

passwdNG

Gerenciamento de usuário/senha no Linux

Valor 1,5 na média

A APS consiste em desenvolver um software que substitua o comando `passwd` do Linux. Lembrando que o comando `passwd` é o comando responsável por tratar senhas de usuários no Linux, então o software que iremos desenvolver, e que chamaremos por enquanto de `passwdNG` (`passwd` New Generation), deve implementar e estender o comando `passwd`, bem como suas funcionalidades.

A APS pode ser feita de forma individual ou em grupo. Contudo a APS só pode ser feita em grupo (duas pessoas ou mais), se a equipe estender as funções básicas (que serão apresentadas a seguir). A equipe deve ter um líder e o mesmo deve justificar, para o professor o que cada pessoa deve fazer e é o professor quem libera ou não a entrada de mais membros na equipe. Obrigatoriamente devem existir no mínimo duas equipes e tais equipes também devem utilizar linguagens de programação distintas.

Funções básicas do `passwdNG`:

1. Github – Tanto o código quanto documentação deve ser mantida no Github, tal link deve ser passado para o professor e cada membro da equipe deve ter um usuário relacionado com o projeto criado pelo líder da equipe.
2. Documentação - Deve haver documentação a respeito do que foi desenvolvido. O código fonte também deve estar comentado. Tanto documentação quanto comentários do código devem estar em inglês.
3. Funções mínimas - O `passwdNG` deve implementar as funções básicas no `passwd` original:
 - a) Adicionar senha para um usuário;
 - b) Apagar senha para um usuário;
 - c) Travar a senha de um usuário;
 - d) Destruir a senha de um usuário;
4. Recomendação/configuração de senhas - No `passwdNG`, deve ser possível configurar a recomendação da composição da senha:
 - a) Quantidade de caracteres minúsculos (26 letras a-z);
 - b) Quantidade de caracteres maiúsculos (26 letras A-Z);
 - c) Quantidade de números (10 números de 0-9);
 - d) Quantidade de caracteres especiais (aproximadamente 32 caracteres, tais como !, \$, *, etc);
 - e) Quantidade total de dígitos da senha;
 - f) Se pode repetir senhas e o período;
 - g) Período para troca de senhas;
 - h) Informar para o usuário se a senha é fraca ou não (senhas de dicionário, senhas muito conhecidas, senhas em branco, etc) .
5. Interface - O `passwdNG` deve ter uma interface texto (não pode ser *hard-code*) que permita:
 - a) Permitir a configuração de todas as características apresentadas no item 4;
 - b) Visualizar todos os usuários do sistema, seu UID, GID principal, diretório home, shell, e grupos que esse usuário faz parte.
 - c) O `passwdNG` deve avisar, caso o UID de algum usuário mude, principalmente se algum usuário se tornar root.

- d) O passwdNG deve ter uma interface administrativa, no qual o administrador pode ver estatísticas de todos os usuários do sistema, qualidade/nível das senhas, trocas de senhas/períodos e tudo que possa ajudar na administração de usuários e senhas.
- 6. Segurança - A equipe deve tomar cuidado para não enfraquecer o processo de criação e manutenção das senhas. Ou seja, todo o software passwdNG deve ser pensado para evitar problemas de segurança, posteriores.

Funções extras (extensão):

O passwdNG pode ter como funcionalidades extras:

1. Autenticação em mais de um passo. No que se refere a autenticação de usuários há cinco fatores básicos que podemos levar em consideração durante uma autenticação (a equipe pode escolher um ou mais desses passos para tentar implementar):
 - a) Alguma coisa que você conhece, tal como desafio – perguntas;
 - b) Alguma coisa que você tem, pode ser chave de identificação (chave criptográfica) em um pendrive, ou token.
 - c) Alguma coisa que você é, tal como a impressão digital de uma pessoa;
 - d) Alguma coisa que você faz, por exemplo gestos em uma tela touch screen.
2. Desenvolvimento de uma interface mais amigável para o passwdNG (para o item 5, das configurações básicas):
 - a) Interface Web;
 - b) Interface Android;
 - c) Interface gráfica (GTK, QT, etc);
3. Sistema para recuperação de senha, por exemplo por e-mail ou celular.
4. Uso de banco de dados. O passwd NG pode usar banco de dados para armazenar dados de usuários/senhas, bem como configurações. Contudo esse banco de dados configurado e utilizado de forma segura.
5. A equipe pode sugerir alguma funcionalidade extra a mais (isso deve ser liberado pelo professor). Exemplo autenticação usando Oauth.

Obs0: A data de entrega será definida em sala de aula.

Obs1: Usem os programas de quebra de senha como início!

Obs2: Qualquer dúvida entrem em contato com o professor (luizsantos@utfpr.edu.br).

1. Equipe:

- a) Link no Github: <https://github.com/tuchinski/>
- b) Membros:
 - o Leonardo Tuchinski
 - o Ana Frozza
 - o Vitor Cimetan
 - o Geovani Pedroso
 - o Pedro Perozin
 - o Allison Sampaio
 - o Rafael Menezes
- c) Linguagem de programação: Python
- d) Funcionalidades extras/extensão: Interface gráfica, recuperação de senha por e-mail, autenticação em dois passos.

2. Equipe:

- a) Link no Github: <https://github.com/rodps/passwdNG>
- b) Membros: Rodrigo
- c) Linguagem de programação: Python

- d) Funcionalidades extras/extensão:
- 3. Equipe 3:
 - a) Link no Github: <https://github.com/eboliveira/APS-Seg>
 - b) Membros:
 - Claudia
 - Danilo
 - Daniel
 - Eduardo
 - c) Linguagem de programação: Javascript
 - d) Funcionalidades extras/extensão: Interface gráfica e Banco de dados
- 4. Equipe X...