Browser monitoring and use cases

ZABBIX
SUMMIT
2024

Aleksandrs Petrovs-Gavrilovs

Training technical administrator

Browser monitoring

BROWSER MONITORING

Browser monitoring or **Browser Synthetic Monitoring**

A way to constantly checks website or application performance by simulating user actions, ensuring the production site runs smoothly regardless of user activity.

▶ Early issue detection during development or in a live environment

▶ Simulates user interactions with scripted tests

▶ Ensures a proactive check on application health

▶ Long-term and relevant health statistics

▶ Constant quality control

Types of browser monitoring

▶ **Single-URL Monitors**
- Simulate a user visiting a specific URL and provide information such as availability and performance metrics.

▶ **Browser Clickpaths**
- Simulated user navigation through a site, automatically testing availability and performance at scheduled intervals. This approach can be used to detect and report UI bugs or unexpected functionality.

▶ **Mobile Simulation**
- Simulates mobile interactions to assess the availability and performance of mobile versions of websites.

▶ **HTTP(S) monitors**
- Simple HTTP requests. You can use them to monitor the availability of your API endpoints or perform simple HTTP checks for single-resource availability
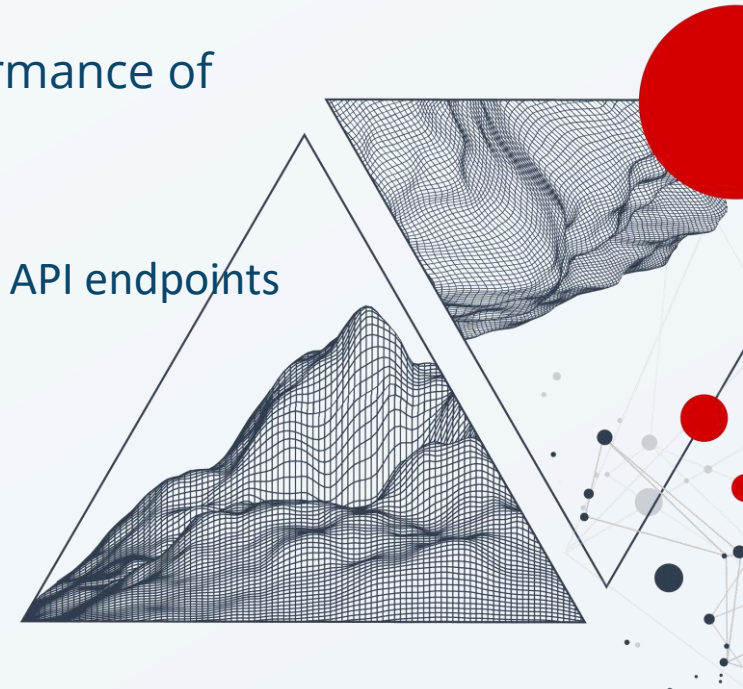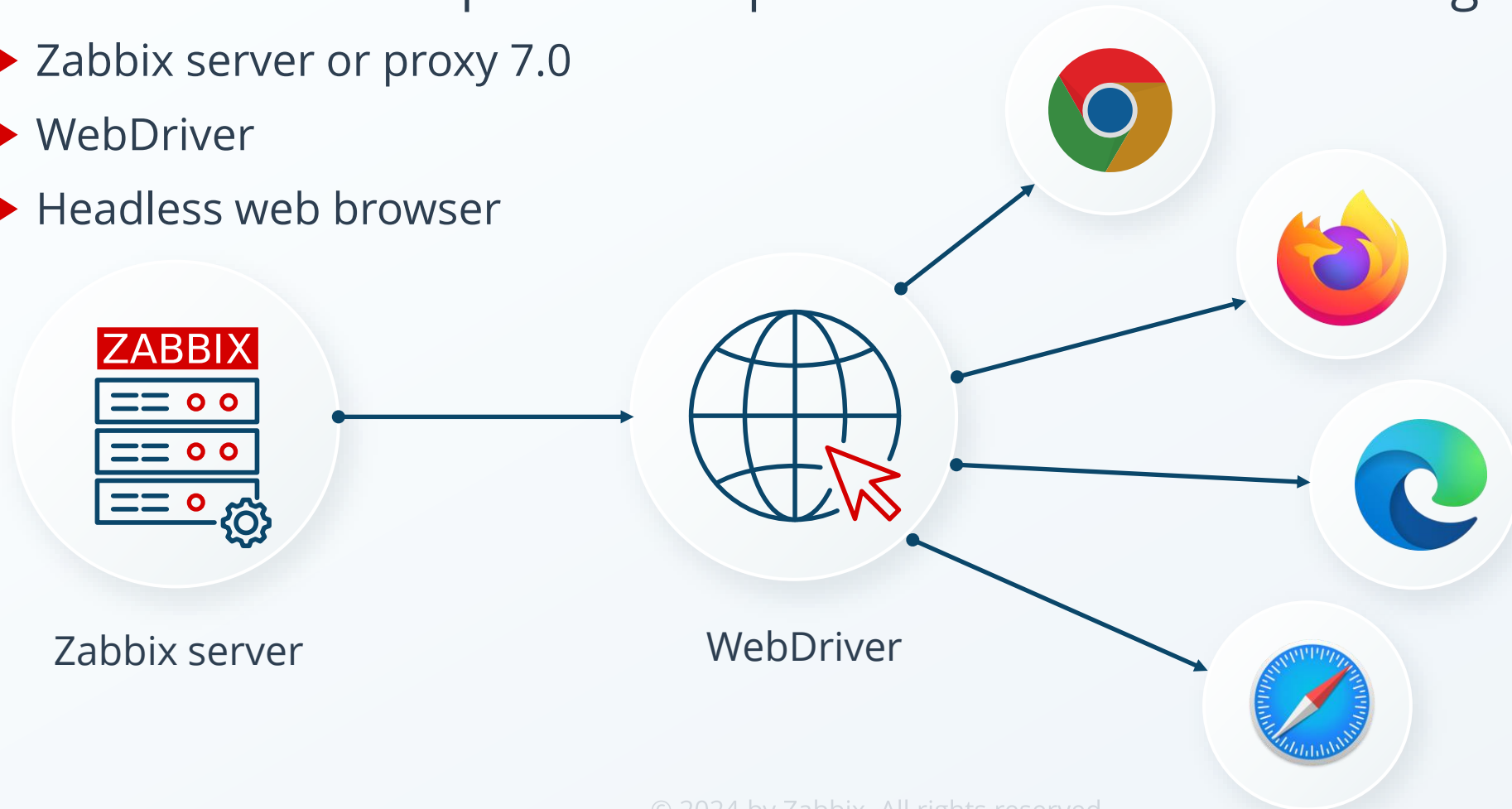
Website monitoring setup

