

Einrichtung der Infrastruktur

- Installation Agent für Betriebssysteme mit apt Packetmanager
- Installation Agent für Betriebssysteme mit rpm Packetmanager
- Installation Agent für andere Betriebssysteme ohne systemd / xinetd (Anbindung per ssh)
tar.gz
- Installation Agent für Windows
- Anlegen von Hostgruppen und deren Verzeichnisse
- Anlegen von Hosts
- Agents am CheckMK Server registrieren
- Checks Schwellenwerte / Limits anpassen
- Benachrichtigungen für bestimmte Checks deaktivieren
- Wartungszeiten - geplante Downtimes festlegen
- Discovery anpassen -> Netzwerkkarten ausschliessen

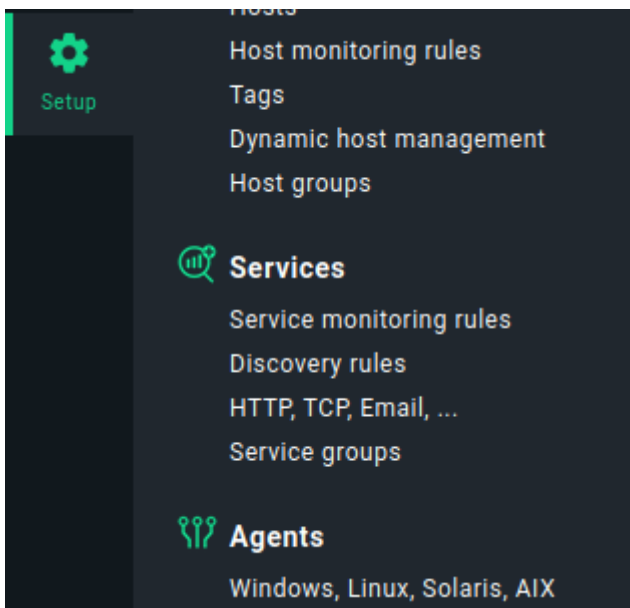
Installation Agent für Betriebssysteme mit apt Packetmanager

Nun agents für die Hosts installieren

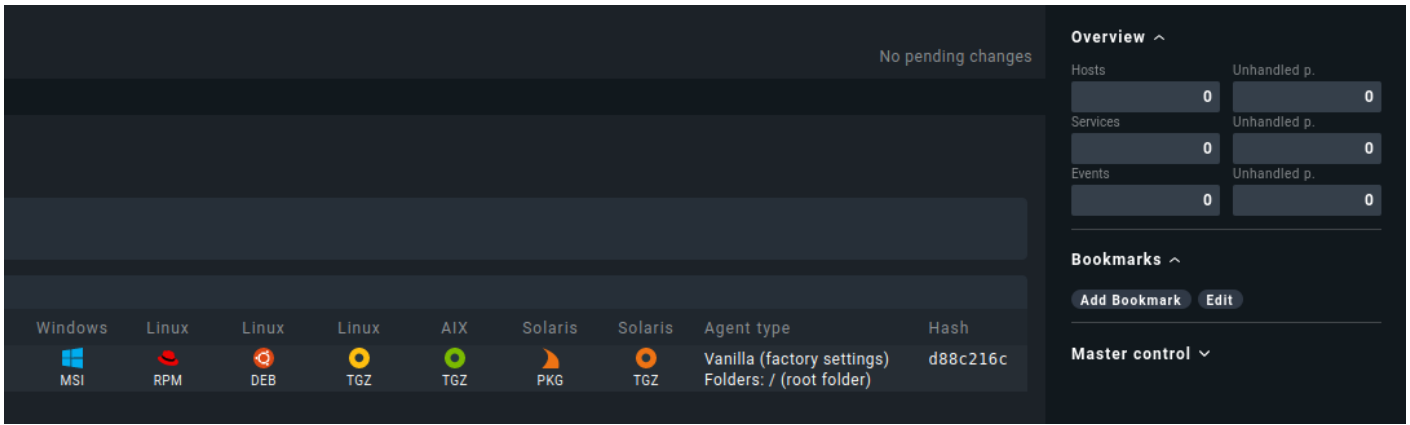
Auf dem zu überwachen des Host wo der Agent installiert werden soll, ein Verzeichnis checkmk_setup im root verzeichnis erstellen

```
mkdir -p /root/checkmk_setup
```

Nun mit dem Webbrowser auf dem Checkmk Server einloggen und dann Dazu auf Setup -> Agents -> Windows, Linux, Solaris , AIX anklicken



Nun auf DEB mit dem Ubuntu Symbol anklicken und downloaden



Nun per scp die datei auf den server ins Home Verzeichnis von root/checkmk_stup übertragen
check_mk übertragen

```
scp Downloads/check-mk-agent_2.1.0p16-d88c216c6ad53a29_all.deb root@192.168.0.30:/root/checkmk_setup
```

Nun auf dem Server einloggen und in das Verzeichnis /root/checkmk_setup gehen und die rpm
Datei installieren

```
cd /root/checkmk_setup  
apt install ./check-mk-agent_2.1.0p16-d88c216c6ad53a29_all.deb
```

Ausgabe:

```
Paketlisten werden gelesen... Fertig  
Abhängigkeitsbaum wird aufgebaut... Fertig  
Statusinformationen werden eingelesen... Fertig  
Hinweis: »check-mk-agent« wird an Stelle von »./check-mk-agent_2.1.0p16-d88c216c6ad53a29_all.deb«  
gewählt.  
Vorgeschlagene Pakete:  
  time  
Die folgenden NEUEN Pakete werden installiert:  
  check-mk-agent  
0 aktualisiert, 1 neu installiert, 0 zu entfernen und 0 nicht aktualisiert.  
Es müssen noch 0 B von 4.517 kB an Archiven heruntergeladen werden.  
Nach dieser Operation werden 4.555 kB Plattenplatz zusätzlich benutzt.  
Holen:1 /root/checkmk_setup/check-mk-agent_2.1.0p16-d88c216c6ad53a29_all.deb check-mk-agent all  
2.1.0p16-1.d88c216c6ad53a29 [4.517 kB]  
Vormals nicht ausgewähltes Paket check-mk-agent wird gewählt.  
(Lese Datenbank ... 35073 Dateien und Verzeichnisse sind derzeit installiert.)  
Vorbereitung zum Entpacken von .../check-mk-agent_2.1.0p16-d88c216c6ad53a29_all.  
deb ...
```

```
Entpacken von check-mk-agent (2.1.0p16-1.d88c216c6ad53a29) ...
check-mk-agent (2.1.0p16-1.d88c216c6ad53a29) wird eingerichtet ...
```

```
Deploying systemd units: check-mk-agent.socket cmk-agent-ctl-daemon.service chec
k-mk-agent-async.service check-mk-agent@.service
```

```
Deployed systemd
```

```
Creating/updating cmk-agent user account ...
```

```
WARNING: The agent controller is operating in an insecure mode! To secure the co
nnection run `cmk-agent-ctl register`.
```

```
Activating systemd unit 'check-mk-agent.socket'...
```

```
Created symlink /etc/systemd/system/sockets.target.wants/check-mk-agent.socket →
/lib/systemd/system/check-mk-agent.socket.
```

```
Activating systemd unit 'cmk-agent-ctl-daemon.service'...
```

```
Created symlink /etc/systemd/system/multi-user.target.wants/cmk-agent-ctl-daemon
.service → /lib/systemd/system/cmk-agent-ctl-daemon.service.
```

```
Activating systemd unit 'check-mk-agent-async.service'...
```

```
Created symlink /etc/systemd/system/multi-user.target.wants/check-mk-agent-async
.service → /lib/systemd/system/check-mk-agent-async.service.
```

```
N: Der Download wird als root und nicht Sandbox-geschützt durchgeführt, da auf die Datei
»/root/checkmk_setup/check-mk-agent_2.1.0p16-d88c216c6ad53a29_all.deb« durch den Benutzer »_apt« nicht
zugriffen werden kann. - pkgAcquire::Run (13: Keine Berechtigung)
```

**!!!NUR durchführen wenn die Agents
später NICHT registriert werden sollen,
was im Kaptitel Grundkonfiguration
beschrieben wird!!!**

Siehe [hier](#) !!!

Um Agents auch auf ipv4 lauschen zu lassen in der config file folgendes ändern.
Aber erstmal die Dienste stoppen und terminieren

```
systemctl stop check-mk-agent.socket  
killall cmk-agent-ctl  
killall check_mk_agent
```

Nun die Datei `/lib/systemd/system/check-mk-agent.socket` bearbeiten

```
nano /lib/systemd/system/check-mk-agent.socket
```

Und folgendes auskommentieren

```
#ListenStream=/run/check-mk-agent.socket
```

Und genau unter der Auskommentierten Zeile diese einfügen

```
ListenStream=0.0.0.0:6556
```

Speichern, `systemctl reloaden` und dienst n eustarten

```
systemctl daemon-reload  
systemctl restart check-mk-agent.socket
```

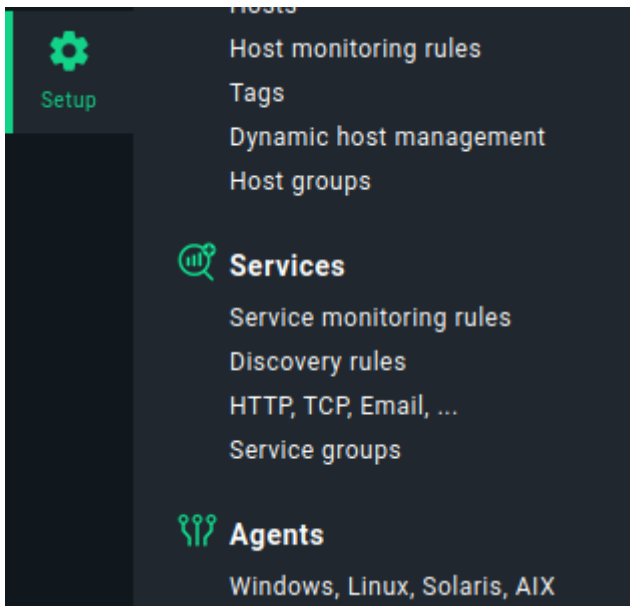
Installation Agent für Betriebssysteme mit rpm Packetmanager

Nun agents für die Hosts installieren

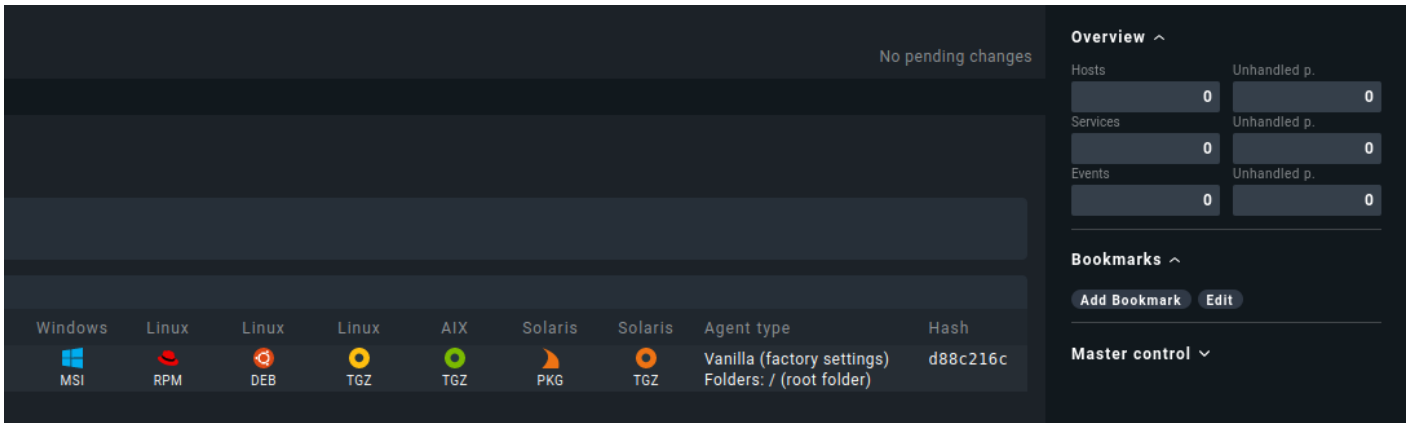
Auf dem zu überwachen des Host wo der Agent installiert werden soll, ein Verzeichnis checkmk_setup im root verzeichnis erstellen

```
mkdir -p /root/checkmk_setup
```

Nun mit dem Webbrowser auf dem Checkmk Server einloggen und dann Dazu auf Setup -> Agents -> Windows, Linux, Solaris , AIX anklicken



Nun auf RPM mit dem Redhat Symbol anklicken und downloaden



Nun per scp die datei auf den server ins Home Verzeichnis von root/checkmk_stup übertragen
check_mk übertragen

```
scp Downloads/check-mk-agent-2.1.0p16-d88c216c6ad53a29.noarch.rpm  
root@192.168.0.30:/root/checkmk_setup
```

Nun auf dem Server einloggen und in das Verzeichnis /root/checkmk_setup gehen und die rpm
Datei installieren

```
cd /root/checkmk_setup  
rpm -i check-mk-agent-2.1.0p16-d88c216c6ad53a29.noarch.rpm
```

Ausgabe:

```
Removing deployed systemd units: check-mk-agent-async.service, check-mk-agent.socket, cmk-agent-ctl-  
daemon.service, check-mk-agent@.service  
  
Deploying systemd units: check-mk-agent-async.service check-mk-agent.socket cmk-agent-ctl-daemon.service  
check-mk-agent@.service  
Deployed systemd  
Creating/updating cmk-agent user account ...  
  
WARNING: The agent controller is operating in an insecure mode! To secure the connection run `cmk-agent-ctl  
register`.  
  
Activating systemd unit 'check-mk-agent-async.service'...  
Created symlink /etc/systemd/system/multi-user.target.wants/check-mk-agent-async.service →  
/lib/systemd/system/check-mk-agent-async.service.  
Activating systemd unit 'check-mk-agent.socket'...  
Created symlink /etc/systemd/system/sockets.target.wants/check-mk-agent.socket → /lib/systemd/system/check-
```

```
mk-agent.socket.  
Activating systemd unit 'cmk-agent-ctl-daemon.service'...  
Created symlink /etc/systemd/system/multi-user.target.wants/cmk-agent-ctl-daemon.service →  
/lib/systemd/system/cmk-agent-ctl-daemon.service.  
2022-11-20 16:34:32,252 - ha-helper[28335] - INFO - Synchronizing to node 192.168.177.2  
2022-11-20 16:34:32,253 - ha-helper[28335] - INFO - Synchronizing... / -> root@192.168.177.2:/  
2022-11-20 16:34:32,510 - ha-helper[28335] - INFO - Synchronization / -> root@192.168.177.2:/ done  
280 run_parts async call
```

!!!NUR durchführen wenn die Agents später NICHT registriert werden sollen, was im Kaptitel Grundkonfiguration beschrieben wird!!!

Um Agents auch auf ipv4 lauschen zu lassen in der config file folgendes ändern.
Aber erstmal die Dienste stoppen und terminieren

```
systemctl stop check-mk-agent.socket  
killall cmk-agent-ctl  
killall check_mk_agent
```

Nun die Datei /lib/systemd/system/check-mk-agent.socket bearbeiten

```
nano /lib/systemd/system/check-mk-agent.socket
```

Und folgendes auskommentieren

```
#ListenStream=/run/check-mk-agent.socket
```

Und genau unter der Auskommentierten Zeile diese einfügen

```
ListenStream=0.0.0.0:6556
```

Speichern, systemctl reloaden und dienst n eustarten

```
systemctl daemon-reload
```

```
systemctl restart check-mk-agent.socket
```

Installation Agent für andere Betriebssysteme ohne systemd / xinetd (Anbindung per ssh) tar.gz

Beschreibung

Es gibt Systeme auf denen gibt es kein Systemd und xinetd.

Der Checkmk Service lauscht normalerweise auf einen TCP Port.

Da es aber keine Services ohne xinetd und systemd gibt, gibt es eine weitere Methode. und zwar das abrufen des Agents über ssh.

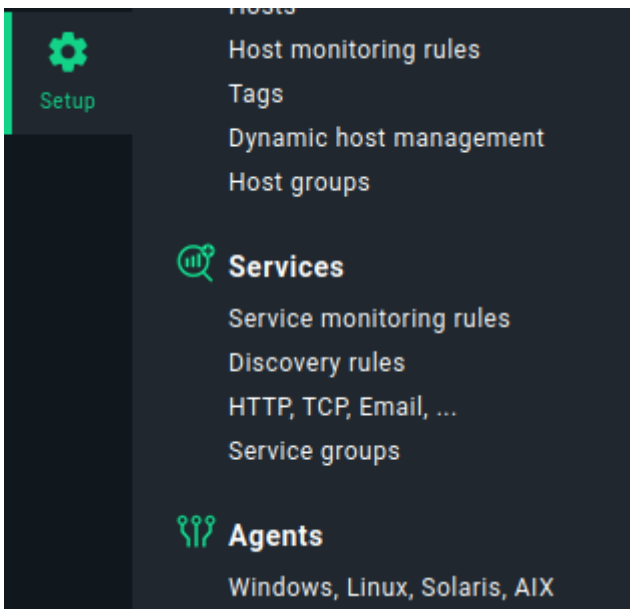
Allerdings funktioniert für diese Methode die Agent Bakery natürlich nicht.

Agents für die Hosts installieren / entpacken

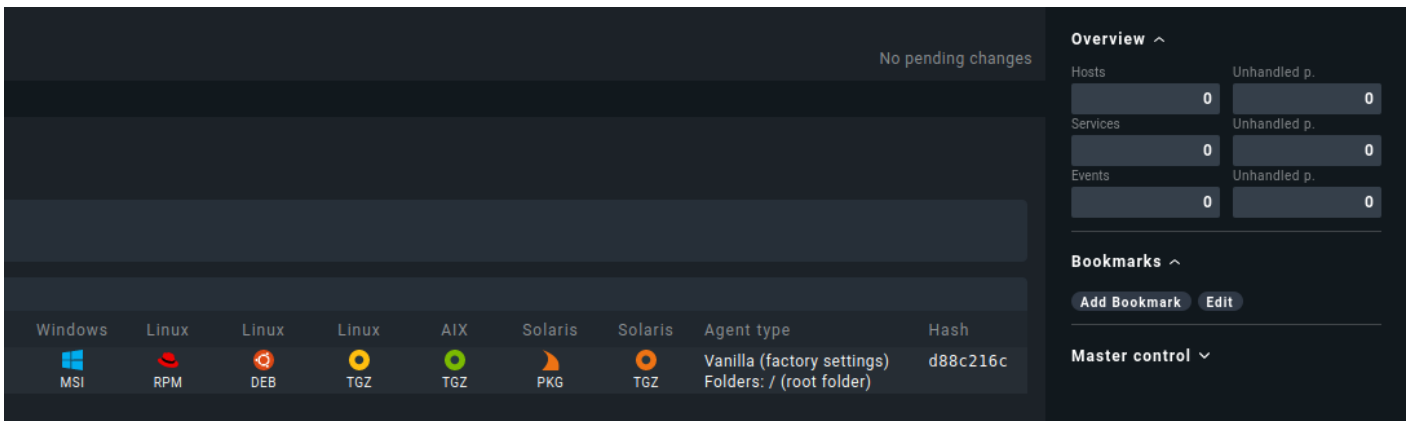
Auf dem zu überwachenden des Host wo der Agent installiert werden soll, ein Verzeichnis checkmk_setup im root verzeichnis erstellen

```
mkdir -p /root/checkmk_setup
```

Nun mit dem Webbrowser auf dem Checkmk Server einloggen und dann Dazu auf Setup -> Agents -> Windows, Linux, Solaris , AIX anklicken



Nun auf tar gz bei Linux mit dem gelben Symbol anklicken und downloaden



Nun per scp die Datei auf den zu überwachenden Server ins Home Verzeichnis von root/checkmk_stup übertragen check_mk übertragen

```
scp Downloads/check-mk-agent_2.1.0p16-d88c216c6ad53a29.tar.gz root@192.168.0.30:/root/checkmk_stup
```

Nun die check-mk-agent_2.1.0p16-d88c216c6ad53a29.tar.gz tar auf dem zu überwachendem Host entpacken

```
tar -xf check-mk-agent_2.1.0p16-d88c216c6ad53a29.tar.gz -C /
```

SSH Schlüssel falls nicht schon vorhanden auf der Checkmkinstanz erstellen

Auf dem Checkmk server inloggen.
Dann in die instanz einloggen

```
omd su <instanzname>
```

Beispiel

```
omd su monitoring
```

Nun den Schlüssel erstellen ohne passphrase

```
ssh-keygen -t ed25519
```

Ausgabe

```
OMD[monitoring]:~$ ssh-keygen -t ed25519
Generating public/private ed25519 key pair.
Enter file in which to save the key (/omd/sites/monitoring/.ssh/id_ed25519):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /omd/sites/monitoring/.ssh/id_ed25519.
Your public key has been saved in /omd/sites/monitoring/.ssh/id_ed25519.pub.
The key fingerprint is:
cc:87:34:d2:ed:87:ed:f7:1b:ec:58:1f:7c:23:00:e2 monitoring@mycmkserver
The key's randomart image is:
+--[ED25519 256]--+
|          |
|   ..    |
|  ..+..   |
|   .=.+.o |
|    ES+.o |
|   .o.o   |
|     ...B.|
|     .=.*)|
|     .o+  |
+-----+
```

Die Schlüssel liegen dann im `.ssh` Verzeichnis des Monitoring Verzeichnisses.
Dateien anzeigen

```
ll .ssh
```

Ausgabe

```
OMD[monitoring]:~$ ll .ssh
total 8
-rw----- 1 monitoring monitoring 1679 Feb 20 14:18 id_ed25519
```

```
-rw-r--r-- 1 monitoring monitoring 398 Feb 20 14:18 id_ed25519.pub
```

Der private Schlüssel heißt `id_ed25519` und ist nur für den Instanzbenutzer lesbar (`-rw-----`) — und das ist auch gut so! Der öffentliche Schlüssel `id_ed25519.pub` sieht etwa so aus:

Ausgabe:

```
OMD[monitoring]:~$ cat .ssh/id_ed25519.pub
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIGb6AaqRPIbEmDnBkeIW3Q6Emb5lr2QEbWEQLmA5pb48
monitoring@mycmkserver
```

Nun auf dem zu überwachendem Host in einer zweiten SSH-Sitzung einloggen.
und die `authorized_keys` Datei bearbeiten

```
nano /root/.ssh/authorized_keys
```

Dort am ende den öffentlichen Schlüssel eintragen.

Davor aber noch ein command, der dient dazu, das nur der Agent aufgerufen werden kann und auch gleichzeitig beim einloggen aufgerufen wird.

Die ganze zeile in der `authorized_keys` Datei sähe dann so aus

```
command="/usr/bin/check_mk_agent" ssh-ed25519
AAAAC3NzaC1lZDI1NTE5AAAAIGb6AaqRPIbEmDnBkeIW3Q6Emb5lr2QEbWEQLmA5pb48
monitoring@mycmkserver
```

Nun ausloggen.

Testen der Verbindung

Nun auf dem Checkmkserver wieder in die instanz einloggen, wenn nicht noch von der vorherigen Sitzung eingeloggt

```
omd su <instanzname>
```

Beispiel

```
omd su monitoring
```

Nun einfach `ssh root@<iphostname>` und den Fingerprint des Schlüssels mit der Eingabe von `yes` bestätigen.

Der Fingerprint wird nur beim ersten mal abgefragt.

```
ssh root@<ipfromtargetserver>
```

Ausgabe

```
OMD[monitoring]:~$ ssh root@myhost23
The authenticity of host 'myhost23 (10.11.12.13)' can't be established.
ECDSA key fingerprint is SHA256:IWgVK+LtsMgjHUbdsA1FK12PdmVQGqaEY4TE8TEps3w.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
<<<check_mk>>>
Version: 2.1.0b6
AgentOS: linux
Hostname: myhost123
AgentDirectory: /etc/check_mk
DataDirectory: /var/lib/check_mk_agent
SpoolDirectory: /var/lib/check_mk_agent/spool
PluginsDirectory: /usr/lib/check_mk_agent/plugins
LocalDirectory: /usr/lib/check_mk_agent/local
<<<df>>>
```

fertig, Verbindung steht.

