



Comandos remotos via agente Zabbix

Aécio Pires e Willeson Souza

JOÃO PESSOA-PB

OUT/2013

Histórico de Atualizações

Data	Versão	Responsável	Alterações
22 out 2013	1.0	Aécio Pires e Willeson Souza	Versão inicial

Licença de Uso



Este trabalho está licenciado sob uma Licença Creative Commons Atribuição-Uso Não-Comercial 2.5 Brasil. Para ver uma cópia desta licença, visite <http://creativecommons.org/licenses/by-nc/2.5/br/> ou envie uma carta para Creative Commons, 171 Second Street, Suite 300, San Francisco, California 94105, USA.

Sumário

1- Introdução.....	4
2- Configurando o agente Zabbix.....	5
3- Configurando o sudo no host GNU/Linux cliente	8
4- Configurando a action para reiniciar o Apache 2.....	8
5- Testando a Ação.....	12

Comandos remotos via agente Zabbix

1 - Introdução

O objetivo deste tutorial é executar comandos remotos via agente Zabbix. Com comandos remotos, você pode definir que um determinado comando pré-definido seja automaticamente executado no host monitorado quando for atendida alguma condição. Assim, os comandos remotos são um mecanismo para monitoramento pró-ativo inteligente.

Nos usos mais óbvios deste recurso, você pode tentar reiniciar automaticamente alguns serviços (servidor web, middleware, CRM) ou usar o comando IPMI 'reboot' para reiniciar algum servidor remoto, se ele não responder as requisições.

Atenção:

Comandos remotos, na versão 2.0.9, não são suportados em proxies Zabbix. Então é necessária uma conexão direta entre o servidor Zabbix e o agente.

Neste tutorial será mostrado como reiniciar o Apache automaticamente em um host com Debian 7.0 monitorado por um servidor Zabbix 2.0.9. A partir deste caso de uso, você pode pegar a ideia geral e adaptar de acordo com a sua necessidade.

O ambiente de testes é mostrado na Figura 1:

Aqui, o cliente é monitorado. É criada uma condição de monitoramento, na qual se o Apache ficar determinado tempo sem responder as requisições, será reiniciado

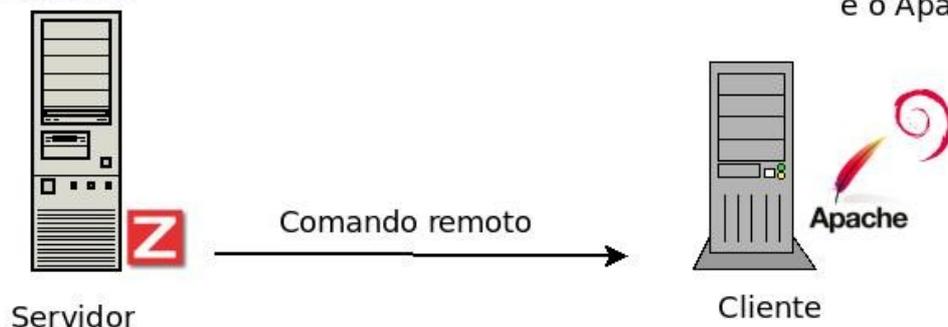


Figura 1: Ambiente de testes.

2 - Configurando o agente Zabbix

Os comandos remotos são executados no agente Zabbix e podem ser scripts personalizados. No arquivo de configuração **/usr/local/etc/zabbix_agentd.conf**, certifique-se de que o parâmetro **EnableRemoteCommands** esteja definido com o valor **1** e descomentado.

Antes:

```
# EnableRemoteCommands=0  
# LogRemoteCommands=0
```

Depois:

```
EnableRemoteCommands=1  
LogRemoteCommands=1
```

Depois disso reinicie o daemon do agente Zabbix.

```
$ sudo /etc/init.d/zabbix-agentd stop  
$ sudo /etc/init.d/zabbix-agentd start
```

Atenção:

A localização do arquivo `zabbix_agentd.conf` e o comando para reiniciar o agente Zabbix varia de acordo com os procedimentos de instalação do Zabbix em cada distro GNU/Linux.

