Autorização de Operações ao nível do Sistema de Ficheiros

Henrique Faria and Paulo Bento

Universidade do Minho, Mestrado Integrado em Engenharia Informática

Resumo Este trabalho pretende abordar o sistema de gestão de ficheiros libfuse. No desenvolvimento deste projeto utilizaram-se as seguintes linguagens e ferramentas. MongoDB para a base de dados, HTML e CSS para criar as páginas web, Flask (python) para fazer a ligação da base de dados ás páginas web e enviar emails através do Gmail e Jinja2 para manipular objetos trocados entre as páginas web e o Flask.

Palavras-chave: Libfuse · MongoDB · HTML · CSS · Flask · Jinja2

1 Descrição do funcionamento do trabalho desenvolvido

Neste trabalho começamos por criar a base de dados *fuse* as coleções do mongo onde se guardam os utilizadores*(nome do utilizador no sistema e email)*, códigos por validar*(nome do utilizador do sistema e código)*, um registo de logs de acesso ao sistema *(nome do utilizador no sistema, timestamp da data de acesso e um marcador que diz se o acesso é válido ou se já se tornou inválido)* e um registo de possiveis tentativas de ransomware *(nome do utilizador no sistema e timestamp da data de acesso)*.

Posteriormente procedeu-se á implementação, em Python, do mecanismo de controlo de acessos ao sistema de ficheiros. Para poder acedr ao sistema de ficheiros, o utilizador tem de fornecer um email válido no qual vai receber um código de segurança gerado aleatóriamente e que permanece válido por 1 minuto. Caso o código inserido pelo utilizador esteja errado mas dentro do limite de tempo este é informado que se enganou o código e que ainda tem tempo para voltar a tentar. Caso o código seja inserido após o tempo limite o código é considerado inválido e o utilizador é retornado á página inicial para poder pedir um novo código. Caso o código inserido pelo cliente seja válido e tenha sido inserido dentro do prazo limite é garantido ao utilizador acesso por 2 minutos ao sistema de ficheiros. Após os 2 minutos de acesso garantido, um novo email é enviado ao utilizador para que este possa readquirir acesso ao sistema por mais 2 minutos.

Para que este istema funcione todos os utilizadores válido têm de estar registados na base de dados na coleção *users*. Sempre que um código é requisitado é adicionado um documento com o userId e o código á coleção *validCodes* que acaba por ser removido da base de dados quando o utilizador se autentica corretamente ou quando o tempo para se autenticar acaba. Ao procedr corretamente á inserção do código secreto o documento usado para guardar o código e o nome do utilizador é descartado da base de dados e porcede-se á inserção na coleção *log* um documento que regista o nome do utilizador que acedeu aos ficheiros um timestamp que indica o momento que que acedeu aos mesmos e se o acesso se encontra válido ou se já foi invalidado. Adicionalmente foi adicionada a funcionalidade de, ao alterar um ficheiro, caso a alteração caia na suspeita de se poder tratar de *ransomware*, o ficheiro é salvo numa diretoria chamada *safe* para podermos recuperar o estado pré-ataque.

2 Passos para execução do trabalho

Para correr o trabalho é necessário ter uma base de dados mongo instalada no computador. São necessários apenas os documentos referentes aos utilizadores no sistema. No nosso caso, os documentos eram os seguintes:

{ "_id": ObjectId("5e183e53ad2e22d27a2f1ffb"), "userId": "henrique", "email":"henriquejosefaria@gmail.com"}

{ "_id": ObjectId("5e183e53ad2e22d27a2f1ffc"), "userId": "paulo", "email":"Paulo_jorge_000_hotmail.com"}

Note-se que o campo userId resulta da execução do comando "id -nu".

De seguida serão precisos realizar os seguintes passos pela ordem em que aparecem:

- Cria a pasta onde será montado o sistema de ficheiros.
- Ligar o servidor com o comando python3 webApp.py.
- Proceder á montagem e uso do sistema de ficheiros com o comando: *pyhton3 Passthrough.py / dir*.

  A *"/"* permite encontrar o caminho desde a raiz até á diretoria em que nos encontramos.
  A pasta *"dir"* é a diretoria criada para a implementação do sistema de ficheiros.

A curta duração do acesso serve o propósito de mostrar a funcionalidade do sistema de negação de acesso após um tempo estipulado e pré estabelecido, bem como a reaquisição do acesso por parte do utilizador.