Grupo de Resposta a Incidentes de Segurança – UFRJ

Última modificação: 07/10/05

Restrição de Acesso à Servidores Apache Baseada em Autenticação por Senha

GRIS-2005-T-002

Universidade Federal do Rio de Janeiro
Instituto de Matemática
Departamento de Ciência da Computação
Grupo de Resposta a Incidentes de Segurança

Rio de Janeiro, RJ – Brasil

Restrição de Acesso à Servidores Apache Baseada em Autenticação por Senha

GRIS-2005-T-002

Victor Batista da Silva Santos

A versão mais recente deste documento pode ser obtida na página oficial do GRIS

GRIS – Grupo de Resposta a Incidentes de Segurança
CCMN Bloco I 1° andar
Salas: I-1021
Av. Brigadeiro Trompowski, s/nº
Cidade Universitária - Rio de Janeiro/RJ
CEP: 21949-900
Telefone: +55 (21) 2598-3309

Sumário:

1. Introdução

Muitas vezes desejamos colocar dados em servidores web mas não queremos que os mesmos sejam públicos. Embora uma Firewall bem configurada possa resolver quando é apenas com o tráfego externo que estamos preocupados, a situação fica mais delicada quando apenas determinadas pessoas podem acessar tais dados, independente de onde elas estejam fisicamente localizadas, ou quando é apenas um determinado diretório que precisamos restringir, enquanto o restante das informações disponíveis no servidor deve permanecer público.

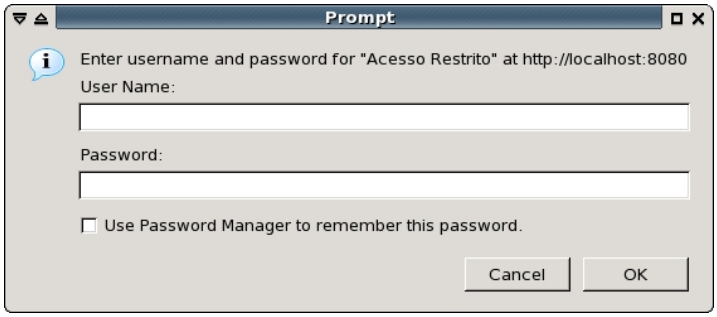
A maioria dos servidores web de hoje em dia possui uma ou outra maneira de resolver esse problema. O *Apache HTTP Server* (http://httpd.apache.org) é, segundo pesquisa realizada pela Netcraft, o servidor web mais utilizado no mundo, e aqui vamos falar um pouco sobre como ele trata esse tipo de necessidade.

Faremos uma breve descrição sobre o arquivo *".htaccess"*, apresentando suas características principais, estrutura e ensinando como criar e manter um arquivo desse tipo para para restringir aos usuários acesso ao seu servidor Apache.

2. O que é o arquivo *.htaccess*?

O arquivo *.htacess* (notem o ponto na frente do nome) é um arquivo de texto especial do Apache que restringe o acesso de usuários ao seu servidor, exigindo autenticação por intermédio de senha. Com ele, fica fácil proteger arquivos e diretórios que contenham informações sensíveis.

O *.htaccess* deve ser criado no mesmo diretório o qual ele protegerá, e restringirá acesso não somente a este diretório como também a todos os subdiretórios decorrentes. Isso porque sempre que o servidor percebe a existência de um arquivo *.htaccess* em qualquer um dos diretórios que fazem parte do caminho ao recurso solicitado pelo usuário, ele não envia o recurso, e sim uma mensagem de erro (401) complementada por uma ou mais linhas indicando a necessidade de autenticação (campo WWW-Authenticate do cabeçalho de resposta). Ao receber tal cabeçalho, a maioria dos navegadores web de hoje em dia abrem automaticamente uma janela solicitando login e senha de autenticação, como na figura abaixo.



3. Configurando o Apache para restringir acessos

Antes de mais nada, você precisará certificar-se de que o Apache está configurado para aceitar arquivos *.htaccess* como arquivos especiais. Para configurar esse comportamento, você precisará editar o arquivo de configuração do Apache, que é o *"httpd.conf"*. Geralmente ele está localizado no diretório *"/etc/httpd/conf"* ou no *"var/www/conf/"*.

