

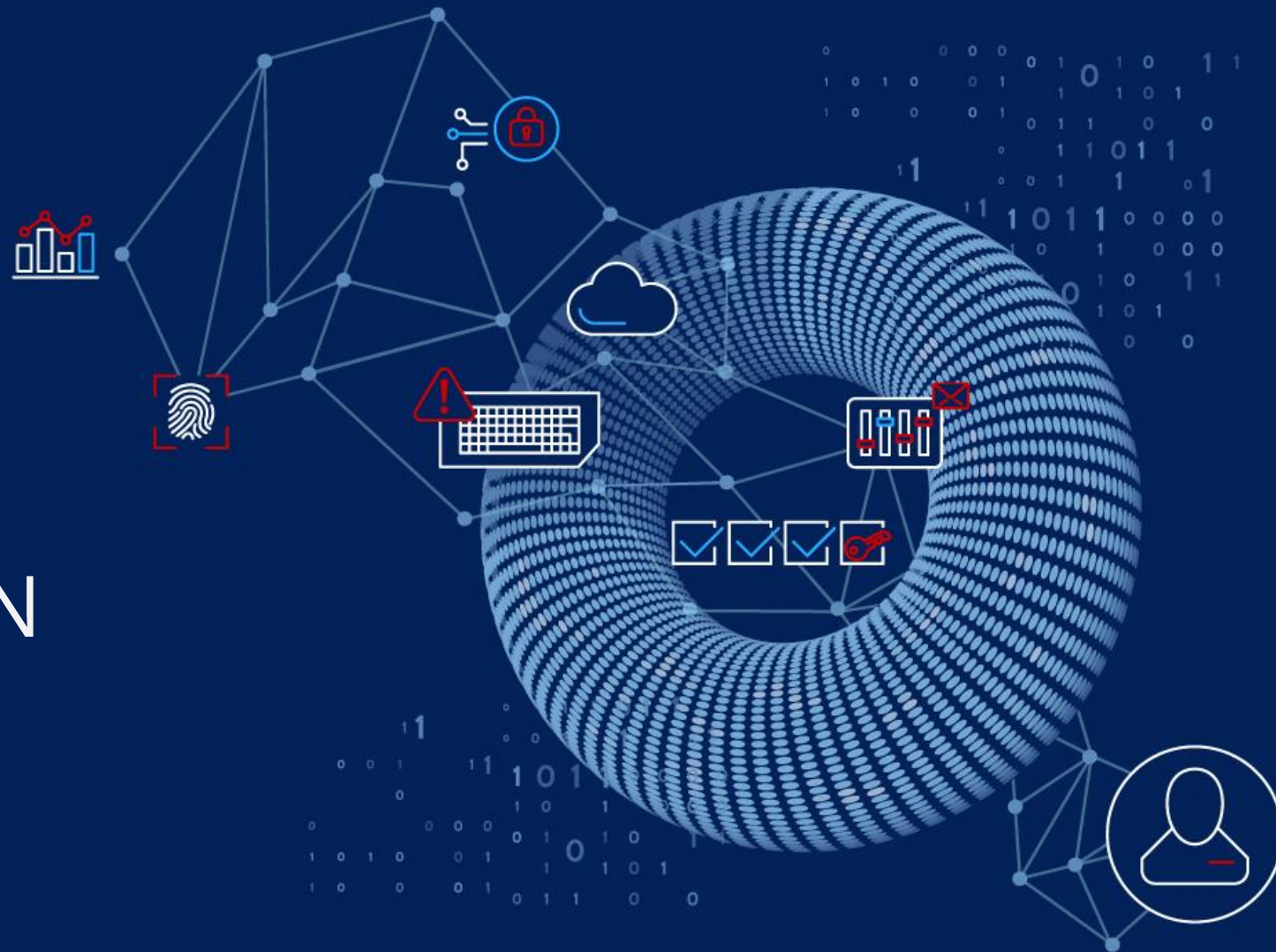
ZABBIX

6.4

SNMP BULK DATA COLLECTION

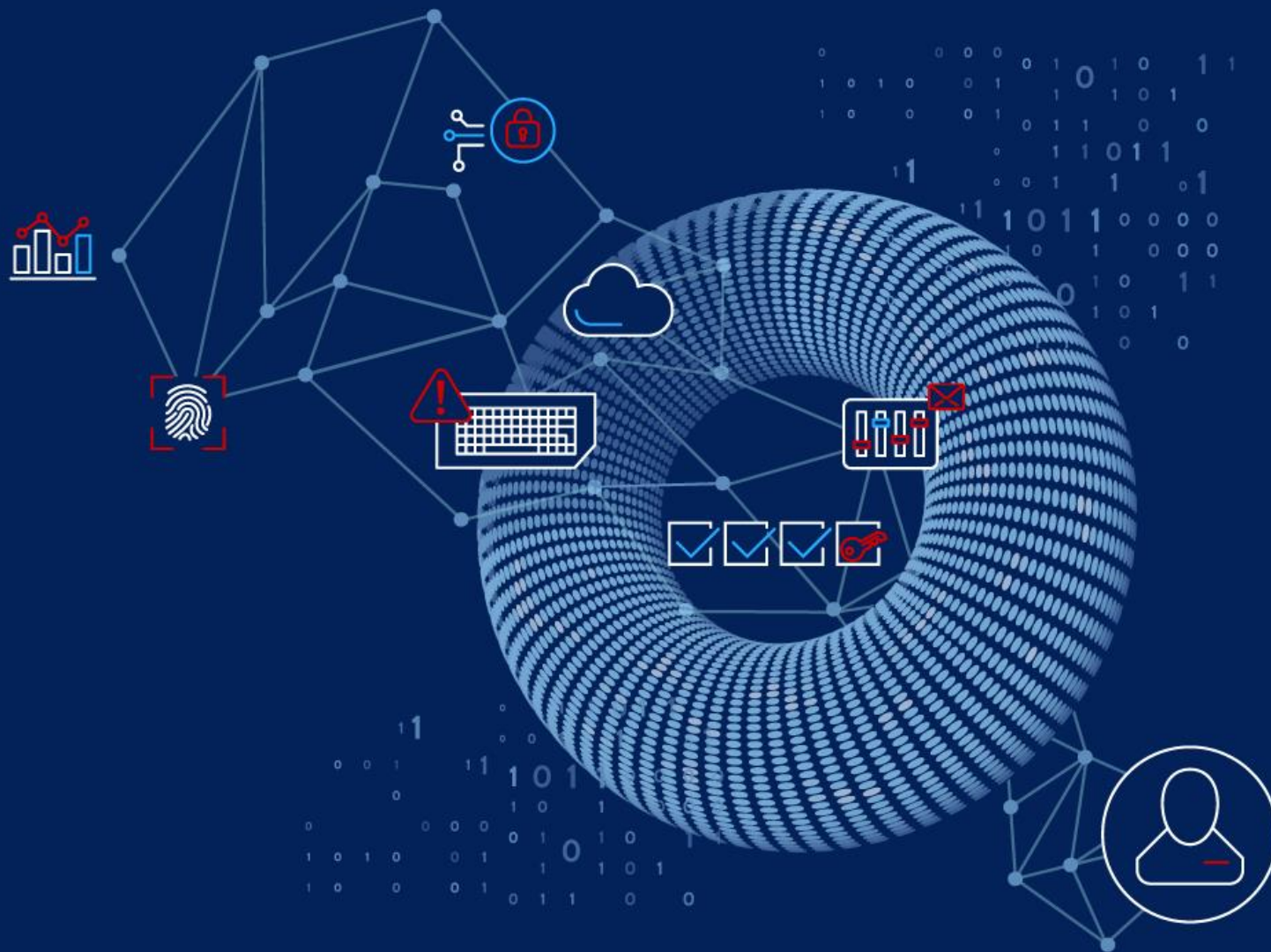
Kaspars Mednis

Chief trainer



01

SNMP OVERVIEW



Is SNMP still an important tool for monitoring in 2023?

