

DNS Prinsip Kerja Beserta Contohnya

Pandu Arizona Sukono - 32609
Jurusan Teknik Elektro FT UGM,
Yogyakarta

I. PENDAHULUAN

Pada awal berkembang internet, seorang user yang akan menghubungi komputer user lain harus menyebutkan alamat IP address komputer yang hendak dituju. Seiring dengan berkembangnya teknologi internet dan semakin berlipatnya jumlah pengguna aplikasi internet, maka akanlah sangat menyusahkan jika kita harus mengingat IP address setiap user yang berupa angka-angka. Untuk mengatasi hal ini, dikembangkanlah sistem penterjemahan IP address yang berupa angka ke nama yang lebih mudah diingat maupun sebaliknya.

Pada awalnya sistem penamaan IP address menggunakan sistem host table. Dalam sistem ini, setiap komputer menyimpan file host.txt yang berisikan daftar kombinasi IP address dengan nama dari setiap komputer yang terhubung ke internet. Jadi setiap ada penambahan user ataupun perubahan data user, kita harus memperbaharui file host.txt melalui FTP ke seluruh host yang ada di internet. Tentunya sistem penamaan IP seperti ini sangatlah tidak efektif dan sudah tidak mampu menangani kebutuhan yang ada saat ini. Oleh karena itu pada tahun 1984, Paul Mockapertis mengusulkan sistem penamaan IP yang baru yaitu menggunakan Domain Name System (DNS), sistem inilah yang digunakan hingga saat ini.

II. AWAL MULA DOMAIN NAME SYSTEM (DNS)

A. Sejarah DNS

Sebelum dipergunakannya DNS, jaringan komputer menggunakan HOSTS files yang berisi informasi dari nama komputer dan IP address-nya. Di Internet, file ini dikelola secara terpusat dan di setiap lokasi harus di copy versi terbaru dari HOSTS files, dari sini bisa dibayangkan betapa repotnya jika ada penambahan 1 komputer di jaringan, maka kita harus copy versi terbaru file ini ke setiap lokasi. Dengan makin meluasnya jaringan internet, hal ini makin merepotkan, akhirnya dibuatkan sebuah solusi dimana DNS di desain menggantikan fungsi HOSTS files, dengan kelebihan unlimited database size, dan performace yang baik. DNS adalah sebuah aplikasi services di Internet yang menerjemahkan sebuah domain name ke IP address. Sebagai contoh, www untuk penggunaan di Internet, lalu diketikan nama domain, misalnya: yahoo.com maka akan di petakan ke sebuah IP mis 202.68.0.134. Jadi DNS dapat di analogikan pada pemakaian buku telepon, dimana orang yang kita kenal berdasarkan nama untuk menghubunginya kita harus memutar nomor telepon di pesawat telepon. Sama persis, host komputer mengirimkan queries berupa nama komputer dan domain name server ke DNS, lalu oleh DNS dipetakan ke IP address.

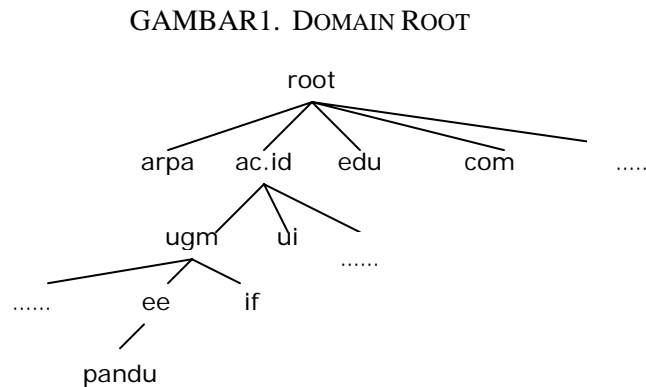
B. Domain Name System (DNS)

