

Task. Zabbix Tools

Testing Infrastructure:

Vagrantfile to spin up 2 VMs (virtualbox):

- zabbix server, provisioned by Vagrant provisioner
- Linux VM with zabbix agent, script for registration on zabbix server, all provisioned by Vagrant provisioner

Task:

1. Configure the agent for replying to the specific server in passive and active mode.

```
sed -i -e '/Server=127.0.0.1/s/127.0.0.1/192.168.33.11/'  
/etc/zabbix/zabbix_agentd.conf
```

```
sed -i -e '/ServerActive=127.0.0.1/s/127.0.0.1/192.168.33.11/'  
/etc/zabbix/zabbix_agentd.conf
```

2. Use zabbix_sender to send data to server manually (use zabbix_sender with key -vv for maximal verbosity).

All hosts / zabbix2 Enabled ZBX SNMP JMX IPMI Applications 10 Items 32 Triggers 15 Graphs 5 Discovery rules 2 Web

Name

Type

Key

Type of information

History storage period (in days)

Allowed hosts

New application

Applications

-None-

CPU

Filesystems

General

Memory

Network interfaces

OS

Performance

Processes

Security

Populates host inventory field

Description

Enabled ☒

```
[root@zabbix2 ~]# zabbix_sender -z 192.168.33.11 -s "zabbix2" -k zabbix.sender -o 777 -vv  
zabbix_sender [22647]: DEBUG: answer [{"response":"success","info":"processed: 1; failed: 0; total: 1;  
seconds spent: 0.000237"}]  
info from server: "processed: 1; failed: 0; total: 1; seconds spent: 0.000237"  
sent: 1; skipped: 0; total: 1  
[root@zabbix2 ~]#
```

ZABBIX Monitoring Inventory Reports Configuration Administration

Dashboard Problems Overview Web Latest data Triggers Graphs Screens Maps Discovery IT services

Latest data

Filter ▲

Host groups: type here to search Name:

Hosts: **zabbix2** type here to search Show items without data: ☒

Application: type here to search Show details: ☐

Name ▲	Last check	Last value	Change
▶ CPU (13 items)			
▶ General (5 items)			
▶ Memory (5 items)			
▶ OS (8 items)			
▶ Performance (13 items)			
▶ Processes (2 items)			
▶ Security (2 items)			
▶ Zabbix agent (3 items)			
▼ - other - (1 item)			
<input type="checkbox"/> sender	2018-07-05 13:44:27	777	History

3. Use `zabbix_get` as data receiver and examine zabbix agent sending's.
For both VMs use vagrant box "sbeliakou/centos-7.3-x86_64-minimal"

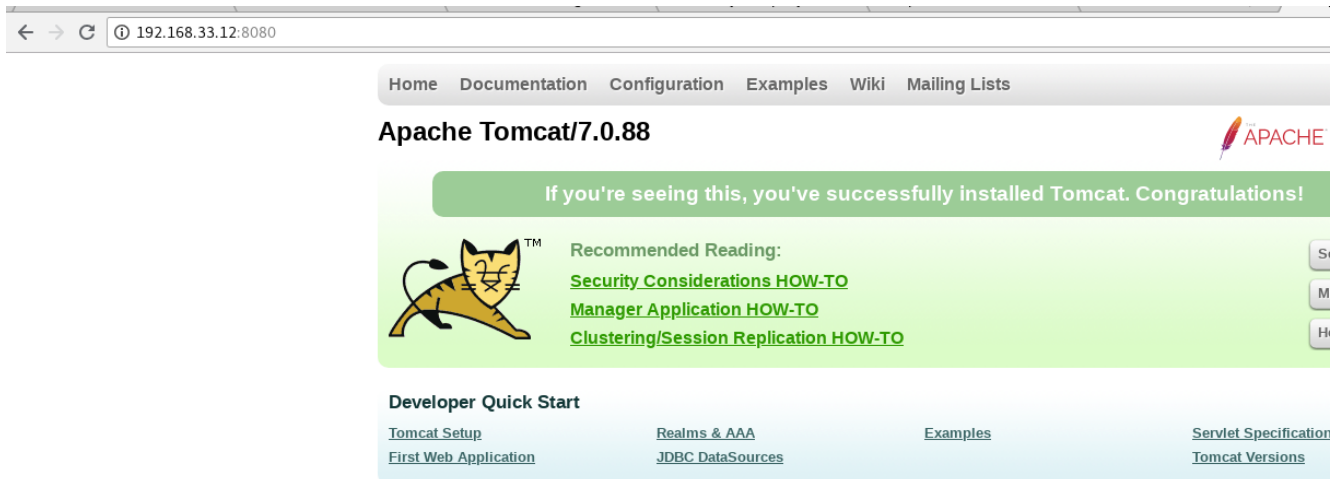
```
[root@zabbix1 ~]# zabbix_get -s 192.168.33.12 -p 10050 -k 'system.users.num'
1
[root@zabbix1 ~]# zabbix_get -s 192.168.33.12 -p 10050 -k 'system.hostname'
zabbix2
[root@zabbix1 ~]# zabbix_get -s 192.168.33.12 -p 10050 -k 'vm.memory.size[total]'
1928577024
[root@zabbix1 ~]#
```

