PL
POWERING FIBER WITH GIS
CHR
IMPROVING CX WITH AI
ETIYA
THE NEW DX FRONTIER
PREVENTING NETWORK OUTAGES
MOBILE-FIRST OMNICHANNEL CX
STEAMING CONTENT TO THE EDGE
SCALING NETWORKS AND SERVICES
SIMPLIFYING CX COMPLEXITY
CloudSmartz
COMMON SENSE CX
DATA-DRIVEN CUSTOMER EXPERIENCES
DIGITAL CX

www.pipelinepub.com
Volume 19, Issue 5

Six Ways to Prevent Network Outages in 2023

By: Dritan Suljoti

We are living in an era in which Internet resilience is critical. The last eighteen months have shown us that outages are a fact of life on today's Internet – whether a micro-outage taking down the checkout page on the mecca of eCommerce sites, Amazon, for two days of intermittent failures in the lead up to Christmas 2022 or the mega outage that took down Facebook, Messenger, WhatsApp and Instagram on October 4, 2021, costing significant revenue losses and untold damages to reputation and brand. That's why we see companies looking to detect issues faster and at a much earlier stage – ideally before end users are impacted.

At Catchpoint, we think of Internet resilience as the combination of four factors: availability, reachability, performance, and reliability. As part of this, we proactively monitor our customers' sites and services for outages 365/24/7 and create industry benchmarks for public use when we publish our findings and analysis of significant outages.

Drawing on years of experience supporting our customers (who have some of the world's busiest websites) and instituting our own incident management policies, we've put together some key lessons learned from recent failures. What follows are six ways to prevent outages in 2023.

Assume no service is immune to failure

IT must operate by assuming that no service is too big to fail. There is a fallacy that just because a service is very big and widely used, it won't go down. Over the last eighteen months, however, we have seen failures happen to many Internet giants: Facebook, Salesforce, Amazon, Google Cloud, Spotify, Ticketmaster… the list goes on. Nobody is immune. At the start of this year, on January 11th, 2023, we saw the FAA's NOTAM system (needed for pilot safety measures) go down, making headline news and massively disrupting air travel. Despite the fact the outage was dealt with in around 90 minutes, chaos ensued with around 7,000 flights delayed and 1,100 canceled. It's hard to calculate the economic cost, but it's safe to assume that it would have been in the hundreds of millions. That may appear high, but Gartner analysis from 2014 (and still widely cited) put the average cost of an outage at $5,600 per minute (that's $6,700 in today's terms). Gartner also notes that large enterprises will see costs closer to $9,000 per minute (that's $11,000 today). These numbers don't factor in the long-tail impact of lost productivity or damage to reputation. Moreover, as per Dun & Bradstreet, 59 percent of Fortune 500 companies suffer from a minimum of 1.6 hours of downtime each week. Translate that to an average cost of $643,200 to $1,056,000, and you can see why it's so important to be proactive about preparing for when, not if, the next outage occurs.

Rethink what "you can't control"

There are certain things within your IT team's control: containers, VMs, hardware, code, service configurations, and so on. Typically, we have systems in place to monitor these individual components of the system stack, alongside other processes that allow us to pay significant attention to these areas, as we should.

When an outage happens, however, this type of monitoring (usually infrastructure monitoring, tracing, or logging) is not sufficient to get in front of the situation. Issues will happen in areas beyond your control. You can't ignore this. You must plan and solve for it. Moreover, the widescale move to cloud-based applications over the last several years has made it increasingly challenging to determine where issues lie, which can leave companies at the mercy of third-party providers and networks. Without access to independent monitoring sources that can see across the Internet landscape, IT teams are left with poor to no visibility into components that are a critical part of their infrastructure, but outside their production environments. What may feel outside a company's control, however, with proper planning, best practice monitoring and observability techniques, and robust relationship-building with third parties.

Institute Internet Performance Monitoring

Instituting Internet Performance Monitoring (IPM) will allow you to monitor every component within the Internet stack that impacts your business yet might traditionally be perceived as beyond your control, or not worth considering because we think the system won't break (since we didn't make a change to it). That means not enough attention is given to BGP, TCP configuration, DNS, SSL certificates, the networks our data travels along, or indeed any of the points of failure in the infrastructure we rarely alter. Since cloud has abstracted much of the underlying network from ops, network and dev teams, this problem has been compounded. It can make it much harder to perceive a problem at all and when an issue does occur due to one of these fundamental components, we are caught by surprise. To avoid this, you need to continuously monitor across the Internet stack using IPM and put a plan into place so that when an issue does occur, you're prepared. This brings us to point four.

Implement a proper change management process

Most outages are the result of a change in code or configuration, either performed manually by an employee or the accidental result of automation. Either way, something changed in the system that led to failure. The easiest way to prevent this would be not to make any changes. However, as businesses (and the systems that support them) know, change is necessary to scale and grow.

Because change will occur and thus failure will, too, the best thing you can do is institute a rigorous change management system. What does this mean? First, ensure your team members know exactly what to do if a failure happens due to a change, and where to turn (under high pressure) if they don't. This involves forward planning and testing so that if change breaks a system, there is a plan in place for the team to follow. Back to point one, every moment spent on determining the root cause of an issue costs money. Second, track every change. Third, test every change before it is deployed. Fourth, while implementing change, continuously monitor key services, transactions, and outputs to understand what negative impact the change may be having so that you can immediately address it.

Develop an observability plan beyond logs and tracing

Observability is about more than just logs and tracing. It's also about baselining and understanding trends over time. Ensure you are not only looking at information during and after an incident, but also drawing on observability to analyze trends, for instance, by establishing baselines before an outage hits so that you know what to compare it to. This might include understanding (i) how the time of day or week impacts the performance of your application or service; (ii) how long a DNS lookup takes for your DNS vendor so that if they schedule a maintenance window and update the system, you have a benchmark to compare it against; or (iii) if you're updating your network device firmware, is it now dropping connections or adding latency to each packet sent?

Practice (and then practice again)

This is perhaps the most important lesson of all. You cannot be proactive enough in preparing for when the next outage will hit. Many of the teams we saw experience outages over the last 18 months were not prepared. This meant that when an issue happened, it took too long to identify what the issue was and then pinpoint the cause, making the Mean Time to Repair (MTTR) far slower than it could have been.

A final note of caution. Many of us are now reliant on the cloud not just for hosting our infrastructure, but also for services that our developers would have previously coded and maintained. When a key service of a major cloud provider goes down, the ripple effect across other products and companies can lead to a massive chain of failures. We saw this in November 2021 with Google Cloud and again, in AWS' trifecta of outages, in December 2021. We may not even realize that we could be the downstream victim of another company's failure, but our mutual interdependence on a handful of key vendors for important services like hosting or DNS services makes it essential that we plan not only for our own failures, but also those of the third parties that underpin our services and applications. Remember, with careful planning, rigorous monitoring and observability practices, and a thorough change management plan, what can feel beyond your control can actually lead to fast resolution with minimal business impact.