DESENVOLVIMENTO E IMPLEMENTAÇÃO DE UM PROXY DNS EM UMA REDE HETEROGÊNEA

Lucio H. Franco[1], Ulisses T. V. Guedes[2], Antonio Montes[1], Benício Carvalho[1]

{lucio,montes,benicio}@lac.inpe.br[1] - ulisses@dem.inpe.br[2]

Laboratório Associado de Computação e Matemática Aplicada[1]

Mecânica Espacial e Divisão de Controle[2]

Instituto Nacional de Pesquisas Espaciais

Av. dos Astronautas, 1758 – 122270-010 – São José dos Campos, SP - Brasil

Roteiro

- Introdução
- Arquitetura, Protocolo e Taxonomia do DNS
- DNS em Redes Heterogêneas
- Implementação do Sistema de Proxy DNS
- Implementação do Patch
- Funcionamento do Sistema Implementado
- Testes, Resultados e Monitoração do sistema
- Trabalhos Futuros
- Considerações Finais
- Referências

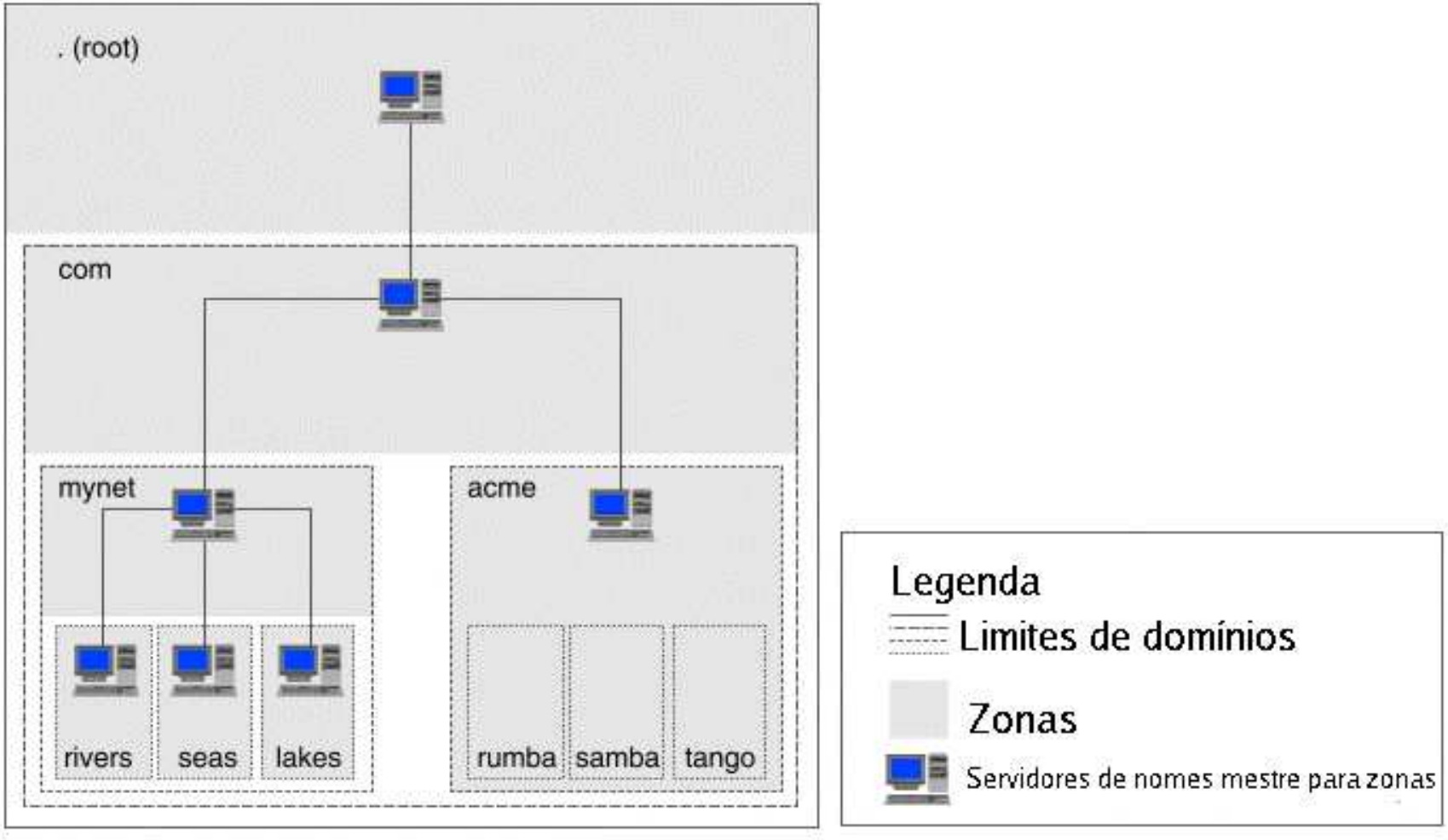
Introdução

- A resolução de nomes na Internet é feita pelo sistema de nome de domínio (DNS)
- Cada *site* (departamento de uma universidade, empresa, etc) mantém somente sua própria base de informações
- Primeira implementação recebeu o nome de BIND (*Berkley Internet Name Domain*)
- Serviços semelhantes ao BIND também estão disponíveis em outros protocolos (ex: WINS)

Introdução

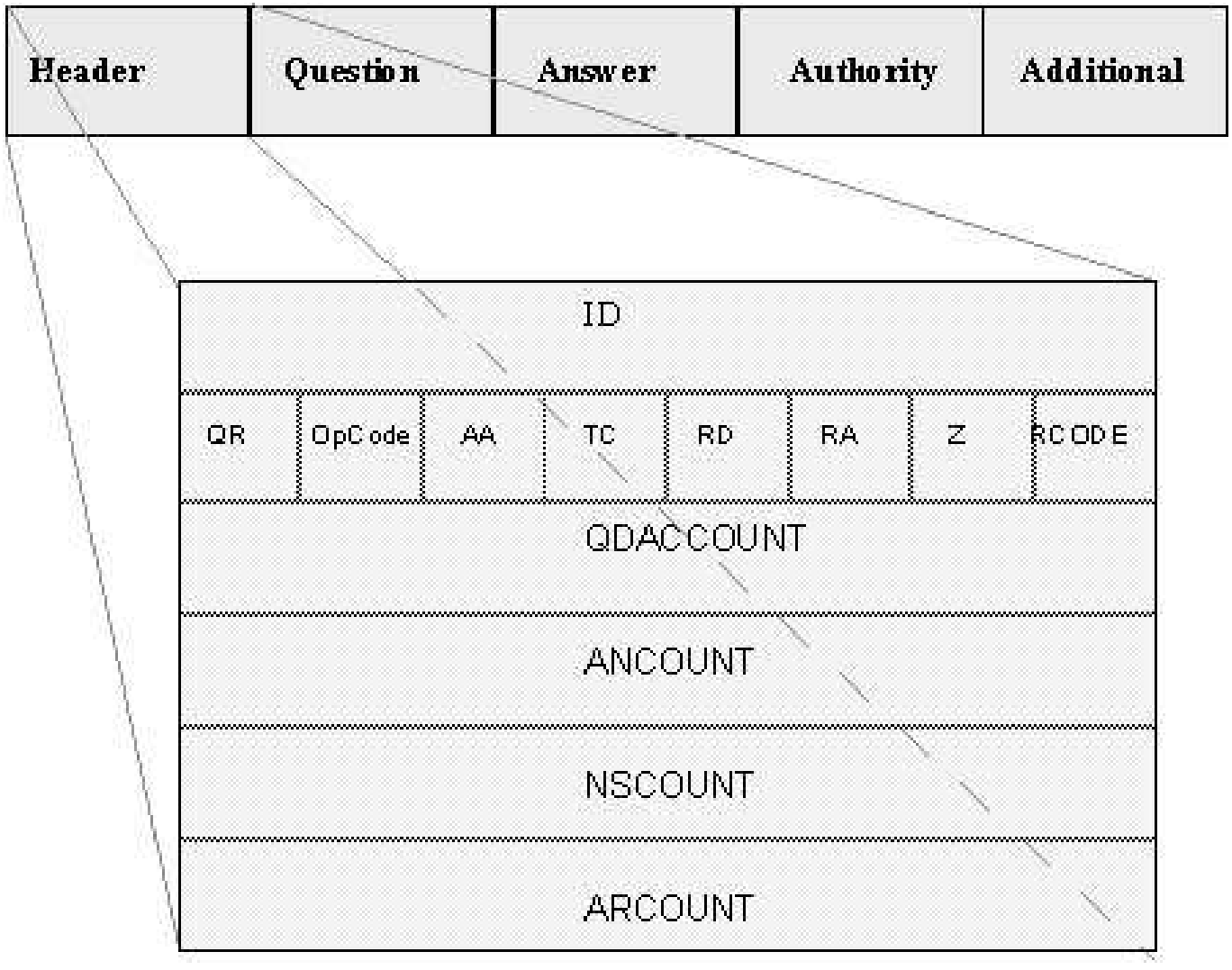
- Há duas formas distintas para a resolução de nomes:
  - A aplicação tem o nome e pergunta ao DNS o endereço *IP*
  - A aplicação tem o endereço *IP* e deseja conhecer o nome correspondente

