**How to enable SNMP on Linux machines and Mac OS**

Top of Form

Search For

Bottom of Form

With Domotz [network monitoring software](https://www.domotz.com/) you can monitor Linux machines and MAC OS via SNMP.

It is sufficient to install an SNMP deamon on the Linux machine and make sure that the Domotz agent is using the same *read-only community* for that device.

In this articles you can find a list of instructions to learn how to enable SNMP for Linux and Mac OS.

Here is a list of instructions for installing SNMP deamons on most popular Linux distributions and on Mac OS:

[Debian](read://https_help.domotz.com/?url=https%3A%2F%2Fhelp.domotz.com%2Fuser-guide%2Fhow-to-enable-snmp-on-linux-machines%2F#L1)

[Ubuntu](read://https_help.domotz.com/?url=https%3A%2F%2Fhelp.domotz.com%2Fuser-guide%2Fhow-to-enable-snmp-on-linux-machines%2F#L2)

[CentOS](read://https_help.domotz.com/?url=https%3A%2F%2Fhelp.domotz.com%2Fuser-guide%2Fhow-to-enable-snmp-on-linux-machines%2F#L3)

[FreeBSD](read://https_help.domotz.com/?url=https%3A%2F%2Fhelp.domotz.com%2Fuser-guide%2Fhow-to-enable-snmp-on-linux-machines%2F#L4)

[Red Hat Enterprise Linux (RHEL)](read://https_help.domotz.com/?url=https%3A%2F%2Fhelp.domotz.com%2Fuser-guide%2Fhow-to-enable-snmp-on-linux-machines%2F#L5)

[Mac OS](read://https_help.domotz.com/?url=https%3A%2F%2Fhelp.domotz.com%2Fuser-guide%2Fhow-to-enable-snmp-on-linux-machines%2F#L6)

For more information about configuring SNMP in Domotz see [here](https://help.domotz.com/user-guide/snmp-tcp-services-monitoring/#2).

At the end of this article, you will also[find a practical use case for monitoring through Domotz the HW resources of your Linux machine, and a template to achieve that.](read://https_help.domotz.com/?url=https%3A%2F%2Fhelp.domotz.com%2Fuser-guide%2Fhow-to-enable-snmp-on-linux-machines%2F#download)

**How to enable SNMP on Linux**

**Debian**

**1. Installation**

$ sudo apt-get update

$ sudo apt-get install snmpd

**2. Configuration**

Edit the file: */etc/snmp/snmpd.conf* with a text editor of your choice.

Add the following line:  
*rocommunity public*

Note: The default Ubuntu “snmpd” configuration specifically denies almost everything useful with this configuration (limiting access to just enough information to tell that the machine is a Linux one):  
  
view systemonly included .1.3.6.1.2.1.1  
view systemonly included .1.3.6.1.2.1.25.1  
rocommunity public default -V systemonly  
  
This configuration locks the machine down, which may be “safe” if it will be on an insecure network with little SNMP administration knowledge available. However, if you are willing to monitor the performance of your machine with an external systems, comment them and replace the access with a secure mechanism (either with SNMP v3 or different community string).

Comment the line:  
*#agentAddress udp:127.0.0.1:161*

Uncomment the line:   
*agentAddress udp:161,udp6:[::1]:161*

Restart the snmpd service:

$ */etc/init.d/snmpd restart*

**Ubuntu**

**1. Installation**

$ sudo apt-get update

$ sudo apt-get install snmpd

**2. Configuration**

Edit the file: */etc/snmp/snmpd.conf* with a text editor of your choice.

Add the following line:  
*rocommunity public*

Note: The default Ubuntu “snmpd” configuration specifically denies almost everything useful with this configuration (limiting access to just enough information to tell that the machine is a Linux one):  
  
view systemonly included .1.3.6.1.2.1.1  
view systemonly included .1.3.6.1.2.1.25.1  
rocommunity public default -V systemonly  
  
This configuration locks the machine down, which may be “safe” if it will be on an insecure network with little SNMP administration knowledge available. However, if you are willing to monitor the performance of your machine with an external systems, comment them and replace the access with a secure mechanism (either with SNMP v3 or different community string).

