**Cara Agar *Host* Mendapatkan IP *Address***

* *Hard-coded,* secara manual meng *assign* IP *Address* dengan membagi *subnetting.* Digunakan ketika kita ingin meng assign IP *Address* yang permanen.
* DHCP (*Dynamic Host Configuration Protocol*), IP *Address* nya berubah - ubah.

**DHCP**

Mendapatkan secara dinamis IP *Address* dimana dapat di *renew* dan di *reuse.*

* Ketika pertama kali belum memiliki IP *Address,* akan dilakukan *broadcast* oleh host
* Kemudian DHCP server akan membalas *host* dan DHCP akan melakukan *offering* IP *Address*
* *Client* akan memberikan *request*
* DHCP server akan memberikan *acknowledge*

**Informasi yang Diberikan DHCP**

* *Address* dari *first-hop router* untuk *client*
* Nama dan IP *Address* dari DNS server
* *Network Mask*

***Hierarchical Addressing***

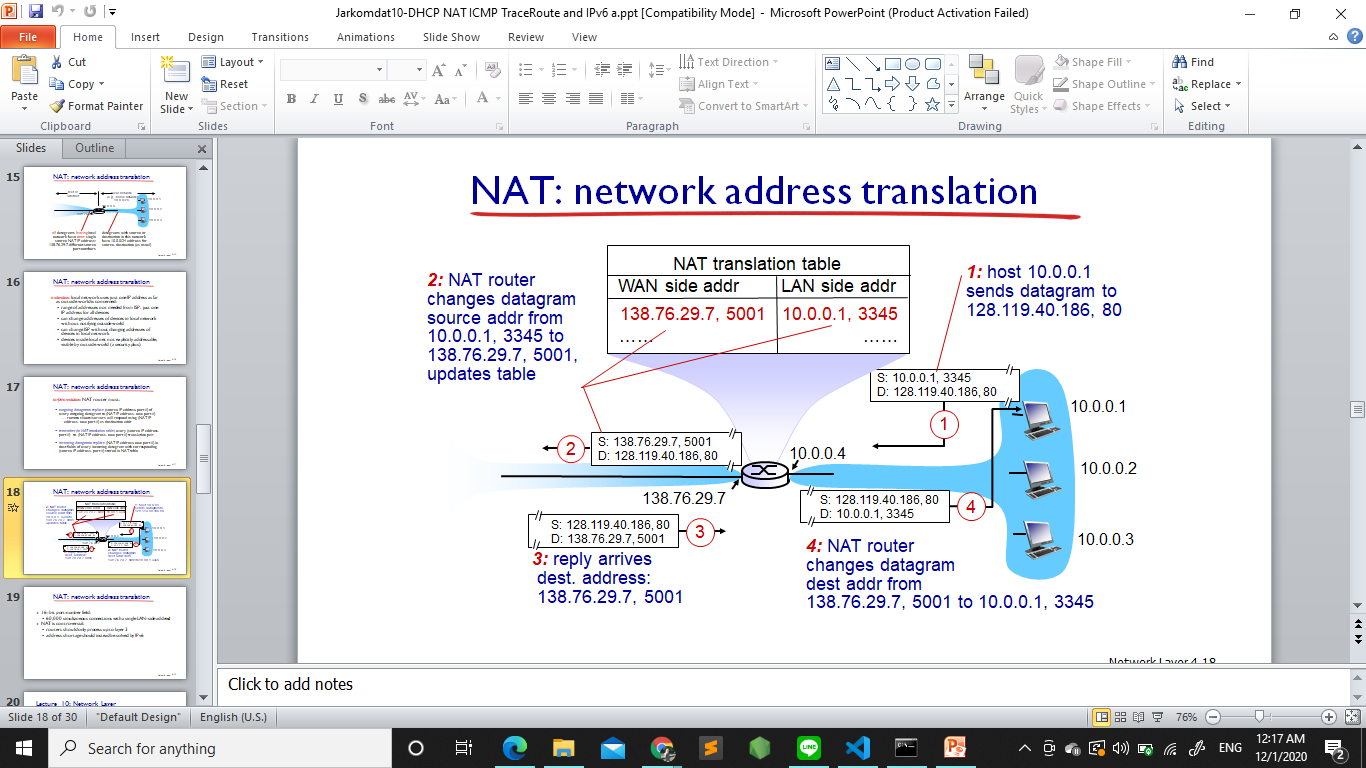
Misalkan kita ingin mengakases scele.cs.ui.ac.id

* Pertama yang kita lakukan adalah melihat dulu domain .id -> .ac -> .ui -> .cs -> scele
* Kita harus melihat dulu domain globalnya

***Network Address Translation***

Ketika kita berselancar di internet, IP *Address* yang digunakan bukan IP *Address* public. IP *Address* tersebut di *translate* sehingga seolah - olah kita menggunakan IP *Address public.*

* NAT ini digunakan karena *range address* sangat jarang.
* Dengan NAT kita dapat dengan mengganti IP *Address* sesuai dengan keinginan
* Kita dapat mengubah ISP tanpa mengubah IP *Address*
* *Device* di dalam *local network* tidak dapat diakses dari luar sehingga perlu di translasi



**Implementasi NAT**

* *Outgoing* datagram akan di *replace*
* Akan mengingat *translation* tabelnya.
* *Incoming* datagram akan di *replace* sesuai dengan tabelnya

**ICMP**

Protokol yang digunakan untuk mengontrol status dari pesan kita. Digunakan oleh *host* dan *router* untuk berkomunikasi satu sama lain.

