

Serviços de Rede

Especialização em Redes e Sistemas Distribuídos

Emerson Ribeiro de Mello

emerson @ das.ufsc.br

<http://www.das.ufsc.br/~emerson/srede>

Universidade do Planalto Catarinense

9 de fevereiro de 2005

Sumário

- 1 Serviço de nomes
- 2 FTP – File Transfer Protocol
- 3 Servidor WEB
- 4 Serviço de Email
- 5 Administração remota
- 6 Inetd e Xinetd

Sumário

- 1 Serviço de nomes
- 2 FTP – File Transfer Protocol
- 3 Servidor WEB
- 4 Serviço de Email
- 5 Administração remota
- 6 Inetd e Xinetd

Arquitetura da Internet

- A Internet está baseada nos protocolos TCP/IP
- Cada máquina conectada na rede está endereçada através de um número IP
- Trabalhar com números é ideal para máquinas
 - IPv4 – 200.135.68.47
 - IPv6 – fe80:0:226:54ff:8:800:200C:20d6
- Porém, não para humanos. . .

Arquitetura da Internet

- A Internet está baseada nos protocolos TCP/IP
- Cada máquina conectada na rede está endereçada através de um número IP
- Trabalhar com números é ideal para máquinas
 - IPv4 – 200.135.68.47
 - IPv6 – fe80:0:226:54ff:8:800:200C:20d6
- Porém, não para humanos. . .

Serviço de nomes

- Faz o mapeamento entre nomes, ideais para humanos, e números, ideais para máquinas
- É a base para as demais aplicações
- Repositório “centralizado” que facilita a localização por todos da rede
- Alguns exemplos:
 - Arquivo `/etc/hosts`
 - DNS
 - *NameService* do CORBA

Arquivo /etc/hosts

- Relaciona endereços IP com nomes, ideal para
 - Endereços constantemente acessados
 - Quando não se dispõe de um Serviço de Nomes
 - Número pequeno de entradas
- Não é ideal quando
 - Grande número de entradas
 - Constante entrada/saída de registros
 - Necessidade de compartilhar a tabela com mais de uma máquina

Arquivo /etc/hosts

Exemplo de arquivo

```
# /etc/hosts  
127.0.0.1 localhost  
10.0.0.1 guaja.local.br guaja
```

- A configuração correta deste arquivo é imprescindível para o perfeito funcionamento do sistema

História...

- Em 1970 a ARPAnet era composta por poucas máquinas
- Um arquivo HOSTS.TXT continha as relações entre nomes e endereços de todas as máquinas da rede
- A SRI-NIC era a mantenedora do HOSTS.TXT
 - Sempre que um administrador alterasse algo em sua rede, ele enviava um e-mail para a NIC
 - Periodicamente todos os administradores puxavam via FTP o arquivo HOSTS.TXT
- Com o crescimento da rede surgiram alguns problemas
 - Tráfego e carga dos servidores da NIC
 - Colisões de nomes – duas máquinas não podem possuir um mesmo nome
 - Consistência entre todos os participantes da rede

História...

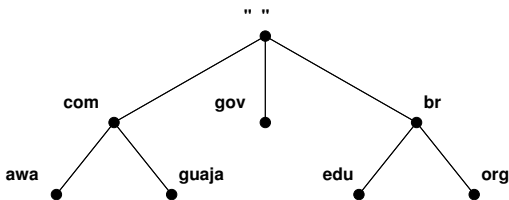
- Uma nova proposta deveria solucionar os problemas, tendo
 - Administração local dos dados , mas mantendo a disponibilidade global dos mesmos
 - Atualização dos dados de maneira mais rápida e fácil
 - Descentralização da administração, eliminando os problemas de gargalho
 - Sistema hierárquico de nomes, para garantir a unicidade dos mesmos
- DNS – *Domain Name System*
 - O conceito de domínios de nomes foi proposto nas RFCs 882 e 883, em 1983
 - As RFCs 1033, 1034 e 1035 são as atuais especificações do DNS (1987)

História...

