NVIDIA DOCA IPS

参考应用指南

MLNX-15-060464 _v1.2|2022 年 1 月

目录

第 1 章 简介

入侵防御系统 (IPS) 是一种监控网络是否存在恶意活动或违反政策的应用程序。

IPS 使用深度数据包检测 (DPI) 引擎根据预定义的 Suricata 签名扫描网络流中的恶意内容。被视为恶意的数据包将被丢弃并打印相应的消息。

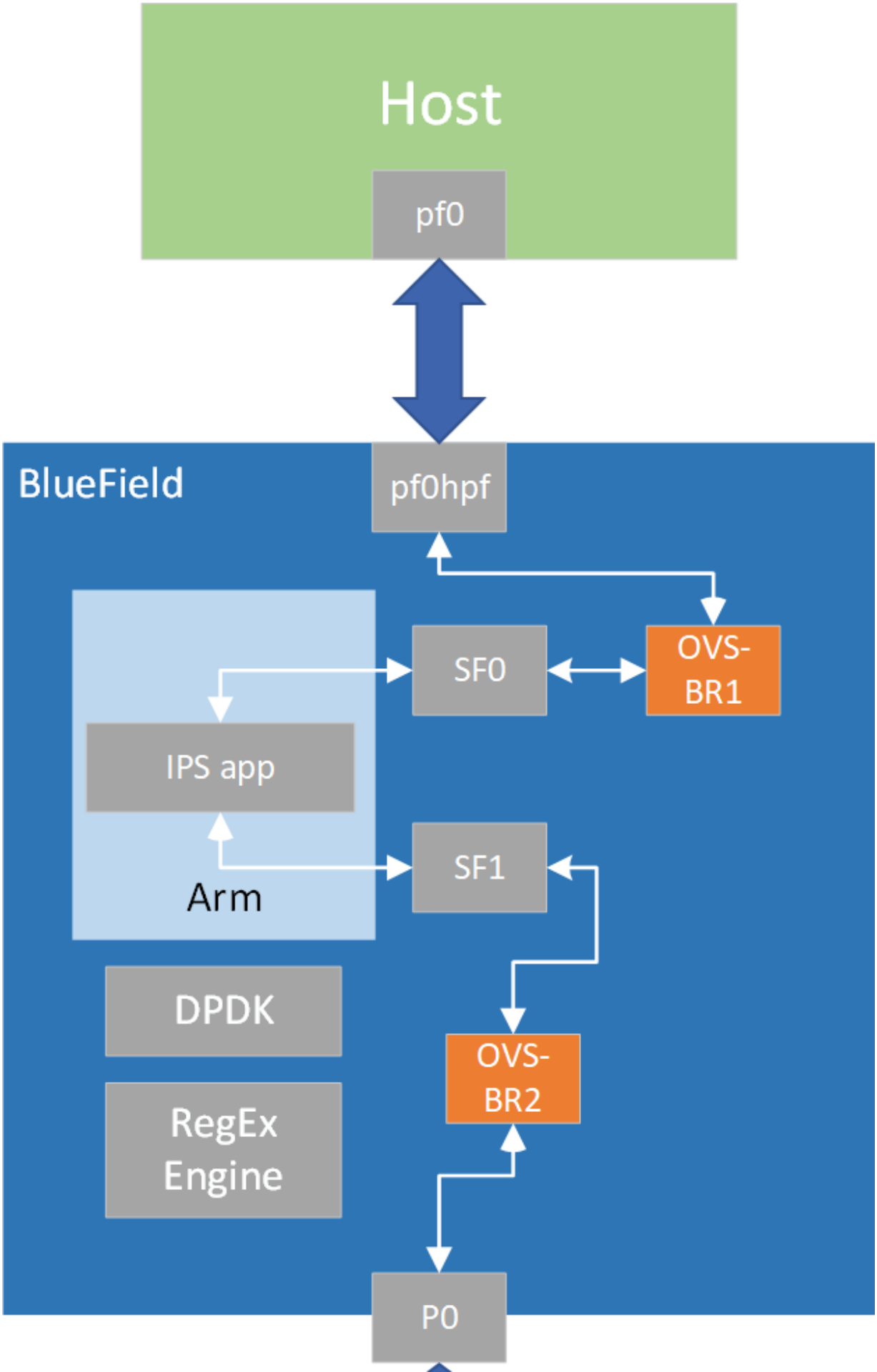
IPS 支持 NetFlow 协议，可将数据从 DPU 发送到远程 NetFlow 收集器进行进一步分析。

