home-assistant / **plugin-dns** Public

Code · **Issues** 4 · Pull requests 1 · Actions · Security · Insights

New issue

Jump to bottom

Not resolving local host names #20

Closed **ssummer** opened this issue on Dec 26, 2020 · 104 comments

ssummer commented on Dec 26, 2020 • edited

In home assistant, if I use hostnames in configuration.yaml they fail to resolve (eg using platform snmp). Using the corresponding IP addresses works, but is far from ideal and means I have to use fixed IP addresses and configuration becomes less unreadable and requires more maintenance.

Trying to work out what is the root issue is, has led me to hassio-dns seemingly not working as I would expect. The homeassistant docker container appears to be using 172.30.32.3 for dns which is the hassio-dns container. From within the homeassistant container this local hostname fails to resolve:
dig pdu3.netality.co.uk
and I can see from the dig output it is going out to the Internet to do the DNS lookup.

But
dig pdu3.netality.co.uk @192.168.66.1
from home assistant does work (not surprisingly).

The strange thing is from within the hassio-dns container, fully qualified and unqualified local hostname lookups do work:
dig pdu3.netality.co.uk
dig pdu3
But
dig pdu3.netality.co.uk @172.30.32.3
does not work (unsurprisingly).

What I would expect is that fully qualified and unqualified hostnames should resolve as I have set the search domain to netality.co.uk and dns server to 192.168.66.1 in HassOS.

I don't think this has ever worked with hassio-dns.

Versions:
armhf-hassio-dns:2020.11.0
raspberrypi2-homeassistant:2020.12.1
armhf-hassio-supervisor:latest (2020.12.7)
HassOS: 5.9

👍 3

ssummer commented on Dec 26, 2020 Author

Was doing a bit more poking around in hassio-dns - I tried killing the coredns process (which caused a new one to be started) - it then started to work:
From within hassio-dns:
dig pdu3.netality.co.uk @172.30.32.3
dig pdu3 @172.30.32.3
dig pdu3.netality.co.uk
dig pdu3
all worked as expected and homeassistant container then starts to resolve local names (both fully qualified and unqualified):
dig pdu3.netality.co.uk
dig pdu3

But then after a few minutes it stops working again. So seems like something odd going on with coredns.

ssummer commented on Dec 26, 2020 Author

/etc/corefile

```
.:53 {
    log
    errors
    loop

    hosts /config/hosts {
        fallthrough
    }
    template ANY AAAA local.hass.io hassio {
        rcode NOERROR
    }
    mdns
    forward . dns://192.168.66.1 dns://192.168.66.1 dns://127.0.0.1:5553 {
        except local.hass.io
        policy sequential
        health_check 5s
    }
    fallback REFUSED . dns://127.0.0.1:5553
    fallback SERVFAIL . dns://127.0.0.1:5553
    fallback NXDOMAIN . dns://127.0.0.1:5553
    cache 10
}

.:5553 {
    log
    errors

    forward . tls://1.1.1.1 tls://1.0.0.1 {
        tls_servername cloudflare-dns.com
        except local.hass.io
        health_check 10s
    }
    cache 30
}
```

Zixim commented on Dec 27, 2020

Similar to #6 ?
Which is basically : if for some reason core-dns doesn't get an answer from your locally hosted server, core-dns will try the fallback. The fallback has of course no clue about your local hosts.
Core-DNS will not revert back to the dns server that you configured, but keep on using the fallback, until you do a `ha dns restart`

ssummer commented on Dec 27, 2020 • edited

Author

It didn't feel the same when I raised the issue, but now I can see it could be the same issue. In my case local names do not seem to resolve at all - when Hass starts, all hostname entries in configuration.yaml fail to load with resolve errors. But perhaps something else fails to resolve and it flips to the fallback before processing configuration.yaml?

Coredns shouldn't flip to permanently use the fallback - that is plainly broken behaviour.

#6 says it was fixed some time ago - but as you said and I see it appears it isn't. I don't understand this comment under this ticket:
"You will all time see that because of health check. Please use Home Assistant Container if you have an issue with that"
I'm using the standard Home Assistant image (ie HassOS).

It doesn't seem like too much to ask to for HA to use the specified local dns resolver and it seems like a lot of people have been affected by this for a long time. Also having hardcoded fallback server IPs that can't be changed doesn't seem right. Perhaps a better solution would at least allow the default fallback server to be overridden in HA and ideally, optionally disabled completely.

Zixim commented on Dec 27, 2020

I'm also using the default HAOS.
What I understand from the cryptic `"You will all time see that because of health check. Please use Home Assistant Container if you have an issue with that"` is that we're better off using any installation method that does not use core-DNS when wanting to have your own DNS setup.
Words fail me to describe how wrong this reasoning is...but hey...it's open source, and you can always issue a pull request & other snide remarks...

TLDR : HAOS is the *only* service that has recurring dns issues in my multi-VM home setup. coreDNS is broken, dev can't/won't fix.
enf of.

ssummer commented on Dec 27, 2020

Author

I don't see the point of even having the option to set the name server in HA, if it doesn't work and it doesn't use it. It's also very misleading to instead use another entirely different, undocumented name server and I don't really want another random company to see my (supposedly internal to my network) dns lookups.
I agree this is the only application that has this problem that I am using/have ever used. I am using the most simple HA setup - a single raspberry pi dedicated to running the stock HAOS.
I repeat - it doesn't seem unreasonable to expect that HA uses the name server that is specified.

ssummer commented on Dec 28, 2020 • edited

Author

I think I have found something;

```
forward . dns://192.168.66.1 dns://192.168.66.1 dns://127.0.0.1:5553
```

According to the coredns docs this will load balance requests amongst each 'upstream' - this will result in 1 in 3 dns requests going to the second coredns instance (port 5553) which will use cloudflare server - so this will certainly cause errors for local hostnames.
I think the 'dns://127.0.0.1:5553' from the above line should be removed - the fallback plugin entries will ensure that if the local dns server doesn't have the answer it will go to cloudflare.

👍 1

ssummer closed this as completed on Dec 28, 2020

ssummer reopened this on Dec 28, 2020

ssummer commented on Dec 28, 2020

Author

So I tried removing dns://127.0.0.1:5553 from the first forward line in /etc/corefile and restarted coredns. I then changed all my IP addresses to hostnames in configuration.yaml and then restarted home assistant and success - all the hostnames now resolve fine. However if I restart the host my changes will get overwritten again. So seems like the /usr/share/corefile.tempio file in plugin-dns would need changing to implement this properly:

```
forward . {{ JoinString .servers " " }} {{ if len .locals | eq 0 }}dns://127.0.0.11{{ else }}{{ JoinString .locals " " }}{{ end }} dns://127.0.0.1:5553 {
```

However there may be another issue with this line - if the "servers" and "locals" in /config/coredns.json are different (not sure where each of these comes from) then a similar issue will still occur as dns lookups will flip between those two - in my case both are the same, so it's fine: dns://192.168.66.1
It looks to me like it was expected that coredns would try each upstream in turn - but that is not the case - it treats them all as equals and will load balance across them. Sequential option has been chosen, so first dns query one will use the first upstream, the second dns query will use the second, the third will use the third upstream and then the fourth query will wrap round and use the first upstream and so on.

👍 1

craSH commented on Jan 17, 2021 • edited

Just chiming in here to say that I'm running in to the same problems on my home network, and I found this issue once I got down in to the coredns container's config (I didn't know exactly where to look for issues before getting that deep in). I agree with everything posted here so far, and am surprised there isn't more activity here - I imagine lots of HA users have their setup communicating with local devices via internal hostnames.

My local network uses a subdomain from a real internet zone, not a fake TLD or the mDNS .local zone (e.g. lan.example.com)

I have not yet looked in to how the main HA config makes its way into the coredns config, but if we could have an option to entirely disable the use of 3rd party DoT that really seems desirable. One of the reasons I use HA vs. another system is that I thought it left me in control of my data, and now this update has started sending DNS requests for my local resources to a third party DoT server without my knowing or wanting that to occur. I run my own local recursors and authoritative DNS for those resources, and would prefer HA use those systems which I control.

@ssummer and **@Zixim** I'm happy to help debug or provide more info for my setup, and dig into making some test PRs and the like (but I haven't done any dev/contribution to the HA codebases in the past).

Edit: Changed DoH references to DoT, as the coredns config is using `tls://` , not `https://` for it's external DNS.

craSH commented on Jan 17, 2021

I found another problem which I think may actually be the root cause of this issue. I noticed that I still actually see all the requests for my local resources hitting my local DNS server, but all responses are coming back SERVFAIL (DNS level failure) - it seem that coredns is requiring DNSSEC signed records from local resolvers. I performed a packet capture while running the following queries for my local PurpleAir device, from the HassIO host (via the HACP ssh/terminal addon) to demonstrate.

`dig purpleair-5df.hamwan.tlr.im.` - use system resolver, which ends up routing the request through coredns in the plugin-dns container.

This causes a query to hit my local DNS server like so, as decoded with Wireshark:

```
Frame 1: 110 bytes on wire (880 bits), 110 bytes captured (880 bits)
Ethernet II, Src: Raspberr_7c:67:22 (dc:a6:32:7c:67:22), Dst: Wibrain_45:44:90 (00:1e:06:45:44:90)
Internet Protocol Version 4, Src: homeassistant.lan.tlr.im (10.9.7.13), Dst: dnsdist.lan.tlr.im (10.9.7.21)
User Datagram Protocol, Src Port: 39937 (39937), Dst Port: domain (53)
Domain Name System (query)
    Transaction ID: 0x6684
    Flags: 0x0120 Standard query
        0... .... .... .... = Response: Message is a query
        .000 0... .... .... = Opcode: Standard query (0)
        .... ..0. .... .... = Truncated: Message is not truncated
        .... ...1 .... .... = Recursion desired: Do query recursively
        .... .... .0.. .... = Z: reserved (0)
        .... .... ..1. .... = AD bit: Set
        .... .... ...0 .... = Non-authenticated data: Unacceptable
    Questions: 1
    Answer RRs: 0
    Authority RRs: 0
    Additional RRs: 1
    Queries
        purpleair-5df.hamwan.tlr.im: type A, class IN
            Name: purpleair-5df.hamwan.tlr.im
            [Name Length: 27]
            [Label Count: 4]
            Type: A (Host Address) (1)
            Class: IN (0x0001)
    Additional records
        <Root>: type OPT
            Name: <Root>
            Type: OPT (41)
            UDP payload size: 2048
            Higher bits in extended RCODE: 0x00
            EDNS0 version: 0
            Z: 0x8000
                1... .... .... .... = DO bit: Accepts DNSSEC security RRs
                .000 0000 0000 0000 = Reserved: 0x0000
            Data length: 12
            Option: COOKIE
                Option Code: COOKIE (10)
                Option Length: 8
                Option Data: b770ff1ea6bc60a0
                Client Cookie: b770ff1ea6bc60a0
                Server Cookie: <MISSING>
    [Response In: 2]
```

And the corresponding response, which in my case is a **SERVFAIL** because my local DNS server does have DNSSEc signed records available for my local zone (which I suspect is the case for most users with local DNS from their home routers/etc!)

```
Domain Name System (response)
    Transaction ID: 0x6684
    Flags: 0x8582 Standard query response, Server failure
        1... .... .... .... = Response: Message is a response
        .000 0... .... .... = Opcode: Standard query (0)
        .... .1.. .... .... = Authoritative: Server is an authority for domain
        .... ..0. .... .... = Truncated: Message is not truncated
        .... ...1 .... .... = Recursion desired: Do query recursively
        .... .... 1... .... = Recursion available: Server can do recursive queries
        .... .... .0.. .... = Z: reserved (0)
        .... .... ..0. .... = Answer authenticated: Answer/authority portion was not authenticated by the server
        .... .... ...0 .... = Non-authenticated data: Unacceptable
        .... .... .... 0010 = Reply code: Server failure (2)
    Questions: 1
    Answer RRs: 0
    Authority RRs: 0
    Additional RRs: 1
    Queries
        purpleair-5df.hamwan.tlr.im: type A, class IN
            Name: purpleair-5df.hamwan.tlr.im
            [Name Length: 27]
            [Label Count: 4]
            Type: A (Host Address) (1)
            Class: IN (0x0001)
    Additional records
        <Root>: type OPT
            Name: <Root>
            Type: OPT (41)
            UDP payload size: 1232
            Higher bits in extended RCODE: 0x00
            EDNS0 version: 0
            Z: 0x8000
                1... .... .... .... = DO bit: Accepts DNSSEC security RRs
                .000 0000 0000 0000 = Reserved: 0x0000
            Data length: 0
    [Request In: 1]
    [Time: 0.006167000 seconds]
```

Note near the bottom within the OPT record, the `DO` flag is set "DO bit: Accepts DNSSEC security RRs" - from dig(1):

> +[no]dnssec
> Requests DNSSEC records be sent by setting the DNSSEC OK bit (DO) in the OPT record in the additional section of the query.

I can stimulate the same response if I force the DO flag to be set with dig(1) and specify my local DNS server explicitly:

`dig purpleair-5df.hamwan.tlr.im. @10.9.7.21 +dnssec` - use my local network resolver directly, with the same `DO` flag set like the coredns requests.

The same request/response is observed in this case. I'll show the dig output here (the wireshark decoded results are essentially the same):

```
; <<>> DiG 9.16.6 <<>> purpleair-5df.hamwan.tlr.im. @10.9.7.21 +dnssec
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: SERVFAIL, id: 65085
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags: do; udp: 1232
;; QUESTION SECTION:
;purpleair-5df.hamwan.tlr.im.   IN      A

;; Query time: 9 msec
;; SERVER: 10.9.7.21#53(10.9.7.21)
;; WHEN: Sun Jan 17 10:20:58 PST 2021
;; MSG SIZE  rcvd: 56
```

Now, I can verify the same request works without the `DO` flag set by passing `+nodnssec` to dig(1).

`dig purpleair-5df.hamwan.tlr.im. @10.9.7.21 +nodnssec` - use my local network resolver directly, without requesting DNSSEC records:

```
; <<>> DiG 9.16.6 <<>> purpleair-5df.hamwan.tlr.im. @10.9.7.21 +nodnssec
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 41365
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
;; QUESTION SECTION:
;purpleair-5df.hamwan.tlr.im.   IN      A

;; ANSWER SECTION:
purpleair-5df.hamwan.tlr.im. 900 IN     A       192.168.88.35

;; Query time: 9 msec
;; SERVER: 10.9.7.21#53(10.9.7.21)
;; WHEN: Sun Jan 17 10:21:49 PST 2021
;; MSG SIZE  rcvd: 72
```

From what I can tell, the CoreDNS configuration as currently defined will cause any local queries which return SERVFAIL to immediately retry at the external DNS over TLS server (e.g. Cloudflare) due to this bit of configuration in the `.:53 {}` block:

```
fallback SERVFAIL . dns://127.0.0.1:5553
```

This is probably a violation of the EDNS0 RFC (a SHOULD statement in it, at least) (https://tools.ietf.org/html/rfc2671#section-5.3):

> 5.3. Responders who do not understand these protocol extensions are
> expected to send a response with RCODE NOTIMPL, FORMERR, or
> SERVFAIL. Therefore use of extensions should be "probed" such that
> a responder who isn't known to support them be allowed a retry with
> no extensions if it responds with such an RCODE. If a responder's
> capability level is cached by a requestor, a new probe should be
> sent periodically to test for changes to responder capability.

It is not clear to me how one configured CoreDNS to not set the `DO` flag for it's initial requests, but that's probably not the correct approach anyways - I think making the CoreDNS configuration somehow retry without the flag upon receiving an initial response of NOTIMPL, FORMERR, or SERVFAIL is ideal.

To further verify this, I suppose I'll setup DNSSec for my internal zones and see if that "fixes" it, too.

craSH commented on Jan 17, 2021

Once upgrading my local zones to support DNSSEC, this issue does go away for me, at least somewhat validating my analysis above. But I still want to emphasize this is important to fix, as most users will probably not have DNSSEC signed local DNS in their local home networks.

