

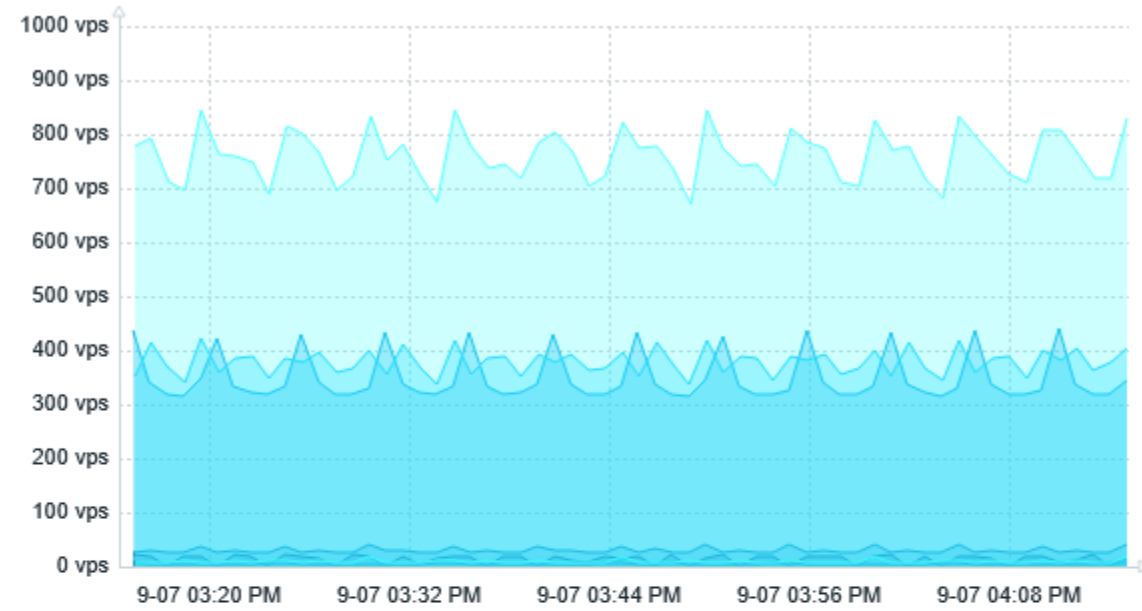
The strange world of Zabbix Diagnostics



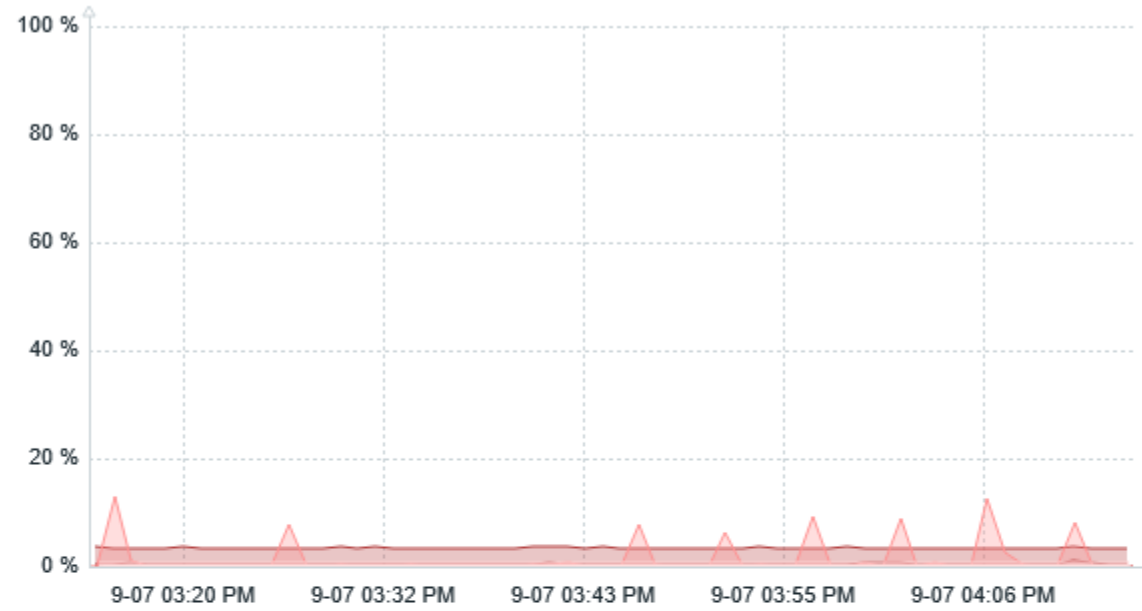
Troubleshooting Zabbix



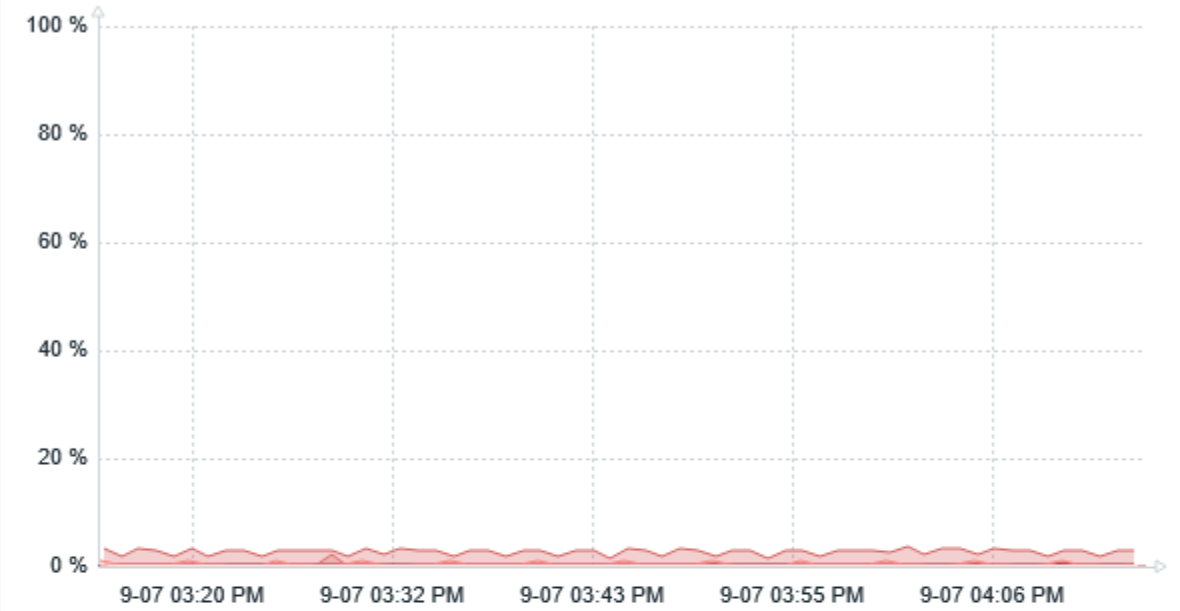
Values processed per second



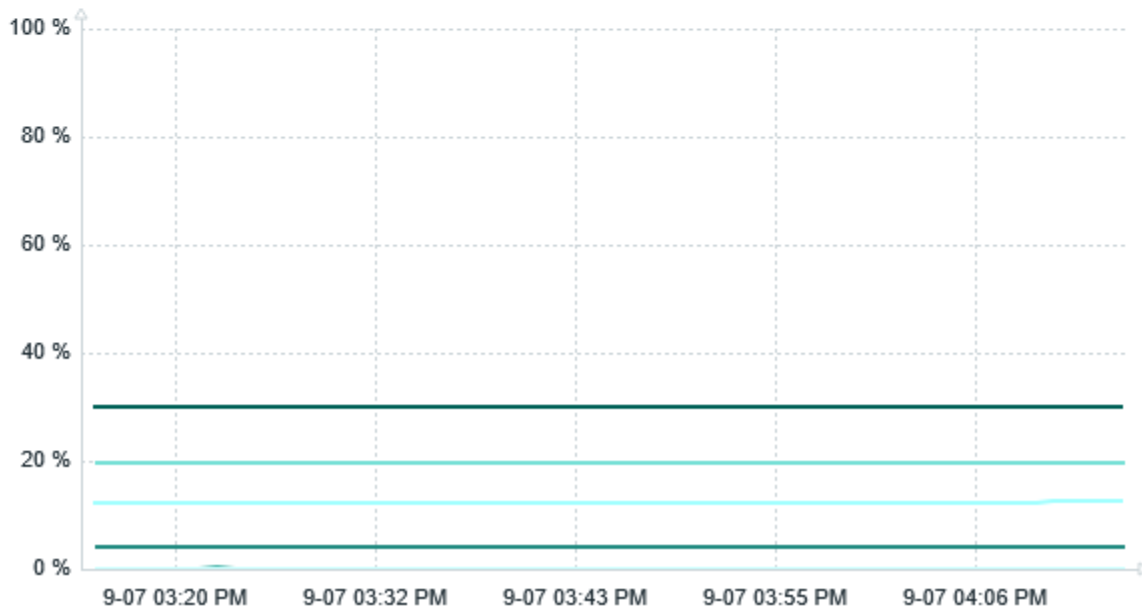
Utilization of data collectors



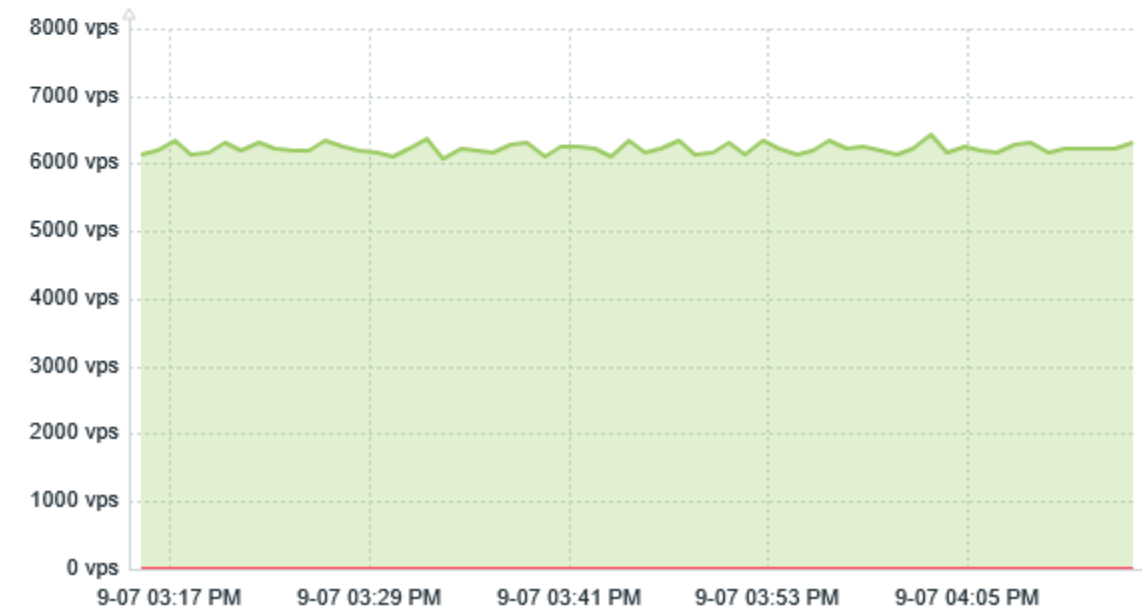
Utilization of internal processes



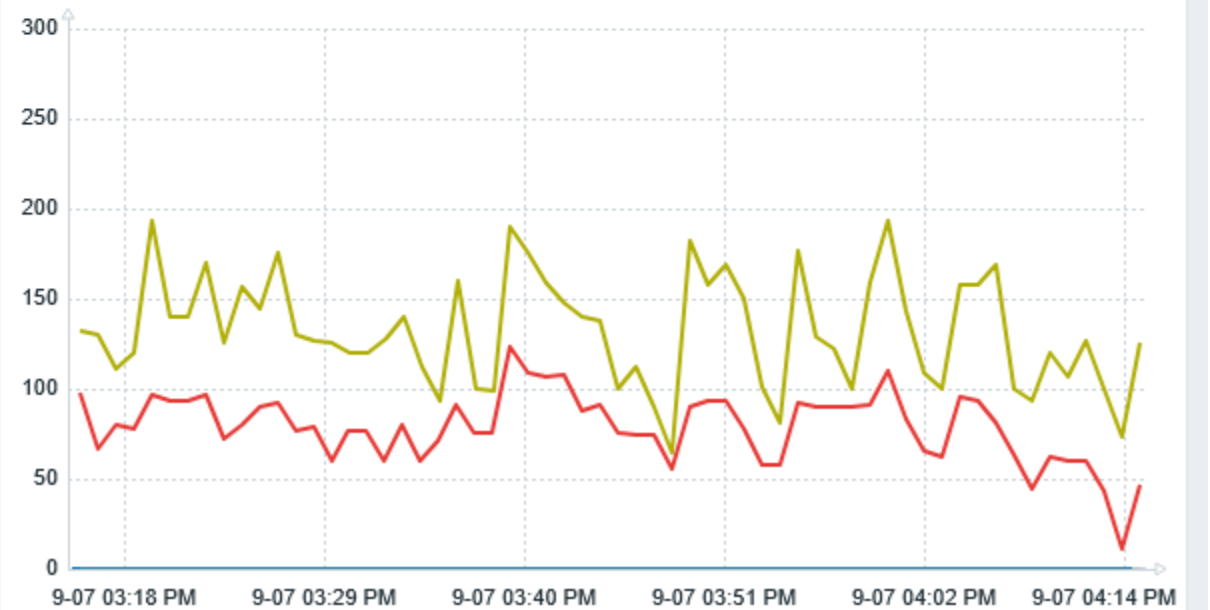
Cache usage



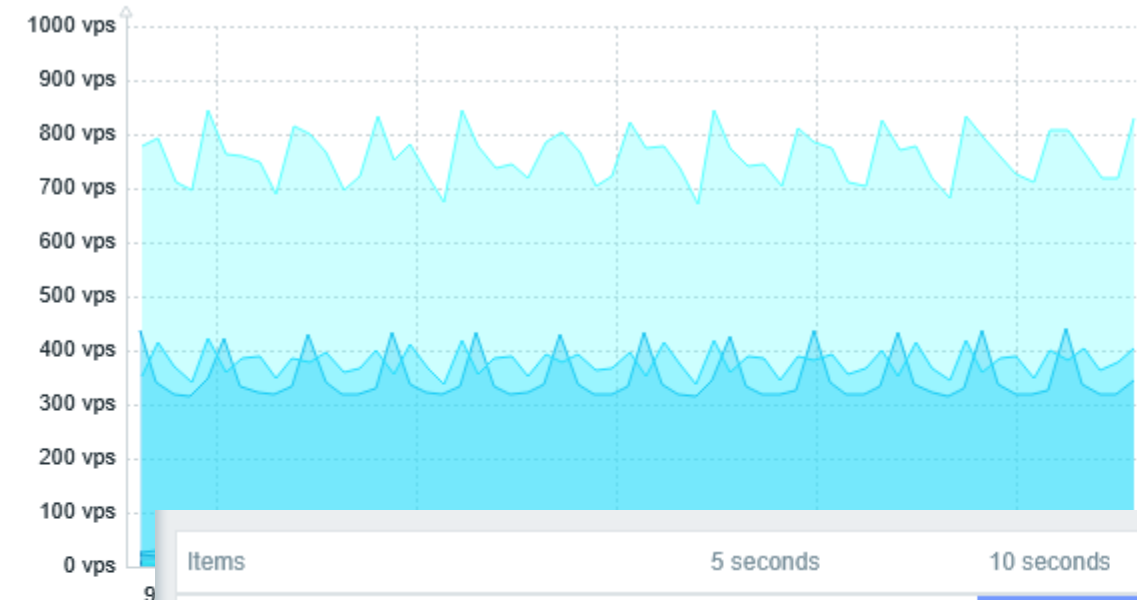
Value cache effectiveness



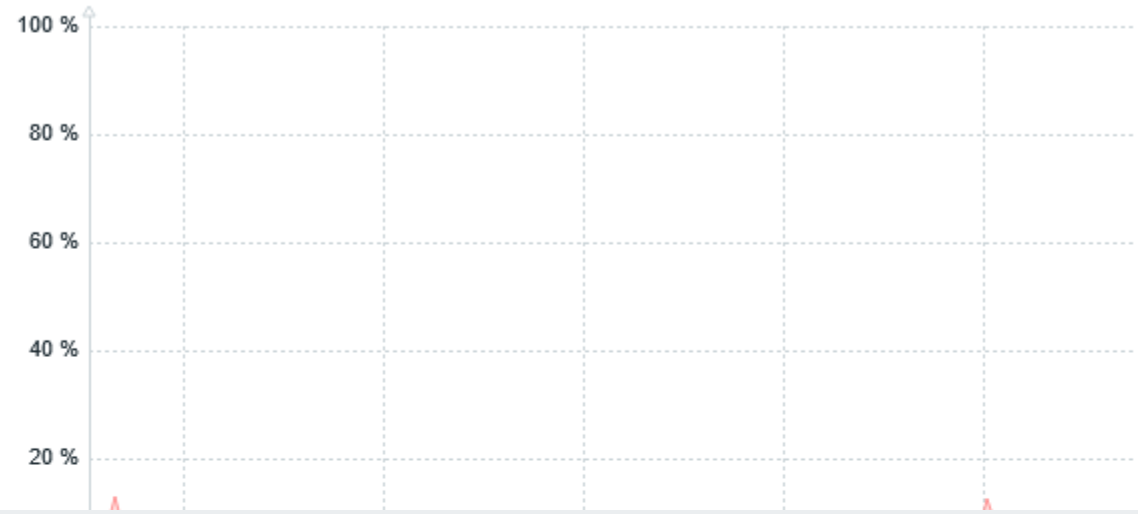
Queue size



Values processed per second



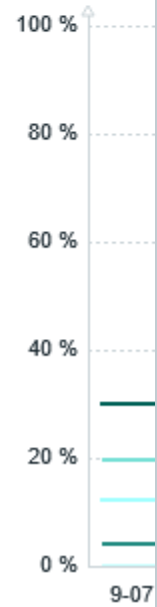
Utilization of data collectors



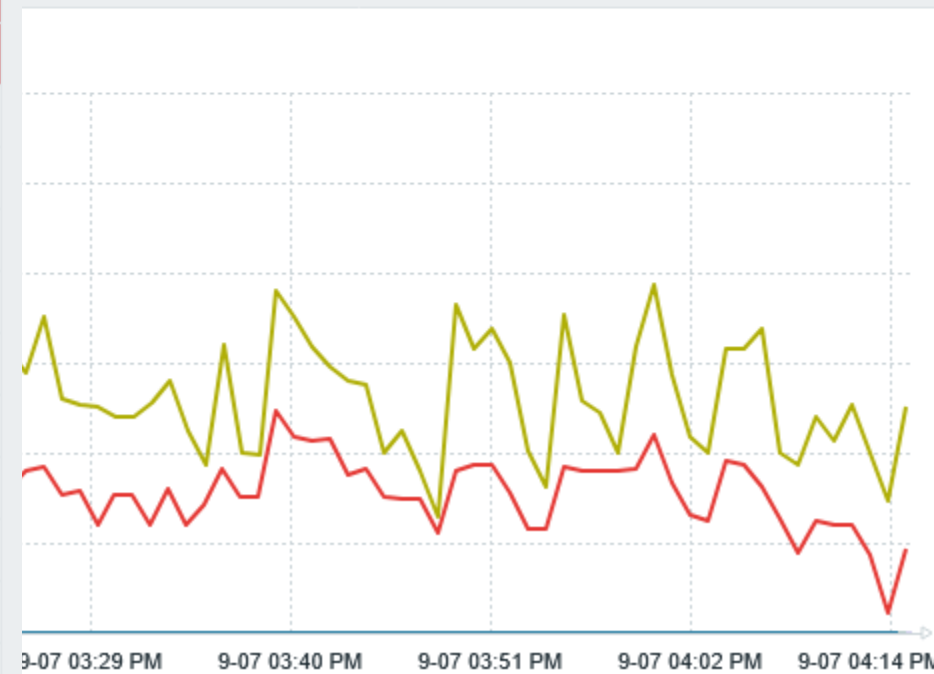
Utilization of internal processes



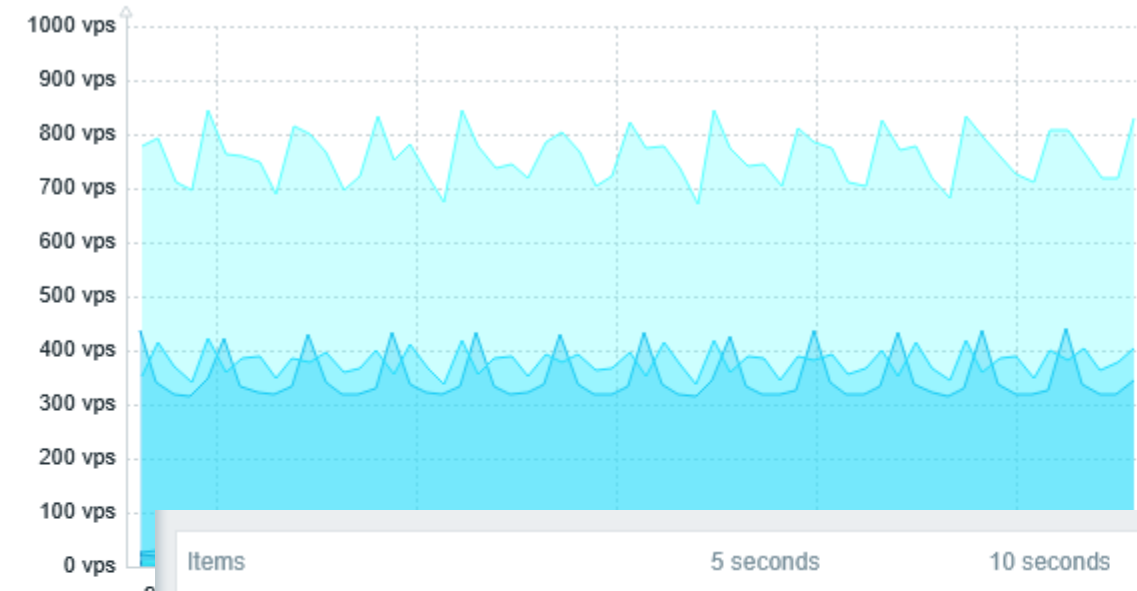
Cache usage



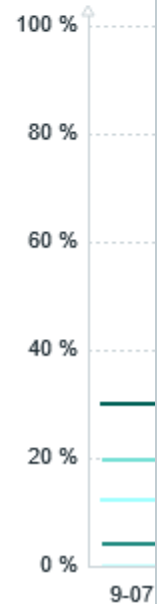
Items	5 seconds	10 seconds	30 seconds	1 minute	5 minutes	More than 10 minutes
Zabbix agent	0	14	8	11	1	87
Zabbix agent (active)	0	0	0	0	0	1
Simple check	0	0	0	0	0	0
SNMP agent	0	16	0	0	0	0
Zabbix internal	0	0	0	0	0	0
External check	0	0	0	0	0	0
Database monitor	0	0	0	0	0	0
HTTP agent	0	2	0	0	0	0
IPMI agent	0	0	0	0	0	0
SSH agent	0	0	0	0	0	0
TELNET agent	0	0	0	0	0	0
JMX agent	0	0	0	0	0	0
Calculated	0	0	0	0	0	0
Script	0	0	0	0	0	0
Browser	0	0	0	0	0	0



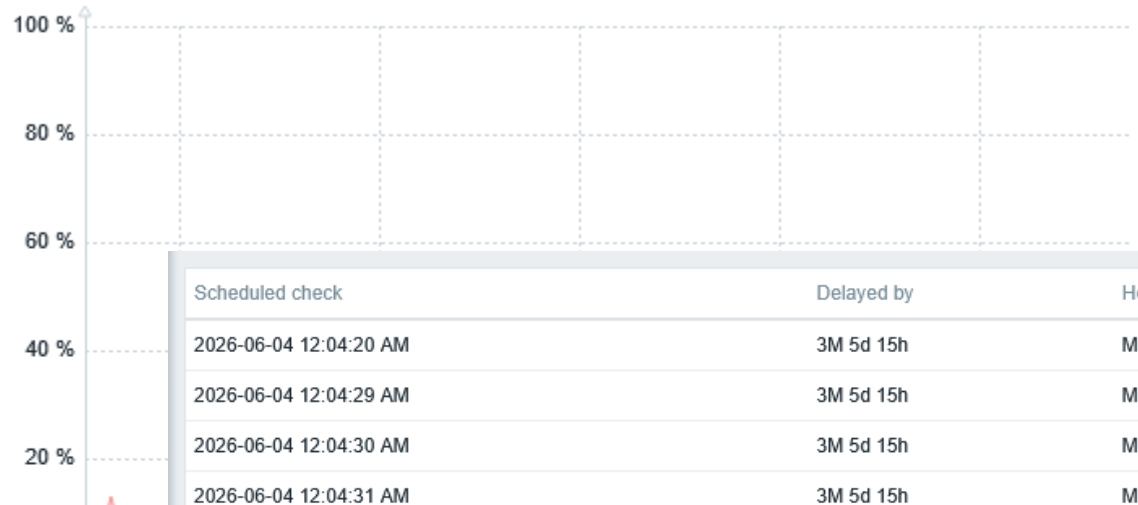
Values processed per second



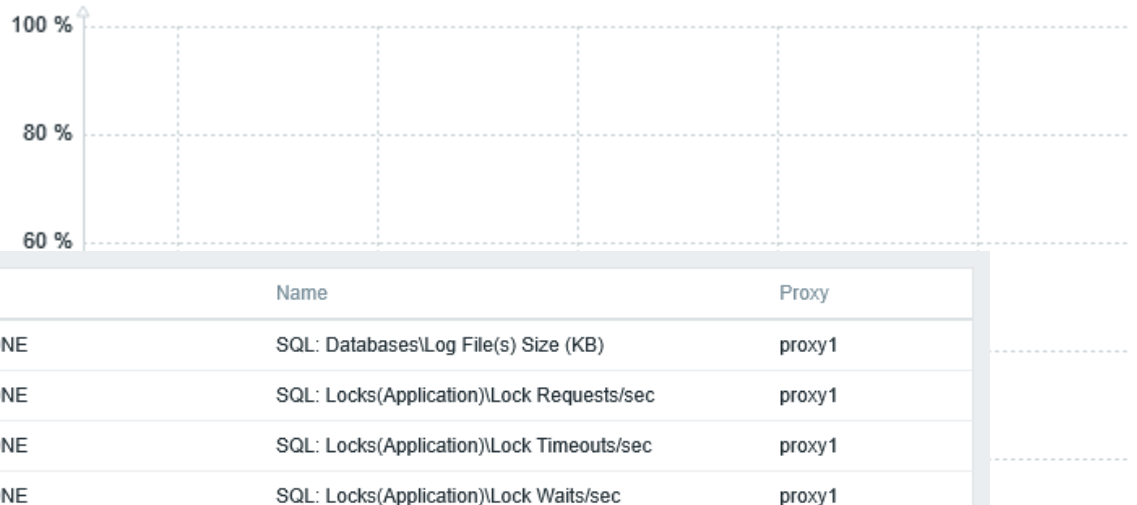
Cache usage



Utilization of data collectors



Utilization of internal processes



Items	5 seconds	10 seconds	30 seconds
Zabbix agent	0	14	8
Zabbix agent (active)	0	0	0
Simple check	0	0	0
SNMP agent	0	16	0
Zabbix internal	0	0	0
External check	0	0	0
Database monitor	0	0	0
HTTP agent	0	2	0
IPMI agent	0	0	0
SSH agent	0	0	0
TELNET agent	0	0	0
JMX agent	0	0	0
Calculated	0	0	0
Script	0	0	0
Browser	0	0	0

Scheduled check	Delayed by	Host	Name	Proxy
