ZABBIX 2023 Conference LATIN AMERICA

AUTOMATIZANDO TAREFAS DIÁRIAS COM ZABBIX

HERNANDES MARTINS

Hernandes Martins

- Atuando com Zabbix há mais de 12 anos
- ZCS-ZCP-ZCE-ZCT(Trainner)
- Colaborador nas redes sociais, Zabbix Fórum Oficial nos canais da comunidade Zabbix Brasil e internacionais.
- Consultor Especialista Zabbix na Lunio(Zabbix Partner)

Lunio 10 ANOS

ZABBIX 2023 Conference LATIN AMERICA

ZABBIX CERTIFIED PARTNER

Lunio® 10 ANOS

Somos **Partner da Zabbix SIA** e responsáveis por sustentar os ambientes de monitoramento complexos e desafiadores.

A **Lunio Cloud Solutions** é uma empresa especializada em Monitoramento, serviços de TI, Cloud Computing e Segurança, criada para possibilitar aos nossos clientes a utilização da tecnologia a favor de seus negócios.

Com mais de 10 anos de atuação no mercado, situada no polo de tecnologia do Porto Digital em Recife (PE).

Agenda

1. O poder da automação com Zabbix
2. Tarefas diarias com Zabbix
3. Como automatizar suas tarefas
4. Cases e exemplos de uso

ZABBIX 2023 Conference LATIN AMERICA

Lunio 10 ANOS

1.

O poder da automação do Zabbix

É fundamental conhecer o potencial que o Zabbix oferece em relação a automação.

Principais automações do Zabbix

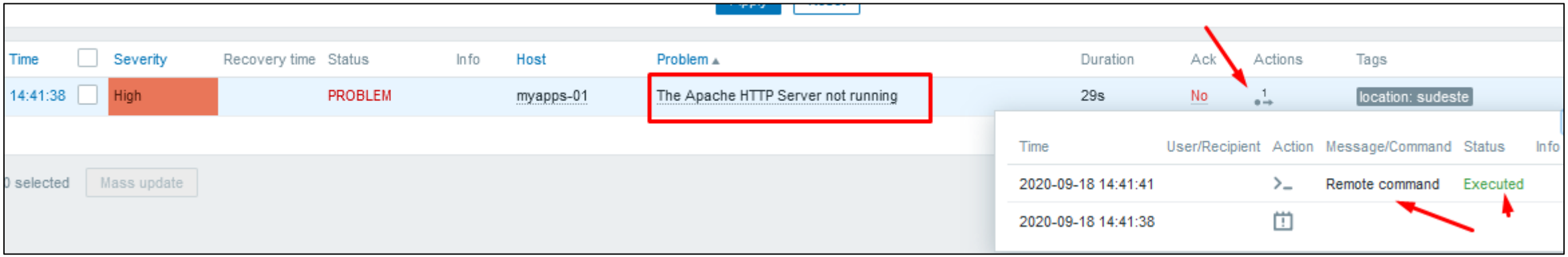
1. Automação de alertas

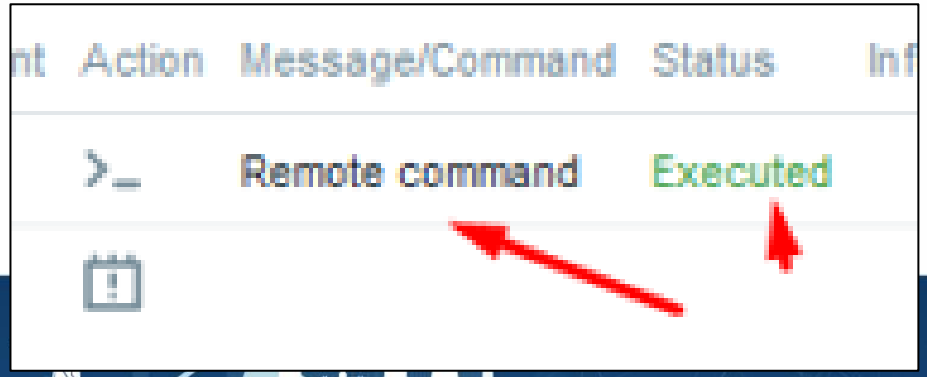
- É possível configurar alertas com base nos eventos de monitoramento.
- Personalizar alertas e definir ações automáticas para lidar com eles.

Not classified	Information	Warning	Average	High	Disaster

1. Automação de alertas

- Execução de scripts ou reinicialização de serviços com comando remoto





1. Automação de alertas

- Execução de integrações nativas e escalonamento de notificações

Media types

Name ▲	Type
Brevis.one	Webhook
Discord	Webhook
Email	Email
Email (HTML)	Email
Event-Driven Ansible	Webhook
Express.ms	Webhook
Github	Webhook
GLPi	Webhook
Gmail	Email
Gmail relay	Email

iLert	Webhook
iTop	Webhook
Jira	Webhook
Jira ServiceDesk	Webhook
Jira with CustomFields	Webhook
Line	Webhook
ManageEngine ServiceDesk	Webhook
Mattermost	Webhook
MS Teams	Webhook
Office365	Email
Office365 relay	Email
Opsgenie	Webhook
OTRS	Webhook
OTRS CE	Webhook

PagerDuty	Webhook
Pushover	Webhook
Redmine	Webhook
Rocket.Chat	Webhook
ServiceNow	Webhook
SIGNL4	Webhook
Slack	Webhook
SMS	SMS
SolarWinds Service Desk	Webhook
SysAid	Webhook
Telegram	Webhook
TOPdesk	Webhook
VictorOps	Webhook
Zammad	Webhook
Zendesk	Webhook

1. Automação de alertas

- Requer configuração de triggers
- Requer configuração de ações
- Requer configuração do agente para permitir execução de comandos remotos

2. Automações agendadas

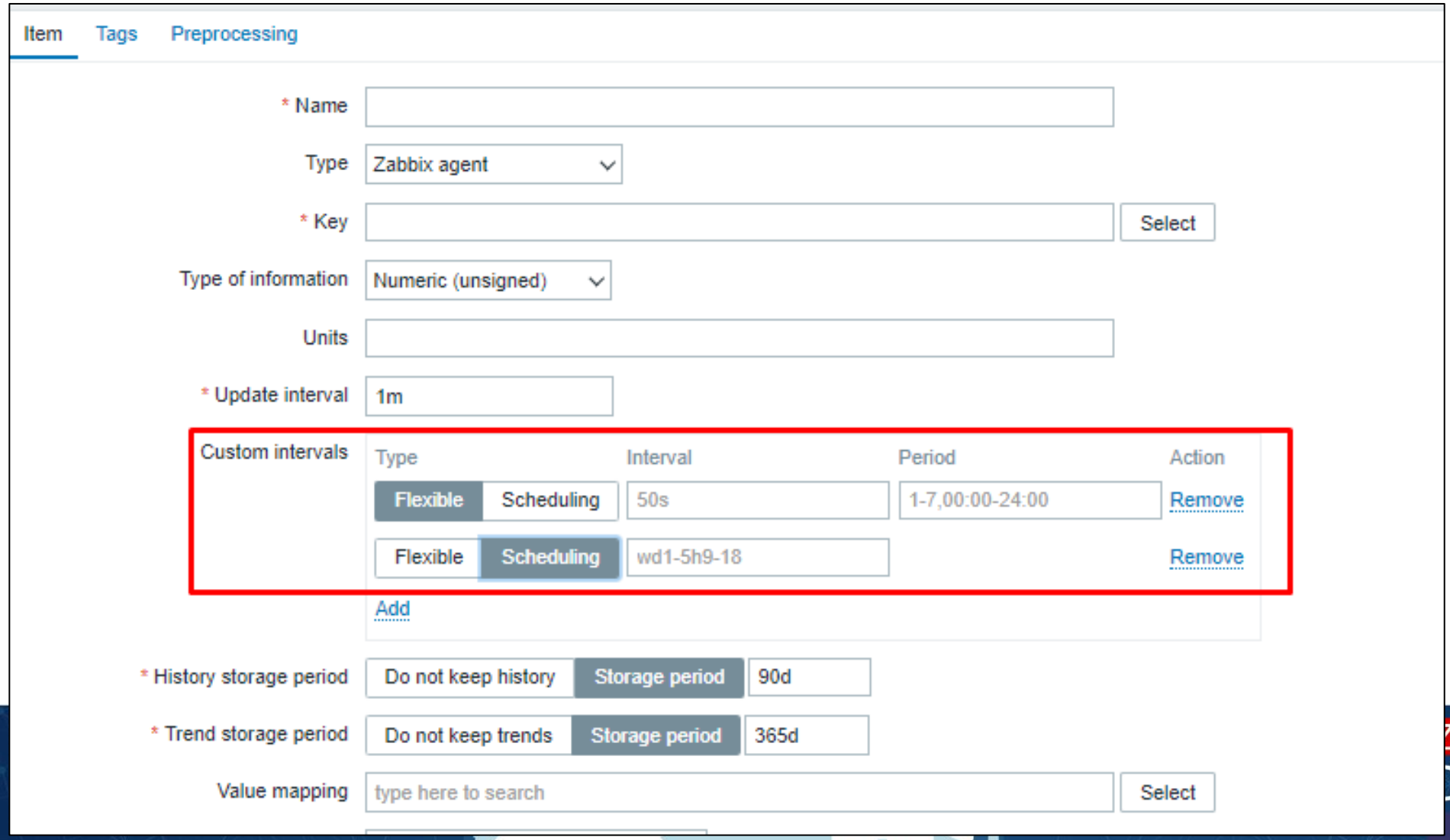
1) Coletas de dados agendadas

2) Período de manutenção

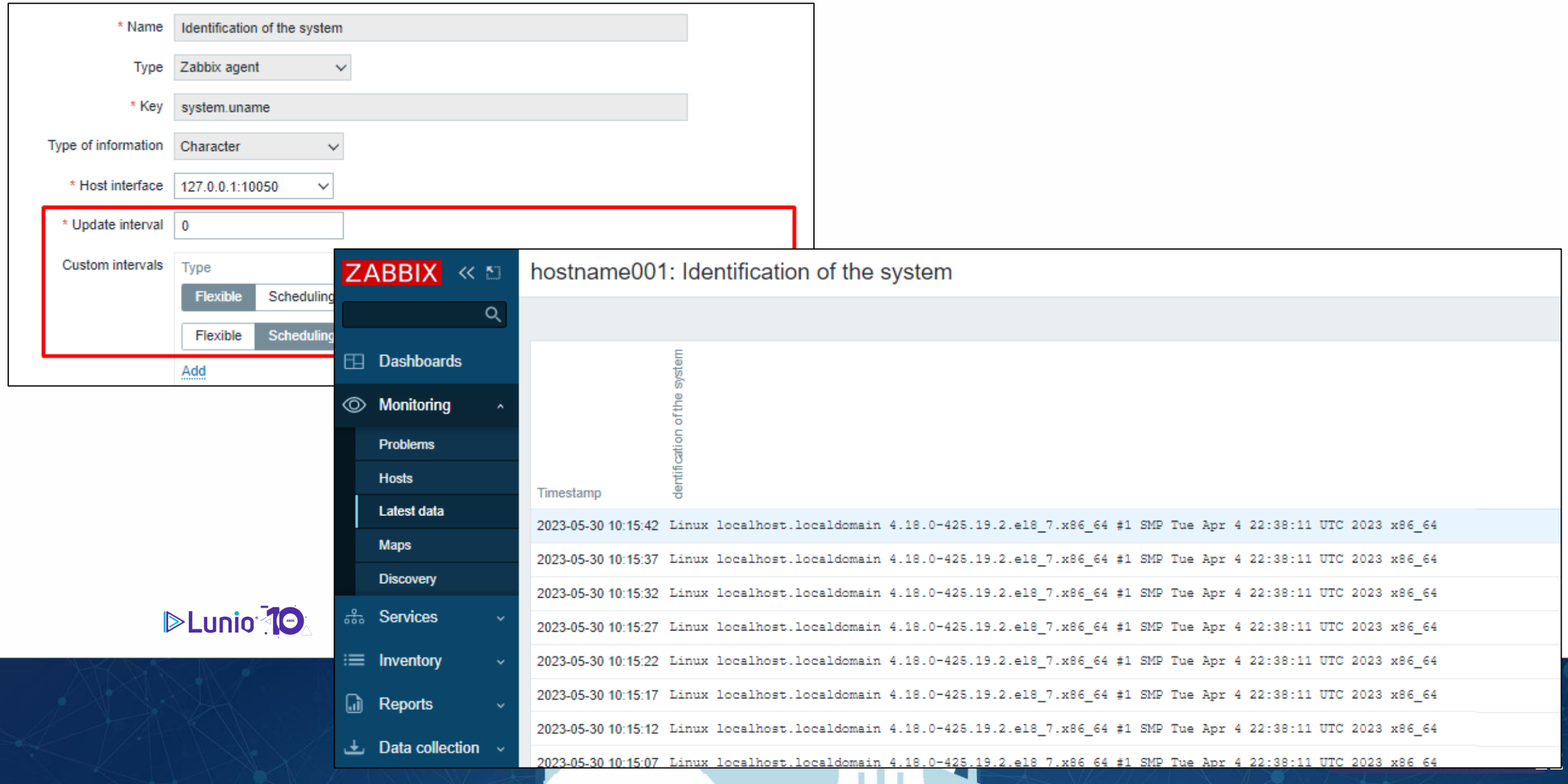
3) Envio de relatórios

2. Coletas de dados agendadas

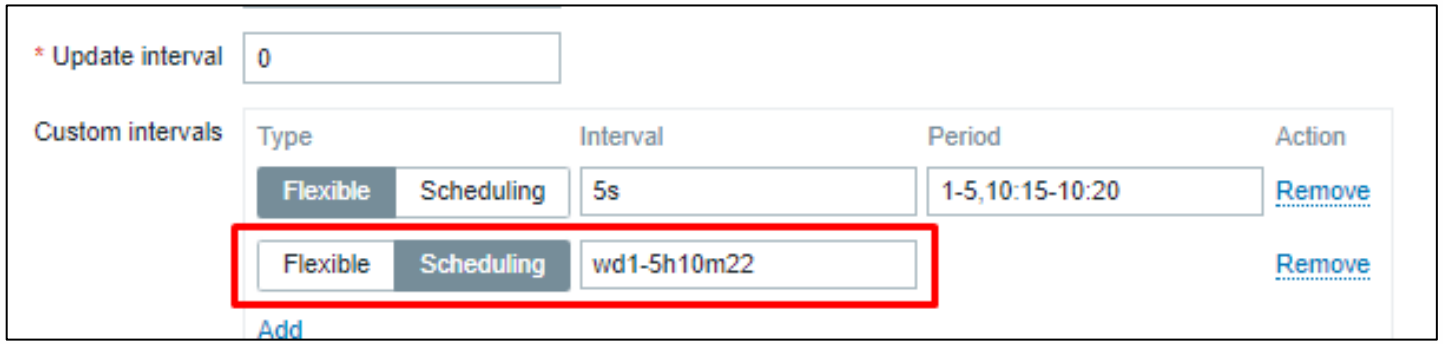
- permite agendar um horário ou intervalo de coleta específico para a coleta de dados.