Domain Name System (DNS) adalah distribute database system yang digunakan untuk pencarian nama komputer (name resolution) di jaringan yang menggunakan TCP/IP (Transmission Control Protocol/Internet Protocol). DNS biasa digunakan pada aplikasi yang terhubung ke Internet seperti web browser atau e-mail, dimana DNS membantu memetakan host name sebuah komputer ke IP address. Selain digunakan di Internet, DNS juga dapat di implementasikan ke private network atau intranet dimana DNS memiliki keunggulan seperti:

- 1) Mudah, DNS sangat mudah karena user tidak lagi direpotkan untuk mengingat IP address sebuah komputer cukup host name (nama Komputer).
- 2) Konsisten, IP address sebuah komputer bisa berubah tapi host name tidak berubah.
- 3) Simple, user hanya menggunakan satu nama domain untuk mencari baik di Internet maupun di Intranet.

C. Prinsip Kerja Pendelegasian Domain

Sistem database yang digunakan DNS adalah sistem hirarki. Sistem ini dipilih karena sistem ini cocok digunakan untuk sistem terdistribusi, konsisten untuk setiap host, dan *updateable*. Sistem ini digambarkan sebagai sebuah *tree* yang memiliki beberapa cabang. Cabang-cabang ini mewakili domain, dan dapat berupa host, subdomain, ataupun top level domain, yang digambarkan sebagai berikut:



a. Gambar contoh Domain Root

Domain teratas adalah root. Domain ini diwakili oleh titik. Selanjutnya, domain yang terletak tepat di bawah root disebut top level domain. Beberapa contoh top level domain ini antara lain com, edu, gov, dan lain-lain. Turunan dari top level domain disebut subdomain. Domain yang terletak setelah top level domain adalah second level domain, dan domain yang berada di bawah second level domain disebut third level domain, begitu seterusnya.

Pembentukan dan pembacaannya dimulai dari node yang paling bawah berurut dari node yang paling bawah hingga node yang paling atas (root). Sebagai contoh untuk nama host **pandu.ee.ugm.ac.id**, kita dapat menguraikan domainnya menjadi :

- . merupakan root domain,
- id merupakan top level domain,
- ac merupakan second level domain,
- ugm merupakan third level domain,
- ee merupakan fourth level domain,
- pandu merupakan fifth level domain yang menunjukkan nama host.

Dari contoh di atas kita dapat mengetahui aturan penulisan nama host, yaitu dimulai dari kiri ke kanan untuk node yang paling bawah hingga node yang paling atas, dan setiap domain dipisahkan dengan titik. Seperti yang telah disampaikan sebelumnya bahwa selain berfungsi untuk *mapping* alamat IP ke nama host, DNS juga berfungsi sebaliknya, yaitu *reverse mapping* dari nama host ke alamat IP. Masalah aturan *mapping* dan *reverse mapping* ini akan dibahas pada bagian konfigurasi zone.

Sistem hirarki DNS di atas tentunya tidak mungkin di atur oleh satu server saja. Sistem hirarki DNS ini dipecah-pecah menjadi zona-zona. Sebuah zona meliputi seluruh host yang terdapat di dalam sebuah domain dan dapat berupa level domain yang mana saja.

Di dalam sebuah jaringan, setiap zona harus memiliki name server sendiri. Name server inilah yang akan melayani penerjemahan dari IP ke nama host atau sebaliknya di zona yang bersangkutan. Jika pada suatu domain ingin dibentuk zona baru, maka kita dapat membuat sebuah primary name server yang akan mengarahkan setiap request mapping ke name server yang authoritative. Penggunaan lebih dari satu name server ini bertujuan untuk membagi beban kerja name server dan sebagai sistem backup name server.

Perbedaan antara primary name server dengan secondary name server terletak dari bagaimana name server itu memperoleh datanya. Primary name server memperoleh data dari data yang tersimpan di harddisknya, sedangkan secondary name server memperoleh data dari data replika yang ia peroleh dari primary server name. Dengan demikian, untuk setiap perubahan data dalam DNS, kita cukup mengubah data pada primary name server-nya saja.

