

Identificar processos e núcleos do Linux com alta utilização da CPU no FTD

Contents

[Introdução](#)

[Pré-requisitos](#)

[Requisitos](#)

[Componentes Utilizados](#)

[Informações de Apoio](#)

[Troubleshooting](#)

[Identificar o processo do Linux que consome muita CPU](#)

[Alto uso da CPU nos núcleos do sistema Cisco Secure Firewall](#)

[Informações Relacionadas](#)

Introdução

Este documento descreve o alto uso da CPU no Cisco Secure Firewall Threat Defense (FTD), onde o alto se refere ao uso da CPU de 90% ou mais.

Pré-requisitos

Requisitos

- Cisco Secure Firewall Management Center (FMC)
- Cisco Secure Firewall Threat Defense (FTD).

Componentes Utilizados

- Cisco Secure Firewall Management Center executando a versão 7.4.2.
- Cisco Secure Firewall Threat Defense executando a versão 7.4.2.

Informações de Apoio

O Cisco Secure Firewall é uma imagem unificada de Lina e Snort. Os processos comprometem o back-end do sistema Cisco Secure Firewall. Os processos e threads no Cisco Secure Firewall têm afinidade, que se refere à capacidade de vincular um processo ou thread a um núcleo de CPU específico ou a um conjunto de núcleos. Alguns processos, como Lina ou Snort, têm afinidade exclusiva com núcleos específicos, o que significa que nenhum outro processo ou segmento pode ser executado nesses núcleos.

Troubleshooting

Identificar o processo do Linux que consome muita CPU

Para identificar os processos que consomem muita CPU no Cisco Secure Firewall, consulte estas saídas:

a saída do comando `top` nos ajuda a identificar os processos que consomem a maior parte da CPU.

```
top - 17:27:37 up 62 days, 22:52, 2 users, load average: 0.41, 0.55, 0.48
Tasks: 158 total, 1 running, 157 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.9 us, 1.1 sy, 0.7 ni, 93.4 id, 3.9 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 7981.8 total, 821.0 free, 3184.3 used, 3976.5 buff/cache
MiB Swap: 5378.2 total, 4361.2 free, 1017.0 used. 4563.9 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
3394	root	25	5	425680	4860	4652	S	7.6	0.1	6494:36	loggerd
3685	root	0	-20	2958720	1.4g	130268	S	6.0	18.0	5062:29	lina
4494	root	1	-19	1339224	509956	44688	S	1.7	6.2	1659:12	snort3
3979	root	20	0	2270592	141480	16840	S	0.7	1.7	637:29.39	SFDataCorrelator

Pressione 1 e verifique a utilização da CPU por núcleo.

```
top - 17:30:36 up 62 days, 22:55, 2 users, load average: 0.33, 0.47, 0.45
Tasks: 157 total, 1 running, 156 sleeping, 0 stopped, 0 zombie
%Cpu0 : 3.0 us, 0.0 sy, 0.0 ni, 97.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
%Cpu1 : 0.0 us, 0.0 sy, 0.0 ni,100.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
%Cpu2 : 0.0 us, 0.3 sy, 0.0 ni, 99.7 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
%Cpu3 : 1.3 us, 5.3 sy, 2.7 ni, 90.7 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 7981.8 total, 831.5 free, 3173.6 used, 3976.7 buff/cache
MiB Swap: 5378.2 total, 4361.2 free, 1017.0 used. 4574.7 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
3394	root	25	5	425680	4860	4652	S	7.6	0.1	6494:36	loggerd
3685	root	0	-20	2958720	1.4g	13026	S	6.0	18.0	5062:29	lina
4494	root	1	-19	1339224	509956	44688	S	1.7	6.2	1659:12	snort3
3979	root	20	0	2270592	141480	16840	S	0.7	1.7	637:29.39	SFDataCorrelator

Pressione c para verificar os detalhes do processo.

```
top - 17:31:53 up 62 days, 22:56, 2 users, load average: 0.34, 0.45, 0.45
Tasks: 157 total, 1 running, 156 sleeping, 0 stopped, 0 zombie
%Cpu0 : 3.7 us, 1.0 sy, 0.0 ni, 95.3 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
%Cpu1 : 0.0 us, 0.3 sy, 0.0 ni, 99.7 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
%Cpu2 : 0.0 us, 0.0 sy, 0.0 ni,100.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
%Cpu3 : 2.0 us, 3.7 sy, 2.7 ni, 91.7 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
```

