排除无线局域网控制器CPU负载故障

目录

简介

本文档介绍如何监控Catalyst 9800无线LAN控制器上的CPU使用情况，另外还介绍了几个配置建议。

了解CPU使用情况

在深入了解CPU负载故障排除之前，您需要了解Catalyst 9800无线LAN控制器中如何使用CPU的基础知识，以及有关软件架构的一些详细信息。

通常，Catalyst 9800最佳实践文档定义一组可以防止应用程序级问题的良好配置设置，例如对mDNS使用位置过滤或确保始终启用客户端排除。建议您将这些建议与此处显示的主题一起应用。

平台基础知识

Catalyst 9800控制器设计为灵活的平台，可针对不同的网络负载，专注于水平扩展。内部开发命名为“eWLC”，e表示“弹性”，表示同一软件架构能够从小型单CPU嵌入式系统运行到多个CPU/核心大型设备。

每个WLC都有两个不同的“端”：

- 控制平面：处理所有“管理”交互，如CLI、UI、Netconf，以及客户端和AP的所有自注册过程。
- 数据平面：负责实际数据包转发和CAPWAP解封、AVC策略实施等功能。

控制层面

- 大多数Cisco IOS-XE进程在BinOS（Linux内核）下运行，具有自己的专用调度程序和监控命令。
- 有一组称为无线网络控制后台守护程序(WNCD)的关键进程，每个进程都有一个本地内存数据库，可以处理大多数无线活动。每个CPU拥有一个WNCD，用于将负载分散到每个系统的所有可用CPU核心
- 在AP加入期间完成WNCD间的负载分配。当AP执行CAPWAP加入控制器时，内部负载均衡器使用一组可能的规则分配AP，以确保正确使用所有可用的CPU资源。
- Cisco IOS®代码运行在它自己的名为IOSd的进程上，并且有CPU调度程序。这涉及到特定功能，例如CLI、SNMP、组播和路由。

在简化视图中，控制器具有控制和数据平面之间的通信机制，即“传送”，将流量从网络发送到控制平面，而“注入”，将帧从控制平面推送到网络。

作为可能导致CPU使用率较高的故障排除调查的一部分，您需要监控传送机制，以评估哪些流量将到达控制平面并可能导致高负载。

数据层面

对于Catalyst 9800控制器，这是作为思科数据包处理器(CPP)的一部分运行的，CPP是用于开发数据包转发引擎的软件框架，用于多种产品和技术。

该架构支持跨不同硬件或软件实施的通用功能集，例如，允许9800CL与9800-40在不同吞吐量级别使用类似的功能。

AP负载均衡

在CAPWAP AP加入过程中，WLC在CPU之间执行负载均衡，主要区别在于AP站点标记名称。其理念是每个AP代表一个特定CPU负载，来自其客户端活动和AP本身。有多种机制来执行这种平衡：

- 如果AP使用“default-tag”，则会在所有CPU/WNCD之间以循环方式对其进行平衡，每个新的AP加入将转至下一个WNCD。这是最简单的方法，但意义不大：
  - 这是次优方案，因为同一RF漫游域中的AP将频繁进行WNCD间漫游，包括额外的进程间通信。实例间的漫游速度较慢，但所占百分比较小。
  - 对于FlexConnect（远程）站点标签，无可用的PMK密钥分发。这意味着您无法对Flex模式进行快速漫游，影响OKC/FT漫游模式。

    通常，默认标记可用于低负载方案（例如，低于9800平台的AP和客户端负载的40%），并且仅在不需要快速漫游时用于FlexConnect部署。
- 如果AP具有自定义站点标记，则属于站点标记名称的AP首次加入控制器时，该站点标记将分

配给特定WNCD实例。具有相同标记的所有后续附加AP加入将分配给同一WNCD。这可确保在同一个站点标记中的AP之间漫游，这在一个WCND上下文中发生，从而提供更优的流量，同时降低CPU使用率。支持跨WNCD漫游，只是不像WNCD内漫游那样理想。

- 默认负载均衡决策：将标签分配给WNCD时，负载均衡器会选择此时站点标签计数最低的实例。由于该站点标记可能具有的总负载未知，因此可能导致次优平衡方案。这取决于AP加入的顺序、已定义的站点标签数量以及它们之间的AP计数是否不对称

- 静态负载平衡：为防止向WNCD分配不平衡的站点标记，17.9.3及更高版本中引入了site load命令，以允许管理员预定义每个站点标记的预期负载。这在处理园区场景或多个分支机构（每个分支机构映射到不同的AP计数）时尤其有用，可确保负载在WNCD之间均匀分布。

例如，如果您使用一台9800-40处理一个总部，再加上5个分支机构，但具有不同的AP计数，则配置可能如下所示：

```
wireless tag site office-main
 load 120

wireless tag site branch-1
 load 10

wireless tag site branch-2
 load 12

wireless tag site branch-3
 load 45

wireless tag site branch-4
 load 80

wireless tag site branch-5
 load 5
```

