



How to check the server temperature via Proxmox



Before starting, make sure to back up your system. Any changes are your own responsibility.

Access the main screen of your Proxmox.

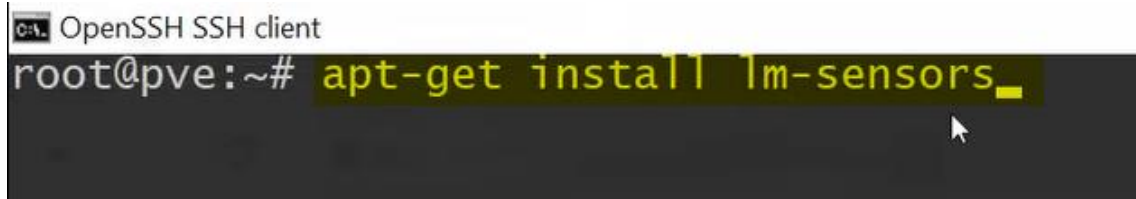
The screenshot shows the Proxmox web interface. On the left, the 'Datacenter' tree lists various nodes. The main panel displays the 'Summary' page for a node. The 'CPU Thermal State' is highlighted, showing temperatures for 16 CPUs ranging from 36 to 31 °C. A red arrow points to the 'CPU Thermal State' section. A watermark 'github.com/felipemoreiravieira' is visible.

Access the shell terminal.

The screenshot shows the Proxmox web interface with the 'Shell' option selected in the left sidebar. The terminal displays the Debian GNU/Linux login prompt. A red arrow points to the 'Shell' option. A watermark 'github.com/felipemoreiravieira' is visible.

Install the lm-sensors package.

```
apt-get install lm-sensors
```



```
OpenSSH SSH client
root@pve:~# apt-get install lm-sensors_
```

After installation, you should have access to the sensors.

Access the sensors.

```
sensors
```



```
OpenSSH SSH client
root@pve:~# apt-get install lm-sensors
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
lm-sensors is already the newest version (1:3.6.0-7).
The following package was automatically installed and is no longer required:
  libopenbsd0
Use 'apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 163 not upgraded.
root@pve:~# sensors
drivetemp-scsi-4-0
Adapter: SCSI adapter
temp1: +32.0°C

coretemp-isa-0000
Adapter: ISA adapter
Package id 0: +42.0°C (high = +79.0°C, crit = +89.0°C)
Core 0: +42.0°C (high = +79.0°C, crit = +89.0°C)
Core 1: +41.0°C (high = +79.0°C, crit = +89.0°C)
Core 2: +41.0°C (high = +79.0°C, crit = +89.0°C)
Core 3: +42.0°C (high = +79.0°C, crit = +89.0°C)
Core 4: +41.0°C (high = +79.0°C, crit = +89.0°C)
Core 5: +40.0°C (high = +79.0°C, crit = +89.0°C)
Core 6: +39.0°C (high = +79.0°C, crit = +89.0°C)
Core 7: +41.0°C (high = +79.0°C, crit = +89.0°C)

drivetemp-scsi-5-0
Adapter: SCSI adapter
temp1: +32.0°C

drivetemp-scsi-3-0
Adapter: SCSI adapter
temp1: +37.0°C (low = +0.0°C, high = +60.0°C)
(crit low = -40.0°C, crit = +70.0°C)
(lowest = +24.0°C, highest = +37.0°C)

root@pve:~#
```