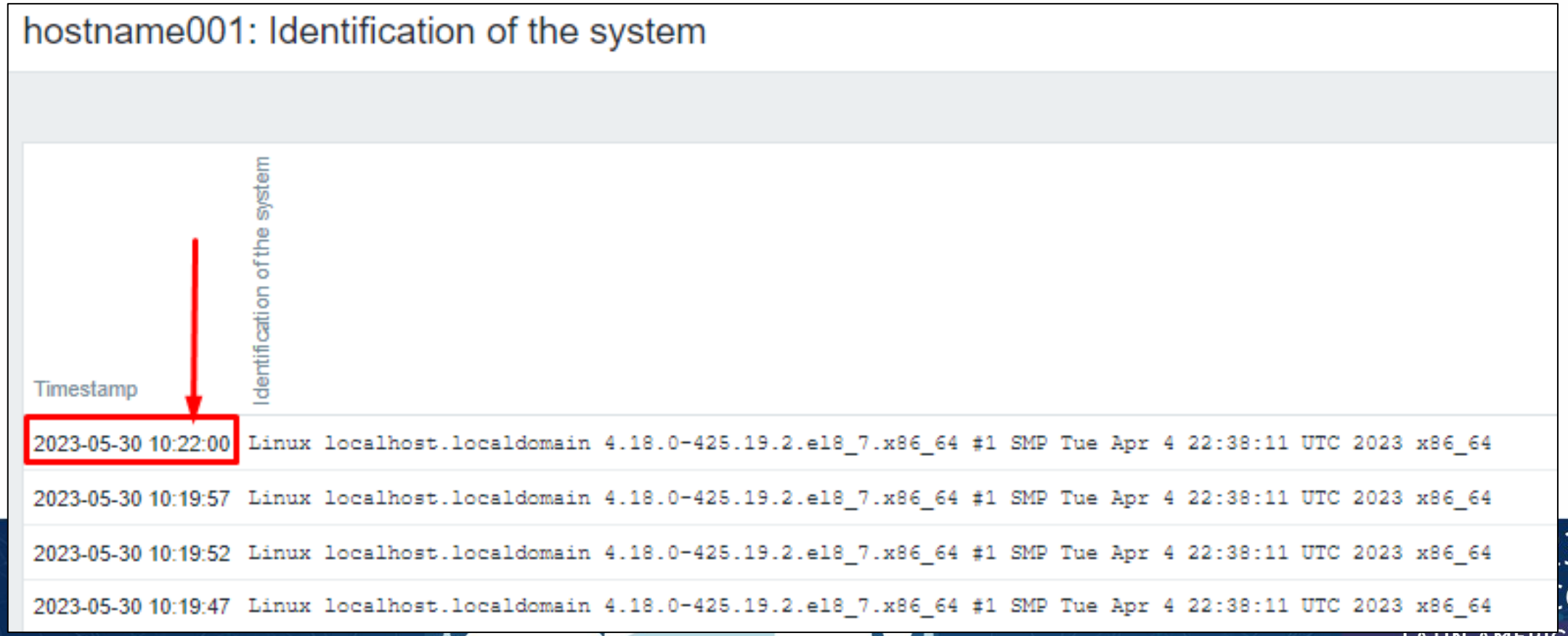


2. Coletas de dados agendadas



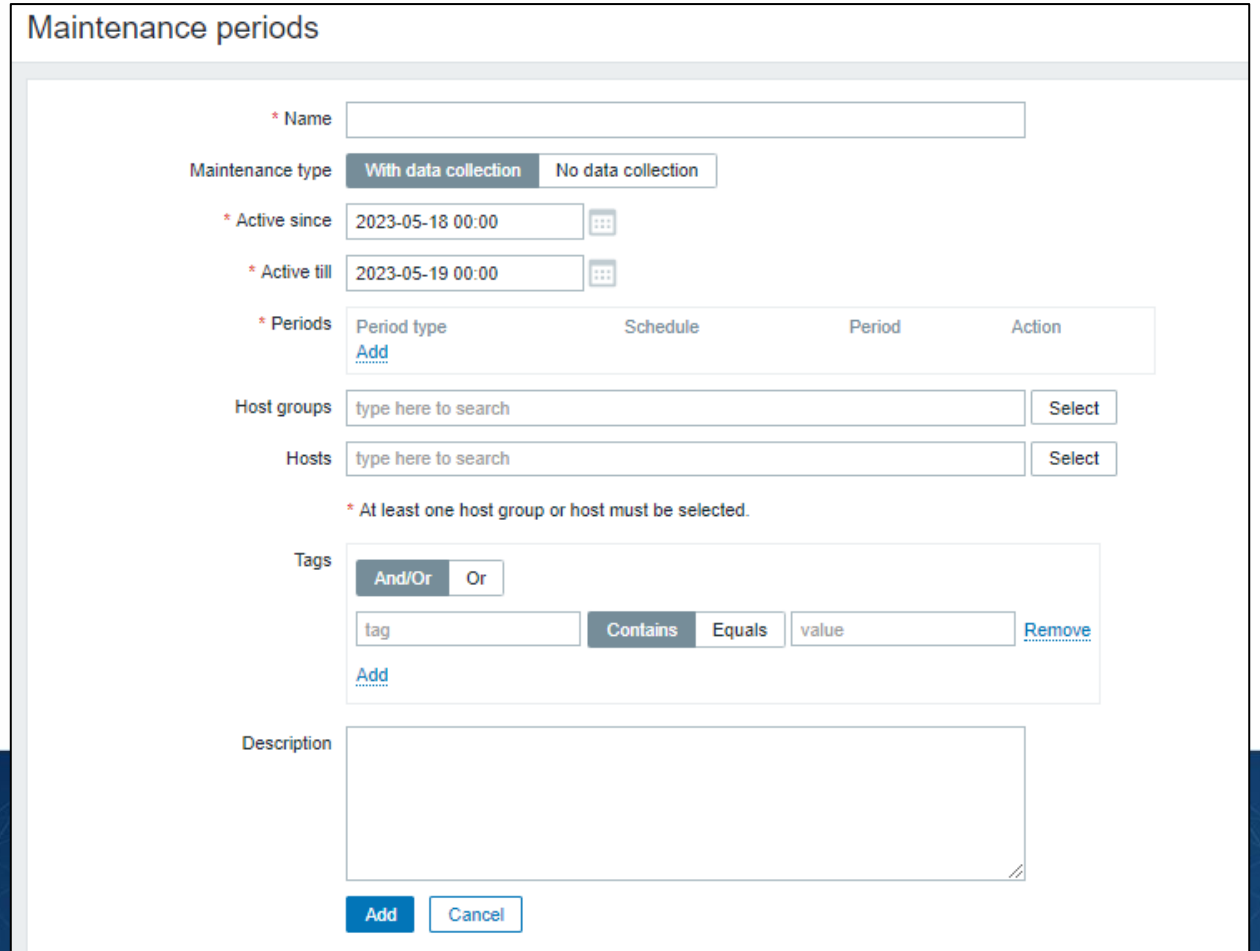
2. Coletas de dados agendadas



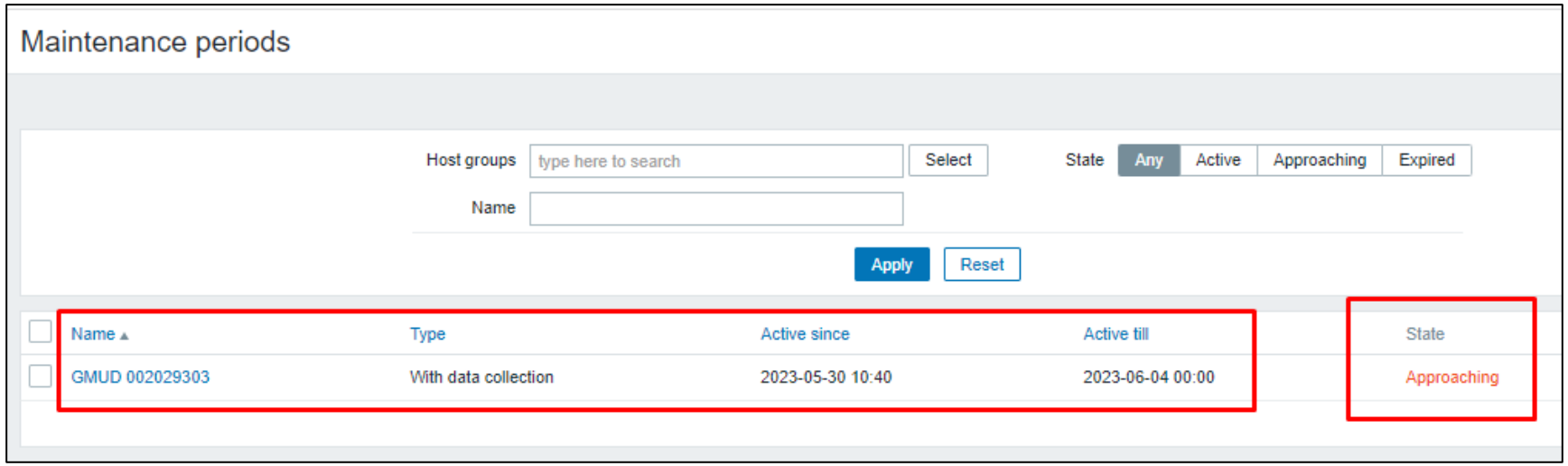


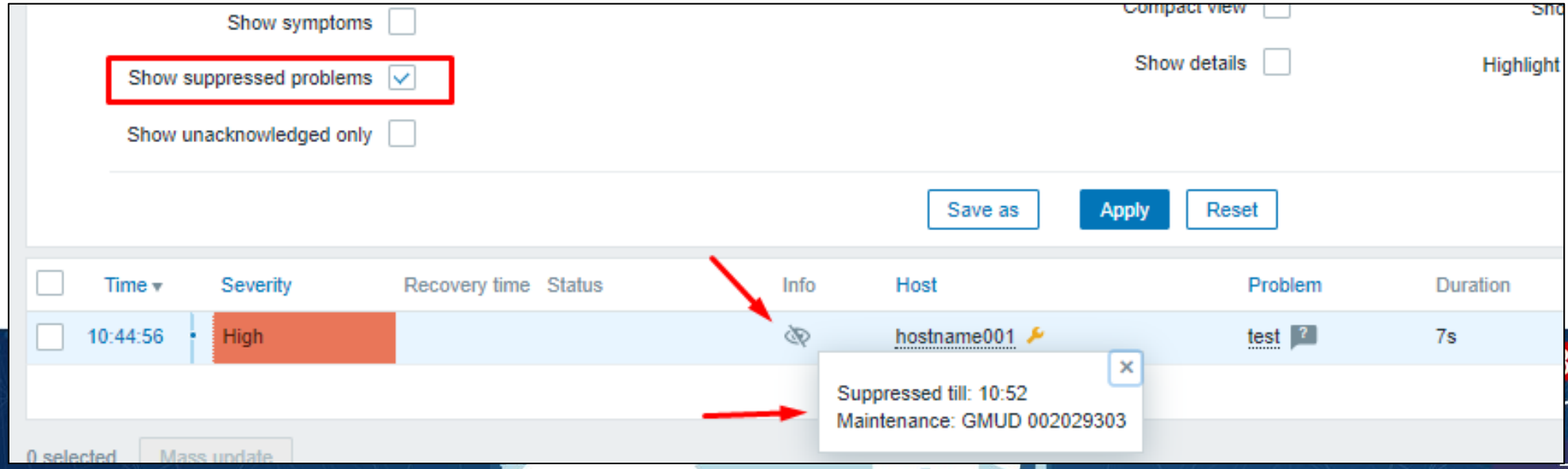
2. Período de manutenção

- É possível configurar períodos de manutenção automáticos no Zabbix.
- Evitando notificações desnecessárias durante períodos de manutenção planejada.



2. Período de manutenção



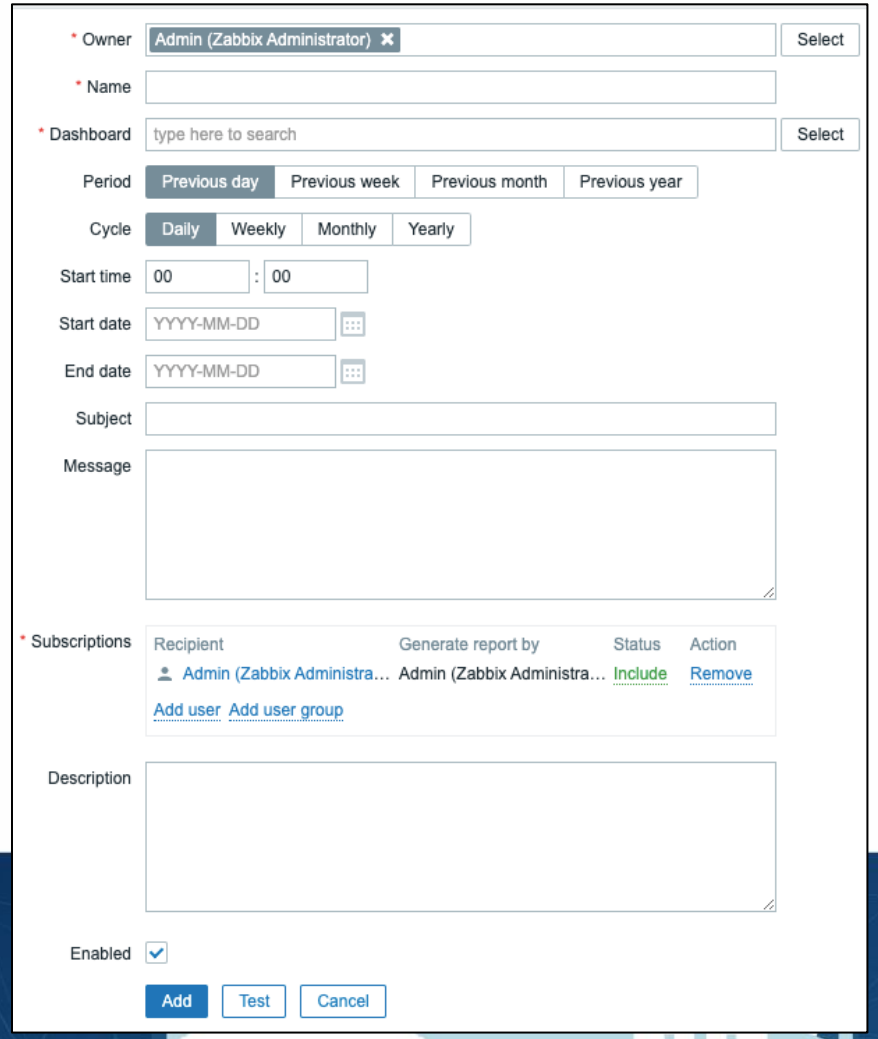


2. Automação de relatórios

- Geração relatórios com base nos dados coletados pelo Zabbix.
- Agendamento de relatórios para serem gerados em intervalos regulares.
- Envio de relatórios por e-mail para destinatários específicos.

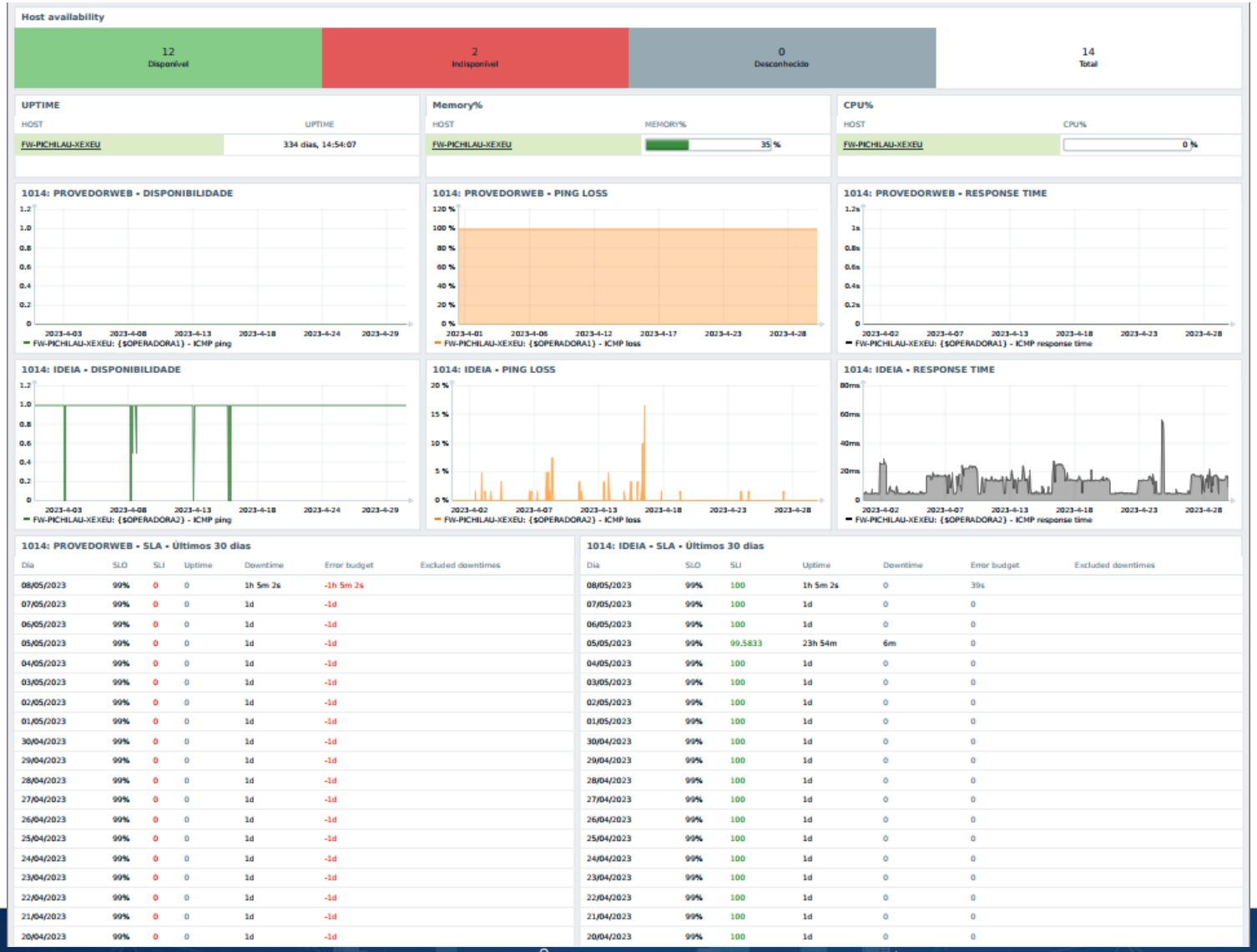
2. Automação de relatórios

- Exemplo de automação de relatórios nativos



2. Automação de relatórios

2. Automação de relatórios



3. Automação com Network Discovery

- Com o Zabbix pode descobrir automaticamente novos dispositivos em sua rede.
- Configurando regras de descoberta e ações automáticas para adicionar hosts ao monitoramento.