New dig results after enabling DNSSSEC for my `hamwan.tlr.im.` zone:

Direct query to my DNS server demonstrating new DNSSEC support (`DO` flag set):

```
dig purpleair-5df.hamwan.tlr.im. @10.9.7.21 +dnssec

; <<>> DiG 9.16.6 <<>> purpleair-5df.hamwan.tlr.im. @10.9.7.21 +dnssec
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 42046
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags: do; udp: 1232
;; QUESTION SECTION:
;purpleair-5df.hamwan.tlr.im.   IN      A

;; ANSWER SECTION:
purpleair-5df.hamwan.tlr.im. 900 IN     RRSIG   A 13 4 900 20210128000000 20210107000000 55748 hamwan.tlr.im. C2AjuHRgGOAe/etgrasytF5vAXb5dHONUXY81Y2KGdKqGXxb5XsKp2V/
7quJXooLryvZB6QcQjJMzXUgZaGyYw==
purpleair-5df.hamwan.tlr.im. 900 IN     A       192.168.88.35

;; Query time: 9 msec
;; SERVER: 10.9.7.21#53(10.9.7.21)
;; WHEN: Sun Jan 17 11:55:26 PST 2021
;; MSG SIZE  rcvd: 181
```

Query to local DNS resolver (coredns / plugin-dns) with explicit dnssec requested, and returned properly (`DO` flag set):

```
dig purpleair-5df.hamwan.tlr.im. +dnssec

; <<>> DiG 9.16.6 <<>> purpleair-5df.hamwan.tlr.im. +dnssec
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 23367
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags: do; udp: 4096
; COOKIE: 40e4e84a9c3a9bca (echoed)
;; QUESTION SECTION:
;purpleair-5df.hamwan.tlr.im.   IN      A

;; ANSWER SECTION:
purpleair-5df.hamwan.tlr.im. 10 IN      A       192.168.88.35
purpleair-5df.hamwan.tlr.im. 10 IN      RRSIG   A 13 4 900 20210128000000 20210107000000 55748 hamwan.tlr.im. C2AjuHRgGOAe/etgrasytF5vAXb5dHONUXY81Y2KGdKqGXxb5XsKp2V/
7quJXooLryvZB6QcQjJMzXUgZaGyYw==

;; Query time: 0 msec
;; SERVER: 172.30.32.3#53(172.30.32.3)
;; WHEN: Sun Jan 17 11:56:18 PST 2021
;; MSG SIZE  rcvd: 247
```

Query to local DNS resolver (coredns / plugin-dns) without explicit dnssec requested (no `DO` flag set):

```
dig purpleair-5df.hamwan.tlr.im.

; <<>> DiG 9.16.6 <<>> purpleair-5df.hamwan.tlr.im.
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 10039
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: d3a85ca6efd535be (echoed)
;; QUESTION SECTION:
;purpleair-5df.hamwan.tlr.im.   IN      A

;; ANSWER SECTION:
purpleair-5df.hamwan.tlr.im. 8  IN      A       192.168.88.35

;; Query time: 0 msec
;; SERVER: 172.30.32.3#53(172.30.32.3)
;; WHEN: Sun Jan 17 11:57:16 PST 2021
;; MSG SIZE  rcvd: 111
```

Zixim commented on Jan 17, 2021 • edited

@craSH

> **@ssummer** and **@Zixim** I'm happy to help debug or provide more info for my setup, and dig into making some test PRs and the like (but I haven't done any dev/contribution to the HA codebases in the past).

I feel like I mis-explained : the bug-report I opened about this whole coreDNS mess was closed by the dev, after a kind of **won't fix** remark. Perhaps you could start from scratch with a new bug-report, we might get lucky, with 2020 being behind us now.

I wrote "_it's open source, and you can always issue a pull request _" because that's a typical answer users get from devs, when the dev feels like the users are asking too much. It hasn't happened (yet) in this case, but the way in which the dev acknowledged the bug & told me to use another installation method sure did taste like that.

craSH commented on Jan 17, 2021

@Zixim Agreed, I actually followed you there, but was trying to be an open minded as possible in my first post about this :D
Personally, I'd probably opt for a pull request that undoes adding the coredns plugin at all, I don't quite see the benefit for this project.. but that's a strong opinion in of itself.

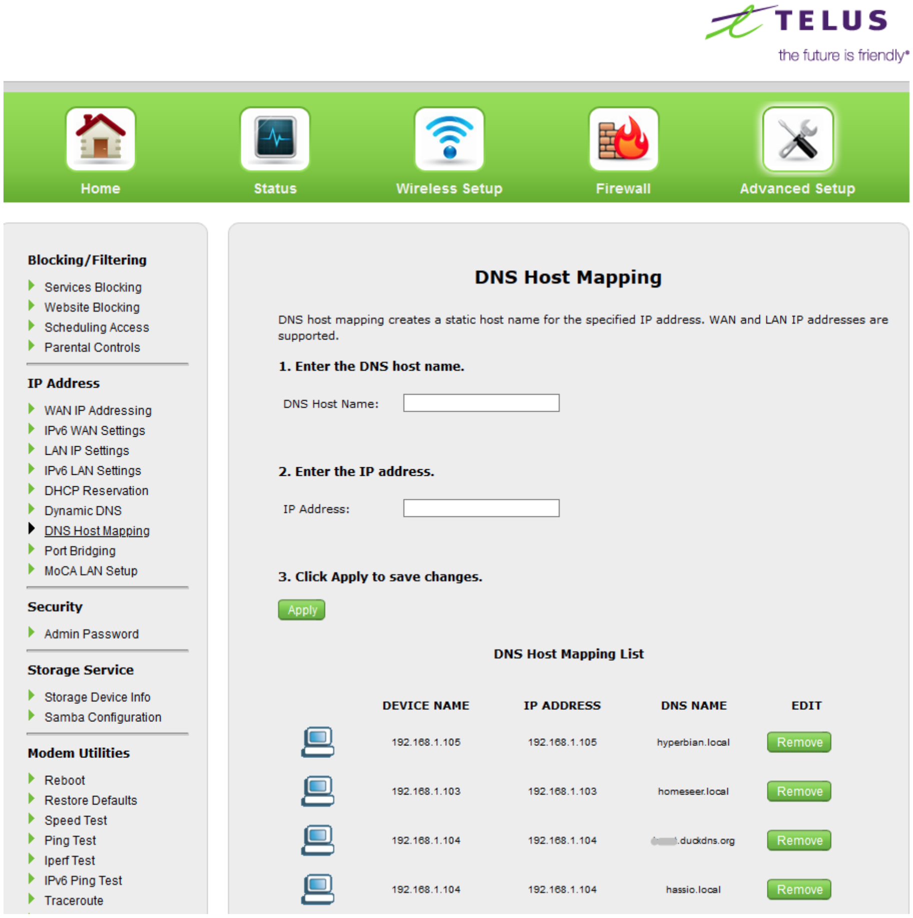
Is there a chat server folks hang out on we could discuss this further, without flooding GH issue threads?

danielbrunt57 commented on Jan 19, 2021 • edited

I've been troubleshooting 1st time setup of a REST sensor with this error off and on for a few weeks now:

```
Logger: homeassistant.components.rest.data
Source: components/rest/data.py:67
Integration: rest (documentation, issues)
First occurred: 9:23:26 PM (21 occurrences)
Last logged: 10:15:57 PM
Error fetching data: https://[my_domain].duckdns.org:8123/api/config failed with [Errno -2] Name does not resolve
```

and I've finally ended up here. None of my router defined static entries can be resolved by ha dns so i think I am in the right place. I know TELUS will just say "dns sec?? what's that?"



DNS Host Mapping

DNS host mapping creates a static host name for the specified IP address. WAN and LAN IP addresses are supported.

1. Enter the DNS host name.

DNS Host Name:

2. Enter the IP address.

IP Address:

3. Click Apply to save changes.

DNS Host Mapping List

DEVICE NAME	IP ADDRESS	DNS NAME	EDIT
192.168.1.105	192.168.1.105	hyperbian.local	Remove
192.168.1.103	192.168.1.103	homeseer.local	Remove
192.168.1.104	192.168.1.104	[redacted].duckdns.org	Remove
192.168.1.104	192.168.1.104	hassio.local	Remove

I've removed my split DNS entry for [my_domain].duckdns.org and let it resolve to the public IP but the REST sensor now says

```
Logger: homeassistant.components.rest.data
Source: components/rest/data.py:67
Integration: rest (documentation, issues)
First occurred: 10:37:16 PM (1 occurrences)
Last logged: 10:37:16 PM
Error fetching data: https://[my_domain].duckdns.org:8123/api/config failed with
```

since the router does not support loop-back.

I managed to finally get my REST sensor working using the following config:

```
  - platform: rest
    name: Hassio Configuration
#    resource: !secret resource_hassio_main_config
    resource: https://homeassistant.local.hass.io:8123/api/config
    verify_ssl: false
    authentication: basic
    value_template: >
      {{ value_json.version }}
    json_attributes:
      - components
      - unit_system
      - config_dir
      - version
    headers:
      Content-Type: application/json
      Authorization: !secret api_bearer_token
      User-Agent: Home Assistant REST sensor
```

AND I now know more about `ha dns` . Forcing DNSSEC is definitely wrong in my opinion. There should at least be a way to disable it for local DNS...

superm1 mentioned this issue on Jan 19, 2021

DNS Plugin is ignoring my local DNS on startup #17

Closed

Zixim commented on Jan 25, 2021

@maiko29 your issue absolutely does not sound like any kind of DNS issue.
You should seek help on the forum or on discord.

McGiverGim commented on Feb 12, 2021

Sorry, I don't understand exactly all of this (my knowledge about networking or linux is limited) but I think I have the same problem commented here.

My local device names are not being resolved. They were some weeks ago, but I don't know when this changed.

My router acts as DNS, and is assigned by DHCP to all the devices, including the raspberry pi with Home Assistant, and things like `localdevicename.mydomain` were working, but now they don't. Other computers in my network work continue resolving the names ok, so it seems a problem with the DNS used in Home Assistant.

Something I can do/help with to fix this?

McGiverGim commented on Feb 12, 2021 • edited

If it helps... the nslookup resolves the name, but it throws an error:

```
➜  ~ nslookup camara-pasillo.piminet
Server:         172.30.32.3
Address:        172.30.32.3#53

Name:   camara-pasillo.piminet
Address: 192.168.100.20
** server can't find camara-pasillo.piminet: NXDOMAIN

➜  ~ ping camara-pasillo.piminet
ping: bad address 'camara-pasillo.piminet'
```

As you can see, the address is finding the correct IP (192.168.100.20 in this case), but it throws an NXDOMAIN error later that produces that this IP is not used.
As I said, my knowledge about DNS and Linux is very reduced, so I don't know how to interpret that.

danielbrunt57 commented on Feb 12, 2021

My understanding of this problem is HA requires a DNSSEC lookup. My router does not support that so I had to add & configure the AdGuard Home addon for HA to use for its secure DNS lookups. I could not find any other way...

Zixim commented on Feb 13, 2021

> HA requires a DNSSEC lookup.

Don't think so.
My local dns resolver (Pi-hole) doesn't use DNSSEC for local hostname resolution.
All works fine, untill Broken CoreDNS decides it needs to stop using my dns server and instead start using a HARDCODED dns server on the internet, which of course has no idea about my local hosts.
Worse, CoreDNS then leaks internal hostnames to some server in the internet, and we can't stop it doing that.

No developer wants to fix this glaring security bug, so we should try to get the feature request voted up :
https://community.home-assistant.io/t/improve-privacy-stop-using-hardcoded-dns/273496
Please help.

McGiverGim commented on Feb 13, 2021

I've found a workaround modifying the coredns template... I don't know if this can have some drawbacks, but it seems to work and I have not found any problem until now.

In the `hassio_dns` docker, that contains the `coredns` server, there is a template file with the configuration of the `coredns` . I've modified that:

plugin-dns/rootfs/usr/share/tempio/corefile
Lines 11 to 13 in b3827bb

```
11      template ANY AAAA local.hass.io hassio {
12          rcode NOERROR
13      }
```

adding my local domain (piminet) at the end:

```
template ANY AAAA local.hass.io hassio piminet {
        rcode NOERROR
}
```

In this way it returns `NOERROR` in place of `NXDOMAIN` and now it works and resolves the local domains without problem, at least in my case 😀 .

If this solution is ok, maybe some real Home Assistant developer can add a new option to the `ha dns` CLI command to add the local domain here automatically.

❤️ 4

Zixim commented on Feb 13, 2021

on next update, your edits will get wiped out

McGiverGim commented on Feb 13, 2021

I know. This is the reason why I ask to include it in the base by a real Home Assistant developer, if the solution is acceptable and does not break other things.

stale bot commented on May 12, 2021

This issue has been automatically marked as stale because it has not had recent activity. It will be closed if no further activity occurs. Thank you for your contributions.

stale bot added the stale label on May 12, 2021

Zixim commented on May 12, 2021

not solved. Need developer input.

👍 13

stale bot removed the **stale** label on May 12, 2021

jherby2k commented on Jul 6, 2021

Bumping so this doesn't go stale. Surprised this isn't getting more traction!

👍 4

57 hidden items
Load more...

bentasker commented on Mar 23

It's very likely because the queries are being incorrectly forwarded onto Cloudflare by coredns.

I tried to repro the behaviour you're seeing but failed (I added a record to my local server for `windows-11.brunt.lan`):

```
➜  ~ docker exec -it hassio_dns bash
bash-5.1# nslookup windows-11.brunt.lan 127.0.0.1
Server:         127.0.0.1
Address:        127.0.0.1#53

Name:   windows-11.brunt.lan
Address: 192.168.3.55
```

If you query against Cloudflare, you get an `NXDOMAIN` (as you should) though, so it's not clear why that turns into a `SERVFAIL` .

Sounds like you've already got it, but if you install the privileged SSH addon (https://github.com/hassio-addons/addon-ssh) that'll give you access to `docker` , which means you can then do `docker logs hassio_dns` for a clue of what's going on.

I would say, though, that I'm using my workaround (and I blocked supervisor auto-updates) so it's quite possible there's a new "base" coredns configuration file floating about. If you want to send me yours I'll try find time to test against that.

> supervisor is okay...

In the output there you've got two servers - docker's built in one, and the hassio_dns container, the latter fails.

> After some more experiments, I now wonder whether I'm looking at a completely different issue. I noticed that from within the hassio_dns container, nslookup media-server.lan works fine (with either a stock or a modified corefile). It also works from within the homeassistant container.

You'll like this, not a lot, but you'll like this.

The `hassio_dns` container does not use it's own service to resolve names - it forwards them onto Docker's built in resolver, so queries will bypass `coredns` (which helps tell us that's where your problem lies).

Either there's something nuts in the coredns config, or something's changed in `coredns` itself.

bentasker commented on Mar 23

Actually, going back over your screenshots, I think there's two issues being conflated here.

You've got DNSSEC turned on for `.lan` which may well be why you're getting a SERVFAIL (because coredns is failing to validate the RRs).

When you first tested, you were getting NXDOMAINs (like me). That'll likely be because your queries were going to CF.