Task. Web Monitoring with Zabbix

Testing Infrastructure:

Vagrantfile to spin up 2 VMs (virtualbox):

- zabbix server, provisioned by Vagrant provisioner
- Zabbix agents on both VMs, provisioned by Vagrant provisioner
- Install Tomcat 7 on 2nd VM, deploy any “hello world” application



Cluster - HA JSP Sample

HttpSession Information:

- Served From Server: **192.168.33.12**
- Server Port Number: **8080**
- Executed From Server: **zabbix2**
- Executed Server IP Address: **127.0.0.1**
- Session ID: **641A259EB6319CF2536B46470FB01F5C**
- Session Created: Thu Jul 05 12:52:49 BST 2018
- Last Accessed: Thu Jul 05 12:52:49 BST 2018
- Session will go inactive in **1800 seconds**

Enter session attribute data:

Name of Session Attribute:

Value of Session Attribute:

Data retrieved from the HttpSession:

INSTRUCTIONS

- Add session data using the form. Upon pressing ADD SESSION DATA, the current session data will be listed.
- Click on RELOAD PAGE to display the current session data without adding new data.
- Click on CLEAR SESSION to invalidate the current session.

Tasks:

4. Configure WEB check:

1. Scenario to test Tomcat availability as well as Application health

Web monitoring

[All hosts](#) / [zabbix2](#) [Enabled](#) [ZBX](#) [SNMP](#) [JMX](#) [IPMI](#) [Applications 10](#) [Items 47](#) [Triggers 19](#) [Graphs 9](#) [Discovery rules 2](#) [Web scenarios](#)

[Scenario](#) [Steps](#) [Authentication](#)

Name

Application

New application

Update interval (in sec)

