

目 录

1 Context	1-1
1.1 Context简介	1-1
1.1.1 Context的应用.....	1-1
1.1.2 缺省Context和非缺省Context	1-1
1.2 Context配置限制和指导	1-2
1.2.1 分配VLAN时的注意事项	1-2
1.2.2 分配接口时的注意事项	1-2
1.3 Context配置任务简介	1-3
1.4 创建Context	1-3
1.5 为Context分配资源	1-4
1.5.1 为Context分配CPU/磁盘/内存资源.....	1-4
1.5.2 为Context分配接口.....	1-5
1.5.3 为Context分配VLAN.....	1-5
1.5.4 限制Context的吞吐量.....	1-6
1.5.5 限制Context安全策略规则总数.....	1-6
1.5.6 限制Context会话并发数.....	1-6
1.5.7 限制Context会话新建速率.....	1-7
1.6 启动Context	1-7
1.7 访问和管理Context	1-8
1.8 Context显示和维护	1-8
1.9 Context典型配置举例	1-9

1 Context

1.1 Context简介

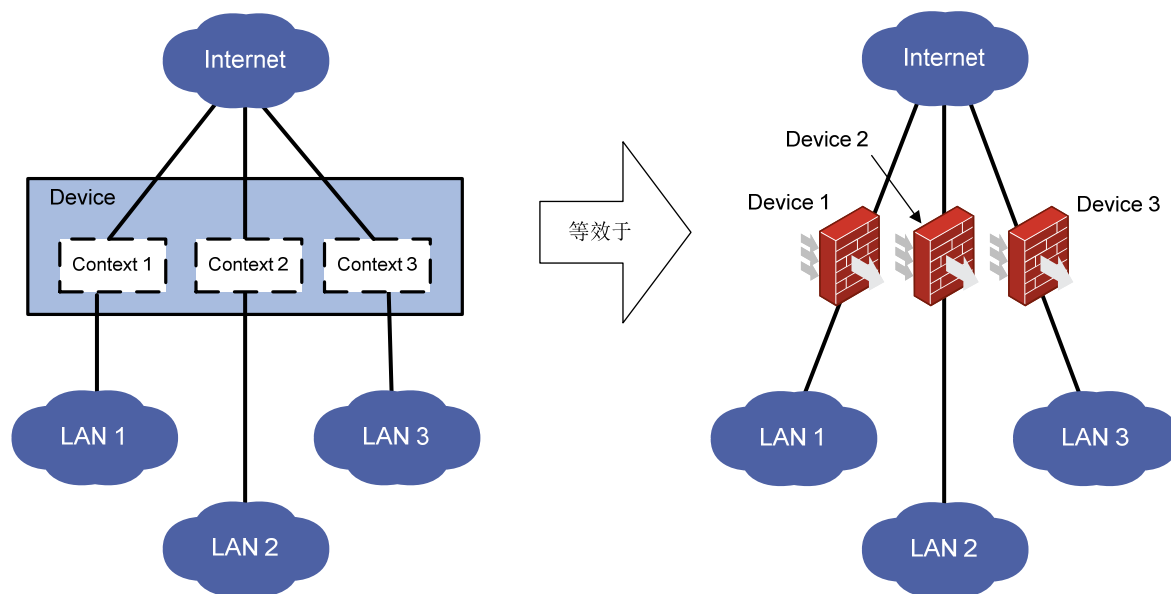
通过虚拟化技术将一台物理设备划分成多台逻辑设备，每台逻辑设备就称为一个 **Context**。每个 **Context** 拥有自己专属的软硬件资源，独立运行。

对于用户来说，每个 **Context** 就是一台独立的设备，方便管理和维护；对于管理者来说，可以将一台物理设备虚拟成多台逻辑设备供不同的分支机构使用，可以保护现有投资，提高组网灵活性。

