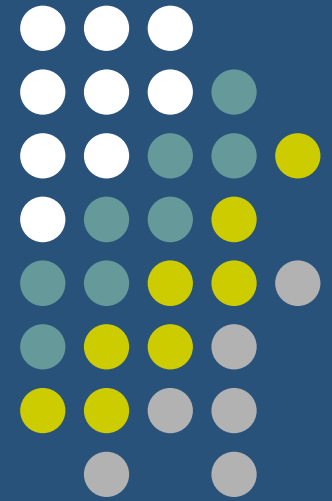


Sistemas Distribuídos

Módulo 5 – Network File System (NFS)



Introdução



- Proposto inicialmente pela Sun Microsystems para sistemas Unix, no início da década de 1990
- Suporte a sistemas heterogêneos (software e hardware)
- Utiliza a RPC para comunicação entre cliente e servidor



Características básicas

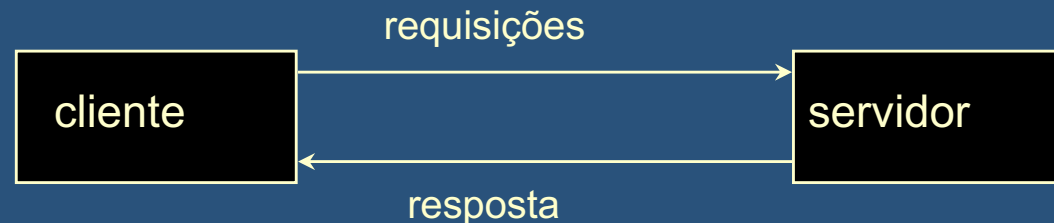
- Objetivo: Compartilhamento de Arquivos
- Arquitetura: Cliente/Servidor
 - Servidor NFS: exporta diretórios para disponibilizá-los a clientes remotos.
 - lista de diretórios exportados: /etc/exports
 - comando: exportfs
 - Cliente NFS: monta os diretórios disponibilizados no servidor. Os arquivos montados com sucesso são automaticamente compartilhados.

Protocolos do NFS



- Protocolos:

- protocolo: conjunto de mensagens enviadas entre o cliente e o servidor. Para cada mensagem, o servidor executa um conjunto de rotinas.



- O NFS possui dois protocolos:

- montagem de arquivos
- acesso a arquivos e diretórios

NFS – Montagem de Diretórios



- 1) O cliente envia um path ao servidor e pede permissão para montá-lo
- 2) O servidor verifica se o path existe e se ele foi exportado
- 3) O servidor envia a autorização para manipulação de arquivo

NFS – Montagem de Diretórios



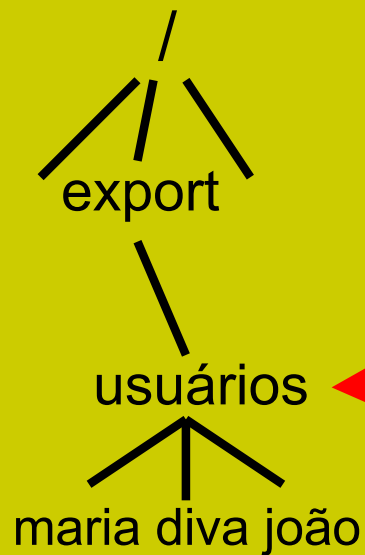
- Autorização para Manipulação de Arquivos:
esta autorização contém os seguintes dados:
tipo de sistema de arquivo, disco, número do i-nodo, segurança, etc
 - todos os acessos ao arquivo serão via os dados contidos na autorização

NFS – Montagem de Diretórios

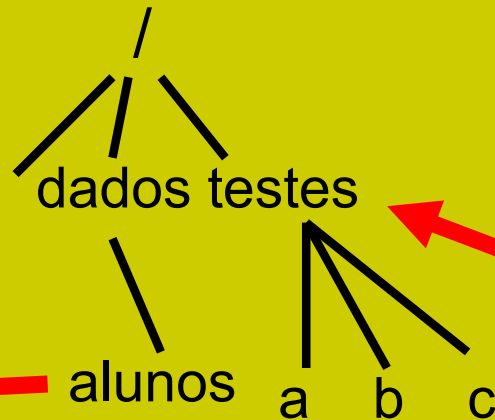


- montagem manual: feita manualmente sempre que se deseja acessar um diretório remoto
- montagem automática: feita no momento do boot (/etc/rc)
- automontagem : um conjunto de diretórios remotos é associado a um diretório local. Quando um arquivo remoto for aberto pela primeira vez, o primeiro servidor que responder terá seu diretório montado no cliente.
 - O usuário deve ter certeza de que os diretórios remotos pertencentes ao conjunto de automontagem serão sempre idênticos (tipicamente READ-ONLY).

