

Zabbix与Elasticsearch集成

Andy zhou

自我介绍



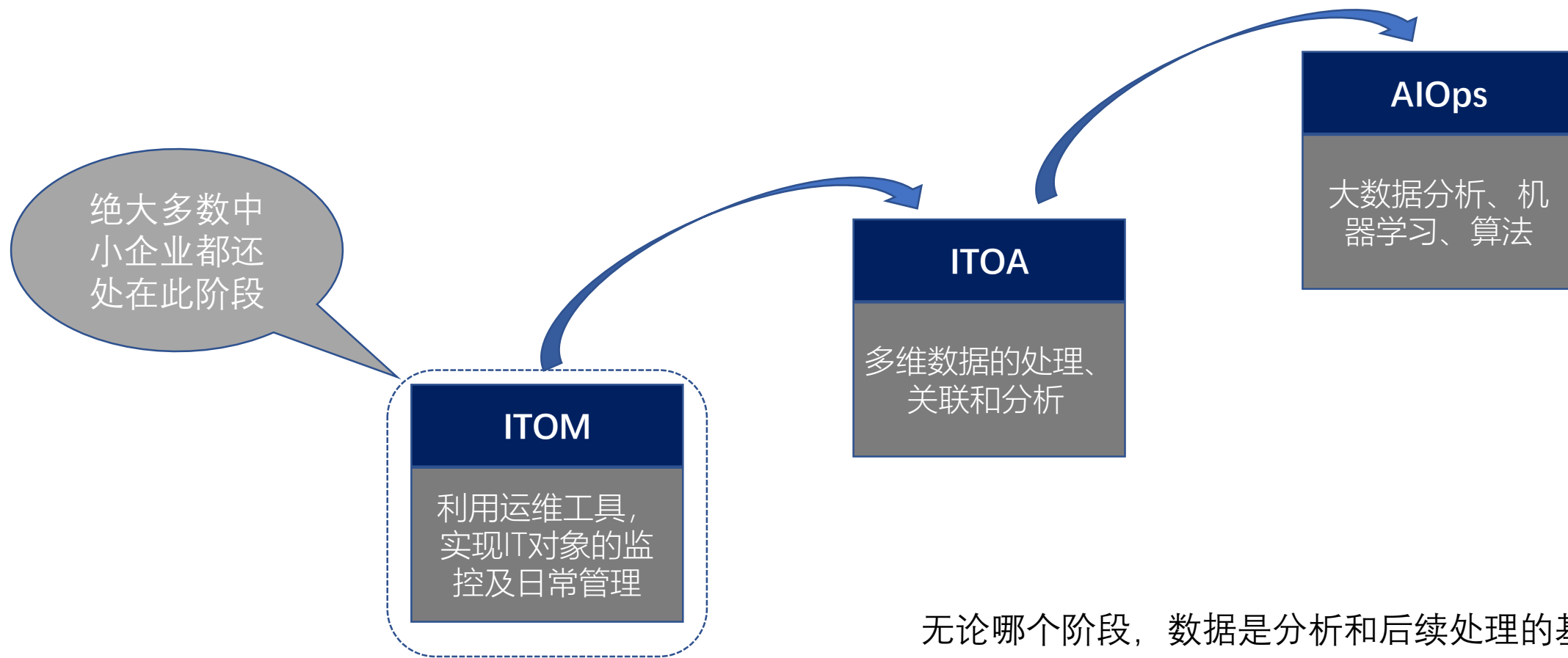
大中华区第一位Zabbix认证培训师、Zabbix架构师，近10年IT运维经验，5年Zabbix使用经验，长期从事在ITOM、ITOA领域，熟悉IBM Tivoli、BMC Patrol等软件产品，参与过国内多个大中型保险、金融公司IT运维项目的咨询、架构和管理，拥有非常丰富的IT运维管理实战经验。

近几年来，致力于开源软件领域的咨询和研究，长期活跃在国内外的各大开源社区。

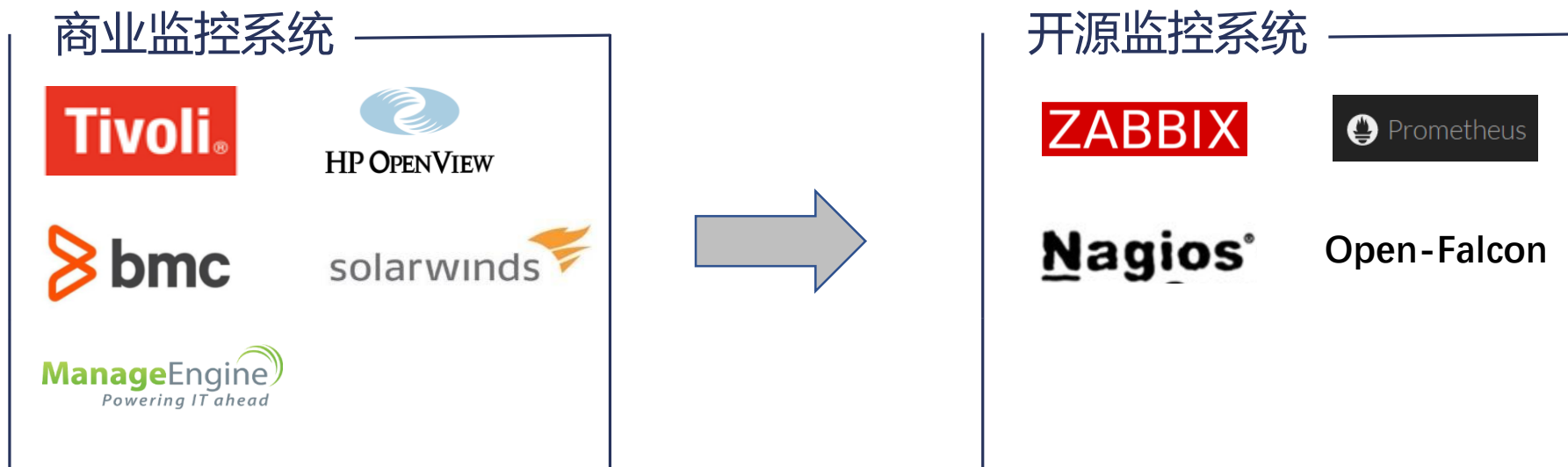
01

数据的采集

IT运维管理的发展阶段：



开源是未来的技术发展趋势和方向，越来越多客户选择用开源监控系统替换原有商业监控系统



Zabbix是众多开源监控系统中最受用户欢迎的一款!