Na instalação do agente Zabbix, foi configurado um item e uma trigger específica para monitorar o Apache2. Você pode ver como o item e a trigger foram criadas nas Figuras 2 e 3.

Host:

Nome:

Tipo:

Chave:

Interface do host:

Tipo de informação:

Tipo de dados:

Unidades:

Usar multiplicador customizado:

Intervalo atualização (em seg):

Intervalo flexível

Intervalo	Período	Ação
Não foram definidos intervalos flexíveis.		

Novo intervalo flexível

Intervalo (em segundos)	<input type="text" value="50"/>	Período	<input type="text" value="1-7,00:00-24:00"/>	<input type="button" value="Adicionar"/>
-------------------------	---------------------------------	---------	--	--

Manter histórico (em dias):

Manter estatísticas (em dias):

Armazenar valor:

Mostrar valor: [mostrar mapeamento de valores](#)

Nova aplicação

Aplicações

- General
- Memory
- Network interfaces
- OS
- Performance
- Processes**

Preenche o campo inventário do host:

Descrição:

Status:

Figura 2: Item para monitorar o funcionamento do Apache2.

Nome

Expressão

[Construtor de expressão](#)

Evento com geração de múltiplos incidentes

Descrição

URL

Risco

Ativo

Figura 3: Trigger para monitorar o funcionamento do Apache2.

A chave usada no item mostrado na Figura 2 foi obtido na página https://www.zabbix.com/documentation/2.0/manual/config/items/itemtypes/zabbix_agent conforme mostra a Figura 4.

net.tcp.service[service,<ip>,<port>]			
Checks if service is running and accepting TCP connections.	0 - service is down 1 - service is running	service - either of: <i>ssh, ntp, ldap, smtp, ftp, http, pop, nntp, imap, tcp, https, telnet</i> ip - IP address (default is 127.0.0.1) port - port number (by default standard service port number is used)	<i>Example key:</i> net.tcp.service[ftp,,45] - can be used to test the availability of <u>FTP</u> server on TCP port 45. Note that these checks may result in additional messages in system daemon logfiles (<u>SMTP</u> and <u>SSH</u> sessions being logged usually). Checking of encrypted protocols (like <u>IMAP</u> on port 993 or <u>POP</u> on port 995) is currently not supported. As a workaround, please use net.tcp.port for checks like these. Checking of <u>LDAP</u> and <u>HTTPS</u> by Windows agent is currently not supported. Note that the telnet check looks for a prompt (':' at the end). Old naming: check_service[*] Note that before Zabbix 1.8.3 version service.ntp should be used instead of ntp . https and telnet services are supported since Zabbix 2.0.

Figura 4: Informações sobre o uso da chave (key) net.tcp.service.

Segundo a documentação, quando for obtido um valor igual a 1 para a chave **net.tcp.service** significa que o serviço está ativo e recebendo as requisições. Quando for obtido um valor igual a 0, significa que o serviço está inativo.

A execução do comando remoto para reiniciar o Apache será baseado nos valores coletados para este item e trigger. Então, você pode criar com os mesmos parâmetros para testar o seu ambiente.

Vale citar que **debian7** é o nome do host cliente usado neste teste. Você deve mudar o nome do host cliente de acordo com o seu ambiente

