

Bug #8987



Web GUI main page very slow to load if wan interface is enabled but not connected.

Added by [Arnaldo Pirrone](#) 7 months ago. Updated 10 days ago.

Status:	New	Start date:	10/01/2018
Priority:	Very Low	Due date:	
Assignee:	-	% Done:	<div></div> 0%
Category:	Web Interface	Estimated time:	
Target version:	Future	Affected Architecture:	
Affected Version:			

Description

Hi,
 I noticed this annoying bug in pfSense 2.4.4:
 by configuring the wan interface and leaving it disconnected,
 the main page of the web GUI becomes very slow to load (you must wait many minutes!) though you can reach every other page.
 you can verify this by enabling and disabling the WAN interface from the interfaces menu (always available), then by clicking on the logo.

History



Updated by [Jim Pingle](#) 7 months ago

#1

- **Priority** changed from *Normal* to *Very Low*
- **Target version** set to *Future*

There are several things on the dashboard that need working DNS and connectivity, like the update check, packages widget, etc.

Without connectivity it has to wait for DNS to timeout for various things. There isn't a way around that.

Using the local DNS Resolver/Forwarder can help -- see [#1407](#)

There may be something more we can do here in the future, but it's doubtful.



Updated by [James Howel](#) 3 months ago

#2

To add to this bug we've been using pfSense 2.3.5 for an internal project and its been working brilliantly.

We're using pfSense as a cluster but it has no internet connectivity and 2.3.5 has no issues with this.

I upgraded the first cluster node with 2.4.4_p1 and noticed this laggy behaviour generally when trying to load the dashboard. After spending several hours trying to figure out what was wrong I ended up having to roll back to snapshot and leave the cluster on 2.3.5.

Testing this further, I have found that on top of the dashboard hanging for up to 2 minutes, certain other configuration saves behave the same way, such as DNS Resolver, Advanced Settings and others. Firewall Rules saves do not exhibit the same issue.

Essentially this will preclude pfSense from being deployed where there is no internet connectivity as this issue means that configuration will take 10 times longer to perform and for some people they will see this as a bug and opt to use another platform.

I'm a long term pfSense user and its gaining traction on this project but ultimately will have to be abandoned without some resolution or workaround to this problem as this will be a showstopper for using pfSense here.



Updated by [Luke Hamburg](#) 3 months ago

#3

If you remove all widgets from the dashboard does that help at all? It's probably a widget that's causing this delay.



Updated by [James Howel](#) 3 months ago

#4

Hi Luke,

Thanks for the suggestion but I've tried that, same issue.

It looks like whatever is timing out due to no internet connectivity affects several areas so it's not just the dashboard...

James



Updated by [Joshua Sign](#) 3 months ago

#5

Hi,

I just test it :

- Loading dashboard normaly takes about 1 second or less.
- Without WAN connectivity, it takes about 30 seconds.
- Adding theses options in resolv.conf, it now takes only 5 seconds :

```
options timeout:1
options attempts:1
```

I don't know if it can help, but when WAN connectivity goes down, we got an alert in systems logs like :

```
Jan 15 15:51:52 php-fpm /rc.linkup: Hotplug event detected for WAN(wan) static IP (xxx.xxx.xxx.xxx)
Jan 15 15:51:51 kernel em0: link state changed to DOWN
Jan 15 15:51:51 check_reload_status Linkup starting em0
```

So if these options are good to use when connexion is down (there are many cases to consider), it can be a workaround to add them 'on fly' in resolv.conf when WAN goes down and then remove them when WAN goes UP.

It's a suggestion...



Updated by **Maverick Phillips** 3 months ago

#6

Hello,

One of my two firewalls has developed this issue - I can confirm disabling the WAN adapter resolved this slowness !



Updated by **James Howel** 3 months ago

#7

Hi Joshua,

Thanks for looking at this.

We don't have a WAN in a down state, it is connected but it has no NAT and is basically just another interface but on a totally isolated network.

It appears that if pfSense has NEVER been connected to the internet, the way it behaves with the timeouts is different to IF it has been connected at least once and then disconnected.

Build scenario: 2.4.4 p1 build, 2 interfaces, vsphere 6.5 VM, no configuration aside from adding ip addresses from console and setting the admin pass from gui. Never internet connected.

- Dashboard access once logged in: 66 seconds
- General page save after adding one dns ip: 63 seconds
- General page save with no config changes: 66 seconds
- Rules save: instant
- Post rules save Apply Changes: Instant

Architecturally, there is a significant dependency on internet connectivity in pfSense which for most people is fine but if an internal pfSense tier or the entire pfSense implementation can never be connected to the internet then it's very slow to configure and troubleshoot. When compared to an offline hardware firewall from another vendor pfSense behaves like its buggy and slow.