还支持连接跟踪以跟踪所有网络连接或流，这有助于识别组成流的所有数据包，从而更好地处理网络流量。

本文档介绍了如何在主机和 DPU 上构建和运行 IPS 应用程序。

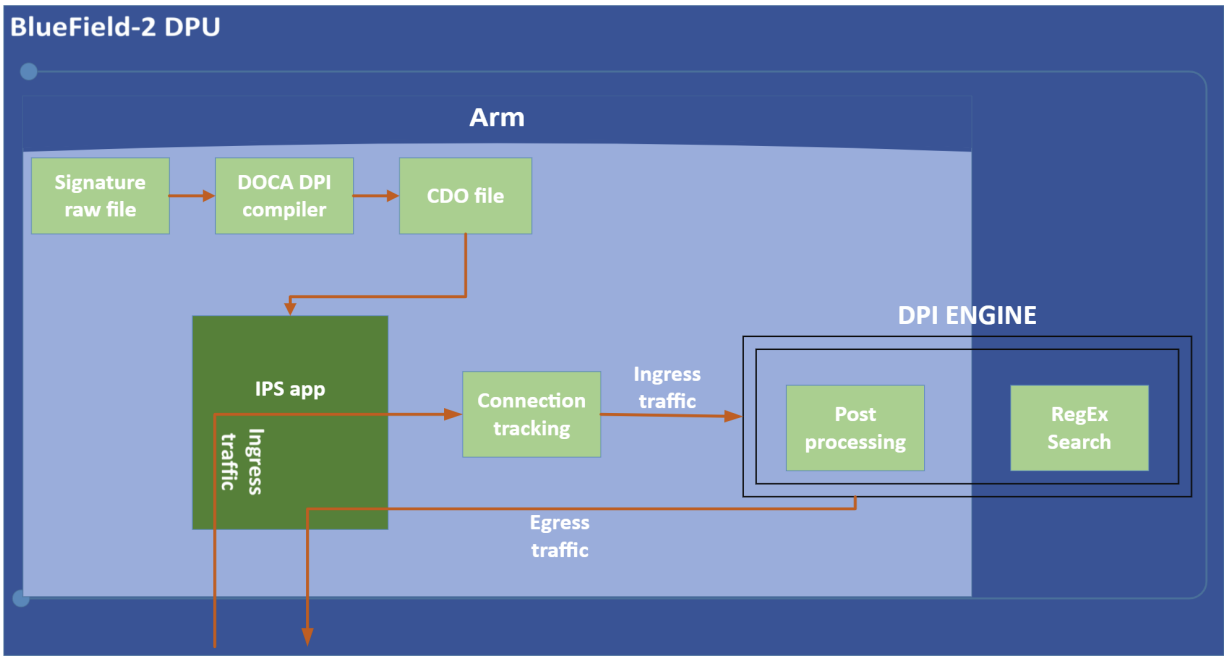
第 2 章 系统设计

IPS 应用程序设计为在 BlueField 实例上作为“线路碰撞”运行，它拦截来自线路的流量，并将其传递给对等端口。


Host
pf0
BlueField
pf0hpf
SF0
OVS-
BR1
IPS app
Arm
SF1
DPDK
RegEx
Engine
OVS-
BR2
P0

第 3 章 应用程序架构

IPS 在基于 DPDK 的状态流跟踪 (SFT) 上运行，以识别每个数据包所属的流，然后使用 DPI 处理 L7 分类。



1. 签名由DPI编译器编译，然后加载到DPI引擎。参见DOCA DPI 编译器 了解更多信息。

2. 使用利用连接跟踪硬件卸载的状态表模块来识别入口流量。

3. 根据 DPI 引擎编译的签名数据库扫描流量。
4. 进行后期处理以做出匹配决策。
5. 识别匹配的流并将丢弃操作卸载到硬件以提高性能，因为不需要进一步检查。