Network discovery

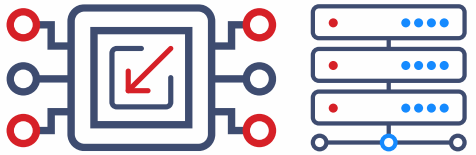
Condição

Service	= SSH
Device status	= up
Uptime	= 3600s

Ação

Adiciona Grupos
AtribuiTemplates

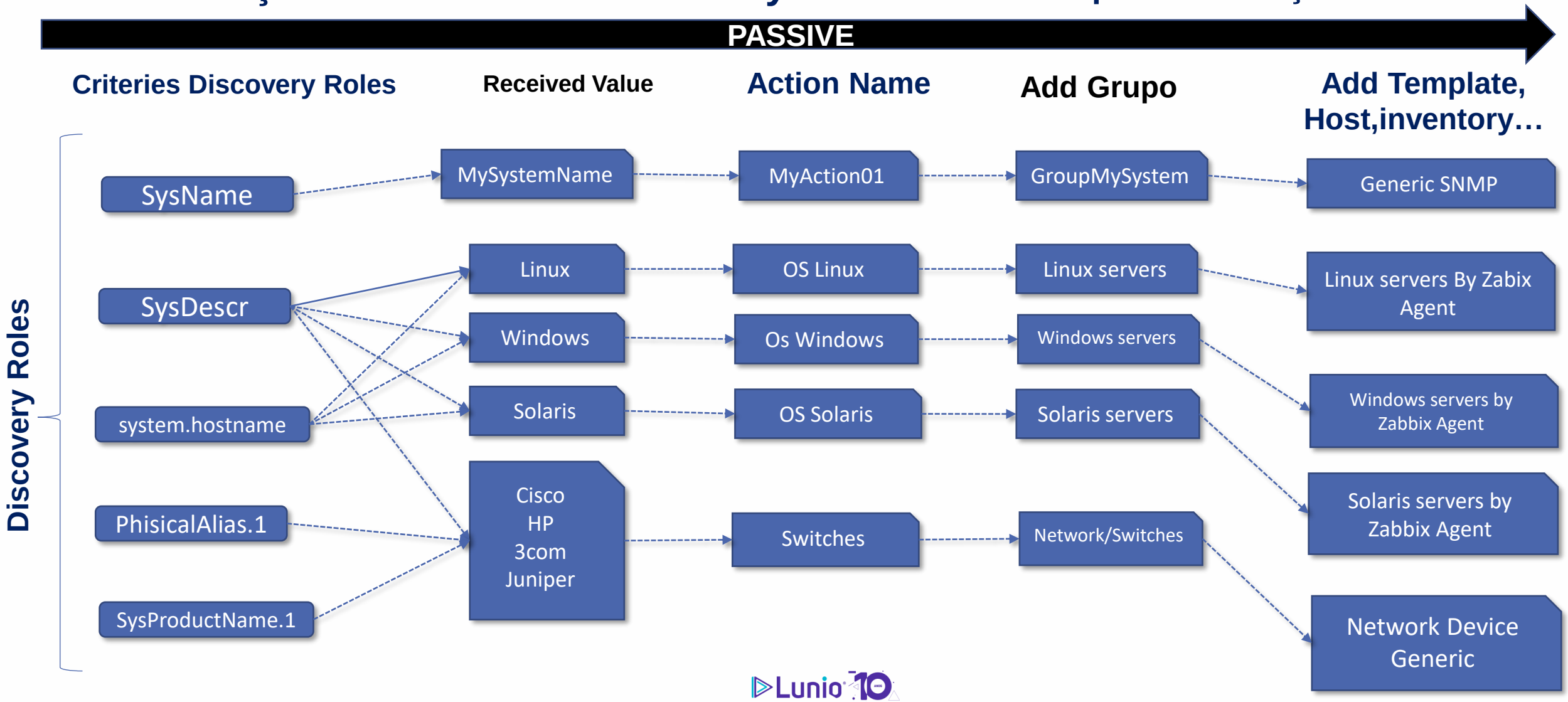
Network scan



Novos dispositivos

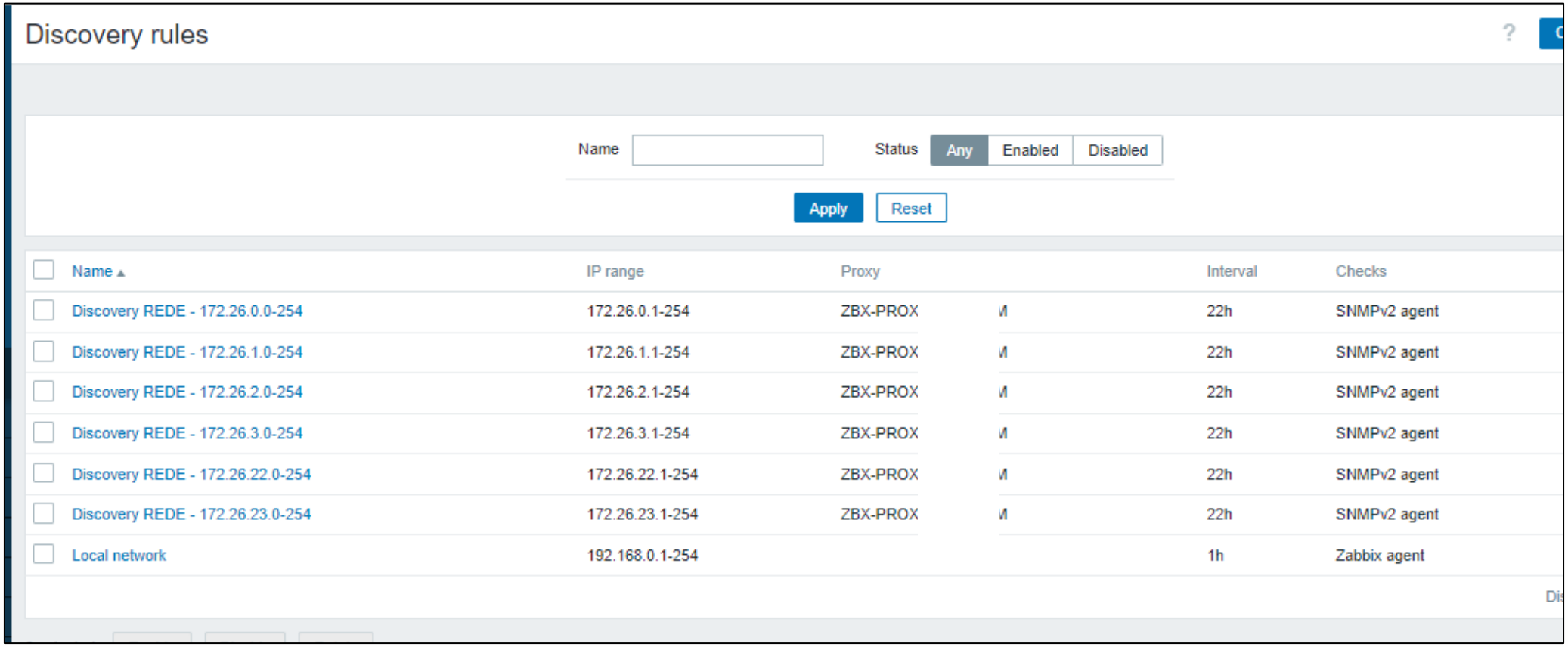
3. Automação com Network Discovery

Principais automações do Zabbix



3. Automação com Network Discovery

Exemplo de automação com Network Discovery



4. Automação com Auto registro

- Um novo servidor ou dispositivo foi criado, é possível registrar-se automaticamente.
- Configurando ações de auto registro para adicionar hosts ao monitoramento.

