Guia Prático Sobre Segurança Do Joomla

O Joomla Como Ferramenta De Democratização Da Internet

O Joomla é um script muito rico em funcionalidades: ferramenta de vendas, angariação de contactos, comunicação com clientes e visitantes, adaptando-se às diferentes necessidades de milhões de empresas, habilitando-as a actuar no mercado da internet.

É também um **pilar da democratização da internet**, porque baixou substancialmente o preço da criação dum site empresarial. O bilhete de entrada na internet tornou-se **econômicamente acessível** para qualquer empresa.

Veja por exemplo o nosso tutorial, aplicável com as necessárias adaptações ao Joomla 2.5.x e 3.x.x :

Como Criar Um Site Em Joomla Para A Sua Empresa

Até foi criado um nicho de mercado, onde milhares de freelancers aprenderam sozinhos, lendo alguns tutoriais gratuitos na internet, a montar sites para empresas, usando o Joomla, e fornecem os seus serviços a empresas e a preços acessíveis.

Por tudo isto, **o Joomla é Muito Popular**. Já foi baixado mais de 35 milhões de vezes.

Buraco De Segurança No Joomla

Este contexto é perfeito para o surgimento dum cenário paralelo, mas adverso: um **buraco grande em termos de segurança**, por onde entram spammers, phishing sites, malware, hackers e todos aqueles que se perdem pelo lado negro da sociedade.

Que contexto permissivo é esse?

Como o preço de aquisição é quase grátis, as empresas que usam Joomla desvalorizam a manutenção do site, incluindo a respectiva segurança.

A própria responsabilidade parece diluída, porque parece que cai tudo do céu… Há uma expectativa irrealista que todos os passos seguintes à aquisição do site são grátis ou quase grátis.

Como é fácil aprender a montar um site em Joomla, os novos programadores pensam que não precisam de saber código e desleixam muito a segurança do script.

Como o Joomla é código aberto (open source) e usado por milhões, também é, por isso mesmo, um alvo privilegiado de hackers, que vasculham o código à procura de vulnerabilidades.

Importa saber que **as extensões são a parte fraca do código do Joomla em contraponto com o código Core**, dado que são contribuições de milhares de programadores da comunidade, com diferentes níveis de qualificação e experiência, que

expandem as funcionalidades do Joomla, mas multiplicam os vectores de ataque e, consequentemente, o número de portas e janelas que facilitam intrusões.

Solução De Segurança Fácil De Implementar

Se o Joomla é fácil de usar e se essa característica é um dos atrativos, então **a solução para o problema de falta de segurança também deve ser fácil de aplicar**. Não podemos sugerir ao usuário do Joomla que aprenda programação e procure no código do Joomla e no código das extensões usadas todas as vulnerabilidades de segurança que podem afetar uma qualquer instalação do Joomla.

Até porque aprender programação não seria suficiente. A maioria dos programadores não tem a competência necessária para criar código seguro. Seria necessário que o usuário aprendesse programação e fosse um programador excepcional.

E **qual o objetivo das medidas que vamos publicar de seguida**?

> Não vamos limpar o código do Joomla. Vamos apenas dificultar a tarefa dos hackers, reduzindo os vetores de ataque, de modo a prevenir 99% das violações de segurança.

Chega de conversa. Vamos direto às medidas que você deve implementar para proteger o seu Joomla.

Segurança Local

A primeira vulnerabilidade do Joomla é você ou qualquer usuário que aceda ao Joomla com poderes de administrador. A afirmação é um pouco dramática. Mas, **é muito importante que você cuide da segurança do computador que você usa para aceder à zona de administração do Joomla** ou à conta de hospedagem / alojamento web.

Mais ainda se você usa esse computador para aceder a home banking, ao paypal ou para guardar ou transmitir qualquer informação privada, valiosa e / ou sensível.

Quando lemos sobre **vulnerabilidades dos navegadores / browsers, do Flash, do Java, do Javascript** e outras, temos que agir em conformidade, sob pena de pagarmos o preço de contemplarmos a realidade, sem nos adaptarmos a ela.

A sugestão que faço é muito simples. Compre um PC, Mac, Portátil ou Tablet barato. Usado, se necessário. Instale o Ubuntu. Configure o navegador / browser com uma política de segurança restritiva em termos de Javascript e cookies. Não instale Flash ou Java. E **use esse computador seguro para aceder APENAS ao seu Joomla, à vossa hospedagem / alojamento web e eventualmente a homebanking, paypal e sites similares**.

O APENAS não é decorativo. Sem excepções!

Se optar por Ubuntu, não vai poder descarregar uma cópia ilegal do Dreamweaver? Não

instale nada ilegal neste computador seguro. **O software ilegal é uma plataforma privilegiada de distribuição de malware.**

Cópia De Segurança Semanal ou Diária

Acabei de escrever 731 palavras. Se ocorrer agora mesmo uma falha catastrófica no meu PC, tenho uma cópia da parte do artigo que já escrevi, com a excepção destas últimas palavras. **Quantas palavras** você já escreveu no seu Joomla? **Quanto tempo** já investiu? O design é personalizado? **Quantas extensões** já procurou e instalou?

> Você tem uma cópia do seu Joomla? Da semana passada? Do ano passado? Onde? É cópia de quê? Dos ficheiros e da base de dados? Sabe restaurar o seu Joomla a partir dessa cópia de segurança?

