Gmail

John Loop <pccitizen@gmail.com>

John Loops' 3-17-2024 "imonitorg" [and now also "iotsnoop"] newsletter: New project: IOT monitoring raspberry pi4 APN - connect to your home router and sub your IOT to watch them!

John Loop <jdloop@imonitorg.com> Tue, Mar 19, 2024 at 1:24 PM
To: "pccitizen@gmail.com" <pccitizen@gmail.com>

Dear imonitor[g] users, potential users, former users, and interested parties: [and NOW potential iotsnoop users!]

Apologies if you do not want to receive this newsletter, sent 3,4 times per year [just reply to remove your email]. I have collected email addresses from some of my friends in hopes you might be interested.

*Summary for those of you new to this newsletter:**

I am working on a project to perform Internet/Local Network monitoring. I use a small device[raspberry pi3B or pi3B+], along with custom scripts to perform this service. I have deployed this device to 25+ partners across the country representing many different access technologies and ISPs who help in the development. As of 2024 I now have Virtual Machine [VirtualBox] images, so you do not even need a separate raspberry pi device! I initially targeted the service in 2018 to assist in troubleshooting for my friends on our mountain, but it is applicable universally, and I have helpers across the country on Windstream, ATT, Ting, Spectrum, Comcast, CenturyLink, Tmobile, Starlink and several other ISPs. It spans all access technologies from ADSL, VDSL, WADSL, cellular, cable, satellite and fiber. I also created a "generic" version which is completely standalone, and has no connection to my servers at all. I posted these generic images on sourceforge.net. I also established a github page pointing to it. There is absolutely no third party server in the internet which you have to access to collect your info --it is all on a webpage on your local raspberry pi! All you have to do is download an image for a microSD card [you can get it from me if you want], plug it into your raspberry pi3B or 3B+, connect the ethernet and enjoy. Emphasizing again: There is NO THIRD PARTY SERVER used! You can configure an email using your gmail account as a relay.

As of January 2024, I am now working on an extension of this project, called "IOT [The Internet of Things] monitoring" and I call this "iotsnoop." I have the domainname "iotsnoop.com" registered to record this project. This email introduces the iotsnoop project to interested parties.

3-17-2024 Newsletter

The advances in Internet technology and devices is truly mind boggling. We have absolutely no idea where this is going, especially with "AI" coming on stream. Like you no doubt, I am availing myself of "bard" and "chatgpt" in my daily work. You no longer need manuals, or to scroll thru google search results! Just ask a specific enough question of bard or chatgpt, and there you have it. You have to be specific and you need to ask more than once and cross reference, but it is amazing. It solves all my bash scripting syntax e.g. which is notoriously difficult! Just DON'T ask anything reglious or political for heaven's sake. This is where it may really get dangerous!

So one of the MOST dangerous aspects of this Internet blitz is all the gadgets you put in your home. Every one of your new house gadgets now has a wifi interface and checks in with some remote server [at least if you want the "service" such as remote control/interrogation]! In the words of one of my favorite people, Steve Gibson, "what could possibly go wrong!?" Just think about it. You have software/firmware from dozens of companies running on gadgets INSIDE your home, INSIDE your network. What could possibly go wrong!? I have maybe 20 gadgets like this. I know some of my "subs" have MANY more and are fully "into" this new universe. Caution would be advised!!

What you should know about most of these IOT gadgets is that they run on VERY outdated code, typically Linux/busybox OS which may be years old, and will NEVER be updated [WHO is going to update your light bulb code!!]. Once vulnerabilities are uncovered, and especially if the remote SERVERs get compromised, malicious agents will be running ON IOT gadgets on your network. This is a disaster waiting to explode. It rivals the previous problem of routers having open ports on the internet!

The first line of defense we have is to put all these UNKNOWN gadgets on their separate network, isolated from your main PCs, phones. Most routers have a "guest" network available to accomplish this -of course the IOT gadgets are almost all wifi. If you have your IOT gadgets on your main network, your primary PCs/phones are open to possible probing/compromise by compromised IOT devices. The IOT devices are communicating in real time with servers all over the world. What happens if the servers get compromised!? They have direct access to your home network via the -now- compromised IOT gadget. We no longer need "ports" open on the Internet to be susceptible, all we need are IOT gadgets ALREADY on our inside network, and which can be leveraged or compromised by questionable actors. In technical terms, these IOT gadgets perform the outbound TCP connection, so it is TRIVIAL that the far end can connect back over that ALLOWED outbound connection.

To address this concern I have been working on additions to my "imonitorg" project. I decided to transition to a raspberry pi4 because of the added capability needed. I call this "iotsnoop" and it allows you to use the pi4 as a wifi "access point" on your network to terminate all the IOT gadgets, i.e. it has its own wifi SSID/key, just like other wifi APNs. The added functions on the pi4 gobble up the packets originating from the IOT devices and analyzes them in hopes of monitoring the IOT network, much as imonitorg monitors your Internet network. The "version 1" characteristics of "iotsnoop" are as follows:

1. The wifi is purposely limited to 802.11g -50Mb/s. WHY would you want your IOT gobbling up your bandwidth! It does not offer the latest wifi protocols for this reason! It is also limited to ipv4 [not ipv6].

2. All IOT DNS queries are captured. Separate counts are done for UDP packets, TCP sessions, and IP host pairs. DNS queries can be revealing!

3. All packets are captured using up to a maximum of 1GB storage buffer [this is version 1]. This represents about 3Mb/s average for an hour [rcv+tmt], or 40 Mb/s for about 5 minutes. The capture is stopped and alerts are displayed if this is exceeded. Watching my own IOT network of 12+ devices, they rarely exceed 200MB capture in an hour unless I stream. I will be working on a version 2 to expand this by different capture practices.

4. A 32 bit raspberry pi4 is used -running buster. The image will not work on raspberry pi 3B[+] or a 64 bit raspberry pi4. I felt the added performance of the pi4 was needed, and 4GB RAM is recommended [8 GB very good tho!], but the 64 bit version is pretty new and rare still.

5. Capture is done for an hour and then the results are analyzed and displayed in the form of lists and plots, and alerts.

6. I am working on various algorithms to display results, such as collecting mdns hostnames which help ID the IOT devices, and parsing DNS queries for each IP/IOT, PLUS showing a rolling graph of hourly results. Looking at the plots will give a gross indication of usage, and further interrogation of the data is possible. Watching the hour to hour DNS actively is also revealing. From hour to hour the same set of IOT will generate the same DNS query traffic. Malicious code may show wildly differing DNS queries. Future versions may use APIs to submit the DNS query list to sites which will flag malicious domains!

7. The "iotsnoop" functionality is in addition to the "imonitorg" functionality provided on previous pi3B, pi3B+, ovas! "Imonitorg" functionality has been ported, and all its functions [save a few such as pihole] have been transported to iotsnoop pi4! I am recommending a 4GB pi4 raspberry, and an 8GB if you feel like it. Amazon has many: https://www.amazon.com/s?k=raspberry+pi+4&crid=2QJ5PFKCT6E42&sprefix=rasp%2Caps%2C276&ref=nb_sb_ss_ts-doa-p_3_4 Ask me if you

want a recommend.

