OX PowerDNS Cloud Control

Reference

Feb 16, 2021

Release 1.0.0

Contents

1 Global

1.1 dnsdist

dnsdist defaults

- **dnsdistConsistentHashingBalancingFactor**: Set the maximum imbalance between the number of outstanding queries intended for a given server, based on its weight, and the actual number, when using the chashed consistent hashing load-balancing policy. Default is 0, which disables the bounded-load algorithm. (default: *0*)

- **dnsdistDo53Locals**: Default amount of Do53 listen sockets per dnsdist pod (default: *1*)

- **dnsdistDo53TcpFastOpenQueueSize**: Default size of the TCP Fast Open queue on Do53 listen sockets (Dnsdist default)

- **dnsdistDo53TcpListenQueueSize**: Default size of the listen queue on Do53 listen sockets (Dnsdist default)

- **dnsdistPort**: Port on which dnsdist will listen for Do53 traffic (default: *5353*)

- **dnsdistReadinessDo53ProbeInterval**: How often (in seconds) the agent will judge the health state of dnsdist via tests against the Do53 listeners. (default: *2*)

- **dnsdistReadinessDo53QDomain**: Domain used in the query to judge whether the Do53 listeners of a dnsdist container are healthy and ready for traffic. (default: *a.root-servers.net*)

- **dnsdistReadinessDo53QTimeout**: Number of seconds after which the query used to judge whether the Do53 listeners of a dnsdist container are healthy and ready for traffic time out. (default: *1*)

- **dnsdistReadinessDo53QType**: Type of query used to judge whether the Do53 listeners of a dnsdist container are healthy and ready for traffic. (default: *A*)

- **dnsdistReadinessFailureThreshold**: When a probe fails, Kubernetes will try this many times before marking the container as Unready. Updates deployment, resulting in respawn of dnsdist pods (default: *2*)

- **dnsdistReadinessInitialDelaySeconds**: Number of seconds after the containers have started before readiness probes are initiated. Updates deployment, resulting in respawn of dnsdist pods. (default: *5*)

- **dnsdistReadinessPeriodSeconds**: How often (in seconds) to perform the readiness probes. Updates deployment, resulting in respawn of dnsdist pods (default: *2*)

- **dnsdistReadinessStateProbeInterval**: How often (in seconds) the agent will judge the health state of the dnsdist agent. (default: *2*)

- **dnsdistReadinessSuccessThreshold**: Minimum consecutive succesfull probes before a container is marked Ready. Updates deployment, resulting in respawn of dnsdist pods (default: *2*)

- **dnsdistReadinessTimeoutSeconds**: Number of seconds after which the readiness probes time out. Updates deployment, resulting in respawn of dnsdist pods (default: *1*)

- **dnsdistReplicas**: Default number of replicas in a dnsdist deployment (default: *2*)

- **dnsdistRevisionHistoryLimit**: Default 'revisionHistoryLimit' for dnsdist deployments (default: *0*)

- **dnsdistRoundRobinFailOnNoServer**: By default the roundrobin load-balancing policy will still try to select a backend even if all backends are currently down. Setting this to true will make the policy fail and return that no server is available instead. (default: *false*)

- **dnsdistServFailWhenNoServer**: If true, return a ServFail when no servers are available, instead of the default behaviour of dropping the query. (default: *false*)

- **dnsdistServiceAccount**: Name of the service account used by dnsdist (default: *dnsdist-serviceaccount*)

- **dnsdistWebserverACL**: Netmasks to allow webserver traffic from (default: *127.0.0.1/32, 192.168.0.0/16, 10.0.0.0/8, 172.16.0.0/12*)

- **dnsdistWebserverPort**: Port on which the dnsdist webserver will listen (default: *8083*)

- **dnsdistWeightedBalancingFactor**: Set the maximum imbalance between the number of outstanding queries intended for a given server, based on its weight, and the actual number, when using the whashed or wrandom load-balancing policy. Default is 0, which disables the bounded-load algorithm. (default: *0*)