É recorrente lermos notícias de violações de segurança nos computadores do Pentagon, da NASA, da CIA. **Você acha que o seu Joomla é uma fortaleza à prova de hacker?** Se você está a ler este artigo, você não pensa isso, certo? Então, escreva 10 vezes num papel e cole no seu monitor:

“Eu preciso de fazer uma cópia de segurança semanal do meu Joomla.”

Publica diariamente e não pode perder vários dias de conteúdos? Faça uma cópia diária!

Não instale a versão gratuita do AkeebaBackup para depois deixar a cópia valiosa na vossa conta de hospedagem / alojamento web! A versão gratuita do AkeebaBackup não permite enviar a cópia para outro servidor ou para a vossa conta de email.

Esta história das cópias de segurança faz-me lembrar o desleixo com que as pessoas em geral tratam a saúde e o respectivo corpo. **Confiam que a morte é apenas amanhã. Mas, a morte é hoje mesmo.** O James Gandolfini (Tony Soprano) morreu há pouco tempo de ataque cardíaco. Tinha 51 anos.

Não tem tempo para fazer uma cópia diária ou semanal?

Instale o XCloner, para executar as cópias de segurança do seu Joomla. E o JPrc Cronjobs, para automatizar esse processo.

Configure o XCloner para enviar as cópias para a vossa conta de email ou para uma conta FTP noutro servidor.

Use o Jcron para agendar a execução das cópias de segurança. Na Task, seleccione Web Address e coloque o URL do ficheiro respectivo do XCloner:

```
http://www.oseudominio.com/administrator/components/com_cloner/cloner.cron.php
```

Configure para que as cópias sejam executadas durante a madrugada. **Não execute cópias de segurança durante o dia.** É uma tarefa muito intensiva em termos de recursos do servidor. É o mesmo que ouvir música no volume máximo durante a noite.

O servidor de alojamento partilhado é como uma comunidade. E você não quer, nem deve ser um vizinho abusivo… até porque isso terá consequências negativas no desempenho do servidor, o que acabará por afetar o tempo de acesso ao seu Joomla.

- Minutes – Seleccione 0, para correr aos 0 minutos da hora a seguir indicada.
- Hours – Indique 1 (ou 2 ou 3) para que o cron corra às 03:00 horas, hora do servidor.
- Days – Every day.
- Months – Every month.

Se optar por usar o XCloner, procure no código do ficheiro .htaccess master, a excepção que permite o acesso através ao Akeeba e substitua por esta regra:

```
RewriteRule
^administrator/components/com_xcloner-backupandrestore/cloner.cron\.php$ - [L]
```

Segurança Das Passwords

Não perca o seu tempo a seguir todas estas recomendações de segurança para depois **usar uma password como admin, 12345, qwerty ou igual ao seu domínio**. Isso é o mesmo que fechar a porta de casa e depois deixar a chave pendurada no alpendre à vista do mundo inteiro…

Siga estas regras para criar uma senha fácil de memorizar, mas que não é fácil de crackar.

- **Escolha uma palavra** para a vossa senha. No meu exemplo, vou usar a palavra silencio.
- **Seleccione um número** para a vossa senha. Por exemplo, o ano do nascimento do seu animal de estimação. O número que seleccionei foi o 1917.
- Decida que regra usar para **combinar a palavra com o número**. Por exemplo, pode alternar letras e números: s1i9l1e7ncio
- **Pense 1 caracter especial**. No meu exemplo, pensei o caracter +
- Decida uma segunda regra para **combinar o caracter com as letras e os números**. Por exemplo, colocar o caracter na 5ª posição e na 7 posição da senha: s1i9+l+1e7ncio
- E como precisamos de maiúsuclas e minúsculas, temos que decidir **quais as letras que vão ser maiúsculas**. Para simplificar, vamos optar pela primeira e última letra: S1i9+l+1e7nciO

Vamos criar uma segunda senha e com regras simples.

Palavra: margarida (nome duma flor)

Número: 13579 (números ímpar até 10)
Caracter especial: =
Regra 1: alternar letras e números
Regra 2: colocar caracter especial no fim
Regra 3: colocar maíuscula no início

M1a3r5g7a9rida=

Este método é uma adaptação do trabalho Simple Formula For Strong Passwords, do Bernie Thomas, publicado pelo SANS Institute.

Permissões De Ficheiros E Pastas

Qual o **PHP handler** que o servidor web usa para interpretar o código PHP? **Se usa o mod_php / DSO** (Dynamic Shared Object), então você vai precisar de usar permissões 777 nalgumas pastas para que o seu Joomla funcione normalmente, especialmente a funcionalidade de upload de media / imagens.

Essa notícia é má. **Você não deve usar permissões 777. É inseguro.**

Pergunte sempre ao fornecedor de hospedagem / alojamento web qual o PHP handler que usam no servidor web? **Dê preferência a suPHP ou FastCGI**. Rejeite mod_php (DSO).

A recomendação habitual é para você usar permissões 644 nos ficheiros e 755 nas pastas. Mas, eu vou sugerir uma abordagem mais rigorosa:

Mude as permissões dos ficheiros PHP para 400 e das pastas para 711.

Esta recomendação é para o caso do servidor web usar como PHP handler o suPHP ou o FastCGI. Caso seja mod_php, não tenho qualquer sugestão em termos de permissões.