Attempts

Agent

HTTP proxy

Variables

Headers

Enabled ☒

Add

Cancel

Name

URL

Post

Variables

Headers

Follow redirects ☒

Retrieve only headers ☐

Timeout

Required string

Required status codes

Name

FirstPage

URL

http://192.168.33.12:8080/clusterjsp

Post

Variables

Headers

Follow redirects

☒

Retrieve only headers

☐

Timeout

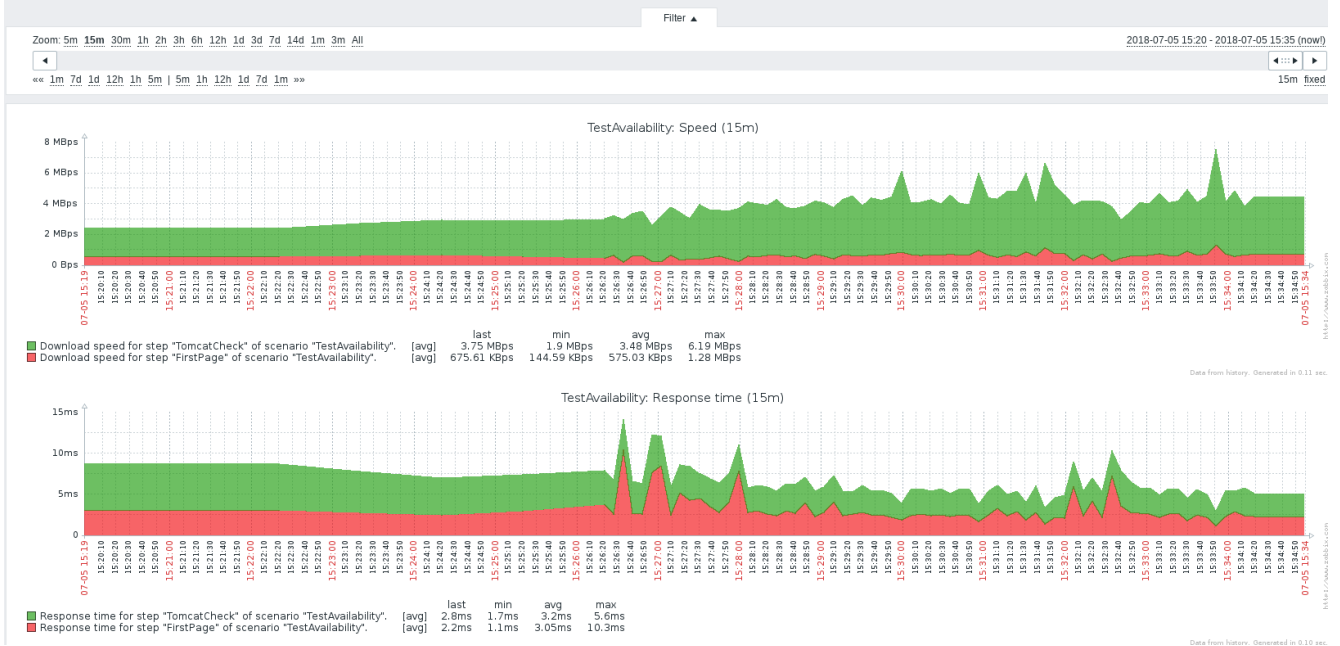
15

Required string

Required status codes

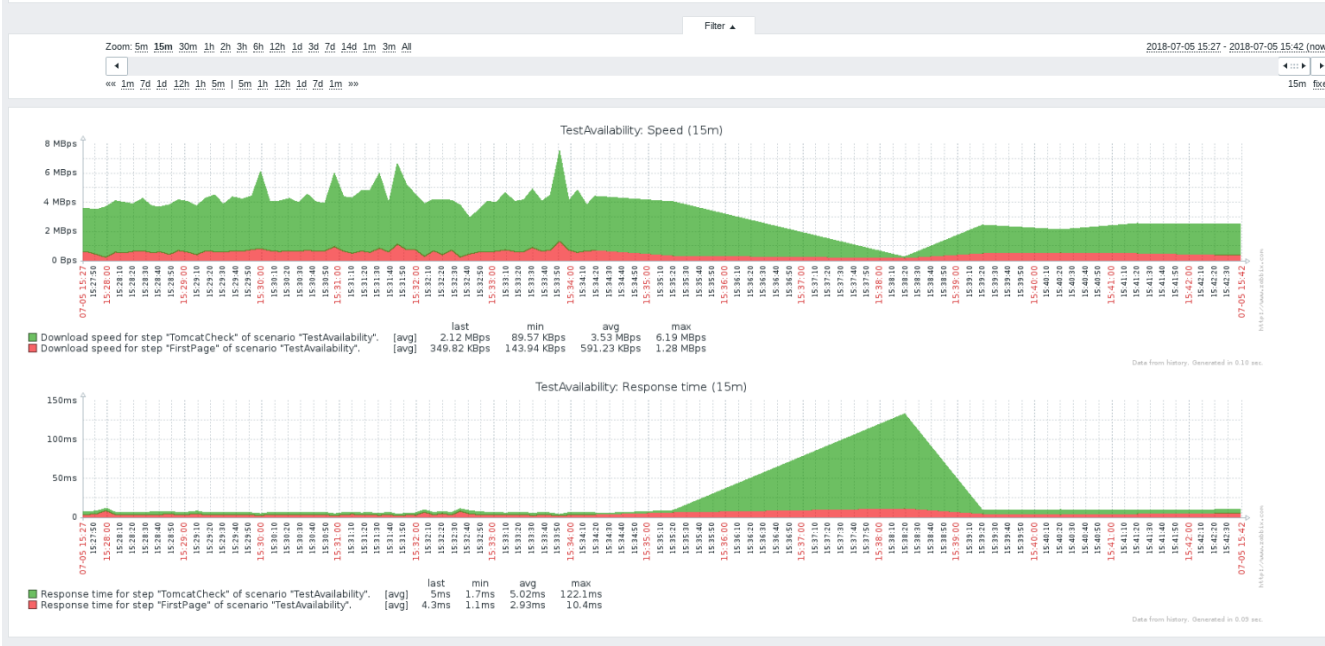
200

TomcatCheck	3.75 MBps	2.8ms	200	OK
FirstPage	675.61 KBps	2.2ms	200	OK
TOTAL		5ms		OK