Chegada de novos dispositivos

Hostname,
Metadata

Auto-registro

Condição

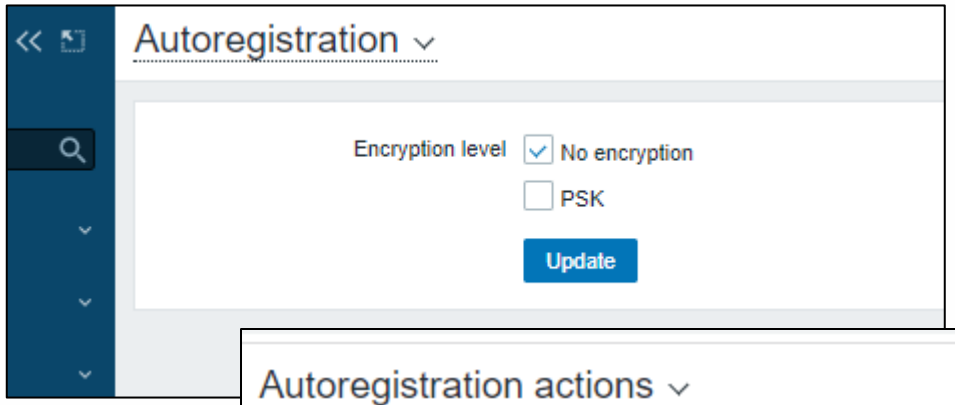
Host name = Hostname
Metadata = Linux

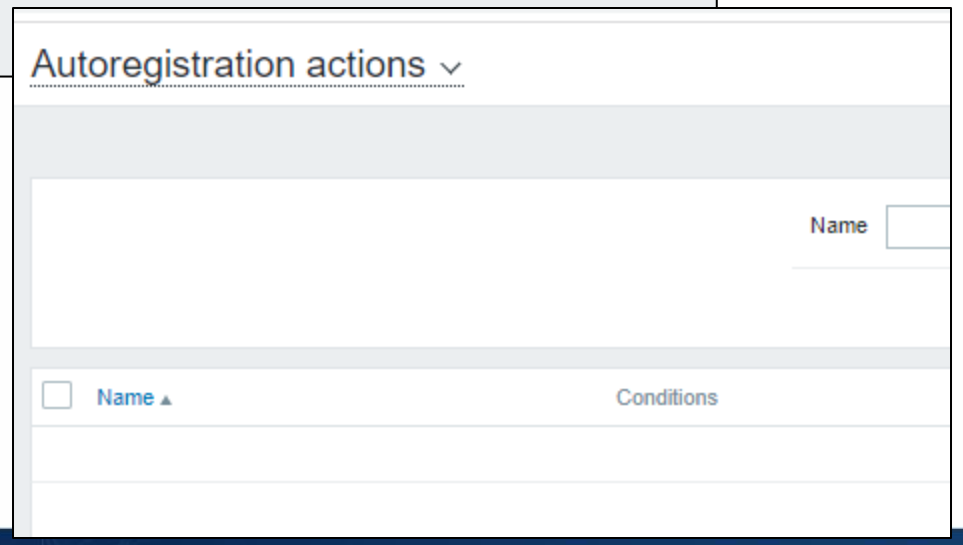
Operação

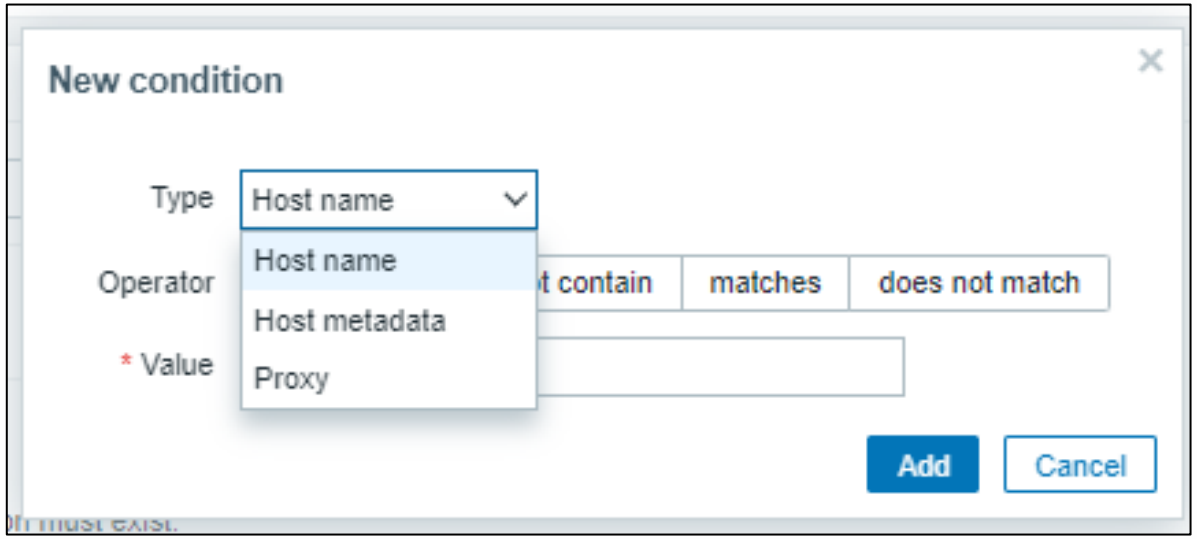
Criar Host
Atribuir Template

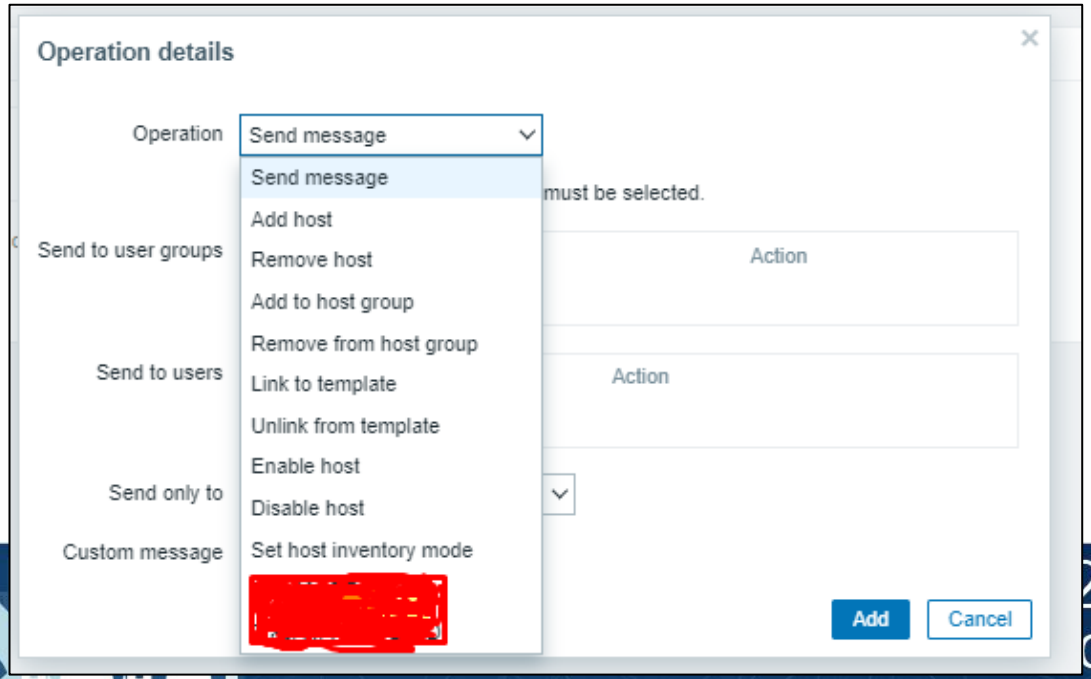
4. Automação com Auto registro

- Exemplo de automação com auto registro

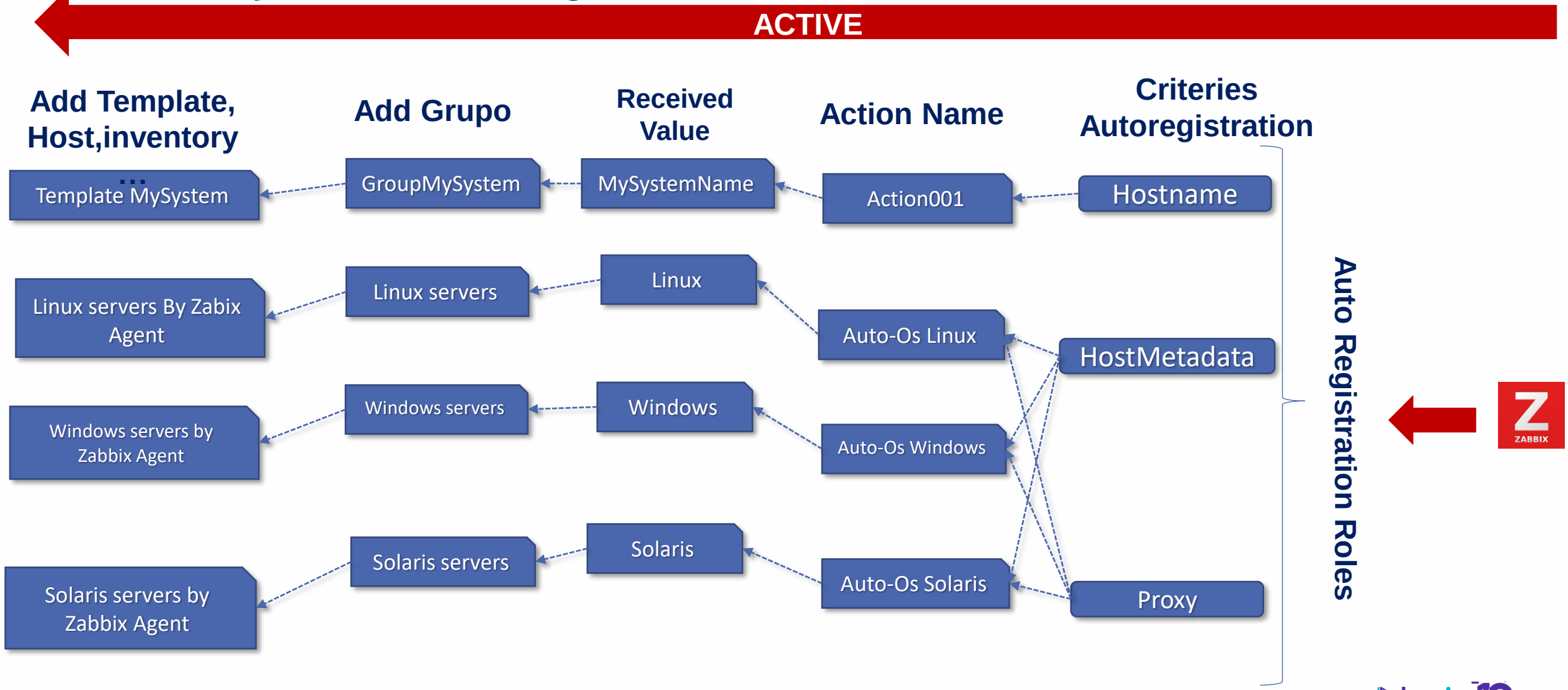








4. Automação com Auto registro

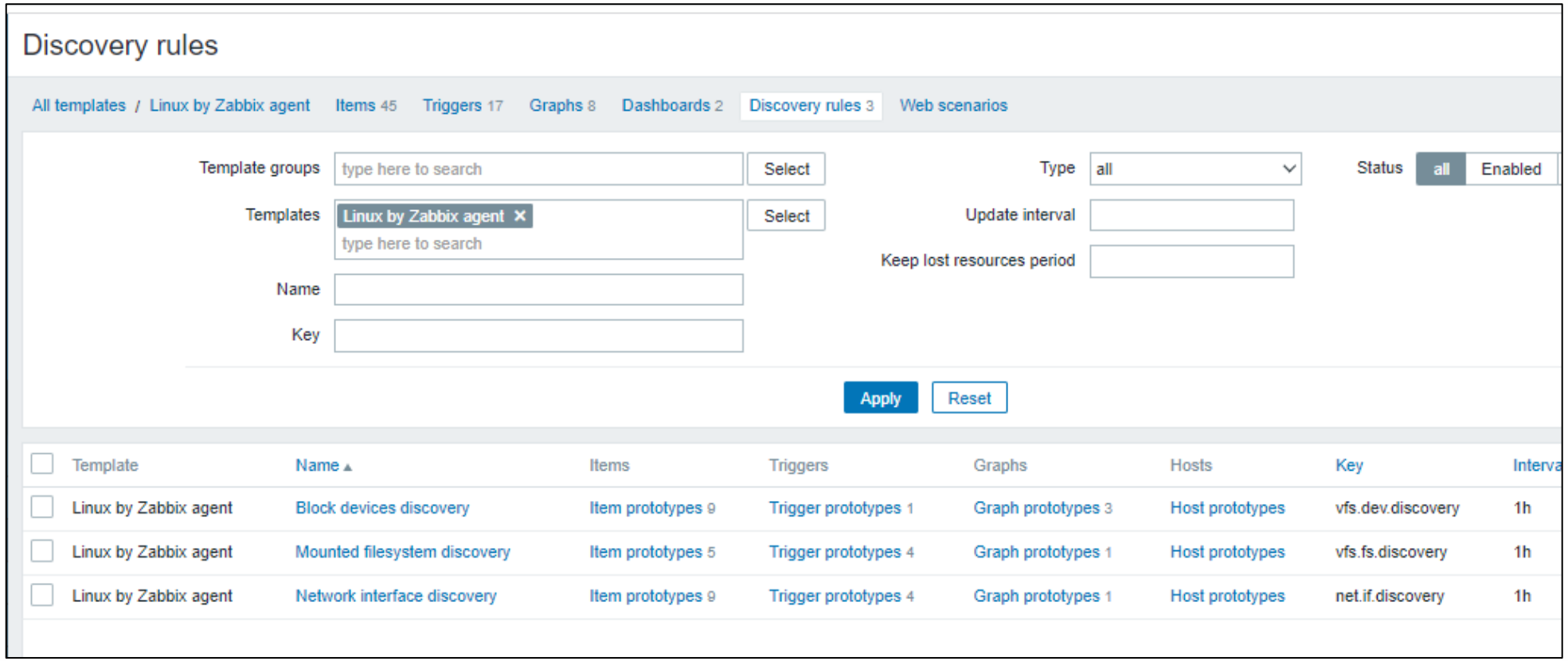


5. Descoberta de baixo nível(Low Level Discovery)

- A descoberta de baixo nível (LLD - Low-Level Discovery) é um recurso que permite que você descubra e crie automaticamente itens, gráficos, triggers e hosts prototypes. É especialmente útil quando você tem um grande número de itens a serem monitorados e não deseja adicionar cada um manualmente no Zabbix.

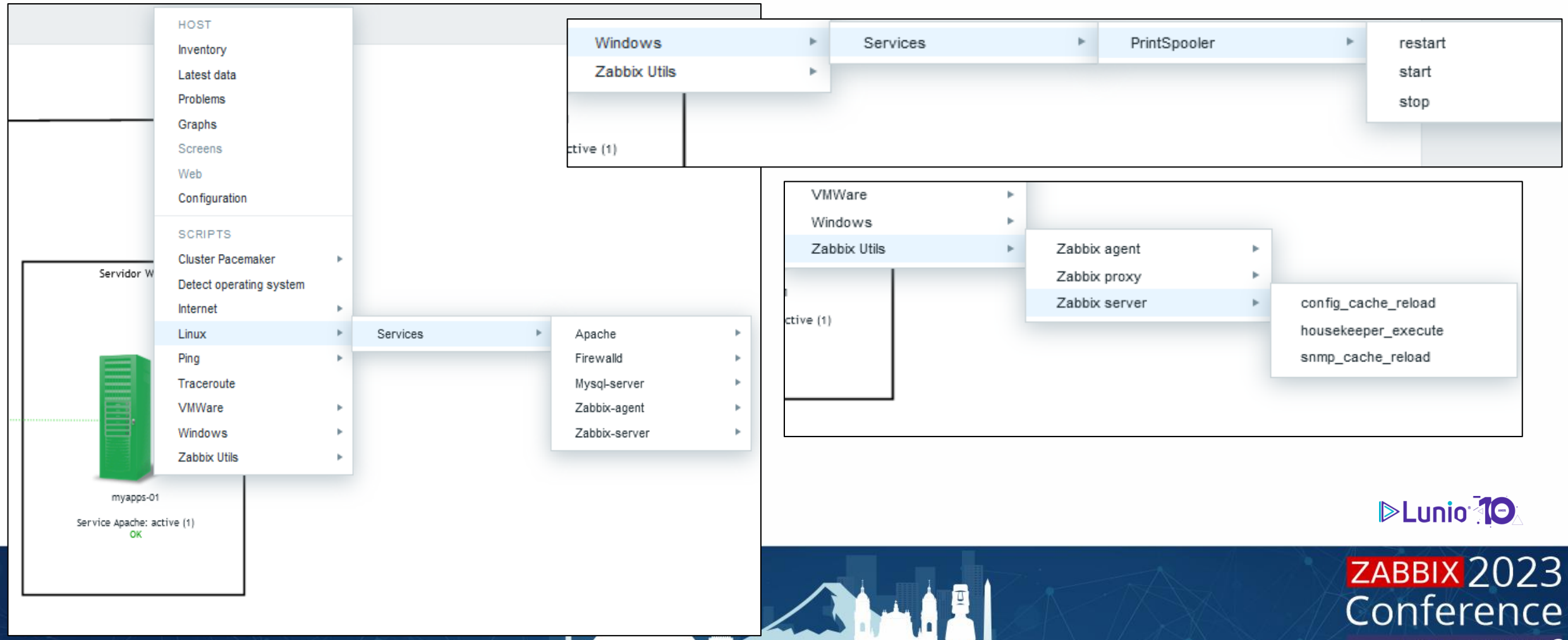
5. Descoberta de baixo nível(Low Level Discovery)

Exemplo de automação de LLD



6. Automações via interface web

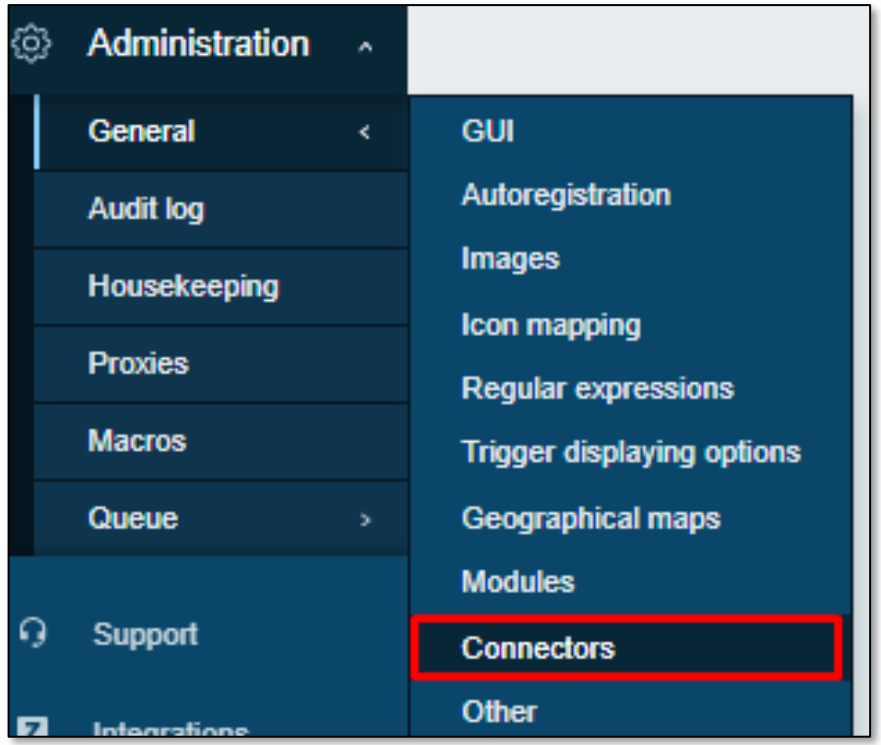
- É possível utilizar do zabbix via interface web

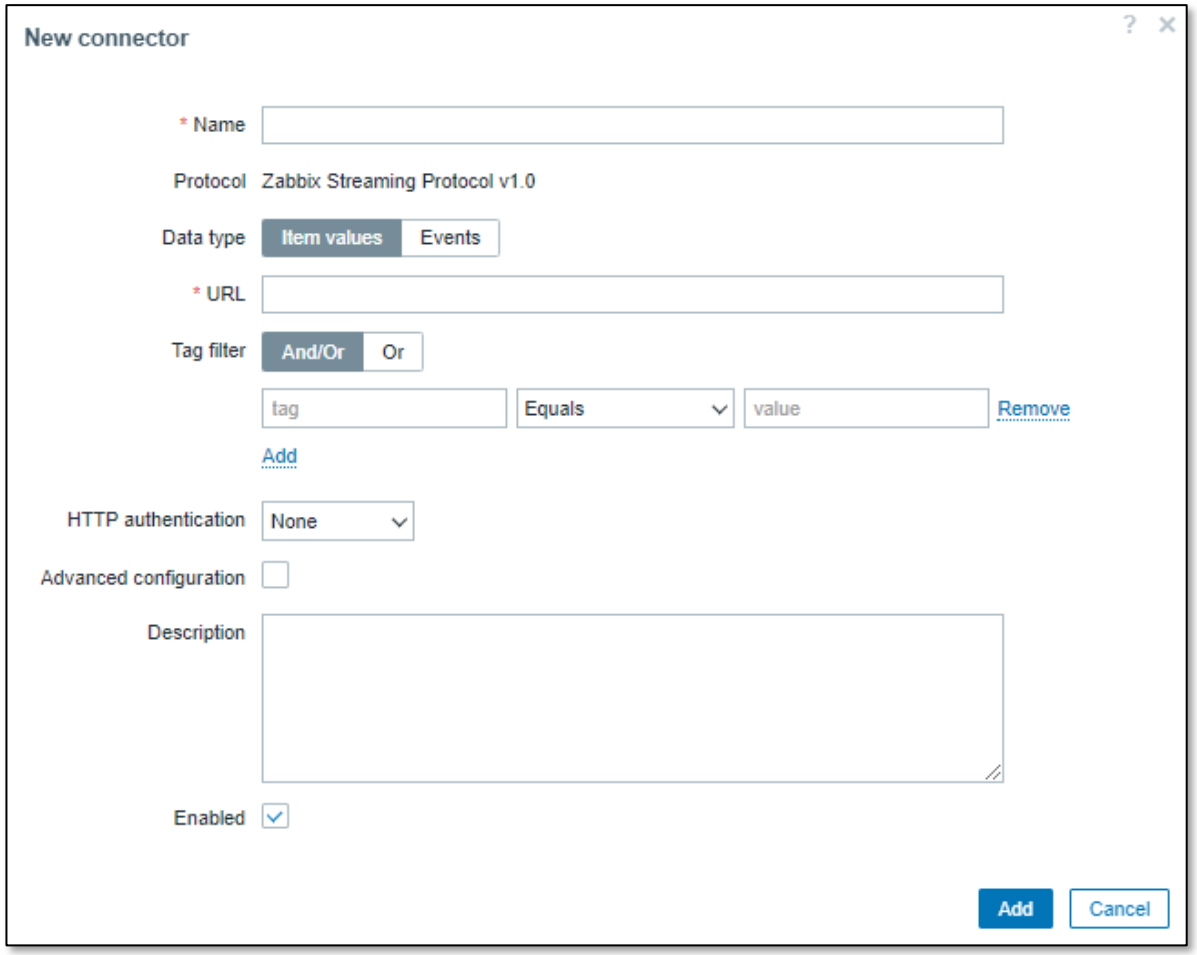


7. Zabbix Streaming Protocol v1.0

- Novidade no zabbix 6.4
- É possível configurar e enviar os valores coletados dos itens para outros sistemas via HTTP.