在此场景中，您不希望主办公室标记与Branch-3和Branch-4位于同一WNCD上，总共存在6个站点标记，并且平台有5个WNCD，因此负载最高的站点标记可能会降落在相同的CPU上。通过使用load命令，您可以创建可预测AP负载均衡拓扑。

load命令是预期大小提示，不必精确匹配AP计数，但它通常设置为可能加入的预期AP。

- 在单个控制器处理大型建筑的情况下，仅为该特定平台创建与WNCD数量相同的站点标签会更容易、更简单（例如，C9800-40有5个，C9800-80有8个）。将同一区域或漫游域中的AP分配到同一站点标记，以最小化WNCD间通信。
- RF负载平衡：这将使用RRM中的RF邻居关系在WNCD实例之间平衡AP，并根据AP彼此之间的距离创建子组。必须在AP已启动并运行一段时间后执行，并且无需配置任何静态负载均衡设置。从17.12及更高版本开始提供。

如何确定存在多少个WNCD？

对于硬件平台，WNCD计数是固定的：9800-40有5,9800-80有8。对于9800CL（虚拟），WNCD的数量将取决于初始部署期间使用的虚拟机模板。

通常，如果要了解系统中运行的WNCD的数量，可以在所有控制器类型之间使用此命令：

<#root>

```
9800-40#show processes cpu platform sorted  |  count wncd
Number of lines which match regexp =

5
```

具体而言，对于9800-CL，您可以使用命令 **show platform software system all** 收集虚拟平台上的详细信息：

<#root>

```
9800cl-1#show platform software system all
Controller Details:
==================
VM Template: small
Throughput Profile: low
AP Scale: 1000
Client Scale: 10000

WNCD instances: 1
```

监控AP负载均衡

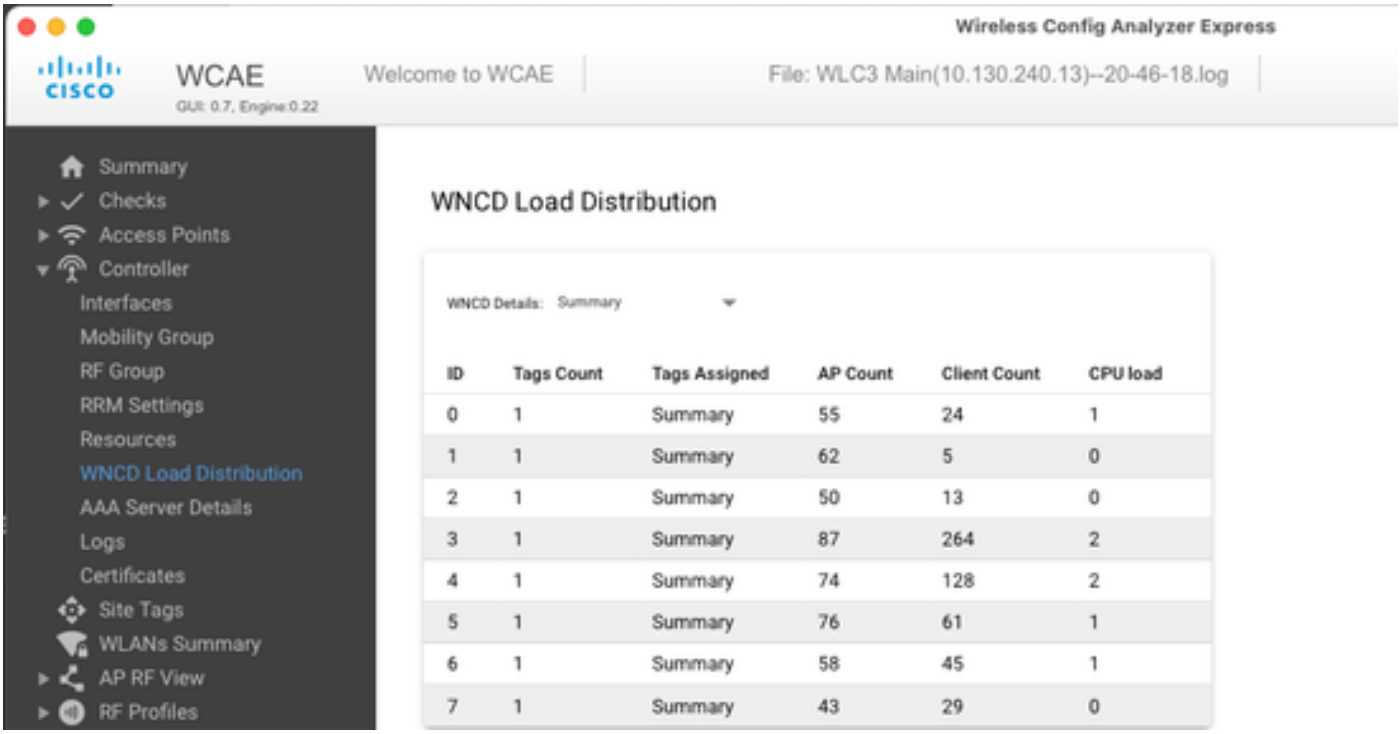
AP到WNCD分配在AP CAPWAP加入过程中应用，因此无论使用何种平衡方法，在操作过程中都不会发生更改，除非存在网络范围的CAPWAP重置事件，其中所有AP都会断开连接并重新加入。