NFS – Montagem de Diretórios



Máquina 1



Máquina 2



Máquina 3



NFS – Acesso a Arquivos

- As chamadas Unix tradicionais para tratamento de arquivos são aceitas pelo NFS, com exceção de “open” e “close”.
- Ao invés do “open”, nós temos uma operação de lookup, que retorna uma autorização para a manipulação do arquivo. Não gera entrada na tabela de arquivos.



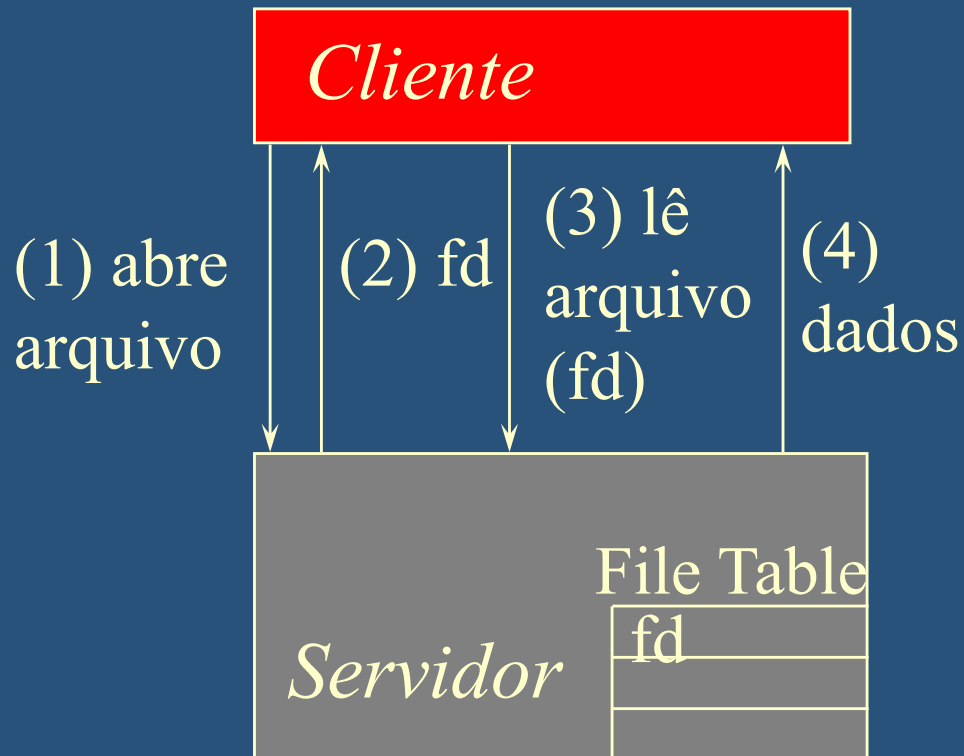
NFS – Acesso a Arquivos

- Read e write: contêm as informações contidas na autorização para manipulação de arquivo, além das informações inerentes à leitura ou escrita.
- O tratamento de operações sobre arquivos no NFS é “stateless” (não depende de estado anterior para ser executado).
- Algumas operações podem não se executar corretamente em um ambiente stateless. Ex: bloqueio de arquivos.