MONITORING REQUIREMENTS

There are several components required for browser monitoring:

▶ Zabbix server or proxy 7.0

▶ WebDriver

▶ Headless web browser

Zabbix server

WebDriver

ZABBIX SERVER CONFIGURATION

The Zabbix server configuration file has two new parameters:

- ▶ WebDriverURL
- ▶ StartBrowserPollers

```
####### Browser monitoring #######

### Option: WebDriverURL
#          WebDriver interface HTTP[S] URL. For example http://localhost:4444 used with
#          Selenium WebDriver standalone server.
#
# Mandatory: no
# Default:
# WebDriverURL=
WebDriverURL=http://192.168.0.1:4444

### Option: StartBrowserPollers
#          Number of pre-forked instances of browser item pollers.
#
# Mandatory: no
# Range: 0-1000
# Default:
# StartBrowserPollers=1
StartBrowserPollers=3
```

BROWSER ITEM
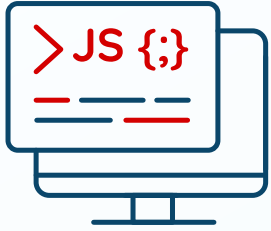
In Zabbix 7.0, a new item type is introduced: Browser

▶ Based on JavaScript

▶ Supports the browser class

▶ Support various new methods to work with pages

New item   ✕

Item   Tags   Preprocessing

* Name   Zabbix website

Type   Browser

* Key   website.get.data   Select

Type of information   Text

Parameters

Name	Value	Action
		Remove

Add

* Script   var browser = new Browser(Browser.chromeOptions());...

* Update interval   1m

JavaScript   ✕

```
1  var browser = new Browser(Browser.chromeOptions());
2
3  try {
4          browser.navigate("http://example.com");
5          browser.collectPerfEntries();
6  }
7  finally {
8          return JSON.stringify(browser.getResult());
9  }
```

