

# CMDBs and Check\_MK

## whoami?

- working with Linux since 20 years
- working at Heinlein Support since 3.5 years
- working with Check\_MK since 3 years
- trainer & consultant for Check\_MK (and other topics)

# **part 1**

## **automate check\_MK config?**

## automation makes life easier

- configuration is reproducible
- monitor deployed hosts, not manually configured hosts
  - always up to date
- central repository for configuration
  - easier documentation
- fewer manual steps
  - fewer errors

## sources of automation

- CMDB  
configuration management database
  - information pool
  
- deployment tools
  - SaltStack
  - Puppet
  - Chef
  - Ansible
  - ...
  
- dynamically generated hostnames

## what can be automated?

- host properties
  - Hostnames
  - Location in folder tree
  - Tags
  - IP addresses
- discovery
- config activation
  
- NOT
  - Rules
  - Tag groups and aux tags

## **part 2**

# **how to automate check\_MK config?**

## method 1

### wato\_import.py

- imports CSV
  - folder;hostname;hostalias;ip address
- creates a **new** hosts.mk in folder
  - overwrites existing hosts
- no handling of host tags
- no deletion or modification of hosts
- suitable for one time imports at the start of a fresh site
  - folders with assigned tags
  
- .../share/doc/check\_mk/treasures




## method 2 creating hosts.mk

- write your own code to create hosts.mk and .wato in folders
- extend `all_hosts`
- update `ipaddresses`
- extend `extra_host_conf['alias']`
- update `host_attributes`
- create `.wato` file
  - ```
{'attributes': {},  
  'num_hosts': 1,  
  'title': u'Folder Name'}
```

## method 2

# protection against WATO changes

- `_lock=True` in `hosts.mk`
- `{ 'lock' : True }` and `{ 'lock_subfolders' : True }` in `.wato`
- instead of `True` arbitrary text which is displayed in folder view
- no new hosts, delete, copy or move, no editing of host properties
- no "Save & go to services"
  - Discovery still possible through 
- no inheritance of host tags from folders
  - all tags have to be defined at the host
- set `'num_hosts'` in `.wato` manually when folder is locked

# Service Discovery & Activation for file based methods

- `cmk -I`
- `cmk -O`

## method 3

### WATO web API

- talks JSON
- authenticates only automation users
- add single host
- edit single host
- delete single host
- get single host
- get all hosts
- discover services for one host
- activate changes

# WATO web API request basics

- endpoint:
  - `http://{server}/{omdsite}/check_mk/webapi.py`
- URL parameters
  - `_username`
  - `_secret`
  - `action` (add\_host, edit\_host, etc.)
  - `mode` (optional)
- payload data
  - request (JSON formatted)
  - `{ "hostname": "serv001" }`

## WATO web API add host

- `action=add_host`
  - `request={`
    - `"attributes": {`
      - `"tag_criticality": "prod",`
      - `"tag_agent": "cmk-agent",`
      - `"alias": "Server 001",`
      - `"ipaddress": "127.0.0.1",`
    - `},`
    - `"folder": "os/linux",`
    - `"hostname": "serv001"`
  - `}`
- Tags: `"tag_<taggroupid>": "<tagid>"`

## WATO web API

### edit host

- `action=edit_host`
- `request={`
  - `"attributes": {`
    - `"site": "testsite2",`
    - `"tag_networking": "dmz"`
  - `},`
  - `"unset_attributes": ["tag_criticality"],`
  - `"hostname": "serv001"`
  - `}`
- hostname is unique even on distributed sites
- unset attributes
- inherits attributes from folder properties

## WATO web API

### get, discover, delete host

- `action=delete_host`
- `action=get_host`
  - `effective_attributes=1` for inherited attributes
- `action=discover_services`
  - mode in new, remove, fixall, refresh
- `request={ "hostname": "serv001" }`



## WATO web API get all hosts, activate changes

- `action=get_all_hosts`
  - `effective_attributes=1` for inherited attributes
- `action=activate_changes`
  - `allow_foreign_changes=1` to allow non-API changes
  - serializes multiple sites → timeout possible
  - mode in `dirty, all, specific`
  - `request={ "sites": ["site_nr1", "site_nr2"] }`

## status checks, commands: multisite web API

- use views in multisite
- get HTML, JSON or Python data structures
- execute commands by "emulating" browser request
- setting downtime on service Dummy host localhost:
- ```
view.py?_do_confirm=yes
    &_transid=-1
    &_do_actions=yes
    &service=Dummy
    &host=localhost
    &site=
    &view_name=service
    &_down_2h=2+hours
    &_down_comment=TEST
```

# CLI for WATO Web-API

## omdclient

- omd-host-crud
  - create, read, update & delete
- omd-activate
- omd-puppet-enc
  - Uses Puppet External Node Classifier to create/remove hosts
- omd-nagios-ack
- omd-nagios-downtime
  
- early stage of development
- fixed environment (tag groups "role" & "instance")

## CMDB product i-doit

- tight integration with check\_MK
- presented last year on this conference
- service trees used as configurations for check\_mk
- corresponding result data fed back to i-doit.
- live analysis of the impact on the service tree in the case of malfunctions of individual components.

