Cloudflare outage on July 17, 2020

July 17, 2020 7:22 PM

John Graham-Cumming

Today a configuration error in our backbone network caused an outage for Internet properties and Cloudflare services that lasted 27 minutes. We saw traffic drop by about 50% across our network. Because of the architecture of our backbone this outage didn't affect the entire Cloudflare network and was localized to certain geographies.

The outage occurred because, while working on an unrelated issue with a segment of the backbone from Newark to Chicago, our network engineering team updated the configuration on a router in Atlanta to alleviate congestion. This configuration contained an error that caused all traffic across our backbone to be sent to Atlanta. This quickly overwhelmed the Atlanta router and caused Cloudflare network locations connected to the backbone to fail.

The affected locations were San Jose, Dallas, Seattle, Los Angeles, Chicago, Washington, DC, Richmond, Newark, Atlanta, London, Amsterdam, Frankfurt, Paris, Stockholm, Moscow, St. Petersburg, São Paulo, Curitiba, and Porto Alegre. Other locations continued to operate normally.

For the avoidance of doubt: this was not caused by an attack or breach of any kind.

We are sorry for this outage and have already made a global change to the backbone configuration that will prevent it from being able to occur again.

The Cloudflare Backbone

Cloudflare operates a *backbone* between many of our data centers around the world. The backbone is a series of private lines between our data centers that we use for faster and more reliable paths between them. These links allow us to carry traffic between different data centers, without going over the public Internet.

We use this, for example, to reach a website origin server sitting in New York, carrying requests over our private backbone to both San Jose, California, as far as Frankfurt or São Paulo. This additional option to avoid the public Internet allows a higher quality of service, as the private network can be used to avoid Internet congestion points. With the backbone, we have far greater control over

where and how to route Internet requests and traffic than the public Internet provides.

Timeline

All timestamps are UTC.

First, an issue occurred on the backbone link between Newark and Chicago which led to backbone congestion in between Atlanta and Washington, DC.

In responding to that issue, a configuration change was made in Atlanta. That change started the outage at 21:12. Once the outage was understood, the Atlanta router was disabled and traffic began flowing normally again at 21:39.

