

Servidor de diretórios para descoberta de servidores

Na atividade prática 02, foi solicitado a comunicação entre cliente e servidor assumindo que os IP e Porta de todos são globalmente conhecidos.

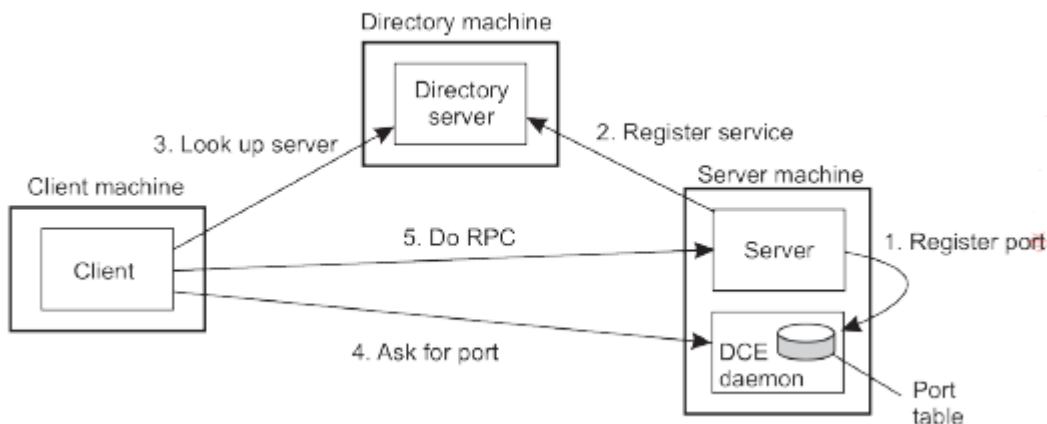
Na atividade prática 03, foi solicitado a conexão e comunicação entre clientes e servidores, mas a parte de descoberta de IP:Porta do Servidor era feita através de um módulo de mapeamento que já vinha implementado na construção dos arquivos e do sistema.

Portanto neste trabalho está sendo cobrado a parte de descoberta dos servidores, para isso, será necessário implementar um servidor de diretórios.

O que é um servidor/serviço de diretórios?

Toda aplicação RPC cliente-servidor, em geral, é baseada num serviço de diretórios, isto é, para que o cliente possa contatar o servidor desejado, ele precisa antes saber o endereço do servidor para destinar a mensagem para o caminho certo, mas para descobrir o endereço do servidor desejado, o cliente antes contacta um servidor/serviço de diretórios que contém uma lista com todos os servidores ativos no momento e esse servidor de diretórios irá retornar o IP e Porta do servidor desejado pelo cliente.

Na Figura a seguir está um exemplo dos passos relacionados ao processo de localização do servidor:



O primeiro passo é o servidor iniciar e se registrar em uma porta específica da máquina.

No segundo passo, o servidor envia suas informações para o servidor de diretório, isto é, seu nome, ip e porta que está funcionando para registro na lista.

Então o cliente faz a busca no servidor de diretórios procurando pelo nome do servidor que deseja e obtém como resposta o IP:PORTA do servidor.

Por fim, o cliente contata diretamente o servidor desejado.

Na imagem mostra que existe um processo de descobrir a porta através do DCE Daemon, mas isso só é necessário se o servidor de diretórios só tiver conhecimento do IP, mas no nosso caso, ele terá conhecimento tanto do IP quanto da Porta do servidor, logo, não precisa fazer isso.

O que terão que fazer?

A implementação de:

- **Servidor de diretórios:**

O servidor de diretório irá conter uma lista com os servidores e deve conter uma função que permita que servidores se registrem, uma função para apagar o registro da lista de servidores e uma função para atualizar o IP para caso o servidor tenha seu IP alterado.

Não é permitido que a lista tenha dois servidores com o mesmo nome, então é necessário fazer um controle.

Somente o servidor que se registrou pode remover o seu registro do servidor de diretórios, então assim que um servidor é registrado, o servidor de diretórios devolve um token/senha para o servidor que será necessário para remoção futura. O mesmo vale para atualização do IP na lista de servidores.

- **2 servidores:**

Um servidor "**calculadora**" que deve conter as funções básicas de matemática (soma, subtração, multiplicação e divisão).

Um servidor de criptografia, esse servidor irá receber uma string e fará a criptografia, depois retornando para o cliente a string criptografada e a chave pública, mas também deve conter uma função para fazer o processo contrário, para isso, vai receber a string e a sua chave pública e retornará a string descriptografada.

O tipo de criptografia utilizada será da escolha de vocês.

Ambos os servidores devem conter uma função de registro, isto é, enviar o seu nome, ip e porta para o servidor de diretórios.

- **2 ou mais clientes:**

Pelo menos um cliente para a calculadora e um cliente para o servidor de criptografia.

O projeto deve ser feito no AWS, utilizando máquinas diferentes para os clientes e servidores.

Considere que o Servidor de Diretórios é globalmente conhecido entre os clientes e servidores, dessa forma todos conhecem seu IP e porta.

Dicas:

Utilizem a linguagem Python e o RPyC.

No servidor de diretórios, utilizar um Threaded Server.

O arquivo constRPyC deve estar presente em todas as máquinas com as informações do IP e porta do servidor de diretórios.