Week 4

Session 2

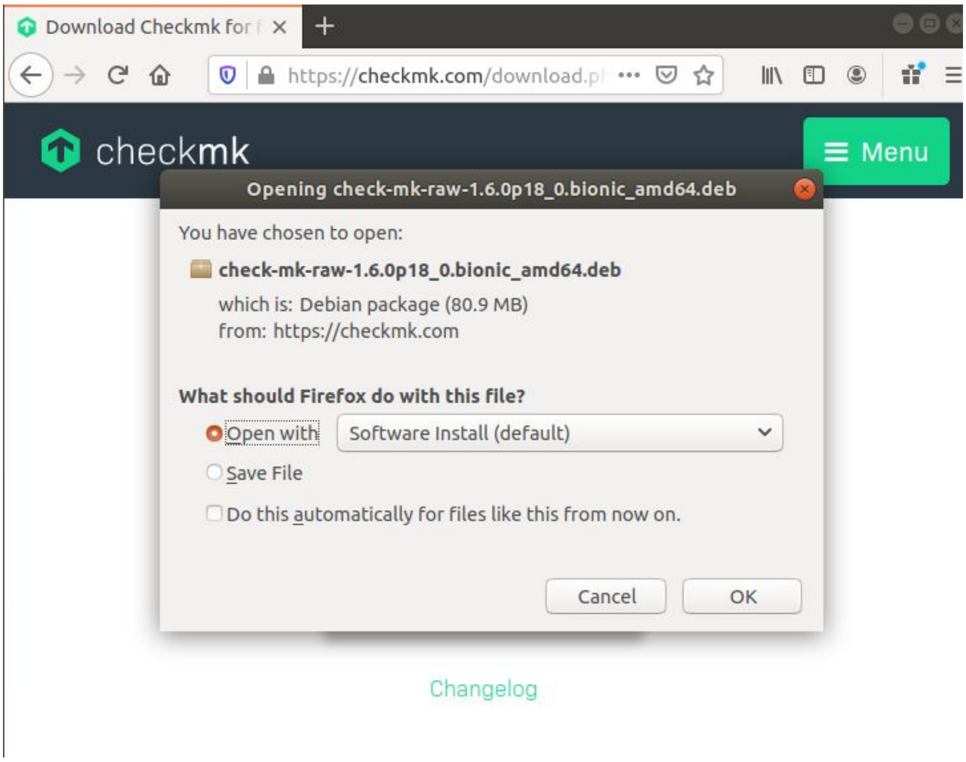
Monitoring

Olga Chernukhina

Checkmk

Install checkmk on one of your clients.

On the client machine go into checkmk website and download free version with the appropriate parameters (for me - Ubuntu 18.04, stable version) and then install it with the default software.



Opening check-mk-raw-1.6.0p18_0.bionic_amd64.deb

You have chosen to open:

check-mk-raw-1.6.0p18_0.bionic_amd64.deb

which is: Debian package (80.9 MB)
from: https://checkmk.com

What should Firefox do with this file?

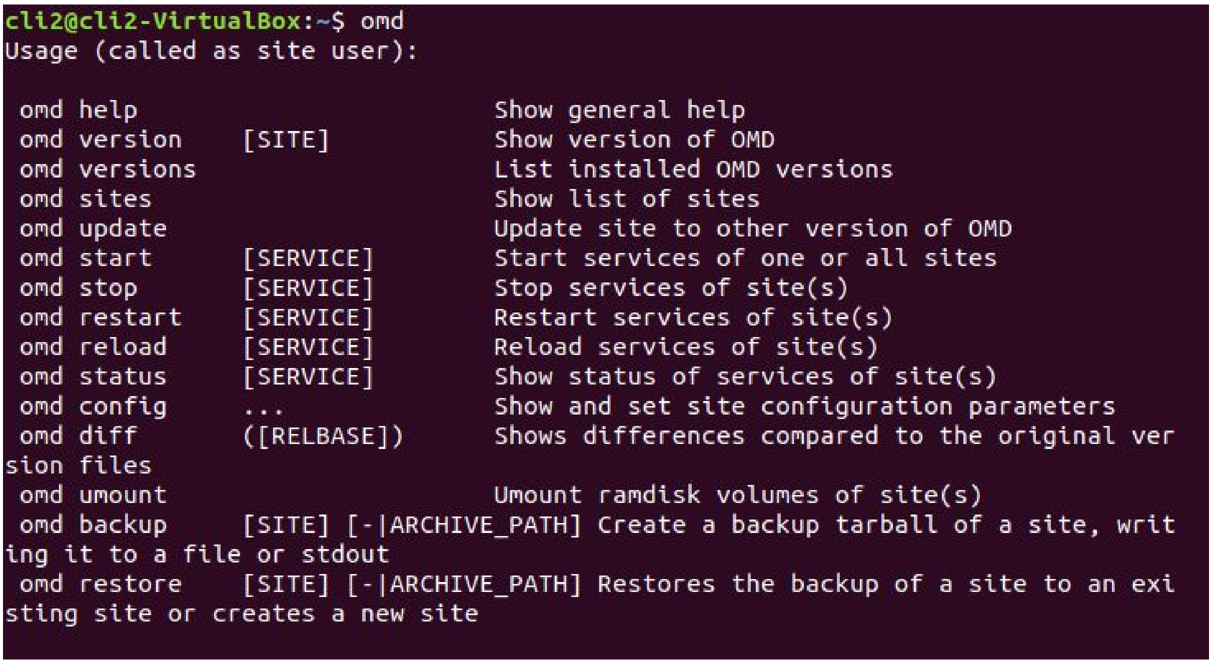
● Open with: Software Install (default)

○ Save File

□ Do this automatically for files like this from now on.

