SECURING ZABBIX 6.0 LTS

KĀRLIS SALIŅŠ
Technical Support Engineer, Zabbix, Latvia
WHY DO WE NEED “SECURITY”? 

• INDUSTRY STANDARDS 
• BUSINESS NEEDS 
• TO PREVENT DATA BREACH
ACHIEVING SECURE ENVIRONMENT

- Using encryption to protect data
- Role-based access
- Audit logging for visibility
- ISO standards
SECURITY IN ZABBIX

- Encryption
- Certificates
- PSK
- API tokens
- HashiCorp Vault
HashiCorp Vault

Zabbix Agents

Zabbix Proxies

Commandline utilities

Role Based Access

Zabbix Web interface

HTTPS

bcrypt user passwords

password policy

Zabbix Database
BUILT-IN USER TYPES

- Zabbix Super Admin
  - Unlimited Access

- Zabbix Admin
  - Can create hosts and templates
  - Permission based access

- Zabbix User
  - Permission based access
  - Can view monitoring
OBTAINING API TOKENS

{
  "jsonrpc": "2.0",
  "method": "user.login",
  "params": {
    "user": "Karlis",
    "password": "K4rL!s"
  },
  "id": 1,
  "auth": null
}
NEW WAY OF OBTAINING API TOKENS
NEW WAY OF OBTAINING API TOKENS

Name: Karlis token
User: Karlis
Auth token: cf759668fd86b9376cea9025678201f5f6524fd57bce1e311a51c7d7c557349e
Expires at: 2021-12-31 00:00:00
Description: Token generated from Zabbix GUI
Enabled: ✔

Copy to clipboard
NEW WAY OF OBTAINING API TOKENS

API tokens

Name | Created by users | Expires in less than
-----|-----------------|---------------------

Users | Select

Status | Any | Enabled | Disabled

<table>
<thead>
<tr>
<th>Name</th>
<th>User</th>
<th>Expires at</th>
<th>Created at</th>
<th>Created by user</th>
<th>Last accessed at</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Karlis token</td>
<td>Karlis</td>
<td>2021-12-31 00:00:00</td>
<td>2021-11-03 07:45:44</td>
<td>Admin (Zabbix Administrator)</td>
<td>Never</td>
<td>Enabled</td>
</tr>
<tr>
<td>Unique token</td>
<td>guest</td>
<td>2021-11-30 00:00:00</td>
<td>2021-11-03 07:41:22</td>
<td>Admin (Zabbix Administrator)</td>
<td>Never</td>
<td>Enabled</td>
</tr>
</tbody>
</table>
USING API TOKENS

0424bd59b807674191e7d77572075f33

{
  "jsonrpc": "2.0",
  "method": "choose.method",
  "params": {
    "param": "one",
    "param": "two"
  },
  "id": 1,
  "auth": 0424bd59b807674191e7d77572075f33
}
SECRET MACROS

- Value of macro is not displayed
- Value is not cloned / exported with Host / Template
- Secret macros are stored in database
- Database connection and access MUST be secured
VAULT USAGE

- HashiCorp vault can be used as storage for secrets
- Vault is a tool for securely accessing secrets, such as passwords
- Vault provides a unified interface to any secret, while providing tight access control and recording a detailed audit log
- Initially vault is sealed and must be unsealed using unseal keys
- Secrets are stored in Zabbix configuration cache
- The values of secrets are retrieved on every Zabbix configuration update
### Option: VaultToken

# Vault authentication token that should have been generated
# exclusively for Zabbix server with read only permission

**VaultToken=verysecretrandomlygeneratedvaultstring**

### Option: VaultURL

# Vault server HTTP[S] URL. System-wide CA certificates directory
# will be used if SSLCAlocation is not specified.

**VaultURL=https://my.organization.vault:8200**

### Option: VaultDBPath

# Vault path from where credentials for database will be retrieved
# by keys 'password' and 'username'.

**VaultDBPath=my/secret/location**
WHAT’S NEW IN ZABBIX 6.0 LTS

- Audit log upgrades
- Password complexity requirements
- TLS/SSL website certificate monitoring
- User permissions for the service tree
UPGRADED AUDIT LOG