65346 characters remaining

Apply   Cancel

Taking screenshots

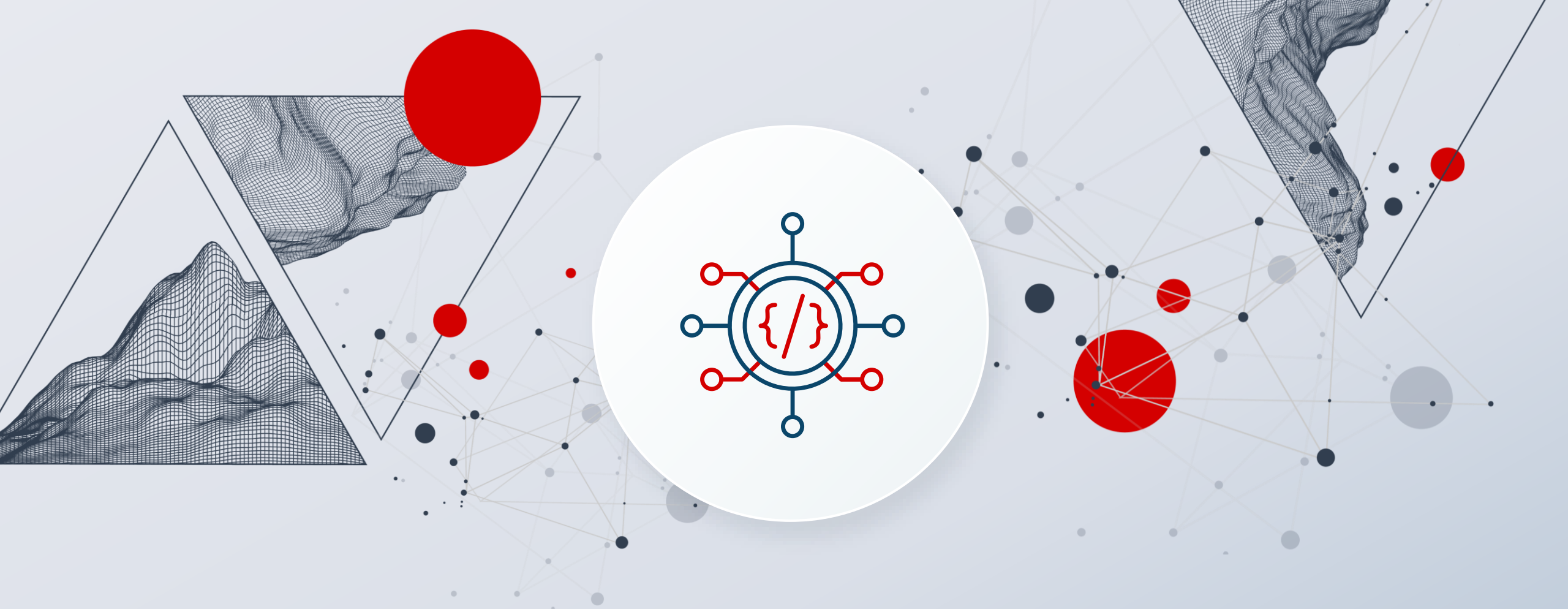# The browser item can take screenshots of the monitored pages:

- ▶ To take a screenshot of the page – use the *getScreenshot()* method
- ▶ Screenshot is included in the JSON object in base64 format
- ▶ It is extracted into a binary dependent item
- ▶ Can be viewed on the dashboards or sent via media type

10

Screenshot size: up to 8K.

▶ Default screenshot size 1920x1080px.

▶ Screenshot can be displayed using the "Item history" widget.

▶ Screenshot size is specified by the *setScreenSize(x,y)* method.

Objects and methods

BROWSER OBJECTS AND METHODS

Browser object - manages WebDriver session through out the whole item execution.

> ▶ (initialize session => execute all actions in script => close session).

Element - an element that is found on the opened webpage.

Cookie - a cookie returned by the **getCookies()** method.

Alert - a webpage alert returned by the **getAlert()** method.

Result - represents the collected statistics data gathered by **getResult()** method. Contains the following:

▶ Session statistics (in example total session duration)

▶ Error information (if such occurred)

▶ Performance data (if such collected)

▶ Any collected data that is written directly to the result object

Browser object methods:

Retrieving predefined browser options	chromeOptions() firefoxOptions() edgeOptions() safariOptions()	Error related operations	getError() setError(message) discardError()
Timeout management	setScriptTimeout(timeout) setSessionTimeout(timeout) setElementWaitTimeout(timeout)	URL and page related operations	navigate(url) getUrl() getPageSource() getAlert() getCookies() addCookie(cookie)
Collecting data	collectPerfEntries(mark) getRawPerfEntries() getResult() getScreenshot()	Locating elements on page	findElement(strategy, selector) findElements(strategy, selector)

Element object methods:

getAttribute(name)	Return the value of the given attribute of the object, for example: class, id, style, display e.t.c.	click()	Click on element.
getProperty(name)	Return the value of the given property of the object, for example: className, baseURIe.t.c.	sendKeys(keys)	Send keys to element. Method is used for filling data into editable elements.
getText()	Return element text.	clear()	Remove the content of editable element

Browser Monitoring Objects and Methods

Use cases

Out-of-the-box approach

Quick start

Create a host that represents your page

Link "Website by Browser" template

Specify the page to monitor

Simple start

Results

Website zabbix.com Get data [?]

Website zabbix.com Navigation decodedBody size [?]	2m 2s	331.62 KB
Website zabbix.com Navigation DNS lookup time [?]	2m 2s	0
Website zabbix.com Navigation domContentLoaded time [?]	2m 2s	0
Website zabbix.com Navigation encodedBody size [?]	2m 2s	3
Website zabbix.com Navigation load event time [?]	2m 2s	3
Website zabbix.com Navigation request time [?]	2m 2s	0
Website zabbix.com Navigation resource fetch time [?]	2m 2s	0
Website zabbix.com Navigation response time [?]	2m 2s	0
Website zabbix.com Navigation service worker processin... [?]		
Website zabbix.com Navigation TCP handshake time [?]	2m 2s	0
Website zabbix.com Navigation TLS negotiation time [?]	2m 2s	0
Website zabbix.com Navigation transfer size [?]	2m 2s	3
Website zabbix.com Resource decodedBody size [?]	2m 2s	8

