

## Docker

Docker คือ engine ตัวหนึ่งที่มีการทำงานในลักษณะจำลองสภาพแวดล้อมขึ้นมาบนเครื่อง server เพื่อใช้ในการ run service ที่ต้องการ มีการทำงานคล้ายคลึงกับ Virtual Machine เช่น VMWare, VirtualBox, XEN, KVM แต่ข้อแตกต่างที่ชัดเจนคือ Virtual Machine ที่รู้จักกันก่อนหน้านี้ นั้น เป็นการจำลองทั้ง OS เพื่อใช้งานและหากต้องการใช้งาน service ใดๆ จึงทำการติดตั้งเพิ่มเติมบน OS นั้นๆ แต่สำหรับ docker แล้วจะใช้ container ในการจำลองสภาพแวดล้อมขึ้นมา เพื่อใช้งานสำหรับ 1 service ที่ต้องการใช้งานเท่านั้น โดยไม่ต้องมีส่วนของ OS เข้าไปเกี่ยวข้องเหมือน Virtual Machines อื่นๆ Docker นั้น เป็นที่รู้จักกันอย่างแพร่หลายในช่วง 1-2 ปีที่ผ่านมา เนื่องจากสามารถใช้งานได้อย่างสะดวกและตอบสนองความต้องการของ ผู้พัฒนาโปรแกรม (Developer) หรือ ผู้ดูแลระบบ (System admin)

## Docker image

Docker image เป็นเหมือนตัวต้นแบบของ container ซึ่งภายในจะประกอบด้วย application ต่างๆ ที่มีการติดตั้งไว้เพื่อใช้งานสำหรับ service นั้นๆ รวมทั้งมีการ config ค่าต่างๆ ไว้เรียบร้อยแล้ว จากนั้นก็นำมาสร้างเป็น docker image บน registry เพื่อนำใช้งาน ทั้งนี้ผู้ใช้งานสามารถยังสร้าง docker image สำหรับใช้งานเองได้อีกด้วย

## Docker container

Docker container สามารถมองได้เสมือนกล่อง ซึ่งนำ docker image มาติดตั้ง เพื่อให้สามารถใช้งาน service ที่ต้องการจาก image นั้นๆ ได้ โดยใน container แต่ละตัวจะมีการใช้งาน RAM, CPU, ไฟล์ config ต่างๆ เป็นของแต่ละ container เอง และยังสามารถสั่ง start, stop ได้ที่ container นั้นๆ อีกด้วย

## ความน่าสนใจของ docker

Docker engine สามารถใช้งานได้บนหลาย platform ทั้งบน Linux, Mac และ Windows

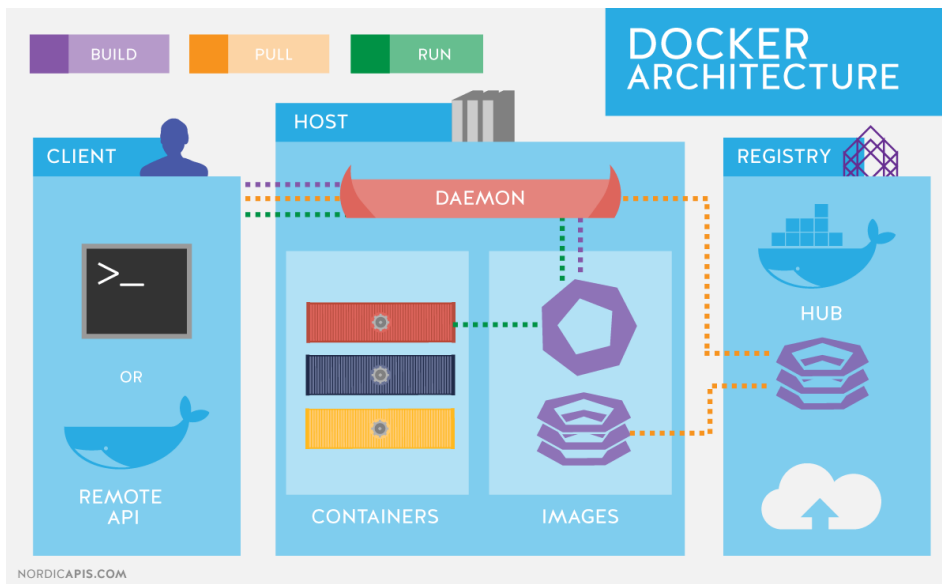
Docker มีขนาดเล็ก สามารถใช้งาน และติดตั้งได้อย่างรวดเร็ว และสะดวกในการ start / stop หรือแม้แต่การย้ายไปใช้งานสำหรับเครื่อง server อื่นที่มีการ run docker engine ก็ยังสามารถทำได้โดยไม่ซับซ้อน

