Service Chaining and Unintended Consequences

Lori MacVittie, 2013-13-02

#webperf #infosec #ado #SDN #cloud

Service chaining is a popular term today to describe a process in the network that's been done in the land of application integration for a long time. Service chaining in a nutshell is basically orchestration of network services. This concept is being put forth as the way future data center networks will be designed and execute in the future.

Its unintended consequence is, of course, that chaining can have a profound impact on performance, particularly when (or if) those chains extend across providers.

Let's consider an existing service chaining example that's challenging for SSL in terms of performance.

The Rest of the "SSL Performance" Story

Now, we're all aware that SSL handshaking introduces latency. It has to because in addition to the already time-consuming process of performing cryptographic functions, it requires additional round trips between the client (browser) and server (or intermediate network proxy acting as the endpoint, such as a load balancer or ADC) to exchange the information needed to encrypt and decrypt subsequent communication.

But that's not all it needs to do. The certificate offered up by the server-side device is increasingly suspect thanks to a variety of incidents in which basically forged certificates were used to impersonate a site and trick the user into believing the site was safe. As the SSL Everywhere movement continues to grow, so has the decision by browsers to properly validate certificates by querying an OCSP (Online Certificate Status Protocol) responder as to the status of the certificate (this is increasingly favored over the use of CRL (Certificate Revocation Lists) to address certain shortcomings of the technology).

What this means is that during the SSL handshake, the client makes a request to an OCSP responder. It's an additional service in the connection chain that adds time to the "load" process. Thus, it needs to be as fast as possible because it's counted in the "load time" for a page, if not technically then from the perspective of the user which, as we all know, is what really counts.

So the browser makes a request to the responder. It does this by choosing a responder from a list of those that support the CA (Certificate Authority, the issuer of the certificate in question). While there are a large number of global CAs, the actual number of global CAs for SSL is fairly small. Thus the responder is almost certainly very large and likely to see billions of requests a day, from around the globe. This "link in the chain" is increasingly important to the overall performance experienced by the end-user. Its impact on mobile users, in particular, is worthy of note given the impact of mobile networks and constrained device capabilities, as noted by Mike Belshe, one of the folks who helped create the SPDY protocol (emphasis mine):

> But this process is pretty costly, especially on mobile networks. For my own service, I just did a quick trace over 3G:
>
> - DNS (1334ms)
> - TCP handshake (240ms)
> - SSL handshake (376ms)
> - **Follow certificate chain (1011ms) — server should have bundled this.**
> - **DNS to CA (300ms)**
> - **TCP to CA (407ms)**
> - **OCSP to CA #1 (598ms) — StartSSL CA uses connection close on each!**
> - **TCP to CA #2 (317ms)**
> - **OCSP to CA #2 (444ms)**
> - Finish SSL handshake (1270ms)
>
> -- Rethinking SSL for Mobile Apps

The emphasized portions of the transaction indicate those related to the certificate verification process being carried out by the browser as a security precaution. Over a non-mobile network, one would expect the performance to improve, but the impact on "regular" browsers should not be underestimated, either. Early last year Adam Langley noted this and proposed to disable OSCP validation in Chrome: .

> The median time for a successful OCSP check is ~300ms and the mean is nearly a second. This delays page loading and discourages sites from using HTTPS. They are also a privacy concern because the CA learns the IP address of users and which sites they're visiting.
>
> On this basis, we're currently planning on disabling online revocation checks in a future version of Chrome.
>
> http://www.imperialviolet.org/2012/02/05/crlsets.html

I'll save the security-related arguments for another time, but suffice to say that the impact of service chaining on performance in the case of SSL and certificate validation is significant enough at times to be noticed.

Key Takeaway

Now certainly service chaining in other contexts, say in the data center network, would not experience the same magnitude of delay based purely on the fact that we're talking about LAN speeds rather than what often end up being inter- or cross-continental communications. Still, the very real impact of service chaining, particularly when such chains are comprised of a long string of services, should not be ignored or underestimated. Such chains introduce additional latency, often in the form of unnecessary, duplicated functions as well as the possibility of failure. Load and utilization monitoring and scaling strategies of individual (dependent) services is a vital to the overall success of any architecture which employs an orchestrated (chained) services strategy.

$$\sum_{i=1}^{perf(n)} i$$

And while technologies like SDN and cloud offer corrective action in the face of failure, it should be noted that such corrections tend to be **reactions to** failure. That means at least one user experiences a failure before a correction is made. In some cases that failure will go unnoticed except for a lengthier response time, but the key takeaway there is that it is **noticeable**.

And when it comes to web application performance, *noticeable* degradations are not something the business or operations, for that matter, likes to see. Not even for a single user.

F5 Networks, Inc. | 401 Elliot Avenue West, Seattle, WA 98119 | 888-882-4447 | f5.com

F5 Networks, Inc.
Corporate Headquarters
info@f5.com

F5 Networks
Asia-Pacific
apacinfo@f5.com

F5 Networks Ltd.
Europe/Middle-East/Africa
emeainfo@f5.com

F5 Networks
Japan K.K.
f5j-info@f5.com