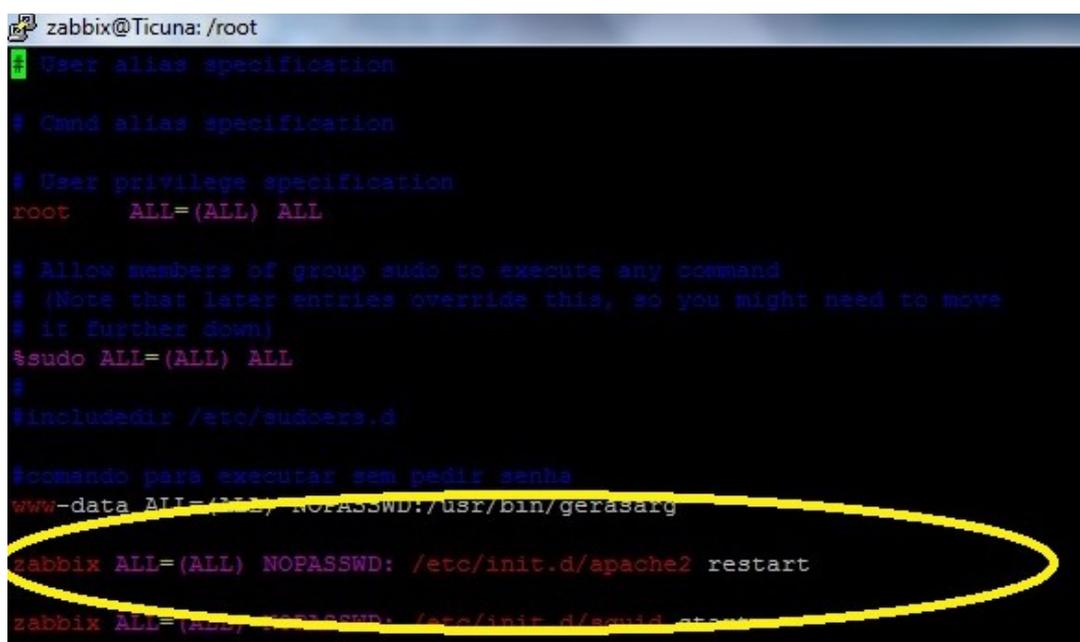
3 - Configurando o sudo no host GNU/Linux cliente

Digite o comando **visudo**, ele abrirá o arquivo **/etc/sudoers**. No final desse arquivo, adicione a linha abaixo para dar a permissão ao usuário zabbix de reiniciar o Apache sem solicitar senha.

visudo

Linha a ser adicionada ao arquivo.

```
zabbix ALL=(ALL) NOPASSWD: /etc/init.d/apache2 restart
```



```
zabbix@Ticuna: /root
# User alias specification

# Command alias specification

# User privilege specification
root    ALL=(ALL) ALL

# Allow members of group sudo to execute any command
# (Note that later entries override this, so you might need to move
# it further down)
%sudo  ALL=(ALL) ALL
#

#includedir /etc/sudoers.d

#comando para executar sem pedir senha
www-data ALL=(ALL) NOPASSWD: /usr/bin/gerasaig
zabbix ALL=(ALL) NOPASSWD: /etc/init.d/apache2 restart
zabbix ALL=(ALL) NOPASSWD: /etc/init.d/squid restart
```

Figura 5: Configurando a permissão para o usuário Zabbix do equipamento cliente executar um comando sem pedir senha do sudo.

4 - Configurando a action para reiniciar o Apache 2

Acesse o menu **Configuração (Configuration) > Ações (Actions)**. Clique no botão **Criar Ação (Create Action)**.

Conforme a Figura 6, preencha os campos da aba **Ação (Action)**.

Monitoramento | Inventário | Relatórios | **Configuração** | Administração | Extras

Grupos de hosts | Templates | Hosts | Manutenção | Web | **Ações** | Telas | Slideshows | Mapas | Autobusca

Histórico: Configuração de hosts » Dashboard » Gráficos personalizados » Dashboard » Configuração de ações

CONFIGURAÇÃO DE AÇÕES

Ação | Condições | Ações

Nome:

Duração padrão de cada passo da operação: (mínimo de 60 segundos)

Assunto padrão:

Mensagem padrão:

Mensagem da recuperação:

Ativo:

Figura 6: Preenchendo a aba Ações.

Nome (Name): {HOSTNAME} DAEMON APACHE OFF

Assunto padrão (Default subject): {TRIGGER.STATUS}:
{TRIGGER.NAME}

Mensagem padrão (Default message): {TRIGGER.NAME}:
{TRIGGER.STATUS} Last value: {ITEM.LASTVALUE} Por favor verifique
o serviço.

Conforme a Figura 7, preencha os campos da aba **Condições**
(**Condictions**).