Einbinden des Hosts in Checkmk

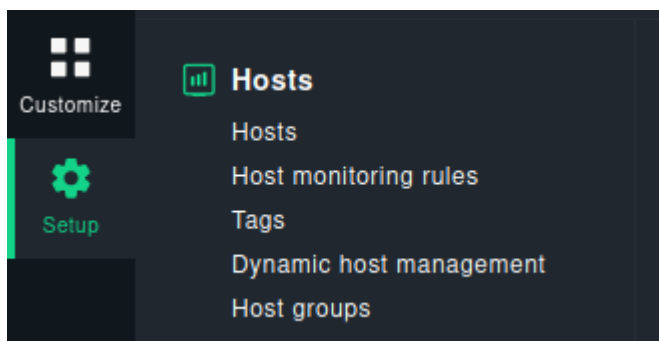
Nun da die SSH Verbindung hergestellt ist, muss jetzt die SSH verbindung im Host in Checkmk eingetragen werden.

Dazu in der Weboberfläche von check mk gehen, den Host anlegen wenn noch nicht geschehen. Sollte der Host schon bestehen. Kann das anlegen natürlich übersprungen werden.

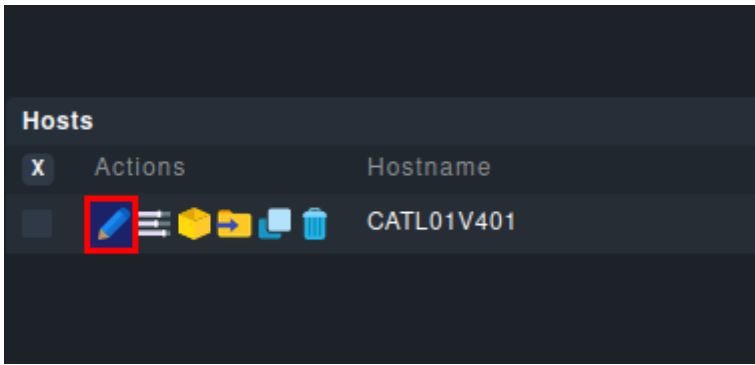
Wichtig ist nur das in den Host Einstellungen folgendes gemacht ist

Hostanpassen

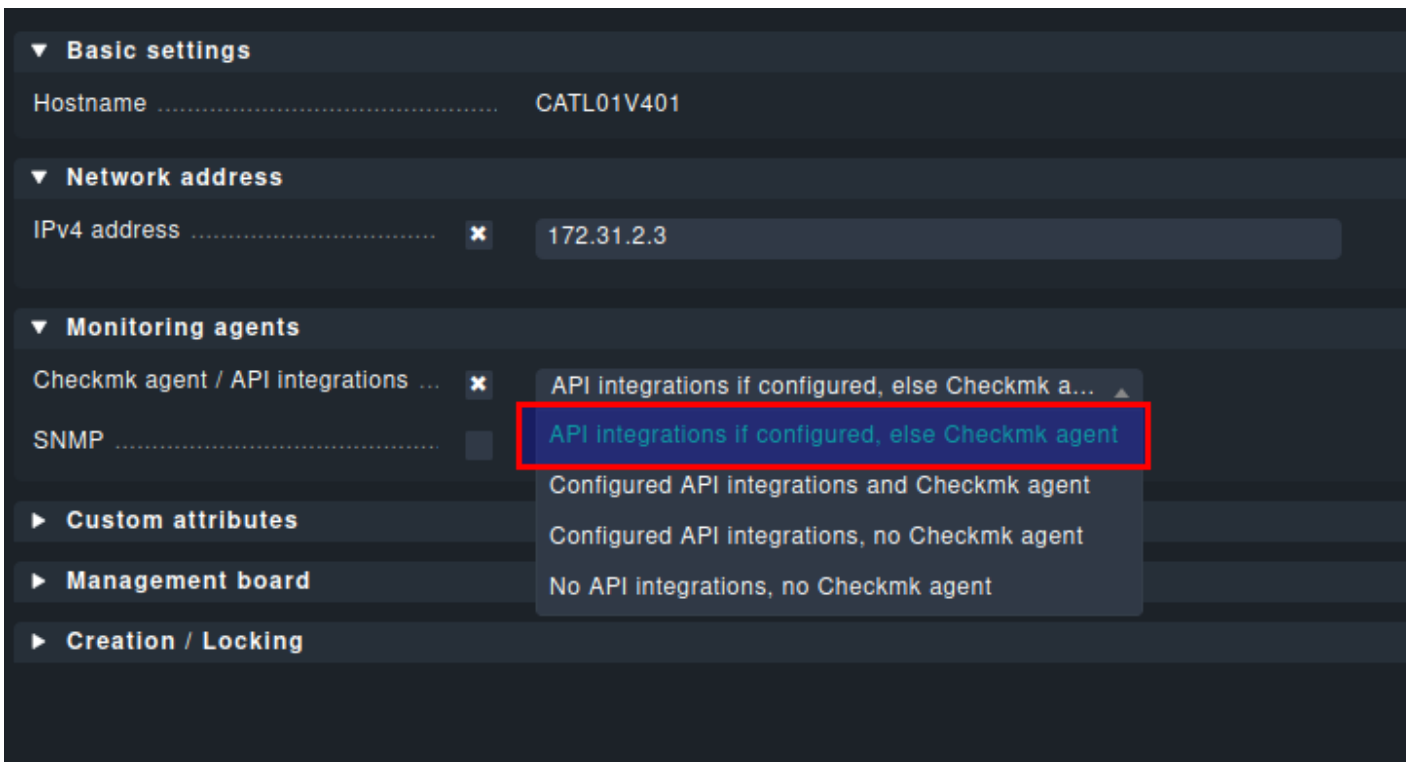
Dazu wieder auf Setup -> Hosts



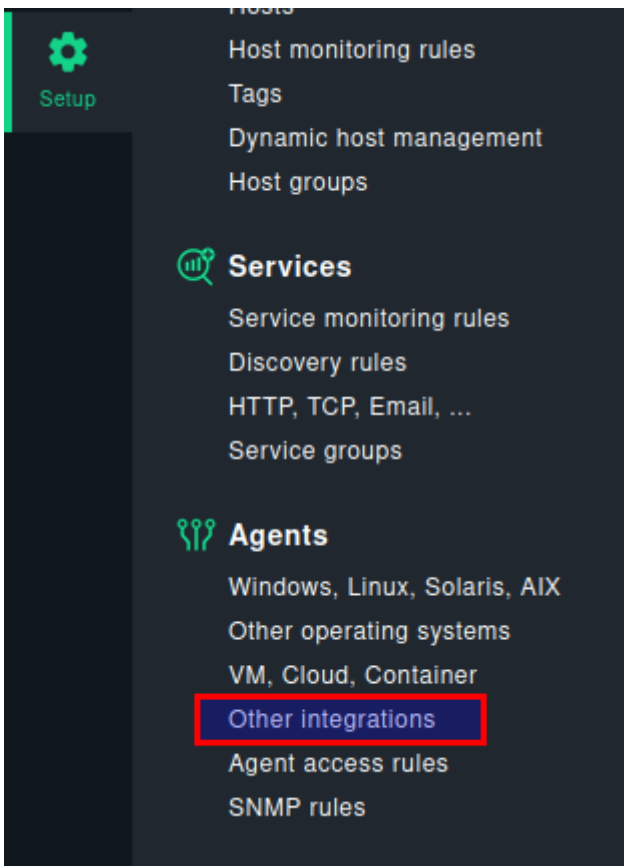
Nun auf edit Host für den Host den man editieren will gehen (Stift)



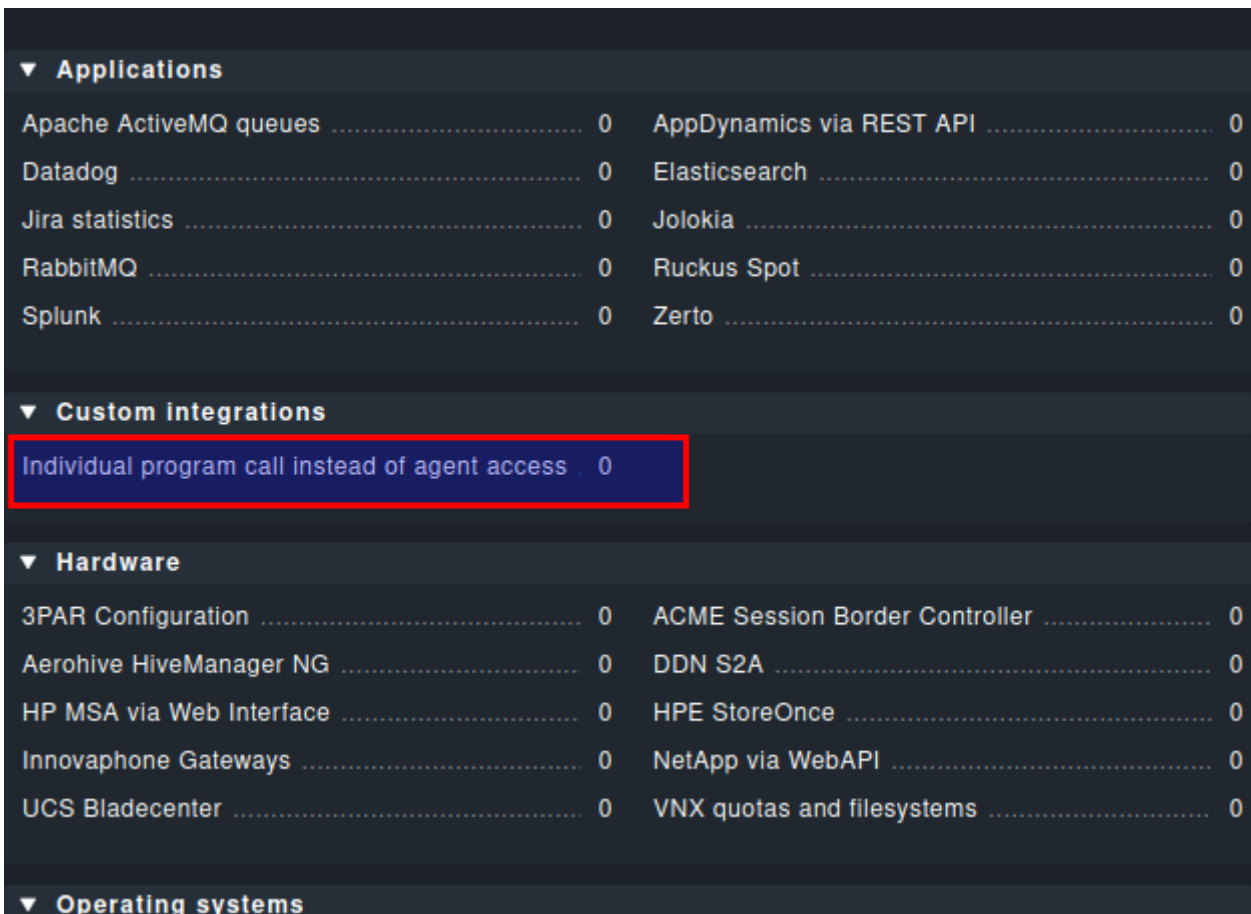
Nun unter Monitoring Agents / API intergration: API integrations if configured, else Checkmk agent auswählen



Wenn die Einstellungen beim Host getätigt sind
Geht es weiter auf : Setup -> Agents -> Other integrations




-> Custom integrations -> Individual program call instead of agent access klicken





Nun auf Add rule klicken

Individual program call instead of agent access

Setup > Agents > Other integrations > Individual program call instead of agent access

Related Rules Display Help 

 Add rule

 Other integrations

Matching: The first matching rule defines the parameter.

There are no rules defined in this set.

Hier nun folgende Sachen ausfüllen:

Description : Agent connection through SSH

Command to Execute : `ssh -T root@$HOSTADDRESS$`

Explicit Hosts: unser(e) Server die, die Verbindung über SSH aufbauen sollen.

Hinweis: Wir nutzen hier als Variable die `$HOSTADDRESS$` dies bedeutet, dazu zum verbinden die IP vom Host genommen wird.

nun auf save klicken.

Rule Related Display Help

Save Abort Individual program call instead of agent access


For agent based checks Check_MK allows you to specify an alternative program that should be called by Check_MK instead of connecting the agent via TCP. That program is for example useful for monitoring via SSH. You can use monitoring macros here. The most important are:

- \$HOSTADDRESS\$: The IP address of the host
- \$HOSTNAME\$: The name of the host
- \$_HOSTTAGS\$: List of host tags
- \$_HOSTADDRESS_4\$: The IPv4 address of the host
- \$_HOSTADDRESS_6\$: The IPv6 address of the host
- \$_HOSTADDRESS_FAMILY\$: The primary address family of the host

All custom attributes defined for the host are available as \$_HOST[VARIABLE]\$. Replace [VARIABLE] with the *upper case* name of your variable. For example, a host attribute can only be used with the permission "Can add or modify executables".
 HINT: The individual program is called from the current working directory. You should therefore specify absolute path names in scripts (by using environment variables like \$PWD in the console).

Rule Properties

Description Agent connection through SSH

Comment 

Documentation URL

Rule activation do not apply this rule

Individual program call instead of agent access

Command line to execute ssh -T root@\$HOSTADDRESS\$

Conditions

Condition type Explicit conditions

Folder Main

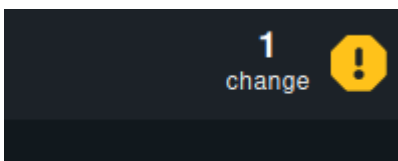
Host tags **Add tag condition**

Host labels **Add label condition**

Explicit hosts (Select hostname)

Negate: make rule apply for all but the above hosts

Nun wieder oben rechts auf change klicken





Nun auf activate on selected sites

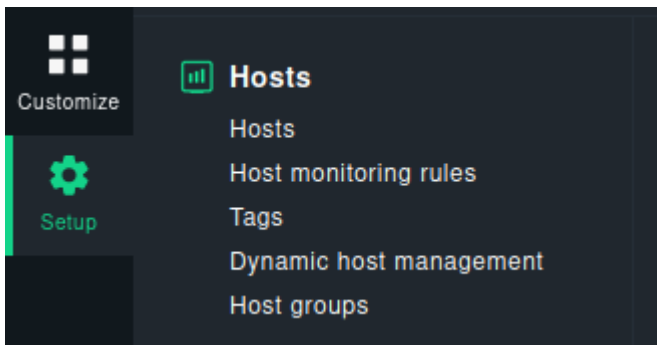
Changes Related Display Help

Activate on selected sites

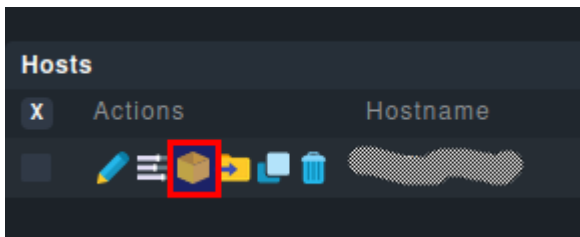
Activation status

Actions	Site	Status
 	Local site catmonitoring	online

Nun wieder auf Setup -> Hosts



Nun auf edit Services für den Host gehen (gelber Würfel)



Nun wenn gewünscht über das X Services deaktivieren, die NICHT überwacht werden sollen. Soll alles übernommen werden einfach auf Accept all

Accept all Rescan Monitor undecided services Remove vanished services Properties of host CATL01V402

Undecided services: 26 | Vanished services: 0 | New host labels: 2 | Vanished host labels: 0 | Changed host labels: 0

No fresh discovery information available. Using latest cached information. Please perform a rescan in case you want to discover the current state.

▼ Discovered host labels (2)

Status	Host labels
New	cmk/device_type:vm cmk/os_family:linux

▼ Undecided services - currently not monitored (26)

State	Service	Status detail
+ x OK	Check_MK Agent	Version: 2.1.0p20, OS: linux, Transport via SSH, Agent plugins: 0, Local checks:
+ x OK	CPU load	15 min load: 0.00, 15 min load per core: 0.00 (2 cores)
+ x OK	CPU utilization	Total CPU: 2.09%
+ x OK	Disk IO SUMMARY	Initializing counters
+ x WARN	Filesystem /	89.16% used (3.10 of 3.48 GB, warn/crit at 80.00%/90.00%)
+ x OK	Filesystem /var	43.32% used (17.37 of 40.09 GB)
+ x OK	Filesystem /var/efw	19.72% used (23.66 of 119.96 MB)
+ x OK	Filesystem /var/log	29.41% used (7.85 of 26.69 GB)
+ x OK	Interface 02	[br0], (up), MAC: 00:50:56:9E:09:57, Speed: unknown
+ x OK	Interface 05	[eth0], (up), MAC: 00:50:56:9E:54:66, Speed: 10 GBit/s
+ x OK	Interface 06	[eth1], (up), MAC: 00:50:56:9E:09:57, Speed: 10 GBit/s
+ x OK	Interface 07	[tap0], (up), MAC: BE:CA:FA:A1:8C:54, Speed: 10 MBit/s

Fertig

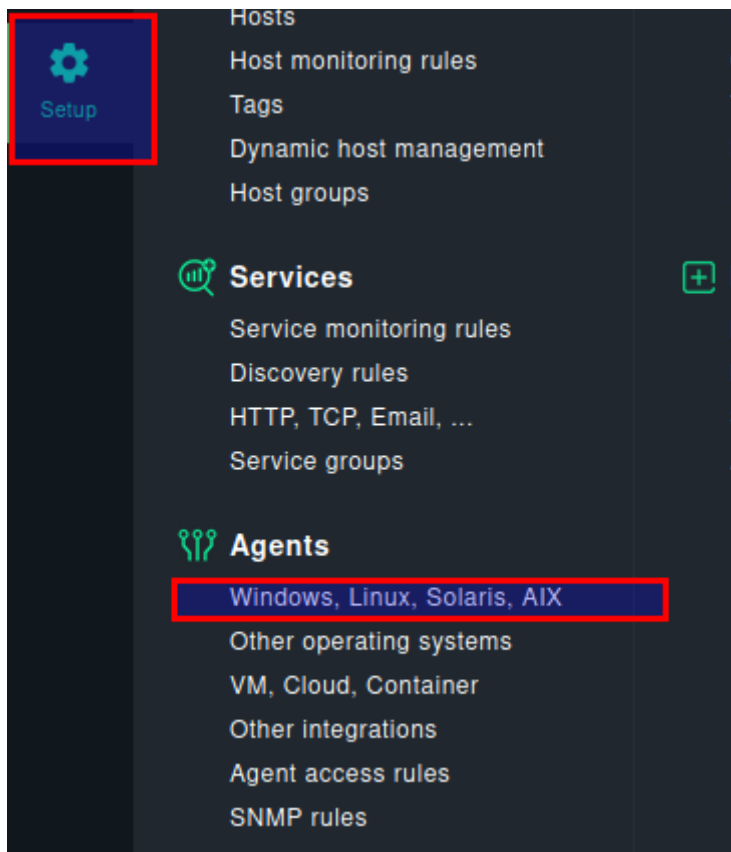
Installation Agent für Windows

Beschreibung:

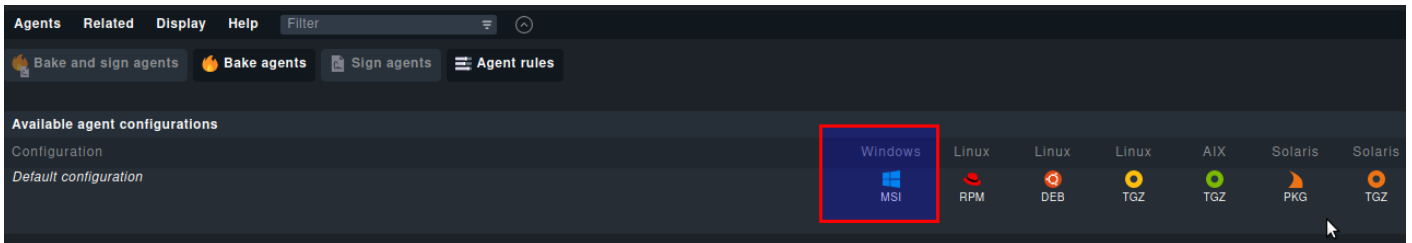
Installation Agent für Windows. Dieser wird dann ähnlich wie bei Linux über die Kommandozeile installiert.

Installation Agent

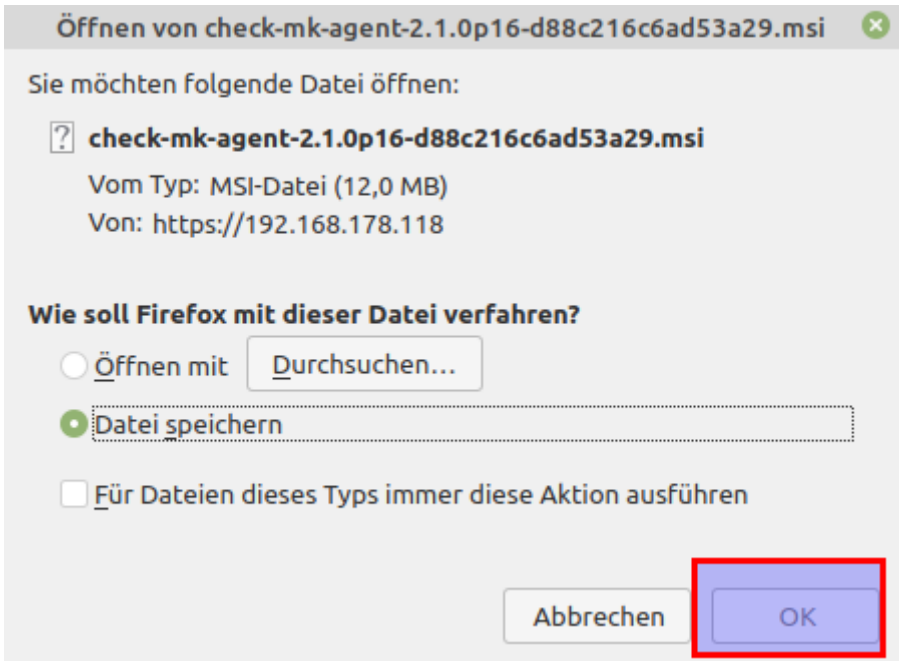
Auf der Check MK Seite anmelden und dann auf Setup -> Agents -> Windows, Linux, Solaris, AIX



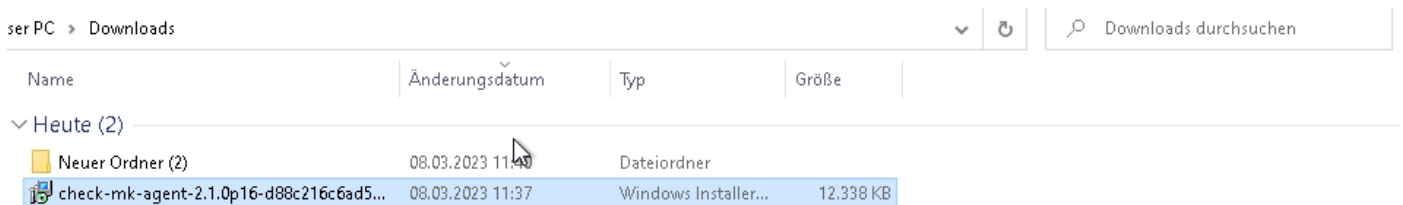
Dort auf das Windows MSI Paket klicken.