- **dnsdistAgentResources** (dnsdist agent resource allocation defaults)

  – **limits** (Limit amounts)

    * **cpu**: Limit amount of CPU (Kubernetes default)

    * **memory**: Limit amount of memory (Kubernetes default)

  – **requests** (Request amounts)

    * **cpu**: Request amount of CPU (Kubernetes default)

    * **memory**: Request amount of memory (Kubernetes default)

- **dnsdistECS** (dnsdist ECS functions)

  – **setECSOverride**: Whether to override an existing EDNS Client Subnet option present in the query (only effective when the backend server has annotation useClientSubnet = true). (Dnsdist default)

  – **setECSSourcePrefixV4**: Truncate the requestors IPv4 address to this length, in bits (only effective when the backend server has annotation useClientSubnet = true). (Dnsdist default)

  – **setECSSourcePrefixV6**: Truncate the requestors IPv6 address to this length, in bits (only effective when the backend server has annotation useClientSubnet = true). (Dnsdist default)

- **dnsdistRPCServerResources** (dnsdist RPC server resource allocation defaults)

- **limits** (Limit amounts)

  * **cpu**: Limit amount of CPU (Kubernetes default)

  * **memory**: Limit amount of memory (Kubernetes default)

- **requests** (Request amounts)

  * **cpu**: Request amount of CPU (Kubernetes default)

  * **memory**: Request amount of memory (Kubernetes default)

- **dnsdistResources** (dnsdist resource allocation defaults)

  - **limits** (Limit amounts)

    * **cpu**: Limit amount of CPU (Kubernetes default)

    * **memory**: Limit amount of memory (Kubernetes default)

  - **requests** (Request amounts)

    * **cpu**: Request amount of CPU (Kubernetes default)

    * **memory**: Request amount of memory (Kubernetes default)

- **dnsdistRingBuffers** (Ringbuffer configuration)

  - **retries**: Number of shards to attempt to lock without blocking before giving up and simply blocking while waiting for the next shard to be available. Defaults to 5 if there is more than 1 shard, else it defaults to 0. (Dnsdist default)

  - **shards**: Number of shards to use to limit lock contention, defaults to 1. (Dnsdist default)

  - **size**: Maximum amount of queries to keep in the ringbuffer, defaults to 10000. (Dnsdist default)

- **dnsdistSecurityPolling** (Security polling configuration)

  - **securityPollInterval**: Interval between security pollings, in seconds. Defaults to 3600. (Dnsdist default)

  - **securityPollSuffix**: Domain name from which to query security update notifications. Setting this to an empty string disables secpoll. (Dnsdist default)

- **dnsdistStateResources** (dnsdist state resource allocation defaults)

  - **limits** (Limit amounts)

    * **cpu**: Limit amount of CPU (Kubernetes default)

    * **memory**: Limit amount of memory (Kubernetes default)

  - **requests** (Request amounts)

    * **cpu**: Request amount of CPU (Kubernetes default)

    * **memory**: Request amount of memory (Kubernetes default)

- **dnsdistTuning** (dnsdist tuning functions)

  - **setCacheCleaningDelay**: Interval in seconds between two runs of the cache cleaning algorithm, removing expired entries. (Dnsdist default)

- **setCacheCleaningPercentage**: Percentage of the cache that the cache cleaning algorithm will try to free by removing expired entries. By default (100), all expired entries are removed. (Dnsdist default)

- **setMaxTCPClientThreads**: Maximum amount of TCP client threads, handling TCP connections. By default this value is 10, unless more than 10 TCP listen sockets have been defined. (Dnsdist default)

- **setMaxTCPConnectionDuration**: Maximum duration of an incoming TCP connection, in seconds. 0 (the default) means unlimited. (Dnsdist default)

- **setMaxTCPConnectionsPerClient**: Maximum number of TCP connections per client. 0 (the default) means unlimited. (Dnsdist default)