Arquitetura do DNS



Hierarquia das zonas e domínios

O Protocolo DNS



Protocolo DNS

Taxonomia dos DNSs

- As RFCs 1034 e 1035 dão uma idéia que todos os DNSs são fundamentalmente iguais, porém, há várias funções distintas que eles podem desempenhar
- DNS devem ser classificados dentro de duas classes: aplicações servidoras de conteúdo e *proxies* DNS

Aplicações Servidoras de Conteúdo

- Disponibilizam o conteúdo do DNS para toda a Internet
- Os dados são lidos de uma base de dados ou são gerados internamente
- Aplicações servidoras de conteúdo proveêm uma referência para o próximo nó da árvore de DNS e não uma resolução completa de um nome

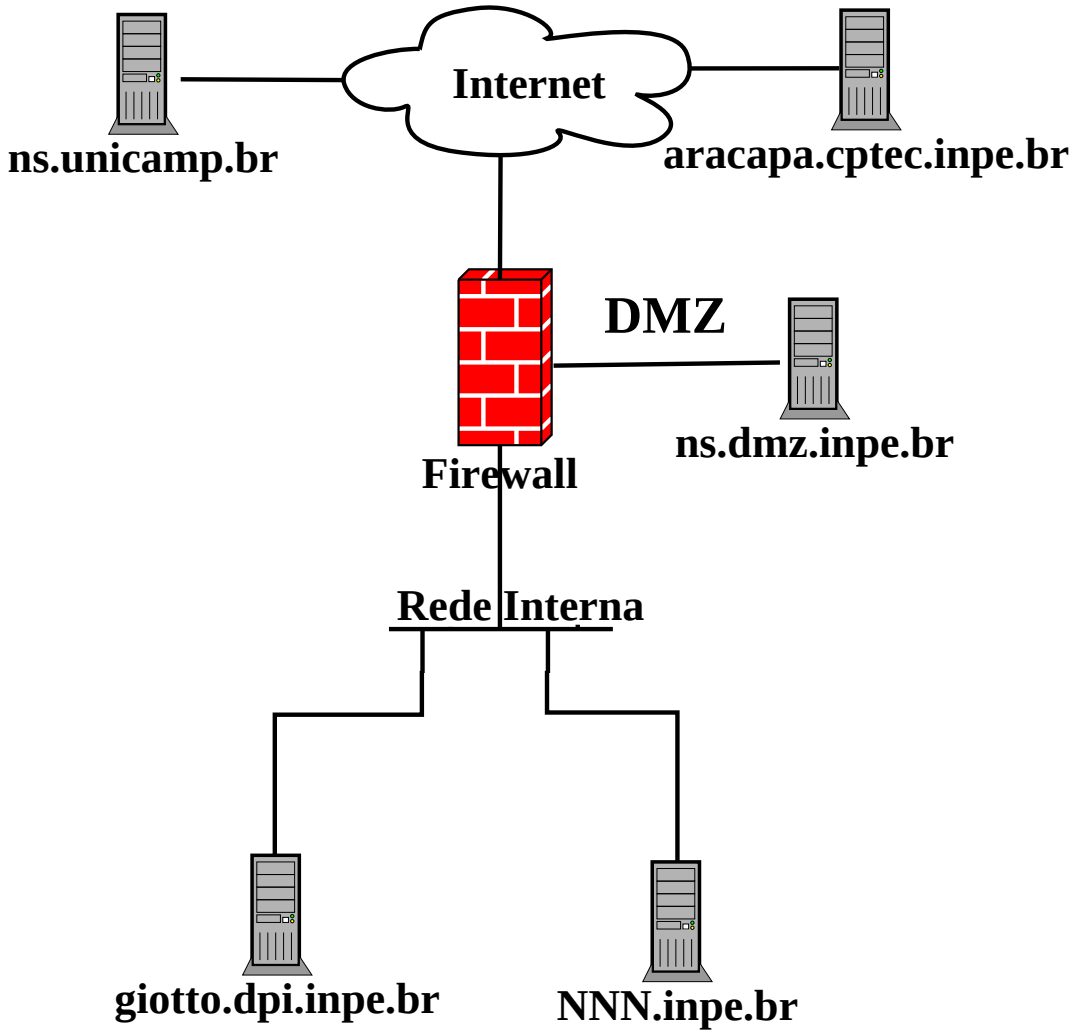
Proxy DNS

- Atuam como intermediários entre aplicações clientes e aplicações servidoras de resolução de nomes
- Geralmente fazem cache dos dados recebidos, reduzindo o tráfego e a latência dos dados freqüêntemente requisitados
- São categorizados em: aplicações *proxies* de resolução e aplicações *forwarding proxy*

DNS em Redes Heterogêneas - Cenário

- Rede INPE
  - Classe B: 150.163.0.0/16
  - 1 servidor de nomes mestre em `giotto.dpi.inpe.br`
  - 7 servidores de nomes escravos, sendo uma deles fora dessa rede
  - 45 domínios em 20 servidores de nomes departamentais
  - Uma arquitetura de DNS hierarquica com delegações para os subdomínios
  - BIND é a implementação predominante

DNS em Redes Heterogêneas - Cenário



Arquitetura de DNS da Rede INPE

Implementação do Sistema de Proxy DNS

Vantagens	**Desvantagens**
Segurança (menos máquina, sem acesso a tabelas "locais")	Não elimina os DNSs departamentais
Simples. Fácil gerenciamento e controle do tráfego	Requer alta disponibilidade
DNSs departamentais não mais acessíveis pela Internet	Escolha dos root servers

Testes com BIND

- Inicialmente, optou-se utilizar aplicações de código aberto, conhecidas e de ampla utilização como o *BIND*