- ⚡ SNMP, as an Internet standard, is over 30 years old
- ⚡ The first version SNMPv1 was published in 1988
- ⚡ SNMP has evolved over the years, and several versions of the protocol have been released, including SNMPv2c and SNMPv3

SNMP provides a **comprehensive** and **standardized** way to monitor:

- ⚡ A standardized protocol that is widely adopted in many network environments
- ⚡ Supported by a wide range of network equipment and operating systems
- ⚡ Many IoT devices can be monitored using SNMP
- ⚡ A cost-effective solution for network monitoring and management
- ⚡ SNMPv3 supports authentication and encryption of SNMP traffic

A quick overview of SNMP versions:

⚡ Version 1

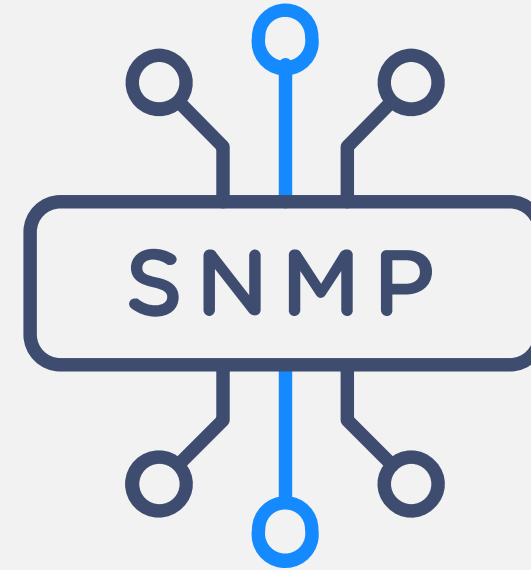
- ✓ plain-text community string
- ✓ only 32-bit counters supported

⚡ Version 2c

- ✓ plain-text community
- ✓ adds support for 64-bit counters
- ✓ introduces GETBULK command

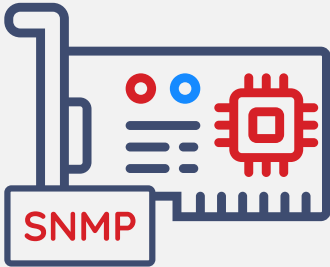
⚡ Version 3

- ✓ adds authentication
- ✓ adds encryption
- ✓ improved error reporting and reliability
- ✓ adds multiple SNMP contexts



Zabbix supports all three SNMP versions out of the box:

- 📶 Implemented using the SNMP agent interface
- 📶 The parameters depend on the SNMP version being used



Interfaces

Type	IP address	DNS name	Connect to	Port	Default
SNMP		device.example.com	IP DNS	161	<input checked="" type="radio"/> Remove

* SNMP version SNMPv1

Interfaces

Type	IP address	DNS name	Connect to	Port	Default
SNMP		device.example.com	IP DNS	161	<input checked="" type="radio"/> Remove

* SNMP version SNMPv2

Interfaces

Type	IP address	DNS name	Connect to	Port	Default
SNMP		device.example.com	IP DNS	161	<input checked="" type="radio"/> Remove