7. Zabbix Streaming Protocol v1.0





8. Event-Driven Ansible webhook

- Novidade no zabbix 6.4

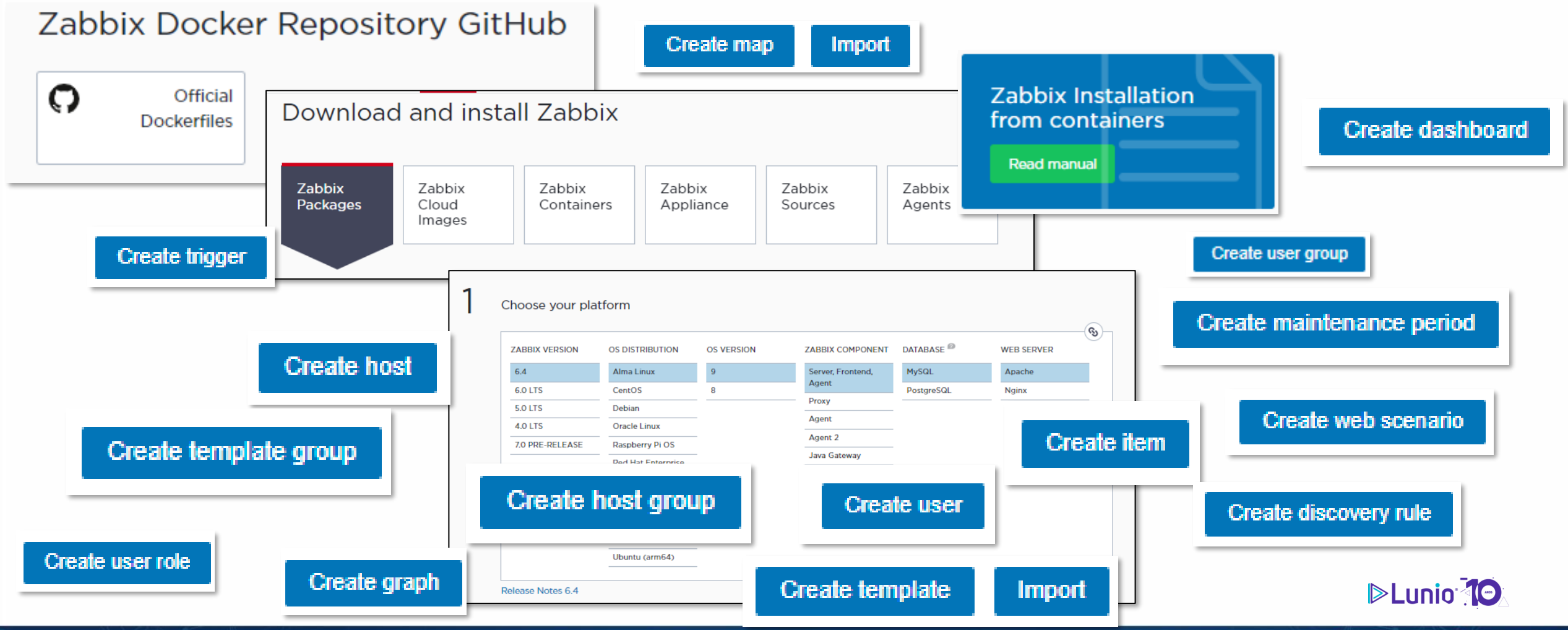
Nesta primeira abordagem, vimos como é possível simplificar a gestão do monitoramento utilizando automações nativas do zabbix.

2.

Quais são as suas tarefas diárias com Zabbix?

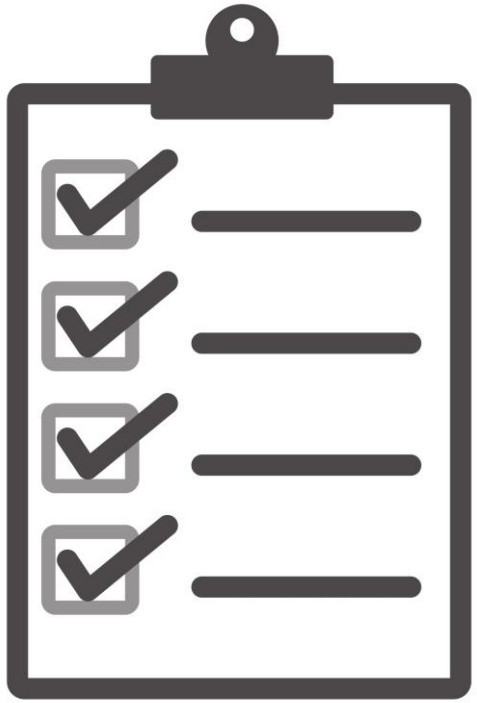
Conhecer a sua rotina de tarefas, permite identificar de forma rápida o que pode ser automatizado

Quais são as suas tarefas diárias com Zabbix?



Quais são as suas tarefas diárias com Zabbix?

1. Tarefas de instalação
2. Tarefas de Configuração
3. Tarefas de Administração



1. Tarefas de instalação

- validação de pré-requisitos/arquitetura
- execução das tarefas de instalação
- validação de instalação e configuração

2. Tarefas de Configuração

- organização de grupos de usuários e usuários
- organização de grupos de hosts
- população de hosts e atribuição de templates
- validação de coleta de dados
- Criação de mapas e dashboards

3. Tarefas de Administração

- geração de relatório de saúde diário
- atualização e migração de ambiente

3.

Como automatizar suas tarefas?

Ganhe velocidade e padronização, sempre: ‘Automatize! Automatize! e Automatize!’

1. Use Zabbix como código

O mesmo conceito de infraestrutura como código (IaC) pode ser aplicado para o Zabbix, ou seja é possível gerenciar e automatizar as tarefas utilizando códigos.

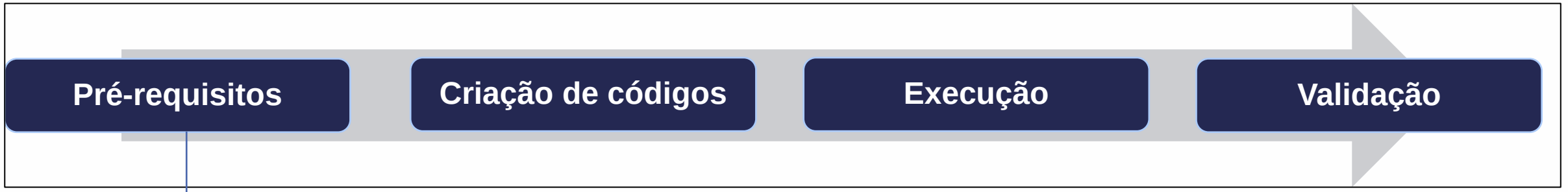
No zabbix temos diversar formas de automatizar, mas tambem em alguns casos precisamos executar tarefas manuais para instalação, configuração e administração

Ex: Chegada de novos clientes, Upgrade de versão

Ex: Pipeline resumida

1. Tarefas de instalação

Como automatizar suas tarefas?



Tarefas

- Validar arquitetura necessária
  1. Cpu
  2. Disco
  3. Memória
  4. Conectividade Internet

Obtendo o código

1. Cpu
   ```
   # lscpu
   ```
2. Disco
   ```
   # df -h
   ```
3. Memória
   ```
   # free -h
   ```
4. Internet/Rede /Conectividade
   ```
   # speedtest-cli --json
   ```

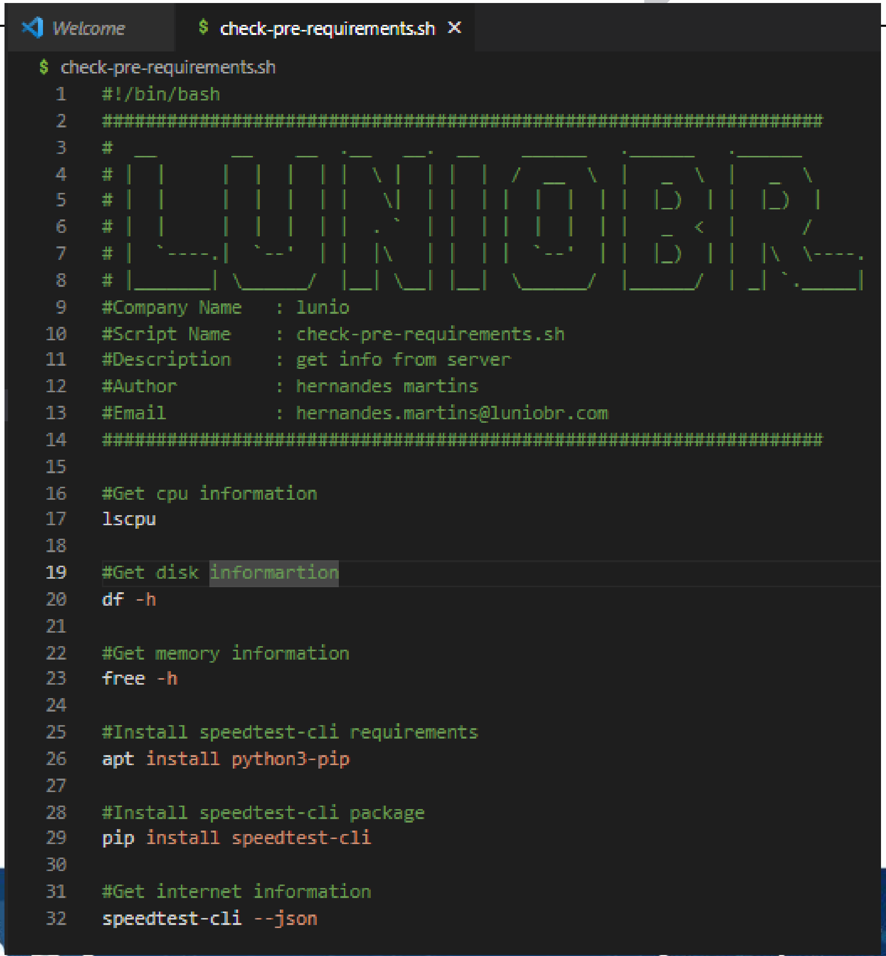
1. Tarefas de instalação

Como automatizar suas tarefas?

Criação de códigos → Execução → Validação

Escrevendo código

1. shell
2. python
3. PHP
4. Perl
5. Go



1. Tarefas de instalação

Como automatizar suas tarefas?

```
root@trn-ubnt-22-04-srv-01:~# cat check-pre-requirements.sh
#!/bin/bash
##########################################################################
#  __      __    __   .__   __.  __    ______   .______    .______
# |  |    |  |  |  | |  \ |  | |  |  /  __  \  |   _  \   |   _  \
# |  |    |  |  |  | |   \|  | |  | |  |  |  | |  |_)  |  |  |_)  |
# |  |    |  |  |  | |  . `  | |  | |  |  |  | |   _  <   |      /
# |  `----.|  `--'  | |  |\   | |  | |  `--'  | |  |_)  |  |  |\  \----.
# |_______| \______/  |__| \__| |__|  \______/  |______/   | _| `._____|
#Company Name    : lunio
#Script Name     : check-pre-requirements.sh
#Description     : get info from server
#Author          : hernandes martins
#Email           : hernandes.martins@luniobr.com
##########################################################################

#Get cpu information
lscpu

#Get disc informartion
df -h

#Get memory information
free -h

#Install speedtest-cli requirements
apt install python3-pip

#Install speedtest-cli package
pip install speedtest-cli

#Get internet information
speedtest-cli --json
```

```
root@trn-ubnt-22-04-srv-01:~# ./check-pre-requirements.sh
Architecture:            x86_64
  CPU op-mode(s):        32-bit, 64-bit
  Address sizes:         39 bits physical, 48 bits virtual
  Byte Order:            Little Endian
CPU(s):                  2
  On-line CPU(s) list:   0,1
Vendor ID:               GenuineIntel
  Model name:            Intel(R) Core(TM) i7-7500U CPU @ 2.70GHz
    CPU family:          6
    Model:               142
    Thread(s) per core:  1
    Core(s) per socket:  2
    Socket(s):           1
```

```
Filesystem                          Size  Used Avail Use% Mounted on
tmpfs                               393M  1.2M  392M   1% /run
/dev/mapper/ubuntu--vg-ubuntu--lv    12G  6.1G  4.6G  58% /
tmpfs                               2.0G     0  2.0G   0% /dev/shm
tmpfs                               5.0M     0  5.0M   0% /run/lock
/dev/sda2                           2.0G  130M  1.7G   8% /boot
tmpfs                               393M  4.0K  393M   1% /run/user/0
               total        used        free      shared  buff/cache   available
Mem:           3.8Gi       774Mi       2.1Gi        10Mi       992Mi       2.8Gi
Swap:          2.1Gi          0B       2.1Gi
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
```

```
Successfully installed speedtest-cli-2.1.3
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager. It is recommended to use a virtual environment instead: https://pip.pypa.io/warnings/venv
```

{"download": 2393853.268266305, "upload": 6057604.407889884, "ping": 91.635, "server": {"url": "http://speedtest.cabotelecom.com.br:8080/speedtest/upload.php", "lat": "-5.7833", "lon": "-35.2000", "name": "Natal", "country": "Brazil", "cc": "BR", "sponsor": "Cabo Telecom", "id": "22461", "host": "speedtest.cabotelecom.com.br:8080", "d": 259.925124317594, "latency": 91.635}, "timestamp": "2023-03-10T17:21:29.877986Z", "bytes_sent": 8339456, "bytes_received": 3638608, "share": null, "client": {"ip": "186.208.7.246", "lat": "-8.1136", "lon": "-35.0145", "isp": "Lideri Telecom", "ispmrating": "3.7", "rating": "0", "ispdlavg": "0", "ispulavg": "0", "loggedin": "0", "country": "BR"}}

1. Tarefas de instalação

Como automatizar suas tarefas?