Nele você deve procurar a entrada "*<Directory>*" que deseja restringir. Em nosso caso, vamos restringir acesso a todo o servidor, então procuraremos pela entrada *"/"*. Em nosso sistema exemplo ela se encontra da seguinte maneira:

```
<Directory "/">
   Options Indexes FollowSymLinks
   AllowOverride None
</Directory>
```

Note que os valores entre as tags *<Directory>* e *</Directory>* podem variar dependendo de como seu servidor está configurado. Aqui, a entrada que nos interessa é a *"AllowOverride"*, que colocamos em negrito. Mude o valor desta entrada de *"None"* para *"AuthConfig"*:

```
<Directory "/">
   Options Indexes FollowSymLinks
   AllowOverride AuthConfig
</Directory>
```

A opção *"AllowOverride AuthConfig"* diz para o Apache procurar pelos arquivos *.htaccess* nos diretórios web e aplicar as regras contidas no arquivo em todos os diretórios e subdiretórios onde ele esteja localizado. Note que devemos substituir o parâmetro de *AllowOverride* apenas se ele for *"None"*. Esta diretiva do Apache controla outras características que não somente autenticação. Assim, se em seu sistema for encontrado algo como

AllowOverride Indexes Limit

apenas adicione o parâmetro AuthConfig, sem apagar os anteriores:

AllowOverride AuthConfig Indexes Limit

Note ainda que, caso a diretiva esteja definida como *"All"*, o parâmetro *AuthConfig* já está incluído e nenhuma modificação precisa ser feita.

Colocada esta opção, é só reiniciar ou recarregar o servidor Web ele estará apto a receber e interpretar arquivos *.htaccess*. Configurado o servidor web, estamos prontos para criar nosso arquivo *.htacess* e começar a restrigir acessos.

4. Criando o arquivo .htaccess

Conforme mencionado antes, o arquivo *.htaccess* deve ser criado no diretório que ele protegerá. Uma boa sugestão é criar um diretório chamado "restrito" (por exemplo), no diretório raiz de seu site, e colocar todos os diretórios e arquivos que queira proteger dentro deste diretório.

Apesar de ser considerado um arquivo especial pelo Apache, trata-se de um arquivo de texto comum, que pode ser editado em seu editor de textos preferido. Apenas lembre-se de que o nome do arquivo precisa ser *".htaccess"* - sempre em minúsculas, começando com um ponto (".") e sem extensão (nada de ".txt"). O "ponto na frente" deve ser familiar para usuários de sistemas *NIX, e indica arquivos ocultos nesse sistema. A falta de extensão costuma ser um entrave para usuários Windows, que criam arquivos de texto e, dependendo da configuração de seu sistema, não conseguem ver a extensão do arquivo. Nesse caso, procure sempre desmarcar a opção *"Ocultar extensões de tipos de arquivos conhecidos"* nas preferências de seu sistema.

4.1 Criando login e senha de acesso

Antes de mais nada precisamos criar os pares usuário/senha que terão acesso à nossa área restrita. Para isso usaremos o utilitário *"htpasswd"*, que vem junto com o Apache e cria arquivos de senhas criptografadas em um formato que o servidor entende. Aqui iremos criar senhas para os usuários *monica e cebolinha*:

- Primeiro, criamos um diretório para guardar o arquivo de senhas e entramos nele. Em nosso exemplo, vamos criar o diretório *"auth"* dentro do diretório *"/var/www"*.

```
$ mkdir /var/www/auth
$ cd /var/www/auth
```

- Agora, usamos o *htpasswd* para criar os usuários e suas senhas, no arquivo que chamaremos de "acesso".

```
$ htpasswd -c -m acesso monica
New password:
Re-type new password:
Adding password for user monica

$ htpasswd  -m acesso cebolinha
New password:
Re-type new password:
Adding password for user cebolinha
```

Note que o parâmetro *"-c"* foi utilizado para criar o arquivo chamado *acesso*, que não existia até então. Como este arquivo foi criado para conter o primeiro usuário (monica), o parâmetro *"-c"* não deve ser utilizado nas inclusões subsequentes. A opção *"-m"* força a utilização de senhas criptografadas com o método MD5. Caso ela seja omitida, o sistema utilizará a função CRYPT. É possível ainda, ao invés dos algoritmos anteriores, forçar a codificação por SHA através do parâmetro *"-s"*. Note ainda que o utilitário *"htpasswd"* pode não estar em sua variável *PATH* e, nesse caso, devemos especificar o caminho completo onde ele se encontra – normalmente no mesmo prefixo que o Apache. Ou seja, se o servidor Apache foi instalado no diretório */var/www*, o utilitário *htpasswd* provavelmente se encontra em */var/www/bin/htpasswd*.

O resultado que temos é o arquivo */var/www/auth/acesso* contendo os usuários que terão acesso às áreas restritas, seguidos por suas respectivas senhas criptografadas:

```
monica:$apr1$.mZaf/..$5.mktXVjPifJvEJNtduiB1
cebolinha:$apr1$F067i...$auIBe0uyb1DSuEUJgQv011
```

Os valores da senha vão naturalmente mudar de acordo com as escolhas de seus usuários e senhas.

4.2 Restringindo acesso a usuários

Agora que o arquivo de usuários e senhas foi criado, vamos enfim criar o arquivo *.htaccess* que irá utilizá-lo. Nosso arquivo conterá as seguintes linhas:

```
AuthName "Acesso Restrito"
AuthType Basic
AuthUserFile /var/www/auth/acesso
require valid-user
```

Vejamos agora o que cada uma delas faz:

AuthName: O nome que aparece como mensagem de Login.

