

Autor: [Pedro Daniel Gonçalves Antunes]

Data: [2025/04/23]

Curso/Disciplina: [LSIRC/PPR]

Instituição: [ESTG]

NIM:[8230068]

Ficha:[Ficha Prática 7]

1. Ambiente de Rede e Testes de Conectividade

Para a realização dos testes e comandos SNMP, foi preparado o seguinte ambiente com duas máquinas virtuais Ubuntu Server:

Host	IP	Função
Server1	192.168.1.10	NMS (Network Management Station) com ferramentas SNMP
Server2	192.168.1.11	Agente SNMP monitorizado (client)

 Configuração de IPs

Server1

```
# network: {config: disabled}
network:
  ethernets:
    enp0s3:
      addresses:
        - 192.168.100.10/24
      gateway4: 192.168.100.1
      nameservers:
        addresses: [8.8.8.8]
  version: 2
```

Server2

```
# network: {config: disabled}
network:
  ethernets:
    enp0s3:
      addresses:
        - 192.168.100.11/24
      gateway4: 192.168.100.1
      nameservers:
        addresses: [8.8.8.8]
  version: 2
```

Ambos os servidores foram configurados com IPs estáticos na mesma rede (`192.168.1.0/24`), garantindo acessibilidade direta entre os dois.

Testes de Conectividade

Para garantir que a comunicação entre os dois hosts estava operacional, foram realizados testes de **ping** bidirecionais:

SERVER 1 -> SERVER 2

```
valid_ifc forever preferred_ifc forever
server1@Server1:~$ ping 192.168.100.11
PING 192.168.100.11 (192.168.100.11) 56(84) bytes of data.
64 bytes from 192.168.100.11: icmp_seq=1 ttl=64 time=0.972 ms
64 bytes from 192.168.100.11: icmp_seq=2 ttl=64 time=0.497 ms
64 bytes from 192.168.100.11: icmp_seq=3 ttl=64 time=0.544 ms
^C
--- 192.168.100.11 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2002ms
rtt min/avg/max/mdev = 0.497/0.671/0.972/0.213 ms
```

SERVER 2 -> SERVER 1

```
server1@Server1:~$ ping 192.168.100.10
PING 192.168.100.10 (192.168.100.10) 56(84) bytes of data.
64 bytes from 192.168.100.10: icmp_seq=1 ttl=64 time=0.497 ms
64 bytes from 192.168.100.10: icmp_seq=2 ttl=64 time=0.591 ms
64 bytes from 192.168.100.10: icmp_seq=3 ttl=64 time=0.462 ms
^C
--- 192.168.100.10 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2035ms
rtt min/avg/max/mdev = 0.462/0.516/0.591/0.054 ms
```

Os resultados mostraram tempos de resposta consistentes e sem perda de pacotes, validando que a camada de rede estava corretamente configurada para os testes SNMP subsequentes.

1.1 Consulta do Nome da Máquina host2 (sysName)

Para responder à questão 1.1, foi necessário consultar a OID correspondente ao nome do sistema, a partir do `host1` usando o protocolo SNMP v2c com a comunidade `public`.

Antes disso, foi confirmada a correta configuração do ficheiro `/etc/snmp/snmpd.conf` no `host2`, onde a comunidade `public` foi atribuída para leitura:

```
# sysLocation - location of the system
sysLocation    "Lab Redes"
sysContact     admin@host2.local
sysName host2
# sysServices: The proper value for the sysServices object.
# arguments: sysServices_number
sysServices    72
```

Com esta configuração, foi possível executar o seguinte comando no `host1`:

```
server1@Server1:~$ snmpget -v2c -c public 192.168.100.11 1.3.6.1.2.1.1.5.0
SNMPv2-MIB::sysName.0 = STRING: host2
server1@Server1:~$
```

O output do comando retornou o nome atual atribuído ao `host2`, que

inicialmente era o hostname configurado localmente. Esse valor foi usado como referência para a alteração na próxima questão (1.2).

1.2 Alteração do Nome Atribuído ao host2 para “client”

Para alterar o nome da máquina `host2` via SNMP, foi necessário permitir escrita através do ficheiro de configuração do agente SNMP localizado em:

```
# Read-only access to everyone to the systemonly view
rocommunity public default -V systemonly
rocommunity6 public default -V systemonly
rwcommunity private 192.168.100.10
```

Foi executado o seguinte comando no `host1` para alterar o nome:

```
snmpset -v2c -c private 192.168.1.11 SNMPv2-
MIB::sysName.0 s "client"
```

```
SNMPv2-MIB::sysName.0 = STRING: client
```

A alteração foi bem-sucedida, e o nome do host `host2` passou a ser “client”, como confirmado no passo seguinte (1.3).

1.3 Listar todas as variáveis sob o grupo “system” (OID 1.3.6.1.2.1.1)

Para listar todas as variáveis do grupo `system` na MIB SNMP, foi utilizado o comando `snmpwalk` a partir do `host1`, com acesso via SNMP v2c à comunidade `public`.

O grupo `system` encontra-se sob o OID `1.3.6.1.2.1.1` e contém variáveis como:

- sysDescr
- sysObjectID
- sysUpTime
- sysContact
- sysName
- sysLocation
- sysServices

O comando executado foi:

```
server1@Server1:~$ snmpwalk -v2c -c public 192.168.100.11 1.3.6.1.2.1.1
```

