System Administration Runbook

Nimbus Cloud Platform

Effective: 2026-07-15 Owner: Marcus Chen, Lead SRE CONFIDENTIAL

Internal Use Only — Do Not Distribute

1 Document Control

Version	Date	Author	Approver	Change Summary
1.0	2026-07-15	M. Chen	S. Okonkwo	Initial release. Covers NCP v4.2 production environment.
0.3	2026-06-28	M. Chen	S. Okonkwo	Added TLS certificate renewal procedure and escalation matrix.
0.2	2026-06-10	M. Chen	J. Kowalski	Incorporated incident response checklist and SEV classification.
0.1	2026-05-22	M. Chen	—	Draft: service overview and restart procedures.

2 System Architecture Overview

The Nimbus Cloud Platform (NCP) is a multi-tenant cloud infrastructure platform serving enterprise customers across three geographic regions: us-east, eu-west, and ap-southeast. The platform processes approximately 2.8 million API requests per minute at peak load and maintains a 99.95% uptime SLA for tier-1 customers.

INFORMATION

NCP runs on a containerized microservices architecture orchestrated by the NCP Orchestrator. All production services are deployed across a minimum of three availability zones per region. Database services use active-passive replication with automated failover triggered at 30 seconds of primary unresponsiveness.

Core Production Services

Service	Port	Description
ncp-api-gateway	8443	Primary ingress controller; terminates TLS, enforces rate limits, routes requests to backend services.
ncp-auth-service	9090	Identity and access management; OAuth 2.0 / OIDC provider with hardware MFA enforcement.
ncp-database-primary	5432	Managed PostgreSQL 15 cluster; primary write node for all transactional and session data.
ncp-message-queue	5672	Asynchronous message broker; handles event streaming, job dispatch, and dead-letter queues.
ncp-object-store	9000	S3-compatible blob storage for customer assets, database backups, and static content delivery.
ncp-orchestrator	6443	Container orchestration control plane; manages service deployment, autoscaling, and health probes.

3 Alert Severity Classification

Every incident must be assigned a severity level at the moment of declaration. The severity determines response time commitments, escalation paths, and post-incident review requirements.

Level	Label	Response	Definition & Examples
SEV-1	Critical	$<$ 5 minutes	Complete service outage affecting multiple regions or all customers. Data loss or corruption in progress. Security breach with active exploitation. **Example:** ncp-api-gateway returning 503 across all zones.
SEV-2	Major	$<$ 15 minutes	Significant service degradation. Partial outage affecting a single region or a subset of customers. **Example:** ncp-auth-service latency above 5 seconds in eu-west.
SEV-3	Minor	$<$ 1 hour	Isolated issue affecting a single customer or non-critical component. No data loss risk. **Example:** ncp-object-store slow listing for one tenant bucket.
SEV-4	Informational	$<$ 24 hours	Cosmetic issue, planned maintenance notification, or documentation gap. No customer impact. **Example:** Monitoring dashboard showing stale CPU metrics for ncp-message-queue.

CRITICAL

SEV-1 Declaration Authority: Only the on-call Lead SRE or Infrastructure Manager may declare or downgrade a SEV-1 incident. Any engineer may declare SEV-2 through SEV-4. All severity changes must be logged in the incident channel with justification.

4 Common Operational Procedures

4.1 Procedure 4.1: Service Restart

Restarting a production service requires connection draining and health verification. Never restart more than one service simultaneously without explicit approval from the Infrastructure Manager.

Step 1 — Assess Current State

Query the service status and note any active alerts or anomalies.

TERMINAL

```
$ ncp-cli service status ncp-api-gateway
SERVICE             STATUS    UPTIME     CPU%   MEM%    CONNS
ncp-api-gateway     RUNNING   14d 3h      12%   2.1G     1,847
```

Step 2 — Notify Affected Teams

Post a heads-up in `#ncp-incidents` with the service name, planned restart window, and expected impact duration. Wait for acknowledgement from at least one team lead before proceeding.

Step 3 — Drain Connections

Enable drain mode to stop new connections from reaching the instance while allowing existing requests to complete.

