

# 基准流程最佳实践白皮书

## 目录

[简介](#)

[基线](#)

[什么是基线？](#)

[为什么要使用基线？](#)

[基准目标](#)

[核心基线流程图](#)

[基准程序](#)

[第1步：编制硬件、软件和配置清单](#)

[第2步：验证路由器是否支持SNMP MIB](#)

[第3步：轮询并记录来自路由器的特定SNMP MIB对象](#)

[第4步：分析数据以确定阈值](#)

[第5步：修复发现的即时问题](#)

[第6步：测试阈值监控](#)

[第7步：使用SNMP或RMON实施阈值监控](#)

[其它 MIB](#)

[路由器 MIB](#)

[Catalyst 交换机 MIB](#)

[串行链路 MIB](#)

[RMON 报警和事件配置命令](#)

[警报](#)

[事件](#)

[RMON 警报和事件实现](#)

[相关信息](#)

## 简介

本文档介绍高可用性网络的基线建立概念和过程。它包括网络基线化和阈值化的关键成功因素，可帮助评估成功。此外，本文档还提供了根据思科高可用性服务(HAS)团队确定的最佳实践指南实施的基准和阈值流程的重要详细信息。

本文档将逐步指导您完成基线建立过程。一些当前的网络管理系统(NMS)产品可以帮助自动化此过程，但是，无论您使用自动化工具还是手动工具，基线建立过程都保持不变。如果使用这些NMS产品，则必须调整适用于您的唯一网络环境的默认阈值设置。必须有一个智能选择这些阈值的流程，以便它们有意义且正确。

## 基线

[什么是基线？](#)

基线是一个过程，用于定期研究网络以确保网络的工作情况符合设计意图。它远非记录特定时间点的网络健康状态的一个报告那么简单。通过执行基线过程，您可以获得以下信息：

- 获取有关硬件和软件运行状况的宝贵信息
- 确定网络资源的当前利用率
- 做出有关网络警报阈值的准确决策
- 确定当前的网络问题
- 预测未来问题

下图说明了查看基线的另一种方法。

红线，即网络中断点，是网络中断的点，由硬件和软件工作原理的知识决定。绿线，即网络负载，是指随着新应用的添加和其他此类因素而出现的网络中负载的自然变化。

基线的用途是确定：

- 您的网络处于绿线位置
- 网络负载增长速度
- 希望预测两者何时会交叉

通过定期执行基线，您可以了解当前状态并推断故障何时发生并提前进行准备。这还有助于您就网络升级的预算资金何时、何地以及如何使用做出更明智的决策。

## 为什么要使用基线？

基线过程可帮助您确定网络中的关键资源限制问题并正确规划。这些问题可描述为控制平面资源或数据平面资源。控制平面资源对于设备内的特定平台和模块是唯一的，并且可能会受到许多问题的影响，包括：

- 数据利用率
- 启用的功能
- 网络设计

控制平面资源包括以下参数：

- CPU 利用率
- 内存利用率
- 缓冲区利用率

数据平面资源仅受流量的类型和数量影响，包括链路利用率和背板利用率。通过为关键区域的资源利用率设定基准，可以避免严重的性能问题，甚至避免网络崩溃。

随着延迟敏感型应用（如语音和视频）的引入，基线现在变得比以往任何时候都重要。传统的传输控制协议/互联网协议(TCP/IP)应用程序可以容忍一定的延迟。语音和视频基于用户数据报协议(UDP)，不允许重新传输或网络拥塞。

由于应用的新组合，基线功能可帮助您了解控制平面和数据平面资源利用率问题，并主动规划更改和升级，以确保持续成功。

数据网络已经存在多年。直到不久以前，保持网络运行一直是一个相当宽宏大量的过程，其中还留有犯错的余地。随着IP语音(VoIP)等延迟敏感型应用越来越被广泛接受，运行网络的工作变得越来越困难，对精度的要求也越来越高。为了更加准确并为网络管理员提供坚实的网络管理基础，了解网络的运行方式非常重要。为此，您必须完成一个称为基线的过程。