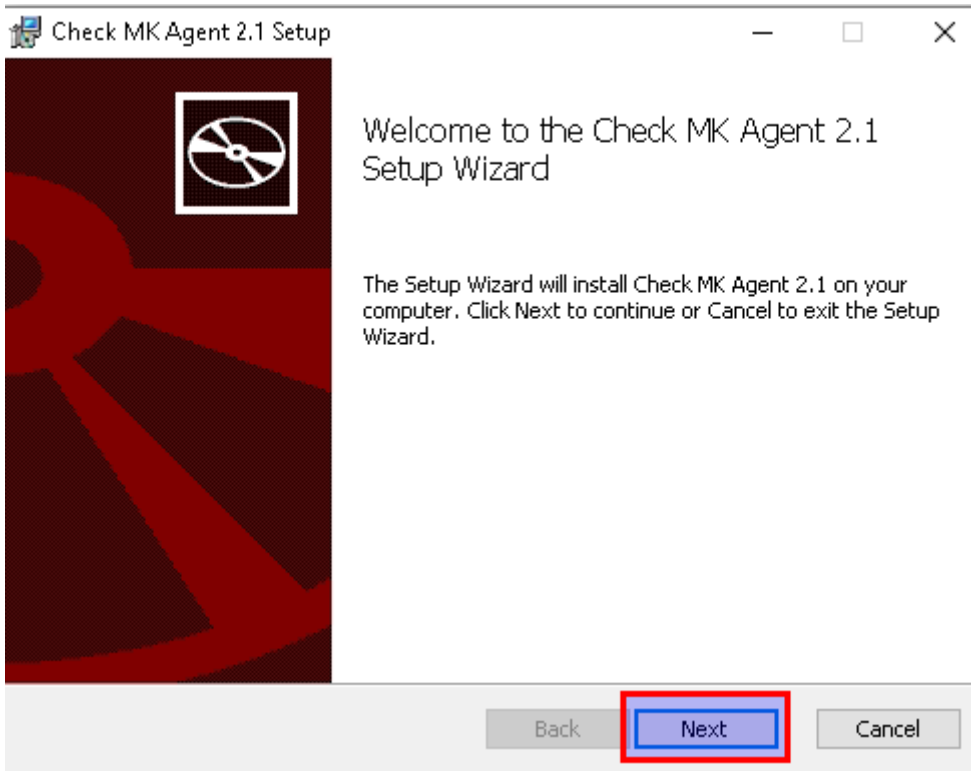
Nun erscheint der Download Dialog eures Webbrowsers kann bei jedem anders aussehen, auf jeden Fall die Datei speichern.



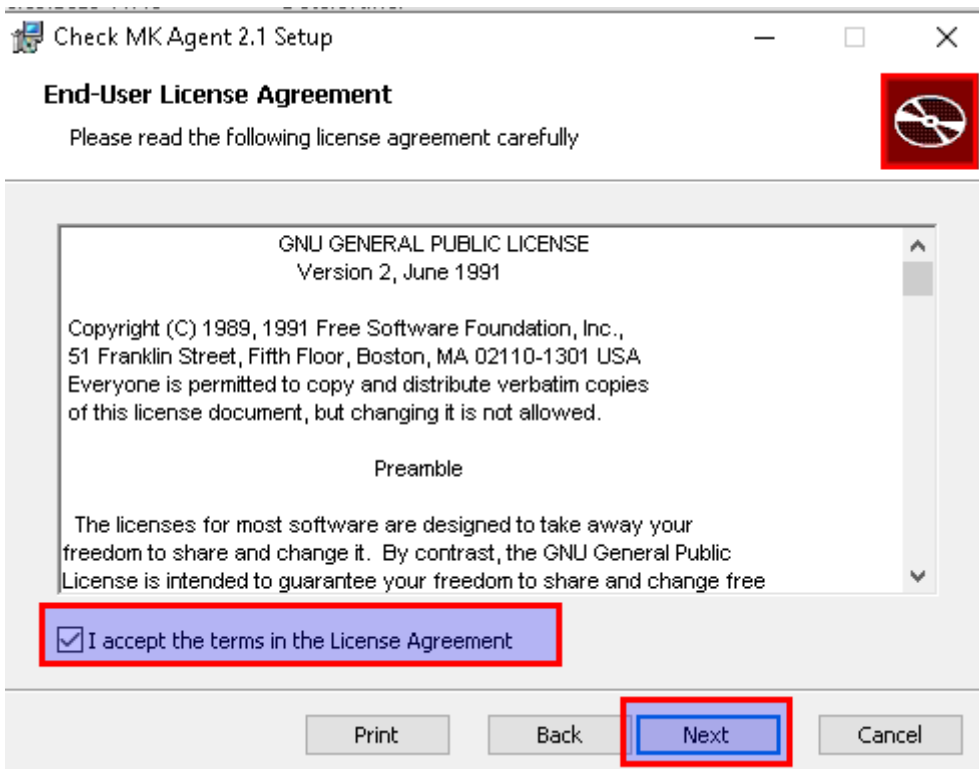
Nun liegt diese Datei im Download Ordner. Dort ein Doppelklick drauf



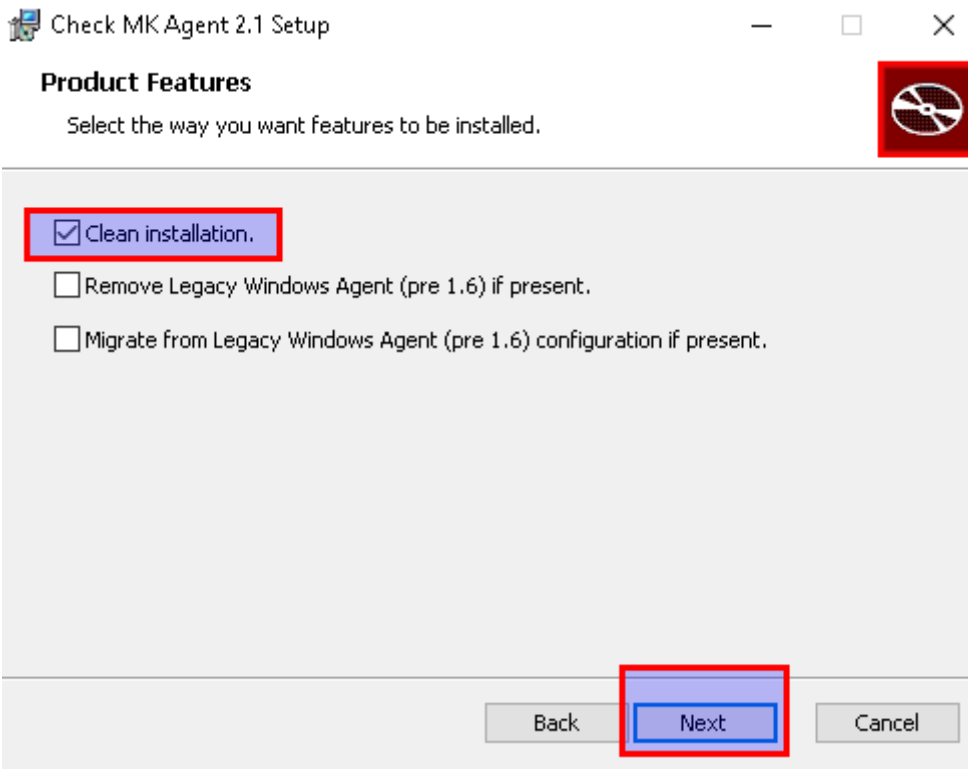
Nun startet das Setup, auf Next klicken



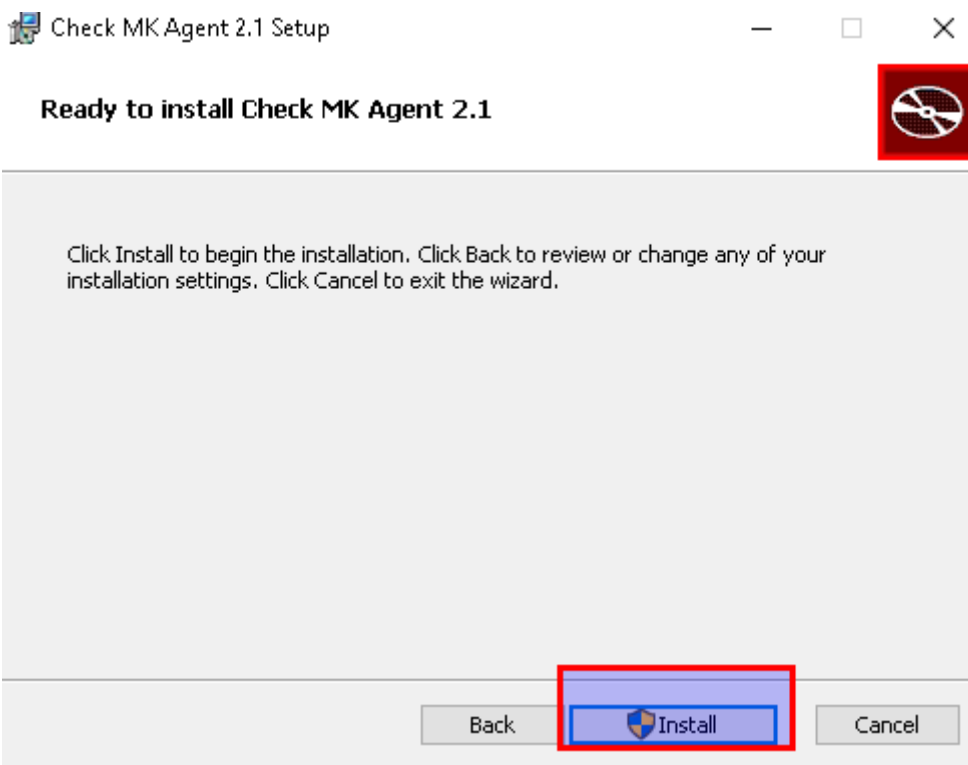
Lizenbedingungen akzeptieren und auf Next klicken



Clean install auswählen, alles andere abwählen und auf next klicken



Nun auf Install klicken



Nun auf Finish klicken



Completed the Check MK Agent 2.1 Setup Wizard

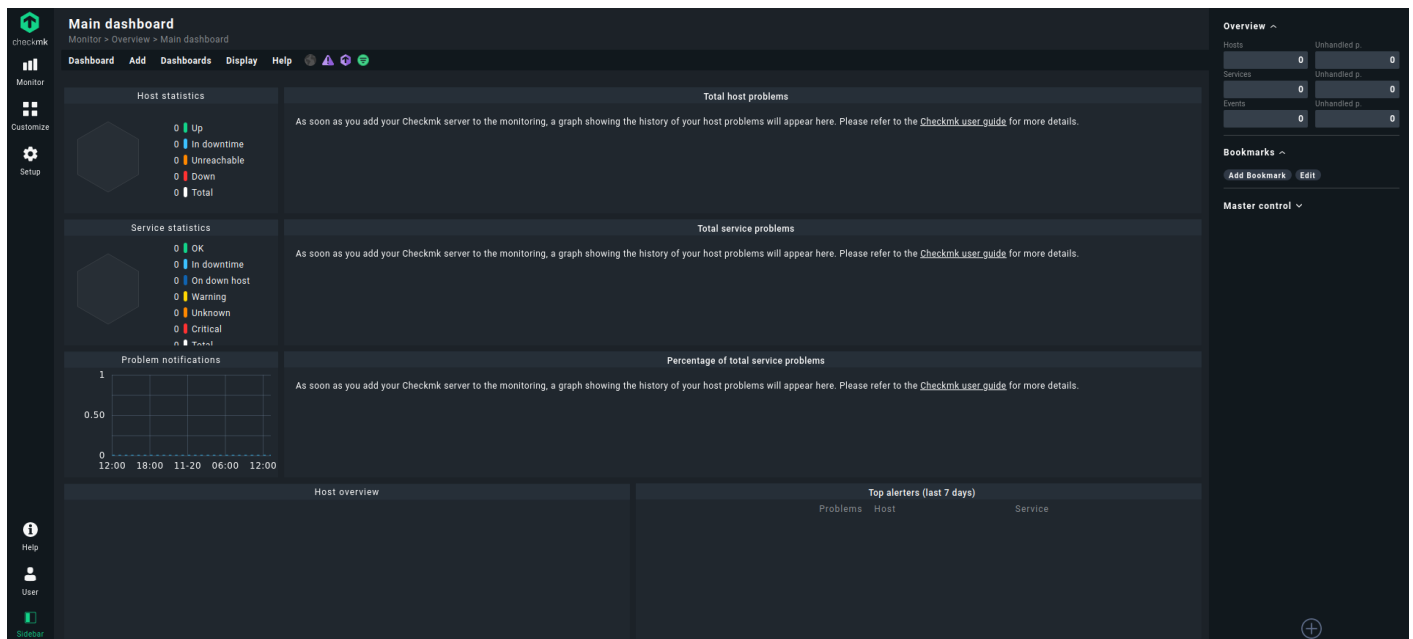
Click the Finish button to exit the Setup Wizard.

Back **Finish** Cancel

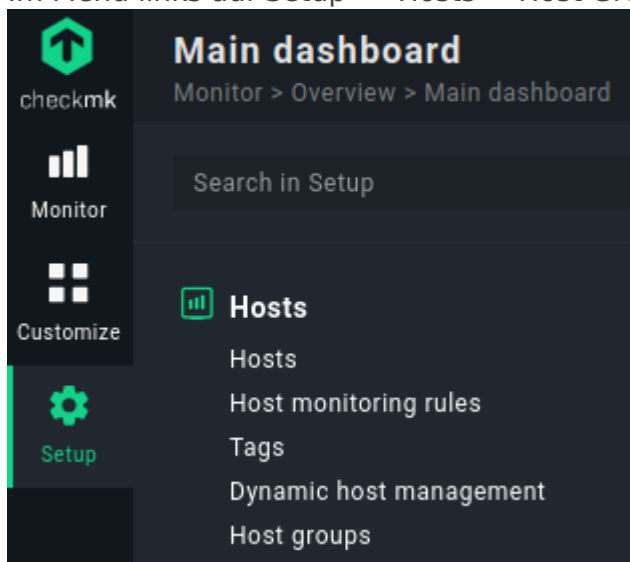
Registrieren

Anlegen von Hostgruppen und deren Verzeichnisse

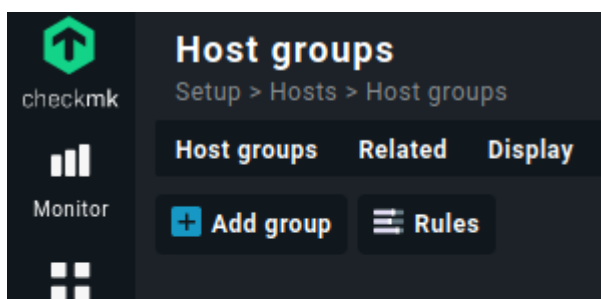
Nach dem Einloggen sehen wir die Hauptseite



Im Menü links auf Setup -> Hosts -> Host Groups.



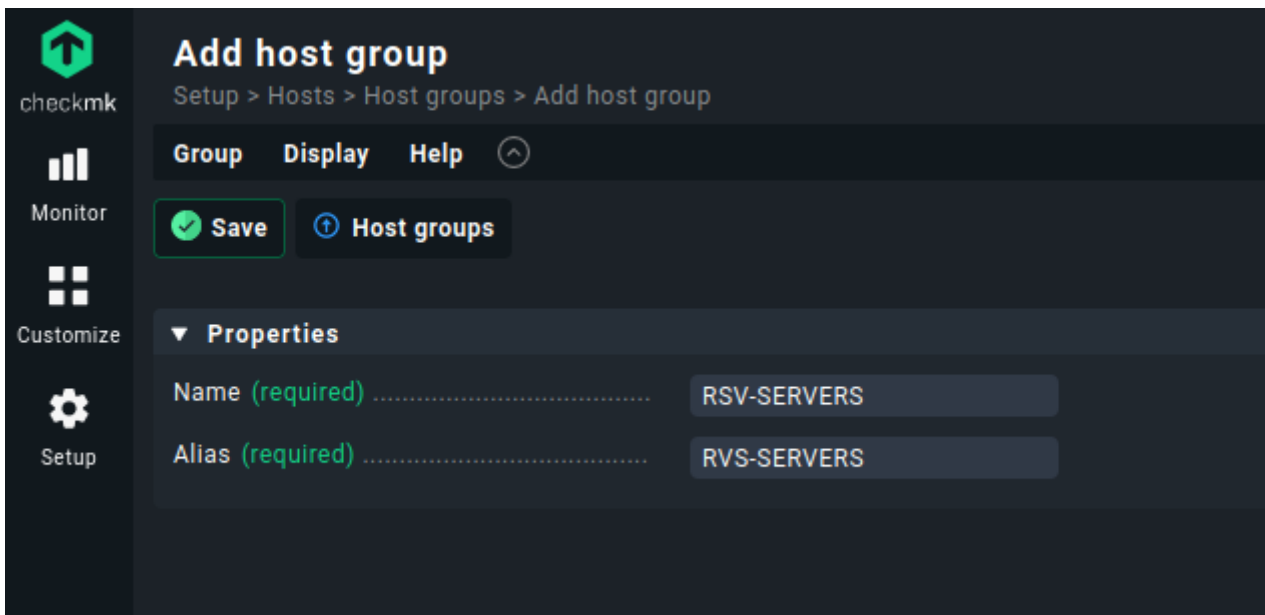
Nun auf den Button dd group klicken



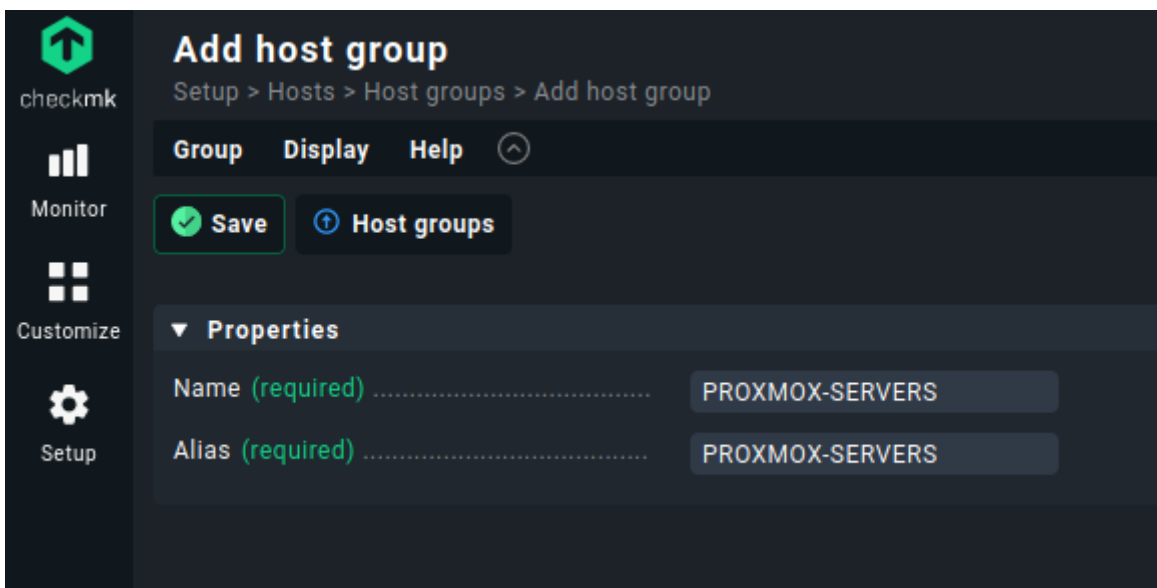
und einen Gruppennamen vergeben.

bei uns RVS-SERVERS

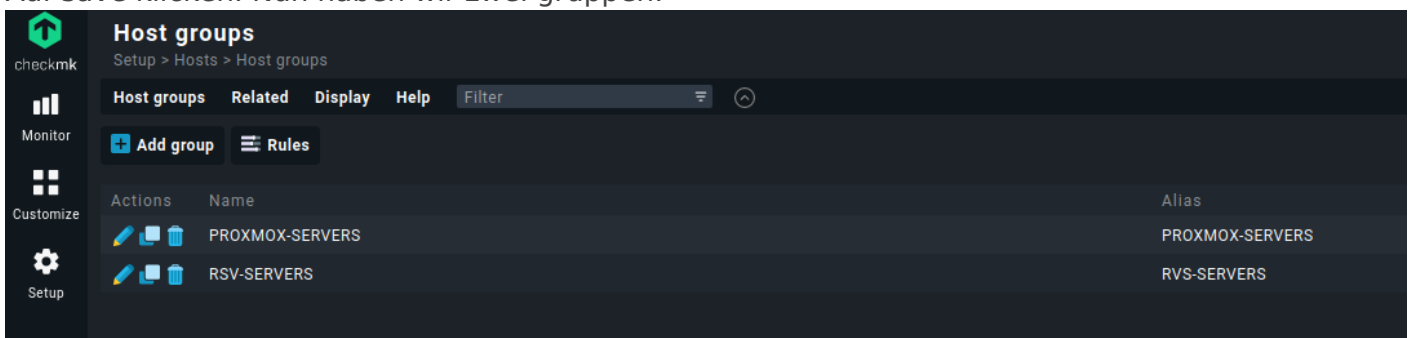
Dann auf Save



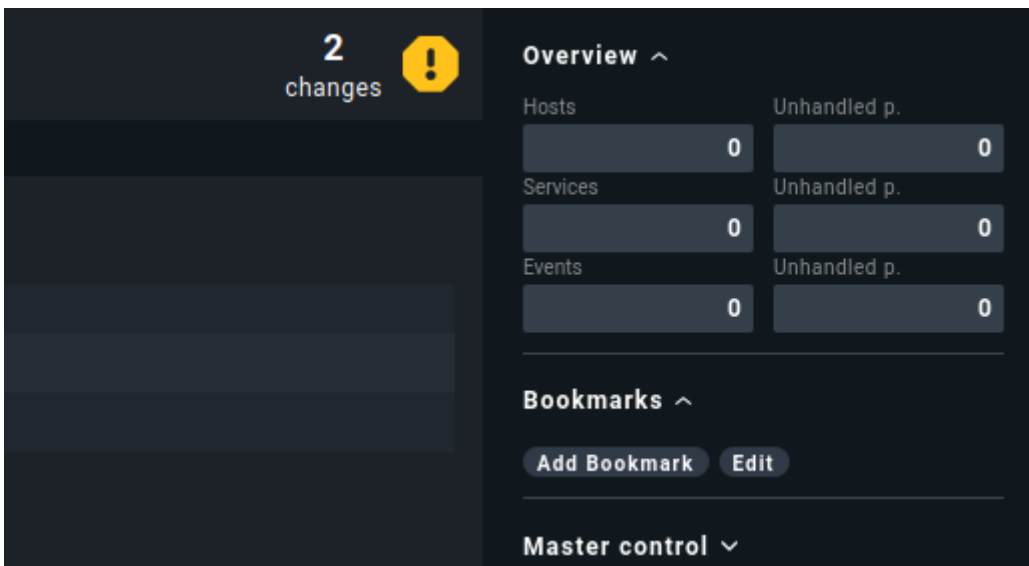
Nun eine weitere Gruppe anlegen namens PROXMOX-SERVERS



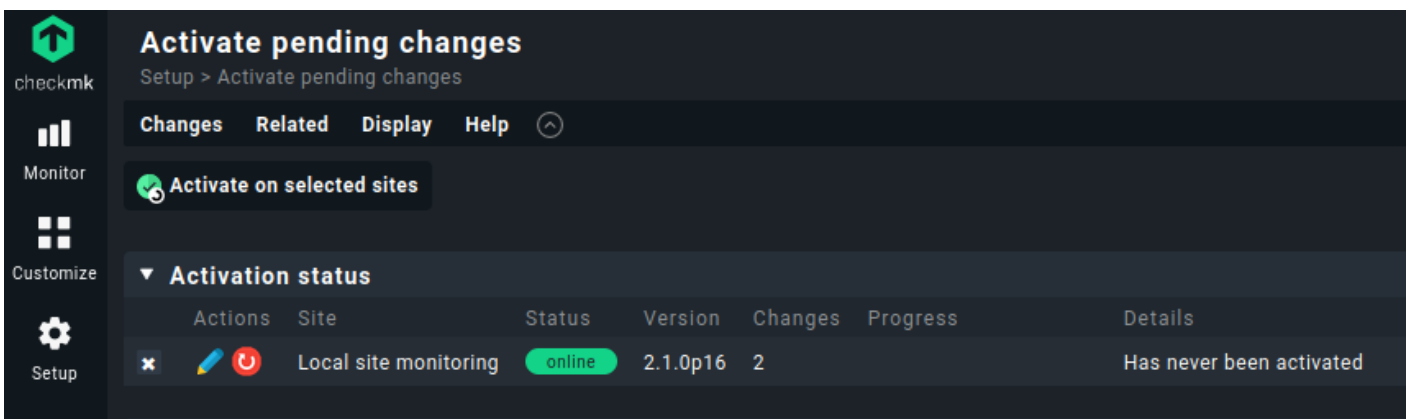
Auf save klicken. Nun haben wir zwei gruppen.