Criação de códigos

Escrevendo código

1. shell
2. python
3. PHP
4. Perl
5. Go

Welcome | check-pre-requirements.sh

check-pre-requirements.sh

```bash
#!/bin/bash
####################################################################
#  __       __    __  .__   __.  __    ______   .______   .______
# |  |     |  |  |  | |  \ |  | |  |  /  __  \  |   _  \  |   _  \
# |  |     |  |  |  | |   \|  | |  | |  |  |  | |  |_)  | |  |_)  |
# |  |     |  |  |  | |  . `  | |  | |  |  |  | |   _  <  |      /
# |  `----.|  `--'  | |  |\   | |  | |  `--'  | |  |_)  | |  |\  \----.
# |_______| \______/  |__| \__| |__|  \______/  |______/  | _| `._____|
#Company Name    : lunio
#Script Name     : check-pre-requirements.sh
#Description     : get info from server
#Author          : hernandes martins
#Email           : hernandes.martins@luniobr.com
####################################################################

#Get cpu information
lscpu

#Get disk informartion
df -h

#Get memory information
free -h

#Install speedtest-cli requirements
apt install python3-pip

#Install speedtest-cli package
pip install speedtest-cli

#Get internet information
speedtest-cli --json
```

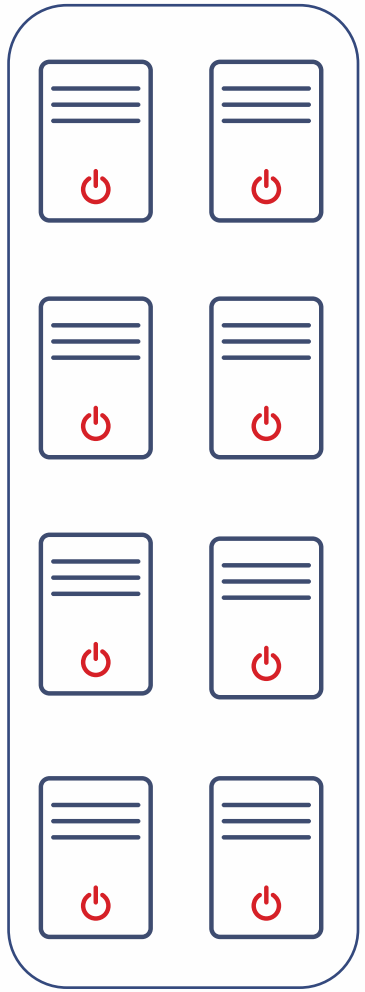
1. Tarefas de instalação

Como automatizar suas tarefas?

Execução → Validação

Utilizando ferramentas de automação

Servers/Containers



Lunio 10

Use Automações com a API do Zabbix

1. Com a api do Zabbix é possível automatizar muitos mais tarefas.

2. É possível fazer realizara as tarefas citadas anteriormente

2. Tarefas de configuração

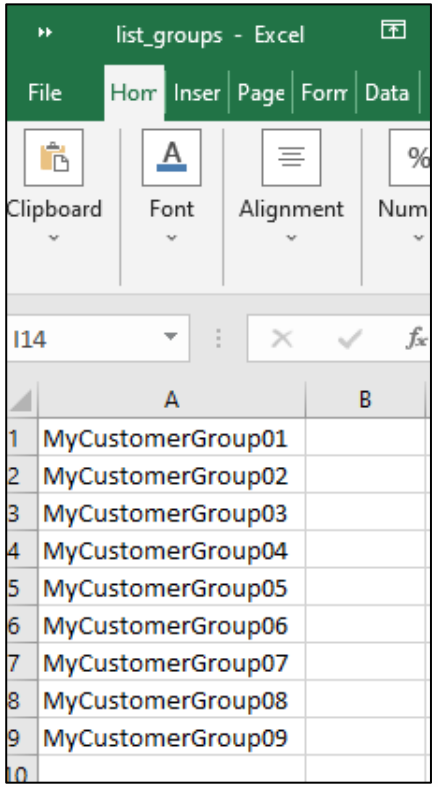
- organização de grupos de usuários e usuários
- organização de grupos de hosts
- população de hosts
- atribuição de templates
- validação de coleta de dados
- criação de mapas e dashboards

2. Tarefas de configuração

Como automatizar suas tarefas?

1. Cadastro de grupos de hosts

- Faça um código para criar grupos
- É possível ler um arquivo csv para criar grupos



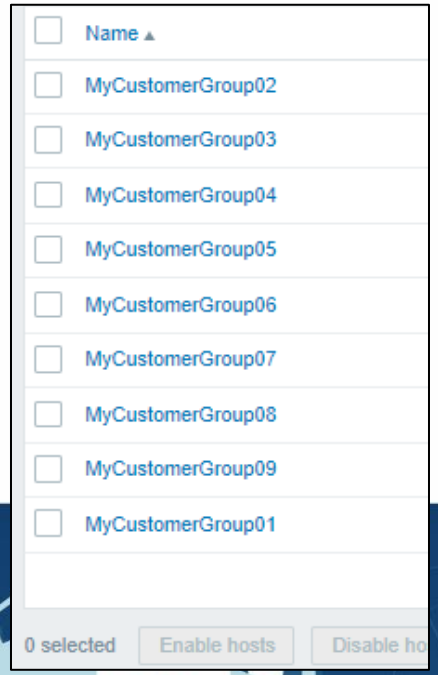
create_group_in_mass.py

```python
#Company Name    : lunio
#Script Name     : create_hosts_in_mass2.py
#Description     : create hosts in mass
#Author          : hernandes martins
#Created Date    : 2022/02/12
#Email           : hernandes.martins@luniobr.com
#--------------------------------------------------------------------------------
# version ='1.0'
# --------------------------------------------------------------------------------
from zabbix_api import ZabbixAPI
import csv

#Enviroment      : Testing
zapi = ZabbixAPI(server="http://192.168.56.119/zabbix")
zapi.login("Admin", "zabbix")

#Enviroment      : Production
#zapi = ZabbixAPI(server="http://192.168.56.119/zabbix")
#zapi.login("Admin", "zabbix")

f = csv.reader(open('/mnt/e/list_groups_csv.csv'), delimiter=';')
for [group] in f:
    print("Registering: ",group)
    groupcreated = zapi.hostgroup.create({
        "name": group,
        "Description": "Please insert your description here"
    })
```

2. Tarefas de configuração


```
hernandes@HPMDSTP-01:~/python_scripts$ /bin/python3 /home/hernandes/python_scripts/create_group_in_mass.py
Registering:  MyCustomerGroup01
Registering:  MyCustomerGroup02
Registering:  MyCustomerGroup03
Registering:  MyCustomerGroup04
Registering:  MyCustomerGroup05
Registering:  MyCustomerGroup06
Registering:  MyCustomerGroup07
Registering:  MyCustomerGroup08
Registering:  MyCustomerGroup09
hernandes@HPMDSTP-01:~/python_scripts$
```

Novos grupos criados via api



2. Tarefas de configuração

Como automatizar suas tarefas?

Criar Usuários

- Faça um Código para cadastrar usuários

	A	B	C	D	E	F
1	student01	123@zabbix	student01	surname-student-01	surname-student-01@labmeetup.com	1
2	student02	123@zabbix	student02	surname-student-02	surname-student-01@labmeetup.com	1
3	student03	123@zabbix	student03	surname-student-03	surname-student-01@labmeetup.com	1
4	student04	123@zabbix	student04	surname-student-04	surname-student-01@labmeetup.com	1
5	student05	123@zabbix	student05	surname-student-05	surname-student-01@labmeetup.com	1
6	student06	123@zabbix	student06	surname-student-06	surname-student-01@labmeetup.com	1
7	student07	123@zabbix	student07	surname-student-07	surname-student-01@labmeetup.com	1
8	student08	123@zabbix	student08	surname-student-08	surname-student-01@labmeetup.com	1
9	student09	123@zabbix	student09	surname-student-09	surname-student-01@labmeetup.com	1
10	student10	123@zabbix	student10	surname-student-10	surname-student-01@labmeetup.com	1

create_user_in_mass2.py > ...

```python
#
#Company Name    : lunio
#Script Name     : create_hosts_in_mass2.py
#Description     : create hosts in mass
#Author          : hernandes martins
#Created Date    : 2022/02/12
#Email           : hernandes.martins@luniobr.com
#----------------------------------------------------------------------------
# version ='1.0'
# ---------------------------------------------------------------------------
from zabbix_api import ZabbixAPI
import csv

#Enviroment      : Testing
zapi = ZabbixAPI(server="http://192.168.56.112/zabbix")
zapi.login(api_token='d07db288c947dd218f1bbad6adb6c5e13d173da4d3796fb6f65a9263ea7acc8d')

#Enviroment      : Production
#zapi = ZabbixAPI(server="http://192.168.56.119/zabbix")
#zapi.login("Admin", "zabbix")

f = csv.reader(open('/mnt/d/listadeusers.csv'), delimiter=',')
for [alias,zpasswd,name,surname,typeuser,usergroupid] in f:
    print("Creating user:",alias,"from line "), f.line_num
    usercreated = zapi.user.create({
        "username": alias,
        "passwd": zpasswd,
        "roleid": '3',
        "name": name,
```

2. Tarefas de configuração

Como automatizar suas tarefas?

```
hernandes@HPMDSTP-01:~/python_scripts$ /bin/python3 /home/hernandes/python_scripts/create_user_in_mass2.py
Creating user: student01 from line
Creating user: student02 from line
Creating user: student03 from line
Creating user: student04 from line
Creating user: student05 from line
Creating user: student06 from line
Creating user: student07 from line
Creating user: student08 from line
Creating user: student09 from line
Creating user: student10 from line
Creating user: student11 from line
Creating user: student12 from line
Creating user: student13 from line
Creating user: student14 from line
Creating user: student15 from line
Creating user: student16 from line
Creating user: student17 from line
Creating user: student18 from line
Creating user: student19 from line
Creating user: student20 from line
Creating user: student21 from line
Creating user: student22 from line
Creating user: student23 from line
Creating user: student24 from line
Creating user: student25 from line
Creating user: student26 from line
```

Username | User roles: type here to search
Name: student | User groups: type here to search
Last name

Apply | Reset

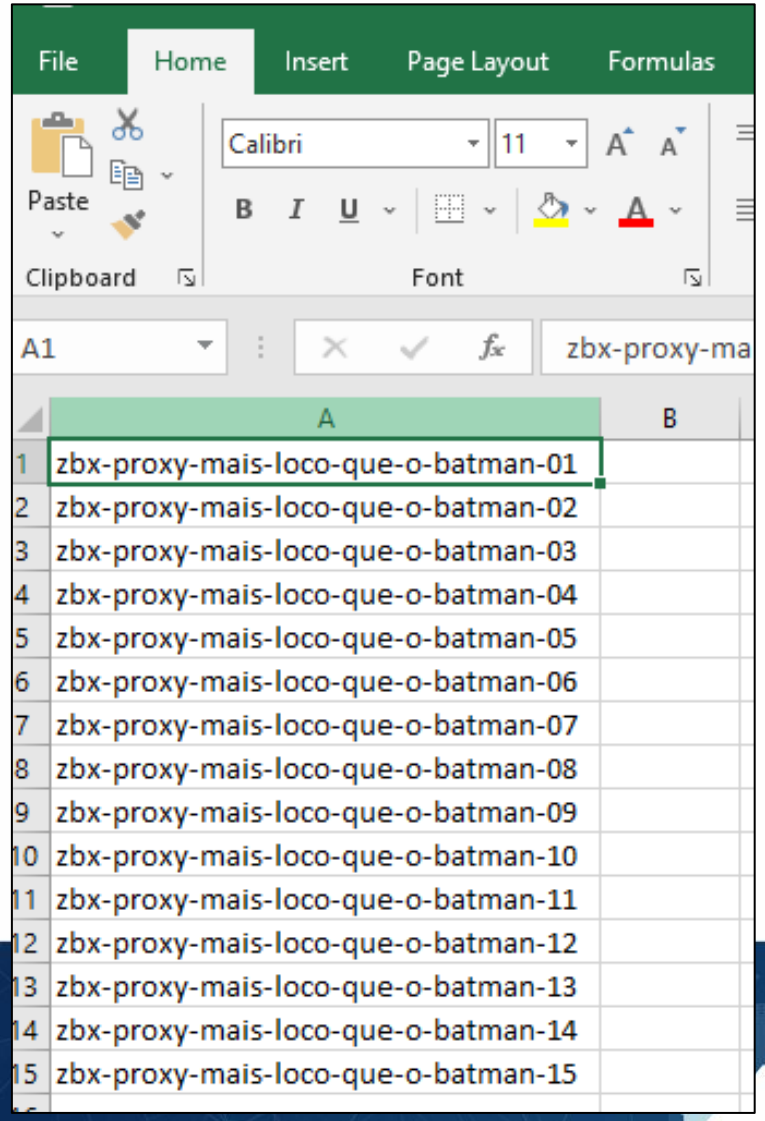
Username ▲	Name	Last name	User role	Groups	Is online?	Login	Fronte
student02	student02	surname-student-02	Super admin role	Internal	No	Ok	Intern
student03	student03	surname-student-03	Super admin role	Internal	No	Ok	Intern
student04	student04	surname-student-04	Super admin role	Internal	No	Ok	Intern
student05	student05	surname-student-05	Super admin role	Internal	No	Ok	Intern
student06	student06	surname-student-06	Super admin role	Internal	No	Ok	Intern
student07	student07	surname-student-07	Super admin role	Internal	No	Ok	Intern
student08	student08	surname-student-08	Super admin role	Internal	No	Ok	Intern

Novos usários criados via api

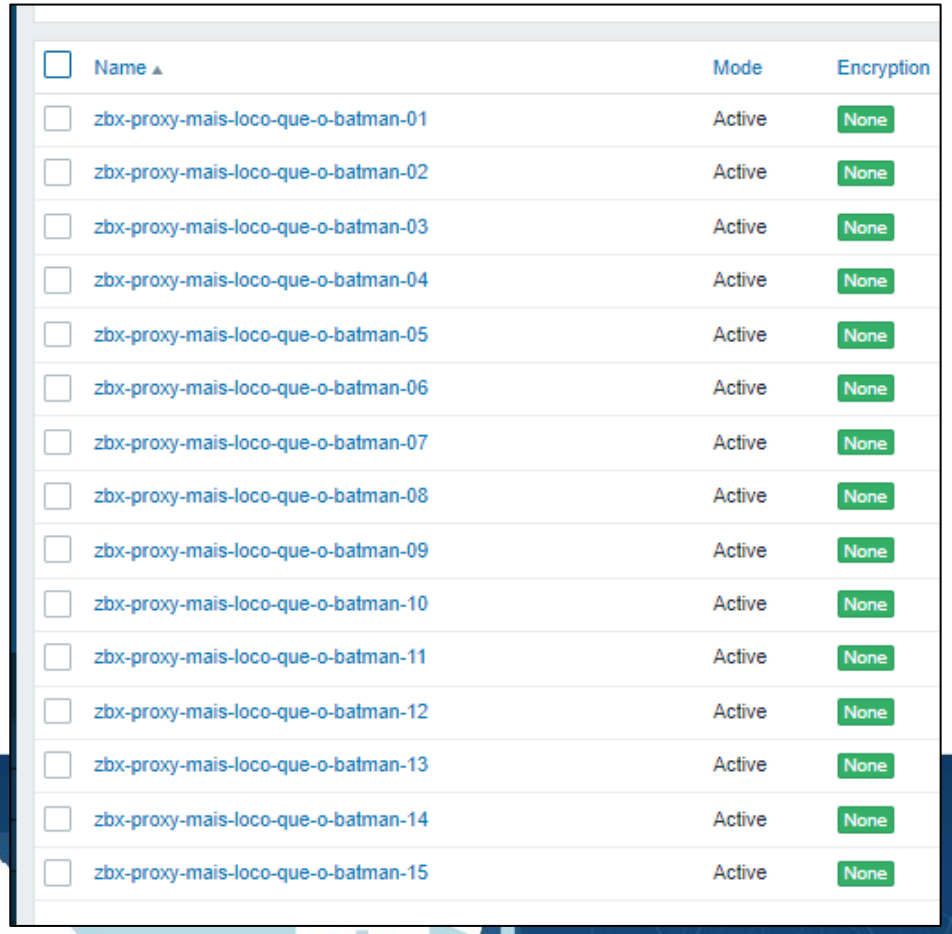
2. Tarefas de configuração

Como automatizar suas tarefas?

População do Zabbix
Cadastro de Proxies
Faça um código para criar zabbix proxies




```python
f = csv.reader(open('/mnt/d/listadeproxies.csv'), delimiter=',')
for [zbxprxname] in f:
    print("Proxy",zbxprxname," created Ra tá tá tá!")
    proxycriado = zapi.proxy.create({
        "host": zbxprxname,
        "status": 5,
    })
```



2. Tarefas de configuração

Como automatizar suas tarefas?

