2025 36th International Teletraffic Congress (ITC 36)

On The Evolution of Home Gateways

Tim Neubert
University of Augsburg
Augsburg, Germany
tim.neubert@uni-a.de

Rolf Winter
TH Augsburg
Augsburg, Germany
rolf.winter@tha.de

Jonas Winkler
TH Augsburg
Augsburg, Germany
jonas.winker@tha.de

***Abstract*—Most commonly, residential broadband customers connect to the internet via a home gateway. These devices are often highly capable and feature-rich, implementing essential functions such as Network Address Translation (NAT), while also integrating a WiFi access point, multiple switch ports, and a range of upper-layer features such as parental controls or a built-in media server. Since we are all behind such devices most of the time, they really shape the way we experience the internet. And for that, the most basic services, such as DNS forwarding or Network Address Translation, must work well. But do they?**

About 15 years ago, a measurement study [1] characterized a large number of such devices at the time. But in the last 15 years, a lot has happened. This paper has another look at home gateway characteristics and reproduces those findings to see whether things have improved since. It relies on a diverse router population with devices released over the course of the last 20 years and also includes devices that are still included as part of residential broadband packages of ISPs today.

***Index Terms*—Home Gateways, CPEs, Behavior, Characteristics, Measurements.**

I. Introduction

A home gateway is a special type of customer premises equipment (CPE). It primarily serves as the WiFi access point and Ethernet switch in our homes and connects us as a router to the public Internet via fiber, DSL or cable. These devices can vary significantly in feature set, hardware specifications, and ultimately in price, but their main function is to connect us to the public internet, with everything else coming as an added bonus.

They are special in a number of ways. For one, most customers simply stick to the home gateway that was given to them by their ISP. In fact, customers sometimes have no other choice, in particular if they are not tech-savvy. Also, they might not receive regular updates, if any at all, during their lifetime. Their hardware is also not upgradable, and they usually just sit somewhere in a corner of our homes and perform their function unaltered but dutifully for many years.

Besides shaping the way we experience the internet, these home gateways - a prime example of a type of middlebox - also have a significant impact on the evolution of the internet. In particular, their role as a Network Address Translator requires home gateways to understand protocols above IP in order to create suitable entries in their NAT tables and to fully translate all packets. This, e.g., is one reason why QUIC - the transport protocol developed for HTTP/3 - is using UDP as a substrate instead of running directly over IP because UDP is understood by even the oldest NAT routers. This makes QUIC instantly deployable but also departs from the traditional protocol layering, where transport protocols run directly on top of IP.

To the best of our knowledge, the last time these home gateways have been thoroughly characterized was back in 2010 [1]. In the 15 years since, a lot of change happened on the internet. E.g., HTTP/3 did not exist. In fact, HTTP/3's predecessor HTTP/2 was only finally specified in 2015. In 2010, the first iPad was released, and Blackberry sales were still on a steady rise. On the internet, 15 years come with transformational changes. Given their importance and the technological progress since, it seems about time to reevaluate the behavior of home gateways and to reproduce the findings in [1].

II. Related Work

The study we are reproducing [1] measured aspects such as NAT binding timeouts, the maximum number of simultaneous TCP bindings or ICMP behavior. These are still important characteristics today and have an impact not just on the user-perceived quality of the internet service, but also on the way applications should use the network. It could answer questions such as what an appropriate number of parallel TCP connections would be or how long a TCP connection can remain silent before a packet should be sent, just to keep potential NAT binding state alive. These characteristics are also important for internet protocol development. For example, they were relevant during the development of QUIC [2], as QUIC relies on UDP which has significantly shorter NAT binding timeouts compared to TCP. The study conducted these measurements on a range of home gateways back in 2010 [1]. As such, all the related work found in [1] applies. But home gateways have since also received some attention for other purposes and for other reasons.

For example, home gateways have been identified as one of the major sources of end-to-end latency due to excessively large buffers, which was coined "bufferbloat" [3]. To combat bufferbloat, a number of AQM schemes have been specified such as Codel [4] or PIE [5] and have also been evaluated extensively (e.g. in [6]). The original study did contain a

This work has been partially funded by the German Federal Ministry for Economic Affairs and Climate as part of the MAVERIC project.

978-3-948377-03-8/19/$31.00 ©2025 ITC

bufferbloat analysis, but the methodology and tooling to measure bufferbloat has since evolved quite a bit.

Home gateways have also been used to actually perform measurements from those devices, not to measure these devices themselves. Most prominently, they have been used to measure access network characteristics [7]. These measurements are often an important input for regulatory bodies, many of which regularly publish their results, such as the FCC in their Measuring Broadband America initiative. But also the other side of the home gateway, the home wireless network, has been analyzed [8] and they have been used to analyze application performance by looking at the traffic that they forward [9].

Home gateways are clearly a type of middlebox, in particular because of their use of Network Address Translation. Middleboxes do play an important role on today's internet, and understanding their behavior and identifying them has been the focus of various studies such as [10], [11] or [12]. But to the best of our knowledge, a dedicated study on home gateways using a number of these devices, treating them as black boxes and measuring them broadly in a tightly controlled environment has not been done since 2010. For very specific questions, we are aware of measurements using home gateways, e.g., the one described in [13]. Here a new extension to ICMP is proposed, and the authors evaluated which actual choice of ICMP code points will increase the likelihood of deployment on the internet. For this, they sent a number of different ICMP messages through numerous home gateways to see which messages are delivered successfully and justify their choice of code point accordingly. But these are not anywhere close to being as broad as the ones described in this paper, and are not as immediately relevant for users and applications today. The closest study that has been done since is a crowdsourced study to characterize NATs [14], with a focus on their mapping and filtering behavior and their compliance with IETF recommendations. Because the measurement was crowdsourced, it was able to capture a lot of devices. However, as it was not done not a controlled environment, not all behavior can be safely attributed to the NATs, and not all measurements presented in this paper could be done in [14].

