将LIBfc与VMware和Cisco VIC结合使用来排除启动器/目标通信故障

目录

简介

本文档介绍如何使用隐藏的libfc调试来低级地了解ESXi中光纤通道(FC)通信中使用的端口登录(PLOGI)过程。 通过启用debug_logging，我们能够看到融合网络适配器(CNA)有关扩展链路服务(ELS)帧的信息，例如交换矩阵登录(FLOGI)、端口登录(PLOGI)，这是我们通常无法看到的。 如果没有Finisar工具或SPAN，并且您想确保主机在FC堆栈中完成/未完成的内容，则此功能非常有用。

作者：思科TAC工程师Brian Hopkins。

支持的配置

目前，仅ESX上支持此功能，而ESX上有思科虚拟接口卡(VIC)，据我所知，其他适配器不支持此功能。

识别当前设置

您可以在ESXi主机上使用以下命令来确保尚未设置此值：

```
From the CLI of ESXi:
esxcli system module parameters list -m libfc_92

esxcli system module parameters list -m libfcoe_92
```
输出应如下所示，注意未为**debug_logging**配置值，这是我们在后续步骤中将更改的值。

```
~ # tail /var/log/vmkernel.log | grep <6>
~ # esxcli system module parameters list -m libfc_92
Name                Type  Value  Description
----------------    ----  -----  ------------------------------------------------------------------
debug_logging       int          a bit mask of logging levels
heap_initial        int          Initial heap size allocated for the driver.
heap_max            int          Maximum attainable heap size for the driver.
min_exch_pool_elem  int          Miminum number of elements guaranteed to be allocated for exchange pool.
rec_tov             int          REC timeout value
skb_mpool_initial   int          Driver's minimum private socket buffer memory pool size.
skb_mpool_max       int          Maximum attainable private socket buffer memory pool size for the driver.
~ # esxcli system module parameters list -m libfcoe_92
Name                Type  Value  Description
----------------    ----  -----  ------------------------------------------------------------------
debug_logging       int          a bit mask of logging levels
heap_initial        int          Initial heap size allocated for the driver.
heap_max            int          Maximum attainable heap size for the driver.
skb_mpool_initial   int          Driver's minimum private socket buffer memory pool size.
skb_mpool_max       int          Maximum attainable private socket buffer memory pool size for the driver.
~ # _
```

更改LIBfc debug_logging设置

要获取更多信息以显示在ESXi的/var/log/vmkernel.log文件中，我们需要启用debug_logging，并且必须重新启动主机：

```
esxcli system module parameters set -p debug_logging=0xf -m libfc_92

esxcli system module parameters set -p debug_logging=0xf -m libfcoe_92
```
输入以下命令后，您可以再次检查以确保值现在设置为0xf:

```
~ # esxcli system module parameters set -p debug_logging=0xf -m libfc_92
~ # esxcli system module parameters set -p debug_logging=0xf -m libfcoe_92
~ # esxcli system module parameters list -m libfcoe_92
Name                Type  Value  Description
----------------    ----  -----  ------------------------------------------------------------------
debug_logging       int   0xf    a bit mask of logging levels
heap_initial        int          Initial heap size allocated for the driver.
heap_max            int          Maximum attainable heap size for the driver.
skb_mpool_initial   int          Driver's minimum private socket buffer memory pool size.
skb_mpool_max       int          Maximum attainable private socket buffer memory pool size for the driver.
~ # esxcli system module parameters list -m libfc_92
Name                Type  Value  Description
----------------    ----  -----  ------------------------------------------------------------------
debug_logging       int   0xf    a bit mask of logging levels
heap_initial        int          Initial heap size allocated for the driver.
heap_max            int          Maximum attainable heap size for the driver.
min_exch_pool_elem  int          Miminum number of elements guaranteed to be allocated for exchange pool.
rec_tov             int          REC timeout value
skb_mpool_initial   int          Driver's minimum private socket buffer memory pool size.
skb_mpool_max       int          Maximum attainable private socket buffer memory pool size for the driver.
```