- **setMaxTCPQueriesPerConnection**: Maximum number of queries in an incoming TCP connection. 0 (the default) means unlimited. (Dnsdist default)

- **setMaxTCPQueuedConnections**: Maximum number of TCP connections queued (waiting to be picked up by a client thread), defaults to 1000. 0 means unlimited. (Dnsdist default)

- **setMaxUDPOutstanding**: Maximum number of outstanding UDP queries to a given backend server, defaults to 65535. (Dnsdist default)

- **setStaleCacheEntriesTTL**: TTL for expired cache entries to be eligible as answer when no backends are available for a query, in seconds. (Dnsdist default)

- **setTCPRecvTimeout**: Read timeout on TCP connections from the client, in seconds. (Dnsdist default)

- **setTCPSendTimeout**: Write timeout on TCP connections from the client, in seconds. (Dnsdist default)

- **setTCPUseSinglePipe**: Whether the incoming TCP connections should be put into a single queue instead of using per-thread queues. Defaults to false. (Dnsdist default)

- **setUDPTimeout**: Maximum time dnsdist will wait for a response from a backend over UDP, in seconds. Defaults to 2. (Dnsdist default)

1.2 recursor

Recursor defaults

- **recursorContainerPort**: Port on which recursor will listen for traffic (default: *5353*)

- **recursorReplicas**: Default number of replicas in a recursor deployment (default: *2*)

- **recursorRevisionHistoryLimit**: Default 'revisionHistoryLimit' for recursor deployments (default: *0*)

- **recursorServiceAccount**: Name of the service account used by recursor (default: *recursor-serviceaccount*)

- **recursorWebserverACL**: Netmasks to allow webserver traffic from (default: *127.0.0.1/32, 192.168.0.0/16, 10.0.0.0/8, 172.16.0.0/12*)

- **recursorWebserverPort**: Port on which the dnsdist webserver will listen (default: *8083*)

- **recursorResources** (recursor resource allocation defaults)

- **limits** (Limit amounts)
    * **cpu**: Limit amount of CPU (Kubernetes default)
    * **memory**: Limit amount of memory (Kubernetes default)
- **requests** (Request amounts)
    * **cpu**: Request amount of CPU (Kubernetes default)
    * **memory**: Request amount of memory (Kubernetes default)

1.3 registrySecrets

Configuration of credentials of the Cloud Control image registry.

- **email**: Email address associated with the secret, can be a dummy value but may not be omitted. (default: *cloudcontrol@admin.company.com*)
- **password**: Password or token for authentication
- **registry**: FQDN of the image registry (default: *registry.open-xchange.com*)
- **username**: Username for authentication

1.4 resolver

Resolver defaults

- **resolverBackendDefaultPort**: Port on which the endpoints belonging to a resolver partition will receive traffic (default: *53*)
- **resolverServicePort**: Port on which a resolver service will listen (unused in normal operation, as the service is only used to discover the corresponding endpoints). (default: *53*)

2 Instances

2.1 dnsdists

Configuration of dnsdist instances

Key: name (Name of the dnsdist instance)

2.1.1 Parameters

- **do53Locals**: Default amount of Do53 listen sockets per dnsdist pod (default: *1*)
- **do53TcpFastOpenQueueSize**: Default size of the TCP Fast Open queue on Do53 listen sockets (Dnsdist default)
- **do53TcpListenQueueSize**: Default size of the listen queue on Do53 listen sockets (Dnsdist default)
- **replicas**: Default number of replicas in a dnsdist deployment (default: *2*)
- **revisionHistoryLimit**: Default 'revisionHistoryLimit' for dnsdist deployments (default: *0*)
- **verbose**: Be verbose (default: *false*)

2.1.2 Parameter Sets

- **config** (Dnsdist Configuration)

    – **addEDNSToSelfGeneratedResponses**: Whether to add EDNS to self-generated responses, provided that the initial query had EDNS. (Dnsdist default)

