

Pengenalan Jaringan dengan cisco

By Syahdan Aziz

Pengantar Buku

Mengawali pengantar buku ini, penulis ingin secara khidmat bersyukur kepada Allah SWT atas kemudahan yang diberikan dalam menyusun buku ini. Alhamdulillah. Segala Puji hanya milik allah. Tuhan semesta alam. Selanjutnya penulis ingin bershallowat kepada rasulullah shallallahu ‘ alaihi wa sallam, sosok manusia paling sempurna yang selalu penulis rindukan.

Dengan Judul “ Pengenalan Jaringan Dengan Cisco “ Menyediakan berbagai materi seperti : Jaringan Hari Ini, Dasar Switch dan konfigurasinya, Protokol dengan modelnya, Physical Layer, Bilangan Dalam Sistem, Data Link Layer, Teknologi Ethernet, Network Layer, Address Resolution, Dasar Router Dan Konfigurasinya, IPv4, IPv6, ICMP, Transport Layer, Application Layer, Dasar Keamanan Jaringan, Simulasi Membuat Jaringan

Buku ini adalah sepintas dari ilmu yang saya dapatkan dari beberapa hasil pembelajaran selama 4 tahun, Hasil translate dari Netacad Introduction to network serta penulis menambahkan materi materi lain yang mungkin membantu untuk melengkapi tulisan di buku ini

Tak Lupa, penulis mengucapkan banyak terima kasih kepada semua pihak yang telah memberikan semangat dan motivasi agar buku ini bisa terselesaikan, terutama terhadap Ibunda Nur Azizah dan Ayahanda Syahrudin yang telah mensupport penulis dalam berbagai aspek, dan kisah-kisah inspirasi para ulama terdahulu tentang pentingnya menulis agar ilmu itu tidak hilang

Tentu saja buku ini masih sangat jauh dari kata sempurna. Untuk Itu, Penulis memohon dengan sangat apabila terdapat penjelasan yang kurang jelas atau ingin memberikan masukan atau saran kepada penulis agar nantinya bisa memperbaiki buku ini lebih baik lagi

Pembaca bisa menghubungi penulis lewat kontak berikut :

E-mail : syahdanaziz546@gmail.com

Facebook : <https://www.facebook.com/syahdanazizitns>

Website : <https://intelektualpeople.id>

Jakarta, December 2021



Syahdan Aziz

Contents

| | |
|--|-----|
| ~ Networking Today ~..... | 6 |
| Pengaruh Jaringan dalam kehidupan kita | 8 |
| Komponen Jaringan | 9 |
| Representasi Jaringan dan Topologi | 11 |
| Jenis Jenis Jaringan..... | 13 |
| Membuat Jaringan Handal | 16 |
| Tren Jaringan | 20 |
| Keamanan Jaringan..... | 27 |
| Profesional IT | 30 |
| ~ Basic Switch And End Device Configuration ~..... | 31 |
| Akses Cisco IOS | 33 |
| Navigasi IOS Cisco..... | 40 |
| Struktur Command Line IOS | 44 |
| Konfigurasi Dasar Perangkat | 48 |
| Menyimpan Konfigurasi | 52 |
| Port dan Addresses..... | 56 |
| Konfigurasi IP Address..... | 60 |
| Verifikasi Konektivitas | 64 |
| ~ Protocols dan Model ~..... | 66 |
| THE RULES..... | 68 |
| PROTOCOL | 75 |
| Rangkaian Protocol..... | 79 |
| Organisasi Standard | 86 |
| Model Referensi..... | 90 |
| Enkapsulasi Data..... | 95 |
| Data Access | 99 |
| ~ Physical Layer ~ | 107 |
| Tujuan dibentuknya physical layer | 109 |
| Karakteristik Physical Layer | 112 |
| Kabel Tembaga..... | 119 |
| Kabel UTP | 125 |
| Kabel Fiber Optik | 132 |
| Media Wireless | 139 |
| ~ Sistem Nomor ~..... | 142 |
| Sistem bilangan binary..... | 144 |

| | |
|--|-----|
| Sistem bilangan hexadesimal | 150 |
| Tujuan Data link layer | 155 |
| Topologies | 160 |
| Frame Data Link | 173 |
| ~ Ethernet Switching ~ | 179 |
| Ethernet Frame..... | 181 |
| Ethernet MAC Address..... | 186 |
| MAC Address Table | 194 |
| Metode Switch Speed dan Forwarding | 199 |
| ~ Network Layer ~..... | 205 |
| Paket IPv4..... | 214 |
| Paket IPv6..... | 216 |
| Cara Kerja Router | 221 |
| ~ Address Resolution ~..... | 231 |
| MAC dan IP..... | 233 |
| Cara Kerja ARP | 236 |
| IPv6 Neighbor Discovery..... | 243 |
| ~ Basic Router Configuration ~ | 245 |
| Mengkonfigurasi Settingan awal router | 247 |
| Mengkonfigurasi Interface | 250 |
| Mengkonfigurasi Default Gateway | 254 |
| ~ IPv4 Addressing ~ | 257 |
| Struktur Alamat IPv4 | 259 |
| IPv4 Unicast, Broadcast, Dan Multicast | 267 |
| Tipe Tipe IPv4 Address | 271 |
| Segmentasi Jaringan | 278 |
| Subnetwork IPv4..... | 283 |
| Subnet Prefix /16 dan /8..... | 287 |
| Persyaratan untuk memenuhi subnet..... | 294 |
| Variable Length Subnet Mask..... | 301 |
| Desain Terstruktur | 308 |
| ~ IPv6 Addressing ~ | 311 |
| Masalah IPv4 | 313 |
| Representasi Alamat IPv6..... | 316 |
| Jenis Jenis Alamat IPv6 | 322 |
| Konfigurasi Static GUA dan LLA | 329 |
| Alamat Dinamis untuk IPv6 GUAs..... | 333 |
| Alamat Dinamis untuk IPv6 LLAs..... | 341 |
| IPv6 Multicast Address | 345 |

| | |
|---|-----|
| Subnet pada IPv6 | 347 |
| ~ ICMP ~..... | 351 |
| Pesan ICMP | 353 |
| Test Ping dan Traceroute | 358 |
| ~ Transport Layer ~ | 363 |
| Transportasi Data | 365 |
| Gambaran Umum TCP | 374 |
| Gambaran Umum UDP | 378 |
| Nomor Port | 381 |
| Proses Komunikasi TCP | 386 |
| Reliability dan Flow Control..... | 395 |
| Komunikasi UDP..... | 403 |
| ~ Application Layer ~..... | 408 |
| Session, Presentation, Application Layer..... | 410 |
| Peer To Peer..... | 415 |
| Email Dan Web Protocol | 419 |
| Layanan IP Address | 427 |
| File Transfer Protocol | 436 |
| ~ Network Security Fundamentals ~..... | 439 |
| Ancaman Dan Kerentanan Keamanan | 441 |
| Serangan Jaringan | 446 |
| Mitigasi Serangan Jaringan..... | 453 |
| Keamanan Jaringan..... | 460 |
| ~ Build A Small Network ~..... | 466 |
| Perangkat Dalam Jaringan Kecil | 468 |
| Aplikasi dan Protokol jaringan kecil | 475 |
| Skala ke jaringan yang lebih besar | 481 |
| Verifikasi Konektivitas | 484 |
| Perintah Host dan IOS | 493 |
| Metodologi Troubleshooting..... | 508 |
| Skenario Troubleshooting | 513 |

BAB 1

~ *Networking Today* ~

Judul Bab : Networking Today

Tujuan Bab : Menjelaskan kemajuan teknologi jaringan modern

Link Test Pemahaman : <https://s.id/QwLC>

| Judul Materi | Tujuan Materi |
|--|---|
| Pengaruh Jaringan dalam kehidupan kita | Menjelaskan bagaimana jaringan mempengaruhi kehidupan kita |
| Komponen Jaringan | Menjelaskan bagaimana host dan perangkat jaringan digunakan |
| Representasi Jaringan dan Topologi | Menjelaskan Representasi jaringan dan bagaimana mereka menggunakan topology jaringan |
| Jenis-jenis umum pada jaringan | Membandingkan karakteristik dari jenis jaringan |
| Koneksi Jaringan | Menjelaskan Bagaimana LANs dan WANs tersambung ke internet |
| Keandalan/ <i>Reliable</i> Jaringan | Menjabarkan 4 persyaratan dasar jaringan yang handal/ <i>reliable</i> |
| Trend Jaringan | Menjelaskan bagaimana trend seperti <i>BYOD</i> , <i>Video collaboration</i> , <i>Video</i> , dan <i>Cloud computing</i> mengubah cara kita berkomunikasi |
| Keamanan Jaringan | Mengidentifikasi Ancaman/ <i>Threats</i> keamanan dasar dan solusi untuk semua jaringan |
| Professional IT | Menjelaskan peluang kerja di bidang jaringan |

Pengaruh Jaringan dalam kehidupan kita

Zaman semakin maju dan kedepan, teknologi tidak akan diam apalagi mundur. Internet adalah suatu hal yang sudah umum dipakai di zaman ini dari kota sampai ke desa. Namun ada beberapa hal yang harus kita pahami bahwa internet dapat membuat dampak terhadap kita.

1. Menghubungkan kita semua

Kita tau bahwa manusia itu mempunyai kebutuhan berinteraksi dengan orang lain. Komunikasi layaknya udara, air, makanan dan tempat tinggal. Di zaman ini, penggunaan jaringan sudah tidak asing lagi, kita terhubung melalui jaringan internet tidak seperti sebelumnya.

Orang yang memiliki kepentingan personal atau kelompok dapat berkomunikasi dengan instan menggunakan jaringan internet. Peristiwa atau berita pun dapat tersebar dengan sangat cepat. Bahkan kita dapat terhubung dalam game yang berbeda benua.

2. Tidak ada batasan

Perkembangan dalam teknologi jaringan adalah suatu perubahan paling signifikan di dunia saat ini. Mereka dapat menciptakan dunia tanpa batas jarak dan waktu, hambatan hambatan dalam berkomunikasi semakin berkurang dan semakin bebasnya berpendapat.

3. Merubah perilaku

Jika kita melihat zaman sekarang dan dulu sangat beda. Internet telah merubah cara interaksi sosial, komersial, politik dan pribadi kita sendiri. Dalam dunia pendidikan internet sangat merubah perilaku pelajar. Pelajar lebih harus aktif dalam mencari sumber informasi pada internet sehingga membangun pemikiran analisis.

Dalam dunia ekonomi juga berubah. Sekarang kita belanja bisa lewat hp saja, uang juga sudah disimpan dalam bentuk digital sehingga itu sangat memudahkan manusia hingga akhirnya merubah perilaku jual dan beli di masyarakat.

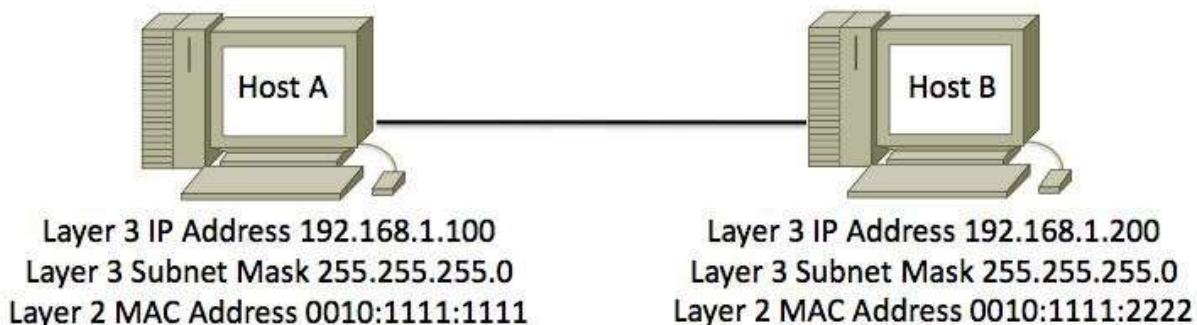
Komponen Jaringan

Ketika kita ingin memahami sebuah sesuatu baiknya kita paham sebuah komponen dari hal tersebut. Sehingga ketika mengalami permasalahan akan cepat ditangani. Kita hanya perlu fokus terhadap komponen apa yang bermasalah

Berikut beberapa komponen yang dijelaskan pada materi ini:

1. Host

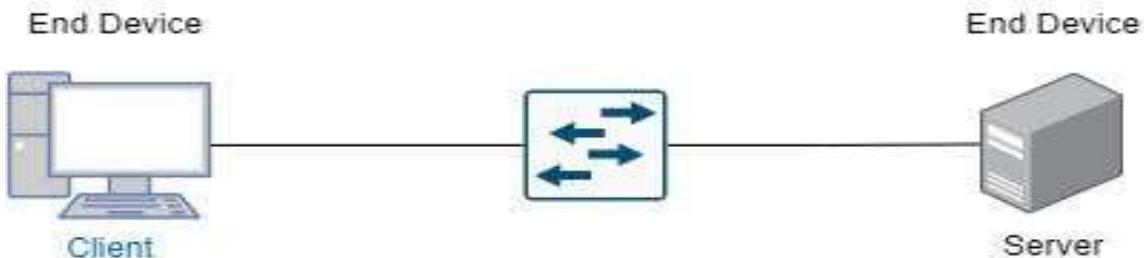
Semua komputer yang terhubung ke jaringan dan berpartisipasi langsung dalam komunikasi jaringan disebut sebagai host. Host secara khusus mengacu pada perangkat di jaringan yang diberi nomor untuk tujuan komunikasi. Nomor ini mengidentifikasi host dalam jaringan tertentu. Nomor ini disebut Internet Protocol (IP)



2. End Devices

Perangkat jaringan yang paling dikenal orang adalah end devices, yaitu handphone, tablet, komputer. Untuk membedakan satu end device dari yang lain, setiap end device jaringan memiliki alamat jaringan. End device adalah perangkat *source* atau *destination* yang dikirimkan melalui jaringan

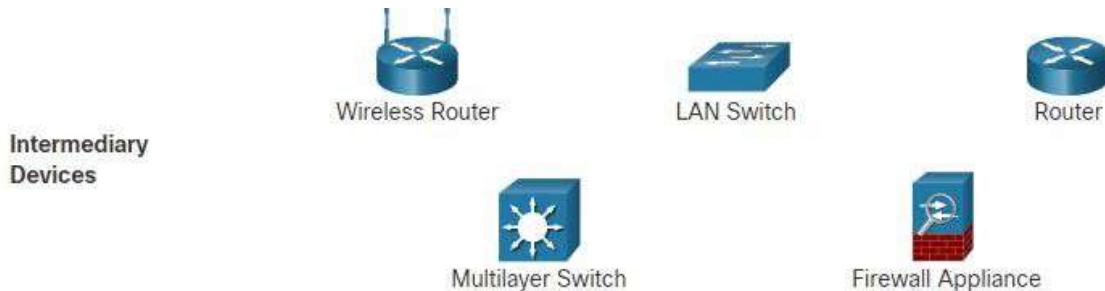
Server adalah komputer dengan perangkat lunak yang memberikan informasi, seperti email atau halaman web, ke perangkat lain dalam jaringan sedangkan **Client** adalah komputer dengan perangkat lunak yang meminta dan menampilkan informasi yang diperoleh dari server



3. Intermediary Device

Perangkat yang menghubungkan beberapa end device ke jaringan untuk membuat sebuah jaringan berskala disebut intermediary devices. Intermediary menyediakan koneksi dan memastikan data berjalan di seluruh jaringan

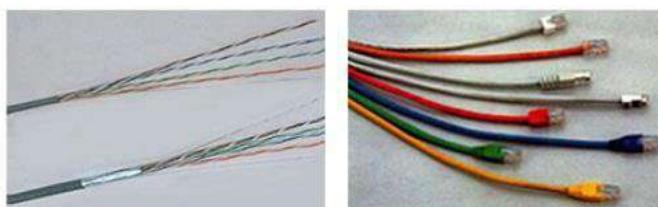
Intermediary devices menggunakan alamat perangkat end devices, bersama dengan informasi tentang *interkoneksi* jaringan, untuk menentukan jalur yang harus diambil melalui jaringan



4. Network Media

Perangkat tidak akan bisa berkomunikasi melalui jaringan tanpa adanya media. Media menyediakan saluran dimana pesan berjalan dari *source* ke *destination*. Jaringan modern menggunakan 3 jenis media untuk menghubungkan perangkat

- **Kabel Logam** - Data diubah menjadi impuls listrik



- **Kabel Kaca atau serat plastik** – Data diubah menjadi pulsa cahaya



- **Non kabel/ Nirkabel** – Data diubah melalui modulasi frekuensi tertentu dari gelombang elektromagnetik

Representasi Jaringan dan Topologi

Arsitek dan administrator jaringan harus dapat menunjukkan seperti apa jaringan mereka nantinya, Mereka harus dapat dengan mudah melihat komponen mana yang terhubung ke komponen lain, di mana lokasinya, dan bagaimana mereka akan dihubungkan.

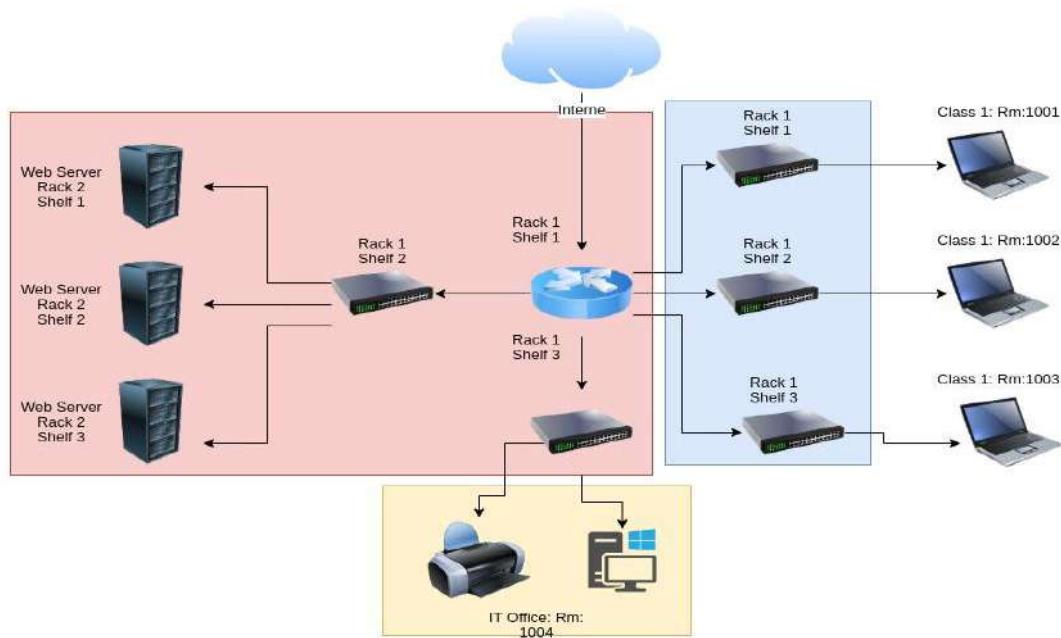
Diagram jaringan sering kali menggunakan simbol, untuk mewakili berbagai perangkat dan koneksi yang membentuk jaringan. Diagram memberikan cara mudah untuk memahami bagaimana perangkat terhubung dalam jaringan besar. Jenis “gambar” jaringan ini dikenal sebagai **diagram topologi**.

Topologi Jaringan

Diagram topologi adalah dokumentasi wajib bagi siapa pun yang bekerja dengan jaringan. Mereka menyediakan peta visual tentang bagaimana jaringan terhubung. Ada dua jenis diagram topologi: **Physical** dan **Logical**.

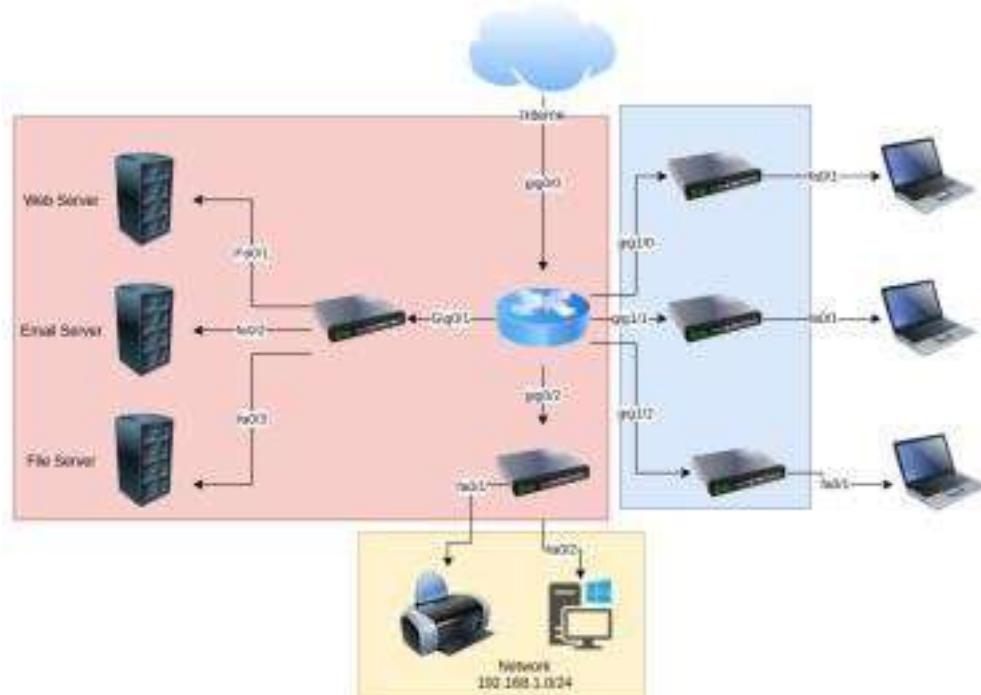
1. Topologi Physical

Diagram topologi physical menggambarkan lokasi fisik antara host dan pemasangan kabel, seperti yang ditunjukkan pada gambar. Anda dapat melihat bahwa ruangan tempat perangkat ini berada diberi label dalam topologi fisik ini.



2. Topology Logical

Diagram topology logical menggambarkan perangkat, port, dan skema pengalamanan jaringan , seperti yang ditunjukan pada gambar. Anda dapat melihat end device mana yang terhubung ke intermediary device dan media apa yang digunakan



Jenis Jenis Jaringan

Setelah sudah membaca materi komponen, representasi, dan topology jaringan maka anda siap untuk melanjutkan materi tentang berbagai jenis jaringan

Jaringan dari berbagai ukuran

Jaringan tersedia dalam berbagai ukuran. Mulai dari jaringan sederhana yang terdiri dari dua komputer, hingga jaringan yang menghubungkan jutaan perangkat. Jaringan rumah sederhana memungkinkan Anda berbagi *resource*, seperti dokumen, gambar, dan musik, diantara beberapa end devices.

Jaringan kantor kecil dan kantor rumah (SOHO) memungkinkan orang untuk bekerja dari rumah, atau kantor jarak jauh. Banyak pekerja mandiri menggunakan jenis jaringan ini untuk mengiklankan dan menjual produk, memesan persediaan, dan berkomunikasi dengan pelanggan.

Bisnis dan organisasi besar menggunakan jaringan untuk menyediakan *konsolidasi*, penyimpanan, dan akses ke informasi di server jaringan. Jaringan menyediakan email, pesan instan, dan kolaborasi di antara karyawan. Banyak organisasi menggunakan koneksi jaringan mereka ke internet untuk menyediakan produk dan layanan kepada pelanggan.

Internet adalah jaringan terbesar yang pernah ada. Faktanya, istilah internet berarti “Jaringan dari jaringan”. Ini adalah kumpulan jaringan pribadi dan publik yang saling berhubungan.

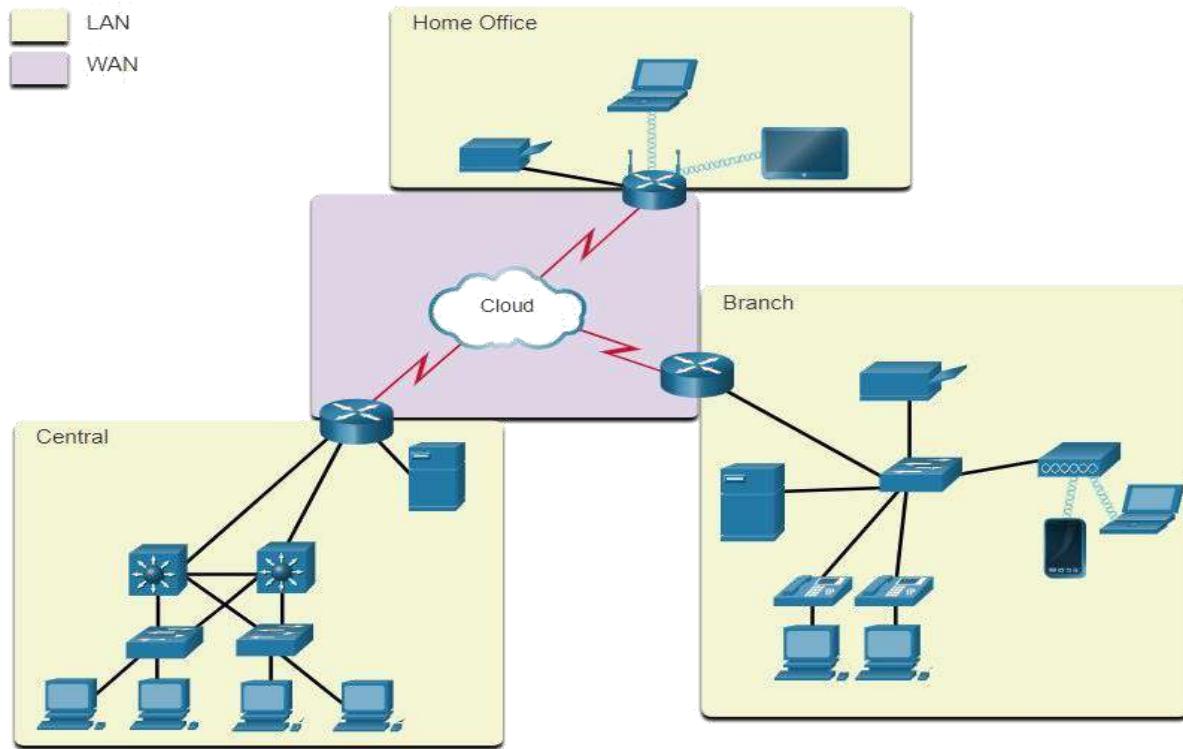
Dalam bisnis kecil dan rumah, banyak komputer berfungsi sebagai server dan klien di jaringan. Jenis jaringan ini disebut jaringan peer-to-peer

LANs dan WANs

Untuk mengetahui jenis jenis jaringan maka kita harus tau infrastruktur jaringan pada saat ini bervariasi dalam hal :

- Ukuran area yang dicakup
- Jumlah pengguna yang terhubung
- Jumlah dan jenis layanan yang tersedia

Dua jenis infrastruktur jaringan yang paling umum adalah Local Area Networks (LAN), dan Wide Area Networks (WAN). LAN adalah infrastruktur jaringan yang menyediakan akses ke pengguna dan End Device di wilayah geografis kecil. LAN biasanya digunakan di departemen dalam bisnis, rumah, atau jaringan bisnis kecil. WAN adalah infrastruktur jaringan yang menyediakan akses ke jaringan lain melalui wilayah geografis yang luas, yang biasanya dimiliki dan dikelola oleh perusahaan yang lebih besar atau penyedia layanan telekomunikasi. Angka tersebut menunjukkan LAN terhubung ke WAN.



LAN

- LAN menghubungkan end device di area terbatas seperti rumah, sekolah, gedung kantor, atau kampus
- LAN biasanya dikelola oleh satu organisasi atau individu
- LAN biasanya menyediakan bandwidth berkecepatan tinggi ke end device internal dan intermediary device

WAN

- WAN adalah interkoneksi LAN melalui wilayah geografis yang luas seperti antara kota, negara bagian, provinsi, negara, atau benua
- WAN biasanya dikelola oleh beberapa penyedia layanan
- WAN biasanya menyediakan sambungan lebih lambat antara LAN

The Internet

Internet adalah kumpulan jaringan yang saling berhubungan di seluruh dunia (internetworks, atau disingkat internet). Internet tidak dimiliki oleh individu atau kelompok mana pun). Memastikan komunikasi yang efektif di seluruh infrastruktur yang beragam ini membutuhkan penerapan teknologi dan standar yang konsisten dan diakui secara umum serta kerja sama dari banyak lembaga administrasi jaringan.

Ada organisasi yang dikembangkan untuk menjaga struktur dan standarisasi *protokol* dan proses internet. Organisasi-organisasi ini termasuk Internet Engineering Task Force (IETF), Internet Corporation for Assigned Names and Numbers (ICANN), dan Internet Architecture Board (IAB), ditambah banyak lagi lainnya.

Intranets and Extranets

Ada dua istilah lain yang mirip dengan istilah internet: Intranet dan extranet

Intranet adalah istilah yang sering digunakan untuk merujuk pada koneksi pribadi LAN dan WAN milik suatu organisasi. Intranet dirancang agar hanya dapat diakses oleh anggota organisasi, karyawan, atau orang lain dengan otorisasi. Berikut contohnya:

1. Kantor A terletak di jakarta dan bandung lalu kantor A mempunyai sebuah aplikasi untuk operasional kantor, karyawan yang di jakarta maupun di bandung tetap bisa mengakses aplikasi tersebut dengan koneksi intranet

Extranet adalah koneksi pribadi namun memberikan akses yang aman kepada organisasi yang berbeda yang memerlukan akses. Berikut contohnya :

1. Perusahaan yang menyediakan akses ke pemasok dan kontraktor luar
2. Rumah sakit yang menyediakan sistem pemesanan untuk dokter sehingga mereka dapat membuat janji untuk pasiennya
3. Kantor pendidikan lokal yang memberikan informasi anggaran dan personil ke sekolah-sekolah di distriknya

Membuat Jaringan Handal

Pernahkah Anda ketika kerja tiba tiba internet terputus ? Seperti yang Anda ketahui sekarang, internet tidak mati, Anda hanya kehilangan koneksi ke arah sana. Ini sangat membuat frustasi. Dengan begitu banyak orang didunia yang mengandalkan akses jaringan untuk bekerja dan belajar,Maka dari itu kita akan membuat jaringan yang dapat diandalkan atau disebut **Reliable Network**.

Arsitektur Jaringan

Peran jaringan telah berubah dari jaringan data-only (mengirim dan menyimpan data saja) menjadi sistem yang memungkinkan koneksi orang, perangkat, dan informasi dalam lingkungan jaringan yang terkumpul dan kaya media (youtube,facebook,google), Agar jaringan berfungsi secara efisien dan tumbuh dalam jenis lingkungan ini, Jaringan harus dibangun di atas arsitektur jaringan standar.

Jaringan juga mendukung berbagai aplikasi dan layanan. Mereka harus beroperasi pada berbagai jenis kabel (fiber optik, utp, dll) dan perangkat (router,komputer,handphone), yang membentuk infrastruktur fisik. Tapi dalam konteks ini kita tidak membicarakan infrastruktur fisik tetapi mengacu pada teknologi yang mendukung infrastruktur dan layanan dan aturan terprogram, atau *protokol*, yang memindahkan data dan melintasi jaringan

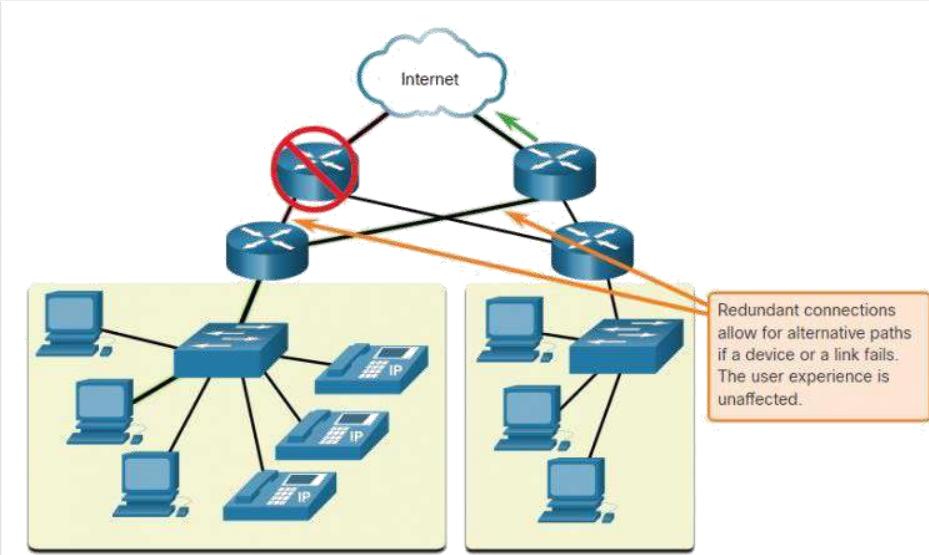
Seiring perkembangan jaringan, kami telah mempelajari bahwa ada empat karakteristik dasar yang harus ditangani oleh arsitek jaringan untuk memenuhi harapan pengguna (**Fault tolerance, Scalability, Quality of Service, Network Security**)

1. Fault Tolerance

Fault tolerance (Jaringan toleransi kesalahan) adalah salah satu arsitektur yang membuat sebuah *backupan link* ketika terjadi kegagalan. Fault tolerance dibangun untuk memungkinkan pemulihan cepat ketika kegagalan atau koneksi terputus terjadi.

Jaringan bergantung pada beberapa jalur antara sumber dan tujuan. Jika satu jalur gagal, Pesan langsung dikirim melalui jalur yang berbeda. Memiliki beberapa jalur ke suatu tujuan dikenal sebagai **redundansi**.

Contoh : Sebuah perusahaan menggunakan 2 ISP, ISP A dan ISP B ketika terjadi *downtime* pada ISP A maka ISP B akan menggantikan jalur internet pada perusahaan, sebaliknya jika ISP B mengalami *downtime* maka ISP A akan menggantikan jalur internet pada perusahaan

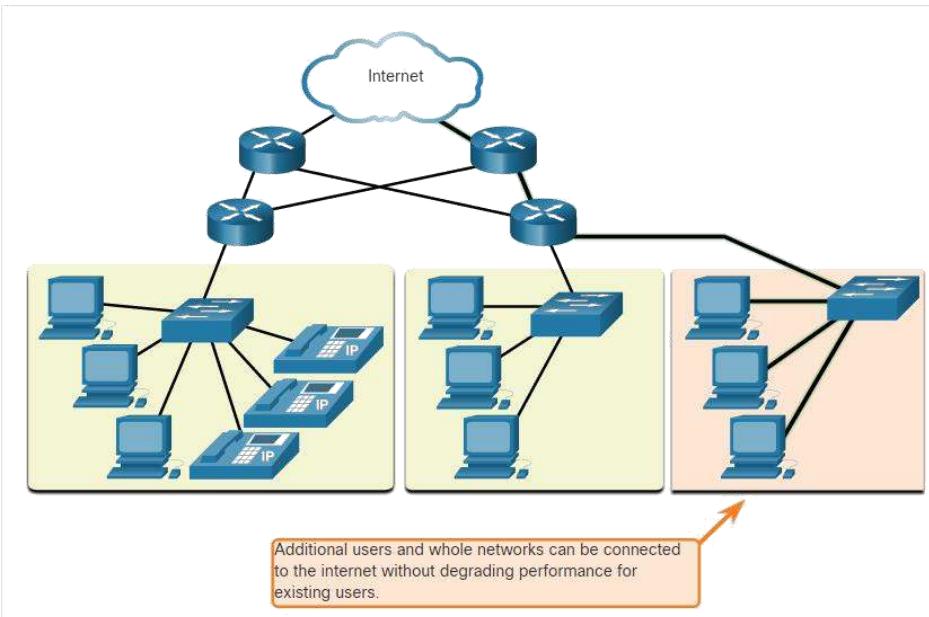


2. Scalability

Jaringan yang dapat diskalakan (diperbesar maupun diperkecil) dapat mengikuti perkembangannya pengguna dan aplikasi baru. Ini dilakukan tanpa menurunkan kinerja layanan yang sedang diakses oleh pengguna yang ada.

Jaringan dapat diskalakan karena perencanaannya mengikuti standar dan protocol yang diterima. ini kemungkinan vendor perangkat lunak dan perangkat keras fokus pada peningkatan produk dan layanan tanpa harus merancang perangkat dan aturan baru untuk beroperasi dalam jaringan.

Contoh : Sebuah perusahaan memiliki perangkat end device sebanyak 6, lalu kita akan membelikannya sebuah switch yang memiliki port 8 atau bahkan lebih, ini sangat penting karena jika suatu saat end device bertambah maka tidak perlu membeli ulang switch dan mengkonfigurasi ulang switch, tinggal memanfaatkan port yang kosong saja



3. Quality Of Services (QoS)

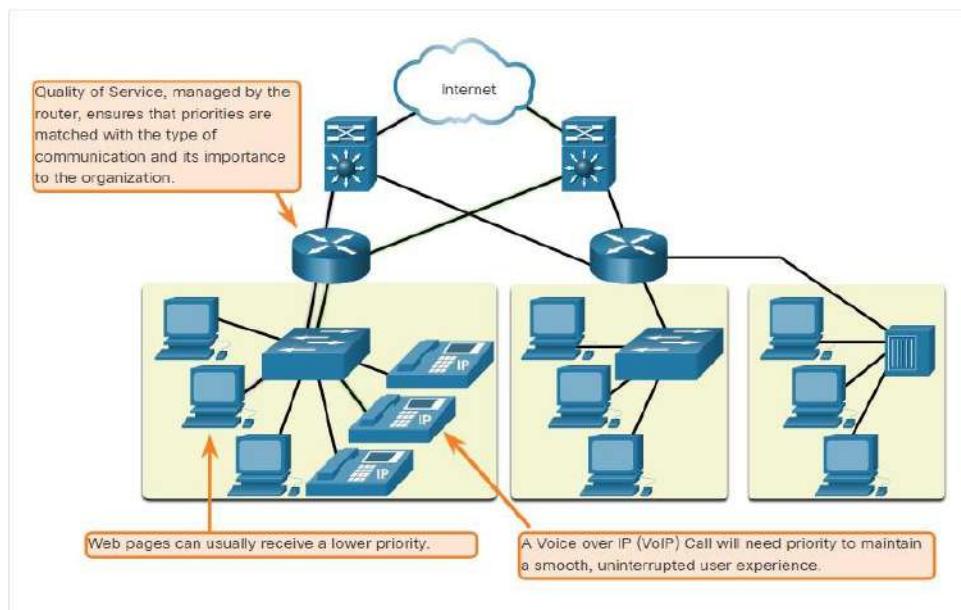
Kebutuhan jaringan yang semakin meningkat saat ini. Aplikasi baru yang tersedia bagi pengguna melalui jaringan, seperti transmisi suara (telepon) dan video langsung (zoom,gmeet,jitsi), menciptakan harapan yang lebih tinggi untuk kualitas layanan yang diberikan

Pernahkah anda mencoba menonton video dengan jeda dan jeda yang konstan (Buffering) ? Itu terjadi karena data,suara,dan konten video terus menyatu ke jaringan yang sama, QoS menjadi mekanisme utama untuk mengelola kemacetan dan memastikan pengiriman konten yang handal ke semua pengguna. Kemacetan terjadi ketika permintaan bandwidth melebihi jumlah yang tersedia.

Ketika komunikasi simultan (bersamaan) dilakukan di seluruh jaringan, permintaan bandwidth jaringan dapat melebihi ketersediaannya, sehingga menyebabkan kemacetan jaringan. Ketika volume traffic lebih besar dari yang dapat diangkut melalui jaringan, perangkat akan menyimpan paket di memori hingga sumber daya tersedia untuk mengirimkannya.

Ketika satu pengguna meminta halaman web, dan pengguna lainnya sedang melakukan panggilan telepon. Dengan kebijakan QoS, router dapat mengatur aliran data dan traffic suara, memprioritaskan komunikasi suara jika jaringan mengalami kemacetan.

Contoh : Ketika kita melakukan video teleconference (Zoom, Gmeet, Jitsi) lalu ada seorang yang sedang mendownload sebuah file yang ukurannya sangat besar itu akan mengganggu video teleconference kita, maka dari itu QoS akan membatasi bandwidth setiap user agar tidak terjadi kemacetan traffic atau mengganggu aktifitas user la



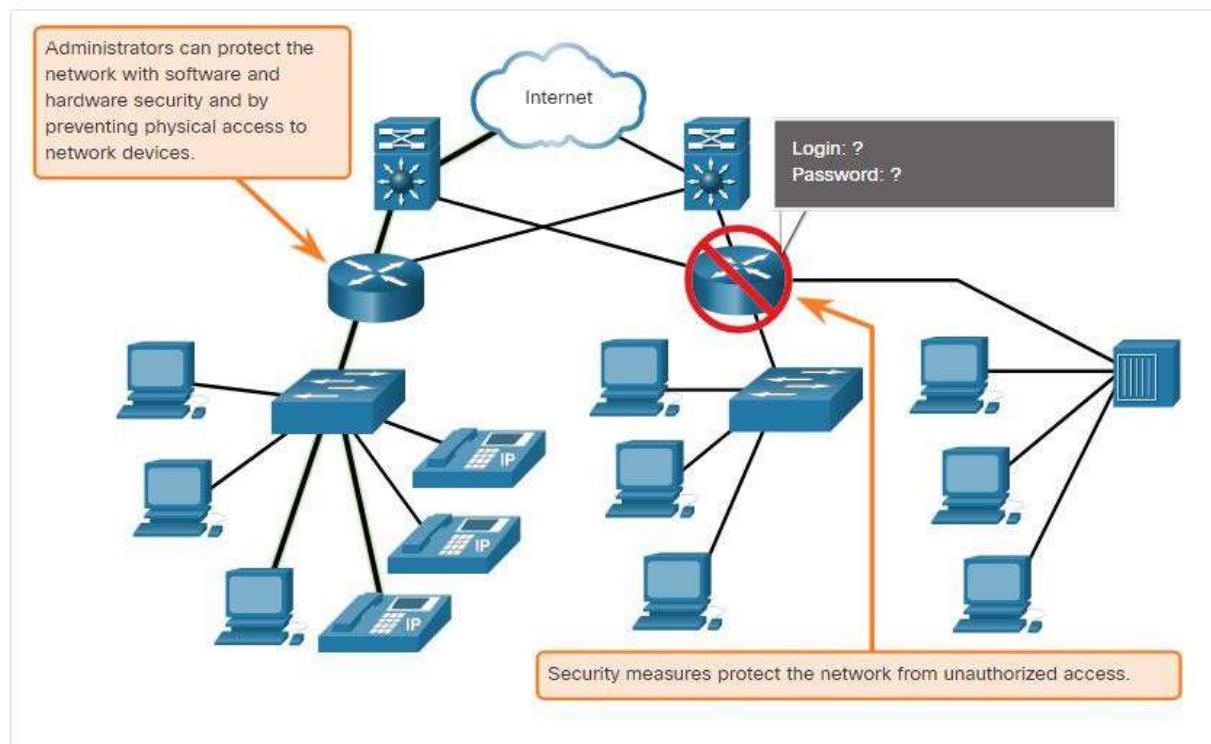
4. Network Security

Infrastruktur jaringan, layanan, dan data yang terdapat pada perangkat yang terhubung ke jaringan adalah aset pribadi dan bisnis yang penting. Administrator jaringan harus menangani dua jenis masalah keamanan jaringan: Keamanan infrastruktur jaringan dan keamanan informasi

Mengamankan Infrastruktur jaringan mencakup pengamanan perangkat secara fisik yang menyediakan konektivitas jaringan dan mencegah akses tidak sah ke perangkat lunak manajemen yang ada di dalamnya

Administrator jaringan juga harus melindungi informasi yang terdapat di dalam paket yang dikirim melalui jaringan, dan informasi yang disimpan pada perangkat yang terhubung ke jaringan. Untuk mencapai tujuan keamanan jaringan, ada tiga persyaratan utama

- **Confidentiality/Kerahasiaan** data berarti bahwa hanya penerima yang dituju dan berwenang yang dapat mengakses dan membaca data.
- **Integrity/Integritas** data menjamin pengguna bahwa informasi belum diubah dalam transmisi, dari asal ke tujuan
- **Availability/Ketersediaan** data menjamin pengguna akses yang tepat waktu dan dapat diandalkan ke layanan data untuk pengguna yang berwenang.



Tren Jaringan

Anda tahu banyak tentang jaringan yang sekarang, terbuat dari apa, bagaimana menghubungkan kita, dan apa yang diperlukan agar tetap dapat diandalkan. Tetapi jaringan, seperti yang lainnya, terus berubah. Ada beberapa tren dalam jaringan yang harus Anda ketahui.

Tren Baru

Saat teknologi baru dan perangkat End Device masuk ke pasar, bisnis dan konsumen harus terus menyesuaikan diri dengan lingkungan yang selalu berubah ini. Ada beberapa tren jaringan yang mempengaruhi organisasi dan konsumen:

1. Bring Your Own Device (BYOD)

Konsep perangkat apa pun yang bisa dibawa, untuk konten apa pun yang bisa diakses, dengan cara apa pun koneksinya adalah tren global utama yang memerlukan perubahan signifikan pada cara kami menggunakan perangkat dan menghubungkannya dengan aman ke jaringan. Ini disebut Bring Your Own Device (BYOD).

BYOD memungkinkan kalian memiliki kebebasan untuk menggunakan alat pribadi untuk mengakses informasi dan berkomunikasi di seluruh jaringan bisnis atau kampus. Dengan pertumbuhan perangkat konsumen, dan penurunan biaya, karyawan dan siswa mungkin memiliki perangkat komputasi dan jaringan yang canggih untuk penggunaan pribadi. Ini termasuk laptop, notebook, tablet, ponsel pintar, dan e-reader. Ini dapat dibeli oleh perusahaan atau sekolah, dibeli oleh individu, atau keduanya.



2. Kolaborasi Online

Seorang ingin terhubung ke jaringan, tidak hanya untuk akses ke aplikasi data, tetapi juga untuk berkolaborasi satu sama lain. Kolaborasi didefinisikan sebagai “tindakan bekerja dengan orang lain atau orang lain dalam proyek bersama.” Alat kolaborasi, seperti Zoom Meeting, yang ditunjukkan pada gambar, memberi karyawan, siswa, guru, pelanggan, dan mitra cara untuk langsung terhubung, berinteraksi, dan mencapai tujuan mereka.



Kolaborasi adalah prioritas kritis dan strategis yang digunakan organisasi untuk tetap kompetitif. Kolaborasi juga menjadi prioritas dalam pendidikan. Siswa perlu berkolaborasi untuk membantu satu sama lain dalam pembelajaran, untuk mengembangkan keterampilan tim yang digunakan di dunia kerja, dan untuk bekerja sama dalam proyek berbasis tim.

3. Cloud Computing

Cloud Computing adalah salah satu cara perangkat mengakses dan menyimpan data. Cloud Computing memungkinkan kita untuk menyimpan file pribadi, bahkan mencadangkan seluruh drive di server melalui internet. Aplikasi seperti pengolah kata (Google Docs) dan pengeditan foto (Canva) dapat diakses menggunakan cloud.

Untuk bisnis, Cloud Computing memperluas kemampuan IT tanpa memerlukan investasi dalam infrastruktur baru, melatih personel baru, atau melisensikan perangkat lunak baru. Layanan ini tersedia sesuai permintaan dan dikirimkan secara ekonomis ke perangkat apa pun di mana pun di dunia tanpa mengorbankan keamanan atau fungsi.

Cloud Computing dimungkinkan karena pusat data. Pusat data adalah fasilitas yang digunakan untuk menampung sistem komputer dan komponen. Sebuah pusat data dapat menempati satu ruangan di sebuah gedung, satu lantai atau lebih, atau seluruh bangunan berukuran gudang. Pusat data biasanya sangat mahal untuk dibangun dan dipelihara. Untuk alasan ini, hanya organisasi besar yang menggunakan pusat data yang dibangun secara pribadi untuk menampung data mereka dan memberikan layanan kepada pengguna. Organisasi yang lebih kecil yang tidak mampu mengelola pusat data pribadinya sendiri dapat mengurangi biaya kepemilikan secara keseluruhan dengan menyewa server dan layanan penyimpanan dari organisasi pusat data yang lebih besar di cloud

Untuk **Security**, **Reliability**, dan **Fault Tolerance**, penyedia cloud seringkali menyimpan data di pusat data terdistribusi. Alih-alih menyimpan semua data seseorang atau organisasi di satu pusat data, itu disimpan di beberapa pusat data di lokasi berbeda.

Contoh : Google tidak menyimpan data hanya di amerika saja namun google juga menyimpan data di beberapa negara juga

| Jenis Cloud | Deskripsi |
|----------------------|--|
| Public Cloud | Aplikasi dan layanan berbasis cloud yang ditawarkan di Cloud Public tersedia untuk umum. Layanan mungkin gratis atau ditawarkan dengan model bayar-per-penggunaan, seperti membayar penyimpanan online. Cloud publik menggunakan internet untuk menyediakan layanan. (Google Drive) |
| Private Cloud | Aplikasi dan layanan berbasis cloud yang ditawarkan di cloud pribadi ditujukan untuk organisasi atau entitas tertentu, seperti pemerintah. Private Cloud dapat disiapkan menggunakan jaringan pribadi organisasi, meskipun biaya pembuatan dan pemeliharaannya bisa mahal. Private Cloud juga dapat dikelola oleh organisasi luar dengan keamanan akses yang ketat. (IDcloudhost, Jagoanhosting) |

| | |
|------------------------|---|
| Hybrid Cloud | Hybrid Cloud terdiri dari dua atau lebih awan (contoh: sebagian pribadi, sebagian publik), di mana setiap bagian tetap menjadi objek yang berbeda, tetapi keduanya terhubung menggunakan arsitektur tunggal. Individu di cloud hybrid akan dapat memiliki derajat akses ke berbagai layanan berdasarkan hak akses pengguna. (AWS, GCP, Alibabacloud) |
| Community Cloud | Community Cloud dibuat untuk penggunaan eksklusif oleh entitas atau organisasi tertentu. Perbedaan antara Public cloud dan Community cloud adalah kebutuhan fungsional yang telah disesuaikan untuk komunitas. Misalnya, organisasi perawatan kesehatan harus tetap mematuhi kebijakan dan undang-undang (misalnya, HIPAA) yang memerlukan otentikasi dan kerahasiaan khusus. Community Cloud digunakan oleh banyak organisasi yang memiliki kebutuhan dan perhatian serupa. Community Cloud mirip dengan lingkungan Cloud Computing , tetapi dengan tingkat keamanan, privasi, dan bahkan kepatuhan peraturan Public Cloud yang ditetapkan. (AWS Student) |

Tren Teknologi di Rumah

Tren jaringan tidak hanya mempengaruhi cara kita berkomunikasi di tempat kerja dan di sekolah, tetapi juga mengubah banyak aspek di rumah. Tren rumah terbaru termasuk ‘teknologi smarthome’.

1. Smarthome

Teknologi smart home terintegrasi ke dalam peralatan sehari-hari, yang kemudian dapat dihubungkan dengan perangkat lain untuk membuat peralatan lebih ‘pintar’ atau otomatis. Misalnya, Anda bisa menyiapkan makanan dan memasukkannya ke dalam oven untuk dimasak sebelum meninggalkan rumah pada hari itu. Anda memprogram oven pintar Anda untuk makanan yang ingin Anda masak. Ini juga akan dihubungkan dengan ‘kalender acara’ Anda sehingga dapat menentukan jam berapa Anda harus tersedia untuk makan dan menyesuaikan waktu mulai dan lama memasak yang sesuai. Ia bahkan dapat menyesuaikan waktu dan suhu memasak berdasarkan perubahan jadwal. Selain itu, smartphone atau koneksi tablet memungkinkan Anda terhubung ke oven secara langsung, untuk membuat penyesuaian yang diinginkan. Ketika makanan sudah siap, oven mengirimkan pesan peringatan kepada Anda (atau seseorang yang Anda tentukan) bahwa makanan sudah selesai dan dipanaskan

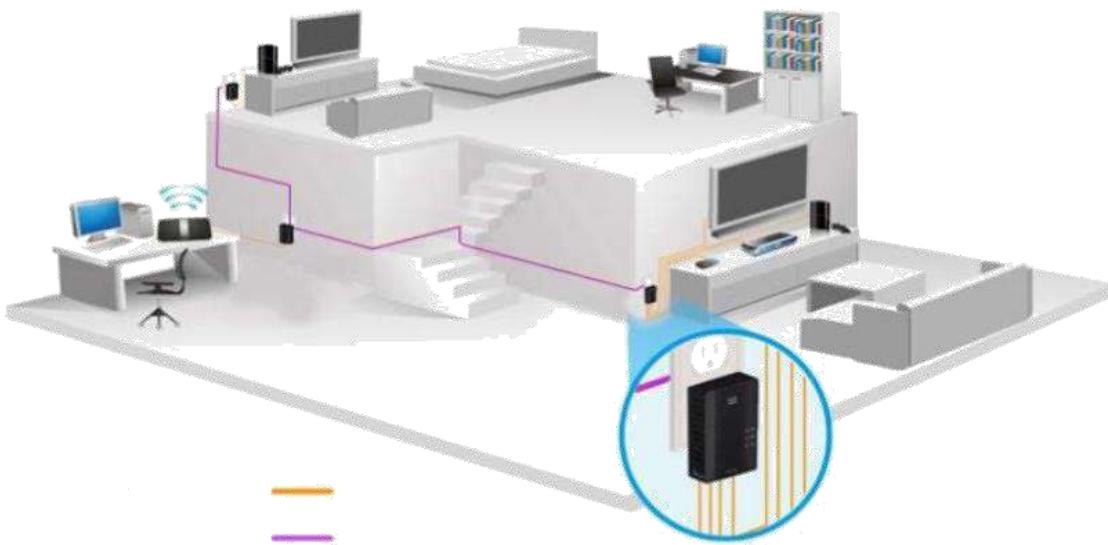
Teknologi smart home saat ini sedang dikembangkan untuk semua ruangan di dalam rumah. Itu akan menjadi lebih umum karena jaringan rumah dan teknologi internet berkecepatan tinggi berkembang.



Smartphone diperbarui dari cloud dengan status perangkat smart home dan smart car. Pengguna kemudian dapat menggunakan smartphone berinteraksi dengan smart home dan smart car. Contohnya adalah beberapa produk (bardi, TP-Link)

2. Jaringan Powerline

Jaringan saluran listrik untuk jaringan rumah menggunakan kabel listrik yang ada untuk menghubungkan perangkat, seperti yang ditunjukkan pada gambar.



Dengan menggunakan adaptor saluran listrik standar, perangkat dapat tersambung ke LAN di mana pun terdapat stopkontak listrik. Tidak ada kabel data yang perlu dipasang, dan hanya ada sedikit atau tidak ada listrik tambahan yang digunakan. Menggunakan kabel yang sama yang menyalurkan listrik, jaringan powerline mengirimkan informasi dengan mengirimkan data pada frekuensi tertentu.

Jaringan powerline sangat berguna ketika titik akses nirkabel tidak dapat menjangkau semua perangkat di rumah. Jaringan powerline bukanlah pengganti kabel khusus di jaringan data. Namun, ini adalah alternatif ketika kabel jaringan data atau komunikasi nirkabel tidak memungkinkan atau tidak efektif (Contoh : Token Listrik)

Tren Broadband Nirkabel

Di banyak area di mana kabel dan DSL tidak tersedia, nirkabel dapat digunakan untuk menyambung ke internet.

1. Penyedia Layanan Internet Nirkabel

Wireless Internet Service Provider (WISP) adalah ISP yang menghubungkan pelanggan ke titik akses atau hotspot yang ditentukan menggunakan teknologi nirkabel serupa yang ditemukan di jaringan Wireless Local Area Network (WLAN). WISP lebih umum ditemukan di lingkungan pedesaan di mana layanan *DSL* atau kabel tidak tersedia.

Meskipun menara transmisi terpisah dapat dipasang untuk antena, biasanya antena dipasang ke struktur tinggi yang ada, seperti menara air atau menara radio. Piring atau antena kecil dipasang di atap pelanggan dalam jangkauan pemancar *WISP*. Unit akses pelanggan terhubung ke jaringan kabel di dalam rumah. Dari sudut pandang pengguna rumahan, pengaturannya tidak jauh berbeda dengan *DSL* atau layanan kabel. Perbedaan utamanya adalah sambungan dari rumah ke ISP adalah nirkabel, bukan kabel fisik. (Contoh : Seorang menyewa layanan ISP lalu orang tersebut menjualkan bandwidth lagi terhadap pelanggan dengan harga yang murah)

2. Layanan Broadband Nirkabel

Solusi nirkabel lain untuk rumah dan bisnis kecil adalah broadband nirkabel, seperti yang ditunjukkan pada gambar.



Solusi ini menggunakan teknologi seluler yang sama dengan Smartphone. Antena dipasang di luar rumah yang menyediakan koneksi nirkabel atau kabel untuk perangkat di rumah. Di banyak area, broadband nirkabel rumah bersaing langsung dengan layanan DSL dan kabel.

Keamanan Jaringan

Anda pasti pernah mendengar atau membaca berita tentang jaringan perusahaan yang dibobol, memberikan akses kepada pelaku yang tidak bertanggung jawab akan menimbulkan ancaman ke informasi pribadi ribuan pelanggan. Untuk alasan ini, keamanan jaringan akan selalu menjadi prioritas utama para administrator

Ancaman keamanan

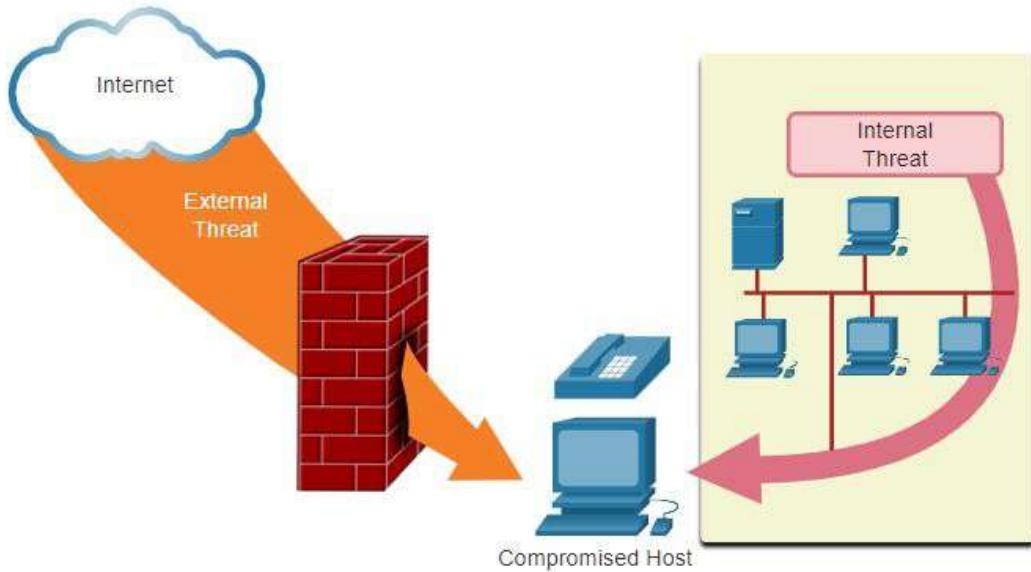
Keamanan jaringan merupakan bagian yang penting dari jaringan komputer, terlepas dari apakah jaringan tersebut berada di rumah dengan satu koneksi ke internet atau merupakan perusahaan dengan ribuan pengguna. Keamanan jaringan harus mempertimbangkan lingkungan, serta alat dan persyaratan jaringan. Keamanan harus dapat mengamankan data sambil tetapi memungkinkan kualitas layanan yang diharapkan pengguna dari jaringan.

Mengamankan jaringan melibatkan *protokol*, teknologi, perangkat, alat, dan teknik untuk melindungi data dan mengurangi ancaman. ada dua ancaman yaitu **ancaman internal** atau **eksternal**. Banyak ancaman keamanan jaringan eksternal saat ini berasal dari internet. ada beberapa ancaman eksternal yang umum terhadap jaringan :

1. **Viruses,Worms,Trojan Horses** Ini berisi perangkat lunak atau kode berbahaya yang berjalan pada perangkat pengguna.
2. **Spyware And Adware** Ini adalah jenis perangkat lunak yang diinstal di perangkat pengguna. Perangkat lunak kemudian secara diam-diam mengumpulkan informasi tentang pengguna.
3. **Zero-Day Attack** Juga disebut serangan zero-hour, ini terjadi pada hari pertama kerentanan diketahui.
4. **Denial of service attack** Serangan ini memperlambat atau merusak aplikasi dan proses pada perangkat jaringan
5. **Data interception and theft** Serangan ini menangkap informasi pribadi dari jaringan organisasi.
6. **Identity Theft** Serangan ini mencuri kredensial login pengguna untuk mengakses data pribadi.

Sama pentingnya untuk mempertimbangkan ancaman internal. Telah banyak penelitian yang menunjukkan bahwa pembobolan data yang paling umum terjadi karena pengguna internal jaringan. Hal ini dapat dikaitkan dengan perangkat yang hilang atau dicuri, penyalahgunaan yang tidak disengaja oleh karyawan, dan dalam lingkungan bisnis, bahkan karyawan yang berniat jahat. Dengan strategi BYOD yang terus berkembang, data perusahaan jauh lebih

rentan. Oleh karena itu, ketika mengembangkan sebuah kebijakan jaringan, penting untuk mengatasi ancaman external dan internal



Solusi dari ancaman jaringan

Tidak ada satupun solusi yang dapat melindungi jaringan dari berbagai ancaman yang ada. Untuk alasan ini, keamanan harus diterapkan dalam beberapa lapisan, menggunakan lebih dari satu solusi keamanan, Jika satu komponen keamanan gagal mengidentifikasi dan melindungi jaringan, komponen lainnya mungkin berhasil.

Implementasi keamanan jaringan rumah biasanya agak mendasar. Biasanya, Anda menerapkannya di end device, serta pada titik koneksi ke internet, dan bahkan dapat mengandalkan layanan yang dikontrak dari ISP (*Firewall* yang sudah disetting dari ISP).

Ini adalah komponen dasar untuk jaringan rumah atau kantor kecil :

1. **Antivirus dan antispyware** Aplikasi ini membantu melindungi end devices agar tidak terinfeksi perangkat lunak berbahaya.
2. **Firewall filtering** Aplikasi memblokir akses tidak sah ke dalam dan ke luar jaringan. Ini mungkin termasuk sistem firewall berbasis host yang mencegah akses tidak sah ke End Device, atau layanan pemfilteran dasar di router rumah untuk mencegah akses tidak sah dari dunia luar ke jaringan

Sebaliknya, implementasi keamanan jaringan untuk jaringan perusahaan biasanya terdiri dari banyak komponen yang dibangun ke dalam jaringan untuk memantau dan memfilter traffic. Idealnya, semua komponen bekerja sama, yang meminimalkan perawatan dan meningkatkan keamanan

Jaringan yang besar dan jaringan perusahaan juga menggunakan firewall filtering dan antivirus tapi ada beberapa tambahan pengamanan, diantaranya :

1. **Dedicated firewall systems** ini memberikan kemampuan firewall yang lebih canggih yang dapat menyaring arus data dalam jumlah besar dengan lebih banyak perincian
2. **Access Control List (ACL)** ini lebih lanjut memfilter akses dan penerusan arus data berdasarkan alamat IP dan aplikasi
3. **Intrusion Prevention Systems (IPS)** ini mengidentifikasi ancaman yang menyebar cepat, seperti serangan zero-day atau zero-hour.
4. **Virtual Firewall Network (VPN)** ini memberikan akses aman ke dalam organisasi untuk pekerja jarak jauh

Persyaratan keamanan jaringan harus mempertimbangkan lingkungan, serta berbagai aplikasi, dan persyaratan komputasi. Baik lingkungan rumah dan bisnis harus dapat mengamankan datanya dan tetapi membuat kualitas layanan yang diharapkan pengguna dari setiap teknologi. Selain itu, solusi keamanan yang diterapkan harus dapat beradaptasi dengan tren jaringan yang berkembang dan berubah ubah.

Profesional IT

Karir

Belajar untuk memiliki karir dibidang IT, memanglah banyak tantangannya sehingga kita harus memiliki mentor dan paduan yang cukup kuat karena kalau tidak kita bisa salah tujuan dan arah, dalam kasus ini buku ini memberitahu keterampilan yang dibutuhkan untuk mencocokan jenis pekerjaan yang tersedia di bidang IT.

Menjadi *Network Engineer*, Network Operation Center, Provisioning, InfraNetwork, Network Cyber Security adalah impian bagi orang yang sudah mempelajari ilmu jaringan. Anda dapat mengimplementasikan ilmu jaringan secara langsung sesuai dengan ilmu yang ada dibuku ini

Buku ini hanya sebagian kecil dari specialist yang ada, semisal di buku ini tidak terlalu mendalam tentang cyber security ataupun network operation center dll. Tetapi buku ini menjadikan referensi atau dasar yang dijadikan untuk menjadi profesional

Pekerjaan Jaringan

IT semakin berkembang jauh dan pesat terutama dibidang jaringan, ada beberapa pekerjaan yang penulis tau dan pernah menjadikan pengalamannya

1. Network Engineer, di posisi ini biasanya kita yang melakukan konfigurasi saat ada problem ataupun instalasi
2. Network Operation Center, di posisi ini biasanya kita melakukan monitoring 24 Jam (beserta shift) lalu melaporkan ketika ada problem di jaringan
3. Provisioning, di posisi ini biasanya ada masalah pada daerah pelanggan (fisik) ini juga bisa disebut tim lapangan, tapi terkadang jika skala pelanggannya kecil maka network engineer yang turun
4. InfraNetwork, di posisi ini biasanya ketika ada instalasi kabel atau survey lokasi, namun ketika skala pelanggannya kecil maka network engineer yang turun
5. Network Cyber Security, di posisi ini biasanya ketika jaringan sudah besar dan kompleks sehingga membutuhkan specialist dibidang security, namun ketika skala pelanggannya kecil kadang security tidak terlalu diperhatikan
6. Dan banyak hal lainnya (IT Support, IT Helpdesk, IT Field)

BAB 2

~ Basic Switch And End Device Configuration ~

Judul Bab : Dasar konfigurasi switch dan end device

Tujuan Bab : Praktek pengaturan awal termasuk kata sandi, pengalamatan IP, dan parameter gateway default pada switch dan end device.

Link Test Pemahaman : <https://s.id/Qxfn>

| Judul Materi | Tujuan Materi |
|---------------------------------|---|
| Akses Cisco iOS | Menjelaskan cara mengakses perangkat Cisco iOS. |
| Navigasi iOS | Menjelaskan cara menavigasi Cisco iOS untuk mengkonfigurasi perangkat jaringan. |
| Struktur Perintah | Menjelaskan struktur perintah perangkat lunak Cisco iOS. |
| Konfigurasi Perangkat Dasar | Mengkonfigurasikan perangkat Cisco iOS menggunakan CLI. |
| Menyimpan Konfigurasi | Menggunakan perintah iOS untuk menyimpan konfigurasi yang sedang berjalan. |
| Port dan Alamat | Menjelaskan bagaimana perangkat berkomunikasi di seluruh media jaringan. |
| Mengkonfigurasi Pengalamatan IP | Mengkonfigurasikan perangkat host dengan alamat IP. |
| Verifikasi Konektivitas | Memverifikasi koneksi antara dua End Device. |

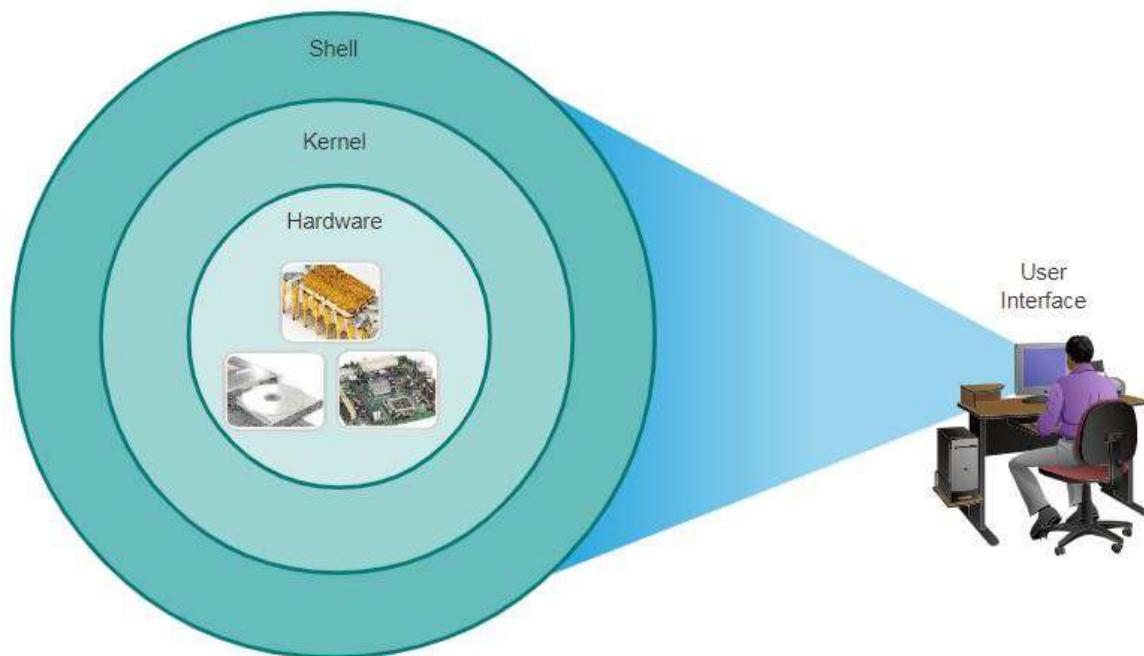
Akses Cisco IOS

Cisco merupakan perusahaan global dalam bidang telekomunikasi yang terletak pada San Jose, California (Amerika Serikat) yang memfokuskan Networking Hardware and Software. Cisco mempunyai visi yang mengubah bagaimana cara hidup, bekerja, bermain dan belajar yang mempunyai motto Welcome to The Human Network. Cisco didirikan oleh Leonard Bosack,Sandy Lerner,Chuck Robbins

Setelah kita mengetahui apa itu cisco sekarang kita belajar tentang sistem yang ada di dalam cisco

Sistem Operasi

Semua End Device dan intermediary device memerlukan ***Operating System (OS)***. Seperti yang ditunjukkan pada gambar, OS yang berinteraksi langsung dengan perangkat keras komputer dikenal sebagai ***kernel***. Bagian yang berinteraksi dengan aplikasi dan pengguna dikenal sebagai ***shell***. Pengguna dapat berinteraksi dengan ***shell*** menggunakan **Graphical User Interface** atau **Command Line Interface (CLI)**



Shell

User Interface yang memungkinkan pengguna untuk meminta tugas tertentu dari komputer, bisa memakai GUI atau CLI

CLI (Command Line Interface)

CLI seperti ubuntu-server, DOS, Windows Core server membutuhkan sedikit resource sehingga sangat stabil jika dibandingkan dengan GUI. Oleh karena itu Keluarga **Operating system** yang digunakan pada banyak perangkat Cisco disebut Cisco Internetwork Operating System (IOS). Cisco IOS digunakan pada banyak **router** dan **switch** cisco terlepas dari jenis atau ukuran perangkatnya. Setiap router perangkat atau jenis switch menggunakan versi Cisco IOS yang berbeda. Operating system cisco lainnya termasuk IOS XE, IOS XR, dan NX-OS

```
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1
000
    link/ether 08:00:27:91:82:de brd ff:ff:ff:ff:ff:ff
    inet 192.168.137.78/24 brd 192.168.137.255 scope global enp0s3
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe91:82de/64 scope link tentative
        valid_lft forever preferred_lft forever
root@debian:~# ping 1.1.1.1
PING 1.1.1.1 (1.1.1.1) 56(84) bytes of data.
64 bytes from 1.1.1.1: icmp_seq=1 ttl=57 time=4.25 ms
64 bytes from 1.1.1.1: icmp_seq=2 ttl=57 time=4.51 ms
^C
--- 1.1.1.1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 4.254/4.386/4.518/0.132 ms
root@debian:~# ping 192.168.137.78
PING 192.168.137.78 (192.168.137.78) 56(84) bytes of data.
64 bytes from 192.168.137.78: icmp_seq=1 ttl=64 time=0.015 ms
^C
--- 192.168.137.78 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.015/0.015/0.015/0.000 ms
root@debian:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default 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: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1
000
    link/ether 08:00:27:91:82:de brd ff:ff:ff:ff:ff:ff
    inet 192.168.137.78/24 brd 192.168.137.255 scope global enp0s3
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe91:82de/64 scope link
        valid_lft forever preferred_lft forever
root@debian:~# _
```

GUI (Graphic User Interface)

GUI seperti Windows, macOS, Linux KDE, Apple iOS, atau Android memungkinkan pengguna untuk berinteraksi dengan sistem menggunakan ikon grafis, menu, dan jendela. Contoh GUI pada gambar lebih ramah pengguna dan membutuhkan lebih sedikit pengetahuan tentang struktur perintah dasar yang mengontrol sistem. Untuk alasan ini, sebagian besar pengguna mengandalkan lingkungan GUI.



Note: OS pada router rumah biasanya disebut firmware. Metode yang paling umum mengkonfigurasi router rumah adalah dengan menggunakan GUI berbasis web.

Kernel

Berkomunikasi antara perangkat keras (hardware) dan perangkat lunak (software) komputer dan mengelola bagaimana sumber daya perangkat keras digunakan untuk memenuhi persyaratan perangkat lunak.

Hardware

Bagian fisik komputer termasuk elektronik yang mendasarinya dan bisa kita lihat dan pegang.

Tujuan dibuatnya OS

OS jaringan mirip dengan OS komputer. Melalui GUI, OS komputer memungkinkan pengguna untuk melakukan hal berikut :

1. Menggunakan mouse untuk membuat pilihan dan menjalankan program
2. Memasukan teks dan perintah berbasis teks
3. Lihat Output pada monitor

OS jaringan berbasis CLI (misalnya, cisco IOS pada switch atau router) memungkinkan teknisi jaringan untuk melakukan hal berikut:

1. Menggunakan keyboard untuk menjalankan program jaringan berbasis CLI
2. Menggunakan keyboard untuk memasukkan teks dan perintah berbasis teks
3. Lihat output di monitor

Perangkat jaringan cisco menjalankan versi tertentu dari cisco IOS. Versi IOS tergantung pada jenis perangkat yang digunakan dan fitur yang diperlukan. Meskipun semua perangkat dilengkapi dengan IOS default dan rangkaian fitur untuk mendapatkan kemampuan tambahan.

Contoh daftar rilis perangkat lunak IOS untuk Cisco Catalyst 2960 Switch

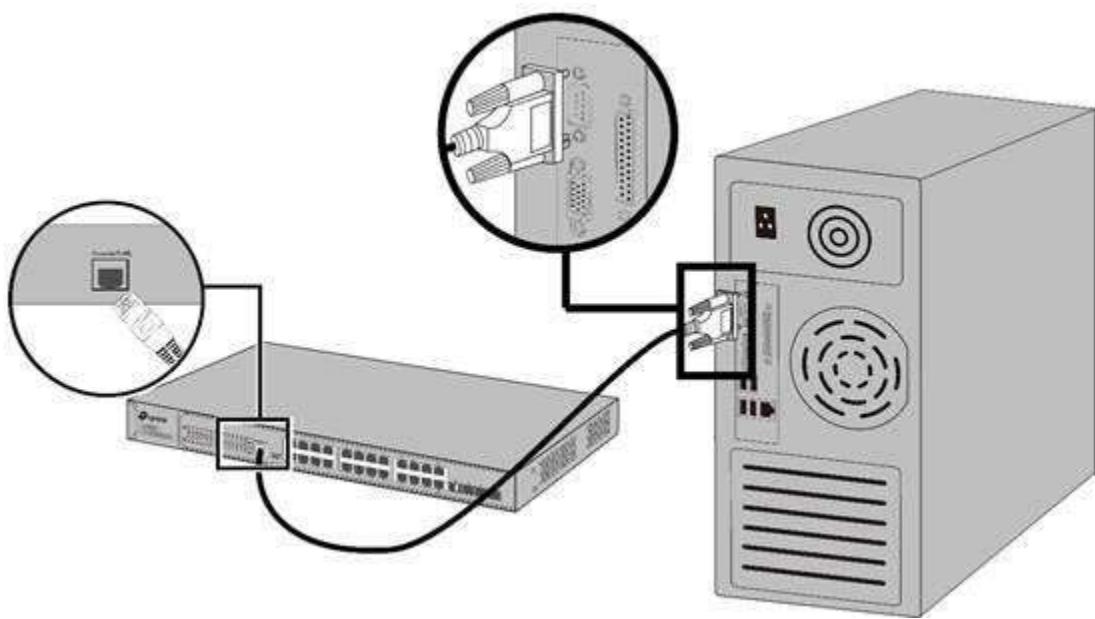
The screenshot shows the Cisco Software Download page for the Catalyst 2960-24LC-S Switch. On the left, there's a sidebar with a search bar, 'Expand All' and 'Collapse All' buttons, and dropdown menus for 'Latest Release' (set to 6.3.4) and 'All Release' (showing options 5.8.9.1, 6, and 5). The main content area displays the 'Catalyst 2960-24LC-S Switch' page for Release 6.3.4. It includes a star rating section, 'Related Links and Documentation', and 'Release Notes for 6.3.4'. Below this, a table lists file information for two installers: 'Network Assistant English Mac Installer' (cna-mac-k9-installer-6-3-4-en.zip) and 'Network Assistant English Installer' (cna-windows-k9-installer-6-3-4-en.exe). Both files were released on 19-Dec-2018 and have sizes of 34.43 MB and 70.35 MB respectively. At the bottom, there's a footer with links to 'Contacts', 'Feedback', 'Help', 'Site Map', 'Terms & Conditions', 'Privacy Statement', 'Cookie Policy', and 'Trademarks'.

Metode Akses (Cara mengakses perangkat jaringan)

Sebuah switch akan meneruskan traffic secara default dan tidak perlu dikonfigurasi secara *eksplisit* untuk beroperasi. Misalnya, dua host terkonfigurasi yang terhubung ke switch yang sama akan dapat berkomunikasi. Terlepas dari default switch, semua switch harus di config dan diamankan

Console

Console adalah port manajemen fisik yang menyediakan akses *out-of-band* ke perangkat Cisco. Akses *out-of-band* mengacu pada akses melalui saluran manajemen khusus yang digunakan hanya untuk tujuan pemeliharaan perangkat. Keuntungan menggunakan port console adalah perangkat dapat diakses meskipun tidak ada layanan jaringan yang dikonfigurasi, seperti melakukan konfigurasi awal.



Secure Shell (SSH)

SSH adalah in-band dan metode yang direkomendasikan untuk membuat sambungan CLI aman dari jarak jauh, melalui *Virtual Interface*, melalui jaringan. Tidak seperti koneksi konsol, koneksi SSH memerlukan layanan jaringan aktif pada perangkat, termasuk **interface** yang aktif dikonfigurasi dengan alamat (IP). Sebagian besar versi Cisco IOS menyertakan server SSH dan client SSH yang dapat digunakan untuk membuat sesi SSH dengan perangkat lain

Telnet

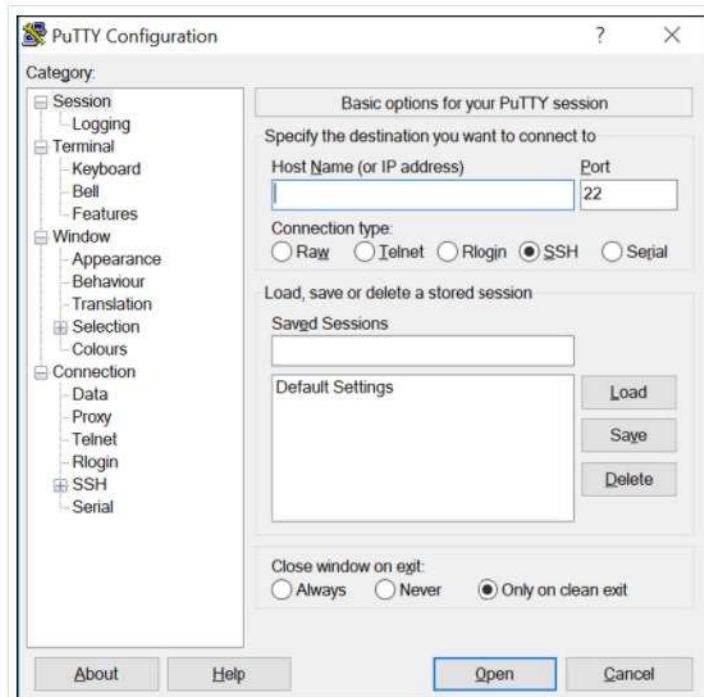
Telnet adalah in-band yang tidak aman. Metode untuk membuat sesi CLI dari jarak jauh, melalui **Virtual Interface**, melalui jaringan. Tidak seperti SSH, Telnet tidak menyediakan koneksi *terenkripsi* yang aman dan sebaiknya hanya digunakan di lingkungan lab. Otentifikasi pengguna, kata sandi, dan perintah dikirim melalui jaringan dalam bentuk *Plain Text*

Note : Beberapa perangkat, seperti router, mungkin juga mendukung port tambahan yang sudah lama digunakan untuk membuat sesi CLI dari jarak jauh, melalui koneksi telepon menggunakan modem. Mirip dengan koneksi console, Port AUX out-of-band dan tidak memerlukan layanan jaringan untuk dikonfigurasi atau tersedia

Program Terminal Emulator

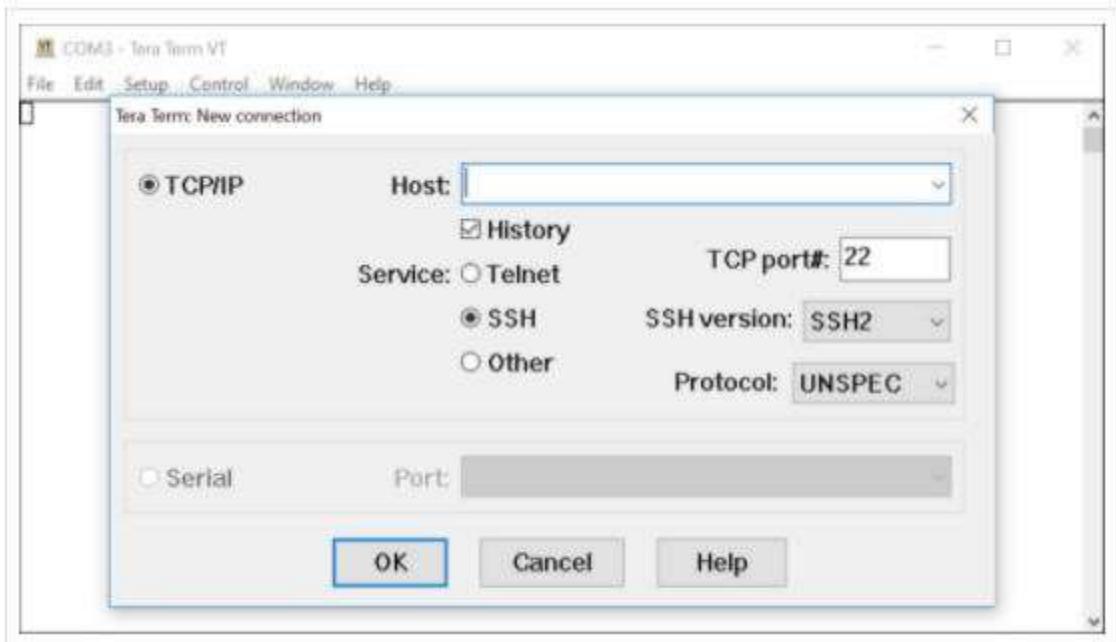
Beberapa program emulasi terminal yang dapat Anda gunakan untuk menghubungkan ke perangkat jaringan baik dengan koneksi serial melalui port **console**, atau dengan koneksi **SSH / Telnet**. Program-program ini memungkinkan Anda meningkatkan produktivitas dengan menyesuaikan ukuran jendela, mengubah ukuran font, dan mengubah skema warna

Putty



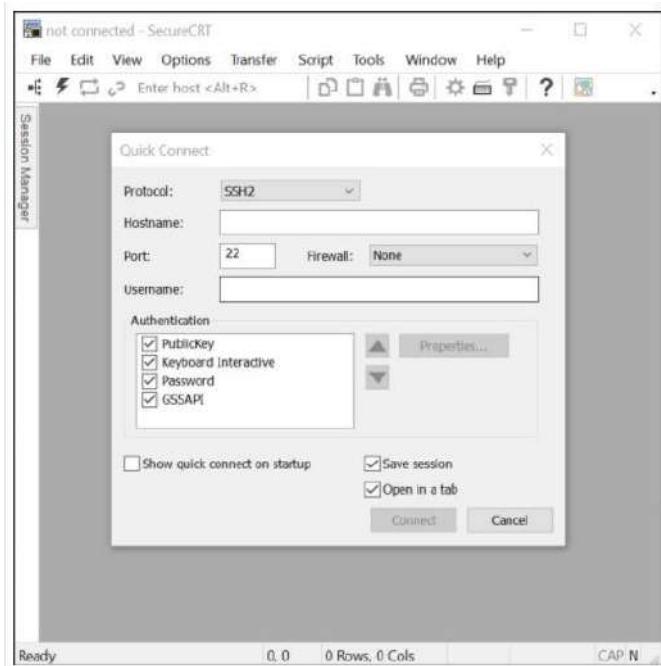
www.putty.org

Terra term



ttssj2.osdn.jp

SecureCRT



www.vandyke.com

Navigasi IOS Cisco

Anda telah mengetahui bahwa semua perangkat jaringan memerlukan OS dan dapat dikonfigurasi menggunakan CLI atau GUI. Menggunakan CLI dapat memudahkan administrator jaringan mengontrol dengan fleksibilitas yang lebih tepat dari pada menggunakan GUI. Materi ini membahas penggunaan CLI untuk menavigasi Cisco IOS.

Primary Command Modes

Sebagai fitur keamanan, perangkat lunak cisco memisahkan akses manajemen ke dalam dua mode perintah berikut

User EXEC Mode

Mode ini memiliki kemampuan terbatas tetapi berguna untuk pengoperasian dasar. Ini hanya mengizinkan sejumlah perintah pemantauan “show” dasar tetapi tidak mengizinkan eksekusi perintah apapun yang mungkin mengubah konfigurasi perangkat. Ini juga sering disebut sebagai mode “Read Only”

```
Motherboard serial number      : FOC103248MJ
Power supply serial number    : DCA102133JA
Model revision number         : B0
Motherboard revision number   : C0
Model number                  : WS-C2960-24TT
System serial number          : FOC1033Z1EY
Top Assembly Part Number     : 800-26671-02
Top Assembly Revision Number : B0
Version ID                   : V02
CLEI Code Number              : COM3K00BRA
Hardware Board Revision Number: 0x01

Switch  Ports  Model           SW Version       SW Image
-----  -----  -----           -----
*      1      26    WS-C2960-24TT  12.2           C2960-LANBASE-
M

Cisco IOS Software, C2960 Software (C2960-LANBASE-M), Version 12.2(25)FX,
RELEASE SOFTWARE (fc1)
Copyright (c) 1986-2005 by Cisco Systems, Inc.
Compiled Wed 12-Oct-05 22:05 by pt_team

Press RETURN to get started!

Switch>
```

Privileged EXEC Mode

Mode ini memungkinkan akses ke semua perintah dan fitur, dan juga pengguna dapat menggunakan perintah pemantauan “*show*” dan menjalankan konfigurasi “*configure*” sampai perintah management (membuat user dan password)

```
Power supply serial number      : DCA102133JA
Model revision number          : B8
Motherboard revision number    : C8
Model number                   : WS-C2960-24TT
System serial number           : FOC1033Z1EY
Top Assembly Part Number      : 800-26671-02
Top Assembly Revision Number  : B8
Version ID                     : V82
CLEI Code Number               : COM3K00BRA
Hardware Board Revision Number: 0x01

Switch  Ports  Model           SW Version      SW Image
-----  ----   -----           -----
*     1      26    WS-C2960-24TT  12.2           C2960-LANBASE-
M

Cisco IOS Software, C2960 Software (C2960-LANBASE-M), Version 12.2(25)FX,
RELEASE SOFTWARE (fc1)
Copyright (c) 1986-2005 by Cisco Systems, Inc.
Compiled Wed 12-Oct-05 22:05 by pt_team

Press RETURN to get started!

Switch>enable
Switch#
```

Konfigurasi Mode dan Sub Konfigurasi Mode

Untuk mengkonfigurasi perangkat, pengguna harus masuk ke mode konfigurasi global, yang biasa disebut Global configuration

Dari Global Configuration, perubahan konfigurasi CLI dibuat yang mempengaruhi pengoperasian perangkat secara keseluruhan. Global Configuration dikenali oleh *prompt* yang diakhiri dengan **(config)#** setelah *nama perangkat*, seperti **Switch(config)#**

Global Configuration diakses sebelum mode konfigurasi spesifik lainnya (line dan interface config), dari mode konfigurasi global, pengguna dapat memasuki mode sub konfigurasi yang berbeda. Masing masing mode ini memungkinkan konfigurasi bagian atau fungsi tertentu dari perangkat IOS. Dua mode sub konfigurasi umum meliputi:

- **Line Configuration Mode** – Digunakan untuk mengkonfigurasi akses Console, SSH, Telnet, atau AUX
- **Interface Configuration Mode** – Digunakan untuk mengkonfigurasi port switch atau antarmuka jaringan router

Saat CLI digunakan, mode diidentifikasi oleh *prompt* baris perintah yang unik untuk mode tersebut. Secara default, setiap *prompt* dimulai dengan nama perangkat, Mengikuti nama, sisa *prompt* menunjukkan mode. Misalnya, *prompt* default untuk **Line Configuration Mode** adalah **Switch (Config-line) #** dan prompt default untuk **Interface configuration Mode** adalah **switch (config-if)#**

contoh Line Configuration Mode

```
Switch>enable
Switch#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#line console 0
Switch(config-line)#
```

contoh Interface Configuration Mode

```
Switch>enable
Switch#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#interface fastethernet 0/1
Switch(config-if)#
```

Navigasi dengan mode IOS

Berbagai perintah digunakan untuk masuk dan keluar dari *prompt*. Untuk berpindah dari User EXEC Mode ke Privileged Exec Mode, Gunakan perintah **enable**, Gunakan **disable** untuk sebaliknya

Note : Mode Privileged EXEC terkadang disebut sebagai *enable mode*

Untuk masuk ke Global Configuration, gunakan perintah **configure terminal** dari mode Privileged Exec Mode, untuk mengembalikan ke Privileged Exec Mode lagi gunakan perintah **exit**.

Ada banyak mode konfigurasi. Misalnya, untuk masuk ke Line Configuration mode, Anda menggunakan perintah **Line** diikuti dengan jenis dan nomor yang ingin anda akses. Gunakan perintah **exit** untuk keluar dari mode sub configuration dan kembali ke Global configuration

```
Switch(config)# line console 0
Switch(config-line)# exit
Switch(config)#
```

Untuk berpindah dari mode sub konfigurasi apa pun dari **Global configuration** ke mode satu langkah di atasnya dalam *hierarki* mode, masukkan perintah **exit**. Untuk berpindah dari mode sub konfigurasi apapun ke **Privileged Exec Mode**, masukkan perintah **end** atau masukkan kombinasi tombol **Ctrl + Z**.

```
Switch(config-line)# end
Switch#
```

Anda juga dapat berpindah langsung dari satu mode sub konfigurasi ke mode lainnya. Perhatikan bagaimana setelah memilih sebuah *Interface, prompt* berubah dari **(config-line) #** ke **(config-if) #**.

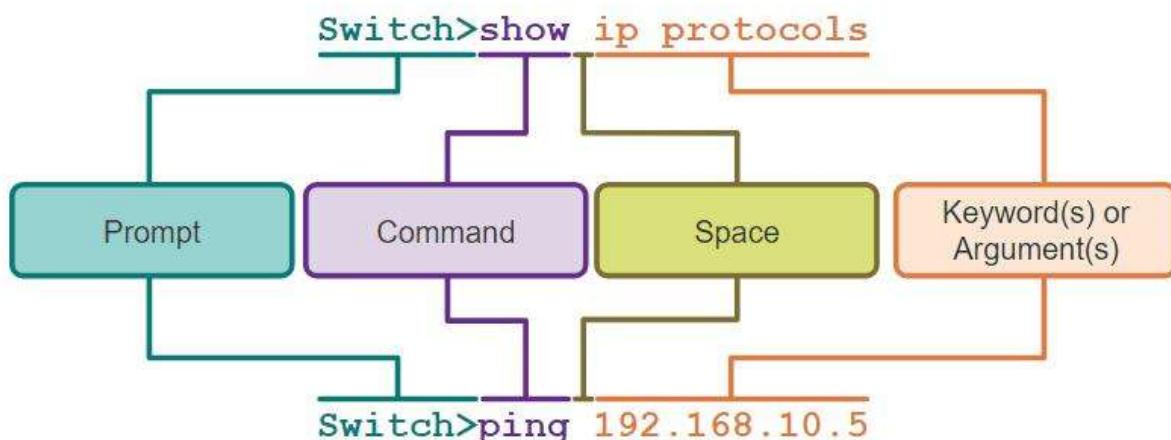
```
Switch(config-line)# interface FastEthernet 0/1
Switch(config-if)#
```

Struktur Command Line IOS

Pembahasan kali ini mencakup struktur dasar command untuk Cisco IOS. Administrator jaringan harus mengetahui struktur command IOS dasar agar dapat menggunakan CLI untuk konfigurasi perangkat

Dasar Command Line IOS

Perangkat Cisco IOS mendukung banyak command. Setiap command IOS memiliki format atau *sintaks* tertentu, dan hanya dapat dijalankan dalam mode yang sesuai. *Sintaks* umum untuk sebuah command, yang ditunjukkan pada gambar, adalah command yang diikuti dengan Keyword(s) dan Argument(s) yang sesuai.



- **Keyword** – Ini adalah parameter khusus yang ditentukan dalam Operating system (ip protocols)
- **Argument** – Ini belum ditentukan sebelumnya (menentukan sendiri); argument adalah nilai atau variabel yang ditentukan oleh pengguna (192.168.10.5)

Setelah memasukan setiap command lengkap, termasuk Keyword dan argument apa pun, tekan tombol *enter* untuk mengirimkan command ke perangkat maka perangkat akan segera menjalankannya

Fitur Help Pada IOS

IOS memiliki dua bentuk fitur Help yang tersedia: **context-sensitive** dan **command syntax check**

Context-sensitive adalah bantuan yang memungkinkan anda dengan cepat menemukan jawaban atas pertanyaan-pertanyaan ini

- command mana yang tersedia di setiap command mode
- command mana yang dimulai dengan karakter atau kelompok karakter tertentu
- Argument dan Keyword yang tersedia untuk command tertentu

Untuk mengakses bantuan peka konteks, cukup masukan tanda tanya ?, di CLI

Command syntax checker memverifikasi bahwa command yang valid telah dimasukkan oleh pengguna. Saat command dimasukkan, command akan dievaluasi dari kiri ke kanan. Jika command telah memahami perintah tersebut, tindakan yang diminta akan dijalankan, dan CLI kembali ke prompt yang sesuai. Namun, jika command tidak dapat memahami perintah yang dimasukkan, itu akan memberikan umpan balik yang menjelaskan apa yang salah dengan command tersebut

Hot Keys and shortcuts

IOS CLI menyediakan shortcuts dan shortcuts yang mempermudah konfigurasi, pemantauan, dan *troubleshooting*, perintah dan kata kunci yang dapat dipersingkat menjadi jumlah minimum karakter yang mengidentifikasi pilihan unik. Misalnya, perintah *configure* dapat disingkat menjadi *conf* karena *configure* adalah satu satunya perintah yang dimulai dengan *conf*. Versi yang lebih pendek adalah *con* tetapi tidak akan berfungsi karena lebih dari satu perintah dimulai dengan *con*. Keywords juga bisa dipersingkat.

Ini adalah contoh contoh shortcut pada IOS Cisco:

| Kombinasi Tombol | Deskripsi |
|--|---|
| Tab | Melengkapi entri command |
| Backspace | Menghapus karakter disebelah kiri kursor |
| Ctrl + D | Menghapus karakter disebelah kanan kursor |
| Ctrl + K | Menghapus semua karakter dari kursor hingga akhir baris perintah |
| Esc + D | Menghapus semua karakter dari kursor hingga akhir baris perintah |
| Ctrl + (U/X) | Menghapus semua karakter dari kursor kembali ke awal command |
| Ctrl + W | Menghapus kata di sebelah kiri kursor |
| Ctrl + A | Memindahkan kursor ke awal baris |
| Panah kiri atau Ctrl + B | Memindahkan kursor ke satu kata dari kiri |
| Esc + B | Memindahkan kursor mundur satu kata ke kiri. |
| Esc + F | Memindahkan kursor ke depan satu kata ke kanan. |
| Panah kanan atau Ctrl + F | Memindahkan kursor ke satu kata dari kanan |
| Ctrl + E | Memindahkan kursor ke akhir baris perintah. |
| Panah atas atau Ctrl + P | Mengingat perintah sebelumnya di buffer riwayat, dimulai dengan perintah terbaru. |
| Panah bawah atau Ctrl + N | Sebaliknya dengan panah atas |
| Ctrl + R atau Ctrl + I atau Ctrl + L | Menampilkan kembali prompt sistem dan baris perintah setelah pesan konsol diterima. |

Catatan : Meskipun tombol DEL biasanya menghapus karakter di sebelah kanan prompt, struktur perintah IOS tidak mengenali tombol DEL

Ketika output perintah menghasilkan lebih banyak teks daripada yang dapat ditampilkan di jendela terminal, IOS akan menampilkan prompt “–More–” . Tabel berikut menjelaskan penekanan tombol yang dapat digunakan saat prompt ini ditampilkan.

```
63488K bytes of flash-simulated non-volatile configuration memory.
Base ethernet MAC Address      : 0003.E4DD.7E02
Motherboard assembly number    : 73-9832-06
Power supply part number       : 341-0097-02
Motherboard serial number      : FOC103248MJ
Power supply serial number     : DCA102133JA
Model revision number         : B0
Motherboard revision number   : C0
Model number                   : WS-C2960-24TT
System serial number          : FOC1033Z1EY
--More-- |
```

| Kombinasi Tombol | Deskripsi |
|-------------------------|---------------------|
| Enter | Menampilkan 1 baris |
| Space | Menampilkan 1 layar |
| Selain tombol diatas | Mengakhiri tampilan |

Kombinasi kombinasi tombol untuk keluar dari operasi command

| Kombinasi Tombol | Deskripsi |
|-------------------------|---|
| Ctrl + C | <ul style="list-style-type: none"> Saat berada dalam mode konfigurasi apa pun, akhiri mode konfigurasi dan kembali ke Privileged Exec Mode. Saat dalam mode setup, batalkan kembali ke prompt perintah. |
| Ctrl + Z | <ul style="list-style-type: none"> Saat berada dalam mode konfigurasi apa pun, akhiri mode konfigurasi dan kembali ke Privileged Exec Mode. |
| Ctrl + Shift + 6 | <ul style="list-style-type: none"> Digunakan untuk membatalkan pencarian DNS, traceroutes, ping, dll. |

Konfigurasi Dasar Perangkat

Anda telah banyak belajar tentang Cisco IOS, navigasi IOS, dan struktur command. Sekarang, Anda siap untuk mengkonfigurasi perangkat! Perintah konfigurasi pertama pada perangkat apapun adalah harus memberinya **nama perangkat** atau nama host yang unik.

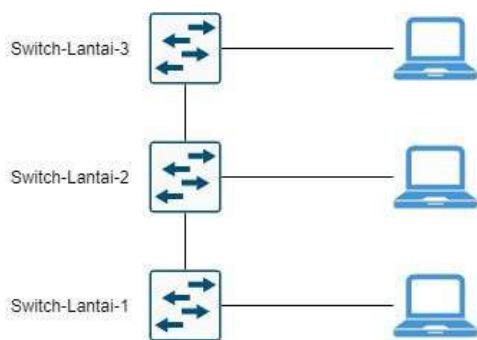
Nama Perangkat

Secara Default, semua perangkat memiliki nama default, Misalnya, Switch Cisco IOS adalah “Switch”. Masalahnya adalah jika semua switch dalam jaringan dibiarkan dengan nama defaultnya, akan sulit mengidentifikasi perangkat tertentu. Misalnya, bagaimana Anda tahu bahwa Anda terhubung ke perangkat yang tepat saat mengaksesnya dari jarak jauh menggunakan SSH ?

Nama Host memberikan konfirmasi bahwa Anda terhubung ke perangkat yang benar. Nama default harus diubah menjadi sesuatu yang lebih deskriptif. Dengan memilih nama secara bijak, lebih mudah untuk mengingat, mendokumentasikan, dan mengidentifikasi perangkat jaringan. Berikut adalah beberapa aturan penamaan penting untuk host :

- Mulai dengan Huruf
- Nggak ada spasi
- Hanya bisa memakai huruf, angka, garis
- Kurang dari 64 kata untuk penamaannya

Administrator harus memilih penamaan yang membuat mudah dan intuitif untuk mengidentifikasi perangkat tertentu. Nama host yang digunakan di perangkat IOS mempertahankan huruf besar dan karakter huruf kecil. Misalnya, gambar tersebut menunjukkan bahwa 3 Switch, yang mencakup tiga lantai berbeda, saling terhubung bersama dalam suatu jaringan



Buatlah penamaan yang digunakan menggabungkan lokasi dan tujuan setiap perangkat. Dokumentasi jaringan harus menjelaskan bagaimana nama-nama ini dipilih sehingga perangkat tambahan dapat diberi nama yang sesuai

Ketika penamaan perangkat telah diidentifikasi, langkah selanjutnya adalah menggunakan CLI untuk menerapkan nama ke perangkat. dari Privileged Exec Mode ke Global Configuration dengan memasukan perintah seperti ini

```
| Switch>enable  
| Switch#configure terminal  
| Enter configuration commands, one per line. End with CNTL/Z.  
| Switch(config)#hostname Sw-Switch-Floor1  
| Sw-Switch-Floor1(config)#
```

Note: Untuk mengembalikan namanya ke default bisa menggunakan command *no hostname*

Selalu pastikan dokumentasi diperbarui setiap kali perangkat ditambahkan atau dimodifikasi. Identifikasi perangkat dalam dokumentasi menurut lokasi, tujuan, dan alamatnya.

Paduan untuk membuat Password

Penggunaan kata sandi yang lemah atau mudah ditebak terus menjadi perhatian keamanan terbesar dalam jaringan. Perangkat jaringan, termasuk router nirkabel rumah, harus selalu memiliki kata sandi yang dikonfigurasi untuk membatasi akses administratif.

Cisco IOS dapat dikonfigurasi untuk menggunakan kata sandi mode *hierarki* untuk memungkinkan hak akses yang berbeda ke perangkat jaringan.

Semua perangkat jaringan harus membatasi akses administratif dengan mengamankan **Privileged Exec Mode**, **User EXEC mode**, dan **akses Telnet** dengan kata sandi. Selain itu, semua password harus dienkripsi dan pemberitahuan hukum disediakan (MOTD).

Saat memilih password, gunakan password yang kuat yang tidak mudah ditebak. Ada beberapa hal penting yang perlu dipertimbangkan saat memilih password:

- Gunakan password yang panjangnya lebih dari delapan karakter.
- Gunakan kombinasi huruf besar dan kecil, angka, karakter khusus, dan / atau urutan numerik.
- Hindari menggunakan kata sandi yang sama untuk semua perangkat.
- Jangan gunakan kata-kata umum karena mudah ditebak.

Gunakan google untuk menemukan pembuat kata sandi. Banyak yang memungkinkan Anda untuk mengatur panjang, kumpulan karakter, dan parameter lainnya.

Konfigurasi Password

Saat Anda pertama kali menyambungkan ke perangkat, Anda berada dalam **user EXEC mode**. Mode ini diamankan menggunakan console.

Untuk mengamankan **User exec mode**, masuk ke **Privileged Exec Mode** lalu masuk ke **Global Configuration** setelah itu masuk ke **Line configuration Console** menggunakan perintah **line console 0**, seperti yang ditunjukkan dalam contoh. Nol digunakan untuk mewakili interface console pertama (dan dalam banyak kasus satu-satunya). Selanjutnya, tentukan password user EXEC menggunakan perintah password Terakhir, aktifkan akses EXEC pengguna menggunakan perintah login .

```
Sw-Floor-1# configure terminal
Sw-Floor-1(config)# line console 0
Sw-Floor-1(config-line)# password cisco
Sw-Floor-1(config-line)# login
Sw-Floor-1(config-line)# end
Sw-Floor-1#
```

Akses console sekarang akan membutuhkan kata sandi sebelum mengizinkan akses ke user EXEC mode

Untuk memiliki akses administrator ke semua perintah IOS termasuk mengkonfigurasi perangkat, Anda harus mendapatkan akses **Privileged Exec Mode**. Ini adalah metode akses yang paling penting karena menyediakan akses lengkap ke perangkat.

Untuk mengamankan akses **Privileged Exec Mode**, gunakan perintah **enable** di **Global Configuration**, seperti yang ditunjukkan dalam contoh.

```
Sw-Floor-1# configure terminal
Sw-Floor-1(config)# enable secret class
Sw-Floor-1(config)# exit
Sw-Floor-1#
```

Virtual Terminal (VTY) memungkinkan akses jarak jauh (remote) menggunakan Telnet atau SSH ke perangkat. Banyak switch Cisco mendukung hingga 16 jalur (line) VTY yang diberi nomor 0 hingga 15.

Untuk mengamankan jalur VTY, masuk ke Line VTY Mode menggunakan perintah **Line vty 0 15** di **Global Configuration**. Selanjutnya, tentukan kata sandi VTY menggunakan perintah **password**. Terakhir, aktifkan akses VTY menggunakan perintah **login**.

Contoh pengamanan line VTY pada switch

```
Sw-Floor-1# configure terminal
Sw-Floor-1(config)# line vty 0 15
Sw-Floor-1(config-line)# password cisco
Sw-Floor-1(config-line)# login
Sw-Floor-1(config-line)# end
Sw-Floor-1#
```

Banner MOTD

Meskipun meminta password adalah salah satu cara untuk mencegah personil yang tidak berwenang keluar dari jaringan, penting untuk menyediakan metode untuk menyatakan bahwa hanya personil yang berwenang yang boleh mencoba mengakses perangkat. Untuk melakukan ini, tambahkan banner motd ke perangkat.

Banner dapat menjadi bagian penting dari proses hukum jika seseorang dituntut karena membobol perangkat. Beberapa sistem hukum tidak mengizinkan penuntutan, atau bahkan pemantauan pengguna, kecuali pemberitahuan terlihat.

Untuk membuat pesan spanduk atau peringatan di perangkat jaringan, gunakan perintah **banner motd#jangan masuk kesini#** di Global Configuration “#” dalam sintaks perintah disebut karakter pembatas. Itu dimasukkan sebelum dan sesudah pesan. Karakter pembatas dapat berupa karakter apa saja selama tidak muncul dalam pesan. Untuk alasan ini, simbol seperti “#” sering digunakan. Setelah perintah dijalankan, spanduk akan ditampilkan pada semua upaya berikutnya untuk mengakses perangkat hingga Banner dihapus.

Contoh berikut menunjukkan langkah-langkah untuk mengkonfigurasi banner di Sw-Floor-1.

```
Sw-Floor-1# configure terminal
Sw-Floor-1(config)# banner motd #jangan masuk kesini#
```

Menyimpan Konfigurasi

Anda sekarang tahu bagaimana melakukan konfigurasi dasar pada sebuah switch, termasuk password dan banner motd. Materi ini akan menunjukkan cara menyimpan konfigurasi Anda.

File Konfigurasi

Sebelum itu di sebuah perangkat sistem ada pasti ada yang namanya file. Ada dua file sistem yang menyimpan konfigurasi perangkat:

- **startup-config** – Ini adalah file konfigurasi yang disimpan di NVRAM. Ini berisi semua perintah yang akan digunakan oleh perangkat saat startup atau reboot. switch tidak kehilangan kontennya saat perangkat dimatikan.
- **running-config** – Ini disimpan dalam Random Access Memory (RAM). Ini mencerminkan konfigurasi saat ini. Mengubah konfigurasi yang sedang berjalan mempengaruhi pengoperasian perangkat Cisco dengan segera. RAM akan kehilangan semua kontennya saat perangkat dimatikan atau dimulai ulang.

show running-config di privileged EXEC mode digunakan untuk melihat konfigurasi saat ini. Seperti yang ditunjukkan pada contoh, perintah akan mencantumkan konfigurasi lengkap yang saat ini disimpan dalam RAM.

Untuk melihat file konfigurasi startup, gunakan perintah **show startup-config** di privileged EXEC mode

```
Sw-Floor-1# show running-config
Building configuration...
Current configuration : 1351 bytes
!
! Last configuration change at 00:01:20 UTC Mon Mar 1 1993
!
version 15.0
no service pad
service timestamps debug datetime msec
service timestamps log datetime msec
service password-encryption
!
hostname Sw-Floor-1
!
(output omitted)
```

Jika daya ke perangkat hilang, atau jika perangkat di-restart, semua perubahan konfigurasi akan hilang kecuali jika telah disimpan. Untuk menyimpan perubahan yang dibuat pada konfigurasi yang sedang berjalan ke file konfigurasi startup, gunakan perintah **copy running-config startup-config**

Running Configuration dan Startup Configuration

Jika perubahan yang dibuat pada Running Configuration tidak memberikan efek yang diinginkan dan Running Configuration belum disimpan, Anda dapat memulihkan perangkat ke konfigurasi sebelumnya. Hapus perintah yang diubah satu per satu, atau muat ulang perangkat menggunakan perintah **reload** untuk memulihkan **startup-config**

Kelemahan menggunakan perintah **reload** untuk menghapus running-config yang tidak disimpan adalah waktu singkat perangkat akan offline, menyebabkan waktu henti jaringan.

Saat reload dimulai, IOS akan mendeteksi bahwa konfigurasi saat ini memiliki perubahan yang tidak disimpan ke startup-config. Sebuah prompt akan muncul untuk menanyakan apakah akan menyimpan perubahan. Tekan Y untuk yes dan tekan N untuk no

```
Sw-Switch-Floor1#reload  
System configuration has been modified. Save? [yes/no]:
```

Alternatifnya, jika perubahan yang tidak diinginkan disimpan ke startup-config, mungkin perlu untuk menghapus semua konfigurasi. Ini membutuhkan penghapusan startup-config dan memulai ulang perangkat. Konfigurasi startup dihapus dengan menggunakan perintah **erase startup-config**. Setelah perintah dikeluarkan, switch akan meminta Anda untuk konfirmasi. Tekan Enter untuk menerima.

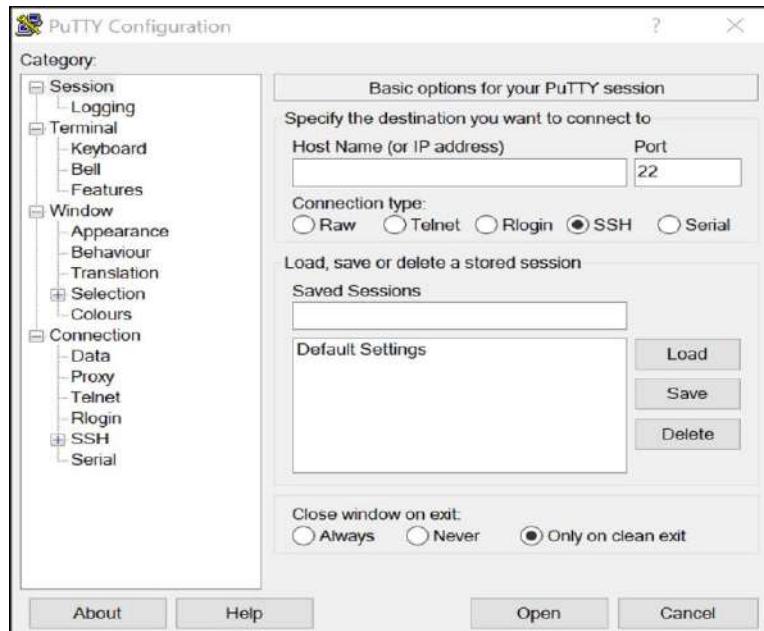
Setelah menghapus konfigurasi startup dari NVRAM, reboot perangkat untuk menghapus file konfigurasi yang sedang berjalan dari RAM. Saat memuat ulang, switch akan memuat konfigurasi startup default yang awalnya dikirimkan bersama perangkat

Menyimpan Konfigurasi ke File Teks

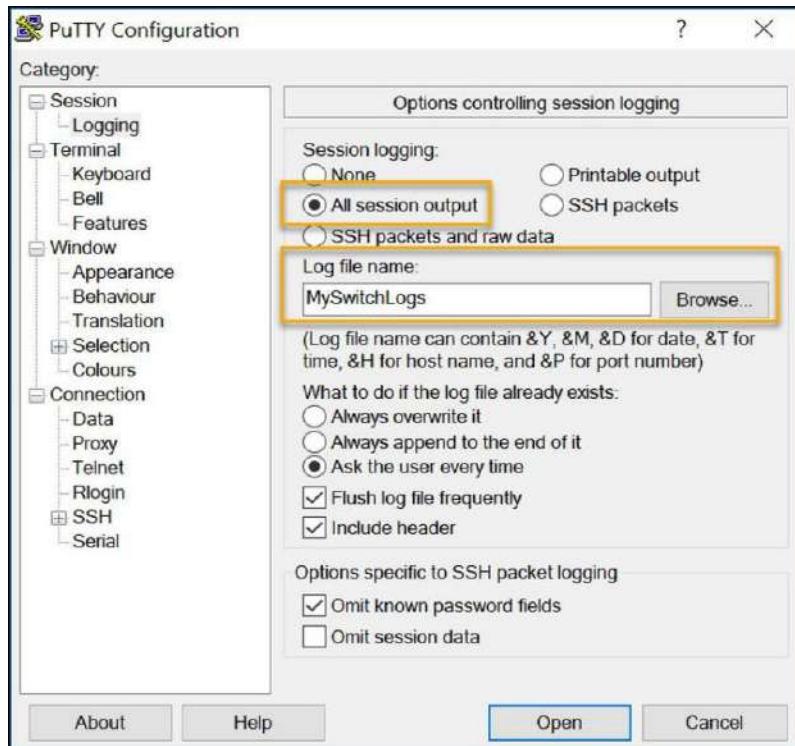
File konfigurasi juga dapat disimpan dan diarsipkan ke dokumen teks. Urutan langkah ini memastikan bahwa menyalin file konfigurasi untuk diedit atau digunakan kembali nanti.

Misalnya, anggap switch telah dikonfigurasi, dan konfigurasi yang berjalan telah disimpan di perangkat.

Langkah 1. Buka perangkat lunak emulasi terminal, seperti PuTTY atau Tera Term, yang sudah terhubung ke switch.



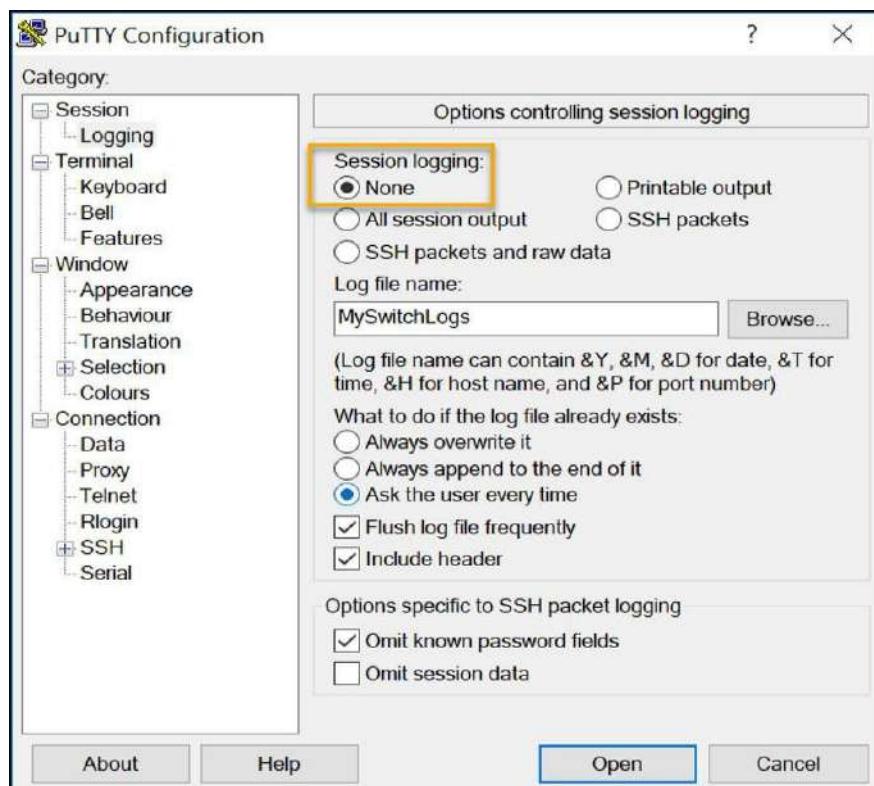
Langkah 2. Aktifkan logging di perangkat lunak terminal dan tetapkan nama dan lokasi file untuk menyimpan file log. Gambar tersebut menampilkan bahwa Semua output sesi akan disimpan ke file yang ditentukan (mis. MySwitch Logs).



Langkah 3. Jalankan perintah show running-config atau show startup-config. Teks yang ditampilkan di jendela terminal akan ditempatkan ke dalam file yang dipilih.

```
Sw-Floor-1# show running-config
Building configuration...
(output omitted)
```

Langkah 4. Nonaktifkan logging di Putty terminal. Gambar tersebut menunjukkan cara menonaktifkan logging dengan memilih opsi None sesi logging.



File teks yang dibuat dapat digunakan sebagai catatan tentang bagaimana perangkat saat ini diimplementasikan. File tersebut mungkin memerlukan pengeditan sebelum digunakan untuk memulihkan konfigurasi yang disimpan ke perangkat.

Untuk memulihkan file konfigurasi ke perangkat:

- Langkah 1. Masuk ke Global Configuration pada perangkat.
- Langkah 2. Salin dan tempel file teks ke jendela terminal yang terhubung ke switch

Teks di file tersebut akan diterapkan sebagai perintah di CLI dan menjadi konfigurasi yang berjalan di perangkat. Ini adalah metode yang nyaman untuk mengkonfigurasi perangkat secara manual.

Port dan Addresses

Selamat, Anda telah melakukan konfigurasi perangkat dasar! Tentu saja, kesenangan belum berakhir. Jika Anda ingin End Devices Anda berkomunikasi satu sama lain, Anda harus memastikan bahwa masing-masing perangkat memiliki alamat IP yang sesuai dan terhubung dengan benar. Anda akan mempelajari tentang alamat IP, port perangkat, dan media yang digunakan untuk menghubungkan perangkat dalam Materi ini.

Alamat IP

Penggunaan alamat IP adalah sarana utama yang memungkinkan perangkat untuk menemukan satu sama lain dan membangun komunikasi ujung ke ujung di internet. Setiap End Devices di jaringan harus dikonfigurasi dengan alamat IP. Contoh End Devices meliputi ini:

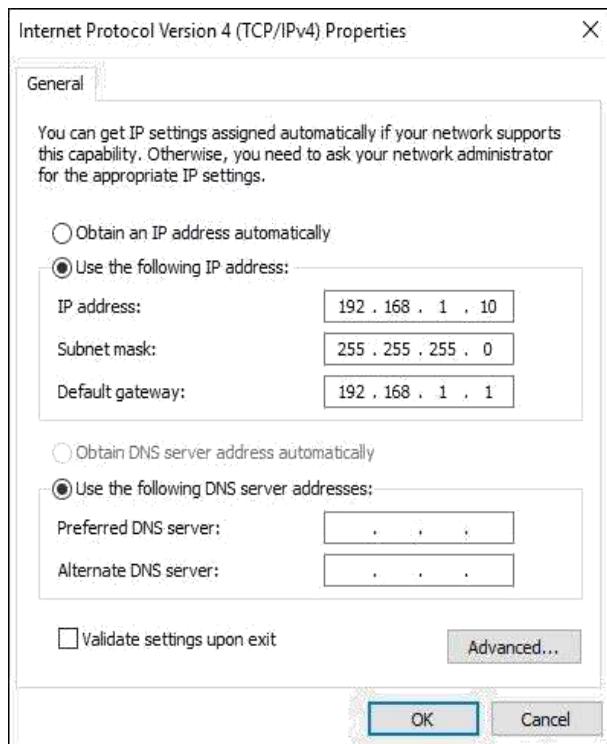
- Komputer (workstation, laptop, server file, server web)
- Printer jaringan
- Telepon VoIP
- CCTV
- Smartphone

Struktur alamat IPv4 disebut notasi desimal bertitik dan diwakili oleh empat angka desimal antara 0 dan 255. Alamat IPv4 ditetapkan ke perangkat individu yang terhubung ke jaringan.

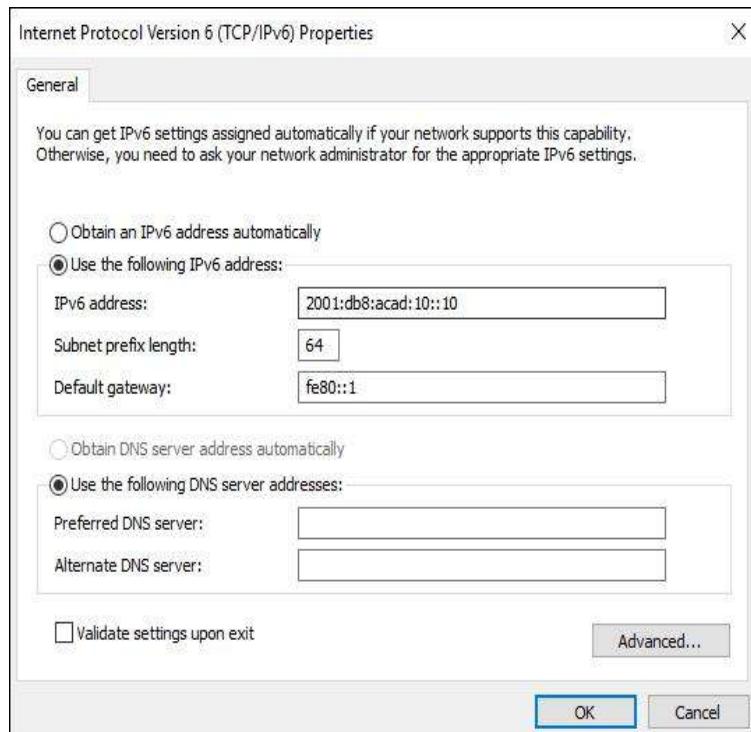
Catatan : IP dalam Materi ini mengacu pada protokol IPv4 dan IPv6. IPv6 adalah versi IP terbaru dan menggantikan IPv4 yang lebih umum.

Dengan alamat IPv4, subnet mask juga diperlukan. Subnet mask IPv4 adalah nilai 32-bit yang membedakan *network portion* alamat dari *host portion*. Ditambah dengan alamat IPv4, subnet mask menentukan pada subnet mana perangkat menjadi anggota.

Contoh pada gambar menampilkan alamat IPv4 (192.168.1.10), subnet mask (255.255.255.0), dan default gateway (192.168.1.1) yang ditetapkan ke host. Alamat Default gateway adalah **alamat IP router yang akan digunakan host untuk mengakses jaringan jarak jauh (remote), termasuk internet.**

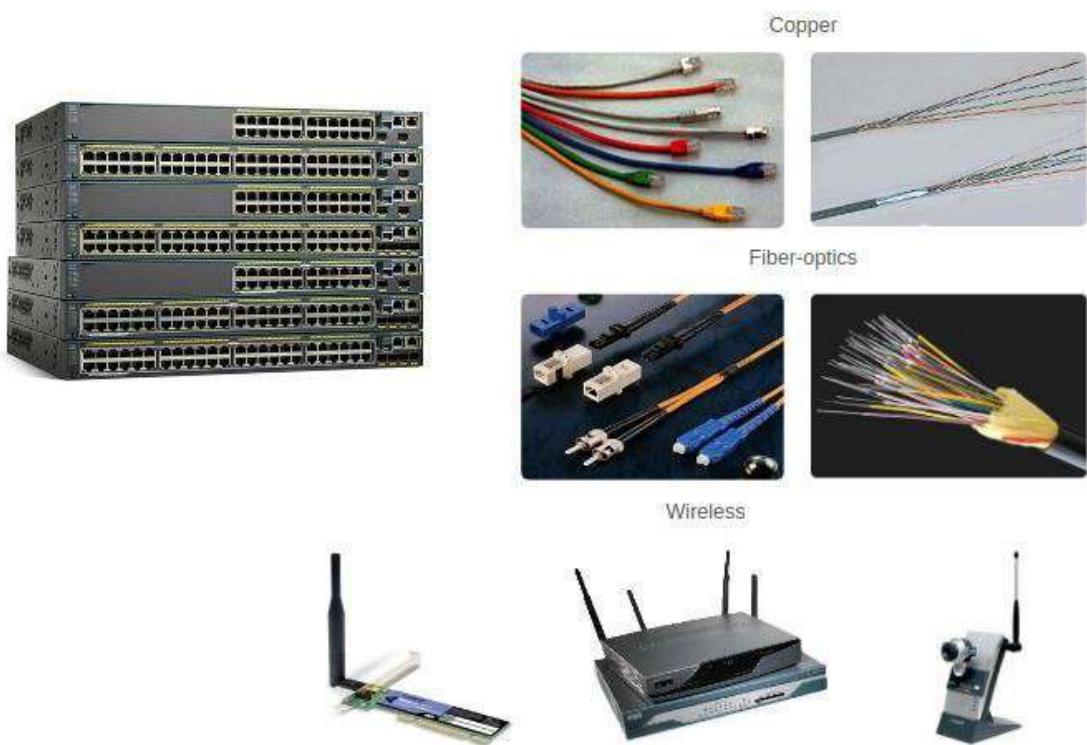


Alamat IPv6 berukuran 128 bit dan ditulis sebagai string nilai heksadesimal. Setiap empat bit diwakili oleh satu digit heksadesimal; untuk total 32 nilai heksadesimal. Grup yang terdiri dari empat digit heksadesimal dipisahkan oleh titik dua (:). Alamat IPv6 tidak peka huruf besar dan kecil dan dapat ditulis dalam huruf kecil atau huruf besar.



Interface dan Port

Komunikasi jaringan bergantung pada Interface perangkat End Devices, Interface perangkat jaringan, dan kabel yang menghubungkannya. Setiap Interface fisik memiliki spesifikasi, atau standar, yang menentukannya. Kabel yang menghubungkan ke Interface harus dirancang agar sesuai dengan standar fisik Interface. Jenis media jaringan antara lain kabel tembaga twisted pair, kabel fiber optic, kabel coaxial, atau wireless, seperti terlihat pada gambar.



Jenis media jaringan yang berbeda memiliki fitur dan manfaat yang berbeda pula. Tidak semua media jaringan memiliki karakteristik yang sama. Tidak semua media cocok untuk tujuan yang sama. Berikut adalah beberapa perbedaan antara berbagai jenis media:

- Jarak media berhasil membawa sinyal
- Lingkungan tempat media akan dipasang
- Jumlah data dan kecepatan pengirimannya
- Biaya media dan pemasangan

Tidak hanya setiap link di internet memerlukan jenis media jaringan tertentu, tetapi setiap link juga membutuhkan teknologi jaringan tertentu. Misalnya, Ethernet adalah teknologi jaringan area lokal (LAN) yang paling umum digunakan saat ini. Port ethernet terdapat di perangkat End devices, Switch, dan perangkat jaringan lain yang secara fisik dapat tersambung ke jaringan menggunakan kabel.

Switch Cisco IOS Layer 2 memiliki port fisik untuk menghubungkan perangkat. Port ini tidak mendukung alamat IP Layer 3. Oleh karena itu, Switch memiliki satu atau lebih Switch Virtual Interface (SVI). Ini adalah Virtual Interface karena tidak ada perangkat keras fisik pada perangkat yang terkait dengannya. SVI dibuat dalam perangkat lunak.

Interface Virtual memungkinkan Anda mengelola Switch dari jarak jauh melalui jaringan menggunakan IPv4 dan IPv6. Setiap switch dilengkapi dengan satu SVI yang muncul dalam konfigurasi default “out-of-the-box.” SVI default adalah Interface VLAN1.

Catatan : Switch Layer 2 tidak membutuhkan alamat IP. Alamat IP yang ditetapkan ke SVI digunakan untuk mengakses Switch dari jarak jauh. Alamat IP tidak diperlukan untuk Switch untuk menjalankan operasinya.

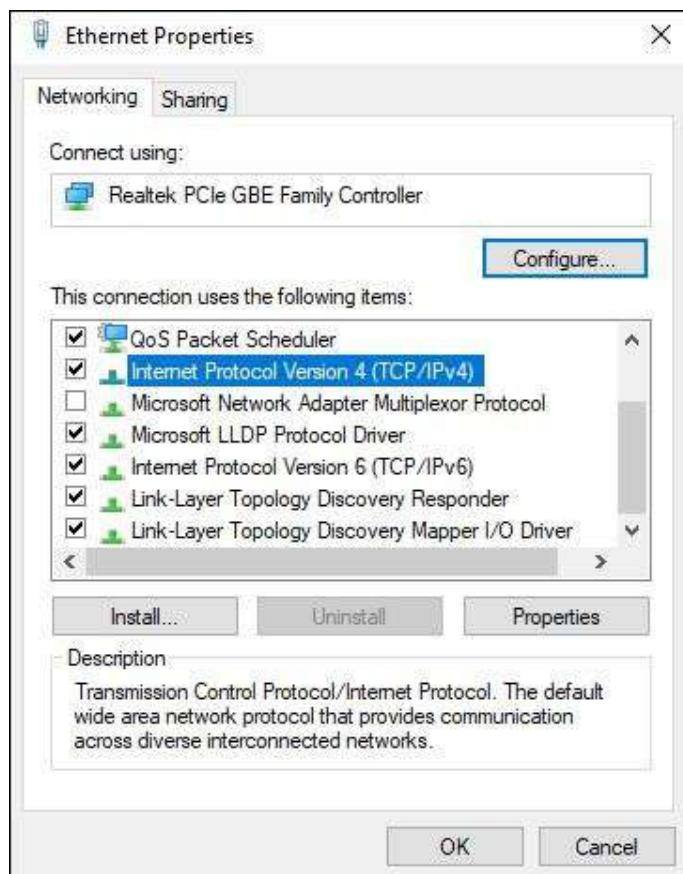
Konfigurasi IP Address

Sama seperti Anda memerlukan nomor telepon teman Anda untuk mengirim pesan atau menelepon mereka, End Device di jaringan Anda memerlukan alamat IP sehingga mereka dapat berkomunikasi dengan perangkat lain di jaringan Anda. Dalam Materi ini, Anda akan menerapkan koneksi dasar dengan mengkonfigurasi pengalaman IP pada switch dan PC.

Konfigurasi IP Static pada Windows

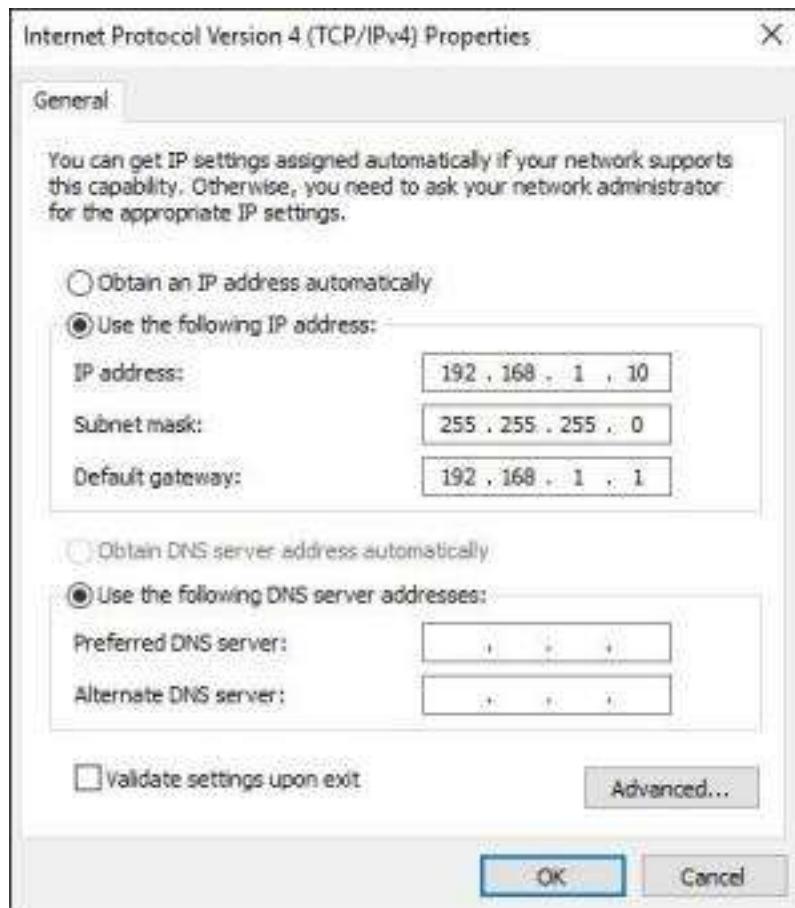
Informasi alamat IPv4 dapat dimasukkan ke end device secara manual (static) atau secara otomatis menggunakan Dynamic Host Configuration Protocol (DHCP).

Untuk mengkonfigurasi alamat IPv4 secara manual pada host Windows, buka **Control Panel > Network Sharing Center > Change Adapter Setting** dan pilih adaptor. Selanjutnya klik kanan dan pilih Properties untuk menampilkan Local Area Connection Properties , seperti yang ditunjukkan pada gambar.



Klik **Internet Protocol Version 4 (TCP / IPv4)** dan klik **Properties** untuk membuka jendela **Internet Protocol Version 4 (TCP / IPv4) Properties**, yang ditunjukkan pada gambar. Konfigurasikan alamat IPv4 dan informasi subnet mask, dan Default Gateway

Catatan : Opsi pengalaman dan konfigurasi IPv6 mirip dengan IPv4.



Catatan : Alamat server DNS adalah alamat IPv4 dan IPv6 dari Domain name server (DNS), yang digunakan untuk menerjemahkan alamat IP ke nama domain, seperti www.cisco.com

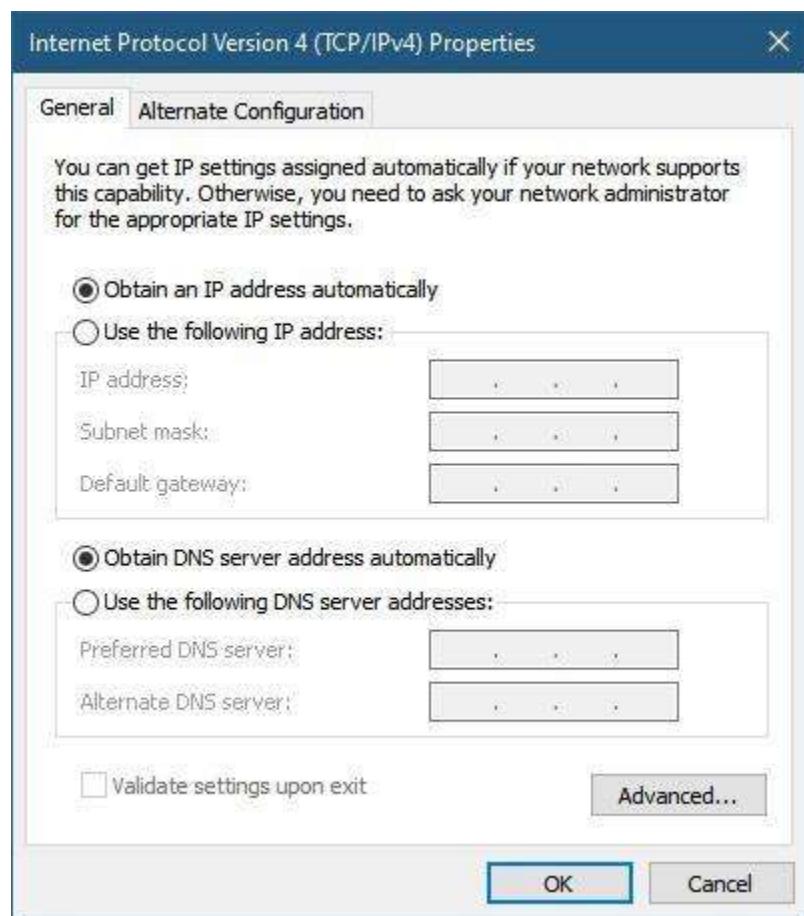
Konfigurasi Alamat IP Otomatis untuk End Device

End Device biasanya secara default menggunakan DHCP untuk konfigurasi alamat IPv4 otomatis. DHCP adalah teknologi yang digunakan di hampir setiap jaringan. Cara terbaik untuk memahami mengapa DHCP begitu populer adalah dengan mempertimbangkan semua pekerjaan ekstra yang harus dilakukan tanpanya.

Dalam jaringan, DHCP mengaktifkan konfigurasi alamat IPv4 otomatis untuk setiap End Device yang mengaktifkan DHCP. Bayangkan jumlah waktu yang diperlukan jika setiap kali Anda terhubung ke jaringan, Anda harus memasukkan alamat IPv4, subnet mask, Default gateway, dan server DNS secara manual. Lipat gandakan dengan setiap pengguna dan setiap perangkat dalam organisasi dan Anda melihat masalahnya. Konfigurasi manual juga meningkatkan kemungkinan kesalahan konfigurasi dengan menduplikasi alamat IPv4 perangkat lain.

Seperti yang ditunjukkan pada gambar, untuk mengkonfigurasi DHCP pada PC Windows, Anda hanya perlu memilih **Obtain an IP address automatically** and **Obtain DNS server address automatically**. PC Anda akan mencari server DHCP dan diberi pengaturan alamat yang diperlukan untuk berkomunikasi di jaringan.

Catatan : IPv6 menggunakan DHCPv6 dan SLAAC (Stateless Address Autoconfiguration) untuk alokasi alamat dinamis.



Konfigurasi Switch Virtual Interface

Untuk mengakses switch dari jarak jauh, alamat IP dan subnet mask harus dikonfigurasi pada SVI. Untuk mengkonfigurasi SVI pada switch, gunakan perintah **Interface vlan 1** pada Global Configuration . Vlan 1 bukanlah Interface fisik yang sebenarnya tetapi Virtual Interface. Selanjutnya tetapkan alamat IPv4 menggunakan perintah **ip address <ip-address> <subnet-mask>** . Terakhir, aktifkan Virtual Interface menggunakan perintah **no shutdown** .

Setelah perintah ini dikonfigurasi, Switch memiliki semua elemen IPv4 yang siap untuk komunikasi melalui jaringan.

Catatan : Mirip dengan Windows, Switch yang dikonfigurasi dengan alamat IPv4 biasanya juga harus memiliki gateway default yang ditetapkan. Ini dapat dilakukan dengan menggunakan perintah **ip default-gateway <ip-address>** .

```
Sw-Floor-1# configure terminal
Sw-Floor-1(config)# interface vlan 1
Sw-Floor-1(config-if)# ip address 192.168.1.20 255.255.255.0
Sw-Floor-1(config-if)# no shutdown
Sw-Floor-1(config-if)# exit
Sw-Floor-1(config)# ip default-gateway 192.168.1.1
```

Verifikasi Konektivitas

Pada Materi sebelumnya, Anda menerapkan konektivitas dasar dengan mengkonfigurasi pengalaman IP pada switch dan PC. Kemudian Anda memverifikasi konfigurasi dan konektivitas Anda, karena, apa gunanya mengkonfigurasi perangkat jika Anda tidak memverifikasi bahwa konfigurasi berfungsi? Anda akan melanjutkan proses ini dalam Materi ini. Dengan menggunakan CLI, Anda akan memverifikasi antarmuka dan switch dan router di jaringan Anda.

Dengan cara yang sama seperti Anda menggunakan perintah dan utilitas seperti **ipconfig** untuk memverifikasi konfigurasi jaringan host PC, Anda juga menggunakan perintah untuk memverifikasi interface dan pengaturan alamat intermediary device seperti switch dan router.

