Pessoal tudo bem com todos? Espero que sim.

Peço desculpas por não ter feito mais nenhum post mas final de ano a correria fica imensa como sabem, principalmente para nós que trabalhamos com TI.

Bem, já que tivemos esse tempo parados vamos retomar com um assunto muito legal. O monitoramento de APIs direto no Zabbix sem precisar usar nenhum Script para processar os dados.

Como todos nós sabemos, monitorar o ambiente produtivo é uma questão de sucesso; imaginem o seu usuário acessando um aplicativo qualquer e a API ao qual o mesmo se conecta está fora ou com algum problema de acesso, provavelmente o seu “cliente” não vai ficar feliz e vai te ligar bem chateado.

Então que tal aproveitar que a nova versão do Zabbix 4.0 tem suporte a monitores/sensores HTTP?

Ok, sem mais papo vamos colocar a mão na massa.

visão global

Este tipo de item permite a pesquisa de dados usando o protocolo HTTP / HTTPS. O trapping também é possível usando o remetente Zabbix ou o protocolo do emissor Zabbix.

A verificação do item HTTP é executada pelo servidor Zabbix. No entanto, quando os hosts são monitorados por um proxy Zabbix, as verificações de itens HTTP são executadas pelo proxy.

As verificações de item HTTP não requerem nenhum agente em execução em um host que está sendo monitorado.

O agente HTTP suporta HTTP e HTTPS. O Zabbix seguirá opcionalmente redirecionamentos (veja a opção Seguir redireciona abaixo). O número máximo de redirecionamentos é codificado para 10 (usando a opção cURL CURLOPT_MAXREDIRS).

Configuração

Para configurar um item HTTP:

- Vá para: Configuração → Hosts
- Clique nos itens na linha do host
- Clique em Criar item
- Insira os parâmetros do item no formulário

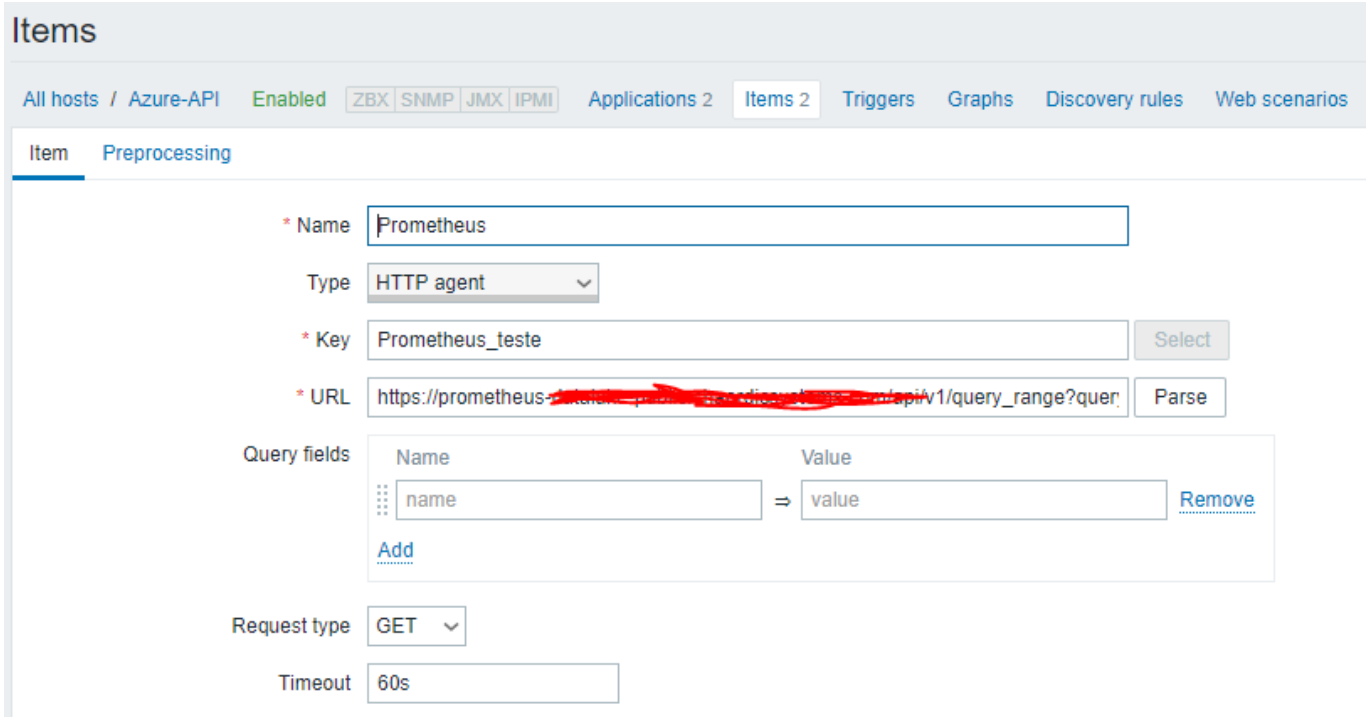
Todos os campos de entrada obrigatórios estão marcados com um asterisco vermelho.