Now, they're probably going to your local recursor but responses for `.lan` aren't properly authenticated (at least in coredns's view)

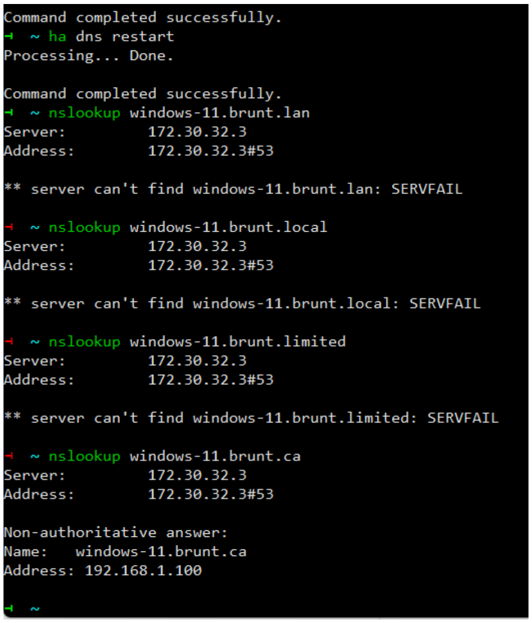
Does

```
nslookup windows-11.brunt.local
```

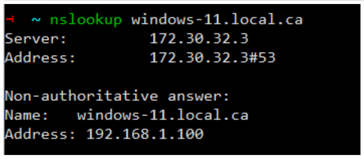
work from within the homeassistant container?

danielbrunt57 commented on Mar 23 • edited

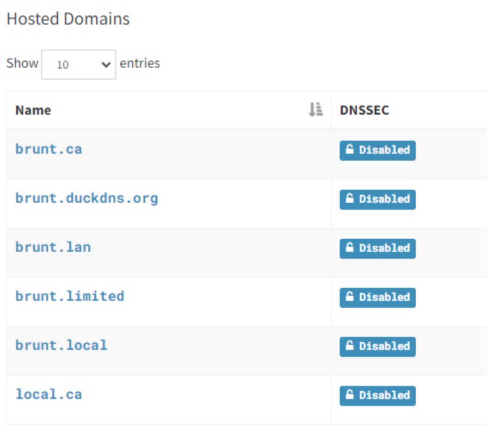
Thank you for your valuable feedback!! No, it does not work. I've also removed DNSSEC on .lan.



```
Command completed successfully.
➜ ~ ha dns restart
Processing... Done.

Command completed successfully.
➜ ~ nslookup windows-11.brunt.lan
Server:         172.30.32.3
Address:        172.30.32.3#53

** server can't find windows-11.brunt.lan: SERVFAIL

➜ ~ nslookup windows-11.brunt.local
Server:         172.30.32.3
Address:        172.30.32.3#53

** server can't find windows-11.brunt.local: SERVFAIL

➜ ~ nslookup windows-11.brunt.limited
Server:         172.30.32.3
Address:        172.30.32.3#53

** server can't find windows-11.brunt.limited: SERVFAIL

➜ ~ nslookup windows-11.brunt.ca
Server:         172.30.32.3
Address:        172.30.32.3#53

Non-authoritative answer:
Name:   windows-11.brunt.ca
Address: 192.168.1.100

➜ ~
```



```
➜ ~ nslookup windows-11.local.ca
Server:         172.30.32.3
Address:        172.30.32.3#53

Non-authoritative answer:
Name:   windows-11.local.ca
Address: 192.168.1.100
```

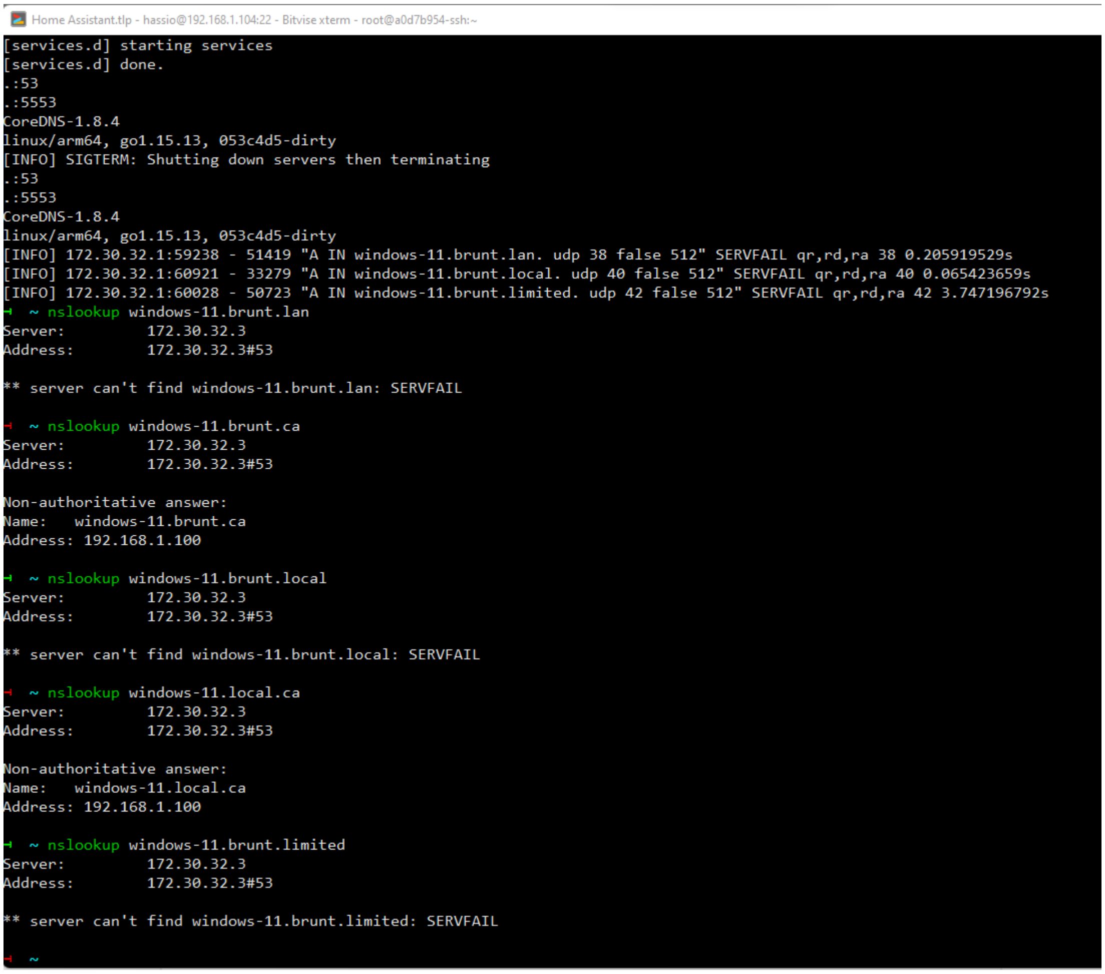


Hosted Domains

Show 10 entries

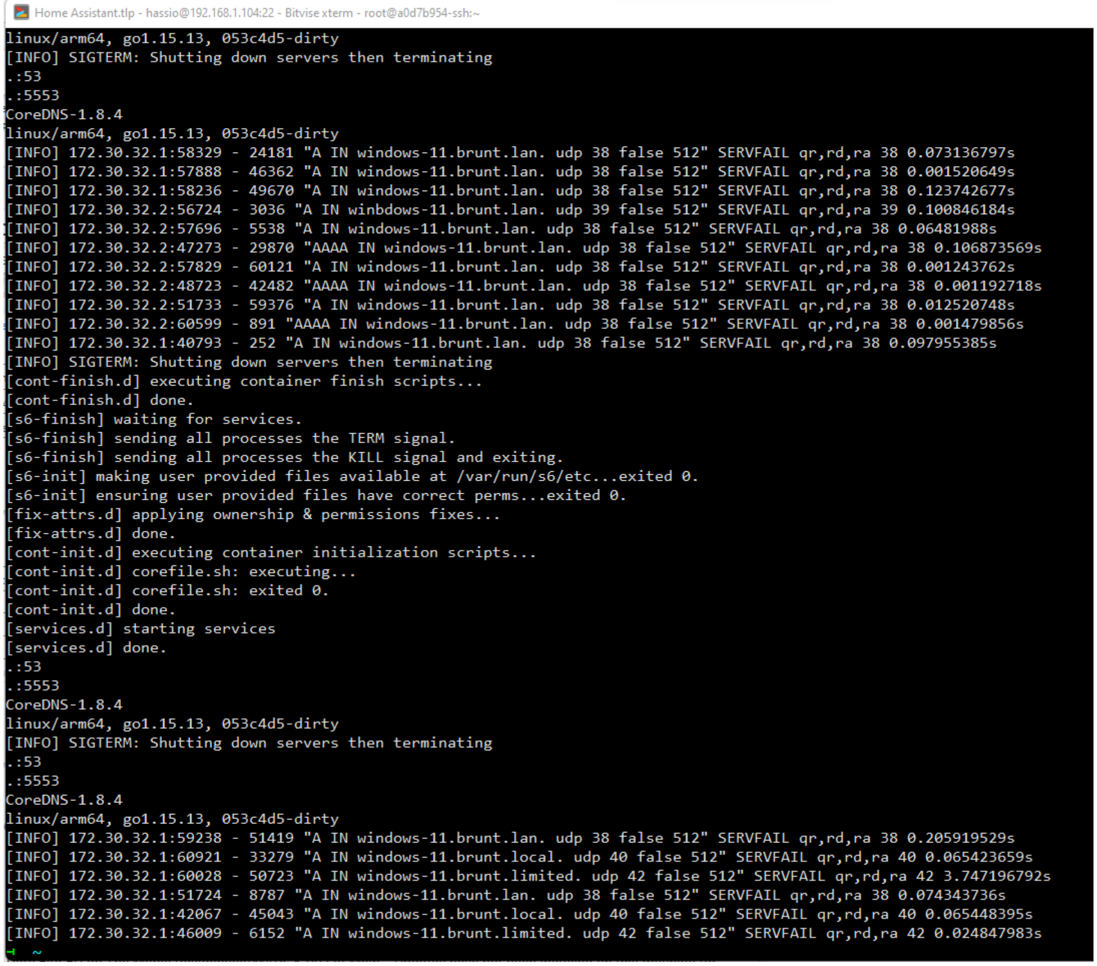
Name	DNSSEC
brunt.ca	🔓 Disabled
brunt.duckdns.org	🔓 Disabled
brunt.lan	🔓 Disabled
brunt.limited	🔓 Disabled
brunt.local	🔓 Disabled
local.ca	🔓 Disabled

danielbrunt57 commented on Mar 23


Home Assistant.tlp - hassio@192.168.1.104:22 - Bitvise xterm - root@a0d7b954-ssh:~

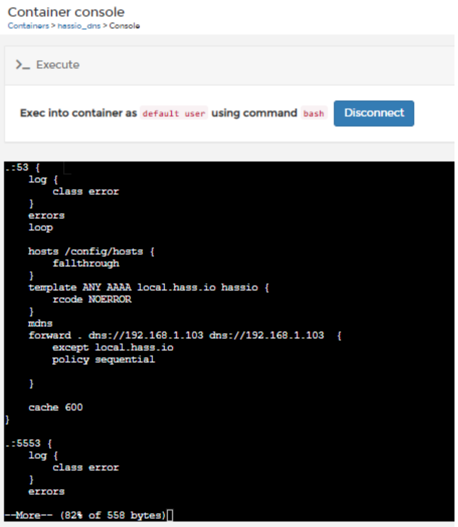
```
[services.d] starting services
[services.d] done.
.:53
.:5553
CoreDNS-1.8.4
linux/arm64, go1.15.13, 053c4d5-dirty
[INFO] SIGTERM: Shutting down servers then terminating
.:53
.:5553
CoreDNS-1.8.4
linux/arm64, go1.15.13, 053c4d5-dirty
[INFO] 172.30.32.1:59238 - 51419 "A IN windows-11.brunt.lan. udp 38 false 512" SERVFAIL qr,rd,ra 38 0.205919529s
[INFO] 172.30.32.1:60921 - 33279 "A IN windows-11.brunt.local. udp 40 false 512" SERVFAIL qr,rd,ra 40 0.065423659s
[INFO] 172.30.32.1:60028 - 50723 "A IN windows-11.brunt.limited. udp 42 false 512" SERVFAIL qr,rd,ra 42 3.747196792s
➜  ~ nslookup windows-11.brunt.lan
Server:         172.30.32.3
Address:        172.30.32.3#53

** server can't find windows-11.brunt.lan: SERVFAIL

➜  ~ nslookup windows-11.brunt.ca
Server:         172.30.32.3
Address:        172.30.32.3#53

Non-authoritative answer:
Name:   windows-11.brunt.ca
Address: 192.168.1.100

➜  ~ nslookup windows-11.brunt.local
Server:         172.30.32.3
Address:        172.30.32.3#53

** server can't find windows-11.brunt.local: SERVFAIL

➜  ~ nslookup windows-11.local.ca
Server:         172.30.32.3
Address:        172.30.32.3#53

Non-authoritative answer:
Name:   windows-11.local.ca
Address: 192.168.1.100

➜  ~ nslookup windows-11.brunt.limited
Server:         172.30.32.3
Address:        172.30.32.3#53

** server can't find windows-11.brunt.limited: SERVFAIL

➜  ~
```

Only the .l* queries are processed by hassio_dns (and fail):



Home Assistant.tlp - hassio@192.168.1.104:22 - Bitvise xterm - root@a0d7b954-ssh:~

```
linux/arm64, go1.15.13, 053c4d5-dirty
[INFO] SIGTERM: Shutting down servers then terminating
.:53
.:5553
CoreDNS-1.8.4
linux/arm64, go1.15.13, 053c4d5-dirty
[INFO] 172.30.32.1:58329 - 24181 "A IN windows-11.brunt.lan. udp 38 false 512" SERVFAIL qr,rd,ra 38 0.073136797s
[INFO] 172.30.32.1:57888 - 46362 "A IN windows-11.brunt.lan. udp 38 false 512" SERVFAIL qr,rd,ra 38 0.001520649s
[INFO] 172.30.32.1:58236 - 49670 "A IN windows-11.brunt.lan. udp 38 false 512" SERVFAIL qr,rd,ra 38 0.123742677s
[INFO] 172.30.32.2:56724 - 3036 "A IN winbdows-11.brunt.lan. udp 39 false 512" SERVFAIL qr,rd,ra 39 0.100846184s
[INFO] 172.30.32.2:57696 - 5538 "A IN windows-11.brunt.lan. udp 38 false 512" SERVFAIL qr,rd,ra 38 0.06481988s
[INFO] 172.30.32.2:47273 - 29870 "AAAA IN windows-11.brunt.lan. udp 38 false 512" SERVFAIL qr,rd,ra 38 0.106873569s
[INFO] 172.30.32.2:57829 - 60121 "A IN windows-11.brunt.lan. udp 38 false 512" SERVFAIL qr,rd,ra 38 0.001243762s
[INFO] 172.30.32.2:48723 - 42482 "AAAA IN windows-11.brunt.lan. udp 38 false 512" SERVFAIL qr,rd,ra 38 0.001192718s
[INFO] 172.30.32.2:51733 - 59376 "A IN windows-11.brunt.lan. udp 38 false 512" SERVFAIL qr,rd,ra 38 0.012520748s
[INFO] 172.30.32.2:60599 - 891 "AAAA IN windows-11.brunt.lan. udp 38 false 512" SERVFAIL qr,rd,ra 38 0.001479856s
[INFO] 172.30.32.1:40793 - 252 "A IN windows-11.brunt.lan. udp 38 false 512" SERVFAIL qr,rd,ra 38 0.097955385s
[INFO] SIGTERM: Shutting down servers then terminating
[cont-finish.d] executing container finish scripts...
[cont-finish.d] done.
[s6-finish] waiting for services.
[s6-finish] sending all processes the TERM signal.
[s6-finish] sending all processes the KILL signal and exiting.
[s6-init] making user provided files available at /var/run/s6/etc...exited 0.
[s6-init] ensuring user provided files have correct perms...exited 0.
[fix-attrs.d] applying ownership & permissions fixes...
[fix-attrs.d] done.
[cont-init.d] executing container initialization scripts...
[cont-init.d] corefile.sh: executing...
[cont-init.d] corefile.sh: exited 0.
[cont-init.d] done.
[services.d] starting services
[services.d] done.
.:53
.:5553
CoreDNS-1.8.4
linux/arm64, go1.15.13, 053c4d5-dirty
[INFO] SIGTERM: Shutting down servers then terminating
.:53
.:5553
CoreDNS-1.8.4
linux/arm64, go1.15.13, 053c4d5-dirty
[INFO] 172.30.32.1:59238 - 51419 "A IN windows-11.brunt.lan. udp 38 false 512" SERVFAIL qr,rd,ra 38 0.205919529s
[INFO] 172.30.32.1:60921 - 33279 "A IN windows-11.brunt.local. udp 40 false 512" SERVFAIL qr,rd,ra 40 0.065423659s
[INFO] 172.30.32.1:60028 - 50723 "A IN windows-11.brunt.limited. udp 42 false 512" SERVFAIL qr,rd,ra 42 3.747196792s
[INFO] 172.30.32.1:51724 - 8787 "A IN windows-11.brunt.lan. udp 38 false 512" SERVFAIL qr,rd,ra 38 0.074343736s
[INFO] 172.30.32.1:42067 - 45043 "A IN windows-11.brunt.local. udp 40 false 512" SERVFAIL qr,rd,ra 40 0.065448395s
[INFO] 172.30.32.1:46009 - 6152 "A IN windows-11.brunt.limited. udp 42 false 512" SERVFAIL qr,rd,ra 42 0.024847983s
➜  ~
```