D. Sistem Hirarkis DNS

Sistem hirarki DNS dari hal diatas tentunya tidak mungkin di atur oleh satu server saja. Sistem hirarki DNS ini dipecah-pecah menjadi zona-zona. Sebuah zona meliputi seluruh host yang terdapat di dalam sebuah domain dan dapat berupa level domain yang mana saja. Di dalam sebuah jaringan, setiap zona harus memiliki name server sendiri. Name server inilah yang akan melayani penerjemahan dari IP ke nama host atau sebaliknya di zona yang bersangkutan. Jika pada suatu domain ingin dibentuk zona baru, maka kita dapat membuat sebuah primary name server yang akan mengarahkan setiap request mapping ke name server yang authoritative. Penggunaan lebih dari satu name server ini bertujuan untuk membagi beban kerja name server dan sebagai sistem backup name server.

Perbedaan antara primary name server dengan secondary name server terletak dari bagaimana name server itu memperoleh datanya. Primary name server memperoleh data dari data yang tersimpan di harddisknya, sedangkan secondary name server memperoleh data dari data replika yang ia peroleh dari primary server name. Dengan demikian, untuk setiap perubahan data dalam DNS, kita cukup mengubah data pada primary name server-nya saja.

III. STRUKTUR DAN PRINSIP KERJA DNS

A. Struktur DNS

Domain Name Space merupakan sebuah hirarki pengelompokan domain berdasarkan nama, yang terbagi menjadi beberapa bagian diantaranya:

Sebuah nama domain biasanya terdiri dari dua bagian atau lebih (secara teknis disebut label), dipisahkan dengan titik.

Label paling kanan menyatakan top-level domain - domain tingkat atas/tinggi (misalkan, alamat www.bestcyber.blogspot.com memiliki top-level domain com).

Setiap label di sebelah kirinya menyatakan sebuah sub-divisi atau subdomain dari domain yang lebih tinggi. Catatan: "subdomain" menyatakan ketergantungan relatif, bukan absolut. Contoh: blogspot.com merupakan subdomain dari domain com, dan bestcyber.blogspot.com dapat membentuk subdomain dari domain blogspot.com (pada prakteknya, bestcyber.blogspot.com sesungguhnya mewakili sebuah nama host - lihat dibawah). Secara teori, pembagian seperti ini dapat mencapai kedalaman 127 level, dan setiap label dapat terbentuk sampai dengan 63 karakter, selama total nama domain tidak melebihi panjang 255 karakter. Tetapi secara praktek, beberapa pendaftar nama domain (domain name registry) memiliki batas yang lebih sedikit.

Terakhir, bagian paling kiri dari bagian nama domain (biasanya) menyatakan nama host. Sisa dari nama domain menyatakan cara untuk membangun jalur logis untuk informasi yang dibutuhkan; nama host adalah tujuan sebenarnya dari nama sistem yang dicari alamat IP-nya. Contoh: nama domain `www.bestcyber.blogspot.com` memiliki nama host "www".

DNS memiliki kumpulan hirarki dari DNS servers. Setiap domain atau subdomain memiliki satu atau lebih authoritative DNS Servers (server DNS otoritatif) yang mempublikasikan informasi tentang domain tersebut dan nama-nama server dari setiap domain di-"bawah"-nya. Pada puncak hirarki, terdapat root servers- induk server nama: server yang ditanyakan ketika mencari (menyelesaikan/resolving) dari sebuah nama domain tertinggi (top-level domain)..

B. Root-Level Domains

Domain ditentukan berdasarkan tingkatan kemampuan yang ada di struktur hirarki yang disebut dengan level. Level paling atas di hirarki disebut dengan root domain. Root domain di ekspresikan berdasarkan periode dimana lambang untuk root domain adalah (".").