TERMINAL

```
$ ncp-cli drain enable ncp-api-gateway --timeout 90
Drain mode enabled. 1,847 active connections. Waiting...
All connections drained (87s). Safe to proceed.
```

Step 4 — Execute Restart

Perform a graceful restart and monitor the service log for errors.

TERMINAL

```
$ ncp-cli restart ncp-api-gateway --graceful
Stopping ncp-api-gateway... [OK]
Starting ncp-api-gateway... [OK]
$ ncp-cli logs ncp-api-gateway --since 2m | grep -i error
(no output -- clean restart)
```

Step 5 — Verify Health

Confirm the service is accepting traffic and responding within SLA thresholds.

TERMINAL

```
$ ncp-cli health-check ncp-api-gateway --sla
HEALTH CHECK: ncp-api-gateway
Status:       HEALTHY
Latency p99:  42ms (SLA: < 100ms)
Error rate:   0.00% (SLA: < 0.01%)
All checks passed.
```

VERIFICATION

Service restart completed successfully. Update the incident ticket if one was open and post a confirmation in `#ncp-incidents`. Disable drain mode if it was not automatically cleared: `$ ncp-cli drain disable ncp-api-gateway`.

4.2 Procedure 4.2: Disk Space Emergency Cleanup

Low disk space on production nodes can cause service failures, database corruption, and log loss. Act immediately when any filesystem exceeds 85% utilization.

WARNING

Do not delete files from `/data/ncp/database/` or `/data/ncp/object-store/` manually. These paths are managed by their respective services and improper cleanup can cause irrecoverable data loss.

Step 1 — Identify Consumption

Locate the largest directories and file types consuming space.

TERMINAL

```
$ df -h /data/ncp
Filesystem      Size   Used   Avail   Use%   Mounted on
/dev/sdb1       500G   462G    14G    97%    /data/ncp

$ du -sh /data/ncp/* | sort -rh | head -5
187G    /data/ncp/logs
142G    /data/ncp/object-store
 89G    /data/ncp/database
 31G    /data/ncp/temp
 13G    /data/ncp/backups
```

Step 2 — Rotate and Compress Logs

Archive logs older than 7 days and remove raw logs older than 30 days.

TERMINAL

```
$ ncp-logrotate --service all --older-than 7d --compress
Rotating logs older than 7 days...
  ncp-api-gateway:      compressed 2.3G -> 0.4G
  ncp-auth-service:     compressed 1.1G -> 0.2G
  ncp-database-primary: compressed 4.7G -> 0.9G
Done. Freed approximately 6.6G.

$ find /data/ncp/logs -name "*.log" -mtime +30 -delete
$ echo $?
0
```

Step 3 — Purge Temporary Files

Clear the temp directory. Confirm no active processes are writing to files being removed.

TERMINAL

```
$ lsof +D /data/ncp/temp 2>/dev/null | wc -l
0
$ rm -rf /data/ncp/temp/*
$ df -h /data/ncp | grep /data/ncp
/dev/sdb1       500G   412G    64G    87%    /data/ncp
```

VERIFICATION

Disk usage reduced from 97% to 87%. Monitor for 24 hours. If usage trends above 90% within that window, escalate to the Infrastructure Manager for capacity planning.

4.3 Procedure 4.3: TLS Certificate Renewal

Production TLS certificates are managed by ncp-certbot with automated renewal. Manual intervention is required only when automated renewal fails or when certificates are within 72 hours of expiry.

WARNING

A certificate expiry on ncp-api-gateway will cause a complete platform outage (SEV-1). Certificate renewal must never be deferred once expiry is within the 7-day warning window.

Step 1 — Check Certificate Status

List all managed certificates and their remaining validity periods.

TERMINAL

```
$ ncp-certbot list
CERTIFICATE                 EXPIRY       REMAINING   STATUS
api.nimbuscloud.io          2026-08-12   28 days     VALID
auth.nimbuscloud.io         2026-07-18    3 days     RENEWAL_DUE
console.nimbuscloud.io      2026-09-01   48 days     VALID
```

Step 2 — Execute Renewal

Renew certificates flagged as `RENEWAL_DUE`. The `--dry-run` flag validates without committing changes.