02

Zabbix简介

Zabbix简介



ZABBIX

1

Zabbix是一个成熟的企业级、开源、分布式监控解决方案

2

Zabbix集系统监控和网络监控为一体，同时支持第三方集成

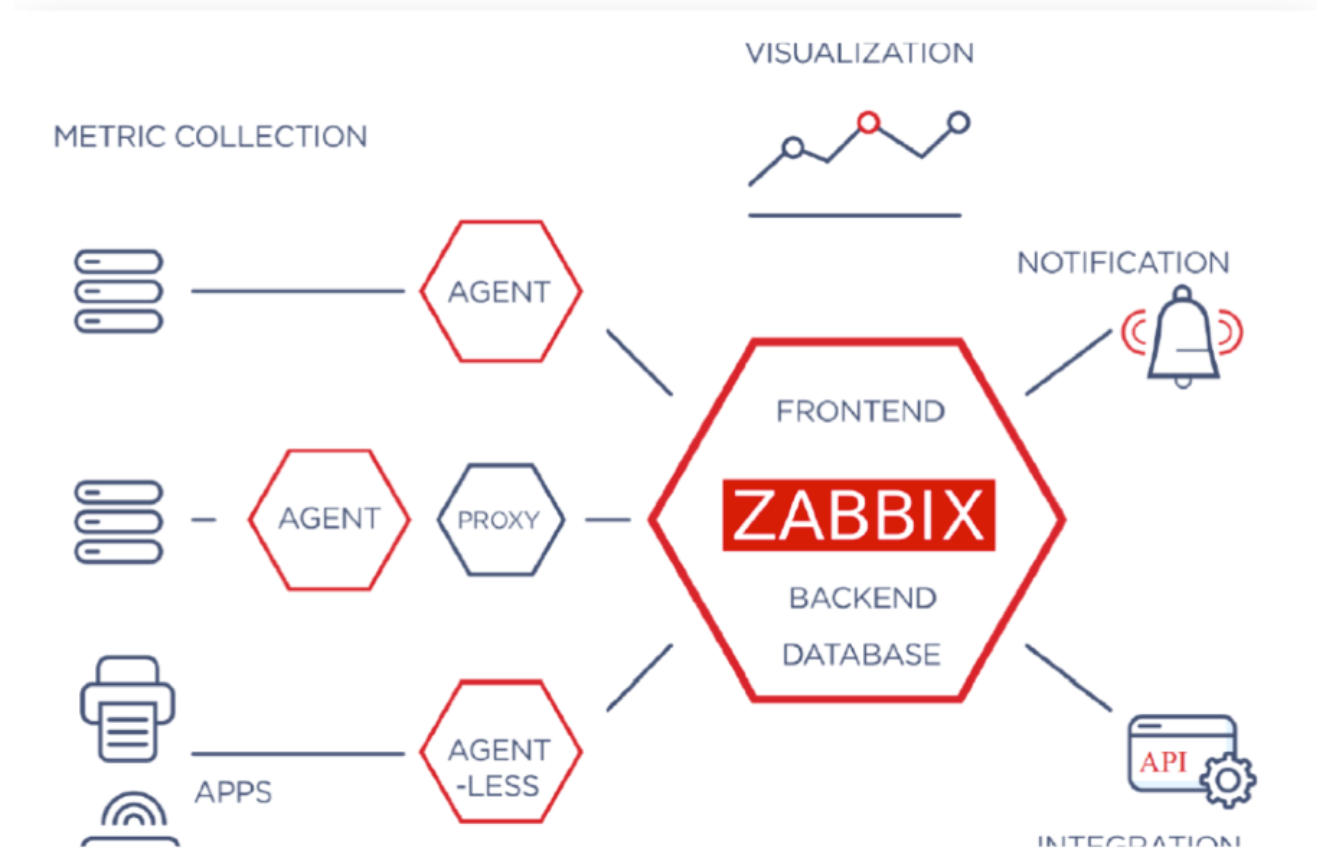
3

Zabbix提供丰富的监控方法和监控扩展方式

4

Zabbix使用灵活的告警机制，允许用户为几乎任何事件配置基于邮件、微信、短信等多样告警

Zabbix功能组件



■ Zabbix Server

负责接收agent或proxy发送数据的核心组件，所有配置、统计数据及数据处理都由它组织进行；

■ Database

专用于存储所有配置信息，以及收集到的性能数据；

■ Zabbix Proxy

适用于分布式环境，代理server收集部分数据转发到server，可以减轻server的压力；

■ Zabbix Agent

部署在被监控的主机上，负责收集主机性能数据，如CPU、内存、磁盘等，并将数据发往server端或proxy端。

Zabbix主要功能

主机性能监控

支持各类操作系统、数据库、中间件、应用URL、存储设备、虚拟化设备、日志等对象的性能监控

- 监控对象全面覆盖

- 丰富多样的数据采集方式

网络性能监控

支持各类交换机、路由器、防火墙、负载均衡设备等设备的性能监控

- 监控配置简单

- 可自动发现网络设备

灵活的告警

简单高效的trigger配置和告警发送机制

- 支持邮件、短信、微信、钉钉

- 告警实时准确

简单的资产管理

简单的设备资产信息配置管理，方便设备的管理和维护

- 操作方便

- 功能实用性高

- 自动采集丰富

性能报表展现

支持设备实时性能报表查询和定制，告警事件查看

- 实时生成性能报表

- 配置方便、灵活

强大的API接口

对外提供API接口，便于系统集成

- 利于其他系统的集成

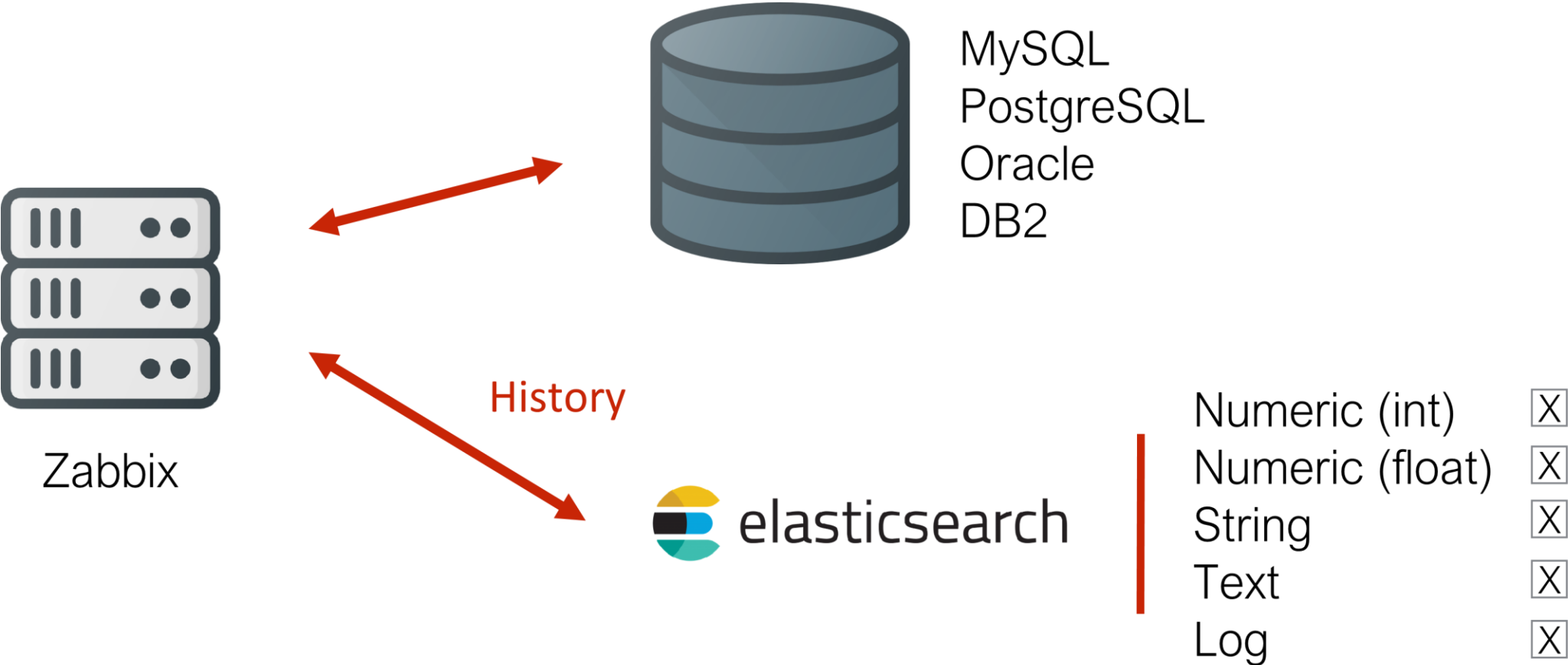
- 利于二次开发

- 方便自身管理

03

如何将Zabbix数据存入ES?

①History数据直接存入ES



①History数据直接存入ES

配置Elasticsearch 支持Zabbix的监控项类型: uint, dbl, str, log, text, 对应如下:

Zabbix监控项类型	Zabbix数据表	Elasticsearch类型
Numeric(unsigned)	history_uint	uint
Numeric(float)	history	dbl
Character	history_str	str
Log	history_log	log
Text	history_text	text

为每个类型添加Elasticsearch mapping:

```
curl -H "Content-Type:application/json" -XPUT http://127.0.0.1:9200/uint -d ' { "settings" :  
{ "index" : { "number_of_replicas" : 1, "number_of_shards" : 5 } }, "mappings" : { "values" :  
{ "properties" : { "itemid" : { "type" : "long" }, "clock" : { "format" : "epoch_second", "type" :  
"date" }, "value" : { "type" : "long" } } } } } '
```

①History数据直接存入ES

在Zabbix Server配置文件中配置历史数据保存入Elasticsearch中:

```
### Option: HistoryStorageURL
# History storage HTTP[S] URL.
#
# Mandatory: no
# Default:
HistoryStorageURL=127.0.0.1:9200

### Option: HistoryStorageTypes
# Comma separated list of value types to be sent to the history storage.
#
# Mandatory: no
# Default:
# HistoryStorageTypes=uint,dbl,str,log,text
HistoryStorageTypes=uint,dbl,str,log,text

### Option: HistoryStorageDateIndex
# Enable preprocessing of history values in history storage to store values in different indices based on date.
# 0 - disable
# 1 - enable
#
# Mandatory: no
# Default:
# HistoryStorageDateIndex=0
```