- Utilizando uma versão atualizada e sem vulnerabilidades reportadas até o momento

Testes com BIND

- Configuração, para atuar como *proxy*, é feita através das opções de configuração `forward` e `forwarders`, abaixo um exemplo:
  - `allow_recursion { redeinpe; };`
  - `forward only;`
  - `forwarders { giotto; fractal; lagavullin; patroa; sputnik; };`
  - `forward` (ONLY | FIRST)

Testes com BIND

- 2 *Views*:
- `public`:
  - Tabelas do domínio DMZ.INPE.BR (direta e reversa)
- `private`:
  - Zonas do domínio INPE.BR
  - `type forward;`
  - `forward only;`
  - `forwarders { giotto; fractal; lagavullin; patroa; sputnik; };`

Testes com BIND - Resultados

- Implementação operacional, porém, sem atender a todas as funcionalidades esperadas
- Somente conseguiu responder consultas com bit de recursividade ativo
- "Não" oferecia outra forma de configuração
- Possibilidade de implementação do nosso próprio *proxy* DNS

Testes com DJBDNS

- Outra implementação focada na segurança, desempenho e robustez
- Processos específicos e distintos. Destacam-se:

  Tinydns: contêm as tabelas de nomes, aplicação servidora de conteúdo

  Dnscache: resolvedor para os clientes DNS, permite o controle por *hosts* e redes cadastrados

Testes com DJBDNS

- *Proxy* habilitado através da cláusula `FORWARDONLY` no diretório de configuração do `dnscache`
- O sistema apresentou o mesmo comportamento do *BIND*
- Opção do desenvolvimento de um *patch* para o BIND e/ou para o DJBDNS para verificação do bit de autoridade das respostas

Implementação do Patch

BIND	**DJBDNS**
Problemas não críticos de segurança	US$500 para quem reportar uma vulnerabilidade da última versão disponível
Muitas linhas de código - difícil entendimento	Um arquivo para cada função e/ou processo do sistema
Ausência de documentação e comentários no código	Ausência de documentação e comentários no código - Uso de mnemônicos