III. Experimental Methodology

A. Lab setup

We reconstructed the experimental setup as described in [1] as closely as possible with certain differences, either necessary or irrelevant to the experimental results. For example, we use a slightly different convention for assigning VLAN IDs and IP ranges and our switching infrastructure uses 1 Gbit/s interfaces instead of the 100 Mbit/s interfaces used in [1]. The former should be irrelevant, but the latter necessary, as modern home gateways these days come with faster switch ports. We also operate our own DNS server in order to have better control over experiments involving DNS.

The authors of [1] unfortunately never published the code which was used for the measurements and for the coordination of those measurements. We made architectural changes by having a central test manager which communicates with test coordinators running locally on the test server which sits on the WAN side of the home gateways and on the test client which is located on the LAN side. The home gateways are all isolated from each other using VLANs, and the management traffic is physically isolated in its own network to not interfere with the tests. The whole test setup is depicted in Fig. 1.

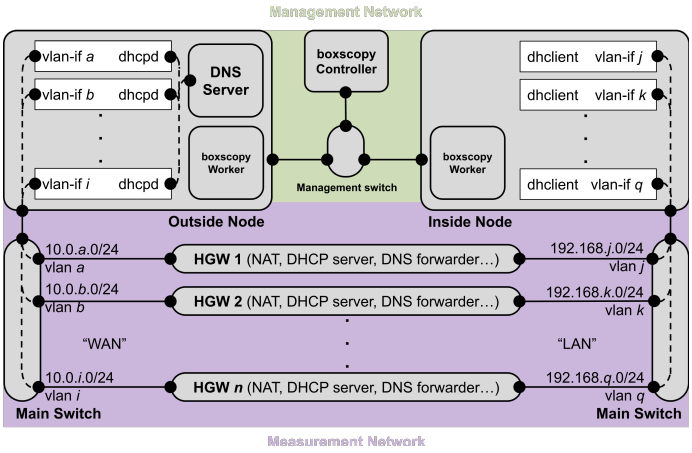

Fig. 1. Setup of the experimental testbed reproducing the one used in [1]

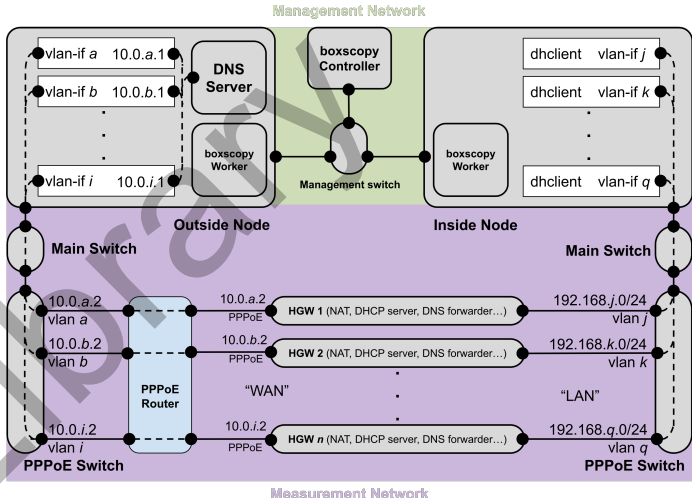

Fig. 2. Lab setup for devices requiring PPPoE

Also, some of the tests were not sufficiently specified to know exactly how they were performed - although most were. We believe we have accurately re-implemented the ones where some details were lacking based on the results we have seen in our lab experiments.

A difference to the original lab setup, of course, is the population of the tested home gateways. Part of our population are devices that were produced around the time of the original paper, but also contains devices from the whole period leading to the present and current models which are still being sold or shipped by ISPs today, covering the device evolution of this whole period. Some vendors from that time have disappeared or stopped building home gateways. For example, A-Link

seems to have disappeared and Apple has stopped producing its Airport product line in 2018. The overall home gateway population only has one device in common: the D-Link DIR-600. However, our model is a hardware revision that runs a later firmware version. While it is unfortunate that we could not validate our results with the ones from the original study by running them against the same device, this still is quite interesting as we can see if there are observable improvements from hardware revisions and/or firmware upgrades. The home gateways we tested can be found in Table V at the end of this paper.

One final difference to our lab setup is that we also included devices that are genuinely provided by ISPs to their customers as part of their residential broadband internet offering. This complicates the lab setup, as we needed to reverse engineer a few parameters that these home gateways used or expected from the network in order to function properly. For these devices, we needed to also add a PPPoE server to mimic the providers BNG (Broadband Network Gateway) to hand out addresses on the WAN-side of the home gateways. The slightly altered test setup for these devices can be seen in Fig. 2.

IV. Measurement Results

We have reproduced the measurements conducted in [1]. In this section, we not only describe those measurements but also present the results we obtained with our router population. We also refer to the original paper's name for the measurement (e.g. TCP-1), so that a direct comparison and more background information can be obtained from [1] if necessary.