- Better API operations logging
- Better support regarding high amount of items/devices/etc.
- Overall quality of life improvements
- Added various new metrics to be logged:
  - Script execution
  - Global macro change
  - LLD changes
  - etc.
<table>
<thead>
<tr>
<th>Time</th>
<th>User</th>
<th>IP</th>
<th>Resource</th>
<th>Action</th>
<th>ID</th>
<th>Description</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>2021-11-03</td>
<td>karls</td>
<td>192.168.100.200</td>
<td>Macro</td>
<td>Update</td>
<td>6</td>
<td>($\text{TEST_MACRO}$)</td>
<td>globalmacro.value: testvalue =&gt; testvalue23</td>
</tr>
<tr>
<td>2021-11-03</td>
<td>karls</td>
<td>192.168.100.200</td>
<td>User</td>
<td>Add</td>
<td>68</td>
<td>karls1</td>
<td></td>
</tr>
<tr>
<td>2021-11-03</td>
<td>karls</td>
<td>192.168.100.200</td>
<td>Macro</td>
<td>Add</td>
<td>6</td>
<td>($\text{TEST_MACRO}$)</td>
<td></td>
</tr>
<tr>
<td>2021-11-03</td>
<td>karls</td>
<td>192.168.100.200</td>
<td>Script</td>
<td>Execute</td>
<td>2</td>
<td></td>
<td>script.execute_on: =&gt; 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>script.hostid: =&gt; 10084</td>
<td>script.command: =&gt; /usr/sbin/traceroute 192.168.7.203 2&gt;&amp;1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>script.error: =&gt; sh: /usr/sbin/traceroute: No such file or directory</td>
<td></td>
</tr>
<tr>
<td>2021-11-03</td>
<td>karls</td>
<td>192.168.100.200</td>
<td>Script</td>
<td>Execute</td>
<td>1</td>
<td></td>
<td>script.execute_on: =&gt; 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>script.hostid: =&gt; 10084</td>
<td>script.command: =&gt; /usr/bin/ping -c 3 192.168.7.203 2&gt;&amp;1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>script.error: =&gt; PING 192.168.7.203 (192.168.7.203) 56 (84) bytes of data. 64 bytes from ms 64 bytes from 192.168.7.203: icmp_seq=2 ttl=64 time=0.073 ms 64 bytes from 192.168.7.203 ping statistics --- 3 packets transmitted, 3 received, 0% packet loss, time 5 0.073/0.081/0.088/0.012 ms</td>
<td></td>
</tr>
<tr>
<td>2021-11-03</td>
<td>karls</td>
<td>192.168.100.200</td>
<td>Script</td>
<td>Execute</td>
<td>3</td>
<td></td>
<td>script.execute_on: =&gt; 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>script.hostid: =&gt; 10084</td>
<td>script.command: =&gt; nmap --O 192.168.7.203 2&gt;&amp;1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>script.error: =&gt; TCP/IP fingerprinting (for OS scan) requires root privileges. QUITTING!</td>
<td></td>
</tr>
</tbody>
</table>
COMPLEX USER PASSWORDS

- Password no longer cannot be *password*
- Implemented password complexity and policy, encryption in bcrypt
- Implemented password requirements:
  - 8 characters long
  - Must not contain name/last name/username
  - Not easy to guess (abcd1234, asdf1234)
  - ~1 million most common passwords cannot be used
- Fresh installation Admin password does not change
Password = qwerty1234

Name = John

Password = John1234
MONITORING OF CERTIFICATE ATTRIBUTES

- Zabbix Agent 2 built-in plugin starting from Zabbix Agent2 5.0.15
- Displays information about certificate in a website
- Template available out of the box
- `web.certificate.get[<website_DNS_name>,port,ip]`
web.certificate.get[www.zabbix.com]  

```json
[s]{{"x509":{
  "version":3,
  "serial_number":"0f5bd7fa1129ddf854e2745a3e8dc788",
  "signature_algorithm":"ECDSA-SHA256",
  "issuer":"CN=Cloudflare Inc ECC CA-3,O=Cloudflare\\ Inc.,C=US",
  "not_before":{"value":"Jun 08 00:00:00 2021 GMT","timestamp":1623110400},
  "not_after":{"value":"Jun 07 23:59:59 2022 GMT",
  "timestamp":1654646399},
  "subject":"CN=zabbix.com,O=Cloudflare\\ Inc.,L=San Francisco,ST=California,C=US",
  "public_key_algorithm":"ECDSA",
  "alternative_names":["*.zabbix.com","zabbix.com"],
  "result":{"value":"valid",
  "message":"certificate verified successfully"},
  "sha1_fingerprint":"e759419726b0599484d75977b5e0c8f6a4fa6728",
  "sha256_fingerprint":"0ffeef9b263219decf7db55c32ba65cd59bfe72b83841aa6fb720c830281fe71"}}
```

