

TUTORIAL LINUX SERVER

**LANGKAH BELAJAR MENJADI LINUX
ADMINISTRATOR YANG HANDAL**



Areta Informatic College :

Blok CD No.2 Dasana Xenter, Bonang
Tangerang - Banten, Indonesia



0853 1226 1634



admin@aretacollege.com www.aretacollege.com

Silabus belajar

1. Setting IP Address
 - a. Menggunakan nmtui
 - b. Menggunakan konfigurasi command line
2. SSH Server
 - a. Instalasi ssh
 - b. Akses ssh
 - c. Rubah port ssh
 - d. Upload file ke sever menggunakan sftp
3. DNS Server
 - a. Install DNS Server
 - b. Konfigurasi DNS Server
 - c. Open Port
 - d. Testing DNS Server
4. Web Server
 - a. Install web server Apache
 - b. Konfigurasi WebServer
 - c. Open Port
 - d. Konfigurasi Virtual Host
5. Database Server
 - a. Install database mysql server
 - b. Konfigurasi database
 - c. Install PhpMyadmin
 - i. Phpmyadmin bisa di akses publik
6. Instalall Aplikasi PHP
 - a. Install Wordpress
 - b. Install PHP CRUD
 - c. Install PHP dengan CodeIgniter
 - d. Install Owncloud
7. Install FTP
 - a. Install ftp server
 - b. Testing ftp
8. Rsync
 - a. Install rsync
 - b. Konfigurasi Rsync
 - c. Testing
9. Cron Server
 - a. Install Cron Server
 - b. Konfigurasi Cron
10. Firewall

1. Setting IP Address

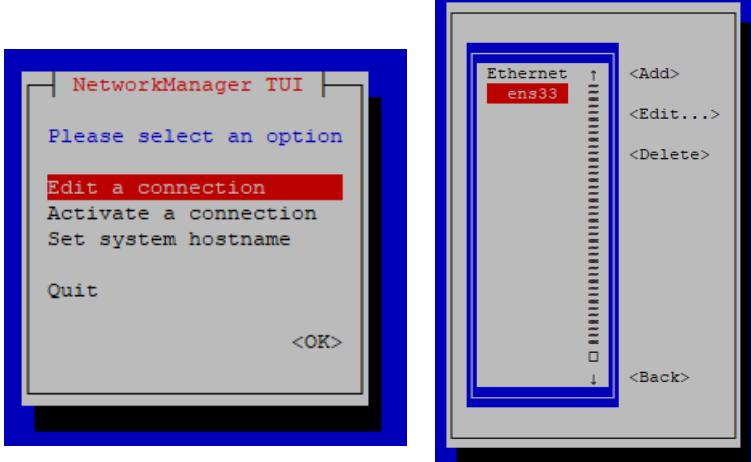
a. Menggunakan nmtui

Konfigurasi ip address pada server menggunakan nmtui yaitu dengan gui cukup mudah yaitu dengan perintah seperti ini :

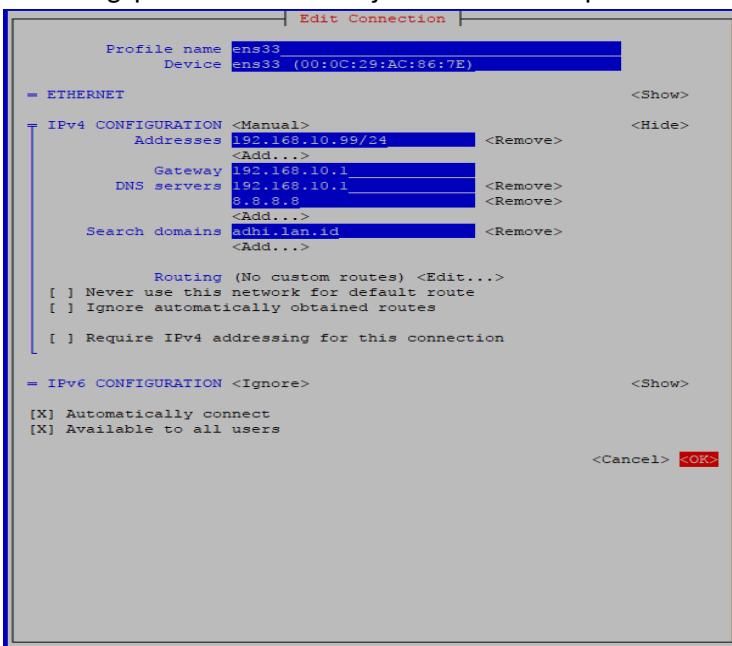
```
[root@adhi ~]# nmtui
```

Setelah itu akan muncul gui konfigurasi ip address :

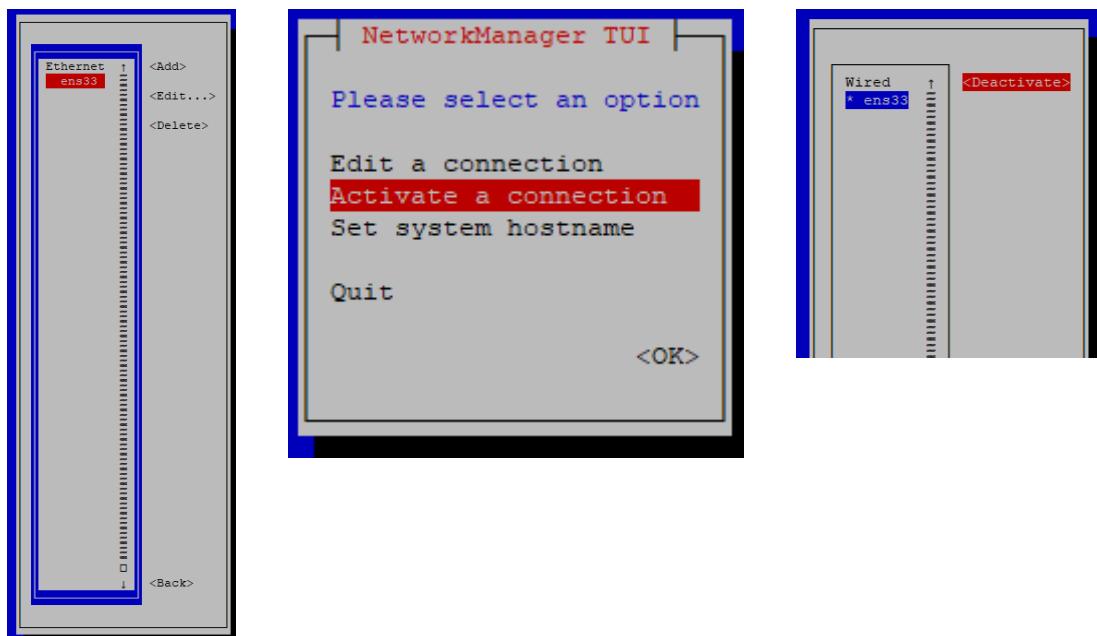
- a) Enter edit a connection → pilih ens33 yaitu ethernet server → pilih Edit :



