ACHIEVING INTERNET RESILIENCE

Preventing Outages in 2023

What We Can Learn From Recent Failures

catchpoint

Contents

Introduction 3

Six Lessons Learned from the Recent Failures 5

Incident Review: January 26, 2023 – Microsoft Cloud Outage Causes Global Workforce Disruptions 13

Incident Review: December 5-7, 2022 – How Catchpoint's IPM Platform Detected Amazon's Two-Day Search Issue 20

Incident Review: August 2022 – 3 Lessons from a DNS Resolution Failure Incident 25

Incident Review: February 22, 2022 – Slack Outage – Good Morning! Here's 16 Minutes of Stress! 30

Incident Review: December 7, December 15 and December 22, 2021 – What Can We Learn from AWS' December Outagepalooza? 34

Incident Review: November 16, 2021 – Google Cloud Outage has Widespread Downstream Impact 38

Incident Review: October 7, 2021 – The Ripple Effect of A BGP Misconfiguration at Telia 42

Incident Review: October 4, 2021 – A Case of Social Networks Going Anti-social at Facebook 45

Incident Review: September 29-30, 2021 – Issues Caused by the Let's Encrypt DST Root CA X3 Expiration 51

Incident Review: September 7, 2021 – A Case of Dr. BGP Hijack or Mr. BGP Mistake at Spectrum? 57

Conclusion 60

About Catchpoint 61

Introduction

The last eighteen months of Internet outages or failures, from Amazon to Facebook, showed how connected and relatively fragile the Internet ecosystem is. Increasingly, we are seeing problems when even a small component goes askew. This, coupled with an increasing dependence on the Internet for all types of business activity, means Internet Resilience has become a mission-critical priority.

The likelihood of an issue having a cascade effect has compounded due to:

- **The Internet is the new network:** as companies shut down their legacy networks and run their enterprise without a LAN, WAN or OnPrem datacenter, everything is now on the Internet.
- **The increasing centralization of Internet infrastructure:** this results in every outage having a wider footprint.
- **Greater complexity:** as systems become more complex, it becomes more difficult to find the origin of an issue and respond quickly.
- **The sustained shift to hybrid work:** this is leading to entire workforces becoming reliant on distributed cloud-based applications. Employees are impacted, as well as customers.
- **The move to cloud-based applications:** making it increasingly difficult to know where issues lie — and frequently, problems lie beyond IT's control. This can leave business at the mercy of third-party networks and providers.
- **The fragility and complexity of the Internet:** the Internet requires multiple networks, protocols, agents and sub-systems to work together, often in milliseconds, to perform. Each component part is not a magically resilient, infallible network.

"The Future Enterprise must address unevenness in connectivity across different environments and locations as employees, customer, and partners increasingly look for — and expect — digital experiences supported by ubiquitous, reliable and robust connectivity."

IDC Research

It is obvious that you must implement an incident response process, but what is less obvious, yet even more crucial is the fact that you need a change management process that nudges teams to better handle changes that can have catastrophic results.

That way, your IT teams and the wider business have a robust plan to follow when an issue occurs (and it will occur, no matter how careful you are). A comprehensive plan should include information that allows you to best understand where problems originate from, exactly what is happening, and what your response to the incident should be.

At Catchpoint, we report on major outages as they happen. We are able to detect outages before anyone else, whether they shut down your entire site or just one page or function, frequently before even the impacted company updates their status page that there is an issue. Our powerful Internet Performance Monitoring platform allows us to discover the areas affected, often precisely those that teams do not think have been impacted.

In a collective effort to understand how better to stem outages in the future and ensure greater Internet Resilience in 2023, we have gathered the top pieces of analysis from our blog written over the last eighteen months in this White Paper.

Explore Our Blog

Let's begin by discussing our top six lessons learned and then we'll dive into the details of the incidents themselves.

Six Lessons Learned from the Recent Failures

1 Monitor what matters

All of IT understands there are certain things we have under our control and others that are not. Since we tend to perform the most activity around the areas we can directly control, they're what we pay most attention to. Therefore, we monitor our containers, VMs, hardware, and code. We have unit and automated tests running on our test environments. We have systems that consume all the logs we can afford to consume.

While these are definitely great practices, they are not enough to get in front of outages. Issues can and will occur in other parts of the system — and you can't simply ignore this fact, or you will get into trouble.

Don't just monitor the components of the system you have within your direct control. Develop an IPM strategy to monitor what you deliver to your consumers or users. Monitor their output and performance, even if certain things in the Internet Stack appear not to be within your obvious control, such as third parties like CDNs, managed DNS, and backbone ISPs. While these might not be your code or hardware, your users will be impacted by any issues they experience, and so will your business.

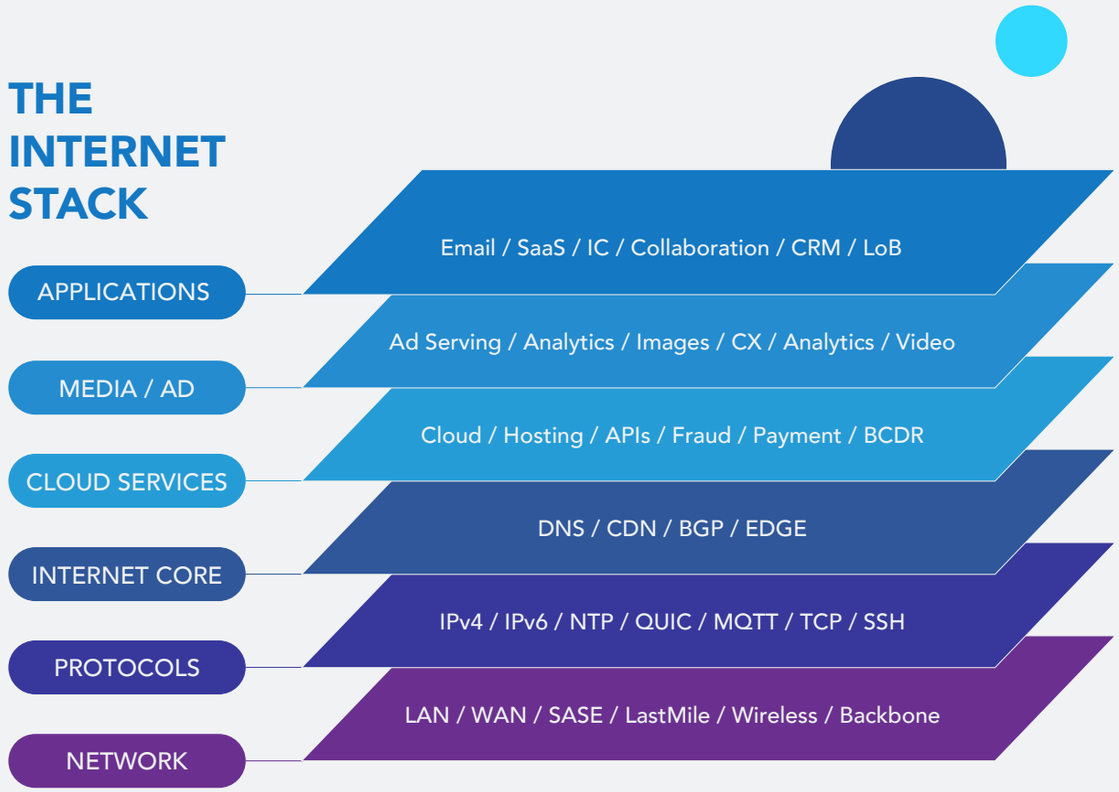
Be ready to act: If you are monitoring the entire Internet Stack, you can act when components go awry that are impacting your users. Examples of actions you could take include switching to a backup solution, and clearly communicating to your users what's going on while your teams figure out what to do.

2 Map your Internet Stack

We assume that things we haven't changed or modified will continue to work as always, today, and tomorrow. In other words, we think that because we didn't make a change, the system won't break. Hence, we don't pay as much attention to DNS, BGP, TCP configuration, SSL, the networks the data traverses, or any single points of failure in the complex web of infrastructure that comprises the Internet (what we at Catchpoint call the Internet Stack).

This issue is exacerbated by the way in which cloud has abstracted a lot of the underlying network from Dev, Ops and network teams. That can make it harder to find a problem. The result is that when these fundamental components fail, it catches us by surprise. It takes a long time to detect, confirm, or find root cause, because teams were not properly prepared.

Therefore, ensure that you continuously monitor these aspects of your system. Put into place a plan so your teams are trained and know what to do in case of failure. Finally, practice your response, because the longer your team doesn't use a skill (and therefore loses muscle memory), the longer your outage could last.



3 Intelligently automate

Automation has been the biggest move of the last decade in IT organization. It helps us be efficient, removes room for error, and simplifies complex tasks. However, we tend not to apply the same rigor to automation as we do to production systems. We can't ignore the importance of designing and testing automation to make sure there are no bugs hiding in the code or script.

For example, look at the Facebook failure from October 2021. A logic problem in the automation led to all the DNS servers being taken out of the BGP announcement, because the servers were not in a proper state.

The result was that there were no DNS servers available, and Facebook couldn't do the simplest thing any site should during an outage: issue a notice for their users that explains they are down and working on fixing the situation.

Ultimately, this issue was a design flaw in the automation logic. It could have been caught before it was implemented, or at least during testing of the automation script/code. The designer of the system should have thought about what happens if all the DNS servers were impacted… what can automation do?

If you have a general lack of testing and limited awareness of these sorts of issues, you can be taken by surprise when things go wrong. In addition, if you wait until a failure of automation happens to understand the quality of your automation framework and scripts, it will leave too much to do and too much debt collected. Hence, it's best to integrate testing into the automation design and implementation.

To automate intelligently, follow these four simple processes:

1. Apply proper design processes, such as documentation and design reviews.
2. Ensure you consider edge cases. They are deadly traps.
3. Perform functional, automated, and regression testing of automation scripts.
4. Test periodically, even if no change to the automation code was made, to ensure everyone knows how it works and what it does.

4 "Trust and verify"

When we at Catchpoint are working with companies new to monitoring Internet performance, we often hear, "We are not sure why this AWS outage impacted us since we are not in AWS."

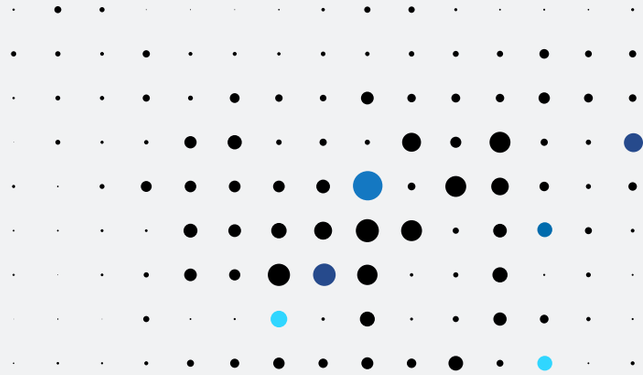
As we help them peel through the layers of their providers across the Internet Stack, they discover that they indeed have vendors in the critical path of their business transactions who are on AWS and were impacted. Then, they understand that the outage spread like a virus to impact them.

If you have taken courses on managing people or delegation, you probably have heard that one of the key strategies for successfully delegating work is "trust and verify." This strategy holds true in operating complex systems that rely on multiple teams, and often other vendors like cloud compute, CDN, Managed DNS, etc.

This phrase means that when another team or vendor is making a change, you should not only trust that they did their job in analyzing and planning the change, but also verify that what they have planned is not going to impact you.

For example, if you receive a notice that your key provider is planning a major system update in a week's time, have your team ready with:

- A crisis call plan (who is on call, what should they do, who to page if there's an issue, etc.).
- A plan to mitigate failure from the other party (and make sure that you have tested it ahead of time).
- A clear understanding of what will be impacted by any failure.
- A communication plan and templates that can be easily populated with need-to-know information for users and customers.
- A monitoring and observability plan that covers all the bases.



5 Implement an Internet Performance Monitoring plan

A thorough, well-thought-out approach to Internet Performance Monitoring (IPM) is essential. This plan should include the ability to analyze trends over time, and therefore it's critical to establish baselines for how things looked before the change, so you can compare it with how things look after.

Some examples of why this is important include:

- If you're updating your firewall rules or network device firmware, are you sure that it hasn't started dropping some connections, or added latency to every packet sent?
- If your DNS vendor has scheduled a maintenance window to upgrade their systems, do you know how long a DNS lookup took before, so you can make sure it hasn't gotten slower?
- If your engineering team is pushing an updated version of your application, do you have all the relevant baselines measured? Speed? Size? Responsiveness? What else is important to your users?
- Do you have data on how time of day, day of week, holidays, or special events change the performance of the application/service?
- Are you monitoring services from both inside the production environment (i.e., inside your firewall) and outside by deploying APM and IPM to ensure 360-degree visibility into the experience for both your internal systems and the external world?
- Are you looking only at code tracing, logs, and CPU, or also continuously testing from an IPM perspective i.e. what matters the most: the output of the service from the perspective of the consumer (whether user, machine, etc.)?

"No service is too big to fail. Over the last 18 months, failures happened from the giants of the Internet: Akamai, Amazon, Facebook, Fastly, and Salesforce. No one can be 100% certain it won't happen to them — or it won't happen again, and how seriously it will impact them. That's why it's critical that you fully understand what you need to do when (not if) such a failure happens."

Dritan Suljoti, Chief Product and Technology Officer, Catchpoint

6 Practice, practice, practice

The biggest lesson of all might be that businesses need to ensure their teams practice what to do when failure happens. What we at Catchpoint observed in most of the outages of the last eighteen months is that a lot of teams were not prepared. It took too long to identify issues and figure out what to do, and the Mean Time to Repair (MTTR).

You need to ensure your teams are ready for failure. Business systems have grown in complexity. Likewise, the Internet is also daily growing in complexity. Single points of failure are spread out throughout the complexity. Organizations rely on external providers more than ever and those provider systems have often become even more complex than our own.

When a key service of a cloud provider goes down, for instance, it can lead to a huge chain of failures across interdependent products and companies.

Implement the following steps, so you'll be prepared for failures:

- Put a monitoring strategy in place that centers IPM. This is what will detect and let your team react at speed from a user perspective when an "outage hits the fan."
- Design and test your crisis process.
- Develop robust playbooks or run books.
- Plan for the outages of key vendors.
- When practicing a crisis, turn it into a game — but do practice, so that people are not struggling when an outage occurs.

Catchpoint's Managed Services Program

Work with our seasoned team of experts year-round

Catchpoint can provide you with 24/7/365 support

Rely on our team of experts that can provide training, on-boarding assistance, best practices, best-practice processes, and many years of expertise to ensure your organization can achieve Internet Resilience.

Our team can extend or complement yours, provide regular KPI updates, and offer optimization opportunities. You get world-class expertise and an extra layer of protection.

Find out more at: https://www.catchpoint.com/services-support

Need extra coverage during peak periods?

Choose Catchpoint's Internet Resilience Program.

Need expert help year-round?

Choose Catchpoint's Managed Services Program.

"We've been able to develop a fantastic working business relationship with Catchpoint, both with our account manager, our support engineer, and the technical support that's available to us 24/7. If you're considering purchasing Catchpoint, and you're looking for a tool that can provide you with a high level of granularity, a 24/7 support team, an account team that's here to help you work through any objective, there's really no other tool on the market right now that can provide that level of support and detail."

– Dominic Trelford, Supervisor, Network Security Ops, Jenny Craig

Catchpoint's Internet Resilience Program

Augment your observability team when you need it most

Get the essential coverage you need during peak periods

"Catchpoint's Professional Services team extends great value through 24x7 proactive monitoring of our 2000+ websites. They played a key role in achieving zero downtime across all our websites during Black Friday."

– Martin Norato Auer, Vice President of CX Observability and Automation Foundations, SAP

Whether you're preparing for a holiday event or hot product launch, you need to be ready when high traffic hits. At any time throughout the year, our award-winning team is ready to provide you with 24/7 incident management support.

During Black Friday 2022, we supported some of the world's largest retailers. Here are some highlights:

- We ran hundreds of tests, continuously, from thousand of vantage points around the world to identify anything in the Internet Stack that could impact customer experience
- Detected dozens of issues that ranged from broken content links, to upcoming SSL certificate expiration, and application failures
- Quickly remediated business-impacting site issues, DNS performance, latency, and other critical incidents that would have significantly affected revenue and brand

When the Internet is critical to your business, it is critical to have the best technology, the best expertise, and the best team by your side. This is what the Catchpoint team offers.

Find out more at:
https://www.catchpoint.com/internet-resilience-program

Incident Review: January 26, 2023 – Microsoft Cloud Outage Causes Global Workforce Disruptions

By Ahamed Ali, Alessandro Improta, Anna Jones and Luca Sani

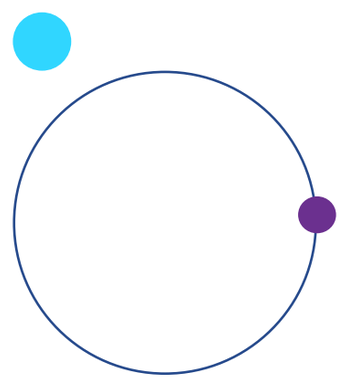
Many of us (indeed 1 billion plus users worldwide) rely on Microsoft for our essential work activities and were impacted Wednesday January 25th when the cloud service provider experienced a prolonged outage. Internet Resilience is a business priority because when critical workforce services like Microsoft go down, global teams are hugely disrupted. The sooner IT teams can detect the problem, isolate its cause, and troubleshoot, the sooner your workforce can resume its normal operations and limit impact to the business.

Multiple Microsoft services started to fail from Jan 25, 2023 - 07:08 UTC/02:08 EST across the globe and the issue continued to impact users for around five hours. The outage didn't only impact Microsoft 365 services, such as Teams, Outlook, and SharePoint Online, but also Microsoft games such as HALO and security features like Microsoft Defender for Identity, along with its crown jewel cloud offering, Azure. Based on incident MO502273 on the Service health dashboard, the preliminary root cause was identified as "a wide-area networking (WAN) routing change [that] resulted in users being unable to access multiple Microsoft 365 services."