CLI命令show wireless loadbalance tag affinity可用于轻松查看所有WNCD实例间AP负载均衡的当前状态：

```
98001#show wireless loadbalance tag affinity
Tag                     Tag type   No of AP's Joined   Load Config   Wncd Instance
-------------------------------------------------------------------------------------
Branch-tag              SITE TAG   10                  0             0
Main-tag                SITE TAG   200                 0             1
default-site-tag        SITE TAG   1                   NA            2
```

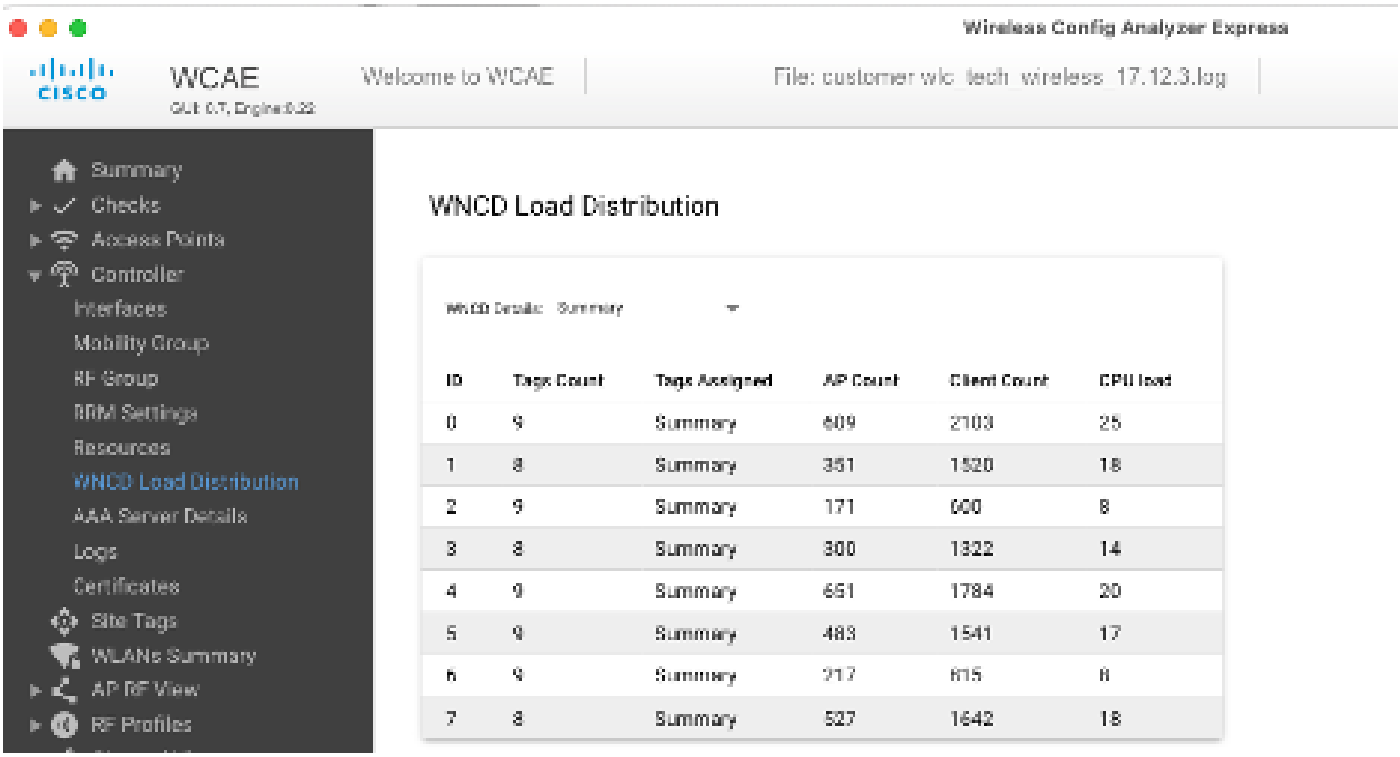
如果要将AP分布与客户端计数和CPU负载相关联，最简单的方法是使用WCAE支持工具并加载繁忙时间采取的show tech wireless。该工具汇总了来自与其关联的每个AP的WNCD客户端计数。

在使用率和客户端计数较低的情况下，适当平衡的控制器示例：



ID	Tags Count	Tags Assigned	AP Count	Client Count	CPU load
0	1	Summary	55	24	1
1	1	Summary	62	5	0
2	1	Summary	50	13	0
3	1	Summary	87	264	2
4	1	Summary	74	128	2
5	1	Summary	76	61	1
6	1	Summary	58	45	1
7	1	Summary	43	29	0

另一个示例（对于负载较高的控制器），显示正常CPU利用率：



ID	Tags Count	Tags Assigned	AP Count	Client Count	CPU load
0	9	Summary	609	2703	25
1	8	Summary	351	1520	18
2	9	Summary	171	600	8
3	8	Summary	300	1322	14
4	9	Summary	651	1784	20
5	9	Summary	483	1541	17
6	9	Summary	217	615	8
7	8	Summary	527	1642	18

推荐的AP负载均衡机制是什么？

简而言之，您可以总结中的不同选项：

- 小型网络，无需快速漫游，控制器负载不到40%：默认标记。
- 如果需要快速漫游(OKC、FT、CCKM)或大量客户端：

- 单个建筑：创建与CPU数量相同的站点标签（取决于平台）
- 在17.12版本之前或少于500个AP计数：多栋建筑、多个分支机构或大型园区：为每个物理RF位置创建一个站点标记，并为每个站点配置load命令。
- 超过500个AP的17.12及更高版本：使用RF负载均衡。