Este comando percorre todos os OIDs contidos sob o ramo `system`, retornando os valores atualmente configurados e reportados pelo agente SNMP no `host2`.

```
SNMPv2-MIB::sysContact.0 = STRING: admin@host2.local
SNMPv2-MIB::sysName.0 = STRING: client
SNMPv2-MIB::sysLocation.0 = STRING: "Lab Redes"
SNMPv2-MIB::sysServices.0 = INTEGER: 72
SNMPv2-MIB::sysORLastChange.0 = Timeticks: (0) 0:00:00.00
SNMPv2-MIB::sysORID.1 = OID: SNMP-FRAMEWORK-MIB::snmpFrameworkMIBCompliance
SNMPv2-MIB::sysORID.2 = OID: SNMP-MPD-MIB::snmpMPDCompliance
SNMPv2-MIB::sysORID.3 = OID: SNMP-USER-BASED-SM-MIB::usmMIBCompliance
SNMPv2-MIB::sysORID.4 = OID: SNMPv2-MIB::snmpMIB
SNMPv2-MIB::sysORID.5 = OID: SNMP-VIEW-BASED-ACM-MIB::vacmBasicGroup
SNMPv2-MIB::sysORID.6 = OID: TCP-MIB::tcpMIB
SNMPv2-MIB::sysORID.7 = OID: UDP-MIB::udpMIB
SNMPv2-MIB::sysORID.8 = OID: IP-MIB::ip
SNMPv2-MIB::sysORID.9 = OID: SNMP-NOTIFICATION-MIB::snmpNotifyFullCompliance
SNMPv2-MIB::sysORID.10 = OID: NOTIFICATION-LOG-MIB::notificationLogMIB
SNMPv2-MIB::sysORDescr.1 = STRING: The SNMP Management Architecture MIB.
SNMPv2-MIB::sysORDescr.2 = STRING: The MIB for Message Processing and Dispatching.
SNMPv2-MIB::sysORDescr.3 = STRING: The management information definitions for the SNMP User-based Security Model.
SNMPv2-MIB::sysORDescr.4 = STRING: The MIB module for SNMPv2 entities
SNMPv2-MIB::sysORDescr.5 = STRING: View-based Access Control Model for SNMP.
SNMPv2-MIB::sysORDescr.6 = STRING: The MIB module for managing TCP implementations
SNMPv2-MIB::sysORDescr.7 = STRING: The MIB module for managing UDP implementations
SNMPv2-MIB::sysORDescr.8 = STRING: The MIB module for managing IP and ICMP implementations
SNMPv2-MIB::sysORDescr.9 = STRING: The MIB modules for managing SNMP Notification, plus filtering.
SNMPv2-MIB::sysORDescr.10 = STRING: The MIB module for logging SNMP Notifications.
SNMPv2-MIB::sysORUpTime.1 = Timeticks: (0) 0:00:00.00
SNMPv2-MIB::sysORUpTime.2 = Timeticks: (0) 0:00:00.00
SNMPv2-MIB::sysORUpTime.3 = Timeticks: (0) 0:00:00.00
SNMPv2-MIB::sysORUpTime.4 = Timeticks: (0) 0:00:00.00
SNMPv2-MIB::sysORUpTime.5 = Timeticks: (0) 0:00:00.00
SNMPv2-MIB::sysORUpTime.6 = Timeticks: (0) 0:00:00.00
SNMPv2-MIB::sysORUpTime.7 = Timeticks: (0) 0:00:00.00
SNMPv2-MIB::sysORUpTime.8 = Timeticks: (0) 0:00:00.00
SNMPv2-MIB::sysORUpTime.9 = Timeticks: (0) 0:00:00.00
SNMPv2-MIB::sysORUpTime.10 = Timeticks: (0) 0:00:00.00
server1@Server1:~$ _
```

1.4 Obter Toda a Informação Conhecida pelo Agente no host2 (client)

O agente SNMP do `host2` (client) disponibiliza uma árvore completa de OIDs. Para recolher toda a informação exposta, pode ser utilizado o comando `snmpwalk` sem especificar um OID, iniciando assim a varredura a partir da raiz da MIB.

O comando utilizado no `host1` foi:

```
server1@Server1:~$ snmpwalk -v2c -c public 192.168.100.11 > output_snmp_total_txt_
```

Este comando percorre todos os ramos disponíveis no agente SNMP, listando informação sobre o sistema, interfaces de rede, armazenamento, processos, estatísticas de protocolos e muito mais.

1.5 Configuração do SNMP v3 com autenticação (authNoPriv)

Nesta etapa, foi configurado o agente SNMP no `host2` para operar com SNMP versão 3, utilizando **autenticação sem privacidade (authNoPriv)**. O objetivo foi aceder remotamente ao `sysName` exigindo credenciais seguras.

✅ Passo 1: Criar utilizador SNMPv3 no `host2`

No ficheiro `/var/lib/snmp/snmpd.conf` foi adicionada a seguinte configuração:

```
#
# createUser username (MD5|SHA|SHA-512|SHA-384|SHA-256|SHA-224) authpassphrase [DES|AES] [privpassp
# e.g.
# createUser authPrivUser SHA-512 myauthphrase AES myprivphrase
#
# This should be put into /var/lib/snmp/snmpd.conf
#
# rouser: a SNMPv3 read-only access username
# arguments: username [noauth|auth|priv [OID | -V VIEW [CONTEXT]]]
rouser authPrivUser authpriv -V systemonly
createUser root_auth MD5 lab123123
rouser root_auth auth
```

Esta configuração define um utilizador `root_auth` com autenticação MD5 e password `lab123123`.

✅ Passo 2: Executar consulta SNMPv3 no `host1`

Foi utilizado o comando `snmpget` no `host1` para consultar remotamente o nome da máquina (`sysName`):

```
server1@Server1:~$ snmpget -v3 -u root_auth -a MD5 -A lab123123 -l authNoPriv 192.168.100.11 1.3.6.1.2.1.1.5.0
SNMPv2-MIB::sysName.0 = STRING: client
server1@Server1:~$ _
```

Este comando validou com sucesso a configuração do SNMPv3 e provou que a autenticação está funcional.

✅ Conclusão da Pergunta 1

Com esta série de testes, foi possível compreender na prática o funcionamento do protocolo SNMP nas suas diferentes versões. Utilizando o NET-SNMP, demonstrou-se a consulta e alteração de variáveis, bem como a utilização do modo SNMPv3 com autenticação segura. As comunicações entre os dois servidores foram validadas, provando a correta instalação e configuração do agente SNMP no host monitorizado.

Pergunta 2 – Monitorização com Ferramentas SNMP

Nesta fase, o objetivo é experimentar diferentes ferramentas que recorrem ao SNMP para gerir/monitorizar a rede.

2.1 Instalação e comparação de ferramentas SNMP

No âmbito deste trabalho, foram instaladas e configuradas duas ferramentas de monitorização baseadas em SNMP: **Cacti** e **Zabbix**. O objetivo foi testar e comparar as suas funcionalidades, desempenho e usabilidade na monitorização de sistemas via protocolo SNMP.

Cada ferramenta foi configurada para comunicar com um host remoto (Server2), onde o SNMP foi devidamente ativado e ajustado. As métricas recolhidas incluem **CPU, memória, processos, utilização de disco e load average**, gerando gráficos e dashboards a partir destes dados.

Ambas demonstraram capacidades distintas:

- O **Cacti** mostrou-se eficaz na geração de **gráficos RRD simples e leves**, com configuração básica de polling via cron.
- O **Zabbix**, por outro lado, destacou-se como uma plataforma de **monitorização completa**, com deteção automática de hosts, alertas e visualizações dinâmicas.

Através desta comparação prática, foi possível validar os diferentes pontos fortes e limitações de cada uma, destacando a importância da escolha da ferramenta consoante as necessidades específicas do ambiente a monitorizar.