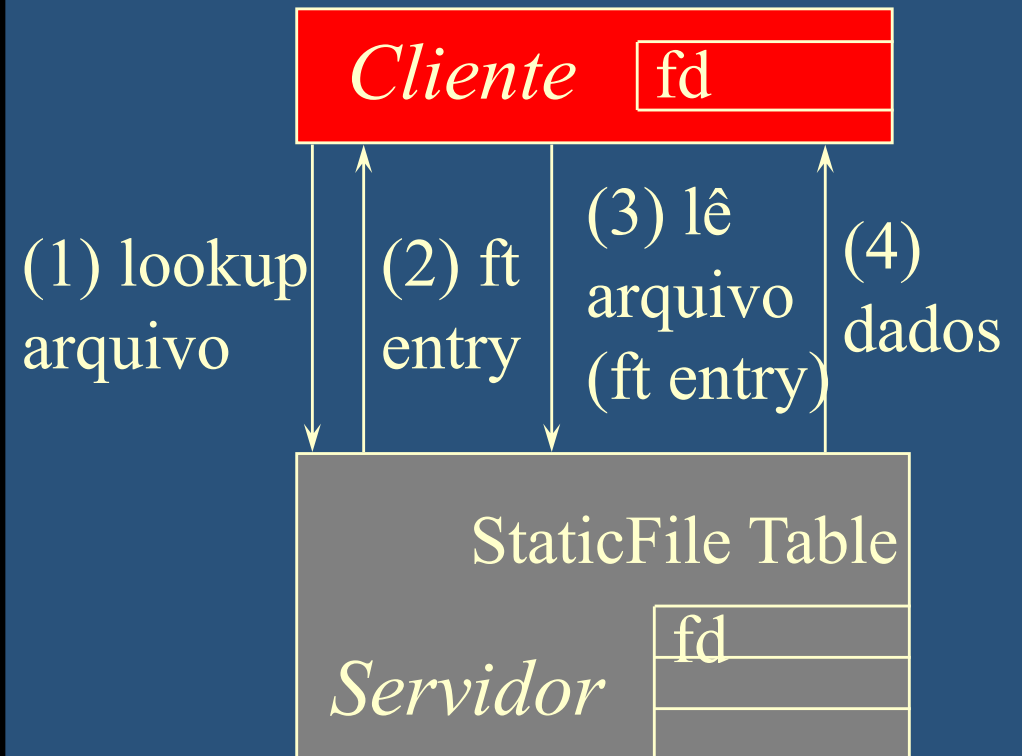
NFS – Servidor Stateless



- Servidor Tradicional



- Servidor stateless





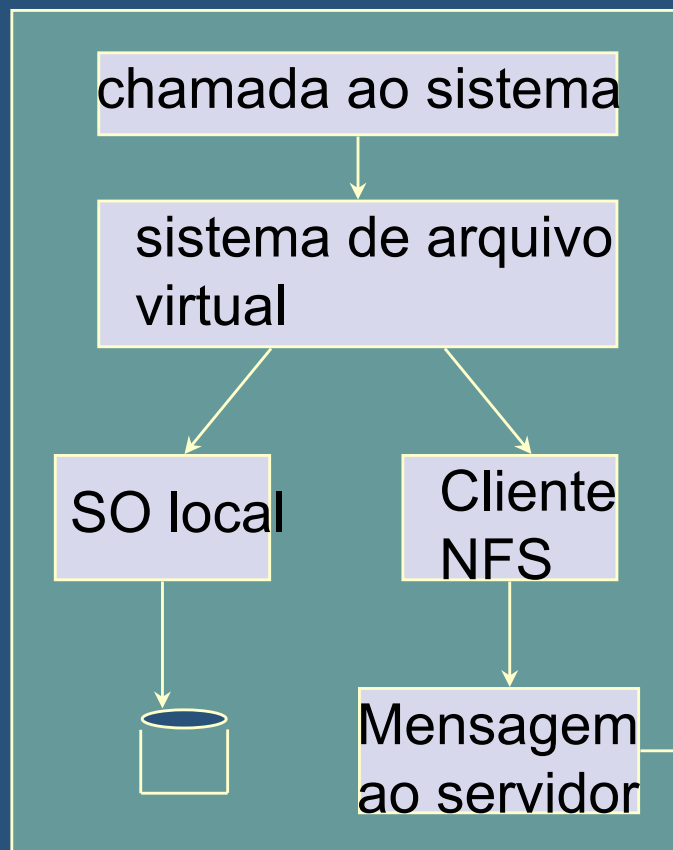
NFS – Servidor Stateless

- O NFS usa os bits rwx do Unix para proteção e solicita acessos a arquivos via RPC.
- Nesse caso, o cliente pode simplesmente alterar estes dados na mensagem de manipulação e executar operações para as quais não está autorizado, burlando o sistema de proteção
- Para resolver este problema, a cada cliente, é associada uma chave criptografada:
 - RPC com criptografia
 - Kerberos integrado ao NFS