8. I am restricting this application to a "generic" capability and not providing a "captive subscriber" capability like we use with the current 25 subs on imonitorg. But, there is still the ability to use your gmail account as a relay and receive a daily email from the iotsnoop pi4. I will not be hosting a server to terminate the connections for development like I did for imonitorg. The "iotsnoop" will NOT be checking into a cloud server -all data is accessible on the web server on the iotsnoop, and/or via a daily email.

9. The "iotsnoop" pi4 is only meant to be used BEHIND your ISP router, much like a wifi APN you buy to place behind your router. It should not be used as the ISP router! There is no firewalling on the pi4, and it is routing, not NATing to the ISP router. Version 2 etc will probably do NAT and have some firewalling [tho many feel NAT is enough!]. The version 1 is only performing isolation by putting the IOT on a separate network for now.

10. The "iotsnoop" must be connected via ethernet to your ISP router [or a switch behind it] -not wifi -the wifi is used for the IOT APN. It will come up running! and the "iotsnoopg" SSID will be visible [key is "iotsnoop"]. Just find the iotsnoop IP in your router and browse to it -from your home private network- to see the plots/stats [config your email there for daily email]. It should be on your main network if you placed it there. The webpage runs simple HTTP, not HTTPS, but this is not a problem on your home network. Just make an http exception when your browser complains about no https. Then walk your IOT onto the new "iotsnoopg" SSID.

11. Go to https://iotsnoop.com to follow iotsnoop progress, and get the sourceforge link for the raspberry pi4 image to download

12. If you feel challenged about all this, just ask me -I will send you an imaged SDcard and order info for a pi4.

13. Streaming [internet] devices like your smart TVs, and even Dish/DirectTV [because they use Internet for on demand stuff] should not be put on IOT snoop because they will saturate the iotsnoop. I have spent lots of time making the iotsnoop robust in the potential face of this possibility. The iotsnoop essentially uses "tshark" which is a command line version of wireshark to capture packets. Much of my work has been learning tshark and its options!

Version 2 will investigate the following

1. Perform NAT and perhaps some firewalling.

2. Increase performance of IOT packet collection beyond 3Mb/s average [tho WHY you would want your IOT to use your bandwidth is questionable!!]. This will use options on tshark to limit capture at the front end e.g.

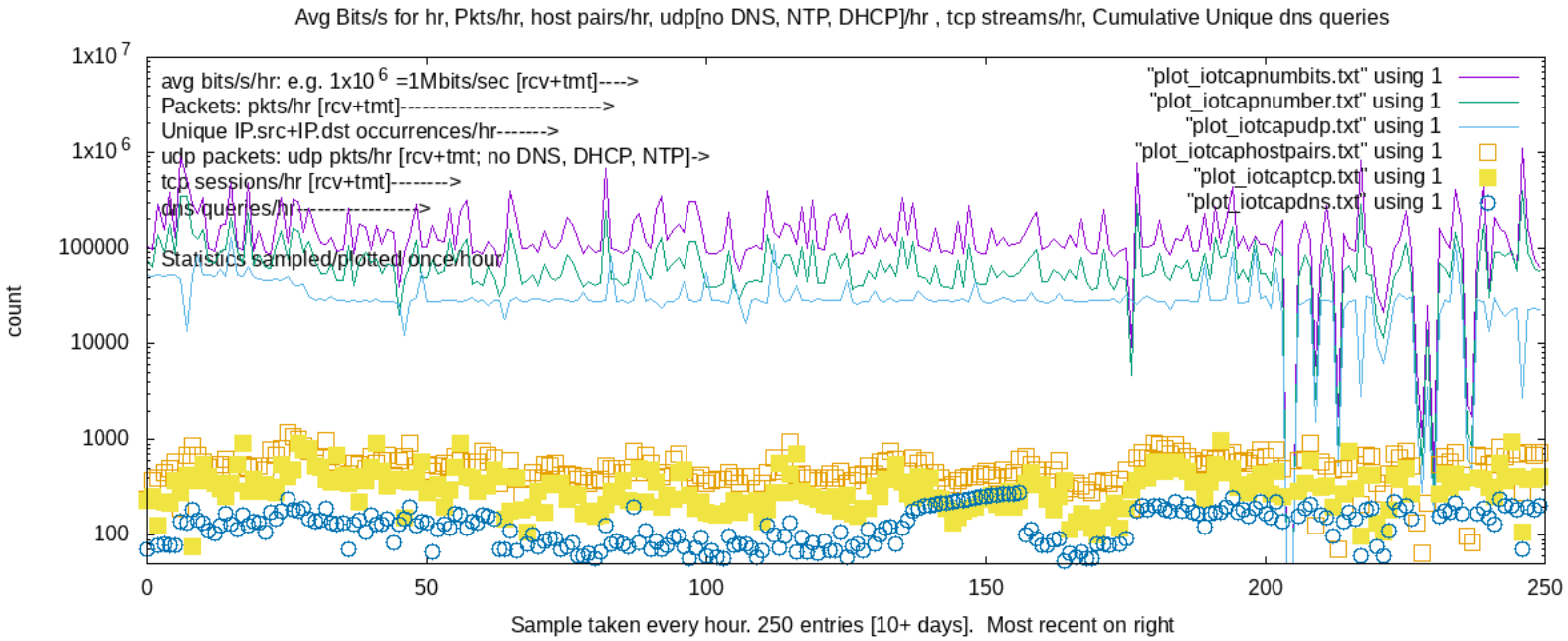
3. There will be an option to set the iotsnoop SSID to your home network, and you can simply change the pi4 SSID and all your IOT will magically appear on the iotsnoop [once you disable the original SSID].

Version 3 may likely investigate implementing an API to query malicious domain services using the DNS queries recorded.

PLEASE let me know of suggestions. I hope your are as excited as me. I am posting the first image to my sourceforge page soon. This will be a normal raspberry pi SDcard. Because of the intensity of the writes, the SDcard should eventually be transitioned to a USB. I will have followup insructions and images to facilitate this.

Needless to say, I am "desperate" for testers for the gadget. Please join our club!

To whet your appetite, I have included three screenshots of a browser pointed to the iotsnoop [all the previous imonitorg web functions are still there as well]. The first shows a rolling log plot showing 250 hours of data which is plotted hourly. You can see details of IOT activity plotted [I will have a page explaining. There is also a "manual" linked on the pi home page]. The second screen shot shows some detail which is gathered from interrogating the packets captured. The third image is a web image of the archive of the IOT activity available in an iframe on the pi4 [collected from the email link].