2. Ambiente de Teste

- **VM1**: Servidor com Cacti e Zabbix instalados
- **VM2**: Dispositivo monitorizado (Ubuntu 22.04 LTS com SNMP configurado)
- **Rede**: 192.168.1.0/24

3. Ferramentas Instaladas

Cacti

- Recolhe dados via SNMP
- Gera gráficos RRDTool com polling a cada 5 minutos
- Requer configuração de cron + timezone + SNMP

Zabbix

- Plataforma de monitorização completa
 - Suporte a SNMP, ICMP, Zabbix Agent, etc.
 - Inclui sistema de alertas, dashboards e discovery
-

4. Configurações Realizadas

4.1 SNMP na VM Monitorizada (VM2)

- Instalação de `snmp` e `snmpd`
- Edição de `/etc/snmp/snmpd.conf` com:

```
# system + hrSystem groups only
view systemonly included .1.3.6.1.2.1.1
view systemonly included .1.3.6.1.2.1.25.1
view all included .1 80

# rocommunity: a SNMPv1/SNMPv2c read-only access community name
# arguments: community [default|hostname|network/bits] [oid | -V view]

# Read-only access to everyone to the systemonly view
rocommunity public default -V all
rocommunity6 public default -V all
rwcommunity private 192.168.100.0/24 view all
```

4.2 PHP e MariaDB

4.3 Cacti

- Instalação via APT

- Criação de crontab para o utilizador `www-data` :

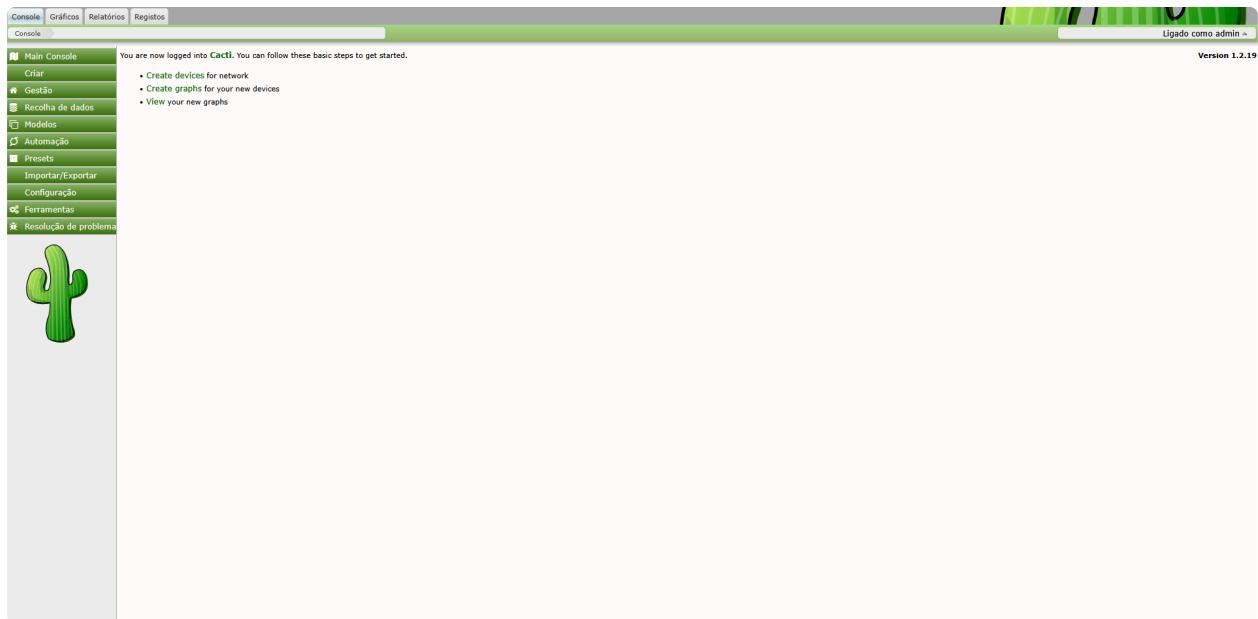
```
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').
#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h  dom mon dow   command
*/5 * * * * php /usr/share/cacti/site/poller.php > /dev/null 2>&1
```

4.4 Zabbix

- Instalação do Zabbix Server e frontend
- Adição do host via IP e aplicação de template `SNMP Device`
- Templates aplicados: `Linux by SNMP`, `Generic by SNMP`, `Network Generic Device by SNMP`
- Verificação dos itens e dados recolhidos no painel de gráficos

5. Exemplos de Gráficos Gerados (Cacti)

Cacti Main Menu



Configuração Server 2

Host2 (192.168.1.11)

SNMP Information

Sistema: Linux Server2 5.15.0-138-generic #148-Ubuntu SMP Fri Mar 14 19:05:48 UTC 2025 x86_64
Uptime: 1346580 (22 dias, 3 horas, 46 minutos)
Nome do servidor: client
Localização: Lab Redes
Contacto: admin@host2.local

*Create New Device
*Create Graphs for this Device
*Re-Index Device
*Enable Device Debug
*Repopulate Poller Cache
*Data Source List
*Lista dos gráficos

Device [edit: Host2]

General Device Options

Descrição: Host2

Hostname: 192.168.1.11

Localização: Nenhum

Poller Association: Main Poller

Device Site Association: Edge

Device Template: Cacti Stats

Number of Collection Threads: 1 Thread (default)

Disable Device: ☐

SNMP Options

SNMP Version: Version 2

SNMP Community String: public

SNMP Port: 161

SNMP Timeout: 1000

Maximum OIDs Per Get Request: 10 OIDs

Bulk Walk Maximum Repetitions: Auto Detect on Re-Index

Availability/Reachability Options

Downed Device Detection: SNMP Uptime

Ping Timeout Value: 400

Ping Retry Count: 1

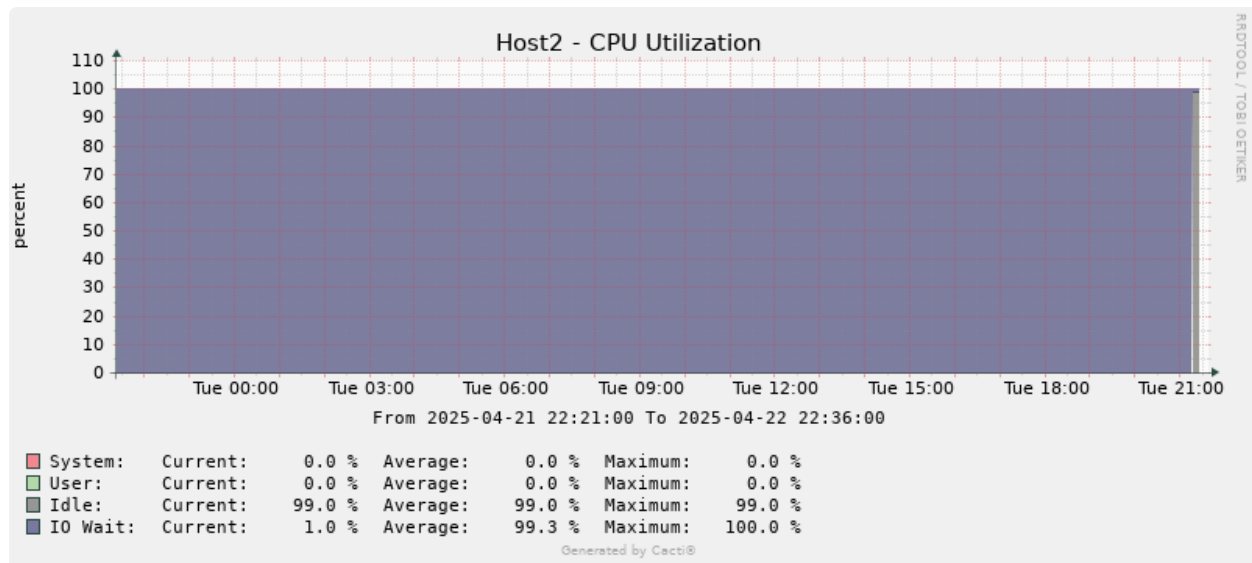
Opções adicionais

Notas:

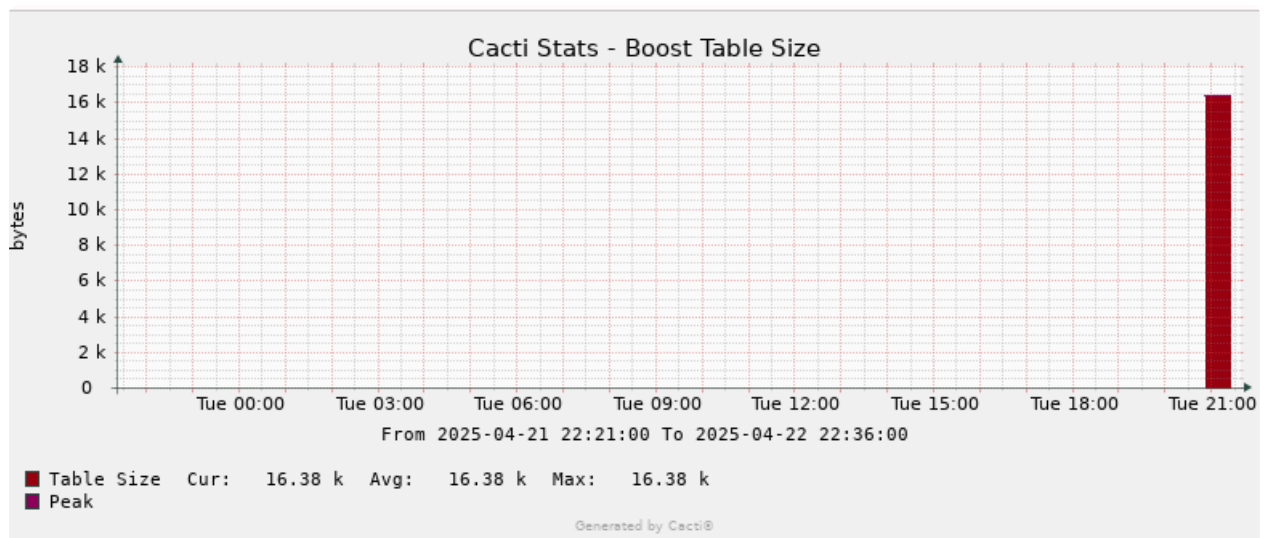
External ID:

CPU Utilization -> Representa a percentagem de uso da CPU (User, System, Idle, IO Wait). Mostra que o SNMP está a reportar

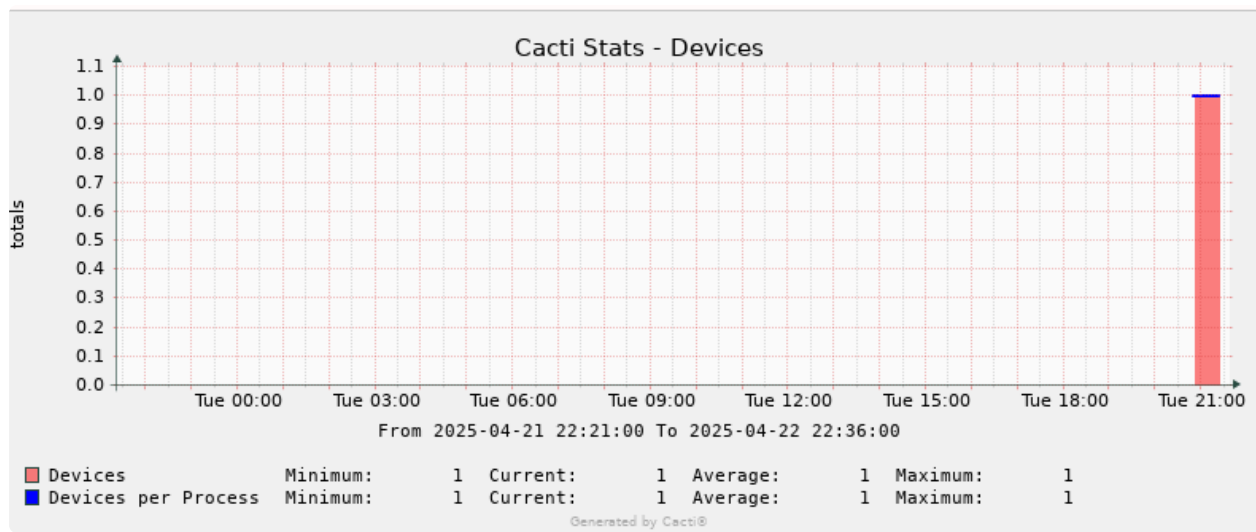
corretamente os dados de carga do sistema.



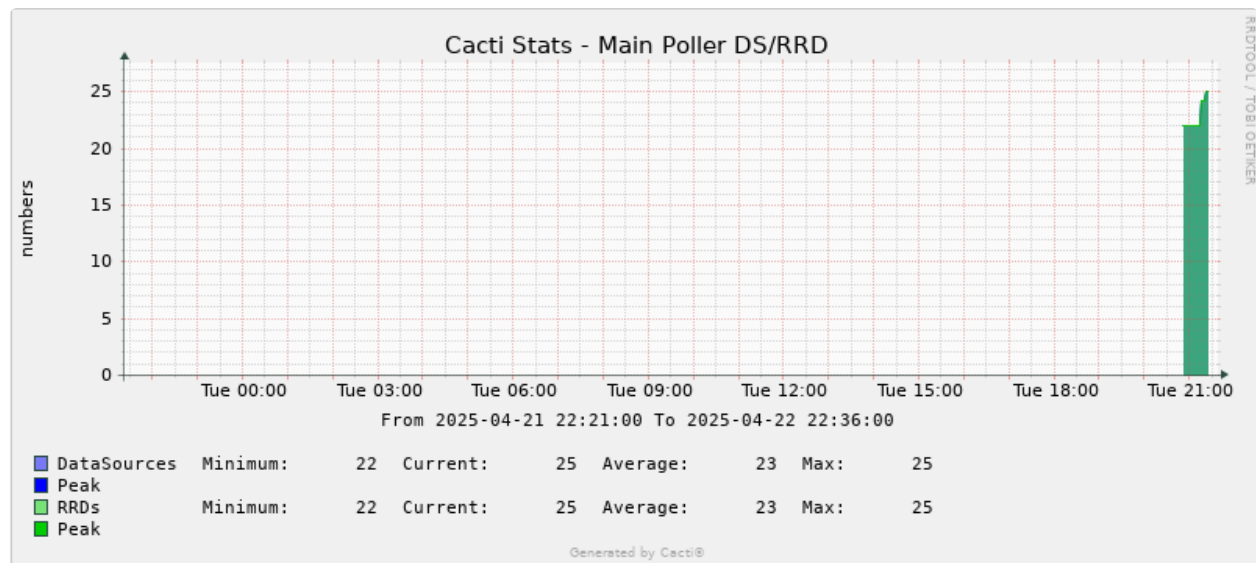
Boost Table Size -> Mostra o tamanho atual da tabela Boost usada pelo Cacti para acelerar a leitura e escrita dos dados de polling.