- Uma nova proposta deveria solucionar os problemas, tendo
 - Administração local dos dados , mas mantendo a disponibilidade global dos mesmos
 - Atualização dos dados de maneira mais rápida e fácil
 - Descentralização da administração, eliminando os problemas de gargalho
 - Sistema hierárquico de nomes, para garantir a unicidade dos mesmos
- DNS – *Domain Name System*
 - O conceito de domínios de nomes foi proposto nas RFCs 882 e 883, em 1983
 - As RFCs 1033, 1034 e 1035 são as atuais especificações do DNS (1987)

DNS – *Domain Name System*

- DNS é um banco de dados distribuído
 - Uma árvore invertida, com profundidade máxima de 127 níveis
- Arquitetura cliente/servidor
 - **Name Servers** – provê informações sobre alguns segmentos da base
 - **Resolvers** – Geralmente, rotinas de uma biblioteca que geram consultas e as enviam através da rede para os *Name Servers*

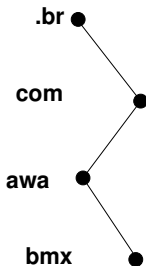


Estrutura do DNS

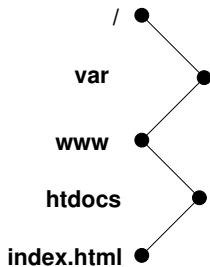
Domínio de nomes vs. Estrutura de diretório

- Todo nome (diretório) deve ser único
- Um domínio
 - Um nome é uma seqüência de rótulos iniciando pelo nó final e chegando até a raiz
- Caminho absoluto de um arquivo
 - O caminho absoluto de um arquivo consiste de nomes relativos iniciando pela raiz e chegando até o nó final

Estrutura do DNS (cont.)



bmx.awa.com.br



/var/www/htdocs/index.html



Necessidade do uso do DNS

- Rede local
- Rede TCP/IP composta por diversas subredes
- Conectado à Internet

Estrutura da árvore do DNS

- Cada nó possui um rótulo, o qual pode possuir até 63 caracteres
 - O rótulo “em branco” é reservado para a raiz
 - Nós irmãos não podem possuir um mesmo rótulo
- A posição do domínio no espaço de nomes de domínio pode ser
 - Domínio de nível superior é filho da raiz
 - Domínio de primeiro nível é filho de um nó de nível superior
 - Domínio de segundo nível é filho de um nó de primeiro nível, etc. . .
- Não há regras para especificar como deverão ser os rótulos
 - Existem tradições na nomenclatura – as vezes são quebradas

Estrutura da árvore do DNS

- Domínios de nível superior (TLDs)
 - com, edu, gov, org, mil, net, org
 - Com o crescimento da Internet, novos TLDs foram definidos: name, info, pro
 - IACNN – Internet Corporation for Assigned Names and Numbers é o órgão responsável
- TLDs com duas letras para cada país foram criadas seguindo seguindo a norma ISO 3166
 - Grã-Bretanha é uma exceção. De acordo com a norma o domínio deveria ser “gb”, mas é utilizado do domínio “uk”
 - Outra exceção são os subdomínios (edu, com), lá tratados como (ac, co)

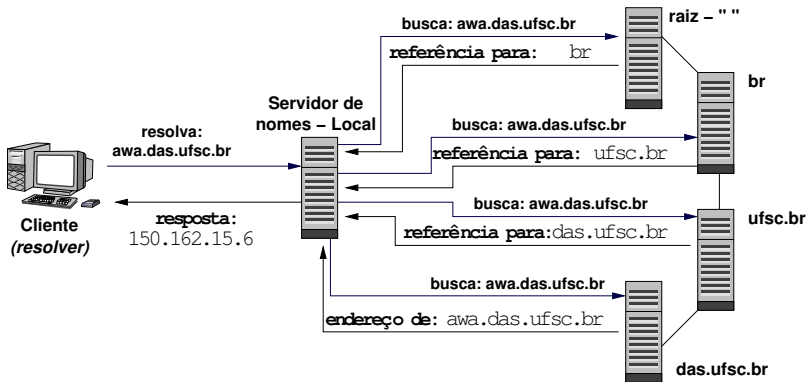
Delegação

- Domínios podem ser divididos em subdomínios
- Administração descentralizada
 - Deixar a responsabilidade da administração de um subdomínio para outra organização
 - Subdomínios podem ser divididos em mais subdomínios
- Domínios superiores mantêm somente ponteiros para a raiz dos subdomínios, delegando as consultas a eles