Microsoft expand on root cause

Redmond elaborated on the root cause on January 27, saying the global outage was caused by a single router IP address change that led to packet forwarding issues between all other routers in its WAN. "As part of a planned change to update the IP address on a WAN router, a command given to the router caused it to send messages to all other routers in the WAN, which resulted in all of them recomputing their adjacency and forwarding tables," Microsoft said. "During this re-computation process, the routers were unable to correctly forward packets traversing them."

While the change was planned, the "command" given to the router was clearly an error – indeed a costly one, which led to the widespread service impact, hitting in waves that peaked approximately every thirty minutes, as shared on the Microsoft Azure service status page (which itself was impacted, intermittently displaying "504 Gateway Time-out errors").

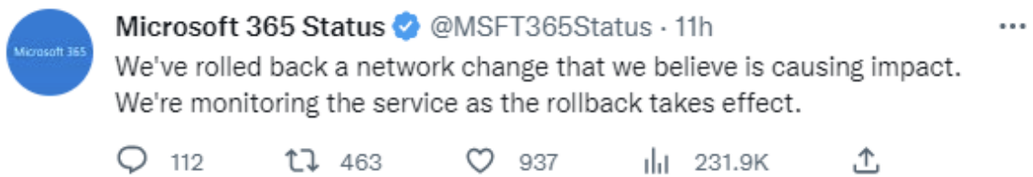


January 25th MS Twitter updates

Over Twitter, Microsoft kept users informed as they troubleshot the issue.

Microsoft 365 Status page message (Twitter)

An hour later, they rolled back the change to mitigate the issue.



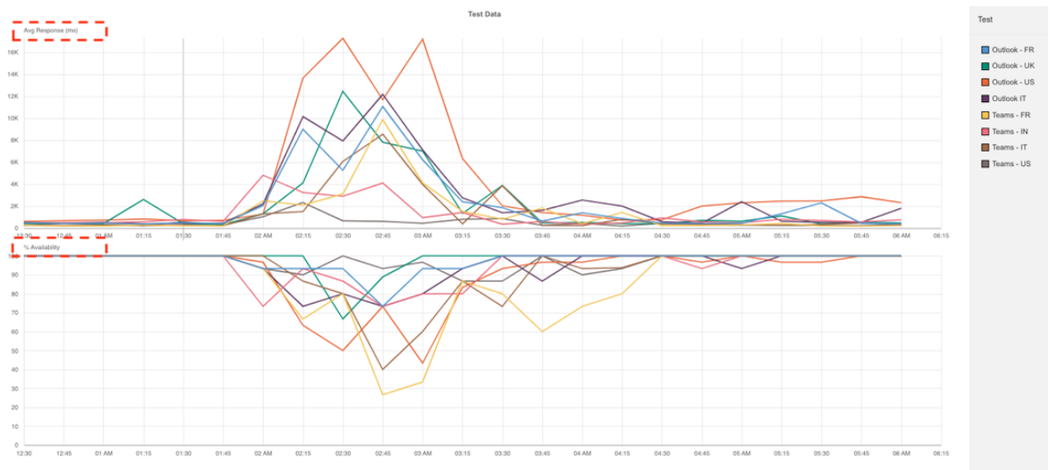
Microsoft 365 Status page message (Twitter)

Five hours later, Microsoft reported on a return to normal for almost all its services.

Microsoft 365 Status page message (Twitter)

Immediate detection by Catchpoint's IPM platform

Catchpoint's IPM platform detected the issue as soon as it started at Jan 25, 2023 - 07:08 UTC/02:08 EST, for Bing, Teams and Outlook applications. Thanks to our continually growing global observability network, we were able to observe an increased connect, wait times, and an availability drop across the world.



Performance graph screenshot showing increased response time and a direct impact to availability (Catchpoint)

The test failure was caused by a test timeout i.e., sending a request for a file and not getting any response for c.30 seconds. The first type of errors users saw would have been a 503 status code; below, the Response Header from Catchpoint's tests indicate a 503 Service Unavailable error for Microsoft's Search engine, Bing.

Request

```
GET / HTTP/1.1
Host: www.bing.com
User-Agent: Mozilla/4.0 (compatible; Catchpoint)
Accept-Encoding: deflate, gzip, br
Accept: */*
Accept-Language: en-us
Connection: Keep-Alive
```

Response

```
HTTP/1.1 503 Service Unavailable
Cache-Control: no-store
Content-Length: 13269
Content-Type: text/html
X-Azure-ExternalError: 0x80072ee2,OriginTimeout
X-MSEdge-Ref: Ref A: EA71E19CA34043A5B6AABB1B39C49378 Ref B: BY3EDGE0111 Ref C: 2023-01-25T08:05:4
Set-Cookie: _EDGE_S=F=1&SID=3B195E9324B8631738234C3725F462CB; path=/; httponly; domain=bing.com
Set-Cookie: _EDGE_V=1; path=/; httponly; expires=Mon, 19-Feb-2024 08:05:49 GMT; domain=bing.com
Set-Cookie: MUID=28BABB3DB297620127EDA999B3DB63F1; samesite=none; path=/; secure; expires=Mon, 19-
Set-Cookie: MUIDB=28BABB3DB297620127EDA999B3DB63F1; path=/; httponly; expires=Mon, 19-Feb-2024 08:
Date: Wed, 25 Jan 2023 08:05:49 GMT
```

503 Service Unavailable Error (Catchpoint)

Fortunately for the US and most of EMEA, the outage largely occurred during the early hours of the morning before anyone had started their working day.

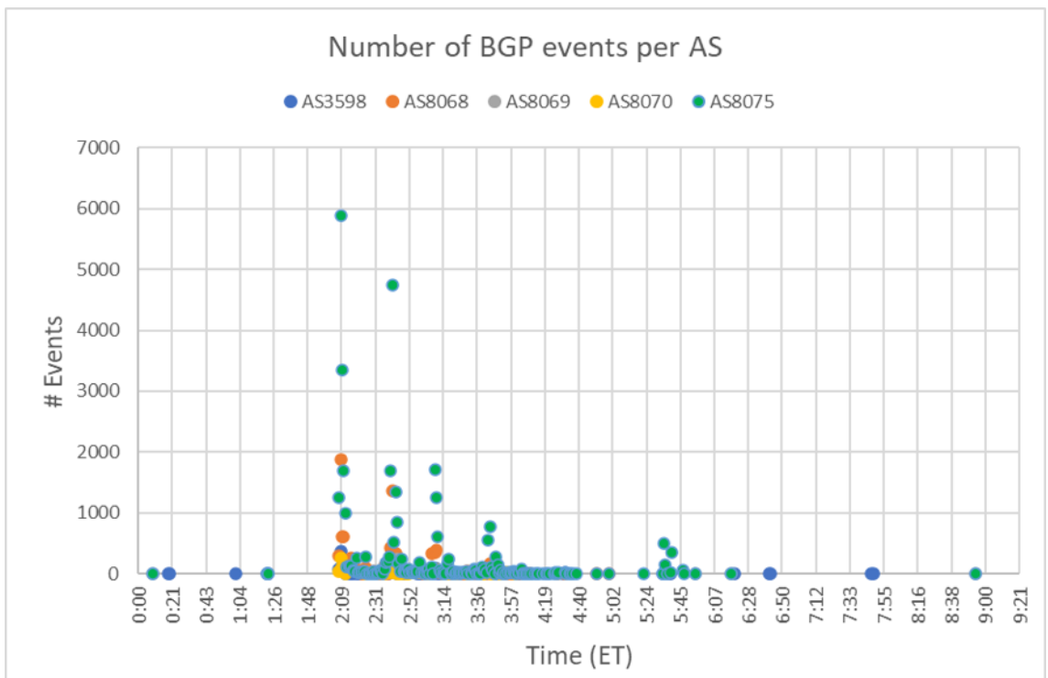
At Catchpoint, however, as for many global businesses, we have a large percentage of our workforce based in India (where it was early afternoon) and our employees bore out their initial confusion at finding core Microsoft services unavailable. One of our QA Managers in Engineering reported Teams being almost impossible to use for two and a half hours, feeling disconcerted on seeing "Status Unknown" on colleagues' statuses, and being abruptly ejected from a Teams call then unable to rejoin.

However, once alerted to the outage by our Managed Services team who were observing the telemetry our Internet Performance Monitoring (IPM) platform was detecting, she was able to switch over to the mobile version of Teams and could happily resume the call.

Likewise, one of our System Administrators found that his Outlook had simply stopped working. He wasn't receiving any emails (including no doubt concerns about the outage itself), nor could he send them. When he became aware of the data Catchpoint was unearthing, he too understood there was a global outage and instead of switching off until service resumed, began to proactively communicate with colleagues to find workarounds until service normalized.

Now, the networking aspect (and supporting BGP data)!

Our telemetry matched what Microsoft announced. However, in addition, we decided to dig into the BGP data since it's the natural starting point for a networking-based investigation and Microsoft said it was a networking/WAN issue. At Catchpoint, we collect BGP data in real-time from more than 50 partners distributed all over the globe – and the number is growing! From a quick look at the BGP events collected by a few of our peers, it is easy to identify the exact moment when the problem arose (and probably when Microsoft's customers started complaining). In the graph below, we analyzed all the BGP updates generated by our peer selection, which were carrying a route towards networks originally announced by one of the 5 AS numbers (ASNs) mostly used by Microsoft.



Number of BGP events per AS used by Microsoft, as seen from Catchpoint peers (Catchpoint)

As you can see, there was a spike in BGP events starting at 02:08AM ET/07:08 UTC , where the routing activity of each of the 5 ASNs spiked upwards. This was recorded by all the peers we analyzed. Most of the events recorded were network announcements, but we also caught several network withdrawals – meaning that any service running behind the withdrawn networks were not reachable from the selected peers – and likely by a lot more people around the world.

To confirm that, we can double-check the telemetry that Catchpoint produced across the duration of the outage, which revealed many connection timeouts, as can be seen in the following Waterfalls and Sankey charts.

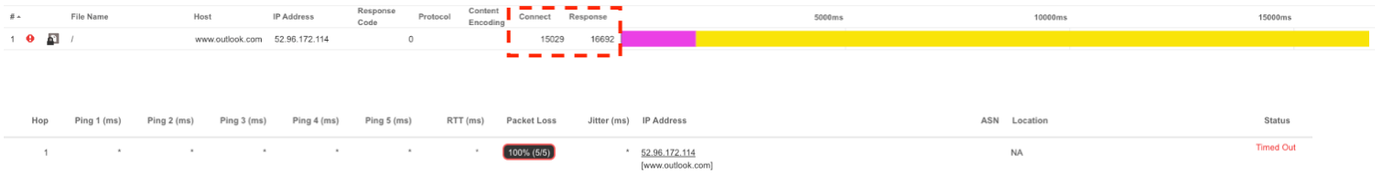


Fig 1 Waterfall showing high connect time and Ping packet loss (Catchpoint)

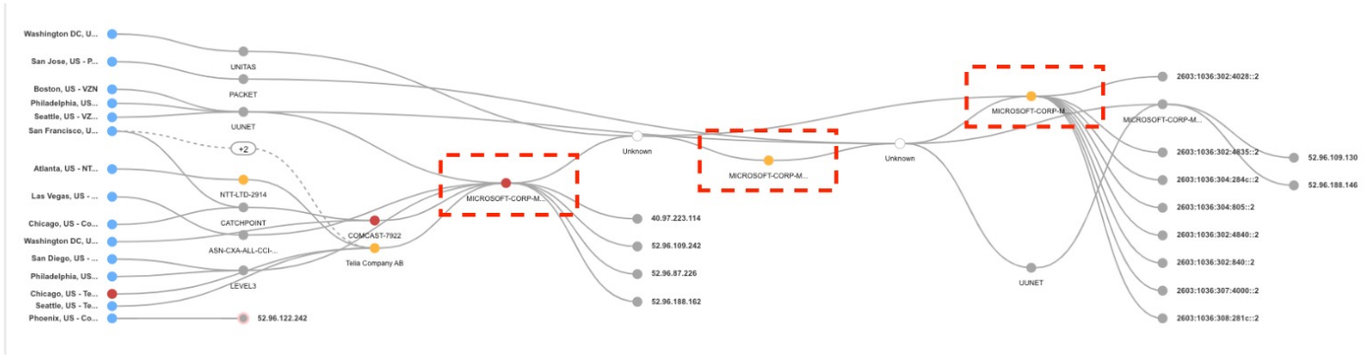
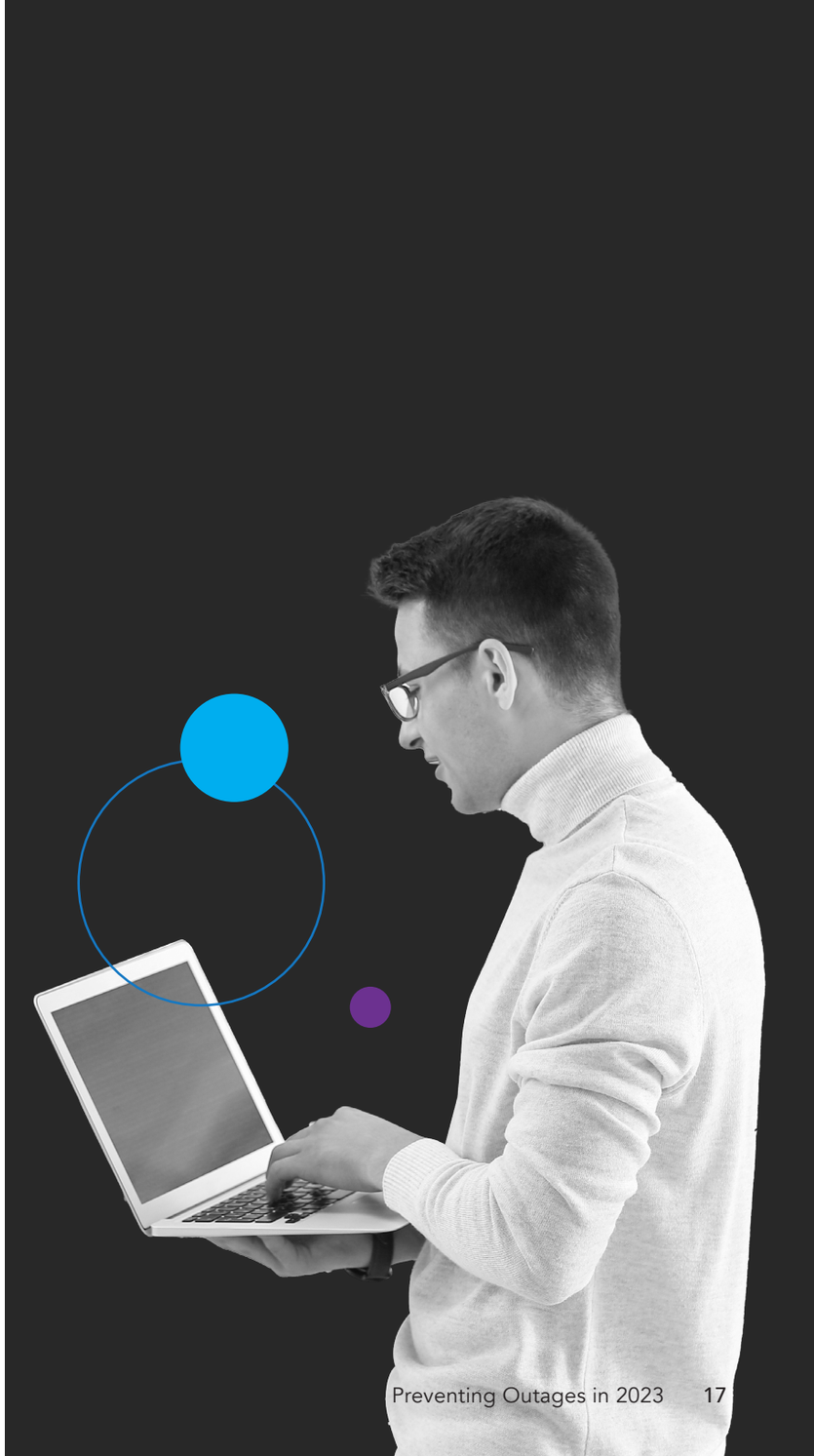


Fig 2 Hop by hop visualization from the Catchpoint Smartboard showing Packet loss at Microsoft managed hops across the US (Catchpoint)



Our Synthetic Traceroute tests were actively monitoring the network from our employees to Microsoft on a continuous basis. This allowed us to track the impact of the outage before and after on a global basis, as is demonstrated in the following three Sankey charts.

