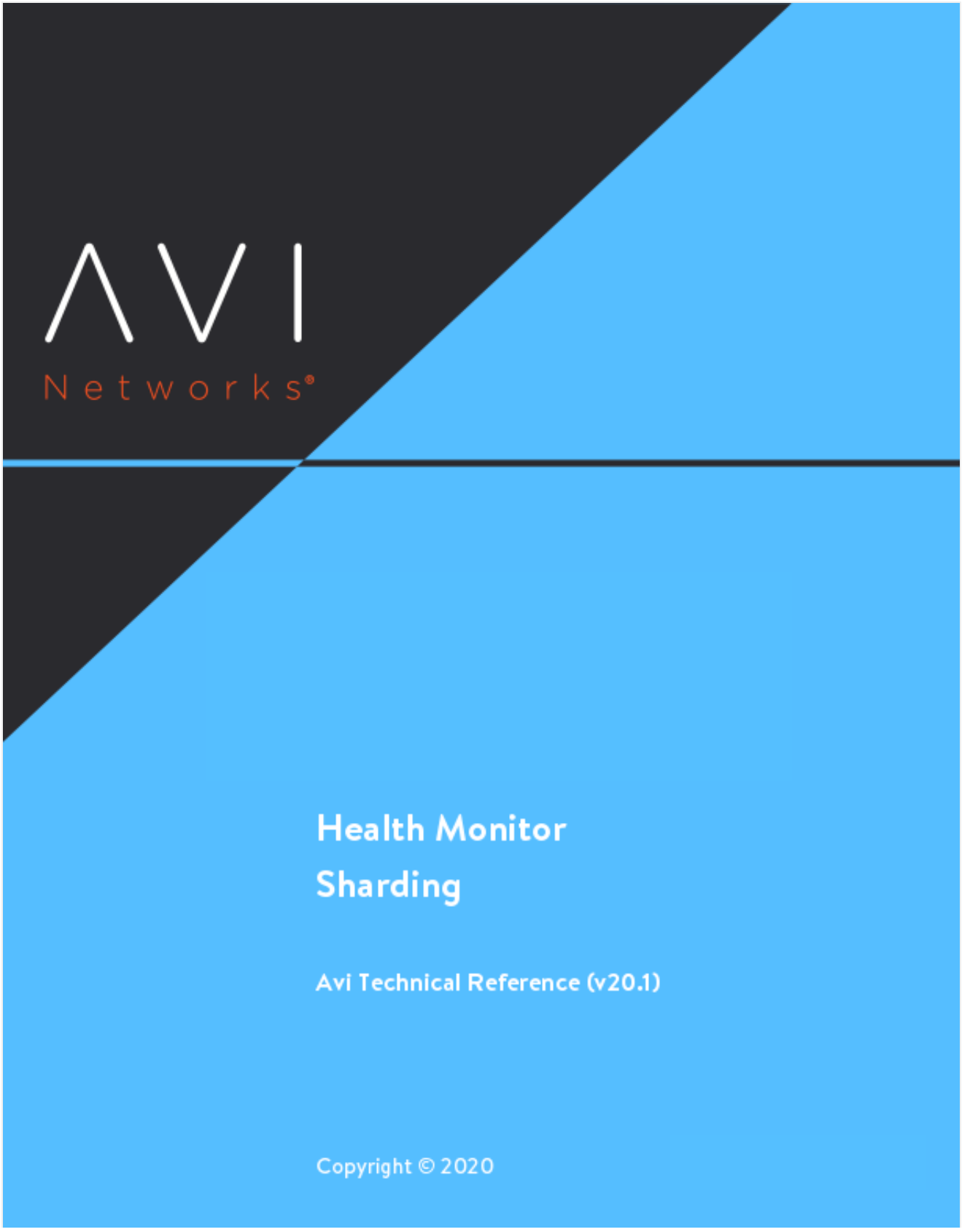

Health Monitor Sharding

Avi Technical Reference (v20.1)

Health Monitor Sharding

view online

Overview

Health monitor sharding enables selective monitoring for SEs on Avi Vantage. With this feature, all the SEs do not have to participate in health monitoring. Starting with release 18.2.3, health monitor sharding is supported for GSLB services. Using this feature, the load on DNS virtual service is reduced too.

When health monitoring (HM) sharding is not enabled and datapath monitoring is enabled for GSLB services, all the Service Engines where the DNS virtual service is placed are responsible for monitoring the GSLB pool members.

For example, if there are 1000 GSLB services and if a DNS virtual service is placed over 2 Service Engines, both the SEs would health monitor all 1000 GSLB services based on the configuration. If there are multiple DNS virtual services involved for a particular domain, then all the SEs perform health probing. In the case, where a DNS VS is present on multiple sites, there will be probing done from other DNS SEs also.

Use Case

- It is useful in the deployments where SEs are deployed in huge numbers.
- Resiliency within a system within a site. Without HM sharding, resiliency within a site in case of DNS is not cost-effective as another SE with the same configuration is required. Another option is to have DNS virtual services across sites, i.e., site-level resiliency. But, this required a third-party application (for example, Infoblox) to monitor the availability of the Avi DNS VS(s) and remove it(them) from active resolvers if the DNS-VS goes down.
- To reduce the load on the back-end system as selective monitoring is performed.

How it Works

A shard server runs on the Avi Controller leader and a shard client runs on each SE.

