

Capítulo 9

Aplicações

Problema

- Aplicações precisam de seus próprios protocolos.
- Essas aplicações são parte protocolo de rede (no sentido que elas trocam mensagens com seus pares em outras máquinas) e parte um programa de aplicação tradicional (no sentido que elas interagem com o sistema de janelas, o sistema de arquivos, e também com o usuário).
- Este capítulo explora algumas das aplicações mais populares da atualidade.

Visão Geral do Capítulo

- Aplicações Tradicionais
- Aplicações Multimídia
- Serviços de Infraestrutura
- Redes Sobrepostas
- O que veremos: serviços de infraestrutura e aplicações tradicionais

Aplicações Tradicionais

- As originais e mais populares
 - SMTP (Email)
 - FTP
 - Telnet
 - The World Wide Web

Serviços de Infraestrutura

- Existem alguns protocolos que são essenciais para o funcionamento da Internet, mas que não se encaixam naturalmente em um modelo estrito de camadas.
- Um desses protocolos é o Domain Name System (DNS)—não é uma aplicação que os usuários normalmente invocam explicitamente, mas é um serviço que quase todas outras aplicações dependem.
- O serviço de nomes é usado para traduzir nomes de host em endereços de host; a existência desse serviço permite que os usuários referenciem hosts remotos pelos nomes ao invés de endereços.

Serviços de Infraestrutura

■ Serviço de Nomes (DNS)

- Na maior parte do livro, nós usamos endereços para identificar hosts.
- Muito embora endereços sejam apropriados para processamento nos roteadores, endereços não são adequados para usuários.
- É por essa razão que um nome também é tipicamente atribuído a cada host em uma rede.
- Nomes de hosts diferem de endereços de hosts em duas maneiras.
 - Primeiro, eles são geralmente de comprimento variável e mnemônico, tornando-os mais fáceis de serem lembrados por humanos.
 - Segundo, nomes geralmente não contém informações que ajudem a rede a localizar (rotear o pacote para) o host.

Serviços de Infraestrutura

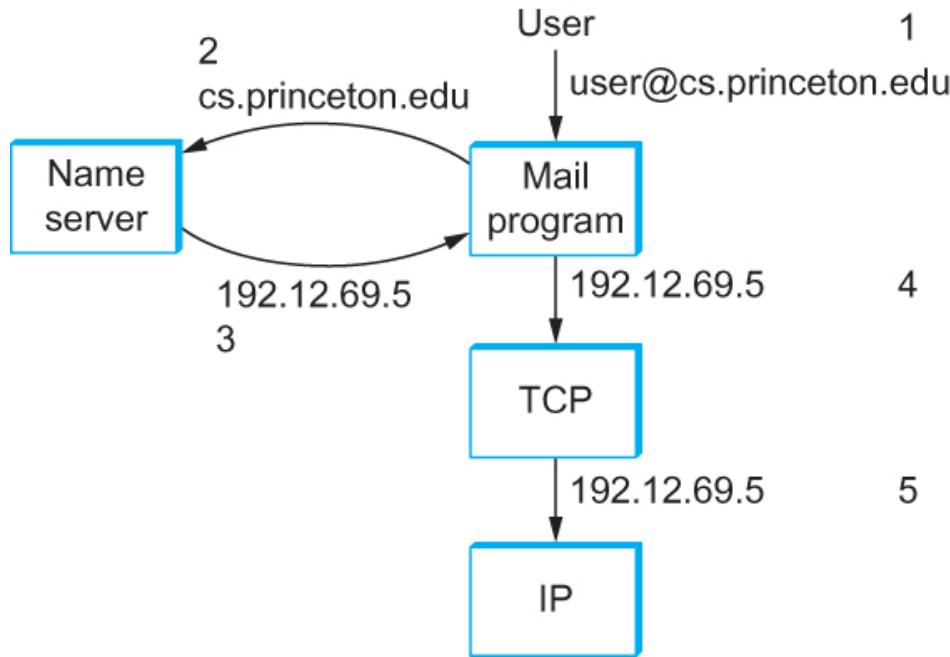
■ Serviço de Nomes (DNS)

■ Primeiro introduziremos a terminologia básica.