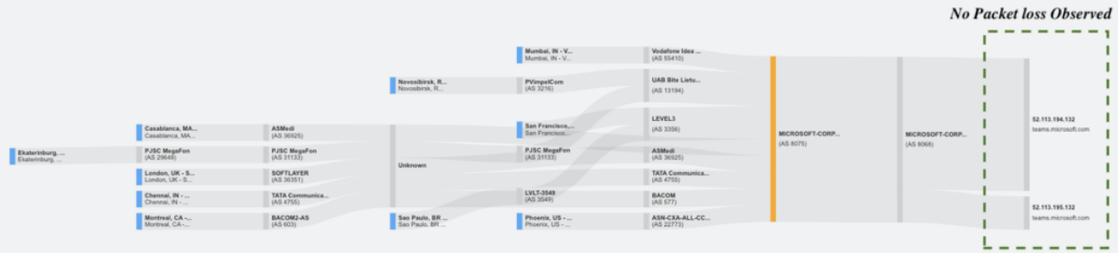


Fig 3 Before the outage: Our data shows no packet loss for the Microsoft endpoint destination IP (Catchpoint)

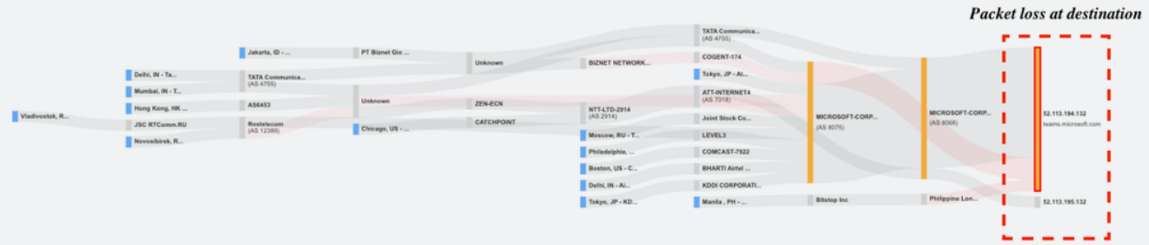


Fig 4 During the outage: We can see extensive packet loss at the Microsoft endpoint destination IP (Catchpoint)

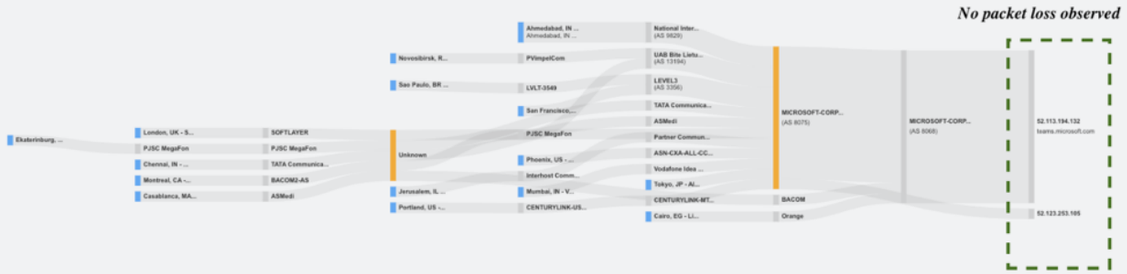


Fig 5 After the outage: We no longer see any packet loss for the Microsoft endpoint destination IP (Catchpoint

From Figure 1, we can see that the test failures were due to a connection time out. Most monitoring tools would only indicate there is a connection issue, but as you can see, Catchpoint helps network teams determine at which level the issue is happening. Figure 2 clearly indicates global packet loss at hops managed by Microsoft. Figures 3 to 5, meanwhile, show that before and after outage, there is no packet loss whereas during the outage we see increased packet loss for the destination IP. During a critical outage, this kind of granular information helps network teams to quickly narrow down the root cause of the issue.

In today's competitive landscape, organizations can't afford to see workforce productivity levels come to a standstill by not being able to send instant messages, emails, or accessing their key documents. This also means hundreds of calls to the Help Desk team from employees to understand the situation. Without a proper monitoring tool, it takes a considerable amount of time for the helpdesk teams to even realize it is not their fault and offer clarity and workarounds to their workforce.

Initially, based on the patterns observed (503 errors turning into connection timed out errors), we suspected this was a network outage caused by changes following a services failure. However, the same pattern is generated when planned network changes, moving traffic from one pod/datacenter to another, by stopping application services at a location and routing users to a different location, which - based on Microsoft's initial postmortem - is what happened here. **This is why analyzing telemetry data is extremely important: know your architecture and evaluate the difference caused by any changes performed before and during the incident.**

Our six key takeaways to mitigate the damage

There are various ways to mitigate the damage caused by this type of extensive outage, depending on whether you are the unfortunate service provider experiencing it or its downstream, inadvertent casualty. After all, what happened to Microsoft's expert professional network engineers can happen to any one of us! Our six key takeaways:

1. Communicate, communicate, communicate – when an outage occurs, be prepared to tell your workforce and customers what is going on as quickly as possible so they can find workarounds.
2. Make sure your communications are fast and accessible for every user. Not everyone will go on Twitter or in this instance, be able to access the MS admin page with more detailed notes. As a matter of course, service providers should route people to a down page that is on a different infrastructure from the core service so that ALL your users know exactly what is happening and won't blame it on their own WiFi or ISP and experience further disruption (and anxiety) as a result.
3. For service providers and those reliant on them, implement an IPM strategy that allows you to monitor precisely what your distributed employees and global customers are experiencing 365/24/27.
4. Monitor the entire Internet Stack you rely on to deliver your content (including DNS, CDN, ISP, BGP, TCP configuration, SSL, and other cloud services, etc.), even if you assume it is out of your control.
5. Be prepared for outages by developing runbooks (for example, switching to a backup solution) and practicing recovery.
6. Whenever change is scheduled, ensure your team is ready for any outages that may occur (intentionally or not) with a crisis call plan that includes a communication plan and templates, a plan to mitigate failures from third-parties, and a best practices monitoring and observability plan.

In its PIR, Microsoft acknowledged its own takeaways:

How are we making incidents like this less likely or less impactful?

- *We have blocked highly impactful commands from getting executed on the devices (Completed)*
- *We will require all command execution on the devices to follow safe change guidelines (Estimated completion: February 2023)*

Published on January 26th, 2023 (Updated on February 2nd, 2023)

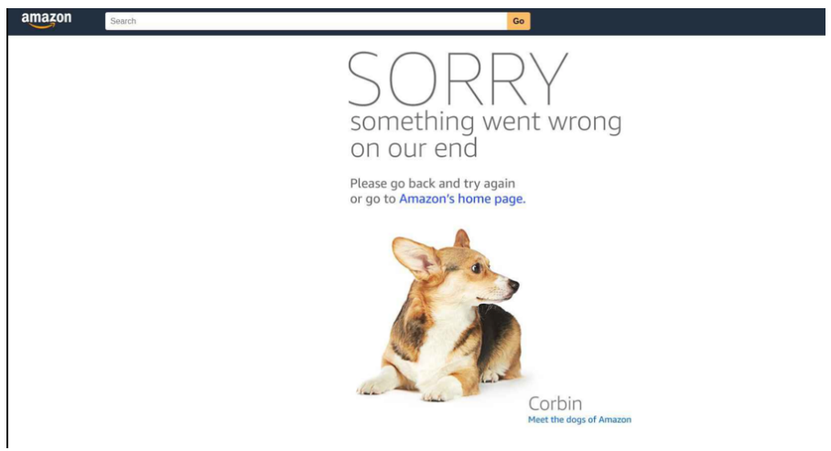
Incident Review: December 5-7, 2022 – How Catchpoint's IPM Platform Detected Amazon's Two-Day Search Issue

By Nilabh Mishra

Not all Internet outages take a website down. Some may impact a smaller subsection of users or only affect one part of a site's functionality. Moreover, because of their relative "hidden" nature, organizations may not always know about them immediately since fewer users will be making complaints. However, such incidents can still have serious consequences, thus you want to detect them as soon as possible so you can quickly mitigate and resolve issues.

Detection isn't as easy as it sounds. Many sites rely on basic uptime monitoring— sometimes limited to just monitoring their home page—to detect slowdowns and outages, which can mean that a company experiencing intermittent or partial site failures misses detection.

Earlier this month, through our ability to conduct transaction monitoring, we detected just such an issue at Amazon.



The outage message impacted users saw (Amazon)

Detecting Amazon's Search failures

Catchpoint systems began to detect initial failures related to Amazon's Search function at 12:51:02 ET on December 05, 2022. As you can see below, these were intermittent in nature.

Amazon test data (Catchpoint)

The errors lasted for 22 hours until December 7, 2022 – 11:13:44 ET, remaining intermittent, yet continuing to impact users worldwide wishing to search for products on Amazon's desktop and mobile sites. According to our synthetic data, around 20% of worldwide users were impacted across the entire time frame. For a certain percentage of the users, Search was completely down and unusable for the entire 22 hours. End users were greeted with the error message above when trying to search for items to buy on the site.

Any kind of negative user experience like this (we detected a similar incident several years ago) can have a serious impact on brand reputation and ultimately, the bottom line of the affected company.

Did you know...

12 to 1	88%	1 Second
It takes 12 positive user experiences to make up for one unresolved negative experience.	Of web visitors are less likely to return to a site after a bad experience.	Delay in page-load can cause a 7% loss in customers.

Which layer of the Internet Stack was responsible?

Catchpoint's Internet Performance Monitoring platform was able to identify precisely which layer of the Internet Stack was responsible for the issue. In this case, from looking at the headers (see below), we were quickly able to confirm the problem was the result of an HTTP 503 being returned by Amazon CloudFront.

HTTP/2 503

accept-ranges: bytes

content-encoding: gzip

content-length: 1203

content-type: text/html

date: Wed, 07 Dec 2022 11:43:59 GMT

etag: "a6f-5ebe10a81c900-gzip"

last-modified: Tue, 25 Oct 2022 19:37:40 GMT

server: Server

strict-transport-security: max-age=47474747; includeSubDomains; preload

vary: Content-Type,Accept-Encoding,User-Agent

via: 1.1 e9c14790ff63b931e8e58434b28761a0.cloudfront.net (CloudFront)

x-amz-cf-id: SMiU34GSY5nJUG2-Vokc40wC3fhVdtQ9bvSzIvIaj6FoYaF08y28eg==

x-amz-cf-pop: LAX50-P3

x-amz-rid: Y0C84P3AADKT2S5BETKJ

x-cache: Error from cloudfront

Rapid detection of the problem, along with the ability to rapidly pinpoint root cause, lets you see and troubleshoot interruptions as they occur. The issue, which ultimately impacted a fifth of Amazon global users (imagine the impact on global revenue), could have been resolved far more quickly with an IPM solution.

Challenges with the golden triangle of observability

In recent years, there has been a great deal of focus in APM on "the golden triangle of observability": logs, traces and metrics. One major problem with the kind of approach that focuses solely on the golden triangle is the delay introduced in detecting intermittent incidents.

The golden triangle is amazing, but if it's all that we are dependent on when trying to detect issues early, it suffers from a few major shortcomings.

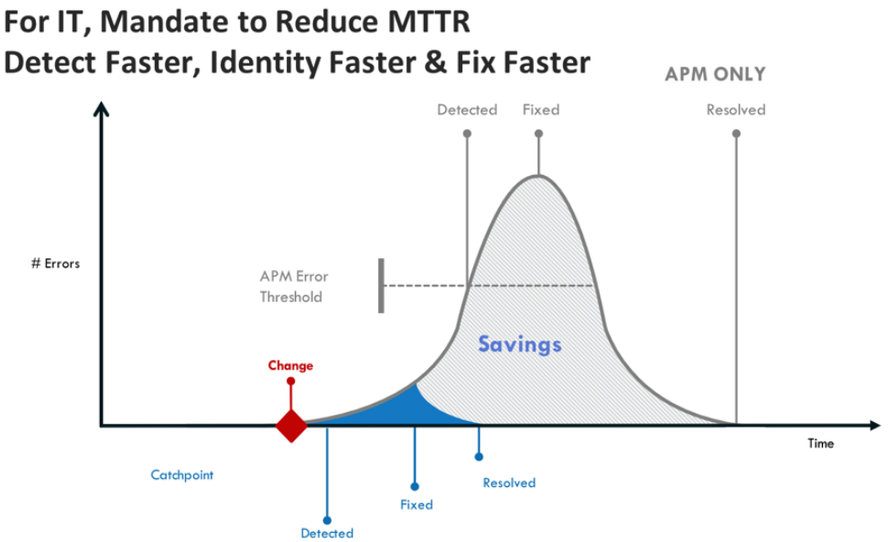


Diagram showing the delay in detecting issues caused by using only APM (Catchpoint)

The most important of all is the delay we see in detecting issues that are caused by a high error threshold.

Most of the systems that depend on logs and traces to detect problems have error thresholds set as high. This is largely to avoid false positives, especially in a situation where the number of hits is high.

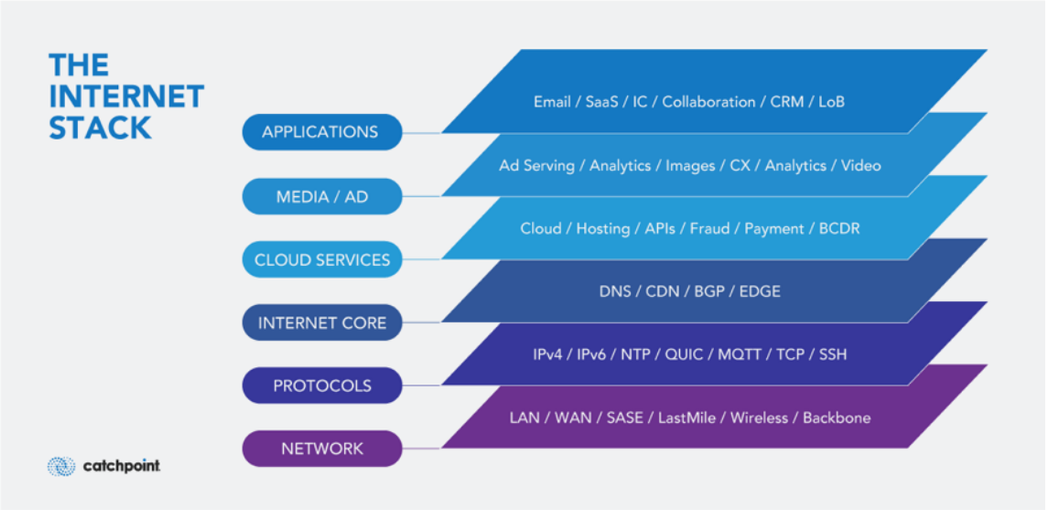
However, if we look at the diagram above, we can see how high thresholds set on the APM side can result in delays in detection of issues post a change (for example, increased errors due to a bad deploy).

Why IPM is important here

This is exactly when a strategy built around Internet Performance Monitoring, specifically proactive targeted monitoring, becomes important because the thresholds set are lower, allowing us to:

1. Detect issues faster and at a much earlier stage – often before actual end users are impacted.
2. Detect intermittent issues that may not be impacting 100% of traffic (this was the case with Amazon Search).

Internet Synthetic Monitoring is extremely targeted and allow us to set tighter thresholds around the myriad components of the Internet Stack:



The Internet Stack (Catchpoint)

Why you need a best-of-breed monitoring strategy

When it comes to outages like these, it is extremely important to have a monitoring strategy in place which makes use of best-of-breed solutions that are targeted to perform specific activities, including:

- IPM, which includes:
    1. Proactive Monitoring to detect issues quicker, understanding which layer of the Internet Stack the problem is originating from, and which functionality is impacted.
    2. Reactive Monitoring to ascertain the impact from a business standpoint (pageviews, revenue, conversions, and geographic impact).
- APM (logs, traces and metrics) to better understand the internal components of the system which may be impacted.

As Steve McGhee, Reliability Advocate, SRE, Google Cloud, highlighted in his Conclusion for Catchpoint's 2023 SRE Report, there is a reason why experts never depend on a single solution, tool or platform to accomplish their tasks in the best possible manner. *"When it comes to skilled labor, or 'operations' perhaps,"* writes Steve, "*you want teams to be able to reach for the right tool at the right time, not to be impeded by earlier decisions about what they think they might need in the future."*

Published on December 21, 2022

"What's so bad about cascading failures? They take down your whole service, most of the time, because the effects of them spread, unless you have some sharding approach, perhaps. They don't self-heal. Once you're in this cycle, it stays that way until you intervene. You don't really get warning of them. You can think you're fine, everything looks healthy, then you're on that cliff edge, and you just step over it."

~Laura Nolan, SRE, Slack

Incident Review: August 2022 – 3 Lessons from a DNS Resolution Failure Incident

By Alessandro Improta, Anna Jones and Dritan Suljoti

The Domain Name System sits at the center of the internet. At Catchpoint, monitoring DNS is a core use case for many of our customers.

How we troubleshot a DNS resolution failure

We recently ran into a reachability issue experienced by one of our eCommerce customers (a billion-dollar company in the S&P 500, headquartered in the US) who relies on a third-party DNS provider. The eCommerce company was experiencing issues around the DNS authoritative name servers they were relying on to resolve a critical page on their website. We detected the issue at multiple Catchpoint backbone nodes distributed across the US and found SERVFAIL errors occurring, seriously impacting website reachability.

We drilled down into the problem, looking across the entire resolution chain to pinpoint where the problem occurred and then shared that granular DNS service data with the customer so they could take it to their DNS provider to resolve the issue.

In this article, we identify the three key observability lessons that can be taken away from this incident.

1) Monitor the key CNAMES behind or in front of your domains

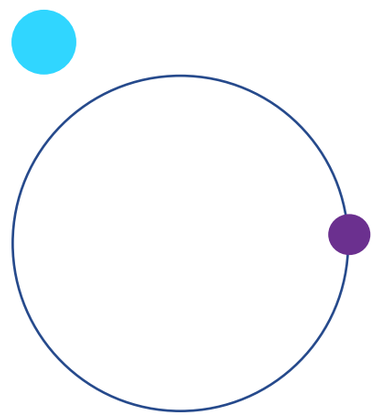
Often companies rely on third-party SaaS providers for key aspects of their digital presence. In this case our customer relied on a third-party SaaS provider, and its domain simply had a CNAME to it.

The following response leads the resolver to the CNAME [nameofcompany].phenompeople.com, which is resolved querying the GTLD authoritative name server (10-11) – identified now as d.gtld-servers.net (192.31.80.30) – and then the Phenompeople domain authoritative name server (12 and 15) – identified as pdns09.domaincontrol.com (97.74.110.54).

This is where DNSSEC comes into play.

DNSSEC is a feature introduced by IETF to "*... provide origin authentication and integrity protection for DNS data, as well as a means of public key distribution*" (cit. RFC4033). It consists of a sequence of DNS requests and replies aimed at verifying the authenticity of the records being returned using cryptography (via RRSIG and DNSKEY records) and a chain of trust (via DS records).