Exemplos

Exemplo 1

Envie solicitações GET simples para recuperar dados do Prometheus.

Aqui temos um exemplo da chamada que será usada (alguns dados serão ocultos pois trata de um serviço em produção).



Para configurar devemos preencher os seguintes campos:

Name (Nome para o sensor)

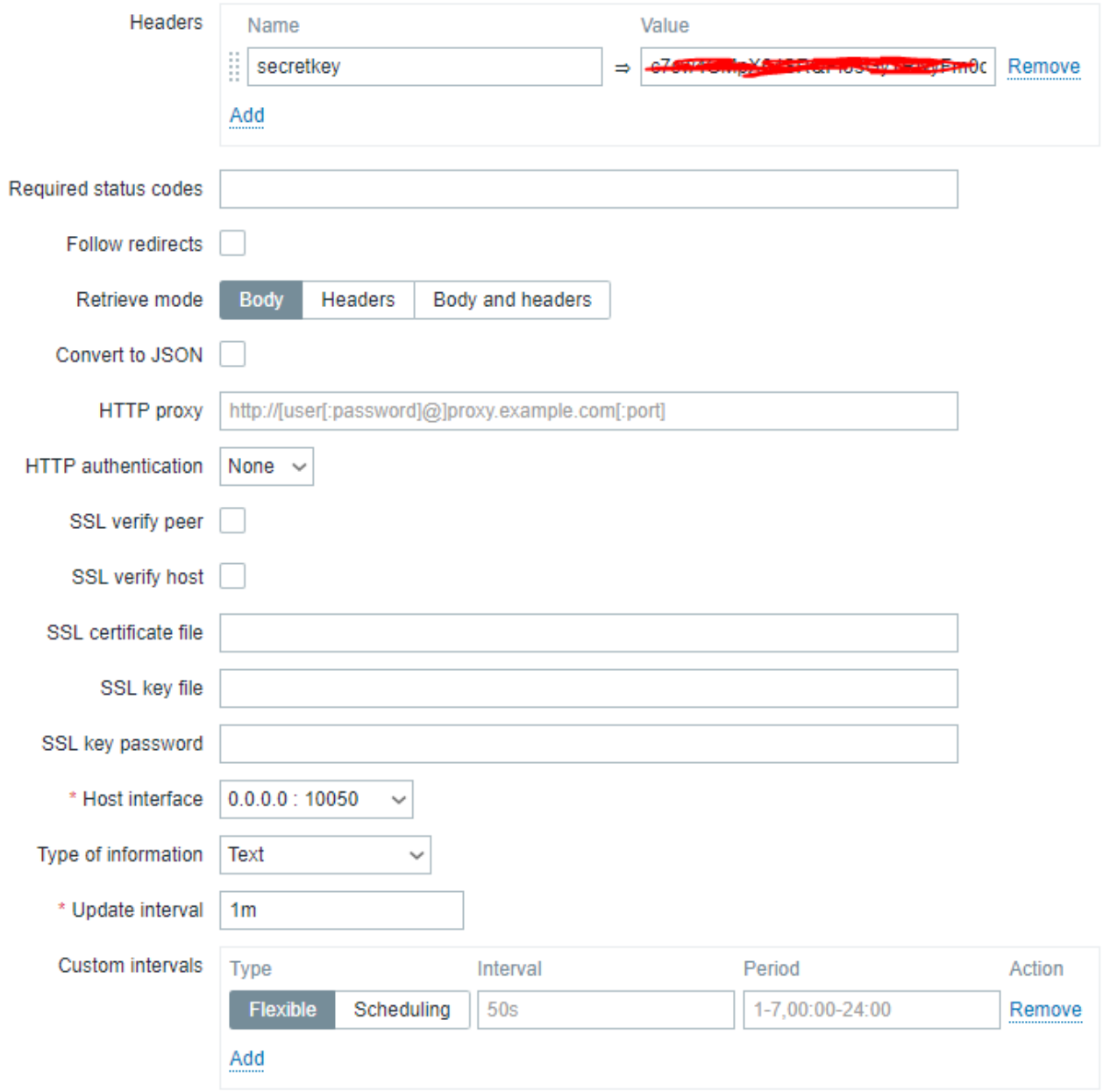
Type (definir como HTTP agent)