População do Zabbix

Cadastro de hosts e atribuição de templates/proxy aos hosts

- Faça um Código para cadastrar hosts
- Adicione modulos para cadastrar hosts com multiplos parâmetros.

	A	B	C
1	hostname	127.0.0.1	10564
2	hostname	127.0.0.1	10001
3	hostname	127.0.0.1	10564
4	hostname	127.0.0.1	10001
5	hostname	127.0.0.1	10564
6	hostname	127.0.0.1	10001
7	hostname	127.0.0.1	10564
8	hostname	127.0.0.1	10001
9	hostname	127.0.0.1	10564

```
f = csv.reader(open('/mnt/d/listadehosts-meetup.csv'), delimiter=',')
for [hostname,ip,zbxtmplate] in f:
        #create host
        hostcriado = zapi.host.create({
        "host": hostname,
        "status": 0,
        "interfaces": [
                {
                "type": 1,
                "main": 1,
                "useip": 1,
                "ip": ip,
                "dns": "",
                "port": "10564"
                }],
        "groups": [
                {
                "groupid": 2
                }
        ],
        "templates": [
                {
                "templateid": zbxtmplate
                }
        ]
    })
        print("Host", hostname, "created successfully pow pow!")
```

EMS OUTPUT DEBUG CONSOLE TERMINAL GITLENS

2. Tarefas de configuração

Como automatizar suas tarefas?

Hosts

Name ▲	Items	Triggers	Graphs	Discovery	Web	Interface	Proxy	Templates	Status	Availability
hostname001	Items 3	Triggers 3	Graphs	Discovery	Web	127.0.0.1:10050		ICMP Ping	Enabled	ZBX
hostname01	Items 3	Triggers 3	Graphs	Discovery	Web	127.0.0.1:10050		ICMP Ping	Enabled	ZBX
hostname02	Items 3	Triggers 3	Graphs	Discovery	Web	127.0.0.1:10050		ICMP Ping	Enabled	ZBX
hostname03	Items 3	Triggers 3	Graphs	Discovery	Web	127.0.0.1:10050		ICMP Ping	Enabled	ZBX
hostname04	Items 3	Triggers 3	Graphs	Discovery	Web	127.0.0.1:10050		ICMP Ping	Enabled	ZBX
hostname05	Items 3	Triggers 3	Graphs	Discovery	Web	127.0.0.1:10050		ICMP Ping	Enabled	ZBX
hostname06	Items 3	Triggers 3	Graphs	Discovery	Web	127.0.0.1:10050		ICMP Ping	Enabled	ZBX
hostname07	Items 3	Triggers 3	Graphs	Discovery	Web	127.0.0.1:10050		ICMP Ping	Enabled	ZBX
hostname08	Items 3	Triggers 3	Graphs	Discovery	Web	127.0.0.1:10050		ICMP Ping	Enabled	ZBX
hostname09	Items 3	Triggers 3	Graphs	Discovery	Web	127.0.0.1:10050		ICMP Ping	Enabled	ZBX
hostname10	Items 3	Triggers 3	Graphs	Discovery	Web	127.0.0.1:10050		ICMP Ping	Enabled	ZBX
hostname11	Items 3	Triggers 3	Graphs	Discovery	Web	127.0.0.1:10050		ICMP Ping	Enabled	ZBX
hostname12	Items 3	Triggers 3	Graphs	Discovery	Web	127.0.0.1:10050		ICMP Ping	Enabled	ZBX
hostname13	Items 3	Triggers 3	Graphs	Discovery	Web	127.0.0.1:10050		ICMP Ping	Enabled	ZBX

2. Tarefas de configuração

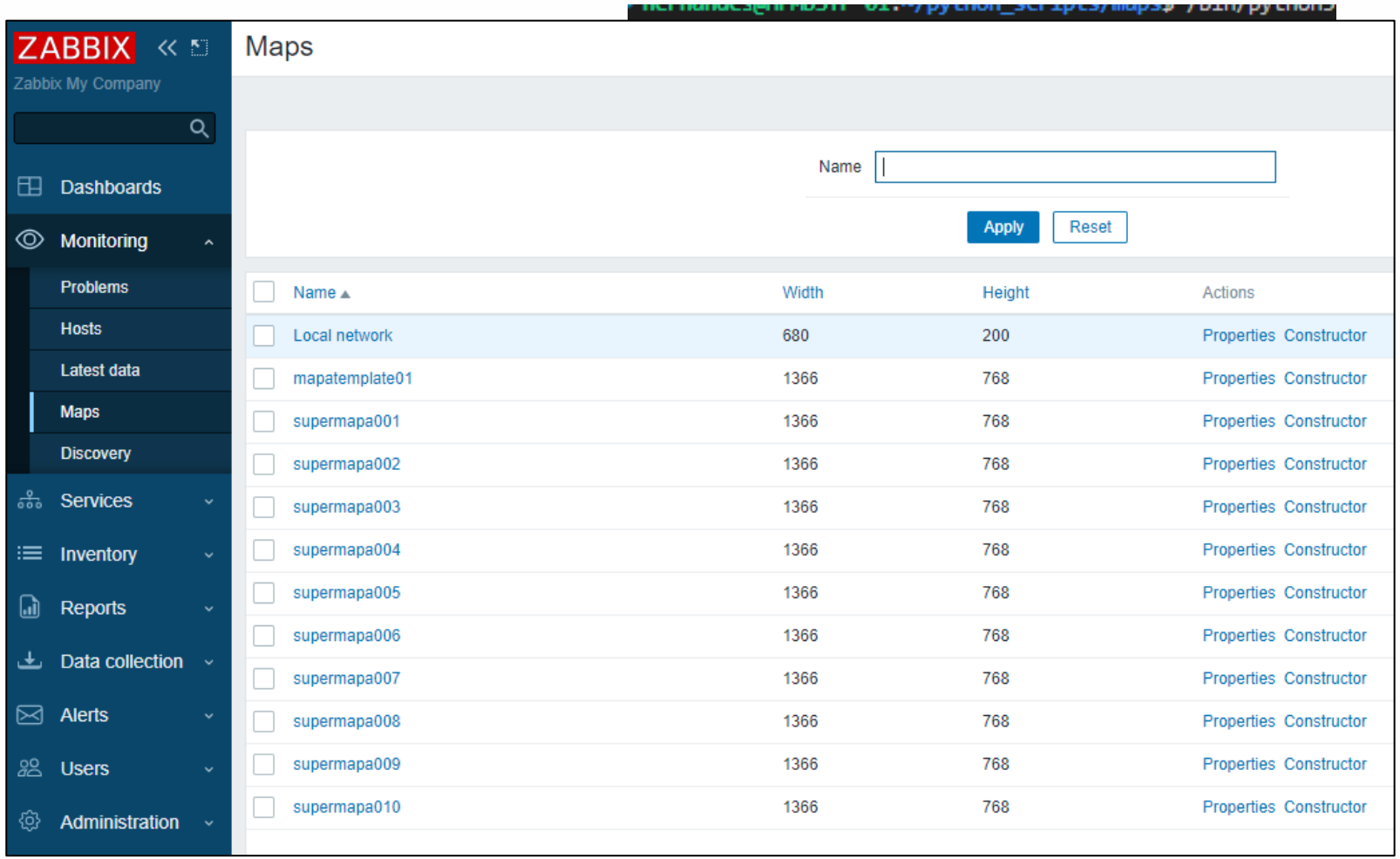
Como automatizar suas tarefas?

Automação de Mapas

Cadastro de mapas

Faça um código para criar mapas

	A	B	C	D
1	supermapa001	2	Topologia Titulo01	
2	supermapa002	2	Topologia Titulo02	
3	supermapa003	2	Topologia Titulo03	
4	supermapa004	2	Topologia Titulo04	
5	supermapa005	2	Topologia Titulo05	
6	supermapa006	2	Topologia Titulo06	
7	supermapa007	2	Topologia Titulo07	
8	supermapa008	2	Topologia Titulo08	
9	supermapa009	2	Topologia Titulo09	
10	supermapa010	2	Topologia Titulo10	
11				



3. Tarefas de Administração

- Geração de relatório de saúde diário

  É possível automatizar o envio de relatórios de saúde do ambiente zabbix

- Atualização e migração de ambiente

  Faça um código para migrar seu ambiente de produção para ambiente de homologação diariamente utilizando a api do zabbix.

3. Tarefas de Administração

Migração de ambiente via Zabbix API

Zabbix Server old
6.4

Zabbix Server New
7.0

Zabbix Server Production

Zabbix Server Testing

```python
#!/usr/bin/env python
# -*- coding: utf-8 -*-
# This script migrates hosts and their information to another remote zabbix.
# Created by Hernandes Martins 11/2022
# Reviewd by Rafael Magalhaes 12/2022

# Python Modules
from zabbix_api import ZabbixAPI
import ssl
import configparser

# Read config file config.ini
config = configparser.ConfigParser()
config.read('config.ini')

# Read parameters in file config.ini
zbxsourceaurl = config['DefaultConfigSource']['zbxsourceurl']
zbxsourceauser = config['DefaultConfigSource']['zbxsourceuser']
zbxsourceapass = config['DefaultConfigSource']['zbxsourcepass']
zbxsourceatoken = config['DefaultConfigSource']['zbxsourcetoken']
zbxdestburl = config['DefaultConfigDest']['zbxdesturl']
zbxdestbuser = config['DefaultConfigDest']['zbxdestuser']
zbxdestbpass = config['DefaultConfigDest']['zbxdestpass']
zbxdestatoken = config['DefaultConfigDest']['zbxdesttoken']

zbxgrouptomigrate = config['DefaultConfigDest']['zbxdestgroup']
#print("Debug group to migrate", zbxgrouptomigrate )
timeout = config['DefaultConfig']['timeout']

# Functions Parameters
def connect_zbx_source():
    zapi = ZabbixAPI(server=zbxsourceaurl,timeout=int(timeout))
    zapi.validate_certs = False
    ssl._create_default_https_context = ssl._create_unverified_context
    zapi.login(zbxsourceauser,zbxsourceapass)
    print("Connected to Zabbix API Version %s" % zapi.api_version())
    return zapi

def connect_zbx_dest():
    zapi = ZabbixAPI(server=zbxdestburl,timeout=int(timeout))
    zapi.validate_certs = False
```

3. Tarefas de Administração

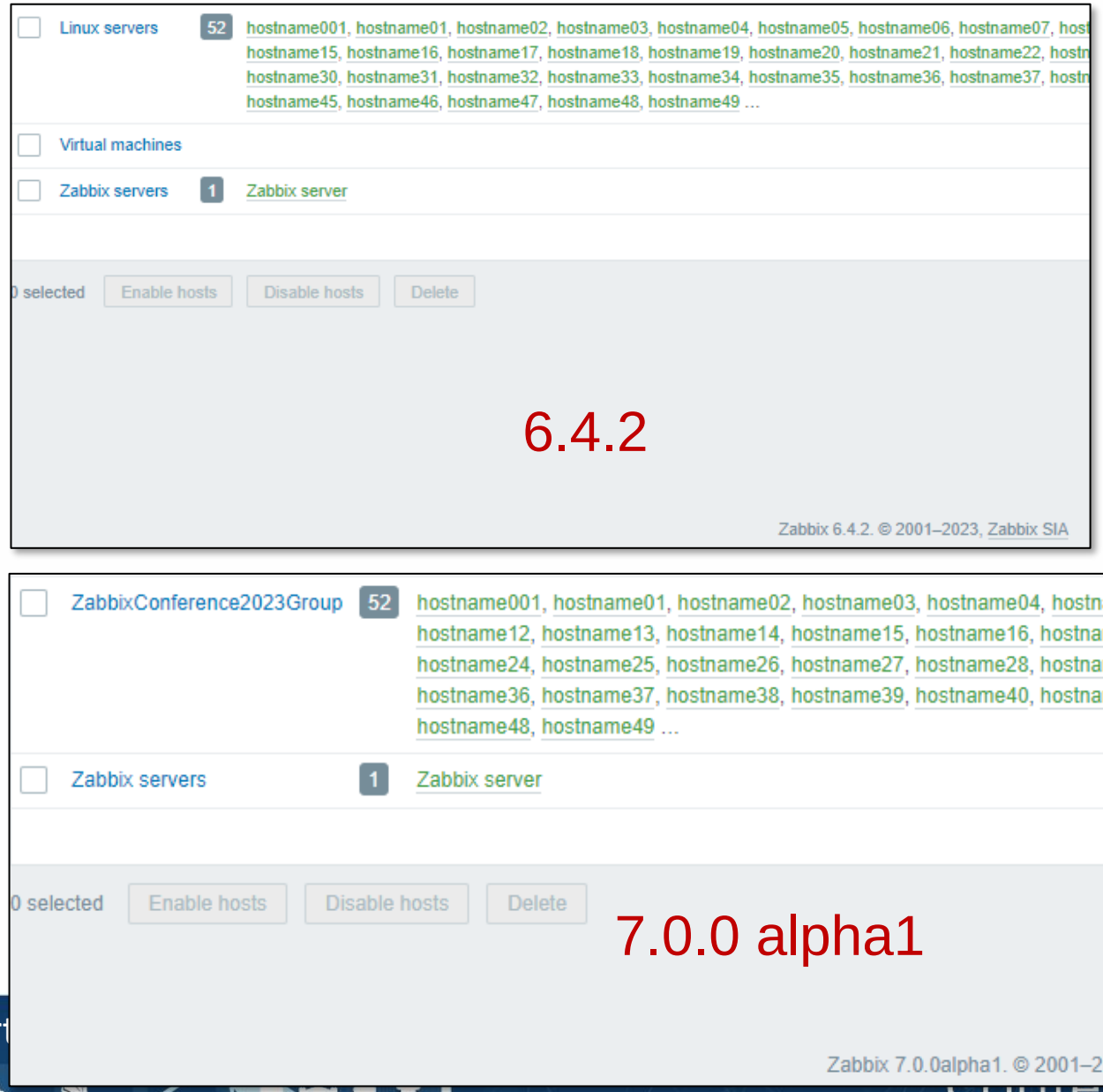
Migração de ambiente via Zabbix API

```
hernandes@HPMDSTP-01:~/python_scripts/zbx_migrate_import_hosts$ /bin/python3 /home/hernandes/python_sc
################################################################################
###################### START INFORMATION ######################################
################################################################################

Total hosts Zabbix server source:  53
Total hosts Zabbix server dest:  1

################################################################################
###################### GETTING AND IMPORTING HOSTS ############################
################################################################################
--- Host exist in zabbix server B:  Zabbix server
>>>>> Importing hostname001  -  127.0.0.1 : 10050  to zabbix server B.
>>>>> Importing hostname01  -  127.0.0.1 : 10050  to zabbix server B.
>>>>> Importing hostname02  -  127.0.0.1 : 10050  to zabbix server B.
>>>>> Importing hostname03  -  127.0.0.1 : 10050  to zabbix server B.
>>>>> Importing hostname04  -  127.0.0.1 : 10050  to zabbix server B.
>>>>> Importing hostname05  -  127.0.0.1 : 10050  to zabbix server B.
>>>>> Importing hostname06  -  127.0.0.1 : 10050  to zabbix server B.
>>>>> Importing hostname07  -  127.0.0.1 : 10050  to zabbix server B.
>>>>> Importing hostname08  -  127.0.0.1 : 10050  to zabbix server B.
```

```
>>>>> Importing hostname42  -  127.0.0.1 : 10050  to zabbix server B.
>>>>> Importing hostname43  -  127.0.0.1 : 10050  to zabbix server B.
>>>>> Importing hostname44  -  127.0.0.1 : 10050  to zabbix server B.
>>>>> Importing hostname45  -  127.0.0.1 : 10050  to zabbix server B.
>>>>> Importing hostname46  -  127.0.0.1 : 10050  to zabbix server B.
>>>>> Importing hostname47  -  127.0.0.1 : 10050  to zabbix server B.
>>>>> Importing hostname48  -  127.0.0.1 : 10050  to zabbix server B.
>>>>> Importing hostname49  -  127.0.0.1 : 10050  to zabbix server B.
>>>>> Importing hostname50  -  127.0.0.1 : 10050  to zabbix server B.
>>>>> Importing hostname51  -  127.0.0.1 : 10050  to zabbix server B.

################################################################################
################## LUNIO REPORT HOSTS MIGRATION ################################
################################################################################

Getting total host list Zabbix server source:  53
Getting total host list Zabbix server dest:  54
```