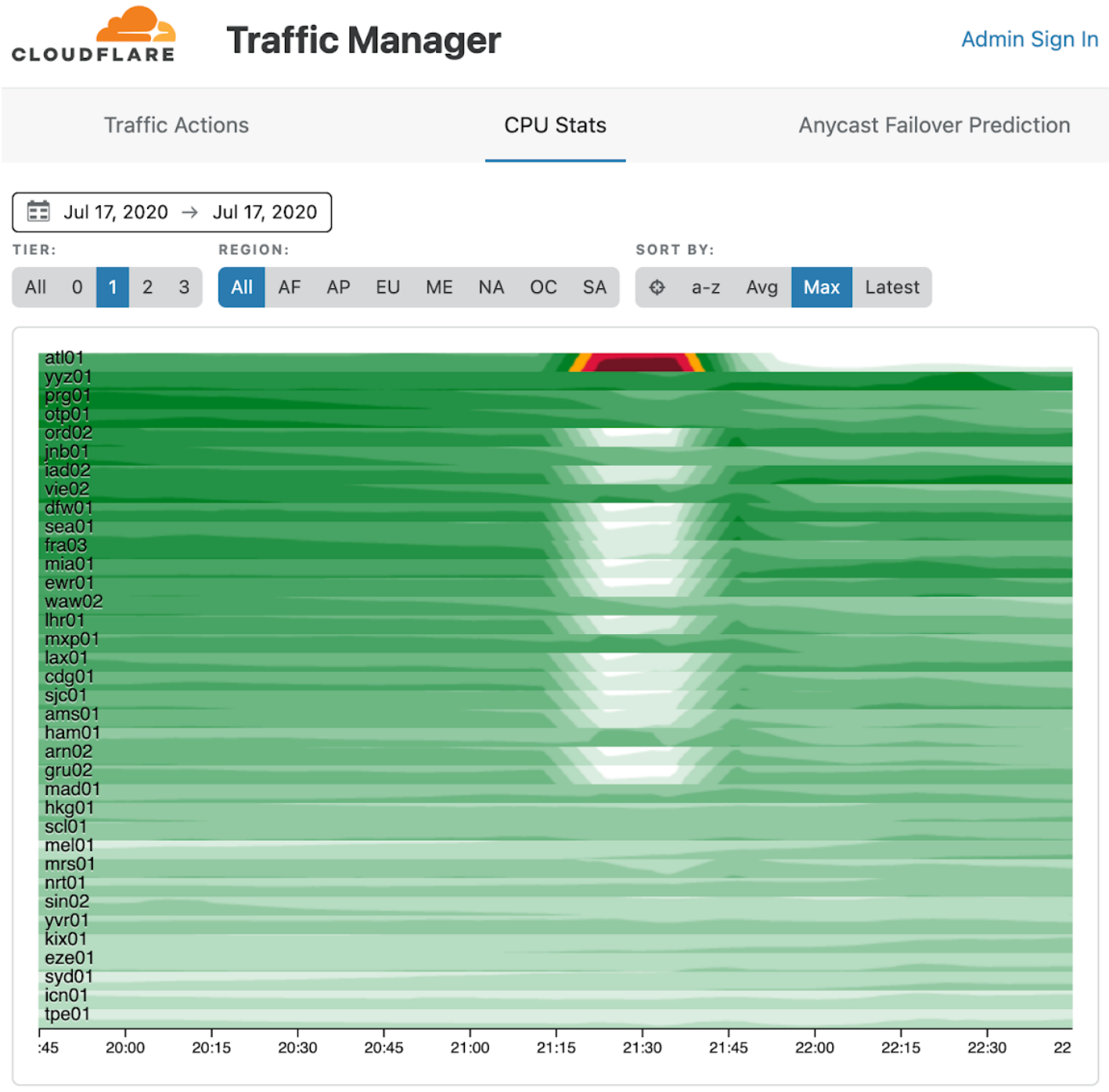
Shortly after, we saw congestion at one of our core data centers that processes logs and metrics, causing some logs to be dropped. During this period the edge network continued to operate normally.

- 20:25: Loss of backbone link between EWR and ORD
- 20:25: Backbone between ATL and IAD is congesting
- 21:12 to 21:39: ATL attracted traffic from across the backbone
- 21:39 to 21:47: ATL dropped from the backbone, service restored
- 21:47 to 22:10: Core congestion caused some logs to drop, edge continues operating
- 22:10: Full recovery, including logs and metrics

Here’s a view of the impact from Cloudflare’s internal traffic manager tool. The red and orange region at the top shows CPU utilization in Atlanta reaching

overload, and the white regions show affected data centers seeing CPU drop to near zero as they were no longer handling traffic. This is the period of the outage.

Other, unaffected data centers show no change in their CPU utilization during the incident. That's indicated by the fact that the green color does not change during the incident for those data centers.



What happened and what we're doing about it

As there was backbone congestion in Atlanta, the team had decided to remove some of Atlanta's backbone traffic. But instead of removing the Atlanta routes from the backbone, a one line change started leaking all BGP routes into the backbone.

```
{master}[edit]
atl01# show | compare
[edit policy-options policy-statement 6-BBONE-OUT term 6-SITE-LOCAL from]
!       inactive: prefix-list 6-SITE-LOCAL { ... }
```

The complete term looks like this:

```
from {
    prefix-list 6-SITE-LOCAL;
}
then {
    local-preference 200;
    community add SITE-LOCAL-ROUTE;
    community add ATL01;
    community add NORTH-AMERICA;
    accept;
}
```

This term sets the local-preference, adds some communities, and accepts the routes that match the prefix-list. Local-preference is a transitive property on iBGP sessions (it will be transferred to the next BGP peer).

The correct change would have been to deactivate the term instead of the prefix-list.