ผู้ใช้งาน docker ไม่จำเป็นต้องติดตั้ง OS อีกครั้งเพื่อติดตั้ง container รวมทั้งไม่จำเป็นต้อง config เพิ่มเติมในส่วนที่ไม่จำเป็นอีกด้วย

Docker มีความต้องการในการใช้ CPU, RAM และพื้นที่น้อยกว่า Virtual Machine ทั้งนี้ในทรัพยากรที่มีเท่ากัน docker สามารถใช้งาน container ได้มากกว่า Virtual Machine

เนื่องจากผู้ใช้งาน สามารถสร้าง docker image ได้เอง จาก dockerfile ดังนั้นการใช้งาน docker ยังช่วยลด ปัญหาสภาพแวดล้อมที่ต่างกัน ที่มักพบเมื่อบาง application สามารถทำงานได้บน development server แต่ไม่สามารถใช้งานบน production server ได้

Docker ยังมี docker registry ซึ่งผู้ใช้งานสามารถเลือก pull image ต่างๆ ที่มีการสร้างไว้ให้แล้วมาใช้งาน โดยมี Docker Hub เป็น registry หลักในการเรียกใช้ image



## APACHE WEB SERVER

Apache Web Server ( httpd ) หรือ Apache HyperText Transfer Protocol (HTTP) คือ ซอฟต์แวร์สำหรับเปิดให้บริการ Server บนโพรโทคอล HTTP โดยสามารถทำงานได้บนหลายระบบปฏิบัติการ เช่น Linux Server Windows Server และอื่นๆ

ที่มาของชื่อ Apache มาจากกลุ่มคนที่ช่วยสร้างแพตช์ไฟล์สำหรับโครงการ NCSA httpd 1.3 ซึ่งกลายมาเป็นที่มาของชื่อ A PAChy server และในอีกความหมายหนึ่งยังกล่าวถึงเผ่าอะแพชีหรืออาปาเช่ ซึ่งเป็นเผ่าอินเดียนแดงที่มีความสามารถในการรบสูง

การที่อาปาเช่เป็นซอฟต์แวร์ที่อยู่ในลักษณะของ open source ที่เปิดให้บุคคลทั่วไปสามารถเข้ามาร่วมพัฒนาส่วนต่างๆ ของอาปาเช่ได้ ซึ่งทำให้เกิดเป็น โมดูล ที่เกิดประโยชน์มากมาย เช่น mod\_perl, mod\_python หรือ mod\_php ซึ่งเป็นโมดูลที่ทำให้อาปาเช่สามารถใช้ประโยชน์ และทำงานร่วมกับภาษาอื่นได้ แทนที่จะเป็นเพียงเซิร์ฟเวอร์ที่ให้บริการเพียงแค่อะแดปเตอร์ อย่างเดียว นอกจากนี้อาปาเช่เองยังมีความสามารถอื่นๆ ด้วย เช่น การยืนยันตัวตนบุคคล (mod\_auth, mod\_access, mod\_digest) หรือเพิ่มความปลอดภัยในการสื่อสารผ่านโพรโทคอล https (mod\_ssl) นอกจากนี้ ก็ยังมีโมดูลอื่นๆ ที่ได้รับความนิยมใช้ เช่น mod\_vhost ทำให้สามารถสร้างโฮสต์เสมือน www.sample.com, wiki.sample.com, mail.sample.com หรือ www.ilovewiki.org ภายในเครื่องเดียวกันได้ หรือ mod\_rewrite เป็นเครื่องมือที่จะช่วยให้ url ของเว็บนั้นอ่านง่ายขึ้น ยกตัวอย่างเช่น จากเดิมต้องอ้างถึงเว็บไซต์แห่งหนึ่งด้วยการ

พิมพ์ `http://www.yourdomain.com/board/question.php?action=viewtopic&qid=2xDffw` แต่หลังจากใช้ mod\_rewrite จะทำให้สั้นลง กลายเป็น `http://www.yourdomain.com/board/question/2xDffw` ซึ่งที่อยู่หลังนี้จะขึ้นอยู่กับว่าผู้ดูแลเว็บไซต์ต้องการให้อยู่ในลักษณะใด



