

« به نام خالق آرامش »

نام کتاب: امنیت وب سایت با Htaccess

نام نویسنده: سید سعید حسینی

تعداد صفحات: ۸۶ صفحه

تاریخ انتشار: _____



امنیت وب سایت با

Htaccess

مقدماتی-متوسط



سید سعید حسینی



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

مقدمه نویسنده

امروز هیچ تجارتی را نمی توان بدون داشتن یک وب سایت جهت معرفی خود و یا معرفی محصولات و کالاها و یا حتی فروش و در یک عرصه کاملاً الکترونیکی تجارت الکترونیکی را در نظر گرفت وجود بی شمار وب سایت های مختلف که به عرصه خود در تمامی بخش ها مشغول هستند هر روز با تعدادی از این ها در ارتباط هستیم .با تمامی این پیشرفت ها تعداد زیادی از افرادی که دارای وب سایت هستند دانش کمی ابتدا در خصوص طراحی و ایجاد یک وب دارند و ثانیاً در حوزه امنیت این فضا باید گفت بدون دانش هستند . لذا داشتن یک سایت ما را مجبور خواهد کرد که به دنیای امنیت پا بگذاریم . ولی به لحاظ تخصصی بودن این حوزه علمی و گستردگی آن ما را بر آن داشت که با دانش حداقلی بتوانیم تا حدودی وب سایت خود را امن نماییم. کتاب حاضر به این موضوع می پردازد و با یادگیری فایل Htaccess که یکی از مهمترین فایل های موجود در یک وب استست بتوانید به امنیت دارایی الکترونیکی خود کمک نمایید.

مؤلف

سید سعید حسینی



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

تقديم به

يا باصالح از پرده برون آي كه دلم غرق تمناست
تقصير دلم نيست تماشاي تو زيباست. . .

اللهم عجل لوليک الفرج...



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

7.....	مقدمه
7.....	تعریف سرور
9.....	انواع وب سرور
9.....	سرورهای اینترنتی
10.....	انواع حملات بر روی وب سرورها :
11.....	ابزارهای رایج حمله به وب سرور :
12.....	آشنایی با وب سرور آپاچی
12.....	وب سرور
15.....	دلیل استفاده از وب سرور APACHE
16.....	APACHE رایگان است.
16.....	کد منبع APACHE به رایگان در دسترس است.
18.....	قابلیت های وب سرور APACHE
19.....	دستیابی به منابع مورد نیاز
21.....	اطمینان از صحت نصب برنامه APACHE
22.....	فصل دوم
22.....	پیکربندی HTACCESS
32.....	معرفی زبان پیشفرض (DEFAULTCHARSET) :
33.....	تنظیم TIMEZONE
84	HTTP://HTACCESSBUILDER.COM
85	HTTP://WWW.HTACCESSREDIRECT.NET
86	HTTP://WWW.HTACCESSEDITOR.COM/EN.SHTML
86	HTTP://WWW.GENERATEIT.NET/MOD-REWRITE/INDEX.PHP



مقدمه

استفاده از شبکه‌های کامپیوتری در سالیان اخیر روندی تصاعدی پیدا کرده است. شبکه‌های کامپیوتری، زیر ساخت مناسب برای سازمان‌ها و موسسات را در رابطه با تکنولوژی اطلاعات فراهم می‌نمایند. مقوله تکنولوژی اطلاعات به همان اندازه که جذاب و موثر است، در صورت عدم رعایت اصول امنیت به همان میزان و یا شاید بیشتر، نگران کننده و مسئله آفرین خواهد بود. اغلب سازمان‌های دولتی و خصوصی در کشور، دارای وب سایت اختصاصی خود در اینترنت می‌باشند. سازمان‌ها برای ارائه وب سایت، یا خود امکانات مربوطه را فراهم نموده و با نصب تجهیزات سخت افزاری و تهیه پهنای باند لازم، اقدام به عرضه سایت خود در اینترنت نموده‌اند یا از امکانات مربوط به شرکت‌های ارائه دهنده خدمات میزبانی استفاده می‌نمایند. بدون تردید سرویس دهنده وب یکی از مهمترین نرم افزارهای موجود در دنیای اینترنت محسوب می‌گردد. کاربرانی که به سایت یک سازمان یا موسسه متصل و درخواست اطلاعاتی را می‌نمایند، خواسته آنان در نهایت در اختیار سرویس دهنده وب گذاشته می‌شود. سرویس دهنده وب، اولین نقطه ورود اطلاعات و آخرین نقطه خروج اطلاعات از یک سایت است. نصب و پیکربندی مناسب چنین نرم افزار مهمی، بسیار حائز اهمیت بوده و تدابیر امنیتی خاصی را طلب می‌نماید.

تعریف سرور

به سرویس‌گیرنده‌ها و استفاده کنندگان از سرویس‌ها، میزبان و به سرویس دهنده‌ها و ارائه کنندگان سرویس، سرور گفته می‌شود. سرورها بسته به نوع خدماتی که ارائه می‌دهند دسته‌بندی‌های متفاوتی دارند. به عنوان مثال، به سرورهای زیر نگاهی بیاندازید:

- Mail Server: سروری که به کاربران شبکه خدمات ایمیل را ارائه می‌دهد.
- DHCP Server: سروری که به کاربران شبکه به طور خودکار آدرس IP می‌دهد.
- DNS Server: سروری که امکان تبدیل درخواست کاربران شبکه، را برای دسترسی به سایت‌های اینترنت، با ارائه آدرس واقعی سایت مزبور می‌دهد، بدین ترتیب دیگر نیازی به حفظ بودن آدرس‌های IP سایت‌ها نخواهد بود.



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

- **Web Server:** سروری که خدمات تحت وب را از قبیل میزبانی وب و ... را ارائه می‌دهد.

وب سرور در واقع به دو معنی است:

- برنامه کامپیوتری که مسئول قبول کردن درخواستهای Http از مشتریان است که همان مرورگرهای وب هستند و پاسخ‌ها را به همراه یک سری اطلاعات به آنها پست می‌کنند. این پاسخ‌ها صفحات Html هستند. بطور مثال اگر در صفحه مرورگر آدرس <http://piau.ac.ir/index.php> را وارد کنید، یک درخواست به دامنه‌ای که نامش piau.ac.ir است، فرستاده می‌شود. آنگاه سرور صفحه [index.php](http://piau.ac.ir/index.php) را می‌فرستد.
- کامپیوتری است که یک برنامه‌ی کامپیوتری را اجرا می‌کند و کارایی اش همانند مطالبی است که در بالا گفته شد. هر کامپیوتری می‌تواند با نصب نرم افزار سرور به وب سرور تبدیل شود.

در عمل بسیاری از وب سرورها، ویژگیهای زیر را نیز پیاده‌سازی می‌کنند:

- شناسایی: درخواست شناسایی اختیاری قبل از اجازه دسترسی به انواع منابع
- نه تنها مفاهیم استاتیک (مفاهیم فایلی که بر روی سیستم فایلی وجود دارد) بلکه مفاهیم دینامیک را با یک یا چند ساختار مانند CGI, SSI, ASP, ASP.NET, PHP, JSP, FastCGI, SCGI اداره می‌کند.
- پشتیبانی از HTTPS تا به کاربران اجازه دهد اتصالات مطمئنی به سرور را بر روی پورت ۴۴۳ به جای ۸۰ برقرار کنند.
- فشرده‌سازی مطالب تا بتوان از حجم پاسخها کم کرد (توسط کدسازی (GZIP).
- پشتیبانی از فایل‌های بزرگ تا بتواند فایل‌های بزرگ‌تر از ۲ گیگا بایت را سرویس‌دهی کند.
- کنترل کردن پهنای باند: تا سرعت پاسخها را محدود کند و شبکه را پر ازدحام نکند و قادر باشد تعداد بیشتری مشتری را سرویس دهی کند.



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

- ترجمه مسیر: وب سرورها قادرند تا کامپوننت مسیر URL را به منابع فایل سیستم محلی (برای درخواستهای استاتیک) و نام برنامه داخلی یا خارجی (برای درخواستهای دینامیک) نگاشت کنند

برای مثال کاربرد آدرس زیر را درخواست می‌کند: `http://www.example.com/path/file.html` مرورگر وب کاربر آنرا به یک اتصال به `http://www.example.com` با درخواست `http 1.1` ترجمه می‌کند: `GET/path/file.html.php HTTP/1.1 HOST:http://www.example.com` وب سرور بر روی `http://www.example.com`، مسیر درخواستی را به آدرس مسیر اصلی اضافه می‌کند. آنگاه اگر وب سرور فایلی داشته باشد، آن را خوانده و پاسخ را که مجموعه‌ای از مطالب فایل است به عنوان پاسخ می‌فرستد.

انواع وب سرور

- وب سرور داخلی روی شبکه Intranet
- وب سرور خارجی روی شبکه Internet
- روی شبکه خصوصی قرار می‌گیرند.
- از اطلاعات مختص به شرکت نگهداری می‌کند.
- دسترسی به این سرور فقط از طریق کاربران داخلی است.
- روی شبکه عمومی قرار می‌گیرد.
- از اطلاعات کالاها، خدمات و تجارت شرکت نگهداری می‌کند.
- دسترسی به این سرور از طریق تمام کاربران امکان پذیر که ریسک بالایی دارد.

سرورهای اینترنتی

سهام استفاده بازار، از نرم افزارهای وب سرور، در زیر نشان داده شده است که در برآورد Netcraft در ژانویه ۲۰۰۹ منتشر شده است.



