregistering or deleting snapshots

CMP **4. Compute / EC2** — Risk: Medium | Priority: Medium

Dev/test instances running outside working hours

Why: Non-production instances that run all night or all weekend can create steady avoidable spend.
Verify: Compare tags, schedules, CloudWatch metrics, and team working hours.
Next: Propose a schedule or stop/start automation for owner review

GOV **5. Governance / EC2** — Risk: Low | Priority: Medium

Instances missing owner or environment tags

Why: Unowned compute is hard to verify and tends to stay running longer than needed.
Verify: Find instances missing owner, environment, application, or service tags.
Next: Add owner and environment tags before cleanup decisions

Service-by-service checklist

Use the CSV for owner, status, savings, and notes. Use this PDF for review prompts.

STO **6. Storage / EBS Volumes** — Risk: Medium | Priority: High

Unattached EBS volumes

Why: Detached volumes can keep billing even when no instance is using them.
Verify: Check volume state, age, size, type, tags, recent snapshots, and owner.
Next: Confirm owner and snapshot need before scheduling cleanup

STO **7. Storage / EBS Volumes** — Risk: Medium | Priority: Medium

Oversized volumes or gp2 volumes for review

Why: Oversized volumes and older volume types may create avoidable storage spend.
Verify: Review volume size, used filesystem space, volume type, IOPS, throughput, and current regional cost details.
Next: Flag resize or volume-type review with the owner

STO **8. Storage / EBS Volumes** — Risk: Medium | Priority: Medium

Volumes attached to stopped, rarely used, or unclear resources

Why: Attached storage can remain expensive when the related compute is stopped, rarely used, or missing context.
Verify: Join EC2 state, volume attachments, owner tags, filesystem usage, and CloudWatch activity.
Next: Review with the instance owner and decide keep, snapshot, resize, or remove

STO **9. Storage / EBS Snapshots** — Risk: Medium | Priority: High

Old manual snapshots outside retention policy

Why: Manual snapshots can accumulate after incidents, migrations, or one-time backups.
Verify: Review age, description, source volume, creator, backup policy, and restore purpose.
Next: Group snapshots by owner and retention decision

STO **10. Storage / EBS Snapshots** — Risk: Medium | Priority: Medium

Snapshots with unclear ownership or restore purpose

Why: Snapshots without a clear owner are hard to classify as backup, migration, or waste.
Verify: Check tags, descriptions, related volumes, AMIs, backup jobs, and change history.
Next: Mark Needs owner until someone can explain the snapshot

Service-by-service checklist

Use the CSV for owner, status, savings, and notes. Use this PDF for review prompts.

STO **11. Storage / EBS Snapshots** — Risk: High | Priority: High

Snapshots tied to AMIs

Why: Deleting snapshots used by AMIs can break launches or recovery workflows.
Verify: Map snapshot IDs to AMIs before deleting or changing retention.
Next: Review AMI dependencies before any snapshot cleanup

NET **12. Networking / Elastic IP** — Risk: Low | Priority: High

Unattached Elastic IPs

Why: Elastic IPs that are not associated with resources can create avoidable charges.
Verify: Check association status, tags, allocation age, DNS references, and recent change history.
Next: Release only after owner confirms it is not reserved

NET **13. Networking / Elastic IP** — Risk: Medium | Priority: Medium

Elastic IPs reserved for unclear reasons

Why: Reserved addresses may support DNS, allowlists, failover, or old architecture decisions.
Verify: Review tags, Route 53 records, firewall allowlists, partner integrations, and runbooks.
Next: Document dependency or schedule release with owner approval

NET **14. Networking / Load Balancers** — Risk: High | Priority: High

Load balancers with little traffic or no healthy targets

Why: Load balancers can keep hourly charges even when traffic is low or targets are unhealthy.
Verify: Review request counts, target health, listeners, target groups, access logs, and tags.
Next: Confirm ownership and traffic path before cleanup

NET **15. Networking / Load Balancers** — Risk: Medium | Priority: Medium

Load balancers tied to old test environments

Why: Test or preview environments can leave load balancers, target groups, and DNS records behind.
Verify: Compare environment tags, DNS records, target groups, certificates, and recent deployments.
Next: Ask the environment owner whether the stack is still needed

Service-by-service checklist

Use the CSV for owner, status, savings, and notes. Use this PDF for review prompts.