* SNMP version SNMPv3

Max repetition count ? 10

Context name

Security name zabbix

Security level authPriv

Authentication protocol SHA512

Authentication passphrase { \$AUTHENTICATION.PASSPHRADE }

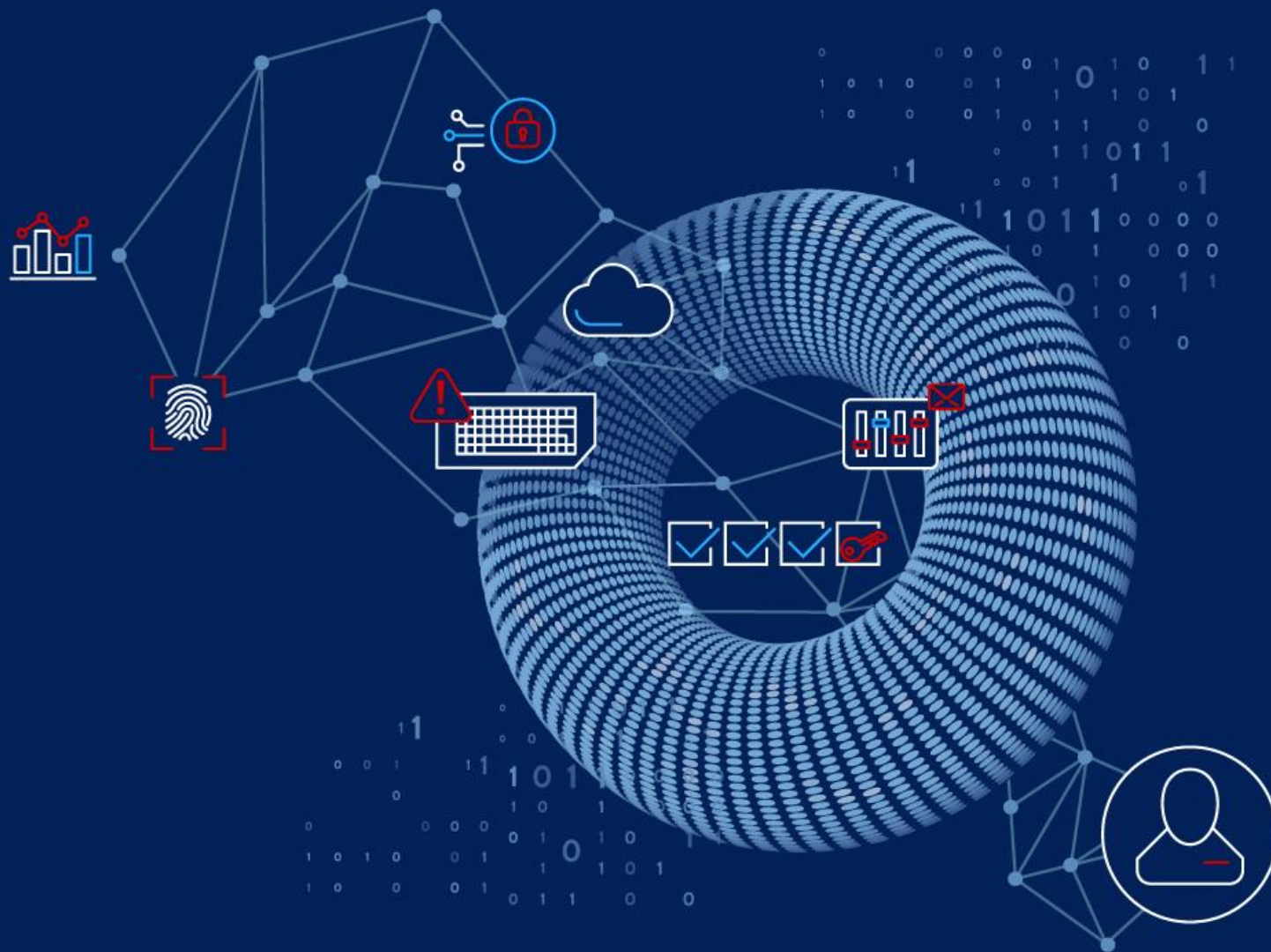
Privacy protocol AES256

Privacy passphrase { \$PRIVACY.PASSPHRASE }

☒ Use combined requests

02

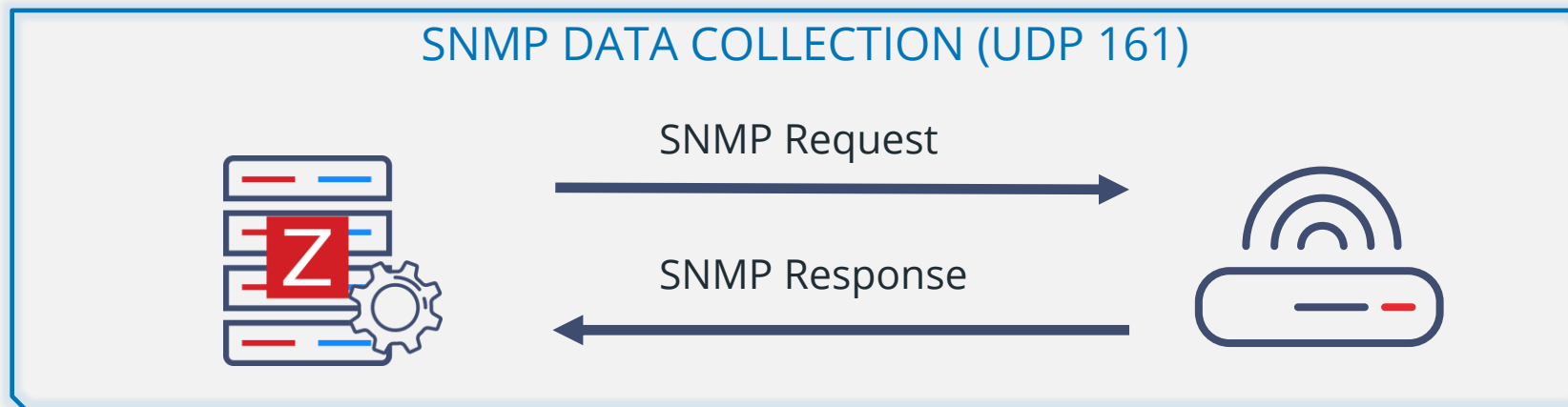
GATHERING SNMP DATA



Modern SNMP agents support multiple commands:

- ⚡ GET a request to collect a **single or multiple values** from the device
- ⚡ GETNEXT a command to retrieve the **next object** in a sequence of objects
- ⚡ GETBULK a command to pull data tables using **lots of GETNEXT** commands

The device sends back **RESPONSE** or **ERROR** message



```
IP 10.10.10.1.53436 > 20.20.20.2.161: C="zabbix" GetRequest(29) .1.3.6.1.2.1.1.3.0
IP 20.20.20.2.161 > 10.10.10.1.53436: C="zabbix" GetResponse(33) .1.3.6.1.2.1.1.3.0=234995968
```

Simple SNMP GET command collects a single metric

```
GetRequest(29) .1.3.6.1.2.1.25.1.1.0
GetResponse(33) .1.3.6.1.2.1.25.1.1.0=260489736
```

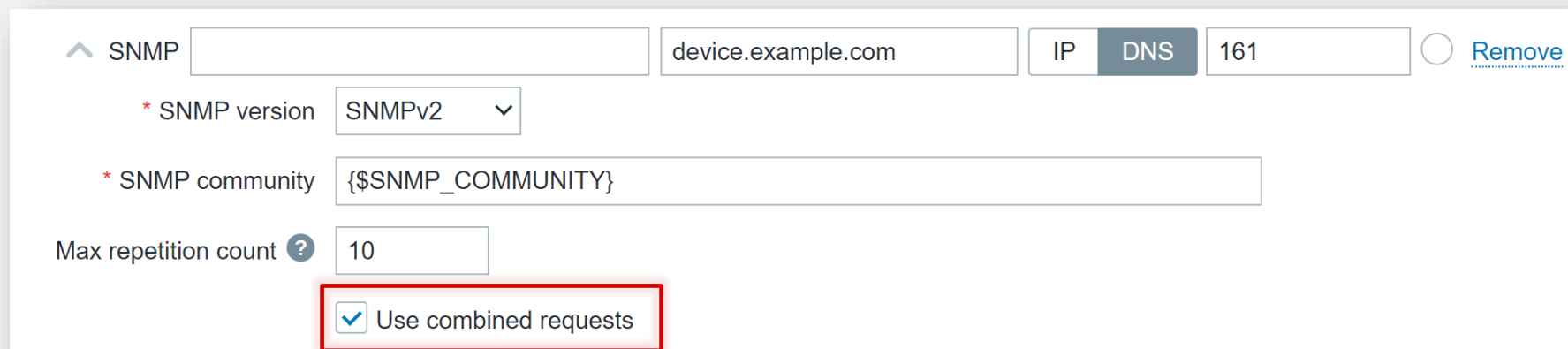
It is possible to collect multiple metrics at once with a single GET request

```
GetRequest(75) .1.3.6.1.2.1.25.1.1.0 1.3.6.1.4.1.2021.4.3.0 .1.3.6.1.2.1.1.3.0
GetResponse(89) .1.3.6.1.2.1.25.1.1.0=260481729 1.3.6.1.4.1.2021.4.3.0=4194300
                  .1.3.6.1.2.1.1.3.0=260480378
```

In Zabbix these requests are named "combined requests"

⚡ In previous versions they were named "bulk requests"

⚡ Not all SNMP devices support combined requests



The screenshot shows the Zabbix configuration interface for an SNMP item. At the top, there is a header bar with 'SNMP' on the left, a text input field containing 'device.example.com', and buttons for 'IP', 'DNS', and '161'. A 'Remove' button is on the right. Below this, there are configuration fields: 'SNMP version' set to 'SNMPv2', 'SNMP community' set to '{\$SNMP_COMMUNITY}', and 'Max repetition count' set to '10'. At the bottom, there is a checkbox labeled 'Use combined requests' which is checked and highlighted with a red border.

SNMP

* SNMP version

* SNMP community

Max repetition count

☒ Use combined requests

Where should "combined requests" be used?