## 基准目标

基线的目标是：

1. 确定网络设备的当前状态
2. 将该状态与标准性能指南进行比较
3. 设置阈值，以便在状态超过这些准则时发出警报

由于存在大量数据以及分析数据所花费的时间，您必须首先限制基线的范围，以便更轻松地了解整个过程。从网络核心处着手，这是最符合逻辑、有时也是最有益的地方。网络的这一部分通常最小，需要最稳定的网络。

为简单起见，本文档说明如何对一个非常重要的简单网络管理协议管理信息库(SNMP MIB)进行基线调整：cpmCPUTotal5min。cpmCPUTotal5min是思科路由器的中央处理器(CPU)的五分钟衰减平均值，并且是控制平面性能指标。基线将在Cisco 7000系列路由器上执行。

了解该过程后，即可将其应用于大量SNMP数据库中可用于大多数思科设备的任何数据，例如：

- 综合业务数字网络(ISDN)使用情况
- 异步传输模式(ATM)信元丢失
- 可用系统内存

## 核心基线流程图

以下流程图显示了核心基准流程的基本步骤。尽管可以使用产品和工具执行其中的某些步骤，但它们在灵活性和易用性方面往往存在差距。即使您计划使用网络管理系统(NMS)工具执行基线建立，这仍是学习该过程和了解网络实际工作原理的好练习。此过程还可以消除一些NMS工具工作方式中的某些神秘之处，因为大多数工具基本上都执行相同的工作。

## 基准程序

第1步：编制硬件、软件和配置清单

编制硬件、软件和配置的清单非常重要，原因有几个。首先，在某些情况下，Cisco SNMP MIB特定于您运行的Cisco IOS版本。有些MIB对象会被新对象替换或有时会被完全删除。收集数据后硬件资产是最重要的，因为在初始基线后需要设置的阈值通常基于Cisco设备的CPU类型、内存量等。配置清单对于确保您了解当前配置也很重要：您可能需要在基线之后更改设备配置，以调整缓冲区等。

这部分是Cisco网络基线的最有效方法是使用CiscoWorks2000 Resource Manager Essentials (Essentials)。如果网络中正确安装了此软件，则Essentials数据库中应包含所有设备的当前清单。您只需查看资产清单，看看是否存在任何问题。

下表是从Essentials导出并在Microsoft Excel中编辑的Cisco路由器类软件清单报告的示例。从此清单中，请注意，您必须使用在12.0x和12.1x Cisco IOS版本中找到的SNMP MIB数据和对象标识符(OID)。

设备名	路由器类型	version	软件版本
field-2500a.embu-mlab.cisco.com	Cisco 2511	M	12.1(1)
qdm-7200.embu-mlab.cisco.com	Cisco 7204	B	12.1(1)E
voip-3640.embu-mlab.cisco.com	Cisco 3640	0x00	12.0(3c)
wan-1700a.embu-mlab.cisco.com	Cisco 1720	0x101	12.1(4)
wan-2500a.embu-mlab.cisco.com	Cisco 2514	L	12.0(1)
wan-3600a.embu-mlab.cisco.com	Cisco 3640	0x00	12.1(3)
wan-7200a.embu-mlab.cisco.com	Cisco 7204	B	12.1(1)E
172.16.71.80	Cisco 7204	B	12.0(5T)

如果Essentials未安装在网络中，可以使用UNIX工作站上的UNIX命令行工具snmpwalk查找IOS版本。如下例所示。如果您不确定此命令如何工作，请在UNIX提示符处键入man snmpwalk以了解详细信息。当您开始选择基线的MIB OID时，IOS版本非常重要，因为MIB对象取决于IOS。另请注意，通过了解路由器类型，您稍后可以确定应该为CPU、缓冲区等使用什么阈值。

```
nsahpov6% snmpwalk -v1 -c private 172.16.71.80 system
system.sysDescr.0 : DISPLAY STRING- (ascii): Cisco Internetwork Operating System Software
IOS (tm) 7200 Software (C7200-JS-M), Version 12.0(5)T, RELEASE SOFTWARE (fc1)
Copyright (c) 1986-2001 by cisco Systems, Inc.
Compiled Fri 23-Jul-2001 23:02 by kpma
system.sysObjectID.0 : OBJECT IDENTIFIER:
```

