IPvs-sysctl

/proc/sys/net/ipv4/vs/* Variables:

am_droprate - INTEGER

default 10

It sets the always mode drop rate, which is used in the mode 3 of the drop_rate defense.

amemthresh - INTEGER

default 1024

It sets the available memory threshold (in pages), which is used in the automatic modes of defense. When there is no enough available memory, the respective strategy will be enabled and the variable is automatically set to 2, otherwise the strategy is disabled and the variable is set to 1.

backup_only - BOOLEAN

- 0 - disabled (default)
- not 0 - enabled

If set, disable the director function while the server is in backup mode to avoid packet loops for DR/TUN methods.

conn_reuse_mode - INTEGER

1 - default

Controls how ipvs will deal with connections that are detected port reuse. It is a bitmap, with the values being:

0: disable any special handling on port reuse. The new connection will be delivered to the same real server that was servicing the previous connection.

bit 1: enable rescheduling of new connections when it is safe. That is, whenever expire_nodest_conn and for TCP sockets, when the connection is in TIME_WAIT state (which is only possible if you use NAT mode).

bit 2: it is bit 1 plus, for TCP connections, when connections are in FIN_WAIT state, as this is the last state seen by load balancer in Direct Routing mode. This bit helps on adding new real servers to a very busy cluster.

conntrack - BOOLEAN

- 0 - disabled (default)
- not 0 - enabled

If set, maintain connection tracking entries for connections handled by IPVS.

This should be enabled if connections handled by IPVS are to be also handled by stateful firewall rules. That is, iptables rules that make use of connection tracking. It is a performance optimisation to disable this setting otherwise.

Connections handled by the IPVS FTP application module will have connection tracking entries regardless of this setting.

Only available when IPVS is compiled with CONFIG_IP_VS_NFCT enabled.

cache_bypass - BOOLEAN

- 0 - disabled (default)
- not 0 - enabled

If it is enabled, forward packets to the original destination directly when no cache server is available and destination address is not local (iph->daddr is RTN_UNICAST). It is mostly used in transparent web cache cluster.

debug_level - INTEGER

- 0 - transmission error messages (default)
- 1 - non-fatal error messages
- 2 - configuration
- 3 - destination trash
- 4 - drop entry
- 5 - service lookup
- 6 - scheduling
- 7 - connection new/expire, lookup and synchronization
- 8 - state transition
- 9 - binding destination, template checks and applications
- 10 - IPVS packet transmission
- 11 - IPVS packet handling (ip_vs_in/ip_vs_out)
- 12 or more - packet traversal

Only available when IPVS is compiled with CONFIG_IP_VS_DEBUG enabled.

Higher debugging levels include the messages for lower debugging levels, so setting debug level 2, includes level 0, 1 and 2 messages. Thus, logging becomes more and more verbose the higher the level.

drop_entry - INTEGER

- 0 - disabled (default)

The drop_entry defense is to randomly drop entries in the connection hash table, just in order to collect back some memory for new connections. In the current code, the drop_entry procedure can be activated every second, then it randomly scans 1/32 of the whole and drops entries that are in the SYN-RECV/SYNACK state, which should be effective against syn-flooding attack.

The valid values of drop_entry are from 0 to 3, where 0 means that this strategy is always disabled, 1 and 2 mean automatic modes (when there is no enough available memory, the strategy is enabled and the variable is automatically set to 2, otherwise the strategy is disabled and the variable is set to 1), and 3 means that the strategy is always enabled.

drop_packet - INTEGER

- 0 - disabled (default)

The drop_packet defense is designed to drop 1/rate packets before forwarding them to real servers. If the rate is 1, then drop all the incoming packets.

The value definition is the same as that of the drop_entry. In the automatic mode, the rate is determined by the follow formula: rate = amemthresh / (amemthresh - available_memory) when available memory is less than the available memory threshold. When the mode 3 is set, the always mode drop rate is controlled by the /proc/sys/net/ipv4/vs/am_droprate.

expire_nodest_conn - BOOLEAN

- 0 - disabled (default)
- not 0 - enabled

The default value is 0, the load balancer will silently drop packets when its destination server is not available. It may be useful, when user-space monitoring program deletes the destination server (because of server overload or wrong detection) and add back the server later, and the connections to the server can continue.

If this feature is enabled, the load balancer will expire the connection immediately when a packet arrives and its destination server is not available, then the client program will be notified that the connection is closed. This is equivalent to the feature some people requires to flush connections when its destination is not available.

expire_quiescent_template - BOOLEAN

- 0 - disabled (default)
- not 0 - enabled

When set to a non-zero value, the load balancer will expire persistent templates when the destination server is quiescent. This may be useful, when a user makes a destination server quiescent by setting its weight to 0 and it is desired that subsequent otherwise persistent connections are sent to a different destination server. By default new persistent connections are allowed to quiescent destination servers.