- Primeiro, um espaço de nomes (*name space*) define o conjunto de nomes possíveis.
 - Um espaço de nomes pode ser plano (nomes não são divisíveis em componentes) ou ele pode ser hierárquico (Nomes de arquivos em Unix são um exemplo óbvio).
- Segundo, o sistema de nomes (*name system*) mantém uma coleção de associações (*bindings*) de nomes para valores. O valor pode ser qualquer coisa que o sistema de nomes deve retornar quando um nome é fornecido; em muitos casos, o valor é um endereço.
- Finalmente, o mecanismo de resolução é um procedimento que, quando invocado com um nome, retorna o valor correspondente. Um servidor de nomes é uma implementação específica do mecanismo de resolução que está disponível em uma rede e que pode ser consultado enviando-se uma mensagem para ele.

Serviços de Infraestrutura

■ Serviço de Nomes (DNS)



Nomes traduzidos em endereços. Os números 1-5 mostram a sequência de passos no processo.

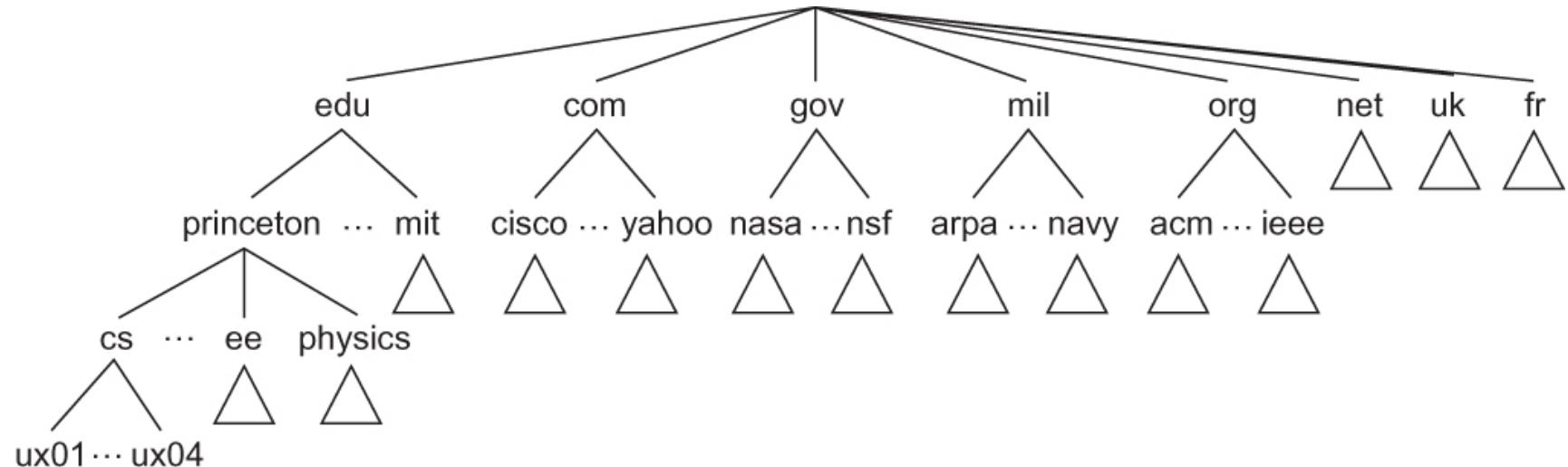
Serviços de Infraestrutura

■ Hierarquia de Domínios

- DNS implementa um espaço de nomes hierárquico para objetos da Internet. Diferente de nomes de arquivo em Unix, que são processados da esquerda para a direita e que possuem barras como separadores, nomes em DNS são processados da direita para a esquerda e possuem pontos como separadores.
- Da mesma maneira que a hierarquia de nomes em Unix, a hierarquia DNS pode ser visualizada como uma árvore, em que cada nó da árvore corresponde a um domínio, e as folhas correspondem aos hosts sendo nomeados.

Serviços de Infraestrutura

■ Hierarquia de Domínios



Exemplo de uma hierarquia de domínios.

