Segurança para o Joomla

Requisitos de segurança

- Servidor de hospedagem confiável
- SSL para site e admin
- Configurações do Joomla e do PHP
- Senhas fortes
- Evite ao máximo instalar extensões de terceiros e se instalar seja criterioso
- Extensões para reforçar a segurança
- Backup regular do site
- Atualizações do Joomla e das extensões de terceiros
- Faxina removendo extensões não usadas
- Cuidado com extensões de terceiros (sempre visite antes o site de Extensões Vulneráveis (https://vel.joomla.org/live-vel))
- Sanear seu desktop
- Não use ftp para transferência de arquivos. Prefira o cpanel
- Cuidado com as permissões de arquivos no sistema de arquivos

Checklist de Segurança para Joomla

- Se possível/viável escolher a melhor hospedagem do mercado, não a mais barata;
- Utilizar sempre a última versão do CMS e das extensões;
- Efetue um backup completo de todos os arquivos e do banco e restaure localmente
- Efetuar backup completo com frequência, especialmente antes de instalar novas extensões ou efetuar alterações como adição de conteúdo
- Ativar URLs amigáveis e mod_rewrite
- Mover configuration.php para fora do public_html, usando:
require_once(dirname(__FILE__) . '/../../portal.cfg');
- Bloquear cadastro de usuários pelo site caso não tenha necessidade: Configuração Global - Sistema - Permitir Cadastro de Usuários - Não
- Alterar metatags em Configuração Global - Configurações de Meta Dados (Trocar Joomla por outra palavra)
- Adicionar para a tag <head> do template (para ocultar Joomla na origem do código HTML), no início do index.php:
 <?php $this->setGenerator('Ribafs - Desenvolvimento Web'); ?>
 ou
 <?php $this->setGenerator(''); ?>

- Faça sempre o download do Joomla do site oficial - http://joomla.org
- Cheque o hash MD5 do arquivo baixado:
md5sum Joomla_3.7.5-Stable-Full_Package.zip
bd67cb02627e60bffef5e3b4ba3b2ece Joomla_3.7.5-Stable-Full_Package.zip
- Algumas extensões úteis:
    Firebug/Inspetor

- Instalar os principais navegadores para testar o site:
  Firefox, Chrome, Internet Explorer, Opera, Safari
- Mantenha os arquivos de configuração, logs e os diretórios de upload (repositórios de documentos, imagens e chache) fora do public_html.
- Remover desnecessários:
Arquivos
Extensões (se não precisa remova e não simplesmente desabilite. Quando precisar instale)
- Sempre antes de instalar novas extensões:
  - faça um backup completo do site e instale localmente
  - Verifique se a extensão é confiável em:
    https://docs.joomla.org/Archived:Vulnerable_Extensions_List
  - Faça o download do site do criador
  - Teste bastante localmente e somente então envie para o servidor
  - Evite instalar extensões que tenham código criptografado
- Sempre que possível evite hospedar seu site em servidores compartilhados
- Use um servidor de SSL, pelo menos para o administrator
- Use o .htaccess
- Atualize para a versão 3 e última do Joomla

Ferramentas para melhorar a segurança

Usar a ferramenta joomscan

Um bom tutorial
http://www.100security.com.br/joomscan/

Download
https://github.com/rezasp/joomscan

No Linux Mint 18.1 instalar antes
sudo apt-get install libswitch-perl

- Descompactar e acessar a pasta
- Atualizar
./joomscan.pl update

Checar a atualização
svn co https://joomscan.svn.sourceforge.net/svnroot/joomscan joomscan