## 第2步：验证路由器是否支持SNMP MIB

现在，您已经拥有了想要轮询基线的设备的资产，可以开始选择要轮询的特定OID。如果您提前确认您想要的的数据确实存在，这样可以省去很多麻烦。cpmCPUTotal5min MIB对象在CISCO-PROCESS-MIB中。

要查找要轮询的OID，您需要思科CCO网站上提供的转换表。要从Web浏览器访问此网站，请转到[Cisco MIBs页](#)，然后单击OIDs链接。

要从FTP服务器访问此网站，请键入ftp://ftp.cisco.com/pub/mibs/oid/。您可以从此站点下载已解码并按OID编号排序的特定MIB。

以下示例摘自CISCO-PROCESS-MIB.oid表。本示例显示cpmCPUTotal5min MIB的OID为。1.3.6.1.4.1.9.9.109.1.1.1.1.5。

注意：不要忘记将“。”添加到OID的开头，否则尝试轮询时会出错。您还需要在OID的末尾添加“。1”以实例化它。这会告知设备您正在查找的OID的实例。在某些情况下，OID具有多个特定类型数据的实例，例如当路由器有多个CPU时。

<#root>

```
ftp://ftp.cisco.com/pub/mibs/oid/CISCO-PROCESS-MIB.oid
### THIS FILE WAS GENERATED BY MIB2SCHEMA
"org" "1.3"
"dod" "1.3.6"
"internet" "1.3.6.1"
"directory" "1.3.6.1.1"
"mgmt" "1.3.6.1.2"
"experimental" "1.3.6.1.3"
"private" "1.3.6.1.4"
"enterprises" "1.3.6.1.4.1"
"cisco" "1.3.6.1.4.1.9"
"ciscoMgmt" "1.3.6.1.4.1.9.9"
"ciscoProcessMIB" "1.3.6.1.4.1.9.9.109"
"ciscoProcessMIBObjects" "1.3.6.1.4.1.9.9.109.1"
"ciscoProcessMIBNotifications" "1.3.6.1.4.1.9.9.109.2"
"ciscoProcessMIBConformance" "1.3.6.1.4.1.9.9.109.3"
"cpmCPU" "1.3.6.1.4.1.9.9.109.1.1"
"cpmProcess" "1.3.6.1.4.1.9.9.109.1.2"
"cpmCPUTotalTable" "1.3.6.1.4.1.9.9.109.1.1.1"
"cpmCPUTotalEntry" "1.3.6.1.4.1.9.9.109.1.1.1.1"
"cpmCPUTotalIndex" "1.3.6.1.4.1.9.9.109.1.1.1.1.1"
"cpmCPUTotalPhysicalIndex" "1.3.6.1.4.1.9.9.109.1.1.1.1.2"
"cpmCPUTotal5sec" "1.3.6.1.4.1.9.9.109.1.1.1.1.3"
"cpmCPUTotal1min" "1.3.6.1.4.1.9.9.109.1.1.1.1.4"
"cpmCPUTotal5min" "1.3.6.1.4.1.9.9.109.1.1.1.1.5"
```

有两种常用方法可以轮询MIB OID以确保其可用且正常运行。最好在开始批量数据收集之前执行此

操作，以免浪费时间轮询某些不存在的内容，导致数据库为空。一种方法是使用来自NMS平台(例如HP OpenView Network Node Manager (NNM)或CiscoWorks Windows)的MIB步行器，并输入要检查的OID。

以下是HP OpenView SNMP MIB walker的一个示例。

轮询MIB OID的另一种简单方法是使用UNIX命令snmpwalk，如下例所示。

```
nsahpov6% cd /opt/OV/bin
nsahpov6% snmpwalk -v1 -c private 172.16.71.80 .1.3.6.1.4.1.9.9.109.1.1.1.1.5.1
cisco.ciscoMgmt.ciscoProcessMIB.ciscoProcessMIBObjects.cpmCPU.cpmCPUTotalTable.cpmCPUTotalEntry.cpmCPU
```

在这两个示例中，MIB都返回了一个值0，这意味着在该轮询周期中，CPU平均利用率为0%。如果您很难让设备使用正确的数据做出响应，请尝试对设备执行ping操作并通过Telnet访问设备。如果仍有问题，请检查SNMP配置和SNMP社区字符串。您可能需要找到备用MIB或IOS的其他版本才能使之正常工作。