Se tiver acesso shell, execute estes comandos, na mesma pasta onde estão os ficheiros index.php, configuration.php, CHANGELOG.php do seu Joomla. Em vez de descrever a pasta, indicando ficheiro concretos do Joomla, poderia indicar a web root ou document root do servidor web ou da conta de hospedagem / alojamento web, mas este tutorial é um convite a que todos executem estas recomendações de segurança, mesmo aqueles que não sabem a web root ou a document root do servidor web:

```
find . -type f -name '*.php' -exec chmod 400 {} \;
```

```
find . -type d -exec chmod 711 {} \;
```

Se quiser mudar para um **nível de permissões mais permissivo**, execute:

```
find . -type f -exec chmod 644 {} \;
```

```
find . -type d -exec chmod 755 {} \;
```

Caso não tenha acesso shell, faça o upload dum ficheiro fechaporta.php para a pasta inicial do seu Joomla, onde estão os ficheiros index.php, configuration.php, CHANGELOG.php, para restringir as permissões. Coloque este código no ficheiro:

```
<?
shell_exec("find . -type d -exec chmod 711 {} \\;");
shell_exec("find . -type f -name '*.php' -exec chmod 400 {} \\;");
?>
```

E o upload dum ficheiro abreporta.php, para a mesma pasta, para colocar as permissões num nível mais permissivo. Coloque este código no ficheiro:

```
<?
shell_exec("find . -type d -exec chmod 755 {} \\;");
shell_exec("find . -type f -name '*.php' -exec chmod 644 {} \\;");
?>
```

Para executar qualquer um destes ficheiros, aceda ao ficheiro respetivo através do navegador / browser:

```
wwww.oseudominio.com/fechaporta.php
```

ou

```
wwww.oseudominio.com/abreporta.php
```

É importante que **apague os ficheiros depois de usá-los**. Ou seja, se fechar a porta, não deixe a chave na porta…

No ficheiro .htaccess mais abaixo, permitimos o acesso através do browser a estes 2 ficheiro fechaporta.php e abreporta.php. Quando não estiver a usá-los, faça uncomment dessas 2 linhas no código do ficheiro .htaccess master. E, quando precisar de usá-los, remova o # no início de cada linha.

```
# Assim, não é possível aceder aos ficheiros. Remova o # para poder aceder aos mesmos.
# Adicione o # para não permitir o acesso aos ficheiro.
#RewriteCond %{REQUEST_FILENAME} !/fechaporta\.php
```

```
#RewriteCond %{REQUEST_FILENAME} !/abreporta\.php
```

Experimente, teste e veja quais são as permissoes mais restritivas que pode usar no seu Joomla, sem prejudicar o respectivo funcionamento. Se precisar de alternar para um nível mais permissivo apenas quando está a instalar alguma extensão, utilize os ficheiros que disponibilizamos para alterar o nível de permissões.

Ficheiro .htaccess Master

O próprio Joomla recomenda este ficheiro .htaccess master, que deverá ser colocado na mesma pasta que os ficheiros index.php, configuration.php, CHANGELOG.php. O código que publicamos abaixo tem algumas alterações a esse ficheiro.

> É obrigatório usar este ficheiro, caso queria reduzir substancialmente os vetores de ataque ao seu Joomla.

Se encontrar uma dificuldade insuperável na implementação do ficheiro .htaccess master, publicado mais abaixo, **proteja no mínimo a pasta images**, dado que é o ponto mais fraco do Joomla, para onde quase sempre é feito o upload duma ou várias shell. Em vez do ficheiro .htaccess master, coloque um ficheiro .htaccess com o código seguinte, na pasta images:

```
AddHandler cgi-script .php .pl .py .jsp .asp .htm .shtml .sh .cgi

Options -ExecCGI
```

O código não vai impedir o upload duma shell. Mas, o hacker não vai poder aceder à shell, dado que, por causa deste código, **não é permitido a execução de ficheiros PHP, Perl, Python, Java, ASP, HTML, SHTML ou CGI na pasta images**.

Experimente também usar este ficheiro .htaccess nas **pastas media, cache e tmp**. Comece pela pasta images e teste e depois repita para cada pasta: adicionar + testar.

Vamos retornar ao ficheiro .htaccess master, que publico mais abaixo.

> Se der erro à primeira tentativa, faça uncomment (adicionar # no início da linha de código) sucessivamente de partes do código até encontrar a parte do código que está a causar esse erro. O uncomment desativa a linha de código respectiva.

Não se esqueça de **substituir o domínio exemplo.com pelo seu domínio**, no código do ficheiro.

Temos que agredecer aos respetivos autores, cujos nomes constam no próprio código. Devemos gratidão a cada um deles.

Algumas observações a considerar na aplicação do código:

- O RewriteCond e RewriteRule funcionam como causa e efeito. Se acontecer a

primeira, aplica-se a segunda.