Tshark environment: MAX Defined by "/root/DEFtshark.txt"

Findiothosts enabled? YES Must be YES for IOT collection
MAX: Full IP packet capture on wlan0 -recommended minimum 8GB pi4
MED: IP packet capture of source IP only on wlan0/TCP-SYN only -recommended minimun 4GB pi4
MIN: IP packet capture of source IP only on wlan0/TCP-SYN only/first 100 bytes -might be OK 1GB pi4

Findiothosts/tshark alert log:

```
Sat 09 Mar 2024 05:04:18 PM EST IOT pkts with Home network src or dst IP
Sat 09 Mar 2024 06:03:22 PM EST IOT pkts with Home network src or dst IP
Sat 09 Mar 2024 07:11:33 PM EST IOT pkts with Home network src or dst IP
WARNING! tshark not running at start of findiothosts: Sat 09 Mar 2024
08:01:02 PM EST LastPkts: 311722
Mar  9 15:50:34 iotsnoop kernel: [200073.310224] oom_reaper: reaped
process 20571 (tshark), now anon-rss:0kB, file-rss:0kB, shmem-rss:0kB
Sat 09 Mar 2024 09:03:08 PM EST IOT pkts with Home network src or dst IP
Sat 09 Mar 2024 10:02:53 PM EST IOT pkts with Home network src or dst IP
Sun 10 Mar 2024 12:02:43 AM EST IOT pkts with Home network src or dst IP
Sun 10 Mar 2024 03:02:48 AM EDT IOT pkts with Home network src or dst IP
```

Detailed iotsnoop Statistics:

[Left]Cumulative -unique- IOT dns queries -> not capped! [Right]Unique queries: Total/LastHour
Ideally the number should not grow over time, except as your IOT gadgets increase.
If the number grows significantly, then there is likely trouble on your IOT network!

```
0.166.168.192.in-addr.arpa
0.datadog.pool.ntp.org
0.debian.pool.ntp.org
0.ubuntu.pool.ntp.org
117.129.31.72.in-addr.arpa
145.50.168.192.in-addr.arpa
181.7.31.72.in-addr.arpa
18577a809395027ddfcf89cf96674a1099486fef49cffd970b6b7220fe431e6.us-east-
1.prod.service.minerva.devices.a2z.com
191.50.168.192.in-addr.arpa
```

```
11:38:AM queries: 594  0
12:01:PM queries: 594  4
13:01:PM queries: 594  1
13:36:PM queries: 596  98
14:01:PM queries: 599  100
15:01:PM queries: 605  132
16:02:PM queries: 623  219
17:01:PM queries: 632  218
18:01:PM queries: 643  206
19:01:PM queries: 649  172
20:01:PM queries: 657  205
```

To clear above, as pi: cd tests/iothosts; cat /dev/null >Longiotcapdns.txt; cat /dev/null >iotcapdnsINFO.txt

[Left]Last hour iot hosts/pkt counts. [Right] Archive of hosts/pkts[both rcv+tmt]/mdns name:
Each Archive dated block represents iot hosts/pkt counts/names for previous hour [rcv+tmt]

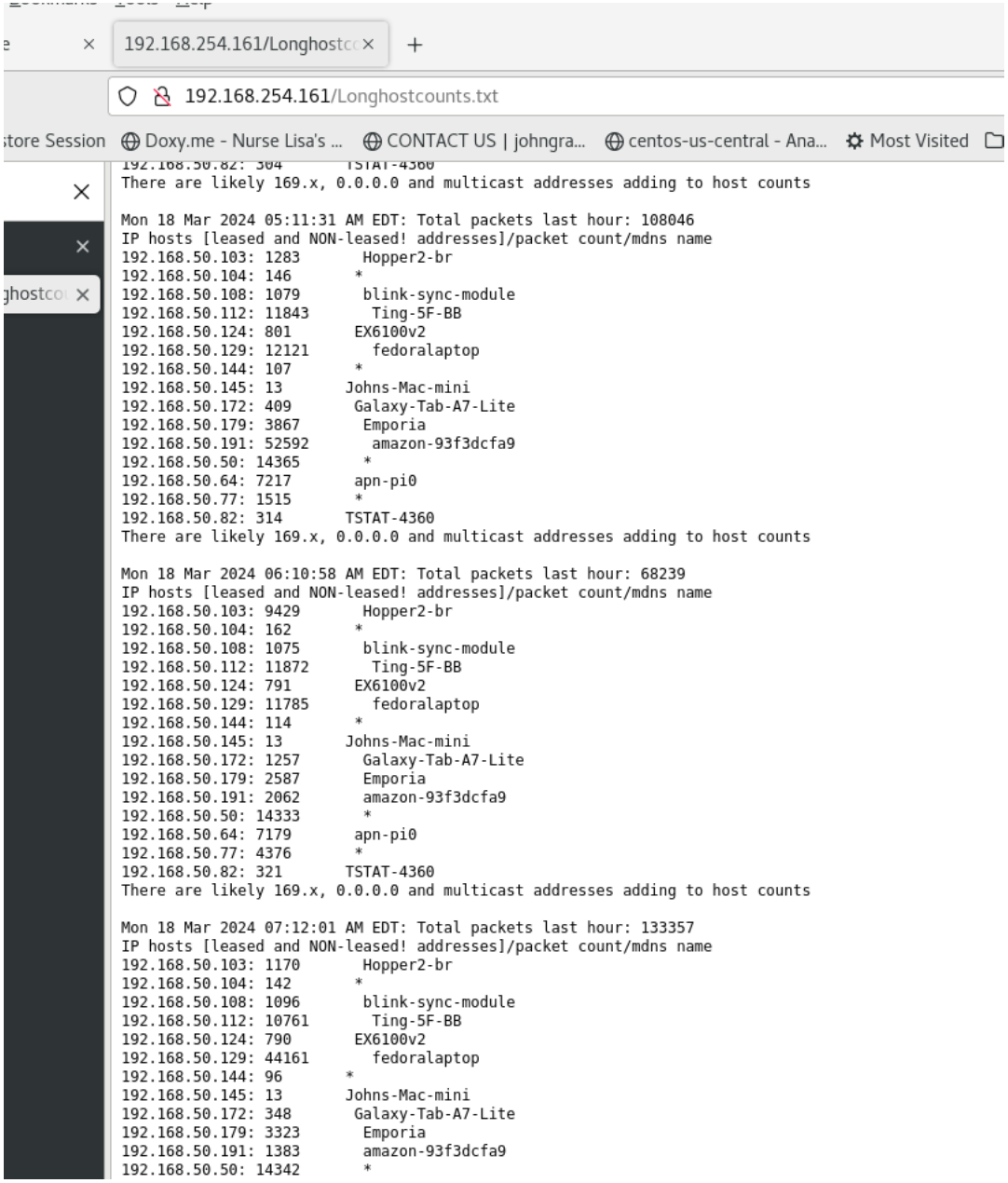
```
09:12: Total packets last
hour: 50827
192.168.50.103: 1746
192.168.50.108: 1066
192.168.50.112: 9829
192.168.50.130: 5169
192.168.50.144: 28
192.168.50.145: 54
192.168.50.158: 4484
192.168.50.163: 603
192.168.50.172: 362
```

```
192.168.50.104: 5969 Hopper2-br
192.168.50.112: 18728 Ting-5F-BB
192.168.50.130: 792 blink-sync-module
192.168.50.132: 3280 fedoralaptop
192.168.50.144: 35 orig-pi0
192.168.50.163: 630 EX6100v2
192.168.50.172: 2252 Galaxy-Tab-A7-Lite
192.168.50.179: 1855 Emporia
192.168.50.193: 1136 Galaxy-A14-5G
192.168.50.64: 38 apn-pi0
192.168.50.67: 46647 imonitorg-ubuntu-VirtualBox
```

