GEARFLOW LAB PRODUCTION SRE RUNBOOK 08 | NGINX 1.27 + CADDY 2.8 + TRAEFIK V3 + CLOUDFLARED

Reverse-Proxy & Zero-Trust Tunnel Directive Matrix (SSE, WSS & UFW)

Open Matrix Generator ->

Incident Context: Exposing `n8n` MCP SSE, `Ollama` token streams, `FastAPI` WebSockets, or `PgBouncer` behind default proxy configs triggers 60-second read drops (`504`), Cloudflare 100-second silent timeouts (`524`), buffered token stalls (`408.7 ms` vs `24.8 ms` TTFT), and `0.0.0.0` Docker UFW bypasses.

1. Side-by-Side 4-Engine Directive Translation & Empirical ARM64 Telemetry

Capability	Nginx 1.27	Caddy 2.8	Traefik v3.2	Cloudflared Tunnel
Unbuffered SSE	`proxy_buffering off; X-Accel-Buffering: no`	`reverse_proxy { flush_interval -1 }`	Streams natively; add `X-Accel-Buffering=no`	`disableChunkedEncoding: false` + **`15s : ping`**
Read Timeout	`proxy_read_timeout 3600s; proxy_send_timeout 3600s;`	`transport http { read_timeout 3600s }`	`respondingTimeouts. readTimeout=3600s`	`keepAliveTimeout: 3600s` (Edge `524` at `100.0s`!)
WebSocket 101	`proxy_http_version 1.1; Upgrade + $connection_upgrade`	Automatic RFC 6455 `101` upgrade built-in	Automatic RFC 6455 `101` upgrade built-in	Supported natively over QUIC / HTTP2 tunnel
Keepalive Pool	`keepalive 64; + Connection ""; (10k reqs)`	`keepalive 65s keepalive_idle_conns 64`	Managed Go `Transport` idle pool per service	`keepAliveConnections: 100` (`tcpKeepAlive: 30s`)
Rate Limiting	`limit_req_zone 60r/s + burst=120 nodelay`	`caddy-ratelimit` plugin (xcaddy build)	`middlewares.<n>. ratelimit.average=60`	Cloudflare WAF / Zero Trust Access Policy
Real Visitor IP	`set_real_ip_from` CF CIDRs + `CF-Connecting-IP`	`trusted_proxies static + client_ip_headers`	`forwardedHeaders. trustedIPs=<CF_CIDRs>`	Injects `CF-Connecting-IP` & `X-Forwarded-Proto`
Lab RSS / RPS	`18-36 MiB` | `24.8k+` H2 RPS (`1.85ms` p50)	`14-34 MiB` | `10.0k` H2 RPS (`2.21ms` p50)	`23-67 MiB` | `20.1k` H2 RPS (`3.24ms` p50)	`54-66 MiB` | Outbound QUIC (0 WAN ports open)

2. Mandatory `127.0.0.1` UFW Loopback & `n8n` Proxy Env

```yaml
# NEVER publish "5678:5678" (0.0.0.0 bypasses Ubuntu UFW via PREROUTING!)
services:
  n8n:
    image: n8nio/n8n:latest
    restart: unless-stopped
    ports:
      - "127.0.0.1:5678:5678"
    environment:
      - N8N_HOST=n8n.example.com
      - N8N_PROTOCOL=https
      - N8N_PROXY_HOPS=1
      - N8N_SECURE_COOKIE=true
      - WEBHOOK_URL=https://n8n.example.com/
```

Keepalive Race Rule: Set Uvicorn `--timeout-keep-alive 65` (> `60s` Nginx `keepalive_timeout`) so backend workers never send `TCP FIN` while Nginx reuses pooled sockets (`HTTP 502`).

3. Pre-Cutover Syntax, Socket & `curl -i -N` Checklist

```bash
# 1. Validate config offline before reloading workers
sudo nginx -t && sudo systemctl reload nginx
caddy validate --config /etc/caddy/Caddyfile
cloudflared tunnel --config /etc/cloudflared/config.yml ingress validate

# 2. Confirm zero 0.0.0.0 wildcard upstream leaks (must show 127.0.0.1)
sudo ss -tulpn | grep -E ':(5678|11434|5080|8000|6432)\b'

# 3. Verify TLS 1.3 + ALPN h2 + unbuffered SSE chunk streaming
openssl s_client -connect n8n.example.com:443 -tls1_3 -alpn h2 </dev/null \
  2>/dev/null | grep -E "Protocol|ALPN"
curl -i -N --http2 -H "Accept: text/event-stream" \
  --max-time 15 https://n8n.example.com/mcp/sse
```

1-Click Prefill Matrix Configs: n8n SSE + Nginx | Ollama Bearer + Caddy 2 | FastAPI WSS + Traefik v3 | OpenObserve + Cloudflared | All Tools (/tools/)