NET **16. Networking / NAT Gateway** — Risk: High | Priority: High

NAT Gateways with steady high data processing charges

Why: NAT Gateway hourly and data processing charges can become a major cost driver.
Verify: Review Cost Explorer usage types, VPC flow logs if available, route tables, and subnet mappings.
Next: Map traffic sources before changing network paths

NET **17. Networking / NAT Gateway** — Risk: High | Priority: High

Private subnet traffic routed through NAT by default

Why: Default routes can send traffic through NAT even when a service-specific path may be better.
Verify: Inspect private route tables, common AWS service calls, endpoint availability, and traffic patterns.
Next: Identify workloads that may benefit from VPC endpoints

NET **18. Networking / NAT Gateway** — Risk: High | Priority: Medium

NAT Gateways in low-traffic or multi-AZ patterns

Why: Unused environments and multi-AZ NAT designs can create steady hourly charges.
Verify: Compare NAT gateways by AZ, environment, data volume, route tables, and resilience requirements.
Next: Review whether each NAT gateway has a current business purpose

DB **19. Database / RDS** — Risk: High | Priority: High

Idle or low-connection DB instances

Why: Databases with little usage can still carry instance, storage, backup, and license-related costs.
Verify: Review connections, CPU, memory, I/O, storage, backups, replicas, and application ownership.
Next: Ask owner whether to keep, downsize, pause, or retire

DB **20. Database / RDS** — Risk: High | Priority: Medium

Oversized instance classes

Why: RDS class choices can dominate a small AWS bill when sized for peaks that no longer exist.
Verify: Review performance metrics, connection patterns, maintenance windows, and right-sizing options.
Next: Create a right-sizing review ticket for the database owner

Service-by-service checklist

Use the CSV for owner, status, savings, and notes. Use this PDF for review prompts.

DB **21. Database / RDS** — Risk: Medium | Priority: Medium

Storage growth that has not been reviewed

Why: Storage, backups, and log growth can keep increasing after usage patterns change.
Verify: Review allocated storage, free storage, autoscaling, backup retention, logs, and growth trend.
Next: Review storage growth and retention settings with the owner

DB **22. Database / RDS** — Risk: Medium | Priority: Medium

Old manual RDS snapshots

Why: Manual snapshots can remain after migrations, one-time backups, or incident response work.
Verify: Review snapshot age, tags, source DB, restore purpose, retention policy, and owner.
Next: Group snapshots by owner and retention decision

DB **23. Database / RDS** — Risk: High | Priority: Medium

Dev/test databases or replicas running full-time

Why: Non-production databases, Multi-AZ, and read replicas need current business justification.
Verify: Review environment tags, instance schedules, Multi-AZ setting, replicas, traffic, and owner notes.
Next: Confirm whether full-time runtime or replica capacity is still needed

STO **24. Storage / S3** — Risk: Medium | Priority: High

Buckets without lifecycle rules or storage-class review

Why: Old objects may stay in expensive classes when access patterns have changed.
Verify: Review lifecycle policies, storage class analysis, object age, access patterns, and restore needs.
Next: Draft lifecycle candidates for owner review

STO **25. Storage / S3** — Risk: Medium | Priority: High

Old object versions and noncurrent versions accumulating

Why: Versioned buckets can grow quietly when old object versions are never expired.
Verify: Check versioning, noncurrent object counts, lifecycle rules, delete markers, and retention needs.
Next: Review noncurrent version expiration with the bucket owner

Service-by-service checklist

Use the CSV for owner, status, savings, and notes. Use this PDF for review prompts.