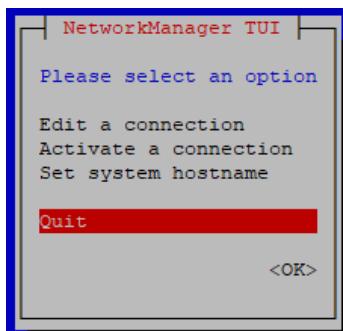
- b) Lalu setting ip address server → jika sudah selesai pilih ok dan enter :



- c) Pilih back → pilih active connection → pilih disabled dan deactivate lalu activate:



- d) Pilih quit dan cek ip address server sudah up atau masih down dengan perintah ip a :



```
[root@adhi ~]# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN
qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast
state UP qlen 1000
    link/ether 00:0c:29:ac:86:7e brd ff:ff:ff:ff:ff:ff
    inet 192.168.10.99/24 brd 192.168.10.255 scope global ens33
        valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:feac:867e/64 scope link tentative dadfailed
        valid_lft forever preferred_lft forever
[root@adhi ~]#
```

Install tools network dan firewalld:

```
[root@192 ~]# yum install net-tools && yum install firewalld
```

b. Menggunakan konfigurasi command line

```
[root@adhi ~]# vi /etc/sysconfig/network-scripts/ifcfg-ens33
```

Isi konfigurasi :

```
TYPE=Ethernet
BOOTPROTO=none
DNS1=192.168.10.1
DNS2=8.8.8.8
DEFROUTE=yes
IPV4_FAILURE_FATAL=no
IPV6INIT=no
IPV6_AUTOCONF=yes
IPV6_DEFROUTE=yes
IPV6_PEERDNS=yes
IPV6_PEERROUTES=yes
IPV6_FAILURE_FATAL=no
IPV6_ADDR_GEN_MODE=stable-privacy
NAME=ens33
UUID=68bb280e-f7aa-480b-b27d-13b25600b9b9
DEVICE=ens33
ONBOOT=yes
ZONE=public
IPADDR=192.168.10.99
PREFIX=24
GATEWAY=192.168.10.1
DOMAIN=adhi.lan.id
```

2. SSH Server

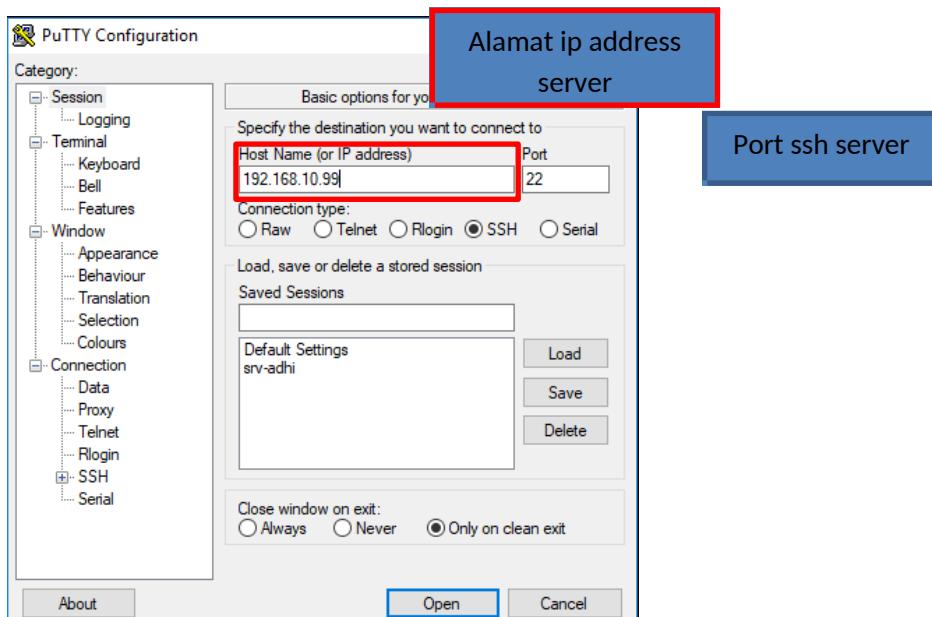
a. Instalasi ssh

```
[root@adhi ~]# yum install openssh
```

b. Akses ssh

Tools yang di gunakan adalah putty, tetapi banyak tool yang bisa di gunakan yaitu mobaxterm, kitty, mRemoteNG, Xshell4, Bitvise SSH Client.

masukan ip address server dan port server jika sudah masukan user dan password server



c. Rubah port ssh

Backup konfigurasi awal terlebih dahulu supaya bisa default jika terjadi kesalahan konfigurasi

```
[root@adhi ~]# cd /etc/ssh/  
[root@adhi ssh]# cp sshd_config sshd_config_backup
```

konfigurasi ssh yang akan di rubah port ssh server :

```
[root@adhi ssh]# nano sshd_config
```

Isi konfigurasi :

```
Port 2017  
#AddressFamily any  
#ListenAddress 0.0.0.0
```

Jika sudah konfigurasi port, lalu open port pada security linux centos nya, install terlebih dahulu tool manage port :

```
[root@adhi ssh]# yum -y install policycoreutils-python
```

Lalu open open ssh yang baru :

```
[root@adhi ssh]# semanage port -a -t ssh_port_t -p tcp 2017
[root@adhi ssh]# firewall-cmd --permanent --add-port=2017/tcp
success
[root@adhi ssh]# firewall-cmd --reload
success
```

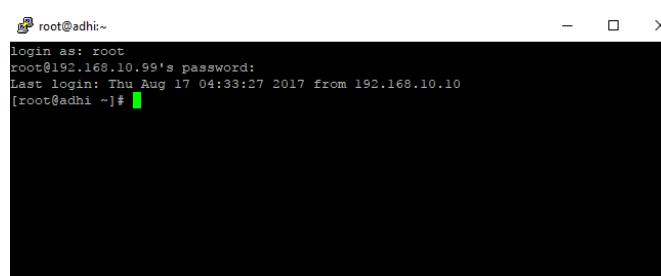
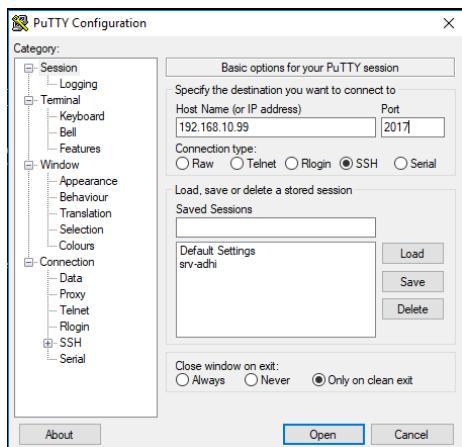
restart service ssh server :

```
[root@adhi ssh]# service sshd restart
```