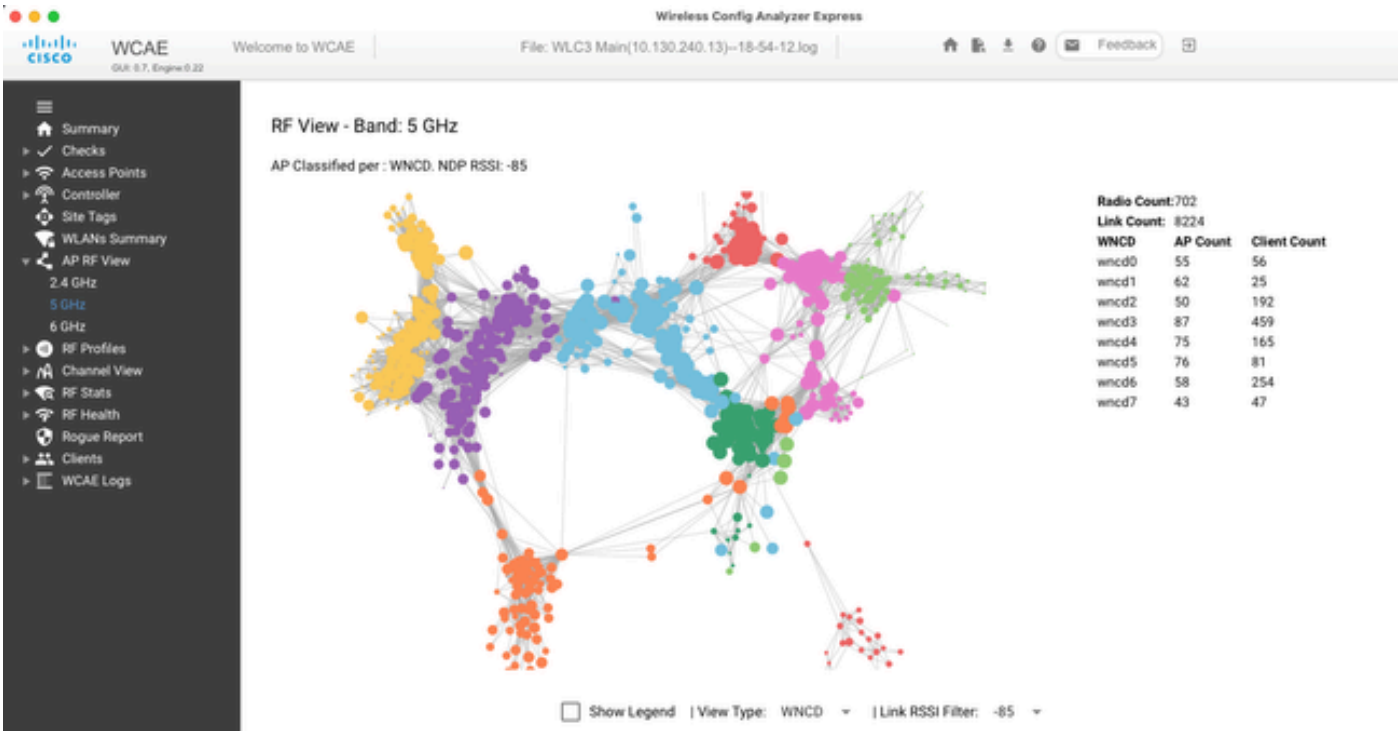
此500个AP阈值用于标记应用负载均衡机制是否有效，因为默认情况下它将AP分组为100个单元的数据块。

AP WNCD分布可视化

有些方案需要执行更高级的AP平衡，而且最好对AP如何在CPU间分布进行精细控制，例如，在非常高密度的方案中，主要负载度量是客户端计数，而不仅仅关注系统中存在的AP的数量。

大型活动便是一个很好的示例：一座大楼可以托管数千个客户端，以及数百个无线接入点，您需要将负载分配到尽可能多的CPU上，但同时也优化漫游。因此，除非有必要，否则您不会在WNCD上漫游。您想防止在同一物理位置混合不同WNCD/站点标签中的多个AP的“椒盐区”情况。

为了帮助微调并提供分布的可视化，您可以使用WCAE工具，并利用AP RF视图功能：



这使我们能够看到AP/WNCD分发，只需将View Type设置为WNCD。此处每种颜色代表WNCD/CPU。您还可以将RSSI过滤器设置为-85，以避免低信号连接，控制器中的RRM算法也会过滤低信号连接。

在上一个与Ciscolive EMEA 24对应的示例中，您可以看到大多数相邻的AP在同一个WNCD中交叉排列整齐，交叉重叠非常有限。

分配给同一WNCD的站点标签获得相同的颜色。

监控控制平面CPU使用情况

请务必记住Cisco IOS-XE架构的概念，并牢记CPU使用率的两个主要“视图”。一个来自历史上的Cisco IOS支持，另一个来自主要支持，具有跨所有进程和内核的CPU整体视图。

