SEG12 - Semana 1 - Sessão 10

Francisco Marcelo, Marcelo Karam e Felipe Scarel

06-08-2018

Servidor Web

As atividades desta sessão serão realizadas na máquina virtual *Server_Linux*, com pequenas exceções apontadas pelo enunciado dos exercícios.

O objetivo de um servidor web é, em essência, servir conteúdo para a *world wide web*. Esse objetivo é atingido servindo requisições enviadas ao servidor através do protocolo HTTP, bem como protocolos relacionados. Nesta sessão iremos instalar e configurar o servidor web Apache, um dos mais populares servidores HTTP *open source* do mundo.

1) Instalação do servidor web Apache

Instale o servidor web Apache (pacote `apache2`). Teste o funcionamento da instalação acessando a página web a partir de qualquer navegador (seja na máquina física, *Client_Linux* ou *Win7-padrao*).

2) Configuração de *virtualhosts*

Virtualhosts, ou servidores virtuais, podem ser utilizados nos seguintes casos comuns:

- Hospedar múltiplos *sites* diferentes em um mesmo endereço IP;
- Hospedar múltiplos *sites*, cada um com seu IP específico.

Destes, o primeiro cenário é o mais usual, e o que será abordado nesta atividade.

No servidor web Apache instalado em nosso servidor Debian, os arquivos de configuração de todos os *sites* devem ser colocados no pasta `/etc/apache2/sites-available`. Esses *sites* podem estar ativos ou inativos:

- Para ativar um *site*, basta criar um *link* simbólico do arquivo original para a pasta `/etc/apache2/sites-enabled` e recarregar o servidor Apache. Esse *link* pode ser criado manualmente, ou através do comando `a2ensite` ("*Apache 2 enable site*").
- Para desabilitar um *site*, toma-se o caminho oposto: apague o *link* simbólico da pasta `/etc/apache2/sites-enabled`, ou use o comando `a2dissite` ("*Apache 2 disable site*").

Relembrando a sessão 7 — DNS e NFS, criamos duas entradas `CNAME` apontando para a máquina *Server_Linux*, quais sejam:

```
# cat /etc/bind/db.empresa.com.br | grep 'CNAME *servidor'
www       IN    CNAME         servidor
meusite   IN    CNAME         servidor
```

1. Crie dois *virtualhosts* na máquina *Server_Linux*, um respondendo requisições enviadas para `www.empresa.com.br` e outro para `meusite.empresa.com.br`.
2. Crie pastas específicas para cada *virtualhost* dentro do diretório `/var/www`.
3. Crie arquivos `index.html` na raiz dessas pastas que identifiquem cada um dos *virtualhosts*.

4. Acesse os nomes de domínio a partir de um navegador (seja na máquina física, *Client_Linux* ou *Win7-padrao*) e verifique que suas configurações surtiram efeito.

3) Configuração de criptografia SSL

O protocolo HTTP não possui nenhum recurso de criptografia e, por consequência, todo o tráfego de rede gerado entre cliente e servidor poderia ser visualizado por um atacante. Para aumentar a segurança de aplicações web, é interessante habilitar o suporte a conexões cifradas através do *Secure Sockets Layer* (SSL).

1. Habilite o módulo SSL do Apache através do comando `a2enmod` ("*Apache 2 enable module*").
2. Crie um certificado auto-assinado RSA de 4096 bits para o *virtualhost* `meusite.empresa.com.br`, com validade de um ano. Armazene a chave pública na pasta `/etc/ssl/certs`, e a chave privada em `/etc/ssl/private`. Tenha atenção às permissões de arquivo e usuário/grupo dono.
3. Configure o *virtualhost* `meusite.empresa.com.br` para utilizar o protocolo HTTPS em qualquer conexão. Redirecione qualquer conexão sem criptografia direcionada à porta 80/HTTP para a porta 443/HTTPS.
4. Acesse o domínio `meusite.empresa.com.br` a partir de um navegador (seja na máquina física, *Client_Linux* ou *Win7-padrao*) e verifique que suas configurações surtiram efeito.

4) Autenticação e acesso a conteúdo restrito usando LDAP

Autenticação de usuários, especialmente em áreas sensíveis de um *site*, é integral à configuração de segurança de servidores web. Em particular, estamos interessados em habilitar autenticação para uma área restrita do *virtualhost* `meusite.empresa.com.br`.

1. Habilite o módulo de autenticação LDAP do Apache, `authnz_ldap`, através do comando `a2enmod`.
2. Crie uma pasta `/restrito` dentro da raiz do *virtualhost*. Dentro dessa pasta, crie um arquivo `index.html` que possa ser usado para testar a configuração.
3. Configure o *virtualhost* para requerer autenticação quando um usuário tentar acessar a URL `meusite.empresa.com.br/restrito`. Exija que o cliente forneça uma combinação de usuário/senha válida e existente na base LDAP local.
4. Acesse a URL `meusite.empresa.com.br/restrito` a partir de um navegador (seja na máquina física, *Client_Linux* ou *Win7-padrao*) e verifique que suas configurações surtiram efeito.

5) Habilitando páginas pessoais de usuários

O módulo `userdir` do Apache permite a um usuário publicar seu próprio *site*, localizado dentro da sua pasta pessoal. Ele procura uma pasta com nome `public_html` dentro do diretório *home* do usuário e, caso existente, serve o conteúdo dessa pasta via HTTP.

1. Habilite o módulo páginas pessoais do Apache, `userdir`, através do comando `a2enmod`.
2. Crie a pasta `public_html` dentro do diretório *home* do usuário `aluno` e insira dentro dela um

arquivo `index.html` que permita testar a configuração.

3. Configure o sistema para que todos os usuários criados futuramente já tenham a pasta `public_html` criada automaticamente eu seus diretórios *home*.
4. Teste o acesso à página pessoal do usuário `aluno` a partir de um navegador (seja na máquina física, *Client_Linux* ou *Win7-padrao*), verificando que suas configurações surtiram efeito.