### 第3步：轮询并记录来自路由器的特定SNMP MIB对象

轮询MIB对象和记录输出有多种方法。提供现成产品、共享软件产品、脚本和供应商工具。所有前端工具都使用SNMP get 进程获取信息。主要区别在于配置的灵活性以及在数据库中记录数据的方式。再次查看处理器MIB，了解这些不同方法的工作原理。

既然您知道路由器支持OID，您就需要决定轮询它的频率以及如何记录。Cisco建议每隔5分钟轮询CPU MIB。更低的间隔会增加网络或设备上的负载，并且由于MIB值是五分钟平均值，因此轮询它比平均值更频繁没有帮助。通常也建议基线轮询至少有两个星期，以便您能够分析网络上至少两个星期的业务周期。

以下屏幕显示如何使用HP OpenView网络节点管理器6.1版添加MIB对象。从主屏幕中选择Options > Data Collection & Thresholds。

然后选择Edit > Add > MIB Objects。

从菜单中，添加OID字符串，然后单击Apply。现在，您已经将MIB对象输入到HP OpenView平台，以便可以轮询它。

接下来，您必须让HP OpenView知道要为该OID轮询的路由器。

从Data Collection菜单中，选择Edit > Add > MIB Collections。

在Source字段中，输入要轮询的路由器的域名系统(DNS)名称或IP地址。

从“Set Collection Mode”列表中选择Store， No Thresholds。

将轮询间隔设置为5m，持续5分钟间隔。

单击 Apply。

您必须选择文件>保存，更改才会生效。

要验证收集设置是否正确，请突出显示路由器的收集汇总行，然后选择操作>测试SNMP。这将检查社区字符串是否正确并将轮询所有OID实例。

单击Close，让集合运行一周。在每周周期结束时，提取数据进行分析。

如果将数据转储到ASCII文件并将其导入电子表格工具（例如Microsoft Excel），则更容易分析数据。要对HP OpenView NNM执行此操作，您可以使用命令行工具snmpColDump。配置的每个集合都会写入/var/opt/OV/share/databases/snmpCollect/目录中的文件。

使用以下命令将数据解压缩到名为testfile的ASCII文件：

```
<#root>
```

```
snmpColDump /var/opt/OV/share/databases/snmpCollect/cpmCPUTotal5min.1 >
```

```
testfile
```

注意：cpmCPUTotal5min.1是HP OpenView NNM在OID轮询开始时创建的数据库文件。

生成的测试文件类似于以下示例。

```
03/01/2001 14:09:10 nsa-gw.cisco.com 1  
03/01/2001 14:14:10 nsa-gw.cisco.com 1  
03/01/2001 14:19:10 nsa-gw.cisco.com 1  
03/01/2001 14:24:10 nsa-gw.cisco.com 1  
03/01/2001 14:29:10 nsa-gw.cisco.com 1  
03/01/2001 14:34:10 nsa-gw.cisco.com 1  
03/01/2001 14:39:10 nsa-gw.cisco.com 1  
03/01/2001 14:44:10 nsa-gw.cisco.com 1  
03/01/2001 14:49:10 nsa-gw.cisco.com 1  
03/01/2001 14:54:10 nsa-gw.cisco.com 1  
03/01/2001 14:59:10 nsa-gw.cisco.com 1  
03/.....
```

一旦测试文件输出位于UNIX工作站上，您就可以使用文件传输协议(FTP)将其传输到PC上。

您还可以使用自己的脚本收集数据。为此，请每五分钟对CPU OID执行snmpget，并将结果转储到.csv文件中。

#### 第4步：分析数据以确定阈值

现在您拥有了一些数据，可以开始分析它。基准的此阶段确定可用于准确衡量性能或故障的阈值设置，并且在启用阈值监控时不会设置太多警报。最简单的方法之一就是数据导入电子表格（如Microsoft Excel）并绘制散点图。通过此方法，您可以非常轻松地查看特定设备在监控特定阈值的