As you can see from the following snippet, the latter response of the GTLD server contains a RRSIG record and two DS records – meaning that phenompeople.com is signed and the resolver is supposed to perform the DNSSEC validation.



```
> Frame 11: 527 bytes on wire (4216 bits), 527 bytes captured (4216 bits)
> Ethernet II, Src: Cisco_20:5d:84 (18:8b:9d:20:5d:84), Dst: VMware_2d:de:c7 (00:0c:29:2d:de:c7)
> Internet Protocol Version 4, Src: 192.31.80.30, Dst: 10.116.40.5
> User Datagram Protocol, Src Port: 53, Dst Port: 63996
v Domain Name System (response)
    Transaction ID: 0xc5ba
  > Flags: 0x8010 Standard query response, No error
    Questions: 1
    Answer RRs: 0
    Authority RRs: 5
    Additional RRs: 5
  > Queries
  v Authoritative nameservers
    > phenompeople.com: type NS, class IN, ns pdns09.domaincontrol.com
    > phenompeople.com: type NS, class IN, ns pdns10.domaincontrol.com
    > phenompeople.com: type DS, class IN
    > phenompeople.com: type DS, class IN
    v phenompeople.com: type RRSIG, class IN
        Name: phenompeople.com
        Type: RRSIG (Resource Record Signature) (46)
        Class: IN (0x0001)
        Time to live: 86400 (1 day)
        Data length: 183
        Type Covered: DS (Delegation Signer) (43)
        Algorithm: RSA/SHA-256 (8)
        Labels: 2
        Original TTL: 86400 (1 day)
        Signature Expiration: Aug 10, 2022 07:12:07.000000000 W. Europe Daylight Time
        Signature Inception: Aug  3, 2022 06:02:07.000000000 W. Europe Daylight Time
        Key Tag: 32298
        Signer's name: com
        Signature: 529e4af8d6b34e1f002ab4fb7135d0d4a6450e0ea7d587258f96689eadd6a0cecf2a3bbe…
  v Additional records
    > pdns09.domaincontrol.com: type AAAA, class IN, addr 2603:5:21e2::36
    > pdns09.domaincontrol.com: type A, class IN, addr 97.74.110.54
    > pdns10.domaincontrol.com: type A, class IN, addr 173.201.78.54
    > pdns10.domaincontrol.com: type AAAA, class IN, addr 2603:5:22e2::36
    > <Root>: type OPT
    [Request In: 10]
    [Time: 0.021024000 seconds]
```

This is correctly performed by the resolver, which then queries one of the Phenompeople domain authoritative name servers – now identified as pdns10.domaincontrol.com (173.201.78.54) – to request for its DNSKEY record, to proceed with the DNSSEC validation.

2) Monitor TCP not just UDP

And here comes the problem.

The answer provided by pdns10.domaincontrol.com is too long to be handled in UDP, hence the authoritative name server replies with a message with the TC (Truncated) flag being set, as you can see from the following snippet.

```
> Frame 14: 1495 bytes on wire (11960 bits), 1495 bytes captured (11960 bits)
> Ethernet II, Src: Cisco_20:5d:84 (18:8b:9d:20:5d:84), Dst: VMware_2d:de:c7 (00:0c:29:2d:de:c7)
> Internet Protocol Version 4, Src: 173.201.78.54, Dst: 10.116.40.5
v User Datagram Protocol, Src Port: 53, Dst Port: 42727
    Source Port: 53
    Destination Port: 42727
    Length: 1461
    Checksum: 0xc4c5 [unverified]
    [Checksum Status: Unverified]
    [Stream index: 4]
  > [Timestamps]
    UDP payload (1453 bytes)
v Domain Name System (response)
    Transaction ID: 0xa90b
  v Flags: 0x8610 Standard query response, No error
      1... .... .... .... = Response: Message is a response
      .000 0... .... .... = Opcode: Standard query (0)
      .... .1.. .... .... = Authoritative: Server is an authority for domain
      .... ..1. .... .... = Truncated: Message is truncated
      .... ...0 .... .... = Recursion desired: Don't do query recursively
      .... .... 0... .... = Recursion available: Server can't do recursive queries
      .... .... .0.. .... = Z: reserved (0)
      .... .... ..0. .... = Answer authenticated: Answer/authority portion was not authenticated by the server
      .... .... ...1 .... = Non-authenticated data: Acceptable
      .... .... .... 0000 = Reply code: No error (0)
    Questions: 1
    Answer RRs: 5
    Authority RRs: 0
    Additional RRs: 1
  > Queries
  > Answers
  > Additional records
    [Request In: 13]
    [Time: 0.000544000 seconds]
```

Consequently, the resolver triggers the very same query to the very same server but now using the Transmission Control Protocol (TCP). However, the TCP connection is torn down by the server. See packet number 20 in the following snippet.

We saw that the same happens when we manually performed a dig related to DNSSEC from that very same machine to the same authoritative name server.

```
[root@us-wa-sea-te-01 ~]# dig @173.201.78.54 -t DNSKEY phenompeople.com +dnssec
;; Truncated, retrying in TCP mode.
;; communications error: end of file
;; communications error: end of file
```

In this case, we can see from the following tcpdump trace that the server replied (5) with an initial TCP packet of size 1460 – exactly the value of MSS being negotiated in the TCP handshake (2), but before sending the second TCP packet containing the trailing data of the DNS reply, the server decided to tear down the TCP session (7).

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	10.116.40.5	173.201.78.54	TCP	76	36053 → 53 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=1102129646 TSecr=0 WS=128
2	0.000455	173.201.78.54	10.116.40.5	TCP	68	53 → 36053 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460 SACK_PERM=1 WS=64
3	0.000470	10.116.40.5	173.201.78.54	TCP	56	36053 → 53 [ACK] Seq=1 Ack=1 Win=29312 Len=0
4	0.000514	10.116.40.5	173.201.78.54	DNS	103	Standard query 0x0bd4 DNSKEY phenompeople.com OPT
5	0.002120	173.201.78.54	10.116.40.5	TCP	1516	53 → 36053 [ACK] Seq=1 Ack=48 Win=29120 Len=1460 [TCP segment of a reassembled PDU]
6	0.002127	10.116.40.5	173.201.78.54	TCP	56	36053 → 53 [ACK] Seq=48 Ack=1461 Win=32128 Len=0
7	0.002131	173.201.78.54	10.116.40.5	TCP	62	53 → 36053 [FIN, PSH, ACK] Seq=1461 Ack=48 Win=29120 Len=0
8	0.002298	10.116.40.5	173.201.78.54	TCP	76	37676 → 53 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=1102129648 TSecr=0 WS=128
9	0.002330	10.116.40.5	173.201.78.54	TCP	56	36053 → 53 [FIN, ACK] Seq=48 Ack=1462 Win=32128 Len=0

3) Monitor for Anycast, not just for your cloud

Please note that the very same query performed from Italy or from Las Vegas leads to a proper reply. This is due to the managed DNS provider relying on Anycast for their name servers and only a subset of their locations having DNS servers not configured to properly handle DNS queries over TCP.

```
[bgpadmin@qa-cos-ice04 ~]$ dig @173.201.78.54 -t DNSKEY +dnssec phenompeople.com
;; Truncated, retrying in TCP mode.

; <<>> DiG 9.11.4-P2-RedHat-9.11.4-26.P2.el7_9.8 <<>> @173.201.78.54 -t DNSKEY +dnssec phenompeople.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 48856
;; flags: qr aa rd; QUERY: 1, ANSWER: 5, AUTHORITY: 3, ADDITIONAL: 1
;; WARNING: recursion requested but not available

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags: do; udp: 1472
;; QUESTION SECTION:
;phenompeople.com.              IN      DNSKEY

;; ANSWER SECTION:
phenompeople.com.       3600    IN      DNSKEY  257 3 8 AwEAAej9dUXd8+VEJa0D08TSRhTlCaSUW7npXO4nL1muPpNDU
7kLVs18 rlmyemwF4+Bmcbl2cZgyysPxknw+p4nKY2Ug4tx3Jie6uchf52qKe2mR gSHt2SDwST+goUr4UR4CtiB3Otc3IuCojOfLnYQe
SrpjDwloRqxWMvkn yEQapJNZR/5X1CYhVAMbC3w26vmjQ81yyellrYPeGLCIWGOrGl3H8+JF I2/3yf5opvATp6VyepahBq0z0dTSZWM
IpptUZz9FItThLxuORxFizpa/ W8g2M5SUfHneOEWgnGT1GKk3oLXkrYvPxezc69BBjln3/IMbk/w29/hT xnEWEKSUXeM=
phenompeople.com.       3600    IN      DNSKEY  257 3 8 AwEAAalTjdNxvVdvyP0r7+U7a0g1ELcfZptfjmQM6EleyEyAQ
tFMWUAE WjFwul0+MAjIYc9g8QJ9ih5KFALsKUm2suGM4YSyjBH6JBuGCy0hBUZD DvH5BQ35OPqSA1jK35jO6dQaWmhOq9/+mD0daA2s
Ycrgy1EbYr4hO+u0 9STuOWUiv2jD7rZHjySInVYb260Sx2mdzwAgI7Q1xAVOQXELhwdsxQSu R6JPm+SjCoN7q9yOqfvHyFr11DLgB6y
ab02TES5R2bl2ZtHImmDVBadx XYenxLjMSbMICMvESJA2GmwPMLVRrFRuz4mbSYFQYQi9mGYQ7z6BkpZn H70Mo9vpBvs=
phenompeople.com.       3600    IN      DNSKEY  256 3 8 AwEAAapGXBzA4u9v9C16lrQdR2J5it1XcR8TiUQi4SQxSGJaJ
Dm6W4zv H+O+fQ+IEeD7/D9Xm1a/6qQ51qACAX51/6gURF7fahl3gZEH4HRxQSIY 5+xJprfLYSHNHiPj1LgHiF7AbAaNiyUCDPU+pcC/
PKy47ZULRmrUMzZG K29mA3BKN9RIhuIhZSuzfZIxNlY56G6I9qzgjg6WxLqW1G04G0Q09cdY wt3GZP8PlZ3LoMcBtlcvhyOvi9jFyNN
PnwX0KlstljW3SgxJmYFNKw5G eizvxctg2PaG7VV3FaRVc3pjXTUUo2ay5v/v9CLLF4GDxcJspXZ5YpRA gK4UquCzU90=
phenompeople.com.       3600    IN      DNSKEY  256 3 8 AwEAAfQkxk35lLZJG0EuwKXIl6fJATXaekUv8u+Sc3O516ctw
tblIgKl agfGxE1MA21t/VUjQyzRIkNUmQWV22EyfNFe3c0SOUME+7Gt19PN4Z2z IIZ0jT6q59xyEqfiI/qpVy6gts+tyzpSi3FAIOmo
fUemHsRWFrNUyOnP OjpkEmGfhEETgZ6QpSMhjjz89pASMqyyu6DtOw64/dbTVQA8uMMPFc9N IEPTA+JGJvWGMDzKQPZYciKVQjSe7Zr
LLxwePsD9m8QvWxrdkUGCB+cl Dr4Z/Kkwz4ptZa8jVmKxjcCs0hBDJmyBhJwCE1wB8dOi48D3hSRBileD via0qiRXn6U=
phenompeople.com.       3600    IN      RRSIG   DNSKEY 8 2 3600 20220822103535 20220807103535 64477 pheno
mpeople.com. 3jnSxyNQPEKk7AFAzhl5aVWaYukpciXW2jKXJiy0P2G3HYZTrAXvsxQ7 LKvDhba2MfgU2YHSXBskTly+PC4PUbjFqjf
LzyTxEuWTFiep7WqUoXfP XxeyAofHzVPe4GeL4nBx7fS/gCrTNX4cUIlRvGbiy+Iwift1pgGY40Ks QJBSbn11mnpbzU4fd/5c9Ykbf9
9PZ6TK8qL3bV7LdMFohHXGEqS1Bxb2 STAkQfTlA6SaBzM1MTdjF/p6Lq6J5lWndVap5wulTuxljDZkmOK/0C8Y BomScpvCEAIHtOzfc
ak2cTL9CX7GnVvK7x6cXPYbVu1sa7lgbGllFiNC JaabXw==

;; AUTHORITY SECTION:
phenompeople.com.       3600    IN      NS      pdns10.domaincontrol.com.
```

We did a few more tests on that name server from one of the nodes showing the issue, and we were able to identify that dig will fail only when the +dnssec option is set, meaning that the DO bit is set in the DNS request – or in other words, the request from dig to the resolver to provide RRSIG records along with the requested records. Hereafter a snippet of the successful dig requesting for DNSKEY in TCP.

We believe that the issue on pdns10.domaincontrol.com is somehow related to the size of the reply being generated. All the replies received from that server on the nodes showing the issue are indeed smaller than the MSS set in the TCP session (1440 bytes). And DNSSEC is known to generate large messages – indeed the size of the reply we received while running dig from Italy was 1813 bytes. As proof of that, we found that the same issue is experienced when running dig from the node seeing the issue with type ANY – which usually leads to large responses (when enabled).

```
[root@us-wa-sea-te-01 ~]# dig @173.201.78.54 phenompeople.com -t ANY
;; communications error: end of file
;; communications error: end of file
[root@us-wa-sea-te-01 ~]#
```

Incident resolution

We recommended that our client contact the SaaS provider in question (the one they had the CNAME to) and ask them to reach out to their managed DNS provider (that operates the impacted DNS servers). Clearly the issue was prevalent and impacting more than just our customer. The domains managed by this DNS provider that had DNSSEC enabled would have pockets of their end users impacted with a complete outage. At the time of writing, the provider has not yet fixed the issue. We hope that armed with the details we shared with them directly, we are able to put an end to this problem and thereby restore access to impacted users and improve overall customer digital experience.

Three take aways

What should you be monitoring in relation to DNS? This incident demonstrates three clear takeaways:

1. **Monitor the key CNAMES behind or in front of your domains.** Your DNS configuration and infrastructure might be fine, but your business relies on SaaS providers which might still have an issue that impacts your users and business. The customer doesn't know if it's your CDN or your SaaS provider having the problem. They just see your brand in the affected domain.

2. **Monitor TCP not just UDP.** DNSSEC failures are not just about signatures, but are the name servers properly handling DNS over TCP? Make sure to monitor them.

1. **Monitor for Anycast, not for your cloud.** DNS is often anycast-ed… don't just monitor from a few locations on AWS and assume the rest of the world can reach you. You need to monitor from as widely dispersed a set of locations as your customers.

When DNS resolution failure happens, you need a scalpel not a sword

Despite its importance in ensuring the resilience and availability of the web, the pivotal role of DNS in translating web domains to the correct IP addresses is often overlooked. Moreover, this foundational technology is highly complex, can be problematic and can lead, for example, to DNS resolution failure which causes an outage to your website, application, or service. Such failures can be difficult to discover and to troubleshoot due to the many layers involved in the network of services that ensure the URL you enter into your browser is the same website you are taken to. If there is a break at any point within this network of services, your website could become inaccessible, or worse, be redirected to another, possibly malicious, site.

If you outsource DNS to a third-party DNS provider, it's even more important to have access to in-depth analysis of the health of your DNS service to ensure it's working as planned and meeting your Service Level Agreement so you can enter into quick remediation when necessary, and/or hold them accountable in instances of failure.

Published on September 15, 2022

Incident Review: February 22, 2022 – Slack Outage – Good Morning! Here's 16 Minutes of Stress!

By Raj Jathar

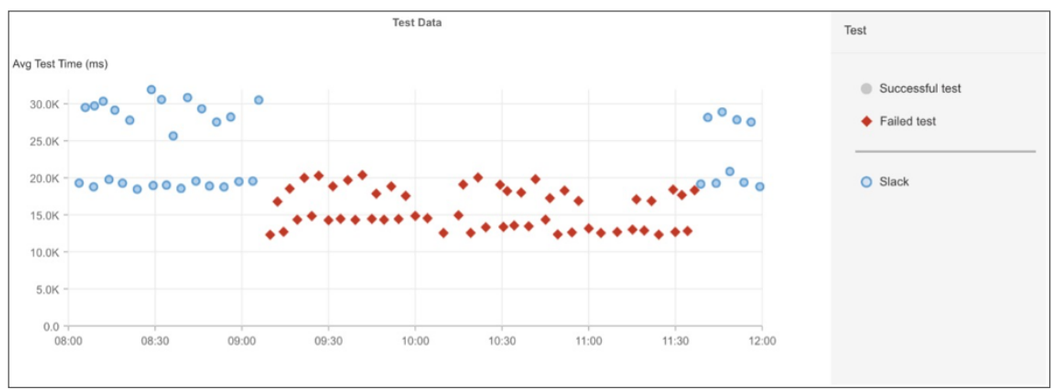
For some time now, people have understood the importance of early warning systems, whether for detecting earthquakes and tsunamis, military defense, or business and financial crises. Why should service providers, especially those delivering software as a service (SaaS,) be any different? In a world where time is money and minutes mean millions, it is vital for organizations to keep a very close eye on the supply and delivery chain of their service to their end users, both business and consumer.

According to Techjury, Slack has more than 10 million daily active users, of which 3 million are paying subscribers who spend an average of 9 hours logged in to Slack each day. So yes, it is important for Slack to fully understand how their service is performing and what their users are experiencing. However, if you're one of the 600,000 companies worldwide that relies on Slack to keep communication flowing throughout the enterprise, it's critical for you to have your finger on that pulse as well. As a customer, the only thing worse than having a productivity application go down is wasting even more time thinking it might be you and trying to fix something you can't.

Watering down their SLA

This is not without precedent. In 2019, according to TechTarget, Slack watered down its cloud service level agreement (SLA) after outages forced the vendor to issue $8.2 million in credits in a single quarter. "Compounding the financial impact of the downtime was an exceptionally generous credit payout multiplier in our contracts dating from when we were a very young company," Slack's finance chief, Allen Shim, told analysts on a conference call at the time. "We've adjusted those terms to be more in line with industry standards, while still remaining very customer friendly." So, while subsequent outages should not have quite the same bottom-line impact, the intangible effects of issues such as damage to the brand, resolution efforts, loss of productivity, and vulnerability to competition remain and are harder to quantify.

Now, looking at it from a customer's perspective: When the service started having trouble yesterday around 9 AM ET, there was no indication of an issue by Slack. However, if you were a user of Catchpoint, you wouldn't need one, as you would have been alerted to it right away (see below).