Varrer site procurando vulnerabilidades
./joomscan.pl -u http://www.joomla.org

Relação de extensões e ferramentas que reforçam a segurança

- joomlascan - https://github.com/rezasp/joomscan
- AdminTools - https://www.akeebabackup.com/products/admin-tools.html
- Plugin osolcapcha - http://www.outsource-online.net/osol-captcha-for-joomla.html
- com_encrypt - http://www.ratmilwebsolutions.com/category/10-encryption-configuration.html
- jHackGuard - https://www.siteground.com/joomla-hosting/joomla-extensions/ver1.5/jhack.htm
- jadmin_bruteforceprotection - https://www.siteguarding.com/en/website-extensions
- jAdminProtection - https://www.siteguarding.com/en/website-extensions
- jGraphicalCaptchaProtection - https://www.siteguarding.com/en/website-extensions
- Plugin osolcapcha - http://www.outsource-online.net/osol-captcha-for-joomla.html
- OSOLCaptcha - https://github.com/osolgithub/OSOLCaptcha4Joomla3
- SimpleBackup - https://github.com/ribafs/com_simplebackup
- AdminExile - https://www.richeyweb.com/software/joomla/plugins/1-adminexile
- SecurityCheck - https://securitycheck.protegetuordenador.com/downloads/securitycheck-j3x/securitycheck-j3x-2-8-21
- Brute Force Stop - https://extensions.joomla.org/extensions/extension/access-a-security/site-security/brute-force-stop/
- AskMyAdmin - https://extensions.joomla.org/extensions/extension/access-a-security/site-security/askmyadmin/
- https://geekflare.com/security-extensions-to-protect-joomla-website/
- https://extensions.joomla.org/extensions/extension/access-a-security/site-security/centrora-security/
- https://www.incapsula.com/joomla-extension/joomla-plugin.html
- https://www.siteguarding.com/en/antivirus-website-protection-for-joomla

Free scanner para sites online
https://www.siteguarding.com/ - bom relatório com recomendações

Monitorando sites
https://geekflare.com/monitor-website-uptime/

Segurança na Web

Alterar permissões de arquivos:
Alterar todos os arquivos para 644 e todas as pastas para 755 com:
find . -type f -exec chmod 644 {} \;
find . -type d -exec chmod 755 {} \;

Depois criar algumas exceções...
configuration.php – 400
index.php do site – 400

index.php do template padrão – 400
Permissões de pastas:
includes e libraries – 500

Adicionar ao .htaccess:

```
# Block out any script trying to set a mosConfig value through the URL
RewriteCond %{QUERY_STRING} mosConfig_[a-zA-Z_]{1,21}(=|\%3D) [OR]

# Block out any script trying to base64_encode crap to send via URL
RewriteCond %{QUERY_STRING} base64_encode.*\(.*\) [OR]
# Block out any script that includes a <script> tag in URL
RewriteCond %{QUERY_STRING} (\<|%3C).*script.*(\>|%3E) [NC,OR]
# Block out any script trying to set a PHP GLOBALS variable via URL
RewriteCond %{QUERY_STRING} GLOBALS(=|\[|\%[0-9A-Z]{0,2}) [OR]
# Block out any script trying to modify a _REQUEST variable via URL
RewriteCond %{QUERY_STRING} _REQUEST(=|\[|\%[0-9A-Z]{0,2})
# Send all blocked request to homepage with 403 Forbidden error!
RewriteRule ^(.*)$ index.php [F,L]
```

Lembre que:
O Joomla possui uma equipe que em 4 horas consegue lançar uma versão estável do produto após uma invasão.
A maioria dos ataques ocorre pelo fato dos arquivos estarem com 777 ou usuário instalou componentes "não confiáveis".
Existe o Security Strike no Joomla! que cuida somente deste assunto
https://docs.joomla.org/Security_Strike_Team
https://developer.joomla.org/security-centre.html
https://volunteers.joomla.org/teams/security-strike-team
Para verificar sites que foram hackeados/defaced:
http://www.zone-h.org/archive?zh=1

Componente para criptografar senhas
Dá para notar seu trabalho.
Logo após digitar a senha e teclar Enter ou clicar em Acessar observe que ele enche a caixa da senha com bolinhas, mostrando que ele está enviando algo diferente do que digitamos.
O componente com_encrypt requer o módulo bcmath do php.

Sempre que o usuário fizer login a senha será criptografada antes de ser enviada para o servidor.
Ao chegar ao servidor será descriptografada.

O mesmo autor do componente criou vários plugins para outros módulos e extensões de terceiros:
http://www.ratmilwebsolutions.com/category/4-encryption-configuration-plugins.html

Opcionalmente podemos gerar uma nova chave de criptografia, mas talvez não seja necessário pois uma é gerada automaticamente a cada 180 dias.
Também podemos alterar a frequência de geração de chaves e seu tamanho.
O componente criptografa a senha de login do form de login do administrator por padrão e já vem com vários outros recursos marcados por padrão:
Back-end login, Back-end edit profile, Back-end edit profile repeat password, Update RSA private KEY, Joomla off-line login, Front-end login module, Front-end login, Create account, Create account repeat password, Edit profile, Edit profile repeat password, Reset password e Reset password confirm

Download
http://www.ratmilwebsolutions.com/category/10-encryption-configuration.html

Ajuda
http://www.ratmilwebsolutions.com/documentation/47-encryptioncomponenthelp.html

Segurança no Joomla
Dicas de segurança no joomla.
Muitas pessoas utilizam o CMS Joomla, no entanto a maior parte destas "esquece-se" do fator segurança nos seus sites. Existem pequenos pormenores extremamente fáceis de implementar que aumentarão consideravelmente a segurança do teu site Joomla.