Sekarang server ssh bukan port 22 lagi melainkan menggunakan port 2017, untuk cek port nya dengan perintah tersebut :

```
[root@adhi ssh]# netstat -tulpn
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address          Foreign Address
State      PID/Program name
tcp        0      0 127.0.0.1:25           0.0.0.0:*
LISTEN    1915/master
tcp        0      0 0.0.0.0:2017          0.0.0.0:*
LISTEN    2816/sshd
tcp6       0      0 :::80                 :::*
LISTEN    2189/httpd
tcp6       0      0 ::1:25                :::*
LISTEN    1915/master
tcp6       0      0 :::2017              :::*
LISTEN    2816/sshd
```

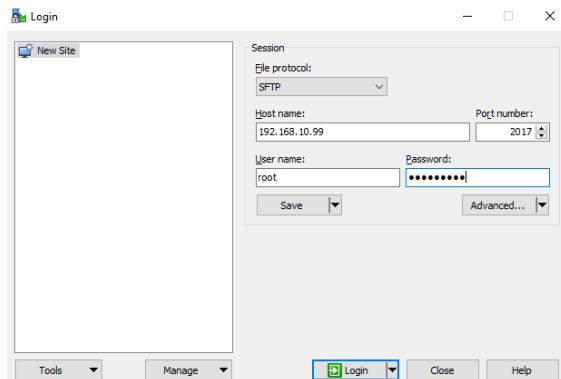
sekarang test dengan putty dengan port 22 maka tidak akan bisa akses, melainkan dengan port 2017



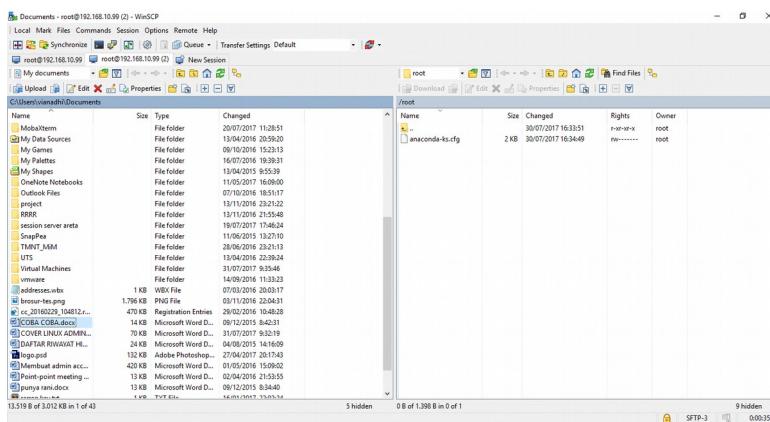
Upload file ke sever menggunakan sftp

suatu ketika seorang system administrator ingin mengupload suatu file ke server salah satu nya dengan sftp (secure file transfer protocol) yaitu service ssh bukan hanya untuk remote server tetapi juga bisa mengupload dan download file ke server. Langsung saja ke tkp.

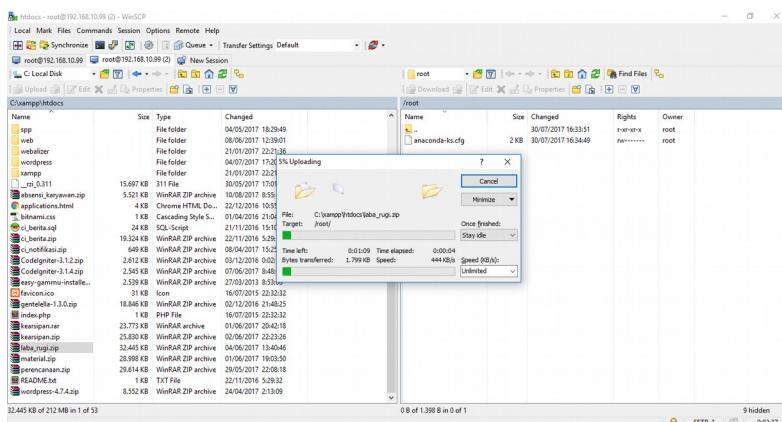
Salah satu tools untuk upload file yaitu winscp atau pun filezile, kali ini menggunakan winscp. isikan ip address server → port ssh → user dan password server → klik login



Lalu akan muncul data komputer sysadmin dan server



Drag dari komputer sysadmin ker server



3. DNS Server

DOMAIN NAME SYSTEM

Domain Name System (DNS) digunakan untuk traslasi dari IP address ke nama host yang lebih mudah di ingat sehingga klient dapat terhubung ke web server atau email server menggunakan nama domain bukan IP address. Saat berselancar dari satu website ke website lainnya atau mengirim e-mail, tentu saja domain name system telah berperan di dalam nya.

Sebagai contoh, penggunaan URL <http://adhi.lan.id> dan alamat email rani@adhi.lan.id lebih mudah untuk diingat dibandingkan dengan alamat IP <http://192.168.10.99>. Alamat IP 192.168.10.99 adalah IP yang digunakan oleh mesin server adhi.lan.id dalam mengerjakan tugasnya seperti pengalaman dan perutungan untuk berkomunikasi antara host dengan host yang lain dalam sebuah jaringan.

