Integrating Zabbix with alerting and ITSM systems the easy way

Arturs Lontons

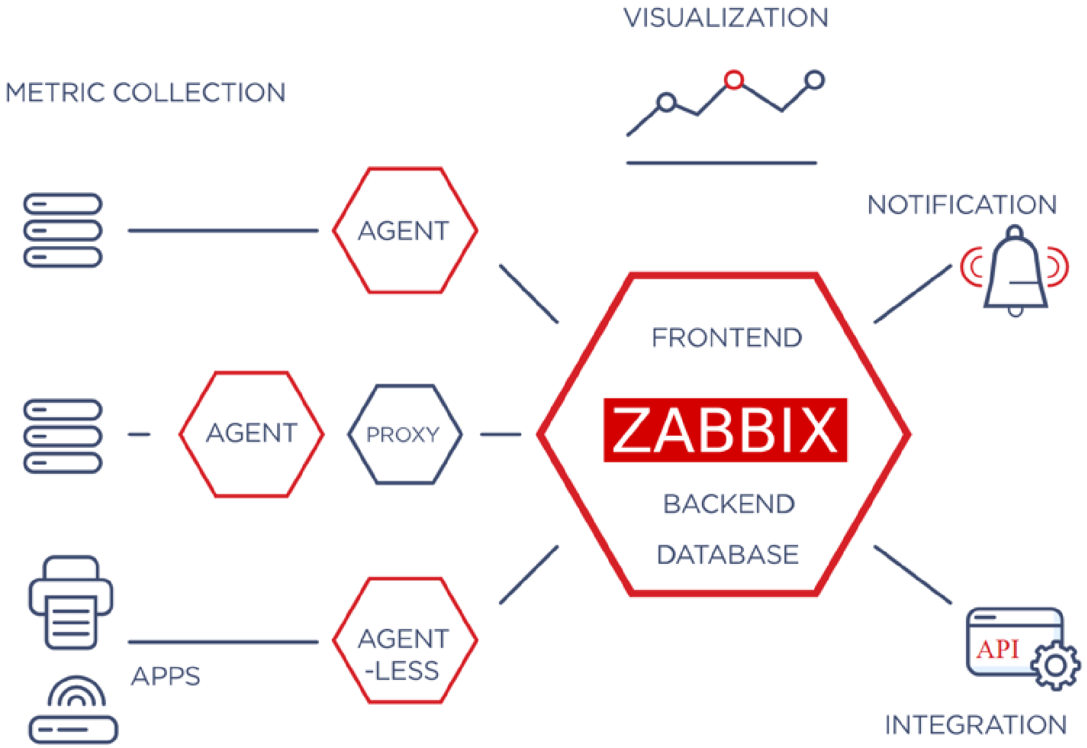
ZABBIX Technical Support Engineer

ZABBIX 2020 Conference BENELUX

zabbix

Zabbix - Reacting to problems

Without a doubt - Zabbix is THE BEST monitoring tool out there - many monitoring approaches can be used for different infrastructures



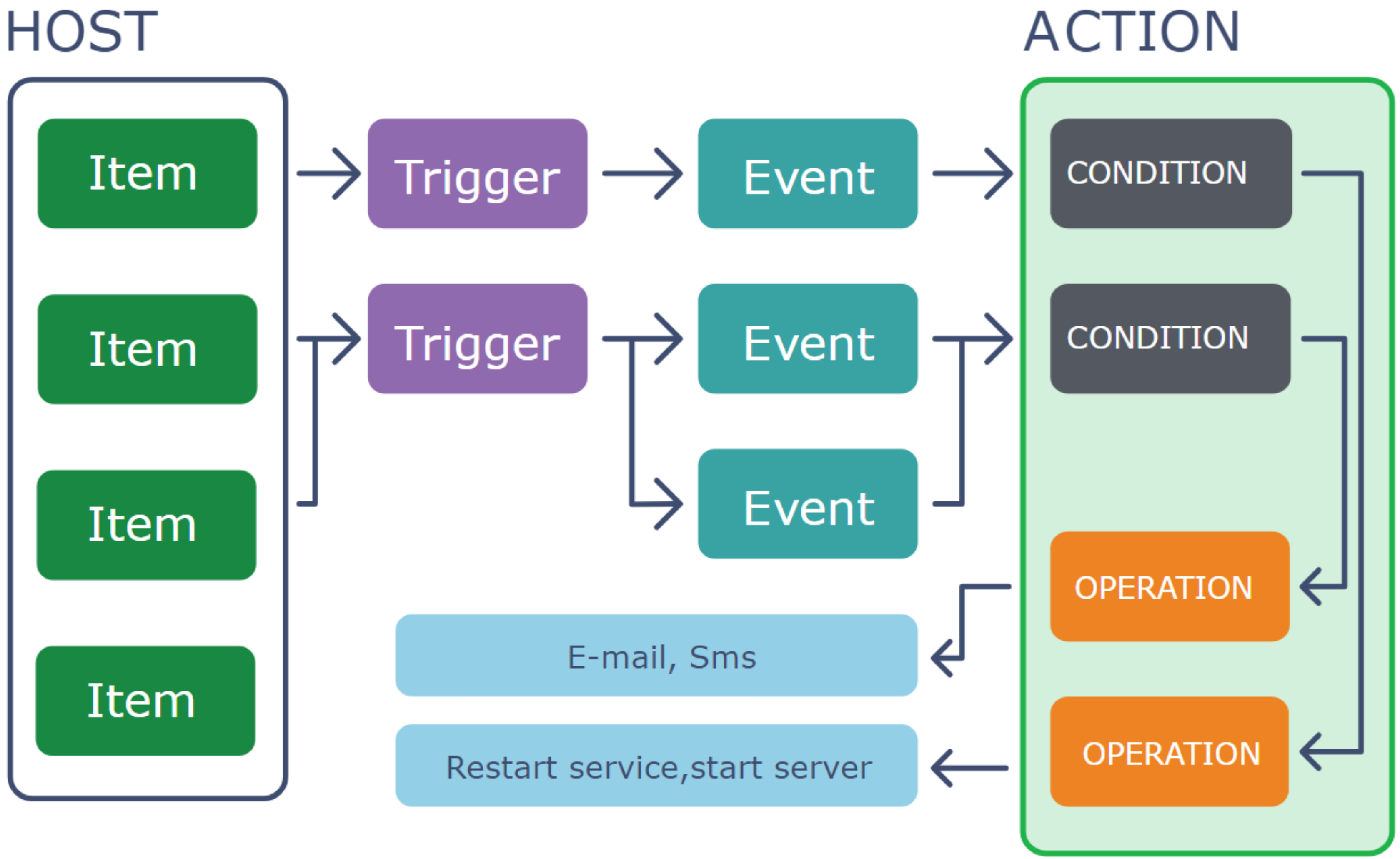
Zabbix - notifications

But monitoring is just the first step!

What can we do when there is a problem?

Zabbix - notifications

We define Actions which contain conditions and operations!



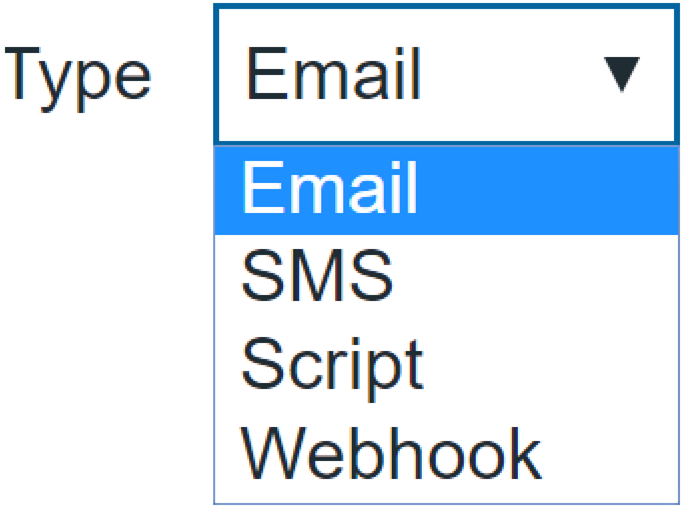
Media types

Media are the delivery channels used for sending notifications and alerts from Zabbix.

- Email
- SMS
- Script

And, as of 4.4:

- Webhook

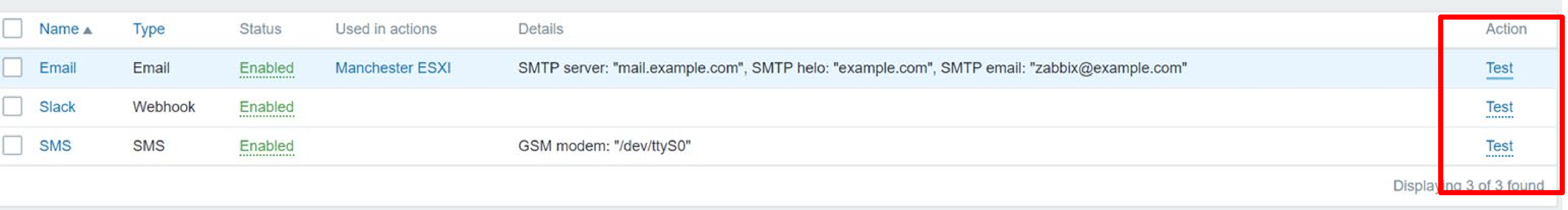


Type Email ▼

Email

React to events by notifying your team members via e-mail.

- As of 4.2 the message can either be HTML or Plain text
- Configure the SMTP parameters
- Create the media for your individual users
- Test it...
- ...Not by simulating a problem! Use the **Test** button! (new in 4.2)

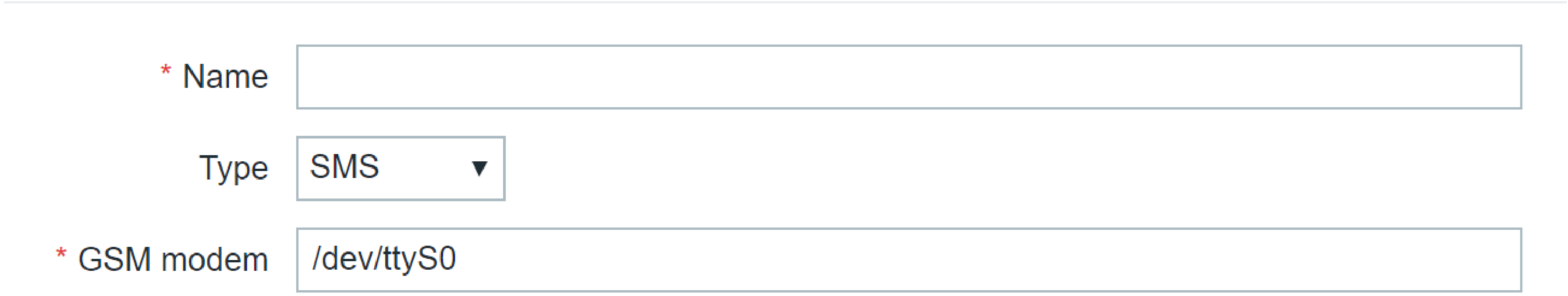


□	Name ▲	Type	Status	Used in actions	Details	Action
□	Email	Email	Enabled	Manchester ESXI	SMTP server: "mail.example.com", SMTP helo: "example.com", SMTP email: "zabbix@example.com"	Test
□	Slack	Webhook	Enabled			Test
□	SMS	SMS	Enabled		GSM modem: "/dev/ttyS0"	Test