6. 流终止由 SFT 中设置为 60 秒的可配置老化计时器完成。当流卸载时，无法跟踪和销毁。

第 4 章 配置流程

1.解析应用参数。

```
arg_parser_init();
```

a).初始化Arg Parser资源。

b). 注册DOCA通用标志。

```
注册_ips_params（）；
```

c). 注册IPS应用程序标志。

```
arg_parser_start();
```

d). 解析DPDK标志并调用rte_eal_init()功能。

e). 解析APP标志。

2.初始化DPDK。

```
dpdk_初始化（）；
```

a).初始化SFT。

b). 初始化DPDK端口，包括内存池分配。

3.初始化IPS应用资源，包括DPI引擎和NetFlow。

```
ips_初始化（）；
```

4.配置DPI报文处理。

```
ips_worker_lcores_运行（）；
```

a). 配置DPI入队数据包。

b). 将作业发送到 RegEx 引擎。

c).配置DPI出队数据包。

5.IPS破坏。

```
ips_destroy（）；
```

a).停止并释放DPI资源。

b). 销毁netflow资源。

c).停止SFT。

d). 免费IPS资源。

第 5 章。 正在运行应用程序蓝色领域

1. 参考DOCA 安装指南 有关如何安装BlueField相关软件的详细信息。
2. IPS 应用程序二进制文件位于 /选择/mellanox/doca/examples/ips/bin/doca_ips。
3. 重建应用程序：

   a). 运行：

   ```
   cd /opt/mellanox/doca/examples/ips/src meson /tmp/build
   忍者-C /tmp/build
   ```

   doca_ips是在 / 下创建的tmp/构建。

   b). 构建过程取决于PKG_CONFIG_PATH环境变量来定位 DPDK 库。如果变量意外损坏并且构建失败，请运行以下命令：

   - 对于 Ubuntu：

     ```
     导出 PKF_CONFIG_PATH=$PKG_CONFIG_PATH:/opt/mellanox/dpdk/lib/aarch64-linux-gnu/pkgconfig
     ```

   - 对于 CentOS：

     ```
     导出 PKG_CONFIG_PATH=$PKG_CONFIG_PATH:/opt/mellanox/dpdk/lib64/pkgconfig
     ```

4. 运行前设置

   a). IPS 示例基于 DPDK 库。因此，需要用户提供 DPDK 标志并分配大页面。运行：

   ```
   echo 2048 > /sys/kernel/mm/hugepages/hugepages-2048kB/nr_hugepages
   ```

   b). 确保 RegEx 引擎处于活动状态：

   ```
   systemctl status mlx-regex
   ```

   如果状态为非活动（活动：失败）跑步：

   ```
   systemctl 启动 mlx-regex
   ```

5. 运行应用程序：

   ```
   用法：doca_ips [DPDK 标志] -- [DOCA 标志] [程序标志]

   DOCA 标志：
     -h，--帮助                          打印帮助概要
     -l，--日志级别                      设置日志级别为了应用程序 <CRITICAL=0, DEBUG=4>

   程序标志：
     -p，--打印匹配                      与 DPI 引擎匹配时打印 FID
   ```

```
  -n，--netflow                          收集 netflow 统计数据并根据
conf 文件
  -o，--output-csv <路径>                CSV 文件输出的路径 从有效 PDD 编译的 CDO 文件的路径
  -c，--cdo <路径>
```

例如：

```
/opt/mellanox/doca/examples/ips/bin/doca_ips -a 0000:03:00.0,class=regex -a
 辅助：mlx5_core.sf.4，sft_en=1 -a 辅助：mlx5_core.sf.5，sft_en=1 –– -- cdo /root/ips.cdo -p -n
```

笔记：SFT 最多支持 64 个队列。因此，应用程序无法使用超过 64 个核心运行。要限制核心数量，请运行：

```
/opt/mellanox/doca/examples/ips/bin/doca_ips -a 0000:03:00.0,class=regex -a 辅助：
mlx5_core.sf.4,sft_en=1 -a 辅助：mlx5_core.sf.5,sft_en=1 - l 0-64 –– --cdo /root/ips.cdo -p -n
```