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

جدول ۱۰۰: سهم استفاده بازار، از نرم افزارهای وب سرور

فروشنده محصول	نام محصول	وب سایتهای میزبانی شده	درصد
بنیاد نرم افزار آپاچی	سرور آپاچی	۹۶,۵۳۱,۰۳۳	۵۲/۰۵%
مایکروسافت	IIS	۴۷,۴۰۲,۳۶۱	۲۲/۹۰%
گوگل	GWS	۹,۸۶۴,۳۰۳	۵/۳۲%
Nginx	nginx	۳,۴۶۲,۵۵۱	۱/۸۷%
Lighttpd	lighttpd	۲,۹۸۹,۴۱۶	۱/۶۱%
Oversee	Oversee	۱,۸۴۷,۰۳۹	۱/۰۰%
دیگر	-	۹,۷۵۶,۶۵۰	۵/۳۶%
مجموع	-	۱۸۵,۴۷۴,۴۶۶	۱۰۰/۰۰%

انواع حملات بر روی وب سرورها :

حملات پیمایش پوشه (traversal attacks Directory): این نوع از حملات باگ های موجود در وب سرور را به کار گرفته تا دسترسی غیرمجاز به فایل ها و پوشه هایی که در مسیر عمومی قرار ندارند را بکار بگیرند. زمانی که هکر دسترسی را بدست آورد می تواند اطلاعات حیاتی را دریافت کرده و دستورات دلخواه خود را بر روی وب سرور اجرا کند یا بدافزار نصب کند . حملات رد سرویس (Service Attacks Denial of): با استفاده از این نوع حمله و سرور ممکن است از سرویس دهی خارج شود و سرویس دهی صورت نپذیرد .



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

سرقت سیستم نام دامنه (Name System Hijacking Domain): با استفاده از این نوع حمله اطلاعات دی آن اس به نقطه‌ای که هکر می‌خواهد تغییر یافته. در نتیجه همه ترافیکی که بایستی به سرور اصلی هدایت شود به سرور هکر هدایت می‌شود.

شنود (Sniffing): داده‌های رمزنگاری نشده بر روی شبکه ممکن است به منظور دسترسی بدون مجوز بکار گرفته شوند و شنود شوند.

فیشینگ (Phishing): این نوع حمله با ایجاد یک صفحه جعلی و هدایت کاربران به صفحه جعلی به منظور سرقت اطلاعات انجام می‌شود.

دیفیس (Defacement): این نوع حمله معمولاً توسط هکرها خریداری می‌گردد. حملات دیفیس با استفاده از یکی از متدهای تزریق اسکریپت یا دیگر متدهای حمله به منظور قرار دادن امضای هکر بر روی یکی از صفحات سایت به دلایل شخصی، مذهبی و اعتقادی و یا حتی ضربه زدن به وجه اجتماعی سایت هدف صورت می‌پذیرد.

خوب اکنون که انواع حملات را شناختید این حملات توسط چه ابزارهایی صورت می‌پذیرند؟

ابزارهای رایج حمله به وب سرور:

متاسپلویت (Metasploit): این یک ابزار متن باز می‌باشد که دارای نسخه تجاری هم می‌باشد. به منظور بکارگیری اکسپلویت‌های موجود در سمت سرور بکار گرفته می‌شود. این ابزار به صورت پیش‌فرض دارای اکسپلویت‌های زیادی هست ولی شما در صورت تسلط به زبان رومی قادر به نوشتن اکسپلویت‌های سفارشی نیز هستید. متاسپلویت در اصل ابزاری به منظور تست آسیب‌پذیری به شمار می‌رود.

Mpack: یک ابزار بکارگیری وب است. به زبان PHP نوشته شده است و توسط پایگاه داده MySQL پشتیبانی می‌شود. زمانی که وب سرور توسط Mpack مورد حمله قرار گرفت همه ترافیک آن به سکن سایت‌های داندلود مخرب هدایت می‌شود.

زئوس (Zeus): این ابزار قادر است سیستم بکار گرفته شده را به یک بوت، زامبی تبدیل کند. بوت کامپیوتر هک شده است که به منظور دیگر حملات اینترنتی از آن استفاده می‌شود. بوتنت مجموعه از کامپیوترهای بکار گرفته شده است. بوتنت‌ها را می‌توان در ارسال ایمیل‌های اسپم و حملات DOS مورد استفاده قرار داد.

Neosplit: این ابزار را می‌توان به منظور نصب حذف یا کپی برنامه‌ها مورد استفاده قرار داد.



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

آشنایی با وب سرور آپاچی

امروزه اطلاعات، در دنیای ما نقش بسیار مهمی دارند و بسیاری از این اطلاعات توسط اینترنت انتقال پیدا می‌کنند. متداولترین پروتکلی که برای انتقال اطلاعات از آن استفاده می‌شود، پروتکل HTTP است. پروتکل HTTP به عنوان پروتکلی سریع، قوی و با بار کم بر روی CPU و حافظه سرور طراحی شده است و البته برای جلوگیری از کاهش پیدا کردن کارایی وب بر اثر استفاده بسیار زیاد از این پروتکل، باید کارایی اش را بهینه سازی کرد. دو استراتژی اصلی برای بهینه سازی کارایی وجود دارد:

۱- بهینه سازی کارایی وب سرور

۲- بهینه سازی کارایی پروتکل HTTP

وب سرور

البته یکی از چیزهایی که باعث می‌شود افراد سردرگم شوند حالت طراحی وب و مفهوم وب سرور است. بیشتر مردم فکر می‌کنند سرور یک ماشین فیزیکی بزرگ مثل سیستم کامپیوتری است که در یک اتاق سرد نگهداری می‌شود و یا حتی مثل سیستم هکرهاست! که همه فکر می‌کنند زیر زمین هستند.



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

وب سرور ها کامپیوتر هایی هستند که صفحات وب را آماده نمایش می کنند هر وب سرور یک ای پی اختصاصی و دامنه دارد. برای مثال زمانی که ادرسی را در مرورگر تایپ می کنید برای مثال <http://www.esecurity.ir/index.html> درخواستی برای وب سرور فرستاده میشود وب سروری که دامنه اش [esecurity.ir](http://www.esecurity.ir) است سپس سرور صفحه ای به نام `index.html` را بازخوانی و به مرورگر ارسال می کند. هر کامپیوتری با نصب برنامه سرور و اتصال به اینترنت می تواند یک وب سرور باشد. برنامه های متعددی برای این کار وجود دارد که از معروف ترین آن ها می توان به `Apache`، `NCSA` و `Netscape` اشاره نمود. وب سرور ها امکان دارد به دلایل متفاوتی کند شوند مثلاً در خواست های متعددی که در مدت زمان کوتاهی ارسال میشوند میتوانند وب سرور را کند کنند اما با توجه به پردازش بسیار سریع این وب سرور ها این کند شدن هم معمولاً به چشم کاربر نیامده و متوجه آن نمیشود. دیتایی که توسط پردازنده فرستاده میشود باید در پروتکلی که `HTTP` نام دارد مطابقت داده شود تا از ارتباط سرور ها با یکدیگر اطمینان حاصل گشته و خطایی به وجود نیاید. وب سرور ها علاوه بر انتشار صفحات وب کارهای دیگری از قبیل دانلود فایل ها از طریق `FTP`، خدمات مربوط به ارسال ایمیل و... را انجام میدهد. برای انتخاب یک وب سرور مناسب باید به گزینه هایی از قبیل چگونگی کارکرده آن با بقیه سیستم عامل ها و سرور ها، قدرت مدیریت برنامه نویسی آن در سمت سرور، ویژگی های امنیتی، موتورهای جست و جوگر، تجهیزات ساختمان سایت و... که ممکن است نیاز باشد توجه نمود.

البته وب سرور نرم افزار هایی مثل آپاچی روی یک سیستم کامپیوتری اختصاص داده شده است که می توانید از ویژگی های میزبانی وب اکثر سیستم عامل های ساخته شده مثل `IIS` و `پندوز` استفاده کنید و ضمناً خوب است بدانید اوبونتو هم یک وب سرور محسوب می شود. در حقیقت راه اندازی یک وب سرور باعث می شود صفحات وبی که از کامپیوتر های دیگر ارایه شده، آسانتر و سریعتر باز شوند. البته وب سرور جنبه های پیچیده تری هم دارد، مثل ارائه محتوای پویا (داینامیک) با اشکال یا محتوای صفحاتی که اطلاعات ورودی کاربر را می پذیرد، پردازش آن ها، و حتی ایجاد صفحات سفارشی جدید.



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

وب سایت های سطح بالاتر شما را قادر می سازند با استفاده از نرم افزار هایی مانند آپاچی، که توانایی پردازش ورودی اطلاعات کاربران را دارد، به طور خودکار صفحات وب را ایجاد کنید و با استفاده از زبان های برنامه نویسی وب مانند PHP، جاوا و ... آن ها را کامل تر سازید.

لینوکس (Linux) به هسته ی سیستم عامل های شبه یونیکس می گویند که در سال ۱۹۹۱ توسط لینوس توروالدز توسعه یافت. برخی به سیستم عامل هایی که از هسته ی لینوکس استفاده می کنند نیز لینوکس می گویند (در این دانشنامه هم منظور از لینوکس، سیستم عامل هایی است که از هسته ی لینوکس استفاده می کنند).

امروزه لینوکس بر روی اکثر ابزار های هوشمند اطراف ما نظیر ساعت های هوشمند، تلفن های همراه، تبلت ها، مسیریاب ها، کامپیوتر های خانگی، سرور ها، ابر کامپیوتر ها و ... استفاده می شود. اکثر توزیع های لینوکس، بسته کامل نصبی LAMP را برای نصب به صورت آماده در خود دارند. با توجه به بررسی های انجام گرفته توسط W3Techs در اکتبر سال 2013، 58.5% سهم بازار وب سرور ها بین دو توزیع محبوب Debian و Ubuntu تقسیم شده بود، در حالیکه سه توزیع RHEL، Fedora و CentOS با هم دیگر تنها 37.3% از سهم بازار را در اختیار داشته اند.



نقش وب سرور LAMP را رسماً وب سرور Apache آپاچی ایفا می کند. سرورس دهنده ی اچ تی تی پی آپاچی که بیشتر آن را آپاچی می نامند، یک برنامه ی وب

سرور است که نقش کلیدی در توسعه دنیای وب ایفا می کند و اکثر شرکت های هاستینگ از این وب سرور برای ارائه خدمات میزبانی خود استفاده می کنند. در سال ۲۰۰۹ آپاچی اولین برنامه ی وب سرور بود که حدود ۱۰۰ میلیون وب سایت به کمک آن سرورس دهی می کردند. آپاچی به طور معمول در محیط های بر پایه یونیکس و لینوکس استفاده می شود.

این برنامه تحت مجوز آپاچی بوده و به صورت متن باز (Open Source) است. و از سال ۱۹۹۶ تاکنون پرکاربردترین برنامه در حوزه ی خود است. تخمین زده شده است که این وب سرور محبوب تا ژوئن سال 2014 میلادی بیش از 52.27% وب سایت های فعال را پشتیبانی کرده است .