修改Zabbix前端配置文件，选择历史数据的类型:

```
<?php
// Zabbix GUI configuration file.
global $DB;

$DB['TYPE'] = 'MYSQL';
$DB['SERVER'] = 'localhost';
$DB['PORT'] = '0';
$DB['DATABASE'] = 'zabbix';
$DB['USER'] = 'zabbix';
$DB['PASSWORD'] = 'zabbix';


// Schema name. Used for IBM DB2 and PostgreSQL.
$DB['SCHEMA'] = '';

$ZBX_SERVER = '192.168.25.137';
$ZBX_SERVER_PORT = '10051';
$ZBX_SERVER_NAME = 'Zabbix 4.0.4';

$IMAGE_FORMAT_DEFAULT = IMAGE_FORMAT_PNG;
$HISTORY['url'] = 'http://127.0.0.1:9200';
$HISTORY['types'] = ['uint','dbl','str','log','text'];
~
~
```

重启Zabbix Server和http服务，生效配置。

①History数据直接存入ES

 kibana

Discover

Visualize

Dashboard

Timelion

Canvas

Maps

Machine Learning

Infrastructure

Logs

APM

Uptime

Dev Tools

Monitoring

Management

D

Default

370 hits

New Save Open Share Inspect Auto-refresh

>_ Search... (e.g. status:200 AND extension:PHP) Options Refresh

Add a filter +

str*

Selected fields

? _source

Available fields

t _id

t _index

_score

t _type

clock

itemid

ns

ttl

t value

Table

JSON

View single document

t _id

🔍

🔍

🔍

🔍

Nk_o8mkBRRoCwSŵwJH2R

t _index

🔍

🔍

🔍

🔍

str

_score

🔍

🔍

🔍

🔍

1

t _type

🔍

🔍

🔍

🔍

values

clock

🔍

🔍

🔍

🔍

1,554,558,519

itemid

🔍

🔍

🔍

🔍

29,359

ns

🔍

🔍

🔍

🔍

660,454,183

ttl

🔍

🔍

🔍

🔍

172,800

t value

🔍

🔍

🔍

🔍

zabbix_server (Zabbix) 4.0.4

itemid: 29,359 value: zabbix_server (Zabbix) 4.0.4 clock: 1,554,558,519 ns: 660,454,183 ttl: 172,800 _id: Nk_o8mkBRRoCwSŵwJH2R _type: values _index: str _score: 1

🔍

itemid: 29,359 value: zabbix_server (Zabbix) 4.0.4 clock: 1,554,558,769 ns: 183,832,952 ttl: 172,800 _id: dk_s8mkBRRoCwSŵwJ48 _type: values _index: str _score: 1

▶

itemid: 29,359 value: zabbix_server (Zabbix) 4.0.4 clock: 1,554,558,909 ns: 192,429,350 ttl: 172,800 _id: bk_u8mkBRRoCwSŵw7Ec _type: values _index: str _score: 1

▶

itemid: 29,359 value: zabbix_server (Zabbix) 4.0.4 clock: 1,554,558,929 ns: 177,222,841 ttl: 172,800 _id: Qk_u8mkBRRoCwSŵw7QN _type: values _index: str _score: 1

▶

itemid: 29,359 value: zabbix_server (Zabbix) 4.0.4 clock: 1,554,559,129 ns: 425,393,811 ttl: 172,800 _id: qk_x8mkBRRoCwSŵw2NVo _type: values _index: str _score: 1