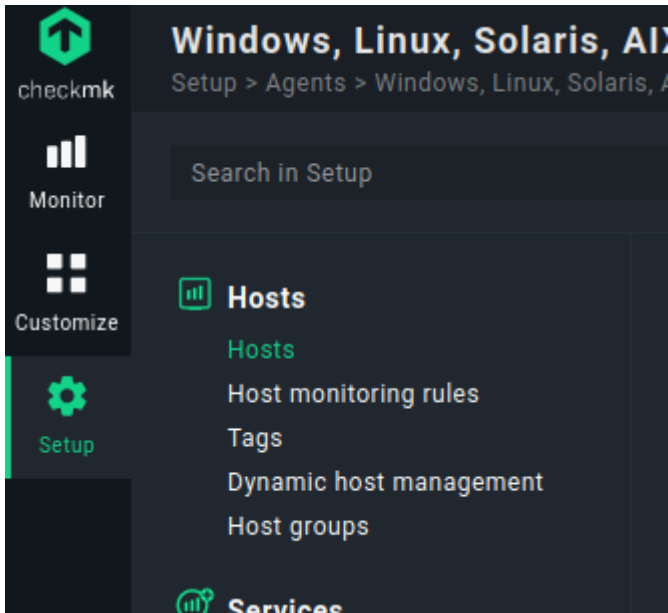
Nun oben rechts auf den Button changes klicken.
Denn unsere Änderungen die wir gemacht haben sind noch nicht aktiv.
Die Zahl zeigt an wie viele Änderungen wir gemacht haben.



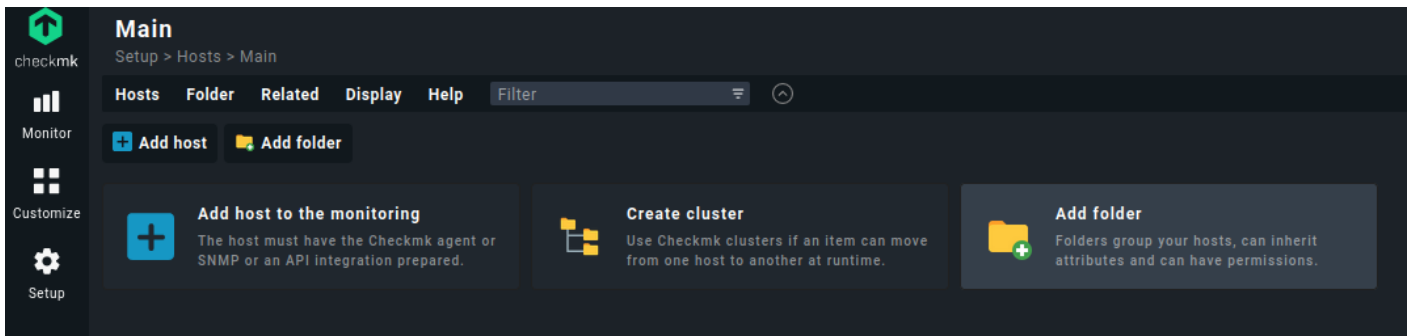
Nun auf Activate on selected sites klicken



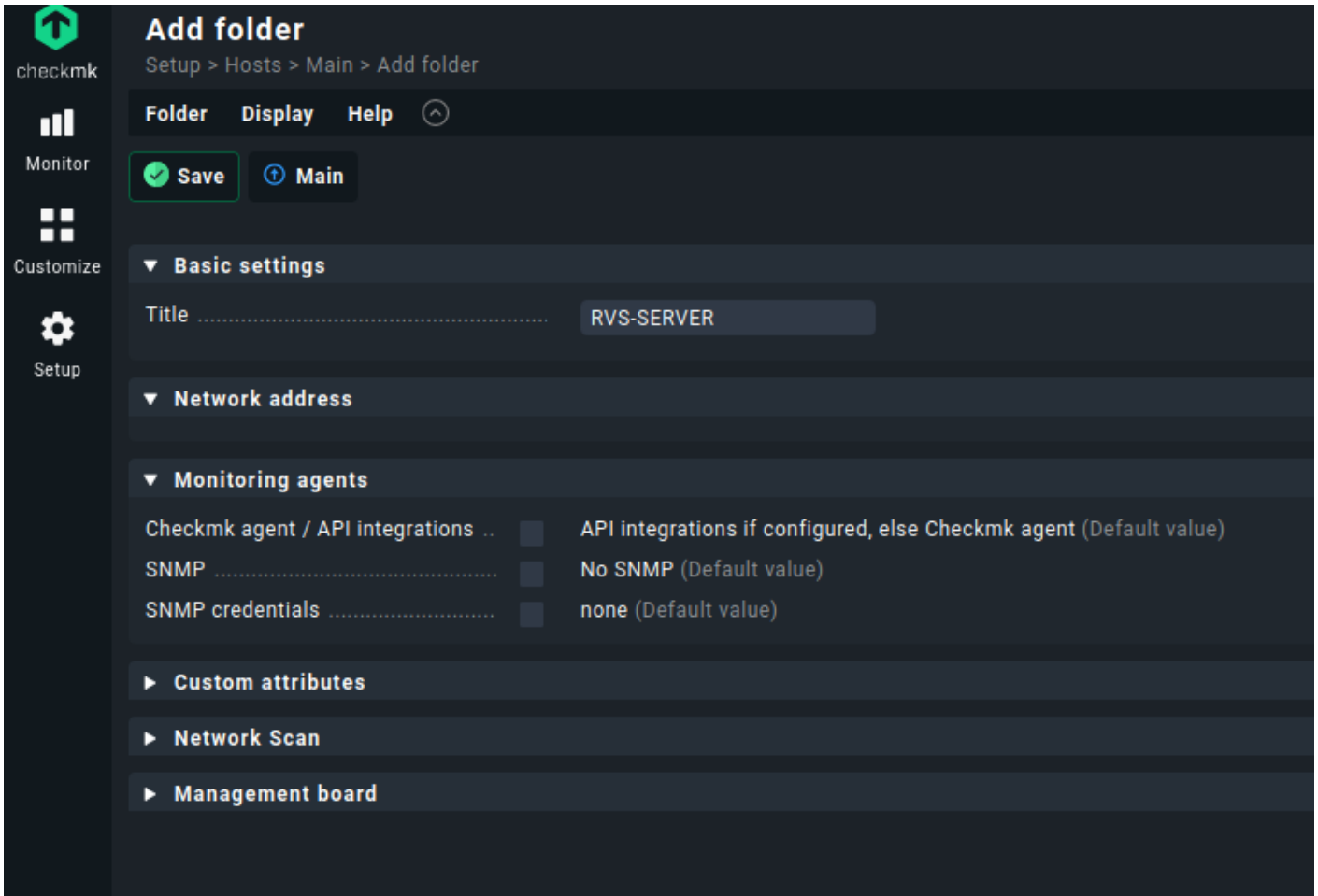
Nun gehen wir wieder auf Setup -> Hosts -> Hosts



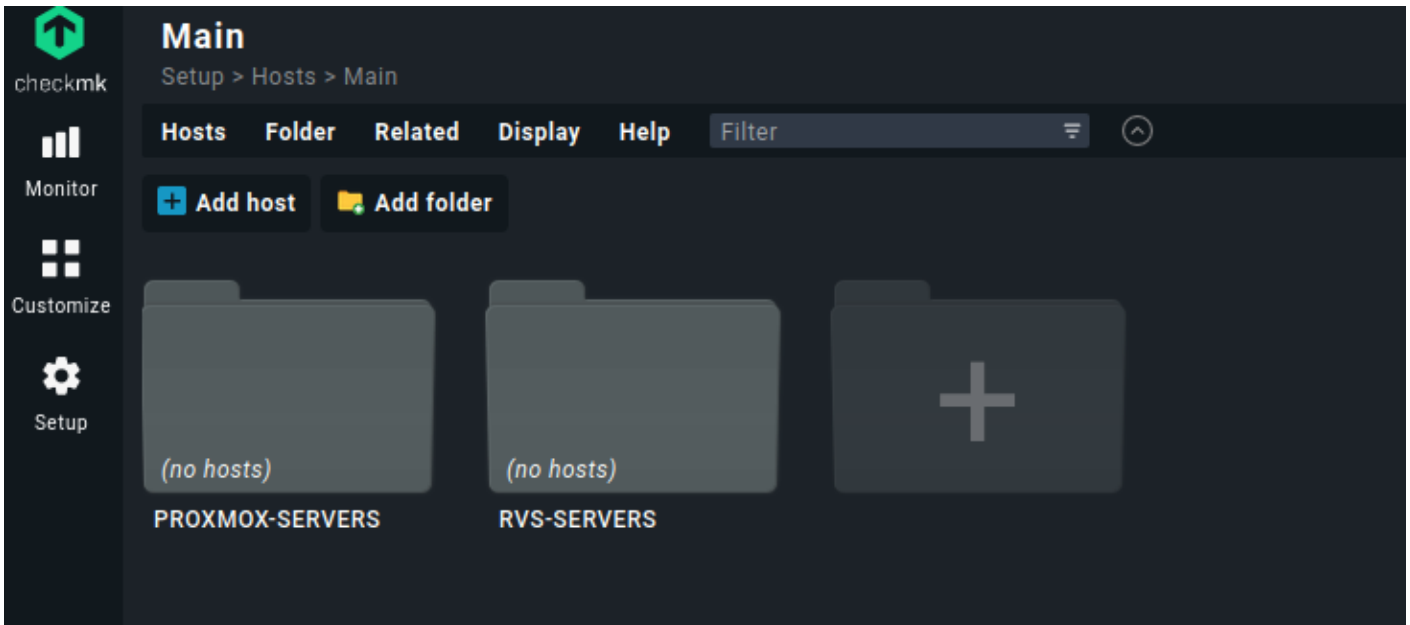
Dort erstellen wir Verzeichnisse mit den gleichen Namen wie die Gruppen.
Dazu klicken wir auf Add folder




Nun den Verzeichnisnamen eingeben. RVS-SERVERS (im Screenshot steht server hab das s vergessen)



Das gleiche auch für PROXMOX-SERVERS
Dann sieht das ganze so aus.



Wieder Änderungen übernehmen

2 changes 

Overview ^


Hosts	0	Unhandled p.	0
Services	0	Unhandled p.	0
Events	0	Unhandled p.	0

Bookmarks ^

[Add Bookmark](#) [Edit](#)

Master control v

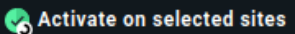
Nun auf Activate on selected sites klicken

 checkmk




Activate pending changes

Setup > Activate pending changes

Changes Related Display Help ↻



Activation status

Actions	Site	Status	Version	Changes	Progress	Details
  	Local site monitoring	online	2.1.0p16	2		Has never been activated

Fertig.

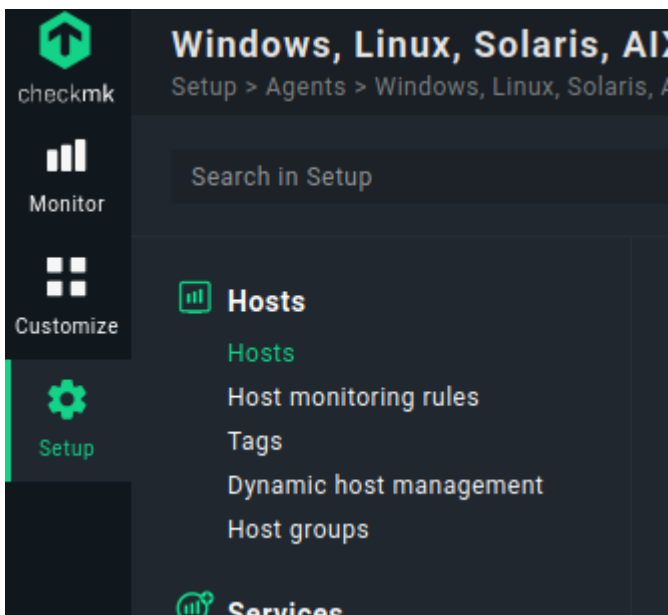
Anlegen von Hosts

Nun die Hosts anlegen.

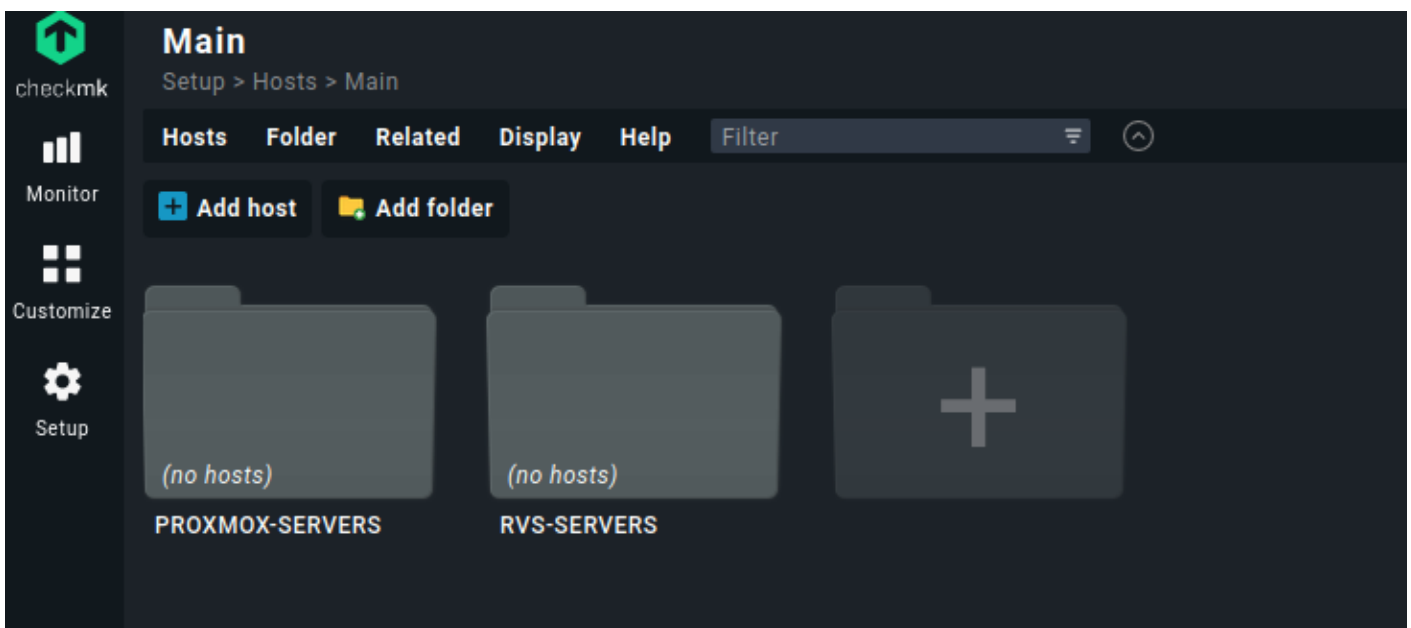
Falls noch nicht getan die Agents auf die Hosts die überwacht werden sollen installieren

Siehe abschnitt Einrichtung

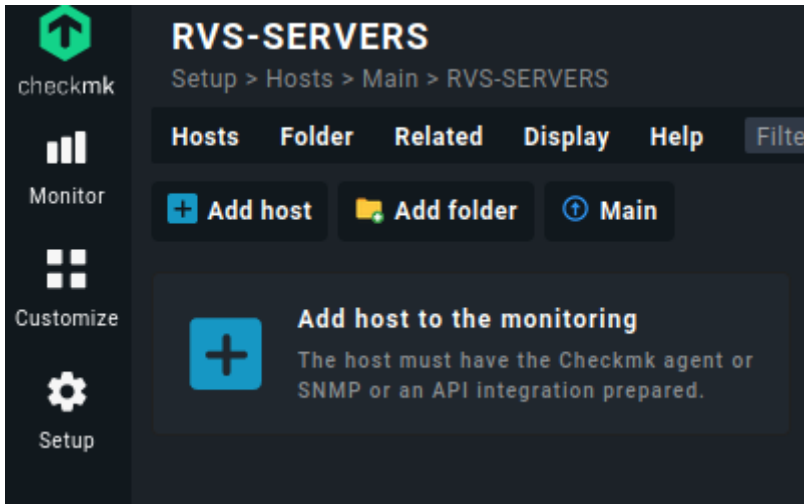
Nun Auf -> Setup -> Hosts klicken



Nun haben wir eine Übersicht unserer Verzeichnisse



In das Verzeichnis reinklicken wo wir gerne den / die Host(s) erstellen möchten und auf Add Host to the monitoring klicken.



Nun auszufüllen

Hostname : name des hostes rein.

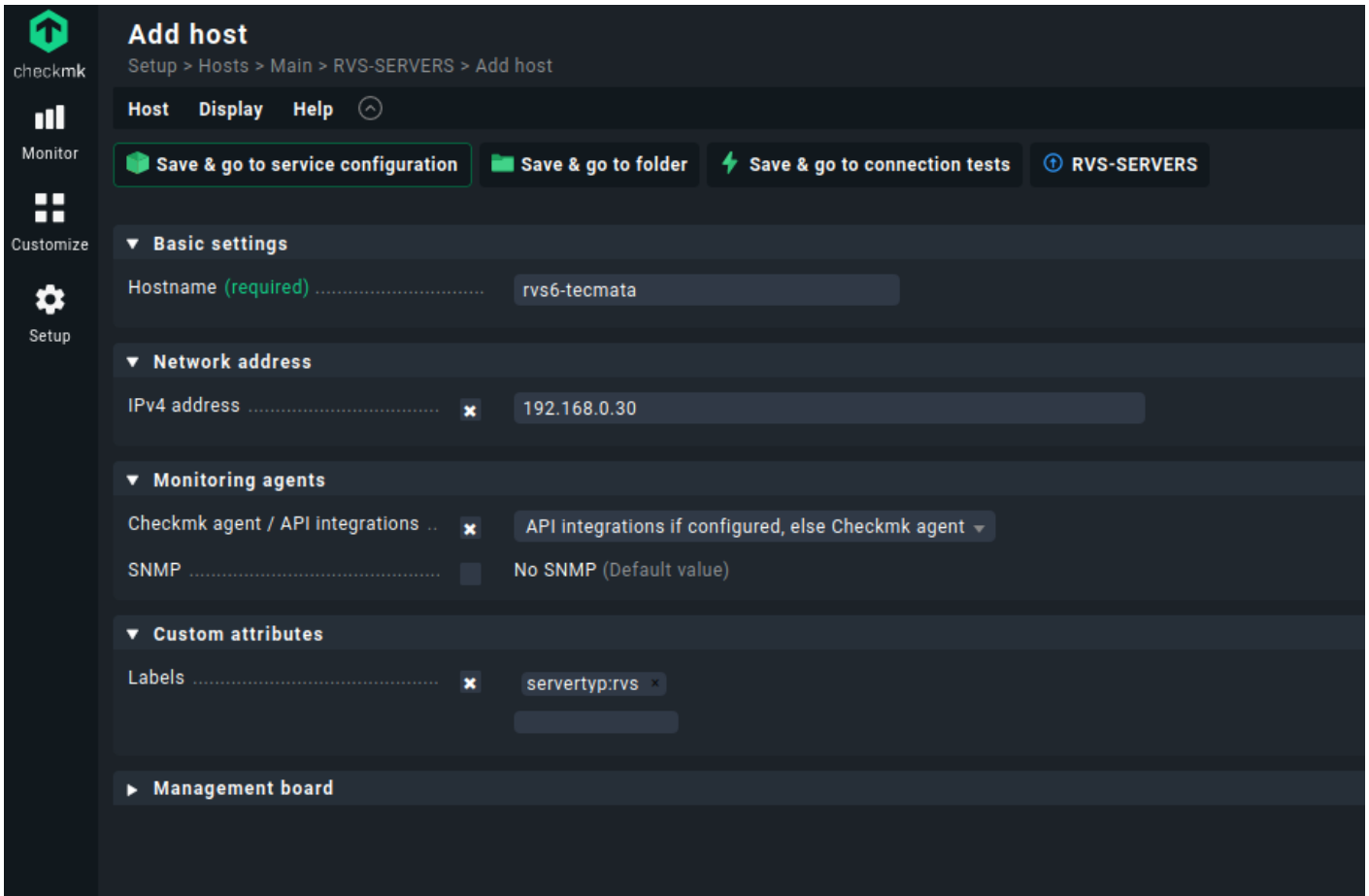
IPv4 Adresse Haken rein und die ip Adresse eintargen

Checkmk agent / API integration : haken rein

Label : sinnvolle label vergeben, können auch mehrer sein.

Wir definieren hier ein neues Label mit einem wert servertyp : rvs

Nun auf Save & go to services configuration



Nun bekommen wir eine Liste mit allen Diensten.

Es werden auch zwei neue Labels erstellt: devicetyp und os_family

Wenn wir Dienste nicht überwacht haben wollen klicken wir auf das rote x bei dem jeweiligen Dienst

Hier gibt es eine Checkmk Warnung weil wir den Agent noch nicht registriert haben, kommt später.

checkmk
Monitor
Customize
Setup

Services of host rvs6-tecmata

Setup > Hosts > Main > RVS-SERVERS > Properties of host rvs6-tecmata > Services of host rvs6-tecmata

Actions Host Settings Display Help

Accept all Rescan Monitor undecided services Remove vanished services Properties of host rvs6-tecmata

Undecided services: 26 | Vanished services: 0 | New host labels: 2 | Vanished host labels: 0 | Changed host labels: 0

Discovered host labels (2)

Status Host labels

New cmk/device_type:vm cmk/os_family:linux

Undecided services - currently not monitored (26)

	State	Service	Status detail
+ x	WARN	Check_MK Agent	Version: 2.1.0p16, OS: linux, TLS is not activated on monitored host (see details) WARN , Agent plugins: 0, Local checks: 0
+ x	OK	CPU load	15 min load: 0.08, 15 min load per core: 0.02 (4 cores)
+ x	OK	CPU utilization	Total CPU: 1.79%
+ x	OK	Disk IO SUMMARY	Initializing counters
+ x	OK	Filesystem /	37.25% used (1.81 of 4.85 GB)
+ x	OK	Filesystem /var	7.66% used (2.93 of 38.28 GB)
+ x	OK	Filesystem /var/efw	17.51% used (20.74 of 118.46 MB)
+ x	OK	Filesystem /var/log	6.8% used (1.74 of 25.54 GB)
+ x	OK	Interface 02	[br0], (up), MAC: 06:E2:B6:D9:53:1B, Speed: 1 GBit/s
+ x	OK	Interface 05	[ha0], (up), MAC: 0E:F8:33:DA:82:58, Speed: 1 GBit/s
+ x	OK	Interface 06	[eth0], (up), MAC: 06:E2:B6:D9:53:1B, Speed: 1 GBit/s
+ x	OK	Interface 07	[eth1], (up), MAC: 92:24:F2:76:79:CF, Speed: 1 GBit/s
+ x	OK	Interface 08	[eth2], (up), MAC: 46:3D:B3:A0:E3:80, Speed: 1 GBit/s
+ x	OK	Interface 09	[tap0], (up), MAC: FE:EC:72:A8:55:36, Speed: 10 MBit/s
+ x	OK	Interface 10	[tap2], (up), MAC: 86:54:4F:91:15:14, Speed: 10 MBit/s
+ x	OK	Kernel Performance	Counter initialization

Help
User
Setup

Wir sehen hier einen Fehler beim Dienst Postfix und beim systemd clam av.

