PLAYBOOK

A Cloud FinOps 90-Day Runbook for AWS, Azure, and GCP

A week-by-week runbook for the first 90 days of a FinOps engagement. Quick wins, governance, and operating model rollout. Written as a runbook, not as marketing copy.

By Vishal Shukla · VP of Technology, ViitorCloud Enterprise

Why This Exists

Industry research in 2026 puts cloud waste between 25 and 35 percent of total cloud spend for the average enterprise. Idle resources, over-provisioned instances, orphaned storage, dev environments running 24/7, and commitments that no longer match the workload they were bought for. A well-run FinOps engagement closes most of that gap within twelve months. Realised savings typically land in the 20 to 35 percent range during the first quarter post-implementation. Best-case engagements hit 30 to 50 percent.

This runbook is the first 90 days of that engagement. Three phases, thirteen weeks, sized for AWS, Azure, and GCP. The structure has been tested across engagements in BFSI, healthcare, and the public sector, where the cost question is rarely "can we save money" and almost always "can we save money without breaking anything."

How to Use This

This is a runbook, not a discussion. The phases run in sequence. Activities inside each phase run in parallel. The runbook assumes a delivery team of one FinOps lead, two cloud engineers, and a named finance partner from the client side. It assumes the program owner has executive sponsorship and a signed savings target.

Specific savings ranges are listed where they are known. Where the range is wide, the wide range reflects the actual variance we see across estates. Do not read the higher end as a commitment. Read it as the upper bound of what is achievable when the estate is in good condition and the team is motivated.

Phase 1. Quick Wins, Weeks 1 to 4

The goal of Phase 1 is to land a defensible number of dollars saved inside the first month. Visible savings buy the program time, credibility, and the budget for Phase 2 and Phase 3.

WEEK 1. ESTATE AUDIT AND BASELINE

Objective. Establish a clean baseline of the cloud spend you will be optimising against. No optimisation work yet.

Activities.

- Ingest cost data from AWS Cost Explorer plus Cost and Usage Reports, Azure Cost Management, and GCP Billing Reports into a single warehouse or BI tool.
- Tag the current state. Document which accounts, subscriptions, and projects are in scope. Document the executive sponsor and the named finance partner.
- Pull 30 days of utilisation data from AWS CloudWatch, Azure Monitor, and Google Cloud Monitoring. Anything less than 30 days will miss monthly batch jobs, end-of-quarter spikes, and patterns that do not show up in a two-week window.
- Identify the top 20 cost drivers by service. This is almost always 80 percent of the bill.

Deliverables. A baseline cost report. A signed savings target written into the engagement SOW. A top 20 cost drivers list with named owners.

Expected Savings This Week. Zero. This week pays for itself through what it makes possible in weeks 2 through 4.

WEEK 2. IDLE AND ORPHANED RESOURCE CLEANUP

Objective. Eliminate spend on resources that are doing no work.

Activities.

- Identify idle compute. EC2 instances with under 5 percent CPU utilisation for 14 consecutive days. Azure VMs in the same condition. GCE instances similarly.
- Identify orphaned storage. Unattached EBS volumes, unattached managed disks in Azure, persistent disks not in use in GCP. Snapshots older than the retention policy.
- Identify zombie load balancers, idle databases, unused NAT gateways, and unattached static IPs.
- Apply the standard cleanup decision tree. Stop, snapshot and delete, or migrate to a cheaper tier.

Deliverables. A documented cleanup list with savings per resource. Executed cleanups. A residual list of resources flagged for further investigation.

Expected Savings. Typically 5 to 12 percent of in-scope cloud spend, banked against the savings target on the SOW.

WEEK 3. RIGHTSIZING ON COMPUTE

Objective. Resize over-provisioned compute to actual usage.

Activities.

- Run AWS Compute Optimizer, Azure Advisor, and GCP Recommender against the 30 days of utilisation data collected in Week 1.
- Cross-check recommendations against the business calendar. Avoid downsizing right before end-of-quarter or seasonal spikes.
- For top recommendations, execute rightsizing changes in dev and test first. Production changes go through normal change-management approval.