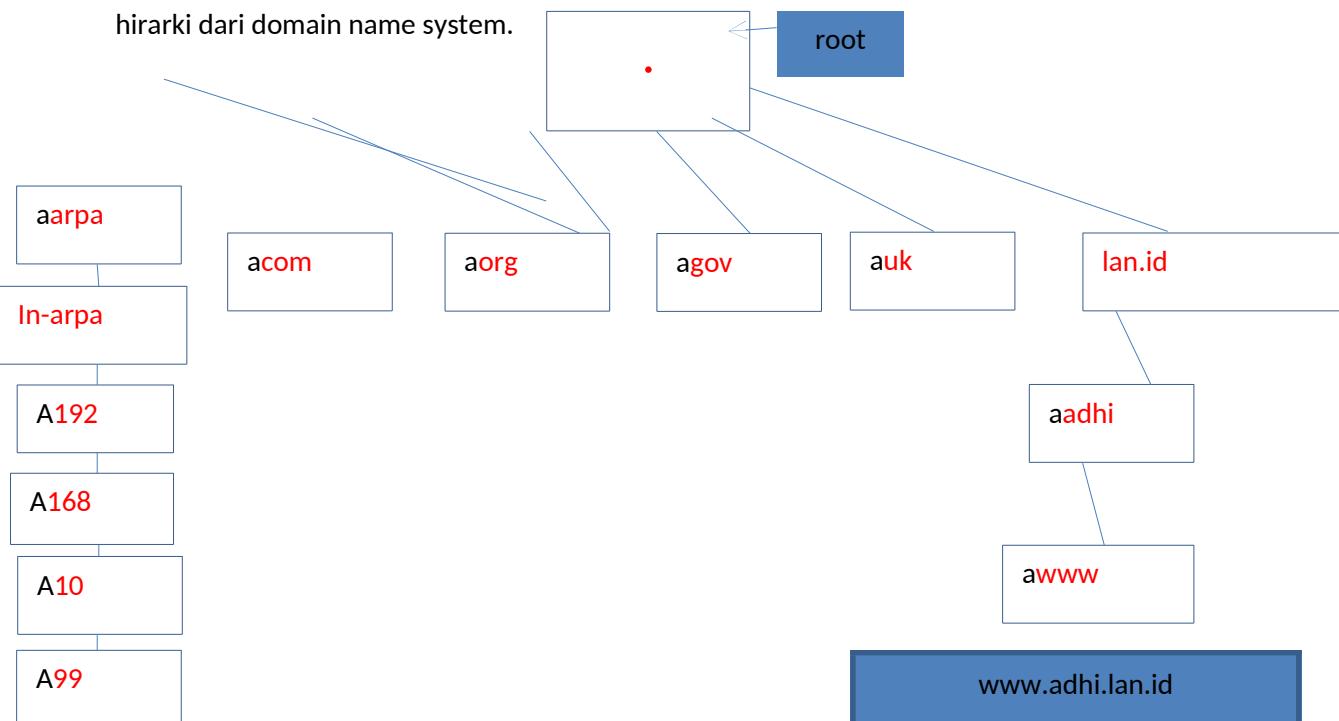
Sejarah Singkat Domain Name System (DNS)

Domain Name System pertama kali di kembangkan oleh Paul Mockapetris pada tahun 1983. Setahun kemudian tepatnya 1984, empat mahasiswa Berkeley – Douglas Terry, Mark Painter, David Riggle dan Songnian Zhou mengimplementasikan pertama kali ke mesin UNIX.

Software yang dikembangkan di Berkeley ini kemudian di beri nama oleh Dunlop tahun 1985 dengan BIND-Berkeley Internet Name Domain. Seiring dengan waktu BIND kemudian menjadi software DNS paling banyak digunakan di Internet terutama untuk sistem UNIX.

Struktur Domain Name System

DNS menggunakan sebuah hirarki untuk pengaturan sistem database terdistribusi yang dimilikinya. Pohon DNS (DNS Tree) terdiri dari root level, top level domain (TLD), second level domain, sub-domain, host. Top level domain (TLD) ditempati oleh com, org, gov, uk, id. Berikut hirarki dari domain name system.



Bagaimana DNS bekerja

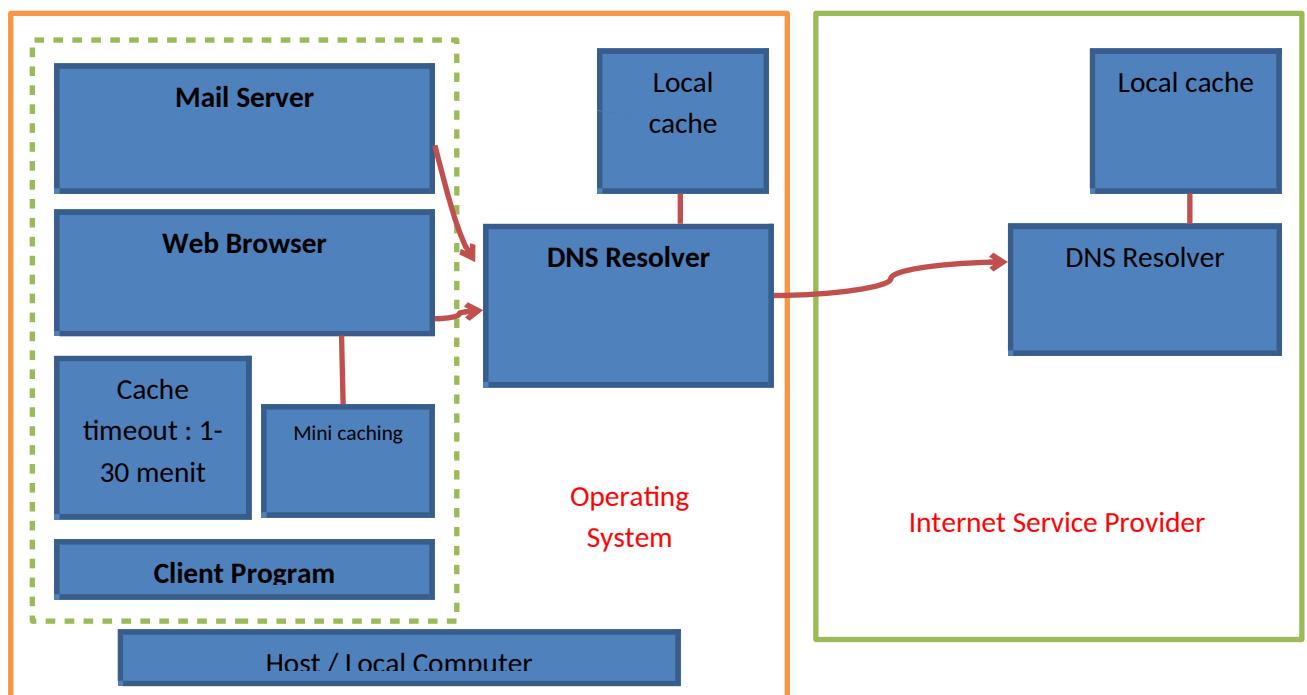
Interaksi pengguna sebagai klien tidak akan berhubungan secara langsung dengan DNS resolver, pengguna lebih banyak berinteraksi secara langsung dengan program aplikasi seperti web browser (Safari, Mozilla, Opera, dan client mail seperti Outlook, Kmail, Mozilla, Thunderbird). Ketika pengguna melakukan aktifitas, yang meminta pencarian DNS(hampir seluruh aktifitas yang dilakukan di Internet pasti akan meminta pencarian DNS), program tersebut akan meminta ke DNS resolver yang terdapat pada sistem operasi yang sedang digunakan pengguna.