    – **allowEmptyResponse**: Set to true (defaults to false) to allow empty responses (qd-count=0) with a NoError or NXDomain rcode (default) from backends. dnsdist drops these responses by default because it can't match them against the initial query since they don't contain the qname, qtype and qclass, and therefore the risk of collision is much higher than with regular responses. (Dnsdist default)

    – **consistentHashingBalancingFactor**: Maximum imbalance between the number of outstanding queries intended for a given server, based on its weight, and the actual number, when using the chashed consistent hashing load-balancing policy. Default is 0, which disables the bounded-load algorithm. (Dnsdist default)

    – **payloadSizeOnSelfGeneratedAnswers**: Set the UDP payload size advertised via EDNS on self-generated responses. In accordance with RFC 6891, values lower than 512 will be treated as equal to 512. (Dnsdist default)

- **roundRobinFailOnNoServer**: By default the roundrobin load-balancing policy will still try to select a backend even if all backends are currently down. Setting this to true will make the policy fail and return that no server is available instead. (Dnsdist default)

- **servFailWhenNoServer**: If set, return a ServFail when no servers are available, instead of the default behaviour of dropping the query. (Dnsdist default)

- **verboseHealthChecks**: Set whether health check errors should be logged. This is turned off by default. (Dnsdist default)

- **weightedBalancingFactor**: Maximum imbalance between the number of outstanding queries intended for a given server, based on its weight, and the actual number, when using the whashed or wrandom load-balancing policy. Default is 0, which disables the bounded-load algorithm. (Dnsdist default)

- **ecs** (dnsdist ECS functions)

  - **setECSOverride**: Whether to override an existing EDNS Client Subnet option present in the query (only effective when the backend server has annotation useClientSubnet = true). (Dnsdist default)

  - **setECSSourcePrefixV4**: Truncate the requestors IPv4 address to this length, in bits (only effective when the backend server has annotation useClientSubnet = true). (Dnsdist default)

  - **setECSSourcePrefixV6**: Truncate the requestors IPv6 address to this length, in bits (only effective when the backend server has annotation useClientSubnet = true). (Dnsdist default)

- **readiness** (Readiness probe Configuration)

  - **do53ProbeInterval**: How often (in seconds) the agent will judge the health state of dnsdist via tests against the Do53 listeners. (default: *2*)

  - **do53QDomain**: Domain used in the query to judge whether the Do53 listeners of a dnsdist container are healthy and ready for traffic. (default: *a.root-servers.net*)

  - **do53QTimeout**: Number of seconds after which the query used to judge whether the Do53 listeners of a dnsdist container are healthy and ready for traffic time out. (default: *1*)

  - **do53QType**: Type of query used to judge whether the Do53 listeners of a dnsdist container are healthy and ready for traffic. (default: *A*)

  - **failureThreshold**: When a probe fails, Kubernetes will try this many times before marking the container as Unready. Updates deployment, resulting in respawn of dnsdist pods (default: *2*)

  - **initialDelaySeconds**: Number of seconds after the containers have started before readiness probes are initiated. Updates deployment, resulting in respawn of dnsdist pods. (default: *5*)

  - **periodSeconds**: How often (in seconds) to perform the readiness probes. Updates deployment, resulting in respawn of dnsdist pods (default: *2*)

  - **stateProbeInterval**: How often (in seconds) the agent will judge the health state of the dnsdist agent. (default: *2*)

- **successThreshold**: Minimum consecutive succesfull probes before a container is marked Ready. Updates deployment, resulting in respawn of dnsdist pods (default: *2*)

- **timeoutSeconds**: Number of seconds after which the readiness probes time out. Updates deployment, resulting in respawn of dnsdist pods (default: *1*)

- **resources** (dnsdist resource allocation defaults)

  - **limits** (Limit amounts)

    * **cpu**: Limit amount of CPU (Kubernetes default)

    * **memory**: Limit amount of memory (Kubernetes default)

  - **requests** (Request amounts)

    * **cpu**: Request amount of CPU (Kubernetes default)

    * **memory**: Request amount of memory (Kubernetes default)