Ação	Condições	Ações
Tipo do cálculo <input type="text" value="E / OU"/> (A) e (B)		
Condições		
	Texto	Nome
(A)	Status de manutenção não em "manutenção"	Remover
(B)	Valor da trigger = "INCIDENTE"	Remover
Nova condição		
	<input type="text" value="Trigger"/>	<input type="text" value="="/>
	<input type="text" value="debian7: O Apache está ativo?"/>	
	Adicionar	
	Selecionar	
<input type="button" value="Salvar"/> <input type="button" value="Cancelar"/>		

Figura 7: Definindo as condições.

Tipo de cálculo (Type of calculation): E/OU (AND/OR)

Condições (Conditions): (A) Status de manutenção não em "manutenção" (Maintenance status not in "maintenance")

(B) Valor da trigger = "INCIDENTE" (Trigger value = "PROBLEM")

Nova condição (New condition): Escolha a opção trigger referente ao monitoramento do apache2. Neste caso é a trigger "O Apache está ativo?" do host debian7. Clique em **Adicionar (Add)**.

Conforme a Figura 8, preencha os campos da aba **Ações (Actions)**. Clique no botão **Nova (New)**.

Nesse caso o Apache será reiniciado conforme mencionado acima.

5 - Testando a Ação

Agora teste o funcionamento da ação. Acesse o host cliente e pare a execução do Apache.

```
$ sudo /etc/init.d/apache2 stop
```

Agora veja no dashboard do Zabbix a trigger sendo acionada.

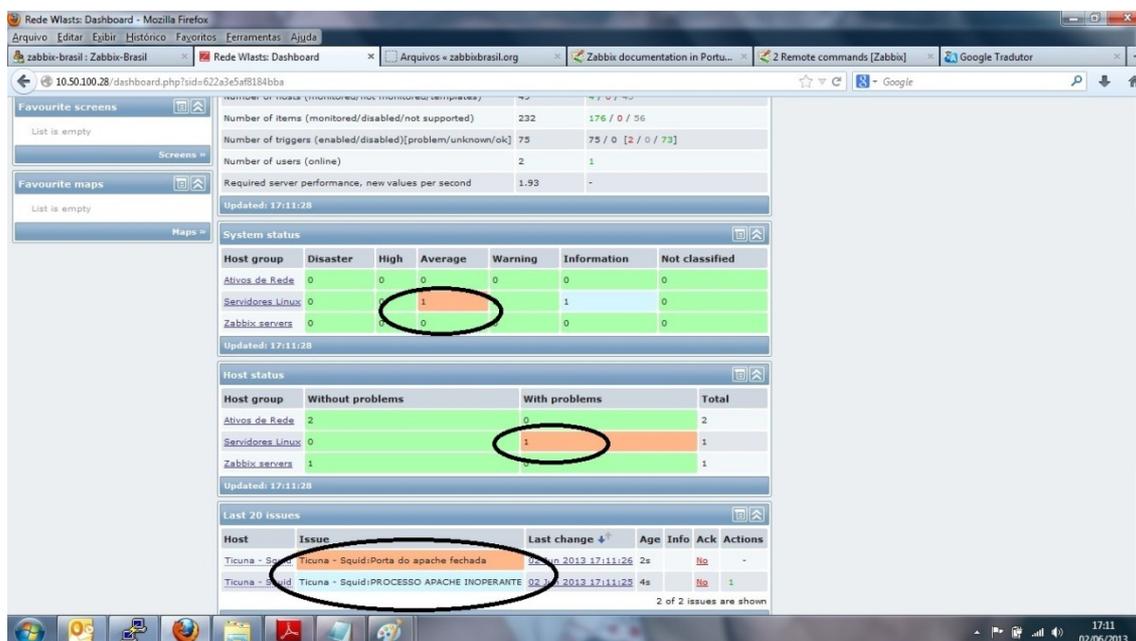


Figura 9: Dashboard alertando que o Apache parou de ser executado.

