

# CF IP Access Rules — ALAI LAN Bypass

# CF IP Access Rules — ALAI LAN Bypass

**Zone:** alai.no **Zone ID:** 3dc40d9c37fee79c4281f7e86870c0b5 **Account ID:** d0ac2afb6bb5b298723b85a114151a04 **Last updated:** 2026-04-28 **MC references:** [#9956](#) (WAF whitelist), [#9546](#) (CF Access bypass)

“ Two CF layers, two rule types — both require ALAI LAN egress ( 92.221.168.61 ) in their respective allowlists.

## Layer 1 — IP Access Rules (WAF / bot / rate-limit bypass)

API: /zones/{zone\_id}/firewall/access\_rules/rules

| Rule ID                          | IP               | Mode      | Created    | Notes                                                         |
|----------------------------------|------------------|-----------|------------|---------------------------------------------------------------|
| 94994e3badcd4349815190038940bf19 | 92.221.168.61/32 | whitelist | 2026-04-28 | ALAI LAN egress (Klofta) — Mac Studio/ANVIL + Mac Air + peers |

**Effect:** Suppresses CF WAF/bot detection (error 1010), rate-limit, security level checks. Auth gates (CF Access) still apply.

# Layer 2 — CF Access (Zero Trust auth bypass) policies

API: `/accounts/{account_id}/access/apps/{app_id}/policies/{policy_id}`

| App Name            | App ID                                            | Domain                            | Policy ID                                         | Decision | IPs in include                                                               |
|---------------------|---------------------------------------------------|-----------------------------------|---------------------------------------------------|----------|------------------------------------------------------------------------------|
| All ALAI Services   | <code>cd7cf0f0-ab37-4b06-8d51-9f042fd7a4f6</code> | <code>*.alai.no</code>            | <code>cecc0b27-192e-4d09-be80-27d792945a60</code> | bypass   | 46.46.253.33, 20.240.61.67, 46.46.251.40, 46.46.247.60, <b>92.221.168.61</b> |
| All Studio Services | <code>f4d85fab-1c4b-4a48-97a6-ea982e7444e2</code> | <code>*.basicconsulting.no</code> | <code>b72086cd-2995-403e-872f-b2c29d3aac39</code> | bypass   | same 5 IPs                                                                   |
| lightrag.alai.no    | <code>c62b46b1-43f4-4967-9b99-cabfefb6b99b</code> | lightrag.alai.no                  | <code>913922cd-c637-4999-b0e9-ef25f9e35fae</code> | bypass   | 46.46.253.33, 20.240.61.67, 46.46.251.40, 92.221.168.61                      |
| ollama.alai.no      | <code>bdc17e6a-94c2-42d9-b5ce-37c1c37ac016</code> | ollama.alai.no                    | <code>162b2533-bd29-497e-9ada-db8684da869d</code> | bypass   | 46.46.253.33, 20.240.61.67, 46.46.251.40, 92.221.168.61                      |

**Effect:** Skips Zero Trust auth (302 redirect to `cloudflareaccess.com`). Direct backend response.

**Auth:** Cloudflare Global API Key (`john@basicconsulting.no`) via Bitwarden — required for `/access/*` endpoints (regular `cf-api-token` insufficient scope).

## Why this exists

ALAI internal automation (Python klijenti, curl skripte, CI agenti) konektujući se na `*.alai.no` servise iz ALAI LAN egress IP-a hit-ovali su CF WAF/bot detection (error 1010), uzrokujući 46h LightRAG outage 2026-04-20 i konstantne automation failures. IP Access Rule sa mode=whitelist suprimira WAF/bot blocks za saobraćaj iz ovog IP-a.

## Kako radi (CF layer order)

1. Request stiže na CF edge
2. **IP Access Rules** se evaluiraju — ako IP match-uje whitelist, WAF/bot je bypassed

3. **CF Access (Zero Trust)** se evaluira — auth redirect i dalje važi bez obzira na IP whitelist
4. Origin reached

So: whitelist suprimira WAF/bot, ali NE preskače CF Access autentikaciju. Dva nezavisna sloja.