Item history

Timestamp	Summit agenda
2024-09-11 15:52:41 AM	
2024-09-11 14	
2024-09-11 13	
2024-09-11 12	
2024-09-11 11	

ZABBIX

PRODUCT SOLUTIONS SUPPORT & SERVICES TRAINING PARTNERS COMMUNITY ABOUT US DOWNLOAD

Overview   Venue   Registration   Speakers   Agenda   F.A.Q.   Terms & Conditions

Agenda

Day 1: Oct 4, 2024
Riga (GMT+3)   MAIN STAGE

09:00   Registration

09:30   Summit Opening

09:40   KEYNOTE
Alexel Vladishev, CEO & Founder, ZABBIX, Latvia

10:20   When Zabbix Meets AI - Enhancing Zabbix
monitoring workflows with AI features
Tomas Hermanek, CEO & Founder, initMAX s.r.o., Czech
Republic

10:50   TBA
Dmitrijs Lamberts, Head of Technical Support, Zabbix, Latvia

11:10   Browser monitoring in Zabbix 7.0: use case
overview
Aleksandrs Petrovs-Gavrilovs, Technical Support Engineer,
Trainer, Zabbix, Latvia

11:30   Lunch

12:30   Zabbix meets AI - Leverage machine learning for

Day 1: Oct 4, 2024
Riga (GMT+3)   BUSINESS TRACK

14:30   The evolution of IT monitoring landscape
Luciano Alves, Country Manager, Zabbix Latam, Brazil

15:00   When two OpenSource projects meet
Evgeny Yurchenko, Founder & Tech lead, BGmot, Canada

15:30   Zabbix for MSPs
Andre Morton, , AGM Network Consultancy, United Kingdom

16:00   Coffee break

16:30   Business impact of root cause problem detection
Steve Destivelle, CEO, IZI-IT, France

Authenticate

Quick and a bit more advanced

Create a template that represents your login page

Link it to hosts that represent the pages

Monitor if login is working, confirm with screenshots

Potential variables - user macros

ZABBIX

▶ You may need to monitore multiple pages

▶ Credential (hopefully) will be diferrent

▶ Various strategies may need to be applied

- CSS selector
- Link text
- Partial link text
- Tag name
- Xpath

Template macros	Inherited and template macros

Macro	Value	
{$LINK}	https://support.zabbix.com/	T ˅
{$LOGIN.FORM.BUTTON.PATH}	input#login	T ˅
{$LOGIN.FORM.FIELD.PATH}	input#login-form-username	T ˅
{$LOGIN.FORM.PASS.FIELD.PATH}	input#login-form-password	T ˅
{$LOGIN.FORM.STRATEGY.TYPE}	css selector	T ˅
{$PAGE.PASSWORD}	••••••	👁 ˅
{$PAGE.USERNAME}	••••••	👁 ˅

Configure the master item


```javascript
3   var browser, result;
4   browser = new Browser(Browser.chromeOptions());
5   browser.setScreenSize(Number(1980), Number(1020));
6   screens = [];
7   const screenshot...
8   try {
9
10  var params = JSON.parse(value); // Parse the JSON string containing parameters passed from Zabbix.
11
12  browser.navigate(params.url);
13  screens.push(browser.getScreenshot());
14
15
16  // enter username
17  var el = browser.findElement(params.strategy, params.loginfield);
18  if (el === null) {throw Error("cannot find name input field");}
19  el.sendKeys(params.username);
20  screens.push(brows...
21
22  // enter password
23  el = browser.findElement(params.strategy, params.passwordfield);
24  if (el === null) { throw Error("cannot find password input field"); }
25  el.sendKeys(params.password);
26  screens.push(browser.getScreenshot());
27
28  // login
29  el = browser.findElement(params.strategy, params.button);
30  if (el === null) { throw Error("cannot find login button"); }
31  el.click();
```
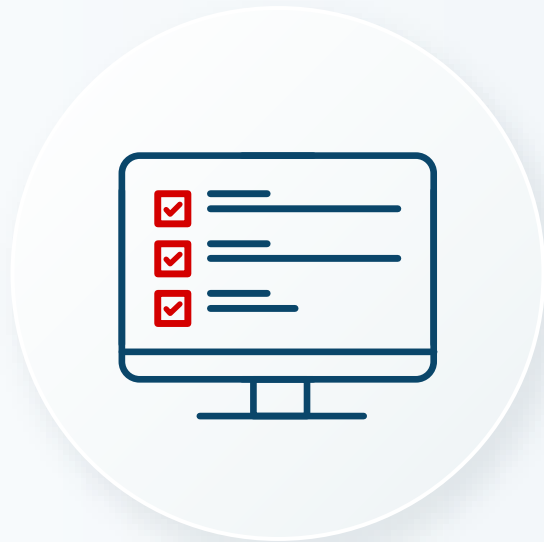
Parsing parameters