C. Top Level Domain

Selain penjelasan diatas, pada bagian dibawah ini adalah contoh dari top-level domains:

f com Organisasi Komersial

f edu Institusi pendidikan atau universitas

f org Organisasi non-profit

f net Networks (backbone Internet)

f gov Organisasi pemerintah non militer

f mil Organisasi pemerintah militer

f num No telpon

f arpa Reverse DNS

f xx dua-huruf untuk kode negara (id:Indonesia, sg:singapura, au:australia,dll)

Top-level domains dapat berisi second-level domains dan hosts.

D. Second-Level Domains

Second-level domains dapat berisi host dan domain lain, yang disebut dengan subdomain. Untuk contoh: Domain Bujangan, `bujangan.com` terdapat komputer (host) seperti `server1.bujangan.com` dan subdomain `training.bujangan.com`. Subdomain `training.bujangan.com` juga terdapat komputer (host) seperti `client1.training.bujangan.com`.

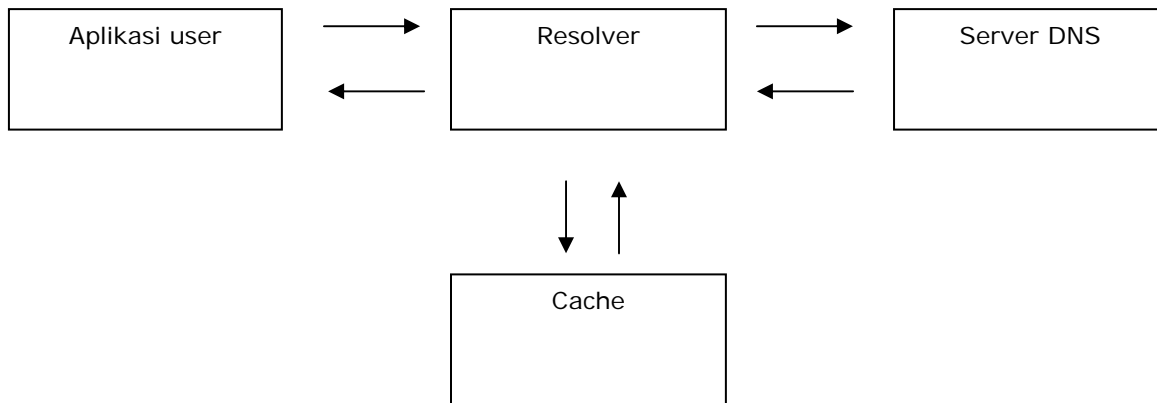
E. Host Names

Domain name yang digunakan dengan host name akan menciptakan fully qualified domain name (FQDN) untuk setiap komputer. Sebagai contoh, jika terdapat `filesserver1.detik.com`, dimana `filesserver1` adalah host name dan `detik.com` adalah domain name.

F. Struktur Kerja DNS

Model kerja server DNS dapat digambarkan sebagai berikut :

GAMBAR2. MODEL KERJA SERVER DNS



b. Gambar contoh Model Kerja Server DNS

Bagian resolver adalah bagian dari program aplikasi yang berfungsi menjawab setiap pertanyaan tentang domain. Untuk menjawab pertanyaan, resolver dapat mencari jawabannya dengan memeriksa isi cache dan meneruskan pertanyaan ke server DNS atau bertanya langsung ke server DNS. Cache adalah bagian yang menyimpan pertanyaan-pertanyaan tentang domain yang pernah diajukan sebelumnya. Setiap aplikasi internet yang ingin menghubungi host lain pasti akan berinteraksi dengan server DNS melalui resolver.