Servidores de Nomes e Zonas

- Servidores de Nomes (*Name Servers*)
 - aplicativos que armazenam informações sobre um espaço de nomes de domínio (uma Zona)
 - informações oriundas de um arquivo ou de um outro NS
 - NS são responsáveis somente pela sua Zona
 - Um NS pode ser responsável por múltiplas Zonas
- Zona (Zone)
 - Domínios, subdomínios que foram delegados são tratados como Zonas
 - Uma Zona e um Domínio podem compartilhar um mesmo nome de domínio, mas contém nós diferentes

Resolução



Resolução

Tipos de consultas

- Recursiva
 - O cliente envia uma “consulta recursiva” ao servidor de nomes
 - Questiona um servidor de nomes remoto e segue as referências, até receber a resposta
 - O servidor não pode informar ao cliente apenas uma referência para um outro servidor de nomes
 - É possível configurar servidores para ignorar/recusar consultas recursivas
- Iterativa
 - Menos trabalhoso para o servidor que foi consultado
 - Verifica em sua base local (cache, etc.) se encontra a resposta
 - Caso não, encontra os servidores que estão mais próximos do domínio pesquisado
 - Retorna a referência do servidor para quem originou a consulta
 - O cliente que escolhe qual servidor irá consultar

Endereços para nomes

- Nomes para endereços – é mais fácil para humanos
- Endereços para nomes ?
 - Máquinas trabalham melhor com números
 - Arquivos de registro (*logs*) armazenam números
- Dado um nome, encontrar um endereço é simples
- Dado um endereço, encontrar um nome não é tão simples
- Para isso criou-se o DNS Reverso
- Muitos administradores esquecem de configurar isso !

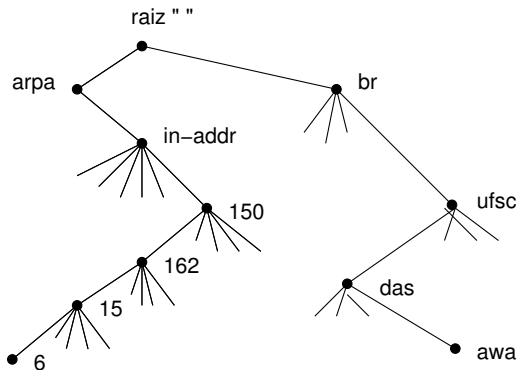
Endereços para nomes

- Nomes para endereços – é mais fácil para humanos
- Endereços para nomes ?
 - Máquinas trabalham melhor com números
 - Arquivos de registro (*logs*) armazenam números
- Dado um nome, encontrar um endereço é simples
- Dado um endereço, encontrar um nome não é tão simples
- Para isso criou-se o DNS Reverso
- Muitos administradores esquecem de configurar isso !

Endereços para nomes (cont.)

- Criam-se nomes de domínio que usam endereços como rótulos
- Trata-se do domínio: `in-addr.arpa`
 - Até 256 subdomínios – primeiro octeto do endereço IP
 - Cada subdomínio pode ter até 256 domínios, relacionados ao segundo octeto
 - No quarto nível estão associados os nomes completos
- Diferença entre endereços e nomes
 - Endereços IP são mais específicos da esquerda para a direita
 - Nomes são mais específicos da direita para a esquerda

Endereços para nomes (cont.)



6.15.162.150.in-addr.arpa
awa.das.ufsc.br

awa.das.ufsc.br
150.162.15.6

Resolução - Raiz

- Atualmente é formado por 13 entidades
 - Não confundir como somente 13 servidores
 - `a.root-servers.net`, ..., `m.root-servers.net`
 - Nem toda consulta passa obrigatoriamente pela raiz
- A Raiz sabe somente para quem deverá ser encaminhada a próxima consulta
 - `.com` → lista de servidores; `.br` → lista de servidores, etc.
- Ponto forte do sistema é a “diversidade”
 - estrutura organizacional
 - *hardware* e *software* utilizado

Tipos de Servidores de Nomes

- Primário
 - Servidor responsável pela Zona, o qual mantém todas as informações sobre as máquinas
 - Carrega as informações a partir de um arquivo
 - Geralmente denominado como: Mestre (*master*)
- Secundário
 - Obtém as informações sobre a Zona do servidor mestre responsável pela Zona – não obrigatório
 - Transferência de zona
 - As vezes chamado de Escravo (*slave*)

Tanto o primário quanto o secundário são responsáveis pela Zona.