Serviços de Infraestrutura

■ Servidores de Nomes

- Veremos agora como a hierarquia de nomes de domínio é de fato implementada.
- O primeiro passo é partitionar a hierarquia em subárvores chamadas *zonas*.
- Cada zona pode ser vista como alguma autoridade administrativa que é responsável por uma porção da hierarquia.
- Por exemplo, o primeiro nível da hierarquia, também conhecido como TLD – Top-Level Domains, forma uma zona que é gerenciada pela Internet Corporation for Assigned Names and Numbers (ICANN).
- No segundo nível, há, por exemplo, uma zona que corresponde à Princeton University (princeton.edu).

Serviços de Infraestrutura

■ Servidor de Nomes

- Cada servidor de nome implementa as informações da zona como uma coleção de registros de recurso (*resource records*).
- Um registro de recurso é uma associação nome-valor, ou mais especificamente a 5-tupla que contém os seguintes campos:
<Nome, Valor, Tipo, Classe, TTL>
- Classe → IN representa Internet. DNS foi pensado como um serviço de nomes genérico, mas hoje em dia existe uma única classe (IN).

Serviços de Infraestrutura

■ Servidor de Nomes

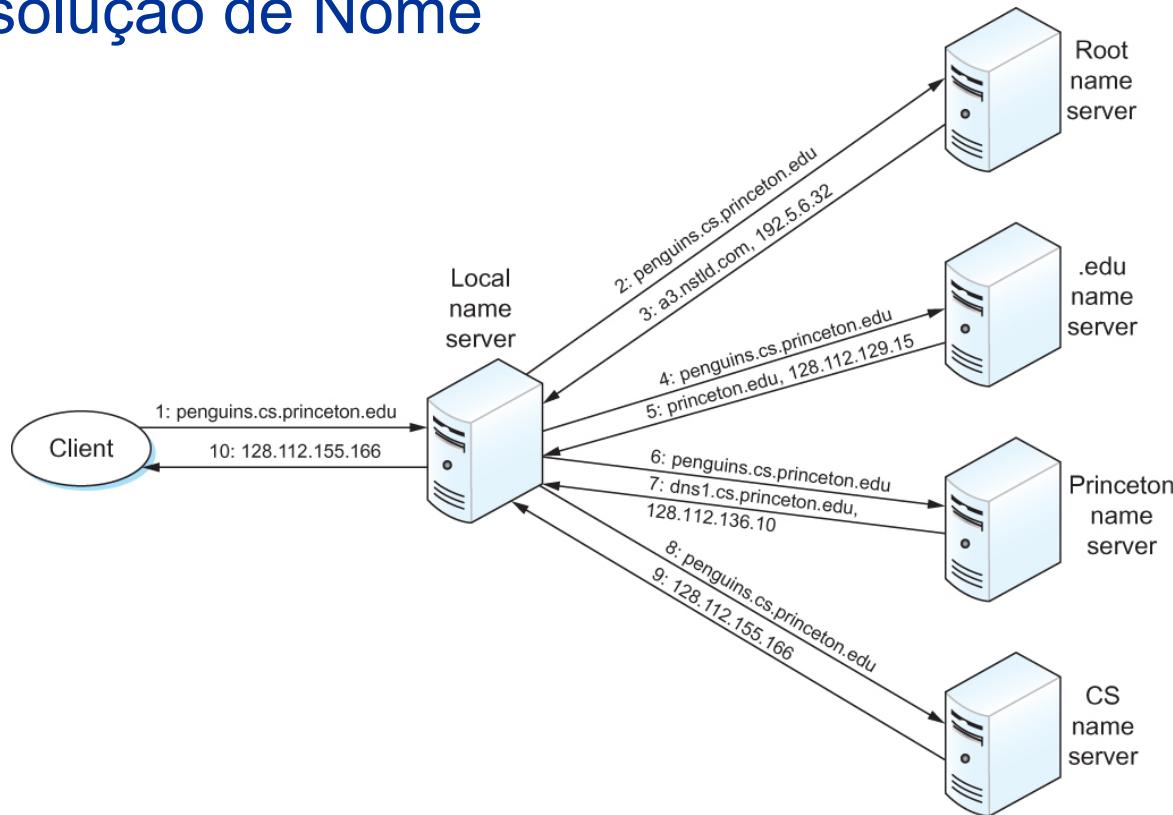
- Os campos Nome e Valor são exatamente o que você esperaria, enquanto o campo Tipo especifica como o Valor deve ser interpretado.
 - Por exemplo, Tipo = A indica que o Valor é um endereço IP. Assim, o registro A implementa o mapeamento nome-endereço que temos utilizado. Outros tipos de recurso incluem:
 - NS: O campo Valor indica o nome de um host que está rodando o servidor de nomes para resolver nomes dentro do domínio especificado.
 - CNAME: O campo Valor indica o nome canônico de um host; este registro é usado para definir apelidos (*aliases*).
 - MX: O campo Valor indica o nome de um host que está rodando o servidor de email que aceita mensagens eletrônicas do domínio especificado.