MiB Mem : 7981.8 total, 831.4 free, 3173.7 used, 3976.7 buff/cache
MiB Swap: 5378.2 total, 4361.2 free, 1017.0 used. 4574.5 avail Mem

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
3394	root	25	5	425680	4860	4652	S	7.3	0.1	6494:55	/ngfw/usr/local/sf/bin/loggerd
3685	root	0	-20	2958720	1.4g	130268	S	6.0	18.0	5062:45	lina -p 3562 -t -l
4494	root	1	-19	1339224	509956	44688	S	2.3	6.2	1659:17	/ngfw/var/sf/detection_engines/ad5
3418	root	20	0	1640540	16824	13900	S	0.7	0.2	274:48.05	/ngfw/usr/local/sf/bin/adi
3979	root	20	0	2270592	141480	16840	S	0.7	1.7	637:31.13	/ngfw/usr/local/sf/bin/SFDataCorre
3380	root	20	0	3860	2852	2336	S	0.3	0.0	6:27.14	/bin/bash /ngfw/usr/local/sf/bin/p

Considere um exemplo de um processo (monetdb) consumindo CPU alta depois de verificar pelo comando top que monetdb é o culpado.

Você pode rastrear merovingian.log no modo especialista ngfw/var/log/monetdb/merovingian.log.

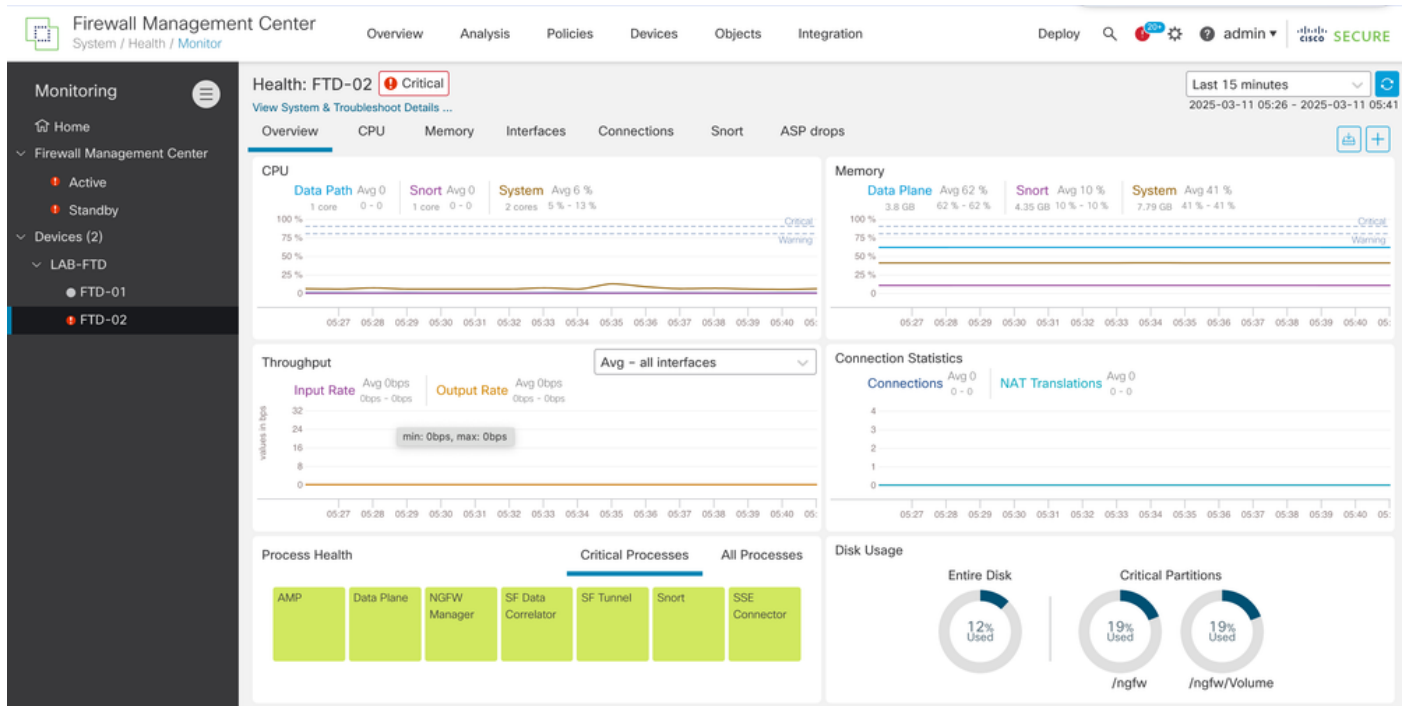
```
root@FirePower:/# cd /ngfw/var/log/#catmerovingian.log
```

```
2025-03-06 23:24:11 ERR eventdb[26531]: #client12: createExceptionInternal:
!ERROR: SQLException:sql.drop_table:42000!DROP TABLE: unable to drop table connectionevent_1720808160_0
2025-03-06 23:24:11 ERR eventdb[26531]: #client12: createExceptionInternal:
!ERROR: SQLException:sql.drop_table:42000!DROP TABLE: unable to drop table
connectionevent_1720971900_0 (there are database objects which depend on it)
```

