Service Health Check Runbook

A reusable operational path for validating web, DNS, Cloudflare, tunnel, origin service, and server-host availability.

VALIDATE · ISOLATE · RECOVER

DOCUMENT CODE OPS-002

6	11	P1-P3	2026
VALIDATION LAYERS	OPERATIONAL CHECKS	SEVERITY MODEL	DOCUMENTED

DOCUMENT CONTROL

OPS-002 is a reusable operational runbook with a defined scope.

OPS-002

SERVICE HEALTH CHECK RUNBOOK

Author: Jean Claude Munyakazi
Series: IT Operations Notes
Status: Portfolio operational document
Review cycle: After material infrastructure changes

Purpose

Validate service availability and isolate the failed dependency before recovery.

Target

Public HTTPS service delivered through Cloudflare Tunnel to an Apache origin server.

Evidence Standard

Record commands, observed output, timestamps, corrective action, and final validation.

OPERATIONAL INTENT

The runbook separates symptoms from the failed service layer.

Availability

Can an external client resolve the name, connect, and receive the intended HTTP response?

Dependency

Which layer fails first: DNS, edge, tunnel, Apache, or the Linux server host?

Recovery

Apply only the action appropriate to the failed layer, then validate internally and externally.

CONTROL PRINCIPLE · Do not restart every component. Preserve evidence and recover the failed dependency.

VALIDATION
ARCHITECTURE

Six layers provide a fast fault-isolation sequence.

01	PUBLIC URL	HTTP status + timing
02	DNS	A / AAAA resolution
03	CLOUDFLARE EDGE	Headers + edge response
04	TUNNEL	cloudflared service + logs
05	APACHE	Service + config + local origin
06	LINUX HOST	Load + memory + disk + units

CHECK 01 · PUBLIC URL

The endpoint returns HTTP 200 within the defined timeout.

COMMAND

```
curl -sS -o /dev/null --max-time 15 \
  -w 'HTTP=%{http_code} DNS=%{time_namelookup}s
CONNECT=%{time_connect}s TOTAL=%{time_total}s\n' \
  https://munyakazi.org
```

EXPECTED · HTTP 200 or intended redirect, completed within the timeout

OBSERVED · HTTP=200 · DNS=0.003521s · CONNECT=0.050197s · TOTAL=0.546903s

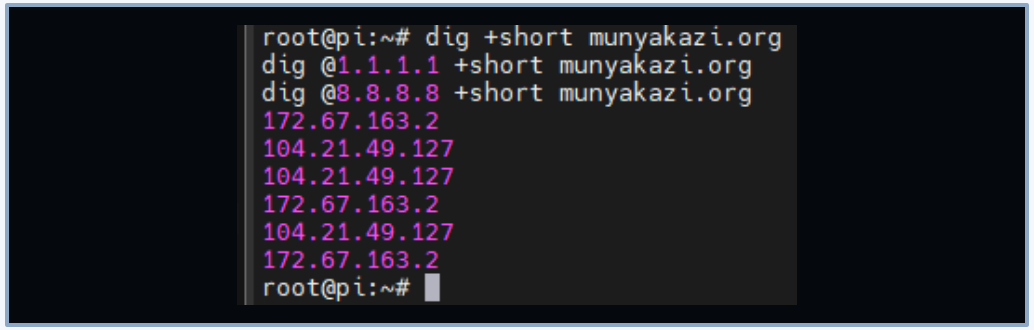
CHECK 02 · DNS

Name resolution must succeed before edge or origin troubleshooting.

COMMAND

```
dig +short munyakazi.org
dig @1.1.1.1 +short munyakazi.org
dig @8.8.8.8 +short munyakazi.org
```

EXPECTED · Public DNS resolvers return consistent A records for the Cloudflare-managed endpoint



OBSERVED · Public resolvers 1.1.1.1 and 8.8.8.8 consistently returned Cloudflare addresses 104.21.49.127 and 172.67.163.2.

CHECK 03 · CLOUDFLARE EDGE

The response code classifies where the investigation should move next.

HEADER CHECK

```
curl -sSI https://munyakazi.org
curl -sS -D - -o /dev/null https://munyakazi.org
```

2xx / 3xx	Public path is available
403 / 429	Edge policy, WAF, access, or rate control
502 / 504	Tunnel or origin cannot complete the request
522 / 523	Origin reachability or network path problem
1033	Cloudflare Tunnel cannot find a healthy connector

CHECK 04 · TUNNEL

cloudflared must be active, connected, and free of repeated origin errors.