Desligar os relatórios de erro
Um deles é desligar os relatórios de erros, os relatórios de erros além de diminuírem a velocidade do site indicarão também ao "hackers" falhas na segurança deste. Isto pode ser desativado em 'Configuração Geral -> Sistema'.
Depois de desativada esta função não te será permitido visualizar os erros gerados pelo Joomla, o que é uma coisa boa uma vez que o utilizador comum não os vê (o que não era muito profissional) e os hackers não podem forçar erros de forma a descobrirem métodos de comprometer o sistema.

Utilizar um componente SEF

A maioria dos hackers utilizam o comando 'inurl:' do Google para procurarem por falhas em websites. Uma boa solução para contrariar este potencial risco é instalar um componente que reescreva os Url, aconselho o SH404SEF ou o Artio-JoomSef.

O componente SEF irá trazer-lhe também bastantes vantagens a nível de SEO (rank mais elevado aos "olhos" do Google).

Mover o ficheiro configuration.php para fora da raiz.
Mova simplesmente o ficheiro de configuração para qualquer pasta que você queira dentro do site e atribua-lhe um novo nome. No exemplo utilizei 'joom.conf'.
Crie um novo ficheiro de configuração na raiz com o nome de configuration.php contendo o seguinte código:

```
<?php
require( dirname( __FILE__ ) . '/../joom.conf' );
?>
```

Realize backups regulares

Esta tarefa pode ser feita através do Cpanel de qualquer conta de alojamento, no entanto existem também alguns componente muito bons que realizam esta tarefa. O meu favorito é o JoomlaPack. Um backup semanal caso atualize o seu site regularmente é uma boa opção, ou então backups mensais.

Não mostrar que versões das extensões utiliza

Em primeiro lugar qualquer admin de um website deveria ter uma lista de todas as extensões que utiliza e fazer o update a estas quando sai-se uma nova versão. No entanto todos nos sabemos que o tempo não chega para tudo e muitas vezes fazer um update a uma extensão pode ser um bocado moroso. É então boa política remover a versão da extensão que utiliza a quando da instalação desta, isto pode ser feito editando os ficheiros da extensão com o notepad por exemplo.

Segunda parte

Um site em Joomla! é muito mais do que instalá-lo no servidor, mover alguns módulos de posição, instalar componentes, plugins e pronto! Já temos um site completo, feito em três dias e podemos ganhar mais de mil reais do nosso cliente.

Sinceramente, pessoal, o Joomla é tão complicado de usar quanto se programar um site do zero. Claro que você não terá mais a necessidade de digitar todas as linhas de código, mas eventuais alterações serão necessárias e é importante saber o que, onde e por que está sendo feita aquela mudança.

Além disso, a segurança é muito importante. Hoje existe uma gama enorme de componentes e módulos para Joomla, mas antes de usarem, perguntem-se: "este componente é seguro?". A maioria das invasões em sites Joomla! é feita através do próprio cms mal configurado ou de seus componentes desatualizados. Experiência própria: é muito mais difícil você contornar uma invasão do que prevenir que ela não aconteça.

Trabalho com o Joomla há mais de três anos, desde a versão 1.0.12, e desde lá já aprendi muito, tomei muito na cabeça e hoje me viro tranquilo, tanto é que tenho mais de 20 clientes em minha região e todos utilizam o Joomla!, mas a cada nova atualização de componentes, preciso dar atenção a estes sites, pois é a segurança dos dados e informações dos mesmos que estão em jogo.

Por isso minha gente, tenho um sério conselho a dar a vocês: Estudem!

Estudem muito o Joomla, pesquisem sobre servidores web (apache), sobre dicas de segurança no PHP, informações sobre servidores de e-mail, segurança de arquivo, permissões de acesso a pastas e arquivos, etc...

Mostrei apenas o caminho das pedras, agora é Google na veia e tempo e disciplina para estudar. Hoje existem mil vezes mais materiais sobre esse assunto do que quando comecei. Inclusive a maioria mais detalhada e em português, no "meu tempo" os bons artigos e tutoriais eram em inglês.

