DirectAdmin startips — supervisor / auto-restart de IPs virtuales

Fecha: 2026-04-30

- **Autor:** Abdelkarim Mateos
- **Estado:** plan de desarrollo, NO IMPLEMENTADO. Reservado para sesión futura.
- **Origen:** incidente Burcode 2026-04-30 (cloud500 IP virtual `87.98.230.68` caída → diagnóstico erróneo y reescritura de 178 A records → reversión).
- **Informe relacionado:** `informes/2026-04/2026-04-30-burcode-cloud500-fail500-dns-glue-muerto.md`
- **Memory:** `~/.claude/projects/-Users-abkrim-claude/memory/feedback-da-startips-no-protegido.md`

1. Problema

`startips.service` en DirectAdmin es `Type=oneshot`:

```
[Unit]
Description=Start the additional IPs
Wants=network-online.target
After=syslog.target network.target network-online.target
Requires=network.target

[Service]
Type=oneshot
ExecStart=/usr/local/directadmin/scripts/startips

[Install]
WantedBy=multi-user.target
```

El binario `/usr/local/directadmin/scripts/startips` recorre `/usr/local/directadmin/data/admin/ip.list` y aplica IPs adicionales con `ip addr add` sobre la interfaz primaria. Tras correr, el servicio queda `inactive (dead)` con `code=exited, status=0/SUCCESS`. **No hay supervisor.**

Si las IPs virtuales se caen tras boot por cualquier motivo:

- Reset/restart del subsistema de red (`systemctl restart networkd`, `netplan apply`, mantenimiento del proveedor).
- OOM killing de `networkd`.
- vRack/cloud-init re-aplicando configuración fresca.
- Cambio en `ens3` que limpia los aliases.

...las IPs no vuelven hasta el siguiente reboot completo o hasta que alguien lance `systemctl start startips.service` o `bash /usr/local/directadmin/scripts/startips` manualmente.

Impacto operativo:

- Dominios DA con virtual host ligado a IP virtual quedan inalcanzables.
- LE auto-renew falla para esos dominios (HTTP-01 challenge no responde).
- Diagnóstico desde fuera puede confundirse con "servidor antiguo desmantelado" (el operador o un agente terminan reescribiendo DNS por error — caso real Burcode 2026-04-30).

El operador reporta que es **problema recurrente** en su parque DA — segundo frente de trabajo abierto.

2. Diseño propuesto

2.1 Componentes

A) Health-check periódico de IPs virtuales

Script en `~/utilidades/directadmin/check-virtual-ips.sh` (versionado en monorepo `gitlab.castris.com/root/utilidades`):

- Lee `/usr/local/directadmin/data/admin/ip.list`.
- Para cada IP, comprueba si está activa en la interfaz primaria (`ip -4 addr show <iface> | grep <ip>`).
- Si UNA IP de la lista NO está en la interfaz, lanza:
  1. `systemctl start startips.service` (idempotente — el servicio ya existe).
  2. Verifica que tras el restart la IP aparece.

3. Si tras el restart sigue sin aparecer, alerta (mail/Telegram) y NO bucla.

B) Systemd timer para el health-check

En `/etc/systemd/system/startips-watchdog.{service,timer}`:

```
# startips-watchdog.service
[Unit]
Description=Watchdog for DirectAdmin virtual IPs
After=startips.service

[Service]
Type=oneshot
ExecStart=/usr/local/sbin/check-virtual-ips.sh
StandardOutput=journal
StandardError=journal
```

```
# startips-watchdog.timer
[Unit]
Description=Run virtual IP watchdog every 5 minutes

[Timer]
OnBootSec=2min
OnUnitActiveSec=5min
AccuracySec=30s

[Install]
WantedBy=timers.target
```

Frecuencia 5 min = compromiso entre detección rápida y carga mínima. Ajustable per-server.

C) Alertas

Para no llenar el inbox, política de "alerta solo si auto-fix falla":

- Si auto-restart resuelve → log en journald, sin alerta.
- Si auto-restart falla 2 ciclos consecutivos → mail al operador + Telegram (vía bot existente).
- Métrica opcional: contador en `/var/log/startips-watchdog.metrics` con histórico de detecciones.

D) (Opcional) Métricas a Zabbix

Si Castris tiene Zabbix, exponer:

- Item `startips.virtual_ips.expected` (entero, lectura de `ip.list`).
- Item `startips.virtual_ips.active` (entero, IPs activas en interfaz que están en `ip.list`).
- Trigger: si `expected != active` durante 2 ciclos → warning.

3. Plan de trabajo (otra sesión)

Fase 1 — POC manual (1-2 horas)

1. Escribir `check-virtual-ips.sh` con dry-run + apply.
2. Probar en cloud500 (caso conocido). Forzar caída con `ip addr del 87.98.230.68 dev ens3` y verificar detección + auto-fix.
3. Validar logs en journald.

Fase 2 — Empaquetado (1 hora)

1. Mover script a `~/utilidades/directadmin/`.
2. Crear systemd unit + timer.
3. Documentar en `~/utilidades/directadmin/README.md`.
4. Empaquetar en script de instalación que copie unit + binarios + habilite el timer.