2026-06-04 12:04:20 AM	3M 5d 15h	MSSQL ONE	SQL: Databases\Log File(s) Size (KB)	proxy1
2026-06-04 12:04:29 AM	3M 5d 15h	MSSQL ONE	SQL: Locks(Application)\Lock Requests/sec	proxy1
2026-06-04 12:04:30 AM	3M 5d 15h	MSSQL ONE	SQL: Locks(Application)\Lock Timeouts/sec	proxy1
2026-06-04 12:04:31 AM	3M 5d 15h	MSSQL ONE	SQL: Locks(Application)\Lock Waits/sec	proxy1
2026-06-04 12:04:32 AM	3M 5d 15h	MSSQL ONE	SQL: Locks(Application)\Lock Wait Time (ms)	proxy1
2026-06-04 12:04:33 AM	3M 5d 15h	MSSQL ONE	SQL: Locks(Application)\Number of Deadlocks/sec	proxy1
2026-06-04 12:04:34 AM	3M 5d 15h	MSSQL ONE	SQL: Locks(Object)\Lock Requests/sec	proxy1
2026-06-04 12:04:35 AM	3M 5d 15h	MSSQL ONE	SQL: Locks(Object)\Lock Timeouts/sec	proxy1
2026-06-04 12:04:36 AM	3M 5d 15h	MSSQL ONE	SQL: Locks(Object)\Lock Waits/sec	proxy1
2026-06-04 12:04:42 AM	3M 5d 15h	MSSQL ONE	SQL: Locks(_Total)\Lock Wait Time (ms)	proxy1
2026-06-04 12:04:37 AM	3M 5d 15h	MSSQL ONE	SQL: Locks(Object)\Lock Wait Time (ms)	proxy1
2026-06-04 12:04:38 AM	3M 5d 15h	MSSQL ONE	SQL: Locks(Object)\Number of Deadlocks/sec	proxy1
2026-06-04 12:04:39 AM	3M 5d 15h	MSSQL ONE	SQL: Locks(_Total)\Lock Requests/sec	proxy1
2026-06-04 12:04:41 AM	3M 5d 15h	MSSQL ONE	SQL: Locks(_Total)\Lock Waits/sec	proxy1
2026-06-04 12:04:43 AM	3M 5d 15h	MSSQL ONE	SQL: Locks(_Total)\Number of Deadlocks/sec	proxy1
2026-06-04 12:09:03 AM	3M 5d 15h	MSSQL TWO	SQL: Databases\Log File(s) Size (KB)	proxy1
2026-06-04 12:09:12 AM	3M 5d 15h	MSSQL TWO	SQL: Locks(Application)\Lock Requests/sec	proxy1
2026-06-04 12:09:13 AM	3M 5d 15h	MSSQL TWO	SQL: Locks(Application)\Lock Timeouts/sec	proxy1
2026-06-04 12:04:40 AM	3M 5d 15h	MSSQL ONE	SQL: Locks(_Total)\Lock Timeouts/sec	proxy1
2026-06-04 12:09:14 AM	3M 5d 15h	MSSQL TWO	SQL: Locks(Application)\Lock Waits/sec	proxy1
2026-06-04 12:09:15 AM	3M 5d 15h	MSSQL TWO	SQL: Locks(Application)\Lock Wait Time (ms)	proxy1