Este e-mail foi escrito como um alerta aos desavisados, para não saírem por ai usando o Joomla! sem considerar o uso de medidas sobre segurança.

Isso evitará os seus sites de serem invadidos e assim o indivíduo não vai sair por ai xingando todo mundo em qualquer fórum destinado ao Joomla!, falando mal do sistema para qualquer um que aparecer, alegando que "não é seguro".
Quem faz o Joomla ser seguro é você".
Escrito por Roberto Jonikaites para o Yahoogrupos – Curso de Design para Joomla! De Bruno Ávila.

Este artigo foi encontrado no site abaixo, mas não mais o encontrei em minha última tentativa de visita: http://www.joomlarj.com.br/site/index.php?option=com_content&view=article&id=26:seguranca-no-joomla-parte-2&catid=15:seguranca-no-joomla&Itemid=15

Segurança e phpini

Adicionar diretamente ao php.ini, para o caso de se ter acesso ao php.ini no servidor.

session.save_path = "/var/www/html/tmp"
cgi.force_redirect = 1
allow_url_fopen= 0
display_errors = 0
expose_php = 0
magic_quotes_gpc = 0

memory_limit = 8388608
#open_basedir = 1
post_max_size = 262144
upload_max_filesize = 262144
upload_tmp_dir = "/var/www/html/tmp"
disable_functions = proc_open, popen, disk_free_space, set_time_limit, leak, tempfile, exec, system, shell_exec, passthru, curl_exec, curl_multi_exec, parse_ini_file, show_source, apache_get_modules,apache_get_version,apache_getenv,apache_note, apache_setenv, disk_free_space, diskfreespace, dl, highlight_file, ini_alter, ini_restore, openlog, proc_nice, symlink, phpinfo

Adicionar ao configuration.php, para o caso de não ter acesso direto ao php.ini

```
    ini_set('session.save_path', '/var/www/html/tmp');
    ini_set('cgi.force_redirect', 1);
    ini_set('allow_url_fopen', 0);
    ini_set('display_errors', 0);
    ini_set('allow_url_include', 0);
    ini_set('expose_php', 0);
    ini_set('magic_quotes_gpc', 0);
    ini_set('post_max_size', '262144'); // Ajustar a gosto
    ini_set('upload_max_filesize', '262144'); // Ajustar a gosto
    ini_set('upload_tmp_dir','/var/www/html/tmp');
    // Funções a serem desabilitadas
    $disfunctions = 'proc_open, popen, disk_free_space, set_time_limit, leak, tempfile, exec, system, shell_exec, passthru, curl_exec, curl_multi_exec, parse_ini_file, show_source, apache_get_modules, apache_get_version, apache_getenv, apache_note, apache_setenv, disk_free_space, diskfreespace, dl, highlight_file, ini_alter, ini_restore, openlog, proc_nice, symlink, phpinfo';
    ini_set('disable_functions', $disfunctions);
```

Verificar existência e as versões no seu servidor:

zend_extension=/usr/local/php52/lib/php/extensions/ioncube.so
zend_extension_manager.optimizer=/usr/local/Zend/lib/Optimizer-3.3.3
zend_extension_manager.optimizer_ts=/usr/local/Zend/lib/Optimizer_TS-3.3.3
zend_optimizer.version=3.3.3
zend_extension=/usr/local/Zend/lib/ZendExtensionManager.so
zend_extension_ts=/usr/local/Zend/lib/ZendExtensionManager_TS.so

Vários dos recursos acima você precisará confirmar com o suporte do seu servidor para ver se estão disponíveis.

Recomendações sobre Segurança

Recomendações sobre segurança

- Usar senhas de no mínimo 6 caracteres. Quanto mais melhor, mas de 8 a 10 tá bom.
- Misturar caracteres alfabéticos maiúsculas, minúsculas, números e caracteres especiais como:
 -, _, *, $, !, %
- Não use senhas fáceis como data de nascimento, número de identidade, nomes de filhos e cônjuges.
- Procure não usar palavras do mundo real
- Pense num episódio que apenas você conhece ou lembra e forme uma frase com suas iniciais
- Crie senhas posicionais, por exemplo: primeira letra da última fila, primeira letra da primeira fila, última letra da última fila, última letra da primeira fila e assim por diante.
- Mesmo que ilógicas as senhas devem ser, para você, de fácil memorização, pois você deve evitar anotar as senhas
- Evite usar a mesma senha para todos os seus acessos
- Atualize com uma certa frequência suas senhas

Evitar o uso do ftp para transferir/baixar arquivos para/do servidor, pois ele envia seus dados (login e senha) em texto claro.