Displaying 3 of 3 found

SMS

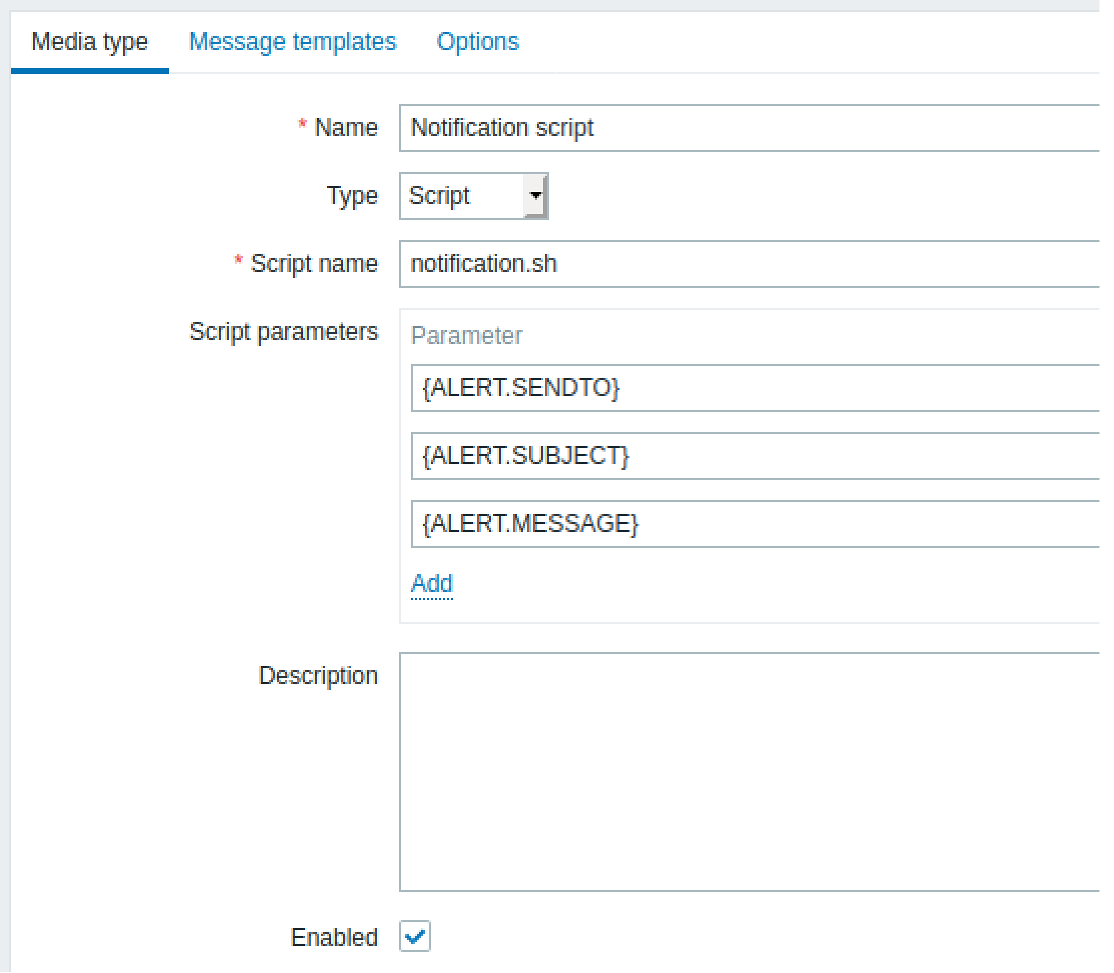
- Simple to configure
- Requires a GSM modem attached
- Tested models provided in the documentation
- Connected via a serial port or USB
- More complex configuration in case of a virtualized environment
- Create the media for your individual users



* Name	
Type	SMS ▾
* GSM modem	/dev/ttyS0

Alert script

- Executed by the Zabbix server
- Used to integrate Zabbix with 3rd party systems
- Can pass Macros as parameters
- Can be written in any language
- Examples available on Zabbix share or the Integrations section



* Name: Notification script

Type: Script

* Script name: notification.sh

Script parameters:

Parameter
{ALERT.SENDTO}
{ALERT.SUBJECT}
{ALERT.MESSAGE}

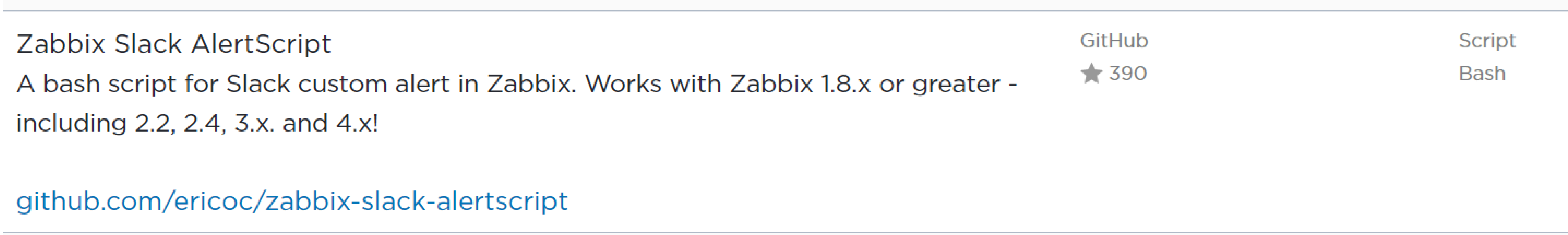
Description:

Enabled: ☑

Out of the box integrations

ZABBIX

And that was pretty much it for Zabbix 4.0! You were fully expected to create and use your own alert scripts for integration or use the community provided solutions.



Zabbix Slack AlertScript

A bash script for Slack custom alert in Zabbix. Works with Zabbix 1.8.x or greater - including 2.2, 2.4, 3.x. and 4.x!

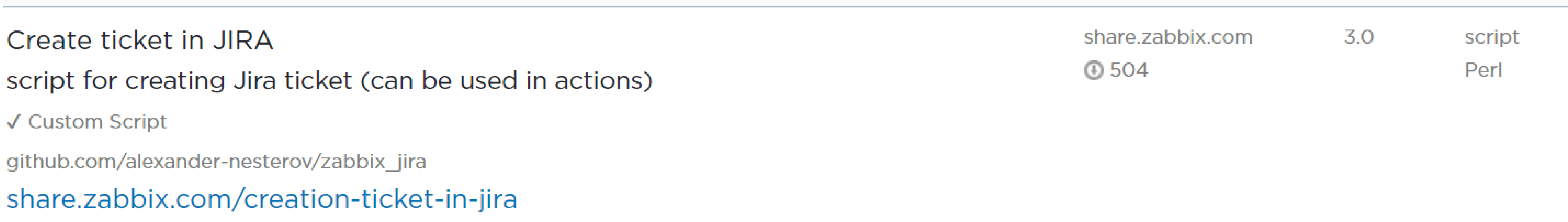
github.com/ericoc/zabbix-slack-alertscript

GitHub

★ 390

Script

Bash



Create ticket in JIRA

script for creating Jira ticket (can be used in actions)

✓ Custom Script

github.com/alexander-nesterov/zabbix_jira

share.zabbix.com/creation-ticket-in-jira

share.zabbix.com

504

3.0

script

Perl

Out of the box integrations

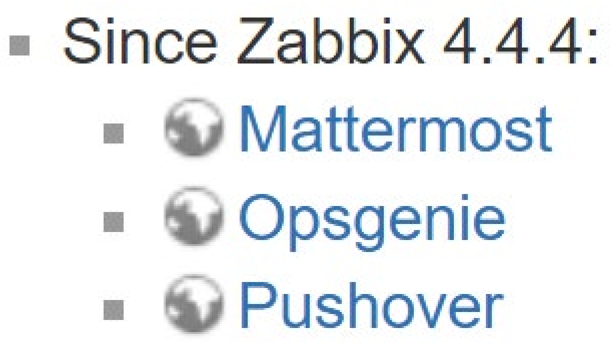
Problem:

How can we provide portable out of the box integrations?

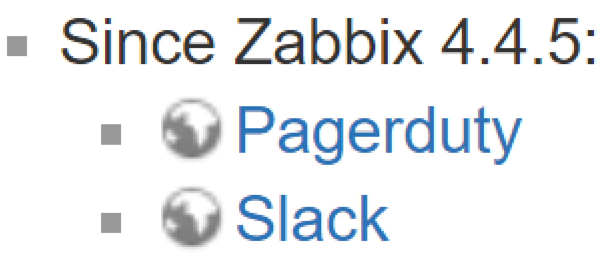
Solution:

Zabbix 4.2 - Support of JavaScript (in preprocessing)

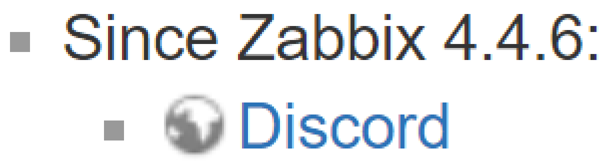
Zabbix 4.4 - Webhooks and programmable logic for actions and notifications



- Since Zabbix 4.4.4:
  - Mattermost
  - Opsgenie
  - Pushover



- Since Zabbix 4.4.5:
  - Pagerduty
  - Slack

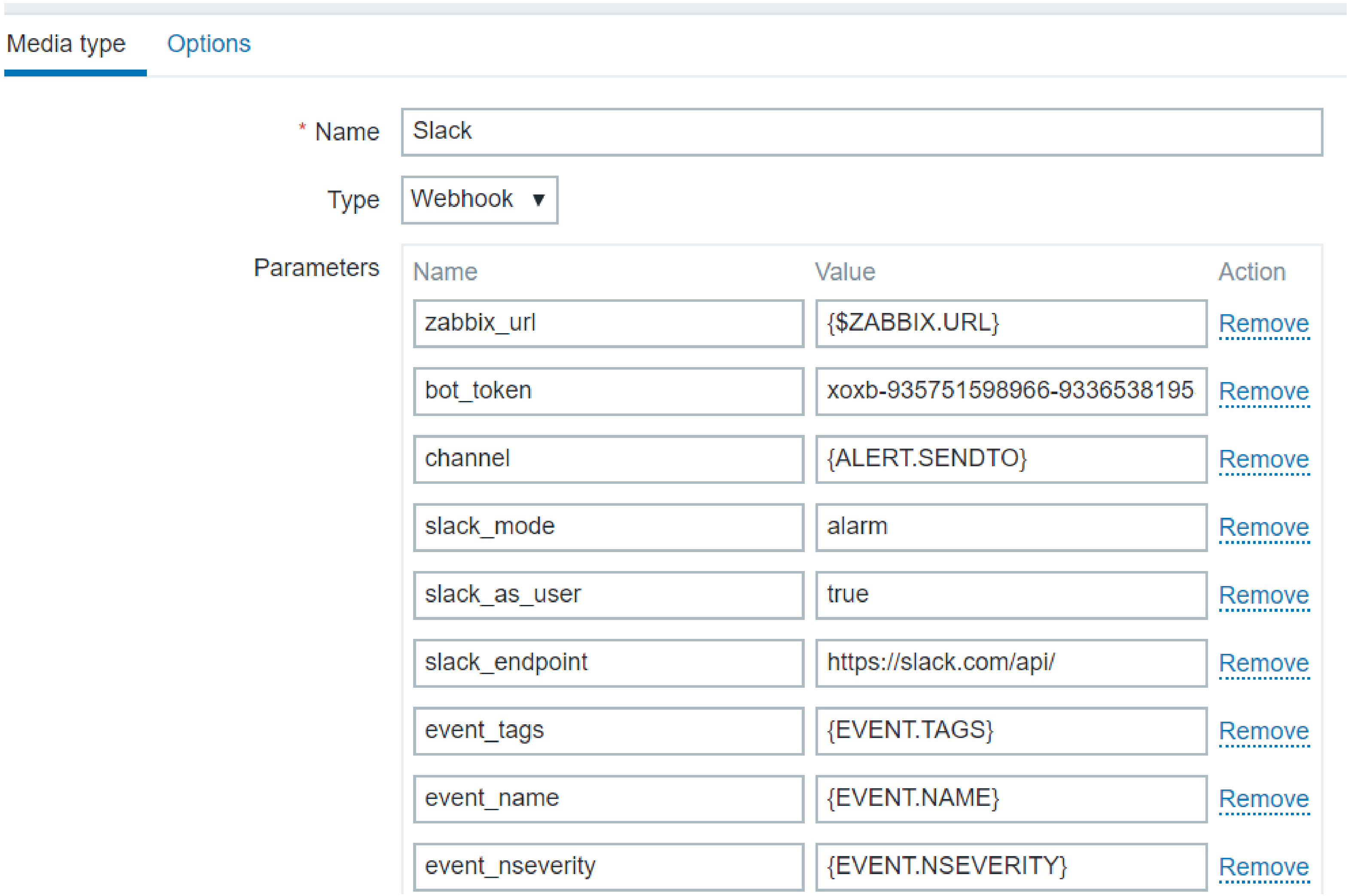


- Since Zabbix 4.4.6:
  - Discord

Webhooks

How do the webhooks actually work?