Devices -> Indica o número de dispositivos monitorizados no sistema e quantos dispositivos por processo estão a ser utilizados.

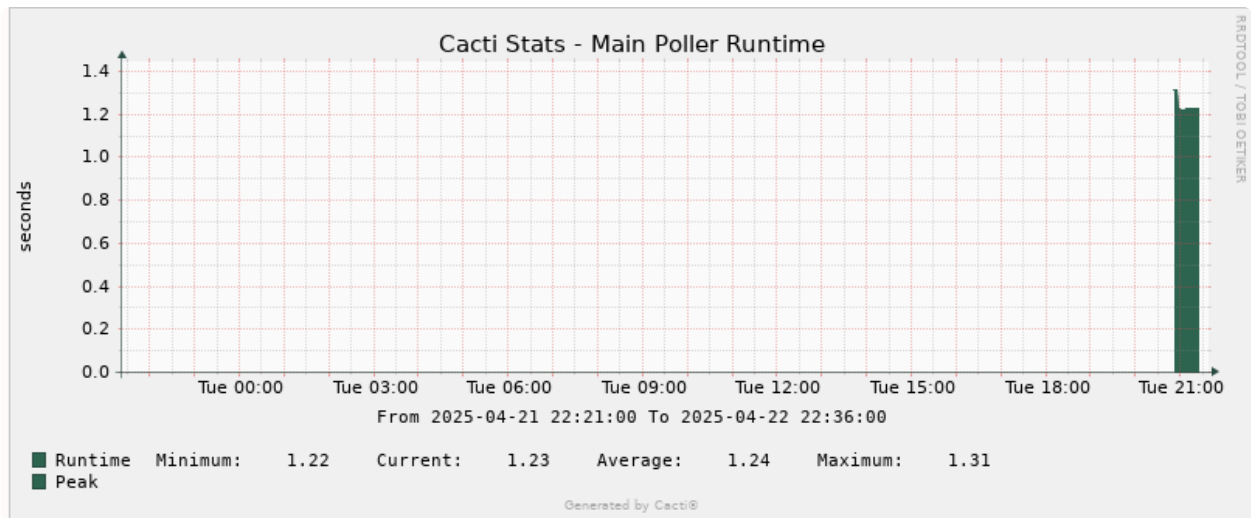


Main Poller DS/RRD -> Estatísticas do Poller principal: número de DataSources e ficheiros RRD em uso, mostrando que o poller está ativo e a funcionar corretamente.

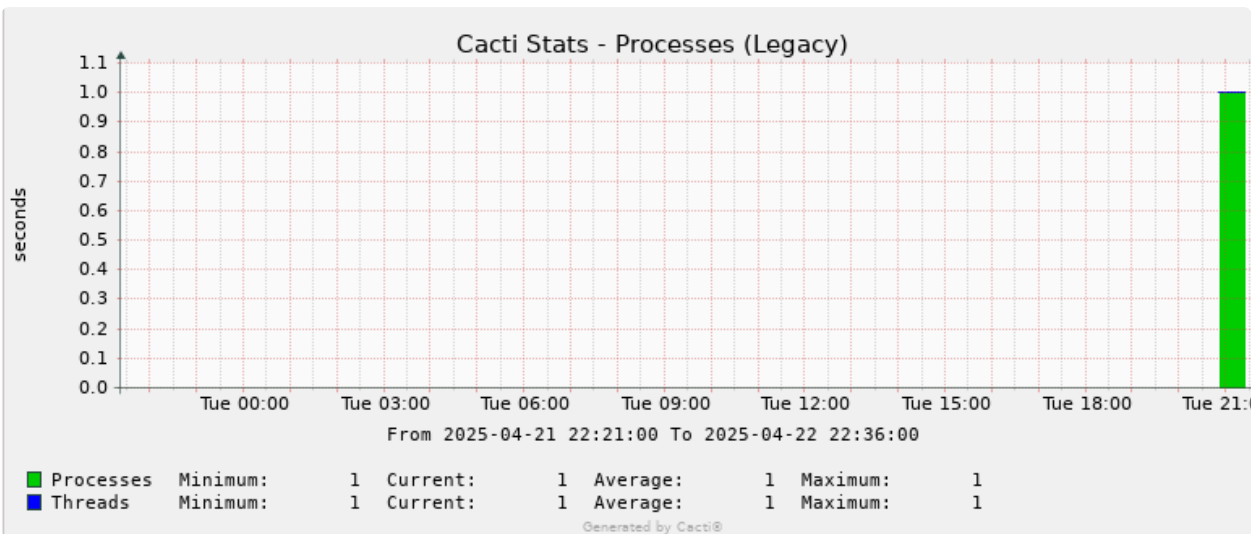


Main Poller Runtime -> Tempo médio de execução do processo de polling. Confirma que o processo está estável, eficiente e concluído

com sucesso.



Processes -> Mostra a quantidade de processos e threads utilizados pelo poller durante as execuções programadas. Indica estabilidade e uso consistente dos recursos.

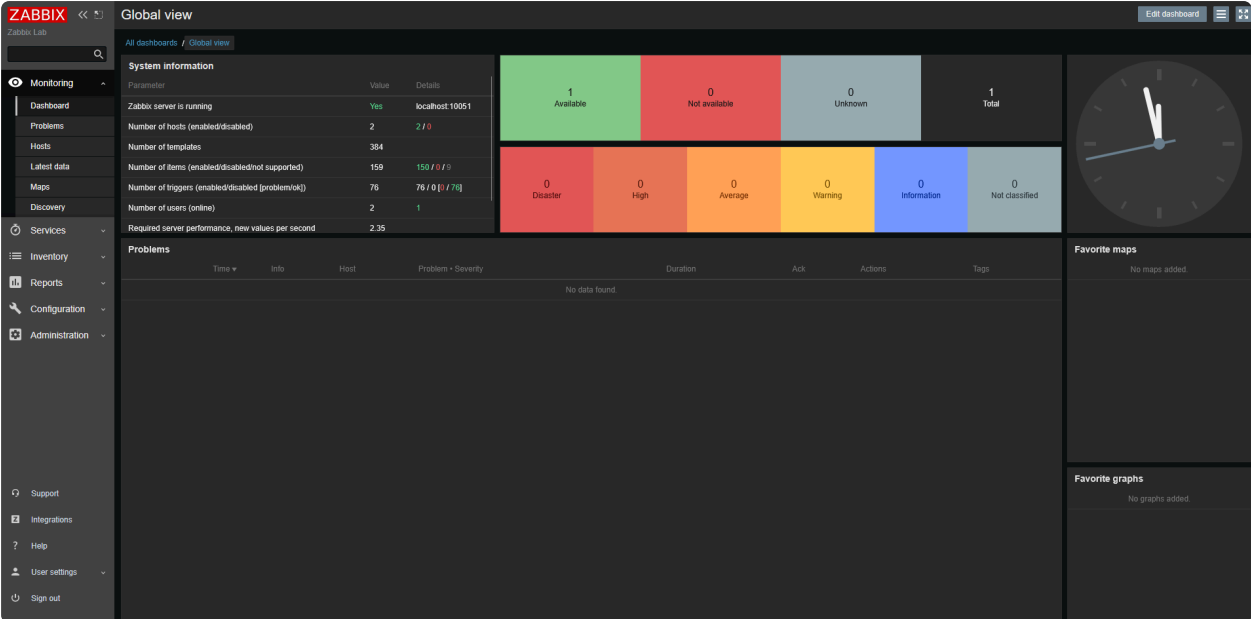


Gráficos gerados pelo cacti (todos)