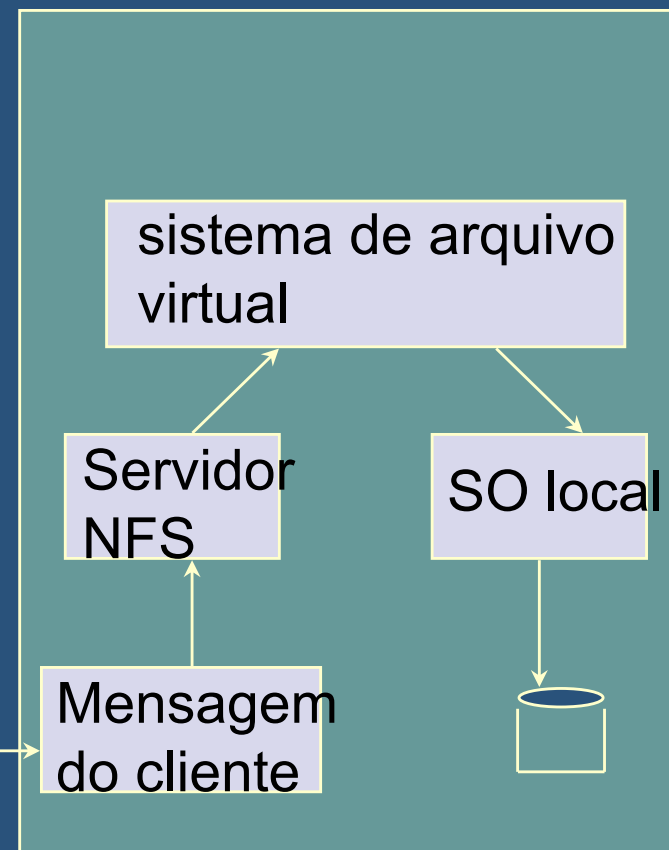
NFS – Implementação no SunOS



Cliente



Servidor



NFS – Implementação no SunOS



- Em primeira instância, os arquivos possuem v-nodos.
- Existe um v-nodo por arquivo (local ou remoto).
- O v-nodo identifica um arquivo virtual.
- O v-nodo pode apontar para um i-nodo (local) ou para um r-nodo (remoto).
- O r-nodo contém as informações da mensagem de autorização de manipulação de arquivo.

Caching



- Para melhorar o desempenho do sistema de arquivos, são usadas buffer caches em memória RAM.
- Read-ahead: antecipa os acessos de leitura, buscando dados próximos dos dados mais recentemente lidos
- Delayed-write: escreve os dados alterados em memória e, em um instante posterior, escreve-os em disco.
- Operação sync: escreve em disco todos os dados modificados há mais de 30 segundos (default)



Caching no Servidor NFS

- A buffer cache é usada na leitura de maneira similar ao Unix.
- Na escrita, existem duas opções:
 - Write-through: os dados são escritos na buffer cache e no disco
 - Write-back: os dados são escritos na buffer cache. Só serão escritos em disco quando uma operação commit for executada



Caching no Cliente NFS

- Os resultados das operações read, write, getattr, lookup e readdir são mantidos na cache do cliente.
- Caso os clientes desejem saber sobre a consistência dos dados, devem interrogar o servidor (polling)

Caching no Cliente NFS

Polling do Servidor - Leituras



- Cada dado na buffer cache possui dois timestamps:
 - T_c : tempo da última validação do dado
 - T_m : tempo da última modificação no servidor
- O dado está válido se:
 - $T - T_c < t$ OU $T_{mcliente} = T_{mservidor}$, onde
 - T = tempo atual
 - t = tempo de freshness

Caching no Cliente NFS

Polling do Servidor - Leituras



- O valor $T_{mservidor}$ é checado a cada t .
- No solaris, t é setado por arquivo e varia normalmente entre 3s e 30s.
- O $T_{mservidor}$ é obtido através da chamada `getattr`, somente se $T - T_c \geq t$.
- Caso $T_{mservidor}$ seja diferente de $T_{mcliente}$, o dado é buscado do servidor.

Caching no Cliente NFS

Polling do Servidor - Escritas



- Quando o dado é modificado na buffer cache, ele é marcado como dirty.
- Os dados são enviados ao servidor quando for executada a operação sync ou quando o arquivo for fechado.

NFS – Resumo de Características



Transparência de acesso	Sim
Transparência de localização	Não força um espaço único de nomes, porém pode ser implementado
Transparência de migração	Não
Escalabilidade	Limitada
Replicação de arquivos	R/O files, R/W com NIS
Heterogeneidade de HW	Sim

NFS – Resumo de Características



Heterogeneidade de OS	Sim
Tolerância a Falhas	Sim (stateless execution)
Consistência	Limitada
Segurança	Sim, com Kerberos e RPC-SEC
Eficiência	Sim