这会将应用程序限制为使用 65 个核心（核心 0 到核心 64）。也就是说，1 个核心用于主线程，64 个核心用作工作线程。

使用 JSON 文件：

```
doca_ips --json [json_文件]
```

例如：

```
/opt/mellanox/doca/examples/ips/bin/doca_ips –-json /root/ips_params.json
```

笔记：必须根据以下要求启用子功能可扩展功能设置指南 。

笔记：旗帜 -0000：03：00.0，class = regex -a 辅助：mlx5_core.sf.4，sft_en = 1 -a 辅助：mlx5_core.sf.5，sft_en = 1是
正确使用应用程序所必需的。修改这些标志会导致意外行为，因为仅支持 2 个端口。SF 编号是任意且可配置的。但是，RegEx 设备不是，必须在端口 0 上启动。

有关 DPDK 可用标志的更多信息，请使用 -时长在 --separator 之前：

```
/opt/mellanox/doca/examples/ips/bin/doca_ips -h
```

有关该应用程序的更多信息，请使用 -时长在 -- 分隔符之后：

```
/opt/mellanox/doca/examples/ips/bin/doca_ips -- -h
```

第 6 章 Arg 解析器 DOCA 标志

参阅NVIDIA DOCA Arg 解析器用户指南 了解更多信息。

旗帜类型	短旗	长标志/JSON 钥匙	描述	JSON 内容
DPDK 标志	一个	设备	将 PCIe 设备添加到 探测设备	“设备”： [{“设备”：“注册 前任”，“ID”：“00 {“设备”：“科幻”，“ID”：“4”，“的真的}， {“设备”：“科幻”，“ID”：“5”，“的真的}，]
	升	核心列表	列出要运行的核心	“核-列表”：“0-4”
通用旗帜	升	日志级别	设置应用程序的日志级别： ▸ 严重=0 ▸ 错误=1 ▸ 警告=2 ▸ 信息=3 ▸ 调试=4	“日志级别”：4
	时长	帮助	打印帮助概要	不适用
程序标志	页	打印匹配	打印 FID 时匹配的 DPI 引擎	“打印匹配”：真的
	n	净流	收集网络流量统计和	“网络流”：错误的

旗帜类型	短旗	长标志/JSON 钥匙	描述	JSON 内容
			根据发送 .conf文件	
	o	输出-csv	CSV 文件输出的路径	“输出-文件名：“/tmp/ips_stats.csv”
	丙	首席运营官	从编译的 CDO 文件的路径 有效的 PDD	“cdo”：“/tmp/ips.cdo”