Next step is to access this file:

```
vim /usr/share/perl5/PVE/API2/Nodes.pm
```

```
OpenSSH SSH client
root@pve:~# apt-get install lm-sensors
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
lm-sensors is already the newest version (1:3.6.0-7).
The following package was automatically installed and is no longer required:
  libopencc0
Use 'apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 163 not upgraded.
root@pve:~# sensors
drivetemp-scsi-4-0
Adapter: SCSI adapter
temp1: +32.0°C

coretemp-isa-0000
Adapter: ISA adapter
Package id 0: +42.0°C (high = +79.0°C, crit = +89.0°C)
Core 0: +42.0°C (high = +79.0°C, crit = +89.0°C)
Core 1: +41.0°C (high = +79.0°C, crit = +89.0°C)
Core 2: +41.0°C (high = +79.0°C, crit = +89.0°C)
Core 3: +42.0°C (high = +79.0°C, crit = +89.0°C)
Core 4: +41.0°C (high = +79.0°C, crit = +89.0°C)
Core 5: +40.0°C (high = +79.0°C, crit = +89.0°C)
Core 6: +39.0°C (high = +79.0°C, crit = +89.0°C)
Core 7: +41.0°C (high = +79.0°C, crit = +89.0°C)

drivetemp-scsi-5-0
Adapter: SCSI adapter
temp1: +32.0°C

drivetemp-scsi-3-0
Adapter: SCSI adapter
temp1: +37.0°C (low = +0.0°C, high = +60.0°C)
(crit low = -40.0°C, crit = +70.0°C)
(lowest = +24.0°C, highest = +37.0°C)

root@pve:~# vim /usr/share/perl5/PVE/API2/Nodes.pm
```

You will look for the text:

“version_text”

```
OpenSSH SSH client

my $meminfo = PVE::ProcFSTools::read_meminfo();
$res->{memory} = {
    free => $meminfo->{memfree},
    total => $meminfo->{memtotal},
    used => $meminfo->{memused},
};

$res->{ksm} = {
    shared => $meminfo->{memshared},
};

$res->{swap} = {
    free => $meminfo->{swapfree},
    total => $meminfo->{swaptotal},
    used => $meminfo->{swapused},
};

$res->{pveversion} = PVE::pvecfg::package() . "/" .
    PVE::pvecfg::version_text();

# $res->{thermalstate} = 'sensors -j';

my $dinfo = df('/', 1); # output is bytes

$res->{rootfs} = {
    total => $dinfo->{blocks},
    avail => $dinfo->{bavail},
    used => $dinfo->{used},
    free => $dinfo->{blocks} - $dinfo->{used},
};

return $res;
}});

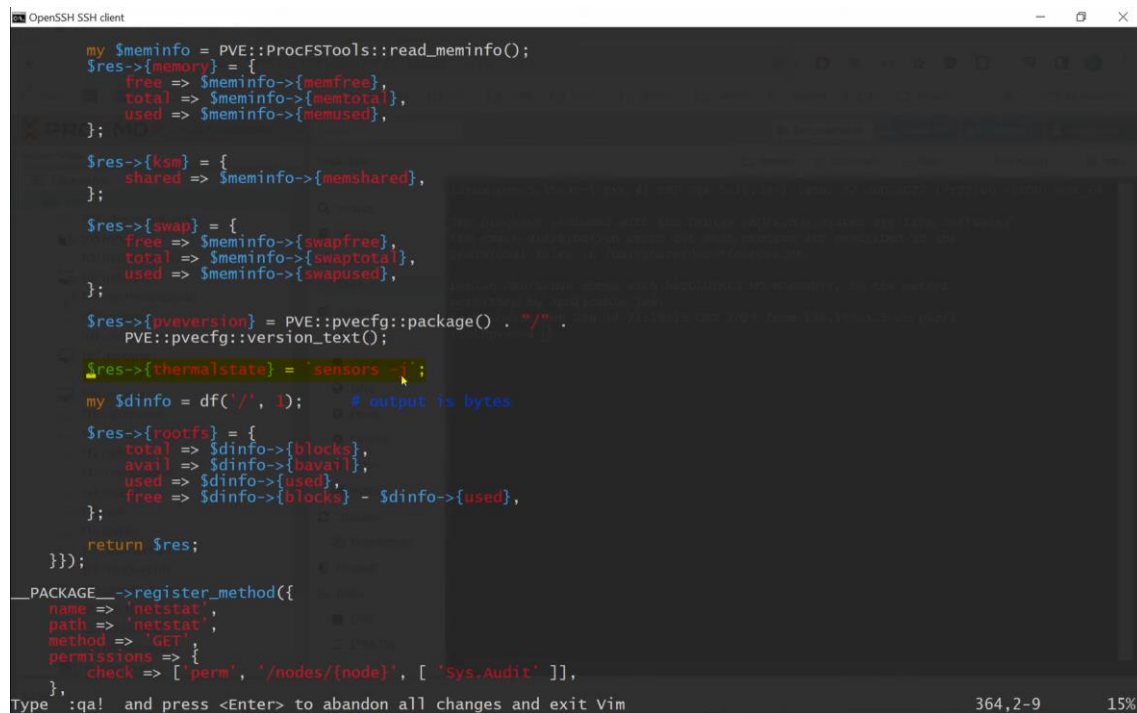
PACKAGE->register_method([
    name => 'netstat',
    path => 'netstat',
    method => 'GET',
    permissions => {
        check => ['perm', '/nodes/{node}', [ 'Sys.Audit' ]],
    },
], /version_text_
```