Implementação do Patch para DJBDNS

- Possibilidade de ativar e desativar o funcionamento do novo comportamento do sistema através da variável `TOTALAUTH`
- Desenvolvimento inicial do sistema focado no protocolo *UDP*
- Sem cache local. Váriavel `CACHESIZE` igual a 0
- Catalogados todos os nomes e endereços *IP* os quais a rede INPE responde como autoridade de DNS

Implementação do Patch para DJBDNS

```
#define MAXUDP 200
static struct udpclient {
  struct query q;
  struct taia start;
  uint64 active; /* query number */
  iopause_fd *io;
  char ip[4];
  uint16 port;
  char id[2];
  int auth; //authoritative answer
} u[MAXUDP];
```

Funcionamento do Sistema Implementado

- O sistema com o *patch* do dnscache trabalha como um *proxy* DNS puramente. Intermediando comunicações entre requisições de consultas de DNS e DNS Internos autoridades de zonas.

Arquitetura do Proxy DNS na Rede INPE

Funcionamento do Sistema Implementado

```
if (totalauth) {
    x->auth=0;
    treatpacket=treat_packet2(buf+13, len-13);
    if (treatpacket == 0) //ok
            x->auth=1;

    if (x->auth == 0 && !okclient(x->ip))
            return;
  } else {
    if (!okclient(x->ip)) return; //allow
  }
//dnscache does only recursive query - checking bit
buf[2] |= 1;
```

Verifi cação de uma consulta de DNS

Funcionamento do Sistema Implementado

```
if(totalauth) {
    if (u[j].auth == 1) {
        response[2] |= 4;
    }
}
...
u[j].auth = 0;
```

Verifi cação da resposta da consulta

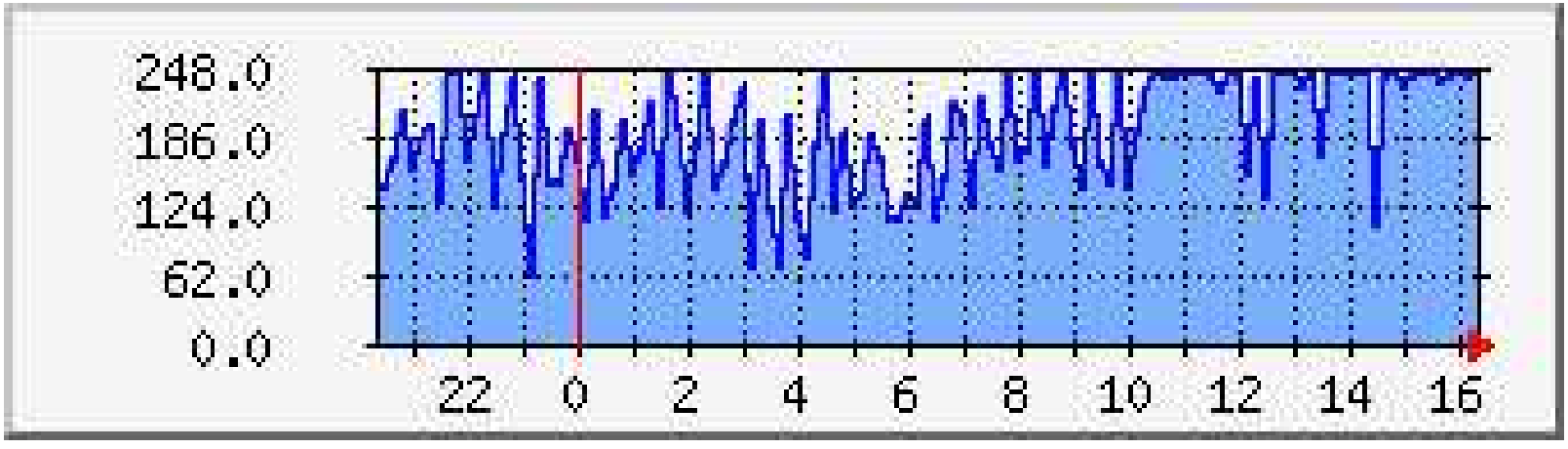
Testes e Resultados

- Migração, para o uso do sistema, de forma progressiva com redirecionamento no *firewall*
- `Dnscache` consegue tratar 200 requisições do protocolo *UDP* e 20 conexões do protocolo *TCP* simultâneamente. Utilizou-se um número maior para se efetuar os primeiros testes
- Depois do monitoramento foi possível a migração total do acesso externo de DNS para este sistema