+ x	OK	Kernel Performance	Counter initialization
+ x	OK	Memory	Total virtual memory: 21.95% - 1.26 GB of 5.74 GB, 8 additional details available
+ x	OK	Mount options of /	Mount options exactly as expected
+ x	OK	Mount options of /var	Mount options exactly as expected
+ x	OK	Mount options of /var/efw	Mount options exactly as expected
+ x	OK	Mount options of /var/log	Mount options exactly as expected
+ x	OK	Number of threads	409, Usage: 1.31%
+ x	CRIT	Postfix status	Status: the Postfix mail system is not running
+ x	CRIT	Systemd Service Summary	Total: 142, Disabled: 28, Failed: 5, 2 services failed (check-mk-agent, rngd) CRIT , 2 static services failed (check-mk-agent@, efw-clamav-cleanup) CRIT
+ x	OK	TCP Connections	Established: 5
+ x	OK	Uptime	Up since Nov 18 2022 08:22:42, Uptime: 2 days 9 hours

Help
User
Setup

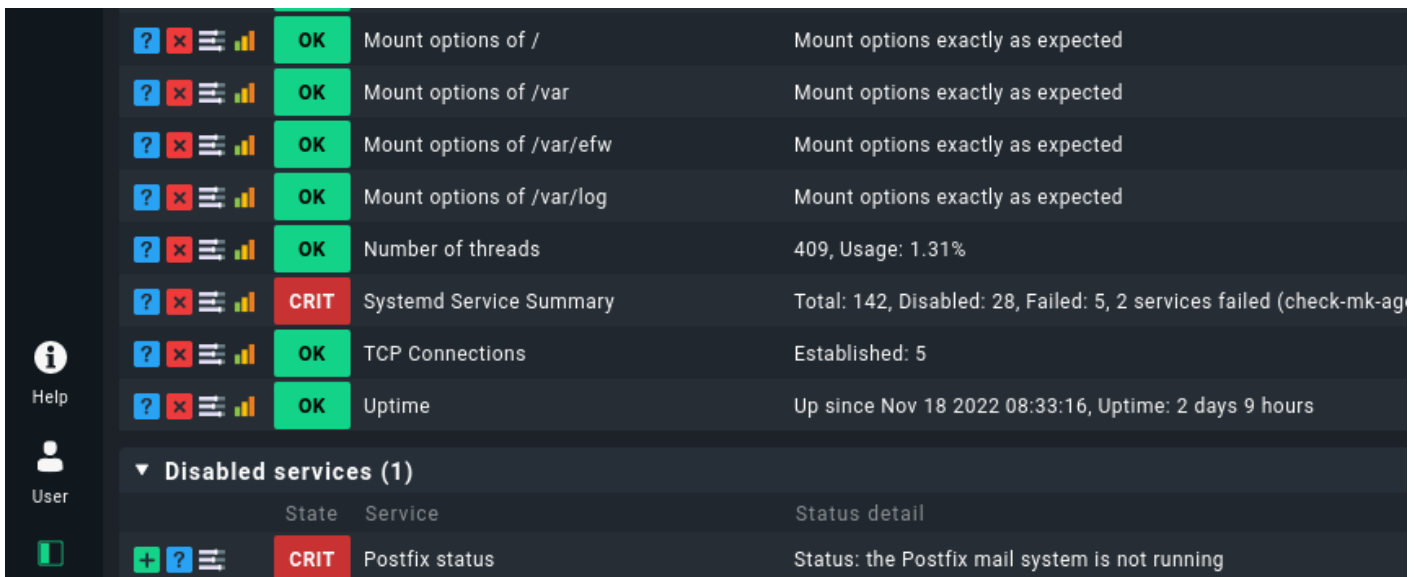
Den Postfix fehler bekommen wir auch nicht weg, da auf diesem System Postfix unkonfiguriert ist. Entweder postfix konfigurieren wenn benötigt oder den Dienst von der Überwachung ausschließen. Wir klicken hier auf das rote X bei Postfix, weil wir den Dienst hier nicht brauchen.

Um die in der Vergangenheit liegenden Fehler von Systemd zu löschen einfach im Terminal mit dem Befehl.

```
systemctl reset-failed
```

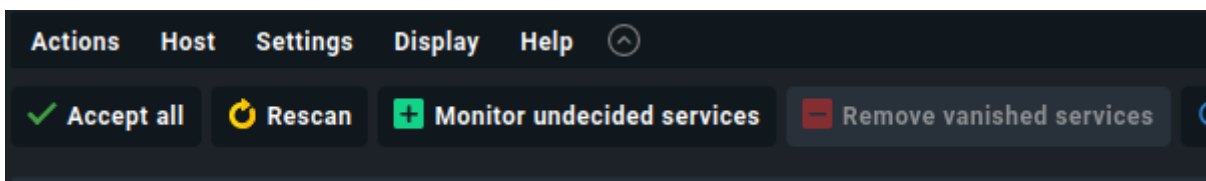
Das nur so nebenbei.

Nun ist der Postfix Dienst ausgeschlossen, weil wir diesen ja mit dem roten X entfernt haben. Dieser steht unter Disabled Service, durch klick auf das grüne plus können wir diesen jederzeit wieder hinzufügen. Auch später, falls wir uns entscheiden doch postfix auf dem System zu nutzen



State	Service	Status detail
OK	Mount options of /	Mount options exactly as expected
OK	Mount options of /var	Mount options exactly as expected
OK	Mount options of /var/efw	Mount options exactly as expected
OK	Mount options of /var/log	Mount options exactly as expected
OK	Number of threads	409, Usage: 1.31%
CRIT	Systemd Service Summary	Total: 142, Disabled: 28, Failed: 5, 2 services failed (check-mk-ag)
OK	TCP Connections	Established: 5
OK	Uptime	Up since Nov 18 2022 08:33:16, Uptime: 2 days 9 hours
Disabled services (1)		
CRIT	Postfix status	Status: the Postfix mail system is not running

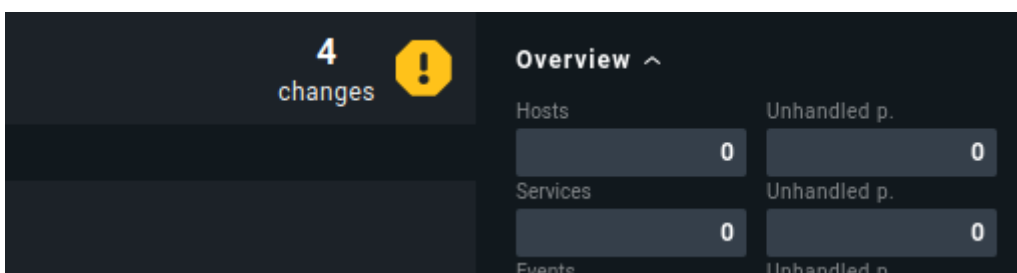
Wenn alle Dienste aussortiert, die nicht gebraucht werden. Hier nur postfix dann auf Accept all klicken.



Actions Host Settings Display Help

Accept all Rescan Monitor undecided services Remove vanished services

Nun die Änderungen wieder bestätigen

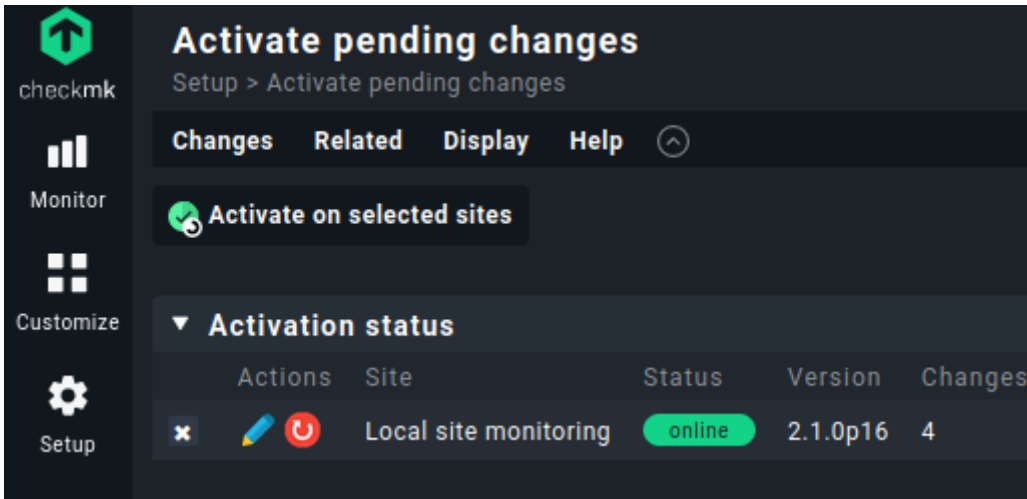


4 changes

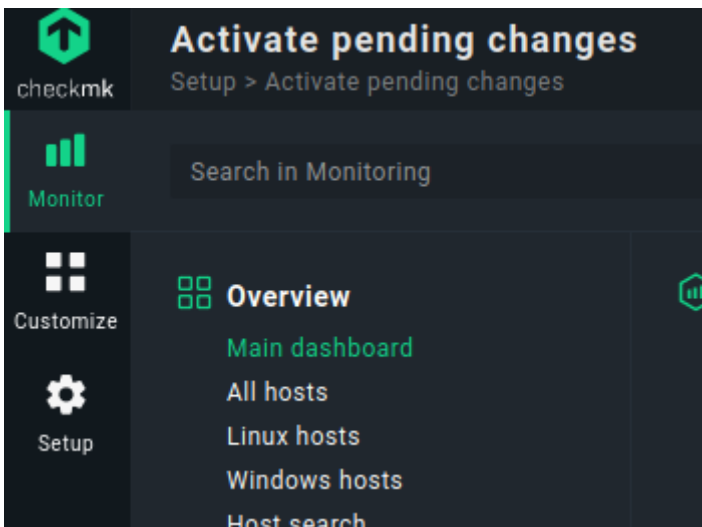
Overview ^

Hosts	0	Unhandled p.	0
Services	0	Unhandled p.	0
Events		Unhandled p.	

Wieder active on selected sites



Nun Auf Monitor und dann Main Dashboard.



Ein Dienst Critical, und eine Warnung. Die Warnung kommt noch vom nicht registriertem Client. Das kommt im nächsten Abschnitt, denn damit ein Client registriert werden kann muss er erst im System angelegt sein, was wir in diesem Abschnitt gemacht haben



checkmk

Main dashboard

Monitor > Overview > Main dashboard



Monitor



Customize



Setup

Dashboard Add Dashboards Display Help

Host statistics



1 Up
0 In downtime
0 Unreachable
0 Down
1 Total

As soon as you add your Checkmk server to the monitoring, a

Service statistics



25 OK
0 In downtime
0 On down host
1 Warning
0 Unknown
1 Critical
27 Total

As soon as you add your Checkmk server to the monitoring, a

Problem notifications



As soon as you add your Checkmk server to the monitoring, a

Host overview

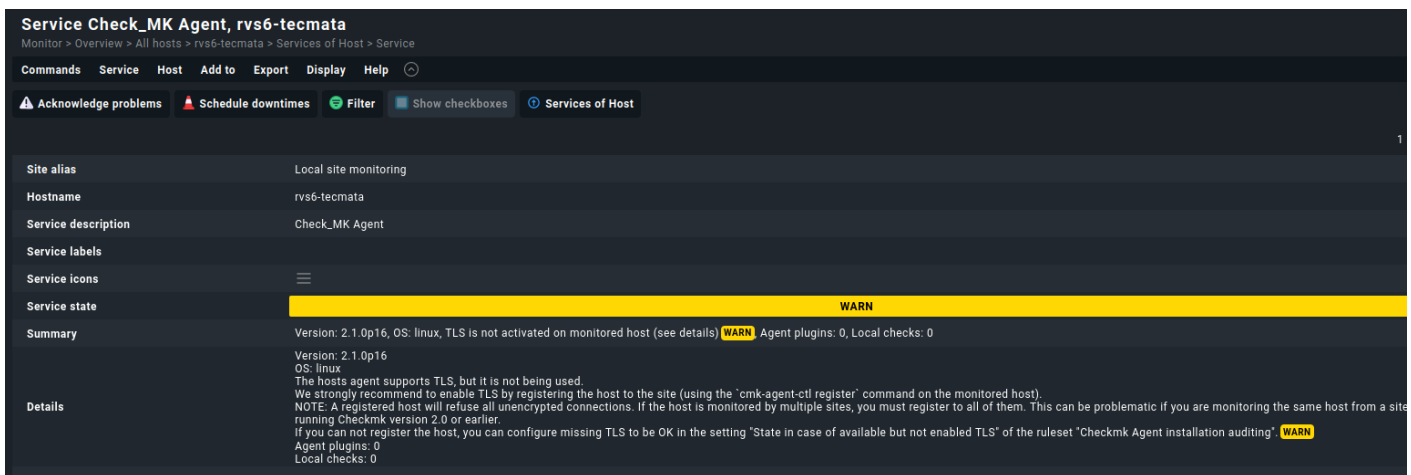
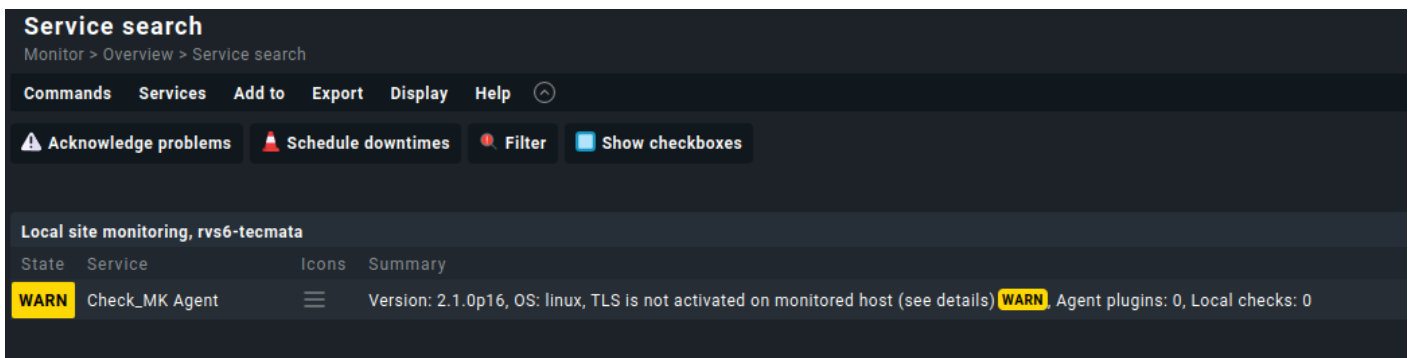
Agents am CheckMK Server registrieren

Beschreibung:

Wenn im Service AGENT diese Warnung steh, ist der Host mit seinem agent beim Checkmk Server noch nicht registriert.

Der host wird erst im Checkmk angelegt und dann übers Terminal registriert. Ansonsten steht diese Meldung hier.

Denn der Host kann den Agent ja erreichen, aber unverschlüsselt und für jeden.



Warum das ganze?

Wenn der Agent unverschlüsselt läuft, kann jeder über den Port Daten vom Host abgreifen

Befehl:

```
telnet 192.168.0.34 6556
```

Die Ausgabe

```
<<<check_mk>>>
Version: 2.1.0p16
AgentOS: linux
Hostname: rvs6-tecmata
AgentDirectory: /etc/check_mk
DataDirectory: /var/lib/check_mk_agent
SpoolDirectory: /var/lib/check_mk_agent/spool
PluginsDirectory: /usr/lib/check_mk_agent/plugins
LocalDirectory: /usr/lib/check_mk_agent/local
FailedPythonReason:
SSHClient:
<<<cmk_agent_ctl_status:sep(0)>>>
{"version":"2.1.0p16","agent_socket_operational":true,"ip_allowlist":[],"allow_legacy_pull":true,"connections":[]}
<<<checkmk_agent_plugins_lnx:sep(0)>>>
pluginsdir /usr/lib/check_mk_agent/plugins
localdir /usr/lib/check_mk_agent/local
<<<labels:sep(0)>>>
<<<df>>>
tmpfs          tmpfs  1616236  1568  1614668  1% /run
/dev/mapper/rvs6test--vg-root ext4  243008916 169841472 60750420 74% /
tmpfs          tmpfs  8081168  46800  8034368  1% /dev/shm
tmpfs          tmpfs  5120    0    5120    0% /run/lock
/dev/nvme0n1p2 ext2  481642  124858  331799  28% /boot
/dev/nvme0n1p1 vfat  523248  3512  519736  1% /boot/efi
/dev/fuse      fuse  131072  52  131020  1% /etc/pve
tmpfs          tmpfs  1616232  0  1616232  0% /run/user/0
<<<df>>>
[df_inodes_start]
tmpfs          tmpfs  2020292  1131  2019161  1% /run
/dev/mapper/rvs6test--vg-root ext4  15507456 61994 15445462 1% /
tmpfs          tmpfs  2020292  106  2020186  1% /dev/shm
tmpfs          tmpfs  2020292  21  2020271  1% /run/lock
/dev/nvme0n1p2 ext2  124928  352  124576  1% /boot
/dev/nvme0n1p1 vfat  0  0  0  - /boot/efi
/dev/fuse      fuse  262144  38  262106  1% /etc/pve
tmpfs          tmpfs  404058  20  404038  1% /run/user/0
```

```
[df_inodes_end]
[df_lsblk_start]
{
  "blockdevices": [
    {"name":"/dev/sda", "uuid":null},
    {"name":"/dev/sda1", "uuid":"77085686-971d-498a-8692-0fd8048da3a0"},
    {"name":"/dev/sr0", "uuid":null},
    {"name":"/dev/mapper/nvme0n1p3_crypt", "uuid":"ZZLFwZ-JMTb-Qdcu-akhk-5gEc-d2kd-6idwqE"},
    {"name":"/dev/mapper/rvs6test--vg-root", "uuid":"da0060ca-629b-4f24-96e4-2bc3dd1dde9a"},
    {"name":"/dev/mapper/rvs6test--vg-swap_1", "uuid":"08efe098-7e3a-4ab0-b961-5154eb194d89"},
    {"name":"/dev/nvme0n1", "uuid":null},
    {"name":"/dev/nvme0n1p1", "uuid":"04B7-49AB"},
    {"name":"/dev/nvme0n1p2", "uuid":"0c954140-e9ed-457d-8833-f4564326cdd5"},
    {"name":"/dev/nvme0n1p3", "uuid":"663f7527-3144-4b54-9768-454e65aad4b4"}
  ]
}
[df_lsblk_end]
<<<systemd_units>>>
[list-unit-files]

...
<<<timesyncd>>>
  Server: 129.70.132.34 (2.debian.pool.ntp.org)
Poll interval: 34min 8s (min: 32s; max 34min 8s)
  Leap: normal
  Version: 4
  Stratum: 2
  Reference: 81468952
  Precision: 1us (-24)
Root distance: 3.287ms (max: 5s)
  Offset: +2.726ms
  Delay: 15.787ms
  Jitter: 5.769ms
Packet count: 2634
  Frequency: -9.423ppm
[[[1674394719]]]
<<<timesyncd_ntpmessage:sep(10)>>>
NTPMessage={ Leap=0, Version=4, Mode=4, Stratum=2, Precision=-24, RootDelay=381us,
RootDispersion=3.097ms, Reference=81468952, OriginateTimestamp=Sun 2023-01-22 14:38:39 CET,
ReceiveTimestamp=Sun 2023-01-22 14:38:39 CET, TransmitTimestamp=Sun 2023-01-22 14:38:39 CET,
```

```
DestinationTimestamp=Sun 2023-01-22 14:38:39 CET, Ignored=no PacketCount=2634, Jitter=5.769ms }  
Timezone=Europe/Berlin  
<<<local:sep(0)>>>  
Connection closed by foreign host.
```

Nachdem der Server registriert wurde, sieht die Ausgabe nun so aus. Nun kann nur noch der Checkmk Server mit dem Zertifikat die Daten abrufen und keine dritten mehr. Denn das wäre ja noch so ohne Probleme gegangen, wie die obere Ausgabe ja zeigt.

```
oot@checkmk:~# telnet 192.168.0.34 6556  
Trying 192.168.0.34...  
Connected to 192.168.0.34.  
Escape character is '^]'.  
16  
  
Is still waiting here...  
No Output anymore
```

Wie registrieren wir den Server nun?

Unter Linux:

Dazu auf dem Server einloggen wo der Agent läuft und mit `cmk-agent-ctl` hinzufügen
Hier einmal die Hilfe die wir mit Parameter `-h` bekommen. Keine Panik, weiter unten zeige ich ein Beispiel.

Der Befehl

```
cmk-agent-ctl -h  
  
Ausgabe:  
  
cmk-agent-ctl 2.1.0p16  
Checkmk agent controller.  
  
USAGE:  
  cmk-agent-ctl <SUBCOMMAND>
```

FLAGS:

- h, --help Prints help information
- V, --version Prints version information

SUBCOMMANDS:

- daemon Run as daemon and handle all pull and push connections
- delete Delete a connection to a Checkmk instance
- delete-all Delete all connections to Checkmk sites
- dump Collect monitoring data and write it to standard output
- help Prints this message or the help of the given subcommand(s)
- import Import a pull connection from file or standard input
- proxy-register Register with a Checkmk site on behalf of another host
- pull Handle incoming connections from Checkmk sites collecting monitoring data
- push Push monitoring data to all Checkmk sites configured for 'push'
- register Register with a Checkmk site
- status Query the registration status of this host

Hier die Parameter für den Befehl register aus der Hilfe

```
cmk-agent-ctl register -h
```

Ausgabe:

```
cmk-agent-ctl-register 1.0.0
```

```
Register with a Checkmk site
```

Register with a Checkmk instance for monitoring. The required information can be read from a config file or must be passed via command line.

USAGE:

```
cmk-agent-ctl register [FLAGS] [OPTIONS]
```

FLAGS:

-d, --detect-proxy Detect and use proxy settings configured on this system for outgoing HTTPS connections.

The default is to ignore configured proxies and to connect directly

-h, --help Prints help information

--trust-cert Blindly trust the server certificate of the Checkmk site

--validate-api-cert Enable TLS certificate validation for querying the agent receiver port from the Checkmk

REST API. By default, certificate validation is disabled because it is not security-relevant at this stage, see werk #14715

-V, --version Prints version information
-v, --verbose Enable verbose output. Use once (-v) for logging level INFO and twice (-vv) for logging level DEBUG

OPTIONS:

-H, --hostname <host-name> Name of this host in the monitoring site
-P, --password <password> Password for API user. Can also be entered interactively
-s, --server <server> Address of the Checkmk site in the format "<server>" or "<server>:<port>"
-i, --site <site> Name of the Checkmk site
-U, --user <user> API user to use for registration

Welche Parameter brauchen wir?