Add this command:

```
$res->{thermalstate} = `sensors -j`;
```

basically, this calls the sensors program and obtains the CPU temperature, storing it in this variable.

Save and exit.



```
my $meminfo = PVE::ProcFSTools::read_meminfo();
$res->{memory} = {
    free => $meminfo->{memfree},
    total => $meminfo->{memtotal},
    used => $meminfo->{memused},
};

$res->{ksm} = {
    shared => $meminfo->{memshared},
};

$res->{swap} = {
    free => $meminfo->{swapfree},
    total => $meminfo->{swaptotal},
    used => $meminfo->{swapused},
};

$res->{pveversion} = PVE::pvecfg::package() . "/" .
    PVE::pvecfg::version_text();

$res->{thermalstate} = `sensors -j`;

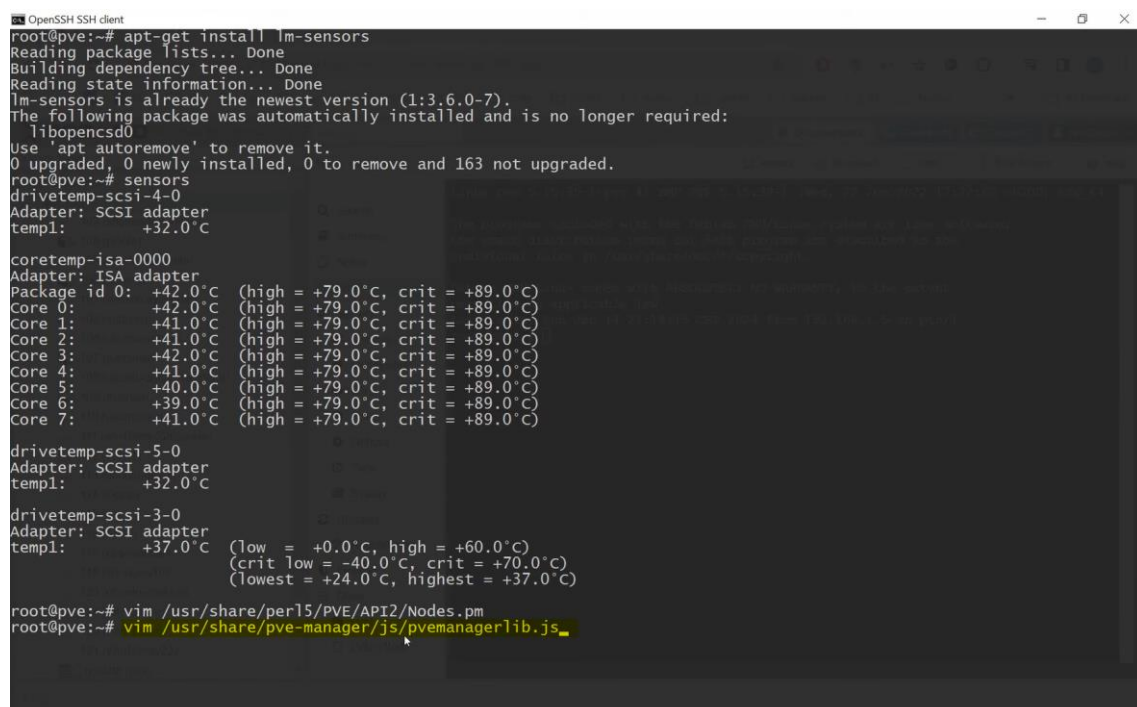
my $dinfo = df('/', 1); # output is bytes
$res->{rootfs} = {
    total => $dinfo->{blocks},
    avail => $dinfo->{bavail},
    used => $dinfo->{used},
    free => $dinfo->{blocks} - $dinfo->{used},
};

return $res;
});

__PACKAGE__->register_method({
    name => 'netstat',
    path => 'netstat',
    method => 'GET',
    permissions => {
        check => ['perm', '/nodes/{node}', [ 'Sys.Audit' ]],
    },
},
);
Type :qa! and press <Enter> to abandon all changes and exit Vim
```