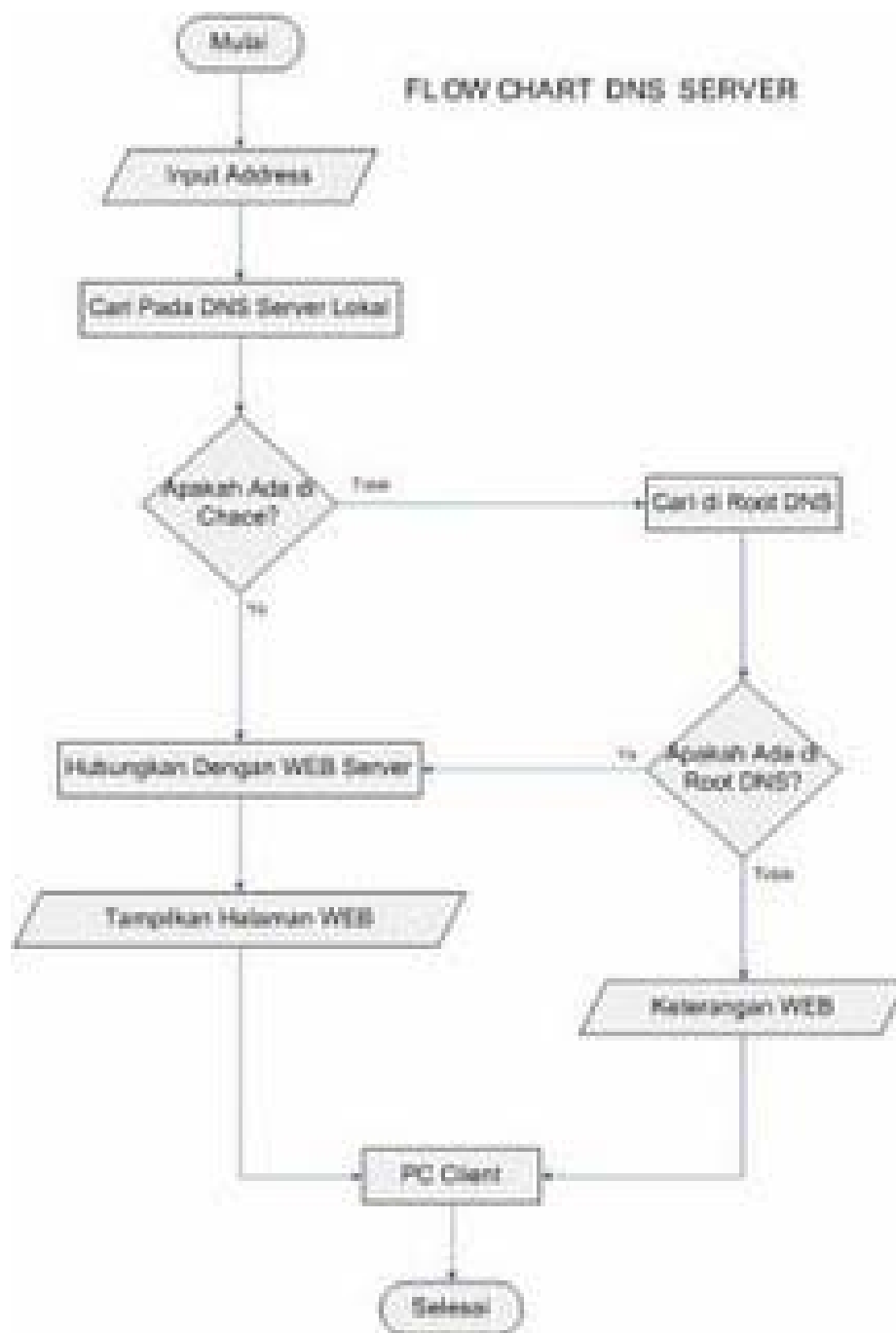
Jika server DNS tidak memiliki jawaban dari pertanyaan domain (name server bukan authoritative name server dari domain yang ditanyakan), maka server DNS akan meneruskan pertanyaan tersebut ke name server yang dianggapnya lebih tahu. Proses pengalihan ke name server (NS) lain (proses *name resolution*) ini merupakan proses iteratif yang berlangsung hingga diperoleh alamat lengkap sebuah host.

Sebagai contoh jika ada user yang ingin menghubungi `pandu.ee.ugm.ac.id`, maka mula-mula NS akan menghubungi root server. Karena root server tidak memiliki informasi untuk host `pandu.ee.ugm.ac.id`, maka root server akan memberikan referensi kepada NS untuk menghubungi authoritative name server terdekat yaitu name server untuk domain *id*. Dari name server domain *id*, NS kembali memperoleh referensi untuk menghubungi name server domain *ac*. Proses ini terus berlangsung hingga NS memperoleh informasi lengkap tentang host `pandu.ee.ugm.ac.id`.