A. TCP

1) TCP Timeout (TCP-1): Fig. 3 presents the measured NAT binding timeouts for TCP, i.e. the time a NAT table entry for an established but otherwise unused TCP connection will remain in the table before it is being removed. Prematurely removing an entry will result in packets arriving at the NAT via the WAN interface for that particular TCP connection being dropped, rendering the connection unusable.

In our case, only four devices meet the requirement stated in RFC 5382 that a NAT must keep the binding alive for at least 2 hours and 4 minutes (7440 seconds) [15], with only a single device using that exact value. The rationale behind this requirement is to enable applications to protect the NAT binding using the TCP keep-alive option, which sends keep-alive packets every 7200 seconds. One gateway unfortunately sets its binding timeout to exactly 7200 seconds, which, in combination with latencies, might hinder applications from successfully using the keep-alive option. The remaining 18 home gateways all stay well below the recommended timeout values, with one device even going as low as one minute. The default TCP keep-alive mechanism would therefore not reliably work here, and the end hosts would need to tweak their TCP settings. Interestingly, these timeout values do not correlate with the age of the home gateways, suggesting that TCP NAT binding timeouts have not changed significantly over time.

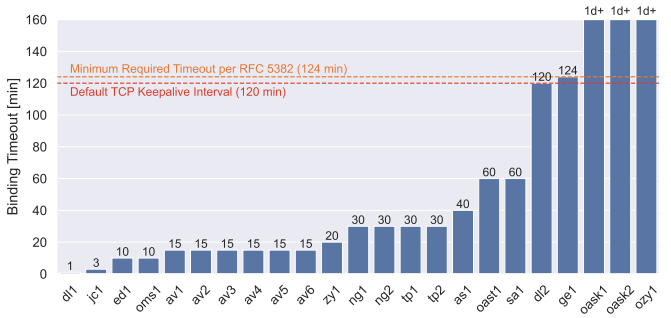

Fig. 3. TCP Timeout for established connection with completed 3-way handshake

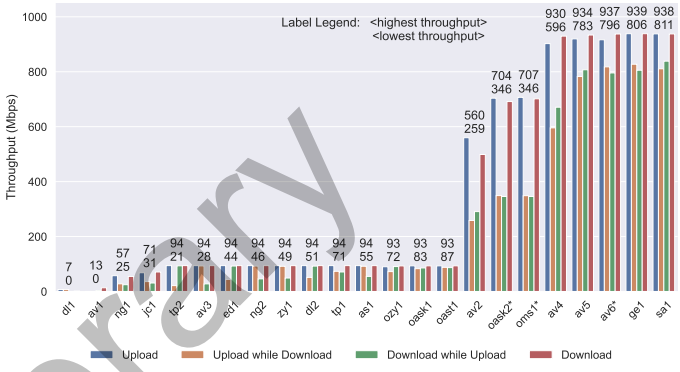

Fig. 4. Results of the TCP-2 measurement. The bar labels display the minimum and maximum throughput measures across all types of traffic of each device. Device tags with an asterisk are (likely) limited by the testbed setup

2) Throughput and latency (TCP-2 and TCP-3): From the original study, the TCP-2 and TCP-3 experiments measured the throughput and latency, respectively. Three scenarios were evaluated, each varying the direction of the data transfer: From WAN to LAN (download), from LAN to WAN (upload) and finally both directions simultaneously to evaluate whether traffic in one direction affects performance in the other direction or not.

This particular test did not need to be implemented from scratch, as we could rely on established tools. We used the netperf [16] benchmarking utility for this measurement, as it provides both latency and throughput measurements within a single test.

Figures 4 and 5 show the results of the TCP-2 and TCP-3 measurements, respectively. The results of both measurements reveal distinct groups of devices.

For 11 out of the 23 devices, the 100 Mbit/s Ethernet interfaces clearly limit the throughput. The five best-performing devices fully utilize their 1 Gbit/s Ethernet interfaces and are among the newest in the tested population. On the lower end, the two oldest devices, *av1* and *dl1*, appear to struggle under the measurements, offering only very low speeds, with *dl1* even dropping below 1 Mbit/s in the download test. While most of the devices achieve full interface rate for unidirectional

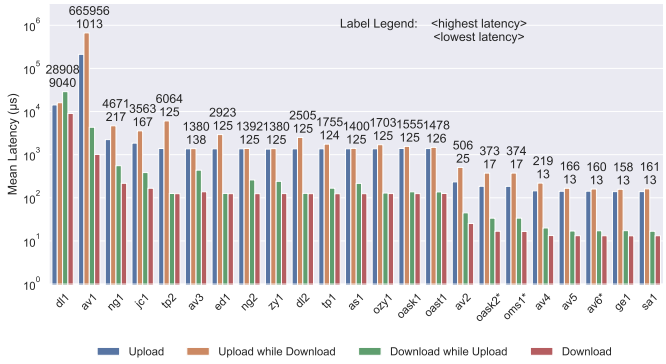

Fig. 5. Results of the TCP-3 measurement. Bar labels display the minimum and maximum latency measured across all types of traffic of each device. Device tags with an asterisk are likely limited by the testbed setup.