Next step is to access this file:

```
vim /usr/share/pve-manager/js/pvemanagelib.js
```



```
root@pve:~# apt-get install lm-sensors
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
lm-sensors is already the newest version (1:3.6.0-7).
The following package was automatically installed and is no longer required:
  libopenbsd0
Use 'apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 163 not upgraded.
root@pve:~# sensors
drivetemp-scsi-4-0
Adapter: SCSI adapter
temp1: +32.0°C

coretemp-isa-0000
Adapter: ISA adapter
Package id 0: +42.0°C (high = +79.0°C, crit = +89.0°C)
Core 0: +42.0°C (high = +79.0°C, crit = +89.0°C)
Core 1: +41.0°C (high = +79.0°C, crit = +89.0°C)
Core 2: +41.0°C (high = +79.0°C, crit = +89.0°C)
Core 3: +42.0°C (high = +79.0°C, crit = +89.0°C)
Core 4: +41.0°C (high = +79.0°C, crit = +89.0°C)
Core 5: +40.0°C (high = +79.0°C, crit = +89.0°C)
Core 6: +39.0°C (high = +79.0°C, crit = +89.0°C)
Core 7: +41.0°C (high = +79.0°C, crit = +89.0°C)

drivetemp-scsi-5-0
Adapter: SCSI adapter
temp1: +32.0°C

drivetemp-scsi-3-0
Adapter: SCSI adapter
temp1: +37.0°C (low = +0.0°C, high = +60.0°C)
(crit low = -40.0°C, crit = +70.0°C)
(lowest = +24.0°C, highest = +37.0°C)

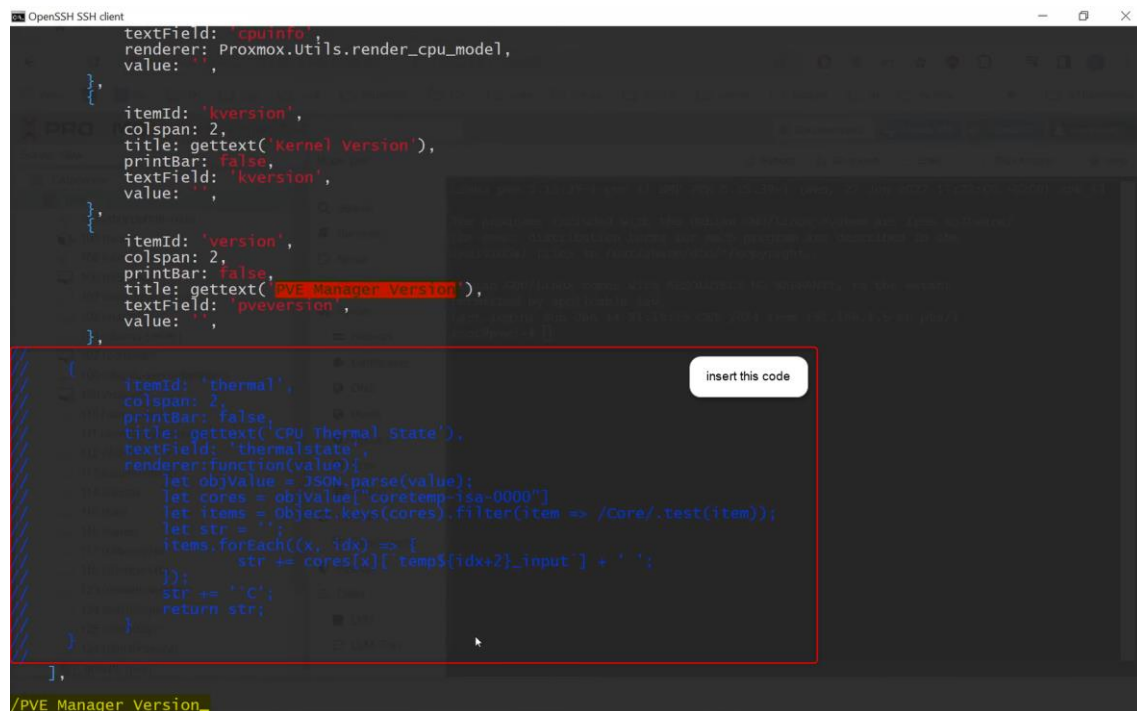
root@pve:~# vim /usr/share/pve-manager/js/pvemanagelib.js
```

You will look for the text:

“PVE Manager Version”

Add this command:

```
itemId: 'thermal',
colspan: 2,
printBar: false,
title: gettext('CPU Thermal State'),
textField: 'thermalstate',
renderer: function(value) {
    let objValue = JSON.parse(value);
    let cores = objValue["coretemp-isa-0000"];
    let items = Object.keys(cores).filter(item => /Core/.test(item));
    let str = "";
    items.forEach((x, idx) => {
        str += cores[x]['temp${idx+2}_input'] + ' ';
    });
    str += '°C';
    return str;
}
```



```
OpenSSH SSH client
textField: 'cpuinfo',
renderer: Proxmox.Utils.render_cpu_model,
value: ,
},
{
  itemId: 'kversion',
  colspan: 2,
  title: gettext('kernel version'),
  printBar: false,
  textField: 'kversion',
  value: ,
},
{
  itemId: 'version',
  colspan: 2,
  printBar: false,
  title: gettext('PVE Manager Version'),
  textField: 'pveversion',
  value: ,
},
{
  itemId: 'thermal',
  colspan: 2,
  printBar: false,
  title: gettext('CPU Thermal State'),
  textField: 'thermalstate',
  renderer: function(value) {
    let objValue = JSON.parse(value);
    let cores = objValue["coretemp-isa-0000"];
    let items = Object.keys(cores).filter(item => /Core/.test(item));
    let str = "";
    items.forEach((x, idx) => {
      str += cores[x]['temp${idx+2}_input'] + ' ';
    });
    str += '°C';
    return str;
  },
},
],
/PVE Manager Version_
```

This is basically the file that the UI (User Interface) uses to store the CPU thermal information.

Save and exit.


```
OpenSSH SSH client
},
  itemId: 'kversion',
  colspan: 2,
  title: gettext('Kernel Version'),
  printBar: false,
  textField: 'kversion',
  value: '',
},
  itemId: 'version',
  colspan: 2,
  printBar: false,
  title: gettext('PVE Manager Version'),
  textField: 'pveversion',
  value: '',
},
  itemId: 'thermal',
  colspan: 2,
  printBar: false,
  title: gettext('CPU Thermal State'),
  textField: 'thermalstate',
  renderer: function(value) {
    let objValue = JSON.parse(value);
    let cores = objValue['coretemp-isa-0000'];
    let items = Object.keys(cores).filter(item => /Core/.test(item));
    let str = '';
    items.forEach((x, idx) => {
      str += cores[x]['temp${idx+2}_input'] + ' ';
    });
    str += '°C';
    return str;
  },
},
updateTitle: function() {
  var me = this;
  var uptime = Proxmox.Utils.render_uptime(me.getRecordValue('uptime'));
  36624,5 72%
```

Next, you need to restart the PVE proxy.