## Conclusion

- file based methods fast
- file based methods need knowledge about internal data structures
- file based methods need shell access to monitoring site
  
- Web-API may be slow
- Web-API talks JSON
- Web-API needs https access to monitoring site

# part 3

## CMDB sync

# CMDB: LANsweeper XML export

```
→ <DocumentElement>
  <Server>
    <AssetTypename>Linux</AssetTypename>
    <AssetName>www01.example.com</AssetName>
    <Description>Webserver 01</Description>
    <IPAddress>192.0.2.1</IPAddress>
    <Landscape>Production</Landscape>
    <Monitored>yes</Monitored>
    <Business_Classification>Marketing</Business_Classification>
    <RelationType>Supported By</RelationType>
    <Name>Robert Sander</Name>
    <RelationComment>Primary</RelationComment>
  </Server>
</DocumentElement>
```

→ multiple <Server> elements per host due to relationship

## mapping

XML element	Check_MK attribute
AssetTypename	folder
AssetName	hostname
Description	alias
IPAddress	ipaddress & monitoring site / location
Landscape	tag_criticality
Monitored	tag_criticality (no → offline)
Business_Classification	tag_customer
RelationType	tag_owner
Name	tag_contact-primary
RelationComment	tag_contact-backup



## Python API calls

- using Python Requests: HTTP for Humans module
- ```
def api_request(params, data=None, errmsg='Error'):  
    if data:  
        resp = requests.post(apiurl,  
                              params=params,  
                              data='request=%s' % json.dumps(data))  
    else:  
        resp = requests.get(apiurl,  
                             params=params)  
  
    resp = resp.json()  
    if resp['result_code'] == 1:  
        raise RuntimeError('%s: %s' % ( errmsg,  
                                       resp['result'] ))  
  
    return resp['result']
```

## params & data

```
→ params = { '_username': ...,  
             '_secret': ...,  
             'action': 'add_host' }
```

```
→ data = { 'hostname': 'www01.example.com',  
           'folder': 'server/linux',  
           'attributes': {  
             'alias': 'Webserver 01',  
             'ipaddress': '192.0.2.1',  
             'site': 'monsite01',  
             'tag_criticality': 'prod',  
             'tag_customer': 'marketing',  
             'tag_contact-primary': 'rsander',  
             'tag_owner': 'hketchup',  
           }  
}
```

# part 4

## demo

## links

- [https://mathias-kettner.de/checkmk\\_wato\\_cmdb.html](https://mathias-kettner.de/checkmk_wato_cmdb.html)
- [https://mathias-kettner.de/checkmk\\_wato\\_webapi.html](https://mathias-kettner.de/checkmk_wato_webapi.html)
- [https://mathias-kettner.de/checkmk\\_multisite\\_automation.html](https://mathias-kettner.de/checkmk_multisite_automation.html)
- <https://github.com/tskirvin/omdclient>
- <http://www.i-doit.com/product/interfaces/>
- [http://mathias-kettner.de/check\\_mk\\_konferenz\\_videos\\_2014\\_9.html](http://mathias-kettner.de/check_mk_konferenz_videos_2014_9.html)
- <http://www.python-requests.org>

- Natürlich und gerne stehe ich Ihnen jederzeit mit Rat und Tat zur Verfügung und freue mich auf neue Kontakte.
  - Robert Sander
  - Mail: [r.sander@heinlein-support.de](mailto:r.sander@heinlein-support.de)
  - Telefon: 030/40 50 51 - 43
  
- Wenn's brennt:
  - Heinlein Support 24/7 Notfall-Hotline: 030/40 505 - 110

**Wir suchen:**

Admins, Consultants, Trainer!

**Wir bieten:**

Spannende Projekte, Kundenlob, eigenständige Arbeit, keine Überstunden, Teamarbeit

...und natürlich: Linux, Linux, Linux...

**<http://www.heinlein-support.de/jobs>**

# Heinlein Support hilft bei allen Fragen rund um Linux-Server

## HEINLEIN AKADEMIE

Von Profis für Profis: Wir vermitteln die oberen 10% Wissen: geballtes Wissen und umfangreiche Praxiserfahrung.

## HEINLEIN HOSTING

Individuelles Business-Hosting mit perfekter Maintenance durch unsere Profis. Sicherheit und Verfügbarkeit stehen an erster Stelle.

## HEINLEIN CONSULTING

Das Backup für Ihre Linux-Administration: LPIC-2-Profis lösen im CompetenceCall Notfälle, auch in SLAs mit 24/7-Verfügbarkeit.

## HEINLEIN ELEMENTS

Hard- und Software-Appliances und speziell für den Serverbetrieb konzipierte Software rund ums Thema eMail.