All 23 Gráficos				
Nome do Gráfico	Dispositivo	Nome da Fonte	Tamanho	
Cacti Stats - Boost Average Row Size	Host2	Modelo Cacti Stats - Boost Average Row Size	200x700	<input type="checkbox"/>
Cacti Stats - Boost Memory	Host2	Modelo Cacti Stats - Boost Memory	200x700	<input type="checkbox"/>
Cacti Stats - Boost Records	Host2	Modelo Cacti Stats - Boost Records	200x700	<input type="checkbox"/>
Cacti Stats - Boost Runtime	Host2	Modelo Cacti Stats - Boost Runtime	200x700	<input type="checkbox"/>
Cacti Stats - Boost Table Size	Host2	Modelo Cacti Stats - Boost Table Size	200x700	<input type="checkbox"/>
Cacti Stats - Boost Timing Detail	Host2	Modelo Cacti Stats - Boost Timing Detail	200x700	<input type="checkbox"/>
Cacti Stats - Boost Updates	Host2	Modelo Cacti Stats - Boost Updates	200x700	<input type="checkbox"/>
Cacti Stats - Devices	Host2	Modelo Cacti Stats - Devices	200x700	<input type="checkbox"/>
Cacti Stats - Export Duration (Legacy)	Host2	Modelo Cacti Stats - Export Duration (Legacy)	200x700	<input type="checkbox"/>
Cacti Stats - Export Graphs (Legacy)	Host2	Modelo Cacti Stats - Export Graphs (Legacy)	200x700	<input type="checkbox"/>
Cacti Stats - Main Poller DS/RRD	Host2	Modelo Cacti Stats - Main Poller DS/RRD	200x700	<input type="checkbox"/>
Cacti Stats - Main Poller Runtime	Host2	Modelo Cacti Stats - Main Poller Runtime	200x700	<input type="checkbox"/>
Cacti Stats - Processes (Legacy)	Host2	Modelo Cacti Stats - Processes (Legacy)	200x700	<input type="checkbox"/>
Cacti Stats - Recache (Legacy)	Host2	Modelo Cacti Stats - Recache (Legacy)	200x700	<input type="checkbox"/>
Cacti Stats - Syslog Activity	Host2	Modelo Cacti Stats - Syslog Activity	200x700	<input type="checkbox"/>
Cacti Stats - Syslog Alerts/Reports	Host2	Modelo Cacti Stats - Syslog Alerts/Reports	200x700	<input type="checkbox"/>
Cacti Stats - Syslog Runtime	Host2	Modelo Cacti Stats - Syslog Runtime	200x700	<input type="checkbox"/>
Cacti Stats - Thold Runtime	Host2	Modelo Cacti Stats - Thold Runtime	200x700	<input type="checkbox"/>
Cacti Stats - Total Poller Items	Host2	Modelo Cacti Stats - Total Poller Items	200x700	<input type="checkbox"/>
Cacti Stats - User Logins	Host2	Modelo Cacti Stats - User Logins	200x700	<input type="checkbox"/>
Cacti Stats - User Sessions	Host2	Modelo Cacti Stats - User Sessions	200x700	<input type="checkbox"/>
Cacti Stats - User Types	Host2	Modelo Cacti Stats - User Types	200x700	<input type="checkbox"/>
Host2 - CPU Utilization	Host2	Modelo Net-SNMP - CPU Utilization	200x700	<input type="checkbox"/>

6. Exemplos de Gráficos Gerados (Zabbix)

Zabbix Main Menu



Zabbix Configuração do Server2

Name	Interface	Availability	Tags	Status	Latest data	Problems	Graphs	Dashboards	Web
Server2	192.168.1.11:161	OK	class:os target:linux	Enabled	Latest data 3s	Problems	Graphs 7	Dashboards 2	Web

Host

Host

IPMI

Tags

Macros

Inventory

Encryption

Value mapping

* Host name

Server2

Visible name

Server2

Templates

Name

Linux by SNMP

type here to search

Select

Action

Unlink

Unlink and clear

* Groups

Linux servers

type here to search

Select

Interfaces

Type

SNMP

IP address

192.168.1.11

DNS name

Connect to

IP

DNS

Port

161

Default

Remove

* SNMP version

SNMPv2

* SNMP community

public

Use bulk requests

Add

Description

Monitored by proxy

(no proxy)

Enabled

Update

Clone

Full clone

Delete

Cancel

CPU Jumps -> Este gráfico mostra as interrupções e trocas de contexto por segundo. Os valores estão dentro do esperado, indicando estabilidade no uso da CPU.

Server2: Linux: CPU jumps

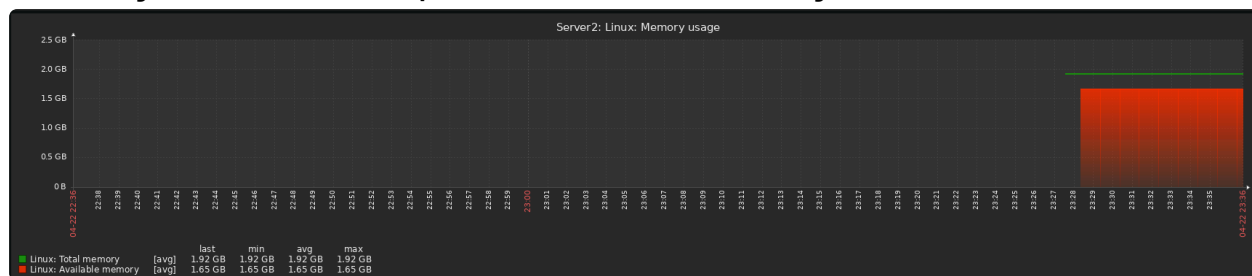
	last	min	avg	max
Linux: Context switches per second [avg]	127.9643	115.3945	119.3592	127.9643
Linux: Interrupts per second [avg]	441.8315	430.5167	440.6545	448.0696

CPU Usage -> O uso da CPU está próximo de 0%, o que é comum em servidores sem carga ativa. Indica que o agente está a reportar corretamente dados do sistema.

