

COMPUTER NETWORK MANAGEMENT

Week - 5

HTTP Server and Local Domain Name System (Local DNS)

Universitas Kristen Wira Wacana Sumba
Lecturer - Fajar Hariadi

Contents

- 1 **HTTP Server**
- 2 **Implementasi HTTP Server**
- 3 **DNS Server**
- 4 **Implementasi DNS Lokal**

01

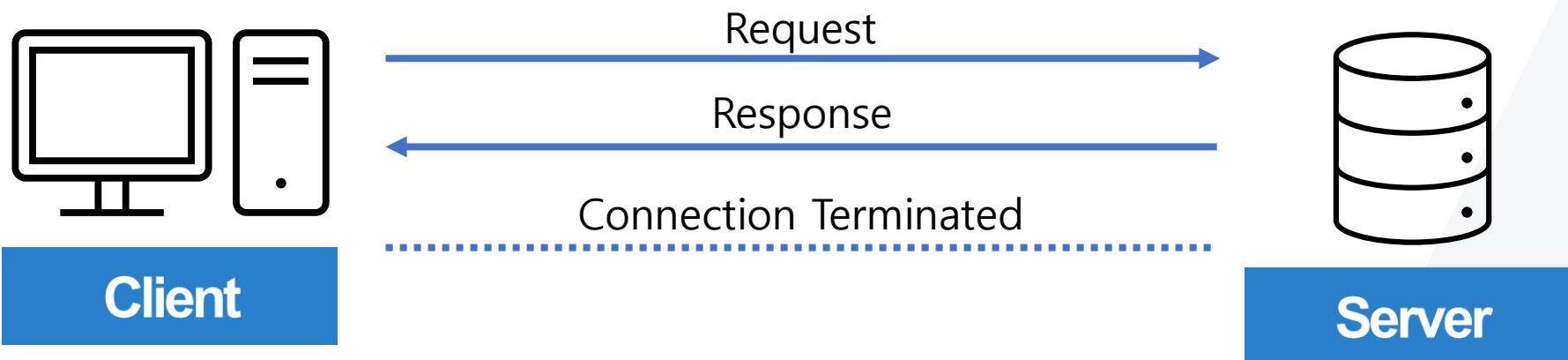
HTTP Server

Definisi HTTP

- **HTTP** merupakan singkatan dari HyperText Transfer Protocol. Sebuah protokol yang bekerja pada Layer 7 OSI layer – Application layer yang dibuat oleh Tim Berner.
- **Hypertext** itu sendiri merupakan format text khusus dengan penulisan standar di seluruh dunia yang dikenal dengan HyperText Markup Language (HTML)
- **HTTP dan HTML** dibuat bersamaan untuk menghasilkan halaman web interaktif yang merupakan asal dari world wide web (www) dimana protokol ini merupakan salah satu protokol yang paling sering digunakan oleh orang untuk mengakses konten di internet.

Koneksi HTTP

- HTTP merupakan protokol yang bekerja secara client-server
- Client meminta suatu halaman web (html) ke server, kemudian server merespons permintaan client dengan mengirimkan halaman yang diminta, setelah selesai dikirimkan, maka koneksi antara client server akan ditutup
- Apabila client membutuhkan halaman lainnya, maka proses akan dimulai dari awal kembali dengan membuat koneksi dan client mengirimkan permintaan lagi ke server



HTTP Request

- HTTP Request merupakan permintaan informasi yang pada umumnya dibuat oleh browser client kepada server untuk memuat halaman web sesuai dengan permintaan dari user client
- HTTP Request berisi beberapa informasi tambahan dari client kepada server dimana setiap request akan berisi bagian berikut :
 - HTTP Version
 - URL
 - HTTP Method
 - HTTP Request Headers
 - HTTP Body

HTTP Request - Cont

➤ HTTP Version

Versi http yang digunakan (1, 2, 3)

➤ URL

alamat yang menunjuk kepada informasi atau lokasi halaman web di server yang diminta oleh client

➤ HTTP Method

jenis metode yang diminta oleh client agar dilaksanakan oleh server. Jenis method yang ada adalah :

HTTP Request - Cont

- **GET**
metode yang digunakan untuk menerima data dari server :
- **POST**
Metode yang digunakan untuk menambahkan data baru ke server :
- **PUT**
Metode yang digunakan untuk memodifikasi data yang telah ada di server dengan cara mengirimkan keseluruhan isi data terbaru
- **PATCH**
Metode yang digunakan untuk memodifikasi data yang telah ada di server dengan cara mengirimkan bagian yang hendak diperbaharui saja
- **DELETE**
Metode yang digunakan untuk menghapus data yang ada di server

HTTP Request - Cont

- **HTTP Request Headers** – bagian ini berisi data jenis browser yang digunakan oleh client, jenis data yang diinginkan oleh client. Bagian ini juga dapat berisi cookies yang merupakan informasi terkait data sebelumnya yang telah dikirimkan server ketika menangani permintaan sebelumnya
- **HTTP Body** – bagian ini merupakan bagian yang sifatnya opsional berisi informasi yang dibutuhkan oleh server ketika menangani permintaan client seperti usernam dan password client, data atau file yang diupload dari client ke server

HTTP Request - Contoh



HTTP Response

- ▶ HTTP Response merupakan pesan yang dikirimkan oleh server sesuai dengan permintaan dari client. Pesan ini merupakan balasan terhadap http request. Informasi yang terdapat pada pesan ini terdiri dari:
 - HTTP Status Code – berisi kode angka yang merupakan indikasi respon yang diberikan server seperti indikasi sukses, informasi tambahan, redirect, atau error pada sisi server maupun client
 - HTTP Response header – berisi informasi terkait server dan informasi resource yang dikirimkan
 - HTTP Body – bagian ini sifatnya opsional, jika request berhasil maka akan berisi data dalam bentuk HTML untuk ditampilkan di halaman browser yang dimiliki oleh client

HTTP Request - Cont

```
HTTP/1.1 200 OK
Date: Sat, 20 Apr 2024 16:27:18 GMT
Server: Apache
Last-Modified: Mon, 18 May 1992 13:38:38 GMT
ETag: "47d-2824824ad1380"
Accept-Ranges: bytes
Content-Length: 1149
Connection: close
Content-Type: text/html
```