通常，您可以使用命令show processes cpu platform sorted收集整个Cisco IOS-XE的所有进程的详细信息：

9800cl-1#show processes cpu platform sorted

CPU utilization for five seconds: 8%, one minute: 14%, five minutes: 11%
Core 0: CPU utilization for five seconds: 6%, one minute: 11%, five minutes: 5%
Core 1: CPU utilization for five seconds: 2%, one minute: 8%, five minutes: 5%
Core 2: CPU utilization for five seconds: 4%, one minute: 12%, five minutes: 12%
Core 3: CPU utilization for five seconds: 19%, one minute: 23%, five minutes: 24%

Pid	PPid	5Sec	1Min	5Min	Status	Size	Name
19953	19514	44%	44%	44%	S	190880	ucode_pkt_PPE0
28947	8857	3%	10%	4%	S	1268696	linux_iosd-imag
19503	19034	3%	3%	3%	S	247332	fman_fp_image
30839	2	0%	0%	0%	I	0	kworker/0:0
30330	30319	0%	0%	0%	S	5660	nginx
30329	30319	0%	1%	0%	S	20136	nginx
30319	30224	0%	0%	0%	S	12480	nginx
30263	1	0%	0%	0%	S	4024	rotee
30224	8413	0%	0%	0%	S	4600	pman
30106	2	0%	0%	0%	I	0	kworker/u11:0
30002	2	0%	0%	0%	S	0	SarIosdMond
29918	29917	0%	0%	0%	S	1648	inet_gethost

这里需要强调几个要点：

- 进程ucode_pkt_PPE0正在处理9800L和9800CL平台上的数据平面，预计它始终会看到较高的利用率，甚至高于100%。这是实施的一部分，这不构成问题。

- 区分峰值使用与持续负载并隔离给定场景中的预期情况非常重要。例如，收集非常大的CLI输出(如show tech wireless)可能会在IOSd、smand、pubd进程上生成峰值负载，因为收集非常大的文本输出（执行数百个CLI命令）时，该输出不会出现问题，并且负载会在输出完成之后下降。

Pid	PPid	5Sec	1Min	5Min	Status	Size	Name
19371	19355	62%	83%	20%	R	128120	smand

```
27624   27617     53%     59%     59%  S         1120656  pubd
 4192    4123     11%      5%      4%  S         1485604  linux_iosd-imag
```

- 在高客户端活动时间内，WNCD核心的峰值利用率是预期的。可以看到80%的峰值，而不会对功能造成任何影响，而且它们通常不会构成问题。

```
  Pid    PPid    5Sec    1Min    5Min  Status       Size  Name
-----------------------------------------------------------------------------------
21094   21086     25%     25%     25%  S          978116  wncd_0
21757   21743     21%     20%     20%  R         1146384  wncd_4
22480   22465     18%     18%     18%  S         1152496  wncd_7
22015   21998     18%     17%     17%  S          840720  wncd_5
21209   21201     16%     18%     18%  S          779292  wncd_1
21528   21520     14%     15%     14%  S          926528  wncd_3
```