Comment the line:  
*#agentAddress udp:127.0.0.1:161*

Uncomment the line:   
*agentAddress udp:161,udp6:[::1]:161*

Restart the snmpd service:

$ */etc/init.d/snmpd restart*

3. **Allow SNMP ports in Firewal**l

Execute the following commands to allow necessary ports:

$ *ufw allow 161/udp*

$ *ufw allow 162/udp*

**CentOS**

**1. Installation**

$ *yum update*

$ *yum install net-snmp*

**2. Configuration**

Edit the file: */etc/snmp/snmpd.conf*

Add the following line:  
*rocommunity public*

Replace the line below:  
*view systemview included .1.3.6.1.2.1.25.1.1*  
with the following line:  
*view systemview included .1.3.*

Restart the snmpd service:

$ */etc/init.d/snmpd restart*

**3. Allow SNMP ports in Firewall**

Execute the following commands to allow necessary ports:

$ *firewall-cmd --zone=public --add-port=161/udp --permanent*

$ *firewall-cmd --zone=public --add-port=162/udp --permanent*

$ *firewall-cmd --reload*

**FreeBSD**

**1. Fetch Ports collection and install Installation**

$ *portsnap fetch*

$ *portsnap extract*

$ *cd /usr/ports/net-mgmt/net-snmp*

$ *make install clean*

**2. Configuration**

Copy the following configuration file:  
*cp /usr/local/share/snmp/snmpd.conf.example /usr/local/etc/snmpd.conf*

Add the following line:  
*rocommunity public*

Comment the line:  
*#agentAddress udp:127.0.0.1:161*

Uncomment the line:  
*agentAddress udp:161,udp6:[::1]:161*

**3. Allow SNMP ports in Firewall**

Edit the */etc/rc.conf* file and add the following lines:

*snmpd\_enable=”YES”*  
*snmpd\_conffile=”/usr/local/etc/snmpd.conf”*

Uncomment the line:  
*agentAddress udp:161,udp6:[::1]:161*

**4. Start the Service**

$ *service snmpd start*

Edit the */etc/rc.conf* file and add the following lines:

**Red Hat Enterprise Linux (RHEL)**

**1. Installation**

*yum install net-snmp*

**2. Configuration**

Edit the file: */etc/snmp/snmpd.conf*

Add the following lines:  
*rocommunity public  
agentAddress udp:161,udp6:[::1]:161*

Start the snmpd service:  
*systemctl enable snmpd && systemctl start snmpd*

**3. Allow SNMP ports in Firewall**

Execute the following commands to allow necessary ports:

*firewall-cmd --zone=public --add-port=161/udp --permanent*

*firewall-cmd --zone=public --add-port=162/udp --permanent*

*firewall-cmd --reload*

Edit the */etc/rc.conf* file and add the following lines:

**How to enable SNMP on Mac OS**

**1. Configuration**

Edit the file: */etc/snmp/snmpd.conf*

Add the following line:  
*rocommunity public*

**2. Start the SNMP service**

Execute the following commands to allow necessary ports:

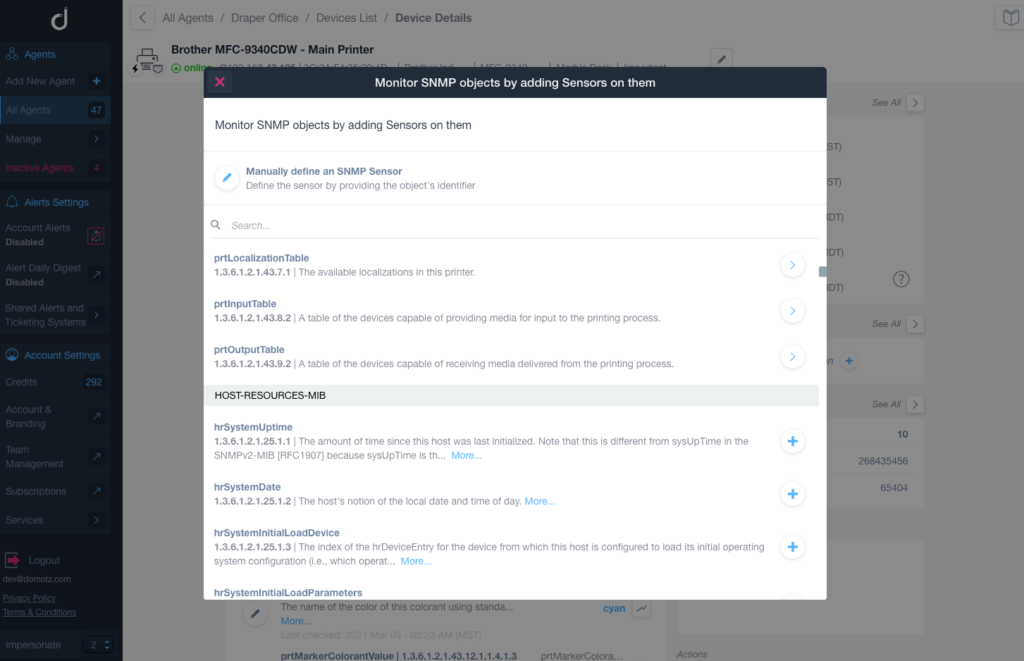
*sudo launchctl load -w /System/Library/LaunchDaemons/org.net-snmp.snmpd.plist*