- Passes the event related parameters
- Uses the provided API token
- Specify the endpoint in the parameter

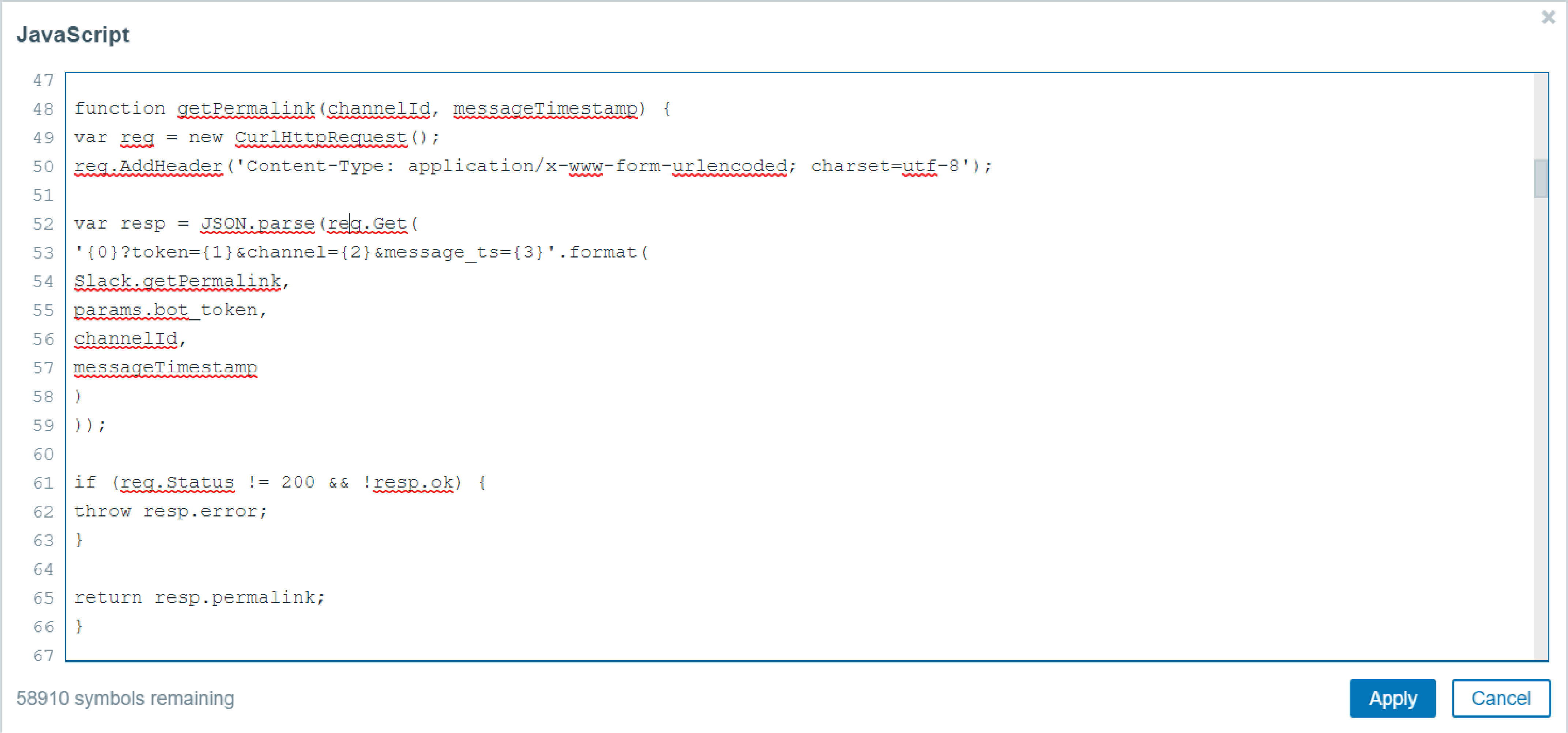


* Name	Slack
Type	Webhook ▾
Parameters	

Name	Value	Action
zabbix_url	{$ZABBIX.URL}	Remove
bot_token	xoxb-935751598966-9336538195	Remove
channel	{ALERT.SENDTO}	Remove
slack_mode	alarm	Remove
slack_as_user	true	Remove
slack_endpoint	https://slack.com/api/	Remove
event_tags	{EVENT.TAGS}	Remove
event_name	{EVENT.NAME}	Remove
event_nseverity	{EVENT.NSEVERITY}	Remove

Webhooks - script

The most important part of the webhook is the script:



JavaScript

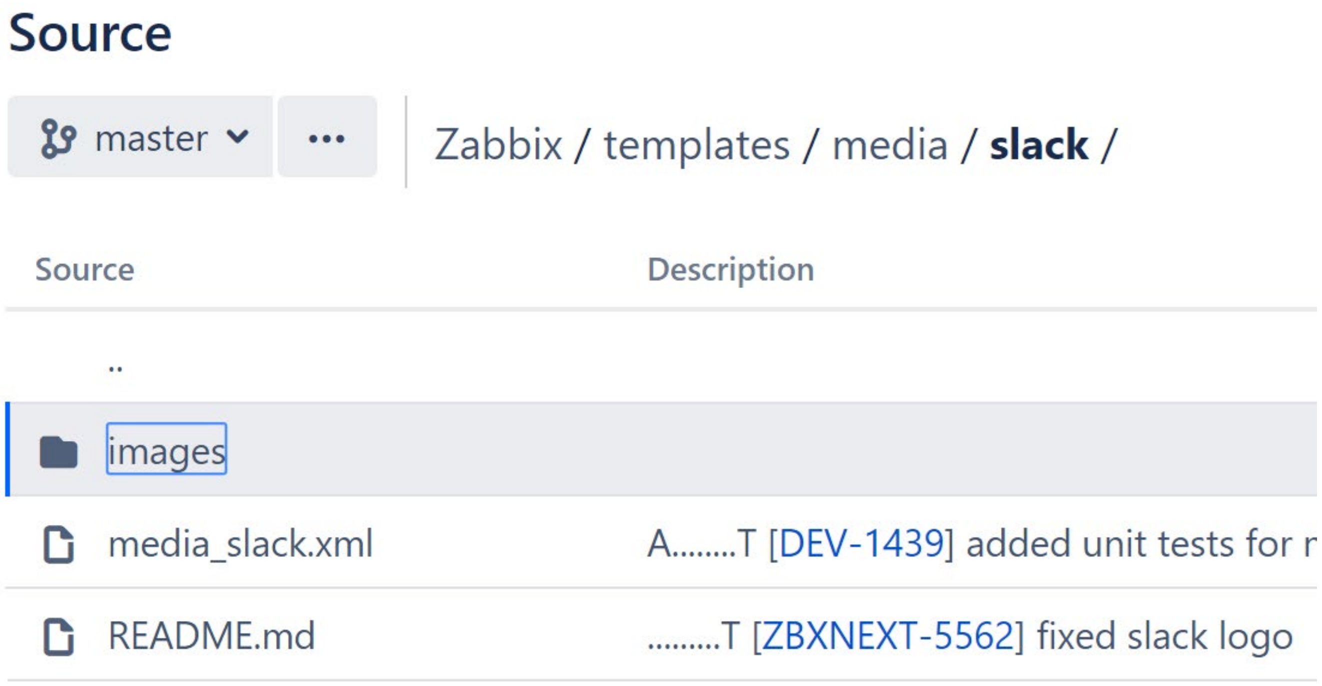
```
function getPermalink(channelId, messageTimestamp) {
var req = new CurlHttpRequest();
req.AddHeader('Content-Type: application/x-www-form-urlencoded; charset=utf-8');

var resp = JSON.parse(req.Get(
'{0}?token={1}&channel={2}&message_ts={3}'.format(
Slack.getPermalink,
params.bot_token,
channelId,
messageTimestamp
)
));

if (req.Status != 200 && !resp.ok) {
throw resp.error;
}

return resp.permalink;
}
```

58910 symbols remaining

Webhooks - script

- Oh, I still have to write the script though… :(

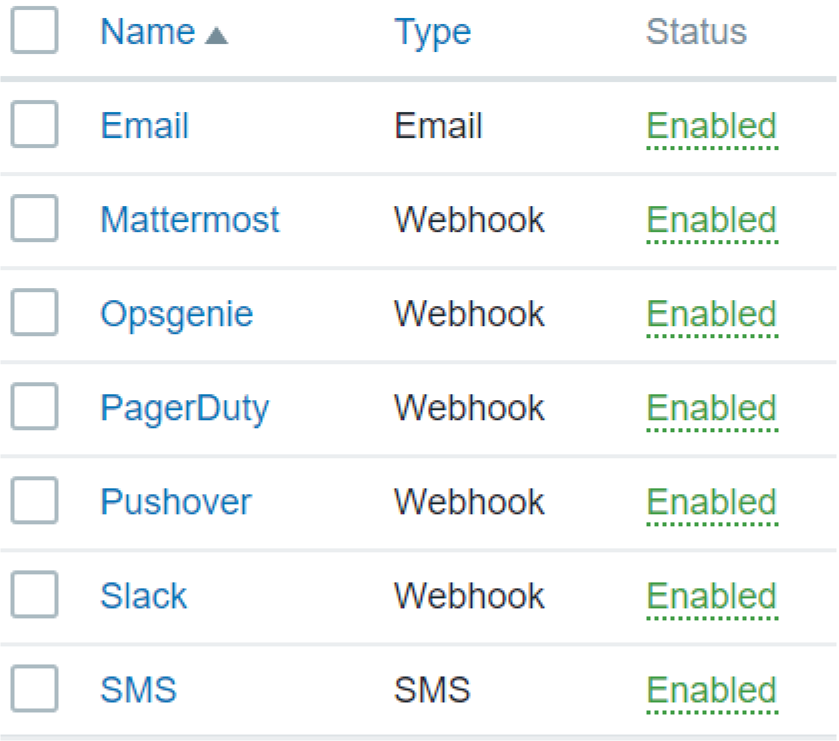
Not anymore! WE provide the script! It’s available out of the box or you can simply import it from our git repo!



Source

master ⌄ ··· | Zabbix / templates / media / **slack** /

Source	Description
..	
images	
media_slack.xml	A........T [DEV-1439] added unit tests for r
README.md	.........T [ZBXNEXT-5562] fixed slack logo



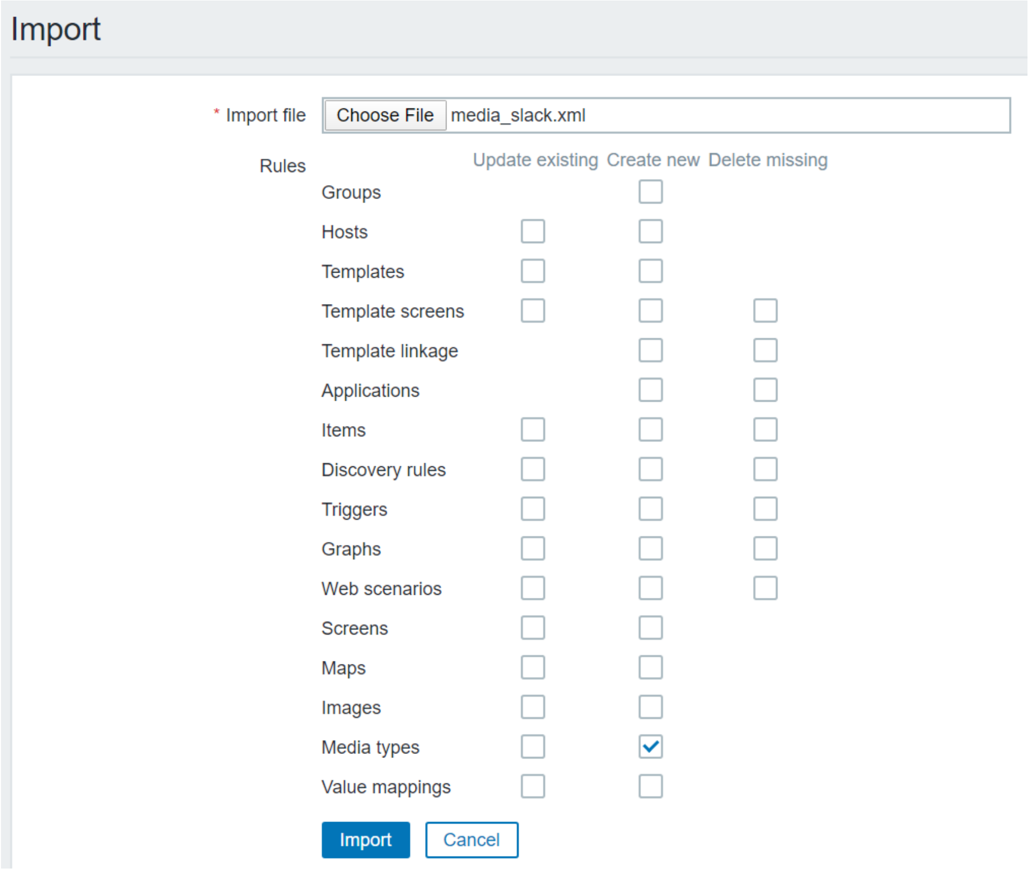
□	Name ▲	Type	Status
□	Email	Email	Enabled
□	Mattermost	Webhook	Enabled
□	Opsgenie	Webhook	Enabled
□	PagerDuty	Webhook	Enabled
□	Pushover	Webhook	Enabled
□	Slack	Webhook	Enabled
□	SMS	SMS	Enabled

Script vs Webhooks

ZABBIX

How else is this better than a custom script? It's Portable!

export/import the media type:



Import

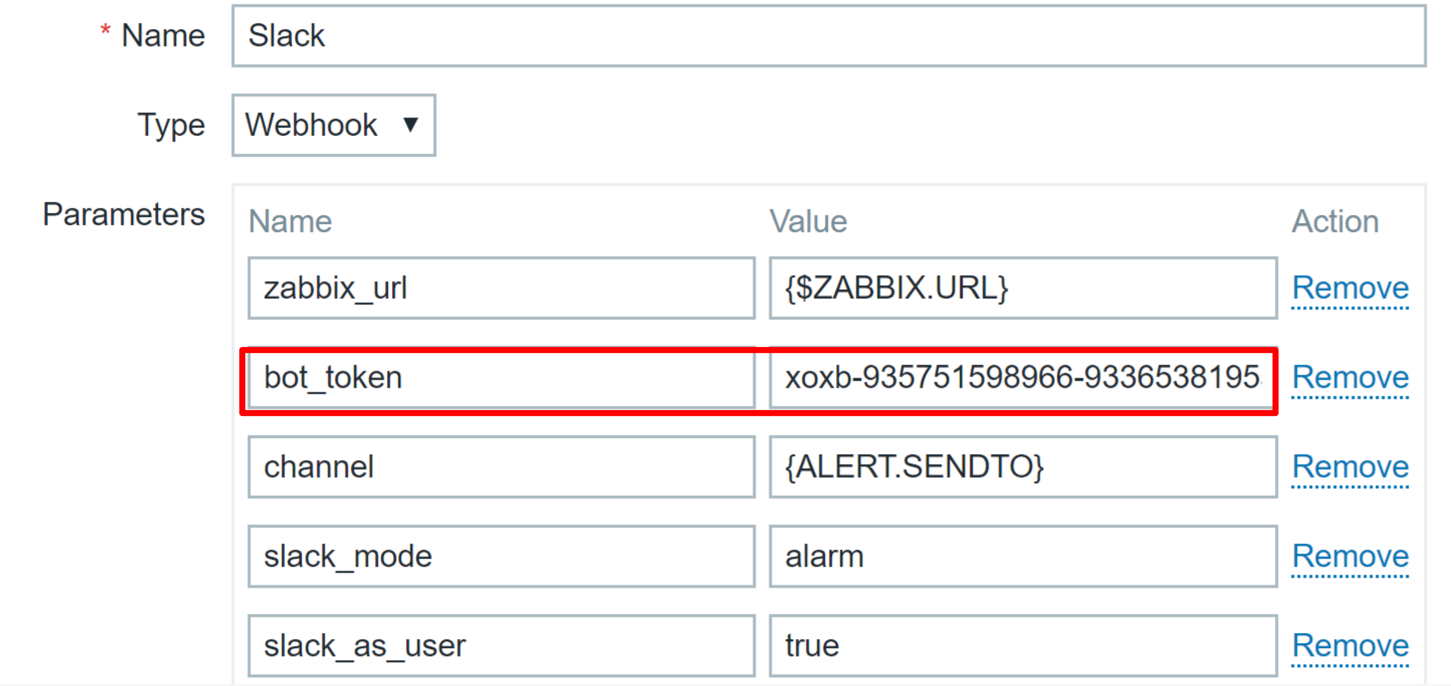
* Import file Choose File media_slack.xml

Rules

	Update existing	Create new	Delete missing
Groups		□	
Hosts	□	□	
Templates	□	□	
Template screens	□	□	□
Template linkage		□	□
Applications		□	□
Items	□	□	□
Discovery rules	□	□	□
Triggers	□	□	□
Graphs	□	□	□
Web scenarios	□	□	□
Screens	□	□	
Maps	□	□	
Images	□	□	
Media types	□	☑	
Value mappings	□	□	

Import Cancel

Copy your API token:



* Name: Slack

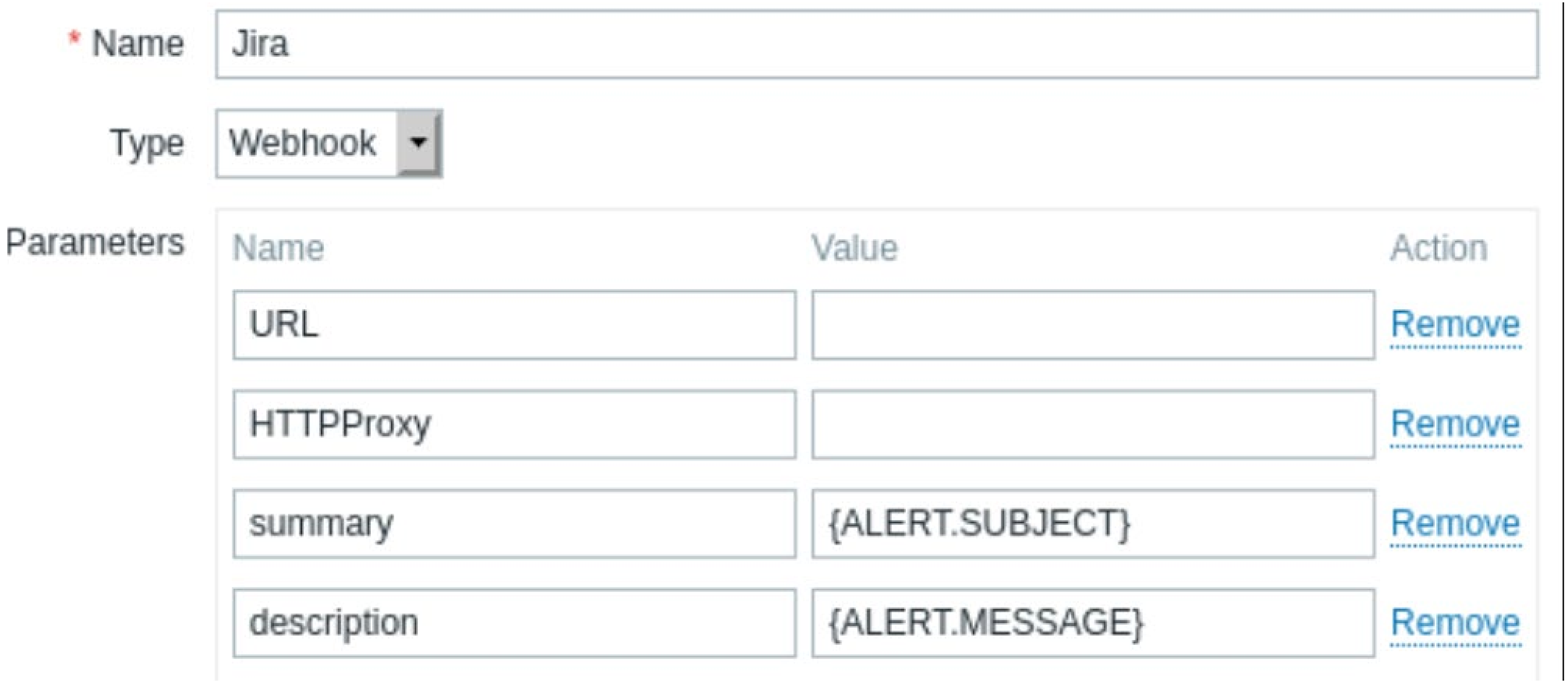
Type: Webhook ▼

Parameters

Name	Value	Action
zabbix_url	{$ZABBIX.URL}	Remove
bot_token	xoxb-935751598966-9336538195	Remove
channel	{ALERT.SENDTO}	Remove
slack_mode	alarm	Remove
slack_as_user	true	Remove

Webhooks - additional configuration

Starting with Zabbix 5.0 you can now specify HTTP Proxy as a parameter in Webhooks (Same logic as for HTTP agent item)



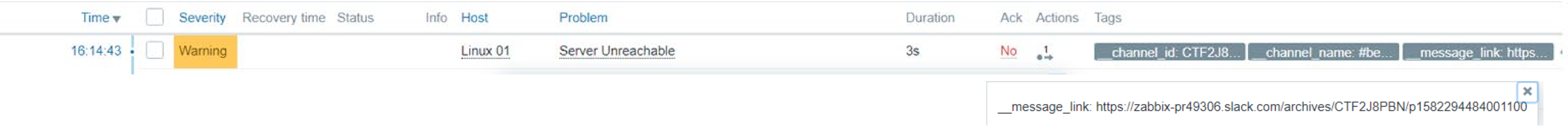
* Name	Jira
Type	Webhook

Parameters

Name	Value	Action
URL		Remove
HTTPProxy		Remove
summary	{ALERT.SUBJECT}	Remove
description	{ALERT.MESSAGE}	Remove

Webhooks - additional configuration

With Webhooks we can also populate our tags with data from the external system, for example:



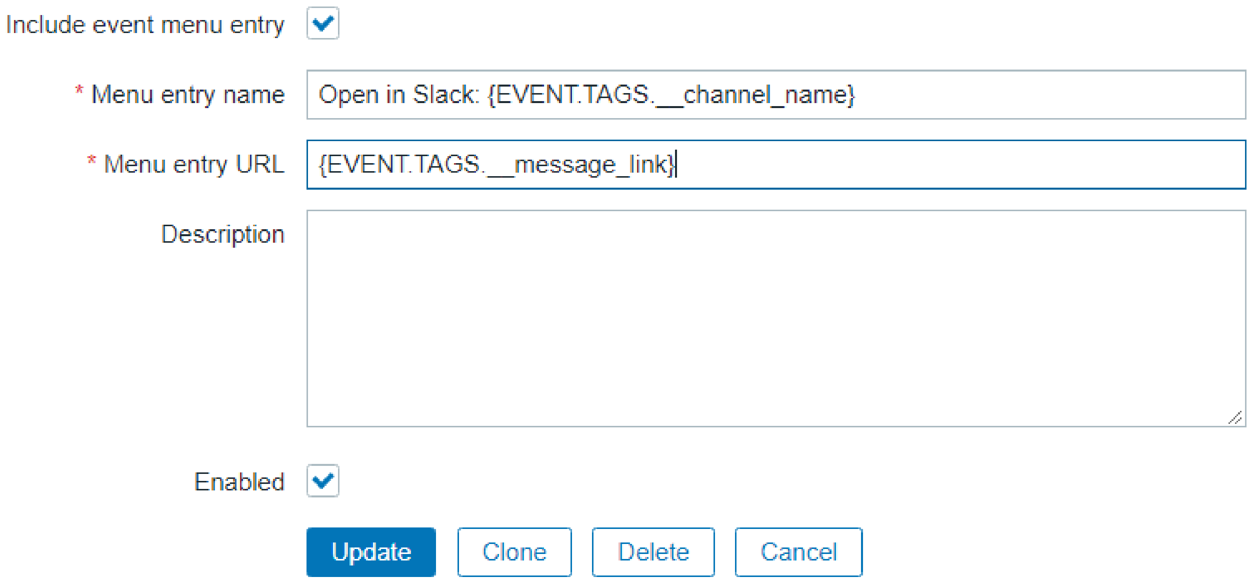
Time ▾	□	Severity	Recovery time	Status	Info	Host	Problem	Duration	Ack	Actions	Tags
16:14:43	□	Warning				Linux 01	Server Unreachable	3s	No	1	__channel_id: CTF2J8... __channel_name: #be... __message_link: https...