DNS resolver tentunya memiliki cache tersendiri yang disimpan secara lokal. DNS resolver lokal inilah yang berfungsi untuk menjawab setiap permintaan DNS oleh program aplikasi yang di akses oleh klien. Domain name system yang pernah di akses oleh pengguna akan tersimpan di cache tersebut untuk di gunakan pada kesempatan lain saat pengguna meminta DNS yang sama. Resolver akan menggunakan nilai yang terdapat di dalam cache untuk program yang memerlukan DNS tersebut.

Jika DNS yang dimaksud belum terdapat dalam cache lokal, maka DNS resolver akan mengirimkan permintaan ke server DNS tertentu. Pada gambar berikut, resolver yang terdapat di sistem operasi klien akan mengirimkan permintaan ke server DNS yang terdapat di penyedia layanan Internet (ISP) tempat pengguna berlangganan Internet. Hasil pencarian akan di berikan kepada DNS resolver yang terdapat pada sisi pengguna untuk selanjutnya di berikan kepada program yang membutuhkan. DNS tersebut juga akan di salin ke cache (DNS resolver ISP) untuk pengguna berikutnya. Jika permintaan juga belum dimiliki oleh resolver pihak ISP, makan DNS resolver ISP akan melakukan pemintaan ke recursive DNS search tentunya dalam lingkup pencarian secara global.

Selain cache yang dimiliki oleh DNS resolver, program aplikasi seperti web browser juga memiliki DNS mini cache untuk mengurangi referensi ke DNS resolver. Cache ini umumnya memiliki masa singkat dalam hitungan 1-30 menit.

Contoh DNS Resolver



Caching

Karena jumlah permintaan yang besar dari sistem DNS dan tentu saja membebani server yang bersangkutan. Sebuah mekanisme sangat dibutuhkan untuk mengurangi beban dari masing-masing server DNS. Ketika sebuah resolver klien menerima jawaban DNS, informasi tersebut akan di-cache untuk jangka waktu tertentu sehingga resolver tidak harus selalu meminta ke server ada program aplikasi di sisi klien yang membutukan informasi DNS tertentu dan telah tersedia di cache resolver.

Jangka waktu tertentu di mana informasi DNS masih di-cache oleh resolver dapat di-set oleh administrator pada bagian Time To Live(TTL). Penjelasan tentang cache akan dijelaskan saat instalasi BIND.

a. Install DNS Server

```
[root@adhi]# yum install bind bind-utils bind-*
```

b. Konfigurasi DNS Server

```
[root@adhi ~]# nano /etc/named.conf
```

Isi konfigurasi :

```
options {
    listen-on port 53 { 127.0.0.1; any; }; // tambahkan
konfigurasi any
    listen-on-v6 port 53 { ::1; };
    directory      "/var/named";
    dump-file     "/var/named/data/cache_dump.db";
    statistics-file "/var/named/data/named_stats.txt";
    memstatistics-file "/var/named/data/named_mem_stats.txt";
    allow-query    { localhost; any; }; // tambahkan konfigurasi
any

// tambahkan konfigurasi untuk file include dns forward dan reverse
dibawah // konfigurasi file hint

zone "." IN {
    type hint;
    file "named.ca";
};

zone "adhi.lan.id" IN {
    type master;
    file "db.adhi.lan.id";
    allow-update { none; };
};

zone "10.168.192.in-addr.arpa" IN {
    type master;
    file "10.168.192.db";
    allow-update { none; };
};
```

Buat file forward dns di /var/named :

```
[root@adhi ~]# cd /var/named/  
[root@adhi named]# nano db.adhi.lan.id
```

Isi konfigurasi forward dns :

```
; BIND data file for local loopback interface  
;  
$TTL    604800  
@       IN      SOA     ns1.adhi.lan.id. vianadhi.adhi.lan.id. (  
                      09082017      ; Serial  
                      604800        ; Refresh  
                      86400         ; Retry  
                     2419200       ; Expire  
                     604800 )      ; Negative Cache TTL  
;  
@       IN      NS      ns1.adhi.lan.id.  
@       IN      A       192.168.10.99  
@       IN      MX 10   mail.adhi.lan.id.  
ns1    IN      A       192.168.10.99  
www    IN      A       192.168.10.99  
mail   IN      A       192.168.10.99  
ftp    IN      A       192.168.10.99  
portal IN      A       192.168.10.99  
aplikasi IN      A       192.168.10.99
```

Time To Live (TTL). TTL adalah waktu dalam detik sebagai indikasi lamanya sebuah resolver menyimpan resource record (RR) di cache sebelum dihapus dan memintanya kembali ke server domain. Jika parameter TTL dikosongkan maka secara default akan menggunakan minimum waktu di set pada start of authority (SOA).

```
$TTL    604800
```

Start Of Authority (SOA). Catatan otoritas awal yang mengacu pada DNS server yang menyediakan informasi tentang domain Internet.

```
@       IN      SOA     ns1.adhi.lan.id. vianadhi.adhi.lan.id. (
```

09082017	;	Serial
604800	;	Refresh
86400	;	Retry
2419200	;	Expire
604800)	;	Negative Cache TTL

Adhi.lan.id adalah name server utama domain ini dan vianadhi.adhi.lan.id adalah administrator/orang yang bertanggung jawab terhadap domain tersebut.

Refresh timeout adalah waktu yang menjadikan sebagai acuan kapan sebuah slave server akan memeriksa perubahan zona yang ada di server utama (master). Slave server akan membandingkan nilai dari serial number dan jika serial number yang ada di server utama lebih besar maka slave master akan menyalin data zona yang ada.

Retry timeout adalah waktu yang dijadikan acuan kapan sebuah slave server akan memeriksa kembali serial number dari server utama jika slave tidak mendapatkan jawaban atau respon.

Expire adalah sebagai acuan berapa lama slave master mengambil alih kendali pelayanan data otoritas domain saat aster server sedang down.