Cancel OK

Check the installation completed well by running `omd` command:



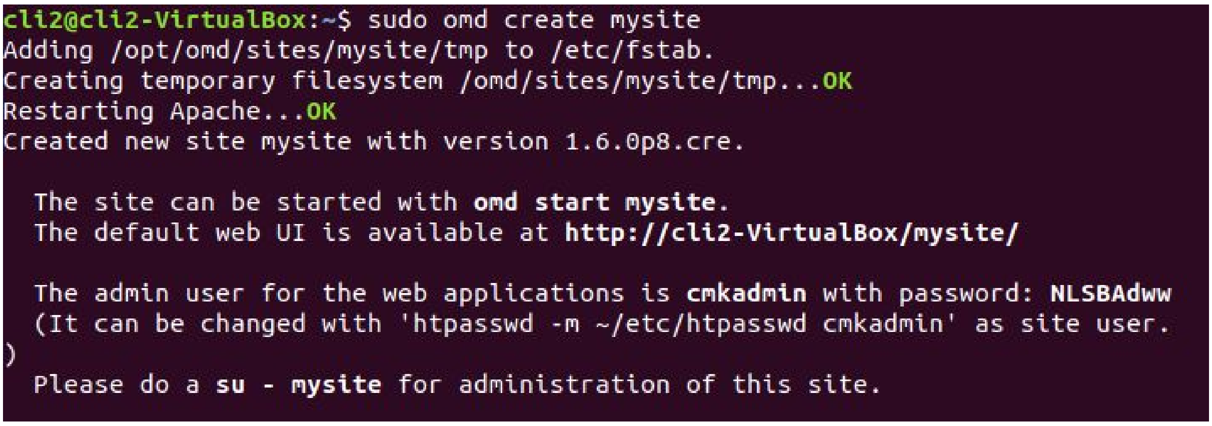
```
cli2@cli2-VirtualBox:~$ omd
Usage (called as site user):

 omd help                            Show general help
 omd version    [SITE]               Show version of OMD
 omd versions                        List installed OMD versions
 omd sites                           Show list of sites
 omd update                          Update site to other version of OMD
 omd start      [SERVICE]            Start services of one or all sites
 omd stop       [SERVICE]            Stop services of site(s)
 omd restart    [SERVICE]            Restart services of site(s)
 omd reload     [SERVICE]            Reload services of site(s)
 omd status     [SERVICE]            Show status of services of site(s)
 omd config     ...                  Show and set site configuration parameters
 omd diff       ([RELBASE])          Shows differences compared to the original ver
sion files
 omd umount                          Umount ramdisk volumes of site(s)
 omd backup     [SITE] [-|ARCHIVE_PATH] Create a backup tarball of a site, writ
ing it to a file or stdout
 omd restore    [SITE] [-|ARCHIVE_PATH] Restores the backup of a site to an exi
sting site or creates a new site
```

Create two sites

Sites:

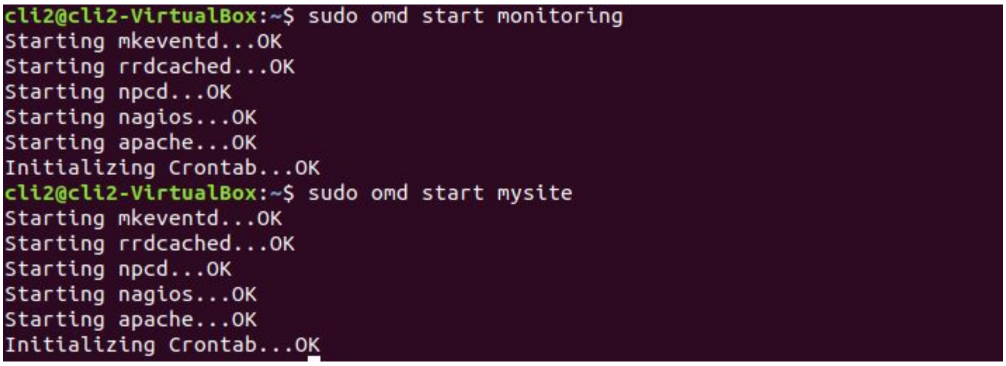
1) monitoring
2) mysite

For each sitename run `sudo omd create sitename` to create a monitoring instance and remember the login and password so that later you will be able to login to the monitoring website via the specified link