HTTP Status Code

HTTP Response Header

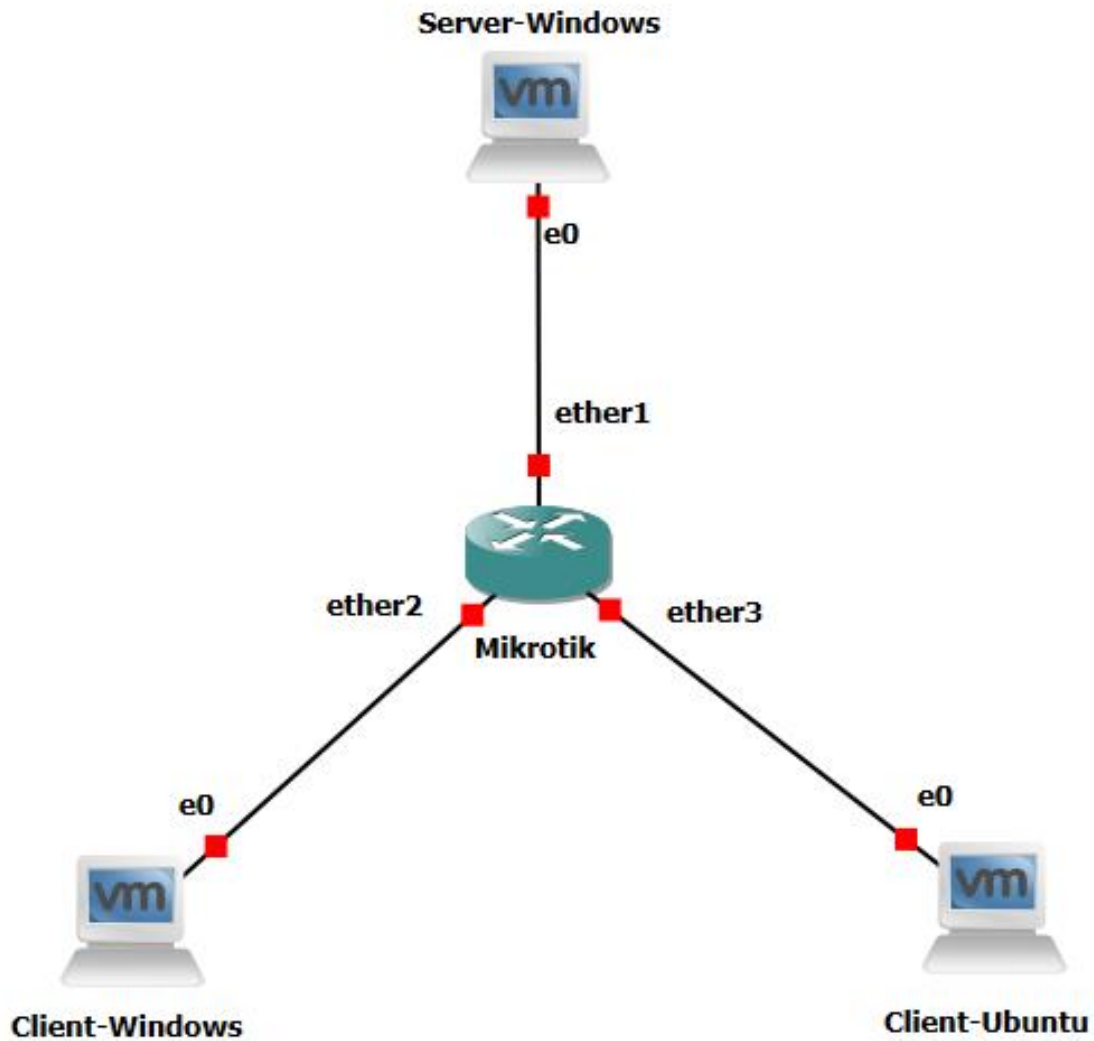
```
<TITLE>What is Hypertext?</TITLE>
<NEXTID 20>
<H1>What is HyperText</H1>Hypertext is text which is not constrained to be linear.<P>
Hypertext is text which contains <A NAME=0 HREF=Terms.html#link>links</A> to other texts. The term was
coined by <A NAME=1 HREF=Xanadu.html#Nelson>Ted Nelson</A> around 1965 (see <A NAME=12 HREF=../History.html>History</A> ).<P>
HyperMedia is a term used for hypertext which is not constrained to
be text: it can include graphics, video and <A NAME=9 HREF=Talks/YesWeCan.snd>sound</A> , for example. Apparently
Ted Nelson was the first to use this term too.<P>
Hypertext and HyperMedia are concepts, not products.<P>
See also:
<UL>
<LI><A NAME=2 HREF=Terms.html>A list of terms</A> used in hypertext litterature.
<LI><A NAME=19 HREF=../Conferences/Overview.html>Conferences</A>
<LI><A NAME=7 HREF=../Products/Overview.html>Commercial (and academic) products</A>
<LI>A newsgroup on hypertext, <A NAME=5 HREF=news:alt.hypertext>"alt.hypertext"</A> .
<LI><A NAME=4 HREF=TheProject.html>WorldWideWeb is a project</A> which uses hypertext concepts.
<LI><A NAME=10 HREF=../Standards/Overview.html>Standards</A> .</A>
</UL>
```

HTTP Body

02

Implementasi HTTP Server

Topologi



Konfigurasi IP Address

Perangkat	Interface	IP Address
Mikrotik	ether1	192.168.1.1/24
Mikrotik	ether2	192.168.2.1/24
Mikrotik	ether3	192.168.3.1/24
Server-Windows	e0	192.168.1.2/24
Client-Windows	e0	192.168.2.2/24
Client-Ubuntu	e0	192.168.3.2/24

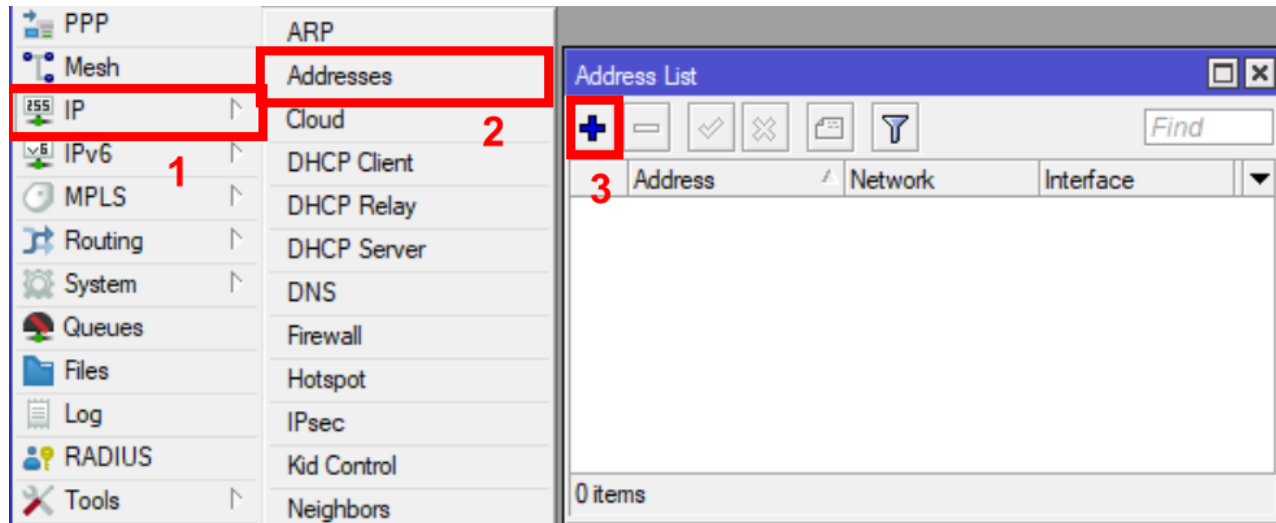
Mikrotik

Konfigurasi IP Address

Perangkat	Interface	IP Address
Mikrotik	ether1	192.168.1.1/24
Mikrotik	ether2	192.168.2.1/24
Mikrotik	ether3	192.168.3.1/24

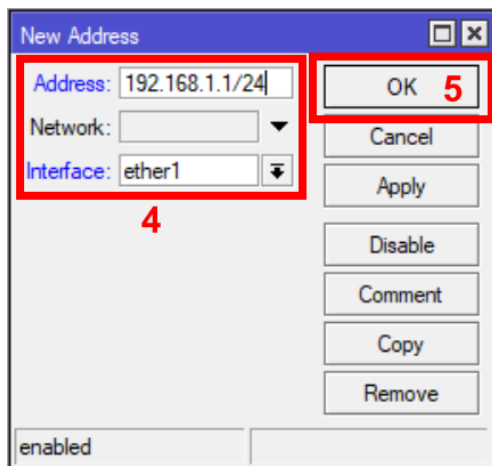
Akses mikrotik pakai Winbox pilih menu IP → **Addresses**

Akan muncul windows baru pilih icon Tambah (+)



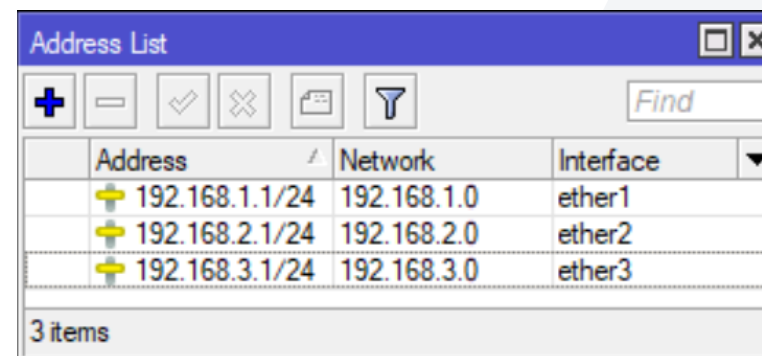
Buat untuk setiap interface

Hasilnya akan seperti ini



Masukkan Address dan interface sesuai tabel

Lalu pilih **OK**



Server - XAMPP

← → ↻ [apachefriends.org/download.html](https://www.apachefriends.org/download.html) ☆

Apache Friends

Download

Hosting

Community

About

Search..