- ⚡ This approach works well for **scalar metrics** (OIDs that ends with .0)
- ⚡ OIDs from different MIB trees can be collected in a single request
- ⚡ This results in improved performance due to fewer SNMP requests

<input type="checkbox"/>	...	System contact details		system.contact[sysContact.0]	15m	2w		SNMP agent	Enabled	component: system
<input type="checkbox"/>	...	System description		system.descr[sysDescr.0]	15m	2w		SNMP agent	Enabled	component: system
<input type="checkbox"/>	...	System location		system.location[sysLocation.0]	15m	2w		SNMP agent	Enabled	component: system
<input type="checkbox"/>	...	System name	Triggers 1	system.name[sysName.0]	15m	2w		SNMP agent	Enabled	component: system
<input type="checkbox"/>	...	System object ID		system.objectid[sysObjectID.0]	15m	2w		SNMP agent	Enabled	component: system
<input type="checkbox"/>	...	Uptime (hardware)	Triggers 1	system.hw.uptime[hrSystemUptime.0]	30s	2w	0d	SNMP agent	Enabled	component: system
<input type="checkbox"/>	...	Uptime (network)	Triggers 1	system.net.uptime[sysUpTime.0]	30s	2w	0d	SNMP agent	Enabled	component: system

Item Tags 1 Preprocessing 1

* Name

System location

Type

SNMP agent

* Key

system.location[sysLocation.0]

Type of information

Character

* SNMP OID

1.3.6.1.2.1.1.6.0

Item Tags 1 Preprocessing 2

* Name

Uptime (hardware)

Type

SNMP agent

* Key

system.hw.uptime[hrSystemUptime.0]

Select

Type of information

Numeric (unsigned)

* SNMP OID

1.3.6.1.2.1.25.1.1.0

For **tabular metrics** the GETBULK approach is preferred:

- ⚡ Entire table can be read row by row using GETBULK requests
- ⚡ This is faster operation for the SNMP agent

Get incoming traffic (ifInOctets) for all network interfaces

```
snmpbulkget -v2c -czabbix -On device.example.com .1.3.6.1.2.1.2.2.1.10
```

```
GetBulk(29)  N=0 M=10  .1.3.6.1.2.1.2.2.1.10
```

```
GetResponse(207)  .1.3.6.1.2.1.2.2.1.10.1=523436839  
                  .1.3.6.1.2.1.2.2.1.10.2=19466  
                  .1.3.6.1.2.1.2.2.1.10.3=240426935  
                  .1.3.6.1.2.1.2.2.1.10.4=13933636  
                  .1.3.6.1.2.1.2.2.1.10.5=24492  
                  .1.3.6.1.2.1.2.2.1.10.6=23271802  
                  .1.3.6.1.2.1.2.2.1.10.7=1022187  
                  .1.3.6.1.2.1.2.2.1.10.8=15378997
```

SNMP GETBULK command utilizes two additional parameters:

- ⚡ Non-repetition is the number of OIDs to be retrieved a single time without repetition (0 by default)
- ⚡ Max-repetition is the number of values retrieved in one request (10 by default)

The max repetition count is specified in the SNMP interface settings

Interfaces	Type	IP address	DNS name	Connect to	Port	Default
^ SNMP			device.example.com	IP DNS	161	<input checked="" type="radio"/> Remove
	* SNMP version	SNMPv2 ▼				
	* SNMP community	{\${SNMP.COMMUNITY}}				
	Max repetition count ?	<input type="text" value="10"/>				
	<input checked="" type="checkbox"/> Use combined requests					

The default value of **10** for max repetitions works well with most devices:

⚡ If there are more values left, Zabbix repeats the SNMPBULKGET request

```
snmpbulkget -v2c -cpublic -On -Cr10 -Cn0 training.lan .1.3.6.1.2.1.2.2.1.10
```

```
.1.3.6.1.2.1.2.2.1.10.1 = Counter32: 2360  
.1.3.6.1.2.1.2.2.1.10.2 = Counter32: 524454878  
.1.3.6.1.2.1.2.2.1.10.3 = Counter32: 19466  
.1.3.6.1.2.1.2.2.1.10.4 = Counter32: 241042778  
.1.3.6.1.2.1.2.2.1.10.5 = Counter32: 13964510  
.1.3.6.1.2.1.2.2.1.10.6 = Counter32: 24492  
.1.3.6.1.2.1.2.2.1.10.7 = Counter32: 23323498  
.1.3.6.1.2.1.2.2.1.10.8 = Counter32: 1022257  
.1.3.6.1.2.1.2.2.1.10.9 = Counter32: 15412552  
.1.3.6.1.2.1.2.2.1.10.10 = Counter32: 475059
```

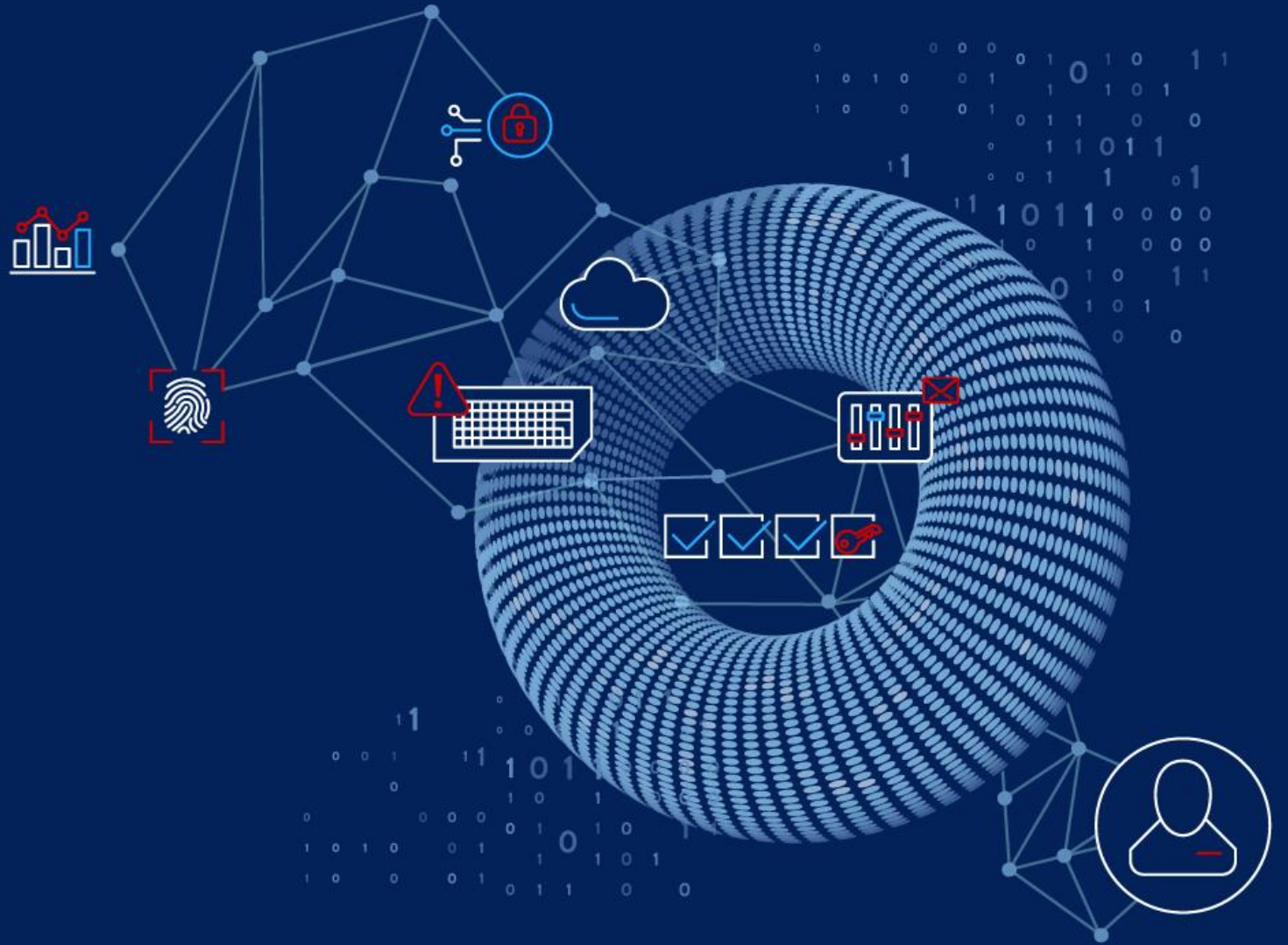
⚡ BULKGET continues from the index of 10 (the max-repetition count)

```
snmpbulkget -v2c -cpublic -On -Cr10 -Cn0 training.lan .1.3.6.1.2.1.2.2.1.10.10
```

```
.1.3.6.1.2.1.2.2.1.10.11 = Counter32: 2360  
.1.3.6.1.2.1.2.2.1.10.12 = Counter32: 524454878  
.1.3.6.1.2.1.2.2.1.10.13 = Counter32: 19466  
.1.3.6.1.2.1.2.2.1.10.14 = Counter32: 241042778  
.1.3.6.1.2.1.2.2.1.10.15 = Counter32: 13964510  
.1.3.6.1.2.1.2.2.1.10.16 = Counter32: 24492
```