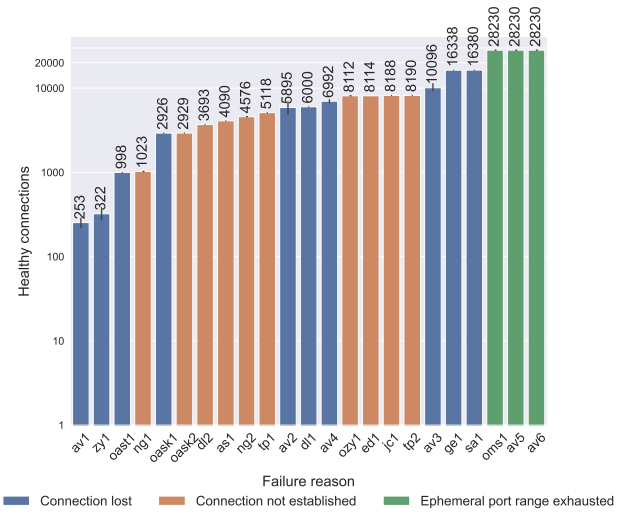

Fig. 6. Number of TCP bindings to a single port

traffic, many have problems achieving line rate when upload and download occur at the same time.

The latency measurements indicate that devices which provide higher throughput also tend to exhibit lower latencies. But unlike the throughput measurements, there exists an imbalance between traffic directions. Across the entire device population, upload traffic generally shows a latency approximately ten times greater than download traffic. That said, for most devices, these latencies are overall fairly low.

It should be noted that, for some devices, these measurements might be affected by our lab setup—for example, by our PPPoE server, which performs routing in software, or by default device settings such as ”green” interface modes that limit interface speeds. One device features a 2.5 Gbit/s interface, which we could not evaluate, as all of our infrastructure interfaces were limited to 1 Gbit/s. We marked all devices affected by known limitations with an asterisk in our figures.

By direct comparison with the results from [1], it is clearly visible that home gateway performance has greatly improved over time. In the original study, only two-thirds of the tested gateways were able to saturate a 100 Mbit/s Ethernet link. In contrast, almost all of our gateways achieve this, with only a few exceptions. Our tests also show that many of our devices feature latencies below 1 ms for download traffic, with only devices with 100 Mbit/s interfaces peaking above 1 ms for their upload traffic. Making more detailed comparisons with the results in [1] is difficult, as no accurate numbers can be extracted from the diagrams of that paper.

3) TCP Bindings (TCP-4): This measurement determines the maximum number of TCP NAT bindings to a single TCP port on the external (WAN-side) node. Fig. 6 shows our results, highlighting one of the most significant differences between our device population and that of the reference paper. The *dl2* gateway of our population is the same model as the *dl6* gateway in [1], except that our model is a later hardware revision and features a more up-to-date firmware. This makes a noticeable difference when it comes to the number of bindings, since in the original paper the device could only handle around 135 connections, whereas our model comes close to 3700 concurrent connections to a single server port. In general and not surprisingly, it seems that newer devices differentiate themselves by being capable of more concurrent connections than older ones, with the latest AVM devices in our tests being able to handle the full ephemeral port range of our testing client. During this test, we also discovered very odd behavior by one box: After crossing a certain connection threshold, the box invalidates all existing bindings - of all potential options to limit the amount of NAT bindings, this is probably the worst.

B. UDP

1) UDP Binding Timeout with a single outbound packet (UDP-1): The UDP-1 test determines the NAT binding timeout for an entry created by a single outgoing UDP packet without a response. Fig. 7 displays the medians of each device's NAT binding timeout for this test. Our home gateway population shares the same low-end as the original paper with a binding timeout of merely 30 seconds, which is below the minimum of 120 seconds as prescribed by RFC 4787 [17]. In fact, a lot of devices stay below that minimum value. Our population's many high-end devices, however, have a timeout of around 300 seconds, which also used by several devices in the reference population, and matches the minimum recommended value of [17]. Our measured median and mean timeouts are higher compared to [1], but it can be assumed that this is due to our particular device population and likely not showing any kind of trend.

2) Binding Timeouts for UDP Streams (UDP-2, UDP-3 and UDP-5): The Linux conntrack implementation has the notion of UDP streams, which refers to UDP flows involving multiple UDP packets in a short timeframe after the initial UDP packet. A different NAT binding timeout can be set for such streams, which are the subject of the UDP-2, UDP-3 and UDP-5 tests. These three differ slightly. For UDP-2, after an initial packet is sent to the outside node, only the outside node sends packets

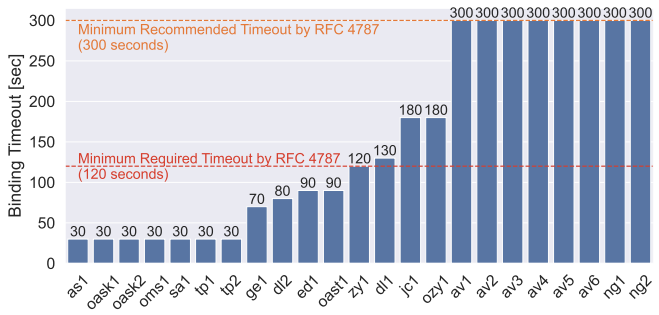

Fig. 7. Binding timeout for UDP connections with a single outgoing datagram

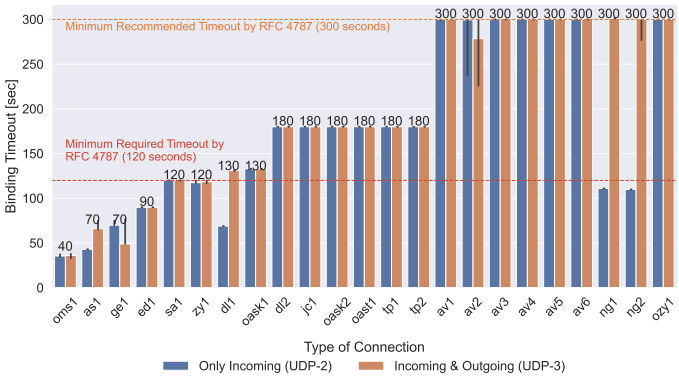

Fig. 8. Binding timeouts for UDP streams

back with an increasing delay between two packets until these do not make it through the NAT anymore. UDP-3 builds on UDP-2 but the inside node answers every received packet. Finally, UDP-5 also builds on UDP-2 but does not connect to random ports, but well-known service ports, which includes SNMP, TFTP, NTP, DNS, but also the HTTP port and - which was not tested at the time since it was not specified back then - QUIC.

