Nama : Rizky Soufi Gustiawan
NIM : 09011281520111
Kelas : SK 5 C

Tugas Jarkom

Layanan dari TCP

Memiliki layanan flow control: Untuk mencegah data terlalu banyak dikirimkan pada satu waktu, yang akhirnya membuat "macet" jaringan internetwork IP, TCP mengimplementasikan layanan flow control yang dimiliki oleh pihak pengirim yang secara terus menerus memantau dan membatasi jumlah data yang dikirimkan pada satu waktu. Untuk mencegah pihak penerima untuk memperoleh data yang tidak dapat disangganya (buffer), TCP juga mengimplementasikan flow control dalam pihak penerima, yang mengindikasikan jumlah buffer yang masih tersedia dalam pihak penerima.

Network File System (NFS). Pelayanan akses file-file jarak jauh yg memungkinkan klien-klien untuk mengakses file-file pada komputer jaringan jarak jauh walaupun file tersebut disimpan secara lokal. (lihat RFC 1001 dan 1002 untuk keterangan lebih lanjut)

1. World Wide Web

Aplikasi ini pada prinsipnya mirip dengan aplikasi gopher, yakni penyediaan database yang dapat diakses tidak hanya berupa text, namun dapat berupa gambar/image, suara, video. penyajiannya pun dapat dilakukan secara live. Dengan demikian, jenis informasi yang dapat disediakan sangat banyak dan dapat dibuat dengan tampilan yang lebih menarik. Hal ini dimungkinkan karena Web menggunakan teknologi hypertext. Karena itu, protokol yang digunakan untuk aplikasi ini dikenal dengan nama Hypertext-transfer-protocol (HTTP).

2. Archie

Aplikasi FTP memungkinkan kita mentransfer file dari manapun di seluruh dunia. Hal itu dengan anggapan bahwa kita telah mengetahui lokasi di mana file yang kita cari berada. Namun jika kita belum mengetahui di mana file yang kita cari berada, kita memerlukan aplikasi untuk membantu kita mencari di mana file tersebut berada.
Cara kerja Archie dapat dijelaskan sebagai berikut. Server Archie secara berkala melakukan anonymous ftp ke sejumlah FTP Server dan mengambil informasi daftar seluruh file yang ada pada FTP Server tersebut. Daftar ini disusun berdasarkan letak file dalam direktori/sub direktori, sehingga mudah untuk menemukan file tersebut. File-file yang berisi daftar file tiap FTP Server ini merupakan database dari Archie Server. Jika ada query ke Archie Server yang menanyakan suatu file, server mencari dalam daftar tadi dan mengirimkan seluruh jawaban yang berkaitan dengan file tersebut. Informasi yang diberikan adalah alamat FTP Server yang memiliki file tersebut dan letak file tersebut dalam struktur direktori.

3. Wide Area Information Services (WAIS)

WAIS merupakan salah satu servis pada internet yang memungkinkan kita mencari melalaui materi yang terindeks dan menemukan dokumen/artikel berdasarkan isi artikel tersebut. Jadi pada dasarnya, WAIS memberikan layanan untuk mencari artikel yang berisi kata-kata kunci yang kita ajukan sebagai dasar pencarian.
Aplikasi WAIS biasanya berbasis text. Untuk membuat suatu dokumen dapat dicari melalaui WAIS Server, harus dibuat terlebih dahulu index dari dokumen tersebut. Setiap kata dalam

dokumen tersebut diurut dan dihitung jumlahnya. Jika ada query dari client, index akan diperiksa dan hasilnya, yakni dokumen yang memiliki kata-kata tersebut ditampilkan. Karena kemungkinan ada banyak dokumen yang memiliki kata-kata yang kita ajukan, maka beberapa dokumen yang memiliki kata kunci tersebut diberi skor/nilai. Dokumen yang paling banyak mengandung kata-kata kunci akan mendapat skor tertinggi. Dengan demikian, user mendapatkan informasi kemungkinan terbesar dari bebarapa dokumen yang mengandung kumpulan kata yang diajukannya.

4. FAX di Internet

Mesin FAX sebagai pengirim dan penerima berita tertulis melalaui telepon saat ini hampir dimiliki oleh semua kantor. Melalaui gateway Internet FAX, pengiriman FAX dapat dilakukan melalaui e-mail. Gateway akan menerjemahkan pesan e-mail tersebut dan menghubungi mesin FAX tujuan melalui jalur telepon secara otomatis. Tentu saja, akses untuk ini terbatas (private).