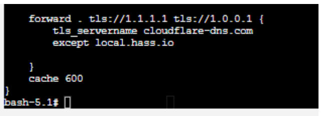


Container console

Containers > hassio_dns > Console

Exec into container as `default user` using command `bash`

```
.:53 {
    log {
        class error
    }
    errors
    loop

    hosts /config/hosts {
        fallthrough
    }
    template ANY AAAA local.hass.io hassio {
        rcode NOERROR
    }
    mdns
    forward . dns://192.168.1.103 dns://192.168.1.103  {
        except local.hass.io
        policy sequential

    }

    cache 600
}

.:5553 {
    log {
        class error
    }
    errors

--More-- (82% of 558 bytes)
```



```
    forward . tls://1.1.1.1 tls://1.0.0.1 {
        tls_servername cloudflare-dns.com
        except local.hass.io

    }
    cache 600
}
bash-5.1#
```

danielbrunt57 commented on Mar 23

bash-5.1# cat /etc/corefile

```
.:53 {
    log {
        class error
    }
    errors
    loop

    hosts /config/hosts {
        fallthrough
    }
    template ANY AAAA local.hass.io hassio {
        rcode NOERROR
    }
    mdns
    forward . dns://192.168.1.103 dns://192.168.1.103  {
        except local.hass.io
        policy sequential

    }

    cache 600
}

.:5553 {
    log {
        class error
    }
    errors

    forward . tls://1.1.1.1 tls://1.0.0.1 {
        tls_servername cloudflare-dns.com
        except local.hass.io

    }
    cache 600
}
```

bentasker commented on Mar 23

Config looks normal enough. Unfortunately the console log doesn't tell us much.

I don't think it'll be this, but just to be safe, can you do

```
cat /config/hosts
```

on the `hassio_dns` container?

The fairly wild variation in query response times in the logs suggest the result *probably* isn't being generated locally.

From within that container, if you do

```
nslookup windows-11.brunt.local 192.168.1.103
```

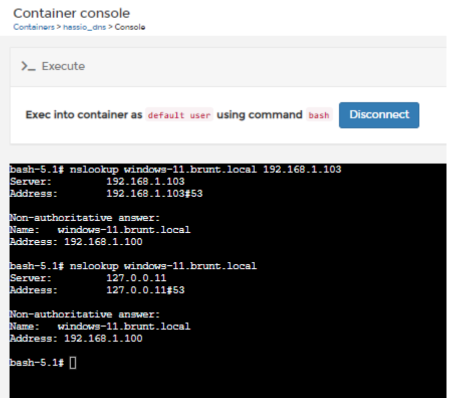
does it work?

I think we're probably getting into packet capture territory here though - need to try and identify whether it's receiving the SERVFAIL from upstream (based on the response timings, I suspect so) or generating locally.

danielbrunt57 commented on Mar 23

bash-5.1# cat /config/hosts

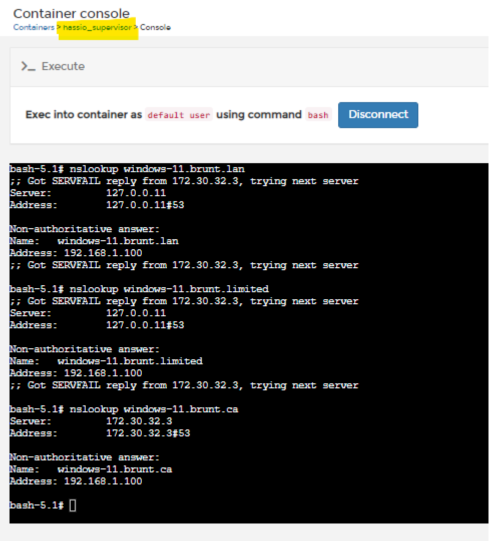
```
127.0.0.1 localhost localhost.local.hass.io
172.30.32.2 hassio hassio.local.hass.io supervisor supervisor.local.hass.io
172.30.32.1 homeassistant homeassistant.local.hass.io home-assistant home-assistant.local.hass.io
172.30.32.3 dns dns.local.hass.io
172.30.32.6 observer observer.local.hass.io
172.30.33.0 core-mosquitto core-mosquitto.local.hass.io
172.30.33.1 a0d7b954-sonweb a0d7b954-sonweb.local.hass.io
172.30.33.2 a0d7b954-zwavejs2mqtt a0d7b954-zwavejs2mqtt.local.hass.io
172.30.33.3 core-mariadb core-mariadb.local.hass.io
172.30.32.1 core-samba core-samba.local.hass.io
172.30.33.6 core-duckdns core-duckdns.local.hass.io
172.30.32.1 a0d7b954-ftp a0d7b954-ftp.local.hass.io
172.30.32.1 a0d7b954-motioneye a0d7b954-motioneye.local.hass.io
172.30.33.10 core-configurator core-configurator.local.hass.io
172.30.33.12 core-nginx-proxy core-nginx-proxy.local.hass.io
172.30.32.1 a0d7b954-esphome a0d7b954-esphome.local.hass.io
172.30.33.13 a0d7b954-wireguard a0d7b954-wireguard.local.hass.io
172.30.33.14 45df7312-zigbee2mqtt 45df7312-zigbee2mqtt.local.hass.io
172.30.33.15 a0d7b954-sqlite-web a0d7b954-sqlite-web.local.hass.io
172.30.33.16 a0d7b954-phpmyadmin a0d7b954-phpmyadmin.local.hass.io
172.30.33.17 f4f71350-ewelink-smart-home-slug f4f71350-ewelink-smart-home-slug.local.hass.io
172.30.33.7 a0d7b954-influxdb a0d7b954-influxdb.local.hass.io
172.30.33.18 core-check-config core-check-config.local.hass.io
172.30.33.9 a0d7b954-unifi a0d7b954-unifi.local.hass.io
172.30.33.18 68e874ae-coredns-fix 68e874ae-coredns-fix.local.hass.io
172.30.32.1 a0d7b954-ssh a0d7b954-ssh.local.hass.io
172.30.33.8 cebe7a76-hassio-google-drive-backup cebe7a76-hassio-google-drive-backup.local.hass.io
172.30.32.1 a0d7b954-traccar a0d7b954-traccar.local.hass.io
172.30.33.4 a0d7b954-nut a0d7b954-nut.local.hass.io
172.30.32.1 a0d7b954-nodered a0d7b954-nodered.local.hass.io
172.30.33.11 e4641267-portainer e4641267-portainer.local.hass.io
```


Container console
Containers > hassio_dns > Console
>_ Execute
Exec into container as default user using command bash
Disconnect

```
bash-5.1# nslookup windows-11.brunt.local 192.168.1.103
Server:         192.168.1.103
Address:        192.168.1.103#53

Non-authoritative answer:
Name:   windows-11.brunt.local
Address: 192.168.1.100

bash-5.1# nslookup windows-11.brunt.local
Server:         127.0.0.11
Address:        127.0.0.11#53

Non-authoritative answer:
Name:   windows-11.brunt.local
Address: 192.168.1.100

bash-5.1# ▯
```

danielbrunt57 commented on Mar 23

Inside hassio_dns, .l* works but not from external requests.
Here is supervisor which fails via hassio_dns but succeeds via docker's DNS:


Container console
Containers > hassio_supervisor > Console
>_ Execute
Exec into container as default user using command bash
Disconnect

```
bash-5.1# nslookup windows-11.brunt.lan
;; Got SERVFAIL reply from 172.30.32.3, trying next server
Server:         127.0.0.11
Address:        127.0.0.11#53

Non-authoritative answer:
Name:   windows-11.brunt.lan
Address: 192.168.1.100
;; Got SERVFAIL reply from 172.30.32.3, trying next server

bash-5.1# nslookup windows-11.brunt.limited
;; Got SERVFAIL reply from 172.30.32.3, trying next server
Server:         127.0.0.11
Address:        127.0.0.11#53

Non-authoritative answer:
Name:   windows-11.brunt.limited
Address: 192.168.1.100
;; Got SERVFAIL reply from 172.30.32.3, trying next server

bash-5.1# nslookup windows-11.brunt.ca
Server:         172.30.32.3
Address:        172.30.32.3#53

Non-authoritative answer:
Name:   windows-11.brunt.ca
Address: 192.168.1.100

bash-5.1# ▯
```

bash-5.1# cat /etc/resolv.conf

```
search local.hass.io

nameserver 172.30.32.3

nameserver 127.0.0.11

options ndots:0
```

bentasker commented on Mar 23

OK, so, that tells us it's either `coredns` itself, *or* the way it's forming upstream queries.

In the `hassio_dns` container, try

```
apk add tcpdump
tcpdump -i any -s0 -w /config/dns.pcap -v port 53
```

Then in another terminal run your query (lets do it from the home assistant container). You should see the counter increase in tcpdump. Ctrl-c it

If you can copy the capture down and attach it, it's easiest, but failing that

```
tcpdump -r /config/dns.pcap -v -x
```

Will dump a load of info

danielbrunt57 commented on Mar 23

Here's the pcap after nslookup windows-11.brunt.limited; nslookup windows-11.brunt.ca from homeassistant:

```
bash-5.1# tcpdump -i any -s0 -w /config/dns.pcap -v port 53
tcpdump: data link type LINUX_SLL2
tcpdump: listening on any, link-type LINUX_SLL2 (Linux cooked v2), snapshot length 262144 bytes
^C36 packets captured
36 packets received by filter
0 packets dropped by kernel
bash-5.1# cat /config/dns.pcap
òl;b}[BGQEGa@@  53hversionhome-assistantiol;b`}[BGQEGb@@  53kNversionhome-assistantiol;b�fB ERZ@@ gb5>hversionhome-assistantiol;b�fB ER@@ g5>kNversionhome-assistantiol;bBGQE<?5g
5hversionhome-assistantio

CDZ

h

h
,b<@b:
.e
       home-assistantio6
ي"-j    XOc�,~Dh}Yi_?)DA77vF˳o
                             {rO)l;B E¡@@e  5hversionhome-assistantioversionhome-assistantioCDZversionhome-assistantiohversionhome-assistantiohl;b+BGQE<?g 5kNversionhome-
assistantio

,&G CDZ

,&G h

,&G h
,b<ˁb:
.,e
      home-assistantio
                      |zZ8"l::|ẩ v89 `gd%y"|d-*33d(?豫)l;B E框@@@  5ȝ$kNversionhome-assistantioversionhome-assistantio,&G CDZversionhome-assistantio,&G hversionhome-assistantio,&G
hl;bI

]BGQEI@@j  55)o

dataservice

accuweathercoml;b

]BGQEI@@i  55-

dataservice
   accuweathercoml;b
                    hB ET[@@
                             gb5@-
                                  dataservice
                                             accuweathercoml;b
                                                              hB ET@@ g5@)o
                                                                           dataservice
                                                                                      accuweathercoml;BGQE="?g 5�{)o
                                                                                                                    dataservice
                                                                                                                               accuweathercom
                                                                                                                                            accuweather-prodapigeenet9,


accuweatherdnJb6)l;bm


B E@@  56)o

dataservice

accuweathercom

dataservice

accuweathercomaccuweather-prodapigeenetaccuweather-prodapigeenet
                                               ccuweatherdnapigeenet
                                                                    accuweatherdnapigeenet6l;bBGQE=$?g 5bm-
                                                                                                            dataservice
                                                                                                                       accuweathercom
                                                                                                                                    accuweather-prodapigeenet9,
                                                                                                                                                               accuweatherdnJnHns-
1671   awsdns-16coukawsdns-hostmasteramazon$ uQ)l;bGB E3@@̃  5q-

dataservice
 accuweathercom
               dataservice
                          accuweathercom,accuweather-prodapigeenetaccuweather-prodapigeenet,
                                                                                             ccuweatherdnapigeenetdnapigeenet,Kns-1671   awsdns-16coukawsdns-hostmasteramazoncom
uQl;bI

ZBGQEF(@=  %52
windows-11bruntlimitedl;b
                         K
                          eB EQ\@@ gb5=�
windows-11bruntlimitedl;b
                         eBGQEQ?cg 5b=
windows-11bruntlimited)l;b=
                           ZB EF@@M  5%2
windows-11bruntlimitedl;b9XBGQEDM@@  Ӂ50ــapiopenweathermaporgl;b~XBGQEDN@@  Ӂ50apiopenweathermaporgl;bcB EO]@@ gb5;}apiopenweathermaporgl;bcB EO@@ g5;}
ــapiopenweathermaporgl;b%BGQE�>O?g 5kــapiopenweathermaporg



)l;bB E'@@ά  5ӁــapiopenweathermaporgapiopenweathermaporgXapiopenweathermaporgXapiopenweathermaporgXl;b
BGQE>P?g 5apiopenweathermaporgKns-1790  awsdns-31coukawsdns-hostmasteramazoncom uQ)l;bB E(@@ʹ 5ӁapiopenweathermaporgopenweathermaporgXKns-1790  awsdns-31coukawsdns-
hostmasteramazoncom uQl;b2UBGQEAI@!  &5-�o
windows-11bruntcal;br`B EL^@@ gb58zo
windows-11bruntcal;bpBGQE\B?^g 5bHIo
windows-11bruntca
                 <d)l;bxB Ed7@@  5&Po
windows-11bruntca
windows-11bruntca<dl;bUBGQEAJ@   55-�
windows-11bruntcal;b`B EL_@@ gb58z
windows-11bruntcal;BGQE|B?=g 5bhغ
```

```
windows-11bruntca$pdnsdanielx*0 :)l;bB E8@@  55uǼ
windows-11bruntcabruntcaX4pdnsbruntcadanielbruntcax*0   :bash-5.1#
```

bentasker commented on Mar 23

Unfortunately can't read it from a `cat` as it's a binary file - you'd need to `scp` it down.