The NAT binding timeouts of our home gateways for UDP flows that involve multiple datagrams over its lifetime (UDP-2 and UDP-3) are shown in Figure 8. First, it can be observed that only *as1*, *dl1*, *ng1* and *ng2* measurably treat incoming-only and bidirectional streaming traffic differently, with them providing longer timeouts for bidirectional traffic.

For both Netgear devices and *dl1*, the incoming-only timeout is even shorter than their respective UDP-1 timeout. Interestingly, the UDP-1 timeout of these three devices is greater than the 120 seconds required by RFC4787 [17], but their UDP-2 timeout is not. This however only applies to the incoming-only traffic, as the UDP-3 (bidirectional traffic) timeouts are equal to the UDP-1 results.

All other devices treat both kinds of streaming traffic equally.

Besides the previous three devices, no other device features a shorter timeout for streaming-traffic than for request-response type traffic (UDP-1). Moreover, all devices with compliant UDP-1 timeouts also have compliant streaming timeouts—at least for bidirectional traffic. Devices that implement the recommended timeout of at least 300 seconds do so for both request-response and streaming traffic. Overall, for streaming traffic, 19 out of our 23 devices are in compliance with the timeout requirements of RFC4787 [17].

To discover the presence of special timeouts for well-known service ports, the test for inbound-only streaming UDP traffic has been repeated for various destination ports. The results of this experiment (UDP-5) can be found in Figure 9. Special timeouts can only be observed for two protocols in our device population. First, TFTP is subject to a larger NAT timeout for multiple devices. Additionally, on TP-Link devices, traffic via the DNS port 53 uses the default binding timeout for UDP-1-type traffic even when used in a streaming fashion. *ge1* also alters the timeout for DNS traffic, but instead of shortening the binding's lifetime, it extends it to 180 seconds, which is greater than all previous UDP binding timeouts for this device. In Fig. 9 some bars are missing. This is due to the respective gateways blocking traffic on the given ports.

3) Source-Port Selection (UDP-4): RFC4787 [17] refers to the concept of *port preservation* to describe when a NAT preserves the source port number of the original packet when translating it and forwarding it to the WAN. The UDP-4 test evaluates this port preservation behavior of a home gateway's NAT, and also checks whether a binding can be reused immediately after its timeout expires.

When performing NAT, 22 of our 23 devices keep the source port of the original packet, whereas *dl1* as the only device picks a new random source port. This is generally in line with the original paper, where the majority of devices also kept the source port, even if back then a significantly larger fraction (20%) did not.

C. Other Protocols

1) SCTP & DCCP: In this measurement, we try to establish an SCTP and a DCCP connection through the NAT, and try to send data from the client to the server and vice versa. If this cycle succeeds, the test passes. In contrast to [1], it is possible to successfully establish a DCCP connection through some of the devices. Notably, the only gateways in our population that support DCCP are from AVM. We can only speculate if they explicitly added DCCP support to their NAT, or if this is just a byproduct of DCCP support as part of Linux 2.6.14 [18]. For other gateways, we also observed similar behavior as the original paper, where some devices just rewrite the IP header without adjusting the DCCP checksum, rendering the packet invalid. Others simply drop DCCP packets.

As for SCTP, the share of devices supporting the protocol is significantly higher in our population than in [1] (19/23 vs. 18/34). Since we have home gateways with different release dates from the same vendor, one could, for example, speculate that Netgear somewhere between 2008 and 2010 added SCTP support, as *ng2* supports it whereas *ng1* does not.

These results, however, clearly show the effect of internet ossification as a result of widely deployed middleboxes on

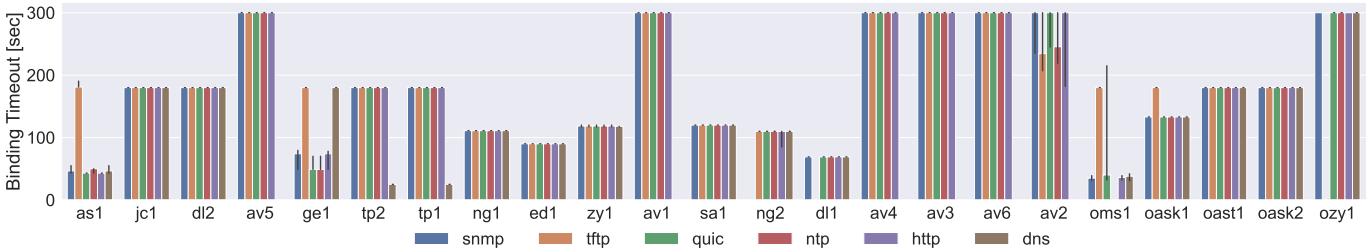

Fig. 9. UDP Binding Timeouts for Inbound Streaming Connections on specific Service Ports