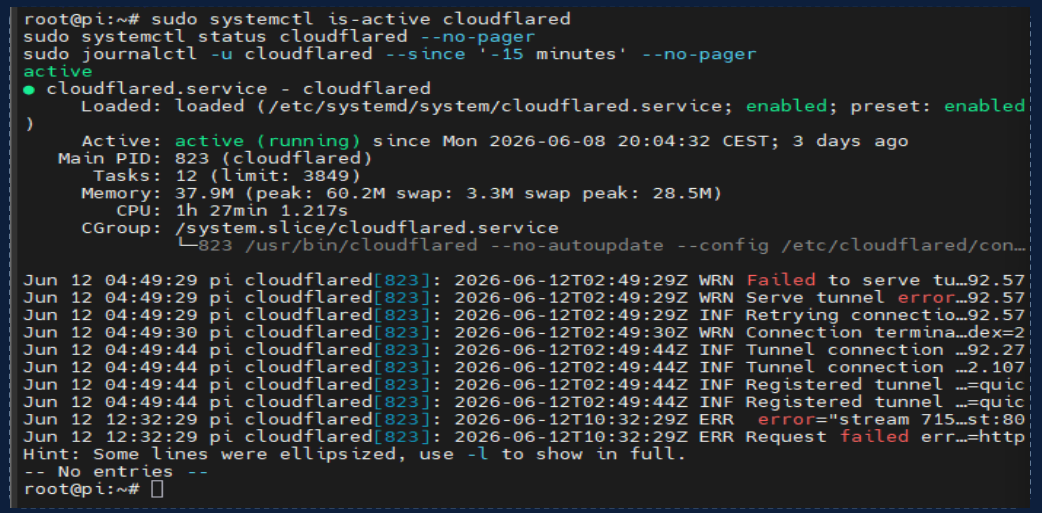
COMMAND

```
systemctl is-active cloudflared
systemctl status cloudflared --no-pager
journalctl -u cloudflared --since '-15 minutes' --no-pager
```

FOCUSED LOG FILTER

```
journalctl -u cloudflared --since '-15 min' \
  | grep -Ei 'error|failed|timeout|origin'
```

EXPECTED · Active service with healthy connector sessions and no recurring origin failure



OBSERVED · was active and running. Temporary tunnel errors appeared in the logs, followed by successful reconnection and QUIC tunnel registration.

CHECK 05 · APACHE

Service state and configuration syntax identify an origin-service failure.

COMMAND

```
systemctl status apache2 --no-pager
apache2ctl configtest
apache2ctl -S
```

EXPECTED · Apache active and enabled; configuration test returns Syntax OK.

```
root@pi:~# sudo systemctl is-active apache2
sudo systemctl status apache2 --no-pager
sudo apachectl configtest
active
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: enabled
)
    Drop-In: /etc/systemd/system/apache2.service.d
             └─override.conf
     Active: active (running) since Wed 2026-06-10 06:53:40 CEST; 2 days ago
       Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 29531 (apache2)
      Tasks: 11 (limit: 3849)
     Memory: 778.7M (peak: 3.2G swap: 1.6M swap peak: 533.2M)
        CPU: 2h 42min 44.131s
     CGroup: /system.slice/apache2.service
             ├─ 29531 /usr/sbin/apache2 -k start
             ├─ 84605 /usr/sbin/apache2 -k start
             ├─ 88360 /usr/sbin/apache2 -k start
             ├─ 88365 /usr/sbin/apache2 -k start
             ├─ 88370 /usr/sbin/apache2 -k start
             ├─140089 /usr/sbin/apache2 -k start
             ├─140090 /usr/sbin/apache2 -k start
             ├─148338 /usr/sbin/apache2 -k start
             ├─148345 /usr/sbin/apache2 -k start
             ├─148346 /usr/sbin/apache2 -k start
             └─148353 /usr/sbin/apache2 -k start

Jun 10 06:53:39 pi systemd[1]: Starting apache2.service - The Apache HTTP Server...
Jun 10 06:53:40 pi systemd[1]: Started apache2.service - The Apache HTTP Server.
Syntax OK
root@pi:~#
```

OBSERVED · Apache was active and running. The service was enabled, all worker processes were operational, and apache2ctl configtest returned Syntax OK.

A local request separates Apache health from tunnel and edge routing.