Name Server (NS). Digunakan untuk mendefinisikan server mana saja yang merupakan otoritas dari sebuah zona dan mendelegasikan subdomain ke organisasi lain.

```
@ IN NS ns1.adhi.lan.id.
```

Mail Exchanger (MX). Rekaman MX dibutuhkan oleh sebuah sistem mail untuk merutekan e-mail secara efisien.

```
@ IN MX 10 mail.adhi.lan.id.
```

Address (A). Data untuk catatan 'A' berisi alamat IP (Internet address) yang merupakan pemetaan hostname ke IP address. Setiap komputer atau host harus memiliki catatan ini dengan alamat IP masing-masing yang sifatnya statis dan bukan dinamis.

```
@ IN A 192.168.10.99
```

Kemudian buat sebuah file lain dengan nama 10.168.192.db sebagai reverse zone dari db.adhi.lan.id isi konfigurasi reverse dns :

```
[root@adhi named]# nano 10.168.192.db
```

Isi konfigurasi :

```
; BIND reverse data file for local loopback interface
;
$TTL    604800
@       IN      SOA     ns1.adhi.lan.id. vianadhi.adhi.id. (
                          09082017      ; Serial
                          604800        ; Refresh
                          86400         ; Retry
                          2419200       ; Expire
                          604800 )       ; Negative Cache TTL
;
@       IN      NS      ns1.adhi.lan.id.
@       IN      PTR     adhi.lan.id.
ns1    IN      A       192.168.10.99
www    IN      A       192.168.10.99
mail   IN      A       192.168.10.99
ftp    IN      A       192.168.10.99
portal  IN      A       192.168.10.99
aplikasi IN      A       192.168.10.99
99     IN      PTR     ns1.adhi.lan.id.
99     IN      PTR     www.adhi.lan.id.
99     IN      PTR     mail.adhi.lan.id.
99     IN      PTR     ftp.adhi.lan.id.
99     IN      PTR     portal.adhi.lan.id.
```

99 IN PTR aplikasi.adhi.lan.id.

Konfigurasi resolv.conf :

```
[root@adhi named]# nano /etc/resolv.conf
```

Isi konfigurasi :

```
# Generated by NetworkManager
#resolv dns lokal
search adhi.lan.id
nameserver 192.168.10.99

#resolv internet
nameserver 192.168.10.1
nameserver 8.8.8.8
```

cek konfigurasi permission,ownership dan SELinux dns pada file /etc/named.conf dan folder /var/named :

```
[root@adhi named]# chgrp named -R /var/named/
[root@adhi named]# chown -v root:named /etc/named.conf
ownership of '/etc/named.conf' retained as root:named
[root@adhi named]# restorecon -rv /var/named/
[root@adhi named]# restorecon -rv /etc/named.conf
```

Cek file konfigurasi DNS dan ZONES file jika terjadi error / salah penulisan pada sintak.

```
[root@adhi named]# named-checkconf /etc/named.conf
[root@adhi named]# named-checkzone adhi.lan.id
/var/named/db.adhi.lan.id
zone adhi.lan.id/IN: loaded serial 9082017
OK
[root@adhi named]# named-checkzone adhi.lan.id
/var/named/10.168.192.db
zone adhi.lan.id/IN: loaded serial 9082017
OK
```

restart service :

```
[root@adhi named]# systemctl enable named
Created symlink from
/etc/systemd/system/multi-user.target.wants/named.service to
/usr/lib/systemd/system/named.service.
[root@adhi named]# systemctl restart named
```

c. Open Port

```
[root@adhi named]# firewall-cmd --permanent --add-port=53/tcp
success
[root@adhi named]# firewall-cmd --permanent --add-port=53/udp
success
```

```
[root@adhi named]# firewall-cmd --reload
success
```

d. Testing DNS Server

```
[root@adhi named]# dig adhi.lan.id

; <>> DiG 9.9.4-RedHat-9.9.4-50.el7_3.1 <>> adhi.lan.id
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 20762
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL:
2

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;adhi.lan.id.           IN      A

;; ANSWER SECTION:
adhi.lan.id.        604800  IN      A      192.168.10.99

;; AUTHORITY SECTION:
adhi.lan.id.        604800  IN      NS      ns1.adhi.lan.id.

;; ADDITIONAL SECTION:
ns1.adhi.lan.id.    604800  IN      A      192.168.10.99

;; Query time: 0 msec
;; SERVER: 192.168.10.99#53(192.168.10.99)
;; WHEN: Thu Aug 17 07:21:48 WIB 2017
;; MSG SIZE  rcvd: 90
```

4. Web Server

Untuk menampilkan halaman demi halaman dari situs yang kita miliki pada halaman web browser dan dapat diakses oleh orang banyak dibutukan sebuah web server. Di sisi klien sendiri dibutuhkan sebuah web browser seperti Iceweasel, Konqueror, Internet Explorer, Opera, Mozilla, Chrome dll.

a. Install web server Apache

```
[root@adhi ~]# yum install httpd  
[root@localhost ~]# systemctl enable httpd  
[root@localhost ~]# systemctl start httpd
```

b. Konfigurasi WebServer

```
[root@adhi ~]# nano /etc/httpd/conf/httpd.conf
```

isi konfigurasi :

```
#mengaktifkan mode rewrite  
#posisi line 151  
# AllowOverride controls what directives may be placed in .htaccess  
files.  
# It can be "All", "None", or any combination of the keywords:  
# Options FileInfo AuthConfig Limit  
#  
AllowOverride All  
  
#load index lebih cepat dengan yang pertama index.php  
#posisi line 164  
# DirectoryIndex: sets the file that Apache will serve if a directory  
# is requested.  
#  
<IfModule dir_module>  
    DirectoryIndex index.phpindex.html  
</IfModule>  
  
#tambahkan di line paling bawah untuk tempat virtual host  
IncludeOptional sites-enabled/*.conf
```

Buat script untuk mengaktifka a2ensite dan a2dissite untuk mendisable virtual host :

```
[root@adhi httpd]# nano /usr/local/bin/a2ensite
```

Isi script :

```
#!/bin/bash
if test -d /etc/httpd/sites-available && test -d /etc/httpd/sites-
enabled ; then
echo "-----"
else
mkdir /etc/httpd/sites-available
mkdir /etc/httpd/sites-enabled
fi

avail=/etc/httpd/sites-available/$1.conf
enabled=/etc/httpd/sites-enabled/
site=`ls /etc/httpd/sites-available/` 

if [ "$#" != "1" ]; then
        echo "Use script: a2ensite virtual_site"
        echo -e "\nAvailable virtual hosts:\n$site"
        exit 0
else

        if test -e $avail; then
                sudo ln -s $avail $enabled
        else

                echo -e "$avail virtual host does not exist! Please create one!\n$site"
                exit 0
        fi
        if test -e $enabled/$1.conf; then

                echo "Success!! Now restart Apache server: sudo systemctl restart
httpd"
        else
                echo -e "Virtual host $avail does not exist!\nPlease see available
virtual hosts:\n$site"
                exit 0
        fi
fi
```

script a2dissite :

```
[root@adhi httpd]#nano /usr/local/bin/a2dissite
```

Isi script :

```
#!/bin/bash
avail=/etc/httpd/sites-enabled/$1.conf
enabled=/etc/httpd/sites-enabled
site=`ls /etc/httpd/sites-enabled/` 

if [ "$#" != "1" ]; then
    echo "Use script: a2dissite virtual_site"
    echo -e "\nAvailable virtual hosts: \n$site"
    exit 0
else

    if test -e $avail; then
        sudo rm $avail
    else
        echo -e "$avail virtual host does not exist! Exiting!"
        exit 0
    fi

    if test -e $enabled/$1.conf; then
        echo "Error!! Could not remove $avail virtual host!"
    else
        echo -e "Success! $avail has been removed!\nPlease restart Apache:
        sudo systemctl restart httpd"
        exit 0
    fi
fi
fi
```

buat permission agar scrip bisa di eksekusi :

```
[root@adhi ~]# chmod +x /usr/local/bin/a2*
```