Base de dados do DNS

- Não armazena somente uma lista associando IP e nomes (como o /etc/hosts)
- Base formada por Registro de Recurso (*Resource Record* - RR)
 - Inicia na primeira coluna, obrigatoriamente
- Cada registro possui um tipo associado e uma classe
- Classes estão relacionadas a um tipo de rede ou de software
 - IN para endereçamento TCP/IP
 - HS para endereçamento Hesiod (utilizado pelo Kerberos)
 - entre outras. . .

RR – Alguns tipos principais

SOA	Autoridade para os dados deste domínio
NS	Servidor de nomes para este domínio
A	Associação de um nome para endereços
CNAME	Nomes canônicos (apelidos – <i>alias</i>)
MX	Mail Exchanger
PTR	Mapeamento de endereços para nomes (reverso)

Servidor – Bind 9.3

Arquivos de configuração

- `/etc/named.conf` – Arquivo de configuração
- `named.ca` – Referência para as raízes (cache)
- `named.local` – Informações sobre o domínio local

/etc/named.conf

```
options{
    directory "/var/named"; };
zone "."{
    type hint;
    file "named.ca";
};
zone "0.0.127.in-addr.arpa"{
    type master;
    file "named.local";
};
// Nosso dominio
zone "awa.com.br"{
    type master;
    file "awa.zone";
};
//Configurando reverso
zone "15.162.150.ind-addr.arpa"{ //nossa rede
    type master;
    file "awa.rev";
};
```


Arquivo: named.ca

- Arquivo da InterNIC contendo as 13 raízes
- Pode ser obtido através:
 - `ftp://ftp.rs.internic.net/domain`
 - `dig @A.ROOT-SERVERS.NET . ns`

```
; formerly NS.INTERNIC.NET
.                3600000   IN   NS     A.ROOT-SERVERS.NET.
A.ROOT-SERVERS.NET. 3600000   A    198.41.0.4
;
; formerly NS1.ISI.EDU
.                3600000   NS   B.ROOT-SERVERS.NET.
B.ROOT-SERVERS.NET. 3600000   A    192.228.79.201
;
; formerly C.PSI.NET
.                3600000   NS   C.ROOT-SERVERS.NET.
C.ROOT-SERVERS.NET. 3600000   A    192.33.4.12
```

Arquivo: named.local

```
$TTL 1d          ; Tempo pelo qual as informações serão
                  ; consideradas válidas para outros servidores
@   IN   SOA     localhost.  admin.localhost.  (
1997022700      ; Serial -- deve ser mudado a cada alteração no arquivo
28800           ; Refresh -- tempo para o servidor secundário fazer
                  ; consulta ao primário
14400           ; Retry -- tempo para tentativa, caso o primário esteja
                  ; indisponível
3600000         ; Expire -- se o primário não for consultado neste intervalo
                  ; as informações sobre o domínios no secundário serão descartada
86400 )         ; Minimum -- Valor mínimo para o TTL usado pelos registros
                  ; que não possuem valores TTL explícitos
IN      NS      localhost.
1 IN    PTR     localhost.
```

Arquivo: awa.zone

```
@      IN      SOA      guaja.awa.com.br. admin.awa.com.br(
                          2005010100 360000 3600 3600000 360000 )
; servidores de nomes (O nome @ está implícito)
      IN      NS      localhost.
      IN      NS      variant.awa.com.br.
; servidor de e-mail
      IN      MX      10  mail.awa.com.br.
      IN      MX      20  guaja.awa.com.br.
; definição das máquinas
guaja  IN      A       150.162.15.2
variant IN     A       150.162.15.3
mail   IN      A       150.162.15.4
; definição dos apelidos
www    IN      CNAME   guaja.awa.com.br.
ftp    IN      CNAME   guaja.awa.com.br.
```