COMMAND

```
curl -I --max-time 10 http://127.0.0.1
sudo ss -ltnp | grep -E ':(80|443)\b'
```

EXPECTED · Local origin returns the intended HTTP response or redirect, and Apache listens on ports 80 and 443.

```
root@pi:~# curl -I --max-time 10 http://127.0.0.1
sudo ss -ltnp | grep -E ':(80|443)\b'
HTTP/1.1 301 Moved Permanently
Date: Fri, 12 Jun 2026 15:58:42 GMT
Server: Apache/2.4.58 (Ubuntu)
Vary: Accept-Encoding,Cookie
Permissions-Policy: private-state-token-redemption=(self "https://www.google.com" "https://www.gstatic.com" "https://recaptcha.net" "https://challenges.cloudflare.com" "https://hcaptcha.com"), private-state-token-issuance=(self "https://www.google.com" "https://www.gstatic.com" "https://recaptcha.net" "https://challenges.cloudflare.com" "https://hcaptcha.com")
X-Redirect-By: WordPress
Location: https://munyakazi.org/
Content-Type: text/html; charset=UTF-8

LISTEN 0      511                *:443              *:*    users:(("apache2",pid=148353,fd=6),("apache2",pid=148346,fd=6),("apache2",pid=148345,fd=6),("apache2",pid=148338,fd=6),("apache2",pid=140090,fd=6),("apache2",pid=140089,fd=6),("apache2",pid=88370,fd=6),("apache2",pid=88365,fd=6),("apache2",pid=88360,fd=6),("apache2",pid=84605,fd=6),("apache2",pid=29531,fd=6))
LISTEN 0      511                *:80               *:*    users:(("apache2",pid=148353,fd=4),("apache2",pid=148346,fd=4),("apache2",pid=148345,fd=4),("apache2",pid=148338,fd=4),("apache2",pid=140090,fd=4),("apache2",pid=140089,fd=4),("apache2",pid=88370,fd=4),("apache2",pid=88365,fd=4),("apache2",pid=88360,fd=4),("apache2",pid=84605,fd=4),("apache2",pid=29531,fd=4))
root@pi:~#
```

OBSERVED · The local origin returned HTTP 301 Moved Permanently and redirected to https://munyakazi.org/. Apache was listening successfully on TCP ports 80 and 443

CHECKS 07-09 · SERVER HOST

Resource and operating-system evidence reveal capacity or host-level causes.

HOST HEALTH COMMAND SET

```
uptime
free -h
df -h
df -i
systemctl --failed
dmesg -T | tail -n 50
```

LOAD	`uptime / top`	No sustained saturation
MEMORY	`free -h`	Available memory and swap
DISK	`df -h`	Sufficient free capacity
INODES	`df -i`	No inode exhaustion
UNITS	`systemctl --failed`	No relevant failed service
KERNEL	`dmesg -T | tail`	No storage or OOM event

CHECK 10 · TARGETED RECOVERY

Correct only the failed layer and preserve the evidence trail.

DNS	Correct the record or zone configuration
EDGE	Review policy, WAF, access, or certificate state
TUNNEL	Restart cloudflared only after capturing logs
APACHE	Fix syntax or content, then graceful reload
HOST	Recover capacity, failed units, or operating-system resources

CHECK 11 · FINAL VALIDATION

Recovery is complete only when internal and external checks both pass.

COMMAND

```
curl -sSI http://127.0.0.1
curl -sSI https://munyakazi.org
systemctl is-active apache2 cloudflared
```

Internal Gate

The local origin returns the intended response without tunnel or edge dependency.

External Gate

Public DNS, Cloudflare edge, tunnel, and origin complete the request path.

Operational Gate

Services remain active, logs remain stable, and the incident record captures the result.

OPERATIONAL CONTROL

Severity and incident records turn a health check into a repeatable process.

P1 · Critical

Public service unavailable; broad impact; immediate response and communication.

P2 · Major

Degraded or intermittent service; material user impact; prioritized remediation.

P3 · Minor

Limited impact or preventive finding; schedule correction and monitor.

Minimum Incident Record

- Date and detection time
- User-visible symptom
- Failed validation layer
- Commands and observed output
- Corrective action
- Internal and external validation
- Closure time and follow-up

Evidence makes the result auditable and suitable for handover.

OPS-002 · CLOSING

From symptom to evidence, fault isolation, and controlled recovery.

01

VALIDATE

Check the request path in dependency order.

02

ISOLATE

Stop at the first failed layer and capture evidence.

03

RECOVER

Apply the targeted corrective action.

04

PROVE

Revalidate internally and externally, then document.

MAGISTER OPERIS SYSTEMS · PROFESSIONAL BLOG · IT OPERATIONS NOTES