Check Konektivitas pada windows dan cisco

Windows adalah OS yang sering digunakan pada umumnya, oleh karena itu penulis ingin memberikan contoh verifikasi konektivitas di OS Windows, dengan topology berikut



Gunakan perintah **ipconfig** untuk melihat ip address pada adapter

```
Windows IP Configuration

Ethernet adapter Ethernet:

  Connection-specific DNS Suffix  . :
  Link-local IPv6 Address . . . . . : fe80::821:b23f:813d:89eb%13
  IPv4 Address. . . . . : 10.255.254.207
  Subnet Mask . . . . . : 255.255.255.0
  Default Gateway . . . . . : 10.255.254.1

Ethernet adapter VirtualBox Host-Only Network:

  Connection-specific DNS Suffix  . :
  Link-local IPv6 Address . . . . . : fe80::5dd:3064:7bf5:430f%17
  IPv4 Address. . . . . : 192.168.1.2
  Subnet Mask . . . . . : 255.255.255.0
  Default Gateway . . . . . :
```

Gunakan perintah **show ip interface brief** pada router cisco untuk melihat ip address

```
Router(config-if)#do show ip int bri
Interface          IP-Address      OK? Method Status          Protocol
GigabitEthernet0/0/0 10.255.254.1   YES manual up           down
GigabitEthernet0/0/1 unassigned     YES unset administratively down down
GigabitEthernet0/0/2 unassigned     YES unset administratively down down
Vlan1              unassigned     YES unset administratively down down
```

Lalu uji koneksi antara router dan pc

PC

```
C:\Users\SkyderAmzLee>ping 10.255.254.1

Pinging 10.255.254.1 with 32 bytes of data:
Reply from 10.255.254.1: bytes=32 time<1ms TTL=64

Ping statistics for 10.255.254.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Router