By removing the prefix-list condition, the router was instructed to send all its BGP routes to all other backbone routers, with an increased local-preference of 200. Unfortunately at the time, local routes that the edge routers received from our compute nodes had a local-preference of 100. As the higher local-

preference wins, all of the traffic meant for local compute nodes went to Atlanta compute nodes instead.

With the routes sent out, Atlanta started attracting traffic from across the backbone.

We are making the following changes:

- Introduce a maximum-prefix limit on our backbone BGP sessions - this would have shut down the backbone in Atlanta, but our network is built to function properly without a backbone. This change will be deployed on Monday, July 20.

- Change the BGP local-preference for local server routes. This change will prevent a single location from attracting other locations' traffic in a similar manner. This change has been deployed following the incident.

Conclusion

We've never experienced an outage on our backbone and our team responded quickly to restore service in the affected locations, but this was a very painful period for everyone involved. We are sorry for the disruption to our customers and to all the users who were unable to access Internet properties while the outage was happening.

We've already made changes to the backbone configuration to make sure that this cannot happen again, and further changes will resume on Monday.

Post Mortem Outage Engineering

41 Comments **Cloudflare Blog** **Login**

Recommend 21 Tweet Share **Sort by Best**

Join the discussion…

LOG IN WITH OR SIGN UP WITH DISQUS

Name

Bob Whitley • a month ago • edited

I really appreciate the detailed explanation and transparency. I'm used to other organizations that have had issues like this just mysteriously fixing them and/or with vague explanations so it is refreshing to get this level of detail especially so quickly after the event occurred.

30 • Reply • Share ›

Michael Schirrmeister • a month ago • edited

i feel sick for this poor guy who messed this up. I hope he / she did not suffer any consequences or were even fired so can continue to be part of your team.That sounds like the worst day of her / his technician life. But mistakes happen to the best

7 • Reply • Share ›

Steven • a month ago

Amazing how a single configuration error can cause such a hassle.

5 • Reply • Share ›

Billy O'Neal • a month ago

So for one brief moment all the data traffic got routed through Atlanta, just like all the flight traffic

4 • Reply • Share ›

Louis-Benoit JOURDAIN • a month ago

Not so long ago, Cloudflare provided some explanation about BGP routes leak to explain the issue with Verizon and suggested to implement RPKI and a prefix limit.

https://blog.cloudflare.com...
"A BGP session can be configured with a hard limit of prefixes to be received. This means a router can decide to shut down a session if the number of prefixes goes above the threshold. Had Verizon had such a prefix limit in place, this would not have occurred. It is a best practice to have such limits in place. It doesn't cost a provider like Verizon anything to have such limits in place. And there's no good reason, other than sloppiness or laziness, that they wouldn't have suchlimits in place."

Any reason, besides sloppiness or laziness, the prefix limit was not implemented on the backbone after Verizon's incident?

1 ^ | v • Reply • Share ›

John Airey • a month ago

Good update but the images are missing

1 ^ | v • Reply • Share ›

Val Vesa Mod ➚ John Airey • a month ago

What images are missing?

^ | v • Reply • Share ›

John Airey ➚ Val Vesa • a month ago

see more

see more

• Reply • Share ›

Brian • a month ago

All major router changes should be triple checked before being activated and not by the same engineer.

4 | 5 • Reply • Share ›

Jason Mathew ➚ Brian • a month ago

Said like a true man who has never made a mistake in his life!

3 | 1 • Reply • Share ›

Kugel ➚ Brian • a month ago

pretty sure there is a change management team with all pre checks but in the end we are human even multiple can fail to notice an issue, also really appreciate the transparency but tbh i expected more redundancy for a company like cloudflare.

| 1 • Reply • Share ›

Stokkolm ➚ Kugel • a month ago

The redundancy is there, that's the worst part of the issue. The mistake effectively removed their redundancy. The problem here is not Cloudflare's redundancy or lack thereof, it's with the BGP protocol and how it works and in this instance, one engineer's lack of attention or knowledge about how to correctly address the network congestion that they were experiencing at the time.