Se precisar usar o ftp use o FileZilla, que usa o sftp.

https://filezilla-project.org/

Instalação:

Debian e derivados
sudo apt-get install filezilla

Windows 64
https://filezilla-project.org/download.php?platform=win64

Referências sobre Segurança

https://docs.joomla.org/Security
https://extensions.joomla.org/category/access-a-security/site-security/
https://docs.joomla.org/Security_Checklist
https://developer.joomla.org/security.html
https://www.siteground.com/tutorials/joomla/joomla-security.htm
https://geekflare.com/joomla-security/
https://www.keycdn.com/blog/joomla-security/

https://extensions.joomla.org/extensions/extension/communication/live-support/onwebchat/

Configurando o .htaccess

Referências
https://httpd.apache.org/docs/current/pt-br/howto/htaccess.html
https://my.justhost.com/cgi/help/htaccess
http://www.devin.com.br/htaccess/

No geral, você nunca deve usar arquivos .htaccess a não ser que você não tenha acesso ao arquivo de configuração principal do Apache.

Arquivos .htaccess devem ser usados em casos onde os provedores de conteúdo do site precisem fazer mudanças na configuração do servidor por diretório, mas não tem acesso root ao sistema do servidor. Caso o administrador do servidor não esteja disposto a fazer mudanças frequentes nas configurações do servidor, é desejável permitir que os usuários possam fazer essas mudanças através de arquivos .htaccess eles mesmos. Isso é particularmente verdade, por exemplo, em casos onde provedores estão fornecendo múltiplos sites para usuários em apenas uma máquina, e querem que seus usuários possam alterar suas configurações.

É o caso dos servidores de hospedagem compartilhada.

No entanto, de modo geral, o uso de arquivos .htaccess deve ser evitado quando possível. Quaisquer configurações que você considerar acrescentar em um arquivo .htaccess, podem ser efetivamente colocadas em uma seção <Directory> no arquivo principal de configuração de seu servidor.

Existem duas razões principais para evitar o uso de arquivos .htaccess.

A primeira delas é a performance. Quando AllowOverride é configurado para permitir o uso de arquivos .htaccess, o Apache procura em todos diretórios por arquivos .htaccess.

A segunda consideração é relativa à segurança. Você está permitindo que os usuários modifiquem as configurações do servidor, o que pode resultar em mudanças que podem fugir ao seu controle. Considere com cuidado se você quer ou não dar aos seus usuários esses privilégios. Note também que dar aos usuários menos privilégios que eles precisam, acarreta em pedidos de suporte técnico adicionais.

O uso de arquivos .**htaccess** pode ser totalmente **desabilitado**, ajustando a diretriz AllowOverride na seção <Directory> para none:

AllowOverride None

Para habilitar:
AllowOverride All

Definir os arquivos de índice

.htaccess

DirectoryIndex index.php index.html

Criando páginas de erro customizadas:

ErrorDocument 404 /404.html

Páginas de erro:

- 401 - Authorization Required
- 400 - Bad request
- 403 – Forbidden
- 404 - Wrong page
- 500 - Internal Server Error

ErrorDocument 401 /erros/falhaautorizacao.html
ErrorDocument 404 /erros/naoencontrado.html
ErrorDocument 403 /erros/acessonegado.html
ErrorDocument 500 /erros/errointerno.html

Permitir que arquivos de diretório sejam listados:
Options All +Indexes

Impedir a listagem de diretório:
Options ExecCGI Includes IncludesNOEXEC SymLinksIfOwnerMatch -Indexes

ou
No directory listings
<IfModule autoindex>
  IndexIgnore *
</IfModule>

Bloquear certos IPs:

- order allow,deny
- deny from 123.123.123.123 #specify a specific address
- deny from 123.123.123.123/30 #specify a subnet range
- deny from 123.123.* #specify an IP address wildcard

allow from all

Permitir certos IPs:

order deny,allow
allow from 123.123.123.123 #specify a specific address
allow from 123.123.123.123/30 #specify a subnet range
allow from 123.123.* #specify an IP address wildcard
deny from all

Redirecionar de um arquivo para outro:

Redirect /redirect_from.html http://www.newsite.com/folder/redirect_to.html

Redirecionar de uma pasta para outra:

Redirect /redirect_from http://www.newsite.com/redirect_to

 # Deixa a Intranet acessar

Order allow,deny
allow from 192.168.0.
deny from all

 # Deixa todo mundo acessar, menos o IP 192.168.0.25

Order deny,allow
deny from 192.168.0.25
allow from all

Redirecionar páginas de erro 404 para a index do site:

Supondo que o site está na pasta /joomla

1) Criar no raiz a pasta

erros

2) Dentro da pasta criar o arquivo 404.php contendo:

<?php
header('location: /joomla/index.php');
3) Criar o arquivo .htaccess na pasta do site contendo:

ErrorDocument 404 /erros/404.php

Remover os arquivos:
README.txt
LICENSE.txt
web.config.txt - só para windows
robots.txt

Renomear:

htaccess.txt para .htaccess

Rodar o joomscan:
sudo apt install libswitch-perl
https://github.com/rezasp/joomscan

Atualizar
svn co https://joomscan.svn.sourceforge.net/svnroot/joomscan joomscan

joomscan.pl update
joomscan.pl check
joomlscan download

./joomscan.pl -pv -u http://localhost/joomla | less
./joomscan.pl -pv -u http://localhost/joomla > relatorio.txt
joomscan.pl -pv -u victim.com -x localhost:8080

Use sempre a última versão estável do Joomla
Mantenha o Joomla e todas as extensões de terceiros atualizados
Evite instalar templates e outras extensões piratas, pois podem conter softwares maliciosos que venham a comprometer a sua segurança

```
Usage:  joomscan.pl -u <string> -x proxy:port
        -u <string>      = joomla Url

        ==Optional==
        -x <string:int>  = proXy to tunnel
        -c <string>      = cookie (name=value;)
        -g "<string>"   = desired useraGent string within "
        -nv             = No Version fingerprinting check
        -nf             = No Firewall detection check
        -nvf/-nfv        = No version+firewall check
        -pe             = Poking version only
                         (and Exit the scanner)
        -ot             = Output to Text file (target-joexploit.txt)
        -oh             = Output to Html file (target-joexploit.htm)
        -vu             = Verbose (output every Url scan)
        -sp             = Show completed Percentage

Example:
        joomscan.pl -pv -u victim.com -x localhost:8080

Check:   joomscan.pl check
        This option will check if the scanner update is available or not.
```

Update: joomscan.pl update
This option will check and update the local database if newer
version is available.

Download: joomscan.pl download
- Download the scanner latest version as a single zip file - joomscan-latest.zip.

Defense: joomscan.pl defense
This option will give you a defensive note.

About: joomscan.pl story
This option will give you a short story about joomscan.

Read: joomscan.pl read DOCFILE
DOCFILE – changelog,release_note,readme,credits,faq,owasp_project

Nunca confie nos usuários
Sempre sanitize inputs dos usuários
Use funções nativas do Joomla para receber inputs

Critérios ao criar usuário. Limitar seus privilégios adequadamente.

Usar bons recursos para melhorar a segurança:
boa hospedagem
.htaccess (proteger o administrator para acesso somente do seu IP)
extensões
proteger o administrator
ver categoria segurança no ribafs.org
verificar acessos suspeitos nos softwares de estatísticas do servidor
bloquear certos ips
em todos os diretórios de cada extensão instalada manter um index.html vazio para evitar listagem de diretórios
Em todas as extensões usar o JEXEC para impedir acesso direto aos arquivos php
Reforçar com uma segunda camada de senha (do Apache) no administrator
Se possível usar SSL em todo o site ou pelo menos no administrator

A maioria dos ataques são resultantes da utilização indiscriminada de templates piratas, provedores ruins, versões desatualizadas e permissões equivocadas para os diretórios e arquivos.

Poderiam ser facilmente evitados com medidas básicas relativas ao uso do CMS, dentre elas:
 manter a versão do CMS atualizada;
 hospedar o projeto em servidores idôneos e bem configurados;
 utilizar url's amigáveis;
 templates comerciais, somente os adquiridos nos clubes de template; e
 utilizar extensões que potencializam a segurança da área de administração.

robot.txt
Por questões de segurança, recomendo a exclusão desse arquivo imediatamente após a instalação do Joomla e publicação do site em ambiente remoto.

Mais sobre segurança em para Joomla
http://ribafs.org/portal/joomla-3/seguranca.html