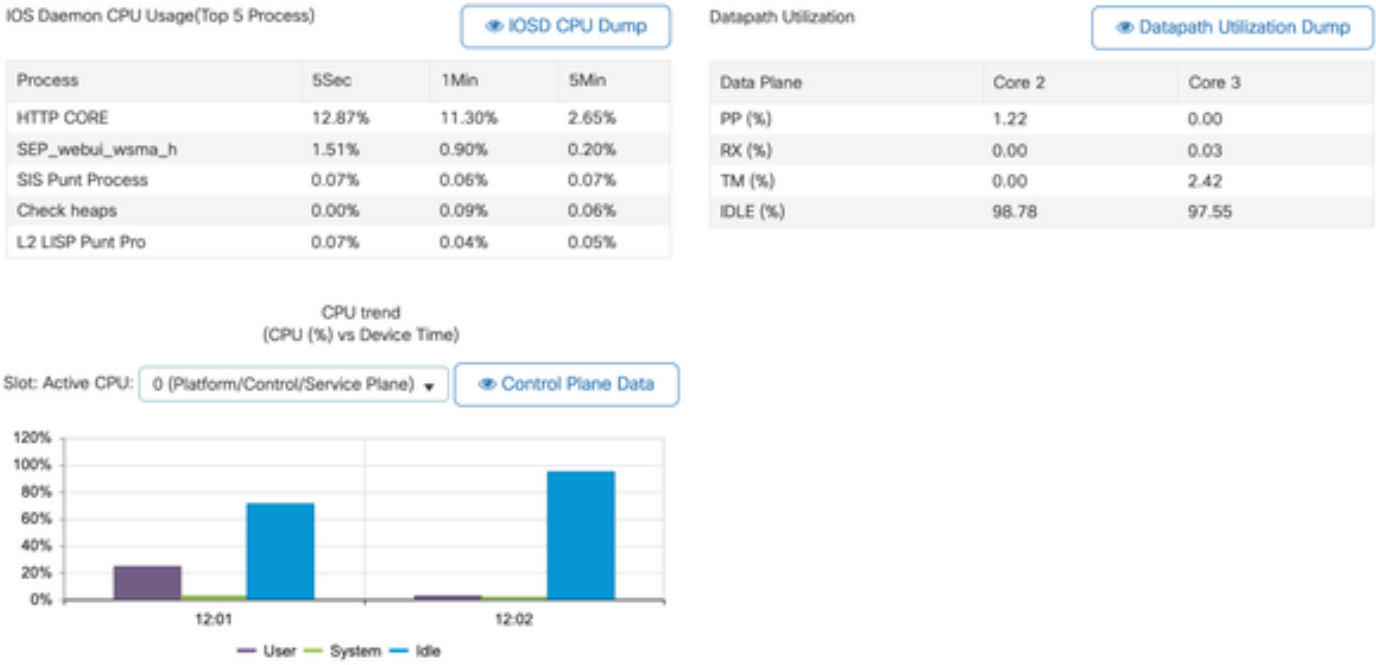
- 必须调查一个进程超过15分钟的持续高CPU使用率（高于90%）。

- 您可以使用命令show processes cpu sorted监控IOSd CPU使用率。这与Cisco IOS-XE列表的linux_iosd-imag进程部分中的活动相对应。

```
9800cl-1#show processes cpu sorted

CPU utilization for five seconds: 2%/0%; one minute: 3%; five minutes: 3%
 PID Runtime(ms)     Invoked      uSecs   5Sec   1Min   5Min TTY Process
 215         81          88        920  1.51%  0.12%  0.02%   1 SSH Process
 673     164441     7262624         22  0.07%  0.00%  0.00%   0 SBC main process
 137    2264141   225095413         10  0.07%  0.04%  0.05%   0 L2 LISP Punt Pro
 133     534184    21515771         24  0.07%  0.04%  0.04%   0 IOSXE-RP Punt Se
 474    1184139    56733445         20  0.07%  0.03%  0.00%   0 MMA DB TIMER
   5          0           1          0  0.00%  0.00%  0.00%   0 CTS SGACL db cor
   6          0           1          0  0.00%  0.00%  0.00%   0 Retransmission o
   2     198433      726367        273  0.00%  0.00%  0.00%   0 Load Meter
   7          0           1          0  0.00%  0.00%  0.00%   0 IPC ISSU Dispatc
  10    3254791      586076       5553  0.00%  0.11%  0.07%   0 Check heaps
   4         57          15       3800  0.00%  0.00%  0.00%   0 RF Slave Main Th
   8          0           1          0  0.00%  0.00%  0.00%   0 EDDRI_MAIN
```

- 您可以使用9800 GUI快速查看IOSd负载、每个核心使用情况和数据平面负载：



可从Monitoring/System/CPU Utilization选项卡获取它。

每个过程是什么？

确切的过程列表会因控制器型号以及Cisco IOS-XE版本而异。这是一些关键进程的列表，并不打算包括所有可能的条目。

进程名	它有哪些功能？	评估
wncd_x	处理大多数无线操作。根据9800型号，可以有1到8个实例	在繁忙时段您会看到高使用率的峰值。报告利用率在几分钟内是否停滞在95%或以上
linux_iosd-imag	IOS进程	如果收集大量CLI输出(show tech)，预计会看到高利用率 SNMP操作过大或过于频繁可能会导致高CPU
nginx	Web 服务器	此过程可以显示峰值，并且只应在持续的高负载上报告
ucode_pkt_PPE0	9800CL/9800L中的数据平面	使用 **show platform hardware chassis active qfp datapath utilization** 命令监视此组件
埃兹曼	用于接口的芯片组管理器	此处持续的CPU使用率较高可能表

		示硬件问题或可能的内核软件问题。应进行报告
dbm	数据库管理器	应报告此处的CPU使用率持续较高
odm_X	Operation Data Manager跨多个进程处理统一数据库	加载的系统预期的CPU使用率较高
rogued	处理欺诈功能	应报告此处的CPU使用率持续较高
smand	外壳管理器。处理CLI解析和不同进程之间的交互	处理大量CLI输出时预期的CPU使用率高。应报告在缺少负载时持续的CPU使用率较高
emd	外壳管理器。处理CLI解析和不同进程之间的交互	处理大量CLI输出时预期的CPU使用率高。应报告无负载时的CPU使用率持续较高
pubd	遥测处理的一部分	大型遥测订用预期的CPU使用率较高。应报告无负载时的CPU使用率持续较高

高CPU保护机制

Catalyst 9800无线LAN控制器围绕网络或无线客户端活动具有广泛的保护机制，可防止由于意外或故意的情况而导致CPU使用率过高。有几个关键功能旨在帮助您遏制有问题的设备：

客户端排除

默认情况下，这是启用的，并且是无线保护策略的一部分，可以根据策略配置文件启用或禁用。这可以检测多种不同的行为问题，从网络中移除客户端，并将其设置为“临时排除列表”。当客户端处于此排除状态时，AP不与其通信，从而阻止任何进一步操作。