## phpMyAdmin

phpMyAdmin คือโปรแกรมที่ถูกพัฒนาโดยใช้ภาษา [PHP](#) เพื่อใช้ในการบริหารจัดการฐานข้อมูล MySQL แทนการคีย์คำสั่ง เนื่องจากถ้าเราจะใช้ฐานข้อมูลที่เป็น MySQL บางครั้งจะมีความลำบากและยุ่งยากในการทำงาน ดังนั้นจึงมีเครื่องมือในการจัดการฐานข้อมูล MySQL ขึ้นมาเพื่อให้สามารถจัดการ ตัว [DBMS](#) ที่เป็น MySQL ได้ง่ายและสะดวกยิ่งขึ้น โดย phpMyAdmin ก็ถือเป็นเครื่องมือชนิดหนึ่งในการจัดการนั่นเอง

phpMyAdmin เป็นส่วนต่อประสานที่สร้างโดยภาษา PHP ซึ่งใช้จัดการฐานข้อมูล MySQL ผ่านเว็บเบราว์เซอร์ โดยสามารถที่จะทำการสร้างฐานข้อมูลใหม่ หรือทำการสร้าง TABLE ใหม่ๆ และยังมี function ที่ใช้สำหรับการทดสอบการ query ข้อมูลด้วยภาษา SQL พร้อมกันนั้น ยังสามารถทำการ insert delete update หรือแม้กระทั่งใช้ คำสั่งต่างๆ เหมือนกับการใช้ภาษา SQL ในการสร้างตารางข้อมูล

phpMyAdmin เป็นโปรแกรมประเภท MySQL Client ตัวหนึ่งที่ใช้ในการจัดการข้อมูล MySQL ผ่าน [web browser](#) ได้โดยตรง phpMyAdmin ตัวนี้จะทำงานบน [Web server](#) เป็น PHP Application ที่ใช้ควบคุมจัดการ MySQL Server

ความสามารถของ phpMyAdmin คือ

1. สร้างและลบ Database
2. สร้างและจัดการ Table เช่น แทรก record, ลบ record, แก้ไข record, ลบ Table, แก้ไข field
3. โหลดเท็กซ์ไฟล์เข้าไปเก็บเป็นข้อมูลในตารางได้
4. หาผลสรุป (Query) ด้วยคำสั่ง SQL



## Web Hosting

เว็บโฮสติ้ง (Web Hosting) คือ รูปแบบการให้บริการที่อนุญาตให้ผู้ให้บริการ สามารถนำเว็บเพจของตนเอง เพื่อออนไลน์บนอินเทอร์เน็ตได้ ซึ่งเว็บโฮสติ้งมีอีกชื่อหนึ่งว่า ' HSP ' ย่อมาจาก Hosting Service Provider หรือผู้ให้บริการโฮสติ้ง เป็นธุรกิจที่นำเอาเทคโนโลยีที่จำเป็นสำหรับงานสร้างเว็บไซต์ มาให้บริการ โดยมีวัตถุประสงค์เพื่อให้เว็บไซต์นั้นสามารถมองเห็นได้บนอินเทอร์เน็ต

ทุกเว็บไซต์ที่ออนไลน์บนอินเทอร์เน็ตจะต้องได้รับการฝาก หรือเก็บไว้บนคอมพิวเตอร์พิเศษ ที่เรียกว่า เว็บเซิร์ฟเวอร์ ซึ่งเซิร์ฟเวอร์นี้จะทำหน้าที่เป็นตัวติดต่อกับทุกหนทุกแห่งตลอด 24 ชั่วโมง เพื่อให้เว็บไซต์ ของคุณ สามารถเข้าถึงได้จากทุกที่ทุกเวลาในโลกที่มีการต่อเชื่อม อินเทอร์เน็ตง่าย ๆ แค่มพิมพ์ชื่อเว็บไซต์ของคุณ (Domain Name) ตัวอย่างเช่น <http://www.yourcompany.com/>

ผู้ให้บริการจะทำการติดตั้งระบบทั้งหมดให้คุณ เมื่อมีบุคคลที่พิมพ์ชื่อเว็บไซต์ของคุณ (Web address) ซื่อนั้นจะถูกส่งตามเส้นทางจากเครื่องคอมพิวเตอร์หนึ่งไปยังอีกเครื่องหนึ่งไปเรื่อยๆ เซิร์ฟเวอร์ที่คุณได้ฝากเว็บไซต์ไว้ (Host computer) ในเวลาเพียงเสี้ยววินาที