Layanan dari UDP

Digunakan untuk multimedia streaming, yang sangat memberikan toleransi kehilangan segment cukup baik dan yang sangat tidak sensitive terhadap kerusakan atau kehilangan segment
Contoh protokol aplikasi yang menggunakan UDP :

1. DNS (Domain Name System) 53

DNS (Domain Name System) adalah sebuah sistem yang menyimpan informasi tentang nama host maupun nama domain dalam bentuk basis data tersebar (distributed database) di dalam jaringan komputer, misalkan: Internet. Domain Name System ini merupakan sistem penamaan hirarkis yang nantinya didistribusikan untuk suatu komputer, jasa, atau sumber daya terhubung ke Internet maupun jaringan pribadi. DNS biasanya digunakan sebuah Layanan Nama Domain untuk menyelesaikan permintaan untuk nama-nama website menjadi alamat IP untuk tujuan menemukan layanan komputer serta perangkat di seluruh dunia.
Fungsi DNS (Domain Name System) :
Fungsi dasar dari DNS (Domain Name System) adalah untuk menerjemahkan atau mentranslasikan alamat ip menjadi sebuah nama domain dan juga sebaliknya. Contohnya saja alamat facebook.com, google.com, dan situs-situs lainnya merupakan alamat ip dari situs tersebut yang kemudian ditranslasikan menjadi sebuah nama domain.

Manfaat DNS (Domain Name System) :

Manfaat yang paling umum dari DNS (Domain Name System) tentu saja untuk mempermudah pengguna dalam mengakses situs yang kita buat. Secara umum manusia lebih mudah mengingat kata dari pada mengingat angka, karena itu para pengguna internet akan lebih mudah untuk mengingat alamat situs kita berupa nama domain daripada berupa alamat ip.

2. SNMP, (Simple Network Management Protocol) 161, 162

SNMP adalah sebuah protokol yang dirancang untuk memberikan kemampuan kepada pengguna untuk memantau dan mengatur jaringan komputernya secara sistematis dari jarak jauh atau dalam satu pusat kontrol saja. Pengolahan ini dijalankan dengan menggumpulkan data dan melakukan penetapan terhadap variabel-variabel dalam elemen jaringan yang dikelola.

Elemen-elemen SNMP

Manajer adalah pelaksana dan manajemen jaringan. Pada kenyataannya manager ini merupakan komputer biasa yang ada pada jaringan yang mengoperaksikan perangkat lunak untuk manajemen jaringan. Manajer ini terdiri atas satu proses atau lebih yang berkomunikasi dengan agen-agennya dan dalam jaringan. Manajer akan mengumpulkan informasi dari agen dari jaringan yang diminta oleh administrator saja bukan semua informasi yang dimiliki agen. MIB atau *Manager Information Base*, dapat dikatakan sebagai struktur basis data variabel dari elemen jaringan yang dikelola. Struktrur ini bersifat hierarki dan memiliki aturan sedemikian rupa sehingga informasi setiap variabel dapat dikelola atau ditetapkan dengan mudah.

3. TFTP (Trivial File Transfer Protocol) 69

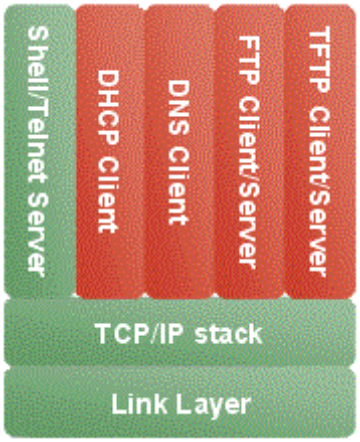
TFTP merupakan sebuah protokol sederhana untuk transfer file antar komputer yang sama maupun berbeda jaringan. TFTP dirancang khusus dengan ukuran kecil dan didimplementasikan. Oleh sebab itu TFTP mempunyai lebih banyak kekurangan dibandingkan dengan protokol FTP biasa. Tugas yang dikerjakan oleh TFTP adalah membaca dan menulis file atau mail dari/ke komputer server.

TFTP tidak dapat me-list direktori dan tidak mempunyai kelengkapan untuk keamanan user. Pada konsep LTSP dibutuhkan TFTP karena protokol ini digunkan untuk mengambil image kernel dari komputer server ke komputer client. Protokol ini memiliki jumlah memori yang sedikit untuk menjalankan kodenya, sehingga dapat dengan mudah dipasang pada bootROM komputer.