情况下会创建异常警报的次数。不执行基线操作而打开阈值是不明智的，因为这可能从超过所选阈值的设备创建警报风暴。

要将测试文件导入Excel电子表格，请打开Excel并选择文件>打开，然后选择数据文件。

然后，Excel应用程序会提示您导入文件。

完成后，导入的文件应类似于以下屏幕。

使用散点图可以更轻松地查看各种阈值设置对网络的影响。

要创建散点图，请突出显示导入的文件中的C列，然后单击Chart Wizard图标。然后，按照图表向导中的步骤创建散点图。

在“图表向导”第1步中，如下所示，选择标准类型选项卡，然后选择XY（散点图）图表类型。然后，单击下一步。

在图表向导第2步中，选择数据范围选项卡，然后选择数据范围和列选项，如下所示。单击 Next。

在图表向导第3步中，输入图表标题以及X轴和Y轴值，然后单击下一步。

在“图表向导”步骤4中，选择是希望散点图位于新页面上还是作为现有页面中的对象。

单击完成，将图表放置在所需的位置。

“如果呢？”分析

您现在可以使用散点图进行分析。但是，在继续之前，您需要询问以下问题：

- 供应商（在本例中供应商是思科）建议将什么作为此MIB变量的阈值？

一般情况下，思科建议核心路由器的平均CPU使用率不超过60%。之所以选择60%，是因为路由器需要一些开销，以防遇到故障或网络发生故障。思科估计，如果路由协议必须重新计算或重新收敛，核心路由器大约需要40%的CPU开销。这些百分比取决于您使用的协议，以及网络的拓扑和稳定性。

- 如果使用60%作为阈值设置会怎样？

如果您在散点图水平方向绘制60的线条，您会看到任何数据点都不会超过60%的CPU使用率。因此，在网络管理系统(NMS)站点上设置的阈值60不会在轮询期间触发阈值警报。此路由器可以接受百分比60。但是，请注意散点图中的某些数据点接近60。如果知道路由器何时接近60%的阈值，您就可以提前知道CPU接近60%，并规划在达到该阈值时该怎么做，这会很有帮助。

- 如果我将阈值设置为50%会怎样？

估计此路由器在此轮询周期中四次达到50%的利用率，并且每次都会生成阈值警报。当您查看路由器组以了解不同的阈值设置会执行什么操作时，此过程变得更为重要。例如，“如果我将整个核心网络的阈值设置为50%会怎样？”要知道，只选择一个数字是非常困难的。



## CPU阈值“假设”分析

您可用于简化这一操作的一种策略是Ready、Set、Go阈值方法。此方法连续使用三个阈值数字。

- 就绪-您设置的阈值，可以预测哪些设备将来可能需要关注
- Set -用作早期指示器的阈值，可提醒您开始计划修复、重新配置或升级
- Go -您和/或供应商认为是故障情况的阈值，需要采取某些操作进行修复；在本例中为60%

下表显示“就绪”、“设置”、“开始”策略的策略。

阈值	操作	结果
45%	进一步调查	行动计划选项列表
50%	制定行动计划	行动计划步骤列表
60%	实施行动计划	路由器不再超过阈值。返回就绪模式

“就绪”、“设置”、“开始”方法会更改前面讨论的原始基线图表。下图显示更改后的基线图表。如果可以确定图表上的其他交点，您现在有比以前更多的时间进行计划和反应。

请注意，在此过程中，注意重点是网络中的例外情况，而不是其他设备。我们假定，只要设备低于阈值，它们就会正常工作。

如果您从头开始考虑这些步骤，您将为保持网络正常运行做好充分准备。执行此类计划对于预算计划也非常有用。如果您知道您的前五名go路由器、中间集路由器和底部ready路由器是什么，您可以根据路由器的种类和您的行动计划选项轻松规划升级所需的预算。相同的策略可用于广域网(WAN)链路或任何其他MIB OID。

### 第5步：修复发现的即时问题

这是基线过程中较简单的部分之一。一旦您确定了哪些设备超出了go阈值，您就应当制定一个行动计划，使这些设备重新回到阈值之下。

您可以向思科技术支持中心(TAC)提交支持请求，或者联系您的系统工程师了解可用选项。你不应该想当然地认为，让事情回到正常水平需要花钱。某些CPU问题可以通过更改配置来解决，以确保所有进程都以最有效的方式运行。例如，某些访问控制列表(ACL)可能会使路由器CPU的运行率非常高，这是因为数据包通过路由器所经过的路径。在某些情况下，您可以实施NetFlow交换来更改数据包交换路径并减少ACL对CPU的影响。无论存在什么问题，在此步骤中都必须将所有路由器恢复至阈值之下，这样您就可以稍后实施阈值，而不会有使用过多阈值警报泛洪NMS站点的风险。