3 | • Reply • Share ›

Asuka Shikinami Langley • a month ago • edited

While I appreciate the in-depth explanation, I can't help but feel like it's not a good idea to have a system where a single bad line of code can knock out literally half the internet for over an hour.

1 | 1 • Reply • Share ›

BurtonHohman ➚ Asuka Shikinami Langley • a month ago

That's correct, and that's why they are likely making changes. The problem with so many systems is that they work until they don't, and there's a balance between QAing every single possible problem and pushing things live. I would assume this incident will have the re-evaluate how they handing their configuration changes. I would also hope this serves as other companies doing a similar thing with

how open Cloudflare was in what happened. It likely will also result in some companies evaluating backups to Cloudflare as a result of the incident as well.

2 ^ | v • Reply • Share ›

Michael Peterman • 23 days ago

I agree, a very good example about how in IT it can just be a simple error. And how to correctly fix it in the future. Lets this be an example for all big providers what true transparency looks like.

^ | v • Reply • Share ›

Subhan Azeem Qureshi • a month ago

The outage duration could have been a few minutes if the engineer used 'commit confirmed 3' instead of 'commit' . Human errors are always possible, but precautions can be taken to minimize impact

^ | v • Reply • Share ›

John Doak • a month ago

I love cloud flare articles, very transparent which I find refreshing.

I don't think the post-mortem here addresses the underlying issue: there was no software tooling to do this job safely and that is portable to all Cloudflare routing platforms.

Setting max-prefix number sounds great, until someone disables it (I've seen it happen). And this implies that CloudFlare is single vendor for backbone routers. You can do similar disasters on other platforms.

These Juniper routers (that is what the output is from) have a beautiful policy engine, but it is complicated.

Google had the same problems in early year but this type of configuration was removed in favor of tooling. The tooling would make the router the least preferential so that traffic would flow through that router only when it had no other route.

Those basic tools had a lot of draining options and eliminated this class of error.

The tooling became much more advanced over the years, moving from "tooling" to services to an entire ecosystem.

This feels like a growth problem where network growth is outstripping network expertise. The answer to that is to move expertise into tools.

^ | v • Reply • Share ›

Phil Clemens • a month ago

Appreciate the transparency, all the way down to the policy config.

^ | v • Reply • Share ›

SPAM ADDRESS • a month ago • edited

While I appreciate the open and honest transparency, this royally sucked for me personally, and everyone in IT as a whole. I am the senior system architect in charge of a very large public/private cloud, so of course when you went down, we were perceived as down, and it was NOT our fault. Of course to us, IT folks, we know better, but not the client, they are usually IT illiterate. No user listens or understands when outside forces (like this one) damages my (our) uptime reliability. All the client knows is that we are down, when no, we are not down. Your public DNS provider and core backbone was down, which was also down for us as well, and the majority of the east coast. That's not our fault, nor can we do anything about it, BUT the clients want refunds, who is paying for that, us, you?! Ultimately we will end up eating YOUR mistake in DOLLARS, because WE have to make it right for OUR customers so we lose in reputation, and we lose in financial aspects because we have to make it right with the clients. This means giving them back money they paid us for reliable, uptime when WE had no part of this outage so why are WE forced to pay for your poor Q/A of code changes (or lack thereof Q/A code changed)? That really sucks. We were 100% up throughout this entire ordeal, meaning, we were physically up and online, but with no access over the internet, clients cannot reach in and work, so to them, we are perceived as down. This makes us look poor in the eyes of our clients and it was nothing we could prevent, nor solve ourselves so it left us without a clue what was going on, while holding our heads down low not knowing what is happening. This makes us look terrible, we strive for 4 9's (yes I know, no one rarely gets 4 - 9's uptime) but we are damn close, until this type of crap happens.

^ | v • Reply • Share ›

Vikas Robert • a month ago

Good explanation about the issue happen.. Didn't the change team analyse the routes and preference value in prefix list before change ? This was like a major change and failed...