**Monitoring SNMP OiD through Domotz**

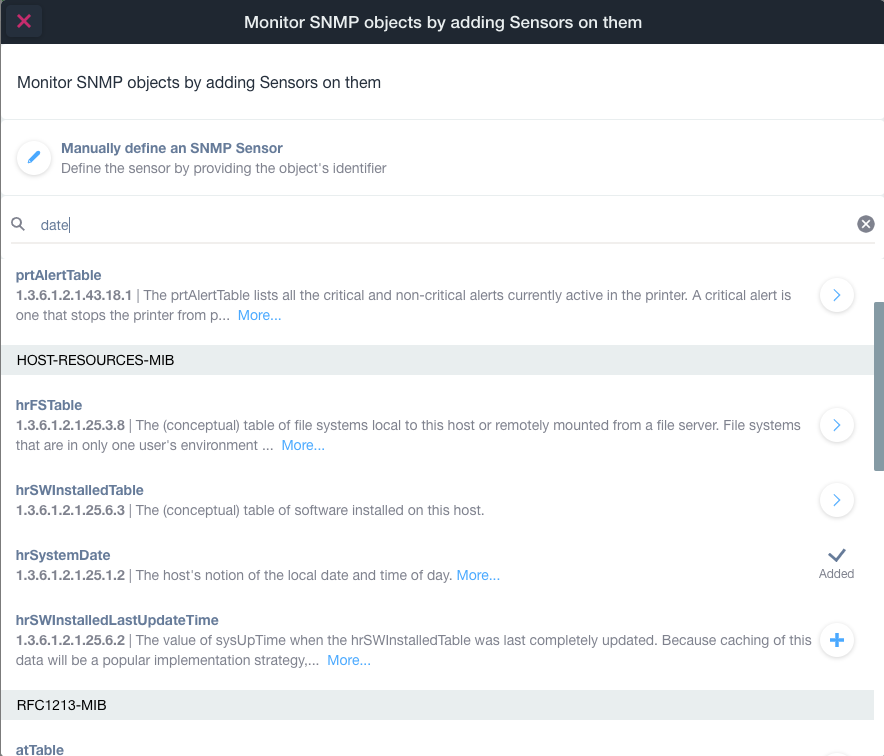
Once the SNMP daemon has been configured on the Linux machine, and the Community Strings (for V1 and V2 – or Credentials for V3) are matching between the Linux machine and the Domotz Agent, Domotz will automatically browse and report all the possible MIBs available for that specific Linux machine.

In particular, you can search with Domotz for any possible specific OiD available on that specific Linux Machine.

Open the Domotz App or the WebApp. Identify your Linux Machine. Under the tab **SNMP/TCP** click on **Add an SNMP Sensor**. You will be proposed with the entire list of SNMP OiD available for that specific device, and the possibility to search among the descriptions (available in the MIB available for that device):



You can for instance search for all the possible OiD which contains the word “CPU” in their description, and then add the relevant ones.

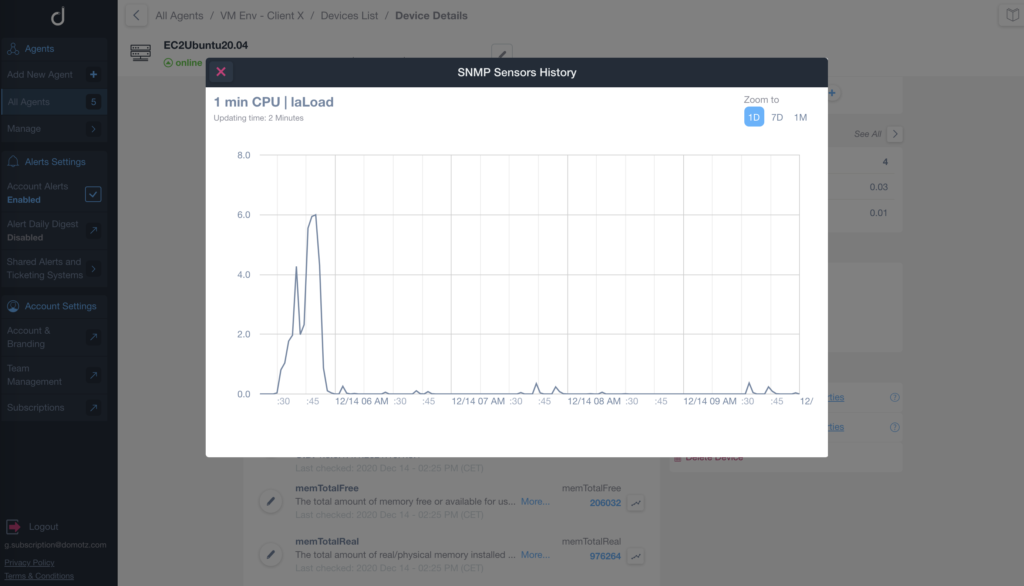


By doing this, you can start monitoring the resource consumptions of your Linux machine.