TABLE I
RESULTS OF THE UDP-4 EXPERIMENT

	as1	av1	av2	av3	av4	av5	av6	dl1	dl2
Port Preservation	•	•	•	•	•	•	•		•
	ed1	ge1	jc1	ng1	ng2	oask1	oask2	oast1	oms1
Port Preservation	•	•	•	•	•	•	•	•	•
	ozy1	sa1	tp1	tp2	zy1				
Port Preservation	•	•	•	•	•				

TABLE II
RESULTS OF THE SCTP AND DCCP MEASUREMENTS. • INDICATES THAT THE TEST PASSED SUCCESSFULLY

router	SCTP	DCCP
as1	•	Checksum
av1	•	•
av2	•	•
av3	•	•
av4	•	•
av5	•	•
av6	•	•
dl1	•	Checksum
dl2	•	Checksum
ed1	•	Checksum
ge1	•	Checksum
jc1	•	Checksum
ng1	Dropped	Dropped
ng2	•	Checksum
oask1	•	Checksum
oask2	•	Checksum
oast1	Dropped	Dropped
oms1	•	Checksum
ozy1	•	Checksum
sa1	Dropped	Dropped
tp1	•	Checksum
tp2	•	Checksum
zy1	Dropped (only Response)	Checksum

the internet. SCTP and DCCP are both protocols which have been around for quite a while, but with these NAT routers commonly deployed at the edge of the internet, ubiquitously availability of these two transport protocols has still not materialized 15 years after the study in [1].

D. ICMP

ICMP is an important part of the TCP/IP protocol suite, as it conveys potential problems to hosts. This means that ICMP packets can arrive at the NAT router in response to a TCP or UDP packet, that, for example, could not be delivered for some reason. We tested the NAT support for a range of ICMP packets as a response to TCP and UDP packets.

The results of these tests are shown in Table III. In comparison to the original paper, our population has fewer cases of routers not translating ICMP packets. It can be observed however that all AVM devices and *zy1* do not translate any *Parameter Problem* messages. *dl1* is a special case, as it includes a SoHo firewall. Therefore, we attribute its unwillingness to translate any ICMP message to this firewall implementation. Also noteworthy, ICMP messages which have been deprecated quite some time ago, such as Source Quench [19] in 2012, are still being translated by most home gateways, the exception being our firewall device and *zy1*.

E. DNS

Out of our population of 23 devices, only the DNS forwarders of 6 devices do not support DNS over TCP, of which the most recent one was released in 2015. That is a massive improvement over the device population in [1]. In addition to verifying the basic query functionality, we also investigated whether the devices cache the DNS entries they query, something the original paper did not investigate. To our surprise, only about two thirds of our devices do caching correctly: 8 devices simply always query the upstream DNS server. *as1*, *oask1* and *sa1* are special cases, as they implement caching correctly for queries via UDP, but queries arriving via TCP are not cached at all.

V. CONCLUSION

In short, things certainly have improved regarding home gateway implementations during the last 15 years. One can clearly observe that, for example, with increasing hardware capabilities, certain limitations observed in [1] have disappeared—such as the relatively small number of TCP bindings some devices could hold. Most devices in our population can keep significantly more bindings than the best device from [1]. However, this applies not only to current home gateways, but also to devices released around the time of [1]'s

TABLE III
RESULTS OF THE ICMP MEASUREMENT. EACH ● INDICATES THAT THE RESPECTIVE ICMP MESSAGE PASSES THROUGH THE NAT.

	TCP										UDP									
	TTL Exceeded	Reass. Time. Ex.	Param. Prob.	Net Unreach.	Host Unreach.	Proto Unreach.	Port Unreach.	Frag. Needed	Src. Route Fail.	Source Quench	TTL Exceeded	Reass. Time. Ex.	Param. Prob.	Net Unreach.	Host Unreach.	Proto Unreach.	Port Unreach.	Frag. Needed	Src. Route Fail.	Source Quench
as1	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
av1	●	●		●	●	●	●	●	●	●	●	●		●	●	●	●	●	●	●
av2	●	●		●	●	●	●	●	●	●	●	●		●	●	●	●	●	●	●
av3	●	●		●	●	●	●	●	●	●	●	●		●	●	●	●	●	●	●
av4	●	●		●	●	●	●	●	●	●	●	●		●	●	●	●	●	●	●
av5	●	●		●	●	●	●	●	●	●	●	●		●	●	●	●	●	●	●
av6	●	●		●	●	●	●	●	●	●	●	●		●	●	●	●	●	●	●
dl1																				
dl2	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
ed1	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
ge1	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
jc1	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
ng1	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
ng2	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
oask1	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
oask2	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
oast1	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
oms1	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
ozy1	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
sa1	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
tp1	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
tp2	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
zy1	●	●		●	●	●	●	●	●		●	●		●	●	●	●	●	●	

TABLE IV
RESULTS OF THE DNS MEASUREMENT. ● SIGNIFIES THAT QUERYING AND CACHING IS FUNCTIONING AS EXPECTED. ○ SIGNIFIES THAT QUERYING WORKS, BUT CACHING DOES NOT. IF QUERYING IS NOT POSSIBLE, NO SYMBOL IS PRESENT.