03

SNMP WALK[*] ITEM



Zabbix introduced a new SNMP agent item `walk[*]` in version 6.4

- ⚡ This item is used both for SNMP **discovery** and **data collection**
- ⚡ Item uses SNMP BULKGET commands for data collection
- ⚡ A single or multiple OIDs can be specified at once

Item Tags 1 Preprocessing

* Name

SNMP walk network interfaces

Type

SNMP agent

* Key

net.if.walk

Select

Type of information

Text

* Host interface

device.example.com:161

* SNMP OID

walk[1.3.6.1.2.1.31.1.1.1.1,1.3.6.1.2.1.31.1.1.1.6,1.3.6.1.2.1.31.1.1.1.10]

* Update interval

1m

×

```
.1.3.6.1.2.1.31.1.1.1.1 = STRING: lo
.1.3.6.1.2.1.31.1.1.1.2 = STRING: eth0
.1.3.6.1.2.1.31.1.1.1.3 = STRING: eth1
.1.3.6.1.2.1.31.1.1.1.4 = STRING: podman1
.1.3.6.1.2.1.31.1.1.1.10.1 = Counter64: 2360
.1.3.6.1.2.1.31.1.1.1.10.2 = Counter64: 5875130
.1.3.6.1.2.1.31.1.1.1.10.3 = Counter64: 20032
.1.3.6.1.2.1.31.1.1.1.10.4 = Counter64: 274374233
.1.3.6.1.2.1.31.1.1.1.6.1 = Counter64: 2360
.1.3.6.1.2.1.31.1.1.1.6.2 = Counter64: 536085556
.1.3.6.1.2.1.31.1.1.1.6.3 = Counter64: 20026
.1.3.6.1.2.1.31.1.1.1.6.4 = Counter64: 246481741
```

×

.1.3.6.1.2.1.31.1.1.1.1.1 = STRING: lo
.1.3.6.1.2.1.31.1.1.1.1.2 = STRING: eth0
.1.3.6.1.2.1.31.1.1.1.1.3 = STRING: eth1
.1.3.6.1.2.1.31.1.1.1.1.4 = STRING: podman1

Interface name
used in Item name

.1.3.6.1.2.1.31.1.1.1.10.1 = Counter64: 2360
.1.3.6.1.2.1.31.1.1.1.10.2 = Counter64: 5875130
.1.3.6.1.2.1.31.1.1.1.10.3 = Counter64: 20032
.1.3.6.1.2.1.31.1.1.1.10.4 = Counter64: 274374233
.1.3.6.1.2.1.31.1.1.1.6.1 = Counter64: 2360
.1.3.6.1.2.1.31.1.1.1.6.2 = Counter64: 536085556
.1.3.6.1.2.1.31.1.1.1.6.3 = Counter64: 20026
.1.3.6.1.2.1.31.1.1.1.6.4 = Counter64: 246481741

Interface traffic
used in Item values

The SNMP walk[*] item is designed to be used as a master item:

- ⚡ The master item collects all SNMP data at once
- ⚡ Dependent items extract values from the master item using preprocessing
- ⚡ Dependent items do not have individual update intervals
- ⚡ Additionally, an LLD rule can also be created as a dependent item

Dependent items are displayed with their master item name as a prefix

Master item: Dependent item

<input type="checkbox"/>	Name ▲	Triggers	Key	Interval	History	Trends	Type	Status	Tags	Info
<input type="checkbox"/>	... SNMP raw data: Interface Fa3/0/1: Bits received		net.if.in[11001]		90d	365d	Dependent item	Enabled		
<input type="checkbox"/>	... SNMP raw data: Interface Fa3/0/2: Bits received		net.if.in[11002]		90d	365d	Dependent item	Enabled		
<input type="checkbox"/>	... SNMP raw data: Interface Fa3/0/3: Bits received		net.if.in[11003]		90d	365d	Dependent item	Enabled		
<input type="checkbox"/>	... SNMP raw data: Interface Fa3/0/4: Bits received		net.if.in[11004]		90d	365d	Dependent item	Enabled		
<input type="checkbox"/>	... SNMP raw data		snmp.raw.data	1m	90d		SNMP agent	Enabled		

Displaying 5 of 5 found

<input type="checkbox"/>	Host	Name ▲	Items	Triggers	Graphs	Hosts	Key	Interval	Type	Status	Info
<input type="checkbox"/>	SNMP Host	SNMP walk network interfaces: Network interfaces discovery	Item prototypes 2	Trigger prototypes	Graph prototypes	Host prototypes	net.if.discovery		Dependent item	Enabled	

Displaying 1 of 1 found



Discovery rule

Preprocessing 1

LLD macros

Filters 2

Overrides

* Name

Network interfaces discovery

Type

Dependent item

* Key

net.if.discovery

* Master item

SNMP Host: SNMP walk network interfaces

Select

* Keep lost resources period

30d

Description

Discovering interfaces from IF-MIB.

Enabled

☒

Update

Clone

Execute now

Test

Delete

Cancel

A new preprocessing step has been introduced - **SNMP walk to JSON**:

🔊 Takes raw SNMP data as input

🔊 Transforms data into JSON format while creating LLD macros

The screenshot shows the configuration interface for the 'SNMP walk to JSON' preprocessing step. The interface includes tabs for 'Discovery rule', 'Preprocessing 1', 'LLD macros', 'Filters 2', and 'Overrides'. The 'Preprocessing steps' section shows a list with one step: '1: SNMP walk to JSON'. The 'Parameters' section includes fields for 'Field name' (set to '{#IFNAME}'), 'OID prefix' (set to '1.3.6.1.2.1.31.1.1'), 'Format' (set to 'Unchanged'), and 'Action' (set to 'Remove'). There are also 'Add', 'Update', 'Clone', 'Execute now', 'Test', 'Delete', and 'Cancel' buttons. A 'Custom on fail' checkbox is present, and a 'Test all steps' button is at the bottom right.