- **ringBuffers** (Ringbuffer configuration)

  - **retries**: Number of shards to attempt to lock without blocking before giving up and simply blocking while waiting for the next shard to be available. Defaults to 5 if there is more than 1 shard, else it defaults to 0. (Dnsdist default)

  - **shards**: Number of shards to use to limit lock contention, defaults to 1. (Dnsdist default)

  - **size**: Maximum amount of queries to keep in the ringbuffer, defaults to 10000. (Dnsdist default)

- **securityPolling** (Security polling configuration)

  - **securityPollInterval**: Interval between security pollings, in seconds. Defaults to 3600. (Dnsdist default)

  - **securityPollSuffix**: Domain name from which to query security update notifications. Setting this to an empty string disables secpoll. (Dnsdist default)

- **service** (Configuration of the Kubernetes service object)

  - **type**: TODO Type of service (NodePort, ClusterIP, Loadbalancer, etc)

- **tuning** (dnsdist tuning functions)

  - **setCacheCleaningDelay**: Interval in seconds between two runs of the cache cleaning algorithm, removing expired entries. (Dnsdist default)

  - **setCacheCleaningPercentage**: Percentage of the cache that the cache cleaning algorithm will try to free by removing expired entries. By default (100), all expired entries are removed. (Dnsdist default)

  - **setMaxTCPClientThreads**: Maximum amount of TCP client threads, handling TCP connections. By default this value is 10, unless more than 10 TCP listen sockets have been defined. (Dnsdist default)

  - **setMaxTCPConnectionDuration**: Maximum duration of an incoming TCP connection, in seconds. 0 (the default) means unlimited. (Dnsdist default)

  - **setMaxTCPConnectionsPerClient**: Maximum number of TCP connections per client. 0 (the default) means unlimited. (Dnsdist default)

- **setMaxTCPQueriesPerConnection**: Maximum number of queries in an incoming TCP connection. 0 (the default) means unlimited. (Dnsdist default)

- **setMaxTCPQueuedConnections**: Maximum number of TCP connections queued (waiting to be picked up by a client thread), defaults to 1000. 0 means unlimited. (Dnsdist default)

- **setMaxUDPOutstanding**: Maximum number of outstanding UDP queries to a given backend server, defaults to 65535. (Dnsdist default)

- **setStaleCacheEntriesTTL**: TTL for expired cache entries to be eligible as answer when no backends are available for a query, in seconds. (Dnsdist default)

- **setTCPRecvTimeout**: Read timeout on TCP connections from the client, in seconds. (Dnsdist default)

- **setTCPSendTimeout**: Write timeout on TCP connections from the client, in seconds. (Dnsdist default)

- **setTCPUseSinglePipe**: Whether the incoming TCP connections should be put into a single queue instead of using per-thread queues. Defaults to false. (Dnsdist default)

- **setUDPTimeout**: Maximum time dnsdist will wait for a response from a backend over UDP, in seconds. Defaults to 2. (Dnsdist default)

2.1.3 Instance Sets

pools (Configuration of pools)

Key: name (Name of the pool)

- **ecs**: Set to true if dnsdist should add EDNS Client Subnet information to the query before looking up into the cache, when all servers from this pool are down. If at least one server is up, the preference of the selected server is used, this parameter is only useful if all the backends in this pool are down and have EDNS Client Subnet enabled, since the queries in the cache will have been inserted with ECS information. Default is false. (Dnsdist default)

- **serverGroups**: List of [Name] of recursor & resolver instances whose endpoints will be member of this pool (Dnsdist default)

- **serverPolicy**: Policy for dnsdist to use to select the server in this pool to send a query to. Supported policies are 'leastOutstanding', 'firstAvailable', 'wrandom', 'whashed', 'chashed' & 'roundrobin' (Dnsdist default)

- **packetcache** (Configurable cache that holds responses from prior requests served by the pool)