Or do

```
tcpdump -r /config/dns.pcap -v -x
```

To get a text dump.

danielbrunt57 commented on Mar 23

```
bash-5.1# tcpdump -r /config/dns.pcap -v -x
reading from file /config/dns.pcap, link-type LINUX_SLL2 (Linux cooked v2), snapshot length 262144
Warning: interface names might be incorrect
11:51:00.032028 eth0  In  IP (tos 0x0, ttl 64, id 62049, offset 0, flags [DF], proto UDP (17), length 71)
    172.30.32.1.41111 > hassio-dns.53: 26845+ A? version.home-assistant.io. (43)
        0x0000:  4500 0047 f261 4000 4011 b003 ac1e 2001
        0x0010:  ac1e 2003 a097 0035 0033 9885 68dd 0100
        0x0020:  0001 0000 0000 0000 0776 6572 7369 6f6e
        0x0030:  0e68 6f6d 652d 6173 7369 7374 616e 7402
        0x0040:  696f 0000 0100 01
11:51:00.032096 eth0  In  IP (tos 0x0, ttl 64, id 62050, offset 0, flags [DF], proto UDP (17), length 71)
    172.30.32.1.41111 > hassio-dns.53: 27470+ AAAA? version.home-assistant.io. (43)
        0x0000:  4500 0047 f262 4000 4011 b002 ac1e 2001
        0x0010:  ac1e 2003 a097 0035 0033 9885 6b4e 0100
        0x0020:  0001 0000 0000 0000 0776 6572 7369 6f6e
        0x0030:  0e68 6f6d 652d 6173 7369 7374 616e 7402
        0x0040:  696f 0000 1c00 01
11:51:00.032669 eth0  Out IP (tos 0x0, ttl 64, id 58202, offset 0, flags [DF], proto UDP (17), length 82)
    hassio-dns.49506 > pdns.brunt.ca.53: 26845+ [1au] A? version.home-assistant.io. (54)
        0x0000:  4500 0052 e35a 4000 4011 c90f ac1e 2003
        0x0010:  c0a8 0167 c162 0035 003e 8e80 68dd 0100
        0x0020:  0001 0000 0000 0001 0776 6572 7369 6f6e
        0x0030:  0e68 6f6d 652d 6173 7369 7374 616e 7402
        0x0040:  696f 0000 0100 0100 0029 0800 0000 8000
        0x0050:  0000
11:51:00.032730 eth0  Out IP (tos 0x0, ttl 64, id 60591, offset 0, flags [DF], proto UDP (17), length 82)
    hassio-dns.35218 > pdns.brunt.ca.53: 27470+ [1au] AAAA? version.home-assistant.io. (54)
        0x0000:  4500 0052 ecaf 4000 4011 bfba ac1e 2003
        0x0010:  c0a8 0167 8992 0035 003e 8e80 6b4e 0100
        0x0020:  0001 0000 0000 0001 0776 6572 7369 6f6e
        0x0030:  0e68 6f6d 652d 6173 7369 7374 616e 7402
        0x0040:  696f 0000 1c00 0100 0029 0800 0000 8000
        0x0050:  0000
11:51:00.034429 eth0  In  IP (tos 0x0, ttl 63, id 15508, offset 0, flags [none], proto UDP (17), length 243)
    pdns.brunt.ca.53 > hassio-dns.49506: 26845$ 4/0/1 version.home-assistant.io. A 172.67.68.90, version.home-assistant.io. A 104.26.4.238, version.home-assistant.io. A
104.26.5.238, version.home-assistant.io. RRSIG (215)
        0x0000:  4500 00f3 3c94 0000 3f11 b035 c0a8 0167
        0x0010:  ac1e 2003 0035 c162 00df 081f 68dd 81a0
        0x0020:  0001 0004 0000 0001 0776 6572 7369 6f6e
        0x0030:  0e68 6f6d 652d 6173 7369 7374 616e 7402
        0x0040:  696f 0000 0100 01c0 0c00 0100 0100 0000
        0x0050:  c800 04ac 4344 5ac0 0c00 0100 0100 0000
        0x0060:  c800 0468 1a04 eec0 0c00 0100 0100 0000
        0x0070:  c800 0468 1a05 eec0 0c00 2e00 0100 0000
        0x0080:  c800 6500 010d 0300 0001 2c62 3ccb 4062
        0x0090:  3a0c 2086 c90e 686f 6d65 2d61 7373 6973
        0x00a0:  7461 6e74 0269 6f00 360a dda4 8f22 af2d
        0x00b0:  6a09 d6e4 584f 63f8 0280 7f2c ad7e 87aa
        0x00c0:  4441 0868 7d12 cf92 695f a63f 2902 e544
        0x00d0:  be41 37a4 f29b c4f3 3776 46c3 f5e4 cbb3
        0x00e0:  6f84 8c0c 107b 724f 0000 2902 0000 0080
        0x00f0:  0000 00
11:51:00.034956 eth0  Out IP (tos 0x0, ttl 64, id 41349, offset 0, flags [DF], proto UDP (17), length 194)
    hassio-dns.53 > 172.30.32.1.41111: 26845$ 3/0/0 version.home-assistant.io. A 172.67.68.90, version.home-assistant.io. A 104.26.4.238, version.home-assistant.io. A 104.26.5.238
(166)
        0x0000:  4500 00c2 a185 4000 4011 0065 ac1e 2003
        0x0010:  ac1e 2001 0035 a097 00ae 9900 68dd 81a0
        0x0020:  0001 0003 0000 0000 0776 6572 7369 6f6e
        0x0030:  0e68 6f6d 652d 6173 7369 7374 616e 7402
        0x0040:  696f 0000 0100 0107 7665 7273 696f 6e0e
        0x0050:  686f 6d65 2d61 7373 6973 7461 6e74 0269
        0x0060:  6f00 0001 0001 0000 00c8 0004 ac43 445a
        0x0070:  0776 6572 7369 6f6e 0e68 6f6d 652d 6173
        0x0080:  7369 7374 616e 7402 696f 0000 0100 0100
        0x0090:  0000 c800 0468 1a04 ee07 7665 7273 696f
        0x00a0:  6e0e 686f 6d65 2d61 7373 6973 7461 6e74
        0x00b0:  0269 6f00 0001 0001 0000 00c8 0004 681a
        0x00c0:  05ee
11:51:00.041117 eth0  In  IP (tos 0x0, ttl 63, id 15510, offset 0, flags [none], proto UDP (17), length 279)
    pdns.brunt.ca.53 > hassio-dns.35218: 27470$ 4/0/1 version.home-assistant.io. AAAA 2606:4700:20::ac43:445a, version.home-assistant.io. AAAA 2606:4700:20::681a:4ee, version.home-
assistant.io. AAAA 2606:4700:20::681a:5ee, version.home-assistant.io. RRSIG (251)
        0x0000:  4500 0117 3c96 0000 3f11 b00f c0a8 0167
        0x0010:  ac1e 2003 0035 8992 0103 fb18 6b4e 81a0
        0x0020:  0001 0004 0000 0001 0776 6572 7369 6f6e
        0x0030:  0e68 6f6d 652d 6173 7369 7374 616e 7402
        0x0040:  696f 0000 1c00 01c0 0c00 1c00 0100 0001
        0x0050:  2c00 1026 0647 0000 2000 0000 0000 00ac
        0x0060:  4344 5ac0 0c00 1c00 0100 0001 2c00 1026
        0x0070:  0647 0000 2000 0000 0000 0068 1a04 eec0
        0x0080:  0c00 1c00 0100 0001 2c00 1026 0647 0000
        0x0090:  2000 0000 0000 0068 1a05 eec0 0c00 2e00
        0x00a0:  0100 0001 2c00 6500 1c0d 0300 0001 2c62
        0x00b0:  3ccb a462 3a0c 8486 c90e 686f 6d65 2d61
        0x00c0:  7373 6973 7461 6e74 0269 6f00 8f0b 7c7a
        0x00d0:  5ae0 38cf 226c 3ada 3aff f47c acc7 bb20
        0x00e0:  76e7 1a38 f5dd 8c9a 3960 6764 aca3 25ee
        0x00f0:  c179 8cff 227c 642d e32a 9de7 b6c2 3333
        0x0100:  ce64 0703 f228 e096 3fe8 b1ab 0000 2902
        0x0110:  0000 0080 0000 00
11:51:00.041470 eth0  Out IP (tos 0x0, ttl 64, id 41350, offset 0, flags [DF], proto UDP (17), length 230)
    hassio-dns.53 > 172.30.32.1.41111: 27470$ 3/0/0 version.home-assistant.io. AAAA 2606:4700:20::ac43:445a, version.home-assistant.io. AAAA 2606:4700:20::681a:4ee, version.home-
assistant.io. AAAA 2606:4700:20::681a:5ee (202)
        0x0000:  4500 00e6 a186 4000 4011 0040 ac1e 2003
        0x0010:  ac1e 2001 0035 a097 00d2 9924 6b4e 81a0
        0x0020:  0001 0003 0000 0000 0776 6572 7369 6f6e
        0x0030:  0e68 6f6d 652d 6173 7369 7374 616e 7402
        0x0040:  696f 0000 1c00 0107 7665 7273 696f 6e0e
        0x0050:  686f 6d65 2d61 7373 6973 7461 6e74 0269
        0x0060:  6f00 001c 0001 0000 012c 0010 2606 4700
        0x0070:  0020 0000 0000 0000 ac43 445a 0776 6572
        0x0080:  7369 6f6e 0e68 6f6d 652d 6173 7369 7374
        0x0090:  616e 7402 696f 0000 1c00 0100 0001 2c00
        0x00a0:  1026 0647 0000 2000 0000 0000 0068 1a04
        0x00b0:  ee07 7665 7273 696f 6e0e 686f 6d65 2d61
        0x00c0:  7373 6973 7461 6e74 0269 6f00 001c 0001
        0x00d0:  0000 012c 0010 2606 4700 0020 0000 0000
        0x00e0:  0000 681a 05ee
11:51:01.002889 eth0  In  IP (tos 0x0, ttl 64, id 62200, offset 0, flags [DF], proto UDP (17), length 73)
    172.30.32.1.38172 > hassio-dns.53: 10607+ A? dataservice.accuweather.com. (45)
        0x0000:  4500 0049 f2f8 4000 4011 af6a ac1e 2001
        0x0010:  ac1e 2003 951c 0035 0035 9887 296f 0100
        0x0020:  0001 0000 0000 0000 0b64 6174 6173 6572
        0x0030:  7669 6365 0b61 6363 7577 6561 7468 6572
        0x0040:  0363 6f6d 0000 0100 01
11:51:01.002979 eth0  In  IP (tos 0x0, ttl 64, id 62201, offset 0, flags [DF], proto UDP (17), length 73)
    172.30.32.1.38172 > hassio-dns.53: 11774+ AAAA? dataservice.accuweather.com. (45)
        0x0000:  4500 0049 f2f9 4000 4011 af69 ac1e 2001
        0x0010:  ac1e 2003 951c 0035 0035 9887 2dfe 0100
        0x0020:  0001 0000 0000 0000 0b64 6174 6173 6572
        0x0030:  7669 6365 0b61 6363 7577 6561 7468 6572
        0x0040:  0363 6f6d 0000 1c00 01
11:51:01.003263 eth0  Out IP (tos 0x0, ttl 64, id 58203, offset 0, flags [DF], proto UDP (17), length 84)
    hassio-dns.49506 > pdns.brunt.ca.53: 11774+ [1au] AAAA? dataservice.accuweather.com. (56)
        0x0000:  4500 0054 e35b 4000 4011 c90c ac1e 2003
        0x0010:  c0a8 0167 c162 0035 0040 8e82 2dfe 0100
        0x0020:  0001 0000 0000 0001 0b64 6174 6173 6572
        0x0030:  7669 6365 0b61 6363 7577 6561 7468 6572
        0x0040:  0363 6f6d 0000 1c00 0100 0029 0800 0000
        0x0050:  8000 0000
11:51:01.003290 eth0  Out IP (tos 0x0, ttl 64, id 60592, offset 0, flags [DF], proto UDP (17), length 84)
    hassio-dns.35218 > pdns.brunt.ca.53: 10607+ [1au] A? dataservice.accuweather.com. (56)
        0x0000:  4500 0054 ecb0 4000 4011 bfb7 ac1e 2003
        0x0010:  c0a8 0167 8992 0035 0040 8e82 296f 0100
        0x0020:  0001 0000 0000 0001 0b64 6174 6173 6572
        0x0030:  7669 6365 0b61 6363 7577 6561 7468 6572
        0x0040:  0363 6f6d 0000 0100 0100 0029 0800 0000
        0x0050:  8000 0000
11:51:01.180979 eth0  In  IP (tos 0x0, ttl 63, id 15650, offset 0, flags [none], proto UDP (17), length 170)
    pdns.brunt.ca.53 > hassio-dns.35218: 10607 3/0/1 dataservice.accuweather.com. CNAME accuweather-prod.apigee.net., accuweather-prod.apigee.net. CNAME accuweather.dn.apigee.net.,
accuweather.dn.apigee.net. A 54.189.189.19 (142)
        0x0000:  4500 00aa 3d22 0000 3f11 aff0 c0a8 0167
        0x0010:  ac1e 2003 0035 8992 0096 7f7b 296f 8180
        0x0020:  0001 0003 0000 0001 0b64 6174 6173 6572
        0x0030:  7669 6365 0b61 6363 7577 6561 7468 6572
        0x0040:  0363 6f6d 0000 0100 01c0 0c00 0500 0100
        0x0050:  0007 0800 1d10 6163 6375 7765 6174 6865
        0x0060:  722d 7072 6f64 0661 7069 6765 6503 6e65
        0x0070:  7400 c039 0005 0001 0000 012c 0011 0b61
        0x0080:  6363 7577 6561 7468 6572 0264 6ec0 4ac0
        0x0090:  6200 0100 0100 0000 0500 0436 bdbd 1300
        0x00a0:  0029 0200 0000 8000 0000
11:51:01.181357 eth0  Out IP (tos 0x0, ttl 64, id 41366, offset 0, flags [DF], proto UDP (17), length 248)
    hassio-dns.53 > 172.30.32.1.38172: 10607 3/0/0 dataservice.accuweather.com. CNAME accuweather-prod.apigee.net., accuweather-prod.apigee.net. CNAME accuweather.dn.apigee.net.,
accuweather.dn.apigee.net. A 54.189.189.19 (220)
        0x0000:  4500 00f8 a196 4000 4011 001e ac1e 2003
        0x0010:  ac1e 2001 0035 951c 00e4 9936 296f 8180
        0x0020:  0001 0003 0000 0000 0b64 6174 6173 6572
        0x0030:  7669 6365 0b61 6363 7577 6561 7468 6572
        0x0040:  0363 6f6d 0000 0100 010b 6461 7461 7365
        0x0050:  7276 6963 650b 6163 6375 7765 6174 6865
        0x0060:  7203 636f 6d00 0005 0001 0000 0005 001d
        0x0070:  1061 6363 7577 6561 7468 6572 2d70 726f
        0x0080:  6406 6170 6967 6565 036e 6574 0010 6163
        0x0090:  6375 7765 6174 6865 722d 7072 6f64 0661
        0x00a0:  7069 6765 6503 6e65 7400 0005 0001 0000
        0x00b0:  0005 001b 0b61 6363 7577 6561 7468 6572
        0x00c0:  0264 6e06 6170 6967 6565 036e 6574 000b
        0x00d0:  6163 6375 7765 6174 6865 7202 646e 0661
        0x00e0:  7069 6765 6503 6e65 7400 0001 0001 0000
        0x00f0:  0005 0004 36bd bd13
11:51:01.202580 eth0  In  IP (tos 0x0, ttl 63, id 15652, offset 0, flags [none], proto UDP (17), length 238)
    pdns.brunt.ca.53 > hassio-dns.49506: 11774 2/1/1 dataservice.accuweather.com. CNAME accuweather-prod.apigee.net., accuweather-prod.apigee.net. CNAME accuweather.dn.apigee.net.
(210)
        0x0000:  4500 00ee 3d24 0000 3f11 afaa c0a8 0167
        0x0010:  ac1e 2003 0035 c162 00da 6dbb 2dfe 8180
        0x0020:  0001 0002 0001 0001 0b64 6174 6173 6572
        0x0030:  7669 6365 0b61 6363 7577 6561 7468 6572
        0x0040:  0363 6f6d 0000 1c00 01c0 0c00 0500 0100
        0x0050:  0007 0800 1d10 6163 6375 7765 6174 6865
        0x0060:  722d 7072 6f64 0661 7069 6765 6503 6e65
        0x0070:  7400 c039 0005 0001 0000 012c 0011 0b61
        0x0080:  6363 7577 6561 7468 6572 0264 6ec0 4ac0
        0x0090:  6e00 0600 0100 0003 8400 4807 6e73 2d31
        0x00a0:  3637 3109 6177 7364 6e73 2d31 3602 636f
        0x00b0:  0275 6b00 1161 7773 646e 732d 686f 7374
        0x00c0:  6d61 7374 6572 0661 6d61 7a6f 6ec0 2400
        0x00d0:  0000 0100 001c 2000 0003 8400 1275 0000
        0x00e0:  0151 8000 0029 0200 0000 8000 0000
11:51:01.202881 eth0  Out IP (tos 0x0, ttl 64, id 41369, offset 0, flags [DF], proto UDP (17), length 307)