Setiap request domain akan disimpan di dalam cache untuk jangka waktu tertentu. Hal ini bertujuan untuk mempercepat proses name resolution. Sebagai contoh jika sebelumnya ada user yang me-request informasi untuk `pandu.ee.ugm.ac.id`, maka untuk user yang me-request `fileserver.lapi.ugm.ac.id` akan diarahkan ke name server yang memiliki korelasi terdekat dengan domain yang pernah di-request yang tersimpan di dalam cache, yaitu name server *ugm.ac.id*.

Flowchart dari Prinsip Kerja DNS adalah sebagai berikut :

GAMBAR3. FLOW CHART PRINSIP KERJA DNS



c. Gambar Flow Chart Prinsip Kerja DNS

IV. CONTOH PENERAPAN DNS

Setelah kita mengetahui apakah itu DNS, bagaimana sejarahnya, mengerti bagaimana struktur hingga memahami prinsip kerjanya maka disini akan diberikan contoh bagaimana penerapan DNS didunia nyata. Disini ada beberapa contoh sebagai penggambaran cara kerja DNS. Contohnya yaitu :

A. *Contoh 1: Contoh Umum Menggunakan Browser*

Di dunia nyata, user tidak berhadapan langsung dengan DNS resolver - mereka berhadapan dengan program seperti web browser (Mozilla Firefox, Safari, Opera, Internet Explorer, Netscape, Konqueror dan lain-lain dan klien mail (Outlook Express, Mozilla Thunderbird dan lain-lain). Ketika user melakukan aktivitas yang meminta pencarian DNS (umumnya, nyaris semua aktivitas yang menggunakan Internet), program tersebut mengirimkan permintaan ke DNS Resolver yang ada di dalam sistem operasi. DNS resolver akan selalu memiliki cache (lihat diatas) yang memiliki isi pencarian terakhir. Jika cache dapat memberikan jawaban kepada permintaan DNS, resolver akan menggunakan nilai yang ada di dalam cache kepada program yang memerlukan.

Kalau cache tidak memiliki jawabannya, resolver akan mengirimkan permintaan ke server DNS tertentu. Untuk kebanyakan pengguna di rumah, Internet Service Provider(ISP) yang menghubungkan komputer tersebut biasanya akan menyediakan server DNS: pengguna tersebut akan mendata alamat server secara manual atau menggunakan DHCP untuk melakukan pendataan tersebut. Jika administrator sistem telah mengkonfigurasi sistem untuk menggunakan server DNS mereka sendiri, DNS resolver umumnya akan mengacu ke server nama mereka. Server nama ini akan mengikuti proses yang disebutkan di Teori DNS, baik mereka menemukan jawabannya maupun tidak. Hasil pencarian akan diberikan kepada DNS resolver; diasumsikan telah ditemukan jawaban, resolver akan menyimpan hasilnya di cache untuk penggunaan berikutnya, dan memberikan hasilnya kepada software yang meminta pencarian DNS tersebut.

Sebagai bagian akhir dari kerumitan ini, beberapa aplikasi seperti web browser juga memiliki DNS cache mereka sendiri, tujuannya adalah untuk mengurangi penggunaan referensi DNS resolver, yang akan meningkatkan kesulitan untuk melakukan debug DNS, yang menimbulkan kerancuan data yang lebih akurat. Cache seperti ini umumnya memiliki masa yang singkat dalam hitungan 1 menit.