.1.3.6.1.2.1.31.1.1.1.1.1.1.1.1.1.1 = STRING: lo
.1.3.6.1.2.1.31.1.1.1.1.1.1.1.1.1.2 = STRING: eth0
.1.3.6.1.2.1.31.1.1.1.1.1.1.1.1.1.3 = STRING: eth1
.1.3.6.1.2.1.31.1.1.1.1.1.1.1.1.1.4 = STRING: podman1
.1.3.6.1.2.1.31.1.1.1.1.1.1.1.1.1.10.1 = Counter64: 2360
.1.3.6.1.2.1.31.1.1.1.1.1.1.1.1.1.10.2 = Counter64: 588242412
.1.3.6.1.2.1.31.1.1.1.1.1.1.1.1.1.10.3 = Counter64: 20032
.1.3.6.1.2.1.31.1.1.1.1.1.1.1.1.1.10.4 = Counter64: 274481263
.1.3.6.1.2.1.31.1.1.1.1.1.1.1.1.1.1.6.1 = Counter64: 2360
.1.3.6.1.2.1.31.1.1.1.1.1.1.1.1.1.1.6.2 = Counter64: 560415523
.1.3.6.1.2.1.31.1.1.1.1.1.1.1.1.1.1.6.3 = Counter64: 20096
.1.3.6.1.2.1.31.1.1.1.1.1.1.1.1.1.1.6.4 = Counter64: 246573377



```
[  
  {  
    "{#SNMPINDEX}":"2","{#IFNAME}":"eth0"  
  },  
  {  
    "{#SNMPINDEX}":"4","{#IFNAME}":"podman1"  
  },  
  {  
    "{#SNMPINDEX}":"3","{#IFNAME}":"eth1"  
  },  
  {  
    "{#SNMPINDEX}":"1","{#IFNAME}":"lo"  
  }  
]
```


Item prototypes are also created as dependent items:

- ⚡ LLD macros are used to define the item prototype
- ⚡ New preprocessing step called **SNMP walk value** is used to extract values

Item prototype Tags Preprocessing 3

* Name

Type

* Key

Type of information

* Master item

Units

Item prototype Tags 1 Preprocessing 3

Preprocessing steps	Name	Parameters	Custom on fail	Actions
1:	SNMP walk value	1.3.6.1.2.1.31.1.1.1.6.{#SNMPINDEX}	<input type="text" value="Unchanged"/>	Test Remove
2:	Change per second		<input type="text"/>	Test Remove
3:	Custom multiplier	8	<input type="text"/>	Test Remove

[Add](#) [Test all steps](#)

<input type="checkbox"/>	Name ▼	Triggers	Key	Interval	History	Trends	Type	Status	Tags	Info
<input type="checkbox"/>	... SNMP walk network interfaces		net.if.walk	1m	7d		SNMP agent	Enabled	component: network	component: raw
<input type="checkbox"/>	... Network interfaces discovery: SNMP walk network interfaces: Interface podman1: Bits sent		net.if.out[4]		90d	365d	Dependent item	Enabled	component: network	
<input type="checkbox"/>	... Network interfaces discovery: SNMP walk network interfaces: Interface podman1: Bits received		net.if.in[4]		90d	365d	Dependent item	Enabled	component: network	
<input type="checkbox"/>	... Network interfaces discovery: SNMP walk network interfaces: Interface lo: Bits sent		net.if.out[1]		90d	365d	Dependent item	Enabled	component: network	
<input type="checkbox"/>	... Network interfaces discovery: SNMP walk network interfaces: Interface lo: Bits received		net.if.in[1]		90d	365d	Dependent item	Enabled	component: network	
<input type="checkbox"/>	... Network interfaces discovery: SNMP walk network interfaces: Interface eth1: Bits sent		net.if.out[3]		90d	365d	Dependent item	Enabled	component: network	
<input type="checkbox"/>	... Network interfaces discovery: SNMP walk network interfaces: Interface eth1: Bits received		net.if.in[3]		90d	365d	Dependent item	Enabled	component: network	
<input type="checkbox"/>	... Network interfaces discovery: SNMP walk network interfaces: Interface eth0: Bits sent		net.if.out[2]		90d	365d	Dependent item	Enabled	component: network	
<input type="checkbox"/>	... Network interfaces discovery: SNMP walk network interfaces: Interface eth0: Bits received		net.if.in[2]		90d	365d	Dependent item	Enabled	component: network	

Displaying 9 of 9 found

.1.3.6.1.2.1.31.1.1.1.1 = STRING: lo
 .1.3.6.1.2.1.31.1.1.1.2 = STRING: eth0
 .1.3.6.1.2.1.31.1.1.1.3 = STRING: eth1
 .1.3.6.1.2.1.31.1.1.1.4 = STRING: podman1
 .1.3.6.1.2.1.31.1.1.10.1 = Counter64: 2360
 .1.3.6.1.2.1.31.1.1.10.2 = Counter64: 588242412
 .1.3.6.1.2.1.31.1.1.10.3 = Counter64: 20032
 .1.3.6.1.2.1.31.1.1.10.4 = Counter64: 274481263
 .1.3.6.1.2.1.31.1.1.6.1 = Counter64: 2360
 .1.3.6.1.2.1.31.1.1.6.2 = Counter64: 560415523
 .1.3.6.1.2.1.31.1.1.6.3 = Counter64: 20096
 .1.3.6.1.2.1.31.1.1.6.4 = Counter64: 246573377



Interface podman1: bits sent 246573377
 Interface podman1: bits received 274481263
 Interface lo: bits sent 2360
 Interface lo: bits received 2360
 Interface eth1: bits sent 20032
 Interface eth1: bits received 20096
 Interface eth0: bits sent 588242412
 Interface eth0: bits received 560415523

<input type="checkbox"/> Host	Name ▲	Last check	Last value	Change	Tags	Info
<input type="checkbox"/> SNMP Host	Interface eth0: Bits received	3s	69.63 Mbps	+69.63 Mbps	component: network	Graph
<input type="checkbox"/> SNMP Host	Interface eth0: Bits sent	3s	493.13 Kbps	+489.97 Kbps	component: network	Graph
<input type="checkbox"/> SNMP Host	Interface eth1: Bits received	3s	12.48 Mbps		component: network	Graph
<input type="checkbox"/> SNMP Host	Interface eth1: Bits sent	3s	56.72 Kbps		component: network	Graph
<input type="checkbox"/> SNMP Host	Interface lo: Bits received	3s	0 bps		component: network	Graph
<input type="checkbox"/> SNMP Host	Interface lo: Bits sent	3s	0 bps		component: network	Graph
<input type="checkbox"/> SNMP Host	Interface podman1: Bits received	3s	592 bps	+48 bps	component: network	Graph
<input type="checkbox"/> SNMP Host	Interface podman1: Bits sent	3s	648 bps	-8 bps	component: network	Graph
<input type="checkbox"/> SNMP Host	SNMP walk network interfaces ?	3s	.1.3.6.1.2.1.31.1.1....		component: network component: raw	History
Displaying 9 of 9 found						



