|  |  |
| --- | --- |
| Title | Combining Remote Link Monitoring with FGCP cluster High Availability |
| External [ X ] Internal Only [ ] | |
| Applicable Product(s), Firmware(s): | FortiOS 5.4 and 5.6 |
| Question / Description of problem | This article is aimed at providing some technical tips for using and configuring Remote Link Monitoring in combination with a FGCP HA cluster.  Like Link Monitoring can be combined with a FGCP cluster in order to detect a local link failure and thus ensure the high availability of a cluster, Remote Link Monitoring can be used to detect a remote failure, either on a remote link or remote equipment, and potentially trigger a cluster failover to avoid a traffic interruption. |
| Solution | Essentially, implementing a Remote Link Monitor comes firstly to select one or more network servers or services located downstream of the cluster, then configure the cluster to periodically monitor access to those network resources, and trigger a cluster failover in case the primary unit would lose access to those resources.  There is no real limitation on how Remote Link Monitoring can be applied. It can either be implemented in conjunction with Port Monitoring or in standalone mode. It can be composed of one or several remote devices accessible from one or several ports of the FGCP HA cluster.   1. **Configuring and adding Remote Link Monitoring in a HA cluster configuration:**   Configuring a Remote Link Monitor basically requires the following information to be specified:   1. The IP address of the remote device(s) which need to be monitored 2. The FortiGate unit interface that is used for the remote link monitoring 3. The protocol used to enforce the remote link monitoring 4. The ‘nominal’ PENALTY to be applied would a heath check failure occur 5. The time interval in between two health check attempts 6. The number of tolerable health check failures before a failover is triggered   Example of a Remote Link Monitoring configuration as defined above:  # show system link-monitor  config system link-monitor  edit ha-link-monitor  set server 10.10.10.10 <<< 1  set srcintf port1 <<< 2  set protocol ping <<< 3 (ping is the default option setting)  set ha-priority 1 <<< 4 (1 is the default value)  set interval 5 <<< 5 (5 is the default value)  set failtime 2 <<< 6 (5 is the default value)  end  Once a Remote Link Monitor (also referred as Remote IP Monitoring or PING server Monitoring in the documentation) has been defined, it can then be integrated in the cluster HA configuration and the following information specified:   1. The FortiGate unit interface(s) that are being used for the remote link monitoring 2. The failover threshold value (value against which the ‘global’ PENALTY value is compared) 3. The type of failover mechanism (automated fail back or not upon expiration of the FLIP timer) 4. The HA remote link monitoring FLIP timeout value   Example of a HA cluster configuration combined with PING server monitoring:  # show system ha  config system ha  …  set pingserver-monitor-interface port1 <<< 1  set pingserver-failover-threshold 0 <<< 2 (0 is the default value)  set pingserver-slave-force-reset disable <<< 3 (option enabled by default)  set pingserver-flip-timeout 60 <<< 4 (by default set to 60 minutes)  …  end  Note: having the “pingserver-failover-threshold” variable set to ‘0’ is a mean to trigger a HA failover right after the remote link failure is detected.   1. **Remote Link Monitoring considerations:**   It should be noted that, alike port monitoring, remote link monitoring is just a mechanism to activate the FGCP HA election process. The decision to trigger a failover or not is ultimately taken by the HA process itself, based on the HA parameters value such as the “override” parameter being enabled or not, the HA priority value set on each cluster units, and so on.  