TFTP menggunakan protokol UDP sebagai transport karena tidak membutuhkan terciptanya koneksi sebelum permintaan transfer file dapat terlaksana. Karena menggunakan protokol UDP yang tidak membentuk koneksi sebelum berhubungan, maka keamanan dalam pengiriman data tidak dapat dijamin.
Dibuat berdasarkan protokol yang sebelumnya Easy File Transfer Protocol (-EFTP-), yang merupakan bagian dari kumpulan Protokol PARC Universal Packet (-PUP-). Saat pengembangan Protokol TCP/IP, TFTP merupakan protokol pertama kali yang diimplementasikan dalam HOST jaringan,

Karena Ssederhana, Berbaris, Trivial File Transfer Protocol memungkinkan Klien Mendapatkan File atau Meletakkan File ke Remote Host. Salah satu kegunaan utamanya adalah pada tahap awal Node Boot dari Jaringan Area Lokal.

TFTP Sangat sederhana dan tanpa Otentikasi. Didefinisikan pada tahun 1980 serta menjadi Standar pada tahun 1981 dengan Spesifikasi RFC 1350. Karena memang sangat Sederhana dan tidak ada Otentikasi, TFTP kurang dalam Fitur Keamanan maka tidak dianjurkan menggunakan TFTP.



Kecil dalam ukuran, Keuntungan besar bagi Sistem.

► Embedded Sistem dapat memiliki TFTP di ROM dan menggunakannya untuk mendapatkan pemetaan Memori awal ketika sistem dinyalakan.

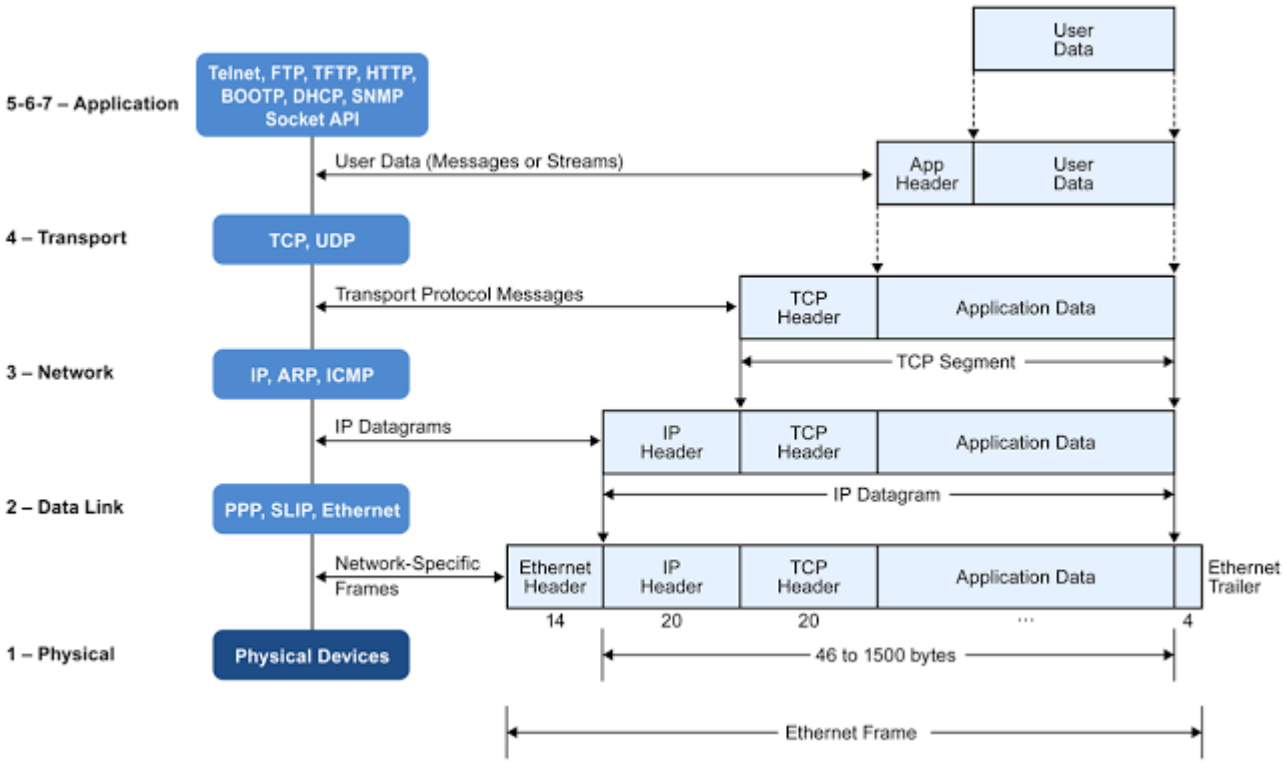
► Ketika pembaruan gambaran memori awal yang dibutuhkan, pembharuan gambaran sudah cukup dan tidak memerlukan perubahan ke sistem itu sendiri