Passing parameters

Make screenshots for each step

Fill in any form

First name *

Aleksandrs

Last name *

Petrovs-Gavrilovs

Company *

SIA ZABBIX

Country / Territory *

Latvia

Phone number *

371 67784742

Business email address *

zabbix@zabbix.com

Comments

Enter your comments or questions here

By clicking below, you are agreeing to the Zabbix Priv

Submit Training Request

JavaScript

```
34  // enter name
35  var el = browser.findElement("xpath", "//label[text()='First name']/following::input");
36  if (el === null) {throw Error("cannot find name input field name");}
37  el.sendKeys("Aleksandrs");
38
39  // enter last name
40  var el = browser.findElement("xpath", "//label[text()='Last name']/following::input");
41  if (el === null) {throw Error("cannot find name input field last name");}
42  el.sendKeys("Petrovs-Gavrilovs");
43
44  // enter company name
45  var el = browser.findElement("xpath", "//label[text()='Company']/following::input");
46  if (el === null) {throw Error("cannot find name input field company name");}
47  el.sendKeys("SIA Zabbix");
48
49  // select country
50  var el = browser.findElement("css selector", "select#country");
51  if (el === null) {throw Error("cannot find country selection");}
52  el.sendKeys("Latvia");
```

Filling form

Click one

Zabbix certificate granted

We hereby certify that **Aleksandrs Petrovs-Gavrilovs** has passed **Zabbix Certified Trainers course 7.0** exam on **2024-04-21** and received a certificate **CT-2404-003.**

Download certificate | **Download badge** | Perform

ZABBIX

CERTIFICATE

CT-2404-001   APRIL 22, 2024

This Certificate accredits that

JavaScript

```javascript
10  // enter name
11  var el = browser.findElement("xpath", "//label[text()='First Name']/following::input");
12  if (el === null) {throw Error("cannot find name input field");}
13  el.sendKeys("Aleksandrs");
14
15  // enter last name
16  var el = browser.findElement("xpath", "//label[text()='Last name']/following::input");
17  if (el === null) {throw Error("cannot find name input field");}
18  el.sendKeys("Petrovs-Gavrilovs");
19
20  // enter cert number
21  var el = browser.findElement("xpath", "//label[text()='Certificate number']/following::input");
22  if (el === null) {throw Error("cannot find name input field");}
23  el.sendKeys("CT-2404-003");
24
25  // select vers
26  var el = brows                                icate_validation>fieldset>div:nth-of-type(5)>select");
27  if (el === null) {throw Error("cannot find name input field");}
28  el.sendKeys("7.0");
29
30  // check certificate
31  var el = browser.findElement("xpath", "//button[text()='Check Certificate']");
32  if (el === null) {throw Error("cannot find name input field");}
33  el.click();
34
35      result = browser.getResult();
36      screenshot = browser.getScreenshot();
37      result.screenshot = browser.getScreenshot();
38  }
```

Filling diferrent form

Selecting version

ZABBIX
OFFICIAL MERCH

Home   Catalog

Microfiber cloth with Zabbix logo
€2,00 EUR
− 0 +

Mousepad with Zabbix logo (230x200mm)
€5,00 EUR
− 0 +

Red mug with Zabbix logo
€10,00 EUR
− 0 +

Red umbrella with Zabbix logo
€20,00 EUR
− 0 +

Socks with Zabbix logo (Colorful)
€5,00 EUR
− 99 +

Stocking up on socks

Socks with Zabbix logo (Dark Blue)
€5,00 EUR
− 0 +

Socks with Zabbix logo (Grey)
€5,00 EUR
− 0 +

Socks with Zabbix logo (Mou...
€5,00 EUR

Towel with Zabbix logo
€20,00 EUR
− 0 +

Host macros   Inherited and host macros

Macro   Value
{$CARD.NUMBER}   ••••••••••••••
{$EXPIRY.DATE}   ••••••••
{$CVC}   •••