Arquivo: awa.zone

```
$ORIGIN 15.162.150.in-addr.arpa.  
@      IN   SOA   guaja.awa.com.br. admin.awa.com.br(  
                2005010100 360000 3600 3600000 360000 )  
; servidores de nomes (O nome @ está implícito)  
        IN   NS   localhost.  
        IN   NS   variant.awa.com.br.  
; endereços para nomes canônicos  
2      IN   PTR   guaja.awa.com.br.  
3      IN   PTR   variant.awa.com.br.  
4      IN   PTR   mail.awa.com.br.
```

Resolvendo nomes

- A resolução é feita através de uma biblioteca, a qual contém uma coleção de funções
 - Não é uma aplicação especial
 - `gethostbyname()` e `gethostbyaddr()` são funções principais
- Quando invocadas, verificam arquivos de configuração para saber onde buscar os nomes
 - Versões antigas do GNU/Linux → `/etc/host.conf`
 - Versões que usam Glibc versão 2 → `/etc/nsswitch.conf`

Arquivo: /etc/host.conf

```
## Determina a ordem de busca
## opcoes aceitas: bind --> para NS,
## hosts --> /etc/hosts e nis --> para NIS
order hosts, bind
## Determina se em /etc/hosts pode haver
## maquina associada com varios IP
multin on
## Protecao contra Spoof - confronta o nome e IP
nospoof on
## Gera registro no syslog para tentativas de spoof
alert on
```

Arquivo: /etc/nsswitch.conf

- O conceito de serviço de nomes foi estendido, suportando diferentes bases de dados
- A base de dados associada com a resolução de endereços: *hosts*
- A base *networks* esta relacionada aos nomes de redes

```
## Definindo a ordem  
hosts:      files dns  
networks:   files
```

Arquivo: /etc/resolv.conf

```
## domain ou search  
## search dominio1.com dominio2.com  
domain ap.com.br  
nameserver 150.162.15.1  
nameserver 150.162.15.10  
nameserver 150.162.15.100
```


Sumário

- 1 Serviço de nomes
- 2 FTP – File Transfer Protocol**
- 3 Servidor WEB
- 4 Serviço de Email
- 5 Administração remota
- 6 Inetd e Xinetd

Protocolo para transferência de arquivos

- Proposto inicialmente em 1971, pela RFC 114
- Definido em 1985 pela RFC 959
 - Promover o compartilhamento de arquivos
 - Encorajar, direta ou indiretamente, o uso de computadores remotos
 - Abstrair, para os usuários, os diferentes tipos de armazenamento de arquivos
 - Permitir a transferência de dados de forma confiável e eficiente

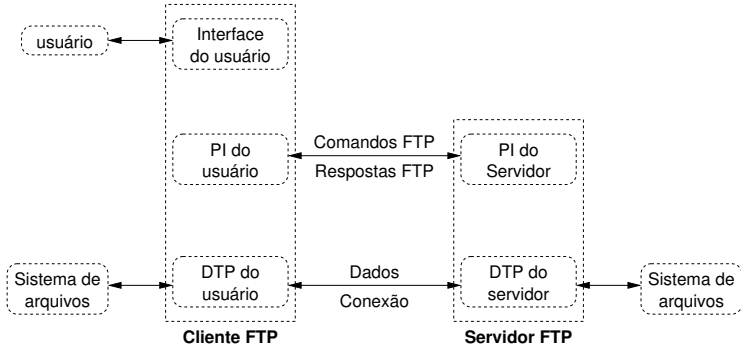
Modelo

- A conexão inicial de um cliente para um servidor FTP
 - Denominada: conexão de controle
 - Parte de uma porta aleatória para a porta conhecida do servidor (porta 21)
 - O protocolo TCP utiliza ambas extremidades para identificar uma conexão
- Criação de uma nova conexão para transferência dos dados
 - Usa a conexão de controle para coordenar as portas que serão utilizadas
 - O cliente atribui de forma aleatória uma porta local que não esteja em uso
 - e usa a mesma para conectar o processo de transferência de dados na porta 20

Modelo (cont.)