Displaying 21 of 21 found

Values processed per second



Utilization of data collectors



Utilization of internal processes



Parameter	Value	Details
Zabbix server is running	Yes	192.168.1.10:10051
Zabbix server version	7.0.25	New update available
Zabbix frontend version	7.0.25	New update available
Software update last checked	2026-09-07	
Latest release	7.0.30	Release notes
Number of hosts (enabled/disabled)	2231	1673 / 558
Number of templates	338	
Number of items (enabled/disabled/not supported)	209378	164641 / 40474 / 4263
Number of triggers (enabled/disabled [problem/ok])	103870	82401 / 21469 [677 / 81724]
Number of users (online)	271	8
Required server performance, new values per second	1638.43	
Global scripts on Zabbix server	Disabled	
High availability cluster	Enabled	Fail-over delay: 1 minute

Name	Address	Last access	Status
ZBX01	192.168.1.10:10051	4s	Active
ZBX02	192.168.1.11:10051	4s	Standby

Agent	Item	Value	Unit	Time	Host
TELNET agent	0	0	0	2026-06-04 12:09:13 AM	MSSQL TWO
JMX agent	0	0	0	2026-06-04 12:04:40 AM	MSSQL ONE
Calculated	0	0	0	2026-06-04 12:09:14 AM	MSSQL TWO
Script	0	0	0	2026-06-04 12:09:15 AM	MSSQL TWO
Browser	0	0	0		

Name	Proxy
SQL: Databases\Log File(s) Size (KB)	proxy1
SQL: Locks(Application)\Lock Requests/sec	proxy1
SQL: Locks(Application)\Lock Timeouts/sec	proxy1
SQL: Locks(Application)\Lock Waits/sec	proxy1
SQL: Locks(Application)\Lock Wait Time (ms)	proxy1
SQL: Locks(Application)\Number of Deadlocks/sec	proxy1
SQL: Locks(Object)\Lock Requests/sec	proxy1
SQL: Locks(Object)\Lock Timeouts/sec	proxy1
SQL: Locks(Object)\Lock Waits/sec	proxy1
SQL: Locks(_Total)\Lock Wait Time (ms)	proxy1
SQL: Locks(Object)\Lock Wait Time (ms)	proxy1
SQL: Locks(Object)\Number of Deadlocks/sec	proxy1
SQL: Locks(_Total)\Lock Requests/sec	proxy1
SQL: Locks(_Total)\Lock Waits/sec	proxy1
SQL: Locks(_Total)\Number of Deadlocks/sec	proxy1
SQL: Databases\Log File(s) Size (KB)	proxy1
SQL: Locks(Application)\Lock Requests/sec	proxy1
SQL: Locks(Application)\Lock Timeouts/sec	proxy1
SQL: Locks(_Total)\Lock Timeouts/sec	proxy1
SQL: Locks(Application)\Lock Waits/sec	proxy1
SQL: Locks(Application)\Lock Wait Time (ms)	proxy1



Values processed per second



Utilization of data collectors



Utilization of internal processes



Parameter	Value	Details
Zabbix server is running	Yes	192.168.1.10:10051
Zabbix server version	7.0.25	New update available
Zabbix frontend version	7.0.25	New update available
Software update last checked	2026-09-07	
Latest release	7.0.30	Release notes
Number of hosts (enabled/disabled)	2231	1673 / 558
Number of templates	338	
Number of items (enabled/disabled/not supported)	209378	164641 / 40474 / 4263
Number of triggers (enabled/disabled [problem/ok])	10	
Number of users (online)	27	
Required server performance, new values per second	16	
Global scripts on Zabbix server	Dis	
High availability cluster	En	

Name	Address
ZBX01	192.168.1.10:10051
ZBX02	192.168.1.11:10051

TELNET agent	0	0	0
JMX agent	0	0	0
Calculated	0	0	0
Script	0	0	0
Browser	0	0	0

Parameter	Value	Details
Zabbix server is running	Yes	192.168.1.10:10051
Zabbix server version	7.0.25	New update available
Zabbix frontend version	7.0.25	New update available
Software update last checked	2026-09-07	
Latest release	7.0.30	Release notes
Number of hosts (enabled/disabled)	2231	1673 / 558
Number of templates	338	
Number of items (enabled/disabled/not supported)	209378	164641 / 40474 / 4263
Number of triggers (enabled/disabled [problem/ok])	10	
Number of users (online)	27	
Required server performance, new values per second	16	
Global scripts on Zabbix server	Dis	
High availability cluster	En	

```
2035055:20260908:125141.249 itemid:23266 active range:601 hits:11 count:12 perc:0.253646%
2035055:20260908:125141.249 itemid:23252 active range:601 hits:22 count:12 perc:0.253646%
2035055:20260908:125141.249 itemid:45499 active range:601 hits:22 count:12 perc:0.253646%
2035055:20260908:125141.249 =====
2035055:20260908:125141.249 === memory statistics for value cache size ===
2035055:20260908:125141.249 free chunks of size 64 bytes: 1
2035055:20260908:125141.249 free chunks of size 80 bytes: 1
2035055:20260908:125141.249 free chunks of size >= 256 bytes: 1
2035055:20260908:125141.249 min chunk size: 64 bytes
2035055:20260908:125141.249 max chunk size: 14240 bytes
2035055:20260908:125141.249 memory of total size 124656 bytes fragmented into 378 chunks
2035055:20260908:125141.249 of those, 14384 bytes are in 3 free chunks
2035055:20260908:125141.249 of those, 110272 bytes are in 375 used chunks
2035055:20260908:125141.249 of those, 6032 bytes are used by allocation overhead
2035055:20260908:125141.249 =====
2035055:20260908:125141.249 value cache is fully used: please increase ValueCacheSize configuration parameter
2035055:20260908:125141.249 cannot allocate 65536 bytes for itemid:51200 in value cache
```

2026-06-04 12:09:15 AM	3M 5d 15h	MSSQL TWO	SQL: Locks(Application)\Lock Wait Time (ms)	proxy1
------------------------	-----------	-----------	---	--------

Values processed per second



Utilization of data collectors



Utilization of internal processes

Parameter	Value	Details
Zabbix server is running	Yes	192.168.1.10:10051
Zabbix server version	7.0.25	New update available
Zabbix frontend version	7.0.25	New update available

Name	Proxy
SQL: Databases\Log File(s) Size (KB)	proxy1
SQL: Locks(Application)\Lock Requests/sec	proxy1
SQL: Locks(Application)\Lock Timeouts/sec	proxy1
SQL: Locks(Application)\Lock Waits/sec	proxy1
SQL: Locks(Application)\Lock Wait Time (ms)	proxy1
SQL: Locks(Application)\Number of Deadlocks/sec	proxy1
SQL: Locks(Object)\Lock Requests/sec	proxy1

```
== history cache diagnostic information ==
Items:0 values:0 time:0.000014
Memory.data:
  size: free:16776832 used:0
  chunks: free:1 used:0 min:16776832 max:16776832
  buckets:
    256+:1
Memory.index:
  size: free:4175928 used:17880
  chunks: free:2 used:5 min:2624 max:4173304
```

```
6 active range:601 hits:11 count:12 perc:0.253646%
2 active range:601 hits:22 count:12 perc:0.253646%
9 active range:601 hits:22 count:12 perc:0.253646%
```

```

statistics for value cache size ==
  of size      64 bytes:      1
  of size      80 bytes:      1
  of size >= 256 bytes:      1
size:          64 bytes
size:         14240 bytes
total size 124656 bytes fragmented into 378 chunks
    14384 bytes are in          3 free chunks
    110272 bytes are in        375 used chunks
    6032 bytes are used by allocation overhead

```

Name
ZBX01
ZBX02

TELNET a
JMX agent
Calculated

Script	0	0	0			
Browser	0	0	0	0	0	0

All dashboards / Zabbix server health

Values processed per second

Utilization of data collectors

Utilization of internal processes

Parameter

Value

Details

Zabbix server is running

Yes

192.168.1.10:10051

Zabbix server version

Zabbix frontend version

Software update last checked

Latest release

Number of hosts (enabled)

Number of templates

Number of items (enabled)

Number of triggers (enabled)

Number of users (online)

Required server performance

Global scripts on Zabbix

High availability cluster

Name

ZBX01

ZBX02

TELNET agent

JMX agent

Calculated

Script

0

0

0

0

0

0

Browser

0

0

0

0

0

0

zabbix-server.service - Zabbix Server

Loaded: loaded (/usr/lib/systemd/system/zabbix-server.service; disabled; preset: enabled)

Active: active (running) since Mon 2026-09-07 10:33:45 CEST; 1 day 2h ago

Process: 314673 ExecStart=/usr/sbin/zabbix_server -c \$CONFFILE (code=exited, status=0/SUCCESS)

Main PID: 314675 (zabbix_server)

Tasks: 69 (limit: 4563)

Memory: 68.1M (peak: 83.3M)

CPU: 7min 31.599s

CGroup: /system.slice/zabbix-server.service

—314675 /usr/sbin/zabbix_server -c /etc/zabbix/zabbix_server.conf

—314677 "/usr/sbin/zabbix_server: ha manager"

—314679 "/usr/sbin/zabbix_server: service manager #1 [processed 0 events, updated 0 event tags, deleted 0 problems, s

—314680 "/usr/sbin/zabbix_server: configuration syncer [synced configuration in 0.018930 sec, idle 10 sec]"

—314685 "/usr/sbin/zabbix_server: alert manager #1 [sent 0, failed 0 alerts, idle 5.020764 sec during 5.020879 sec]"

—314686 "/usr/sbin/zabbix_server: alerter #1 started"

—314687 "/usr/sbin/zabbix_server: alerter #2 started"

—314688 "/usr/sbin/zabbix_server: alerter #3 started"

—314689 "/usr/sbin/zabbix_server: preprocessing manager #1 [queued 3, processed 2 values, idle 5.146275 sec during 5.

—314690 "/usr/sbin/zabbix_server: lld manager #1 [processed 0 LLD rules, idle 5.168234sec during 5.168368 sec]"

—314691 "/usr/sbin/zabbix_server: lld worker #1 [processed 1 LLD rules, idle 477.980568 sec during 478.002374 sec]"

—314692 "/usr/sbin/zabbix_server: lld worker #2 [processed 1 LLD rules, idle 602.004457 sec during 602.025851 sec]"

—314694 "/usr/sbin/zabbix_server: housekeeper [deleted 120 hist/trends, 0 items/triggers, 0 events, 0 sessions, 0 ala

—314695 "/usr/sbin/zabbix_server: timer #1 [updated 0 hosts, suppressed 0 events in 0.000955 sec, idle 59 sec]"

—314696 "/usr/sbin/zabbix_server: http poller #1 [got 0 values in 0.000030 sec, idle 5 sec]"

—314700 "/usr/sbin/zabbix_server: browser poller #1 [got 0 values in 0.000019 sec, idle 5 sec]"

—314717 "/usr/sbin/zabbix_server: discovery manager #1 [processing 0 rules, 0 unsaved checks]"

=====

is fully used: please increase ValueCacheSize configuration parameter

cate 65536 bytes for itemid:51200 in value cache

Proxy

proxy1

proxy1

proxy1

proxy1

proxy1

proxy1

proxy1

proxy1

proxy1

06 PM

MSSQL TWO

SQL: Locks(Application)\Lock Wait Time (ms)

proxy1

07 04:14 PM

Displaying 21 of 21 found



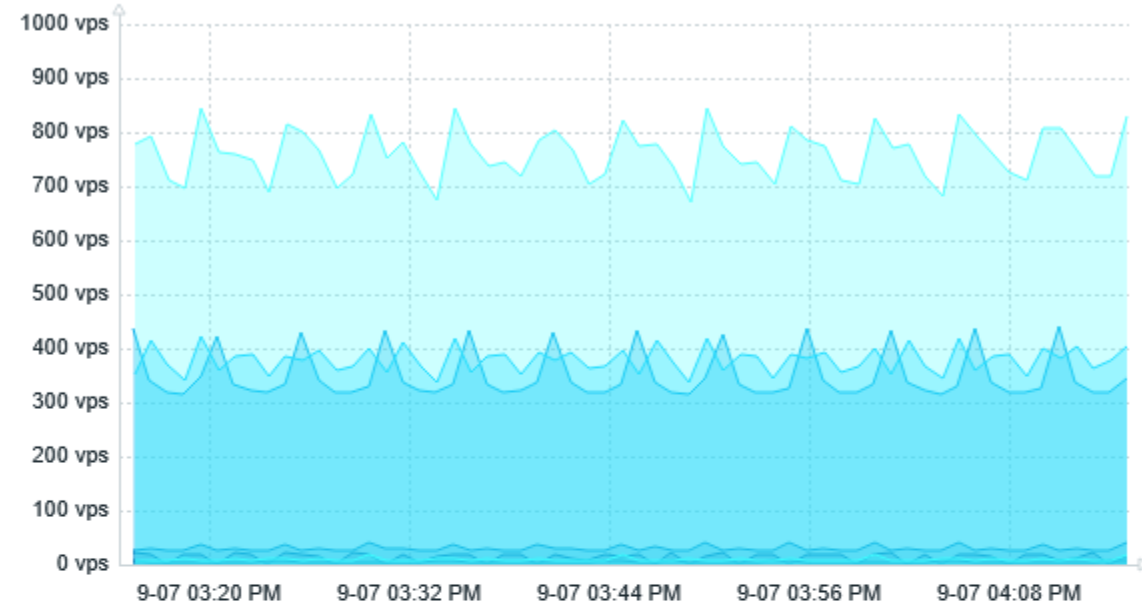
To confuse your enemy, you must first confuse
yourself.