Key (coloque um nome de chave, recomendo usar o nome do sensor + alguma referência para o mesmo)

URL (endereço que vai ser chamado API)

Request type (defina conforme necessidade da API)

Timeout (recomendo avaliar o tempo que é necessário para carregar os dados)



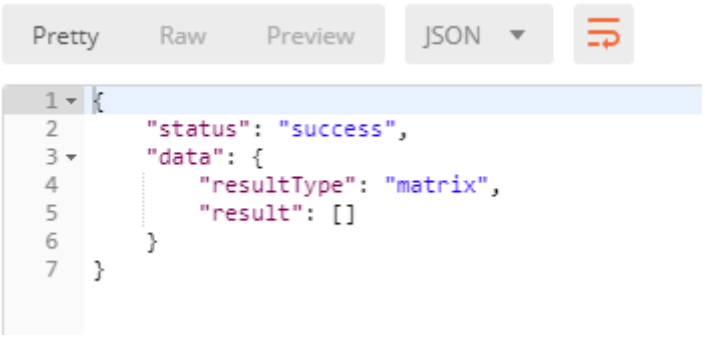
Headers (se necessário defina conforme especificação da API usada)

Type of information (Preencha com o tipo de retorno – texto ou número)

Update interval (Intervalo entre coletas)

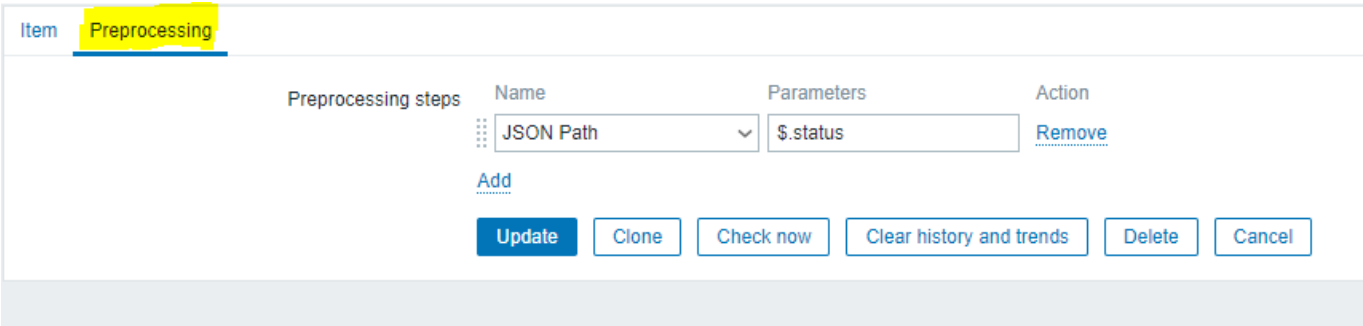
Editando o Preprocessing

Aqui eu tenho o seguinte retorno a partir da API:



Sendo assim vou pegar o nó que identifica o status de sucesso da minha aplicação.

No Zabbix:



Acesse a opção (Guia) Preprocessing.