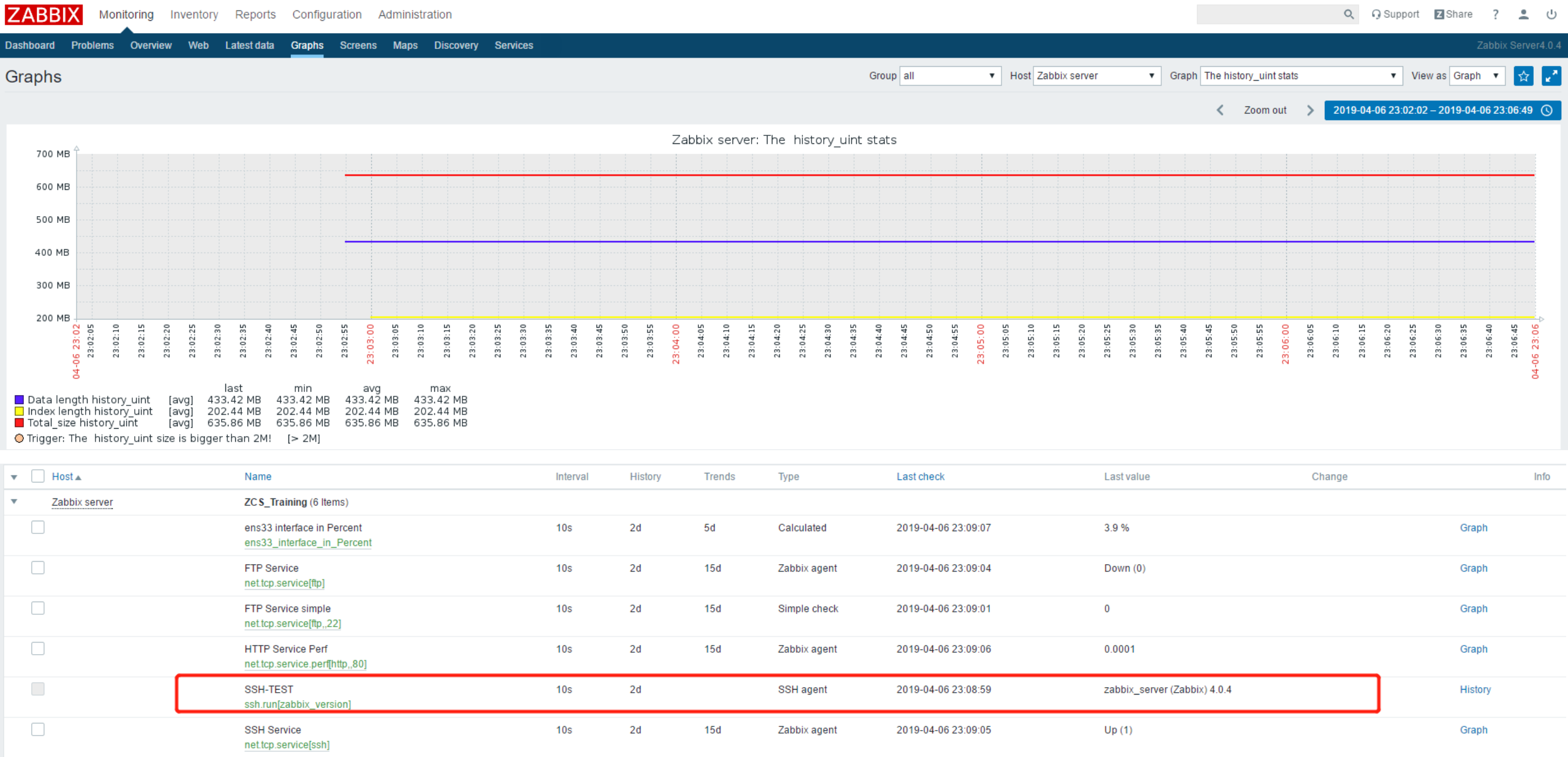
▶

itemid: 29,359 value: zabbix_server (Zabbix) 4.0.4 clock: 1,554,559,269 ns: 533,008,689 ttl: 172,800 _id: fU_z8mkBRRoCwSŵw_vAQ _type: values _index: str _score: 1

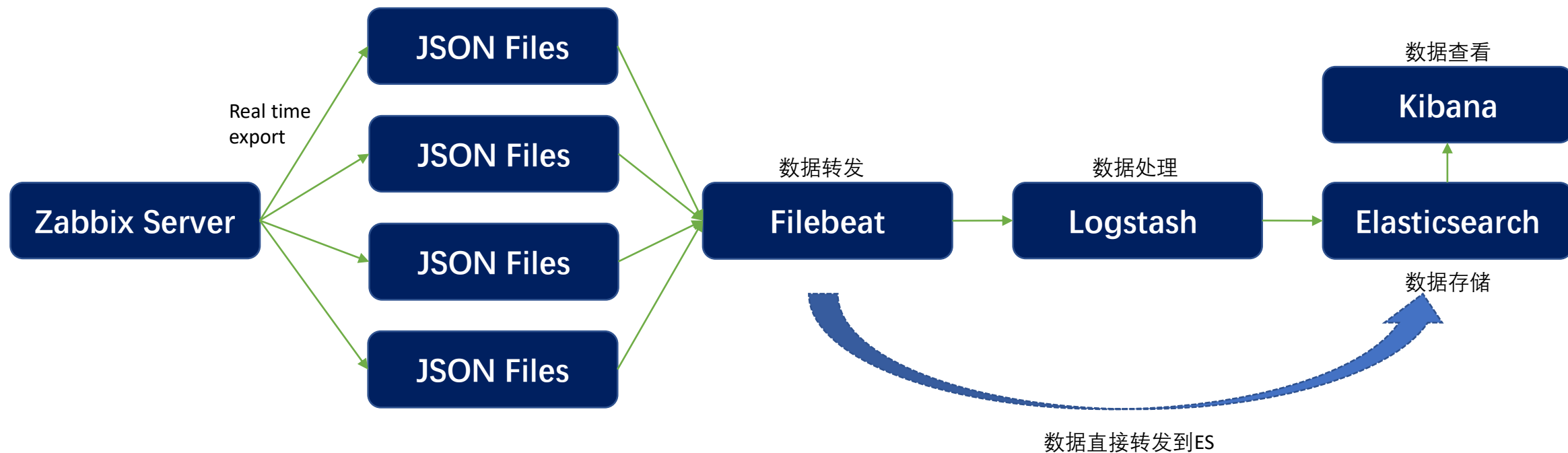
▶

itemid: 29,359 value: zabbix_server (Zabbix) 4.0.4 clock: 1,554,559,379 ns: 459,657,772 ttl: 172,800 _id: t1D18mkBRRoCwSŵwqAVY _type: values _index: str _score: 1