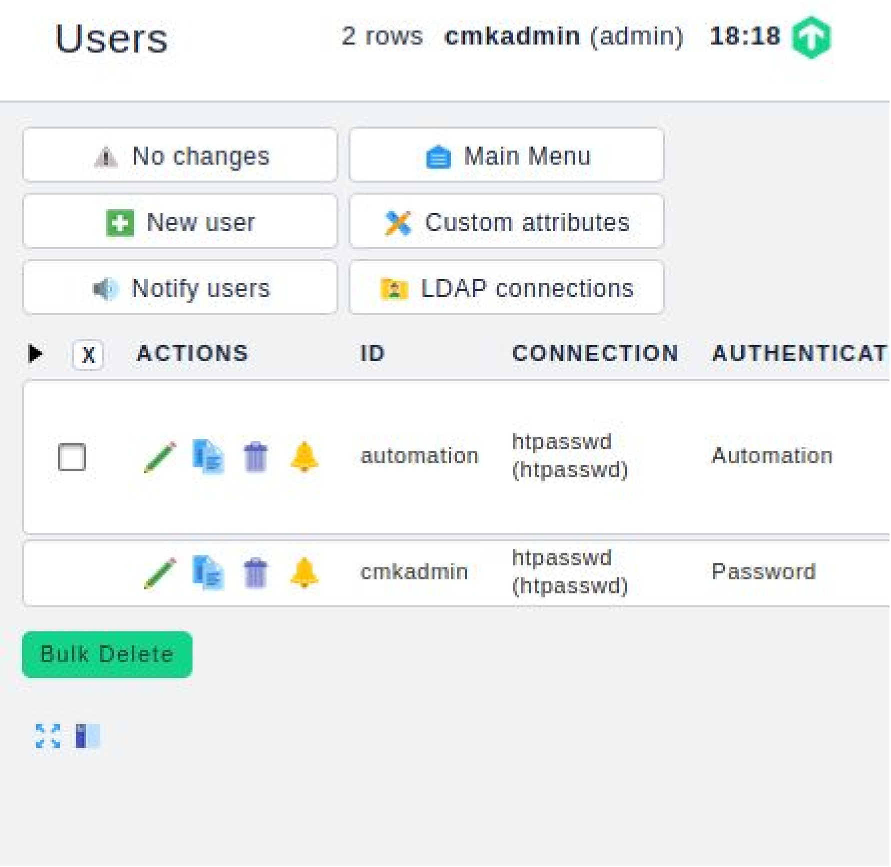
```
cli2@cli2-VirtualBox:~$ sudo omd create mysite
Adding /opt/omd/sites/mysite/tmp to /etc/fstab.
Creating temporary filesystem /omd/sites/mysite/tmp...OK
Restarting Apache...OK
Created new site mysite with version 1.6.0p8.cre.

  The site can be started with omd start mysite.
  The default web UI is available at http://cli2-VirtualBox/mysite/

  The admin user for the web applications is cmkadmin with password: NLSBAdww
  (It can be changed with 'htpasswd -m ~/etc/htpasswd cmkadmin' as site user.
)
  Please do a su - mysite for administration of this site.
```

Start each site:



```
cli2@cli2-VirtualBox:~$ sudo omd start monitoring
Starting mkeventd...OK
Starting rrdcached...OK
Starting npcd...OK
Starting nagios...OK
Starting apache...OK
Initializing Crontab...OK
cli2@cli2-VirtualBox:~$ sudo omd start mysite
Starting mkeventd...OK
Starting rrdcached...OK
Starting npcd...OK
Starting nagios...OK
Starting apache...OK
Initializing Crontab...OK
```

Later in the WATO - Configuration - Users tab you should change the cmkadmin's password for the purpose of security:

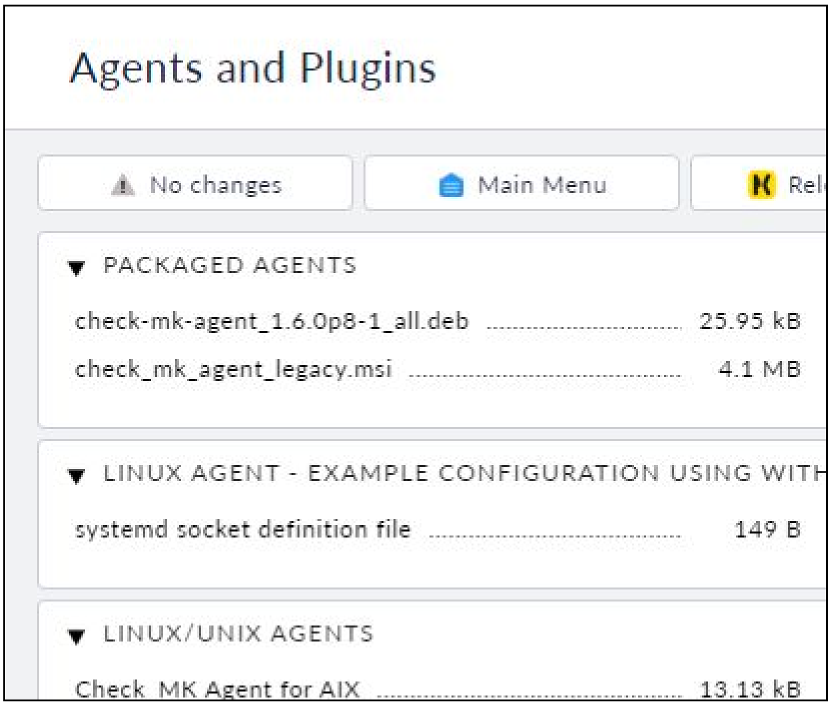


Users 2 rows **cmkadmin** (admin) **18:18**

X	ACTIONS	ID	CONNECTION	AUTHENTICAT
□		automation	htpasswd (htpasswd)	Automation
		cmkadmin	htpasswd (htpasswd)	Password