第七章。 正在运行应用程序主持人

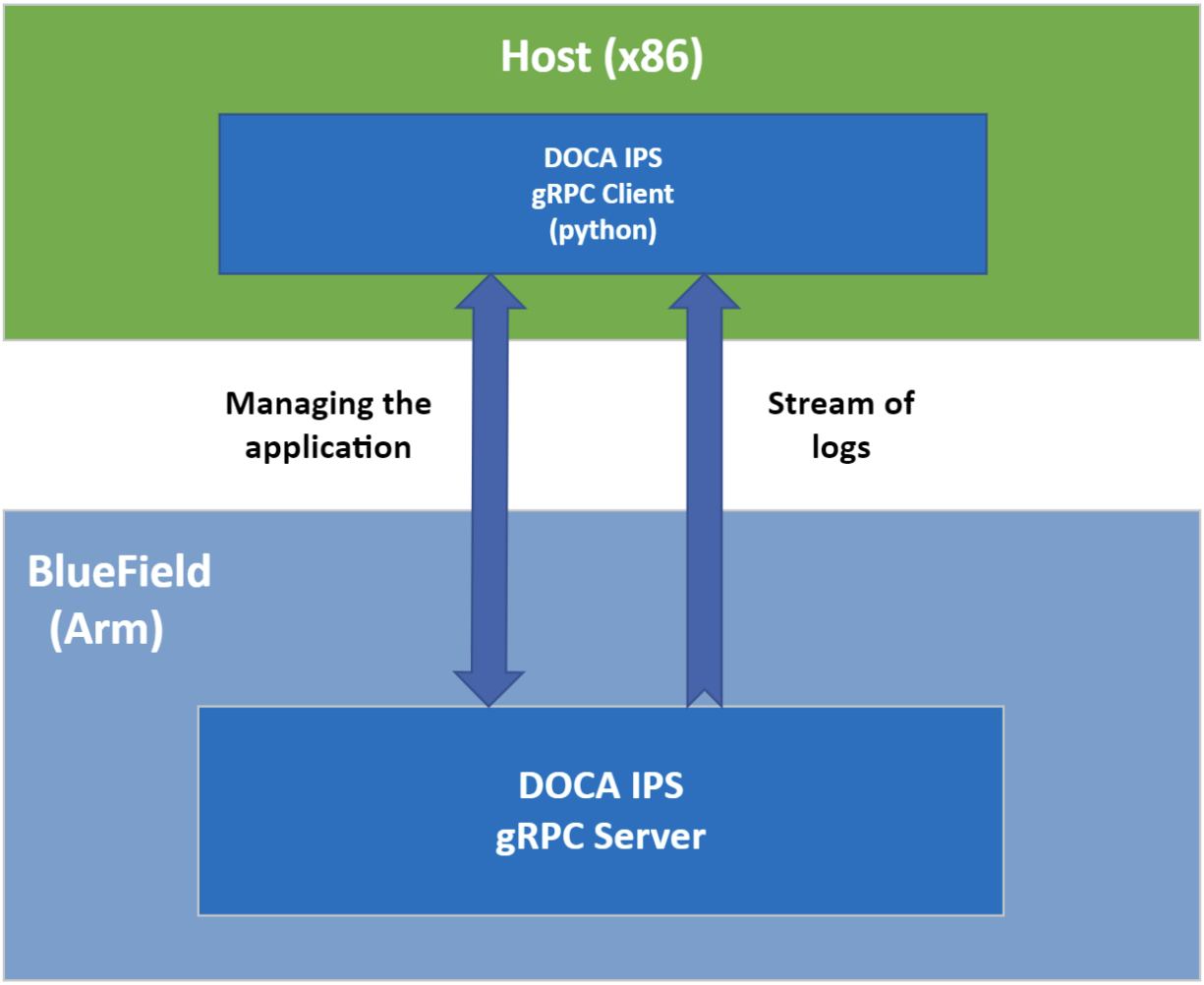
主机执行示例：

```
cd /opt/mellanox/doca/examples/ips/
./doca_ips -a 0000：21：00.0，class=regex -a 0000：21：00.3 -a 0000：21：00.4 -- --cdo ~/ips.cdo
```

请参阅NVIDIA DOCA 虚拟功能用户指南 。

第八章。 管理启用 gRPC 的功能主办方申请

有关在 BlueField 上运行 gRPC 应用服务器的说明，请参阅NVIDIA DOCA gRPC 基础设施用户指南。



要运行支持 gRPC 的应用程序的 Python 客户端：

```
./doca_ips_gRPC_client.py -d/--debug <服务器地址[:服务器端口]>
```

例如：

```
/opt/mellanox/doca/examples/ips/bin/grpc/client/doca_ips_gRPC_client.py 192.168.104.2
```

笔记：请参阅已知问题 2872829NVIDIA DOCA 发行说明 关于使用 DOCA gRPC 编排器部署支持 gRPC 的应用程序。

笔记：请参阅已知问题 2872883NVIDIA DOCA 发行说明 关于 gRPC python 客户端的执行。

第九章。 部署容器化应用程序

IPS 示例支持基于容器的部署。请参阅NVIDIA DOCA 容器部署指南 了解更多信息。

特定于应用程序的配置步骤可以在 NGC 的应用程序下找到 容器页面 。

第 10 章 参考文献

- /opt/mellanox/doca/示例/ips/src/ips.c
- /opt/mellanox/doca/示例/ips/src/grpc/ips.proto
- /opt/mellanox/doca/示例/ips/bin/ips_suricata_rules_example