①History数据直接存入ES



②利用Zabbix Export功能将数据存入ES



②利用Zabbix Export功能将数据存入ES

在Zabbix Server配置文件中配置数据导出目录和数据文件大小：

Export导出的Json数据文件：

```
### Option: ExportDir
#   Directory for real time export of events, history and trends in newline delimited JSON format.
#   If set, enables real time export.
#
# Mandatory: no
# Default:
ExportDir=/opt/ZabbixData

### Option: ExportFileSize
#   Maximum size per export file in bytes.
#   Only used for rotation if ExportDir is set.
#
# Mandatory: no
# Range: 1M-1G
# Default:
ExportFileSize=1G
```

```
[root@TestServer ZabbixData]# pwd
/opt/ZabbixData
[root@TestServer ZabbixData]# ll
total 172752
-rw-rw-r-- 1 zabbix zabbix 42080023 Apr  4 00:22 history-history-syncer-1.ndjson
-rw-rw-r-- 1 zabbix zabbix 43266399 Apr  4 00:22 history-history-syncer-2.ndjson
-rw-rw-r-- 1 zabbix zabbix 42996187 Apr  4 00:22 history-history-syncer-3.ndjson
-rw-rw-r-- 1 zabbix zabbix 43746820 Apr  4 00:22 history-history-syncer-4.ndjson
-rw-rw-r-- 1 zabbix zabbix      346 Jan 22 22:08 history-main-process-0.ndjson
-rw-rw-r-- 1 zabbix zabbix      8148 Apr  4 00:18 problems-history-syncer-1.ndjson
-rw-rw-r-- 1 zabbix zabbix      8810 Apr  4 00:18 problems-history-syncer-2.ndjson
-rw-rw-r-- 1 zabbix zabbix      7851 Apr  4 00:18 problems-history-syncer-3.ndjson
-rw-rw-r-- 1 zabbix zabbix      8417 Apr  4 00:18 problems-history-syncer-4.ndjson
-rw-rw-r-- 1 zabbix zabbix         0 Nov  7 22:06 problems-main-process-0.ndjson
-rw-rw-r-- 1 zabbix zabbix         0 Nov  7 22:06 problems-task-manager-1.ndjson
-rw-rw-r-- 1 zabbix zabbix 670633 Mar 30 07:28 trends-history-syncer-1.ndjson
-rw-rw-r-- 1 zabbix zabbix 796311 Mar 30 07:28 trends-history-syncer-2.ndjson
-rw-rw-r-- 1 zabbix zabbix 778804 Mar 31 23:12 trends-history-syncer-3.ndjson
-rw-rw-r-- 1 zabbix zabbix 676889 Mar 31 23:13 trends-history-syncer-4.ndjson
-rw-rw-r-- 1 zabbix zabbix      23056 Jan 22 22:08 trends-main-process-0.ndjson
[root@TestServer ZabbixData]#
```

数据实时的从Zabbix Server中导出至Json文件中。

②利用Zabbix Export功能将数据存入ES

Zabbix Export导出的性能数据：

```
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "Memory", "itemid": 23309, "name": "Free swap space", "clock": 1541599229, "ns": 50140237, "value": 2147479552 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "CPU", "Performance", "itemid": 23297, "name": "Processor load (5 min average per core)", "clock": 1541599397, "ns": 280922165, "value": 0.010000 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "CPU", "Performance", "itemid": 23299, "name": "CPU idle time", "clock": 1541599399, "ns": 281103743, "value": 99.113861 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "CPU", "Performance", "itemid": 23301, "name": "CPU iowait time", "clock": 1541599401, "ns": 283790509, "value": 0.008360 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "CPU", "Performance", "itemid": 23305, "name": "CPU system time", "clock": 1541599405, "ns": 291496908, "value": 0.430602 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "Memory", "itemid": 23310, "name": "Free swap space in %", "clock": 1541599410, "ns": 297102509, "value": 100.000000 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "Zabbix server", "itemid": 23256, "name": "Utilization of escalator internal processes, in %", "clock": 1541599416, "ns": 300921381, "value": 0.101540 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "Memory", "itemid": 23316, "name": "Available memory", "clock": 1541599416, "ns": 302014988, "value": 877416448 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "Zabbix server", "itemid": 23259, "name": "Utilization of http poller data collector processes, in %", "clock": 1541599419, "ns": 303561244, "value": 0.050796 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "Zabbix server", "itemid": 23266, "name": "Utilization of self-monitoring internal processes, in %", "clock": 1541599426, "ns": 309264397, "value": 0.000000 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "Zabbix server", "itemid": 23270, "name": "Utilization of unreachable poller data collector processes, in %", "clock": 1541599430, "ns": 315271842, "value": 0.000000 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "Zabbix server", "itemid": 10075, "name": "Number of processed log values per second", "clock": 1541599435, "ns": 319998655, "value": 0.000000 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "Zabbix server", "itemid": 23275, "name": "Zabbix history index cache, % used", "clock": 1541599435, "ns": 320021874, "value": 0.220510 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "Network interfaces", "itemid": 28499, "name": "Incoming network traffic on ens33", "clock": 1541599439, "ns": 325595274, "value": 1392 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "Filesystems", "itemid": 28504, "name": "Free inodes on /boot (percentage)", "clock": 1541599444, "ns": 331998059, "value": 99.937248 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "Filesystems", "itemid": 28505, "name": "Free disk space on /", "clock": 1541599445, "ns": 330280903, "value": 43006644224 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "Filesystems", "itemid": 28508, "name": "Free disk space on /boot (percentage)", "clock": 1541599448, "ns": 338483565, "value": 84.736347 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "CPU", "Performance", "itemid": 23295, "name": "Processor load (15 min average per core)", "clock": 1541599455, "ns": 346250904, "value": 0.012500 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "CPU", "Performance", "itemid": 23299, "name": "CPU idle time", "clock": 1541599459, "ns": 352532482, "value": 99.297776 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "CPU", "Performance", "itemid": 23301, "name": "CPU iowait time", "clock": 1541599461, "ns": 355744651, "value": 0.008360 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "CPU", "Performance", "itemid": 23305, "name": "CPU system time", "clock": 1541599465, "ns": 361150623, "value": 0.384519 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "Zabbix server", "itemid": 23252, "name": "Utilization of alerter internal processes, in %", "clock": 1541599472, "ns": 364947771, "value": 0.000000 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "Zabbix server", "itemid": 23256, "name": "Utilization of escalator internal processes, in %", "clock": 1541599476, "ns": 369111619, "value": 0.135387 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "Memory", "itemid": 23316, "name": "Available memory", "clock": 1541599476, "ns": 370533675, "value": 877289472 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "Zabbix server", "itemid": 23266, "name": "Utilization of self-monitoring internal processes, in %", "clock": 1541599486, "ns": 378545429, "value": 0.000000 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "Zabbix server", "itemid": 23270, "name": "Utilization of unreachable poller data collector processes, in %", "clock": 1541599490, "ns": 382364683, "value": 0.000000 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "Zabbix server", "itemid": 10076, "name": "Number of processed numeric (unsigned) values per second", "clock": 1541599496, "ns": 388683841, "value": 0.349602 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "Zabbix server", "itemid": 23276, "name": "Zabbix trend write cache, % used", "clock": 1541599496, "ns": 388733137, "value": 0.173378 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "Network interfaces", "itemid": 28499, "name": "Incoming network traffic on ens33", "clock": 1541599499, "ns": 393074257, "value": 1400 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "Network interfaces", "itemid": 28501, "name": "Outgoing network traffic on virbr0", "clock": 1541599501, "ns": 393984158, "value": 0 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "Processes", "itemid": 23291, "name": "Number of running processes", "clock": 1541599511, "ns": 435197582, "value": 1 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "Zabbix server", "itemid": 23258, "name": "Utilization of housekeeper internal processes, in %", "clock": 1541599838, "ns": 876467019, "value": 0.000000 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "Zabbix server", "itemid": 23264, "name": "Utilization of poller data collector processes, in %", "clock": 1541599844, "ns": 880384685, "value": 0.037248 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "Zabbix server", "itemid": 25667, "name": "Utilization of preprocessing manager internal processes, in %", "clock": 1541599847, "ns": 881878421, "value": 0.000000 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "Zabbix server", "itemid": 10074, "name": "Number of processed character values per second", "clock": 1541599854, "ns": 897310591, "value": 0.000000 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "Zabbix server", "itemid": 23274, "name": "Zabbix history write cache, % used", "clock": 1541599854, "ns": 897379802, "value": 0.000000 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "Zabbix server", "itemid": 10077, "name": "Number of processed text values per second", "clock": 1541599857, "ns": 899298259, "value": 0.000000 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "Zabbix server", "itemid": 23277, "name": "Number of processed values per second", "clock": 1541599857, "ns": 899627796, "value": 1.148672 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "Network interfaces", "itemid": 28497, "name": "Incoming network traffic on virbr0-nic", "clock": 1541599857, "ns": 899897794, "value": 0 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "Network interfaces", "itemid": 28502, "name": "Outgoing network traffic on ens33", "clock": 1541599862, "ns": 904166967, "value": 4272 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "Filesystems", "itemid": 28506, "name": "Free disk space on /boot", "clock": 1541599866, "ns": 909942451, "value": 900964352 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "Filesystems", "itemid": 28511, "name": "Used disk space on /", "clock": 1541599871, "ns": 931188850, "value": 7425228800 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "Processes", "itemid": 23291, "name": "Number of running processes", "clock": 1541599871, "ns": 934318625, "value": 3 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "CPU", "Performance", "itemid": 23294, "name": "Interrupts per second", "clock": 1541599874, "ns": 925025248, "value": 268 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "CPU", "Performance", "itemid": 23297, "name": "Processor load (5 min average per core)", "clock": 1541599877, "ns": 930520857, "value": 0.002500 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "CPU", "Performance", "itemid": 23299, "name": "CPU idle time", "clock": 1541599879, "ns": 932796622, "value": 99.268456 }
{ "host": "Zabbix server", "groups": [ "Zabbix servers" ], "applications": [ "CPU", "Performance", "itemid": 23301, "name": "CPU iowait time", "clock": 1541599881, "ns": 938431443, "value": 0.008360 }
```