  - **cookieHashing**: Whether EDNS Cookie values will be hashed, resulting in separate entries for different cookies in the packet cache. This is required if the backend is sending answers with EDNS Cookies, otherwise a client might receive an answer with the wrong cookie. (Dnsdist default)

  - **deferrableInsertLock**: Whether the cache should give up insertion if the lock is held by another thread, or simply wait to get the lock. (Dnsdist default)

  - **keepStaleData**: Whether to suspend the removal of expired entries from the cache when there is no backend available in at least one of the pools using this cache. (Dnsdist default)

- **maxEntries**: Max (Dnsdist default)

- **maxNegativeTTL**: Cache a NXDomain or NoData answer from the backend for at most this amount of seconds, even if the TTL of the SOA record is higher. (Dnsdist default)

- **maxTTL**: Cap the TTL for records to his number. (Dnsdist default)

- **minTTL**: Do not cache entries with a TTL lower than this. (Dnsdist default)

- **numberOfShards**: Number of shards to divide the cache into, to reduce lock contention. (Dnsdist default)

- **parseECS**: Whether any EDNS Client Subnet option present in the query should be extracted and stored to be able to detect hash collisions involving queries with the same qname, qtype and qclass but a different incoming ECS value. (Dnsdist default)

- **staleTTL**: When the backend servers are not reachable, and global configuration setStaleCacheEntriesTTL is set appropriately, TTL that will be used when a stale cache entry is returned. (Dnsdist default)

- **temporaryFailureTTL**: On a SERVFAIL or REFUSED from the backend, cache for this amount of seconds. (Dnsdist default)

2.2 recursors

Configuration of recursor instances

Key: name (Name of the recursor instance)

2.2.1 Parameters

- **replicas**: Default number of replicas in a recursor deployment (default: *2*)

- **revisionHistoryLimit**: Default 'revisionHistoryLimit' for recursor deployments (default: *0*)

2.2.2 Parameter Sets

- **annotations** (Settings to be applied to each recursor instance when added to dnsdist as a server)

  - **addXPF**: Add the client's IP address and port to the query, along with the original destination address and port. Default is disabled (0)

  - **checkClass**: Number to use as QCLASS in the health-check query, default is DNSClass.IN

  - **checkInterval**: The time in seconds between health checks

  - **checkName**: String to use as QNAME in the health-check query, default is "a.root-servers.net."

  - **checkTimeout**: The timeout (in milliseconds) of a health-check query, default to 1000 (1s)

- **checkType**: String to use as QTYPE in the health-check query, default is "A"

- **disableZeroScope**: Disable the EDNS Client Subnet 'zero scope' feature, which does a cache lookup for an answer valid for all subnets (ECS scope of 0) before adding ECS information to the query and doing the regular lookup. This requires the *parseECS* option of the corresponding cache to be set to true

- **maxCheckFailures**: Allow this amount of check failures before declaring the backend down, default is 1

- **mustResolve**: Set to true when the health check MUST return a RCODE different from NXDomain, ServFail and Refused. Default is false, meaning that every RCODE except ServFail is considered valid

- **order**: The order of servers in this set, used by the *leastOutstanding* and *firstAvailable* policies

- **qps**: Limit the number of queries per second to this amount, when using the *firstAvailable* policy

- **reconnectOnUp**: Close and reopen the sockets when a server transits from Down to Up. This helps when an interface is missing when dnsdist is started. Default is false

- **retries**: The number of TCP connection attempts to servers in this set, for a given query

- **rise**: Require NUM consecutive successful checks before declaring the backend up, default is 1

- **setCD**: Set the CD (Checking Disabled) flag in the health-check query, default is false

- **sockets**: Number of sockets (and thus source ports) used toward the backend server, defaults to 1

- **tcpConnectTimeout**: The timeout (in seconds) of a TCP connection attempt

- **tcpFastOpen**: Whether to enable TCP Fast Open

- **tcpRecvTimeout**: The timeout (in seconds) of a TCP read attempt

- **tcpSendTimeout**: The timeout (in seconds) of a TCP write attempt

- **useClientSubnet**: Add the client's IP address in the EDNS Client Subnet option when forwarding the query to this backend