Preencha os steps

Name -> JSON Path

Parameters -> $.status

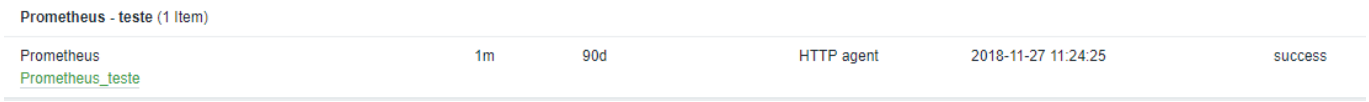
Em parameters é que está a solução desse sensor, para cada nível abaixo devemos colocar da seguinte forma.

Exemplo:

$.body.exemplo.exemplo1 (aqui faremos a leitura do valor em exemplo1)

Após colocar as informações necessárias clique em ADD.

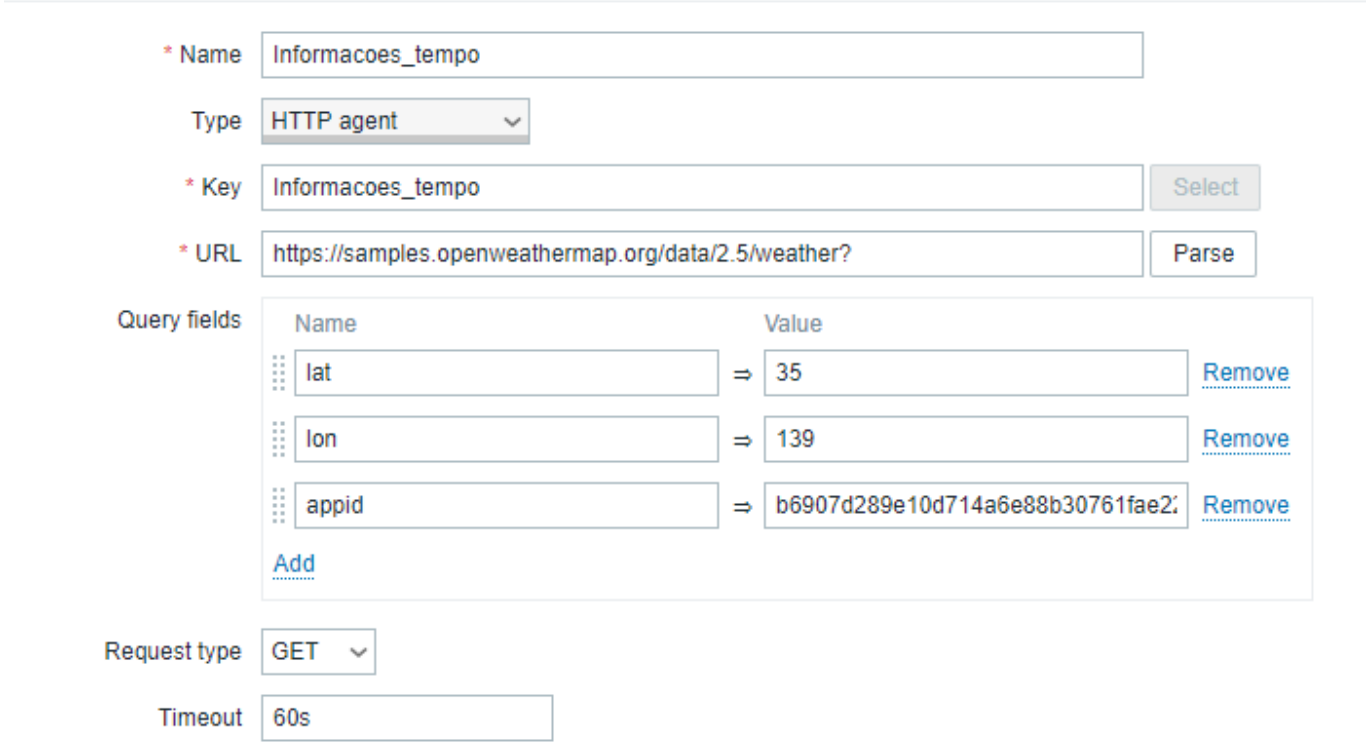
Alguns minutos depois já pode conferir o resultado em “Latest Data”



Acima coloquei um exemplo real de uso (meu caso), mas como também quero que todos aprendam aqui vai um exemplo usando a API do https://openweathermap.org

Vamos lá.