```
Router#ping 10.255.254.207

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.255.254.207, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 0/0/0 ms
Router#
```

BAB 3

~ *Protocols dan Model* ~

Judul Bab : Protocols and Models

Tujuan Bab : Menjelaskan bagaimana protokol jaringan memungkinkan perangkat untuk mengakses resource jaringan lokal dan jarak jauh.

Link Test Pemahaman : <https://s.id/QxgM>

| Judul Materi | Tujuan Materi |
|---------------------|---|
| The Rules | Menjelaskan jenis aturan yang diperlukan untuk berkomunikasi |
| Protokol | Menjelaskan kenapa protokol diperlukan dalam komunikasi jaringan |
| Suite Protokol | Menjelaskan mengapa kita harus mengikuti rangkaian protokol |
| Organisasi Standard | Menjelaskan peran organisasi standard dalam menetapkan protokol untuk interoperabilitas jaringan |
| Model Referensi | Menjelaskan bagaimana model TCP / IP dan model OSI digunakan untuk memfasilitasi standardisasi dalam proses komunikasi. |
| Enkapsulasi Data | Menjelaskan bagaimana enkapsulasi data memungkinkan data diangkut ke seluruh jaringan. |
| Data Access | Menjelaskan bagaimana host lokal mengakses resource lokal di jaringan. |

THE RULES

Jaringan bervariasi dalam ukuran, bentuk, dan fungsi. Mereka bisa menjadi serumit perangkat yang terhubung melalui internet, atau sesederhana dua komputer yang terhubung langsung satu sama lain dengan satu kabel, dan apa pun di antaranya. Namun, hanya memiliki koneksi fisik berkabel atau nirkabel antara end device tidak cukup untuk mengaktifkan komunikasi. Agar komunikasi terjadi, perangkat harus tahu “bagaimana” Jaringan berkomunikasi.

Dasar-dasar Komunikasi

Orang bertukar pikiran menggunakan banyak metode komunikasi yang berbeda. Namun, semua metode komunikasi memiliki tiga elemen yang sama:

- **Sumber pesan (Sender)** – Sumber pesan adalah orang, atau perangkat elektronik, yang perlu mengirim pesan ke individu atau perangkat lain.
- **Message Destination (Receiver)** – Tujuan menerima pesan dan menafsirkannya.
- **Channel** – jalur perjalanan pesan dari sumber ke tujuan.

Protokol Komunikasi

Mengirim pesan, baik dengan komunikasi tatap muka atau melalui jaringan, diatur oleh aturan yang disebut protokol. Protokol ini khusus untuk jenis metode komunikasi yang digunakan. Dalam komunikasi pribadi kita sehari-hari, aturan yang kita gunakan untuk berkomunikasi melalui satu media, seperti panggilan telepon, belum tentu sama dengan aturan penggunaan media lain, seperti mengirim surat.

Proses pengiriman surat mirip dengan komunikasi yang terjadi di jaringan komputer.

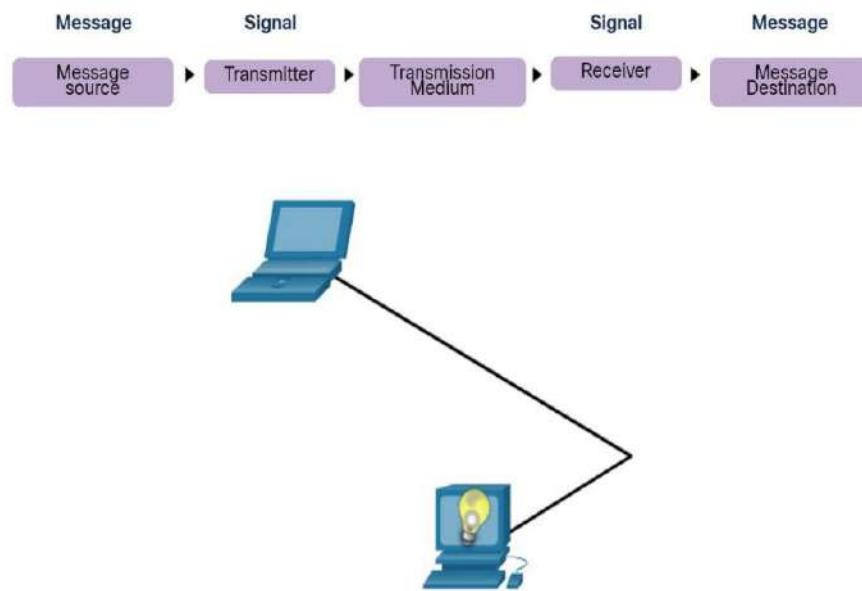
Contoh Analogi

Sebelum berkomunikasi, mereka harus menyepakati cara berkomunikasi. Jika komunikasi menggunakan suara, mereka harus menyetujui bahasanya terlebih dahulu. Selanjutnya, ketika mereka memiliki pesan untuk dibagikan, mereka harus dapat memformat pesan itu dengan cara yang dapat dimengerti.

Jika seseorang menggunakan bahasa Inggris, tetapi struktur kalimatnya buruk, pesannya dapat dengan mudah disalahpahami. Masing-masing tugas ini menjelaskan protokol yang digunakan untuk menyelesaikan komunikasi.

Contoh Jaringan

Sebelum berkomunikasi, perangkat harus setuju tentang cara berkomunikasi. Mereka juga harus memformat pesan dengan cara yang dapat dimengerti.



Pembentukan Aturan

Sebelum berkomunikasi satu sama lain, individu harus menggunakan aturan atau kesepakatan yang ditetapkan untuk mengatur percakapan. Pertimbangkan pesan ini misalnya:

humans communication between govern rules. It is very difficult to understand messages that are not correctly formatted and do not follow the established rules and protocols. A structure of grammar, language, punctuation and sentence make a configuration humanly comprehensible for many individuals.

Perhatikan betapa sulitnya membaca pesan karena tidak diformat dengan benar. Ini harus ditulis menggunakan aturan (yaitu, **protokol**) yang diperlukan untuk komunikasi yang efektif. Contoh ini menunjukkan pesan yang sekarang diformat dengan benar untuk bahasa dan tata bahasa.

Rules govern communication between humans. It is very difficult to understand messages that are not correctly formatted and do not follow the established rules and protocols. The structure of the grammar, the language, the punctuation and the sentence make the configuration humanly understandable for many different individuals.

Protokol harus memperhitungkan persyaratan berikut untuk berhasil menyampaikan pesan yang dipahami oleh penerima:

- Pengirim dan penerima yang teridentifikasi
- Bahasa dan tata bahasa umum
- Kecepatan dan waktu pengiriman
- Persyaratan konfirmasi atau Acknowledgment

Persyaratan Protokol Jaringan

Protokol yang digunakan dalam komunikasi jaringan memiliki banyak ciri mendasar ini. Selain mengidentifikasi sumber dan tujuan, protokol komputer dan jaringan menentukan detail bagaimana pesan dikirimkan melalui jaringan. Protokol komputer yang umum mencakup persyaratan berikut:

- Message encoding
- Message formatting and encapsulation
- Message size
- Message timing
- Message delivery options

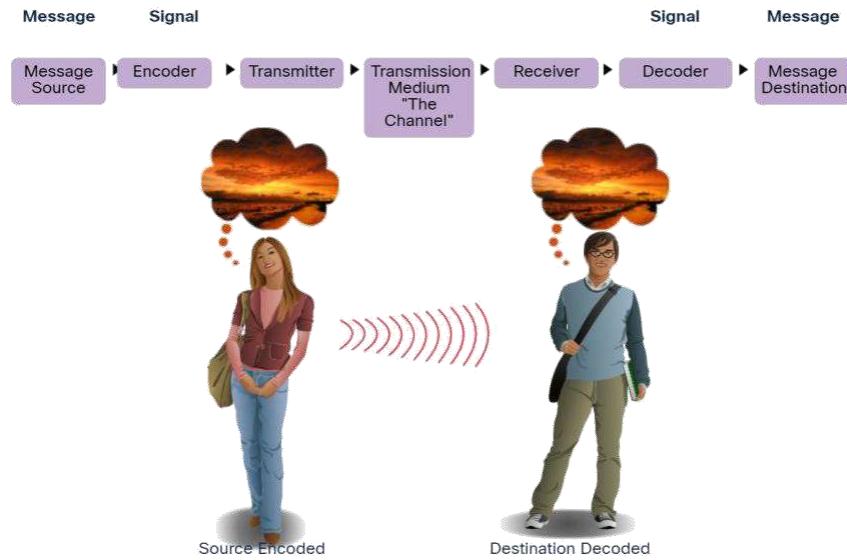
Message Encoding

Salah satu langkah pertama untuk mengirim pesan adalah pengkodean. Pengkodean atau **encoding** adalah proses mengubah informasi menjadi bentuk lain yang dapat diterima, untuk transmisi. Penguraian kode atau **decoding** adalah membalikkan proses ini untuk menafsirkan informasi.

Analogi

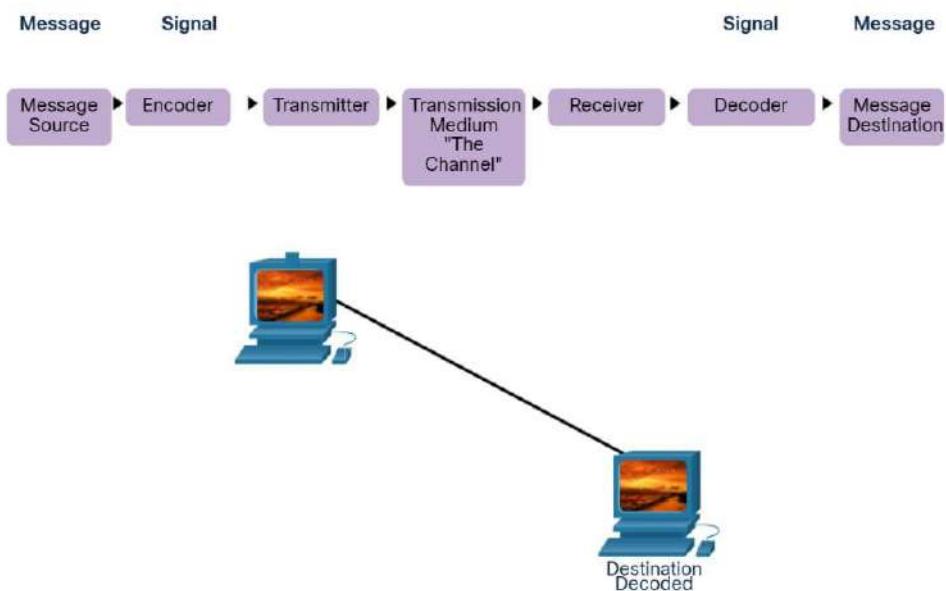
Bayangkan seseorang menelpon temannya untuk membahas detail matahari terbenam yang indah.

Untuk mengkomunikasikan pesannya, dia mengubah pikirannya menjadi bahasa yang disepakati. Dia kemudian mengucapkan kata-kata tersebut dengan menggunakan suara dan infleksi dari bahasa lisan yang menyampaikan pesan tersebut. Temannya mendengarkan deskripsi dan menerjemahkan suara untuk memahami pesan yang diterimanya



Contoh Jaringan

Encoding antara host harus dalam format yang sesuai untuk media. Pesan yang dikirim di seluruh jaringan pertama kali diubah menjadi bit oleh host pengirim. Setiap bit dikodekan ke dalam pola tegangan pada kabel tembaga, cahaya inframerah dalam serat optik, atau gelombang mikro untuk sistem nirkabel. Host tujuan menerima dan memecahkan kode sinyal untuk menafsirkan pesan.



Message formatting and encapsulation

Ketika pesan dikirim dari sumber ke tujuan, itu harus menggunakan format atau struktur tertentu. Format pesan tergantung pada jenis pesan dan saluran yang digunakan untuk menyampaikan pesan tersebut.

Contoh Analogi

Contoh umum yang membutuhkan format yang benar dalam komunikasi manusia adalah saat mengirim surat.

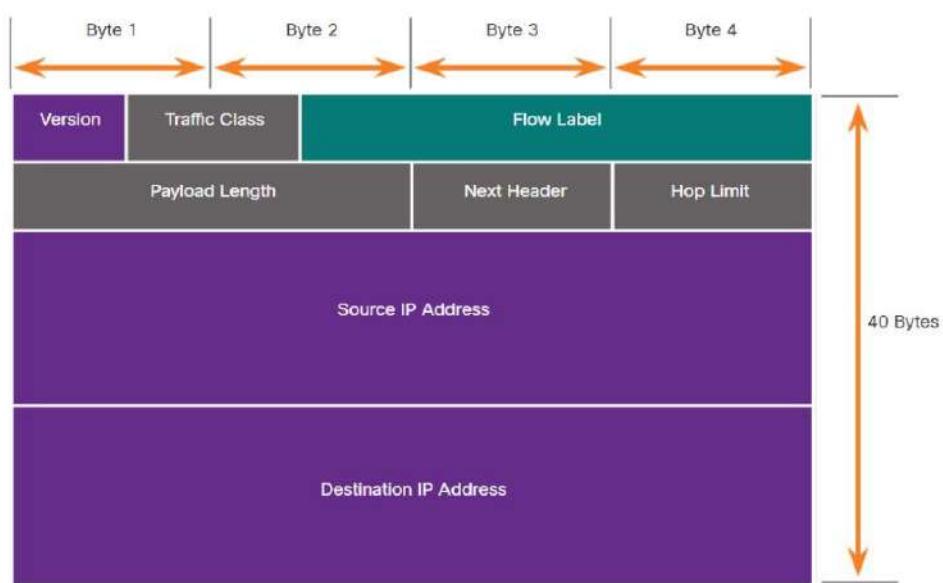
Sebuah amplop memiliki alamat pengirim dan penerima, masing-masing terletak di tempat yang tepat pada amplop. Jika alamat tujuan dan format tidak benar, surat tidak terkirim.

Contoh Jaringan

Mirip dengan mengirim surat, pesan yang dikirim melalui jaringan komputer mengikuti aturan format khusus untuk dikirim dan diproses.

Internet Protocol (IP) adalah protokol dengan fungsi yang mirip dengan contoh amplop. Dalam gambar, *field* paket Internet Protocol versi 6 (IPv6) mengidentifikasi sumber paket dan tujuannya. IP bertanggung jawab untuk mengirim pesan dari sumber pesan ke tujuan melalui satu atau lebih jaringan.

Catatan:*Field* paket IPv6 dibahas secara rinci dalam materi lain.



Message Size

Aturan komunikasi lainnya adalah Message Size. Message Size penting karena jika ukuran pesan yang diterima terlalu besar perangkat pun tidak dapat menerimanya

Contoh Analogi

Saat orang berkomunikasi satu sama lain, pesan yang mereka kirim biasanya dipecah menjadi beberapa bagian atau kalimat yang lebih kecil. Kalimat-kalimat ini dibatasi ukurannya pada apa yang dapat diproses oleh penerima pada satu waktu.

Contoh Jaringan

Ketika ukuran *Long message* dikirim dari satu host ke host lain melalui jaringan, perlu untuk memecah pesan menjadi potongan-potongan yang lebih kecil. Aturan yang mengatur ukuran potongan, atau frame, yang dikomunikasikan di seluruh jaringan sangat ketat. Mereka juga bisa berbeda, tergantung pada saluran yang digunakan. Frame yang terlalu panjang atau terlalu pendek tidak dikirim.

Pembatasan ukuran frame memerlukan host sumber untuk memecah *Long message* menjadi potongan-potongan individu yang memenuhi persyaratan ukuran minimum dan maksimum. *Long message* akan dikirim dalam bingkai terpisah, dengan setiap frame berisi sepotong pesan asli. Setiap frame juga akan memiliki informasi pengalamatan sendiri. Pada host penerima, potongan-potongan individu dari pesan direkonstruksi ke dalam pesan asli.

Waktu Pesan

Waktu pesan juga sangat penting dalam komunikasi jaringan. Pengaturan waktu pesan meliputi:

- **Flow Control** – Ini adalah proses mengelola kecepatan transmisi data. Kontrol aliran menentukan seberapa banyak informasi dapat dikirim dan kecepatan pengirimannya. Misalnya, jika satu orang berbicara terlalu cepat, penerima akan sulit mendengar dan memahami pesannya. Dalam komunikasi jaringan, terdapat protokol jaringan yang digunakan oleh perangkat sumber dan tujuan untuk bernegosiasi dan mengatur arus informasi.
- **Response Timeout** – Jika seseorang mengajukan pertanyaan dan tidak mendengar jawaban dalam jangka waktu yang dapat diterima, orang tersebut berasumsi bahwa

tidak ada jawaban yang datang dan bereaksi sesuai dengannya. Orang tersebut mungkin mengulangi pertanyaannya atau sebaliknya, melanjutkan percakapan. Host di jaringan menggunakan protokol jaringan yang menentukan berapa lama menunggu tanggapan dan tindakan apa yang harus diambil jika waktu habis tanggapan terjadi.

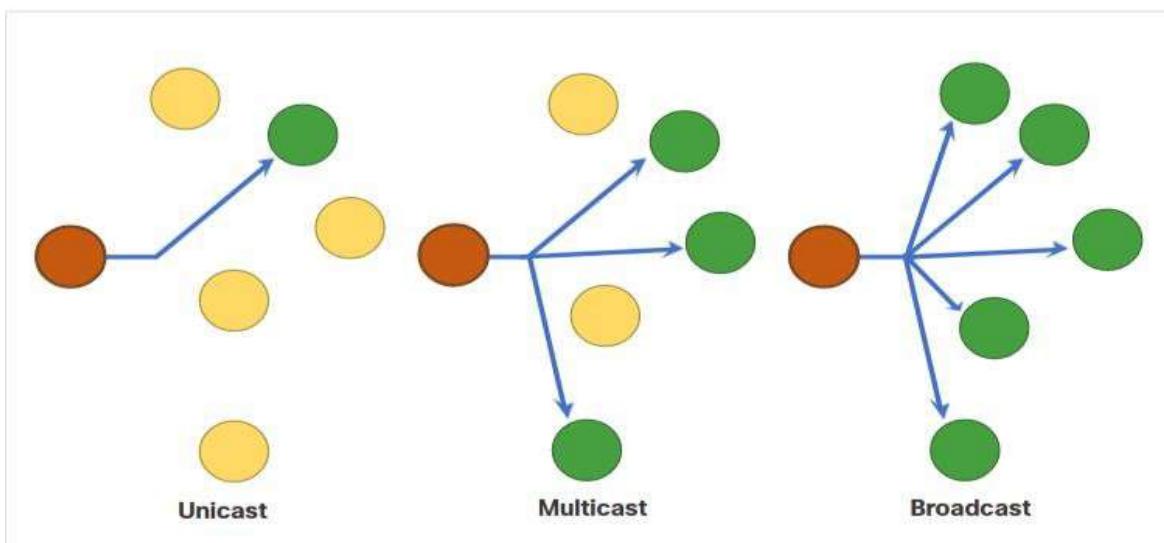
- **Access Method** – Ini menentukan kapan seseorang dapat mengirim pesan. Contohnya dua orang yang berbicara pada saat yang sama, lalu “benturan informasi” terjadi, dan keduanya perlu mundur dan memulai lagi. Demikian juga, saat perangkat ingin mengirim pada LAN nirkabel, **Network Interface Card (NIC)** WLAN perlu menentukan apakah media nirkabel tersedia.

Opsi Pengiriman Pesan

Sebuah pesan dapat disampaikan dengan berbagai cara. Terkadang, seseorang ingin mengkomunikasikan informasi kepada satu individu. Di lain waktu, orang tersebut mungkin perlu mengirimkan informasi ke sekelompok orang pada waktu yang sama, atau bahkan ke semua orang di area yang sama.

Komunikasi jaringan memiliki opsi pengiriman serupa untuk berkomunikasi. ada tiga jenis komunikasi data antara lain:

- **Unicast** – Informasi sedang dikirim ke satu perangkat ujung.
- **Multicast** – Informasi sedang dikirim ke satu atau lebih **End Device**.
- **Broadcast** – Informasi sedang dikirim ke semua **End Device**.



PROTOCOL

Anda tahu bahwa agar perangkat jaringan dapat berkomunikasi melalui jaringan, setiap perangkat harus mematuhi seperangkat aturan yang sama. Aturan-aturan ini disebut protokol dan memiliki banyak fungsi dalam jaringan. Materi ini memberi Anda gambaran umum tentang protokol jaringan.

Ringkasan Protokol Jaringan

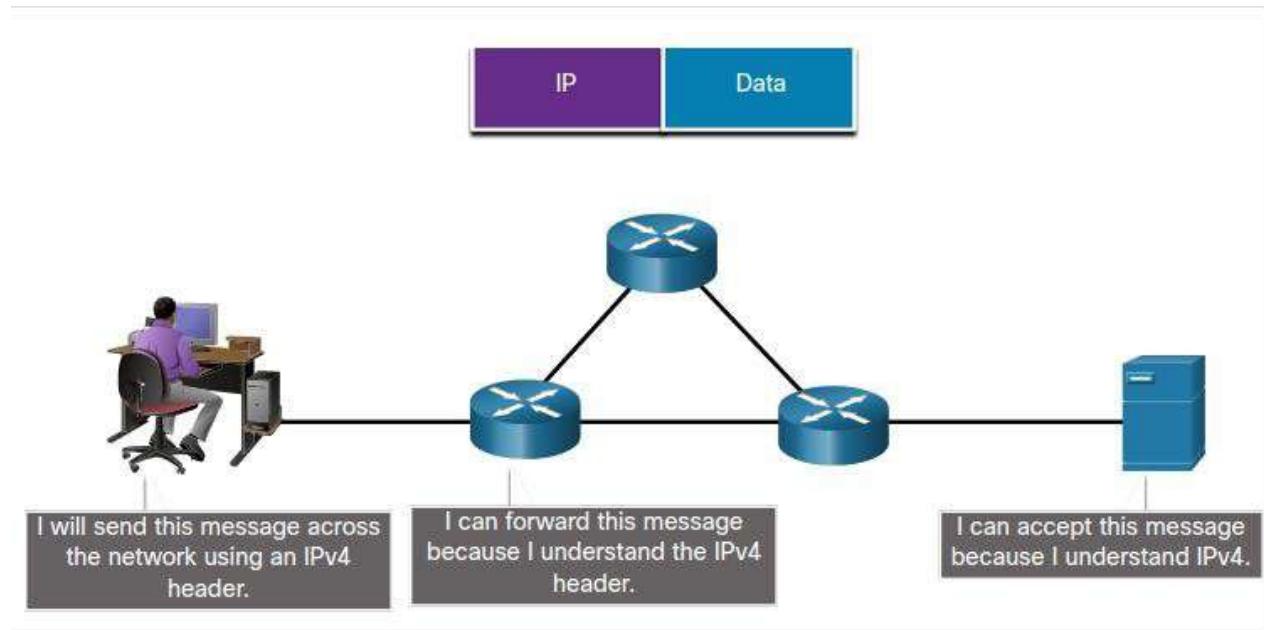
Protokol jaringan menentukan format umum dan seperangkat aturan untuk bertukar pesan antar perangkat. Protokol diimplementasikan oleh End Devices dan **Intermediary Device** dalam perangkat lunak, perangkat keras, atau keduanya. Setiap protokol jaringan memiliki fungsi, format, dan aturan komunikasi masing-masing.

Tabel berikut menjelaskan berbagai jenis protokol

| Jenis Protokol | Deskripsi |
|---------------------------------|--|
| Network Communication Protocols | Protokol memungkinkan dua atau lebih perangkat untuk berkomunikasi melalui satu atau lebih jaringan. Teknologi Ethernet melibatkan berbagai protokol seperti IP, Transmission Control Protocol (TCP), HyperText Transfer Protocol (HTTP), dan banyak lagi. |
| Network Security Protocols | Protokol mengamankan data untuk menyediakan otentikasi, integritas data, dan enkripsi data. Contoh protokol aman termasuk Secure Shell (SSH), Secure Sockets Layer (SSL), dan Transport Layer Security (TLS). |
| Routing Protocols | Protokol memungkinkan router untuk bertukar informasi router, membandingkan informasi jalur, dan kemudian memilih jalur terbaik ke jaringan tujuan. Contoh Routing Protocols termasuk Open Shortest Path First (OSPF) dan Border Gateway Protocol (BGP). |
| Service Discovery Protocols | Protokol digunakan untuk deteksi otomatis perangkat atau layanan. Contoh Service Discovery Protocols termasuk Dynamic Host Configuration Protocol (DHCP) yang menemukan layanan untuk alokasi alamat IP, dan Domain Name System (DNS) yang digunakan untuk melakukan terjemahan alamat nama-ke-IP. |

Fungsi Protokol Jaringan

Protokol jaringan bertanggung jawab atas berbagai fungsi yang diperlukan untuk komunikasi jaringan antara End Device. Misalnya, pada gambar, bagaimana komputer mengirim pesan, melalui beberapa perangkat jaringan, ke server?



Gambar tersebut menunjukkan bagaimana protokol **IPv4** dapat digunakan untuk mengirim pesan dari komputer melalui jaringan ke server. Di tengah gambar ada tiga router yang dihubungkan bersama dalam segitiga. Router di sebelah kiri terhubung ke komputer. Router di sebelah kanan terhubung ke server.

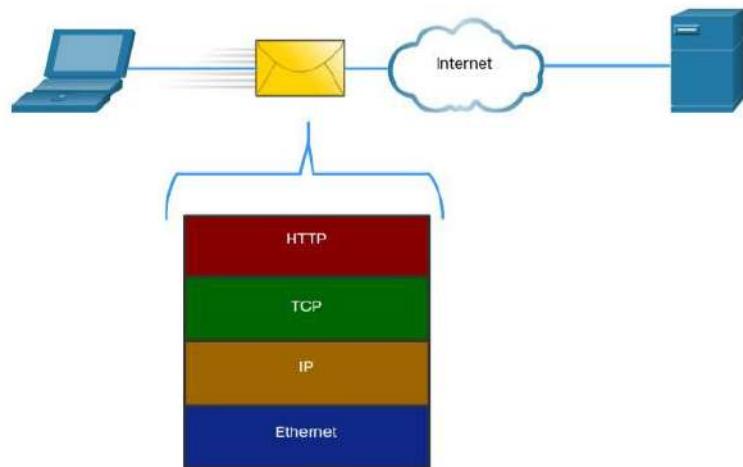
Komputer dan perangkat jaringan menggunakan protokol yang disepakati untuk berkomunikasi. Tabel mencantumkan fungsi dari protokol ini.

| Fungsi | Deskripsi |
|-------------|---|
| Addressing | Ini mengidentifikasi pengirim dan penerima pesan yang dituju menggunakan skema pengalamanan yang ditentukan. Contoh protokol yang menyediakan pengalamanan termasuk Ethernet, IPv4, dan IPv6. |
| Reliability | Fungsi ini memberikan jaminan mekanisme pengiriman jika pesan hilang atau rusak saat transit. TCP menyediakan pengiriman yang terjamin. |

| | |
|-----------------------|---|
| Flow Control | Fungsi ini memastikan bahwa data mengalir dengan kecepatan yang efisien antara dua perangkat yang berkomunikasi. TCP menyediakan layanan kontrol aliran. |
| Sequencing | Fungsi ini secara unik memberi label pada setiap segmen data yang ditransmisikan. Perangkat penerima menggunakan informasi urutan untuk memasang kembali informasi dengan benar. Ini berguna jika segmen data hilang, tertunda atau diterima rusak. TCP menyediakan layanan pengurutan. |
| Error Detection | Fungsi ini digunakan untuk menentukan apakah data menjadi rusak selama transmisi. Berbagai protokol yang menyediakan deteksi kesalahan termasuk Ethernet, IPv4, IPv6, dan TCP. |
| Application Interface | Fungsi ini berisi informasi yang digunakan untuk proses-proses komunikasi antara aplikasi jaringan. Misalnya, saat mengakses halaman web, protokol HTTP atau HTTPS digunakan untuk berkomunikasi antara proses web klien dan server. |

Interaksi Protokol

Pesan yang dikirim melalui jaringan komputer biasanya memerlukan penggunaan beberapa protokol, masing-masing dengan fungsi dan formatnya sendiri. Gambar tersebut menunjukkan beberapa protokol jaringan umum yang digunakan saat perangkat mengirimkan permintaan ke server web untuk halaman webnya.



Protokol pada gambar dijelaskan sebagai berikut:

- **Hypertext Transfer Protocol (HTTP)** – Protokol ini mengatur cara server web dan klien web berinteraksi. HTTP mendefinisikan konten dan pemformatan permintaan dan tanggapan yang dipertukarkan antara klien dan server. Baik klien dan perangkat lunak server web menerapkan HTTP sebagai bagian dari aplikasi. HTTP bergantung pada protokol lain untuk mengatur bagaimana pesan diangkut antara klien dan server.
- **Transmission Control Protocol (TCP)** – Protokol ini mengatur percakapan individu. TCP bertanggung jawab untuk menjamin pengiriman informasi yang andal dan mengelola kontrol aliran antara End Device.
- **Internet Protocol (IP)** – Protokol ini bertanggung jawab untuk mengirimkan pesan dari pengirim ke penerima. IP digunakan oleh router untuk meneruskan pesan ke beberapa jaringan.
- **Ethernet** – Protokol ini bertanggung jawab atas pengiriman pesan dari satu NIC ke NIC lain di jaringan area lokal (LAN) Ethernet yang sama.

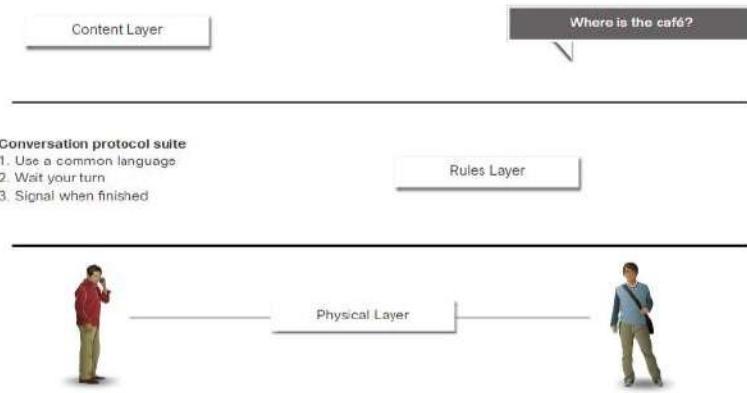
Rangkaian Protocol

Dalam banyak kasus, protokol harus dapat bekerja dengan protokol lain sehingga pengalaman online Anda memberikan semua yang Anda butuhkan untuk komunikasi jaringan. Rangkaian protokol dirancang untuk bekerja dengan mulus satu sama lain.

Rangkaian Protokol Dalam Jaringan

Protokol suite atau **rangkaian protocol** adalah sekelompok protokol yang saling terkait yang diperlukan untuk melakukan fungsi komunikasi.

Salah satu cara terbaik untuk memvisualisasikan bagaimana rangkaian protocol berinteraksi adalah dengan melihat interaksi sebagai tumpukan. Tumpukan atau stack protokol menunjukkan bagaimana masing-masing protokol dalam sebuah rangkaian diimplementasikan. Protokol dilihat dari segi lapisan atau layer, dengan setiap layanan tingkat yang lebih tinggi bergantung pada fungsionalitas yang ditentukan oleh protokol yang ditampilkan di tingkat yang lebih rendah. **Bottom layer** berkaitan dengan pemindahan data melalui jaringan dan menyediakan **Upper Layer**, yang difokuskan pada konten pesan yang dikirim.



Seperti yang diilustrasikan pada gambar, kita dapat menggunakan layer untuk menggambarkan aktivitas yang terjadi dalam komunikasi tatap muka. Di bagian bawah adalah **Physical Layer** tempat kami memiliki dua orang dengan suara yang mengucapkan kata-kata dengan lantang. Di tengah adalah **Rules Layer** yang mengatur persyaratan komunikasi termasuk bahasa yang umum harus dipilih. Di bagian atas adalah **Content Layer** dan disinilah konten komunikasi sebenarnya diucapkan.

Evolusi Rangkaian Protokol

Rangkaian protocol atau suite protocol adalah seperangkat protokol yang bekerja sama untuk menyediakan layanan komunikasi jaringan yang komprehensif. Sejak tahun 1970-an telah ada beberapa rangkaian protokol yang berbeda, beberapa dikembangkan oleh organisasi standar dan lainnya dikembangkan oleh berbagai vendor.

Selama evolusi komunikasi jaringan dan internet ada beberapa suite protokol yang bersaing, seperti yang ditunjukkan pada gambar.

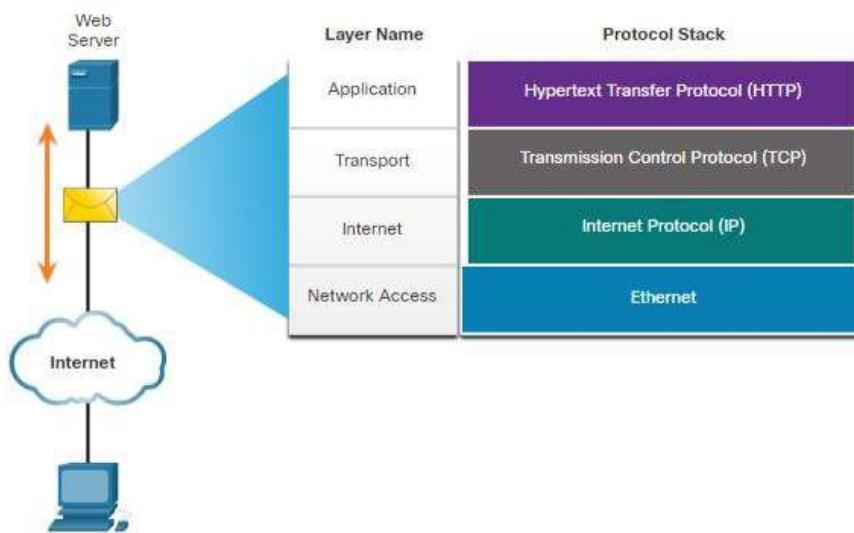
| TCP/IP Layer Name | TCP/IP | ISO | AppleTalk | Novell Netware |
|-------------------|-------------------------------|------------------------------|---------------------|----------------|
| Application | HTTP DNS DHCP FTP | ACSE ROSE TRSE SESE | AFP | NDS |
| Transport | TCP UDP | TP0 TP1 TP2 TP3 TP4 | ATP AEP NBP RTMP | SPX |
| Internet | IPv4 IPv6 ICMPv4 ICMPv6 | CONP/CMNS CLNP/CLNS | AARP | IPX |
| Network Access | | Ethernet | ARP WLAN | |

- **Internet Protocol Suite atau TCP / IP** – Ini adalah rangkaian protokol paling umum dan relevan yang digunakan saat ini. Rangkaian protokol TCP / IP adalah rangkaian protokol open-standard yang dikelola oleh Internet Engineering Task Force (IETF).
- **Protokol Open System Interconnection (OSI)** – Ini adalah keluarga protokol yang dikembangkan bersama pada tahun 1977 oleh Organisasi Internasional untuk Standardisasi (ISO) dan International Telecommunications Union (ITU). Protokol OSI juga menyertakan model seven layer yang disebut model referensi OSI. Model referensi OSI mengkategorikan fungsi protokolnya. Saat ini OSI terutama dikenal dengan model berlapisnya. Sebagian besar protokol OSI telah digantikan oleh TCP / IP.

- **AppleTalk** – Rangkaian protokol berpemilik berumur pendek yang dirilis oleh Apple Inc. pada tahun 1985 untuk perangkat Apple. Pada tahun 1995, Apple mengadopsi TCP / IP untuk menggantikan AppleTalk.
- **Novell NetWare** – Rangkaian protokol dan sistem operasi jaringan berpemilik berumur pendek yang dikembangkan oleh Novell Inc. pada tahun 1983 menggunakan protokol jaringan IPX. Pada tahun 1995, Novell mengadopsi TCP / IP untuk menggantikan IPX.

Contoh Protokol TCP / IP

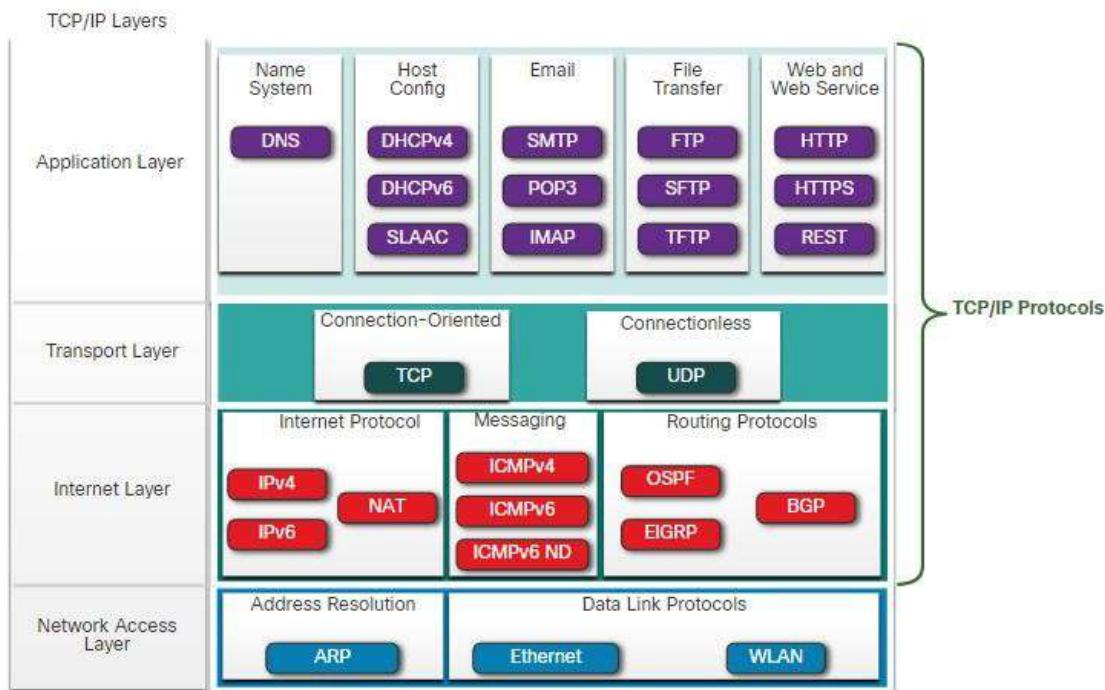
Protokol TCP / IP tersedia untuk application, transport, dan Internet Protocol. Tidak ada protokol TCP / IP di Network Access Layer. Protokol Network Access Layer yang paling umum adalah protokol Ethernet dan WLAN (LAN nirkabel). Protokol Network Access Layer bertanggung jawab untuk mengirimkan paket IP melalui media fisik.



Gambar tersebut menunjukkan contoh tiga protokol TCP / IP yang digunakan untuk mengirim paket antara browser web host dan server web. **HTTP**, **TCP**, dan **IP** adalah protokol TCP / IP yang digunakan. Pada **Network Access Layer**, Ethernet digunakan dalam contoh. Namun, ini juga bisa menjadi standar nirkabel seperti WLAN atau layanan seluler.

Paket Paket Protokol TCP / IP

Saat ini, rangkaian protokol TCP / IP mencakup banyak protokol dan terus berkembang untuk mendukung layanan baru. Beberapa yang lebih populer ditunjukkan pada gambar.



TCP / IP adalah rangkaian protokol yang digunakan oleh internet dan jaringan saat ini. TCP / IP memiliki dua aspek penting bagi vendor dan produsen:

- **Open standard protocol suite** – Ini berarti ini tersedia secara bebas untuk umum dan dapat digunakan oleh vendor mana pun pada perangkat keras atau perangkat lunak mereka.
- **Paket protokol berbasis standar** – Ini berarti telah didukung oleh industri jaringan dan disetujui oleh organisasi standar. Ini memastikan bahwa produk dari produsen yang berbeda dapat beroperasi dengan sukses.

Application Layer

Name System

- **DNS** – Sistem Nama Domain. Menerjemahkan nama domain seperti cisco.com, menjadi alamat IP.

Konfigurasi Host

- **DCHPv4** – Protokol Konfigurasi Host Dinamis untuk IPv4. Server DCHPv4 secara dinamis memberikan informasi pengalamanan IPv4 ke klien DCHPv4 saat start-up dan memungkinkan alamat untuk digunakan kembali saat tidak lagi diperlukan.
- **DCHPv6** – Protokol Konfigurasi Host Dinamis untuk IPv6. DCHPv6 mirip dengan DCHPv4. Server DCHPv6 secara dinamis memberikan informasi pengalamanan IPv6 ke klien DCHPv6 saat start-up.
- **SLAAC** – Konfigurasi Otomatis Alamat Tanpa **Status**. Metode yang memungkinkan perangkat mendapatkan informasi pengalamanan IPv6 tanpa menggunakan server DCHPv6.

Mail

- **SMTP** – Protokol Transfer Surat Sederhana. Memungkinkan klien untuk mengirim email ke server email dan memungkinkan server untuk mengirim email ke server lain.
- **POP3** – Post Office Protocol version 3. Memungkinkan klien untuk mengambil email dari server email dan mengunduh email ke aplikasi email lokal klien.
- **IMAP** – Protokol Akses Pesan Internet. Memungkinkan client mengakses email yang disimpan di server email serta mengelola email di server.

Transfer File

- **FTP** – Protokol Transfer File. Menetapkan aturan yang memungkinkan pengguna di satu host untuk mengakses dan mentransfer file ke dan dari host lain melalui jaringan. FTP adalah protokol pengiriman file yang andal, berorientasi koneksi, dan diakui.
- **SFTP** – Protocol Transfer File SSH. Sebagai perpanjangan dari protokol Secure Shell (SSH), SFTP dapat digunakan untuk membuat sesi transfer file yang aman di mana transfer file dienkripsi. SSH adalah metode login jarak jauh aman yang biasanya digunakan untuk mengakses baris perintah perangkat.

- **TFTP** – Protocol Transfer File Sepele. Protokol transfer file sederhana tanpa koneksi dengan upaya terbaik, pengiriman file tanpa pengakuan. Ini menggunakan lebih sedikit overhead daripada FTP.

Web dan Layanan Web

- **HTTP** – Protokol Transfer Hiperteks. Serangkaian aturan untuk bertukar teks, gambar grafik, suara, video, dan file multimedia lainnya di World Wide Web.
- **HTTPS** – HTTP Aman. Bentuk HTTP aman yang mengenkripsi data yang dipertukarkan melalui World Wide Web.
- **REST** – Transfer Negara Perwakilan. Layanan web yang menggunakan antarmuka pemrograman aplikasi (API) dan permintaan HTTP untuk membuat aplikasi web.

Transport Layer

Connection-Oriented

Transmission Control Protocol (TCP) – Memungkinkan komunikasi yang andal antara proses yang berjalan pada host terpisah dan menyediakan transmisi yang anda dan diakui yang mengkonfirmasi pengiriman yang berhasil

Connectionless

User Data Protocol (UDP) – Mengaktifkan proses yang berjalan di satu host untuk mengirim paket ke proses yang berjalan di host lain. Namun, UDP tidak mengkonfirmasi transmisi paket yang berhasil

Internet Layer

Protocol Internet

- **IPv4** – Internet Protocol version 4. Menerima segmen pesan dari lapisan transport, mengemas pesan ke dalam paket, dan mengalamatkan paket untuk pengiriman ujung ke ujung melalui jaringan. IPv4 menggunakan alamat 32-bit.
- **IPv6** – IP versi 6. Mirip dengan IPv4 tetapi menggunakan alamat 128-bit.
- **NAT** – Terjemahan Alamat Jaringan. Menerjemahkan alamat IPv4 dari jaringan pribadi menjadi alamat IPv4 publik yang unik secara global.

Messaging

- **ICMPv4** – Protocol Pesan Kontrol Internet untuk IPv4. Memberikan umpan balik dari host tujuan ke host sumber tentang kesalahan dalam pengiriman paket.
- **ICMPv6** – ICMP untuk IPv6. Fungsionalitas yang mirip dengan ICMPv4 tetapi digunakan untuk paket IPv6.
- **ICMPv6 ND** – ICMPv6 *Neighbour Discovery*. Termasuk empat pesan protokol yang digunakan untuk resolusi alamat dan deteksi alamat duplikat.

Routing Protocol

- **Open Short Path First (OSPF)** – Protokol routing link-state yang menggunakan desain hierarki berdasarkan area. OSPF adalah *routing protocol* interior standar terbuka.
- **Enhanced Interior Gateway Routing Protocol (EIGRP)** – Protokol perutean standar terbuka yang dikembangkan oleh Cisco yang menggunakan metrik komposit berdasarkan bandwidth, penundaan, beban, dan keandalan.
- **Border Gateway Protocol (BGP)** – Protokol perutean gateway eksterior standar terbuka yang digunakan antara Penyedia Layanan Internet (ISP). BGP juga biasa digunakan antara ISP dan klien pribadi besar mereka untuk bertukar informasi perutean.

Network Access Layer

Resolusi Alamat

Address Resolution Protocol (ARP) – Menyediakan pemetaan alamat dinamis antara alamat IPv4 dan alamat perangkat keras.

Catatan : Anda mungkin melihat dokumentasi lain yang menyatakan bahwa ARP beroperasi di Internet Layer (OSI Layer 3). Namun, dalam kursus ini kami menyatakan bahwa ARP beroperasi pada lapisan Akses Jaringan (OSI Layer 2) karena tujuan utamanya adalah menemukan alamat MAC tujuan. Alamat MAC adalah alamat Layer 2.

Data Link Protocol

- **Ethernet** – Mendefinisikan aturan untuk kabel dan standar pensinyalan **Network Access Layer**
- **WLAN** – Jaringan Area Lokal Nirkabel. Menentukan aturan untuk pensinyalan nirkabel di frekuensi radio 2,4 GHz dan 5 GHz.

Organisasi Standard

Saat membeli ban baru untuk sebuah mobil, ada banyak pabrikan yang bisa Anda pilih. Masing-masing memiliki setidaknya satu jenis ban yang cocok untuk mobil Anda. Itu karena industri otomotif menggunakan standar saat membuat mobil. Ini sama dengan protokol. Karena ada banyak produsen komponen jaringan yang berbeda, mereka semua harus menggunakan standar yang sama. Dalam jaringan, standar dikembangkan oleh organisasi standar internasional.

Open Standard

Open Standard mendorong *interoperabilitas*, *persaingan*, dan *inovasi*. Mereka juga menjamin bahwa produk dari tidak ada satu perusahaan pun yang dapat memonopoli pasar atau memiliki keuntungan yang tidak adil atas persaingannya.

Contoh bagusnya adalah saat membeli **Router** nirkabel untuk rumah. Ada banyak pilihan berbeda yang tersedia dari berbagai vendor, yang semuanya menggabungkan protokol standar seperti IPv4, IPv6, DHCP, SLAAC, Ethernet, dan 802.11 Wireless LAN. **Open Standard** ini juga memungkinkan klien yang menjalankan sistem operasi Apple OS X untuk mengunduh halaman web dari server web yang menjalankan sistem operasi Linux. Ini karena kedua sistem operasi menerapkan protokol standar terbuka, seperti yang ada dalam rangkaian protokol TCP / IP.

Open Standard biasanya adalah organisasi *neutral-vendor*, didirikan untuk mengembangkan dan mempromosikan konsep **Open Standard**. Organisasi-organisasi ini penting dalam memelihara internet terbuka dengan spesifikasi dan protokol yang dapat diakses secara bebas yang dapat diterapkan oleh vendor mana pun.

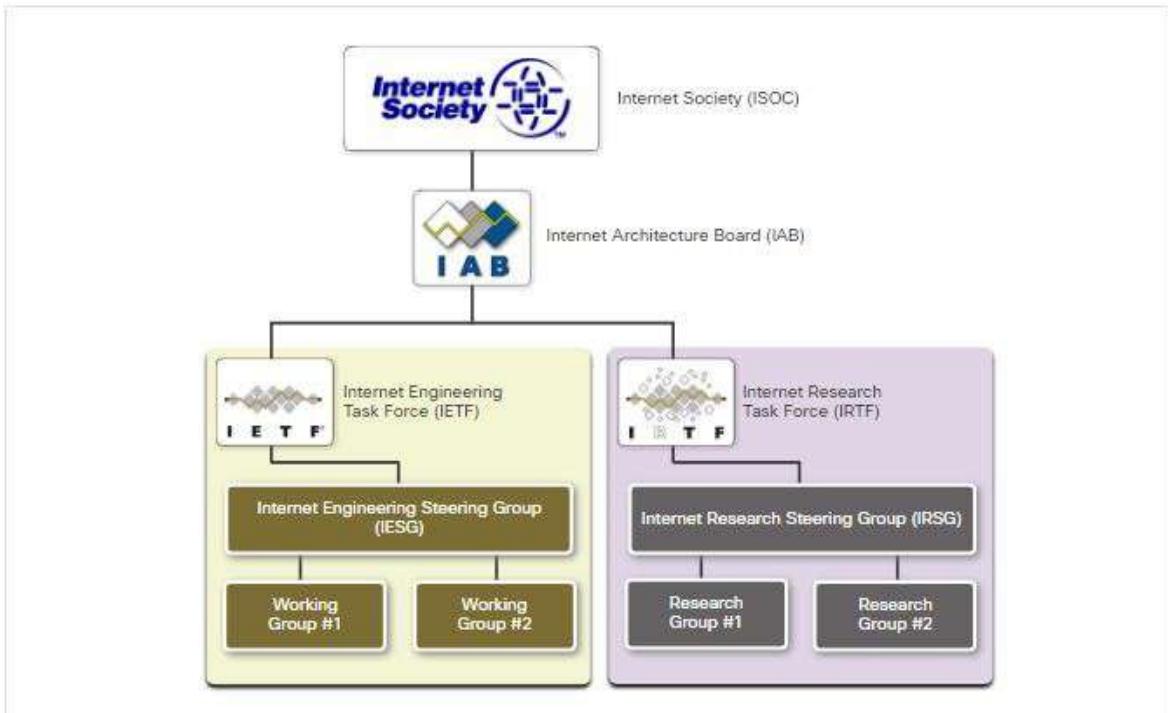
Open Standard dapat menyusun seperangkat aturan sepenuhnya sendiri atau, dalam kasus lain, dapat memilih protokol berpemilik sebagai dasar standar. Jika protokol berpemilik digunakan, biasanya melibatkan vendor yang membuat protokol.



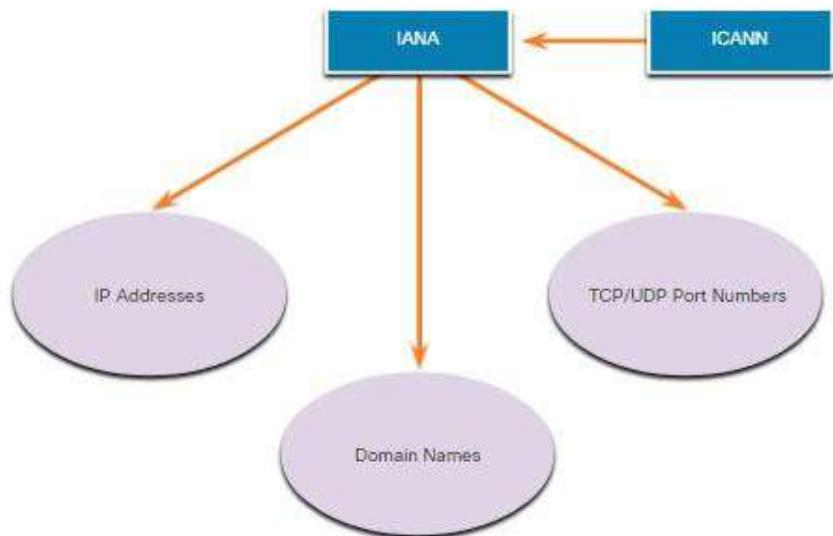
Beberapa Organisasi Open Standard

Standar Internet

Berbagai organisasi memiliki tanggung jawab yang berbeda untuk mempromosikan dan membuat standar untuk internet dan protokol TCP / IP.



- **Internet Society (ISOC)** – Bertanggung jawab untuk mempromosikan perkembangan terbuka dan evolusi penggunaan internet di seluruh dunia.
- **Internet Architecture Board (IAB)** – Bertanggung jawab atas keseluruhan pengelolaan dan pengembangan standar internet.
- **Internet Engineering Task Force (IETF)** – Mengembangkan, memperbarui, dan memelihara teknologi internet dan TCP / IP. Ini termasuk proses dan dokumen untuk mengembangkan protokol baru dan memperbarui protokol yang ada, yang dikenal sebagai dokumen Request for Comments (RFC).
- **Internet Research Task Force (IRTF)** – Berfokus pada penelitian jangka panjang terkait internet dan protokol TCP / IP seperti Anti-Spam Research Group (ASRG), Crypto Forum Research Group (CFRG), dan Peer-to-Peer Research Group (P2PRG).



- **Internet Corporation for Assigned Names and Numbers (ICANN)** – Berbasis di Amerika Serikat, ICANN mengoordinasikan alokasi alamat IP, pengelolaan nama domain, dan penetapan informasi lain yang digunakan dalam protokol TCP / IP.
- **Internet Assigned Numbers Authority (IANA)** – Bertanggung jawab untuk mengawasi dan mengelola alokasi alamat IP, manajemen nama domain, dan pengidentifikasi protokol untuk ICANN.

Standar Elektronik dan Komunikasi

Organisasi standar lainnya memiliki tanggung jawab untuk mempromosikan dan menciptakan standar elektronik dan komunikasi yang digunakan untuk mengirimkan paket IP sebagai sinyal elektronik melalui media kabel atau nirkabel.

Organisasi standar ini meliputi:

- **Institute of Electrical and Electronics Engineers (IEEE)** – Organisasi teknik kelistrikan dan elektronik yang didedikasikan untuk memajukan inovasi teknologi dan menciptakan standar di berbagai bidang industri termasuk tenaga dan energi, perawatan kesehatan, telekomunikasi, dan jaringan . Standar penting jaringan IEEE mencakup standar 802.3 Ethernet dan 802.11 WLAN. Cari di internet untuk standar jaringan IEEE lainnya.
- **Electronic Industries Alliance (EIA)** – Organisasi terkenal karena standarnya yang berkaitan dengan kabel listrik, konektor, dan rak 19 inci yang digunakan untuk memasang peralatan jaringan.
- **Telecommunications Industry Association (TIA)** – Organisasi yang bertanggung jawab untuk mengembangkan standar komunikasi di berbagai bidang termasuk peralatan radio, menara seluler, perangkat Voice over IP (VoIP), komunikasi satelit, dan banyak lagi.
- **International Telecommunications Union-Telecommunication Standardization Sector (ITU-T)** – Salah satu organisasi standar komunikasi terbesar dan tertua. ITU-T menetapkan standar untuk kompresi video, Internet Protocol Television (IPTV), dan komunikasi broadband, seperti digital subscriber line (DSL).

Model Referensi

Anda tidak dapat benar-benar melihat paket nyata berjalan melintasi jaringan nyata, seperti Anda dapat melihat komponen mobil yang sedang dipasang di jalur perakitan. jadi, ada baiknya memiliki cara berpikir tentang jaringan sehingga Anda dapat membayangkan apa yang terjadi. Sebuah model berguna dalam situasi ini.

Manfaat Menggunakan Layered Model

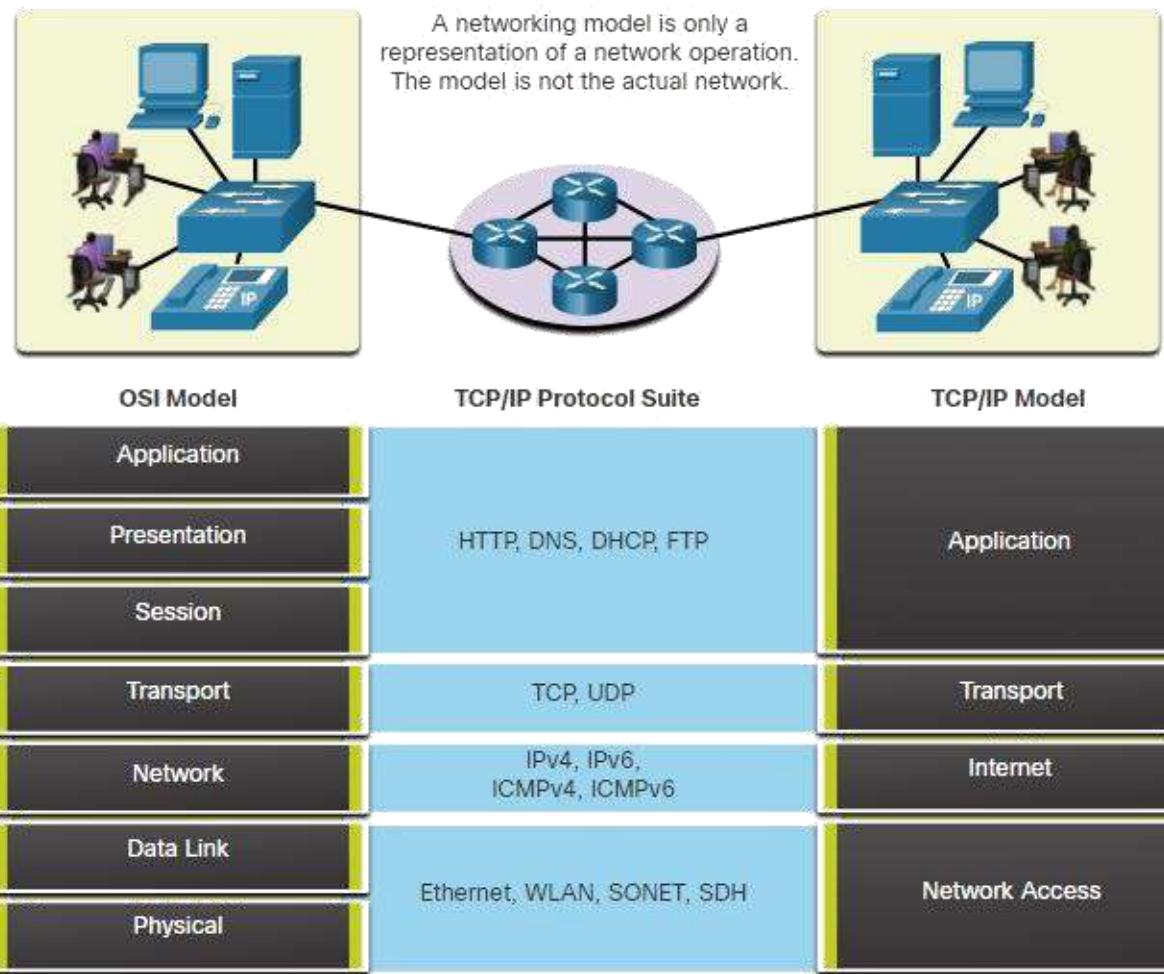
Konsep kompleks seperti bagaimana jaringan beroperasi bisa jadi sulit untuk dijelaskan dan dipahami. Untuk alasan ini, Layered Model digunakan untuk modularisasi operasi jaringan menjadi layer yang dapat dikelola.

Berikut adalah keuntungan menggunakan Layered Model untuk mendeskripsikan protokol dan operasi jaringan:

- Membantu dalam desain protokol karena protokol yang beroperasi pada *Layer* tertentu telah menentukan informasi yang mereka tindak dan *Interface* yang ditentukan ke *Layer* di **Upper Layer** dan di **Bottom Layer**
- Membina persaingan karena produk dari vendor yang berbeda dapat bekerja sama
- Mencegah perubahan teknologi atau kemampuan dalam satu *Layer* agar tidak memengaruhi *Layer* lain di **Upper Layer** dan di **Bottom Layer**
- Menyediakan bahasa umum untuk mendeskripsikan fungsi dan kapabilitas jaringan

Seperti yang ditunjukkan pada gambar, ada dua **Layered Models** yang digunakan untuk menggambarkan operasi jaringan:

- Model Referensi Open System Interconnection (OSI)
- Model Referensi TCP / IP



Model Referensi OSI

Model referensi **OSI** menyediakan daftar lengkap fungsi dan layanan yang dapat terjadi di setiap *Layer*. Jenis model ini memberikan konsistensi dalam semua jenis protokol dan layanan jaringan dengan mendeskripsikan apa yang harus dilakukan pada *Layer* tertentu, tetapi tidak menentukan bagaimana cara melakukannya.

Referensi OSI juga menjelaskan interaksi setiap *Layer* dengan *Layer* langsung di **Upper Layer** dan **Bottom Layer**. Protokol TCP / IP yang dibahas ini disusun berdasarkan model OSI dan TCP / IP. Tabel menunjukkan detail tentang setiap *Layer* model OSI. Fungsionalitas setiap *Layer* dan hubungan antar *Layer* akan menjadi lebih karena protokolnya dibahas lebih detail.

| Layer OSI | Deskripsi |
|--------------|---|
| Application | berisi protokol yang digunakan untuk komunikasi proses-ke-proses. |
| Presentation | menyediakan representasi umum dari data yang ditransfer antara layanan Application Layer |
| Session | menyediakan layanan ke Presentation Layer untuk mengatur dialognya dan mengelola pertukaran data. |
| Transport | mendefinisikan layanan untuk mensegmentasikan, mentransfer, dan memasang kembali data untuk komunikasi individu antara End Device |
| Network | menyediakan layanan untuk bertukar potongan data individu melalui jaringan antara End Device yang diidentifikasi. |
| Data Link | menjelaskan metode untuk bertukar frame data antara perangkat melalui media umum |
| Physical | menjelaskan cara mekanis, listrik, fungsional, dan prosedural untuk mengaktifkan, memelihara, dan menonaktifkan koneksi fisik untuk transmisi bit ke dan dari perangkat jaringan. |

Model Protokol TCP / IP

Model protokol TCP / IP untuk komunikasi internetwork dibuat pada awal 1970-an dan terkadang disebut sebagai **model internet**. Jenis model ini sangat cocok dengan struktur rangkaian protokol tertentu. Model TCP / IP adalah model protokol karena menjelaskan fungsi yang terjadi pada setiap lapisan protokol dalam rangkaian TCP / IP. TCP / IP juga digunakan sebagai model referensi. Tabel menunjukkan detail tentang setiap lapisan model OSI.

| Model TCP / IP | Deskripsi |
|----------------|--|
| Application | Merepresentasikan data untuk pengguna, ditambah encoding dan kontrol dialog. |
| Transportasi | Mendukung komunikasi antara berbagai perangkat di berbagai jaringan. |
| Internet | Menentukan jalur terbaik melalui jaringan. |

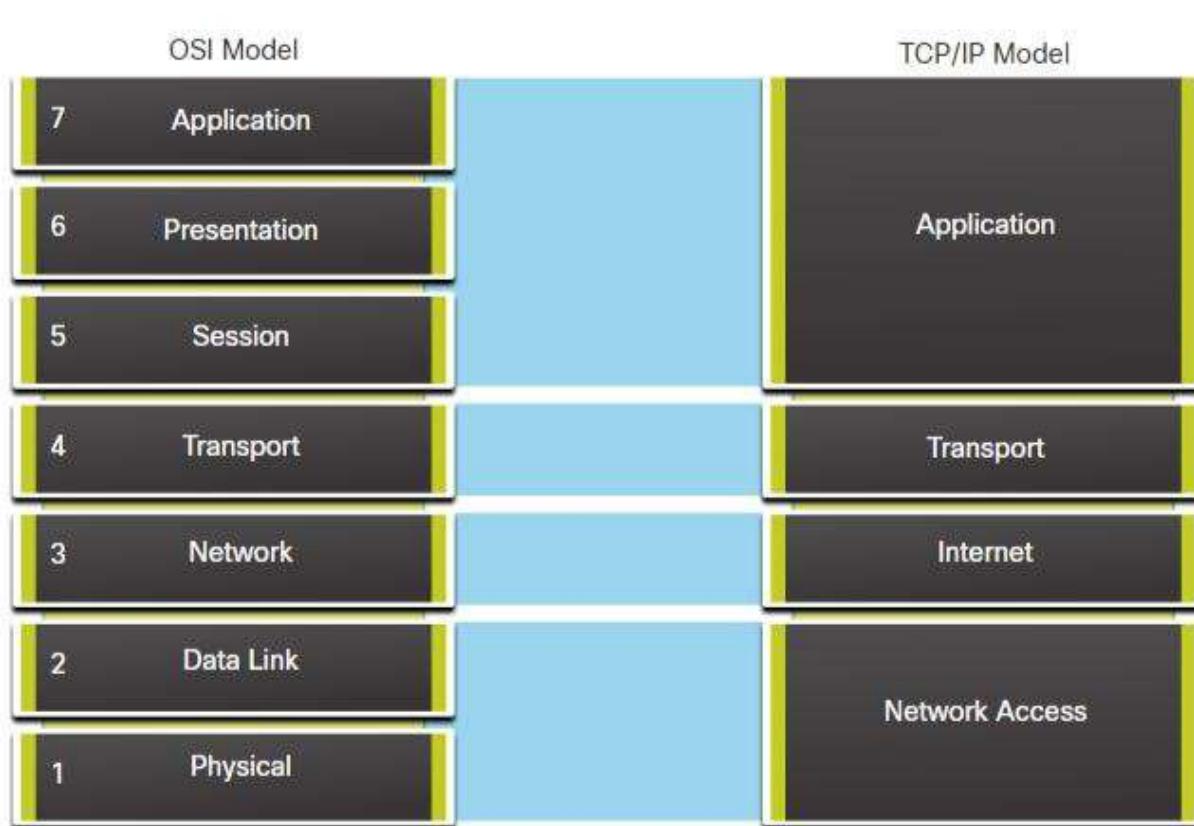
| | |
|----------------|---|
| Network Access | Mengontrol perangkat keras dan media yang membentuk jaringan. |
|----------------|---|

Definisi standar dan protokol TCP / IP didiskusikan dalam forum publik dan ditentukan dalam rangkaian IETF RFC yang tersedia untuk umum. Sebuah RFC dibuat oleh para insinyur jaringan dan dikirim ke anggota IETF lainnya untuk mendapatkan *comments*.

Perbandingan Model OSI dan TCP / IP

Protokol yang menyusun rangkaian protokol TCP / IP juga dapat dijelaskan dalam kerangka model referensi OSI. Dalam model OSI, Network Access Layer dan Application Layer dari model TCP / IP dibagi lagi untuk menggambarkan fungsi *diskrit* yang harus terjadi pada lapisan ini.

Pada Network Access Layer, rangkaian protokol TCP / IP tidak menentukan protokol mana yang akan digunakan saat transmisi melalui media fisik; itu hanya menjelaskan *handoff* dari internet layer ke protokol jaringan fisik. Lapisan OSI 1 dan 2 membahas prosedur yang diperlukan untuk mengakses media dan sarana fisik untuk mengirim data melalui jaringan.



Kesamaan utama ada pada **Transport Layer** dan **Network Layer**; namun, kedua model tersebut berbeda dalam cara menghubungkannya dengan *Layer* di **Upper Layer** dan di **Bottom Layer** setiap *Layer*:

- OSI Layer 3, Network Layer, memetakan langsung ke Internet Layer TCP / IP. Layer ini digunakan untuk mendeskripsikan protokol yang menangani dan merutekan pesan melalui sebuah internetwork.
- OSI Layer 4, Transport Layer, memetakan langsung ke Transport Layer TCP / IP. Layer ini menjelaskan layanan dan fungsi umum yang menyediakan pengiriman data yang teratur dan andal antara host sumber dan tujuan.
- TCP / IP, Application Layer mencakup beberapa protokol yang menyediakan fungsionalitas khusus untuk berbagai aplikasi End Device. Model OSI Layers 5, 6, dan 7 digunakan sebagai acuan bagi pengembang dan vendor perangkat lunak aplikasi untuk menghasilkan aplikasi yang beroperasi pada jaringan.
- Baik model TCP / IP dan OSI biasanya digunakan saat mengacu pada protokol di berbagai *Layer*. Karena model OSI memisahkan Data Link dari Physical Layer, ini biasanya digunakan saat mengacu pada Bottom Layer

Enkapsulasi Data

Mensegmentasi Pesan

Mengetahui model referensi **OSI** dan model protokol **TCP / IP** akan berguna saat Anda mempelajari tentang bagaimana data dienkapsulasi saat bergerak melintasi jaringan. Ini tidak sesederhana mengirim surat secara fisik melalui sistem surat.

Secara teori, satu komunikasi, seperti video atau pesan email dengan banyak lampiran besar, dapat dikirim melalui jaringan dari sumber ke tujuan sebagai aliran bit yang besar dan tidak terputus. Namun, ini akan menimbulkan masalah bagi perangkat lain yang perlu menggunakan saluran atau *Link* komunikasi yang sama. Aliran data yang besar ini akan mengakibatkan **delay** yang signifikan. Lebih lanjut, jika ada link dalam infrastruktur jaringan yang saling berhubungan yang gagal selama transmisi, pesan lengkap akan hilang dan harus dikirim ulang secara penuh.

Pendekatan yang lebih baik adalah dengan membagi data menjadi bagian-bagian yang lebih kecil dan lebih mudah dikelola untuk dikirim melalui jaringan. **Segmentasi** adalah *proses membagi aliran data menjadi unit-unit yang lebih kecil untuk transmisi melalui jaringan*. Segmentasi diperlukan karena jaringan data menggunakan rangkaian protokol **TCP / IP** yang mengirim data dalam paket IP individu. Setiap paket dikirim secara terpisah, serupa dengan pengiriman surat panjang sebagai rangkaian kartu pos individu. Paket yang berisi segmen untuk tujuan yang sama dapat dikirim melalui jalur yang berbeda.

Ini mengarah pada pengelompokan pesan yang memiliki dua manfaat utama:

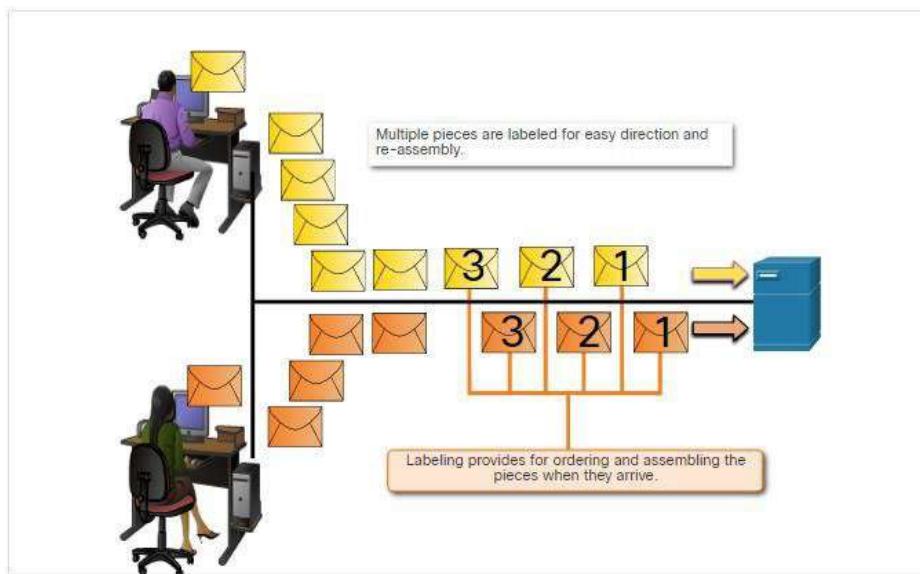
Meningkatkan kecepatan – Karena aliran data yang besar tersegmentasi menjadi paket-paket, sejumlah besar data dapat dikirim melalui jaringan tanpa mengikat *Link* komunikasi. Ini memungkinkan banyak percakapan berbeda untuk disisipkan di jaringan yang disebut **multiplexing**.

Meningkatkan efisiensi -Jika satu segmen gagal mencapai tujuannya karena kegagalan jaringan atau kemacetan jaringan, hanya segmen itu yang perlu ditransmisikan ulang alih-alih mengirim ulang seluruh aliran data.

Sequencing

Tantangan untuk menggunakan **segmentasi** dan **multiplexing** untuk mengirimkan pesan melalui jaringan adalah tingkat kerumitan yang ditambahkan ke dalam proses. Bayangkan jika Anda harus mengirim surat 100 halaman, tetapi setiap amplop hanya dapat menampung satu halaman. Oleh karena itu, diperlukan 100 amplop dan setiap amplop harus dialamatkan satu per satu. Ada kemungkinan bahwa surat 100 halaman dalam 100 amplop berbeda tiba rusak. Akibatnya, informasi di dalam amplop perlu menyertakan nomor urut untuk memastikan bahwa penerima dapat memasang kembali halaman-halaman tersebut dengan urutan yang benar.

Dalam komunikasi jaringan, setiap segmen pesan harus melalui proses yang serupa untuk memastikan bahwa pesan tersebut sampai ke tujuan yang benar dan dapat dipasang kembali menjadi konten pesan asli, seperti yang ditunjukkan pada gambar. **TCP** bertanggung jawab untuk mengurutkan individu.

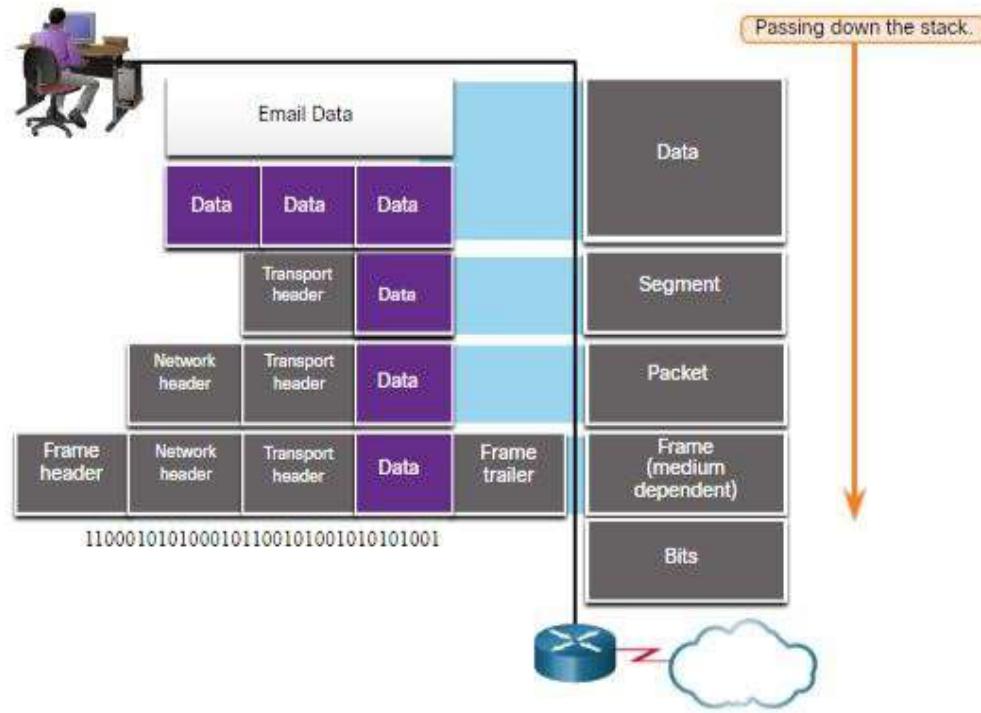


Protocol Data Unit (PDU)

Saat data aplikasi diturunkan ke *stack* protokol dalam perjalanannya untuk ditransmisikan melalui media jaringan, berbagai informasi protokol ditambahkan di setiap tingkat. Ini dikenal sebagai proses enkapsulasi.

Catatan : Meskipun UDP PDU disebut datagram, paket IP terkadang juga disebut sebagai datagram IP.

Bentuk yang diambil sepotong data pada setiap *Layer* disebut unit data protokol (PDU). Selama enkapsulasi, setiap *Layer* yang berhasil meng enkapsulasi PDU yang diterimanya dari Upper Layer sesuai dengan protokol yang digunakan. Pada setiap tahapan proses, PDU memiliki nama yang berbeda untuk mencerminkan fungsi barunya. Meskipun tidak ada konvensi penamaan universal untuk PDU, dalam Materi ini, PDU diberi nama sesuai dengan protocol suite TCP / IP. PDU untuk setiap bentuk data ditunjukkan pada gambar.



- **Data** – Istilah umum untuk PDU yang digunakan pada **Application Layer**
- **Segment** – **Transport layer** PDU
- **Packet** – **Network Layer** PDU
- **Frame** – **Data Link Layer** PDU
- **Bits** – **Physical Layer** PDU yang digunakan saat mentransmisikan data secara fisik melalui media

Catatan : Jika header Transport adalah TCP, maka itu adalah segment. Jika header Transport adalah UDP maka itu adalah datagram.

Contoh Enkapsulasi

Saat pesan dikirim di jaringan, proses enkapsulasi bekerja dari Upper Layer ke Bottom Layer. Pada setiap *Layer*, informasi *Layer* atas dianggap data dalam protokol yang dienkapsulasi. Misalnya, segmen TCP dianggap sebagai data dalam paket IP.

Contoh De-enkapsulasi

Proses ini dibalik pada host penerima dan dikenal sebagai de-enkapsulasi. De-enkapsulasi adalah proses yang digunakan oleh perangkat penerima untuk menghapus satu atau lebih header protokol. Data di-de-enkapsulasi saat bergerak naik tumpukan menuju aplikasi End Device.

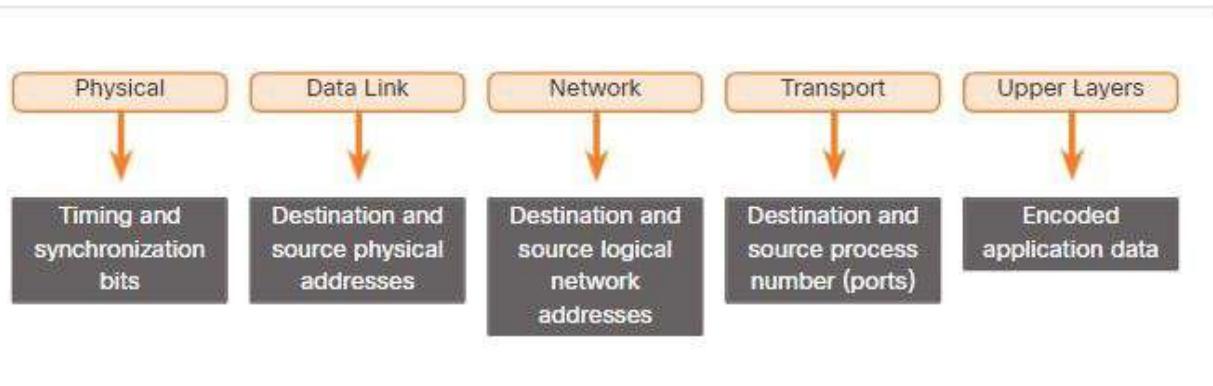
Data Access

Seperti yang baru saja Anda pelajari, penting untuk mengelompokkan pesan dalam jaringan. Tetapi pesan-pesan yang tersegmentasi itu tidak akan kemana-mana jika tidak ditangani dengan benar. Materi ini memberikan gambaran umum tentang alamat jaringan.

Alamat / Addresses

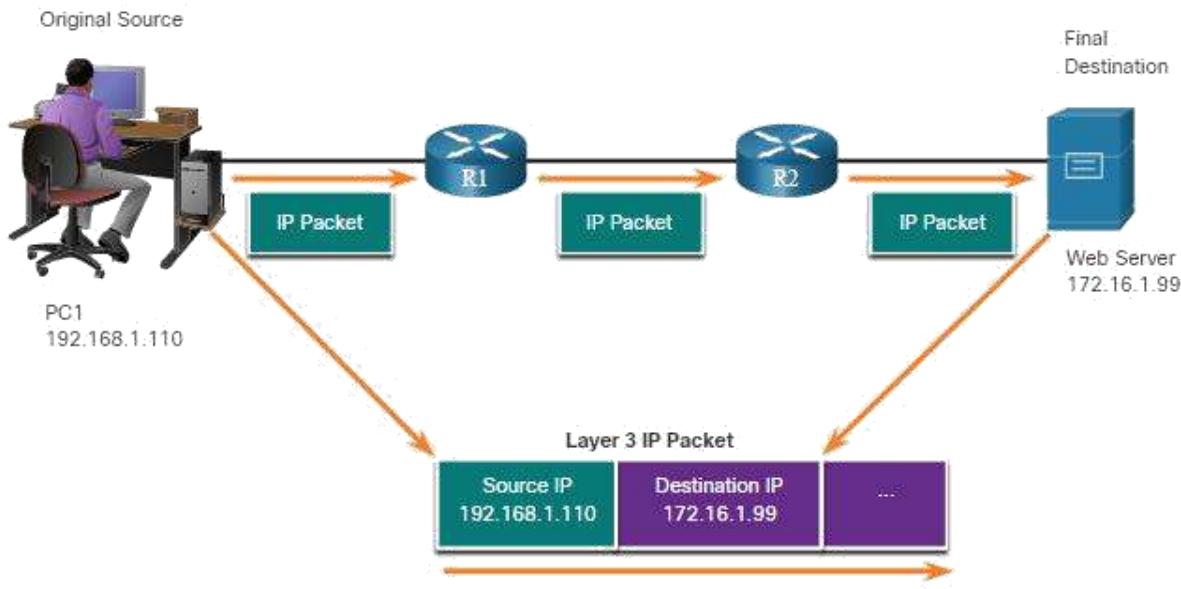
Network Layer dan **Data Link** bertanggung jawab untuk mengirimkan data dari perangkat sumber ke perangkat tujuan. Seperti yang ditunjukkan pada gambar, protokol di kedua *Layer* berisi alamat sumber dan tujuan, tetapi alamatnya memiliki tujuan yang berbeda:

- **Network layer source and destination addresses** – Bertanggung jawab untuk mengirimkan paket IP dari sumber asli ke tujuan akhir, yang mungkin berada di jaringan yang sama atau jaringan jarak jauh.
- **Data link layer source and destination addresses** – Bertanggung jawab untuk mengirimkan *frame* data link dari satu Network Interface Connector (NIC) ke NIC lain di jaringan yang sama.



Layer 3 Or Logical Address

IP Address adalah Network Layer, atau Layer 3, atau Logical Layer yang digunakan untuk mengirimkan paket IP dari sumber asli ke tujuan akhir, seperti yang ditunjukkan pada gambar.



Paket IP berisi dua IP Address:

- **Source IP Address** – perangkat pengirim yang merupakan sumber asli paket.
- **Destination IP Address** – perangkat penerima, yang merupakan tujuan akhir paket.

Alamat IP berisi dua bagian:

- **Network portion (IPv4) or Prefix (IPv6)** – Bagian paling kiri dari alamat yang menunjukkan jaringan di mana alamat IP tersebut adalah anggota. Semua perangkat di jaringan yang sama akan memiliki network portion yang sama dari alamat tersebut.
- **Host portion (IPv4) or Interface ID (IPv6)** – Bagian tersisa dari alamat yang mengidentifikasi perangkat tertentu di jaringan. Porsi ini unik untuk setiap perangkat atau **Interface** di jaringan.

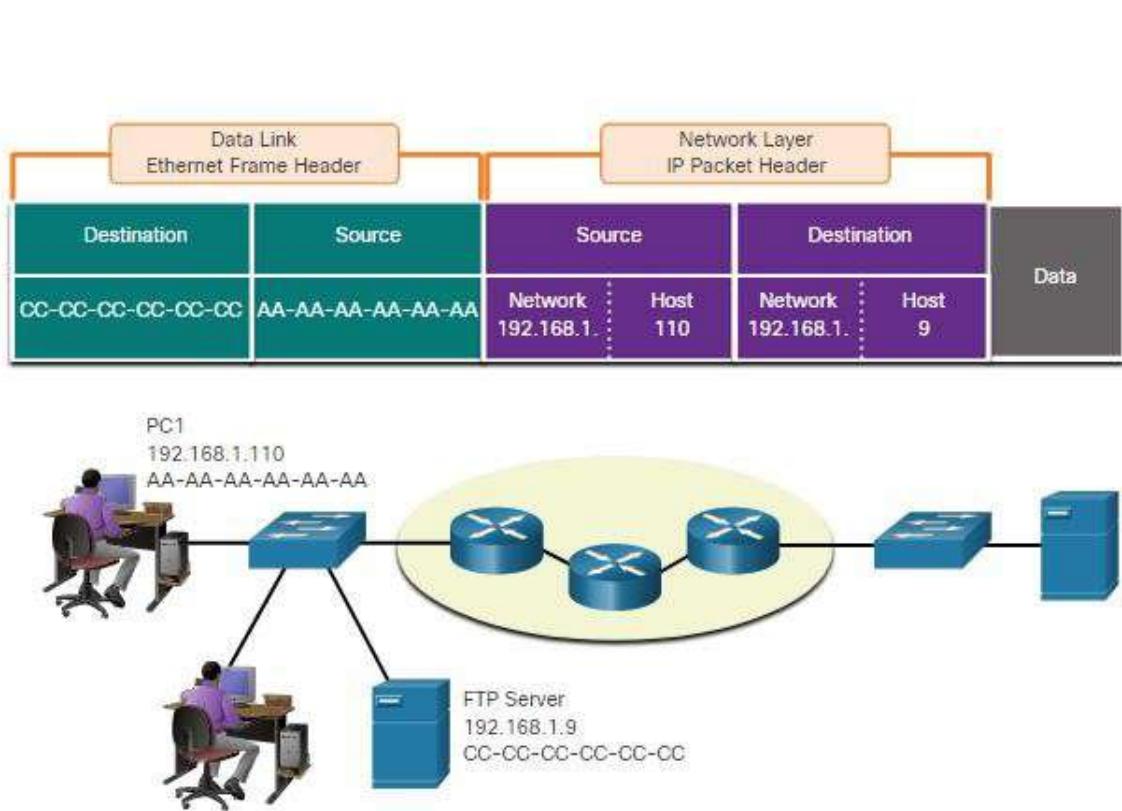
Catatan : Subnet mask (IPv4) atau prefix-length (IPv6) digunakan untuk mengidentifikasi bagian jaringan dari **IP Address** dari bagian host.

Perangkat di Jaringan yang Sama

Dalam contoh ini kita memiliki komputer klien, PC1, yang berkomunikasi dengan server FTP di jaringan IP yang sama.

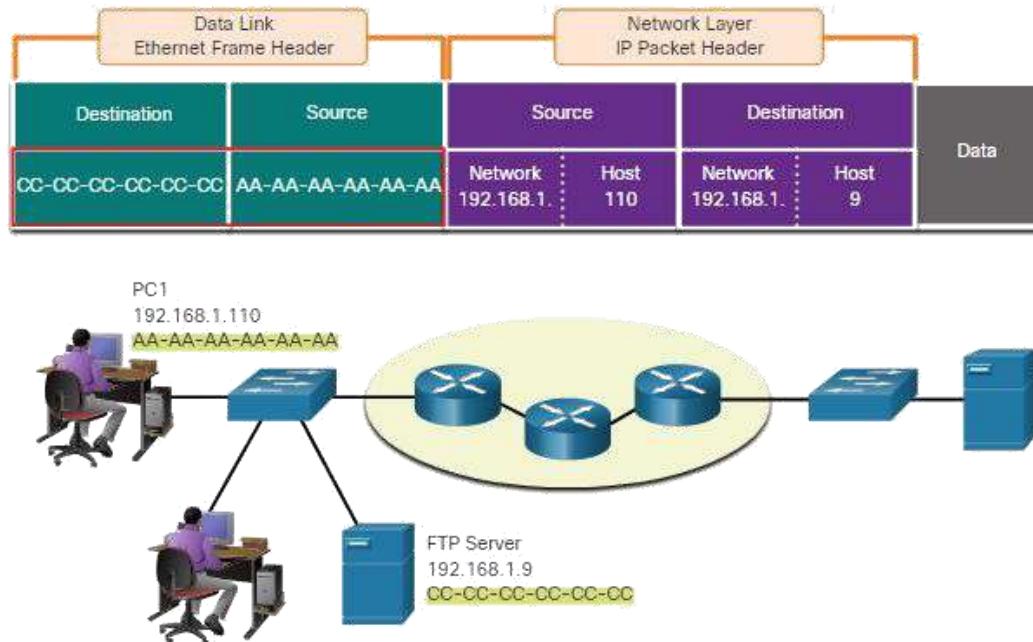
- **Source IPv4 address** – IPv4 dari perangkat pengirim adalah komputer klien PC1: 192.168.1.110.
- **Destination IPv4 address** – IPv4 dari perangkat penerima adalah server FTP: 192.168.1.9.

Perhatikan pada gambar bahwa network portion dari alamat **IPv4** sumber dan alamat **IPv4** tujuan berada di jaringan yang sama. Perhatikan pada gambar bahwa network portion dari alamat IPv4 sumber dan network portion dari alamat IPv4 tujuan adalah sama



Peran Data Link: Jaringan IP yang Sama

Ketika pengirim dan penerima paket IP berada di jaringan yang sama, Data Link Frame dikirim langsung ke perangkat penerima. Pada jaringan Ethernet, Data Link Frame dikenal sebagai alamat Ethernet Media Access Control (MAC), seperti yang disorot pada gambar.



Alamat MAC secara fisik tertanam di Ethernet NIC.

- **Source MAC address** – Ini adalah Data Link Frame, atau alamat MAC Ethernet, perangkat yang mengirim Data Link Frame dengan paket IP yang dienkapsulasi. Alamat MAC dari Ethernet NIC PC1 adalah AA-AA-AA-AA-AA-AA, ditulis dalam notasi heksadesimal.
- **Destination MAC address** – Jika perangkat penerima berada di jaringan yang sama dengan perangkat pengirim, ini adalah Data Link Frame dari perangkat penerima. Dalam contoh ini, alamat MAC tujuan adalah alamat MAC dari server FTP: CC-CC-CC-CC-CC-CC, ditulis dalam notasi heksadesimal.

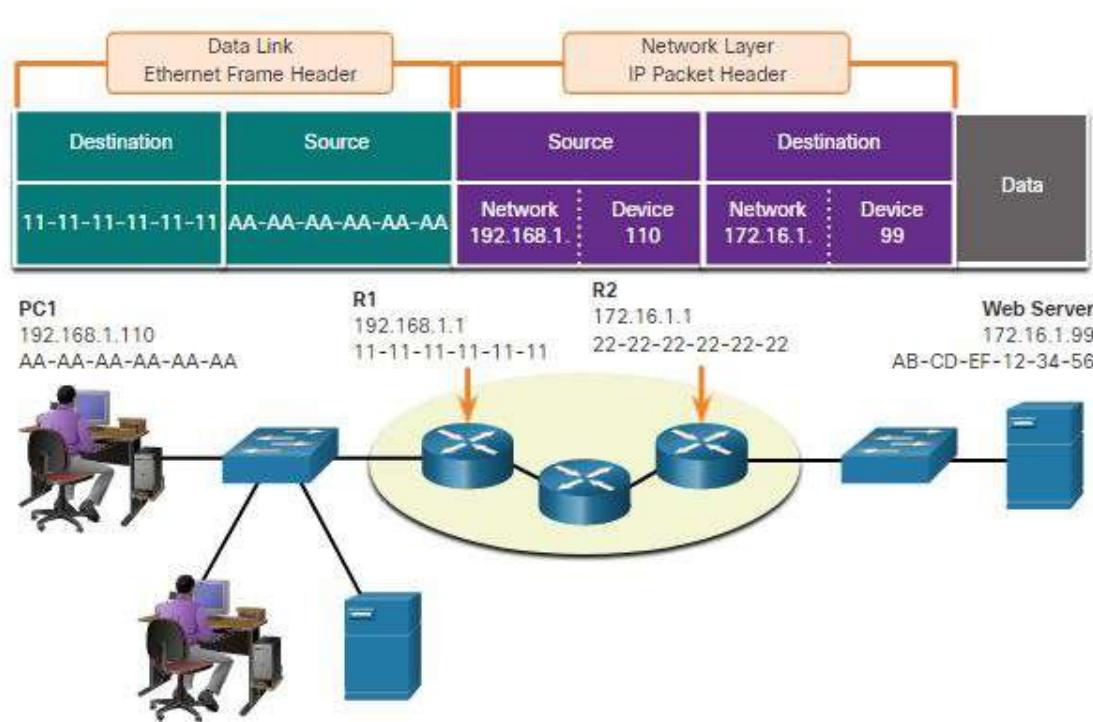
Frame dengan paket IP yang dienkapsulasi sekarang dapat dikirim dari PC1 langsung ke server FTP.

Peran Network Layer

Ketika pengirim paket berada di jaringan yang berbeda dari penerima, Sumber **IP Address** dan tujuan akan mewakili host di jaringan yang berbeda. Ini akan ditunjukkan oleh network portion dari alamat IP dari host tujuan.

- **Source IPv4 address** – IPv4 dari perangkat pengirim adalah komputer klien PC1: 192.168.1.110.
- **Destination IPv4 address** – IPv4 dari perangkat penerima adalah Server Web: 172.16.1.99.

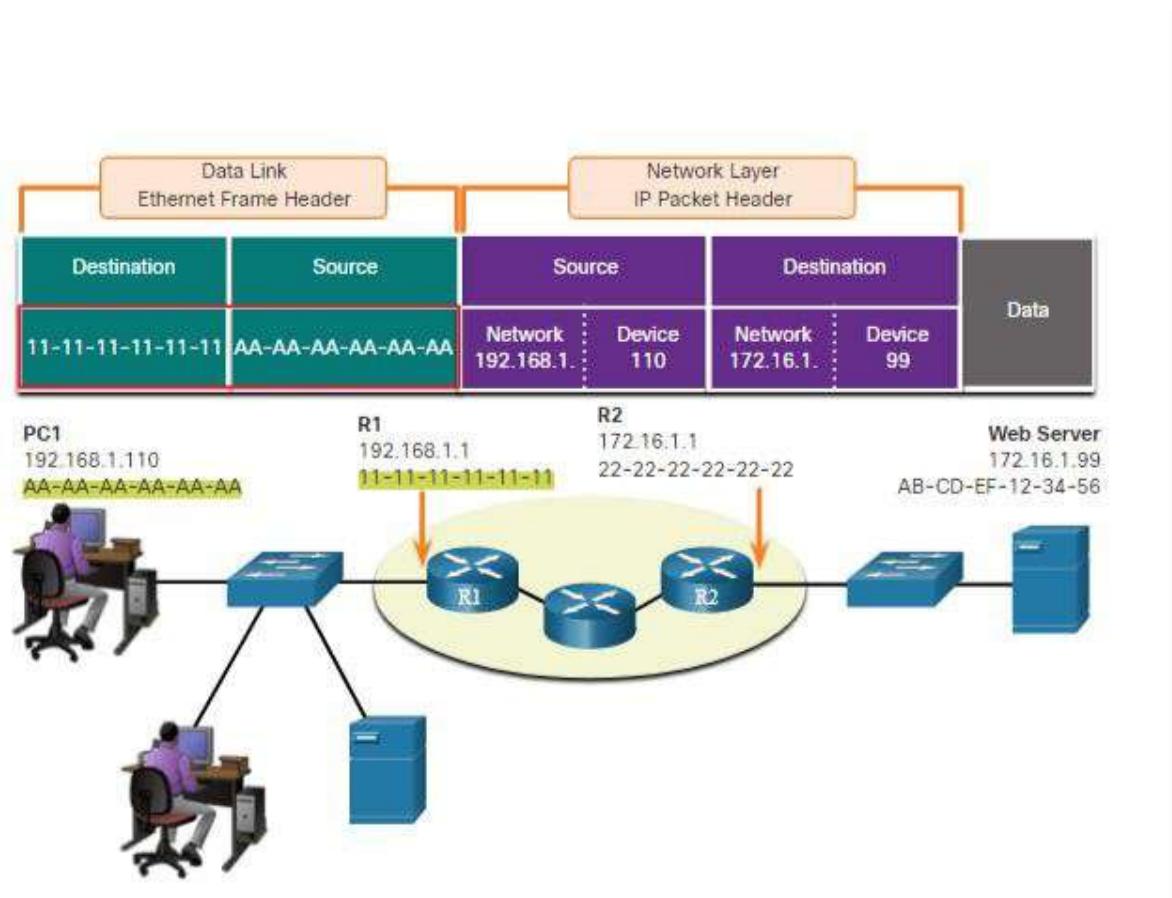
Perhatikan pada gambar bahwa network portion dari alamat IPv4 sumber dan alamat IPv4 tujuan berada di jaringan yang berbeda.



Peran Data Link: Jaringan IP Berbeda

Ketika pengirim dan penerima paket IP berada di jaringan yang berbeda, Ethernet frame tidak dapat dikirim langsung ke host tujuan karena host tidak dapat dijangkau secara langsung di jaringan pengirim. Frame Ethernet harus dikirim ke perangkat lain yang dikenal sebagai router atau Default Gateway. Dalam contoh, Default Gateway adalah R1. R1 memiliki Frame Ethernet yang berada di jaringan yang sama dengan PC1. Ini memungkinkan PC1 mencapai router secara langsung.

- Source MAC address – MAC Address dari perangkat pengirim adalah PC1. MAC Address dari Interface Ethernet PC1 adalah AA-AA-AA-AA-AA-AA.
- Destination MAC address – Jika perangkat penerima, tujuan IP Address, berada di jaringan yang berbeda dari perangkat pengirim, perangkat pengirim menggunakan alamat MAC Address dari *gateway* atau *Default Router*. Dalam contoh ini, tujuan MAC Address adalah MAC Address dari Interface Ethernet R1, 11-11-11-11-11-11. Ini adalah Interface yang terpasang ke jaringan yang sama dengan PC1, seperti yang ditunjukkan pada gambar.



Frame Ethernet dengan paket IP yang dienkapsulasi sekarang dapat dikirim ke R1. R1 meneruskan paket ke tujuan, Server Web. Ini mungkin berarti bahwa R1 meneruskan paket ke router lain atau langsung ke Server Web jika tujuannya ada di jaringan yang terhubung ke R1.

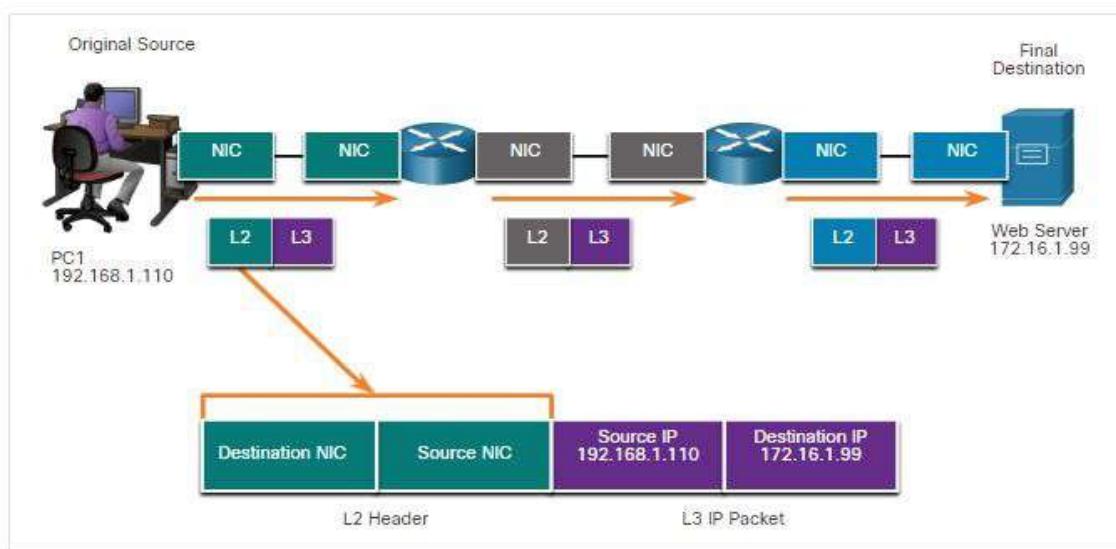
IP Address dari Default Gateway harus dikonfigurasi pada setiap host di jaringan lokal. Semua paket ke tujuan di jaringan jarak jauh dikirim ke Default Gateway.

Data Link Frame

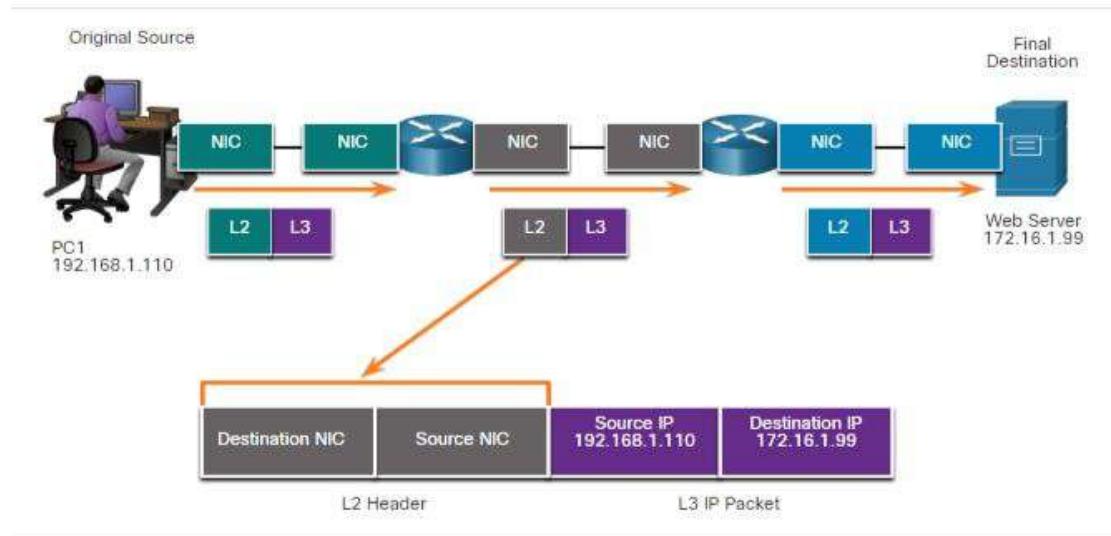
Data Link Frame Atau Layer 2 memiliki peran yang berbeda. Tujuan dari Data Link Frame adalah untuk mengirimkan Link Frame data dari satu Interface jaringan ke Interface jaringan lain di jaringan yang sama.

Sebelum paket IP dapat dikirim melalui jaringan kabel atau nirkabel, paket tersebut harus dienkapsulasi dalam Data Link Frame, sehingga dapat dikirim melalui media fisik.

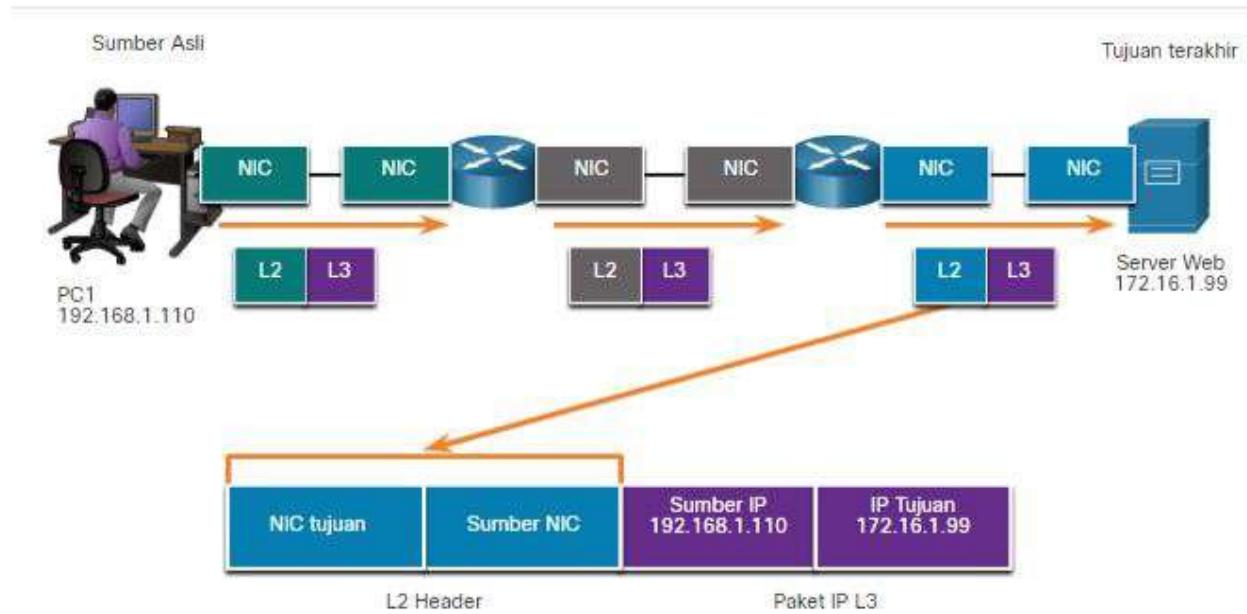
Host ke Router



Router to Router



Router to Server



Saat paket IP bergerak dari host-ke-router, router-ke-router, dan terakhir router-ke-host, di setiap titik sepanjang jalan paket IP dikenyal dalam Data Link Frame baru. Setiap Data Link Frame berisi Source data link address dari kartu NIC yang mengirim frame, dan Destination data link address dari kartu NIC yang menerima frame.

Layer 2, protokol data link hanya digunakan untuk mengirimkan paket dari NIC-ke-NIC di jaringan yang sama. Router menghapus informasi Layer 2 seperti yang diterima pada satu NIC dan menambahkan informasi Data Link Frame baru sebelum meneruskan keluar NIC dalam perjalanan menuju tujuan akhir.

Paket IP dikemas dalam **data link Frame** data yang berisi informasi berikut ini:

- **Source data link address** – Alamat fisik NIC yang mengirimkan **data link address**
- **Destination data link address** – Alamat fisik NIC yang menerima **data link address**. Alamat ini bisa jadi router hop berikutnya atau alamat perangkat tujuan akhir.

BAB 4

~ *Physical Layer* ~

Judul Bab: Physical Layer

Tujuan Bab: Menjelaskan bagaimana protokol physical layer, Services, dan network media support komunikasi di jaringan yang berbeda

Link Test Pemahaman : <https://forms.gle/vPTV49U5KvWwXByv7>

| Judul Materi | Tujuan Materi |
|-----------------------------------|--|
| Tujuan dibentuknya physical layer | Menjelaskan fungsi dan tujuan jaringan di physical layer |
| Karakteristik Physical Layer | Menjelaskan karakteristik dari physical layer |
| Kabel tembaga | Mengidentifikasi karakteristik dasar dari kabel tembaga |
| Kabel UTP | Menjelaskan bagaimana kabel UTP digunakan di jaringan ethernet |
| Kabel Fiber Optic | Menjelaskan bagaimana kabel fiber optic dan apa keunggulan dari kabel lain |
| Wireless Media | Menghubungkan perangkat dengan media kabel dan wireless |

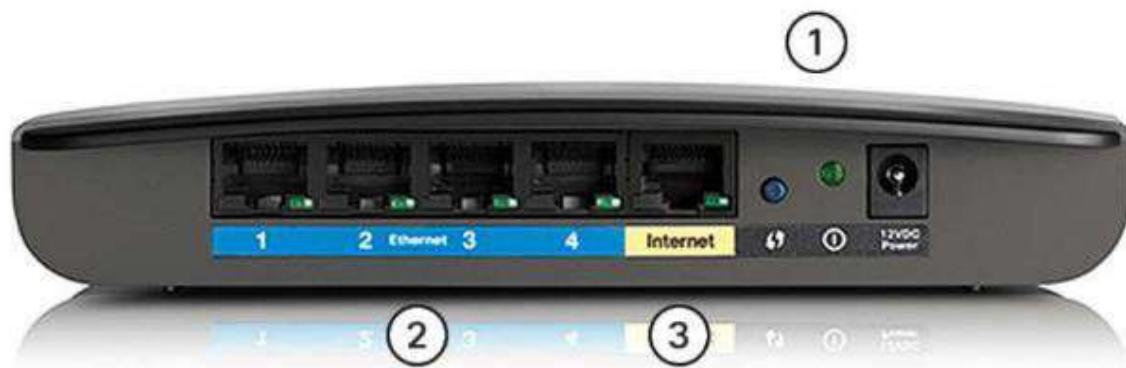
Tujuan dibentuknya physical layer

Baik menghubungkan ke printer di rumah atau situs web di negara lain, sebelum komunikasi jaringan dapat terjadi, koneksi fisik ke jaringan lokal harus dibuat. Koneksi fisik dapat berupa *cable connection* menggunakan kabel atau *wireless connection* menggunakan gelombang radio.

Physical Connection

Jenis koneksi fisik yang digunakan tergantung pada pengaturan jaringan. Misalnya, di banyak kantor perusahaan, karyawan memiliki komputer desktop atau laptop yang secara fisik terhubung, melalui kabel, ke **switch**. Ini biasanya menggunakan jaringan **kabel**. Data dikirim melalui kabel fisik.

Selain koneksi kabel, banyak bisnis juga menawarkan koneksi nirkabel untuk laptop, tablet, dan smartphone. Dengan perangkat nirkabel, data dikirim menggunakan gelombang radio. Konektivitas nirkabel biasa terjadi ketika individu dan bisnis sama-sama menemukan kelebihannya. Perangkat pada jaringan nirkabel harus terhubung ke titik akses nirkabel (AP) atau router nirkabel seperti yang ditunjukkan pada gambar.



Router nirkabel

Ini adalah komponen Access Point:

1. Antena nirkabel (Ini tertanam di dalam versi router yang ditunjukkan pada gambar di atas.)
2. Beberapa switchport Ethernet
3. Port internet

Mirip dengan kantor perusahaan, kebanyakan rumah menawarkan konektivitas kabel dan nirkabel ke jaringan. Gambar tersebut menunjukkan router rumah dan laptop yang terhubung ke jaringan area lokal (LAN).

Koneksi Kabel ke Router Nirkabel



Network Interface Cards

Network Interface Card (NIC) menghubungkan perangkat ke jaringan. Ethernet NIC digunakan untuk koneksi kabel, seperti yang ditunjukkan pada gambar, sedangkan NIC jaringan area lokal nirkabel (WLAN) digunakan untuk nirkabel. Perangkat **End Devices** mungkin menyertakan satu atau kedua jenis NIC. Printer jaringan, misalnya, mungkin hanya memiliki NIC Ethernet, dan oleh karena itu, harus terhubung ke jaringan menggunakan kabel Ethernet. Perangkat lain, seperti tablet dan smartphone, mungkin hanya berisi WLAN NIC dan harus menggunakan koneksi nirkabel.

Koneksi Kabel Menggunakan Ethernet NIC



Tidak semua koneksi fisik sama, dalam hal tingkat kinerja, saat menghubungkan ke jaringan.

Physical Layer

Physical Layer OSI menyediakan sarana untuk mengangkut bit yang membentuk Data Link Frame melalui media jaringan. Layer ini menerima Full Frame dari Data link Layer dan mengkodekan sebagai rangkaian sinyal yang dikirimkan ke media lokal. Bit yang dikodekan yang terdiri dari Frame diterima oleh End Devices atau Intermediary Devices.

Physical Layer mengkodekan Frame dan menciptakan sinyal gelombang listrik, optik, atau radio yang mewakili bit di setiap Frame. Sinyal ini kemudian dikirim melalui media, satu per satu.

Physical Layer tujuannya mengambil sinyal individu ini dari media, mengembalikannya ke representasi bit mereka, dan meneruskan bit ke Data Link Layer dengan Full Frame.

Karakteristik Physical Layer

Di Materi sebelumnya, Anda memperoleh gambaran umum tingkat tinggi tentang **Physical Layer** dan tempatnya di jaringan. Materi ini membahas lebih dalam tentang spesifikasi **Physical Layer**. Termasuk didalamnya adalah komponen dan media yang digunakan untuk membangun suatu jaringan, serta standar yang dibutuhkan agar semuanya dapat bekerja sama.

Standard Physical Layer

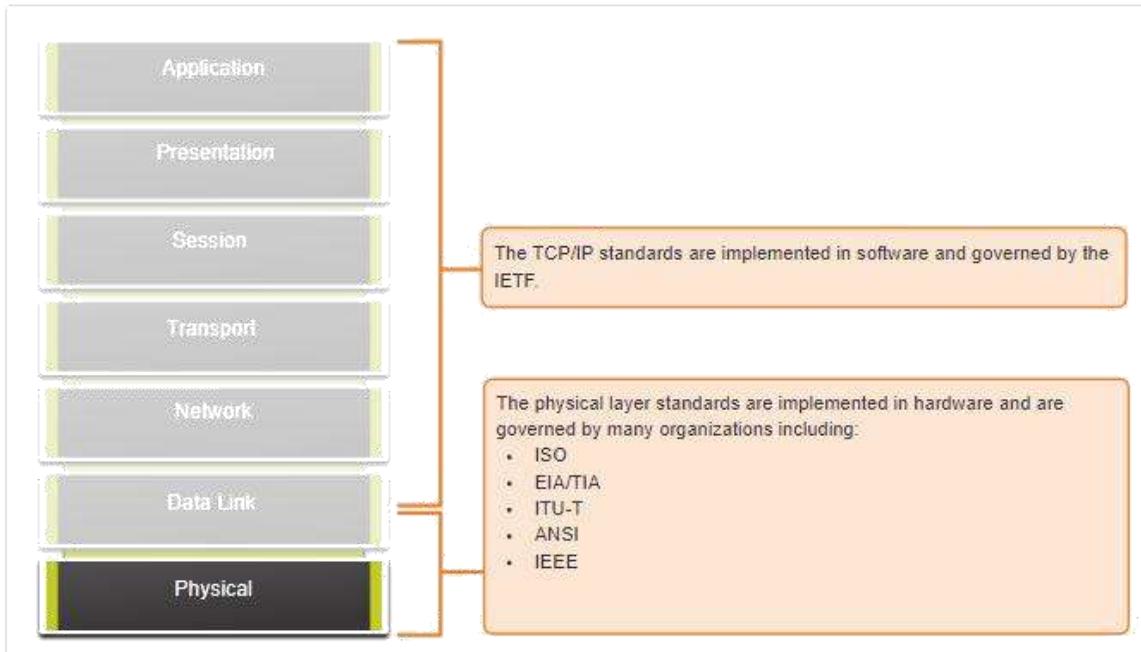
Protokol dan operasi **Upper Layer** dilakukan dengan menggunakan software yang dirancang oleh software engineer dan ilmuwan komputer. Layanan dan protokol dalam rangkaian TCP / IP ditentukan oleh Internet Engineering Task Force (IETF).

Physical Layer terdiri dari sirkuit elektronik, media, dan konektor yang dikembangkan oleh para ilmuwan komputer. Oleh karena itu, adalah tepat bahwa standar yang mengatur perangkat keras ini ditentukan oleh organisasi teknik kelistrikan dan komunikasi yang relevan.

Ada banyak organisasi internasional dan nasional yang berbeda, organisasi pemerintah yang mengatur, dan perusahaan swasta yang terlibat dalam menetapkan dan memelihara standar Physical Layer. Misalnya, standar perangkat keras Physical Layer, media, pengkodean, dan pensinyalan ditentukan dan diatur oleh organisasi standar berikut:

- International Organization for Standardization (ISO)
- Telecommunications Industry Association/Electronic Industries Association (TIA/EIA)
- International Telecommunication Union (ITU)
- American National Standards Institute (ANSI)
- Institute of Electrical and Electronics Engineers (IEEE)
- National telecommunications regulatory authorities including the Federal Communication Commission (FCC) in the USA and the European Telecommunications Standards Institute (ETSI)

Selain itu, sering kali ada grup standar kabel regional seperti CSA (Canadian Standards Association), CENELEC (European Committee for Electrotechnical Standardization), dan JSA / JIS (Japanese Standards Association), yang mengembangkan spesifikasi lokal.



Standar **Physical Layer** membahas tiga bidang fungsional:

- Komponen Fisik
- Encoding
- Signaling

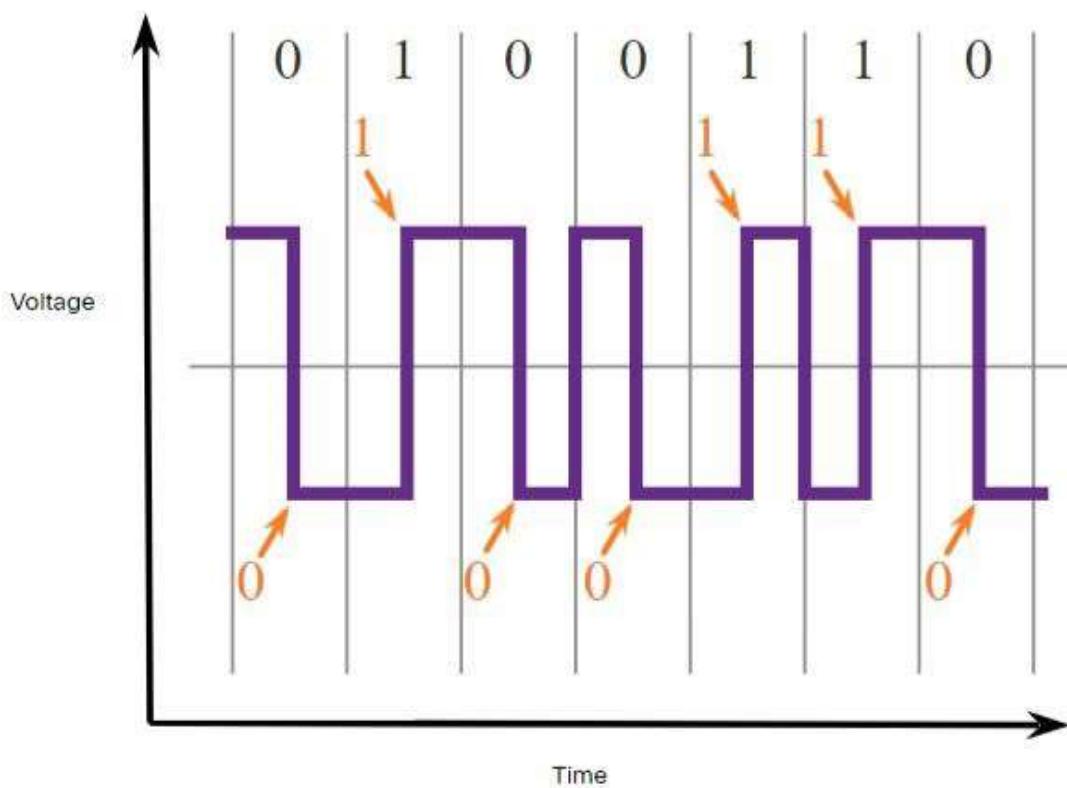
Komponen Fisik

Komponen fisik adalah perangkat keras elektronik, media, dan konektor lain yang mengirimkan sinyal yang mewakili bit. Komponen perangkat keras seperti beberapa **NIC**, **Interface** dan **konektor**, **bahan kabel**, dan **desain kabel** semuanya ditentukan dalam standar yang terkait dengan Physical Layer. Berbagai port dan Interface pada router Cisco 1941 juga merupakan contoh komponen fisik dengan konektor dan beberapa pin out tertentu yang dihasilkan dari standar.

Pengkodean/Encoding

Encoding atau line encoding adalah metode untuk mengubah aliran bit data menjadi “kode” yang telah ditentukan. Kode adalah pengelompokan bit yang digunakan untuk memberikan pola yang dapat diprediksi yang dapat dikenali oleh pengirim dan penerima. Dengan kata lain, encoding adalah metode atau pola yang digunakan untuk merepresentasikan informasi digital, ini mirip dengan bagaimana kode Morse menyandikan pesan menggunakan serangkaian titik dan garis.

Misalnya, pengkodean Manchester merepresentasikan 0 bit oleh transisi tegangan tinggi ke rendah, dan 1 bit direpresentasikan sebagai transisi tegangan rendah ke tinggi. Contoh encoding Manchester diilustrasikan pada gambar. Transisi terjadi di tengah setiap periode bit. Jenis pengkodean ini digunakan di Ethernet 10 Mbps. Kecepatan data yang lebih cepat membutuhkan pengkodean yang lebih kompleks. Encoding Manchester digunakan dalam standar Ethernet yang lebih lama seperti **10BASE-T**. Ethernet **100BASE-TX** menggunakan pengkodean 4B / 5B dan 1000BASE-T menggunakan pengkodean 8B / 10B.



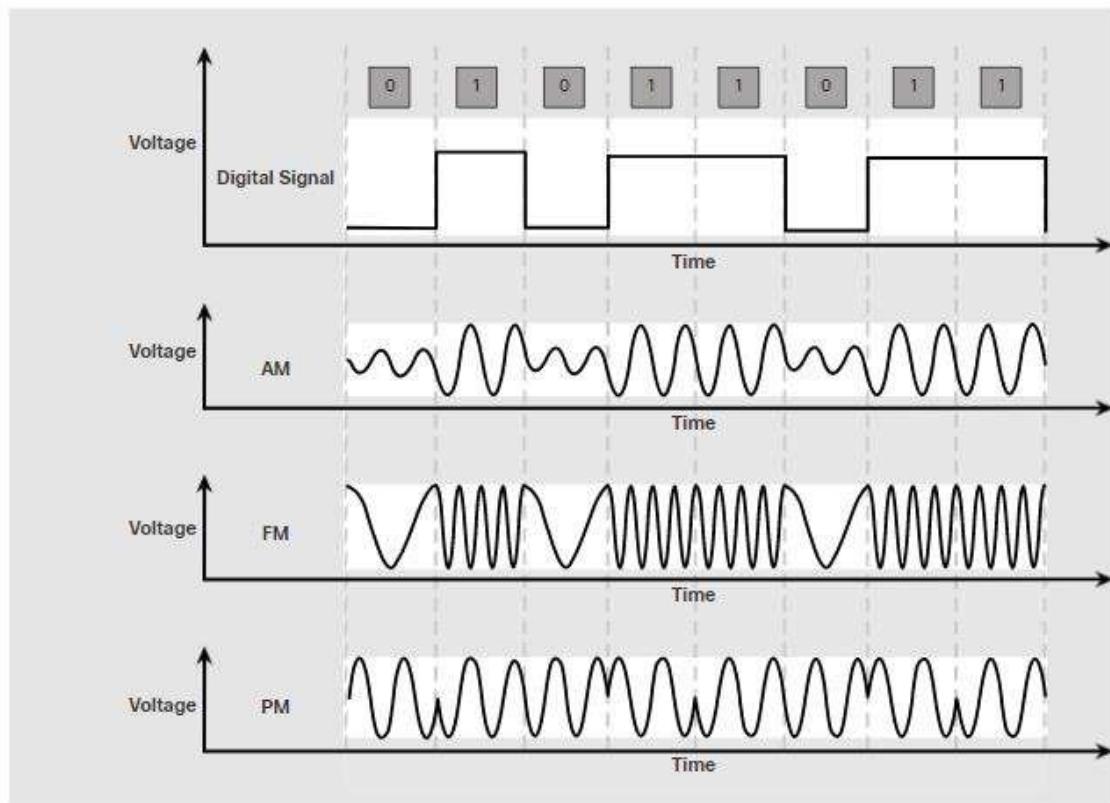
Pensinyalan/Signaling

Physical Layer harus menghasilkan sinyal listrik, optik, atau nirkabel yang mewakili angka “1” dan “0” di media. Cara bit direpresentasikan disebut metode Signaling. Standar Physical Layer harus menentukan jenis sinyal yang mewakili “1” dan jenis sinyal yang mewakili “0”. Ini bisa sesederhana perubahan level sinyal listrik atau **pulse optik**. Misalnya, **long pulse** mungkin mewakili 1 sedangkan **short pulse** mungkin mewakili 0.

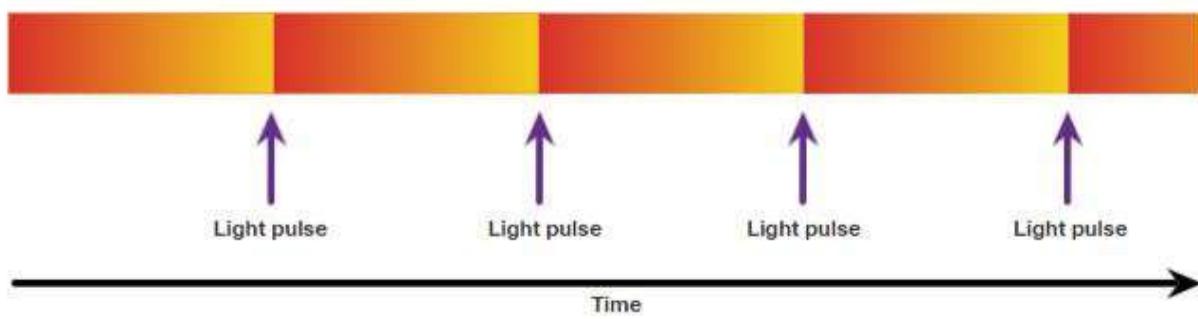
Ini mirip dengan metode **Signaling** yang digunakan dalam kode Morse, yang mungkin menggunakan serangkaian nada nyala, lampu, atau klik untuk mengirim teks melalui kabel telepon atau antar kapal di laut.

Angka-angka itu menunjukkan sinyal

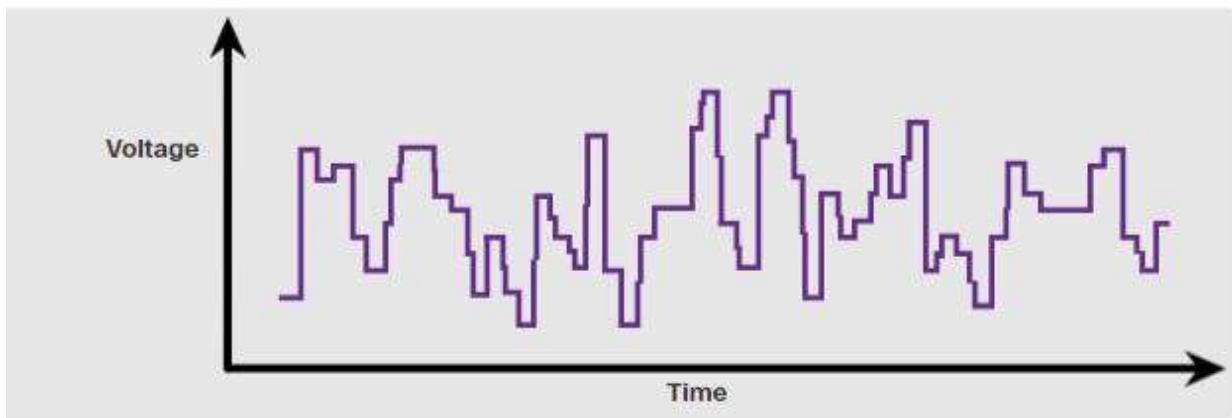
Sinyal Microwave Pada Nirkabel



Light Pulse Pada Kabel Serat Optik



Electrical Signals Pada Copper Cable



Bandwidth

Media fisik yang berbeda mendukung transfer bit dengan kecepatan berbeda. Transfer data biasanya dibahas dalam istilah **bandwidth**. **Bandwidth** adalah kapasitas di mana suatu media dapat membawa data. Bandwidth digital mengukur jumlah data yang dapat mengalir dari satu tempat ke tempat lain dalam jumlah waktu tertentu. **Bandwidth** biasanya diukur dalam kilobit per detik (kbps), megabit per detik (Mbps), atau gigabit per detik (Gbps). **Bandwidth** terkadang dianggap sebagai kecepatan bit berjalan, namun ini tidak akurat. Misalnya, pada Ethernet 10Mbps dan 100Mbps, bit-bit tersebut dikirim dengan kecepatan listrik. Perbedaannya adalah jumlah bit yang ditransmisikan per detik.

Kombinasi berbagai faktor menentukan bandwidth praktis jaringan:

- Properti media fisik
- Teknologi yang dipilih untuk memberi sinyal dan mendekripsi sinyal jaringan

Sifat media fisik, teknologi saat ini, dan hukum fisika semuanya berperan dalam menentukan **bandwidth** yang tersedia.

Tabel menunjukkan satuan ukuran yang umum digunakan untuk **bandwidth**

| Unit of Bandwidth | Abbreviation | Equivalence |
|---------------------|--------------|---------------------------------------|
| Bits per second | bps | 1 bps = fundamental unit of bandwidth |
| Kilobits per second | Kbps | 1 Kbps = 1,000 bps = 103 bps |

| | | |
|---------------------|------|---|
| Megabits per second | Mbps | $1 \text{ Mbps} = 1,000,000 \text{ bps} = 106 \text{ bps}$ |
| Gigabits per second | Gbps | $1 \text{ Gbps} = 1,000,000,000 \text{ bps} = 109 \text{ bps}$ |
| Terabits per second | Tbps | $1 \text{ Tbps} = 1,000,000,000,000 \text{ bps} = 1012 \text{ bps}$ |

Terminologi Bandwidth

Istilah yang digunakan untuk mengukur kualitas bandwidth antara lain:

- Latency
- Throughput
- Goodput

Latensi

Latency mengacu pada jumlah waktu, termasuk penundaan, untuk perjalanan data dari satu titik ke titik lainnya.

Di internetwork, atau jaringan dengan banyak segmen, throughput tidak bisa lebih cepat daripada link paling lambat di jalur dari sumber ke tujuan. Meskipun semua, atau sebagian besar, segmen memiliki bandwidth tinggi, ini hanya akan mengambil satu segmen di jalur dengan throughput rendah untuk membuat hambatan dalam throughput seluruh jaringan.

Throughput

Throughput adalah ukuran transfer bit melintasi media selama periode waktu tertentu.

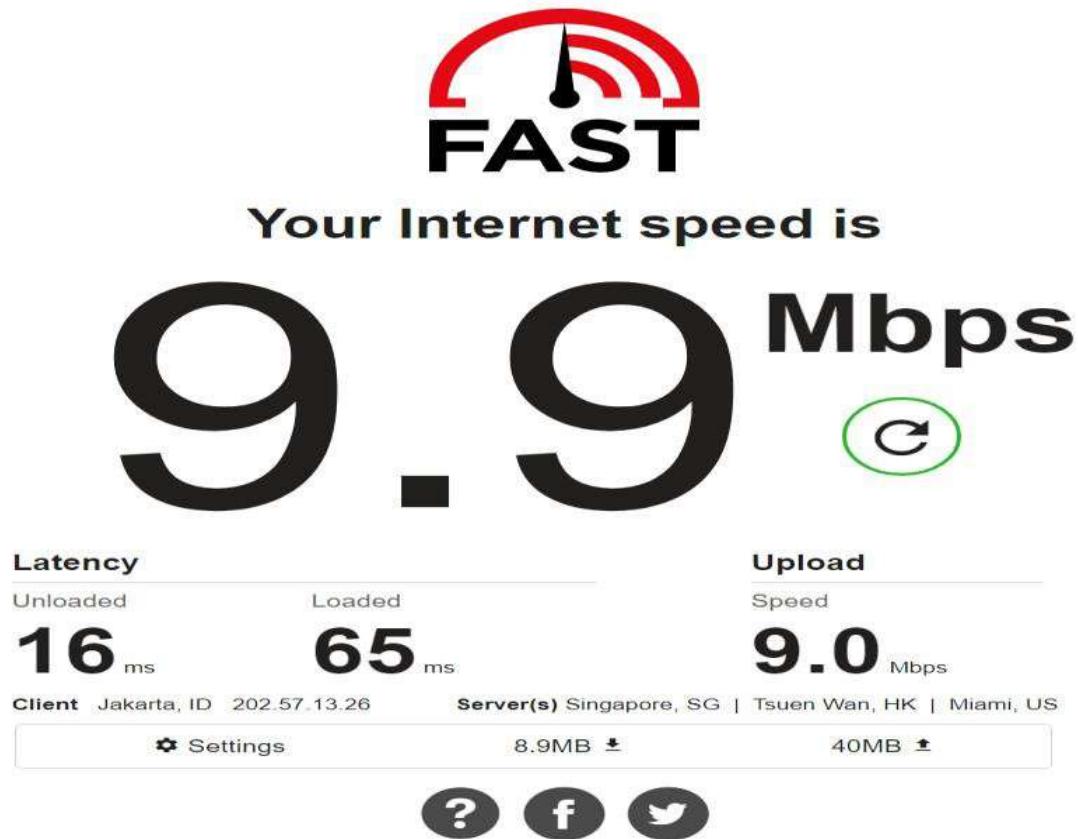
Karena sejumlah faktor, throughput biasanya tidak cocok dengan bandwidth yang ditentukan dalam implementasi Physical Layer. Throughput biasanya lebih rendah dari bandwidth. Ada banyak faktor yang mempengaruhi throughput:

- Jumlah Traffic
- Jenis Traffic
- Latency yang dibuat oleh jumlah perangkat jaringan yang ditemukan antara sumber dan tujuan

Ada banyak tes kecepatan online yang dapat mengungkapkan throughput koneksi internet. Gambar tersebut memberikan hasil sampel dari uji kecepatan.

Goodput

Ada pengukuran ketiga untuk menilai transfer data yang dapat digunakan; itu dikenal sebagai goodput. Goodput adalah ukuran data yang dapat digunakan yang ditransfer selama periode waktu tertentu. Goodput adalah throughput dikurangi overhead Traffic untuk membuat sesi, ucapan terima kasih, enkapsulasi, dan bit yang ditransmisikan ulang. Goodput selalu lebih rendah daripada throughput, yang umumnya lebih rendah dari bandwidth.



Kabel Tembaga

Kabel tembaga adalah jenis kabel yang paling umum digunakan dalam jaringan saat ini. Faktanya, kabel tembaga bukan hanya satu jenis kabel. Ada tiga jenis kabel tembaga yang masing-masing digunakan dalam situasi tertentu.

Karakteristik Kabel Tembaga

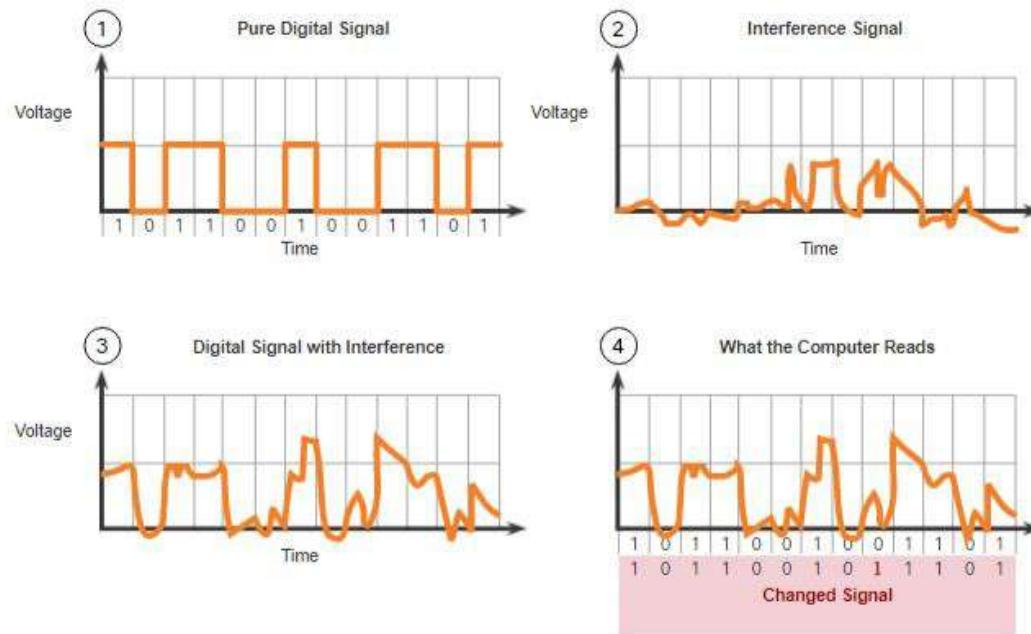
Jaringan menggunakan media tembaga karena murah, mudah dipasang, dan resistansinya rendah terhadap arus listrik. Namun, media tembaga dibatasi oleh jarak dan gangguan sinyal.

Data ditransmisikan pada kabel tembaga sebagai **electricity Pulse**. Detektor di **Network Interface** perangkat tujuan harus menerima sinyal yang dapat berhasil diterjemahkan agar cocok dengan sinyal yang dikirim. Namun, semakin jauh perjalanan sinyal, semakin memburuk. Ini disebut sebagai pelemahan sinyal. Untuk alasan ini, semua media tembaga harus mengikuti batasan jarak yang ketat seperti yang ditentukan oleh standar pedoman.

Nilai waktu dan tegangan **electricity Pulse** juga rentan terhadap gangguan dari dua sumber:

- **Electromagnetic interference (EMI)** Atau **radio frequency interference (RFI)** – Sinyal EMI dan RFI dapat *mendistorsi* dan merusak sinyal data yang dibawa oleh media tembaga. Sumber potensial EMI dan RFI termasuk gelombang radio dan perangkat elektromagnetik, seperti lampu fluoresen atau motor listrik.
- **Crosstalk** – **Crosstalk** adalah gangguan yang disebabkan oleh medan listrik atau magnet dari suatu sinyal pada satu kabel ke sinyal di kabel yang berdekatan. Di sirkuit telepon, **crosstalk** dapat mengakibatkan mendengar bagian percakapan suara lain dari sirkuit yang berdekatan. Secara khusus, ketika arus listrik mengalir melalui kabel, itu menciptakan medan magnet melingkar kecil di sekitar kabel, yang dapat diambil oleh kabel yang berdekatan.

Gambar tersebut menunjukkan bagaimana transmisi data dapat dipengaruhi oleh interferensi.



1. Sinyal digital murni ditransmisikan
2. Pada medium, ada sinyal interferensi
3. Sinyal digital rusak oleh sinyal interferensi.
4. Komputer penerima membaca sinyal yang diubah. Perhatikan bahwa 0 bit sekarang diartikan sebagai 1 bit.

Untuk mengatasi efek negatif dari **EMI** dan **RFI**, beberapa jenis kabel tembaga dibungkus dengan pelindung logam dan memerlukan koneksi arde yang tepat.

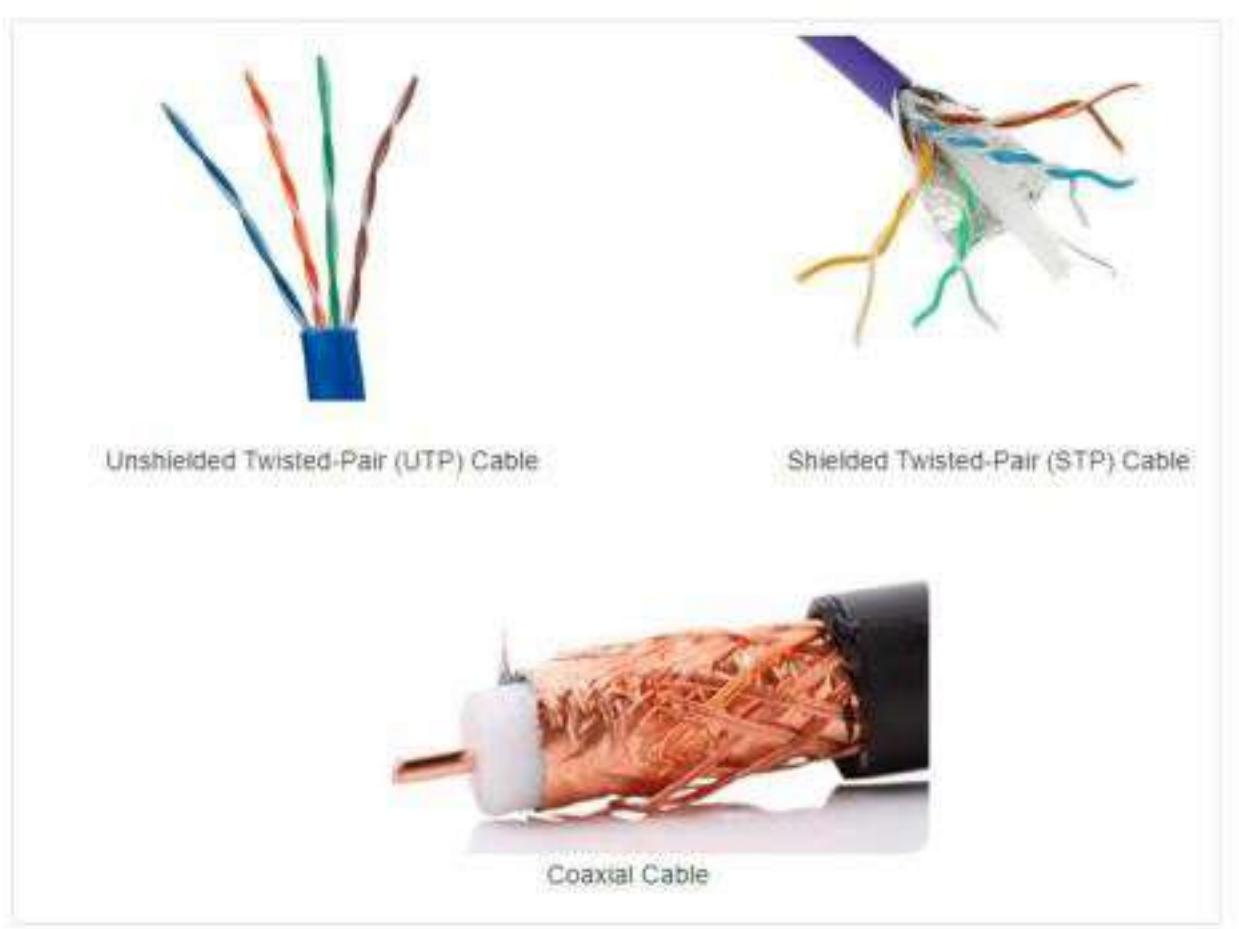
Untuk mengatasi efek negatif crosstalk, beberapa jenis kabel tembaga memiliki pasangan kabel sirkuit berlawanan yang *dipilih* bersama, yang secara efektif membantalkan **crosstalk**.

Kerentanan kabel tembaga terhadap *Noise* elektronik juga dapat dibatasi dengan menggunakan rekomendasi berikut:

- Memilih jenis atau kategori kabel yang paling sesuai dengan lingkungan jaringan tertentu
- Merancang infrastruktur kabel untuk menghindari sumber gangguan yang diketahui dan potensial dalam struktur bangunan
- Menggunakan teknik pemasangan kabel yang mencakup penanganan dan *terminasi* kabel yang benar

Jenis Kabel Tembaga

Ada tiga jenis utama media tembaga yang digunakan dalam jaringan.

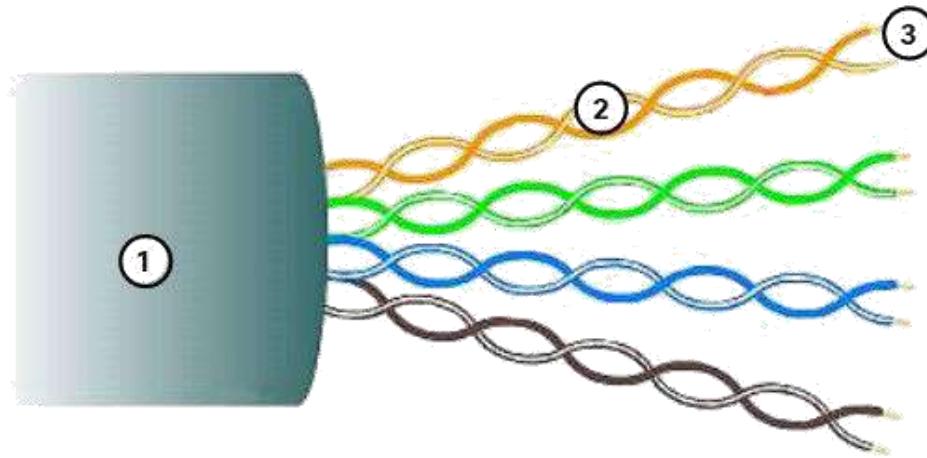


Unshielded Twisted-pair (UTP)

Kabel **unshielded twisted-pair (UTP)** adalah media jaringan yang paling umum. Kabel UTP, diakhiri dengan konektor RJ-45, digunakan untuk menghubungkan host jaringan dengan perangkat jaringan perantara, seperti **Switch** dan **router**.

Pada LAN, kabel **UTP** terdiri dari empat pasang kabel berkode warna yang telah dipilih menjadi satu dan kemudian dibungkus dalam selubung plastik fleksibel yang melindungi dari kerusakan fisik ringan. Pemutaran kabel membantu melindungi dari gangguan sinyal dari kabel lain.

Seperti yang terlihat pada gambar, kode warna mengidentifikasi pasangan dan kabel individu dan membantu dalam terminasi kabel.



Angka-angka pada gambar menunjukkan beberapa karakteristik utama dari kabel **UTP**:

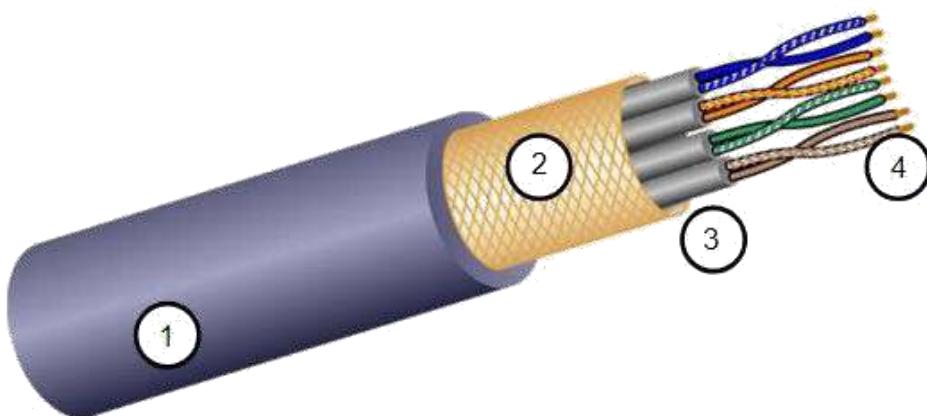
1. **Outer Jacket** melindungi kabel tembaga dari kerusakan fisik.
2. **Twisted-pair** melindungi sinyal dari gangguan.
3. **Isolasi plastik** berkode warna secara elektrik mengisolasi kabel dari satu sama lain dan mengidentifikasi setiap pasangan.

Shielded Twisted Pair(STP)

Shielded twisted-pair (STP) memberikan perlindungan noise yang lebih baik daripada kabel UTP. Namun, dibandingkan dengan kabel UTP, kabel STP jauh lebih mahal dan sulit untuk dipasang. Seperti kabel UTP, STP menggunakan konektor RJ-45.

Kabel **STP** menggabungkan teknik pelindung untuk melawan **EMI** dan **RFI**, dan memutar kawat untuk melawan **crosstalk**. Untuk mendapatkan manfaat penuh dari pelindung, kabel STP diakhiri dengan konektor data **STP Shielded** khusus. Jika kabel tidak *ground* dengan benar, pelindung dapat bertindak sebagai antena dan menangkap sinyal yang tidak diinginkan.

Kabel **STP** yang ditampilkan menggunakan empat pasang kabel, masing-masing dibungkus dengan **Shield Foil**, yang kemudian dibungkus dengan **Overall metalic braid** atau **foil**.



Angka-angka dalam gambar menunjukkan beberapa fitur utama kabel STP:

1. Outer Jacket
2. Braided
3. Foil Shields
4. Twisted Pair

Kabel Coaxial

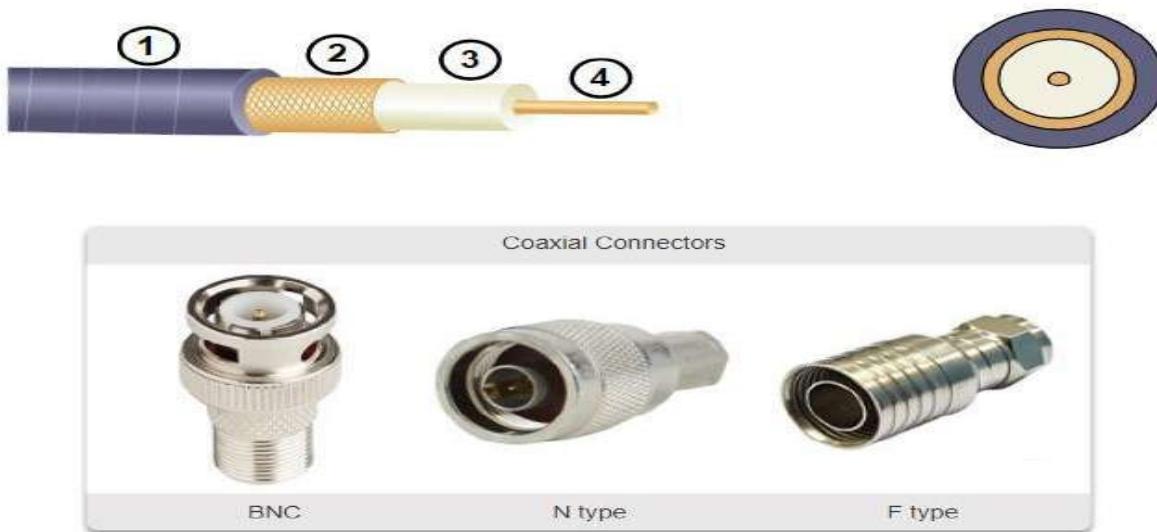
Kabel **Coaxial**, atau singkatnya **coax**, mendapatkan namanya dari fakta bahwa ada dua konduktor yang berbagi sumbu yang sama. Seperti yang ditunjukkan pada gambar, kabel koaksial terdiri dari:

- Konduktor tembaga digunakan untuk mengirimkan sinyal elektronik.
- Lapisan isolasi plastik fleksibel mengelilingi konduktor tembaga.
- Bahan insulasi dikelilingi oleh anyaman anyaman tembaga, atau lembaran logam, yang berfungsi sebagai kabel kedua di sirkuit dan sebagai pelindung konduktor dalam. Lapisan atau perisai kedua ini juga mengurangi jumlah gangguan elektromagnetik luar.
- Seluruh kabel ditutup dengan jaket kabel untuk mencegah kerusakan fisik ringan.

Ada berbagai jenis konektor yang digunakan dengan kabel **coax**. Konektor **Bayonet Neill – Concelman (BNC)**, **tipe N**, dan **tipe F** ditunjukkan pada gambar.

Meskipun kabel **UTP** pada dasarnya telah menggantikan kabel **Coaxial** dalam instalasi **Ethernet** modern, desain kabel **Coaxial** digunakan dalam situasi berikut:

- **Wireless installations** – **Kabel Coaxial** memasang antena ke perangkat nirkabel. Kabel koaksial membawa energi **Radio Frequency (RF)** antara antena dan peralatan radio.
- **Cable internet installations** – Penyedia layanan kabel menyediakan koneksi internet kepada pelanggan mereka dengan mengganti bagian kabel koaksial dan elemen pendukung amplifikasi dengan kabel **serat optik**. Namun, kabel di dalam tempat pelanggan masih berupa kabel **coax**.



Angka-angka pada gambar menunjukkan beberapa fitur utama kabel Coaxial:

1. Outer Jacket
2. Braided copper shielded
3. Plastic Insulation
4. Copper Conductor

Kabel UTP

Pada Materi sebelumnya, Anda telah mempelajari sedikit tentang pemasangan kabel tembaga **unshielded twisted-pair (UTP)**. Karena kabel **UTP** adalah standar untuk digunakan di **LAN**, Materi ini menjelaskan secara rinci tentang kelebihan dan keterbatasannya, dan apa yang dapat dilakukan untuk menghindari masalah.

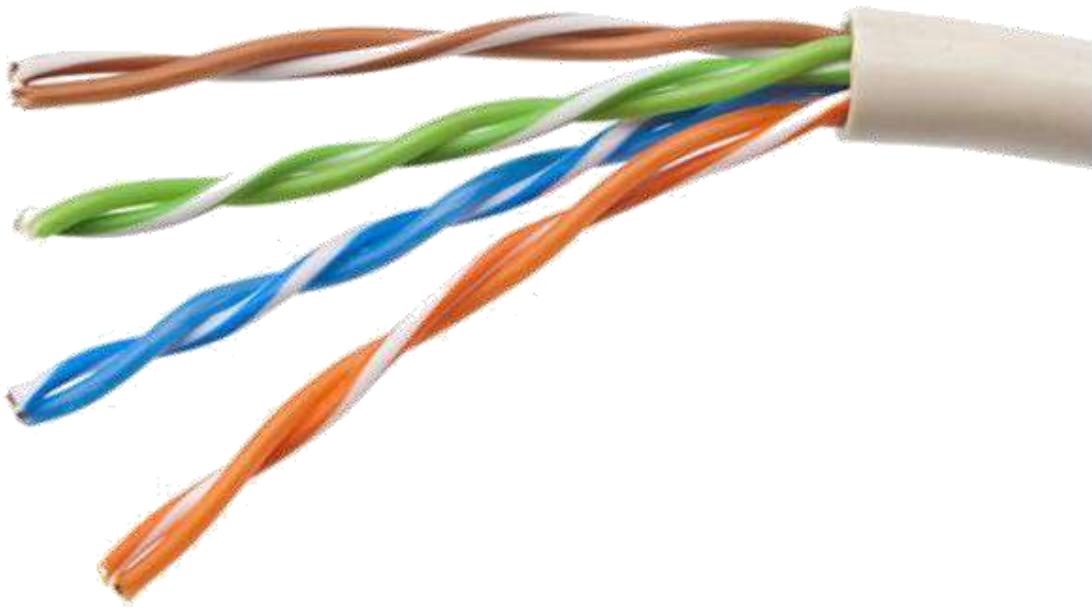
Properti Kabel UTP

Saat digunakan sebagai media jaringan, kabel **UTP** terdiri dari empat pasang kabel tembaga berkode warna yang telah dipilih menjadi satu dan kemudian dibungkus dalam selubung plastik fleksibel. Ukurannya yang kecil dapat menguntungkan selama pemasangan.

Kabel **UTP** tidak menggunakan pelindung untuk melawan efek **EMI** dan **RFI**. Sebaliknya, desainer kabel telah menemukan cara lain untuk membatasi efek negatif **crosstalk**:

- **Cancellation**— *Design engineer* sekarang memasangkan kabel di sirkuit. Ketika dua kabel dalam rangkaian listrik ditempatkan berdekatan, medan magnetnya berlawanan satu sama lain. Oleh karena itu, dua medan magnet *Cancellation/Membatalkan* satu sama lain dan juga *Cancellation* sinyal **EMI** dan **RFI** di luar.
- **Memvariasikan jumlah lilitan per pasangan kabel** – Untuk lebih meningkatkan efek *Cancellation* kabel sirkuit yang dipasangkan, perancang memvariasikan jumlah lilitan setiap pasangan kabel dalam sebuah kabel. Kabel UTP harus mengikuti spesifikasi yang tepat yang mengatur berapa banyak lilitan atau jalinan yang diizinkan per meter (3,28 kaki) kabel. Perhatikan pada gambar bahwa pasangan jingga / jingga putih bengkok kurang dari pasangan biru / biru putih. Setiap pasangan berwarna dipelintir beberapa kali.

Kabel **UTP** hanya mengandalkan efek *Cancellation* yang dihasilkan oleh pasangan kabel bengkok untuk membatasi degradasi sinyal dan secara efektif menyediakan pelindung diri untuk pasangan kabel dalam media jaringan.



Standar dan Konektor Kabel UTP

Kabel **UTP** sesuai dengan standar yang ditetapkan bersama oleh TIA / EIA. Secara khusus, TIA / EIA-568 menetapkan standar kabel komersial untuk instalasi **LAN** dan standar yang paling umum digunakan di lingkungan kabel **LAN**. Beberapa elemen yang didefinisikan adalah sebagai berikut:

- Jenis kabel
- Panjang kabel
- Konektor
- Pemutusan kabel
- Metode pengujian kabel

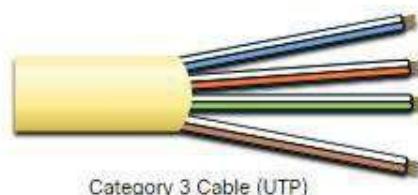
Karakteristik kelistrikan kabel tembaga ditentukan oleh Institute of Electrical and Electronics Engineers (IEEE). IEEE menilai pemasangan kabel UTP menurut kinerjanya. Kabel ditempatkan ke dalam kategori berdasarkan kemampuannya untuk membawa kecepatan bandwidth yang lebih tinggi. Misalnya, kabel **Category 5** biasanya digunakan dalam instalasi **Fast Ethernet** 100BASE-TX. Kategori lain termasuk kabel Enhanced **Category 5**, **Category 6**, dan **Category 6a**.

Kabel dalam **Category** yang lebih tinggi dirancang dan dibuat untuk mendukung kecepatan data yang lebih tinggi. Karena teknologi **Ethernet** kecepatan **gigabit** baru sedang dikembangkan dan diadopsi, **Category 5e** sekarang menjadi jenis kabel yang minimal dapat diterima, dengan **Category 6** menjadi jenis yang direkomendasikan untuk instalasi gedung baru.

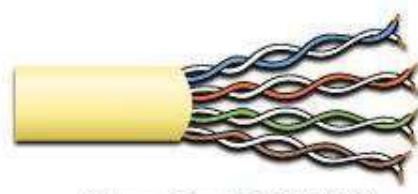
Gambar tersebut menunjukkan tiga **Category** kabel UTP:

- **Category 3** awalnya digunakan untuk komunikasi suara melalui jalur suara, tetapi kemudian digunakan untuk transmisi data.
- **Category 5** dan **5e** digunakan untuk transmisi data. Kategori 5 mendukung 100Mbps dan Kategori 5e mendukung 1000 Mbps
- **Category 6** memiliki pemisah tambahan di antara setiap pasangan kabel untuk mendukung kecepatan yang lebih tinggi. **Category 6** mendukung hingga 10 Gbps.
- **Category 7** juga mendukung 10 Gbps.
- **Category 8** mendukung 40 Gbps.

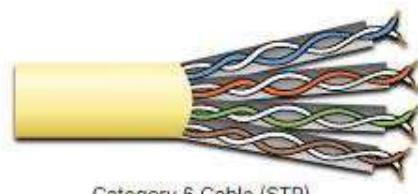
Beberapa pabrikan membuat kabel yang melebihi spesifikasi TIA / EIA **Category 6a** dan menyebutnya sebagai **Category 7**.



Category 3 Cable (UTP)



Category 5 and 5e Cable (UTP)



Category 6 Cable (STP)

UTP RJ-45 Plugs

Kabel **UTP** biasanya diakhiri dengan konektor **RJ-45**. Standar TIA / EIA-568 menjelaskan kode warna kabel ke penetapan pin (pinouts) untuk kabel **Ethernet**.

Seperti yang ditunjukkan pada gambar, konektor **RJ-45** adalah komponen *male*, yang dikerutkan di ujung kabel.



UTP RJ-45 Sockets

Soket, yang ditunjukkan pada gambar, adalah komponen *female* dari perangkat jaringan, dinding, outlet partisi bilik, atau panel patch. Ketika diakhiri dengan tidak benar, setiap kabel berpotensi menjadi sumber penurunan kinerja physical layer.



Poorly Terminated UTP Cable

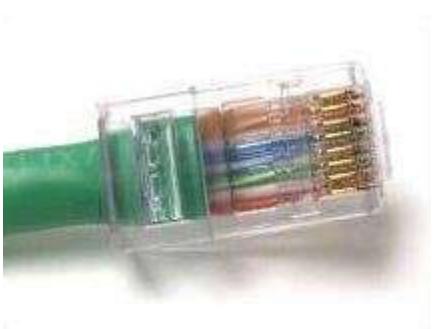
Gambar ini menunjukkan contoh **Terminated UTP Cable** dengan buruk. Konektor yang buruk ini memiliki kabel yang terbuka, tidak terpilih, dan tidak seluruhnya tertutup oleh sarungnya.



Properly Terminated UTP Cable

Gambar berikutnya menunjukkan **Terminated UTP Cable** dengan benar. Ini adalah konektor yang baik dengan kabel yang dilepas hanya sejauh yang diperlukan untuk memasang konektor.

Terminated UTP Cable dengan benar menunjukkan jaket kabel yang memanjang ke konektor **RJ45** cukup untuk dikerutkan dengan kencang dengan kedelapan kabel mencapai ujung konektor



Catatan : **Terminated UTP Cable** yang tidak tepat dapat mempengaruhi kinerja transmisi.

Kabel UTP Straight-through dan Crossover

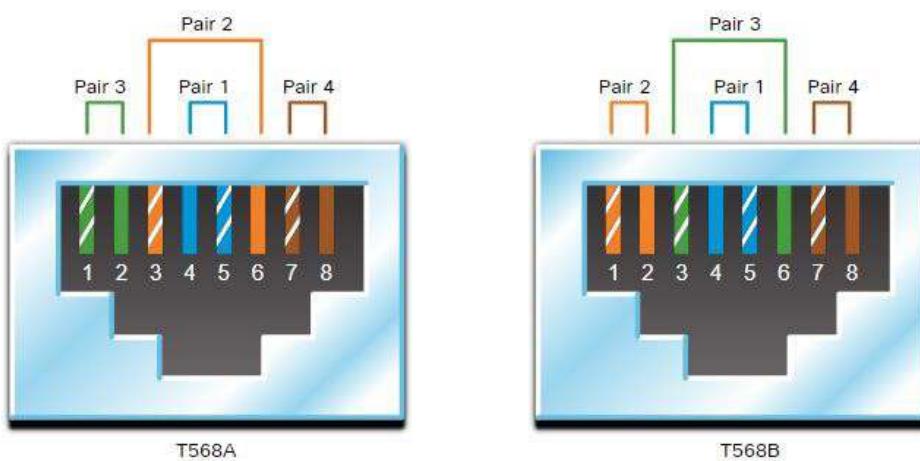
Situasi yang berbeda mungkin memerlukan kabel **UTP** untuk disambungkan sesuai dengan konvensi kabel yang berbeda. Ini berarti bahwa setiap satu kabel harus dihubungkan dalam urutan yang berbeda ke set pin yang berbeda di konektor **RJ-45**.

Berikut ini adalah jenis kabel utama yang diperoleh dengan menggunakan konvensi perkabelan tertentu:

- **Ethernet Straight-through** – Jenis kabel jaringan yang paling umum. Ini biasanya digunakan untuk menghubungkan **host** ke **switch** dan **Switch** ke **router**.
- **Ethernet Crossover** – Kabel yang digunakan untuk menghubungkan perangkat serupa. Misalnya, untuk menghubungkan **switch** ke **switch**, **host** ke **host**, atau **router** ke **router**. Namun, kabel crossover sekarang dianggap warisan karena **NIC** menggunakan **medium-dependent interface crossover (auto-MDIX)** untuk secara otomatis mendeteksi jenis kabel dan membuat koneksi internal.

Catatan : Jenis kabel lainnya adalah kabel *rollover*, yang merupakan hak milik Cisco. Ini digunakan untuk menghubungkan workstation ke router atau port console switch.

Menggunakan **crossover** atau kabel **straight-through** secara tidak benar antar perangkat mungkin tidak merusak perangkat, tetapi koneksi dan komunikasi antar perangkat tidak akan berlangsung. Ini adalah kesalahan umum dan memerlukan verifikasi bahwa koneksi perangkat sudah benar harus menjadi tindakan pemecahan masalah pertama jika koneksi tidak tercapai.



Tabel menunjukkan jenis kabel UTP, standar terkait, dan aplikasi tipikal kabel ini.

Jenis dan Standar Kabel

| Jenis Kabel | Standar | Aplikasi |
|---------------------------|--|---|
| Ethernet Straight-through | Kedua ujung T568A atau kedua ujung T568B | Menghubungkan host jaringan ke perangkat jaringan seperti switch atau hub |
| Ethernet Crossover | Satu ujung T568A, ujung lainnya T568B | Menghubungkan dua host jaringan Menghubungkan dua perangkat perantara jaringan (beralih ke switch atau router ke router) |
| Rollover | Kepemilikan Cisco | Menghubungkan port serial stasiun kerja ke port console router, menggunakan adaptor |

Kabel Fiber Optik

Seperti yang telah Anda pelajari, kabel **Fiber Optik** adalah jenis kabel lain yang digunakan dalam jaringan. Karena mahal, ini tidak umum digunakan pada berbagai jenis kabel tembaga. Tetapi kabel **Fiber Optik** memiliki sifat tertentu yang menjadikannya pilihan terbaik dalam situasi tertentu, yang akan Anda temukan dalam Materi ini.

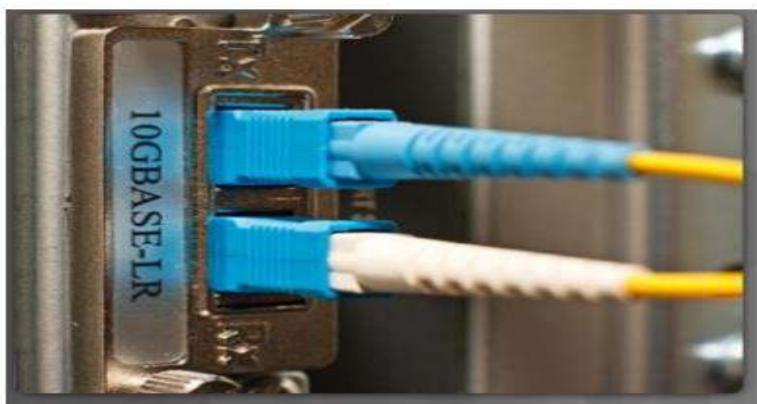
Properti Kabel Fiber-Optic

Kabel Fiber Optik mentransmisikan data dalam jarak yang lebih jauh dan pada bandwidth yang lebih tinggi daripada media jaringan lainnya. Tidak seperti kabel tembaga, kabel Fiber Optik dapat mengirimkan sinyal dengan atenuasi yang lebih sedikit dan sepenuhnya kebal terhadap **EMI** dan **RFI**. Fiber Optik biasanya digunakan untuk menghubungkan perangkat jaringan.

Fiber Optik adalah untaian/rajutan kaca yang sangat murni, sangat tipis dan fleksibel, tidak lebih besar dari sehelai rambut manusia. Bit dikodekan pada serat sebagai impuls cahaya. Kabel **Fiber Optik**

bertindak sebagai pemandu gelombang, atau “light pipe”, untuk mengirimkan cahaya di antara kedua ujungnya dengan **loss signal** yang minimal.

Sebagai analogi, pertimbangkan gulungan handuk kertas kosong dengan bagian dalamnya dilapisi seperti cermin. Panjangnya seribu meter, dan penunjuk laser kecil digunakan untuk mengirim sinyal kode Morse dengan kecepatan cahaya. Pada dasarnya begitulah cara kerja kabel serat optik, hanya saja diameternya lebih kecil dan menggunakan teknologi cahaya yang canggih.



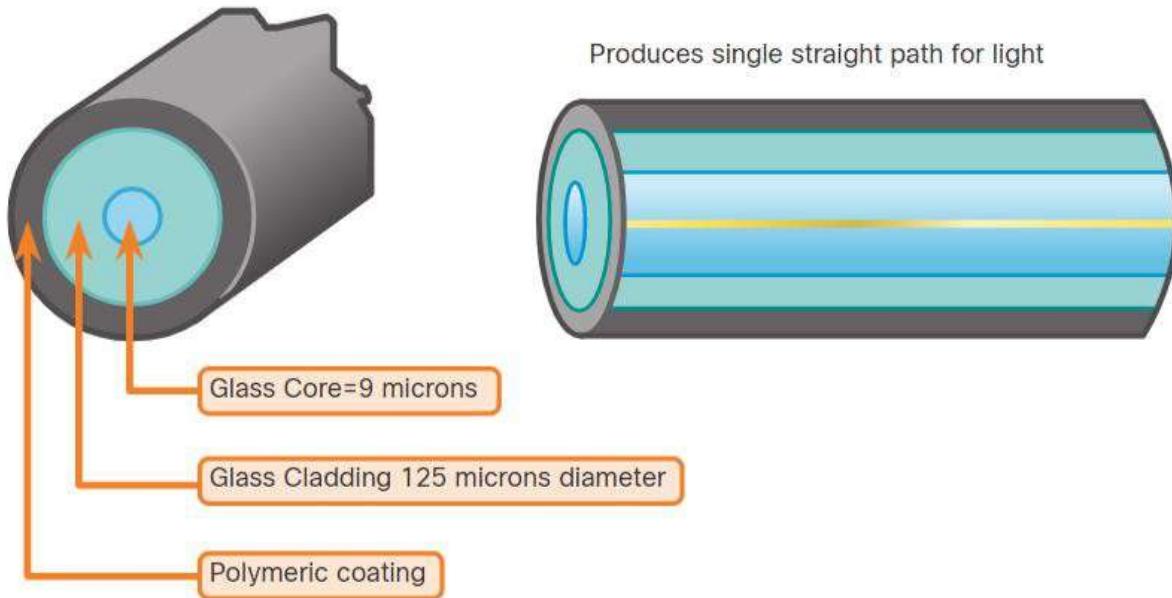
Jenis Media Fiber

Kabel Fiber Optik secara luas diklasifikasikan menjadi dua jenis:

- Single-mode fiber (SMF)
- Multimode fiber (MMF)

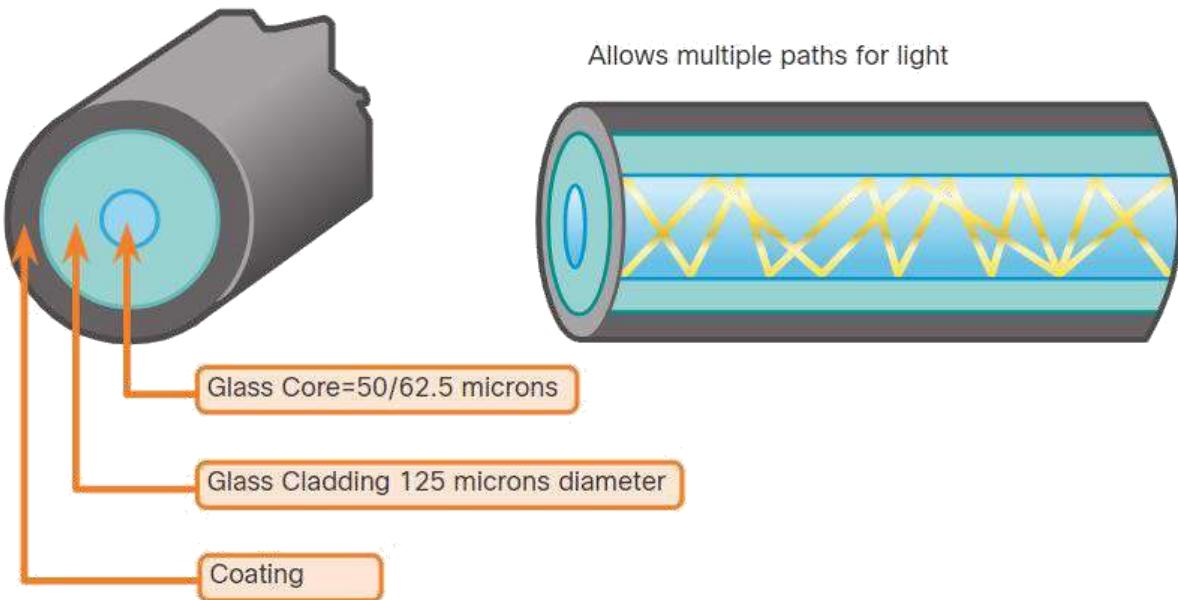
Single-mode fiber (SMF)

SMF terdiri dari *core* yang sangat kecil dan menggunakan teknologi laser yang mahal untuk mengirimkan satu sinar cahaya, seperti yang ditunjukkan pada gambar. **SMF** populer dalam situasi jarak jauh yang mencakup ratusan kilometer, seperti yang diperlukan dalam aplikasi telepon jarak jauh dan TV kabel.



Multimode fiber

MMF terdiri dari *core* yang lebih besar dan menggunakan pemancar LED untuk mengirim **pulse-light**. Secara khusus, cahaya dari LED memasuki **multimode fiber** pada sudut yang berbeda, seperti yang ditunjukkan pada gambar. Populer di LAN karena dapat didukung oleh LED berbiaya rendah. Ini menyediakan bandwidth hingga 10 Gb / s melalui panjang link hingga 550 meter.



Salah satu perbedaan yang disorot antara **MMF** dan **SMF** adalah jumlah *dispersi*. *Dispersi* mengacu pada penyebaran **light pulse** dari waktu ke waktu. Peningkatan dispersi berarti peningkatan kehilangan kekuatan sinyal. **MMF** memiliki dispersi yang lebih besar dari **SMF**. Itulah mengapa **MMF** hanya dapat melakukan perjalanan hingga 500 meter sebelum kehilangan sinyal.

Penggunaan Kabel Fiber Optic

Pengkabelan **Fiber Optic** sekarang digunakan di empat jenis industri:

- **Enterprise Networks** – Digunakan untuk aplikasi pemasangan kabel *backbone* dan perangkat infrastruktur interkoneksi
- **Fiber-to-the-Home (FTTH)** – Digunakan untuk menyediakan layanan broadband yang selalu aktif ke rumah dan bisnis kecil
- **Long-Haul Networks** – Digunakan oleh penyedia layanan untuk menghubungkan negara dan kota
- **Submarine Cable Networks** – Digunakan untuk menyediakan solusi berkapasitas tinggi dan berkecepatan tinggi yang andal yang mampu bertahan di lingkungan bawah laut yang keras hingga jarak lintas samudra. Cari di internet untuk “submarine cables telegeography map” untuk melihat berbagai peta online.

Note: Fokus penulis dalam Materi ini adalah penggunaan **FO** di dalam perusahaan.

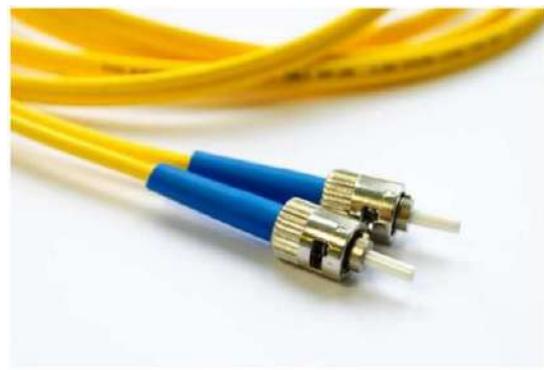
Konektor Fiber Optic

Berbagai konektor Fiber Optic tersedia. Perbedaan utama di antara jenis konektor adalah dimension dan metode coupling. Bisnis Terminate jenis konektor yang akan digunakan, berdasarkan peralatan mereka.

Catatan : Beberapa switch dan router memiliki port yang mendukung konektor fiber optic melalui small form-factor pluggable (SFP). Cari di internet untuk berbagai jenis SFP.

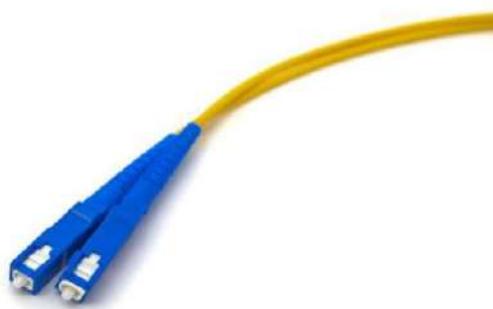
Konektor Straight-Tip (ST)

ST-Connector adalah salah satu jenis konektor pertama yang digunakan. Konektor mengunci secara aman dengan mekanisme jenis bayonet “Twist-on/twist-off” (di puter puter).



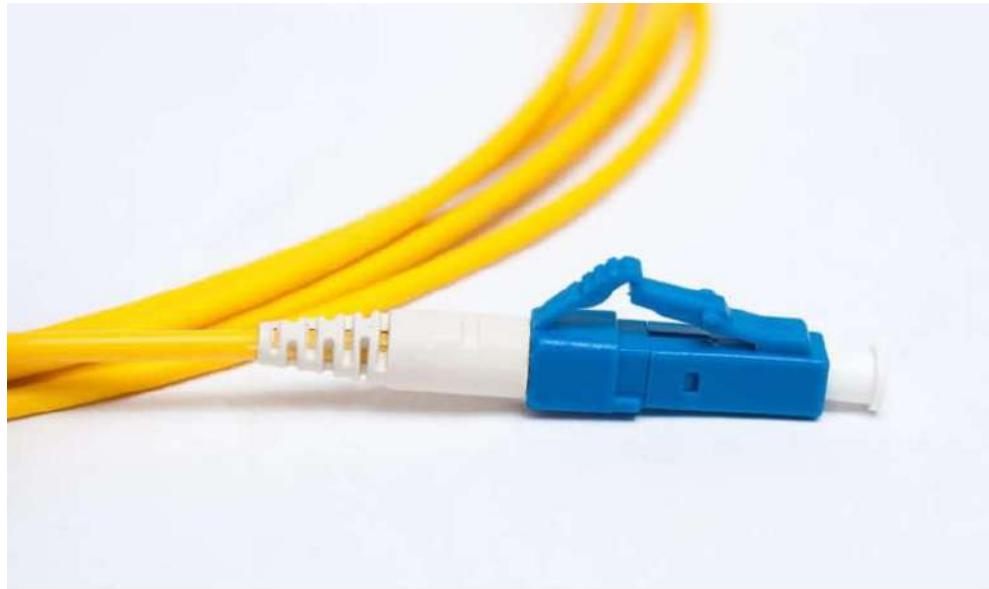
Subscriber Connecter (SC)

Konektor SC terkadang disebut sebagai **square connector** atau **standard connector**. Mereka adalah konektor LAN dan WAN yang diadopsi secara luas yang menggunakan mekanisme push-pull untuk memastikan positif insertion. Jenis konektor ini digunakan dengan multimode dan single mode



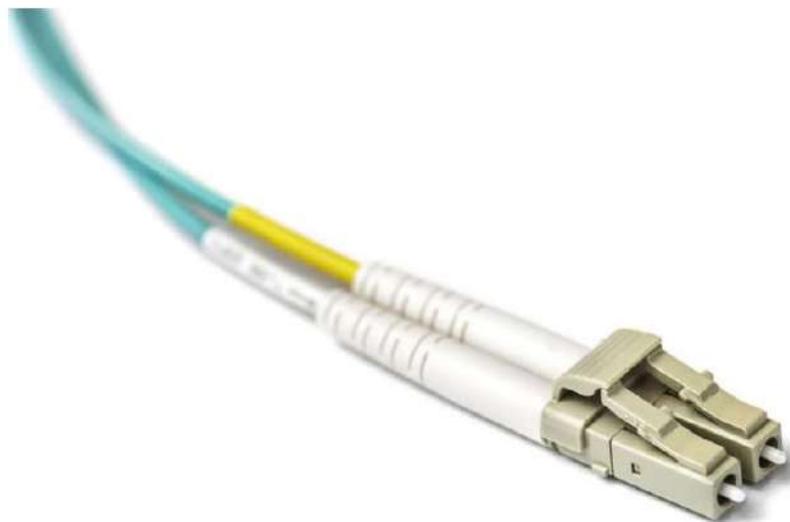
Lucent Connector (LC) Simplex Connector

Konektor simpleks **LC** adalah versi yang lebih kecil dari konektor **SC**. Ini kadang-kadang disebut **small connector** atau **local connector** , **LC** dengan cepat semakin populer karena ukurannya yang lebih kecil.



Duplex Multimode LC Connector

Konektor LC multimode duplex mirip dengan konektor simpleks LC, tetapi menggunakan konektor *dupleks*.



Sampai saat ini, cahaya hanya dapat bergerak dalam satu arah melalui **Fiber Optic**. Dua Fiber diperlukan untuk mendukung operasi *dupleks* penuh. Oleh karena itu, kabel patch menggabungkan dua kabel **Fiber Optic** dan **terminate** mereka dengan sepasang konektor **single-fiber connectors**. Beberapa konektor fiber menerima fiber pengirim dan penerima dalam satu konektor yang dikenal sebagai konektor *dupleks*, seperti yang ditunjukkan pada Konektor LC Multimode Duplex pada gambar. Standar BX seperti 100BASE-BX menggunakan panjang gelombang yang berbeda untuk mengirim dan menerima melalui satu fiber.

Fiber Patch Cords

Fiber Patch Cords diperlukan untuk menghubungkan perangkat infrastruktur. Penggunaan warna membedakan kabel patch single-mode dan multimode. *Yellow Jacket* untuk kabel **fiber single mode** dan oranye (atau aqua) untuk kabel **fiber multimode**.

SC-SC Multimode Patch Cord



LC-LC Single-mode Patch Cord



ST-LC Multimode patch Cord



ST-SC Single-Mode patch cord



Catatan : Kabel fiber harus dilindungi dengan tutup plastik kecil saat tidak digunakan.

Fiber Versus Tembaga

Ada banyak keuntungan menggunakan kabel **fiber optic** dibandingkan dengan kabel tembaga. Tabel menyoroti beberapa perbedaan ini.

Saat ini, di sebagian besar lingkungan perusahaan, serat optik terutama digunakan sebagai kabel *backbone* untuk lalu lintas tinggi, koneksi **point to point** antara fasilitas distribusi data. Ini juga digunakan untuk interkoneksi gedung di kampus multi gedung. Karena kabel **fiber optic** tidak menghantarkan listrik dan memiliki kehilangan sinyal yang rendah, kabel tersebut sangat cocok untuk penggunaan ini.

Perbandingan Kabel UTP dan Serat Optik

| Masalah Implementasi | Kabel UTP | Kabel Serat Optik |
|-----------------------------------|--------------------------------|-------------------------------------|
| Bandwidth didukung | 10 Mb / dtk – 10 Gb / dtk | 10 Mb / dtk – 100 Gb / dtk |
| Jarak | Relatif pendek (1 – 100 meter) | Relatif panjang (1 – 100.000 meter) |
| Kekebalan terhadap EMI dan RFI | Rendah | Tinggi (Sangat kebal) |
| Kekebalan terhadap bahaya listrik | Rendah | Tinggi (Sangat kebal) |
| Biaya media dan konektor | Terendah | Paling tinggi |
| Keterampilan instalasi diperlukan | Terendah | Paling tinggi |
| Tindakan pengamanan | Terendah | Paling tinggi |

Media Wireless

Anda mungkin melihat ini menggunakan tablet atau smartphone yang hanya dimungkinkan menggunakan media wireless, yang merupakan cara ketiga untuk terhubung ke Physical Layer dari jaringan.

Properti Media Wireless

Media wireless membawa sinyal elektromagnetik yang mewakili digit biner komunikasi data menggunakan frekuensi radio atau microwave.

Media wireless menyediakan opsi mobilitas terbesar dari semua media, dan jumlah perangkat yang diaktifkan wireless terus meningkat. wireless sekarang menjadi cara utama pengguna terhubung ke jaringan rumah dan perusahaan.

Ini adalah beberapa keterbatasan wireless:

- **Coverage area** – Teknologi komunikasi data wireless bekerja dengan baik di lingkungan terbuka. Namun, bahan konstruksi tertentu yang digunakan dalam bangunan dan struktur, dan medan lokal, akan membatasi cakupan yang efektif.
- **Interference** – wireless rentan terhadap gangguan dan dapat terganggu oleh perangkat umum seperti telepon wireless rumah tangga, beberapa jenis lampu neon, oven microwave, dan komunikasi wireless lainnya.
- **Security** – Cakupan komunikasi wireless tidak memerlukan akses ke *untaian fisik* media. Oleh karena itu, perangkat dapat membuat koneksi atau mendengar secara mudah dan ini termasuk rentan untuk keamanan. Keamanan jaringan adalah komponen utama administrasi jaringan wireless.
- **Shared medium** – WLAN beroperasi dalam half-duplex, yang berarti hanya satu perangkat yang dapat mengirim atau menerima pada satu waktu. Media wireless dibagikan di antara semua pengguna wireless. Banyak pengguna yang mengakses WLAN secara bersamaan menghasilkan pengurangan bandwidth untuk setiap pengguna.

Meskipun wireless meningkat popularitasnya untuk konektivitas desktop. tembaga dan fiber adalah media Physical Layer paling populer untuk penyebaran perangkat Intermediary Devices, seperti router dan switch.

Tipe Media Wireless

Standar industri **IEEE** dan telekomunikasi untuk komunikasi data wireless mencakup data link dan Physical layer. Dalam setiap standar ini, spesifikasi physical layer diterapkan ke area yang mencakup yang berikut:

- Pengkodean/encoding sinyal data ke radio
- Frekuensi dan daya transmisi
- Persyaratan penerimaan sinyal dan decoding
- Desain dan konstruksi antena

Ini adalah standar **wireless**:

Teknologi Wi-Fi (IEEE 802.11)

Wireless LAN (WLAN), yang biasa disebut sebagai Wi-Fi. **WLAN** menggunakan protokol berbasis **contention-based** yang dikenal sebagai **carrier sense multiple access/collision avoidance (CSMA/CA)**. NIC wireless harus terlebih dahulu mendengarkan sebelum mentransmisikan untuk menentukan apakah saluran radio jelas. Jika perangkat wireless lain ditransmisikan, maka NIC harus menunggu sampai saluran bersih. Wi-Fi adalah merek dagang dari Aliansi Wi-Fi. Wi-Fi digunakan dengan perangkat WLAN bersertifikat berdasarkan standar IEEE 802.11.

Bluetooth (IEEE 802.15)

Standar **wireless personal area network (WPAN)**, umumnya dikenal sebagai **“Bluetooth.”** Ini menggunakan proses pemasangan perangkat untuk berkomunikasi jarak dari 1 hingga 100 meter.

WiMAX (IEEE 802.16)

Umumnya dikenal sebagai **Worldwide Interoperability for Microwave Access (WiMAX)**, standar wireless ini menggunakan topologi point-to-multipoint untuk menyediakan akses broadband wireless.

Zigbee (IEEE 802.15.4)

Zigbee adalah spesifikasi yang digunakan untuk tingkat data rendah, komunikasi daya rendah. Ini ditujukan untuk aplikasi yang membutuhkan jarak pendek, kecepatan data rendah,

dan masa pakai baterai yang lama. **Zigbee** biasanya digunakan untuk lingkungan industri dan Internet of Things (IoT) seperti saklar lampu **wireless** dan pengumpulan data perangkat medis.

Catatan: Teknologi **wireless** lainnya seperti komunikasi seluler dan satelit juga dapat menyediakan koneksi jaringan data. Namun, teknologi **wireless** ini berada di luar lingkup untuk pelajaran ini.

Wireless Local Area Network

Implementasi data wireless umum memungkinkan perangkat untuk terhubung secara wireless melalui LAN. Secara umum, WLAN memerlukan perangkat jaringan berikut:

- **Wireless Access Point (AP)** – Ini memusatkan sinyal **wireless** dari pengguna dan terhubung ke infrastruktur jaringan berbasis tembaga yang ada, seperti **Ethernet**. Router wireless rumah dan bisnis kecil mengintegrasikan fungsi router, switch, dan access point ke dalam satu perangkat.
- **Adaptor NIC wireless** – Ini menyediakan kemampuan komunikasi wireless untuk host jaringan.

Seiring berkembangnya teknologi, sejumlah standar berbasis **WLAN Ethernet** telah muncul. Saat membeli perangkat **wireless**, pastikan kompatibilitas dan *interoperabilitas*.



Ketika Access Point, Router, Switch Dijadikan 1

Manfaat teknologi komunikasi data **wireless** terbukti, terutama penghematan pada kabel tempat yang mahal dan kenyamanan mobilitas tuan rumah. Administrator jaringan harus mengembangkan dan menerapkan kebijakan dan proses keamanan yang ketat untuk melindungi **WLAN** dari ancaman serangan dari luar

BAB 5

~ *Sistem Nomor* ~

Judul Bab : Sistem Nomor

Tujuan Bab: Menghitung nomor diantara desimal, biner, dan hexadesimal sistem

Link Test Pemahaman : <https://s.id/QxiX>

| Judul Materi | Tujuan Materi |
|-----------------------------|--|
| Sistem bilangan binary | Menghitung nomor diantara sistem desimal dan biner |
| Sistem bilangan hexadesimal | Menghitung nomor diantara sistem desimal dan hexadesimal |

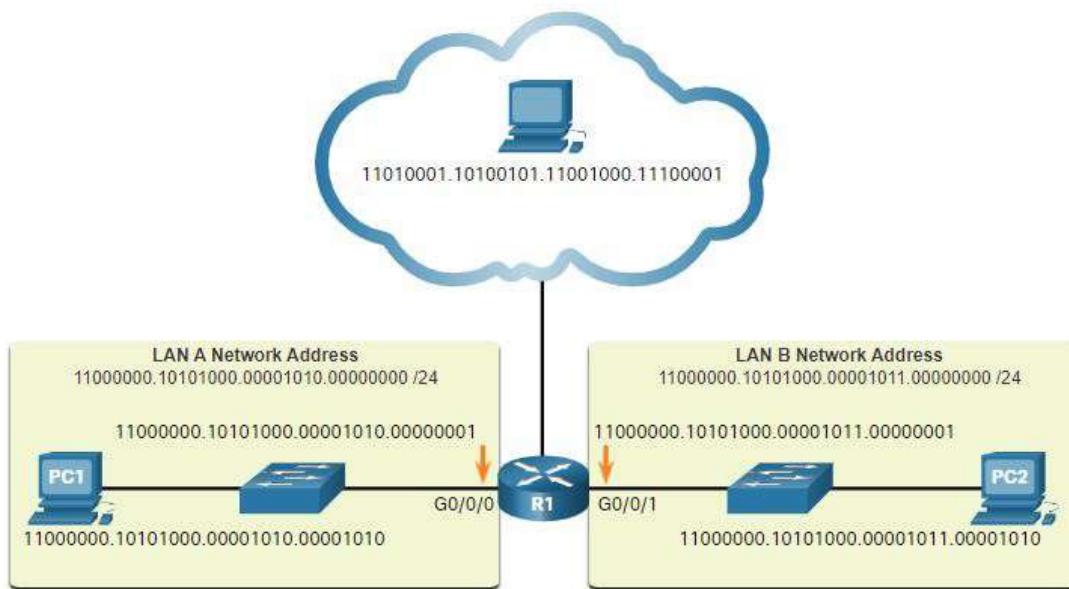
Sistem bilangan binary

Alamat **IPv4** dimulai sebagai **Binary** yaitu serangkaian hanya 1s dan 0s. Ini menjadikan sistem bilangan biner sulit dikelola, sehingga administrator jaringan harus mengkonversinya menjadi **desimal**. Pembahasan kali ini menunjukkan kepada Anda beberapa cara untuk melakukan hal ini.

Alamat Biner dan IPv4

Binary adalah sistem penomor yang terdiri dari digit 0 dan 1 yang disebut **bit**. Sebaliknya, sistem penomoran desimal terdiri dari 10 digit yang terdiri dari digit 0 – 9.

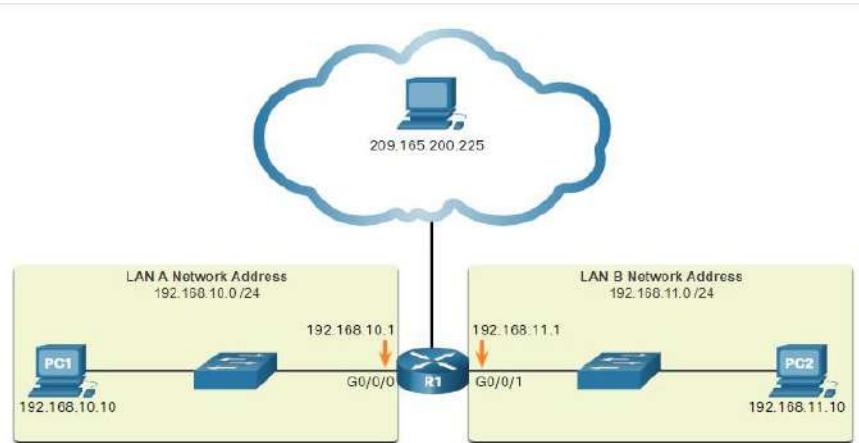
Binary penting bagi kita untuk memahami karena **host**, **server**, dan perangkat jaringan menggunakan alamat **Binary**. Secara khusus, mereka menggunakan alamat IPv4 biner, seperti yang ditunjukkan pada gambar, untuk mengidentifikasi satu sama lain.



Setiap alamat terdiri dari serangkaian **32 bit**, dibagi menjadi empat bagian yang disebut **oktet**. Setiap oktet berisi **8 bit (atau 1 byte)** dipisahkan dengan **titik**. Misalnya, PC1 dalam gambar ditetapkan alamat **IPv4** 11000000.10101000.00001010.00001010. Alamat default gateway nya adalah interface Ethernet R1 Gigabit 11000000.10101000.00001010.000000001.

Binary bekerja dengan baik dengan host dan perangkat jaringan. Namun, sangat susah dibaca dengan baik dengan manusia.

Untuk kemudahan penggunaan oleh orang-orang, alamat IPv4 biasanya dinyatakan dalam notasi **dotted-decimal**. PC1 diberi alamat IPv4 192.168.10.10, dan alamat default gateway nya adalah 192.168.10.1, seperti yang ditunjukkan pada gambar.



Untuk pemahaman yang solid tentang **troubleshooting jaringan**, perlu diketahui manajemen **binary** dan mendapatkan keterampilan praktis mengkonversi antara **binary** dan alamat IPv4 **dotted-decimal**.

Notasi Posisi Binary

Belajar mengkonversi **Biner** ke **desimal** membutuhkan pemahaman tentang notasi posisi. Notasi posisi berarti *bahwa digit mewakili nilai yang berbeda tergantung pada "posisi" yang ditempati digit dalam urutan angka*. Anda sudah tahu sistem nomor nomor yang paling umum, sistem notasi desimal (basis 10).

Sistem notasi posisi desimal beroperasi seperti yang dijelaskan dalam tabel.

| | | | | |
|---------------------------|----------|----------|----------|----------|
| Radix | 10 | 10 | 10 | 10 |
| Posisi dalam Angka | 3 | 2 | 1 | 0 |
| Menghitung | (10^3) | (10^2) | (10^1) | (10^0) |
| Nilai posisi | 1000 | 100 | 10 | 1 |

Catatan: $n0 = 1$.

Poin berikut ini menjelaskan setiap baris tabel.

- Baris 1, Radix adalah basis angka. Notasi desimal didasarkan pada 10, oleh karena itu radix adalah 10.
- Baris 2, Posisi dalam jumlah mempertimbangkan posisi angka desimal dimulai dengan, dari kanan ke kiri, 0 (posisi 1), 1 (posisi ke-2), 2 (posisi ke-3), 3 (posisi ke-4). Angka-angka ini juga mewakili penggunaan nilai eksponensial untuk menghitung nilai posisi di baris ke-4.
- Baris 3 menghitung nilai posisi dengan mengambil radix dan menaikkannya dengan nilai eksponensial posisinya di baris 2.
- Baris 4 nilai posisi mewakili unit ribuan, ratusan, puluhan, dan satu.

Untuk menggunakan sistem posisi, cocokkan angka tertentu dengan nilai posisinya. Contoh dalam tabel menggambarkan bagaimana notasi posisi digunakan dengan angka desimal 1234.

| | Ribu | Ratusan | Puluhan | Yang Satu |
|-----------------------------|-----------------|----------------|---------------|--------------|
| Nilai Posisi | 1000 | 100 | 10 | 1 |
| Nomor Desimal (1234) | 1 | 2 | 3 | 4 |
| Menghitung | 1×1000 | 2×100 | 3×10 | 4×1 |
| Tambahkan mereka... | 1000 | + 200 | + 30 | + 4 |
| Hasil | 1,234 | | | |

Sebaliknya, notasi posisi biner beroperasi seperti yang dijelaskan dalam tabel.

| | | | | | | | | |
|---------------------------|---------|---------|---------|---------|---------|---------|---------|---------|
| Radix | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |
| Posisi dalam Angka | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Menghitung | (2^7) | (2^6) | (2^5) | (2^4) | (2^3) | (2^2) | (2^1) | (2^0) |
| Nilai posisi | 128 | 64 | 32 | 16 | 8 | 4 | 2 | 1 |

Poin berikut ini menjelaskan setiap baris tabel.

- Baris 1, Radix adalah basis angka. Notasi biner didasarkan pada 2, oleh karena itu radix adalah 2.

- Baris 2, Posisi dalam angka mempertimbangkan posisi angka biner dimulai dengan, dari kanan ke kiri, 0 (posisi 1), 1 (posisi ke-2), 2 (posisi ke-3), 3 (posisi ke-4). Angka-angka ini juga mewakili penggunaan nilai eksponensial untuk menghitung nilai posisi di baris ke-4.
 - Baris 3 menghitung nilai posisi dengan mengambil radix dan menaikkannya dengan nilai eksponensial posisinya di baris 2
- Catatan: $n0 = 1$.**
- Baris 4 nilai posisi mewakili satuan dari satu, dua, empat, delapan, dll.

Contoh dalam tabel menggambarkan bagaimana bilangan biner 11000000 sesuai dengan angka 192. Jika bilangan biner telah 10101000, maka angka desimal yang sesuai adalah 168.

| Nilai Posisi | 128 | 64 | 32 | 16 | 8 | 4 | 2 | 1 |
|------------------------------|----------------|---------------|---------------|---------------|--------------|--------------|--------------|--------------|
| Bilangan Biner (11000000) | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| Menghitung | 1×128 | 1×64 | 0×32 | 0×16 | 0×8 | 0×4 | 0×2 | 0×1 |
| Tambahkan mereka.. | 128 | + 64 | + 0 | + 0 | + 0 | + 0 | + 0 | + 0 |
| Hasil | 192 | | | | | | | |

Konversi Biner ke Desimal

Alamat IPv4 menjadi empat oktet 8-bit. Selanjutnya terapkan nilai posisi Binary ke bilangan Binary oktet pertama dan hitung sesuai.

Misalnya, pertimbangkan bahwa 11100000.10101000.00001011.00001010 adalah alamat **Binary IPv4** dari **host**. Untuk mengonversi alamat **Binary** menjadi **desimal**, mulailah dengan **oktet** pertama, seperti yang ditunjukkan dalam tabel. Masukkan bilangan **Binary** 8-bit di bawah nilai posisi baris 1 lalu hitung untuk menghasilkan angka desimal 192. Angka ini masuk ke **oktet** pertama dari notasi **dotted decimal**.

| | | | | | | | | |
|----------------------------------|-----|------|-----|-----|-----|-----|-----|-----|
| Nilai Posisi | 128 | 64 | 32 | 16 | 8 | 4 | 2 | 1 |
| Bilangan Biner (11000000) | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| Menghitung | 128 | 64 | 32 | 16 | 8 | 4 | 2 | 1 |
| Tambahkan mereka... | 128 | + 64 | + 0 | + 0 | + 0 | + 0 | + 0 | + 0 |
| Hasil | 192 | | | | | | | |

Selanjutnya konversi **oktet** kedua 10101000 seperti yang ditunjukkan dalam **tabel**. Nilai **desimal** yang dihasilkan adalah 168, dan masuk ke oktet kedua.

| | | | | | | | | |
|---------------------------------|-----|-----|------|-----|-----|-----|-----|-----|
| Nilai Posisi | 128 | 64 | 32 | 16 | 8 | 4 | 2 | 1 |
| Binary Number (10101000) | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| Menghitung | 128 | 64 | 32 | 16 | 8 | 4 | 2 | 1 |
| Tambahkan mereka... | 128 | + 0 | + 32 | + 0 | + 8 | + 0 | + 0 | + 0 |
| Hasil | 168 | | | | | | | |

Mengkonversi **oktet** ketiga 00001011 seperti yang ditunjukkan dalam tabel.

| | | | | | | | | |
|---------------------------------|-----|-----|-----|-----|-----|-----|-----|-----|
| Nilai Posisi | 128 | 64 | 32 | 16 | 8 | 4 | 2 | 1 |
| Binary Number (00001011) | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 |
| Menghitung | 128 | 64 | 32 | 16 | 8 | 4 | 2 | 1 |
| Tambahkan mereka... | 0 | + 0 | + 0 | + 0 | + 8 | + 0 | + 2 | + 1 |
| Hasil | 11 | | | | | | | |

Mengkonversi **oktet** keempat 00001010 seperti yang ditunjukkan dalam tabel. Ini melengkapi alamat IP dan menghasilkan **192.168.11.10**.

| | | | | | | | | |
|---------------------------------|-----|-----|-----|-----|-----|-----|-----|-----|
| Nilai Posisi | 128 | 64 | 32 | 16 | 8 | 4 | 2 | 1 |
| Binary Number (00001010) | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |
| Menghitung | 128 | 64 | 32 | 16 | 8 | 4 | 2 | 1 |
| Tambahkan mereka... | 0 | + 0 | + 0 | + 0 | + 8 | + 0 | + 2 | + 0 |
| Hasil | 10 | | | | | | | |

Alamat IPv4

Seperti disebutkan di awal Materi ini, **router** dan komputer hanya memahami **Binary**, sementara manusia bekerja dalam **desimal**. Penting bagi Anda untuk mendapatkan pemahaman menyeluruh tentang dua sistem penomor nama ini dan bagaimana mereka digunakan dalam jaringan.

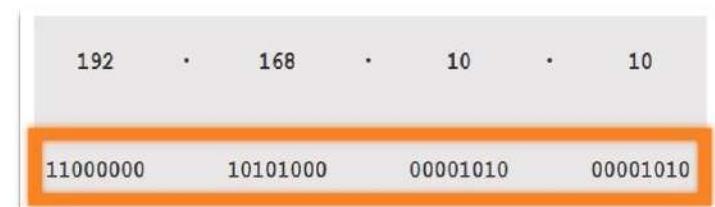
Dotted Decimal Address



Oktet



32-bit addresses



Sistem bilangan hexadesimal

Sekarang Anda tahu cara mengkonversi **biner** ke **desimal** dan **desimal** ke **binary**. Anda memerlukan keterampilan itu untuk memahami alamat IPv4 di jaringan Anda. Tetapi Anda sama mungkin menggunakan alamat IPv6 di jaringan Anda. Untuk memahami alamat IPv6, Anda harus dapat mengkonversi **heksadesimal** menjadi **desimal** dan sebaliknya.

Alamat Heksadesimal dan IPv6

Sama seperti desimal adalah sistem angka dasar sepuluh, heksadesimal adalah sistem *enam belas dasar*. Sistem angka enam belas dasar *menggunakan angka 0 hingga 9 dan huruf A hingga F*. Angka tersebut menunjukkan nilai desimal dan heksadesimal yang setara untuk binary 0000 hingga 1111.

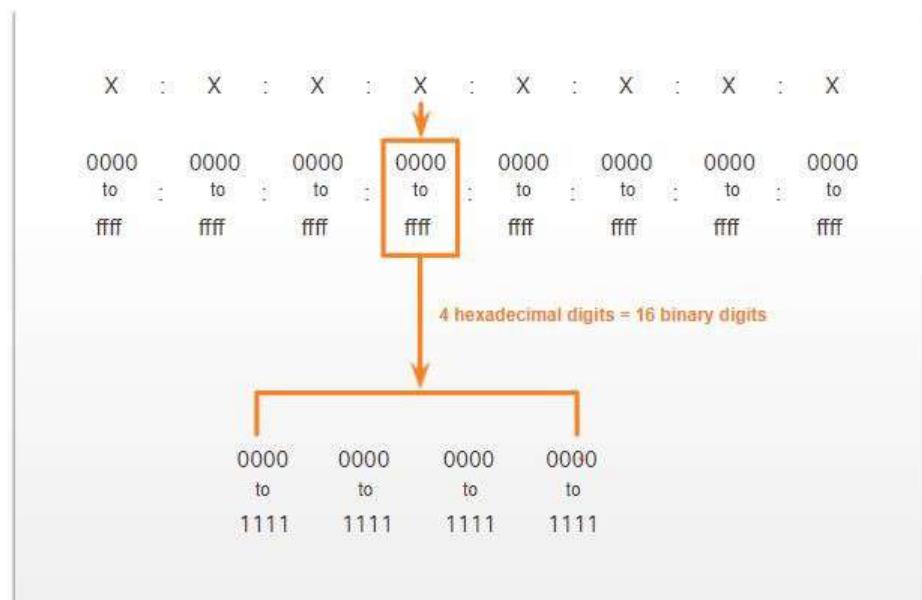
| Decimal | Binary | Hexadecimal |
|---------|--------|-------------|
| 0 | 0000 | 0 |
| 1 | 0001 | 1 |
| 2 | 0010 | 2 |
| 3 | 0011 | 3 |
| 4 | 0100 | 4 |
| 5 | 0101 | 5 |
| 6 | 0110 | 6 |
| 7 | 0111 | 7 |
| 8 | 1000 | 8 |
| 9 | 1001 | 9 |
| 10 | 1010 | A |
| 11 | 1011 | B |
| 12 | 1100 | C |
| 13 | 1101 | D |
| 14 | 1110 | E |
| 15 | 1111 | F |

Binary dan heksadesimal bekerja sama dengan baik karena lebih mudah untuk mengekspresikan nilai sebagai digit heksadesimal tunggal daripada sebagai empat bit biner.

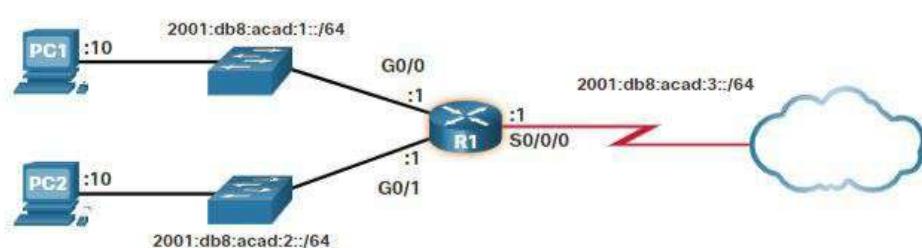
Sistem penomor heksadesimal digunakan dalam jaringan untuk mewakili alamat IP Versi 6 dan alamat MAC Ethernet.

Alamat IPv6 memiliki panjang 128 bit dan setiap 4 bit diwakili oleh satu digit heksadesimal; dengan total 32 nilai heksadesimal. Alamat IPv6 *tidak peka huruf besar/kecil dan dapat ditulis dalam huruf kecil atau huruf besar*.

Seperti yang ditunjukkan pada gambar, format yang disukai untuk menulis alamat **IPv6** adalah x:x:x:x:x:x:x:x, dengan masing-masing “x” terdiri dari empat nilai **heksadesimal**. Jika mengacu pada 8 bit alamat **IPv4** kita menggunakan istilah **oktet**. Dalam **IPv6**, **hextet** adalah istilah tidak resmi yang digunakan untuk merujuk pada segmen 16 bit atau empat nilai **heksadesimal**. Setiap “x” adalah **hextet** tunggal, 16 bit, atau empat digit **heksadesimal**.



Contoh topologi dalam gambar menampilkan alamat **heksadesimal IPv6**.



Konversi Desimal ke Heksadesimal

Mengkonversi angka **desimal** menjadi nilai **heksadesimal** sangat mudah. Ikuti langkah-langkah yang tercantum:

1. Mengkonversi angka **desimal** menjadi string **binary** 8-bit.
2. Bagi string **binary** dalam grup empat dimulai dari posisi paling kanan.
3. Konversikan setiap empat bilangan **binary** menjadi digit **heksadesimal** yang setara.

Contohnya menyediakan langkah-langkah untuk mengonversi **168** ke **heksadesimal**.

Misalnya, **168** dikonversi menjadi hex menggunakan proses tiga langkah.

1. **168** dalam biner adalah **10101000**.
2. **10101000** dalam dua kelompok dari empat digit biner adalah **1010** dan **1000**.
3. **1010** adalah hex **A** dan **1000** adalah hex **8**.

Jawaban: **168** adalah **A8** dalam heksadesimal.

Konversi Heksadesimal ke Desimal

Mengkonversi angka **heksadesimal** menjadi nilai **desimal** juga mudah. Ikuti langkah-langkah yang tercantum:

1. Mengonversi angka **heksadesimal** menjadi string **binary** 4-bit.
2. Buat pengelompokan **binary** 8-bit mulai dari posisi paling kanan.
3. Konversikan setiap pengelompokan **binary** 8-bit menjadi digit **desimal** yang setara.

Contoh ini menyediakan langkah-langkah untuk mengonversi **D2** ke desimal.

1. **D2** dalam string biner 4-bit adalah **1101** dan **0010**.
2. **1101** dan **0010** adalah **11010010** dalam pengelompokan 8-bit.
3. **11010010** dalam biner setara dengan **210** dalam desimal.

Jawaban: **D2** dalam heksadesimal adalah **210** dalam desimal.

BAB 6

~ Data Link Layer ~

Judul Bab: Data Link Layer

Tujuan Bab : Menjelaskan bagaimana media access control di layer data link mendukung komunikasi di beda jaringan

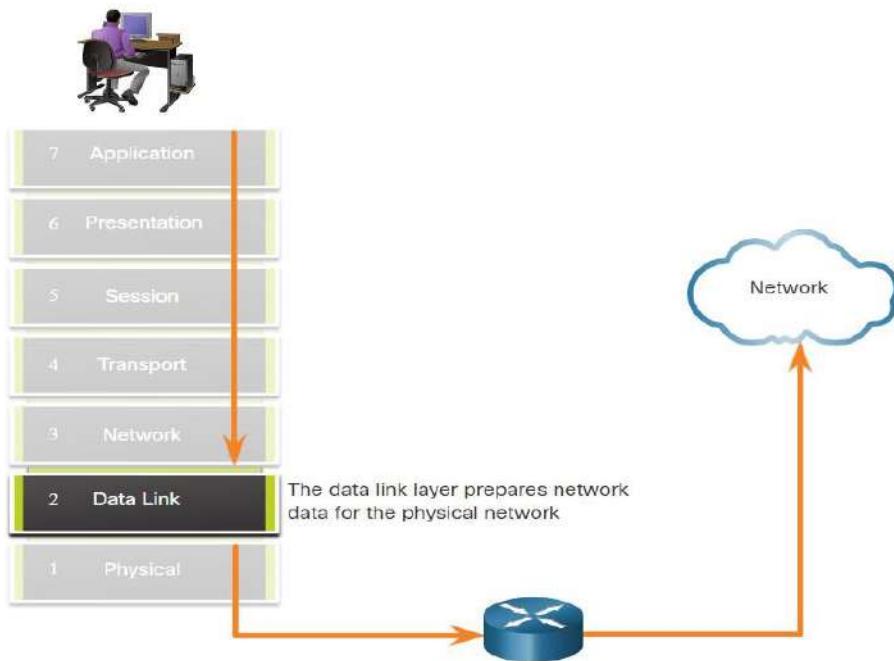
Link Test Pemahaman : <https://s.id/QxkE>

| Judul Materi | Tujuan Materi |
|-----------------------------|---|
| Tujuan dari data link layer | Menjelaskan tujuan dan fungsi data link layer dalam mempersiapkan komunikasi untuk transmisi pada media tertentu. |
| Beberapa Topologi | Membandingkan karakteristik dari media access control di topologi WAN dan LAN |
| Data Link Frame | Menjelaskan karakteristik dan fungsi dari data link frame |

Tujuan Data link layer

Data Link Layer dari model OSI (Layer 2), seperti yang ditunjukkan pada gambar, data link fungsinya adalah menyiapkan data jaringan untuk jaringan fisik (physical layer). Data Link Layer bertanggung jawab atas komunikasi Network Interface Card (NIC) ke Network Interface Card. Data Link Layer melakukan hal berikut:

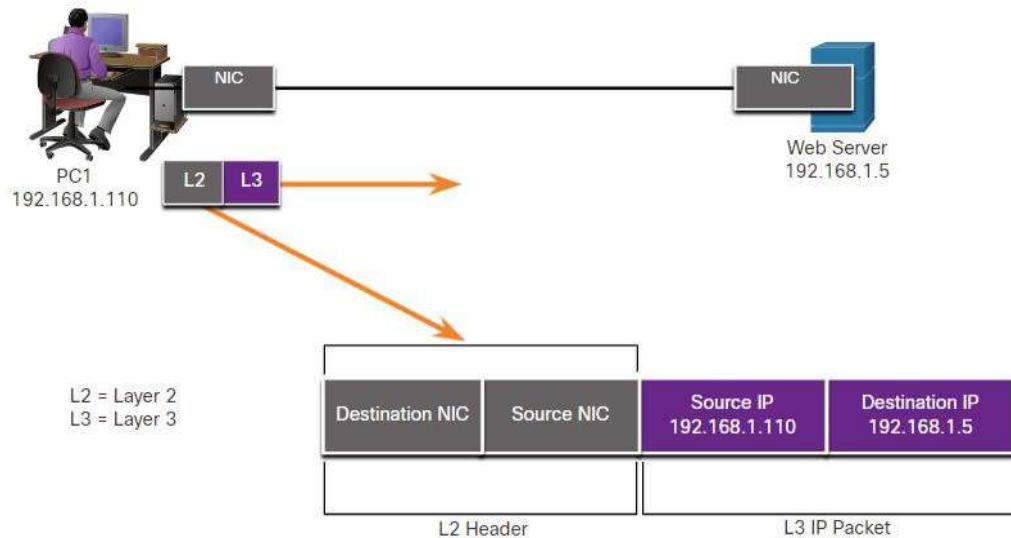
- Memungkinkan upper layer untuk mengakses media. Protokol upper layer sama sekali tidak menyadari jenis media yang digunakan untuk meneruskan data.
- Menerima data, biasanya paket Layer 3 (yaitu, IPv4 atau IPv6), dan merangkumnya ke dalam frame Layer 2.
- Mengontrol bagaimana data ditempatkan dan diterima di media.
- Pertukaran frame antara node akhir melalui media jaringan.
- Menerima data enkapsulasi, biasanya paket Layer 3, dan mengarahkannya ke protokol upper layer yang tepat.
- Melakukan deteksi kesalahan dan menolak frame yang rusak.



Dalam jaringan komputer, node/host adalah perangkat yang dapat menerima, membuat, menyimpan, atau meneruskan data di sepanjang jalur komunikasi. Node/host dapat berupa End Devices seperti laptop atau ponsel, atau Intermediary Devices seperti Switch Ethernet.

Tanpa Data Link Layer, protokol Network Layer seperti IP, harus membuat ketentuan untuk menghubungkan ke setiap jenis media yang bisa ada di sepanjang jalur pengiriman. Selain itu, setiap kali teknologi jaringan atau media baru dikembangkan IP, harus beradaptasi.

Gambar menampilkan contoh bagaimana Data Link Layer menambahkan tujuan Layer 2 Ethernet dan sumber informasi NIC ke paket Layer 3. Kemudian akan mengonversi informasi ini ke format yang didukung oleh Physical Layer (yaitu, Layer 1).

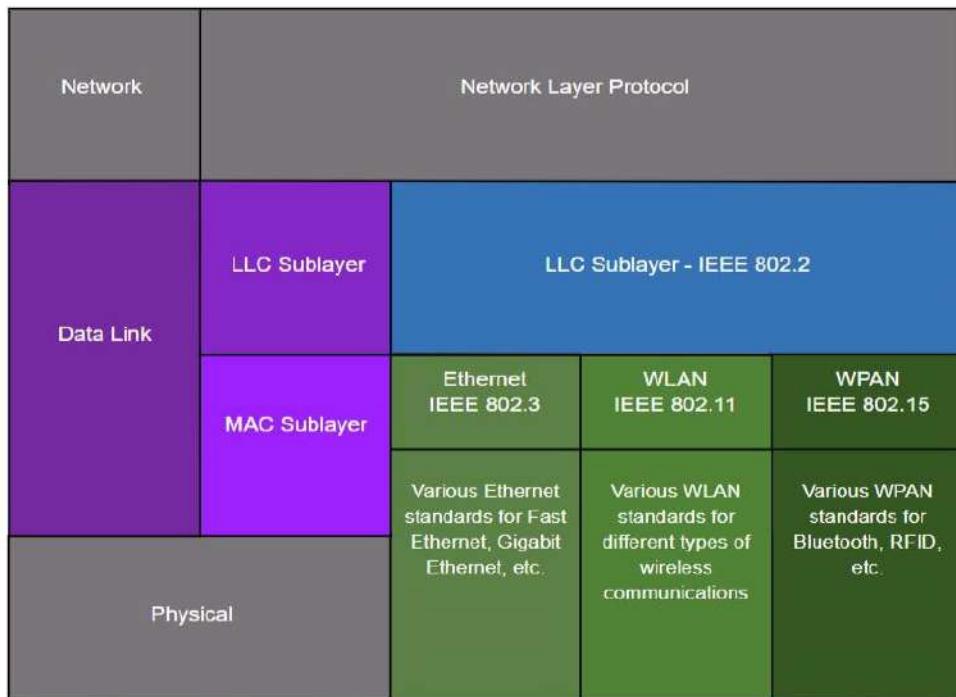


Sublayer Link Data IEEE 802 LAN/MAN

Standar LAN/MAN IEEE 802 khusus untuk LAN Ethernet, Wireless LAN (WLAN), Wireless Personal Area Network (WPAN) dan jenis jaringan area lokal dan metropolitan lainnya. Data Link Layer IEEE 802 LAN/MAN terdiri dari dua sublayer berikut:

- Logical Link Control (LLC) – Sublayer IEEE 802.2 ini berkomunikasi antara perangkat lunak jaringan di upper layer dan perangkat keras perangkat di bottom layer. Ini menempatkan informasi dalam Frame yang mengidentifikasi protokol Network Layer mana yang digunakan untuk Frame. Informasi ini memungkinkan beberapa protokol Layer 3, seperti IPv4 dan IPv6, untuk menggunakan Interface dan media jaringan yang sama.
- Media Access Control (MAC) – Menerapkan sublayer ini (IEEE 802.3, 802.11, atau 802.15) dalam perangkat keras. Ini bertanggung jawab atas enkapsulasi data dan kontrol akses media. MAC menyediakan pengalaman data link dan terintegrasi dengan berbagai teknologi Physical Layer.

Gambar menunjukkan dua sublayer (LLC dan MAC) dari Data Link Layer



Sublayer LLC mengambil data protokol jaringan, yang biasanya merupakan paket IPv4 atau IPv6, dan menambahkan informasi kontrol Layer 2 untuk membantu mengirimkan paket ke node tujuan.

Sublayer MAC mengontrol NIC dan perangkat keras lainnya yang bertanggung jawab untuk mengirim dan menerima data pada media LAN/MAN berkabel atau nirkabel.

Sublayer MAC menyediakan enkapsulasi data:

- Frame delimiting – Proses frame menyediakan pembatas penting untuk mengidentifikasi field dalam frame. frame delimiting ini menyediakan sinkronisasi antara node transmisi dan penerimaan.
- Addressing - Menyediakan alamat sumber dan tujuan untuk mengangkut Frame Layer 2 antara perangkat pada media bersama yang sama.
- Error Detection – Termasuk *trailer* yang digunakan untuk mendeteksi kesalahan transmisi.

Sublayer MAC juga menyediakan kontrol akses media, yang memungkinkan beberapa perangkat berkomunikasi melalui media bersama (half duplex). Komunikasi full dupleks tidak memerlukan kontrol akses.

Menyediakan Akses ke Media

Setiap lingkungan jaringan, paket yang ditemui saat mereka melakukan perjalanan dari host lokal ke host jarak jauh dapat memiliki karakteristik yang berbeda. Misalnya, LAN Ethernet biasanya terdiri dari banyak host yang bersaing untuk akses pada media jaringan. Sublayer MAC menyelesaikan ini. Dengan link serial metode akses hanya dapat terdiri dari koneksi langsung antara hanya dua perangkat, biasanya dua router. Oleh karena itu, mereka tidak memerlukan teknik yang digunakan oleh sublayer IEEE 802 MAC.

Interface router merangkum packet ke dalam Frame yang sesuai. Metode MAC yang sesuai digunakan untuk mengakses setiap Link. Dalam pertukaran paket Network Layer yang diberikan, mungkin ada banyak Data Link Layer dan transisi media.

Pada setiap hop di sepanjang jalur, router melakukan fungsi Layer 2 berikut:

1. Menerima frame dari media
2. De-enkapsulasi frame
3. Re-enkapsulasi paket ke dalam frame baru
4. Meneruskan frame baru yang sesuai ke media segmen jaringan fisik tersebut

Standar Data Link Layer

Protokol Data Link layer umumnya tidak ditentukan oleh Request for Comments (RFC), tidak seperti protokol upper layer rangkaian TCP/IP. Internet Engineering Task Force (IETF) mempertahankan protokol dan layanan fungsional untuk rangkaian protokol TCP/IP di upper layer, tetapi mereka tidak menentukan fungsi dan pengoperasian Network Access TCP/IP.

Organisasi teknik yang menentukan open-standard yang berlaku untuk Network Access Layer (yaitu, Physical dan data link layer) meliputi yang berikut:

- Institute of Electrical and Electronics Engineers (IEEE)
- International Telecommunication Union (ITU)
- International Organization for Standardization (ISO)
- American National Standards Institute (ANSI)

Logo Organisasi Teknik



Topologies

Seperti yang Anda pelajari Materi sebelumnya, Data Link Layer menyiapkan data jaringan untuk jaringan fisik. Data Link Layer harus tahu Logical Topology jaringan untuk dapat menentukan apa yang diperlukan untuk mentransfer Frame dari satu perangkat ke perangkat lain. Pembahasan kali ini menjelaskan cara kerja Data Link Layer dengan Logical Topology yang berbeda.

Physical and logical topology

Topologi adalah *pengaturan, atau hubungan, perangkat jaringan dan interkoneksi di antara mereka*.

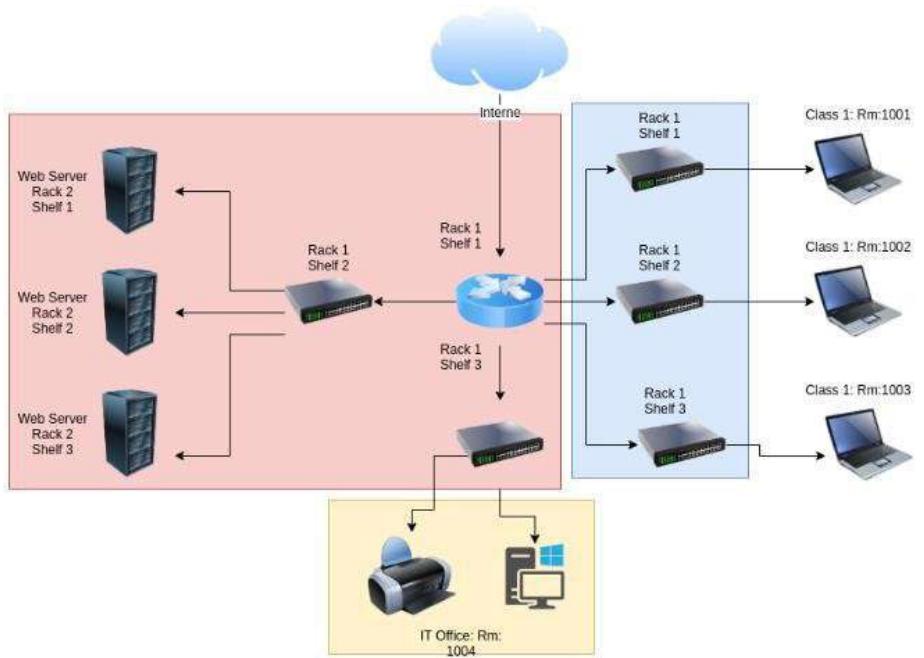
Ada dua jenis topologi yang digunakan saat menggambarkan jaringan **LAN** dan **WAN**:

- **Topology Physical** – Mengidentifikasi koneksi fisik dan bagaimana End Devices dan Intermediary Devices (yaitu, router, switch, dan Access Point) saling terhubung. Topology juga dapat mencakup lokasi perangkat tertentu seperti nomor kamar dan lokasi di rak peralatan. Topology Physical biasanya point-to-point atau *star*.
- **Topology Logical** – Mengacu pada cara jaringan mentransfer frame dari satu node ke node berikutnya. Topology ini mengidentifikasi koneksi virtual menggunakan Interface perangkat dan skema mengatasi IP Layer 3.

Data Link Layer “melihat” Topology Logical saat mengontrol akses data ke media. Ini adalah Topology Logical yang mempengaruhi jenis framing jaringan dan MAC yang digunakan.

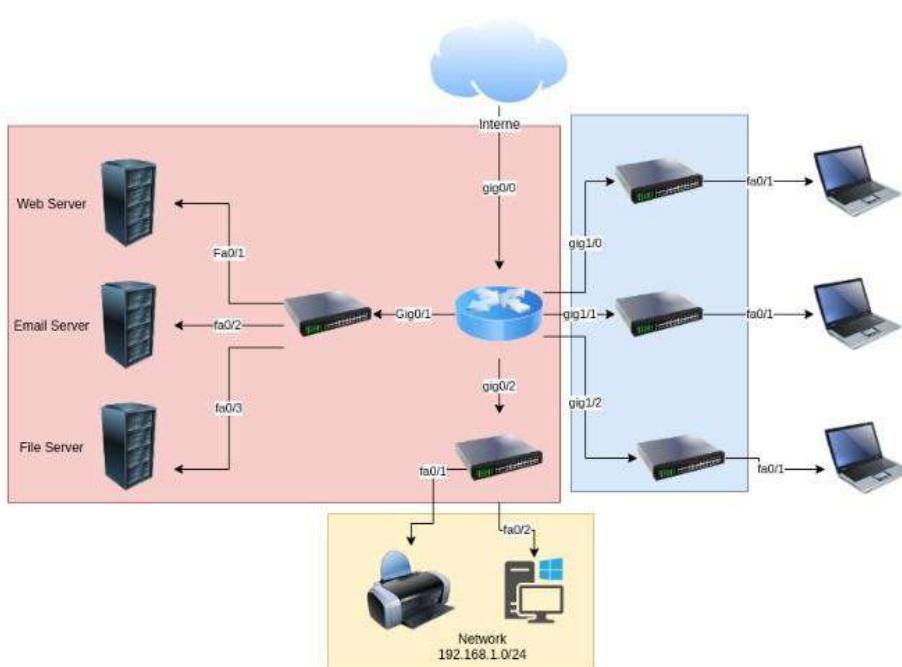
Gambar menampilkan contoh **Topology Physical** untuk jaringan sampel kecil.

Topology Physical



Gambar berikutnya menampilkan contoh **Topology Logical** untuk jaringan yang sama.

Topology Logical



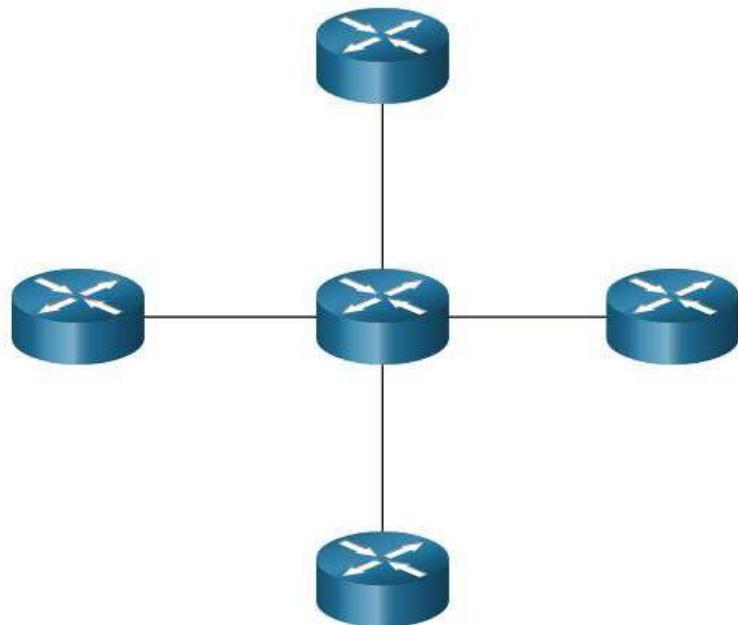
WAN Topologies

Point To Point



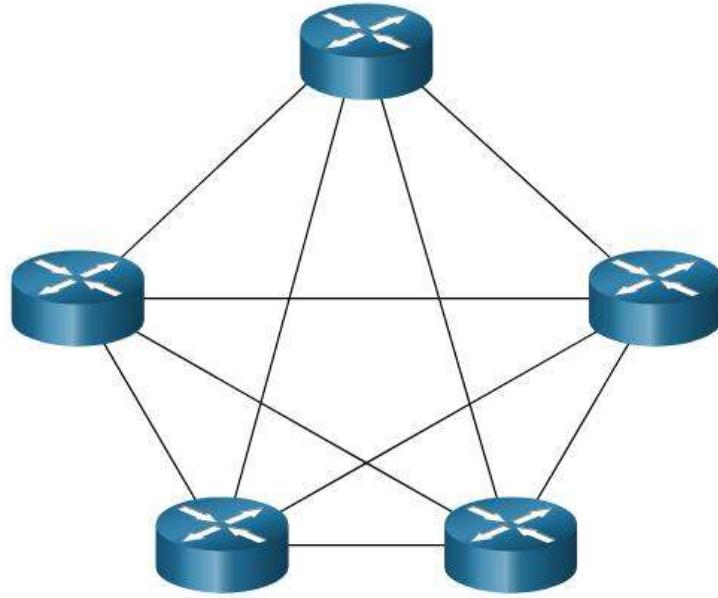
Point To Point adalah topologi WAN yang paling sederhana dan paling umum. Ini terdiri dari link permanen antara dua *endpoints*.

Hub And Spoke



Hub and spoke adalah versi WAN dari topologi *star* dimana router pusat menghubungkan router cabang melalui penggunaan link point to point. Router cabang tidak dapat bertukar data dengan router cabang lain tanpa melalui router pusat

Mesh



Topology ini memberikan ketersediaan tinggi tetapi mengharuskan setiap router saling terhubung ke setiap router lainnya. Oleh karena itu, biaya administrasi dan fisiknya bisa signifikan. Setiap **Link** pada dasarnya adalah link point-to-point ke router lain.

Hibrida

Topology Hibrida adalah variasi atau kombinasi dari topologi apa pun. Misalnya, *partial mesh* adalah topologi hybrid di mana beberapa tetapi tidak semua **End Devices** saling terhubung.

WAN Point-to-Point

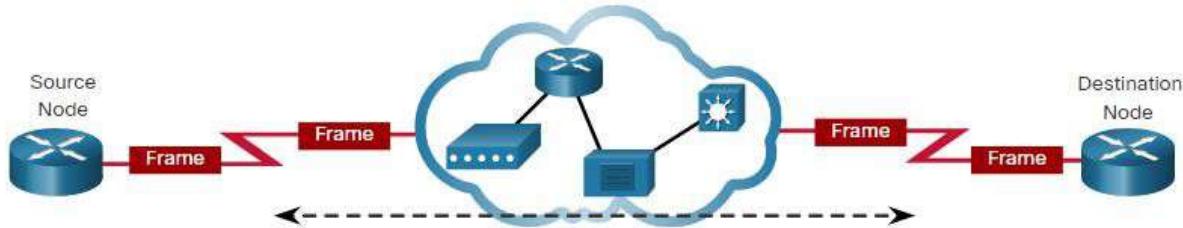
Topology point-to-point secara langsung menghubungkan dua **node**, seperti yang ditunjukkan pada gambar. Dalam pengaturan ini, dua node tidak harus berbagi media dengan host lain. Selain itu, ketika menggunakan protokol komunikasi serial seperti Point-to-Point Protocol (PPP), node tidak harus membuat penentuan tentang apakah frame masuk Diizinkan untuk itu atau node lain. Oleh karena itu, protokol Data Link bisa sangat sederhana, karena semua Frame di media hanya dapat melakukan perjalanan 2 node. Node menempatkan Frame pada media di satu ujung dan Frame tersebut diambil dari media oleh node di ujung lain sirkuit point-to-point.



Topology point-to-point terbatas pada dua node/host.

Catatan: Koneksi point-to-point melalui Ethernet mengharuskan perangkat untuk menentukan apakah frame masuk Diizinkan untuk node ini.

Node sumber dan tujuan mungkin secara tidak langsung terhubung satu sama lain melalui beberapa jarak geografis menggunakan beberapa Intermediary Device. Namun, penggunaan perangkat fisik dalam jaringan tidak mempengaruhi Topology Logical, seperti yang diilustrasikan dalam gambar. Dalam gambar, menambahkan koneksi fisik perantara mungkin tidak mengubah Topology Logical. Koneksi point-to-point yang logis adalah sama.



Topologi LAN

Dalam LAN multiaccess, **End Devices** (yaitu, node) saling terhubung menggunakan *topologi star atau topologi extended star*. Dalam jenis topologi ini, End Devices terhubung ke Intermediary Device pusat, dalam hal ini, switch Ethernet. topologi extended star ini dengan menghubungkan beberapa switch Ethernet. topologi star atau topologi extended star mudah dipasang, sangat dapat diskalakan (mudah ditambahkan dan dihapus end devices), dan mudah dipecahkan. Topologi star awal interkoneksi end devices menggunakan hub Ethernet.

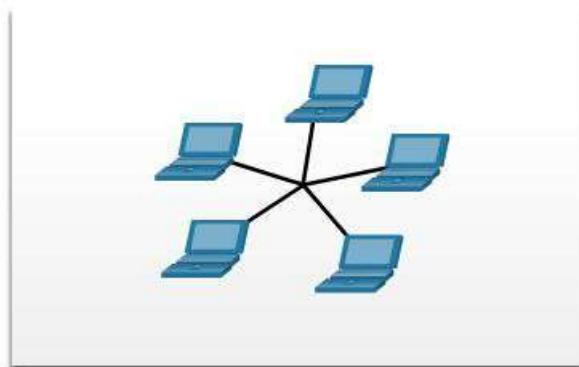
Kadang-kadang mungkin hanya ada dua perangkat yang terhubung di LAN Ethernet. Contohnya adalah dua router yang saling terhubung. Ini akan menjadi contoh Ethernet yang digunakan pada topologi point-to-point.

Legacy LAN Topologies

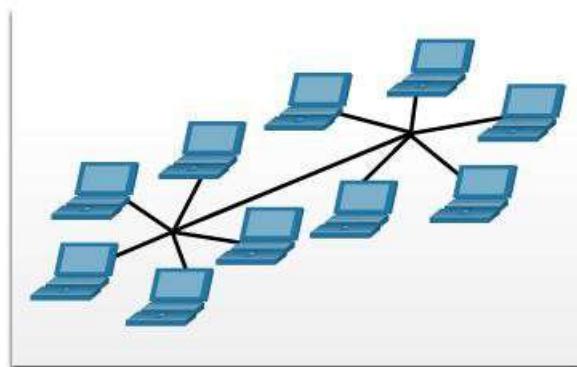
Teknologi **legacy Token Ring LAN** dan **Ethernet** versi lama mencakup dua jenis **topologi** lainnya:

- **Bus** – Semua node akhir dirantai satu sama lain dan dihentikan dalam beberapa bentuk di setiap ujungnya. Perangkat infrastruktur seperti **switch** tidak diperlukan untuk menghubungkan **end devices**. Jaringan **Legacy Ethernet** sering menjadi topologi bus menggunakan kabel **coaxial** karena murah dan mudah diatur.
- **Ring** – node akhir terhubung ke tetangga masing-masing membentuk cincin. Cincin tidak perlu dihentikan, tidak seperti di topologi bus. Jaringan Legacy **Fiber Distributed Data Interface (FDDI)** dan **Token Ring** menggunakan topologi cincin.

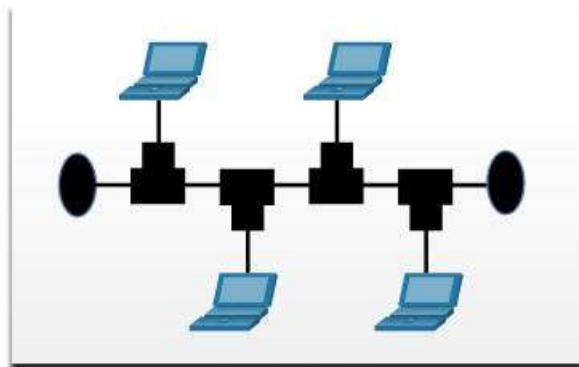
Gambar tersebut mengilustrasikan bagaimana **end devices** saling terhubung pada LAN. Garis lurus dalam grafik jaringan biasanya mewakili **LAN Ethernet** termasuk **star** dan **star extended**.



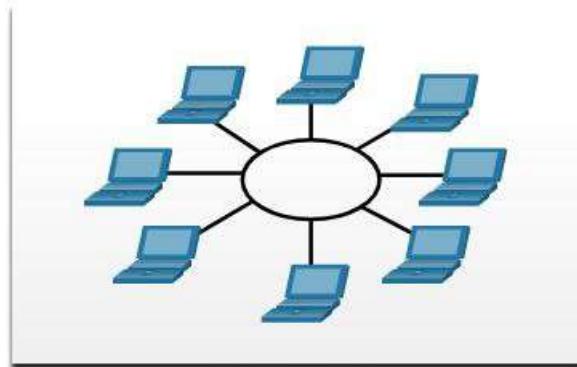
Star Topology



Extended Star Topology



Bus Topology



Ring Topology

Komunikasi Half Duplex dan Full Duplex

Memahami komunikasi **dupleks** penting ketika membahas **topologi LAN** karena mengacu pada arah transmisi data antara dua perangkat. Ada dua mode umum dupleks.

Komunikasi Half Duplex

Kedua perangkat dapat mengirimkan dan menerima di media tetapi tidak dapat melakukannya secara bersamaan. **WLAN** dan **legacy topologi bus** dengan hub Ethernet menggunakan mode half dupleks. Half-duplex hanya memungkinkan satu perangkat untuk mengirim atau menerima pada satu waktu pada media bersama.

Komunikasi Full Duplex

Kedua perangkat dapat secara bersamaan mengirimkan dan menerima di media bersama. Data Link Layer mengasumsikan bahwa media tersedia untuk transmisi untuk kedua node kapan saja. Switch Ethernet beroperasi dalam mode full duplex secara default, tetapi mereka dapat beroperasi dalam half dupleks jika terhubung ke perangkat seperti hub Ethernet.

Singkatnya, komunikasi half dupleks membatasi pertukaran data ke satu arah pada satu waktu. Full Dupleks memungkinkan pengiriman dan penerimaan data terjadi secara bersamaan.

Penting bahwa dua interface yang saling terhubung, seperti host NIC dan Interface pada Switch Ethernet, beroperasi menggunakan mode dupleks yang sama. Jika tidak, akan ada ketidakcocokan dupleks menciptakan inefisiensi dan latensi pada Link.

Access Control Methods

ETHERNET LAN dan **WLAN** adalah contoh jaringan **multi access**. Jaringan **multi access** adalah jaringan yang dapat memiliki dua atau lebih **end devices** yang mencoba mengakses jaringan secara bersamaan.

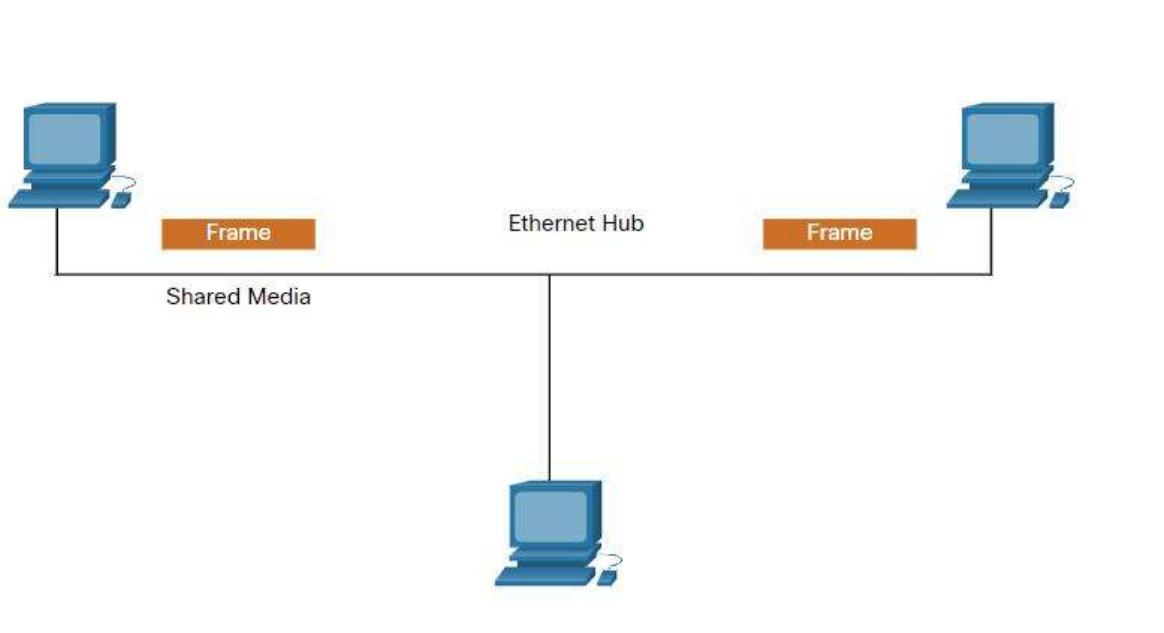
Beberapa jaringan **multi access** memerlukan aturan untuk mengatur bagaimana perangkat berbagi media fisik. Ada dua **metode kontrol akses** dasar untuk media bersama:

- Contention-based access
- Controlled access

Contention-based access

Dalam jaringan Contention-based access *multiaccess*, semua node beroperasi dalam half dupleks, bersaing untuk penggunaan media. Namun, hanya satu perangkat yang dapat mengirim pada satu waktu. Oleh karena itu, ada proses jika lebih dari satu perangkat mengirimkan pada saat yang sama. Contoh Contention-based access meliputi yang berikut ini:

- **Carrier sense multiple access with collision detection (CSMA/CD)** yang digunakan pada **legacy bus-topology**
- **Carrier sense multiple access with collision avoidance (CSMA/CA)** yang digunakan pada **WLAN**



Contention-Based Access – CSMA/CD

Contoh jaringan **Contention-Based Access** meliputi yang berikut ini:

- Wireless LAN (uses CSMA/CA)
- Legacy bus-topology Ethernet LAN (uses CSMA/CD)
- Legacy Ethernet LAN using a hub (uses CSMA/CD)

Jaringan ini beroperasi dalam mode **half dupleks**, yang berarti hanya satu perangkat yang dapat mengirim atau menerima pada satu waktu. Ini memerlukan proses untuk mengatur kapan perangkat dapat mengirim dan apa yang terjadi ketika beberapa perangkat mengirim pada saat yang sama.

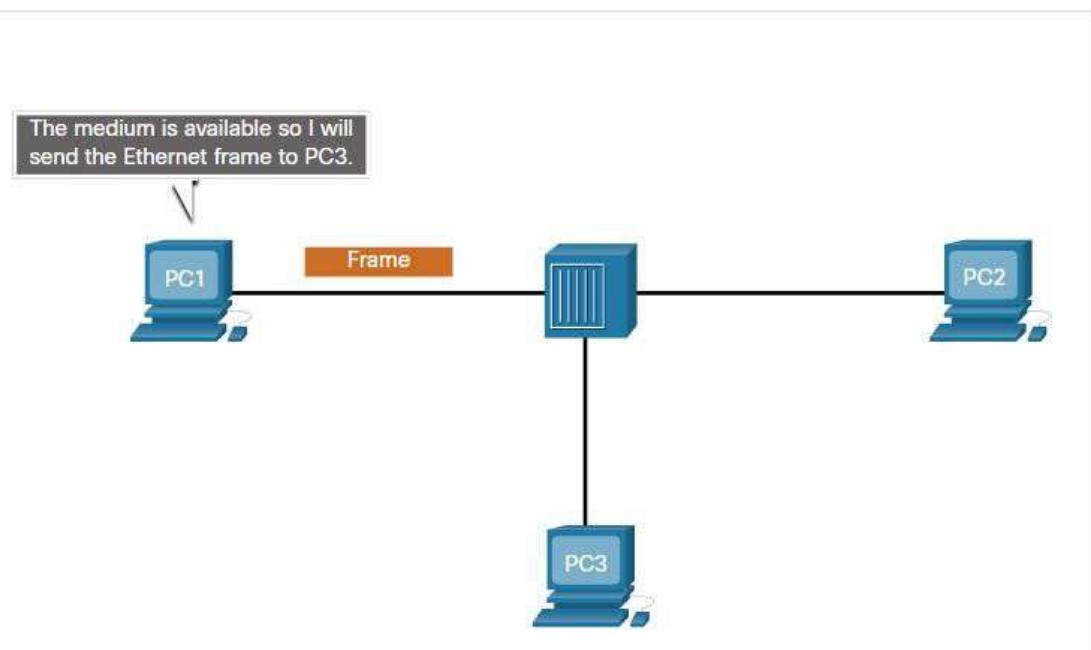
Jika dua perangkat mentransmisikan pada saat yang sama, **collision** akan terjadi. Untuk **ETHERNET** lama, kedua perangkat akan mendeteksi **collision** pada jaringan. Ini adalah bagian **collision detection (CD) CSMA / CD**. NIC membandingkan data yang dikirimkan dengan data yang diterima, atau dengan mengenali bahwa **amplitudo** sinyal lebih tinggi dari biasanya di media. Data yang dikirim oleh kedua perangkat akan rusak dan perlu dikirim ulang.

Proses CSMA/CD di **ETHERNET LAN** lama yang menggunakan hub. PC1 Mengirim frame hub Menerima Frame hub Mengirim frame

1. PC1 Mengirim Frame

PC1 memiliki frame Ethernet untuk dikirim ke PC3. NIC PC1 perlu menentukan apakah ada perangkat yang ditransmisikan pada media. Jika tidak mendeteksi sinyal pembawa (dengan kata lain, ia tidak menerima transmisi dari perangkat lain), itu akan mengasumsikan jaringan tersedia untuk dikirim.

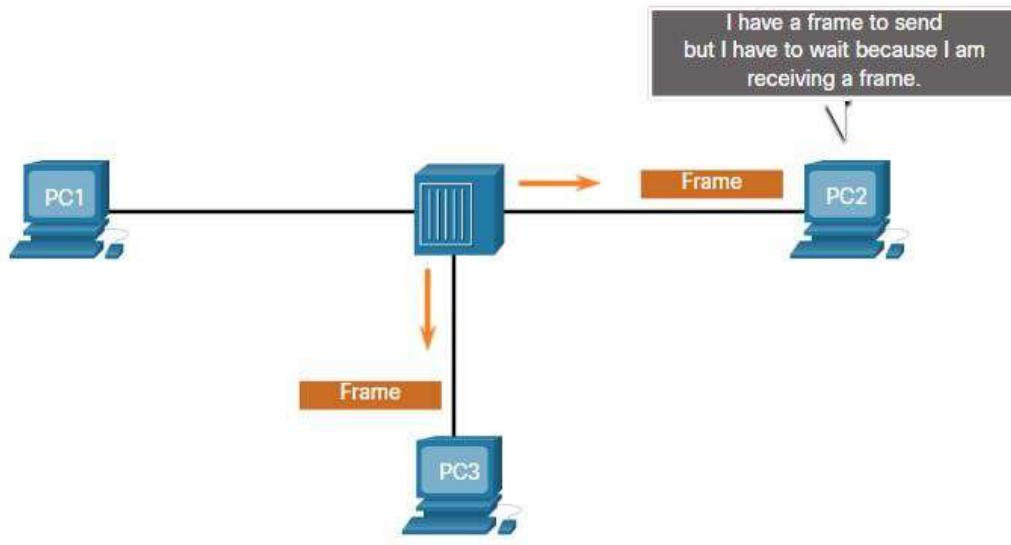
NIC PC1 mengirimkan Ethernet Frame ketika media tersedia, seperti yang ditunjukkan pada gambar.



2. Hub menerima Frame dari PC1

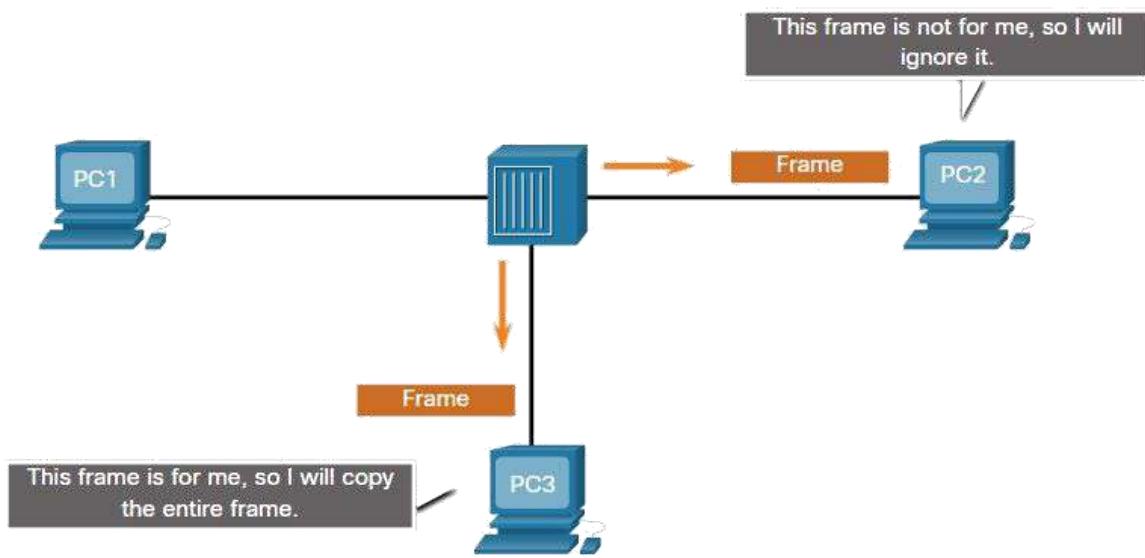
Hub Ethernet menerima dan mengirim **frame**. Hub Ethernet juga dikenal sebagai repeater multiport. Setiap bit yang diterima pada port yang masuk diregenerasi dan dikirim ke semua port lainnya, seperti yang ditunjukkan pada gambar.

Jika perangkat lain, seperti PC2, ingin mengirimkan, tetapi saat ini menerima frame, ia harus menunggu sampai saluran jelas, seperti yang ditunjukkan pada gambar.



3. Hub mengirim Frame

Semua perangkat yang terpasang pada hub akan menerima Frame. Namun, karena Frame memiliki Alamat Data Link tujuan untuk PC3, hanya perangkat yang akan menerima dan menyalin di seluruh Frame. Semua NIC perangkat lain akan mengabaikan Frame, seperti yang ditunjukkan pada gambar.

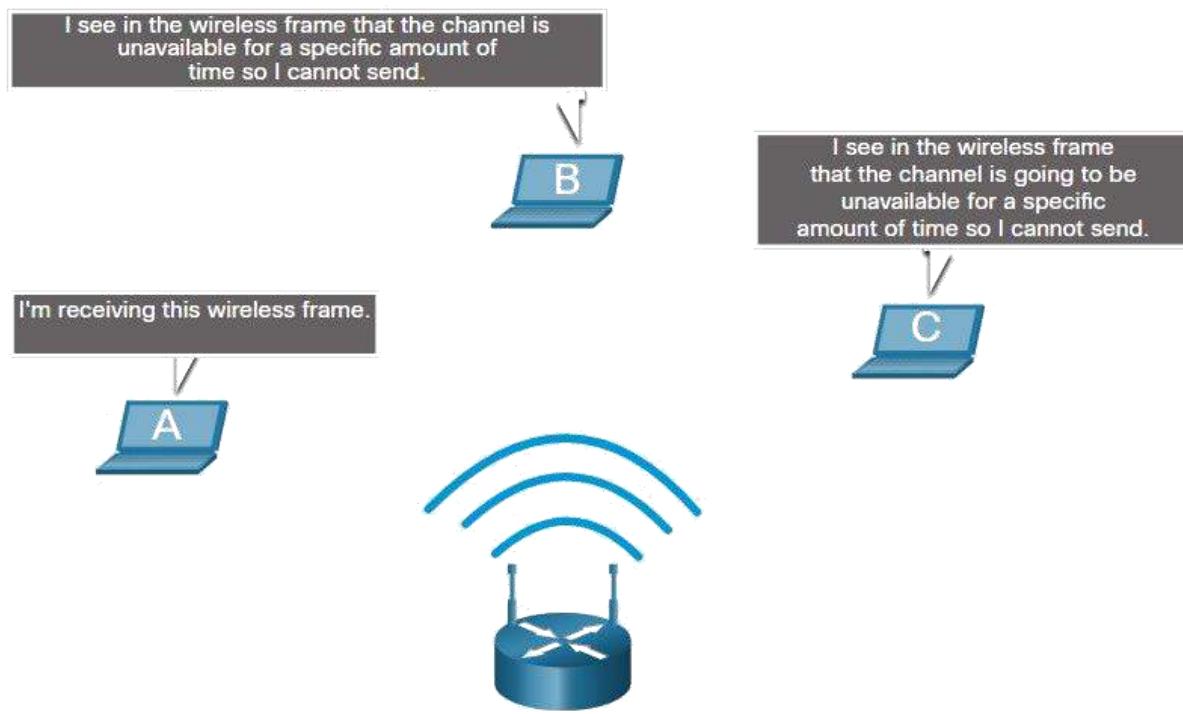


Contention-Based Access – CSMA/CA

Bentuk lain dari **CSMA** yang digunakan oleh **IEEE 802.11 WLANs** adalah **carrier sense multiple access/collision avoidance (CSMA/CA)**.

CSMA/CA menggunakan metode yang mirip dengan **CSMA/CD** untuk mendeteksi apakah media sudah tidak tersedia. **CSMA/CA** menggunakan teknik tambahan. Di lingkungan **wireless** mungkin tidak dimungkinkan bagi perangkat untuk mendeteksi **Collision**. **CSMA / CA** tidak mendeteksi **collision** tetapi berusaha menghindarinya *dengan menunggu sebelum mentransmisikan*. Setiap perangkat yang mengirimkan mencakup durasi waktu yang dibutuhkan untuk transmisi. Semua perangkat **wireless** lainnya menerima informasi ini dan mengetahui berapa lama media tidak tersedia.

Dalam gambar, jika host A menerima **Frame** nirkabel dari **Access Point**, host B, dan C juga akan melihat **Frame** dan berapa lama media tidak tersedia.



Setelah perangkat nirkabel mengirim **Frame 802.11**, penerima mengembalikan **acknowledgment** sehingga pengirim tahu **Frame** tiba.

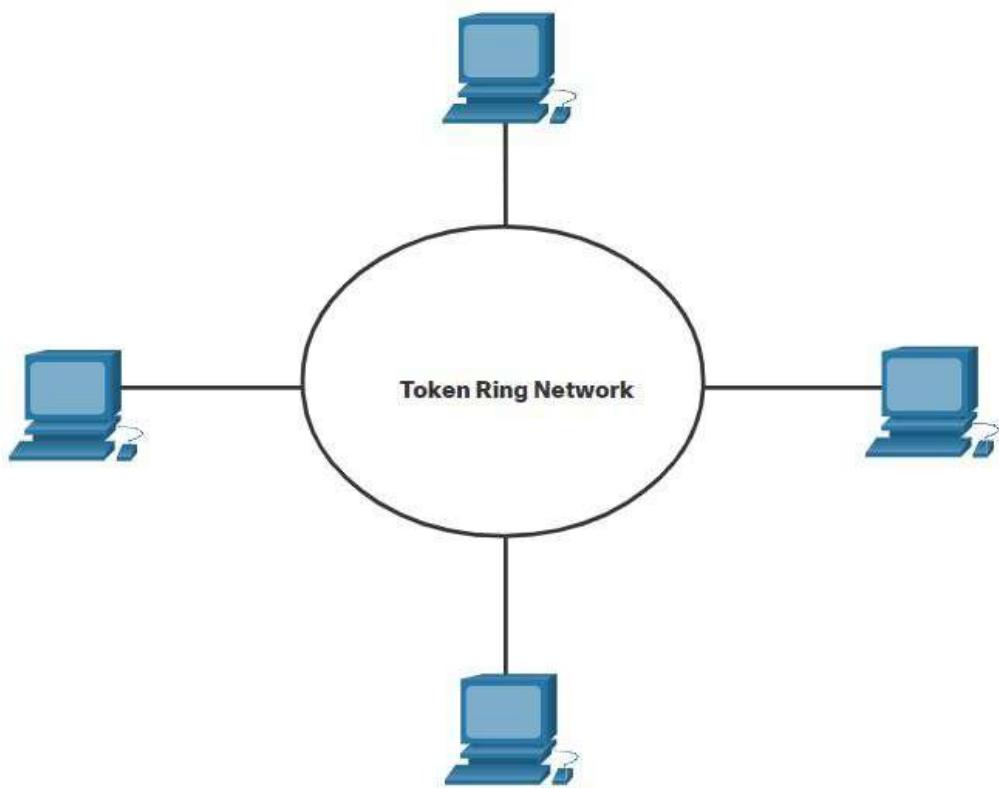
Topology LAN menggunakan ethernet dengan hub atau WLAN yang menggunakan sistem contention-based tidak dapat diskalakan dibawah penggunaan media yang berat

Catatan: ETHERNET yang menggunakan switch tidak menggunakan contention-based systems karena Switch dan host NIC beroperasi dalam mode Full Duplex.

Controlled access

Dalam jaringan **Controlled access multiaccess**, setiap **node** memiliki waktu tersendiri untuk menggunakan media. Jenis *deterministik* jaringan *legacy* ini tidak efisien karena perangkat harus menunggu gilirannya untuk mengakses media. Contoh jaringan **multi access** yang menggunakan **Control Access** meliputi yang berikut ini:

- Legacy Token Ring
- Legacy ARCNET



Setiap **node** harus menunggu gilirannya untuk mengakses media jaringan.

Catatan: Hari ini, jaringan Ethernet beroperasi dalam full duplex dan tidak memerlukan Access Method.

Frame Data Link

Materi ini membahas secara rinci apa yang terjadi pada **Frame Data Link** saat bergerak melalui jaringan. Informasi yang ditambahkan ke **frame** ditentukan oleh protokol yang sedang digunakan.

Frame

Data Link layer menyiapkan data yang dienkapsulasi (biasanya paket IPv4 atau IPv6) untuk transportasi di seluruh media lokal dengan merangkumnya dengan **header** dan **trailer** untuk membuat **Frame**.

Protokol **Data Link** bertanggung jawab atas komunikasi **NIC-ke-NIC** dalam jaringan yang sama. Meskipun ada banyak protokol **Data link layer** yang berbeda yang menjelaskan **Frame Data Link**, setiap jenis **Frame** memiliki tiga bagian dasar:

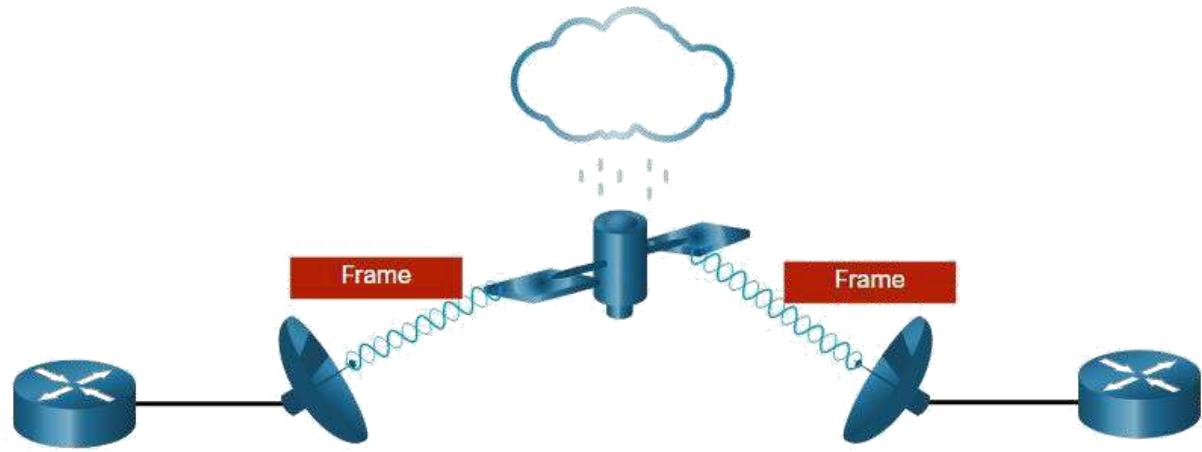
- Header
- Data
- Trailer

Tidak seperti protokol enkapsulasi lainnya, **Frame Data Link** menambahkan informasi dalam bentuk trailer di akhir **frame**.

Semua protokol **Frame Data Link** merangkum data dalam **fields** data **frame**. Namun, struktur **frame** dan **fields** yang terkandung dalam **header** dan **trailer** bervariasi sesuai dengan protokol.

Tidak ada satu struktur **frame** yang memenuhi kebutuhan semua transportasi data di semua jenis media. Bergantung pada *environment*, jumlah **Control Information** yang diperlukan dalam **frame** bervariasi agar sesuai dengan persyaratan **kontrol akses dari media yang digunakan** dan **logical topology**. Misalnya, frame **WLAN** harus menyertakan prosedur untuk menghindari **Collision** dan karenanya memerlukan informasi kontrol tambahan jika dibandingkan dengan **frame Ethernet**.

Seperti yang ditunjukkan pada gambar, dalam *environment* yang rapuh, lebih banyak kontrol diperlukan untuk memastikan pengiriman. **fields header** dan **trailer** lebih besar karena **Control Information** lebih banyak diperlukan.

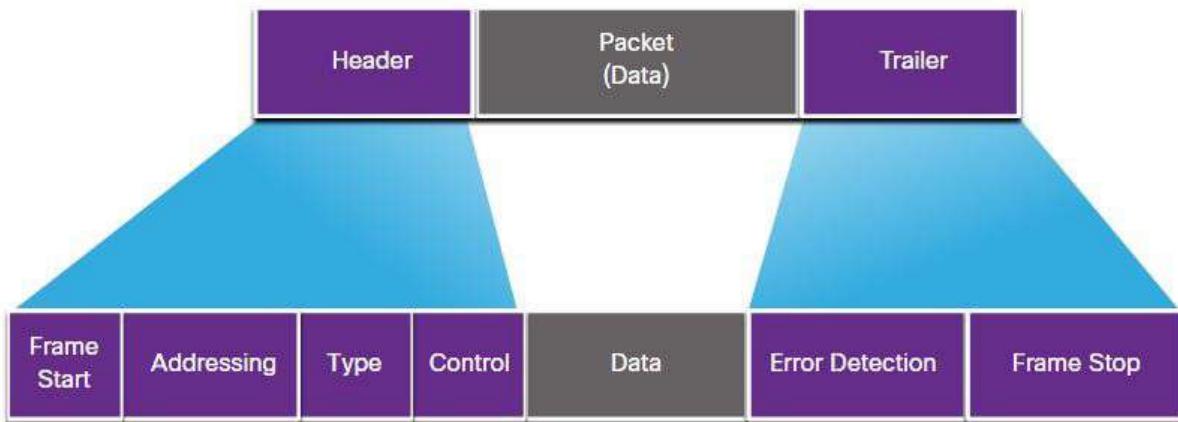


Diperlukan upaya yang lebih besar untuk memastikan pengiriman. Ini berarti **overhead** yang lebih tinggi dan tingkat transmisi yang lebih lambat.

Frame Fields

Framing memecah aliran menjadi pengelompokan yang dapat diuraikan, dengan **control information** yang disisipkan di **header** dan **trailer** sebagai nilai di **field** yang berbeda. Format ini memberikan sinyal fisik struktur yang diakui oleh **node** dan **decode** ke dalam paket di tujuan.

Frame Fields generik ditampilkan dalam gambar. Tidak semua protokol mencakup semua **fields** ini. Standar untuk protokol **data link** tertentu menentukan **frame format** yang aktual.



Frame Fields meliputi yang berikut ini:

- **Frame start and stop indicator flags** – Digunakan untuk mengidentifikasi batas awal dan akhir **frame**.
- **Addressing** – Menunjukkan **node** sumber dan tujuan pada media.
- **Type** – Mengidentifikasi protokol Layer 3 di **field** data.
- **Control** – Mengidentifikasi layanan kontrol aliran khusus seperti **Quality Of Services (QoS)**. QoS memberikan prioritas penerusan pada jenis pesan tertentu. Misalnya, **Frame voice over IP (VoIP)** biasanya menerima prioritas karena sensitif terhadap *delay*.
- **Data** – Berisi **Frame Payloads** (yaitu, **packet header**, **segment header**, dan data).
- **Error Detection** – Disertakan setelah data untuk membentuk **trailer**.

Protokol **Data Link Layer** menambahkan **trailer** ke akhir setiap **Frame**. Dalam proses yang disebut **error detection**, **trailer** menentukan apakah **frame** tiba tanpa kesalahan. Ini menempatkan **logical summary** atau matematika dari bit yang terdiri dari **frame** dalam trailer. **data link layer** menambahkan **error detection** karena sinyal di media dapat mengalami gangguan, distorsi, atau kehilangan yang secara substansial akan mengubah nilai bit yang diwakili sinyal tersebut.

Node transmisi membuat **logical summary** dari isi **frame**, yang dikenal sebagai nilai **cyclic redundancy check (CRC)**. Nilai ini ditempatkan di **field frame check sequence (FCS)** untuk mewakili isi **frame**. Dalam **trailer Ethernet**, FCS menyediakan metode untuk **node** penerima untuk menentukan apakah **frame** mengalami kesalahan transmisi.

Layer 2 Addresses

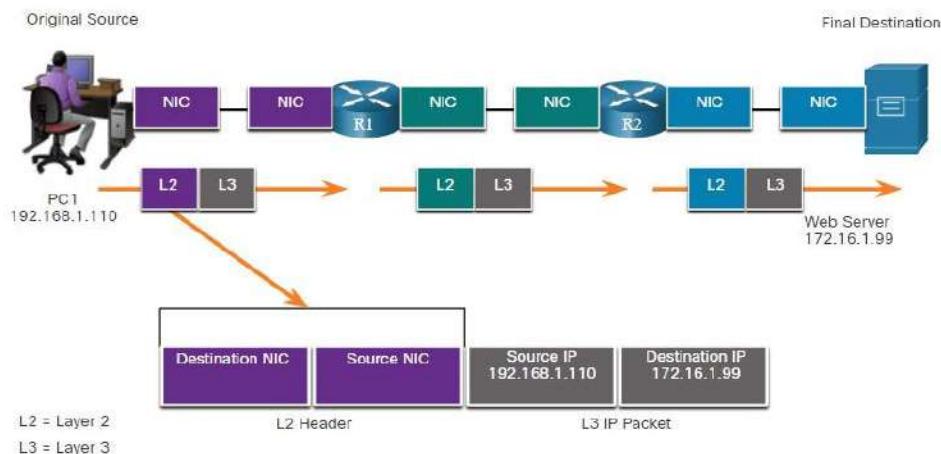
Data Link Layer menyediakan alamat yang digunakan dalam mengangkut **Frame** di media lokal bersama. Alamat perangkat pada **layer** ini disebut sebagai **alamat fisik**. Data link layer terkandung dalam **frame header** dan menentukan **node** tujuan **frame** pada jaringan lokal. Biasanya di awal **frame**, sehingga **NIC** dapat dengan cepat menentukan apakah cocok dengan alamat **Layer 2** sendiri sebelum menerima sisa **frame**. **Frame Header** mungkin juga berisi alamat sumber **frame**.

Tidak seperti **Logical address Layer 3**, yang *hierarkis*, alamat fisik tidak menunjukkan pada jaringan apa perangkat berada. Sebaliknya, alamat fisiknya unik untuk perangkat tertentu. Perangkat masih akan berfungsi dengan alamat fisik **Layer 2** yang sama meskipun perangkat berpindah ke jaringan atau subnet lain. Oleh karena itu, alamat Layer 2 hanya digunakan untuk menghubungkan perangkat dalam **shared media** yang sama, pada jaringan **IP** yang sama.

Gambar-gambar menggambarkan fungsi alamat Layer 2 dan Layer 3. Ketika paket IP bepergian dari host-ke-router, router-to-router, dan akhirnya router-to-host, di setiap node di sepanjang jalan paket IP dienkapsulasi dalam frame data link baru. Setiap frame data link berisi **source data link address** dari yang mengirim frame, dan **destination data link address** yang menerima frame.

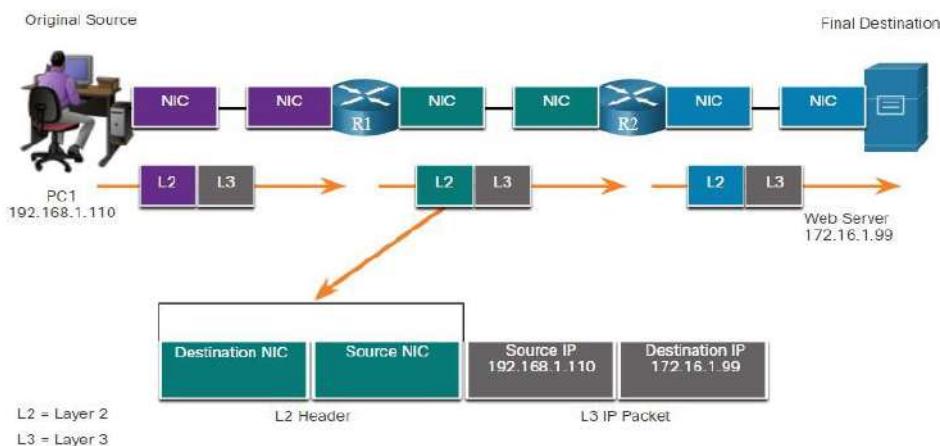
Host-to-Router

Original Source merangkum paket IP Layer 3 dalam **Frame** Layer 2. Di **Frame header**, host menambahkan alamat **Layer 2**-nya sebagai source dan alamat **Layer 2** untuk R1 sebagai destination.



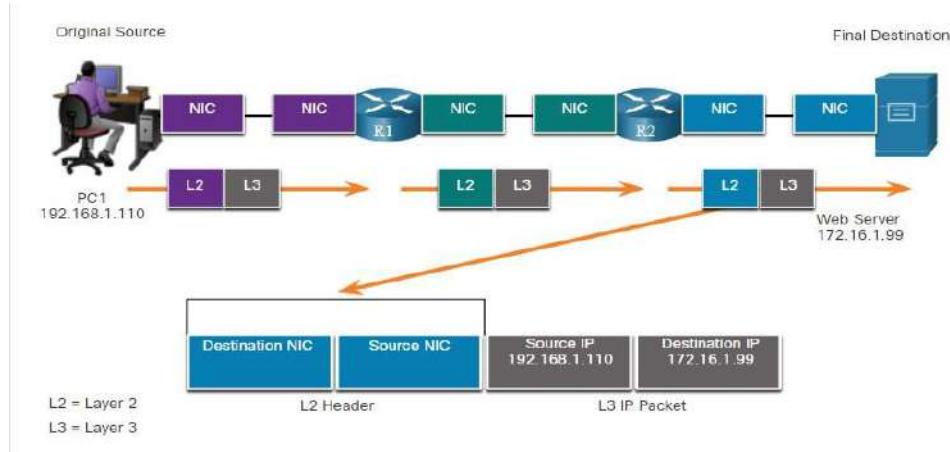
Router-To-Router

R1 mengenkapsulasi paket Layer 3 IP dalam **Frame** Layer 2 baru. Di header frame, R1 menambahkan alamat Layer 2 sebagai source dan alamat Layer 2 untuk R2 sebagai destination.



Router-To-Host

R2 mengenkapsulasi paket Layer 3 IP dalam **frame** Layer 2 baru. Di **header frame**, R2 menambahkan alamat Layer 2 sebagai source dan alamat Layer 2 untuk server sebagai destination.



Alamat **Data link layer** hanya digunakan untuk pengiriman lokal. Alamat pada layer ini tidak memiliki arti di luar jaringan lokal. Bandingkan ini dengan Layer 3, di mana alamat di header paket dibawa dari host sumber ke host tujuan, terlepas dari jumlah hop jaringan di sepanjang rute.

Jika data harus diteruskan ke **segment** jaringan lain, **Intermediary Devices**, seperti **router**, diperlukan. **Router** harus menerima **Frame** berdasarkan alamat fisik dan de-enkapsulasi **frame** untuk memeriksa alamat **hierarkis**, yang merupakan alamat **IP**. Dengan menggunakan alamat **IP**, **router** dapat menentukan lokasi jaringan perangkat tujuan dan jalur terbaik untuk mencapainya. Ketika tahu di mana untuk meneruskan **paket**, router kemudian membuat **frame** baru untuk paket, dan **frame** baru dikirim ke **segmen** jaringan berikutnya menuju tujuan akhir.

Frame LAN dan WAN

Protokol **Ethernet** digunakan oleh **Kabel LAN**. Komunikasi **wireless** berada di bawah protokol **WLAN** (IEEE 802.11). Protokol ini dirancang untuk jaringan **multi akses**.

WAN secara tradisional menggunakan jenis protokol lain untuk berbagai jenis topologi **point-to-point**, **hub-spoke**, dan **full-mesh**. Beberapa protokol **WAN** umum selama bertahun-tahun telah mencakup:

- Point-to-Point Protocol (PPP)
- High-Level Data Link Control (HDLC)
- Frame Relay
- Asynchronous Transfer Mode (ATM)
- X.25

Protokol Layer 2 ini sekarang diganti di **WAN** oleh **Ethernet**.

Dalam jaringan **TCP/IP**, semua protokol OSI Layer 2 bekerja dengan **IP** di OSI Layer 3. Namun, protokol Layer 2 yang digunakan tergantung pada **topologi logis** dan **media fisik**.

Setiap protokol melakukan **kontrol akses media** untuk **topologi logis Layer 2** tertentu. Ini berarti bahwa sejumlah perangkat jaringan yang berbeda dapat bertindak sebagai **node** yang beroperasi pada **Data Link Layer** saat menerapkan protokol ini. Perangkat ini termasuk **NIC** pada komputer serta **interface** pada **router** dan **switch Layer 2**.

Protokol Layer 2 yang digunakan untuk **topologi** jaringan tertentu ditentukan oleh teknologi yang digunakan untuk mengimplementasikan **topologi** tersebut. **Teknologi** yang digunakan ditentukan oleh ukuran jaringan, dalam hal jumlah host dan lingkup geografis, dan layanan yang akan disediakan melalui jaringan.

LAN biasanya menggunakan teknologi bandwidth tinggi yang mampu mendukung sejumlah besar host. Area geografis LAN yang relatif kecil (bangunan tunggal atau kampus multi-bangunan) dan kepadatan pengguna yang tinggi membuat teknologi ini hemat biaya.

Namun, menggunakan teknologi bandwidth tinggi biasanya tidak hemat biaya untuk **WAN** yang mencakup area geografis besar (kota atau beberapa kota, misalnya). Biaya **link fisik** jarak jauh dan teknologi yang digunakan untuk membawa sinyal di atas jarak tersebut biasanya menghasilkan kapasitas bandwidth yang lebih rendah.

Perbedaan bandwidth biasanya menghasilkan penggunaan protokol yang berbeda untuk **LAN** dan **WAN**.

Protokol lapisan Data link meliputi:

- Ethernet
- 802.11 Wireless
- Point-to-Point Protocol (PPP)
- High-Level Data Link Control (HDLC)
- Frame Relay

BAB 7

~ *Ethernet Switching* ~

Judul Bab : Ethernet Switching

Tujuan Bab : Menjelaskan bagaimana ethernet bekerja di switched network

Link Test Pemahaman : <https://forms.gle/yPJ4L4yecKmE5Z4Q8>

| Judul Materi | Tujuan Materi |
|--------------------------------------|--|
| Ethernet Frame | Menjelaskan bagaimana ethernet sublayers terkait pada frame fields |
| Ethernet MAC Address | Menjelaskan Ethernet MAC Address\ |
| MAC Address Table | Menjelaskan bagaimana switch membuat mac address table dan meneruskan frames |
| Switch Speeds dan Forwarding methods | Menjelaskan metode switch forwarding dan setting port yang tersedia di layer 2 switch port |

Ethernet Frame

Di Materi ini dimulai dengan diskusi teknologi Ethernet termasuk penjelasan sublayer MAC dan Ethernet Frame fields .

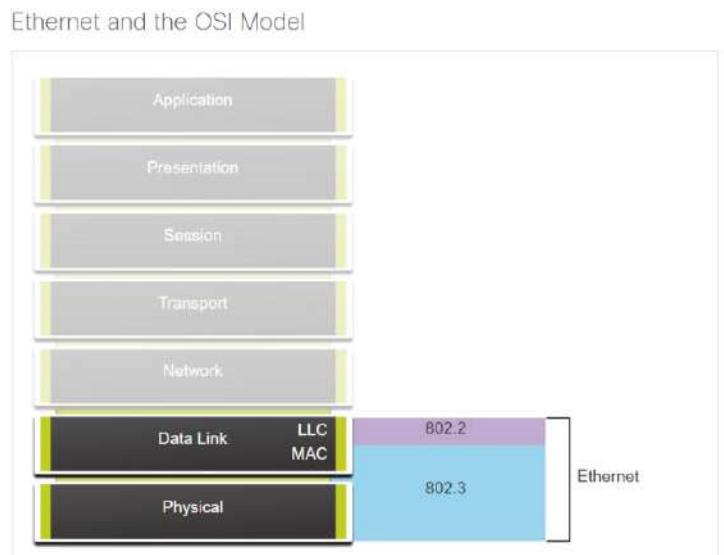
Enkapsulasi Ethernet

Ethernet adalah salah satu dari dua teknologi LAN yang digunakan saat ini, dengan yang lain adalah Wireless LAN (WLAN). Ethernet menggunakan komunikasi kabel, termasuk twisted pair, fiber-optic links, and coaxial cables.

Ethernet beroperasi di data link layer dan Physical Layer. Ini adalah keluarga teknologi jaringan yang didefinisikan dalam standar **IEEE 802.2** dan **802.3**. Ethernet mendukung bandwidth data berikut:

- 10 Mbps
- 100 Mbps
- 1000 Mbps (1 Gbps)
- 10.000 Mbps (10 Gbps)
- 40.000 Mbps (40 Gbps)
- 100.000 Mbps (100 Gbps)

Seperti yang ditunjukkan pada gambar, standar Ethernet mendefinisikan protokol Layer 2 dan teknologi Layer 1.

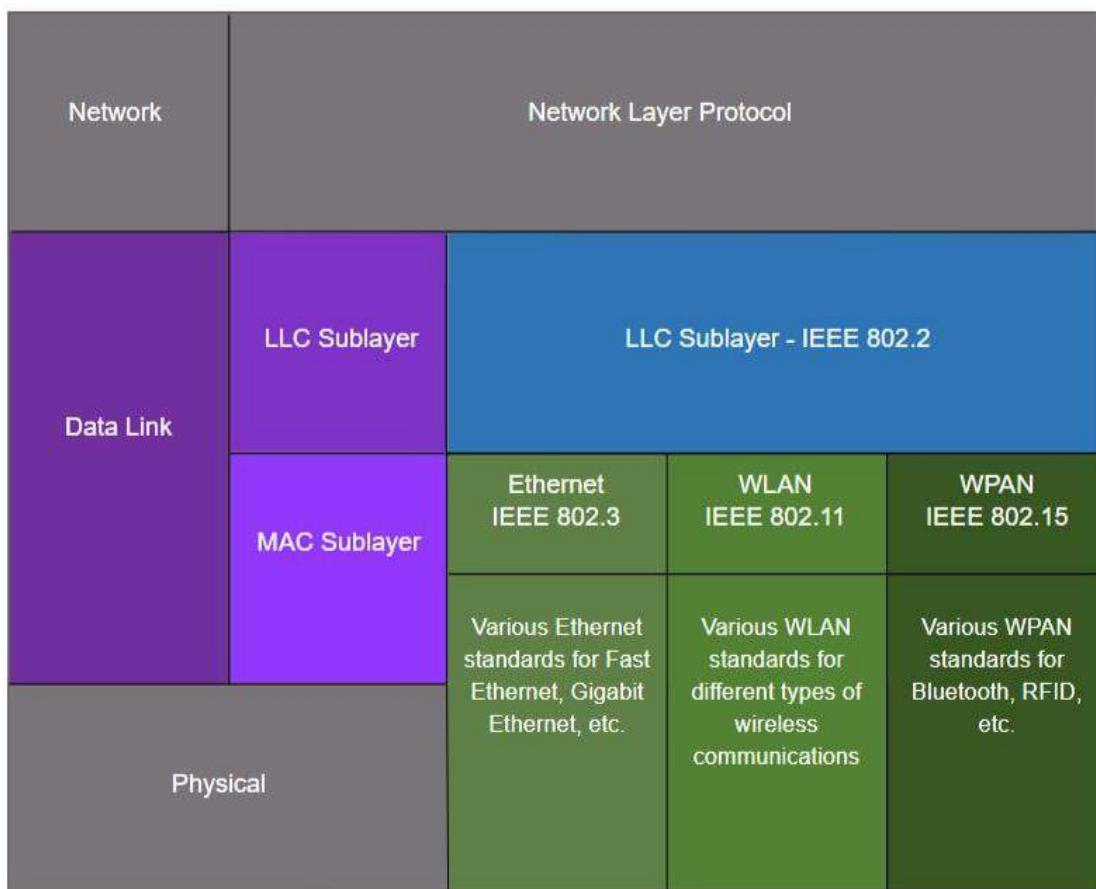


Sublayer Data Link

Protokol LAN/MAN IEEE 802, termasuk Ethernet, menggunakan dua sublayer terpisah berikut dari data link layer untuk beroperasi. Mereka adalah **Logical Link Control (LLC)** dan **Media Access Control (MAC)**, seperti yang ditunjukkan pada gambar.

Ingat bahwa LLC dan MAC memiliki peran berikut dalam Data Link Layer:

- **Sublayer LLC – Sublayer IEEE 802.2** ini berkomunikasi antara perangkat lunak jaringan di upper layer dan perangkat keras perangkat di bottom layer. Ini menempatkan informasi dalam frame yang mengidentifikasi protokol layer jaringan mana yang digunakan untuk frame. Informasi ini memungkinkan beberapa protokol Layer 3, seperti IPv4 dan IPv6, untuk menggunakan interface dan media jaringan yang sama.
- **MAC Sublayer – Sublayer ini (IEEE 802.3, 802.11, atau 802.15 misalnya)** diimplementasikan dalam perangkat keras dan bertanggung jawab atas enkapsulasi data dan **kontrol akses media**. Ini menyediakan mengatasi **Data Link Layer** dan terintegrasi dengan berbagai teknologi lapisan fisik.



MAC Sublayer

Sublayer MAC bertanggung jawab atas enkapsulasi data dan mengakses media.

Enkapsulasi Data

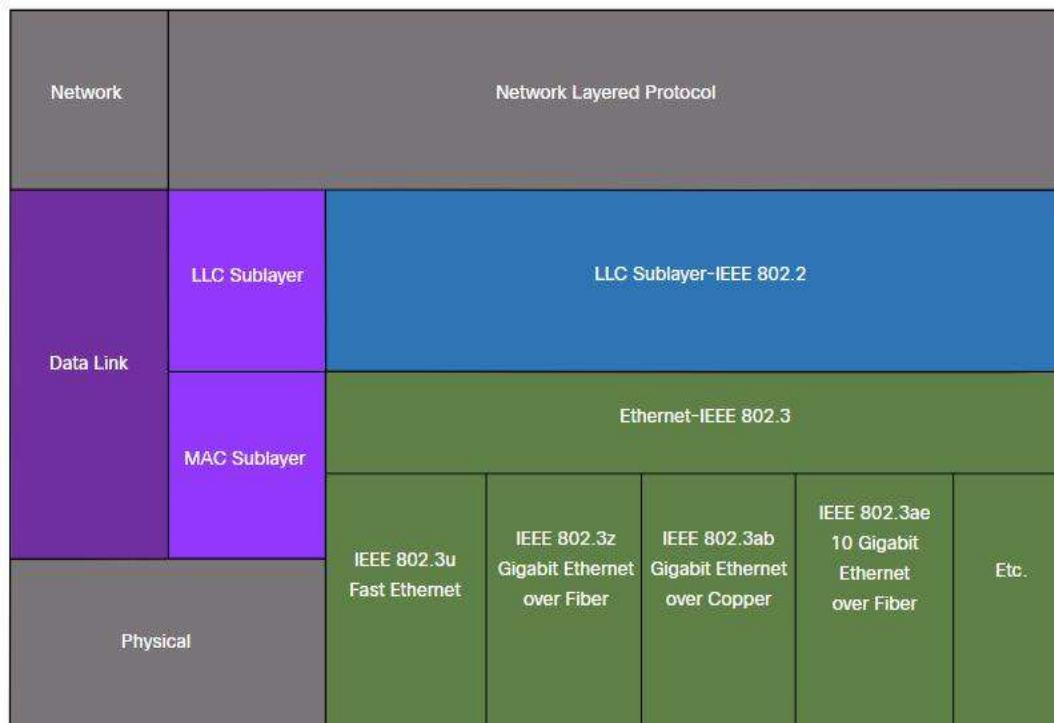
Enkapsulasi data IEEE 802.3 mencakup hal-hal berikut:

- **Frame Ethernet** – Ini adalah struktur internal Ethernet Frame.
- **Address Ethernet** – Frame Ethernet menyertakan source mac address dan destination untuk mengirimkan **frame Ethernet** dari **Ethernet NIC** ke **Ethernet NIC** pada LAN yang sama.
- **Error Detection Ethernet** – **Frame Ethernet** menyertakan **trailer frame check sequence (FCS)** yang digunakan untuk deteksi kesalahan.

Media Access

Seperti yang ditunjukkan pada gambar, sublayer IEEE 802.3 MAC Termasuk spesifikasi untuk berbagai standar komunikasi Ethernet atas berbagai jenis media termasuk tembaga dan serat optic.

Standar Ethernet Di MAC Sublayer



Ingatlah bahwa **Legacy Ethernet** menggunakan topologi atau hub bus, adalah shared media yang half duplex. Ethernet lebih dari half-duplex medium menggunakan contention-based access method, carrier sense multiple access/collision detection (CSMA/CD). Ini memastikan bahwa hanya satu perangkat yang mentransmisikan pada satu waktu. CSMA/CD memungkinkan beberapa perangkat untuk berbagi media yang sama, mendeteksi **collision** ketika lebih dari satu perangkat mencoba untuk mengirimkan secara bersamaan. Ini juga menyediakan algoritma *back-off* untuk retransmission.

ETHERNET LAN saat ini menggunakan **Switch** yang beroperasi dalam **Full Duplex**. Komunikasi **full-duplex** dengan **switch Ethernet** tidak memerlukan **kontrol akses** melalui **CSMA/CD**.

Ethernet Frame Fields

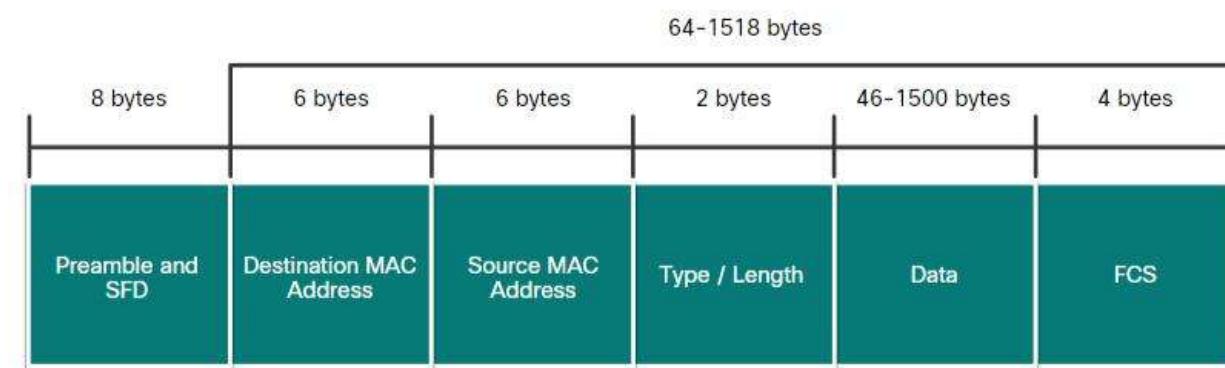
Ukuran **Frame Ethernet** minimum adalah 64 byte dan maksimum yang diharapkan adalah 1518 byte. Ini termasuk semua byte dari field destination mac address melalui frame check sequence (FCS) . **Preamble Fields** tidak disertakan saat menggambarkan ukuran **Frame**.

Catatan: Ukuran Frame mungkin lebih besar jika persyaratan tambahan disertakan, seperti VLAN Tagging.

Setiap **Frame** dengan panjang kurang dari 64 byte dianggap sebagai “**collision fragment**” atau “**runt frame**” dan secara otomatis dibuang dengan menerima stasiun. **Frame** dengan lebih dari 1500 byte data dianggap “**jumbo**” atau “**baby giant frame**”.

Jika ukuran **Frame** yang ditransmisikan kurang dari minimum, atau lebih besar dari maksimum, perangkat penerima me-**Drop**-kan **Frame**. Frame yang akan di **drop** kemungkinan merupakan hasil dari **collision** atau sinyal lain yang tidak diinginkan. Mereka dianggap tidak sah. **frame jumbo** biasanya didukung oleh sebagian besar **switch FastEthernet** dan **Gigabit Ethernet** dan **NIC**.

Gambar menunjukkan setiap **fields** dalam **frame Ethernet**.



Detail field frame Ethernet

| Fields | Deskripsi |
|--|---|
| Preamble and Start Frame Delimiter Fields | Preamble (7 byte) dan Start Frame Delimiter (SFD), juga disebut fields Start of Frame (1 byte), digunakan untuk sinkronisasi antara perangkat pengirim dan penerimaan. Delapan byte pertama frame ini digunakan untuk mendapatkan perhatian dari node penerima. Pada dasarnya, beberapa byte pertama memberi tahu penerima untuk bersiap-siap untuk menerima frame baru. |
| Destination MAC Address Field | Fields 6 byte ini adalah pengidentifikasi untuk penerima yang dituju. Seperti yang akan Anda ingat, alamat ini digunakan oleh Layer 2 untuk membantu perangkat dalam menentukan apakah Frame ditujukan kepada mereka. Alamat dalam Frame dibandingkan dengan alamat MAC di perangkat. Jika ada kecocokan, perangkat akan menerima Frame. Bisa menjadi alamat unicast, multicast, atau broadcast |
| Source MAC Address Field | Fields 6-byte ini mengidentifikasi NIC yang berasal atau Frame Interface. |
| Type / Length | Fields 2-byte ini mengidentifikasi protokol upper layer yang dienkapsulasi dalam Ethernet Frame. Nilai yang sama adalah, dalam heksadesimal, 0x800 untuk IPv4, 0x86DD untuk IPv6 dan 0x806 untuk ARP. Catatan: Anda mungkin juga melihat Fields ini disebut sebagai EtherType, Type, atau Length. |
| Data Fields | Fields ini (46 – 1500 byte) berisi data yang dienkapsulasi dari Layer yang lebih tinggi, yang merupakan Layer 3 PDU generik, atau lebih umum, paket IPv4. Semua Frame harus panjangnya minimal 64 byte. Jika paket kecil dienkapsulasi, bit tambahan yang disebut bantalan/pad digunakan untuk meningkatkan ukuran Frame ke ukuran minimum ini. |
| Frame Check Sequence Field | Fields Frame Check Sequence (FCS) (4 byte) digunakan untuk mendeteksi kesalahan dalam Frame. Ini menggunakan cyclic redundancy check (CRC). Perangkat pengirim menyertakan hasil CRC di Fields Frame FCS . Perangkat penerima menerima Frame dan menghasilkan CRC untuk mencari kesalahan. Jika perhitungan cocok, tidak ada kesalahan yang terjadi. Perhitungan yang tidak cocok adalah indikasi bahwa data telah berubah; oleh karena itu, bingkai didrop. Perubahan data bisa menjadi akibat dari gangguan sinyal listrik yang mewakili bit. |

Ethernet MAC Address

Dalam jaringan, alamat IPv4 diwakili menggunakan sistem angka sepuluh basis desimal dan sistem nomor basis biner 2. Alamat IPv6 dan alamat Ethernet diwakili menggunakan sistem angka enam belas basis heksadesimal. Untuk memahami heksadesimal, Pertama-tama Anda harus sangat akrab dengan biner dan desimal.

MAC Address dan Heksadesimal

Sistem penomoran heksadesimal menggunakan angka 0 hingga 9 dan huruf A hingga F.

MAC Address Ethernet terdiri dari nilai biner 48-bit. Heksadesimal digunakan untuk mengidentifikasi alamat Ethernet karena satu digit heksadesimal mewakili empat bit biner. Oleh karena itu, Ethernet MAC Address 48-bit dapat diekspresikan hanya menggunakan nilai 12 heksadesimal.

Angka tersebut membandingkan nilai desimal dan heksadesimal yang setara untuk biner 0000 hingga 1111.

Desimal dan Setara Biner 0 hingga F Heksadesimal

| Decimal | Binary | Hexadecimal |
|---------|--------|-------------|
| 0 | 0000 | 0 |
| 1 | 0001 | 1 |
| 2 | 0010 | 2 |
| 3 | 0011 | 3 |
| 4 | 0100 | 4 |
| 5 | 0101 | 5 |
| 6 | 0110 | 6 |
| 7 | 0111 | 7 |
| 8 | 1000 | 8 |
| 9 | 1001 | 9 |
| 10 | 1010 | A |
| 11 | 1011 | B |
| 12 | 1100 | C |
| 13 | 1101 | D |
| 14 | 1110 | E |
| 15 | 1111 | F |

Mengingat bahwa 8 bit (satu byte) adalah pengelompokan biner umum, biner 00000000 hingga 11111111 dapat diwakili dalam heksadesimal sebagai rentang 00 hingga FF, seperti yang ditunjukkan pada gambar berikutnya.

Setara Desimal, Biner, dan Heksadesimal Terpilih

| Decimal | Binary | Hexadecimal |
|---------|-----------|-------------|
| 0 | 0000 0000 | 00 |
| 1 | 0000 0001 | 01 |
| 2 | 0000 0010 | 02 |
| 3 | 0000 0011 | 03 |
| 4 | 0000 0100 | 04 |
| 5 | 0000 0101 | 05 |
| 6 | 0000 0110 | 06 |
| 7 | 0000 0111 | 07 |
| 8 | 0000 1000 | 08 |
| 10 | 0000 1010 | 0A |
| 15 | 0000 1111 | 0F |
| 16 | 0001 0000 | 10 |
| 32 | 0010 0000 | 20 |
| 64 | 0100 0000 | 40 |
| 128 | 1000 0000 | 80 |
| 192 | 1100 0000 | C0 |
| 202 | 1100 1010 | CA |
| 240 | 1111 0000 | F0 |
| 255 | 1111 1111 | FF |

Saat menggunakan heksadesimal, nol di depan selalu ditampilkan untuk menyelesaikan representasi **8-bit**. Misalnya, dalam tabel, nilai **biner 0000 1010** ditampilkan dalam heksadesimal sebagai **0A**.

Angka heksadesimal sering diwakili oleh nilai yang didahului oleh 0x (misalnya, 0x73) untuk membedakan antara nilai desimal dan heksadesimal dalam dokumentasi.

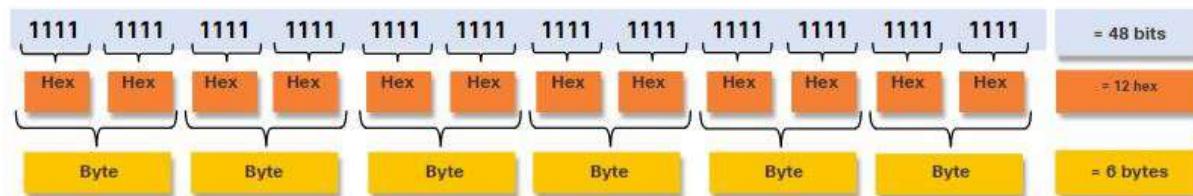
Heksadesimal juga dapat diwakili oleh subskrip 16, atau nomor heksa diikuti oleh H (misalnya, 73H).

Anda mungkin harus mengonversi antara nilai desimal dan heksadesimal. Jika konversi tersebut diperlukan, konversikan nilai desimal atau heksadesimal ke biner, lalu konversikan nilai biner ke desimal atau heksadesimal sebagaimana mestinya.

Ethernet MAC Address

Di LAN Ethernet, setiap perangkat jaringan terhubung ke shared media yang sama. MAC Address digunakan untuk mengidentifikasi sumber fisik dan perangkat tujuan (NIC) pada segmen jaringan lokal. MAC Address menyediakan metode untuk identifikasi perangkat pada Data Link Layer.

Ethernet MAC Address adalah alamat 48-bit yang dinyatakan menggunakan 12 digit heksadesimal, seperti yang ditunjukkan pada gambar. Karena byte sama dengan 8 bit, kita juga dapat mengatakan bahwa alamat MAC adalah 6 byte panjangnya.

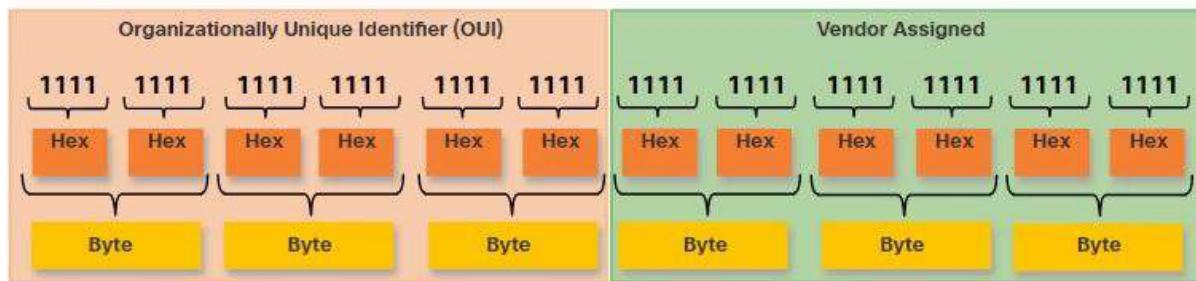


Semua MAC Address harus unik untuk perangkat Ethernet atau Interface Ethernet. Untuk memastikan hal ini, semua vendor yang menjual perangkat Ethernet harus mendaftar ke **IEEE** untuk mendapatkan kode unik 6 heksadesimal (yaitu, 24-bit atau 3-byte) yang disebut **organizationally unique identifier (OUI)**.

Saat vendor menetapkan **MAC Address** ke perangkat atau **Ethernet Interface**, vendor harus melakukan sebagai berikut:

- Gunakan **OUI** yang ditetapkan sebagai 6 digit **heksadesimal** pertama.
- Tetapkan nilai unik dalam 6 digit **heksadesimal** terakhir.

Oleh karena itu, **Ethernet MAC Address** terdiri dari kode **OUI** vendor 6 heksadesimal diikuti oleh nilai yang ditetapkan vendor 6 heksadesimal , seperti yang ditunjukkan pada gambar.



Misalnya, asumsikan bahwa Cisco perlu menetapkan **MAC Address unik** ke perangkat baru. IEEE telah menugaskan Cisco **OUI 00-60-2F**. Cisco kemudian akan mengkonfigurasi perangkat dengan kode vendor unik seperti **3A-07-BC**. Oleh karena itu, **Ethernet MAC Address** dari perangkat itu adalah **00-60-2F-3A-07-BC**. Mengecek mac address bisa melalui situs <https://macvendors.com/>

Tanggung jawab vendor untuk memastikan bahwa tidak ada perangkatnya yang diberi **MAC Address** yang sama. Namun, dimungkinkan agar **MAC Address** duplikat ada karena kesalahan yang dilakukan selama manufaktur, kesalahan yang dilakukan dalam beberapa metode implementasi mesin virtual, atau modifikasi yang dilakukan menggunakan salah satu dari beberapa alat perangkat lunak. Bagaimanapun, perlu untuk memodifikasi alamat **MAC** dengan **NIC** baru atau melakukan modifikasi melalui perangkat lunak.

Frame Processing

Terkadang MAC Address disebut sebagai **Burned-In Address (BIA)** karena alamatnya hard coded ke dalam Read-Only Memory (ROM) pada NIC. Ini berarti bahwa alamat dikodekan ke dalam chip ROM secara permanen.

Catatan: Pada sistem operasi PC modern dan **NIC**, dimungkinkan untuk mengubah alamat **MAC** dalam perangkat lunak. Ini berguna ketika mencoba untuk mendapatkan akses ke jaringan yang memfilter berdasarkan **BIA**. Akibatnya, pemfilteran atau pengendalian lalu lintas berdasarkan alamat **MAC** tidak lagi aman.

Ketika komputer Menyala, NIC menyalin alamat MAC-nya dari ROM ke RAM. Ketika perangkat meneruskan pesan ke jaringan Ethernet, header Ethernet menyertakan ini:

- **Source MAC address** – Ini adalah MAC Address dari perangkat sumber NIC.
- **Destination MAC address** – Ini adalah MAC Address perangkat tujuan NIC.

Ketika NIC menerima Frame Ethernet, ia memeriksa Destination MAC Address untuk melihat apakah cocok dengan MAC Address fisik yang disimpan dalam RAM. Jika tidak ada kecocokan, perangkat akan membuang **Frame**. Jika ada kecocokan, ia melewati **Frame** ke **Upper Layer**, di mana proses **de-enkapsulasi** berlangsung.

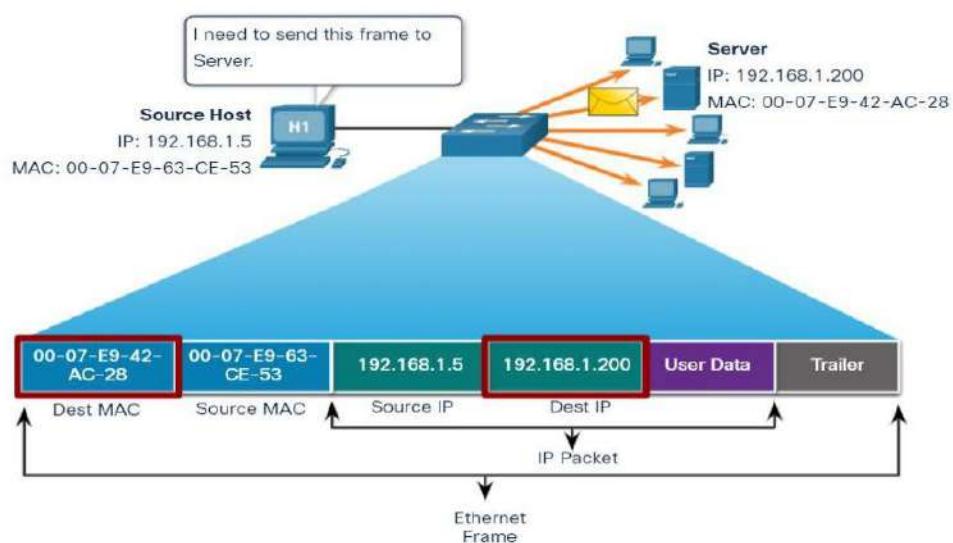
Catatan: Ethernet NIC juga akan menerima Frame jika Destination MAC Address adalah Broadcast atau grup multicast di mana host adalah anggota.

Perangkat apa pun yang menjadi Source atau Destination Frame Ethernet, akan memiliki Ethernet NIC dan karenanya, MAC Address. Ini termasuk workstation, server, printer, perangkat seluler, dan router.

Unicast MAC Address

Di Ethernet, MAC Address yang berbeda digunakan untuk komunikasi unicast, broadcast, dan multicast Layer 2.

Unicast MAC Address adalah alamat unik yang digunakan saat frame dikirim dari satu perangkat transmisi ke satu perangkat tujuan.



Sebagai contoh , host dengan alamat IPv4 192.168.1.5 (source) meminta halaman web dari server di alamat unicast IPv4 192.168.1.200. Agar paket unicast dikirim dan diterima, alamat IP tujuan harus berada di header paket IP. MAC Address tujuan terkait juga harus ada di header frame Ethernet. IP Address dan MAC Address digabungkan untuk mengirimkan data ke satu host tujuan tertentu.

Proses yang digunakan host untuk menentukan alamat MAC tujuan yang terkait dengan alamat IPv4 dikenal sebagai **Address Resolution Protocol (ARP)**. Proses yang digunakan host sumber untuk menentukan **Destination MAC Address** yang terkait dengan alamat **IPv6** dikenal sebagai **Neighbor Discovery (ND)**.

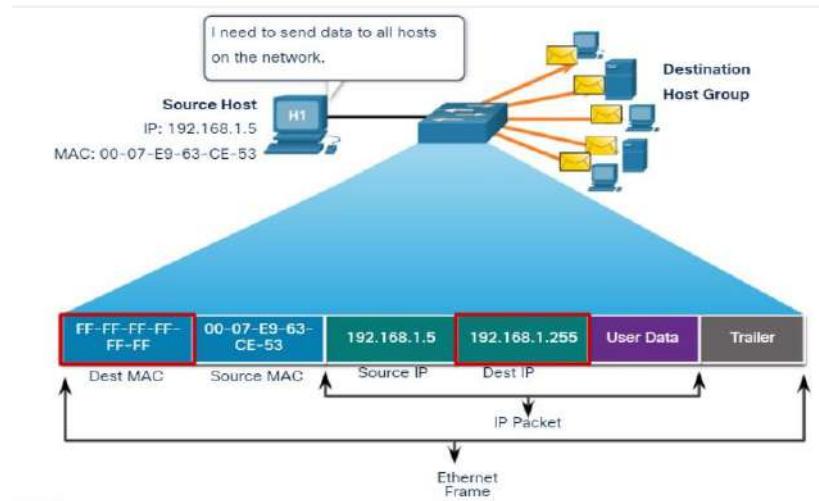
Catatan: Source MAC Address harus selalu unicast.

Broadcast MAC Address

Broadcast Ethernet Frame diterima dan diproses oleh setiap perangkat di LAN Ethernet. Fitur Broadcast Ethernet adalah sebagai berikut:

- Memiliki Destination MAC Address FF-FF-FF-FF-FF-FF dalam heksadesimal (48 yang dalam biner).
- Hal ini membanjiri semua port switch **Ethernet** kecuali port yang masuk.
- tidak diteruskan oleh router.

Jika data yang dienkapsulasi adalah paket broadcast IPv4, ini berarti paket berisi Destination IPv4 Address yang memiliki semua yang (1) di bagian host. Penomoran di alamat ini berarti bahwa semua host di jaringan lokal (Broadcast domain) akan menerima dan memproses paket.



Sebagai contoh, Source host mengirim paket broadcast IPv4 ke semua perangkat di jaringannya. Destination IPv4 address adalah Broadcast Address, 192.168.1.255. Ketika paket Broadcast IPv4 dienkapsulasi dalam Frame Ethernet, Destination MAC Address adalah Broadcast MAC Address FF-FF-FF-FF-FF-FF dalam heksadesimal (48 yang dalam biner).

DHCP untuk IPv4 adalah contoh protokol yang menggunakan Broadcast Ethernet Address dan IPv4.

Namun, tidak semua Broadcast Ethernet membawa paket Broadcast IPv4. Misalnya, ARP Request tidak menggunakan IPv4, tetapi pesan ARP dikirim sebagai Broadcast Ethernet.

Multicast MAC Address

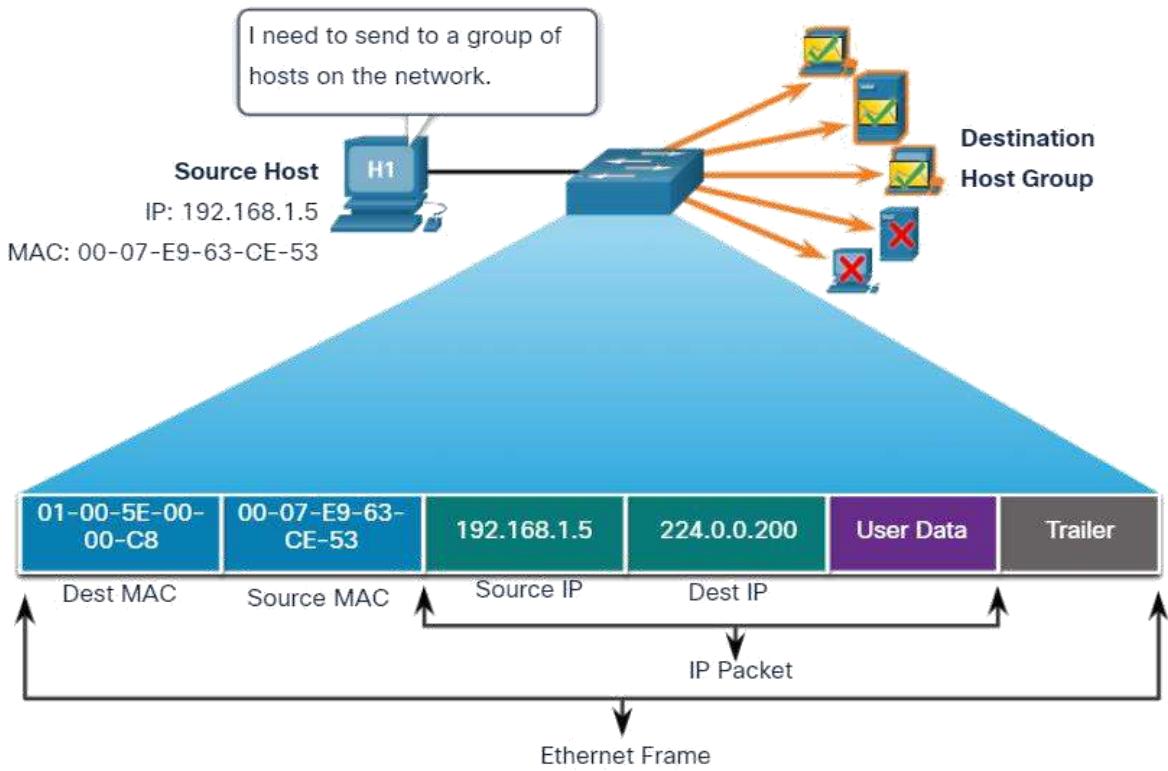
Frame multicast Ethernet diterima dan diproses oleh sekelompok perangkat di LAN Ethernet yang termasuk dalam grup multicast yang sama. Fitur multicast Ethernet adalah sebagai berikut:

- Destination MAC Address 01-00-5E ketika data yang dienkapsulasi adalah paket multicast IPv4 dan Destination MAC Address 33-33 ketika data yang dienkapsulasi adalah paket multicast IPv6.
- Destination MAC Address multicast lain yang dipesan ketika data yang dienkapsulasi bukan IP, seperti Spanning Tree Protocol (STP) dan Link Layer Discovery Protocol (LLDP).
- Hal ini membanjiri semua port switch Ethernet kecuali port masuk, kecuali switch dikonfigurasi untuk multicast snooping.
- Ini tidak diteruskan oleh router, kecuali router dikonfigurasi untuk merutekan paket multicast.

Jika data yang dienkapsulasi adalah paket multicast IP, perangkat yang termasuk dalam grup multicast diberi alamat IP group multicast. Kisaran alamat multicast IPv4 adalah 224.0.0.0 hingga 239.255.255.255. Rentang alamat multicast IPv6 dimulai dengan ff00::/8. Karena alamat multicast mewakili sekelompok alamat (kadang-kadang disebut grup host), mereka hanya dapat digunakan sebagai Packet Destination. Source akan selalu menjadi alamat unicast.

Seperti halnya alamat unicast dan broadcast, alamat IP multicast memerlukan alamat MAC multicast yang sesuai untuk mengirimkan Frame di jaringan lokal. Alamat MAC multicast dikaitkan dengan menggunakan informasi alamat dari alamat multicast IPv4 atau IPv6.

Protokol routing dan protokol jaringan lainnya menggunakan penanganan multicast. Aplikasi seperti perangkat lunak video dan gambar juga dapat menggunakan alamat multicast, meskipun aplikasi multicast tidak umum.



MAC Address Table

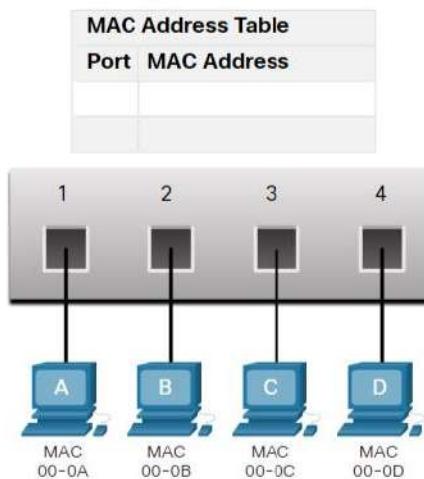
Sekarang setelah Anda mengetahui semua tentang MAC Address Ethernet, sekarang saatnya untuk berbicara tentang bagaimana switch menggunakan alamat ini untuk meneruskan (atau membuang) frame ke perangkat lain di jaringan. Jika switch baru saja meneruskan setiap frame yang diterimanya semua port, jaringan Anda akan sangat padat sehingga mungkin akan berhenti total.

Switch Fundamental

Layer 2 Ethernet Switch menggunakan MAC Address Layer 2 untuk membuat keputusan Forwarding. Ini sama sekali tidak menyadari data (protokol) yang dibawa dalam bagian data Frame, seperti paket IPv4, pesan ARP, atau paket IPv6 ND. Switch membuat keputusan Forwarding hanya berdasarkan pada MAC Address Ethernet Layer 2.

Ethernet Switch memeriksa tabel MAC Addressnya untuk membuat keputusan Forwarding untuk setiap Frame, tidak seperti hub Ethernet Legacy yang mengulangi bit keluar semua port kecuali port yang masuk. Dalam angka tersebut, Switch 4 port baru saja dinyalakan.

Catatan: MAC Address dipersingkat di seluruh materi ini untuk tujuan demonstrasi.



Tabel alamat MAC switch kosong

Catatan: Tabel alamat MAC terkadang disebut sebagai tabel **content addressable memory (CAM)**. Meskipun istilah tabel **CAM** cukup umum, Di Materi ini kami akan menyebutnya sebagai tabel MAC Address.

Switch Learning dan Forwarding

Switch secara dinamis menyusun tabel MAC Address dengan memeriksa Source MAC Address dari Frame Yang diterima pada port. Switching meneruskan Frame dengan mencari kecocokan antara Destination MAC Address dalam Frame dan entri dalam tabel MAC Address.

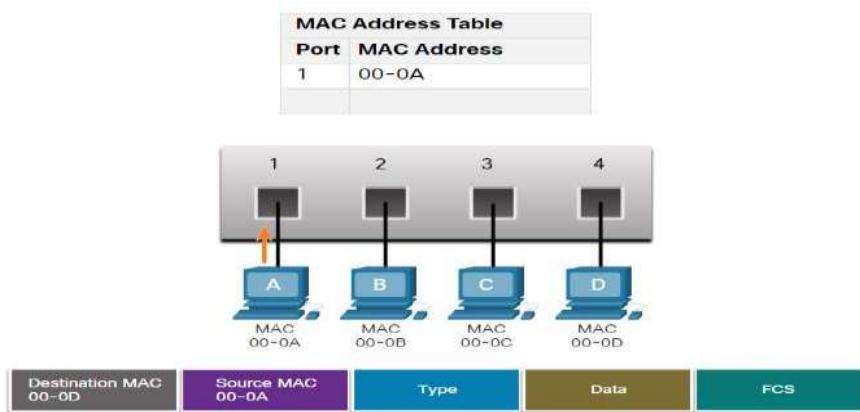
Learn

Memeriksa Source MAC Address

Setiap Frame yang masuk ke Switch diperiksa untuk dipelajari sebagai informasi baru. Ini dilakukan dengan memeriksa Frame dari Source MAC Address dan nomor port di mana Frame memasuki Switch. Jika Source MAC Address tidak ada, maka akan ditambahkan ke tabel dengan nomor port yang masuk. Jika Source MAC Address memang ada, Switch akan memperbarui timer refresh untuk entri tersebut dalam tabel. Secara default, sebagian besar Switch Ethernet menyimpan entri dalam tabel selama 5 menit.

Dalam gambar misalnya, PC-A mengirimkan Frame Ethernet ke PC-D. Tabel memperlihatkan Switch menambahkan MAC Address untuk PC-A ke Tabel MAC Address.

Catatan: Jika Source MAC Address memang ada dalam tabel tetapi pada port yang berbeda, Switch memperlakukan ini sebagai entri baru. Entri diganti menggunakan MAC Address Yang sama tetapi dengan nomor port yang lebih saat ini.



1. PC-A mengirimkan Frame Ethernet.
2. Switch menambahkan nomor port dan MAC Address untuk PC-A ke Tabel MAC Address.

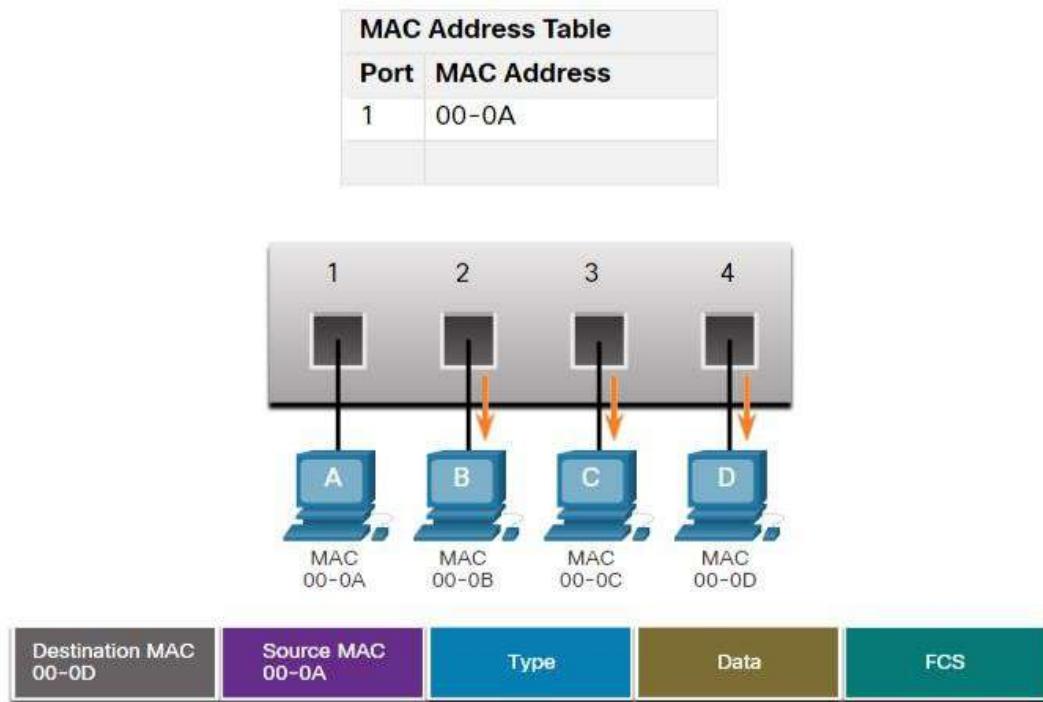
Forwarding

Mencari Destination MAC Address

Jika Destination MAC Address adalah unicast address, Switch akan mencari kecocokan antara Frame Destination MAC Address dan entri dalam tabel alamat MAC-nya. Jika Destination MAC Address berada dalam tabel, itu akan meneruskan Frame keluar port yang ditentukan. Jika Destination MAC Address tidak ada dalam tabel, Switch akan meneruskan Frame keluar semua port kecuali port yang masuk. Ini disebut *Unknown unicast*.

Seperti yang ditunjukkan pada gambar, Switch tidak memiliki Destination MAC Address dalam tabelnya untuk PC-D, sehingga mengirimkan Frame keluar semua port kecuali port 1.

Catatan: Jika Destination MAC Address adalah broadcast atau multicast, Frame juga membanjiri semua port kecuali port yang masuk.



1. Destination MAC Address Tidak ditemukan di dalam table.
2. Switch meneruskan Frame keluar semua port lainnya.

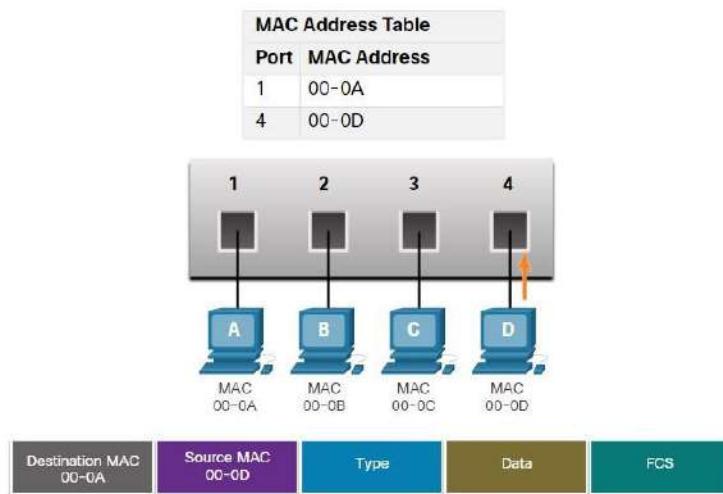
Memfilter Frame

Saat Switch menerima Frame dari perangkat yang berbeda, switch dapat mengisi tabel MAC Address-nya dengan memeriksa Source MAC Address dari setiap Frame. Saat tabel MAC Address switch berisi Destination MAC Address, ia dapat memfilter Filter dan meneruskan satu port.

Lihat Contoh Ilustrasi

PC-D to Switch

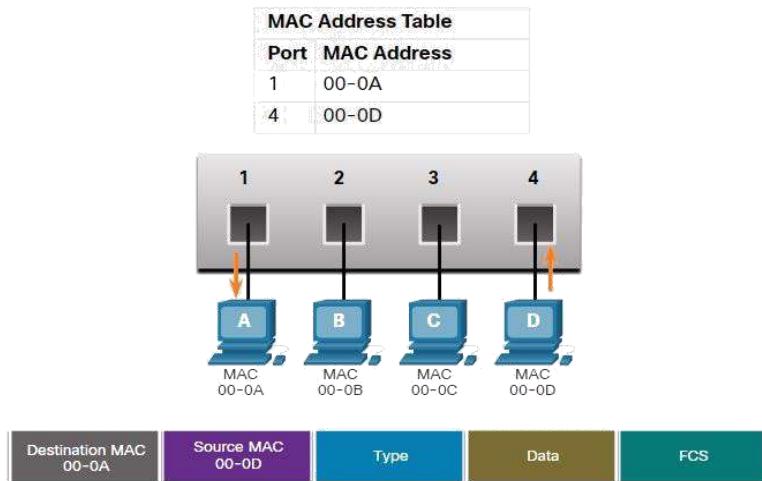
Dalam gambar, PC-D membalsas PC-A. Switch melihat Frame MAC Address PC-D masuk pada port 4. Switch kemudian menempatkan MAC Address PC-D ke dalam Tabel MAC Address yang terkait dengan port 4.



Switch menambahkan nomor port dan MAC Address untuk PC-D ke tabel MAC Address-nya.

Switch to PC-A

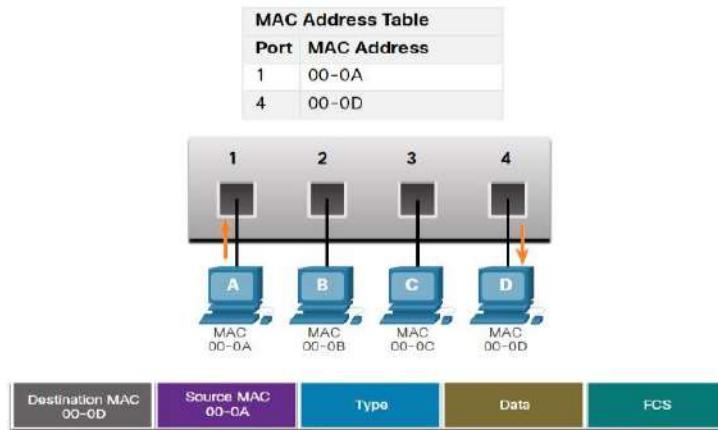
Selanjutnya, karena switch memiliki Destination MAC Address untuk PC-A di Tabel MAC Address, switch hanya akan mengirim Frame keluar port 1, seperti yang ditunjukkan pada gambar.



1. Switch memiliki entri Destination MAC Address.
2. Switch memfilter Frame, mengirimkannya hanya keluar port 1.

PC-A to Switch to PC-D

Selanjutnya, PC-A mengirim Frame lain ke PC-D seperti yang ditunjukkan pada gambar. Tabel MAC Address sudah berisi MAC Address untuk PC-A; oleh karena itu, timer refresh lima menit untuk entri tersebut di reset. Selanjutnya, karena tabel switch berisi Destination MAC Address untuk PC-D, ia hanya mengirimkan Frame keluar port 4.



1. Switch menerima Frame lain dari PC-A dan menyegarkan timer untuk entri MAC Address untuk port 1.
2. Switch memiliki entri terbaru untuk Destination MAC Address dan memfilter Frame , meneruskannya hanya keluar port 4.

Metode Switch Speed dan Forwarding

Seperi yang Anda pelajari di materi sebelumnya, **Switch** menggunakan tabel **MAC Address** mereka untuk menentukan port mana yang akan digunakan untuk **memforward Frame**. Dengan **Switch Cisco**, sebenarnya ada dua metode **Frame Forwarding** dan ada alasan bagus untuk menggunakannya alih-alih yang lain, tergantung pada situasinya.

Metode Frame Forwarding pada Switch Cisco

Switch menggunakan salah satu metode **Forwarding** berikut untuk beralih data antar port jaringan:

- **Store-and-forward switching** - Metode Frame Forwarding ini menerima seluruh Frame dan menghitung CRC. CRC menggunakan rumus matematika, berdasarkan jumlah bit (1s) dalam Frame, untuk menentukan apakah Frame yang diterima memiliki kesalahan. Jika CRC valid, Switch mencari Destination Address, yang menentukan Out Interface. Kemudian Frame diteruskan keluar dari port yang benar.
- **Cut-through switching** – Metode Frame Forwarding ini meneruskan Frame sebelum sepenuhnya diterima. Minimal, Destination Frame Address harus dibaca sebelum Frame dapat diteruskan.

Keuntungan besar dari Store-and-forward switching adalah bahwa ia menentukan apakah Frame memiliki kesalahan sebelum menyebarluaskan Frame. Ketika kesalahan terdeteksi dalam Frame, Switch akan membuang Frame. Membuang Frame dengan kesalahan mengurangi jumlah bandwidth yang dikonsumsi oleh data yang rusak. **Store-and-forward switching** diperlukan untuk analisis **Quality Of Service (QoS)** pada jaringan yang **terkonvergensi** di mana klasifikasi **Frame** untuk prioritas **Traffic** diperlukan. Misalnya, aliran data **voice over IP (VoIP)** perlu memiliki prioritas melalui **Traffic** penjelajahan web.

Cut-Through Switch

Dalam Cut-Through Switch, Switch bertindak berdasarkan data segera setelah diterima, bahkan jika transmisi tidak selesai. Switch Buffer cukup mengirim Frame untuk membaca Destination MAC Address sehingga dapat menentukan port mana yang harus diteruskan datanya. Destination MAC Address terletak di 6 byte pertama Frame setelah *preamble*. Switch mencari Destination MAC Address dalam tabel switching-nya, menentukan port Out Interface , dan meneruskan Frame ke tujuannya melalui port switch yang ditunjuk. Switch tidak melakukan pemeriksaan kesalahan pada Frame.

Ada dua varian switching cut-through:

- **Fast-forward switching** - Fast-forward switching menawarkan tingkat latensi terendah. Fast-forward switching segera meneruskan paket setelah membaca Destination Address. Karena Fast-forward switching mulai diteruskan sebelum seluruh paket diterima, mungkin ada kalanya paket disampaikan dengan kesalahan. Ini jarang terjadi, dan tujuan NIC membuang paket yang rusak setelah diterima. Dalam mode Fast-Forward, latensi diukur dari bit pertama yang diterima ke bit pertama yang ditransmisikan. Fast-forward switching adalah metode switching cut-through yang khas.
- **Fragment-free switching** - Dalam Fragment-free switching, Switch menyimpan 64 byte pertama dari Frame sebelum meneruskan. Fragment-free switching dapat dilihat sebagai kompromi antara Store-and-forward switching dan Fast-Forward Switching. Alasan Fragment-free switching hanya menyimpan 64 byte pertama dari Frame adalah bahwa sebagian besar kesalahan jaringan dan Collision terjadi selama 64 byte pertama. Fragment-free switching mencoba meningkatkan Switching Fast-Forward dengan melakukan pemeriksaan kesalahan kecil pada 64 byte pertama Frame untuk memastikan bahwa Collision belum terjadi sebelum meneruskan Frame. Fragment-free switching adalah kompromi antara latensi tinggi dan integritas tinggi Store-and-forward switching , dan latensi rendah dan berkurangnya integritas Fast-Forward Switching.

Beberapa **Switch** dikonfigurasi untuk melakukan **Cut-Through Switch** berdasarkan per-port sampai ambang kesalahan yang ditentukan pengguna tercapai, dan kemudian secara otomatis berubah menjadi **store-and-forward**. Ketika tingkat kesalahan jatuh di bawah ambang batas, port secara otomatis berubah kembali ke pengalihan **cut-through**.

Memory Buffering on Switches

Switch Ethernet dapat menggunakan teknik **buffering** untuk menyimpan **Frame** sebelum meneruskannya. **Buffering** juga dapat digunakan ketika **Destination Port** sibuk karena kemacetan. **Switch** menyimpan **Frame** sampai dapat ditransmisikan.

Seperti yang ditunjukkan dalam tabel, ada dua metode **Memory buffering** :

| Metode | Deskripsi |
|--------------------------------|--|
| Port-based memory | <ul style="list-style-type: none">• Frame disimpan dalam antrean yang ditautkan ke port incoming dan outgoing tertentu.• Frame ditransmisikan ke port outgoing hanya ketika semua Frame di depan dalam antrian telah berhasil ditransmisikan.• Dimungkinkan untuk satu Frame untuk menunda transmisi semua Frame dalam memori karena port tujuan yang sibuk.• Delay ini terjadi bahkan jika Frame lain dapat ditransmisikan ke port tujuan terbuka. |
| Shared Memory Buffering | <ul style="list-style-type: none">• Menyimpan semua Frame ke Memory Buffer umum yang dibagikan oleh semua port switch dan jumlah Memory Buffer yang diperlukan oleh port dialokasikan secara dinamis.• Frame dalam Buffer secara dinamis ditautkan ke port tujuan yang memungkinkan paket diterima pada satu port dan kemudian ditransmisikan pada port lain, tanpa memindahkannya ke antrian yang berbeda. |

Shared memory buffering juga menghasilkan kemampuan untuk menyimpan **Frame** yang lebih besar dengan **Frame** yang berpotensi rusak. Ini penting dengan peralihan asimetris yang memungkinkan tarif data yang berbeda pada port yang berbeda seperti saat menghubungkan server ke port switch 10 Gbps dan PC ke port 1 Gbps.

Setelan Dupleks dan Speed

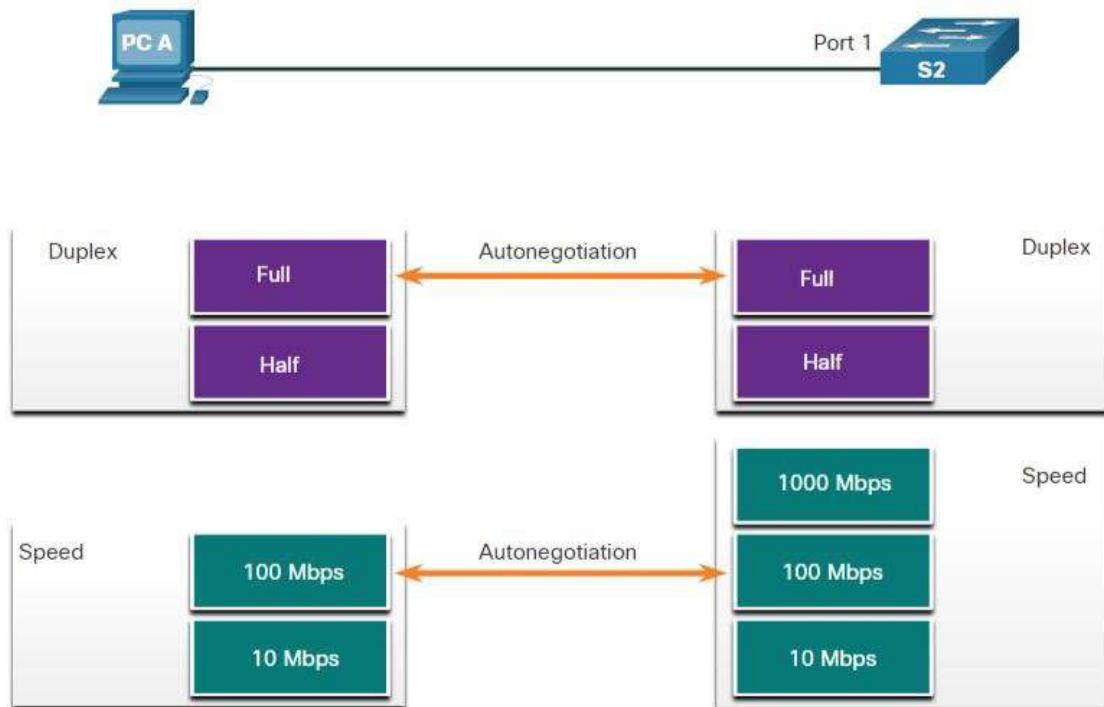
Dua pengaturan paling dasar pada switch adalah bandwidth (kadang-kadang disebut sebagai “speed”) dan pengaturan dupleks untuk setiap port switch individu. Sangat penting bahwa pengaturan dupleks dan bandwidth cocok antara port switch dan perangkat yang terhubung, seperti komputer atau Switch lain.

Ada dua jenis setelan **dupleks** yang digunakan untuk komunikasi pada jaringan **Ethernet**:

- **Full-Dupleks** – Kedua ujung koneksi dapat mengirim dan menerima secara bersamaan.
- **Half-duplex** – Hanya satu ujung koneksi yang dapat dikirim pada satu waktu.

Autonegotiation adalah fungsi opsional yang ditemukan pada sebagian besar **switch Ethernet** dan **NIC**. Ini memungkinkan dua perangkat untuk secara otomatis menegosiasikan kemampuan **kecepatan** dan **dupleks** terbaik. **Full Duplex** dipilih jika kedua perangkat memiliki kemampuan bersama dengan bandwidth umum tertinggi mereka.

Dalam gambar, **Ethernet NIC** untuk **PC-A** dapat beroperasi dalam dupleks penuh atau setengah dupleks, dan dalam 10 Mbps atau 100 Mbps.



PC-A terhubung untuk beralih S2 pada port 1, yang dapat beroperasi dalam **Full duplex** atau **Half dupleks**, dan dalam **10 Mbps**, **100 Mbps** atau **1000 Mbps (1 Gbps)**. Jika kedua perangkat menggunakan **auto negotiation**, mode operasi akan **full-duplex** dan **100 Mbps**.

Catatan: Sebagian besar **Switch Cisco** dan **NIC Ethernet** default ke **autonegotiation** untuk **Speed** dan **dupleks**. Port **Ethernet Gigabit** hanya beroperasi dalam dupleks penuh.

Ketidakcocokan **dupleks** adalah salah satu penyebab paling umum masalah kinerja pada link **Ethernet 10/100 Mbps**. Ini terjadi ketika satu port pada **Link** beroperasi pada **Half dupleks** sementara port lain beroperasi pada **Full dupleks**, seperti yang ditunjukkan pada gambar.



S2 akan terus mengalami collision karena S1 terus mengirim frame kapan saja ia memiliki sesuatu untuk dikirim.

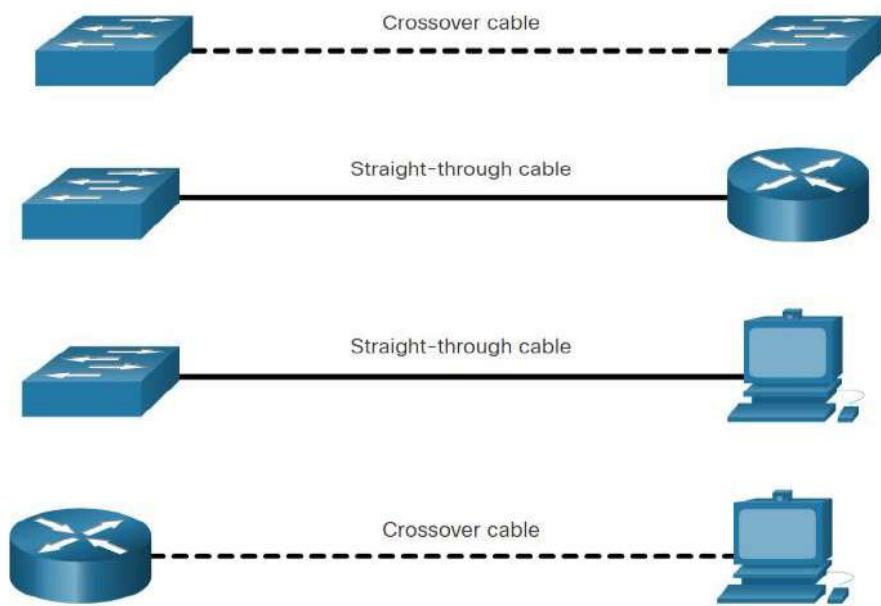
Ketidakcocokan dupleks terjadi ketika satu atau kedua port pada link di reset, dan proses autonegotiation tidak mengakibatkan kedua mitra link memiliki konfigurasi yang sama. Ini juga dapat terjadi ketika pengguna mengkonfigurasi ulang satu sisi link dan lupa untuk mengkonfigurasi ulang sisi lain. Kedua sisi link harus memiliki autonegotiation pada, atau kedua belah pihak harus memiliki off. Praktik terbaik adalah mengkonfigurasi kedua port switch Ethernet sebagai full-duplex.

Auto-MDIX

Koneksi antar perangkat pernah memerlukan penggunaan kabel crossover dan straight-through. Jenis kabel yang diperlukan tergantung pada jenis perangkat yang saling terhubung.

Misalnya, gambar mengidentifikasi jenis kabel yang benar yang diperlukan untuk saling terhubung switch-to-switch, switch-to-router, switch-to-host, atau router-to-host perangkat. Kabel crossover digunakan saat menghubungkan seperti perangkat, dan kabel straight digunakan untuk menghubungkan tidak seperti perangkat.

Catatan: Koneksi langsung antara router dan host memerlukan koneksi cross-over.



Sebagian besar perangkat switch sekarang mendukung fitur automatic medium-dependent interface crossover (auto-MDIX). Ketika diaktifkan, switch secara otomatis mendeteksi jenis kabel yang melekat pada port dan mengkonfigurasi interface yang sesuai. Oleh karena itu, Anda dapat menggunakan crossover atau kabel straight untuk koneksi ke port copper cable 10/100/1000 pada switch, terlepas dari jenis perangkat di ujung koneksi lainnya.

Fitur auto-MDIX diaktifkan secara default pada switch yang menjalankan Cisco IOS Release 12.2(18)SE atau yang lebih baru. Namun, fitur tersebut bisa dinonaktifkan. Untuk alasan ini, Anda harus selalu menggunakan jenis kabel yang benar dan tidak bergantung pada fitur AUTO-MDIX. Auto-MDIX dapat diaktifkan kembali menggunakan perintah mdix

BAB 8

~ *Network Layer* ~

Judul Bab : Network Layer

Tujuan Bab : Menjelaskan bagaimana router menggunakan network layer protocol dan mengaktifkan koneksi end to end

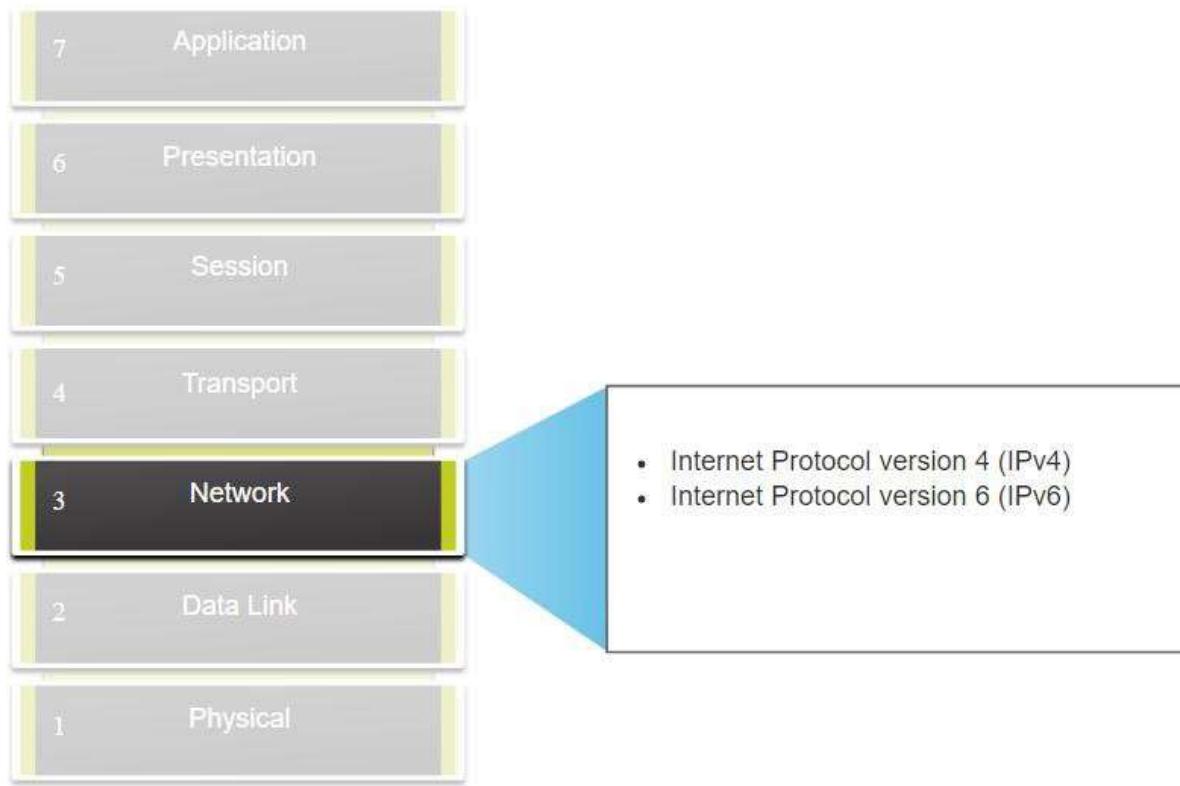
Link Test Pembahasan : <https://forms.gle/hX3E5TQpQz2CfXi6A>

| Judul Materi | Tujuan Materi |
|------------------------------------|--|
| Karakteristik Network Layer | Menjelaskan bagaimana network layer menggunakan IP Protocol untuk membuat reliable network |
| Packet IPv4 | Menjelaskan peran major header field dalam paket IPv4 |
| Packet IPv6 | Menjelaskan peran major header filed dalam paket IPv6 |
| Cara Kerja Router | Menjelaskan bagaimana perangkat jaringan menggunakan routing table untuk mengirim paket ke destination network |
| Router Routing Tables | Menjelaskan fungsi bagian bagian di routing table |

Karakteristik Network Layer

Network Layer, atau OSI Layer 3, menyediakan layanan untuk memungkinkan End Devices bertukar data di seluruh jaringan. Seperti yang ditunjukkan pada gambar, IP versi 4 (IPv4) dan IP versi 6 (IPv6) adalah protokol komunikasi Network Layer. Protokol Network Layer lainnya termasuk protokol routing seperti Open Shortest Path First (OSPF) dan protokol message seperti Internet Control Message Protocol (ICMP).

Network Layer



Protokol Network Layer

- Protokol Internet versi 4 (IPv4)
- Protokol Internet versi 6 (IPv6)

Untuk mencapai komunikasi end-to-end di seluruh batas jaringan, protokol **Network Layer** melakukan empat operasi dasar:

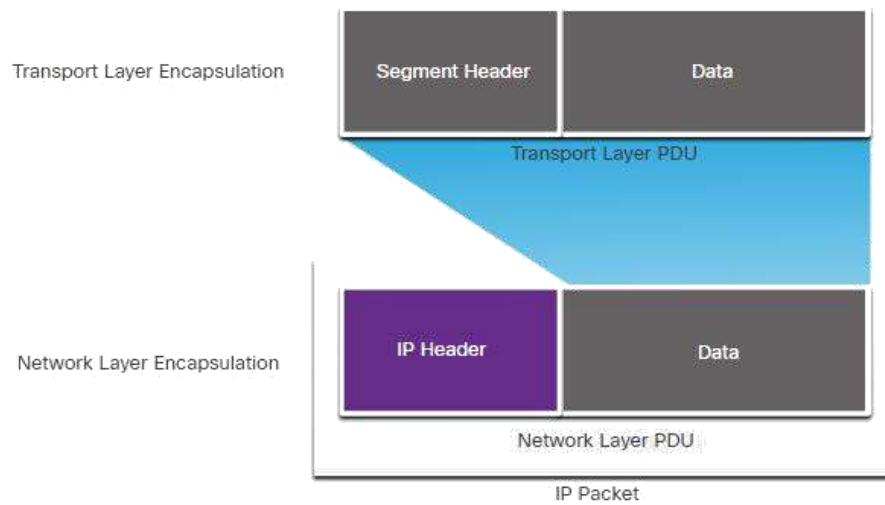
- **Addressing End Device** – End Device harus dikonfigurasi dengan alamat IP unik untuk identifikasi di jaringan.
- **Enkapsulasi** - Network Layer merangkum Protocol Data Unit (PDU) dari Transport Layer menjadi paket. Proses enkapsulasi menambahkan informasi header IP, seperti IP Address host source (pengiriman) dan Destination (menerima). Proses enkapsulasi dilakukan oleh source paket IP.
- **Routing** - Network Layer menyediakan layanan untuk mengarahkan paket ke host tujuan di jaringan lain. Untuk melakukan perjalanan ke jaringan lain, paket harus diproses oleh router. Peran router adalah memilih jalur terbaik dan paket langsung ke host tujuan dalam proses yang dikenal sebagai routing. Paket dapat melintasi banyak router sebelum mencapai host tujuan. Setiap router yang dilewati paket untuk mencapai host tujuan disebut hop.
- **De-enkapsulasi** - Ketika paket tiba di Network Layer host tujuan, host memeriksa header IP packet. Jika Destination IP Address di dalam header cocok dengan IP Address-nya sendiri, header IP dihapus dari paket. Setelah paket dienkapsulasi oleh Network Layer, Layer 4 PDU yang dihasilkan diteruskan ke layanan yang sesuai di transport layer. Proses de-enkapsulasi dilakukan oleh host tujuan paket IP.

Tidak seperti **Transport Layer (OSI Layer 4)**, yang mengelola transportasi data antara proses yang berjalan pada setiap host, protokol komunikasi **Network Layer** (yaitu, **IPv4** dan **IPv6**) menentukan struktur paket dan pemrosesan yang digunakan untuk membawa data dari satu **host** ke **host** lain. Beroperasi tanpa memperhatikan data yang dibawa dalam setiap paket memungkinkan **Network Layer** untuk membawa paket untuk beberapa jenis komunikasi antara beberapa **host**.

Enkapsulasi IP

IP merangkum/enkapsulasi segment Transport Layer (layer tepat di atas Network Layer) atau data lain dengan menambahkan header IP. Header IP digunakan untuk mengirimkan paket ke host tujuan.

Gambar ini menggambarkan bagaimana lapisan transportasi PDU dienkapsulasi oleh lapisan jaringan PDU untuk membuat paket IP.



Proses enkapsulasi data berdasarkan **layer** memungkinkan layanan pada **layer** yang berbeda untuk berkembang dan skala tanpa mempengaruhi **layer** lain. Ini berarti segmen **Transport layer** dapat dengan mudah dikemas oleh **IPv4** atau **IPv6** atau dengan protokol baru yang mungkin dikembangkan di masa depan.

Header **IP** diperiksa oleh perangkat **Layer 3** (yaitu, **router** dan **switch Layer 3/MLS**) saat bepergian melintasi jaringan ke tujuannya. Penting untuk dicatat, bahwa informasi **IP Address** tetap sama sejak paket meninggalkan host sumber sampai tiba di host tujuan, kecuali ketika diterjemahkan oleh perangkat yang melakukan **Network Address Translation (NAT)** untuk **IPv4**.

Router menerapkan **Routing protokol** untuk merutekan paket antar jaringan. **Perutean** yang dilakukan oleh **Intermediary Device** ini memeriksa **Network Layer** yang ditujukan di header paket. Dalam semua kasus, bagian data paket, yaitu **transport layer** yang dienkapsulasi **PDU** atau data lainnya, tetap tidak berubah selama proses **Network Layer**.

Karakteristik IP

IP dirancang sebagai protokol dengan *overhead* rendah. Ini hanya menyediakan fungsi yang diperlukan untuk mengirimkan paket dari sumber ke tujuan melalui sistem jaringan yang saling terhubung. Protokol ini tidak dirancang untuk melacak dan mengelola aliran paket. Fungsi-fungsi ini, jika diperlukan, dilakukan oleh protokol lain di layer lain, terutama TCP di Layer 4.

Ini adalah karakteristik dasar IP:

- **Connectionless** – Tidak ada koneksi dengan tujuan yang dibuat sebelum mengirim paket data.
- **Best Effort** – IP secara *inherent* tidak dapat diandalkan karena pengiriman paket tidak dijamin.
- **Media Independent** – Pengoperasian independen dari media (yaitu, tembaga, serat optik, atau nirkabel) yang membawa data.

ConnectionLess

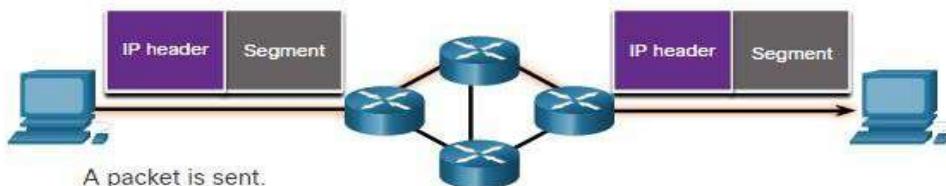
IP Connectionless, yang berarti bahwa tidak ada koneksi end-to-end khusus yang dibuat oleh IP sebelum data dikirim. Komunikasi tanpa koneksi secara konseptual mirip dengan mengirim surat kepada seseorang tanpa memberi tahu penerima terlebih dahulu.

ConnectionLess – Analogi



Komunikasi data ConnectionLess bekerja dengan prinsip yang sama. Seperti yang ditunjukkan pada gambar, IP tidak memerlukan pertukaran informasi kontrol awal untuk membuat koneksi end-to-end sebelum paket diteruskan.

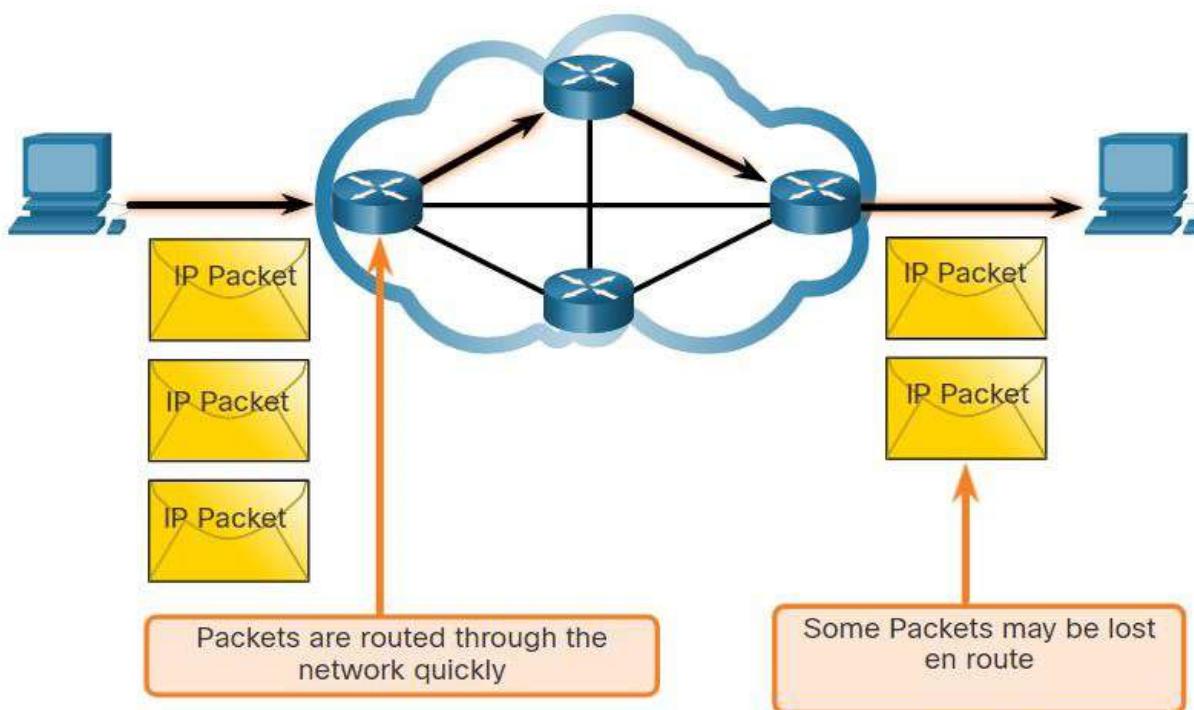
ConnectionLess – Jaringan



Best Effort

IP juga tidak memerlukan **field tambahan** di header untuk mempertahankan koneksi yang dibuat. Proses ini sangat mengurangi **overhead IP**. Namun, tanpa koneksi end-to-end yang telah ditetapkan sebelumnya, pengirim tidak menyadari apakah perangkat tujuan ada dan fungsional saat mengirim paket, juga tidak menyadari apakah tujuan menerima paket, atau jika perangkat tujuan dapat mengakses dan membaca paket.

Protokol IP tidak menjamin bahwa semua paket yang dikirimkan, pada kenyataannya, diterima. Angka tersebut menggambarkan karakteristik pengiriman yang tidak dapat diandalkan atau upaya terbaik dari **protokol IP**.



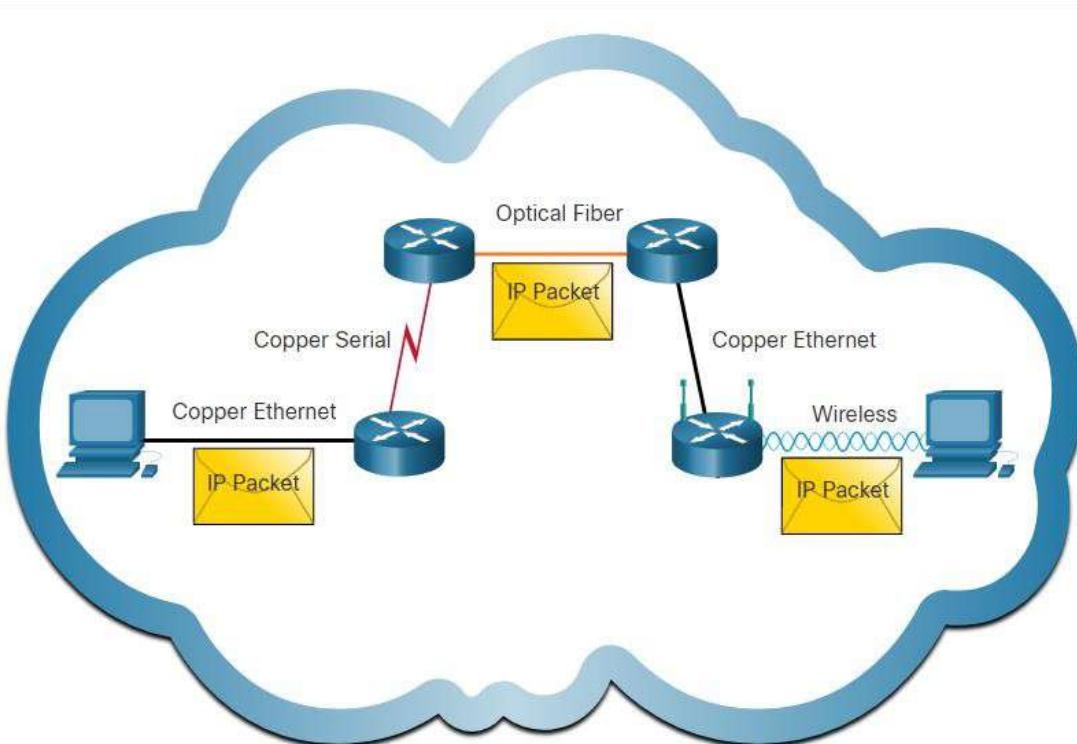
Sebagai protokol **Network Layer** yang tidak dapat diandalkan, **IP** tidak menjamin bahwa semua paket yang dikirim akan diterima. **Protokol** lain mengelola proses pelacakan paket dan memastikan pengiriman mereka.

Media Independen

Tidak dapat diandalkan berarti bahwa **IP** tidak memiliki kemampuan untuk mengelola dan memulihkan dari paket yang tidak terkikis atau korup. Ini karena sementara paket **IP** dikirim dengan informasi tentang lokasi pengiriman, mereka tidak berisi informasi yang dapat diproses untuk memberi tahu pengirim apakah pengiriman berhasil. Paket dapat tiba di tujuan rusak, tidak berurutan, atau tidak sama sekali. **IP** tidak menyediakan kemampuan untuk transaksi ulang paket jika terjadi kesalahan.

Jika paket di luar pesanan dikirim, atau paket hilang, maka aplikasi yang menggunakan data, atau layanan **upper layer**, harus mengatasi masalah ini. Hal ini memungkinkan **IP** berfungsi sangat efisien. Dalam rangkaian protokol **TCP/IP**, *Reliability* adalah peran **protokol TCP** pada **transport layer**.

IP beroperasi secara independen dari media yang membawa data pada **bottom layer** tumpukan protokol. Seperti yang ditunjukkan pada gambar, paket **IP** dapat dikomunikasikan sebagai sinyal elektronik melalui kabel tembaga, sebagai sinyal optik melalui cahaya, atau secara nirkabel sebagai sinyal radio.



Paket IP dapat melakukan perjalanan melalui media yang berbeda.

Data Link Layer OSI bertanggung jawab untuk mengambil paket IP dan mempersiapkannya untuk transmisi melalui media komunikasi. Ini berarti bahwa pengiriman paket IP tidak terbatas pada media tertentu.

Namun, ada satu karakteristik utama media yang dipertimbangkan **Network Layer**: ukuran maksimum **PDU** yang dapat diangkut oleh setiap media. Karakteristik ini disebut sebagai **Maximum Transmission Unit (MTU)**. Bagian dari komunikasi kontrol antara **Data Link Layer** dan **Network Layer** adalah pembentukan ukuran maksimum untuk paket. **Data Link Layer** melewati nilai **MTU** hingga **Network Layer**. **Network Layer** kemudian menentukan seberapa besar paket bisa dikirim.

Dalam beberapa kasus, perangkat menengah, biasanya **router**, harus membagi paket **IPv4** saat meneruskannya dari satu media ke media lain dengan **MTU** yang lebih kecil. Proses ini disebut **fragmenting paket**, atau **fragmentasi**. **Fragmentasi** menyebabkan latensi. Paket **IPv6** tidak dapat difragmentasi oleh router.

Paket IPv4

IPv4 adalah salah satu protokol komunikasi Network Layer Yang utama. Header paket IPv4 digunakan untuk memastikan bahwa paket ini dikirim ke perhentian berikutnya dalam perjalanan ke Destination End Devices.

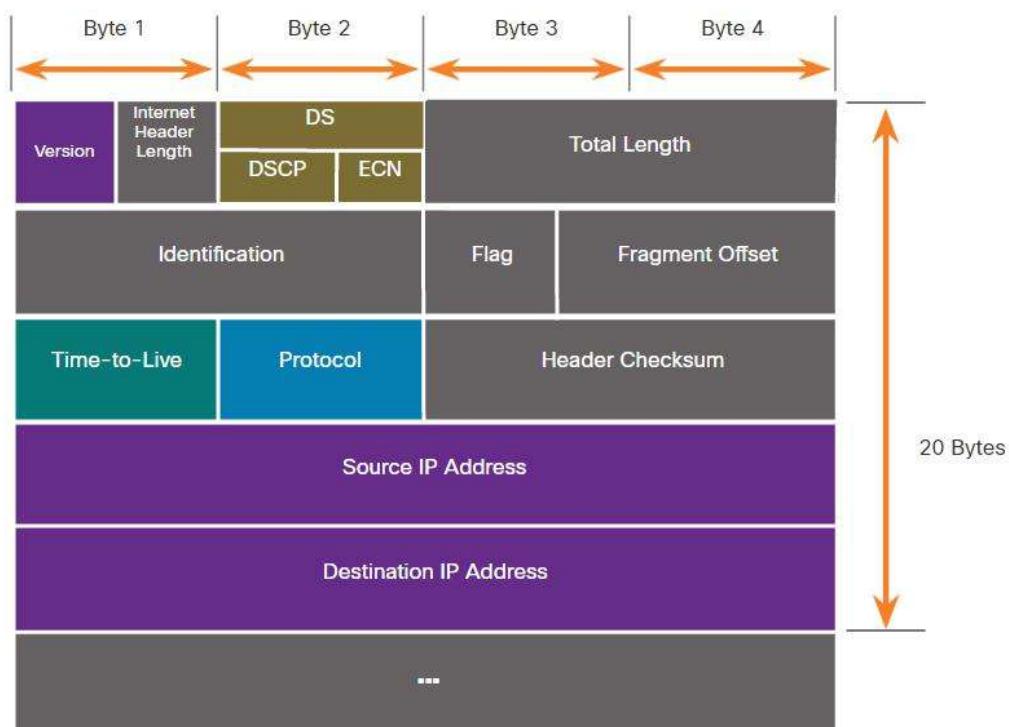
Header Paket IPv4

Header paket IPv4 terdiri dari Field yang berisi informasi penting tentang paket. Field ini berisi bilangan biner yang diperiksa oleh Layer 3 Process

Field Header Packet IPv4

Nilai biner dari setiap Field mengidentifikasi berbagai pengaturan paket IP. Protocol header diagrams, yang dibaca dari kiri ke kanan, dan atas ke bawah, menyediakan visual untuk dirujuk saat membahas Field protokol. Diagram header protokol IP dalam gambar mengidentifikasi Field paket IPv4.

Field di Header Paket IPv4



Field penting di header IPv4 meliputi yang berikut ini:

- **Versi** - Berisi nilai biner 4-bit yang diatur ke 0100 yang mengidentifikasi ini sebagai paket **IPv4**.
- **Differentiated Services or DiffServ (DS)** - Sebelumnya disebut **Type Of Service (ToS)**, **Field DS** adalah Field 8-bit yang digunakan untuk menentukan prioritas setiap paket. Enam bit paling signifikan dari Field DiffServ adalah bit differentiated services code point (DSCP) yang berbeda dan dua bit terakhir adalah bit explicit congestion notification (ECN).
- **Checksum Header** – Ini digunakan untuk mendeteksi korupsi di header IPv4.
- **Time To Live (TTL)** – TTL berisi nilai biner 8-bit yang digunakan untuk membatasi masa pakai paket. Perangkat Source IPv4 Address. menetapkan nilai TTL awal. Ini berkurang satu kali paket diproses oleh router. Jika Field TTL menurunkan ke nol, Router akan membuang paket dan mengirim pesan Melalui Internet Control Message Protocol (ICMP) ke Source IPv4 Address. Karena router mendekrementasi TTL dari setiap paket, router juga harus menghitung ulang Checksum Header.
- **Protocol – Field** ini digunakan untuk mengidentifikasi protokol tingkat berikutnya. Nilai biner 8-bit ini menunjukkan jenis muatan data yang dibawa paket, yang memungkinkan Network Layer untuk meneruskan data ke protokol Upper Layer yang sesuai. Nilai umum termasuk ICMP (1), TCP (6), dan UDP (17).
- **Source IPv4 Address** – Ini berisi nilai biner 32-bit yang mewakili paket Source IPv4 Address . Source IPv4 Address selalu merupakan alamat unicast.
- **Destination IPv4 Address** – Ini berisi nilai biner 32-bit yang mewakili paket Destination IPv4 Address. Destination IPv4 Address adalah alamat unicast, multicast, atau broadcast.

Dua **Field** yang paling sering direferensikan adalah **Source** dan **destination IPv4 Address**. **Fields** ini mengidentifikasi dari mana paket itu berasal dan ke mana ia akan pergi. Biasanya, alamat ini tidak berubah saat bepergian dari sumber ke tujuan.

The Internet Header Length (IHL), Total Length, and Header Checksum fields digunakan untuk mengidentifikasi dan memvalidasi paket.

Field yang lainnya digunakan untuk menyusun ulang paket yang terfragmentasi. Secara khusus, paket **IPv4** menggunakan **Field Identification, Flags, and Fragment** untuk melacak fragmen. **Router** mungkin harus memecah paket **IPv4** saat meneruskannya dari satu media ke media lainnya dengan **MTU** yang lebih kecil.

Paket IPv6

IPv4 masih digunakan hari ini. Materi ini tentang **IPv6**, yang pada akhirnya akan menggantikan **IPv4**. Untuk lebih memahami mengapa Anda perlu mengetahui protokol **IPv6**, itu membantu mengetahui keterbatasan **IPv4** dan keuntungan dari **IPv6**.

Batasan IPv4

Selama bertahun-tahun, protokol dan proses tambahan telah dikembangkan untuk mengatasi tantangan baru. Namun, bahkan dengan perubahan, **IPv4** masih memiliki tiga masalah utama:

- **Penipisan alamat IPv4 -** IPv4 memiliki sejumlah alamat publik unik yang tersedia. Meskipun ada sekitar 4 miliar alamat **IPv4**, meningkatnya jumlah perangkat baru yang mendukung **IP**, koneksi yang selalu aktif, dan potensi pertumbuhan wilayah yang kurang berkembang telah meningkatkan kebutuhan akan lebih banyak alamat.
- **Kurangnya konektivitas end-to-end** - Network Address Translation (NAT) adalah teknologi yang umumnya diimplementasikan dalam jaringan IPv4. NAT menyediakan cara bagi beberapa perangkat untuk berbagi satu IPv4 Address publik. Namun, karena IPv4 Address Public terbagikan, IPv4 Address dari host jaringan internal disembunyikan. Ini bisa bermasalah untuk teknologi yang membutuhkan konektivitas end-to-end.
- **Peningkatan kompleksitas jaringan** – Meskipun NAT telah memperpanjang umur IPv4 itu hanya dimaksudkan sebagai mekanisme transisi ke IPv6. NAT dalam berbagai implementasinya menciptakan kompleksitas tambahan dalam jaringan, menciptakan latensi dan membuat pemecahan masalah lebih sulit.

IPv6 Overview

Pada awal 1990-an, **Internet Engineering Task Force (IETF)** semakin khawatir tentang masalah dengan **IPv4** dan mulai mencari penggantinya. Kegiatan ini mengarah pada pengembangan **IP versi 6 (IPv6)**. **IPv6** mengatasi keterbatasan **IPv4** dan merupakan peningkatan yang kuat dengan fitur yang lebih sesuai dengan tuntutan jaringan saat ini dan dapat diperkirakan.

Penyempurnaan yang disediakan **IPv6** meliputi yang berikut ini:

- **Increased address space** – IPv6 Address didasarkan pada pengapian hierarki 128-bit dibandingkan dengan IPv4 dengan 32 bit.
- **Improved packet handling** – Header IPv6 telah disederhanakan dengan Field yang lebih sedikit.
- **Eliminates the need for NAT** – Dengan sejumlah besar alamat IPv6 publik, NAT antara alamat IPv4 Private dan IPv4 public tidak diperlukan. Ini menghindari beberapa masalah yang diinduksi NAT yang dialami oleh aplikasi yang membutuhkan koneksi end-to-end.

Ruang **IPv4 Address** adalah **32-bit** menyediakan sekitar 4.294.967.296 alamat unik. Ruang **IPv6 Address** menyediakan 340.282.366.920.938.463.463.374.607.431.768.211.456, atau 340 alamat tidak terisi. Ini kira-kira setara dengan setiap butir pasir di Bumi.

Perbandingan Ruang Alamat IPv4 dan IPv6

| Nama Nomor | Notasi Ilmiah | Jumlah Nol |
|-------------------------------|-----------------------------|--|
| 1 Ribu | 10^3 | 1,000 |
| 1 Juta | 10^6 | 1,000,000 |
| 1 Miliar (IPv4) | 10^9 | 1,000,000,000 |
| 1 Triliun | 10^{12} | 1,000,000,000,000 |
| 1 Kuadrilir | 10^{15} | 1,000,000,000,000,000 |
| 1 Quintillion | 10^{18} | 1,000,000,000,000,000,000 |
| 1 Sextillion | 10^{21} | 1,000,000,000,000,000,000,000 |
| 1 Septilion | 10^{24} | 1,000,000,000,000,000,000,000,000 |
| 1 Oktilion | 10^{27} | 1,000,000,000,000,000,000,000,000,000 |
| 1 Nonnilion | 10^{30} | 1,000,000,000,000,000,000,000,000,000,000 |
| 1 Desilion | 10^{33} | 1,000,000,000,000,000,000,000,000,000,000,000 |
| 1 Undecillion (IPv6) | 10^{36} | 1,000,000,000,000,000,000,000,000,000,000,000,000,000,000,000,000 |

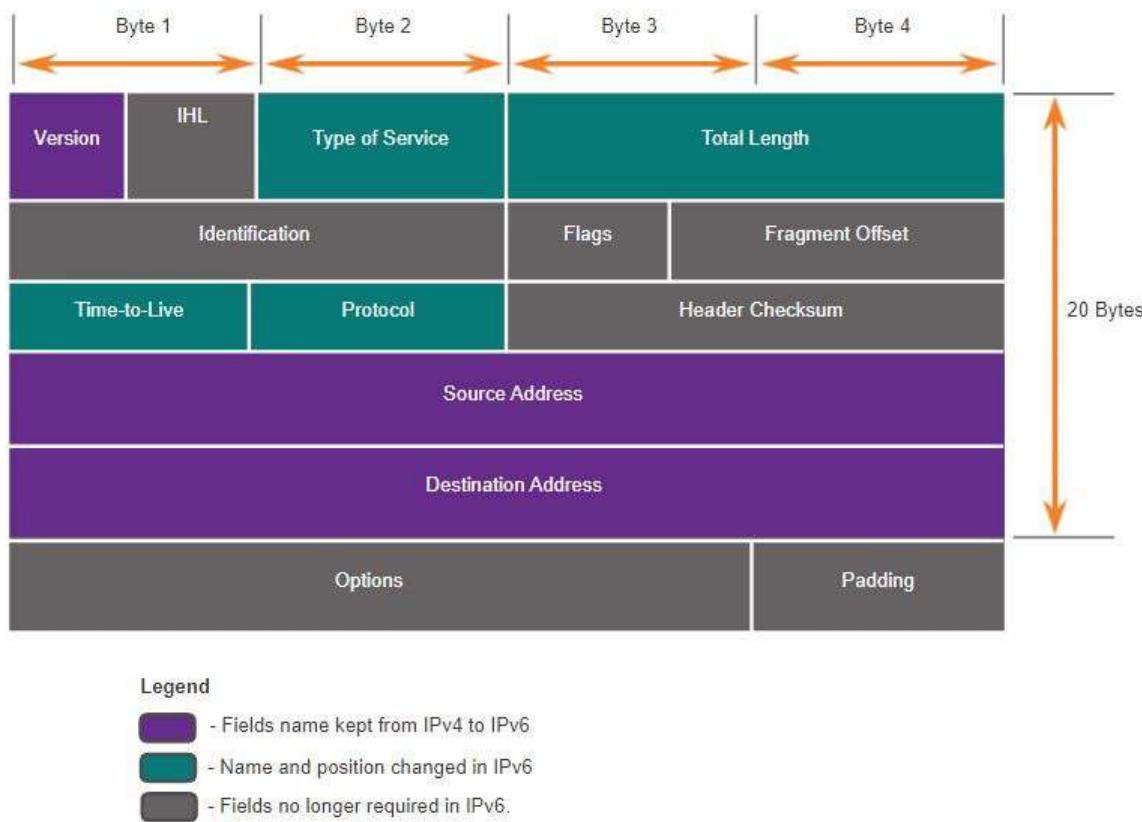
Field Header Paket IPv4 di Header Paket IPv6

Salah satu peningkatan desain utama IPv6 melalui IPv4 adalah header IPv6 yang disederhanakan.

Misalnya, header IPv4 terdiri dari header panjang variabel 20 oktet (hingga 60 byte jika bidang Opsi digunakan) dan 12 field basic header, tidak termasuk field Opsi dan field Padding.

Untuk IPv6, beberapa field tetap sama, beberapa field telah mengubah nama dan posisi, dan beberapa field IPv4 tidak lagi diperlukan, seperti yang disorot dalam gambar.

Header Paket IPv4



Sebaliknya, header IPv6 yang disederhanakan menunjukkan gambar berikutnya terdiri dari header length tetap 40 oktet (sebagian besar karena Length Source and Destination IPv6 Address).

Header IPv6 yang disederhanakan memungkinkan pemrosesan header IPv6 yang lebih efisien.

Header Paket IPv6



IPv6 Packet Header

Diagram **header protokol IP** dalam gambar mengidentifikasi **Field paket IPv6**.

Field di Header Paket IPv6



Field di header paket IPv6 meliputi yang berikut ini:

- **Versi** - Field ini berisi nilai biner 4-bit yang diatur ke 0110 yang mengidentifikasi ini sebagai paket IP versi 6.
- **Traffic Class** - Field 8-bit ini setara dengan **Field IPv4 Differentiated Services (DS)**.
- **Flow Label** - Field 20-bit ini menunjukkan bahwa semua paket dengan Flow Label yang sama menerima jenis penanganan yang sama oleh router.
- **Payload Length** - Field 16-bit ini menunjukkan Length porsi data atau Payload paket IPv6. Ini tidak termasuk header Length IPv6, yang merupakan header 40-byte tetap.
- **Next Header** - Field 8-bit ini setara dengan Field Protokol IPv4. Ini menunjukkan jenis Payload data yang dibawa paket, memungkinkan Network Layer untuk meneruskan data ke protokol Upper Layer yang sesuai.
- **Hop Limit – Field 8-bit** ini menggantikan **Field TTL IPv4**. Nilai ini dibatalkan oleh nilai 1 oleh setiap router yang meneruskan paket. Ketika penghitung mencapai 0, paket dibuang, dan pesan **ICMPv6 Time Exceeded** diteruskan ke host pengirim,. Ini menunjukkan bahwa paket tidak mencapai tujuannya karena batas **hop** terlampaui. Tidak seperti **IPv4**, **IPv6** tidak menyertakan **Checksum Header IPv6**, karena fungsi ini dilakukan pada **Bottom dan Upper Layer**. Ini berarti **checksum** tidak perlu dihitung ulang oleh setiap router ketika mendekrementasi **Field Hop Limit**, yang juga meningkatkan kinerja jaringan.
- **Source IPv6 Address** - Field 128-bit ini mengidentifikasi **IPv6 Address** dari host pengirim.
- **Destination IPv6 Address** - Bidang 128-bit ini mengidentifikasi **IPv6 Address** dari host penerima.

Paket IPv6 mungkin juga berisi **Extension Header (EH)**, yang menyediakan informasi Network Layer opsional. Extension Header bersifat *opsional* dan ditempatkan di antara header IPv6 dan **Payload**. EHs digunakan untuk fragmentasi, keamanan, untuk mendukung mobilitas dan banyak lagi.

Tidak seperti **IPv4**, router tidak memecah paket **IPv6** yang **di routing**.

Cara Kerja Router

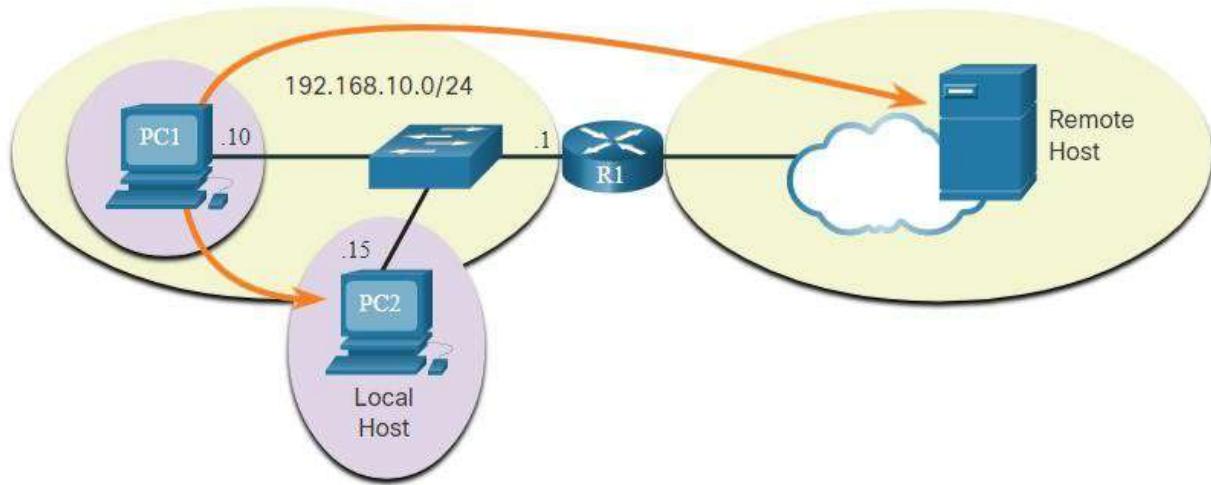
Dengan IPv4 dan IPv6, paket selalu dibuat di Source Host. Source Host harus dapat mengarahkan paket ke Destination Host. Untuk melakukan ini, End Device membuat Routing Table mereka sendiri. Materi ini membahas bagaimana End Device menggunakan Routing Table.

Host Forwarding Decision

Peran lain dari **Network Layer** adalah untuk mengarahkan paket antara **host**. Host dapat mengirim paket ke yang berikut:

- **Itself** – Host dapat melakukan ping sendiri dengan mengirim paket ke alamat IPv4 khusus 127.0.0.1 atau alamat IPv6 ::1, yang disebut sebagai Interface loopback. Ping Interface loopback menguji protokol TCP / IP pada host.
- **Local Host** – Ini adalah Destination host yang berada di jaringan lokal yang sama dengan host pengirim. Destination dan Source Address di jaringan yang sama.
- **Remote Host** – Ini adalah Destination host pada jaringan jarak jauh. Source dan Destination Host tidak berbagi alamat jaringan yang sama.

Gambar ini menggambarkan PC1 terhubung ke **local host** pada jaringan yang sama, dan ke **remote host** yang terletak di jaringan lain.



Apakah paket ditakdirkan untuk local host atau remote host ditentukan oleh perangkat End Devices. Source End Device menentukan apakah Destination IP Address berada pada jaringan yang sama dengan Source Device itu sendiri. Metode penentuan bervariasi menurut versi IP:

- **Di IPv4** – Source Device menggunakan subnet sendiri bersama dengan IPv4 address sendiri dan Destination IPv4 Address untuk membuat penentuan ini.
- **Di IPv6** – Router lokal mengadvertise alamat jaringan lokal (Prefix) ke semua perangkat di jaringan.

Dalam jaringan rumah atau bisnis, Anda mungkin memiliki beberapa perangkat kabel dan nirkabel yang saling terhubung menggunakan Intermediary Device, seperti Switch LAN atau Wireless Access Point (WAP). Intermediary Devices ini menyediakan interkoneksi antara local host di jaringan lokal. local host dapat saling menjangkau dan berbagi informasi tanpa perlu perangkat tambahan apa pun. Jika host mengirim paket ke perangkat yang dikonfigurasi dengan jaringan IP yang sama dengan perangkat host, paket hanya diteruskan keluar dari host interface , melalui intermediary device , dan ke destination device secara langsung.

Tentu saja, dalam kebanyakan situasi kami ingin perangkat kami dapat terhubung di luar segmen jaringan lokal, seperti ke rumah lain, bisnis, dan internet. Perangkat yang berada di luar segmen jaringan lokal dikenal sebagai **remote host**. Ketika perangkat sumber mengirim paket ke perangkat tujuan jarak jauh, maka bantuan **router** dan **routing** diperlukan. **routing** adalah proses mengidentifikasi jalur terbaik ke tujuan. **routing** yang tersambung ke segmen jaringan lokal disebut sebagai Default Gateway.

Default Gateway

Default Gateway adalah perangkat jaringan (yaitu, **router** atau **Switch Layer 3**) yang dapat me routing traffic ke jaringan lain. Jika Anda menggunakan analogi bahwa jaringan seperti ruangan, maka Default Gateway seperti pintu. Jika Anda ingin sampai ke ruangan atau jaringan lain, Anda perlu menemukan pintu.

Di jaringan, **Default Gateway** biasanya merupakan **router** dengan fitur-fitur ini:

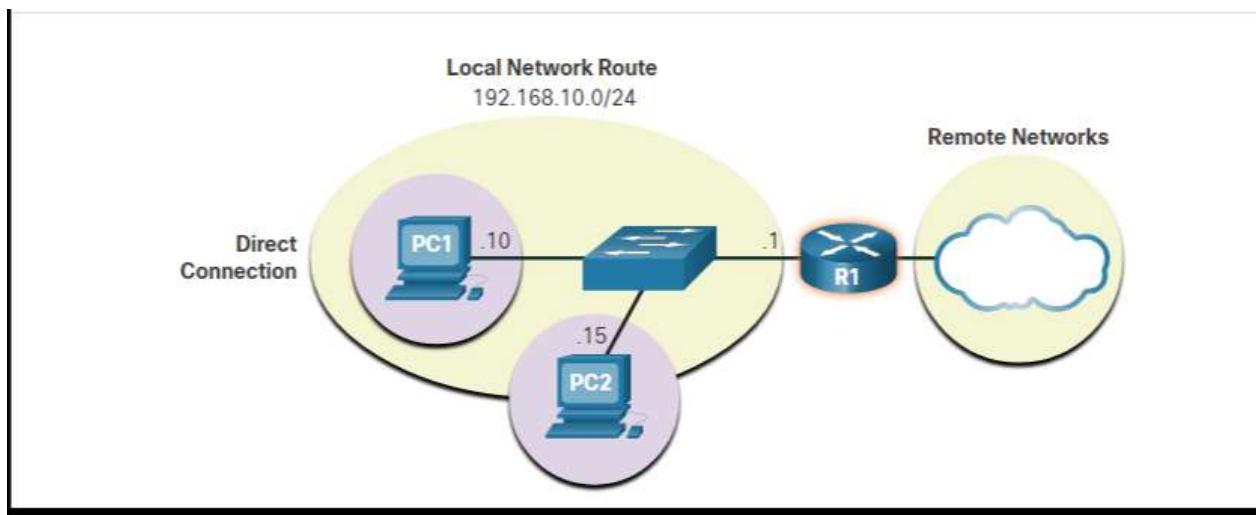
- Ini memiliki alamat **IP lokal** dalam rentang alamat yang sama dengan **host** lain di jaringan lokal.
- Ini dapat menerima data ke jaringan **lokal** dan meneruskan data keluar dari jaringan lokal.
- Ini dapat **me routing traffic** ke jaringan lain.

Default Gateway diperlukan untuk mengirim **traffic** di luar jaringan lokal. **traffic** tidak dapat diteruskan di luar jaringan lokal jika tidak ada **Default Gateway**, alamat **Default Gateway** tidak dikonfigurasi, atau **Default Gateway** mati.

Sebuah Perangkat Menuju Default Gateway

Host Routing Table biasanya akan menyertakan Default Gateway. Di IPv4, host menerima IPv4 Address Default Gateway baik secara dinamis dari Dynamic Host Configuration Protocol (DHCP) atau dikonfigurasi secara static/manual. Di IPv6, router meng advertise Default Gateway Address atau host dapat dikonfigurasi secara manual/static.

Dalam gambar, PC1 dan PC2 dikonfigurasi dengan alamat **IPv4** 192.168.10.1 sebagai **Default Gateway**.



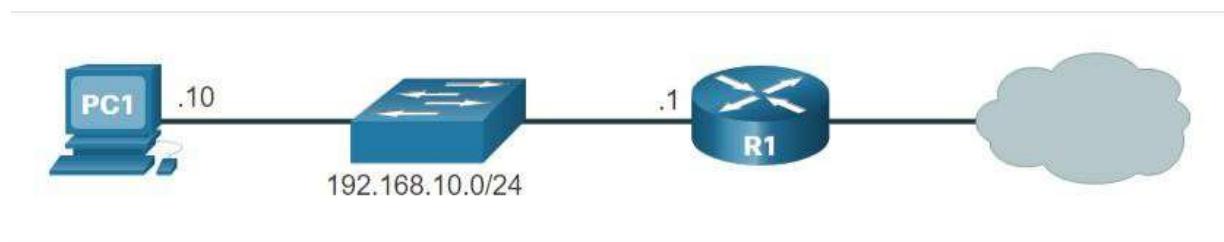
Memiliki Default Gateway yang dikonfigurasi membuat default route dalam routing table PC. Default Gateway adalah route atau jalur yang akan diambil komputer Anda ketika mencoba menghubungi jaringan remote.

PC1 dan PC2 akan memiliki default route untuk mengirim semua lalu lintas yang ditakdirkan ke jaringan remote ke R1.

Routing Table Host

Pada host Windows, perintah **route print** atau **netstat -r** dapat digunakan untuk menampilkan **routing table host**. Kedua perintah menghasilkan output yang sama. Output mungkin tampak luar biasa pada awalnya, tetapi cukup mudah dimengerti.

Gambar menampilkan contoh topologi dan output yang dihasilkan oleh perintah **netstat -r**.



Routing Table IPv4 untuk PC1

