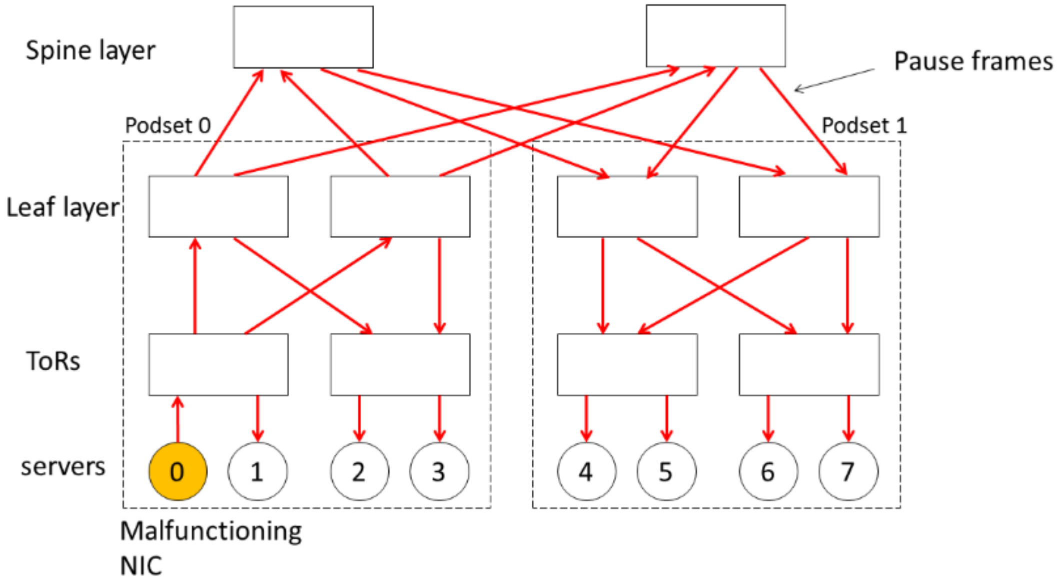
4.3 NIC PFC pause frame storm

> PFC pause frames prevent packets from been dropped by pausing the upstream devices. But PFC can cause collateral damage to innocent flows due to the head-ofline blocking. We illustrate the worst-case scenario in Figure 5 1: 1. The malfunctioning NIC of server 0 continually sends pause frames to its ToR switch; 2. The ToR switch in turn pauses all the rest ports including all the upstream ports to the Leaf switches; 3. The Leaf switches pause the Spine switches; 4. The Spine switches pause the rest of the Leaf switches; 5. The rest of the Leaf switches pause their ToR switches; 6. The ToR switches pause the servers that connect to them. In this case, a single malfunctioning NIC may block the entire network from transmitting. We call this NIC PFC pause frame storm, or PFC storm for abbreviation. To our surprise, we observed PFC storms in our networks multiple times and PFC storms caused several incidents which we will describe in Section 6. The root-cause of the PFC storm problem is a bug in the NIC's receiving pipeline. The bug stopped the NIC from handling the packets it received. As a result, the NIC's receiving buffer filled, and the NIC began to send out pause frames all the time. We have worked with the NIC provider to fix this NIC bug. Furthermore, to prevent PFC storms from hurting the network, we have implemented two watchdogs at both the NIC and the ToR switches as follows. On the NIC side, we worked with the NIC provider to build a PFC storm prevention watchdog. This is possible because the NIC has a separate micro-controller which can be used to monitor the NIC receiving side pipeline. Once the NIC micro-controller detects the receiving pipeline has been stopped for a period of time (default to 100ms) and the NIC is generating the pause frames, the micro-controller will disable the NIC from generating pause frames. Our experience with PFC storm is that once the NIC enters the storm mode, the server is disconnected from the network since the NIC is not functioning well anymore. The NIC watchdog is not able to rescue the server. Instead, its goal is to prevent the pause frame storms from hurting the rest of the network. On the ToR switch side, we worked with the switch providers to build a switch watchdog to monitor the server facing ports. Once a server facing egress port is queuing packets which cannot be drained, and at the same time, the port is receiving continuous pause frames from the NIC, the switch will disable the lossless mode for the port and discard the lossless packets to and from the NIC. Similar to the NIC side watchdog, it is to prevent pause frames from the malfunctioning NIC from propagating into the network. Once the switch detects that the pause frames from the NIC disappear for a period of time (default to 200ms), it will re-enable the lossless mode for the port. These two watchdogs are complementary to each other. One of them should be sufficient to stop the NIC PFC storm. We have implemented both for double insurance. Note that there is a small difference in the two watchdog implementations. The switch watchdog will reenable the lossless mode once pause frames are gone, whereas the NIC watchdog does not re-enable the lossless mode. This is because we have observed once the NIC enters the PFC storm mode, it never comes back. Hence re-enabling the lossless mode is not needed for the NIC. We also have observed that the NIC PFC storm problem typically can be fixed by a server reboot. Hence once the NIC is not functioning, our server management system will try to repair (reboot, reimage etc.) the server. Repairing takes tens of minutes. The NIC watchdog is to limit the damage of the problematic NIC to hundreds of milliseconds before server repair kicks in. Once the server is repaired successfully and pause frames from the servers are gone, the switch can reenable the lossless mode for the corresponding switch port automatically. Knowledgable readers may wonder about the interactions between the two watchdogs. Once the NIC watchdog disables the NIC pause frames, the switch watchdog will re-enable the lossless mode for the corresponding switch port. The packets to the NIC will either dropped by the switch (if the MAC address of the NIC times out) or dropped by the NIC (since the NIC receiving pipeline is not functioning). In either case, the NIC PFC storm cannot cause damage to the whole network. We recommend both switches and NICs should implement the watchdogs for NIC PFC storm prevention.