Add

Credit card details

JavaScript

```
 1  var browser, result;
 2  var    opts = Browser.chromeOptions();
 3  opts.capabilities.alwaysMatch['goog:chromeOptions
 4  browser = new Browser(opts);
 5  browser.setScreenSize(Number(1980), Number(1020))
 6  try {
 7      browser.navigate("https://shop.zabbix.com/collections/summit-2024?page=2");
 8      browser.collectPerfEntries("open page");
 9
10      el = browser.findElement("css selector", "ul#product-grid>li:nth-of-type(5)>div>div>div:nth-of-type(2)>quick-add-bulk
11      if (el === null) {
12       throw Error("cannot find training");
13      }
14      el.sendKeys("99");
```
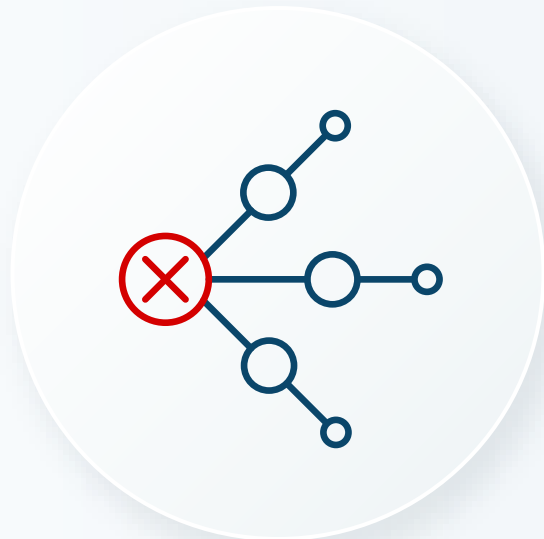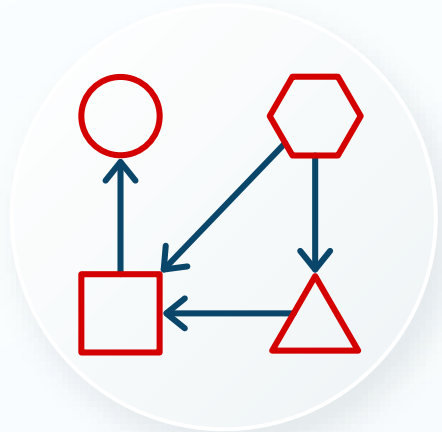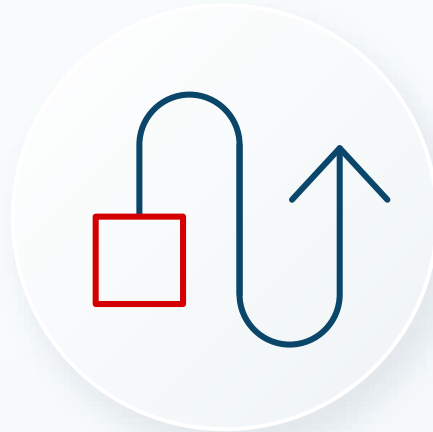
Pair count

Navigate via multiple paths

Need to make some clicks

Construct the path user must follow to get to the page

Make Zabbix follow the same path and alternative path in case of issues

Monitor if the path can be used by users

Navigate


```
11   //var el = browser.findElement("xpath", "//a[@href='/training']//span");
12   if (el === null) {
13       throw Error("cannot find traininglink");
14   }
15   el.click();
16
17   el = browser.findElement("link text", "Training Application Form");
18   if (el === null) {
19       throw Error("cannot find application link");
20   }
21   el.click();
22
23   el = browser.findElement("xpath", "//span[text()='Certified Specialist']");
24   if (el === null) {
25       throw Error("cannot select training");
26   }
27   el.click();
28
29   el = browser.findElement("xpath", "//td[text()='GMT+9 (Asia & Oceania)']");
30   if (el === null) {
31       throw Error("cannot find specific training");
32   }
33   el.click();
```

31

If-else scenarios


```
var el = browser.findElement("xpath", "//a[@href='/trainning']//span");
if (el === null) {
    var el = browser.findElement("css selector", "footer#footer>div>div:nth-of-type(3)>dl:nth-of-type(4)>dt>a");
    el.click();
} else {
    throw Error("cannot find training");
}
```

Links to training

IF - ELSE

Links to training

Get data - detect problems

Need to make some clicks

Decide which elements do you want to monitor (produce, tickets, users etc)

Navigate to the source of information

Parse the page to get the data

Discover the elements

```
JavaScript

43   // search for ZBXNEXT
44   el = browser.findElement("css selector", "textarea#advanced-search");
45   if (el === null) { throw Error("cannot find login button"); }
46   el.clear();
47   el.sendKeys("project = ZBXNEXT AND resolution = Unresolved ORDER BY created DESC"
48
49   // click search
50   el = browser.findElement("xpath", "//button[text()= search']");
51   if (el === null) { throw Error("cannot find login button"); }
52   el.click();
53
54   // wait for page to load
55   Zabbix.sleep(1000);
56
57   //get the ZBXNEXT
58   el = browser.findElements("xpath", "//*[contains(text(), '')]");
59   var links = [];
60   for (var n = 0; n < el.length; n++) {
61       links.push(el[n].getAttribute('href'));
62   }
63   links = uniq(links);
64   var lld = [];
65   for (i = 0; i < links.length; i++) {
66       if (links[i].match(/.ZBXNEXT.*/)) {
67           var row = {};
68           // Remove the '/browse/' string from the link
69           var modifiedLink = links[i].replace('/browse/', '');
70           row["{#ZBXNEXT}"] = modifiedLink;
71           lld.push(row);
```