The scenario detailed below is based on the assumption that the HA “override” parameter is enabled and the cluster “preferred” master is set with a higher “priority” value than the slave. This type of setting is typically used when there is a need to have one of the cluster units acting, as far as possible, as a “preferred” master unit.  Using Remote Link Monitor in conjunction with FGCP cluster High Availability:  Each time a remote link monitoring failure is detected by the HA cluster master unit, the ‘global’ PENALTY that is by default set to ‘0’ is incremented by the ‘nominal’ PENALTY value (the “ha-priority” parameter value) and compared to the failover threshold value (the “pingserver-failover-threshold” variable value ). When the threshold value is reached, the ‘global’ PENALTY value of the master is compared with the one of the slave and, if it is higher, the FLIP timer is started and a failover occurs. The new master starts monitoring the remote link on its own and will handle any remote link monitoring failure as described above i.e. the ‘global’ PENALTY will be incremented by its ‘nominal’ PENALTY value, up to the point the failover threshold value is reached. The action taken at the time the FLIP timer elapses will then depend upon the settings of the “pingserver-slave-force-reset” variable value.   * “pingserver-slave-force-reset” variable is set to “enable” (default setting)   When the FLIP timer elapses, the “preferred” master ‘global’ PENALTY is reset. Regardless of the remote link monitoring status on the new master, the cluster automatically returns to normal operation i.e. a failover occurs since the HA “override” parameter is enabled and the HA “priority” of the “preferred” master is higher than the one of the new master. The FLIP timer is started and the “preferred” master unit starts remote link monitoring again. If the remote link is restored the cluster continues to operate normally. If, however, the remote link is still down, remote link failover causes the cluster to failover again at the time the FLIP timer expires. This sequence, known as FLIP-FLOP failover, will repeat each time the FLIP timer expires, up until the failed remote link is restored.   * “pingserver-slave-force-reset” variable is set to “disable”   With this setting, the “preferred” master ‘global’ PENALTY is not reset when the FLIP timer elapses. This way, there will be no FLIP-FLOP failover if the new master does not detect any remote link failover failure, the drawback being that the “preferred” master will never get a chance to become master again, even if the remote link is restored on its side. Only a manual failover (likely after restoring the ping server failure) or a remote link failure on the new master side can trigger a failover. Indeed, in case the new master also experiences a remote link failure, its ‘global’ PENALTY will be increased and become equal to the one of the “preferred” master, thus causing the HA election process to start. In this case, the “preferred” master will take the cluster ownership back since the HA “override” parameter is enabled and the HA “priority” of the “preferred” master is higher than the one of the new master.   1. **Troubleshooting the Remote Link Monitor process:**   Verifying and controlling the Link Monitor can be done using the following commands set:  # diag debug application link-monitor -1  # diag debug console timestamp enable  # diag debug enable  It has to be noted that, in a FGCP HA cluster, only the master unit can perform remote link monitoring. By design, a slave unit cannot perform any monitoring since it has no active routing table. This can be verified from the following command excerpt which was recorded on a FGCP HA cluster configured with link monitor and HA settings defined previously. In the example below, FGT1 is configured as “preferred” master. It is master at the beginning of the test and becomes slave after it loses connectivity with the remote ping server.   * Snapshot of a link monitor test performed on FGT1 (master unit)   1. 08:37:14: link monitor PING test towards the remote server (10.219.5.237) is done. It is successful   2. 08:37:19: 5 seconds later (cf. ‘interval’ variable setting) another PING test is done. It is successful   3. 08:37:24: 5 seconds later another successful PING test is done. It is successful   4. 🡪 a loss of connectivity in between FGT1 and the remote ping server is simulated   5. 08:37:29: 5 seconds later another PING test is done but fails. It is done a second time (cf. ‘failtime’ variable setting) and also fails.   6. 08:37:31: link monitor is flagged as non-operational (cf. ‘ha-link-monitor is dead’ message)   7. 08:37:33: routing table is deactivated on FGT1 - failover occurs (FGT2 becomes master)   8. 08:37:37: the PING test cycle is re-initiated but no packets are effectively issued since routing table is inactive   9. 08:37:42: idem than step 8   10. 08:37:47: idem than step 8   FGT1 # diag debug application link-monitor -1FGT1 # diag debug console timestamp enableFGT1 # diag debug enable  2019-07-03 08:37:14 lnkmtd::ping\_send\_msg(256): --> ping 10.219.5.237 seq\_no=24344, icmp id=0, send 40 bytes  2019-07-03 08:37:14 lnkmtd:: ha-link-monitor send check request, try 1  2019-07-03 08:37:14 lnkmtd::ping\_match(71): try matching ping response 10.219.5.237  2019-07-03 08:37:14 lnkmtd::ping\_do\_addr\_up(57): ha-link-monitor->10.219.5.237(10.219.5.237), rcvd  2019-07-03 08:37:14 monitor\_peer\_recv-1790: lnkmtd: ha-link-monitor send time 1562135834s 205177us, revd time 1562135834s 206031us  2019-07-03 08:37:14 lnkmtd: ha-link-monitor all servers are probed after 1 times  2019-07-03 08:37:14 policy route related to the monitor(ha-link-monitor) may be added  2019-07-03 08:37:14 lnkmt\_ha\_mstate\_build-182: monitor=ha-link-monitor, state=0,send sz=78  2019-07-03 08:37:14 rcvd cmd = 0  2019-07-03 08:37:19 lnkmtd::ping\_send\_msg(256): --> ping 10.219.5.237 seq\_no=24345, icmp id=0, send 40 bytes  2019-07-03 08:37:19 lnkmtd:: ha-link-monitor send check request, try 1  2019-07-03 08:37:19 lnkmtd::ping\_match(71): try matching ping response 10.219.5.237  2019-07-03 08:37:19 lnkmtd::ping\_do\_addr\_up(57): ha-link-monitor->10.219.5.237(10.219.5.237), rcvd  2019-07-03 08:37:19 monitor\_peer\_recv-1790: lnkmtd: ha-link-monitor send time 1562135839s 205390us, revd time 1562135839s 206305us  2019-07-03 08:37:19 lnkmtd: ha-link-monitor all servers are probed after 1 times  2019-07-03 08:37:19 policy route related to the monitor(ha-link-monitor) may be added  2019-07-03 08:37:19 lnkmt\_ha\_mstate\_build-182: monitor=ha-link-monitor, state=0,send sz=78  2019-07-03 08:37:19 rcvd cmd = 0  2019-07-03 08:37:24 lnkmtd::ping\_send\_msg(256): --> ping 10.219.5.237 seq\_no=24346, icmp id=0, send 40 bytes  2019-07-03 08:37:24 lnkmtd:: ha-link-monitor send check request, try 1  2019-07-03 08:37:24 lnkmtd::ping\_match(71): try matching ping response 10.219.5.237  2019-07-03 08:37:24 lnkmtd::ping\_do\_addr\_up(57): ha-link-monitor->10.219.5.237(10.219.5.237), rcvd  2019-07-03 08:37:24 monitor\_peer\_recv-1790: lnkmtd: ha-link-monitor send time 1562135844s 205655us, revd time 1562135844s 206595us  2019-07-03 08:37:24 lnkmtd: ha-link-monitor all servers are probed after 1 times  2019-07-03 08:37:24 policy route related to the monitor(ha-link-monitor) may be added  2019-07-03 08:37:24 lnkmt\_ha\_mstate\_build-182: monitor=ha-link-monitor, state=0,send sz=78  2019-07-03 08:37:24 rcvd cmd = 0  2019-07-03 08:37:29 lnkmtd::ping\_send\_msg(256): --> ping 10.219.5.237 seq\_no=24347, icmp id=0, send 40 bytes  2019-07-03 08:37:29 lnkmtd:: ha-link-monitor send check request, try 1  2019-07-03 08:37:30 lnkmtd::ping\_send\_msg(256): --> ping 10.219.5.237 seq\_no=24348, icmp id=0, send 40 bytes  2019-07-03 