- For Kubernetes workloads, run rightsizing on pod requests and limits. EKS, AKS, and GKE all have native tooling. Kubecost is the cross-cloud option.

Deliverables. A rightsizing changelog with old size, new size, and savings per change. A list of recommendations held back, with the reason for each.

Expected Savings. 15 to 25 percent on the compute slice of in-scope spend. Best-case engagements with severely over-provisioned estates have hit 60 percent on compute, but plan for the middle of the range.

WEEK 4. COMMITMENT OPTIMISATION

Objective. Optimise the existing commitment posture (Reserved Instances, Savings Plans, CUDs) and identify the coverage gap.

Activities.

- Audit existing AWS Reserved Instances and Savings Plans for utilisation. Anything below 90 percent utilisation is a candidate for modification, sale where allowed, or expiry without renewal.
- Audit Azure Reservations and Savings Plans for compute similarly.
- Audit GCP Committed Use Discounts similarly.
- Model the new commitment posture against the rightsized workloads from Week 3, not against the pre-rightsizing baseline. Buying commitments at the old size locks in inefficiency you just removed.
- For predictable, stable workloads, recommend 1-year no-upfront commitments first. Reserve 3-year commitments for workloads you would bet on for three years.

Deliverables. A commitment audit report. A recommended commitment posture with a target coverage percentage. A defensible target is 70 to 80 percent of the rightsized baseline. A purchase plan, scheduled for execution in Phase 2.

Expected Savings. 30 to 60 percent on workloads brought under commitment, applied to the portion of spend that is stable and predictable. Quarterly review is required because workload profiles shift.

Phase 2. Governance and Tagging, Weeks 5 to 8

The goal of Phase 2 is to make Phase 1 last. Quick wins decay if the discipline that produced them is not encoded into governance. The savings will reappear as new spend within two quarters if tagging is not enforced and cost ownership is not assigned.

WEEK 5. TAGGING STANDARD AND ENFORCEMENT ARCHITECTURE

Objective. Design a tagging standard that supports cost attribution at the team and workload level, and an enforcement architecture that makes the standard durable.

Activities.

- Define the required tag schema. Minimum recommended tags: Owner, CostCentre, Environment, Project, and DataClassification. Stop there until adoption is solid. Adding more before enforcement is in place produces noise, not insight.
- Choose the enforcement layer. AWS Organisations Service Control Policies. Azure Policy. GCP Organisation Policy Service. These enforce at the API layer.
- Choose the audit layer. AWS Tag Editor and Resource Groups. Azure Resource Graph. GCP Asset Inventory.
- Document the tag schema and enforcement architecture as a single artifact the engineering team will reference.

Deliverables. A tagging standard document. An enforcement architecture diagram. A signed-off tag schema from the platform engineering and finance teams.

Expected Savings This Week. Zero direct savings. This week's work is what makes the next twelve months of savings defensible.

WEEK 6. TAGGING ROLLOUT

Objective. Apply the tagging standard across the in-scope cloud estate.

Activities.

- Run the existing inventory through the new tag schema. Identify resources that fail compliance.
- Auto-tag where possible (account-level defaults, naming-convention inference). Open tickets for the residual non-compliant resources, owned by the named team owner from Week 1.
- Enforce the tag schema for new resources through policy-as-code. Terraform modules, Crossplane compositions, or hyperscaler-native policy as appropriate.
- Set the compliance target: 90 percent of in-scope resources tagged correctly by end of Week 8.

Deliverables. A tagging compliance dashboard. A list of resources with named tickets for tagging remediation. Enforcement policies merged into the IaC repository.

Expected Savings. Indirect. Tagged spend is attributable spend. Attributable spend is accountable spend.

WEEK 7. COST ATTRIBUTION AND SHOWBACK

Objective. Turn tagged spend into team-level visibility.

Activities.

- Build a showback dashboard at the team and workload level using the new tags. Tableau, Power BI, Grafana, or hyperscaler-native dashboards all work.
- Walk through the dashboard with each named team owner. The first walkthrough is always uncomfortable, and that is the point.
- Publish the showback report on a monthly cadence to engineering leadership and finance. The first published month becomes the baseline for the operating cadence in Phase 3.