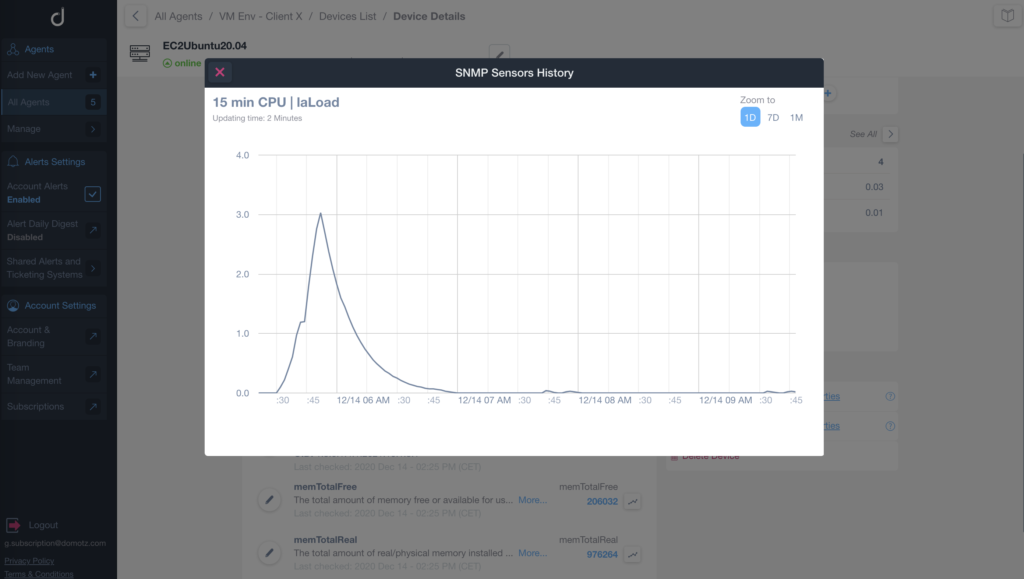
Below in this article a few example of resource usage statistics extracted by Domotz out of a Linux machine using this mechanism.

**CPU Usage**

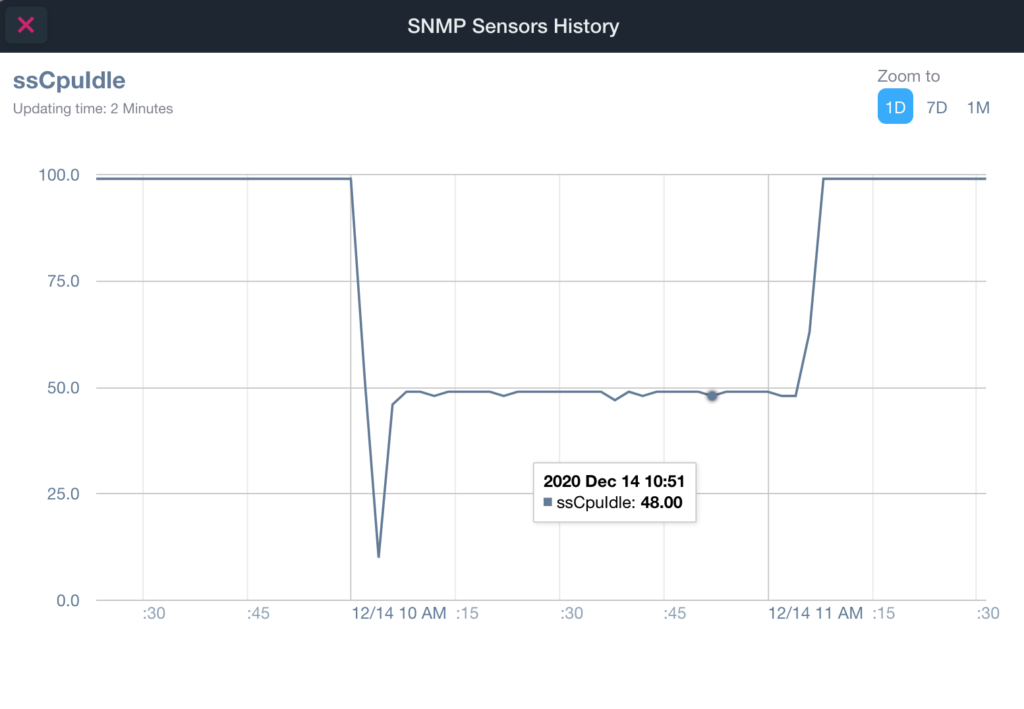
Through Domotz you can monitor either the spikes in the CPU usage as recorded by the CPU Load on 1 Min average:



Or the sustained CPU usage as recorded by the CPU Load on 15 Min average:

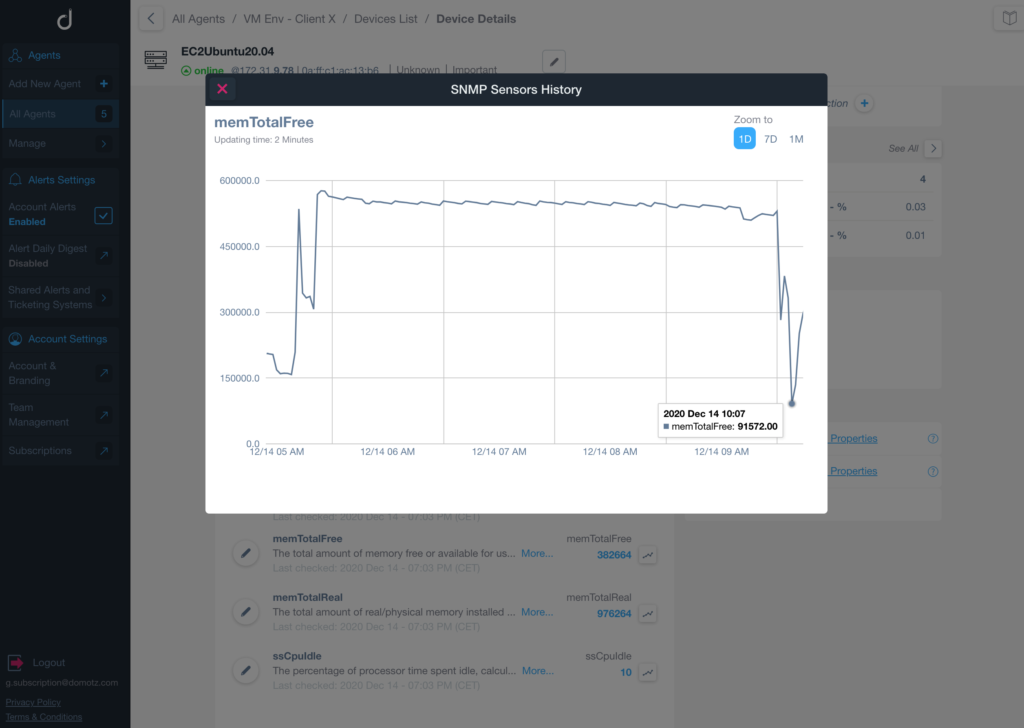


It is also possible to monitor the residual (idle) CPU available in the server:



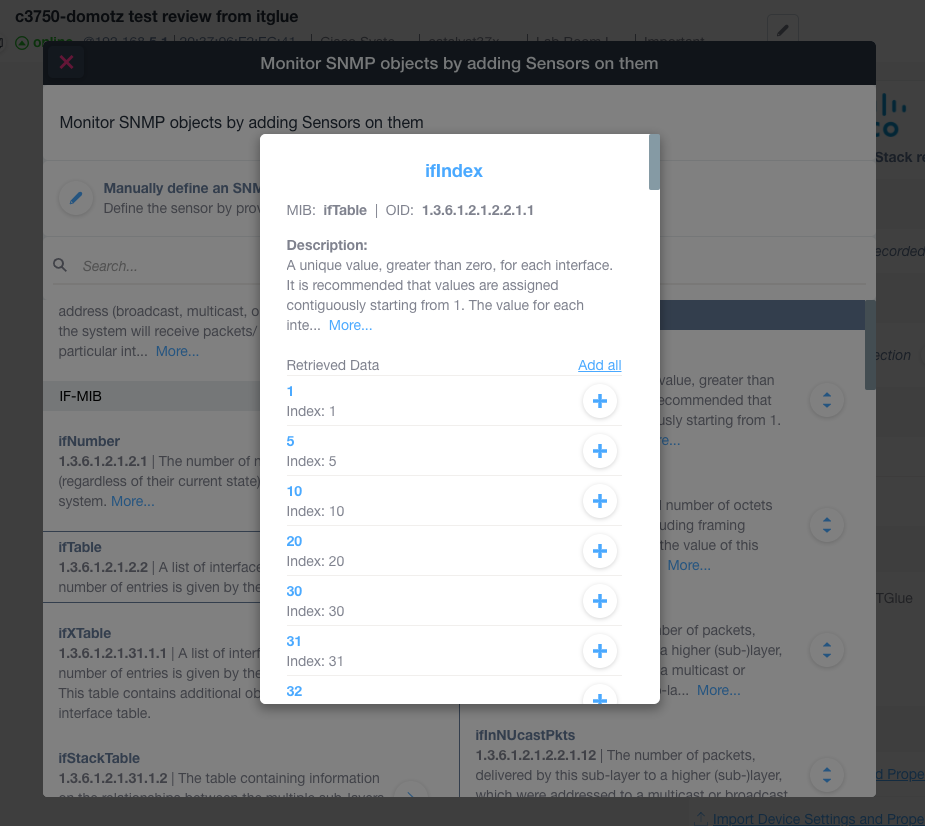
**Memory Usage**

Through Domotz you can monitor the total available Memory on the Linux machine, or even more importantly the residual memory still free and available on the same:



**Volume or Storage occupancy**

It is also possible to add the monitoring of any volume or storage and their occupancy via the default hrStorageTable:

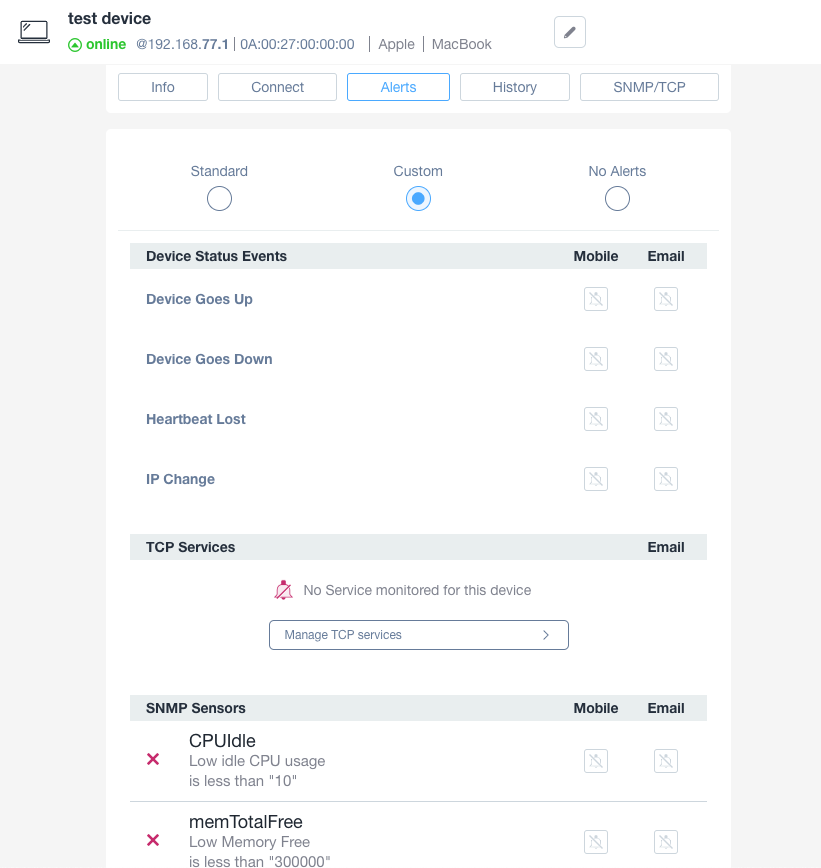


For more information about configuring SNMP in Domotz see [here](https://help.domotz.com/user-guide/snmp-tcp-services-monitoring/#2).

**Alerts on resource usage**

Both in the case of the above examples, you can configure alerts so that you are notified if any unexpected load is affecting your server or Linux machine.

For example you can monitor if the CPU Idle at a certain point goes below a certain threshold, or if the residual memory is too low to have a safe environment.



In order to configure alerts, visit the **Alert**tab, then select the **Custom** type under the **Personal Alerts**, and **Configure an SNMP Alert**.

You will be be provided with the option to select the previously added SNMP OiD as input for the alert.