排除计时器超时后（默认为60秒），允许客户端再次关联。

客户端排除有多个触发因素：

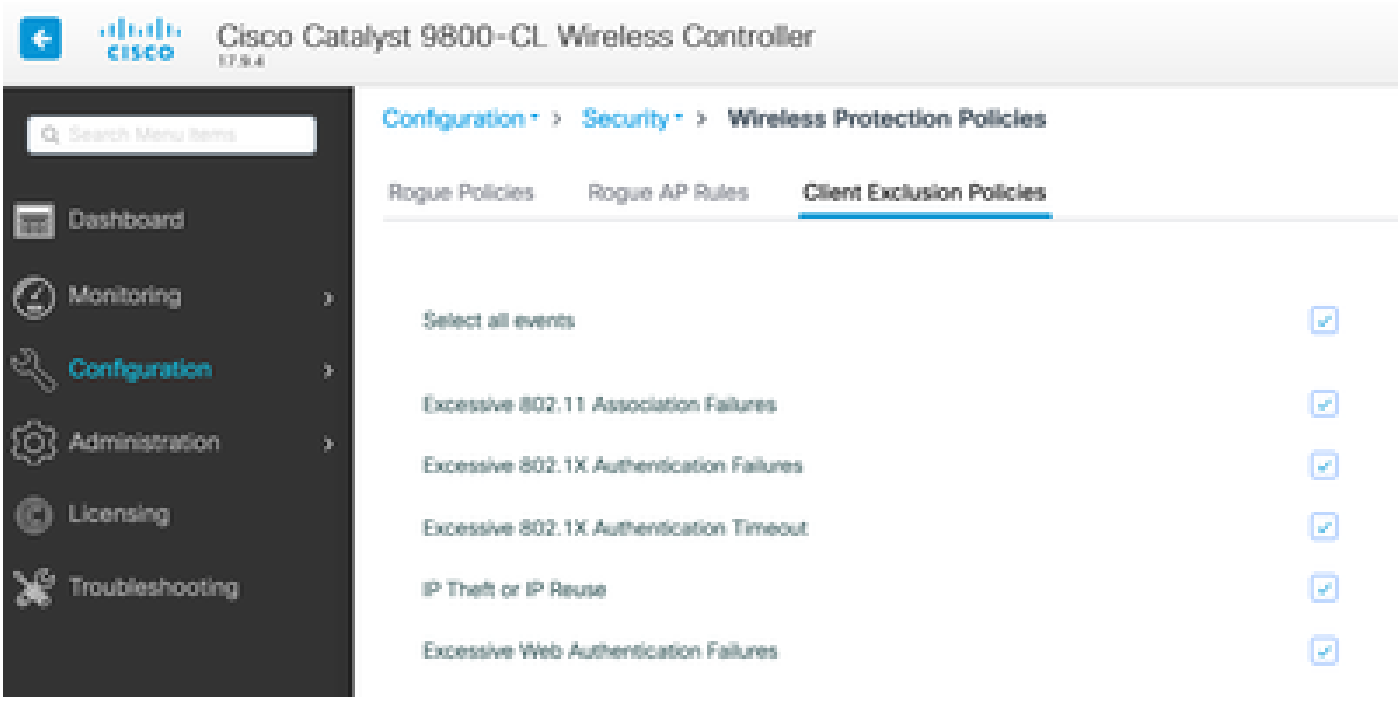
- 反复关联失败
- 3个或多个webauth、PSK或802.1x身份验证错误
- 重复的身份验证超时（客户端无响应）

- 尝试重复使用已注册到其他客户端的IP地址
- 生成ARP泛洪

客户端排除功能可保护您的控制器、AP和AAA基础设施(Radius)免受可能导致高CPU的几种高活动类型的影响。一般来说，除非故障排除练习或兼容性要求需要，否则不建议禁用任何排除方法。

默认设置适用于几乎所有情况，并且仅在某些特殊情况下适用，以增加排除时间或禁用某些特定触发器。例如，由于无法轻松修补客户端缺陷，某些传统或专业化客户端（IOT/医疗）可能需要禁用关联故障触发器

您可以在UI： Configuration/Wireless Protection/Client Exclusion Policies：



ARP排除触发器的设计目的是在全局级别永久启用，但可以在每个策略配置文件上进行自定义。您可以使用命令sh wireless profile policy all查看此特定输出来检查状态：

```
ARP Activity Limit
  Exclusion                 : ENABLED
  PPS                       : 100
  Burst Interval            : 5
```

针对数据流量的控制平面保护

这是数据平面中的一种高级机制，用于确保发送到控制平面的流量不超过预定义的阈值集。该功能称为“传送策略器”，在几乎所有情况下，都不需要触碰它们，即使这样，也只能在与Cisco支持配合使用时执行。