- **O código duma condição do RewriteCond deve estar todo na mesma linha.** Procure no código qualquer quebra de linha indevida e corrija. Por exemplo, se encontrar o código duma condição RewriteCond numa linha e depois numa segunda linha e um RewriteCond ou o RewriteRule na terceira linha, remova a quebra de linha de modo que haja apenas uma linha de código RewriteCond.
- Se aparecer um **Internal Server Error**, depois de adicionar o master .htaccess file, experimente fazer uncomment a partes do código até encontrar o código respopnsável pelo erro.
- Veja esta parte do código, caso tenha problemas com plugins ou templates. Verifique também o nível de permissões dos ficheiros e pastas.

```
## Uncomment this line if you have extensions which require direct access
to their own

## custom index.php files. Note that this is UNSAFE and the developer
should be ashamed

## for being so lame, lazy and security unconscious.

# RewriteRule ^(components|modules|plugins|templates)/([^/]
+/)*(index\.php)?$ - [L]

## Uncomment the following line if your template requires direct access to
PHP files

## inside its directory, e.g. GZip compressed copies of its CSS files

# RewriteRule ^templates/([^/]+/)*([^/.]+\.)+php$ - [L]

RewriteRule ^(components|modules|plugins|templates)/ - [F]
```

Código Do Ficheiro .htaccess Master

```
##############################################################################

## The Master .htaccess

## Version 2.5 (proposed) - May 16th, 2011

## Nicholas K. Dionysopoulos

## Lead Developer, AkeebaBackup.com


RewriteEngine On


# RewriteBase /


########## Begin - No directory listings

## Note: +FollowSymlinks may cause problems and you might have to remove it
```

```
IndexIgnore *
Options +FollowSymLinks All -Indexes
########## End - No directory listings

########## Begin - File execution order, by Komra.de
DirectoryIndex index.php index.html
########## End - File execution order

ServerSignature Off

########## Begin - Common hacking tools and bandwidth hoggers block
## By SigSiu.net and @nikosdion.
# This line also disables Akeeba Remote Control 2.5 and earlier
SetEnvIf user-agent "Indy Library" stayout=1
# WARNING: Disabling wget will also block the most common method for
# running CRON jobs. Remove if you have issues with CRON jobs.
SetEnvIf user-agent "Wget" stayout=1
# The following rules are for bandwidth-hogging download tools
SetEnvIf user-agent "libwww-perl" stayout=1
SetEnvIf user-agent "Download Demon" stayout=1
SetEnvIf user-agent "GetRight" stayout=1
SetEnvIf user-agent "GetWeb!" stayout=1
SetEnvIf user-agent "Go!Zilla" stayout=1
SetEnvIf user-agent "Go-Ahead-Got-It" stayout=1
SetEnvIf user-agent "GrabNet" stayout=1
SetEnvIf user-agent "TurnitinBot" stayout=1
# This line denies access to all of the above tools
deny from env=stayout
########## End - Common hacking tools and bandwidth hoggers block

########## Begin - Automatic compression of resources

########## Begin - Add optional bad user agent or IP blocking code
#
```

```
# If you need to block certain user agents or IP addresses and
# other signatures, place that code here. Ensure the rules use
# the correct RewriteRule syntax and the [F] flag.
 RewriteCond %{HTTP_USER_AGENT} ^BlackWidow [OR]
 RewriteCond %{HTTP_USER_AGENT} ^Bot\ mailto:craftbot@yahoo.com [OR]
 RewriteCond %{HTTP_USER_AGENT} ^ChinaClaw [OR]
 RewriteCond %{HTTP_USER_AGENT} ^Custo [OR]
 RewriteCond %{HTTP_USER_AGENT} ^DISCo [OR]
 RewriteCond %{HTTP_USER_AGENT} ^Download\ Demon [OR]
 RewriteCond %{HTTP_USER_AGENT} ^eCatch [OR]
 RewriteCond %{HTTP_USER_AGENT} ^EirGrabber [OR]
 RewriteCond %{HTTP_USER_AGENT} ^EmailSiphon [OR]
 RewriteCond %{HTTP_USER_AGENT} ^EmailWolf [OR]
 RewriteCond %{HTTP_USER_AGENT} ^Express\ WebPictures [OR]
 RewriteCond %{HTTP_USER_AGENT} ^ExtractorPro [OR]
 RewriteCond %{HTTP_USER_AGENT} ^EyeNetIE [OR]
 RewriteCond %{HTTP_USER_AGENT} ^FlashGet [OR]
 RewriteCond %{HTTP_USER_AGENT} ^GetRight [OR]
 RewriteCond %{HTTP_USER_AGENT} ^GetWeb! [OR]
 RewriteCond %{HTTP_USER_AGENT} ^Go!Zilla [OR]
 RewriteCond %{HTTP_USER_AGENT} ^Go-Ahead-Got-It [OR]
 RewriteCond %{HTTP_USER_AGENT} ^GrabNet [OR]
 RewriteCond %{HTTP_USER_AGENT} ^Grafula [OR]
 RewriteCond %{HTTP_USER_AGENT} ^HMView [OR]
 RewriteCond %{HTTP_USER_AGENT} HTTrack [NC,OR]
 RewriteCond %{HTTP_USER_AGENT} ^Image\ Stripper [OR]
 RewriteCond %{HTTP_USER_AGENT} ^Image\ Sucker [OR]
 RewriteCond %{HTTP_USER_AGENT} Indy\ Library [NC,OR]
 RewriteCond %{HTTP_USER_AGENT} ^InterGET [OR]