Sun Tzu (allegedly)

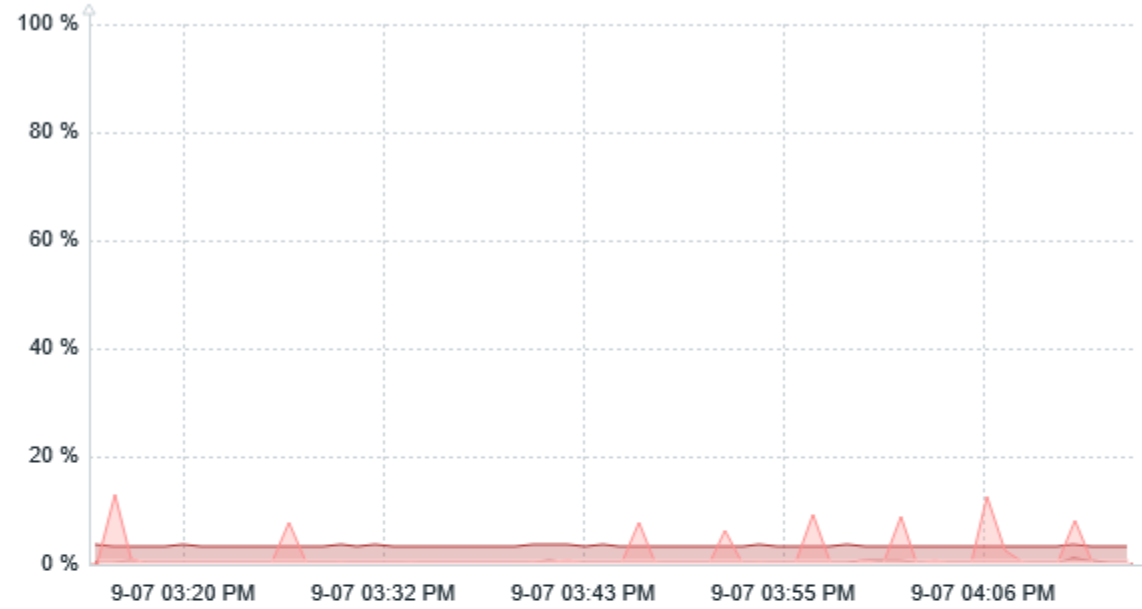
Zabbix Server Health



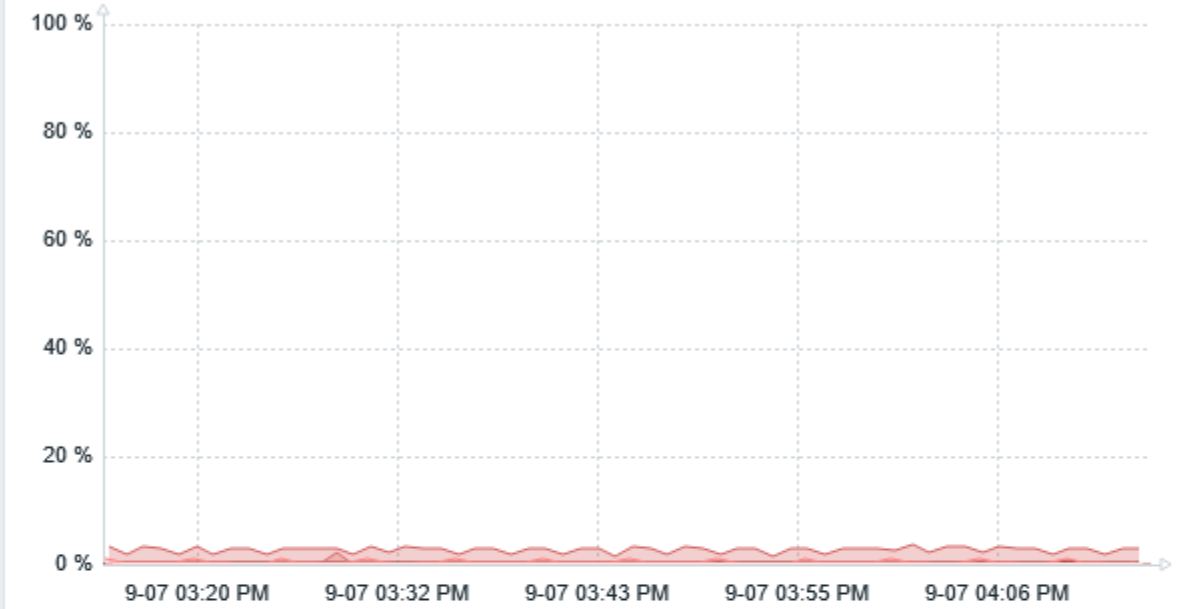
Values processed per second



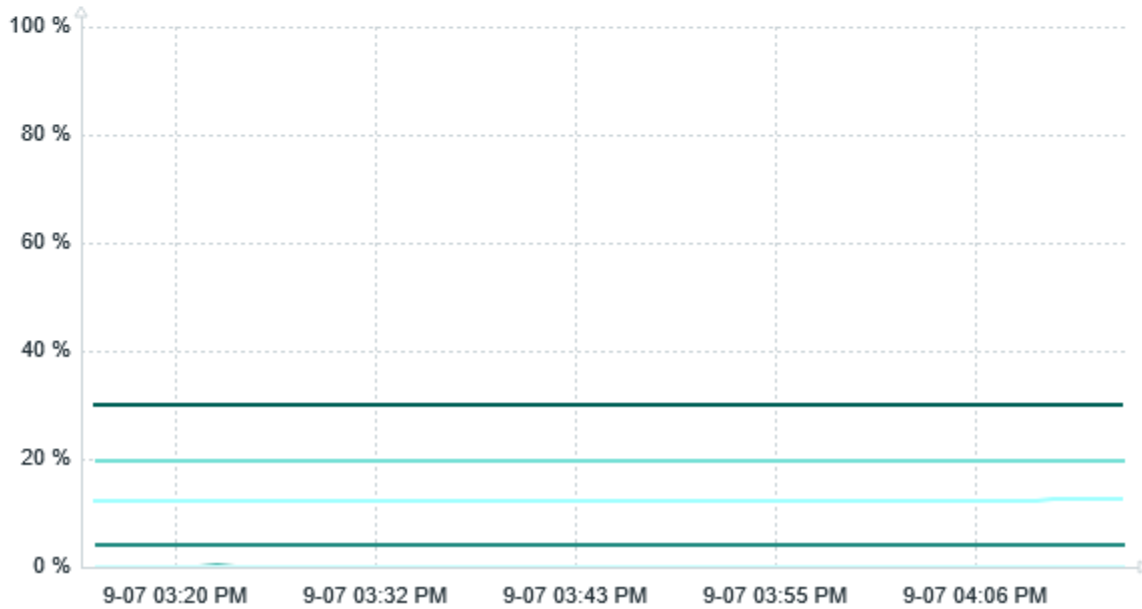
Utilization of data collectors



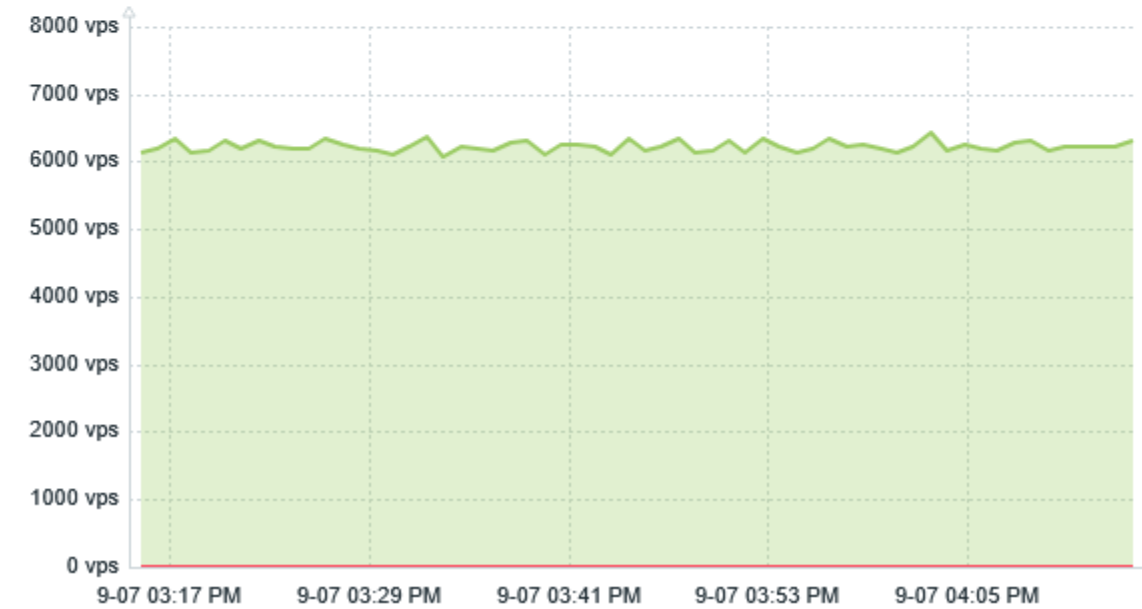
Utilization of internal processes



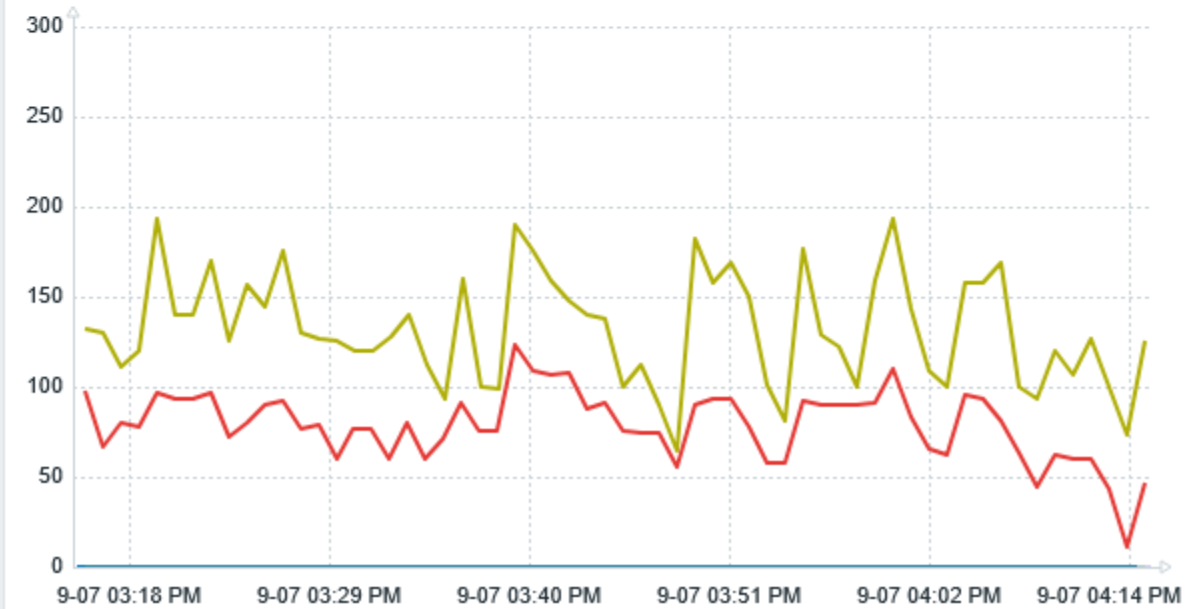
Cache usage



Value cache effectiveness



Queue size

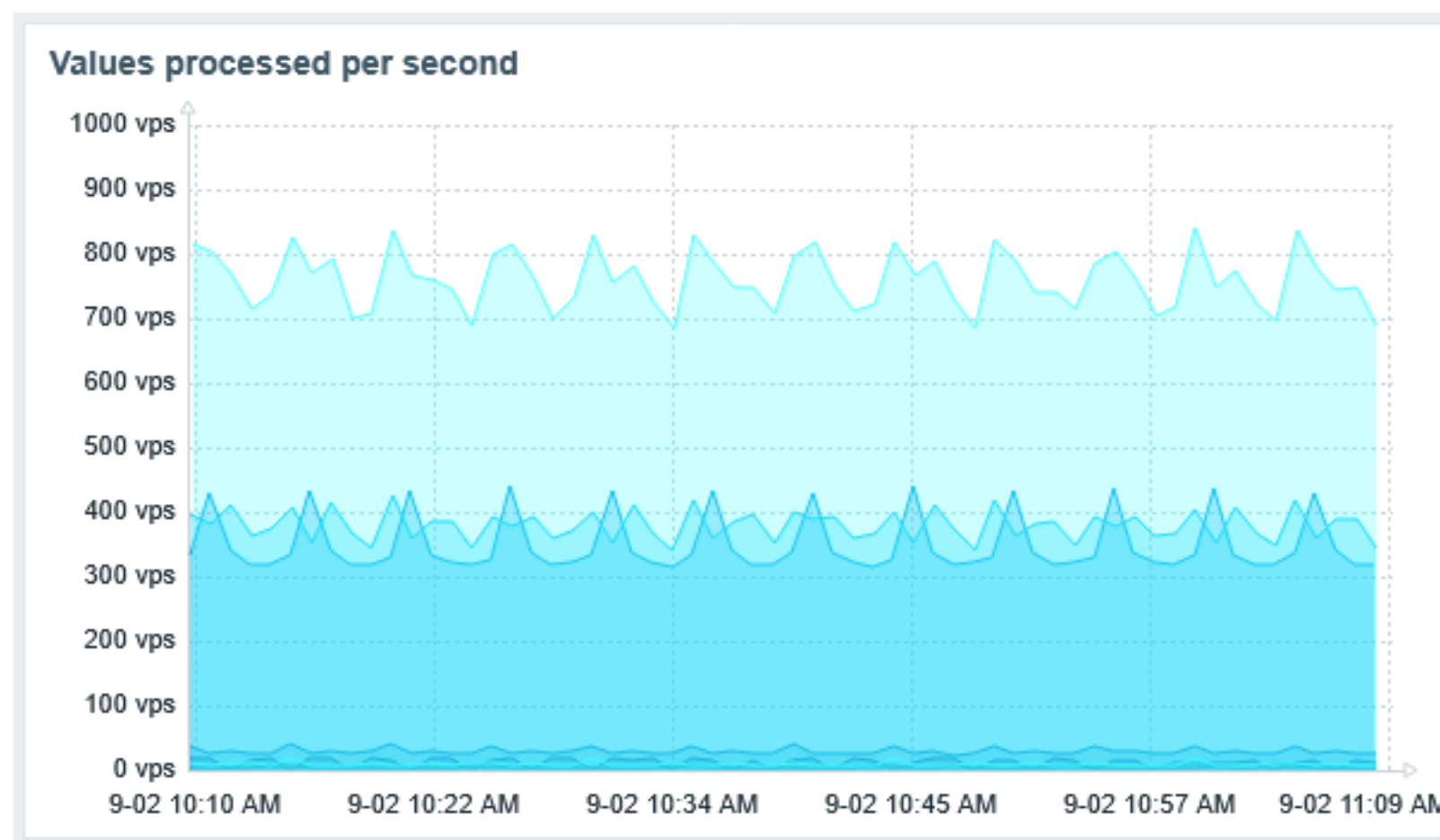


Your favorite dashboard

Zabbix Server Health

Values per second

Measuring Zabbix size



`zabbix[wcache,values,<mode>]`

```
static void    hc_add_item_values(dc_item_value_t *values, int values_num)
{
    dc_item_value_t    *item_value;
    int                i;
    zbx_hc_item_t      *item;

    for (i = 0; i < values_num; i++)
    {
        (...)
        if (SUCCEED != hc_clone_history_data(&data, item_value))
        {
            do
            {
                UNLOCK_CACHE;

                zabbix_log(LOG_LEVEL_DEBUG, "History cache is full. Sleeping for 1 second.");
                sleep(1);

                LOCK_CACHE;
            }
            while (SUCCEED != hc_clone_history_data(&data, item_value));

            item = hc_get_item(item_value->itemid);

            (...)
            item->values_num++;
        }
    }
}
```