To clear above right, as pi: cd tests/iothosts; cat /dev/null >Longhostcounts.txt

IOT DHCP leases assigned by dnsmasq -infer device from mdns name

```
1710077574 48:b4:23:56:1b:81 192.168.50.158 * *
1710077874 cc:9e:a2:21:0a:bc 192.168.50.180 amazon-48f31398d 01:cc:9e:a2:21:0a:bc
1710077957 08:00:27:49:ee:53 192.168.50.67 imonitorg-ubuntu-VirtualBox 01:08:00:27:49:ee:53
1710077981 ba:ac:e3:cd:74:5c 192.168.50.57 Galaxy-A14-5G 01:ba:ac:e3:cd:74:5c
1710078168 10:52:1c:b9:49:40 192.168.50.179 Emporia 01:10:52:1c:b9:49:40
1710078301 b8:27:eb:ab:ef:45 192.168.50.64 apn-pi0 01:b8:27:eb:ab:ef:45
1710078309 b8:5f:98:eb:b3:d9 192.168.50.78 * 01:b8:5f:98:eb:b3:d9
1710078471 b8:27:eb:02:b9:d3 192.168.50.145 orig-pi0 01:b8:27:eb:02:b9:d3
```

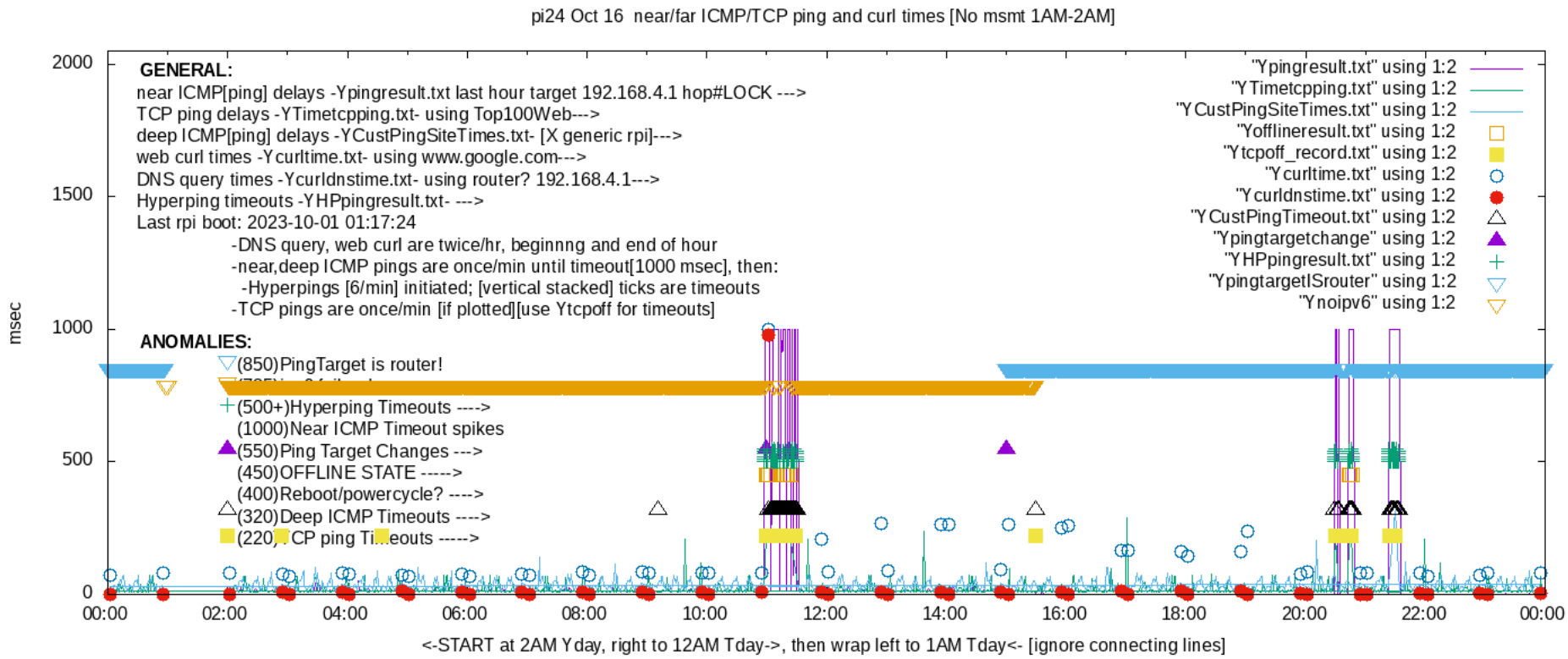


This is the previous November 2023 newsletter for reference:

11-1-2023 Newsletter

At the top I will posit this plot to indicate a fascinating subject of this month's newsletter to tickle your interest. Go to section 7 to see this discussion. If you are interested in this kind of plotting/info gathering for your internet connection, you should consider joining our community, or get a sourceforge raspberrypi image and install your own raspberry pi! Go to https://imonitorg.com to start.

pi24Problem



Several topics to discuss this month/quarter:

1. When you **LOCK** your ping target, the scripts will use router as near ICMP ping target unless you specifically configure a LOCK ping target. Using the router as the ping target compromises the Internet statistics performed, tho it is warranted in some cases. A new, special plot line will perform a "roof" on the plotting, indicating this compromise.

If the ICMP ping target specfication is the normal-default "AUTO," each hour the script automatically determines the near ICMP ping target starting at hop5 and working down to hop1 [your router]. It picks the furthest target pingable. It is also possible to specify the hop to start with because there are complications in the routes occasionally. Some people are so close to the Internet that hop5 is on the other side of the country [exaggerating here, but you get the idea!].

2. Because of the increasing importance of ipv6 to our internet, I have added a plot to indicate, minute-to-minute, when ipv6 is NOT available to the host. This is also a "roofline" plot below the "ping target is router roofline." Almost all OS's these days will try ipv6 connections first, esp if ipv6 is enabled on the host, and there is an ipv6 entry for the website, etc. In almost any case, both ipv4 and ipv6 DNS queries are made for websites.