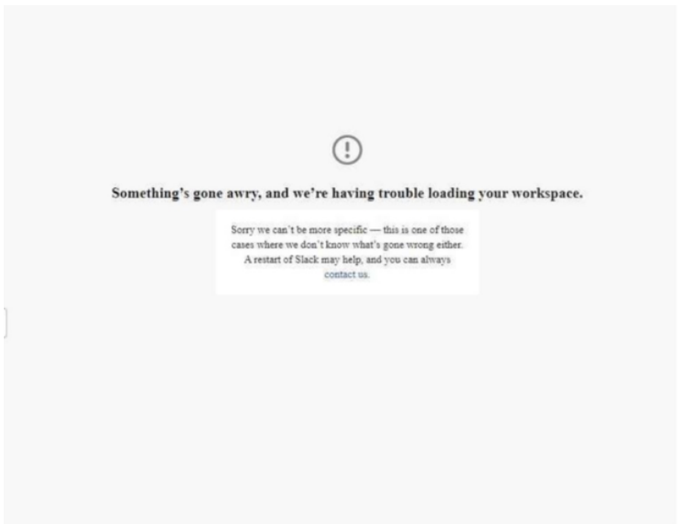


Failures for Slack tests from Feb 22, 2022 starting at 09:09:48 ET (Catchpoint)

Once alerted, the organizational reaction to this event varies by user type and position. If I'm the head of IT at a company that uses Slack, my first step is to check Slack's service site to see what they report: several minutes into the failure there was no update from Slack. In my role, I would need to verify the error before I start to raise the alarm internally.

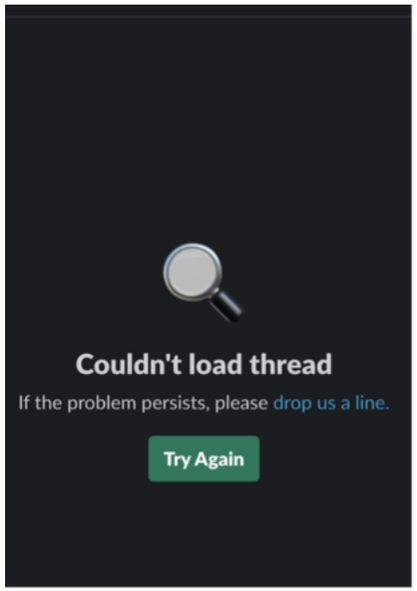
Error notification messages

Fortunately, I don't even have to log in to Slack myself to do that; the Catchpoint tests capture the error notifications that end users see, such as this one right after successfully logging in:



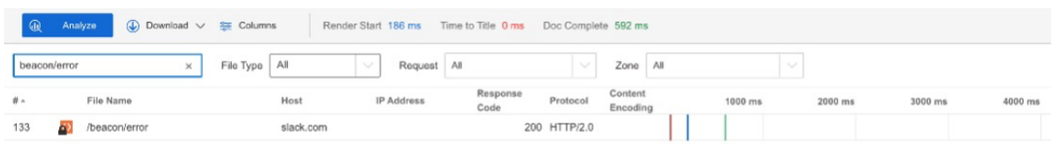
Error notification (Catchpoint)

And this one when users try to fetch conversations:



Error notification (Catchpoint)

Not only that, but Catchpoint waterfall reports can also see the error request on the page, which seems to be posting back the error captured on the application (this request is seen only in failed test runs).



Waterfall chart (Catchpoint)

Houston, we have a problem! At this point I would have successfully navigated the "detect" stage of an incident lifecycle to identify (next stage) the event as a bona-fide incident. Then comes the hard part: triage. Whom do I have to notify – or worse, wake up? Whose breakfast am I am going to interrupt to help diagnose this?

Notifications, notifications: who to tell?

Let's start with the folks who are guilty until proven innocent: the network team. The chart below shows where in the many steps the test is failing:

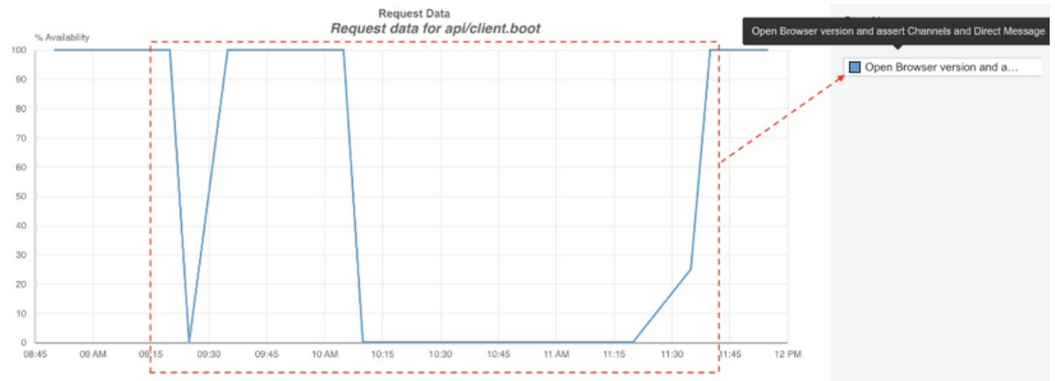
Availability chart (Catchpoint)

Okay, so it's not the network. However, let's make doubly sure and look at what requests are failing; we can do that because with Catchpoint we have the ability to emulate user activity, which shows that users were able to log in and take quite a few actions before getting an error from the Slack servers.

Emulated user activity (Catchpoint)

This incriminates the "<root>/api/client.boot?_x_id=noversion-1645542869.027&_x_version_ts=noversion&_x_gantry=true&fp=e3" request.

The chart below, showing the before and after, points to it being one of the key calls responsible for conversations and messages page functionality.



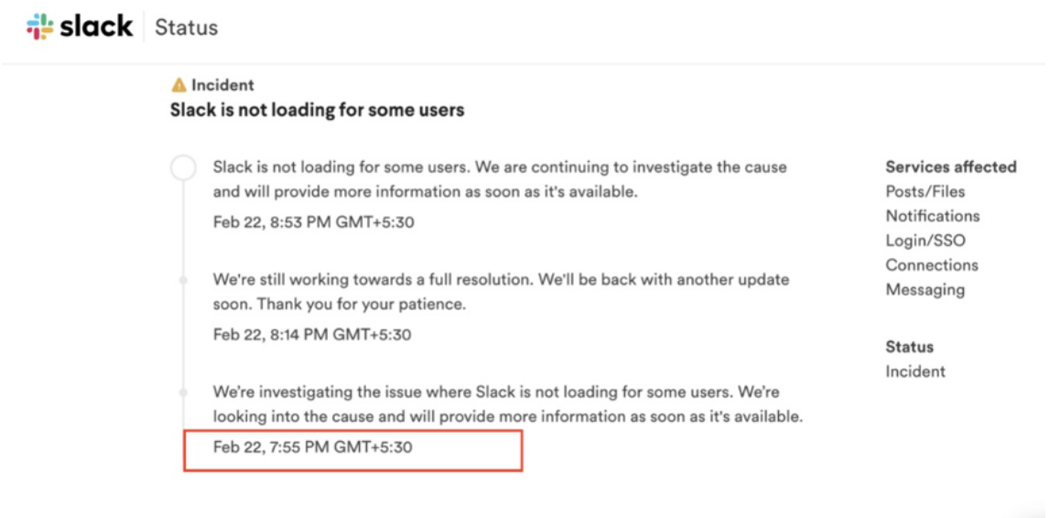
Request data chart (Catchpoint)

Confirmation… I can let the network team go on with their day (they are Mountain Dew guys anyway)!

However, if I'm the head of the Help Desk, right about now I'm spitting out my very hot Dunkin'. I can see that while users are able to log in, they can't do a whole lot else, which means – particularly because this is the first thing in the morning – I'm going to have a whole lot of people fumbling around trying to figure out what's going on and flooding my support staff with tickets and calls! I catch my breath and send out an email letting my user community know that Slack is having an outage and that they should seek other means of communication until further notice.

How many minutes do you have?

I would have been able to do all of this in a matter of minutes, which makes a difference because it took Slack approximately 16 minutes to inform the world about what Catchpoint users would have already known:



Incident status report (Slack)

Can your organization afford to lose 16 minutes of productivity per employee? And have your employees opening tickets or calling the help desk for situations beyond your organization's control? If you're like most enterprises and the answer is no, and you're not already invested in an industry-leading observability solution like Catchpoint, the good news is that it's not too late.

Published on Feb 23, 2022

Incident Review: December 7, December 15 and December 22, 2021 – What Can We Learn from AWS' December Outagepalooza?

By Carol Hildebrand and Raj Jathar

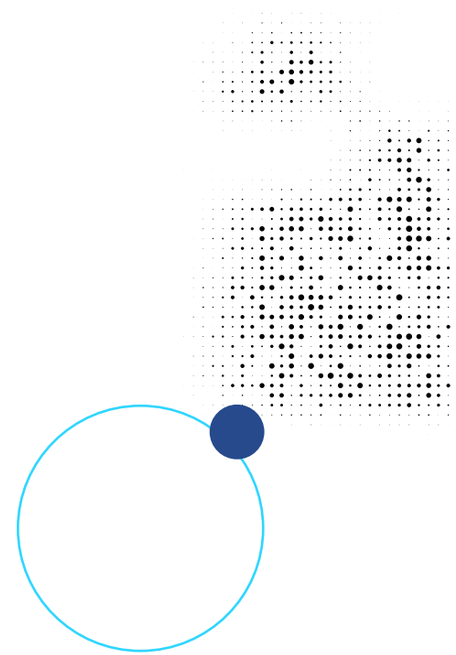
2021's slew of Internet outages or disruptions show how connected and relatively fragile the Internet ecosystem is. Case in point: December's trifecta of Amazon Web Services (AWS) outages, which really brought home the fact that no service is too big to fail:

12/07/2021. Millions of users were affected by this extended outage originating in the US-EAST-1 region, which took down major online services such as Amazon, Amazon Prime, Amazon Alexa, Venmo, Disney+, Instacart, Roku, Kindle, and multiple online gaming sites. The outage also took down the apps that power warehouse, delivery, and Amazon Flex workers—in prime holiday shopping season. The AWS status dashboard noted that the root cause of the outage was an impairment of several network devices.

12/15/2021. Originating in the US-West-2 region in Oregon and US-West-1 in Northern California, this incident lasted about an hour and brought down major services such as Auth0, Duo, Okta, DoorDash, Disney, the PlayStation Network, Slack, Netflix, Snapchat, and Zoom. According to the AWS status dashboard, "The issue was caused by network congestion between parts of the AWS Backbone and a subset of Internet Service Providers, which was triggered by AWS traffic engineering, executed in response to congestion outside of our network."

12/22/2021. This incident was triggered by a data center power outage in the U.S.-EAST-1 Region, causing a cascade of issues for AWS customers such as Slack, Udemy, Twilio, Okta, Imgur, Jobvite and even the NY Court system web site. Although the outage itself was relatively brief, related effects proved vexingly persistent, as some AWS users continued to experience problems related to the issue up to 17 hours later.

The reality is, the next outage is not if, but when, where, and for how long. Pretending they don't exist or won't happen is not only pointless but harmful to your business. Looking back at the three December outages, we see four key takeaways:



1. Early detection is key to handling outages like the AWS incidents.

Catchpoint observed all three outages well before they hit the AWS status page:

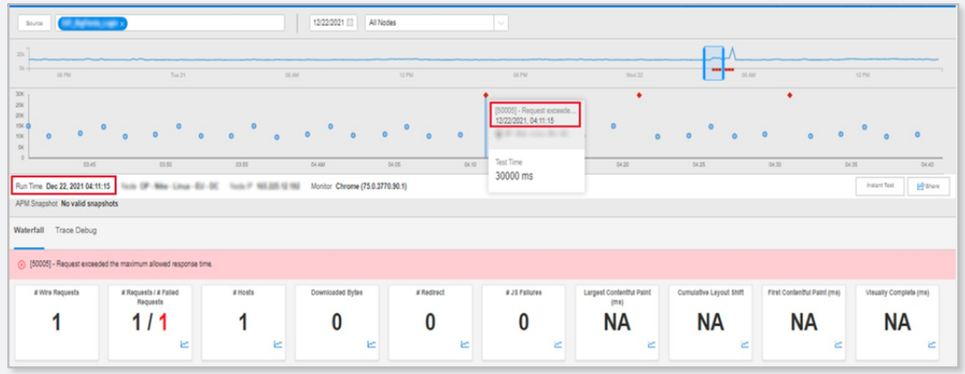
12/7/2021: Here at Catchpoint, we observed connectivity issues for AWS servers starting at 10:33 AM ET, considerably earlier than the announcement posted to the AWS Service Health Dashboard at 12:37 PM EST.

Waterfall graph showing 504 error response for HTML page of Amazon site (Catchpoint)

12/15/2021: Catchpoint noted the outage at approximately 10:15 AM ET, once again before the AWS announcement at about 10:43 AM ET.

User sentiment analysis (Catchpoint)

12/22/2021: Catchpoint first observed issues at 07:11 AM ET, 24 minutes ahead of the AWS announcement.



Proactive Chrome browser observer showing AWS outage (Catchpoint)

Early detection allows companies to fix problems potentially before they impact customers and implement contingency plans to ensure smooth failover as soon as possible. If the issue continues, it also allows them to proactively inform customers with precise details about the situation and assure them that their teams are working on it.

2. Comprehensive observability helps your team react at speed to outages.

While it may be tempting to leave AWS monitoring to, well, AWS, that could leave you in the dark, observability-wise. A comprehensive digital observability plan should include not only your own technical elements, but also components within the Internet Stack that appear to be not within your control. For example, you need insight into the systems of third-party vendors such as content delivery networks (CDNs), managed DNS providers, and backbone Internet service providers (ISPs).

While these might not be your code or hardware, your users will still be impacted by any issues they experience, and so will your business. If you are observing your end-to-end experiences, using an IPM strategy you can act when failures happen within any one of these dependencies that impact your users.

It also means continuous monitoring of your systems to detect failure of fundamental components such as DNS, BGP, TCP configuration, SSL, the networks the data traverses, or any single point of failure in an infrastructure that we rarely change.

This issue is exacerbated by the fact that cloud has abstracted a lot of the underlying network from development, operations, and network teams. That can make it harder to find a problem.

As a result, it can catch us by surprise when these fundamental components fail. If teams are not properly prepared, it adds needless — and costly — time to detect, confirm, or find root cause. Therefore, ensure that you continuously monitor these aspects of your system and train your teams on what to do in case of failure.

3. Ensuring your company's availability and business continuity is not a solo endeavor.

The AWS incidents all clearly illustrate the downstream effect that an outage at one company can have on others. Digital infrastructure will assuredly continue to grow more complex and interconnected. Enterprises today run systems that run across multiple clouds. They also rely on multiple teams, often including a raft of other vendors, such as cloud compute, CDNs, and managed DNS.

When issues originating with outside entities such as partners and third-party providers can bring down your systems, it is time to build a collaborative strategy designed to support your extended digital infrastructure and ensure digital resilience. For that, digital experience observability into every service provider involved in the delivery of your content is crucial.

4. Depending on only a monitoring solution hosted within the environment being monitored is not enough.

While there are many monitoring solutions out there, make sure that you have a "break glass system" to be able to failover to a solution outside of the environment being monitored. Many visibility solutions are located in the cloud, which makes them vulnerable when cloud technologies go down.

This is why ThousandEyes, Datadog, Splunk (SignalFX), and NewRelic all reported impacts from the 12/07/21 and 12/15/21 events.

During the first event, Datadog reported delays that impacted multiple products, Splunk (SignalFX) reported that their AWS cloud metric syncer data ingestion was impacted, and NewRelic reported that some AWS Infrastructure and polling metrics were delayed in the U.S.

There were also a number of issues triggered by the 12/15/2021 event:

- Datadog reported delays in collecting AWS integration metrics.
- ThousandEyes reported degradation to API services.
- New Relic reported that their synthetics user interface was impacted, as well as some of the APM alerting and the data ingestion for their infrastructure metrics.
- Dynatrace reported that some of their components hosted on AWS cluster were impacted.
- Splunk (Rigor and SignalFX) reported increased error rates in the West coast of the U.S. and a degradation in the performance of their Log Observer.

Lack of observability is never a good thing, but over the course of an outage, it is significantly worse.

"The first thing that would be useful is to have a monitoring system that has failure modes which are uncorrelated with the infrastructure it is monitoring."

~Adrian Cockcroft, Tech Advisor, Independent; Partner and Analyst, OrionX.net

Published on Feb 15, 2022

Incident Review: November 16, 2021 – Google Cloud Outage has Widespread Downstream Impact

By Dritan Suljoti

Outages on the Internet always catch you by surprise, whether you are the end user or the Head of SRE or DevOps trying to keep a clear mind while you execute your incident playbook.

As people in charge of ensuring reliable services for our customers, our normal experience of outages involves surfing a deluge of fire alarms and crisis calls as we work to solve the problem as quickly as we can. We often forget, therefore, what an outagew means to the end user.

On Tuesday, November 16, 2021, however, I was reminded of exactly how the shoe feels on the other foot.

In the shoes of the end user: the broken Google bot

On Tuesday, as I was trying to purchase something for my home on Homedepot.com, my browser rendered an unusual page: the Google bot page with its 404 message.

Google 404 Error Page (Google)

Surprised, I clicked "reload." Nope, still the same page. I typed the URL again with www. and then without it, but still the same broken Google bot greeted me.

"Well, there's no way Google acquired Home Depot," I thought (although its parent company is missing a company that starts with the letter H).

Being a tech geek, the next thing in my mind was maybe I was somehow using Google's Public DNS 8.8.8.8 and the DNS lookup was failing and Google had decided to launch a new feature that routed non-resolvable domain to their IP (a technique we've seen a few companies use before)? But no, neither of those was it either.