1.1.1 Context的应用

如 [图 1-1](#) 所示，LAN 1、LAN 2 和 LAN 3 是三个不同的局域网，它们通过同一台设备连接到外网。通过虚拟化技术，能让一台设备当三台设备使用。具体做法是，在 **Device** 上创建三个 **Context**（**Context 1**、**Context 2**、**Context 3**），分别负责 LAN 1、LAN 2、LAN 3 的安全接入。LAN 1、LAN 2、LAN 3 的网络管理员可以（也只能）分别登录到自己的设备进行配置、保存、重启等操作，不会影响其它网络的使用，其效果等同于 LAN 1、LAN 2 和 LAN 3 分别通过各自的设备 **Device 1**、**Device 2**、**Device 3** 接入 Internet。

图1-1 Context 组网示意图



1.1.2 缺省Context和非缺省Context

- 设备支持Context功能后，整台物理设备就是一个Context，称为缺省Context，如 [图 1-1](#) 中的 **Device**。当用户登录物理设备时，实际登录的就是缺省Context。用户在物理设备上的配置实质就是对缺省Context的配置。缺省Context的名称为Admin，编号为1。缺省Context不需要创建，不能删除。

- 与缺省Context相对应的是非缺省Context，如 [图 1-1](#) 中的Context 1、Context 2、Context 3。非缺省Context是管理员在设备上通过命令行创建的，可分配给不同的接入网络使用。
- 缺省 Context 拥有对整台物理设备的所有权限，它可以使用和管理设备所有的资源。缺省 Context 下可以创建/删除非缺省 Context，给非缺省 Context 分配 CPU 资源/磁盘/内存空间、接口、VLAN，没有分配的 CPU 资源/磁盘/内存空间、接口、VLAN 由缺省 Context 使用和管理。
- 非缺省 Context 下不可再创建/删除非缺省 Context，它只能使用缺省 Context 分配给自己的资源，并在缺省 Context 指定的资源限制范围内工作，不能抢占其他 Context 或者系统剩余的资源。
- 非缺省 Context 下不支持共享口的报文捕获功能，关于报文捕获功能的详细描述请参见“网络管理和监控配置指导”中的“报文捕获配置”。

1.2 Context配置限制和指导

非缺省 Context 中的 DPI 业务功能使用缺省 Context 中的应用层检测引擎对报文进行匹配，当创建、删除、关闭和重启非缺省 Context 时，缺省 Context 中的应用层检测引擎会重新激活，激活期间设备上的所有 Context 均不能对报文进行 DPI 业务处理。

1.2.1 分配VLAN时的注意事项

- 对于共享 VLAN，请先在缺省 Context 内创建 VLAN，再通过 `allocate vlan` 命令将指定 VLAN 分配给指定的 Context 使用。
- VLAN 1 不能被共享。
- 端口的缺省 VLAN 不能被共享。
- 已经创建了 VLAN 接口的 VLAN 不能被共享。

1.2.2 分配接口时的注意事项

- (1) 有些接口可以创建子接口，这样的接口我们称为父接口。分配父接口与子接口时：
 - 不能将子接口独占分配给 Context。
 - 如果子接口已经被分配，则不能再分配其父接口。
 - 如果父接口已经被分配，则不能再分配其子接口。
- (2) 分配聚合接口与成员接口时：
 - 聚合接口只能共享分配给 Context。
 - 不能将成员接口共享分配给 Context。
- (3) 如果接口已经被共享分配，则不能再独占分配。需将共享分配配置取消后，才能独占分配。
- (4) 不允许独占分配逻辑接口。
- (5) 禁止将 IRF 物理端口分配给自定义 Context。
- (6) 当三层物理子接口与聚合子接口作为冗余口的成员端口时，禁止把其主接口共享给自定义 Context。

1.3 Context配置任务简介

表1-1 Context 配置任务简介

配置任务		说明	详细配置
创建Context		必选	1.4
为Context分配资源	为Context分配CPU/磁盘/内存资源	必选	1.5.1
	为Context分配接口	必选	1.5.2
	为Context分配VLAN	可选	1.5.3
	限制Context的吞吐量	可选	1.5.4
	限制Context安全策略规则总数	可选	1.5.5
	限制Context会话并发数	可选	1.5.6
	限制Context会话新建速率	可选	1.5.7
启动Context		必选	1.6
访问和管理Context		必选	1.7