There are various reasons why ipv6 is not being used by some hosts/networks. There are a few ISPs which do not seem to provide the ipv6. There are pi placements actually behind a second router which is not capable of passing ipv6. These second routers need to be configured [if not by default] to perform the "router advertisement" necessary for the hosts on the network to discover ipv6 gateways. OR they can often be configured in a "transparent" "pass" mode. In any case, it is not clear that ipv6 works cleanly in this case, and there is no universal accepted way of dealing with this.

And of course, several users have disabled ipv6 in their routers. This is not a problem, but going forward I would encourage people to enable it. It is more difficult to disable IPv6 in hosts, and it is almost universally enabled, and your host will almost always display at least an "FE80" IPv6 link local address. And, there is a tremendous amount of ipv6 activity going on behind the scenes!!

A couple things to understand about ipv6. There is no NAT'ing with ipv6 [unless you bend over backward to configure this...]. Each host inside a network has a unique **GLOBAL** ipv6 address, so it is able to connect -outward- to ipv6 hosts on the internet as long as the Internet host has an ipv6 address. The thing that is not universally possible with existing routers yet, is the ability to allow Internet hosts with ipv6 addresses to REACH/connect to ipv6 hosts inside our networks [no pinholing/port mapping for ipv6]. This is an intended behavior, since it "mimicks" the behavior of our existing ipv4 NAT routers, which block ALL unsolicited IP packets from entering our networks [UDP, TCP]. Some recent routers do include the ability to port map [pinhole] TCP/UDP ports for ipv6 packets to connect to hosts inside the network.

And of course if you are not confused enough, when you look at your ipv6 addresses, you may have quite a few! This is a discussion for another day, but the "Internet" ipv6 is the "global" one beginning with 200X. The addresses beginning with "Fe80" are all local to the network and they work independently of whether you have an IPv6 address on the Internet! You will be amazed at the ipv6 traffic on your network, esp the multicast stuff going on.

Just remember that even if you do not enable IPv6 on your router, there is a tremendous amount of IPv6 present on your network with modern OS's.

3. Censys and shodan scans.

Censys.io scans the entire Internet on all ports every few days. Incredible site. It will discover if you have a service running. This is such a wonderful resource. It should be revealing to you. The pi does a scan of your IPv4 every saturday [only the managed pis] using grc.com, but censys.io is doing this full time and circles around to your IP every day or so, and not at the same time! Shodan.io does a similar thing, but is almost old art by this time.

The censys and shodan local links listed in your daily email is perfectly safe to click and see the report for your IP!! The report is actually saved on the pi, so you are browsing to the pi to see the report. You can of course go directly to censys.io at any time!

4. Monitor undervoltages

The pi monitors undervoltages it experiences and reports them in the alert listing in the daily email. The microUSB connector is sadly not a very reliable connector [replaced by the USB C on the p4], and we have found that you actually need to "reseat/wiggle" it sometimes to make it behave!! Running the pi from a strict USB port is usually NOT sufficient since the pi requires up to 1.5A at 5V, and the USB ports [differences between ver 1,2,3] typically provide 1A or less! The temperature of the pi is also reported. It can make it to 70deg C sometimes, so it may need to be repositioned to allow better venting.

5. Off network SYNs

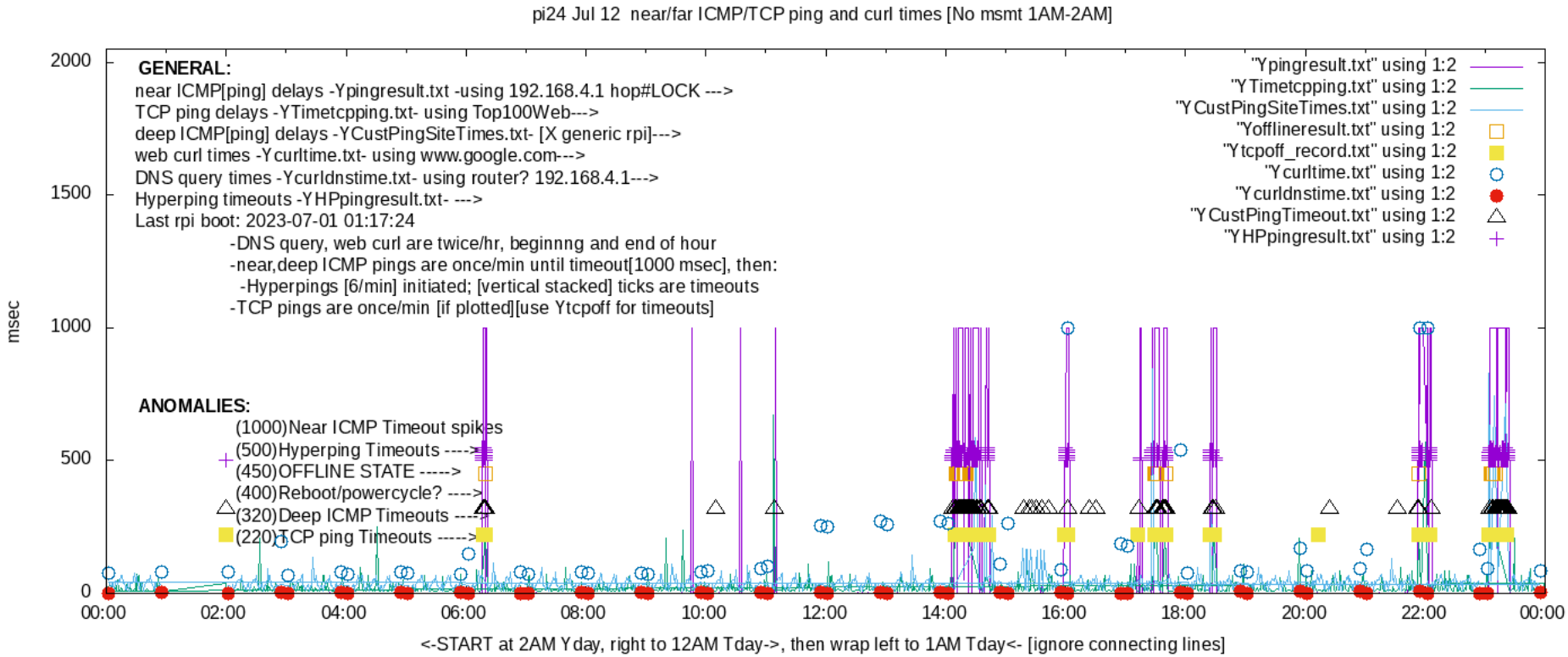
Earlier this year, a script was added to monitor TCP SYNs received from off-network. These should never occur! It would mean that there is a pinhole in your router and external sites are probing the pi, OR there are different networks in your home which are probing the network the pi is on. These are reported in the daily email if they occur. Niel's pi24 network is unique in that it is a /22. The pi scripts only work for /24, so I am only able to do host discovery for the first 255 addresses. All other things are equal. But there will naturally be "off-network" SYNs that may occur. SO I have to turn this detection OFF.

6. Ethernet errors.

Ethernet errors are also reported. It is not entirely clear what is happening with these. Ethernet "drops" will occur if protocols are received that are not able to be processed by the pi, such as vlan, bpdu, cdp, netware, apple, etc packets. These may be normal in some cases. I am continuing to evaluate the errored/drop indications.