Giving up on being a consumer and blind to what was incomprehensible, I simply logged onto the Catchpoint platform to see what was going on. The answer was immediately clear: multiple sites failing, all experiencing the same error message and all of them customers of Google Cloud. I visited Google's status page, and nothing was posted yet… and there was still nothing on it for another thirty minutes from when the problems started.

Let's take a quick look at the incident itself.

The latest outage of 2021

Tuesday November 16, starting soon after midday ET, many companies not owned by Google saw their websites knocked offline, replaced by the Google 404 page.

What was going on?

Google didn't acquire your favorite site then shut it down. In fact, it was collateral damage due to the latest outage of 2021, this time on Google Cloud, which many, many companies rely on for hosting. The impact on a lot of these companies would have been lost revenue and possible damage to business reputation.

Catchpoint saw a sudden burst of test failures

At Catchpoint, we saw a sudden burst of test failures, beginning at 12:39pm ET. It impacted many companies, large and small. Some of the businesses that were affected include the likes of Nest, 1800Flowers, CNET, Home Depot, Etsy, Priceline, Spotify, and Google itself.

By around 13:10, the problem was partially resolved. However, according to Google, the issue did not get fully resolved for all impacted products for almost two hours, lasting until 14:28 am ET. Some companies were back online quickly, while others continued to experience errors or long loading times for some time.

Below is a chart showing availability of many of these sites. You can easily see where the sudden plunge from the cliff edge took place, diving from steady high availability to 0%.

Failed tests show the impact on many of our customers (Catchpoint)

Spotify dashboard showing availability at 0% (Catchpoint)

The customer impact varied according to the Google Cloud service it depended on. For instance, Google App Engine saw an 80% decrease in traffic in central parts of the U.S. and portions of Western Europe. Google Cloud Networking customers were unable to make changes to website load balancing, which led to the 404 error pages.

Indeed, it was not just web pages that were impacted, but multiple Google Cloud products, including:

- Google Cloud Networking
- Google Cloud Functions
- Google Cloud Run
- Google App Engine
- Google App Engine Flex
- Apigee
- Firebase

"A latent bug in a network configuration service"

Google Cloud apologized for the service outage and any inconvenience it caused to its downstream customers. On its status page, the organization specified root cause as "a latent bug in a network configuration service which was triggered during a leader election charge." The cloud giant has assured customers there are now "two forms of safeguards protecting against the issue happening in the future."

With the massive adoption of public cloud, this latest incident (in a long year of outages) illustrates how significant the impact a public cloud vendor outage can be downstream. It also illustrates how vulnerable enterprises are to third-party vendor outages.

You can clearly see that many enterprises rely on public Internet, services, and infrastructure to conduct their business and deliver digital experiences to their clients. While there are many positives to this situation, the challenge is that those same businesses have little to no control over the underlying infrastructure on which their organizations run.

Three key lessons from the Google Cloud outage

Below are three critical lessons that we took away from this outage, which you can apply to your own business:

LESSON 1

While failure is bound to happen, don't overlook the importance of communicating to the end user through a proper error page.

Don't assume your users will find your status page or go to Twitter to see your communications. They will have already moved on to your next competitor and will eventually read about it in the news.

If there was one thing that confused the end users in this instance, it was the Google error page that greeted people. Most people would have expected an error message from the company they were trying to reach, not the hosting company. It's a little unclear if Google Cloud allows folks to modify this. Perhaps it's not possible.

However, any company should be ready for such failures, and implement a process where they are able to change the DNS or CDN configuration to point people to a proper error page with their own branding and messaging to apologize for the failure in their own words. And ideally, make it fun. Don't be afraid to be human and relate to the end users. A proper error page is always better than confusing error pages (as in this instance), obscure errors (such as "the server failed to respond"), or worse, nothing at all and hanging on into infinity to connect to the server.

LESSON 2

Ensure you implement proper observability of your services, which means an IPM approach that monitors from outside your firewall, datacenter, or cloud.

While many observability platforms have defined "observability" to fit their products (tracing and logging) - in reality observability had its origins long before tracing came about. In control theory, observability is defined as a measure of how well the internal states of a system can be inferred from the knowledge of its external outputs.

This won't have been the first time that a company will have learned they are down from the news or a customer complaining. You do not want to find out about the problem in this way. Far better to stay ahead of it by observing your services using an IPM platform from outside your cloud provider. When you are relying on code tracing and logs alone, you won't see the problem.

Be proactive and stay on top of your services, and the services and infrastructure providers you rely on that are single points of failure.

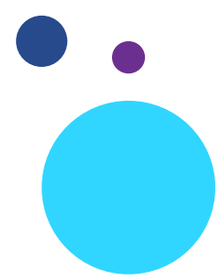
LESSON 3

Finally, track the SLA of your services, and know your MTTR.

You need to track how good your teams and providers are at resolving issues. This is how you build trust and verify people are doing what they are accountable for.

Real time data from an independent monitoring and observability solution will allow you to find out precisely when the issue started and when it was resolved. You cannot rely on status pages to be accurate about the impact the problem had on your site. Everyone will have been impacted differently: slightly earlier or later, shorter, or longer…

Published on Nov 18, 2021



Incident Review: October 7, 2021 – The Ripple Effect of A BGP Misconfiguration at Telia

By Alessandro Improta, Anna Jones, and Luca Sani

On Thursday October 7th, Telia, a major backbone carrier in Europe, suffered from a network routing issue between 16:00 and 17:15 UTC. This had a significant ripple effect with several other major companies reporting outages at around the same time.

Companies affected included:

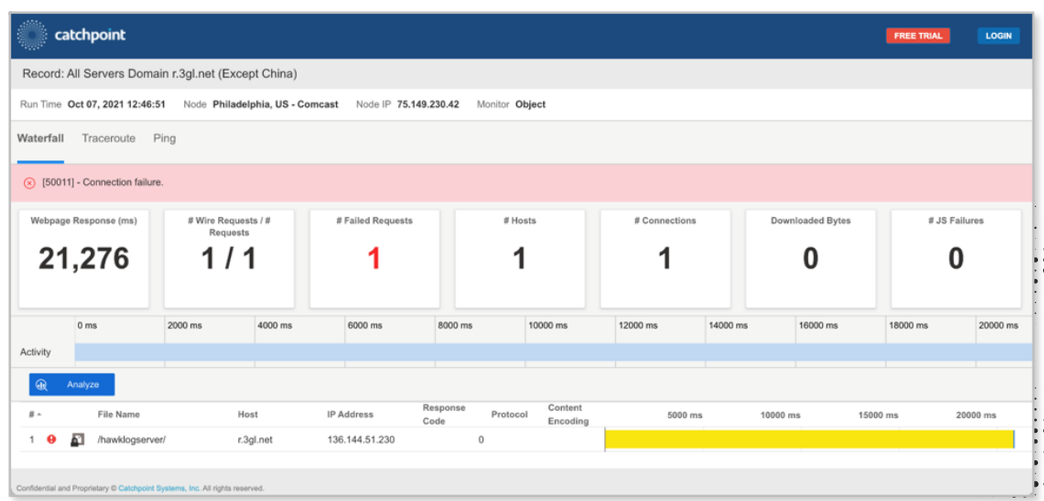
- Cloudflare
- Equinix Metal
- Fastly
- Firebase
- NS1

It's always startling to see the secondary and tertiary effects that a major outage can have. In this instance, it briefly caused some of the world's biggest infrastructure and content delivery networks to have serious performance issues in the U.S., Canada, and 12 other countries.

There is a common theme in the slew of outages the last eighteen months have brought us: the added – and painful – impact that an outage in one part of the Internet Stack can have on all third-party dependencies.

What we saw at Catchpoint

Many of our customers saw huge increases in webpage response times at the time of the Telia incident. We saw increases in response time across portions of the U.S. East Coast and Europe.

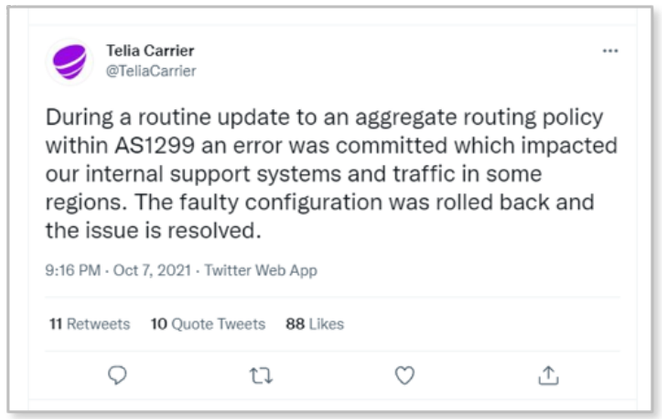


Waterfall showing STCP connection timeout of request to Equinix Metal. (Catchpoint)

The mystery is solved: BGP misconfiguration

What caused the outage to occur?

At 21:17 UTC, Telia Carrier posted the following statement:



Outage Tweet from Telia. (Twitter/@TeliaCarrier)

Just as with previous Facebook and Spectrum outages, there was a common villain: BGP. In this instance, the outage was directly caused by a BGP misconfiguration.

"An engineer in another department of a large company may change their own process for the better after reading about an incident written by someone in an unrelated department that didn't directly impact them at the time. This is where distribution comes into play. At the extreme end of this, which I'm hoping we're trending towards as an industry overall, is making postmortems public to get the maximum downstream learning impacts across the entire industry and not just within a single company."

~John Egan, Former co-founder and Product Lead Workplace, Facebook

Ultimately, only a postmortem from Telia could shed light exactly on what happened. However, from digging into the BGP data available, we uncovered some remarkably interesting facts…

Let's return to the time of the cri(me)sis

Earlier in the evening, Telia shared this email with their customers (which we accessed from the outage.org mailing list). This note included details of the root cause:

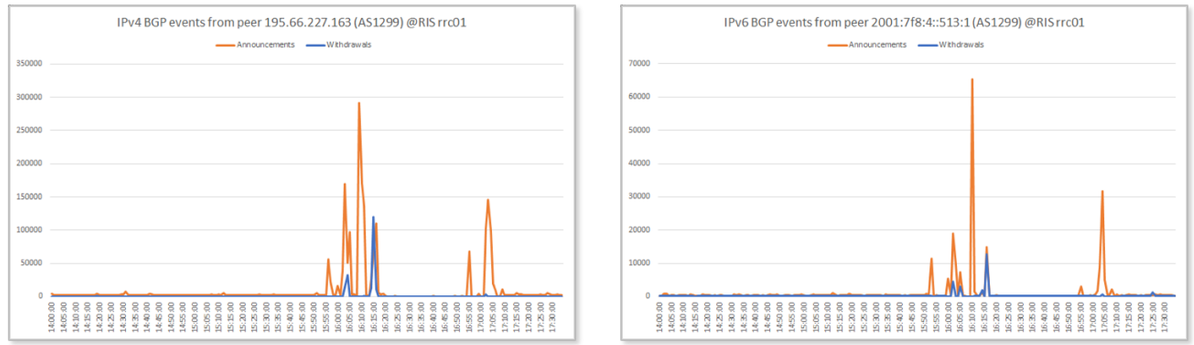
"Dear Customer,

We regret to inform you that your services were affected by an incident that occurred at 16:00 UTC during a routine update of a routing policy for aggregated prefixes in the Telia Carrier IP Core network. This caused traffic to prefixes contained within the aggregates to be blackholed, resulting in an impact on some parts of the network.

When the underlying problem source was traced, the configuration was rolled back to the earlier working version of the routing policy (17:05 UTC). Affected services started to recover gradually after this operation was applied."

The times specified are exactly when we saw the first batch of BGP events.

We focused our attention on rrc01, the RIS route collector that RIPE NCC deployed at the London Internet Exchange (LINX). This router collects data directly from an IPv4 and an IPv6 peer from Telia (AS1299). As can be seen from the following graphs, at around 16:00 UTC, the number of networks announced and withdrawn from the two peers spiked upwards across both peers.



Data collected from IPv4 and IPv6 peers showing withdrawn networks. (London Internet Exchange)

It is interesting to note that the two peers of AS1299 generated BGP events related to about 500k IPv4 networks and 32k IPv6 networks. In other words, more than 50% of the full set of IPv4 routes were affected, as were more than 30% of the full IPv6 routes shared by the peers. This gives us a rough initial idea of how widely the outage at Telia affected the entire Internet.

What can you do to tackle BGP misconfigurations?

It's the duty of every network operator to avoid misconfigurations in the router they manage. However, no one is perfect and BGP (misconfigurations) happens! That's why businesses need a strong IPM structure in place to alert them as soon as anything like this begins to spread in the wild. It is critical to react swiftly to the issue, to minimize the disservice for end users.

It's easy to see from the BGP data that, as soon as the BGP instability at Telia started, several operators decided to temporarily switch off their peering with AS1299 and/or attempted to route the traffic on alternative routes.

While it is impossible to avoid BGP misconfiguration completely, network operators should follow common sense rules and apply some of the best practices advocated by MANRS. This will help enable them to minimize the chances of a BGP misconfiguration.

Published on Oct 8, 2021

Incident Review: October 4, 2021 – A Case of Social Networks Going Anti-social at Facebook

By Zachary Henderson, Alessandro Improta, and Anna Jones

In a highly unusual state of events, Facebook, Instagram, WhatsApp, Messenger, and Oculus VR were down simultaneously around the world for an extended period, Monday, October 4th.

The social network and some of its key apps started to display error messages before 16:00 UTC. They were down until approximately 21:05 UTC, when things gradually began to return to normal.

Facebook Tweet acknowledging the outage. (Twitter/@Facebook)

Can humanity survive hours without the most important social media conglomerate of our time? On a more serious note, as some users pointed out on Twitter, did the global outage highlight the challenges of such a dominant single technological point of failure?

Facebook is everywhere… it's beyond just social media

What quickly became clear to us at Catchpoint was the fact that the outage was impacting the page load time of many popular websites that are not powered by Facebook. Why? Because Facebook ads and marketing tags are on almost every major website.

Here's an aggregate of the 95th percentile of onload event time, referred to as document complete, alongside the availability and Catchpoint "bottleneck time" impact metric of the site's embedded Facebook content, across the IR Top 100 sites, as measured from Catchpoint's external synthetic monitoring vantage points.

Metrics showing significantly higher page load times for Facebook users. (Catchpoint)

Metrics showing significantly higher page load times for Facebook users. (Catchpoint)

Note the measurement of document complete spikes and sustains at 20+ second higher at 15:40 UTC. These indicate that overall page load times for users were much higher than normal.

"People and businesses around the world rely on us every day to stay connected. We understand the impact that outages like these have on people's lives, as well as our responsibility to keep people informed about disruptions to our services. We apologize to all those affected, and we're working to understand more about what happened today so we can continue to make our infrastructure more resilient."

~Santosh Janardhan, VP, Infrastructure, Facebook

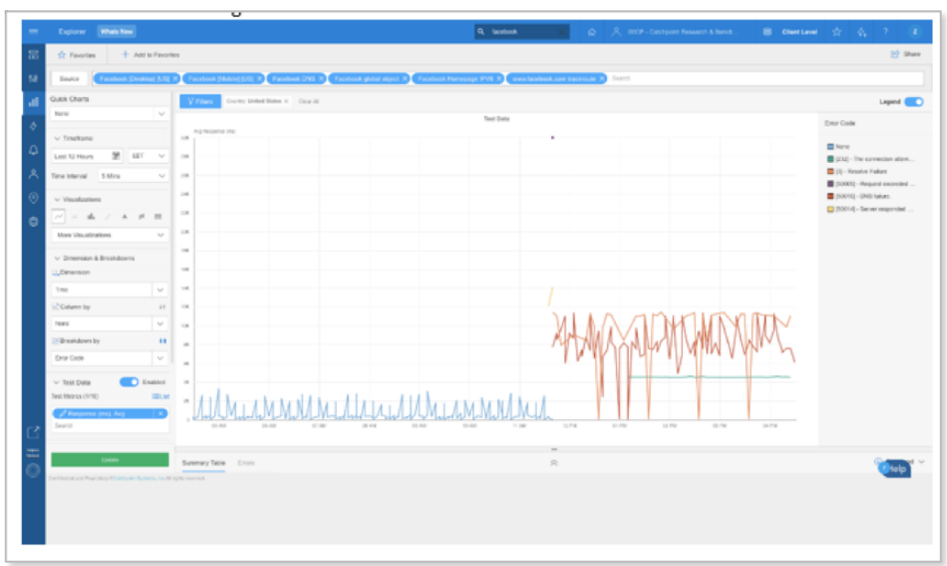
Alarms started at Catchpoint when we detected server failures

Here at Catchpoint, alarms started to trigger around 15:40 UTC. These alarms resulted from the fact that some of our HTTP tests for Facebook, WhatsApp, Instagram, and Oculus domains started to return HTTP error 503 (service unavailable).

It's worth noting that we do this type of monitoring as part of a benchmarking process. In this way, we can provide insights into the Internet as a whole.

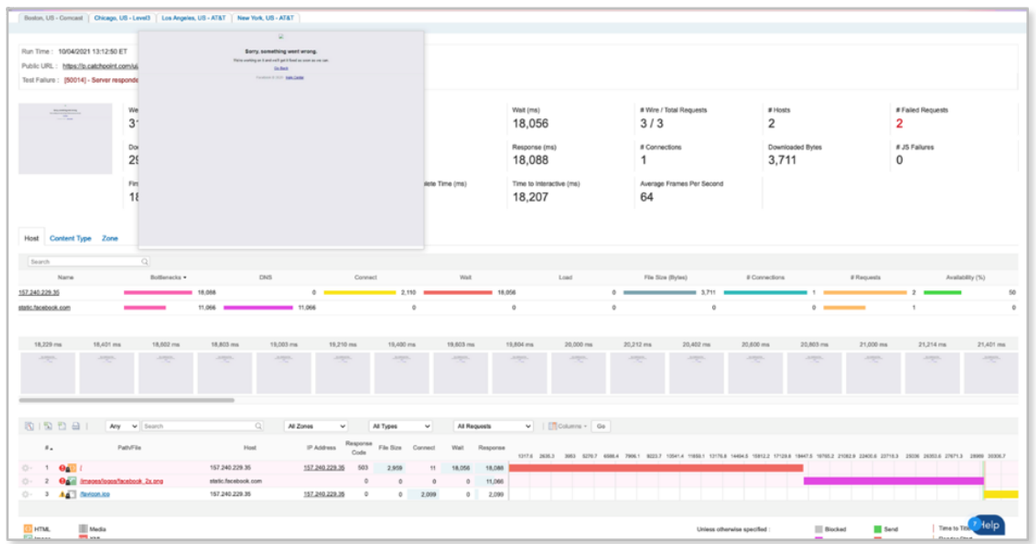
We usually see that Facebook is a highly stable system. The business has built a scalable, reliable, global service. Therefore, when we saw alarms about a Facebook outage, it was easy to determine this was a significant problem.