Add one server and one client to one site

First, open WATO - Configuration - Monitoring Agents and download the one that is suitable for Debian



Agents and Plugins

▼ PACKAGED AGENTS

check-mk-agent_1.6.0p8-1_all.deb	25.95 kB
check_mk_agent_legacy.msi	4.1 MB

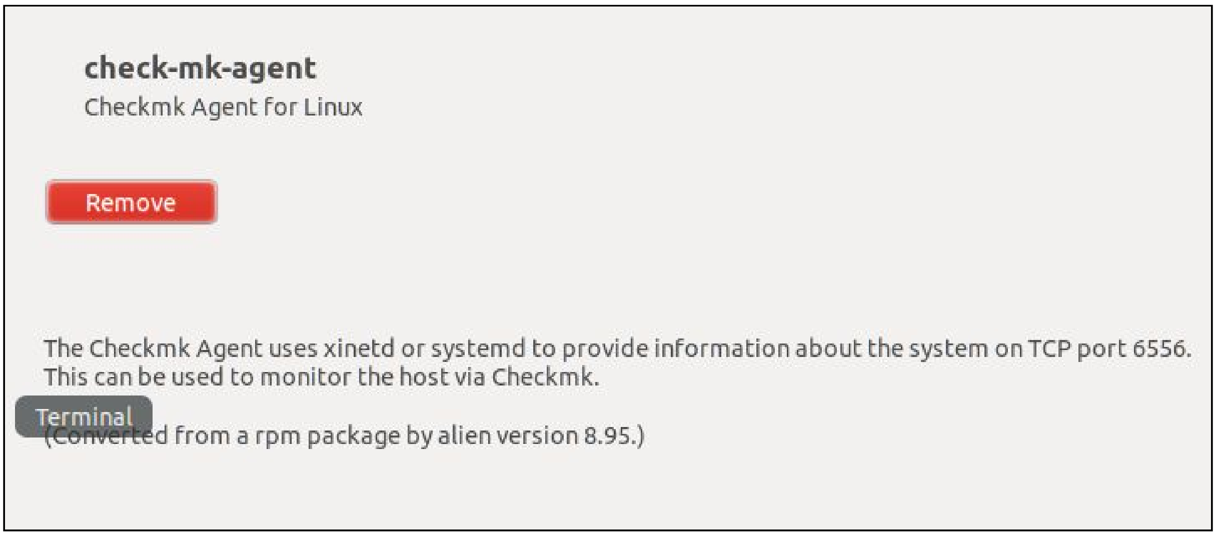
▼ LINUX AGENT - EXAMPLE CONFIGURATION USING WITH

systemd socket definition file	149 B

▼ LINUX/UNIX AGENTS

Check_MK Agent for AIX	13.13 kB

Install it:



check-mk-agent

Checkmk Agent for Linux

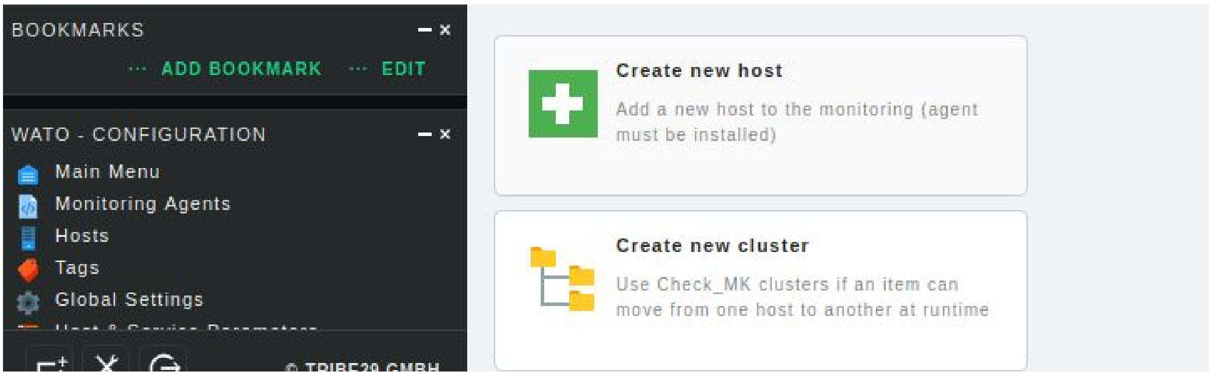
The Checkmk Agent uses xinetd or systemd to provide information about the system on TCP port 6556. This can be used to monitor the host via Checkmk.