    hassio-dns.53 > 172.30.32.1.38172: 11774 2/1/0 dataservice.accuweather.com. CNAME accuweather-prod.apigee.net., accuweather-prod.apigee.net. CNAME accuweather.dn.apigee.net.
(279)
        0x0000:  4500 0133 a199 4000 4011 ffdf ac1e 2003
        0x0010:  ac1e 2001 0035 951c 011f 9971 2dfe 8180
        0x0020:  0001 0002 0001 0000 0b64 6174 6173 6572
        0x0030:  7669 6365 0b61 6363 7577 6561 7468 6572
        0x0040:  0363 6f6d 0000 1c00 010b 6461 7461 7365
        0x0050:  7276 6963 650b 6163 6375 7765 6174 6865
        0x0060:  7203 636f 6d00 0005 0001 0000 012c 001d
        0x0070:  1061 6363 7577 6561 7468 6572 2d70 726f
        0x0080:  6406 6170 6967 6565 036e 6574 0010 6163
        0x0090:  6375 7765 6174 6865 722d 7072 6f64 0661
        0x00a0:  7069 6765 6503 6e65 7400 0005 0001 0000
        0x00b0:  012c 001b 0b61 6363 7577 6561 7468 6572
        0x00c0:  0264 6e06 6170 6967 6565 036e 6574 0002
        0x00d0:  646e 0661 7069 6765 6503 6e65 7400 0006
        0x00e0:  0001 0000 012c 004b 076e 732d 3136 3731
        0x00f0:  0961 7773 646e 732d 3136 0263 6f02 756b
        0x0100:  0011 6177 7364 6e73 2d68 6f73 746d 6173
        0x0110:  7465 7206 616d 617a 6f6e 0363 6f6d 0000
        0x0120:  0000 0100 001c 2000 0003 8400 1275 0000
        0x0130:  0151 80
11:51:02.739751 eth0  In  IP (tos 0x0, ttl 64, id 62248, offset 0, flags [none], proto UDP (17), length 70)
    172.30.32.1.53541 > hassio-dns.53: 53953+ A? windows-11.brunt.limited. (42)
        0x0000:  4500 0046 f328 0000 4011 ef3d ac1e 2001
        0x0010:  ac1e 2003 d125 0035 0032 9884 d2c1 0100
        0x0020:  0001 0000 0000 0000 0a77 696e 646f 7773
        0x0030:  2d31 3105 6272 756e 7407 6c69 6d69 7465
        0x0040:  6400 0001 0001
11:51:02.740107 eth0  Out IP (tos 0x0, ttl 64, id 58204, offset 0, flags [DF], proto UDP (17), length 81)
    hassio-dns.49506 > pdns.brunt.ca.53: 53953+ [1au] A? windows-11.brunt.limited. (53)
        0x0000:  4500 0051 e35c 4000 4011 c90e ac1e 2003
        0x0010:  c0a8 0167 c162 0035 003d 8e7f d2c1 0100
        0x0020:  0001 0000 0000 0001 0a77 696e 646f 7773
        0x0030:  2d31 3105 6272 756e 7407 6c69 6d69 7465
        0x0040:  6400 0001 0001 0000 2908 0000 0080 0000
        0x0050:  00
11:51:02.765061 eth0  In  IP (tos 0x0, ttl 63, id 15880, offset 0, flags [none], proto UDP (17), length 81)
    pdns.brunt.ca.53 > hassio-dns.49506: 53953 ServFail 0/0/1 (53)
        0x0000:  4500 0051 3e08 0000 3f11 af63 c0a8 0167
        0x0010:  ac1e 2003 0035 c162 003d 85c1 d2c1 8182
        0x0020:  0001 0000 0000 0001 0a77 696e 646f 7773
        0x0030:  2d31 3105 6272 756e 7407 6c69 6d69 7465
        0x0040:  6400 0001 0001 0000 2902 0000 0080 0000
        0x0050:  00
11:51:02.765501 eth0  Out IP (tos 0x0, ttl 64, id 41752, offset 0, flags [DF], proto UDP (17), length 70)
    hassio-dns.53 > 172.30.32.1.53541: 53953 ServFail 0/0/0 (42)
        0x0000:  4500 0046 a318 4000 4011 ff4d ac1e 2003
        0x0010:  ac1e 2001 0035 d125 0032 9884 d2c1 8182
        0x0020:  0001 0000 0000 0000 0a77 696e 646f 7773
        0x0030:  2d31 3105 6272 756e 7407 6c69 6d69 7465
        0x0040:  6400 0001 0001
11:51:03.006201 eth0  In  IP (tos 0x0, ttl 64, id 62285, offset 0, flags [DF], proto UDP (17), length 68)
    172.30.32.1.53676 > hassio-dns.53: 56487+ A? api.openweathermap.org. (40)
        0x0000:  4500 0044 f34d 4000 4011 af1a ac1e 2001
        0x0010:  ac1e 2003 d1ac 0035 0030 9882 dca7 0100
        0x0020:  0001 0000 0000 0000 0361 7069 0e6f 7065
        0x0030:  6e77 6561 7468 6572 6d61 7003 6f72 6700
        0x0040:  0001 0001
11:51:03.006270 eth0  In  IP (tos 0x0, ttl 64, id 62286, offset 0, flags [DF], proto UDP (17), length 68)
    172.30.32.1.53676 > hassio-dns.53: 57362+ AAAA? api.openweathermap.org. (40)
        0x0000:  4500 0044 f34e 4000 4011 af19 ac1e 2001
        0x0010:  ac1e 2003 d1ac 0035 0030 9882 e012 0100
        0x0020:  0001 0000 0000 0000 0361 7069 0e6f 7065
        0x0030:  6e77 6561 7468 6572 6d61 7003 6f72 6700
        0x0040:  001c 0001
11:51:03.006596 eth0  Out IP (tos 0x0, ttl 64, id 58205, offset 0, flags [DF], proto UDP (17), length 79)
    hassio-dns.49506 > pdns.brunt.ca.53: 57362+ [1au] AAAA? api.openweathermap.org. (51)
        0x0000:  4500 004f e35d 4000 4011 c90f ac1e 2003
        0x0010:  c0a8 0167 c162 0035 003b 8e7d e012 0100
        0x0020:  0001 0000 0000 0001 0361 7069 0e6f 7065
        0x0030:  6e77 6561 7468 6572 6d61 7003 6f72 6700
        0x0040:  001c 0001 0000 2908 0000 0080 0000 00
11:51:03.006644 eth0  Out IP (tos 0x0, ttl 64, id 60593, offset 0, flags [DF], proto UDP (17), length 79)
    hassio-dns.35218 > pdns.brunt.ca.53: 56487+ [1au] A? api.openweathermap.org. (51)
        0x0000:  4500 004f ecb1 4000 4011 bfbb ac1e 2003
        0x0010:  c0a8 0167 8992 0035 003b 8e7d dca7 0100
        0x0020:  0001 0000 0000 0001 0361 7069 0e6f 7065
        0x0030:  6e77 6561 7468 6572 6d61 7003 6f72 6700
        0x0040:  0001 0001 0000 2908 0000 0080 0000 00
11:51:03.074533 eth0  In  IP (tos 0x0, ttl 63, id 15951, offset 0, flags [none], proto UDP (17), length 127)
    pdns.brunt.ca.53 > hassio-dns.35218: 56487 3/0/1 api.openweathermap.org. A 192.241.245.161, api.openweathermap.org. A 192.241.187.136, api.openweathermap.org. A 192.241.167.16
(99)
        0x0000:  4500 007f 3e4f 0000 3f11 aeee c0a8 0167
        0x0010:  ac1e 2003 0035 8992 006b 05cb dca7 8180
        0x0020:  0001 0003 0000 0001 0361 7069 0e6f 7065
        0x0030:  6e77 6561 7468 6572 6d61 7003 6f72 6700
        0x0040:  0001 0001 c00c 0001 0001 0000 0e10 0004
        0x0050:  c0f1 f5a1 c00c 0001 0001 0000 0e10 0004
        0x0060:  c0f1 bb88 c00c 0001 0001 0000 0e10 0004
        0x0070:  c0f1 a710 0000 2902 0000 0080 0000 00
11:51:03.074990 eth0  Out IP (tos 0x0, ttl 64, id 41767, offset 0, flags [DF], proto UDP (17), length 182)
    hassio-dns.53 > 172.30.32.1.53676: 56487 3/0/0 api.openweathermap.org. A 192.241.245.161, api.openweathermap.org. A 192.241.187.136, api.openweathermap.org. A 192.241.167.16
(154)
        0x0000:  4500 00b6 a327 4000 4011 fece ac1e 2003
        0x0010:  ac1e 2001 0035 d1ac 00a2 98f4 dca7 8180
        0x0020:  0001 0003 0000 0000 0361 7069 0e6f 7065
        0x0030:  6e77 6561 7468 6572 6d61 7003 6f72 6700
        0x0040:  0001 0001 0361 7069 0e6f 7065 6e77 6561
        0x0050:  7468 6572 6d61 7003 6f72 6700 0001 0001
        0x0060:  0000 0258 0004 c0f1 f5a1 0361 7069 0e6f
        0x0070:  7065 6e77 6561 7468 6572 6d61 7003 6f72
        0x0080:  6700 0001 0001 0000 0258 0004 c0f1 bb88
        0x0090:  0361 7069 0e6f 7065 6e77 6561 7468 6572
        0x00a0:  6d61 7003 6f72 6700 0001 0001 0000 0258
        0x00b0:  0004 c0f1 a710
11:51:03.075274 eth0  In  IP (tos 0x0, ttl 63, id 15952, offset 0, flags [none], proto UDP (17), length 166)
    pdns.brunt.ca.53 > hassio-dns.49506: 57362 0/1/1 (138)
        0x0000:  4500 00a6 3e50 0000 3f11 aec6 c0a8 0167
        0x0010:  ac1e 2003 0035 c162 0092 e108 e012 8180
        0x0020:  0001 0000 0001 0001 0361 7069 0e6f 7065
        0x0030:  6e77 6561 7468 6572 6d61 7003 6f72 6700
        0x0040:  001c 0001 c010 0006 0001 0000 0384 004b
        0x0050:  076e 732d 3137 3930 0961 7773 646e 732d
        0x0060:  3331 0263 6f02 756b 0011 6177 7364 6e73
        0x0070:  2d68 6f73 746d 6173 7465 7206 616d 617a
        0x0080:  6f6e 0363 6f6d 0000 0000 0100 001c 2000
        0x0090:  0003 8400 1275 0000 0151 8000 0029 0200
        0x00a0:  0000 8000 0000
11:51:03.075507 eth0  Out IP (tos 0x0, ttl 64, id 41768, offset 0, flags [DF], proto UDP (17), length 173)
    hassio-dns.53 > 172.30.32.1.53676: 57362 0/1/0 (145)
        0x0000:  4500 00ad a328 4000 4011 fed6 ac1e 2003
        0x0010:  ac1e 2001 0035 d1ac 0099 98eb e012 8180
        0x0020:  0001 0000 0001 0000 0361 7069 0e6f 7065
        0x0030:  6e77 6561 7468 6572 6d61 7003 6f72 6700
        0x0040:  001c 0001 0e6f 7065 6e77 6561 7468 6572
        0x0050:  6d61 7003 6f72 6700 0006 0001 0000 0258
        0x0060:  004b 076e 732d 3137 3930 0961 7773 646e
        0x0070:  732d 3331 0263 6f02 756b 0011 6177 7364
        0x0080:  6e73 2d68 6f73 746d 6173 7465 7206 616d
        0x0090:  617a 6f6e 0363 6f6d 0000 0000 0100 001c
        0x00a0:  2000 0003 8400 1275 0000 0151 80
11:51:08.295474 eth0  In  IP (tos 0x0, ttl 64, id 63561, offset 0, flags [none], proto UDP (17), length 65)
    172.30.32.1.57638 > hassio-dns.53: 42863+ A? windows-11.brunt.ca. (37)
        0x0000:  4500 0041 f849 0000 4011 ea21 ac1e 2001
        0x0010:  ac1e 2003 e126 0035 002d 987f a76f 0100
        0x0020:  0001 0000 0000 0000 0a77 696e 646f 7773
        0x0030:  2d31 3105 6272 756e 7402 6361 0000 0100
        0x0040:  01
11:51:08.295794 eth0  Out IP (tos 0x0, ttl 64, id 58206, offset 0, flags [DF], proto UDP (17), length 76)
    hassio-dns.49506 > pdns.brunt.ca.53: 42863+ [1au] A? windows-11.brunt.ca. (48)
        0x0000:  4500 004c e35e 4000 4011 c911 ac1e 2003
        0x0010:  c0a8 0167 c162 0035 0038 8e7a a76f 0100
        0x0020:  0001 0000 0000 0001 0a77 696e 646f 7773
        0x0030:  2d31 3105 6272 756e 7402 6361 0000 0100
        0x0040:  0100 0029 0800 0000 8000 0000
11:51:08.320722 eth0  In  IP (tos 0x0, ttl 63, id 16898, offset 0, flags [none], proto UDP (17), length 92)
    pdns.brunt.ca.53 > hassio-dns.49506: 42863 1/0/1 windows-11.brunt.ca. A 192.168.1.100 (64)
```

```
        0x0000:  4500 005c 4202 0000 3f11 ab5e c0a8 0167
        0x0010:  ac1e 2003 0035 c162 0048 49b3 a76f 8180
        0x0020:  0001 0001 0000 0001 0a77 696e 646f 7773
        0x0030:  2d31 3105 6272 756e 7402 6361 0000 0100
        0x0040:  01c0 0c00 0100 0100 0000 3c00 04c0 a801
        0x0050:  6400 0029 0200 0000 8000 0000
11:51:08.320916 eth0  Out IP (tos 0x0, ttl 64, id 42039, offset 0, flags [DF], proto UDP (17), length 100)
    hassio-dns.53 > 172.30.32.1.57638: 42863 1/0/0 windows-11.brunt.ca. A 192.168.1.100 (72)
        0x0000:  4500 0064 a437 4000 4011 fe10 ac1e 2003
        0x0010:  ac1e 2001 0035 e126 0050 98a2 a76f 8180
        0x0020:  0001 0001 0000 0000 0a77 696e 646f 7773
        0x0030:  2d31 3105 6272 756e 7402 6361 0000 0100
        0x0040:  010a 7769 6e64 6f77 732d 3131 0562 7275
        0x0050:  6e74 0263 6100 0001 0001 0000 003c 0004
        0x0060:  c0a8 0164
11:51:08.321433 eth0  In  IP (tos 0x0, ttl 64, id 63562, offset 0, flags [none], proto UDP (17), length 65)
    172.30.32.1.59445 > hassio-dns.53: 48146+ AAAA? windows-11.brunt.ca. (37)
        0x0000:  4500 0041 f84a 0000 4011 ea20 ac1e 2001
        0x0010:  ac1e 2003 e835 0035 002d 987f bc12 0100
        0x0020:  0001 0000 0000 0000 0a77 696e 646f 7773
        0x0030:  2d31 3105 6272 756e 7402 6361 0000 1c00
        0x0040:  01
11:51:08.321675 eth0  Out IP (tos 0x0, ttl 64, id 58207, offset 0, flags [DF], proto UDP (17), length 76)
    hassio-dns.49506 > pdns.brunt.ca.53: 48146+ [1au] AAAA? windows-11.brunt.ca. (48)
        0x0000:  4500 004c e35f 4000 4011 c910 ac1e 2003
        0x0010:  c0a8 0167 c162 0035 0038 8e7a bc12 0100
        0x0020:  0001 0000 0000 0001 0a77 696e 646f 7773
        0x0030:  2d31 3105 6272 756e 7402 6361 0000 1c00
        0x0040:  0100 0029 0800 0000 8000 0000
11:51:08.323473 eth0  In  IP (tos 0x0, ttl 63, id 16899, offset 0, flags [none], proto UDP (17), length 124)
    pdns.brunt.ca.53 > hassio-dns.49506: 48146 0/1/1 (96)
        0x0000:  4500 007c 4203 0000 3f11 ab3d c0a8 0167
        0x0010:  ac1e 2003 0035 c162 0068 e7db bc12 8180
        0x0020:  0001 0000 0001 0001 0a77 696e 646f 7773
        0x0030:  2d31 3105 6272 756e 7402 6361 0000 1c00
        0x0040:  01c0 1700 0600 0100 000e 1000 2404 7064
        0x0050:  6e73 c017 0664 616e 6965 6cc0 1778 85c2
        0x0060:  e700 002a 3000 000e 1000 093a 8000 000e
        0x0070:  1000 0029 0200 0000 8000 0000
11:51:08.323628 eth0  Out IP (tos 0x0, ttl 64, id 42040, offset 0, flags [DF], proto UDP (17), length 137)
    hassio-dns.53 > 172.30.32.1.59445: 48146 0/1/0 (109)
        0x0000:  4500 0089 a438 4000 4011 fdea ac1e 2003
        0x0010:  ac1e 2001 0035 e835 0075 98c7 bc12 8180
        0x0020:  0001 0000 0001 0000 0a77 696e 646f 7773
        0x0030:  2d31 3105 6272 756e 7402 6361 0000 1c00
        0x0040:  0105 6272 756e 7402 6361 0000 0600 0100
        0x0050:  0002 5800 3404 7064 6e73 0562 7275 6e74
        0x0060:  0263 6100 0664 616e 6965 6c05 6272 756e
        0x0070:  7402 6361 0078 85c2 e700 002a 3000 000e
        0x0080:  1000 093a 8000 000e 10
```