I've also discovered that it's also fundamentally impossible to install any packages in an offline state which is another problem but that's a separate issue.

James



Updated by **Corey Bock** 3 months ago

#8

Maverick Phillips wrote:

Hello,

One of my two firewalls has developed this issue - I can confirm disabling the WAN adapter resolved this slowness !

I've recently just discovered this issue on configs made with 2.4.4-RELEASE-p2 for both the SG8860 and the XG1541.

GUI has become almost unusable on both after updating. This is a problem as we're often teching the hardware in a shop offline. Also, in the event a single WAN setup loses connectivity --right when you'd wanna log in and check it out-- you're unable to get in for at least 60 seconds.

I can also confirm that disabling the offline WAN interfaces will resolve the problem. However, I noticed that only statically assigned WAN interfaces cause this issue. If I enable a WAN interface set to DHCP, even if it's not online, the problem does not exist as it does with the same interface set to static.

I hope this helps get this resolved!



Updated by **Joshua Sign** 3 months ago

#9

James Howel wrote:

It appears that if pfSense has NEVER been connected to the internet, the way it behaves with the timeouts is different to IF it has been connected at least once and then disconnected.

...

Architecturally, there is a significant dependency on internet connectivity in pfSense which for most people is fine but if an internal pfSense tier or the entire pfSense implementation can never be connected to the internet then it's very slow to configure and troubleshoot. When compared to an offline hardware firewall from another vendor pfSense behaves like its buggy and slow.

I just try to understand, i will be able to try to reproduce it, i hope next week.

In the case you explain, i understand that internet connectivity is not possible and has never be done.

But does dns resolution is possible ?

For local domains and remote domains (like google.com) ?

The case you talk about sound like a DNS latency problem.

Do you try to play with it ?

Do you use a spécific DNS configuration ?

And finally, how are you doing for now : you always wait 1 minute between pages, or do you find a workaround ?

Josh_



Updated by **Tom Embt** 3 months ago

#10

I have also noticed this issue on my home pfSense. I was able to reproduce it reliably with a VM and it appears to happen when the WAN interface has an IP address assigned (and thus has a related DNS server in resolv.conf) but that connectivity is no longer working.

It happens to my home pfSense when the WAN interface has a DHCP lease and then something breaks for a while at the link level. For the VM reproduction I installed under VirtualBox with the WAN interface bridged to my laptop wifi and the LAN interface on a "host only network". If I allow the VM's WAN to get an IP on my network and then shut off wifi on my laptop, attempts to load the pfSense dashboard take 60+ seconds. It does not necessarily break immediately, which I believe to be because of DNS cache. Going to Services > DNS Resolver and restarting that service will cause it to break if it was not already doing so.

Doing some tcpdump of outbound DNS when this is breaking leads me to the hostname "ews.netgate.com", which is what it's trying and failing to look up. I confirmed that host-filing that IP temporarily makes the problem disappear. I locally munged some code for troubleshooting, and it would seem that in the above case at least, the issue is src/etc/inc/copyright.inc . Looks like that copyright download functionality is a fairly recent addition.



Updated by **Jamie Donovan** about 2 months ago

#11

This is affecting our company's setup as well. Static public IPs /29 (total 5 available IPs) with one hooked up with a virtual IP alias on pfsense.

However, the connection is up and running – no interfaces are disconnected (apart from openvpn servers that have not been assigned interfaces). And even after removing the virtual IP the dashboard is still stuck for what seems like minutes.

With dhcp it works fine, but obviously static assignment is the only option here. I see no workarounds atm.

Edit: I see that I was missing DNS servers after reconfiguring. The workaround for me is to enter DNS servers manually. DNS Forwarder & DNS Resolver services are both turned off; this pfsense works purely as a router/firewall, no DNS or DHCP services.



Updated by **Joshua Sign** about 2 months ago

#12

could you confirm that adding DNS entries can be a workaround ? (if you can try to do it for testing purpose)
How many seconds are you waiting for the dashboard with this... And is it acceptable if it is ?

Tks
Josh_



Updated by **Pieter** . about 2 months ago

#13

We had the same issue. It's a pfSense 2.4.4p2 installation in an air-gapped environment and has never touched the internet. The home page is unbearably slow to load, but all the other pages load just fine.

We made a network capture and saw repeated DNS requests to ews.netgate.com when loading the home page. When we did a host override in the DNS for ews.netgate.com to localhost, the home page loaded almost instantly, just like the other pages.

A little digging showed that in src/usr/local/www/index.php on line 469 the copyget.inc file is included. In this file there is an attempt to download the file <https://ews.netgate.com/copyright> every 24 hours or if the local copyright file doesn't exist. In an air-gapped environment, this file isn't updated so there is an attempt to download a fresh copy every time the homepage loads.