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

ویژگی‌های این برنامه بسیار گسترده است: از پشتیبانی زبان‌های برنامه‌نویسی سمت سرور مانند PHP, Perl, Python و TCL گرفته تا احراز هویت TLS و SSL. یکی از کاربردی‌ترین موارد مربوط به آپاچی وب سرور برای برنامه‌نویسان استفاده از پرونده فایل htaccess است. برنامه‌نویس می‌تواند با اعمال تغییراتی در این پرونده که بر هر شاخه‌ای قابل اضافه شدن است دستورات ویژه‌ی آن شاخه را به سرور ارائه دهد. برای نمونه اگر بخواهد که در صورت وارد کردن نشانی aa.html نام آن باقی بماند ولی در واقع پرونده main.php?page=bb اجرا شود به وسیله‌ی این پرونده قادر به اعمال دستورش خواهد بود.

دلیل استفاده از وب سرور Apache

حقیقت این است که وب سرور Apache به یکی از عوامل موفقیت وب تبدیل شده است. با وجودی که این ادعا ممکن است برای عده‌ای ناخوشایند باشد، اما دلیل زیادی برای اثبات این واقعیت وجود دارد. برر سی‌های اخیر حاکی از آن است که بیشترین وب سایت‌های موجود در حال حاضر از وب سرور Apache به عنوان سرویس دهنده وب استفاده می‌کنند. این دلیل را می‌توان علت این موفقیت برشمرد:

- رایگان است.
- کد منبع Apache به رایگان در دسترس است. (به این گونه نرم‌افزارها اصطلاحاً "کدباز" یا open source گفته می‌شود.)
- بر روی مجموعه‌ای بسیار متنوعی از سیستم‌های عامل قابل استفاده است.
- Apache دائماً در حال توسعه و افزایش قابلیت‌های جدید است.



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

- Apache بسیار توانمند بوده و به واسطه طراحی ماجولار به راحتی قابل توسعه است.

Apache رایگان است.

وب سرور Apache ضمن برخورداری از ویژگی‌ها و توانایی‌های قابل توجه کاملاً رایگان است. با این وجود از آنجا که به واسطه انتشار این وب سرور هیچ‌گونه منفعتی نصیب سازمان توسعه دهنده آن یعنی Apache software foundation نمی‌شود طبیعی است که نرم‌افزار مزبور از هیچ‌گونه پشتیبانی فنی از طریق تلفن یا به صورت online که در مورد نرم‌افزارهای تجاری شاهد آن هستیم در رابطه با وب سرور Apache مرسوم نیست.

با این همه مستندات بسیار جامعی از نرم‌افزار مرود بحث از طریق وب سایت مربوطه در دسترس علاقه‌مندان قرار دارد.

کد منبع Apache به رایگان در دسترس است.

چنانچه در برنامه‌نویسی تبحر دارید می‌توانید کد منبع Apache را آن گونه که مورد نیاز شماست دستخوش تغییر کنید. با این حال بیشتر کاربران با هدف تغییر نحوه عملکرد Apache دست به این اقدام نمی‌زنند، بلکه صرفاً نحوه پیکربندی آن را تغییر می‌دهند، و با کمپایل مجدد کد منبع Apache انتظاراتی که وب سرور دارند، تأمین می‌کنند. از این رو یکی در صورتی که به یک وب سرور مختصر و سبک نیاز دارید می‌توانید Apache را به گونه‌ای کمپایل کنید که تنها نیازهای خواسته شده را تأمین کند. به این ترتیب چنانچه با مشکل حادی برخورد کردید یا مایلید تغییراتی را در کد منبع Apache اعمال کنید می‌توانید با برخورداری از دانش برنامه‌نویسی این کار را انجام دهید. Apache بر روی مجموعه بسیار متنوعی از سیستم‌های عامل قابل استفاده است.



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

نسخه‌های مختلف وب سرور Apache به منظور بهره‌گیری تحت سیستم‌های عامل متداولی از جمله این موارد توسعه یافته است.

– unix

– liunix

– ویندوز (شامل نسخه‌هایی x9 تا XP هر چند که نسخه‌های تحت NT و 2000 از قابلیت‌های بیشتری برخوردارند).

– Novell Netware

– Mac OSX

منهای برخی اختلافات جزئی مانند استقرار فایل‌های وب سرور Apache در سیستم فایل‌ملکرد این وب سرور در تمامی محیط‌های عامل فوق یکسان است. Apache دائماً در حال توسعه و افزایش قابلیت‌های جدید است.

مسئولیت توسعه و بهبود کارایی وب سرور Apache به عهده سازمان software foundation یعنی پدید آورنده آن است. شگفت‌انگیز است، به طوری که تنها پس از گذشت تنها چند روز اشکالات و شکاف‌های امنیتی یافت شده اصلاح می‌شود. نکته مذکور Apache را به پایدارترین و در عین حال ایمن‌ترین وب سرور حال حاضر تبدیل کرده است. البته این ویژگی یکی از مهم‌ترین دلایل مقبولیت آن است.

مزیت دیگری که به واسطه سرعت بالای توسعه و انتشار نسخه‌های جدید نرم‌افزار به دست می‌آید، مجموعه‌ای ارزشمند از قابلیت‌هاست، که البته وب سرور Apache نیز از این قاعده مستثنی نیست.



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

به این ترتیب تکنولوژی جدیدی که همه روزه در درارتباط با شبکه جهانی اینترنت با آنها مواجه می شویم، پیش از هر وب سرور دیگری توسط Apache پیاده سازی شده و مورد پشتیبانی قرار می گیرند.

قابلیت های وب سرور Apache

وب سرور Apache نام خود را از شیوه های که نخستین بار برای توسعه آن به کار رفت، گرفته است. اساساً این وب سرور از مجموعه ای مؤلفه نرم افزاری یا اصطلاحاً patch (با تلفظ پچ) تشکیل شده است به گونه ای که بسیاری از توسعه دهندگان برای تاکید این موضوع از اصطلاح "a patchy server" یا تلفظ "اچی سرور" به معنی سروری که از مجموعه ای مؤلفه های نرم افزاری ایجاد شده است، استفاده می کردند.

مدت هاست که توسعه وب سرور Apache در قالب قطعات یا ماچول های نرم افزاری مستقل صورت می گیرد. توسعه وب سرور Apache بر اساس این ماچول (اصطلاحاً توسعه "ماچولار") موجب شده که بهره گیری از Apache به عنوان وب سرور سربار کمتری را به سخت افزار و سیستم عامل میزبانی که بر روی آن مستقر شده است تحمیل کند.

به بیان بهتر در استفاده از این وب سرور تنها ماچول های مورد نیاز بر روی سخت افزار و سیستم عامل میزبان مستقر می شوند. این ویژگی همچنین روند توسعه و پشتیبانی از ماچول هایی را که شرکت و برنامه نویسان مستقل (اصطلاحاً third parties) به منظورهای مختلف توسعه می دهند، تسهیل می کند.

وب سرور Apache تقریباً تمامی تکنولوژی اینترنت را که در ارتباط با وب تویعه پیدا کرده اند به خوبی مورد پشتیبانی قرار می دهد. این قابلیت حتی شامل برخی تکنولوژی های اختصاصی از جمله Microsoft Frontpage Extensions نیز می شود. وب سرور Apache تمام مشخصات



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

پروتکل HTTP اسکریپت‌نویسی، احراز هویت و قابلیت استفاده از سایر تکنولوژی‌ها را به خوبی مورد پشتیبانی قرار داده است.

دسته‌ای از قابلیت‌های وب سرور Apache عبارتند از:

- پشتیبانی جامع و کامل از پروتکل HTTP
- قابلیت پیکربندی بالا و امنیت قابل ملاحظه
- پشتیبانی از PHP
- پشتیبانی از CGI و سایر زبان‌های اسکریپت‌نویسی

دستیابی به منابع مورد نیاز

هر آنچه که در ارتباط با نصب وب سرور Apache بدان نیاز دارید از طریق وب سایت Apache به آدرس [http:// www. Apache. Otg](http://www.Apache.Otg) قابل دستیابی است. از طریق این آدرس اینترنتی می‌توانید کد منبع نسخه اجرایی (اصطلاحاً binaryversion) وب سرور Apache را برای محیط عامل UNIX و linux و همچنین شکل صفحه اصلی وب سایت Apache را نشان می‌دهد.

نسخه‌های مختلفی از وب سرور Apache برای بهره برداری تحت نسخه‌های مختلف سیستم عامل Linux توسعه یافته است. برای مثال کاربران سیستم عامل Redhat Linux می‌توانند از طریق مکانیزم Redhat package Manager یا اصطلاحاً RPM وب سرور Apache را بر روی سیستم خود نصب کنند.

نصب برنامه Apache تحت سیستم عامل Linux از طریق کمپایل کد منبع



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

چنانچه مایل به کمپایل Apache باشید، لازم است ابتدا کد منبع مناسبی را در اختیار داشته باشید. برای شروع فایل مورد نظر را از آدرس مذکور در یک فهرست موقت یا در موقعیت `usr/src/` سیستم فایب که معمولاً برای این منظور از آن استفاده می شود بارگذاری کنید. برای باز کردن آرشیو کد منبع (و عموماً هر آرشیو دیگر) روشهای مختلفی وجود دارد. روش مورد استفاده برای این کار به برنامه های نصب شده بر روی کامپیوتر و نوع آرشیوی که بارگذاری کرده اید بستگی دارد.

چنانچه نسخه ای از آرشیو را که با استفاده از برنامه فشرده ساز `gzip` فشرده شده است، بارگذاری کرده اید. (در این صورت فایلی با پسوند `tar.gz` در اختیار دارید.) پیش از این هر گونه اقدامی برای باز کردن آرشیو، لازم است با استفاده از همین برنامه فشرده ساز آن را از حالت فشرده خارج سازید. فرمان زیر که از فهرست نژمیزبان فایل `tar.gz` `httpd-2.0` را به طور توأم انجام می دهد. (علامت `$` اعلان سیستم عامل `unix` است.)

```
tra-zxvf httpd-2.0.8.tar.z $
```

اما اگر نسخه ای از آرشیو را با استفاده از برنامه فشرده سازی `bzip` یا `bzip2` (یا نسخه های دیگری از این برنامه فشرده ساز) فشرده شده است بارگذاری کرده اید (در این صورت فایلی با پسوند `tar.Z` در اختیار دارید) با صدور فرمان زیر علاوه بر اینکه فایل مورد نظر را از حالت فشرده خارج می کنید بلکه آرشیون مزبور را نیز باز می کنید:

```
tar -zxvf httpd-2.0.8.tar.z $
```

نتیجه اجرای هر دو فرمان مذکور یکسان است، به طوری که در نهایت فایل های حاوی کد منبع Apache در فهرست تحت عنوان `httpd-2.0` مستقر می شوند. که علامت ستاره بیانگر شماره نسخه `minor` است.