bentasker commented on Mar 23

Ah-hah!

```
11:51:02.739751 eth0  In  IP (tos 0x0, ttl 64, id 62248, offset 0, flags [none], proto UDP (17), length 70)
    172.30.32.1.53541 > hassio-dns.53: 53953+ A? windows-11.brunt.limited. (42)
        0x0000:  4500 0046 f328 0000 4011 ef3d ac1e 2001
        0x0010:  ac1e 2003 d125 0035 0032 9884 d2c1 0100
        0x0020:  0001 0000 0000 0000 0a77 696e 646f 7773
        0x0030:  2d31 3105 6272 756e 7407 6c69 6d69 7465
        0x0040:  6400 0001 0001
11:51:02.740107 eth0  Out IP (tos 0x0, ttl 64, id 58204, offset 0, flags [DF], proto UDP (17), length 81)
    hassio-dns.49506 > pdns.brunt.ca.53: 53953+ [1au] A? windows-11.brunt.limited. (53)
        0x0000:  4500 0051 e35c 4000 4011 c90e ac1e 2003
        0x0010:  c0a8 0167 c162 0035 003d 8e7f d2c1 0100
        0x0020:  0001 0000 0000 0001 0a77 696e 646f 7773
        0x0030:  2d31 3105 6272 756e 7407 6c69 6d69 7465
        0x0040:  6400 0001 0001 0000 2908 0000 0080 0000
        0x0050:  00
11:51:02.765061 eth0  In  IP (tos 0x0, ttl 63, id 15880, offset 0, flags [none], proto UDP (17), length 81)
    pdns.brunt.ca.53 > hassio-dns.49506: 53953 ServFail 0/0/1 (53)
        0x0000:  4500 0051 3e08 0000 3f11 af63 c0a8 0167
        0x0010:  ac1e 2003 0035 c162 003d 85c1 d2c1 8182
        0x0020:  0001 0000 0000 0001 0a77 696e 646f 7773
        0x0030:  2d31 3105 6272 756e 7407 6c69 6d69 7465
        0x0040:  6400 0001 0001 0000 2902 0000 0080 0000
```

So, what we can see here is

- Your query for `windows-11.brunt.limited` coming in
- The upstream query going out (but to `pdns.brunt.ca`)
- `pdns.brunt.ca` returning a SERVFAIL

So, we know the `SERVFAIL` is coming from upstream. I assume `pdns.brunt.ca` is a reverse name for `192.168.1.103` ?

Do

```
nslookup 192.168.1.103
```

to confirm that.

Looking at the packets themselves:

If we decode your query we can see flags etc

```
▸ Internet Protocol Version 4
▸ User Datagram Protocol
▾ Domain Name System (query)
    • Transaction ID: 0xd2c1
    ▾ Flags: 0x0100 Standard query
        • 0... .... .... .... = Response: Message is a query
        • .000 0... .... .... = Opcode: Standard query (0)
        • .... ..0. .... .... = Truncated: Message is not truncated
        • .... ...1 .... .... = Recursion desired: Do query recursively
        • .... .... .0.. .... = Z: reserved (0)
        • .... .... ...0 .... = Non-authenticated data: Unacceptable
    • Questions: 1
    • Answer RRs: 0
    • Authority RRs: 0
    • Additional RRs: 0
    ▾ Queries
        ▾ windows-11.brunt.limited: type A, class IN
            • Name: windows-11.brunt.limited
            • Name Length: 24
            • Label Count: 3
            • Type: A (Host Address) (1)
            • Class: IN (0x0001)
```

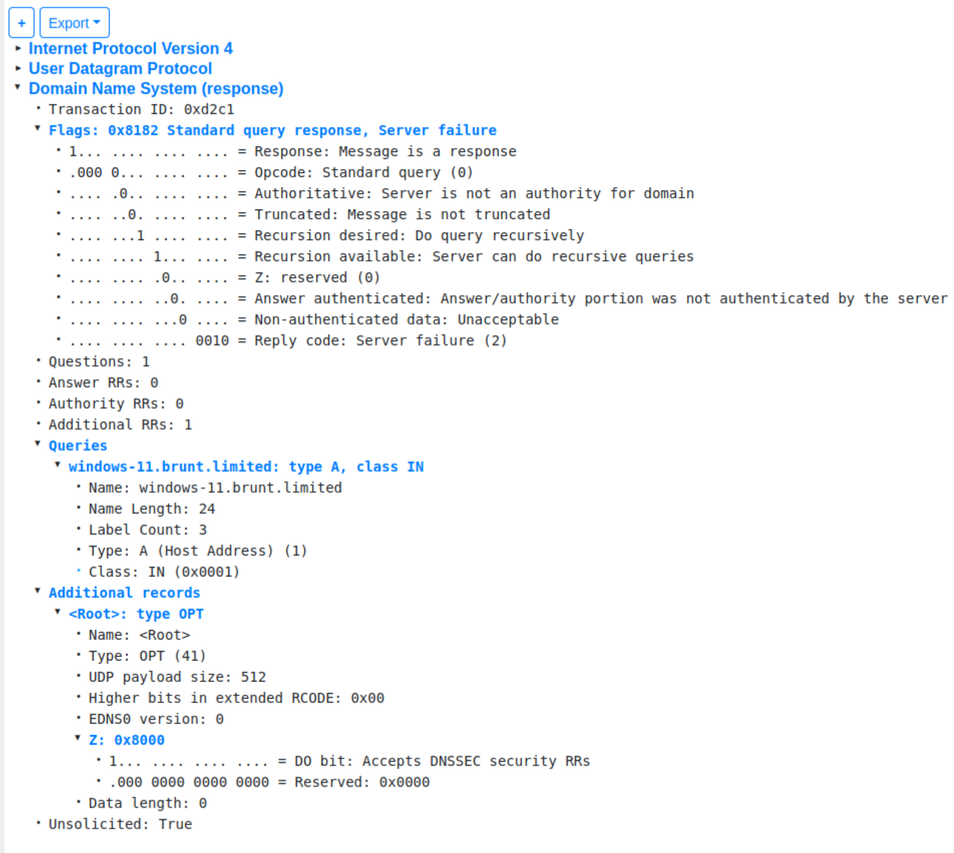
Looks pretty bog standard. How's the upstream query look?

+ Export ▾

```
▸ Internet Protocol Version 4
▸ User Datagram Protocol
▾ Domain Name System (query)
    • Transaction ID: 0xd2c1
    ▾ Flags: 0x0100 Standard query
        • 0... .... .... .... = Response: Message is a query
        • .000 0... .... .... = Opcode: Standard query (0)
        • .... ..0. .... .... = Truncated: Message is not truncated
        • .... ...1 .... .... = Recursion desired: Do query recursively
        • .... .... .0.. .... = Z: reserved (0)
        • .... .... ...0 .... = Non-authenticated data: Unacceptable
    • Questions: 1
    • Answer RRs: 0
    • Authority RRs: 0
    • Additional RRs: 1
    ▾ Queries
        ▾ windows-11.brunt.limited: type A, class IN
            • Name: windows-11.brunt.limited
            • Name Length: 24
            • Label Count: 3
            • Type: A (Host Address) (1)
            • Class: IN (0x0001)
    ▾ Additional records
        ▾ <Root>: type OPT
            • Name: <Root>
            • Type: OPT (41)
            • UDP payload size: 2048
            • Higher bits in extended RCODE: 0x00
            • EDNS0 version: 0
            ▾ Z: 0x8000
                • 1... .... .... .... = DO bit: Accepts DNSSEC security RRs
                • .000 0000 0000 0000 = Reserved: 0x0000
            • Data length: 0
```

It's added an EDNS record - again, doesn't look too objectionable.

So, response packet



```
▸ Internet Protocol Version 4
▸ User Datagram Protocol
▾ Domain Name System (response)
    • Transaction ID: 0xd2c1
  ▾ Flags: 0x8182 Standard query response, Server failure
      • 1... .... .... .... = Response: Message is a response
      • .000 0... .... .... = Opcode: Standard query (0)
      • .... .0.. .... .... = Authoritative: Server is not an authority for domain
      • .... ..0. .... .... = Truncated: Message is not truncated
      • .... ...1 .... .... = Recursion desired: Do query recursively
      • .... .... 1... .... = Recursion available: Server can do recursive queries
      • .... .... .0.. .... = Z: reserved (0)
      • .... .... ..0. .... = Answer authenticated: Answer/authority portion was not authenticated by the server
      • .... .... ...0 .... = Non-authenticated data: Unacceptable
      • .... .... .... 0010 = Reply code: Server failure (2)
    • Questions: 1
    • Answer RRs: 0
    • Authority RRs: 0
    • Additional RRs: 1
  ▾ Queries
    ▾ windows-11.brunt.limited: type A, class IN
        • Name: windows-11.brunt.limited
        • Name Length: 24
        • Label Count: 3
        • Type: A (Host Address) (1)
        • Class: IN (0x0001)
  ▾ Additional records
    ▾ <Root>: type OPT
        • Name: <Root>
        • Type: OPT (41)
        • UDP payload size: 512
        • Higher bits in extended RCODE: 0x00
        • EDNS0 version: 0
      ▾ Z: 0x8000
          • 1... .... .... .... = DO bit: Accepts DNSSEC security RRs
          • .000 0000 0000 0000 = Reserved: 0x0000
        • Data length: 0
    • Unsolicited: True
```

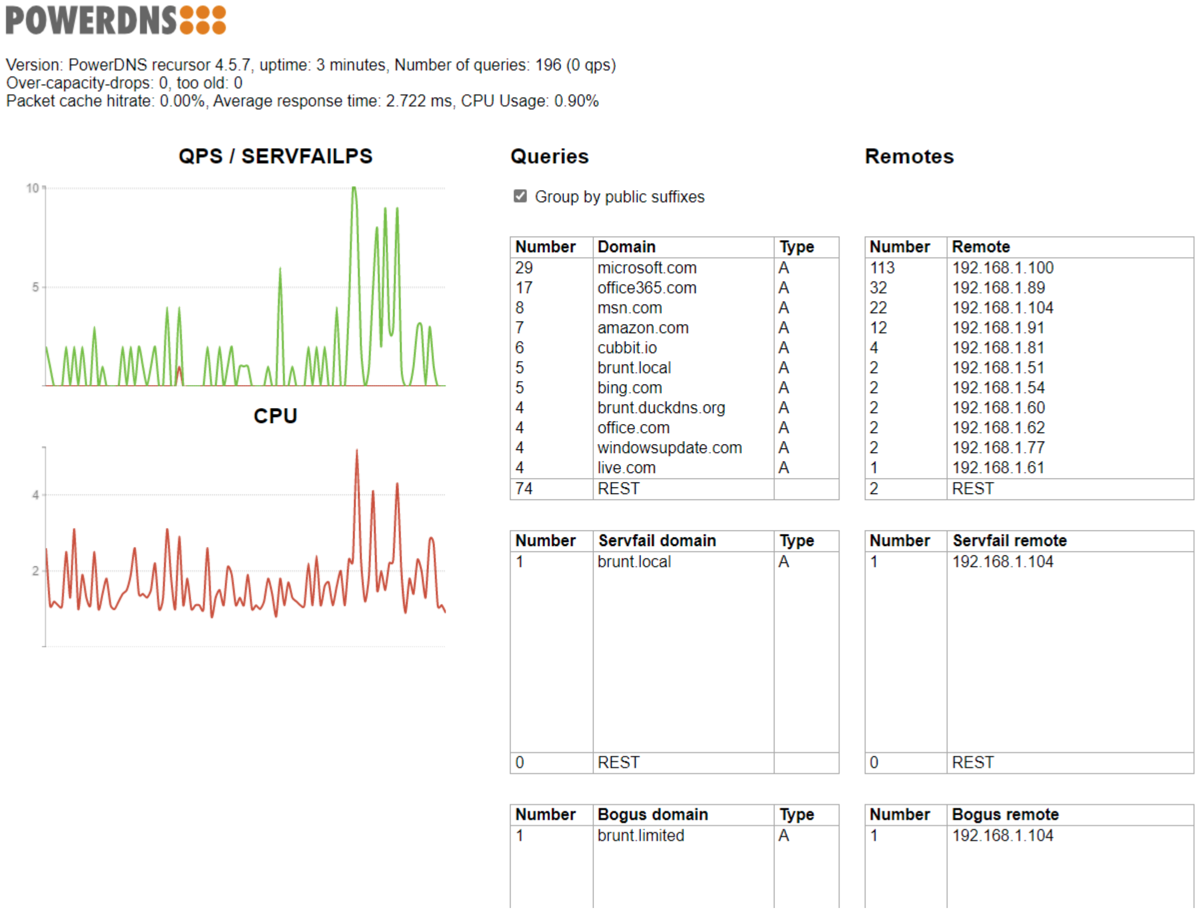
Nothing really telling there, other than that it's a `SERVFAIL`

Flag wise, they're basically identical to my working queries.

What's your upstream DNS - is it actually pdns or is that just a coincidental name? Either way, you'll need to check the logs there. If it *is* `pdns` then there are a variety of possibilities, especially if you're using regex matches in any backends or similar

danielbrunt57 commented on Mar 23

So, the query *IS* going from homeassistant to hassio_dns to pdns but generating a SERVFAIL. Yet a direct query from homeassistant to pdns succeeds as does a query direct from hassio_dns to pdns and windows-11 to pdns. I am seeing the failure in my pdns recursor...