Alto uso da CPU nos núcleos do sistema Cisco Secure Firewall

Estas são as etapas de solução de problemas em caso de alta utilização da CPU nos núcleos de sistema do Cisco Secure Firewall.

1. Confirme o alto uso do sistema da CPU no monitoramento de integridade do FMC.
 - Verifique a coluna de notificação no FMC para verificar o alto uso da CPU.
 - Navegue até System > Health > Monitor e observe os núcleos afetados.



2. Verifique os núcleos de CPU afetados no FTD CLI.

- Verifique a afinidade da CPU e confirme se os núcleos afetados são o núcleo do sistema.

```
> pmtool show affinity
```

```
Received status (0):
```

```
Affinity Status
```

```
System CPU Affinity: 3,9 (desired 3,9 )
```

```
..
```

```
<#root>
```

```
admin@firepower:~$
```

```
sudo mpstat -P 3,9 1
```

```
Linux 4.18.45-yocto-standard (firepower)      03/03/25      _x86_64_      (64 CPU)
```

```
1:06:15      CPU
```

```
%usr    %nice    %sys
```

```
%iowait  %irq    %soft  %steal  %guest  %gnice
```

```
%idle
```

```
1:06:15      3
```

```
4
```

.00 54.00 24.00

0.00 0.00 0.00 0.00 0.00 0.00

14.00

1:06:15 9

0.00 74.00 25.00

0.00 0.00 0.00 0.00 0.00 0.00

0.00

1:06:15 CPU

%usr %nice %sys

 %iwait %irq %soft %steal %guest %gnice %idle

1:06:16 3

6

.00 44.00 34.00

0.00 0.00 0.00 0.00 0.00 0.00

4

.00

1:06:16 9

2.00 74.00 22.00

0.00 0.00 0.00 0.00 0.00 0.00

0.00

1:06:16 CPU

%usr %nice %sys

 %iwait %irq %soft %steal %guest %gnice %idle

1:06:16 3

6

.00 34.00 24.00

0.00 0.00 0.00 0.00 0.00 0.00

34.00

1:06:16 9

0.00 74.00 24.00

0.00 0.00 0.00 0.00 0.00 0.00

0.00

...

Average:	9	3.71	67.43	22.14	0.00	0.00	0.00	0.00	0.00	0.00	9.71
----------	---	------	-------	-------	------	------	------	------	------	------	------

- Run `mpstat` para monitorar a utilização da CPU nos núcleos do sistema. Neste exemplo, refere-se ao intervalo de monitoramento de 1/segundo.

Quanto maior for o uso da CPU, menor será o %idle porque equivale a 100 - total de outra coluna %.