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

اطمینان از صحت نصب برنامه Apache

پس از نصب و راه اندازی برنامه Apache به منظور اطمینان از صحت عملیات به سادگی می توانید مرورگر اینترنت خود را باز کرده و نام ماشین میزبان وب سرور Apache را در فیلد آدرس آن بنویسید. بر روی ماشین میزبان وب سرور Apache کافی است آدرس زیر را در فیلد آدرس مرورگر اینترنت وارد کنید:

`http:// localhost`

با این اقدام باید نتیجه ای شبیه به شکل را مشاهده کنید. در صورتی که با استفاده از کامپیوتر دیگری به ماشین میزبان وب سرور Apache متصل شده اید. کافی است عبارت `local host` را در آدرس فوق با نام کامل آن اصطلاحاً `fully qualified name` یا آدرس IP مربوطه جایگزین کنید.

وب سروری که به درستی پیکربندی نشده است می تواند به سرعت به عاملی برای نفوذ به ماشین میزبان یا حتی شبکه ای که ماشین مزبور روی آن مستقر شده تبدیل شود پس از نصب وب سرور بی درنگ آن را متوقف کرد. برای متوقف کردن وب سرور کافی است فرمان زیر را صادر کنید.

```
$ usr/sbin/ Apachectl stop
```



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

فصل دوم

پیکربندی Htaccess



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

htaccess یک فایل پیکر بندی برای وب سایت هایی است که از سرور آپاچی استفاده می کنند. وقتی این فایل در یکی از پوشه های وب سایت قرار می گیرد، وب سرور آپاچی بر روی می کند که چه دستوراتی در این فایل وجود دارد و بعد طبق این دستورات، آن قسمت از سایت که **htaccess** در آن قرار دارد را پیکر بندی می کند. این فایل می تواند برای فعال و یا غیر فعال کردن یک سری از توابع و ویژگی های وب سرور آپاچی مورد استفاده قرار بگیرد.

فایل **htaccess** یک فایل ساده اسکی (ASCII) می باشد که می توانید آن را به وسیله نرم افزارهای متنی ساده نظیر **Notepad** و یا **SimpleText** به وجود آورید. دقت داشته باشید که این فایل دارای هیچ نامی نمی باشد و فقط دارای پسوند است که پسوند آن، **htaccess** می باشد. برای درست کردن این فایل می توانید فایل متنی خود را که مثلا به صورت **test.txt** می باشد، تغییر نام داده و آن را به صورت **htaccess** در آورید.

تذکر: اگر از ویندوز استفاده می کنید و هنگام تغییر نام دادن فایل به **htaccess**، پیغام خطایی مینی بر اینکه «می بایست برای فایل خود نامی را اختیار کنید» دریافت می کنید، فایل را بر روی سرور آپلود کرده و سپس در سرور آن را تغییر نام دهید. این کار به وسیله نرم افزارهای **FTP** نظیر **CuteFTP** یا **AbsoluteFTP** بسیار ساده می باشد.

در هنگام استفاده از فایل **htaccess** دانستن چهار نکته از اهمیت به سزایی برخوردار است:
نکته اول: فایل **htaccess** را می بایست به صورت **ASCII** آپلود کرده و مجوز دسترسی به آن را بر روی **644** (یا **-rw-r--r--**) تنظیم نمایید.

نکته دوم: فایل **htaccess** را می توانید به پوشه های مختلفی اعمال نمایید. به عنوان مثال اگر آن را بر روی **root** آپلود نمایید، تنظیمات آن به کل وب سایت اعمال خواهد شد و اگر آن را درون پوشه ای نظیر **/images** آپلود نمایید، تنظیمات آن به پوشه **images** و زیرپوشه های آن اعمال خواهد شد.
نکته سوم: هر **htaccess** را می بایست درون یک خط تایپ نمایید. یعنی در انتهای هر دستور، می بایست یکبار کلید **Enter** را فشار دهید.



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

نکته چهارم: به دلیل آنکه نحوه پیکربندی این فایل مانند پیکربندی فایل اصلی سرور (httpd.conf) است تصمیمات بر اساس شرایط اخذ می شود. البته تمامی امکانات httpd.conf را شامل نمی شود!

استفاده از این فایل در همه موارد پیشنهاد نمی شود زیرا امنیت وب سرور را تحت شعاع قرار می دهد. اما در مواقعی که سرور به صورت اشتراکی و اصطلاحاً share شده خدمت رسانی می کند و تعداد زیادی سایت بر روی آن قرار دارد پیشنهاد آن است که از فایل htaccess استفاده شود. زیرا هر سایت باید توانایی پیکر بندی قسمت مربوط به خود را دارا باشد. دستوراتی که در یک فایل htaccess مورد استفاده قرار می گیرند، شامل فرامینی جهت کنترل موارد زیر هستند:

چند نکته :

اگر از وب سرور آپاچی برای وب سایت خود استفاده می کنید، دیگر نیازی نیست برای هر مشکل جزئی و کوچک، با هاست تان تماس بگیرید. با یادگیری قابلیت های مهم فایل htaccess، می توانید کنترل کامل وبسایت خود را برعهده بگیرید.

در هنگام استفاده از فایل htaccess، دانستن سه نکته از اهمیت به سزایی برخوردار است:

نکته اول: فایل htaccess را می بایست به صورت ASCII آپلود کرده و مجوز دسترسی به آن را بر روی 644 (یا rw-r--r--) تنظیم نمایید.

نکته دوم: فایل htaccess را می توانید به پوشه های مختلفی اعمال نمایید. به عنوان مثال اگر آن را بر روی root آپلود نمایید، تنظیمات آن به کل وبسایت اعمال خواهد شد و اگر آن را درون پوشه ای نظیر images/ آپلود نمایید، تنظیمات آن به پوشه images و زیرپوشه های آن اعمال خواهد شد.



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

نکته سوم: هر `htaccess` را می‌بایست درون یک خط تاپی نمایش دهید. یعنی در انتهای هر دستور، می‌بایست یکبار کلید `Enter` را فشار دهید.

در میان تعداد زیاد ابزارهای مدیریت وب سایت، `Htaccess` یکی از مهم‌ترین آن‌ها به شما می‌رود که با یک سری تغییرات ساده و دست‌نویسی‌ها سریع می‌توانید دسترس به بسیاری از بخش‌های وب سایت خود را تغییر داده، نحوه آدرس‌دهی به آن‌ها را عوض کنید و بسیاری از کارهای مفید و جذاب دیگر را روی وب سایت خود پیاده‌سازی کنید.

مهمترین بخش‌های این فایل و نحوه‌ی دست‌نویسی

هر فردی که وب سایت خود را بر روی `Apache` قرار داده است، مطمئناً روزی به یک آموزش کامل و جامع در مورد کار با `Htaccess` پیدا کرده است.

این فایل توانایی کار با هر گونه زبان تحت وب از `PHP` گرفته تا `Ruby` را نیز دارد.

چرا از `Htaccess` استفاده کنیم؟

این یک سوال مهم است که قبل از شروع کار باید به آن پاسخ داد که ممکن است شما بعد از دیدن

این سوال سریعاً بپرسید اصلاً یک فایل `Htaccess` چیست و چه می‌کند؟

و من باید پاسخ بدهم که این یک فایل تنظیماتی مهم است که توسط سرور `Apache` مورد استفاده

قرار می‌گیرد و می‌تواند به وب سرور بگوید که چگونه دستورات موجود در `header` های `HTTP`

را هندل کند.

بزرگترین دلیل استفاده از این فایل بحث امنیت می‌باشد؛ شما می‌توانید با استفاده از فایل

`Htaccess` روی فایل‌های خود پسورد بگذارید، دسترسی به آن‌ها را محدود کنید، کاربر را به

صفحات 404 بفرستید و...



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

در شروع باید نحوه دسترسی به فایل `Htaccess` را به شما آموزش دهم لذا مراحل زیر را انجام دهید

وارد سنی پنل سایت خود شوید

[http:// example.com/cpanel](http://example.com/cpanel)

[http:// example.com:2082](http://example.com:2082)



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly



بر روی **File Manager** کلیک نمایید



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

The screenshot displays a web hosting control panel interface. On the left, a 'Webmail' sidebar contains a 'Stats' section with the following data:

- Main Domain: **zamani-sec.ir**
- Home Directory: **/home/ywajjjuv**
- Last Login From: **198.144.105.76**
- Disk Space Usage: **NA / ∞**
- Monthly Bandwidth Transfer: **0 bytes / 195.31 GB**
- Email Accounts: **0 / ∞**
- Subdomains: **0 / ∞**
- Parked Domains: **0 / ∞**
- Addon Domains: **0 / 1**
- FTP Accounts: **0 / ∞**
- Mailing Lists: **0 / ∞**
- All SQL Databases: **2 / ∞**
- MySQL Databases: **2**
- MySQL Disk Space: **4.71 MB**
- Mailing List Disk: **n MB**

The main area is divided into three sections:

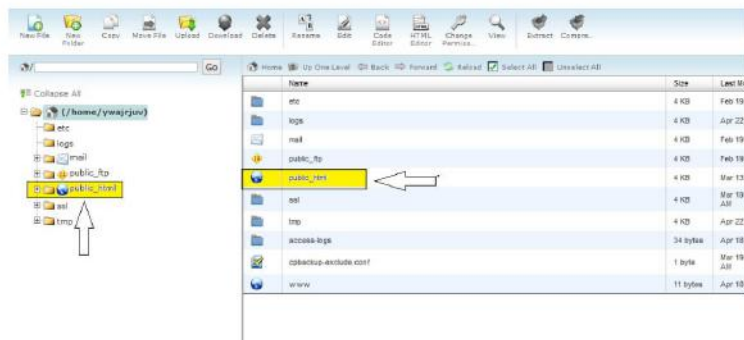
- Mail:** Contains icons for Email Accounts, Webmail, BoxTrapper, SpamAssassin™, Forwarders, Autoreponder, Default Address, Mailing Lists, Account-Level Filtering, User-Level Filtering, Email Trace, Import Addresses and Forwarders, Email Authentication, and MX Entry.
- Files:** Contains icons for Backups, Backup Wizard, File Manager (highlighted in yellow with a red arrow), Legacy File Manager, Disk Space Usage, Web Disk, FTP Accounts, and FTP Session Control.
- Logs:** Contains icons for Latest Structure, Bandwidth, Webalizer, Webalizer STD, Raw Access Log, and Analog Stats.