ดังนั้นการออนไลน์เว็บไซต์ของคุณบนอินเทอร์เน็ต สิ่งแรกที่คุณจำเป็นต้องมีคือ เซิร์ฟเวอร์ ซึ่งอย่างไรก็ตามการติดตั้งระบบเว็บเซิร์ฟเวอร์เป็นของตนเองสามารถทำได้ แต่มีค่าใช้จ่ายที่ค่อนข้างสูงมากและจำเป็นอย่างยิ่งที่จะต้อง มีผู้เชี่ยวชาญด้านเทคนิคดูแล ดังนั้น บริษัทส่วนใหญ่จะไม่ดำเนินการลงทุนเพื่อเป็นเจ้าของเอง และนี่คือที่มาของบริการเว็บโฮสติ้ง เว็บโฮสติ้งที่ดีจะต้องให้บริการทั้งเครื่องมืออำนวยความสะดวกและคำแนะนำแก่ผู้ให้บริการ

บริการโฮสติ้งส่วนใหญ่มีข้อดีอันหนึ่งที่จัดได้ว่า เป็นการให้บริการจะดำเนินการแบบเบ็ดเสร็จ (Outsourced service) ซึ่งผู้ให้บริการไม่ต้องยุ่งยากกับระบบเซิร์ฟเวอร์แต่อย่างใด ผู้ให้

บริการเว็บโฮสติ้งที่ดีที่สุด จะต้องมียระบบที่ใช้งานง่าย และให้ผู้ใช้งานไม่ต้องรู้สึกวุ่นวายเกี่ยวกับ ฟังก์ชันที่ซับซ้อนของระบบ เพื่อให้คุณสามารถเผยแพร่และนำเสนอผลงานได้อย่างง่ายดาย รวดเร็ว ที่สำคัญยังคงไว้ซึ่งระบบรักษาความปลอดภัยที่ดีเยี่ยม

## ขั้นตอนในการติดตั้ง Docker

\*เพื่อไม่ให้เกิดปัญหาติดขัดระหว่างการติดตั้ง กรุณาอัปเดต บ่อยๆ

```
[root@docker]# yum update -y
```

วิธีการติดตั้ง Docker

```
[root@docker ~]# yum install docker
```

```
CentOS Linux 8 - AppStream          5.5 kB/s | 4.3 kB   00:00
CentOS Linux 8 - AppStream          4.2 MB/s | 6.3 MB   00:01
CentOS Linux 8 - BaseOS             4.9 kB/s | 3.9 kB   00:00
CentOS Linux 8 - BaseOS            1.5 MB/s | 2.3 MB   00:01
CentOS Linux 8 - Extras             495 B/s | 1.5 kB    00:03
CentOS Linux 8 - Extras            10 kB/s | 9.2 kB    00:00
Docker CE Stable - x86_64           25 kB/s | 3.5 kB    00:00
Docker CE Stable - x86_64           40 kB/s | 11 kB     00:00
Docker CE Test - x86_64             33 kB/s | 3.5 kB    00:00
Docker CE Test - x86_64            45 kB/s | 14 kB     00:00
```

Error:

Problem: problem with installed package docker-ce-cli-1:20.10.4-3.el8.x86\_64

- package docker-ce-cli-1:20.10.4-3.el8.x86\_64 conflicts with docker provided by podman-docker-2.2.1-7.module\_el8.3.0+699+d61d9c41.noarch
- package docker-ce-cli-1:19.03.13-3.el8.x86\_64 conflicts with docker provided by podman-docker-2.2.1-7.module\_el8.3.0+699+d61d9c41.noarch
- package docker-ce-cli-1:19.03.14-3.el8.x86\_64 conflicts with docker provided by podman-docker-2.2.1-7.module\_el8.3.0+699+d61d9c41.noarch
- package docker-ce-cli-1:19.03.15-3.el8.x86\_64 conflicts with docker provided by podman-docker-2.2.1-7.module\_el8.3.0+699+d61d9c41.noarch
- package docker-ce-cli-1:20.10.0-3.el8.x86\_64 conflicts with docker provided by podman-docker-2.2.1-7.module\_el8.3.0+699+d61d9c41.noarch
- package docker-ce-cli-1:20.10.1-3.el8.x86\_64 conflicts with docker provided by podman-docker-2.2.1-7.module\_el8.3.0+699+d61d9c41.noarch

```

- package docker-ce-cli-1:20.10.2-3.el8.x86_64 conflicts with docker provided by podman-docker-2.2.1-7.module_el8.3.0+699+d61d9c41.noarch
- package docker-ce-cli-1:20.10.3-3.el8.x86_64 conflicts with docker provided by podman-docker-2.2.1-7.module_el8.3.0+699+d61d9c41.noarch
- package docker-ce-cli-1:20.10.5-3.el8.x86_64 conflicts with docker provided by podman-docker-2.2.1-7.module_el8.3.0+699+d61d9c41.noarch
- package docker-ce-cli-1:20.10.0-1.1.beta1.el8.x86_64 conflicts with docker provided by podman-docker-2.2.1-7.module_el8.3.0+699+d61d9c41.noarch
- package docker-ce-cli-1:20.10.0-2.1.rc1.el8.x86_64 conflicts with docker provided by podman-docker-2.2.1-7.module_el8.3.0+699+d61d9c41.noarch
- package docker-ce-cli-1:20.10.0-2.2.rc2.el8.x86_64 conflicts with docker provided by podman-docker-2.2.1-7.module_el8.3.0+699+d61d9c41.noarch
- conflicting requests
(trial to add '--alloweraseing' to command line to replace conflicting packages or '--skip-broken' to skip uninstalable packages or '--nobest' to use not only best candidate packages)