Tidak seperti FTP, TFTP berjalan diatas UDP (Port 69).

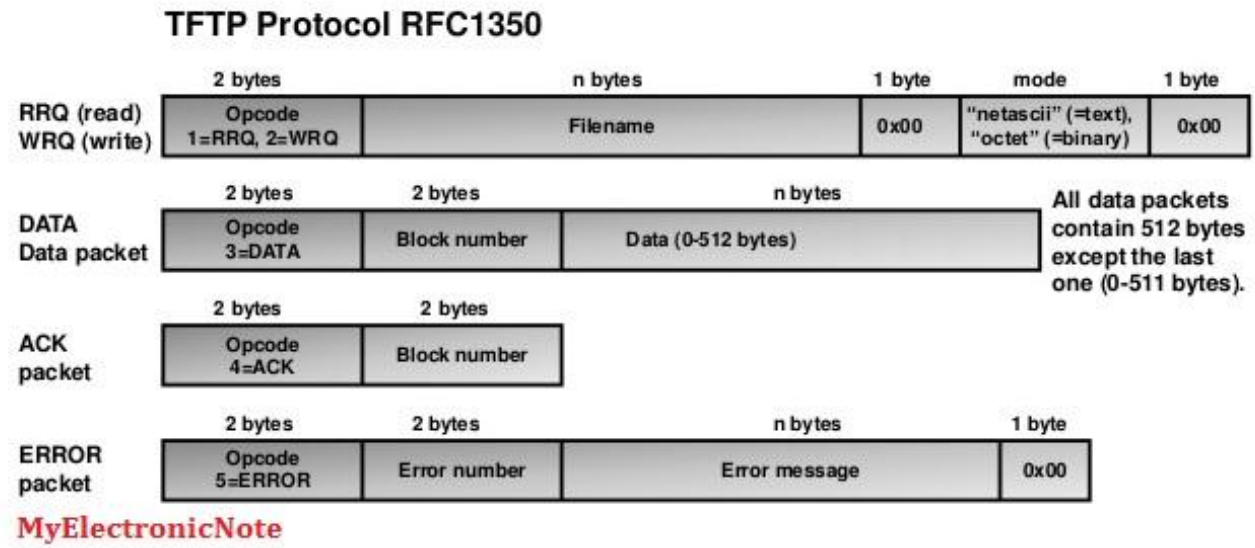
► Sejak UDP tidak dapat diandalkan, TFTP menggunakan batas waktu dan transmisi untuk memastikan data.

► Pengiriman sisi mengirimkan file dalam ukuran tetap (512 Bbyte) blok dan menunggu Pengakuan untuk setiap blok sebelum mengirim berikutnya.

► Penerima mengakui setiap blok pada penerimaan.



Akhirnya, **TFTP** pun digunakan untuk melakukan **BOOTING Komputer** seperti halnya Router jaringan komputer yang tidak memiliki perangkat penyimpanan data. Protokol ini kini masih digunakan untuk mentransfer berkas-berkas kecil antar host di dalam sebuah jaringan, seperti halnya ketika terminal jarak jauh X Window System atau Thin Client lainnya melakukan Proses Booting dari sebuah Host Jaringan atau Server.