Counting while cloning

```

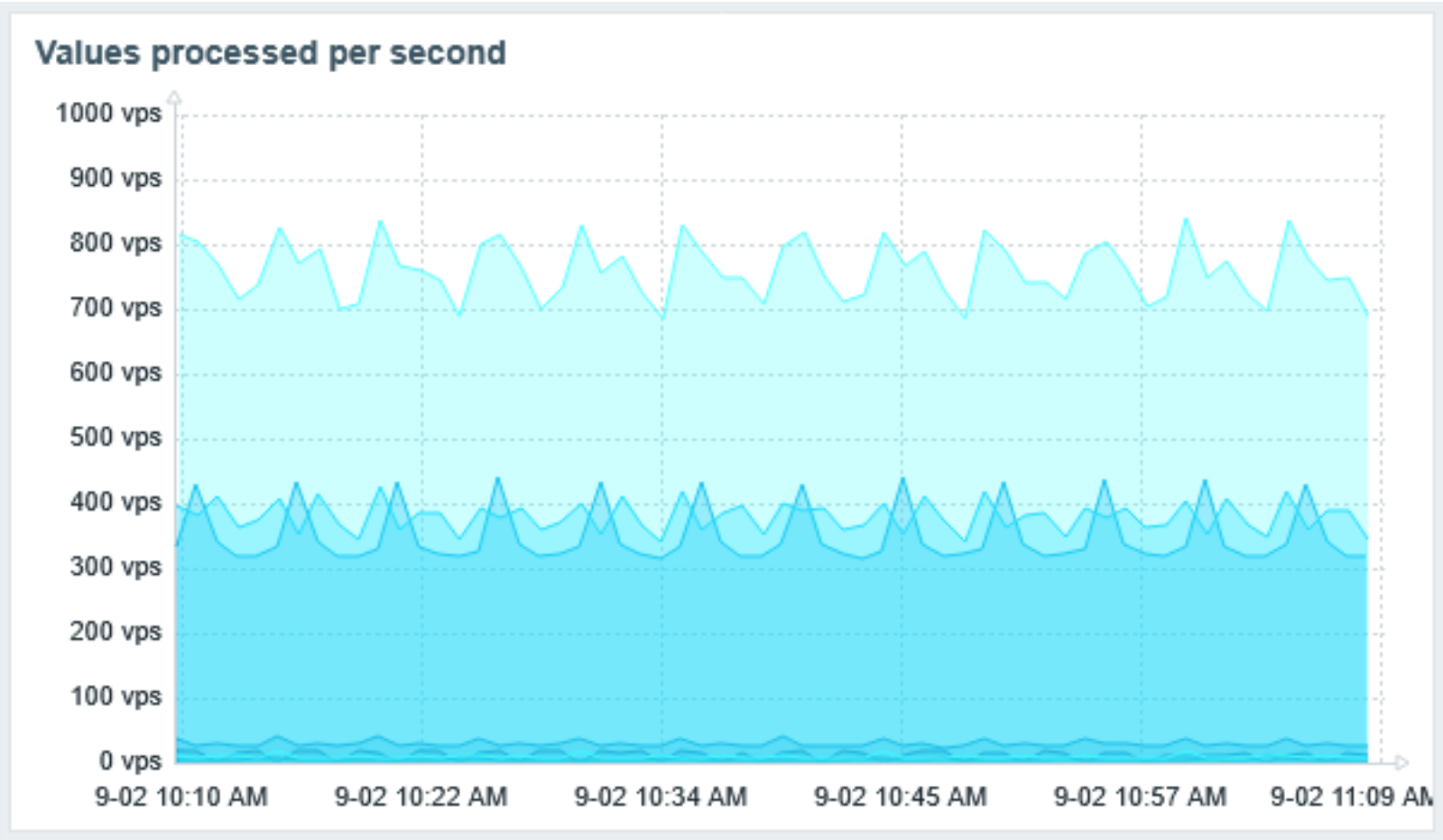
static int hc_clone_history_data(zbx_hc_data_t **data, const dc_item_value_t *item_value)
{
    (...)
    if (0 == (ZBX_DC_FLAG_NOVALUE & item_value->flags))
    {
        (...)
        switch (item_value->item_value_type)
        {
            case ITEM_VALUE_TYPE_FLOAT:
                cache->stats.history_float_counter++;
                break;
            case ITEM_VALUE_TYPE_UINT64:
                cache->stats.history_uint_counter++;
                break;
            case ITEM_VALUE_TYPE_STR:
                cache->stats.history_str_counter++;
                break;
            case ITEM_VALUE_TYPE_TEXT:
                cache->stats.history_text_counter++;
                break;
            case ITEM_VALUE_TYPE_LOG:
                cache->stats.history_log_counter++;
                break;
            case ITEM_VALUE_TYPE_BIN:
                cache->stats.history_bin_counter++;
                break;
            case ITEM_VALUE_TYPE_NONE:
            default:
                THIS_SHOULD_NEVER_HAPPEN;
                exit(EXIT_FAILURE);
        }
        cache->stats.history_counter++;
    }
    (...)
    item
    return SUCCEED;
}

```

second.");

Counting while cloning

Measuring Zabbix size



zabbix[wcache,values,<mode>]

Parameter	Value	Details	
Zabbix server is running	Yes	192.168.1.10:10051	
Zabbix server version	7.0.25	New update available	
Zabbix frontend version	7.0.25	New update available	
Software update last checked	2026-09-07		
Latest release	7.0.30	Release notes	
Number of hosts (enabled/disabled)	2231	1673 / 558	
Number of templates	338		
Number of items (enabled/disabled/not supported)	209378	164641 / 40474 / 4263	
Number of triggers (enabled/disabled [problem/ok])	103870	82401 / 21469 [677 / 81724]	
Number of users (online)	271	8	
Required server performance, new values per second	1638.43		
Global scripts on Zabbix server	Disabled		
High availability cluster	Enabled	Fail-over delay: 1 minute	
Name	Address	Last access	Status
ZBX01	192.168.1.10:10051	4s	Active
ZBX02	192.168.1.11:10051	4s	Standby

Required server performance, new values per second 1638.43

Required server performance

```
static void update_required_performance(const ZBX_DC_ITEM *dc_item, zbx_dc_status_diff_proxy_t *proxy_diff,
                                       zbx_dc_status_diff_t *diff)
{
    int delay;
    char *delay_s;

    delay_s = dc_expand_user_and_func_macros_dyn(dc_item->delay, &dc_item->hostid, 1, ZBX_MACRO_ENV_NONSECURE);

    if (SUCCEED == zbx_interval_preproc(delay_s, &delay, NULL, NULL) && 0 != delay)
    {
        int item_count;

        if (0 < (item_count = get_active_item_count_rec(dc_item)))
        {
            diff->required_performance += 1.0 / delay * item_count;

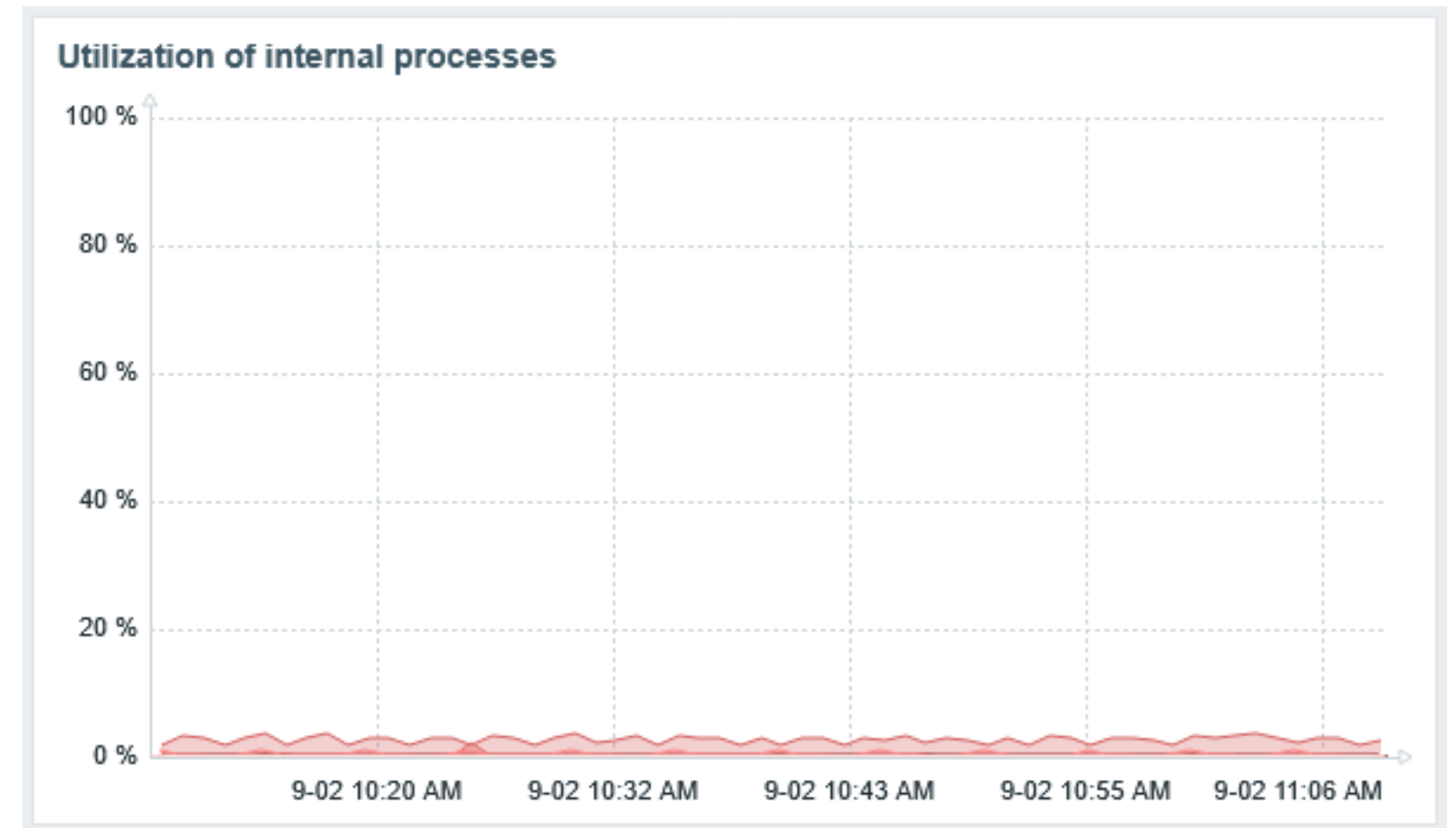
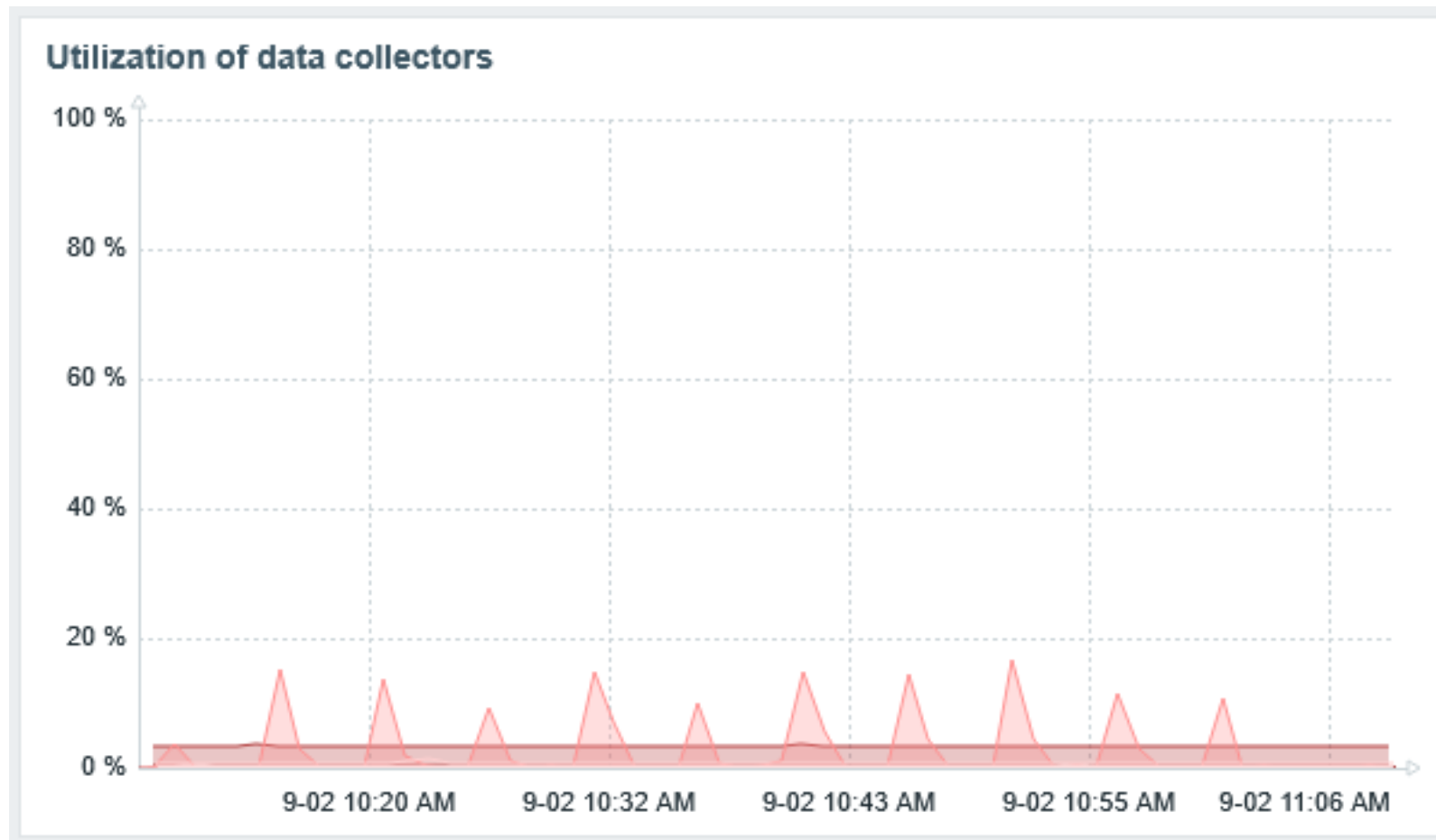
            if (NULL != proxy_diff)
                proxy_diff->required_performance += 1.0 / delay * item_count;
        }
    }

    zbx_free(delay_s);
}
```