1.4 创建Context

创建 Context 相当于构造了一台新的设备。

创建 Context 时，通过 **vlan-unshared** 参数可选择是否和其它 Context 共享 VLAN：

- 如果选择和其它 Context 共享 VLAN，需要在缺省 Context 内创建并配置 VLAN，再分配给非缺省 Context。共享 VLAN 由多个 Context 共同所有。VLAN 1 为系统缺省 VLAN，由缺省 Context 独有，不能分配给非缺省 Context。
- 如果选择不和其它 Context 共享 VLAN，请登录该 Context，并使用 **vlan** 命令创建 VLAN 2～VLAN 4094。VLAN 1 为缺省 VLAN，用户不能手工创建和删除。Context 各自使用和管理 VLAN，互不干扰。

表1-2 创建 Context

操作	命令	说明
进入系统视图	system-view	-
创建Context，并进入Context视图	context context-name [id context-id] [vlan-unshared]	缺省情况下，设备上存在缺省Context，名称为Admin，编号为1
配置Context的描述信息	description text	缺省情况下，缺省Context描述信息为DefaultContext。非缺省Context没有配置描述信息

1.5 为Context分配资源

1.5.1 为Context分配CPU/磁盘/内存资源

如果设备上创建了多个 Context，这些 Context 会共享设备的 CPU/磁盘/内存资源，为了防止一个 Context 过多的占用 CPU/磁盘/内存，而导致其它 Context 无法运行，需要限制 Context 对 CPU/磁盘/内存资源的使用。

(1) CPU 权重

当 CPU 无法满足所有 Context 的处理需求时，系统将按照 CPU 权重值为每个 Context 分配处理时间。通过调整 Context 的权重，可以使指定的 Context 获得更多的 CPU 资源，保证关键业务的运行。例如：在三个 Context 中，将处理关键业务的 Context 的 CPU 权重设置为 2，其余两个 Context 的 CPU 权重设置为 1，则当 CPU 处理能力不足时，将为关键业务 Context 提供 2 倍于其它 Context 的处理时间。

(2) 磁盘空间上限

执行 **limit-resource disk** 命令前，请使用 **display context resource** 命令查看 Context 当前实际已经使用的磁盘空间大小。配置值应大于 Context 当前实际已经使用的磁盘空间大小，否则，会导致 Context 申请新的磁盘空间失败，从而无法进行文件夹创建、文件拷贝和保存等操作。

请在 Context 启动后，配置磁盘上限。因为，Context 创建后，如果没有启动，磁盘使用值为 0，此时如果配置磁盘上限，请尽量不要配置过小的上限，否则，可能导致 Context 启动不了。

(3) 内存空间上限

执行 **limit-resource memory** 命令前，请使用 **display context resource** 命令查看 Context 当前实际已经使用的内存空间大小。配置值应大于 Context 当前实际已经使用的内存空间大小，否则，会导致 Context 申请内存失败引起功能异常。

请在 Context 启动后，再配置内存上限，并且配置的上限值不应过小，以免 Context 内业务申请不到内存后引起功能不正常。

表1-3 为 Context 分配 CPU/磁盘/内存资源

操作	命令	说明
进入系统视图	system-view	-
进入Context视图	context context-name	-
指定Context的CPU权重	limit-resource cpu weight weight-value	缺省情况下，Context的CPU权重为10
配置Context可使用的磁盘空间上限	limit-resource disk slot slot-number cpu cpu-number ratio limit-ratio	缺省情况下，Context可以使用其所在成员设备上的所有磁盘空间 如果设备上有多块磁盘，该命令对所有磁盘生效
配置Context可使用的内存空间上限	limit-resource memory slot slot-number cpu cpu-number ratio limit-ratio	缺省情况下，Context可以使用其所在成员设备上的所有内存空间，每个Context可使用的内存空间上限为空闲内存空间值