JIRA search query

Looking for elements

☐	•••	Discover latest ZBXNEXT: Get ZBXNEXT-9413	Get.data[ZBXNEXT-9413]
☐	•••	Discover latest ZBXNEXT: Get ZBXNEXT-9414	Get.data[ZBXNEXT-9414]
☐	•••	Discover latest ZBXNEXT: Get ZBXNEXT-9415	Get.data[ZBXNEXT-9415]
☐	•••	Discover latest ZBXNEXT: Get ZBXNEXT-9417	Get.data[ZBXNEXT-9417]
☐	•••	Discover latest ZBXNEXT: Get ZBXNEXT-9419	Get.data[ZBXNEXT-9419]
☐	•••	Discover latest ZBXNEXT: Get ZBXNEXT-9420	Get.data[ZBXNEXT-9420]
☐	•••	Discover latest ZBXNEXT: Get ZBXNEXT-9421	Get.data[ZBXNEXT-9421]
☐	•••	Discover latest ZBXNEXT: Get ZBXNEXT-9422	Get.data[ZBXNEXT-9422]
☐	•••	Discover latest ZBXNEXT: Get ZBXNEXT-9423	Get.data[ZBXNEXT-9423]
☐	•••	Discover latest ZBXNEXT: Get ZBXNEXT-9424	Get.data[ZBXNEXT-9424]
☐	•••	Discover latest ZBXNEXT: Get ZBXNEXT-9425	Get.data[ZBXNEXT-9425]
☐	•••	Discover latest ZBXNEXT: Get ZBXNEXT-9426	Get.data[ZBXNEXT-9426]
☐	•••	Discover latest ZBXNEXT: Get ZBXNEXT-9427	Get.data[ZBXNEXT-9427]
☐	•••	Discover latest ZBXNEXT: Get ZBXNEXT-9428	Get.data[ZBXNEXT-9428]
☐	•••	Discover latest ZBXNEXT: Get ZBXNEXT-9431	Get.data[ZBXNEXT-9431]
☐	•••	Discover latest ZBXNEXT: Get ZBXNEXT-9432	Get.data[ZBXNEXT-9432]
☐	•••	Discover latest ZBXNEXT: Get ZBXNEXT-9433	Get.data[ZBXNEXT-9433]
☐	•••	Discover latest ZBXNEXT: Get ZBXNEXT-9434	Get.data[ZBXNEXT-9434]

Getting the info about elements

Item prototype

Item prototype   Tags   Preprocessing 1

* Name	{#ZBXNEXT} creation date
Type	Browser
* Key	Get.date[{#ZBXNEXT}] Select
Type of information	Numeric (unsigned)
Parameters	Name
	url
	web
	Add
* Script	var params = JSON.parse(value);…
Units	unixtime

```
browser.navigate(params.web + params.url);
var el = browser.findElements("css selector", "span#created-val>time");
var date= [];
for (var n = 0; n < el.length; n++) { date.push(el[n].getAttribute('datetime')); }
date = uniq(date);

return (date);
```

Getting the attribute

☐	Zabbix support portal	ZBXNEXT-9473 creation date	41m 58s	2024-08-29 11:13:51 …
☐	Zabbix support portal	ZBXNEXT-9474 creation date	41m 53s	2024-08-30 07:49:52 …
☐	Zabbix support portal	ZBXNEXT-9475 creation date	41m 47s	2024-08-30 10:01:13 …
☐	Zabbix support portal	ZBXNEXT-9477 creation date	41m 42s	2024-08-31 03:19:00 …
☐	Zabbix support portal	ZBXNEXT-9479 creation date	41m 36s	2024-08-31 07:40:56 …
☐	Zabbix support portal	ZBXNEXT-9481 creation date	41m 31s	2024-09-01 05:34:41 …

Creating the trigger

Trigger prototype

Trigger prototype   Tags   Dependencies

* Name	A new {#ZBXNEXT} has been created
Event name	A new {#ZBXNEXT} has been created
Operational data	

Severity | Not classified | Information | Warning | Average | High | Disaster |

* Expression   `(now()-last(/Zabbix support portal/Get.date[{#ZBXNEXT}]))/86400 <{$DAYS.SINCE.ZBXNEXT.CREATED}`   Add

	Time ▼	Severity	Recovery time	Status	Info	Host	Problem	Operational data
☐	09:43:00 AM •	Information		PROBLEM		Zabbix support portal	A new ZBXNEXT-9487 has been created	2024-09-17 12:33:03 PM
☐	09:42:32 AM •	Information		PROBLEM		Zabbix support portal	A new ZBXNEXT-9486 has been created	2024-09-17 04:28:44 AM

Send screenshots

Get the screenshots

	Name ▲	Triggers	Key	Interval	History	Trends	Type	Status
☐ ⋯	Discover latest ZBXNEXT: Most recent ZBXNEXT-9484 screenshot	Triggers 1	extract.screen[ZBXNEXT-9484]	12h	31d		HTTP agent	Enabled
☐ ⋯	Discover latest ZBXNEXT: Most recent ZBXNEXT-9485 screenshot	Triggers 1	extract.screen[ZBXNEXT-9485]	12h	31d		HTTP agent	Enabled
☐ ⋯	Discover latest ZBXNEX		36]	12h	31d		HTTP agent	Enabled
☐ ⋯	Discover latest ZBXNEX		37]	12h	31d		HTTP agent	Enabled
☐ ⋯	Discover latest ZBXNEX							
☐ ⋯	Discover latest ZBXNEX							
☐ ⋯	Discover latest ZBXNEX							

Request body type [Raw data] [**JSON data**] [XML data]

Request body

```
{
    "jsonrpc": "2.0",
    "method": "item.get",
    "params": {
        "output": ["itemid"],
        "hostids": "10651",
        "search": {
            "key_": "screenshot[{#ZBXNEXT}]"
        }
    },
    "auth":
"f2b116aec1e9a5262ec9244f0ee36df63f0de369f16c0761ee4e42
    "id": 1
}
```