08:37:30 lnkmtd:: ha-link-monitor send check request, try 2  2019-07-03 08:37:30 lnkmtd: ha-link-monitor have tried 2 times, and will restart after 3 seconds  2019-07-03 08:37:31 lnkmtd: ha-link-monitor is dead.  2019-07-03 08:37:31 policy route related to the monitor(ha-link-monitor) may be removed  2019-07-03 08:37:31 lnkmt\_ha\_mstate\_build-182: monitor=ha-link-monitor, state=1,send sz=78  2019-07-03 08:37:31 rcvd cmd = 0  2019-07-03 08:37:33 lnkmt\_proute\_refresh-582  2019-07-03 08:37:37 rcvd cmd = 0  2019-07-03 08:37:42 rcvd cmd = 0  2019-07-03 08:37:47 rcvd cmd = 0   * Snapshot of a link monitor test performed on FGT2 (slave unit)   1. 08:37:14: PING test cycle triggers but no packets are effectively issued since routing table is inactive 08:37:19: PING test cycle triggers but no packets are effectively issued since routing table is inactive   2. 08:37:24: PING test cycle triggers but no packets are effectively issued since routing table is inactive   3. 08:37:29: PING test cycle triggers but no packets are effectively issued since routing table is inactive   4. 08:37:33: failover occurs (FGT2 becomes master) - routing table is activated and all interfaces are brought UP   5. 08:37:37: link monitor PING test towards the remote server (10.219.5.237) is done. It is successful   6. 08:37:42: 5 seconds later (cf. ‘interval’ variable setting) another PING test is done. It is successful   7. 08:37:47: 5 seconds later another successful PING test is done. It is successful   FGT2 # diag debug application link-monitor -1FGT2 # diag debug console timestamp enableFGT2 # diag debug enable  2019-07-03 08:37:14 rcvd cmd = 0  2019-07-03 08:37:19 rcvd cmd = 0  2019-07-03 08:37:24 rcvd cmd = 0  2019-07-03 08:37:29 rcvd cmd = 0  2019-07-03 08:37:33 lnkmt\_proute\_refresh-582  2019-07-03 08:37:33 bring up 'mgmt2'  2019-07-03 08:37:33 bring up 'mgmt2' since all associated intfs are okay  2019-07-03 08:37:33 bring up 'port1'  2019-07-03 08:37:33 bring up 'port1' since all associated intfs are okay  …  2019-07-03 08:37:33 bring up 'wan1'  2019-07-03 08:37:33 bring up 'wan1' since all associated intfs are okay  2019-07-03 08:37:33 bring up 'wan2'  2019-07-03 08:37:33 bring up 'wan2' since all associated intfs are okay  2019-07-03 08:37:37 lnkmtd::ping\_send\_msg(256): --> ping 10.219.5.237 seq\_no=2, icmp id=0, send 40 bytes  2019-07-03 08:37:37 lnkmtd:: ha-link-monitor send check request, try 1  2019-07-03 08:37:37 lnkmtd::ping\_match(71): try matching ping response 10.219.5.237  2019-07-03 08:37:37 lnkmtd::ping\_do\_addr\_up(57): ha-link-monitor->10.219.5.237(10.219.5.237), rcvd  2019-07-03 08:37:37 monitor\_peer\_recv-1790: lnkmtd: ha-link-monitor send time 1562135857s 305897us, revd time 1562135857s 306586us  2019-07-03 08:37:37 lnkmtd: ha-link-monitor all servers are probed after 1 times  2019-07-03 08:37:37 policy route related to the monitor(ha-link-monitor) may be added  2019-07-03 08:37:37 lnkmt\_ha\_mstate\_build-182: monitor=ha-link-monitor, state=0,send sz=78  2019-07-03 08:37:37 rcvd cmd = 0  2019-07-03 08:37:42 lnkmtd::ping\_send\_msg(256): --> ping 10.219.5.237 seq\_no=3, icmp id=0, send 40 bytes  2019-07-03 08:37:42 lnkmtd:: ha-link-monitor send check request, try 1  2019-07-03 08:37:42 lnkmtd::ping\_match(71): try matching ping response 10.219.5.237  2019-07-03 08:37:42 lnkmtd::ping\_do\_addr\_up(57): ha-link-monitor->10.219.5.237(10.219.5.237), rcvd  2019-07-03 08:37:42 monitor\_peer\_recv-1790: lnkmtd: ha-link-monitor send time 1562135862s 305933us, revd time 1562135862s 306853us  2019-07-03 08:37:42 lnkmtd: ha-link-monitor all servers are probed after 1 times  2019-07-03 08:37:42 policy route related to the monitor(ha-link-monitor) may be added  2019-07-03 08:37:42 lnkmt\_ha\_mstate\_build-182: monitor=ha-link-monitor, state=0,send sz=78  2019-07-03 08:37:42 rcvd cmd = 0  2019-07-03 08:37:47 lnkmtd::ping\_send\_msg(256): --> ping 