Serviços de Infraestrutura

- Exemplos de registros de recurso
 - <edu, a3.nstld.com, NS, IN>
 - <a3.nstld.com, 192.5.6.32, A, IN>
 - <com, a.gtld-servers.net, NS, IN>
 - <a.gtld-servers.net, A, IN>
 - <princeton.edu, dns.princeton.edu, NS, IN>
 - <dns.princeton.edu, 128.112.129.15, A, IN>

Serviços de Infraestrutura

■ Resolução de Nome



Resolução de nome na prática. Os números 1–10 mostram a sequência de passos no process.

Serviços de Infraestrutura

■ Resolução de Nome

- Cada host é configurado com o endereço do servidor de nomes de seu domínio (dns1.cs.princeton.edu)
- O servidor de nomes possui um mapeamento estático para os servidores do domínio raiz que é armazenado em um arquivo de configuração.
- O servidor de nomes armazena as respostas para suas consultas em cache para evitar o envio de mensagens na Internet e melhorar o tempo de resposta.
- Utilize os comandos abaixo para visualizar o processo de resolução de nomes
 - dig +trace www.cs.princeton.edu
 - dig +trace cs.princeton.edu MX
 - dig +trace facom.ufms.br MX

DNS Root Servers

13 root servers (see <http://www.root-servers.org/>)

Labeled A through M



news [see all news items](#)

2016-12-02 [Announcement of IPv6 addresses](#)

2016-06-29 [Root DNS events of 2016-06-25](#)

2016-02-08 [The 2015 Root Server Operators' Exercise on Emergency Response](#)

meeting agendas

2016-11-13 [IETF 97/Seoul \(PDF\)](#)

2016-07-17 [IETF 96/Berlin \(PDF\)](#)

2016-04-03 [IETF 95/Buenos Aires \(PDF\)](#)

2015-11-01 [IETF 94/Japan \(PDF\)](#)



DNS Caching

Realizar todas essas consultas leva tempo

E tudo isso antes da comunicação de fato começar

E.g., latência de 1 segundo antes de iniciar o download de uma página Web

Caching pode diminuir o overhead significativamente

Os servidores do topo da hierarquia raramente mudam

Sites populares (e.g., www.cnn.com) são visitados com frequência

Servidor DNS local geralmente tem a informação em cache

Como o cache do DNS funciona

Servidores DNS armazenam respostas das consultas

As respostas incluem um campo “time to live” (TTL)

Servidor apaga a entrada do cache depois que o TTL expira

Caching

Resolvedores colocam em cache as respostas DNS

Resposta rápida para traduções repetidas

Outras consultas podem reutilizar partes de um lookup anterior

NS records para domínios ficam em cache por mais tempo

Respostas negativas também são colocadas em cache

Erros de digitação, “localhost”, etc.

Dados em cache são descartados periodicamente (timeout)

Tempo de vida (TTL) dos dados controlado pelo dono dos dados

TTL é passado com cada registro

Consistência do Cache DNS

Objetivo: assegurar que dados em cache estão atualizados

Considerações no projeto do DNS

Dados em cache são “read only”

Invalidação explícita seria muito cara

Servidor teria de manter informações sobre todos os resolvedores que receberam respostas dele

Evitando informações desatualizadas

Respostas incluem um campo “time to live” (TTL)

Apaga a entrada em cache depois que o TTL expira

Faz cache negativo (para links mortos, erros de digitação)

Falhe rapidamente e não sobrecarregue os servidores gTLD

Definindo o Time To Live (TTL)