Search

🇺🇸 EN

Download

XAMPP is an easy to install Apache distribution containing MariaDB, PHP, and Perl. Just download and start the installer. It's that easy.

XAMPP for **Windows** 8.0.30, 8.1.25 & 8.2.12

Version	Checksum	Size
8.0.30 / PHP 8.0.30 What's Included?	md5 sha1 Download (64 bit)	144 Mb
8.1.25 / PHP 8.1.25 What's Included?	md5 sha1 Download (64 bit)	148 Mb
8.2.12 / PHP 8.2.12 What's Included?	md5 sha1 Download (64 bit)	149 Mb

[Requirements](#) [More Downloads »](#)

Windows XP or 2003 are not supported. You can download a compatible version of XAMPP for these platforms [here](#).

XAMPP for **Linux** 8.0.30, 8.1.25 & 8.2.12

Version	Checksum	Size
8.0.30 / PHP 8.0.30 What's Included?	md5 sha1 Download (64 bit)	151 Mb

Documentation/FAQs

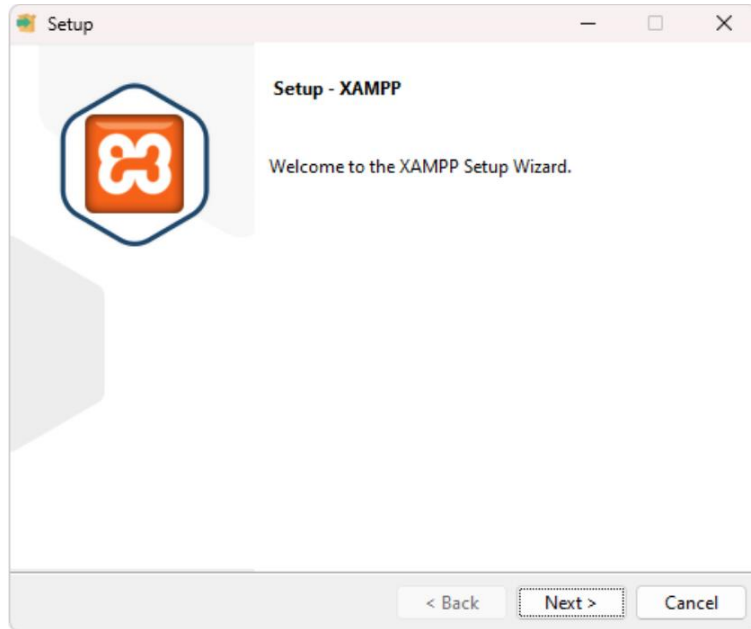
There is no real manual or handbook for XAMPP. We wrote the documentation in the form of FAQs. Have a burning question that's not answered here? Try the [Forums](#) or [Stack Overflow](#).

- [Linux FAQs](#)
- [Windows FAQs](#)
- [OS X FAQs](#)

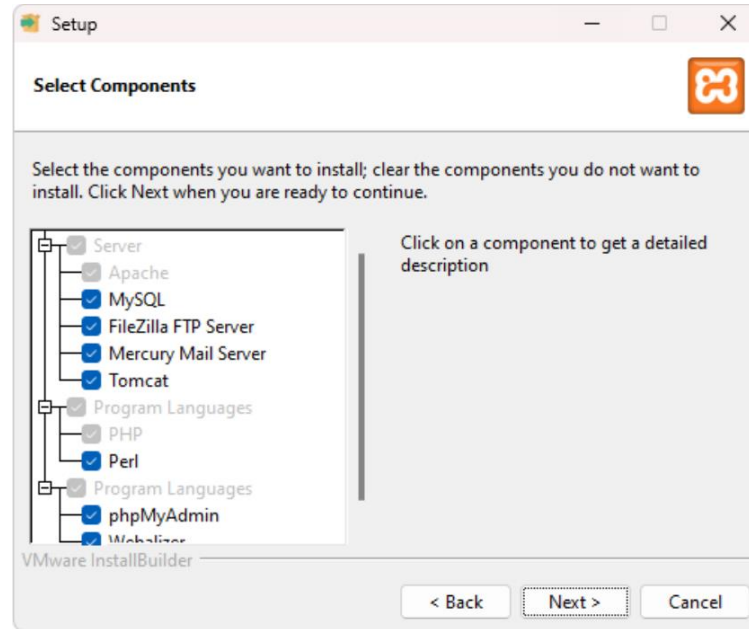
Download XAMPP di website berikut:

<https://www.apachefriends.org/download.html>

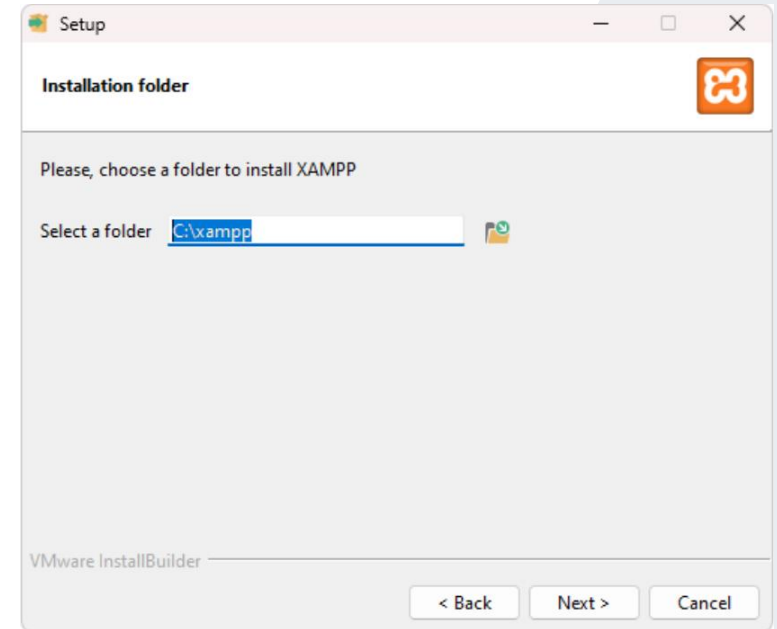
Server – Instalasi XAMPP



Pilih Next

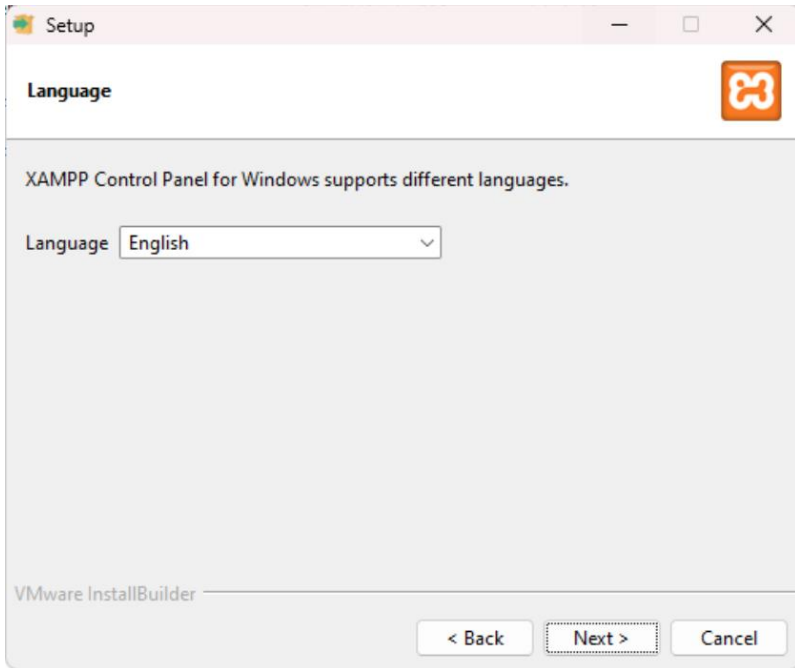


Pilih Next

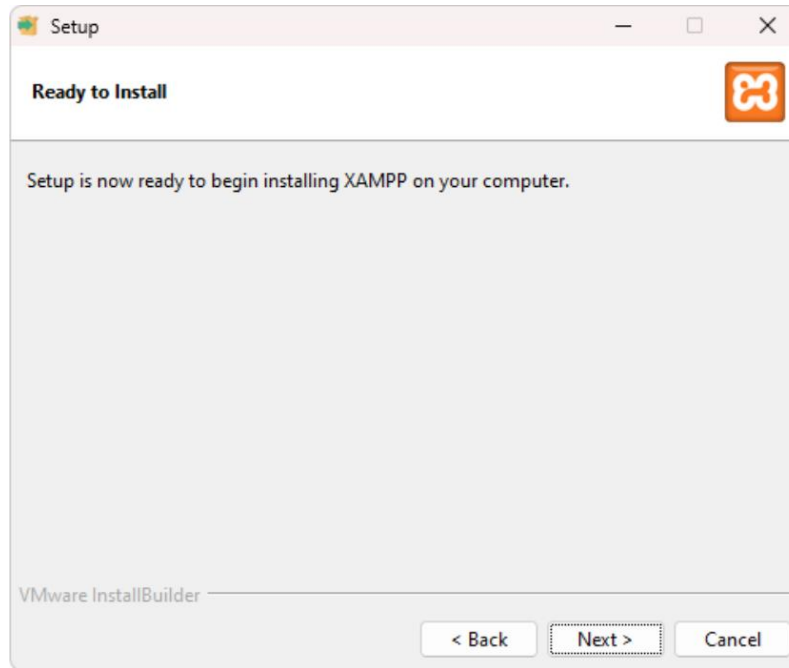


Pilih Next

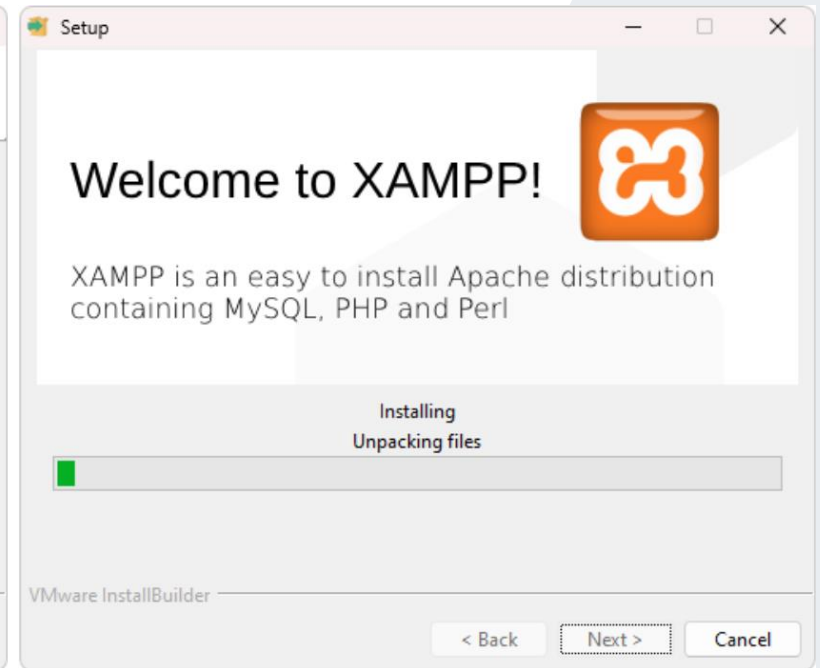
Server – Instalasi XAMPP



Pilih Next



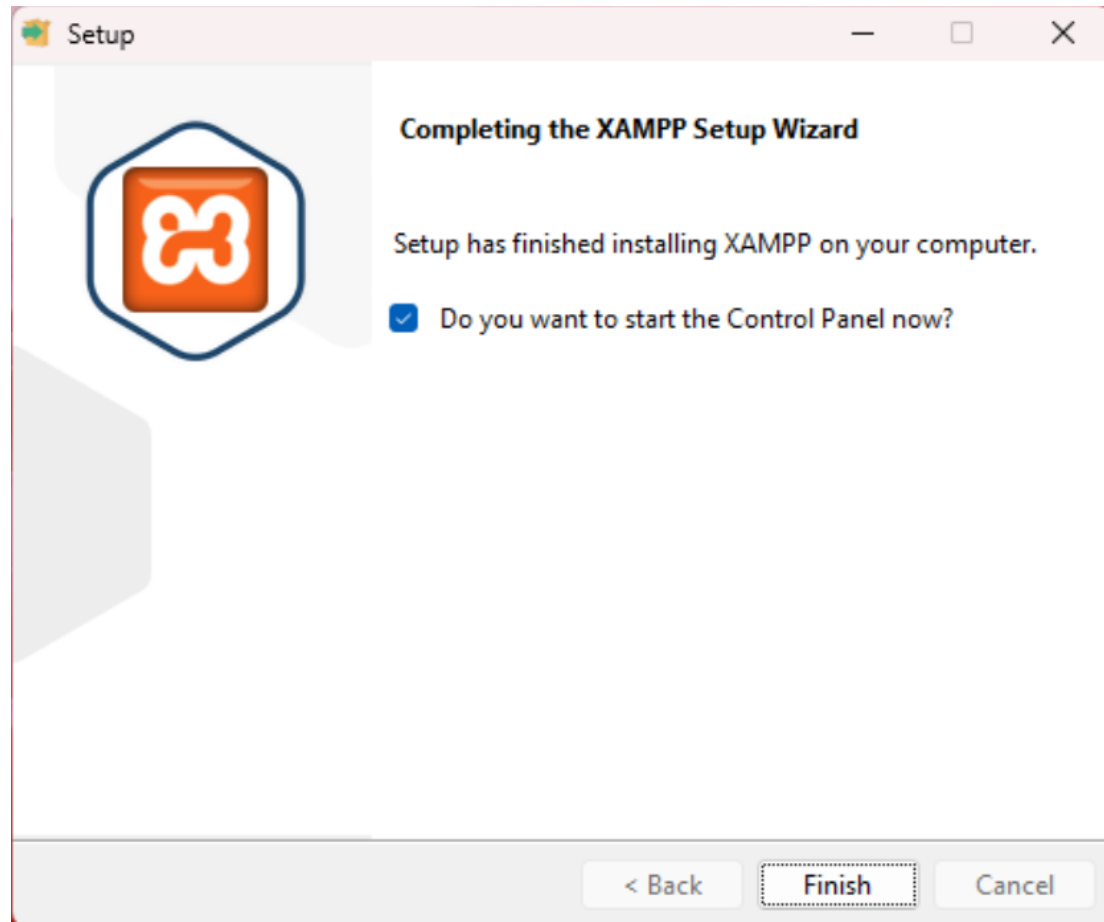
Pilih Next



Pilih Next

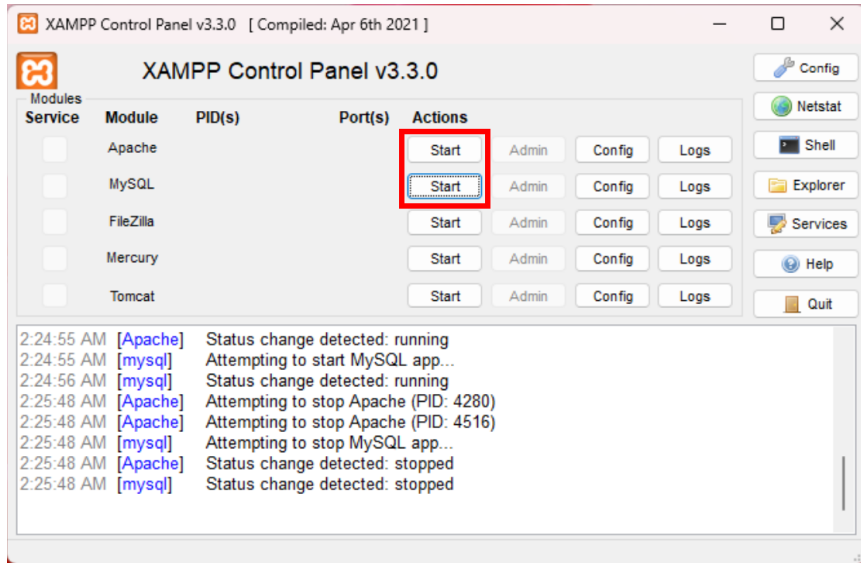
Server – Instalasi XAMPP

Tunggu Proses instalasi sampai selesai



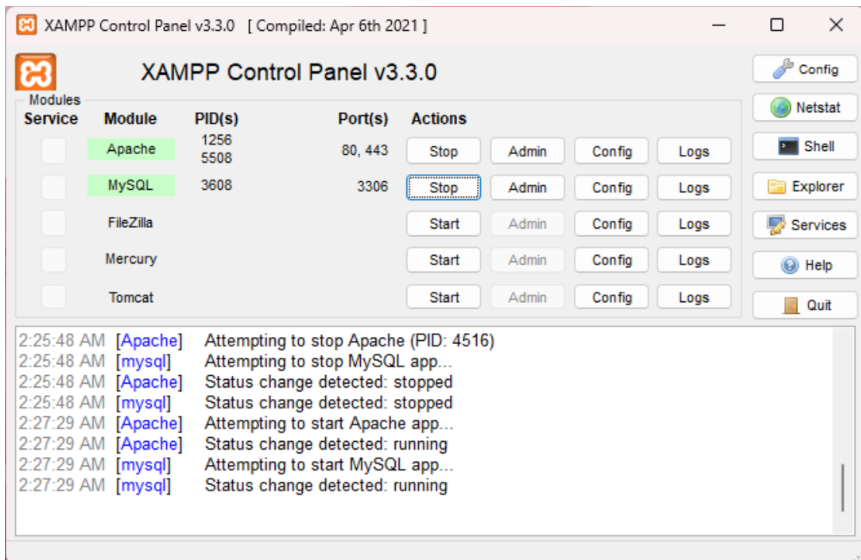
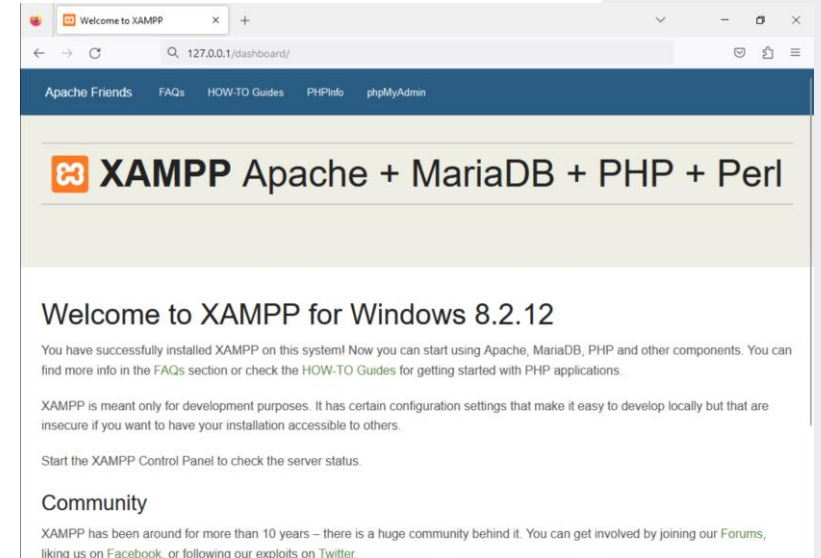
Pilih Finish

Server – Menjalankan Server



Buka XAMPP Control Panel Start Apache dan MySQL

Buka browser dan ketik alamat ip 127.0.0.1 di url untuk mencoba mengakses server



Pastikan Apache dan MySQL running

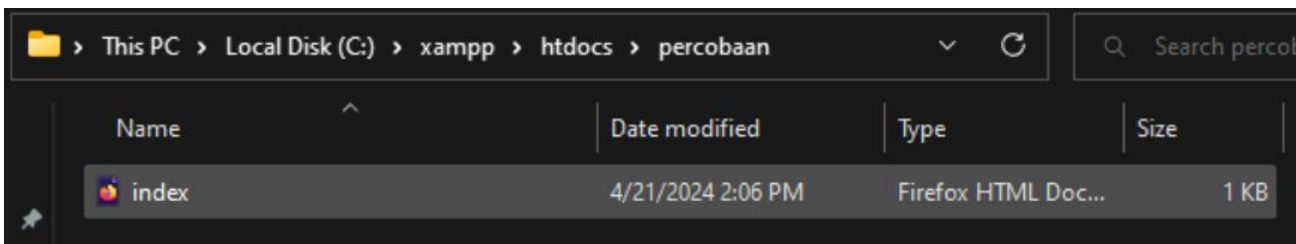