The snapshot below is from the Catchpoint Data Explorer. It shows the server failures that first alerted us to the outage.



Data Explorer snapshot showing Facebook server failures. (Catchpoint)

Five minutes later, we saw that the TTL of the DNS records of Facebook had expired. Shortly after, it became clear that no Facebook nameserver was available, and every DNS query towards www.facebook.com was resulting in a SERVFAIL error (meaning a DNS query failed because an answer cannot be given).



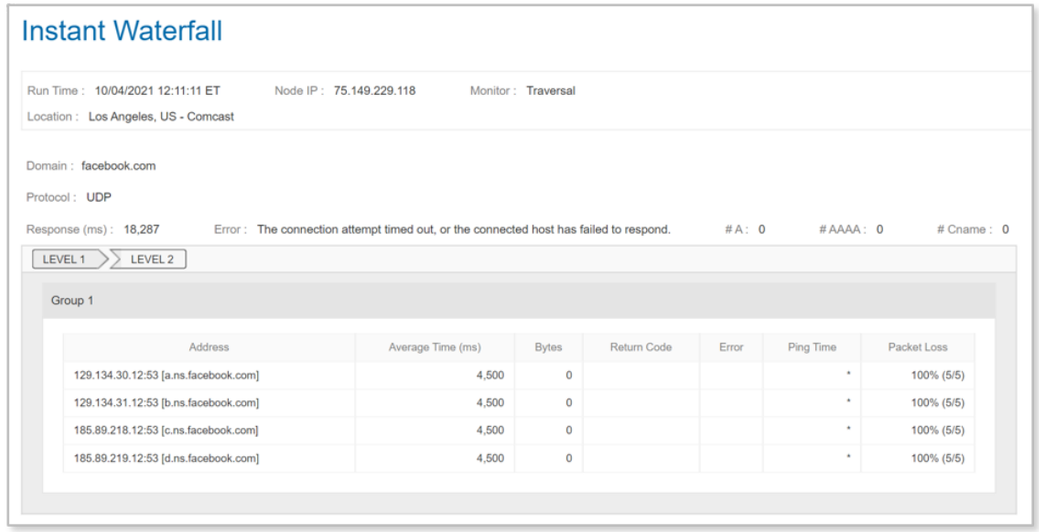
Facebook DNS records showing SERVFAIL errors. (Catchpoint)

Below are examples of the types of HTTP headers 503 errors seen initially:

- HTTP/2 503
- access-control-allow-origin: *
- content-length: 2959
- content-type: text/html;
- charset=utf-8date: Mon, 04 Oct 2021 16:48:36 GMT
- proxy-status: no_server_available;

You can see that it was initially returning a server failure. When DNS records were cached, Facebook's edge was unable to find an upstream proxy server as part of their communication setup.

The next set of screenshots show that when we queried Facebook's top-level domain servers, they weren't working.



Instant Waterfall showing the top-level Facebook domain servers are down. (Catchpoint)

Everything up to now led us to think that the cause of the issue was DNS... But was it?

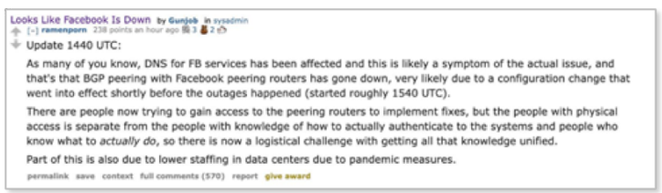
"Facebook is the Internet in a lot of communities in Southeast Asia," says. "Use of the Internet is really just Facebook and WhatsApp."

~Ross Tapsell, Senior Lecturer, Australian National University, Asia and the Pacific

A tale of badge failure and BGP

A lot of speculation around the incident occurred on the surviving social media platforms.

Tweet sharing rumor that Facebook door badges were not working. (Tweet/Sheera Frenkel)

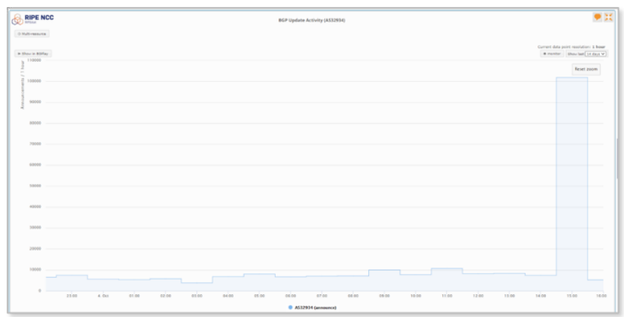


Reddit comment about BGP peering involved in Facebook incident (Reddit/uGunjob)

We may never know if the Facebook technical staff were indeed locked out of the server room and unable to fix their routers. At the same time, there is some truth to this final speculation: BGP was, indeed, heavily involved in this incident.

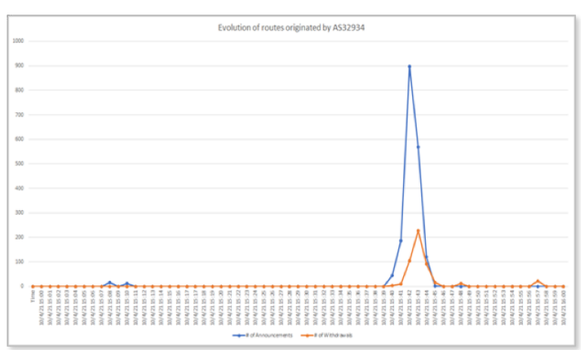
A deep dive into the BGP data

Facebook manages AS 32934. The networks it originates are usually stable, as can be seen from RIPEstat (RIPEstat - Ui2013/AS32934).



RIPE NCC data showing a spike in the number of BGP events. (RIPEstat)

Something changed, however, at around 15:40 UTC. At that time, you could clearly see a spike in the number of BGP events. We focused on BGP data collected by RIS rrc10 collector deployed at the Milan Internet Exchange (MIX) between 15:00 UTC and 16:00 UTC.



Evolution of routes originated by AS32934. (Milan Internet Exchange)

From a quick look at the snapshot of 08:00 UTC, AS 32934 was originating 133 IPv4 networks and 216 IPv6 networks. The update messages made it easy to spot that Facebook withdrew the routes to reach eight of those IPv4 networks and fourteen of those IPv6 networks around 15:40 UTC. This was exactly the time when

all the Catchpoint alerts started to trigger, and people began to complain about outages.

Even though just a handful of networks experienced outages, this incident demonstrates that it's not the quantity of networks that matters. Some of the withdrawn routes were related to the Authoritative DNS nameservers of Facebook, which couldn't be reached any more. This led to DNS resolutions from all over the world failing. Eventually, it resulted in DNS resolvers being flooded with requests.

Tweet from Cloudflare's CTO on the flood of DNS traffic during the Facebook outage. (Twitter/@jgrahamc)

Authoritative nameservers play a key role in DNS resolution, since they possess information on how to resolve a specific hostname under their authority.

Having a quick response is key, as long as your badge is working!

How quickly you detect and get to the heart of an outage matters. Your runbooks also matter.

Sometimes fixing an escalation means you need to ensure your systems are different from one another. In this case, the badge systems your employees use to sign in and fix things should never be dependent on the thing you're trying to fix.

Troubleshooting in these types of instances is rarely straightforward. In Facebook's case, the symptoms were HTTP and DNS errors, which then impacted BGP.

Update from Facebook's postmortem analysis

On October 5, 2021, the Facebook team released a very good post-mortem analysis of the incident.

The source of the incident was not caused by DNS or BGP, but by a maintenance routine job performed by Facebook staff aimed at assessing the availability of global backbone capacity. This backfired (unintentionally), taking down all the connections in their backbone network. Consequently, the Facebook routers couldn't speak to their data centers. This triggered a safety mechanism in which the BGP routes towards their DNS servers were withdrawn from the network, as we saw in our analysis.

Kudos to the Facebook team for the prompt recovery, but most importantly, for their transparency!

Published on Oct 04, 2021

Incident Review: September 29-30, 2021 – Issues Caused by the Let's Encrypt DST Root CA X3 Expiration

By Sergey Katsev

As a monitoring and observability company, we have a lot of monitoring and observability built into our systems, as well. We have the tandard monitoring programs in place to make sure that our systems are performing properly, data is flowing through our infrastructure, etc. At the same time, we continually observe for any sudden changes to tests that our customers are running.

On September 29, 2021, 19:22 UTC, we started to see a wave of alerts. The alerts originated from some of the web tests from our active observers, occurring when our Let's Encrypt "R3" certificate expired.

Another example of this happened in 2020 when the Sectigo AddTrust root certificate expired. Let's be clear, these types of incidents are pretty rare. The difference with this event was that a lot more servers rely on Let's Encrypt certificates.

The root cause of the crisis was not Catchpoint, our product, or any employee. Instead, it was an issue with changes to the certificate path by a certificate issuer. Furthermore, as we work with many vendors, we've received updates from some which indicate that solving this problem is as easy as downloading the latest OS updates. While this is true for some, it does not solve the problem in general!

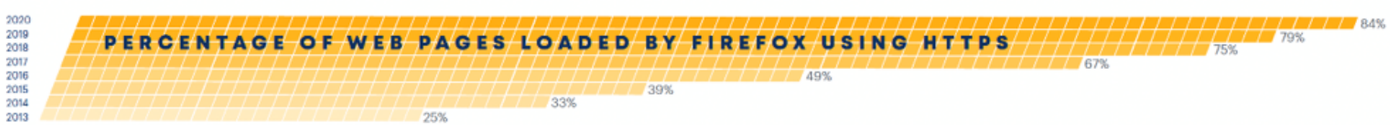
Below we explain why, and how to solve it on the server-side so that all your clients can access your web service without issues. Let's dive into our incident review.

Do you trust Let's Encrypt?

Before we get into the weeds, I just want to say that I, personally, trust Let's Encrypt. They're a great company that has made certificate management accessible to everyone, and they are extremely developer friendly. Additionally, partially because of them, the number of websites using encryption has skyrocketed in recent years.

Encryption is extremely important on the Internet. It's the basis for secure communications. Whether you're checking your bank balance, buying a new pair of socks from an e-tailer, or talking to your friends, you do so with the assumption that this transaction is secure.

In the last approximately eight years (2013-2020), the percentage of web pages using HTTPS has gone from 25% to more than 84%! There are several reasons for this incredible growth. One of them is that Google has been gradually forcing sites to use HTTPS by making HTTP-based sites "not secure." Still, no matter the cause, in that amount of time Let's Encrypt has gone from issuing certificates for about 50 million websites to over 230 million!



Increase in percentage of web pages loaded by Firefox using HTTPS. (Let's Encrypt)

At the same time, it doesn't matter that I trust Let's Encrypt if computers don't. That's exactly what happened on the evening of September 29 and again on the morning of September 30, 2021.

"In the last year alone, Let's Encrypt have grown their market share quite a lot and as a CA becomes larger, its certificates enable more of the Web to operate and as a result, when something like this comes along, they have the potential to cause more problems. This is nothing to do with what Let's Encrypt have done, or have not done, this still comes down to the same underlying problem that devices out in the ecosystem aren't being updated as they should be."

~Scott Helme, security researcher

How digital certificate trust works

Here's a quick summary of how certificate trust works on the internet:

- **Root certificates:** There are a handful of Root certificates. These are issued by major companies under a lot of scrutiny and are installed in the Certificate Stores of computers worldwide by the company that developed and maintains the OS. If you have a computer that can connect to an HTTPS website, you have such a certificate store. The MacOS laptop I'm writing this on has 161 "System Root certificates" installed.

- **Chain of Trust:** When someone launches a website nowadays, they must support HTTPS. Therefore, they purchase a certificate from a provider. There are many providers to choose from. Some have their own Root certificate and others have a Certificate Authority certificate, which was signed by one of the Root certificates or by another Certificate Authority. In this way, there is a chain of trust from the website's certificate all the way to the root certificate.

- **Intermediate certificates:** When you go to the HTTPS website, the server hosting the website sends the certificate during the SSL handshake with the client (browser/HTTP client). It might also send you one or more intermediate certificates. These intermediate certificates are what it thinks you might need to connect the chain of trust from the server certificate to one of the root certificates you installed on your computer.

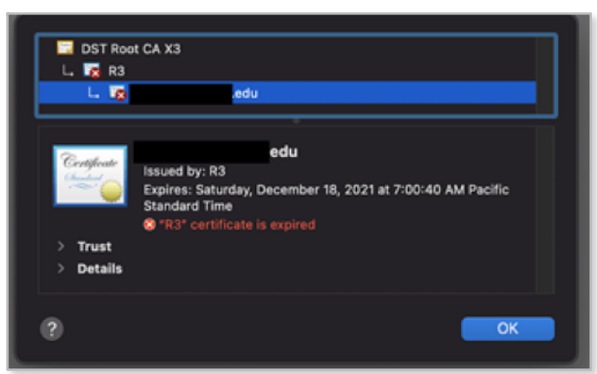
- **Validation process:** Your browser "walks" the chain of trust, from the server certificate up to the root. If it makes it all the way to the root and finds it in its "store," the chain is validated, and the connection is allowed to proceed. Otherwise, you get a security warning like the one below.

R3 certificate expiry and the chain of trust

Let's go through the details of what happened.

As mentioned, the first problems we saw with web tests from our synthetic nodes began at 19:22 UTC on September 29, when the Let's Encrypt "R3" certificate expired. Here's the certificate information for this intermediate certificate: https://crt.sh/?id=3479778542.

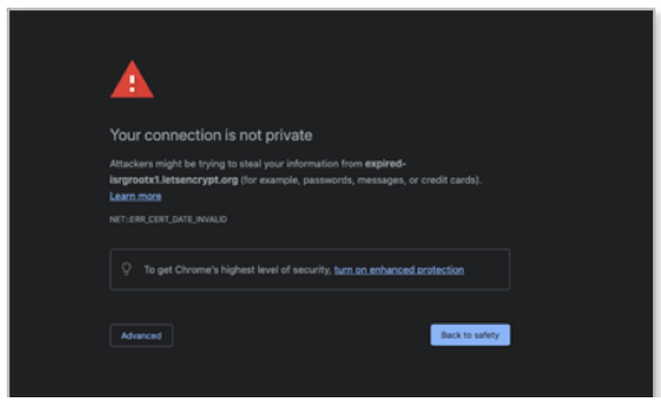
The most important piece of information here is the expiration date. Since this is an intermediate certificate, this means that Let's Encrypt used this certificate to sign other certificates for their customers. For example, see the following screen capture.



DST Root CA X3 expiration date on an intermediate certificate. (Let's Encrypt)

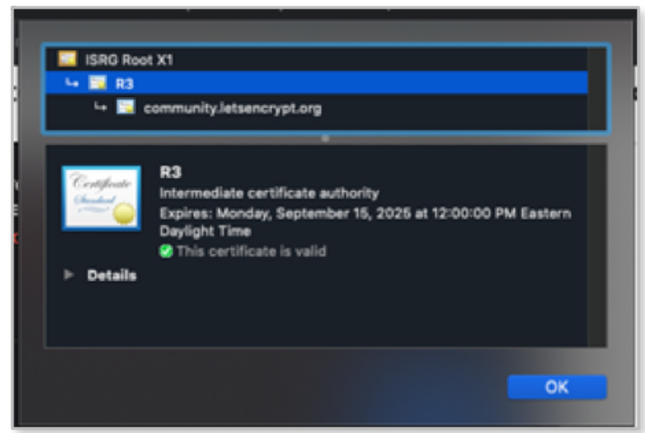
This screenshot shows a web site whose certificate was signed with this R3 certificate. As soon as the certificate expired, this website was no longer accessible!

A browser which tried to validate the website's certificate would walk the chain of trust and find the intermediate certificate expired. That's when you get scary looking errors in your browser, like the one below.



Browser message: "Your connection is not private." (Let's Encrypt)

Let's Encrypt published a new R3 certificate! The new expiration date is in 2025 – plenty far away. Everything's great, right? Well, not quite.



New Let's Encrypt R3 certificate. (Let's Encrypt)

It turns out that the certificate needs to be updated in your computer – usually through a Windows or MacOS update – before it works. A lot of people don't update their computers as often as they should, though. Even worse, a lot of embedded devices rarely if ever update their certificates! Someone here at Catchpoint mentioned that his kids couldn't watch their favorite streaming show from his SmartTV because of this issue!

Let's say you got the latest R3 intermediate certificate installed in your whole fleet of devices. Now you can

go to any of those millions of sites with Let's Encrypt certificates, right? Well, sort of...until 14:00 UTC the following day.