(Converted from a rpm package by alien version 8.95.)

Check the installation worked fine by running `check_mk_agent` in the terminal. It should print out a too-many-lettered text.

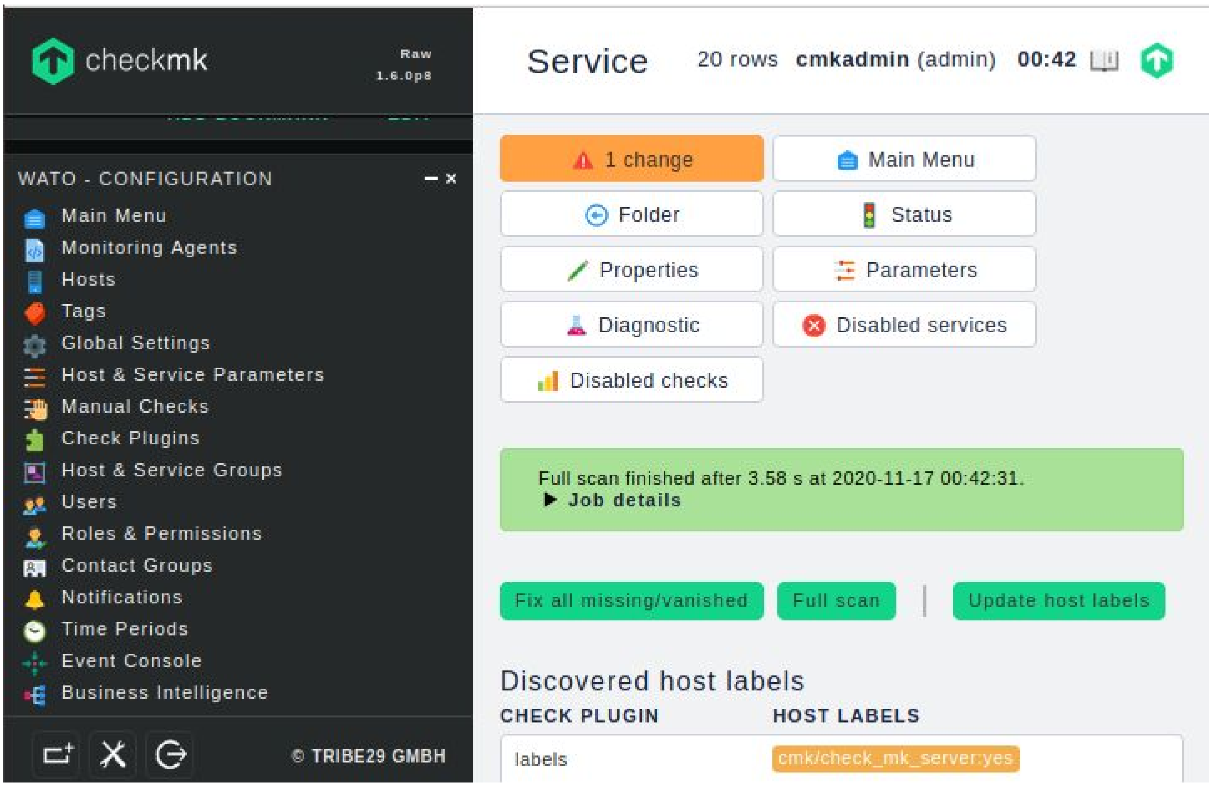
Adding server and client to **mysite**:

Go to WATO - Configuration - Hosts and click “Create new host”:



1) Give it a hostname - client or server
2) Tick “IPv4 address” box and type in 127.0.0.1 since the site is hosted on your machine
3) Tick MK_agent box to enable its monitoring by the previously installed agent

Tap save&go to services button:

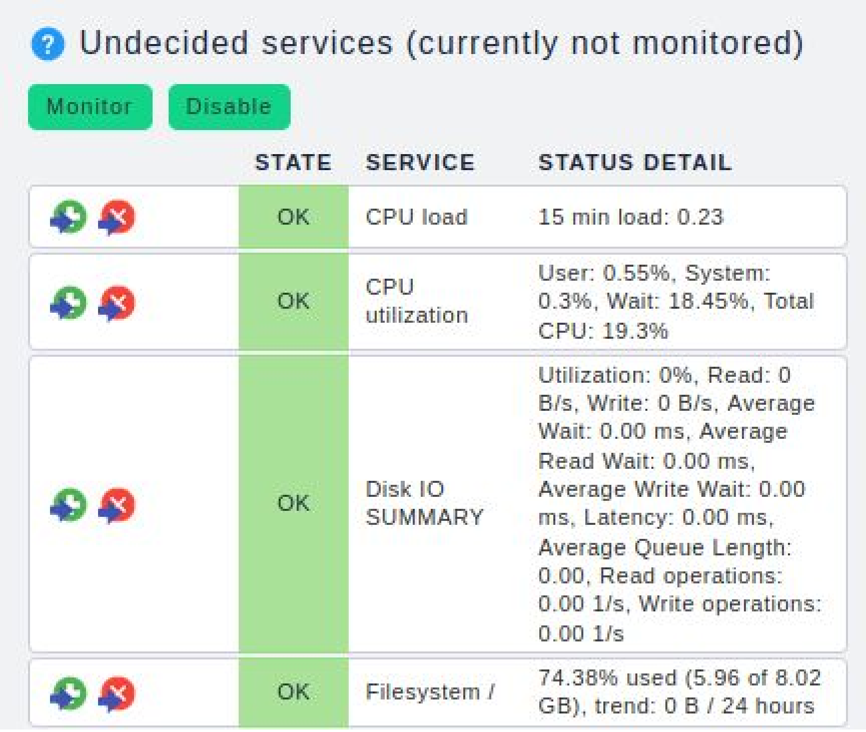


Full scan finished after 3.58 s at 2020-11-17 00:42:31.

Discovered host labels

CHECK PLUGIN	HOST LABELS
labels	cmk/check_mk_server:yes

See the list of not monitored services and press “Monitor”

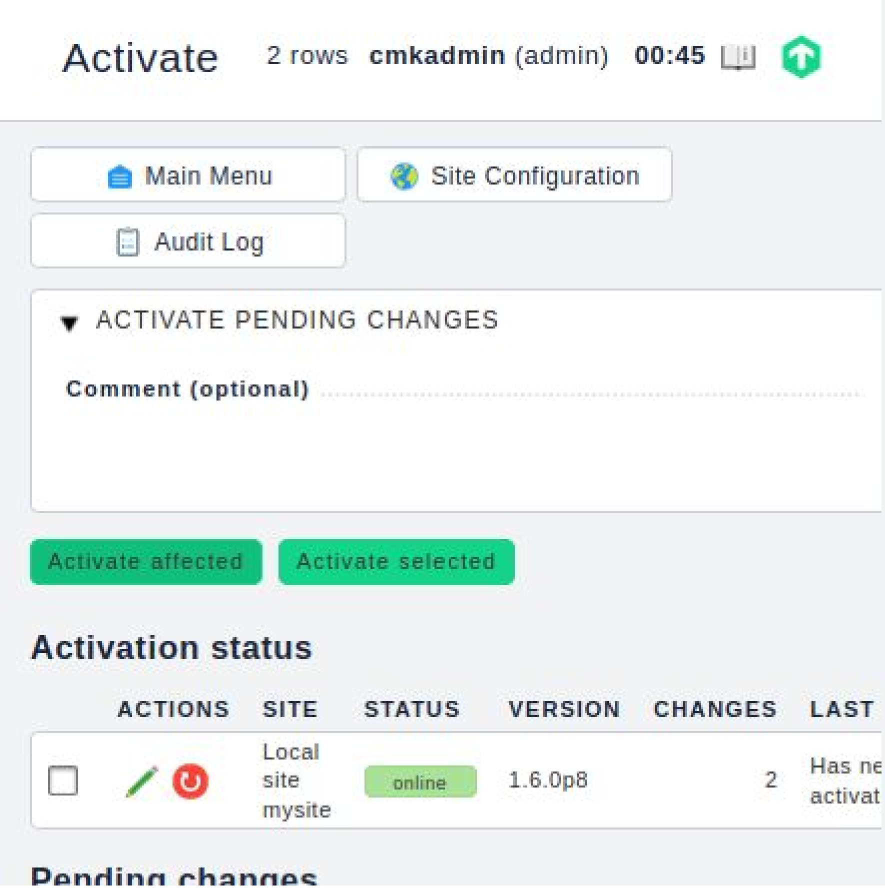


Undecided services (currently not monitored)

Monitor Disable

	STATE	SERVICE	STATUS DETAIL
	OK	CPU load	15 min load: 0.23
	OK	CPU utilization	User: 0.55%, System: 0.3%, Wait: 18.45%, Total CPU: 19.3%
	OK	Disk IO SUMMARY	Utilization: 0%, Read: 0 B/s, Write: 0 B/s, Average Wait: 0.00 ms, Average Read Wait: 0.00 ms, Average Write Wait: 0.00 ms, Latency: 0.00 ms, Average Queue Length: 0.00, Read operations: 0.00 1/s, Write operations: 0.00 1/s
	OK	Filesystem /	74.38% used (5.96 of 8.02 GB), trend: 0 B / 24 hours

Then press “N changes” button at the top and activate the affected changes:



Activate 2 rows **cmkadmin** (admin) **00:45**

Main Menu Site Configuration

Audit Log

▼ ACTIVATE PENDING CHANGES

Comment (optional)

Activate affected Activate selected

Activation status

	ACTIONS	SITE	STATUS	VERSION	CHANGES	LAST
□		Local site mysite	online	1.6.0p8	2	Has ne activat