__message_link: https://zabbix-pr49306.slack.com/archives/CTF2J8PBN/p1582294484001100

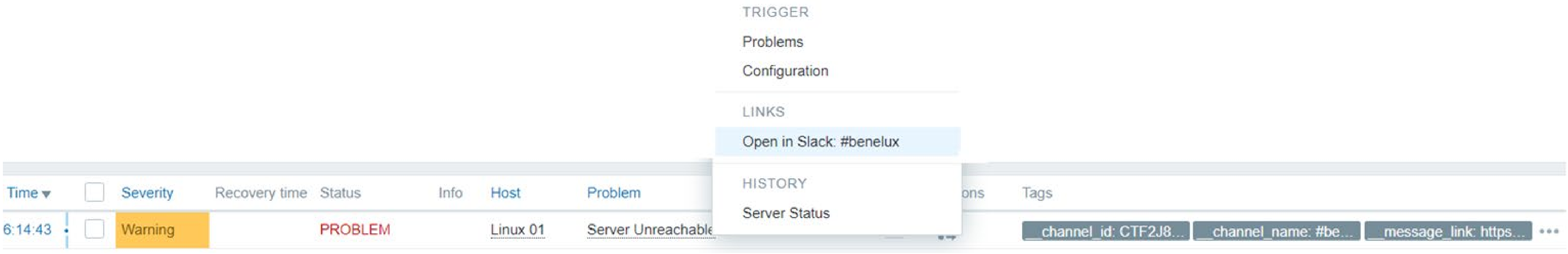
- Channel ID
- Channel name
- Message link
- And many more

Webhooks - additional configuration

You can add a direct link to the ticket:



Include event menu entry	☑
* Menu entry name	Open in Slack: {EVENT.TAGS.__channel_name}
* Menu entry URL	{EVENT.TAGS.__message_link}
Description	
Enabled	☑

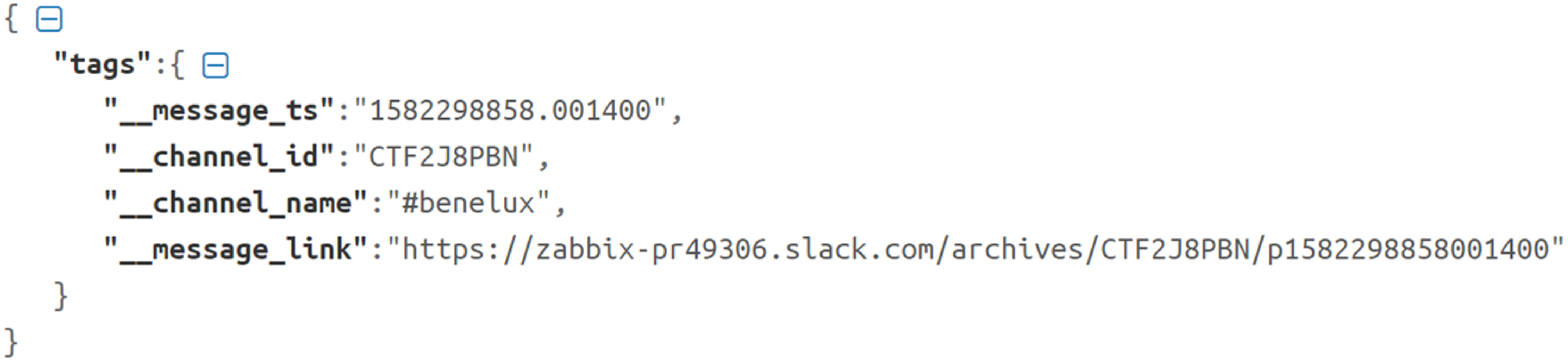


Time ▾	□	Severity	Recovery time	Status	Info	Host	Problem	ons	Tags
6:14:43	□	Warning		PROBLEM		Linux 01	Server Unreachable		__channel_id: CTF2J8... __channel_name: #be... __message_link: https... •••

Webhooks - tags

How are the tag values obtained?

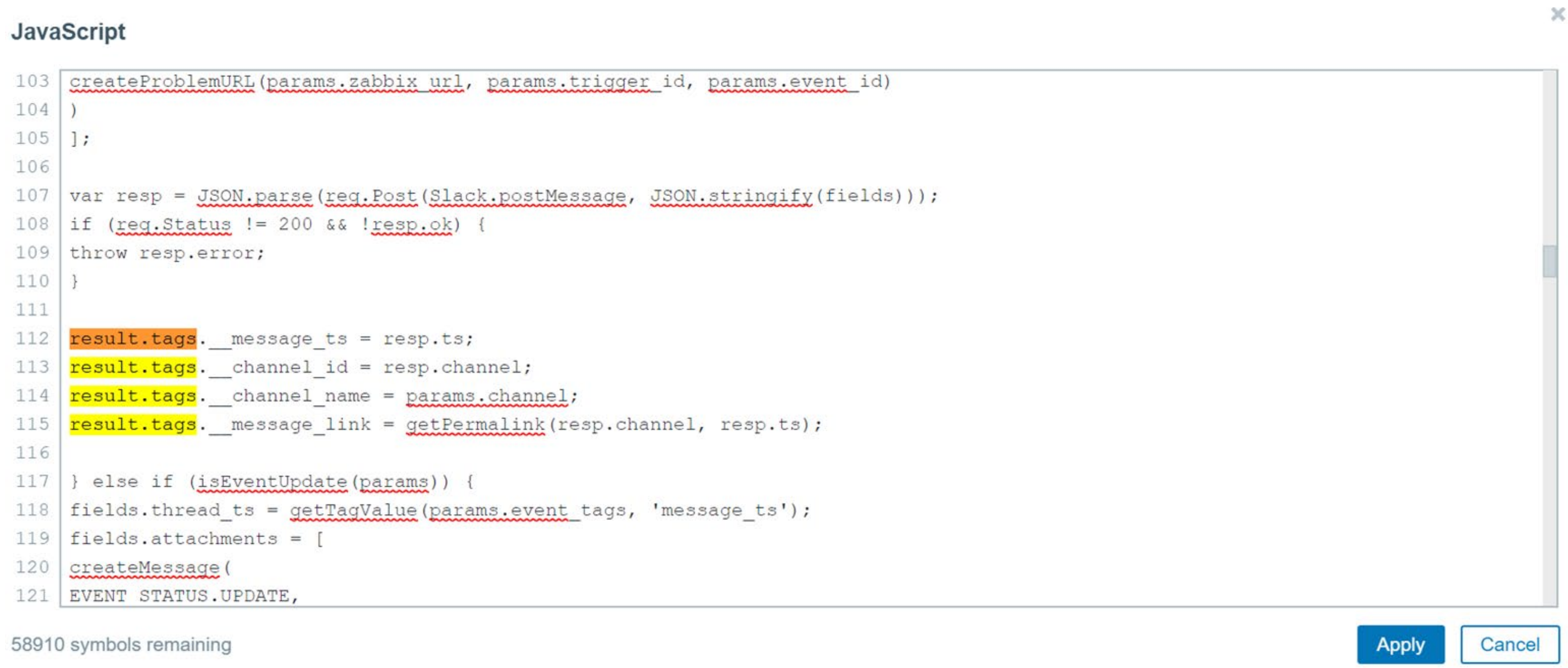
The returned JSON response should contain our tags:



```
{ ⊟
   "tags":{ ⊟
      "__message_ts":"1582298858.001400",
      "__channel_id":"CTF2J8PBN",
      "__channel_name":"#benelux",
      "__message_link":"https://zabbix-pr49306.slack.com/archives/CTF2J8PBN/p1582298858001400"
   }
}
```

Webhooks - tags

Once again - we should use JavaScript to format our returned data accordingly:



```
createProblemURL(params.zabbix_url, params.trigger_id, params.event_id)
)
];

var resp = JSON.parse(req.Post(Slack.postMessage, JSON.stringify(fields)));
if (req.Status != 200 && !resp.ok) {
throw resp.error;
}

result.tags.__message_ts = resp.ts;
result.tags.__channel_id = resp.channel;
result.tags.__channel_name = params.channel;
result.tags.__message_link = getPermalink(resp.channel, resp.ts);

} else if (isEventUpdate(params)) {
fields.thread_ts = getTagValue(params.event_tags, 'message_ts');
fields.attachments = [
createMessage(
EVENT_STATUS.UPDATE,
```

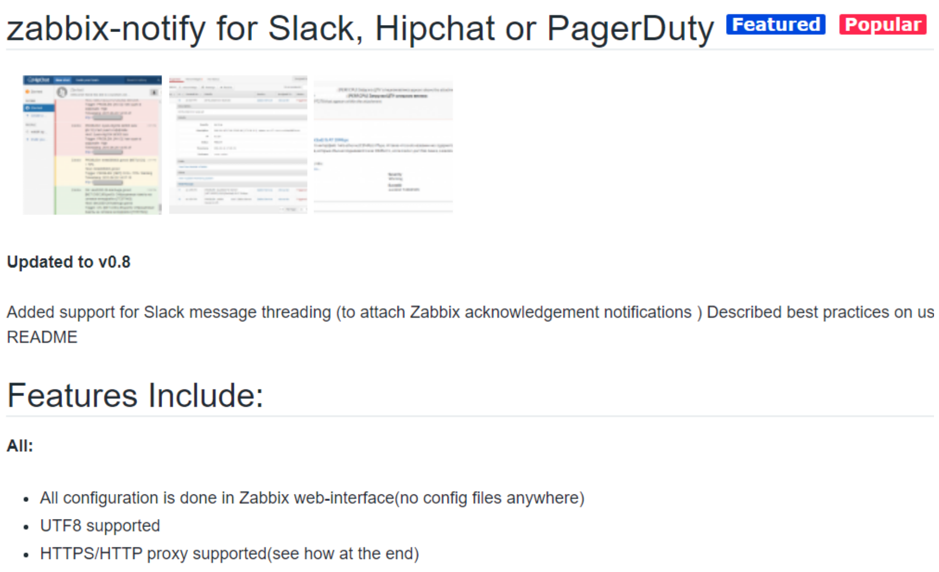
58910 symbols remaining

Integration example - legacy approach

Let’s take a look at the steps needed to implement Zabbix integration with Slack WITHOUT using Webhooks - the legacy way!

As an example we will use zabbix-notify. A solution available on our community share

- Featured integration
- Uses AlertScript
- Written in Perl



zabbix-notify for Slack, Hipchat or PagerDuty Featured Popular

Updated to v0.8

Added support for Slack message threading (to attach Zabbix acknowledgement notifications) Described best practices on us README

Features Include:

All:

- All configuration is done in Zabbix web-interface(no config files anywhere)
- UTF8 supported
- HTTPS/HTTP proxy supported(see how at the end)

Integration example - legacy approach

ZABBIX

- Step 1: Install the required modules and dependencies

We need to make sure that *LWP* and *JSON::XS* modules are installed.

The command depends on your favorite flavor of a package manager, for example:

```
yum install perl-JSON-XS perl-libwww-perl perl-LWP-Protocol-https perl-parent
```

Integration example - legacy approach

- Step 2: Download the .tar and install it into your system

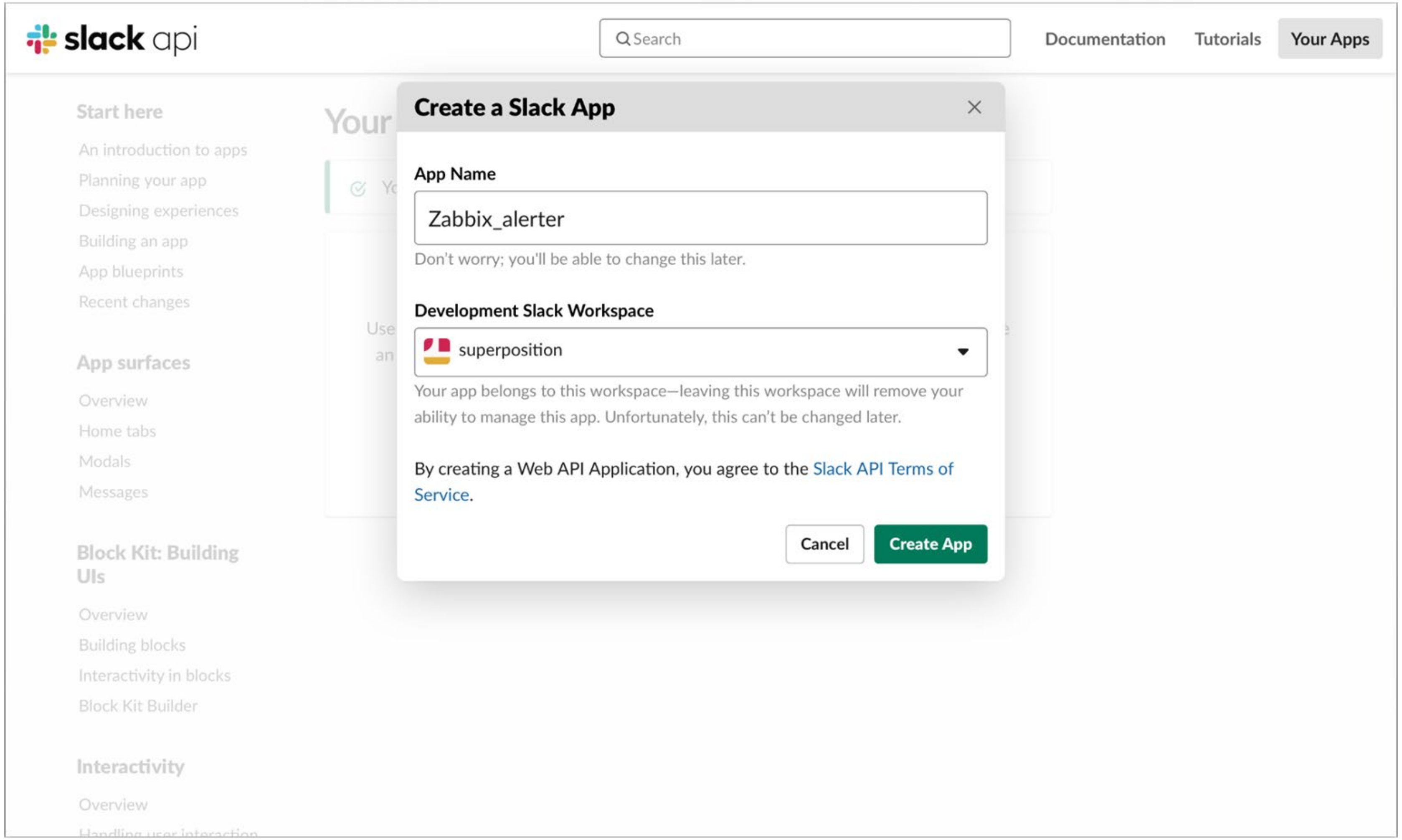
```
perl Makefile.PL INSTALLSITESCRIPT=/usr/local/share/zabbix/alertscripts
make test
make install
```

INSTALLSITESCRIPT should point towards your AlertScriptsPath directory, as per your Zabbix server configuration.