我们仍未完成，在重新启动ESXi主机之前，您将看不到**新日志显示**。 重新启动ESXi主机后，您可以通过运行以下命令来验证在vmkernel.log文件中是否看到此新更新的数据：

```
cat /var/log/vmkernel.log | grep "<6>"
```
由于所有命令都具有此<6>报头，因此易于查找，因此我在下面包含了此新有用信息的片段，显示FLOGI和PLOGI状态：

```
2016-04-01T16:12:39.672Z cpu21:8803)<6>fnic : 3 :: vNIC flags 0x8 luns per tgt 256
2016-04-01T16:12:39.672Z cpu21:8803)<6>fnic : 3 :: vNIC flogi_retries 8 flogi timeout 4000
2016-04-01T16:12:39.672Z cpu21:8803)<6>fnic : 3 :: vNIC plogi retries 8 plogi timeout 20000
2016-04-01T16:12:39.672Z cpu21:8803)<6>fnic : 3 :: vNIC io throttle count 16 link dn timeout 30000
2016-04-01T16:12:39.672Z cpu21:8803)<6>fnic : 3 :: vNIC port dn io retries 30 port dn timeout 30000
2016-04-01T16:12:39.673Z cpu21:8803)<6>fnic : 3 :: vNIC interrupt mode: MSI-X
2016-04-01T16:12:39.673Z cpu21:8803)<6>fnic : 3 :: vNIC resources avail: wq 2 cp_wq 1 raw_wq 1 rq 1 cq 3 intr 4
2016-04-01T16:12:39.673Z cpu21:8803)<6>fnic : 3 :: firmware uses non-FIP mode
2016-04-01T16:12:39.680Z cpu21:8803)<6>host3: lport ffffffff: Entered RESET state from reset state
             <6>Broadcom NetXtreme II CNIC Driver cnic v1.74.04.v50.1 (September 11, 2012)
             <6>bnx2fc: Broadcom NetXtreme II FCoE Driver bnx2fc v1.74.02.v50.2 (Aug 28, 2012)
2016-04-01T16:12:40.341Z cpu1:8761)<6>host2: libfc: Link up on port (     0)
2016-04-01T16:12:40.341Z cpu1:8761)<6>host2: lport      0: Entered FLOGI state from reset state
2016-04-01T16:12:40.354Z cpu2:8763)<6>host2: lport      0: Received a FLOGI accept
2016-04-01T16:12:40.354Z cpu2:8763)<6>host2: Assigned Port ID 10003
2016-04-01T16:12:40.354Z cpu2:8763)<6>host2: fip: received FLOGI LS_ACC using non-FIP mode
2016-04-01T16:12:40.354Z cpu2:8763)<6>host2: lport  10003: Entered DNS state from FLOGI state
2016-04-01T16:12:40.354Z cpu2:8763)<6>host2: rport fffffc: Login to port
2016-04-01T16:12:40.354Z cpu2:8763)<6>host2: rport fffffc: Port entered PLOGI state from Init state
2016-04-01T16:12:40.356Z cpu18:8733)<6>host2: rport fffffc: Received a PLOGI accept
2016-04-01T16:12:40.357Z cpu18:8733)<6>host2: rport fffffc: Port is Ready
2016-04-01T16:12:40.357Z cpu18:8733)<6>host2: rport fffffc: work event 1
2016-04-01T16:12:40.357Z cpu18:8733)<6>host2: rport fffffc: callback ev 1
2016-04-01T16:12:40.357Z cpu18:8733)<6>host2: lport  10003: Received a 1 event for port (fffffc)
```

将LIBfc debug_logging更改回原始设置：

您可以通过在下面插入2个命令并重新启动ESXi主机将此更改回默认值。 我们基本上只是将更改从以前的更改归零，将此值设置回默认值：

```
esxcli system module parameters set -p debug_logging= -m libfc_92

esxcli system module parameters set -p debug_logging= -m libfcoe_92
```
您可以再次运行相同的命令以确保更改成功：

```
From the CLI of ESXi:
esxcli system module parameters list -m libfc_92

esxcli system module parameters list -m libfcoe_92
```
它们应该如下所示：

```
~ # tail /var/log/vmkernel.log | grep <6>
~ # esxcli system module parameters list -m libfc_92
Name                Type   Value   Description
----------------    ----   -----   --------------------------------------------------------------------
debug_logging       int            a bit mask of logging levels
heap_initial        int            Initial heap size allocated for the driver.
heap_max            int            Maximum attainable heap size for the driver.
min_exch_pool_elem  int            Minimum number of elements guaranteed to be allocated for exchange pool.
rec_tov             int            REC timeout value
skb_mpool_initial   int            Driver's minimum private socket buffer memory pool size.
skb_mpool_max       int            Maximum attainable private socket buffer memory pool size for the driver.
~ # esxcli system module parameters list -m libfcoe_92
Name                Type   Value   Description
----------------    ----   -----   --------------------------------------------------------------------
debug_logging       int            a bit mask of logging levels
heap_initial        int            Initial heap size allocated for the driver.
heap_max            int            Maximum attainable heap size for the driver.
skb_mpool_initial   int            Driver's minimum private socket buffer memory pool size.
skb_mpool_max       int            Maximum attainable private socket buffer memory pool size for the driver.
~ # _
```

重新启动ESX主机后，您可以通过使用以下命令检查以确保调试已在日志中完成：

```
tail /var/log/vmkernel.log | grep "<6>"
```