Buat lah folder untuk membuat virtual host:

```
[root@adhi ~]# cd /etc/httpd/
[root@adhi httpd]# mkdir sites-available sites-enable
```

c. Konfigurasi Virtual Host

```
[root@adhi httpd]# cd sites-available/  
[root@adhi sites-available]# nano adhi.lan.id.conf
```

Isi konfigurasi :

```
<VirtualHost *:80>  
    ServerName adhi.lan.id  
    ServerAlias www.adhi.lan.id  
    DocumentRoot "/var/www/html/adhi.lan.id/"  
    <Directory "/var/www/html/adhi.lan.id/">  
        Options Indexes FollowSymLinks MultiViews  
        # AllowOverride controls what directives may be placed  
        # in .htaccess files.  
        AllowOverride All  
        # Controls who can get stuff from this server file  
        Order allow,deny  
        Allow from all  
    </Directory>  
    <IfModule mpm_peruser_module>  
        ServerEnvironment apache apache  
    </IfModule>  
        ErrorLog /var/log/httpd/adhi.lan.id.log  
        CustomLog /var/log/httpd/adhi.lan.id-access.log combined  
</VirtualHost>
```

Catatan :

ServerName adalah nama domain yang di ruting ke web server yang di panggil

```
ServerName adhi.lan.id  
ServerAlias www.adhi.lan.id
```

AllowOverride adalah meredirect directory karena ini penting jika tidak di aktifkan menjadi ALL

```
AllowOverride All
```

DocumentRoot adalah tempat dimana php,css,html di letakan

```
DocumentRoot "/var/www/html/adhi.lan.id/"  
<Directory "/var/www/html/adhi.lan.id/">  
    Options Indexes FollowSymLinks MultiViews
```

Error Log adalah untuk melihat log web server yang error

```
ErrorLog /var/log/httpd/adhi.lan.id.log
```

CustomLog adalah melihat log siapa saja yang request ke server web

```
CustomLog /var/log/httpd/adhi.lan.id-access.log combined
```

Folder di /var/www/html/adhi.lan.id harus owner permissin apache apache karena supaya di kenali folder tersebut adalah owner web server apache

```
<IfModule mpm_peruser_module>
    ServerEnvironment apache apache
</IfModule>
```

Buat folder dan permission di folder /var/www/html/adhi.lan.id

```
[root@localhost ~]# cd /var/www/html/
[root@localhost html]# mkdir adhi.lan.id
[root@localhost adhi.lan.id]# nano index.html

Isi script :
<html>
<head>
<title>belajar linux server</title>
</head>
<body>

<p>Hello Areta ini virtual host pertama</p>

</body>
</html>

[root@localhost adhi.lan.id]# cd ..
[root@localhost html]# chown apache:apache -R adhi.lan.id/
```

Aktifkan Virtual host di server

```
[root@localhost html]# cd
[root@localhost ~]# a2ensite adhi.lan.id
-----
Success!! Now restart Apache server: sudo systemctl restart httpd
```

Restart service web server apache

```
[root@localhost ~]#systemctl restart httpd.service
```

d. Open Port

```
[root@localhost ~]#firewall-cmd --permanent --add-port=80/tcp
Success
[root@localhost ~]#firewall-cmd --reload
Success
```

e. Testing web server



f. Membuat Virtual Host ke dua

```
[root@localhost ~]#cd /etc/httpd/sites-available/  
[root@localhost sites-available]#nano portal.adhi.lan.id
```

Isi konfigurasi :

```
<VirtualHost *:80>  
    ServerName portal.adhi.lan.id  
    DocumentRoot "/var/www/html/portal/"  
<Directory "/var/www/html/portal/">  
    Options Indexes FollowSymLinks MultiViews  
    # AllowOverride controls what directives may be placed in .htaccess  
    files.  
    AllowOverride All  
    # Controls who can get stuff from this server file  
    Order allow,deny  
    Allow from all  
</Directory>  
<IfModule mpm_peruser_module>  
    ServerEnvironment apache apache  
</IfModule>  
    ErrorLog /var/log/httpd/portal.adhi.lan.id.log  
    CustomLog /var/log/httpd/portal.adhi.lan.id-access.log combined  
</VirtualHost>
```

Buat folder dan buat permission web server apache

```
[root@localhost sites-available]# cd /var/www/html/  
[root@localhost html]# mkdir portal  
[root@localhost html]# cd portal/  
[root@localhost portal]# nano index.html  
[root@localhost portal]# cd ..  
[root@localhost html]# chown apache:apache -R portal/  
[root@localhost html]# cd
```

Restart service web server

```
[root@localhost ~]# a2ensite portal.adhi.lan.id  
-----  
Success!! Now restart Apache server: sudo systemctl restart httpd  
[root@localhost ~]# systemctl restart httpd
```

5. Install dependsis php dan

```
[root@localhost ~]# rpm --import /etc/pki/rpm-gpg/RPM-GPG-KEY*
[root@localhost ~]# yum -y install epel-release
[root@localhost ~]# yum update
[root@localhost ~]# rpm -Uvh http://rpms.remirepo.net/enterprise/remi-release-.rpm
[root@localhost ~]# yum -y install yum-utils
[root@localhost ~]# yum-config-manager --enable remi-php71
[root@localhost ~]# yum update
[root@localhost ~]# yum -y install php php-opcache
```

Restart Service web server

```
[root@localhost ~]# systemctl restart httpd
```

Testing php information module dan version

```
[root@adhi ~]# nano /var/www/html/info.php
Isi scrip phpinfo

<?php
phpinfo();
?>
```

Testing di browser

The screenshot shows a web browser window displaying the output of a PHP info script. The title bar says "192.168.20.99/info.php". The page has a blue header with "PHP Version 7.1.8" and the PHP logo. Below the header is a table of PHP configuration settings.