Misalkan saja kita browsing di warnet, dan akan menghubungi www.google.com .Rangkaian proses yang akan dijalankannya adalah sebagai berikut:

- 1) PC kita mengontak Server DNS lokal (biasanya terletak pada jaringan ISP) untuk menanyakan IP Address www.google.com .
- 2) Server DNS lokal akan melihat ke dalam cache-nya .
- 3) Jika data itu terdapat di dalam cache server DNS server lokal, maka server tersebut akan memberikan alamat IP tersebut ke Browser. Jika tidak, maka server tersebut mengontak server DNS di atasnya (biasanya disebut Root DNS server) untuk mengetahui alamat IP dari DNS server yang mengelola domain www.google.com.
- 4) Jika domain www.google.com benar-benar exist, maka Root DNS akan mendapatkan alamat IP server DNS www.google.com,kemudian alamat dikirim ke server DNS lokal kita .
- 5) Server DNS lokal akan mengontak Server DNS www.google.com untuk menanyakan alamat IP dari www.google.com , dan Server DNS www.google.com memberikan data alamat IP www.google.com
- 6) Server DNS lokal memberitahu alamat IP untuk www.google.com kepada Browser/Client (PC kita).

7) Kemudian kita menggunakan alamat itu untuk diisikan ke dalam IP Packet untuk menghubungi www.google.com .

V. PENUTUP

Sistem yang dijabarkan diatas memberikan skenario yang disederhanakan. DNS meliputi beberapa fungsi lainnya yaitu nama host dan alamat IP tidak berarti terhubung secara satu-banding-satu. Banyak nama host yang diwakili melalui alamat IP tunggal: gabungan dengan pengasuhan maya (virtual hosting), hal ini memungkinkan satu komputer untuk melayani beberapa situs web. Selain itu, sebuah nama host dapat mewakili beberapa alamat IP: ini akan membantuk toleransi kesalahan (fault tolerance dan penyebaran beban (load distribution), juga membantu suatu situs berpindah dari satu lokasi fisik ke lokasi fisik lainnya secara mudah.

Ada cukup banyak kegunaan DNS selain menerjemahkan nama ke alamat IP. Contoh:, agen pemindahan surat Mail transfer agents(MTA) menggunakan DNS untuk mencari tujuan pengiriman E-mail untuk alamat tertentu. Domain yang menginformasikan pemetaan exchange disediakan melalui rekod MX (MX record) yang meningkatkan lapisan tambahan untuk toleransi kesalahan dan penyebaran beban selain dari fungsi pemetaan nama ke alamat IP.

Kerangka Peraturan Pengiriman (Sender Policy Framework) secara kontroversi menggunakan keuntungan jenis rekod DNS, dikenal sebagai rekod TXT. Menyediakan keluwesan untuk kegagalan komputer, beberapa server DNS memberikan perlindungan untuk setiap domain. Tepatnya, tigabelas server akar (root servers) digunakan oleh seluruh dunia. Program DNS maupun sistem operasi memiliki alamat IP dari seluruh server ini. Amerika Serikat memiliki, secara angka, semua kecuali tiga dari server akar tersebut. Namun, dikarenakan banyak server akar menerapkan anycast, yang memungkinkan beberapa komputer yang berbeda dapat berbagi alamat IP yang sama untuk mengirimkan satu jenis services melalui area geografis yang luas, banyak server yang secara fisik (bukan sekedar angka) terletak di luar Amerika Serikat.

DNS menggunakan TCP dan UDP di port komputer 53 untuk melayani permintaan DNS. Nyaris semua permintaan DNS berisi permintaan UDP tunggal dari klien yang diikuti oleh jawaban UDP tunggal dari server. Umumnya TCP ikut terlibat hanya ketika ukuran data jawaban melebihi 512 byte, atau untuk pertukaran zona DNS zone transfer

VI. KESIMPULAN

DNS adalah hasil pengembangan dari metode pencarian host name terhadap IP address di Internet. Pada DNS client (resolver) mengirimkan queries ke Name Server (DNS). Name Server akan menerima permintaan dan memetakan nama komputer ke IP address. Domain Name Space adalah pengelompokan secara hirarki yang terbagi atas root-level domains, top-level domains, second-level domains, dan host names.