<#root>

admin@firepower:~\$

sudo

sar -P 3,9 1

Linux 4.18.45-yocto-standard (firepower) 03/03/24 _x86_64_ (64 CPU)

1:06:17 CPU

%user %nice %system

%iowait %steal

%idle

1:06:18 3

5

.00 54.00 28.00

0.00 0.00

14.00

1:06:18 9

1.00 80.00 19.00

0.00 0.00

0.00

1:06:18 CPU

%user %nice %system

%iowait %steal

%idle

1:06:19 3

5

.00	41.00	22.00
0.00	0.00	

24.00

1:06:19 9

2.00	84.00	12.00
0.00	0.00	

0.00

1:06:19 CPU

%user	%nice	%system
%iowait	%steal	

%idle

1:06:20 3

1.00	67.00	15.00
0.00	0.00	

4

.00

1:06:20 9

5

.00	68.00	24.00
0.00	0.00	

0.00

1:06:20 CPU

%user	%nice	%system
%iowait	%steal	

%idle

1:09:21 3

6.00	26.00	27.00
0.00	0.00	

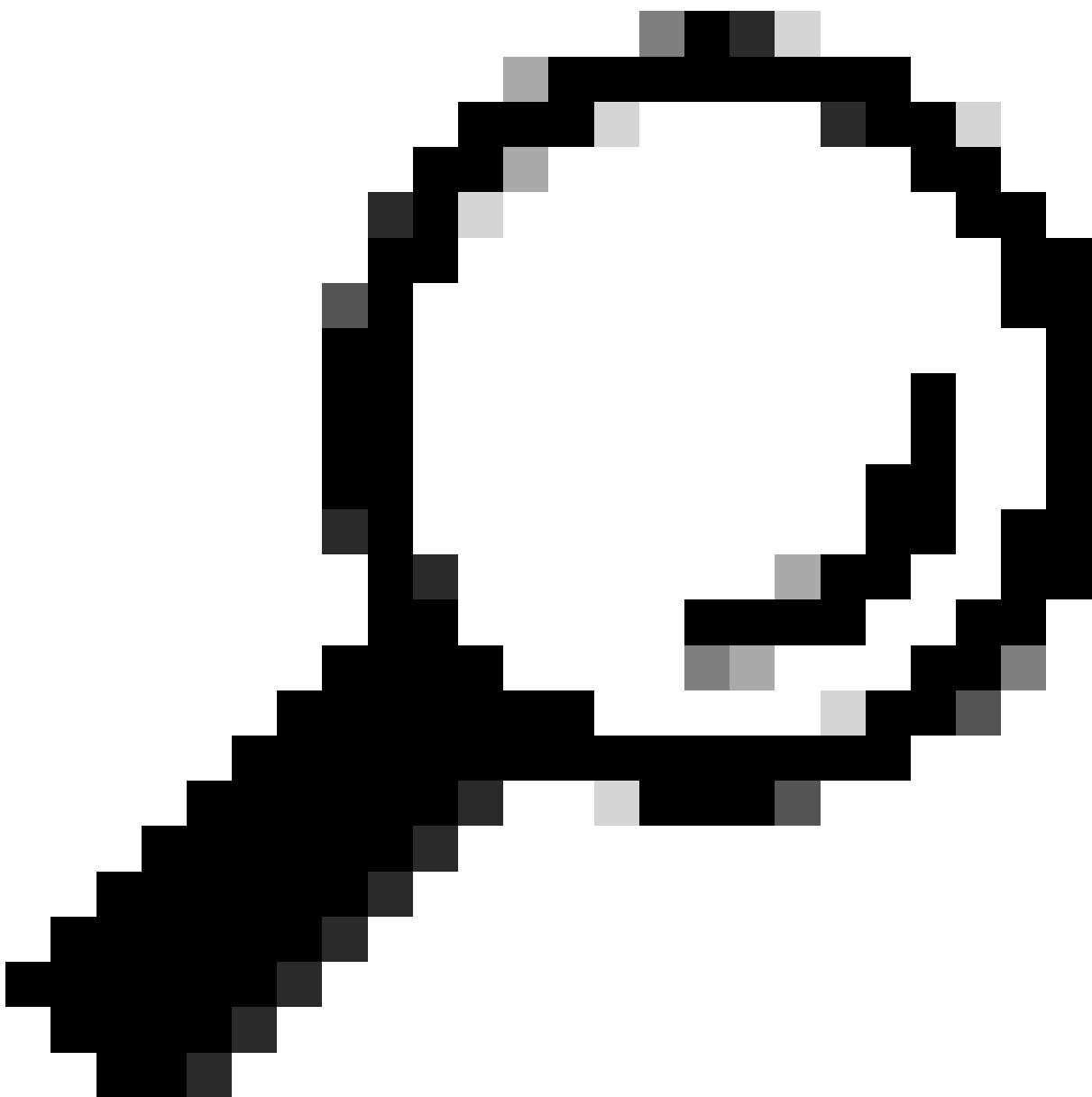
41.00

1:09:21 9

```
0.00      86.00      14.00
          0.00      0.00
0.00
```

O título da coluna é apresentado como:

- %user - Uso da CPU no espaço do usuário (código do aplicativo) com o valor padrão nice.
 - %nice - uso da CPU no espaço do usuário (código do aplicativo) com um valor positivo.
 - %system - uso da CPU no espaço do kernel. %idle - 100% menos o restante dos campos
-



Tip: Execute este comando para encontrar a lista de threads em execução em núcleos específicos.

```
pidstat -h -t -p ALL 1 | awk '($10=="%CPU") || (US$ 10==XX || $10==AA) && $9+0 >
```

10,00)