Diindikasikan dengan tulisan menjadi berlatar belakang warna hijau

Jika muncul halaman seperti gambar maka server sudah berjalan dengan baik

Server – Halaman Web

Buat sebuah file html pada directory
C:\xampp\htdocs\percobaan

```
<h1 style="text-align: center;">Halaman Web  
Server</h1>  
  
<p>Ini merupakan contoh halaman web yang  
diletakkan di sebuah server HTTP dan diakses  
menggunakan LAN</p>
```



Buka browser dan ketik alamat ip
127.0.0.1/percobaan di url untuk mencoba
halaman web yang dibuat

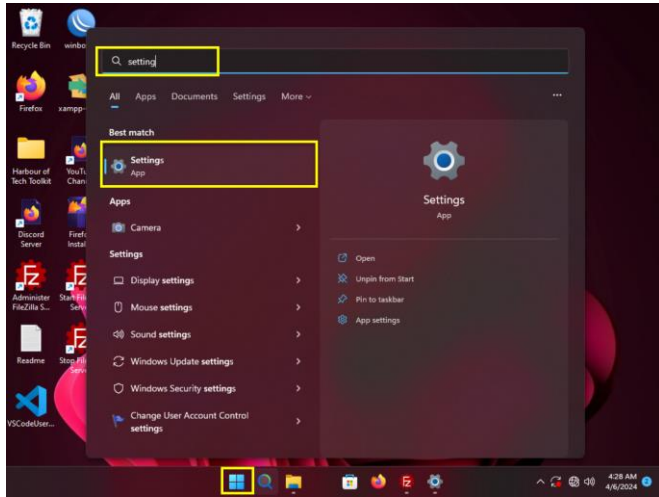


Jika muncul halaman seperti gambar maka server
dan halaman web sudah berjalan dengan baik

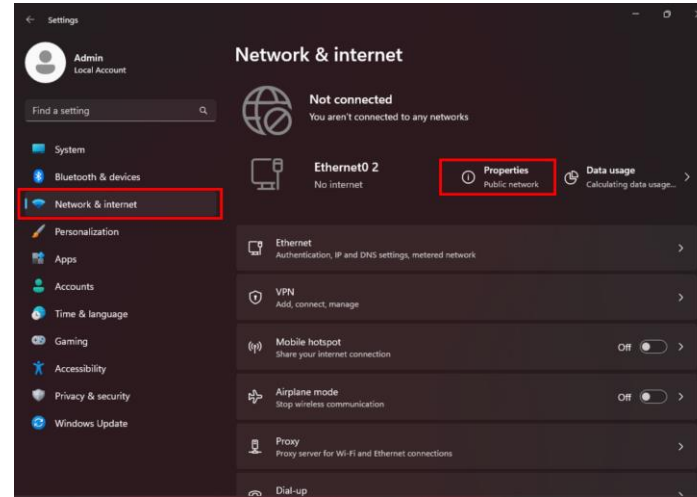
Server – IP Address

Konfigurasi IP Address

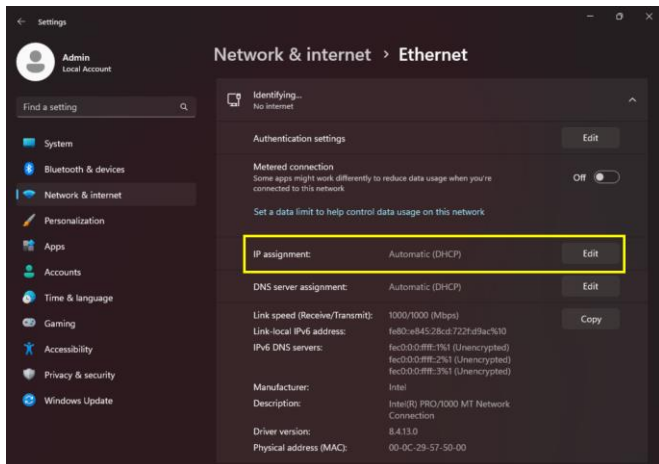
Perangkat	Interface	IP Address
Server-Windows	e0	192.168.1.2/24