Zabbix Server Health

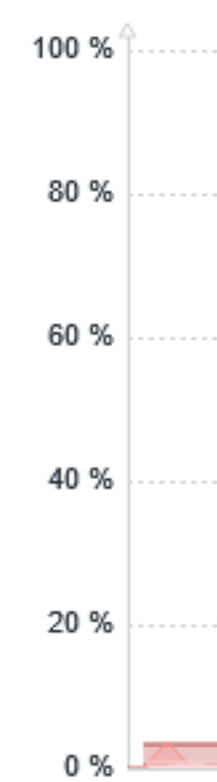
Process utilization

Utilization of processes



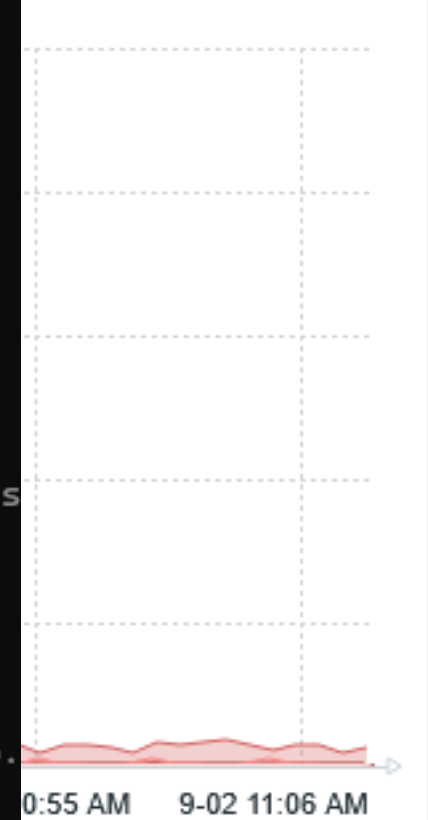
Utilization of processes

Utilization of data collectors



```
● zabbix-server.service - Zabbix Server
   Loaded: loaded (/usr/lib/systemd/system/zabbix-server.service; disabled; preset: enabled)
   Active: active (running) since Mon 2026-09-07 10:33:45 CEST; 1 day 2h ago
   Process: 314673 ExecStart=/usr/sbin/zabbix_server -c $CONFFILE (code=exited, status=0/SUCCESS)
   Main PID: 314675 (zabbix_server)
     Tasks: 69 (limit: 4563)
    Memory: 68.1M (peak: 83.3M)
       CPU: 7min 31.599s
    CGroup: /system.slice/zabbix-server.service
           └─314675 /usr/sbin/zabbix_server -c /etc/zabbix/zabbix_server.conf
             └─314677 "/usr/sbin/zabbix_server: ha manager"
               └─314679 "/usr/sbin/zabbix_server: service manager #1 [processed 0 events, updated 0 event tags, deleted 0 problems, s
                 └─314680 "/usr/sbin/zabbix_server: configuration syncer [synced configuration in 0.018930 sec, idle 10 sec]"
                   └─314685 "/usr/sbin/zabbix_server: alert manager #1 [sent 0, failed 0 alerts, idle 5.020764 sec during 5.020879 sec]"
                     └─314686 "/usr/sbin/zabbix_server: alerter #1 started"
                       └─314687 "/usr/sbin/zabbix_server: alerter #2 started"
                         └─314688 "/usr/sbin/zabbix_server: alerter #3 started"
                           └─314689 "/usr/sbin/zabbix_server: preprocessing manager #1 [queued 3, processed 2 values, idle 5.146275 sec during 5.
                             └─314690 "/usr/sbin/zabbix_server: lld manager #1 [processed 0 LLD rules, idle 5.168234sec during 5.168368 sec]"
                               └─314691 "/usr/sbin/zabbix_server: lld worker #1 [processed 1 LLD rules, idle 477.980568 sec during 478.002374 sec]"
                                 └─314692 "/usr/sbin/zabbix_server: lld worker #2 [processed 1 LLD rules, idle 602.004457 sec during 602.025851 sec]"
                                   └─314694 "/usr/sbin/zabbix_server: housekeeper [deleted 120 hist/trends, 0 items/triggers, 0 events, 0 sessions, 0 ala
                                     └─314695 "/usr/sbin/zabbix_server: timer #1 [updated 0 hosts, suppressed 0 events in 0.000955 sec, idle 59 sec]"
                                       └─314696 "/usr/sbin/zabbix_server: http poller #1 [got 0 values in 0.000030 sec, idle 5 sec]"
                                         └─314700 "/usr/sbin/zabbix_server: browser poller #1 [got 0 values in 0.000019 sec, idle 5 sec]"
                                           └─314717 "/usr/sbin/zabbix_server: discovery manager #1 [processing 0 rules, 0 unsaved checks]"
```

Utilization of internal processes



What is “busy?”

The percentage of time a particular Zabbix process or a group of processes spends in a busy state.

- Connecting to the database and querying
- Processing tasks
- Fetching and updating configuration cache
- Availability updates
- Processing alerts and notifications
- Processing IPC messages
- Preprocessing, aggregation, discovery, housekeeping, polling, etc.
- Sending data over network connections

What is "idle?"

The percentage of time a particular Zabbix process or a group of processes spends in idle state.

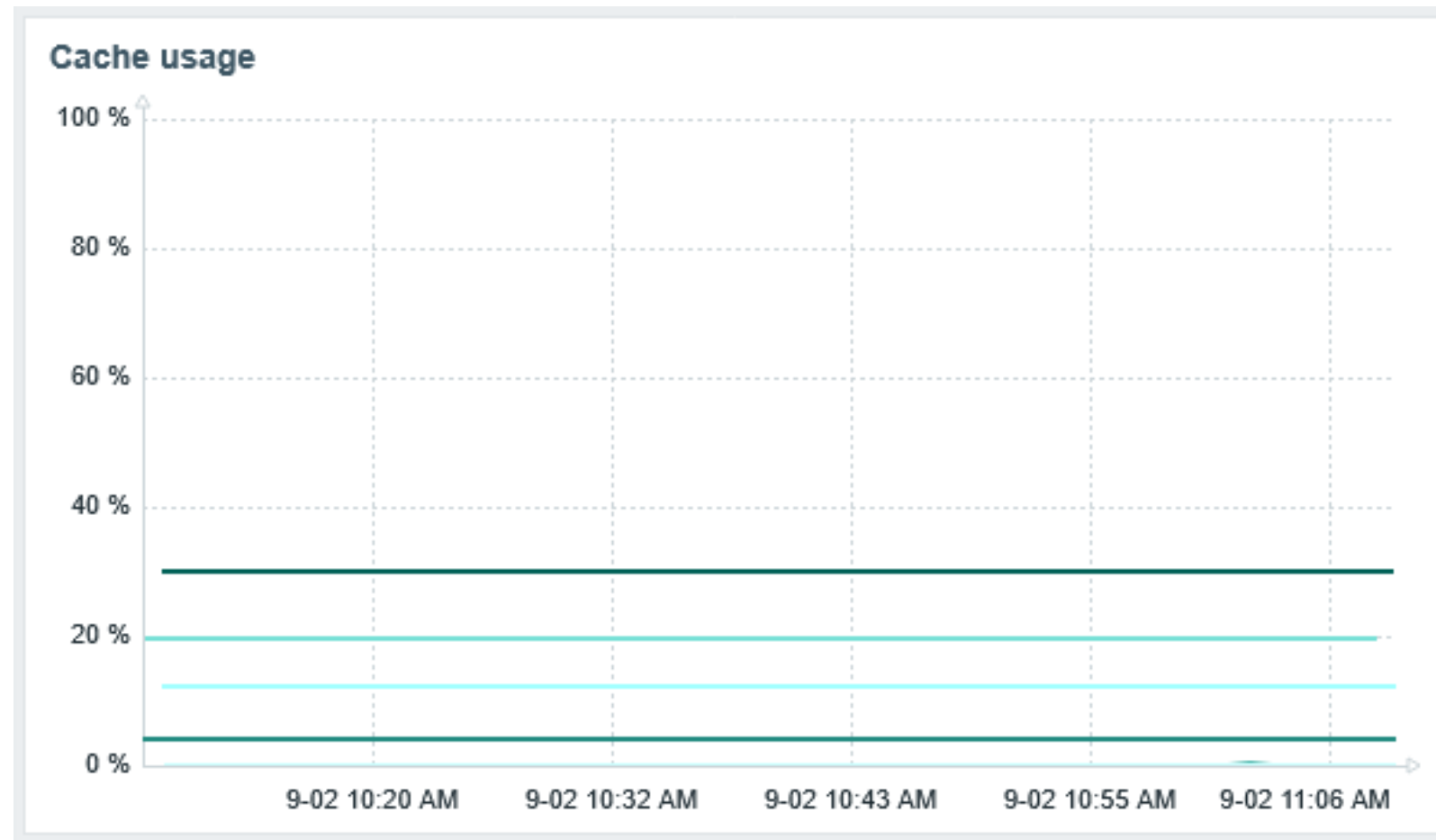
- Sleeping
- Waiting for IPC messages or other events
- Waiting for the next cycle/check
- Waiting for timers/timeouts
- Waiting for work from queues
- Waiting for scheduler events

Zabbix Server Health

Caches

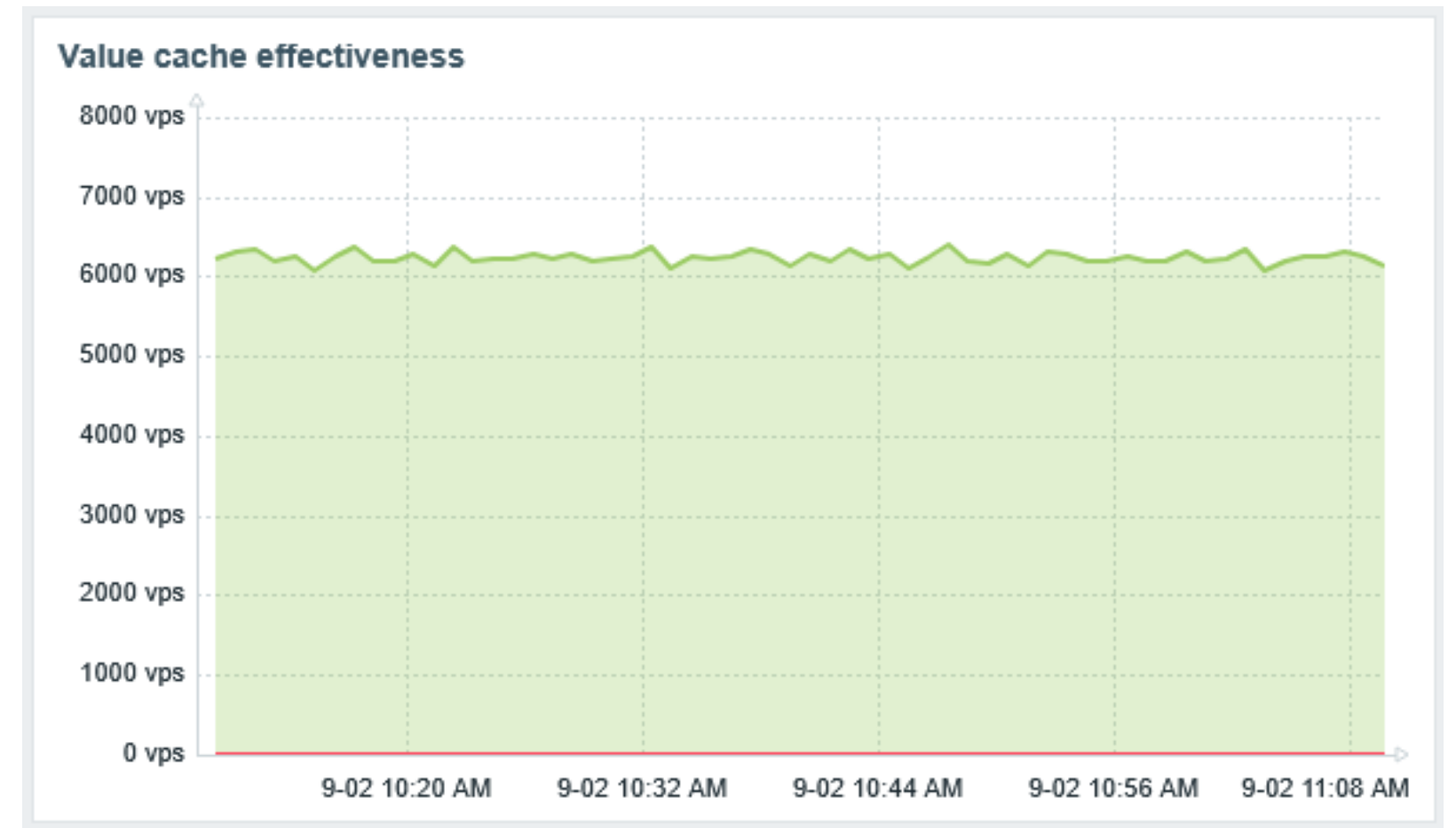


Cache makes Zabbix fast



Assorted cache usage

Do not let the percentage deceive you.



Hits and Misses

Green is best, red is bad.

Value cache

Utilization

diaginfo=valuecache

```
# zabbix_server -R diaginfo=valuecache
== value cache diagnostic information ==
Items:94 values:798 mode:0 time:0.000037
Memory:
  size: free:8331880 used:49816
  chunks: free:27 used:382 min:24 max:8324936
  buckets:
    24:1
    32:5
    48:4
    56:1
    64:3
    80:1
    96:1
    256+:11
```

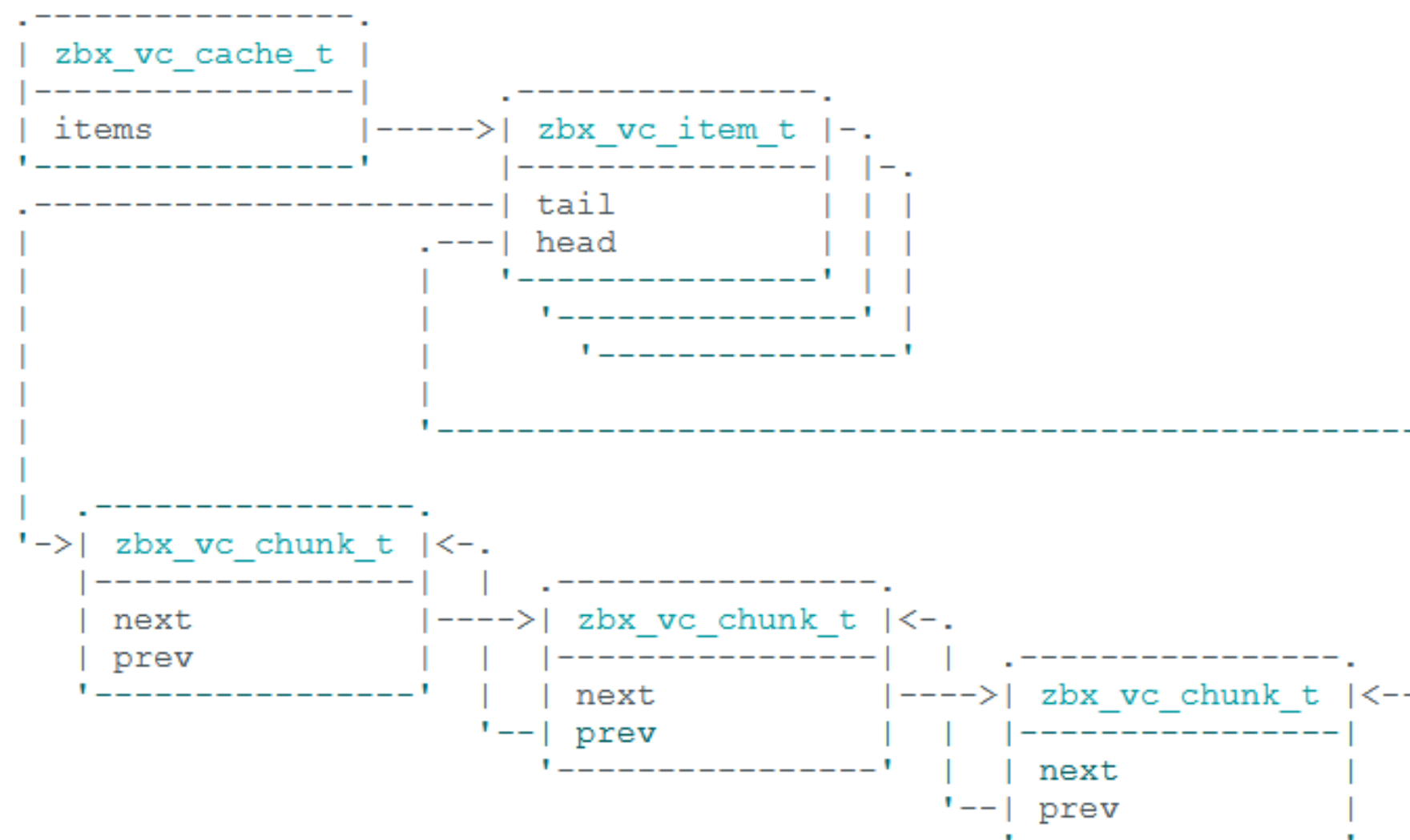
Memory allocation

```
Top.values:
  itemid:51189 values:20 request.values:16
  itemid:51190 values:18 request.values:16
  itemid:23259 values:13 request.values:11
  itemid:29823 values:13 request.values:11
  itemid:23258 values:13 request.values:11
  (...)
Top.request.values:
  itemid:51189 values:20 request.values:16
  itemid:51190 values:18 request.values:16
  itemid:23259 values:13 request.values:11
  itemid:29823 values:13 request.values:11
  itemid:23258 values:13 request.values:11
  (...)
==
```