### 第6步：测试阈值监控

此步骤涉及使用将在生产网络中使用的工具在实验室中测试阈值。监控阈值有两种常用方法。您必须确定哪种方法最适合您的网络。

- 使用SNMP平台或其他SNMP监控工具进行轮询和比较的方法

此方法使用更多的网络带宽来轮询流量，并占用您的SNMP平台上的处理周期。

- 在路由器中使用远程监控(RMON)警报和事件配置，因此只有在超过阈值时才发送警报

此方法可以降低网络带宽使用量，但也会增加路由器的内存和CPU利用率。

## 使用SNMP实施阈值

要使用HP OpenView NNM设置SNMP方法，请像设置初始轮询时那样选择Options > Data Collection & Thresholds。不过，这次应在“收藏”菜单中选择存储、检查阈值而不是“存储、无阈值”。设置阈值后，可以通过发送多次ping和/或多次SNMP遍历来提高路由器上的CPU利用率。如果无法强制CPU足够高以触发阈值，则可能必须降低阈值。无论如何，您应确保阈值机制正常工作。

使用此方法的局限性之一是无法同时实施多个阈值。您需要三个SNMP平台来设置三个不同的同时阈值。[Concord Network Health](#)和[Trinagy TREND](#)等工具可以为同一OID实例提供多个阈值。

如果您的系统一次只能处理一个阈值，则可以考虑串行方式的“就绪”、“设置”、“开始”策略。也就是说，当持续达到ready阈值时，即开始进行调查，并将阈值提升到该设备的设置级别。持续达到设置级别后，开始制定行动计划，并将该设备的阈值提高到go级别。然后，当持续达到转至阈值时，实施您的行动计划。这应该与三种同时阈值方法一样有效。更改SNMP平台阈值设置只需要多花一点时间。

## 使用RMON警报和事件实施阈值

使用RMON警报和事件配置，您可以让路由器监控自身是否存在多个阈值。当路由器检测到超过阈值情况时，它会向SNMP平台发送SNMP陷阱。必须在路由器配置中设置SNMP陷阱接收器，才能转发陷阱。警报和事件之间存在关联。警报会检查给定阈值的OID。如果达到阈值，警报进程将触发事件进程，该进程可以发送SNMP陷阱消息、创建RMON日志条目或同时发送两者。有关此命令的详细信息，请参阅[RMON警报和事件配置命令](#)。

以下路由器配置命令使路由器每300秒监控一次cpmCPUTotal5min。如果CPU使用率超过60%，它将触发事件1；如果CPU使用率回落到40%，它将触发事件2。在这两种情况下，SNMP陷阱消息都将使用社区专用字符串发送到NMS工作站。

要使用Ready、Set、Go方法，请使用以下所有配置语句。

```
rmon event 1 trap private description "cpu hit60%" owner jharp
rmon event 2 trap private description "cpu recovered" owner jharp
rmon alarm 10 cpmCPUTotalTable.1.5.1 300 absolute rising 60 1 falling 40 2 owner jharp
```

```
rmon event 3 trap private description "cpu hit50%" owner jharp
rmon event 4 trap private description "cpu recovered" owner jharp
rmon alarm 20 cpmCPUTotalTable.1.5.1 300 absolute rising 50 3 falling 40 4 owner jharp
```

```
rmon event 5 trap private description "cpu hit 45%" owner jharp
rmon event 6 trap private description "cpu recovered" owner jharp
rmon alarm 30 cpmCPUTotalTable.1.5.1 300 absolute rising 45 5 falling 40 6 owner jharp
```