1.5.2 为Context分配接口

设备上的所有接口都属于缺省 Context, 不属于任何非缺省 Context。请给非缺省 Context 分配接口, 它才能和网络中的其它设备通信。

为了提高设备接口的利用率, 在给 Context 分配接口时, 可以选择:

- 独占方式分配 (不带 **share** 参数)。使用该方式分配的接口仅归该 Context 所有、使用。用户登录该 Context 后, 能查看到该接口, 并执行接口支持的所有命令。
- 共享方式分配 (带 **share** 参数): 表示将一个接口分配给多个 Context 使用, 这些 Context 共享这个物理接口, 但是在各个 Context 内会创建一个同名的虚接口, 这些虚接口具有不同的 MAC 地址和 IP 地址。设备从共享的物理接口接收报文后交给对应的虚拟接口处理; 出方向, 虚拟接口处理完报文后, 会交给共享的物理接口发送。使用该方式, 可以提高设备接口的利用率。通过共享方式分配的接口:
 - 在缺省 Context 内仍然存在该接口, 该接口可执行接口支持的所有命令;
 - 在非缺省 Context 内, 会新建一个同名接口, 用户登录这些 Context 后, 能查看到该接口, 但只能执行 **shutdown**、**description** 以及网络/安全相关的命令。



注意

- 当三层子接口作为冗余口的成员端口时, 禁止把其主接口共享给自定义 Context。
- 不允许独占分配逻辑接口

表1-4 为 Context 分配接口

操作	命令	说明
进入系统视图	system-view	-
进入Context视图	context context-name	-
为Context分配接口	allocate interface { <i>interface-type</i> <i>interface-number</i> } <1-24> [share] allocate interface <i>interface-type</i> <i>interface-number1 to interface-type</i> <i>interface-number2</i> [share]	二者选其一 缺省情况下, 设备上的所有接口都属于缺省Context, 不属于任何非缺省Context

1.5.3 为Context分配VLAN

创建 Context 时, 如果不选择 **vlan-unshared** 参数, 则表示和其它 Context 共享 VLAN。对于共享 VLAN, 请先在缺省 Context 内创建 VLAN, 再通过 **allocate vlan** 命令将指定 VLAN 分配给指定的 Context 使用。

表1-5 为 Context 分配 VLAN

操作	命令	说明
进入系统视图	system-view	-
进入Context视图	context context-name	-

操作	命令	说明
为Context分配VLAN	allocate vlan <i>vlan-id</i> &<1-24>	二者选其一
	allocate vlan <i>vlan-id1</i> to <i>vlan-id2</i>	缺省情况下，没有为Context分配VLAN

1.5.4 限制Context的吞吐量

为了防止一个 Context 的报文过多而导致其它 Context 的报文被丢弃，需要限制 Context 的吞吐量。当启用吞吐量限制时，系统优先处理协议报文，对于超过限制值的业务报文会被丢弃。。

表1-6 限制 Context 的吞吐量

操作	命令	说明
进入系统视图	system-view	-
进入Context视图	context <i>context-name</i>	-
设置Context的吞吐量限制	capability throughput { kbps pps } <i>value</i>	缺省情况下，各Context不做吞吐量限制，按实际能力转发

1.5.5 限制Context安全策略规则总数

一个 Context 内可以配置多条安全策略规则。如果不加限制，会出现大量规则占用过多内存的情况，影响 Context 的其它功能正常运行。所以，请根据需要为 Context 设置安全策略规则总数限制。当规则总数达到限制值时，后续不能新增规则。关于安全策略的详细描述请参见“安全配置指导”中的“安全策略”。

表1-7 限制 Context 安全策略规则总数

操作	命令	说明
进入系统视图	system-view	-
进入Context视图	context <i>context-name</i>	-
设置Context的安全策略规则总数限制	capability security-policy-rule maximum <i>max-value</i>	缺省情况下，不限制Context的安全策略规则总数

1.5.6 限制Context会话并发数

如果一个 Context 建立了太多会话表会导致其他 Context 的会话由于内存不够而无法建立，为了防止这种情况，需要限制 Context 建立会话表的数量。