- Utiliza duas conexões diferentes
 - conexão de controle – para os comandos FTP
 - conexão de dados – para a transferência de dados
- Um cliente FTP é formado por:
 - Interface do usuário
 - Interpretador de protocolo (PI)
 - Processo de transferência de dados (DTP)
- Um servidor FTP é formado por:
 - Interpretador de protocolo (PI)
 - Processo de transferência de dados (DTP)

Modelo (cont.)



Relações entre o TELNET e o FTP

- O FTP usa o protocolo virtual de rede (NVT) do TELNET na conexão de controle
 - Os PIs do cliente e do servidor, podem implementar as regras do Telnet diretamente em suas próprias funções
 - FTP usa bem pouco do TELNET, o que resultaria em pouco código a ser implementado
 - ou, fazer de um “módulo TELNET” presente no sistema
 - Programação modular, compartilhamento de código – facilidade

Modos de transferência de arquivos

- Modo texto: ASCII
 - Utilizado para transferência de arquivos textos
 - Diferentes arquiteturas, tratam de forma diferente os arquivos texto
 - Ex: DOS: CR/LF, Unix: LF
 - O aplicativo que está enviando, obrigatoriamente converte para o padrão NVT-ASCII
 - Os caracteres são tratados como dados de 8 bits
- Modo binário: BINARY
 - Utilizado para transferência de arquivos binários

Formas de transmissão dos dados

Através da conexão de controle estabelece a forma como os dados serão transferidos

- Quando o cliente solicita uma conexão de dados (transferir arquivo, listar diretório, etc. . .)
 - Ativa
 - O cliente “ouve” em uma porta não privilegiada uma conexão originada pelo servidor, oriunda da porta 20
 - O servidor que inicia a conexão de dados
 - Passiva
 - O cliente solicita ao servidor para “ouvir” em uma porta não privilegiada
 - O cliente, através de uma porta não privilegiada, conecta na porta definida anteriormente
 - É o cliente quem inicia a conexão de dados

Algumas implementações

- wu-ftpd
 - Desenvolvida na Universidade de Washington
 - Era padrão na maioria das distribuições GNU/Linux
 - Caiu em desuso: complexo e falhas de segurança
- ProFTPd
 - Está presente na maioria dos GNU/Linux
 - Configuração simples
- vsFTPd
 - Juntamente com o proftpd, vem substituindo o wu-ftpd
 - Promessa de ser uns dos mais seguros e rápidos
- MS IIS
 - Além de servidor HTTP, o IIS traz FTP, NNTP, etc. . .

Trivial FTP - TFTP

- Feito para aplicações que não necessitam de interações complexas entre clientes/servidores
- Não fornece autenticação, se preocupa simplesmente com a transferência de arquivos
- Tamanho reduzido – ideal para aplicações embarcadas
- Diferente do FTP, o TFTP usa o UDP como transporte, porta 69

TFTP – Processo de transferência

1 Estabelecimento da conexão

- No primeiro pacote enviado pelo cliente, é informado o nome do arquivo e se o mesmo será escrito ou lido

2 Transferência dos dados

- São trocadas mensagens TFTP, onde uma parte envia dados e a outra envia confirmações de recebimento

3 Encerramento da Conexão

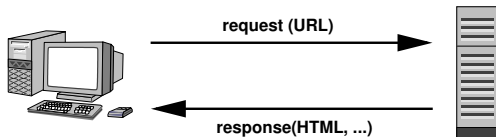
- Após enviar a última mensagem contendo dados e sua confirmação, a conexão é encerrada
- É possível enviar mensagens erros – essas mensagens encerram a conexão

Sumário

- 1 Serviço de nomes
- 2 FTP – File Transfer Protocol
- 3 Servidor WEB**
- 4 Serviço de Email
- 5 Administração remota
- 6 Inetd e Xinetd

Hypertext Transfer Protocol – HTTP

- Protocolo para transferência de hipertexto
- Protocolo pedido/resposta
- Define a interação entre clientes e servidores
- Está por trás da comunicação WWW
 - Linguagem HTML



Servidor HTTP

- Por padrão ouve na porta 80
- Mapeia os caminhos da URL no sistema de arquivos local
 - Caminho: “/”
 - `/var/www/htdocs`
 - Caminho: “/imagens/foto.jpg”
 - `/var/www/htdocs/imagens/foto.jpg`