You can then either decide to receive the notification via Email or Push Notification, as Personal Alerts, or receive the notifications on your preferred channel, via the Shared Alert configuration:

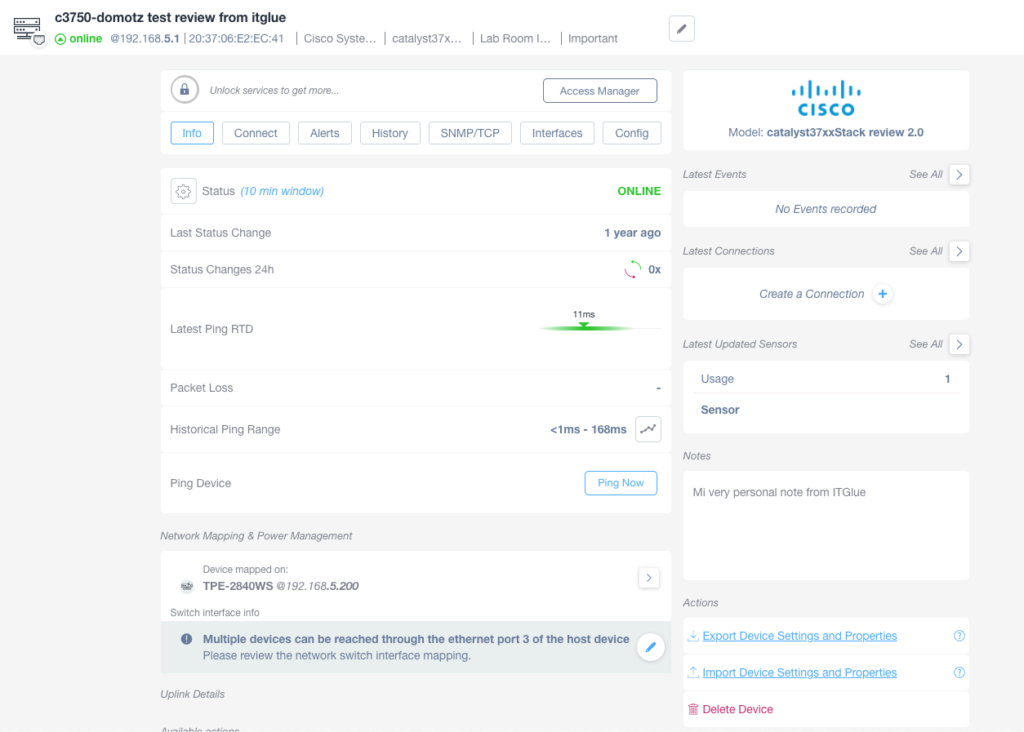
* through shared email addresses
* Your ticketing system or PSA tool integrated with Domotz (e.g. Datto Autotask, ConnectWise Manage, Syncro MSP, Zendesk, Freshservice, etc)
* Your company chat system (e.g. Microsoft Teams or Slack)
* Any Web Server through Web Hooks (or possible third party integrations via Zapier or Node-Red)

**Add Resource Usage monitoring via templateing**

With Domotz it is possible to configure the monitoring and alerting via SNMP for a specific Linux machine (or generically for any server) and then use that configuration as a template for all the other servers that need to be monitored.

This can be achieved by the **Export/Import Device Settings and Properties** capability.

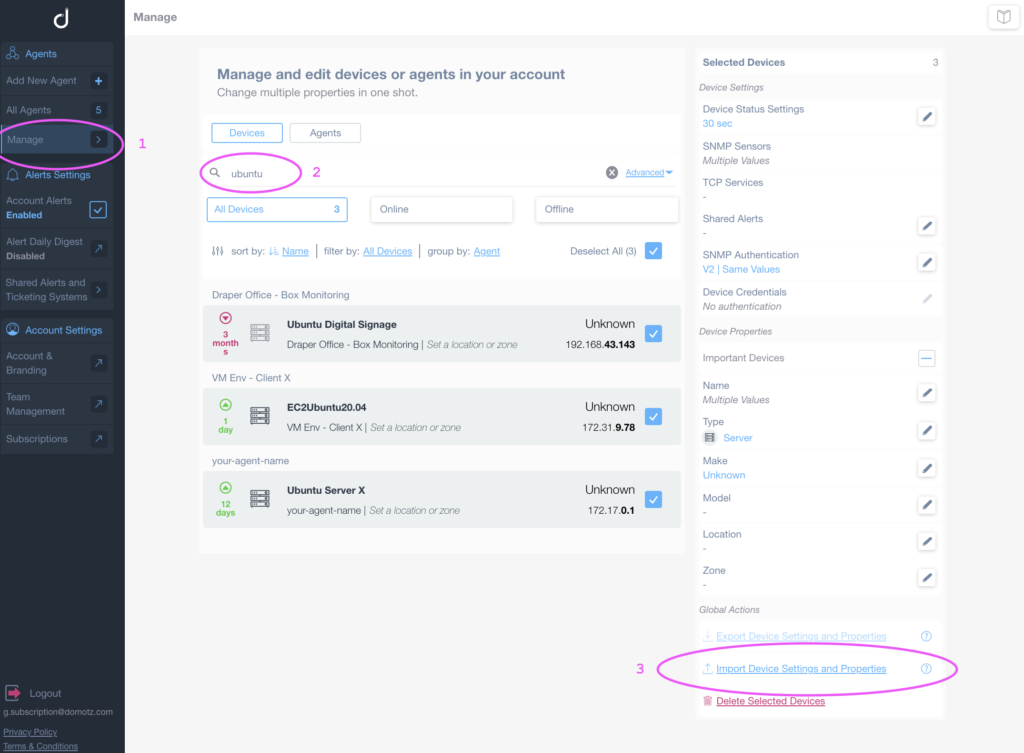
Once you have completed the configuration of the SNMP parameters you are willing to monitor for a Linux machine, and the alerts’ thresholds to get notified, click on the **Export Device Settings and Properties**.