注意

本文件仅供参考，不应视为对产品的特定功能、状况或质量的保证。NVIDIA Corporation 及其任何直接或间接子公司和附属公司（统称“NVIDIA”）不对本文件中所含信息的准确性或完整性作出任何明示或暗示的陈述或保证，也不对其中的任何错误承担任何责任。NVIDIA 对此类信息的后果或使用，或因使用此类信息而导致的任何专利或其他第三方权利侵权不承担任何责任。本文件不承诺开发、发布或交付任何材料（定义如下）、代码或功能。

NVIDIA 保留随时对本文档进行更正、修改、增强、改进和任何其他更改的权利，恕不另行通知。

客户应在下订单之前获取最新的相关信息，并应验证此类信息是最新且完整的。

NVIDIA 产品的销售须遵守订单确认时提供的 NVIDIA 标准销售条款和条件，除非 NVIDIA 授权代表与客户签署的单独销售协议（“销售条款”）另有约定。NVIDIA 在此明确反对将任何客户一般条款和条件应用于本文件中提及的 NVIDIA 产品购买。本文件不直接或间接构成任何合同义务。

NVIDIA 产品的设计、授权或保证不适用于医疗、军事、飞机、太空或生命支持设备，也不适用于 NVIDIA 产品故障或失灵可能导致人身伤害、死亡或财产或环境损害的应用。NVIDIA 对在此类设备或应用中纳入和/或使用 NVIDIA 产品不承担任何责任，因此此类纳入和/或使用由客户自行承担风险。

NVIDIA 不保证基于本文档的产品适合任何特定用途。NVIDIA 不一定会测试每种产品的所有参数。客户应自行负责评估和确定本文档中包含的任何信息的适用性，确保产品适合客户计划的应用程序，并对应用程序进行必要的测试，以避免应用程序或产品出现故障。客户产品设计中的缺陷可能会影响 NVIDIA 产品的质量和可靠性，并可能导致本文档中未包含的额外或不同的条件和/或要求。NVIDIA 不承担与任何违约、损害、成本或问题相关的任何责任，这些违约、损害、成本或问题可能基于或归因于：(i) 以任何违反本文档的方式使用 NVIDIA 产品或 (ii) 客户产品设计。

本文件项下的任何 NVIDIA 专利权、版权或其他 NVIDIA 知识产权均未明示或暗示授予任何许可。NVIDIA 发布的有关第三方产品或服务的信息并不构成 NVIDIA 使用此类产品或服务的许可或此类产品或服务的保证或认可。使用此类信息可能需要根据第三方的专利或其他知识产权获得第三方许可，或根据 NVIDIA 的专利或其他知识产权获得 NVIDIA 许可。

仅当事先获得 NVIDIA 书面批准、未经修改地复制且完全遵守所有适用的出口法律和法规并附带所有相关条件、限制和声明时，才允许复制本文档中的信息。

本文档以及所有 NVIDIA 设计规范、参考板、文件、图纸、诊断、列表和其他文档（统称或单独称为“材料”）均按“原样”提供。NVIDIA 不就材料作出任何明示、暗示、法定或其他形式的保证，并明确否认所有关于非侵权、适销性和适用于特定用途的暗示保证。在法律允许的范围内，NVIDIA 在任何情况下均不对因使用本文档而造成的任何损害负责，包括但不限于任何直接、间接、特殊、偶发、惩罚性或后果性损害，无论该等损害是如何造成的，也无论责任理论如何，即使 NVIDIA 已被告知存在此类损害的可能性。无论客户因何种原因可能遭受任何损害，NVIDIA 对客户就本文所述产品承担的累计责任应根据产品销售条款进行限制。

商标

NVIDIA、NVIDIA 徽标和 Mellanox 是 Mellanox Technologies Ltd. 和/或 NVIDIA Corporation 在美国和其他国家/地区的商标和/或注册商标。注册商标 Linux®是根据 Linux 基金会的再授权而使用的，Linux 基金会是 Linus Torvalds 的独家授权者，Linus Torvalds 是该商标的全球所有者。其他公司和产品名称可能是与其相关的各自公司的商标。

版权

NVIDIA 公司 | 2788 San Tomas Expressway，圣克拉拉，CA 95051 http://www.nvidia.com