

Windows Monitoring over SSH

The ultimate security for invoking a Checkmk-agent is offered by invoking it via Secure Shell – in Linux in the form of an implementation of *OpenSSH*. Since Windows 10 and Windows Server 2019 OpenSSH is already included.

Step-by-step guide

1. Start OpenSSH Server

```
net start sshd
```

2. Create SSH Key pair

```
ssh-keygen
```

```
C:\ProgramData\checkmk\agent\plugins>ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (C:\Users\IEUser\.ssh/id_rsa):
Created directory 'C:\Users\IEUser\.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in C:\Users\IEUser\.ssh/id_rsa.
Your public key has been saved in C:\Users\IEUser\.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:dH/h0I/vQ56C5rRICjbxEgxSRSGs2nvCdx+rasQkWQU ieuser@MSEEDGEWIN10
The key's randomart image is:
+---[RSA 2048]---+
|.oE*o|
|..o  |
|..+ . . . o|
|. + + . . . o +|
|.. + + S . + .|
|. . o + . . o|
|. o = ... .. o o|
| + = = oooo.. =|
| =.o.+o.o0 . o|
+---[SHA256]-----+
C:\ProgramData\checkmk\agent\plugins>
```

3. Create the file 'authorized_keys' in C:\Users<USERNAME>\.ssh\

```
fsutil file createnew authorized_keys 2000
```

- a. Put in the public key from Checkmk Site User

4. Now you can login without Password to the Windows machine

```
OMD[magnis_master]:~$ ssh IEUser@192.168.2.106

Microsoft Windows [Version 10.0.17763.379]
(c) 2018 Microsoft Corporation. Alle Rechte vorbehalten.

ieuser@MSEEDGEWIN10 C:\Users\IEUser>
```

5. Modify the authorized_keys on the Windows Host and restrict access to the execution on the agent

```
command="\"C":\\Program Files (x86)\\checkmk\\service\\check_mk_agent.exe\" test" ssh-rsa
AAAAC3NzaC1lZDI1NTE5AAAAIGb6AaqRPlbEmDnBkeIW3Q6Emb5lr2QEbWbEQLmA5pb48 mysite@mycmkserver
```

6. Go to Checkmk and configure a special agent for the Windows Hosts

- a. WATO CONFIGURATION Host & Service Parameters Datasource Programs Individual program call instead of agent access

Edit rule: Individual program call instead of agent access

Abort Prefef. conditions

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 must output the agent's data on st via SSH. You can use monitoring macros here. The most important are:

- \$HOSTADDRESS: The IP address of the host
- \$HOSTNAMES: 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[VARNAME]\$. Replace [VARNAME] with the upper case name of your variable. For example, a host attribute named foo with the value bar with the permission "Can add or modify executables".

RULE PROPERTIES

Description: Monitoring Windows Host over SSH

Comment:

Documentation URL:

Rule activation: do not apply this rule

INDIVIDUAL PROGRAM CALL INSTEAD OF AGENT ACCESS

Command line to execute: ssh IEUser@\$ _HOSTADDRESS_ 4\$

CONDITIONS

Condition type: Explicit conditions

Folder: Main directory

Host tags: Add tag condition

Host labels: Add label condition

Explicit hosts: Windows

Negate: make rule apply for all but the above hosts

Save Export

7. Modify the Datasource of the Windows Host
 - a. WATO CONFIGURATION Hosts Edit the properties of this host

DATA SOURCES

Check_MK Agent: Normal Checkmk agent, or special agent if configured

SNMP: No SNMP (Default value)

Piggyback: Use piggyback data from other hosts if present (Default value)

8. Now you can stop the Checkmk Agent Service on the Windows Host

```
net stop CheckMkService
```

9. Diagnosis

```
OMD[nagnis_master]:~$ cmk -d Windows_SSH |more
<<<check_mk>>>
Version: 1.6.0p19
BuildDate: Nov 16 2020
AgentOS: windows
Hostname: MSEDGEWIN10
Architecture: 64bit

or via SSH

OMD[nagnis_master]:~/ $ ssh IEUser@192.168.2.128 " "
or
OMD[nagnis_master]:~/ $ ssh -T IEUser@192.168.2.128
```



If OpenSSH is not already installed on your System, you can install it by following this manual: https://docs.microsoft.com/de-de/windows-server/administration/openssh/openssh_install_firstuse

If you want to learn more about configuring OpenSSH under Windows, please take a look at this manual: https://docs.microsoft.com/de-de/windows-server/administration/openssh/openssh_server_configuration

A more detailed manual: <https://forum.checkmk.com/t/windows-agent-abfrage-uber-ssh/22421>

Related articles

Content by label

There is no content with the specified labels