A *json* file (like a template file) is generated and saved on your client with all the Domotz settings and properties for that specific device.

This same *template json* file can then be edited (e.g. to replace or remove some areas) and then used to apply the same SNMP settings (e.g. OiD to be monitored and their threshold) to all the other Linux machines that you are monitoring through Domotz.

This can be achieved from the [Global Search and Manage functionality](https://help.domotz.com/user-guide/dashboards-global-search/#D2). Search for all the Linux machines you want to apply the template, select them with the multi-select check box, and then click on **Import Device Settings and Properties**:

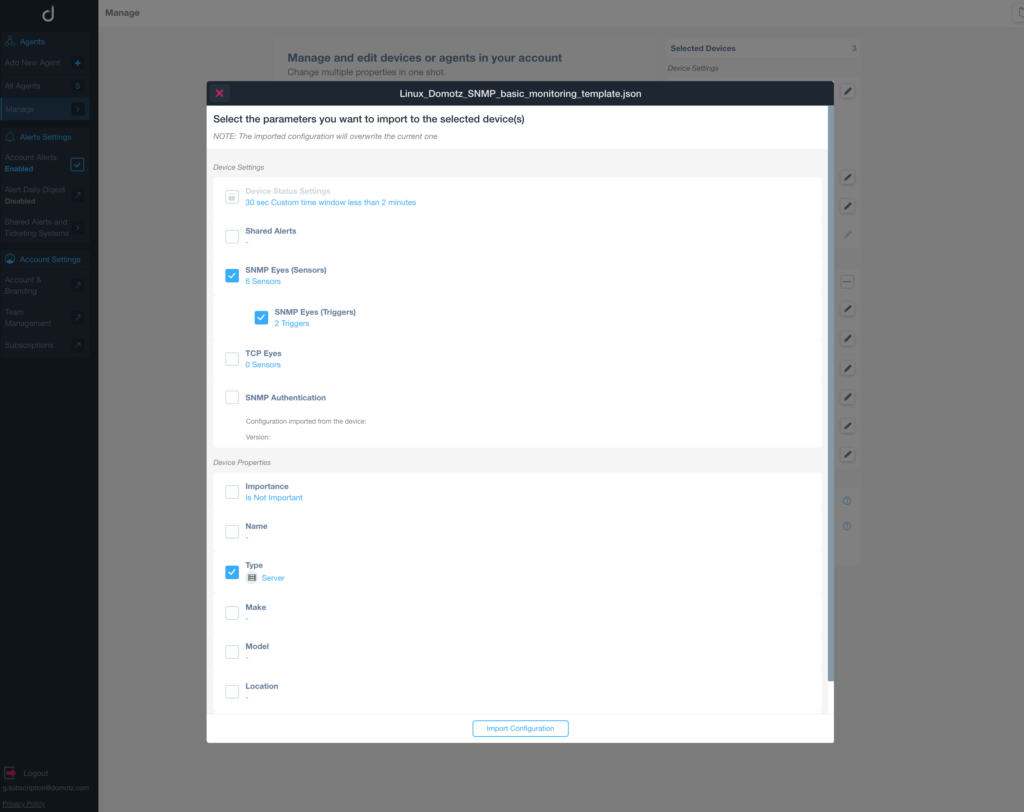


**Preconfigured Template for Linux Machine Resource usage Monitoring**

The following file can be used as a baseline to build a template for monitoring resource usage of your Linux Machine:

[Linux\_Domotz\_SNMP\_basic\_monitoring\_template.jsonDownload](https://1chz0y1u0oxp1mqijz44l0ra-wpengine.netdna-ssl.com/wp-content/uploads/2020/12/Linux_Domotz_SNMP_basic_monitoring_template.json_.zip)

In particular, once used for the **Import Device Settings and Properties** functionality, it will create 6 SNMP sensors (on CPU and Memory usage) and two triggers (alerts) on the usage of those resources:



Each device will then have the SNMP and TCP configured with the OiDs for monitoring CPU and Memory usage via SNMP:

