

Apache as a reverse proxy to a web application

[Take a snapshot backup of your server from Pouta Dashboard](#)

[Try out accessing the service \(11.txt not working yet\)](#)

[Start simple http server to some non standard port \(4010 here\)](#)

[Have a closer look what ports are listening and what is the process behind the port](#)

[Try out accessing the service from cli](#)

[Enable the modules needed for reverse proxy functionality](#)

[Configure apache tips.<yourhost>.ilab.fi.conf for the Python test http server](#)

[Reload apache](#)

[Test the service is available through the Apache site](#)

[Try out accessing the service from browser](#)

[Installing Node.js with Apt Using a NodeSource PPA](#)

[Create simple expressjs application and start it](#)

[Start application](#)

[Try out accessing it with curl from command line](#)

[Configure Apache to provide access to the service for url http\(s\)://tips.<yourhost>.ilab.fi/totd](#)

[Restart Apache and try out](#)

[Install PM2](#)

[Start current express.js application with pm2](#)

[Configure PM2 to start the nodejs application automatically](#)

[Boot both hosts to verify the configuration](#)

Verify this works with browser

Take a snapshot backup of your server from Pouta Dashboard

➤ before proxy Snapshot Active Private No Launch ▾

Try out accessing the service (11.txt not working yet)

← → C  tips.jingjing.ilab.fi/tips/11.txt

Not Found

The requested URL was not found on this server.

Apache/2.4.52 (Ubuntu) Server at tips.jingjing.ilab.fi Port 443

```
jingjing@jingjing-serverb-2023:~$ curl https://jingjing.ilab.fi/totd/654
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>404 Not Found</title>
</head><body>
<h1>Not Found</h1>
<p>The requested URL was not found on this server.</p>
<hr>
<address>Apache/2.4.52 (Ubuntu) Server at jingjing.ilab.fi Port 443</address>
</body></html>
```

```
jingjing@jingjing-serverb-2023:~$ curl ifconfig.me
86.50.230.95jingjing@jingjing-serverb-2023:~$
```

Start simple http server to some non standard port (4010 here)

```
[jingjing@jingjing-serverb-2023:~$ cd /opt/totd/tips
[jingjing@jingjing-serverb-2023:/opt/totd/tips$ ls
1.txt 10.txt 2.txt 3.txt 4.txt 5.txt 6.txt 7.txt 8.txt 9.txt
[jingjing@jingjing-serverb-2023:/opt/totd/tips$ python3 -m http.server 4010
Serving HTTP on 0.0.0.0 port 4010 (http://0.0.0.0:4010/) ...
^C
Keyboard interrupt received, exiting.
[jingjing@jingjing-serverb-2023:/opt/totd/tips$ python3 -m http.server 4010 &
[1] 106304
[jingjing@jingjing-serverb-2023:/opt/totd/tips$ Serving HTTP on 0.0.0.0 port 4010 (http://0.0.0.0:4010/) ...
[jingjing@jingjing-serverb-2023:/opt/totd/tips$ fg
python3 -m http.server 4010

^C
Keyboard interrupt received, exiting.
```

Have a closer look what ports are listening and what is the process behind the port

```
[jingjing@jingjing-serverb-2023:~$ sudo apt install net-tools
[sudo] password for jingjing:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
net-tools is already the newest version (1.60+git20181103.0eebece-1ubuntu5).
0 upgraded, 0 newly installed, 0 to remove and 1 not upgraded.
[jingjing@jingjing-serverb-2023:~$ netstat -anp | grep "LISTEN "
(Not all processes could be identified, non-owned process info
will not be shown, you would have to be root to see it all.)
tcp      0      0 0.0.0.0:4010          0.0.0.0:*          LISTEN      107124/python3
tcp      0      0 127.0.0.53:53        0.0.0.0:*          LISTEN      -
tcp      0      0 0.0.0.0:22          0.0.0.0:*          LISTEN      -
tcp6     0      0 ::::443           ::::*               LISTEN      -
tcp6     0      0 ::::22            ::::*               LISTEN      -
tcp6     0      0 ::::80            ::::*               LISTEN      -
```

Try out accessing the service from cli

```
[jingjing@jingjing-serverb-2023:/opt/totd/tips$ python3 -m http.server 4010
Serving HTTP on 0.0.0.0 port 4010 (http://0.0.0.0:4010/) ...
127.0.0.1 - - [16/Apr/2023 08:16:52] "GET / HTTP/1.1" 200 -
```



jingjinyang - j

```
[jingjing@jingjing-serverb-2023:~$ curl localhost:4010
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Directory listing for /</title>
</head>
<body>
<h1>Directory listing for /</h1>
<hr>
<ul>
<li><a href="1.txt">1.txt</a></li>
<li><a href="10.txt">10.txt</a></li>
<li><a href="2.txt">2.txt</a></li>
<li><a href="3.txt">3.txt</a></li>
<li><a href="4.txt">4.txt</a></li>
<li><a href="5.txt">5.txt</a></li>
<li><a href="6.txt">6.txt</a></li>
<li><a href="7.txt">7.txt</a></li>
<li><a href="8.txt">8.txt</a></li>
<li><a href="9.txt">9.txt</a></li>
</ul>
<hr>
</body>
</html>
```



```
[jingjing@jingjing-serverb-2023:/opt/totd/tips$ python3 -m http.server 4010
Serving HTTP on 0.0.0.0 port 4010 (http://0.0.0.0:4010/) ...
127.0.0.1 - - [16/Apr/2023 08:16:52] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [16/Apr/2023 08:17:41] "GET /3.txt HTTP/1.1" 200 -

```



```
[jingjing@jingjing-serverb-2023:~$ curl localhost:4010/3.txt
`tar'
-c : Creates Archive
-x : Extract the archive
-f : creates archive with given filename
-t : displays or lists files in archived file
-v : Displays Verbose Information
-z : zip, tells tar command that creates tar file using gzip
-j : filter archive tar file using tbzip
tar -zcvf ~/package.tar.gz tmp/
tar -jxvf package.tar.bz2
```

Enable the modules needed for reverse proxy functionality

```
[jingjing@jingjing-serverb-2023:~$ cd /etc/apache2/sites-available/
[jingjing@jingjing-serverb-2023:/etc/apache2/sites-available$ sudo a2enmod proxy
[[sudo] password for jingjing:
Enabling module proxy.
To activate the new configuration, you need to run:
    systemctl restart apache2
[jingjing@jingjing-serverb-2023:/etc/apache2/sites-available$ sudo a2enmod proxy_http
Considering dependency proxy for proxy_http:
Module proxy already enabled
Enabling module proxy_http.
To activate the new configuration, you need to run:
    systemctl restart apache2
[jingjing@jingjing-serverb-2023:/etc/apache2/sites-available$ cd ..
[jingjing@jingjing-serverb-2023:/etc/apache2$ cd sites-enabled/
```

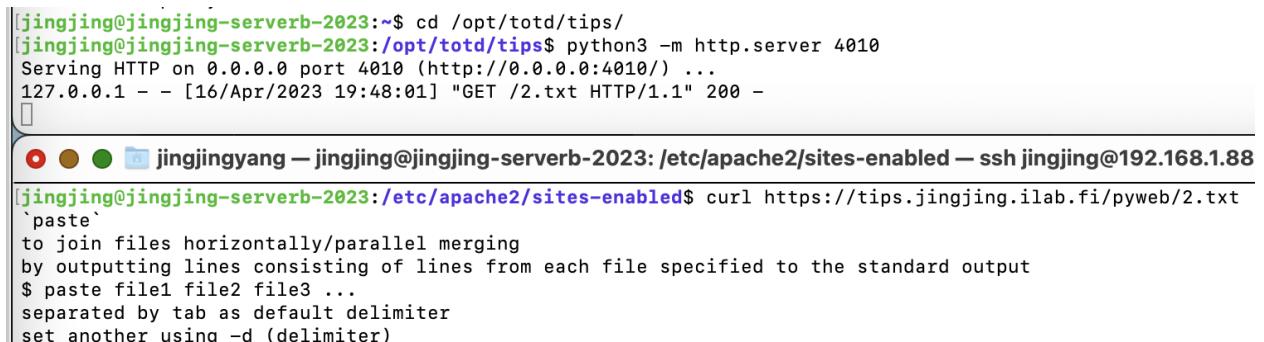
Configure apache tips.<yourhost>.ilab.fi.conf for the Python test http server

```
[jingjing@jingjing-serverb-2023:/etc/apache2/sites-enabled$ sudo nano tips.jingjing.ilab.fi-le-ssl.conf
[jingjing@jingjing-serverb-2023:/etc/apache2/sites-enabled$ cat tips.jingjing.ilab.fi-le-ssl.conf | grep "Proxy"
    ProxyPass "/pyweb" "http://localhost:4010"
    ProxyPassReverse "/pyweb" "http://localhost:4010" _
```

Reload apache

```
[jingjing@jingjing-serverb-2023:/etc/apache2/sites-enabled$ sudo systemctl reload apache2.service
[[sudo] password for jingjing:
```

Test the service is available through the Apache site



The screenshot shows a terminal window with the following content:

```
[jingjing@jingjing-serverb-2023:~$ cd /opt/totd/tips/
[jingjing@jingjing-serverb-2023:/opt/totd/tips$ python3 -m http.server 4010
Serving HTTP on 0.0.0.0 port 4010 (http://0.0.0.0:4010/) ...
127.0.0.1 - - [16/Apr/2023 19:48:01] "GET /2.txt HTTP/1.1" 200 -
[redacted]
● ● ● 📡 jingjingyang — jingjing@jingjing-serverb-2023: /etc/apache2/sites-enabled — ssh jingjing@192.168.1.88
[jingjing@jingjing-serverb-2023:/etc/apache2/sites-enabled$ curl https://tips.jingjing.ilab.fi/pyweb/2.txt
`paste'
to join files horizontally/parallel merging
by outputting lines consisting of lines from each file specified to the standard output
$ paste file1 file2 file3 ...
separated by tab as default delimiter
set another using -d (delimiter)
```

```
[jingjing@jingjing-serverb-2023:~$ cd /opt/totd/tips/  
[jingjing@jingjing-serverb-2023:/opt/totd/tips$ python3 -m http.server 4010  
Serving HTTP on 0.0.0.0 port 4010 (http://0.0.0.0:4010/) ...  
127.0.0.1 - - [16/Apr/2023 19:48:01] "GET /2.txt HTTP/1.1" 200 -  
127.0.0.1 - - [16/Apr/2023 19:48:37] "GET / HTTP/1.1" 200 -  
□  
● ● ● jingjinyang — jingjing@jingjing-serverb-2023: /etc/apache2/sites-enabled — ssh jingjing@192.1  
[jingjing@jingjing-serverb-2023:/etc/apache2/sites-enabled$ curl https://tips.jingjing.ilab.fi/pyweb  
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">  
<html>  
<head>  
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">  
<title>Directory listing for /</title>  
</head>  
<body>  
<h1>Directory listing for /</h1>  
<hr>  
<ul>  
<li><a href="1.txt">1.txt</a></li>  
<li><a href="10.txt">10.txt</a></li>  
<li><a href="2.txt">2.txt</a></li>  
<li><a href="3.txt">3.txt</a></li>  
<li><a href="4.txt">4.txt</a></li>  
<li><a href="5.txt">5.txt</a></li>  
<li><a href="6.txt">6.txt</a></li>  
<li><a href="7.txt">7.txt</a></li>  
<li><a href="8.txt">8.txt</a></li>  
<li><a href="9.txt">9.txt</a></li>  
</ul>  
<hr>  
</body>  
</html>
```

Try out accessing the service from browser



Directory listing for /

- [1.txt](#)
- [10.txt](#)
- [2.txt](#)
- [3.txt](#)
- [4.txt](#)
- [5.txt](#)
- [6.txt](#)
- [7.txt](#)
- [8.txt](#)
- [9.txt](#)

```
`paste`  
to join files horizontally/parallel merging  
by outputting lines consisting of lines from each file specified to the standard output  
$ paste file1 file2 file3 ...  
separated by tab as default delimiter  
set another using -d (delimiter)
```

Installing Node.js with Apt Using a NodeSource PPA

```
jingjing@jingjing-serverb:2023:/etc/apache2/sites-enabled$ cd
jingjing@jingjing-serverb:2023:~$ curl -sL https://deb.nodesource.com/setup_18.x -o nodesource_setup.sh
jingjing@jingjing-serverb:2023:~$ sudo bash nodesource_setup.sh
[sudo] password for jingjing:

## Installing the NodeSource Node.js 18.x repo...

## Populating apt-get cache...

+ apt-get update
Hit:1 http://nova.clouds.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://nova.clouds.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:3 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:4 http://nova.clouds.archive.ubuntu.com/ubuntu jammy-backports InRelease [108 kB]
Fetched 337 kB in 1s (380 kB/s)
Reading package lists... Done

## Confirming "jammy" is supported...

+ curl -sLf -o /dev/null 'https://deb.nodesource.com/node_18.x/dists/jammy/Release'
## Adding the NodeSource signing key to your keyring...

+ curl -s https://deb.nodesource.com/gpgkey/nodesource.gpg.key | gpg --dearmor | tee /usr/share/keyrings/nodesource.gpg >/dev/null
## Creating apt sources list file for the NodeSource Node.js 18.x repo...

+ echo 'deb [signed-by=/usr/share/keyrings/nodesource.gpg] https://deb.nodesource.com/node_18.x jammy main' > /etc/apt/sources.list.d/nodesource.list
+ echo 'deb-src [signed-by=/usr/share/keyrings/nodesource.gpg] https://deb.nodesource.com/node_18.x jammy main' >> /etc/apt/sources.list.d/nodesource.list

## Running `apt-get update` for you...

+ apt-get update
Hit:1 http://nova.clouds.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 https://deb.nodesource.com/node_18.x jammy InRelease [4563 B]
Get:3 http://nova.clouds.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:4 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:5 http://nova.clouds.archive.ubuntu.com/ubuntu jammy-backports InRelease [108 kB]
Get:6 https://deb.nodesource.com/node_18.x jammy/main amd64 Packages [776 B]
Fetched 342 kB in 1s (355 kB/s)
Reading package lists... Done

## Run `sudo apt-get install -y nodejs` to install Node.js 18.x and npm
## You may also need development tools to build native addons:
  sudo apt-get install gcc g++ make
## To install the Yarn package manager, run:
  curl -sL https://dl.yarnpkg.com/debian/pubkey.gpg | gpg --dearmor | sudo tee /usr/share/keyrings/yarnkey.gpg >/dev/null
  echo "deb [signed-by=/usr/share/keyrings/yarnkey.gpg] https://dl.yarnpkg.com/debian stable main" | sudo tee /etc/apt/sources.list.d/yarn.list
  sudo apt-get update && sudo apt-get install yarn
```

```
jingjing@jingjing-serverb-2023:~$ sudo apt install nodejs
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
nodejs
0 upgraded, 1 newly installed, 0 to remove and 1 not upgraded.
Need to get 28.7 MB of archives.
After this operation, 187 MB of additional disk space will be used.
Get:1 https://deb.nodesource.com/node_18.x jammy/main amd64 nodejs amd64 18.16.0-deb-1nodesource1 [28.7 MB]
Fetched 28.7 MB in 1s (44.4 MB/s)
Selecting previously unselected package nodejs.
(Reading database ... 65941 files and directories currently installed.)
Preparing to unpack .../nodejs_18.16.0-deb-1nodesource1_amd64.deb ...
Unpacking nodejs (18.16.0-deb-1nodesource1) ...
Setting up nodejs (18.16.0-deb-1nodesource1) ...
Processing triggers for man-db (2.10.2-1) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
jingjing@jingjing-serverb-2023:~$ node -v
v18.16.0
```

Create simple expressjs application and start it

```
[jingjing@jingjing-serverb-2023:~$ git clone https://gitlab.tamk.cloud/sweng-groups/tla-express-tips.git
Cloning into 'tla-express-tips'...
remote: Enumerating objects: 64, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 64 (delta 0), reused 0 (delta 0), pack-reused 60
Receiving objects: 100% (64/64), 34.89 KiB | 1.16 MiB/s, done.
Resolving deltas: 100% (18/18), done.
[jingjing@jingjing-serverb-2023:~$ cd tla-express-tips/nodeapp/
[jingjing@jingjing-serverb-2023:~/tla-express-tips/nodeapp$ npm install
added 89 packages, and audited 90 packages in 2s

10 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
npm notice New minor version of npm available! 9.5.1 -> 9.6.4
npm notice Changelog: https://github.com/npm/cli/releases/tag/v9.6.4
npm notice Run npm install -g npm@9.6.4 to update!
npm notice
```

Start application

```
[jingjing@jingjing-serverb-2023:~/tla-express-tips/nodeapp$ node app.js
Example app listening at http://localhost:3000
[
  'chage.txt',
  'date.txt',
  'find.txt',
  'for-loop.txt',
  'gpasswd.txt',
  'lvm.txt',
  'lvm2.txt',
  'useradd.txt',
  'wc.txt',
  'wget.txt'
]
File in directory: chage.txt
File in directory: date.txt
File in directory: find.txt
File in directory: for-loop.txt
File in directory: gpasswd.txt
File in directory: lvm.txt
File in directory: lvm2.txt
File in directory: useradd.txt
File in directory: wc.txt
File in directory: wget.txt
Number of files is 10
```

```
○ ● ● jingjingyang — jingjing@jingjing-serverb-2023: ~ — ssh jingjing@192.168.1.88 —
[jingjing@jingjing-serverb-2023:~$ netstat -anp | grep "LISTEN "
(Not all processes could be identified, non-owned process info
 will not be shown, you would have to be root to see it all.)
tcp      0      0 127.0.0.53:53          0.0.0.0:*
tcp      0      0 0.0.0.0:22          0.0.0.0:*
tcp6     0      0 :::443              :::*
tcp6     0      0 :::3000             :::*
tcp6     0      0 :::22              :::*
tcp6     0      0 :::80              :::*
```

```
[jingjing@jingjing-serverb-2023:~$ ps -ef | grep 142086
jingjing 142086 126155  0 03:03 pts/0    00:00:00 node app.js
jingjing 142249 127776  0 03:07 pts/1    00:00:00 grep --color=auto 142086
```

Try out accessing it with curl from command line

```
jingjing@jingjing-serverb-2023:~/tla-express-tips/nodeapp$ node app.js
Example app listening at http://localhost:3000
[
  'chage.txt',
  'date.txt',
  'find.txt',
  'for-loop.txt',
  'gpasswd.txt',
  'lvm.txt',
  'lvm2.txt',
  'useradd.txt',
  'wc.txt',
  'wget.txt'
]
File in directory: chage.txt
File in directory: date.txt
File in directory: find.txt
File in directory: for-loop.txt
File in directory: gpasswd.txt
File in directory: lvm.txt
File in directory: lvm2.txt
File in directory: useradd.txt
File in directory: wc.txt
File in directory: wget.txt
Number of files is 10
Param 1 mods to 1 and the file to read is: date.txt
The file requested is date.txt
The path to read is ../tips/linux/date.txt

```

● ● ● jingjingyang — jingjing@jingjing-serverb-2023: ~ — ssh jingjing@192.168.1.88 — 116x36

```
jingjing@jingjing-serverb-2023:~$ curl localhost:3000
Hello Express! Use the /totd/<nr> to fetch tipsjingjing@jingjing-serverb-2023:~$ curl localhost:3000/totd/1
Command: date

--- With date command it is easy to format a string with date and time information.
$ date "+%Y:%m:%d-%H%M%S-DATEDIR"

--- This can be used as part of a variable as well:
$ export MYVAR="HALOO--`date "+%F-%T-DATEDIR"`
$ echo $MYVAR

(Author: Petteri)
```

```
Param 31492 mods to 2 and the file to read is: find.txt
The file requested isfind.txt
The path to read is../tips/linux/find.txt
Param 23115 mods to 5 and the file to read is: lvm.txt
The file requested islvm.txt
The path to read is../tips/linux/lvm.txt
Param 27714 mods to 4 and the file to read is: gpasswd.txt
The file requested isgpasswd.txt
The path to read is../tips/linux/gpasswd.txt
```

█

● ● ● jingjingyang — jingjing@jingjing-serverb-2023: ~ — ssh jingjing...

Find files from current location (.) with precise name. For each hit, run a command – this time grep from the found file ({}) – end the command with semicolon ... escape it to make part of parameter

```
$ find . -name "wisdom.txt" -exec grep -i vote {} \;
```

jingjing@jingjing-serverb-2023:~\$ curl localhost:3000/totd/\$RANDOM
Topic: LVM – Logical Volume Management

LVM is a technique for creating dynamically expandable storage volumes. A mounted volume can be expanded on the fly with no need for restarting services using it.

Logically steps for creating a LVM device are:

- Create a (logical) volume group and add physical devices to it.
- Create a logical partition using the volume group
- Make a file system (like ext4) to the partition and mount it
- Afterwards the logical partition can be expanded to the limit of free space in volume group and volume group by adding physical devices

Steps to create a brand new LVM device:

```
# lsblk (diagnose)
# pvcreate /dev/xvdc /dev/xvde      (not needed necessarily)
# vgcreate vg200 /dev/xvdc /dev/xvde (Create the pool)
# lvcreate -n vision200 -L 2G vg200   (Create a mountable partition)
# mkfs.xfs /dev/vg200/vision200     (Make filesystem)
# mount /dev/vg200/vision200 /vision200
```

jingjing@jingjing-serverb-2023:~\$ curl localhost:3000/totd/\$RANDOM
Topic: gpasswd

With gpasswd command a user can be promoted to a group admin.
Set user u1 as admin to group engineers:

```
$ gpasswd -A u1 engineers
```

Demote user u2 from engineers group admins:

```
$ gpasswd -d u2 engineers
```

See the admins of the group:

```
$ grep engineers /etc/gshadow
```

Configure Apache to provide access to the service for url http(s)://tips.<yourhost>.ilab.fi/totd

```
[jingjing@jingjing-serverb-2023:~$ sudo nano /etc/apache2/sites-enabled/tips.jingjing.ilab.fi-le-ssl.conf
[sudo] password for jingjing:
[jingjing@jingjing-serverb-2023:~$ cat /etc/apache2/sites-enabled/tips.jingjing.ilab.fi-le-ssl.conf | grep "Proxy"
# ProxyPass "/pyweb" "http://localhost:4010"
# ProxyPassReverse "/pyweb" "http://localhost:4010"
ProxyPass "/totd" "http://localhost:3000/totd"
ProxyPassReverse "/totd" "http://localhost:3000/totd"
```

Restart Apache and try out

```
[jingjing@jingjing-serverb-2023:~$ sudo systemctl reload apache2.service
[jingjing@jingjing-serverb-2023:~$ curl http://tips.jingjing.ilab.fi/totd/1
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>301 Moved Permanently</title>
</head><body>
<h1>Moved Permanently</h1>
<p>The document has moved <a href="https://tips.jingjing.ilab.fi/totd/1">here</a>.</p>
<hr>
<address>Apache/2.4.52 (Ubuntu) Server at tips.jingjing.ilab.fi Port 80</address>
</body></html>
[jingjing@jingjing-serverb-2023:~$ curl https://tips.jingjing.ilab.fi/totd/1
Command: date

--- With date command it is easy to format a string with date and time information.
$ date "+%Y:%m:%d-%H%M%S-DATEDIR"

--- This can be used as part of a variable as well:
$ export MYVAR="HALOO-`date "+%F-%T-DATEDIR"`
$ echo $MYVAR

(Author: Petteri)

[jingjing@jingjing-serverb-2023:~$ curl https://tips.jingjing.ilab.fi/totd/$RANDOM
Topic: gpasswd

With gpasswd command a user can be promoted to a group admin.
Set user u1 as admin to group engineers:

$ gpasswd -A u1 engineers

Demote user u2 from engineers group admins:

$ gpasswd -d u2 engineers

See the admins of the group:

$ grep engineers /etc/gshadow
```

Install PM2

```

jingjing@jingjing-serverb-2023:~$ systemctl status apache2.service
● apache2.service - The Apache HTTP Server
  Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
  Active: active (running) since Sun 2023-04-16 12:26:43 UTC; 15h ago
    Docs: https://httpd.apache.org/docs/2.4/
   Process: 114661 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
  Process: 143144 ExecReload=/usr/sbin/apachectl graceful (code=exited, status=0/SUCCESS)
 Main PID: 114666 (apache2)
    Tasks: 55 (limit: 1086)
   Memory: 10.2M
      CPU: 5.741s
     CGroup: /system.slice/apache2.service
             └─114666 /usr/sbin/apache2 -k start
                 ├─143148 /usr/sbin/apache2 -k start
                 ├─143149 /usr/sbin/apache2 -k start
                 └─143149 /usr/sbin/apache2 -k start
jingjing@jingjing-serverb-2023:~$ sudo npm i pm2 -g
npm WARN deprecated uid@3.4.0: Please upgrade to version 7 or higher. Older versions may use Math.random()
() in certain circumstances, which is known to be problematic. See https://v8.dev/blog/math-random for details.

added 184 packages in 19s

12 packages are looking for funding
  run `npm fund` for details
npm notice
npm notice New minor version of npm available! 9.5.1 → 9.6.4
npm notice Changelog: https://github.com/npm/cli/releases/tag/v9.6.4
npm notice Run npm install -g npm@9.6.4 to update!
npm notice

```

Start current express.js application with pm2

```

[jingjing@jingjing-serverb-2023:~$ cd tla-express-tips/nodeapp/
[jingjing@jingjing-serverb-2023:~/tla-express-tips/nodeapp$ pm2 start app.js
[PM2] Starting /home/jingjing/tla-express-tips/nodeapp/app.js in fork_mode (1 instance)
[PM2] Done.

```

id	name	mode	�	status	cpu	memory
0	app	fork	0	online	0%	45.2mb

Configure PM2 to start the nodejs application automatically


```
[PM2] Init System found: systemd
Platform systemd
Template
[Unit]
Description=PM2 process manager
Documentation=https://pm2.keymetrics.io/
After=network.target

[Service]
Type=forking
User=jingjing
LimitNOFILE=infinity
LimitNPROC=infinity
LimitCORE=infinity
Environment=PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin:/usr/bin:/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin
Environment=PM2_HOME=/home/jingjing/.pm2
PIDFile=/home/jingjing/.pm2/pm2.pid
Restart=on-failure

ExecStart=/usr/lib/node_modules/pm2/bin/pm2 resurrect
ExecReload=/usr/lib/node_modules/pm2/bin/pm2 reload all
ExecStop=/usr/lib/node_modules/pm2/bin/pm2 kill

[Install]
WantedBy=multi-user.target

Target path
/etc/systemd/system/pm2-jingjing.service
Command list
[ 'systemctl enable pm2-jingjing' ]
[PM2] Writing init configuration in /etc/systemd/system/pm2-jingjing.service
[PM2] Making script booting at startup...
[PM2] [-] Executing: systemctl enable pm2-jingjing...
Created symlink /etc/systemd/system/multi-user.target.wants/pm2-jingjing.service → /etc/systemd/system/pm2-jingjing.service.
[PM2] [v] Command successfully executed.
+-----+
[PM2] Freeze a process list on reboot via:
$ pm2 save

[[PM2] Remove init script via:
$ pm2 unstartup systemd
jingjing@jingjing-serververb-2023:~/tla-express-tips/nodeapp$ pm2 save
[PM2] Saving current process list...
[PM2] Successfully saved in /home/jingjing/.pm2/dump.pm2
jingjing@jingjing-serververb-2023:~/tla-express-tips/nodeapp$ sudo systemctl status pm2-jingjing
● pm2-jingjing.service - PM2 process manager
   Loaded: loaded (/etc/systemd/system/pm2-jingjing.service; enabled; vendor preset: enabled)
   Active: inactive (dead)
     Docs: https://pm2.keymetrics.io/
```

```

[... https://pm2.keymetrics.io]
[jingjing@jingjing-serverb-2023:~/tla-express-tips/nodeapp$ pm2 kill
[PM2] Applying action deleteProcessId on app [all](ids: [ 0 ])
[PM2] [all](0) ✓
[PM2] [v] All Applications Stopped
[PM2] [v] PM2 Daemon Stopped
[jingjing@jingjing-serverb-2023:~/tla-express-tips/nodeapp$ sudo systemctl status pm2-jingjing
● pm2-jingjing.service - PM2 process manager
  Loaded: loaded (/etc/systemd/system/pm2-jingjing.service; enabled; vendor preset: enabled)
    Active: inactive (dead)
      Docs: https://pm2.keymetrics.io/
[jingjing@jingjing-serverb-2023:~/tla-express-tips/nodeapp$ sudo systemctl daemon-reload
[jingjing@jingjing-serverb-2023:~/tla-express-tips/nodeapp$ sudo systemctl status pm2-jingjing
● pm2-jingjing.service - PM2 process manager
  Loaded: loaded (/etc/systemd/system/pm2-jingjing.service; enabled; vendor preset: enabled)
    Active: inactive (dead)
      Docs: https://pm2.keymetrics.io/
[jingjing@jingjing-serverb-2023:~/tla-express-tips/nodeapp$ sudo systemctl start pm2-jingjing
[jingjing@jingjing-serverb-2023:~/tla-express-tips/nodeapp$ sudo systemctl status pm2-jingjing
● pm2-jingjing.service - PM2 process manager
  Loaded: loaded (/etc/systemd/system/pm2-jingjing.service; enabled; vendor preset: enabled)
    Active: active (running) since Mon 2023-04-17 03:55:35 UTC; 3s ago
      Docs: https://pm2.keymetrics.io/
  Process: 144213 ExecStart=/usr/lib/node_modules/pm2/bin/pm2 resurrect (code=exited, status=0/SUCCESS)
 Main PID: 144220 (PM2 v5.3.0: God)
 Tasks: 22 (limit: 1086)
  Memory: 47.3M
    CPU: 4.018s
   CGroup: /system.slice/pm2-jingjing.service
           ├─144220 "PM2 v5.3.0: God Daemon (/home/jingjing/.pm2)"
           └─144370 "node /home/jingjing/tla-express-tips/nodeapp/app.js" "" "" "" "" ""

Apr 17 03:55:35 jingjing-serverb-2023 pm2[144213]: [PM2] PM2 Successfully daemonized
Apr 17 03:55:35 jingjing-serverb-2023 pm2[144213]: [PM2] Resurrecting
Apr 17 03:55:35 jingjing-serverb-2023 pm2[144213]: [PM2] Restoring processes located in /home/jingjing/.pm2/dump.pm2
Apr 17 03:55:35 jingjing-serverb-2023 pm2[144213]: [PM2] Process /home/jingjing/tla-express-tips/nodeapp/app.js restored
Apr 17 03:55:35 jingjing-serverb-2023 pm2[144213]:
Apr 17 03:55:35 jingjing-serverb-2023 pm2[144213]: id name namespace version mode pid uptime
Apr 17 03:55:35 jingjing-serverb-2023 pm2[144213]: 0 app default 1.0.0 fork 144231 0s
Apr 17 03:55:35 jingjing-serverb-2023 pm2[144213]:
Apr 17 03:55:35 jingjing-serverb-2023 systemd[1]: Started PM2 process manager.
lines 1-23/23 (END)

```

Boot both hosts to verify the configuration

```

[[...]]
[jingjing@jingjing-serverb-2023:~$ sudo shutdown -r now
[[sudo]] password for jingjing:
[jingjing@jingjing-serverb-2023:~$ Connection to jingjing.ilab.fi closed by remote host.
Connection to jingjing.ilab.fi closed.
[jingjing@Ubuntu-MacBookPro:~$ ]

```

```

[jingjing@jingjing-serverb-2023:~$ while true; do date; ls; sleep 1; done
Mon Apr 17 04:08:40 UTC 2023
nodesource_setup.sh tla-express-tips
Mon Apr 17 04:08:41 UTC 2023
nodesource_setup.sh tla-express-tips
Mon Apr 17 04:08:42 UTC 2023
nodesource_setup.sh tla-express-tips
Mon Apr 17 04:08:43 UTC 2023
nodesource_setup.sh tla-express-tips
^C
[jingjing@jingjing-serverb-2023:~$ curl localhost:3000/totd/$RANDOM
Topic: find
Find files from current location (.) with precise name. For each hit, run a command - this time grep from the found file
{} - end the command with semicolon ... escape it to make part of parameter
$ find . -name "wisdom.txt" -exec grep -i vote {} \;

```

Verify this works with browser