JavaScript

```javascript
function (value) {

1   var test = (value)
2
3   // Create a new HttpRequest object
4       var req = new HttpRequest();
5
6       // Define the data to be sent in the request
7       var requestData = {
8           "jsonrpc": "2.0",
9           "method": "history.get",
10          "params": {
11              "output": "extend",
12              "history": 5,
13              "itemids": test,
14              "sortfield": "clock",
15              "sortorder": "DESC",
16              "limit": 1
17          },
18          "auth": "f2b116aec1e9a5262ec9244f0ee36df63f0de369f16c0761ee4e422d0e68927c",
19          "id": 1
20      };
21
```

Send them using email media type
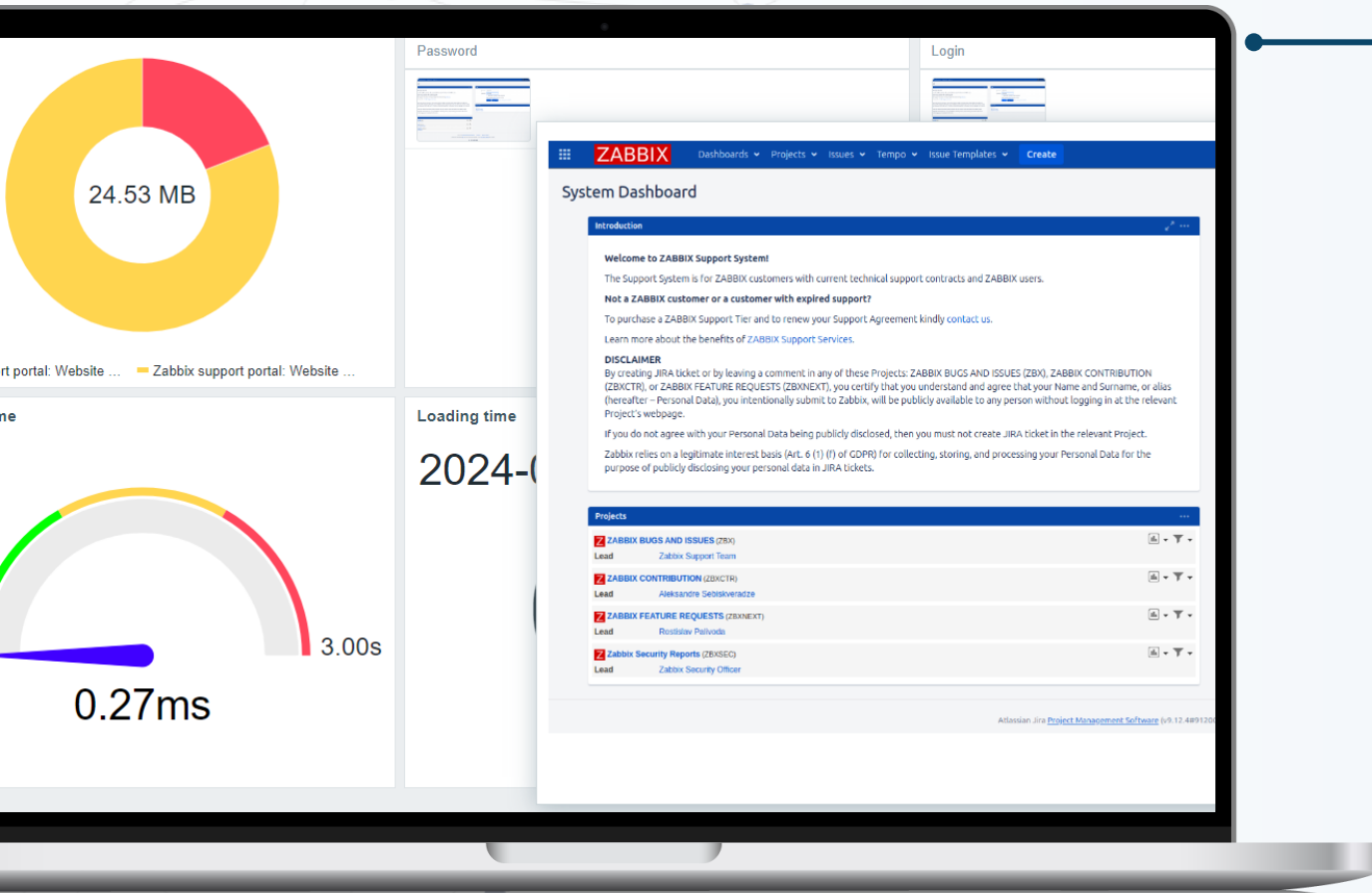
Observe comfortably

Use NoVNC

Build a dashboard

▶ Monitor performance

▶ Observe full set of screenshots

▶ See how the page changes after each new step

▶ Present the most relevant data

43

Thank you

Aleksandrs Petrovs-Gavrilovs

Training Technical Administrator