Keberadaan DNS sangat penting dalam memberikan alamat untuk berinteraksi serta memberikan informasi di alam rimba maya (virtual). Selain itu juga acuannya di pakai sebagai penggunaan legal domain yang terdiri dari aspek:

A. *Registrant*

DNS adalah hasil pengembangan dari metode pencarian host name terhadap IP address di Internet. Pada DNS client (resolver) mengirimkan queries ke Name Server (DNS). Name Server akan menerima permintaan dan memetakan nama komputer ke IP address. Domain Name Space adalah pengelompokan secara hirarki yang terbagi atas root-level domains, top-level domains, second-level domains, dan host names.

Keberadaan DNS sangat penting dalam memberikan alamat untuk berinteraksi serta memberikan informasi di alam rimba maya (virtual).

Tidak satupun individu di dunia yang "memiliki" nama domain kecuali Network Information Centre (NIC), atau pendaftar nama domain (domain name registry). Sebagian besar dari NIC di dunia menerima biaya tahunan dari para pengguna legal dengan tujuan bagi si pengguna legal menggunakan nama domain tersebut. Jadi sejenis perjanjian sewa-menyewa terjadi, bergantung kepada syarat dan ketentuan pendaftar. Bergantung kepada beberapa peraturan penamaan dari para pendaftar, pengguna legal dikenal sebagai "pendaftar" (registrants) atau sebagai "pemegang domain" (domain holders)

ICANN memegang daftar lengkap untuk pendaftar domain di seluruh dunia. Siapapun dapat menemukan pengguna legal dari sebuah domain dengan mencari melalui basis data WHOIS yang disimpan oleh beberapa pendaftar domain.

Di (lebih kurang) 240 country code top-level domains (ccTLDs), pendaftar domain memegang sebuah acuan WHOIS (pendaftar dan nama server). Contohnya, IDNIC, NIC Indonesia, memegang informasi otoritatif WHOIS untuk nama domain .ID.

Namun, beberapa pendaftar domain, seperti VeriSign, menggunakan model pendaftar-pengguna. Untuk nama domain .COM dan .NET, pendaftar domain, VeriSign memegang informasi dasar WHOIS (pemegang domain dan server nama). Siapapun dapat mencari detail WHOIS (Pemegang domain, server nama, tanggal berlaku, dan lain sebagainya) melalui pendaftar.

Sejak sekitar 2001, kebanyakan pendaftar TLD (.ORG, .BIZ, .INFO) telah mengadopsi metode penfatar "tebal", menyimpan otoritatif WHOIS di beberapa pendaftar dan bukan pendaftar itu saja.

B. *Kontak Administratif (Administrative Contact)*

Satu pemegang domain biasanya menunjuk kontak administratif untuk menangani nama domain. Fungsi manajemen didelegasikan ke kontak administratif yang mencakup (diantaranya): keharusan untuk mengikuti syarat dari pendaftar domain dengan tujuan memiliki hak untuk menggunakan nama domain otorisasi untuk melakukan update ke alamat fisik, alamat email dan nomor telepon dan lain sebagainya via WHOIS

C. *Politik*

Banyak penyelidikan telah menyuarakan kritik dari metode yang digunakan sekarang untuk mengatur kepemilikan domain. Umumnya, kritik mengklaim penyalahgunaan dengan monopoli, seperti VeriSign Inc dan masalah-masalah dengan penunjukkan dari top-level domain (TLD). Lembaga internasional ICANN (Internet Corporation for Assigned Names and Numbers) memelihara industri nama domain.

REFERENCES

- [1] Purbo, Onno W. *Standar, Desain, dan Implementasi TCP/IP*. 1998. Elex Media Komputindo
- [2] *ARC Handbook I*. 2006. Amateur Radio Club UGM.
- [3] McNab, Chris. *Network Security Assessment*. 2004. O'Reilly.
- [4] R. Hidayat, "Paper Template in One-Column Format", <http://www.te.ugm.ac.id/~risanuri/elan/>