7. pi24 Case Study -this is really interesting and confounding!!

About 1 year ago, Niel on pi24 was able to switch from Tmobile cellular broadband to a new fiber optic service by Ting. His performance was extraordinary for about a year, showing near perfect 300Mbs/300Mbs performance using the rpi to measure [the rpi cannnot reveal the potential of 1Gbs links. The mobile/dedicated speedtest apps frequently show close to 1Gbs]. Niel's home network is extraordinarily complex for a home network, using multiple eero routers to establish a wireless mesh thruout his "estate." He has dozens and dozens of IOT devices from many different vendors on a single flat network spread thruout his house and grounds, even going so far as to use a /22 network mask [apparently the eero default]. Earlier this year, his access began showing random events characterizing themselves by spotty ping performance for minutes and hours at a time, and out-of-service events during those [as recorded by the pi and its plots]. These happened maybe 1,2 times per week. The pi was also showing an occasional random loss of the ipv6 address [and its gateway]. ...maybe a clue? Here is a plot revealing the characteristics of the pi plotting for a day when it happened: July 11 [plotted July 12]: This performance is not debillating, but at certain times there is definitely congestion and timeouts experienced for some reason! This was happening maybe once a week. Most days the plot was clean as a whistle!



Niel established that the root cause appeared to be somewhere inside his network because, at the time of the problem, he was able to go to the hub/root eero connected to the Ting fiber modem and establish connectivity. He would disconnect/reconnect/power cycle intervening ethernet switches [even swapping cables/plugging in/out] to reclaim his LAN access. He even switched his eero boxes at one time, placing a different one at the hub/root. Eero is supposed to handle all the reconfiguration necessary in this instance [Internet gateway switches hubs!]. None of these attempts seem to have changed anything, as the symptom simply reappeared, always seemingly at random. Maybe once or twice a week. We were not able to correlate the events with anything. The immediate/obvious pi symptom was that it would fail the ping of the ping target***.*** Following this sympton, we switched the ICMP ping target ***to the router*** and LOCKed it [the normal ICMP ping target is calculated automatically -AUTO mode- and is usually several hops into the Internet]. The characteristic symptom on the pi, besides the occasional loss of the ipv6 address/gateway was always this "disappearance" of the router, i.e. the ping timeouts to the router. It was as if the router dropped off the network -it did not respond to ICMP pings within a 1 second window [the timeout value].

So we decided to disable ipv6 on the eero router, thinking maybe its algorithm to rejigger the ipv6 networking after the eero switch was not performing correctly. ***Did he have to actually have to uninstall/disable/power cycle to reconfig ipv6?***

Following this switch/power cycle, his access performed nearly flawless for a couple of months. There were a couple glitches with different characteristics which I recorded [ping to the router was not lost, it occurred further in the Internet] We swithced the ping back to "AUTO" which forced it 5 hops into the Internet... So.... problem fixed???

However, on October 15, there was a major outage lasting nearly an hour at 11AM in the middle of the day which showed the same earlier symptom. So what to do? This is the plot at the top of the page. Niel especially noticed this because he was on zoom calls which he had to reconfigure for his phone hotspot.

We started contemplating what could actually "make the router disappear." Was there a broadcast flood in his switches.. was there a network routing problem with hosts confusing /24 and /22 subnet masks. Was there even a miscreant who got on his network and was performing bot duty under command to DOS/DDOS [denial of service] targets. A single compromised host could tie up the 1Gbs ethernet with such DOS malware! This doesn't seem likely because individual bots would hide themselves by only by using small amounts of bandwidth individually, and using the power of thousands of bots to do the DOS/DDOS damage. Niel was usually able to recover by disconnecting/power cycling a switch which would seem to indicate a stuck/bad/compromised/faulty UDP/TCP connection was interrupted when he did this [interface going down tears down/timesout these connections]. But who/what was doing the damage? Niels' network is flat, and the IOT devices are on the same network as his important hosts.

I have included a plot [at the top of the page] which shows the initiation of this event on Oct 15 [the plot is labelled Oct 16, when it was collected for the previous day]. You can see it about 11AM and lasting for about an hour. On this plot you can see that the ipv6 was nonfunctioning [at least on the pi24] before this. You can also see on the plot when I turned on the ping to the router [LOCK] -the "roofline.". You can then see when Niel decided to turn ipv6 back on. Those two "rooflines" on the plot indicate these "compromising" features.

You can then see what happened at 8:30PM and at 9:30PM. Two more events occurred, not quite as debilitating as the one at 11AM. The worst indication is the "offline" declaration on the plot, signified by the square box underneath the ticks upward. The ticks upward represent ICMP ping failures to the router [ON the network] within a 1 minute interval. Once a ping failure is detected [and noted by the peaks], the ICMP pings are attempted 6 times/minute and failures noted by the ticks. Additional failures of the TCP pings and the customer ICMP pings are noted by the colored boxes and the upward pointing triangles to further validate a serious connectivity event. "Offline" is entirely a subjective declaration of course. HOW do you declare an offline in a statistically multiplexed network. My algorithm declares "offline" and plots "it" by detecting a near ICMP ping failure simultaneously with a tcp ping failure. If they cross in a minute, I declare an "offline." Purely subjective.

As of Oct 17, we still have little clue as to what is actually going on. The events are very random and appear to affect his LAN side network, and the characteristic is just that the router refuses to respond to a ping [or... it times out after a minute which is "not responding" to my algorithm], and thus is very likely not responding to any attempt to pass packets thru it. This is validated by Niel's failed attempts at browsing/etc when this occurs. Another clue is that the pi24 actually occasionally loses ipv6 connectivity, and specifically the ipv6 default route. This does not appear correlated, but it has happened several times during these troubles. It is as if the eero ipv6 "router advertisements" disappear -the pi will pick them up automatically if they appear. So the pi appears to be accurately representing what is happening. But there is little clue as to the cause. The loss of the ipv6 by the pi24 has not happened since Oct 16 yet..... after Niel reenabled the ipv6.

We welcome our intelligent users comments as to what may be occurring in Niel's network!! It would be nice if he could separate his IOT onto a separate guest network to give some protection from any potential compromised IOT devices, but this does not appear to be what is happening. The great risk of having all the IOT devices on your network with your important hosts is of course the danger of the cloud servers getting compromised. Almost all IOT devices have a "phone home" built in for one reason or another. If the "home" gets compromised it is trivial for the "home" to connect back to the IOT device and compromise its purpose, injecting malicious codes. The most obvious use is that of a "bot" which is just an IOT device turned into a slave to a malicious command and control server. [see the following section]

Going forward, Niel will be running a ping utility at another host to verify the "disappearance" of the router. This wil help locate/identify the root sympton.

8. "Credential stuffing" john's imonitorg.com mail server -another fascinating discussion!