Pedido HTTP

- Dada uma URL: `http://www.awa.com.br/`

GET / HTTP/1.1

Host: `www.awa.com.br`

User-Agent: `Mozilla/5.0 (X11; U; Linux i686; pt-BR;
rv:1.7.5) Gecko/20041118 Firefox/1.0`

Accept: `text/xml,application/xml,application/xhtml+xml,
text/html;q=0.9,text/plain;q=0.8,image/png,*/*;q=0`

Accept-Language: `pt-br,en-us;q=0.7,en;q=0.3`

Accept-Encoding: `gzip,deflate`

Accept-Charset: `ISO-8859-1,utf-8;q=0.7,*;q=0.7`

Keep-Alive: `300`

Connection: `keep-alive`

Resposta HTTP

```
HTTP/1.1 200 OK
Date: Thu, 11 Feb 2005 12:44:25 GMT
Server: Apache
Last-Modified: Thu, 24 Oct 2002 14:49:07 GMT
ETag: "e6a158be17f57c48f6e5534baa8cc8df2a8a2c5f"
Accept-Ranges: bytes
Content-Length: 7282
Keep-Alive: timeout=2400, max=3600
Connection: Keep-Alive
Content-Type: text/html

<HTML>.....
```


Alguns códigos HTTP

- 100 - 199 – Informativo
 - 100 Continue – pedido particionado
- 200 - 299 – Pedido do cliente executado com sucesso
 - 200 OK – pedido com sucesso
- 300 - 399 – Pedido do cliente redirecionado
 - 301 Moved Permanently
- 400 - 499 – Pedido do cliente incompleto
 - 404 Not Found – documento não existe
- 500 - 599 – Erros do servidor
 - 502 Bad gateway – servidor (ou proxy) obteve respostas inválidas de outros servidor (ou proxy)

HTTP – Implementações

- Clientes
 - NCSA Mosaic
 - Netscape Navigator
 - MS Internet Explorer
 - Mozilla (oriundo do Netscape)
 - Firefox, Safari, ...
- Servidores
 - NCSA
 - Apache
 - Netscape
 - SUN WS
 - MS IIS, MS PWS
- Estatísticas
 - http://news.netcraft.com/archives/web_server_survey.html

Servidor HTTP: Apache (cont.)

- Iniciado em 1995 por um grupo de *webmaster*
 - Com o intuito de criar uma implementação HTTP robusta e sem custos para a comunidade
- Originalmente surgiu como uma série de *patches* para o servidor NCSA
- Apache Server é derivado de: “A PAtCHy sErver”

Servidor HTTP: Apache

Duas versões principais

- Melhorias 2.0 com relação ao 1.3
 - Mais escalável
 - Reescrita a API para os módulos
 - Suporte ao IPv6
 - Melhor suporte para plataformas não UNIX
 - Configuração simplificada

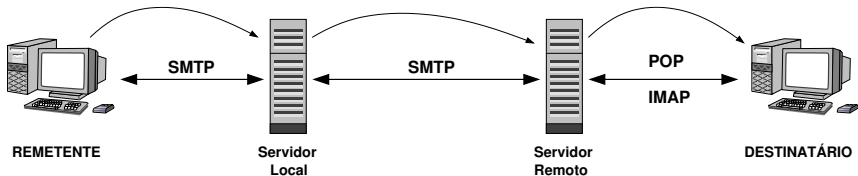
Sumário

- 1 Serviço de nomes
- 2 FTP – File Transfer Protocol
- 3 Servidor WEB
- 4 Serviço de Email**
- 5 Administração remota
- 6 Inetd e Xinetd

Serviço de Email

- **MTA** – *Mail Transfer Agent*
 - Localizado no lado do “servidor”
 - Utiliza o protocolo SMTP para comunicar com outros servidores
 - Sendmail, Postfix, Exim, Qmail
- **MUA** – *Mail User Agent*
 - Localizado no lado do “cliente”
 - Utiliza os protocolos:
 - POP e IMAP para recebimento de e-mails
 - SMTP para envio de e-mails
 - Mutt, Thunderbird, Pine, MS Outlook Express