Execute o comando top no modo especialista para encontrar a lista de processos em execução em núcleos específicos.

3. Verifique se um processo é um multithread.

<#root>

admin@firepower

:

~\$

sudo pidstat -h -t -p \$(pidof snort3) 1

Linux 4.18.45-yocto-standard (firepower)				03/03/25		_x86_64_		(64 CPU)		
# Time	UID	TGID	TID	%usr	%system	%guest	%wait	%CPU	CPU	Command
4:20:05	0	4162	-	279.08	17.17	0.00	0.99	285.15	3	snort3 <--
4:20:05	0	-	4162	3.96	0.99	0.00	0.99	4.95	3	__
snort3										
4:20:05	0	-	4165	0.00	0.00	0.00	0.00	0.00	7	__
snort3										
4:20:05	0	-	4167	0.00	0.00	0.00	0.00	0.00	9	__
snort3										
4:20:05	0	-	4168	0.00	0.00	0.00	0.00	0.00	9	__
snort3										
4:20:05	0	-	4169	0.00	0.00	0.00	0.00	0.00	9	__snort3
4:20:05	0	-	4172	0.00	0.00	0.00	0.00	0.00	9	__snort3
4:20:05	0	-	4175	0.00	0.00	0.00	0.00	0.00	9	__snort3
4:20:05	0	-	4177	0.00	0.00	0.00	0.00	0.00	3	__snort3

<#root>

admin@firepower

:

~\$

sudo

```
pidstat -h -t -p $(pidof rsyslog) 1
```

```
Linux 4.18.45-yocto-standard (firepower)      03/03/25      _x86_64_      (64 CPU)

# Time      UID      TGID      TID      %usr %system %guest  %wait  %CPU  CPU  Command
4:22:47      0        1        -      0.00  0.00  0.00   0.00   0.00  32  init
4:22:47      0        -        1      0.00  0.00  0.00   0.00   0.00  32  |__init <---- single
```