```
C:\Users\SkyderAmzLee>netstat -r
=====
Interface List
13...a8 5e 45 37 15 c4 .....Realtek PCIe GbE Family Controller
17...0a 00 27 00 00 11 .....VirtualBox Host-Only Ethernet Adapter
14...70 66 55 26 af 3d .....Realtek 8822CE Wireless LAN 802.11ac PCI-E NIC
4...72 66 55 26 af 3d .....Microsoft Wi-Fi Direct Virtual Adapter
12...f2 66 55 26 af 3d .....Microsoft Wi-Fi Direct Virtual Adapter #2
10...70 66 55 26 af 3c .....Bluetooth Device (Personal Area Network)
1.....Software Loopback Interface 1
=====

IPv4 Route Table
=====
Active Routes:
Network Destination      Netmask        Gateway        Interface    Metric
          0.0.0.0      0.0.0.0   10.255.254.1  10.255.254.207    25
     10.255.254.0  255.255.255.0  On-link       10.255.254.207    281
  10.255.254.207  255.255.255.255  On-link       10.255.254.207    281
  10.255.254.255  255.255.255.255  On-link       10.255.254.207    281
        127.0.0.0    255.0.0.0   On-link         127.0.0.1    331
        127.0.0.1    255.255.255.255  On-link         127.0.0.1    331
  127.255.255.255  255.255.255.255  On-link         127.0.0.1    331
     192.168.1.0  255.255.255.0  On-link       192.168.1.2    281
  192.168.1.2  255.255.255.255  On-link       192.168.1.2    281
 192.168.1.255  255.255.255.255  On-link       192.168.1.2    281
```

Catatan: Output hanya menampilkan **routing table** IPv4.

Memasukkan perintah **netstat -r** atau perintah **Route print** yang setara menampilkan tiga bagian yang terkait dengan koneksi jaringan TCP/IP saat ini:

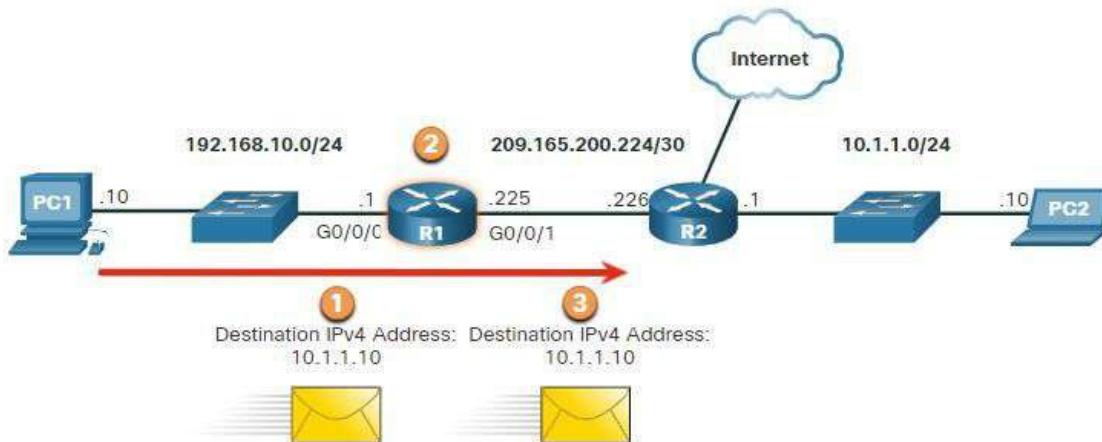
- **Daftar Interface** - Mencantumkan alamat Media Access Control (MAC) dan nomor Interface yang ditetapkan dari setiap Interface berkemampuan jaringan pada host, termasuk adaptor Ethernet, Wi-Fi, dan Bluetooth.
- **Routing Table IPv4** - Mencantumkan semua Route IPv4 yang diketahui, termasuk koneksi langsung, jaringan lokal, dan Default Route Local.
- **Routing Table IPv6** - Mencantumkan semua Route IPv6 yang diketahui, termasuk koneksi langsung, jaringan lokal, dan Default Route Local.

Sebagian besar jaringan juga berisi router, yang merupakan Intermediary Device. Router juga berisi Routing Table. Materi ini mencakup operasi router pada Network Layer. Ketika Host mengirim paket ke host lain, host berkonsultasi dengan Routing Table untuk menentukan kemana harus mengirim paket. Jika host tujuan berada di jaringan remote, paket diteruskan ke default Gateway, yang biasanya merupakan local router.

Keputusan Packet Forwarding Router

Apa yang terjadi ketika paket tiba di **Interface router**?

Router memeriksa paket Destination IP Address dan mencari routing table untuk menentukan ke mana harus meneruskan paket. Routing Table berisi daftar semua alamat jaringan yang diketahui (prefixes) dan di mana meneruskan paket. Entri ini dikenal sebagai routes atau route entries. Router akan meneruskan paket menggunakan entri rute pencocokan terbaik (longest).



1. Paket tiba di gigabit Ethernet 0/0/0 Interface router R1. R1 menghapus header dan trailer Layer 2 Ethernet.
2. Router R1 memeriksa paket destination IPv4 Address dan mencari kecocokan terbaik dalam routing table IPv4-nya. Entri rute menunjukkan bahwa paket ini akan diteruskan ke router R2.
3. Router R1 merangkum paket ke header dan trailer Ethernet baru, dan meneruskan paket ke router hop R2 berikutnya.

Tabel berikut ini memperlihatkan informasi terkait dari routing table R1.

Tabel Routing R1

| Rute | Hop Interfaces |
|-------------------------|----------------|
| 192.168.10.0 /24 | G0/0/0 |
| 209.165.200.224/30 | G0/0/1 |
| 10.1.1.0/24 | melalui R2 |
| Default route 0.0.0.0/0 | melalui R2 |

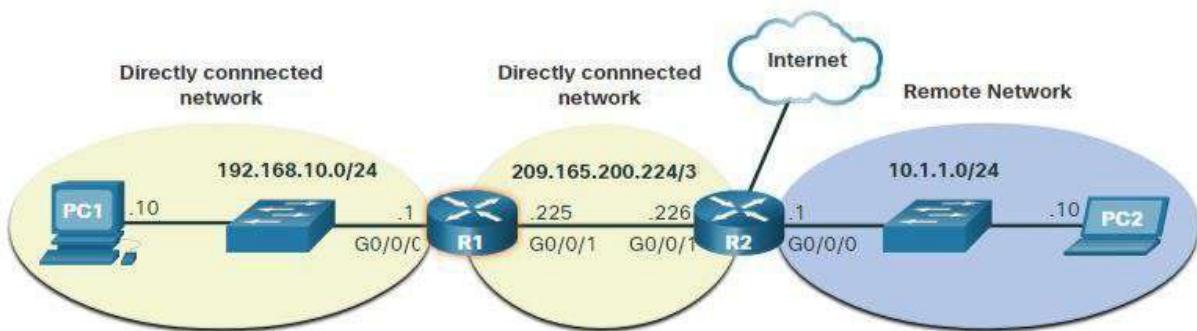
IP Router Routing Table

Routing Table berisi entri rute jaringan yang mencantumkan semua tujuan jaringan yang mungkin diketahui.

Routing Table menyimpan tiga tipe entri rute:

- **Directly-connected networks** - Entri rute jaringan ini adalah Active Router Interfaces. Router menambahkan rute yang Directly Connected ketika Interface dikonfigurasi dengan alamat IP dan diaktifkan. Setiap Interface router terhubung ke segmen jaringan yang berbeda. Dalam gambar, jaringan yang terhubung langsung dalam Routing Table R1 IPv4 adalah 192.168.10.0/24 dan 209.165.200.224/30.
- **Remote networks** - Entri rute jaringan ini tersambung ke Router lain. Router mempelajari tentang jaringan remote baik dengan dikonfigurasi secara eksplisit oleh administrator atau dengan bertukar Routing Information menggunakan protokol Dynamic Routing. Dalam gambar, jaringan jarak jauh dalam tabel Router R1 IPv4 akan menjadi 10.1.1.0/24.
- **Default route** – Seperti host, sebagian besar router juga menyertakan entry route default gateway sebagai pilihan terakhir. Default Route digunakan ketika tidak ada kecocokan yang lebih baik (lebih lama) dalam IP Routing Table. Dalam gambar, Routing Table R1 IPv4 kemungkinan besar akan menyertakan Default Route untuk meneruskan semua paket ke router R2.

Gambar mengidentifikasi jaringan router R1 yang terhubung langsung dan terpencil.



R1 memiliki dua jaringan yang **Directly-Connected**:

- 192.168.10.0/24
- 209.165.200.224/30

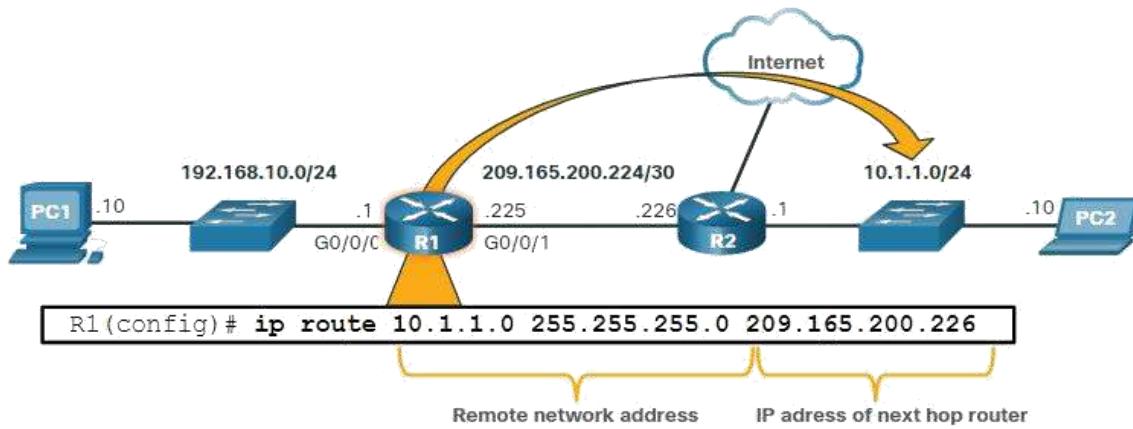
R1 juga memiliki jaringan **remote** (yaitu 10.1.1.0/24 dan internet) yang dapat dipelajarinya.

Router dapat mempelajari tentang jaringan remote dengan salah satu dari dua cara:

- **Static** – Jaringan Remote secara manual dimasukkan ke dalam routing table menggunakan static routing.
- **Dynamic** - Remote routing secara otomatis dipelajari menggunakan protokol Dynamic Routing.

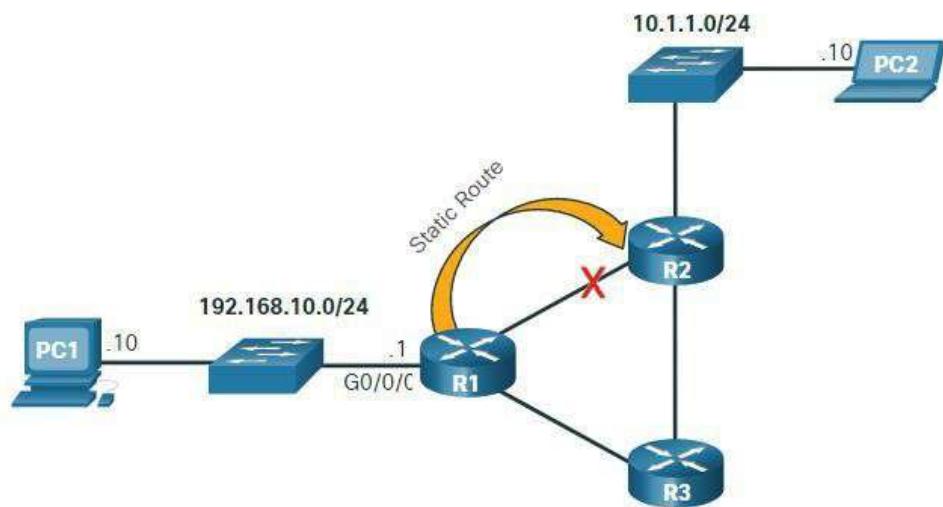
Static Routing

Static Routing adalah entri rute yang dikonfigurasi secara manual. Gambar menunjukkan contoh **Static Routing** yang dikonfigurasi secara manual pada router R1. **Static Routing** mencakup alamat jaringan remote dan alamat IP router hop berikutnya.



R1 dikonfigurasi secara manual dengan **Static Routing** untuk mencapai jaringan 10.1.1.0/24. Jika jalur ini berubah, R1 akan memerlukan **Static Routing** baru.

Jika ada perubahan dalam topologi jaringan, **Static Routing** tidak diperbarui secara otomatis dan harus dikonfigurasi ulang secara manual. Misalnya, pada gambar R1 memiliki **Static Routing** untuk mencapai jaringan 10.1.1.0/24 melalui R2. Jika jalur itu tidak lagi tersedia, R1 harus dikonfigurasi ulang dengan **Static Routing** baru ke jaringan 10.1.1.0/24 melalui R3. Oleh karena itu Router R3 perlu memiliki entri rute dalam tabel peruteannya untuk mengirim paket yang ditakdirkan untuk 10.1.1.0/24 ke R2.



Jika rute dari R1 melalui R2 tidak lagi tersedia, **Static Routing** baru melalui R3 perlu dikonfigurasi. **Static Routing** tidak secara otomatis menyesuaikan untuk perubahan topologi.

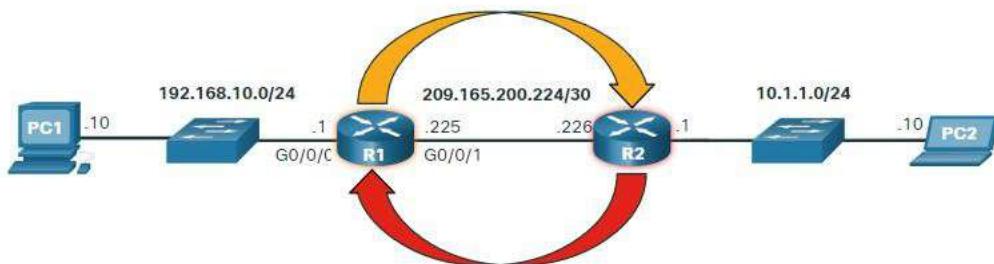
Static Routing memiliki karakteristik berikut:

- **Static Routing** harus dikonfigurasi secara manual.
- Administrator perlu mengkonfigurasi ulang **Static Routing** jika ada perubahan topologi dan **Static Routing** tidak lagi layak.
- **Static Routing** sesuai untuk jaringan kecil dan ketika ada sedikit atau tidak ada Link berlebihan.
- **Static Routing** umumnya digunakan dengan protokol **Dynamic Routing** untuk mengkonfigurasi **Default Route**.

Routing Dynamic

Dynamic Routing Protocol memungkinkan router untuk secara otomatis mempelajari tentang jaringan remote, termasuk Default Route, dari router lain. Router yang menggunakan Dynamic Routing Protocol secara otomatis berbagi Routing Information dengan router lain dan mengkompensasi perubahan topologi apa pun tanpa melibatkan administrator jaringan. Jika ada perubahan dalam topologi jaringan, router berbagi informasi ini menggunakan Dynamic Routing Protocol dan secara otomatis memperbarui Routing Table mereka.

Dynamic Routing Protocol mencakup OSPF dan Enhanced Interior Gateway Routing Protocol (EIGRP). Gambar menunjukkan contoh router R1 dan R2 secara otomatis berbagi informasi jaringan menggunakan Routing OSPF Protocol.

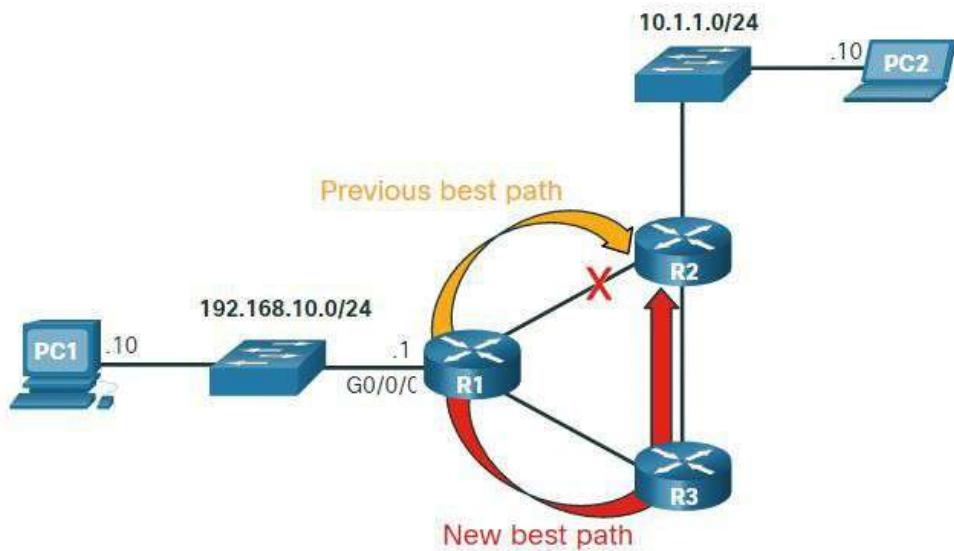


- R1 menggunakan **Routing OSPF Protocol** untuk memberi tahu R2 tentang jaringan 192.168.10.0/24.
- R2 menggunakan **Routing OSPF Protocol** untuk memberi tahu R1 tentang jaringan 10.1.1.0/24.

Konfigurasi dasar hanya mengharuskan administrator jaringan untuk mengaktifkan jaringan yang **Directly Connected** dalam **Dynamic Routing Protocol**. **Dynamic Routing Protocol Akan** secara otomatis dilakukan sebagai berikut:

- Menemukan Jaringan Remote
- Mempertahankan **Routing Information** terbaru
- Pilih jalur terbaik ke jaringan tujuan
- Mencoba menemukan link terbaik baru jika link saat ini tidak lagi tersedia

Ketika router dikonfigurasi secara manual dengan **Routing Static** atau belajar tentang jaringan remote secara dinamis menggunakan **Dynamic Routing Protocol**, alamat jaringan Remote dan alamat hop berikutnya dimasukkan ke dalam IP **Routing Table**. Seperti yang ditunjukkan pada gambar, jika ada perubahan dalam topologi jaringan, router akan secara otomatis menyesuaikan dan berusaha menemukan jalur terbaik baru.



R1, R2, dan R3 menggunakan **Dynamic Routing Protocol OSPF**. Jika ada perubahan topologi jaringan, mereka dapat secara otomatis menyesuaikan diri untuk menemukan jalur terbaik baru.

Catatan: umum bagi beberapa router untuk menggunakan kombinasi kedua **Routing Static** dan **Dynamic Routing Protocol**.

BAB 9

~ *Address Resolution* ~

Judul Bab : Address Resolution

Tujuan Bab : Menjelaskan bagaimana ARP dan ND mengaktifkan komunikasi antara jaringan.

Link Test Pemahaman : <https://s.id/-Qxtm>

| Judul Materi | Tujuan Materi |
|--------------------------------|---|
| MAC dan IP | Membandingkan peran antara MAC Address dan IP Address |
| Cara Kerja ARP | Menjelaskan tujuan dibuatnya ARP |
| Ipv6 Neighbor Discovery | Menjelaskan operasi dari Ipv6 Neighbor Discovery |

MAC dan IP

Terkadang host harus mengirim pesan, tetapi hanya mengetahui **Destination IP Address** Perangkat. Host perlu mengetahui MAC Address perangkat itu, tetapi bagaimana itu dapat ditemukan? Di situlah **Address Resolution Protocol** menjadi Penting.

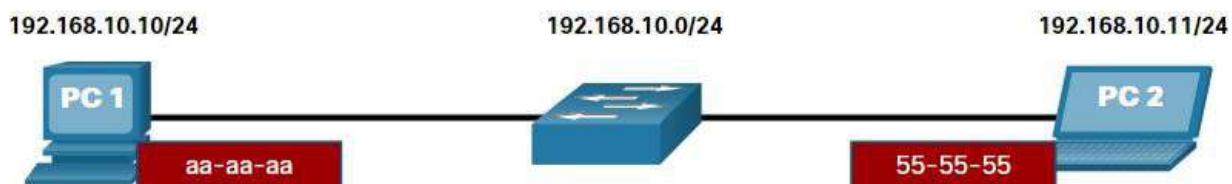
Tujuan pada Jaringan yang Sama

Ada dua alamat utama yang ditetapkan ke perangkat di LAN Ethernet:

- **Physical Address (MAC Address)** – Digunakan untuk komunikasi NIC ke NIC pada jaringan Ethernet yang sama.
- **Logical Address (IP Address)** – Digunakan untuk mengirim paket dari perangkat sumber ke perangkat tujuan. IP Address tujuan mungkin berada pada jaringan IP yang sama dengan sumber atau mungkin berada pada jaringan Remote.

Physical Address Lapisan 2 (yaitu, Ethernet MAC Address) digunakan untuk mengirimkan Data Link data dengan paket IP enkapsulasi dari satu NIC ke NIC lain yang berada di jaringan yang sama. Jika IP Address tujuan berada di jaringan yang sama, Destination MAC Address adalah alamat perangkat tujuan.

Pertimbangkan contoh berikut menggunakan representasi MAC Address yang disederhanakan.



| MAC Tujuan | MAC Sumber | Sumber IPv4 | Tujuan IPv4 |
|------------|------------|---------------|---------------|
| 55-55-55 | aa-aa-aa | 192.168.10.10 | 192.168.10.11 |

Dalam contoh ini, PC1 ingin mengirim paket ke PC2. Gambar menampilkan tujuan Layer 2 dan Source MAC Address dan IPv4 Address Layer 3 yang akan disertakan dalam paket yang dikirim dari PC1.

Frame Ethernet Layer 2 berisi yang berikut:

- **Destination MAC Address** - Ini adalah MAC Address yang disederhanakan dari PC2, 55-55-55.
- **Source MAC Address** – Ini adalah MAC Address yang disederhanakan dari Ethernet NIC di PC1, aa-aa-aa.

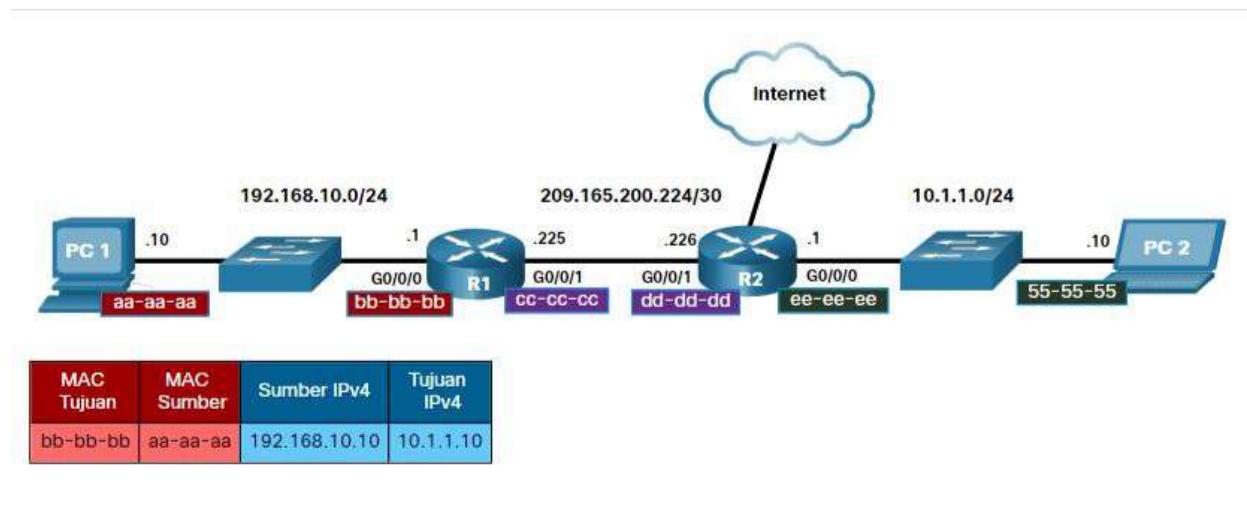
Paket IP Layer 3 berisi yang berikut:

- **Source IPv4 Address** – Ini adalah IP Addressv4 PC1, 192.168.10.10.
- **Destination IPv4 Address**– Ini adalah IP Addressv4 PC2, 192.168.10.11.

Tujuan pada Jaringan Remote

Ketika Destination IP Address (IPv4 atau IPv6) berada di jaringan Remote, Destination MAC Address akan menjadi alamat Default Gateway host (yaitu Interface router).

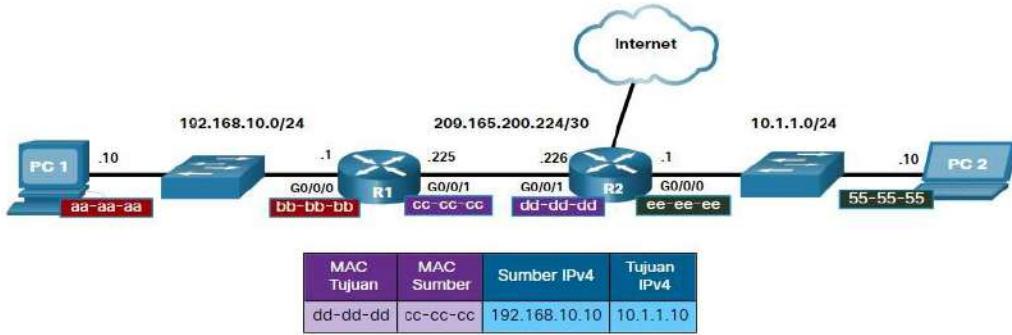
Pertimbangkan contoh berikut menggunakan representasi MAC Address yang disederhanakan.



Dalam contoh ini, PC1 ingin mengirim paket ke PC2. PC2 terletak di jaringan Remote. Karena Destination IPv4 Address tidak berada di jaringan lokal yang sama dengan PC1, Destination MAC Address adalah Default Gateway lokal pada router.

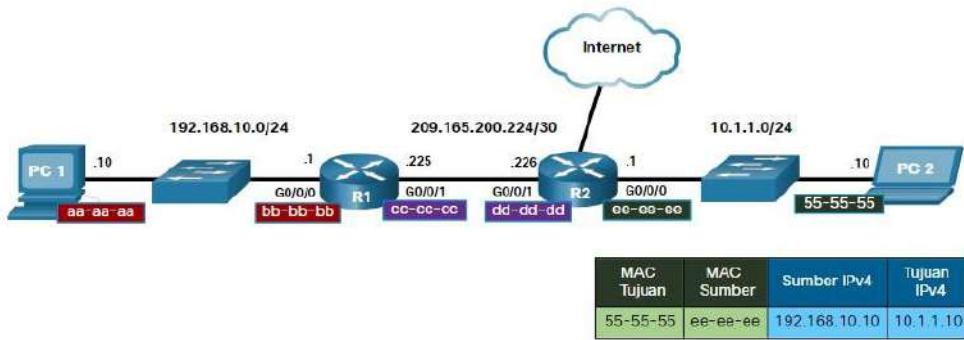
Router memeriksa Destination IPv4 Address untuk menentukan jalur terbaik untuk meneruskan paket IPv4. Ketika router menerima Frame Ethernet, router akan merangkum informasi Layer 2. Menggunakan Destination IPv4 Address, ia menentukan perangkat next-hop, dan kemudian merangkum paket **IPv4** dalam **Data Link** data baru untuk **Interface out**.

Dalam contoh kami, R1 sekarang akan merangkum paket dengan informasi alamat Layer 2 baru seperti yang ditunjukkan pada gambar.



Destination MAC Address baru adalah Interface R2 G0/0/1 dan Source MAC Address baru adalah Interface R1 G0/0/1.

Di sepanjang setiap Link dalam jalur, paket IP dienkapsulasi dalam Frame. Frame ini khusus untuk teknologi Data Link yang terkait dengan Link itu, seperti Ethernet. Jika perangkat next-hop adalah tujuan akhir, Destination MAC Address adalah perangkat Ethernet NIC, seperti yang ditunjukkan pada gambar.



Bagaimana IP Address paket IP dalam aliran data yang terkait dengan MAC Address di setiap Link di sepanjang jalur ke tujuan? Untuk paket IPv4, ini dilakukan melalui proses yang disebut **Address Resolution Protocol (ARP)**. Untuk paket IPv6, prosesnya adalah **ICMPv6 Neighbor Discovery (ND)**.

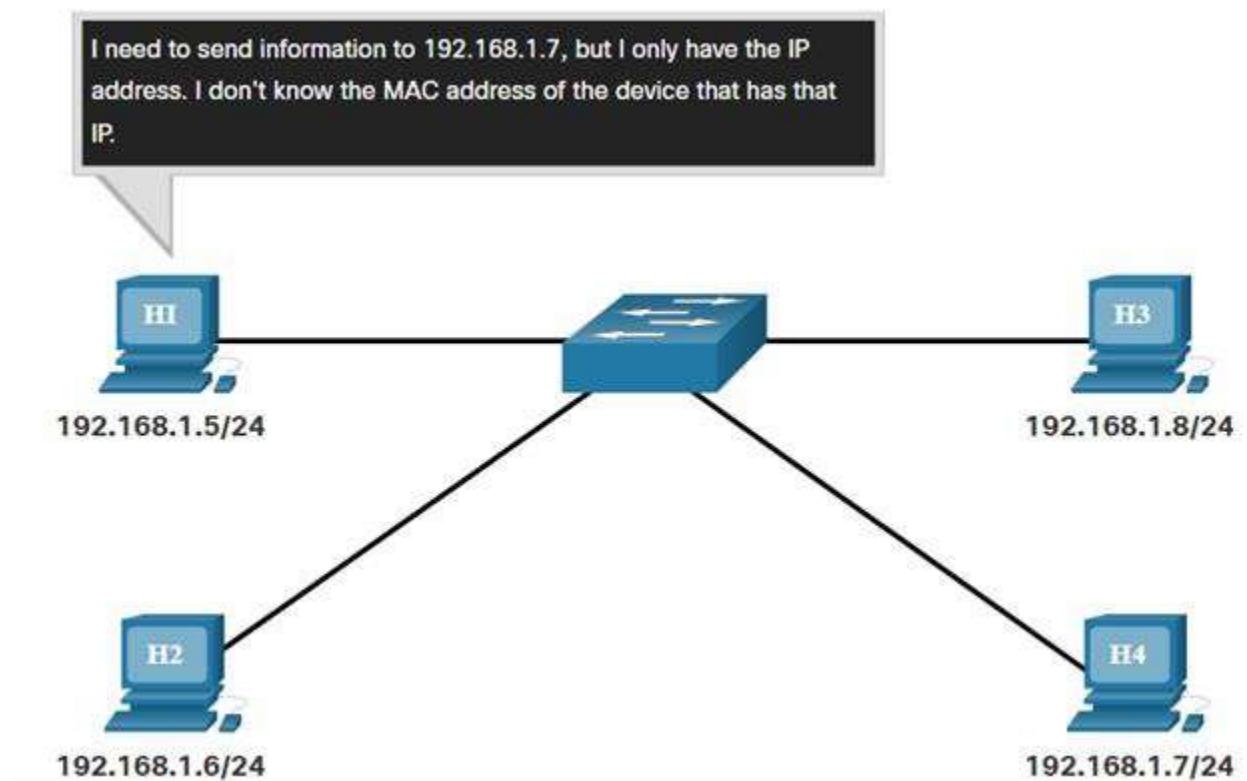
Cara Kerja ARP

Jika jaringan Anda menggunakan protokol komunikasi IPv4, **Address Resolution Protocol**, atau ARP, adalah yang Anda butuhkan untuk memetakan **IPv4 Address** ke **MAC Address**. Materi ini menjelaskan cara kerja ARP.

Setiap perangkat IP di jaringan Ethernet memiliki **Ethernet MAC Address** yang unik. Ketika perangkat mengirim **Frame** Ethernet Layer 2, perangkat berisi dua alamat ini:

- **Destination MAC Address** – Ethernet MAC Address perangkat tujuan pada segmen jaringan lokal yang sama. Jika host tujuan ada di jaringan lain, maka Destination Address dalam Frame adalah Default Gateway (yaitu, router).
- **Source MAC Address** – Ethernet MAC Address NIC pada host sumber.

Angka tersebut menggambarkan masalah saat mengirim **Frame** ke host lain pada segmen yang sama pada jaringan **IPv4**.



Untuk mengirim paket ke host lain di jaringan IPv4 lokal yang sama, host harus mengetahui IPv4 Address dan Destination MAC Address perangkat. Destination IPv4 Address perangkat diketahui atau diselesaikan dengan nama perangkat. Namun, MAC Address harus ditemukan.

Perangkat menggunakan Address Resolution Protocol (ARP) untuk menentukan Destination MAC Address perangkat lokal saat mengetahui IPv4 Address-nya.

ARP menyediakan dua fungsi dasar:

- Mengatasi IPv4 Address ke MAC Address
- Mempertahankan tabel pemetaan IPv4 Address ke MAC

Fungsi ARP

Ketika paket dikirim ke **Data Link Layer** untuk dienkapsulasi ke dalam **Frame Ethernet**, perangkat merujuk ke tabel dalam memorinya untuk menemukan **MAC Address** yang dipetakan ke **IPv4 Address**. Tabel ini disimpan sementara dalam memori RAM dan disebut tabel **ARP** atau **cache ARP**.

Perangkat pengirim akan mencari tabel **ARP**-nya untuk **Destination IPv4 Address** dan **MAC Address** yang sesuai.

- Jika Destination IPv4 Address paket berada di jaringan yang sama dengan Source IPv4 Address, perangkat akan mencari tabel ARP untuk Destination IPv4 Address.
- Jika Destination IPv4 Address berada di jaringan yang berbeda dari Source IPv4 Address, perangkat akan mencari tabel ARP untuk IPv4 Address Default Gateway.

Dalam kedua kasus, pencarian adalah untuk **IPv4 Address** dan **MAC Address** yang sesuai untuk perangkat.

Setiap entri, atau baris, tabel ARP mengikat IPv4 Address dengan MAC Address. Kami menyebut hubungan antara dua nilai peta. Ini hanya berarti Bahwa Anda dapat menemukan IPv4 Address dalam tabel dan menemukan MAC Address yang sesuai. Tabel ARP untuk sementara menyimpan (cache) pemetaan untuk perangkat di LAN.

Jika perangkat menemukan IPv4 Address, MAC Address yang sesuai digunakan sebagai Destination MAC Address dalam Frame. Jika tidak ada entri yang ditemukan, maka perangkat akan mengirimkan ARP Request.

ARP Request

ARP Request dikirim ketika perangkat perlu menentukan MAC Address yang terkait dengan IPv4 Address, dan tidak memiliki entri untuk IPv4 Address dalam tabel ARP-nya.

Pesan ARP dienkapsulasi langsung dalam Frame Ethernet. Tidak ada header IPv4. ARP Request dienkapsulasi dalam Frame Ethernet menggunakan informasi header berikut:

- **Destination MAC Address** – Ini adalah Broadcast Address FF-FF-FF-FF-FF-FF yang mengharuskan semua NIC Ethernet di LAN untuk menerima dan memproses ARP Request.
- **Source MAC Address** – Ini adalah MAC Address pengirim ARP Request.
- **Type** – Pesan ARP memiliki Field type 0x806. Ini menginformasikan penerimaan NIC bahwa bagian data Frame perlu diteruskan ke proses ARP.

Karena ARP Request adalah broadcast, mereka kebanjiran semua port oleh switch, kecuali port penerima. Semua NIC Ethernet pada broadcast proses LAN dan harus mengirimkan ARP Request ke sistem operasinya untuk diproses. Setiap perangkat harus memproses ARP Request untuk melihat apakah IPv4 Address target cocok dengan alamatnya sendiri. Router tidak akan meneruskan broadcast keluar interface lain.

Hanya satu perangkat di LAN yang akan memiliki IPv4 Address yang cocok dengan IPv4 Address target dalam ARP Request. Semua perangkat lain tidak akan membala.

ARP Reply

Hanya perangkat dengan IPv4 Address target yang terkait dengan ARP Request yang akan merespons dengan Arp Reply. Arp Reply dienkapsulasi dalam Frame Ethernet menggunakan informasi header berikut:

- **Destination MAC Address** – Ini adalah MAC Address pengirim ARP Request.
- **Source MAC Address** – Ini adalah MAC Address pengirim Arp Reply.
- **Type** – Pesan ARP memiliki Field type 0x806. Ini menginformasikan penerimaan NIC bahwa bagian data Frame perlu diteruskan ke proses ARP.

Hanya perangkat yang awalnya mengirim ARP Request yang akan menerima Arp Reply unicast. Setelah Arp Reply diterima, perangkat akan menambahkan IPv4 Address dan MAC Address yang sesuai ke tabel ARP-nya. Paket yang ditakdirkan untuk IPv4 Address tersebut sekarang dapat dienkapsulasi dalam Frame menggunakan MAC Address yang sesuai.

Jika tidak ada perangkat yang merespons ARP Request, paket akan di drop karena Frame tidak dapat dibuat.

Entri dalam tabel ARP dicap waktu. Jika perangkat tidak menerima **Frame** dari perangkat tertentu sebelum cap waktu kedaluwarsa, entri untuk perangkat ini dihapus dari tabel ARP.

Selain itu, entri *map static* dapat dimasukkan dalam tabel ARP, tetapi ini jarang dilakukan. Entri tabel ARP static tidak kadaluarsa dari waktu ke waktu dan harus dihapus secara manual.

Catatan: IPv6 menggunakan proses yang mirip dengan ARP untuk IPv4, yang dikenal sebagai ICMPv6 Neighbor Discovery (ND). IPv6 menggunakan neighbor solicitation dan neighbor advertisement messages, mirip dengan ARP Request IPv4 dan Arp Reply.

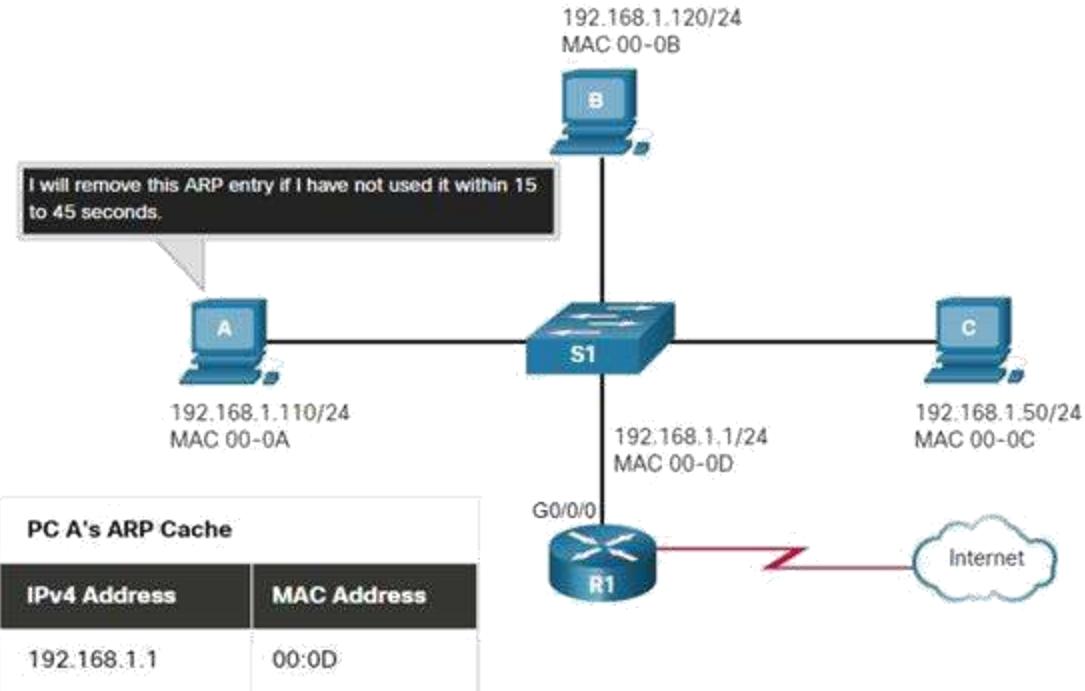
Peran ARP dalam Komunikasi Jarak Jauh

Ketika Destination IPv4 Address tidak berada di jaringan yang sama dengan Source IPv4 Address, perangkat sumber perlu mengirim Frame ke Default Gatewaynya. Ini adalah interface router lokal. Setiap kali perangkat sumber memiliki paket dengan IPv4 Address di jaringan lain, itu akan merangkum paket itu dalam Frame menggunakan Destination MAC Address router.

IPv4 Address Default Gateway disimpan dalam konfigurasi IPv4 host. Ketika host membuat paket untuk tujuan, itu membandingkan Tujuan IPv4 Address dan IPv4 Address sendiri untuk menentukan apakah kedua IPv4 Address terletak di jaringan Layer 3 yang sama. Jika host tujuan tidak berada di jaringan yang sama, sumber memeriksa tabel ARP-nya untuk entri dengan IPv4 Address Default Gateway. Jika tidak ada entri, ia menggunakan proses ARP untuk menentukan MAC Address Default Gateway.

Menghapus Entri dari Tabel ARP

Untuk setiap perangkat, **timer cache ARP** akan menghapus **entri ARP** yang belum digunakan untuk jangka waktu tertentu. Waktu berbeda tergantung pada **sistem operasi** perangkat. Misalnya, **sistem operasi** Windows yang lebih baru menyimpan entri tabel ARP antara 15 dan 45 detik, seperti yang diilustrasikan dalam gambar.



Note: MAC addresses are shortened for demonstration purposes.

Perintah juga dapat digunakan untuk menghapus beberapa atau semua entri secara manual dalam **tabel ARP**. Setelah entri dihapus, proses untuk mengirim **ARP Request** dan menerima **Arp Reply** harus terjadi lagi untuk memasukkan peta dalam **tabel ARP**.

Tabel ARP pada Perangkat Jaringan

Pada router Cisco, perintah **show ip arp** digunakan untuk menampilkan tabel ARP, seperti yang ditunjukkan pada gambar.

```
R1# show ip arp
Protocol Address          Age (min) Hardware Addr Type   Interface
Internet 192.168.10.1      -     a0e0.af0d.e140 ARPA   GigabitEthernet0/0/0
Internet 209.165.200.225    -     a0e0.af0d.e141 ARPA   GigabitEthernet0/0/1
Internet 209.165.200.226    1     a03d.6fe1.9d91 ARPA   GigabitEthernet0/0/1
R1#
```

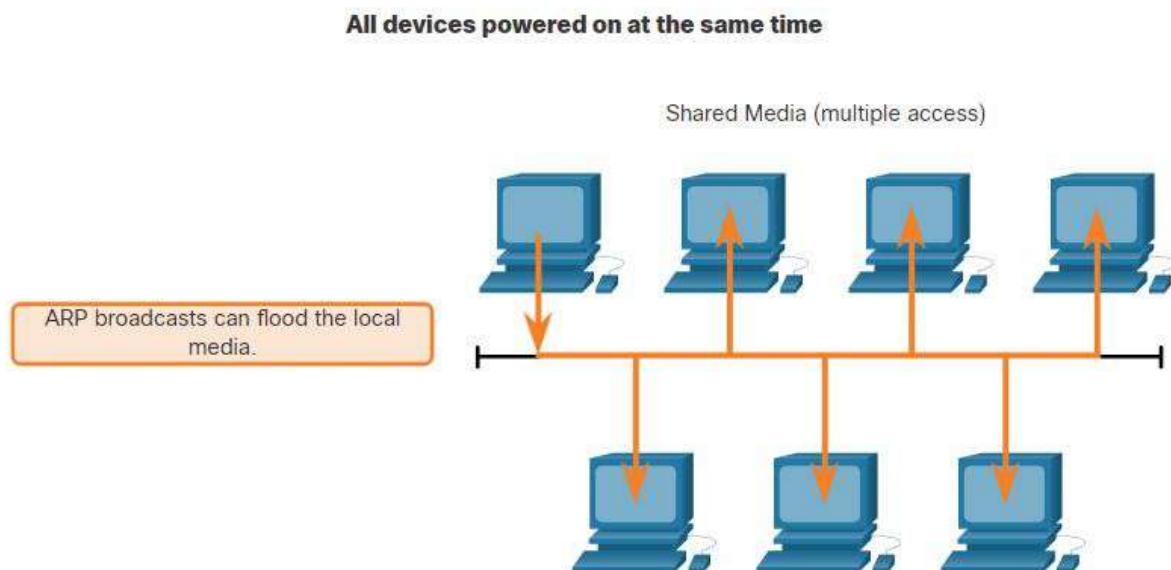
Pada PC Windows 10, **arp -a** digunakan untuk menampilkan tabel ARP, seperti yang ditunjukkan pada gambar.

```
C:\Users\Haekal>arp -a

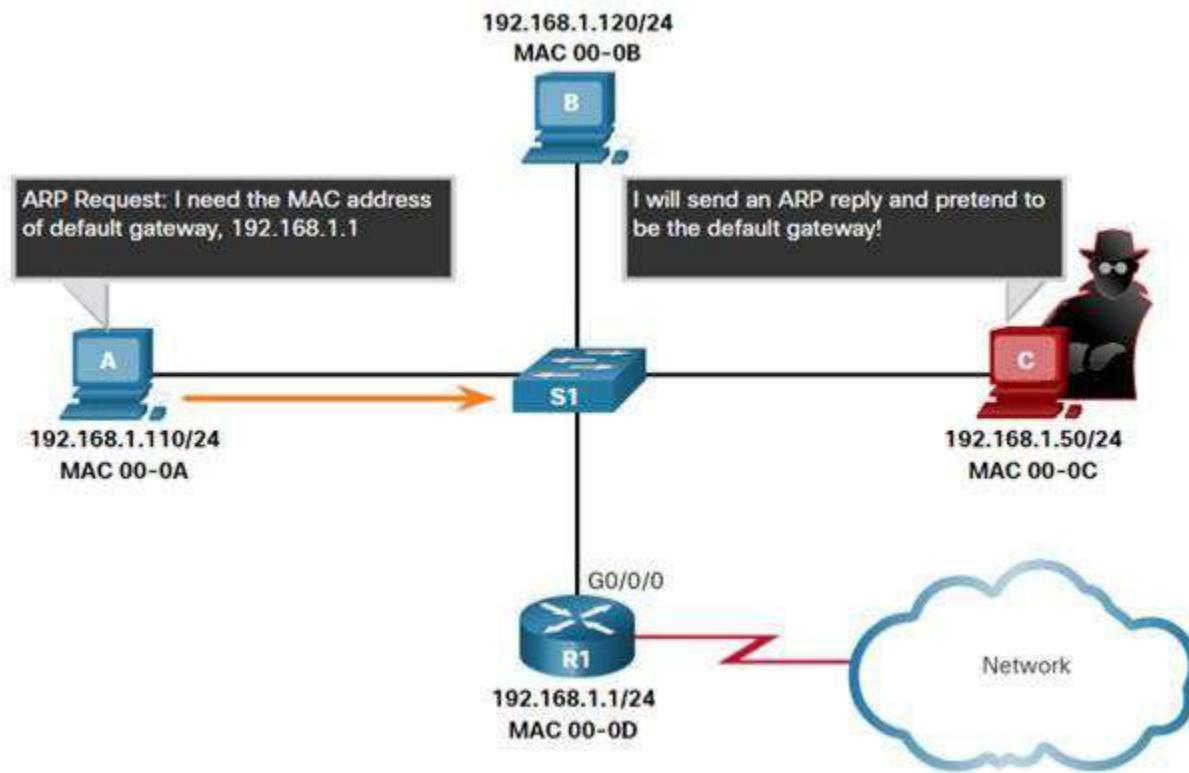
Interface: 192.168.15.2 --- 0x8
  Internet Address      Physical Address      Type
  192.168.15.1          b8-69-f4-86-38-33    dynamic
  192.168.15.7          ff-ff-ff-ff-ff-ff    static
  224.0.0.22             01-00-5e-00-00-16    static
  224.0.0.251            01-00-5e-00-00-fb    static
  224.0.0.252            01-00-5e-00-00-fc    static
  239.255.102.18         01-00-5e-7f-66-12    static
  239.255.255.250        01-00-5e-7f-ff-fa    static
  255.255.255.255        ff-ff-ff-ff-ff-ff    static
```

Masalah ARP – Broadcast ARP dan Spoofing ARP

Sebagai **Broadcast Frame**, **ARP Request** diterima dan diproses oleh setiap perangkat di jaringan lokal. Pada jaringan bisnis biasa, **broadcast** ini mungkin akan berdampak minimal pada kinerja jaringan. Namun, jika sejumlah besar perangkat harus ditenagai dan semua mulai mengakses layanan jaringan pada saat yang sama, mungkin ada beberapa pengurangan kinerja untuk waktu yang singkat, seperti yang ditunjukkan pada gambar. Setelah perangkat mengirimkan **broadcast ARP** awal dan telah mempelajari **MAC Address** yang diperlukan, dampak apa pun pada jaringan akan diminimalkan.



Dalam beberapa kasus, penggunaan ARP dapat menyebabkan potensi risiko keamanan. Seorang hacker dapat menggunakan spoofing ARP untuk melakukan serangan Poisoning ARP. Ini adalah teknik yang digunakan oleh Hacker untuk membalas ARP Request untuk IPv4 Address milik perangkat lain, seperti Default Gateway, seperti yang ditunjukkan pada gambar. Hacker mengirim Arp Reply dengan MAC Address-nya sendiri. Penerima Arp Reply akan menambahkan MAC Address yang salah ke tabel ARP-nya dan mengirim paket ini ke Hacker.



Note: MAC addresses are shortened for demonstration purposes.

IPv6 Neighbor Discovery

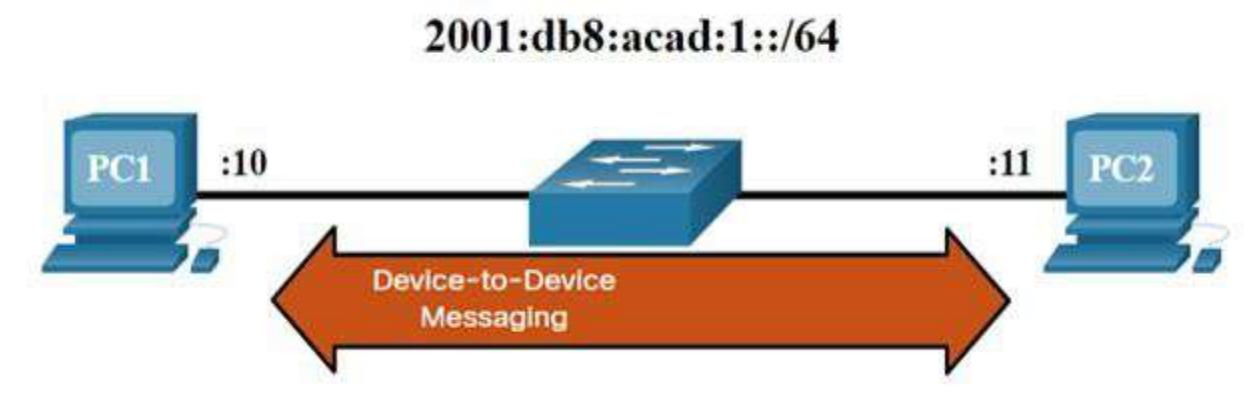
Jika jaringan Anda menggunakan protokol komunikasi IPv6, protokol Neighbor discovery, atau ND, adalah yang Anda butuhkan untuk mencocokkan **IPv6 Address** ke **MAC Address**.

IPv6 Neighbor Discovery Messages

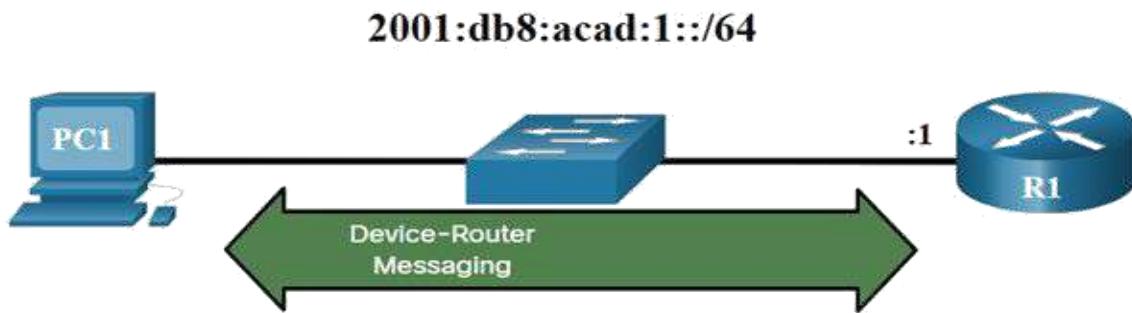
IPv6 Neighbor Discovery Protocol terkadang disebut sebagai ND atau NDP. ND menyediakan **Address resolusi**, **Router Discovery**, dan **Redirection Services** untuk IPv6 menggunakan **ICMPv6**. ICMPv6 ND menggunakan lima pesan **ICMPv6** untuk melakukan layanan ini:

- Neighbor Solicitation messages
- Neighbor Advertisement messages
- Router Solicitation messages
- Router Advertisement messages
- Redirect Message

Neighbor Solicitation dan **Neighbor Advertisement messages** digunakan untuk pesan perangkat ke perangkat seperti **Address Resolution** (mirip dengan ARP untuk **IPv4**). Perangkat termasuk komputer **host** dan **router**.



Router Solicitation dan **Router Advertisement messages** adalah untuk pesan antara perangkat dan router. Biasanya **router discovery** digunakan untuk **dynamic address allocation** dan **stateless address autoconfiguration (SLAAC)**

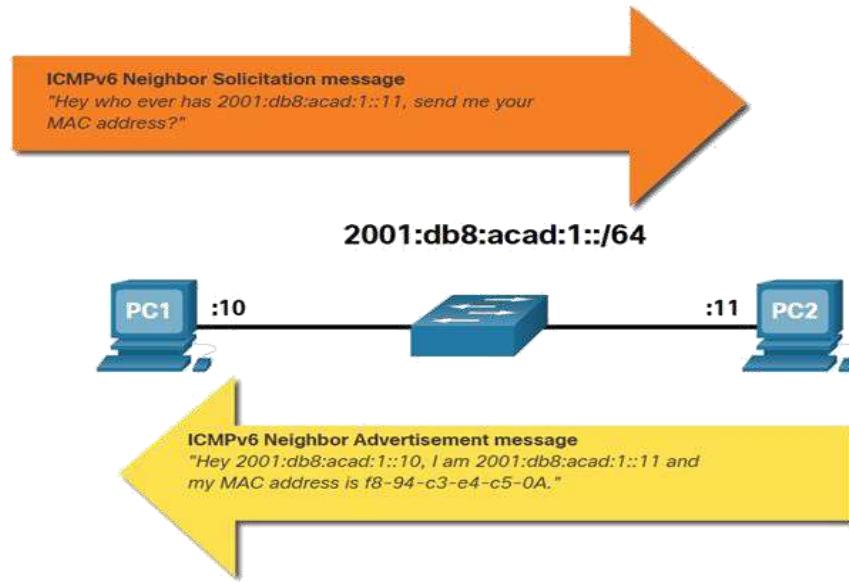


Catatan: Pesan ICMPv6 ND kelima adalah Redirect Message yang digunakan untuk pemilihan next-hop yang lebih baik. Ini di luar lingkup Materi ini.

IPv6 ND didefinisikan dalam IETF RFC 4861.

IPv6 Neighbor Discovery – Address Resolution

Sama seperti ARP untuk IPv4, perangkat IPv6 menggunakan IPv6 ND untuk menentukan MAC Address perangkat yang memiliki IPv6 Address yang dikenal.



ICMPv6 Neighbor Solicitation messages dikirim menggunakan Ethernet Special dan ICMPv6 Neighbor Solicitation messages . Ini memungkinkan Ethernet NIC dari perangkat penerima untuk menentukan apakah Neighbor Solicitation message adalah untuk dirinya sendiri tanpa harus mengirimnya ke sistem operasi untuk diproses.

PC2 membalas permintaan tersebut dengan ICMPv6 Neighbor Advertisement yang menyertakan MAC Address-nya.

BAB 10

~ Basic Router Configuration ~

Judul Bab : Basic Router Configuration

Tujuan Bab : Mengimplementasikan settingan awal untuk router dan end device

Link Test Pemahaman : <https://s.id/Qxw4>

| Judul Materi | Tujuan Materi |
|---------------------------------------|---|
| Mengkonfigurasi Settingan awal router | Mengkonfigurasi settingan awal di Cisco Router |
| Mengkonfigurasi Interface | Mengaktifkan 2 interface di cisco router |
| Mengkonfigurasi Default Gateway | Mengkonfigurasi perangkat untuk menggunakan default gateway |

Mengkonfigurasi Settingan awal router

Langkah-langkah Konfigurasi Router Dasar

1. Mengonfigurasi nama perangkat.

```
Router(config)# hostname hostname
```

2. Mengamankan privileged EXEC mode.

```
Router(config)# enable secret password
```

3. Mengamankan user EXEC mode.

```
Router(config)# line console 0
```

```
Router(config-line)# password password
```

```
Router(config-line)# login
```

4. Mengamankan akses Telnet / SSH jarak jauh.

```
Router(config-line)# line vty 0 4
```

```
Router(config-line)# password password
```

```
Router(config-line)# login
```

```
Router(config-line)# transport input {ssh | telnet}
```

5. Mengamankan semua kata sandi dalam file konfigurasi.

```
Router(config-line)# exit
```

```
Router(config)# service password-encryption
```

6. Memberikan Peringatan hukum

```
Router(config)# banner motd delimiter message delimiter
```

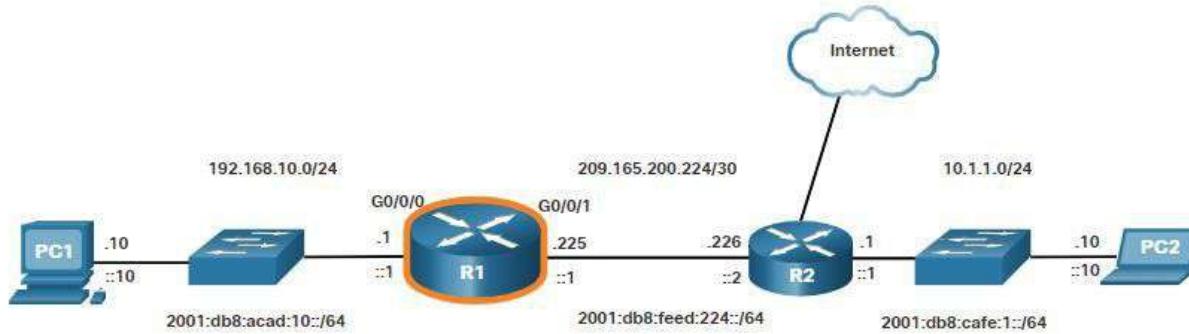
7. Simpan konfigurasi.

```
Router(config)# end
```

```
Router# copy running-config startup-config
```

Contoh Konfigurasi Dasar Routing

Dalam contoh ini, router R1 dalam diagram topologi akan dikonfigurasi dengan pengaturan awal.



Untuk mengkonfigurasi nama perangkat untuk R1, gunakan perintah berikut.

```
Router> enable
Router# configure terminal
Enter configuration commands, one per line.
End with CNTL/Z.
Router(config)# hostname R1
R1(config)#
```

Catatan: Perhatikan bagaimana prompt router sekarang menampilkan nama host router.

Semua akses router harus diamankan. **Privileged EXEC mode** memberi pengguna akses lengkap ke perangkat dan konfigurasinya. Oleh karena itu, ini adalah mode yang paling penting untuk diamankan.

Perintah berikut ini mengamankan **Privileged EXEC mode** dan **user EXEC mode**, mengaktifkan akses Telnet dan SSH, dan mengenkripsi semua kata sandi plaintext (yaitu, user EXEC dan VTY line).

```
R1(config)# enable secret class
R1(config)#
R1(config)# line console 0
R1(config-line)# password cisco
R1(config-line)# login
R1(config-line)# exit
R1(config)#
R1(config)# line vty 0 4
R1(config-line)# password cisco
R1(config-line)# login
R1(config-line)# transport input ssh telnet
R1(config-line)# exit
R1(config)#
R1(config)# service password-encryption
R1(config)#

```

Pemberitahuan hukum memperingatkan pengguna bahwa perangkat hanya boleh diakses oleh pengguna yang diizinkan. Pemberitahuan hukum dikonfigurasi sebagai berikut.

```
R1(config)# banner motd #
Enter TEXT message. End with a new line and the #
*****
WARNING: Unauthorized access is prohibited!
*****
#
R1(config)#

```

Jika perintah sebelumnya dikonfigurasi dan router kehilangan daya secara tidak sengaja, semua perintah yang dikonfigurasi akan hilang. Untuk alasan ini, penting untuk menyimpan konfigurasi ketika perubahan diterapkan. Perintah berikut menyimpan konfigurasi ke NVRAM.

```
R1# copy running-config startup-config
Destination filename [startup-config]?
Building configuration... [OK]
R1#

```

Mengkonfigurasi Interface

Pada Materi ini, router Anda sudah memiliki konfigurasi dasarnya. Langkah selanjutnya adalah mengkonfigurasi Interface mereka. Ini karena router tidak dapat dijangkau oleh **End Devices** hingga **Interface** dikonfigurasi. Ada banyak jenis **Interface** yang tersedia pada router Cisco. Misalnya, router Cisco ISR 4321 dilengkapi dengan dua **Interface** Ethernet Gigabit:

- **GigabitEthernet 0/0/0 (G0/0/0)**
- **GigabitEthernet 0/0/1 (G0/0/1)**

Mengkonfigurasi Router Interface

Tugas untuk mengkonfigurasi **router Interface** sangat mirip dengan **Management SVI** pada **Switch**. Secara khusus, ini termasuk mengeluarkan perintah berikut:

```
Router(config)# interface type-and-number  
Router(config-if)# description description-text  
Router(config-if)# ip address ipv4-address subnet-mask  
Router(config-if)# ipv6 address ipv6-address/prefix-length  
Router(config-if)# no shutdown
```

Catatan: Ketika **Router Interface** diaktifkan, pesan informasi harus ditampilkan mengkonfirmasi **link** yang diaktifkan.

Meskipun perintah **deskripsi** tidak diperlukan untuk mengaktifkan interface, itu adalah praktik yang baik untuk menggunakan. Ini dapat membantu dalam pemecahan masalah pada jaringan produksi dengan memberikan informasi tentang jenis jaringan yang terhubung. Misalnya, jika **interface** terhubung ke ISP atau operator layanan, **perintah** deskripsi akan sangat membantu untuk memasukkan koneksi pihak ketiga dan informasi kontak.

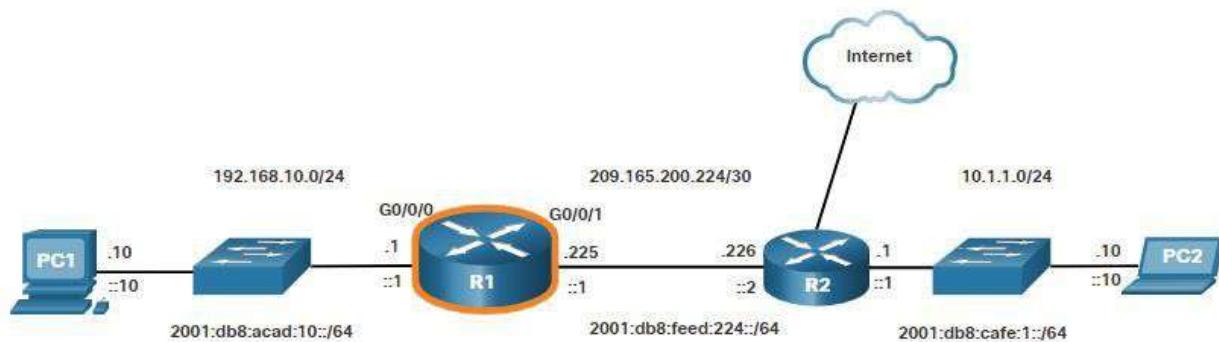
Catatan: *descriptive-text* dibatasi hingga 240 karakter.

Menggunakan **perintah no shutdown** mengaktifkan **interface**. **Interface** juga harus terhubung ke perangkat lain, seperti **switch** atau **router**, agar lapisan fisik aktif.

Catatan: Pada koneksi antar-router di mana tidak ada **switch Ethernet**, kedua **interface** yang saling terhubung harus dikonfigurasi dan diaktifkan.

Contoh Konfigurasi Router Interface

Dalam contoh ini, **Interface R1** yang **Direct-Connected** dalam diagram topologi akan diaktifkan.



Untuk mengkonfigurasi **interface** pada R1, gunakan perintah berikut.

```
R1> enable
```

```
R1# configure terminal
```

```
Enter configuration commands, one per line.
```

```
End with CNTL/Z.
```

```
R1(config)# interface gigabitEthernet 0/0/0
R1(config-if)# description Link to LAN
R1(config-if)# ip address 192.168.10.1 255.255.255.0
R1(config-if)# ipv6 address 2001:db8:acad:10::1/64
R1(config-if)# no shutdown
R1(config-if)# exit
```

```
R1(config)#
```

```
*Aug 1 01:43:53.435: %LINK-3-UPDOWN: Interface GigabitEthernet0/0/0, changed state to down
```

```
*Aug 1 01:43:56.447: %LINK-3-UPDOWN: Interface GigabitEthernet0/0/0, changed state to up
```

```
*Aug 1 01:43:57.447: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0/0, changed state to up
R1(config)#
```

```
R1(config)#
```

```

R1(config)# interface gigabitEthernet 0/0/1
R1(config-if)# description Link to R2
R1(config-if)# ip address 209.165.200.225 255.255.255.252
R1(config-if)# ipv6 address 2001:db8:feed:224::1/64
R1(config-if)# no shutdown
R1(config-if)# exit
R1(config)#
*Aug 1 01:46:29.170: %LINK-3-UPDOWN: Interface GigabitEthernet0/0/1, changed state
to down#
*Aug 1 01:46:32.171: %LINK-3-UPDOWN: Interface GigabitEthernet0/0/1, changed state
to up#
*Aug 1 01:46:33.171: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet0/0/1, changed state to up R1(config)#

```

Catatan: Perhatikan pesan informasi yang memberi tahu kami bahwa G0/0/0 dan G0/0/1 diaktifkan.

Verifikasi Konfigurasi Interface

Ada beberapa perintah yang dapat digunakan untuk memverifikasi konfigurasi Interface. Yang paling berguna dari ini adalah **show ip interface brief** dan **show ipv6 interface brief**, seperti yang ditunjukkan dalam contoh.

```

R1# show ip interface brief
Interface      IP-Address      OK? Method Status      Protocol
GigabitEthernet0/0/0  192.168.10.1  YES manual up          up
GigabitEthernet0/0/1  209.165.200.225 YES manual up          up
Vlan1           unassigned     YES unset administratively down down
R1# show ipv6 interface brief
GigabitEthernet0/0/0    [up/up]
  FE80::201:C9FF:FE89:4501
  2001:DB8:ACAD:10::1
GigabitEthernet0/0/1    [up/up]
  FE80::201:C9FF:FE89:4502
  2001:DB8:FEED:224::1
Vlan1           [administratively down/down]
  unassigned
R1#

```

Perintah Verifikasi Konfigurasi

Tabel meringkas perintah *show* yang lebih populer yang digunakan untuk memverifikasi konfigurasi **Interface**.

| Perintah | Deskripsi |
|--|---|
| show ip interface brief show ipv6 interface brief | Output menampilkan semua Interface , alamat IP mereka, dan statusnya saat ini. Interface yang dikonfigurasi dan terhubung harus menampilkan Status “up” dan Protokol “up”. Hal lain akan menunjukkan masalah dengan konfigurasi atau kabel. |
| show ip route show ipv6 route | Menampilkan konten IP routing tables yang disimpan dalam RAM. |
| show interfaces | Menampilkan statistik untuk semua interface pada perangkat. Namun, perintah ini hanya akan menampilkan informasi alamat IPv4. |
| show ip interfaces | Menampilkan statistik IPv4 untuk semua interface pada router . |
| show ipv6 interface | Menampilkan statistik IPv6 untuk semua interface pada router . |

Mengkonfigurasi Default Gateway

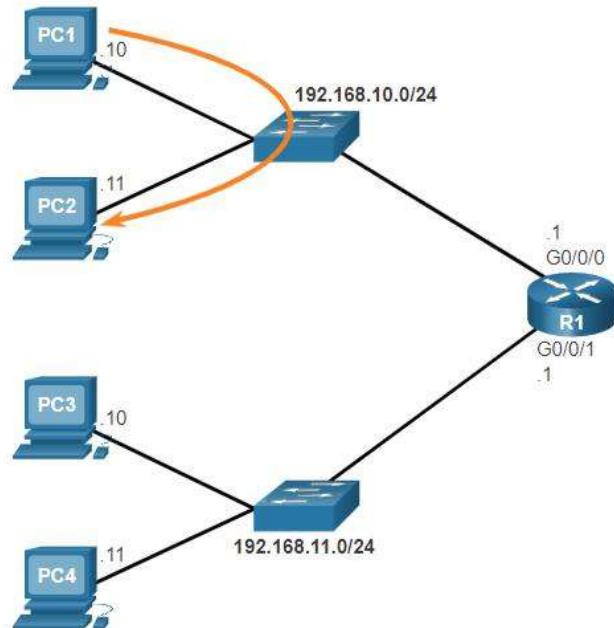
Jika jaringan lokal Anda memiliki beberapa router, Anda harus memilih salah satunya menjadi Router Default Gateway. Materi ini menjelaskan cara mengkonfigurasi Default Gateway pada host dan switch.

Default Gateway di Host

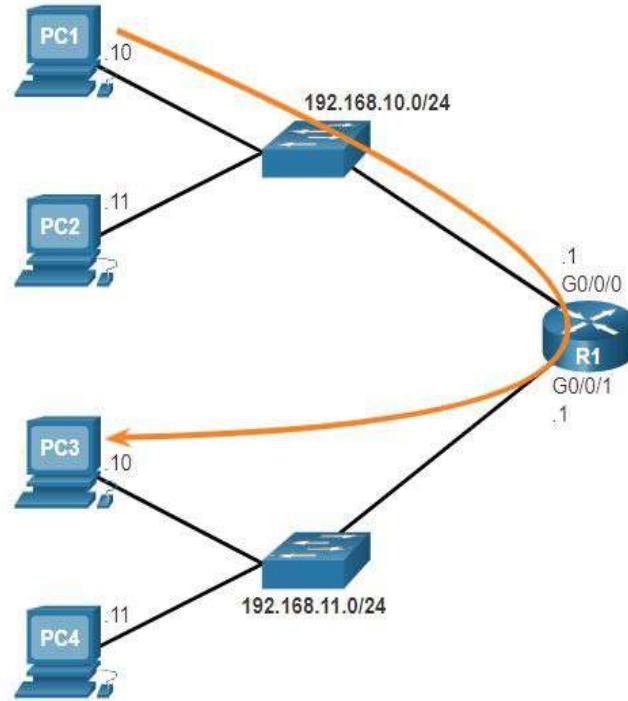
Agar end devices berkomunikasi melalui jaringan, perangkat harus dikonfigurasi dengan informasi alamat IP yang benar, termasuk alamat Default Gateway. Default Gateway hanya digunakan ketika host ingin mengirim paket ke perangkat di jaringan lain. Alamat Default Gateway umumnya adalah alamat interface router yang dilampirkan ke jaringan lokal host. Alamat IP perangkat host dan alamat interface router harus berada di jaringan yang sama.

Misalnya, asumsikan topologi jaringan IPv4 yang terdiri dari router yang menghubungkan dua LAN terpisah. G0/0/0 terhubung ke jaringan 192.168.10.0, sedangkan G0/0/1 terhubung ke jaringan 192.168.11.0. Setiap perangkat host dikonfigurasi dengan alamat Default Gateway yang sesuai.

Dalam contoh ini, jika PC1 mengirim paket ke PC2, maka Default Gateway tidak digunakan. Sebagai gantinya, PC1 membahas paket dengan alamat IPv4 PC2 dan meneruskan paket langsung ke PC2 melalui switch.



Bagaimana jika PC1 mengirim paket ke PC3? PC1 akan membalas paket dengan alamat IPv4 PC3, tetapi akan meneruskan paket ke Default Gateway, yang merupakan Interface G0/0/0 dari R1. Router menerima paket dan mengakses Routing Table untuk menentukan bahwa G0/0/1 adalah Out Interface yang sesuai berdasarkan Destination Address. R1 kemudian meneruskan paket keluar dari Interface yang sesuai untuk mencapai PC3.



Proses yang sama akan terjadi pada jaringan IPv6, meskipun ini tidak ditunjukkan dalam topologi. Perangkat akan menggunakan alamat IPv6 router lokal sebagai Default Gateway mereka.

Default Gateway pada Switch

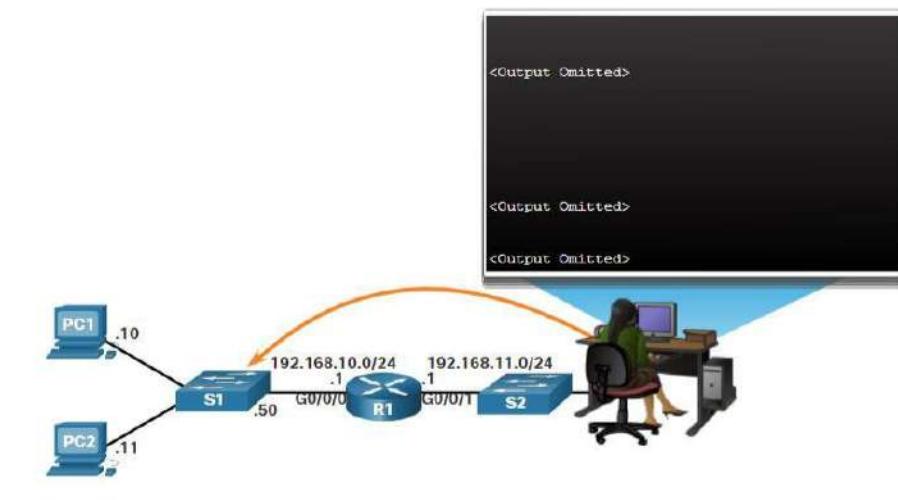
Switch yang menghubungkan komputer klien biasanya adalah perangkat Layer 2. Dengan demikian, Switch Layer 2 tidak memerlukan IP Address untuk berfungsi dengan baik. Namun, konfigurasi IP dapat dikonfigurasi pada Switch untuk memberikan akses remote administrator ke Switch.

Untuk menyambungkan dan mengelola Switch melalui jaringan IP lokal, switch virtual interface (SVI) dikonfigurasi. SVI dikonfigurasi dengan alamat IPv4 dan Subnetmask pada LAN lokal. Switch juga harus memiliki alamat Default Gateway yang dikonfigurasi untuk mengelola Switch dari jarak jauh oleh jaringan lain.

Alamat Default Gateway biasanya dikonfigurasi di semua perangkat yang akan berkomunikasi di luar jaringan lokal mereka.

Untuk mengkonfigurasi Default Gateway IPv4 pada Switch, gunakan perintah `ip default-gateway ip-address` pada global configuration. Alamat *ip* yang dikonfigurasi adalah alamat IPv4 Interface router lokal yang terhubung ke Switch.

Gambar menunjukkan administrator yang membuat koneksi Remote untuk Switch S1 pada jaringan lain.



Dalam contoh ini, host administrator akan menggunakan Default Gateway untuk mengirim paket ke interface R1 G0/0/1. R1 akan meneruskan paket ke S1 dari Interface G0/0/0. Karena alamat IPv4 sumber paket berasal dari jaringan lain, S1 akan memerlukan Default Gateway untuk meneruskan paket ke Interface R1 G0/0/0. Oleh karena itu, S1 harus dikonfigurasi dengan Default Gateway untuk dapat membala dan membuat koneksi SSH dengan host administratif.

Catatan: Paket yang berasal dari komputer host yang terhubung ke switch harus sudah memiliki alamat Default Gateway yang dikonfigurasi pada sistem operasi komputer host mereka.

Switch Workgroup juga dapat dikonfigurasi dengan alamat IPv6 pada SVI. Namun, Switch tidak memerlukan alamat IPv6 Default Gateway untuk dikonfigurasi secara manual. Switch akan secara otomatis menerima Default Gateway dari ICMPv6 Router Advertisement message dari router.

BAB 11

~ IPv4 Addressing ~

Judul Bab : Alamat IPv4

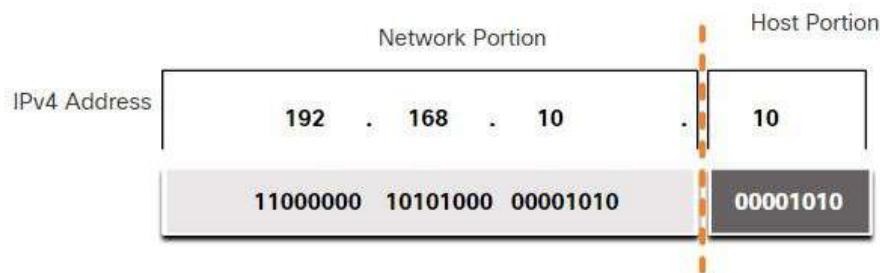
Tujuan Bab : Menghitung sebuah skema subnet dari IPv4 yang efisien

Link Test Pembahasan : <https://s.id/-Qxyc>

| Judul Materi | Tujuan Materi |
|--|--|
| Struktur Alamat IPv4 | Menjelaskan struktur dari alamat IPv4 termasuk <i>network portion</i> , <i>host portion</i> , dan <i>subnet mask</i> |
| IPv4 Unicast, Broadcast, dan Multicast | Membedangkan karakteristik dan menggunakan unicast broadcast dan multicast alamat IPv4 |
| Tipe Tipe Alamat IPv4 | Menjelaskan Public, private, reserved alamat IPv4 |
| Segmentasi Jaringan | Menjelaskan bagaimana segmen jaringan menjadikan komunikasi yang baik |
| Subnet Jaringan IPv4 | Menghitung IPv4 Subnet dari prefix /24 |
| Subnet /16 dan /8 | Menghitung IPv4 Subnet dari prefix /16 dan /8 |
| Persyaratan untuk membuat subnet | Memberikan serangkaian persyaratan untuk subnetting IPv4 |
| Variable Length Subnet Masking | Menjelaskan bagaimana membuat skema alamat yang flexible dengan menggunakan VLSM |
| Desain Terstruktur | Mengimplementasikan sebuah skema alamat VLSM |

Struktur Alamat IPv4

IPv4 Address adalah alamat hierarkis 32-bit yang terdiri dari Network Portion dan Host Portion. Saat menentukan Network Portion vs Host Portion, Anda harus melihat aliran 32-bit, seperti yang ditunjukkan pada gambar.



Network Portion dan Host IPv4 Address

Bit dalam Network Portion alamat harus identik untuk semua perangkat yang berada di jaringan yang sama. Bit dalam Host Portion alamat harus unik untuk mengidentifikasi Host tertentu dalam jaringan. Jika dua Host memiliki pola bit yang sama dalam Network Portion yang ditentukan dari aliran 32-bit, kedua Host tersebut akan berada di jaringan yang sama.

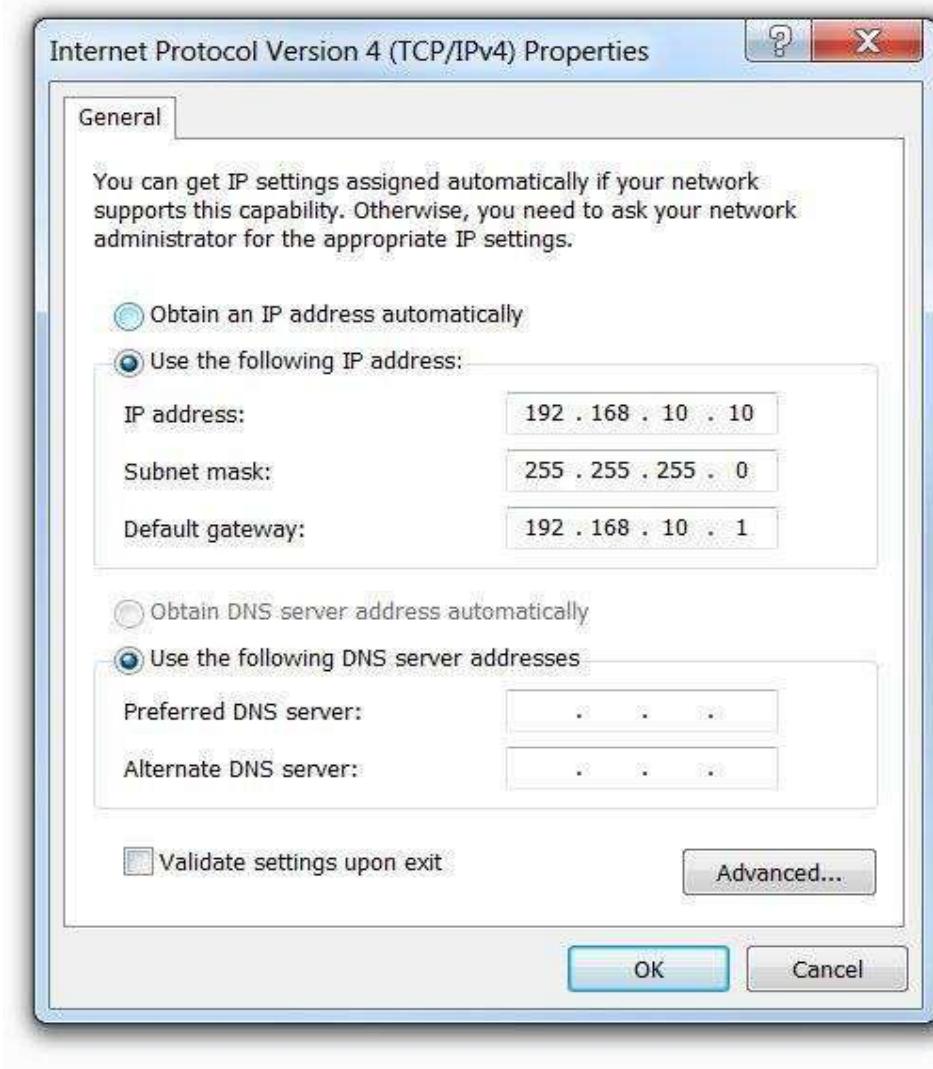
Tetapi bagaimana Host mengetahui bagian mana dari 32-bit yang mengidentifikasi jaringan dan yang mengidentifikasi Host? Itulah peran subnet mask.

Subnet mask

Seperti yang ditunjukkan pada gambar, menetapkan IPv4 Address ke Host memerlukan hal berikut:

- **IPv4 Address** – Ini adalah IPv4 Address unik dari Host.
- **Subnet mask** – Ini digunakan untuk mengidentifikasi Network Portion / Host dari IPv4 Address.

Konfigurasi IPv4 pada Komputer Windows

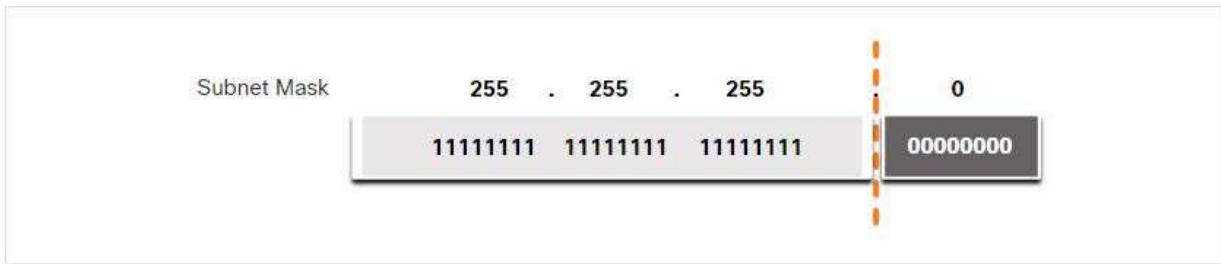


Catatan: IPv4 Address default gateway diperlukan untuk menjangkau jaringan remote dan IPv4 Address server DNS diperlukan untuk menerjemahkan nama domain ke IPv4 Address.

Subnet mask IPv4 digunakan untuk membedakan Network Portion dari Host Portion IPv4 Address. Ketika IPv4 Address ditetapkan ke perangkat, Subnet mask digunakan untuk menentukan Network Address perangkat. Network Address mewakili semua perangkat pada jaringan yang sama.

Gambar berikutnya menampilkan subnet mask 32-bit dalam format desimal dan dotted-decimal.

Subnet mask



Perhatikan bagaimana subnet mask adalah urutan berturut-turut 1 bit diikuti dengan urutan berturut-turut 0 bit.

Untuk mengidentifikasi Network Portion dan Host IPv4 Address, subnet mask dibandingkan dengan bit IPv4 Address untuk bit, dari kiri ke kanan seperti yang ditunjukkan pada gambar.

Mengaitkan IPv4 Address dengan Subnet Mask-nya

Perhatikan bahwa subnet mask sebenarnya tidak berisi Network Portion atau Host dari IPv4 Address, itu hanya memberi tahu komputer di mana mencari bagian dari IPv4 Address yang merupakan Network Portion dan bagian mana yang merupakan Host Portion .

Proses aktual yang digunakan untuk mengidentifikasi **Network Portion** dan **Host Portion** disebut **ANDing**.

The Prefix Length

Mengekspresikan Network Address dan alamat Host dengan alamat dotted-decimal subnet mask dapat menjadi rumit. Untungnya, ada metode alternatif untuk mengidentifikasi subnet mask, metode yang disebut Prefix Length.

Prefix Length adalah jumlah bit yang diatur ke 1 di subnet mask. Ini ditulis dalam “*slash notation*”, yang dikenal dengan garis miring maju (/) diikuti dengan jumlah bit yang diatur ke 1. Oleh karena itu, hitung jumlah bit dalam subnet mask dan prepend dengan garis miring.

Lihat tabel misalnya. Kolom pertama mencantumkan berbagai **subnet mask** yang dapat digunakan dengan alamat **Host**. Kolom kedua menampilkan alamat biner 32-bit yang dikonversi. Kolom terakhir menampilkan **Prefix Length** yang dihasilkan.

Membandingkan Subnet mask dan Prefix Length

| Subnet Mask | 32-bit Address | Prefix Length |
|--------------------|-------------------------------------|----------------------|
| 255.0.0.0 | 11111111.00000000.00000000.00000000 | /8 |
| 255.255.0.0 | 11111111.11111111.00000000.00000000 | /16 |
| 255.255.255.0 | 11111111.11111111.11111111.00000000 | /24 |
| 255.255.255.128 | 11111111.11111111.11111111.10000000 | /25 |
| 255.255.255.192 | 11111111.11111111.11111111.11000000 | /26 |
| 255.255.255.224 | 11111111.11111111.11111111.11100000 | /27 |
| 255.255.255.240 | 11111111.11111111.11111111.11110000 | /28 |
| 255.255.255.248 | 11111111.11111111.11111111.11111000 | /29 |
| 255.255.255.252 | 11111111.11111111.11111111.11111100 | /30 |

Catatan: Network Address juga disebut sebagai prefix atau prefix jaringan. Oleh karena itu, Prefix Length adalah jumlah 1 bit di subnet mask.

Saat mewakili IPv4 Address menggunakan **Prefix Length**, IPv4 Address ditulis diikuti dengan **Prefix Length** tanpa spasi. Misalnya, 192.168.10.10 255.255.255.0 akan ditulis sebagai 192.168.10.10/24. Menggunakan berbagai jenis **Prefix Length** akan dibahas nantinya. Untuk saat ini, fokusnya adalah pada prefix /24 (yaitu 255.255.255.0)

Menentukan Jaringan: LOGICAL AND

Logical AND adalah salah satu dari tiga operasi **Boolean** yang digunakan dalam **Boolean** atau **Digital Logic**. Dua lainnya adalah **OR** dan **NOT**. Operasi **AND** digunakan dalam menentukan Network Address.

Logika **AND** adalah perbandingan dua bit yang menghasilkan hasil yang ditunjukkan di bawah ini. Perhatikan bagaimana hanya 1 **AND** 1 yang menghasilkan 1. Kombinasi lainnya menghasilkan 0.

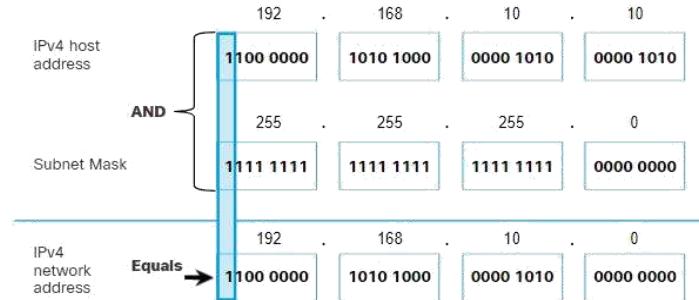
- 1DAN1=1
- 0DAN1=0
- 1DAN0=0
- 0DAN0=0

Catatan: Dalam **digital logic**, 1 mewakili **True** dan 0 mewakili **False**. Saat menggunakan operasi **AND**, kedua nilai input harus True (1) agar hasilnya True (1).

Untuk mengidentifikasi Network Address Host, IPv4 Address secara ANDed, bit demi bit, dengan subnet mask. ANDing antara alamat dan subnet mask menghasilkan Network Address.

Untuk menggambarkan bagaimana **AND** digunakan untuk menemukan **Network Address**, pertimbangkan **Host** dengan IPv4 Address 192.168.10.10 dan **subnet mask** 255.255.255.0, seperti yang ditunjukkan pada gambar:

- **Alamat Host IPv4 (192.168.10.10)** – IPv4 Address Host dalam format desimal dan dotted-decimal.
- **Subnet mask (255.255.255.0)** – Subnet mask Host dalam format desimal dan dotted-decimal.
- **Network Address (192.168.10.0)** – Operasi Logical AND antara IPv4 Address dan subnet mask menghasilkan Network Address IPv4 yang ditunjukkan dalam format desimal dan dotted-decimal.



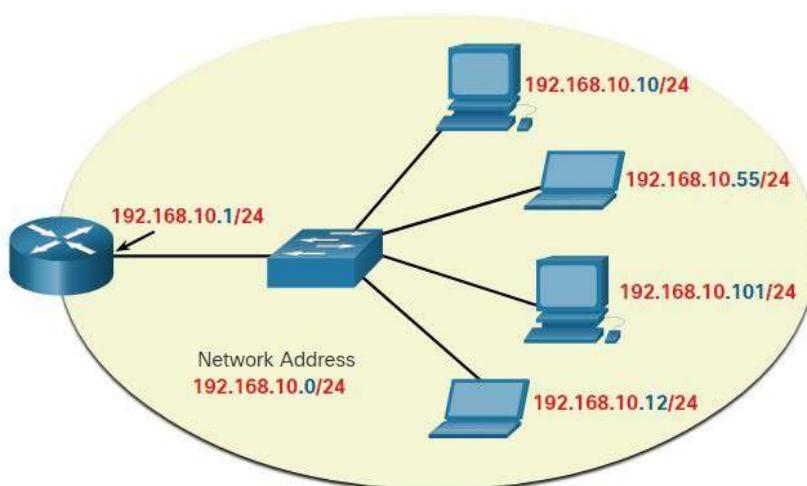
Menggunakan urutan pertama bit sebagai contoh, perhatikan operasi AND dilakukan pada 1-bit alamat Host dengan 1-bit subnet mask. Ini menghasilkan 1 bit untuk Network Address. 1 DAN 1 = 1.

Operasi AND antara alamat Host IPv4 dan subnet mask menghasilkan Network Address IPv4 untuk Host ini. Dalam contoh ini, operasi AND antara alamat Host 192.168.10.10 dan subnet mask 255.255.255.0 (/24), menghasilkan Network Address IPv4 192.168.10.0/24. Ini adalah operasi IPv4 yang penting, karena memberi tahu Host jaringan tempatnya berada.

Network Address, Host, dan Broadcast

Dalam setiap jaringan ada tiga jenis alamat IP:

- **Network Address**
- **Host Address**
- **Broadcast Address**



Network Address

Network Address adalah alamat yang mewakili jaringan tertentu. Perangkat milik jaringan ini jika memenuhi tiga kriteria:

- Ini memiliki **subnet mask** yang sama dengan **Network Address**.
- Ini memiliki bit jaringan yang sama dengan **Network Address**, seperti yang ditunjukkan oleh **subnet mask**.
- Terletak di domain **Broadcast** yang sama dengan **Host** lain dengan **Network Address** yang sama.

Host menentukan **Network Addressnya** dengan melakukan operasi AND antara IPv4 Address dan **subnet mask**

Seperti yang ditunjukkan dalam tabel, **Network Address** memiliki semua bit 0 dalam porsi **Host**, seperti yang ditentukan oleh **subnet mask**. Dalam contoh ini, **Network Address** adalah 192.168.10.0/24. **Network Address** tak bisa dialokasikan ke perangkat.

Network Address, Host, dan Broadcast

| | Network Portion | | | Host Portion | Host Bits |
|---|-----------------|-----------------|-----------------|-----------------|----------------|
| Subnet mask 255.255.255.0 or /24 | 255 11111111 | 255 11111111 | 255 11111111 | 0 00000000 | |
| Network address 192.168.10.0 or /24 | 192 11000000 | 168 10100000 | 10 00001010 | 0 00000000 | All 0s |
| First address 192.168.10.1 or /24 | 192 11000000 | 168 10100000 | 10 00001010 | 1 00000001 | All 0s and a 1 |
| Last address 192.168.10.254 or /24 | 192 11000000 | 168 10100000 | 10 00001010 | 254 11111110 | All 1s and a 0 |
| Broadcast address 192.168.10.255 or /24 | 192 11000000 | 168 10100000 | 10 00001010 | 255 11111111 | All 1s |

Host Address

Host Address adalah alamat yang dapat ditetapkan ke perangkat seperti komputer, laptop, ponsel pintar, kamera web, printer, router, dll. **Host Portion** alamat adalah bit yang ditunjukkan oleh 0 bit di **subnet mask**. Alamat **Host** dapat memiliki kombinasi bit dalam porsi **Host** kecuali untuk semua bit 0 (ini akan menjadi **Network Address**) atau semua 1 bit (ini akan menjadi alamat **Broadcast**).

Semua perangkat dalam jaringan yang sama, harus memiliki **subnet mask** yang sama dan bit jaringan yang sama. Hanya bit **Host** yang akan berbeda dan harus unik.

Perhatikan bahwa dalam tabel, ada alamat **Host** pertama dan terakhir:

- **First Host Addresses** – **First Host** ini dalam jaringan memiliki semua bit 0 dengan bit terakhir (paling kanan) sebagai 1 bit. Dalam contoh ini adalah 192.168.10.1/24.
- **Last Host Addresses** – **Last Host** ini dalam jaringan memiliki semua 1 bit dengan bit terakhir (paling kanan) sebagai 0 bit. Dalam contoh ini adalah 192.168.10.254/24.

Alamat apa pun antara dan termasuk, 192.168.10.1/24 hingga 192.168.10.254/24 dapat ditetapkan ke perangkat di jaringan.

Broadcast Address

Broadcast Address adalah alamat yang digunakan saat diperlukan untuk menjangkau semua perangkat di jaringan IPv4. Seperti yang ditunjukkan dalam tabel, **Broadcast Address** jaringan memiliki semua 1 bit dalam porsi **Host**, seperti yang ditentukan oleh **subnet mask**. Dalam contoh ini, **Network Address** adalah 192.168.10.255/24. Alamat **Broadcast** tidak dapat ditetapkan ke perangkat.

IPv4 Unicast, Broadcast, Dan Multicast

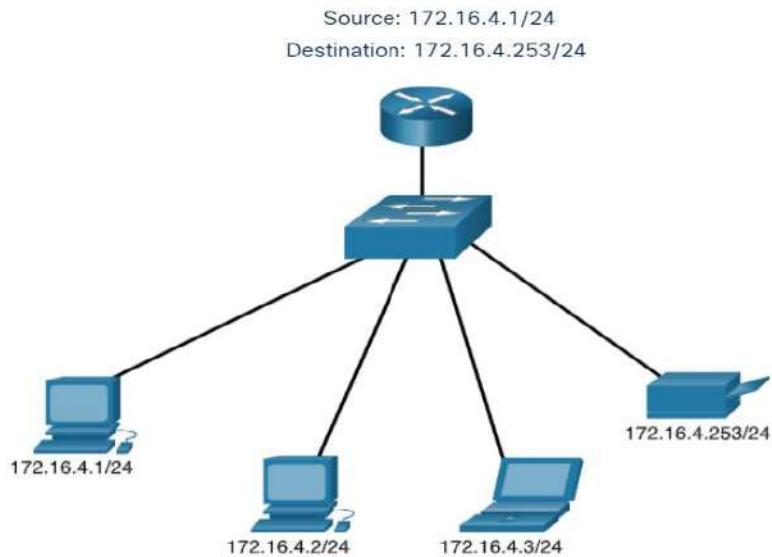
Dalam Materi sebelumnya Anda belajar tentang struktur alamat IPv4; masing-masing memiliki Network Portion dan Host Portion. Ada berbagai cara untuk mengirim paket dari Source Device, dan transmisi yang berbeda ini mempengaruhi Destination IPv4 Address.

Unicast

Transmisi Unicast mengacu pada satu perangkat yang mengirim pesan ke satu perangkat lain dalam komunikasi satu ke satu.

Paket unicast memiliki Destination IP Address yang merupakan alamat unicast yang masuk ke satu penerima. Source IP Address hanya dapat menjadi alamat unicast, karena paket hanya dapat berasal dari satu sumber. Ini terlepas dari apakah Destination IP Address adalah **unicast, broadcast, atau multicast**.

Alamat host IPv4 **unicast** berada di rentang alamat 1.1.1.1 hingga 223.255.255.255. Namun, dalam kisaran ini banyak alamat yang disediakan untuk tujuan khusus. Alamat tujuan khusus ini akan dibahas nanti dalam Materi ini.



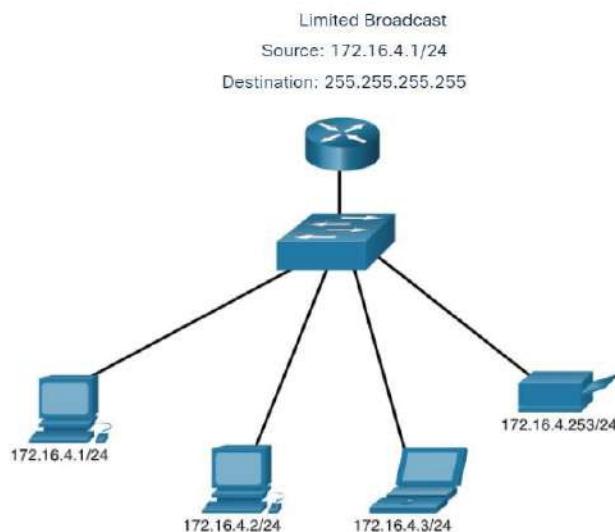
Broadcast

Transmisi broadcast mengacu pada perangkat yang mengirim pesan ke semua perangkat di jaringan dalam komunikasi satu-ke-semua.

Paket broadcast memiliki Destination IP Address dengan semua yang (1) di Host Portion, atau 32 satu (1) bit.

Catatan: IPv4 menggunakan paket **broadcast**. Namun, tidak ada paket **broadcast** dengan IPv6.

Paket broadcast harus diproses oleh semua perangkat di broadcast domain yang sama. Broadcast domain mengidentifikasi semua host di segmen jaringan yang sama. Broadcast dapat diarahkan atau dibatasi. Broadcast terarah dikirim ke semua host pada jaringan tertentu. Misalnya, host di jaringan 172.16.4.0/24 mengirim paket ke 172.16.4.255. Broadcast terbatas dikirim ke 255.255.255.255. Secara default, router tidak meneruskan broadcast.



Paket broadcast menggunakan sumber daya di jaringan dan membuat setiap host penerima di jaringan memproses paket. Oleh karena itu, traffic broadcast harus dibatasi sehingga tidak berdampak buruk pada kinerja jaringan atau perangkat. Karena router memisahkan broadcast domain, jaringan subdividing dapat meningkatkan kinerja jaringan dengan menghilangkan traffic broadcast yang berlebihan.

Broadcast Yang Diarahkan IP

Selain alamat broadcast 255.255.255.255, ada alamat IPv4 broadcast untuk setiap jaringan. Disebut broadcast terarah, alamat ini menggunakan alamat tertinggi dalam jaringan, yang merupakan alamat di mana semua bit host adalah 1. Misalnya, alamat broadcast yang diarahkan untuk 192.168.1.0/24 adalah 192.168.1.255. Alamat ini memungkinkan komunikasi ke semua host dalam jaringan tersebut. Untuk mengirim data ke semua host dalam jaringan, host dapat mengirim satu paket yang ditujukan ke alamat broadcast jaringan.

Perangkat yang tidak terhubung langsung ke jaringan tujuan meneruskan broadcast yang diarahkan IP dengan cara yang sama seperti meneruskan paket IP unicast yang ditakdirkan untuk host di jaringan itu. Ketika paket broadcast terarah mencapai router yang terhubung langsung ke jaringan tujuan, paket itu disiarkan di jaringan tujuan.

Catatan: Karena masalah keamanan dan penyalahgunaan sebelumnya dari pengguna jahat, **broadcast** yang diarahkan dimatikan secara default dimulai dengan Cisco IOS Release 12.0 dengan perintah **no ip directed-broadcasts**.

Multicast

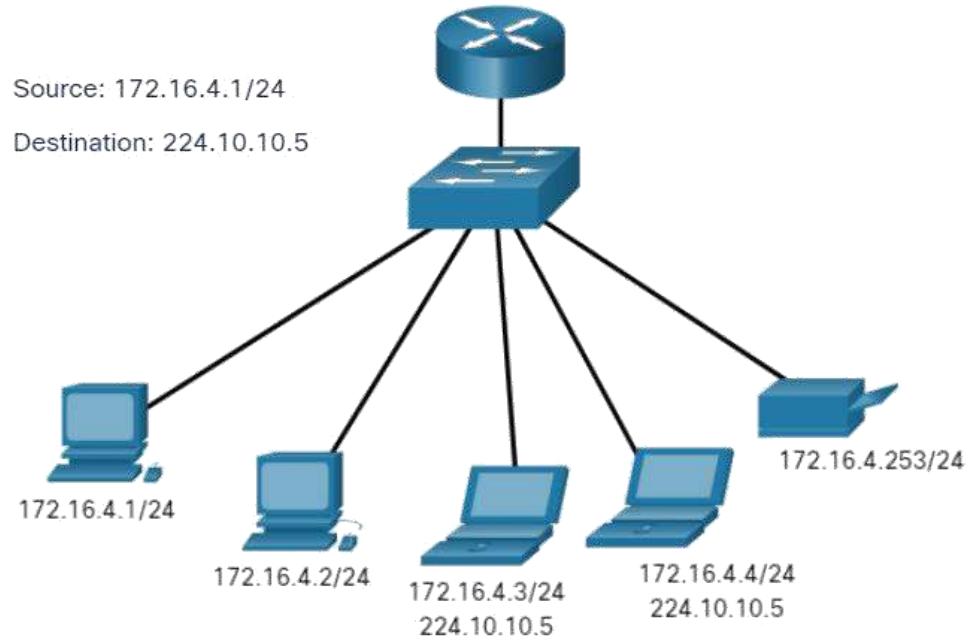
Transmisi multicast mengurangi traffic dengan memungkinkan host mengirim satu paket ke sekumpulan host yang dipilih yang berlangganan ke grup multicast.

Paket multicast adalah paket dengan Destination IP Address yang merupakan alamat multicast. IPv4 telah memesan alamat 224.0.0.0 hingga 239.255.255.255 sebagai rentang multicast.

Host yang menerima paket multicast tertentu disebut klien multicast. Klien multicast menggunakan layanan yang diminta oleh program klien untuk berlangganan ke grup multicast.

Setiap grup multicast diwakili oleh satu alamat tujuan multicast IPv4. Ketika host IPv4 berlangganan ke grup multicast, host memproses paket yang ditujukan ke alamat multicast ini, dan paket yang ditujukan ke alamat unicast yang dialokasikan secara unik.

Protokol perutean seperti OSPF menggunakan transmisi multicast. Misalnya, router yang diaktifkan dengan OSPF berkomunikasi satu sama lain menggunakan alamat multicast OSPF yang dipesan 224.0.0.5. Hanya perangkat yang diaktifkan dengan OSPF yang akan memproses paket ini dengan 224.0.0.5 sebagai Destination IPv4 Address. Semua perangkat lain akan mengabaikan paket ini.



Tipe Tipe IPv4 Address

Sama seperti ada berbagai cara untuk mengirimkan paket IPv4, ada juga berbagai jenis alamat IPv4. Beberapa alamat IPv4 tidak dapat digunakan untuk pergi ke internet, dan yang lain secara khusus dialokasikan untuk **routing** ke internet. Beberapa digunakan untuk memverifikasi koneksi dan yang lain ditetapkan sendiri. Sebagai administrator jaringan, Anda akhirnya akan menjadi sangat akrab dengan jenis alamat IPv4, tetapi untuk saat ini, Anda setidaknya harus tahu apa itu dan kapan menggunakannya.

IPv4 Address Private dan Public

IPv4 Public Address adalah alamat yang di *routing* secara global antara router **Internet Service Provider** (ISP). Namun, tidak semua alamat IPv4 yang tersedia dapat digunakan di internet. Ada blok alamat yang disebut **Private Address** yang digunakan oleh sebagian besar organisasi untuk menetapkan alamat IPv4 ke host internal.

Pada pertengahan 1990-an, dengan diperkenalkannya World Wide Web (WWW), **Private IPv4 Address** diperkenalkan karena menipisnya ruang alamat IPv4. **Private IPv4 Address** tidak unik dan dapat digunakan secara internal dalam jaringan apa pun.

Catatan: Solusi jangka panjang untuk penipisan alamat IPv4 adalah IPv6.

Blok Private Address

| Network Address and Prefix | RFC 1918 Private Address Range |
|----------------------------|--------------------------------|
| 10.0.0.0/8 | 10.0.0.0 - 10.255.255.255 |
| 172.16.0.0/12 | 172.16.0.0 - 172.31.255.255 |
| 192.168.0.0/16 | 192.168.0.0 - 192.168.255.255 |

Catatan: **Private Address** didefinisikan dalam RFC 1918 dan kadang-kadang disebut sebagai ruang alamat RFC 1918.

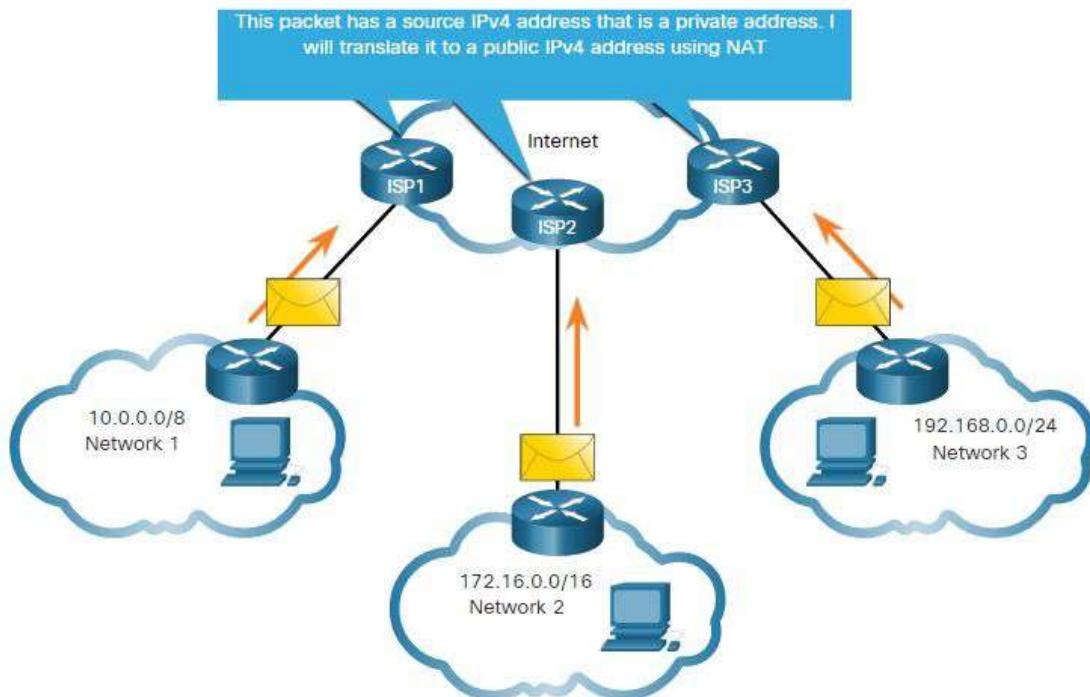
Routing ke Internet

Sebagian besar jaringan internal, dari perusahaan besar hingga jaringan rumah, menggunakan Private IPv4 Address untuk mengatasi semua perangkat internal (intranet) termasuk host dan router. Namun, Private Address tidak dapat di *routing* secara global.

Dalam gambar, jaringan pelanggan 1, 2, dan 3 mengirim paket di luar jaringan internal mereka. Paket-paket ini memiliki Source Ipv4 Address yang merupakan Private Address dan Destination Ipv4 Address yang bersifat publik (dapat diubah secara global). Paket dengan Private Address harus di filter (dibuang) atau diterjemahkan ke alamat publik sebelum meneruskan paket ke ISP.

Diagram adalah topologi jaringan dengan tiga jaringan, masing-masing terhubung ke router ISP yang berbeda. Router ISP melakukan NAT antara setiap jaringan dan Internet.

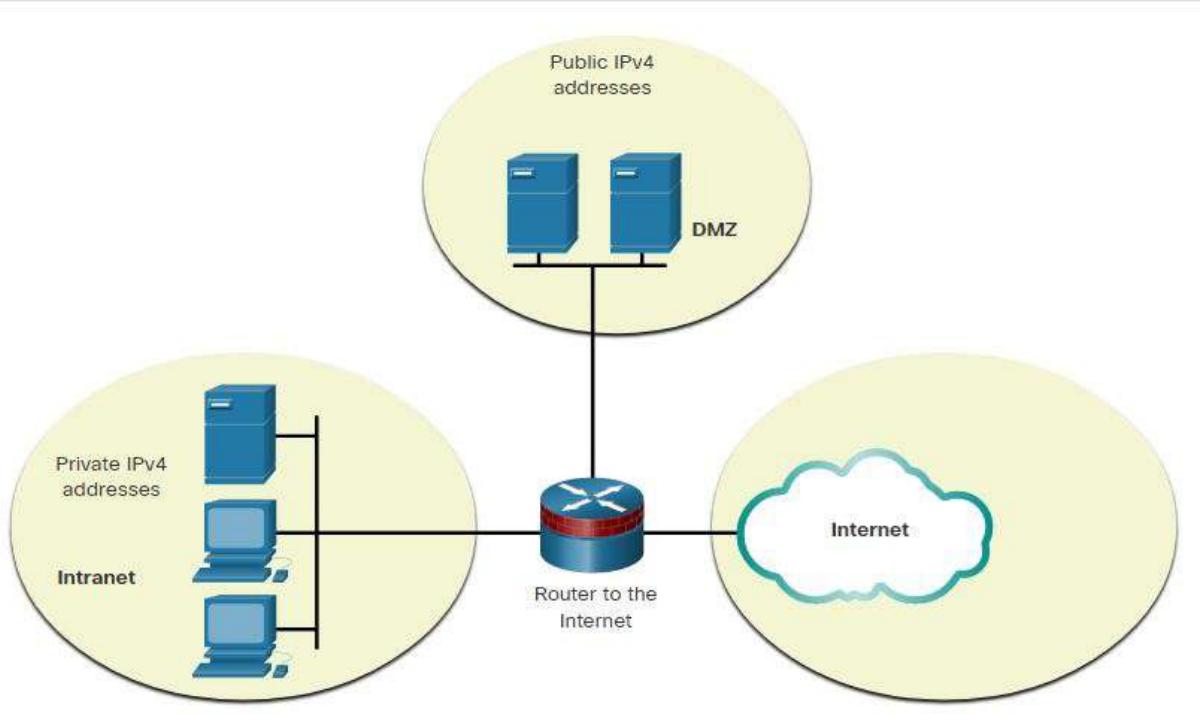
Private IPv4 Address dan Network Address Translation (NAT)



Sebelum ISP dapat meneruskan paket ini, ISP harus menerjemahkan **Source Ipv4 Address**, yang merupakan **Private Address**, ke **IPv4 Public Address** menggunakan Network Address Translation (NAT). NAT digunakan untuk menerjemahkan antara **Private IPv4 Address** dan IPv4 publik. Ini biasanya dilakukan pada router yang menghubungkan jaringan internal ke jaringan ISP. **Private IPv4 Address** di intranet organisasi akan diterjemahkan ke **IPv4 Public Address** sebelum **routing** ke internet.

Catatan: Meskipun, perangkat dengan **Private IPv4 Address** tidak dapat diakses langsung dari perangkat lain di internet, IETF tidak menganggap **Private IPv4 Address** atau NAT sebagai langkah-langkah keamanan yang efektif.

Organisasi yang memiliki **Resource** yang tersedia untuk internet, seperti server web, juga akan memiliki perangkat yang memiliki **IPv4 Public Address**. Seperti yang ditunjukkan pada gambar, bagian jaringan ini dikenal sebagai DMZ (zona demilitarisasi). Router dalam gambar tidak hanya melakukan **routing**, ia juga melakukan NAT dan bertindak sebagai firewall untuk keamanan.



Catatan: **Private IPv4 Address** umumnya digunakan untuk tujuan pendidikan alih-alih menggunakan **IPv4 Public Address** yang kemungkinan besar milik organisasi.

Alamat IPv4 Penggunaan Khusus

Ada alamat tertentu, seperti alamat jaringan dan **broadcast address**, yang tidak dapat ditetapkan ke host. Ada juga alamat khusus yang dapat ditetapkan ke host, tetapi dengan batasan tentang bagaimana host tersebut dapat berinteraksi dalam jaringan.

Loopback address

Loopback address (127.0.0.0 /8 atau 127.0.0.1 hingga 127.255.255.254) lebih sering diidentifikasi sebagai hanya 127.0.0.1, ini adalah alamat khusus yang digunakan oleh **host** untuk mengarahkan lalu lintas ke lalu lintas sendiri. Misalnya, dapat digunakan pada host untuk menguji apakah konfigurasi TCP/IP beroperasi, seperti yang ditunjukkan pada gambar. Perhatikan bagaimana **loopback address** 127.0.0.1 membals perintah ping. Perhatikan juga bagaimana alamat apa pun dalam blok ini akan kembali ke host lokal, yang ditunjukkan dengan **ping kedua** dalam gambar.

Ping Interface Loopback