```

ติดตั้ง Packages ของ Docker

```

[root@docker ~]# yum install
https://download.docker.com/linux/centos/7/x86\_64/stable/Packages/containerd.io-1.2.6-3.3.el7.x86\_64.rpm -y

```

```

Last metadata expiration check: 0:01:46 ago on Sun 07 Mar 2021 03:19:02 AM EST.
containerd.io-1.2.6-3.3.el7.x86_64.rpm          6.0 MB/s | 26 MB   00:04
Dependencies resolved.

```

```

=====
Package           Architecture Version           Repository        Size
=====
Downgrading:
containerd.io      x86_64         1.2.6-3.3.el7     @commandline      26 M
docker-ce          x86_64         3:19.03.15-3.el8  docker-ce-stable  24 M

```

```

Transaction Summary

```

```

=====
Downgrade 2 Packages

```

```

Total size: 50 M

```

```

Total download size: 24 M

```

```

Downloading Packages:
docker-ce-19.03.15-3.el8.x86_64.rpm          4.9 MB/s | 24 MB   00:04
-----
Total                                4.9 MB/s | 24 MB   00:04
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing      :                                1/1
  Running scriptlet: containerd.io-1.2.6-3.3.el7.x86_64      1/1
  Downgrading     : containerd.io-1.2.6-3.3.el7.x86_64      1/4
  Running scriptlet: containerd.io-1.2.6-3.3.el7.x86_64      1/4
  Downgrading     : docker-ce-3:19.03.15-3.el8.x86_64      2/4
  Running scriptlet: docker-ce-3:19.03.15-3.el8.x86_64      2/4
  Running scriptlet: docker-ce-3:20.10.4-3.el8.x86_64      3/4
  Cleanup         : docker-ce-3:20.10.4-3.el8.x86_64      3/4
  Running scriptlet: docker-ce-3:20.10.4-3.el8.x86_64      3/4
  Running scriptlet: containerd.io-1.4.3-3.1.el8.x86_64      4/4
  Cleanup         : containerd.io-1.4.3-3.1.el8.x86_64      4/4
  Running scriptlet: containerd.io-1.4.3-3.1.el8.x86_64      4/4
  Verifying       : docker-ce-3:19.03.15-3.el8.x86_64      1/4
  Verifying       : docker-ce-3:20.10.4-3.el8.x86_64      2/4
  Verifying       : containerd.io-1.2.6-3.3.el7.x86_64      3/4
  Verifying       : containerd.io-1.4.3-3.1.el8.x86_64      4/4

Downgraded:
containerd.io-1.2.6-3.3.el7.x86_64      docker-ce-3:19.03.15-3.el8.x86_64

```

อนุญาตให้ port สามารถผ่าน firewall ได้

```

[root@docker ~]# firewall-cmd --zone=public --add-port=8085/tcp --permanent
// port ที่ต้องการอนุญาตให้ผ่าน firewall
success
[root@docker ~]# firewall-cmd --reload
success

```

ติดตั้ง cgroup เพื่อการใช้งานคำสั่ง systemd

```

[root@docker ~]# yum -y install libcgroup libcgroup-tools
[root@docker ~]# systemctl start cgconfig