At the bottom left of the screenshot, the URL `extension-site-facility/extensions-security.html` is visible.

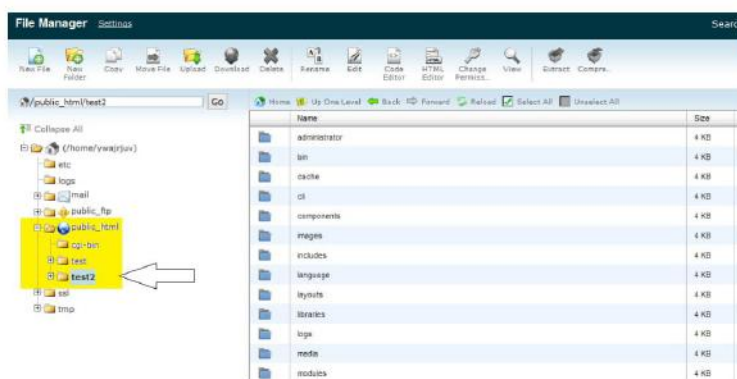
Home Directory را انتخاب نموده و بر روی GO کلیک نمایید



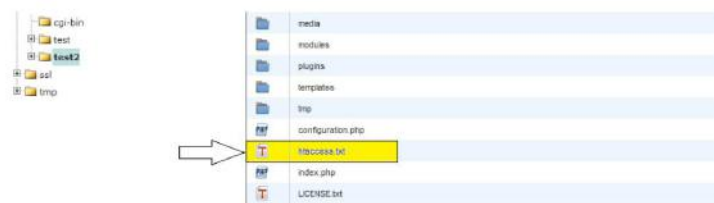
Public Html را انتخاب نمایید زیرا محل قرارگیری وب سایتها در این دایرکتوری می باشد



انتخاب وب سایتمان و ورود به دایرکتوری



و ایجاد فایل Htaccess.txt



و بعد از کانفیگ و اضافه نمودن دستورات نام فایل را به **.htaccess** تغییر داده چون در سیستمهای لینوکسی نقطه قبل از اسم باعث مخفی شدن فایل خواهد شد.

و حالا وارد بحث آموزش نحوه ی دستورنویسی در این فایل می شویم، با ما همراه باشید:

جلوگیری از دسترسی به فایل htaccess

این کد دسترسی سایر کاربران را به فایل htaccess شما محدود خواهد کرد :

کد:

secure htaccess file

```
<Files .htaccess>
order allow,deny
deny from all
</Files>

prevent viewing of a specific file
<Files secretfile.jpg>
order allow,deny
10deny from all
</Files>

multiple file types
<FilesMatch ".(htaccess|htpasswd|ini|phps|fla|psd|log|sh)">
```



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

```
Order Allow,Deny
Deny from all
</FilesMatch>
```

معرفی زبان پیشفرض (DefaultCharset) :

برای اینکه زبان پیشفرض استفاده شده رو به مرورگر ها بگین بیشتر وقتها اون رو در قسمت هدر کدهای سایت قرار میدین. با دستور کوتاه زیر به آپاچی میگی که همیشه صفحات را با زبان خاصی ارسال کن. این عمل برای سئو سایت تاثیری خوبی داره.

```
#pass the default character set
AddDefaultCharset utf-8
```

روشن و آماده استفاده کردن mod_rewrite :

mod_rewrite را می توانید از درون فایل htaccess خود اجرا کنید برای اینکه ماژول mod_rewrite را آماده استفاده کنید

از کد زیر در فایل htaccess استفاده کنید. بهترین مکان برای قرار دادن این کد در همان سطر اول htaccess می باشد.

```
RewriteEngine on
```

فراموش نکنید که htaccess به بزرگی و کوچکی حروف حساس است

و اینکه این کد را در خط اول همه فایل های htaccess ای که در آن ها از mod_rewrite استفاده می کنید قرار دهید.



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

- حالت نمونه کد های mod_rewrite

حالت ساده و نمونه mod_rewrite بصورت زیر است

```
[p]
RewriteRule Pattern Substitution [Flag(s)]
[/p]
```

تنظیم Timezone

بعضی اوقات، شما از توابع date یا mktime در PHP استفاده می کنید، پیغام خنده داری ممکن است به شما بدهد. این کد راه حل آن است. timezone را برای سرور خود ست می کنید. لیستی از timezone ها را می توانید در دیدن لینک ها برای شما امکان پذیر نیست. لطفا ثبت نام کنید یا وارد حساب خود شوید تا بتوانید لینک ها را ببینید.

مشاهده کنید.

کد:

```
SetEnv TZ Asia/Tehran
```

SEO Friendly



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

چرا SEO Friendly ? این روزها، همه موتورهای جستجوی مدرن قابلیت شناسایی 301 Permanent Redirect ها را دارند.

کد:

```
Redirect 301 <a href="http://www.datisdesign.com">http://www.datisdesign.com/</a> /http://www.datisdesign.com
```

گذشتن از پیغام دانلود

کد:

```
AddType application/octet-stream .pdf
AddType application/octet-stream .zip
AddType application/octet-stream .mov
```

Skip WWW

یکی از نکات بهینه سازی برای موتورهای جستجو اطمینان از این نکته است که تنها یک آدرس است که به وب سایت شما اشاره می کند. بنابراین، شما نیاز دارید که همه ترافیک WWW را به سمت non-WWW ارجاع دهید و یا بالعکس :

کد:

```
RewriteEngine On RewriteBase / RewriteCond %{HTTP_HOST} ^www.datisdesign.com [NC] RewriteRule ^(.*)$ http://datisdesign.com/$1 [L,R=301]
```

Cache Files

کش یعنی اینکه به سری از فایل های سایت شما که تغییراتی ندارند و یا در فواصل زمانی زیاد دچار تغییرات قرار میگیرند مشن رو در یک محفظه به نسبت امن در مرورگر کاربر ذخیره کنیم تا در دفعات



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

بعدي مراجعه به سايت اين فايلها از خود سيستم اون بارگذاري بشن و در نهايت سرعت بارگذاري سايت خودمون رو چندين چند برابر كنيم.

كدي كه در پايين قرار دادم كدي هست كه من خودم دارم ازش استفاده ميكنم و به مرور كامل شده و شما مي تونيد از هر بخش اون بنا به نياز خودتون استفاده كنين :

```
#BEGIN Expire headers
<ifModule mod_expires.c>
  ExpiresActive On
  ExpiresDefault "access plus 1 month"
  ExpiresByType image/x-icon "access plus 1 year"
  ExpiresByType image/jpeg "access plus 1 month"
  ExpiresByType image/png "access plus 1 month"
  ExpiresByType image/gif "access plus 1 month"
  ExpiresByType image/jpg "access plus 1 month"
  ExpiresByType application/x-shockwave-flash "access plus 1 month"
  ExpiresByType text/css "access plus 1 month"
  ExpiresByType text/javascript "access plus 1 year"
  ExpiresByType application/javascript "access plus 1 year"
  ExpiresByType application/x-javascript "access plus 1 year"
  ExpiresByType text/html "access plus 600 seconds"
  ExpiresByType application/xhtml+xml "access plus 600 seconds"
  ExpiresByType font/ttf "access plus 1 year"
  ExpiresByType font/woff "access plus 1 year"
</ifModule>

#END Expire headers
#BEGIN Cache-Control Headers
<ifModule mod_headers.c>
<"$<filesMatch "\.(ico|jpe?g|png|gif|swf|woff|ttf)
"Header set Cache-Control "max-age=2592000, public
<filesMatch/>

<"$filesMatch "\.(css)>
"Header set Cache-Control "max-age=2592000, public
<filesMatch/>

<"$filesMatch "\.(js)>
```



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

```
"Header set Cache-Control "max-age=2592000, private
<filesMatch/>
<"$filesMatch "\.(x?html?|php)>
"Header set Cache-Control "max-age=600, private, must-revalidate
<filesMatch/>
</ifModule>
#END Cache-Control Headers

#BEGIN Turn ETags Off
<ifModule mod_headers.c>
  Header unset ETag
</ifModule>
FileETag None
#END Turn ETags Off

#BEGIN Remove Last-Modified Header
<ifModule mod_headers.c>
  Header unset Last-Modified
</ifModule>
#END Remove Last-Modified Header
```

غیرفعال کردن کش برای یکسری از فایلها

برای بعضی از فایلها هم میتوانید کش را غیرفعال کنید :

کد:

```
#explicitly disable caching for scripts and other dynamic files
<filesmatch>Header unset Cache-Control </filesmatch>
```



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

یک انتقال ساده

اگر خواهید که انتقال ساده از یک url به یک url دیگر داشته باشید می توانید از کد زیر استفاده کنید

```
[P]
RewriteRule ^fileone.html$ filetwo.html
[/P]
```

این کد باعث میشود که اگر سرور درخواستی در مورد بازکردن فایل fileone.html دریافت کرد فایل filetwo.html باز شود.