TTL trade-offs

TTL pequeno: respostas rápidas a mudanças

TTL grande: maior taxa de acerto em cache (*cache hit rate*)

Seguindo a hierarquia

Topo da hierarquia: dias ou semanas

Base da hierarquia: segundos ou horas

Tensões na prática

CDNs usam pequenos TTLs para balanceamento de carga e recuperação de falhas

Navegadores guardam em cache por 15-60 segundos

DNS Resource Records

RR format: `(name, value, type, ttl)`

- **Type=A**
 - Name: hostname
 - Value: IP address
- **Type=NS**
 - Name: domain
 - Value: hostname of name server for domain
- **Type=CNAME**
 - Name: alias for some “canonical” (the real) name:
www.ibm.com is really srveast.backup2.ibm.com
 - Value: canonical name
- **Type=MX**
 - Value: name of mailserver associated with name

(Outros) Tipos de Registros...

A

NS

MX

CNAME

TXT

PTR

AAAA

SRV

Descobrindo o seu Servidor DNS Local

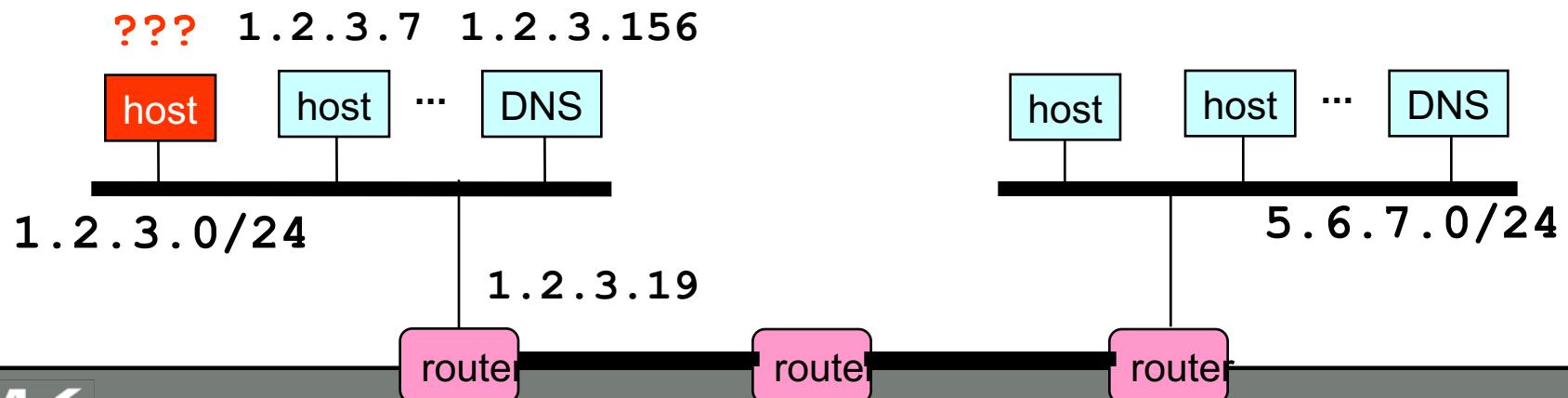
Como Inicializar um Host?

Qual servidor DNS local usar?

Qual endereço IP o host deve usar?

Como enviar pacotes para destinos remotos?

Como garantir que pacotes cheguem?



Confiabilidade

Servidores DNS são replicados

Serviço de nome disponível se pelo menos uma réplica está funcionando

Consultas podem ser distribuídas entre as réplicas

UDP usado para consultas

Deve implementar confiabilidade na aplicação

Tenta servidores alternativos depois de um timeout

Reculo exponencial quando tentar novamente o mesmo servidor

Mesmo identificador para as consultas replicadas

Não se preocupa com qual servidor respondeu

Truques com o DNS

Truque DNS #1: Reverse Lookup

Método

Hierarquia baseado em endereços IP

130.207.7.36

Consulta pelo registro PTR de 36.7.207.130.in-addr.arpa.

Gerenciamento

Autoridade gerencia os endereços a ela atribuídos

Truque DNS #2: Balanceamento de Carga

Servidor responde com vários registros A
A ordem desses registros muda a cada cliente