 RewriteCond %{HTTP_USER_AGENT} ^Internet\ Ninja [OR]
 RewriteCond %{HTTP_USER_AGENT} ^JetCar [OR]
 RewriteCond %{HTTP_USER_AGENT} ^JOC\ Web\ Spider [OR]
```

```
RewriteCond %{HTTP_USER_AGENT} ^larbin [OR]
RewriteCond %{HTTP_USER_AGENT} ^LeechFTP [OR]
RewriteCond %{HTTP_USER_AGENT} ^Mass\ Downloader [OR]
RewriteCond %{HTTP_USER_AGENT} ^MIDown\ tool [OR]
RewriteCond %{HTTP_USER_AGENT} ^Mister\ PiX [OR]
RewriteCond %{HTTP_USER_AGENT} ^Navroad [OR]
RewriteCond %{HTTP_USER_AGENT} ^NearSite [OR]
RewriteCond %{HTTP_USER_AGENT} ^NetAnts [OR]
RewriteCond %{HTTP_USER_AGENT} ^NetSpider [OR]
RewriteCond %{HTTP_USER_AGENT} ^Net\ Vampire [OR]
RewriteCond %{HTTP_USER_AGENT} ^NetZIP [OR]
RewriteCond %{HTTP_USER_AGENT} ^Octopus [OR]
RewriteCond %{HTTP_USER_AGENT} ^Offline\ Explorer [OR]
RewriteCond %{HTTP_USER_AGENT} ^Offline\ Navigator [OR]
RewriteCond %{HTTP_USER_AGENT} ^PageGrabber [OR]
RewriteCond %{HTTP_USER_AGENT} ^Papa\ Foto [OR]
RewriteCond %{HTTP_USER_AGENT} ^pavuk [OR]
RewriteCond %{HTTP_USER_AGENT} ^pcBrowser [OR]
RewriteCond %{HTTP_USER_AGENT} ^RealDownload [OR]
RewriteCond %{HTTP_USER_AGENT} ^ReGet [OR]
RewriteCond %{HTTP_USER_AGENT} ^SiteSnagger [OR]
RewriteCond %{HTTP_USER_AGENT} ^SmartDownload [OR]
RewriteCond %{HTTP_USER_AGENT} ^SuperBot [OR]
RewriteCond %{HTTP_USER_AGENT} ^SuperHTTP [OR]
RewriteCond %{HTTP_USER_AGENT} ^Surfbot [OR]
RewriteCond %{HTTP_USER_AGENT} ^tAkeOut [OR]
RewriteCond %{HTTP_USER_AGENT} ^Teleport\ Pro [OR]
RewriteCond %{HTTP_USER_AGENT} ^VoidEYE [OR]
RewriteCond %{HTTP_USER_AGENT} ^Web\ Image\ Collector [OR]
RewriteCond %{HTTP_USER_AGENT} ^Web\ Sucker [OR]
RewriteCond %{HTTP_USER_AGENT} ^WebAuto [OR]
RewriteCond %{HTTP_USER_AGENT} ^WebCopier [OR]
```

```
 RewriteCond %{HTTP_USER_AGENT} ^WebFetch [OR]

 RewriteCond %{HTTP_USER_AGENT} ^WebGo\ IS [OR]

 RewriteCond %{HTTP_USER_AGENT} ^WebLeacher [OR]

 RewriteCond %{HTTP_USER_AGENT} ^WebReaper [OR]

 RewriteCond %{HTTP_USER_AGENT} ^WebSauger [OR]

 RewriteCond %{HTTP_USER_AGENT} ^Website\ eXtractor [OR]

 RewriteCond %{HTTP_USER_AGENT} ^Website\ Quester [OR]

 RewriteCond %{HTTP_USER_AGENT} ^WebStripper [OR]

 RewriteCond %{HTTP_USER_AGENT} ^WebWhacker [OR]

 RewriteCond %{HTTP_USER_AGENT} ^WebZIP [OR]

 RewriteCond %{HTTP_USER_AGENT} ^Widow [OR]

 RewriteCond %{HTTP_USER_AGENT} ^WWWOFFLE [OR]

 RewriteCond %{HTTP_USER_AGENT} ^Xaldon\ WebSpider [OR]

 RewriteCond %{HTTP_USER_AGENT} ^Zeus

 RewriteCond %{REQUEST_METHOD} ^(HEAD|TRACE|DELETE|TRACK) [NC,OR]

 RewriteCond %{THE_REQUEST} (\\r|\\n|%0A|%0D) [NC,OR]


 RewriteCond %{HTTP_REFERER} (<|>|'|%0A|%0D|%27|%3C|%3E|%00) [NC,OR]

 RewriteCond %{HTTP_COOKIE} (<|>|'|%0A|%0D|%27|%3C|%3E|%00) [NC,OR]

 RewriteCond %{REQUEST_URI} ^/(,|;|:|<|>|">|"<|/|\\\.\.\\).{0,9999} [NC,OR]


 RewriteCond %{HTTP_USER_AGENT} ^$ [OR]

 RewriteCond %{HTTP_USER_AGENT} ^(java|curl|wget) [NC,OR]

 RewriteCond %{HTTP_USER_AGENT} (winhttp|HTTrack|clshttp|archiver|loader|email|
harvest|extract|grab|miner) [NC,OR]

 RewriteCond %{HTTP_USER_AGENT} (libwww-perl|curl|wget|python|nikto|scan)
[NC,OR]

 RewriteCond %{HTTP_USER_AGENT} (<|>|'|%0A|%0D|%27|%3C|%3E|%00) [NC,OR]


########## End - Add optional bad user agent or IP blocking code


########## Begin - Rewrite rules to block out some common exploits

## If you experience problems on your site block out the operations listed below
```

```
## This attempts to block the most common type of exploit `attempts` to Joomla!

#

# If the request query string contains /proc/self/environ (by SigSiu.net)

RewriteCond %{QUERY_STRING} proc/self/environ [OR]

# Block out any script trying to set a mosConfig value through the URL

# (these attacks wouldn't work w/out Joomla! 1.5's Legacy Mode plugin)

RewriteCond %{QUERY_STRING} mosConfig_[a-zA-Z_]{1,21}(=|\%3D) [OR]

# Block out any script trying to base64_encode or base64_decode data within the
URL

RewriteCond %{QUERY_STRING} base64_(en|de)code[^(]*\([^)]*\) [OR]

## IMPORTANT: If the above line throws an HTTP 500 error, replace it with these
2 lines:

# RewriteCond %{QUERY_STRING} base64_encode\(.*\) [OR]

# RewriteCond %{QUERY_STRING} base64_decode\(.*\) [OR]

# Block out any script that includes a <script> tag in URL

RewriteCond %{QUERY_STRING} (<|%3C)([^s]*s)+cript.*(>|%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})

# Return 403 Forbidden header and show the content of the root homepage

RewriteRule .* index.php [F]

#

########## End - Rewrite rules to block out some common exploits


########## Begin - File injection protection, by SigSiu.net

