Towards Cross-Layer Telemetry

Justin Iurman, Frank Brockners, Benoit Donnet



ANRW'21, July 26-30 2021 (virtual)

Microservices



Still confident to debug this one?



Application Performance Management

- Distributed tracing tools (e.g. Jaeger)
- Very useful for (spaghetti) microservices







Application Performance Management

- BUT... what if my DB lookup is slow?
 - Is it the app?
 - Is it the server, or the DB?
 - Is it a network issue?
 - Is it Chuck Norris' fault?

 \rightarrow We don't know **why** or **where** the problem *exactly* is, we only know it is *slow*.



Cross-Layer Telemetry

- <u>Solution</u>: include L3-L4 in tracing tools visibility
- (1) Find a way to correlate APM (e.g. Jaeger) traces with network traffic
 - use IOAM to carry both trace and span IDs in packets



- (2) Find when and how these IDs should be injected
 - (when?) at socket creation vs when sending data
 - (how?) send* API modification vs new syscall vs netlink call



Cross-Layer Telemetry



Trace Start April 22 2021, 01:49:07.015 | Duration 201.52ms | Services 1 | Depth 1 | Total Spans 2

Oms		50.38ms		100.76ms			151.14ms		201.52ms
				•					
Service & Operation	\lor > \Leftrightarrow »	0ms	50.38	Bms		100.76ms		151.14ms	201.52ms
Login API span_login									
	span_login						Service: Login API Duration: 201.52ms Start Time: 0ms		
		> Tags: internal.span.format = proto sampler.param = true sampler.type = const							
		> Process: hostname = c297503f35c5							
									SpanID: 5da04ac4fe5c355d 🔗
Login API ioam_span	0.01ms i				I				
	ioam_span					Service: Login API Duration: 0.01ms Start Time: 102.23ms			
		✓ Tags							
		internal.span.format "proto"							
		<pre>ioam_namespace123_node1 "Hop_Limit=64; Node_Id=1; Ingress_Id=65535; Egress_Id=11; Egress_Queue_Depth=0; "</pre>					(Entry Point)		
		ioam_namespace123_node2 "Hop_Limit=63; Node_Id=2; Ingress_Id=21; Egress_Id=22; Egre				ess_Queue_Depth=897	(Router)		
		<pre>ioam_namespace123_node3 "Hop_Limit=62; Node_Id=3; Ingress_Id=31; Egress_Id=65535; Egress_Queue_Depth=0; "</pre>				(Server)			
		sampler.param	true						

Conclusion

- Hot topic in the industry
- CLT solves challenges in the (microservice) tracing world
- Next steps to improve it
- Github: <u>https://github.com/iurmanj/cross-layer-telemetry</u> (+ demo video)

Thank you!

justin.iurman@uliege.be