以下示例显示以上语句配置的show rmon alarm命令的输出。

```
<#root>
zack#
sh rmon alarm
Alarm 10 is active, owned by jharp
  Monitors cpmCPUTotalTable.1.5.1 every 300 second(s)
  Taking absolute samples, last value was 0
  Rising threshold is 60, assigned to event
1
  Falling threshold is 40, assigned to event
2
  On startup enable rising or falling alarm
Alarm 20 is active, owned by jharp
  Monitors cpmCPUTotalTable.1.5.1 every 300 second(s)
  Taking absolute samples, last value was 0
  Rising threshold is 50, assigned to event
3
  Falling threshold is 40, assigned to event
4
  On startup enable rising or falling alarm
Alarm 30 is active, owned by jharp
  Monitors cpmCPUTotalTable.1.5.1 every 300 second(s)
  Taking absolute samples, last value was 0
  Rising threshold is 45, assigned to event
5
  Falling threshold is 40, assigned to event
6
  On startup enable rising or falling alarm
```

以下示例展示show rmon event命令的输出。

```
<#root>
zack#
sh rmon event
Event 1 is active, owned by jharp
  Description is cpu hit60%
  Event firing causes trap to community
private, last fired 00:00:00
Event 2 is active, owned by jharp
  Description is cpu recovered
  Event firing causes trap to community
private, last fired 02:40:29
Event 3 is active, owned by jharp
  Description is cpu hit50%
  Event firing causes trap to community
private, last fired 00:00:00
Event 4 is active, owned by jharp
  Description is cpu recovered
  Event firing causes trap to community
private, last fired 00:00:00
```

```

Event 5 is active, owned by jharp
Description is cpu hit 45%
Event firing causes trap to community
private, last fired 00:00:00
Event 6 is active, owned by jharp
Description is cpu recovered
Event firing causes trap to community
private, last fired 02:45:47

```

您可能需要尝试这两种方法，以查看哪种方法最适合您的环境。您甚至可能发现结合使用多种方法效果不错。无论如何，都应该在实验室环境中进行测试，以确保一切正常。在实验室中测试后，您只需在一小组路由器上执行有限部署，即可测试向操作中心发送警报的过程。

在这种情况下，您必须降低阈值以测试进程：不建议尝试人为提高生产路由器上的CPU。您还应该确保当警报进入运营中心的NMS工作站时，有一个上报策略，以确保在设备超过阈值时通知您。这些配置已在实验室中使用Cisco IOS 12.1(7)版进行测试。如果您遇到任何问题，应咨询思科工程或系统工程师，以查看您的IOS版本中是否有Bug。

### 第7步：使用SNMP或RMON实施阈值监控

在实验室和有限部署中全面测试阈值监控之后，您就可以跨核心网络实施阈值了。现在，您可以系统地完成此基线过程，了解网络上的其他重要MIB变量，例如缓冲区、可用内存、循环冗余校验(CRC)错误、AMT信元丢失等。

如果使用RMON警报和事件配置，则现在可以从NMS工作站停止轮询。这将减少NMS服务器上的负载并减少网络上的轮询数据量。通过系统地完成重要网络运行状况指示器的此过程，可以轻松达到网络设备使用RMON警报和事件监控自己的程度。

## 其它 MIB

学习此过程后，您可能需要调查其他MIB以做基线和监控。以下小节提供了一些可能会对您有用的OID和说明的简要列表。

### 路由器 MIB

内存特征对于确定路由器的运行状况非常有用。正常运行的路由器几乎总是应该具有可用的缓冲区空间。如果路由器开始耗尽缓冲空间，CPU将不得不更加努力地创建新的缓冲区，并尝试为传入和传出数据包找到缓冲区。有关缓冲区的深入讨论不在本文档的讨论范围之内。但是，一般情况下，正常的路由器应该很少出现缓冲区未命中情况（如果有），并且不应该出现任何缓冲区故障或内存为零的情况。

对象	描述	OID
ciscoMemoryPoolFree	受管设备	1.3.6.1.4.1.9.9.48.1.1.1.6

	上当前未使用的内存池中的字节数	
ciscoMemoryPoolLargestFree	内存池中当前未使用的最大连续字节数	1.3.6.1.4.1.9.9.48.1.1.1.7
bufferElMiss	缓冲区元素未命中数	1.3.6.1.4.1.9.2.1.12
bufferFail	缓冲区分配	1.3.6.1.4.1.9.2.1.46