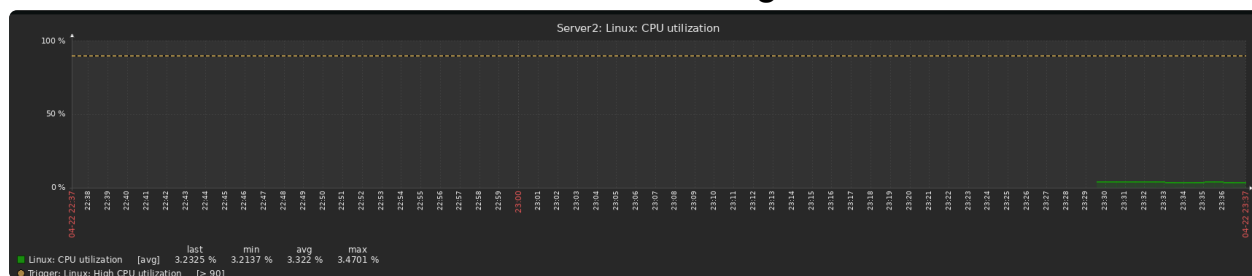
Server2: Linux: CPU usage

	last	min	avg	max
Linux: CPU guest nice time [avg]	0 %	0 %	0 %	0 %
Linux: CPU guest time [avg]	0 %	0 %	0 %	0 %
Linux: CPU softirq time [avg]	0.08325 %	0.03329 %	0.05708 %	0.09999 %
Linux: CPU interrupt time [avg]	0 %	0 %	0 %	0 %
Linux: CPU steal time [avg]	0 %	0 %	0 %	0 %
Linux: CPU iowait time [avg]	0 %	0 %	0.004756 %	0.01665 %
Linux: CPU nice time [avg]	0 %	0 %	0 %	0 %
Linux: CPU user time [avg]	0.08325 %	0 %	0.02378 %	0.08325 %
Linux: CPU system time [avg]	0.08325 %	0.0333 %	0.05946 %	0.08325 %

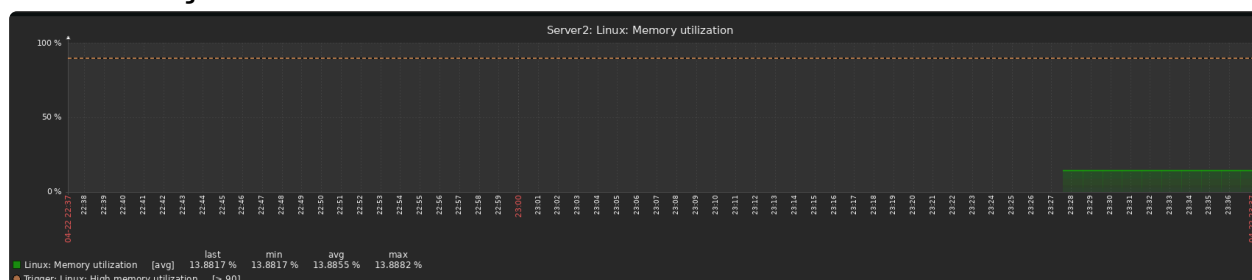
Memory Usage -> Representa a memória total e disponível. A diferença entre ambas permite inferir a utilização real da RAM.



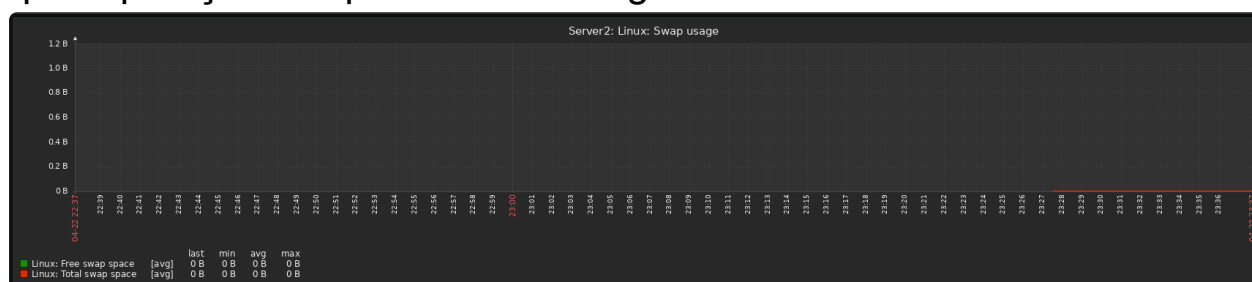
CPU Utilization -> Apresenta a percentagem de utilização global da CPU. Os valores baixos indicam baixa carga no sistema monitorizado.



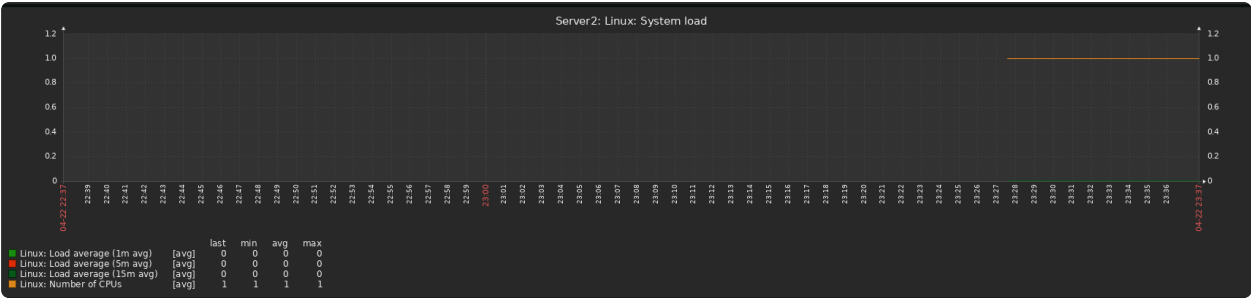
Memory Utilization -> Mostra a percentagem de utilização de memória. Neste caso, cerca de 14%, valor estável para um sistema com serviços básicos.



Swap Usage -> O gráfico mostra 0B de espaço swap. Isso significa que a partição swap não está configurada ou não está a ser usada.



System Load ->Indica o load average do sistema nos últimos 1, 5 e 15 minutos. O valor 0 mostra que o sistema não está sobrecarregado.



7. Comparação Cacti vs Zabbix

Funcionalidade	Cacti	Zabbix
SNMP	Suporte Completo	Suporte Completo
Gráficos	Basedos em RRDTool	Internos, dinâmicos
Alertas	Limitados (Via plugins)	Sim, completos
Descoberta de Hosts	Manual ou semi-auto	Automática
Facilidade de uso	Intuitivo, simples	Mais complexo
Performance	Leve	Mais pesado

8. Conclusão

Ambas as ferramentas têm o seu valor. O **Cacti** é excelente para ambientes simples onde é essencial ter gráficos históricos de desempenho, enquanto o **Zabbix** é mais indicado para monitorização abrangente com alertas e dashboards.

2.2 Adição de Objetos de Monitorização com Cacti

No seguimento da instalação do **Cacti**, foram adicionados diferentes objetos de monitorização com base em SNMP e ICMP, de forma a ampliar a visibilidade sobre o estado da rede e da infraestrutura.

✅ Interfaces de Rede (eth0)

Foi utilizado o menu `Create Graphs for this Host` para adicionar gráficos relativos à interface `eth0` do servidor monitorizado (Server2). Os gráficos criados mostram:

- **Tráfego de entrada (Inbound)**
- **Tráfego de saída (Outbound)**

23) Net-SNMP - Combined SCSI Disk I/O

Is Being Graphed (Editar)

Estes permitem acompanhar o volume de dados em tempo real, identificando padrões de utilização e possíveis gargalos de rede.