ممنوع ورود کردن یک آی پی خاص

اگر خواهید که ورود شخصی با آی پی خاص به وبسایتتان جلوگیری کنید می توانید از کد زیر استفاده کنید

```
$RewriteCond %{REMOTE_ADDR} ^(A.B.C.D)
RewriteRule ^/* http://www.domain.com/sorry.html [L]
```

به جای A B C D اجزای چهارگانه IP مورد نظر رو وارد کنید و به جای http://www.domain.com/sorry.html آدرس مورد نظر که مثلا می تونه یک صفحه حاوی پیغام هشدار باشه رو وارد کنید



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

خلاص شدن از دست Query Strings

اگر بیشتر url ها در وبسایت شما چیزی مانند

`http://www.domain.com/home.html?example=12345abcd`

هست پس باید به فکری برای نجات دادن خودتون بکنید چون در این صورت وبسایت شما بدرستی در سایت هایی نظیر گوگل ذخیره نمیشه و حتی کاربرانتون اگر بخوان آدرس یک صفحه رو توی ذهنشون داشته باشند تا جای دیگر از اون استفاده کنند این عمل براشون دشوار میشه . برای اینکه از دست اینجور آدرس ها خلاص شید از کد زیر استفاده کنید

```
RewriteCond %{QUERY_STRING} ^id=456&lid=789.* [NC]
RewriteRule (.*) http://www.domain.com/$1? [R=301]
```

با این کد نه تنها از دست query ه راحت میشید بلکه از دست اون علامت سوال هم راحت میشید .

این کار به جنبه امنیتی هم داره و اون جلوگیری از هک شدن وبسایت از طریق sql injection هست

دسترسی کاربران را بسته به دامنه آنها محدود کنید

کد:

```
# 1block visitors referred from indicated domains
2IfModule mod_rewrite.c>
3RewriteEngine on
4RewriteCond %{HTTP_REFERER} scumbag.com [NC,OR]
5RewriteCond %{HTTP_REFERER} wormhole.com [NC,OR]
6RewriteRule .* - [F]
7
8IfModule>
```



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

ایجاد محدودیت در آپلود فایل :

با کد زیر حداکثر حجم فایل قابل آپلود رو ۲۰ مگابایت تنظیم کردیم.

```
php_value upload_max_filesize 20M
```

ایجاد محدودیت در حجم پست ارسالی :

با کد زیر همیشه حداکثر حجم هر پست رو ۲ مگابایت تعیین کرد:

```
php_value post_max_size 2M
```

نمایش پیغام **request time** در بازه زمانی مشخص:

request time حداکثر زمانی درخواست فراخوانی یک صفحه هست :

```
php_value max_execution_time 200
```

حداکثر زمان دریافت اطلاعات **POST** و **GET** :

```
php_value max_input_time 250
```

فعال کردن قابلیت **Gzip** :



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

وقتی کاربری از طریق مرورگر سایتش درخواست نمایش سایت شما رو میده این درخواست به سرور شما ارسال میشه و سرور فایلهایی که برای نمایش سایت شما لازم هست رو جمع میکنه و به مرورگر کاربر میفرسته و سایت شما نمایش داده میشه، ولی اگه از قابلیت Gzip استفاده کنید هنگام درخواست سرور فایلهای مورد نیاز رو به صورت فشرده در می یاره و این فایل فشرده رو به مرورگر ارسال میکنه و بعد از خارج شدن از حالت فشرده در سیستم کاربر سایت شما نمایش داده میشه. در حقیقت میزان حجمی که باید توسط مرورگر برای نمایش سایت شما دریافت شه کمتر میشه و این یعنی افزایش قابل توجه سرعت بارگذاری سایت. برای اینکه چک کنید که هاست شما از این قابلیت پشتیبانی میکنه از این ابزار استفاده کنید. اگه پشتیبانی نمیکه باید هاست رو عوض کنید و اگر پشتیبانی کرد کد زیر رو برای فعال شدنش روی سایت خودتون در فایل htaccess قرار بدین (این کدی هست که من برای سایت خودم استفاده میکنم شما میتونین پسوند های مورد نظر خودتون رو با پسوند های درج شده در اینجا تعویض کنین)

```
#BEGIN Compress text files
<ifModule mod_deflate.c>
<filesMatch "\.(css|js|x?html?|php|woff|ttf|png|jpg|gif)"$>
SetOutputFilter DEFLATE
</filesMatch>
</ifModule>
#END Compress text files
```

تصویر پیشفرض

با استفاده از این کد اگر یکی از تصاویر وبسایتتون دچار مشکلی شد و به هر دلیلی لود نشد تصویری که اینجا معرفی میکنید جایگزین اون میشه . با استفاده از این کد ظاهری حرفه ای تر به وبسایتتون بدین

```
RewriteCond %{REQUEST_FILENAME} !-f
RewriteRule ^images/.*.jpg$ /images/default.jpg [L]
```



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

با جایگزین کردن `/images/default.jpg` با آدرس تصویر پیشفرض و تغییر دادن `images/` و `.jpg` محل تصاویر و فرمت تصاویری رو که می خواهید در صورت اشکال با `/images/default.jpg` جایگزین بشه ، کد رو اختصاصی و آماده استفاده در سرور خودتون کنید

جلوگیری از hotlinking

سارقان مطلب و `bandwidth` با کپی کردن لینک فایل های درون سرور شما (مثلا تصاویر ، موزیک ها ، کلیپ های فلش و ...) به پیشرفت سایت خودشان کمک می کنند و به جای آن از پهنای بایند شما استفاده می کنند که می تواند باعث کندی لود شدن صفحات سایت شما و یا حتی در برخی موارد تمام شدن پهنای باند ماهیانه شما بشه . برای اینکه جلوی این افراد رو بگیرید از کد زیر استفاده کنید .

```
$^! RewriteCond %{HTTP_REFERER}
RewriteCond %{HTTP_REFERER} !^http://(www.)?domain.com/ .*$ [NC]
RewriteRule .(gif|jpg|swf|flv|png)$ /feed/ [R=302,L]
```

در کد بالا `domain.com` رو با دامین سایت خودتون عوض کنید

منتقل کردن از چند دامین به یک دامین دیگر

```
$^! RewriteCond %{HTTP_REFERER}
RewriteCond %{HTTP_REFERER} !^http://(www.)?domain.com/ .*$ [NC]
```



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly


```
RewriteRule .(gif|jpg|swf|flv|png)$ /feed/ [R=302,L]
```

اگر از چند دامین برای آدرس دهی با سایتتان استفاده می کنید این امر ممکن است که هر دو دامین را به یک دامین دیگر روی سرور بفرستید

فقط به جای `http://www.domain.net` و `domain.net` دو دامین خودتون رو بنویسید و به جای `http://domain.net` آدرس جدید رو بنویسید

فراموش نکنید که :

* `mod_rewrite` از درون فایل `htaccess` اجرا می شود

* دستورات در فایل `htaccess` به حروف بزرگ و کوچک حساسند

* همیشه قبل از دست کاری فایل `htaccess` از اون نسخه پشتیبان تهیه کنید

اجازه دهی و جلوگیری از دسترسی ها

شما به سادگی با استفاده از این فایل می توانید از ورود آدرس آی پی های خاص به وب سایت خود جلوگیری کنید. دقت کنید که در آدرس های زیر، آدرس سوم بخش چهارم را ندارد و حتما می توانید حدس بزنید که چرا:

```
order allow,deny
deny from 255.0.0.0
deny from 123.45.6
allow from all
```



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

جلوگیری از لیست کردن پوشه ها در مرورگر

ممکن است بعضی اوقات در وب سایت خود پوشه های بازی داشته باشید که تنظیم شده اند تا به صورت پیش فرض قابل مرور باشند. این بدین معناست که کاربر ها می توانند تمامی فایل های لیست شده در آن و ساختار کامل آن را توسط مرورگر خود مشاهده کنند، به عنوان مثال پوشه ی تصاویر و ...

اما بعضی توسعه دهنده ها نمی خواهند چنین اتفاقی بیفتد و قصد جلوگیری از مرور پوشه های خود توسط دیگران را دارند.

خوشبختانه دستور مورد استفاده در چنین شرایطی بسیار ساده است:

Options -Indexes 1

این دستور را می توانید در فایل `Htaccess` هر پوشه ای که قصد جلوگیری از مرور آن را دارید قرار دهید.

امن سازی با رمز عبور

یکی دیگر از کاربردهای استفاده از فایل `Htaccess` را می توان قابلیت امن سازی بخش های خاصی از وب سایت دانست.

برای این کار می بایست فایلی را برای قرارگیری اطلاعات مربوط به ورود تولید کنید و آن را مورد استفاده قرار دهید، این کار باعث جلوگیری از بسیاری از حملات صورت گرفته به وب سایت شما خواهد شد.

یک نمونه از کدهای تولید شده را می توانید در این قسمت مشاهده کنید:

```
AuthType Basic
"AuthName "This Area is Password Protected
AuthUserFile /full/path/to/.htpasswd
Require valid-user
```

چند نکته در خصوص وردپرس



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

امن سازی وردپرس با استفاده از .htaccess

پیشنهاد می کنیم با استفاده از قواعد بخش قبل، دستورات زیر را برای امن سازی ورود به صفحه ی wp-login.php انجام دهید:

```
<Files wp-login.php>
Order Deny,Allow
Deny from All
Satisfy Any
"AuthName "Protected By AskApache
AuthUserFile /web/askapache.com/.htpasswd1
AuthType Basic
Require valid-user
</Files/>
```

بالا بردن امنیت فایل wp-config در htaccess :

برای جلوگیری از سرقت اطلاعات فایل حیاتی و جلوگیری از دسترسی به اطلاعات پایگاه داده سایت کد زیر رو در فایل قرار بدین

```
<files wp-config.php>
order allow,deny
deny from all
</files/>
```

افزایش امنیت محتوای فولدر wp-includes :

کافیه کد زیر رو در htaccess قرار بدین :

```
.Block the include-only files #
RewriteEngine On
```



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

/ RewriteBase

```
RewriteRule ^wp-admin/includes/ - [F,L]
RewriteRule !^wp-includes/ - [S=3]
RewriteRule ^wp-includes/[^\.]+\.\php$ - [F,L]
RewriteRule ^wp-includes/js/tinymce/langs/.+\.\php - [F,L]
RewriteRule ^wp-includes/theme-compat/ - [F,L]
```

حل مشکل تعداد آیتم های فهرست وردپرس :

اگر دقت کرده باشید با شین هنگام ساخت فهرست دسته ها به صورت دستی از یه جایی به بعد دیگه هر چی اضافه میکنیم بعد ذخیره می بینیم اضافه نشده برای حل این محدودیت کد زیر رو تو فایل htaccess قرار بدین :

```
<IfModule mod_php.c>
php_value suhosin.post.max_vars 7000
php_value suhosin.request.max_vars 7000
</ IfModule>
```