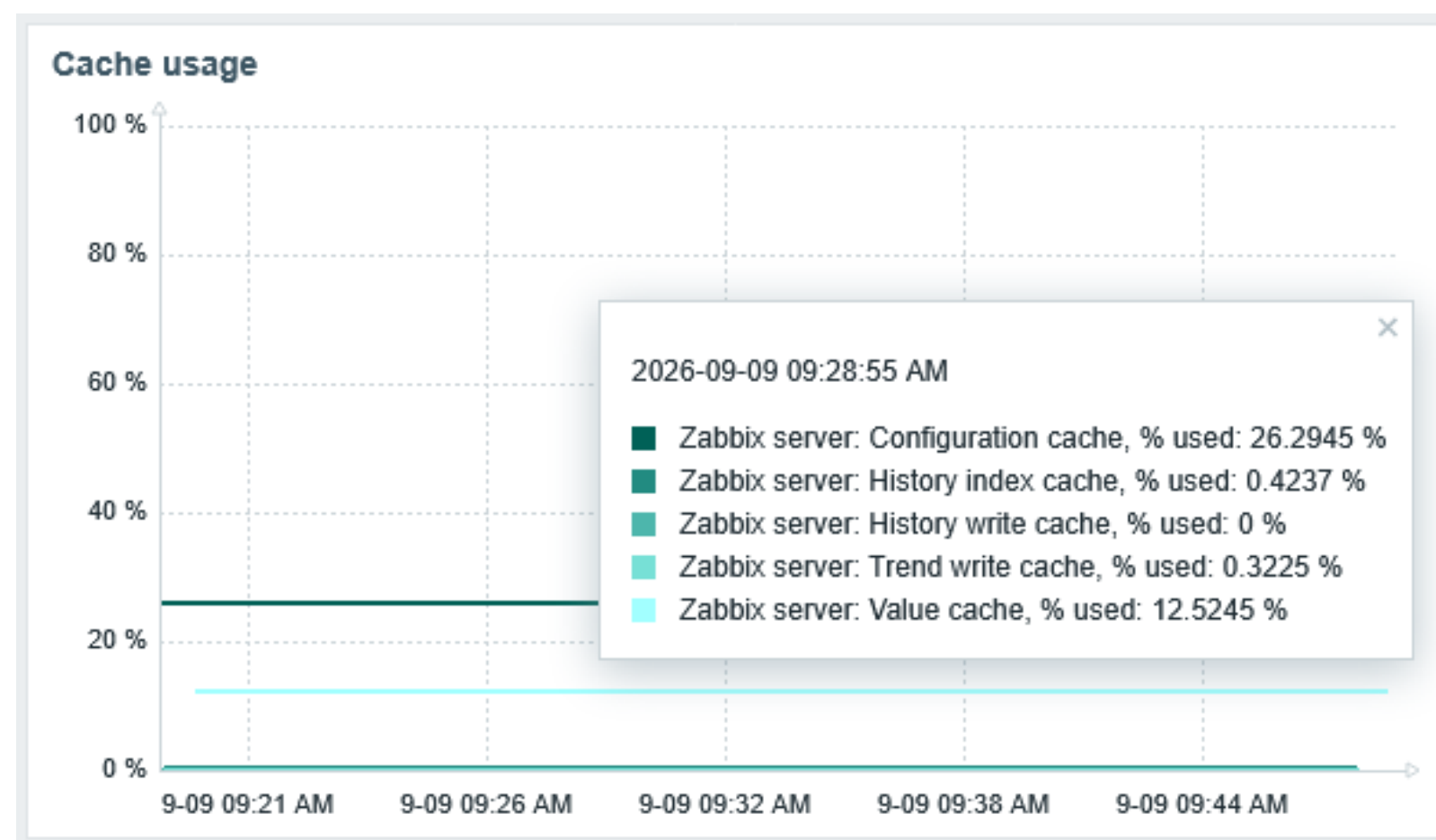
Top 25 items

Low memory mode

- Items that weren't accessed during the last day are removed from cache.
- Items with worst hits/values ratio might be removed from cache to free the space.
- No new items are added to the cache.



Low memory mode with low usage



Value cache looks free

```
# zabbix_server -R diaginfo=valuecache
== value cache diagnostic information ==
Items:0 values:0 mode:1 time:0.000007
Memory:
  size: free:114320 used:16320
  chunks: free:1 used:3 min:114320 max:114320
  buckets:
    256+:1
Top.values:
Top.request.values:
==
```

Only 16 KB used

History cache



diaginfo=historycache

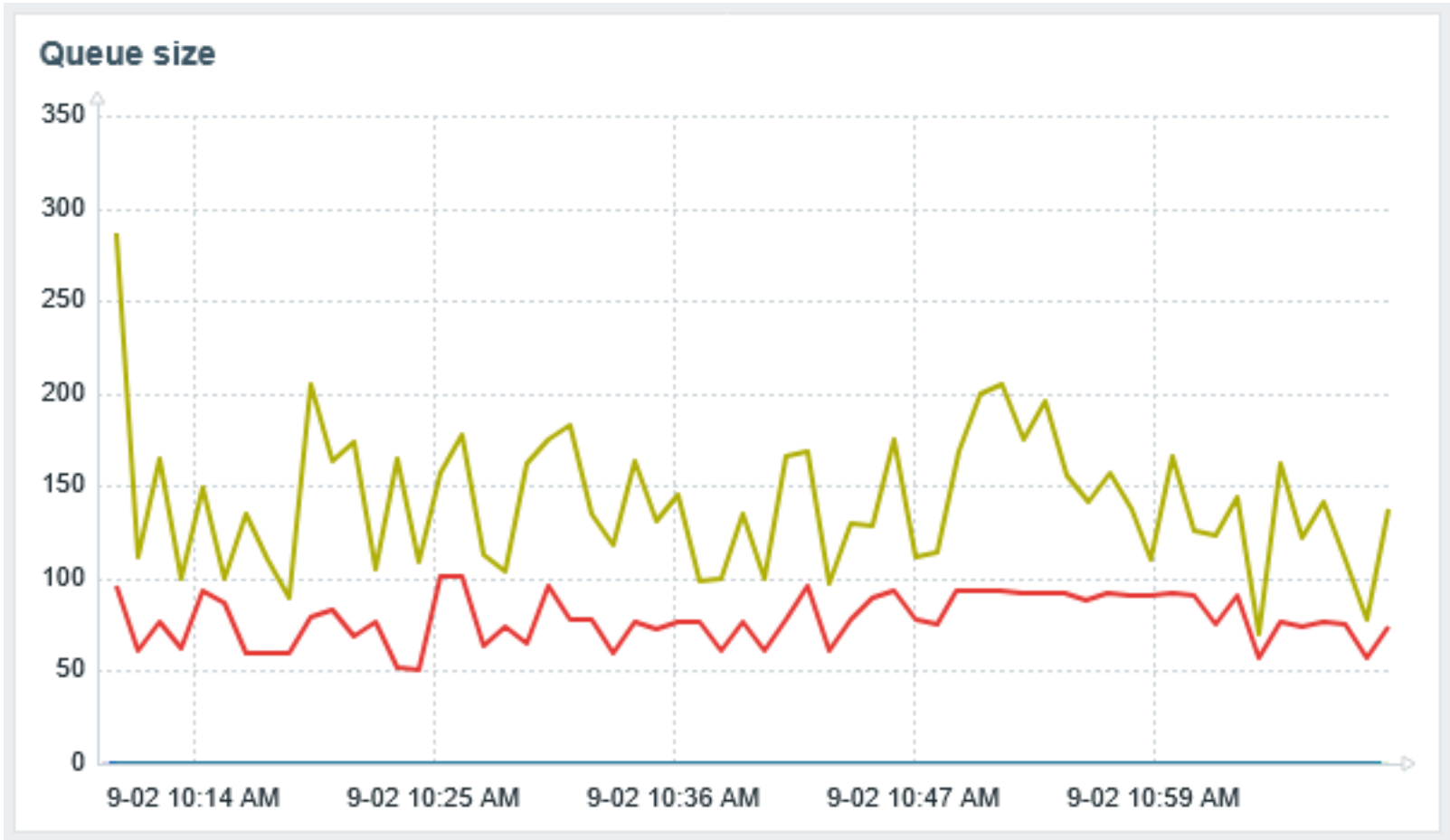
```
zabbix_server -R diaginfo=historycache
== history cache diagnostic information ==
Items:1 values:1 time:0.000004
Memory.data:
  size: free:16776712 used:88
  chunks: free:1 used:2 min:16776712 max:16776712
  buckets:
    256+:1
Memory.index:
  size: free:4176136 used:17672
  chunks: free:1 used:6 min:4176136 max:4176136
  buckets:
    256+:1
Top.values:
  itemid:44789 values:1
==
```

Zabbix Server Health

Queue

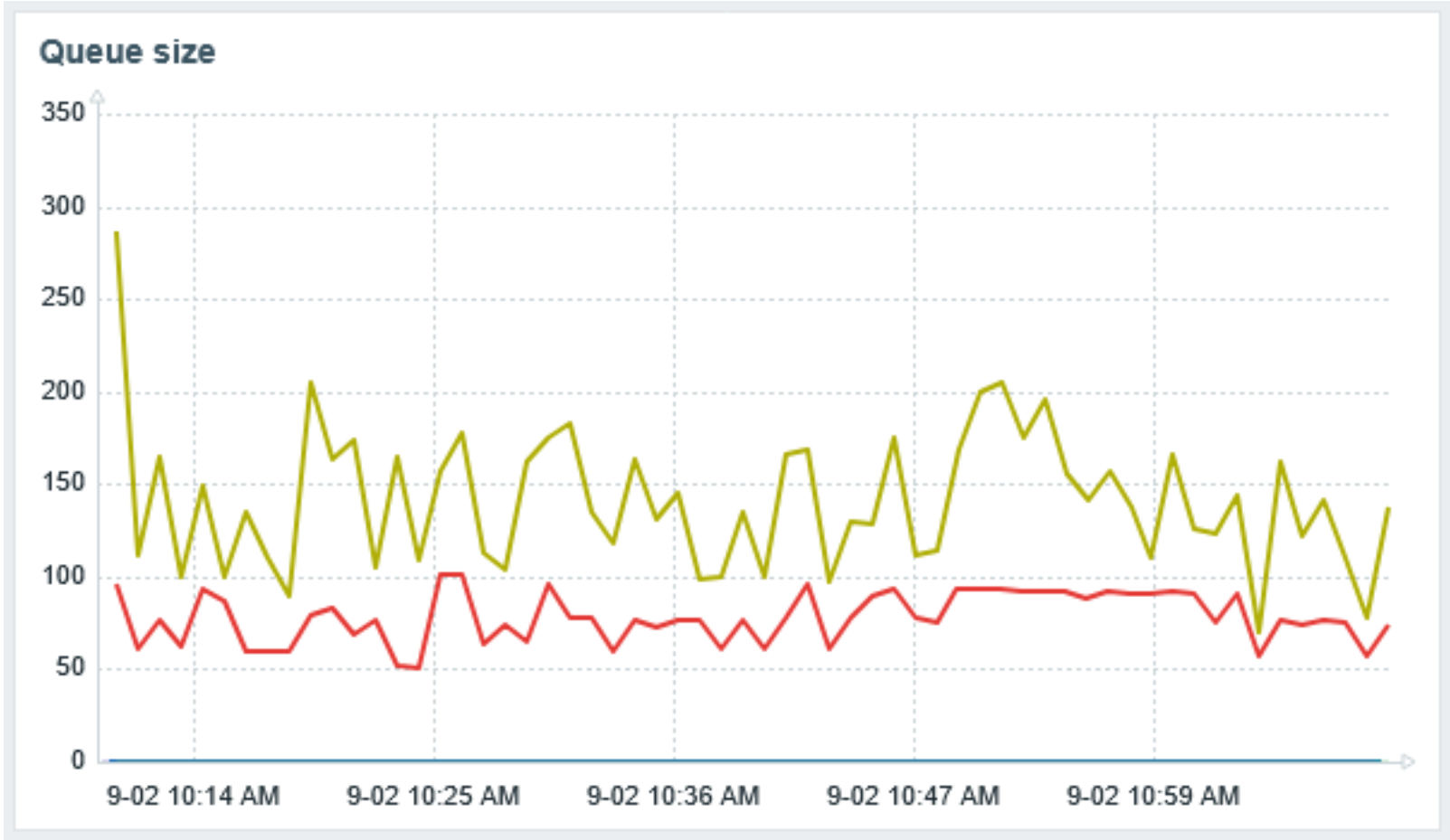


Items in queues



Various Queues

Items in queues



Various Queues

Items	5 seconds	10 seconds	30 seconds	1 minute	5 minutes	More than 10 minutes
Zabbix agent	0	14	8	11	1	87
Zabbix agent (active)	0	0	0	0	0	1
Simple check	0	0	0	0	0	0
SNMP agent	0	16	0	0	0	0
Zabbix internal	0	0	0	0	0	0
External check	0	0	0	0	0	0
Database monitor	0	0	0	0	0	0
HTTP agent	0	2	0	0	0	0
IPMI agent	0	0	0	0	0	0
SSH agent	0	0	0	0	0	0
TELNET agent	0	0	0	0	0	0
JMX agent	0	0	0	0	0	0
Calculated	0	0	0	0	0	0
Script	0	0	0	0	0	0
Browser	0	0	0	0	0	0

Queue overview

```
static void      queue_stats_update (zbx_queue_stats_t *stats, int delay)
{
    if (10 >= delay)
        stats->delay5++;
    else if (30 >= delay)
        stats->delay10++;
    else if (60 >= delay)
        stats->delay30++;
    else if (300 >= delay)
        stats->delay60++;
    else if (600 >= delay)
        stats->delay300++;
    else
        stats->delay600++;
}
```

Only interval (delay) is considered...

```

int      zbx_is_counted_in_item_queue(unsigned char type, const char *key)
{
    switch (type)
    {
        case ITEM_TYPE_ZABBIX_ACTIVE:
            if (0 == strncmp(key, "log[", 4) ||
                0 == strncmp(key, "logrt[", 6) ||
                0 == strncmp(key, "eventlog[", 9) ||
                0 == strncmp(key, "mqtt.get[", ZBX_CONST_STRLEN("mqtt.get[")))
            {
                return FAIL;
            }
            break;
        case ITEM_TYPE_TRAPPER:
        case ITEM_TYPE_DEPENDENT:
        case ITEM_TYPE_HTTPTEST:
        case ITEM_TYPE_SNMPTRAP:
            return FAIL;
    }

    return SUCCEED;
}

```

...with exceptions.