Router	as1	av1	av2	av3	av4	av5
DNS over UDP	●	●	●	●	●	●
DNS over TCP	○		●	●	●	●
Router	av6	dl1	dl2	ed1	ge1	jc1
DNS over UDP	●	○	○	○	●	●
DNS over TCP	●	○	○		●	●
Router	ng1	ng2	oask1	oask2	oast1	oms1
DNS over UDP	○	○	●	●	●	●
DNS over TCP	○	○	○	●	●	●
Router	ozy1	sa1	tp1	tp2	zy1	
DNS over UDP	●	●	○	○	○	
DNS over TCP		○				

publication and in the years shortly after. Sometimes only a hardware revision already improved this particular metric by quite a bit. And this is indeed an important metric — for example, a modern browser such as Firefox may maintain up to 900 concurrent HTTP connections, with up to 6 to a single server. Also, these days, with heavy use of CDN infrastructure, numerous services might be behind a single or a few IP addresses. But not all improvements are a result of more powerful hardware, but can be attributed to firmware changes instead. For example, more devices seem to understand SCTP, and we have even seen support for DCCP, which was not observed in [1]. But even after 15 years, both protocols still lack universal support.

On the other hand, while some metrics have improved, they still do not align with the suggestions of internet standards, even after 15 years. TCP binding timeouts are a good example of this: RFC 5382 [15] suggests to keep bindings for sightly above 2 hours. In [1] timeouts above 24 hours were observed for seven devices, which is likely far too long in practice, while we only had three of those in our device population. On the other hand, we have only seen a single device that follows the recommendation in [15] with all other home gateways implementing far lower timeouts. This might be to keep memory pressure from the NAT table, but then, state handling does not seem to be a real problem for modern home gateways anymore. But if the NATs don't change, presumably for good reasons, then maybe the surrounding recommendations need to be adjusted, and the underlying reasons that led to these recommendations should be reconsidered.

But even if home gateways generally improve over time, we should mention that a lot of these devices can exhibit odd behavior. As previously noted, one device clears out all the NAT bindings once a certain threshold of bindings towards a single IP and port are exceeded. We also found strange behavior in replies to DNS-over-TCP requests, which we could trace back to an old dnsmasq bug, which also highlights the prevalence of outdated software in consumer-grade, low-cost devices.

Our measurements and the ones found in [1] are clearly only a snapshot in time and are only applicable to the specific device population used in each study. When observing the

results, there still is a considerable difference in home gateway model behavior, underlining the importance of studying an even larger population and to continuing this line of work. But, it also proves that some manufacturers have been doing a good job, and continue to do so. In our tests, one particular vendor, of which we had a few devices with a large range of release dates, has produced well-performing devices for about 20 years. However, their products are also among the more expensive ones in our line-up. This again highlights that the device population is important, and while we would have liked to keep vendors and models anonymous, we felt it was necessary to name them.

VI. Future Work

Reproducing the results from [1] was only the first step in continuing the work where the authors of [1] left off, making this an ongoing effort. We plan to add a number of measurements, in particular those that help us understand how these devices contribute to the ossification of the internet, but also on the speed at which these devices receive updates that support newly specified protocols or extensions to existing ones, such as new ICMP types or higher NAT binding timeouts to support, e.g., QUIC.

Since, just like in [1], we were only using the Ethernet ports, we plan to make use of the other interfaces that common home gateways provide to see whether there are any differences in behavior at all. These interfaces include DSL or cable modems, but also the WiFi interface, which is typically a user's first hop towards the internet.

We also still have serveral gateways sitting in our lab for evaluation, which we have not managed to include in this study due to time constraints. These, and hopefully a steadily increasing number of additional devices, will be tested in the near future.

The current suite of tests only included IPv4, similar to the original study. We will extend our set of tests to cover IPv6, too.

Part of our future work will also include non-networking related measurements. In particular, we are interested in the energy consumption of these devices, both when idling and under load.

Finally, we're releasing both our testing code used in this paper and the Python framework we developed to assist in writing middlebox tests called *boxscopy* at https://boxscopy.github.io. We hope this supports other researchers studying middlebox behavior and, ultimately, promotes greater reproducibility in this field through increased sharing of measurements.

References

[1] S. Hätönen, A. Nyrhinen, L. Eggert, S. Strowes, P. Sarolahti, and M. Kojo, "An experimental study of home gateway characteristics," in *Proceedings of the 10th ACM SIGCOMM Conference on Internet Measurement*, ser. IMC '10, New York, NY, USA, 2010, p. 260–266.

[2] A. Langley, A. Riddoch, A. Wilk, A. Vicente, C. Krasic, D. Zhang, F. Yang, F. Kouranov, I. Swett, J. Iyengar, J. Bailey, J. Dorfman, J. Roskind, J. Kulik, P. Westin, R. Tenneti, R. Shade, R. Hamilton, V. Vasiliev, W.-T. Chang, and Z. Shi, "The quic transport protocol: Design and internet-scale deployment," in *Proceedings of the Conference of the ACM Special Interest Group on Data Communication*, ser. SIGCOMM '17. New York, NY, USA: Association for Computing Machinery, 2017, p. 183–196.

[3] J. Gettys, "Bufferbloat: Dark buffers in the internet," *IEEE Internet Computing*, vol. 15, no. 3, pp. 96–96, 2011.