TERMINAL

```
$ ncp-certbot renew auth.nimbuscloud.io --dry-run
Dry run: auth.nimbuscloud.io
Challenge type:   DNS-01
DNS propagation:  OK (4.3s)
Certificate issued by: Let's Encrypt R11
Dry run successful. Ready to renew.

$ ncp-certbot renew auth.nimbuscloud.io
Renewing auth.nimbuscloud.io...
New certificate fingerprint: A7:F3:...:9C
Valid until: 2026-10-16
```

Step 3 — Reload Affected Services

Services that terminated TLS using the old certificate must reload their configuration.

TERMINAL

```
$ ncp-cli reload-certs ncp-auth-service
Reloading TLS certificates for ncp-auth-service... [OK]
New certificate active. Expires: 2026-10-16.
```

VERIFICATION

Post-Renewal Verification

`ncp-certbot list`	Confirm all certificates show VALID status
`curl -svI ...`	Verify the new expiry date appears in the TLS handshake
`ncp-cli health-check ncp-auth-service`	Confirm service health after certificate reload

5 Incident Response Workflow

When an incident is detected (alert, customer report, or engineer observation), follow this workflow without deviation.

INFORMATION

Incident Channel: All incident communication happens in `#ncp-incidents`. Do not use direct messages or side channels for incident coordination. The incident channel is the single source of truth for post-incident review.

Incident Response Checklist

1. **Declare:** Acknowledge the alert in `#ncp-incidents`. Assign a severity level per Section 3. Start a timer.
2. **Triage:** Identify the affected service, region, and customer scope. Post findings to the channel.
3. **Mitigate:** Apply the relevant procedure from Section 4. If no procedure matches, escalate to the Lead SRE immediately.
4. **Communicate:** Post a status update every 15 minutes for SEV-1, every 30 minutes for SEV-2. Include: current state, action in progress, ETA to resolution.
5. **Resolve:** Confirm that all health checks pass and affected customers are restored. Post the resolution time and root cause summary.
6. **Review:** Within 48 hours, the incident owner must draft a post-incident review document covering timeline, root cause, impact, and preventive actions.

CRITICAL

SEV-1 Communication Template

Copy and paste into `#ncp-incidents` at declaration and every 15-minute update:

```
[SEV-1] {service-name} -- {one-line summary}
Declared: {timestamp UTC}
Incident Lead: {name}
Affected: {regions / customer count / feature}
Current Action: {what is being done right now}
Next Update: {timestamp + 15 min}
```

6 Escalation Matrix

Escalate when a severity-appropriate resolution window is exceeded or when the incident owner determines they lack the necessary access or expertise to resolve the issue.

Role	Name	Contact	Escalation Trigger
Lead SRE	Marcus Chen	+1-555-0147 marcus.chen@nimbus.internal	SEV-1 declaration; any incident unresolved for 15 minutes.
Infra. Manager	Sarah Okonkwo	+1-555-0158 sarah.okonkwo@nimbus.internal	SEV-1 unresolved for 30 minutes; multi-service cascading failure.
Security Lead	Elena Rossi	+1-555-0162 elena.rossi@nimbus.internal	Any suspected security incident; unauthorized access; data exfiltration alert.
CTO	David Park	+1-555-0171 david.park@nimbus.internal	SEV-1 unresolved for 2 hours; SEV-2 unresolved for 4 hours; customer-facing data loss.
VP Engineering	Amara Okafor	+1-555-0184 amara.okafor@nimbus.internal	Escalation required beyond CTO; media or regulatory inquiry about an incident.

WARNING

Escalation Protocol: When escalating, always include the incident channel link, current severity, elapsed time, and a one-paragraph summary of actions taken so far. Call the phone number first; follow up with the email if there is no response within 5 minutes. Never hesitate to escalate — over-escalation is a process concern for the post-incident review, not a violation during an active incident.