Status is shared between SEs by the Avi Controller. Each SE is responsible for a certain number of services.

Example: There are 10k GSLB services, each with one health monitor. A DNS virtual service is placed over four SEs, so each SE would be roughly responsible for monitoring 2500 GSLB services.

- *SE1*- 1 to 2500 GSLB services
- *SE2* - 2501 to 5000 GSLB services
- *SE3* - 5001 to 7500 GSLB services
- *SE4*- 7501 to 10000 GSLB services

Note: The order mentioned above is for illustration purpose only.

HM sharding feature reduces heath-monitor traffic or load by reducing the health-monitor probes across multiple DNS(es) and SEs within a DNS virtual service for a domain.

Status propagation of GSLB services across SEs

SEs require updated health monitoring state to correctly process the DNS requests. Each SE performs health monitoring for a set of GSLB services. It will also propagate this information to the state cache manager (SCM). The state cache manager (SCM) propagates status of the GSLB services across the SEs.

The following assumptions are considered to explain the feaute:

- *SE1* executes health monitor probes for GSLB services *Gs1-Gs1k*.
- *SE2* may be executing health-monitor probes for GslbServices *Gs1k ? Gs2k*.
- *SE1* needs the status of GSLB services *Gs1k-Gs2K*.
- *SE2* needs the status of GSLB services *Gs1-Gs1K().
- SCM propagates the statuses from *SE1* and *SE2*.

The SCM has the information of SEs to which the information has to be propagated. This information is retrieved from the shard server (SS) by registering for this information. The SS propagates the shard map information to the SCM whenever there is a change.

SEs in Headless State

Assume that *SE1* goes headless. When a SE goes headless, it waits for the time equal to the `send_interval` time configured. It is amount of time SE waits before declaring itself headless. After the `send_interval` time is expired, SE1 will start monitoring all the GSLB services. This is done to maintain the correct state of GSLB pool members.

The moment Avi Controller sees state of given member is changed , it immediately tries to push the state to other SEs

The `send_interval` time is a common knob which controls many functionalities.

On the SE, it defines:

- Debounce timer for a SE to react to headless behavior
- Batch incoming messages from shard server to improve responsiveness during warm boot. For example, if a DNS-VS is placed on n SEs, then during a warm boot the SE can receive n-1 messages. All n-1 messages arrive within the send interval are batched and processed in one go.

Scale Out

Assumption:
To demonstrate a scale-out event, it is assumed that DNS virtual service is placed over 4 SEs. In the case of a scale-out event, virtual services are placed on a new SE (on the fifth SEs). Whenever virtual service placement happens, the new SE advertises request for registering for the configured domain.

Once the Avi Controller receives this message, it broadcasts the message for domain *xyz.com*. There are now 5 potential receivers (SE1 to SE5). Since a new SE is added now, that new SE needs to know which GSLB services it should monitor. The SE which is health monitoring a GSLB service is the owner of that GSLB service and other SEs are watchers for the same. Each SE maintains consistent hash, which it uses to determine if there are any state changes. If A lookup is performed for consistent hash, it will reveal the SE owner for a given GSLB service. The moment the consistent hash is changed, all SEs come to same conclusion.

In this example,after re-computation of the consistent hash, 5SEs are available for health monitoring. With 5 SEs, each SE will monitor 2000 GSLB services instead of 2500.

Whenever a new resource added, it does not create a complete remap of the results. It only affects delta of result.

The scale-out time is directly proportional to the number of GSLB services.

Scale in

In case of scale in, recomputing consistent hash when the number of points changes remains the same.

Scale-in time is constant; it does not vary linearly with the number of GSLB services. However, the convergence time would vary in this case and would be longer as compared to the scale-out scenario

Notes:

- Neither the Avi Controller nor SE makes a sharding decision when a new GSLB service is configured. The SE will instantiate the GSLB service and unconditionally run the health-monitor probes.
- If both the DNS virtual services are responsible for different domains, i.e., DNS VS 1 is for *xyz.com* whereas DNS VS2 is for *abc.com* then the sharding decision for DNS 1 will be independent of DNS VS2.
- To ensure that we do not overwhelm the system in the presence of flappy/unstable connections , we have a mechanism where any kind of state update will be consumed after the send interval. If a SE gets the message from the Controller that the shard map has been changed, it won?t consume that right away. It will start a timer of send_interval duration and wait for that time. This is to make sure that event has actually happened. After the send_interval has passed, it will recompute the hash and get to know if state has actually changed or not. This is to make sure that a momentary event does not churn the entire system. Consider a scenario where a SE is flapping after every 10-12 second. If with each flap we recalculate the hash and change the state, then it will make the entire system unstable.

Configuring Health Monitor Sharding

Using Avi UI

Health Monitor Sharding can be enabled per site basis. 1. To enable the feature for the desired site, navigate to Infrastructure > GSLB, and select the desired GSLB site.

2. Select the Save and Set DNS Virtual Services as shown below.

3. Enable the check box for HM Sharding as shown below.

Using Avi CLI

Login to the Avi Controller and set value of the `hm_shard_enabled` flag to true under the `configure gslb` mode as shown below.

```
[admin:Avi-Controller-2]: > configure gslb Default
[admin:Avi-Controller-2]: gslb> edit

Change  hm_shard_enabled to True and save the configuration

[admin:Avi-Controller-2]: gslb> save
```