The attempted download is done in a way that blocks the PHP renderer from doing other things, so it waits for a number of timeouts on the DNS request before finishing to process the index.php file. Hence the very long loading time.



Updated by **Luke Hamburg** about 2 months ago

#14

Hmm, nice find Pieter!

Maybe we need a function like **haveWorkingDns()** that returns a bool if DNS is working, and then use that in the include so it only tries to fetch from Netgate if the retval is true...



Updated by **Tom Embt** about 2 months ago

#15

Looks like Pieter and I have come to the same conclusion (see comment 10), hopefully a fix isn't too far out.



Updated by **Carsten F** 12 days ago

#16

Since 3 hours I'm having the exact same issue!



Updated by **John Burwell** 12 days ago

#17

Speaking from some recent experience:

This behavior interferes with troubleshooting if the root cause is a WAN failure, and contributes to misdirected concerns that the firewall itself has failed or is in a degraded state, when it is not (apart from the WAN failure).

This behavior also interferes with configuring a firewall with foreign static WAN IPs before shipping to a remote site for installation.

It also interferes with restoring a backup from a failed firewall device to its replacement firewall device if the interfaces must be reassigned, preventing access to the WAN until reassignment (which requires loading pages that take forever to load).

Edge cases, perhaps, but not the kind of thing anyone wants to deal with when they're trying to fix some other problem.

Based on Pieter's findings, a solution might be to shift the copyright loading mechanism to an asynchronous process, through XHR or something, as is done with the update check, so that the page can finish loading while the Internet times out.



Updated by **Mr Reed** 12 days ago

#18

<https://ews.netgate.com/copyright> is down right now (504 Gateway Timeout): all attempts at loading the dashboard are painfully slow.
comment number 10 was right on the money

This behavior has to be corrected.



Updated by **Troy Emmerson** 12 days ago

#19

Hello Gentlemen,

Been sandbagging this thread as I've ran into this issue several times and I think I have an easy workaround.

I discovered some interesting annoyances about unbound during the boot process that I believe contribute to the issue:

1) When Unbound first loads, and before the pfsense configuration is loaded, it is basically in default config such that it will accept requests and try to resolve them via recursion to the root servers.

2) Even if Unbound is configured for SSL only, as some point in the configuration load it will try to contact configured forwarders via non-SSL (port 53). (not sure what the trigger is for this, but I've confirmed it several times)

If you have your DNS locked down to use only SSL and only to forwarders, and all other DNS traffic is blocked (thus causing Unbound to wait until time-out), either internally, or externally, this will cause Unbound to wait and time-out.

So, following the standard practice method to prevent such time-out delays, is the use case for the proper use of reject rules. (same precedence as with IDENT packets).

Thus by adding a mix of policy and non-policy outbound reject float rules to send a reject back to Unbound when the WAN link is down, or Unbound is attempting to use other than configured server/protocol, this causes unbound to return an immediate NXDOMAIN, thus allowing the boot process to move and not get tied up waiting for DNS lookups.

And after reading over this thread a while back I added similar rules to send a reject back to firewall when accessing <https://ews.netgate.com>, if there is no available outbound link.

These rule settings do require configuring Settings >> Advanced >> Miscellaneous >> Skip rules when gateway is down: set to true, in order to allow the use of "dynamic" policy rules.

Then simply create rules that allow, by policy, the DNS and ews.netgate.com traffic out, when the WAN link is up, and a non-policy reject rule that catches the traffic that will cause the timeout, when the WAN link is down.

Anyway, so far, just by putting rules in place to circumvent protocol time-out during boot, I no longer experience the long delays accessing the GUI during the boot process.



Updated by **Carsten F** 11 days ago

#20

Well I think it's only because ews.netgate.com is down. I've override the host to localhost and this solves the problem a bit. WebGUI now needs around 2s to load. If WAN is down, it loads instant.



Updated by **Troy Emmerson** 10 days ago

#21

With WAN down and that being the only default route, this should result in an "No route to host" error, and no connection attempt, thus no time-out waiting for connection.

Which brings to mind, if System >> Routing >> <WAN gateway> >> Gateway Action is set to enabled, to disabled gateway action events, this would result in the pfSense holding the WAN up and thus available for routing, even if the route is not truly up and ready.

I also forgot to mention the the policy rule that I use to allow the ews.netgate.com traffic out uses a host alias such that the rule only gets instantiated when both the gateway is up and there is a good DNS lookup to fill in the IP for alias entry from the host name. Thus traffic only gets passed with both the DNS are operating (assuming the DNS also needs outbound connectivity to get a good lookup) and the gateway is up, otherwise the with no rule to pass the traffic the reject rule short circuits outbound traffic that may result in a timeout.

But of course none of this will defeat a service outage at the host itself. So ultimately, the functions that attempt the connection should implemented in way to prevent blocking anyway during edge case outages situations.