```
C:\Users\NetAcad> ping 127.0.0.1
Pinging 127.0.0.1 with 32 bytes of data:
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Ping statistics for 127.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
C:\Users\NetAcad> ping 127.1.1.1
Pinging 127.1.1.1 with 32 bytes of data:
Reply from 127.1.1.1: bytes=32 time<1ms TTL=128
Ping statistics for 127.1.1.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
C:\Users\NetAcad>
```

Link-local address

Link-local address (169.254.0.0 /16 atau 169.254.0.1 hingga 169.254.255.254) lebih dikenal sebagai alamat Automatic Private IP Addressing (APIPA) atau alamat yang ditetapkan sendiri. Mereka digunakan oleh klien WINDOWS DHCP untuk mengkonfigurasi sendiri jika tidak ada server DHCP yang tersedia. **Link-local address** dapat digunakan dalam koneksi peer-to-peer tetapi tidak umum digunakan untuk tujuan ini.

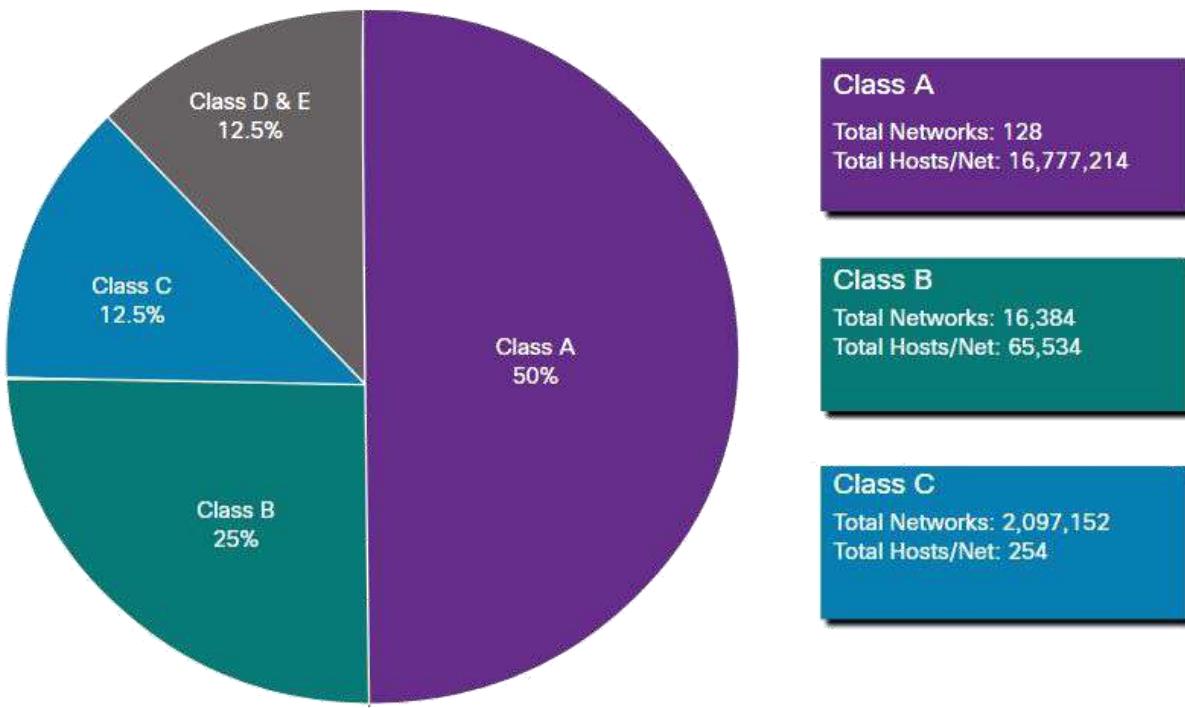
Legacy Classful Addressing

Pada tahun 1981, alamat IPv4 ditetapkan menggunakan classfull address sebagaimana didefinisikan dalam RFC 790 (<https://tools.ietf.org/html/rfc790>), Nomor yang ditetapkan. Pelanggan dialokasikan alamat jaringan berdasarkan salah satu dari tiga kelas, A, B, atau C. RFC membagi rentang unicast ke dalam kelas tertentu sebagai berikut:

- **Kelas A (0.0.0.0/8 hingga 127.0.0.0/8)** – Dirancang untuk mendukung jaringan yang sangat besar dengan lebih dari 16 juta alamat host. Kelas A menggunakan prefix /8 tetap dengan oktet pertama untuk menunjukkan alamat jaringan dan tiga oktet tersisa untuk alamat host (lebih dari 16 juta alamat host per jaringan).
- **Kelas B (128.0.0.0 /16 – 191.255.0.0 /16)** – Dirancang untuk mendukung kebutuhan jaringan ukuran sedang hingga besar dengan hingga sekitar 65.000 alamat host. Kelas B menggunakan prefix tetap / 16 dengan dua oktet urutan tinggi untuk menunjukkan alamat jaringan dan dua oktet yang tersisa untuk alamat host (lebih dari 65.000 alamat host per jaringan).
- **Kelas C (192.0.0.0 /24 – 223.255.255.0 /24)** – Dirancang untuk mendukung jaringan kecil dengan maksimum 254 host. Kelas C menggunakan prefix /24 tetap dengan tiga oktet pertama untuk menunjukkan jaringan dan sisa oktet untuk alamat host (hanya 254 alamat host per jaringan).

Catatan: Ada juga blok multicast Kelas D yang terdiri dari 224.0.0.0 hingga 239.0.0.0 dan blok alamat eksperimental Kelas E yang terdiri dari 240.0.0.0 – 255.0.0.0.

Pada saat itu, dengan jumlah komputer yang terbatas menggunakan internet, mengatasi berkelas adalah cara yang efektif untuk mengalokasikan alamat. Seperti yang ditunjukkan pada gambar, jaringan Kelas A dan B memiliki jumlah alamat host yang sangat besar dan Kelas C memiliki sangat sedikit. Jaringan Kelas A menyumbang 50% dari jaringan IPv4. Hal ini menyebabkan sebagian besar alamat IPv4 yang tersedia tidak digunakan.



Pada pertengahan 1990-an, dengan diperkenalkannya World Wide Web (WWW), **classfull address** ditolak untuk mengalokasikan ruang alamat IPv4 yang terbatas secara lebih efisien. Alokasi **classful address** diganti dengan **classless address**, yang digunakan hari ini. Mengatasi tanpa kelas mengabaikan aturan kelas (A, B, C). Alamat jaringan IPv4 publik (alamat jaringan dan masker subjaringan) dialokasikan berdasarkan jumlah alamat yang dapat dibenarkan.

Assignment of IP Addresses

IPv4 Public Address adalah alamat yang dirutekan secara global melalui internet. **IPv4 Public Address** harus unik.

Alamat IPv4 dan IPv6 dikelola oleh Internet Assigned Numbers Authority (IANA). IANA mengelola dan mengalokasikan blok alamat IP ke Regional Internet Registries (RIR). Kelima RIR tersebut ditunjukkan dalam angka tersebut.

RIR bertanggung jawab untuk mengalokasikan alamat IP ke ISP yang menyediakan blok alamat IPv4 kepada organisasi dan ISP yang lebih kecil. Organisasi juga bisa mendapatkan alamat mereka langsung dari RIR (tunduk pada kebijakan RIR itu).

Registri Internet Regional



- **AfriNIC** (African Network Information Centre) – Africa Region
- **APNIC** (Asia Pacific Network Information Centre) – Asia/Pacific Region
- **ARIN** (American Registry for Internet Numbers) – North America Region
- **LACNIC** (Regional Latin-American and Caribbean IP Address Registry) – Latin America and some Caribbean Islands
- **RIPE NCC** (Réseaux IP Européens Network Coordination Centre) – Europe, the Middle East, and Central Asia

Segmentasi Jaringan

Pernahkah Anda menerima email yang ditujukan kepada setiap orang di kantor atau sekolah Anda? Ini adalah email **Broadcast**. Mudah-mudahan, itu berisi informasi yang perlu diketahui masing-masing dari Anda. Tetapi seringkali **Broadcast** tidak benar-benar berkaitan dengan semua orang dalam **Mail List**. Terkadang, hanya beberapa segmen yang perlu membaca informasi itu.

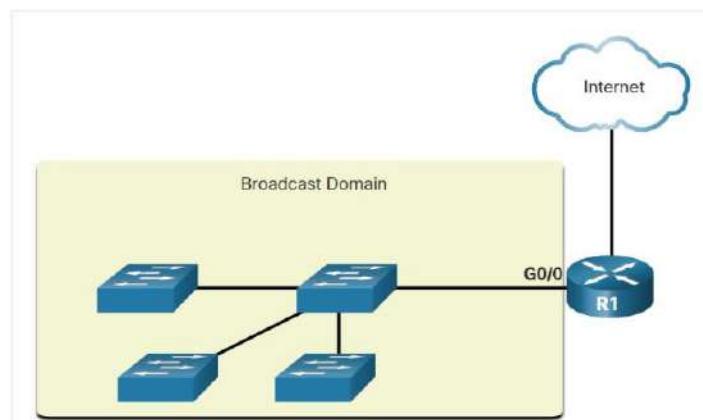
Broadcast Domain Dan Segmentasi

Di LAN Ethernet, perangkat menggunakan **Broadcast** dan **Address Resolution Protocol** (ARP) untuk menemukan perangkat lain.. ARP mengirim **Broadcast** Layer 2 ke alamat IPv4 yang diketahui di jaringan lokal untuk menemukan alamat MAC terkait. Perangkat di ETHERNET JUGA menemukan perangkat lain yang menggunakan layanan. Host biasanya memperoleh konfigurasi alamat IPv4-nya menggunakan Dynamic Host Configuration Protocol (DHCP) yang mengirim **Broadcast** di jaringan lokal untuk menemukan server DHCP.

Switch menyebarkan **Broadcast** ke semua **Interface** kecuali **Interface** tempat **Broadcast** diterima. Misalnya, jika **Switch** dalam gambar menerima **Broadcast**, itu akan meneruskannya ke **Switch** lain dan pengguna lain yang terhubung dalam jaringan.

Router, R1, terhubung ke **Switch** melalui **Interface G0/0**. **Switch** memiliki koneksi ke tiga **Switch** lainnya. **Broadcast Domain** terdiri dari empat **Switch** dan **Interface** router yang terhubung dengannya. Sambungan dari **Router** ke Internet tidak berada dalam **Broadcast Domain**.

Router Segment Broadcast Domain



Router tidak menyebarluaskan **Broadcast**. Ketika router menerima **Broadcast**, router tidak meneruskannya keluar **Interface** lain. Misalnya, ketika R1 menerima **Broadcast** di **Interface** Gigabit Ethernet 0/0, R1 tidak meneruskan **Interface** lain.

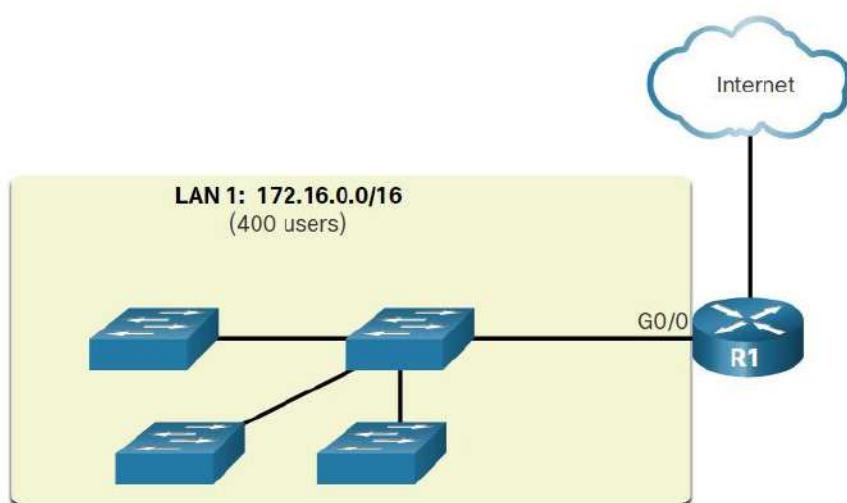
Oleh karena itu, setiap **Interface** router terhubung ke **Broadcast Domain** dan **Broadcast** hanya disebarluaskan dalam **Broadcast Domain** tertentu.

Masalah dengan Besarnya Broadcast Domain

Large Broadcast Domain adalah jaringan yang menghubungkan banyak host. Masalah dengan **Large Broadcast Domain** adalah bahwa host ini dapat menghasilkan **Broadcast** yang berlebihan dan berdampak negatif pada jaringan. Dalam angka tersebut, LAN 1 menghubungkan 400 pengguna yang dapat menghasilkan kelebihan jumlah lalu lintas **Broadcast**. Hal ini mengakibatkan operasi jaringan yang lambat karena jumlah lalu lintas yang signifikan yang dapat disebabkannya, dan operasi perangkat yang lambat karena perangkat harus menerima dan memproses setiap paket **Broadcast**.

Router, R1, terhubung ke **Switch** melalui **Interface** G0/0. **Switch** memiliki koneksi ke tiga **Switch** lainnya. **Broadcast Domain** terdiri dari empat **Switch** dan **Interface** router yang terhubung dengannya. Ini diidentifikasi sebagai LAN1 dengan alamat 172.16.0.0/16. Sambungan dari **Router** ke Internet tidak berada dalam **Broadcast Domain**.

Large Broadcast Domain

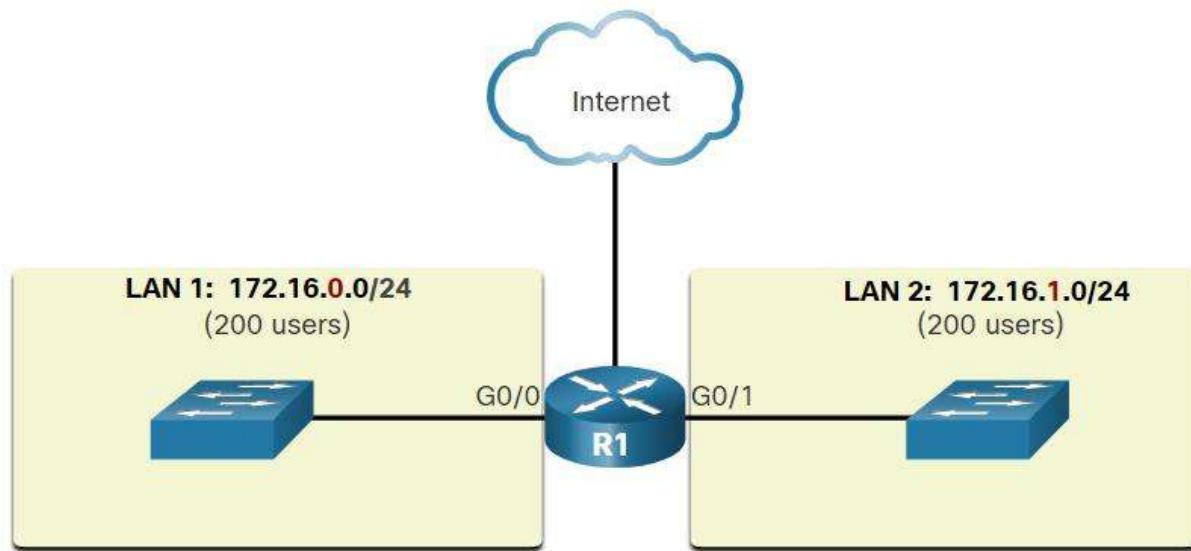


Solusinya adalah mengurangi ukuran jaringan untuk membuat **Broadcast Domain** yang lebih kecil dalam proses yang disebut subnetting. Ruang jaringan yang lebih kecil ini disebut **Subnetwork**.

Dalam gambar tersebut, ke-400 pengguna di LAN 1 dengan alamat jaringan 172.16.0.0 /16 telah dibagi menjadi dua subnet masing-masing 200 pengguna: 172.16.0.0 /24 dan 172.16.1.0 /24. **Broadcast** hanya disebarluaskan dalam **Broadcast Domain** yang lebih kecil. Oleh karena itu, **Broadcast** di LAN 1 tidak akan merambat ke LAN 2.

Router, R1, terhubung ke dua LAN yang mewakili dua **Broadcast Domain** yang berbeda. Terhubung di sebelah kiri melalui G0/0 adalah **Switch** yang mendukung 200 pengguna di LAN 1 dengan alamat jaringan 172.16.0.0/24. Terhubung di sebelah kanan melalui G0/1 adalah switch yang mendukung 200 pengguna di LAN 2 dengan alamat jaringan 172.16.1.0/24.

Berkomunikasi Antar Jaringan



Perhatikan bagaimana Prefix length telah berubah dari jaringan /16 tunggal menjadi dua jaringan /24. Ini adalah dasar dari subnetting: menggunakan bit host untuk membuat subnet tambahan.

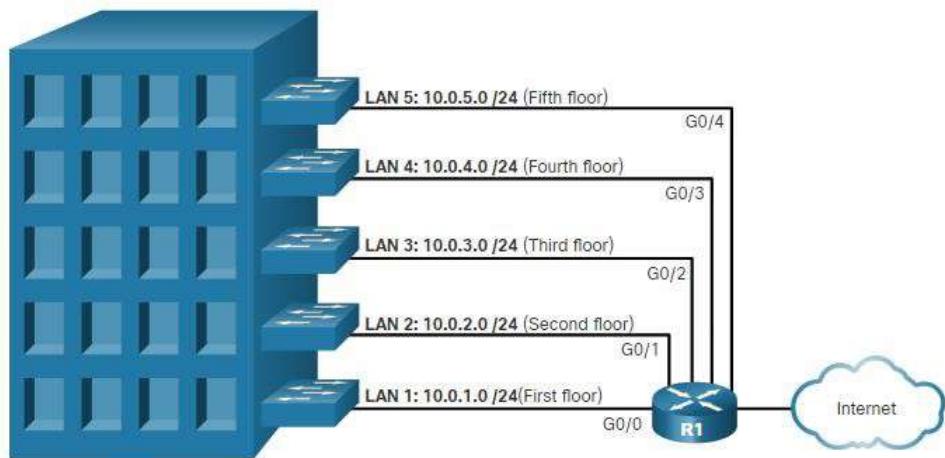
Catatan: Istilah subnet dan jaringan sering digunakan secara bergantian. Sebagian besar jaringan adalah **Subnetwork** dari beberapa blok alamat yang lebih besar.

Alasan Untuk Segmentasi Jaringan

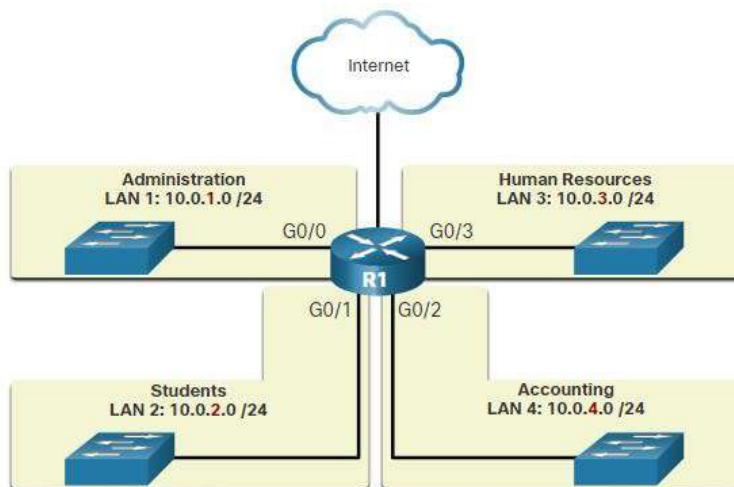
Subnetting mengurangi lalu lintas jaringan secara keseluruhan dan meningkatkan kinerja jaringan. Ini juga memungkinkan administrator untuk menerapkan kebijakan keamanan seperti **Subnetwork** mana yang diizinkan atau tidak diizinkan untuk berkomunikasi bersama. Alasan lain adalah mengurangi jumlah perangkat yang terpengaruh oleh lalu lintas **Broadcast** abnormal karena kesalahan konfigurasi, masalah perangkat keras / perangkat lunak, atau niat jahat.

Ada berbagai cara menggunakan subnet untuk membantu mengelola perangkat jaringan.

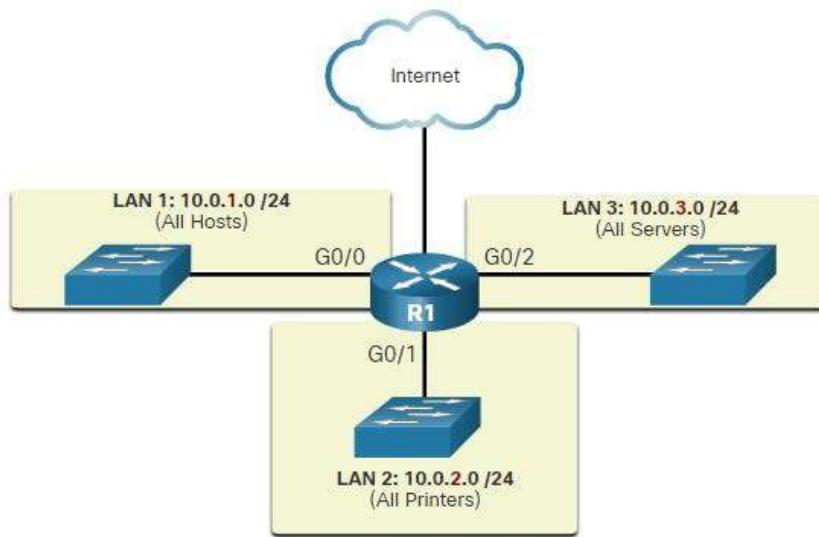
Location



Group Or Function



Device Type



Administrator jaringan dapat membuat **Subnetwork** menggunakan divisi lain yang masuk akal untuk jaringan. Perhatikan pada setiap gambar, subnet menggunakan **Prefix** yang lebih panjang untuk mengidentifikasi jaringan.

Memahami cara mensubnet jaringan adalah keterampilan mendasar yang harus dikembangkan oleh semua administrator jaringan. Berbagai metode telah dibuat untuk membantu memahami proses ini. Meskipun sedikit berlebihan pada awalnya, perhatikan detailnya dan, dengan latihan, subnetting akan menjadi lebih mudah.

Subnetwork IPv4

Dalam materi sebelumnya Anda mempelajari beberapa alasan bagus untuk Segmentasi jaringan. Anda juga belajar bahwa segmentasi jaringan disebut subnetting. Subnetting adalah keterampilan penting yang harus dilakukan saat mengelola jaringan IPv4. Ini agak menakutkan pada awalnya, tetapi menjadi jauh lebih mudah dengan latihan.

Subnet di Batas Oktet

Subnet IPv4 dibuat dengan menggunakan satu atau beberapa bit host sebagai bit jaringan. Ini dilakukan dengan memperluas Subnetmask untuk meminjam beberapa bit dari bagian host alamat untuk membuat bit jaringan tambahan. Semakin banyak bit host yang dipinjam, semakin banyak subnet yang dapat didefinisikan. Semakin banyak bit yang dipinjam untuk meningkatkan jumlah subnet mengurangi jumlah host per subnet.

Jaringan paling mudah di subnetisasi di batas oktet /8, /16, dan /24. Tabel mengidentifikasi **Prefix** ini. Perhatikan bahwa menggunakan **Prefix** yang lebih panjang mengurangi jumlah host per subnet.

Subnetmask pada Batas Oktet

| Prefiks | Subnet Mask | Subnet Mask dalam Biner (n = jaringan, h = host) | # Host |
|---------|---------------|--|------------|
| /8 | 255.0.0.0 | nnnnnnnn.hhhhhhhh.hhhhhhhh.hhhhhhhh 11111111.00000000.00000000.00000000 | 16,777,214 |
| /16 | 255.255.0.0 | nnnnnnnn.nnnnnnnn.hhhhhhhh.hhhhhhhh 11111111.11111111.00000000.00000000 | 65,534 |
| /24 | 255.255.255.0 | nnnnnnnn.nnnnnnnn.nnnnnnnn.hhhhhhhh 11111111.11111111.11111111.00000000 | 254 |

Untuk memahami bagaimana subnetting pada batas oktet dapat berguna, pertimbangkan contoh berikut. Asumsikan perusahaan telah memilih alamat pribadi 10.0.0.0/8 sebagai alamat jaringan internalnya. Alamat jaringan tersebut dapat menghubungkan 16.777.214 host dalam satu **Broadcast Domain**. Jelas, memiliki lebih dari 16 juta host pada satu subnet tidak ideal.

Perusahaan dapat lebih mensubnet alamat 10.0.0.0/8 di batas oktet /16 seperti yang ditunjukkan dalam tabel. Ini akan memberi perusahaan kemampuan untuk mendefinisikan hingga 256 subnet (yaitu, 10.0.0.0/16 – 10.255.0.0/16) dengan setiap subnet yang mampu menghubungkan 65.534 host. Perhatikan bagaimana dua oktet pertama mengidentifikasi bagian jaringan dari alamat sedangkan dua oktet terakhir adalah untuk alamat IP host.

Subnetting Jaringan 10.0.0.0/8 menggunakan /16

| Subnet Address (256 Kemungkinan Subnet) | Rentang Host (65.534 host yang mungkin per subnet) | Broadcast |
|--|---|----------------|
| 10.0.0.0/16 | 10.0.0.1 – 10.0.255.254 | 10.0.255.255 |
| 10.1.0.0/16 | 10.1.0.1 – 10.1.255.254 | 10.1.255.255 |
| 10.2.0.0/16 | 10.2.0.1 – 10.2.255.254 | 10.2.255.255 |
| 10.3.0.0/16 | 10.3.0.1 – 10.3.255.254 | 10.3.255.255 |
| 10.4.0.0/16 | 10.4.0.1 – 10.4.255.254 | 10.4.255.255 |
| 10.5.0.0/16 | 10.5.0.1 – 10.5.255.254 | 10.5.255.255 |
| 10.6.0.0/16 | 10.6.0.1 – 10.6.255.254 | 10.6.255.255 |
| 10.7.0.0/16 | 10.7.0.1 – 10.7.255.254 | 10.7.255.255 |
| ... | ... | ... |
| 10.255.0.0/16 | 10.255.0.1 – 10.255.255.254 | 10.255.255.255 |

Atau, perusahaan dapat memilih untuk mensubnet jaringan 10.0.0.0/8 di batas oktet /24, seperti yang ditunjukkan dalam tabel. Ini akan memungkinkan perusahaan untuk mendefinisikan 65.536 subnet masing-masing mampu menghubungkan 254 host. Batas / 24 sangat populer dalam subnetting karena mengakomodasi sejumlah host yang wajar dan subnet yang nyaman di batas oktet.

Subnetting Jaringan 10.0.0.0/8 menggunakan Prefix /24

| Alamat Subjaringan (65.536 Kemungkinan Subjaringan) | Host Range (254 host yang mungkin per subnet) | Broadcast |
|--|--|----------------|
| 10.0.0.0/24 | 10.0.0.1 – 10.0.0.254 | 10.0.0.255 |
| 10.0.1.0/24 | 10.0.1.1 – 10.0.1.254 | 10.0.1.255 |
| 10.0.2.0/24 | 10.0.2.1 – 10.0.2.254 | 10.0.2.255 |
| ... | ... | ... |
| 10.0.255.0/24 | 10.0.255.1 – 10.0.255.254 | 10.0.255.255 |
| 10.1.0.0/24 | 10.1.0.1 – 10.1.0.254 | 10.1.0.255 |
| 10.1.1.0/24 | 10.1.1.1 – 10.1.1.254 | 10.1.1.255 |
| 10.1.2.0/24 | 10.1.2.1 – 10.1.2.254 | 10.1.2.255 |
| ... | ... | ... |
| 10.100.0.0/24 | 10.100.0.1 – 10.100.0.254 | 10.100.0.255 |
| ... | ... | ... |
| 10.255.255.0/24 | 10.255.255.1 – 10.255.255.254 | 10.255.255.255 |

Subnet dalam Batas Oktet

Contoh yang ditunjukkan sejauh ini meminjam bit host dari **Prefix** jaringan /8, /16, dan /24 yang umum. Namun, subnet dapat meminjam bit dari posisi bit host apa pun untuk membuat **Mask** lain.

Misalnya, alamat jaringan /24 biasanya disubnetisasi menggunakan **Prefix** yang lebih panjang dengan meminjam bit dari oktet keempat. Ini memberi administrator fleksibilitas tambahan saat menetapkan alamat jaringan ke sejumlah kecil **End Devices**.

Lihat tabel untuk melihat enam cara untuk mensubnet jaringan /24.

Subnet jaringan /24

| Prefiks | Subnet Mask | Subnet Mask Dalam Biner (n = jaringan, h = host) | # subnet | # Host |
|---------|-----------------|--|----------|--------|
| /25 | 255.255.255.128 | nnnnnnnn.nnnnnnnn.nnnnnnnn.nhhhhhhh 11111111.11111111.11111111.10000000 | 2 | 126 |
| /26 | 255.255.255.192 | nnnnnnnn.nnnnnnnn.nnnnnnnn.nnhhhhhh 11111111.11111111.11111111.11000000 | 4 | 62 |
| /27 | 255.255.255.224 | nnnnnnnn.nnnnnnnn.nnnnnnnn.nnnhhhhh 11111111.11111111.11111111.11100000 | 8 | 30 |
| /28 | 255.255.255.240 | nnnnnnnn.nnnnnnnn.nnnnnnnn.nnnnhhhh 11111111.11111111.11111111.11110000 | 16 | 14 |
| /29 | 255.255.255.248 | nnnnnnnn.nnnnnnnn.nnnnnnnn.nnnnnhhh 11111111.11111111.11111111.11111000 | 32 | 6 |
| /30 | 255.255.255.252 | nnnnnnnn.nnnnnnnn.nnnnnnnn.nnnnnnh 11111111.11111111.11111111.11111100 | 64 | 2 |

Untuk setiap bit yang dipinjam di oktet keempat, jumlah sub jaringan yang tersedia dua kali lipat, sekaligus mengurangi jumlah alamat host per subnet:

- **/25 baris** – Meminjam 1 bit dari oktet keempat menciptakan 2 subnet yang mendukung masing-masing 126 host.
- **/26 baris** – Meminjam 2 bit menciptakan 4 subnet yang mendukung masing-masing 62 host.
- **/27 baris** – Meminjam 3 bit menciptakan 8 subnet yang mendukung masing-masing 30 host.
- **/28 baris** – Meminjam 4 bit menciptakan 16 subnet yang mendukung masing-masing 14 host.
- **/29 baris** – Meminjam 5 bit menciptakan 32 subnet yang mendukung masing-masing 6 host.
- **/30 baris** – Meminjam 6 bit menciptakan 64 subnet yang mendukung masing-masing 2 host.

Subnet Prefix /16 dan /8

Beberapa subnetting lebih mudah daripada subnetting lainnya. Materi ini menjelaskan cara membuat subnet yang masing-masing memiliki jumlah host yang sama.

Membuat Subnet dengan Prefix /16

Dalam situasi yang membutuhkan sejumlah besar subnet, jaringan IPv4 diperlukan yang memiliki lebih banyak bit host yang tersedia untuk dipinjam. Misalnya, alamat jaringan 172.16.0.0 memiliki default **mask** 255.255.0.0, atau /16. Alamat ini memiliki 16 bit dalam **Network Portion** dan 16 bit dalam **Host Portion**. 16 bit dalam **Host Portion** tersedia untuk dipinjam untuk membuat subnet. Tabel menyoroti semua skenario yang mungkin untuk mensubnetisasi **Prefix /16**.

Subnetmask Prefix /16

| Prefiks | Subnet Mask | Network Address (n = network, h = host) | # subnet | # host |
|---------|---------------|--|----------|--------|
| /17 | 255.255.128.0 | nnnnnnnn.nnnnnnnn.nhhhhhhh.hhhhhhhh 11111111.11111111.10000000.00000000 | 2 | 32766 |
| /18 | 255.255.192.0 | nnnnnnnn.nnnnnnnn.nnhhhhhh.hhhhhhhh 11111111.11111111.11000000.00000000 | 4 | 16382 |
| /19 | 255.255.224.0 | nnnnnnnn.nnnnnnnn.nnnhhhhh.hhhhhhhh 11111111.11111111.11100000.00000000 | 8 | 8190 |
| /20 | 255.255.240.0 | nnnnnnnn.nnnnnnnn.nnnnhhhh.hhhhhhhh 11111111.11111111.11110000.00000000 | 16 | 4094 |
| /21 | 255.255.248.0 | nnnnnnnn.nnnnnnnn.nnnnnhhh.hhhhhhhh 11111111.11111111.11111000.00000000 | 32 | 2046 |
| /22 | 255.255.252.0 | nnnnnnnn.nnnnnnnn.nnnnnngh.hhhhhhhh 11111111.11111111.11111100.00000000 | 64 | 1022 |
| /23 | 255.255.254.0 | nnnnnnnn.nnnnnnnn.nnnnnnnh.hhhhhhhh 11111111.11111111.11111110.00000000 | 128 | 510 |
| /24 | 255.255.255.0 | nnnnnnnn.nnnnnnnn.nnnnnnnn.hhhhhhhh 11111111.11111111.11111111.00000000 | 256 | 254 |

| | | | | |
|-----|-----------------|--|-------|-----|
| /25 | 255.255.255.128 | nnnnnnnn.nnnnnnnn.nnnnnnnn.nhhhhhhh 11111111.11111111.11111111.10000000 | 512 | 126 |
| /26 | 255.255.255.192 | nnnnnnnn.nnnnnnnn.nnnnnnnn.nhhhhhhh 11111111.11111111.11111111.11000000 | 1024 | 62 |
| /27 | 255.255.255.224 | nnnnnnnn.nnnnnnnn.nnnnnnnn.nnnhhhhh 11111111.11111111.11111111.11100000 | 2048 | 30 |
| /28 | 255.255.255.240 | nnnnnnnn.nnnnnnnn.nnnnnnnn.nnnnhhhh 11111111.11111111.11111111.11110000 | 4096 | 14 |
| /29 | 255.255.255.248 | nnnnnnnn.nnnnnnnn.nnnnnnnn.nnnnnhhh 11111111.11111111.11111111.11111000 | 8192 | 6 |
| /30 | 255.255.255.252 | nnnnnnnn.nnnnnnnn.nnnnnnnn.nnnnnngh 11111111.11111111.11111111.11111100 | 16384 | 2 |

Meskipun Anda tidak perlu menghafal tabel ini, Anda masih perlu pemahaman yang baik tentang bagaimana setiap nilai dalam tabel dihasilkan. Jangan biarkan ukuran meja mengintimidasi Anda. Alasannya besar adalah bahwa ia memiliki 8 bit tambahan yang dapat dipinjam, dan, oleh karena itu, jumlah subnet dan host hanya lebih besar.

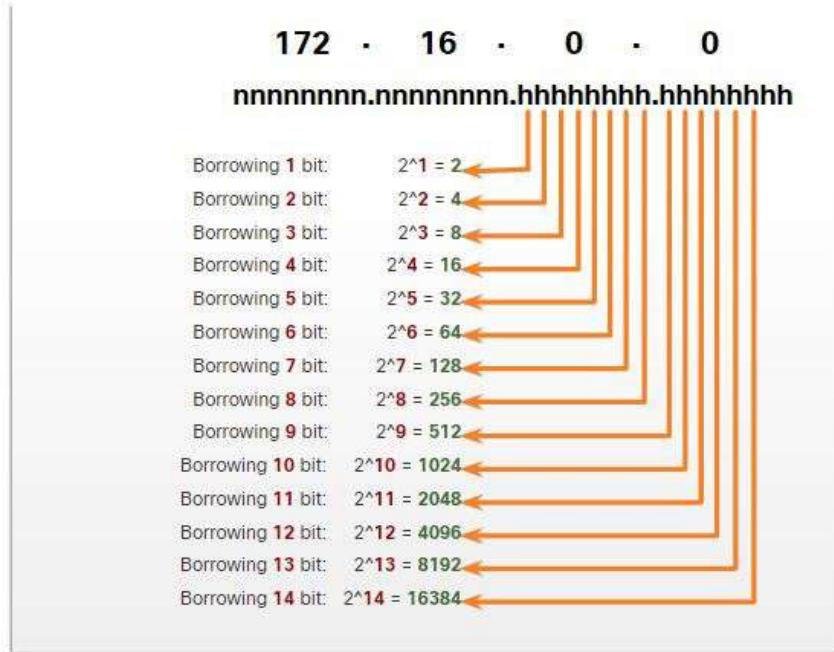
Membuat 100 Subnet dengan Prefix Slash 16

Pertimbangkan perusahaan besar yang memerlukan setidaknya 100 subnet dan telah memilih **private address** 172.16.0.0/16 sebagai alamat jaringan internalnya.

Ketika meminjam bit dari alamat / 16, mulai meminjam bit di oktet ketiga, pergi dari kiri ke kanan. Pinjam sedikit pada satu waktu sampai jumlah bit yang diperlukan untuk membuat 100 subnet tercapai.

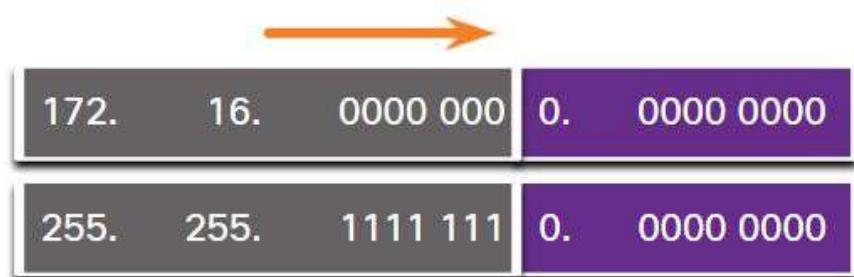
Gambar menampilkan jumlah subnet yang dapat dibuat ketika meminjam bit dari oktet ketiga dan oktet keempat. Perhatikan sekarang ada hingga 14 bit host yang dapat dipinjam.

Jumlah Subnet yang terbentuk



Untuk memenuhi persyaratan 100 subnet untuk perusahaan, 7 bit (yaitu, $2^7 = 128$ subnet) perlu dipinjam (dengan total 128 subnet), seperti yang ditunjukkan pada gambar.

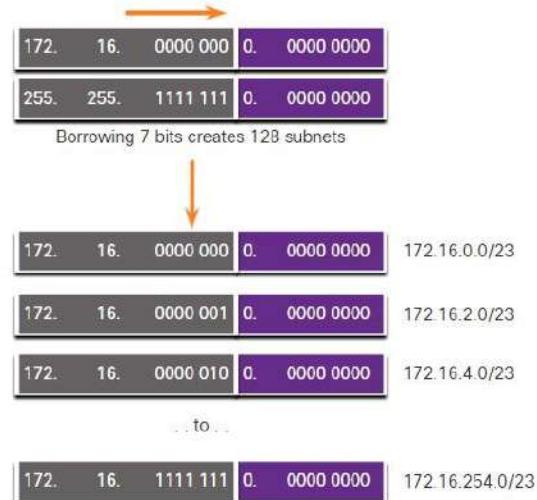
172.16.0.0/23 Network



Ingat bahwa **subnet mask** harus berubah untuk mencerminkan bit yang dipinjam. Dalam contoh ini, ketika 7 bit dipinjam, **mask** diperpanjang 7 bit ke oktet ketiga. Dalam desimal, **mask** diwakili sebagai 255.255.254.0, atau **Prefix /23**, karena oktet ketiga adalah 11111110 dalam biner dan oktet keempat adalah 00000000 dalam biner.

Gambar menampilkan subnet yang dihasilkan dari 172.16.0.0 /23 hingga 172.16.254.0 /23.

Menjadi /23 Subnets



Setelah meminjam 7 bit untuk subnet, ada satu bit host yang tersisa di oktet ketiga, dan 8 bit host tersisa di oktet keempat, dengan total 9 bit yang tidak dipinjam. 29 menghasilkan 512 total alamat host. Alamat pertama disediakan untuk alamat jaringan dan alamat terakhir disediakan untuk alamat siaran, jadi kurangi untuk kedua alamat ini ($2^9 - 2$) sama dengan 510 alamat host yang tersedia untuk setiap / 23 subnet.

Seperti yang ditunjukkan pada gambar, alamat host pertama untuk subnet pertama adalah 172.16.0.1, dan alamat host terakhir adalah 172.16.1.254.

Address Range for 172.16.0.0/23 Subnet

| | | |
|--------------------|------------------------------------|-------------------|
| Network Address | 172. 16. 00 00 00 0 0. 0000 0000 | = 172.16.0.0/23 |
| First Host Address | 172. 16. 00 00 00 0 0. 0000 0001 | = 172.16.0.1/23 |
| Last Host Address | 172. 16. 00 00 00 0 1. 1111 1110 | = 172.16.1.254/23 |
| Broadcast Address | 172. 16. 00 00 00 0 1. 1111 1111 | = 172.16.1.255/23 |

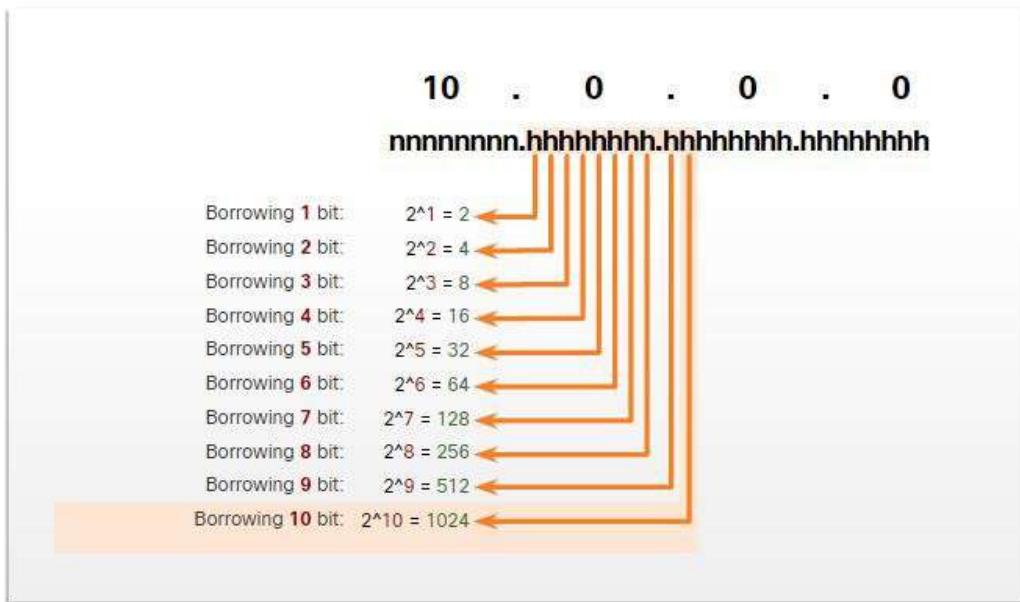
Membuat 1000 Subnet dengan prefix /8

Beberapa organisasi, seperti **small service provider** atau perusahaan besar, mungkin memerlukan lebih banyak subnet. Misalnya, ambil ISP kecil yang memerlukan 1000 subnet untuk kliennya. Setiap klien akan membutuhkan banyak ruang di bagian host untuk membuat subnet sendiri.

ISP memiliki alamat jaringan 10.0.0.0 255.0.0.0 atau 10.0.0.0/8. Ini berarti ada 8 bit dalam **network portion** dan 24 bit host tersedia untuk dipinjam ke arah subnetting. Oleh karena itu, ISP kecil akan subnet jaringan 10.0.0.0/8.

Untuk membuat subnet, Anda harus meminjam bit dari bagian host alamat IPv4 dari internetwork yang ada. Mulai dari kiri ke kanan dengan bit host pertama yang tersedia, pinjam sedikit pada satu waktu sampai Anda mencapai jumlah bit yang diperlukan untuk membuat 1000 subnet. Seperti yang ditunjukkan pada gambar, Anda perlu meminjam 10 bit untuk membuat subnet 1024 ($2^{10} = 1024$). Ini termasuk 8 bit di oktet kedua dan 2 bit tambahan dari oktet ketiga.

Jumlah subnet yang Dibuat



Angka ini menampilkan alamat jaringan dan subnet mask yang dihasilkan, yang dikonversi ke 255.255.192.0 atau 10.0.0.0/18.

Network 10.0.0.0/18

| | | | | |
|------|------|----------|----------|-----------|
| 10. | 0000 | 0000. 00 | 00 0000. | 0000 0000 |
| 255. | 1111 | 1111. 11 | 00 0000. | 0000 0000 |

Angka ini menampilkan subnet yang dihasilkan dari peminjaman 10 bit, membuat subnet dari 10.0.0.0/18 hingga 10.255.128.0/18.

Menghasilkan /18 Subnet

Borrowing 10 bits creates 1024 subnets

| | | | | | |
|------------|------|----------|----------|-----------|-----------------|
| 10. | 0000 | 0000. 00 | 00 0000. | 0000 0000 | 10.0.0.0/18 |
| 255. | 1111 | 1111. 11 | 00 0000. | 0000 0000 | |
| ... to ... | | | | | |
| 10. | 0000 | 0000. 01 | 00 0000. | 0000 0000 | 10.0.64.0/18 |
| 10. | 0000 | 0000. 10 | 00 0000. | 0000 0000 | 10.0.128.0/18 |
| 10. | 0000 | 0000. 11 | 00 0000. | 0000 0000 | 10.0.192.0/18 |
| 10. | 0000 | 0001. 00 | 00 0000. | 0000 0000 | 10.1.0.0/18 |
| 10. | 1111 | 1111. 11 | 00 0000. | 0000 0000 | 10.255.192.0/18 |

Meminjam 10 bit untuk membuat subnet, meninggalkan 14 bit host untuk setiap subnet. Mengurangi dua host per subnet (satu untuk alamat jaringan dan satu untuk alamat Broadcast) sama dengan $2^{14} - 2 = 16382$ per subnet. Ini berarti bahwa masing-masing dari 1000 subnet dapat mendukung hingga 16.382 host.

Gambar ini menampilkan spesifikasi subnet pertama.

Rentang Alamat untuk Subnet 10.0.0.0/18

Network Address

| | | | | | |
|-----|--------------|----|----------|-----------|---------------|
| 10. | 00 00 00 00. | 00 | 00 0000. | 0000 0000 | = 10.0.0.0/18 |
|-----|--------------|----|----------|-----------|---------------|

First Host Address

| | | | | | |
|-----|--------------|----|----------|-----------|---------------|
| 10. | 00 00 00 00. | 00 | 00 0000. | 0000 0001 | = 10.0.0.1/18 |
|-----|--------------|----|----------|-----------|---------------|

Last Host Address

| | | | | | |
|-----|--------------|----|----------|-----------|------------------|
| 10. | 00 00 00 00. | 00 | 11 1111. | 1111 1110 | = 10.0.63.254/18 |
|-----|--------------|----|----------|-----------|------------------|

Broadcast Address

| | | | | | |
|-----|--------------|----|----------|-----------|------------------|
| 10. | 00 00 00 00. | 00 | 11 1111. | 1111 1111 | = 10.0.63.255/18 |
|-----|--------------|----|----------|-----------|------------------|

Persyaratan untuk memenuhi subnet

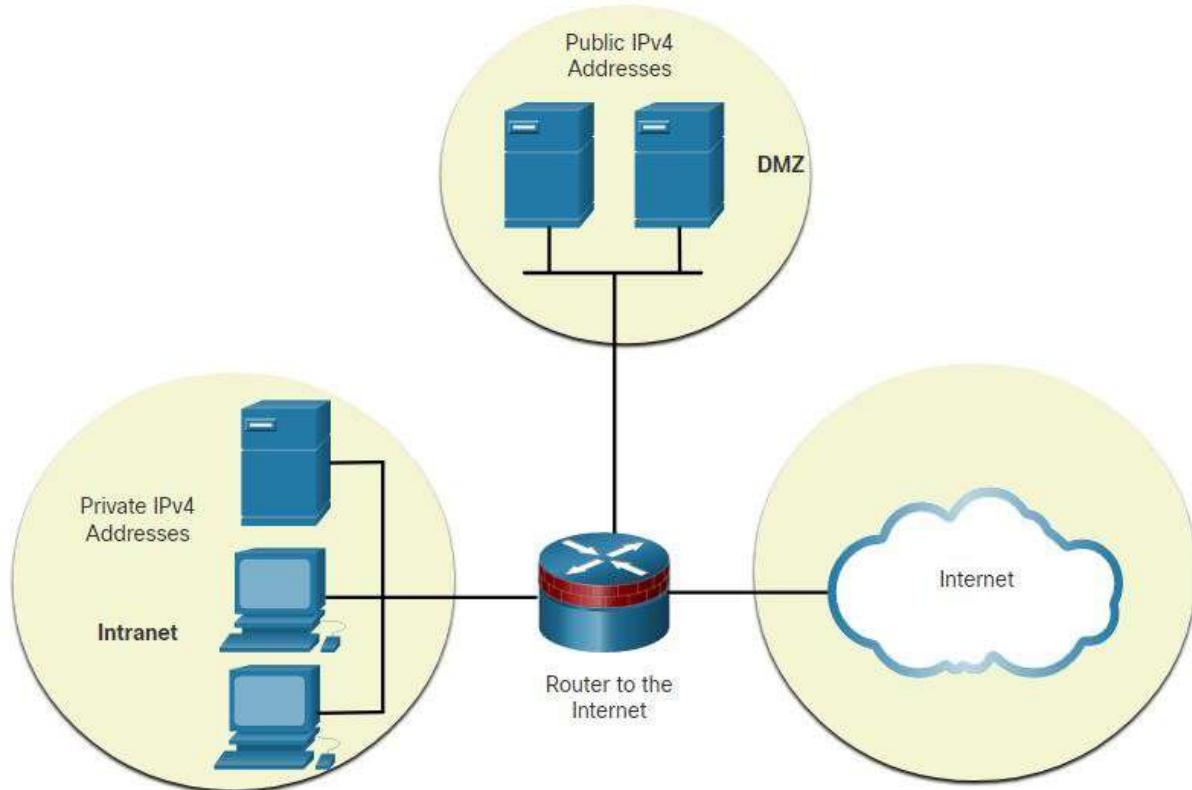
Jaringan Anda dapat menggunakan alamat IPv4 **public** dan **private**. Ini mempengaruhi bagaimana Anda akan mensubnet jaringan Anda.

Subnet Private versus Ruang Alamat IPv4 Publik

Gambar menunjukkan jaringan perusahaan yang khas:

- **Intranet** - Ini adalah bagian internal dari jaringan perusahaan, hanya dapat diakses dalam organisasi. Perangkat di intranet menggunakan alamat IPv4 **privat**.
- **DMZ** - Ini adalah bagian dari jaringan perusahaan yang berisi sumber daya yang tersedia untuk internet seperti server web. Perangkat di DMZ menggunakan alamat IPv4 **publik**.

Ruang Alamat IPv4 Publik dan Privat



Baik internet maupun DMZ memiliki persyaratan dan tantangan subnetting mereka sendiri.

Intranet menggunakan ruang alamat IPv4 privat. Ini memungkinkan anda untuk menggunakan salah satu alamat jaringan IPv4 **privat** termasuk **prefix** 10.0.0.0/8 dengan 24 bit host dan lebih dari 16 juta host. Menggunakan alamat jaringan dengan bit host 24 membuat subnetting lebih mudah dan lebih fleksibel. Ini termasuk subnetting pada batas oktet menggunakan /16 atau /24.

Misalnya, alamat jaringan IPv4 **privat** 10.0.0.0/8 dapat disubnetasi menggunakan **mask** /16. Seperti yang ditunjukkan dalam tabel, ini menghasilkan 256 subnet, dengan 65.534 host per subnet. Jika organisasi memiliki kebutuhan kurang dari 200 subnet, memungkinkan pertumbuhan tertentu, ini memberi setiap subnet lebih dari cukup alamat host.

Subnetting Jaringan 10.0.0.0/8 menggunakan /16

| Alamat Subnet (256 Kemungkinan Subnet) | Range Host (65.534 host yang mungkin per subnet) | Broadcast |
|--|--|----------------|
| 10.0.0.0/16 | 10.0.0.1 – 10.0.255.254 | 10.0.255.255 |
| 10.1.0.0/16 | 10.1.0.1 – 10.1.255.254 | 10.1.255.255 |
| 10.2.0.0/16 | 10.2.0.1 – 10.2.255.254 | 10.2.255.255 |
| 10.3.0.0/16 | 10.3.0.1 – 10.3.255.254 | 10.3.255.255 |
| 10.4.0.0/16 | 10.4.0.1 – 10.4.255.254 | 10.4.255.255 |
| 10.5.0.0/16 | 10.5.0.1 – 10.5.255.254 | 10.5.255.255 |
| 10.6.0.0/16 | 10.6.0.1 – 10.6.255.254 | 10.6.255.255 |
| 10.7.0.0/16 | 10.7.0.1 – 10.7.255.254 | 10.7.255.255 |
| ... | ... | ... |
| 10.255.0.0/16 | 10.255.0.1 – 10.255.255.254 | 10.255.255.255 |

Opsi lain menggunakan alamat jaringan IPv4 **privat** 10.0.0.0/8 adalah subnet menggunakan **mask** /24. Seperti yang ditunjukkan dalam tabel, ini menghasilkan 65.536 subnet, dengan 254 host per subnet. Jika anda membutuhkan lebih dari 256 subnet, maka menggunakan /24 dapat digunakan dengan 254 host per subnet.

Subnetting Jaringan 10.0.0.0/8 menggunakan /24

| Range Host (254 host yang mungkin per subnet) | Broadcast |
|---|----------------|
| 10.0.0.1 – 10.0.0.254 | 10.0.0.255 |
| 10.0.1.1 – 10.0.1.254 | 10.0.1.255 |
| 10.0.2.1 – 10.0.2.254 | 10.0.2.255 |
| ... | ... |
| 10.0.255.1 – 10.0.255.254 | 10.0.255.255 |
| 10.1.0.1 – 10.1.0.254 | 10.1.0.255 |
| 10.1.1.1 – 10.1.1.254 | 10.1.1.255 |
| 10.1.2.1 – 10.1.2.254 | 10.1.2.255 |
| ... | ... |
| 10.100.0.1 – 10.100.0.254 | 10.100.0.255 |
| ... | ... |
| 10.255.255.1 – 10.255.255.254 | 10.255.255.255 |

10.0.0.0/8 juga dapat disubnetisasi menggunakan jumlah panjang **prefix** lainnya, seperti /12, /18, /20, dll. Ini akan memberi administrator jaringan berbagai opsi. Menggunakan alamat jaringan IPv4 **privat** 10.0.0.0/8 memudahkan perencanaan dan implementasi subnet.

Bagaimana dengan DMZ?

Karena perangkat ini harus dapat diakses secara publik dari internet, perangkat di DMZ memerlukan alamat IPv4 publik. Menipisnya ruang alamat IPv4 publik menjadi isu yang dimulai pada pertengahan 1990-an. Sejak 2011, IANA dan empat dari lima RIR telah kehabisan ruang alamat IPv4. Meskipun organisasi melakukan transisi ke IPv6, ruang alamat IPv4 yang tersisa tetap sangat terbatas. Ini berarti sebuah organisasi harus memaksimalkan jumlah alamat IPv4 publiknya sendiri yang terbatas. Ini mengharuskan administrator jaringan untuk mensubnet ruang alamat publik mereka ke subnet dengan **subnet mask** yang berbeda, untuk meminimalkan jumlah alamat host yang tidak digunakan per subnet. Ini dikenal sebagai Variable Length subnet mask (VLSM).

Meminimalkan Alamat IPv4 Host yang Tidak Digunakan dan Memaksimalkan Subnet

Untuk meminimalkan jumlah alamat IPv4 host yang tidak digunakan dan memaksimalkan jumlah subnet yang tersedia, ada dua pertimbangan saat merencanakan subnet: jumlah alamat host yang diperlukan untuk setiap jaringan dan jumlah subnet individu yang diperlukan.

Tabel menampilkan spesifikasi untuk memonetisasi jaringan /24. Perhatikan bagaimana ada hubungan terbalik antara jumlah subnet dan jumlah host. Semakin banyak bit yang dipinjam untuk membuat subnet, semakin sedikit bit host yang tetap tersedia. Jika lebih banyak alamat host diperlukan, lebih banyak bit host diperlukan, menghasilkan lebih sedikit subnet.

Jumlah alamat host yang diperlukan dalam subnet terbesar akan menentukan berapa banyak bit yang harus ditinggalkan dalam **host portion**. Ingat bahwa dua alamat tidak dapat digunakan, sehingga jumlah alamat yang dapat digunakan dapat dihitung sebagai $2^n - 2$.

Subnetting /24 Network

| Subnet mask | Subnet mask dalam Biner (n = jaringan, h = host) | # subnet | # host per subnet |
|---------------------|--|----------|-------------------|
| 255.255.255.1 28 | nnnnnnnn.nnnnnnnn.nnnnnnnn.nhhhhhhh 11111111.11111111.11111111.10000000 | 2 | 126 |
| 255.255.255.1 92 | nnnnnnnn.nnnnnnnn.nnnnnnnn.nnhhhhhh 11111111.11111111.11111111.11000000 | 4 | 62 |
| 255.255.255.2 24 | nnnnnnnn.nnnnnnnn.nnnnnnnn.nnnhhhhh 11111111.11111111.11111111.11100000 | 8 | 30 |
| 255.255.255.2 40 | nnnnnnnn.nnnnnnnn.nnnnnnnn.nnnnhhhh 11111111.11111111.11111111.11110000 | 16 | 14 |
| 255.255.255.2 48 | nnnnnnnn.nnnnnnnn.nnnnnnnn.nnnnnhhh 11111111.11111111.11111111.11111000 | 32 | 6 |
| 255.255.255.2 52 | nnnnnnnn.nnnnnnnn.nnnnnnnn.nnnnnngh 11111111.11111111.11111111.11111100 | 64 | 2 |

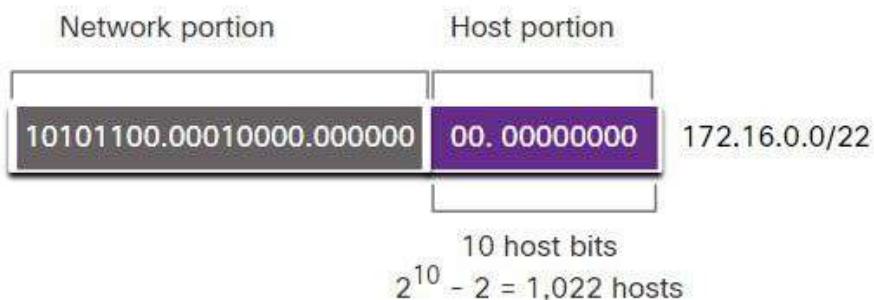
Administrator jaringan harus menyusun skema alamat jaringan untuk mengakomodasi jumlah maksimum host untuk setiap jaringan dan jumlah **subnet**. Skema mengatasi harus memungkinkan pertumbuhan dalam jumlah alamat host per subnet dan jumlah total subnet.

Contoh: Subnetting IPv4 yang Efisien

Dalam contoh ini, kantor pusat perusahaan telah dialokasikan alamat jaringan publik 172.16.0.0/22 (10 bit host) oleh ISP-nya. Seperti yang ditunjukkan dalam gambar, ini akan menyediakan 1.022 alamat host.

Catatan: 172.16.0.0/22 adalah bagian dari ruang alamat **privat** IPv4. Kami menggunakan alamat ini alih-alih alamat IPv4 publik yang sebenarnya.

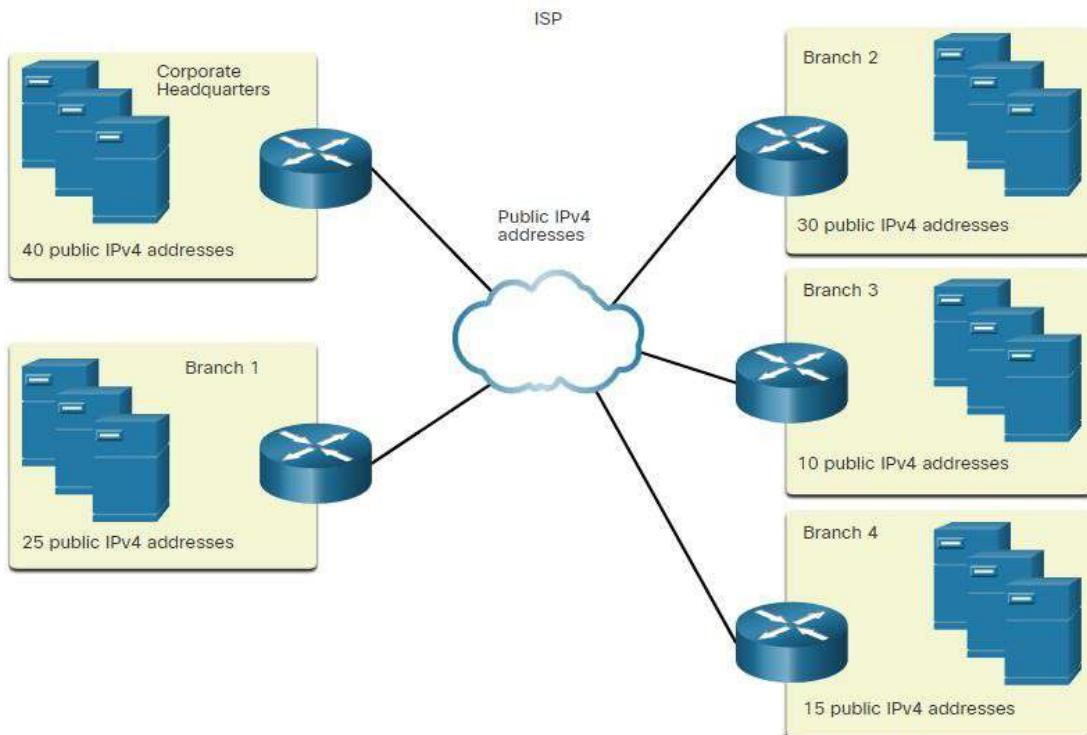
Alamat Jaringan



Kantor pusat perusahaan memiliki DMZ dan empat kantor cabang, masing-masing membutuhkan ruang alamat IPv4 publik sendiri. Kantor pusat perusahaan perlu memanfaatkan ruang alamat IPv4 terbatas dengan sebaik-baiknya.

Topologi yang ditunjukkan pada gambar terdiri dari lima situs; kantor perusahaan dan empat situs cabang. Setiap situs membutuhkan koneksi internet dan oleh karena itu, lima koneksi internet. Ini berarti bahwa organisasi memerlukan 10 subnet dari alamat publik perusahaan 172.16.0.0/22. Subnet terbesar memerlukan 40 alamat.

Topologi Perusahaan dengan Lima Situs



Alamat jaringan 172.16.0.0/22 memiliki 10 bit host, seperti yang ditunjukkan pada gambar. Karena subnet terbesar membutuhkan 40 host, minimal 6 bit host diperlukan untuk menyediakan alamat untuk 40 host. Ini ditentukan dengan menggunakan rumus ini: $2^6 - 2 = 62$ host.

Skema Subnet

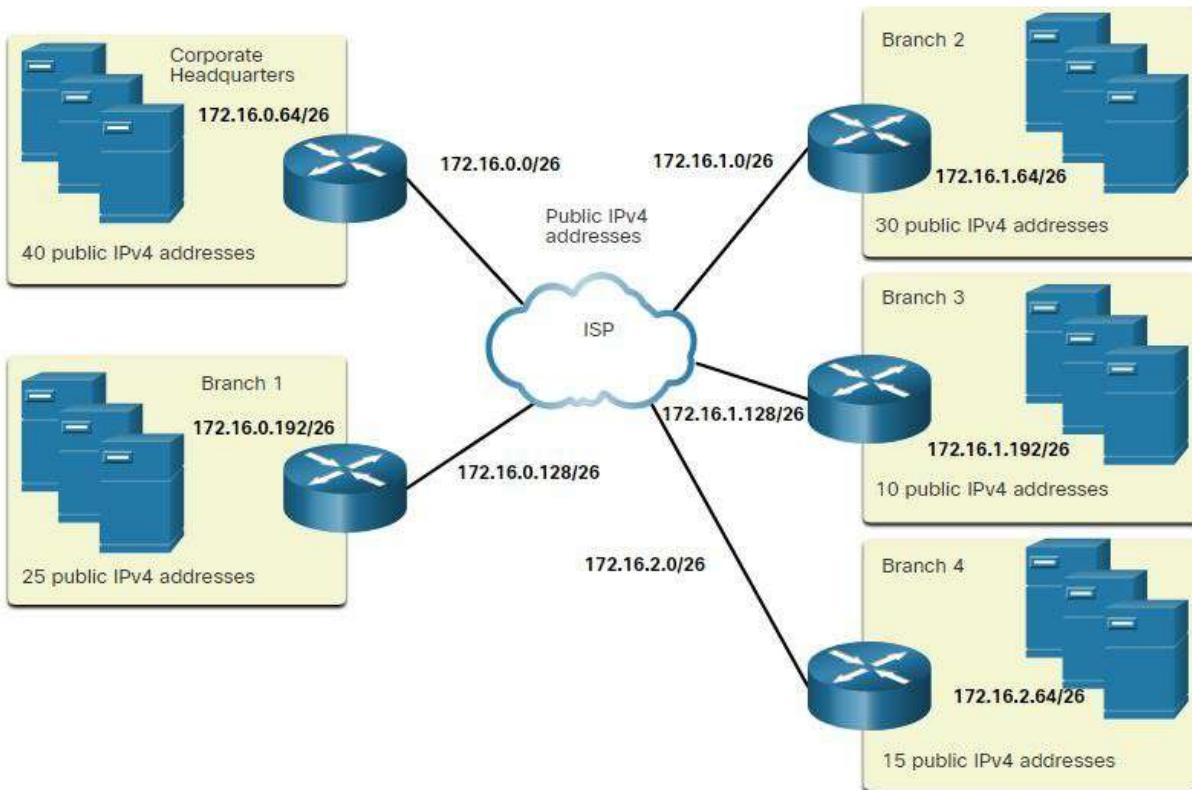
| | Network portion | Host portion | Dotted Decimal |
|---|--------------------------|--------------|-----------------|
| | 10101100.00010000.000000 | 00.00 000000 | 172.16.0.0/22 |
| 0 | 10101100.00010000.000000 | 00.00 000000 | 172.16.0.0/26 |
| 1 | 10101100.00010000.000000 | 00.01 000000 | 172.16.0.64/26 |
| 2 | 10101100.00010000.000000 | 00.10 000000 | 172.16.0.128/26 |
| 3 | 10101100.00010000.000000 | 00.11 000000 | 172.16.0.192/26 |
| 4 | 10101100.00010000.000000 | 01.00 000000 | 172.16.1.0/26 |
| 5 | 10101100.00010000.000000 | 01.01 000000 | 172.16.1.64/26 |
| 6 | 10101100.00010000.000000 | 01.10 000000 | 172.16.1.128/26 |
| Nets 7 – 13 not shown | | | |
| 14 | 10101100.00010000.000000 | 11.10 000000 | 172.16.3.128/26 |
| 15 | 10101100.00010000.000000 | 11.11 000000 | 172.16.3.192/26 |
| 4-bits borrowed from host portion to create subnets | | | |

Menggunakan rumus untuk menentukan subnet menghasilkan 16 subnet: $2^4 = 16$. Karena contoh internetwork membutuhkan 10 subnet, ini akan memenuhi persyaratan dan memungkinkan untuk beberapa pertumbuhan tambahan.

Oleh karena itu, 4 bit host pertama dapat digunakan untuk mengalokasikan subnet. Ini berarti dua bit dari oktet ke-3 dan dua bit dari oktet ke-4 akan dipinjam. Ketika 4 bit dipinjam dari jaringan 172.16.0.0/22, panjang **prefix** baru adalah /26 dengan **subnet mask** 255.255.255.192.

Seperti yang ditunjukkan pada gambar ini, subnet dapat ditetapkan ke setiap lokasi dan koneksi router-ke-ISP.

Subnet Penetapan untuk setiap Situs dan ISP



Variable Length Subnet Mask

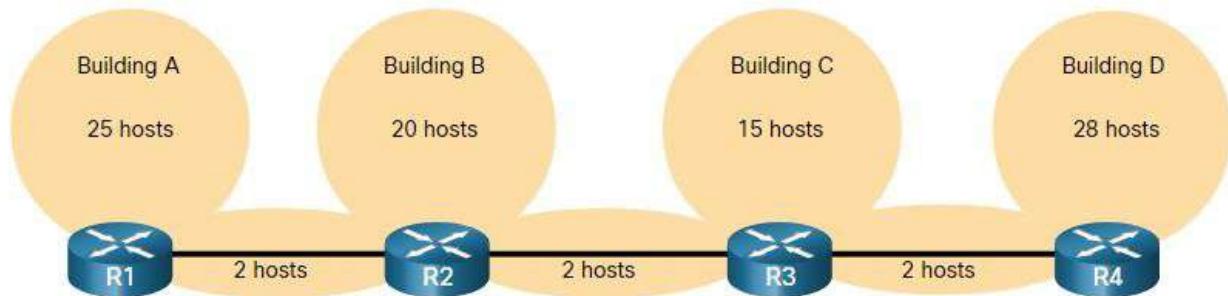
Karena menipisnya ruang alamat IPv4 publik, memaksimalkan alamat host yang tersedia adalah perhatian utama saat mensubnet jaringan IPv4.

Catatan: Alamat IPv6 yang lebih besar memungkinkan perencanaan dan alokasi alamat yang jauh lebih mudah daripada yang memungkinkan IPv4. Melestarikan alamat IPv6 bukanlah masalah. Ini adalah salah satu kekuatan pendorong untuk transisi ke IPv6.

Konservasi Alamat IPv4

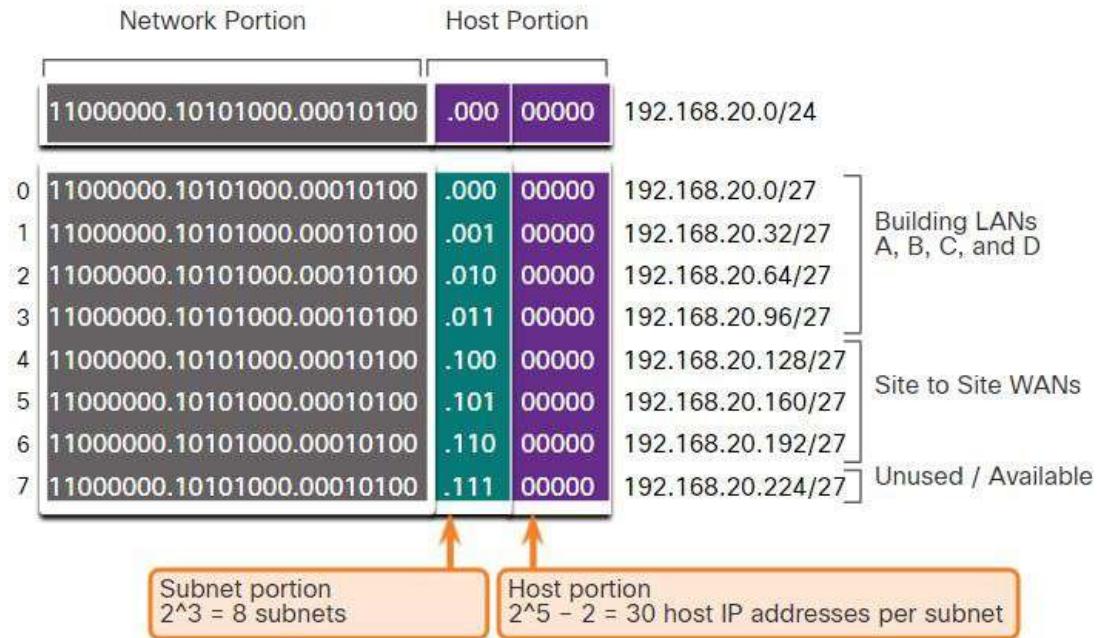
Menggunakan **subnetting tradisional**, jumlah alamat yang sama dialokasikan untuk setiap subnet. Jika semua subnet memiliki persyaratan yang sama untuk jumlah host, atau jika menghemat ruang alamat IPv4 tidak menjadi masalah, blok alamat ukuran tetap ini akan efisien. Biasanya, dengan alamat IPv4 publik, itu tidak terjadi.

Misalnya, topologi yang ditunjukkan dalam gambar membutuhkan tujuh subnet, satu untuk masing-masing dari empat LAN, dan satu untuk masing-masing dari tiga koneksi antara router.

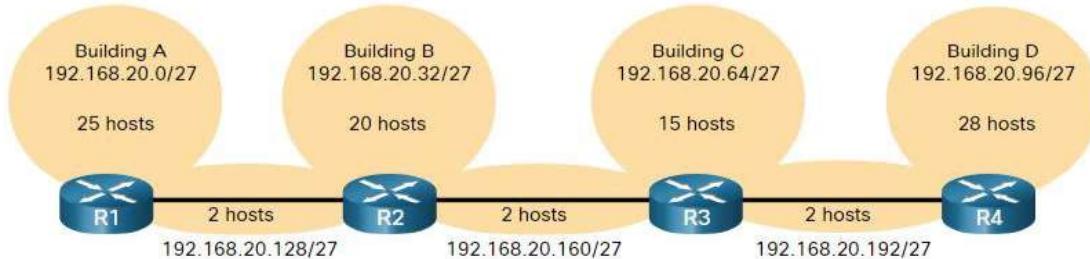


Menggunakan subnetting tradisional dengan alamat yang diberikan 192.168.20.0/24, tiga bit dapat dipinjam dari **host portion** di oktet terakhir untuk memenuhi persyaratan subnet tujuh subnet. Seperti yang ditunjukkan pada gambar, meminjam 3 bit menciptakan 8 subnet dan meninggalkan 5 bit host dengan 30 host yang dapat digunakan per subnet. Skema ini menciptakan subnet yang diperlukan dan memenuhi persyaratan **host LAN**.

Skema Subnet Dasar



Ketujuh subnet ini dapat ditetapkan ke jaringan LAN dan WAN, seperti yang ditunjukkan pada angka tersebut.



Meskipun subnetting tradisional ini memenuhi kebutuhan **LAN** dan membagi ruang alamat menjadi jumlah subnet yang memadai, itu menghasilkan pemborosan alamat yang tidak digunakan secara signifikan.

Misalnya, hanya dua alamat yang diperlukan di setiap subnet untuk tiga **link WAN**. Karena setiap subnet memiliki 30 alamat yang dapat digunakan, ada 28 alamat yang tidak digunakan di masing-masing subnet ini. Seperti yang ditunjukkan pada gambar, ini menghasilkan 84 alamat yang tidak digunakan (28×3).

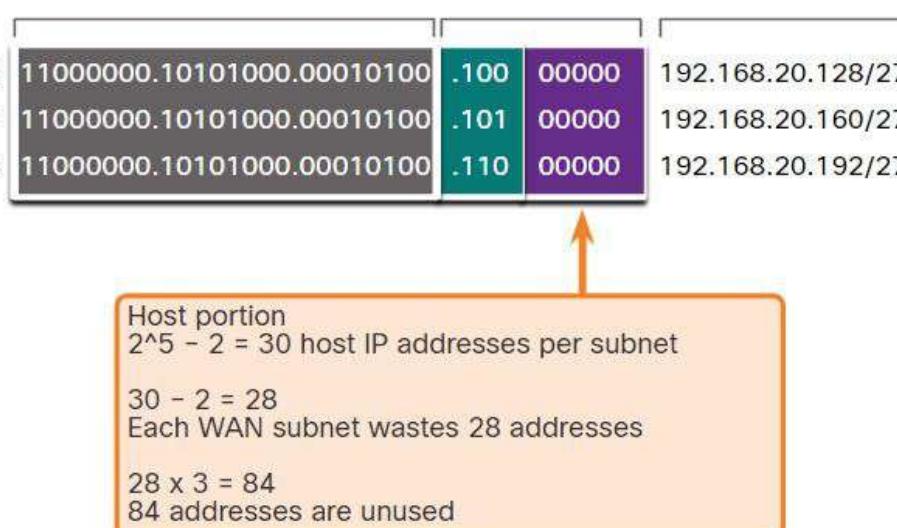
Alamat yang Tidak Digunakan di Subnet WAN

| | Network Portion | Host Portion | Dotted Decimal |
|---|----------------------------|--------------|-------------------|
| 4 | 11000000.10101000.00010100 | .100 00000 | 192.168.20.128/27 |
| 5 | 11000000.10101000.00010100 | .101 00000 | 192.168.20.160/27 |
| 6 | 11000000.10101000.00010100 | .110 00000 | 192.168.20.192/27 |

Host portion
 $2^5 - 2 = 30$ host IP addresses per subnet

$30 - 2 = 28$
Each WAN subnet wastes 28 addresses

$28 \times 3 = 84$
84 addresses are unused



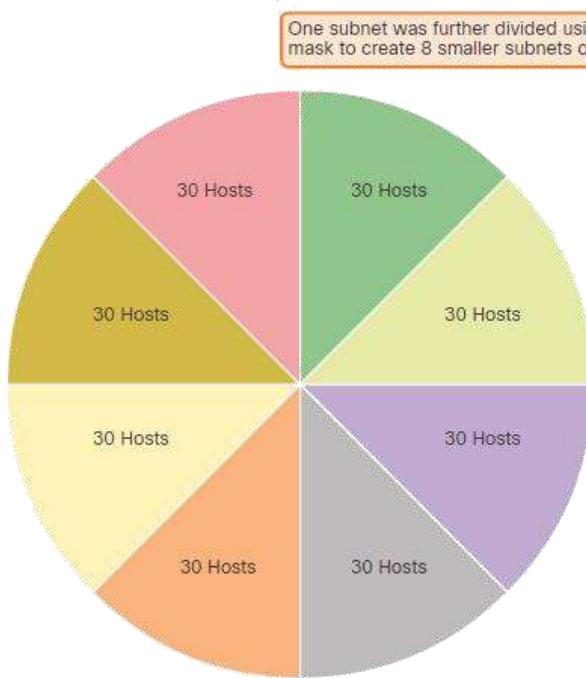
Selanjutnya, ini membatasi pertumbuhan di masa depan dengan mengurangi jumlah total subnet yang tersedia. Penggunaan alamat yang tidak efisien ini adalah karakteristik subnetting tradisional. Menerapkan skema subnetting tradisional untuk skenario ini tidak terlalu efisien dan boros.

Variable Length Subnet Mask (VLSM) dikembangkan untuk menghindari pemarahan alamat dengan memungkinkan kami untuk men subnet subnet.

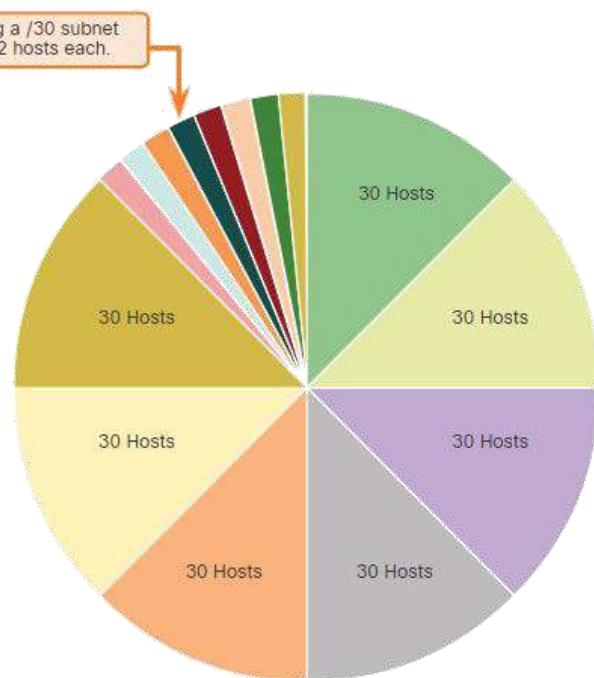
VLSM

Dalam semua contoh subnetting sebelumnya, **subnet mask** yang sama diterapkan untuk semua subnet. Ini berarti bahwa setiap subnet memiliki jumlah alamat host yang tersedia yang sama. Seperti yang diilustrasikan di sisi kiri gambar, subnetting tradisional menciptakan subnet dengan ukuran yang sama. Setiap subnet dalam skema tradisional menggunakan **subnet mask** yang sama. Seperti yang ditunjukkan di sisi kanan gambar, VLSM memungkinkan ruang jaringan dibagi menjadi bagian yang tidak sama. Dengan VLSM, **subnet mask** akan bervariasi tergantung pada berapa banyak bit yang telah dipinjam untuk subnet tertentu, sehingga bagian "variabel" dari VLSM.

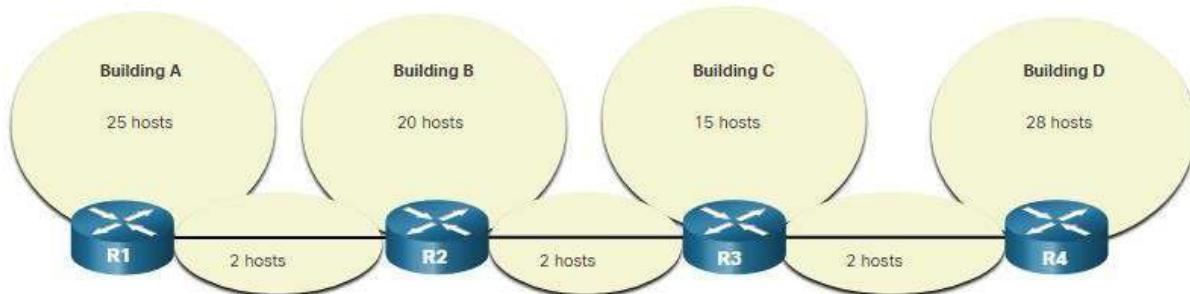
Traditional Subnetting Creates Equal Sized Subnets



Subnets of Varying Sizes

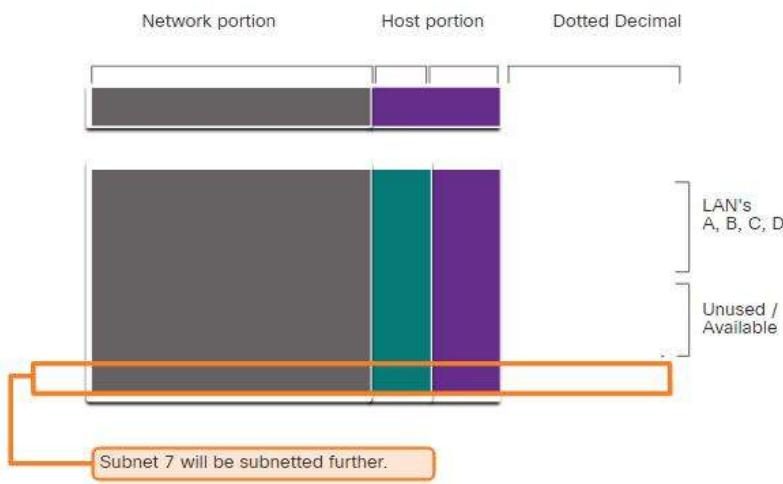


VLSM hanya **subnetting** subnet. Topologi yang sama yang digunakan sebelumnya ditunjukkan pada gambar. Sekali lagi, kita akan menggunakan jaringan 192.168.20.0/24 dan subnet untuk tujuh subnet, satu untuk masing-masing dari empat LAN, dan satu untuk masing-masing dari tiga koneksi antara router.



Angka tersebut memperlihatkan bagaimana jaringan 192.168.20.0/24 disubnetisasikan menjadi delapan subnet berukuran sama dengan 30 alamat host yang dapat digunakan per subnet. Empat subnet digunakan untuk LAN dan tiga subnet dapat digunakan untuk koneksi antara router.

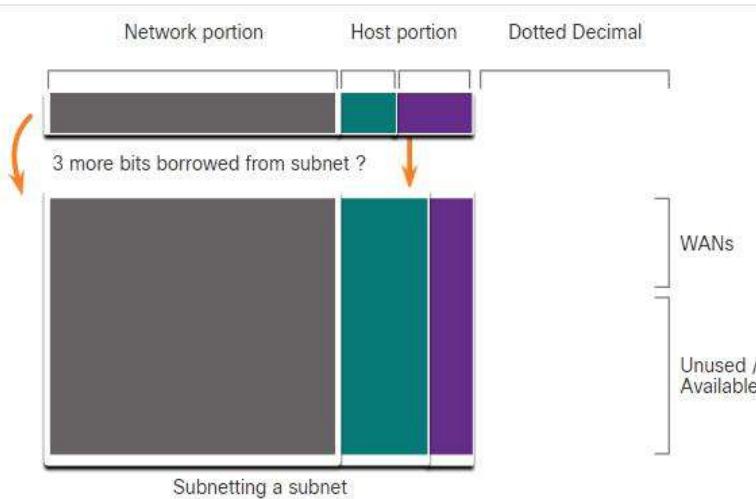
Skema Subnetting Dasar



Namun, koneksi antara router hanya memerlukan dua alamat host per subnet (satu alamat host untuk setiap **interface** router). Saat ini semua subnet memiliki 30 alamat host yang dapat digunakan per subnet. Untuk menghindari pemakanan 28 alamat per subnet, VLSM dapat digunakan untuk membuat subnet yang lebih kecil untuk koneksi antar-router.

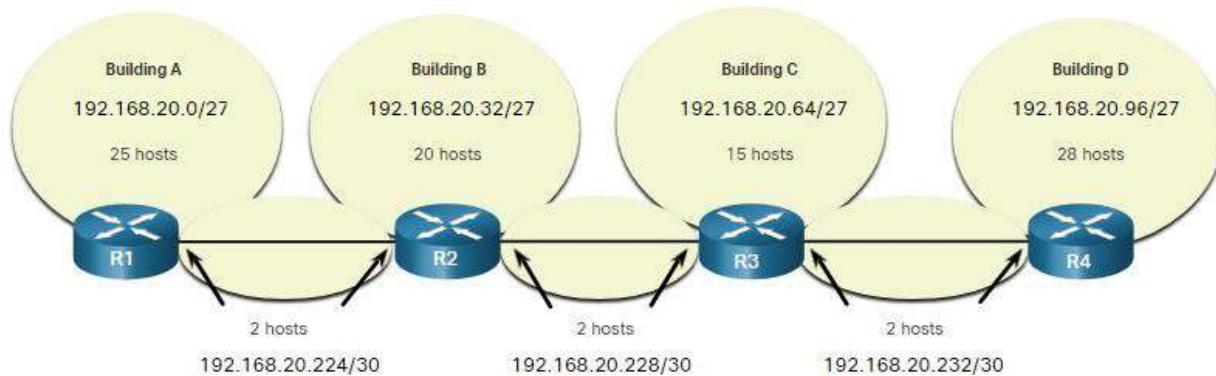
Untuk membuat subnet yang lebih kecil untuk **link** antar-router, salah satu subnet akan dibagi. Dalam contoh ini, subnet terakhir, 192.168.20.224/27, akan disubnetisasi lebih lanjut. Angka tersebut menunjukkan subnet terakhir telah disubnetisasi lebih lanjut dengan menggunakan subnet mask 255.255.255.252 atau /30.

Skema Subnetting VLSM



Mengapa /30? Ingat bahwa ketika jumlah alamat host yang diperlukan diketahui, rumus $2^n - 2$ (di mana n sama dengan jumlah bit host yang tersisa) dapat digunakan. Untuk memberikan dua alamat yang dapat digunakan, dua bit host harus ditinggalkan di bagian host.

Karena ada lima bit host di ruang alamat 192.168.20.224/27 yang di subnetisasi, tiga bit lagi dapat dipinjam, meninggalkan dua bit di **host portion**. Perhitungan pada titik ini persis sama dengan yang digunakan untuk subnetting tradisional. Bit dipinjam, dan rentang subnet ditentukan. Angka tersebut menunjukkan bagaimana empat/27 subnet telah ditetapkan ke LAN dan tiga subnet /30 telah ditetapkan ke **link** antar-router.



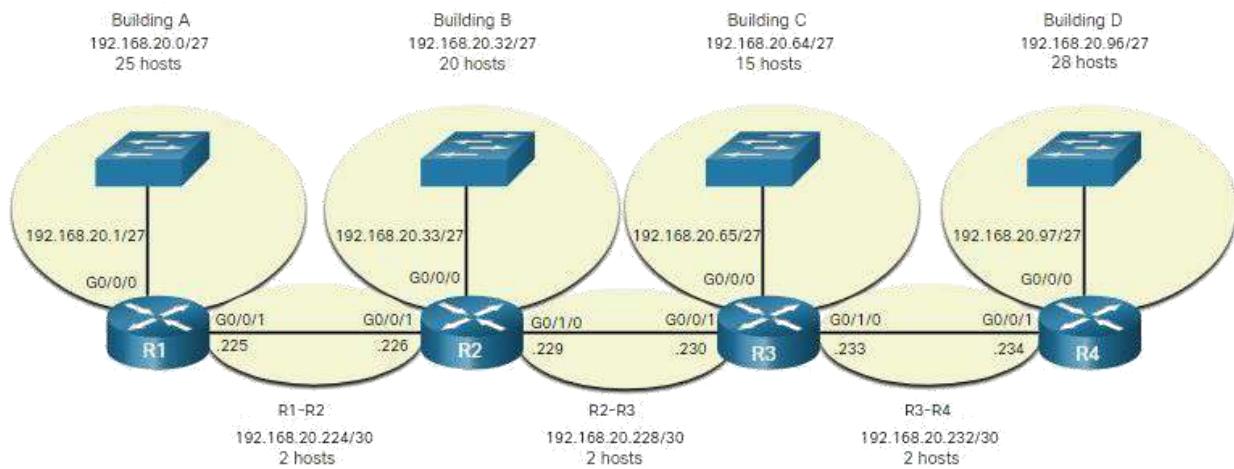
Skema subnetting VLSM ini mengurangi jumlah alamat per subnet ke ukuran yang sesuai untuk jaringan yang memerlukan lebih sedikit subnet. Subnetting subnetting 7 untuk **link** antar-router, memungkinkan subnet 4, 5, dan 6 tersedia untuk jaringan mendatang, serta lima subnet tambahan yang tersedia untuk koneksi antar-router.

Catatan: Saat menggunakan VLSM, selalu mulailah dengan memenuhi persyaratan host subnet terbesar. Lanjutkan berlangganan sampai persyaratan host subnet terkecil terpenuhi.

Penugasan Alamat Topologi VLSM

Menggunakan subnet VLSM, jaringan LAN dan antar-router dapat diatasi tanpa limbah yang tidak perlu.

Gambar menunjukkan penetapan alamat jaringan dan alamat IPv4 yang ditetapkan untuk setiap **interface** router.



Menggunakan skema alamat umum, alamat IPv4 host pertama untuk setiap subnet ditetapkan ke **interface LAN** router. Host di setiap subnet akan memiliki alamat IPv4 host dari berbagai alamat host untuk subnet tersebut dan masker yang sesuai. **Host** akan menggunakan alamat **interface LAN** router terlampir sebagai alamat gateway default.

Tabel memperlihatkan alamat jaringan dan rentang alamat host untuk setiap jaringan. Alamat Default Gateway ditampilkan untuk empat LAN.

| Alamat Jaringan | Range Host Address | Alamat Gateway Asal |
|-------------------|--|---------------------|
| 192.168.20.0/27 | 192.168.20.1/27 hingga 192.168.20.30/27 | 192.168.20.1/27 |
| 192.168.20.32/27 | 192.168.20.33/27 hingga 192.168.20.62/27 | 192.168.20.33/27 |
| 192.168.20.64/27 | 192.168.20.65/27 hingga 192.168.20.94/27 | 192.168.20.65/27 |
| 192.168.20.96/27 | 192.168.20.97/27 hingga 192.168.20.126/27 | 192.168.20.97/27 |
| 192.168.20.224/30 | 192.168.20.225/30 hingga 192.168.20.226/30 | |
| 192.168.20.228/30 | 192.168.20.229/30 hingga 192.168.20.230/30 | |
| 192.168.20.232/30 | 192.168.20.233/30 hingga 192.168.20.234/30 | |

Desain Terstruktur

Sebelum mulai subnetting, Anda harus mengembangkan skema penuh pada **alamat IPv4** untuk seluruh jaringan Anda. Anda perlu mengetahui berapa banyak subnet yang Anda butuhkan, berapa banyak host yang diperlukan subnet tertentu, perangkat apa yang merupakan bagian dari subnet, bagian mana dari jaringan Anda yang menggunakan **private address**, dan yang menggunakan publik, dan banyak faktor penentu lainnya. **Skema** mengatasi yang baik memungkinkan pertumbuhan. **Skema** mengatasi yang baik juga merupakan tanda administrator jaringan yang baik.

Perencanaan Alamat Jaringan IPv4

Merencanakan subnet jaringan IPv4 mengharuskan Anda untuk memeriksa kebutuhan penggunaan jaringan organisasi, dan bagaimana subnet akan terstruktur. Melakukan studi persyaratan jaringan adalah titik awal. Ini berarti melihat seluruh jaringan, baik intranet maupun DMZ, dan menentukan bagaimana setiap area akan disegmentasi. Rencana alamat termasuk menentukan di mana **address conservation** diperlukan (biasanya dalam DMZ), dan di mana ada lebih banyak fleksibilitas (biasanya dalam intranet).

Jika **address conservation** diperlukan, rencana harus menentukan berapa banyak subnet yang diperlukan dan berapa banyak host per subnet. Seperti yang dibahas sebelumnya, ini biasanya diperlukan untuk ruang alamat IPv4 publik dalam DMZ. Ini kemungkinan besar akan mencakup penggunaan VLSM.

Dalam intranet perusahaan, konservasi alamat biasanya kurang dari masalah. Ini sebagian besar disebabkan oleh penggunaan alamat IPv4 pribadi, termasuk 10.0.0.0/8, dengan lebih dari 16 juta alamat IPv4 host.

Untuk sebagian besar organisasi, alamat IPv4 privat memungkinkan lebih dari cukup alamat internal (intranet). Bagi banyak organisasi dan ISP yang lebih besar, bahkan ruang alamat IPv4 pribadi tidak cukup besar untuk mengakomodasi kebutuhan internal mereka. Ini adalah alasan lain mengapa organisasi beralih ke IPv6.

Untuk intranet yang menggunakan alamat IPv4 pribadi dan DMZ yang menggunakan alamat IPv4 publik, **address planning and assignment** adalah penting.

Jika diperlukan, **address planning** termasuk menentukan kebutuhan setiap subnet dalam hal ukuran. Berapa banyak host yang akan ada per subnet? **address planning** juga perlu mencakup bagaimana alamat host akan ditetapkan, yang host akan memerlukan alamat IPv4

statis, dan host mana yang dapat menggunakan DHCP untuk mendapatkan informasi alamat mereka. Ini juga akan membantu mencegah duplikasi alamat, sambil memungkinkan pemantauan dan pengelolaan alamat karena alasan kinerja dan keamanan.

Mengetahui persyaratan alamat IPv4 Anda akan menentukan rentang, atau rentang, alamat host yang Anda terapkan dan membantu memastikan bahwa ada cukup alamat untuk memenuhi kebutuhan jaringan Anda.

Penetapan Alamat Perangkat

Dalam jaringan, ada berbagai jenis perangkat yang memerlukan alamat:

- **End device client** – Sebagian besar jaringan mengalokasikan alamat IPv4 ke perangkat klien secara dinamis, menggunakan Dynamic Host Configuration Protocol (DHCP). Ini mengurangi beban pada staf dukungan jaringan dan hampir menghilangkan kesalahan entri. Dengan DHCP, alamat hanya disewakan untuk jangka waktu tertentu, dan dapat digunakan kembali ketika sewa berakhir. Ini adalah fitur penting untuk jaringan yang mendukung pengguna sementara dan perangkat nirkabel. Mengubah skema subnetting berarti bahwa server DHCP perlu dikonfigurasi ulang, dan klien harus memperbarui alamat IPv4 mereka. Klien IPv6 dapat memperoleh informasi alamat menggunakan DHCPv6 atau SLAAC.
- **Server dan periferal** – Ini harus memiliki alamat IP statis yang dapat diprediksi. Gunakan sistem penomor nama yang konsisten untuk perangkat ini.
- **Server yang dapat diakses dari internet** – Server yang perlu tersedia untuk umum di internet harus memiliki alamat IPv4 publik, paling sering diakses menggunakan NAT. Di beberapa organisasi, server internal (tidak tersedia untuk umum) harus tersedia untuk pengguna jarak jauh. Dalam kebanyakan kasus, server ini diberi **private address** secara internal, dan pengguna diharuskan untuk membuat koneksi jaringan pribadi virtual (VPN) untuk mengakses server. Ini memiliki efek yang sama seolah-olah pengguna mengakses server dari host dalam intranet.
- **Intermediary Devices – Perangkat** ini diberi alamat untuk manajemen jaringan, pemantauan, dan keamanan. Karena kita harus tahu bagaimana berkomunikasi dengan **Intermediary Devices**, mereka harus memiliki alamat yang dapat diprediksi, ditetapkan secara statis.
- **Gateway** – Router dan perangkat firewall memiliki alamat IP yang ditetapkan untuk setiap antarmuka yang berfungsi sebagai gateway untuk host di jaringan itu. Biasanya, antarmuka router menggunakan alamat terendah atau tertinggi dalam jaringan.

Saat mengembangkan skema mengatasi IP, umumnya di sarankan agar Anda memiliki pola yang ditetapkan tentang bagaimana alamat dialokasikan untuk setiap jenis perangkat. Ini menguntungkan administrator saat menambahkan dan menghapus perangkat, memfilter lalu lintas berdasarkan IP, serta menyederhanakan dokumentasi.

BAB 12

~ IPv6 Addressing ~

Judul Bab : Alamat IPv6**Tujuan Bab :** Mengimplementasikan Skema Alamat IPv6.Link Test Pemahaman : <https://s.id/-QxAw>

| Judul Materi | Objektivitas Materi |
|--------------------------------|---|
| Masalah IPv4 | Menjelaskan kenapa kita butuh alamat IPv6 |
| Representasi Alamat IPv6 | Menjelaskan bagaimana alamat IPv6 diwakili |
| Jenis Jenis IPv6 | Membandingkan tipe jaringan IPv6 |
| Konfigurasi Static GUA dan LLA | Menjelaskan bagaimana mengkonfigurasi static global unicast dan link local IPv6 |
| Alamat Dinamis untuk IPv6 GUA | Menjelaskan Bagaimana global unicast dapat terkonfigurasi secara otomatis |
| Alamat Dinamis untuk IPv6 LLA | Mengkonfigurasi link local dapat terkonfigurasi secara otomatis |
| Alamat IPv6 Multicast | Mengidentifikasi Alamat IPv6 |
| Subnet pada IPv6 | Mengimplementasikan sebuah skema subnet Alamat IPv6 |

Masalah IPv4

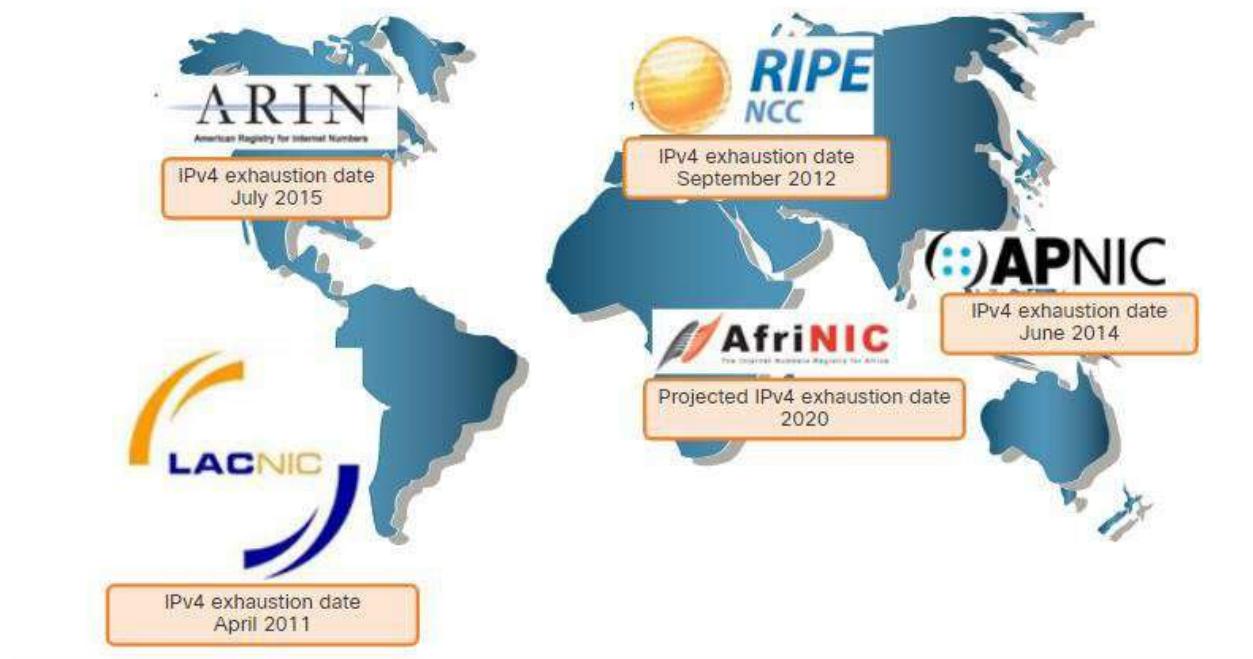
Anda sudah tahu bahwa IPv4 kehabisan alamat. Itu sebabnya Anda perlu belajar tentang IPv6.

Kebutuhan untuk IPv6

IPv6 dirancang untuk menjadi penerus IPv4. IPv6 memiliki ruang alamat 128-bit yang lebih besar, menyediakan 340 undecillion (yaitu, 340 diikuti oleh 36 nol) alamat yang mungkin. Namun, IPv6 lebih dari sekadar alamat yang lebih besar.

Ketika IETF memulai pengembangan penerus IPv4, IETF menggunakan kesempatan ini untuk memperbaiki keterbatasan IPv4 dan menyertakan peningkatan. Salah satu contohnya adalah Internet Control Message Protocol versi 6 (ICMPv6), yang mencakup resolusi alamat dan konfigurasi otomatis alamat yang tidak ditemukan di ICMP untuk IPv4 (ICMPv4).

Menipisnya ruang alamat IPv4 telah menjadi faktor yang memotivasi untuk pindah ke IPv6. Ketika Afrika, Asia dan daerah lain di dunia menjadi lebih terhubung ke internet, tidak ada cukup alamat IPv4 untuk mengakomodasi pertumbuhan ini. Seperti yang ditunjukkan dalam angka tersebut, empat dari lima RIR tersebut telah kehabisan alamat IPv4.



IPv4 memiliki maksimum teoritis 4,3 miliar alamat. Alamat privat yang dikombinasikan dengan Network Address Translation (NAT) telah berperan dalam memperlambat menipisnya ruang alamat IPv4. Namun, NAT bermasalah untuk banyak aplikasi, menciptakan latensi, dan memiliki keterbatasan yang sangat menghambat komunikasi peer-to-peer.

Dengan semakin banyaknya perangkat seluler, penyedia seluler telah memimpin dengan transisi ke IPv6. Dua penyedia seluler teratas di Amerika Serikat melaporkan bahwa lebih dari 90% lalu lintas mereka lebih dari IPv6.

Sebagian besar ISP dan penyedia konten top seperti YouTube, Facebook, dan NetFlix, juga telah melakukan transisi. Banyak perusahaan seperti Microsoft, Facebook, dan LinkedIn beralih ke internal khusus IPv6. Pada tahun 2018, broadband ISP Comcast melaporkan penyebaran lebih dari 65% dan British Sky Broadcasting lebih dari 86%.

Internet of Things

Internet saat ini sangat berbeda dari internet beberapa dekade terakhir. Internet saat ini lebih dari email, halaman web, dan transfer file antar komputer. Internet yang berkembang menjadi Internet of Things (IoT). Tidak lagi satu-satunya perangkat yang mengakses internet adalah komputer, tablet, dan smartphone. Perangkat siap internet yang dilengkapi sensor, besok akan mencakup segala sesuatu mulai dari mobil dan perangkat biomedis, hingga peralatan rumah tangga dan ekosistem alami.

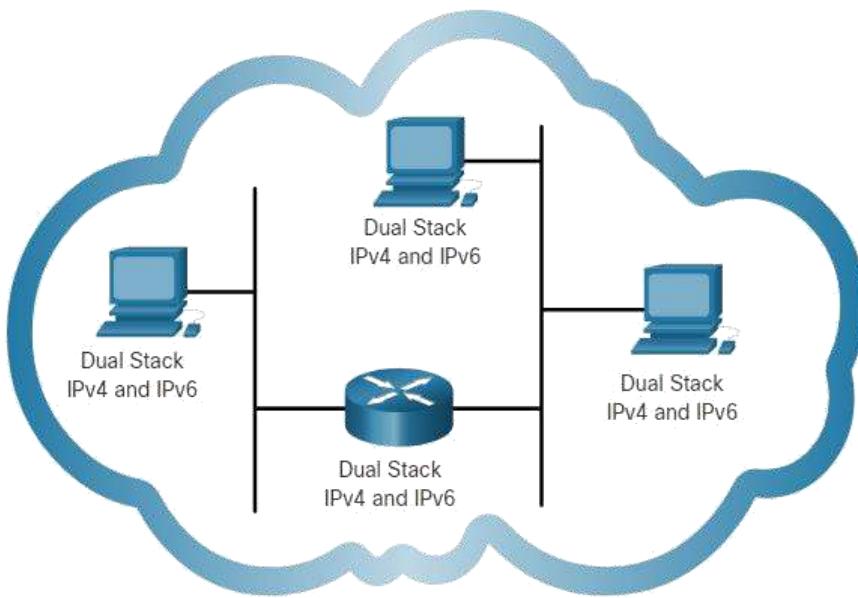
Dengan meningkatnya populasi internet, ruang alamat IPv4 terbatas, masalah dengan NAT dan IoT, waktunya telah tiba untuk memulai transisi ke IPv6.

IPv4 dan IPv6 Koeksistensi

Tidak ada tanggal tertentu untuk pindah ke IPv6. Baik IPv4 dan IPv6 akan hidup berdampingan dalam waktu dekat dan transisi akan memakan waktu beberapa tahun. IETF telah membuat berbagai protokol dan alat untuk membantu administrator jaringan memigrasikan jaringan mereka ke IPv6. Teknik migrasi dapat dibagi menjadi tiga kategori:

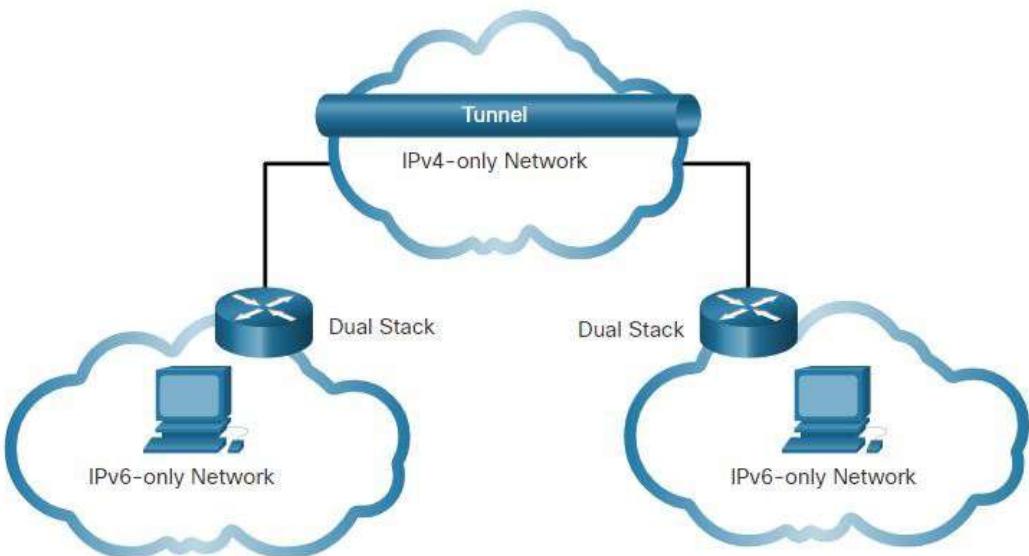
Dual Stack

Dual stack memungkinkan IPv4 dan IPv6 untuk hidup berdampingan pada segment jaringan yang sama. Perangkat **Dual stack** menggunakan **protokol stack** IPv4 dan IPv6 secara bersamaan. Dikenal sebagai IPv6 asli, ini berarti jaringan pelanggan memiliki koneksi IPv6 ke ISP mereka dan dapat mengakses konten yang ditemukan di internet melalui IPv6.



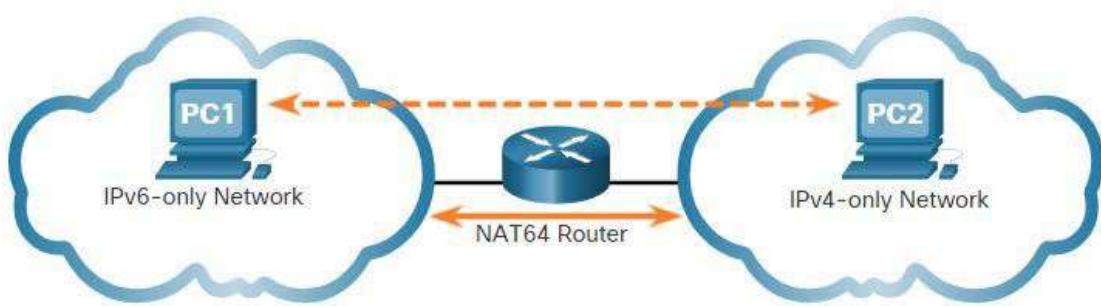
Tunneling

Tunneling adalah metode pengangkutan paket IPv6 melalui jaringan IPv4. Paket IPv6 dienkapsulasi di dalam paket IPv4, mirip dengan jenis data lainnya.



Translation

Network Address Translation 64 (NAT64) memungkinkan perangkat berkemampuan IPv6 untuk berkomunikasi dengan perangkat berkemampuan IPv4 menggunakan teknik terjemahan yang mirip dengan NAT untuk IPv4. Paket IPv6 diterjemahkan ke paket IPv4 dan paket IPv4 diterjemahkan ke paket IPv6.



Catatan: Tunneling dan translation adalah untuk transisi ke IPv6 asli dan hanya boleh digunakan jika diperlukan. Tujuannya harus komunikasi IPv6 asli dari **source** ke **destination**.

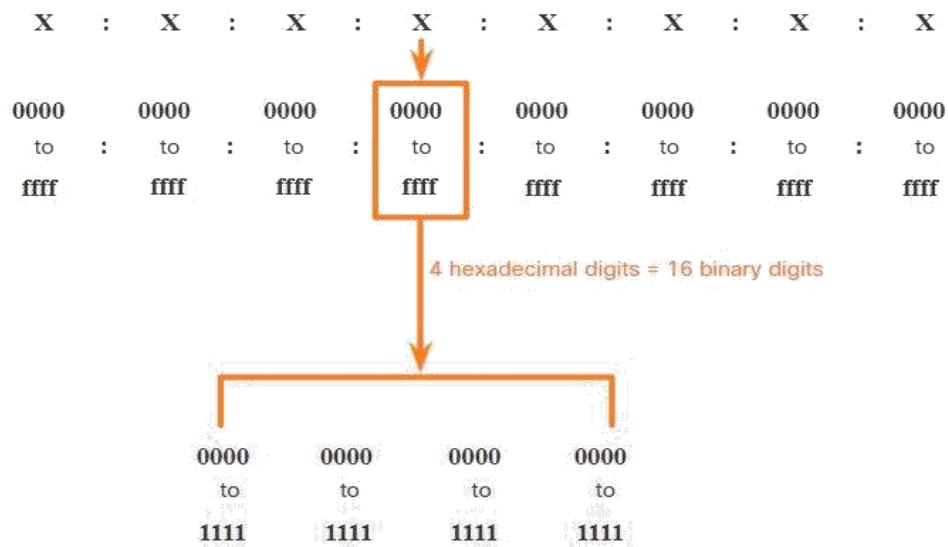
Representasi Alamat IPv6

Langkah pertama untuk mempelajari tentang IPv6 dalam jaringan adalah memahami cara alamat IPv6 ditulis dan diformat. Alamat IPv6 jauh lebih besar daripada alamat IPv4, itulah sebabnya kami tidak mungkin kehabisannya.

Format Mengatasi IPv6

Alamat IPv6 memiliki panjang 128 bit dan ditulis sebagai serangkaian nilai heksadesimal. Setiap empat bit diwakili oleh satu digit heksadesimal; dengan total 32 nilai heksadesimal, seperti yang ditunjukkan pada angka tersebut. Alamat IPv6 tidak peka huruf besar/kecil dan dapat ditulis dalam huruf kecil atau huruf besar.

Segmen atau Hextets 16-bit



Format Pilihan

Gambar sebelumnya juga menunjukkan bahwa format yang disukai untuk menulis alamat IPv6 adalah `x:x:x:x:x:x:x:x`, dengan masing-masing "x" terdiri dari empat nilai heksadesimal. Istilah oktet mengacu pada delapan bit alamat IPv4. Dalam IPv6, **hextet** adalah istilah tidak resmi yang digunakan untuk merujuk pada segmen 16 bit, atau empat nilai heksadesimal. Setiap "x" adalah **hextet** tunggal yang 16 bit atau empat digit heksadesimal.

Format yang disukai berarti Anda menulis alamat IPv6 menggunakan semua 32 digit heksadesimal. Ini tidak selalu berarti bahwa itu adalah metode yang ideal untuk mewakili alamat IPv6. Dalam materi ini, Anda akan melihat dua aturan yang membantu mengurangi jumlah digit yang diperlukan untuk mewakili alamat IPv6.

Ini adalah contoh alamat IPv6 dalam format pilihan.

```
2001 : 0db8 : 0000 : 1111 : 0000 : 0000 : 0000 : 0200
2001 : 0db8 : 0000 : 00a3 : abcd : 0000 : 0000 : 1234
2001 : 0db8 : 000a : 0001 : c012 : 9aff : fe9a : 19ac
2001 : 0db8 : aaaa : 0001 : 0000 : 0000 : 0000 : 0000
fe80 : 0000 : 0000 : 0000 : 0123 : 4567 : 89ab : cdef
fe80 : 0000 : 0000 : 0000 : 0000 : 0000 : 0000 : 0001
fe80 : 0000 : 0000 : 0000 : c012 : 9aff : fe9a : 19ac
fe80 : 0000 : 0000 : 0000 : 0123 : 4567 : 89ab : cdef
0000 : 0000 : 0000 : 0000 : 0000 : 0000 : 0000 : 0001
0000 : 0000 : 0000 : 0000 : 0000 : 0000 : 0000 : 0000
```

Aturan 1 – Hilangkan Nol Terdepan

Aturan pertama untuk membantu mengurangi notasi alamat IPv6 adalah menghilangkan 0s (nol) **leading** dalam hextet apa pun. Berikut adalah empat contoh cara untuk menghilangkan awalan nol:

- 01ab dapat diwakili sebagai 1ab
- 09f0 dapat diwakili sebagai 9f0
- 0a00 dapat diwakili sebagai a00
- 00ab dapat diwakili sebagai ab

Aturan ini hanya berlaku untuk 0s **leading**, TIDAK untuk membentuti 0s, jika tidak alamatnya akan ambigu. Misalnya, hextet “abc” bisa berupa “0abc” atau “abc0”, tetapi ini tidak mewakili nilai yang sama.

Omitting Leading 0s

| Type | Format |
|---------------|---|
| Preferred | 2001 : 0db8 : 0000 : 1111 : 0000 : 0000 : 0000 : 0200 |
| No leading 0s | 2001 : db8 : 0 : 1111 : 0 : 0 : 0 : 200 |
| | |
| Preferred | 2001 : 0db8 : 0000 : 00a3 : ab00 : 0ab0 : 00ab : 1234 |
| No leading 0s | 2001 : db8 : 0 : a3 : ab00 : ab0 : ab : 1234 |
| | |
| Preferred | 2001 : 0db8 : 000a : 0001 : c012 : 90ff : fe90 : 0001 |
| No leading 0s | 2001 : db8 : a : 1 : c012 : 90ff : fe90 : 1 |
| | |
| Preferred | 2001 : 0db8 : aaaa : 0001 : 0000 : 0000 : 0000 : 0000 |

| | |
|---------------|---|
| No leading 0s | 2001 : db8 : aaaa : 1 : 0 : 0 :0:0 |
| | |
| Preferred | fe80 : 0000 : 0000 : 0000 : 0123 : 4567 : 89ab : cdef |
| No leading 0s | fe80 : 0:0:0 : 123 : 4567 : 89ab : cdef |
| | |
| Preferred | fe80 : 0000 : 0000 : 0000 : 0000 : 0000 : 0000 : 0001 |
| No leading 0s | fe80 : 0:0:0 : 0 :0:0:1 |
| | |
| Preferred | 0000 : 0000 : 0000 : 0000 : 0000 : 0000 : 0000 : 0001 |
| No leading 0s | 0:0:0 :0:0 : 0 :0:1 |
| | |
| Preferred | 0000 : 0000 : 0000 : 0000 : 0000 : 0000 : 0000 : 0000 |
| No leading 0s | 0:0:0 :0:0 : 0 :0:0 |

Aturan 2- Double Colon

Aturan kedua untuk membantu mengurangi notasi alamat IPv6 adalah bahwa titik dua (:) dapat menggantikan string tunggal yang berseberangan dari satu atau lebih hextets 16-bit yang terdiri dari semua nol. Misalnya, 2001:db8:cafe:1:0:0:0:1 (leading 0s omitted) dapat diwakili sebagai 2001:db8:cafe:1::1. Titik dua (:) digunakan sebagai tempat dari tiga **hextet** all-0 (0:0:0).

Titik dua (:) hanya dapat digunakan sekali dalam alamat, jika tidak akan ada lebih dari satu alamat yang dihasilkan. Ketika digunakan dengan teknik 0s **leading** yang menghilangkan, notasi alamat IPv6 seringkali dapat sangat berkurang. Ini umumnya dikenal sebagai format terkompresi.

Berikut adalah contoh penggunaan titik dua yang salah: 2001:db8::abcd::1234.

Titik dua digunakan dua kali dalam contoh di atas. Berikut adalah kemungkinan perluasan alamat format terkompresi yang salah ini:

- 2001:db8::abcd:0000:0000:1234
- 2001:db8::abcd:0000:0000:0000:1234
- 2001:db8:0000:abcd::1234
- 2001:db8:0000:0000:abcd::1234

Jika alamat memiliki lebih dari satu string yang berseingin dari semua-0 hexets, praktik terbaik adalah menggunakan titik dua (::) pada string terpanjang. Jika string sama, string pertama harus menggunakan titik dua (::).

Omitting Leading 0s and All Segments

0

| Type | Format |
|--------------------------|--|
| <i>Preferred</i> | 2001 : 0db8 : 0000 : 1111 : 0000 : 0000 : 0200 |
| <i>Compressed/spaces</i> | 2001 : db8 : 0 : 1111 :: 200 |
| <i>Compressed</i> | 2001:db8:0:1111::200 |
| <i>Preferred</i> | 2001 : 0db8 : 0000 : 0000 : ab00 : 0000 : 0000 |
| <i>Compressed/spaces</i> | 2001 : db8 : 0 : 0 : ab00 :: |
| <i>Compressed</i> | 2001:db8:0:0:ab00:: |
| <i>Preferred</i> | 2001 : 0db8 : aaaa : 0001 : 0000 : 0000 : 0000 |
| <i>Compressed/spaces</i> | 2001 : db8 : aaaa : 1 :: |

| | |
|--------------------------|---|
| <i>Compressed</i> | 2001:db8:aaaa:1:: |
| <i>Preferred</i> | fe80 : 0000 : 0000 : 0000 : 0123 : 4567 : 89ab : cdef |
| <i>Compressed/spaces</i> | fe80 :: 123 : 4567 : 89ab : cdef |
| <i>Compressed</i> | fe80::123:4567:89ab:cdef |
| <i>Preferred</i> | fe80 : 0000 : 0000 : 0000 : 0000 : 0000 : 0000 : 0001 |
| <i>Compressed/spaces</i> | fe80 :: 1 |
| <i>Compressed</i> | fe80::1 |
| <i>Preferred</i> | 0000 : 0000 : 0000 : 0000 : 0000 : 0000 : 0000 : 0001 |
| <i>Compressed/spaces</i> | :: 1 |
| <i>Compressed</i> | ::1 |
| <i>Preferred</i> | 0000 : 0000 : 0000 : 0000 : 0000 : 0000 : 0000 : 0000 |
| <i>Compressed/spaces</i> | :: |
| <i>Compressed</i> | :: |

Jenis Jenis Alamat IPv6

Seperti halnya IPv4, ada berbagai jenis alamat IPv6. Bahkan, ada tiga kategori alamat IPv6 yang luas:

Unicast, Multicast, Anycast

- **Unicast** – Alamat unicast IPv6 secara unik mengidentifikasi **interface** pada perangkat yang mendukung IPv6.
- **Multicast** – Alamat multicast IPv6 digunakan untuk mengirim satu paket IPv6 ke beberapa **Destination**.
- **Anycast** – Alamat anycast IPv6 adalah alamat unicast IPv6 apa pun yang dapat ditetapkan ke beberapa perangkat. Paket yang dikirim ke alamat unicast routing ke perangkat terdekat yang memiliki alamat tersebut. Alamat Anycast berada di luar lingkup **Materi** ini.

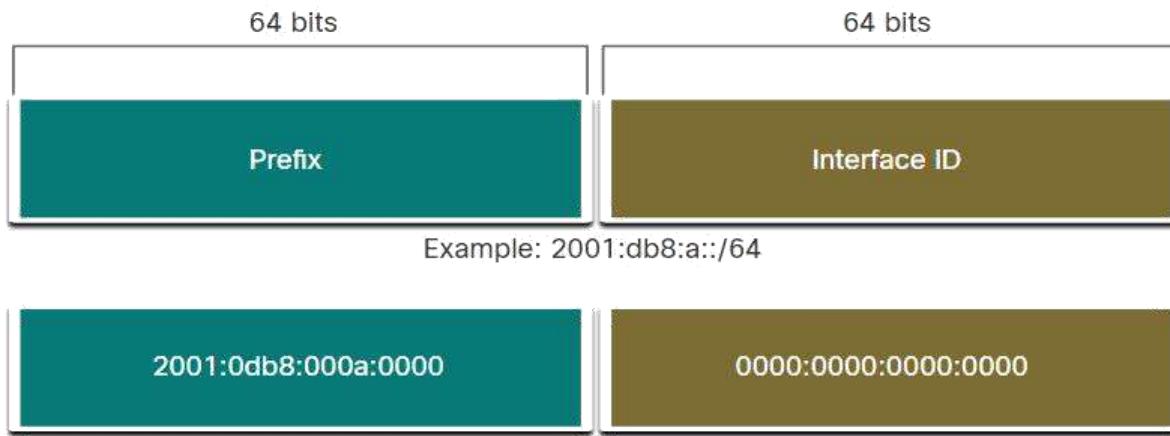
Tidak seperti IPv4, IPv6 tidak memiliki **broadcast address**. Namun, ada alamat multicast IPv6 all-nodes yang pada dasarnya memberikan hasil yang sama.

Prefix length IPv6

Prefix, atau **network portion**, dari alamat IPv4 dapat diidentifikasi dengan **dotted-decimal subnet mask** atau **prefix length** (notasi garis miring). Misalnya, alamat IPv4 192.168.1.10 dengan **dotted-decimal subnet mask** 255.255.255.0 setara dengan 192.168.1.10/24.

Di IPv6 hanya disebut **prefix length**. IPv6 tidak menggunakan notasi **dotted-decimal subnet mask**. Seperti IPv4, **prefix length** diwakili dalam notasi garis miring dan digunakan untuk menunjukkan **network portion** dari alamat IPv6.

Prefix length dapat berkisar antara 0 hingga 128. **Prefix length** IPv6 yang direkomendasikan untuk LAN dan sebagian besar jenis jaringan lainnya adalah /64, seperti yang ditunjukkan pada gambar.



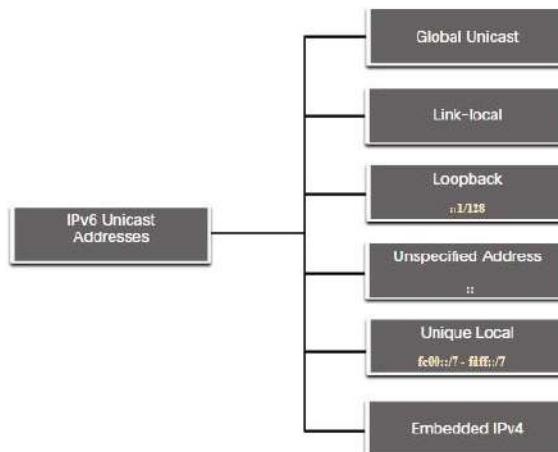
Prefix atau **network portion** alamat adalah 64 bit panjangnya, meninggalkan 64 bit lain untuk **Interface ID** (bagian host) alamat.

Sangat disarankan untuk menggunakan **Interface ID** 64-bit untuk sebagian besar jaringan. Ini karena stateless address autoconfiguration (SLAAC) menggunakan 64 bit untuk **Interface ID**. Ini juga membuat subnetting lebih mudah dibuat dan dikelola.

Jenis Alamat IPv6 Unicast

Alamat unicast IPv6 secara unik mengidentifikasi **interface** pada perangkat yang mendukung IPv6. Paket yang dikirim ke alamat unicast diterima oleh **interface** yang diberi alamat tersebut. Mirip dengan IPv4, **source ipv6 address** harus merupakan alamat unicast. **Destination ipv6 address** dapat berupa alamat unicast atau multicast. Gambar menunjukkan berbagai jenis alamat unicast IPv6.

Alamat IPv6 Unicast



Tidak seperti perangkat IPv4 yang hanya memiliki satu alamat, alamat IPv6 biasanya memiliki dua alamat unicast:

- **Global Unicast Address (GUA)** – Ini mirip dengan alamat IPv4 publik. Ini adalah alamat yang unik dan dapat di-routable internet secara global. GUAs dapat dikonfigurasi secara statis atau ditetapkan secara dinamis.
- **Link Local Address (LLA)** – Ini diperlukan untuk setiap perangkat yang mendukung IPv6. LLAs digunakan untuk berkomunikasi dengan perangkat lain pada **link** lokal yang sama. Dengan IPv6, **link** istilah mengacu pada subnet. LLAs terbatas pada satu **link**. Keunikan mereka hanya harus dikonfirmasi pada **link** itu karena mereka tidak routable di luar **link**. Dengan kata lain, router tidak akan meneruskan paket dengan alamat **Source** atau **Destination** link-local.

Catatan Tentang Unique Local Address

Unique Local Address (rentang fc00::/7 hingga fdff::/7) belum umum diimplementasikan. Oleh karena itu, materi ini hanya mencakup konfigurasi GUA dan LLA. Namun, **Unique Local Address** akhirnya dapat digunakan untuk mengatasi perangkat yang seharusnya tidak dapat diakses dari luar, seperti server internal dan printer.

Unique Local Address IPv6 memiliki beberapa kesamaan dengan **Private Address** RFC 1918 untuk IPv4, tetapi ada perbedaan yang signifikan:

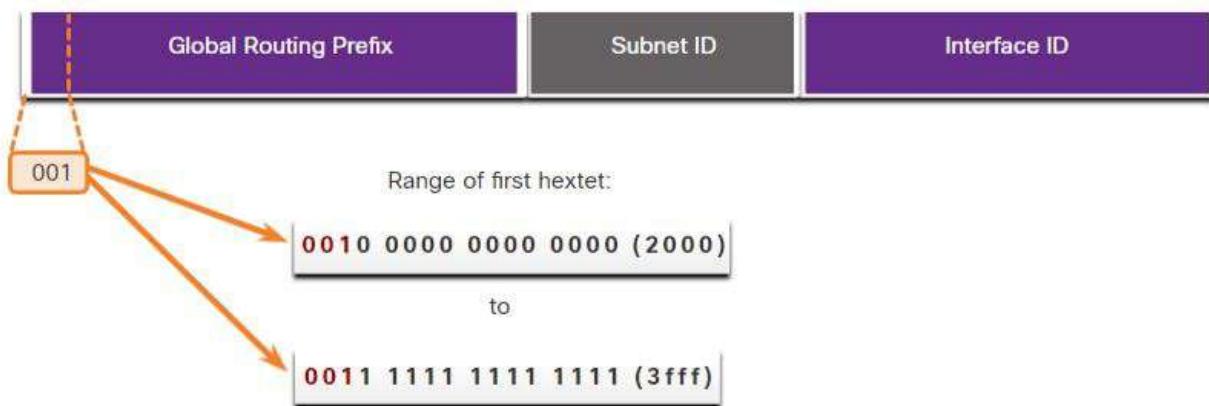
- **Unique Local Address** digunakan untuk alamat lokal dalam situs atau di antara sejumlah situs yang terbatas.
- **Unique Local Address** dapat digunakan untuk perangkat yang tidak perlu mengakses jaringan lain.
- **Unique Local Address** tidak **dirouting** secara global atau diterjemahkan ke alamat IPv6 global.

Catatan: Banyak situs juga menggunakan sifat pribadi alamat RFC 1918 untuk mencoba mengamankan atau menyembunyikan jaringan mereka dari potensi risiko keamanan. Namun, ini tidak pernah dimaksudkan penggunaan teknologi ini, dan IETF selalu merekomendasikan bahwa situs mengambil tindakan pencegahan keamanan yang tepat pada router yang menghadap internet mereka.

IPv6 GUA

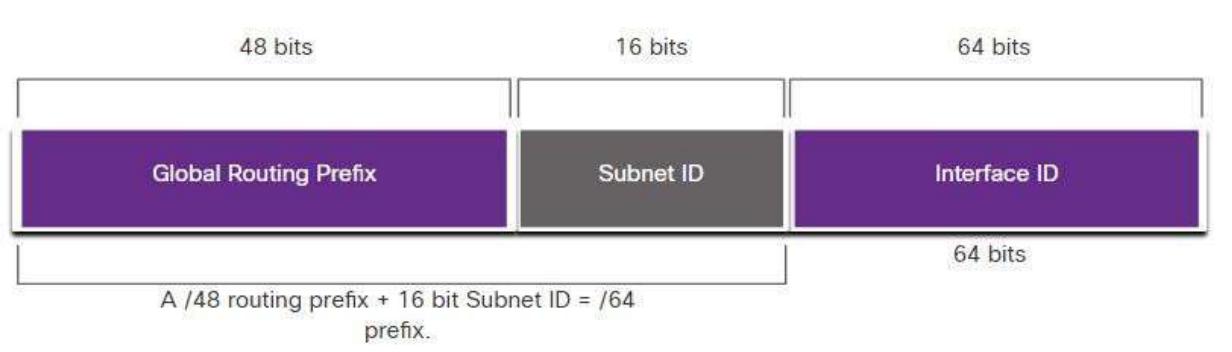
Global Unicast Address (GUAs) IPv6 secara global unik dan dapat routable di internet IPv6. Alamat ini setara dengan alamat IPv4 publik. **The Internet Committee for Assigned Names and Numbers**(ICANN), operator untuk IANA, mengalokasikan blok alamat IPv6 ke lima RIR. Saat ini, hanya **GUAs** dengan tiga bit pertama 001 atau 2000::/3 yang sedang ditetapkan, seperti yang ditunjukkan pada gambar.

Catatan: Alamat 2001:db8::/32 telah disediakan untuk **Destination** dokumentasi, termasuk penggunaan dalam contoh.



Gambar berikutnya menunjukkan struktur dan rentang GUA.

Alamat IPv6 dengan Global Routing Prefix/48 dan Prefix /64



GUA memiliki tiga bagian:

- **Global Routing Prefix**
- **Subnet ID**
- **Interface ID**

Struktur IPv6 GUA

Global Routing Prefix

Global Routing Prefix adalah prefix, atau jaringan, bagian dari alamat yang ditetapkan oleh penyedia, seperti ISP, kepada pelanggan atau situs. Misalnya, adalah umum bagi ISP untuk menetapkan **Global Routing Prefix/48** kepada pelanggannya. **Global Routing Prefix** biasanya akan bervariasi tergantung pada kebijakan ISP.

Angka sebelumnya menunjukkan GUA menggunakan **Global Routing Prefix/48**. /48 ini adalah **Global Routing Prefix** umum yang ditetapkan dan akan digunakan di sebagian besar contoh di seluruh **Materi** ini.

Misalnya, alamat IPv6 2001:db8:acad::/48 memiliki **Global Routing Prefix** yang menunjukkan bahwa **prefix** 48 bit pertama (3 **Hextet**) (2001:db8:acad) adalah bagaimana ISP mengetahui **prefix** ini (jaringan). Titik dua (::) mengikuti **prefix length** /48 berarti sisa alamat berisi semua 0s. Ukuran **Global Routing Prefix** menentukan ukuran **Subnet ID**.

Subnet ID

Subnet ID Field adalah area antara **Global Routing Prefix** dan **Interface ID**. Tidak seperti IPv4 di mana Anda harus meminjam bit dari **host portion** untuk membuat subnet, IPv6 dirancang dengan **Subnetting in Mind**. Subnet ID digunakan oleh organisasi untuk mengidentifikasi **Subnet** dalam situsnya. Semakin besar **Subnet ID**, semakin banyak subnet yang tersedia.

Catatan: Banyak organisasi menerima **Global Routing Prefix/32**. Menggunakan **prefix** /64 yang disarankan untuk membuat **Interface ID** 64-bit, meninggalkan **Subnet ID** 32 bit. Ini berarti organisasi dengan **Global Routing Prefix/32** dan SUBNET ID 32-bit akan memiliki 4,3 miliar subnet, masing-masing dengan 18 perangkat quintillion per subnet. Itu adalah subnet sebanyak ada alamat IPv4 publik!

Alamat IPv6 pada angka sebelumnya memiliki **Global Routing Prefix/48**, yang umum di antara banyak jaringan perusahaan. Ini membuatnya sangat mudah untuk memeriksa berbagai bagian alamat. Menggunakan **prefix length** /64 yang khas, empat **Hextet** pertama adalah untuk **network portion** alamat, dengan hextet keempat menunjukkan **Subnet ID**. Empat **Hextet** sisanya adalah untuk **Interface ID**.

Interface ID

Interface ID IPv6 setara dengan bagian host dari alamat IPv4. Istilah Interface ID digunakan karena satu host mungkin memiliki beberapa **interface**, masing-masing memiliki satu atau beberapa alamat IPv6. Angka tersebut menunjukkan contoh struktur IPv6 GUA. Sangat disarankan bahwa dalam banyak kasus subnet /64 harus digunakan, yang membuat **Interface ID** 64-bit. **Interface ID** 64-bit memungkinkan untuk 18 perangkat quintillion atau host per subnet.

Subnet atau **prefix** /64 (Global Routing Prefix + Subnet ID) menyisakan 64 bit untuk **Interface ID**. Disarankan untuk mengizinkan perangkat yang mendukung SLAAC membuat **Interface ID** 64-bit mereka sendiri. Ini juga membuat pengembangan rencana mengatasi IPv6 sederhana dan efektif.

Catatan: Tidak seperti IPv4, di IPv6, alamat host all-0s dan all-1s dapat ditetapkan ke perangkat. Alamat all-1s dapat digunakan karena **broadcast address** tidak digunakan dalam IPv6. Alamat all-0s juga dapat digunakan, tetapi dicadangkan sebagai alamat Anycast Subnet-Router, dan harus ditetapkan hanya untuk router.

IPv6 LLA

Local Link Address IPv6 (LLA) memungkinkan perangkat untuk berkomunikasi dengan perangkat lain yang diaktifkan IPv6 pada **link** yang sama dan hanya pada **link** tersebut (subnet). Paket dengan **Source** atau **Destination** LLA tidak dapat dialihkan di luar **link** asal paket.

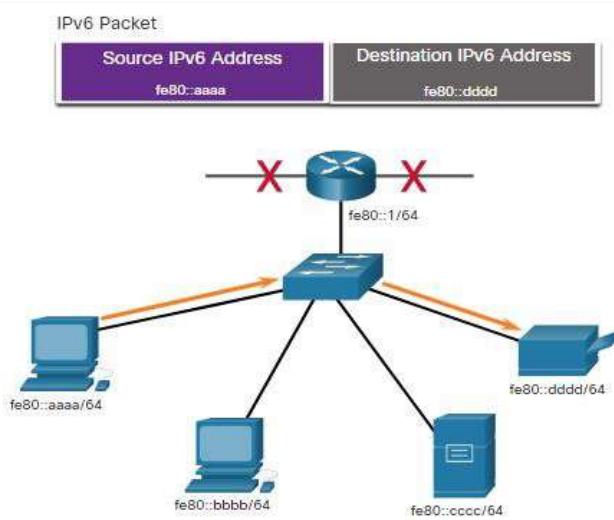
GUA bukan persyaratan. Namun, setiap **interface** jaringan berkemampuan IPv6 harus memiliki LLA.

Jika LLA tidak dikonfigurasi secara manual pada **interface**, perangkat akan secara otomatis membuat sendiri tanpa berkomunikasi dengan server DHCP. Host yang mendukung IPv6 membuat IPv6 LLA meskipun perangkat belum diberi alamat IPv6 unicast global. Ini memungkinkan perangkat berkemampuan IPv6 untuk berkomunikasi dengan perangkat lain yang mendukung IPv6 pada subnet yang sama. Ini termasuk komunikasi dengan gateway default (router).

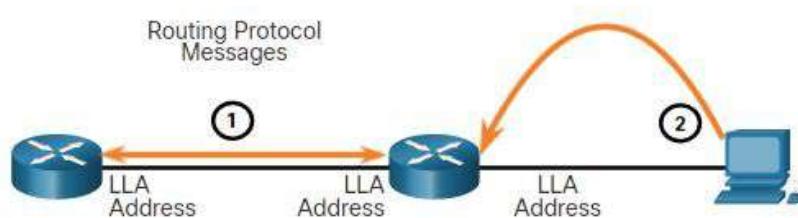
IPv6 LLAs berada di kisaran fe80::/10. /10 menunjukkan bahwa 10 bit pertama adalah 1111 1110 10xx xxxx. Hextet pertama memiliki kisaran 1111 1110 1000 0000 (fe80) hingga 1111 1110 1011 1111 (febf).

Angka tersebut menunjukkan contoh komunikasi menggunakan IPv6 LLAs. PC dapat berkomunikasi langsung dengan printer menggunakan LLAs.

Komunikasi IPv6 Link-Local



Gambar berikutnya menunjukkan beberapa kegunaan untuk IPv6 LLAs.



1. Routers use the LLA of neighbor routers to send routing updates.
2. Hosts use the LLA of a local router as the default-gateway.

Catatan: Biasanya, itu adalah LLA router, dan bukan GUA, yang digunakan sebagai gateway default untuk perangkat lain di link.

Ada dua cara agar perangkat dapat memperoleh LLA:

- **Statis** - Ini berarti perangkat telah dikonfigurasi secara manual.

- **Dinamis** – Ini berarti perangkat membuat **Interface ID** sendiri dengan menggunakan nilai yang dihasilkan secara acak atau menggunakan metode Extended Unique Identifier (EUI), yang menggunakan alamat MAC klien bersama dengan bit tambahan.

Konfigurasi Static GUA dan LLA

Seperti yang Anda pelajari di **Materi** sebelumnya, IPv6 GUAs sama dengan alamat IPv4 publik. Mereka unik secara global dan mudah routable di internet IPv6. IPv6 LLA memungkinkan dua perangkat berkemampuan IPv6 berkomunikasi satu sama lain pada **Link** yang sama (subnet). Sangat mudah untuk secara **Static** mengkonfigurasi IPv6 GUAs dan LLAs pada router untuk membantu Anda membuat jaringan IPv6. **Materi** ini mengajarkan Anda bagaimana melakukan hal itu!

Konfigurasi GUA Static pada Router

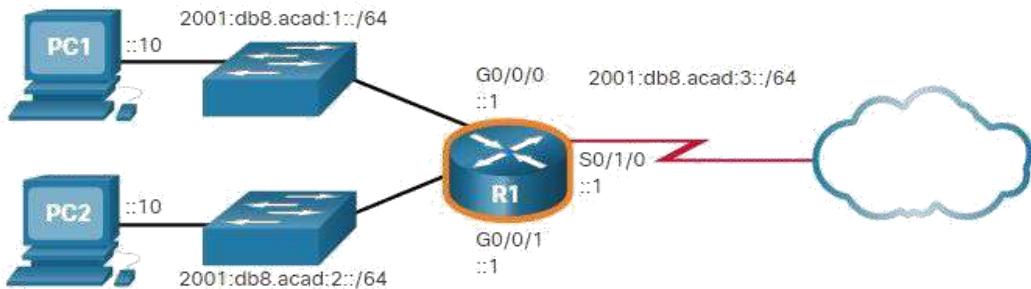
Sebagian besar perintah konfigurasi dan verifikasi IPv6 di IOS Cisco mirip dengan rekan-rekan IPv4 mereka. Dalam banyak kasus, satu-satunya perbedaan adalah **ip** ke **ipv6**

Misalnya, perintah Cisco IOS untuk mengkonfigurasi alamat IPv4 pada **interface** adalah **ip address ip-address subnet-mask**. Sebaliknya, perintah untuk mengkonfigurasi IPv6 GUA pada **interface** adalah **ipv6 address ipv6-address/prefix-length**.

Perhatikan bahwa tidak ada ruang *antara alamat IPV6 dan Prefix Length*.

Konfigurasi contoh menggunakan topologi yang ditunjukkan pada gambar dan subnet IPv6 ini:

- 2001:db8:acad:1:/64
- 2001:db8:acad:2:/64
- 2001:db8:acad:3:/64



Contoh Topologi

Contoh menunjukkan perintah yang diperlukan untuk mengkonfigurasi IPv6 GUA pada **R1 GigabitEthernet 0/0/0**, **GigabitEthernet 0/0/1**, dan **interface Serial 0/1/0**.

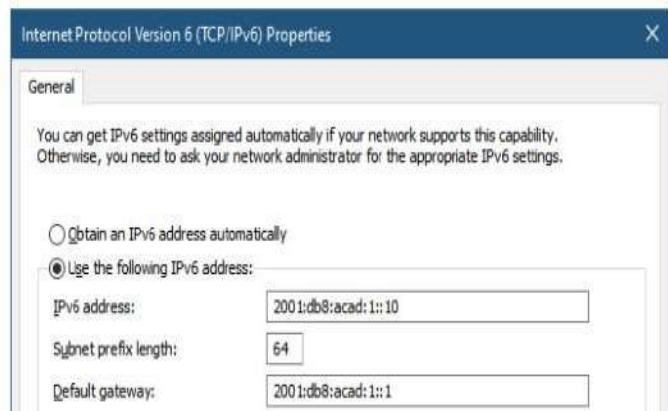
IPv6 GUA Configuration on Router R1

```
R1(config)# interface gigabitethernet 0/0/0
R1(config-if)# ipv6 address 2001:db8:acad:1::1/64
R1(config-if)# no shutdown
R1(config-if)# exit
R1(config)# interface gigabitethernet 0/0/1
R1(config-if)# ipv6 address 2001:db8:acad:2::1/64
R1(config-if)# no shutdown
R1(config-if)# exit
R1(config)# interface serial 0/1/0
R1(config-if)# ipv6 address 2001:db8:acad:3::1/64
R1(config-if)# no shutdown
```

Konfigurasi GUA Static pada Host Windows

Mengonfigurasi alamat IPv6 secara manual pada host mirip dengan mengkonfigurasi alamat IPv4.

Seperti yang ditunjukkan pada gambar, alamat **Default Gateway** yang dikonfigurasi untuk PC1 adalah 2001:db8:acad:1::1. Ini adalah GUA dari **interface R1 GigabitEthernet** pada jaringan yang sama. Atau, alamat **Default Gateway** dapat dikonfigurasi agar sesuai dengan LLA **interface GigabitEthernet**. Menggunakan LLA router sebagai alamat **Default Gateway** dianggap praktik terbaik. Salah satu konfigurasi akan berfungsi



Sama seperti IPv4, mengkonfigurasi alamat **Static** pada klien tidak memperbesar ke lingkungan yang lebih besar. Untuk alasan ini, sebagian besar administrator jaringan dalam jaringan IPv6 akan mengaktifkan **Dynamic Assignment IPv6 address**.

Ada dua cara di mana perangkat dapat memperoleh IPv6 GUA secara otomatis:

- Stateless Address Autoconfiguration (SLAAC)
- Stateful DHCPv6

SLAAC dan DHCPv6 dibahas dalam **Materi** berikutnya.

Catatan: Ketika DHCPv6 atau SLAAC digunakan, LLA router akan secara otomatis ditentukan sebagai alamat gateway default.

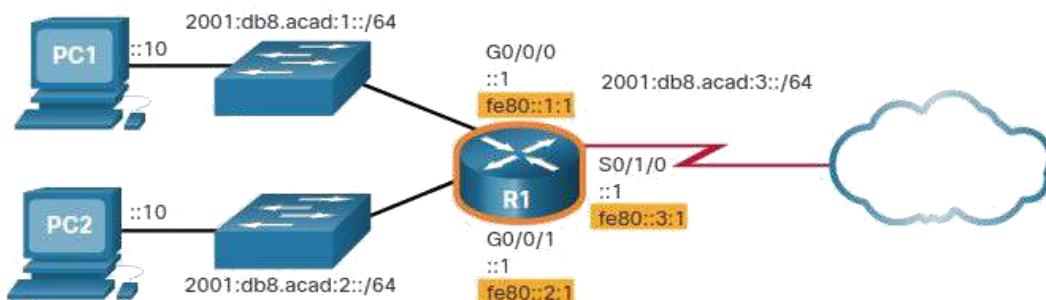
Konfigurasi Static Alamat Unicast Link-Local

Mengonfigurasi LLA secara manual memungkinkan Anda membuat alamat yang dapat dikenali dan lebih mudah diingat. Biasanya, hanya perlu membuat LLAs yang dapat dikenali pada router. Ini bermanfaat karena **ROUTER** digunakan sebagai alamat **Default Gateway** dan dalam **routing advertisement messages**.

LLA dapat dikonfigurasi secara manual menggunakan **ipv6 address ipv6-link-local-address link-local**. Ketika alamat dimulai dengan hextet ini dalam kisaran fe80 hingga febf, parameter **link-local** harus mengikuti alamat.

Gambar menunjukkan contoh topologi dengan IBI pada setiap **interface**.

Contoh Topologi dengan LLAs



Contoh konfigurasi LLA pada router R1.