RewriteCond %{REQUEST_METHOD} GET

RewriteCond %{QUERY_STRING} [a-zA-Z0-9_]=http:// [OR]

RewriteCond %{QUERY_STRING} [a-zA-Z0-9_]=(\.\.//?)+ [OR]

RewriteCond %{QUERY_STRING} [a-zA-Z0-9_]=/([a-z0-9_.]//?)+ [NC]

RewriteRule .* - [F]

########## End - File injection protection
```

```
########## Begin - Basic antispam Filter, by SigSiu.net

## I removed some common words, tweak to your liking

## This code uses PCRE and works only with Apache 2.x.

## This code will NOT work with Apache 1.x servers.

RewriteCond %{QUERY_STRING} \b(ambien|blue\spill|cialis|cocaine|ejaculation|
erectile)\b [NC,OR]

RewriteCond %{QUERY_STRING} \b(erections|hoodia|huronriveracres|impotence|
levitra|libido)\b [NC,OR]

RewriteCond %{QUERY_STRING} \b(lipitor|phentermin|pro[sz]ac|sandyauer|tramadol|
troyhamby)\b [NC,OR]

RewriteCond %{QUERY_STRING} \b(ultram|unicauca|valium|viagra|vicodin|xanax|
ypxaieo)\b [NC]

## Note: The final RewriteCond must NOT use the [OR] flag.

RewriteRule .* - [F]

## Note: The previous lines are a "compressed" version

## of the filters. You can add your own filters as:

## RewriteCond %{QUERY_STRING} \bbadword\b [NC,OR]

## where "badword" is the word you want to exclude.

########## End - Basic antispam Filter, by SigSiu.net


########## Begin - Advanced server protection - query strings, referrer and
config

# Advanced server protection, version 3.2 - May 2011

# by Nicholas K. Dionysopoulos


## Disallow PHP Easter Eggs (can be used in fingerprinting attacks to determine

## your PHP version). See http://www.0php.com/php_easter_egg.php and

## http://osvdb.org/12184 for more information

RewriteCond %{QUERY_STRING} \=PHP[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]
{4}-[0-9a-f]{12} [NC]

RewriteRule .* - [F]


## SQLi first line of defense, thanks to Radek Suski (SigSiu.net) @

## http://www.sigsiu.net/presentations/fortifying_your_joomla_website.html

## May cause problems on legitimate requests
```

```
RewriteCond %{QUERY_STRING} concat[^\(]*\( [NC,OR]

RewriteCond %{QUERY_STRING} union([^s]*s)+elect [NC,OR]

RewriteCond %{QUERY_STRING} union([^a]*a)+ll([^s]*s)+elect [NC]

RewriteRule .* - [F]


## Referrer filtering for common media files. Replace with your own domain name.

## This blocks most common fingerprinting attacks ;)

## Note: Change www\.example\.com with your own domain name, substituting the

## dots with \.  i.e. use www\.example\.com for www.example.com

RewriteRule ^images/stories/([^/]+/)*([^/.]+\.)+(jp(e?g|2)?|png|gif|bmp|css|js|
swf|ico)$ - [L]

RewriteCond %{HTTP_REFERER} .

RewriteCond %{HTTP_REFERER} !^https?://(www\.)?example\.com [NC]

RewriteCond %{REQUEST_FILENAME} -f

RewriteRule \.(jp(e?g|2)?|png|gif|bmp|css|js|swf|ico)$ - [F]


## Disallow visual fingerprinting of Joomla! sites (module position dump)

## Initial idea by Brian Teeman and Ken Crowder, see:

## http://www.slideshare.net/brianteeman/hidden-joomla-secrets

## Improved by @nikosdion to work more efficiently and handle template

## and tmpl query parameters

RewriteCond %{QUERY_STRING} (^|&)tmpl=(component|system) [NC]

RewriteRule .* - [L]

RewriteCond %{QUERY_STRING} (^|&)t(p|emplate|mpl)= [NC]

RewriteRule .* - [F]


## Disallow access to htaccess.txt, configuration.php, configuration.php-dist
and php.ini

RewriteRule ^(htaccess\.txt|configuration\.php(-dist)?|php\.ini)$ - [F]


########## End - Advanced server protection - query strings, referrer and config


########## Begin - Advanced server protection rules exceptions ####
```

```
##

## These are sample exceptions to the Advanced Server Protection 3.1

## rule set further down this file.

##

## Allow UddeIM CAPTCHA

RewriteRule ^components/com_uddeim/captcha15\.php$ - [L]

## Allow Phil Taylor's Turbo Gears

RewriteRule ^plugins/system/GoogleGears/gears-manifest\.php$ - [L]

## Allow JoomlaWorks AllVideos

RewriteRule ^plugins/content/jw_allvideos/includes/jw_allvideos_scripts\.php$ -
[L]

## Allow Admin Tools Joomla! updater to run

RewriteRule ^administrator/components/com_admintools/restore\.php$ - [L]

## Allow Akeeba Backup Professional's integrated restoration script to run

RewriteRule ^administrator/components/com_akeeba/restore\.php$ - [L]

## Allow Akeeba Kickstart

RewriteRule ^kickstart\.php$ - [L]


# Add more rules to single PHP files here


## Allow Agora attachments, but not PHP files in that directory!

RewriteCond %{REQUEST_FILENAME} !(\.php)$

RewriteCond %{REQUEST_FILENAME} -f

RewriteRule ^components/com_agora/img/members/ - [L]