✅ Tempo de Atraso (Ping) para o IP 8.8.8.8

Para medir a latência de rede externa, foi criado um novo dispositivo com o IP `8.8.8.8` (DNS público do Google) e configurado o gráfico do tipo `Unix - Ping Latency`. Este gráfico apresenta:

Ping Google (8.8.8.8)
Ping Results
 ICMP Ping Success (17.171 ms)

*Create New Device
 *Create Graphs for this Device
 *Re-Index Device
 *Enable Device Debug
 *Repopulate Poller Cache
 *Data Source List
 *Lista dos gráficos

Device [edit: Ping Google]

General Device Options

Descrição ⓘ Ping Google
 Hostname ⓘ 8.8.8.8
 Localização ⓘ Nenhum
 Poller Association ⓘ Main Poller
 Device Site Association ⓘ Edge
 Device Template ⓘ Local Linux Machine
 Number of Collection Threads ⓘ 1 Thread (default)
 Disable Device ⓘ ☐

SNMP Options

SNMP Version ⓘ Not In Use
 Bulk Walk Maximum Repetitions ⓘ Auto Detect on Re-Index

Availability/Reachability Options

Downed Device Detection ⓘ Ping
 Ping Method ⓘ ICMP Ping
 Ping Timeout Value ⓘ 400
 Ping Retry Count ⓘ 1

Opções adicionais

Notas ⓘ

External ID ⓘ

Associated Graph Templates

Graph Template Name	Estado
1) Linux - Memory Usage	Is Being Graphed (Editar)
2) Unix - Load Average	Is Being Graphed (Editar)

- Tempo médio de resposta (em milissegundos)
- Estabilidade da ligação
- Detecção de perda de pacotes ou variações abruptas na latência

4) Unix - Ping Latency Is Being Graphed (Editar)

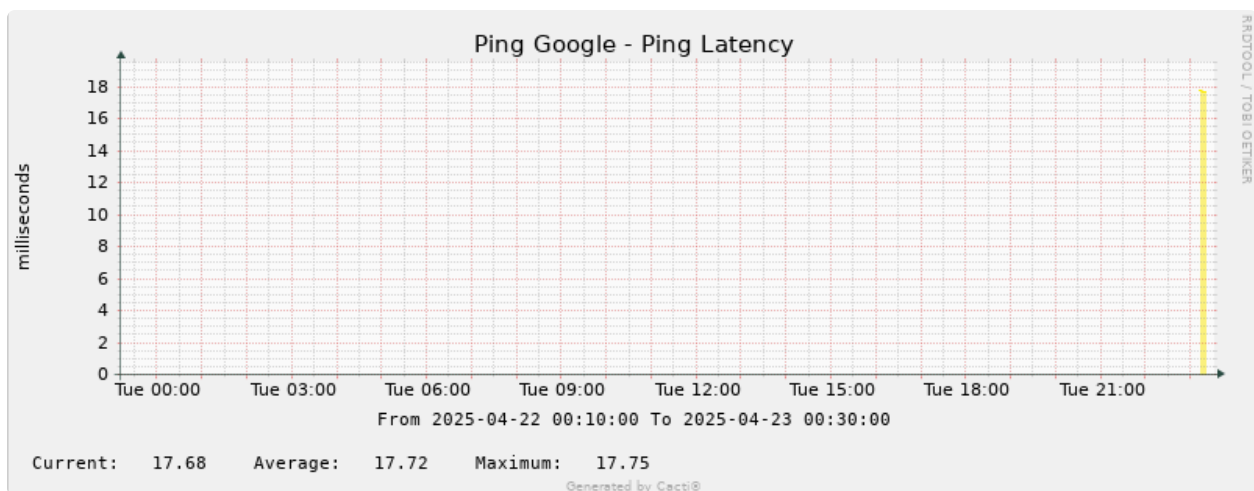
Ambos os objetos começaram a apresentar dados após alguns ciclos de polling, validando a sua correta configuração e funcionamento.

Durante a configuração das ferramentas de monitorização, foram adicionados diferentes objetos para recolha de dados relevantes:

◆ Monitorização da Latência (Ping Google)

Através do Cacti, foi configurado um template de **ping externo ao IP 8.8.8.8 (Google DNS)**. O objetivo é avaliar o tempo de resposta da ligação externa à Internet.

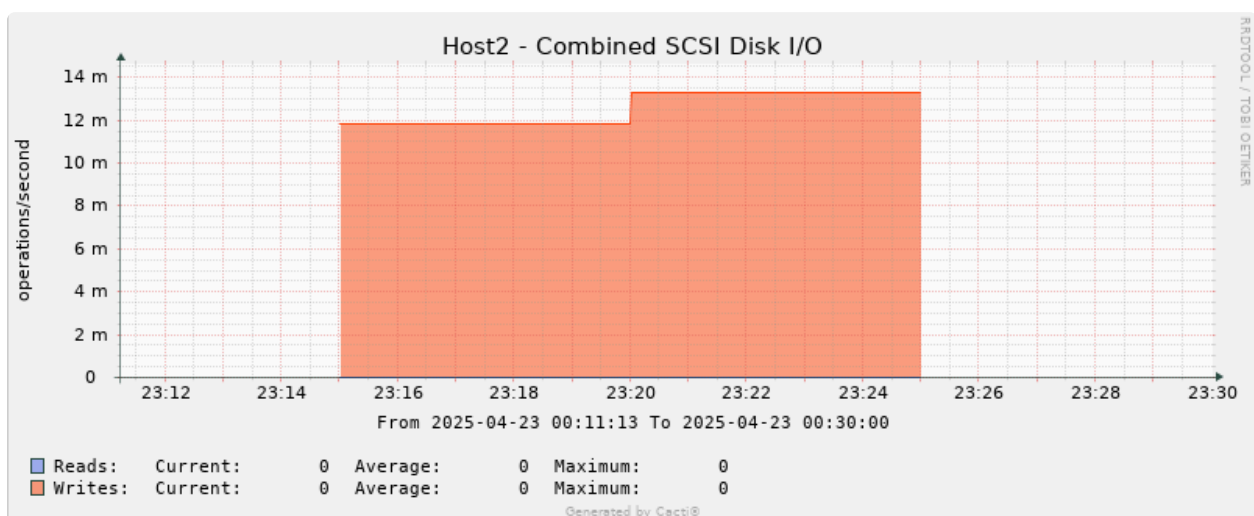
O gráfico mostra uma média estável de ~17ms, o que demonstra uma ligação sem instabilidade significativa.



◆ Monitorização de Disco - I/O combinado

Foi também adicionado o gráfico **SCSI Disk I/O**, que mostra o número de operações de leitura e escrita por segundo no disco do servidor monitorizado.

Verificou-se atividade de escrita superior a 10m ops/segundo durante alguns minutos, sem operações de leitura. Isso reflete que o sistema está a gravar dados ativamente.



✓ Conclusão da Secção 2.2

A adição destes objetos permitiu validar a comunicação SNMP com o host monitorizado e testar métricas essenciais como latência externa, tráfego de rede e desempenho de disco. Todos os gráficos apresentaram dados consistentes, demonstrando que o **Cacti está funcional e corretamente configurado.**