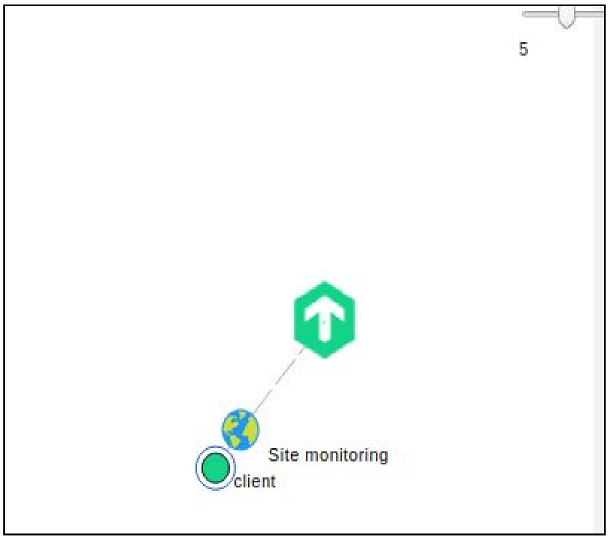
Pending changes

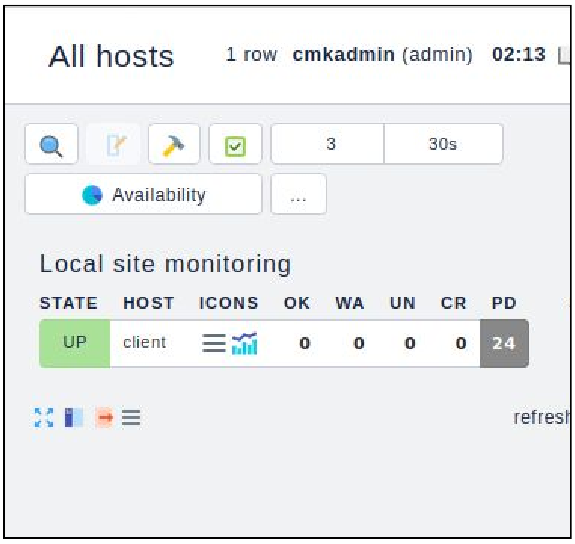
Add the other client to the other site

In the same manner as described above I added the “client” host to **“monitoring”** site.

As a result the following architecture is built:

On “monitoring” site:
client

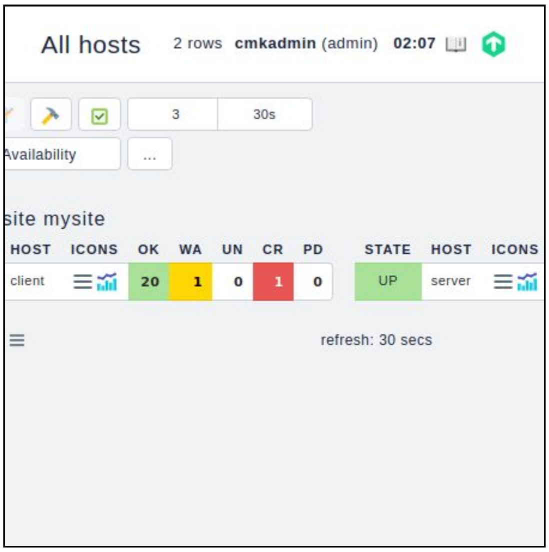



All hosts 1 row **cmkadmin** (admin) **02:13**

Local site monitoring

STATE	HOST	ICONS	OK	WA	UN	CR	PD
UP	client	☰	0	0	0	0	24

On “mysite” site:
client, server

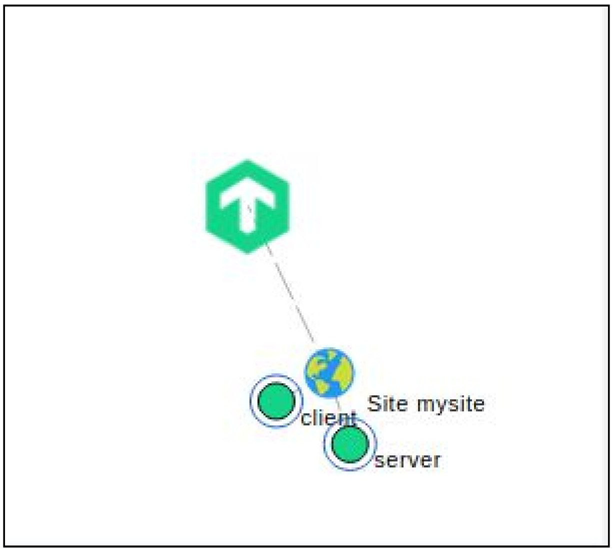

All hosts 2 rows **cmkadmin** (admin) **02:07**

site mysite

HOST	ICONS	OK	WA	UN	CR	PD
client	☰	20	1	0	1	0

STATE	HOST	ICONS
UP	server	☰

refresh: 30 secs



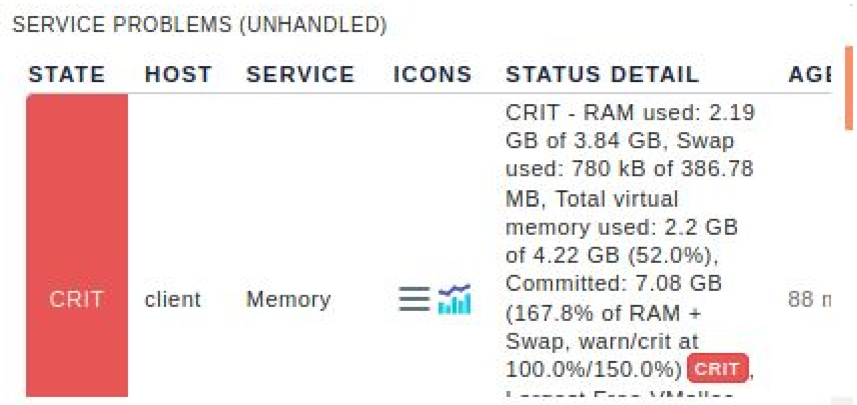
Give a short report of what is going wrong with your server and clients

Go to VIEWS - Host & Services Problems tabs

Here we see the following problem and warning (it is the same for both sites):

Critical memory state:

This problem occurs because of committed memory - private virtual address space of a process. The committed memory does not have a particular location in the system - it could be either ram or swap area or both. Some processes reserve too much memory that is rarely used in reality still, it could not be used by other processes that really might be in high demand for it.



SERVICE PROBLEMS (UNHANDLED)

STATE	HOST	SERVICE	ICONS	STATUS DETAIL	AGI
CRIT	client	Memory	☰	CRIT - RAM used: 2.19 GB of 3.84 GB, Swap used: 780 kB of 386.78 MB, Total virtual memory used: 2.2 GB of 4.22 GB (52.0%), Committed: 7.08 GB (167.8% of RAM + Swap, warn/crit at 100.0%/150.0%) CRIT,	88 n

Bonus

What are IDS, IPS and honeypot?

IDS - this is a set of **software or hardware tools** that identify facts of unauthorized access to the corporate system. The main functions of IDS systems are intrusion network **attacks detection**, predicting and **searching for vulnerabilities**, recognition of the **source of the attack** and ensuring **quality control** of system administration.

IPS are similar to IDS in the sense of functional features. However, IPS **does not allow you to constantly monitor** the situation in real time and, accordingly, perform timely actions to **prevent attacks**. The system helps **prevent the most popular network attacks**, for example, against vulnerable components of information systems and services, attacks aimed at increasing rights and privileges or obtaining unauthorized access to confidential information.

Honeypot is an information tool that helps you **study existing threats** and **identify new** ones. Using the collected data, you can prioritize issues and correctly allocate information security resources. The trap **simulates a computer system** with applications and data, and cybercriminals take it for the real thing. Trapped hackers can be monitored to **learn more about their behavior** and create more effective ways to protect real systems. To make traps more attractive they are **deliberately made vulnerable** via using ports that can be detected by scanning, or untrusted passwords.

What is a DMZ, how many types of it are out there, which one do you prefer.

DMZ is a network segment that contains public services and **separates** them from the local (private) network. It adds an additional **layer of security** on the local network to minimize damage in the event of an attack on one of the public services since the attacker has direct access only to the DMZ hardware.

Configurations:

1) **Three firewalls**
   In this configuration, the first firewall accepts requests from the external network, the second one controls the DMZ network connections, and the third one controls the internal network connections.

2) **Weak Screened**
   For this configuration, a single firewall with at least three network interfaces is used: one for connecting to the provider (WAN), the second to the internal network (LAN), and the third to the DMZ.

3) **Strong Screened**
   To create a DMZ, two firewalls are used: one of them controls connections from the external network to the DMZ, and the second one controls connections from the DMZ to the internal network.

For the purpose of security, I would choose the third configuration, since in this case, **both devices must be compromised** to successfully attack internal resources. In addition, you can configure slower application-level filtering rules on the external screen, providing enhanced local network protection **without negatively affecting the performance** of the internal segment. An even higher level of protection can be provided by using two firewalls **from different manufacturers** and different architectures - this reduces the likelihood that both devices will have the same vulnerability.

For a good price/security ratio I would choose the **second architecture** since the third one is quite expensive to implement. The tradeoff price is the **increased requirements** for hardware and administration: the firewall must handle all traffic going both to the DMZ and to the internal network. At the same time, it becomes a **single point of failure**.

Sources:

https://www.digitalocean.com/community/tutorials/how-to-monitor-server-health-with-checkmk-on-ubuntu-18-04

https://checkmk.com/cms_omd_basics.html

https://ru.bmstu.wiki/index.php?title=DMZ_(Demilitarized_zone)&mobileaction=toggle_view_mobile

http://etutorials.org/Linux+systems/secure+linux-based+servers/Chapter+2.+Designing+Perimeter+Networks/Section+2.2.+Types+of+Firewall+and+DMZ+Architectures/

https://www.kaspersky.ru/resource-center/threats/what-is-a-honeypot

ids vs ips

https://answers.microsoft.com/en-us/windows/forum/windows_10-performance-winpc/committed-memory-full-but-ram-has-empty-space/b39e9c2f-3d64-46f1-b7ce-cddf95bb152e

https://checkmk.com/cms_check_mem.linux.html