POWERDNS

Version: PowerDNS recursor 4.5.7, uptime: 3 minutes, Number of queries: 196 (0 qps)
Over-capacity-drops: 0, too old: 0
Packet cache hitrate: 0.00%, Average response time: 2.722 ms, CPU Usage: 0.90%

Queries

☑ Group by public suffixes

Number	Domain	Type
29	microsoft.com	A
17	office365.com	A
8	msn.com	A
7	amazon.com	A
6	cubbit.io	A
5	brunt.local	A
5	bing.com	A
4	brunt.duckdns.org	A
4	office.com	A
4	windowsupdate.com	A
4	live.com	A
74	REST	

Number	Servfail domain	Type
1	brunt.local	A
0	REST	

Number	Bogus domain	Type
1	brunt.limited	A

Remotes

Number	Remote
113	192.168.1.100
32	192.168.1.89
22	192.168.1.104
12	192.168.1.91
4	192.168.1.81
2	192.168.1.51
2	192.168.1.54
2	192.168.1.60
2	192.168.1.62
2	192.168.1.77
1	192.168.1.61
2	REST

Number	Servfail remote
1	192.168.1.104
0	REST

Number	Bogus remote
1	192.168.1.104

danielbrunt57 commented on Mar 23

I am running pdns-recursor on port 53 and pdns on port 5300 for my local zones.
The recursor.conf contains:

```
#allow-from=127.0.0.0/8, 10.0.0.0/8, 100.64.0.0/10, 169.254.0.0/16, 192.168.0.0/16, 172.16.0.0/12, ::1/128, fc00::/7, fe80::/10
api-key=eaae8df1-df1e-472d-910c-01ec2b7ef719
dnssec=process
etc-hosts-file=/etc/hosts
export-etc-hosts=yes
forward-zones=brunt.limited=127.0.0.1:5300, local.ca=127.0.0.1:5300, brunt.local=127.0.0.1:5300, brunt.lan=127.0.0.1:5300, 1.168.192.in-addr.arpa=127.0.0.1:5300,
brunt.duckdns.org=127.0.0.1:5300, brunt.ca=127.0.0.1:5300
#forward-zones=brunt.local=127.0.0.1:5300, 1.168.192.in-addr.arpa=127.0.0.1:5300, brunt.duckdns.org=127.0.0.1:5300
include-dir=/etc/powerdns/recursor.d
local-address=192.168.1.103
local-port=53
lua-config-file=/etc/powerdns/recursor.lua
webserver=yes
webserver-port=8082
```

bentasker commented on Mar 23

Yes. The primary difference between your direct query and the one via `coredns` is the injection of that additional record - it enables DNSSEC.

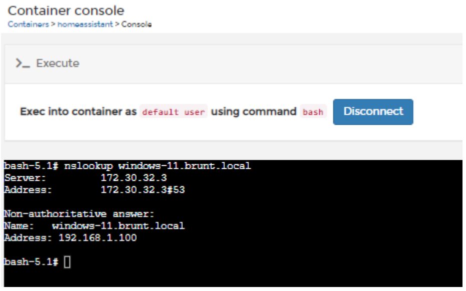
Have you ever had DNSSEC enabled for that domain? If so, pdns will have cached the RRs.

If you do `journalctl -f -u pdns` on the pdns box and run the failing query again, do you get anything interesting?

You could also edit the `pdns` config and set `dnssec off` (reload the service after). If queries start working, you've found the culprit.

danielbrunt57 commented on Mar 23 • edited

I just came back here after troubleshooting numerous pdns options and internet googling. Last article I read was
serverfault.com 1076985 powerdns-auth-and-recursor-bug-with-one-domain
I added dnssec=off to recursor.conf, systemctl restart pdns-recursor and tried nslookup windows-11.brunt.local from homeassistant and...


Container console
Containers > homeassistant > Console
>_ Execute
Exec into container as default user using command bash
Disconnect

```
bash-5.1# nslookup windows-11.brunt.local
Server:         172.30.32.3
Address:        172.30.32.3#53

Non-authoritative answer:
Name:   windows-11.brunt.local
Address: 192.168.1.100

bash-5.1# 
```

Success!
Yes, I did have DNSSEC enabled on the local, limited and lan zones in pdns at one point but later disabled it. pdns-recursor must have that remembered somewhere despite issuing rec_control reload-zones.
Thanks a bunch for working though this with me...I owe you a beer! -Daniel

bentasker commented on Mar 23

Excellent!

I *think* to flush the DNSSEC meta cache you want `pdns_control purge`

danielbrunt57 commented on Mar 23 • edited

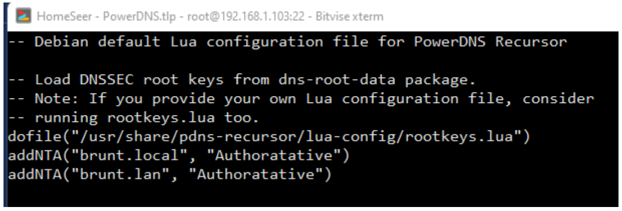
No, that did not work. I have changed dnssec=off to dnssec=process-no-validate and homeassistant is still able to resolve.

bentasker commented on Mar 23

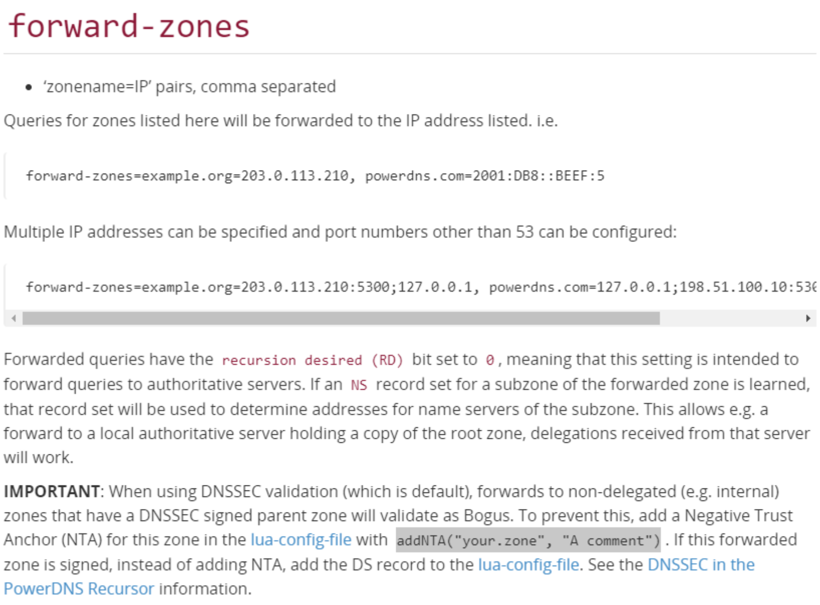
Awesome!

danielbrunt57 commented on Mar 23 • edited

I finally nailed it!! I now have the pdns default setting dnssec=process and .local/.lan resolving from homeassistant.
The magic elixir was an lua record...


HomeSeer - PowerDNS.tlp - root@192.168.1.103:22 - Bitvise xterm

```
-- Debian default Lua configuration file for PowerDNS Recursor

-- Load DNSSEC root keys from dns-root-data package.
-- Note: If you provide your own Lua configuration file, consider
-- running rootkeys.lua too.
dofile("/usr/share/pdns-recursor/lua-config/rootkeys.lua")
addNTA("brunt.local", "Authoratative")
addNTA("brunt.lan", "Authoratative")
```



forward-zones

- 'zonename=IP' pairs, comma separated

Queries for zones listed here will be forwarded to the IP address listed. i.e.

```
forward-zones=example.org=203.0.113.210, powerdns.com=2001:DB8::BEEF:5
```

Multiple IP addresses can be specified and port numbers other than 53 can be configured:

```
forward-zones=example.org=203.0.113.210:5300;127.0.0.1, powerdns.com=127.0.0.1;198.51.100.10:53
```

Forwarded queries have the `recursion desired (RD)` bit set to `0`, meaning that this setting is intended to forward queries to authoritative servers. If an `NS` record set for a subzone of the forwarded zone is learned, that record set will be used to determine addresses for name servers of the subzone. This allows e.g. a forward to a local authoritative server holding a copy of the root zone, delegations received from that server will work.

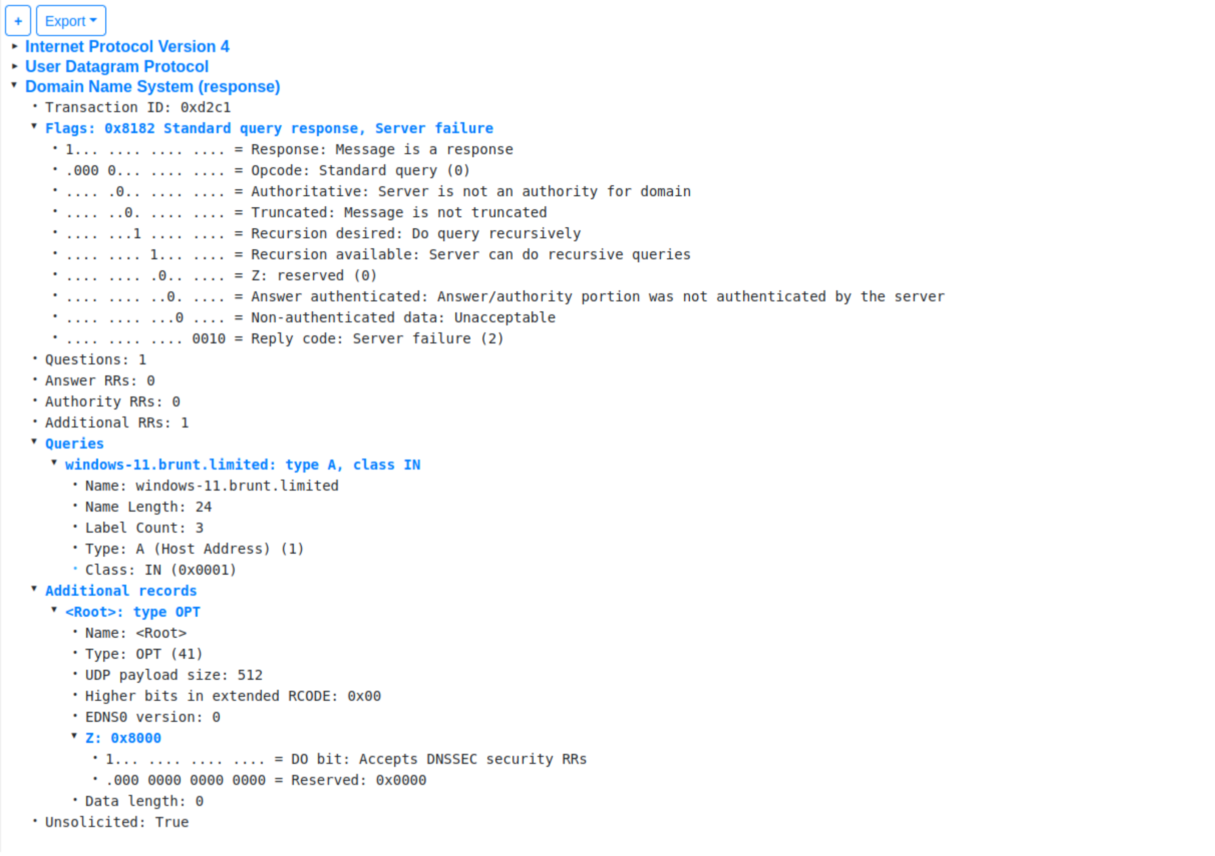
IMPORTANT: When using DNSSEC validation (which is default), forwards to non-delegated (e.g. internal) zones that have a DNSSEC signed parent zone will validate as Bogus. To prevent this, add a Negative Trust Anchor (NTA) for this zone in the lua-config-file with `addNTA("your.zone", "A comment")`. If this forwarded zone is signed, instead of adding NTA, add the DS record to the lua-config-file. See the DNSSEC in the PowerDNS Recursor information.

🎉 1

danielbrunt57 commented on Mar 24 • edited

What did you use to obtain this?

response packet



```
+  Export ▾
▸ Internet Protocol Version 4
▸ User Datagram Protocol
▾ Domain Name System (response)
    • Transaction ID: 0xd2c1
  ▾ Flags: 0x8182 Standard query response, Server failure
      • 1... .... .... .... = Response: Message is a response
      • .000 0... .... .... = Opcode: Standard query (0)
      • .... .0.. .... .... = Authoritative: Server is not an authority for domain
      • .... ..0. .... .... = Truncated: Message is not truncated
      • .... ...1 .... .... = Recursion desired: Do query recursively
      • .... .... 1... .... = Recursion available: Server can do recursive queries
      • .... .... .0.. .... = Z: reserved (0)
      • .... .... ..0. .... = Answer authenticated: Answer/authority portion was not authenticated by the server
      • .... .... ...0 .... = Non-authenticated data: Unacceptable
      • .... .... .... 0010 = Reply code: Server failure (2)
    • Questions: 1
    • Answer RRs: 0
    • Authority RRs: 0
    • Additional RRs: 1
  ▾ Queries
    ▾ windows-11.brunt.limited: type A, class IN
        • Name: windows-11.brunt.limited
        • Name Length: 24
        • Label Count: 3
        • Type: A (Host Address) (1)
        • Class: IN (0x0001)
  ▾ Additional records
    ▾ <Root>: type OPT
        • Name: <Root>
        • Type: OPT (41)
        • UDP payload size: 512
        • Higher bits in extended RCODE: 0x00
        • EDNS0 version: 0
      ▾ Z: 0x8000
          • 1... .... .... .... = DO bit: Accepts DNSSEC security RRs
          • .000 0000 0000 0000 = Reserved: 0x0000
        • Data length: 0
    • Unsolicited: True
```

bentasker commented on Mar 24

> What did you use to obtain this?

I used this - https://hpd.gasmi.net/?data=45000051E35C40004011C90EAC1E2003C0A80167C1620035003D8E7FD2C1010000010000000000010A77696E646F77732D3131056272756E74076C696D6974656400000100010000290800000080000000&force=ipv4

The hex comes from the `tcpdump -x` you provided:

```
11:51:02.740107 eth0  Out IP (tos 0x0, ttl 64, id 58204, offset 0, flags [DF], proto UDP (17), length 81)
    hassio-dns.49506 > pdns.brunt.ca.53: 53953+ [1au] A? windows-11.brunt.limited. (53)
        0x0000:  4500 0051 e35c 4000 4011 c90e ac1e 2003
        0x0010:  c0a8 0167 c162 0035 003d 8e7f d2c1 0100
        0x0020:  0001 0000 0000 0001 0a77 696e 646f 7773
        0x0030:  2d31 3105 6272 756e 7407 6c69 6d69 7465
        0x0040:  6400 0001 0001 0000 2908 0000 0080 0000
        0x0050:  00
```

Take the hex section and trim the indexes off

```
4500 0051 e35c 4000 4011 c90e ac1e 2003
c0a8 0167 c162 0035 003d 8e7f d2c1 0100
0001 0000 0000 0001 0a77 696e 646f 7773
2d31 3105 6272 756e 7407 6c69 6d69 7465
6400 0001 0001 0000 2908 0000 0080 0000
00
```

And stick that in the text-area. It'll whinge about lack of ethernet header, but you can then tell it to treat it as an IPv4 packet and the dissectors will kick in.

danielbrunt57 commented on Mar 24

Thanks! Filing for future reference!

👍 1

Vip0r commented on Apr 3

can confirm issue on my side. HA stops using my internal DNS for whatever reasons and try to resolve local hostnames to the hardcoded cloudflare DNS. Unfortunately it does not fall back until i restart HA / dns

👍 1

mdegat01 mentioned this issue on Apr 4

Cloudflare as fallback only and no healthcheck #82

Merged

mdegat01 commented on Apr 25 Contributor

Fixed by #82

mdegat01 closed this as completed on Apr 25

mdegat01 commented on Apr 25 Contributor

Also note that there is a new option to disable the fallback dns added here: home-assistant/supervisor#3586 as I would guess a number of users on here would be interested in that.

👍 2

alexdelprete commented on Apr 26

Thank you so much Mike. Glad someone in the team finally acknowledged how many issues the old implementation created to many users. I'm not able to test it because I switched to Core, but I'm pretty sure many others will test it and report back.

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

24 participants

and others