```
R1(config)# interface gigabitethernet 0/0/0
R1(config-if)# ipv6 address fe80::1:1 link-local
R1(config-if)# exit
R1(config)# interface gigabitethernet 0/0/1
R1(config-if)# ipv6 address fe80::2:1 link-local
R1(config-if)# exit
R1(config)# interface serial 0/1/0
R1(config-if)# ipv6 address fe80::3:1 link-local
R1(config-if)# exit
```

LLA yang dikonfigurasi secara **Static** digunakan untuk membuatnya lebih mudah dikenali sebagai milik router R1. Dalam contoh ini, semua **interface** router R1 telah dikonfigurasi dengan LLA yang dimulai dengan **fe80::n:1**.

Catatan: LLA yang sama persis dapat dikonfigurasi pada setiap **Link** selama unik pada **Link** tersebut. Ini karena LLAs hanya harus unik pada **Link** itu. Namun, praktik umum adalah membuat LLA yang berbeda pada setiap **interface** router untuk membuatnya mudah untuk mengidentifikasi router dan **interface** tertentu.

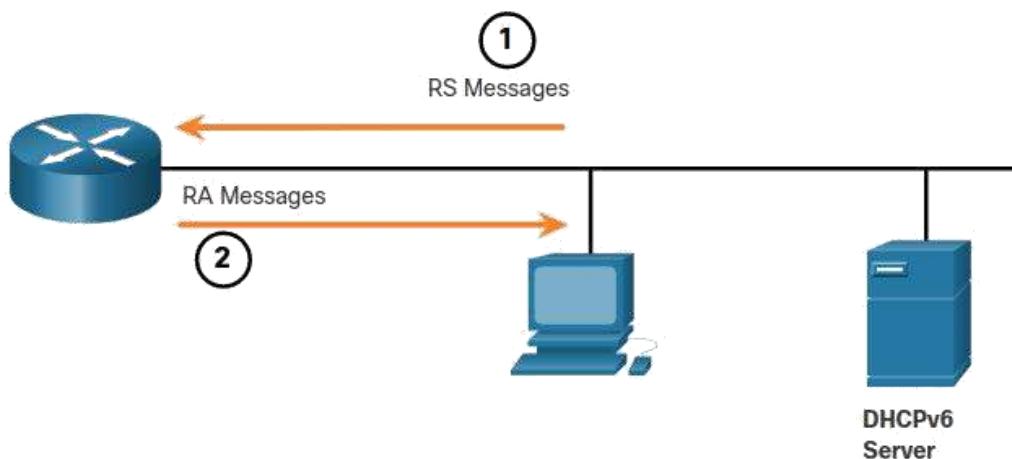
Alamat Dinamis untuk IPv6 GUAs

Jika Anda tidak ingin secara statis mengkonfigurasi IPv6 GUAs, tidak perlu khawatir. Sebagian besar perangkat mendapatkan IPv6 GUAs mereka secara dinamis. materi ini menjelaskan cara kerja proses ini menggunakan pesan **Router Advertisement (RA)** dan **Router Solicitation (RS)**. materi ini agak teknis, tetapi ketika Anda memahami perbedaan antara tiga metode yang dapat digunakan **Router Advertise**, serta bagaimana proses **EUI-64** untuk membuat **Interface ID** berbeda dari proses yang dihasilkan secara acak, Anda akan membuat lompatan besar dalam keahlian IPv6 Anda!

RS and RA Messages

Untuk GUA, perangkat memperoleh alamat secara dinamis melalui pesan **Internet Control Message Protocol versi 6 (ICMPv6)**. Router IPv6 secara berkala mengirimkan pesan ICMPv6 RA, setiap 200 detik, ke semua perangkat yang mendukung IPv6 di jaringan. Pesan RA juga akan dikirim sebagai tanggapan kepada **Host** yang mengirim pesan ICMPv6 RS, yang merupakan permintaan pesan RA. Kedua pesan ditampilkan dalam gambar.

Pesan ICMPv6 RS dan RA



1. Pesan RS dikirim ke semua router IPv6 oleh host yang meminta informasi alamat.
2. Pesan RA dikirim ke semua node IPv6. Jika Metode 1 (hanya SLAAC) digunakan, RA menyertakan **Network Prefix**, **Prefix Length**, dan **default gateway Information**.

Pesan RA ada di **Interface** Ethernet router IPv6. **Router** harus diaktifkan untuk **Routing** IPv6, yang tidak diaktifkan secara default. Untuk mengaktifkan **Router** sebagai router IPv6, perintah ***ipv6 unicast-routing*** harus digunakan.

Pesan ICMPv6 RA adalah saran untuk perangkat tentang cara mendapatkan IPv6 GUA. Keputusan akhir terserah **Operating System** perangkat. Pesan ICMPv6 RA mencakup yang berikut ini:

- **Network Prefix dan Prefix Length** – Ini memberi tahu perangkat tempat jaringan itu berada.
- **default gateway address** – Ini adalah IPv6 LLA, alamat sumber IPv6 dari pesan RA.
- **DNS Addresses dan domain Name** – Ini adalah alamat server DNS dan nama domain.

Ada tiga metode untuk pesan RA:

- **Metode 1: SLAAC** – “Saya memiliki semua yang Anda butuhkan termasuk **Prefix**, **Prefix Length**, dan **default gateway address**.”
- **Metode 2: SLAAC with a stateless DHCPv6 server** – “Berikut adalah informasi saya tetapi Anda perlu mendapatkan informasi lain seperti alamat DNS dari server **Stateless DHCPV6**”
- **Metode 3: Stateful DHCPv6 (no SLAAC)** – “Saya dapat memberi Anda alamat **default gateway** Anda. Anda perlu meminta server DHCPv6 yang nyata untuk semua informasi Anda yang lain.”

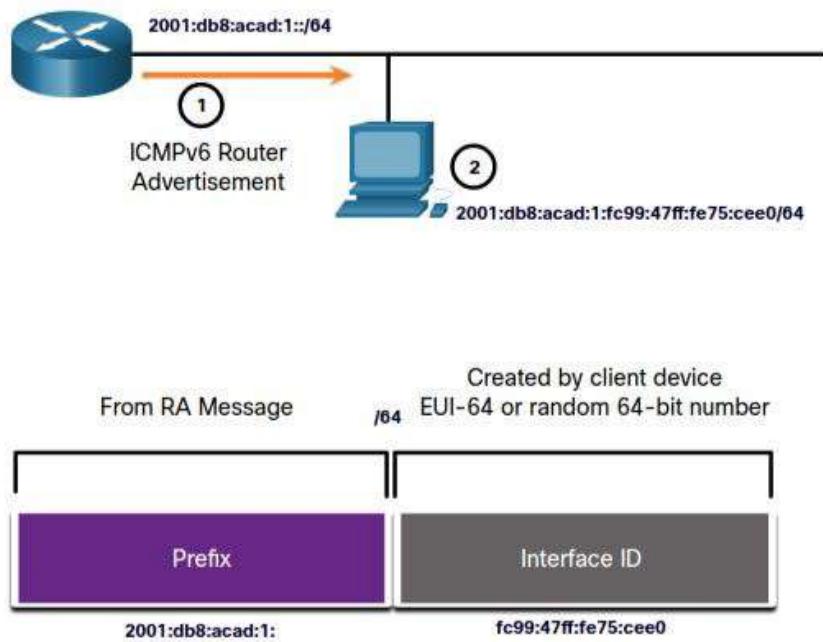
Metode 1: SLAAC

SLAAC adalah metode yang memungkinkan perangkat untuk membuat GUA sendiri tanpa layanan DHCPv6. Menggunakan SLAAC, perangkat mengandalkan pesan ICMPv6 RA dari router lokal untuk mendapatkan informasi yang diperlukan.

Secara default, pesan RA menunjukkan bahwa perangkat penerima menggunakan informasi dalam pesan RA untuk membuat IPv6 GUA sendiri dan semua informasi lain yang diperlukan. Layanan server DHCPv6 tidak diperlukan.

SLAAC is stateless, yang berarti tidak ada server pusat (misalnya, stateful DHCPv6 server) mengalokasikan GUAs dan menyimpan daftar perangkat dan alamatnya. Dengan SLAAC, perangkat klien menggunakan informasi dalam pesan RA untuk membuat GUA sendiri. Seperti yang ditunjukkan pada gambar, dua bagian alamat dibuat sebagai berikut:

- **Prefix** – Ini **di advertise** dalam pesan RA.
- **Interface ID** – Ini menggunakan proses EUI-64 atau dengan menghasilkan nomor acak 64-bit, tergantung pada sistem operasi perangkat.



1. Router mengirim pesan RA dengan **Prefix** untuk **Local Link**.
2. PC menggunakan SLAAC untuk mendapatkan **Prefix** dari pesan RA dan membuat **Interface ID** sendiri.

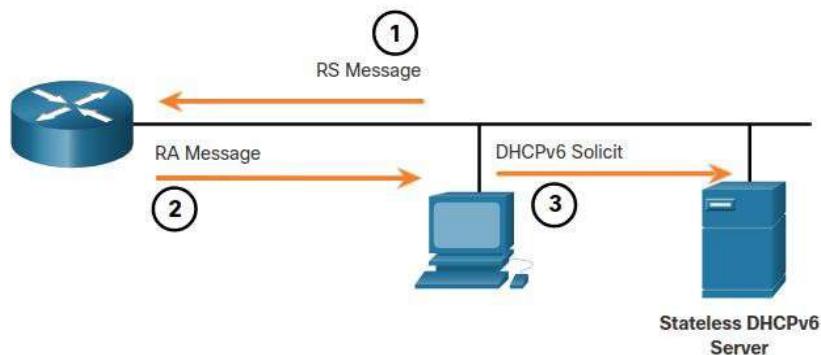
Metode 2: SLAAC dan Stateless DHCPV6

Interface router dapat dikonfigurasi untuk mengirim **Router Advertise** menggunakan SLAAC dan **Stateless DHCPv6**.

Seperti yang ditunjukkan pada gambar, dengan metode ini, pesan RA menyarankan perangkat menggunakan yang berikut:

- SLAAC untuk membuat IPv6 GUA sendiri
- Router LLA, yang merupakan alamat IPv6 sumber RA, sebagai alamat **Default gateway**
- Server **Stateless DHCPv6** untuk mendapatkan informasi lain seperti alamat server DNS dan nama domain

Catatan: Server **Stateless DHCPv6** mendistribusikan alamat server DNS dan nama domain. Ini tidak mengalokasikan GUAs.



1. PC mengirim RS ke semua router IPv6, “Saya perlu menangani informasi.”
2. Router mengirim pesan RA ke semua node IPv6 dengan Metode 2 (SLAAC dan DHCPv6) yang ditentukan. “Berikut adalah **Prefix**, prefiks-length, dan **Default gateway Information** Anda. Tetapi Anda harus mendapatkan informasi DNS dari server DHCPv6.”
3. PC mengirim pesan DHCPv6 Solicit ke semua server DHCPv6. “Saya menggunakan SLAAC untuk membuat alamat IPv6 saya dan mendapatkan alamat **Default gateway** saya, tetapi saya memerlukan informasi lain dari server **Stateless DHCPV6**.”

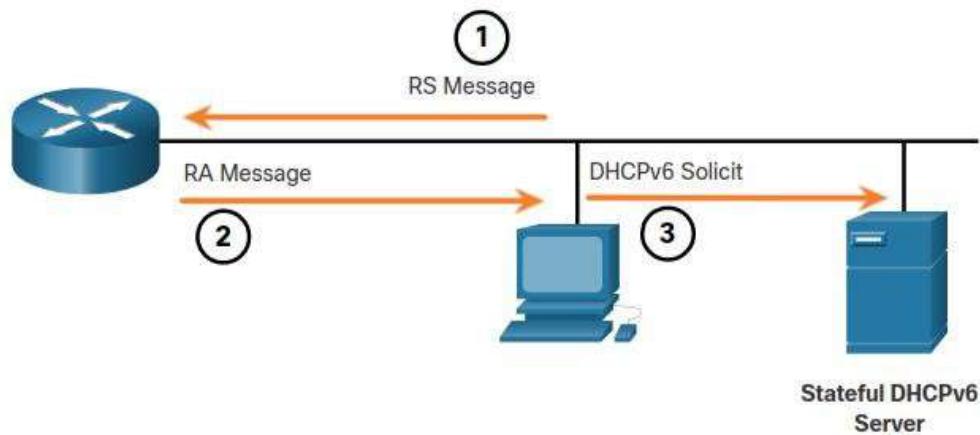
Metode 3: Stateful DHCPv6

Interface router dapat dikonfigurasi untuk mengirim RA menggunakan DHCPv6 yang stateful saja.

DHCPv6 stateful mirip dengan DHCP untuk IPv4. Perangkat dapat secara otomatis menerima informasi alamatnya termasuk GUA, **Prefix Length**, dan alamat server DNS dari server DHCPv6 yang stateful.

Seperti yang ditunjukkan pada gambar, dengan metode ini, pesan RA menyarankan perangkat menggunakan yang berikut:

- Router LLA, yang merupakan alamat IPv6 sumber RA, untuk alamat **Default gateway**.
- Server DHCPv6 yang stateful untuk mendapatkan GUA, alamat server DNS, nama domain, dan informasi lain yang diperlukan.



1. PC mengirim RS ke semua router IPv6, “Saya perlu menangani informasi.”
2. Router mengirim pesan RA ke semua node IPv6 dengan Metode 3 (DHCPv6 stateful) yang ditentukan, “Saya adalah **Default gateway** Anda, tetapi Anda perlu meminta server **DHCPv6 statefull** untuk alamat IPv6 Anda dan informasi alamat lainnya.”
3. PC mengirim pesan DHCPv6 Solicit ke semua server DHCPv6, “Saya menerima alamat **Default gateway** saya dari pesan RA, tetapi saya memerlukan alamat IPv6 dan semua informasi alamat lainnya dari server DHCPv6 statefull.”

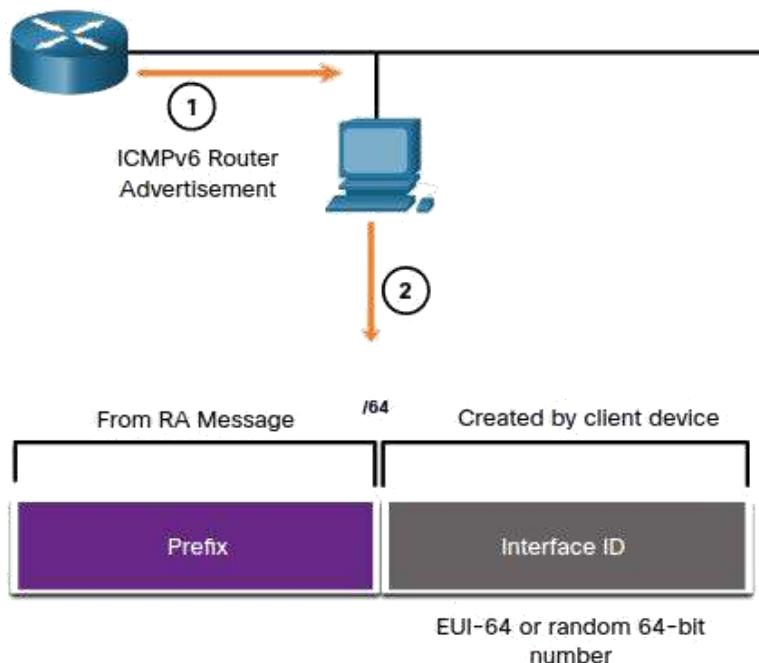
Server DHCPv6 yang stateful mengalokasikan dan memelihara daftar perangkat mana yang menerima alamat IPv6 mana. DHCP untuk IPv4 stateful.

Catatan: Alamat **Default gateway** hanya dapat diperoleh secara dinamis dari pesan RA. Server **Stateless atau Stateful DHCPV6** tidak menyediakan alamat **Default gateway**.

EUI-64 Process vs. Randomly Generated

Ketika pesan RA adalah SLAAC atau SLAAC dengan **Stateless DHCPV6**, klien harus menghasilkan **Interface ID**nya sendiri. Klien mengetahui bagian **Prefix** alamat dari pesan RA, tetapi harus membuat **Interface ID**nya sendiri. **Interface ID** dapat dibuat menggunakan proses EUI-64 atau angka 64-bit yang dihasilkan secara acak, seperti yang ditunjukkan pada gambar.

Membuat Interface ID secara Dinamis



1. **Router** mengirim pesan RA.
2. PC menggunakan **Prefix** dalam pesan RA dan menggunakan EUI-64 atau nomor 64-bit acak untuk menghasilkan **Interface ID**.

Proses EUI-64

IEEE mendefinisikan **Extended Unique Identifier (EUI)** atau proses EUI-64 yang dimodifikasi. Proses ini menggunakan alamat MAC Ethernet 48-bit klien, dan menyisipkan bit 16 lainnya di tengah alamat MAC 48-bit untuk membuat **Interface ID** 64-bit.

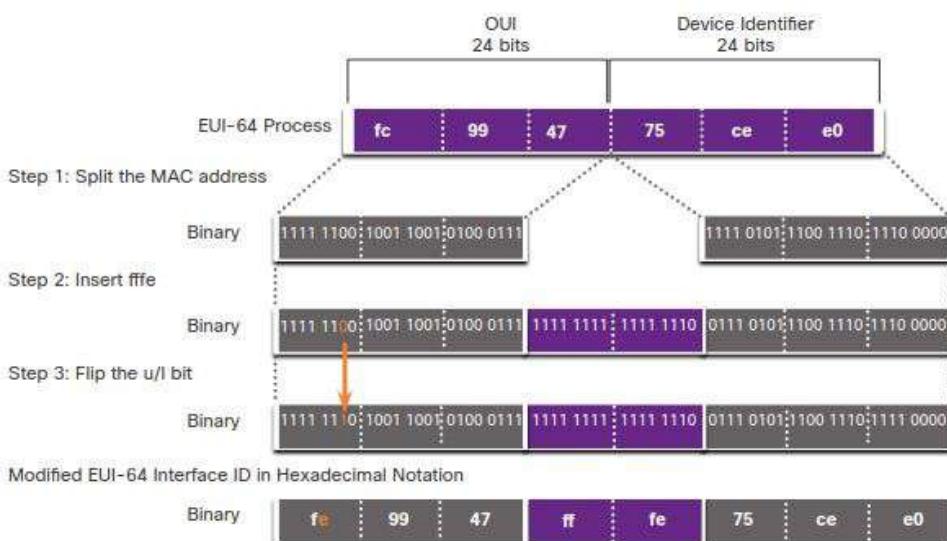
Alamat MAC Ethernet biasanya diwakili dalam heksadesimal dan terdiri dari dua bagian:

- **Organizationally Unique Identifier (OUI)** – kode vendor 24-bit (6 heksadesimal digit) yang ditetapkan oleh IEEE.
- **Device Identifier** – nilai 24-bit (6 digit heksadesimal) yang unik dalam OUI umum.

EUI-64 Interface ID diwakili dalam biner dan terdiri dari tiga bagian:

- 24-bit OUI dari alamat MAC klien, tetapi bit ke-7 (bit Universal/Lokal (U/L) dibalik. Ini berarti bahwa jika bit ke-7 adalah 0, itu menjadi 1, dan sebaliknya.
- Fffe nilai 16-bit yang disisipkan (dalam heksadesimal).
- Pengidentifikasi Perangkat 24-bit dari alamat MAC klien.

Proses EUI-64 diilustrasikan dalam gambar, menggunakan alamat MAC R1 GigabitEthernet fc99:4775:cee0.



Langkah 1: Bagi alamat MAC antara OUI dan pengidentifikasi perangkat.

Langkah 2: Masukkan fffe nilai heksadesimal, yang dalam biner adalah: 1111 1111 1111 1110.

Langkah 3: Konversikan 2 nilai heksadesimal pertama dari OUI ke biner dan balikkan bit U/L (bit 7). Dalam contoh ini, 0 di bit 7 diubah menjadi 1.

Hasilnya adalah **Interface ID** yang dihasilkan EUI-64 dari fe99:47ff:fe75:cee0.

Catatan: Penggunaan bit U / L, dan alasan untuk membalikkan nilainya, dibahas dalam RFC 5342.

Contoh output untuk perintah **ipconfig** menunjukkan IPv6 GUA dibuat secara dinamis menggunakan SLAAC dan proses EUI-64. Cara mudah untuk mengidentifikasi bahwa alamat mungkin dibuat menggunakan EUI-64 adalah **ffff** yang terletak di tengah **Interface ID**.

Keuntungan dari EUI-64 adalah bahwa alamat Mac Ethernet dapat digunakan untuk menentukan **Interface ID**. Ini juga memungkinkan administrator jaringan untuk dengan mudah melacak alamat IPv6 ke **End Devices** menggunakan alamat MAC yang unik. Namun, ini telah menyebabkan kekhawatiran privasi di antara banyak pengguna yang khawatir bahwa paket mereka dapat dilacak ke komputer fisik yang sebenarnya. Karena kekhawatiran ini, **Interface ID** yang dihasilkan secara acak dapat digunakan sebagai gantinya.

Interface ID buatan EUI-64

```
C:\> ipconfig
Windows IP Configuration
Ethernet adapter Local Area Connection:
  Connection-specific DNS Suffix . :
    IPv6 Address . . . . . : 2001:db8:acad:1:fc99:47ff:fe75:cee0
    Link-local IPv6 Address . . . . . : fe80::fc99:47ff:fe75:cee0
    Default Gateway . . . . . : fe80::1
C:\>
```

Interface ID yang Randomly Generated

Tergantung pada sistem operasi, perangkat dapat menggunakan **Interface ID** yang dihasilkan secara acak alih-alih menggunakan alamat MAC dan proses EUI-64. Dimulai dengan Windows Vista, Windows menggunakan **Interface ID** yang dihasilkan secara acak alih-alih yang dibuat dengan EUI-64. Windows XP dan sistem operasi Windows sebelumnya menggunakan EUI-64.

Setelah **Interface ID** ditetapkan, baik melalui proses EUI-64 atau melalui Randomly Generated, **Interface ID** dapat dikombinasikan dengan **Prefix IPv6** dalam pesan RA untuk membuat GUA, seperti yang ditunjukkan pada gambar.

Interface ID Acak yang Dihasilkan 64-bit

```
C:\> ipconfig
Windows IP Configuration
Ethernet adapter Local Area Connection:
  Connection-specific DNS Suffix . :
    IPv6 Address . . . . . : 2001:db8:acad:1:50a5:8a35:a5bb:66e1
    Link-local IPv6 Address . . . . . : fe80::50a5:8a35:a5bb:66e1
    Default Gateway . . . . . : fe80::1
C:\>
```

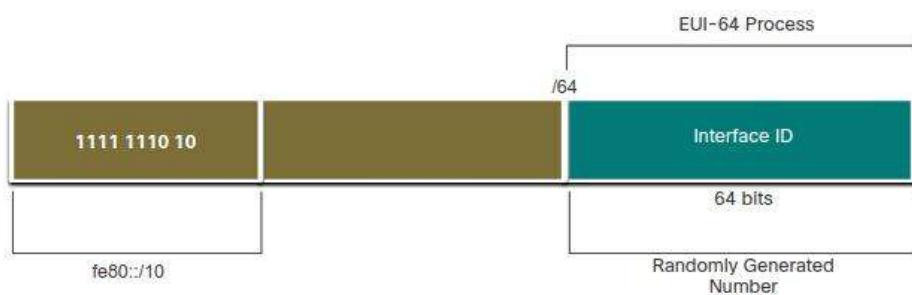
Catatan: Untuk memastikan keunikan alamat unicast IPv6, klien dapat menggunakan proses yang dikenal sebagai **Duplicate Address Detection (DAD)**. Ini mirip dengan permintaan ARP untuk alamatnya sendiri. Jika tidak ada balasan, maka alamatnya unik.

Alamat Dinamis untuk IPv6 LLAs

Semua perangkat IPv6 harus memiliki IPv6 **LLA**. Seperti IPv6 **GUAs**, Anda juga dapat membuat **LLAs** secara dinamis. Terlepas dari bagaimana Anda membuat **LLAs** (dan **GUAs** Anda), penting bagi Anda untuk memverifikasi semua konfigurasi alamat IPv6. materi ini menjelaskan VERIFIKASI konfigurasi IPv6 dan IPv6 yang dihasilkan secara dinamis.

Dynamic LLAs

Angka menunjukkan LLA dibuat secara dinamis menggunakan **Prefix fe80::/10** dan **Interface ID** menggunakan proses EUI-64, atau angka 64-bit yang dihasilkan secara acak.



LLA Dinamis di Windows

Operating System, seperti Windows, biasanya akan menggunakan metode yang sama untuk GUA yang dibuat SLAAC dan LLA yang ditetapkan secara dinamis. Lihat area yang disorot dalam contoh berikut yang diperlihatkan sebelumnya.

EUI-64 Generated Interface ID

```
C:\> ipconfig
Windows IP Configuration
Ethernet adapter Local Area Connection:
Connection-specific DNS Suffix . :
IPv6 Address . . . . . : 2001:db8:acad:1:fc99:47ff:fe75:cee0
Link-local IPv6 Address . . . . : fe80::fc99:47ff:fe75:cee0
Default Gateway . . . . . : fe80::1
C:\>
```

Random 64-bit Generated Interface ID

```
C:\> ipconfig
Windows IP Configuration
Ethernet adapter Local Area Connection:
  Connection-specific DNS Suffix . :
    IPv6 Address . . . . . : 2001:db8:acad:1:50a5:8a35:a5bb:66e1
    Link-local IPv6 Address . . . . . : fe80::50a5:8a35:a5bb:66e1
    Default Gateway . . . . . : fe80::1
C:\>
```

Dynamic LLAs pada Cisco Routers

Router Cisco secara otomatis membuat IPv6 LLA setiap kali GUA ditetapkan ke **Interface**. Secara default, router Cisco IOS menggunakan EUI-64 untuk menghasilkan **Interface ID** untuk semua LLAs pada **Interface** IPv6. Untuk **Interface** serial, router akan menggunakan alamat **MAC Interface Ethernet**. Ingat bahwa LLA harus unik hanya di **Link** atau **network** tersebut. Namun, kelemahan untuk menggunakan LLA yang ditetapkan secara dinamis adalah **Interface ID**nya yang panjang, yang membuatnya sulit untuk mengidentifikasi dan mengingat alamat yang ditetapkan. Contohnya menampilkan alamat MAC pada **Interface** GigabitEthernet 0/0/0 router R1. Alamat ini digunakan untuk membuat LLA secara dinamis pada **Interface** yang sama, dan juga untuk **Interface** Serial 0/1/0.

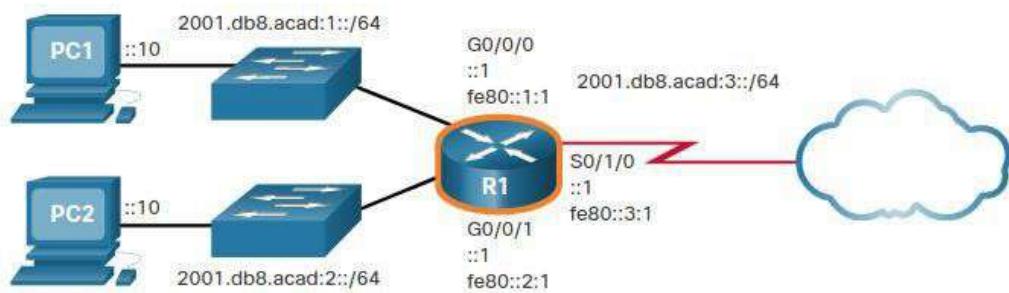
Untuk membuatnya lebih mudah dikenali dan mengingat alamat ini pada router, adalah umum untuk secara statis mengkonfigurasi IPv6 LLAs pada router.

IPv6 LLA Menggunakan EUI-64 pada Router R1

```
R1# show interface gigabitEthernet 0/0/0
GigabitEthernet0/0/0 is up, line protocol is up
  Hardware is ISR4221-2x1GE, address is 7079.b392.3640 (bia 7079.b392.3640)
  (Output omitted)
R1# show ipv6 interface brief
GigabitEthernet0/0/0      [up/up]
  FE80::7279:B3FF:FE92:3640
  2001:DB8:ACAD:1::1
GigabitEthernet0/0/1      [up/up]
  FE80::7279:B3FF:FE92:3641
  2001:DB8:ACAD:2::1
Serial0/1/0                [up/up]
  FE80::7279:B3FF:FE92:3640
  2001:DB8:ACAD:3::1
Serial0/1/1                [down/down]
  unassigned
R1#
```

Verifikasi Konfigurasi Alamat IPv6

Gambar menunjukkan contoh topologi.



show ipv6 interface brief

Perintah ***show ipv6 interface brief*** menampilkan alamat MAC **Interface** Ethernet. EUI-64 menggunakan alamat MAC ini untuk menghasilkan **Interface ID** untuk LLA. Selain itu, perintah ***show ipv6 interface brief*** menampilkan output yang disingkat untuk setiap **Interface**. Output [up/up] pada baris yang sama dengan **Interface** menunjukkan status **Interface** Layer 1/Layer 2. Ini sama dengan kolom Status dan Protokol dalam perintah IPv4 yang setara.

Perhatikan bahwa setiap **Interface** memiliki dua alamat IPv6. Alamat kedua untuk setiap **Interface** adalah GUA yang dikonfigurasi. Alamat pertama, yang dimulai dengan fe80, adalah alamat unicast link-local untuk **Interface**. Ingat bahwa LLA secara otomatis ditambahkan ke **Interface** ketika GUA ditetapkan.

Perhatikan juga bahwa R1 Serial 0/1/0 LLA sama dengan **Interface** GigabitEthernet 0/0/0. **Interface** serial tidak memiliki alamat Mac Ethernet, sehingga Cisco IOS menggunakan alamat MAC **Interface** Ethernet pertama yang tersedia. Ini dimungkinkan karena **Interface** link-local hanya harus unik pada **Link** itu.

Show IPv6 Route

Seperti yang ditunjukkan dalam contoh, ***Show IPv6 Route*** dapat digunakan untuk memverifikasi bahwa jaringan IPv6 dan **interface IPv6 address** tertentu telah diinstal dalam **routing table** IPv6. Perintah ***Show IPv6 Route*** akan menampilkan jaringan IPv6, bukan jaringan IPv4.

Dalam **Routing Table**, C di samping rute menunjukkan bahwa ini adalah jaringan yang **Directly Connected**. Ketika **Interface router** dikonfigurasi dengan GUA dan berada dalam keadaan “up/up”, Prefix IPv6 dan **prefix lenght** ditambahkan ke **Routing table IPv6** sebagai rute yang terhubung.

Catatan: L menunjukkan rute Lokal, alamat IPv6 tertentu yang ditetapkan ke **Interface**. Ini bukan LLA. LLAs tidak disertakan dalam tabel perutean router karena mereka bukan alamat yang dapat dirutekan.

IPv6 GUA yang dikonfigurasi pada **Interface** juga dipasang di **routing table** sebagai **local route**. **local route** memiliki prefix /128. **local route** digunakan oleh **routing table** untuk memproses paket secara efisien dengan alamat tujuan alamat **interface router**.

```
R1# show ipv6 route
IPv6 Routing Table - default - 7 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route

C  2001:DB8:ACAD:1::/64 [0/0]
    via GigabitEthernet0/0/0, directly connected
L  2001:DB8:ACAD:1::1/128 [0/0]
    via GigabitEthernet0/0/0, receive
C  2001:DB8:ACAD:2::/64 [0/0]
    via GigabitEthernet0/0/1, directly connected
L  2001:DB8:ACAD:2::1/128 [0/0]
    via GigabitEthernet0/0/1, receive
C  2001:DB8:ACAD:3::/64 [0/0]
    via Serial0/1/0, directly connected
L  2001:DB8:ACAD:3::1/128 [0/0]
    via Serial0/1/0, receive
L  FF00::/8 [0/0]
    via Null0, receive
R1#
```

ping

Perintah **ping** untuk IPv6 identik dengan perintah yang digunakan dengan IPv4, kecuali bahwa alamat IPv6 digunakan. Seperti yang ditunjukkan dalam contoh, perintah digunakan untuk memverifikasi koneksi Layer 3 antara R1 dan PC1. Saat melakukan ping LLA dari router, Cisco IOS akan meminta pengguna untuk **out interface**. Karena tujuan LLA dapat berada di satu atau lebih **link** atau jaringannya, **router** perlu mengetahui **interface** mana untuk mengirim ping balik.

```
R1# ping 2001:db8:acad:1::10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:ACAD:1::10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
R1#
```

IPv6 Multicast Address

Sebelumnya dalam materi ini, Anda mengetahui bahwa ada tiga kategori alamat IPv6 yang luas: **unicast**, **anycast**, dan **multicast**. Materi ini membahas lebih detail tentang alamat multicast.

Alamat Multicast IPv6 yang Ditetapkan

Alamat multicast IPv6 mirip dengan alamat multicast IPv4. Ingat bahwa alamat multicast digunakan untuk mengirim satu paket ke satu tujuan atau lebih (grup multicast). Alamat multicast IPv6 memiliki **prefix ff00::/8**.

Catatan: Alamat multicast hanya dapat menjadi **destination address** dan bukan **source address**.

Ada dua jenis alamat multicast IPv6:

- **Well-known multicast addresses**
- **Solicited node multicast addresses**

Well-Known IPv6 Multicast Addresses

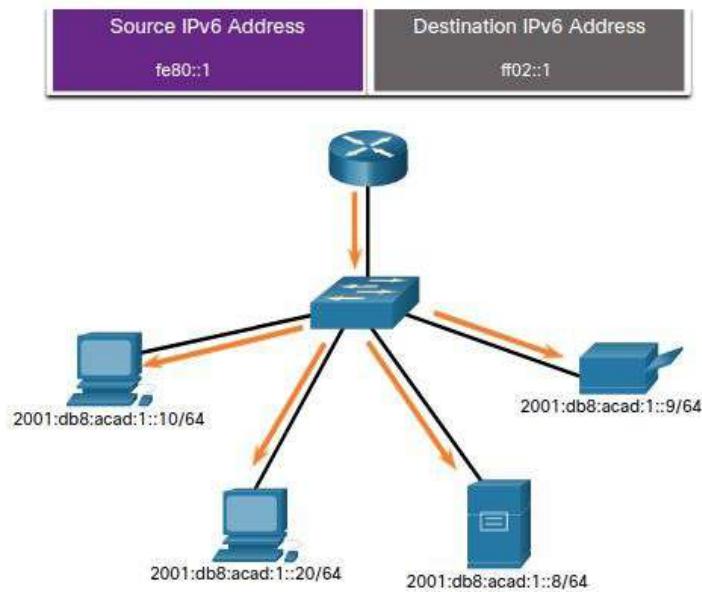
Well-known IPv6 multicast addresses are assigned. **Assigned multicast addresses** adalah alamat multicast yang dipesan untuk grup perangkat yang telah ditentukan sebelumnya. Sebuah **Assigned multicast addresses** adalah alamat tunggal yang digunakan untuk menjangkau sekelompok perangkat yang menjalankan protokol atau layanan umum. Alamat multicast yang ditetapkan digunakan dalam konteks dengan protokol tertentu seperti DHCPv6.

Ini adalah dua grup multicast umum yang ditetapkan IPv6:

- **ff02::1 All-nodes multicast group** – Ini adalah grup multicast yang bergabung dengan semua perangkat berkemampuan IPv6. Paket yang dikirim ke grup ini diterima dan diproses oleh semua **Interface IPv6** pada **Link** atau **Network**. Ini memiliki efek yang sama dengan **Broadcast Address** di IPv4. Gambar menunjukkan contoh komunikasi menggunakan alamat multicast semua node. Router IPv6 mengirim pesan ICMPv6 RA ke grup multicast all-node.
- **ff02::2 All-routers multicast group** – Ini adalah grup multicast yang semua router IPv6 yang bergabung. Router menjadi anggota grup ini ketika diaktifkan sebagai router IPv6 dengan perintah **ipv6 unicast-routing**. Paket yang dikirim ke grup ini diterima dan diproses oleh semua router IPv6 pada **link** atau **network**.

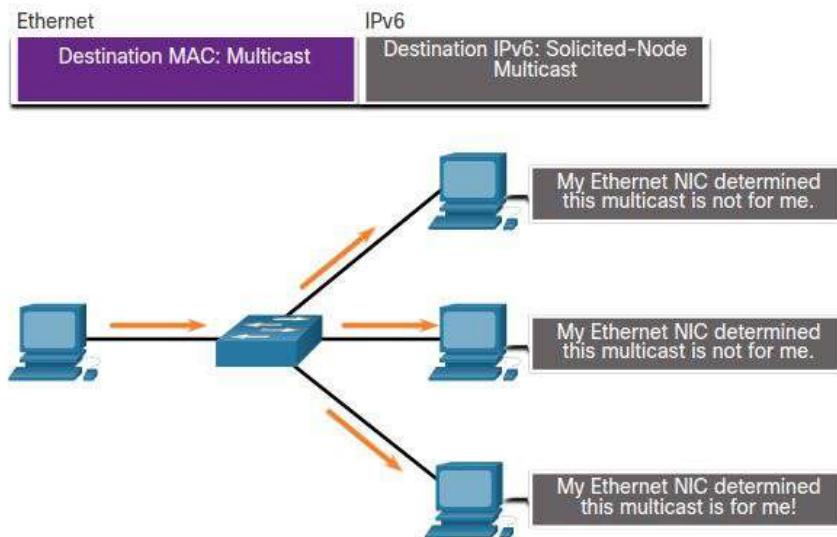
IPv6 All-Nodes Multicast: Pesan RA

Perangkat berkemampuan IPv6 mengirim pesan ICMPv6 RS ke alamat multicast all-routers. Pesan RS meminta pesan RA dari router IPv6 untuk membantu perangkat dalam konfigurasi alamatnya. Router IPv6 merespons dengan pesan RA, seperti yang ditunjukkan.



Solicited-Node IPv6 Multicast Addresses

Alamat multicast node yang diminta mirip dengan alamat multicast all-nodes. Keuntungan dari alamat multicast node yang diminta adalah bahwa itu dipetakan ke alamat multicast Ethernet khusus. Hal ini memungkinkan Ethernet NIC untuk memfilter **Frame** dengan memeriksa **Destination MAC Address** tanpa mengirimkannya ke proses IPv6 untuk melihat apakah perangkat adalah target yang dimaksudkan dari paket IPv6.



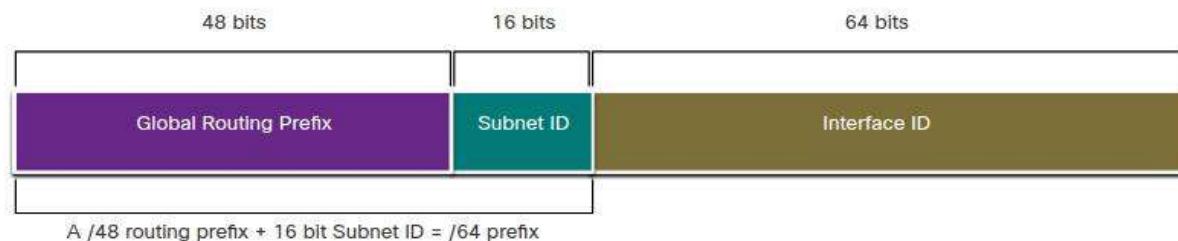
Subnet pada IPv6

Pengenalan materi ini disebutkan mensubnetting jaringan IPv6. Ini juga mengatakan bahwa Anda mungkin menemukan bahwa itu sedikit lebih mudah daripada mensubnetting jaringan IPv4. Anda akan mencari tahu!

Subnet Menggunakan Subnet ID

Ingat bahwa dengan IPv4, kita harus meminjam bit dari **Host Portion** untuk membuat subnet. Ini karena subnetting adalah sebuah **afterthought** dengan IPv4. Namun, IPv6 dirancang dengan **subnetting in mind**. **Field Subnet ID** terpisah di IPv6 GUA digunakan untuk membuat subnet. Seperti yang ditunjukkan pada gambar, **Field Subnet ID** adalah area antara Global Routing Prefix dan **Interface ID**.

GUA dengan Subnet ID 16-bit



Manfaat alamat 128-bit adalah dapat mendukung lebih dari cukup subnet dan host per subnet, untuk setiap jaringan. **Address conservation** bukanlah masalah. Misalnya, jika **Global Routing Prefix** adalah /48, dan menggunakan bit 64 khas untuk **Interface ID**, ini akan membuat **Subnet ID** 16-bit:

- **Subnet ID 16-bit** – Membuat hingga 65.536 subnet.
- **Interface ID 64-bit** – Mendukung hingga 18 alamat IPv6 host quintillion per subnet (yaitu, 18.000.000.000.000.000.000.000.000).

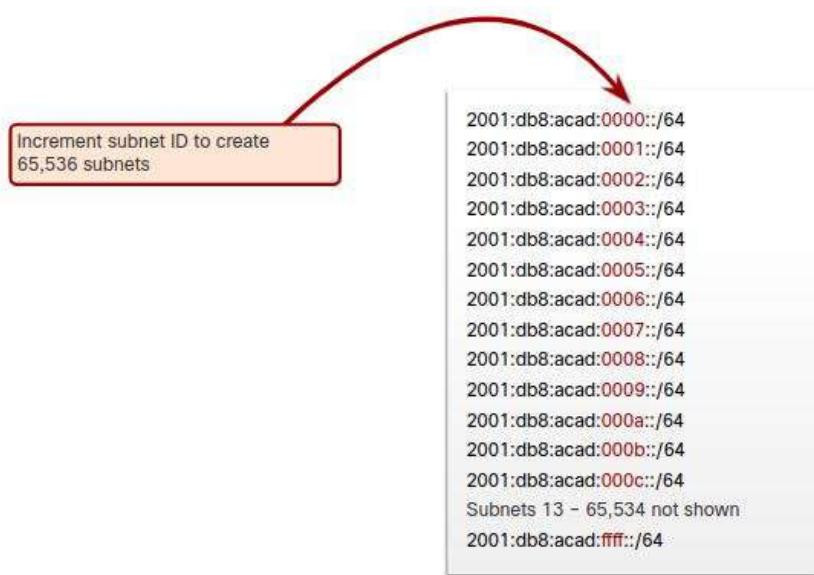
Catatan: **Subnetting Interface ID** 64-bit (atau **Host Portion**) juga dimungkinkan tetapi jarang diperlukan.

Subnetting IPv6 juga lebih mudah diterapkan daripada IPv4, karena tidak ada konversi ke biner yang diperlukan. Untuk menentukan subnet berikutnya yang tersedia, cukup hitung dalam heksadesimal.

Contoh Subnetting IPv6

Misalnya, asumsikan organisasi telah menetapkan **Global Routing Prefix** 2001:db8:acad::/48 dengan **Subnet ID** 16 bit. Ini akan memungkinkan organisasi untuk membuat subnet 65.536 /64, seperti yang ditunjukkan pada gambar. Perhatikan bagaimana **Global Routing Prefix** sama untuk semua subnet. Hanya hektet **Subnet ID** yang ditambah dalam heksadesimal untuk setiap subnet.

Subnetting menggunakan Subnet ID 16-bit

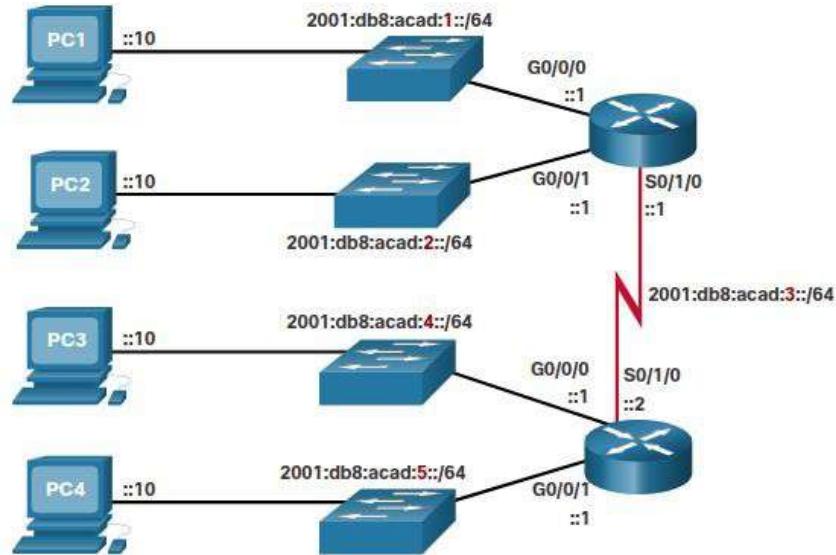


Alokasi Subnet IPv6

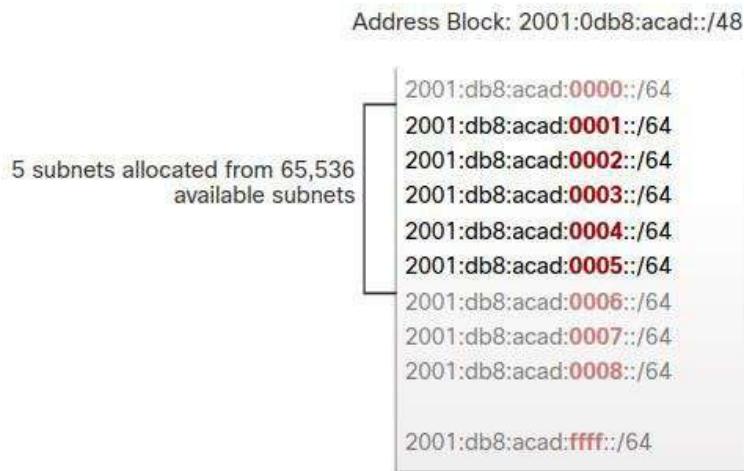
Dengan lebih dari 65.536 subnet untuk dipilih, tugas administrator jaringan menjadi salah satu merancang skema logis untuk mengatasi jaringan.

Seperti yang ditunjukkan pada gambar, contoh topologi membutuhkan lima subnet, satu untuk setiap LAN serta untuk **link serial** antara R1 dan R2. Tidak seperti contoh untuk IPv4, dengan IPv6 subnet link serial akan memiliki panjang **Prefix** yang sama dengan LAN. Meskipun ini mungkin tampak “membuang” alamat, konservasi alamat tidak menjadi perhatian ketika menggunakan IPv6.

Contoh Topologi



Seperti yang ditunjukkan pada gambar berikutnya, lima subnet IPv6 dialokasikan, dengan **Field Subnet ID** 0001 hingga 0005 digunakan untuk contoh ini. Setiap subnet /64 akan menyediakan lebih banyak alamat daripada yang pernah diperlukan.



Router Dikonfigurasi dengan Subnet IPv6

Mirip dengan mengkonfigurasi IPv4, contoh menunjukkan bahwa setiap **Interface** router telah dikonfigurasi untuk berada di subnet IPv6 yang berbeda.

Konfigurasi Alamat IPv6 pada Router R1

```
R1(config)# interface gigabitethernet 0/0/0
R1(config-if)# ipv6 address 2001:db8:acad:1::1/64
R1(config-if)# no shutdown
R1(config-if)# exit
R1(config)# interface gigabitethernet 0/0/1
R1(config-if)# ipv6 address 2001:db8:acad:2::1/64
R1(config-if)# no shutdown
R1(config-if)# exit
R1(config)# interface serial 0/1/0
R1(config-if)# ipv6 address 2001:db8:acad:3::1/64
R1(config-if)# no shutdown
```

BAB 13

~ የጥርክም ~

Judul Bab : ICMP

Tujuan Bab : Menggunakan alat yang bervariasi untuk menguji koneksi jaringan.

Link Test Pemahaman : <https://s.id/QxVk>

| Judul Materi | Tujuan Materi |
|-----------------------------|--|
| Pesan ICMP | Menjelaskan bagaimana ICMP digunakan untuk menguji koneksi jaringan |
| Menguji Ping dan traceroute | Menggunakan fasilitas ping dan traceroute untuk menguji koneksi jaringan |

Pesan ICMP

Dalam materi ini, Anda akan mempelajari tentang berbagai jenis **Internet Control Message Protocols (ICMP)**, dan **tools** yang digunakan untuk mengirimnya.

Pesan ICMPv4 dan ICMPv6

Meskipun IP hanya protokol upaya terbaik, rangkaian **TCP/IP** menyediakan pesan kesalahan dan pesan informasi saat berkomunikasi dengan perangkat IP lain. Pesan-pesan ini dikirim menggunakan layanan ICMP. Tujuan pesan-pesan ini adalah untuk memberikan umpan balik tentang masalah yang terkait dengan pemrosesan paket IP dalam kondisi tertentu, bukan untuk membuat IP dapat diandalkan. Pesan ICMP tidak diperlukan dan seringkali tidak diizinkan dalam jaringan karena alasan keamanan.

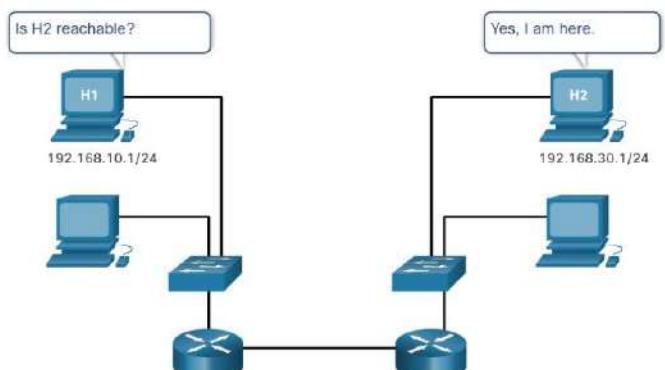
ICMP tersedia untuk IPv4 dan IPv6. ICMPv4 adalah protokol perpesanan untuk IPv4. ICMPv6 menyediakan layanan yang sama ini untuk IPv6 tetapi mencakup fungsionalitas tambahan. Dalam kursus ini, istilah ICMP akan digunakan jika mengacu pada ICMPv4 dan ICMPv6.

Jenis pesan ICMP, dan alasan mengapa pesan dikirim, sangat luas. Pesan ICMP yang umum untuk ICMPv4 dan ICMPv6 dan dibahas dalam materi ini meliputi:

- **Host Reachability**
- **Destination or Service Unreachable**
- **Time Exceeded**

Host Reachability

ICMP Echo Message dapat digunakan untuk menguji **Host Reachability** di jaringan IP. **Host** setempat mengirimkan **Echo Request** ICMP ke **Host**. Jika host tersedia, host tujuan merespons dengan Balasan Echo. Pada gambar, klik tombol Putar untuk melihat animasi ICMP Echo Request/Echo Reply. Penggunaan pesan ICMP Echo ini adalah dasar dari utilitas **ping**.



Destination or Service Unreachable

Ketika host atau gateway menerima paket yang tidak dapat dikirim, host atau gateway dapat menggunakan pesan Tujuan ICMP Yang tidak dapat dijangkau untuk memberi tahu sumber bahwa tujuan atau layanan tidak dapat dijangkau. Pesan akan menyertakan kode yang menunjukkan mengapa paket tidak dapat dikirim.

Beberapa kode **Destination Unreachable** untuk ICMPv4 adalah sebagai berikut:

- 0 – Net unreachable
- 1 – Host unreachable
- 2 – Protocol unreachable
- 3 – Port unreachable

Beberapa kode **Destination Unreachable** untuk ICMPv6 adalah sebagai berikut:

- 0 – No route to destination
- 1 – Communication with the destination is administratively prohibited (e.g., firewall)
- 2 – Beyond scope of the source address
- 3 – Address unreachable
- 4 – Port unreachable

Catatan: ICMPv6 memiliki kode yang mirip tetapi sedikit berbeda untuk pesan **Destination Unreachable**.

Time Exceeded

Pesan ICMPv4 Time Exceeded digunakan oleh router untuk menunjukkan bahwa paket tidak dapat diteruskan karena **Field** Time to Live (TTL) paket **decremented** ke 0. Jika router menerima paket dan **decremented Field** TTL dalam paket IPv4 ke nol, router akan membuang paket dan mengirim pesan Time Exceeded ke host sumber.

ICMPv6 juga mengirimkan pesan Time Exceeded jika router tidak dapat meneruskan paket IPv6 karena paket telah kedaluwarsa. Alih-alih **Field** IPv4 TTL, ICMPv6 menggunakan **Field** IPv6 Hop Limit untuk menentukan apakah paket telah kedaluwarsa.

Catatan: Pesan yang melebihi waktu digunakan oleh **traceroute**.

Pesan ICMPv6

Pesan informasi dan kesalahan yang ditemukan di ICMPv6 sangat mirip dengan pesan kontrol dan kesalahan yang diterapkan oleh ICMPv4. Namun, ICMPv6 memiliki fitur baru dan fungsionalitas yang ditingkatkan yang tidak ditemukan di ICMPv4. Pesan ICMPv6 dienkapsulasi di IPv6.

ICMPv6 menyertakan empat protokol baru sebagai bagian dari **Neighbor Discovery Protocol** (ND atau NDP).

Pesan antara router IPv6 dan perangkat IPv6, termasuk alokasi alamat dinamis adalah sebagai berikut:

- Router Solicitation (RS) message
- Router Advertisement (RA) message

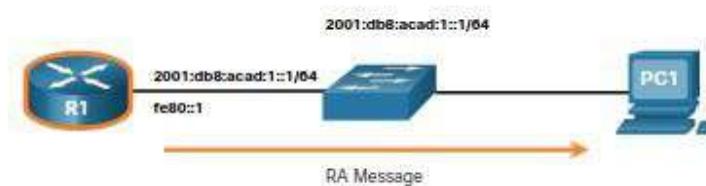
Pesan antar perangkat IPv6, termasuk deteksi alamat duplikat dan resolusi alamat adalah sebagai berikut:

- Neighbor Solicitation (NS) message
- Neighbor Advertisement (NA) message

Catatan: ICMPv6 ND juga menyertakan pesan **redirect**, yang memiliki fungsi serupa dengan pesan **redirect** yang digunakan di ICMPv4.

RA Message

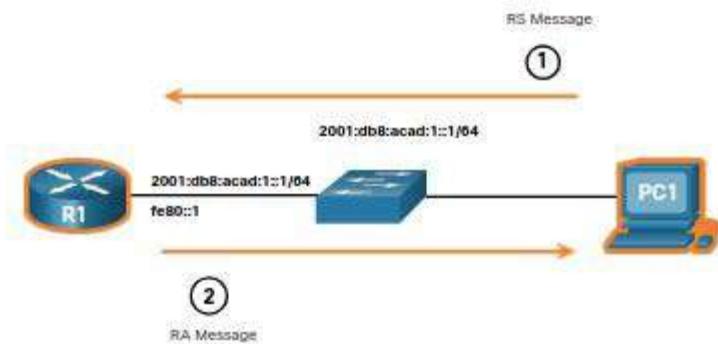
Pesan RA dikirim oleh router berkemampuan IPv6 setiap 200 detik untuk memberikan informasi alamat ke host yang mendukung IPv6. Pesan RA dapat mencakup informasi alamat untuk host seperti **prefix**, **prefix length**, **DNS Address**, dan **domain name**. Host yang menggunakan **Stateless Address Autoconfiguration (SLAAC)** akan mengatur **default gateway** ke alamat link-local router yang mengirim RA.



R1 mengirim pesan RA, "Hai semua perangkat yang mendukung IPv6. Saya R1 dan Anda dapat menggunakan SLAAC untuk membuat alamat unicast global IPv6. **Prefix** adalah 2001:db8:acad:1::/64. By the way, gunakan alamat link-local saya fe80::1 sebagai **default gateway** Anda."

RS Message

Router berkemampuan IPv6 juga akan mengirimkan pesan RA sebagai respons terhadap pesan RS. Dalam gambar, PC1 mengirim pesan RS untuk menentukan cara menerima informasi alamat IPv6 secara dinamis.



R1 membalas RS dengan pesan RA.

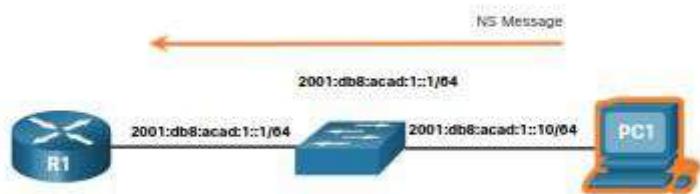
1. PC1 mengirim pesan RS, "Hai, saya baru saja boot up. Apakah ada router IPv6 pada jaringan? Saya perlu tahu cara mendapatkan informasi alamat IPv6 saya secara dinamis."
2. R1 membalas dengan pesan RA. "Hai semua perangkat berkemampuan IPv6. Saya R1 dan Anda dapat menggunakan SLAAC untuk membuat alamat unicast global IPv6. **Prefix** adalah 2001:db8:acad:1::/64. By the way, gunakan alamat link-local saya fe80::1 sebagai **Default Gateway** Anda."

NS Message

Ketika perangkat diberi alamat unicast IPv6 global atau unicast **link local**, perangkat dapat melakukan **duplicate address detection (DAD)** untuk memastikan bahwa alamat IPv6 unik. Untuk memeriksa keunikan alamat, perangkat akan mengirim pesan NS dengan alamat IPv6-nya sendiri sebagai alamat IPv6 yang ditargetkan, seperti yang ditunjukkan pada gambar.

Jika perangkat lain pada jaringan memiliki alamat ini, ia akan merespons dengan pesan NA. Pesan NA ini akan memberi tahu perangkat pengirim bahwa alamat sedang digunakan. Jika pesan NA yang sesuai tidak dikembalikan dalam waktu tertentu, alamat unicast unik dan dapat diterima untuk digunakan.

Catatan: DAD tidak diperlukan, tetapi RFC 4861 merekomendasikan agar DAD dilakukan pada alamat unicast.



Test Ping dan Traceroute

Dalam materi sebelumnya, Anda diperkenalkan ke alat **ping** dan traceroute(**tracert**). Dalam materi ini, Anda akan belajar tentang situasi di mana setiap alat digunakan, dan cara menggunakannya. Ping adalah utilitas pengujian IPv4 dan IPv6 yang menggunakan ICMP Echo Request dan pesan Echo Reply untuk menguji konektivitas antara host.

Ping – Uji Konektivitas

Untuk menguji konektivitas ke host lain di jaringan, **Echo Request** dikirim ke alamat host menggunakan perintah **ping**. Jika host di alamat yang ditentukan menerima **Echo Request**, itu merespons dengan **Echo Reply**. Karena setiap **Echo Reply** diterima, ping memberikan umpan balik tentang waktu antara kapan permintaan dikirim dan kapan balasan diterima. Ini bisa menjadi ukuran kinerja jaringan.

Ping memiliki nilai batas waktu untuk balasan. Jika balasan tidak diterima dalam waktu habis, ping menyediakan pesan yang menunjukkan bahwa respons tidak diterima. Ini mungkin menunjukkan bahwa ada masalah, tetapi juga dapat menunjukkan bahwa fitur keamanan memblokir pesan ping telah diaktifkan di jaringan. Adalah umum untuk ping pertama ke timeout jika resolusi alamat (ARP atau ND) perlu dilakukan sebelum mengirim **Echo Request** ICMP.

Setelah semua permintaan dikirim, utilitas **ping** memberikan ringkasan yang mencakup tingkat keberhasilan dan **round-trip time** rata-rata ke tujuan.

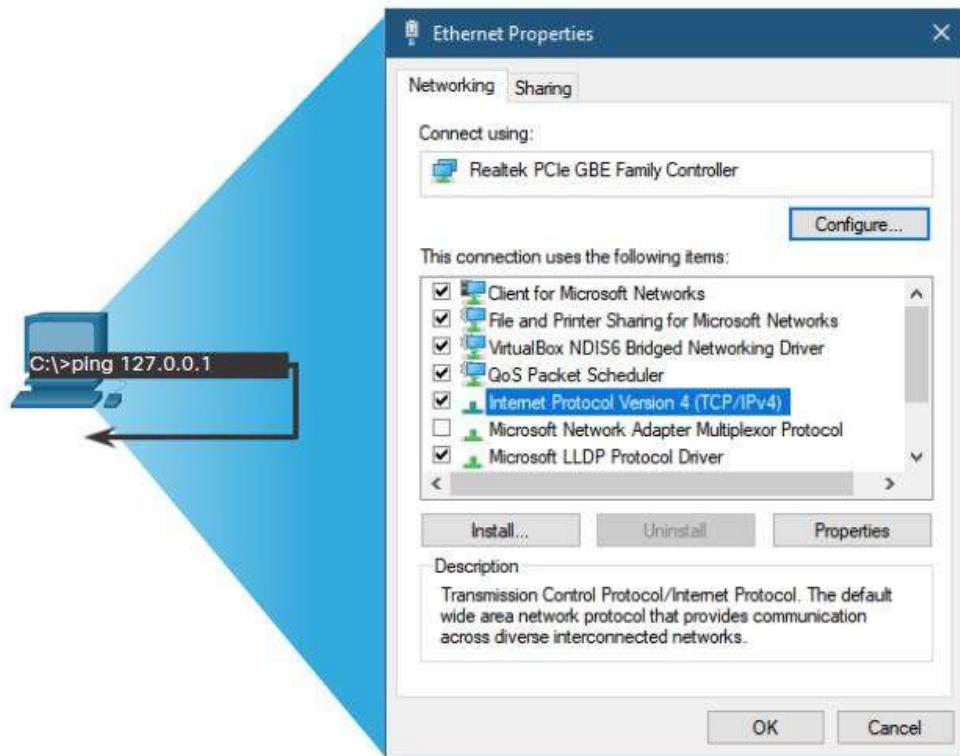
Jenis uji konektivitas yang dilakukan dengan **ping meliputi** yang berikut ini:

- Pinging the local loopback
- Pinging the default gateway
- Pinging the remote host

Ping the Loopback

Ping dapat digunakan untuk menguji konfigurasi internal IPv4 atau IPv6 pada host lokal. Untuk melakukan pengujian ini, **ping** alamat loopback lokal 127.0.0.1 untuk IPv4 (:1 untuk IPv6).

Response dari 127.0.0.1 untuk IPv4, atau ::1 untuk IPv6, menunjukkan bahwa IP diinstal dengan benar pada host. Respon ini berasal dari **network layer**. Namun, respons ini tidak merupakan indikasi bahwa alamat, **subnet**, atau gateway dikonfigurasi dengan benar. Juga tidak menunjukkan apa-apa tentang status **bottom layer**. Ini hanya menguji IP melalui **network layer** IP. Pesan kesalahan menunjukkan bahwa TCP/IP tidak beroperasi pada host.



- Pinging host lokal mengkonfirmasi bahwa TCP / IP diinstal dan bekerja pada host lokal.
- Ping 127.0.0.1 menyebabkan perangkat untuk ping itu sendiri.

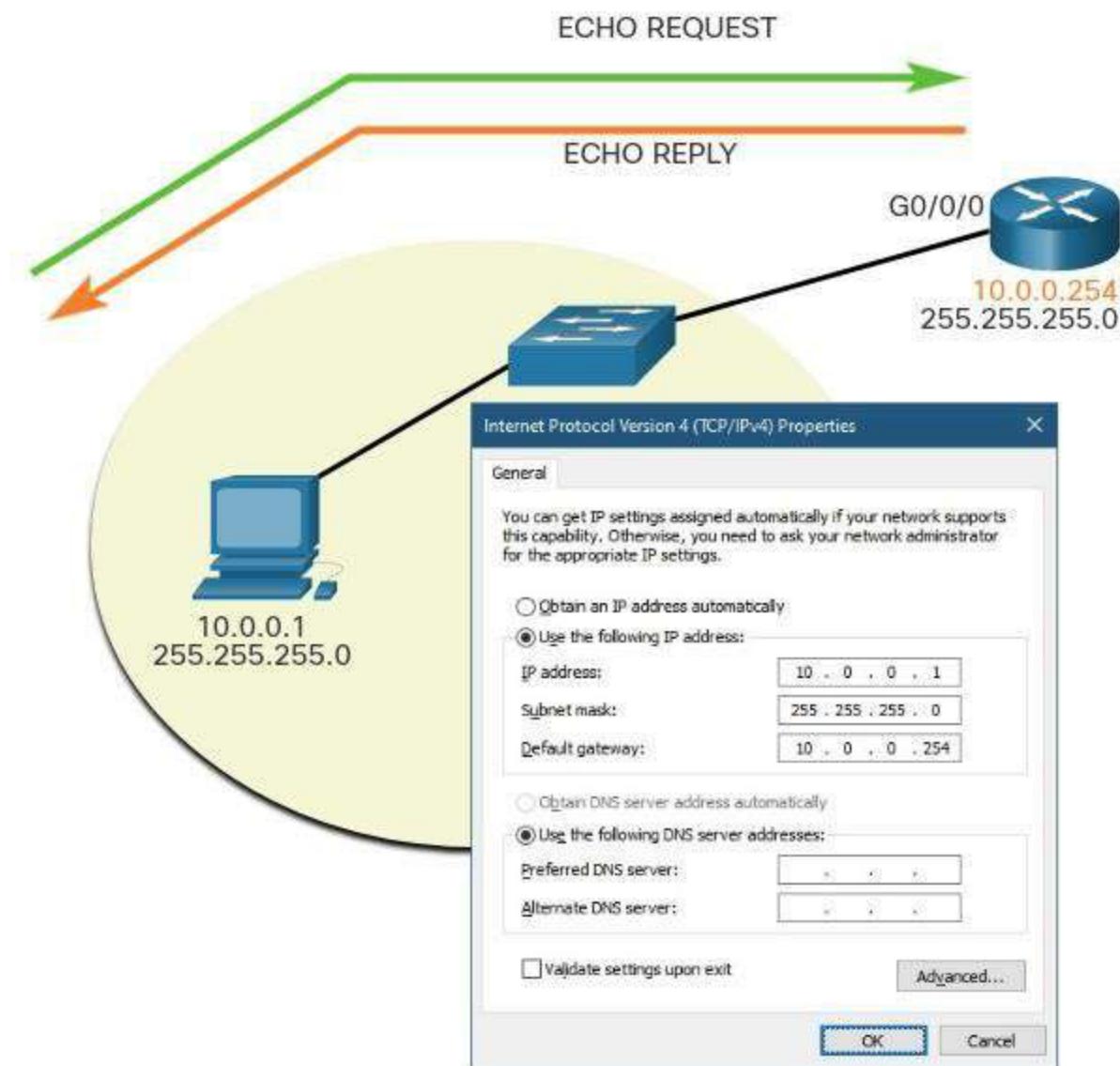
Ping the Default Gateway

Anda juga dapat **menggunakan ping** untuk menguji kemampuan host untuk berkomunikasi di jaringan lokal. Ini umumnya dilakukan dengan ping alamat IP **default gateway** host. Ping yang berhasil ke **default gateway** menunjukkan bahwa host dan **interface router** yang berfungsi sebagai **default gateway** keduanya beroperasi di jaringan lokal.

Untuk pengujian ini, alamat **default gateway** paling sering digunakan karena router biasanya selalu beroperasi. Jika alamat **default gateway** tidak merespons, **ping** dapat dikirim ke alamat IP host lain di jaringan lokal yang diketahui beroperasi.

Jika **default gateway** atau host lain merespons, maka host lokal dapat berhasil berkomunikasi melalui jaringan lokal. Jika **default gateway** tidak merespons tetapi host lain tidak, ini bisa menunjukkan masalah dengan **interface router** yang berfungsi sebagai **default gateway**.

Salah satu kemungkinan adalah bahwa alamat **default gateway** yang salah telah dikonfigurasi pada host. Kemungkinan lain adalah bahwa **interface** antara router mungkin sepenuhnya beroperasi tetapi memiliki keamanan yang diterapkan untuk itu yang mencegahnya memproses atau menanggapi permintaan ping.



host ping **default gateway** nya, mengirim Echo Request ICMP. **default gateway** mengirim Echo Reply yang mengkonfirmasi koneksi.

Ping Host Jarak Jauh

Ping juga dapat digunakan untuk menguji kemampuan host lokal untuk berkomunikasi di internetwork. **host** lokal dapat melakukan ping host IPv4 operasional dari jaringan jarak jauh, seperti yang ditunjukkan pada angka tersebut. Router menggunakan tabel **router** IP-nya untuk meneruskan paket.

Jika ping ini berhasil, pengoperasian sepotong besar internetwork dapat diverifikasi. Ping **yang** sukses di internetwork mengkonfirmasi komunikasi di jaringan lokal, pengoperasian router yang berfungsi sebagai **default gateway**, dan pengoperasian semua router lain yang mungkin berada di jalur antara jaringan lokal dan jaringan host jarak jauh.

Selain itu, fungsionalitas host jarak jauh dapat diverifikasi. Jika host jarak jauh tidak dapat berkomunikasi di luar jaringan lokalnya, itu tidak akan merespons.

Catatan: Banyak administrator jaringan membatasi atau melarang masuknya pesan ICMP ke dalam jaringan perusahaan; oleh karena itu, **kurangnya respons ping** bisa disebabkan oleh pembatasan keamanan.

Traceroute – Uji Jalur

Ping digunakan untuk menguji konektivitas antara dua host tetapi tidak memberikan informasi tentang detail perangkat antara host. Traceroute(**tracert**) adalah utilitas yang menghasilkan daftar hop yang berhasil dicapai di sepanjang jalur. Daftar ini dapat memberikan informasi verifikasi dan pemecahan masalah penting. Jika data mencapai tujuan, maka jejak mencantumkan **interface** setiap router di jalur antara host. Jika data gagal di beberapa hop di sepanjang jalan, alamat router terakhir yang merespons jejak dapat memberikan indikasi dimana masalah atau pembatasan keamanan ditemukan.

Round Trip Time (RTT)

Menggunakan traceroute menyediakan **round-trip time** untuk setiap lompatan di sepanjang jalur dan menunjukkan apakah hop gagal merespons. **round-trip time** adalah waktu yang dibutuhkan paket untuk mencapai host jarak jauh dan untuk respons dari **host** untuk kembali. Tanda bintang (*) digunakan untuk menunjukkan paket yang hilang atau tidak disederhanakan.

Informasi ini dapat digunakan untuk menemukan **router** bermasalah di jalur atau mungkin menunjukkan bahwa **router** dikonfigurasi untuk tidak membalas. Jika layar menunjukkan waktu respons tinggi atau kehilangan data dari hop tertentu, ini adalah indikasi bahwa sumber daya router atau koneksinya mungkin stres.

Batas IPv4 TTL dan IPv6 Hop

Traceroute menggunakan fungsi **field** TTL di IPv4 dan **field** Hop Limit di IPv6 di header Layer 3, bersama dengan pesan ICMP Time Exceeded.

Urutan pertama pesan yang dikirim dari traceroute akan memiliki nilai **field** TTL 1. Ini menyebabkan TTL kehabisan waktu paket IPv4 pada router pertama. Router ini kemudian merespons dengan pesan ICMPv4 Time Exceeded. Traceroute sekarang memiliki alamat hop pertama.

Traceroute kemudian secara progresif mengatur **field** TTL (2, 3, 4...) untuk setiap urutan pesan. Ini memberikan jejak dengan alamat setiap hop saat paket waktu keluar lebih jauh ke bawah jalur. **field** TTL terus ditingkatkan sampai tujuan tercapai, atau ditingkatkan secara maksimal.

Setelah tujuan akhir tercapai, **host** merespons dengan pesan ICMP Port Unreachable atau pesan ICMP Echo Reply alih-alih pesan ICMP Time Exceeded.

BAB 14

~ *Transport Layer* ~

Judul Bab : Transport Layer

Tujuan Bab : Membandingkan pengoperasian protokol transport layer dalam mendukung komunikasi end-to-end.

Link Test Pemahaman : <https://s.id/-Qy0s>

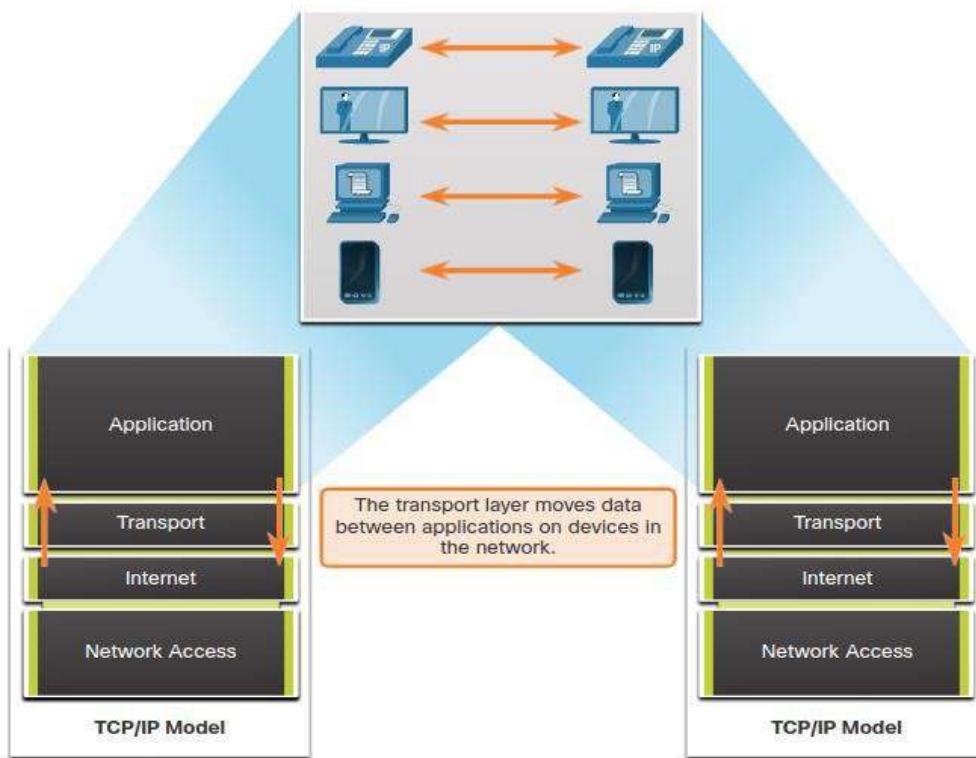
| Judul Materi | Tujuan Materi |
|------------------------------|---|
| Transportasi Data | Menjelaskan tujuan dari transport layer mengatur transportasi data end-to-end |
| Gambaran Umum TCP | Menjelaskan karakteristik TCP |
| Gambaran Umum UDP | Menjelaskan karakteristik UDP |
| Nomor Port | Menjelaskan bagaimana TCP dan UDP menggunakan nomor port |
| Cara kerja komunikasi TCP | Menjelaskan bagaimana pembentukan/establishment sesi TCP dan proses penghentian/termination dengan fasilitas komunikasi yang handal/reliability |
| Reliability dan flow control | Menjelaskan bagaimana protokol TCP data unit mengirim dan memberitahu bahwa paket telah sampai |
| Komunikasi UDP | Membandingkan pengoperasian protokol lapisan transport dalam mendukung komunikasi end-to-end. |

Transportasi Data

Program **Application Layer** menghasilkan data yang harus ditukarkan antara host sumber dan tujuan. **Transport Layer** bertanggung jawab atas komunikasi logis antara aplikasi yang berjalan pada host yang berbeda. Ini mungkin termasuk layanan seperti membuat sesi sementara antara dua host dan transmisi informasi yang dapat diandalkan untuk aplikasi.

Peran Transport Layer

Seperti yang ditunjukkan pada gambar, **Transport Layer** adalah **Link** antara **Application Layer** dan lapisan bawah yang bertanggung jawab untuk transmisi jaringan.



Transport Layer tidak memiliki pengetahuan tentang jenis host tujuan, jenis media di mana data harus bepergian, jalur yang diambil oleh data, kemacetan pada **Link**, atau ukuran jaringan.

Transport Layer mencakup dua protokol:

- **Transmission Control Protocol (TCP)**
- **User Datagram Protocol (UDP)**

Transport Layer Responsibilities

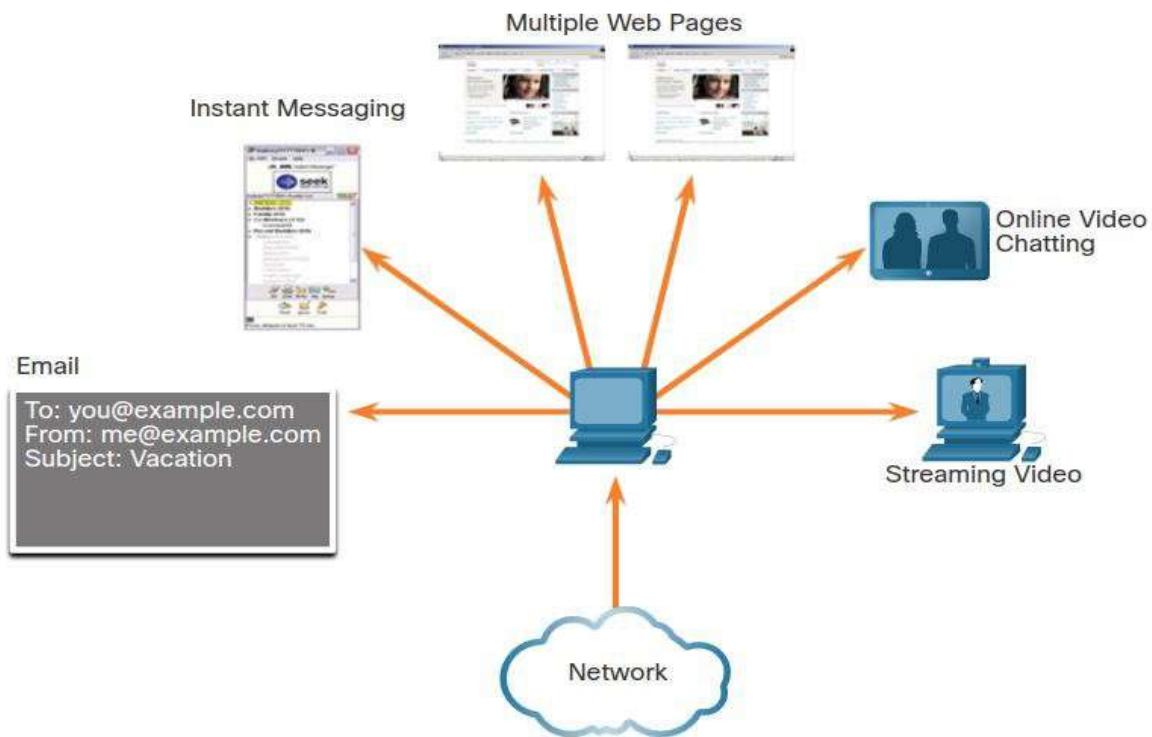
Transport layer memiliki banyak Responsibilities

1. Melacak Percakapan Individual

Pada **Transport Layer**, setiap kumpulan data yang mengalir antara aplikasi sumber dan aplikasi tujuan dikenal sebagai percakapan dan dilacak secara terpisah. Merupakan tanggung jawab **Transport Layer** untuk mempertahankan dan melacak beberapa percakapan ini.

Seperti yang diilustrasikan dalam gambar, host mungkin memiliki beberapa aplikasi yang berkomunikasi di seluruh jaringan secara bersamaan.

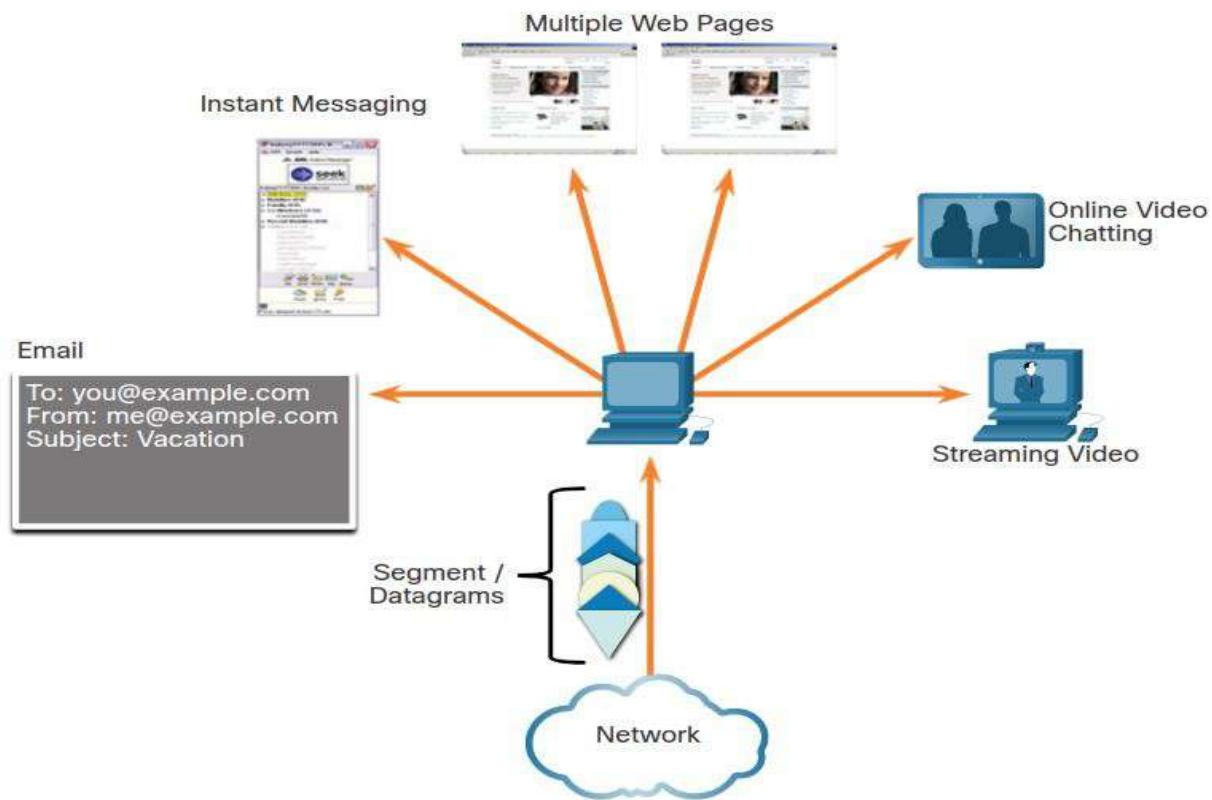
Sebagian besar jaringan memiliki batasan pada jumlah data yang dapat dimasukkan dalam satu paket. Oleh karena itu, data harus dibagi menjadi potongan-potongan yang dapat dikelola.



2. Segmentasi Data dan Menyusun Ulang Segmen

transport layer responsibility untuk membagi data aplikasi menjadi blok berukuran tepat. Tergantung pada protokol **transport layer** yang digunakan, **Transport layer Block** disebut segmen atau datagram. Gambar menggambarkan **Transport Layer** menggunakan **block** yang berbeda untuk setiap percakapan.

Transport Layer membagi data menjadi **block** yang lebih kecil (yaitu, segmen atau datagram) yang lebih mudah dikelola dan diangkut.

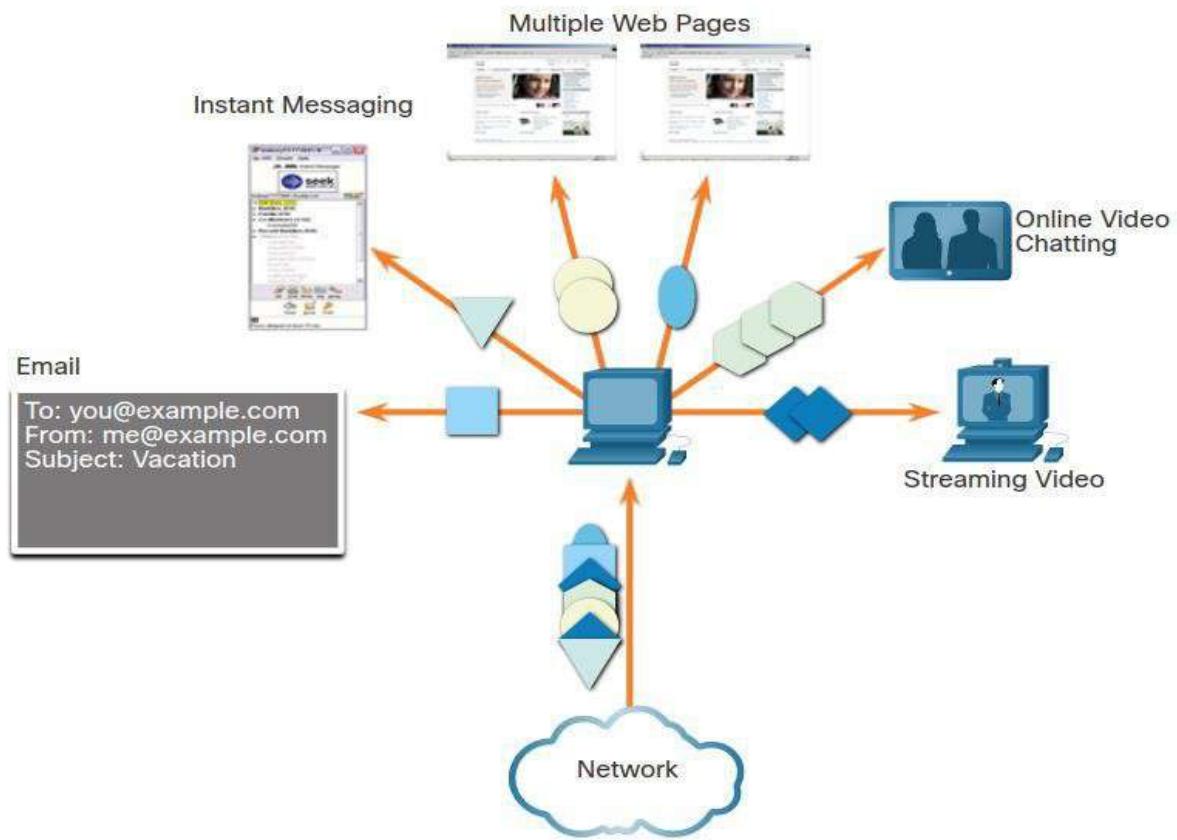


3. Tambah Informasi Header

Protokol **transport layer** juga menambahkan informasi header yang berisi data biner yang diatur ke dalam beberapa **field** ke setiap **block data**. Ini adalah nilai-nilai di **field** ini yang memungkinkan berbagai protokol **transport layer** untuk melakukan fungsi yang berbeda dalam mengelola komunikasi data.

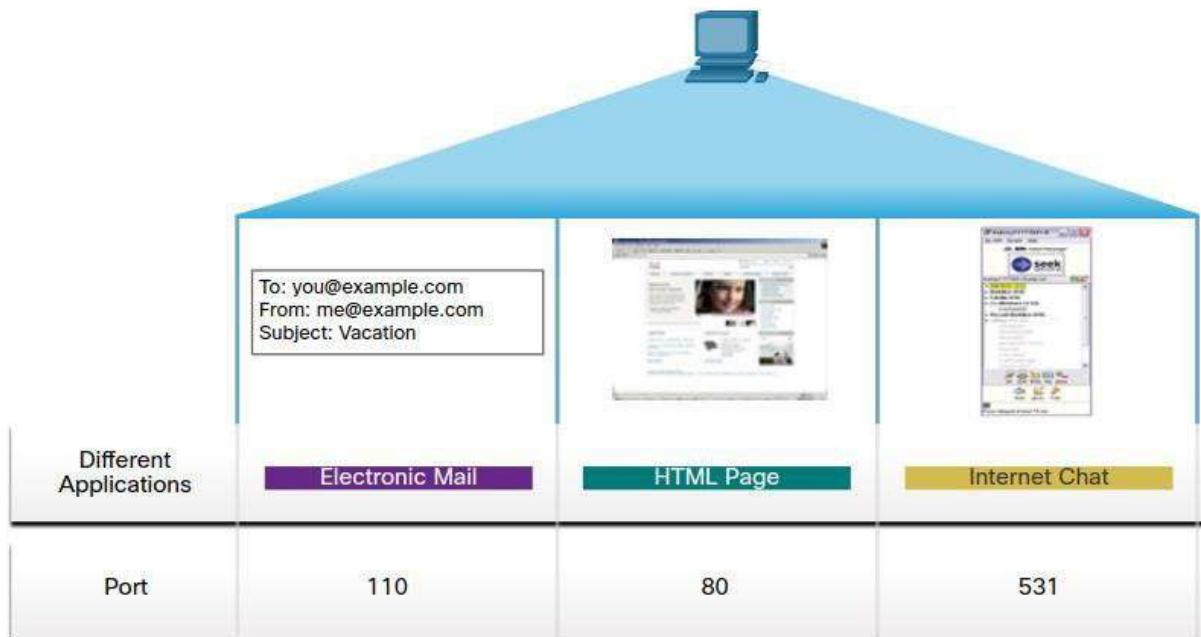
Misalnya, informasi header digunakan oleh host penerima untuk menyusun kembali **block data** ke dalam **data stream** lengkap untuk program **Application Layer** penerima.

Transport Layer memastikan bahwa bahkan dengan beberapa aplikasi yang berjalan pada perangkat, semua aplikasi menerima data yang benar.



4. Mengidentifikasi Aplikasi

Transport Layer harus dapat memisahkan dan mengelola beberapa komunikasi dengan kebutuhan transportasi yang berbeda. Untuk meneruskan **data stream** ke aplikasi yang tepat, **Transport Layer** mengidentifikasi aplikasi target menggunakan pengidentifikasi yang disebut nomor port. Seperti yang diilustrasikan dalam gambar, setiap proses perangkat lunak yang perlu mengakses jaringan diberi nomor port yang unik untuk host tersebut.

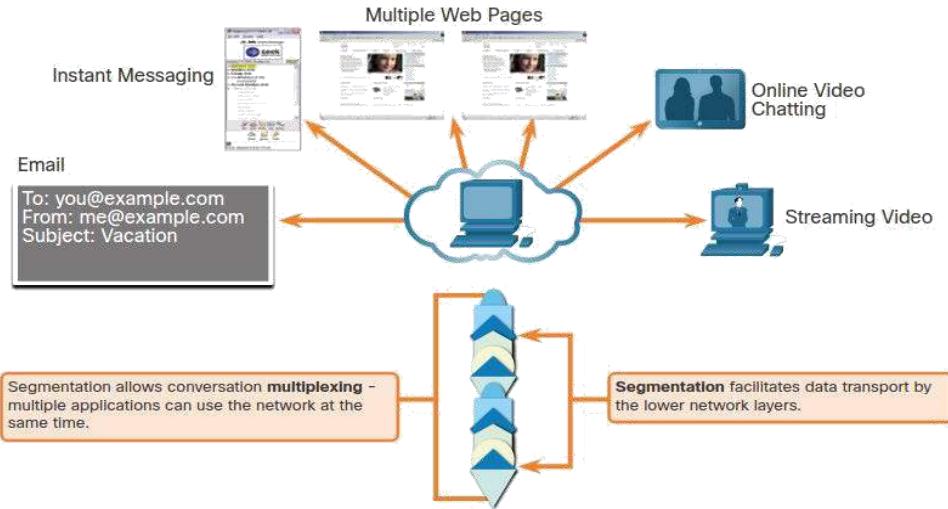


5. Multiplexing Percakapan

Mengirim beberapa jenis data (misalnya, video streaming) di seluruh jaringan, sebagai salah satu **communication stream** lengkap, dapat mengkonsumsi semua bandwidth yang tersedia. Ini akan mencegah percakapan komunikasi lain terjadi pada saat yang sama. Ini juga akan membuat **error recovery and retransmission** data yang rusak.

Seperti yang ditunjukkan pada gambar, **Transport Layer** menggunakan segmentasi dan multipleks untuk memungkinkan percakapan komunikasi yang berbeda untuk diinterleaved pada jaringan yang sama.

Error checking dapat dilakukan pada data di segmen, untuk menentukan apakah segmen diubah selama transmisi.

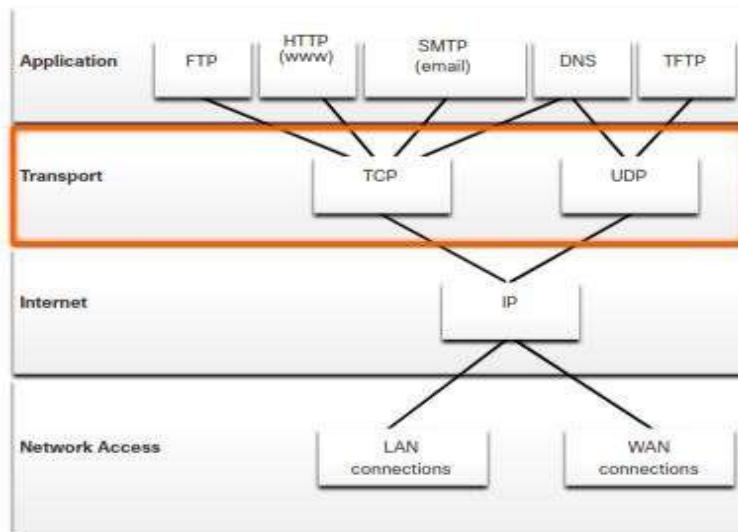


Transport Layer Protocols

IP hanya berkaitan dengan struktur, alamat, dan perutean paket. IP tidak menentukan bagaimana pengiriman atau pengangkutan paket terjadi.

Protokol **Transport Layer** menentukan cara mentransfer pesan antar host, dan bertanggung jawab untuk mengelola persyaratan **reliable** percakapan. **Transport Layer** mencakup protokol TCP dan UDP.

Aplikasi yang berbeda memiliki persyaratan **reliable** transportasi yang berbeda. Oleh karena itu, TCP/IP menyediakan dua protokol **Transport Layer**, seperti yang ditunjukkan pada gambar.



Transmission Control Protocol (TCP)

IP hanya berkaitan dengan struktur, alamat, dan perutean paket, dari pengirim asli ke tujuan akhir. IP tidak bertanggung jawab untuk menjamin pengiriman atau menentukan apakah koneksi antara pengirim dan penerima perlu dibuat.

TCP dianggap sebagai protokol **Transport Layer** berfitur lengkap yang andal,, yang memastikan bahwa semua data tiba di tujuan. TCP mencakup **field** yang memastikan pengiriman data aplikasi. **field** ini memerlukan pemrosesan tambahan oleh **host** pengirim dan penerima.

Catatan: TCP membagi data menjadi segmen.

Transportasi TCP dianalogikan dengan pengiriman paket yang dilacak dari sumber ke tujuan. Jika pesanan pengiriman dipecah menjadi beberapa paket, pelanggan dapat memeriksa secara online untuk melihat urutan pengiriman.

TCP menyediakan **reliable** dan **flow control** menggunakan operasi dasar ini:

- Segmen data angka dan trek yang dikirimkan ke host tertentu dari aplikasi tertentu
- Mengakui data yang diterima
- Kirim ulang data yang tidak diakui setelah beberapa waktu
- Mengurutkan data yang mungkin datang dalam urutan yang salah
- Mengirim data dengan kecepatan efisien yang dapat diterima oleh penerima

Untuk mempertahankan status percakapan dan melacak informasi, TCP harus terlebih dahulu membuat koneksi antara pengirim dan penerima. Inilah sebabnya mengapa TCP dikenal sebagai protokol yang berorientasi pada koneksi.

User Datagram Protocol (UDP)

UDP adalah protokol **Transport Layer** yang lebih sederhana daripada TCP. Ini tidak memberikan **reliable** dan **flow control**, yang berarti membutuhkan lebih sedikit **field** header. Karena proses UDP pengirim dan penerima tidak harus mengelola **reliable** dan **flow control**, ini berarti datagram UDP dapat diproses lebih cepat daripada segmen TCP. UDP menyediakan fungsi dasar untuk mengirimkan datagram antara aplikasi yang sesuai, dengan sangat sedikit overhead dan pengecekan data.

Catatan: UDP membagi data menjadi datagram yang juga disebut sebagai segmen.

UDP adalah protokol **connectionless**. Karena UDP tidak memberikan **reliable** atau **flow control**, UDP tidak memerlukan koneksi **established**. Karena UDP tidak melacak informasi yang dikirim atau diterima antara klien dan server, UDP juga dikenal sebagai protokol **stateless**.

UDP juga dikenal sebagai protokol pengiriman usaha terbaik karena tidak ada **Acknowledgments** bahwa data diterima di tempat tujuan. Dengan UDP, tidak ada proses **Transport Layer** yang menginformasikan pengirim pengiriman yang sukses.

UDP seperti menempatkan surat biasa, tidak terdaftar, dalam surat. Pengirim surat tersebut tidak mengetahui ketersediaan penerima untuk menerima surat tersebut. Kantor pos juga tidak bertanggung jawab untuk melacak surat atau memberi tahu pengirim jika surat itu tidak sampai di tujuan akhir.

Protokol Transport Layer yang Tepat untuk Aplikasi yang Tepat

Beberapa aplikasi dapat mentolerir beberapa kehilangan data selama transmisi melalui jaringan, tetapi keterlambatan transmisi tidak dapat diterima. Untuk aplikasi ini, UDP adalah pilihan yang lebih baik karena membutuhkan lebih sedikit overhead jaringan. UDP lebih disukai untuk aplikasi seperti Voice over IP (VoIP). **Acknowledgments** dan **Retransmission** akan memperlambat pengiriman dan membuat percakapan suara tidak dapat diterima.

UDP juga digunakan oleh aplikasi request-and-reply di mana data minimal, dan retransmission dapat dilakukan dengan cepat. Misalnya, domain name service (DNS) menggunakan UDP untuk jenis transaksi ini. Klien meminta alamat IPv4 dan IPv6 untuk nama domain yang dikenal dari server DNS. Jika klien tidak menerima respons dalam jumlah waktu yang telah ditentukan, itu hanya mengirim permintaan lagi.

Misalnya, jika satu atau dua segmen streaming video langsung gagal tiba, streaming video akan membuat gangguan sesaat di streaming. Ini mungkin muncul sebagai distorsi pada gambar atau suara, tetapi mungkin tidak terlihat oleh pengguna. Jika perangkat tujuan harus memperhitungkan data yang hilang, aliran dapat tertunda sambil menunggu retransmissions, oleh karena itu menyebabkan gambar atau suara sangat terdegradasi. Dalam hal ini, lebih baik untuk membuat media terbaik dengan segmen yang diterima, dan **reliable** di atas.

Untuk aplikasi lain, penting bahwa semua data tiba dan dapat diproses dalam urutan yang tepat. Untuk jenis aplikasi ini, TCP digunakan sebagai protokol transportasi. Misalnya, aplikasi seperti database, browser web, dan klien email, mengharuskan semua data yang dikirim tiba di tujuan dalam kondisi aslinya. Data yang hilang dapat merusak komunikasi,

membuatnya tidak lengkap atau tidak terbaca. Misalnya, penting ketika mengakses informasi perbankan melalui web untuk memastikan semua informasi dikirim dan diterima dengan benar.

Pengembang aplikasi harus memilih jenis protokol transportasi mana yang sesuai berdasarkan persyaratan aplikasi. Video dapat dikirim melalui TCP atau UDP. Aplikasi yang melakukan streaming audio dan video yang disimpan biasanya menggunakan TCP. Aplikasi ini menggunakan TCP untuk melakukan buffering, bandwidth probing, dan **congestion control**, untuk mengontrol pengalaman pengguna dengan lebih baik.

Video dan suara real-time biasanya menggunakan UDP, tetapi juga dapat menggunakan TCP, atau UDP dan TCP. Aplikasi konferensi video dapat menggunakan UDP secara default, tetapi karena banyak firewall memblokir UDP, aplikasi juga dapat dikirim melalui TCP.

Aplikasi yang melakukan streaming audio dan video yang disimpan menggunakan TCP. Misalnya, jika jaringan Anda tiba-tiba tidak dapat mendukung bandwidth yang diperlukan untuk menonton film sesuai permintaan, aplikasi akan menjeda pemutaran. Selama jeda, Anda mungkin melihat “buffering...” saat TCP berfungsi untuk membuat ulang streaming. Ketika semua segmen beres dan tingkat bandwidth minimum dipulihkan, sesi TCP Anda dilanjutkan, dan film dilanjutkan diputar.

Gambaran Umum TCP

Dalam **Materi** sebelumnya, Anda mengetahui bahwa TCP dan UDP adalah dua protokol **Transport Layer**. **Materi** ini memberikan detail lebih lanjut tentang apa yang dilakukan TCP dan kapan ide yang baik untuk menggunakan protokol UDP.

Fitur TCP

Untuk memahami perbedaan antara TCP dan UDP, penting untuk memahami bagaimana setiap protokol menerapkan fitur **Reliability** tertentu dan bagaimana setiap protokol melacak percakapan.

Selain mendukung fungsi dasar segmentasi data dan reassembly, TCP juga menyediakan layanan berikut:

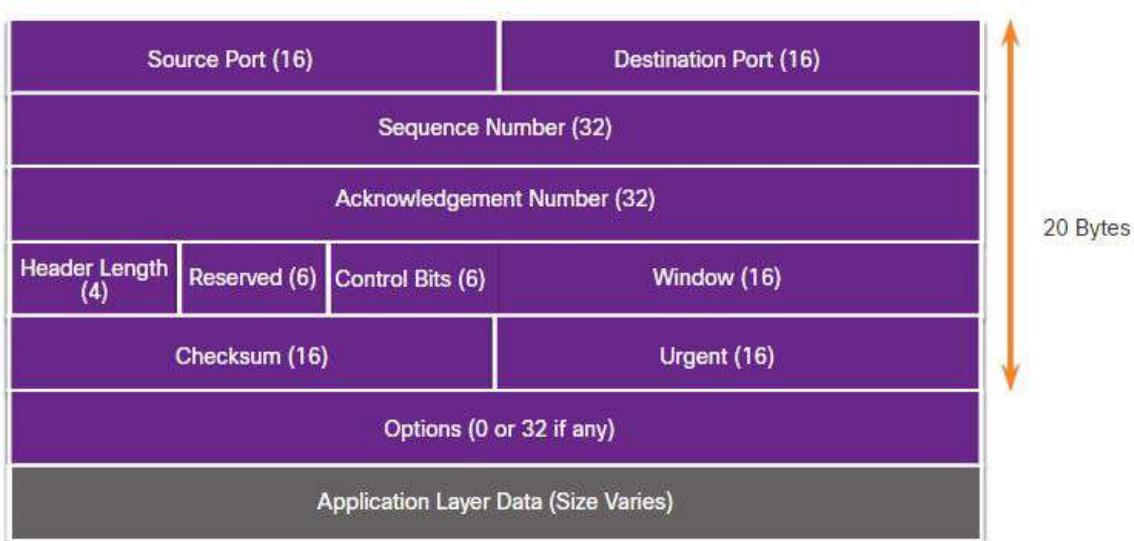
- **Establishes a Session** - TCP adalah protokol berorientasi koneksi yang menegosiasikan dan membuat koneksi permanen (atau sesi) antara perangkat sumber dan tujuan sebelum meneruskan lalu lintas apa pun. Melalui pembentukan sesi, perangkat menegosiasikan jumlah lalu lintas yang dapat diteruskan pada waktu tertentu, dan data komunikasi antara keduanya dapat dikelola dengan cermat.
- **Ensures Reliable Delivery** - Untuk banyak alasan, dimungkinkan bagi segmen untuk menjadi rusak atau hilang sepenuhnya, karena ditransmisikan melalui jaringan. TCP memastikan bahwa setiap segmen yang dikirim oleh sumber tiba di tempat tujuan.
- **Provides Same-Order Delivery** - Karena jaringan dapat menyediakan beberapa rute yang dapat memiliki tingkat transmisi yang berbeda, data dapat tiba dalam urutan yang salah. Dengan menomori dan mengurutkan segmen, TCP memastikan segmen disusun kembali ke dalam urutan yang tepat.
- **Supports Flow Control** - Host jaringan memiliki **Resources** terbatas (yaitu, memori dan daya pemrosesan). Ketika TCP menyadari bahwa **Resources** ini **overtaxed**, TCP dapat meminta agar aplikasi pengirim mengurangi laju aliran data. Ini dilakukan dengan TCP mengatur jumlah data yang ditransmisikan sumber. **Flow Control** dapat mencegah kebutuhan untuk mengirimkan ulang data ketika **Resources** host penerima kewalahan.

Untuk informasi lebih lanjut tentang TCP, cari di internet untuk RFC 793.

TCP Header

TCP adalah protokol yang **statefull** yang berarti melacak keadaan sesi komunikasi. Untuk melacak status sesi, TCP mencatat informasi mana yang telah dikirimnya dan informasi mana yang telah diakui. Sesi yang dinyatakan dimulai dengan pembentukan sesi dan diakhiri dengan penghentian sesi.

Segmen TCP menambahkan 20 byte (yaitu, 160 bit) overhead saat merangkum data **Application Layer**. Gambar memperlihatkan **Field** di header TCP.



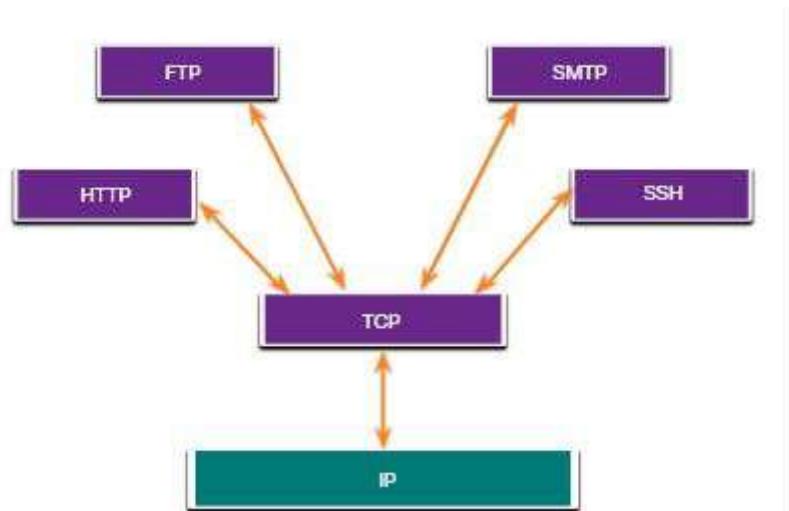
Field Header TCP

Tabel mengidentifikasi dan menguraikan sepuluh **Field** dalam header TCP.

| TCP Header Field | deskripsi |
|-----------------------|---|
| Source Port | Field 16-bit yang digunakan untuk mengidentifikasi aplikasi sumber berdasarkan nomor port. |
| Destination Port | Field 16-bit yang digunakan untuk mengidentifikasi aplikasi tujuan berdasarkan nomor port. |
| Sequence Number | Field 32-bit yang digunakan untuk tujuan reassembly data. |
| Acknowledgment Number | Field 32-bit yang digunakan untuk menunjukkan bahwa data telah diterima dan byte berikutnya diharapkan dari sumber. |
| Header Length | Field 4-bit yang dikenal sebagai "data offset" yang menunjukkan panjang header segmen TCP. |
| Reserved | Field 6-bit yang disediakan untuk digunakan di masa mendatang. |
| Control Bits | Field 6-bit yang menyertakan kode bit, atau flags, yang menunjukkan tujuan dan fungsi segmen TCP. |
| Window Size | Field 16-bit digunakan untuk menunjukkan jumlah byte yang dapat diterima pada satu waktu. |
| Checksum | Field 16-bit digunakan untuk pemeriksaan kesalahan header segmen dan data. |
| Urgent | Field 16-bit digunakan untuk menunjukkan apakah data yang terkandung adalah penting |

Aplikasi yang menggunakan TCP

TCP adalah contoh yang baik tentang bagaimana berbagai Rangkaian **layer protokol TCP / IP** memiliki masing masing peran khusus. TCP menangani semua tugas yang terkait dengan membagi **stream data** menjadi segmen, memberikan **Reability, control data flow**, dan **reordering segmen**. TCP membebaskan aplikasi dari keharusan untuk mengelola tugas-tugas yang tidak diperlukan. Aplikasi, seperti yang ditunjukkan dalam gambar, cukup mengirim **stream data** ke **Transport Layer** dan menggunakan layanan TCP.



Gambaran Umum UDP

Materi ini akan membahas UDP, apa yang dilakukannya, dan kapan UDP digunakan. UDP adalah protokol transportasi melakukan upaya terbaik. UDP adalah protokol transportasi ringan yang menawarkan segmentasi data yang sama dan reassembly sebagai TCP, tetapi tanpa **Reliability** TCP dan **Flow Control**.

Fitur UDP

UDP adalah protokol sederhana yang biasanya dijelaskan dalam hal apa yang tidak dilakukannya dibandingkan dengan TCP.

Fitur UDP meliputi yang berikut ini:

- Data direkonstruksi dalam urutan diterima.
- Setiap segmen yang hilang tidak dikirim balik.
- Tidak ada pembentukan sesi.
- Pengiriman tidak diberitahu tentang ketersediaan **Resources**.

Untuk informasi lebih lanjut tentang UDP, cari di internet untuk RFC.

UDP Header

UDP adalah protokol stateless, yang berarti baik klien, maupun server, melacak keadaan sesi komunikasi. Jika **Reability** diperlukan saat menggunakan UDP sebagai protokol transportasi, maka harus ditangani oleh aplikasi.

Salah satu persyaratan terpenting untuk menyampaikan video dan suara langsung melalui jaringan adalah data terus mengalir dengan cepat. Aplikasi video dan suara langsung dapat mentolerir beberapa kehilangan data dengan efek minimal atau tidak terlihat, dan sangat cocok untuk UDP.

Blok komunikasi di UDP disebut datagram, atau segmen. Datagram ini dikirim sebagai upaya terbaik oleh protokol **Transport Layer**.

Header UDP jauh lebih sederhana daripada header TCP karena hanya memiliki empat **Field** dan membutuhkan 8 byte (yaitu, 64 bit). Gambar memperlihatkan **Field** di header UDP.



UDP Header Field

Tabel mengidentifikasi dan menjelaskan empat **Field** dalam header UDP.

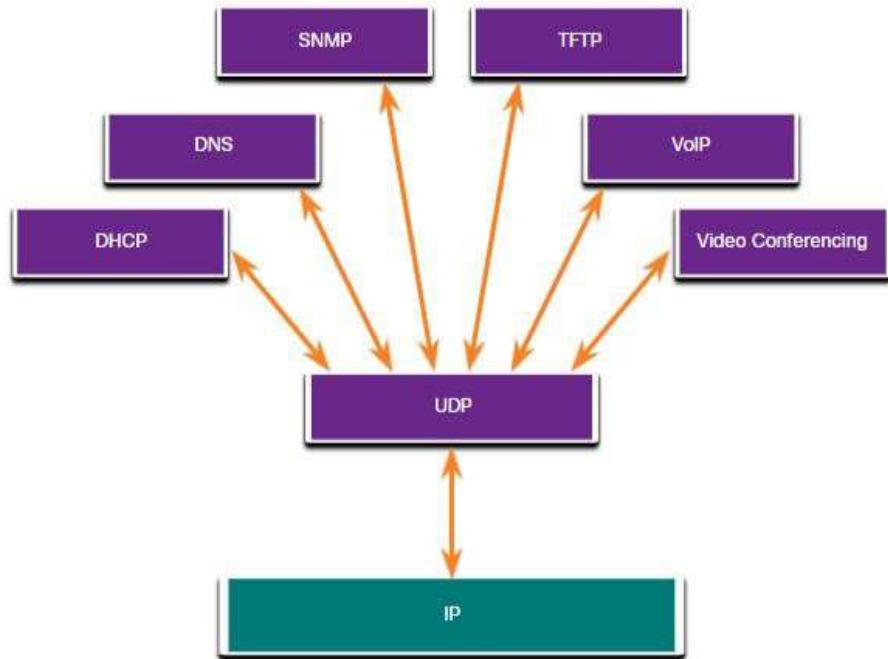
| UDP Header Field | deskripsi |
|------------------|--|
| Source Port | Field 16-bit yang digunakan untuk mengidentifikasi aplikasi sumber berdasarkan nomor port. |
| Destination Port | Field 16-bit yang digunakan untuk mengidentifikasi aplikasi tujuan berdasarkan nomor port. |
| Length | Field 16-bit yang menunjukkan panjang header datagram UDP. |
| Checksum | Field 16-bit digunakan untuk pemeriksaan kesalahan header dan data datagram. |

Aplikasi yang menggunakan UDP

Ada tiga jenis aplikasi yang paling cocok untuk UDP:

- **Live video and multimedia applications** – Aplikasi ini dapat mentolerir beberapa kehilangan data, tetapi memerlukan sedikit atau tidak ada penundaan. Contohnya termasuk VoIP dan video live streaming.
- **Simple request and reply applications** – Aplikasi dengan transaksi sederhana di mana host mengirim permintaan dan mungkin atau mungkin tidak menerima balasan. Contohnya termasuk DNS dan DHCP.
- **Applications that handle reliability themselves** – Komunikasi unidirectional di mana **Flow Control**, **Error Detection**, **Acknowledgments**, dan **Error Recovery** tidak diperlukan, atau dapat ditangani oleh aplikasi. Contohnya termasuk SNMP dan TFTP.

Gambar tersebut mengidentifikasi aplikasi yang membutuhkan UDP.



Nomor Port

Seperti yang telah Anda pelajari, ada beberapa situasi di mana TCP adalah protokol yang tepat untuk pekerjaan itu, dan situasi lain di mana UDP harus digunakan. Apa pun jenis data yang diangkut, baik TCP maupun UDP menggunakan nomor port.

Beberapa Komunikasi Terpisah

Protokol **Transport Layer** **TCP** dan **UDP** menggunakan nomor port untuk mengelola beberapa percakapan simultan. Seperti yang ditunjukkan pada gambar, **Field header** TCP dan UDP mengidentifikasi nomor port **source and destination application**.



Nomor **Source Port** dikaitkan dengan **source application** pada host lokal sedangkan nomor **Destination Port** dikaitkan dengan **destination application** pada **remote host**.

Misalnya, misalkan host memulai **request** halaman web dari server web. Ketika host memulai **request** halaman web, nomor **Source Port** secara dinamis dihasilkan oleh host untuk mengidentifikasi percakapan secara unik. Setiap **request** yang dihasilkan oleh host akan menggunakan nomor **Source Port** yang dibuat secara dinamis yang berbeda. Proses ini memungkinkan beberapa percakapan terjadi secara bersamaan.

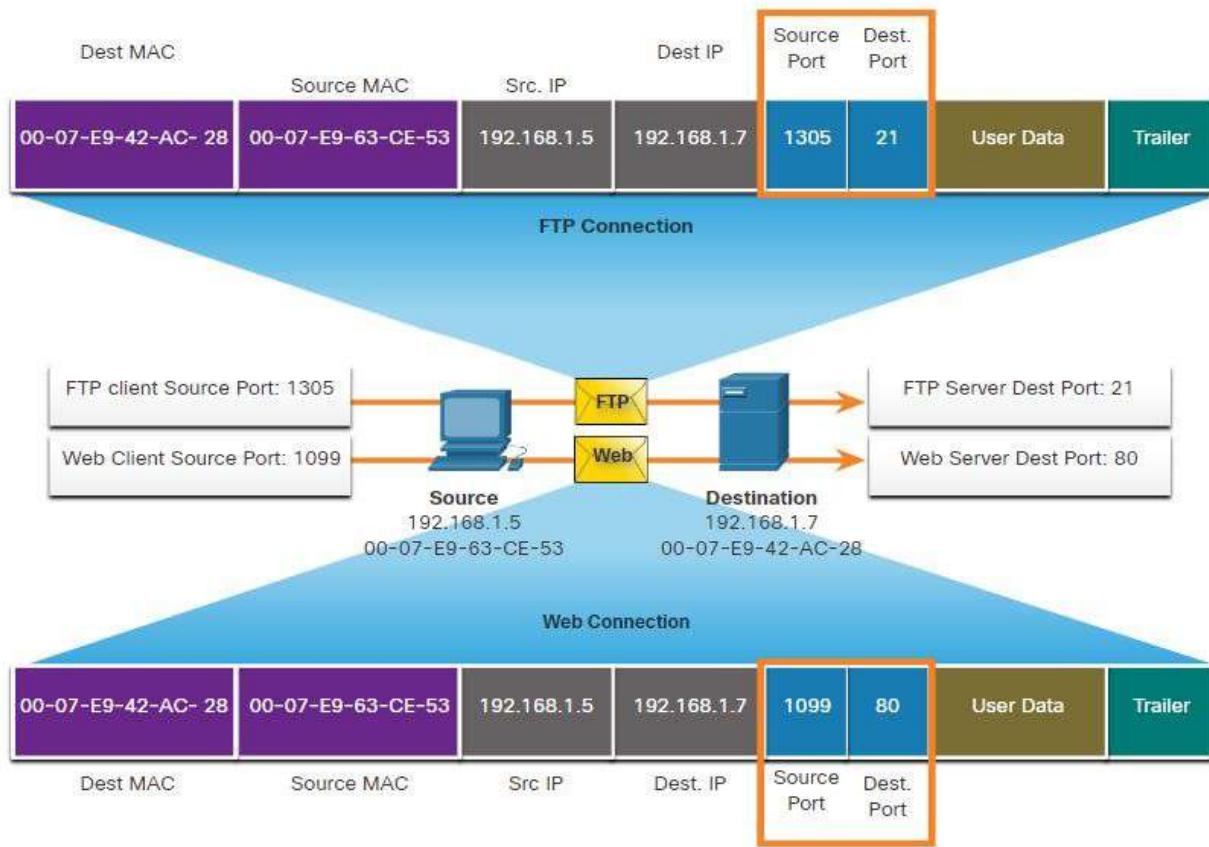
Dalam **request**, nomor **Destination Port** adalah apa yang mengidentifikasi jenis layanan yang diminta dari server web tujuan.. Misalnya, ketika klien menentukan port 80 di **Destination Port**, server yang menerima pesan tahu bahwa layanan web sedang diminta.

Server dapat menawarkan lebih dari satu layanan secara bersamaan seperti layanan web pada port 80 sementara ia menawarkan pembentukan koneksi File Transfer Protocol (FTP) pada port 21.

Pasangan Soket

Source and Destination Port ditempatkan di dalam segmen. Segmen kemudian dienkapsulasi dalam paket IP. Paket IP berisi alamat IP sumber dan tujuan. Kombinasi alamat IP sumber dan nomor **Source Port**, atau alamat IP tujuan dan nomor **Destination Port** dikenal sebagai soket.

Dalam contoh dalam gambar, PC secara bersamaan **request** FTP dan layanan web dari server tujuan.



Dalam contoh, **request** FTP yang dihasilkan oleh PC mencakup alamat MAC Layer 2 dan alamat IP Layer 3. **Request** ini juga mengidentifikasi nomor **Source Port** 1305 (yaitu, yang dihasilkan secara dinamis oleh host) dan **Destination Port**, mengidentifikasi layanan FTP pada port 21. Host juga telah **request** halaman web dari server menggunakan alamat Layer 2 dan Layer 3 yang sama. Namun, ia menggunakan nomor **Source Port** 1099 (yaitu, secara dinamis dihasilkan oleh host) dan **Destination Port** mengidentifikasi layanan web pada port 80.

Soket digunakan untuk mengidentifikasi server dan layanan yang diminta oleh klien. Soket klien mungkin terlihat seperti ini, dengan 1099 mewakili nomor **Source Port**: 192.168.1.5:1099

Soket di server web mungkin 192.168.1.7:80

Bersama-sama, kedua soket ini digabungkan untuk membentuk *pasangan soket*: 192.168.1.5:1099, 192.168.1.7:80

Soket memungkinkan beberapa proses, berjalan pada klien, untuk membedakan diri satu sama lain, dan beberapa koneksi ke proses server untuk dibedakan satu sama lain.

Nomor **Source Port** bertindak sebagai alamat pengirim untuk aplikasi yang **request**. **Transport Layer** melacak port ini dan aplikasi yang memulai **request** sehingga ketika respons dikembalikan, dapat diteruskan ke aplikasi yang benar.

Grup Nomor Port

Internet Assigned Numbers Authority (IANA) adalah organisasi standar yang bertanggung jawab untuk menetapkan berbagai standar addressing, termasuk nomor port 16-bit. 16 bit yang digunakan untuk mengidentifikasi sumber dan nomor **Destination Port** menyediakan berbagai port dari 0 hingga 65535.

IANA telah membagi kisaran angka ke dalam tiga grup port berikut.

| Grup Port | Rentang Angka | deskripsi |
|------------------|---------------------|---|
| Well-known Ports | 0 hingga 1.023 | - Nomor port ini disediakan untuk layanan dan aplikasi umum atau populer seperti browser web, klien email, dan akses remote. – Well-known Ports yang ditentukan untuk aplikasi server umum memungkinkan klien untuk dengan mudah mengidentifikasi layanan terkait yang diperlukan. |
| Registered Ports | 1.024 hingga 49.151 | - Nomor port ini ditetapkan oleh IANA ke entitas yang request untuk digunakan dengan proses atau aplikasi tertentu. |

| | | |
|---------------------------|----------------------|--|
| | | <ul style="list-style-type: none"> - Proses ini terutama merupakan aplikasi individual yang dipilih pengguna untuk diinstal, daripada aplikasi umum yang akan menerima well-known port number. – Misalnya, Cisco telah mendaftarkan port 1812 untuk proses otentikasi server RADIUS-nya. |
| Private and Dynamic Ports | 49.152 hingga 65.535 | Port ini juga dikenal <i>ephemeral ports</i> . OS klien biasanya menetapkan nomor port secara dinamis ketika koneksi ke layanan dimulai. Port dinamis kemudian digunakan untuk mengidentifikasi aplikasi klien selama komunikasi. |

Catatan: Beberapa sistem operasi klien dapat menggunakan nomor port terdaftar alih-alih nomor port dinamis untuk menentukan **Source Port**.

Tabel menampilkan beberapa nomor port umum yang terkenal dan aplikasi terkaitnya.

Well-Known Port Numbers

| Port Number | Protocol | Application |
|-------------|----------|---|
| 20 | TCP | File Transfer Protocol (FTP) – Data |
| 21 | TCP | File Transfer Protocol (FTP) – Control |
| 22 | TCP | Secure Shell (SSH) |
| 23 | TCP | Telnet |
| 25 | TCP | Simple Mail Transfer Protocol (SMTP) |
| 53 | UDP, TCP | Domain Name Service (DNS) |
| 67 | UDP | Dynamic Host Configuration Protocol (DHCP) – Server |
| 68 | UDP | Dynamic Host Configuration Protocol – Client |
| 69 | UDP | Trivial File Transfer Protocol (TFTP) |
| 80 | TCP | Hypertext Transfer Protocol (HTTP) |
| 110 | TCP | Post Office Protocol version 3 (POP3) |
| 143 | TCP | Internet Message Access Protocol (IMAP) |

| | | |
|-----|-----|--|
| 161 | UDP | Simple Network Management Protocol (SNMP) |
| 443 | TCP | Hypertext Transfer Protocol Secure (HTTPS) |

Beberapa aplikasi mungkin menggunakan TCP dan UDP. Misalnya, DNS menggunakan UDP saat klien mengirim **request** ke server DNS. Namun, komunikasi antara dua server DNS selalu menggunakan TCP.

Cari situs web IANA untuk registry port untuk melihat daftar lengkap nomor port dan aplikasi terkait.

Perintah netstat

Koneksi TCP yang tidak dapat dijelaskan dapat menimbulkan ancaman keamanan utama. Mereka dapat menunjukkan bahwa sesuatu atau seseorang terhubung ke host lokal. Terkadang perlu diketahui koneksi TCP aktif mana yang terbuka dan berjalan pada host jaringan. Netstat adalah utilitas jaringan penting yang dapat digunakan untuk memverifikasi koneksi tersebut. Seperti ditunjukkan di bawah ini, masukkan perintah **netstat** untuk mencantumkan protokol yang digunakan, alamat lokal dan nomor port, alamat asing dan nomor port, dan status koneksi.