Timestamp	Value
2023-04-11 00:26:06	<pre>.1.3.6.1.2.1.31.1.1.1.1.1 = STRING: lo .1.3.6.1.2.1.31.1.1.1.1.2 = STRING: eth0 .1.3.6.1.2.1.31.1.1.1.1.3 = STRING: eth1 .1.3.6.1.2.1.31.1.1.1.1.4 = STRING: podman1 .1.3.6.1.2.1.31.1.1.1.10.1 = Counter64: 2334560 .1.3.6.1.2.1.31.1.1.1.10.2 = Counter64: 593325209 .1.3.6.1.2.1.31.1.1.1.10.3 = Counter64: 2210102 .1.3.6.1.2.1.31.1.1.1.10.4 = Counter64: 274672390 .1.3.6.1.2.1.31.1.1.1.6.1 = Counter64: 2360 .1.3.6.1.2.1.31.1.1.1.6.2 = Counter64: 1212933116 .1.3.6.1.2.1.31.1.1.1.6.3 = Counter64: 20096 .1.3.6.1.2.1.31.1.1.1.6.4 = Counter64: 246748076</pre>

It is possible to transform the raw SNMP value:

⚡ to **UTF-8** from Hex-STRING convert hex-string to a UTF-8 string

.1.3.6.1.4.1.1602.1.2.1.4.0 = hex-STRING: 0115434e443846374d485646 => **CND8F7MHV**

⚡ to **MAC** from Hex-STRING return a MAC address string (' ' are replaced by ':')
' :')

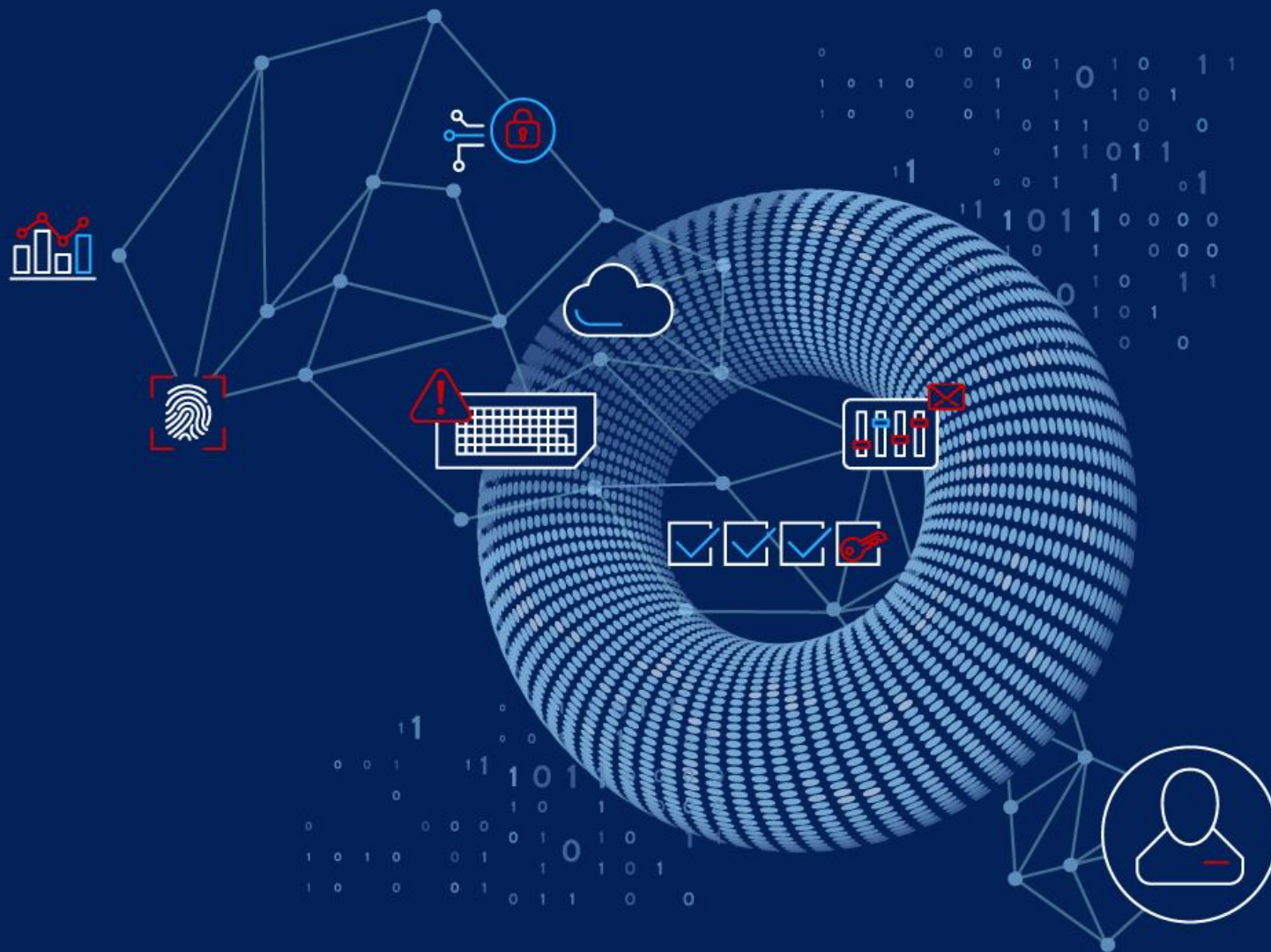
1.3.6.1.2.1.17.4.3.10.20.253.193 = Hex-STRING: 00 01 6C 14 FD C1 => **00:01:6C:14:FD:C1**

⚡ to an **integer** from BITS convert the first 8 bytes into a 64-bit integer

JUNIPER-DOM-MIB::jnxDomCurrentAlarms.559 = BITS: 0A 00 F1 => **15794186**

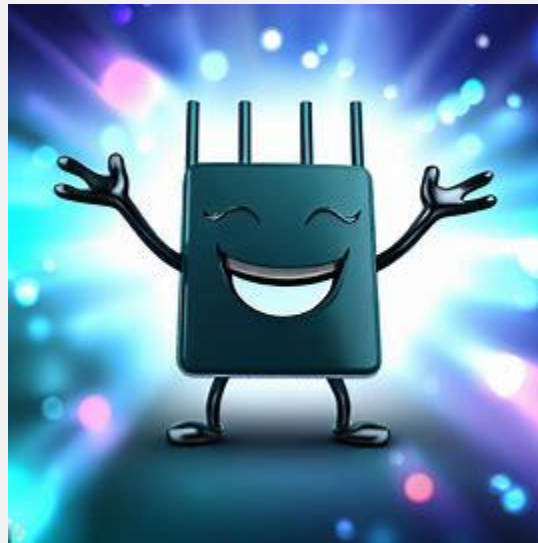
04

CONCLUSION



There are multiple benefits of using the new "walk" method:

- ~ Fewer requests to network device - all data are collected at once
- ~ Reduced network traffic: multiple sets of tabular data retrieved in a single request
- ~ Reduced workload on the SNMP agent
- ~ SNMP GETBULK can be particularly beneficial in large-scale environments
- ~ All the data is being collected at the same time and are consistent



All official Zabbix SNMP templates will be updated soon:

- ⚡ The new "walk" method will be used
- ⚡ Our integration team is working on this, and there is still work in progress
- ⚡ We need to redesign **over 140** SNMP templates.