```



การเปิดใช้งาน docker

```
[root@docker ~]# systemctl start docker // เปิดการใช้งาน docker
[root@docker ~]# systemctl enable docker // ใช้งาน docker ทันทีที่เปิดเครื่อง
[root@docker ~]# systemctl status docker // เช็คสถานะของ docker
```

ดาวน์โหลด images ของ docker มาใช้งาน

```
[root@docker ~]# docker pull centos

Using default tag: latest
latest: Pulling from library/centos
Digest: sha256:5528e8b1b1719d34604c87e11dcd1c0a20bedf46e83b5632cdeac91b8c04efc1
Status: Image is up to date for centos:latest
docker.io/library/centos:latest
```

เช็ค images ของ docker

```
[root@docker ~]# docker images
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
6f88384c9859	centos	"nginx -g 'daemon of...'"	3 minutes ago	Up 3 minutes	0.0.0.0:8089->80/tcp	priceless_mirzakhani

run images ของ docker

```
[root@docker ~]# docker run -d -v /sys/fs/cgroup:/sys/fs/cgroup:ro --cap-add SYS_ADMIN -p 8089:80 -v /webpages:/var/www/html --name systemd_websrv centos /sbin/init
```

เริ่มต้นใช้งาน images ที่ต้องการ

```
[root@docker ~]# docker start 6f88384c9859
```

เข้าใช้งาน docker

```
[root@docker ~]# docker run -it centos bash
```

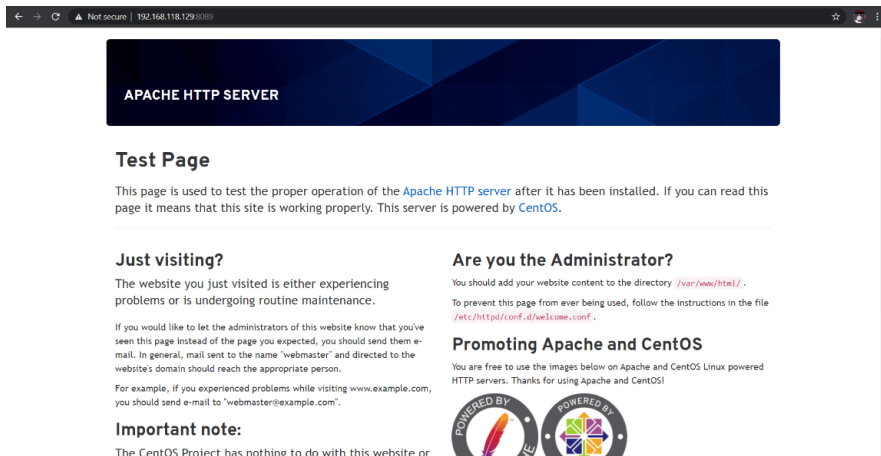
## การติดตั้ง web server

```
[root@4a86e703a679 /]# yum install httpd -y
```

เปิดใช้งาน service ของ webserver

```
[root@4a86e703a679]# systemctl start httpd.service
```

ตรวจสอบ web server ด้วยเลข IP ของ webserver // ในกรณีนี้ได้ทำการเชื่อม docker ด้วย port 8089



## ติดตั้ง Database Server

```
[root@a2589a2c1d83 /]# yum install mariadb-server mariadb
```

เปิดการใช้งาน mariadb

```
[root@94aeb9cdfa9c /]# systemctl start mariadb
```

ติดตั้ง security ของ Database

```
[root@94aeb9cdfa9c /]# mysql_secure_installation
```

NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB

# SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!

In order to log into MariaDB to secure it, we'll need the current password for the root user. If you've just installed MariaDB, and you haven't set the root password yet, the password will be blank, so you should just press enter here.

Enter current password for root (enter for none): กดปุ่ม Enter

OK, successfully used password, moving on...

Setting the root password ensures that nobody can log into the MariaDB root user without the proper authorisation.

Set root password? [Y/n] y

New password: ใส่ password root

Re-enter new password: ใส่ password root อีกครั้งเพื่อยืนยัน

Password updated successfully!

Reloading privilege tables..

... Success!

By default, a MariaDB installation has an anonymous user, allowing anyone to log into MariaDB without having to have a user account created for them. This is intended only for testing, and to make the installation go a bit smoother. You should remove them before moving into a production environment.

Remove anonymous users? [Y/n] y

... Success!

Normally, root should only be allowed to connect from 'localhost'. This ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n] y

... Success!

By default, MariaDB comes with a database named 'test' that anyone can access. This is also intended only for testing, and should be removed before moving into a production environment.

Remove test database and access to it? [Y/n] y

- Dropping test database...

... Success!

- Removing privileges on test database...

... Success!

Reloading the privilege tables will ensure that all changes made so far will take effect immediately.

Reload privilege tables now? [Y/n] **y**

... Success!

Cleaning up...

All done! If you've completed all of the above steps, your MariaDB installation should now be secure.

Thanks for using MariaDB!

## การติดตั้ง PHP

เพิ่ม repository ของ Extra Packages for Enterprise Linux (EPEL) on CentOS version

