

OPENING AND CLOSING TICKETS ON OTRS USING ZABBIX

1. Name: Opening and closing OTRS tickets with Zabbix.
2. Introduction: We will integrate Zabbix with OTRS via web service.
3. Prerequisites: To perform the call opening procedure you must:
 - 1) Installation of python on zabbix server.
 - 2) Installing python-pip on zabbix server.
 - 3) Installing the otrs-python module on the zabbix server.
 - 4) Installing CPAN on the zabbix server.
 - 5) installing the OTRS modules on the zabbix server.
 - 6) Opening script create_chamado_zabbix.py.
 - 7) Closing script close_chamado_zabbix.pl.
 - 8) ZabbixOTRS.yml file.
 - 9) Importing the webservice should be done.
 - 10) Carry out the following information:
 - a) IP of the OTRS server.
 - b) User and password to connect in the other.
 - c) Client in which the call will be opened.
 - d) Title of the call.
 - e) Description of the call.
 - f) Filename registered in the OTRS.
 - g) Service registered in the OTRS.
 - h) Level of SLA registered in the OTRS.
4. Installation procedure:
 - 1) Installation of python and python pip
 - a) yum install python python-pip -y
 - 2) Installing python-otrs with pip
 - a) pip install python-otrs
 - 3) CPAN Installation
 - a) yum install cpan -y
 - 4) Installing the other modules
 - a) scp /opt/otrs/bin/otrs.CheckModules.pl root @ zabbix: / tmp /
 - b) perl -MCPAN -e otrs.CheckModules.pl
 - 5) Installing perl modules.
 - a) yum install "perl(ExtUtils::MakeMaker)" "perl(Sys::Syslog)" -y
 - b) yum install "perl(Archive::Tar)" "perl(Archive::Zip)" "perl(Crypt::Eksblowfish::Bcrypt)" "perl(Crypt::SSLeay)" "perl(Date::Format)" "perl(DBD::Pg)" "perl(Encode::HanExtra)" "perl(IO::Socket::SSL)" "perl(JSON::XS)" "perl(Mail::IMAPClient)" "perl(IO::Socket::SSL)" "perl(ModPerl::Util)" "perl(Net::DNS)" "perl(Net::LDAP)" "perl(Template)" "perl(Template::Stash::XS)" "perl(Text::CSV_XS)" "perl(Time::Piece)" "perl(XML::LibXML)" "perl(XML::LibXSLT)" "perl(XML::Parser)" "perl(YAML::XS)" -y
 - c) yum install perl-SOAP-Lite -y
 - 6) Copy the file "criar_chamado_zabbix.py" to the Zabbix server.
 - a) cp /tmp/criar_chamado_zabbix.py /usr/lib/zabbix/alertscripts/
 - 7) Copy the file "fechar_chamado_zabbix.pl" to the Zabbix server.
 - a) cp /tmp/fechar_chamado_zabbix.pl /usr/lib/zabbix/alertscripts/
 - 8) Importing the web service into OTRS.
 - a) Access the OTRS with an admin account.
 - b) Click on the tab "Administration".
 - c) Under "System Administration" click on "Web Services".
 - d) Under "Actions" click on "Add Web Server".
 - e) Again in actions click on "Import Web Service".
 - f) Choose the file "ZabbixOTRS.yml" and click on "Import".
 - 9) Run the "criar_chamado_zabbix.py" script with the --help parameter
o check the items to be added.

5. Handling the files:

- 1) In the file "criar_chamado_zabbix.py" you should check the call opening Information.
 - a) "Ticket(State='Aberto'" and "Priority='3 normal'" on line 38
 - b) "Type='Incidente'" on line 40
 - c) "Ticket(State='Em atendimento')" on line 49
- 2) In the file "fechar_chamado_zabbix.pl" you should be aware of the following information.
 - a) "\$UserLogin" on line 22.
 - b) "\$Password" on line 23.
 - c) "my \$URL" on line 42.
 - d) "<State>resolvido</State>" in line 126 and 189.

```
[root@Zabbix alertscripts]# python criar_chamado_zabbix.py --help
usage: criar_chamado_zabbix.py [-h] [--otrs SERVER] [--webservice WEBSRV]
                                [--user USER] [--pass PASSWORD]
                                [--customer CUSTOMER] [--title TITLE]
                                [--desc DESCRICAO] [--fila FILA]
                                [--servico SERVICO] [--sla SLA]
                                [--triggerid TGRID] [--host HOST]
                                [--status STATUS]

Criar um ticket.

optional arguments:
  -h, --help            show this help message and exit
  --otrs SERVER          OTRS server address, ex: 10.20.19.47
  --webservice WEBSRV   OTRS Web Service
  --user USER           OTRS user
  --pass PASSWORD       OTRS pass
  --customer CUSTOMER   Customer
  --title TITLE          Titulo do chamado
  --desc DESCRICAO      descricao
  --fila FILA           Fila de atendimento
  --servico SERVICO     Servico de abertura
  --sla SLA             Acordo de nivel de servico a ser utilizado
  --triggerid TGRID     Trigger ID do zabbix
  --host HOST           Nome do host no zabbix
  --status STATUS       Indisponibilidade
```