[4] K. Nichols, V. Jacobson, A. McGregor, and J. Iyengar, "Controlled Delay Active Queue Management," RFC 8289, Jan. 2018. [Online]. Available: https://www.rfc-editor.org/info/rfc8289

[5] R. Pan, P. Natarajan, F. Baker, and G. White, "Proportional Integral Controller Enhanced (PIE): A Lightweight Control Scheme to Address the Bufferbloat Problem," RFC 8033, Feb. 2017. [Online]. Available: https://www.rfc-editor.org/info/rfc8033

[6] N. Khademi, D. Ros, and M. Welzl, "The new aqm kids on the block: An experimental evaluation of codel and pie," in *2014 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2014, pp. 85–90.

[7] S. Sundaresan, W. de Donato, N. Feamster, R. Teixeira, S. Crawford, and A. Pescapè, "Broadband internet performance: a view from the gateway," in *Proceedings of the ACM SIGCOMM 2011 Conference*, ser. SIGCOMM '11. New York, NY, USA: Association for Computing Machinery, 2011, p. 134–145.

[8] I. Pefkianakis, H. Lundgren, A. Soule, J. Chandrashekar, P. Le Guyadec, C. Diot, M. May, K. Van Doorselaer, and K. Van Oost, "Characterizing home wireless performance: The gateway view," in *2015 IEEE Conference on Computer Communications (INFOCOM)*, 2015, pp. 2713–2731.

[9] A. Reggani, F. Schneider, and R. Teixeira, "Tracking application network performance in home gateways," in *2013 9th International Wireless Communications and Mobile Computing Conference (IWCMC)*, 2013, pp. 1150–1155.

[10] S. Huang, F. Cuadrado, and S. Uhlig, "Middleboxes in the internet: A http perspective," in *2017 Network Traffic Measurement and Analysis Conference (TMA)*, 2017, pp. 1–9.

[11] G. Detal, B. Hesmans, O. Bonaventure, Y. Vanaubel, and B. Donnet, "Revealing middlebox interference with tracebox," in *Proceedings of the 2013 Conference on Internet Measurement Conference*, ser. IMC '13. New York, NY, USA: Association for Computing Machinery, 2013, p. 1–8. [Online]. Available: https://doi.org/10.1145/2504730.2504757

[12] M. Honda, Y. Nishida, C. Raiciu, A. Greenhalgh, M. Handley, and H. Tokuda, "Is it still possible to extend tcp?" in *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurements*, ser. IMC '11. New York, NY, USA: Association for Computing Machinery, 2011, p. 181–194.

[13] V. Heinrich and R. Winter, "Towards a stateless reverse traceroute protocol," in *IEEE International Conference on Communications, ICC 2024, Denver, CO, USA, June 9-13, 2024*. IEEE, 2024, pp. 1090–1095.

[14] A. M. Mandalari, M. A. D. Bautista, F. Valera, and M. Bagnulo, "Natwatcher: Profiling nats in the wild," *IEEE Communications Magazine*, vol. 55, no. 3, pp. 178–185, 2017.

[15] B. Ford, S. Guha, K. Biswas, S. Sivakumar, and P. Srisuresh, "NAT Behavioral Requirements for TCP," RFC 5382, Oct. 2008. [Online]. Available: https://www.rfc-editor.org/info/rfc5382

[16] "HewlettPackard/netperf," Hewlett Packard Enterprise. [Online]. Available: https://github.com/HewlettPackard/netperf

[17] C. F. Jennings and F. Audet, "Network Address Translation (NAT) Behavioral Requirements for Unicast UDP," RFC 4787, Jan. 2007.

[18] "Linux gets DCCP [LWN.net] — lwn.net," https://lwn.net/Articles/149756/, [Accessed 09-10-2024].

[19] F. Gont, "Deprecation of ICMP Source Quench Messages," RFC 6633, May 2012.

[0]From the FCC filing database

TABLE V
HOME GATEWAY POPULATION

Tag	Vendor	Model	Released	Support
Freely available devices				
as1	Asus	RT-N12 (D1)	2013	Ended 2020
av1	AVM	FRITZ!Box 7170	2005	Ended 2013
av2	AVM	FRITZ!Box Fon WLAN 7390	2010	Ended 2019
av3	AVM	FRITZ!Box 3272	2013	Ended 2018
av4	AVM	FRITZ!Box 7490	2013	Ended 2024
av5	AVM	FRITZ!Box 7590	2017	Still supported
av6	AVM	FRITZ!Box 5530 Fiber	2020	Still supported
dl1	D-Link	DFL-160	2009	Ended 2017
dl2	D-Link	DIR-600 (B6)	2011	Ended 2020
ed1	Edimax	BR-6428nS (v4)	2016	No longer listed on international site
ge1	Genexis	E600	2021	Still supported
jc1	JCG	JHR-N825R	2012	Product page no longer available
ng1	Netgear	WGR614 v9	2008	Ended 2016
ng2	Netgear	WNR2000 v3	2010	Ended 2023
tp1	TP-Link	TL-WR841N (v11)	2015	Still supported
tp2	TP-Link	TL-WR1043ND	2013	Ended 2017
zy1	Zyxel	NBG-418N v2	2014	Ended 2023
Provider-branded devices				
sa1	Sagemcom	F@ST 5366se (Deutsche Glasfaser Edition)	2019	Still supported
oast1	Astoria Networks	o2 Box 6431	2014	Ended 2021
oask2	Askey	o2 HomeBox 6742	2022	Still supported
oask1	Askey	o2 HomeBox 6741	2020	Still supported
oms1	MitraStar	o2 HomeBox 6642	2022	Still supported
ozy1	Zyxel	o2 HomeBox 6641	2017	Still supported