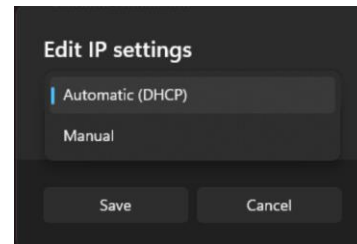
Masuk ke settings



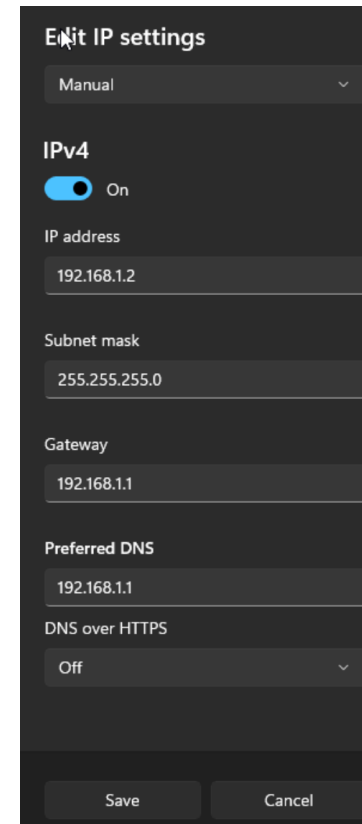
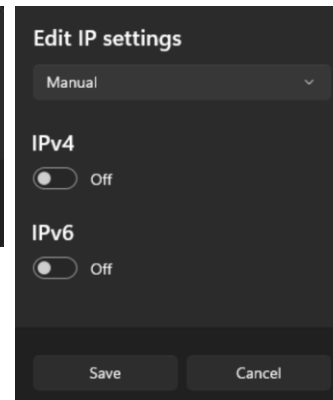
Pilih network & internet lalu properties



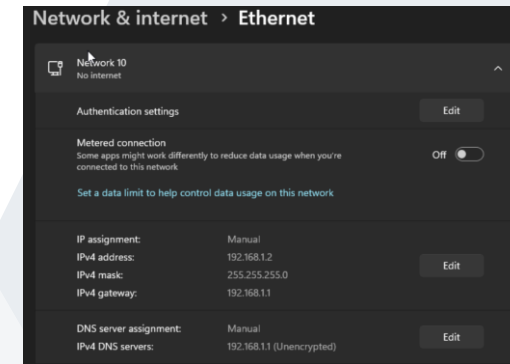
Pilih Edit



Pilih Manual, lalu hidupkan IPv4



Setting IP Address



Hasil Konfigurasi

Client - Windows

Edit IP settings

Manual

IPv4

On

IP address

192.168.2.2

Subnet mask

255.255.255.0

Gateway

192.168.2.1

Preferred DNS

192.168.1.1

DNS over HTTPS

Off

Save

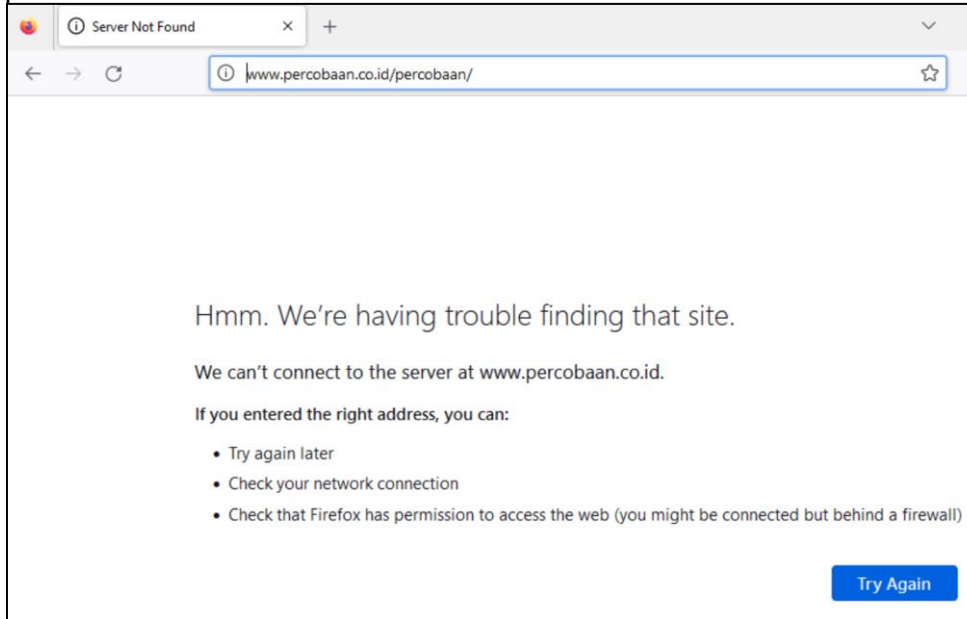
Cancel

Setting IP Address

Buka browser dan akses :
<http://192.168.1.2/percobaan/>



Buka browser dan akses :
<http://www.percobaan.co.id/percobaan>



Konfigurasi IP Address

Perangkat	Interface	IP Address
Client-Windows	e0	192.168.2.2/24

Sampai dengan tahap ini artinya server hanya bisa diakses menggunakan ip addressnya saja tanpa bisa diakses menggunakan URL

Client - Ubuntu

Konfigurasi IP Address

Perangkat	Interface	IP Address
Client-Windows	e0	192.168.2.2/24

Cancel **Wired** Apply

Details Identity **IPv4** IPv6 Security

IPv4 Method

Automatic (DHCP) Link-Local Only

Manual Disable

Shared to other computers

Addresses

Address	Netmask	Gateway	
192.168.3.2	255.255.255.0	192.168.3.1	

DNS Automatic

192.168.1.1

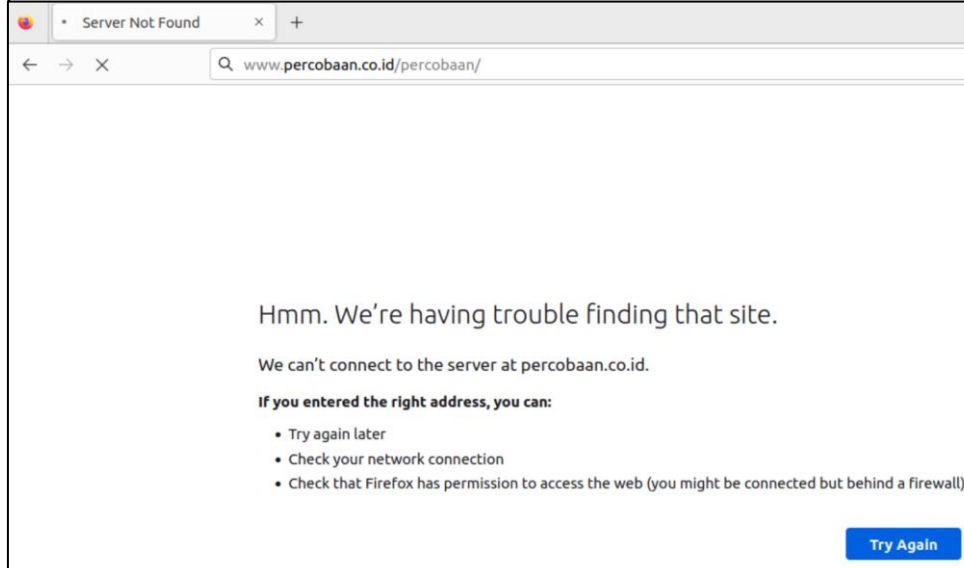
Separate IP addresses with commas

Setting IP Address

Buka browser dan akses :
<http://192.168.1.2/percobaan/>



Buka browser dan akses :
<http://www.percobaan.co.id/percobaan>



Sampai dengan tahap ini artinya server hanya bisa diakses menggunakan ip addressnya saja tanpa bisa diakses menggunakan URL