* ICMP *message* dibawa menggunakan IP datagram
* ICMP terdapat 8 byte IP datagram

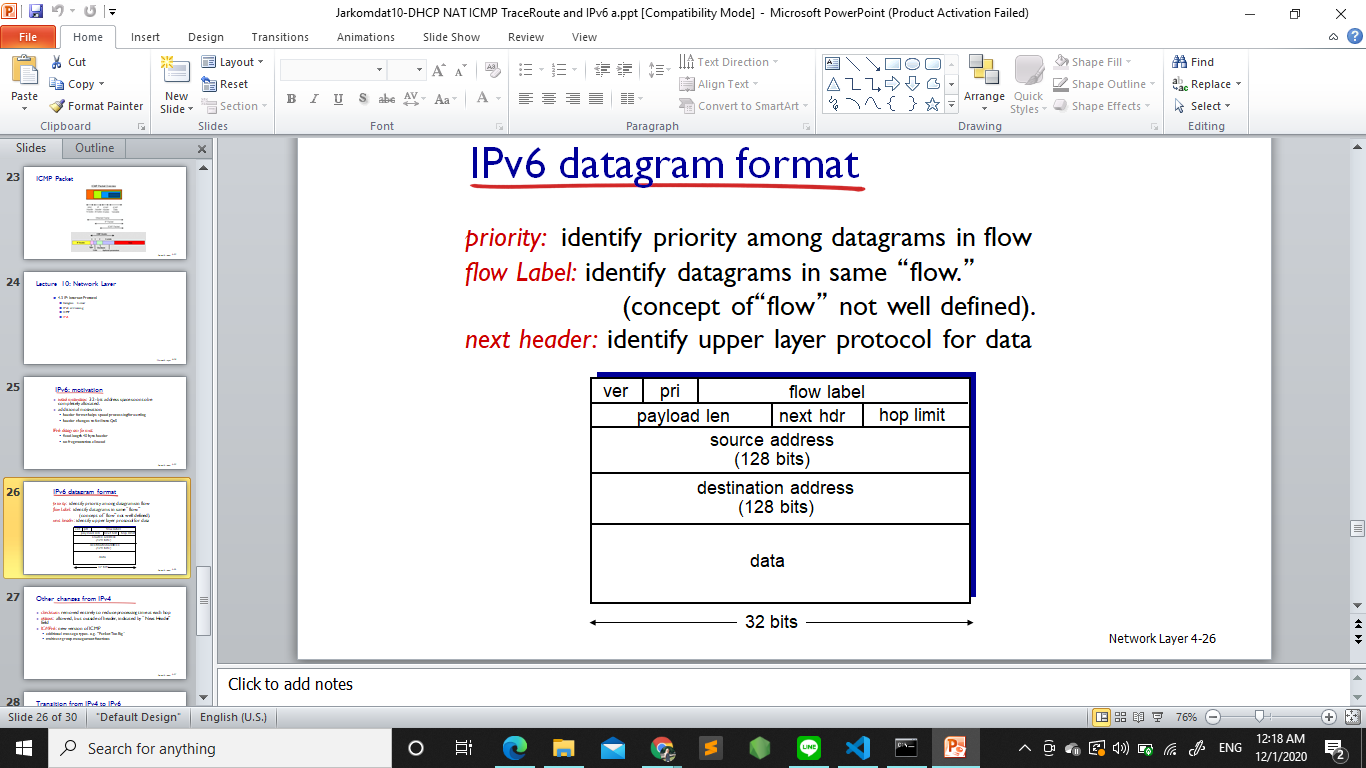
***Traceroute***

Untuk mengetahui status dari tempat kita ke suatu server.

* Time To Live : hoping yang dilakukan dari suatu *router* ke *router* lainnya
* Biasanya akan dilakukan 3 kali
* Kemudian akan di *list*
* Waktu yang dihitung adalah bolak balik
* Akan berhenti ketika sudah sampai destinasi atau bisa kita atur sendiri berapa kali akan dilakukan *hoping*

**IPv6**

Solusi yang dapat digunakan ketika 32-bit IPV4 tidak dapat digunakan.



* Pada IPv6, *checksum* dihapus untuk mengurangi *processing time*
* Karena sudah banyak yang menggunakan IPv4, sehingga perlu dilakukan *tunneling.*

**Transisi dari IPv4 ke IPv6**

* Tidak semua *router* tidak bisa di *upgrade* secara bersamaan.
* Sehingga perlu dilakukan *tunneling,* IPv6 datagram akan dibawa sebagai *payload* di IPv4 datagram