```
C:\> netstat

Active Connections

  Proto  Local Address          Foreign Address        State
  TCP    192.168.1.124:3126    192.168.0.2:netbios-ssn  ESTABLISHED
  TCP    192.168.1.124:3158    207.138.126.152:http   ESTABLISHED
  TCP    192.168.1.124:3159    207.138.126.169:http  ESTABLISHED
  TCP    192.168.1.124:3160    207.138.126.169:http  ESTABLISHED
  TCP    192.168.1.124:3161    sc.msn.com:http      ESTABLISHED
  TCP    192.168.1.124:3166    www.cisco.com:http   ESTABLISHED
  (output omitted)
C:\>
```

Secara default, perintah **netstat** akan mencoba untuk menyelesaikan alamat IP untuk nama domain dan nomor port ke aplikasi terkenal. Opsi **-n** dapat digunakan untuk menampilkan alamat IP dan nomor port dalam bentuk numeriknya.

Proses Komunikasi TCP

Anda sudah tahu dasar-dasar TCP. Memahami peran nomor port akan membantu Anda memahami detail proses komunikasi TCP. Dalam materi ini, Anda juga akan belajar tentang **TCP Three-Way Handshake** dan **session termination**

Proses Server TCP

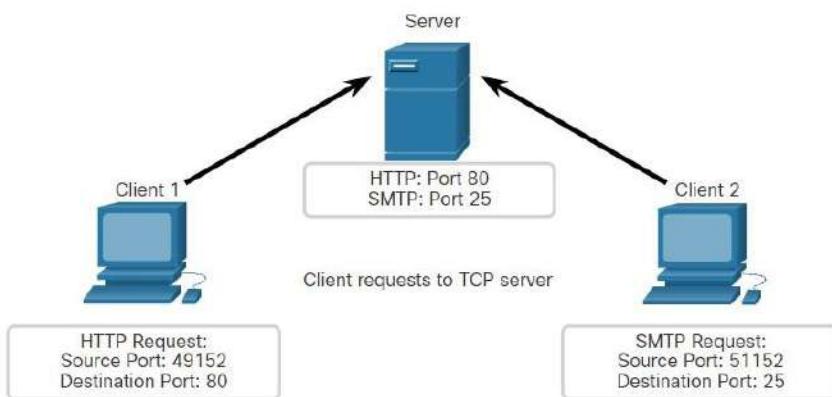
Setiap proses aplikasi yang berjalan pada server dikonfigurasi untuk menggunakan nomor port. Nomor port secara otomatis ditetapkan atau dikonfigurasi secara manual oleh administrator sistem.

Server individual tidak dapat memiliki dua layanan yang ditetapkan ke nomor port yang sama dalam layanan **transport layer** yang sama. Misalnya, host yang menjalankan aplikasi server web dan aplikasi transfer file tidak dapat menggunakan port yang sama, seperti port TCP 80.

Aplikasi server aktif yang ditetapkan ke port tertentu dianggap terbuka, yang berarti bahwa **transport layer** menerima, dan memproses segmen yang ditujukan ke port tersebut. Setiap **request** klien masuk yang ditujukan ke soket yang benar diterima, dan data diteruskan ke aplikasi server. Mungkin ada banyak port yang terbuka secara bersamaan di server, satu untuk setiap aplikasi server aktif.

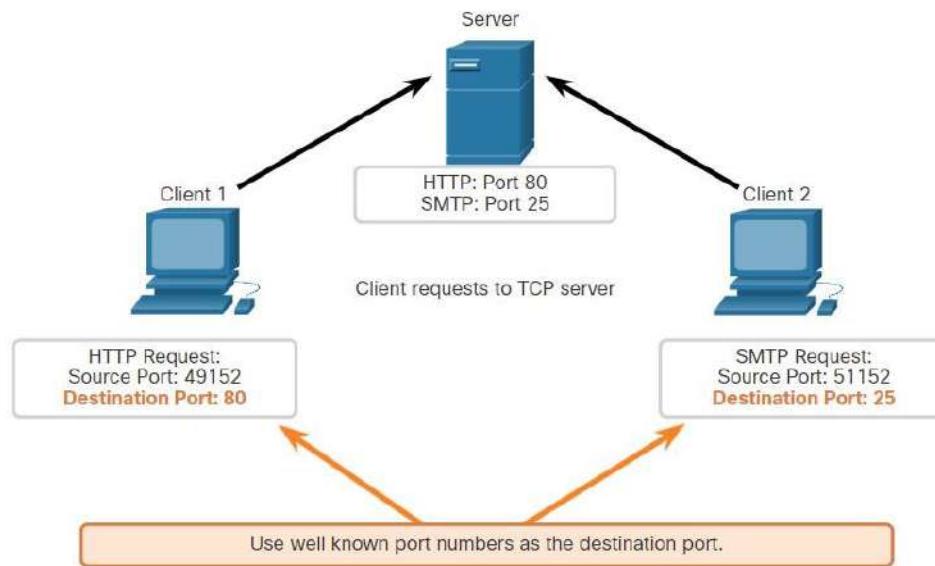
Client Mengirim TCP Request

Client 1 meminta layanan web dan Client 2 meminta layanan email dengan request yang sama.



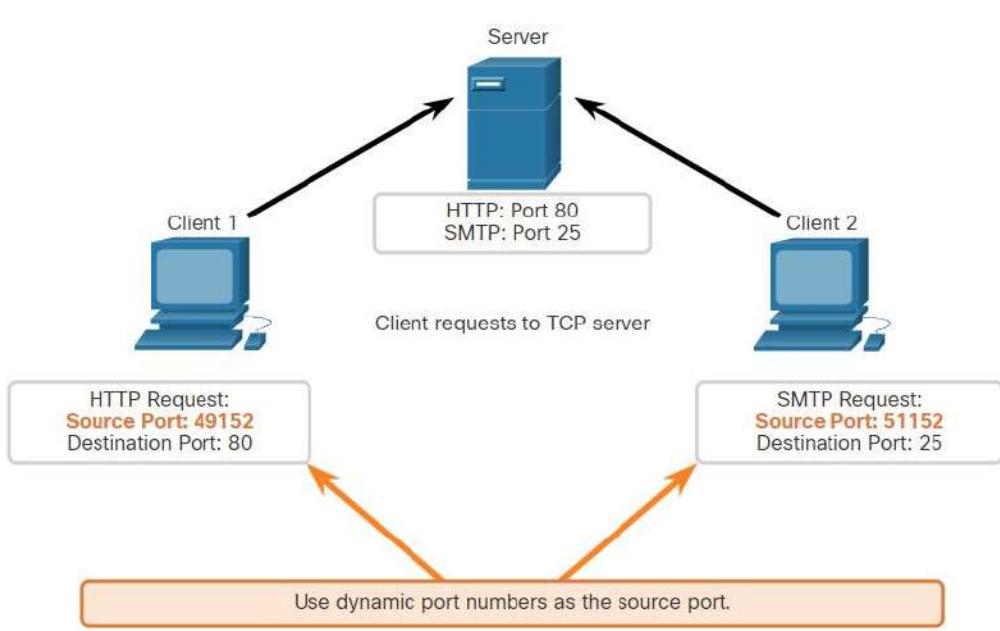
Request Destination Port

Client 1 meminta layanan web menggunakan well-known port destination 80 (HTTP) dan Client 2 meminta layanan email menggunakan well-known port 25 (SMTP).



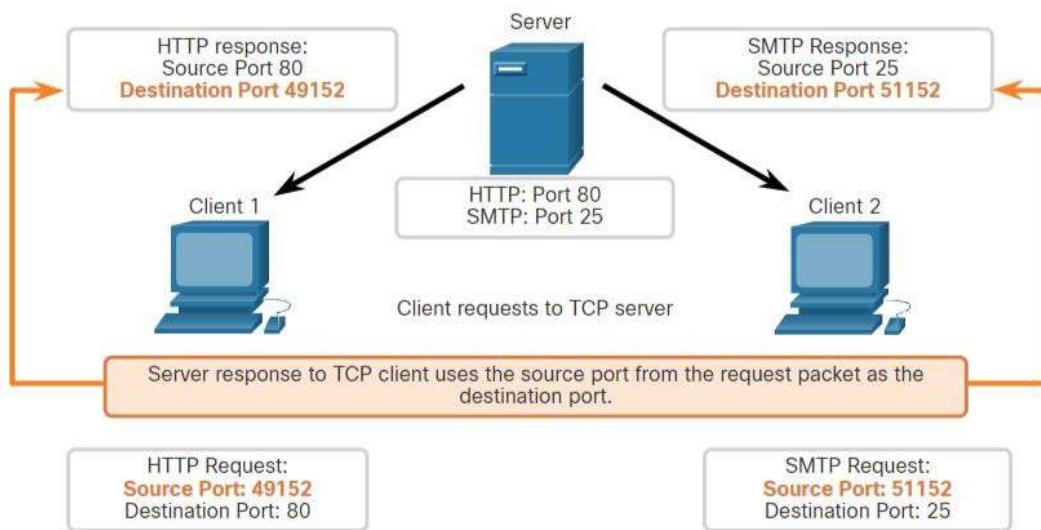
Request Source Port

Permintaan klien secara dinamis menghasilkan nomor **source port**. Dalam hal ini, Klien 1 menggunakan **source port 49152** dan Klien 2 menggunakan **source port 51152**.



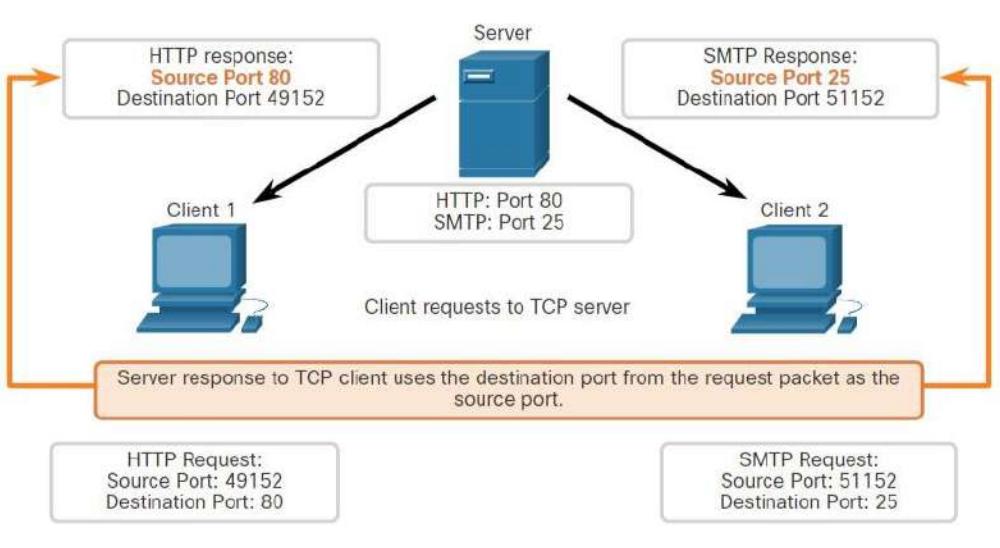
Response Destination Ports

Ketika server menanggapi permintaan klien, server akan membalikkan destination and source port permintaan awal. Perhatikan bahwa respons Server terhadap permintaan web sekarang memiliki **destination port** 49152 dan response email sekarang memiliki **destination port** 51152.



Response Source Port

Source Port dalam respons server adalah Destination port asli dalam permintaan awal.

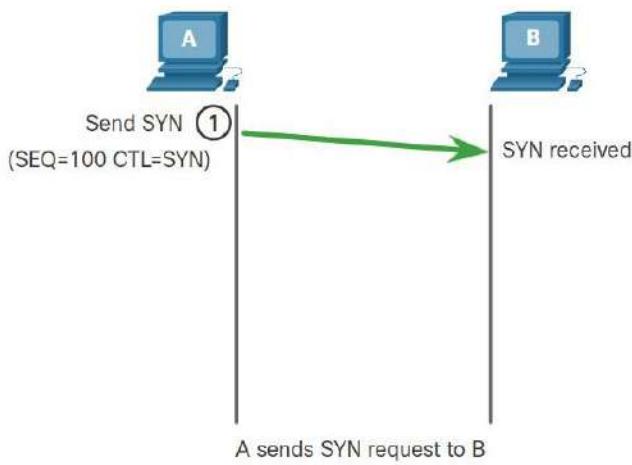


Pembentukan Koneksi TCP

Dalam beberapa budaya, ketika dua orang bertemu, mereka sering saling menyapa dengan **handshake**. Kedua belah pihak memahami tindakan **handshake** sebagai sinyal untuk sapaan. Sambungan pada jaringan serupa. Dalam koneksi TCP, klien host membuat koneksi dengan server menggunakan proses **handshake Three-Way Handshake**.

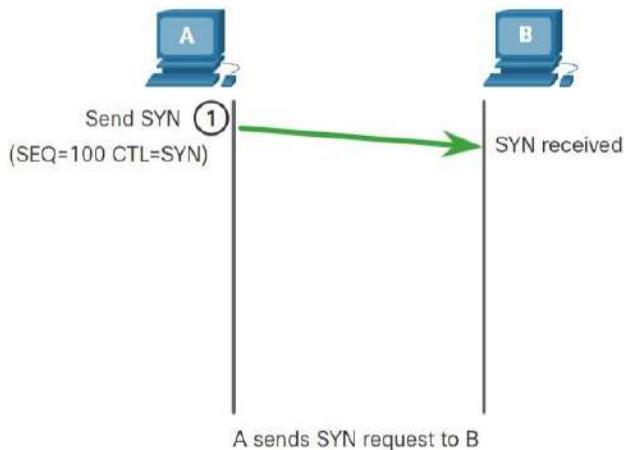
Step 1. SYN

Klien yang memulai meminta sesi komunikasi klien ke server dengan server.



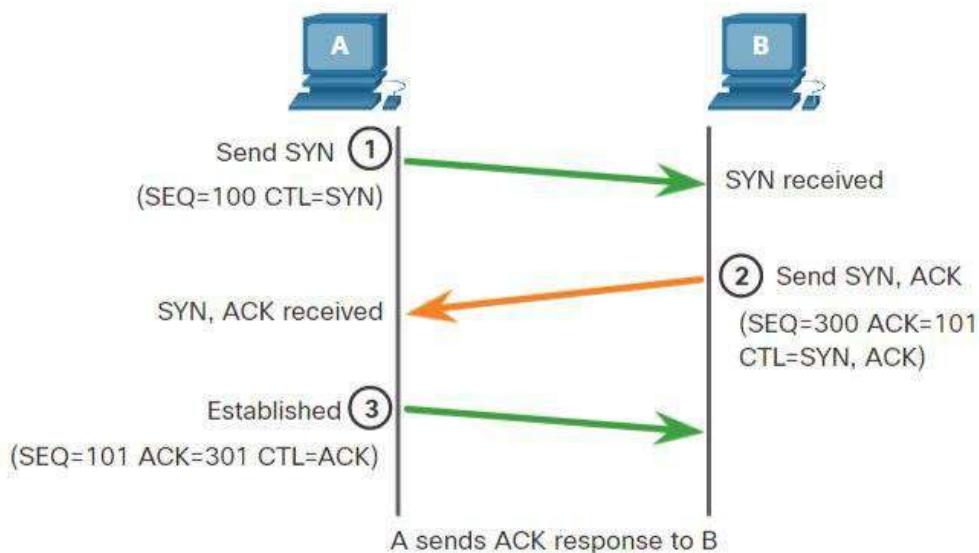
Step 2. ACK and SYN

Klien yang memulai meminta sesi komunikasi klien-ke-server dengan server.



Step 3. ACK

Klien yang memulai **Acknowledgment** sesi komunikasi server-ke-klien.



Handshake Three-Way Handshake memvalidasi bahwa host tujuan tersedia untuk berkomunikasi. Dalam contoh ini, host A telah memvalidasi bahwa host B tersedia.

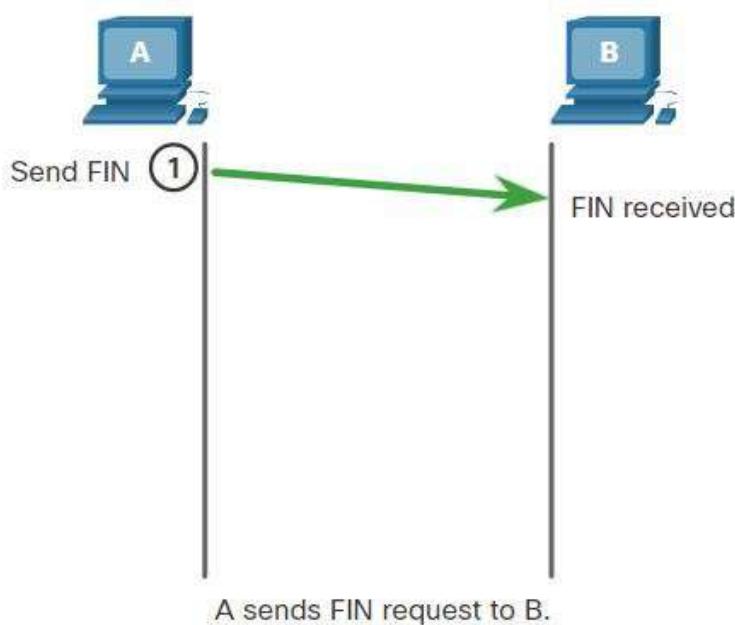
Session Termination

Untuk menutup koneksi, **Finish (FIN) Control Flag** harus diatur di header segment. Untuk mengakhiri setiap sesi TCP **one-way, two-way handshake**, yang terdiri dari segmen FIN dan segmen Acknowledgment (ACK) yang digunakan. Oleh karena itu, untuk mengakhiri satu percakapan yang didukung oleh TCP, diperlukan empat pertukaran untuk mengakhiri kedua sesi. Baik klien atau server dapat memulai penghentian.

Dalam contoh, istilah klien dan server digunakan sebagai *reference for simplicity*, tetapi setiap dua host yang memiliki sesi terbuka dapat memulai proses penghentian.

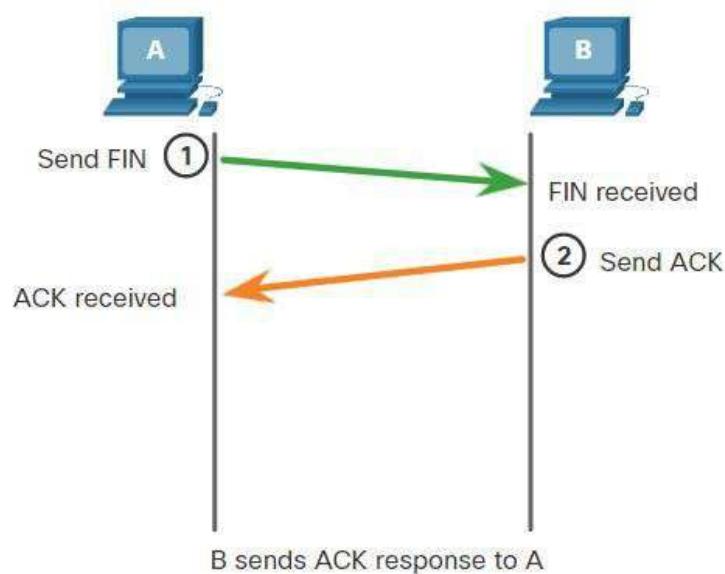
Step 1. FIN

Ketika klien tidak memiliki data lagi untuk dikirim dalam streaming, klien mengirim segmen dengan kumpulan bendera FIN.



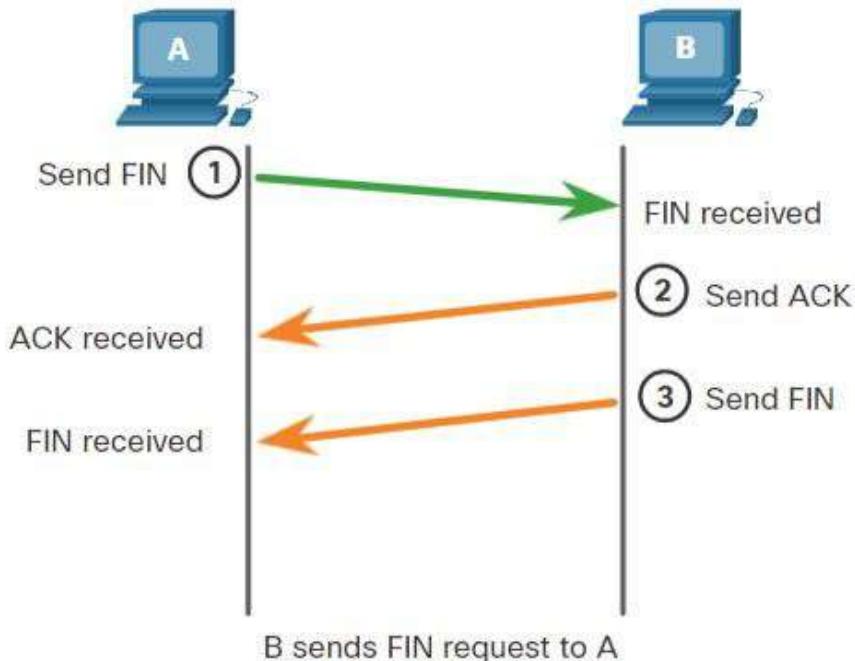
Step 2. ACK

Server mengirimkan ACK untuk **Acknowledgment** penerimaan FIN untuk mengakhiri sesi dari klien ke server.



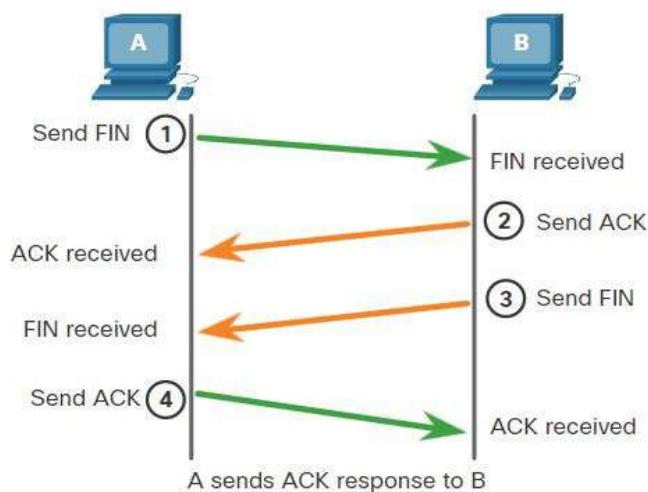
Step 3. FIN

Server mengirimkan FIN ke klien untuk mengakhiri sesi server-ke-klien.



Step 4. ACK

Klien merespons dengan ACK untuk **Acknowledgment** FIN dari server.



Ketika semua segmen telah diakui, sesi ditutup.

Analisis Handshake Three-Way Handshake TCP

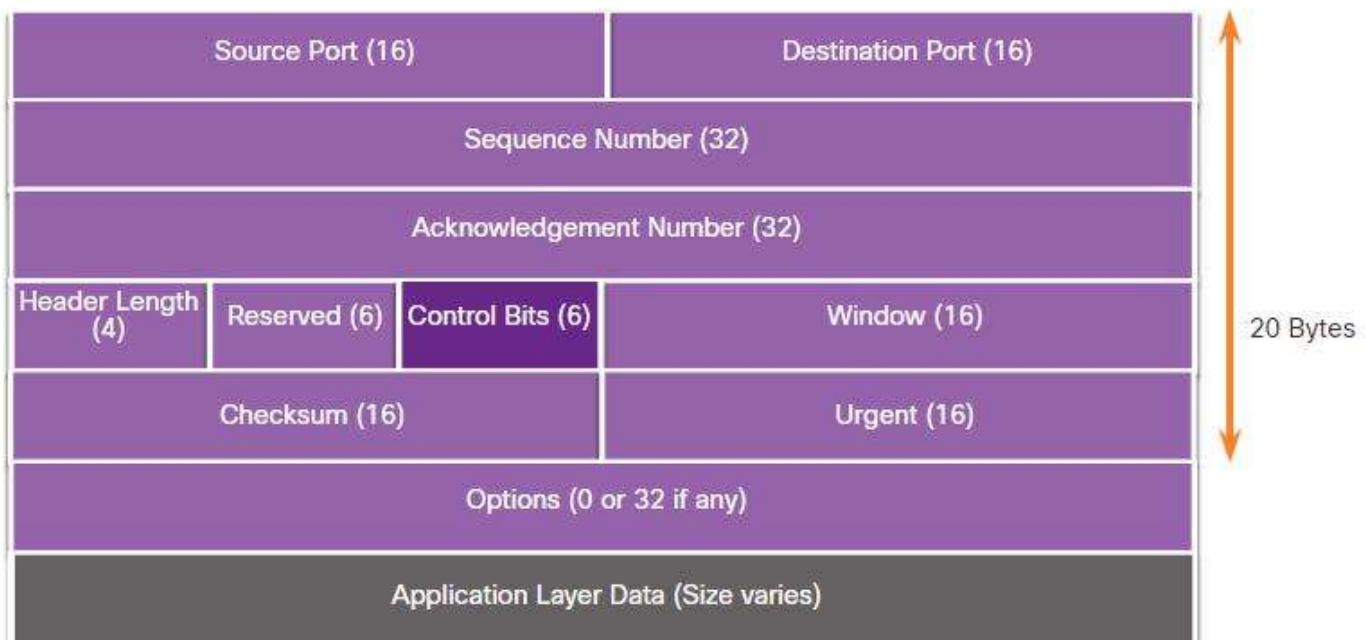
Host mempertahankan status, melacak setiap segmen data dalam sesi, dan bertukar informasi tentang data apa yang diterima menggunakan informasi di header TCP. TCP adalah protokol **full duplex**, di mana setiap koneksi mewakili dua sesi komunikasi **one-way**. Untuk membangun koneksi, **Host** melakukan **Three-Way Handshake**. Seperti yang ditunjukkan pada gambar, bit kontrol di header TCP menunjukkan kemajuan dan status koneksi.

Ini adalah fungsi dari **handshake Three-Way Handshake**:

- Ini menetapkan bahwa perangkat tujuan hadir di jaringan.
- Ini memverifikasi bahwa perangkat tujuan memiliki layanan aktif dan menerima **request** pada nomor **destination port** yang ingin digunakan klien pemulai.
- Ini menginformasikan perangkat tujuan bahwa klien sumber bermaksud untuk membuat sesi komunikasi pada nomor port tersebut.

Setelah komunikasi selesai, sesi ditutup, dan koneksi dihentikan. Mekanisme koneksi dan sesi memungkinkan fungsi keandalan TCP.

Control Bits Field



Enam bit di **Control Bits Field** dari header segmen TCP juga dikenal sebagai **Flags**. **Flags** yang sedikit yang diatur ke hidup atau nonaktif.

Enam **Control Bits Flags** adalah sebagai berikut:

- **URG** – Field penunjuk mendesak signifikan
- **ACK** – **Acknowledgment Flag** yang digunakan dalam pembentukan koneksi dan penghentian sesi
- **PSH** – Fungsi Push
- **RST** – Mengatur ulang koneksi saat terjadi kesalahan atau waktu habis
- **SYN** – Menyinkronkan nomor urutan yang digunakan dalam pembentukan koneksi
- **FIN** – Tidak ada lagi data dari pengirim dan digunakan dalam penghentian sesi

Cari di internet untuk mempelajari lebih lanjut tentang PSH dan URG Flags.

Reliability dan Flow Control

Alasan bahwa TCP adalah protokol yang lebih baik untuk beberapa aplikasi adalah karena, tidak seperti UDP, ia mengirim ulang paket yang di drop dan memberi angka pada paket untuk menunjukkan pesanan yang tepat sebelum pengiriman. TCP juga dapat membantu menjaga flow paket sehingga perangkat tidak menjadi kelebihan beban. materi ini mencakup fitur-fitur TCP secara rinci.

Reliability TCP – Pengiriman yang Dijamin dan Dipesan

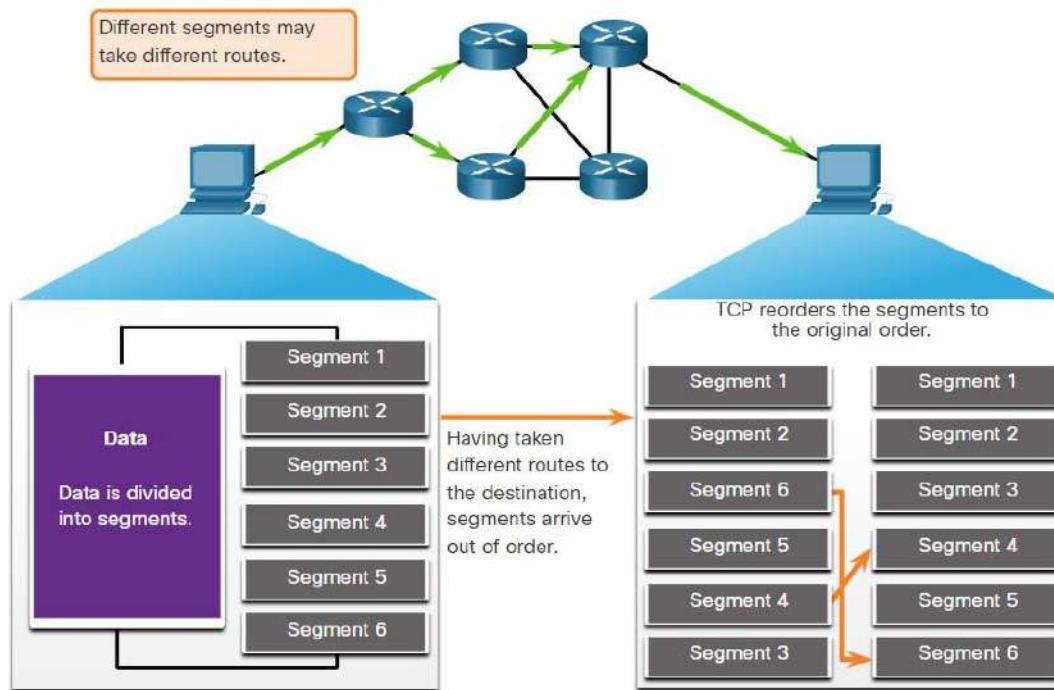
Mungkin ada saat-saat ketika segmen TCP tidak sampai di tujuan mereka. Di lain waktu, segmen TCP mungkin keluar dari urutan. Agar pesan asli dipahami oleh penerima, semua data harus diterima dan data dalam segmen ini harus dipasang kembali ke dalam urutan asli. Nomor urut ditetapkan dalam header setiap paket untuk mencapai tujuan. Nomor urut mewakili byte data pertama dari segmen TCP.

Selama pengaturan sesi, **initial sequence number** (ISN) diatur. ISN ini mewakili nilai awal byte yang ditransmisikan ke aplikasi penerima. Sebagai data ditransmisikan selama sesi, nomor urut meningkat dengan jumlah byte yang telah ditransmisikan. Pelacakan byte data ini memungkinkan setiap segmen diidentifikasi dan diakui secara unik. Segmen yang hilang kemudian dapat diidentifikasi.

ISN tidak dimulai pada angka satu tetapi secara efektif merupakan angka acak. Hal ini untuk mencegah beberapa jenis serangan berbahaya. sederhananya, kita akan menggunakan ISN 1 untuk contoh dalam bab ini.

Nomor urutan segmen menunjukkan cara memasang kembali dan menyusun ulang segmen yang diterima, seperti yang ditunjukkan pada gambar.

Segmen TCP Disusun Ulang di Tujuan



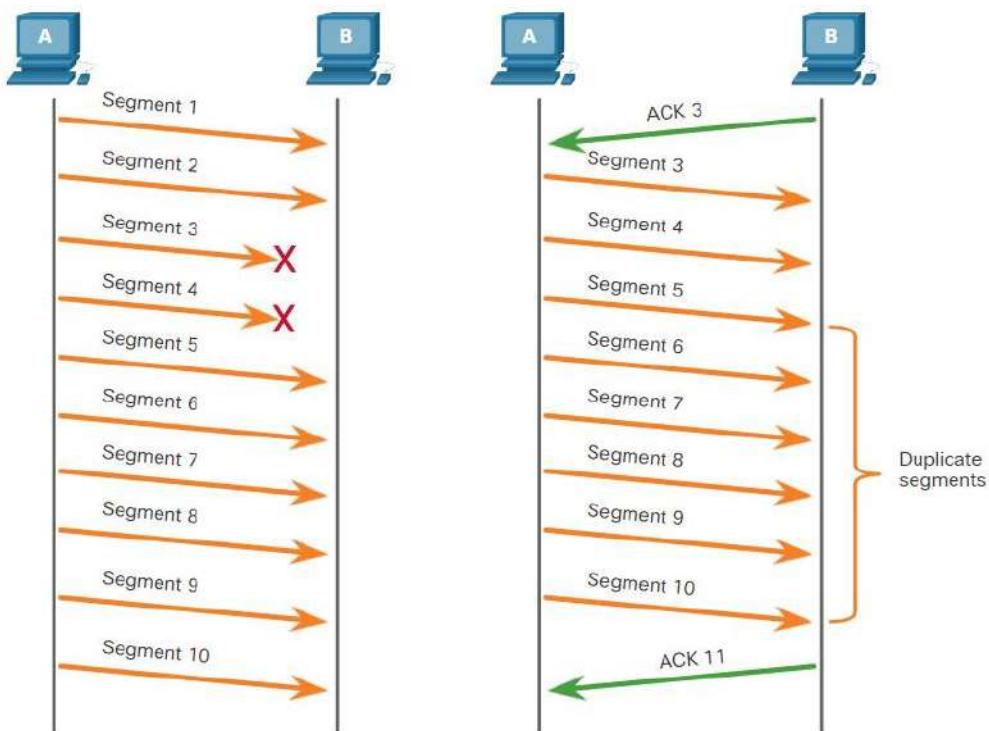
Proses TCP penerima menampilkan data dari segmen ke buffer penerima. Segmen kemudian ditempatkan dalam urutan urutan yang tepat dan diteruskan ke lapisan aplikasi ketika dipasang kembali. Setiap segmen yang datang dengan nomor urut yang tidak teratur diadakan untuk diproses nanti. Kemudian, ketika segmen dengan byte yang hilang tiba, segmen ini diproses secara berurutan.

Reliability TCP – Data Loss and Transmisi Ulang

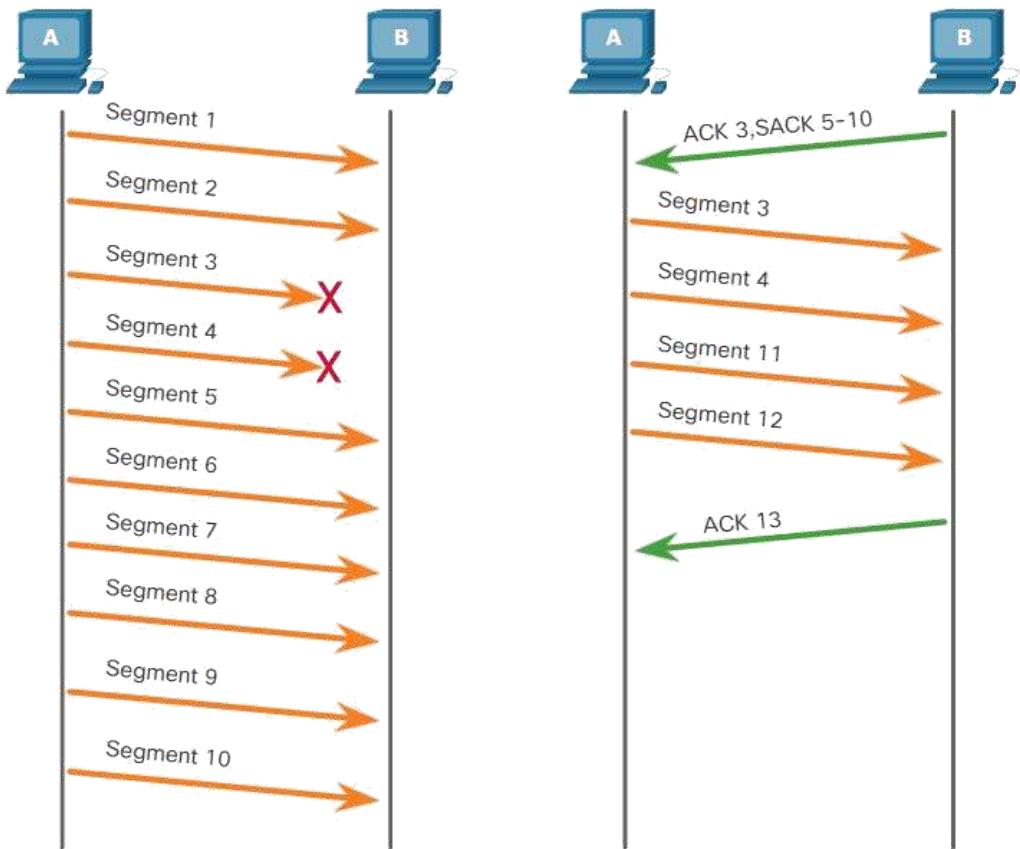
Tidak peduli seberapa baik merancang jaringan, kehilangan data kadang-kadang terjadi. TCP menyediakan metode untuk mengelola kerugian segmen ini. Di antaranya adalah mekanisme untuk mengirimkan kembali segmen untuk data yang **unacknowledgment**.

Nomor Sequence (SEQ) dan Acknowledgement (ACK) digunakan bersama untuk mengkonfirmasi penerimaan byte data yang terkandung dalam segmen yang ditransmisikan. Nomor SEQ mengidentifikasi byte data pertama di segmen yang ditransmisikan. TCP menggunakan nomor ACK yang dikirim kembali ke sumber untuk menunjukkan byte berikutnya yang diharapkan penerima untuk menerima. Ini disebut **expectational acknowledgment**.

Sebelum perangkat tambahan kemudian, TCP hanya bisa mengakui byte berikutnya yang diharapkan. Misalnya, dalam gambar, menggunakan nomor segmen untuk menyederhanakan, host A mengirimkan segmen 1 sampai 10 untuk host B. Jika semua segmen tiba kecuali untuk segmen 3 dan 4, host B akan menjawab dengan **Acknowledgment** yang menentukan bahwa segmen berikutnya yang diharapkan adalah segmen 3. Host A tidak tahu apakah ada segmen lain yang tiba atau tidak. Host A akan, oleh karena itu, mengirim ulang segmen 3 hingga 10. Jika semua segmen yang dikirim kembali berhasil tiba, segmen 5 hingga 10 akan menjadi duplikat. Hal ini dapat menyebabkan **Delay**, **Congestion**, dan **inefisiensi**.



Sistem operasi host saat ini biasanya menggunakan fitur TCP opsional yang disebut selective acknowledgement (SACK), dinegosiasi selama **three way handshake**. Jika kedua host mendukung SACK, penerima dapat secara eksplisit mengakui segmen mana (byte) yang diterima termasuk segmen yang terputus-putus. Oleh karena itu, host pengirim hanya perlu mengirimkan kembali data yang hilang. Misalnya, pada gambar berikutnya, sekali lagi menggunakan nomor segmen untuk kesederhanaan, host A mengirimkan segmen 1 hingga 10 untuk menjadi host B. Jika semua segmen tiba kecuali untuk segmen 3 dan 4, host B dapat mengakui bahwa ia telah menerima segmen 1 dan 2 (ACK 3), dan secara selektif mengakui segmen 5 sampai 10 (SACK 5-10). Host A hanya perlu mengirim ulang segmen 3 dan 4.



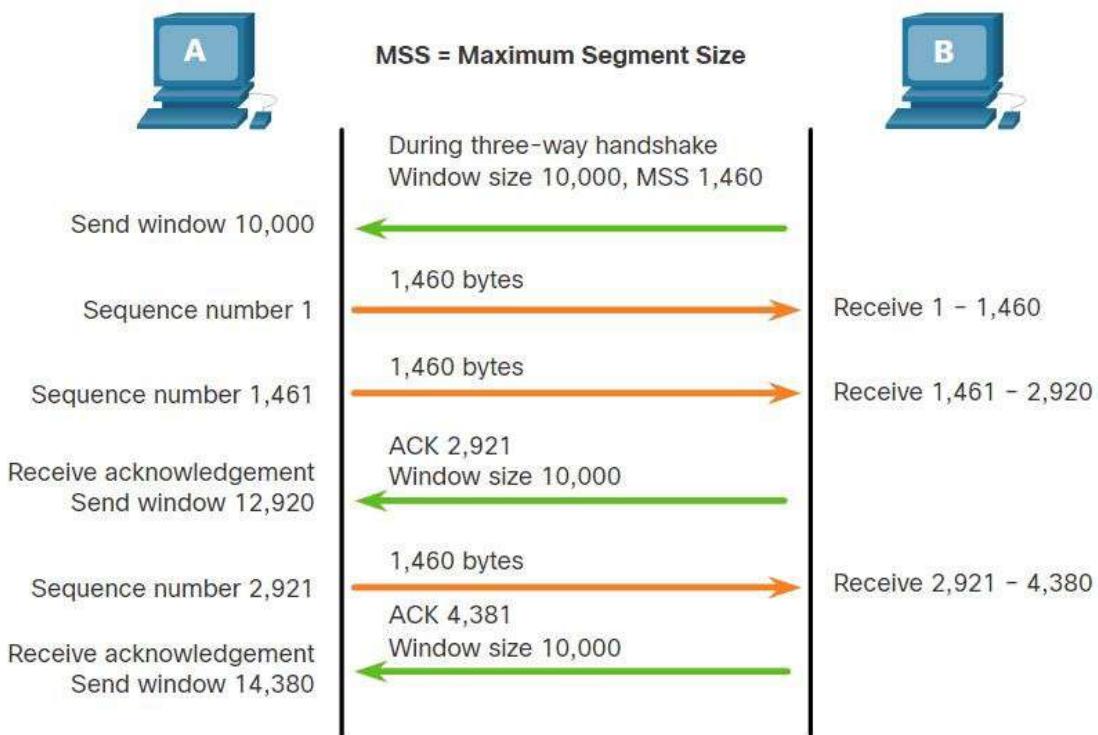
Catatan: TCP biasanya mengirimkan ACK untuk setiap paket lainnya, tetapi faktor-faktor lain di luar lingkup materi ini dapat mengubah perilaku ini.

TCP Flow Control – Window size dan Acknowledgment

TCP juga menyediakan mekanisme untuk **flow control**. Flow control adalah jumlah data yang dapat diterima dan diproses oleh tujuan dengan handal/reliability. **Flow control** membantu menjaga **Reliability** transmisi TCP dengan menyesuaikan laju stream data antara sumber dan tujuan untuk sesi tertentu. Untuk mencapai hal ini, header TCP mencakup field 16-bit yang disebut **window size**.

Angka tersebut menunjukkan contoh window size dan acknowledgement.

Contoh Window size TCP



Window size menentukan jumlah byte yang dapat dikirim sebelum mengharapkan **acknowledgment**. Nomor **acknowledgment** adalah jumlah byte yang diharapkan berikutnya.

Window size adalah jumlah byte yang dapat diterima dan diproses oleh perangkat tujuan sesi TCP pada satu waktu. Dalam contoh ini, **window size** awal PC B untuk sesi TCP adalah 10.000 byte. Dimulai dengan byte pertama, byte nomor 1, byte terakhir PC A dapat mengirim tanpa menerima **acknowledgment** adalah byte 10.000. Ini dikenal sebagai **sent window** PC A. **Window size** termasuk dalam setiap segmen TCP sehingga tujuan dapat memodifikasi **window size** setiap saat tergantung pada ketersediaan buffer.

Window size awal disepakati ketika sesi TCP ditetapkan selama **three way handshake**. Perangkat sumber harus membatasi jumlah byte yang dikirim ke perangkat tujuan berdasarkan **window size** tujuan. Hanya setelah perangkat sumber menerima **acknowledgment** bahwa byte telah diterima, dapat terus mengirim lebih banyak data untuk sesi tersebut. Biasanya, tujuan tidak akan menunggu semua byte untuk **window size** yang akan diterima sebelum membala dengan **acknowledgment**. Ketika byte diterima dan diproses, tujuan akan mengirim **acknowledgment** untuk menginformasikan sumber bahwa ia dapat terus mengirim byte tambahan.

Sebagai contoh, adalah khas bahwa PC B tidak akan menunggu sampai semua 10.000 byte telah diterima sebelum mengirim **acknowledgment**. Ini berarti PC A dapat menyesuaikan **window** pengirimannya karena menerima **acknowledgment** dari PC B. Seperti yang ditunjukkan pada gambar, ketika PC A menerima **acknowledgment** dengan nomor **acknowledgment** 2.921, yang merupakan byte yang diharapkan berikutnya. PC A mengirim **window** akan bertambah 2.920 byte. Ini mengubah **window** kirim dari 10.000 byte menjadi 12.920. PC A sekarang dapat terus mengirim hingga 10.000 byte ke PC B selama tidak mengirim lebih dari **window** pengiriman baru di 12.920.

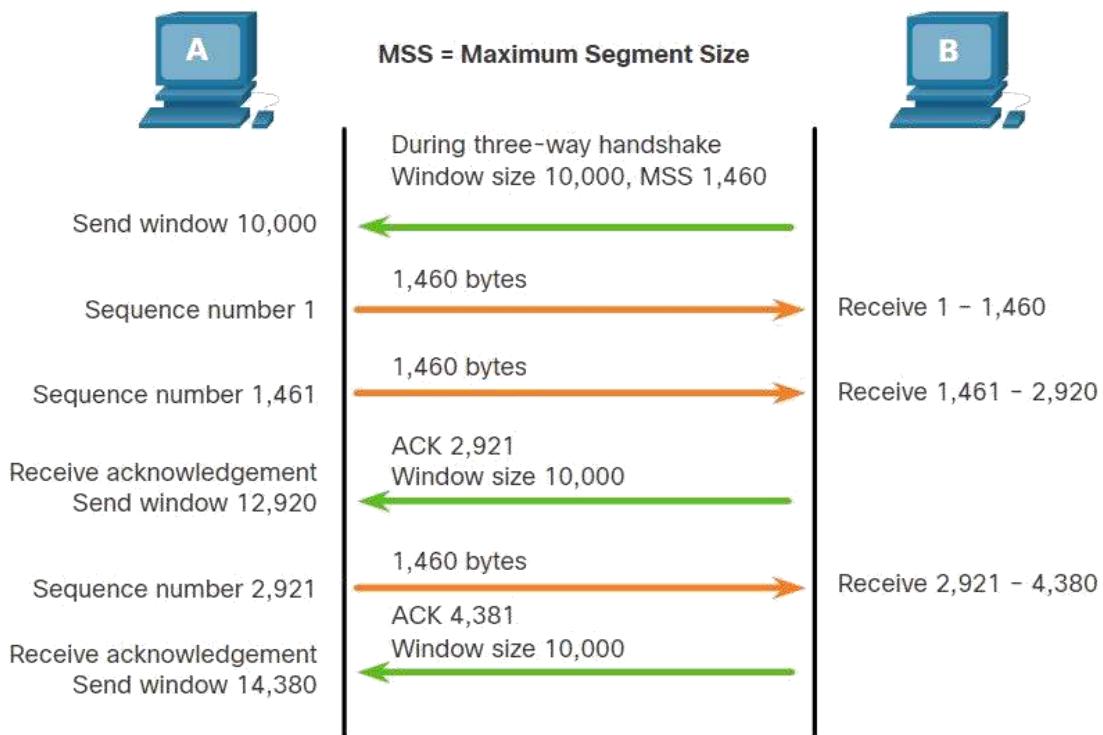
Tujuan mengirim **acknowledgment** karena memproses byte yang diterima, dan penyesuaian terus-menerus dari **window** pengiriman sumber, dikenal sebagai **sliding window**. Pada contoh sebelumnya, **window** kirim PC A bertambah atau meluncur di atas 2.921 byte lainnya dari 10.000 menjadi 12.920.

Jika ketersediaan ruang penyanga tujuan menurun, itu dapat mengurangi **window size**nya untuk menginformasikan sumber untuk mengurangi jumlah byte yang harus dikirim tanpa menerima **acknowledgment**.

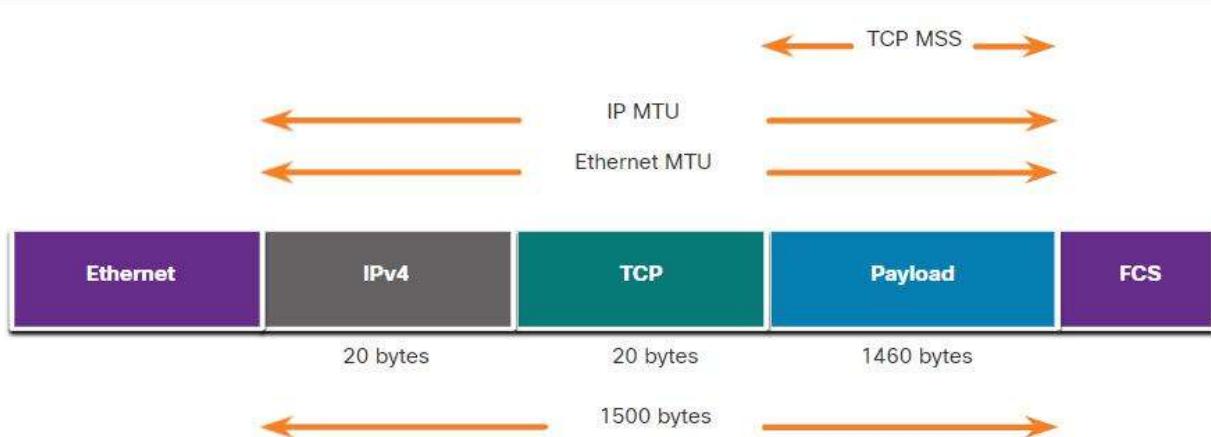
Catatan: Perangkat saat ini menggunakan protokol **sliding window**. Penerima biasanya mengirimkan **acknowledgment** setelah setiap dua segmen yang diterimanya. Jumlah segmen yang diterima sebelum diakui dapat bervariasi. Keuntungan dari **sliding window** adalah memungkinkan pengirim untuk terus mengirimkan segmen, selama penerima mengakui segmen sebelumnya. Rincian **sliding window** berada di luar lingkup materi ini.

Kontrol Aliran TCP – Ukuran Segmen Maksimum (MSS)

Dalam gambar, sumber mentransmisikan 1.460 byte data dalam setiap segmen TCP. Ini biasanya **Maximum Segment Size (MSS)** yang dapat diterima perangkat tujuan. MSS adalah bagian dari field opsi di header TCP yang menentukan jumlah data terbesar, dalam byte, yang dapat diterima perangkat dalam satu segmen TCP. Ukuran MSS tidak termasuk header TCP. MSS biasanya disertakan selama **three way handshake**.



MSS umum adalah 1.460 byte saat menggunakan IPv4. Host menentukan nilai bidang MSS-nya dengan mengurangi header IP dan TCP dari unit transmisi maksimum Ethernet (MTU). Pada antarmuka Ethernet, MTU default adalah 1500 byte. Mengurangi header IPv4 dari 20 byte dan header TCP dari 20 byte, ukuran MSS default akan menjadi 1460 byte, seperti yang ditunjukkan pada gambar.



Flow Control TCP – Congestion Avoidance

Ketika kemacetan terjadi pada jaringan, itu menghasilkan paket yang dibuang oleh router yang kelebihan beban. Ketika paket yang berisi segmen TCP tidak mencapai tujuan mereka, mereka dibiarkan **unacknowledgment**. Dengan menentukan tingkat dimana segmen TCP dikirim tetapi **unacknowledgment**, sumber dapat mengasumsikan tingkat tertentu **congestion** jaringan.

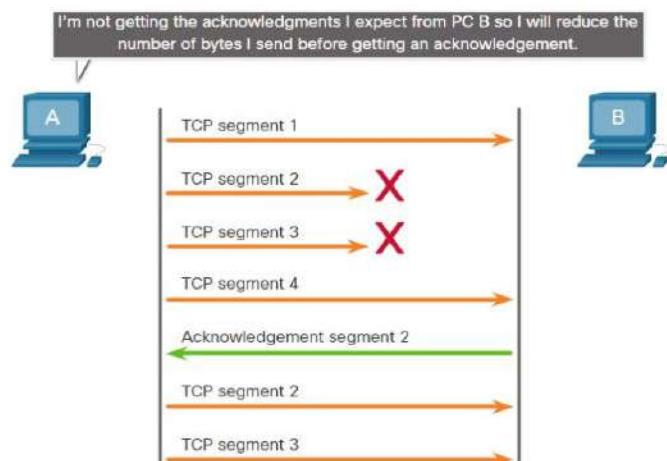
Setiap kali ada kemacetan, transmisi ulang segmen TCP yang hilang dari sumber akan terjadi. Jika transmisi ulang tidak dikontrol dengan benar, transmisi ulang tambahan dari segmen TCP dapat membuat kemacetan menjadi lebih buruk. Tidak hanya paket baru dengan segmen TCP yang diperkenalkan ke dalam jaringan, tetapi efek umpan balik dari segmen TCP yang dikirim kembali yang hilang juga akan menambah **congestion**. Untuk menghindari dan mengendalikan **congestion**, TCP menggunakan beberapa mekanisme penanganan **congestion**, **timer**, dan **algoritma**.

Jika sumber menentukan bahwa segmen TCP **unacknowledgment** atau **unacknowledgment** pada waktu yang tepat, maka dapat mengurangi jumlah byte yang dikirim sebelum menerima **acknowledgment**. Seperti yang diilustrasikan dalam gambar, PC A merasakan ada **congestion** dan oleh karena itu, mengurangi jumlah byte yang dikirimnya sebelum menerima **acknowledgment** dari PC B.

TCP Congestion Control

Angka **acknowledgment** adalah untuk byte yang diharapkan berikutnya dan bukan untuk segmen. Nomor segmen yang digunakan disederhanakan untuk tujuan ilustrasi.

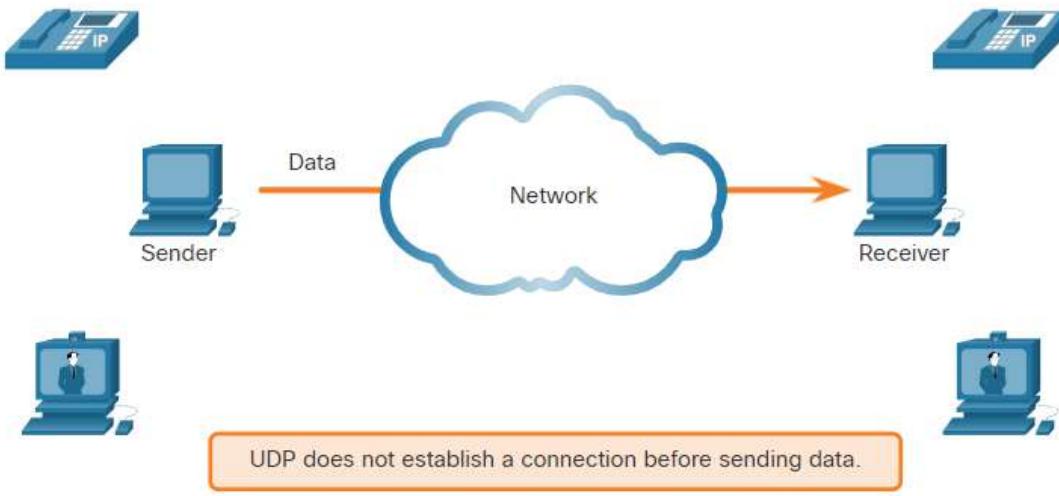
Perhatikan bahwa itu adalah sumber yang mengurangi jumlah byte yang **unacknowledgment** yang dikirimnya dan bukan **window size** yang ditentukan oleh tujuan.



Catatan: Penjelasan mekanisme penanganan actual **congestion**, **timer**, dan **algoritma** berada di luar lingkup materi ini.

Komunikasi UDP

Seperti yang dijelaskan sebelumnya, UDP sangat cocok untuk komunikasi yang perlu cepat, seperti VoIP. Topik ini menjelaskan secara rinci mengapa UDP sangat cocok untuk beberapa jenis transmisi. Seperti yang ditunjukkan pada gambar, UDP tidak membuat koneksi. UDP menyediakan transportasi data Low Overhead karena memiliki header datagram kecil dan tidak ada traffic manajemen jaringan.



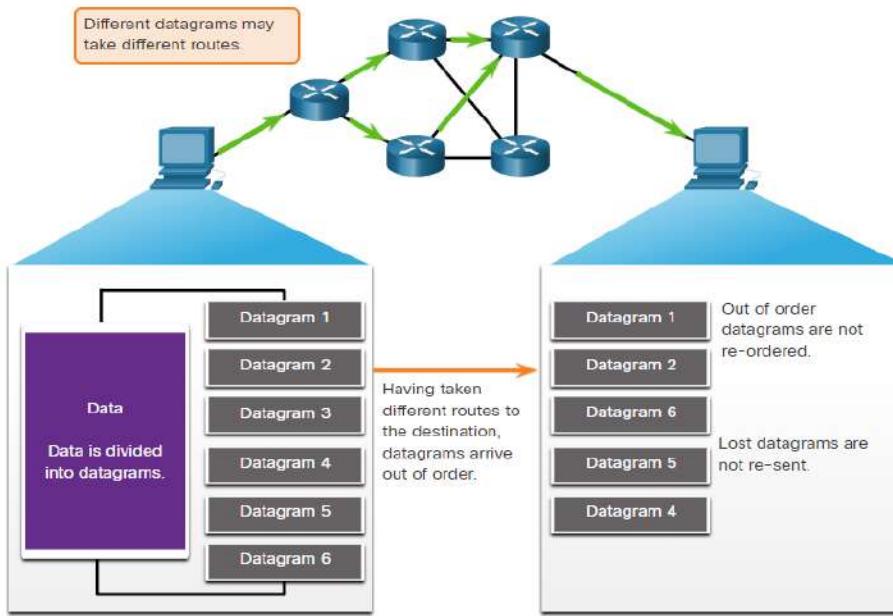
UDP Datagram Reassembly

Seperti segmen dengan TCP, ketika datagram UDP dikirim ke tujuan, mereka sering mengambil jalur yang berbeda dan tiba dalam urutan yang salah. UDP tidak melacak nomor urut seperti TCP. UDP tidak memiliki cara untuk menyusun ulang datagram ke dalam urutan transmisi mereka, seperti yang ditunjukkan pada gambar.

Oleh karena itu, UDP hanya memasang kembali data agar diterima dan meneruskannya ke aplikasi. Jika urutan data penting untuk aplikasi, aplikasi harus mengidentifikasi urutan yang tepat dan menentukan bagaimana data harus diproses.

menunjukkan datagram UDP dikirim secara berurutan tetapi tiba tidak teratur karena kemungkinan rute yang berbeda untuk mencapai tujuan

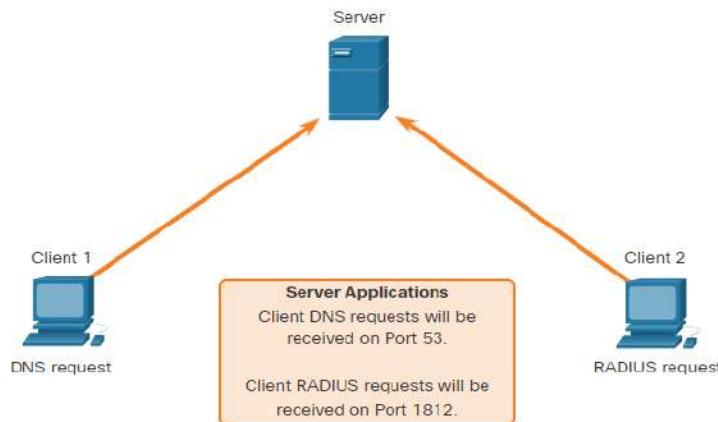
UDP: Connectionless dan Tidak Dapat Diandalkan



Proses dan Permintaan Server UDP

Seperti aplikasi berbasis TCP, aplikasi server berbasis UDP diberi nomor Well-Known Port atau registered, seperti yang ditunjukkan pada gambar. Ketika aplikasi atau proses ini berjalan di server, mereka menerima data yang dicocokkan dengan nomor port yang ditetapkan. Ketika UDP menerima datagram yang ditujukan untuk salah satu port ini, ia meneruskan data aplikasi ke aplikasi yang sesuai berdasarkan nomornya.

Server UDP Mendengarkan Permintaan

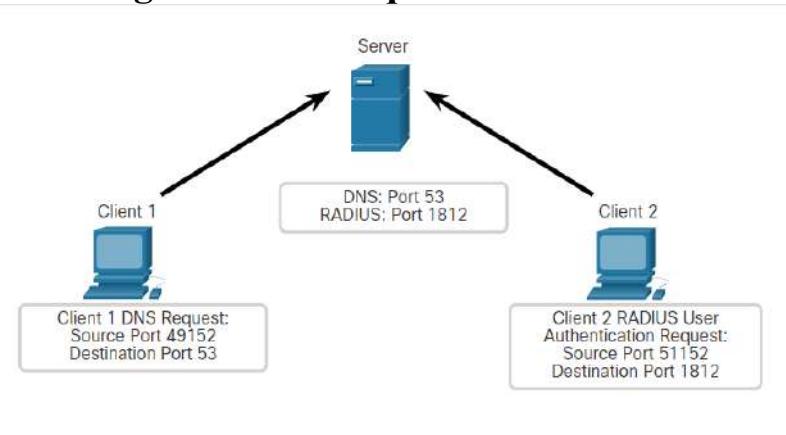


Nota: Server Remote Authentication Dial-in User Service (RADIUS) yang ditampilkan dalam gambar menyediakan layanan otentikasi, otorisasi, dan akuntansi untuk mengelola akses pengguna. Pengoperasian RADIUS berada di luar cakupan untuk materi ini.

Proses Client UDP

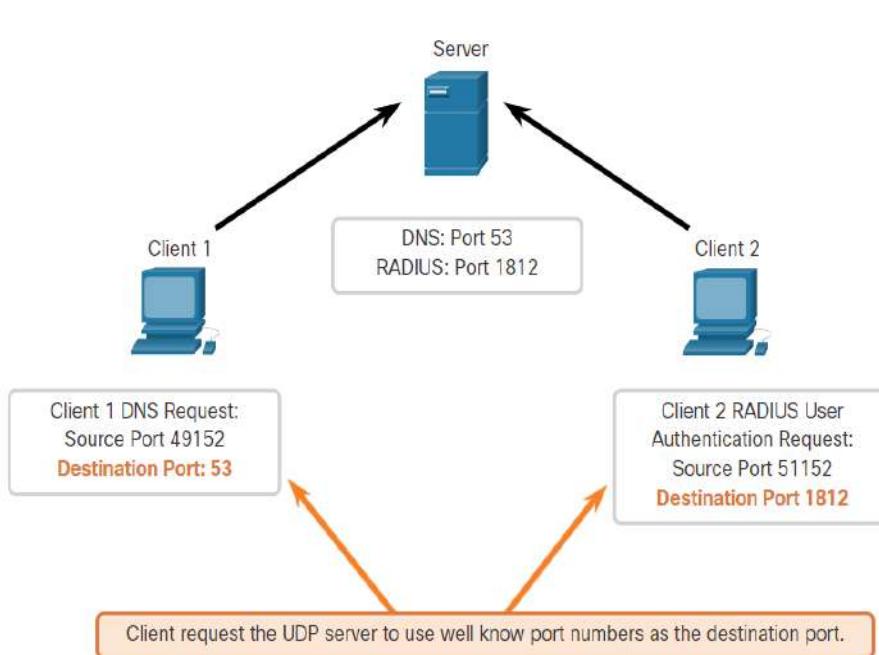
Seperti TCP, komunikasi client-server dimulai oleh aplikasi client yang meminta data dari proses server. Proses client UDP secara dinamis memilih nomor port dari rentang nomor port dan menggunakan ini sebagai source port untuk percakapan. Destination port biasanya adalah nomor Well-Know Port atau registered yang ditugaskan ke proses server. Setelah client memilih source port dan tujuan, pasangan port yang sama digunakan di header semua datagram dalam transaksi. Untuk data yang kembali ke client dari server, nomor source port dan tujuan di header datagram dibalik.

Client Mengirim UDP Request



Client 1 mengirim **DNS Request** sementara Client 2 meminta **layanan otentifikasi RADIUS** dari server yang sama.

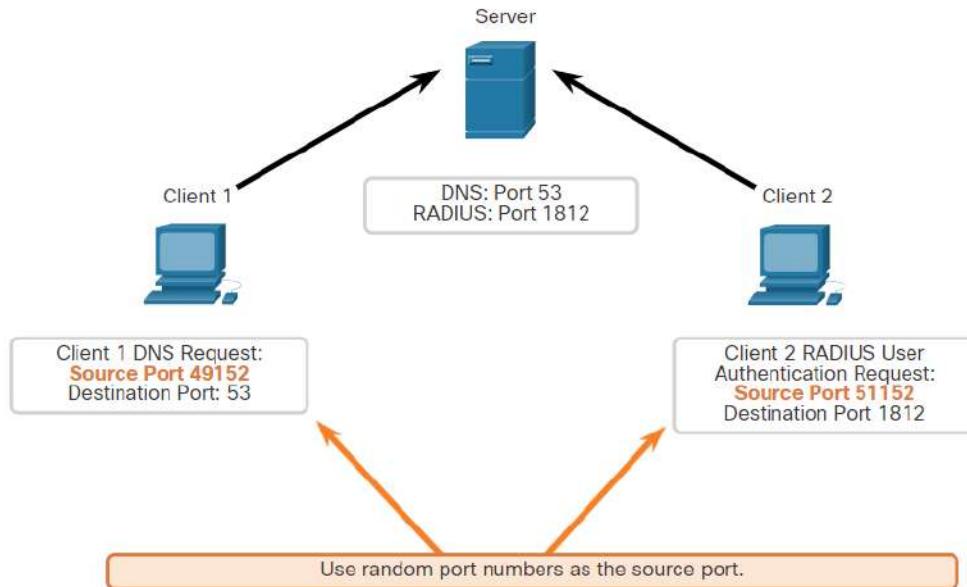
UDP Request Destination Ports



Client 1 mengirim DNS Request menggunakan destination port well-known 53 sementara Client 2 meminta layanan otentifikasi RADIUS menggunakan destination port Registered 1812.

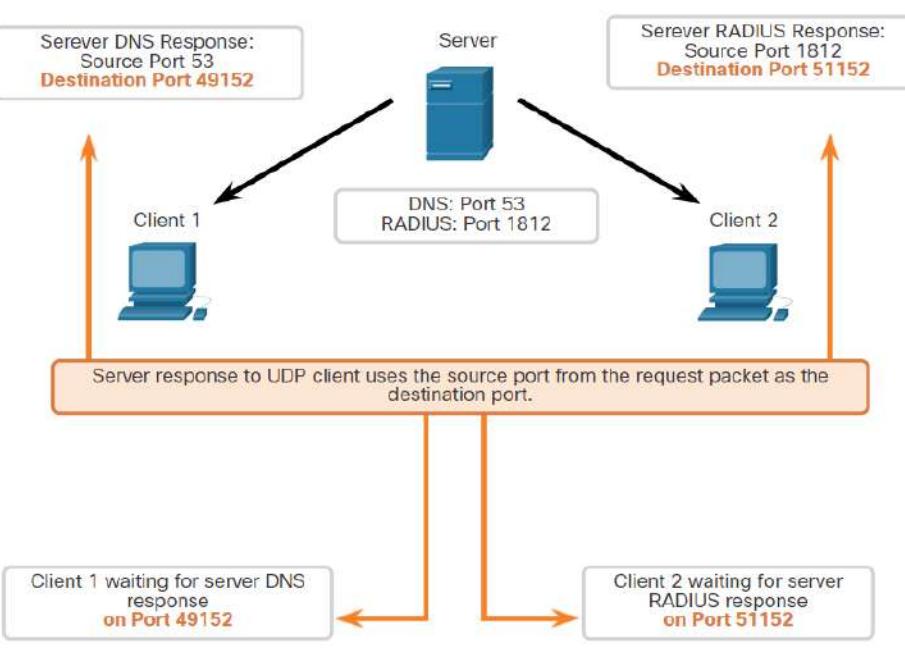
UDP Request Source Ports

Permintaan client secara dinamis menghasilkan nomor source port. Dalam hal ini, Client 1 menggunakan source port 49152 dan Client 2 menggunakan source port 51152.



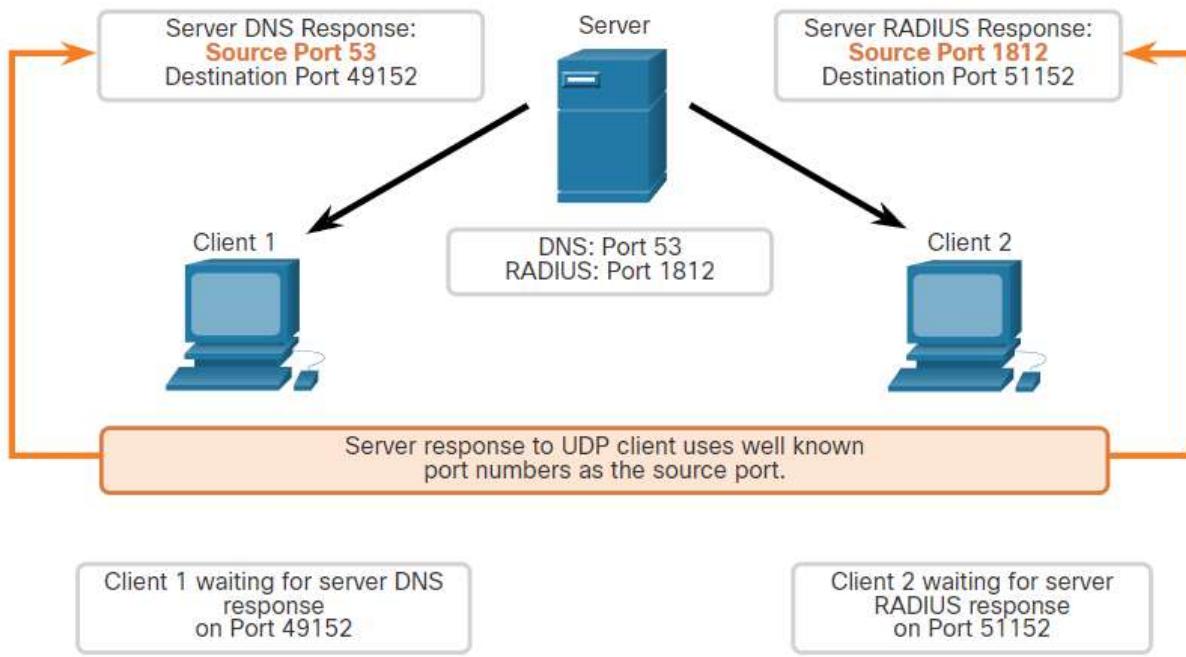
UDP Response Destination

Ketika server menanggapi permintaan klien, ia membalikkan destination port dan sumber dari permintaan awal. Dalam respons Server terhadap permintaan DNS sekarang adalah destination port 49152 dan respons otentikasi RADIUS sekarang menjadi destination port 51152.



UDP Response Source Ports

Source port dalam respons server adalah destination port asli dalam permintaan awal.



BAB 15

~ *Application Layer* ~

Judul Bab: Application Layer

Tujuan Bab: Jelaskan operasi protokol Application Layer dalam memberikan dukungan untuk aplikasi end user.

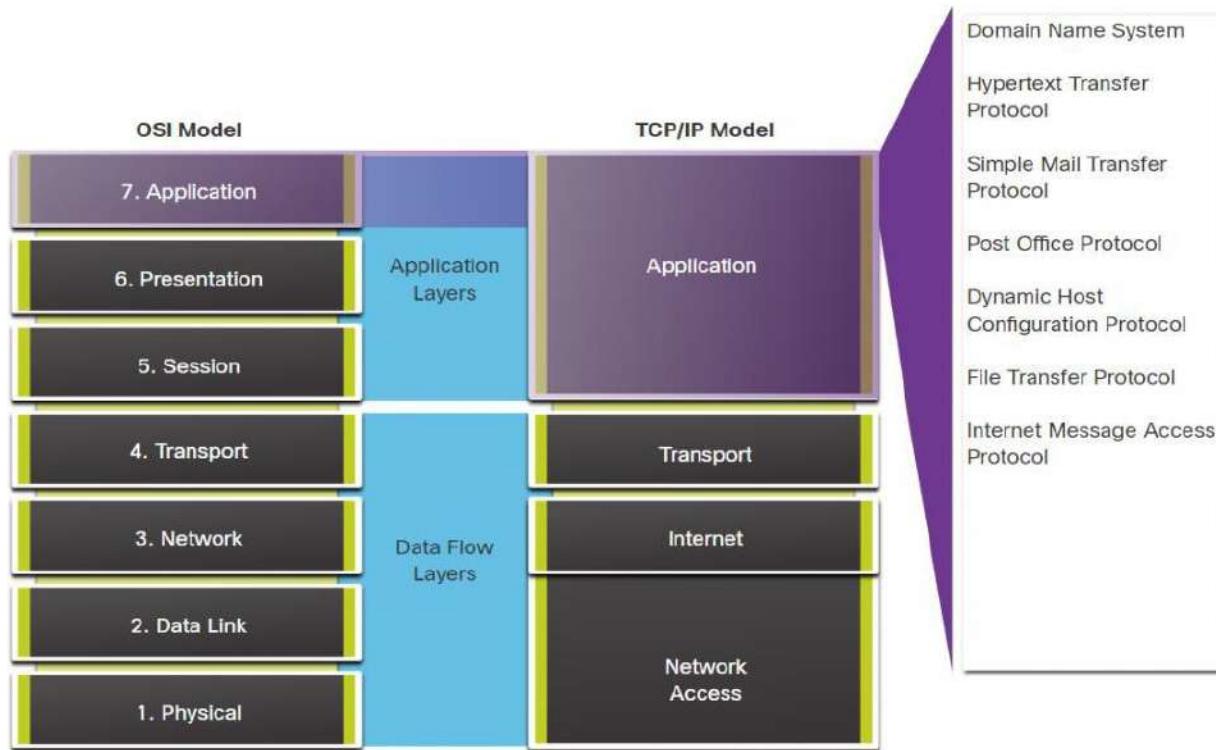
Link Test Pemahaman : <https://s.id/-Qy4X>

| Judul Materi | Tujuan Materi |
|--|---|
| Session, Presentation, Application Layer | Menjelaskan bagaimana fungsi Application layer, Presentation Layer, dan Session Layer bekerja sama untuk menyediakan layanan jaringan ke aplikasi end user. |
| Peer to Peer | Menjelaskan bagaimana aplikasi end user beroperasi di jaringan peer to peer |
| Email dan Web Protocol | Menjelaskan bagaimana email dan web protocol bekerja |
| Layanan IP Address | Menjelaskan bagaimana DNS dan DHCP beroperasi |
| File Transfer Protocol | Menjelaskan bagaimana file transfer protocols beroperasi |

Session, Presentation, Application Layer

Dalam OSI dan model TCP / IP, **application layer** adalah **layer** yang paling dekat dengan **end device**. Seperti yang ditunjukkan pada gambar, itu adalah **layer** yang menyediakan **interface** antara aplikasi yang digunakan untuk berkomunikasi, dan jaringan yang mendasari pesan yang ditransmisikan. Protokol **application layer** digunakan untuk bertukar data antara program yang berjalan di host **source** dan **destination**.

Application layer



Berdasarkan model TCP / IP, **upper layer** model OSI (Session, Presentation, Application) mendefinisikan fungsi **application layer** TCP / IP.

Ada banyak protokol **application layer**, dan protokol baru selalu dikembangkan. Beberapa protokol **application layer** yang paling banyak dikenal termasuk Hypertext Transfer Protocol (HTTP), File Transfer Protocol (FTP), Trivial File Transfer Protocol (TFTP), Internet Message Access Protocol (IMAP), dan protokol Domain Name System (DNS).

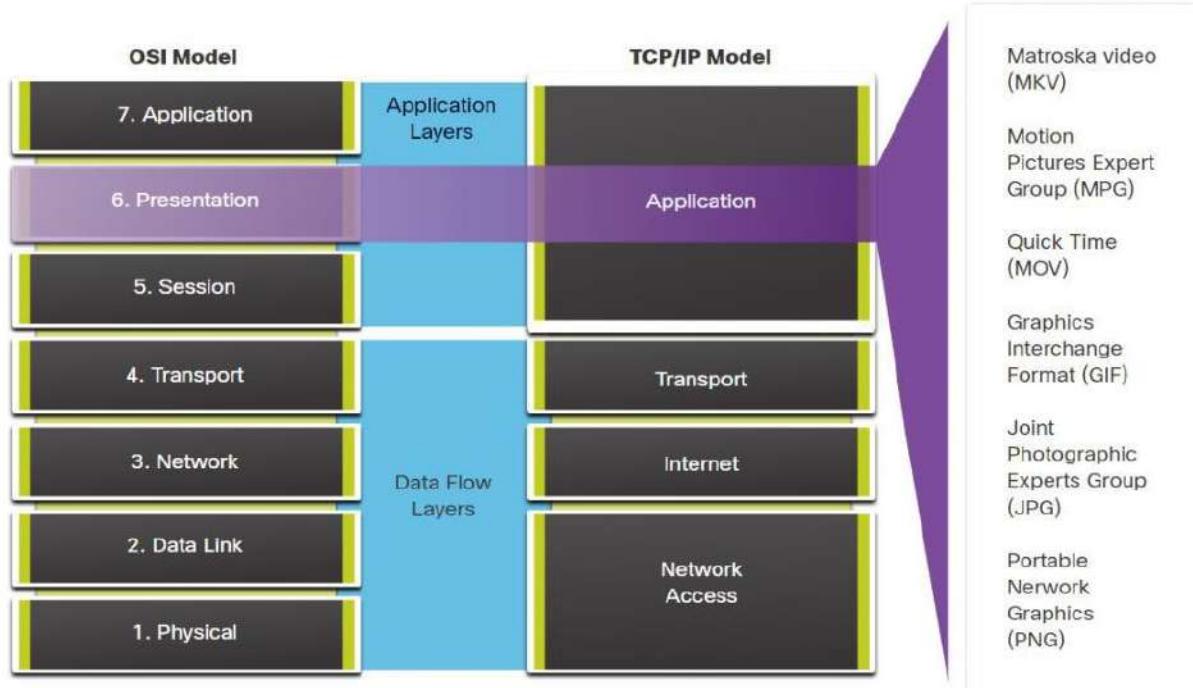
Session dan Presentation Layer

1. Presentation layer

Presentation layer memiliki tiga fungsi utama:

- Memformat, atau menyajikan, data di perangkat **source** ke dalam format yang kompatibel untuk diterima oleh perangkat **destination**.
- Mengompresi data dengan cara yang dapat didekompresi oleh perangkat **destination**.
- Mengenkripsi data untuk transmisi dan mendekripsi data saat diterima.

Seperti yang ditunjukkan pada gambar, **presentation layer** memformat data untuk **application layer**, dan menetapkan standar untuk format file. Beberapa standar terkenal untuk video termasuk Matroska Video (MKV), Motion Picture Experts Group (MPG), dan QuickTime Video (MOV). Beberapa format gambar grafis terkenal adalah Graphics Interchange Format (GIF), Joint Photographic Experts Group (JPG), dan portable network graphics (PNG).



2. Session layer

Sesuai namanya, fungsi pada **session layer** membuat dan memelihara dialog antara aplikasi **source** dan **destination**. **Session layer** menangani pertukaran informasi untuk memulai dialog, membuatnya tetap aktif, dan untuk memulai kembali sesi yang terganggu atau menganggur untuk jangka waktu yang lama.

Protokol Application layer TCP/IP

Protokol aplikasi TCP / IP menentukan format dan informasi kontrol yang diperlukan untuk banyak fungsi komunikasi internet umum. Protokol **application layer** digunakan oleh perangkat **source** dan **destination** selama sesi komunikasi. Agar komunikasi berhasil, protokol **application layer** yang diterapkan pada host **source** dan **destination** harus kompatibel.

A. Name system

DNS – Domain Name System

- TCP, klien UDP 53
- Menerjemahkan nama domain, seperti intelektualpeople.id, ke alamat IP.

B. Host Config

BOOTP – Bootstrap Protocol

- Klien UDP 68, server 67
- Memungkinkan workstation diskless untuk menemukan alamat IP sendiri, alamat IP dari server BOOTP pada jaringan, dan file yang akan dimuat ke dalam memori untuk boot mesin
- BOOTP digantikan oleh DHCP

DHCP – Dynamic Host Configuration Protocol

- Klien UDP 68, server 67
- Secara dinamis menetapkan alamat IP untuk digunakan kembali ketika tidak lagi diperlukan

C. Email

SMTP – Simple Mail Transfer Protocol

- TCP 25
- Memungkinkan klien mengirim email ke server email
- Memungkinkan server mengirim email ke server lain

POP3 – Post Office Protocol 3

- TCP 110
- Memungkinkan klien untuk mengambil email dari server email
- Mengunduh email ke aplikasi surat lokal klien

IMAP – Internet Message Access Protocol

- TCP 143
- Memungkinkan klien untuk mengakses email yang disimpan di server email
- Menyimpan email di server

D. Transfer Berkas

FTP – File Transfer Protocol

- TCP 20 sampai 21
- Menetapkan aturan yang memungkinkan pengguna pada satu host untuk mengakses dan mentransfer file ke dan dari host lain melalui jaringan
- FTP adalah protokol pengiriman file yang andal, berorientasi koneksi, dan diakui.

TFTP – Trivial File Transfer Protocol

- Klien UDP 69
- Protokol transfer file yang sederhana dan tanpa koneksi dengan pengiriman file yang terbaik dan tidak diakui
- Ini menggunakan overhead kurang dari FTP

E. Web

HTTP – Hypertext Transfer Protocol

- TCP 80, 8080
- Seperangkat aturan untuk bertukar teks, gambar grafis, suara, video, dan file multimedia lainnya di World Wide Web

HTTPS – HTTP Aman

- TCP, UDP 443
- Browser menggunakan enkripsi untuk mengamankan komunikasi HTTP
- Mengautentikasi situs web tempat Anda menghubungkan browser Anda

Peer To Peer

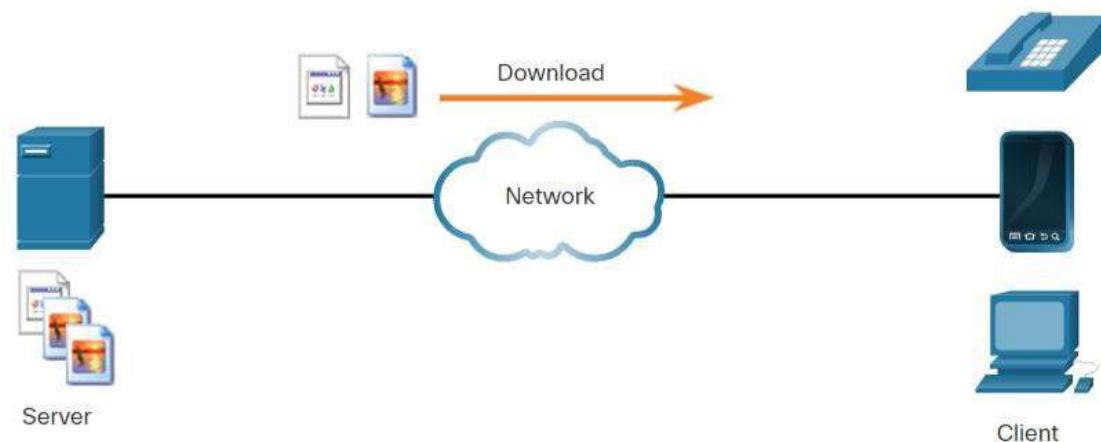
Dalam Materi sebelumnya, Anda belajar bahwa protokol **Application Layer** TCP / IP yang diimplementasikan pada host sumber dan tujuan harus kompatibel. Dalam materi ini Anda akan belajar tentang model klien / server dan proses yang digunakan, yang berada di **Application Layer**. Hal yang sama berlaku untuk jaringan peer-to-peer. Dalam model klien / server, perangkat yang meminta informasi disebut klien dan perangkat yang menanggapi **request** disebut server. Klien adalah kombinasi perangkat keras / perangkat lunak yang digunakan orang untuk langsung mengakses **resource** yang disimpan di server.

Client-Server Model

Proses klien dan server dianggap berada di **Application Layer**. Klien memulai pertukaran dengan meminta data dari server, yang merespons dengan mengirim satu atau lebih **stream** data ke klien. Protokol **Application Layer** menggambarkan format **request** dan **response** antara klien dan server. Selain transfer data yang sebenarnya, pertukaran ini mungkin juga memerlukan otentikasi pengguna dan identifikasi file data yang akan ditransfer.

Salah satu contoh jaringan klien / server adalah menggunakan layanan email ISP untuk mengirim, menerima, dan menyimpan email. Klien email di komputer rumah mengeluarkan **request** ke server email ISP untuk setiap email yang belum dibaca. Server merespons dengan mengirim email yang diminta ke klien. Transfer data dari klien ke server disebut sebagai upload dan data dari server ke klien sebagai download.

Seperti yang ditunjukkan pada gambar, file diunduh dari server ke klien.



Jaringan Peer-to-Peer

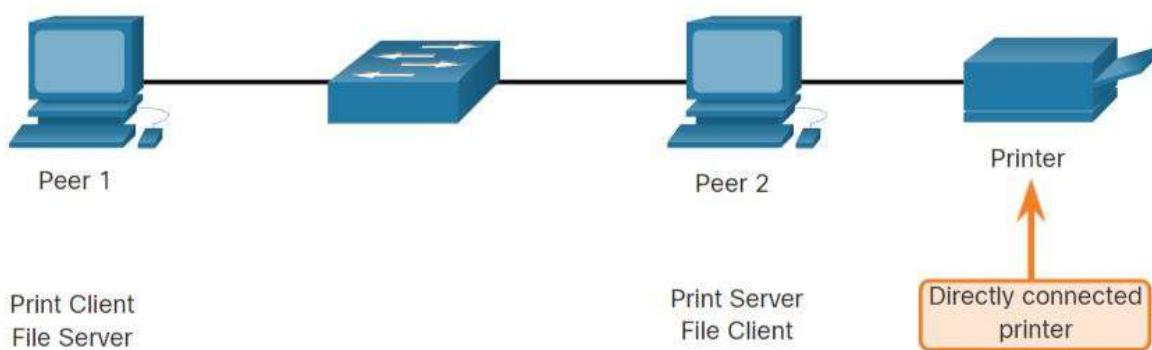
Dalam model jaringan peer-to-peer (P2P), data diakses dari perangkat peer tanpa menggunakan dedicated server.

Model jaringan P2P melibatkan dua bagian: jaringan P2P dan aplikasi P2P. Kedua bagian memiliki fitur yang sama, tetapi dalam prakteknya bekerja sangat berbeda.

Dalam jaringan P2P, dua atau lebih komputer terhubung melalui jaringan dan dapat berbagi **resource** (seperti printer dan file) tanpa memiliki dedicated server. Setiap **end devices** yang terhubung (dikenal sebagai peer) dapat berfungsi sebagai server dan klien. Satu komputer mungkin mengambil peran server untuk satu transaksi sementara secara bersamaan melayani sebagai klien untuk yang lain. Peran klien dan server diatur berdasarkan per **request**.

Selain berbagi file, jaringan seperti ini akan memungkinkan pengguna untuk mengaktifkan game jaringan atau berbagi koneksi internet.

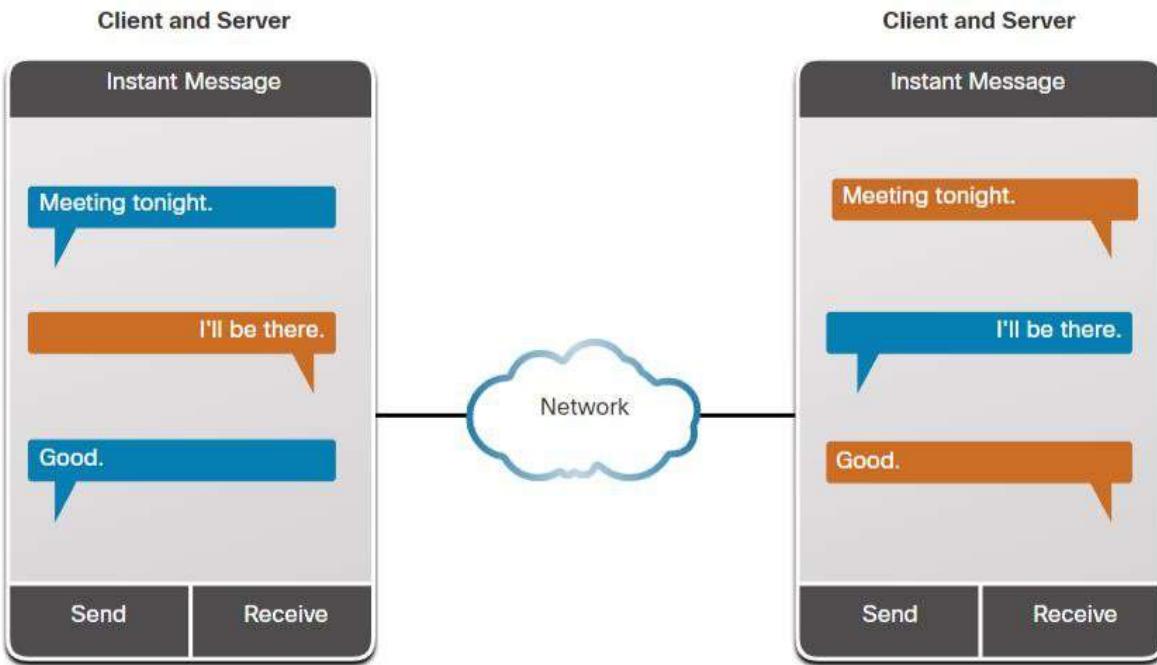
Dalam pertukaran peer-to-peer, kedua perangkat dianggap sama dalam proses komunikasi. Peer 1 memiliki file yang dibagikan dengan Peer 2 dan dapat mengakses printer bersama yang terhubung langsung ke Peer 2 untuk mencetak file. Peer 2 berbagi printer yang terhubung langsung dengan Peer 1 saat mengakses file bersama di Peer 1, seperti yang ditunjukkan pada gambar.



Aplikasi Peer-to-Peer

Aplikasi P2P memungkinkan perangkat untuk bertindak sebagai klien dan server dalam komunikasi yang sama, seperti yang ditunjukkan pada gambar. Dalam model ini, setiap klien adalah server dan setiap server adalah klien. Aplikasi P2P mengharuskan setiap **end devices** menyediakan antarmuka pengguna dan menjalankan **Background Services**.

Beberapa aplikasi P2P menggunakan sistem hibrida di mana berbagi **resource** terdesentralisasi, tetapi indeks yang menunjuk ke lokasi **resource** disimpan dalam direktori terpusat. Dalam sistem hibrida, setiap rekan mengakses server indeks untuk mendapatkan lokasi **resource** yang tersimpan di rekan lain.



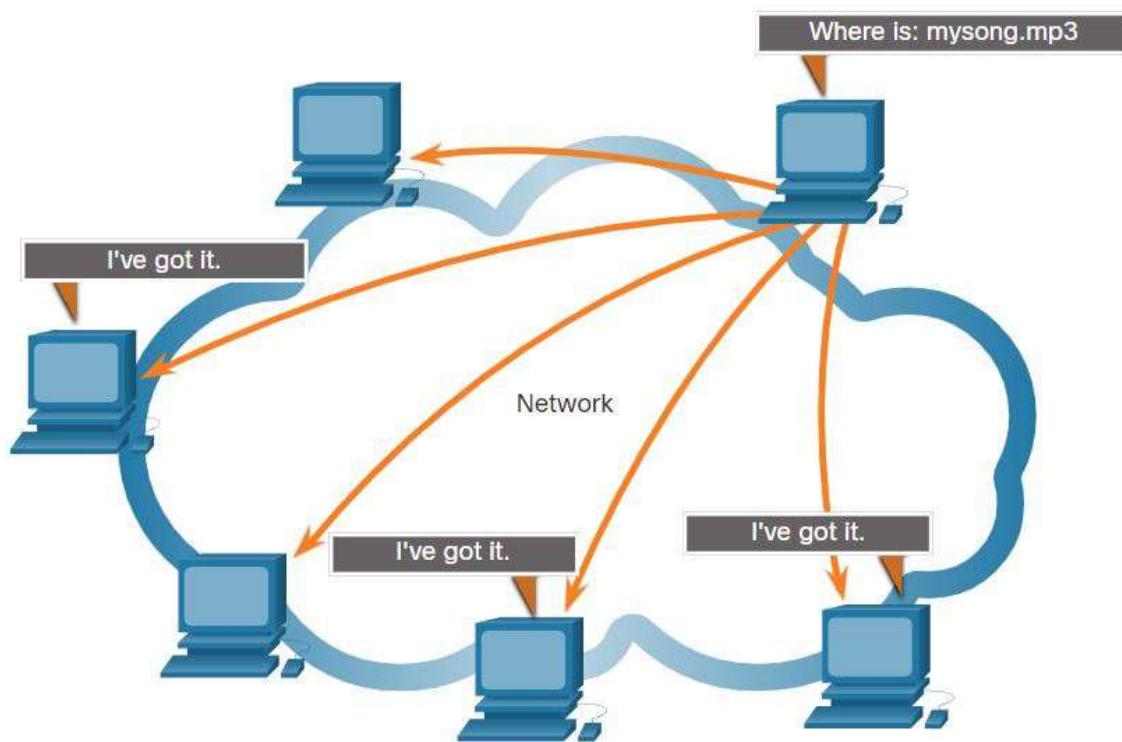
Aplikasi P2P Umum

Dengan aplikasi P2P, setiap komputer dalam jaringan yang menjalankan aplikasi dapat bertindak sebagai klien atau server untuk komputer lain dalam jaringan yang juga menjalankan aplikasi. Jaringan P2P umum mencakup hal-hal berikut:

- BitTorrent
- Direct Connected
- eDonkey
- Freenet

Beberapa aplikasi P2P didasarkan pada protokol Gnutella, di mana setiap pengguna berbagi seluruh file dengan pengguna lain. Seperti yang ditunjukkan pada gambar, gnutella-kompatibel perangkat lunak klien memungkinkan pengguna untuk terhubung ke layanan

Gnutella melalui internet, dan untuk menemukan dan mengakses **resource** yang dimiliki oleh rekan-rekan Gnutella lainnya. Banyak aplikasi klien Gnutella tersedia, termasuk µTorrent, BitComet, DC ++, Deluge, dan emule.



Banyak aplikasi P2P memungkinkan pengguna untuk berbagi potongan banyak file satu sama lain pada saat yang sama. Klien menggunakan file torrent untuk menemukan pengguna lain yang memiliki potongan yang mereka butuhkan sehingga mereka kemudian dapat terhubung langsung dengan mereka. File ini juga berisi informasi tentang komputer pelacak yang melacak pengguna mana yang memiliki potongan file tertentu. Klien meminta potongan dari beberapa pengguna pada saat yang sama. Ini dikenal sebagai segerombolan dan teknologinya disebut BitTorrent. BitTorrent memiliki kliennya sendiri. Tetapi ada banyak klien BitTorrent lainnya termasuk uTorrent, Deluge, dan qBittorrent.

Catatan: Semua jenis file dapat dibagi antara pengguna. Banyak dari file-file ini memiliki hak cipta, yang berarti bahwa hanya pencipta yang memiliki hak untuk menggunakan dan mendistribusikannya. Adalah melanggar hukum untuk mengunduh atau mendistribusikan file berhak cipta tanpa izin dari pemegang hak cipta. Pelanggaran hak cipta dapat mengakibatkan tuntutan pidana dan tuntutan hukum perdata.

Email Dan Web Protocol

Ada protokol khusus **Application Layer** yang dirancang untuk penggunaan umum seperti penjelajahan web dan email. materi pertama memberi Anda gambaran umum tentang protokol ini. materi ini masuk ke lebih detail.

Hypertext Transfer Protocol dan Hypertext Markup Language

Ketika alamat web atau Uniform Resource Locator (URL) diketik ke browser web, browser web membuat koneksi ke layanan web. Layanan web berjalan di server yang menggunakan protokol HTTP. URL dan Uniform Resource Identifiers (URI) adalah nama yang kebanyakan orang kaitkan dengan alamat web.

Untuk lebih memahami bagaimana browser web dan server web berinteraksi, periksa bagaimana halaman web dibuka di browser. Untuk contoh ini, gunakan <http://www.cisco.com/index.html> URL.

Langkah 1

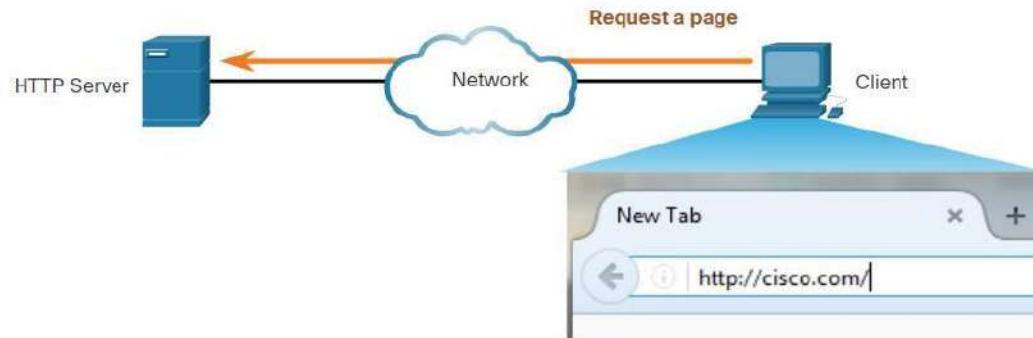
Browser menafsirkan tiga bagian dari URL:

- http (protokol atau skema)
- www.cisco.com (nama server)
- indeks.html (nama file tertentu yang diminta)



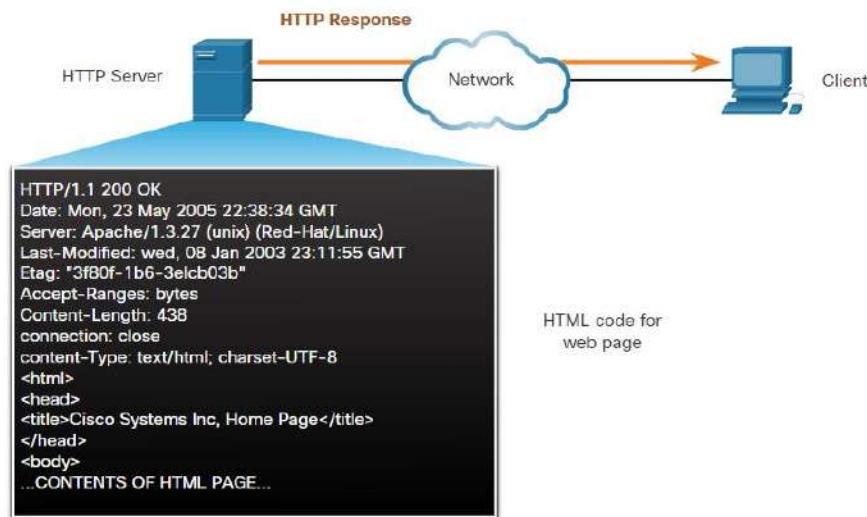
Langkah 2

Browser kemudian memeriksa dengan server nama untuk mengkonversi www.cisco.com ke alamat IP numerik, yang digunakan untuk terhubung ke server. Klien memulai permintaan HTTP ke server dengan mengirim permintaan GET ke server dan meminta file .html indeks.



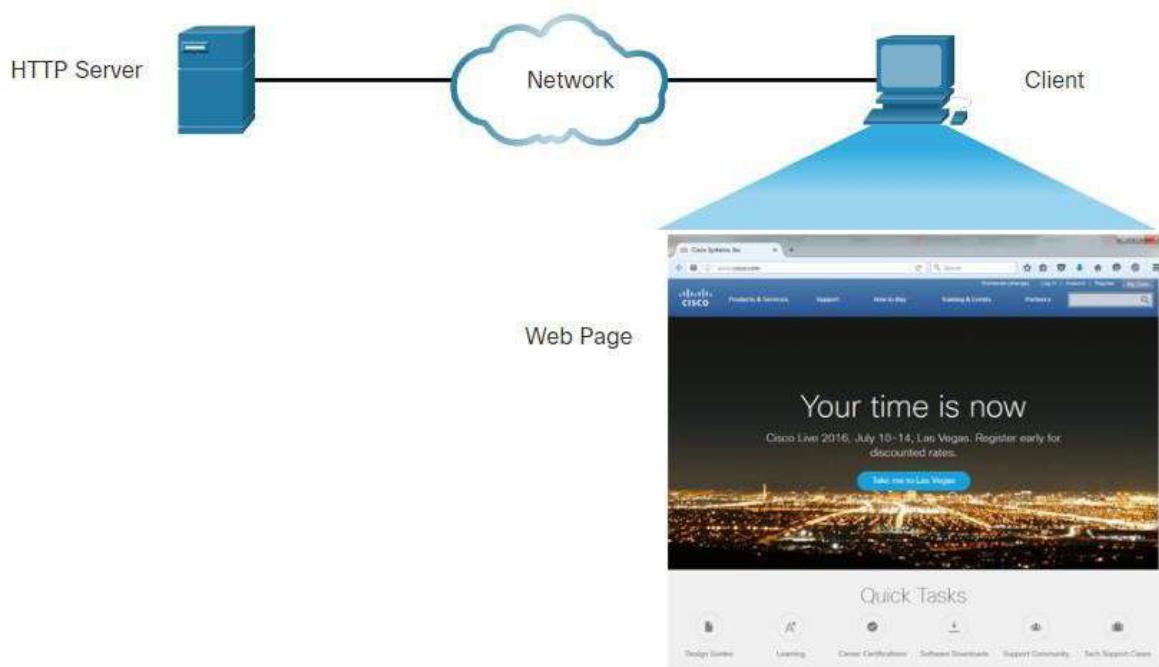
Langkah 3

Menanggapi permintaan tersebut, server mengirimkan kode HTML untuk halaman web ini ke browser.



Langkah 4

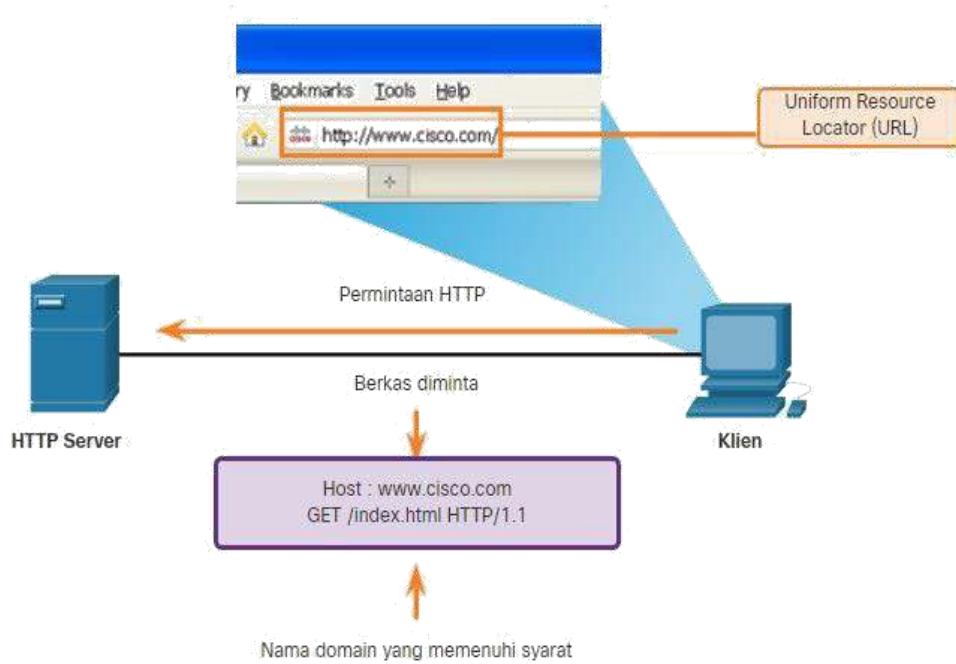
Browser menguraikan kode HTML dan memformat halaman untuk jendela browser.



HTTP DAN HTTPS

HTTP adalah protokol permintaan atau tanggapan. Ketika klien menggunakan web browser, HTTP mengirimkan permintaan ke server web, HTTP menentukan jenis pesan yang digunakan untuk komunikasi itu. Tiga jenis pesan umum adalah GET (lihat gambar), POST, dan PUT:

- **GET** – Ini adalah permintaan klien untuk data. Klien (browser web) mengirim pesan GET ke server web untuk meminta halaman HTML.
- **POST** – Ini mengunggah file data ke server web, seperti data formulir.
- **PUT** – Ini mengunggah sumber daya atau konten ke server web, seperti gambar.

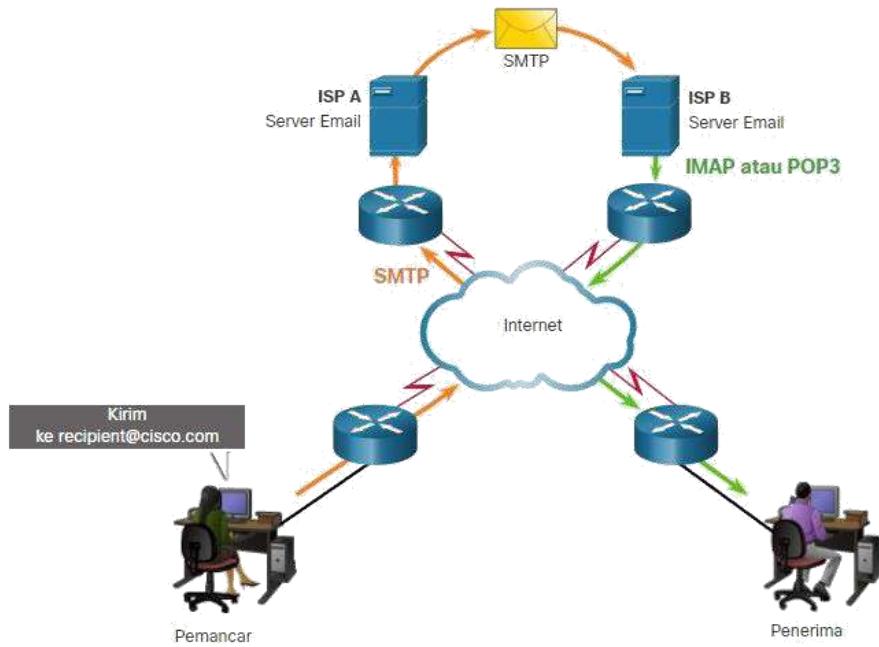


Meskipun HTTP sangat fleksibel, itu bukan protokol yang aman. Pesan permintaan mengirim informasi ke server dalam plaintext yang dapat dicegat dan dibaca. Tanggapan server, biasanya halaman HTML, juga tidak terenkripsi.

Untuk komunikasi yang aman di internet, protokol HTTP Secure (HTTPS) digunakan. HTTPS menggunakan otentikasi dan enkripsi untuk mengamankan data saat melakukan perjalanan antara klien dan server. HTTPS menggunakan proses respons request-server klien yang sama dengan HTTP, tetapi **stream** data dienkripsi dengan Transport Layer Security (TLS) atau pendahulunya Secure Socket Layer (SSL) sebelum diangkut melintasi jaringan.

Protokol Email

Salah satu layanan utama yang ditawarkan oleh ISP adalah hosting email. Untuk berjalan di komputer atau **end device** lainnya, email memerlukan beberapa aplikasi dan layanan, seperti yang ditunjukkan pada gambar. Email adalah metode store-and-forward untuk mengirim, menyimpan, dan mengambil pesan elektronik di seluruh jaringan. Pesan email disimpan dalam database di server email.



Klien email berkomunikasi dengan server email untuk mengirim dan menerima email. Server email berkomunikasi dengan server email lain untuk mengangkut pesan dari satu domain ke domain lainnya. Klien email tidak berkomunikasi langsung dengan klien email lain saat mengirim email. Sebaliknya, kedua klien bergantung pada server email untuk mengangkut pesan.

Email mendukung tiga protokol terpisah untuk operasi: **Simple Mail Transfer Protocol (SMTP)**, **Post Office Protocol (POP)**, dan **IMAP**. Proses **Application Layer** yang mengirimkan surat menggunakan SMTP. Klien mengambil email menggunakan salah satu dari dua protokol **Application Layer**: POP atau IMAP.

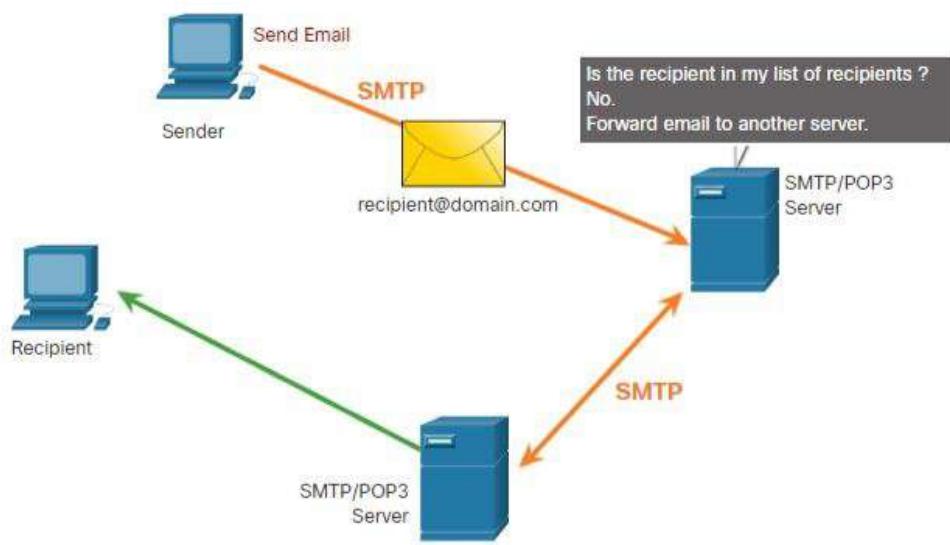
SMTP, POP, dan IMAP

A. SMTP

Format pesan SMTP memerlukan **Message Header** dan **Message Body**. Meskipun **Message Body** dapat berisi sejumlah teks, **Message Header** harus memiliki alamat email penerima yang diformat dengan benar dan alamat pengirim.

Ketika klien mengirim email, proses SMTP klien terhubung dengan proses SMTP server pada port 25 yang terkenal. Setelah koneksi dibuat, klien mencoba mengirim email ke server di seluruh koneksi. Ketika server menerima pesan, itu menempatkan pesan di akun lokal, jika penerima bersifat lokal, atau meneruskan pesan ke server email lain untuk pengiriman.

Server email tujuan mungkin tidak online, atau mungkin sibuk, ketika pesan email dikirim. Oleh karena itu, pesan spools SMTP akan dikirim di lain waktu. Secara berkala, server memeriksa antrian untuk pesan dan mencoba untuk mengirimnya lagi. Jika pesan masih belum dikirim setelah waktu kadaluarsa yang telah ditentukan, pesan dikembalikan ke pengirim sebagai tidak dapat ditinggalkan.



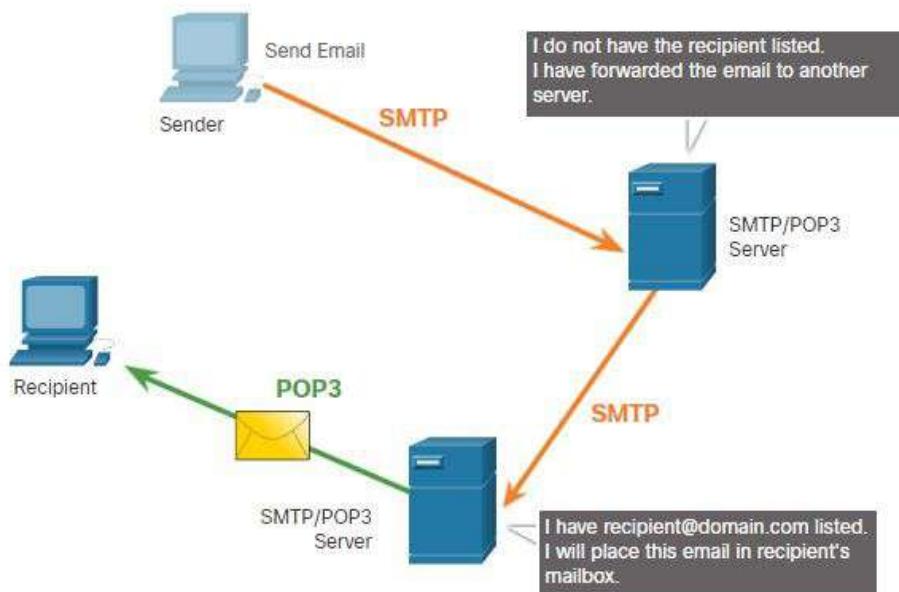
B. POP

POP digunakan oleh aplikasi untuk mengambil email dari server email. Dengan POP, mail diunduh dari server ke klien dan kemudian dihapus di server. Ini adalah operasi default POP.

Server memulai layanan POP dengan mendengarkan secara pasif pada port TCP 110 untuk permintaan koneksi klien. Ketika klien ingin menggunakan layanan, ia mengirimkan permintaan untuk membuat koneksi TCP dengan server, seperti yang ditunjukkan pada gambar. Ketika koneksi dibuat, server POP mengirimkan salam. Klien dan server POP kemudian bertukar perintah dan tanggapan sampai koneksi ditutup atau dibatalkan.

Dengan POP, pesan email diunduh ke klien dan dihapus dari server, sehingga tidak ada lokasi terpusat di mana pesan email disimpan. Karena POP tidak menyimpan pesan, tidak disarankan untuk bisnis kecil yang membutuhkan solusi cadangan terpusat.

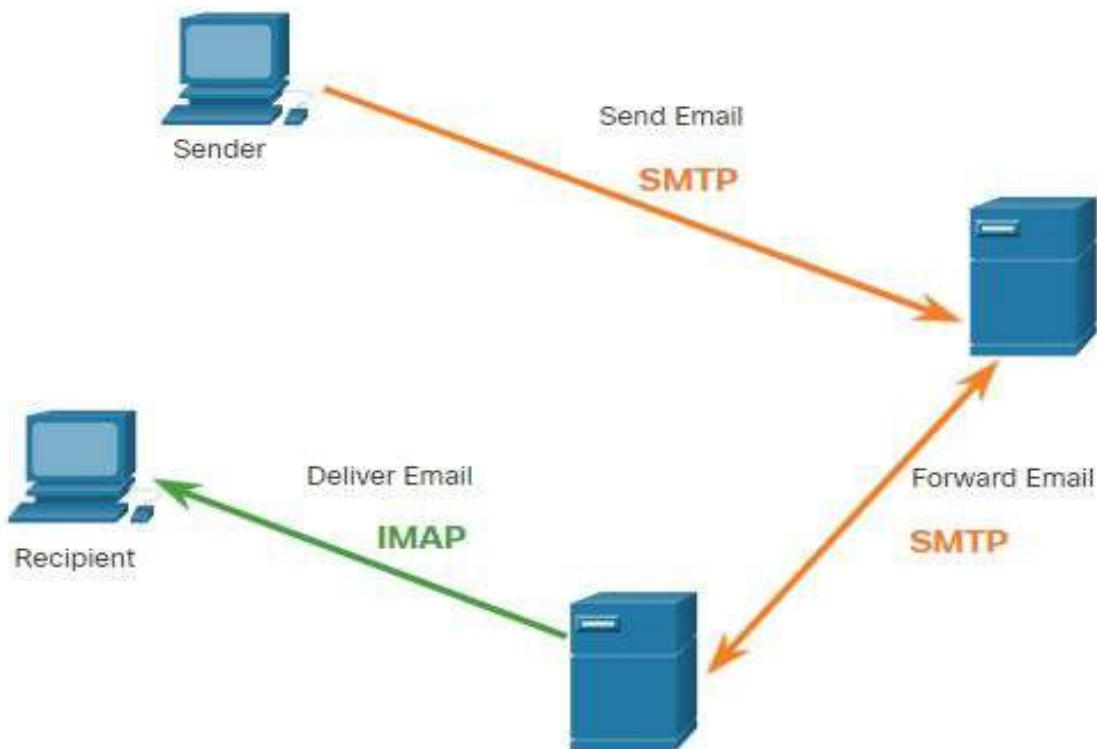
POP3 adalah versi yang paling umum digunakan.



C. IMAP

IMAP adalah protokol lain yang menjelaskan metode untuk mengambil pesan email. Tidak seperti POP, ketika pengguna terhubung ke server berkemampuan IMAP, salinan pesan diunduh ke aplikasi klien, seperti yang ditunjukkan pada gambar. Pesan asli disimpan di server sampai dihapus secara manual. Pengguna melihat salinan pesan dalam perangkat lunak klien email mereka.

Pengguna dapat membuat hierarki file di server untuk mengatur dan menyimpan email. Struktur file itu diduplikasi pada klien email juga. Ketika pengguna memutuskan untuk menghapus pesan, server menyinkronkan tindakan itu dan menghapus pesan dari server.



Layanan IP Address

Ada protokol khusus application layer lain yang dirancang untuk membuatnya lebih mudah untuk mendapatkan alamat untuk perangkat jaringan. Layanan ini sangat penting karena akan sangat memakan waktu untuk mengingat alamat IP. alih-alih URL atau secara manual mengkonfigurasi semua perangkat dalam jaringan menengah hingga besar. materi pertama dalam materi ini memberi Anda gambaran umum tentang protokol ini. materi ini membahas lebih detail tentang layanan pengalamatan IP, DNS dan DHCP.

Layanan Nama Domain

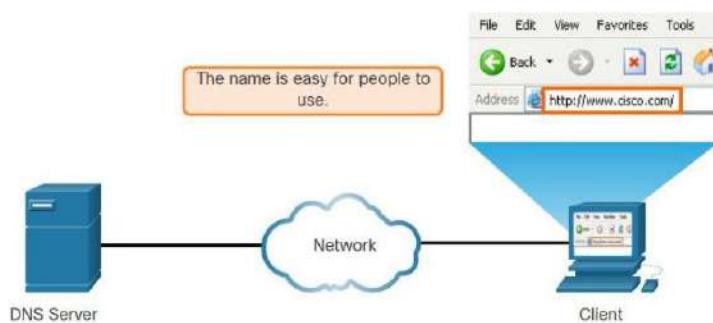
Dalam jaringan data, perangkat diberi label dengan alamat IP numerik untuk mengirim dan menerima data melalui jaringan. Nama domain dibuat untuk mengubah alamat IP menjadi nama yang sederhana dan dapat dikenali.

Di internet, fully-qualified domain names (FQDNs), seperti <http://www.cisco.com>, jauh lebih mudah bagi orang untuk mengingat daripada 198.133.219.25, yang merupakan alamat IP yang sebenarnya untuk server ini. Jika Cisco memutuskan untuk mengubah alamat IP menjadi www.cisco.com, itu transparan bagi pengguna karena nama domain tetap sama. Alamat baru hanya terkait dengan nama domain yang ada dan konektivitas dipertahankan.

Protokol DNS mendefinisikan layanan otomatis yang cocok dengan nama **resource** dengan alamat jaringan **IP** yang diperlukan. Ini termasuk format untuk kueri, tanggapan, dan data. Komunikasi protokol DNS menggunakan format tunggal yang disebut pesan. Format pesan ini digunakan untuk semua jenis kueri klien dan response server, pesan kesalahan, dan transfer informasi catatan **resource** antar server.

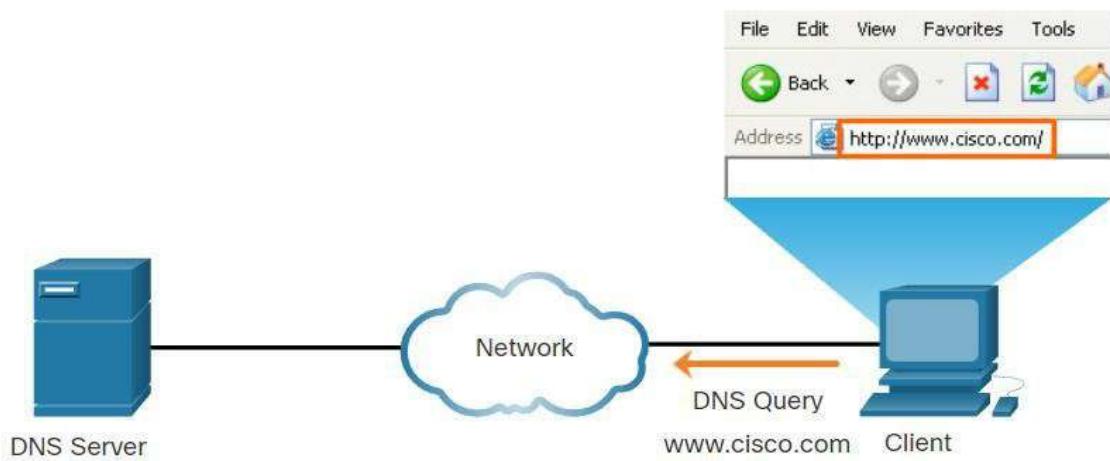
Langkah 1

Pengguna mengetik FQDN
pada browser



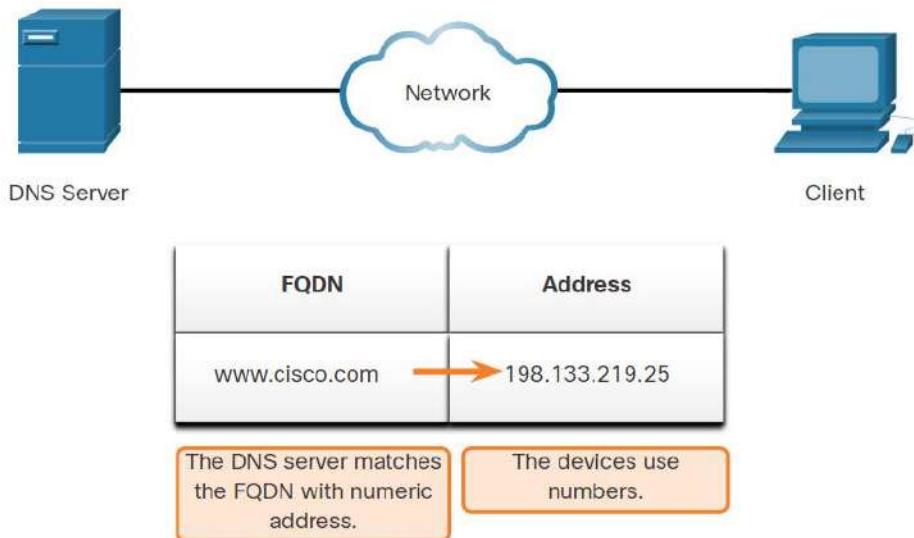
Langkah 2

Kueri DNS dikirim ke server DNS yang ditunjuk untuk komputer klien.



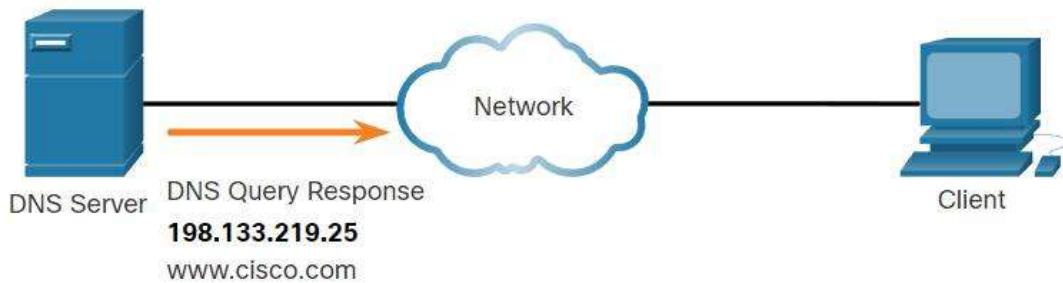
Langkah 3

Server DNS cocok dengan FQDN dengan alamat IP-nya.



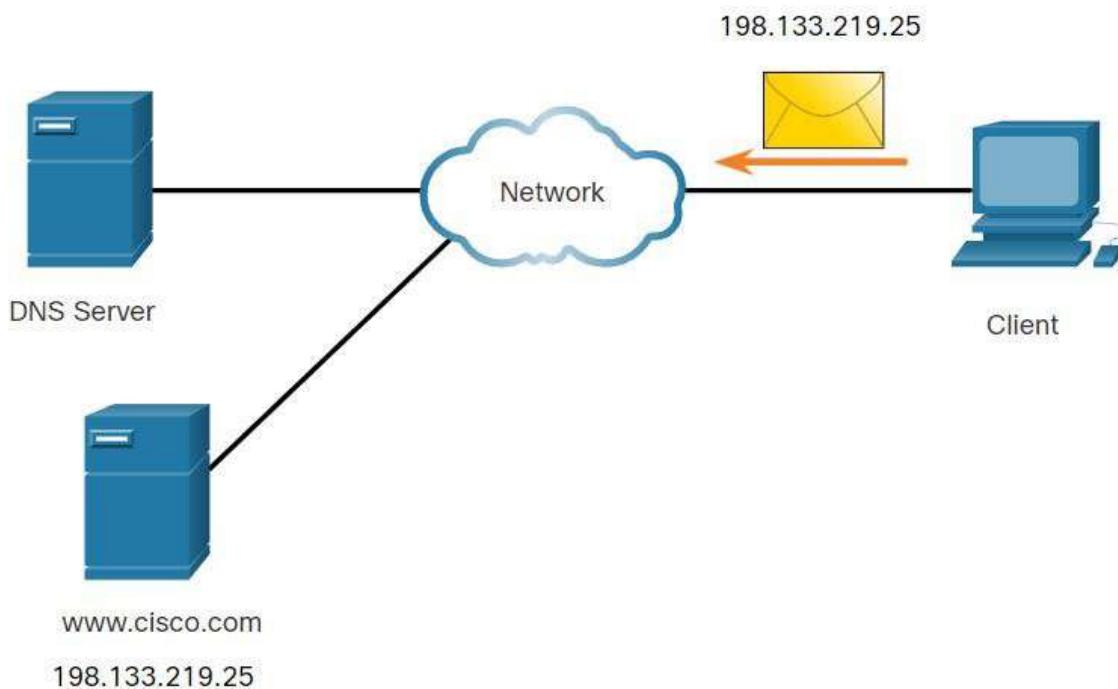
Langkah 4

Respons kueri DNS dikirim kembali ke klien dengan alamat IP untuk FQDN.



Langkah 5

Komputer klien menggunakan alamat IP untuk membuat permintaan dari server.



Format Pesan DNS

Server DNS menyimpan berbagai jenis catatan **resource** yang digunakan untuk menyelesaikan nama. Catatan ini berisi nama, alamat, dan jenis catatan. Beberapa jenis catatan ini adalah sebagai berikut:

- **A** – Alamat IPv4 End Device
- **NS** – Name Server otoritatif
- **AAAA** – Alamat IPv6 End Device (diucapkan quad-A)
- **MX** – Catatan pertukaran surat

Ketika klien membuat kueri, proses DNS server pertama kali melihat catatannya sendiri untuk menyelesaikan nama. Jika tidak dapat menyelesaikan nama dengan menggunakan catatan yang disimpan, ia menghubungi server lain untuk menyelesaikan nama. Setelah kecocokan ditemukan dan dikembalikan ke server permintaan asli, server menyimpan sementara alamat bernomor jika nama yang sama diminta lagi.

Layanan DNS client pada PC Windows juga menyimpan nama yang sebelumnya diselesaikan dalam memori. Perintah **ipconfig /displaydns** menampilkan semua entri DNS cache.

Seperti yang ditunjukkan dalam tabel, DNS menggunakan format pesan yang sama antara server, yang terdiri dari pertanyaan, jawaban, otoritas, dan informasi tambahan untuk semua jenis kueri klien dan tanggapan server, pesan kesalahan, dan transfer informasi catatan **resource**.

| Pesan DNS | Deskripsi |
|------------|---|
| Question | Pertanyaan untuk nameserver |
| Answer | Catatan Resource menjawab pertanyaan |
| Authority | Catatan Resource menunjuk ke arah otoritas |
| Additional | Catatan Resource memegang informasi tambahan |

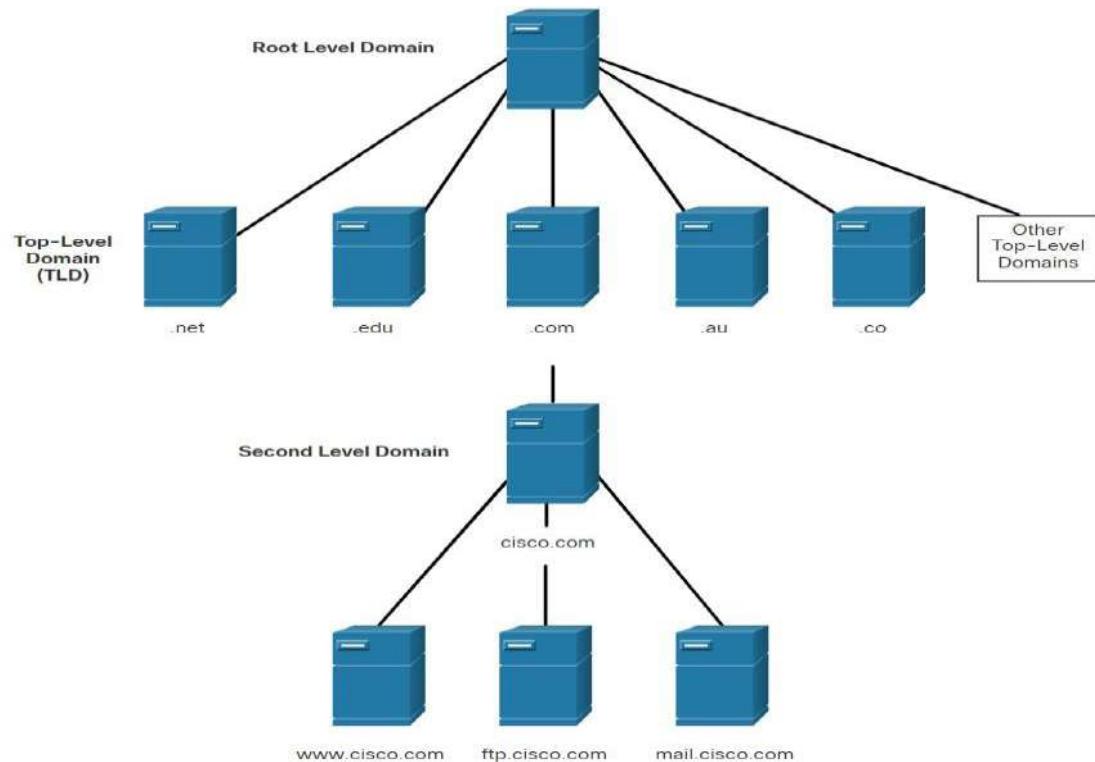
Hirarki DNS

Protokol DNS menggunakan sistem hierarkis untuk membuat database untuk memberikan resolusi nama, seperti yang ditunjukkan pada gambar. DNS menggunakan nama domain untuk membentuk hierarki.

Struktur penamaan dipecah menjadi zona kecil yang dapat dikelola. Setiap server DNS memelihara file database tertentu dan hanya bertanggung jawab untuk mengelola pemetaan nama-ke-IP untuk sebagian kecil dari seluruh struktur DNS. Ketika server DNS menerima permintaan untuk terjemahan nama yang tidak berada dalam zona DNS-nya, server DNS meneruskan permintaan ke server DNS lain dalam zona yang tepat untuk terjemahan. DNS dapat diskalakan karena resolusi nama host tersebar di beberapa server.

Top level domains yang berbeda mewakili jenis organisasi atau negara asal. Contoh **top level domains** adalah sebagai berikut:

- **.com** – bisnis atau industri
- **.org** – organisasi nirlaba
- **.au** – Australia
- **.co** – Kolombia



Perintah nslookup

Saat mengkonfigurasi perangkat jaringan, satu atau beberapa alamat DNS Server disediakan yang dapat digunakan klien DNS untuk resolusi nama. Biasanya ISP menyediakan alamat untuk digunakan untuk server DNS. Ketika aplikasi pengguna meminta untuk terhubung ke perangkat jarak jauh dengan nama, klien DNS yang meminta kueri **nameserver** untuk menyelesaikan nama ke alamat IP.

Sistem operasi komputer juga memiliki utilitas yang disebut Nslookup yang memungkinkan pengguna untuk secara manual query **nameserver** untuk menyelesaikan nama host yang diberikan. Utilitas ini juga dapat digunakan untuk memecahkan masalah resolusi nama dan untuk memverifikasi status **nameserver** saat ini.

Pada gambar ini, ketika perintah **nslookup** dikeluarkan, server DNS default yang dikonfigurasi untuk host Anda ditampilkan. Nama host atau domain dapat dimasukkan pada **prompt nslookup**. Utilitas Nslookup memiliki banyak pilihan yang tersedia untuk pengujian ekstensif dan verifikasi proses DNS.

```
C:\Users> nslookup
Default Server: dns-sj.cisco.com
Address: 171.70.168.183
> www.cisco.com
Server: dns-sj.cisco.com
Address: 171.70.168.183
Name: origin-www.cisco.com
Addresses: 2001:420:1101:1::a
           173.37.145.84
Aliases: www.cisco.com
> cisco.netacad.net
Server: dns-sj.cisco.com
Address: 171.70.168.183
Name: cisco.netacad.net
Address: 72.163.6.223
>
```

Dynamic Host Configuration Protocol

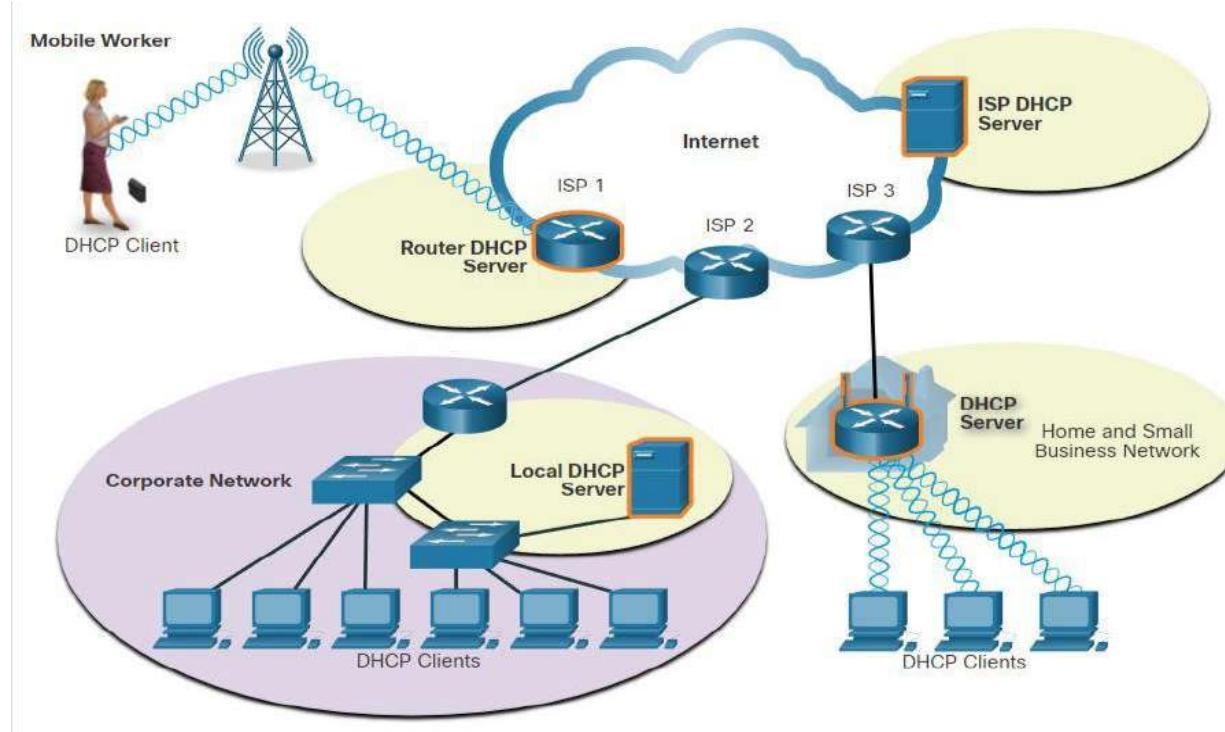
Dynamic Host Configuration Protocol (DHCP) untuk layanan IPv4 mengotomatiskan penugasan alamat IPv4, subnet mask, gateway, dan parameter jaringan IPv4 lainnya. Ini disebut sebagai dynamic address. Alternatif untuk dynamic address adalah static address. Saat menggunakan static address, administrator jaringan secara manual memasukkan informasi alamat IP pada host.

Ketika host terhubung ke jaringan, server DHCP dihubungi, dan alamat diminta. Server DHCP memilih alamat dari rentang alamat yang dikonfigurasi yang disebut **pool** dan menetapkan (menyewakan) ke host.

Pada jaringan yang lebih besar, atau di mana populasi pengguna sering berubah, DHCP lebih disukai untuk penugasan alamat. Pengguna baru mungkin tiba dan membutuhkan koneksi; Orang lain mungkin memiliki komputer baru yang harus terhubung. Daripada menggunakan static address untuk setiap koneksi, lebih efisien untuk memiliki alamat IPv4 yang ditetapkan secara otomatis menggunakan DHCP.

DHCP dapat mengalokasikan alamat IP untuk jangka waktu yang dapat dikonfigurasi, yang disebut **Lease Period**. **Lease Period** adalah pengaturan DHCP yang penting. Ketika masa sewa berakhir atau server DHCP mendapat pesan DHCP RELEASE, alamat dikembalikan ke **pool** DHCP untuk digunakan kembali. Pengguna dapat dengan bebas berpindah dari lokasi ke lokasi dan dengan mudah membangun kembali koneksi jaringan melalui DHCP.

Seperti yang ditunjukkan angka, berbagai jenis perangkat bisa menjadi server DHCP. Server DHCP di sebagian besar jaringan menengah-ke-besar biasanya merupakan server berbasis PC lokal dan khusus. Dengan jaringan rumah, server DHCP biasanya terletak pada router lokal yang menghubungkan jaringan rumah ke ISP.

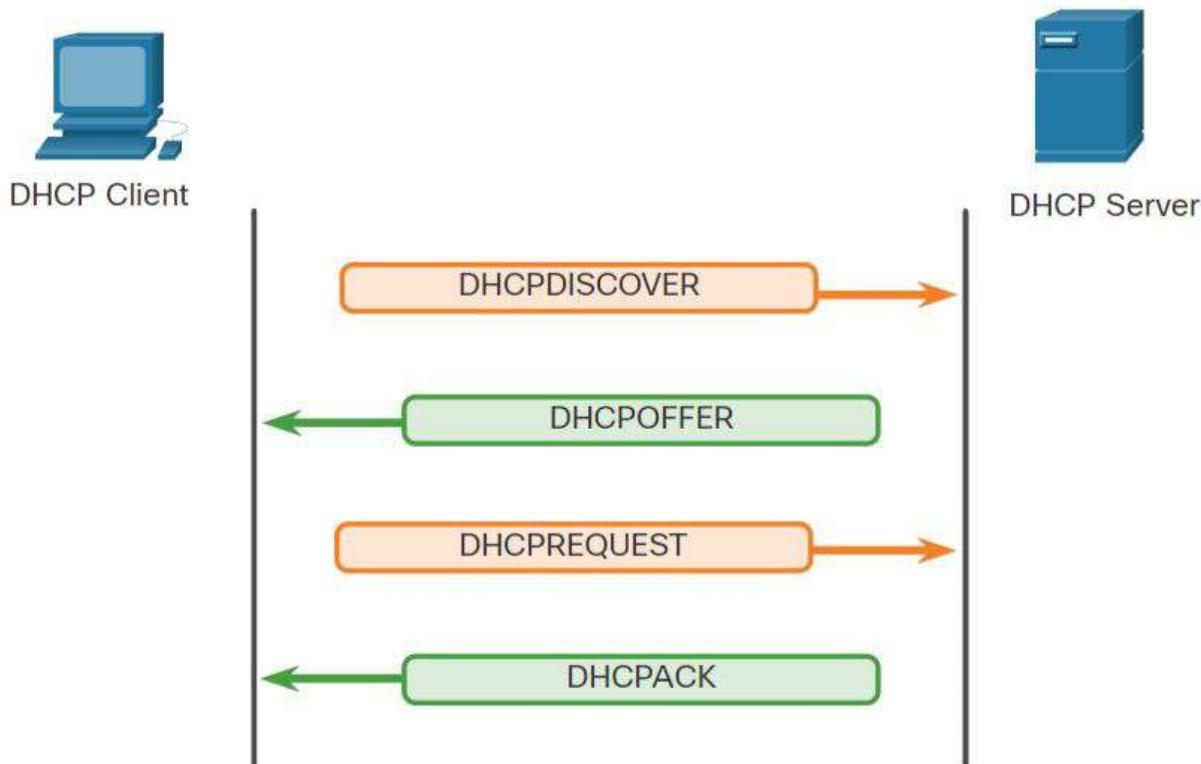


Banyak jaringan menggunakan DHCP dan static address. DHCP digunakan untuk host tujuan umum, seperti perangkat **end device**. Static address digunakan untuk perangkat jaringan, seperti router gateway, switch, server, dan printer.

DHCP untuk IPv6 (DHCPv6) menyediakan layanan serupa untuk klien IPv6. Satu perbedaan penting adalah bahwa DHCPv6 tidak menyediakan alamat gateway default. Ini hanya dapat diperoleh secara dinamis dari **Router Advertisement message** router.

Operasi DHCP

Seperti yang ditunjukkan pada gambar, ketika IPv4, perangkat dhcp-dikonfigurasi boot atau terhubung ke jaringan, klien menyiarkan DHCP discover (DHCPDISCOVER) pesan untuk mengidentifikasi server DHCP yang tersedia pada jaringan. Server DHCP membalas dengan pesan penawaran DHCP (DHCPOFFER), yang menawarkan sewa kepada klien. Pesan penawaran berisi alamat IPv4 dan subnet mask yang akan ditugaskan, alamat IPv4 dari server DNS, dan alamat IPv4 dari gateway default. **lease offer** juga mencakup durasi sewa.



Klien dapat menerima beberapa pesan DHCPOFFER jika ada lebih dari satu server DHCP di jaringan lokal. Oleh karena itu, ia harus memilih di antara mereka, dan mengirimkan pesan permintaan DHCP (DHCPREQUEST) yang mengidentifikasi server eksplisit dan tawaran sewa yang diterima klien. Klien juga dapat memilih untuk meminta alamat yang sebelumnya telah dialokasikan oleh server.

Dengan asumsi bahwa alamat IPv4 yang diminta oleh klien, atau ditawarkan oleh server, masih tersedia, server mengembalikan pesan pengakuan DHCP (DHCPACK) yang mengakui kepada klien bahwa sewa telah diselesaikan. Jika penawaran tidak lagi valid, maka server yang dipilih merespons dengan pesan pengakuan negatif DHCP (DHCPNAK). Jika pesan DHCPNAK dikembalikan, maka proses seleksi harus dimulai lagi dengan pesan DHCPDISCOVER baru yang dikirimkan. Setelah klien memiliki sewa, itu harus diperbarui sebelum berakhirnya sewa melalui pesan DHCPREQUEST lain.

Server DHCP memastikan bahwa semua alamat IP unik (alamat IP yang sama tidak dapat diberikan ke dua perangkat jaringan yang berbeda secara bersamaan). Sebagian besar ISP menggunakan DHCP untuk mengalokasikan alamat kepada pelanggan mereka.

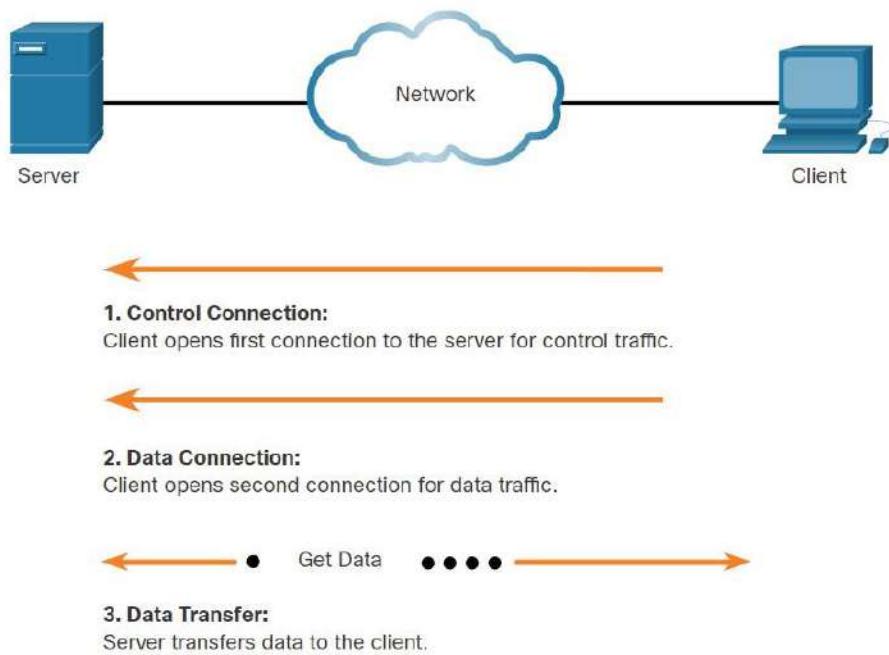
DHCPv6 memiliki satu set pesan yang mirip dengan DHCPv4. Pesan DHCPv6 adalah SOLICIT, ADVERTISE, INFORMATION REQUEST, dan REPLY.

File Transfer Protocol

Seperti yang Anda pelajari dalam materi sebelumnya, dalam model klien / server, klien dapat mengunggah data ke server, dan mengunduh data dari server, jika kedua perangkat menggunakan **File Transfer Protocol** (FTP). Seperti HTTP, email, dan protokol adresing, FTP umumnya digunakan protokol Application Layer. materi ini membahas FTP secara lebih rinci.

Protokol Transfer Berkas

FTP dikembangkan untuk memungkinkan transfer data antara klien dan server. Klien FTP adalah aplikasi yang berjalan pada komputer yang digunakan untuk mendorong dan menarik data dari server FTP.



Berdasarkan perintah yang dikirim di seluruh koneksi kontrol, data dapat diunduh dari server atau diunggah dari klien.

Klien menetapkan koneksi pertama ke server untuk **traffic** control menggunakan port TCP 21. **Traffic** terdiri dari perintah klien dan balasan server.

Klien menetapkan koneksi kedua ke server untuk transfer data aktual menggunakan port TCP 20. Koneksi ini dibuat setiap kali ada data yang akan ditransfer.

Transfer data dapat terjadi di kedua arah. Klien dapat mengunduh (menarik) data dari server, atau klien dapat mengunggah (mendorong) data ke server.

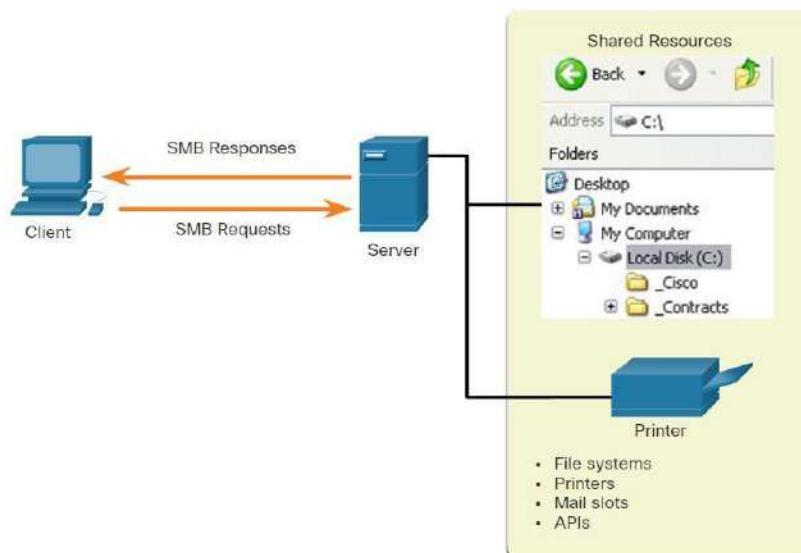
Blok Pesan Server

Server Message Block (SMB) adalah protokol **file-sharing** klien / server yang menggambarkan struktur **resource** jaringan bersama, seperti direktori, file, printer, dan port serial. Ini adalah protokol permintaan-jawaban. Semua pesan SMB berbagi format yang sama. Format ini menggunakan header berukuran tetap, diikuti oleh parameter berukuran variabel dan komponen data.

Berikut adalah tiga fungsi pesan SMB:

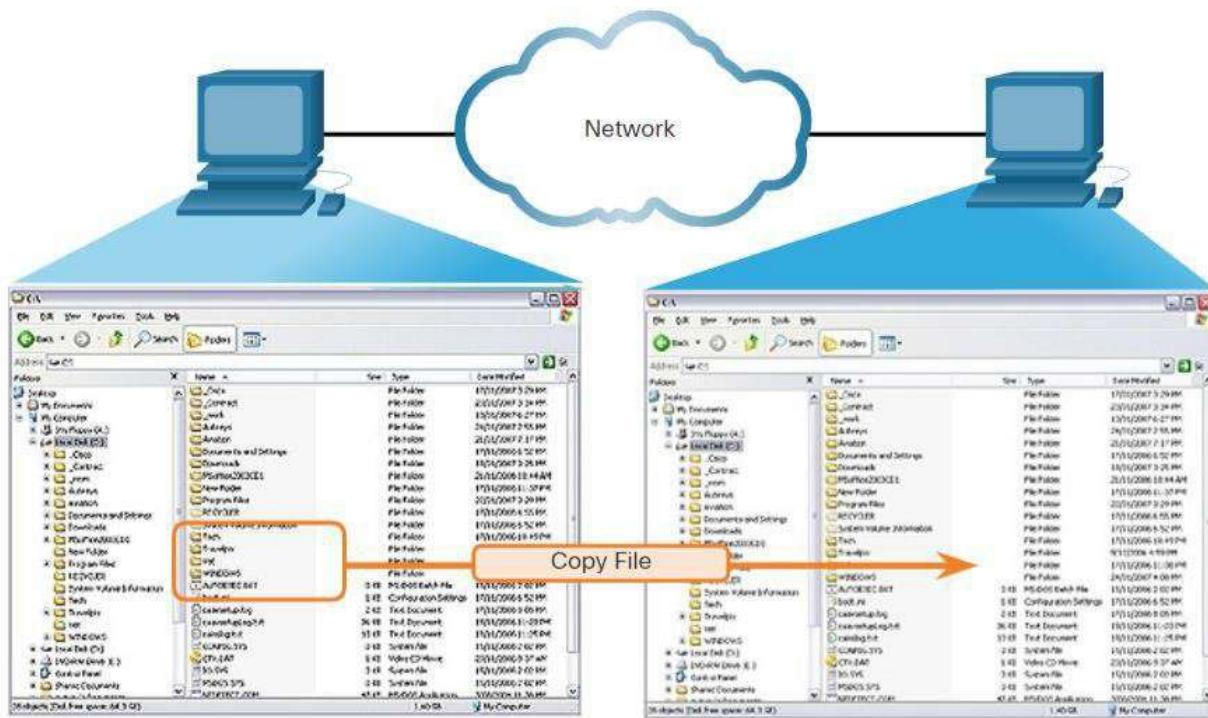
- Memulai, mengautentikasi, dan mengakhiri sesi.
- Kontrol akses file dan printer.
- Mengizinkan aplikasi untuk mengirim atau menerima pesan ke atau dari perangkat lain.

Layanan **file-sharing** dan **print** SMB telah menjadi andalan jaringan Microsoft. Dengan diperkenalkannya seri perangkat lunak Windows 2000, Microsoft mengubah struktur yang mendasari untuk menggunakan SMB. Dalam versi produk Microsoft sebelumnya, layanan SMB menggunakan protokol non-TCP / IP untuk mengimplementasikan **address resolution**. Dimulai dengan Windows 2000, semua produk Microsoft berikutnya menggunakan penamaan DNS, yang memungkinkan protokol TCP / IP untuk secara langsung mendukung berbagi **resource** SMB, seperti yang ditunjukkan pada gambar.



SMB adalah klien / server, protokol permintaan-tanggapan. Server dapat membuat **resource** mereka sendiri tersedia untuk klien di jaringan.

Proses pertukaran file SMB antara PC Windows ditampilkan pada gambar berikutnya.



Sebuah file dapat disalin dari PC ke PC dengan Windows Explorer menggunakan protokol SMB.

Berbeda dengan **file-sharing** yang didukung oleh FTP, klien membuat koneksi jangka panjang ke server. Setelah koneksi ditetapkan, pengguna klien dapat mengakses **resource** di server seolah-olah **resource** bersifat lokal ke host klien.

Sistem operasi LINUX dan UNIX juga menyediakan metode berbagi **resource** dengan jaringan Microsoft dengan menggunakan versi SMB yang disebut SAMBA. Sistem operasi Apple Macintosh juga mendukung pembagian **resource** dengan menggunakan protokol SMB.

BAB 16

~ *Network Security*
fundamentals ~

Judul Bab: Keamanan Jaringan Dasar

Tujuan Bab : Mengkonfigurasikan **Switch** dan **Router** dengan fitur Keamanan pada jaringan

Link Test Pemahaman : <https://s.id/Qy9W>

| Judul Bab | Tujuan Bab |
|---------------------------------|--|
| Ancaman dan kerentanan keamanan | Menjelaskan kenapa keamanan dasar diperlukan pada perangkat jaringan |
| Serangan Jaringan | Mengidentifikasi kerentanan keamanan |
| Mitigasi Serangan Jaringan | Mengidentifikasi teknik mitigasi dasar |
| Keamanan Perangkat | Mengkonfigurasi perangkat jaringan untuk mengurangi ancaman keamanan |

Ancaman Dan Kerentanan Keamanan

Jaringan komputer kabel dan nirkabel sangat penting untuk kegiatan sehari-hari. Penggunaan Personal dan organisasi bergantung pada komputer dan jaringan mereka. *Intrusi* oleh orang yang tidak berwenang dapat mengakibatkan pemadaman jaringan yang mahal dan kehilangan pekerjaan. Serangan terhadap jaringan dapat menghancurkan dan dapat mengakibatkan hilangnya waktu dan uang karena kerusakan, atau pencurian informasi atau aset penting.

Jenis-jenis ancaman

Penyusup dapat memperoleh akses ke jaringan melalui kerentanan perangkat lunak, serangan perangkat keras, atau dengan menebak nama pengguna dan kata sandi seseorang. Penyusup yang mendapatkan akses dengan memodifikasi perangkat lunak atau mengeksplorasi kerentanan perangkat lunak disebut **Threat Actor**.

Setelah **Threat-Actor** mendapatkan akses ke jaringan, empat jenis ancaman mungkin timbul.

A. Information Theft

Pencurian informasi membobol komputer untuk mendapatkan informasi rahasia. Informasi dapat digunakan atau dijual untuk berbagai tujuan seperti ketika seseorang mencuri informasi kepemilikan suatu organisasi, seperti data penelitian dan pengembangan.

B. Data Loss And Manipulation

Kehilangan dan manipulasi data membobol komputer untuk menghancurkan atau mengubah catatan data. Contoh kehilangan data adalah threat actor yang mengirim virus yang memformat ulang hard drive komputer. Contoh manipulasi data adalah membobol sistem catatan untuk mengubah informasi, seperti harga suatu barang.

C. Identity Theft

Pencurian identitas adalah bentuk pencurian informasi di mana informasi pribadi dicuri untuk tujuan mengambil alih identitas seseorang. Dengan menggunakan informasi ini, **Threat Actor** dapat memperoleh dokumen hukum, mengajukan kredit, dan melakukan pembelian online yang tidak sah. Mengidentifikasi pencurian adalah masalah yang berkembang dengan biaya miliaran dolar per tahun.

D. Disruption Of Service

Gangguan layanan mencegah pengguna yang sah mengakses layanan yang menjadi hak mereka. Contohnya termasuk serangan denial of service (DoS) pada server, perangkat jaringan, atau Link komunikasi jaringan.

Jenis Kerentanan

Kerentanan adalah tingkat kelemahan dalam jaringan atau perangkat. Beberapa tingkat kerentanan melekat pada router, switch, desktop, server, dan bahkan perangkat keamanan. Biasanya, perangkat jaringan yang diserang adalah titik akhir, seperti server dan komputer desktop.

Ada tiga kerentanan atau kelemahan utama: kebijakan teknologi, konfigurasi, dan keamanan. Ketiga sumber kerentanan ini dapat membuat jaringan atau perangkat terbuka untuk berbagai serangan, termasuk serangan kode/script berbahaya dan serangan jaringan.

Kerentanan Teknologi

| Kerentanan | Deskripsi |
|------------------------------|--|
| Kelemahan Protokol TCP/IP | Hypertext Transfer Protocol (HTTP), File Transfer Protocol (FTP), dan Internet Control Message Protocol (ICMP) secara inheren tidak aman. Simple Network Management Protocol (SNMP) dan Simple Mail Transfer Protocol (SMTP) terkait dengan struktur inheren tidak aman di mana TCP dirancang. |
| Kelemahan Sistem Operasi | Setiap sistem operasi memiliki masalah keamanan yang harus ditangani. UNIX, Linux, Mac OS, Mac OS X, Windows Server 2012, Windows 7, Windows 8 Mereka didokumentasikan dalam arsip Computer Emergency Response Team (CERT) di http://www.cert.org |
| Kelemahan Peralatan Jaringan | Berbagai jenis peralatan jaringan, seperti router, firewall, dan switch memiliki kelemahan keamanan yang harus diakui dan dilindungi. Kelemahan mereka termasuk perlindungan kata sandi, kurangnya otentikasi, protokol routing, dan lubang firewall. |

Kerentanan Konfigurasi

| Kerentanan | Deskripsi |
|--|--|
| Akun pengguna tanpa jaminan | Informasi akun pengguna dapat dikirimkan secara tidak aman di seluruh jaringan, mengekspos nama pengguna dan kata sandi kepada Threat-Actor . |
| Akun sistem dengan kata sandi yang mudah ditebak | Masalah umum ini adalah hasil dari kata sandi pengguna yang dibuat dengan buruk. |
| Layanan internet yang salah dikonfigurasi | Mengaktifkan JavaScript di browser web memungkinkan serangan melalui JavaScript yang dikendalikan oleh Threat-Actor saat mengakses situs yang tidak tepercaya. Sumber kelemahan potensial lainnya termasuk layanan terminal yang salah dikonfigurasi, FTP, atau server web (misalnya, Microsoft Internet Information Services (IIS), dan Apache HTTP Server). |
| Setelan default tanpa jaminan dalam produk | Banyak produk memiliki pengaturan default yang membuat atau mengaktifkan lubang dalam keamanan. |
| Peralatan jaringan yang salah dikonfigurasi | Kesalahan konfigurasi peralatan itu sendiri dapat menyebabkan masalah keamanan yang signifikan. Misalnya, daftar akses yang salah dikonfigurasi, protokol routing, atau string komunitas SNMP dapat membuat atau mengaktifkan lubang dalam keamanan. |

Kerentanan Kebijakan

| Kerentanan | Deskripsi |
|---|---|
| Kurangnya kebijakan keamanan tertulis | Kebijakan keamanan tidak dapat diterapkan atau ditegakkan secara konsisten jika tidak ditulis. |
| Politik | Pertempuran politik dan perang wilayah dapat membuat sulit untuk menerapkan kebijakan keamanan yang konsisten. |
| Kurangnya kontinuitas otentikasi | Kata sandi yang dipilih dengan buruk, mudah dipecahkan, atau default dapat memungkinkan akses tidak sah ke jaringan. |
| Kontrol akses logis tidak diterapkan | Pemantauan dan audit yang tidak memadai memungkinkan serangan dan penggunaan yang tidak sah untuk melanjutkan, membuang-buang resource perusahaan. Hal ini dapat mengakibatkan tindakan hukum atau penghentian terhadap teknisi TI, manajemen TI, atau bahkan kepemimpinan perusahaan yang memungkinkan kondisi yang tidak aman ini bertahan. |
| Instalasi dan perubahan perangkat lunak dan perangkat keras tidak mengikuti kebijakan | Perubahan yang tidak sah pada topologi jaringan atau pemasangan aplikasi yang tidak disetujui membuat atau mengaktifkan lubang dalam keamanan. |
| Disaster Recovery Plan tidak ada | Kurangnya disaster recovery plan memungkinkan kekacauan, kepanikan, dan kebingungan terjadi ketika bencana alam terjadi atau Threat-Actor menyerang perusahaan. |

Keamanan Fisik

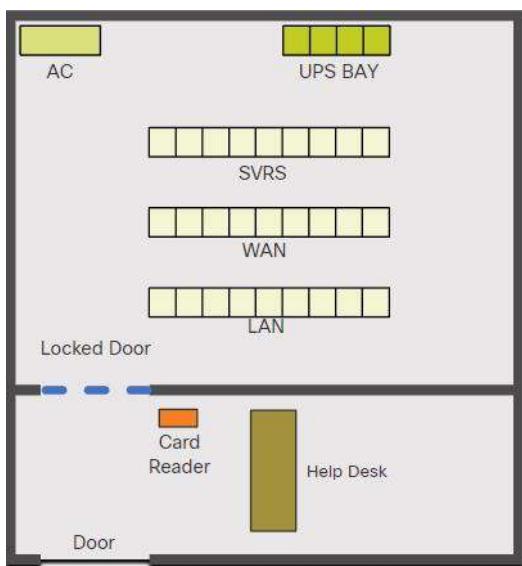
Area rentan yang sama pentingnya dari jaringan yang perlu dipertimbangkan adalah keamanan fisik perangkat. Jika resource jaringan dapat dikompromikan secara fisik, threat actor dapat menolak penggunaan resource jaringan.

Empat kelas ancaman fisik adalah sebagai berikut:

- **Ancaman perangkat keras** - Ini termasuk kerusakan fisik pada server, router, switch, pabrik kabel, dan workstation.
- **Ancaman lingkungan** - Ini termasuk suhu ekstrem (terlalu panas atau terlalu dingin) atau kelembaban ekstrem (terlalu basah atau terlalu kering).
- **Ancaman listrik** - Ini termasuk paku tegangan, tegangan pasokan yang tidak mencukupi (brownouts), daya yang tidak terkondisi (kebisingan), dan kehilangan daya total.
- **Ancaman pemeliharaan** - Ini termasuk penanganan komponen listrik utama yang buruk (debit elektrostatik), kurangnya suku cadang penting, kabel yang buruk, dan pelabelan yang buruk.

Rencana yang baik untuk keamanan fisik harus dibuat dan diimplementasikan untuk mengatasi masalah ini. Angka tersebut menunjukkan contoh rencana keamanan fisik.

Rencanakan Keamanan Fisik untuk Membatasi Kerusakan Pada Peralatan



Langkah 1. Kunci peralatan dan cegah akses tidak sah dari pintu, langit-langit, lantai yang ditinggikan, jendela, saluran, dan ventilasi.

Langkah 2. Pantau dan kontrol masuknya lemari dengan log elektronik.

Langkah 3. Gunakan kamera keamanan.

Serangan Jaringan

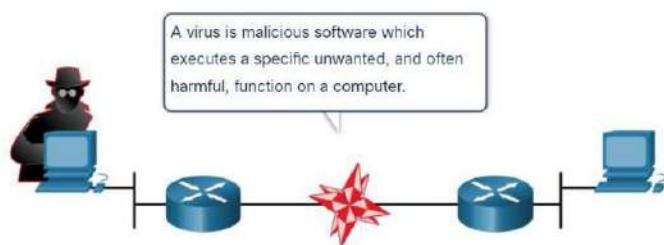
Materi sebelumnya menjelaskan jenis ancaman jaringan dan kerentanan yang memungkinkan ancaman. Topik ini masuk ke lebih detail tentang bagaimana aktor ancaman mendapatkan akses ke jaringan atau membatasi pengguna yang berwenang dari memiliki akses.

Jenis Malware

Malware adalah singkatan dari malicious software. Ini adalah kode atau perangkat lunak yang dirancang khusus untuk merusak, mengganggu, mencuri, atau menimbulkan tindakan “buruk” atau tidak sah pada data, host, atau jaringan. Virus, worm, dan Trojan horse adalah jenis malware.

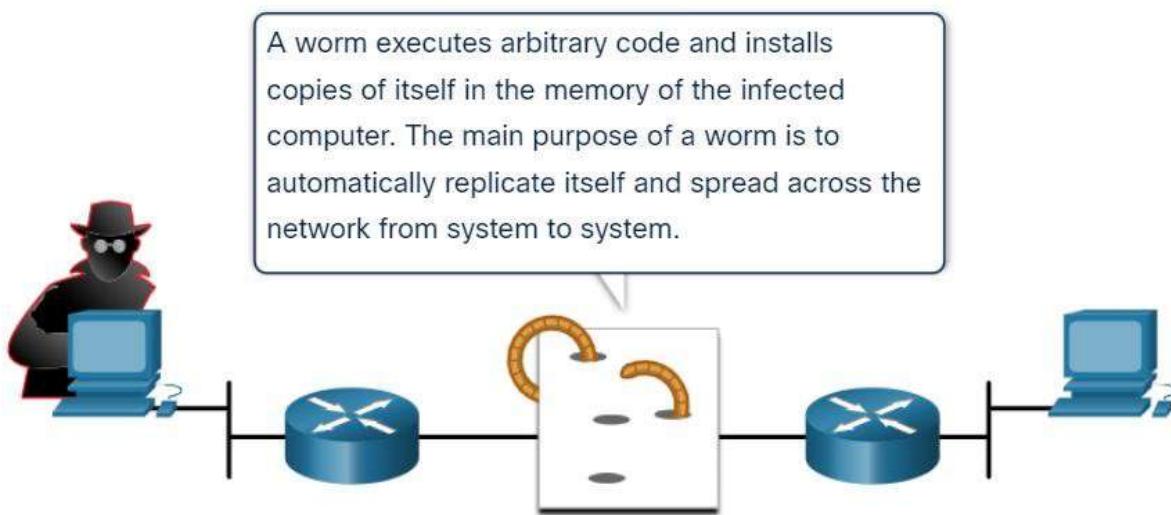
Virus

Virus komputer adalah jenis malware yang menyebar dengan memasukkan salinan dirinya ke dalam, dan menjadi bagian dari, program lain. Ini menyebar dari satu komputer ke komputer lain, meninggalkan infeksi saat bepergian. Virus dapat berkisar dalam tingkat keparahan dari menyebabkan efek yang sedikit mengganggu, untuk merusak data atau perangkat lunak dan menyebabkan kondisi denial of service (DoS). Hampir semua virus melekat pada file yang dapat dieksekusi, yang berarti virus mungkin ada pada sistem tetapi tidak akan aktif atau dapat menyebar sampai pengguna menjalankan atau membuka file atau program host berbahaya. Ketika kode host dijalankan, kode viral juga dieksekusi. Biasanya, program inang terus berfungsi setelah virus menginfeksinya. Namun, beberapa virus menimpa program lain dengan salinan diri mereka sendiri, yang menghancurkan program host sama sekali. Virus menyebar ketika perangkat lunak atau dokumen yang mereka lampirkan ditransfer dari satu komputer ke komputer lain menggunakan jaringan, disk, berbagi file, atau lampiran email yang terinfeksi.



Worm

Worm komputer mirip dengan virus karena mereka mereplikasi salinan fungsional dari diri mereka sendiri dan dapat menyebabkan jenis kerusakan yang sama. Berbeda dengan virus, yang memerlukan penyebaran file host yang terinfeksi, worm adalah perangkat lunak mandiri dan tidak memerlukan program host atau bantuan manusia untuk menyebarkan. Worm tidak perlu melampirkan ke program untuk menginfeksi host dan memasuki komputer melalui kerentanan dalam sistem. Worm memanfaatkan fitur sistem untuk melakukan perjalanan melalui jaringan tanpa bantuan.

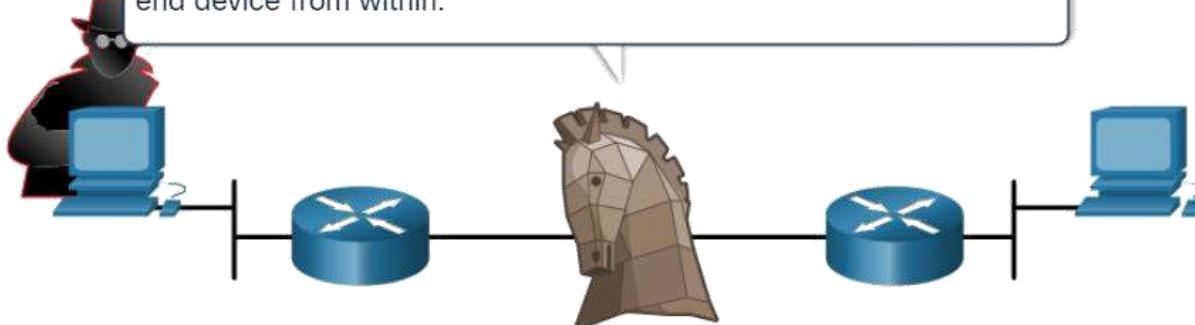


Trojan Horse

Trojan horse adalah jenis malware lain yang dinamai **Wooden Horse** yang digunakan orang Yunani untuk menyusup ke Troy. Ini adalah bagian berbahaya dari perangkat lunak yang terlihat sah. Pengguna biasanya ditipu untuk memuat dan mengeksekusinya di sistem mereka. Setelah diaktifkan, dapat mencapai sejumlah serangan pada host, dari menjengkelkan pengguna (dengan jendela pop-up yang berlebihan atau mengubah desktop) untuk merusak host (menghapus file, mencuri data, atau mengaktifkan dan menyebarkan malware lain, seperti virus). **Trojan horse** juga dikenal untuk membuat **backdoor** untuk memberikan pengguna jahat akses ke sistem.

Tidak seperti virus dan worm, **trojan horse** tidak bereproduksi dengan menginfeksi file lain. Trojan horse harus menyebar melalui interaksi pengguna seperti membuka lampiran email atau mengunduh dan menjalankan file dari internet.

A Trojan horse is a non-self-replicating type of malware. It often contains malicious code that is designed to look like something else, such as a legitimate application or file. When an infected application or file is downloaded and opened, the Trojan horse can attack the end device from within.



Serangan Pengintaian

Selain serangan kode berbahaya, juga mungkin bagi jaringan untuk menjadi mangsa berbagai serangan jaringan. Serangan jaringan dapat diklasifikasikan ke dalam tiga kategori utama:

- **Reconnaissance attacks** - Penemuan dan pemetaan sistem, layanan, atau kerentanan.
- **Access attacks** - Manipulasi data, akses sistem, atau hak istimewa pengguna yang tidak sah.
- **Denial of service** – Penonaktifan atau korupsi jaringan, sistem, atau layanan.

Untuk serangan pengintaian, aktor ancaman eksternal dapat menggunakan alat internet, seperti utilitas **nslookup** dan **whois**, untuk dengan mudah menentukan ruang alamat IP yang ditugaskan ke perusahaan atau entitas tertentu. Setelah ruang alamat IP ditentukan, aktor ancaman kemudian dapat melakukan ping alamat IP yang tersedia untuk umum untuk mengidentifikasi alamat yang aktif. Untuk membantu mengotomatisasi langkah ini, aktor ancaman dapat menggunakan alat **ping sweep**, seperti **fping** atau **gping**. Ini secara sistematis ping semua alamat jaringan dalam rentang tertentu atau subnet. Ini mirip dengan melalui bagian dari buku telepon dan memanggil setiap nomor untuk melihat siapa yang menjawab. 3 cara sederhana pengintaian

A. Internet Query

Threat Actor sedang mencari informasi awal tentang target. Berbagai alat dapat digunakan, termasuk pencarian Google, situs web organisasi, whois, dan banyak lagi.

B. Ping Sweeps

Threat Actor memulai ping sweeps untuk menentukan alamat IP mana yang aktif.

C. Port Scans

Threat Actor melakukan pemindaian port pada alamat IP aktif yang ditemukan.

Serangan Akses

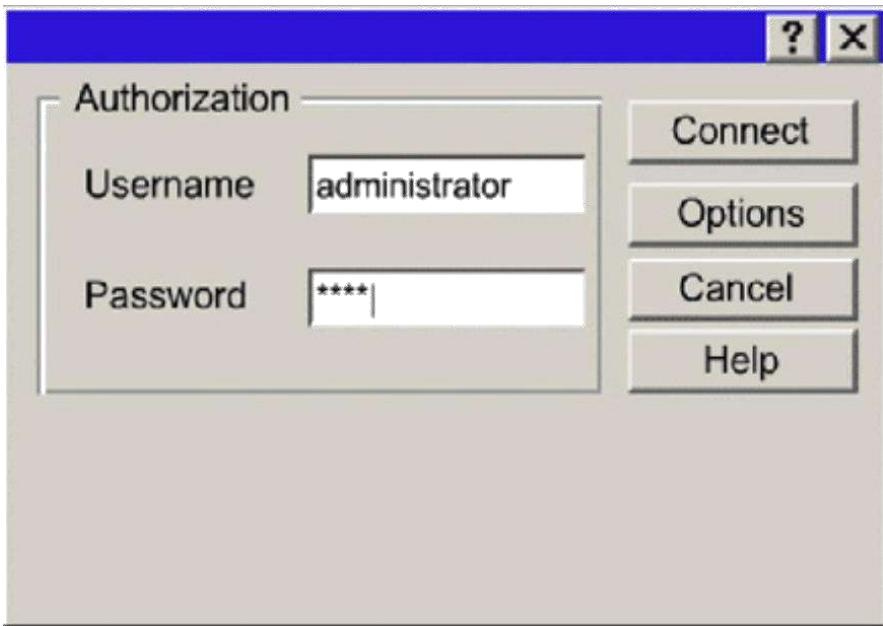
Serangan akses mengeksplorasi kerentanan yang diketahui dalam layanan otentikasi, layanan FTP, dan layanan web untuk mendapatkan akses ke akun web, database rahasia, dan informasi sensitif lainnya. Serangan akses memungkinkan individu untuk mendapatkan akses tidak sah ke informasi yang tidak memiliki hak untuk dilihat. Serangan akses dapat diklasifikasikan menjadi empat jenis: **Password Attacks**, **Trust Exploitation**, **Port Redirection**, dan **man-in-the-middle**.

Serangan Kata Sandi

Aktor ancaman dapat menerapkan serangan kata sandi menggunakan beberapa metode berbeda:

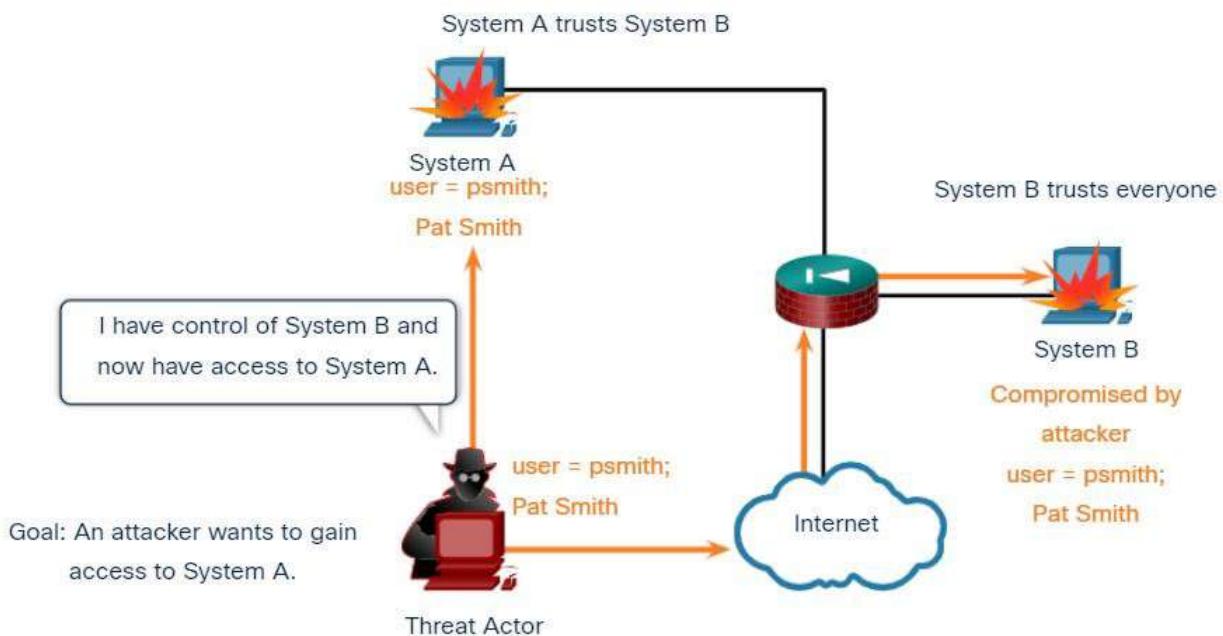
- **brute-force attacks**
- **Trojan horse attacks**
- **Packet Sniffers**

Angka pertama menunjukkan kotak prompt login dengan nama pengguna, administrator dan kata sandi, ****.



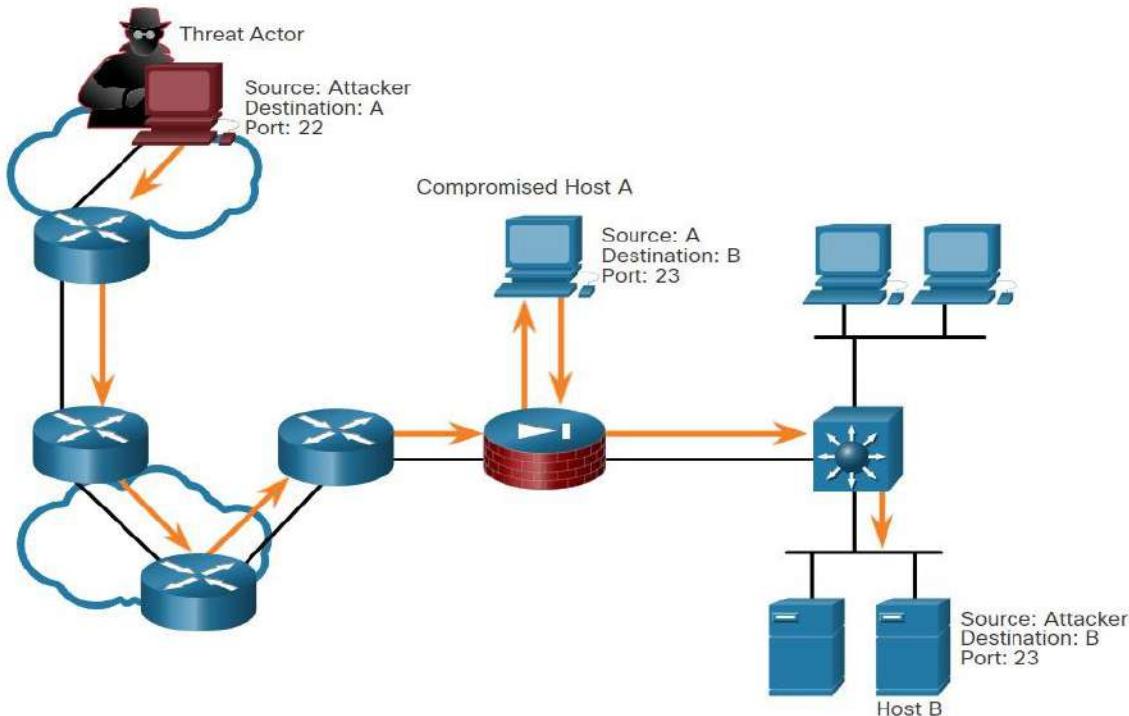
Trust Exploitation

Threat-Actor menggunakan hak istimewa yang tidak sah untuk mendapatkan akses ke sistem, mungkin mengorbankan target. Klik Putar pada gambar untuk melihat contoh eksploitasi kepercayaan.



Port Redirection

Actor-Threat menggunakan sistem yang dikompromikan sebagai dasar untuk serangan terhadap target lain. Contoh dalam gambar menunjukkan aktor ancaman menggunakan SSH (port 22) untuk terhubung ke host A. Host A yang dikompromikan dipercaya oleh host B dan, oleh karena itu, aktor ancaman dapat menggunakan Telnet (port 23) untuk mengaksesnya.



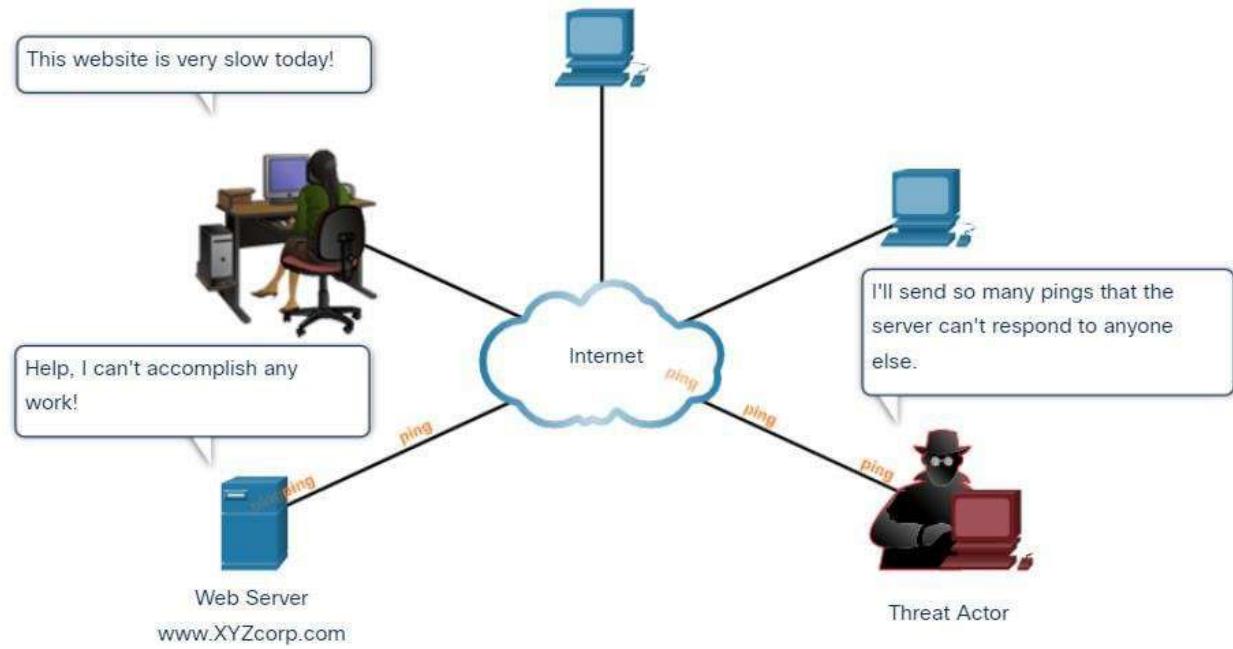
Serangan Denial of Service

Serangan Denial of Service (DoS) adalah bentuk serangan yang paling dipublikasikan dan di antara yang paling sulit dihilangkan. Namun, karena kemudahan implementasi dan kerusakan yang berpotensi signifikan, serangan DoS layak mendapat perhatian khusus dari administrator keamanan.

Serangan DoS mengambil banyak bentuk. Pada akhirnya, mereka mencegah orang yang berwenang menggunakan layanan dengan mengkonsumsi resource sistem. Untuk membantu mencegah serangan DoS, penting untuk tetap up to date dengan pembaruan keamanan terbaru untuk sistem operasi dan aplikasi.

Serangan Dos

Serangan DoS adalah risiko besar karena mengganggu komunikasi dan menyebabkan hilangnya waktu dan uang yang signifikan. Serangan-serangan ini relatif mudah dilakukan, bahkan oleh **Threat-Actor** yang tidak terampil.



Serangan DDos

DDoS mirip dengan serangan DoS, tetapi berasal dari beberapa sumber yang terkoordinasi. Misalnya, **Threat-Actor** membangun jaringan host yang terinfeksi, yang dikenal sebagai zombie. Jaringan zombie disebut botnet. **Threat Actor** menggunakan program command and control (CnC) untuk menginstruksikan botnet zombie untuk melakukan serangan DDoS.

Mitigasi Serangan Jaringan

Sekarang setelah Anda tahu lebih banyak tentang bagaimana Threat-Actor dapat masuk ke jaringan, Anda perlu memahami apa yang harus dilakukan untuk mencegah akses yang tidak sah ini. Materi ini merinci beberapa tindakan yang dapat Anda ambil untuk membuat jaringan Anda lebih aman.

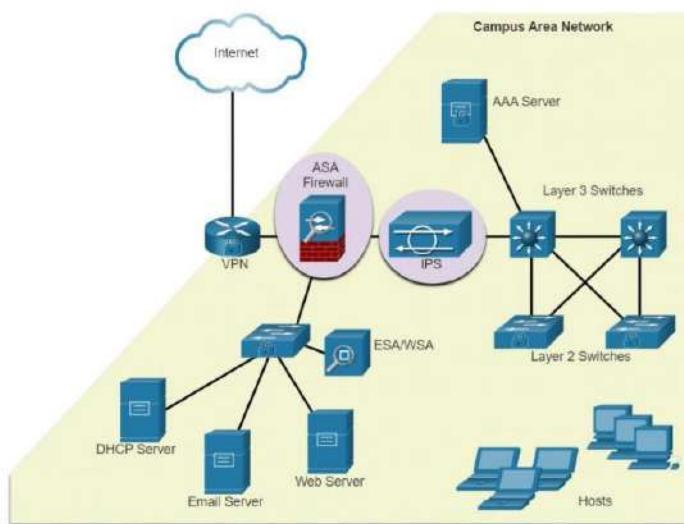
Pendekatan Defense-in-depth approach

Untuk mengurangi serangan jaringan, Anda harus terlebih dahulu mengamankan perangkat termasuk router, switch, server, dan host. Sebagian besar organisasi menggunakan pendekatan **defense-in-depth approach** (juga dikenal sebagai pendekatan berlapis) untuk keamanan. Ini membutuhkan kombinasi perangkat jaringan dan layanan yang bekerja bersama-sama.

Pertimbangkan jaringan dalam gambar. Ada beberapa perangkat dan layanan keamanan yang telah diimplementasikan untuk melindungi pengguna dan asetnya dari ancaman TCP / IP.

Semua perangkat jaringan termasuk router dan switch juga dikunci seperti yang ditunjukkan oleh kunci kombinasi pada ikon masing-masing. Ini menunjukkan bahwa mereka telah diamankan untuk mencegah Threat-Actor mendapatkan akses dan gangguan dengan perangkat.

Beberapa perangkat dan layanan keamanan diimplementasikan untuk melindungi pengguna dan aset organisasi terhadap ancaman TCP / IP.



- **VPN** – Router digunakan untuk menyediakan layanan VPN yang aman dengan situs perusahaan dan dukungan akses jarak jauh untuk pengguna jarak jauh menggunakan **tunnel** terenkripsi yang aman.
- **ASA Firewall** – Perangkat khusus ini menyediakan layanan firewall stateful. Ini memastikan bahwa **traffic** internal dapat keluar dan kembali, tetapi **traffic** eksternal tidak dapat memulai koneksi ke host dalam.
- **IPS – Intrusion Prevention System (IPS)** memantau **traffic** masuk dan keluar mencari malware, tanda tangan serangan jaringan, dan banyak lagi. Jika mengenali ancaman, ia dapat segera menghentikannya.
- **ESA / WSA – Email Security Appliance** (ESA) menyaring spam dan email yang mencurigakan. Alat keamanan web (WSA) menyaring situs malware internet yang dikenal dan mencurigakan.
- **AAA Server** – Server ini berisi database aman tentang siapa yang berwenang untuk mengakses dan mengelola perangkat jaringan. Perangkat jaringan mengautentikasi pengguna administratif menggunakan database ini.

Menyiapkan Cadangan

Mencadangkan konfigurasi dan data perangkat adalah salah satu cara paling efektif untuk melindungi dari kehilangan data. Cadangan data menyimpan salinan informasi di komputer ke media cadangan yang dapat dilepas yang dapat disimpan di tempat yang aman. Perangkat infrastruktur harus memiliki cadangan file konfigurasi dan gambar iOS pada FTP atau server file serupa. Jika komputer atau perangkat keras router gagal, data atau konfigurasi dapat dipulihkan menggunakan salinan cadangan.

Backup harus dilakukan secara teratur seperti yang diidentifikasi dalam kebijakan keamanan. Backup data biasanya disimpan di luar kantor untuk melindungi media cadangan jika terjadi sesuatu pada fasilitas utama. Host Windows memiliki utilitas cadangan dan pemulihan. Penting bagi pengguna untuk mencadangkan data mereka ke drive lain, atau ke penyedia penyimpanan berbasis cloud.

Tabel menunjukkan pertimbangan cadangan dan deskripsinya.

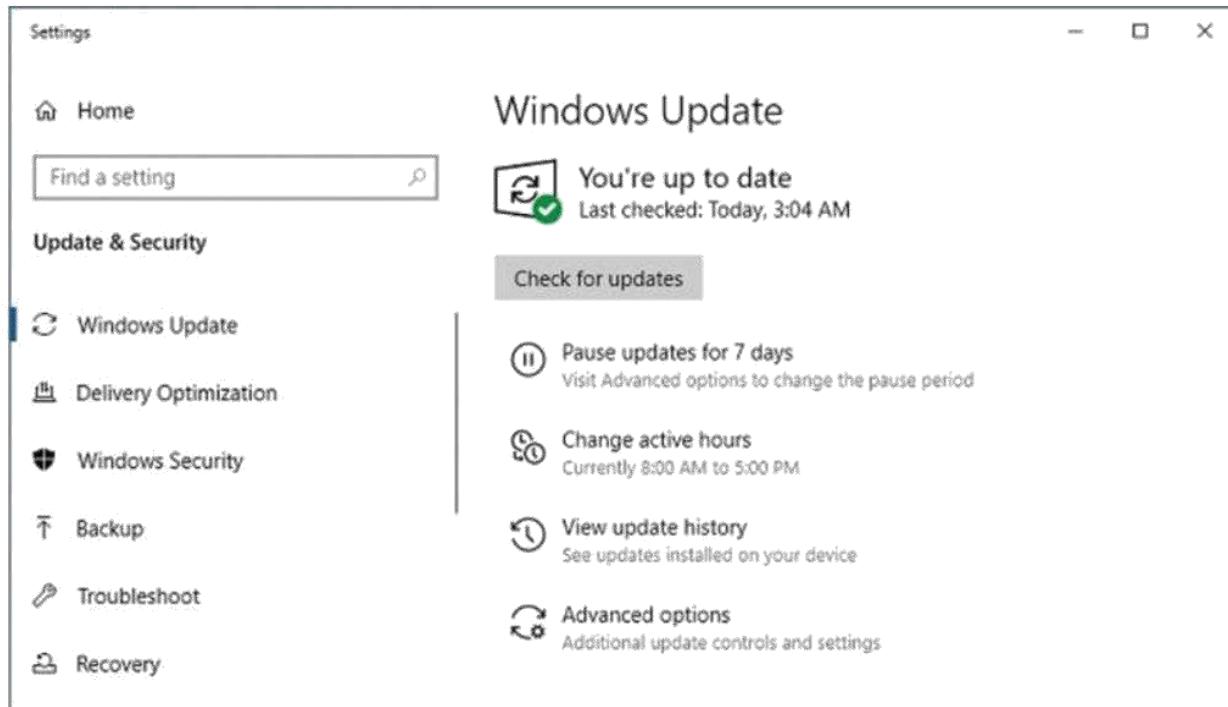
| Pertimbangan | Deskripsi |
|--------------|---|
| Frekuensi | Lakukan backup secara teratur seperti yang diidentifikasi dalam kebijakan keamanan. Backup penuh dapat memakan waktu, oleh karena itu melakukan backup bulanan atau mingguan dengan backup parsial sering file berubah. |
| Validasi | Selalu validasi cadangan untuk memastikan integritas data dan memvalidasi prosedur pemulihan file. |
| Penyimpanan | Cadangan harus diangkut ke lokasi penyimpanan di luar kantor yang disetujui pada rotasi harian, mingguan, atau bulanan, seperti yang dipersyaratkan oleh kebijakan keamanan. |
| Keamanan | Backup harus dilindungi menggunakan password yang kuat. Kata sandi diperlukan untuk memulihkan data. |

Upgrade, Update, dan Patch

Tetap up to date dengan perkembangan terbaru dapat menyebabkan pertahanan yang lebih efektif terhadap serangan jaringan. Ketika malware baru dirilis, perusahaan harus tetap mengikuti versi terbaru dari perangkat lunak antivirus.

Cara paling efektif untuk mengurangi serangan worm adalah mengunduh pembaruan keamanan dari vendor sistem operasi dan menambal semua sistem yang rentan. Mengelola berbagai sistem melibatkan pembuatan gambar perangkat lunak standar (sistem operasi dan aplikasi terakreditasi yang diizinkan untuk digunakan pada sistem klien) yang digunakan pada sistem baru atau yang ditingkatkan. Namun, persyaratan keamanan berubah, dan sistem yang sudah digunakan mungkin perlu menginstal patch keamanan yang diperbarui.

Salah satu solusi untuk pengelolaan patch keamanan kritis adalah untuk memastikan semua sistem akhir secara otomatis men-download update, seperti yang ditunjukkan untuk Windows 10 dalam gambar. Patch keamanan secara otomatis diunduh dan diinstal tanpa campur tangan pengguna.

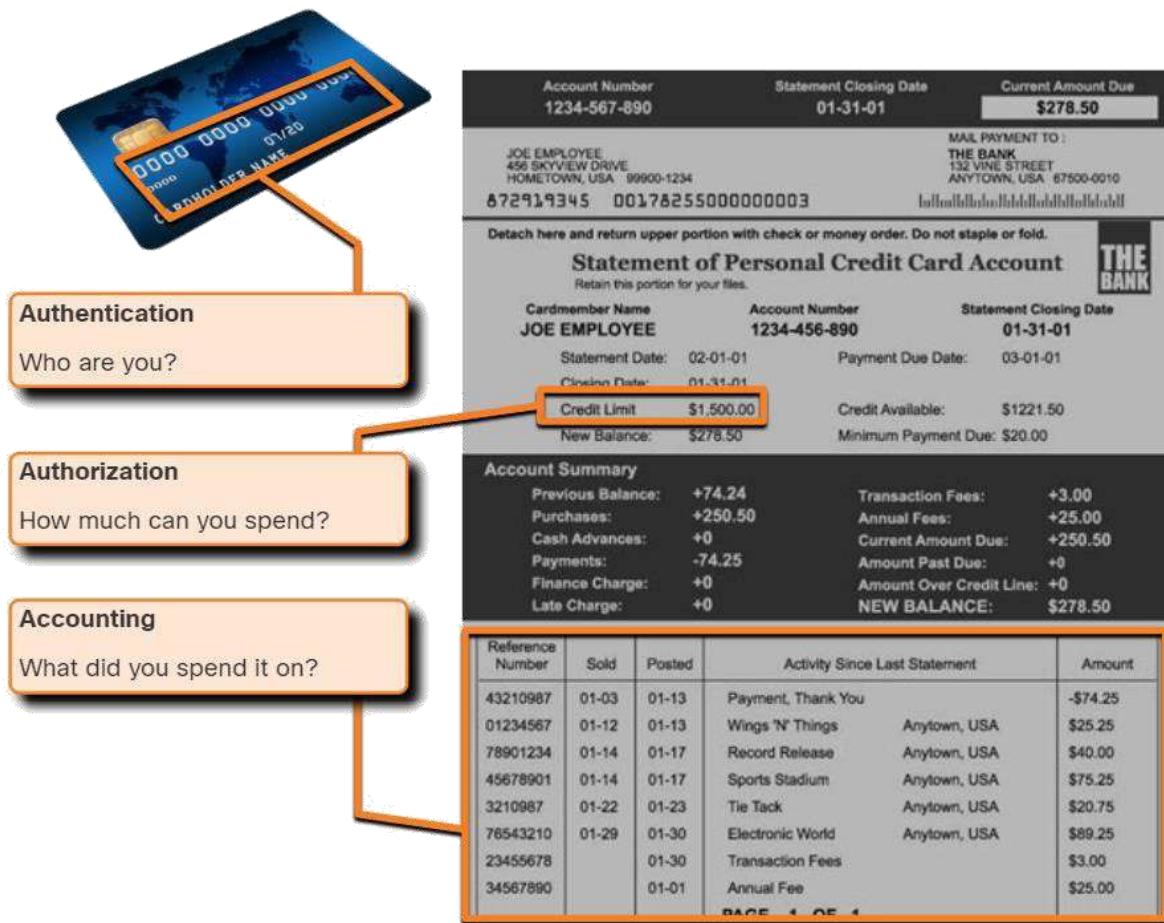


Authentication, Authorization, and Accounting

Semua perangkat jaringan harus dikonfigurasi dengan aman untuk hanya menyediakan individu yang berwenang dengan akses. Otentikasi, otorisasi, dan akuntansi (AAA, atau “triple A”) layanan keamanan jaringan menyediakan kerangka kerja utama untuk mengatur kontrol akses pada perangkat jaringan.

AAA adalah cara untuk mengontrol siapa yang diizinkan untuk mengakses jaringan (authenticate), tindakan apa yang mereka lakukan saat mengakses jaringan (otorisasi), dan membuat catatan tentang apa yang dilakukan saat mereka berada di sana (akuntansi).

Konsep AAA mirip dengan penggunaan kartu kredit. Kartu kredit mengidentifikasi siapa yang dapat menggunakanannya, berapa banyak yang dapat dihabiskan pengguna, dan memperhitungkan barang apa yang dihabiskan pengguna, seperti yang ditunjukkan pada gambar.

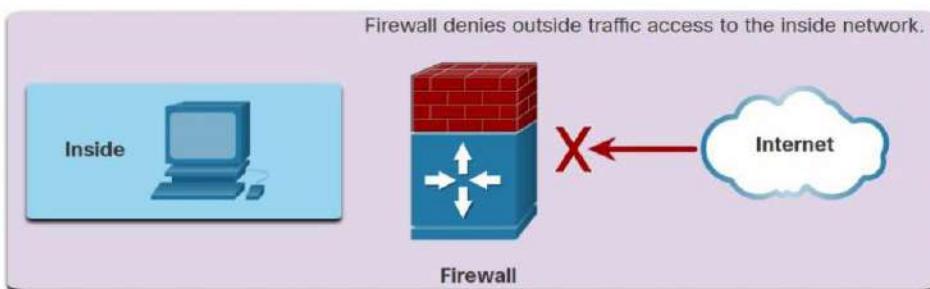
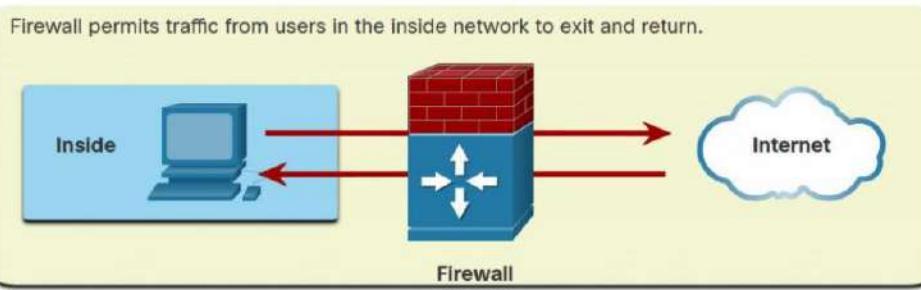


Firewall

Firewall adalah salah satu alat keamanan paling efektif yang tersedia untuk melindungi pengguna dari ancaman eksternal. Firewall melindungi komputer dan jaringan dengan mencegah **traffic** yang tidak diinginkan memasuki jaringan internal.

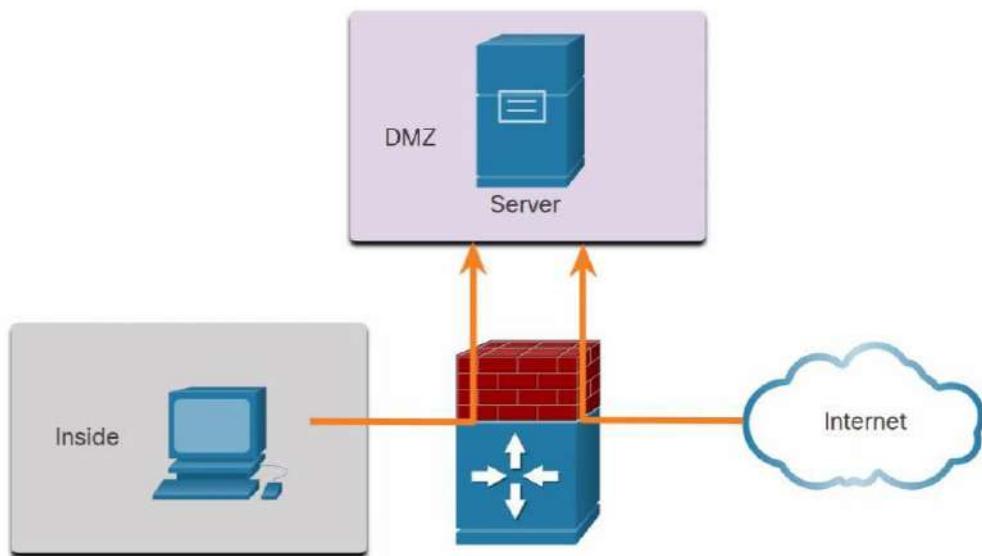
Firewall jaringan berada di antara dua atau lebih jaringan, mengontrol **traffic** di antara mereka, dan membantu mencegah akses yang tidak sah. Misalnya, topologi dalam gambar menggambarkan bagaimana firewall memungkinkan **traffic** dari host jaringan internal untuk keluar dari jaringan dan kembali ke jaringan dalam. Topologi bawah menggambarkan bagaimana **traffic** yang diprakarsai oleh jaringan luar (yaitu, internet) ditolak akses ke jaringan internal.

Operasi Firewall



Firewall dapat memungkinkan pengguna luar mengontrol akses ke layanan tertentu. Misalnya, server yang dapat diakses oleh pengguna luar biasanya terletak di jaringan khusus yang disebut sebagai **Demilitarized Zone** (DMZ), seperti yang ditunjukkan pada gambar. DMZ memungkinkan administrator jaringan untuk menerapkan kebijakan khusus untuk host yang terhubung ke jaringan tersebut.

Topologi Firewall dengan DMZ



Jenis Firewall

Produk firewall dikemas dalam berbagai bentuk. Produk-produk ini menggunakan teknik yang berbeda untuk menentukan apa yang akan diizinkan atau ditolak akses ke jaringan. Mereka termasuk yang berikut:

- **Packet filtering** – Mencegah atau mengizinkan akses berdasarkan alamat IP atau MAC
- **Application filtering** – Mencegah atau memungkinkan akses dengan jenis aplikasi tertentu berdasarkan nomor port
- **URL filtering** – Mencegah atau memungkinkan akses ke situs web berdasarkan URL atau kata kunci tertentu
- **Stateful packet inspection (SPI)** – Paket masuk harus menjadi tanggapan yang sah terhadap permintaan dari host internal. Paket yang tidak diminta diblokir kecuali diizinkan secara khusus. SPI juga dapat mencakup kemampuan untuk mengenali dan menyaring jenis serangan tertentu, seperti denial of service (DoS).

Keamanan Endpoint

Endpoint, atau host, adalah sistem komputer individu atau perangkat yang bertindak sebagai klien jaringan. Titik akhir umum adalah laptop, desktop, server, smartphone, dan tablet. Mengamankan perangkat endpoint adalah salah satu pekerjaan yang paling menantang dari administrator jaringan karena melibatkan sifat manusia. Sebuah perusahaan harus memiliki kebijakan yang terdokumentasi dengan baik dan karyawan harus menyadari aturan-aturan ini. Karyawan perlu dilatih tentang penggunaan jaringan yang tepat. Kebijakan sering termasuk penggunaan perangkat lunak antivirus dan pencegahan intrusi host. Solusi keamanan endpoint yang lebih komprehensif bergantung pada kontrol akses jaringan.

Keamanan Jaringan

Salah satu area jaringan yang membutuhkan perhatian khusus untuk menjaga keamanan adalah perangkat. Anda mungkin sudah memiliki kata sandi untuk komputer, ponsel pintar, atau tablet Anda. Apakah sekuat mungkin? Apakah Anda menggunakan alat lain untuk meningkatkan keamanan perangkat Anda? Topik ini memberitahu Anda caranya.

Cisco AutoSecure

Pengaturan keamanan diatur ke nilai default saat sistem operasi baru diinstal pada perangkat. Dalam kebanyakan kasus, tingkat keamanan ini tidak memadai. Untuk router Cisco, fitur Cisco **AutoSecure** dapat digunakan untuk membantu mengamankan sistem, seperti yang ditunjukkan dalam contoh.

