



**ATS
Capacity Tuning
&
Concurrency
Limiter**

Sudheer Vinukonda
June 2020

ATS Capacity Tuning - Today

- ❖ Overall connection Limit
 - Includes inbound, outbound
 - Needs to be set very high
- ❖ Max Outbound Connections
- ❖ Transaction, Connection timeouts
 - Inactivity/Active Timeouts
 - Malicious, rogue or broken client, network behavior
- ❖ Connection Limits per Origin
 - Service latency issues or failures

- proxy.config.net.connections_throttle: 300000
- proxy.config.http.server_max_connections: 60000
- proxy.config.http.transaction_no_activity_timeout_in: 75
- proxy.config.http.transaction_no_activity_timeout_out: 75
- proxy.config.http.transaction_active_timeout_in: 180
- proxy.config.http.transaction_active_timeout_out: 0

ATS Capacity Tuning - Today

Feature	Pros	Cons
Overall Connection Limit	<ul style="list-style-type: none">● Works for HTTP/1.1 Traffic	<ul style="list-style-type: none">● Concurrency is vastly different for HTTP/2 due to request multiplexing
Connection/Transaction Timeouts	<ul style="list-style-type: none">● Protects against Runaway Transactions, Connection leaks etc● Efficient resource utilization	<ul style="list-style-type: none">● Needs to be tuned relatively low to optimize for capacity/throughput● Upload, long poll etc use cases require higher timeouts
Origin Connection Limits	<ul style="list-style-type: none">● Isolation/resiliency against failures or latency spikes in individual Services	<ul style="list-style-type: none">● “Too late” because majority of the “work” is already completed

ATS Capacity Tuning - Concurrency Limiter..

- ❖ Available resources/capacity
 - CPU, Memory, Network bandwidth
- ❖ Using active connection limits
 - Works well for HTTP/1.1
 - HTTP/2 poses a challenge with Multiplexing
- ❖ Using Max Active requests (Concurrency Limiter)
 - Streams in HTTP/2
- ❖ Reduces/eliminates dependency on timeout configs
 - Active/Inactive/keep-alive timeouts
 - Hard to get right

ATS Capacity Tuning - Changes

Current Config	Proposed Config	Context
<code>proxy.config.net.max_active_connections_in</code>	<code>proxy.config.net.max_requests_in</code>	max number of in-flight inbound requests (concurrency)
<code>proxy.config.http.server_max_connections</code>	<code>proxy.config.net.max_connections_out</code>	max number of outbound connections (active + keepalive)
<code>proxy.config.net.max_connections_in</code>	<code>proxy.config.net.max_connections_in</code>	max number of in-flight requests (concurrency) + idle (socket) connections
<code>proxy.config.net.connections_throttle</code>	<code>proxy.config.net.connections_throttle</code> Eventually deprecate and replace with { "proxy.config.net.max_connections_in" + "proxy.config.net.max_connections_out" }	max number of open connections (FD, rlimits etc)

ATS Capacity Tuning - Changes

- ❖ Should connection be closed when throttling a request?
 - Status code 503?
 - HTTP/1.1
 - Connection: close
 - HTTP/2
 - GoAway
- ❖ Internal requests (PluginVC/FetchSM)?
- ❖ Anything else...?