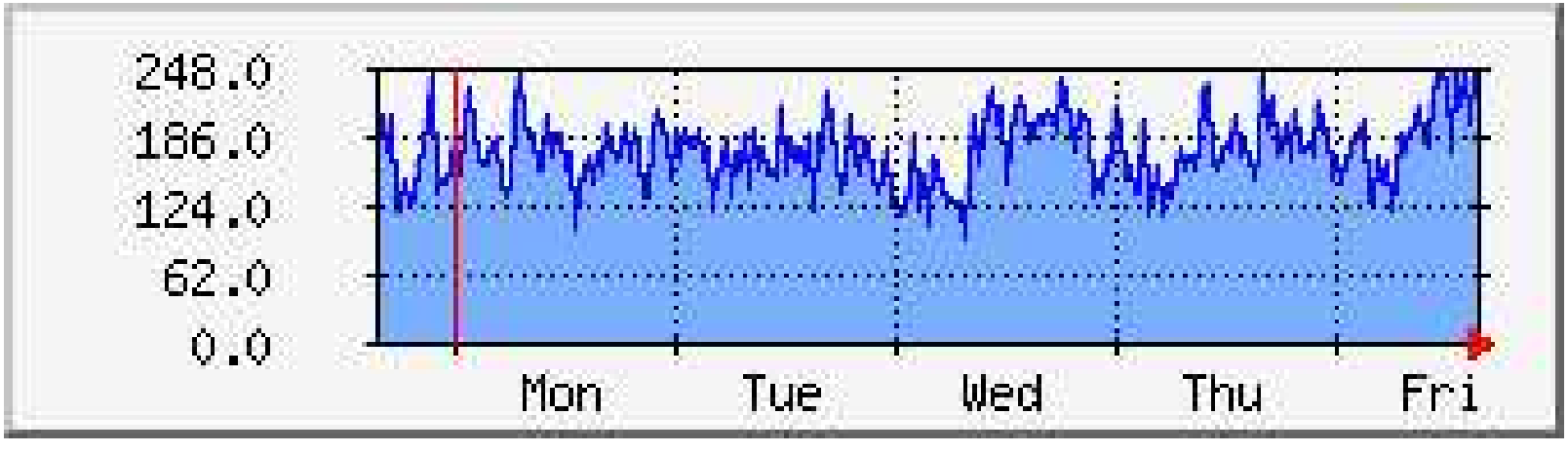
Testes e Resultados

- Em produção sem alterações e interrupções do serviço desde julho de 2003
- Há sistemas redundantes caso hajam falhas de hardware e/ou software
- Foi possível descobrir que haviam DNSs dentro da rede INPE prestando serviço para outras instituições

Monitoração e Gráfi cos



Número de Requisições x Tempo (Horas)



Número de Requisições x Tempo (Dias)

Dados referentes ao dia 24/10/2003

Monitoração e Gráfi cos

Segundos	Qtidade	Porcentagem
0.00	27603	72.74%
0.01	21858	4.90%
0.02	251	0.66%
>10	6571	-

Dados referentes ao dia 24/10/2003 às 14:32

Trabalhos Futuros

- Cadastros dos domínios através de arquivo de configuração
- A comparação de *strings* utilizando *hash*
- Testes com o *patch* para o protocolo *TCP*
- Criar uma arquitetura com caches centralizados dentro da rede interna na tentativa de diminuir os acessos externos das diversas aplicações servidoras de nomes

Considerações Finais

- Maior segurança para as aplicações servidoras de nomes da rede INPE, não quebrando o conceito de implementação de sistemas DNS e nem causando atrados das consultas
- Entendimento das várias formas de implementação e configurações de DNS, buscando adaptação deles a uma arquitetura de rede heterogênea

Referências

- TCP/IP Illustrated Volume 1 - W. Richard Stevens. Addison Wesley - 2001. ISBN 0-201-63346-9
- RFC 1034
  `ftp://ftp.rfc-editor.org/in-notes/rfc1034.txt`
- RFC 1035
  `ftp://ftp.rfc-editor.org/in-notes/rfc1035.txt`
- BIND
  `http://www.isc.org`
  `http://www.cert.org/archive/pdf/dns.pdf`
- DJBDNS
  `http://cr.yp.to/djbdns.html`
  `http://homepages.tesco.net/~J.deBoynePollard/FGA`