03

DNS Server

Domain Name System

- Setiap komputer dalam jaringan diidentifikasi berdasarkan ip address, namun untuk mengingat setiap ip address dalam bentuk angka sangat sulit.
- Karena manusia lebih mudah mengingat kata-kata dibanding angka, sehingga diperlukan sebuah service untuk mengubah ip address menjadi kata-kata atau domain name sebuah website.
- DNS hadir sebagai service/layanan untuk melakukan translasi dari ip address menjadi domain name ataupun sebaliknya, dari domain name menjadi ip address



04

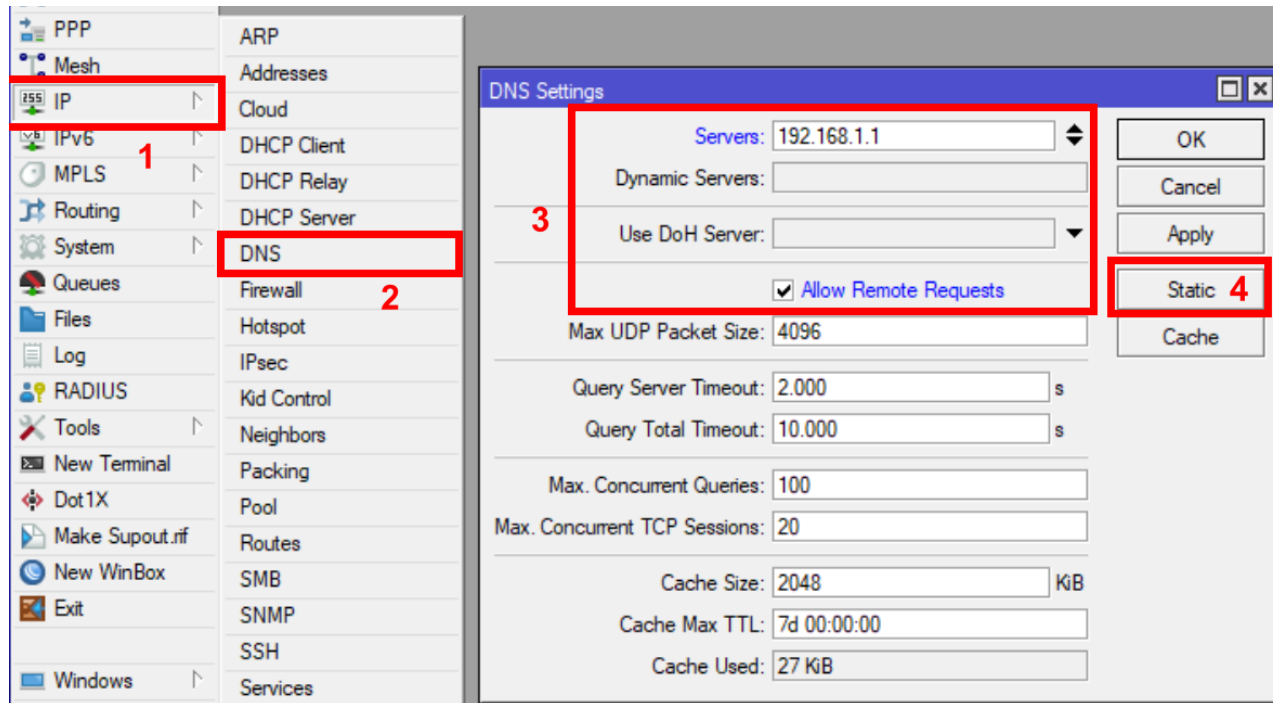
Implementasi DNS Lokal

Implementasi Lokal DNS

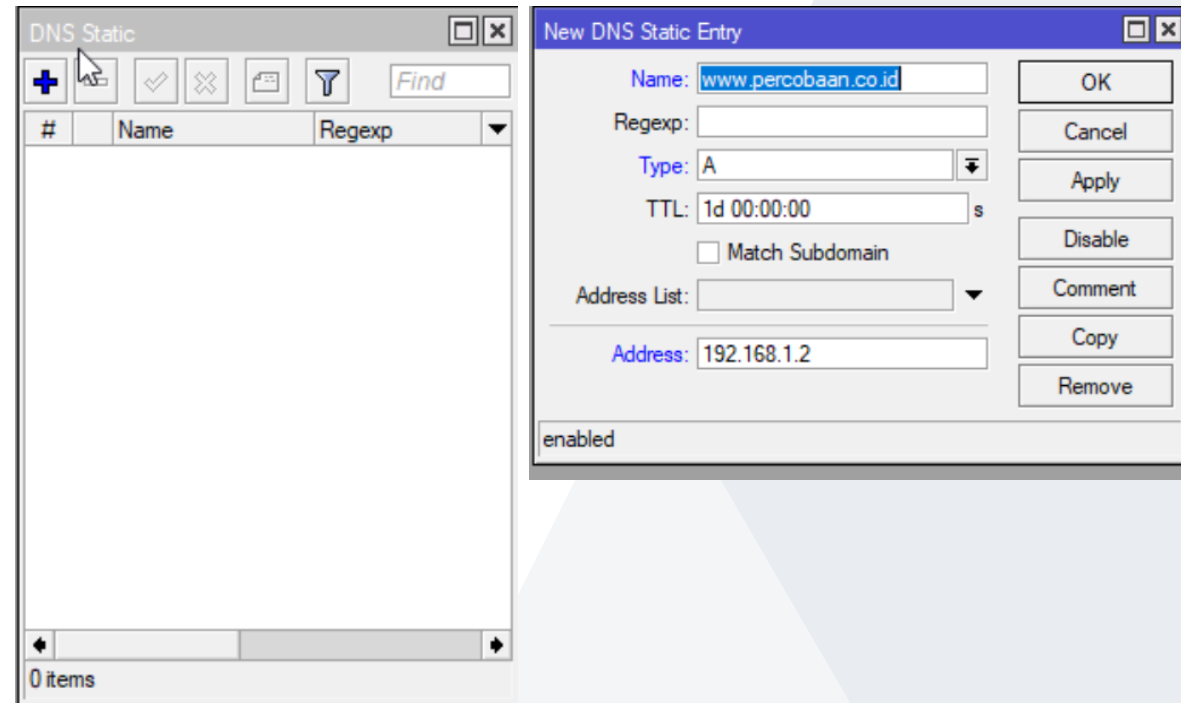
K

Akses router, ke menu **IP** → **DNS**

pada windows baru masukkan ip dns server **192.168.1.1**,
centang **allow remote request**



Tambahkan DNS baru dengan menekan tombol tambah, masukkan name dengan url yang diinginkan dan address dari web server/http server yang ada