---

## Authoritative IP source — NE koristi `curl ifconfig.me` sam

Per [zakon-network-egress-verification.md](#):

- `curl ifconfig.me` vraća VPN exit ako je VPN klijent aktivan (više utun interfejsa)
- Za ISP egress, koristi `tailscale status` peer `direct PEER_IP:PORT` konekcije
- Ili `dig +short myip.opendns.com @resolver1.opendns.com` (DNS-based, često bypassuje VPN HTTP routing)

3-source verifikacija obavezna prije bilo kakvog whitelist task-a.

---

## Verifikacija

### Iz whitelistovanog IP-a

```
curl -sI https://lightrag.alai.no/  
# Expected: HTTP 200 (ili 302 redirect na CF Access – oboje OK, no 1010)
```

## Provjera da rule postoji u CF

```
TOKEN=$(bw get item "Cloudflare Global API Key" --session $(cat /tmp/bw-session))  
curl -s  
"https://api.cloudflare.com/client/v4/zones/3dc40d9c37fee79c4281f7e86870c0b5/firewall/access_r  
ules/rules?configuration.value=92.221.168.61" \  
-H "X-Auth-Email: ..." -H "X-Auth-Key: $TOKEN" | jq
```

## Lista svih IP Access Rules

```
curl -s
"https://api.cloudflare.com/client/v4/zones/3dc40d9c37fee79c4281f7e86870c0b5/firewall/access_rules/rules" \
-H "X-Auth-Email: ..." -H "X-Auth-Key: $TOKEN" | jq '.result[] | {id, mode, configuration, notes}'
```

# Dodavanje novog IP-a u whitelist

## 1. **3-source verifikacija** — Mehanik Phase N gate to enforces:

- VPN check: `ifconfig | grep -c "^utun"`
- Source 1: `curl -s https://api.ipify.org`
- Source 2: `dig +short myip.opendns.com @resolver1.opendns.com`
- Source 3: `tailscale status | grep "direct"`

## 2. **POST CF API:**

```
curl -X POST \

"https://api.cloudflare.com/client/v4/zones/3dc40d9c37fee79c4281f7e86870c0b5/firewall/access_rules/rules" \
-H "X-Auth-Email: ..." -H "X-Auth-Key: $TOKEN" \
-d '{"mode": "whitelist", "configuration": {"target": "ip", "value": "<NEW_IP>"}, "notes": "..."}'
```

## 3. **Validation:** curl iz whitelistovanog IP-a, expect 200

## 4. **Update ovog dokumenta** i [DEPLOY-MAP.md](#) sa novim Rule ID + IP

# Out of scope za whitelist

- **VPN exit IP** (npr. `46.46.247.96` Mullvad/sl.) — rotira, dijeli ga drugi korisnici, ne whitelistovati
- **Azure VM IP** (`20.240.61.67`) — separate firewall layer, ne CF IP whitelist (Azure NSG)

# Related

- [ZAKON NETWORK EGRESS](#) — 3-source verification protocol
- [CF Proxied API BIC Whitelist](#) — Configuration Rule pattern (related but different layer)

- [DEPLOY-MAP — System Infrastructure](#) — canonical map ALAI deploys
  - **Incident origin:** 2026-04-28 ANVIL whitelist task — 4 reverzalne IP claims (memory 46.46.247.60 stale, curl returned VPN exit 46.46.247.96) prije nego što je 92.221.168.61 confirmed kao stvarni LAN egress preko Tailscale peer connections + CEO confirmation. Lessons logged: feedback\_lateral\_thinking\_before\_incapability\_claim, feedback\_memory\_value\_decay\_verify, feedback\_clarify\_machine\_topology, feedback\_vpn\_exit\_vs\_isp\_egress.
- 

Revision #4

Created 2026-04-28 11:26:54 UTC by John

Updated 2026-07-05 20:05:33 UTC by John