4. Verifique se algum processo é executado nos núcleos do sistema.

```
<#root>
```

```
admin@firepower:~$
```

```
sudo pidstat -h -t -p $(pidof EventHandler) 1
```

```
Password:
Linux 4.18.45-yocto-standard (firepower)      03/03/25      _x86_64_      (64 CPU)

# Time      UID      TGID      TID      %usr %system %guest  %wait  %CPU  CPU  Command
04:46:01      0    327455      -      61.00  23.00  0.00   0.00   83.00   3
EventHandler
```

```
<--- All threads run on system cores
```

```
04:46:01      0        -    327455      0.00  0.00  0.00   0.00   0.00   3  |__Eventt
04:46:01      0        -    327456     30.00  0.00  0.00   60.00  40.00   3  |__Eventt
04:46:01      0        -    327457     29.00  9.00  0.00   64.00  33.00   9  |__Eventt
04:46:01      0        -    327458      0.00  0.00  0.00   0.00   0.00   3  |__Eventt
04:46:01      0        -    327459      0.00  0.00  0.00   0.00   0.00   3  |__zmqio
04:46:01      0        -    327465      0.00  0.00  0.00   0.00   0.00   9  |__zmqio
04:46:01      0        -    327466      0.00  0.00  0.00   1.00   0.00   9  |__Eventt
04:46:01      0        -    327468      3.00  2.00  0.00   3.00   5.00   9  |__Eventt
04:46:01      0        -    327470      0.00  0.00  0.00   0.00   0.00   9  |__Eventt
04:46:01      0        -    327475      0.00  0.00  0.00   0.00   0.00   3  |__Eventt
04:46:01      0        -    327476      0.00  0.00  0.00   0.00   0.00   9  |__Eventt
04:46:01      0        -    327477      0.00  0.00  0.00   0.00   0.00   3  |__Eventt
04:46:01      0        -    327478      0.00  0.00  0.00   0.00   0.00   9  |__zmqio
04:46:01      0        -    327479      0.00  0.00  0.00   0.00   0.00   9  |__zmqio
```

5. Verificar se qualquer um dos threads contribui para ou causar alto uso da CPU nos núcleos do sistema.

```
<#root>
```

```
admin@firepower:~$
```

```
sudo pidstat -h -t -p $(pidof EventHandler) 1
```

```
Linux 4.18.45-yocto-standard (firepower)      03/03/25      _x86_64_      (64 CPU)

# Time      UID      TGID      TID      %usr  %system  %guest  %wait  %CPU  CPU  Command
05:13:30    0      327455    -      111.77  19.80    0.00    0.00   136.57  3    EventHa

<----- Process CPU usage = Σ (threads %CPU).

05:13:30    0      -      327455    0.00    0.00    0.00    0.00    0.00    3    |__Event
05:13:30    0      -      327456    33.64    0.00    0.00    65.17   37.54    9    |__Event

<----- Thread on system cores with highest CPU usage

05:13:30    0      -      327457    74.24    15.83    0.00    7.92   94.18    3    |__Event

<----- Thread on system cores with highest CPU usage

05:13:30    0      -      327458    0.00    0.00    0.00    0.00    0.00    3    |__Event
05:13:30    0      -      327459    0.00    0.00    0.00    0.00    0.00    9    |__zmqi
05:13:30    0      -      327465    0.00    0.00    0.00    0.00    0.00    9    |__zmqi
05:13:30    0      -      327466    0.00    0.99    0.00    0.00    0.99    3    |__Event
05:13:30    0      -      327468    1.98    2.97    0.00    5.94    4.85    9    |__Event
05:13:30    0      -      327470    0.00    0.00    0.00    0.99    0.00    3    |__Event
05:13:30    0      -      327475    0.00    0.00    0.00    0.00    0.00    9    |__Event
05:13:30    0      -      327476    0.00    0.00    0.00    0.00    0.00    9    |__Event
05:13:30    0      -      327477    0.00    0.00    0.00    0.00    0.00    9    |__Event
05:13:30    0      -      327478    0.00    0.00    0.00    0.00    0.00    3    |__zmqi
05:13:30    0      -      327479    0.00    0.00    0.00    0.00    0.00    9    |__zmqi
```

- EventHandler é uma CPU multithread executada nos núcleos do sistema.
- Threads TID=327456e TID=327457 têm o maior uso de CPU.
- O uso da CPU do processo é igual à soma do uso da CPU de todos os threads.

Informações Relacionadas

[Guia de Administração do Cisco Secure Firewall Management Center, 7.6](#)

[Guia de configuração de dispositivos do Cisco Secure Firewall Management Center, 7.6](#)

Sobre esta tradução

A Cisco traduziu este documento com a ajuda de tecnologias de tradução automática e humana para oferecer conteúdo de suporte aos seus usuários no seu próprio idioma, independentemente da localização.

Observe que mesmo a melhor tradução automática não será tão precisa quanto as realizadas por um tradutor profissional.

A Cisco Systems, Inc. não se responsabiliza pela precisão destas traduções e recomenda que o documento original em inglês ([link fornecido](#)) seja sempre consultado.