"GOOD" CERTIFICATE
zabbix_agent2 -t web.certificate.get[self-signed.badssl.com]

```
web.certificate.get[self-signed.badssl.com]   
```

```
"version":3,
"serial_number":"c9c0f0107cc53eb0",
"signature_algorithm":"SHA256-RSA",
"issuer":"CN=*.badssl.com,O=BadSSL,L=San Francisco,ST=California,C=US",
"not_before":{"value":"Oct 11 20:03:54 2021 GMT","timestamp":1633982634},
"not_after":{"value":"Oct 11 20:03:54 2023 GMT","timestamp":1697054634},
"subject":"CN=*.badssl.com,O=BadSSL,L=San Francisco,ST=California,C=US",
"public_key_algorithm":"RSA",
"alternative_names":["*.badssl.com","badssl.com"]

"result":{"value":"valid-but-self-signed",
"message":"certificate verified successfully, but determined to be self signed"},
"sha1_fingerprint":"303eabd4efe3b129e56bb54132492777d57b7116",
"sha256_fingerprint":"fc31cc459fbfa26d95f4ba432b27275d2444a88a1c13b6d3bab99a71ac18b96c"}
```
COMMON TRIGGER EXAMPLES

- Invalid certificate

  \[
  \{\text{HOST:cert.validation.str("invalid")}\} = 1
  \]

- Certificate expires in 7 days

  \[
  (\{\text{HOST:cert.not_after.last()}\} - \{\text{HOST:cert.not_after.now()}\}) / 86400 < 7
  \]

- Certificate fingerprint has changed

  \[
  \{\text{TEMPLATE_NAME:cert.sha1_fingerprint.diff()}\}=1
  \]
SERVICE TREE UPGRADES

- Reworked implementation
- Improved scalability
- Better API compatibility
- User permission improvements
MODIFYING ACCESS LEVELS

Access to services

Read-write access to services
- None
- All
- Service list
  - service1

Read-write access to services with tag
- Tag: 
  - Value:

Read-only access to services
- None
- All
- Service list
  - karlis

Read-only access to services with tag
- Tag:
  - Value:
## Permissions View

### Services

**Read-only root service**

- **Status:** OK
- **SLA:** 99.9000
- **Tags:** City, Valmiera

<table>
<thead>
<tr>
<th>Name</th>
<th>Status</th>
<th>Root cause</th>
<th>SLA</th>
<th>Tags</th>
</tr>
</thead>
<tbody>
<tr>
<td>Child service 1</td>
<td>OK</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Child service 2</td>
<td>OK</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Child service 3 with explicit read-write access</td>
<td>OK</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
BEST PRACTICES
BEST PRACTICES

- HashiCorp Vault
- Zabbix Agents
- Zabbix Proxies
- Zabbix Database
- Commandline utilities

Role Based Access

Zabbix Web interface

HTTPS

bcrypt user passwords

password policy
IS YOUR ZABBIX SECURE?

- Are you using encryption on ALL Zabbix components?
- Are you using HTTPS to access frontend?
- Is Zabbix Agent2 database connections secured?
- Is the connection to database secure?
- Are there agent key restrictions in place?
- Are user permissions configured correctly?
- Have important or possibly vulnerable macros been made secret?
- Are most valuable secrets stored in vault?
- Is the newest minor release installed? (4.0.35, 5.0.17, 5.4.7)
LEARN IN-DEPTH SECURITY FEATURES OF ZABBIX

- One day course
- Securing Zabbix components
- Using key vault
- Securing database connections
- RBAC system
- Zabbix Agent key restrictions
Thank you!