# Add more rules for allowing full access (except PHP files) on more directories
here


## Uncomment to allow full access to the cache directory (strongly not
recommended!)

#RewriteRule ^cache/ - [L]

## Uncomment to allow full access to the tmp directory (strongly not
recommended!)

#RewriteRule ^tmp/ - [L]
```

```
# Add more full access rules here


########## End - Advanced server protection rules exceptions ####


########## Begin - Advanced server protection - paths and files

# Advanced server protection, version 3.2 - May 2011

# by Nicholas K. Dionysopoulos


## Back-end protection

## This also blocks fingerprinting attacks browsing for XML and INI files

RewriteRule ^administrator/?$ - [L]

RewriteRule ^administrator/index\.(php|html?)$ - [L]

RewriteRule ^administrator/index[23]\.php$ - [L]

RewriteRule
^administrator/(components|modules|templates|images|plugins)/([^/]+/)*([^/.]+\.)
+(jp(e?g|2)?|png|gif|bmp|css|js|swf|html?|mp(eg?|[34])|avi|wav|og[gv]|xlsx?|
docx?|pptx?|zip|rar|pdf|xps|txt|7z|svg|od[tsp]|flv|mov)$
 - [L]

RewriteRule ^administrator/ - [F]


## Explicitly allow access only to XML-RPC's xmlrpc/index.php or plain xmlrpc/
directory

RewriteRule ^xmlrpc/(index\.php)?$ - [L]

RewriteRule ^xmlrpc/ - [F]


## Disallow front-end access for certain Joomla! system directories

RewriteRule ^includes/js/ - [L]

RewriteRule ^(cache|includes|language|libraries|logs|tmp)/ - [F]


## Allow limited access for certain Joomla! system directories with client-
accessible content

RewriteRule
^(components|modules|plugins|templates)/([^/]+/)*([^/.]+\.)+(jp(e?g|2)?|png|gif|
bmp|css|js|swf|html?|mp(eg?|[34])|avi|wav|og[gv]|xlsx?|docx?|pptx?|zip|rar|pdf|
xps|txt|7z|svg|od[tsp]|flv|mov)$
 - [L]
```

```
## Uncomment this line if you have extensions which require direct access to
their own

## custom index.php files. Note that this is UNSAFE and the developer should be
ashamed

## for being so lame, lazy and security unconscious.

# RewriteRule ^(components|modules|plugins|templates)/([^/]+/)*(index\.php)?$ -
[L]

## Uncomment the following line if your template requires direct access to PHP
files

## inside its directory, e.g. GZip compressed copies of its CSS files

# RewriteRule ^templates/([^/]+/)*([^/.]+\.)+php$ - [L]

RewriteRule ^(components|modules|plugins|templates)/ - [F]



## Disallow access to rogue PHP files throughout the site, unless they are
explicitly allowed

RewriteCond %{REQUEST_FILENAME} \.php$

RewriteCond %{REQUEST_FILENAME} !/index[23]?\.php$

## The next line is to explicitly allow the forum post assistant(fpa-xx)script
to run

RewriteCond %{REQUEST_FILENAME} !/fpa-[a-z]{2}\.php

RewriteCond %{REQUEST_FILENAME} !/fechaporta\.php

RewriteCond %{REQUEST_FILENAME} !/abreporta\.php

RewriteCond %{REQUEST_FILENAME} -f

RewriteRule ^([^/]+/)*([^/.]+\.)+php$ - [F]



########## End - Advanced server protection - paths and files



########## Begin - Google Apps redirection, by Komra.de

## Uncomment the following line to enable:

# RewriteRule ^mail http://mail.google.com/a/example.com [R=301,L]

## If the above doesn't work on your server, try this:

## RewriteRule ^mail http://mail.google.com/a/example.com [R,L]

########## End - Google Apps redirection



########## Begin - Custom redirects
```

```
#
# If you need to redirect some pages, place that code here. Ensure those
# redirects use the correct RewriteRule syntax and the [R=301,L] flags.
#
########## End - Custom redirects

########## Begin - Redirect (www.)olddomain.com to www.example.com
## Note: olddomain.com is your old domain name, you want to redirect FROM,
## whereas www.example.com is the new domain name you want to redirect TO.
## Change those names to reflect your current configuration. Remember, this
## small part of the file is supposed to be placed in www.olddomain.com!
## Note: Replace [R=301,L] with [R,L] if you get error 500.
## Uncomment the following lines to enable:
# RewriteCond %{HTTP_HOST} ^(www\.)?olddomain\.com [NC]
# RewriteRule (.*) http://www.example.com/$1 [R=301,L]
## Note: The above section is only required if you are changing your domain
name.
########## End - Redirect (www.)olddomain.com to www.example.com

########## Begin - Redirect index.php to /
## Note: Change example.com to reflect your own domain name
RewriteCond %{THE_REQUEST} !^POST
RewriteCond %{THE_REQUEST} ^[A-Z]{3,9}\ /index\.php\ HTTP/
RewriteCond %{SERVER_PORT}>s ^(443>(s)|[0-9]+>s)$
RewriteRule ^index\.php$ http%2://www.example.com/$1 [R=301,L]