②利用Zabbix Export功能将数据存入ES

进入Filebeat配置目录/etc/filebeat, 修改filebeat.yml文件:

```
##### Filebeat Configuration Example #####

# This file is an example configuration file highlighting only the most common
# options. The filebeat.reference.yml file from the same directory contains all the
# supported options with more comments. You can use it as a reference.
#
# You can find the full configuration reference here:
# https://www.elastic.co/guide/en/beats/filebeat/index.html
#
# For more available modules and options, please see the filebeat.reference.yml sample
# configuration file.

#===== Filebeat inputs =====

filebeat.inputs:

# Each - is an input. Most options can be set at the input level, so
# you can use different inputs for various configurations.
# Below are the input specific configurations.

- type: log

# Change to true to enable this input configuration.
enabled: true
# enabled: false

# Paths that should be crawled and fetched. Glob based paths.
paths:
# - /var/log/*.log
- /opt/ZabbixExport/history-history-syner-*.ndjson
# - c:\programdata\elasticsearch\logs\*

# Exclude lines. A list of regular expressions to match. It drops the lines that are
# matching any regular expression from the list.
#exclude_lines: ['^DBG']

# Include lines. A list of regular expressions to match. It exports the lines that are
# matching any regular expression from the list.
#include_lines: ['^ERR', '^WARN']

"filebeat.yml" 206L, 7685C
```

②利用Zabbix Export功能将数据存入ES

进入Logstash配置目录/etc/logstash/conf.d，添加01-logstash-initial.conf文件，添加对Json格式数据的处理：

```
[root@testServer conf.d]# cat 01-logstash-initial.conf
input {
  beats {
    port => 5044
    type => "logs"
    codec => json
    add_field => {"DataSource" => "Zabbix"}
    add_field => {"DataFormatVerion" => 1}
    add_hostname => false
  }
}
filter{
  json {
    source => "message"
    target => "json"
  }
  mutate {
    copy => {"json[clock]" => "clock"}
    copy => {"json[host]" => "org_host"}
    copy => {"json[groups]" => "groups"}
    copy => {"json[applications]" => "applications"}
    copy => {"json[itemid]" => "itemid"}
    copy => {"json[name]" => "name"}
    copy => {"json[clock]" => "clock"}
    copy => {"json[ns]" => "ns"}
    copy => {"json[value]" => "value"}
    remove_field => ["json", "message"]
  }
  date{
    match => ["clock","yyyy-MM-dd HH:mm:ss","UNIX"]
    target=> "date_clock"
  }
  grok{
    match => {
      "date_clock" => "%{YEAR:clock_year}-%{MONTHNUM:clock_month}-%{MONTHDAY:clock_day}[T ]%{HOUR:clock_hour}:?%{MINUTE:clock_minute}(?:?%{SECOND:clock_second})?%{ISO8601_TIMEZONE}?"
    }
    add_field => {"clock_date" => "%{clock_year}-%{clock_month}-%{clock_day} %{clock_hour}:%{clock_minute}:%{clock_second}"}
  }
  mutate{
    split => ["clock_date","."]
    copy => { "clock_date[0]" => "clock_date2" }
  }
  mutate{
    convert => ["DataFormatVerion","integer"]
    split => ["applications", ","]
    # add_field => {"applications2" => "applications[0]"}
    copy => { "applications[0]" => "applications2" }
  }
  mutate {
    copy => { "DataSource" => "dest_field1" }
    copy => { "org_host" => "dest_field5" }
    copy => { "DataFormatVerion" => "dest_field3" }
    copy => { "applications2" => "dest_field4" }
    copy => { "name" => "dest_field8" }
    # copy => { "clock_date" => "dest_field9" }
    copy => { "clock_date2" => "dest_field9" }
    copy => { "value" => "dest_field10" }
    join => { "applications" => "," }
    add_field => {
      "message" => "%{dest_field1},{%{dest_field5},{%{dest_field3},{%{dest_field4},{%{dest_field5},{%{dest_field5},{%{dest_field4},{%{dest_field8},{%{dest_field9},{%{dest_field10}"
    }
    # "message" => "%{dest_field1},{%{dest_field3},{%{dest_field3},{%{dest_field4},{%{dest_field3},{%{dest_field3},{%{dest_field4},{%{dest_field8},{%{dest_field9},{%{dest_field10}"
  }
}
}
output {
  elasticsearch {
    hosts => ["192.168.25.128:9200"]
    codec => rubydebug
  }
}
```