Several months ago I noticed -increased- repeated attempts to login to my mail server's imaps "submission" port 993. These were occurring basically every few minutes and from random IP addresses. They were occurring to all the "normal" users, info, admin, user, etc. I eventually noticed they were attempting to login to my actual user.... So they had harvested my email address from the web and were adding it to the attempts. So how good was my password!!?? Some time before this I had added "fail2ban" to my linux centos8 stream services which monitors the mail logs [and the httpd and other logs] for repeated malicious attempts on mail servers and web servers [and esp ssh servers]. I had spent months fine tuning the fail2ban service by increasing the sensitivity of the config settings. You can configure the number of allowed failed attempts, the frequency attempts [even configuring the window]. I eventually had this set at just TWO failed attempts in an hour allowed. I would then ban the IP address for 16 hours [fail2ban does this by manipulating the built in firewall]. But even these rigorous criteria would not defeat what is going on. I noticed the bots were actually rotating the IP addresses every minute. In other words, a single bot was not doing the credential stuffing to the same user ID! It was as if the bots were synchronized amongst themselves and performed only a SINGLE login attempt, thus NOT getting banned. The next IP bot would then attempt the same attempt within the window and not get banned.

So I spent a day developing some bash scripts [I am getting better and better at this!] to scrape the IP from the logs and then determine the source of the IP using the "whois" command. I was able to determine that ***65 countries*** at last count were doing this dance. There is no way that fail2ban can counter this, even banning for[ever] a single failed attempt. It really demonstrates the severity of the cyber war. For a normal mail server sitting on the Internet with hundreds of users, it is a very complex task to monitor and counter this scale of "credential stuffing." In my case, since I am basically the only one using the mail server, I simply set the firewall to block all IPs except my home IP -both ipv4 and ipv6. Real mail servers would have to do much more complex monitoring. You can imagine what gmail, hotmail, etc are doing in their email servers!!!

Just for grins, here is the list of the countries [2 letter codes scraped from the "whois" for the IP addresses scraped from the maillogs] credential stuffing my mailserver. These almost certainly represent bots installed on IOTs/routers/etc in all these countries. I should do more scripting and actually count the country instances for more info! This is a "uniq" list.

```
fail2ban imap login failure countries
country:        AE
country:     AR
country:        AT
country:        AU
country:        BD
country:        BG
country:     BO
country:        CH
country:     CL
country:        cn
```

```
country:        CN
country:     CO
country:     CR
country:        DE
country:     DO
country:        ES
country:        EU
country:        EU # Country is really world wide
country:        FR
country:        GB
country:        GE
country:        hk
country:        HK
country:        ID
country:        IL
country:        IN
country:        IQ
country:        IR
country:        IT
country:        JP
country:        KH
country:        KR
country:        KW
country:        KZ
country:        lb
country:        MA
country:        MK
country:     MX
country:        NG
country:     NI
country:     PE
country:        PH
country:        PK
country:        PL
country:        PT
country:        RO
country:        ru
country:        RU
country:        SA
country:        SE
country:        SG
country:        TH
country:        TN
country:        TW
country:        TZ
country:        UA
country:     US
country:        UZ
country:        VN
country:        ZA
country:        ZZ
```

This is truly mind blowing as to the cyberwar occurring in cyberspace.

9. ipv6 summary of rpi clients as of October 2023

Here are the IPv6 capabilities of the current managed subscribers.

pi1,5,10,15,20 ipv6 Spectrum cable ---IPv6 is enabled

pi3 ipv6 OK Comcast cable Denver --IPv6 OK

pi6 ipv6 pi is double NAT'd and the pi doesn't see IPv6

pi7 ipv6 pi double NAT'd ATT VDSL --no IPv6

pi8 Conexon fiber --IPv6 OK

pi9 is double NAT'd ATT firstnet cellular --no IPv6

pi11 Spectrum cable --IPv6 WAS OK, but not functioning right now. ???

pi12 ATT VDSL --ipv6 OK

ipv6 OKpi13 ATT VDSL --

pi14 Comcast cable or ATT firstnet --ipv6 OK

pi21 double NAT'd Spectrum cable --no ipv6

pi22 ATT fiber --ipv6 OK

pi23 ATT fiber --ipv6 OK

pi24 Ting fiber debugging --ipv6 WAS OK, but is not working reliably.

pi25 Comcast cable --ipv6 OK

pi26 Comcast cable -- No ipv6 ISP??

pi27 custom router config wo IPv6 Longmont fiber --no ipv6

pi28 ATT fiber --ipv6 OK

pi29 Vistabeam WADSL --no IPv6

pi31 Frontier WADSL --no IPv6

pi32 Community fiber Estes Park --No ipv6

New sourceforge images

pi8 is fiber via conexcon, pi28 is fiber via ATT. I no longer have a starlink customer :(Starlink getting expensive! cellular broadband $50/month

10. New sourceforge images

I have new sourceforge images dated 9-1-2023. I will have updated these by 11-1-2023 hopefully, so you may want to wait and watch. Contact me if you want more info, and maybe even a free raspberry pi. I may have one or two extra.

Previous 5-1-2023 newsletter

1. I have been working on creating a VM [virtual machine] image of the imonitorg function. This would be ideal for PCs or servers which are running 24/7 -you would not need a separate raspberry pi 3b or 3b+ device to perform the monitorg function! I have transferred the imonitor[g] scripts from the raspberry pi to an ubuntu OS [LTS 22.04] running in a VM. I then create an image of that VM - an "open virtual machine architecture [ova]" image which should be importable by VM hypervisors such as Oracle "VirtualBox," vmware "VMware," linux "boxes," Apple "parallels," and windows "Hyper-V." There are undoubtedly other hypervisors. I have tested the VM "ova" using Oracle "VirtualBox" on windows 10, rocky8 linux, and centos8 linux.

I am looking forward to testing the "ova" on other hypervisors, specifically vmware[runs on MACos], linux "boxes," and windows "hyper-V" in the coming months. "Parallels" is the MACos hypervisor, but they charge money, so I will probably not be testing this. I have a MAC M1 that I hope to install vmware on and test this. I would love any feedback, beta testers on this.

If you are not familiar with hypervisors, they are a very convenient way to run windows on your MAC, or linux on your windows, or windows on your linux, etc. etc. You just need the base "hypervisor" installed to manage it [plus the "virtualization" enable in the BIOS]. The hypervisor typically will list OS images [VMs in "ova" format] to import, or you can import your own, as we are doing here. The VM is bridged to the host network, to appear as just another PC/device on your network. It is a wonderful way to utilize multiple OS's, and is really superior to "dual boot" and similar techniques. Perhaps you have applications that only run on windows, but you are a "MAC" person. Just install "Parallels" and import the windows VM.

Most imonitor[g] scripts are transferred to the VM, including pihole. Since the VM appears "virtually" on the host network, it appears behind the network adapter of the host, which can be ethernet or wifi. So "wifi or "ethernet" is transparent to the VM. The wifi testing/scanning scripts are thus removed.