STO **26. Storage / S3** — Risk: Low | Priority: Medium

Incomplete multipart uploads

Why: Incomplete multipart uploads can leave storage behind after failed or abandoned uploads.
Verify: List incomplete multipart uploads and review bucket lifecycle cleanup settings.
Next: Add or review lifecycle cleanup for incomplete uploads

GOV **27. Governance / S3** — Risk: Low | Priority: Medium

Buckets missing owner or environment tags

Why: Unowned buckets make retention, lifecycle, and access-pattern decisions harder.
Verify: Find buckets missing owner, environment, application, data classification, or service tags.
Next: Add owner and environment tags before lifecycle changes

OBS **28. Observability / CloudWatch Logs** — Risk: Medium | Priority: High

Log groups with Never Expire retention

Why: Indefinite retention can grow log storage costs long after a service changes or retires.
Verify: List log groups, retention settings, stored bytes, ingestion, owner, and compliance needs.
Next: Set an agreed retention period for owner-approved log groups

OBS **29. Observability / CloudWatch Logs** — Risk: Medium | Priority: High

High-ingestion or verbose log groups

Why: Debug or overly verbose logs can create steady ingestion and storage costs.
Verify: Review incoming bytes, top log groups, application log level, deployments, and owner.
Next: Ask owner whether debug or verbose logs are still intentional

OBS **30. Observability / CloudWatch Logs** — Risk: Medium | Priority: Medium

Old log groups from retired services

Why: Log groups can remain after services, environments, or test stacks are gone.
Verify: Compare log group names, tags, last ingestion time, related services, and retention rules.
Next: Confirm service status before changing retention or deleting logs

Service-by-service checklist

Use the CSV for owner, status, savings, and notes. Use this PDF for review prompts.

CTR **31. Containers / ECR** — Risk: Medium | Priority: Medium

Old or untagged container images

Why: Container images can accumulate across deployments, branches, and retired services.
Verify: Review image age, tags, pull history, repository owner, deployment references, and rollback needs.
Next: Define image retention rules with the service owner

CTR **32. Containers / ECR** — Risk: Low | Priority: Medium

Repositories without lifecycle policies

Why: Repositories from active or retired services can grow if image cleanup is never configured.
Verify: List repositories, lifecycle policies, image counts, last push and pull activity, and owner tags.
Next: Add lifecycle policy candidates for owner review

GOV **33. Governance / Budgets and Alerts** — Risk: Low | Priority: High

Missing budgets, forecast alerts, or review owner

Why: Teams miss cost changes when alerts are absent, stale, or no one owns the review.
Verify: Check AWS Budgets, forecast alerts, thresholds, recipients, linked accounts, service views, and monthly process.
Next: Create or update budgets, forecast alerts, recipients, and monthly review owner

GOV **34. Governance / Tags and Cost Allocation** — Risk: Low | Priority: High

Untagged high-cost resources or inconsistent tags

Why: Missing owner, environment, and service tags make cost allocation and cleanup slower.
Verify: Group spend by tag coverage, find missing owner/environment/application tags, and check tag consistency.
Next: Fix high-cost untagged resources and activate needed cost allocation tags

NET **35. Networking / Data Transfer** — Risk: High | Priority: Medium

Cross-AZ, internet egress, or NAT-related transfer surprises

Why: Data transfer can appear as a bill surprise when services talk across AZs, regions, or the internet.
Verify: Review Cost Explorer transfer usage, NAT charges, architecture diagrams, flow logs, and service placement.
Next: Map top transfer paths before proposing architecture changes

Keep the review safe

The useful output of this checklist is a better review queue, not a pile of risky cleanup tasks.

Before changing anything, confirm:

- The resource owner and business purpose.
- Current usage and expected future usage.
- Dependencies, DNS references, backups, retention requirements, and rollback options.
- The team's normal approval path, change window, and communication plan.

More notes and guides

For more practical AWS cost optimization notes, guides, and checklists, visit:

cloudcostclinic.com

Want this automated?

Cloud Cost Clinic is building a free read-only AWS Waste Scan. Join the early preview at cloudcostclinic.com/#scanner-preview.

Final reminder