



TRABALHO FINAL DA DISCIPLINA

Orientações gerais:

- O trabalho poderá ser realizado em **dupla**
- Os trabalhos serão apresentados no dia **11/09/2015**
 - No caso da opção por dupla, os dois alunos devem possuir domínio sobre o conteúdo do trabalho pois, ambos irão apresentar o trabalho e serão questionados
- **No dia da apresentação será pedido para cada aluno que, individualmente, realize uma modificação ou acréscimo em um dos scripts**
- A nota final de cada aluno será composta por:
 - **Corretude dos scripts**
 - **Apresentação**
 - **Modificação ou acréscimo no script selecionado**

Dicas:

- Não esquecer de dar permissão de execução para os arquivos de script
- Executar os comandos referentes as ações individuais de cada script anteriormente na linha de comando
- Os scripts devem ser criados, executados e agendados como root
- Deixar a passphrase em branco ao gerar as chaves para permitir o SSH sem senha

Cenário: Você é o administrador de redes de uma determinada instituição de ensino. Para facilitar algumas de suas atividades diárias, você deverá criar alguns shell scripts.

Script 1 (adicionar um novo usuário no sistema): O script deve se chamar **novo_usuario.sh**. Existem 3 (três) tipos de usuários que acessam os sistemas da instituição: **professor, técnico e aluno**. Cada tipo de usuário possui um grupo correspondente: **professores, técnicos e alunos**. A senha do usuário deve ser cadastrada por ele no primeiro acesso. Uma **carta de boas-vindas** e o **manual de regras** da instituição devem ser fornecidos para o usuário. São fornecidos como parâmetro para o script o **tipo de usuário** e o **login** do novo usuário.

Pré-requisitos do script novo_usuario.sh:

- Criação dos grupos **professores, tecnicos e alunos** no sistema
- Criação do arquivo **regras.txt** dentro de **/home/**

Parâmetros do script novo_usuario.sh:

- **Tipo de usuário:** [professor, tecnico, aluno]
- **Login:** login do usuário no sistema. O login deve ser digitado em letras minúsculas e sem uso de caracteres especiais ou acentuação (ex.: jsilva, marcos, csantos).

Verificações realizadas pelo script antes da execução das ações:

- Verificar se o **login fornecido já existe no sistema**
- Verificar se o **tipo de usuário informado pertence as opções disponíveis**
- **Obs.: no caso de falha de qualquer umas das verificações, uma mensagem informativa deve ser exibida ao usuário e a execução o script abortada.**

Ações e regras do script novo_usuario.sh

- O diretório HOME do usuário será criado de acordo com seu tipo:
 - professor -> /home/**professores**/**<login>**
 - tecnico -> /home/**tecnicos**/**<login>**
 - aluno -> /home/**alunos**/**<login>**
- O usuário criado deve ser incluído no grupo correspondente
 - professor -> grupo **professores**
 - tecnico -> grupo **tecnicos**
 - aluno -> grupos **alunos**
- A senha do usuário deve ser excluída e expirada para que seja exigida sua modificação no próximo acesso do usuário no sistema
- Copiar o arquivo /home/regras.txt (que representa o manual de regras da instituição) para dentro do diretório HOME do usuário
- A carta de boas-vindas, contendo a data/hora da ativação do cadastro (data/hora de execução do script) e o "cargo" do usuário, deve ser criada dentro do diretório HOME do usuário
 - O nome do arquivo da carta deve seguir o padrão: **welcome_<login>.txt** (ex.: welcome_jsilva.txt)
 - A data/hora de ativação do cadastro deve seguir o padrão:
 - ⇒ **<dia> de <mês> de <ano> às <hora>:<minutos>:<segundos>**
 - ⇒ Ex.: 01 de Jan de 2015 às 15:32:22
 - Exemplo de carta:

Olá **jsilva**, seja bem-vindo(a) à EMPRESA X.

Seu cargo na instituição é de: **aluno**

A ativação do seu cadastro foi realizada em: **01 de Jan de 2015 às 15:32:22**

Bom trabalho!

Script 2 (backup dos dados dos usuários): O script deve se chamar **backup_dados.sh**. A **execução** do script será realizada de forma **automatizada** pelo sistema em **dias/horários pré-determinados**. Os backups devem ser **guardados localmente** em um **formato compactado**. Uma **cópia** do backup compactado deve ser **enviada para um servidor**.

Ações e regras do script backup_dados.sh

- O script deverá ser executado automaticamente nos seguintes horários:
 - Todas as terças, quintas e sábados às 2:15 da manhã
 - Domingo às 13h

- O backup deverá compactar todos os arquivos do diretório /home
- Os arquivos de backups serão armazenados localmente no seguinte local/padrão:
 - **/var/backups/<empresa>/<ano>/<mes>/backup-<data>.tar.gz**
 - Ex.: /var/backups/ifrn/2015/9/backup-20150901.tar.gz
- Uma cópia do arquivo deve ser enviada via SSH para um servidor de IP A.B.C.D (outro computador qualquer na rede) e ser armazenado no diretório /var/backups

Script 3 (relatório gerencial do servidor): O script deve se chamar **relatorio_servidor.sh**. Serão coletadas as informações: **nome do computador; IP, Máscara de rede e MAC placa de rede eth0; quantidade de memória RAM instalada; fabricante, velocidade, número de núcleos e arquitetura do processador; serviços TCP rodando no servidor.**

Exemplo de relatório:

Relatório Gerencial do Servidor **ubuntu-14**

1 - Informações da placa de rede eth0

- IP = **10.0.2.15/255.255.255.0**
- MAC = **08:00:27:eb:eb:50**

2 - Memória RAM (em MB) : **489** Megabytes

3 - Processador

- Fabricante : **GenuineIntel**
- Velocidade : **2300.438** MHz
- Número de núcleos : **1**
- Arquitetura : **x86_64**

4 - Serviços TCP em execução

Conexões Internet Ativas (somente servidores)

Proto	Recv-Q	Send-Q	Endereço Local	Endereço Remoto	Estado	PID/Program name
tcp	0	0	127.0.1.1:53	0.0.0.0:*	OUÇA	2428/dnsmasq
tcp	0	0	0.0.0.0:22	0.0.0.0:*	OUÇA	893/sshd
tcp	0	0	127.0.0.1:631	0.0.0.0:*	OUÇA	6737/cupsd
tcp6	0	0	:::22	:::*	OUÇA	893/sshd
tcp6	0	0	:::1:631	:::*	OUÇA	6737/cupsd

- Obs.: os dados em negrito são obtidos no momento da execução do script