```
Router# auto secure
      --- AutoSecure Configuration ---
*** AutoSecure configuration enhances the security of
the router but it will not make router absolutely secure
from all security attacks ***
```

Selain itu, ada beberapa langkah sederhana yang harus diambil yang berlaku untuk sebagian besar sistem operasi:

- Nama pengguna dan kata sandi default harus segera diubah.
- Akses ke **resource** sistem harus dibatasi hanya pada individu yang berwenang untuk menggunakan **resource** tersebut.
- Setiap layanan dan aplikasi yang tidak perlu harus dimatikan dan dihapus bila memungkinkan.

Seringkali, perangkat yang dikirim dari produsen telah duduk di gudang untuk jangka waktu tertentu dan tidak memiliki patch terbaru yang diinstal. Penting untuk memperbarui perangkat lunak apa pun dan menginstal patch keamanan apapun sebelum implementasi.

Password

Untuk melindungi perangkat jaringan, penting untuk menggunakan kata sandi yang kuat. Berikut adalah panduan standar untuk diikuti:

- Gunakan panjang kata sandi setidaknya delapan karakter, sebaiknya 10 atau lebih karakter. Kata sandi yang lebih panjang adalah kata sandi yang lebih aman.
- Buat kata sandi menjadi rumit. Sertakan campuran huruf besar dan huruf kecil, angka, simbol, dan spasi, jika diizinkan.
- Hindari kata sandi berdasarkan pengulangan, kata kamus umum, urutan huruf atau angka, nama pengguna, nama relatif atau hewan peliharaan, informasi biografi, seperti tanggal lahir, nomor ID, nama leluhur, atau potongan informasi lain yang mudah diidentifikasi.
- Sengaja salah mengeja password. Misalnya, Smith = Smyth = 5mYth atau Security = Secur1ty.
- Sering ganti kata sandi. Jika kata sandi tanpa sadar dikompromikan, jendela peluang bagi **threat actor** untuk menggunakan kata sandi terbatas.
- Jangan menuliskan kata sandi dan biarkan di tempat-tempat yang jelas seperti di atas meja atau monitor.

Tabel menunjukkan contoh kata sandi yang kuat dan lemah.

Kata Sandi lemah

| Kata Sandi lemah | Mengapa lemah |
|------------------|-------------------------------|
| rahasia | Kata sandi kamus sederhana |
| smith | Nama gadis ibu |
| toyota | Membuat mobil |
| bob1967 | Nama dan ulang tahun pengguna |
| Blueleaf23 | Kata-kata dan angka sederhana |

Kata Sandi yang Kuat

| Kata Sandi yang Kuat | Mengapa kuat |
|----------------------|--|
| b67n42d39c | Menggabungkan karakter alfanumerik |
| 12^h u4@1p7 | Menggabungkan karakter alfanumerik, simbol, dan termasuk spasi |

Pada router Cisco, ruang terdepan diabaikan untuk kata sandi, tetapi spasi setelah karakter pertama tidak. Oleh karena itu, salah satu metode untuk membuat kata sandi yang kuat adalah dengan menggunakan bilah spasi dan membuat frasa yang terbuat dari banyak kata. Ini disebut passphrase. Frasa sandi seringkali lebih mudah diingat daripada kata sandi sederhana. Hal ini juga lebih panjang dan lebih sulit untuk menebak.

Keamanan Kata Sandi Tambahan

Kata sandi yang kuat hanya berguna jika rahasia. Ada beberapa langkah yang dapat diambil untuk membantu memastikan bahwa kata sandi tetap rahasia pada router Cisco dan beralih termasuk ini:

- Mengenkripsi semua kata sandi plaintext
- Menetapkan panjang kata sandi minimum yang dapat diterima
- Menghalangi serangan menebak kata sandi brute-force
- Menonaktifkan akses mode EXEC istimewa yang tidak aktif setelah jumlah waktu tertentu.

Seperti yang ditunjukkan dalam konfigurasi sampel pada gambar, perintah konfigurasi global **service password-encryption** mencegah individu yang tidak berwenang melihat kata sandi plaintext dalam file konfigurasi. Perintah ini mengenkripsi semua kata sandi plaintext. Perhatikan dalam contoh, bahwa kata sandi “cisco” telah dienkripsi sebagai “03095A0F034F”.

Untuk memastikan bahwa semua kata sandi yang dikonfigurasi adalah minimal panjang yang ditentukan, gunakan perintah **security passwords min-length** dalam **mode global configuration**. Dalam gambar, setiap kata sandi baru yang dikonfigurasi harus memiliki panjang minimum delapan karakter.

Threat actor dapat menggunakan perangkat lunak cracking kata sandi untuk melakukan serangan brute-force pada perangkat jaringan. Serangan ini terus mencoba menebak kata sandi yang valid sampai satu bekerja. Gunakan **login block-for # attempts #**

within # dalam mode global configuration untuk mencegah jenis serangan ini. Dalam angka misalnya, **login block-for 120 attempts 3 within 60** command akan memblokir upaya login vty selama 120 detik jika ada tiga upaya login yang gagal dalam waktu 60 detik.

Administrator jaringan dapat menjadi terganggu dan secara tidak sengaja membiarkan sesi **Privileged EXEC** terbuka di terminal. Ini dapat memungkinkan akses **threat actor** internal untuk mengubah atau menghapus konfigurasi perangkat.

Secara default, router Cisco akan logout sesi EXEC setelah 10 menit tidak aktif. Namun, Anda dapat mengurangi pengaturan ini menggunakan perintah konfigurasi baris detik **exec-timeout menit**. Perintah ini dapat diterapkan konsol online, saluran tambahan, dan vty. Dalam gambar, kami mengatakan kepada perangkat Cisco untuk secara otomatis memutuskan pengguna yang tidak aktif pada garis vty setelah pengguna menganggur selama 5 menit dan 30 detik.

Aktifkan SSH

Telnet menyederhanakan akses perangkat jarak jauh, tetapi tidak aman. Data yang terkandung dalam paket Telnet ditransmisikan tanpa terenkripsi. Untuk alasan ini, sangat disarankan untuk mengaktifkan Secure Shell (SSH) pada perangkat untuk akses jarak jauh yang aman.

Hal ini dimungkinkan untuk mengkonfigurasi perangkat Cisco untuk mendukung SSH menggunakan enam langkah berikut:

- **Langkah 1. Konfigurasikan nama host perangkat yang unik.** Perangkat harus memiliki nama host unik selain default.
- **Langkah 2. Konfigurasikan nama domain IP.** Konfigurasikan nama domain IP jaringan dengan menggunakan **mode global configuration** perintah **ip domain name**
- **Langkah 3. Membuat sebuah key untuk encrypt SSH traffic.** SSH mengenkripsi lalu lintas antara **source** dan **destination**. Namun, untuk melakukannya, kunci otentikasi unik harus dihasilkan dengan menggunakan perintah konfigurasi global **crypto key generate rsa general-keys modulus**. Bit *modulus* menentukan ukuran kunci dan dapat dikonfigurasi dari 360 bit hingga 2048 bit. Semakin besar nilai bit, semakin aman kuncinya. Namun, nilai bit yang lebih besar juga membutuhkan waktu lebih lama untuk mengenkripsi dan mendekripsi informasi. Panjang modulus minimum yang direkomendasikan adalah 1024 bit.

- **Langkah 4. Verifikasi atau buat entri database lokal.** Buat entri nama pengguna database lokal menggunakan perintah konfigurasi global **username**. Dalam contoh, **secret** parameter digunakan sehingga kata sandi akan dienkripsi menggunakan MD5.
- **Langkah 5. Mengautentikasi terhadap database lokal.** Gunakan perintah konfigurasi **login local** pada line interface untuk mengautentikasi line vty terhadap database lokal.
- **Langkah 6. Aktifkan sesi SSH masuk vty.** Secara default, tidak ada sesi input yang diizinkan pada line vty. Anda dapat menentukan beberapa protokol input termasuk Telnet dan SSH menggunakan perintah **transport input {ssh | telnet}**.

Seperti yang ditunjukkan pada contoh, router R1 dikonfigurasi dalam domain span.com. Informasi ini digunakan bersama dengan nilai bit yang ditentukan dalam kunci crypto menghasilkan perintah **crypto key generate rsa general-keys modulus** untuk membuat kunci enkripsi.

Selanjutnya, entri database lokal untuk pengguna bernama Bob dibuat. Akhirnya, line vty dikonfigurasi untuk mengautentikasi terhadap database lokal dan hanya menerima sesi SSH yang masuk.

```
Router# configure terminal
Router(config)# hostname R1
R1(config)# ip domain name span.com
R1(config)# crypto key generate rsa general-keys modulus 1024
The name for the Keys will be: R1.span.com % The key modulus size is 1024 bits
% Generating 1024 bit RSA keys, keys will be non-exportable...[OK]
Dec 13 16:19:12.079: %SSH-5-ENABLED: SSH 1.99 has been enabled
R1(config)#
R1(config)# username Bob secret cisco
R1(config)# line vty 0 4
R1(config-line)# login local
R1(config-line)# transport input ssh
R1(config-line)# exit
R1(config)#

```

Nonaktifkan Layanan yang Tidak Digunakan

Router dan switch Cisco dimulai dengan daftar layanan aktif yang mungkin atau mungkin tidak diperlukan di jaringan Anda. Nonaktifkan layanan yang tidak digunakan untuk mengurangi **resource** sistem, seperti siklus CPU dan RAM, dan mencegah **threat actor** mengeksplorasi layanan ini. Jenis layanan yang menyala secara default akan bervariasi tergantung pada versi iOS. Misalnya, iOS-XE biasanya hanya akan memiliki port HTTPS dan DHCP yang terbuka. Anda dapat memverifikasi ini dengan perintah **show ip ports all**, seperti yang ditunjukkan pada contoh.

```

Router# show ip ports all
Proto Local Address          Foreign Address          State      PID/Program Name
TCB     Local Address          Foreign Address          (state)
tcp    :::443                 ::::*                  LISTEN     309/[IOS]HTTP CORE
tcp    *:443                 *:*                   LISTEN     309/[IOS]HTTP CORE
udp    *:67                  0.0.0.0:0             LISTEN     387/[IOS]DHCPD Receive
Router#

```

Versi iOS sebelum iOS-XE menggunakan perintah **show control-plane host open-ports**. Kami menyebutkan perintah ini karena Anda mungkin melihatnya di perangkat yang lebih lama. Outputnya serupa. Namun, perhatikan bahwa router yang lebih tua ini memiliki server HTTP yang tidak aman dan Telnet berjalan. Kedua layanan ini harus dinonaktifkan. Seperti yang ditunjukkan pada contoh, nonaktifkan HTTP dengan perintah konfigurasi global **no ip http server**. Nonaktifkan Telnet dengan menentukan hanya SSH dalam perintah konfigurasi baris, **transport input ssh**.

```

Router# show control-plane host open-ports
Active internet connections (servers and established)
Prot      Local Address      Foreign Address      Service      State
tcp        *:23              *:0                  Telnet      LISTEN
tcp        *:80              *:0                  HTTP CORE   LISTEN
udp        *:67              *:0                  DHCPD Receive LISTEN
Router# configure terminal
Router(config)# no ip http server
Router(config)# line vty 0 15
Router(config-line)# transport input ssh

```

BAB 17

~ *Build A Small
Network* ~

Judul Bab : Membuat sebuah jaringan kecil

Tujuan Bab : Mengimplementasikan sebuah desain jaringan untuk membuat jaringan kecil termasuk router, switch dan end device

Link Test Pemahaman : <https://s.id/-QycZ>

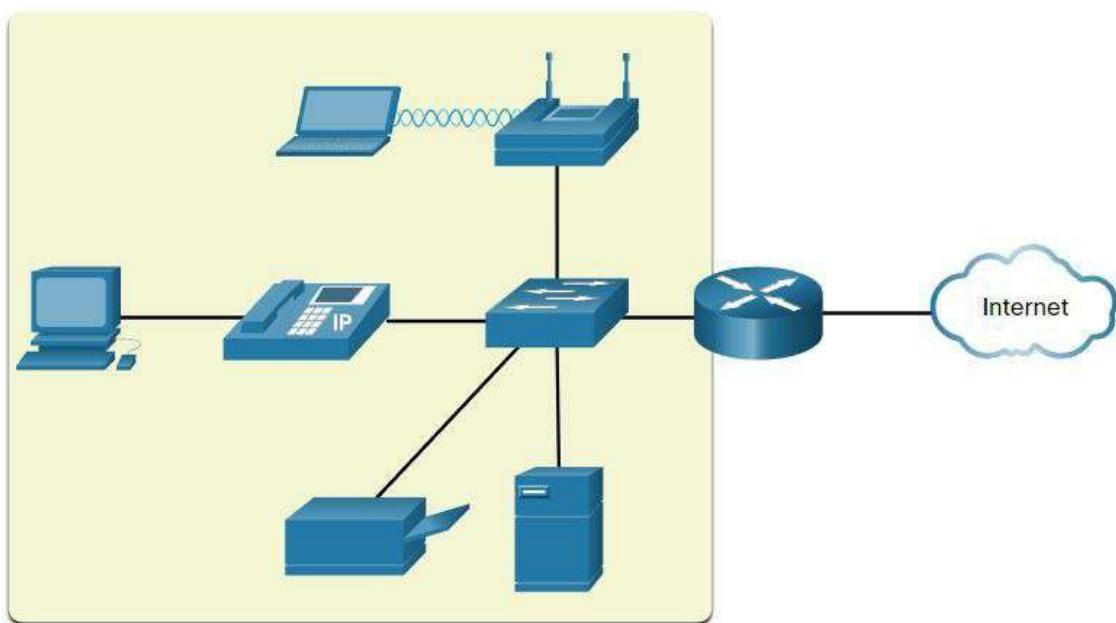
| Judul Materi | Tujuan Materi |
|--------------------------------------|--|
| Perangkat dalam jaringan kecil | Mengidentifikasi sebuah perangkat yang ada pada jaringan kecil |
| Aplikasi dan protokol jaringan kecil | Mengidentifikasi aplikasi dan protokol yang digunakan pada jaringan kecil |
| Skala ke jaringan lebih besar | Menjelaskan bagaimana jaringan kecil berfungsi sebagai dasar dari jaringan yang lebih besar. |
| Verifikasi Konektivitas | Menggunakan output dari perintah ping dan tracert untuk memverifikasi konektivitas dan membangun kinerja jaringan relatif. |
| Perintah IOS dan host | Menggunakan perintah host dan IOS untuk memperoleh informasi tentang perangkat dalam jaringan. |
| Metodologi Troubleshooting | Menjelaskan metodologi Troubleshooting jaringan yang umum |
| Skenario Troubleshooting | Troubleshooting dengan perangkat di jaringan. |

Perangkat Dalam Jaringan Kecil

Desain jaringan kecil biasanya sederhana. Jumlah dan jenis perangkat yang disertakan berkurang secara signifikan dibandingkan dengan jaringan yang lebih besar.

Misalnya, lihat contoh jaringan usaha kecil yang ditunjukkan pada gambar.

Topologi Jaringan Kecil



Jaringan kecil ini membutuhkan router, switch, dan **access point** nirkabel untuk menghubungkan pengguna kabel dan nirkabel, telefon IP, printer, dan server. Jaringan kecil biasanya memiliki koneksi WAN tunggal yang disediakan oleh DSL, kabel, atau koneksi Ethernet.

Jaringan besar memerlukan departemen TI untuk memelihara, mengamankan, dan memecahkan masalah perangkat jaringan dan untuk melindungi data organisasi. Mengelola jaringan kecil membutuhkan banyak keterampilan yang sama dengan yang diperlukan untuk mengelola yang lebih besar. Jaringan kecil dikelola oleh teknisi TI lokal atau oleh profesional yang dikontrak.

Pemilihan Perangkat untuk Jaringan Kecil

Seperti jaringan besar, jaringan kecil memerlukan perencanaan dan desain untuk memenuhi kebutuhan pengguna. Perencanaan memastikan bahwa semua persyaratan, faktor biaya, dan opsi penyebaran diberikan pertimbangan yang tepat.

Salah satu pertimbangan desain pertama adalah jenis **intermediary device** yang digunakan untuk mendukung jaringan.

Biaya

Biaya switch atau router ditentukan oleh kapasitas dan fiturnya. Ini termasuk jumlah dan jenis port yang tersedia dan kecepatan backplane. Faktor-faktor lain yang mempengaruhi biaya adalah kemampuan manajemen jaringan, teknologi keamanan tertanam, dan teknologi switching canggih opsional. Biaya kabel diperlukan untuk menghubungkan setiap perangkat pada jaringan juga harus dipertimbangkan. Elemen kunci lain yang mempengaruhi pertimbangan biaya adalah jumlah redundansi untuk dimasukkan ke dalam jaringan.

Kecepatan dan Jenis Port /Interface

Memilih jumlah dan jenis port pada router atau switch adalah keputusan penting. Komputer yang lebih baru memiliki 1 GBPS NICs built-in. Beberapa server bahkan mungkin memiliki port 10 Gbps. Meskipun lebih mahal, memilih perangkat Layer 2 yang dapat mengakomodasi peningkatan kecepatan memungkinkan jaringan berkembang tanpa mengganti perangkat pusat.

Upgrade

Perangkat jaringan tersedia dalam konfigurasi fisik tetap dan modular. Perangkat konfigurasi tetap memiliki jumlah dan jenis port atau **interface** tertentu dan tidak dapat diperluas. Perangkat modular memiliki slot ekspansi untuk menambahkan modul baru saat persyaratan berkembang. Switch tersedia dengan port tambahan untuk uplink berkecepatan tinggi. Router dapat digunakan untuk menghubungkan berbagai jenis jaringan. Perawatan harus diambil untuk memilih modul dan **interface** yang sesuai untuk media tertentu.

Fitur dan Layanan Sistem Operasi

Perangkat jaringan harus memiliki sistem operasi yang dapat mendukung persyaratan organisasi seperti berikut:

- Layer 3 switching
- Network Address Translation (NAT)
- Dynamic Host Configuration Protocol (DHCP)
- Security
- Quality of service (QoS)
- Voice over IP (VoIP)

Pengalamatan IP untuk Jaringan Kecil

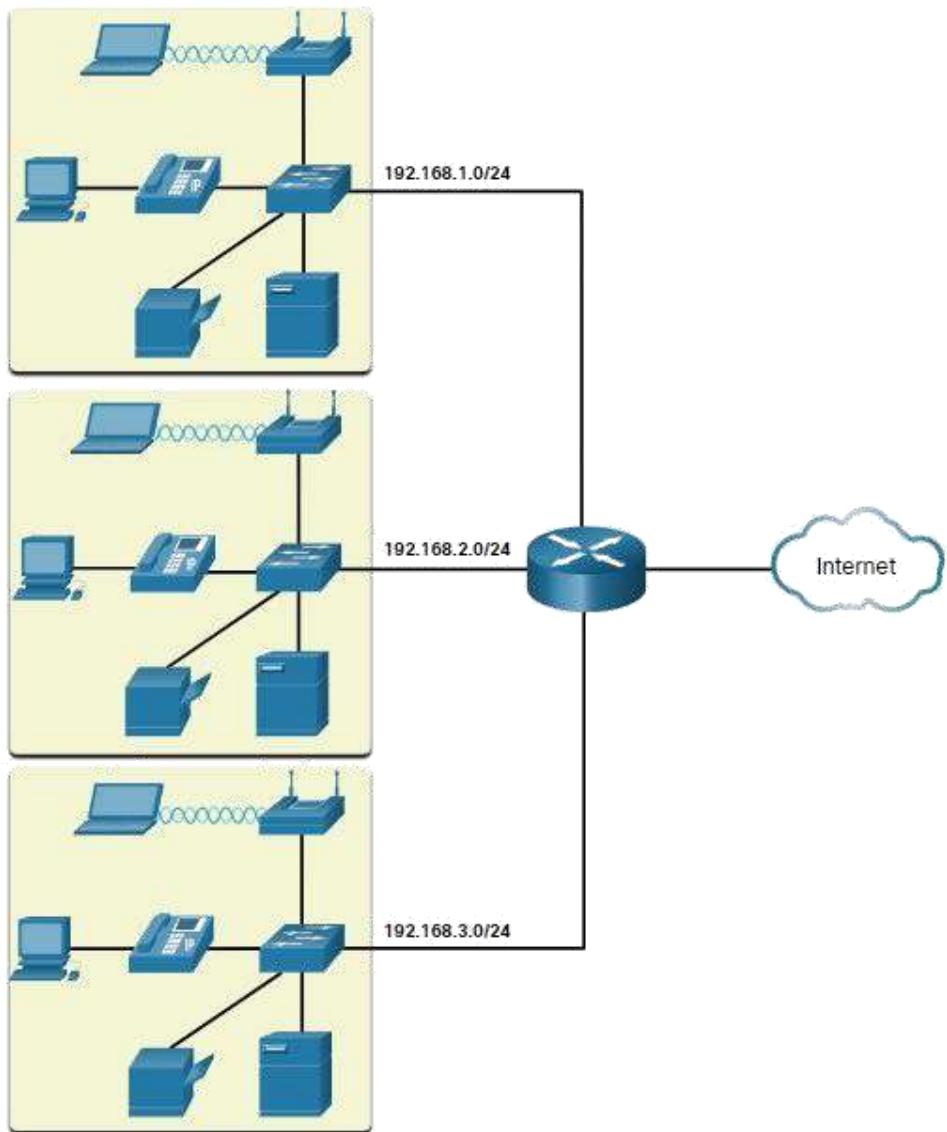
Saat menerapkan jaringan, buat skema pengalamatan IP. Semua host dan perangkat dalam internetwork harus memiliki alamat yang unik.

Perangkat yang akan menjadi faktor dalam skema pengalamatan IP meliputi hal-hal berikut:

- Perangkat **end user** – Jumlah dan jenis koneksi (yaitu, kabel, nirkabel, akses jarak jauh)
- Server dan perangkat periferal (misalnya, printer dan kamera keamanan)
- **Intermediary device** termasuk **switch** dan **access point**

Disarankan agar Anda merencanakan, mendokumentasikan, dan memelihara skema pengalamatan IP berdasarkan jenis perangkat. Penggunaan skema pengalamatan IP yang direncanakan membuatnya lebih mudah untuk mengidentifikasi jenis perangkat dan untuk memecahkan masalah, seperti misalnya, ketika memecahkan masalah **traffic** jaringan dengan penganalisis protokol.

Misalnya, lihat topologi organisasi kecil hingga menengah dalam gambar.

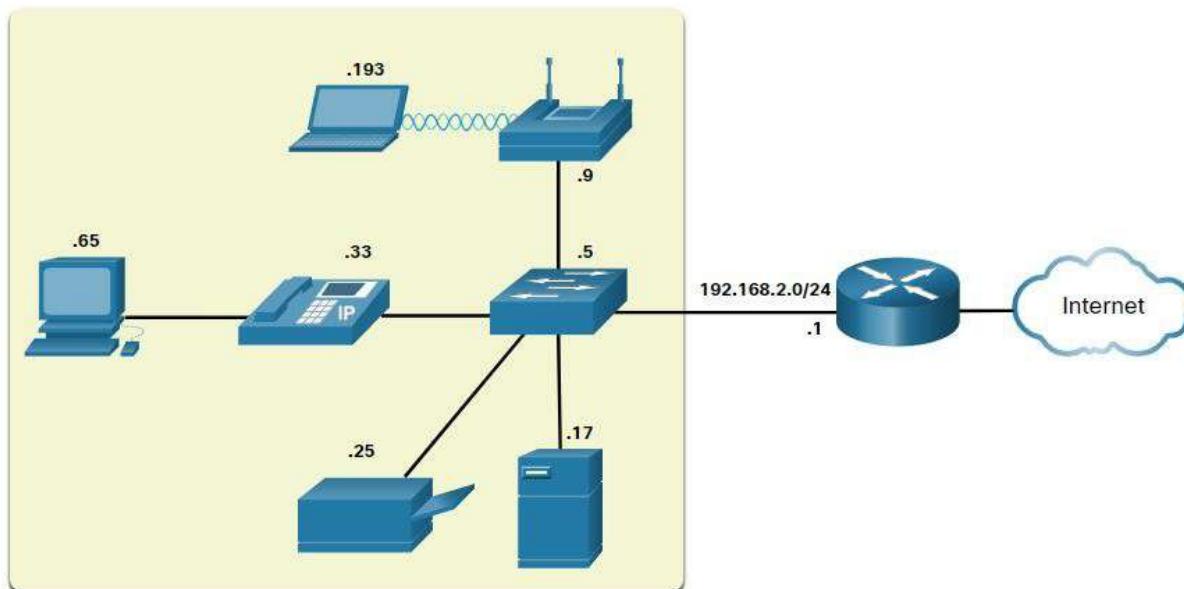


Organisasi ini membutuhkan tiga LAN pengguna (yaitu, 192.168.1.0/24, 192.168.2.0/24, dan 192.168.3.0/24). Organisasi telah memutuskan untuk menerapkan skema pengalamatan IP yang konsisten untuk setiap LAN 192.168.x.0/24 menggunakan rencana beri

| Tipe Perangkat | Rentang Alamat IP yang Dapat Ditetapkan | Dirangkum sebagai... |
|------------------------------|---|----------------------|
| Gateway default (Router) | 192.168.x.1 – 192.168.x.2 | 192.168.x.0/30 |
| Switch (maks 2) | 192.168.x.5 – 192.168.x.6 | 192.168.x.4/30 |
| Access point (maks 6) | 192.168.x.9 – 192.168.x.14 | 192.168.x.8/29 |
| Server (maks 6) | 192.168.x.17 – 192.168.x.22 | 192.168.x.16/29 |
| Printer (maks 6) | 192.168.x.25 – 192.168.x.30 | 192.168.x.24/29 |
| Ponsel IP (maks 6) | 192.168.x.33 – 192.168.x.38 | 192.168.x.32/29 |
| Perangkat kabel (maks 62) | 192.168.x.65 – 192.168.x.126 | 192.168.x.64/26 |
| Perangkat nirkabel (maks 62) | 192.168.x.193 – 192.168.x.254 | 192.168.x.192/26 |

Gambar menampilkan contoh perangkat jaringan 192.168.2.0/24 dengan alamat IP yang ditetapkan menggunakan skema pengalamanan IP yang telah ditentukan.

Diagram ini adalah topologi LAN kecil dengan alamat jaringan 192.168.2.0/24. Ini menunjukkan berbagai perangkat



Misalnya, alamat IP gateway default adalah 192.168.2.1/24, switch adalah 192.168.2.5/24, server adalah 192.168.2.17/24, dll..

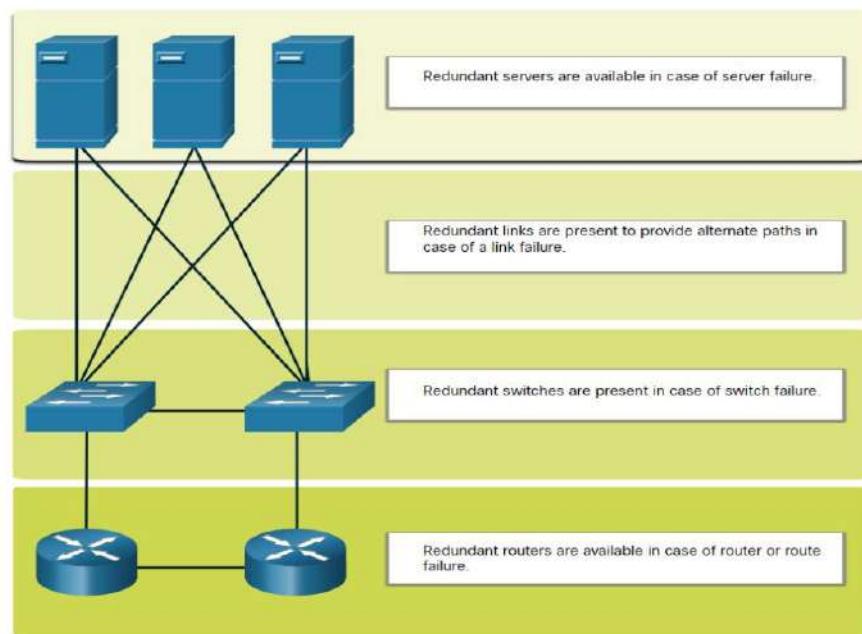
Perhatikan bahwa rentang alamat IP yang dapat ditetapkan sengaja dialokasikan pada batas subnet network untuk menyederhanakan meringkas tipe grup. Misalnya, asumsikan **switch** lain dengan alamat IP 192.168.2.6 ditambahkan ke jaringan. Untuk mengidentifikasi semua **switch** dalam kebijakan jaringan, administrator dapat menentukan alamat jaringan yang diringkas 192.168.x.4/30.

Redundansi dalam Jaringan Kecil

Bagian penting lain dari desain jaringan adalah reliability/keandalan. Bahkan usaha kecil sering sangat bergantung pada jaringan mereka untuk operasi bisnis. Kegagalan jaringan bisa sangat mahal.

Untuk mempertahankan tingkat keandalan yang tinggi, *redundansi* diperlukan dalam desain jaringan. Redundansi membantu menghilangkan satu titik kegagalan.

Ada banyak cara untuk mencapai redundansi dalam jaringan. Redundansi dapat dilakukan dengan memasang peralatan duplikat, tetapi juga dapat dicapai dengan menyediakan link jaringan duplikat untuk area kritis, seperti yang ditunjukkan pada gambar.

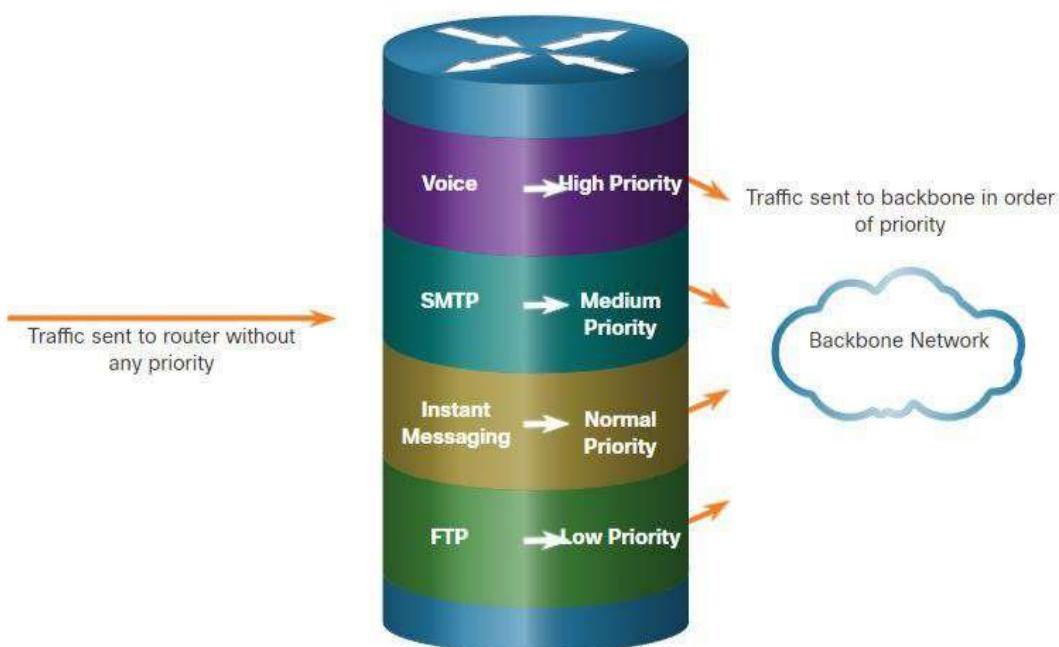


Jaringan kecil biasanya menyediakan satu titik keluar ke internet melalui satu atau lebih gateway default. Jika router gagal, seluruh jaringan kehilangan koneksi ke internet. Untuk alasan ini, mungkin disarankan bagi bisnis kecil untuk membayar penyedia layanan kedua sebagai cadangan.

Manajemen Traffic

Tujuan untuk desain jaringan yang baik, bahkan untuk jaringan kecil, adalah untuk meningkatkan produktivitas karyawan dan meminimalkan downtime jaringan. Administrator jaringan harus mempertimbangkan berbagai jenis **traffic** dan perlakuan mereka dalam desain jaringan.

Router dan switch dalam jaringan kecil harus dikonfigurasi untuk mendukung **traffic** real-time, seperti suara dan video, dengan cara yang tepat relatif terhadap **traffic** data lainnya. Bahkan, desain jaringan yang baik akan menerapkan Quality Of Service (QoS) untuk mengklasifikasikan **traffic** dengan hati-hati sesuai dengan prioritas selama masa **congestion**, seperti yang ditunjukkan pada gambar.



Queue prioritas memiliki empat **queue**. **Queue** prioritas tinggi selalu dikosongkan terlebih dahulu.

Aplikasi dan Protokol jaringan kecil

Materi sebelumnya membahas komponen jaringan kecil, serta beberapa pertimbangan desain. Pertimbangan ini diperlukan ketika Anda hanya menyiapkan jaringan. Setelah Anda mengaturnya, jaringan Anda masih memerlukan jenis aplikasi dan protokol tertentu agar berfungsi.

Aplikasi Umum

Jaringan ini hanya berguna seperti aplikasi yang ada di dalamnya. Ada dua bentuk program perangkat lunak atau proses yang menyediakan akses ke jaringan: aplikasi jaringan dan layanan **application layer**.

Aplikasi Jaringan

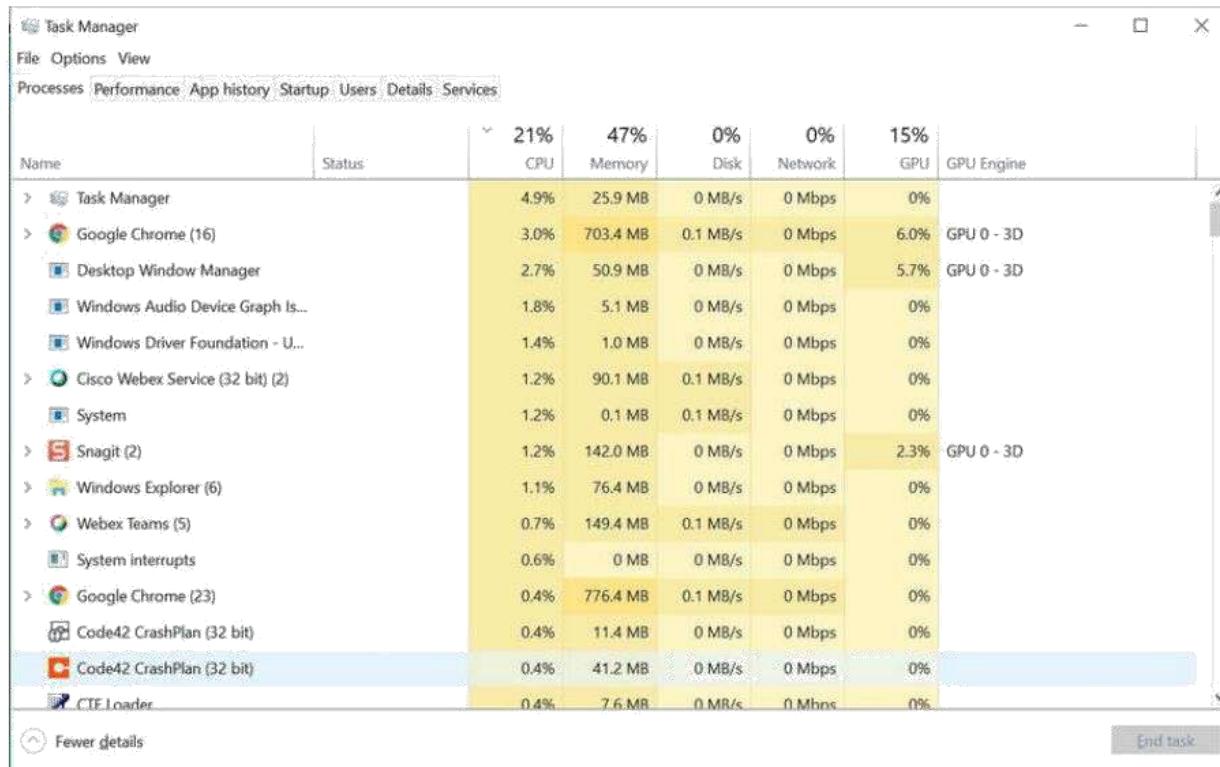
Aplikasi adalah program perangkat lunak yang digunakan untuk berkomunikasi melalui jaringan. Beberapa aplikasi **end user** sadar jaringan, yang berarti bahwa mereka menerapkan protokol **application layer** dan mampu berkomunikasi langsung dengan **bottom layer** protokol. Klien email dan browser web adalah contoh dari jenis aplikasi ini.

Layanan Application layer

Program lain mungkin memerlukan bantuan layanan **application layer** untuk menggunakan **resource** jaringan seperti transfer file atau penampung cetak jaringan. Meskipun transparan bagi karyawan, layanan ini adalah program yang berinteraksi dengan jaringan dan menyiapkan data untuk transfer. Berbagai jenis data, baik teks, grafik atau video, memerlukan layanan jaringan yang berbeda untuk memastikan bahwa mereka dipersiapkan dengan baik untuk diproses oleh fungsi yang terjadi pada **bottom layer** model OSI.

Setiap aplikasi atau layanan jaringan menggunakan protokol, yang menentukan standar dan format data yang akan digunakan. Tanpa protokol, jaringan data tidak akan memiliki cara umum untuk memformat dan mengarahkan data. Untuk memahami fungsi berbagai layanan jaringan, perlu untuk menjadi akrab dengan protokol yang mendasari yang mengatur operasi mereka.

Gunakan Task Manager untuk melihat aplikasi, proses, dan layanan saat ini yang berjalan di PC Windows, seperti yang ditunjukkan pada gambar.



Protokol Umum

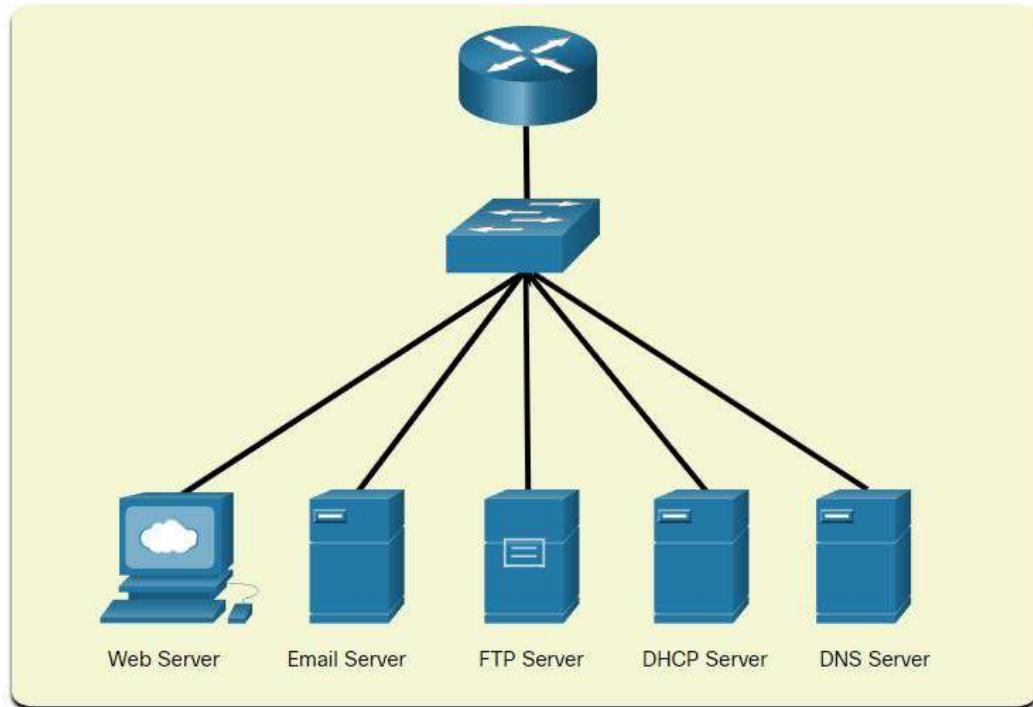
Sebagian besar pekerjaan teknisi, baik dalam jaringan kecil atau besar, dalam beberapa cara akan terlibat dengan protokol jaringan. Protokol jaringan mendukung aplikasi dan layanan yang digunakan oleh karyawan dalam jaringan kecil.

Administrator jaringan biasanya memerlukan akses ke perangkat jaringan dan server. Dua solusi akses jarak jauh yang paling umum adalah Telnet dan Secure Shell (SSH). Layanan SSH adalah alternatif yang aman untuk Telnet. Ketika terhubung, administrator dapat mengakses perangkat server SSH seolah-olah mereka login secara lokal.

SSH digunakan untuk membuat koneksi akses jarak jauh yang aman antara klien SSH dan perangkat berkemampuan SSH lainnya:

- **Perangkat** jaringan – Perangkat jaringan (misalnya, router, switch, access point, dll) harus mendukung SSH untuk menyediakan akses jarak jauh layanan server SSH kepada klien.
- **Server** – Server (misalnya, server web, server email, dll.) harus mendukung akses jarak jauh layanan server SSH kepada klien.

Administrator jaringan juga harus mendukung server jaringan umum dan protokol jaringan terkait yang diperlukan, seperti yang ditunjukkan pada gambar.



1. Server

- Klien web dan server web bertukar **traffic** web menggunakan Hypertext Transfer Protocol (HTTP).
- Hypertext Transfer Protocol Secure (HTTPS) digunakan untuk komunikasi web yang aman.

2. Email Server

- Server email dan klien menggunakan Simple Mail Transfer Protocol (SMTP) untuk mengirim email.
- Klien email menggunakan Post Office Protocol (POP3) atau Internet Message Access Protocol (IMAP) untuk mengambil email.
- Penerima ditentukan menggunakan user@xyz.xxx format.

3. FTP Server

- Layanan File Transfer Protocol (FTP) memungkinkan file untuk diunduh dan diunggah antara klien dan server FTP.
- FTP Secure (FTPS) dan Secure FTP (SFTP) digunakan untuk mengamankan pertukaran file FTP.

4. DHCP Server

- Dynamic Host Configuration Protocol (DHCP) digunakan oleh klien untuk memperoleh konfigurasi IP (yaitu, alamat IP, subnet mask, gateway default dan banyak lagi) dari server DHCP.

5. DNS Server

- Domain Name Server (DNS) menyelesaikan nama domain ke alamat IP (misalnya, cisco.com = 72.163.4.185)
- DNS menyediakan alamat IP dari situs web (yaitu, nama domain) ke host yang meminta.

Catatan: Server dapat menyediakan beberapa layanan jaringan. Misalnya, server bisa menjadi email, FTP, dan server SSH.

Protokol jaringan ini terdiri dari toolset dasar dari seorang profesional jaringan. Masing-masing protokol jaringan ini mendefinisikan:

- Proses di kedua ujung sesi komunikasi
- Jenis pesan
- Sintaks dari pesan
- Arti dari **field information**
- Bagaimana pesan dikirim dan respons yang diharapkan
- Interaksi dengan **bottom layer** berikutnya

Banyak perusahaan telah menetapkan kebijakan menggunakan versi aman (misalnya, SSH, SFTP, dan HTTPS) dari protokol ini bila memungkinkan.

Aplikasi Suara dan Video

Bisnis saat ini semakin menggunakan telepon IP dan media streaming untuk berkomunikasi dengan pelanggan dan mitra bisnis. Banyak organisasi yang memungkinkan karyawan mereka untuk bekerja dari jarak jauh. Seperti yang ditunjukkan oleh gambar tersebut, banyak pengguna mereka masih memerlukan akses ke perangkat lunak dan file perusahaan, serta dukungan untuk aplikasi suara dan video.



Administrator jaringan harus memastikan peralatan yang tepat dipasang di jaringan dan bahwa perangkat jaringan dikonfigurasi untuk memastikan pengiriman prioritas.

Infrastruktur

- Infrastruktur jaringan harus mendukung aplikasi real-time.
- Perangkat dan kabel yang ada harus diuji dan divalidasi.
- Produk jaringan yang lebih baru mungkin diperlukan.

VoIP

- Perangkat VoIP mengubah sinyal telefon analog menjadi paket IP digital.
- Biasanya, VOIP lebih murah daripada solusi telefon IP, tetapi kualitas komunikasi tidak memenuhi standar yang sama.
- Suara jaringan kecil dan video melalui IP dapat diselesaikan dengan menggunakan Skype dan versi non-perusahaan cisco WebEx.

Telepon IP

- Ponsel IP melakukan konversi suara-ke-IP dengan menggunakan server khusus untuk kontrol panggilan dan pensinyalan.
- Banyak vendor menyediakan solusi telepon IP bisnis kecil seperti produk Cisco Business Edition 4000 Series.

Aplikasi Real-Time

- Jaringan harus mendukung mekanisme kualitas layanan (QoS) untuk meminimalkan masalah latensi untuk aplikasi streaming real-time.
- Real-Time Transport Protocol (RTP) dan Real-Time Transport Control Protocol (RTCP) adalah dua protokol yang mendukung persyaratan ini.

Skala ke jaringan yang lebih besar

Jika jaringan Anda adalah untuk bisnis kecil, mungkin, Anda ingin bisnis itu tumbuh, dan jaringan Anda tumbuh bersamanya. Ini disebut scaling jaringan, dan ada beberapa praktik terbaik untuk melakukan ini.

Pertumbuhan Jaringan Kecil

Pertumbuhan adalah proses alami bagi banyak usaha kecil, dan jaringan mereka harus tumbuh sesuai dengan itu. Idealnya, administrator jaringan memiliki cukup lead-time untuk membuat keputusan cerdas tentang menumbuhkan jaringan sejalan dengan pertumbuhan perusahaan.

Untuk skala jaringan, beberapa elemen diperlukan:

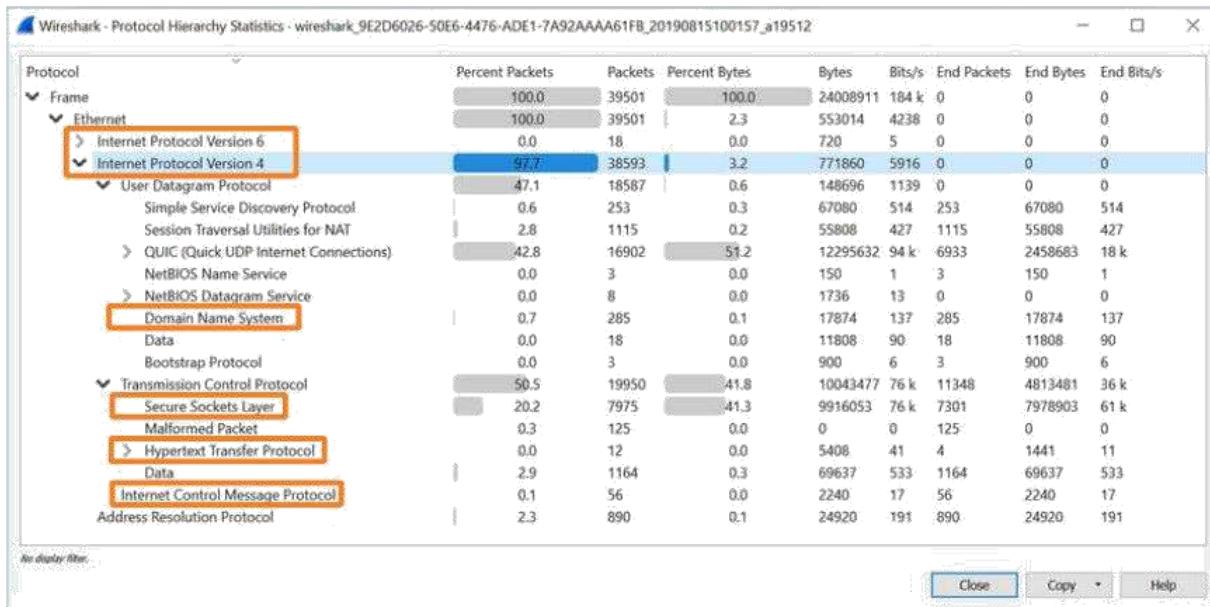
- **Dokumentasi jaringan** – Physical dan logical topology
- **Inventaris perangkat** – Daftar perangkat yang menggunakan atau terdiri dari jaringan
- **Anggaran** – Anggaran TI terperinci, termasuk anggaran pembelian peralatan tahun fiskal
- **Analisis traffic** – Protokol, aplikasi, dan layanan dan persyaratan traffic masing-masing harus didokumentasikan

Elemen-elemen ini digunakan untuk menginformasikan pengambilan keputusan yang menyertai penskalaan jaringan kecil.

Analisis Protokol

Seiring pertumbuhan jaringan, menjadi penting untuk menentukan bagaimana mengelola traffic jaringan. Penting untuk memahami jenis traffic yang melintasi jaringan serta arus traffic saat ini. Ada beberapa alat manajemen jaringan yang dapat digunakan untuk tujuan ini. Namun, penganalisis protokol sederhana seperti Wireshark juga dapat digunakan.

Misalnya, menjalankan Wireshark pada beberapa host utama dapat mengungkapkan jenis traffic jaringan yang mengalir melalui jaringan. Gambar berikut menampilkan statistik hierarki protokol Wireshark untuk host Windows pada jaringan kecil.



Tangkapan layar mengungkapkan host menggunakan protokol IPv6 dan IPv4. Output spesifik IPv4 juga mengungkapkan bahwa host telah menggunakan DNS, SSL, HTTP, ICMP, dan protokol lainnya.

Untuk menentukan pola arus traffic, penting untuk melakukan hal berikut:

- Tangkap traffic selama waktu pemanfaatan puncak untuk mendapatkan representasi yang baik dari berbagai jenis traffic.
- Lakukan penangkapan pada segmen dan perangkat jaringan yang berbeda karena beberapa traffic akan lokal ke segmen tertentu.

Informasi yang dikumpulkan oleh analisis protokol dievaluasi berdasarkan sumber dan tujuan traffic, serta jenis traffic yang dikirim. Analisis ini dapat digunakan untuk membuat keputusan tentang cara mengelola traffic dengan lebih efisien. Hal ini dapat dilakukan dengan mengurangi arus traffic yang tidak perlu atau mengubah pola aliran sama sekali dengan memindahkan server, misalnya.

Terkadang, hanya merlokasi server atau layanan ke segmen jaringan lain meningkatkan kinerja jaringan dan mengakomodasi kebutuhan traffic yang terus meningkat. Di lain waktu, mengoptimalkan kinerja jaringan membutuhkan desain ulang dan intervensi jaringan yang besar.

Pemanfaatan Jaringan Karyawan

Selain memahami perubahan tren traffic, administrator jaringan harus menyadari bagaimana penggunaan jaringan berubah. Banyak sistem operasi menyediakan alat built-in untuk menampilkan informasi tersebut. Misalnya, host Windows menyediakan alat seperti Task Manager, Event Viewer, dan alat Penggunaan Data.

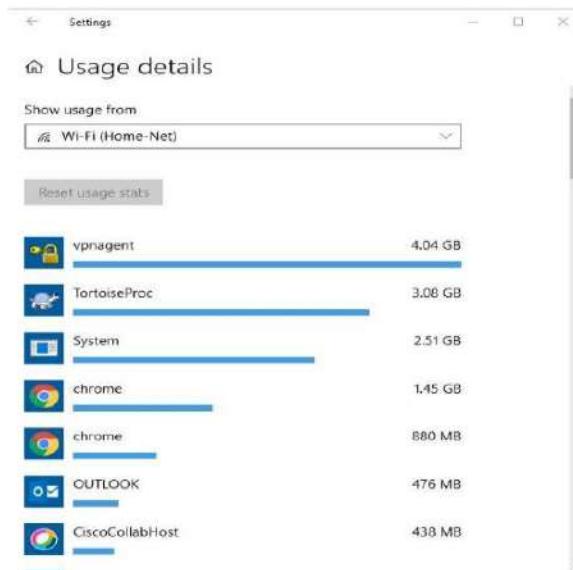
Alat-alat ini dapat digunakan untuk menangkap “snapshot” informasi seperti berikut:

- Versi OS dan OS
- Pemanfaatan CPU
- Pemanfaatan RAM
- Pemanfaatan drive
- Aplikasi non-jaringan
- Aplikasi jaringan

Mendokumentasikan snapshot untuk karyawan dalam jaringan kecil selama periode waktu tertentu sangat berguna untuk mengidentifikasi persyaratan protokol yang berkembang dan arus traffic terkait. Pergeseran dalam pemanfaatan **resource** mungkin memerlukan administrator jaringan untuk menyesuaikan alokasi **resource** jaringan yang sesuai.

Alat Penggunaan Data Windows 10 sangat berguna untuk menentukan aplikasi mana yang menggunakan layanan jaringan pada host. Alat Penggunaan Data diakses menggunakan **Pengaturan > Network & Internet > Penggunaan data > Interface Network** (dari 30 hari terakhir).

Contoh dalam gambar adalah menampilkan aplikasi yang berjalan pada pengguna jarak jauh windows 10 host menggunakan koneksi jaringan Wi-Fi lokal.



Verifikasi Konektivitas

Apakah jaringan Anda kecil dan baru, atau Anda menskalakan jaringan yang ada, Anda akan selalu ingin dapat memverifikasi bahwa komponen Anda terhubung dengan benar satu sama lain dan ke internet. Materi ini membahas beberapa utilitas yang dapat Anda gunakan untuk memastikan bahwa jaringan Anda terhubung.

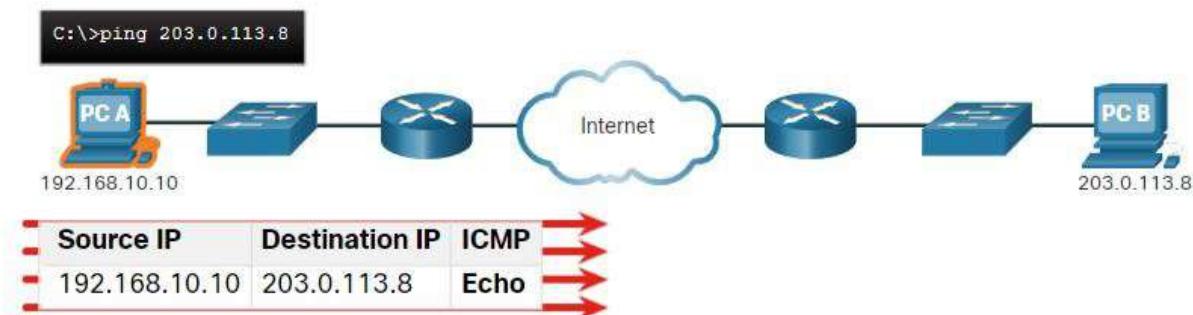
Verifikasi Konektivitas dengan Ping

Perintah **ping** adalah cara paling efektif untuk menguji konektivitas Layer 3 dengan cepat antara alamat IP source dan destination. Perintah ini juga menampilkan berbagai statistik waktu pulang pergi.

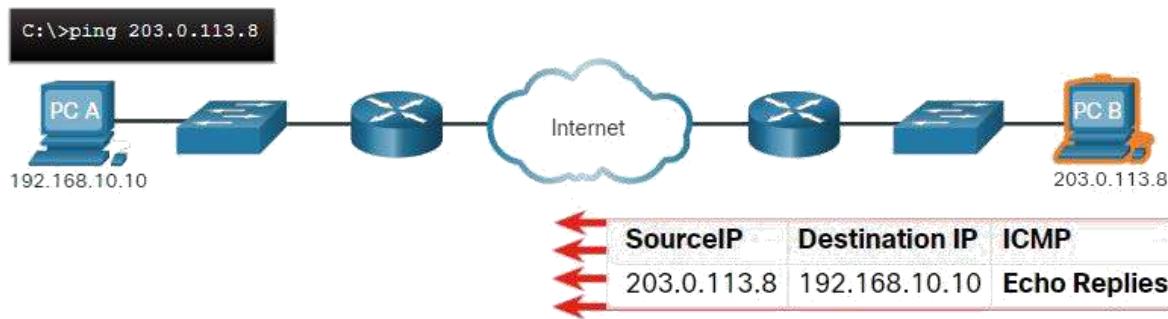
Secara khusus, perintah **ping** menggunakan internet control message protocol (ICMP) echo request (ICMP Type 8) dan echo reply (ICMP Type 0) pesan. Perintah **ping** tersedia di sebagian besar sistem operasi termasuk Windows, Linux, macOS, dan Cisco iOS.

Pada host Windows 10, perintah **ping** mengirimkan empat pesan permintaan echo ICMP berturut-turut dan mengharapkan empat balasan echo ICMP berturut-turut dari destination.

Misalnya, asumsikan PC A ping PC B. Seperti yang ditunjukkan pada gambar, PC Host Windows mengirimkan empat pesan permintaan echo ICMP berturut-turut. kadang-kadang disebut sebagai echo ICMP, untuk PC B (yaitu, 203.0.113.8).



Host destination menerima dan memproses echo ICMP. Seperti yang ditunjukkan pada gambar, PC B merespons dengan mengirim empat pesan balasan echo ICMP ke PC A.



Seperti yang ditunjukkan dalam output perintah, PC-A telah menerima balasan echo dari PC-B yang memverifikasi koneksi jaringan Layer 3.

```
C:\Users\PC-A> ping 10.1.1.10
Pinging 10.1.1.10 with 32 bytes of data:
Reply from 10.1.1.10: bytes=32 time=47ms TTL=51
Reply from 10.1.1.10: bytes=32 time=60ms TTL=51
Reply from 10.1.1.10: bytes=32 time=53ms TTL=51
Reply from 10.1.1.10: bytes=32 time=50ms TTL=51
Ping statistics for 10.1.1.10:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 47ms, Maximum = 60ms, Average = 52ms
C:\Users\PC-A>
```

Output memvalidasi koneksi Layer 3 antara PC A dan PC B.

Output perintah **ping** Cisco iOS bervariasi dari host Windows. Misalnya, ping iOS mengirim lima pesan echo ICMP, seperti yang ditunjukkan dalam output.

```
R1# ping 10.1.1.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
R1#
```

Perhatikan karakter output !!!!!. Perintah **ping** iOS menampilkan indikator untuk setiap balasan echo ICMP yang diterima. Tabel mencantumkan karakter output yang paling umum dari perintah **ping**.

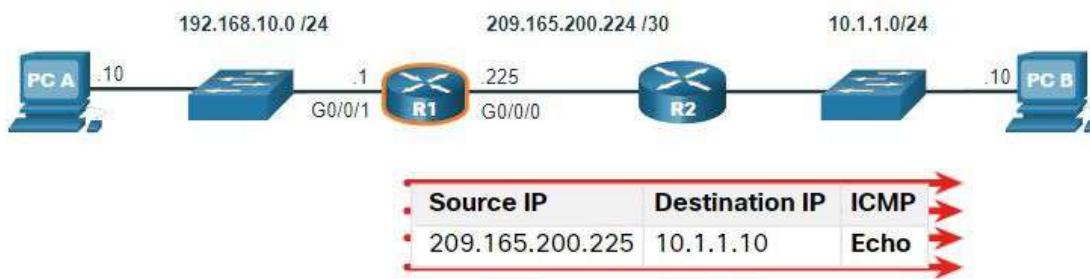
Indikator Ping iOS

| Elemen | Deskripsi |
|--------|--|
| ! | Tanda seru menunjukkan penerimaan pesan balasan echo yang berhasil. Ini memvalidasi koneksi Layer 3 antara source dan destination. |
| . | Periode berarti waktu kadaluarsa menunggu pesan balasan echo. Ini menunjukkan masalah koneksi terjadi di suatu tempat di sepanjang jalan. |
| U | Uppercase U menunjukkan router di sepanjang jalur ditanggapi dengan error message “destination tidak terjangkau” ICMP Tipe 3. Alasan yang mungkin termasuk router tidak tahu arah ke jaringan destination atau tidak dapat menemukan host di jaringan destination. |

Catatan: Kemungkinan balasan ping lainnya termasuk Q, M, ?, atau &. Namun, makna ini berada di luar cakupan untuk materi ini.

Ping Extended

Ping standar menggunakan alamat IP interface yang paling dekat dengan jaringan destination sebagai source ping. Alamat IP source dari perintah **ping 10.1.1.10** pada R1 adalah interface G0/0/0 (yaitu, 209.165.200.225), seperti yang diilustrasikan dalam contoh.



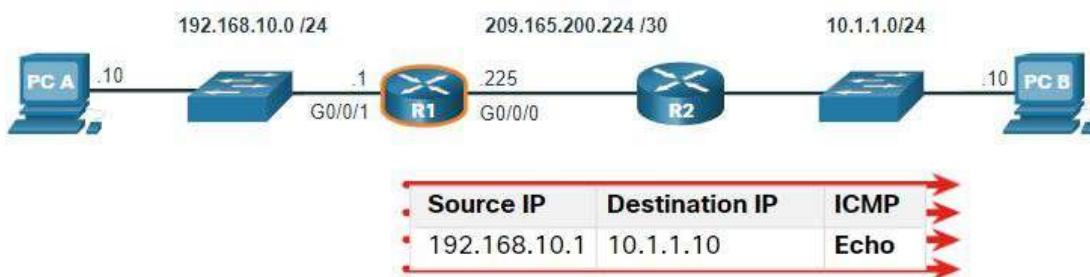
Cisco iOS menawarkan mode “extended” dari perintah ping. Mode ini memungkinkan pengguna untuk membuat jenis ping khusus dengan menyesuaikan parameter yang terkait dengan operasi perintah.

Extended ping dimasukkan dalam Privileged EXEC mode dengan mengetik **ping** tanpa alamat IP destination. Anda kemudian akan diberikan beberapa petunjuk untuk menyesuaikan **pingextended**.

Catatan: Menekan **Enter** menerima nilai default yang ditunjukkan.

Misalnya, asumsikan Anda ingin menguji konektivitas dari LAN R1 (yaitu, 192.168.10.0/24) ke LAN 10.1.1.0. Ini dapat diverifikasi dari PC A. Namun, **ping extended** dapat dikonfigurasi pada R1 untuk menentukan alamat source yang berbeda.

Seperti yang diilustrasikan dalam contoh, alamat IP source dari perintah **ping** extended pada R1 dapat dikonfigurasi untuk menggunakan alamat IP interface G0 /0/1 (yaitu, 192.168.10.1).



Output perintah berikut mengkonfigurasi **ping** extended pada R1 dan menentukan alamat IP source menjadi interface G0 /0/1 (yaitu, 192.168.10.1).

Catatan: Perintah **ping ipv6** digunakan untuk ping extended IPv6.

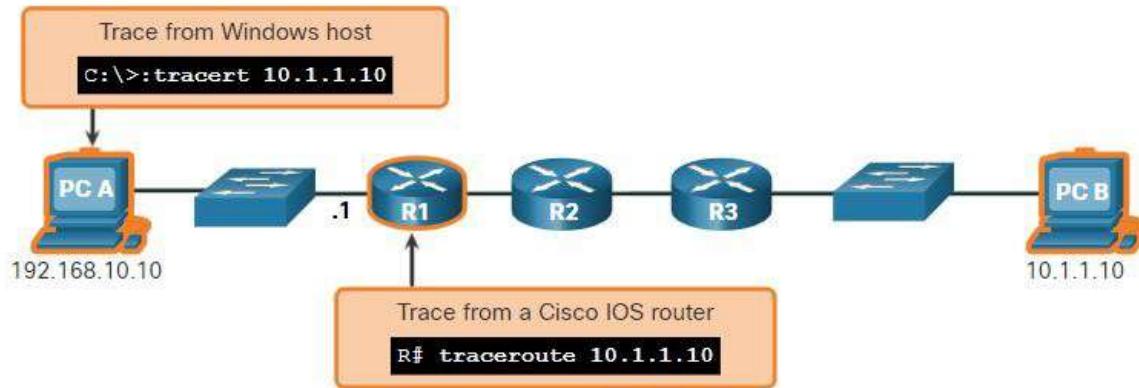
Verifikasi Konektivitas dengan Traceroute

Perintah **ping** berguna untuk menentukan dengan cepat apakah ada masalah konektivitas Layer 3. Namun, itu tidak mengidentifikasi dimana masalah berada di sepanjang jalan.

Traceroute dapat membantu menemukan area masalah Layer 3 dalam jaringan. Jejak mengembalikan daftar hop sebagai paket diarahkan melalui jaringan. Ini dapat digunakan untuk mengidentifikasi titik di sepanjang jalan di mana masalah dapat ditemukan.

Sintaks perintah jejak bervariasi antara sistem operasi, seperti yang diilustrasikan pada gambar.

Perintah Jejak Windows dan Cisco iOS



Berikut ini adalah contoh output perintah **tracert** pada host Windows 10.

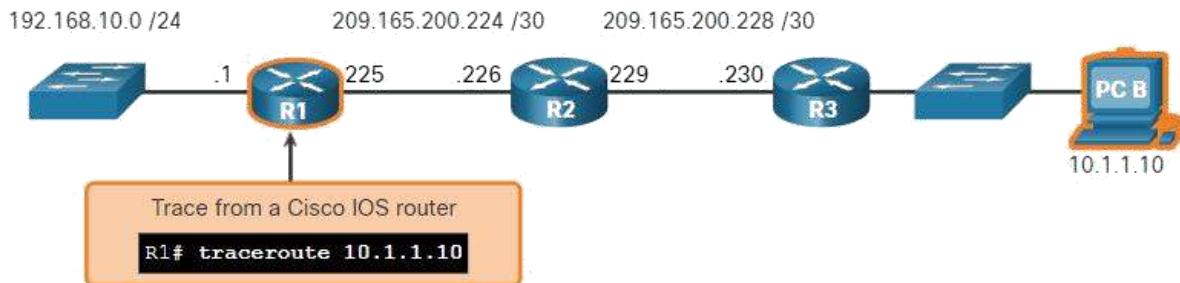
```
C:\Users\PC-A> tracert 10.1.1.10
Tracing route to 10.1.1.10 over a maximum of 30 hops:
  1  2 ms    2 ms    2 ms  192.168.10.1
  2  *        *        *      Request timed out.
  3  *        *        *      Request timed out.
  4  *        *        *      Request timed out.

^C
C:\Users\PC-A>
```

Catatan: Gunakan Ctrl-C untuk mengganggu **jejak** di Windows.

Satu-satunya respons yang berhasil adalah dari gateway di R1. Trace permintaan ke hop berikutnya yang waktunya seperti yang ditunjukkan oleh tanda bintang (*), yang berarti bahwa router hop berikutnya tidak merespons. Permintaan yang sudah waktunya menunjukkan bahwa ada kegagalan dalam pekerjaan internet di luar LAN, atau bahwa router ini telah dikonfigurasi untuk tidak menanggapi permintaan echo yang digunakan dalam jejak. Dalam contoh ini tampaknya ada masalah antara R1 dan R2.

Output perintah **traceroute** Cisco iOS bervariasi dari perintah **tracert** Windows. Misalnya, lihat topologi berikut.



Berikut ini adalah output sampel perintah traceroute dari R1.

```
R1# traceroute 10.1.1.10
Type escape sequence to abort.
Tracing the route to 10.1.1.10
VRF info: (vrf in name/id, vrf out name/id)
 1 209.165.200.226 1 msec 0 msec 1 msec
 2 209.165.200.230 1 msec 0 msec 1 msec
 3 10.1.1.10 1 msec 0 msec
R1#
```

Dalam contoh ini, jejak divalidasi bahwa ia dapat berhasil mencapai PC B.

Timeouts menunjukkan masalah potensial. Misalnya, jika host 10.1.1.10 tidak tersedia, perintah traceroute akan menampilkan output berikut.

```
R1# traceroute 10.1.1.10
Type escape sequence to abort.
Tracing the route to 10.1.1.10
VRF info: (vrf in name/id, vrf out name/id)
 1 209.165.200.226 1 msec 0 msec 1 msec
 2 209.165.200.230 1 msec 0 msec 1 msec
 3 * * *
 4 * * *
 5 *
```

Gunakan **Ctrl-Shift-6** untuk mengganggu traceroute di Cisco iOS.

Catatan : Implementasi Windows traceroute (tracert) mengirimkan Permintaan Echo ICMP. Cisco iOS dan Linux menggunakan UDP dengan nomor port yang tidak valid. Destination akhir akan mengembalikan pesan port ICMP yang tidak terjangkau.

Extended Traceroute

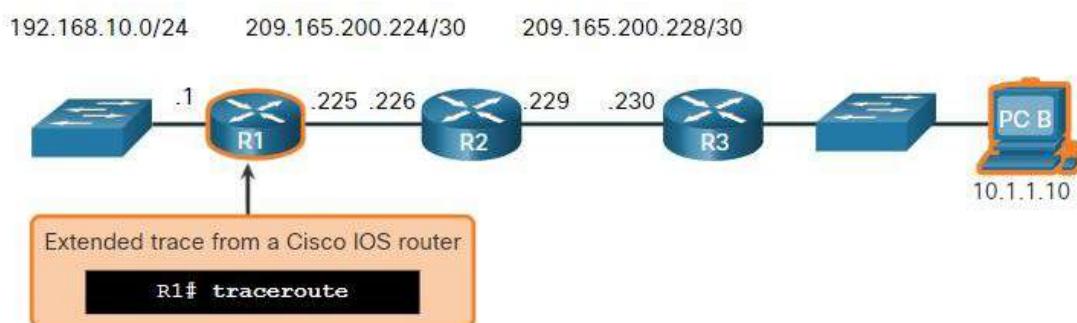
Seperti perintah **ping extended**, ada juga perintah **traceroute extended**. Hal ini memungkinkan administrator untuk menyesuaikan parameter yang terkait dengan operasi perintah. Ini sangat membantu dalam menemukan masalah ketika memecahkan masalah routing loop, menentukan router next-hop yang tepat, atau menentukan di mana paket dijatuhkan atau ditolak oleh router atau firewall.

Perintah **tracert** Windows memungkinkan input beberapa parameter melalui opsi di baris perintah. Namun, itu tidak dipandu seperti perintah iOS traceroute extended. Output berikut menampilkan opsi yang tersedia untuk perintah **tracert** Windows.

Opsi **traceroute** Cisco IOS extended memungkinkan pengguna untuk membuat jenis khusus jejak dengan menyesuaikan parameter yang terkait dengan operasi perintah. Extended traceroute dimasukkan dalam Privileged EXEC mode dengan mengetik **traceroute** tanpa alamat IP destination. IOS akan memandu Anda melalui opsi perintah dengan menyajikan sejumlah petunjuk yang terkait dengan pengaturan semua parameter yang berbeda.

Catatan: Menekan **Enter** menerima nilai default yang ditunjukkan.

Misalnya, asumsikan Anda ingin menguji konektivitas ke PC B dari LAN R1. Meskipun ini dapat diverifikasi dari PC A, **traceroute** extended dapat dikonfigurasi pada R1 untuk menentukan alamat source yang berbeda.



Seperti yang diilustrasikan dalam contoh, alamat IP source dari perintah **traceroute** extended pada R1 dapat dikonfigurasi untuk menggunakan alamat IP interface LAN R1 (yaitu, 192.168.10.1).

```
R1# traceroute
Protocol [ip]:
Target IP address: 10.1.1.10
Ingress traceroute [n]:
Source address: 192.168.10.1
DSCP Value [0]:
Numeric display [n]:
Timeout in seconds [3]:
Probe count [3]:
Minimum Time to Live [1]:
Maximum Time to Live [30]:
Port Number [33434]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Type escape sequence to abort.
Tracing the route to 192.168.10.10
VRF info: (vrf in name/id, vrf out name/id)
 1 209.165.200.226 1 msec 1 msec 1 msec
 2 209.165.200.230 0 msec 1 msec 0 msec
 3  *
      10.1.1.10 2 msec 2 msec
R1#
```

Baseline Jaringan

Salah satu alat yang paling efektif untuk memantau dan memecahkan masalah kinerja jaringan adalah untuk membangun baseline jaringan. Membuat baseline kinerja jaringan yang efektif dicapai selama periode waktu tertentu. Mengukur kinerja pada waktu dan beban yang berbeda-beda akan membantu menciptakan gambaran yang lebih baik tentang kinerja jaringan secara keseluruhan.

Output yang berasal dari perintah jaringan memberikan kontribusi data ke dasar jaringan. Salah satu metode untuk memulai baseline adalah menyalin dan menempelkan hasil dari **ping**, **Trace**, atau perintah relevan lainnya ke dalam file teks. File teks ini dapat dicap waktu dengan tanggal dan disimpan ke dalam arsip untuk pengambilan dan perbandingan nanti.

Di antara item yang perlu dipertimbangkan adalah error message dan waktu respons dari host ke host. Jika ada peningkatan yang cukup besar dalam waktu respons, mungkin ada masalah latensi untuk diatasi.

Misalnya, output **ping** berikut ditangkap dan ditempelkan ke dalam file teks.

Agustus 19, 2019 di 08:14:43

```
C:\Users\PC-A> ping 10.1.1.10
Pinging 10.1.1.10 with 32 bytes of data:
Reply from 10.1.1.10: bytes=32 time<1ms TTL=64
Ping statistics for 10.1.1.10:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
C:\Users\PC-A>
```

Perhatikan waktu **ping** round-trip kurang dari 1 ms.

Sebulan kemudian, ping diulang dan ditangkap.

September 19, 2019 di 10:18:21

```
C:\Users\PC-A> ping 10.1.1.10
Pinging 10.1.1.10 with 32 bytes of data:
Reply from 10.1.1.10: bytes=32 time=50ms TTL=64
Reply from 10.1.1.10: bytes=32 time=49ms TTL=64
Reply from 10.1.1.10: bytes=32 time=46ms TTL=64
Reply from 10.1.1.10: bytes=32 time=47ms TTL=64
Ping statistics for 10.1.1.10:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 46ms, Maximum = 50ms, Average = 48ms
C:\Users\PC-A>
```

Perhatikan kali ini bahwa **waktu ping** round-trip jauh lebih lama menunjukkan masalah potensial.

Jaringan perusahaan harus memiliki baseline yang luas; lebih luas dari yang bisa kita jelaskan dalam materi ini. Alat perangkat lunak kelas profesional tersedia untuk menyimpan dan memelihara informasi dasar. Dalam kursus ini, kami membahas beberapa teknik dasar dan mendiskusikan destination baseline.

Praktik terbaik Cisco untuk proses dasar dapat ditemukan dengan mencari di internet untuk “Baseline Process Best Practices”.

Perintah Host dan iOS

Jika Anda telah menggunakan salah satu alat dalam materi sebelumnya untuk memverifikasi konektivitas dan menemukan bahwa beberapa bagian dari jaringan Anda tidak berfungsi sebagaimana mestinya, sekarang adalah waktu untuk menggunakan beberapa perintah untuk memecahkan masalah perangkat Anda. Perintah Host dan iOS dapat membantu Anda menentukan apakah masalahnya adalah dengan pengalaman IP perangkat Anda, yang merupakan masalah jaringan umum.

Konfigurasi IP pada Host Windows

Memeriksa pengalaman IP pada perangkat host adalah praktik umum dalam jaringan untuk memverifikasi dan memecahkan masalah konektivitas end-to-end. Di Windows 10, Anda dapat mengakses detail alamat IP dari **Network and Sharing Center**, seperti yang ditunjukkan pada gambar, untuk dengan cepat melihat empat pengaturan penting: alamat, **mask**, router, dan DNS.



Namun, administrator jaringan biasanya melihat IP menangani informasi pada host Windows dengan mengeluarkan perintah **ipconfig** di baris perintah komputer Windows, seperti yang ditunjukkan dalam output sampel.

```
C:\Users\PC-A> ipconfig
Windows IP Configuration
(Output omitted)
Wireless LAN adapter Wi-Fi:
  Connection-specific DNS Suffix . :
  Link-local IPv6 Address . . . . . : fe80::a4aa:2dd1:ae2d:a75e%16
  IPv4 Address. . . . . : 192.168.10.10
  Subnet Mask . . . . . : 255.255.255.0
  Default Gateway . . . . . : 192.168.10.1
(Output omitted)
```

Gunakan perintah **ipconfig /all** untuk melihat alamat MAC, serta sejumlah detail mengenai pengalaman Layer 3 perangkat, seperti yang ditunjukkan pada output contoh.

```
C:\Users\PC-A> ipconfig /all
Windows IP Configuration
  Host Name . . . . . : PC-A-00H20
  Primary Dns Suffix . . . . . : cisco.com
  Node Type . . . . . : Hybrid
  IP Routing Enabled. . . . . : No
  WINS Proxy Enabled. . . . . : No
  DNS Suffix Search List. . . . . : cisco.com
(Output omitted)
Wireless LAN adapter Wi-Fi:
  Connection-specific DNS Suffix . :
  Description . . . . . : Intel(R) Dual Band Wireless-AC 8265
  Physical Address. . . . . : F8-94-C2-E4-C5-0A
  DHCP Enabled. . . . . : Yes
  Autoconfiguration Enabled . . . . . : Yes
  Link-local IPv6 Address . . . . . : fe80::a4aa:2dd1:ae2d:a75e%16(Preferred)
  IPv4 Address. . . . . : 192.168.10.10(Preferred)
  Subnet Mask . . . . . : 255.255.255.0
  Lease Obtained. . . . . : August 17, 2019 1:20:17 PM
  Lease Expires . . . . . : August 18, 2019 1:20:18 PM
  Default Gateway . . . . . : 192.168.10.1
  DHCP Server . . . . . : 192.168.10.1
  DHCPv6 IAID . . . . . : 100177090
  DHCPv6 Client DUID. . . . . : 00-01-00-01-21-F3-76-75-54-E1-AD-DE-DA-9A
  DNS Servers . . . . . : 192.168.10.1
  NetBIOS over Tcpip. . . . . : Enabled
```

Jika host dikonfigurasi sebagai klien DHCP, konfigurasi alamat IP dapat diperpanjang menggunakan perintah **ipconfig /release** dan **ipconfig /renew**, seperti yang ditunjukkan pada output sampel.

```
C:\Users\PC-A> ipconfig /release
(Output omitted)
Wireless LAN adapter Wi-Fi:
  Connection-specific DNS Suffix . :
    Link-local IPv6 Address . . . . . : fe80::a4aa:2dd1:ae2d:a75e%16
    Default Gateway . . . . . :
(Output omitted)
C:\Users\PC-A> ipconfig /renew
(Output omitted)
Wireless LAN adapter Wi-Fi:
  Connection-specific DNS Suffix . :
    Link-local IPv6 Address . . . . . : fe80::a4aa:2dd1:ae2d:a75e%16
    IPv4 Address. . . . . : 192.168.1.124
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.1.1
(Output omitted)
C:\Users\PC-A>
```

Layanan DNS Client pada PC Windows juga mengoptimalkan kinerja resolusi nama DNS dengan menyimpan nama yang telah diselesaikan sebelumnya dalam memori. Perintah **ipconfig /displaydns** menampilkan semua entri DNS cache pada sistem komputer Windows, seperti yang ditunjukkan pada output contoh.

```
C:\Users\PC-A> ipconfig /displaydns
Windows IP Configuration
(Output omitted)
netacad.com
-----
Record Name . . . . : netacad.com
Record Type . . . . : 1
Time To Live . . . . : 602
Data Length . . . . : 4
Section . . . . . : Answer
A (Host) Record . . . : 54.165.95.219
(Output omitted)
```

Konfigurasi IP pada Host Linux

Memverifikasi pengaturan IP menggunakan GUI pada mesin Linux akan berbeda tergantung pada distribusi Linux (distro) dan interface desktop. Gambar menunjukkan kotak dialog **Informasi Koneksi** pada distro Ubuntu yang menjalankan desktop Gnome.



Pada baris perintah, administrator jaringan menggunakan perintah **ifconfig** untuk menampilkan status **interface** yang saat ini aktif dan konfigurasi IP mereka, seperti yang ditunjukkan dalam output.

```
[analyst@secOps ~]$ ifconfig
enp0s3    Link encap:Ethernet HWaddr 08:00:27:b5:d6:cb
           inet addr: 10.0.2.15 Bcast:10.0.2.255 Mask: 255.255.255.0
             inet6 addr: fe80::57c6:ed95:b3c9:2951/64 Scope:Link
                  UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
                  RX packets:1332239 errors:0 dropped:0 overruns:0 frame:0
                  TX packets:105910 errors:0 dropped:0 overruns:0 carrier:0
                  collisions:0 txqueuelen:1000
                  RX bytes:1855455014 (1.8 GB) TX bytes:13140139 (13.1 MB)
lo        flags=73 mtu 65536
           inet 127.0.0.1 netmask 255.0.0.0
             inet6 ::1 prefixlen 128 scopeid 0x10
               loop txqueuelen 1000 (Local Loopback)
               RX packets 0 bytes 0 (0.0 B)
               RX errors 0 dropped 0 overruns 0 frame 0
               TX packets 0 bytes 0 (0.0 B)
               TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Perintah **alamat ip** Linux digunakan untuk menampilkan alamat dan propertinya. Hal ini juga dapat digunakan untuk menambah atau menghapus alamat IP.

Catatan: Output yang ditampilkan dapat bervariasi tergantung pada distribusi Linux.

Konfigurasi IP pada Host macOS

Di GUI host Mac, buka **Preferensi Jaringan** > **Lanjutan** untuk mendapatkan informasi pengalaman IP, seperti yang ditunjukkan pada gambar.



Namun, perintah **ifconfig** juga dapat digunakan untuk memverifikasi konfigurasi IP interface yang ditampilkan dalam output.

```
MacBook-Air:~ Admin$ ifconfig en0
en0: flags=8863 mtu 1500
    ether c4:b3:01:a0:64:98
    inet6 fe80::c0f:1bf4%en0 prefixlen 64 secured scopeid 0x5
        inet 10.10.10.113 netmask 0xffffffff broadcast 10.10.10.255
    nd6 options=201
    media: autoselect
    status: active
MacBook-Air:~ Admin$
```

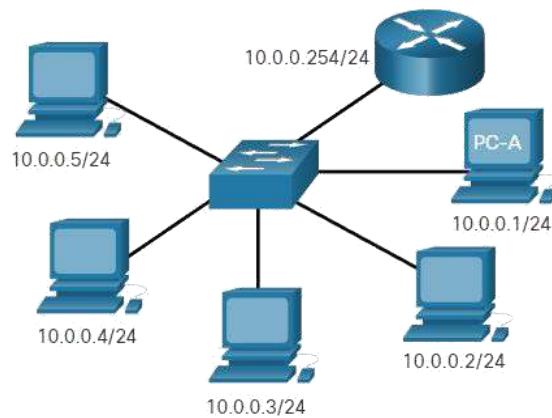
Perintah mac OS berguna lainnya untuk memverifikasi pengaturan IP host termasuk **networksetup -listallnetworkservices** dan **networksetup -getinfo <layanan jaringan>**, seperti yang ditunjukkan pada output berikut.

```
MacBook-Air:~ Admin$ networksetup -listallnetworkservices
An asterisk (*) denotes that a network service is disabled.
iPhone USB
Wi-Fi
Bluetooth PAN
Thunderbolt Bridge
MacBook-Air:~ Admin$
MacBook-Air:~ Admin$ networksetup -getinfo Wi-Fi
DHCP Configuration
IP address: 10.10.10.113
Subnet mask: 255.255.255.0
Router: 10.10.10.1
Client ID:
IPv6: Automatic
IPv6 IP address: none
IPv6 Router: none
Wi-Fi ID: c4:b3:01:a0:64:98
MacBook-Air:~ Admin$
```

Perintah arp

Perintah **arp** dijalankan dari command prompt Windows, Linux, atau Mac. Perintah tersebut mencantumkan semua perangkat yang saat ini ada dalam cache ARP host, yang mencakup alamat IPv4, alamat fisik, dan jenis pengalamatan (statis / dinamis), untuk setiap perangkat.

Misalnya, lihat topologi dalam gambar.



Output dari **arp -a** command pada Windows PC-A host ditampilkan.

```
C:\Users\PC-A> arp -a
Interface: 192.168.93.175 --- 0xc
      Internet Address      Physical Address      Type
        10.0.0.2                d0-67-e5-b6-56-4b    dynamic
        10.0.0.3                78-48-59-e3-b4-01    dynamic
        10.0.0.4                00-21-b6-00-16-97    dynamic
        10.0.0.254              00-15-99-cd-38-d9    dynamic
```

Perintah **arp -a** menampilkan alamat IP yang diketahui dan pengikatan alamat MAC. Perhatikan bagaimana alamat IP 10.0.0.5 tidak termasuk dalam daftar. Ini karena cache ARP hanya menampilkan informasi dari perangkat yang baru-baru ini diakses.

Untuk memastikan bahwa cache ARP diisi, **ping** perangkat sehingga akan memiliki entri dalam tabel ARP. Misalnya, jika PC-A melakukan ping 10.0.0.5, maka cache ARP akan berisi entri untuk alamat IP tersebut.

Cache dapat dibersihkan dengan menggunakan perintah **netsh interface ip delete arpcache** jika administrator jaringan ingin mengisi kembali cache dengan informasi yang diperbarui.

Catatan: Anda mungkin memerlukan akses administrator pada host untuk dapat menggunakan perintah **netsh interface ip delete arpcache**.

Perintah **show** umum Ditinjau Kembali

Dengan cara yang sama bahwa perintah dan utilitas digunakan untuk memverifikasi konfigurasi host, perintah dapat digunakan untuk memverifikasi **interface intermediary devices**. Cisco IOS menyediakan perintah untuk memverifikasi pengoperasian router dan switch interface.

Perintah **show** Cisco IOS CLI menampilkan informasi yang relevan tentang konfigurasi dan pengoperasian perangkat. Teknisi jaringan menggunakan perintah **show** secara ekstensif untuk melihat file konfigurasi, memeriksa status **interface** dan proses perangkat, dan memverifikasi status operasional perangkat. Status hampir setiap proses atau fungsi router dapat ditampilkan menggunakan perintah **show**.

Perintah **show** yang umum digunakan dan kapan menggunakannya tercantum dalam tabel.

| Perintah | Berguna untuk... |
|----------------------------|--|
| show running-config | Untuk memverifikasi konfigurasi dan pengaturan saat ini |
| show interface | Untuk memverifikasi status interface dan melihat apakah ada pesan kesalahan |
| show ip interface | Untuk memverifikasi informasi Layer 3 dari interface |
| show arp | Untuk memverifikasi daftar host yang dikenal di LAN Ethernet lokal |
| show ip route | Untuk memverifikasi informasi Routing Layer 3 |
| show protocols | Untuk memverifikasi protokol mana yang beroperasi |
| show version | Untuk memverifikasi memori, interface , dan lisensi perangkat |

show running-config

Memverifikasi konfigurasi dan pengaturan saat ini

```
R1# show running-config

(Output omitted)

!
version 15.5
service timestamps debug datetime msec
service timestamps log datetime msec
service password-encryption
!
hostname R1
!
interface GigabitEthernet0/0/0
description Link to R2
ip address 209.165.200.225 255.255.255.252
negotiation auto
!
interface GigabitEthernet0/0/1
description Link to LAN
ip address 192.168.10.1 255.255.255.0
negotiation auto
!
router ospf 10
network 192.168.10.0 0.0.0.255 area 0
network 209.165.200.224 0.0.0.3 area 0
!
banner motd ^C Authorized access only! ^C
!
line con 0
password 7 14141B180F0B
login
line vty 0 4
password 7 00071A150754
login
transport input telnet ssh
!
end
R1#
```

Show interface

Memverifikasi status interface dan menampilkan pesan kesalahan apa pun

```
R1# show interfaces
GigabitEthernet0/0/0 is up, line protocol is up
  Hardware is ISR4321-2x1GE, address is a0e0.af0d.e140 (bia a0e0.af0d.e140)
  Description: Link to R2
  Internet address is 209.165.200.225/30
  MTU 1500 bytes, BW 100000 Kbit/sec, DLY 100 usec,
    reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive not supported
  Full Duplex, 100Mbps, link type is auto, media type is RJ45
  output flow-control is off, input flow-control is off
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 00:00:01, output 00:00:21, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/375/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
    5127 packets input, 590285 bytes, 0 no buffer
    Received 29 broadcasts (0 IP multicasts)
    0 runts, 0 giants, 0 throttles
    0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
    0 watchdog, 5043 multicast, 0 pause input
    1150 packets output, 153999 bytes, 0 underruns
    0 output errors, 0 collisions, 2 interface resets
    0 unknown protocol drops
    0 babbles, 0 late collision, 0 deferred
    1 lost carrier, 0 no carrier, 0 pause output
    0 output buffer failures, 0 output buffers swapped out
GigabitEthernet0/0/1 is up, line protocol is up

(Output omitted)
```

Show ip Interface

Memverifikasi informasi Layer 3 dari interface

```
R1# show ip interface
GigabitEthernet0/0/0 is up, line protocol is up
  Internet address is 209.165.200.225/30
  Broadcast address is 255.255.255.255
  Address determined by setup command
  MTU is 1500 bytes
  Helper address is not set
  Directed broadcast forwarding is disabled
  Multicast reserved groups joined: 224.0.0.5 224.0.0.6
  Outgoing Common access list is not set
  Outgoing access list is not set
  Inbound Common access list is not set
  Inbound access list is not set
  Proxy ARP is enabled
  Local Proxy ARP is disabled
  Security level is default
  Split horizon is enabled
  ICMP redirects are always sent
  ICMP unreachable messages are always sent
  ICMP mask replies are never sent
  IP fast switching is enabled
  IP Flow switching is disabled
  IP CEF switching is enabled
  IP CEF switching turbo vector
  IP Null turbo vector
  Associated unicast routing topologies:
    Topology "base", operation state is UP
    IP multicast fast switching is enabled
    IP multicast distributed fast switching is disabled
    IP route-cache flags are Fast, CEF
    Router Discovery is disabled
    IP output packet accounting is disabled
    IP access violation accounting is disabled
    TCP/IP header compression is disabled
    RTP/IP header compression is disabled
    Probe proxy name replies are disabled
    Policy routing is disabled
    Network address translation is disabled
    BGP Policy Mapping is disabled
    Input features: NDI Check
    IPv4 MCCP Redirect outbound is disabled
    IPv4 MCCP Redirect inbound is disabled
    IPv4 MCCP Redirect exclude is disabled
GigabitEthernet0/0/1 is up, line protocol is up

(Output omitted)
```

show arp

Memverifikasi daftar host yang dikenal di LAN Ethernet lokal

```
R1# show arp
Protocol Address      Age (min)  Hardware Addr  Type    Interface
Internet 192.168.10.1      -   a0e0.af0d.e141  ARPA   GigabitEthernet0/0/1
Internet 192.168.10.10     95   c07b.bcc4.a9c0  ARPA   GigabitEthernet0/0/1
Internet 209.165.200.225    -   a0e0.af0d.e140  ARPA   GigabitEthernet0/0/0
Internet 209.165.200.226   138   a03d.6fe1.9d90  ARPA   GigabitEthernet0/0/0
R1#
```

show ip route

Memverifikasi informasi routing Layer 3

```
R1# show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
      a - application route
      + - replicated route, % - next hop override, p - overrides from PfR
Gateway of last resort is 209.165.200.226 to network 0.0.0.0
0*E2 0.0.0.0/0 [110/1] via 209.165.200.226, 02:19:50, GigabitEthernet0/0/0
      10.0.0.0/24 is subnetted, 1 subnets
O     10.1.1.0 [110/3] via 209.165.200.226, 02:05:42, GigabitEthernet0/0/0
      192.168.10.0/24 is variably subnetted, 2 subnets, 2 masks
C     192.168.10.0/24 is directly connected, GigabitEthernet0/0/1
L     192.168.10.1/32 is directly connected, GigabitEthernet0/0/1
      209.165.200.0/24 is variably subnetted, 3 subnets, 2 masks
C     209.165.200.224/30 is directly connected, GigabitEthernet0/0/0
L     209.165.200.225/32 is directly connected, GigabitEthernet0/0/0
O     209.165.200.228/30
      [110/2] via 209.165.200.226, 02:07:19, GigabitEthernet0/0/0
R1#
```

show protocols

Memverifikasi protokol mana yang beroperasi

```
R1# show protocols
Global values:
  Internet Protocol routing is enabled
  GigabitEthernet0/0/0 is up, line protocol is up
    Internet address is 209.165.200.225/30
  GigabitEthernet0/0/1 is up, line protocol is up
    Internet address is 192.168.10.1/24
  Serial0/1/0 is down, line protocol is down
  Serial0/1/1 is down, line protocol is down
  GigabitEthernet0 is administratively down, line protocol is down
R1#
```

show version

Memverifikasi memori, interface, dan lisensi perangkat

```
R1# show version
Cisco IOS XE Software, Version 03.16.08.5 - Extended Support Release
Cisco IOS Software, ISR Software (X86_64_LINUX_IOSD-UNIVERSALK9-M), Version 15.5(3)S8, RELEASE SOFTWARE
(fc2)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2018 by Cisco Systems, Inc.
Compiled Wed 08-Aug-18 10:48 by mcpre

(Output omitted)

ROM: IOS-XE ROMMON
R1 uptime is 2 hours, 25 minutes
Uptime for this control processor is 2 hours, 27 minutes
System returned to ROM by reload
System image file is "bootflash:/isr4300-universalk9.03.16.08.5.155-3.58-ext.SPA.bin"
Last reload reason: LocalSoft

(Output omitted)

Technology Package License Information:
-----
Technology      Technology-package          Technology-package
                 Current        Type            Next reboot
-----
appxk9         appxk9        RightToUse    appxk9
uck9           None          None          None
securityk9     securityk9     Permanent    securityk9
ipbasek9       ipbasek9     Permanent    ipbasek9
cisco ISR4321/K9 (1RU) processor with 1647778K/6147K bytes of memory.
Processor board ID FLM2044W0LT
2 Gigabit Ethernet interfaces
2 Serial interfaces
32768K bytes of non-volatile configuration memory.
4194304K bytes of physical memory.
3207167K bytes of flash memory at bootflash:.
978928K bytes of USB flash at usb0:.
Configuration register is 0x2102
R1#
```

Perintah show cdp neighbors

Ada beberapa perintah iOS lain yang berguna. Cisco Discovery Protocol (CDP) adalah protokol milik Cisco yang berjalan pada data link layer. Karena CDP beroperasi pada **data link layer**, dua atau lebih perangkat jaringan Cisco, seperti router yang mendukung protokol **network layer** yang berbeda, dapat belajar tentang satu sama lain bahkan jika koneksi Layer 3 belum ditetapkan.

Ketika perangkat Cisco boot, CDP dimulai secara default. CDP secara otomatis menemukan perangkat Neighbor Cisco yang menjalankan CDP, terlepas dari protokol atau suite Layer 3 mana yang berjalan. CDP bertukar informasi perangkat keras dan perangkat lunak dengan tetangga CDP yang terhubung langsung.

CDP memberikan informasi berikut tentang setiap perangkat CDP neighbor:

- **Pengenal perangkat** – Nama host yang dikonfigurasi dari switch, router, atau perangkat lain
- **Daftar alamat** – up to sampai satu alamat **network layer** untuk setiap protokol yang didukung
- **Port identifier** – Nama port lokal dan jarak jauh dalam bentuk string karakter ASCII, seperti FastEthernet 0/0
- **Daftar kemampuan** – Misalnya, apakah perangkat tertentu adalah switch Layer 2 atau switch Layer 3
- **Platform** – Platform perangkat keras perangkat – misalnya, router seri Cisco 1841.

Lihat topologi dan output perintah **show cdp neighbor**.



```
R3# show cdp neighbors
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone,
                  D - Remote, C - CVTA, M - Two-port Mac Relay
Device ID      Local Intrfce     Holdtme   Capability  Platform  Port ID
S3            Gig 0/0/1        122        S I       WS-C2960+  Fas 0/5
Total cdp entries displayed : 1
R3#
```

Output menampilkan bahwa **interface** R3 GigabitEthernet 0/0/1 terhubung ke **interface** FastEthernet 0/5 dari S3, yang merupakan switch Cisco Catalyst 2960+. Perhatikan bahwa R3 belum mengumpulkan informasi tentang S4. Ini karena CDP hanya dapat menemukan perangkat Cisco yang terhubung langsung. S4 tidak terhubung langsung ke R3 dan karena itu tidak tercantum dalam output.

Perintah **show cdp neighbors detail** mengungkapkan alamat IP perangkat tetangga, seperti yang ditunjukkan dalam output. CDP akan mengungkapkan alamat IP tetangga terlepas dari apakah Anda dapat melakukan ping tetangga itu atau tidak. Perintah ini sangat membantu ketika dua router Cisco tidak dapat merutekan **data link** bersama mereka. Perintah **show cdp neighbors detail** membantu menentukan apakah salah satu tetangga CDP memiliki kesalahan konfigurasi IP.

Sama membantunya dengan CDP, itu juga bisa menjadi risiko keamanan karena dapat memberikan informasi infrastruktur jaringan yang berguna bagi **Threat Actor**. Misalnya, secara default banyak versi iOS mengirim **CDP Advertisement** keluar semua port yang diaktifkan. Namun, praktik terbaik menunjukkan bahwa CDP harus diaktifkan hanya pada **interface** yang terhubung ke perangkat Cisco infrastruktur lainnya. **CDP Advertisement** harus dinonaktifkan pada port yang menghadap pengguna.

Karena beberapa versi iOS mengirimkan **CDP Advertisement** secara default, penting untuk mengetahui cara menonaktifkan CDP. Untuk menonaktifkan CDP secara global, gunakan perintah konfigurasi global **no cdp run**. Untuk menonaktifkan CDP pada **interface**, gunakan perintah **no cdp enable**.

Perintah **show ip interface brief**

Salah satu perintah yang paling sering digunakan adalah perintah **show ip interface brief**. Perintah ini memberikan output yang lebih singkat daripada perintah **show ip interface**. Ini memberikan ringkasan informasi kunci untuk semua **interface** jaringan pada router.

Misalnya, **output show ip interface brief** menampilkan semua **interface** pada router, alamat IP yang ditugaskan untuk setiap **interface**, jika ada, dan status operasional **interface**.

| Interface | IP-Address | OK? | Method | Status | Protocol |
|----------------------|-----------------|-----|--------|-----------------------|----------|
| GigabitEthernet0/0/0 | 209.165.200.225 | YES | manual | up | |
| GigabitEthernet0/0/1 | 192.168.10.1 | YES | manual | up | |
| Serial0/1/0 | unassigned | NO | unset | down | down |
| Serial0/1/1 | unassigned | NO | unset | down | down |
| GigabitEthernet0 | unassigned | YES | unset | administratively down | down |

Verifikasi Interface Switch

Perintah **show ip interface brief** juga dapat digunakan untuk memverifikasi status **interface** switch, seperti yang ditunjukkan pada output.

| Interface | IP-Address | OK? | Method | Status | Protocol |
|-----------------|-----------------|-----|--------|--------|----------|
| Vlan1 | 192.168.254.250 | YES | manual | up | up |
| FastEthernet0/1 | unassigned | YES | unset | down | down |
| FastEthernet0/2 | unassigned | YES | unset | up | up |
| FastEthernet0/3 | unassigned | YES | unset | up | up |

Interface VLAN1 diberi alamat IPv4 192.168.254.250, telah diaktifkan, dan beroperasi.

Output juga menunjukkan bahwa **interface** FastEthernet0/1 sedang down. Ini menunjukkan bahwa tidak ada perangkat yang terhubung ke **interface** atau perangkat yang terhubung memiliki **interface** jaringan yang tidak beroperasi.

Sebaliknya, output menunjukkan bahwa **interface** FastEthernet0/2 dan FastEthernet0/3 beroperasi. Hal ini ditunjukkan oleh status dan protokol yang ditampilkan sebagai up.

Metodologi Troubleshooting

Dalam dua materi sebelumnya, Anda belajar tentang beberapa utilitas dan perintah yang dapat Anda gunakan untuk membantu mengidentifikasi area masalah di jaringan Anda. Ini adalah bagian penting dari **troubleshooting**. Ada banyak cara untuk memecahkan masalah jaringan. Topik ini merinci proses **troubleshooting** terstruktur yang dapat membantu Anda menjadi administrator jaringan yang lebih baik. Ini juga memberikan beberapa perintah lagi untuk membantu Anda menyelesaikan masalah. Masalah jaringan bisa sederhana atau kompleks, dan dapat dihasilkan dari kombinasi masalah perangkat keras, perangkat lunak, dan koneksi. Teknisi harus dapat menganalisis masalah dan menentukan penyebab kesalahan sebelum mereka dapat menyelesaikan masalah jaringan. Proses ini disebut troubleshooting.

Pendekatan Troubleshooting Dasar

Metodologi **troubleshooting** yang umum dan efisien didasarkan pada metode ilmiah.

| Langkah | Deskripsi |
|--|---|
| Langkah 1. Mengidentifikasi Masalah | Ini adalah langkah pertama dalam proses troubleshooting. Meskipun alat dapat digunakan dalam langkah ini, percakapan dengan pengguna seringkali sangat membantu. |
| Langkah 2. Menetapkan Teori Kemungkinan Penyebab | Setelah masalah diidentifikasi, cobalah untuk menetapkan teori kemungkinan penyebab. Langkah ini sering menghasilkan lebih dari beberapa kemungkinan penyebab masalah. |
| Langkah 3. Menguji Teori untuk Menentukan Penyebab | Berdasarkan kemungkinan penyebabnya, uji teori Anda untuk menentukan mana yang menjadi penyebab masalah. Seorang teknisi akan sering menerapkan prosedur cepat untuk menguji dan melihat apakah itu memecahkan masalah. Jika prosedur cepat tidak memperbaiki masalah, Anda mungkin perlu meneliti masalah lebih lanjut untuk menentukan penyebab pastinya. |
| Langkah 4. Menetapkan Rencana Aksi dan Menerapkan Solusi | Setelah Anda menentukan penyebab pasti dari masalah, buat rencana tindakan untuk menyelesaikan masalah dan menerapkan solusinya. |

| | |
|---|---|
| Langkah 5. Verifikasi Solusi dan Terapkan Tindakan Pencegahan | Setelah Anda memperbaiki masalah, verifikasi fungsionalitas penuh. Jika berlaku, terapkan tindakan pencegahan. |
| Langkah 6. Temuan Dokumen, Tindakan, dan Hasil | Pada langkah terakhir dari proses troubleshooting, dokumentasikan temuan, tindakan, dan hasil Anda. Ini sangat penting untuk referensi di masa depan. |

Untuk menilai masalah, tentukan berapa banyak perangkat di jaringan yang mengalami masalah. Jika ada masalah dengan satu perangkat di jaringan, mulailah proses **troubleshooting** di perangkat itu. Jika ada masalah dengan semua perangkat di jaringan, mulailah proses **troubleshooting** di perangkat tempat semua perangkat lain terhubung. Anda harus mengembangkan metode logis dan konsisten untuk mendiagnosis masalah jaringan dengan menghilangkan satu masalah pada satu waktu.

Mengatasi atau mengeskalasi?

Dalam beberapa situasi, mungkin tidak mungkin untuk menyelesaikan masalah dengan segera. Masalah harus dieskalasikan ketika membutuhkan keputusan manajer, beberapa keahlian khusus, atau tingkat akses jaringan yang tidak tersedia untuk teknisi **troubleshooting**.

Misalnya, setelah **troubleshooting**, teknisi menyimpulkan modul router harus diganti. Masalah ini harus dieskalasikan untuk persetujuan manajer. Manajer mungkin harus mengeskalasikan masalah lagi karena mungkin memerlukan persetujuan dari departemen keuangan sebelum modul baru dapat dibeli.

Kebijakan perusahaan harus dengan jelas menyatakan kapan dan bagaimana seorang teknisi harus mengeskalasi masalah.

Perintah debug

Proses OS, protokol, mekanisme dan peristiwa menghasilkan pesan untuk mengkomunikasikan status mereka. Pesan-pesan ini dapat memberikan informasi berharga saat memecahkan masalah atau memverifikasi operasi sistem. Perintah **debug** iOS memungkinkan administrator untuk menampilkan pesan-pesan ini secara real-time untuk analisis. Ini adalah alat yang sangat penting untuk memantau peristiwa pada perangkat Cisco iOS.

Semua perintah **debug** dimasukkan dalam **Privileged EXEC mode**. Cisco iOS memungkinkan untuk mempersempit output **debug** untuk memasukkan hanya fitur yang relevan atau subfeature. Hal ini penting karena output debugging diberi prioritas tinggi dalam proses CPU dan dapat membuat sistem tidak dapat digunakan. Untuk alasan ini, gunakan perintah **debug** hanya untuk memecahkan masalah tertentu.

Misalnya, untuk memantau status pesan ICMP di router Cisco, gunakan **debug ip icmp**, seperti yang ditunjukkan pada contohnya.

```
R1# debug ip icmp
ICMP packet debugging is on
R1#
R1# ping 10.1.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
R1#
*Aug 20 14:18:59.605: ICMP: echo reply rcvd, src 10.1.1.1, dst 209.165.200.225,topology BASE, dscp 0
topoid 0
*Aug 20 14:18:59.606: ICMP: echo reply rcvd, src 10.1.1.1, dst 209.165.200.225,topology BASE, dscp 0
topoid 0
*Aug 20 14:18:59.608: ICMP: echo reply rcvd, src 10.1.1.1, dst 209.165.200.225,topology BASE, dscp 0
topoid 0
*Aug 20 14:18:59.609: ICMP: echo reply rcvd, src 10.1.1.1, dst 209.165.200.225,topology BASE, dscp 0
topoid 0
*Aug 20 14:18:59.611: ICMP: echo reply rcvd, src 10.1.1.1, dst 209.165.200.225,topology BASE, dscp 0
topoid 0
R1#
```

Untuk membuat daftar deskripsi singkat tentang semua opsi perintah debugging, gunakan perintah **debug ?** dalam **Privileged EXEC mode** di baris perintah.

Untuk menonaktifkan fitur debugging tertentu, tambahkan kata kunci **no** kata kunci di depan perintah **debug**:

```
Router# no debug ip icmp
```

Atau, Anda dapat memasukkan bentuk **perintah undebug** dalam **Privileged EXEC mode**:

```
Router# undebug ip icmp
```

Untuk mematikan semua perintah debug aktif sekaligus, gunakan perintah **undebug all**:

```
Router# undebug all
```

Berhati-hatilah menggunakan beberapa perintah **debug**. Perintah seperti **debug all** dan **debug packet ip** menghasilkan sejumlah besar output dan dapat menggunakan sebagian besar **resource** sistem. Router bisa begitu sibuk menampilkan pesan **debug** sehingga tidak akan memiliki kekuatan pemrosesan yang cukup untuk melakukan fungsi jaringannya, atau bahkan mendengarkan perintah untuk mematikan debugging. Untuk alasan ini, menggunakan opsi perintah ini tidak dianjurkan dan harus dihindari.

Perintah Terminal monitor

Koneksi untuk memberikan akses ke **interface** baris perintah iOS dapat ditetapkan dalam dua cara berikut:

- **Locally** – Koneksi lokal (yaitu, koneksi konsol) memerlukan akses fisik ke router atau port console switch menggunakan kabel rollover.
- **Remote** – Koneksi jarak jauh memerlukan penggunaan Telnet atau SSH untuk membuat koneksi ke perangkat yang dikonfigurasi IP.

Pesan iOS tertentu secara otomatis ditampilkan pada koneksi konsol tetapi tidak pada koneksi jarak jauh. Misalnya, output **debug** ditampilkan secara default pada koneksi konsol. Namun, output **debug** tidak secara otomatis ditampilkan pada koneksi jarak jauh. Ini karena pesan **debug** adalah pesan log yang dicegah agar tidak ditampilkan pada baris vty.

Dalam output berikut misalnya, pengguna membuat koneksi jarak jauh menggunakan Telnet dari R2 ke R1. Pengguna kemudian mengeluarkan perintah **debug ip icmp**. Namun, perintah tersebut gagal menampilkan output **debug**.

```
R2# telnet 209.165.200.225
Trying 209.165.200.225 ... Open
Authorized access only!
User Access Verification
Password:
R1> enable
Password:
R1# debug ip icmp
ICMP packet debugging is on
R1# ping 10.1.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
R1#
! No debug output displayed>
```

Untuk menampilkan pesan log di terminal (konsol virtual), gunakan perintah EXEC khusus **terminal monitor**. Untuk menghentikan pesan log di terminal, gunakan **terminal no monitor** perintah EXEC istimewa.

Misalnya, perhatikan bagaimana perintah **terminal monitor** sekarang telah dimasukkan dan perintah **ping** menampilkan output **debug**.

```
R1# terminal monitor
R1# ping 10.1.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
R1#
*Aug 20 16:03:49.735: ICMP: echo reply rcvd, src 10.1.1.1, dst 209.165.200.225,topology BASE, dscp 0
topoid 0
**Aug 20 16:03:49.737: ICMP: echo reply rcvd, src 10.1.1.1, dst 209.165.200.225,topology BASE, dscp 0
topoid 0
**Aug 20 16:03:49.738: ICMP: echo reply rcvd, src 10.1.1.1, dst 209.165.200.225,topology BASE, dscp 0
topoid 0
**Aug 20 16:03:49.740: ICMP: echo reply rcvd, src 10.1.1.1, dst 209.165.200.225,topology BASE, dscp 0
topoid 0
**Aug 20 16:03:49.741: ICMP: echo reply rcvd, src 10.1.1.1, dst 209.165.200.225,topology BASE, dscp 0
topoid 0
R1# no debug ip icmp
ICMP packet debugging is off
R1#
```

Catatan: Maksud dari perintah **debug** adalah untuk menangkap output langsung untuk waktu yang singkat (yaitu, beberapa detik hingga satu menit atau lebih). Selalu **disable debug** bila tidak diperlukan.

Skenario Troubleshooting

Banyak masalah jaringan umum dapat diidentifikasi dan diselesaikan dengan sedikit usaha. Sekarang setelah Anda memiliki alat dan proses untuk memecahkan masalah jaringan, topik ini meninjau beberapa masalah jaringan umum yang mungkin Anda temukan sebagai administrator jaringan.

Masalah Operasi dupleks dan ketidakcocokan

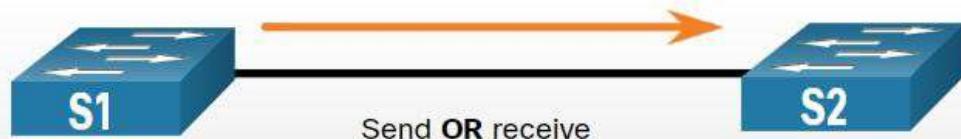
Dalam komunikasi data, *duplex* mengacu pada arah transmisi data antara dua perangkat.

Ada dua mode komunikasi dupleks:

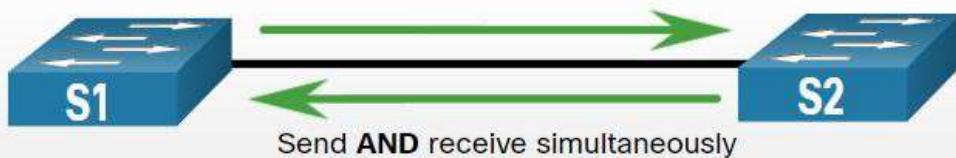
- **Half-duplex** – Komunikasi terbatas pada pertukaran data dalam satu arah pada satu waktu.
- **Full-duplex** – Komunikasi diizinkan untuk dikirim dan diterima secara bersamaan.

Gambar ini menggambarkan bagaimana setiap metode dupleks beroperasi.

Half-Duplex Communication

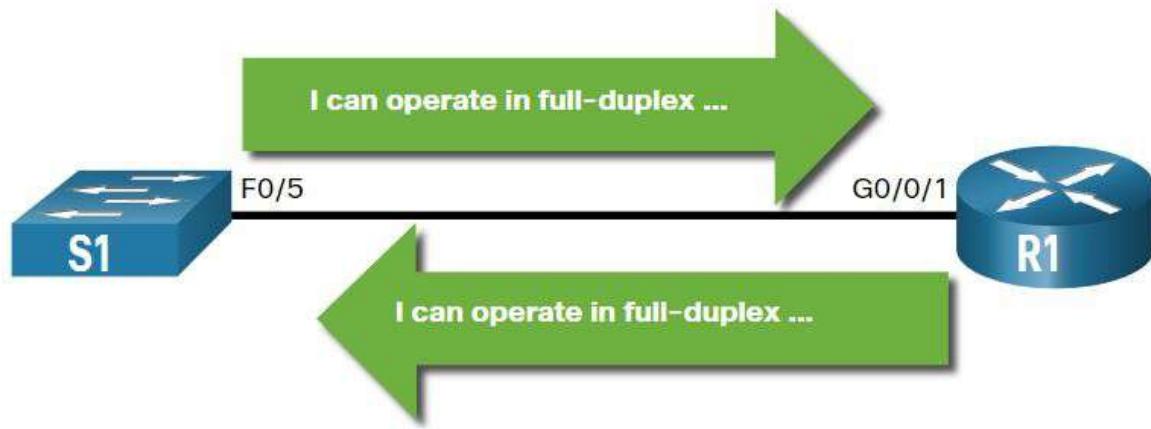


Full-Duplex Communication



Interface Ethernet interkoneksi harus beroperasi dalam mode dupleks yang sama untuk kinerja komunikasi terbaik dan untuk menghindari inefisiensi dan latensi pada link.

Fitur *auto negotiation* Ethernet memfasilitasi konfigurasi, meminimalkan masalah dan memaksimalkan kinerja **link** antara dua **link** Ethernet yang saling terhubung. Perangkat yang terhubung pertama kali mengumumkan kemampuan yang didukung mereka dan kemudian memilih mode kinerja tertinggi yang didukung oleh kedua ujungnya. Misalnya, switch dan router dalam gambar telah berhasil *auto negotiation* mode full-duplex.



Jika salah satu dari dua perangkat yang terhubung beroperasi dalam full-duplex dan yang lainnya beroperasi dalam half-duplex, ketidakcocokan dupleks terjadi. Sementara komunikasi data akan terjadi melalui **link** dengan ketidakcocokan dupleks, kinerja **link** akan sangat buruk.

Ketidakcocokan dupleks biasanya disebabkan oleh **interface** yang salah dikonfigurasi atau dalam kasus yang jarang terjadi oleh negosiasi otomatis yang gagal. Ketidakcocokan dupleks mungkin sulit untuk dipecahkan karena komunikasi antar perangkat masih terjadi.

IP Mengatasi Masalah pada Perangkat iOS

Masalah terkait alamat IP kemungkinan akan menjaga perangkat jaringan jarak jauh dari berkomunikasi. Karena alamat IP bersifat hierarkis, setiap alamat IP yang ditugaskan ke perangkat jaringan harus sesuai dengan rentang alamat dalam jaringan itu. Alamat IP yang salah ditugaskan menciptakan berbagai masalah, termasuk konflik alamat IP dan masalah routing.

Dua penyebab umum dari tugas IPv4 yang salah adalah kesalahan penugasan manual atau masalah terkait DHCP.

Administrator jaringan sering harus secara manual menetapkan alamat IP ke perangkat seperti server dan router. Jika kesalahan dibuat selama penugasan, maka masalah komunikasi dengan perangkat sangat mungkin terjadi.

Pada perangkat iOS, gunakan **show ip interface** atau **show ip interface brief** untuk memverifikasi alamat IPv4 apa yang ada ke **interface** jaringan. Misalnya, mengeluarkan perintah **show ip interface brief** seperti yang ditunjukkan akan memvalidasi status **interface** pada R1.

```
R1# show ip interface brief
Interface          IP-Address      OK? Method Status        Protocol
GigabitEthernet0/0/0 209.165.200.225 YES manual up           up
GigabitEthernet0/0/1 192.168.10.1   YES manual up           up
Serial0/1/0          unassigned     NO  unset  down         down
Serial0/1/1          unassigned     NO  unset  down         down
GigabitEthernet0      unassigned     YES unset administratively down down
R1#
```

IP Mengatasi Masalah pada End device

Di mesin berbasis Windows, ketika perangkat tidak dapat menghubungi server DHCP, Windows akan secara otomatis menetapkan alamat milik rentang 169.254.0.0/16. Fitur ini disebut Automatic Private IP Addressing (APIPA) dan dirancang untuk memfasilitasi komunikasi dalam jaringan lokal. Anggap saja sebagai Windows mengatakan, “Saya akan menggunakan alamat ini dari kisaran 169.254.0.0/16 karena saya tidak bisa mendapatkan alamat lain”.

Seringkali, komputer dengan alamat APIPA tidak akan dapat berkomunikasi dengan perangkat lain dalam jaringan karena perangkat tersebut kemungkinan besar tidak akan termasuk dalam jaringan 169.254.0.0/16. Situasi ini menunjukkan masalah penugasan alamat IPv4 otomatis yang harus diperbaiki.

Catatan: Sistem operasi lain, seperti Linux dan OS X, tidak akan menetapkan alamat IPv4 ke **interface** jaringan jika komunikasi dengan server DHCP gagal.

Sebagian besar **end device** dikonfigurasi untuk mengandalkan server DHCP untuk penugasan alamat IPv4 otomatis. Jika perangkat tidak dapat berkomunikasi dengan server DHCP, maka server tidak dapat menetapkan alamat IPv4 untuk jaringan tertentu dan perangkat tidak akan dapat berkomunikasi.

Untuk memverifikasi alamat IP yang ditetapkan ke komputer berbasis Windows, gunakan perintah **ipconfig**, seperti yang ditunjukkan pada output.

```
C:\Users\PC-A> ipconfig
Windows IP Configuration
(Output omitted)
Wireless LAN adapter Wi-Fi:
  Connection-specific DNS Suffix . :
  Link-local IPv6 Address . . . . . : fe80::a4aa:2dd1:ae2d:a75e%16
    IPv4 Address. . . . . : 192.168.10.10
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.10.1
(Output omitted)
```

Masalah Default gateway

Default gateway untuk **end device** adalah perangkat jaringan terdekat yang dapat meneruskan **traffic** ke jaringan lain. Jika perangkat memiliki alamat **default gateway** yang salah atau tidak ada, perangkat tidak akan dapat berkomunikasi dengan perangkat di jaringan jarak jauh. Karena **default gateway** adalah jalur ke jaringan jarak jauh, alamatnya harus termasuk dalam jaringan yang sama dengan **end device**.

Alamat **default gateway** dapat diatur secara manual atau diperoleh dari server DHCP. Mirip dengan IPv4 menangani masalah, masalah **default gateway** dapat dikaitkan dengan kesalahan konfigurasi (dalam kasus penugasan manual) atau masalah DHCP (jika tugas otomatis digunakan).

Untuk mengatasi masalah **default gateway** yang salah dikonfigurasi, pastikan perangkat memiliki **default gateway** yang benar dikonfigurasi. Jika alamat default diatur secara manual tetapi tidak benar, cukup ganti dengan alamat yang tepat. Jika alamat **default gateway** diatur secara otomatis, pastikan perangkat dapat berkomunikasi dengan server DHCP. Penting juga untuk memverifikasi bahwa alamat IPv4 dan subnet mask yang tepat dikonfigurasi pada **interface router** dan bahwa **interface aktif**.

Untuk memverifikasi **default gateway** pada komputer berbasis Windows, gunakan perintah **ipconfig** seperti yang ditunjukkan.

```
C:\Users\PC-A> ipconfig
Windows IP Configuration
(Output omitted)
Wireless LAN adapter Wi-Fi:
  Connection-specific DNS Suffix . :
  Link-local IPv6 Address . . . . . : fe80::a4aa:2dd1:ae2d:a75e%16
    IPv4 Address. . . . . : 192.168.10.10
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.10.1
(Output omitted)
```

Pada router, gunakan perintah **show ip route** untuk mencantumkan tabel perutean dan verifikasi bahwa **default gateway**, yang dikenal sebagai rute default, telah ditetapkan. Rute ini digunakan ketika alamat tujuan paket tidak cocok dengan rute lain dalam tabel routing-nya.

Misalnya, output memverifikasi bahwa R1 memiliki **default gateway** (yaitu Gateway of last resort) yang dikonfigurasi menunjuk ke alamat IP 209.168.200.226.

```
R1# show ip route | begin Gateway
Gateway of last resort is 209.165.200.226 to network 0.0.0.0
0*E2  0.0.0.0/0 [110/1] via 209.165.200.226, 02:19:50, GigabitEthernet0/0/0
    10.0.0.0/24 is subnetted, 1 subnets
    0      10.1.1.0 [110/3] via 209.165.200.226, 02:05:42, GigabitEthernet0/0/0
        192.168.10.0/24 is variably subnetted, 2 subnets, 2 masks
        C      192.168.10.0/24 is directly connected, GigabitEthernet0/0/1
        L      192.168.10.1/32 is directly connected, GigabitEthernet0/0/1
    209.165.200.0/24 is variably subnetted, 3 subnets, 2 masks
    C      209.165.200.224/30 is directly connected, GigabitEthernet0/0/0
    L      209.165.200.225/32 is directly connected, GigabitEthernet0/0/0
    0      209.165.200.228/30
          [110/2] via 209.165.200.226, 02:07:19, GigabitEthernet0/0/0
R1#
```

Baris pertama yang disorot pada dasarnya menyatakan bahwa gateway ke setiap (yaitu, 0.0.0.0) harus dikirim ke alamat IP 209.165.200.226. Yang kedua disorot menampilkan bagaimana R1 belajar tentang **default gateway**. Dalam hal ini, R1 menerima informasi dari router berkemampuan OSPF lainnya.

Pemecahan masalah DNS

Layanan Nama Domain (DNS) mendefinisikan layanan otomatis yang cocok dengan nama, seperti www.cisco.com, dengan alamat IP. Meskipun resolusi DNS tidak penting untuk komunikasi perangkat, sangat penting bagi **end user**.

Adalah umum bagi pengguna untuk secara keliru menghubungkan pengoperasian **link** internet dengan ketersediaan DNS. Keluhan pengguna seperti “jaringan sedang down” atau “internet sedang down” sering disebabkan oleh server DNS yang tidak terjangkau. Sementara packet routing dan semua layanan jaringan lainnya masih beroperasi, kegagalan DNS sering membawa pengguna ke kesimpulan yang salah. Jika pengguna mengetik dalam nama domain seperti www.cisco.com di browser web dan server DNS tidak dapat dijangkau, nama tidak akan diterjemahkan ke dalam alamat IP dan situs web tidak akan ditampilkan.

Alamat server DNS dapat ditetapkan secara manual atau otomatis. Administrator jaringan sering bertanggung jawab untuk menetapkan alamat server DNS secara manual di server dan perangkat lain, sementara DHCP digunakan untuk secara otomatis menetapkan alamat server DNS kepada klien.

Meskipun umum bagi perusahaan dan organisasi untuk mengelola server DNS mereka sendiri, setiap server DNS yang dapat dijangkau dapat digunakan untuk **meresolve nama**. Pengguna kantor kecil dan kantor rumah (SOHO) sering mengandalkan server DNS yang dikelola oleh ISP mereka untuk **name resolution**. Server DNS yang dikelola ISP ditugaskan untuk pelanggan SOHO melalui DHCP. Selain itu, Google memelihara server DNS publik yang dapat digunakan oleh siapa saja dan sangat berguna untuk pengujian. Alamat IPv4 dari server DNS public Google adalah 8.8.8.8 dan 2001:4860:4860::8888 untuk alamat DNS IPv6-nya.

Cisco menawarkan OpenDNS yang menyediakan layanan DNS aman dengan menyaring phishing dan beberapa situs malware. Anda dapat mengubah alamat DNS Anda ke 208.67.222.222 dan 208.67.220.220 di server DNS Pilihan dan bidang server DNS Alternatif. Fitur-fitur canggih seperti penyaringan konten web dan keamanan tersedia untuk keluarga dan bisnis.

Gunakan **ipconfig /all** seperti yang ditunjukkan untuk memverifikasi server DNS mana yang digunakan oleh komputer Windows.

```
C:\Users\PC-A> ipconfig /all
(Output omitted)
Wireless LAN adapter Wi-Fi:
  Connection-specific DNS Suffix . :
  Description . . . . . : Intel(R) Dual Band Wireless-AC 8265
  Physical Address. . . . . : F8-94-C2-E4-C5-0A
  DHCP Enabled. . . . . : Yes
  Autoconfiguration Enabled . . . . : Yes
  Link-local IPv6 Address . . . . : fe80::a4aa:2dd1:ae2d:a75e%16(PREFERRED)
  IPv4 Address. . . . . : 192.168.10.10(PREFERRED)
  Subnet Mask . . . . . : 255.255.255.0
  Lease Obtained. . . . . : August 17, 2019 1:20:17 PM
  Lease Expires . . . . . : August 18, 2019 1:20:18 PM
  Default Gateway . . . . . : 192.168.10.1
  DHCP Server . . . . . : 192.168.10.1
  DHCPv6 IAID . . . . . : 100177090
  DHCPv6 Client DUID. . . . . : 00-01-00-01-21-F3-76-75-54-E1-AD-DE-DA-9A
  DNS Servers . . . . . : 208.67.222.222
  NetBIOS over Tcpip. . . . . : Enabled
(Output omitted)
```

Perintah **nslookup** adalah alat pemecahan masalah DNS lain yang berguna untuk PC. Dengan **nslookup** pengguna dapat secara manual menempatkan permintaan DNS dan menganalisis respons DNS. Perintah **nslookup** menunjukkan output untuk kueri www.cisco.com. Perhatikan Anda juga bisa memasukkan alamat IP dan **nslookup** akan menyelesaikan namanya.

Catatan:Tidak selalu mungkin untuk mengetik alamat IP di **nslookup** dan menerima nama domain. Salah satu alasan paling umum untuk ini adalah bahwa sebagian besar situs web berjalan di server yang mendukung beberapa situs.

```
C:\Users\PC-A> nslookup
Default Server: Home-Net
Address: 192.168.1.1
> cisco.com
Server: Home-Net
Address: 192.168.1.1
Non-authoritative answer:
Name: cisco.com
Addresses: 2001:420:1101:1::185
           72.163.4.185
> 8.8.8.8
Server: Home-Net
Address: 192.168.1.1
Name: dns.google
Address: 8.8.8.8
>
> 208.67.222.222
Server: Home-Net
Address: 192.168.1.1
Name: resolver1.opendns.com
Address: 208.67.222.222
>
```