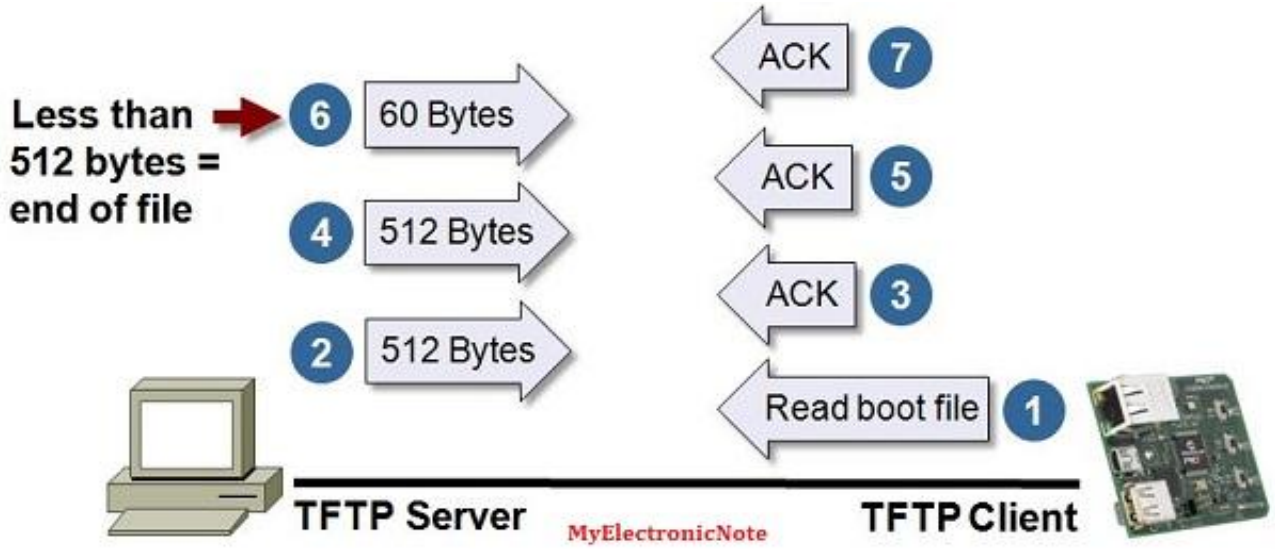


Versi asli TFTP, sebelum direvisi oleh RFC 1350, menampilkan sebuah kelemahan protokol, yang diberinama **Sorcerer's Apprentice Syndrome**, Saat pertama kali diketemukan.

TFTP Pertama kali muncul sebagai bagian dari Sistem Operasi 4.3 BSD. Protokol ini juga masih dimasukkan ke dalam Mac OS X, paling tidak hingga versi 10.5.

Akhir-akhir ini, **TFTP** sering digunakan oleh Worm Komputer, seperti W32.Blaster, sebagai

Metode untuk menyebarkan dirinya dan menginfeksi Host Jaringan
lainnya. **TFTP** digunakan juga untuk menginstal komputer melalui jaringan.



Paket pertama dikirim meminta Transfer File dan menetapkan Interaksi Klien dan Server.
► Juga menentukan Nama File dan apakah File akan dibaca, dipindahkan ke Klien, atau tertulis, dipindahkan ke Server.

Blok File diberi nomor berurutan mulai dari 1.
► Setiap paket data berisi Header yang menentukan jumlah blok yang dibawanya, dan masing-masing pengakuan mengandung jumlah yang blok yang diakui.
► Blok kurang dari 512 byte Sinyal Akhir Diajukan (EOF)

Hal ini dimungkinkan untuk mengirim pesan kesalahan baik di tempat data atau pengakuan.
► Setelah kesalahan, Transfer akan dihentikan.

Setelah membaca atau menulis permintaan yang telah dibuat, Server menggunakan alamat IP dan port UDP nomor klien untuk mengidentifikasi Operasi berikutnya.
► Demikian, tak satu pun dari pesan data atau pesan ack perlu menentukan nama file.

Pesan hilang dapat ditransmisikan ulang setelah waktu habis.
► Namun, sebagian besar kesalahan lain hanya menyebabkan pemutusan Interaksi, karena TFTP adalah dimaksudkan untuk menjadi sederhana!

TFTP memastikan kedatangan datanya dengan mewajibkan setiap sisi untuk menerapkan timeout dan transmisi.
► Jika Pengiriman Data Waktu habis, Akan mengrim ulang Blok Data terakhir.
► Jika Pengakuan Waktu habis, Akan mengrim ulang Pengakuan terakhir.

Masalah, yang dikenal **"Sorcerer's Apprentice Bug"**,
Muncul ketika sebuah pengakuan Data Paket k tertunda, tapi tidak hilang.
► Pengirim mentransmisikan kembali Paket Data.
► Setiap saat Pengakuan tiba, dan Setiap Pemicu pengiriman Paket Data k + 1.
► Masalah ini diperbaiki pada versi terbaru dari TFTP.