- **useProxyProtocol**: Add a proxy protocol header to the query, passing along the client's IP address and port along with the original destination address and port. Default is false

- **weight**: The weight of servers in this set, used by the *wrandom*, *whashed* and *chashed* policies, default is 1

- **config** (Recursor configuration, any configuration item listed in the recursor documentation (https://doc.powerdns.com/recursor/settings.html) can be referenced here. Settings which are explicitly ignored due to conflicts with Cloud Control are filtered by 'webserver*', 'local*', 'disable-syslog', 'daemon', 'cpu-map', 'chroot', 'socket*' and 'config*')

- **resources** (recursor resource allocation defaults)

  - **limits** (Limit amounts)

   * **cpu**: Limit amount of CPU (Kubernetes default)

   * **memory**: Limit amount of memory (Kubernetes default)

 – **requests** (Request amounts)

   * **cpu**: Request amount of CPU (Kubernetes default)

   * **memory**: Request amount of memory (Kubernetes default)

2.3 resolvers

Configuration of external resolvers

Key: name (Name of the resolver instance)

2.3.1 Parameters

- **ips**: List of IP addresses of resolver endpoints

- **port**: Port to send traffic to for resolver endpoints

2.3.2 Parameter Sets

- **annotations** (Settings to be applied to each resolver instance when added to dnsdist as a server)

 – **addXPF**: Add the client's IP address and port to the query, along with the original destination address and port. Default is disabled (0)

 – **checkClass**: Number to use as QCLASS in the health-check query, default is DNSClass.IN

 – **checkInterval**: The time in seconds between health checks

 – **checkName**: String to use as QNAME in the health-check query, default is "a.root-servers.net."

 – **checkTimeout**: The timeout (in milliseconds) of a health-check query, default to 1000 (1s)

 – **checkType**: String to use as QTYPE in the health-check query, default is "A"

 – **disableZeroScope**: Disable the EDNS Client Subnet 'zero scope' feature, which does a cache lookup for an answer valid for all subnets (ECS scope of 0) before adding ECS information to the query and doing the regular lookup. This requires the *parseECS* option of the corresponding cache to be set to true

 – **maxCheckFailures**: Allow this amount of check failures before declaring the back-end down, default is 1

 – **mustResolve**: Set to true when the health check MUST return a RCODE different from NXDomain, ServFail and Refused. Default is false, meaning that every RCODE except ServFail is considered valid

 – **order**: The order of servers in this set, used by the *leastOutstanding* and *firstAvailable* policies

- **qps**: Limit the number of queries per second to this amount, when using the *firstAvailable* policy

- **reconnectOnUp**: Close and reopen the sockets when a server transits from Down to Up. This helps when an interface is missing when dnsdist is started. Default is false

- **retries**: The number of TCP connection attempts to servers in this set, for a given query

- **rise**: Require NUM consecutive successful checks before declaring the backend up, default is 1

- **setCD**: Set the CD (Checking Disabled) flag in the health-check query, default is false

- **sockets**: Number of sockets (and thus source ports) used toward the backend server, defaults to 1

- **tcpConnectTimeout**: The timeout (in seconds) of a TCP connection attempt

- **tcpFastOpen**: Whether to enable TCP Fast Open

- **tcpRecvTimeout**: The timeout (in seconds) of a TCP read attempt

- **tcpSendTimeout**: The timeout (in seconds) of a TCP write attempt

- **useClientSubnet**: Add the client's IP address in the EDNS Client Subnet option when forwarding the query to this backend

- **useProxyProtocol**: Add a proxy protocol header to the query, passing along the client's IP address and port along with the original destination address and port. Default is false

- **weight**: The weight of servers in this set, used by the *wrandom*, *whashed* and *chashed* policies, default is 1

2.4 rulesets

Configuration of custom rules

Key: name (Name of the rule set)