Verifique o log do zabbix agente em `/tmp/zabbix_agentd.log` do host cliente para ver se o agente Zabbix irá mostrar o comando remoto sendo executado.

```
zabbix@debian7: /root
Arquivo Editar Ver Pesquisar Terminal Ajuda
4032:20131022:155827.750 Zabbix Agent stopped. Zabbix 2.0.9 (revision 39085).
5883:20131022:155829.429 Starting Zabbix Agent [debian7]. Zabbix 2.0.9 (revision 39085).
5887:20131022:155829.430 agent #3 started [listener]
5886:20131022:155829.430 agent #2 started [listener]
5885:20131022:155829.430 agent #1 started [listener]
5884:20131022:155829.431 agent #0 started [collector]
5885:20131022:160008.568 Executing command 'sudo /etc/init.d/apache2 restart'
```

Figura 10: Log do agente Zabbix mostrando que o Apache será reiniciado.

Após ter verificado se o Zabbix iniciou o Apache volte ao dashboard e veja se as triggers não é mais acionada.

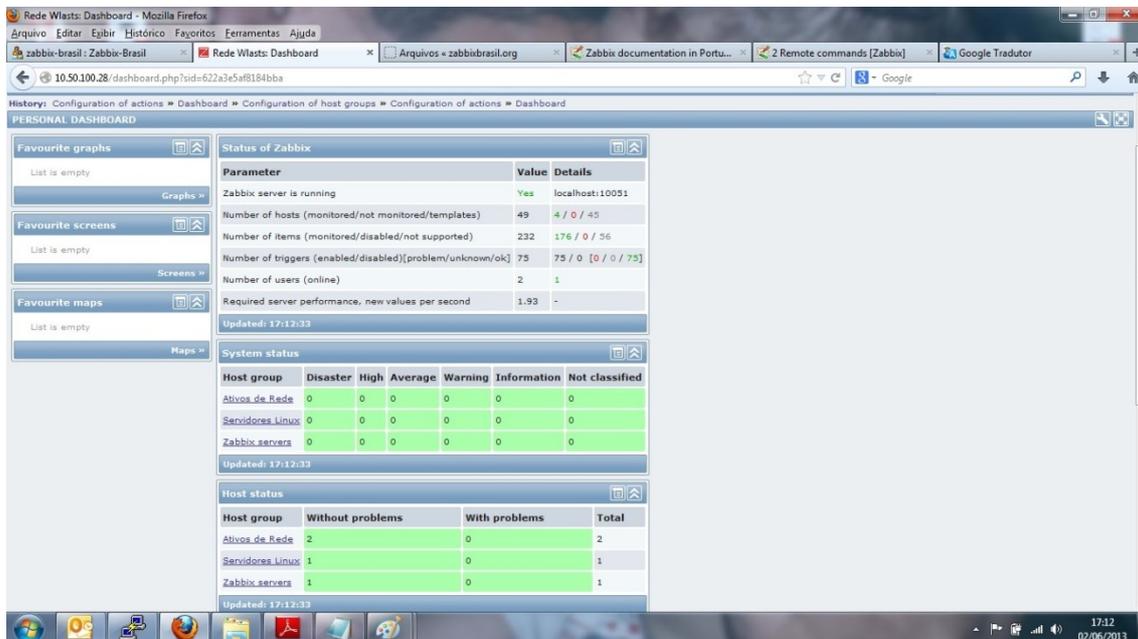


Figura 11: Dashboard alertando que o Apache voltou a funcionar.

Se tudo deu certo, no dashboard não irá aparecer mais mensagem de que o Apache está inoperante.

Considerações finais

Neste tutorial foi mostrada a execução de comandos remotos no Zabbix 2.0.9 no Debian 7.

No site da comunidade Zabbix Brasil http://zabbixbrasil.org/?page_id=7, mais especificamente na página, você pode encontrar outros tutoriais que mostram a instalação dos componentes Zabbix

Agent e Zabbix Proxy, além de mostrar como usar a interface Web do Zabbix para monitorar alguns tipos de equipamentos.

Abraço e que Deus o(a) abençoe. Leia o Salmo 55:22. Jesus é bom, te ama e quer salvar tua alma. :-)

Referências

Manual do Zabbix 2.0. Disponível em: https://www.zabbix.com/documentation/2.0/manual/config/notifications/action/operation/remote_command Acessado em: 22 de outubro de 2013.

[ZABBIX REQUIREMENTS] Zabbix Requirements. Disponível em: <http://www.zabbix.com/documentation/2.0/manual/installation/requirements> Acessado em: 22 de outubro de 2013.