需要注意的是：Context 会话并发数限制对本机流量不生效，例如：FTP、Telnet、SSH 和 HTTP 等业务。

表1-8 为 Context 限制会话并发数

操作	命令	说明
进入系统视图	system-view	-
进入Context视图	context context-name	-
设置Context的单播会话并发数限制	capability session maximum max-number	缺省情况下，不限制该Context允许的单播会话并发数

1.5.7 限制Context会话新建速率

如果一个 Context 的会话新建速率过快会导致其他 Context 由于 CPU 处理能力不够而无法建立会话，为了防止这种情况，需要限制 Context 的会话新建速率。

需要注意的是：Context 会话新建速率限制对本机流量不生效，例如：FTP、Telnet、SSH 和 HTTP 等业务。

表1-9 为 Context 限制会话新建速率

操作	命令	说明
进入系统视图	system-view	-
进入Context视图	context context-name	-
设置Context的会话新建速率限制	capability session rate max-value	缺省情况下，不限制该Context允许的会话新建速率

1.6 启动Context

Context 创建后需要执行 **context start** 命令，才能完成新 Context 的初始化，相当于上电启动。启动后，用户可以登录到该 Context 执行配置。

正常程序启动 Context 时，设备会先做一些检查（比如 Context 的主、备进程能否正常启动），满足条件后，才启动 Context，该命令会保证主备的 Context 状态一致，如果某成员设备上的 Context 启动失败，则会导致所有该 Context 进程启动失败。正常程序启动的 Context 能更好的保证 Context 的业务正常运行，所以，通常情况下，使用 **context start** 命令启动 Context 即可。**force** 参数用于以下场景：在 IRF 环境，如果主备倒换或者配置恢复过程中出现内存不足，会导致部分 Context 虽然可以处理业务，但因为它们的主、备进程状态不一致，这些 Context 一直停留在 **updating** 或者 **inactive** 状态。当内存资源恢复后，执行 **context start force** 命令，设备会在不中断业务的情况下，尽可能修复不正常的 Context 进程，让这些 Context 恢复到正常状态。



提示

在使用 **context start force** 前，用户可以通过 **display context**、**display system internal context configuration-status**、**display system internal context id context-id running-status** 命令查看 Context 的运行情况。

表1-10 启动 Context

操作	命令	说明
进入系统视图	system-view	-
进入Context视图	context context-name	-
启动Context	context start [force]	-

1.7 访问和管理Context

只要用户和设备之间路由可达,就能使用 **switchto context** 命令,通过设备和 Context 的内部连接,登录 Context。

表1-11 登录 Context

操作	命令	说明
进入系统视图	system-view	-
登录Context	switchto context context-name	必选

用户登录 Context 后,可以在 Context 的用户视图执行 **quit** 命令来退出登录。此时,命令视图将从当前 Context 的用户视图返回到缺省 Context 的系统视图。

除了上述方式,用户还可以通过 Context 上的接口,使用该 Context 的 IP 地址进行 Telnet/SSH 登录。

1.8 Context显示和维护

在完成 Context 相关配置后,在任意视图下执行 **display** 命令,可以显示配置后 Context 的运行情况,通过查看显示信息,来验证配置的效果。

在用户视图下,用户可以执行 **reset** 命令清除相关数据信息。

表1-12 Context 显示和维护

操作	命令
显示Context的相关信息	display context [name context-name]
显示Context的接口列表	display context [name context-name] interface
显示Context对CPU/磁盘/内存资源的使用情况	display context [name context-name] resource [cpu disk memory] [slot slot-number cpu cpu-number]
显示Context的VLAN列表	display context [name context-name] vlan