3. Tarefas de Administração

Migração efetuada com sucesso.



https://github.com/mrhernandes/zbx_migrate_import

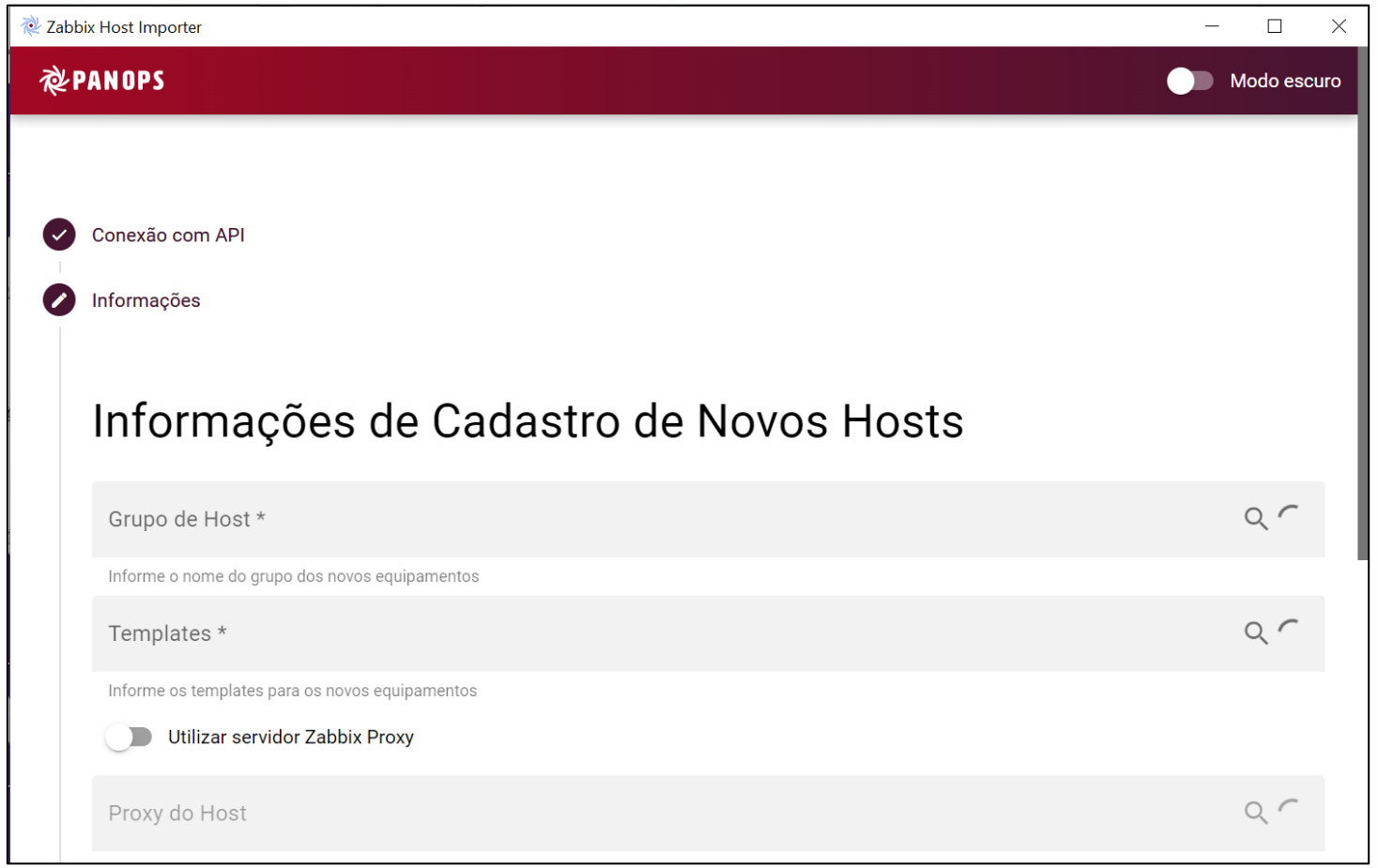
4.

Cases e exemplos de uso

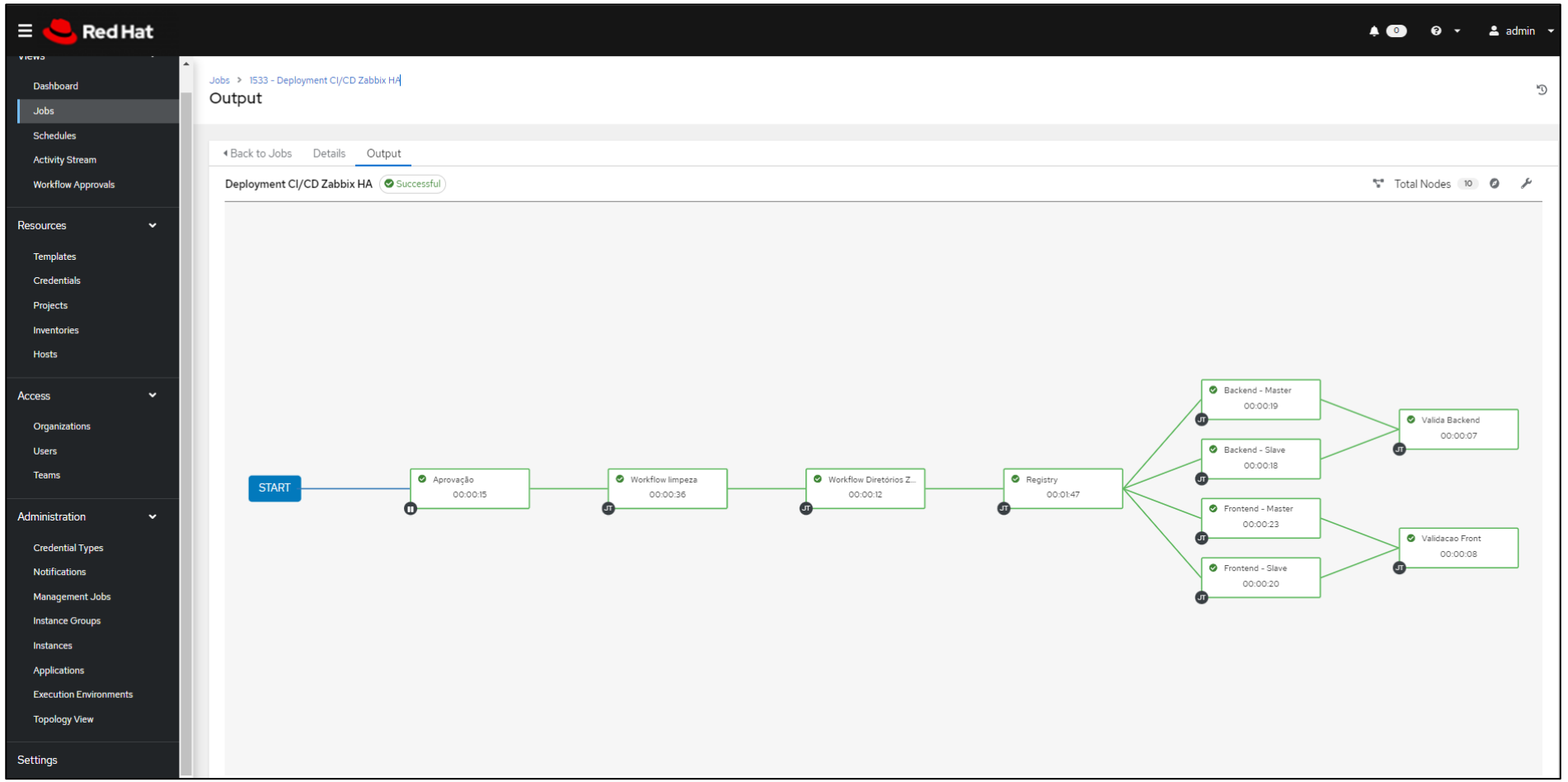
Compartilhar experiências e conhecimento
é importante para o crescimento de todos

1. Aplicação para população de hosts

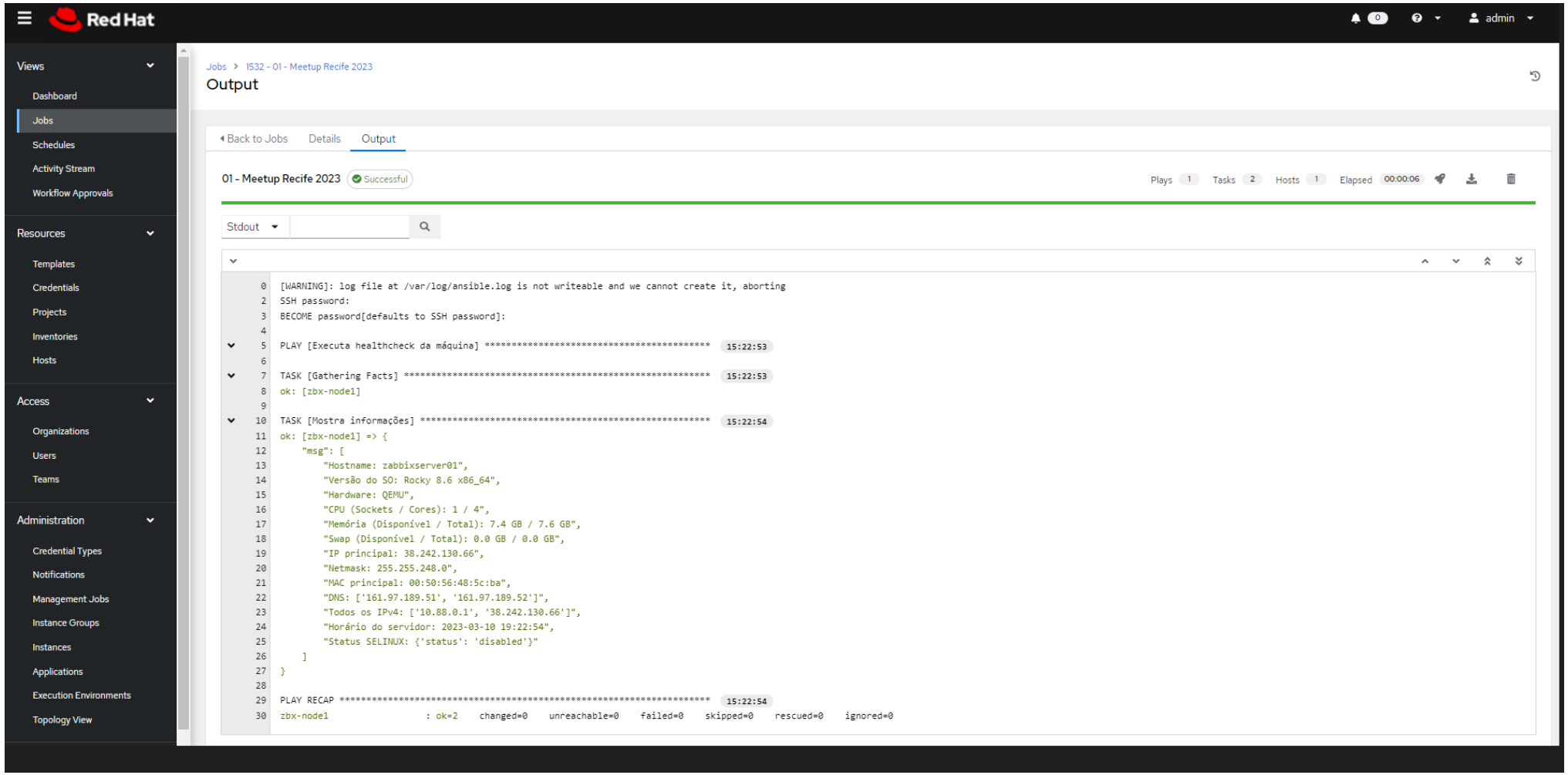
HOSTS MAPAS LUNIO AWX



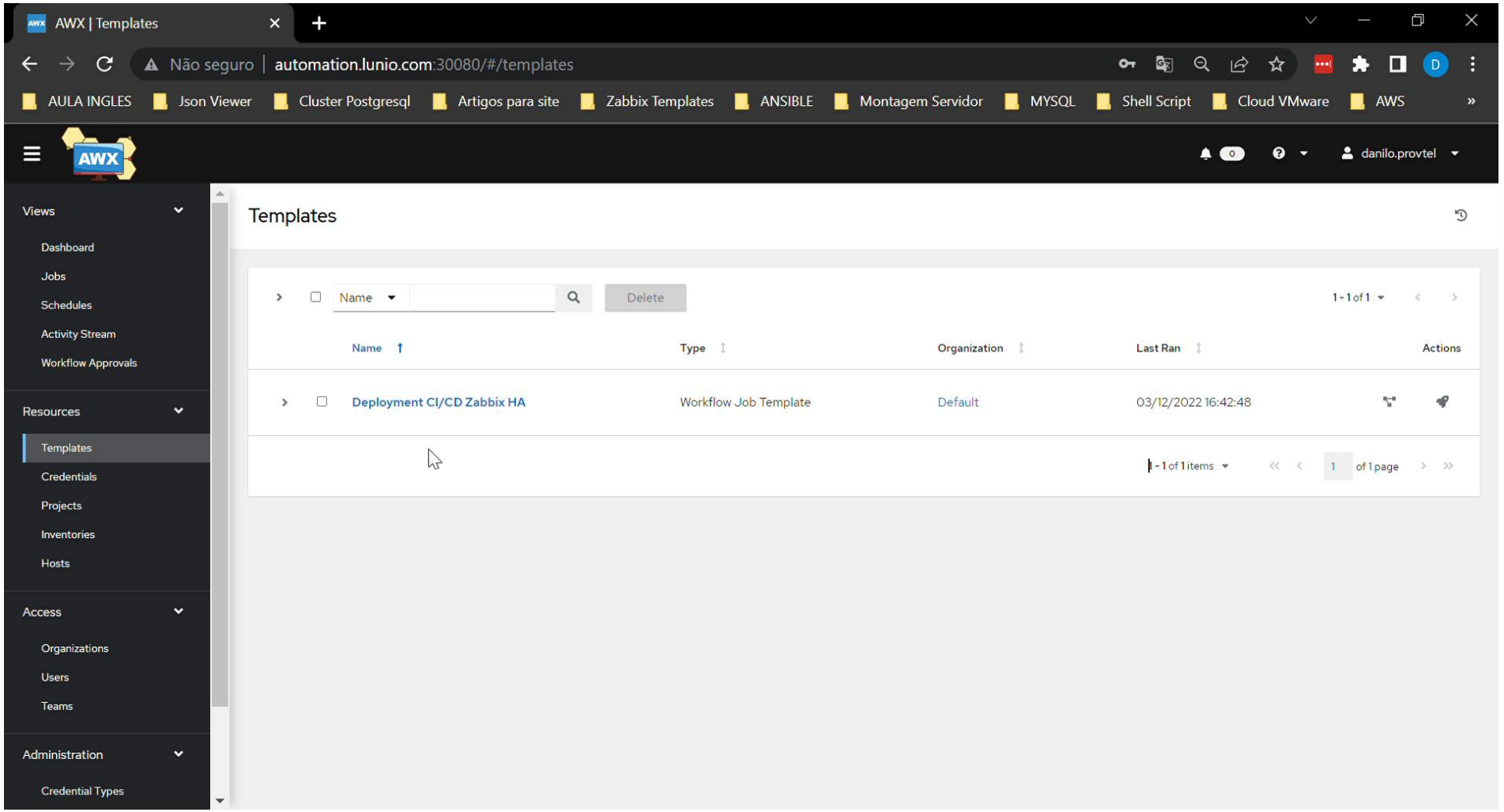
2. Criação de pipeline automatizada para criação de ambientes



2. Criação de pipeline automatizada para criação de ambientes



2. Criação de pipeline automatizada para criação de ambientes



ZABBIX 2023 Conference LATIN AMERICA

PERGUNTAS

Hernandes Martins

EMAIL: hernandes.martins@luniobr.com