6. Script Execution: The script will run from the zabbix server.

7. Connection test and script: Now that we have the script ready and the information collected correctly, we will do a test, execute the command below on the zabbix server.

```
python criar_chamado_zabbix.py --otrs {IP_OTRS} --webservice ZabbixOTRS --user {USER} --
pass {PASS} --customer {COSTUMER} --title "Title of Ticket" --desc "Description of ticket" --fila
"{QUEUE}" --servico "{SERVICE}" --sla "{SLA}"
```

8. Creating action on zabbix

- 1) Access zabbix under Configuration> Actions and click on "Create Action"
- 2) Define a name for the action
- 3) In "Type of calculation" select "Ando / Or" (In later versions you will need to add two conditions so that the option is enabled).
- 4) Under "Conditions" select:
 - a) Maintenance status not in maintenance

- b) * Trigger=Centos7-Test:HTTP service is down on Centos7-Test
 * Depending on the trigger you choose it is necessary to change the description of the call, for a better understanding.

- 5) Check the "Enabled".
- 6) In the "Operations".
- 7) Under "Operations" click on "New".
- 8) In "Operation type" select "Remote command".
- 9) Under "Target list" select "Current host".
- 10) Under "Type" select "Custom script".
- 11) In "Execute on" check the "Zabbix server".
- 12) Under "Commands" copy and paste the following content:

```
/bin/python /usr/lib/zabbix/alertscripts/criar_chamado_zabbix.py --otrs 10.20.19.47 --
webservice ZabbixOTRS --user otrs.isaac --pass pass.isaac --customer cliente --title
"PROBLEMA: {TRIGGER.SEVERITY} - {TRIGGER.NAME} - {TRIGGER.STATUS}"
--desc "O host abaixo esta com problemas.
Name of host: {HOST.HOST}
Name of item: {ITEM.NAME}
Name of trigger: {TRIGGER.NAME}
Status of trigger: {TRIGGER.STATUS}
Severity of trigger: {TRIGGER.SEVERITY}
Date of event: {EVENT.DATE} {EVENT.TIME}
Actual Date:{DATE}{TIME}
Trigger ID: {TRIGGER.ID}
All the time of the problem: {EVENT.AGE}" --fila "Infra" --servico
"INCIDENTE::INFRAESTRUTURA::SERVIDORES::LINUX" --sla "ALTA::24" --triggerid
{TRIGGER.ID} --host {HOST.NAME} --status INDISPONIBILIDADE
```

- 13) You must be careful to change the fields:
 - otrs: which contains the IP of the server.
 - webservice: which contains the name of the webservice you created.
 - user: user name for connection in otrs password: otrs user password.
 - costumer: on whose behalf the call will be opened (applicant).
 - queue: name of the queue for the named caller (If you have more than one queue, I suggest creating other scripts with different queues, in which case you should change item 4 in step 7).
 - service: service registered in otrs sla: sla level registered in otrs.

- 14) Click "Add"
- 15) Click "Add" again.

9. Closing the Ticket:

- 1) Click Recovery operations
- 2) Under "Operations" click on "New"
- 3) In "Operation type" select "Remote command"
- 4) Under "Target list" select "Current host"
- 5) Under "Type" select "Custom script"
- 6) In "Execute on" check the "Zabbix server"
- 7) Under "Commands" copy and paste the following content:


```
perl/usr/lib/zabbix/alertscripts/fechar_chamado_zabbix.pl -subject 'Incident
normalized' -body 'The incident was normalized' -triggerid {TRIGGER.ID}
```

10. Testing the script

- 1) Add a test host under the created conditions of the action, refer to item 4 of step 7, and cause the incident.
- 2) Check the OTRS if the call was opened with the correct specifications.
- 3) After normalizing the event verify that the call has been terminated.

11. Test suggestion for debugging errors: If you have any problems with opening and closing calls, perform the following steps:

- 1) In the option "Recovery operations" do not register any action.
- 2) Perform the steps correctly in "Operations".
- 3) Have a call opened by zabbix itself, informing the trigger ID of the event.
- 4) After this run the script below on zabbix server.

```
perl/usr/lib/zabbix/alertscripts/fechar_chamado_zabbix.pl -subject 'Incident  
normalized' -body 'The incident was normalized' -triggerid 15112
```

5) Note that the value of the trigger ID has changed, you will get this value in the otrs
ticket, in the parameter "Trigger ID: {TRIGGER.ID}".

Thank you!

Telegram: @lakim