If this feature is enabled, the load balancer will expire the persistence template if it is to be used to schedule a new connection and the destination server is quiescent.

ignore_tunneled - BOOLEAN

- 0 - disabled (default)
- not 0 - enabled

If set, ipvs will set the ipvs_property on all packets which are of unrecognized protocols. This prevents us from routing tunneled protocols like ipip, which is useful to prevent rescheduling packets that have been tunneled to the ipvs host (i.e. to prevent ipvs routing loops when ipvs is also acting as a real server).

nat_icmp_send - BOOLEAN

- 0 - disabled (default)
- not 0 - enabled

It controls sending icmp error messages (ICMP_DEST_UNREACH) for VS/NAT when the load balancer receives packets from real servers but the connection entries don't exist.

pmtu_disc - BOOLEAN

- 0 - disabled
- not 0 - enabled (default)

By default, reject with FRAG_NEEDED all DF packets that exceed the PMTU, irrespective of the forwarding method. For TUN method the flag can be disabled to fragment such packets.

secure_tcp - INTEGER

- 0 - disabled (default)

The secure_tcp defense is to use a more complicated TCP state transition table. For VS/NAT, it also delays entering the TCP ESTABLISHED state until the three way handshake is completed.

The value definition is the same as that of drop_entry and drop_packet.

sync_threshold - vector of 2 INTEGERs: sync_threshold, sync_period

default 3 50

It sets synchronization threshold, which is the minimum number of incoming packets that a connection needs to receive before the connection will be synchronized. A connection will be synchronized, every time the number of its incoming packets modulus sync_period equals the threshold. The range of the threshold is from 0 to sync_period.

When sync_period and sync_refresh_period are 0, send sync only for state changes or only once when pkts matches sync_threshold

sync_refresh_period - UNSIGNED INTEGER

default 0

In seconds, difference in reported connection timer that triggers new sync message. It can be used to avoid sync messages for the specified period (or half of the connection timeout if it is lower) if connection state is not changed since last sync.

This is useful for normal connections with high traffic to reduce sync rate. Additionally, retry sync_retries times with period of sync_refresh_period/8.

sync_retries - INTEGER

default 0

Defines sync retries with period of sync_refresh_period/8. Useful to protect against loss of sync messages. The range of the sync_retries is from 0 to 3.

sync_qlen_max - UNSIGNED LONG

Hard limit for queued sync messages that are not sent yet. It defaults to 1/32 of the memory pages but actually represents number of messages. It will protect us from allocating large parts of memory when the sending rate is lower than the queuing rate.

sync_sock_size - INTEGER

default 0

Configuration of SNDBUF (master) or RCVBUF (slave) socket limit. Default value is 0 (preserve system defaults).

sync_ports - INTEGER

default 1

The number of threads that master and backup servers can use for sync traffic. Every thread will use single UDP port, thread 0 will use the default port 8848 while last thread will use port 8848+sync_ports-1.

snat_reroute - BOOLEAN

- 0 - disabled
- not 0 - enabled (default)

If enabled, recalculate the route of SNATed packets from realservers so that they are routed as if they originate from the director. Otherwise they are routed as if they are forwarded by the director.

If policy routing is in effect then it is possible that the route of a packet originating from a director is routed differently to a packet being forwarded by the director.

If policy routing is not in effect then the recalculated route will always be the same as the original route so it is an optimisation to disable snat_reroute and avoid the recalculation.

sync_persist_mode - INTEGER

default 0

Controls the synchronisation of connections when using persistence

0: All types of connections are synchronised

1: Attempt to reduce the synchronisation traffic depending on the connection type. For persistent services avoid synchronisation for normal connections, do it only for persistence templates. In such case, for TCP and SCTP it may need enabling sloppy_tcp and sloppy_sctp flags on backup servers. For non-persistent services such optimization is not applied, mode 0 is assumed.

sync_version - INTEGER

default 1

The version of the synchronisation protocol used when sending synchronisation messages.

0 selects the original synchronisation protocol (version 0). This should be used when sending synchronisation messages to a legacy system that only understands the original synchronisation protocol.

1 selects the current synchronisation protocol (version 1). This should be used where possible.

Kernels with this sync_version entry are able to receive messages of both version 1 and version 2 of the synchronisation protocol.

run_estimation - BOOLEAN

0 - disabled not 0 - enabled (default)

If disabled, the estimation will be stop, and you can't see any update on speed estimation data.

You can always re-enable estimation by setting this value to 1. But be careful, the first estimation after re-enable is not accurate.