Da wir ein selbstsigniertes Zertifikat haben brauchen wir folgende Parameter

```
--trust-cert  
--password <password>  
--server <server>  
--site <sitename/instanzname>  
--user <user>
```

Also würde unser Befehl so aussehen:

```
cmk-agent-ctl register --trust-cert --password *****usersicherspass**** --hostname rvs6-tecmata --server  
192.168.0.33 --site monitoring --user automation
```

Erläuterung der Parameter:

--password das automation user api password
--hostname der Hostname des Servers den wir registrieren wollen, dieser name muss identisch mit dem im checkmk sein, im Feld hostname hier rvs6-tecmata
--server : unser checkmkserver an dem wir den client registrieren wollen
--site unsere site / instanz auf dem checkmk server hier monitoring
--user der api user hier automation

Unter Windows

Eine Eingabeaufforderung mit Admin rechten Öffnen

```
Administrator: Eingabeaufforderung
Microsoft Windows [Version 10.0.19044.2604]
(c) Microsoft Corporation. Alle Rechte vorbehalten.
C:\WINDOWS\system32>
```

Nun den Client mit -h öffnen um die Parameter zu bekommen.

```
C:\WINDOWS\system32>"C:\Program Files (x86)\checkmk\service\cmk-agent-ctl.exe" -h
cmk-agent-ctl 2.1.0p16
Checkmk agent controller.

USAGE:
  cmk-agent-ctl.exe <SUBCOMMAND>

FLAGS:
  -h, --help    Prints help information
  -V, --version Prints version information

SUBCOMMANDS:
  daemon      Run as daemon and handle all pull and push connections
  delete      Delete a connection to a Checkmk instance
  delete-all  Delete all connections to Checkmk sites
  dump        Collect monitoring data and write it to standard output
  help        Prints this message or the help of the given subcommand(s)
  import      Import a pull connection from file or standard input
  proxy-register Register with a Checkmk site on behalf of another host
  pull        Handle incoming connections from Checkmk sites collecting monitoring data
  push        Push monitoring data to all Checkmk sites configured for 'push'
  register    Register with a Checkmk site
```

status Query the registration status of this host

```
C:\WINDOWS\system32>
```

Dann lassen wir uns die Hilfe für den Parameter register anzeigen

```
C:\WINDOWS\system32>"C:\Program Files (x86)\checkmk\service\cmk-agent-ctl.exe" register -h
cmk-agent-ctl.exe-register 1.0.0
Register with a Checkmk site
```

Register with a Checkmk instance for monitoring. The required information can be read from a config file or must be passed via command line.

USAGE:

```
cmk-agent-ctl.exe register [FLAGS] [OPTIONS]
```

FLAGS:

`-d, --detect-proxy` Detect and use proxy settings configured on this system for outgoing HTTPS connections.
 The default is to ignore configured proxies and to connect directly

`-h, --help` Prints help information

`--trust-cert` Blindly trust the server certificate of the Checkmk site

`--validate-api-cert` Enable TLS certificate validation for querying the agent receiver port from the Checkmk REST API. By default, certificate validation is disabled because it is not security-relevant at this stage, see werk #14715

`-V, --version` Prints version information

`-v, --verbose` Enable verbose output. Use once (-v) for logging level INFO and twice (-vv) for logging level DEBUG

OPTIONS:

`-H, --hostname <host-name>` Name of this host in the monitoring site

`-P, --password <password>` Password for API user. Can also be entered interactively

`-s, --server <server>` Address of the Checkmk site in the format "<server>" or "<server>:<port>"

`-i, --site <site>` Name of the Checkmk site

`-U, --user <user>` API user to use for registration

Welche Parameter brauchen wir?

Da wir ein selbstsigniertes Zertifikat haben brauchen wir folgende Parameter

```
--trust-cert  
--password <password>  
--server <server>  
--site <sitename/instanzname>  
--user <user>
```

Also würde unser Befehl so aussehen:

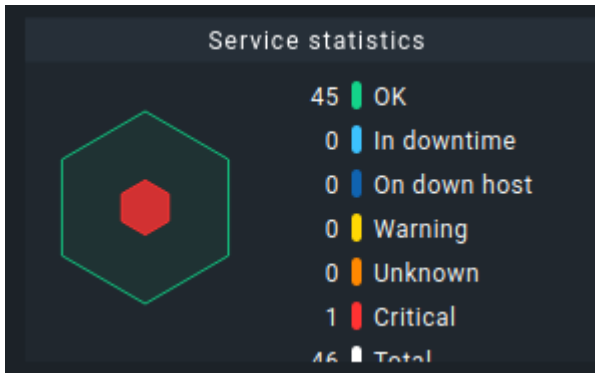
```
"C:\Program Files (x86)\checkmk\service\cmk-agent-ctl.exe" register --trust-cert --password  
*****unersischerspass***** --hostname rvs6-tecmata --server 192.168.0.33 --site monitoring --user automation
```

Wenn alles geklappt hat wird nichts als Ausgabe zurück gegeben.

Nun können wir den Status überprüfen:

```
"C:\Program Files (x86)\checkmk\service\cmk-agent-ctl.exe" status  
Version: 2.1.0p16  
Agent socket: operational  
IP allowlist: any  
  
Connection: ip:8000/monitoring  
  UUID: 00bf5de9-7d67-4c76-a9c5-886310581f19  
  Local:  
    Connection type: pull-agent  
    Certificate issuer: Site 'monitoring' local CA  
    Certificate validity: Wed, 08 Mar 2023 12:14:56 +0000 - Mon, 09 Jul 3021 12:14:56 +0000  
  Remote:  
    Connection type: pull-agent  
    Registration state: operational  
    Host name: jtl-wawiclient  
  
C:\WINDOWS\system32>
```

Nun ist die Warnung weg.



Ob dieser abschnitt noch benötigt wird weiß ich nicht, ich lasse es erstmal stehen

Bleibt der Fehler bestehen und es ist noch eine CRIT Warnung dazu gekommen die lautet [agent] Host is registered for TLS but not using it **CRIT**, Got no information from host **CRIT**, execution time 0.1 sec

Local site monitoring, rvs6-tecmata			
State	Service	Icons	Summary
CRIT	Check_MK	☰ 📧	[agent] Host is registered for TLS but not using it CRIT , Got no information from host CRIT , execution time 0.1 sec

The table displays monitoring data for 'Local site monitoring, rvs6-tecmata'. It has columns for State, Service, Icons, and Summary. A single row shows a 'CRIT' state for the 'Check_MK' service, with a summary message indicating a TLS-related error and an execution time of 0.1 seconds.

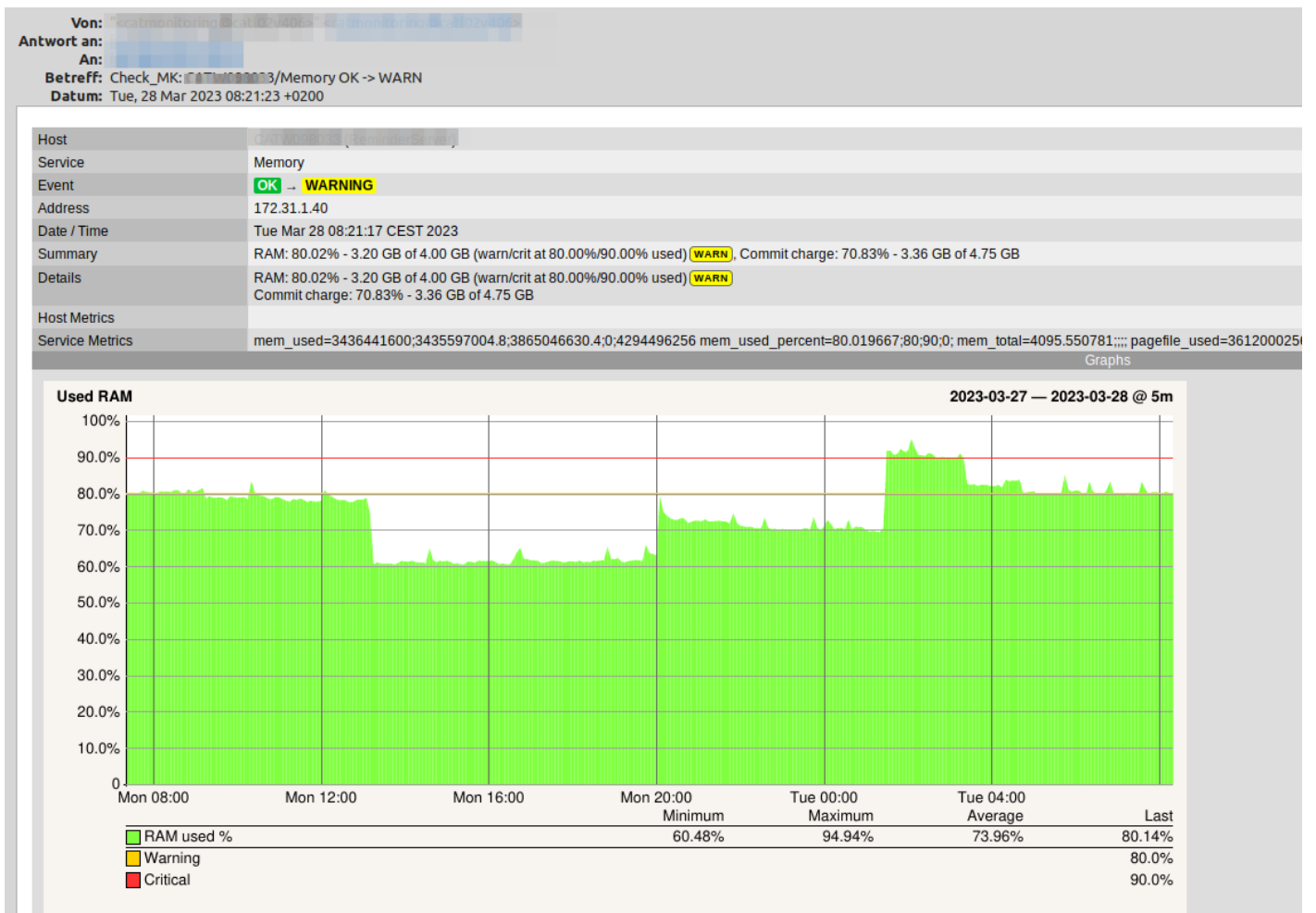
Checks Schwellenwerte / Limits anpassen

Beschreibung:

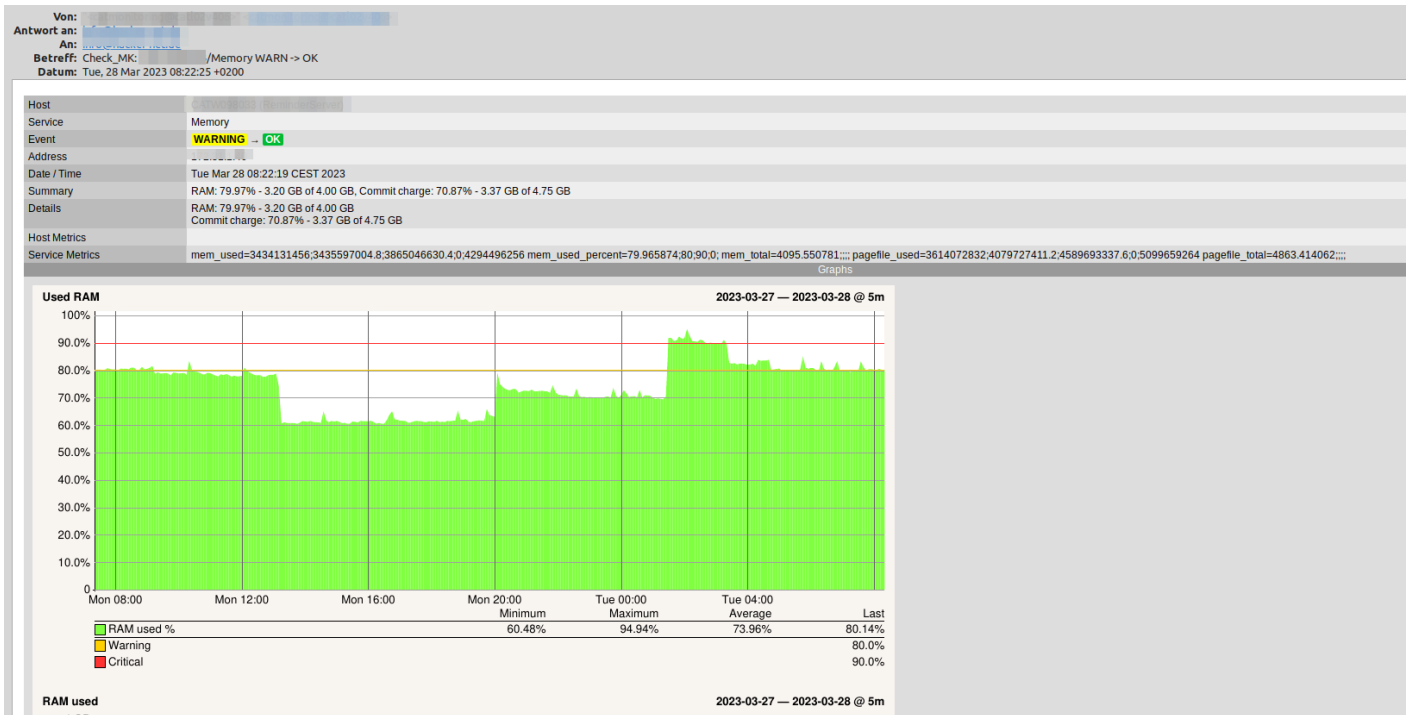
Es gibt Situationen da passen die Standard Schwellenwerte / Limits einfach nicht. Zum Beispiel weil ein Server zu wenig Festplattenspeicher für eine Temporäre Zeit hat oder der RAM übern Limit ist. Aber es eigentlich noch okay ist. Da kein weiterer RAM dazu gekauft wird.

Auszug aus den Emails.

Von OK auf Warning



Danach wieder von Warning auf OK



Einstellen der Limits

Auf dem Dashboard hat man unten rechts eine Schnellübersicht der Alarmer der letzten 7 Tage.

Top alerters (last 7 days)		
Problems	Host	Service
203	[redacted]33	Memory
4	[redacted]1	Interface 15
3	[redacted]1	Interface 12
1	[redacted]1	Check_MK
1	[redacted]1	Interface 07
1	[redacted]2	Interface 11
2	[redacted]1	Interface 13
2	[redacted]1	Interface 14

Klickt man auf den Hosteintrag bekommt, man alle service Alerts. Klickt man auf den Service bekommt man die Alerts nur vom Service für den Host. Wir klicken auf Memory und bekommen dann folgende Ansicht.

Hier sehen wir Limits von 79,95% - 80% bei RAM und Commit charge 70- 72% gerundet

Your query produced more than 1000 results. [Repeat query and allow more results.](#) Note: the shown results are incomplete and do not reflect the sort order.

Tuesday, 2023-03-28

Time	Event	State info	Summary
67 m	SERVICE NOTIFICATION RESULT	EXIT_CODE (SUCCESS)	Spooled mail to local mail transmission agent
67 m	SERVICE NOTIFICATION	NOTIFY (OK)	RAM: 79.95% - 3.20 GB of 4.00 GB, Commit charge: 70.76% - 3.36 GB of 4.75 GB
67 m	SERVICE NOTIFICATION	NOTIFY (OK)	RAM: 79.95% - 3.20 GB of 4.00 GB, Commit charge: 70.76% - 3.36 GB of 4.75 GB
67 m	SERVICE ALERT	HARD (OK)	RAM: 79.95% - 3.20 GB of 4.00 GB, Commit charge: 70.76% - 3.36 GB of 4.75 GB
67 m	SERVICE NOTIFICATION RESULT	EXIT_CODE (SUCCESS)	Spooled mail to local mail transmission agent
67 m	SERVICE NOTIFICATION	NOTIFY (WARNING)	RAM: 80.37% - 3.21 GB of 4.00 GB (warn/crit at 80.00%/90.00% used) WARN Commit charge: 71.16% - 3.38 GB of 4.75 GB
67 m	SERVICE NOTIFICATION	FLAPPINGSTOP (WARNING)	RAM: 80.37% - 3.21 GB of 4.00 GB (warn/crit at 80.00%/90.00% used) WARN Commit charge: 71.16% - 3.38 GB of 4.75 GB
67 m	SERVICE FLAPPING ALERT	STOPPED	Stopped flapping
72 m	SERVICE NOTIFICATION RESULT	EXIT_CODE (SUCCESS)	Spooled mail to local mail transmission agent
72 m	SERVICE NOTIFICATION	NOTIFY (WARNING)	RAM: 80.43% - 3.22 GB of 4.00 GB (warn/crit at 80.00%/90.00% used) WARN Commit charge: 71.29% - 3.39 GB of 4.75 GB
72 m	SERVICE NOTIFICATION	FLAPPINGSTART (WARNING)	RAM: 80.43% - 3.22 GB of 4.00 GB (warn/crit at 80.00%/90.00% used) WARN Commit charge: 71.29% - 3.39 GB of 4.75 GB
72 m	SERVICE FLAPPING ALERT	STARTED	Started flapping
72 m	SERVICE ALERT	HARD (WARNING)	RAM: 80.43% - 3.22 GB of 4.00 GB (warn/crit at 80.00%/90.00% used) WARN Commit charge: 71.29% - 3.39 GB of 4.75 GB
73 m	SERVICE NOTIFICATION RESULT	EXIT_CODE (SUCCESS)	Spooled mail to local mail transmission agent
73 m	SERVICE NOTIFICATION	NOTIFY (OK)	RAM: 79.97% - 3.20 GB of 4.00 GB, Commit charge: 70.87% - 3.37 GB of 4.75 GB
73 m	SERVICE NOTIFICATION	NOTIFY (OK)	RAM: 79.97% - 3.20 GB of 4.00 GB, Commit charge: 70.87% - 3.37 GB of 4.75 GB
73 m	SERVICE ALERT	HARD (OK)	RAM: 79.97% - 3.20 GB of 4.00 GB, Commit charge: 70.87% - 3.37 GB of 4.75 GB
74 m	SERVICE NOTIFICATION RESULT	EXIT_CODE (SUCCESS)	Spooled mail to local mail transmission agent
74 m	SERVICE NOTIFICATION	NOTIFY (WARNING)	RAM: 80.02% - 3.20 GB of 4.00 GB (warn/crit at 80.00%/90.00% used) WARN Commit charge: 70.83% - 3.36 GB of 4.75 GB
74 m	SERVICE NOTIFICATION	NOTIFY (WARNING)	RAM: 80.02% - 3.20 GB of 4.00 GB (warn/crit at 80.00%/90.00% used) WARN Commit charge: 70.83% - 3.36 GB of 4.75 GB

Nun klicken wir oben im Menü auf Service

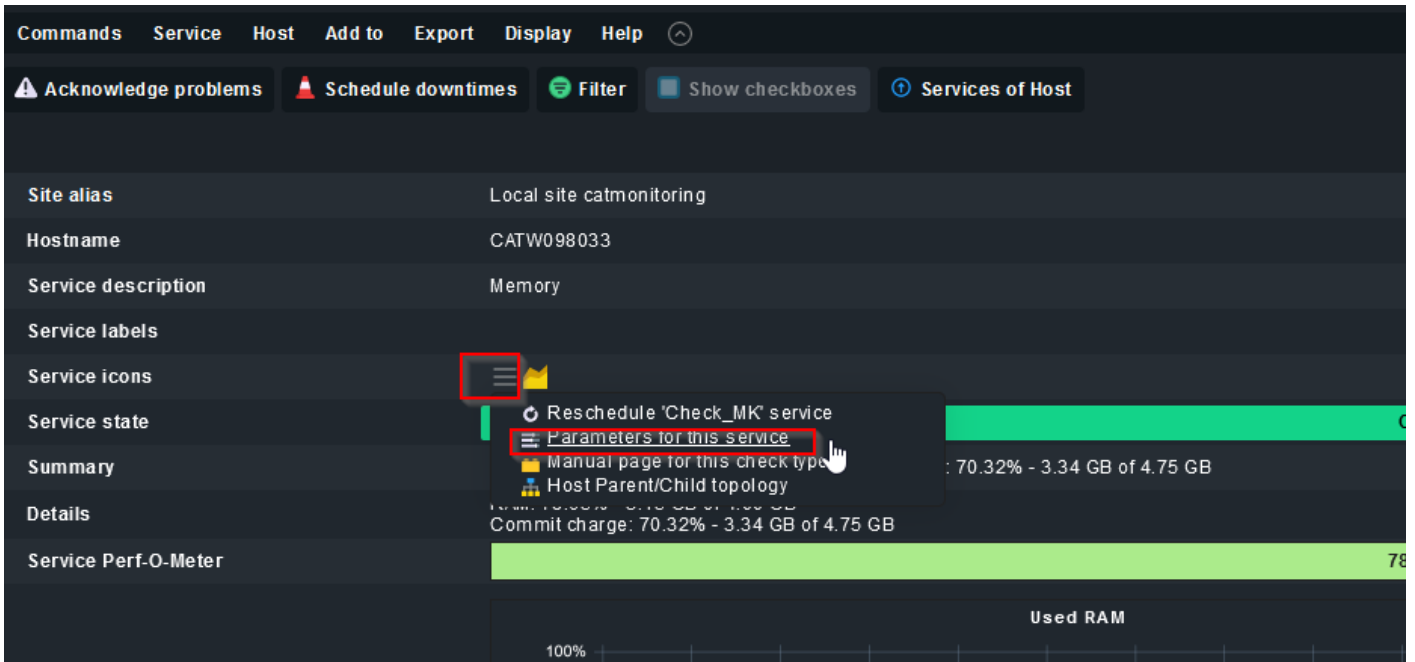
Filter Show checkboxes **Service**

Your query produced more than 1000 results. [Repeat query and allow more results.](#) Note: the shown results are incomplete and do not reflect the sort order.

Tuesday, 2023-03-28

Time	Event	State info	Summary
67 m	SERVICE NOTIFICATION RESULT	EXIT_CODE (SUCCESS)	Spooled mail to local mail transmission agent
67 m	SERVICE NOTIFICATION	NOTIFY (OK)	RAM: 79.95% - 3.20 GB of 4.00 GB, Commit charge: 70.76% - 3.36 GB of 4.75 GB
67 m	SERVICE NOTIFICATION	NOTIFY (OK)	RAM: 79.95% - 3.20 GB of 4.00 GB, Commit charge: 70.76% - 3.36 GB of 4.75 GB

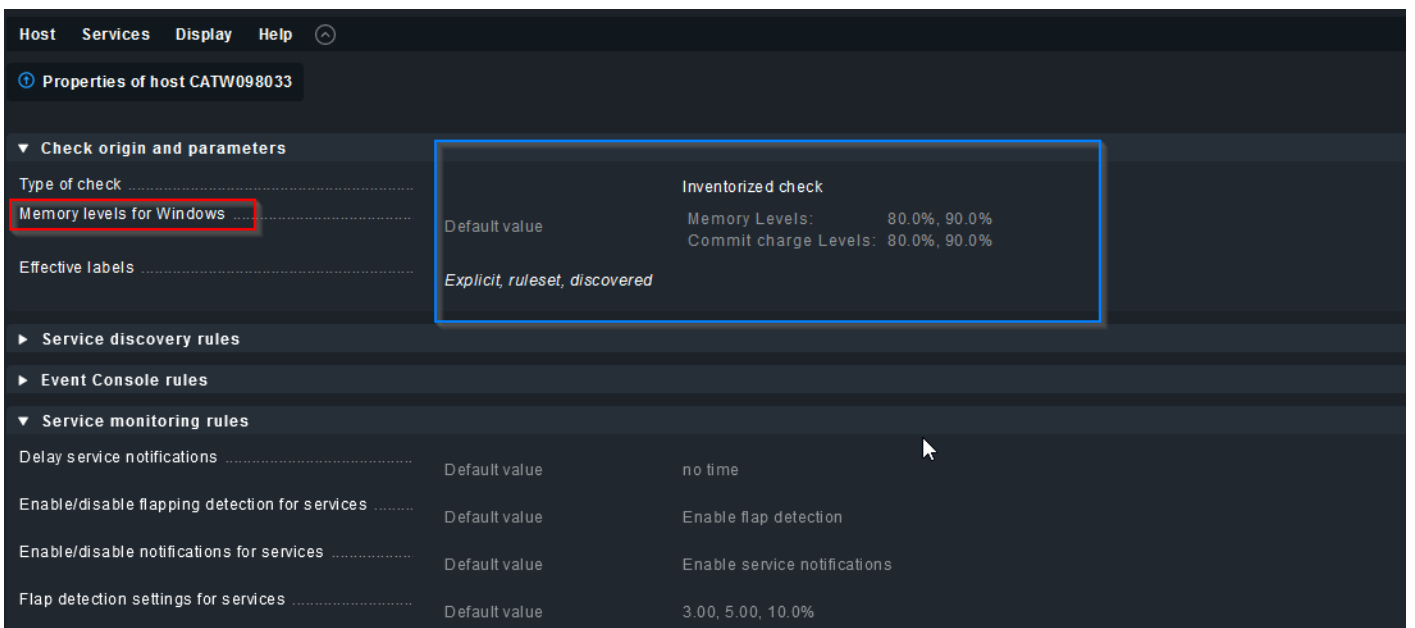
Nun auf Das Burgermenu bei Service Icons und dann auf Parameters for this service klicken



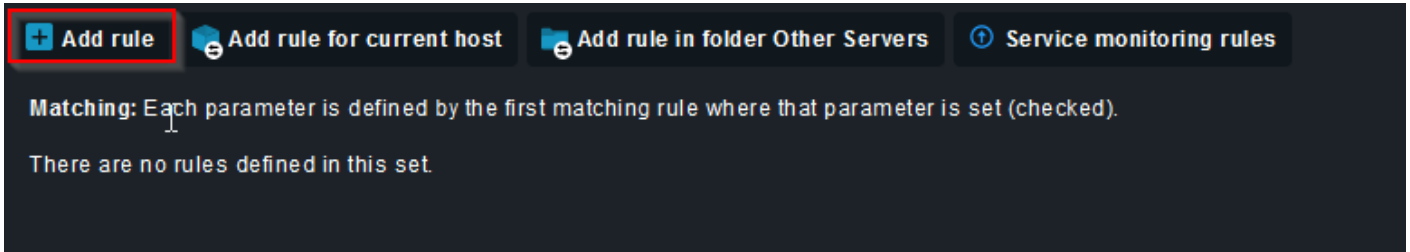
Nun haben wir eine Neue Übersicht.