这种保护的优点是它提供非常详细的洞察力，以了解网络中发生的情况，以及是否存在任何速率增加或每秒数据包意外增加的特定活动。

这仅通过CLI显示，因为它们通常是无需修改的高级功能的一部分。

要查看所有传送策略，请执行以下操作：

9800-l#show platform software punt-policer

Per Punt-Cause Policer Configuration and Packet Counters

Punt Cause	Description	Config Rate(pps) Normal	Config Rate(pps) High	Conform Packets Normal	Conform Packets High	Dropped Packets Normal	Dropped Packets High	Config Burst(pkts) Normal	Config Burst(pkts) High	Config Alert Normal	Config Alert High
2	IPv4 Options	874	655	0	0	0	0	874	655	Off	Off
3	Layer2 control and legacy	8738	2185	33	0	0	0	8738	2185	Off	Off
4	PPP Control	437	1000	0	0	0	0	437	1000	Off	Off
5	CLNS IS-IS Control	8738	2185	0	0	0	0	8738	2185	Off	Off
6	HDLC keepalives	437	1000	0	0	0	0	437	1000	Off	Off
7	ARP request or response	437	1000	0	330176	0	0	437	1000	Off	Off
8	Reverse ARP request or repso	437	1000	0	24	0	0	437	1000	Off	Off
9	Frame-relay LMI Control	437	1000	0	0	0	0	437	1000	Off	Off
10	Incomplete adjacency	437	1000	0	0	0	0	437	1000	Off	Off
11	For-us data	40000	5000	442919246	203771	0	0	40000	5000	Off	Off
12	Mcast Directly Connected Sou	437	1000	0	0	0	0	437	1000	Off	Off

此列表可能很大，包含160个以上的条目，具体取决于软件版本。

在表输出中，您想要检查丢弃的数据包列，以及在高丢弃计数上具有非零值的任何条目。

要简化数据收集，您可以使用命令show platform software punt-policer drop-only，仅过滤带有丢弃的监察器条目。

此功能可用于确定是否存在ARP风暴或802.11探测泛洪（它们使用队列“发往LFTS的802.11数据包”）。LFTS代表Linux转发传输服务)。

无线呼叫准入控制

在所有最近的维护版本中，控制器都有一个活动监控器，以动态响应高CPU使用率，并确保AP CAPWAP隧道在不可持续的压力下保持活动状态。

该功能会检查WNCD负载，并开始限制新客户端活动，以确保留下足够的资源来处理现有连接并保护CAPWAP稳定性。

默认情况下会启用此功能，而且它没有配置选项。

定义了三个保护级别：L1为80%负载，L2为85%负载，L3为89%，每个级别触发不同的传入协议丢弃作为保护机制。一旦负载降低，将自动删除保护。

在正常的网络中，您不应看到L2或L3负载事件，如果它们频繁发生，则应进行调查。

要监控，请使用图中所示的wireless stats cac命令。

9800-l# show wireless stats cac

WIRESLESS CAC STATISTICS

L1 CPU Threshold: 80 L2 CPU Threshold: 85 L3 CPU Threshold: 89
Total Number of CAC throttle due to IP Learn: 0
Total Number of CAC throttle due to AAA: 0
Total Number of CAC throttle due to Mobility Discovery: 0
Total Number of CAC throttle due to IPC: 0
CPU Throttle Stats
L1-Assoc-Drop: 0 L2-Assoc-Drop: 0 L3-Assoc-Drop: 0
L1-Reassoc-Drop: 0 L2-Reassoc-Drop: 0 L3-Reassoc-Drop: 0
L1-Probe-Drop: 12231 L2-Probe-Drop: 11608 L3-Probe-Drop: 93240
L1-RFID-Drop: 0 L2-RFID-Drop: 0 L3-RFID-Drop: 0
L1-MDNS-Drop: 0 L2-MDNS-Drop: 0 L3-MDNS-Drop: 0

mDNS保护

mDNS作为一种协议允许“零接触”方法来发现设备间的服务，但同时，它可能非常活跃，并且如果配置不当，会显著增加负载。

mDNS无需任何过滤，可以轻松地提高WNCD CPU利用率，这有以下几个因素：

- mDNS策略通过无限制学习，控制器将获得所有设备提供的所有服务。这会导致服务列表非常大，包含数百个条目。
- 未过滤的策略设置：这将导致控制器将这些大型服务列表推送到询问谁提供指定服务的每个客户端。
- 某些mDNS特定服务由“所有”无线客户端提供，导致服务数量和活动增加，但操作系统版本有所不同。

您可以使用以下命令检查每个服务的mDNS列表大小：

9800-l# show mdns-sd service statistics

Service Name	Service Count
_ipp._tcp.local	84
_ipps._tcp.local	52
_raop._tcp.local	950
_airplay._tcp.local	988
_printer._tcp.local	13
_googlerpc._tcp.local	12
_googlecast._tcp.local	70
_googlezone._tcp.local	37
_home-sharing._tcp.local	7
_cups._sub._ipp._tcp.local	26

这可以提供获得任何给定查询的规模的概念，它本身并不表示问题，而只是一种监控所跟踪内容的方法。

以下是一些重要的mDNS配置建议：

- 将mDNS传输设置为单一协议：

```
9800-1(config)# mdns-sd gateway

9800-1(config-mdns-sd)# transport ipv4
```

默认情况下，它使用IPv4传输，为了获得性能，建议使用IPv6或IPv4，但不要同时使用两者：

- 请始终在mDNS服务策略中设置位置过滤器，以避免未绑定的查询/响应。一般而言，建议使用“site-tag”，但其他选项可能会起作用，具体取决于您的需求。

我需要更多帮助

如果您看到高CPU负载，并且以上都不起作用，请通过案例与CX联系，并将此数据添加为起点：

- 基础数据，包括接入点/控制器配置以及网络和RF操作值：

```
show tech-support wireless
```

- 所有控制器跟踪的存档。这是一个类似于“黑盒”概念的大文件，可以使用命令进行收集：

```
request platform software trace archive last <days> to-file bootflash:<archive file>
```

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）。