

HA (High Availability)

Troubleshooting Guide

[Introduction](#)

[Requirements](#)

[Components Used](#)

[Case Study](#)

[Problems](#)

[Debug tips and Possible fixes](#)

[Check HA State](#)

[Link Monitor Feature in the HA](#)

Introduction

This document describes common debug commands used to troubleshoot HA issues on the DFL firewall. This document does not provide detailed explanation of HA and associated concepts. We assume the readers is familiar with HA configuration. These troubleshooting issues and solutions are based on the service requests that D-Link Technical Support team have resolved.

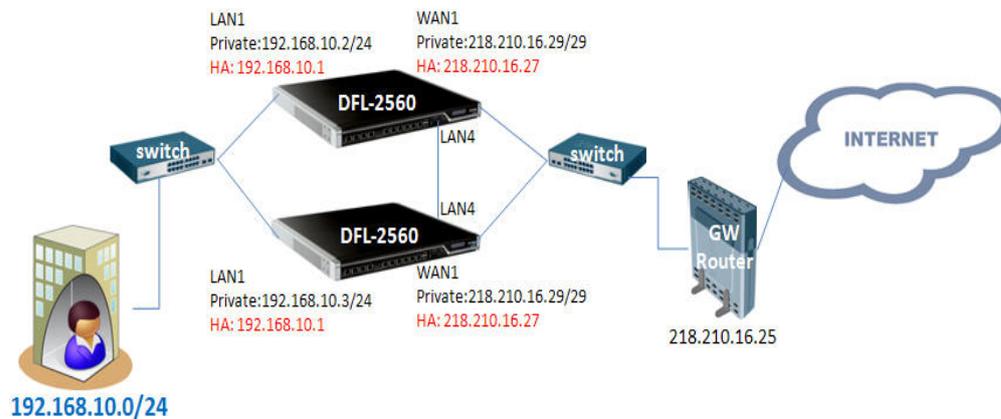
Please refer to HA related issues. It includes a list of common troubleshooting procedures that you might try before making a call to the D-Link HQ Tech Support.

Warning: Some of debug procedures mentioned in this document can lead to a temporary problem of HA function. D-Link recommends you to implements these solutions with caution and make sure the network administrator is well-informed before you start these troubleshooting procedures.

Requirements

- HA description and setup examples from the DFL manual.
- Basic DFL HA configuration.

Case Study



Forever.Inc is a global corporation which is dedicated to Ad. consultant service. Recently, they implemented DFL HA feature with hardware redundancy while improving network availability.

Problems

The HA redundancy is not working as expected, some problems happen in the user network.

Debug tips and Possible fixes

To start with Solutions, some tips about HA which you should be familiar with are listed below.

1. All interfaces should have specific HA address, including the idle interface.
2. About the heart bit packet, it would be better that send heart bit packets via interfaces also (lan, wan interfaces..etc).

Check HA state

When HA malfunctions, we can check HA state via CLI.

 Check **Master** HA state via CLI:

```
DFL-2560:/> ha
This device is a HA MASTER
This device is currently ACTIVE (will forward traffic)
This device has been active: 191 sec
HA cluster peer is DEAD
```

Check **Slave** HA state via CLI:

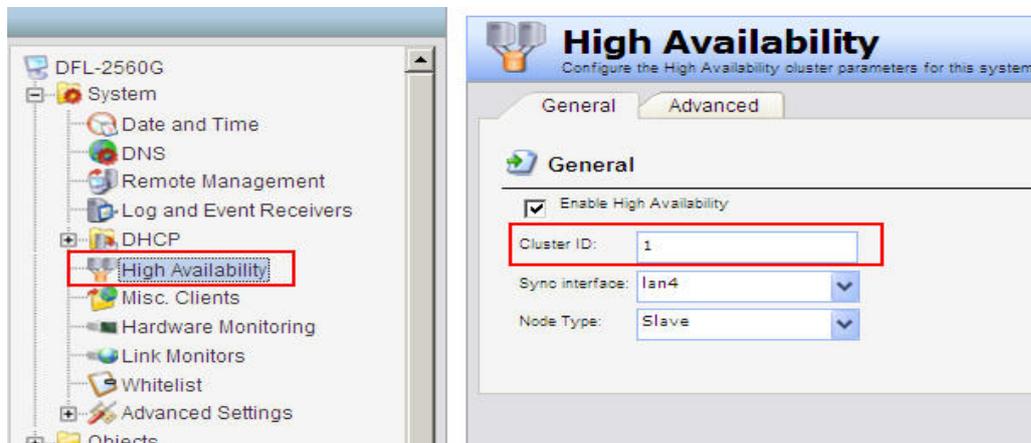
```
DFL-2560G:/> ha
This device is a HA SLAVE
This device is currently ACTIVE (will forward traffic)
```

This device has been active: 246 sec
HA cluster peer is **DEAD**

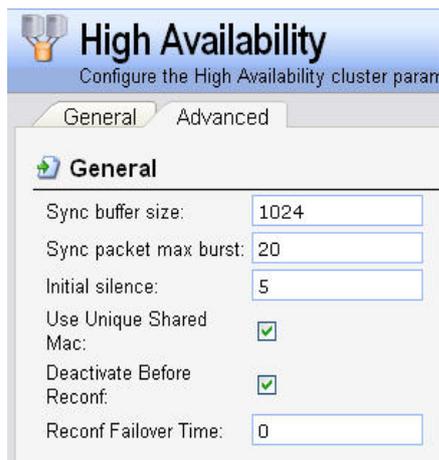
In above example we can see that both Master and Slave are in the "ACTIVE" state and think peer is dead. In such situation, the user network is being very instability and suffering.

The following is some HA debug procedures.

1. Make sure Sync interfaces are connected between each other or corresponding activate interfaces are connected in the same broadcast domain. We need make sure the heartbeat packets can communicated between each other.
2. Check HA cluster number, both firewalls should have the same Cluster ID.



3. Make sure Master and Slave roles.
4. Make sure both firewalls have the same HA advanced parameters.



5. All shared IP addresses must be the same between each other.

[<Top>](#)

Link Monitor Feature in the HA

HA mechanism monitors heartbeat packet to check the status of HA peer device, the firewall thinks that the peer is dead and takes the HA Active ownership if no heartbeat packet received.

However, there might be a problem if only specific interface connection is fail. For example, if we set up both the LAN, WAN and DMZ(sync port) interfaces to sending & receiving heartbeats, when the WAN interface is down, the HA failover will not be triggered. That is because firewalls still can receive heartbeat from other interfaces.

Now we can use Link Monitor feature to avoid this limitation. With Link Monitor feature, DFL monitors link status per interface, and triggers HA failover if monitored interface is down.

The following picture is an example of setting up the Link Monitor function. In this example, we set up the Action to "HA Failover". In the selected option, we selected LAN_PC and wan1_gw, which means, LAN and WAN1 interfaces are under monitoring.