After Tomcat restart:

Step	Speed	Response time	Response code	Status
TomcatCheck	2.06 MBps	5.2ms	200	OK
FirstPage	472.99 KBps	3.2ms	200	OK
TOTAL		8.4ms		OK



5. Configure Triggers to alert once WEB resources become unavailable

Trigger

Dependencies

Name

UnavailableResource

Severity

Not classified

Information

Warning

Average

High

Disaster

Problem expression

{zabbix2.web.test.fail[TestAvailability].last(0)}>0

Add

Expression constructor

OK event generation

Expression

Recovery expression

None

Recovery expression

{zabbix2.web.test.fail[TestAvailability].last(0)}=0

Add

Expression constructor

PROBLEM event generation mode

Single

Multiple

OK event closes

All problems

All problems if tag values match

Tags

tag

value

Remove

Add

Allow manual close

URL

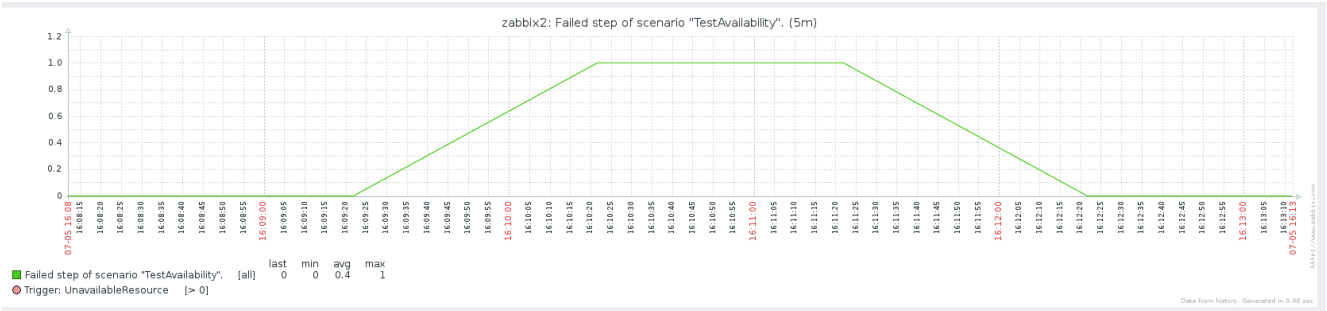
Description



Enabled

Add

Cancel

After Tomcat restart



Time ▾	 Severity	Recovery time	Status	Info	Host	Problem	Duration	Ack	Actions	Tags
16:10:22 ▾	 Disaster	16:12:22	RESOLVED		zabbix2	UnavailableResource	2m	No		

Displaying 1 of 1 found

For both VMs use vagrant box “sbeliakou/centos-7.3-x86_64-minimal”

Task. Java Monitoring with Java

Task:

You should install and configure Zabbix server and agents.

Testing Infrastructure:

Vagrantfile to spin up 2 VMs (virtualbox):

- zabbix server, provisioned by Vagrant provisioner
- Zabbix agents on both VMs, provisioned by Vagrant provisioner
- Install Tomcat 7 on 2nd VM

Tasks:

6. Configure Zabbix to examine Java parameters via Java Gateway (<http://jmxmonitor.sourceforge.net/jmx.html>)

```
[root@zabbix2 tomcat]# vim apache-tomcat-7.0.88/bin/setenv.sh
```

```
CATALINA_OPTS=" \
-Dcom.sun.management.jmxremote \
-Djava.rmi.server.hostname=192.168.33.12 \
-Dcom.sun.management.jmxremote.local.only=false \
-Dcom.sun.management.jmxremote.port=12345 \
-Dcom.sun.management.jmxremote.rmi.port=12345 \
-Dcom.sun.management.jmxremote.authenticate=false \
-Dcom.sun.management.jmxremote.ssl=false \
-Djava.net.preferIPv4Stack=true"
```

```
[root@zabbix2 tomcat]# chmod +x apache-tomcat-7.0.88/bin/setenv.sh
```

```
[root@zabbix2 tomcat]# source apache-tomcat-7.0.88/bin/setenv.sh
```

```
[root@zabbix2 tomcat]# cp catalina-jmx-remote.jar apache-tomcat-7.0.88/lib/
```

```
[root@zabbix2 tomcat]# ll apache-tomcat-7.0.88/lib | grep catalina-jmx-remote.jar
-rw-r--r-- 1 root root 13081 Jul  5 15:41 catalina-jmx-remote.jar
[root@zabbix2 tomcat]#
```

```
[root@zabbix2 tomcat]# vim apache-tomcat-7.0.88/conf/server.xml
```

```
<Listener className="org.apache.catalina.mbeans.JmxRemoteLifecycleListener"
  rmiRegistryPortPlatform="8097"
  rmiServerPortPlatform="8098"/>
```

```
[root@zabbix1 ~]# yum -y install zabbix-java-gateway (on Zabbix server)
```

```
[root@zabbix1 ~]# systemctl start zabbix-java-gateway
```

```
[root@zabbix1 ~]# systemctl enable zabbix-java-gateway
Created symlink from /etc/systemd/system/multi-user.target.wants/zabbix-java-gateway.service to /usr/lib/systemd/system/zabbix-java-gateway.service.
[root@zabbix1 ~]# systemctl restart zabbix-server
[root@zabbix1 ~]#
```

```
[root@zabbix1 ~]# vim /etc/zabbix/zabbix_server.conf (on Zabbix server):
```

```

### Option: JavaGateway
#   IP address (or hostname) of Zabbix Java gateway.
#   Only required if Java pollers are started.
#
# Mandatory: no
# Default:
JavaGateway=192.168.33.11

### Option: JavaGatewayPort
#   Port that Zabbix Java gateway listens on.
#
# Mandatory: no
# Range: 1024-32767
# Default:
JavaGatewayPort=10052

### Option: StartJavaPollers
#   Number of pre-forked instances of Java pollers.
#
# Mandatory: no
# Range: 0-1000
# Default:
StartJavaPollers=5

```

Add JMX interfaces in UI:

Host name

Visible name

Groups In groups Other groups

Discovered hosts

Hypervisors
Linux servers
Templates
Virtual machines
Zabbix servers

New group

Agent interfaces

IP address	DNS name	Connect to	Port
<input type="text" value="192.168.33.12"/>	<input type="text"/>	<input checked="" type="radio"/> IP <input type="radio"/> DNS	<input type="text" value="10050"/>
Add			

SNMP interfaces

[Add](#)

JMX interfaces

<input type="text" value="192.168.33.12"/>	<input type="text"/>	<input checked="" type="radio"/> IP <input type="radio"/> DNS	<input type="text" value="12345"/>
<input type="text" value="192.168.33.12"/>	<input type="text"/>	<input checked="" type="radio"/> IP <input type="radio"/> DNS	<input type="text" value="8097"/>
Add			

Name

JavaHeapMemoryUsed

Type

JMX agent

Key

jmx["java.lang:type=Memory",HeapMemoryUsage.committed]

Select

Host interface

192.168.33.12 : 8097

User name

Password

Type of information

Numeric (unsigned)

Data type

Decimal

Units

Use custom multiplier

☐

1

Update interval (in sec)

10

Custom intervals

Type	Interval	Period	Action
Flexible	Scheduling	50	1-7,00:00-24:00
<div>Add</div>			

History storage period (in days)

7

Trend storage period (in days)

365

Name	Applications	Items	Triggers	Graphs	Discovery	Web	Interface	Templates	Status	Availability	Agent enc
zabbix1	Applications 11	Items 78	Triggers 47	Graphs 14	Discovery 2	Web	127.0.0.1: 10050	Template App Zabbix Server, Template OS Linux (Template App Zabbix Agent)	Enabled	ZBX SNMP JMX IPMI	NONE
zabbix2	Applications 10	Items 48	Triggers 20	Graphs 9	Discovery 2	Web 1	192.168.33.12: 10050	Template OS Linux (Template App Zabbix Agent)	Enabled	ZBX SNMP JMX IPMI	NONE