The first VM ova "vbox_imonitorg9.ova" is available at my sourceforge project page https://sourceforge.net/projects/imonitorg/files/

2. Tom -pi29- and I have been learning about "Carrier grade NAT [CGNAT]." Several ISPs implement their Internet access in this way. It has become increasingly apparent that this does not provide an "ID" on the Internet in the same manner as other ISPs -where "sticky" or "permanent" IP addresses are almost like real street addresses. pi29's "Internet IP" can change multiple times per hour at times. The CGNAT ISP manages the dance of the IP addresses so it can maintain TCP/UDP connectivity, much like cell phones can roam and stay connected. A CGNAT will thus not give you a "presence" on the Internet in the "traditional" way. You cannot run a service on a CGNAT connection, and associate a DNS domain to it. There are ways around this using VPNs and cloud servers of course. I will have to modify my script for CGNAT, because "your IP" on the Internet just doesn't really exist. The technology "TCP handoff" to manage this "dance" was invented for cellular phones that were IP.

I have "customers" like pi13 who have had the same Internet IP for 5 years. My Internet IP has not changed in 3 years -it only seems to change when huge storms rumble thru and send all of the local Spectrum network into a reboot. I would not doubt that this will likely change... no reason they can't save the IP to be sticky. There are many reasons why ISPs like your IP to be "sticky," not the least of course is that you can be ID'd by your IP, and this info can be sold. This is less and less useful with default DNS over https used in browsers [DOH]. In the old days you just rebooted you router to change your IP, but those days seem to be gone.

"StarLink" officially uses CGNAT, but the address change is far less often, at least in the case of pi8 and [former] pi9. You can even pay more money and get a "real" ipv4 address. It is not quite clear why Starlink even does CGNAT, since it is only really used on ipv4 networks to preserve ipv4 address space. Starlink fully supports ipv6, so I suspect it is a transition issue. If a device has an ipv6 address, there is no reason to perform NAT of any type. .....of course there are legacy apps which only do ipv4, so the need for ipv4 will probably never go away.

Cellular networks, such as pi9 and pi14 also use CGNAT and most do ipv6, so there can be very confusing issues in trying to understand what they are doing. Ernie on pi9 [ATT firstnet] has only changed IP addresses once in a month since he has been on ATT firstnet. Matt on pi14 [ATT cellular broadband] has also not changed IP addresses.

In short if you are on a CGNAT-ISP network or a mobile network, you really do not have an "address" on the Internet similar to if you have a cable/fiber/ADSL -wired- access.

3. Cellular Broadband: "Cellular broadband" is appearing more and more. Tmobile may have been the first - Niel pi24 outside Raleigh NC used this for a year. It resembles voice cellular in its delay and congestion as can be seen

by the response graphs. Spectrum provides this almost as a backup service now. I have it on a tablet in addition to regular Spectrum cable -this is undoubtedly Verizon broadband, since Spectrum uses Verizon here in FL. Ernie pi9 switched from Starlink to ATT firstnet, which is a cellular broadband. Matt -pi14 is testing ATT cellular broadband in N GA.

The performance of cellular broadband on these subs is a very good alternative to ADSL, and even the lower echelons of cable speed. It now offers unlimited data and download speeds maybe 50Mbs down. The only negatives are the congestion and delay experienced, since they are in effect competing for voice on the physical airwaves. And of course remember, these technologies are all "in the air" like wifi, so they are not secure like fiber or wires.

The other "air-only" access pis are Tom's WADSL pi29 and 31 in N CO. WADSL is like ADSL except over the air using line of site transmitters/receivers -Tom has an antenna pointed at the server! Performance specs are similar to ADSL. The unique aspect of Tom's pi29 is the CGNAT provisioning.

4. Fiber access: It is amazing how fiber access is accelerating. I can remember working on fiber broadband sysems at Bellcore where this was a dream. This was in 1986 or so. On our imonitor trial, pi22, 23, 24, 27 and 32 are fiber connections now [ATT, Ting, Longmont, Estes Park]. The performance of these access technologies is amazing -just look at their performance plots. https://imonitorg.com/customerplots/rtcustomerplots -click on the first link up there. Access speed is "normally" about 1Gbs downspeed, and is very hard to "measure."

5. Cable access: pi11 is cable access technologies with lower performance [lower cost]. pi 1,3,6.5,10,15,20,21,25,26,16 are higher peformance cable [Comcast, Spectrum, TDS], often approaching 1Gbs downspeed. I remember the telco [ADSL] guy fearing cable would overtake them, and it pretty much has! Centurylink, which is mostly telco wires, is still around as a second choice in many of these areas [like mine].

6. ADSL access: Phil on pi7,12 and 13 and Kevin on pi28 are the last "pioneers" using the phone line pair -ADSL -mostly VDSL technologies. It is amazing that we lived -almost- thru the demise of the phone line pair as an access technology. It was such a truly marvelous step forward when simple ADSL came out about 1995 with 1.5Mbs down, shared with your voice line. Almost 30 years ago!

7. Speedtests: it is increasingly misleading to measure the speed of these connections. It depends so much on the client and the network the ISP puts in place to measure your speed. Just how do you measure speed on a statisically multiplexed connection? If two packets are back to back in the medium, do you declare the speed as "line speed." There are so many algorithms for doing this that it is another study. I like to use "Speed.cloudflare.com" which shows a time graph of your speed. I am sure the speed it "reports" is probably the PEAK it sees.

8. IPv6: Routers allow outgoing IPv6 [and the corresponding return packets], but defaultly block unsolicited incoming IPv6 TCP connections. There are very few routers which allow mapping ports on IPv6 addresses. And don't dare do a DNS entry for your ipv6 address if you can't forward the port. ...you can do all this using ipv4 almost universally. Very confusing to sort out the alternatives!!

I am doing inventory on my pis WRT ipv6 and making sure I understand the situation where it is provided/not provided and when it is disabled/not provided. Several pis appear as a second stage NAT on the home network, and the second stage router is not doing ipv6 passthru so the pi can see the ipv6. Several people have ipv6 turned off, tho this should not be necessary. Almost all applications these days, such as thunderbird and your browser will try ipv6 connections first if they get ipv6 addresses for names. This should "always" work, but if you get delays initially, it is because the ipv6 is timing out [several minutes worth] and will eventually flip to ipv4. I am troubleshooting a problem on my thunderbird where this happens, tho ipv6 is functioning on my connection. Go to "testipv6.com" in a browser, e.g. to get a report on your ipv6 capabilty.

9. I recently added a script to the pis to monitor for "off-network" unsolicited TCP connections to the pi. These should never happen unless you have a port mapped to the pi in your router. They will be triggered e.g. if you switch networks in your home and the pi has not adjusted to the new IP, so it should be transitory.

This is a link to the imonitorg.com homepage, which has a listing of all previous newsletters. https://imonitorg.com

I ardently hope for corrections and discussions of any of the topics above. Thank you.

John Loop 4-20-2023

--