	失败数	
bufferNoMem	由于无可用内存而导致的缓冲区创建失败数	1.3.6.1.4.1.9.2.1.47

### Catalyst 交换机 MIB

对象	描述	OID
cpmCPUTotal5min	过去五分钟内的整体CPU繁忙百分比。此对象从OLD-CISCO-SYSTEM-MIB中删除avgBusy5对象	1.3.6.1.4.1.9.9.109.1.1.1.5
cpmCPUTotal5sec	过去五秒内总CPU繁忙百分比。此对象淘汰来自OLD-CISCO-SYSTEM-MIB的busyPer对	1.3.6.1.4.1.9.9.109.1.1.1.3

	象	
sysTraffic	上一个轮询间隔的带宽利用率百分比	1.3.6.1.4.1.9.5.1.1.8
sysTrafficPeak	自上次清除端口计数器或系统启动以来的峰值流量计值	1.3.6.1.4.1.9.5.1.1.19
sysTrafficPeaktime	自峰值流量计数值出现以来的时间（以百分之一秒为单位）	1.3.6.1.4.1.9.5.1.1.20
portTopNUtilization	系统中端口的利用率	1.3.6.1.4.1.9.5.1.20.2.1.4
portTopNBufferOverFlow	系统中端口的缓冲区溢出数量	1.3.6.1.4.1.9.5.1.20.2.1.10

## 串行链路 MIB

对象	描述	OID
locIfInputQueueDrops	由于输入队列已满而丢弃的数据包数	1.3.6.1.4.1.9.2.2.1.1.26
locIfOutputQueueDrops	由于输出队列已满而丢弃的数据包数	1.3.6.1.4.1.9.2.2.1.1.27
locIfInCRC	出现循环冗余校验和错误的输入数据包的	1.3.6.1.4.1.9.2.2.1.1.12

	数量	
--	----	--

## RMON 报警和事件配置命令

### 警报

可以使用以下语法配置RMON警报：

<#root>

```
rmon alarm number variable interval {delta | absolute} rising-threshold value
[event-number] falling-threshold value [event-number]
[owner string]
```

元素	描述
号码	警报编号，与RMON MIB中alarmTable中的alarmIndex相同。
变量	要监控的MIB对象，转换为RMON MIB的alarmTable中使用的alarmVariable。
间隔	警报监控MIB变量的时间（以秒为单位），与RMON MIB的alarmTable中使用的alarmInterval相同。
增量	测试MIB变量之间的更改，这会影响RMON MIB的alarmTable中的alarmSampleType。
absolute	直接测试每个MIB变量，这会影响RMON MIB的alarmTable中的alarmSampleType。
上升阈值	触发警报的值。
事件编号	（可选）当上升或下降阈值超出其限制时要触发的事件编号。此值与RMON MIB的alarmTable中的alarmRisingEventIndex或alarmFallingEventIndex相同。
下降阈值	警报重置时的值。
所有者字符串	（可选）指定警报的所有者，该所有者与RMON MIB的alarmTable中的alarmOwner相同。

### 事件

RMON事件可以使用以下语法进行配置：

<#root>



```
rmon event number [log] [trap community] [description string]
           [owner string]
```

元素	描述
号码	分配的事件编号，与RMON MIB中eventTable中的eventIndex相同。
日志	(可选) 在触发事件时生成RMON日志条目，并将RMON MIB中的eventType设置为log或log-and-trap。
陷阱社区	(可选) 用于此陷阱的SNMP社区字符串。将此行的RMON MIB中eventType的设置配置为snmp-trap或log-and-trap。此值与RMON MIB中eventTable中的eventCommunityValue相同。
描述字符串	(可选) 指定事件的说明，该说明与RMON MIB的eventTable中的事件说明相同。
所有者字符串	(可选) 此事件的所有者，与RMON MIB的eventTable中的eventOwner相同。

## RMON 警报和事件实现

有关RMON报警和事件实施的详细信息，请阅读网络管理系统最佳实践白皮书的[RMON报警和事件实施](#)部分。

## 相关信息

- [技术支持和文档- 思科系统](#)

## 关于此翻译

思科采用人工翻译与机器翻译相结合的方式将此文档翻译成不同语言，希望全球的用户都能通过各自的语言得到支持性的内容。

请注意：即使是最好的机器翻译，其准确度也不及专业翻译人员的水平。

Cisco Systems, Inc. 对于翻译的准确性不承担任何责任，并建议您总是参考英文原始文档（已提供链接）。