## If the above line throws a 500 error, change [R=301,L] to [R,L]
########## End - Redirect index.php to /

########## Begin - Redirect non-www to www
RewriteCond %{HTTP_HOST} !^www\. [NC]
RewriteRule ^(.*)$ http://www.%{HTTP_HOST}/$1 [R=301,L]
## If the above throws an HTTP 500 error, swap [R=301,L] with [R,L]
########## End - Redirect non-www to www
```

```
########## Begin - Redirect www to non-www
## WARNING: Comment out the non-www to www rule if you choose to use this
# RewriteCond %{HTTP_HOST} ^www\.(.+)$ [NC]
# RewriteRule ^(.*)$ http://%1/$1 [R=301,L]
## If the above throws an HTTP 500 error, swap [R=301,L] with [R,L]
########## End - Redirect non-www to www

########## Begin - Custom internal rewrites
#
# If you need to internally rewrite some specific URL requests,
# place that code here. Ensure those internal rewrites use the
# correct RewriteRule syntax without domain name and with [L] flag.
#
########## End - Custom internal rewrites

########## Begin - Joomla! core SEF Section
#
RewriteRule .* - [E=HTTP_AUTHORIZATION:%{HTTP:Authorization}]
#
# If the requested path and file is not /index.php and the request
# has not already been internally rewritten to the index.php script
RewriteCond %{REQUEST_URI} !^/index\.php
# and the request is for the site root, or for an extensionless URL,
# or the requested URL ends with one of the listed extensions
RewriteCond %{REQUEST_URI} /component/|(/[^.]*|\.(php|html?|feed|pdf|vcf|raw|
ini|zip|json|file))$ [NC]
# and the requested path and file doesn't directly match a physical file
RewriteCond %{REQUEST_FILENAME} !-f
# and the requested path doesn't directly match a physical folder
RewriteCond %{REQUEST_FILENAME} !-d
# internally rewrite the request to the index.php script
RewriteRule .* index.php [L]
```

```
#
########## End - Joomla! core SEF Section

########## Begin - Optimal default expiration time
## Note: this might cause problems and you might have to comment it out by
## placing a hash in front of this section's lines
<IfModule mod_expires.c>
        # Enable expiration control
        ExpiresActive On

        # Default expiration: 1 hour after request
        ExpiresDefault "now plus 1 hour"

        # CSS and JS expiration: 1 week after request
        ExpiresByType text/css "now plus 1 week"
        ExpiresByType application/javascript "now plus 1 week"
        ExpiresByType application/x-javascript "now plus 1 week"

        # Image files expiration: 1 month after request
        ExpiresByType image/bmp "now plus 1 month"
        ExpiresByType image/gif "now plus 1 month"
        ExpiresByType image/jpeg "now plus 1 month"
        ExpiresByType image/jp2 "now plus 1 month"
        ExpiresByType image/pipeg "now plus 1 month"
        ExpiresByType image/png "now plus 1 month"
        ExpiresByType image/svg+xml "now plus 1 month"
        ExpiresByType image/tiff "now plus 1 month"
        ExpiresByType image/vnd.microsoft.icon "now plus 1 month"
        ExpiresByType image/x-icon "now plus 1 month"
        ExpiresByType image/ico "now plus 1 month"
        ExpiresByType image/icon "now plus 1 month"
        ExpiresByType text/ico "now plus 1 month"
```

```
        ExpiresByType application/ico "now plus 1 month"
        ExpiresByType image/vnd.wap.wbmp "now plus 1 month"
        ExpiresByType application/vnd.wap.wbxml "now plus 1 month"
        ExpiresByType application/smil "now plus 1 month"

        # Audio files expiration: 1 month after request
        ExpiresByType audio/basic "now plus 1 month"
        ExpiresByType audio/mid "now plus 1 month"
        ExpiresByType audio/midi "now plus 1 month"
        ExpiresByType audio/mpeg "now plus 1 month"
        ExpiresByType audio/x-aiff "now plus 1 month"
        ExpiresByType audio/x-mpegurl "now plus 1 month"
        ExpiresByType audio/x-pn-realaudio "now plus 1 month"
        ExpiresByType audio/x-wav "now plus 1 month"

        # Movie files expiration: 1 month after request
        ExpiresByType application/x-shockwave-flash "now plus 1 month"
        ExpiresByType x-world/x-vrml "now plus 1 month"
        ExpiresByType video/x-msvideo "now plus 1 month"
        ExpiresByType video/mpeg "now plus 1 month"
        ExpiresByType video/mp4 "now plus 1 month"
        ExpiresByType video/quicktime "now plus 1 month"
        ExpiresByType video/x-la-asf "now plus 1 month"
        ExpiresByType video/x-ms-asf "now plus 1 month"
</IfModule>
########## End - Optimal expiration time
```

https://w3b.com.br/guia-pratico-sobre-seguranca-do-joomla/