PFC 暂停帧通过暂停上游设备来防止数据包丢失。但由于头线阻塞，PFC 可能会对无辜流量造成附带损害。我们在图 5 1 中说明了最坏的情况：

1. 服务器 0 发生故障的 NIC 不断向其 ToR 交换机发送暂停帧；
2. ToR交换机依次暂停所有剩余端口，包括所有到Leaf交换机的上行端口；
3. Leaf交换机暂停Spine交换机；
4. Spine交换机暂停其余Leaf交换机；
5. 其余Leaf交换机暂停其ToR交换机；
6. ToR 交换机暂停连接到它们的服务器。
   在这种情况下，单个故障的 NIC 可能会阻止整个网络传输。我们称这种网卡为PFC暂停帧风暴，简称PFC风暴。令我们惊讶的是，我们在网络中多次观察到 PFC 风暴，并且 PFC 风暴导致了几起事件，我们将在第 6 节中描述这些事件。
   *PFC 风暴问题的根本原因是 NIC 接收管道中的错误。该错误使 NIC 无法处理它收到的数据包。*结果，网卡的接收缓冲区被填满，网卡开始一直向外发送暂停帧。
   我们已与 NIC 提供商合作修复此 NIC 错误。此外，为了防止 PFC 风暴损害网络，我们在 NIC 和 ToR 交换机上实现了两个看门狗，如下所示：
   在NIC方面，我们与NIC提供商合作构建了PFC风暴预防看门狗。
7. 这是可能的，因为 NIC 有一个单独的微控制器，可用于监视 NIC 接收侧管道。一旦网卡微控制器检测到接收管道已停止一段时间（默认为100ms）并且网卡正在生成暂停帧，微控制器将禁止网卡生成暂停帧。
8. 我们对 PFC 风暴的经验是，一旦 NIC 进入风暴模式，服务器就会与网络断开连接，因为 NIC 不再正常工作。 NIC 看门狗无法拯救服务器！事实上，其目标是防止暂停帧风暴损害网络的其余部分。
9. 在 ToR 交换机方面，我们与交换机提供商合作构建了一个交换机看门狗来监控面向服务器的端口。一旦面向出口端口的服务器有无法排出的数据包正在排队，同时该端口从 NIC 接收连续的暂停帧，交换机将禁用该端口的无损模式，并丢弃往返于该端口与网卡之间的无损数据包。
10. 与 NIC 端看门狗类似，它是为了防止故障 NIC 的暂停帧传播到网络中。一旦交换机检测到网卡发出的暂停帧消失一段时间（默认200ms），就会重新启用该端口的无损模式。
11. 这两个看门狗是相辅相成的。其中之一应该足以阻止 NIC PFC 风暴。我们已经实施了双重保险。
12. 请注意，两个看门狗的实现存在细微差别。一旦暂停帧消失，交换机看门狗将重新启用无损模式，而 NIC 看门狗不会重新启用无损模式。这是因为我们观察到一旦 NIC 进入 PFC 风暴模式，它就再也不会回来。因此，NIC 不需要重新启用无损模式。

13. 我们还观察到，NIC PFC 风暴问题通常可以通过服务器重新启动来解决。因此，一旦网卡无法正常工作，我们的服务器管理系统将尝试修复（重新启动、重新映像等）服务器。修复需要几十分钟。网卡看门狗的作用是在服务器修复开始之前将有问题的网卡的损坏限制在数百毫秒内。*一旦服务器修复成功并且来自服务器的暂停帧消失，交换机可以自动重新启用相应交换机端口的无损模式*
14. 知识渊博的读者可能想知道这两个监管机构之间的相互作用： 一旦网卡看门狗禁用网卡暂停帧，交换机看门狗将重新启用相应交换机端口的无损模式。发往 NIC 的数据包将被交换机丢弃（如果 NIC 的 MAC 地址超时），或者被 NIC 丢弃（因为 NIC 接收管道不工作）。无论哪种情况，网卡PFC 风暴都不会对整个网络造成损害。我们建议交换机和 NIC 都应实施用于 NIC PFC 风暴预防的看门狗。


Spine layer
Pause frames
Podset 0
Podset 1
Leaf layer
ToRs
servers
0
1
2
3
4
5
6
7
Malfunctioning
NIC