Integration example - legacy approach

ZABBIX

- Step 3: Create a slack Application



Create a Slack App

App Name: Zabbix_alerter

Don't worry; you'll be able to change this later.

Development Slack Workspace: superposition

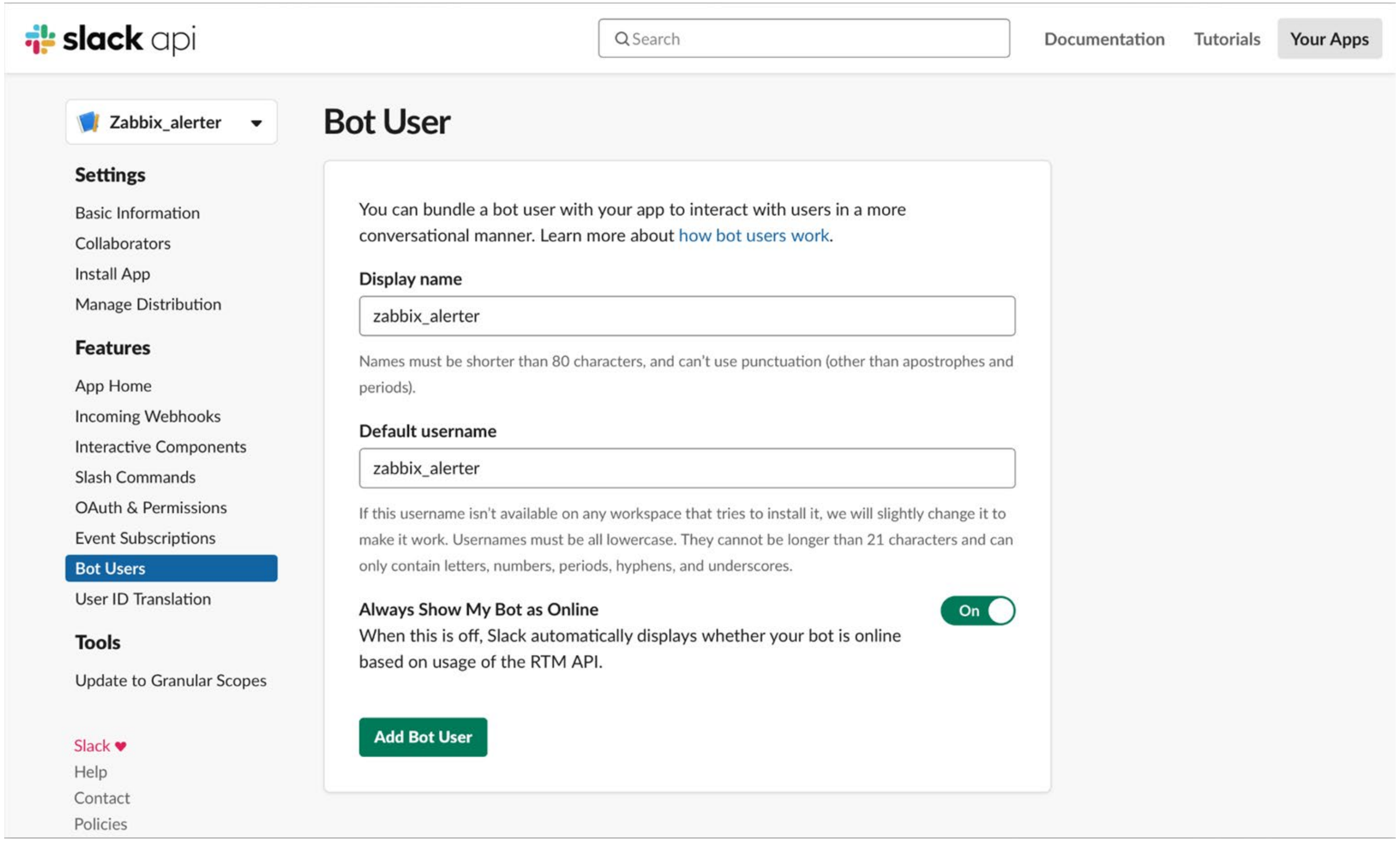
Your app belongs to this workspace—leaving this workspace will remove your ability to manage this app. Unfortunately, this can't be changed later.

By creating a Web API Application, you agree to the Slack API Terms of Service.

Cancel | Create App

Integration example - legacy approach

- Step 4: Create a slack bot user



Bot User

You can bundle a bot user with your app to interact with users in a more conversational manner. Learn more about how bot users work.

Display name

zabbix_alerter

Names must be shorter than 80 characters, and can't use punctuation (other than apostrophes and periods).

Default username

zabbix_alerter

If this username isn't available on any workspace that tries to install it, we will slightly change it to make it work. Usernames must be all lowercase. They cannot be longer than 21 characters and can only contain letters, numbers, periods, hyphens, and underscores.

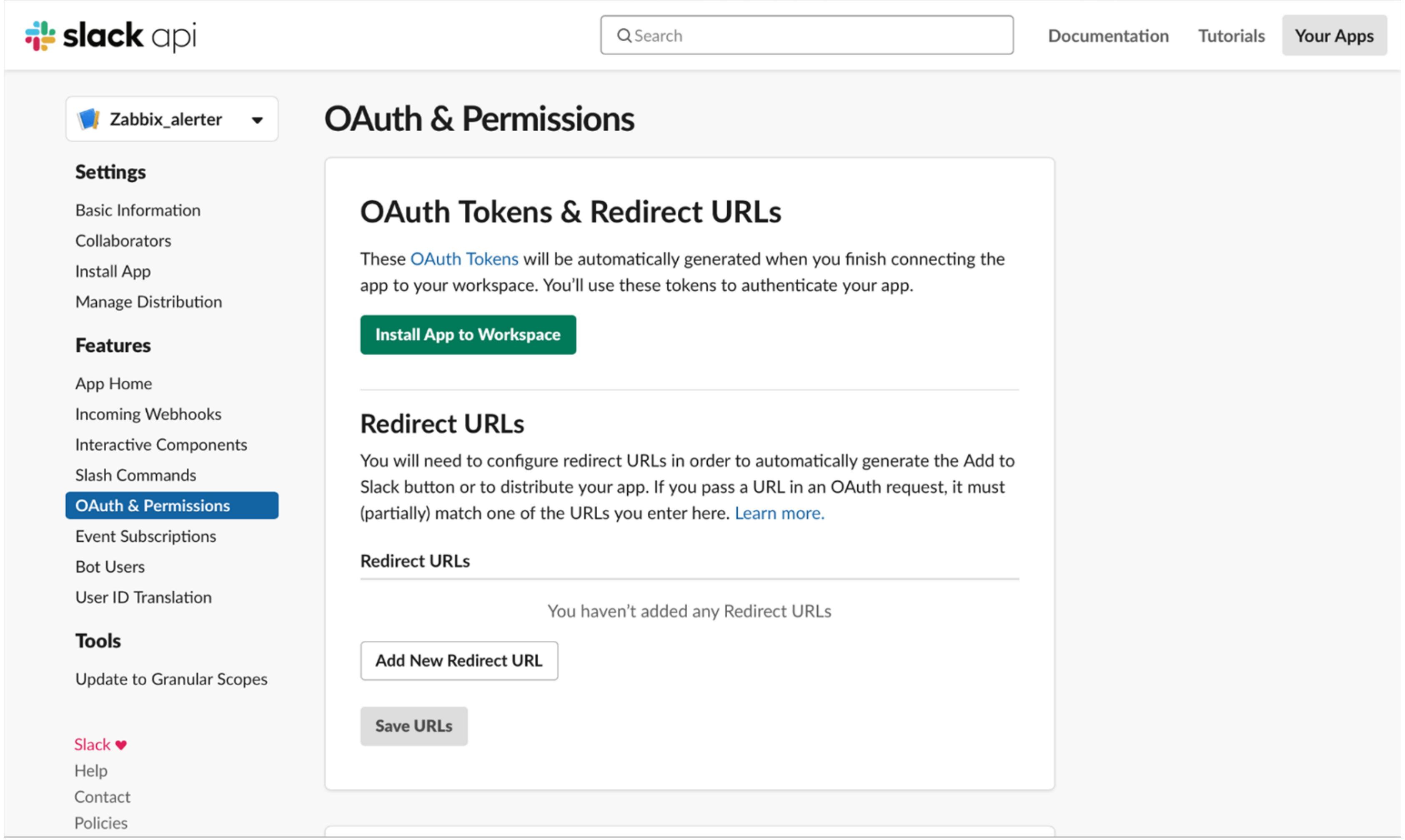
Always Show My Bot as Online On

When this is off, Slack automatically displays whether your bot is online based on usage of the RTM API.

Add Bot User

Integration example - legacy approach

- Step 5: Install app into your workspace



OAuth & Permissions

OAuth Tokens & Redirect URLs

These OAuth Tokens will be automatically generated when you finish connecting the app to your workspace. You'll use these tokens to authenticate your app.

Redirect URLs

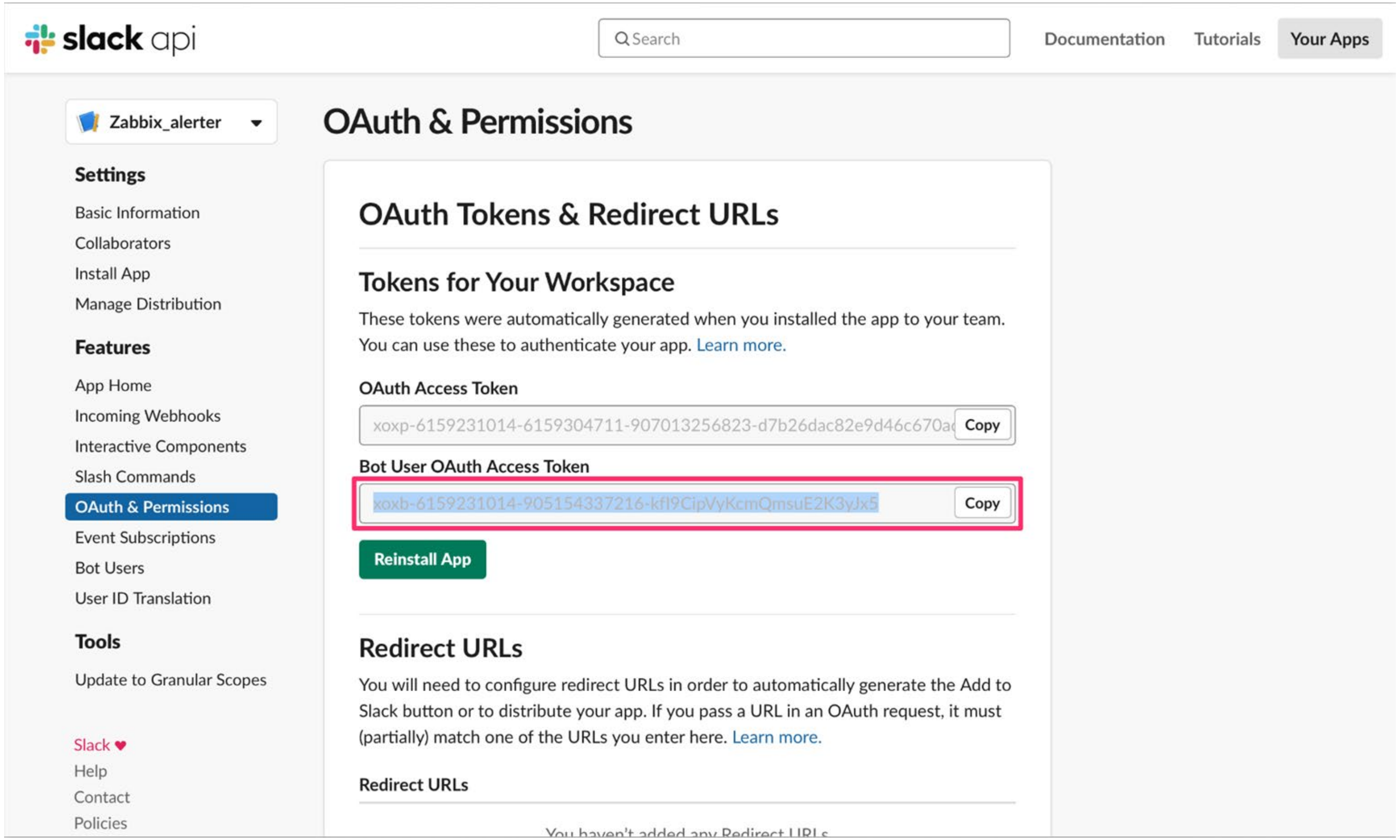
You will need to configure redirect URLs in order to automatically generate the Add to Slack button or to distribute your app. If you pass a URL in an OAuth request, it must (partially) match one of the URLs you enter here. Learn more.