```
[root@94aeb9cdfa9c /]# rpm -Uvh https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm
rpm -Uvh http://rpms.remirepo.net/enterprise/remi-release-7.rpm
[root@94aeb9cdfa9c /]# yum install yum-utils -y
```

ติดตั้ง php 7

```
[root@94aeb9cdfa9c /]# yum install php -y
```

ตรวจสอบเวอร์ชันของ PHP

```
[root@94aeb9cdfa9c /]# php -v
PHP 7.2.24 (cli) (built: Oct 22 2019 08:28:36) ( NTS )
Copyright (c) 1997-2018 The PHP Group
Zend Engine v3.2.0, Copyright (c) 1998-2018 Zend Technologies
```

## การติดตั้ง phpMyAdmin

ติดตั้งส่วนเสริมของ php

```
[root@94aeb9cdfa9c /]# yum -y install httpd php-xml php php-mysqlnd php-json wget policycoreutils-python-utils
```

ดาวน์โหลดไฟล์ zip ของ phpMyAdmin

```
[root@7b887db0a85d /]# wget https://files.phpmyadmin.net/phpMyAdmin/5.1.0/phpMyAdmin-5.1.0-all-languages.zip
```

ติดตั้งตัวแตกไฟล์

```
[root@7b887db0a85d /]# yum install unzip -y
```

แตกไฟล์ zip ของ phpMyAdmin

```
[root@7b887db0a85d /]# unzip phpMyAdmin-5.1.0-all-languages.zip
```

สร้าง tmp ไฟล์

```
[root@7b887db0a85d /]# mkdir tmp
```

เปลี่ยนการเข้าถึงของไฟล์ให้เราสามารถใช้งานได้

```
[root@7b887db0a85d /]# chmod 777 tmp
```

ทำการคัดลอกไฟล์การตั้งค่าของ phpMyAdmin

```
[root@7b887db0a85d /]# cp phpMyAdmin/config.sample.inc.php phpMyAdmin/config.inc.php
```

```
[root@7b887db0a85d /]# semanage fcontext -a -t httpd_sys_rw_content_t  
'/usr/share/phpMyAdmin/' semanage fcontext -a -t httpd_sys_rw_content_t  
'/usr/share/phpMyAdmin/tmp(/.*)' restorecon -Rv '/phpMyAdmin/'
```

เข้า MYSQL แล้วสร้างตาราง

```
[root@7b887db0a85d/]# sudo mysql < /usr/share/phpMyAdmin/sql/create_tables.sql -u root -p
```

ใส่รหัสผ่าน

ทำการตั้งค่าไฟล์ config

```
[root@7b887db0a85d/]# sudo nano /etc/httpd/conf.d/phpMyAdmin.conf
```

```
Alias /phpmyadmin /usr/share/phpMyAdmin/
Alias /dbms /usr/share/phpMyAdmin/
<Directory /usr/share/phpMyAdmin/>
    AddDefaultCharset UTF-8
```

```
<IfModule mod_authz_core.c>
    # Apache 2.4
    <RequireAny>
        Require all granted
    </RequireAny>
</IfModule>
<IfModule !mod_authz_core.c>
    # Apache 2.2
    Order Deny,Allow
    Deny from All
    Allow from 127.0.0.1
    Allow from ::1
</IfModule>
</Directory>
```

```
<Directory /usr/share/phpMyAdmin/setup/>
    <IfModule mod_authz_core.c>
        # Apache 2.4
        <RequireAny>
            Require all granted
        </RequireAny>
```

```
</IfModule>
```

```
<IfModule !mod_authz_core.c>
    # Apache 2.2
    Order Deny,Allow
    Deny from All
    Allow from 127.0.0.1
    Allow from ::1
</IfModule>
```

```
</Directory>
```

เสร็จแล้วทำการกด save ไฟล์ที่ทำการตั้งค่า

ทำการ restart web server httpd

```
[root@7b887db0a85d/]# sudo systemctl restart httpd
```

## การติดตั้ง Web Hosting (Cockpit)

ติดตั้ง EPEL(Extra Package for Enterprise Linux)