1.9 Context典型配置举例

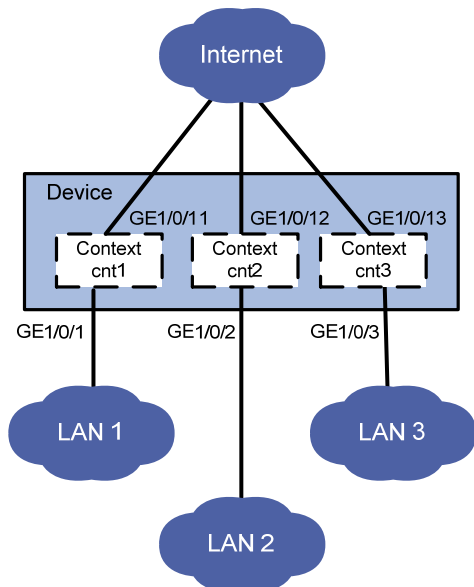
1. 组网需求

将设备 Device 虚拟成三台独立的设备：Context cnt1、Context cnt2、Context cnt3，并分给三个不同的用户网络用作接入设备。

- LAN 1 的用户多，业务需求复杂，因此需要给 Context cnt1 提供较大的磁盘/内存空间使用上限，以便保存配置文件、启动文件和系统信息等；Context cnt2 使用系统缺省的磁盘空间；LAN 3 人员规模小，上网流量比较少，对接入设备的配置及性能要求较低，因此对 Context cnt3 提供较低的 CPU 权重。
- GigabitEthernet1/0/1 和 GigabitEthernet1/0/11 分配给 Context cnt1、GigabitEthernet1/0/2 和 GigabitEthernet1/0/12 分配给 Context cnt2、GigabitEthernet1/0/3 和 GigabitEthernet1/0/13 分配给 Context cnt3。

2. 组网图

图1-2 Context 典型配置组网图



3. 配置步骤

(1) 配置 Context cnt1

创建 Context cnt1，设置描述信息。

```
[Sysname] context cnt1
[Sysname-context-2-cnt1] description context-1
```

配置 Context cnt1 的磁盘使用上限为 60%。

```
[Sysname-context-2-cnt1] limit-resource disk slot 1 cpu 0 ratio 60
```

配置 Context cnt1 的内存使用上限为 60%。

```
[Sysname-context-2-cnt1] limit-resource memory slot 1 cpu 0 ratio 60
```

配置 Context cnt1 的 CPU 权重为 8。

```
[Sysname-context-2-cnt1] limit-resource cpu weight 8
```

启动 Context cnt1。

```
[Sysname-context-2-cnt1] context start
It will take some time to start the context...
Context started successfully.
```

将接口 GigabitEthernet1/0/1 和 GigabitEthernet1/0/11 分配给 Context cnt1。

```
[Sysname-context-2-cnt1] allocate interface gigabitethernet 1/0/1 gigabitethernet 1/0/11
[Sysname-context-2-cnt1] quit
```

(2) 配置 Context cnt2

创建 Context cnt2，设置描述信息

```
[Sysname] context cnt2
[Sysname-context-3-cnt2] description context-2
```

启动 Context cnt2。

```
[Sysname-context-3-cnt2] context start
It will take some time to start the context...
Context started successfully.
```

将接口 GigabitEthernet1/0/2 和 GigabitEthernet1/0/12 分配给 Context cnt2。

```
[Sysname-context-3-cnt2] allocate interface gigabitethernet 1/0/2 gigabitethernet 1/0/12
[Sysname-context-3-cnt2] quit
```

(3) 配置 Context cnt3

创建 Context cnt3，设置描述信息

```
[Sysname] context cnt3
[Sysname-context-4-cnt3] description context-3
```

#配置 Context cnt3 的 CPU 权重为 2。

```
[Sysname-context-4-cnt3] limit-resource cpu weight 2
```

启动 Context cnt3。

```
[Sysname-context-4-cnt3] context start
It will take some time to start the context...
Context started successfully.
```

将接口 GigabitEthernet1/0/3 和 GigabitEthernet1/0/13 分配给 Context cnt3。

```
[Sysname-context-4-cnt3] allocate interface gigabitethernet 1/0/3 gigabitethernet 1/0/13
[Sysname-context-4-cnt3] quit
```

完成上述配置后，可使用 **switchto context** 命令登录到指定的 Context，进行业务相关的配置。