Primeiro vamos configurar os campos:



Devemos colocar o Name, Key, URL e aqui vamos definir o Query fields.

Em URL coloque: https://samples.openweathermap.org/data/2.5/weather?

Em Query fields: lat = 35

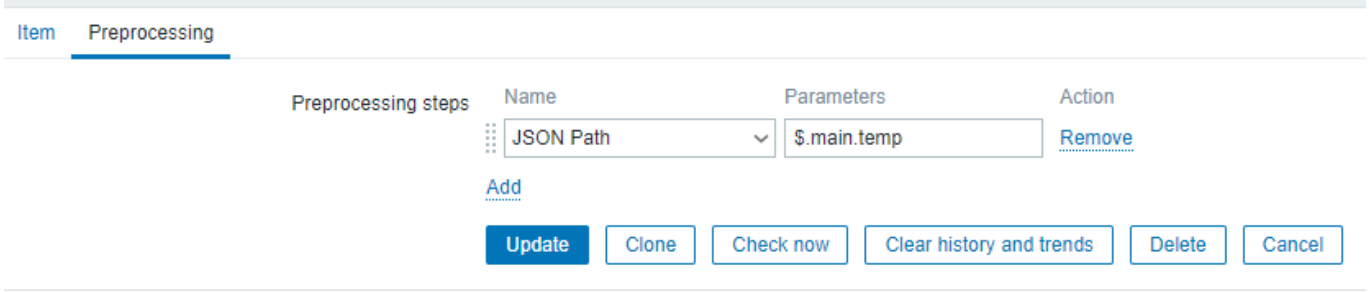
lon = 139

appid = b6907d289e10d714a6e88b30761fae22

Request type: GET

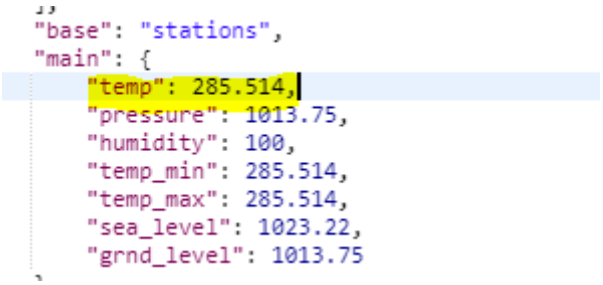
Type of information: Numeric (float)

Agora Ajustando o Preprocessing



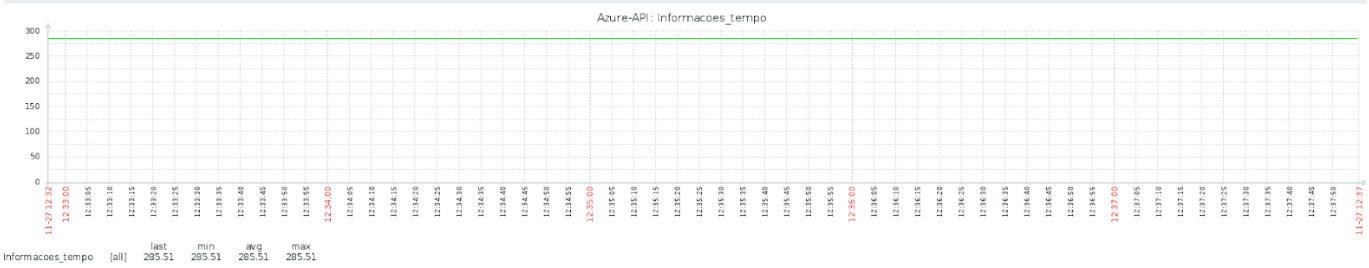
O mapeamento deve ficar dessa forma: $.main.temp

A informação capturada será:



Após finalizado clique em Add.

Alguns minutos após pode conferir em “Latest Data” e você terá todos os dados, como estamos pegando um valor numérico o gráfico já ficará configurado conforme imagem abaixo.



Bem pessoal, espero que estas informações sejam úteis e possam ajudar a todos.

E como sempre qualquer dúvida fico a disposição.

Abraço a todos.