Fase 3 — Despliegue flota DA Castris (30 min)

1. Auditar qué servidores DA tienen IPs adicionales:

```
for s in $(sshctx list --panel directadmin --format raw); do
  n=$(ssh $s 'wc -l < /usr/local/directadmin/data/admin/ip.list')
  echo "$s: $n IPs"
done
```

2. Desplegar el watchdog en los que tienen `>1 IP`.
3. Servidores conocidos con IPs adicionales: cloud500 (Burcode, 4 IPs). Resto pendiente auditoría.

Fase 4 — Despliegue clientes externos (variable)

Servidores semi-managed (Burcode, otros): pedir autorización antes de instalar.

4. Casos de uso

- **Burcode cloud500** (caso semilla del incidente).
- **Flota DA Castris** que tenga IPs virtuales: srv120, srv121, kvm456, dar, amazzal, titrit, bitatrader (auditar).
- **Cualquier DA futuro provisionado con IPs adicionales** (regla preventiva: instalar el watchdog en provisión, junto con CSF/DA).

5. Decisiones pendientes (para sesión de implementación)

5.1 Frecuencia del timer

Opciones:

- **5 min**: detección rápida, carga insignificante. Recomendado.
- **1 min**: detección casi inmediata pero ruido en journald.
- **10-15 min**: caída detectable pero ventana de fallo amplia.

5.2 Política de retry

Opciones:

- **Auto-restart 1 vez + alerta si falla**: simple, evita bucles.
- **Backoff exponencial**: 1 min → 5 min → 15 min → alerta.
- **Solo alerta, sin auto-restart**: dejar decisión al operador.

5.3 Alcance del watchdog

Opciones:

- **Solo IPs virtuales DA** (lo que propone este plan).
- **Extender a otros servicios cuya caída sea silenciosa** (ej. `dataskq` cron, `assp`, etc.). Probable scope-creep — empezar simple.

5.4 Forma de alerta

Opciones:

- **Mail al operador** (existing).
- **Telegram via bot existente** (más rápido).
- **Ambos** (recomendado para incidentes recurrentes).

6. Riesgos y mitigaciones

Riesgo	Mitigación
El watchdog se mete en bucle restartando startips si éste falla siempre	Implementar contador de retries, parar tras N fallos, alertar
Race condition con un cambio legítimo de IP del operador	El watchdog debe leer `ip.list` cada vez (no cachear); si se cambia `ip.list` está reflejado
El propio `startips.service` es buggy en DA y no funciona en algunos escenarios	Documentar exit code de startips; si falla repetidamente, alertar para fix manual
Ruido en journald	Log conciso (1 línea por chequeo OK, n líneas por incidente)

7. Anexos

7.1 Script de detección manual (referencia rápida)

```
#!/bin/bash
# /usr/local/sbin/check-virtual-ips.sh — POC v0
set -euo pipefail
```

```
IP_LIST="/usr/local/directadmin/data/admin/ip.list"
IFACE="$(ip -4 route show default | awk '{print $5; exit}')"
MISSING=()

while read -r ip; do
  [ -z "$ip" ] && continue
  if ! ip -4 addr show "$IFACE" | grep -qE "inet $ip(/| )"; then
    MISSING+=("$ip")
  fi
done < "$IP_LIST"

if [ "${#MISSING[@]}" -eq 0 ]; then
  exit 0  # OK
fi

logger -t startips-watchdog "IPs virtuales caídas: ${MISSING[*]}"
systemctl start startips.service

# verificar tras 2s
sleep 2
STILL_DOWN=()
for ip in "${MISSING[@]}"; do
  if ! ip -4 addr show "$IFACE" | grep -qE "inet $ip(/| )"; then
    STILL_DOWN+=("$ip")
  fi
done

if [ "${#STILL_DOWN[@]}" -eq 0 ]; then
  logger -t startips-watchdog "Auto-restart OK, IPs restauradas: ${MISSING[*]}"
  exit 0
fi

logger -t startips-watchdog "ALERTA: IPs siguen caídas tras restart: ${STILL_DOWN[*]}"
# TODO: enviar mail/Telegram
exit 1
```

7.2 Comandos de auditoría de la flota

```
# ¿Qué servidores DA tienen IPs adicionales?
for s in cloud500 srv120 srv121 kvm456 dar amazzal titrit bitatrader fail500; do
  n=$(ssh $s 'wc -l < /usr/local/directadmin/data/admin/ip.list 2>/dev/null')
  echo "$s: $n IPs"
done

# ¿Y tras un test de stress, qué dominios usan IPs no-primarias?
ssh <server> '
PRIMARY=$(ip route get 8.8.8.8 | awk "/src/ {print \$NF}")
for f in /usr/local/directadmin/data/users/*/domains/*.conf; do
  ip=$(grep "^ip=" "$f" | cut -d= -f2)
  if [ "$ip" != "$PRIMARY" ] && [ -n "$ip" ]; then
    dom=$(basename "$f" .conf)
    echo "  $ip → $dom"
  fi
done | sort | uniq -c | sort -rn
'
```

Aviso

Esta documentación y su contenido, no implica que funcione en tu caso o determinados casos. También implica que tienes conocimientos sobre lo que trata, y que en cualquier caso tienes copias de seguridad. El contenido el contenido se entrega, tal y como está, sin que ello implique ningún obligación ni responsabilidad por parte de Castris

Si necesitas soporte profesional puedes contratar con Castris soporte profesional.

Revision #1
Created 2026-04-30 19:08:14 UTC by Abkrim
Updated 2026-04-30 19:08:41 UTC by Abkrim