10.219.5.237 seq\_no=4, icmp id=0, send 40 bytes  2019-07-03 08:37:47 lnkmtd:: ha-link-monitor send check request, try 1  2019-07-03 08:37:47 lnkmtd::ping\_match(71): try matching ping response 10.219.5.237  2019-07-03 08:37:47 lnkmtd::ping\_do\_addr\_up(57): ha-link-monitor->10.219.5.237(10.219.5.237), rcvd  2019-07-03 08:37:47 monitor\_peer\_recv-1790: lnkmtd: ha-link-monitor send time 1562135867s 305968us, revd time 1562135867s 306875us  2019-07-03 08:37:47 lnkmtd: ha-link-monitor all servers are probed after 1 times  2019-07-03 08:37:47 policy route related to the monitor(ha-link-monitor) may be added  2019-07-03 08:37:47 lnkmt\_ha\_mstate\_build-182: monitor=ha-link-monitor, state=0,send sz=78  2019-07-03 08:37:47 rcvd cmd = 0   1. **Verifying the accessibility of remote devices used by the Link Monitoring process**   Remote devices accessibility cannot be assessed from the slave unit using a PING test. Indeed, despite a PING test executed from a slave unit may show a remote device to be accessible from this unit, it is potentially not. Indeed, the ICMP echo requests are in reality not issued directly by the slave unit but are handed over to the master unit via the HA link. The master unit then forward the requests to the remote device and relay the ICMP echo replies it receives back to the slave unit via the HA link. This can be verified in the example below wherein we can see that a PING test executed from FGT1 (slave unit) is effectively processed by FGT2 (master unit).   * Snapshot of a “successful” PING test performed from the slave unit   FGT1 # execute ping 10.219.5.237  PING 10.219.5.237 (10.219.5.237): 56 data bytes  64 bytes from 10.219.5.237: icmp\_seq=0 ttl=125 time=1.2 ms  64 bytes from 10.219.5.237: icmp\_seq=1 ttl=125 time=1.0 ms  …   * Snapshot of a packet capture running on the master unit and highlighting that ICMP echo requests are passed to the master unit via the HA heartbeat interface then sent to the remote device via port3. ICMP echo replies are received from port3 and forwarded back to the slave unit via the HA heartbeat interface.     FGT2 # diag sniffer packet any 'icmp' 4 0 a  interfaces=[any]  filters=[icmp]  2019-07-03 07:17:07.133523 port\_ha in 169.254.0.2 -> 10.219.5.237: icmp: echo request  2019-07-03 07:17:07.133555 havdlink0 out 169.254.0.65 -> 10.219.5.237: icmp: echo request  2019-07-03 07:17:07.133555 havdlink1 in 169.254.0.65 -> 10.219.5.237: icmp: echo request  2019-07-03 07:17:07.133575 port3 out 10.217.2.30 -> 10.219.5.237: icmp: echo request  2019-07-03 07:17:07.134260 port3 in 10.219.5.237 -> 10.217.2.30: icmp: echo reply  2019-07-03 07:17:07.134270 havdlink1 out 10.219.5.237 -> 169.254.0.65: icmp: echo reply  2019-07-03 07:17:07.134270 havdlink0 in 10.219.5.237 -> 169.254.0.65: icmp: echo reply  2019-07-03 07:17:07.134278 port\_ha out 10.219.5.237 -> 169.254.0.2: icmp: echo reply  2019-07-03 07:17:07.134281 port5 out 10.219.5.237 -> 169.254.0.2: icmp: echo reply  2019-07-03 07:17:08.133626 port\_ha in 169.254.0.2 -> 10.219.5.237: icmp: echo request  2019-07-03 07:17:08.133630 havdlink0 out 169.254.0.65 -> 10.219.5.237: icmp: echo request  2019-07-03 07:17:08.133630 havdlink1 in 169.254.0.65 -> 10.219.5.237: icmp: echo request  2019-07-03 07:17:08.133640 port3 out 10.217.2.30 -> 10.219.5.237: icmp: echo request  2019-07-03 07:17:08.134363 port3 in 10.219.5.237 -> 10.217.2.30: icmp: echo reply  2019-07-03 07:17:08.134370 havdlink1 out 10.219.5.237 -> 169.254.0.65: icmp: echo reply  2019-07-03 07:17:08.134370 havdlink0 in 10.219.5.237 -> 169.254.0.65: icmp: echo reply  2019-07-03 07:17:08.134374 port\_ha out 10.219.5.237 -> 169.254.0.2: icmp: echo reply  2019-07-03 07:17:08.134375 port5 out 10.219.5.237 -> 169.254.0.2: icmp: echo reply  … |
| Internal Notes (optional) |  |
| Related KB articles/BUGs (optional) |  |

Return to [kb@fortinet.com](mailto:kb@fortinet.com)