Deliverables. A showback dashboard, published. A monthly publication cadence agreed with engineering leadership.

Expected Savings. Indirect, but consistent. Teams whose spend is visible to leadership tend to reduce their spend by 5 to 15 percent in the first three months of showback, without any direct intervention.

WEEK 8. COMMITMENT EXECUTION

Objective. Execute the commitment plan designed in Week 4, now that the rightsized baseline has held for four weeks.

Activities.

- Purchase Reserved Instances, Savings Plans, and CUDs according to the plan signed off in Week 4.
- Confirm utilisation tracking is in place against the new commitments.
- Schedule the first quarterly commitment review. The right window is Week 22 of the broader engagement. Before that, commitments are immature.
- Document the buy decisions, including the workloads they cover and the assumption they are based on.

Deliverables. Executed commitment purchases. A commitment tracking dashboard with utilisation alerts. A scheduled quarterly review.

Expected Savings. The full commitment savings (30 to 60 percent on covered workloads) begin to land. The first month of effect is partial. The full effect lands by Week 12.

Phase 3. Operating Model Rollout, Weeks 9 to 13

The goal of Phase 3 is to hand the program over to a sustainable operating cadence. Phase 1 saved money. Phase 2 made the savings attributable. Phase 3 makes the discipline durable.

WEEK 9. OPERATING RHYTHM DESIGN

Objective. Define the recurring cadence that will own FinOps after the build engagement ends.

Activities.

- Define the weekly cadence. A 30-minute FinOps and Engineering review. Standing agenda: anomaly review, optimisation backlog, commitment utilisation, upcoming changes.

- Define the monthly cadence. A finance and engineering joint forecast review against actuals. Variance over a defined threshold (10 percent is a defensible starting point) triggers a deeper review.
- Define the quarterly cadence. Commitment review (RIs, SPs, CUDs). Tag schema review. Operating model retrospective.
- Name the participants. Specifically, the named human who will run each cadence after the build engagement leaves.

Deliverables. An operating cadence document with named owners. Calendar invites issued for the first three cycles.

Expected Savings. Indirect, but compounding. The operating cadence is what holds the savings against drift.

WEEK 10. ANOMALY DETECTION AND ALERTING

Objective. Catch a spend anomaly before it shows up on the monthly bill.

Activities.

- Configure native anomaly detection: AWS Cost Anomaly Detection, Azure Cost Management anomaly alerts, GCP Cost Anomalies in BigQuery billing exports.
- Set anomaly thresholds at the service and team level. The right threshold catches the unusual without producing alert fatigue.
- Route alerts to the named owner from Week 9, with escalation to the FinOps program lead if not acknowledged within 24 hours.
- Document the anomaly response runbook: triage, classification (legitimate change, misconfiguration, attack), and remediation.

Deliverables. Anomaly detection live. Alert routing tested. Anomaly response runbook published.

Expected Savings. Indirect. The savings appear when the first real anomaly is caught and remediated in days, rather than discovered on the next month's bill.

WEEK 11. ENGINEERING ENABLEMENT

Objective. Move the optimisation discipline into the engineering team's normal workflow.

Activities.

- Run engineering enablement sessions for cloud and platform teams. Topics: tagging at deploy time, rightsizing recommendations, idle resource hygiene, commitment-aware provisioning.
- Add cost gates into the deployment pipeline. Terraform plan output annotated with cost delta. Infracost or hyperscaler-native equivalents.
- Add cost dashboards to the engineering team's existing tooling. The team should see their team-level cost without leaving their workflow.

Deliverables. Engineering enablement complete. Cost gates merged into the deployment pipeline. Cost dashboards embedded in the engineering workflow.

Expected Savings. 5 to 10 percent additional reduction over the following two quarters as engineers internalise cost as a non-functional requirement alongside latency and reliability.

WEEK 12. AI WORKLOAD HYGIENE

Objective. Address the highest-growth waste category in 2026.

Activities.