Redirect URLs

You haven't added any Redirect URLs

Integration example - legacy approach

- Step 6: Obtain your bot token



OAuth & Permissions

OAuth Tokens & Redirect URLs

Tokens for Your Workspace

These tokens were automatically generated when you installed the app to your team. You can use these to authenticate your app. Learn more.

OAuth Access Token

xoxp-6159231014-6159304711-907013256823-d7b26dac82e9d46c670a Copy

Bot User OAuth Access Token

xoxb-6159231014-905154337216-kfl9CipVyKcmQmsuE2K3yJx5 Copy

Reinstall App

Redirect URLs

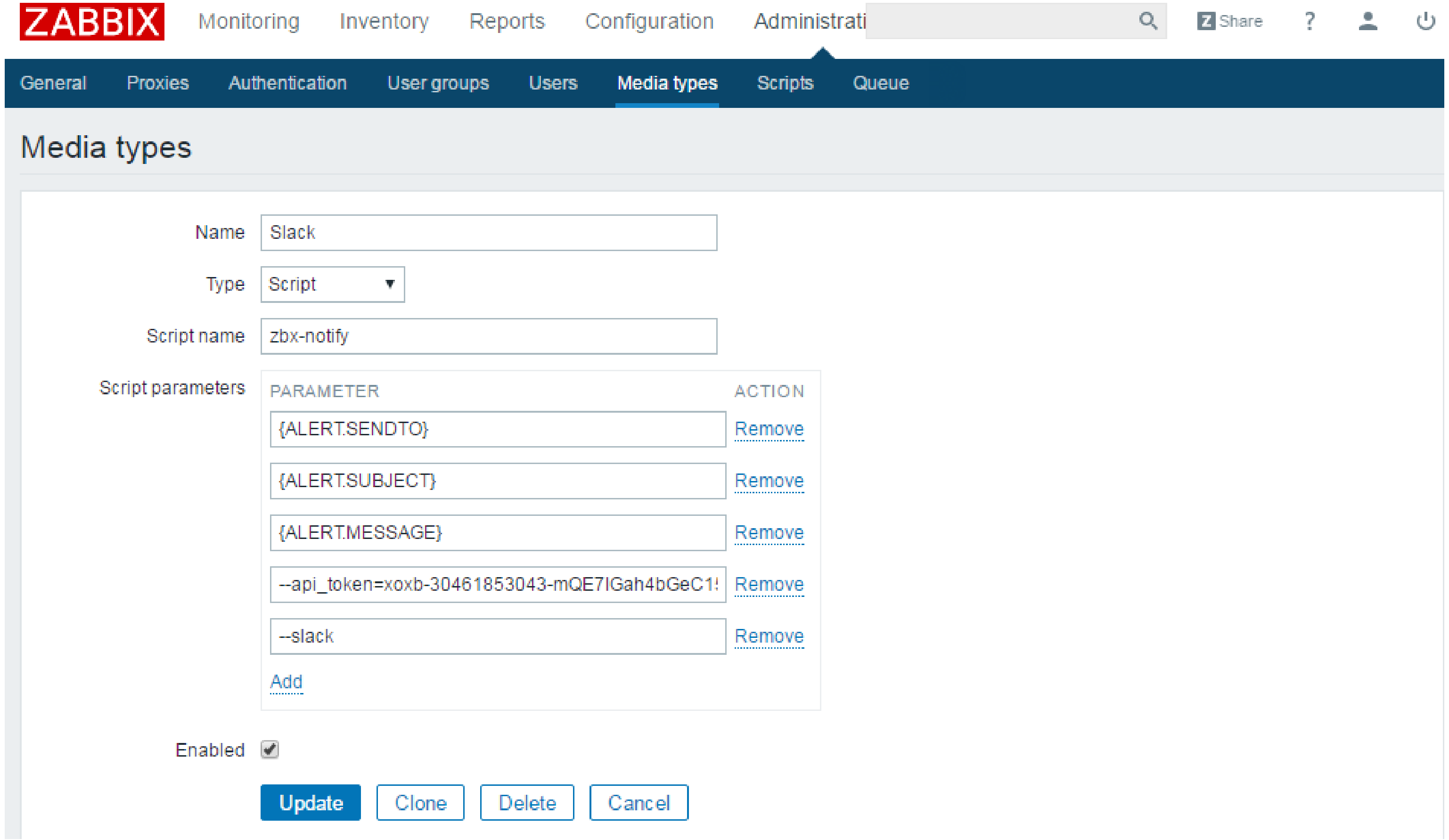
You will need to configure redirect URLs in order to automatically generate the Add to Slack button or to distribute your app. If you pass a URL in an OAuth request, it must (partially) match one of the URLs you enter here. Learn more.

Redirect URLs

You haven't added any Redirect URLs

Integration example - legacy approach

- Step 7: Create a new media type and provide the token



Media types

Name	Slack
Type	Script ▼
Script name	zbx-notify

Script parameters

PARAMETER	ACTION
{ALERT.SENDTO}	Remove
{ALERT.SUBJECT}	Remove
{ALERT.MESSAGE}	Remove
--api_token=xoxb-30461853043-mQE7IGah4bGeC1	Remove
--slack	Remove

Add

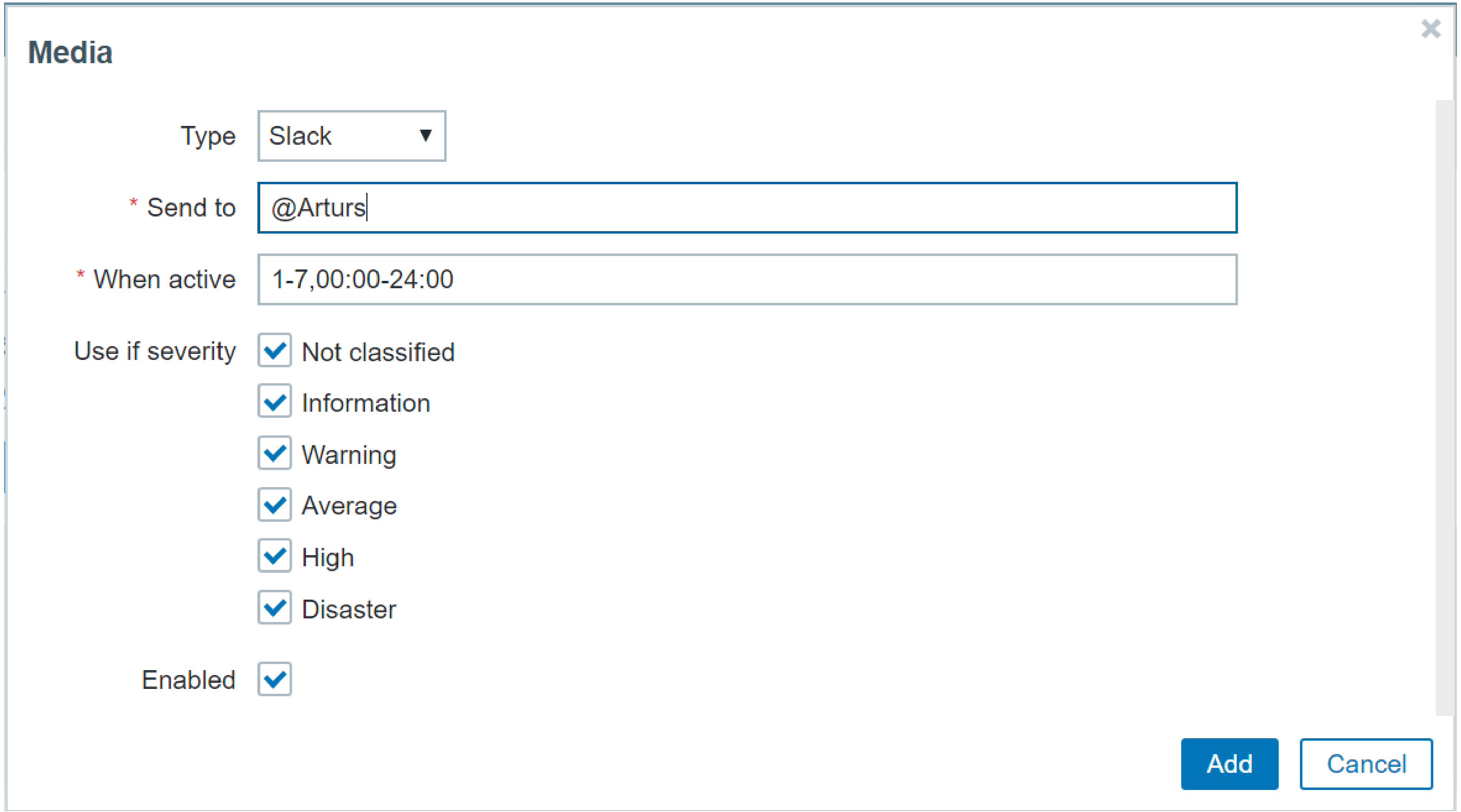
Enabled ☑

Update Clone Delete Cancel

Integration example - legacy approach

ZABBIX

- Step 8: Create a new media for your users



Media

Type: Slack ▾

* Send to: @Arturs

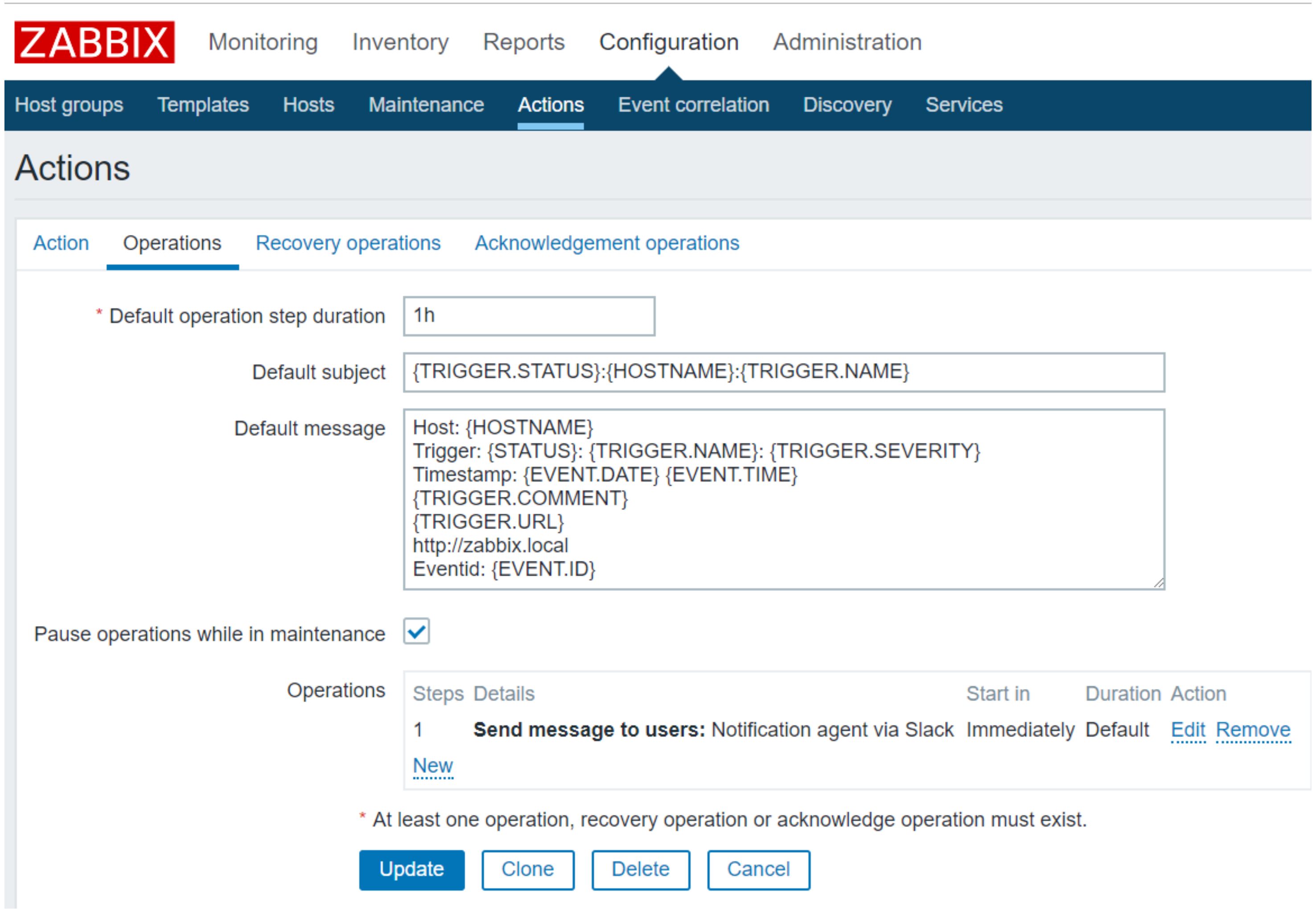
* When active: 1-7,00:00-24:00

Use if severity:
☑ Not classified
☑ Information
☑ Warning
☑ Average
☑ High
☑ Disaster