حذف category از آدرس وردپرسی شما :

شاید شما هم دوست داشته باشید کلمه /category/ رو از آدرس وردپرسی خودتون حذف کنید. این کار علاوه بر اینکه باعث جمع و جور شدن url شما میشه تا حدودی هم در سئو سایتتون موثر هست. البته با تغییر پیوند یکتای مربوط به category هم میشه که این کار رو افزونه های سئو برای شما انجام میدن و این کد رو محض یک راه حل بیرون از چهارچوب اصولی قرار دادم .

```
RewriteRule ^category/(.+)$ http://www.yourblog.com/$1 [R=301,L]
```

قواعد بازنویسی آدرس های HTTP

بازنویسی قواعد آدرس های HTTP یکی از مهم ترین و پرکاربردترین استفاده های htaccess محسوب می شود که امروزه بسیار مورد استفاده قرار می گیرد. به این مثال نگاهی بیندازید:



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

```
Options +FollowSymLinks
RewriteEngine On
/ RewriteBase
```

```
RewriteRule !\.(html|php)$ - [S=4]
RewriteRule ^([^_]*)([^_]*)([^_]*)([^_]*)(.*)$ $1-$2-$3-$4-$5
[E=uscor:Yes]
RewriteRule ^([^_]*)([^_]*)([^_]*)(.*)$ $1-$2-$3-$4 [E=uscor:Yes]
RewriteRule ^([^_]*)([^_]*)(.*)$ $1-$2-$3 [E=uscor:Yes]
RewriteRule ^([^_]*)(.*)$ $1-$2 [E=uscor:Yes]
```

```
$RewriteCond %{ENV:uscor} ^Yes
RewriteRule (.*?) http://d.com/$1 [R=301,L]
```

دستور RewriteEngine و RewriteBase همیشه می توانند روی چنین مقادیر دقیقی تنظیم شود اما شما می بایست ابتدا RewriteEngine را فعال سازید.

روش های زیادی در اینترنت در مورد این موضوع که چگونه می توانید mod_rewrite را در سرور خود فعال سازید وجود دارند.

به نحو دستورات و ساختار آن ها دقت کنید، بعد از دریافت درخواست های HTTP، سریعاً آدرس ها با این نحوها و عبارات منظم تطابق می یابند و کاربر را به صفحه ی مورد نظر هدایت می کنند. دستوراتی مثل [R=301,L] دستورات مهمی هستند که به آن ها پرچم های بازنویسی گفته می شود و می توانید اطلاعات زیادی را در مورد آن ها در اینترنت پیدا کنید.

دستورات بالا ممکن است شما را کمی گیج کرده باشد اما نگران نباشید، من پیشنهاد می کنم نگاهی به این آدرس بیندازید تا به شما در تولید آدرس های مورد نظر کمک کند و همچنین اینجا نیز آموزش بسیار جالبی وجود دارد با یک مثال ساده که می توان به خوبی بازنویسی آدرس ها را با توجه به آن فراگرفت.

```
RewriteRule ^dir/([0-9]+)/?$ /index.php?id=$1 [L]
```

در یادگیری آن عجله نکنید، ممکن است یادگیری تمامی دستورات آن و مسلط شدن روی آن چندین ماه طول بکشد!



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

تعیین ایندکس و صفحه ی پیشفرض در یک پوشه

همانطور که احتمالا می دانید، Apache هنگام درخواست یک پوشه یا صفحه ی اصلی وب سایت، به سراغ فایل هایی مثل index.html, index.php و... می رود اما شما می توانید با استفاده از یک دستور ساده این موضوع را کمی اصلاح کنید و به ترتیب اولویت خود، فایل هایی که Apache باید به صورت پیش فرض در هر پوشه به دنبال آن ها برگردد را تعیین کنید:

```
DirectoryIndex index.html index.cgi index.php 1
```

اگر هاست شما از این فایل پشتیبانی میکند اما از SSI پشتیبانی نمیکند

کافیست خطوط زیر را وارد کنید تا از SSI هم پشتیبانی شود .

```
AddType text/html .shtml
AddHandler server-parsed .shtml
Options Indexed FollowSymLinks Includes
```

خط اول بیان میکند که کلیه فایل ها با پسوند shtml دارا صحت هستند

خط دوم یک handler اضافه میکند خط سوم باعث میشود سرور این فایل ها را اجرا کند .

بنابراین شما خودتان بدون نیاز به درخواست از ادمین سرور توانستید SSI را فعال کنید .

اگر هم دو ست دارید فایل های SSI بصورت جداگانه از فایل های html نبا شدند میتوانید خطوط زیر را اضافه کنید :

```
AddType text/html .shtml .html .htm
AddHandler server-prased .shtml .html .htm
Options Indexes FollowSymLinks Includes
```



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

این خطوط باعث میشوند که کلیه فایل‌ها با پسوند html نیز به سرور برای اجرا شدن منتقل شوند تا اگر کدی که برای SSI باشد داشته باشد را اجرا کند و سپس صفحه شروع به بارگذاری کند .

مجبورسازی برای دانلود فایل‌ها چندرسانه ای

مرورگر بسیاری از کاربران ممکن است قابلیت پخش بعضی فرمت‌های چندرسانه ای را به صورت مستقل داشته باشد اما شما شاید به دلایلی بخواهید از این موضوع جلوگیری کنید و کاربر را مجبور به دانلود چنین فرمت‌هایی کنید. برای انجام این موضوع می‌توانید از دستور ساده‌ی زیر استفاده کرده و فرمت‌های مدنظر خود را نیز به آن‌ها اضافه کنید:

```
AddType application/octet-stream .zip .mp3 .mp4 1
```

سندهای خطای شخصی سازی شده

احتمالا در بسیاری از وب سایت‌ها با صفحات شخصی سازی شده‌ی 404 و ... مواجه شده اید و پرسیده اید که چگونه می‌توان چنین صفحاتی را طراحی کرد؟ پاسخ این سوال واقعا ساده است. با نگاهی به دستورات زیر پاسخ خود را خواهید گرفت:



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

ErrorDocument 100 /100_CONTINUE
ErrorDocument 101 /101_SWITCHING_PROTOCOLS
ErrorDocument 102 /102_PROCESSING
ErrorDocument 200 /200_OK
ErrorDocument 201 /201_CREATED
ErrorDocument 202 /202_ACCEPTED
ErrorDocument 203 /203_NON_AUTHORITATIVE
ErrorDocument 204 /204_NO_CONTENT
ErrorDocument 205 /205_RESET_CONTENT
ErrorDocument 206 /206_PARTIAL_CONTENT
ErrorDocument 207 /207_MULTI_STATUS
ErrorDocument 300 /300_MULTIPLE_CHOICES
ErrorDocument 301 /301_MOVED_PERMANENTLY
ErrorDocument 302 /302_MOVED_TEMPORARILY
ErrorDocument 303 /303_SEE_OTHER
ErrorDocument 304 /304_NOT_MODIFIED
ErrorDocument 305 /305_USE_PROXY
ErrorDocument 307 /307_TEMPORARY_REDIRECT
ErrorDocument 400 /400_BAD_REQUEST
ErrorDocument 401 /401_UNAUTHORIZED
ErrorDocument 402 /402_PAYMENT_REQUIRED
ErrorDocument 403 /403_FORBIDDEN
ErrorDocument 404 /404_NOT_FOUND

ErrorDocument 405 /405_METHOD_NOT_ALLOWED
ErrorDocument 406 /406_NOT_ACCEPTABLE
ErrorDocument 407 /407_PROXY_AUTHENTICATION_REQUIRED
ErrorDocument 408 /408_REQUEST_TIME_OUT
ErrorDocument 409 /409_CONFLICT
ErrorDocument 410 /410_GONE
ErrorDocument 411 /411_LENGTH_REQUIRED
ErrorDocument 412 /412_PRECONDITION_FAILED
ErrorDocument 413 /413_REQUEST_ENTITY_TOO_LARGE



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

ErrorDocument 414 /414_REQUEST_URI_TOO_LARGE
ErrorDocument 415 /415_UNSUPPORTED_MEDIA_TYPE
ErrorDocument 416 /416_RANGE_NOT_SATISFIABLE
ErrorDocument 417 /417_EXPECTATION_FAILED
ErrorDocument 422 /422_UNPROCESSABLE_ENTITY
ErrorDocument 423 /423_LOCKED
ErrorDocument 424 /424_FAILED_DEPENDENCY
ErrorDocument 426 /426_UPGRADE_REQUIRED
ErrorDocument 500 /500_INTERNAL_SERVER_ERROR
ErrorDocument 501 /501_NOT_IMPLEMENTED
ErrorDocument 502 /502_BAD_GATEWAY
ErrorDocument 503 /503_SERVICE_UNAVAILABLE
ErrorDocument 504 /504_GATEWAY_TIME_OUT
ErrorDocument 505 /505_VERSION_NOT_SUPPORTED
ErrorDocument 506 /506_VARIANT_ALSO_VARIES
ErrorDocument 507 /507_INSUFFICIENT_STORAGE
ErrorDocument 510 /510_NOT_EXTENDED

جلوگیری از حملات وب

کد htaccess جلوگیری از حملات dos

برای جلوگیری از بازدید fake و ربات ها بر روی وبسایتان کد زیر را به انتهای htaccess هاست خود بیافزایید

```
SetEnvIfNoCase User-Agent .* allowed_agent  
Order Deny,Allow  
Deny from All  
Allow from env=allowed_agent
```