systemctl restart pveproxy

```
OpenSSH SSH client
root@pve:~# apt-get install lm-sensors
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
lm-sensors is already the newest version (1:3.6.0-7).
The following package was automatically installed and is no longer required:
  libopenbsd0
Use 'apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 163 not upgraded.
root@pve:~# sensors
drivetemp-scsi-4-0
Adapter: SCSI adapter
temp1: +32.0°C

coretemp-isa-0000
Adapter: ISA adapter
Package id 0: +42.0°C (high = +79.0°C, crit = +89.0°C)
Core 0: +42.0°C (high = +79.0°C, crit = +89.0°C)
Core 1: +41.0°C (high = +79.0°C, crit = +89.0°C)
Core 2: +41.0°C (high = +79.0°C, crit = +89.0°C)
Core 3: +42.0°C (high = +79.0°C, crit = +89.0°C)
Core 4: +41.0°C (high = +79.0°C, crit = +89.0°C)
Core 5: +40.0°C (high = +79.0°C, crit = +89.0°C)
Core 6: +39.0°C (high = +79.0°C, crit = +89.0°C)
Core 7: +41.0°C (high = +79.0°C, crit = +89.0°C)

drivetemp-scsi-5-0
Adapter: SCSI adapter
temp1: +32.0°C

drivetemp-scsi-3-0
Adapter: SCSI adapter
temp1: +37.0°C (low = +0.0°C, high = +60.0°C)
(crit low = -40.0°C, crit = +70.0°C)
(lowest = +24.0°C, highest = +37.0°C)

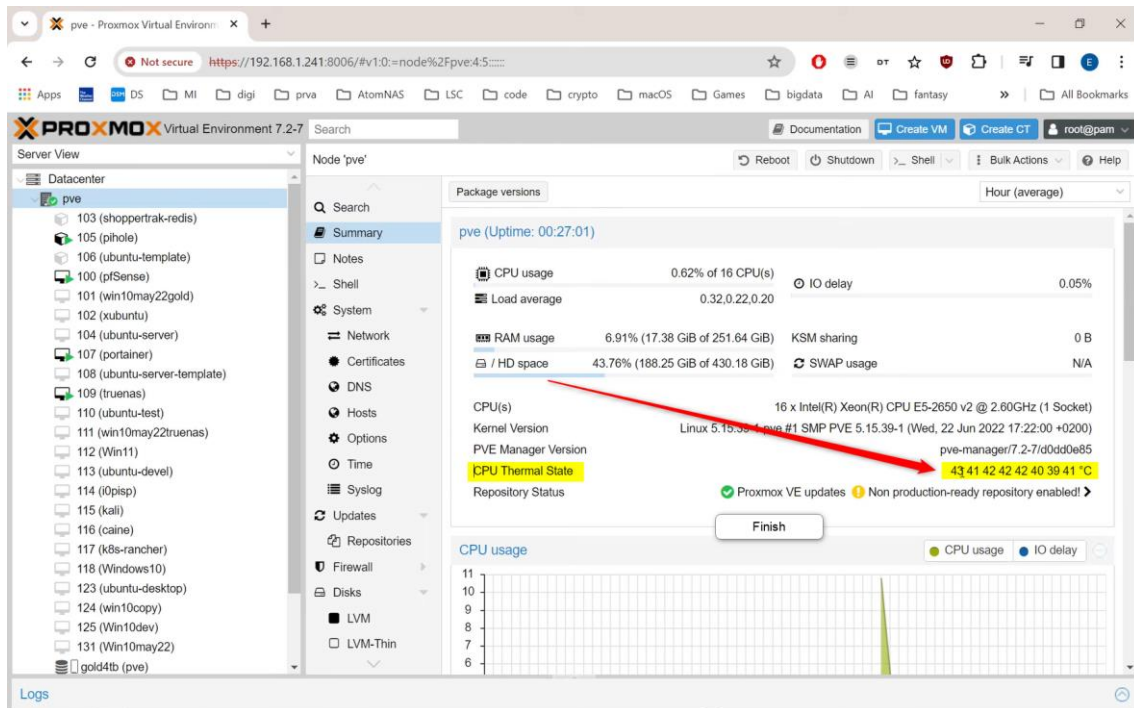
root@pve:~# vim /usr/share/perl5/PVE/API2/Nodes.pm
root@pve:~# vim /usr/share/pve-manager/js/pvmanagerlib.js
root@pve:~# systemctl restart pveproxy_

Restart the terminal,
make sure no one is using it
so as not to interrupt any task
```



Restart the terminal, make sure no one is using it so as not to interrupt any task.

Finally, you can check the CPU temperature on the main page of your Proxmox PVE.



*get the codes from the "read-me" file.