Im Blauen Rahmen Rechts sieht man das dieser Check schon Standardwerte hat, nämlich Warnung bei 80% und Critical bei 90%.

Im roten Rahmen steht der Name des Checks. Diesen klicken wir an, um eine neue Regel zu erstellen.



Nun klicken wir auf Add Rule



Nun geben wir eine Beschreibung für die Rule ein.

unter Value können wir ankreuzen, welche Werte wir definieren wollen.

Wir wollen in diesem Beispiel den Memory und die Commit Change anpassen.

Unsere Werte waren gerundet, 79,95% - 80% bei RAM und Commit change 70- 72%

Also legen wir RAM auf 88% Warn und 95% Crit um ein bisschen Puffer zu haben.

Und commit belieben auf standard auf 80% Warn und 90% Crit.

Wir stellen fest, das Commit mit den Standardwerten ausreicht.

Also Kreuz wieder raus.

unten bei Explicit Hosts ein Kreuz rein und den Host auswählen.

Will man das ganze für eine Gruppe Wählt man die Gruppe aus. Anstatt Main, irgendeine andere.

Dazu muss der Haken dann bei explicit Host natürlich wieder raus.

Nun auf Save klicken

Rule Related Display Help

Save Abort Memory levels for Windows for host 33

▼ Rule Properties

Description Limits für Reminder Server

Comment

Documentation URL

Rule activation do not apply this rule

▼ Value

Memory Levels

Memory usage in percent

Warning at 88.0 %

Critical at 95.0 %

Commit charge Levels

Commit charge in percent (relative to commit limit)

Warning at 80.0 %

Critical at 90.0 %

Averaging

Enable timespecific parameters

▼ Conditions

Condition type Explicit conditions

Folder Main

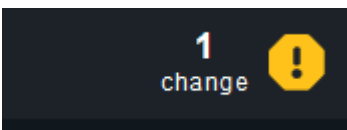
Host tags Add tag condition

Host labels Add label condition

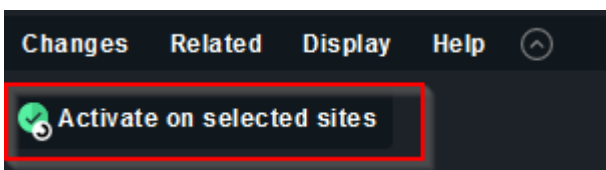
Explicit hosts 33 (Select hostname)

Negate: make rule apply for all but the above hosts

Nun auf Change klicken



Nun auf Activate on selected sites



Fertig.

Benachrichtigungen für bestimmte Checks deaktivieren

Beschreibung:

Wer kennt sie nicht die Windows Ereignisanzeige. Da werden Fehler und Warnungen geloggt, die eigentlich nicht von belang sind. Sind aber im Log als Critical und Checkmk schickt auch sofort ein Alert raus.

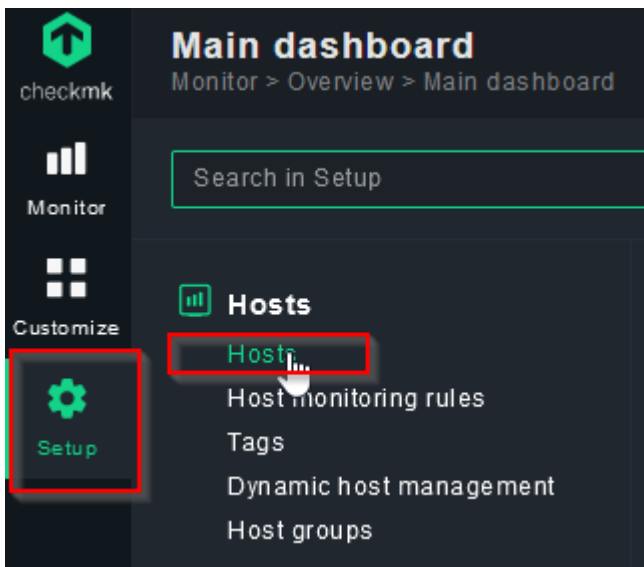
Man möchte die Alerts natürlich im Checkmk System behalten. Aber eben keine Emailbenachrichtigung.

Wie also rausnehmen?

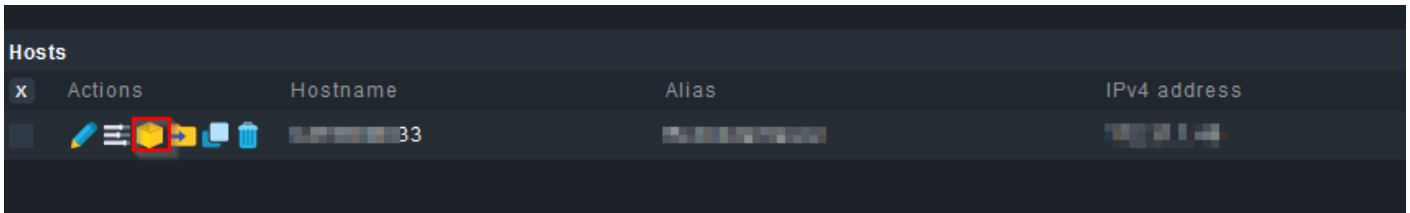
Emailbenachrichtigung für einen Service rausnehmen:

In der Checkmk Oberfläche anmelden.

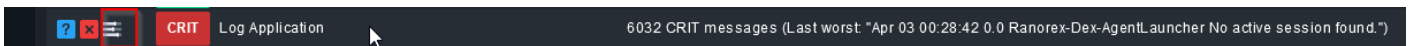
Dann unter Setup -> Hosts den Host raussuchen



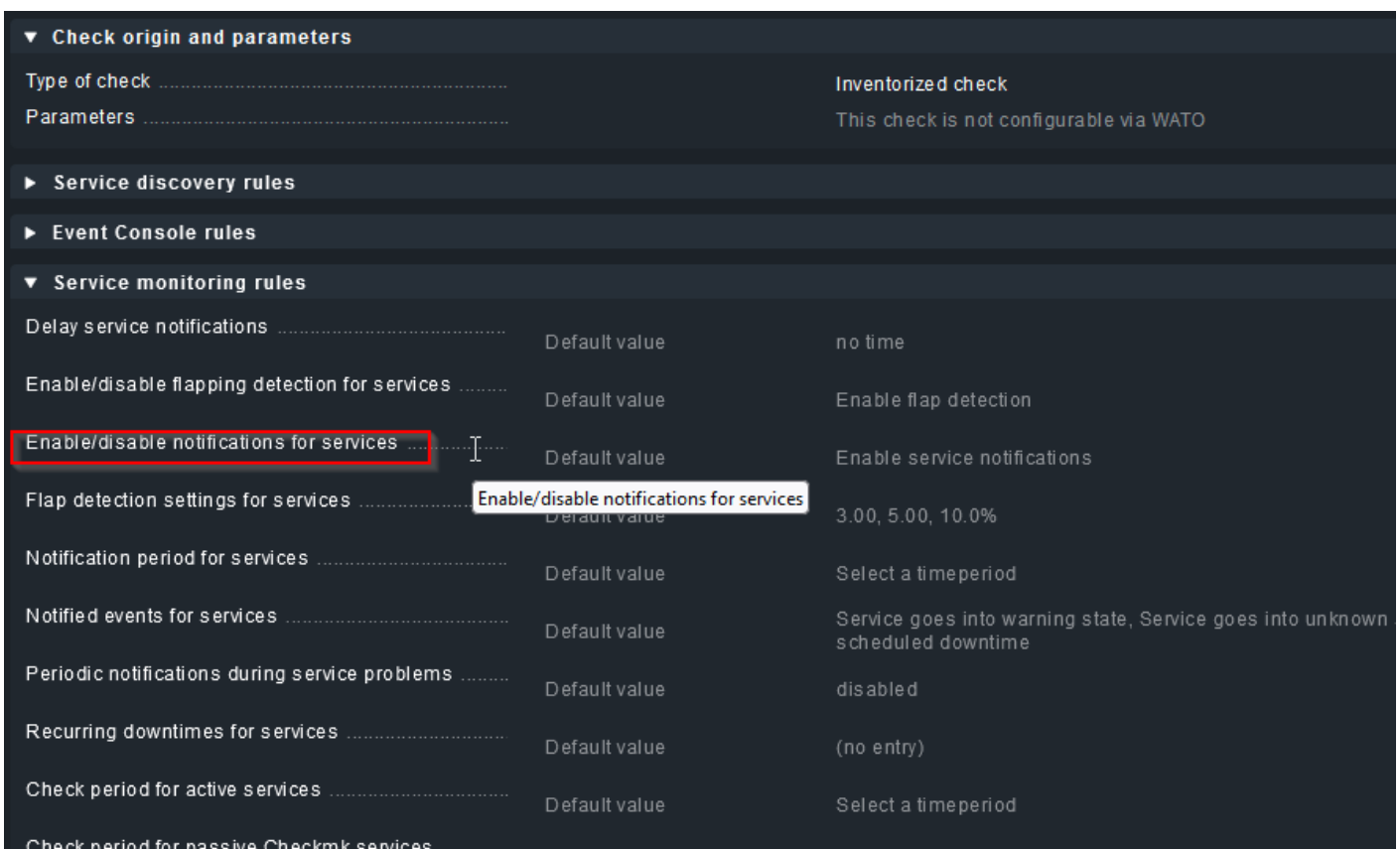
Dort dann auf Services klicken



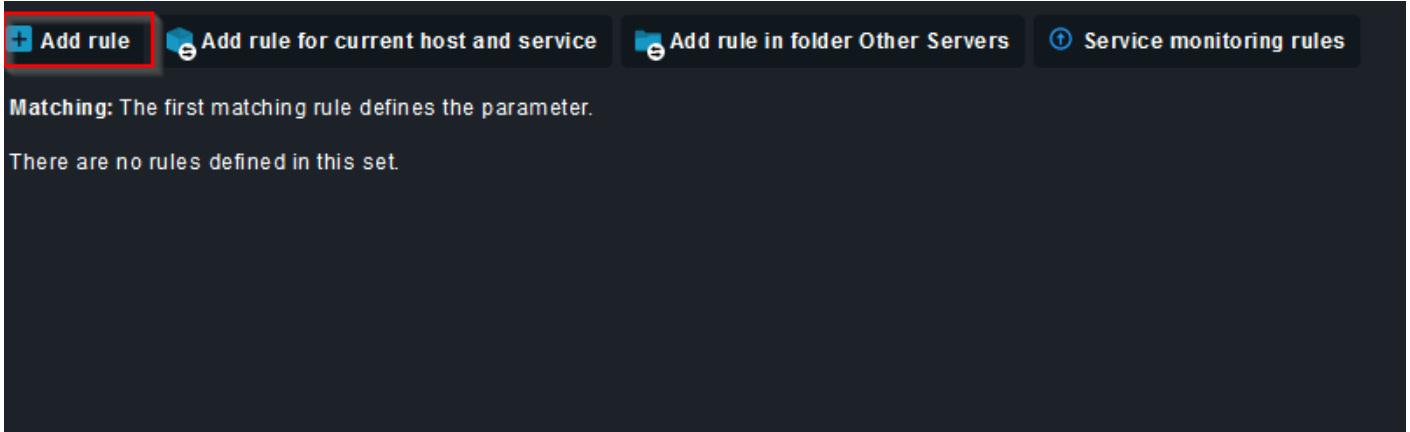
Bei den Service den man ändern möchte auf die Balken klicken
(man kann in diesem Atemzug auch mehrere Services in eine Rule packen. Dann einfach die
Anderen Servicennamen wie hier
Log Application, dann noch hinzufügen.) Sehen wir später im Rule Editor



Nun auf Enable Disable notification for services klicken



Nun Auf add Rule



Nun eine Beschreibung ausfüllen

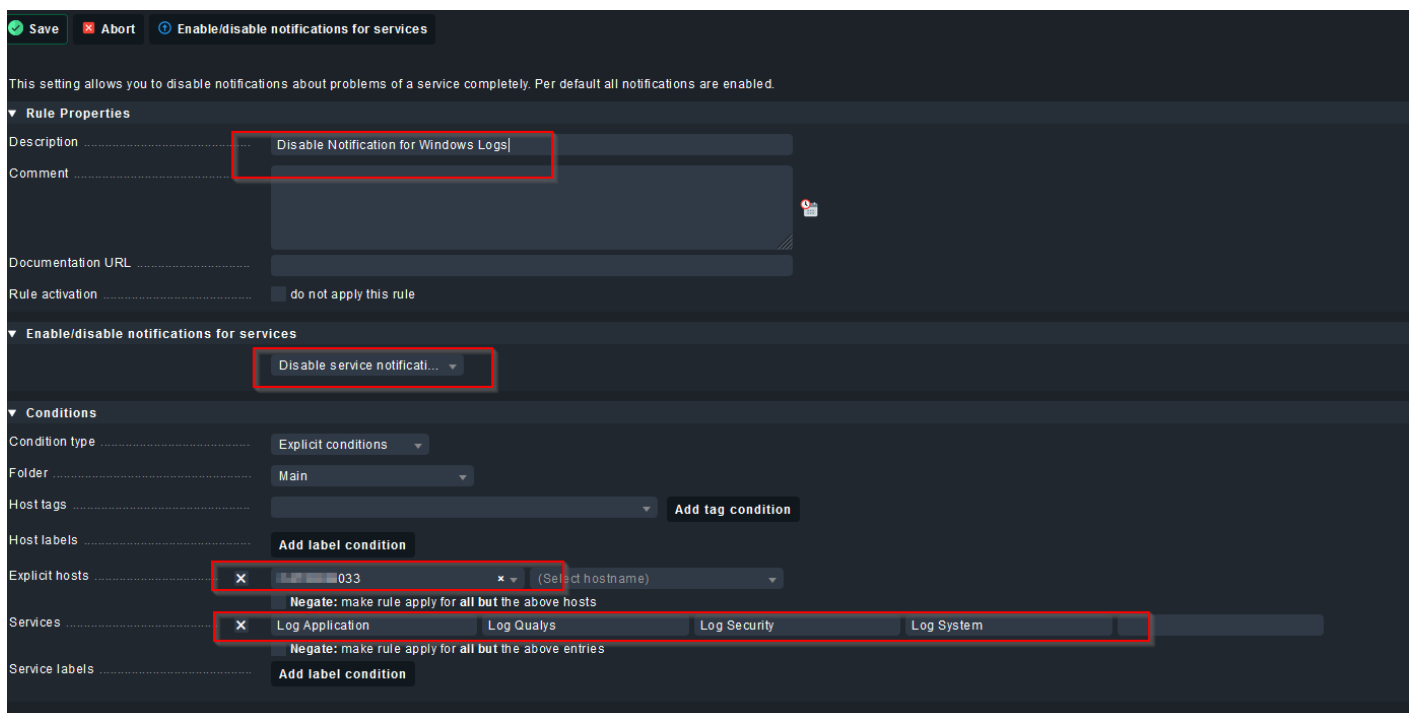
Enable Disable notification for Services auf disabled stellen, denn das wollen wir ja

Bei Condition wieder angeben für wen das gelten soll, Gruppe oder Host etc. In unserem beispiel ein Host.

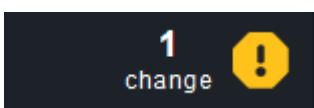
Hätten wir mehr Windows Server hätten wir das anhand eines Labels wie zum Beispiel Windows machen können.

nun die Services, wir können mehre definieren. Einfach den Service Namen eintragen.

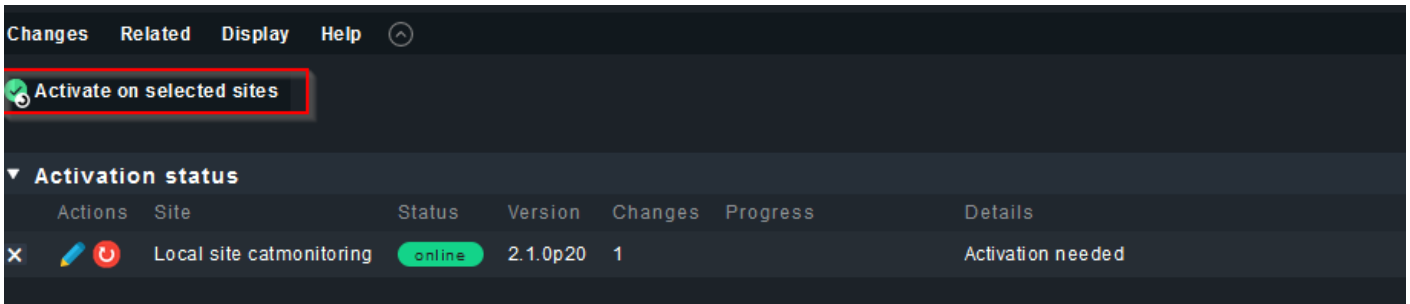
Nun aus Save oben links klicken.



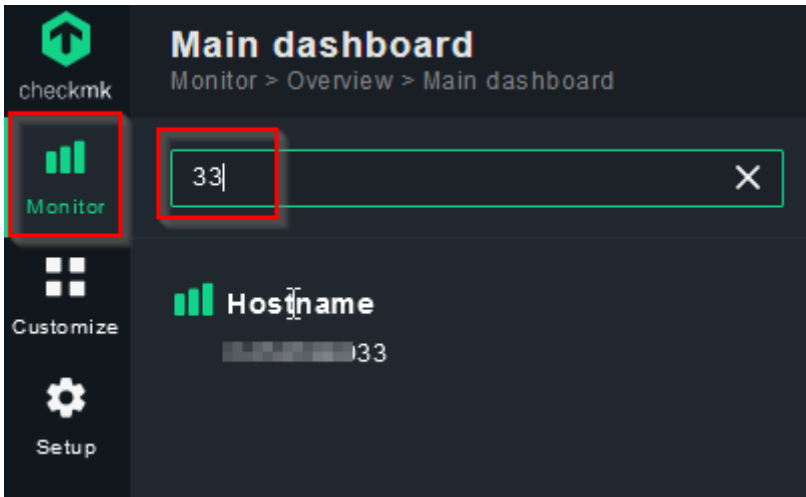
Nun auf change oben rechts klicken



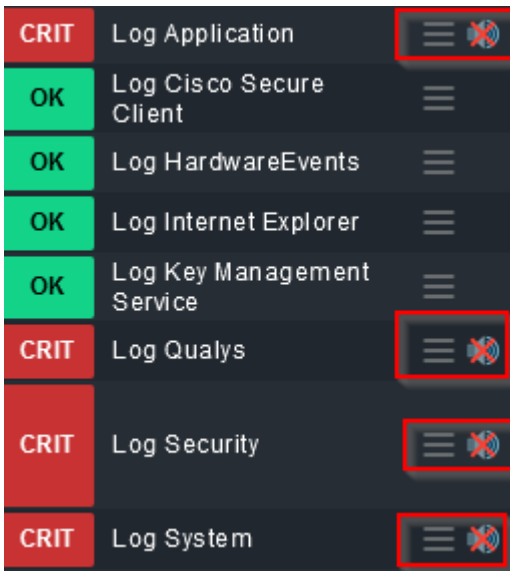
Nun auf Activate on selected sites



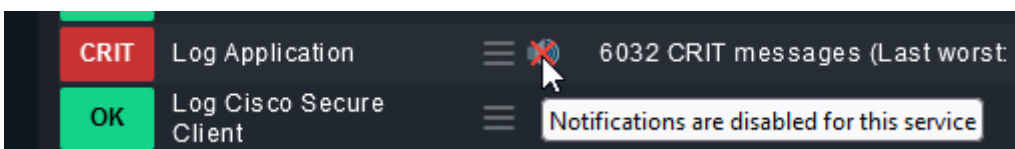
Fertig nun werden für die Dienste keine Notifcations mehr rausgesendet.
 Wenn man jetzt die Services wieder aufruft.
 Dazu einfach in der Suche den host eingeben und drauf klicken



Dann sieht man alle Services und bei denen wo wir die deaktiviert ahben, sieht man ein durchgestrichenes Lautsprechersymbol.



Geht man mit der Maus drüber, steht da Notiifications are disabled for this service



Fertig.

Wartungszeiten - geplante Downtimes festlegen

Beschreibung:

Es kommt immer mal wieder vor das ein Dienst offline genommen wird.

Oder sogar jeden Tag ein Gerät nicht mehr erreichbar ist, zum Beispiel über ein Backup was nachts läuft, oder

eine Zwangstrennung den internet Providers.

Hier kommen Wartungszeiten ins Spiel, die sogenannten "Scheduled Downtimes"

Hier gibt es zwei Kategorien.

Einmal Host und Service.

Es muss ja nur bei einem Update der SQL Datenbank zum Beispiel ja nicht der ganze Host weg sein.

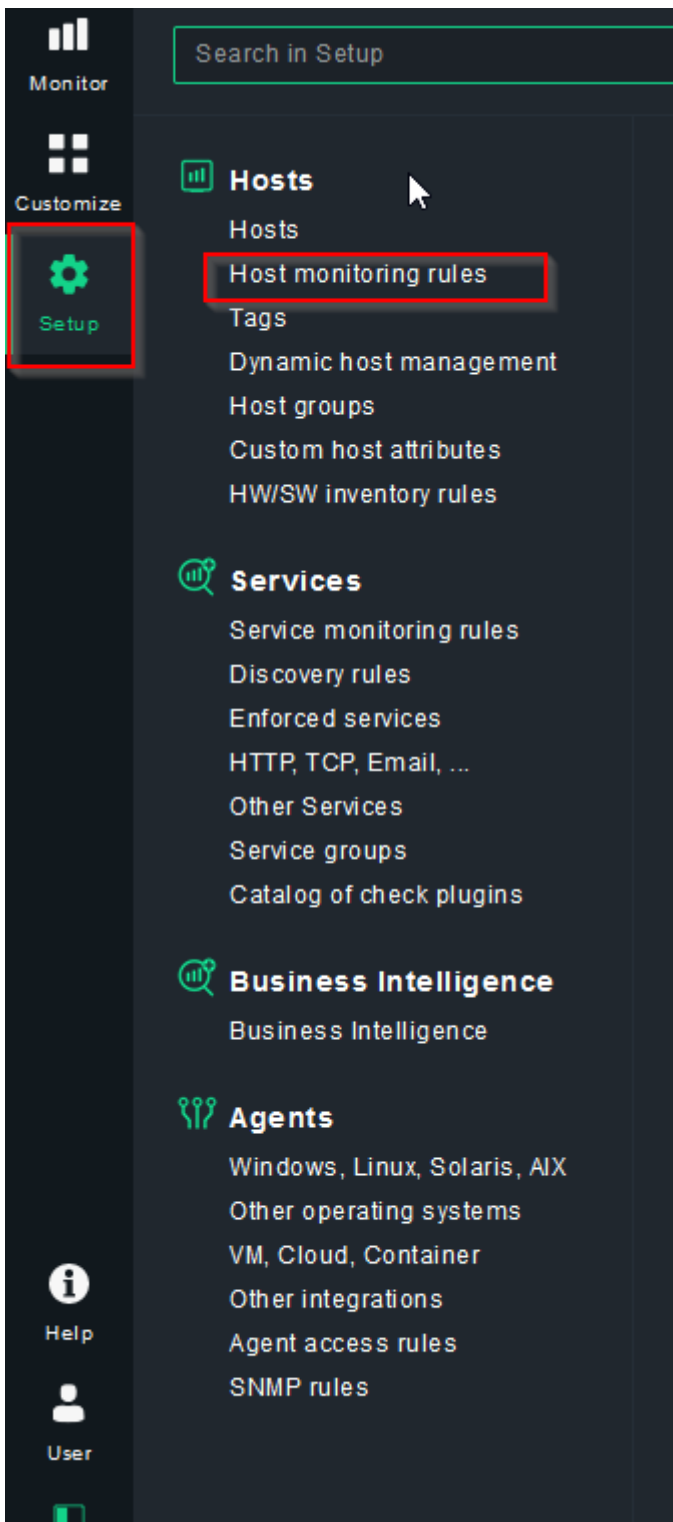
Dann würde man da für einen oder mehrere Services eine Downtime einrichten.

Einrichtung:

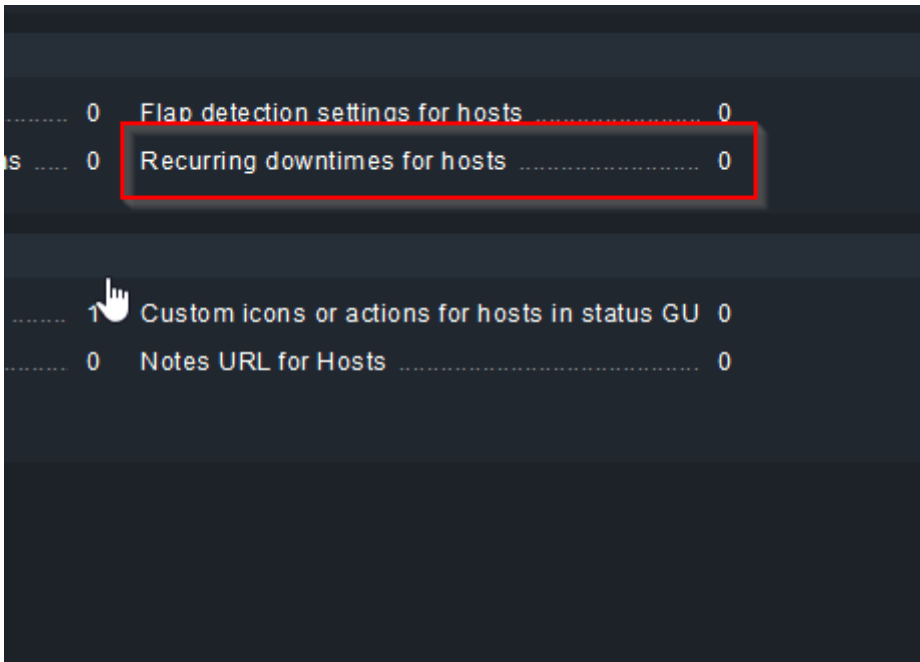
Downtime eines Host definieren

Dazu loggen wir uns in der Checkmk Oberfläche ein.

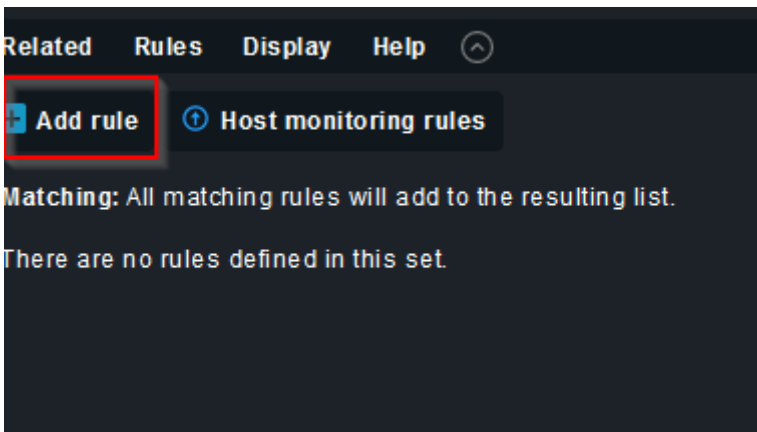
Dann auf Setup ->Host monitoring rules



Dann ganz rechts auf Recurring downtimes for Hosts



Dann auf Add rule klicken



Hier definieren wir ein beispiel das jeden tag um 23:50 - 00:05 eine Downtime stattfindet, wegen Backups und so die Maschine nicht verfügbar ist.

Nun eine beschreibung vergeben für de Rule vergeben

Nun eine beschreibung vergeben, die die Downtime beschreibt

Wann die Downtime zum ersten mal stattfindet, hier 2023.05.04 um 23:50:00 UHR

Will man das die Downtime nur einmal ist, würde man den Haken downtime will not start anymore after einhaken. Was wir aber nicht wollen denn sie soll ja täglich ablaufen.

Dazu dann in repaet this downtime im dropdown menu das intervall auswählen, in diesem Fall day für täglich.

Nun angeben, wie lang die Downtime sein soll. Hier 0 Stunden und 15 min.

Damit beginnt unsere Downtime 23:50 - 0:05 UHR

Jetzt wieder wie bei allen Regeln auswählen ob es Pro Host oder Gruppe oder sonst irgendwelchen Kriterien gelten soll.

New rule: Recurring downtimes for hosts

Setup > Hosts > Host monitoring rules > Recurring downtimes for hosts > New rule: Recurring downtimes for hosts

Rule Related Display Help ↻

Save Abort Recurring downtimes for hosts

This rule set allows you to create downtimes for hosts that are repeated on a regular base.

▼ Rule Properties

Description

Comment

Documentation URL

Rule activation do not apply this rule

▼ Recurring downtimes for hosts

Comment for the downtime (required)

First time the downtime starts (required)

Year	Month	Day	Hour	Minute	Sec.
2023	5	4	23	50	0

Downtime will not start anymore after

Repeat this downtime

Duration
 hours mins

Flexible downtime

▼ Conditions

Condition type

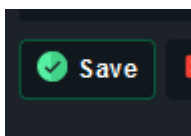
Folder

Host tags

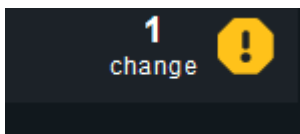
Host labels

Explicit hosts

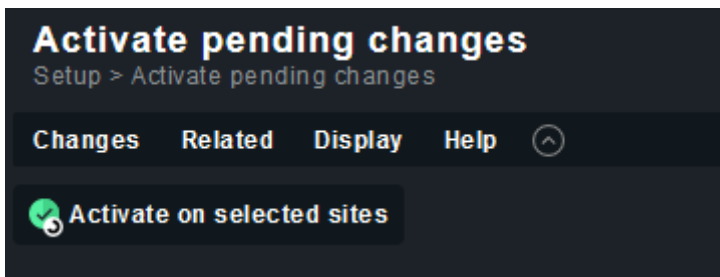
Nun auf Save klicken



Nun auf change klicken



Und auf activate klicken



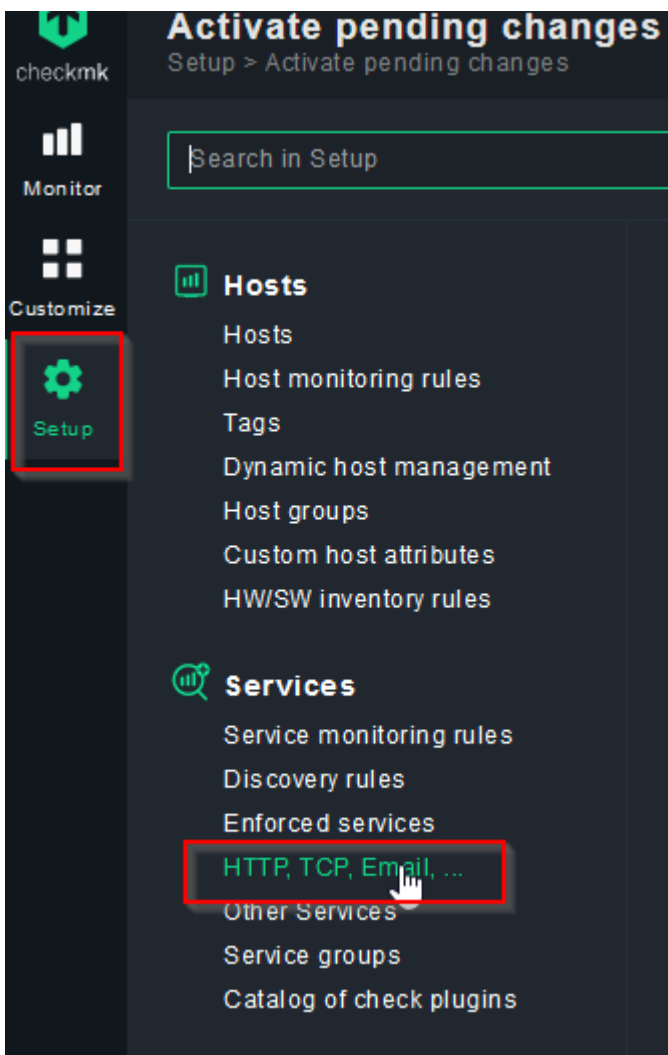
Downtime eines Service definieren

Den Namen des Service herausbekommen.

Damit wir die Regel anhand des Servicenamens definieren können brauchen wir zuerst den Namen der Regel.

In Unserem Beispiel geht es um einen HTTP Check.

Nun Auf Setup -> HTTP, TCP Email Checks klicken



Nun auf Check HTTP Service klicken

▼ HTTP, TCP, Email, ...			
Check DNS service	0	Check Email	0
Check HTML Form Submit	0	Check HTTP service	7
Check SFTP Service	0	Check SMB share access	0
Check TCP port connection	0	Check current routing	0

Nun sehen wir eine Liste mit den Checks und rechts die Service namen stehen.

Manchmal packt Checkmk noch was vor den den namen. Bei hHTTP Check ist es das Wort HTTP oder HTTPS.

Je nachdem ob der Check HTTP oder HTTPS checkt. Darum setzten wir ein .* davor und am ende ein \$

Denn dann ist egal ob der check später mal in HTTP oder HTTPS geändert wird. Weil der Name bleibt ja gleich.

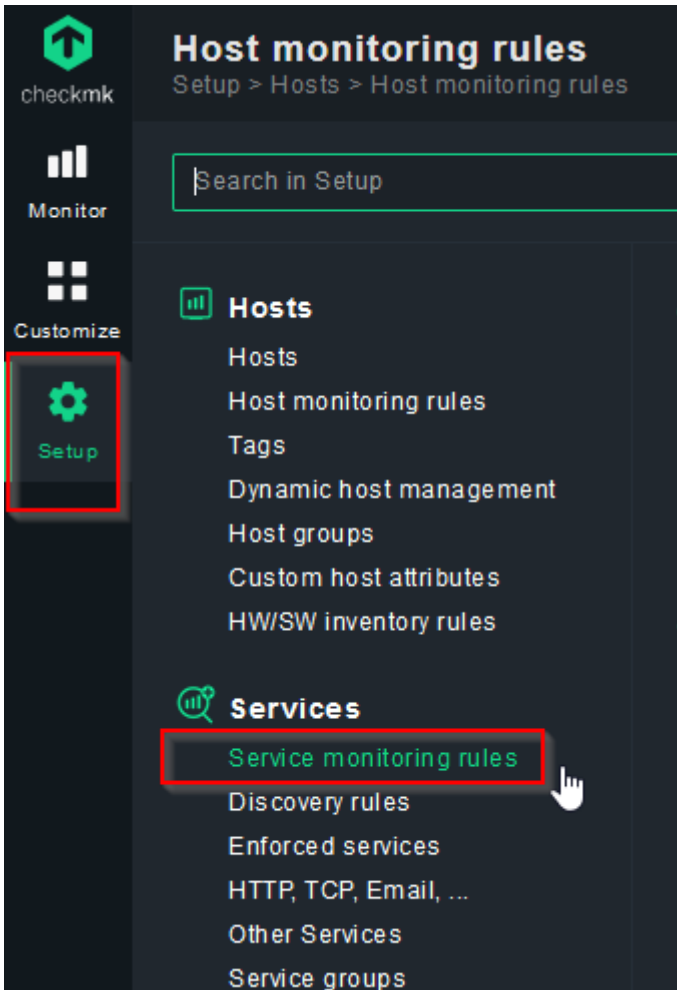
The screenshot shows the configuration for a service named 'check-nvs-prod-https-available'. The configuration details are as follows:

- Service name:** check-nvs-prod-https-available
- Host settings:**
 - Hostname / IP address: [redacted]
 - TCP Port: 443
- Check:**
 - Check the URL, URI to fetch (default is /): /manage/access/login/
 - Check Portal available and response time
 - Mode of the connection: Use SSL/HTTPS for the connection
 - Use SSL with auto negotiation
 - Expected response time: 100.00 ms, 120.00 ms
 - Fixed string to expect in the content: Please enter email address and password to login

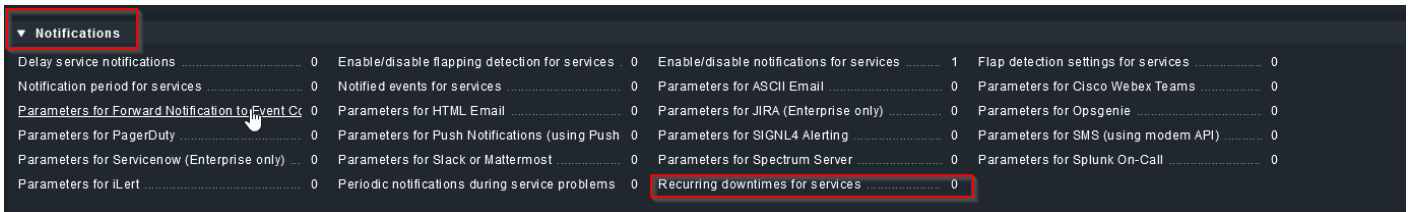
Jetzt haben wir den Namen und können die Downtime regel angelegen

Downtime anlegen

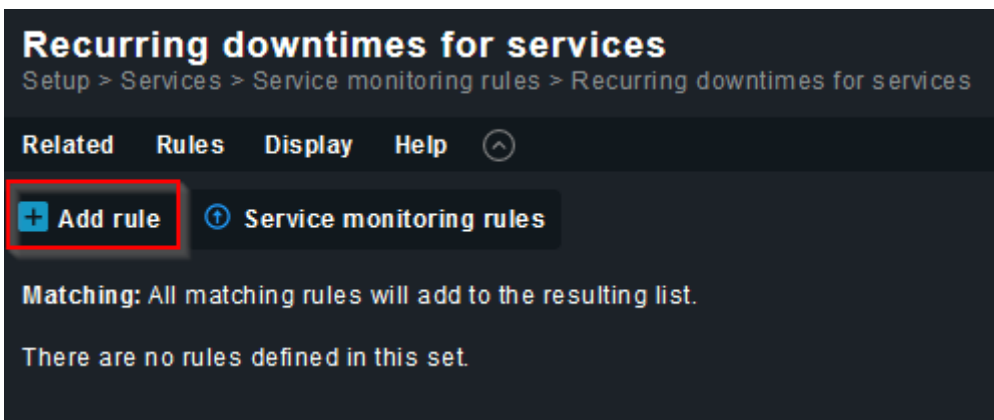
Auf Setup-> Services -> Service monitoring rules klicken



Dann runterscrollen bis Notifications und dann Recurring downtimes for Services anklicken



Dann auf Add rule klicken



Hier definieren wir ein beispiel das jeden tag um 23:50 - 00:05 eine Downtime stattfindet, wegen Backups und so die Maschine nicht verfügbar ist.

Nun eine beschreibung vergeben für de Rule vergeben

Nun eine beschreibung vergeben, die die Downtime beschreibt

Wann die Downtime zum ersten mal stattfindet, hier 2023.05.04 um 23:50:00 UHR

Will man das die Downtime nur einmal ist, würde man den Haken downtime will not start anymore after einhaken. Was wir aber nicht wollen denn sie soll ja täglich ablaufen.

Dazu dann in repeat this downtime im dropdown menu das intervall auswählen, in diesem Fall day für täglich.

Nun angeben, wie lang die Downtime sein soll. Hier 0 Stunden und 15 min.

Damit beginnt unsere Downtime 23:50 - 0:05 UHR

Jetzt wieder wie bei allen Regeln auswählen ob es Pro Host oder Gruppe oder sonst irgendwelchen Kretierien gelten soll.

hier wählen wir service aus und geben den Service Namen an.

Beim Service wird immer der Anfang genommen und am Ende muss ein \$ Zeichen sein, damit das System weiß hier ist Ende.

In unserem Beispiel : `.*check-rvs-prod-https-available$`

This rule set allows you to create downtimes for services that are repeated on a regular base.

▼ Rule Properties

Description Downtime HTTPS Check rvs During backup

Comment

Documentation URL

Rule activation do not apply this rule

▼ Recurring downtimes for services

Comment for the downtime (required)
Downtime HTTPS Check rvs During backup

First time the downtime starts (required)

Year	Month	Day	Hour	Minute	Sec.
2023	5	4	23	50	0

Downtime will not start anymore after

Repeat this downtime
day

Duration
0 hours 15 mins

Flexible downtime

Nun die Condition einstellen.

▼ Conditions

Condition type Explicit conditions ▼

Folder Monitoring-Server ▼

Host tags Add tag condition

Host labels Add label condition

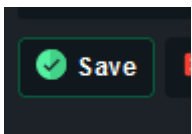
Explicit hosts

Services x *check-rvs-prod-https-available! |

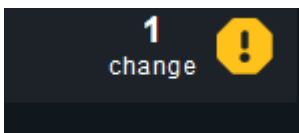
Service labels Add label condition

Negate: make rule apply for all but the above entries

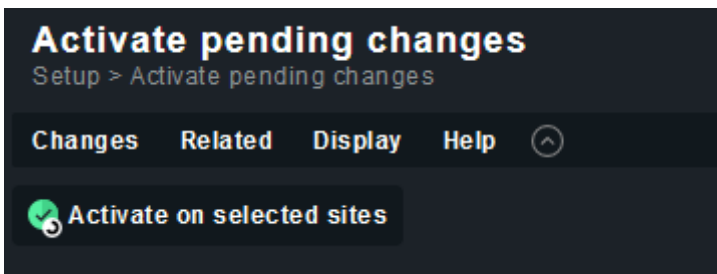
Dann auf Save klicken



Dann auf change klicken.



Nun activate on selected site anklicken.



Fertig

Discovery anpassen -> Netzwerkkarten ausschliessen

Beschreibung:

Manchmal möchte man Geräte nicht erfassen, so wie die tap Adapter von VPNs oder Virtuellen Maschinen.

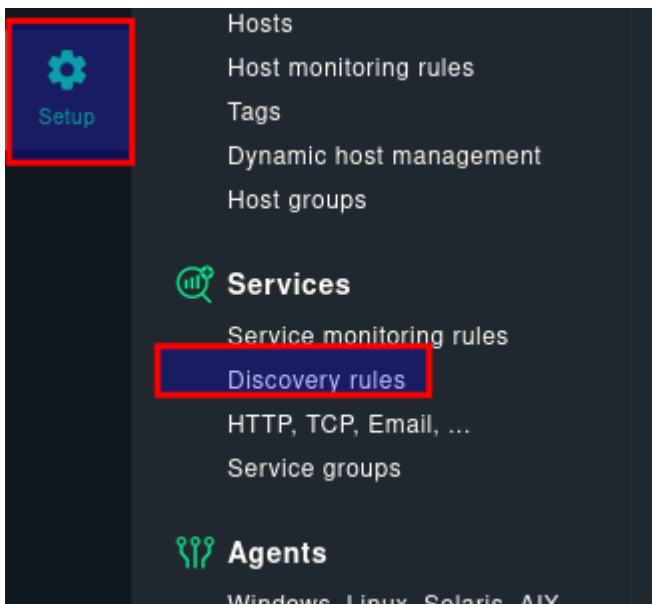
Dazu gibt es eine Möglichkeit Discovery Rules anzulegen.

In dem gleichen Menü kann man noch andere Discovery eigenschaften festlegen.

Hier werden wir die Netzwerkkarten festlegen.

Neue Rule erstellen

Dazu auf Setup -> Discovery Rules gehen.



Dort in unserem Fall auf Network interface and switch port discovery klicken

Related	Display	Help	Filter
Discovery of individual services			
Alertmanager discovery	0	Brocade port discovery	0
Discovery of Datadog monitors	0	CPU multi item discovery	0
Fujitsu storage CA port discovery	0	Disk IO discovery	0
MSSQL backup discovery	0	Filesystem discovery	1
NetApp Qtree discovery	0	Heartbeat CRM discovery	0
NetApp power supply unit discovery	0	IPMI sensor discovery	0
Process discovery (only SNMP)	0	MSSQL counter discovery	0
SAP value discovery	0	MSSQL datafile and transactionlog discovery	0
Windows DHCP pool discovery	0	NetApp fan discovery	0
		NetApp snapvault discovery	0
		Netapp port discovery	0
		Oracle performance discovery	0
		Process discovery	0
		Rittal CMC III discovery	0
		SAP R/3 single value discovery	0
		Solaris service discovery	0
		Systemd single services discovery	0
		VNX quotas and filesystems discovery	0
		eWON discovery	0
Discovery and Checkmk settings			
Disabled checks	0	Disabled services	5
		Periodic service discovery	1

Add Rule anklicken

Network interface and switch port discovery
 Setup > Services > Service discovery rules > Network interface and switch port discovery

Related Rules Display Help ⌵

+ Add rule ⌵ Service discovery rules

Nun folgendes ausfüllen:

Description: was die rule machen soll

Appearance of network interface : Use alias

Conditions for this rule to apply : specify matching condition

Match interface alias (regex) : anhaken

und dort alle interfaces rein die gefunden werden sollen. Im regex format ist .* ein asterisk

The screenshot shows a configuration window for a rule named "Network interface and switch port discovery". The interface is dark-themed and includes a menu bar (Rule, Related, Display, Help) and buttons for Save, Abort, and the rule name. The "Rule Properties" section contains fields for Description (filled with "no tap adapter and so on"), Comment, Documentation URL, and Rule activation (set to "do not apply this rule"). The main configuration area is titled "Network interface and switch port discovery" and includes several sections: "Configure discovery of single interfaces" (checked), "Discover single interfaces" (dropdown), "Appearance of network interface" (set to "Use alias"), "Port numbers" (dropdown), "Generate service labels for discovered interfaces" (checkbox), "Configure grouping of interfaces" (checkbox), "Conditions for this rule to apply" (dropdown), "Match port types" (checkbox), "Match port states" (checkbox), "Match admin states (SNMP with 64-bit counters only)" (checkbox), "Match interface index (regex)" (checkbox), "Match interface alias (regex)" (checked) with input fields for "eno.*", "ibs6", and "enp.*", and "Match interface description (regex)" (checkbox). The "Conditions" section at the bottom shows "Condition type" set to "Explicit conditions" and "Folder" set to "vserver".

Diese dann speichern und übernehmen.

Diese Rule wird dann in Zukunft angewandt.

Sollten die Devices schon erkannt worden sein, die man nicht haben möchte müssen diese trotzdem noch manuell entfernt werden. Diese Rule gilt für Zukünftige Devices.

Fertig