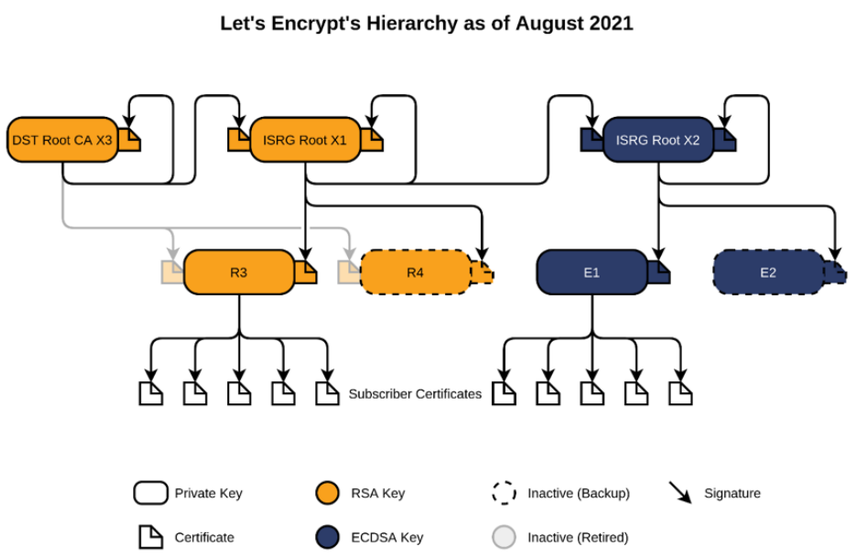
September 30, 14:00 UTC: DST root CAx3 certificate expiry and its consequences

At 14:00 UTC on September 30, the DST Root CA X3 certificate expired. The details are a little confusing. Bear with me.

Originally, the DST Root CA X3 was used to sign all Let's Encrypt certificates (including the R3 intermediate certificate above). Let's Encrypt also cross-signed the certificates using their own ISRG Root X1 certificate. This was done because the DST certificate was already present in most browsers and devices. However, the ISRG certificate was not.

As Let's Encrypt became more well-known and the ISRG certificate was available in all major devices, they stopped relying on the DST certificate.

The following diagram portrays the certificate hierarchy directly from Let's Encrypt.



Let's Encrypt Certificate Hierarchy (Let's Encrypt)

Note that any server certificates ("Subscriber Certificates") that were signed by R3 were signed by either DST root or ISRG root, or, most likely, cross-signed by both.

Here's the DST root certificate's information: https://crt.sh/?id=8395

There are actually two versions of the ISRT Root X1 certificate: https://crt.sh/?id=3958242236 and https://crt.sh/?id=9314791

The second one is self-signed. This is fine for a Root CA certificate which is present in most devices around the world. The first one, though, is signed by DST Root CA X3!

When the DST root certificate expired, this caused problems for two classes of system:

Systems which didn't have an updated copy of the ISRT Root X1 certificate started failing to connect to sites using Let's Encrypt because their site certificate was signed by R3, which was signed by ISRG, which was signed by DST – which had expired!

Systems which did have the proper updated copy of the ISRT Root X1 certificate but wanted to validate the DST Root certificate anyway, because it had cross signed the R3 certificate!

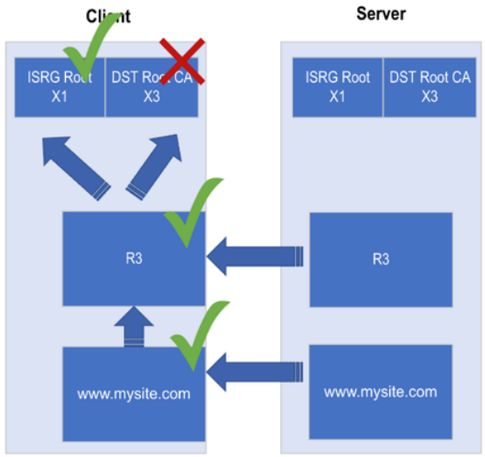
Fixes, fixes…

The first category was relatively easy to fix: update the OS or download the new certificate and install it, assuming it's not an embedded device that hasn't issued an update.

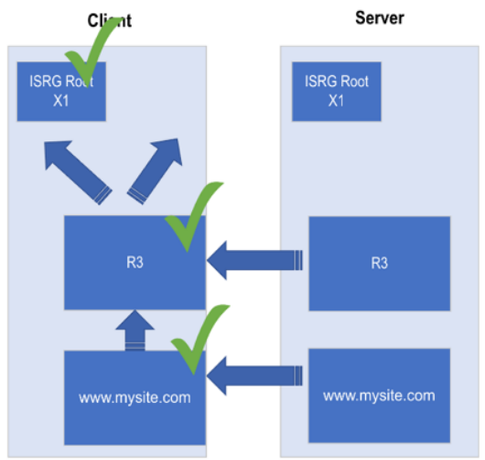
The second one is harder. For example, any software which relies on OpenSSL 1.0.2 or earlier will have this problem – and there's no way for the client to fix it.

Think of it this way:

Usually, your website sends the server certificate and any intermediate certificates that the client might need. As the client walks the chain of certificates, it sees the site certificate signed by R3. Then it sees that R3 is signed by ISRG, but also by DST – some browsers or other HTTP clients only validate one. They then find that the ISRG certificate is valid, and they're satisfied. Others need both to be valid but they're not, because the DST certificate is expired!



Client certificate validation process when the server includes DST certificate (Catchpoint)



Client certificate validation process when the server doesn't include DST certificate. (Catchpoint)

If you run a website and want customers to be able to connect from devices such as these, there's only one fix: Regenerate the certificate that your site uses so that it is no longer cross-signed by the DST certificate. At that point, your server will stop sending it to the client to validate, and the client won't fail to validate it.

This is particularly important if you have HTTPS-based services that aren't being accessed directly by browsers on a laptop. Maybe you're serving RSS feeds or have an API accessed by embedded devices. Maybe your clients access the site through a proxy (there's a higher chance that some of the users trying to access your server are unable to due to this problem).

Fixing this on your server

The first step to fixing this problem was understanding the impact on your customers.

The second step, which many system administrators don't think about, was understanding who your customers are! This is a particularly "weird" issue because there's no way to resolve it for everyone, except I guess switching from Let's Encrypt.

With Let's Encrypt, no matter what you do, someone will be impacted. You have to choose whether your clients are likely to be using old versions of Android (old phones, but also devices like smart TVs), or using old versions of OpenSSL (many other embedded network devices). Or maybe you have a lot of users with non-updated Operating Systems.

Because of the way the certificate chain was put together by Let's Encrypt, they put the onus on each server administrator to decide who to support and who to break.

In conclusion...

Here at Catchpoint, we have a huge footprint of test agents in every corner of the world. These agents have different configurations of hardware, software, firmware, etc. Because of this large fleet, we saw and solved almost every flavor of the issues described above.

However, the customers accessing your website probably don't have a 24/7 Operations team. In theory, we can update all our agents that act as clients to connect to Let's Encrypt-based sites properly or otherwise ignore this server misconfiguration issue. At the same time, the reality is that you cannot expect every user in the world to do this – and often they cannot.

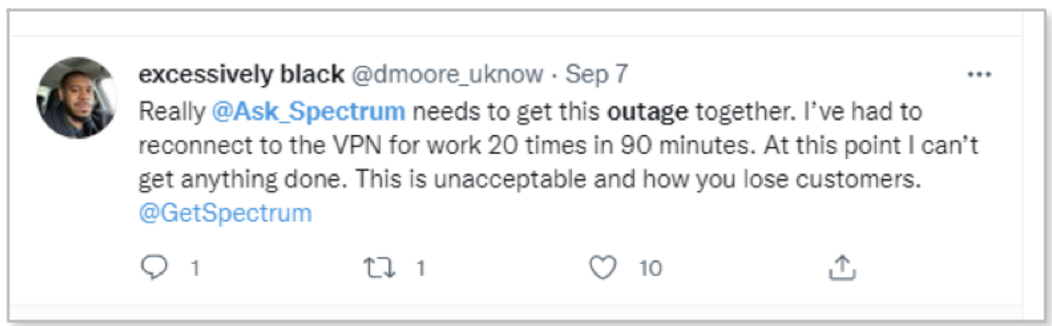
As the owner of that service, you have the solution within your control on the server side, so it is on you to fix it.

Published on Oct 01, 2021

Incident Review: September 7, 2021 – A Case of Dr. BGP Hijack or Mr. BGP Mistake at Spectrum?

By Alessandro Improta and Luca Sani

September 7, 2021, 16:36 UTC: an outage hit Spectrum cable customers in the Midwest of the U.S., including Ohio, Wisconsin, and Kentucky. Users of their broadband and TV services hit social media to voice their annoyance at the disruption it was causing.



Spectrum user complaint on Twitter (Twitter/@dmoore_uknow)

Spectrum user complaint on Twitter (Twitter/@BrettLarter)

Everything was resolved at around 18:11 UTC, and services were restored to users.

Fact checking the Spectrum investigation

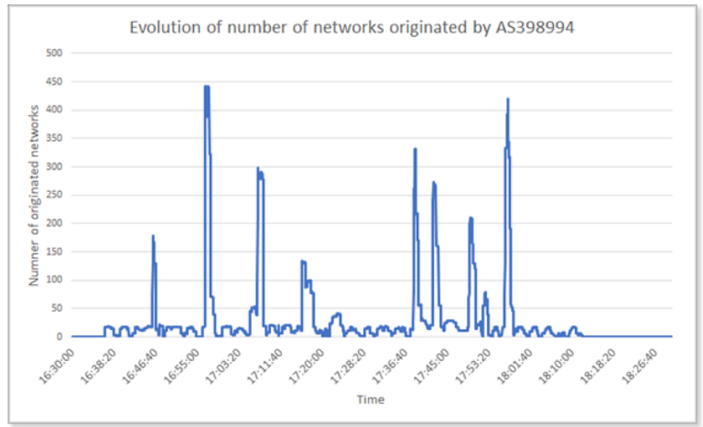
Official Spectrum Tweet acknowledging outage (Twitter/@Ask_Spectrum)

During the outage, Spectrum was vague about the type of issue they were having that was causing it. Our investigation, however, showed it may have involved a BGP route hijack.

A BGP route hijack happens when an autonomous system (AS) claims to be the origin for a network that has been assigned to another AS. If the hijack is accidental, it can lead to a denial of services. If deliberate, at its worst, another AS could attempt to steal sensitive information.

To better understand what happened, we investigated data collected by rrc10, the RIS route collector deployed by RIPE Network Coordination Centre (NCC) at the Milan Internet Exchange (MIX), in Italy. Looking at the closest RIB snapshot available, we could see that Spectrum (AS10796) was announcing 690 networks, most of which routed via its backbone (AS7843).

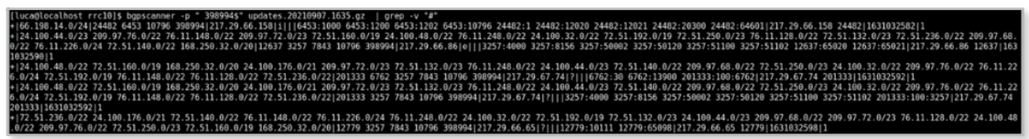
From the update files provided by rrc10 collector between 16:30 and 18:30 UTC, we could further see that KHS USA (AS398994) started to originate 449 of the 690 networks previously originated by Spectrum. This was what started the outage. Routes were seen by most of the RIS peers up to 18:11 UTC, when KHS stopped announcing any routes.



Update files provided by rrc10 collector between 16:30 and 18:30 UTC (Milan Internet Exchange)

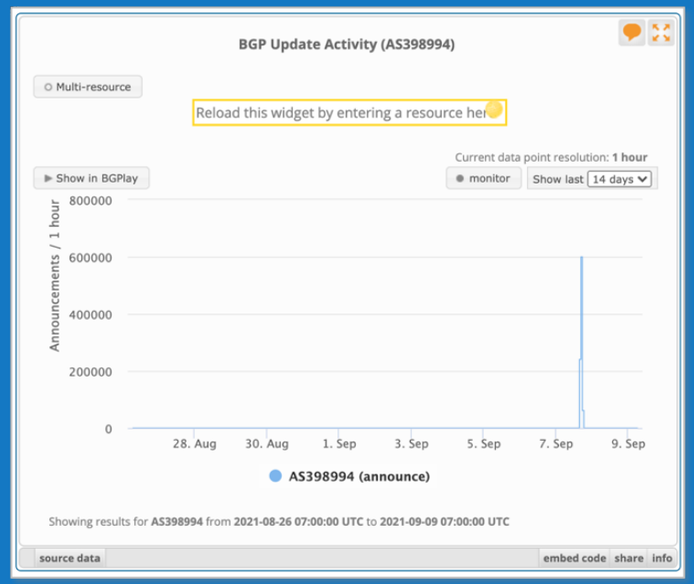
So… was it Dr. BGP hijack or Mr. BGP mistake?

One of the peculiarities of this hijack is that it started in the belly of the attacked AS. KHS announced the Spectrum networks via Spectrum AS itself (AS10796), and the Spectrum backbone (AS7843) propagated them in the wild. Only a few routes were announced via Spectrum AS itself (AS10796) and Tata communications (AS6453), one of Spectrum's providers.

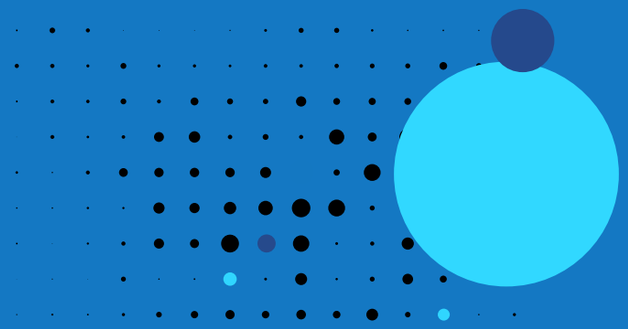


Another oddity is that KHS didn't announce any networks before or after the hijack (this can be seen from the BGP Update activity widget provided by RIPE Stat.

What caused the issue? The outage may have been caused by an experiment at Spectrum. Alternatively, similar scenarios can happen if someone exploits a security vulnerability inside the carrier and finds a way to open a BGP session with one of their routers. This could then lead to a hijacking of the routes to create an outage on purpose. In the past, we have seen similar scenarios where this has been the case, such as those listed here.



BGP Update activity widget. (RIPE Stat)



"BGP hijacking may be the result of a configuration mistake or a malicious act; in either case it is an attack on the common routing system that we all use… The problem is, BGP was created long before security was a major concern. BGP assumes that all networks are trustworthy. Technically, there are no built-in security mechanisms to validate that routes are legitimate."

~Megan Kruse, Director, Partner Engagement and Communications, Internet Society

Lessons for network administrators

Either way, there are a few lessons network administrators can obtain from what happened.

First, around BGP: While a BGP hijack may not have been the case with Spectrum, it is worth mentioning that unrecognized sources absolutely must not be able to set up a BGP session on their own and announce networks at will.

Second, network administrators need to set up automated controls to drop any route announcement related to networks that their customers are not allowed to announce. We suspect that this level of control was missing at the Spectrum router. Otherwise, AS 398994 would not have been able to hijack routes belonging to 449 Spectrum networks.

However, simple controls are not enough for transit AS. If any of them were setting up a list of networks that AS10796 was allowed to announce, that would still not have stopped the spreading of the hijack in the wild. Indeed, AS10796 was the original owner of the networks and signed most of its routes in RPKI, including 431 networks out of the 449 being hijacked.

Dropping routes found invalid via RPKI checks would have been a solution to mitigate the hijack spread. However, even that would not have stopped the hijack itself.

Third, but not least, network administrators must take measures against events like this one by investing in 24/7 BGP monitoring tools. At Catchpoint, we inform customers about hijacks within only a few seconds.

Published on Sep 09, 2021

Conclusion

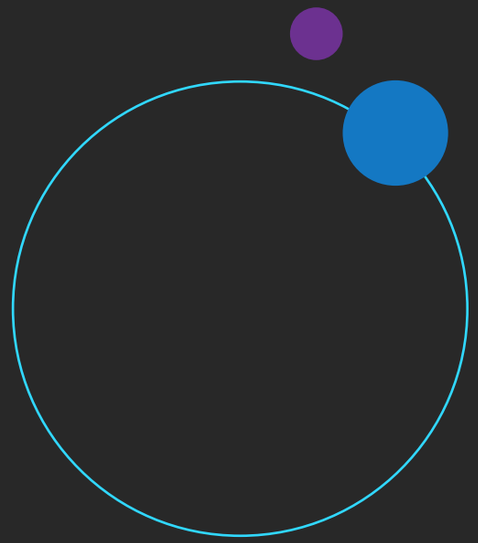
Website downtime can happen to anyone at any time. The costs to business and reputation can be profound.

In a complex digital landscape, which is increasingly reliant on the cloud and Internet for businesses to function, you need to be prepared for any kind of downtime or latency. It doesn't matter if the issue is caused by a bug, a misconfiguration, your cloud provider, an ISP going down, or an issue with one of your providers — if you don't know the source of the problem and what to do next, you'll be hard pressed to minimize damage to your business.

Deep visibility into the full digital experience is essential. With an Internet Performance Monitoring platform built by the experts for the experts, you can take proactive steps to correct issues as quickly as possible, in the moment, and improve your response in the future.

As you develop and improve your IPM strategy, ensure your visibility perspective allows you to gather as much data as possible: from every part of the Internet Stack that impacts every organization — from issues affecting internal networks, cloud providers, or workforce productivity — to BGP routing and global CDN traffic optimization.

In this way, you can respond quickly to issues and be proactive about communication to your own customers. In addition, you can verify that your providers are living up to the SLAs you've contracted. If not, you can demand restitution, switch providers, or add better failover options. Internet Resilience is no longer an option. It's a business imperative for 2023.



About Catchpoint

Catchpoint is the Internet Resilience Company™. The top online retailers, Global2000, CDNs, cloud service providers, and xSPs in the world rely on Catchpoint to increase their resilience by catching any issues in the Internet Stack before they impact their business. Catchpoint's Internet Performance Monitoring (IPM) suite offers synthetics, RUM, performance optimization, high fidelity data and flexible visualizations with advanced analytics. It leverages thousands of global vantage points (including inside wireless networks, BGP, backbone, last mile, endpoint, enterprise, ISPs and more) to provide unparalleled observability into anything that impacts your customers, workforce, networks, website performance, applications and APIs. Learn more at www.catchpoint.com.

Stay up to date on the most recent outages

Explore Our Blog

catchpoint