7. Configure triggers to alert once these parameters changed.

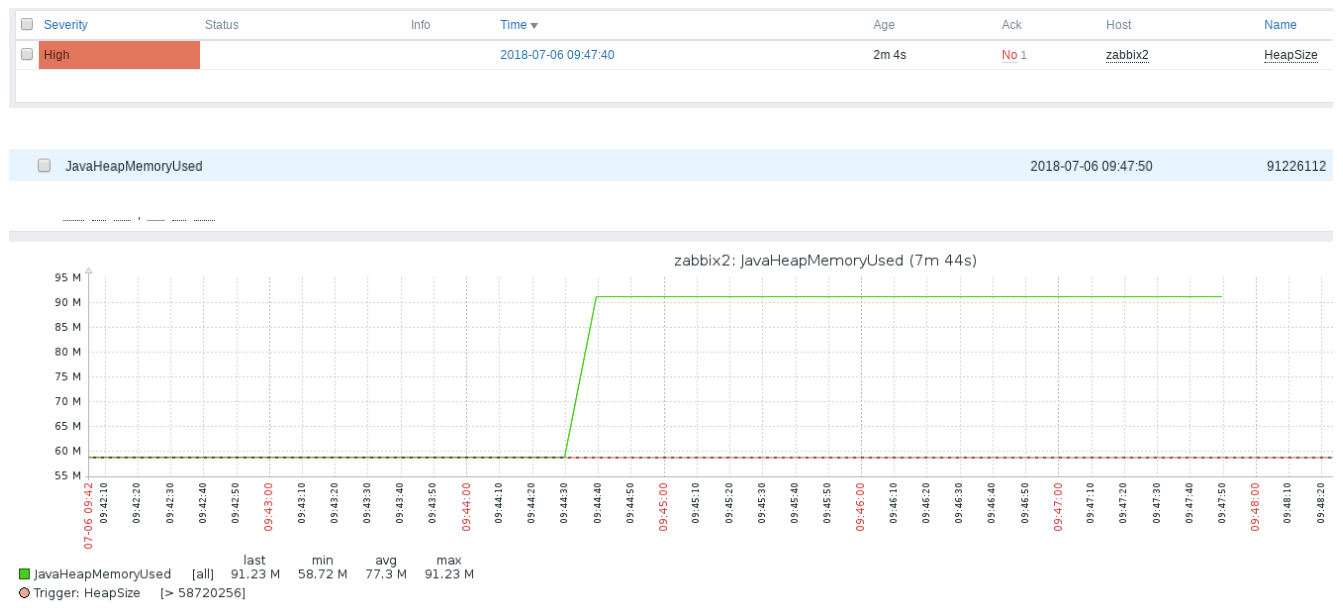
Java Heap Size ~58M

JavaHeapMemoryUsed	2018-07-06 09:43:00	58720256
--------------------	---------------------	----------

Configure trigger

Name	<input type="text" value="HeapSize"/>					
Severity	Not classified	Information	Warning	Average	High	Disaster
Problem expression	<input java.lang:type='Memory\",HeapMemoryUsage.committed].last()}>58720256"/' type="text" value="{zabbix2.jmx[\"/>					Add
Expression constructor						
OK event generation	Expression	Recovery expression	None			
Recovery expression	<input java.lang:type='Memory\",HeapMemoryUsage.committed].last()}<58720256"/' type="text" value="{zabbix2.jmx[\"/>					Add

After Tomcat shutdown

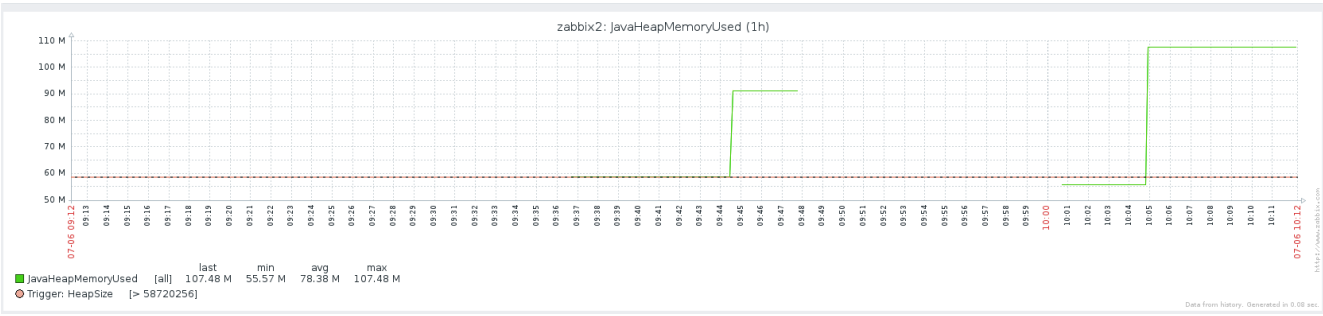


After removing one of an application and starting Tomcat

Severity	Status	Info	Time	Age	Ack	Host	Name
High	OK		2018-07-06 10:00:45	2m 36s	No 1	zabbix2	HeapSize