- Audit GPU and inference workload utilisation. Industry research suggests 30 to 50 percent of AI compute is over-provisioned. The figure tracks our engagement experience.
- Right-size GPU instance types where utilisation supports it. Move batch inference to spot or preemptible instances where the workload tolerates it. Move dev and experimentation workloads to scheduled environments.
- Audit vector store sizing, embedding pipeline frequency, and model-hosting infrastructure for the same waste patterns that show up on traditional compute.

Deliverables. An AI workload cost report with named recommendations. Executed rightsizing on AI compute. A documented review cadence for AI

workload cost. We recommend monthly, not quarterly, given the velocity of change in this area.

Expected Savings. 10 to 25 percent on the AI workload slice of in-scope spend, depending on how much GPU is in the estate.

WEEK 13. HANDOVER AND RETROSPECTIVE

Objective. Hand the operating model over to the client team and lock in the discipline.

Activities.

- Hold a handover session with the named owners from Week 9. Walk through the dashboards, the runbooks, the cadence, and the open items.
- Run a retrospective. What worked, what did not, and what should change in the operating model going forward.
- Produce a final savings report against the SOW target. Realised savings, in-flight savings, and savings deferred to the next quarter.
- Confirm the engagement transition. Either the build partner exits cleanly with a documented support window, or transitions into a managed FinOps retainer on agreed terms.

Deliverables. A final 90-day savings report. A signed-off handover document. The first month of operating cadence run by the client team under observation, not by the build partner.

Expected Savings Against SOW. The realised number lands here. Industry research and our engagement experience converge on 20 to 35 percent first-quarter realised savings against in-scope spend.

What Counts as Success at Day 90

Three measurable outcomes.

- Realised savings against the SOW target. Banked, not modelled.
- Tagging compliance above 90 percent of in-scope resources. Tagging is the foundation of the next 90 days of optimisation.

- An operating cadence run by the client team, with a named owner, that survives the build partner's exit.

If two of three are green, the engagement has succeeded. If only one is green, the engagement has produced point-in-time savings but not durable discipline. That is the most common failure mode of a FinOps program, and it is why Phase 2 and Phase 3 are part of this runbook rather than treated as optional follow-ons.

What Comes After Day 90

The operating cadence runs. Quarterly commitment reviews run. AI workload hygiene runs monthly. Engineering enablement is ongoing. The discipline compounds.

A FinOps program that has reached the end of Phase 3 in good condition typically continues to deliver 5 to 10 percent additional savings per quarter for the next twelve months, as the operating cadence catches new waste before it becomes structural. The discipline, not the tool, is what produces the compounding effect.

How We Use This Runbook

We run this exact runbook on FinOps engagements at ViitorCloud Enterprise, sized to the client's estate. The version we use internally includes the engagement SOW template, the savings target calculator, the tagging schema template, and the operating cadence document template. We will send the working set on request as part of a 30-minute FinOps scoping call.

Book a 30-minute FinOps scoping call.

enterprise@viitorcloud.com · (+91) 84889 64723

References and Tools Cited

- AWS Compute Optimizer, AWS Cost Explorer and Cost and Usage Reports, AWS Cost Anomaly Detection, AWS Organisations Service Control Policies, AWS Tag Editor and Resource Groups.
- Azure Advisor, Azure Cost Management, Azure Policy, Azure Resource Graph.
- GCP Recommender, GCP Billing Reports, GCP Cost Anomalies, GCP Organisation Policy Service, GCP Asset Inventory.
- Kubecost for cross-cloud Kubernetes cost optimisation. Infracost for pre-deploy cost gates.
- Terraform and Crossplane for policy-as-code enforcement.
- FinOps Foundation working groups (rightsizing, FOCUS specification, commitment management) for the underlying discipline framework.
- 2026 industry benchmarks on cloud waste, rightsizing returns, and commitment coverage drawn from FinOps practitioner publications and our own engagement data.

ViitorCloud Enterprise

Deep expertise for enterprise programs that have to hold up in production.

An enterprise practice of ViitorCloud Technologies. ISO 27001 certified, since 2009.

enterprise.viitorcloud.com · enterprise@viitorcloud.com · (+91) 84889 64723