Enabled: ☑

Integration example - legacy approach

- Step 9: Create an action



Actions

* Default operation step duration: 1h

Default subject: {TRIGGER.STATUS}:{HOSTNAME}:{TRIGGER.NAME}

Default message:

```
Host: {HOSTNAME}
Trigger: {STATUS}: {TRIGGER.NAME}: {TRIGGER.SEVERITY}
Timestamp: {EVENT.DATE} {EVENT.TIME}
{TRIGGER.COMMENT}
{TRIGGER.URL}
http://zabbix.local
Eventid: {EVENT.ID}
```

Pause operations while in maintenance ☑

Operations

Steps	Details	Start in	Duration	Action
1	**Send message to users:** Notification agent via Slack	Immediately	Default	Edit Remove

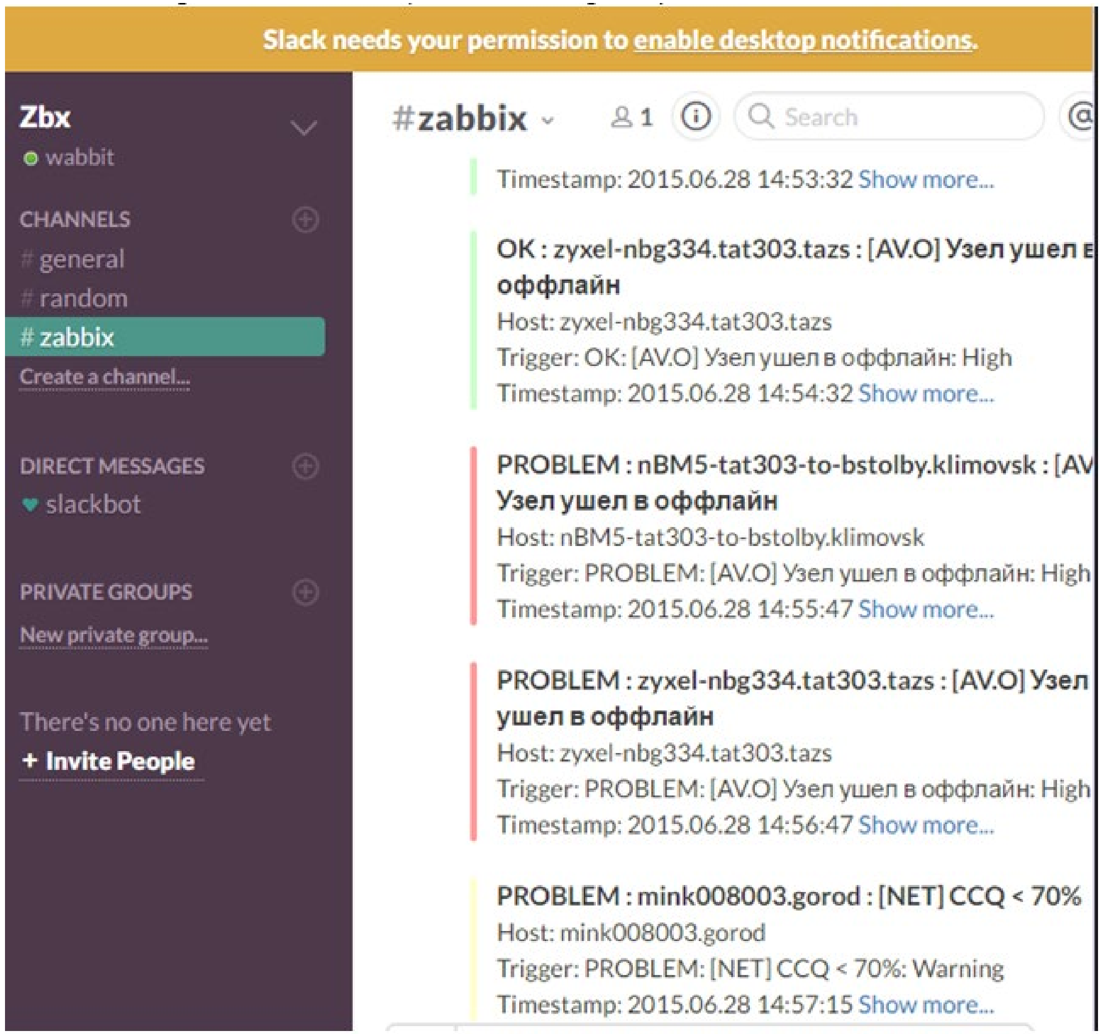
New

* At least one operation, recovery operation or acknowledge operation must exist.

Integration example - legacy approach

ZABBIX

- Step 10: Receive your slack alerts!



Timestamp: 2015.06.28 14:53:32 Show more...

OK : zyxel-nbg334.tat303.tazs : [AV.O] Узел ушел в оффлайн
Host: zyxel-nbg334.tat303.tazs
Trigger: OK: [AV.O] Узел ушел в оффлайн: High
Timestamp: 2015.06.28 14:54:32 Show more...

PROBLEM : nBM5-tat303-to-bstolby.klimovsk : [AV Узел ушел в оффлайн
Host: nBM5-tat303-to-bstolby.klimovsk
Trigger: PROBLEM: [AV.O] Узел ушел в оффлайн: High
Timestamp: 2015.06.28 14:55:47 Show more...

PROBLEM : zyxel-nbg334.tat303.tazs : [AV.O] Узел ушел в оффлайн
Host: zyxel-nbg334.tat303.tazs
Trigger: PROBLEM: [AV.O] Узел ушел в оффлайн: High
Timestamp: 2015.06.28 14:56:47 Show more...

PROBLEM : mink008003.gorod : [NET] CCQ < 70%
Host: mink008003.gorod
Trigger: PROBLEM: [NET] CCQ < 70%: Warning
Timestamp: 2015.06.28 14:57:15 Show more...

Integration example - legacy approach

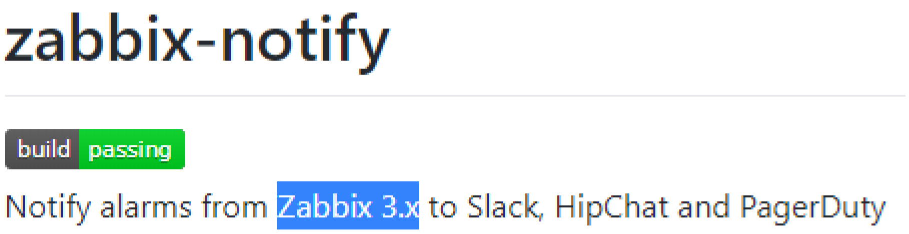
Let’s summarize the additional tasks we had to do to implement such an integration:

- Download and configure the script
- Install custom modules
- Test the integration manually

Some compatibility issues may also be encountered!

Integration example - legacy approach

- No guaranteed long-term support

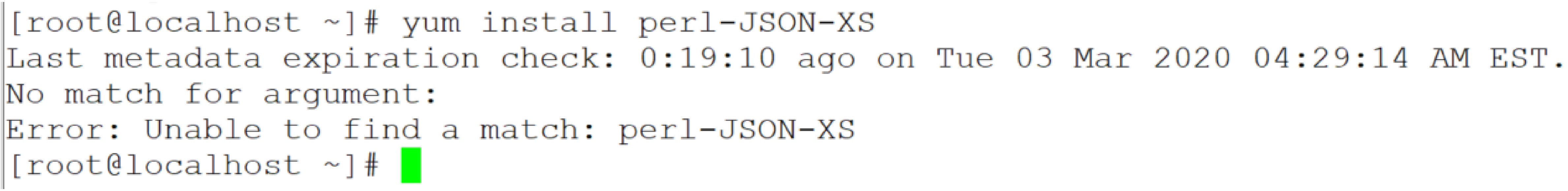


zabbix-notify

build passing

Notify alarms from Zabbix 3.x to Slack, HipChat and PagerDuty

- Possible dependency/compilation issues based on the underliying OS



```
[root@localhost ~]# yum install perl-JSON-XS
Last metadata expiration check: 0:19:10 ago on Tue 03 Mar 2020 04:29:14 AM EST.
No match for argument:
Error: Unable to find a match: perl-JSON-XS
[root@localhost ~]#
```

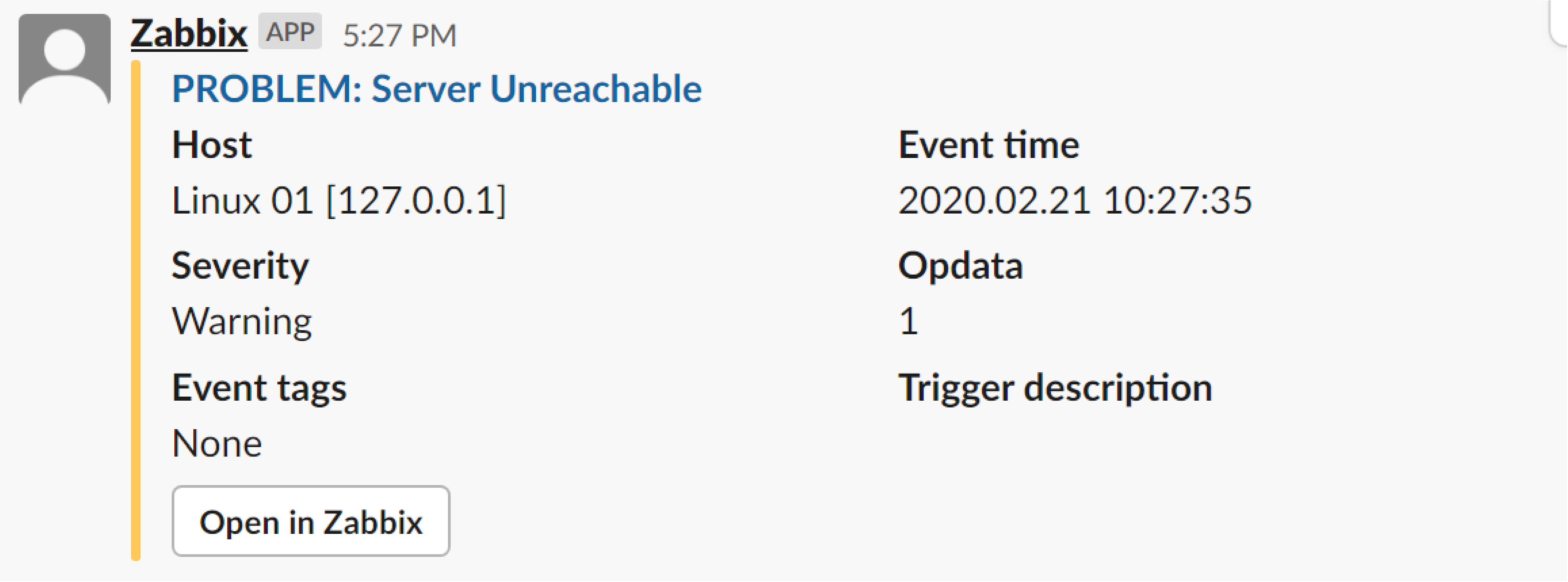
Integration example - Webhooks

Now, let’s look at our official Slack integration!

- Step 1: Repeat steps 3-6 to create a slack Application and obtain the bot user token!
- Step 2: Create a {$ZABBIX.URL} macro which will contain a link to your Zabbix instance
- Step 3: Import the media type
- Step 4: set bot_token to the previously created token
- Step 5: Create a media for the corresponding user

Integration example - Webhooks

And you’re done! Let’s look at our Slack alert:



Zabbix APP 5:27 PM

PROBLEM: Server Unreachable

Host
Linux 01 [127.0.0.1]

Event time
2020.02.21 10:27:35

Severity
Warning

Opdata
1

Event tags
None

Trigger description

Integration example - Webhooks

ZABBIX

While the amount of performed steps is very similar, the integration was quite a bit more streamlined:

- No additional module or script download necessary
- The script was contained in the imported integration
- Really easy to do even for users not familiar with Zabbix configuration
- Fully done within Zabbix and Slack, no need to execute any commands on the OS!
- Guaranteed future support and additional updates!

QUESTIONS?
THANK YOU!

Arturs Lontons

ZABBIX Technical Support Engineer

ZABBIX 2020 Conference BENELUX

zabbix