At start Java Heap Size ~55M, then it starts raising

JavaHeapMemoryUsed	2018-07-06 10:03:50	55574528
--------------------	---------------------	----------



Time	Severity	Recovery time	Status	Info	Host	Problem	Duration	Ack
10:05:00	High		PROBLEM		zabbix2	HeapSize	14m 18s	No
09:47:40	High	10:00:45	RESOLVED		zabbix2	HeapSize	13m 5s	No

For both VMs use vagrant box “sbeliakou/centos-7.3-x86_64-minimal”

Task. Zabbix API

Task:

You should develop a script (on Python 2.x) which registers given host in Zabbix.

Testing Infrastructure:

Vagrantfile to spin up 2 VMs (virtualbox):

- zabbix server, provisioned by Vagrant provisioner
- Linux VM with zabbix agent, script for registration on zabbix server, all provisioned by Vagrant provisioner

Registering Script requirements:

8. Written on Python 2.x
9. Starts at VM startup or on provision phase
10. Host registered in Zabbix server should belong to "CloudHosts" group
11. Host registered in Zabbix server should be linked with Custom template
12. Host registered in Zabbix server should have Name = Hostname (not IP)
13. This script should create group "CloudHosts" if it doesn't exist

For both VMs use vagrant box "sbeliakou/centos-7.3-x86_64-minimal"

```
#!/bin/bash
```

```
path=/home/vagrant
hostname=$(hostname)
```

```
#Generate an authentication token
```

```
curl -d '{"jsonrpc": "2.0","method": "user.login","params":
{"user": "Admin","password": "zabbix"},"id": 1}' -H "Content-Type: application/json-rpc" -X POST
http://192.168.33.13/api_jsonrpc.php > $path/token.tmp
token=$(sed -e 's/^.*"result": "[^"]*"'.*$/1/' $path/token.tmp)
```

```
#Create Host Group CloudHosts if it does not exist
```

```
curl -d '{"jsonrpc": "2.0","method": "hostgroup.get","params": {"output": "extend","filter":
{"name": "CloudHosts"}}, "auth": "'$token'", "id": 1}' -H "Content-Type: application/json-rpc" -X
POST http://192.168.33.13/api_jsonrpc.php > $path/group.tmp
```

```
if ! grep 'CloudHosts' $path/group.tmp;
then
```

```
    curl -d '{"jsonrpc": "2.0","method": "hostgroup.create","params": {"name": "CloudHosts"},"auth":
"'$token'", "id": 1}' -H "Content-Type: application/json-rpc" -X POST
    http://192.168.33.13/api_jsonrpc.php > $path/group.tmp
    group=$(sed -e 's/^.*"groupids": "[^"]*"'.*$/1/' $path/group.tmp)
fi
```

```
#Create a Custom template
```

```
curl -d '{"jsonrpc": "2.0","method": "template.create","params": {"host": "CustomTemplate","groups":
{"groupid": "'$group'" }}, "auth": "'$token'", "id": 1}' -H "Content-Type: application/json-rpc" -X POST
http://192.168.33.13/api_jsonrpc.php > $path/template.tmp
template=$(sed -e 's/^.*"templateids": "[^"]*"'.*$/1/' $path/template.tmp)
```

#Create a Host

```
curl -d '{"jsonrpc": "2.0", "method": "host.create", "params": {"host": "$hostname", "interfaces": [{"type": 1, "main": 1, "useip": 1, "ip": "192.168.33.14", "dns": "", "port": "10050"}], "groups": [{"groupid": "$group"}], "templates": [{"templateid": "$template"}]}, "auth": "$token", "id": 1}' -H "Content-Type: application/json-rpc" -X POST http://192.168.33.13/api_jsonrpc.php
```