^ | v • Reply • Share ›

Scott Chang • a month ago

Thanks for the details. You might want to use bgp as-path filter + prepend if it's internal only (iBGP) without exchanging ISP with community. Also if the links are all private(waive link, fiber or leased line), you may consider enable BGP BFD on interfaces of

BGP peers to reduce the 6 mins BGP convergence time.

Reply • Share ›

Jason Mathew • a month ago

Great work! Love the transparency! This type of document is great for technical folk to fight on your behalf when their non-technical managers as to why we shouldn't use "competitor x". Much better than being lied to like other carriers do and which gives the technical people no leg to stand on.

Reply • Share ›

Mostafa Ammar • a month ago

Thanks for elaboration, hope everything goes fine monday .

Reply • Share ›

PeterIV • a month ago

Interesting, still cannot call an AT&T home phone located in Western Illinois (217)from a Verizon cell phone Southern Indiana (812), So, now I have an 80 year old parent that I cannot call as here locally AT&T has no service. I am just recording all the noise I get when I attempt to call him so as I can file a complaint

Reply • Share ›

adrianTNT • a month ago

Yesterday there were serious issues with sites that have nothing to do with CloudFlare.
Is CloudFlare just trying to get "credit" (free publicity) for a big internet downtime ? From Hetzner host in Germany I could not reach my home IP, none of these are related to CloudFlare. It seemed that Google's DNS and others also misbehaved, I was restarting servers and clearing DNS caches for some hours, it didn't seem CloudFlare related at all to me.
On the other hand I have sites behind CloudFlare, these worked OK.

Reply • Share ›

John Airey ↗ adrianTNT • a month ago

If the authoritative DNS is hosted with cloudflare, the site will go down in minutes. Even cloudflare.com disappeared from the UK, leaving no obvious way to figure out what was going on.

Reply • Share ›

adrianTNT ↗ John Airey • a month ago

Hetzner also mentioned that it is a global issue, not

a CloudFlare issue, same as what I noticed:

> The error already occurred at 23:25 on a global level. Other DNS providers (Google's 8.8.8.8, Cloudflares 1.1.1.1) also appear to be affected.

Reply • Share ›

sc456a • a month ago

Thanks for the explanation, it's always refreshing to see a company take transparency seriously. A few questions:
1) Why doesn't Cloudflare have a fallback system in place in case a backbone router gets overwhelmed? There are a number of ways a router could be overwhelmed by outside forces, not just by an internal misconfiguration. This is concerning.
2) For a significant period after it was apparent that a critical issue was occurring, https://www.cloudflarestatu... showed "All Systems Operational". What's the point of a status page that doesn't clearly illustrate the current status of the network? I believe during the outage the only thing I saw was that a few NA locations were "Re-routed".
3) During the incident it was impossible for us to access the Cloudflare dashboard to make configuration changes, meaning we couldn't greycloud our sites and bypass Cloudflare. Since I assume the Cloudflare dashboard is hosted on the Cloudflare CDN, how is it possible that a routing issue meant we couldn't access the website itself? I assume the answer is that all Cloudflare traffic was being routed through the overwhelmed Atlanta router, but why wouldn't the system automatically detect this and change the routing configuration to an alternative/secondary config alleviate it? Websites and services using Cloudflare could have stayed up even if performance was degraded if something like this were in place, correct?

Reply • Share ›

John Airey ↗ sc456a • a month ago

Read the page. Like many outages this wasn't a single failure but two together. Cloudflare will learn from this, that's for sure.

If you want a real scare, just look how poorly secured some of the root DNS servers are...

Reply • Share ›

Mat ↗ sc456a • a month ago • edited

2) according to a HN comment:

> Because of the way we

> securely connect to StatusPage.io from most locations where our team is
> based. The traffic got blackholed in ATL, keeping us from updating it.
> An employee in our Austin office was finally able to use his Google Fi
> phone and connect through a PoP that wasn't connected to our backbone so
> didn't have traffic blackholed. Something we'll address going forward.

https://news.ycombinator.co...