



Geralmente quando a gente precisa trabalhar com user UserParameter já bate um desespero principalmente pra quem está começando.

E acredititem é perfeitamente normal, depois passa.

Mas hoje vamos mudar um pouco essa situação e criar um monitoramento de “Conexões Ativas” e de brinde vamos definir a porta em que queremos ter essa informação segregada.

No exemplo vou realizar essa configuração em uma maquina Windows Server 2016.

E vou criar um item no Host alvo do Zabbix.

Primeiro Passo – adicionar o UserParameter.

Nota: nos comandos usados pelo UserParameter, estarei usando o netstat

[Pra quem não conhece clique aqui.](#)

Primeiro vamos testar o netstat

Abra o CMD do Windows e digite: netstat -ano | find /C "ESTABLISHED"

```
C:\Windows\system32\cmd.exe
Microsoft Windows [versão 10.0.18363.657]
(c) 2019 Microsoft Corporation. Todos os direitos reservados.

C:\Users\4010225>netstat -ano | find /C "ESTABLISHED"
67
```

Mas então o que fizemos?

Sendo direto usamos o netstat -ano (para trazer todas as conexões)

o find /C "ESTABLISHED" (contar todas as conexões estabelecidas)

O resultado na minha maquina foi de 67 conexões ativas.

Muito simples não é mesmo?

Agora vamos para o Zabbix Agent que está rodando no servidor.

Lá você deve incluir o seguinte argumento:

“Recomendo incluir na sessão de parameter (se atente as informações comentadas)”

```
UserParameter=sockstat.tcp.inuse.count, netstat -ano | find /C "ESTABLISHED"
```

Após incluir o argumento, salve o arquivo e reinicie o Zabbix Agent.

E vamos ao Zabbix agora.

Você deve ter observado que colocamos alguns dados a mais no argumento e isso é simples de entender.

```
UserParameter=sockstat.tcp.inuse.count, netstat -ano | find /C "ESTABLISHED"
```

O Primeiro informa que é um UserParameter

Na sequência a chave do parâmetro e um dos mais importantes dados.

Por fim o comando que será executado.

Com isso já conseguimos seguir para criação do item conforme exemplo.

**Items**

All templates / TCP/UDP Stats Windows Applications 1 Items 12 Triggers Graphs 1 Screens Discovery rules Web scenarios

Item Preprocessing

* Name	TCP Sockets in ESTABLISHED Count												
Type	Zabbix agent												
* Key	sockstat.tcp.inuse.count												
Type of information	Numeric (unsigned)												
Units	sockets												
* Update interval	300s												
Custom intervals	<table border="1"> <thead> <tr> <th>Type</th> <th>Interval</th> <th>Period</th> <th>Action</th> </tr> </thead> <tbody> <tr> <td>Flexible</td> <td>50s</td> <td>1-7,00:00-24:00</td> <td>Remove</td> </tr> <tr> <td>Add</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Type	Interval	Period	Action	Flexible	50s	1-7,00:00-24:00	Remove	Add			
Type	Interval	Period	Action										
Flexible	50s	1-7,00:00-24:00	Remove										
Add													

Neste ponto você deve colocar o nome do Item.

Selecionar o type para Zabbix agent, tenha ciência que é um pre req. para esse monitoramento.

Key: Deve ser incluído o valor da chave que foi criada no passo anterior.

Type of information, deve seguir o tipo de dado obtido neste caso um inteiro.

Units, defina como sockets.

Update interval: Conforme a sua regra de negocio exigir.

No fim clique em Add, aguarde alguns instantes e o valor ficara disponível na ferramenta.

Exemplo abaixo:

TCP Sockets in ESTABLISHED Count sockstat.tcp.inuse.count	300s	3d	7d	Zabbix agent	2020-03-12 14:20:23	187 sockets
--------------------------------------------------------------	------	----	----	--------------	---------------------	-------------

Perfeito, concluimos o monitoramento das conexões usando o UserParameter.

Agora como prometido vou passar para vocês o Parâmetro para obter o valor por porta:

```
UserParameter=sockstat.tcp.count6005, netstat -an | find "6005" /c
```

Observem que a unica diferença foi adicionar a porta, muito simples.

Atente-se que os parâmetros são incluídos nos arquivos de configuração do Agent Ok.

## Surpresa

Para que todos fiquem felizes.

Abaixo vou deixar uma listinha com todos os UserParameter que utilizo, assim já facilita pra quem esta começando e serve para estudo também, espero que aproveitem.

- UserParameter=sockstat.sockets, netstat -ano | find /V "hola" /C
- UserParameter=sockstat.tcp.inuse, netstat -ano | findstr ESTABLISHED
- UserParameter=sockstat.tcp.inuse.count, netstat -ano | find /C "ESTABLISHED"
- UserParameter=sockstat.tcp.orphan, netstat -ano | findstr CLOSE\_WAIT
- UserParameter=sockstat.tcp.orphan.count, netstat -ano | find /C "CLOSE\_WAIT"
- UserParameter=sockstat.tcp.timewait, netstat -ano | findstr TIME\_WAIT
- UserParameter=sockstat.tcp.timewait.count, netstat -ano | find /C "TIME\_WAIT"
- UserParameter=sockstat.tcp.allocated, netstat -ano | findstr LISTENING
- UserParameter=sockstat.tcp.allocated.count, netstat -ano | find /C "LISTENING"
- UserParameter=sockstat.udp.inuse, netstat -ano | findstr UDP
- UserParameter=sockstat.udp.inuse.count, netstat -ano | find /C "UDP"

É isso pessoal, espero que o conteúdo ajude a quem está com esse tipo de dúvida, porque apesar de simples no inicio de carreira considero um dos pontos mais complexos.

Abraços e bons estudos.