Queue

Details

Scheduled check	Delayed by	Host	Name	Proxy
2026-06-04 12:04:20 AM	3M 5d 15h	MSSQL ONE	SQL: Databases\Log File(s) Size (KB)	proxy1
2026-06-04 12:04:29 AM	3M 5d 15h	MSSQL ONE	SQL: Locks(Application)\Lock Requests/sec	proxy1
2026-06-04 12:04:30 AM	3M 5d 15h	MSSQL ONE	SQL: Locks(Application)\Lock Timeouts/sec	proxy1
2026-06-04 12:04:31 AM	3M 5d 15h	MSSQL ONE	SQL: Locks(Application)\Lock Waits/sec	proxy1
2026-06-04 12:04:32 AM	3M 5d 15h	MSSQL ONE	SQL: Locks(Application)\Lock Wait Time (ms)	proxy1
2026-06-04 12:04:33 AM	3M 5d 15h	MSSQL ONE	SQL: Locks(Application)\Number of Deadlocks/sec	proxy1
2026-06-04 12:04:34 AM	3M 5d 15h	MSSQL ONE	SQL: Locks(Object)\Lock Requests/sec	proxy1
2026-06-04 12:04:35 AM	3M 5d 15h	MSSQL ONE	SQL: Locks(Object)\Lock Timeouts/sec	proxy1
2026-06-04 12:04:36 AM	3M 5d 15h	MSSQL ONE	SQL: Locks(Object)\Lock Waits/sec	proxy1
2026-06-04 12:04:42 AM	3M 5d 15h	MSSQL ONE	SQL: Locks(_Total)\Lock Wait Time (ms)	proxy1
2026-06-04 12:04:37 AM	3M 5d 15h	MSSQL ONE	SQL: Locks(Object)\Lock Wait Time (ms)	proxy1
2026-06-04 12:04:38 AM	3M 5d 15h	MSSQL ONE	SQL: Locks(Object)\Number of Deadlocks/sec	proxy1
2026-06-04 12:04:39 AM	3M 5d 15h	MSSQL ONE	SQL: Locks(_Total)\Lock Requests/sec	proxy1
2026-06-04 12:04:41 AM	3M 5d 15h	MSSQL ONE	SQL: Locks(_Total)\Lock Waits/sec	proxy1
2026-06-04 12:04:43 AM	3M 5d 15h	MSSQL ONE	SQL: Locks(_Total)\Number of Deadlocks/sec	proxy1
2026-06-04 12:09:03 AM	3M 5d 15h	MSSQL TWO	SQL: Databases\Log File(s) Size (KB)	proxy1
2026-06-04 12:09:12 AM	3M 5d 15h	MSSQL TWO	SQL: Locks(Application)\Lock Requests/sec	proxy1
2026-06-04 12:09:13 AM	3M 5d 15h	MSSQL TWO	SQL: Locks(Application)\Lock Timeouts/sec	proxy1
2026-06-04 12:04:40 AM	3M 5d 15h	MSSQL ONE	SQL: Locks(_Total)\Lock Timeouts/sec	proxy1
2026-06-04 12:09:14 AM	3M 5d 15h	MSSQL TWO	SQL: Locks(Application)\Lock Waits/sec	proxy1
2026-06-04 12:09:15 AM	3M 5d 15h	MSSQL TWO	SQL: Locks(Application)\Lock Wait Time (ms)	proxy1
				Displaying 21 of 21 found

```
$ QUEUE_DETAILS='{ "request": "queue.get", "type": "details", "limit": "2", "sid": "XXX" }'  
$ zabbix_get -s 127.0.0.1 -k "$QUEUE_DETAILS" -p 10051 --protocol plaintext | jq  
{  
  "response": "success",  
  "data": [  
    {  
      "itemid": 51198,  
      "nextcheck": 1788864609  
    },  
    {  
      "itemid": 51122,  
      "nextcheck": 1788864650  
    }  
  ],  
  "total": 253  
}
```

Queue is queried over TCP, so...

Impersonating other components

- PING='{"request": "status.get", "type": "ping", "sid": "XXX"}'
- STATUS='{"request": "status.get", "type": "full", "sid": "XXX"}'
- QUEUE_DETAILS='{"request": "queue.get", "type": "details", "limit": "2", "sid": "XXX"}'
- QUEUE_OVERVIEW='{"request": "queue.get", "type": "overview", "sid": "XXX"}'
- QUEUE_PROXY='{"request": "queue.get", "type": "overview by proxy", "sid": "XXX"}'
- ACTIVE_CHECKS='{"request": "active checks", "host": "Zabbix server"}'



Profiling

```
zabbix_server -R prof_enable
```



```
=== Profiling statistics for trapper ===  
commands_lock() mutex : locked:1 holding:0.000000 sec waiting:0.000000 sec  
lock_log() mutex : locked:13 holding:0.000149 sec waiting:0.000011 sec  
zbx_dc_get_or_create_session() rwlock : locked:1 holding:0.000001 sec waiting:0.000000 sec  
zbx_dc_get_auto_registration_action_count() rwlock : locked:1 holding:0.000000 sec waiting:0.000001 sec  
zbx_dc_get_expressions_by_names() rwlock : locked:1 holding:0.000000 sec waiting:0.000000 sec  
zbx_dc_check_host_conn_permissions() rwlock : locked:1 holding:0.000005 sec waiting:0.000021 sec  
sm_sync_lock() mutex : locked:12 holding:0.000002 sec waiting:0.000003 sec  
rwlocks : locked:4 holding:0.000007 sec waiting:0.000021 sec  
mutexes : locked:26 holding:0.000151 sec waiting:0.000014 sec  
locking total : locked:30 holding:0.000158 sec waiting:0.000035 sec
```

Before stress

```
=== Profiling statistics for trapper ===
commands_lock() mutex : locked:7 holding:0.000005 sec waiting:0.000002 sec
lock_log() mutex : locked:125 holding:0.001221 sec waiting:0.000360 sec
dc_config_get_host_by_name() rwlock : locked:218 holding:0.000171 sec waiting:0.000071 sec
zbx_dc_items_update_nextcheck() rwlock : locked:218 holding:0.000160 sec waiting:0.000026 sec
zbx_dc_config_history_recv_get_items_by_keys() rwlock : locked:218 holding:0.014068 sec waiting:0.000031 sec
zbx_dc_get_or_create_session() rwlock : locked:7 holding:0.000014 sec waiting:0.000000 sec
zbx_dc_get_auto_registration_action_count() rwlock : locked:7 holding:0.000008 sec waiting:0.000003 sec
zbx_dc_get_expressions_by_names() rwlock : locked:7 holding:0.000001 sec waiting:0.000000 sec
zbx_dc_check_host_conn_permissions() rwlock : locked:7 holding:0.000523 sec waiting:0.000001 sec
hc_add_item_values() mutex : locked:13 holding:0.000331 sec waiting:0.000008 sec
zbx_dc_flush_history() mutex : locked:436 holding:0.007749 sec waiting:0.000047 sec
sm_sync_lock() mutex : locked:132 holding:0.000027 sec waiting:0.000024 sec
rwlocks : locked:682 holding:0.014943 sec waiting:0.000132 sec
mutexes : locked:713 holding:0.009332 sec waiting:0.000441 sec
locking total : locked:1395 holding:0.024275 sec waiting:0.000573 sec
```

Whoops...

```

=== Profiling statistics for trapper ===
commands_lock() mutex : locked:7 holding:0.000005 sec wait
lock_log() mutex : locked:125 holding:0.001221 sec waiting
dc_config_get_host_by_name() rwlock : locked:218 holding:
zbx_dc_items_update_nextcheck() rwlock : locked:218 holding:
zbx_dc_config_history_recv_get_items_by_keys() rwlock :
zbx_dc_get_or_create_session() rwlock : locked:7 holding:
zbx_dc_get_auto_registration_action_count() rwlock : locked:
zbx_dc_get_expressions_by_names() rwlock : locked:7 holding:
zbx_dc_check_host_conn_permissions() rwlock : locked:7 holding:
hc_add_item_values() mutex : locked:13 holding:0.000331 sec wait
zbx_dc_flush_history() mutex : locked:436 holding:0.00774 sec wait
sm_sync_lock() mutex : locked:132 holding:0.000027 sec wait
rwlocks : locked:682 holding:0.014943 sec waiting:0.00013
mutexes : locked:713 holding:0.009332 sec waiting:0.00044
locking total : locked:1395 holding:0.024275 sec waiting:

```

```

size_t zbx_dc_flush_history(void)
{
    LOCK_CACHE;

    if (0 == item_values_num)
    {
        UNLOCK_CACHE;
        return 0;
    }

    hc_add_item_values(item_values, item_values_num);

    cache->history_num += item_values_num;

    int      processing_num = cache->processing_num;
    size_t   count = item_values_num;

    item_values_num = 0;
    string_values_offset = 0;

    UNLOCK_CACHE;

    zbx_vps_monitor_add_collected((zbx_uint64_t)count);

    if (0 != processing_num)
        return 0;

    return count;
}

```

Whoops...

```

static void    hc_add_item_values(dc_item_value_t *values, int values_num)
{
    dc_item_value_t    *item_value;
    int                i;
    zbx_hc_item_t      *item;

    for (i = 0; i < values_num; i++)
    {
        (...)
        if (SUCCEED != hc_clone_history_data(&data, item_value))
        {
            do
            {
                UNLOCK_CACHE;

                zabbix_log(LOG_LEVEL_DEBUG, "History cache is full. Sleeping for 1 second.");
                sleep(1);

                LOCK_CACHE;
            }
            while (SUCCEED != hc_clone_history_data(&data, item_value));

            item = hc_get_item(item_value->itemid);

            (...)
            item->values_num++;
        }
    }
}

```

=== Profile
 commands_lo
 lock_log()
 dc_config_
 zbx_dc_iter
 zbx_dc_conf
 zbx_dc_get
 zbx_dc_get
 zbx_dc_get
 zbx_dc_che
 hc_add_iter
 zbx_dc_flush
 sm_sync_lo
 rwlocks :
 mutexes :
 locking to

values_num);

im;

processing_num;

uint64_t) count);

Whoops...

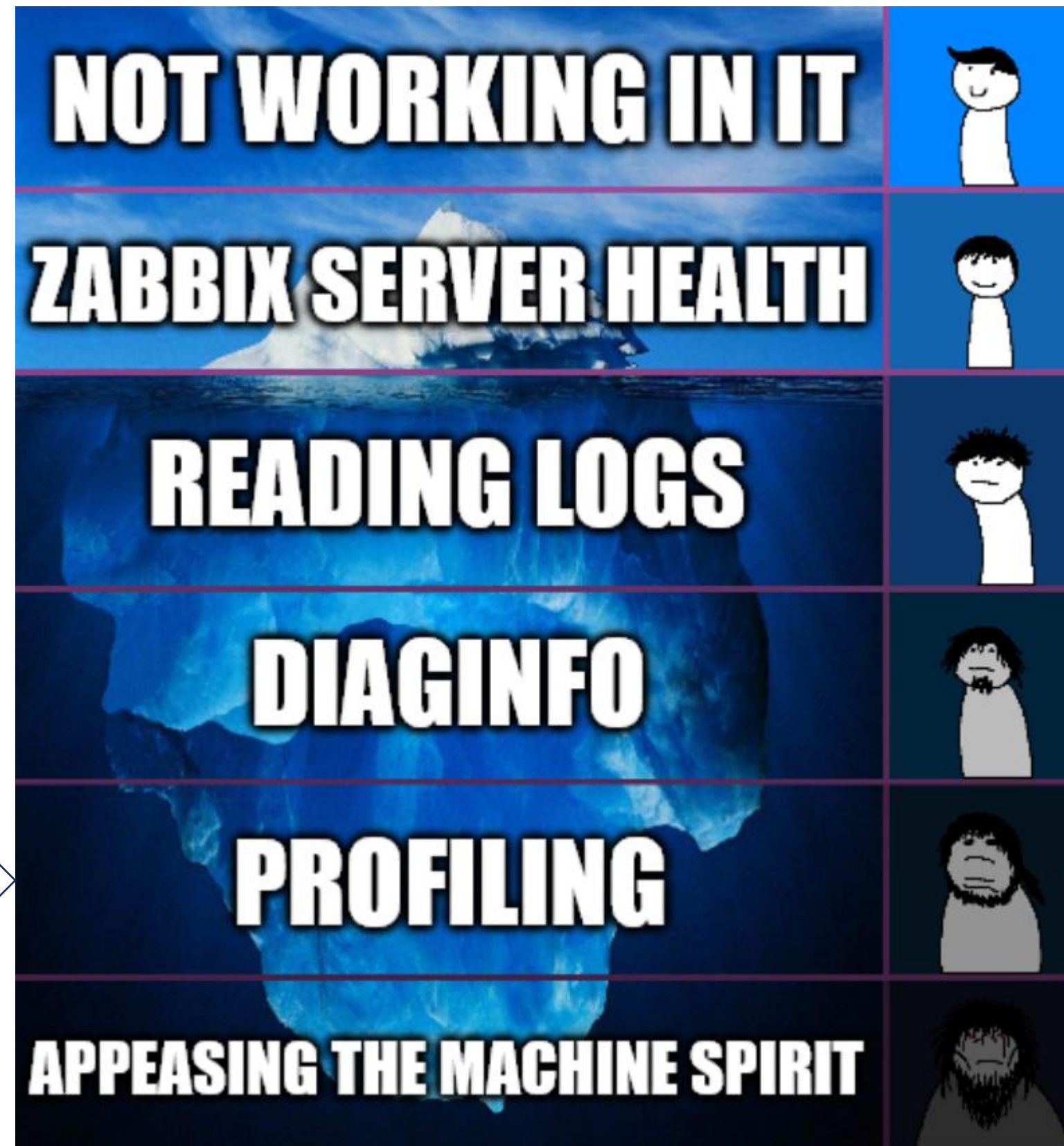
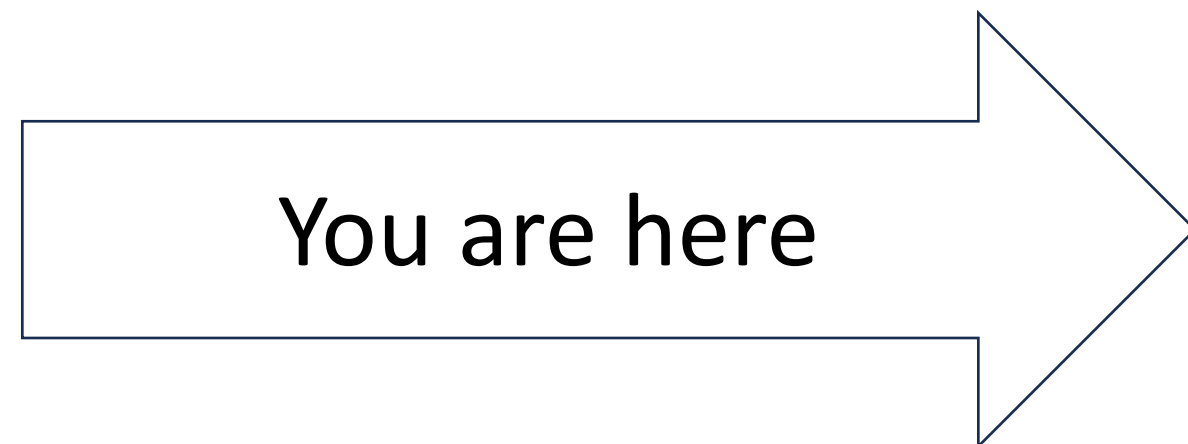
```
# zabbix_server -R diaginfo=historycache
== history cache diagnostic information ==
Items:1 values:105425 time:0.000018
Memory.data:
    size: free:6656016 used:6747200
    chunks: free:2 used:210850 min:15344 max:6640672
    buckets:
        256+:2
Memory.index:
    size: free:4176136 used:17672
    chunks: free:1 used:6 min:4176136 max:4176136
    buckets:
        256+:1
Top.values:
    itemid:51200 values:105425
==
```

```
values_num);
```

```
ssing_num;
```

```
nt64_t) count);
```

Whoops...



Our journey...



Zabbix Certified Trainer

Robert Szulist

APL!TT sp. z o.o.

