

Trabalho de Implementação
Um Inspetor HTTP baseado em Proxy Server

Resumo

O objetivo deste trabalho é aplicar o conhecimento adquirido em sala sobre o funcionamento de redes de computadores construindo um inspetor de tráfego HTTP baseado em um servidor Proxy. Para tal o servidor deverá receber requisições HTTP processando seu cabeçalho para verificação do endereço de destino. Para geração das requisições HTTP um navegador web pode ser utilizado, bastando para isso que o servidor e o navegador sejam propriamente configurados.

Além desta funcionalidade básica, o aplicativo também deve executar mais duas funcionalidades:

- o spider, que consiste em enumerar todas as URLs subjacentes a uma URL definida formando sua árvore hipertextual. Se no processo forem encontradas URLs que saem do domínio da URL raiz, elas não devem ser seguidas.
- cliente recursivo, que nos moldes do aplicativo wget faz o dump do conteúdo a partir de uma URL corrigindo as referências lá contidas para que a árvore obtida seja visualizada no navegador como sendo local.

Funcionamento

O aplicativo deverá executar como um serviço proxy local sobre a porta 8228/TCP. O navegador deverá ser configurado para utilizar proxy no endereço 127.0.0.1:8228. Assim, todas as requisições do navegador serão redirecionadas para o proxy que atuará conforme desejado.

O aplicativo deverá ser usado da seguinte forma:

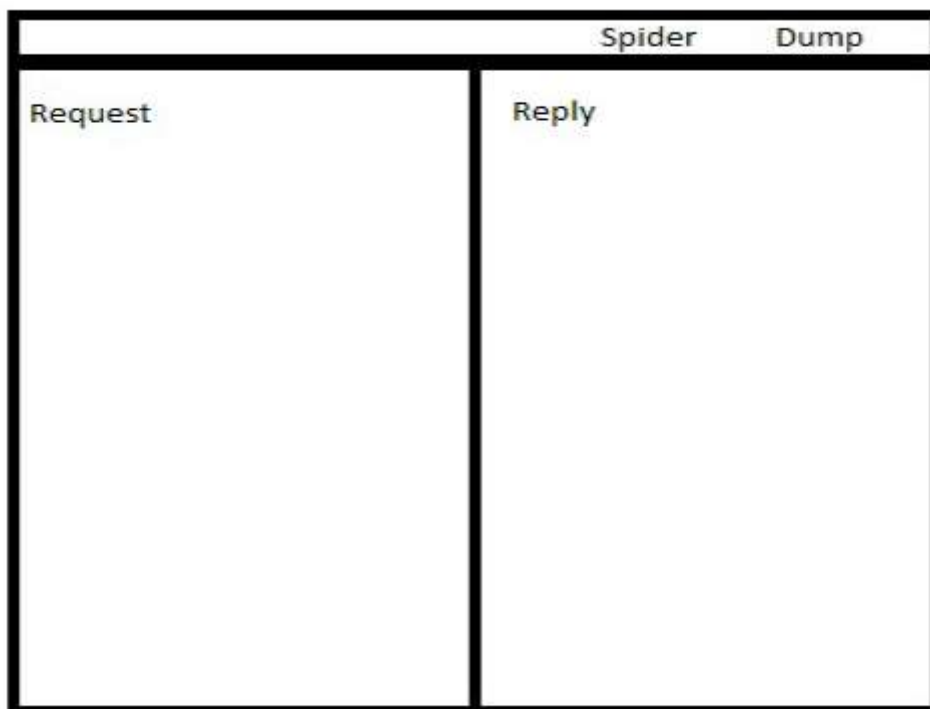
```
# aracne [-p <port>]
```

onde [-p <port>] designa a porta TCP a ser usada pelo proxy; se ausente, a default (8228) deverá ser usada.

Ao ser ativado o aplicativo abrirá uma janela com duas subjanelas (cada uma com um botão) e uma barra contendo dois botões, conforme a figura abaixo.

Os botões na barra ativam as funcionalidades spider e cliente recursivo, abrindo uma nova janela. No caso do spider, mostrando a árvore hipertextual. No outro caso, a evolução dos objetos baixados é mostrada.

Nas subjanelas, tem-se uma para os requests e outra para os replies. Em cada uma delas ocorre a inspeção da URL que foi suprida no navegador. Em cada uma destas subjanelas deve ser possível editar o conteúdo que está sendo enviado ou recebido. Na subjanela request, quando o botão request for clicado, o request interceptado deve ser enviado para o destino correspondente. Já na subjanela reply, quando o botão reply for clicado, o reply interceptado deve ser entregue ao navegador.



Restrições

1. Só devem ser tratadas requisições do tipo HTTP. Não serão realizadas requisições do tipo HTTPS durante os testes.
2. Não será aceita a utilização de bibliotecas proxy ou de HTTP Parser já desenvolvidas.
3. Deve ser usada API Sockets em C ou C++. O uso de outras bibliotecas dever ser aprovado previamente pelo professor da disciplina.
4. O trabalho deve ser realizado em grupos de no máximo 2 alunos.

Relatório

Um relatório do projeto deve ser apresentado, contendo:

- 1 Apresentação teórica o Proxy Server Web, TCP e o protocolo HTTP e das funcionalidades implementadas.
- 2 Documento apresentando a arquitetura do sistema desenvolvido com a descrição da arquitetura produzida e da relação entre dos principais componentes;
- 3 Documentação de todo o código desenvolvido;
- 4 Screenshots e explicação do funcionamento das funcionalidades implementadas.

Avaliação

A avaliação consiste em 3 componentes:

- 1 Código do projeto:
o trabalho deve ser desenvolvido OBRIGATORIAMENTE utilizando-se de uma ferramenta de versionamento online (Ex.:GitHub, BitBucket, etc).
- 2 Apresentação e demonstração do funcionamento
- 3 Nota do Relatório

Data da entrega/apresentação: 28/11/2019