<input type="checkbox"/>	HP Enterprise Switch by SNMP	Hosts	Items 15	Triggers 9	Graphs 1	Dashboards 1	Discovery 8	Web	Zabbix	6.4-0	class: network target: hp target: hp-enterprise
<input type="checkbox"/>	HPE ProLiant BL460 by SNMP	Hosts	Items 13	Triggers 6	Graphs	Dashboards	Discovery 14	Web	Zabbix	6.4-0	class: hardware target: hp target: hp bl460
<input type="checkbox"/>	HPE ProLiant BL920 by SNMP	Hosts	Items 13	Triggers 6	Graphs	Dashboards	Discovery 14	Web	Zabbix	6.4-0	class: hardware target: hp target: hp bl920
<input type="checkbox"/>	HPE ProLiant DL360 by SNMP	Hosts	Items 13	Triggers 6	Graphs	Dashboards	Discovery 14	Web	Zabbix	6.4-0	class: hardware target: hp target: hp bl360
<input type="checkbox"/>	HPE ProLiant DL380 by SNMP	Hosts	Items 13	Triggers 6	Graphs	Dashboards	Discovery 14	Web	Zabbix	6.4-0	class: hardware target: hp target: hp bl380
<input type="checkbox"/>	HP iLO by SNMP	Hosts	Items 16	Triggers 9	Graphs	Dashboards	Discovery 13	Web	Zabbix	6.4-0	class: hardware target: hp target: ilo
<input type="checkbox"/>	Huawei OceanStor 5300 V5 by SNMP	Hosts	Items 16	Triggers 7	Graphs 1	Dashboards	Discovery 8	Web	Zabbix	6.4-0	class: hardware target: huawei class: storage
<input type="checkbox"/>	Huawei VRP by SNMP	Hosts	Items 12	Triggers 6	Graphs	Dashboards 1	Discovery 5	Web	Zabbix	6.4-0	class: network target: huawei target: huawei-vrp
<input type="checkbox"/>	IBM IMM by SNMP	Hosts	Items 15	Triggers 10	Graphs	Dashboards	Discovery 6	Web	Zabbix	6.4-0	class: hardware target: ibm target: imm
<input type="checkbox"/>	Intel_Qlogic Infiniband by SNMP	Hosts	Items 14	Triggers 7	Graphs	Dashboards 1	Discovery 5	Web	Zabbix	6.4-0	class: network target: intel target: qlogic-infiniband

123▶

Displaying 1 to 50 of 142 found

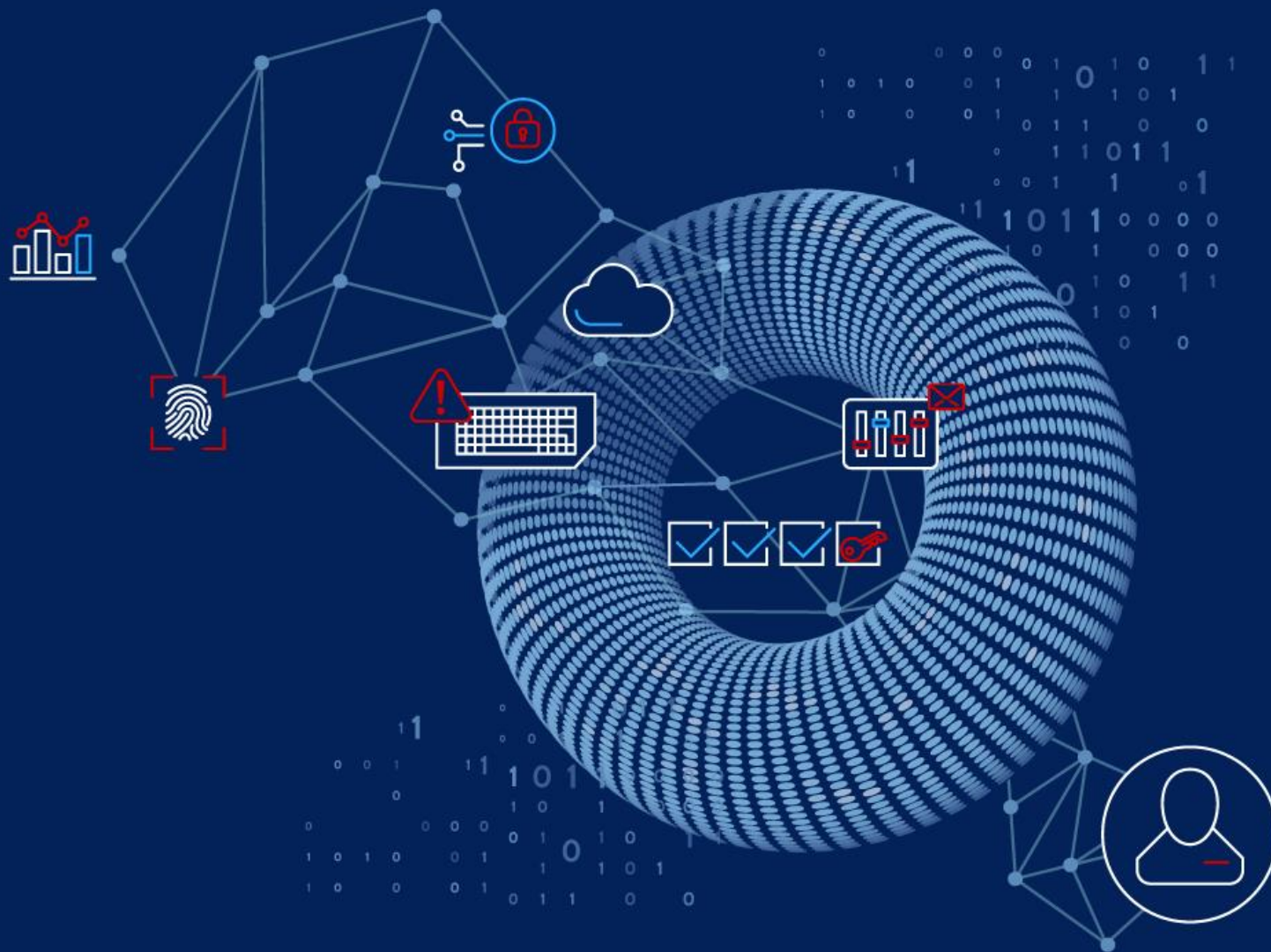
A new one-day **SNMP training course** will be released soon:

- ⚡ Suitable for booth beginners and intermediate users
- ⚡ All three SNMP versions explained in detail
- ⚡ SNMP MIB files and OIDs
- ⚡ SNMP command line utilities
- ⚡ SNMP LLD methods - legacy and the new walk[*] method
- ⚡ SNMP traps
- ⚡ Practical tasks with examples from HP, Cisco, APC, MikroTik and other vendors



05

WORKSHOP



Workshop: New and improved SNMP bulk data collection in Zabbix 6.4

- ⚡ Starts Saturday at **10:15 AM**
- ⚡ Every attendee will be provided with a cloud-based virtual machine
- ⚡ You need a **laptop with Internet connection**
- ⚡ Required software:
 - ✓ SSH client
 - ✓ Web browser
- ⚡ We will create a host and collect SNMP metrics using:
 - ✓ Combined requests
 - ✓ SNMP walk[*] item and LLD rule



ZABBIX

6.4

THANK YOU