②利用Zabbix Export功能将数据存入ES

Discover

Visualize

Dashboard

Timelion

APM

Dev Tools

Monitoring

Management

5,060 hits

zabbix

Add a filter +

logstash-*

Selected fields

? _source

Available fields

@timestamp

@version

_id

_index

_score

_type

beat.hostname

beat.name

beat.version

host.architecture

host.containerized

host.id

host.name

host.os.codename

host.os.family

host.os.platform

host.os.version

input.type

message

offset

Collapse

New Save Open Share Reporting Auto-refresh

Options

_source

message: {"host": "Zabbix server", "groups": ["Zabbix servers"], "applications": ["Zabbix server"], "itemid": 23275, "name": "Zabbix history index cache, % used", "clock": 1544621695, "ns": 375492575, "value": 0.220510} beat.version: 6.5.2 beat.hostname: TestServer beat.name: TestServer source: /opt/ZabbixExport/history-history-syner-2.ndjson input.type: log offset: 312,122 prospector.type: log type: logs @timestamp: December 12th 2018, 21:40:31.997 tags: beats_input_codec_plain_applied @version: 1 host.containerized: true host.os.version: 7 (Core) host.os.platform: centos host.os.codename: Core host.os.family: redhat host.name: TestServer host.id: af04474238d54af6b5eela5b44351705 host.architecture: x86_64 _id: sGilmcBIFfAs42U_hyP _type: doc _index: logstash-2018.12.12 _score: 0.002

message: {"host": "Zabbix server", "groups": ["Zabbix servers"], "applications": ["Zabbix server"], "itemid": 28249, "name": "Zabbix preprocessing queue", "clock": 1544620849, "ns": 476314722, "value": 0} beat.version: 6.5.2 beat.hostname: TestServer beat.name: TestServer source: /opt/ZabbixExport/history-history-syner-3.ndjson offset: 141,122 input.type: log prospector.type: log type: logs @timestamp: December 12th 2018, 21:40:32.000 tags: beats_input_codec_plain_applied @version: 1 host.containerized: true host.os.version: 7 (Core) host.os.platform: centos host.os.family: redhat host.os.codename: Core host.name: TestServer host.id: af04474238d54af6b5eela5b44351705 host.architecture: x86_64 _id: 42ilomcBIFfAs42U_hyP _type: doc _index: logstash-2018.12.12 _score: 0.002

message: {"host": "Zabbix server", "groups": ["Zabbix servers"], "applications": ["Zabbix server"], "itemid": 23273, "name": "Zabbix configuration cache, % used", "clock": 1544620913, "ns": 531999782, "value": 3.435230} beat.version: 6.5.2 beat.hostname: TestServer beat.name: TestServer source: /opt/ZabbixExport/history-history-syner-3.ndjson offset: 148,588 input.type: log prospector.type: log type: logs @timestamp: December 12th 2018, 21:40:32.003 tags: beats_input_codec_plain_applied @version: 1 host.containerized: true host.os.version: 7 (Core) host.os.platform: centos host.os.family: redhat host.os.codename: Core host.name: TestServer host.id: af04474238d54af6b5eela5b44351705 host.architecture: x86_64 _id: C2ilomcBIFfAs42U_h2Q _type: doc _index: logstash-2018.12.12 _score: 0.002

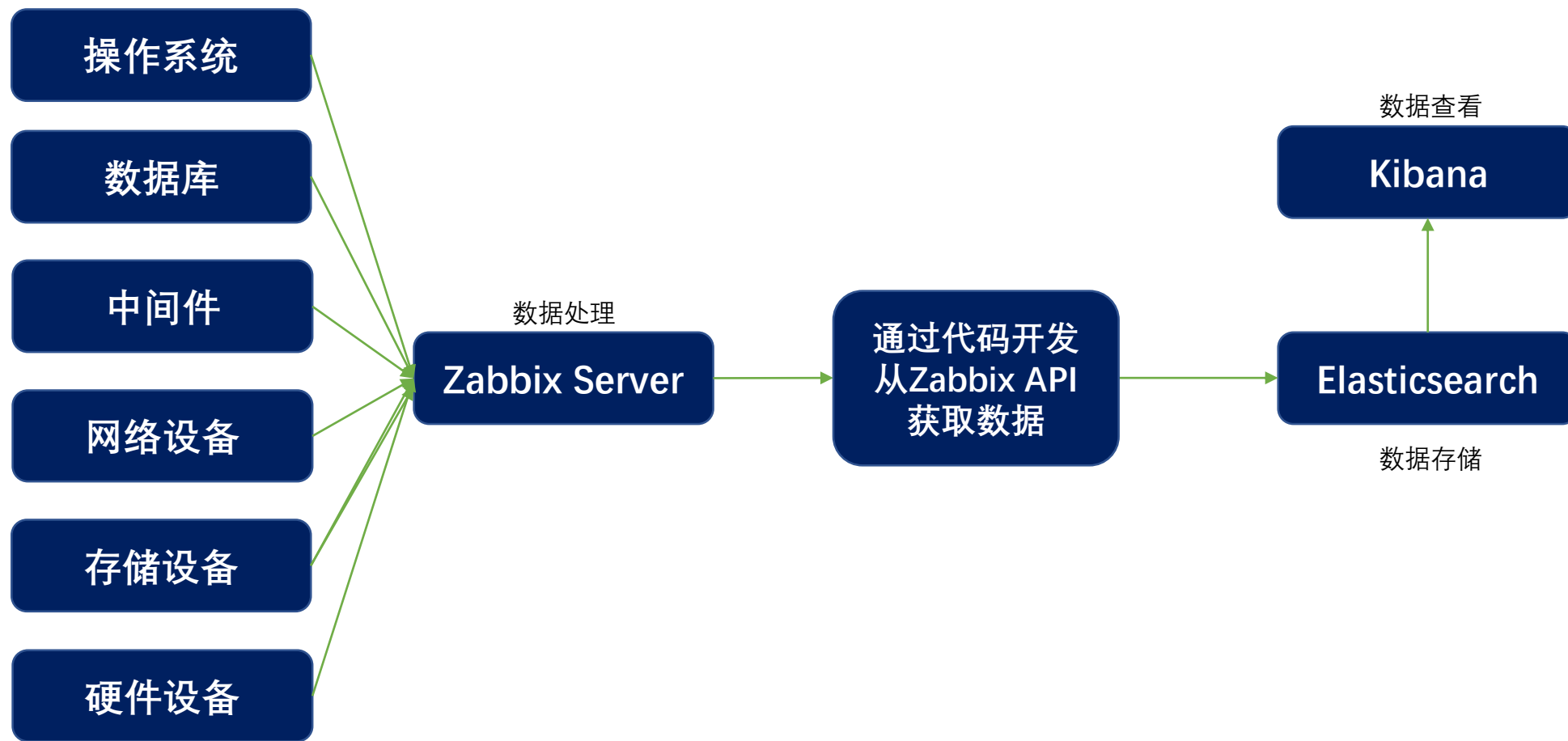
message: {"host": "Zabbix server", "groups": ["Zabbix servers"], "applications": ["Zabbix server"], "itemid": 23274, "name": "Zabbix history write cache, % used", "clock": 1544620914, "ns": 533129662, "value": 0.000000} beat.version: 6.5.2 beat.hostname: TestServer beat.name: TestServer source: /opt/ZabbixExport/history-history-syner-3.ndjson offset: 148,784 input.type: log prospector.type: log type: logs @timestamp: December 12th 2018, 21:40:32.003 tags: beats_input_codec_plain_applied @version: 1 host.os.version: 7 (Core) host.os.platform: centos host.os.family: redhat host.os.codename: Core host.containerized: true host.name: TestServer host.id: af04474238d54af6b5eela5b44351705 host.architecture: x86_64 _id: DGilomcBIFfAs42U_h2Q _type: doc _index: logstash-2018.12.12 _score: 0.002