AuthType: Tipo de autenticação. Pode ser Basic ou Digest.

AuthUserFile: Onde está o arquivo de usuários e senhas que nós criamos no caso /var/www/auth/acesso.

require valid-user: O que o Apache precisa para validar o acesso. Neste caso indicamos que um usuário válido é necessário para acessar a página. Poderiamos ainda colocar a opção "require user" para sermos mais restritos ainda e permitir acesso apenas a determinados usuários, ainda que existam outros dentro do arquivo */var/www/auth/acesso*.

Por exemplo, para que apenas o cebolinha possa acessar o conteúdo do site, ainda que a monica possua uma entrada válida no arquivo de senhas, colocaríamos a diretiva *"require"* da seguinte maneira em nosso arquivo *.htaccess*:

```
AuthName "Acesso Restrito"
AuthType Basic
AuthUserFile /var/www/auth/acesso
require user cebolinha
```

Mais usuários podem ser incluídos na lista, bastando que sejam separados por espaços em branco.

Depois de criado o arquivo *.htaccess*, salve-o e pronto. Agora, sempre que alguém acessar a URL restrita, o servidor irá verificar este arquivo e solicitar no navegador um usuário e senha de acesso.

4.3 Restringindo acesso a grupos

Às vezes um sistema tem muitos usuários, muitas áreas com restrições diferentes ou ambas, e seria mais conveniente simplesmente separar os usuários em grupos do que classificar seus acessos individualmente. Assim, temos uma lista única com grupos de usuários, e em nosso arquivo de acesso precisamos nos preocupar apenas com que grupos podem acessar que áreas de nosso site.

Primeiro teremos que criar um arquivo de texto com os nomes dos grupos. Use o seu editor preferido e crie um arquivo chamado "/var/www/auth/grupos", por exemplo. A sintaxe é simples e consiste no nome do grupo seguido pelo caractere de dois pontos (":") e uma lista de nomes de usuários pertencentes ao grupo, separadas por espaços em branco:

```
conspiradores : cebolinha cascao
turma: monica cebolinha magali cascao
```

Para que estes grupos possuam acesso ao nosso servidor, teremos que modificar o arquivo *.htaccess* indicando a localização do arquivo de grupos e qual grupo deverá ganhar acesso. Em nosso exemplo, para que a área denomidada "Planos Infaliveis" dê acesso apenas aos membros do grupo "conspiradores", nosso arquivo ficaria da seguinte maneira:

```
AuthName "Planos Infaliveis"
AuthType Basic
AuthUserFile /var/www/auth/acesso
AuthGroupFile /var/www/auth/grupos
require group conspiradores
```

A linha *"AuthGroupFile"* indica para o servidor onde está o arquivo de grupos, neste caso em *"var/www/auth/grupos"*. A linha *"require group"* está dizendo que só libera acesso aos membros do grupo conspiradores.

4.4 Restringindo acesso a IP's

Podemos ainda restringir o acesso pelo número de IP do usuário. Isso é um aumento na segurança caso saibamos exatamente quais máquinas ou sub-redes serão utilizadas pelo usuário.

Para isso pode-se aplicar regras do tipo *"deny,allow"* do Apache dentro do *.htaccess*.

Veja o exemplo abaixo:

```
Order deny,allow

allow from 192.168.0.0
deny from all
```

allow from 192.168.0.0 : reserva acesso somente a rede 192.168.0.0

deny from all : Proibe acesso a todos os outros IP's.

A segurança é aumentada ainda mais quando, além da restrição por IP, incluímos a autenticação por senha demonstrada anteriormente.

Nosso exemplo fica assim:

```
Order deny,allow

AuthName "Acesso Restrito"
AuthType Basic
AuthUserFile /var/www/auth/acesso
require valid-user

allow from 192.168.0.0
deny from all
```

Neste exemplo o servidor libera acesso direto a quem estiver na rede 192.168.0.0 e pede senha de acesso aos demais usuários cadastrados no arquivo */var/www/auth/acesso*.

Lembre-se de sempre reiniciar o Apache após alterar alguma coisa em seu arquivo de conifuração. Isso pode ser feito com o comando:

```
$ apachectl restart
```

5.Conclusão

Arquivos *.htaccess* são muito úteis para a implementação de politícas de controle de acesso em servidores Apache. Com eles protegemos diretórios e arquivos contendo informações sensiveis ou que simplesmente não desejamos tornar públicas. Note que neste artigo cobrimos apenas algumas das inúmeras configurações que possui o arquivo *.htaccess*. Aprender todas é uma questão de tempo e prática, e vai de acordo com as necessidades de cada ambiente.

6.Bibliografia :

Apache http server project
http://httpd.apache.org/

Gleydson Mazioli da Silva - Fonte de Estudo e Apredinzado Linux avançado
http://focalinux.cipsga.org.br/guia/avancado

Netcraft: Web Server Survey Archives
http://news.netcraft.com/archives/web_server_survey.html