4. SunRPC port 111.

RPC adalah suatu protokol yang menyediakan suatu mekanisme komunikasi antar proses yang mengijinkan suatu program untuk berjalan pada suatu komputer tanpa terasa adanya eksekusi kode pada sistem yang jauh (remote system). RPC mengasumsi keberadaan dari low-level protokol transportasi seperti TCP atau UDP untuk membawa pesan data dalam komunikasi suatu program. Protokol RPC dibangun diatas protokol eXternal Data Representation (XDR), yang merupakan standar dari representasi data dalam komunikasi remote. Protokol XDR mengubah parameter dan hasil dari tiap servis RPC yang disediakan. Protokol RPC mengijinkan pengguna (users) untuk bekerja dengan prosedur remote sebagaimana bekerja dengan prosedur lokal. Prosedur panggilan remote (remote procedure calls) didefinisikan melalui rutin yang terkandung didalam protokol RPC. Tiap message dari panggilan akan disesuaikan dengan message balikan. Protokol RPC sendiri sebenarnya adalah suatu protokol untuk ”meneruskan pesan” yang mengimplemntasikan protokol non-RPC lain seperti panggilan remote batching dan broadcasting. Protokol ini juga mendukung adanya prosedur callback dan select subroutine pada sisi server. Klien dan Server Klien adalah komputer atau proses yang mengakses suatu servis/layanan atau resources dari proses atau komputer pada suatu jaringan. Server adalah komputer yang menyediakan servis/layanan dan resources, dan yang mengimplementasikan servis jaringan. Tiap servis pada network adalah susunan dari program remote, dan tiap program remote mengimplementasi prosedur remote. Semua prosedur berikut parameternya dan hasilnya didokumentasi secara spesifik pada protokol suatu program. Protokol Message RPC Protokol Message RPC didefinisikan dengan menggunakan deskripsi data eXternal Data Representation (XDR) yang meliputi struktur, enumerasi dan union. Pembahasan lebih lanjut akan diterangkan pada bab berikutnya mengenai implementasi RPC.

Protokol Message ini membutuhkan faktor-faktor pendukung sebagai berikut :

1. Spesifikasi yang unik untuk tiap prosedur call

2. Respon message yang sesuai untuk tiap message yang diminta

3. Otentifikasi klien untuk tiap layanan dan sebaliknya Protokol Message RPC memiliki dua (2) struktur yang berbeda, yaitu call message dan reply message. Tiap klien yang akan melakukan RPC pada suatu server di jaringan akan menerima balasan (reply) berupa hasil dari eksekusi prosedur tersebut. Dengan menggunakan spesifikasi yang unik untuk tiap prosedure remote, maka RPC dapat mencocokkan message balasan untuk tiap call message yang diminta klien. Call Message Tiap call message pada RPC mengandung nilai-nilai unsigned integer yang digunakan untuk mengidentifikasi prosedur remote yang diminta. Nilai-nilai ini adalah :

1. Nomor Program

2. Nomor Versi dari Program

3. Nomor Prosedur Reply Message Reply message yang dikirimkan oleh server jaringan bervariasi tergantung apakah call messages yang diminta klien diterima atau ditolak. Reply

message mengandung informasi yang digunakan untuk membedakan kondisi-kondisi yang diminta sesuai dengan call messages.

Informasi ini antara lain :