```
[root@dns]# yum install epel-release
```

ติดตั้ง Cockpit

```
[root@dns]# yum install webmin
```

เปิดใช้งาน Cockpit

```
[root@dns]# systemctl start webmin
[root@dns]# systemctl enable webmin
```

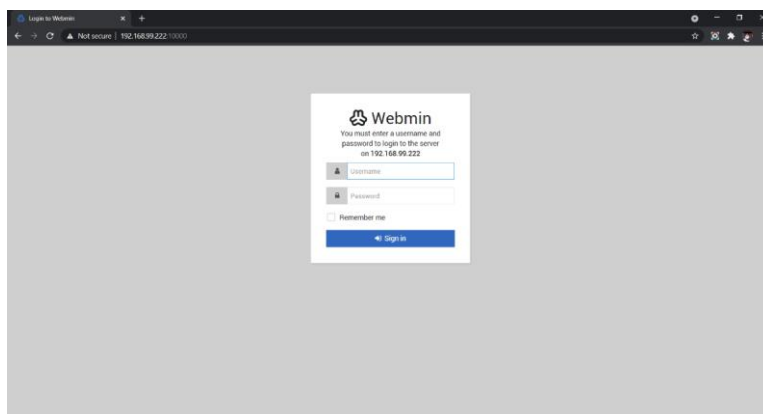
เพิ่ม service ให้สามารถใช้กับ firewall ได้

```
[root@dns]# firewall-cmd --add-service=webmin
root@dns]# firewall-cmd --add-service=webmin --permanent
```

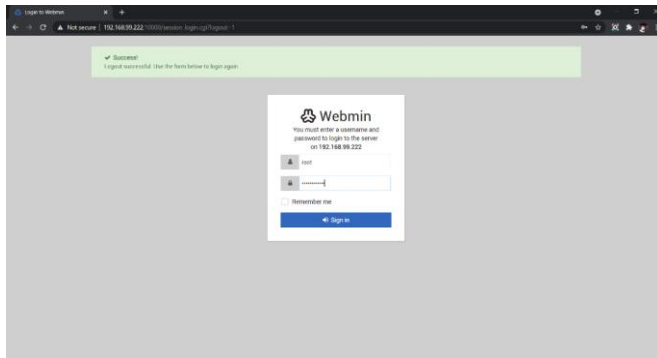
เปิดใช้งาน firewall ใหม่อีกครั้ง

```
root@dns]# firewall-cmd --reload
```

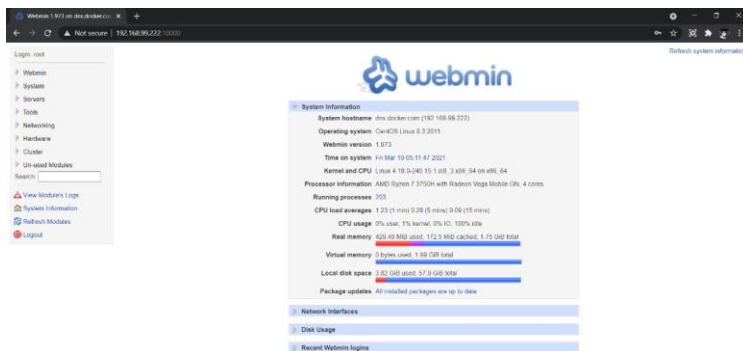
เข้าใช้งาน webmin ด้วย ip – server ตามด้วย port 10000



เข้าใช้งานด้วย Root



หน้า dashboard ของ Webmin





## บรรณานุกรม

วิธีการติดตั้งและใช้งาน Docker เข้าถึงได้จาก <https://www.tecmint.com/install-docker-and-learn-containers-in-centos-rhel-7-6/>

วิธีการติดตั้ง Web Server เข้าถึงได้จาก <https://medium.com/swlh/docker-and-systemd-381dfd7e4628>

คำสั่งต่างๆ ของ Docker เข้าถึงได้จาก <https://igokuz.com/docker-command-เบื้องต้น-101-df1a7fb2a24c>

การติดตั้งไฟล์ cgroup เข้าถึงได้จาก [https://www.server-world.info/en/note?os=CentOS\\_7&p=cgroups&f=1](https://www.server-world.info/en/note?os=CentOS_7&p=cgroups&f=1)

อนุญาต firewall เข้าถึงได้จาก <https://www.thegeekdiary.com/how-to-open-a-ports-in-centos-rhel-7/>