System	Linux adhi.local 3.10.0-514.26.2.el7.x86_64 #1 SMP Tue Jul 4 15:04:05 UTC 2017 x86_64
Build Date	Aug 2 2017 12:14:01
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc
Loaded Configuration File	/etc/php.ini
Scan this dir for additional .ini files	/etc/php.d
Additional .ini files parsed	/etc/php.d/10-opcache.ini, /etc/php.d/20-bcmath.ini, /etc/php.d/20-bz2.ini, /etc/php.d/20-calendar.ini, /etc/php.d/20-ctype.ini, /etc/php.d/20-curl.ini, /etc/php.d/20-dom.ini, /etc/php.d/20-exif.ini, /etc/php.d/20-fileno.ini, /etc/php.d/20-ftp.ini, /etc/php.d/20-gd.ini, /etc/php.d/20-gettext.ini, /etc/php.d/20-iconv.ini, /etc/php.d/20-json.ini, /etc/php.d/20-mbstring.ini, /etc/php.d/20-mysqlind.ini, /etc/php.d/20-pdo.ini, /etc/php.d/20-phar.ini, /etc/php.d/20-posix.ini, /etc/php.d/20-shmop.ini, /etc/php.d/20-simplexml.ini, /etc/php.d/20-sockets.ini, /etc/php.d/20-sysvmsg.ini, /etc/php.d/20-sysvsem.ini, /etc/php.d/20-sysvshm.ini, /etc/php.d/20-tokenizer.ini, /etc/php.d/20-xml.ini, /etc/php.d/20-xsl.ini, /etc/php.d/30-mysqli.ini, /etc/php.d/30-pdo_mysql.ini, /etc/php.d/20-xmlwriter.ini, /etc/php.d/20-xsl.ini, /etc/php.d/30-mysqli.ini, /etc/php.d/30-pdo_mysql.ini, /etc/php.d/30-pdo_sqlite.ini, /etc/php.d/30-wddx.ini, /etc/php.d/30-xmldbreader.ini, /etc/php.d/40-zip.ini
PHP API	20160303
PHP Extension	20160303
Zend Extension	320160303
Zend Extension Build	API320160303.NTS
PHP Extension Build	API20160303.NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	enabled
Zend Memory Manager	enabled
Zend Multibyte Support	provided by mbstring
IPv6 Support	enabled

6. Install Database Server

a. Install database mysql server

```
[root@adhi ~]#yum install mariadb
```

b. Konfigurasi database

```
[root@adhi ~]# systemctl enable mariadb
[root@adhi ~]# systemctl restart mariadb
[root@adhi ~]# 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):
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:
Re-enter new password:
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]
... 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]
- 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]
... 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!

c. Install PhpMyadmin

```
[root@adhi ~]# yum install phpmyadmin
```

i. Phpmyadmin bisa di akses publik

```
[root@adhi ~]# nano /etc/httpd/conf.d/phpMyAdmin.conf
```

Isi konfigurasi :

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

<IfModule mod_authz_core.c>
    # Apache 2.4
<RequireAny>
    #Require ip 127.0.0.1
    #Require ip ::1
    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 All
    Allow from ::1
</IfModule>
</Directory>
```

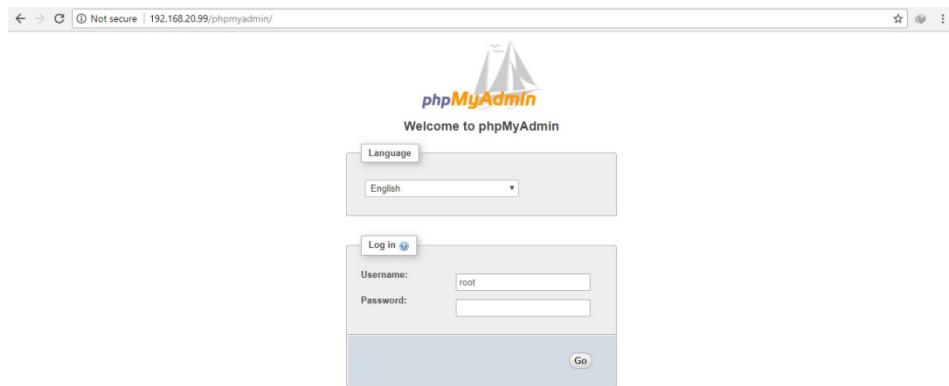
Restart service httpd

```
[root@adhi ~]# systemctl restart httpd
```

buat database

```
# mysql -u root -p
Mysql >CREATE DATABASE wordpress_db DEFAULT CHARACTER SET 'utf8';
Mysql >CREATE USER 'wordpress'@'localhost' IDENTIFIED BY 'sukses889';
Mysql > GRANT ALL PRIVILEGES ON wordpress_db.* TO
'wordpress'@'localhost';
Mysql > FLUSH PRIVILEGES;
Mysql > exit
```


Test di browser sudah bisa akses phpmyadmin : <http://hostname or ip address/phpmyadmin>



7. Instalall Aplikasi PHP

a. Install PopoJiCMS

Server Information		Current Setting	Required Setting
Apache			
mod_rewrite	ON	ON	
PHP Setting			
short_open_tag	OFF	ON	
PHP Extension			
php_curl	ON	ON	
php_gd2	ON	ON	
php_mbstring	ON	ON	
php_mysqli	ON	ON	
php_pdo_mysql	ON	ON	
php_fileinfo	ON	ON	
Directory			
/po-includes/core/	NO	YES	

Note :

Konfigurasi php.ini → aktifkan short_open_tag dan ownership ke apache pada folder popoji

Konfigurasi php.ini

```
[root@adhi ~]# nano /etc/php.ini
Isi konfigurasi :
//konfigurasi jadikan On yang tadi nya Off
short_open_tag = On
```

Disable Selinux

#vim /etc/selinux/config

```
# This file controls the state of SELinux on the system.  
# SELINUX= can take one of these three values:  
#   enforcing - SELinux security policy is enforced.  
#   permissive - SELinux prints warnings instead of enforcing.  
#   disabled - No SELinux policy is loaded.  
#SELINUX=enforcing //silahkan di berikan Command dengan tanda (#)  
SELINUX=disabled //silahkan Tuliskan Script berikut  
# SELINUXTYPE= can take one of these two values:  
#   targeted - Targeted processes are protected,  
#   minimum - Modification of targeted policy. Only selected processes are pr$  
#   mls - Multi Level Security protection.  
SELINUXTYPE=targeted
```

b. Install PHP CRUD

Upload file aplikasi CRUD ke /var/www/html/aplikasi

c. Install PHP dengan CodeIgniter

d. Install Owncloud

8. Install FTP

a. Install ftp server

b. Testing ftp

9. Rsync

a. Install rsync

b. Konfigurasi Rsync

c. Testing

10. Cron Server

d. Install Cron Server

e. Konfigurasi Cron

11. Firewall