Envio e recebimento de e-mail



- 1 MUA contacta o MTA local
- 2 MTA local contacta o MTA remoto
- 3 MTA Remoto armazena o email na caixa do usuário
- 4 Destinatário obtém o email via POP ou IMAP

SMTP – *Simple Mail Transfer Protocol*

- Utiliza o protocolo TCP – porta 25
- E-mails ficam armazenados em uma fila no servidor local, antes de serem encaminhados
- Interação em cliente/servidor através de comandos ASCII
- Fases da comunicação
 - Apresentação
 - Transferência das mensagens
 - Encerramento
- Utiliza os registros **MX** do serviço de nomes

Formato da mensagem

- Cabeçalho do e-mail
 - From:
 - To:
 - Subject:
- Corpo do e-mail
 - A mensagem propriamente dita
 - Somente caracteres ASCII

MIME – Multimedia mail extensions

- Permite a transmissão de dados não-ASCII
 - Permite que dados sejam codificados em cadeias ASCII
- Mensagens MIME informam o tipo dos dados e a codificação utilizada

```
From: jlb@awa.com.br  
To: pib@awa.com.br  
Subject: Planos futuros  
MIME-Version: 1.0  
Content-Transfer-Encoding: base64  
Content-Type: image/jpeg; name="estrutura.jpg"  
base64 encoded data.....
```

Sumário

- 1 Serviço de nomes
- 2 FTP – File Transfer Protocol
- 3 Servidor WEB
- 4 Serviço de Email
- 5 Administração remota**
- 6 Inetd e Xinetd

Telnet

- Um protocolo para terminal remoto
- Oferece três serviços básicos
 - Terminal Virtual de Rede (*NVT*)
 - interface padrão para sistema remotos
 - Mecanismo para negociação de opções
 - Tipo de caracteres ASCII (de 7 ou 8 bits)
 - Tratamento simétrico de ambas partes
 - Não é obrigatório que a entrada de dados seja via teclado
 - Não é obrigatório que a saída de dados seja na tela
 - Qualquer lado pode negociar opções

SSH – *Secure Shell*

- Protocolo de camada de transporte
 - Provê autenticação do servidor, confidencialidade e integridade dos dados
 - Provê compressão dos dados
- Protocolo de autenticação de usuários
 - Provê autenticação do usuário no lado do cliente
- Protocolo de conexão
 - Provê túneis cifrados
 - Encaminhamento de portas e conexões X11
- Versão atual: SSH2 – Principal implementação OpenSSH

Sumário

- 1 Serviço de nomes
- 2 FTP – File Transfer Protocol
- 3 Servidor WEB
- 4 Serviço de Email
- 5 Administração remota
- 6 Inetd e Xinetd**

Particularidades. . .

- **Internet Daemon – Inetd**

- Possibilita que os “daemon” sejam carregados somente quando necessário
- Fica ouvindo as portas dos serviços providos
- Arquivo de configuração `/etc/inetd.conf`
- Segurança fica a cargo do *TCP-Wrappers*

- **eXtended Internet Daemon – Xinetd**

- Limitação de conexões
- Controle de acesso mais granular
- Redirecionamento de conexões
- Configurações em `/etc/xinetd.d/`

Inetd

/etc/inetd.conf

#service	type	proto		user	program to launch
telnet	stream	tcp	nowait	root	/usr/sbin/tcpd in.telnetd
ftp	stream	tcp	nowait	root	/usr/sbin/tcpd in.ftpd
talk	dgram	udp	wait	root	/usr/sbin/tcpd in.talkd

/etc/hosts.allow

```
in.telnetd      : ALL
in.ftpd         : .awa.com.br, plat.guaja.com.br
```

/etc/hosts.deny

```
ALL      : ALL EXCEPT LOCAL
```


Xinetd

```
/etc/xinetd.d/ftp
```

```
service ftp{
    disable           = no
    socket_type       = stream
    wait              = no
    nice              = 10
    user              = root
    server            = /usr/sbin/in.ftpd
    server_args       = -l
    instances         = 4
    access_times      = 2:00-9:00 12:00-24:00
    log_type          = SYSLOG authpriv
    log_on_success    = HOST PID
    log_on_failure    = HOST
    only_from         = 10.1.1.0
}
```