message: {"host": "Zabbix server", "groups": ["Zabbix servers"], "applications": ["Zabbix server"], "itemid": 23628, "name": "Zabbix value cache misses", "clock": 1544621748, "ns": 439911663, "value": 0.000000} beat.version: 6.5.2 beat.hostname: TestServer beat.name: TestServer source: /opt/ZabbixExport/history-history-syner-2.ndjson input.type: log offset: 315,309 prospector.type: log type: logs @timestamp: December 12th 2018, 21:40:32.007 tags: beats_input_codec_plain_applied @version: 1 host.os.version: 7 (Core) host.os.platform: centos host.os.family: redhat host.os.codename: Core host.containerized: true host.name: TestServer host.id: af04474238d54af6b5eela5b44351705 host.architecture: x86_64 _id: L2imomcBIFfAs42UAB2q _type: doc _index: logstash-2018.12.12 _score: 0.002

message: {"host": "Zabbix server", "groups": ["Zabbix servers"], "applications": ["Zabbix server"], "itemid": 23625, "name": "Zabbix value cache hits", "clock": 1544621805, "ns": 507503922, "value": 4.860935} beat.version: 6.5.2 beat.hostname: TestServer beat.name: TestServer source: /opt/ZabbixExport/history-history-syner-2.ndjson input.type: log offset: 318,465 prospector.type: log type: logs @timestamp: December 12th 2018, 21:40:32.010 tags: beats_input_codec_plain_applied @version: 1 host.containerized: true host.os.version: 7 (Core) host.os.family: redhat host.os.

③利用Zabbix API将数据存入ES

Zabbix拥有完善的API接口，基于JSON RPC协议，提供对主机，主机组，监控项，资产，告警，性能等数据的获取和更新，外部程序可以利用Zabbix API接口的功能实现数据的获取。



③利用Zabbix API将数据存入ES

https://www.zabbix.com/documentation/4.0/manual/api/reference

ZABBIX

Zabbix Documentation 4.0

2.2 3.0 4.0 4.2 (current) In development: 4.4 (devel) Unsupported: 1.8 2.0 2.4 3.2 3.4

Log In

Search

Recent Changes Sitemap

Zabbix documentation

Zabbix Manual

1. Introduction

2. Definitions

3. Zabbix processes

4. Installation

5. Quickstart

6. Zabbix appliance

7. Configuration

8. Service monitoring

9. Web monitoring

10. Virtual machine monitoring

11. Maintenance

12. Regular expressions

13. Problem acknowledgement

14. Configuration export/import

15. Discovery

16. Distributed monitoring

17. Encryption

18. Web interface

19. API

Method reference

Action

Alert

API info

Application

Configuration

Correlation

Dashboard

Discovered host

Discovered service

Discovery check

Discovery rule

Event

Graph

Graph item

Graph prototype

History

Host

Host object

host.create

host.delete

host.get

host.get

Description

integer/array host.get(object parameters)

The method allows to retrieve hosts according to the given parameters.

Parameters

(object) Parameters defining the desired output.

The method supports the following parameters.

Parameter	Type	Description
groupids	string/array	Return only hosts that belong to the given groups.
applicationids	string/array	Return only hosts that have the given applications.
dserviceids	string/array	Return only hosts that are related to the given discovered services.
graphids	string/array	Return only hosts that have the given graphs.
hostids	string/array	Return only hosts with the given host IDs.
httptestids	string/array	Return only hosts that have the given web checks.
interfaceids	string/array	Return only hosts that use the given interfaces.
itemids	string/array	Return only hosts that have the given items.
maintenanceids	string/array	Return only hosts that are affected by the given maintenances.
monitored_hosts	flag	Return only monitored hosts.
proxy_hosts	flag	Return only proxies.
proxyids	string/array	Return only hosts that are monitored by the given proxies.
templated_hosts	flag	Return both hosts and templates.
templateids	string/array	Return only hosts that are linked to the given templates.
triggerids	string/array	Return only hosts that have the given triggers.
with_items	flag	Return only hosts that have items.

Table of Contents

host.get

Description

Parameters

Return values

Examples

Retrieving data by name

Retrieving host groups

Retrieving linked templates

Searching by host inventory data

See also

Source

③利用Zabbix API将数据存入ES

```
#!/usr/bin/python
# -*- coding: utf-8 -*-
# python2.6/2.7
# Date: 2019/4/3          Version: 1.0

# grandage_zabbix.py is as the same level as this script.
from grandage_zabbix import ZabbixApi

def getHostname(instance, session):
    hostname_dict = instance.getMethodJson('host.get', session)
    hostname_dict['params'] = { 'output':['host','name']}
    return instance.getZBXResponse(**hostname_dict)

def main():
    zabbix_url = 'http://192.168.25.137/zabbix/api_jsonrpc.php'
    api_instance = ZabbixApi(zabbix_url)
    session = api_instance.login()
    hostname_info_dict = getHostname(api_instance, session)
    for hostname_info in hostname_info_dict:
        print("Visible name:%s, Hostname:%s, Hostid:%s"%(hostname_info['name'], hostname_info['host'], hostname_info['hostid']))
    api_instance.logout(session)

if __name__ == '__main__':
    main()
```



```
[root@TestServer opt]# ./getHostname.py
Visible name:Zabbix server, Hostname:Zabbix server, Hostid:10084
Visible name:Mysql_server, Hostname:Mysql_server, Hostid:10267
Visible name:Tomcat, Hostname:Tomcat, Hostid:10272
Visible name:Mysql_server_clone, Hostname:Mysql_server_clone, Hostid:10279
Visible name:Mysql_server_fullclone, Hostname:Mysql_server_fullclone, Hostid:10280
Visible name:Windows10_server, Hostname:Windows10_server, Hostid:10281
Visible name:Test Server1, Hostname:Test Server1, Hostid:10271
Visible name:Test Server3, Hostname:Test Server3, Hostid:10283
Visible name:Test Server2, Hostname:Test Server2, Hostid:10285
Visible name:Test Server4, Hostname:Test Server4, Hostid:10286
Visible name:Test Server5, Hostname:Test Server5, Hostid:10287
[root@TestServer opt]#
```


04

联系我们

联系我们



138-1772-0274
021-697806188



china@zabbix.com
market@grandage.cn



上海市徐汇区虹梅路1905号



www.zabbix.com/cn
www.grandage.cn



@Zabbix team



<http://i.youku.com/zabbixchina>



微信公众号: Zabbix开源社区
(微信群: Zabbix_China)



微博@Zabbix_China

加入我们

扫码入群



关注公众号



关注微博



Thank you!



专业、垂直、纯粹的 Elastic 开源技术交流社区
<https://elasticsearch.cn/>