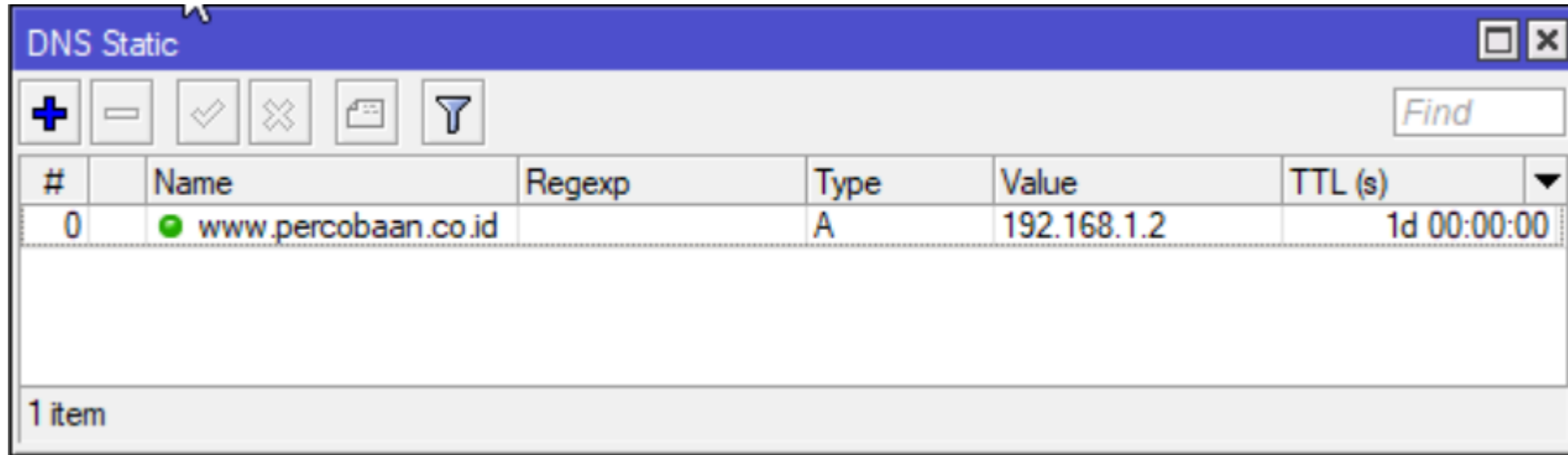
Lalu klik **OK**

Klik **Static**, untuk menambahkan DNS Lokal statis/manual

Implementasi Lokal DNS

K

Hasil DNS Statis akan terlihat seperti gambar berikut:



The screenshot shows the 'DNS Static' window in Windows. The window title is 'DNS Static'. Below the title bar is a toolbar with icons for adding (+), deleting (-), checking (✓), unchecking (✗), refreshing (refresh icon), and filtering (funnel icon). To the right of the toolbar is a search box labeled 'Find'. Below the toolbar is a table with the following columns: '#', 'Name', 'Regexp', 'Type', 'Value', and 'TTL (s)'. The table contains one row with the following data: '# 0', 'Name www.percobaan.co.id', 'Regexp', 'Type A', 'Value 192.168.1.2', and 'TTL (s) 1d 00:00:00'. At the bottom left of the window, it says '1 item'.

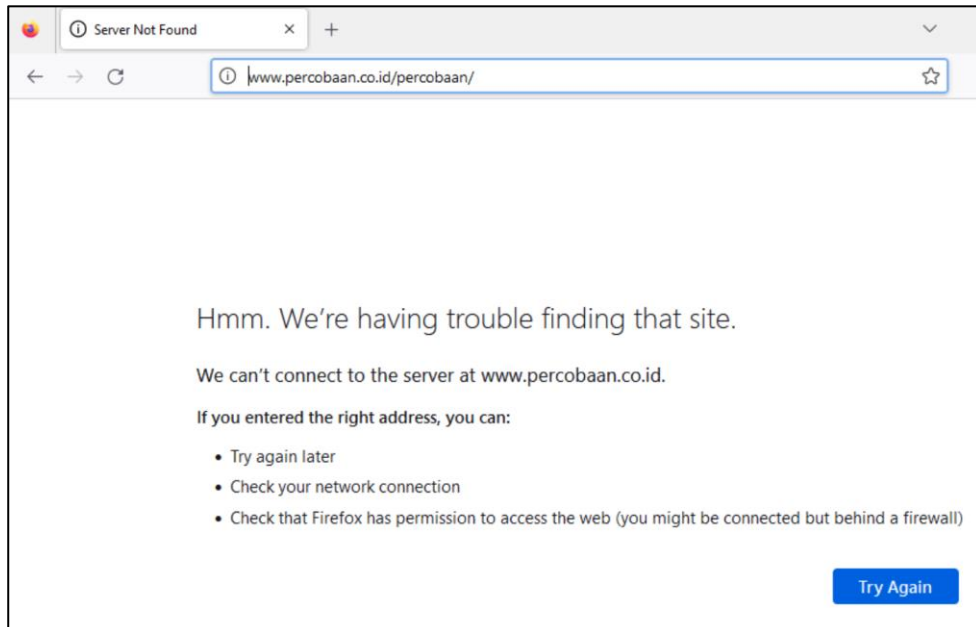
#	Name	Regexp	Type	Value	TTL (s)
0	www.percobaan.co.id		A	192.168.1.2	1d 00:00:00

Uji Coba DNS – Client Windows

K

Buka browser dan akses : <http://www.percobaan.co.id/percobaan/>

Sebelum ada DNS Statis



Setelah ada DNS Statis



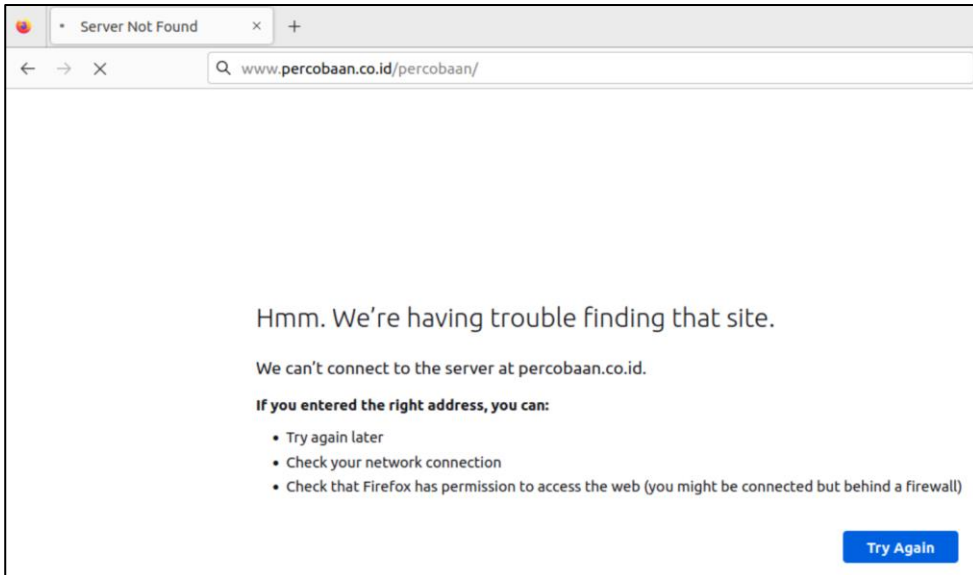
Hal ini membuktikan dengan adanya DNS Lokal statis yang telah dibuat, client jadi dapat mengakses web server menggunakan URL sesuai dengan pengaturan pada DNS Statis

Uji Coba DNS – Client Ubuntu

K

Buka browser dan akses : <http://www.percobaan.co.id/percobaan/>

Sebelum ada DNS Statis



Setelah ada DNS Statis



Hal ini membuktikan dengan adanya DNS Lokal statis yang telah dibuat, client jadi dapat mengakses web server menggunakan URL sesuai dengan pengaturan pada DNS Statis

05

Kesimpulan

Kesimpulan

- HTTP Server merupakan sebuah web server untuk menyediakan halaman web.
- HTTP Server bekerja dengan konsep client server, permintaan dimulai dari client dan akan direspons oleh server dengan mengirimkan data dalam bentuk html
- HTTP Server dapat diakses oleh komputer lain dalam jaringan menggunakan IP Address dari server tersebut

Kesimpulan

- DNS Server merupakan sebuah server yang menyediakan service untuk mengubah ip address menjadi url atau domain name dan sebaliknya
- DNS Server mempermudah client dalam mengakses data di web server dengan mengingat domain name saja tanpa perlu mengingat alamat ip dari server tersebut
- DNS Server dan Web Server dapat bekerja lintas platform
- Alur kerja keseluruhan adalah client meminta DNS untuk mengubah url menjadi ip address, dns mengirimkan ip address dari web server, selanjutnya client merequest data menggunakan ip address web server, dan web server merespon permintaan dengan mengirimkan data dalam bentuk html

Week 6

VLAN