1. RPM mengeksekusi call message dengan sukses

2. Implementasi remote tidak sesuai dengan protokol yang digunakan. Versi yang lebih rendah atau tinggi akan ditolak.

3. Program remote tidak tersedia pada sistem remote

4. Program remote tidak mendukung versi yang diminta klien

5. Nomor prosedur yang diminta tidak ada. Fitur dalam RPC RPC memiliki fitur - fitur sebagai berikut : batching calls, broadcasting calls, callback procedures dan using the select subroutine. Batching Calls Fitur Batching calls mengijinkan klien untuk mengirim message calls ke server dalam jumlah besar secara sequence (berurutan). Batching menggunakan protokol streaming byte seperti TCP / IP sebagai mediumnya. Pada saat melakukan batching, klien tidak menunggu server untuk memberikan reply terhadap tiap messages yang dikirim, begitu pula dengan server yang tidak pernah mengirimkan messages reply. Fitur inilah yang banyak digunakan klien, karena arsitektur RPC didesain agar pada tiap call message yang dikirimkan oleh klien harus ada proses menunggu balasan dari server. Oleh karena itu maka pihak klien harus dapat mengatasi error yang kemungkinan terjadi karena pihak klien tidak akan menerima peringatan apabila terjadi error pada message yang dikirim. Broadcasting Calls Fitur Broadcasting mengijinkan klien untuk mengirimkan paket data ke jaringan dan menunggu balasan dari network. FItur ini menggunakan protokol yang berbasiskan paket data seperti UDP/IP sebagai mediumnya. Broadcast RPC membutuhkan layanan port mapper RPC untuk mengimplementasikan fungsinya. Callback Procedures Fitur Callback Procedures mengijinkan server untuk bertindak sebagai klien dan melakukan RPC callback ke proses yang dijalankan oleh klien. Menggunakan select Subrutin Fitur ini akan memeriksa deskripsi dari suatu file dan messages dalam antrian untuk melihat apakah mereka siap untuk dibaca (diterima) atau ditulis (dikirim), atau mereka dalam kondisi ditahan sementara. Prosedur ini mengijinkan server untuk menginterupsi suatu aktivitas, memeriksa datanya, dan kemudian melanjutkan proses aktivitas tersebut. Otentifikasi RPC Proses otentifikasi adalah proses yang digunakan untuk mengidentifikasi server dan klien pada RPC. Untuk setiap prosedur remote yang dilakukan protokol RPC menyediakan slot yang dipakai sebagai parameter otentifikasi yang berfungsi agar pemanggil (caller) dapat, memberikan identitasnya kepada server. Parameter otentifikasi ini dibuat di paket klien. Otentifikasi RPC terdiri atas beberapa bagian.

Berikut ini adalah bagian-bagian padaotentifikasi RPC :

1. Protokol Otentifikasi RPC Protokol Otentifikasi RPC disediakan sebagai bagian dari protokol RPC. Untuk setiap prosedur remote, semuanya diotentifikasi oleh paket RPC pada server. Parameter yang digunakan adalah respon verifier. Sedangkan pada pihak klien, setiap

paket RPC diberikan parameter otentifikasi dan parameter yang digunakan adalah credential dan verifier.

2. Otentifikasi NULL Otentifikasi NULL digunakan pada sistem dimana pemanggil (caller) RPC tidak mengetahui identitasnya sendiri dan server tidak membutuhkan identitas pemanggil.

3. Otentifikasi UNIX Otentifikasi Unix digunakan pada prosedur remote di sistem UNIX. Jenis otentifikasi ini dibagi dua (2) yaitu otentifikasi pada sisi klien dan otentifiksi pada sisi server. Pada sisi klien, otentifikasi ini akan membuat otentifikasi handle dengan AIX permissions agar dapat berasosiasi dengan parameter credentials pada sistem UNIX. Sedangkan pada sisi server, server harus dapat menentukan tipe otentifikasi yang diberikan oleh pemanggil RPC. Penentuan dukungan terhadap tipe otentifikasi akan memberikan reply yang berbeda.

4. Otentifikasi Data Encryption Standard (DES) Otentifikasi DES membutuhkan keyserv daemon yang harus berjalan baik di sisi server maupun klien. Tiap pengguna pada sistem ini harus memiliki kunci public (public key yang disahkan pada database kunci publik oleh Administrator jaringan tersebut.

5. Protokol Otentifikasi DES Protokol Otentifikasi DES meliputi protokol penanganan DES pada proses otentifikasi RPC. Protokol ini mencakup 64-bit blok data DES yang terenkripsi dan menentukan panjang maksimum untuk user name pada jaringan yang digunakan.

6. Enkripsi Diffie-Hellman Enkripsi Diffie-Hellman digunakan pada pembuatan kunci public pada otentifikasi DES dengan menggunakan 192-bit kunci. Enkripsi ini memiliki dua buah variable onstan, yaitu BASE dan MODULUS yang digunakan pada protokol otentifikasi DES. PC berhubungan hanya dengan proses otentifikasi, tidak dengan kontrol akses terhadap ervices/layanan individual yang diberikan. Tiap layanan mengimplementasikan eraturan mengenai kontrol akses masing-masing. Subsistem otentifikasi pada paket RPC bersifat open-ended, artinya beberapa tentifikasi dapat diasosiasikan pada RPC klien.