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

جلوگیری از اسکن سایت توسط اسکنر ها

برای جلوگیری از اسکن سایت توسط اسکنر ها میتوانید با ایجاد یک فایل htaccess و قرار دادن کد های زیر ، مانع اسکن سایت توسط این اسکنر ها شوید.

```
RewriteEngine On 1
<IfModule mod_rewrite.c> 2
[RewriteCond %{HTTP_USER_AGENT} ^w3af.sourceforge.net [NC,OR] 3
[RewriteCond %{HTTP_USER_AGENT} dirbuster [NC,OR] 4
[RewriteCond %{HTTP_USER_AGENT} nikto [NC,OR] 5
[RewriteCond %{HTTP_USER_AGENT} SF [OR] 6
[RewriteCond %{HTTP_USER_AGENT} sqlmap [NC,OR] 7
[RewriteCond %{HTTP_USER_AGENT} fimap [NC,OR] 8
[RewriteCond %{HTTP_USER_AGENT} nessus [NC,OR] 9
[RewriteCond %{HTTP_USER_AGENT} whatweb [NC,OR] 10
[RewriteCond %{HTTP_USER_AGENT} Openvas [NC,OR] 11
[RewriteCond %{HTTP_USER_AGENT} jbrofuzz [NC,OR] 12
[RewriteCond %{HTTP_USER_AGENT} libwhisker [NC,OR] 13
[RewriteCond %{HTTP_USER_AGENT} webshag [NC,OR] 14
[RewriteCond %{HTTP:Acunetix-Product} ^WVS 15
[RewriteRule ^.* http://127.0.0.1/ [R=301,L] 16
</IfModule/> 17
```



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

کد .htaccess: جلوگیری از حملات SQL Injection

کد .htaccess زیر از حملات SQL Injection تا حد زیادی جلوگیری می‌نماید و از خطر در امان خواهید بود (تقریباً)

```
RewriteEngine On  
Options +FollowSymLinks  
ServerSignature Off
```

```
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).* [NC,OR]  
RewriteCond %{HTTP_COOKIE} ^(<|>|%0A|%0D|%27|%3C|%3E).*  
[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).* [NC,OR]
```

```
RewriteCond %{QUERY_STRING}  
^(<|>|\"|\"|)|%0A|%0D|%22|%27|%3C|%3E|.*(\\*|union|select|insert|cast|s  
et|declare|drop|update|md5|benchmark).* [NC,OR]  
RewriteCond %{QUERY_STRING} ^.*(localhost|loopback|127\\.0\\.0\\.1).*  
[NC,OR]
```



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

```
RewriteCond %{QUERY_STRING} ^.*\[A-Za-z0-9\].* [NC,OR]#  
RewriteCond %{QUERY_STRING}  
^.*(<|>|'|"%0A|%0D|%27|%3C|%3E|).* [NC]  
RewriteRule ^(.*)$ sqlinj.cfm
```

اگر کسی قصد هک کردن وبسایت شما را داشته باشید با پیج `sqlinj.cfm` رو به رو می‌شود. شما می‌توانید یک اسم‌دیگر برای این پیج انتخاب کنید. از این به بعد در زمینه `htaccess` بیشتر پست می‌گذارم

ضمناً این کد جلوی نرم افزار `httrack` عزیز را هم می‌گیرد و نمی‌گذارد قالب شما را با جاش بردارد!

جلوگیری از حملات تزریق کد (SQL INJECTION) به وسیله `htaccess`

همونطور که میدونید حملات تزریق کد به علت نقص امنیتی در پایگاه داده معمولاً به وجود میاد که البته دیگه با پیشرفت و افزایش امنیت و اصول طراحی استاندارد سایت دیگه این باگ از بین میره.

و امکان هک شدن از این طریق نزدیک صفر همیشه اما یک کد قرار میدم که با قرار دادنش در `htaccess` تا حد خیلی زیادی از حملات `sql injection` در امان خواهید بود

و در صورتی که کسی بخواد با این روش شما رو هک کنه یا صفحه `hack.cfm` رو بروی همیشه.

که این صفحه رو خود شما باید طراحی کنید و در هاستتون قرار بدین.



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

از این کد در سیستم های مدیریت محتوا مثل : وردپرس-جوملا-نیوک-دروپال و انجمن ها و چت روم همیشه استفاده کرد

درون فایل htaccess این کد رو قرار بدین :

کد:

```
RewriteEngine On

Options +FollowSymLinks

ServerSignature Off

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

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

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

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

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

RewriteCond %{HTTP_USER_AGENT} ^$ [OR]

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



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

```
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|).)* [NC,OR]

RewriteCond %{QUERY_STRING}
^.*(;<|>|'|"%0A|%0D|%22|%27|%3C|%3E|).*(^*|union|select|insert|cast|s
et|declare|drop|update|md5|benchmark).* [NC,OR]

RewriteCond %{QUERY_STRING} ^.*(localhost|loopback|127\.\0\.\0\.\1).*
[NC,OR]

RewriteCond %{QUERY_STRING} ^.*\.[A-Za-z0-9].* [NC,OR]#

RewriteCond %{QUERY_STRING} ^.*( <|>|'|"%0A|%0D|%27|%3C|%3E|).)*
[NC]

RewriteRule ^(.*)$ hack.cfm
```



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

عوض کردن ایندکس فولدري خاص

اکثر میزبانهای وب فایلی که وظیفه لود شدن در هنگام باز کردن دایرکتوری خاصی را دارد با نام `index` قرار میدهند که با استفاده از `htaccess` میتوانید انرا عوض کنید . خط زیر را به فایل اضافه کنید :

`DirectoryIndex filename.html`

حتما دیده اید که اگر فایل `index.html` در دایرکتوری نبود سرور فایل دیگری مثلا `index.php` را اجرا میکند

روش به اینصورت است که در دستور فوق بعد از `filename.html` یا یک فضای خالی نامهای دیگر را وارد میکنید .

بنابراین سرور در دایرکتوری ویژه ابتدا به دنبال فایل اولی و سپس به دنبال فایل دومی و به همین ترتیب جستجو میکند .

اگر هم هیچ یک از فایلها پیدا نشد پیغام خطای `۴۰۴` ظاهر میشود .



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

htaccess توسط Redirec

حتما میدانید که راههای زیادی برای ریدایرکت کردن فایل‌های خاص به فایل دیگر وجود دارد اما بهترین کار همین است که می‌گوییم

در فایل خط زیر را وارد کنید :

```
Redirect /OldDir/OldFile.html http://www.domainname.com/newdir
```

بنابراین دستور به شرح زیر است

```
Redirect oldlocation newlocation
```

منتهی یادتان باشد که **newlocation** باید نام ادرس کامل سایت باشد .

دسترسی کاربران را بسته به **User-Agent Header** محدود کنید

این روش می‌تواند در مصرف پهنای باند شما صرفه جویی کند، جلوی بعضی از **spider** و **bot** ها را می‌گیرد :

کد:

```
#block visitors referred from indicated domains
<IfModule mod_rewrite.c>
SetEnvIfNoCase ^User-Agent$
.*(craftbot|download|extract|stripper|sucker|ninja|clshhttp|webspider|leache
r|col lector|grabber|webpictures) HTTP_SAFE_BADBOT
SetEnvIfNoCase ^User-Agent$.*(libwww-perl|aesop_com_spiderman)
HTTP_SAFE_BADBOT
Deny from env=HTTP_SAFE_BADBOT
```



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

</ifModule>

مسدود کردن مرورگر های آفلاین و روبات های بد !

مرورگرهای آفلاین نرم افزاری با سرعت بالا برای گرفتن اطلاعات از اینترنت است. این برنامه به شما امکان دریافت کامل اطلاعات یک سایت را به صورت طبقه بندی شده می دهد !!!

روبات های بد یا همون Bad Robots هم اسپایدر هایی هستن که باعث خرابکاری می شن و خلاصه برای سایتتون مفید نیستند !

شما می تونید به وسیله ی کد هایی که در پایین بهتون می دم از این ها جلوگیری کنید و به نوعی شر مزاحم های سایت خودتون رو کم کنید !!!

به روت سایت خود رفته و htaccess رو باز کنید !

دقت کنید شاید این فایل مخفی شده باشد .

این کد ها رو در آخرش اضافه کنید !

کد:

```
RewriteEngine On
RewriteCond %{HTTP_USER_AGENT} ^BlackWidow [OR]
RewriteCond %{HTTP_USER_AGENT} ^Bot\ mailto:craftbot@yahoo.com
[OR]
RewriteCond %{HTTP_USER_AGENT} ^ChinaClaw [OR]
```



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

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]



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

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



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

```
RewriteCond %{HTTP_USER_AGENT} ^Zeus  
RewriteRule ^.* - [F,L]
```

جلوگیری از اجرای فایل در پوشه های با پرمیژن (دسترسی) 777

اگر در سایتتون یک پوشه داشتید که پرم 777 بود و میخوايد جلوی خطر احتمالی رو بگیرید

یک .htaccess باز کنید و توش اینو بزنید .

کد PHP:

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

جلوگیری از حملات LFI



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

حدود ۷۰ تا ۸۰ درصد از وبسایت ها مقابل حملات LFI آسیب پذیر هستند، هکر ها از این حفره جهت بهره برداری و گرفتن shell استفاده میکنند، شما میتوانید از طریق htaccess وبسایت خود را مقابل این حملات ایمن سازی کنید.

در زیر میتوانید یک نمونه از کد php آسیب پذیر در مقابل حملات LFI را مشاهده کنید:
کد:

```
<?php?>
LFI Vulnerable Code //
;redirect = $_GET[redirect]$
;include($redirect)
<?>
```

کد زیر زیر به لطف ارائه SigSiu.net میتواند شما را مقابل این حملات ایمن کند.
کد:

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

در ادامه کد بالا (لوگکت) ممکن است بخواهد از URL encoding بجای http:// استفاده کنید، بدین جهت کد زیر را در خط دوم کد بالا جایگزین کنید:
کد:

```
RewriteCond %{QUERY_STRING} [a-zA-Z0-9_]=http%3A%2F%2F [OR]
```

افزودن کد زیر به فایل htaccess میتواند چند لایه امنیتی در مقابل حملاتی که با استفاده از proc/self/enviro method نفوذ پذیری را ممکن میکنند (لوگکت) اضافه کند:
کد:

```
!proc/self/enviro? no way#
```



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

```
RewriteCond %{QUERY_STRING} proc/self/venv [NC,OR]
```

کد زیر میتواند از شما در مقابل Apache HTTP Server CVE-2011-3192 Denial Of Service Vulnerability که نخستین بار در August 2011 کشف شد محافظ کند، البته این کد مخصوص کسانیست که امکان به روز رسانی سرور خود را ندارند، اگر از نسخه به روز تر از این تاریخ میزبانی میشوید، اجرای این کد لزومی ندارد.

کد:

```
#drop Range header when more than 5 ranges
#CVE-2011-3192
SetEnvIf Range (.{*?){5,} bad-range=1
RequestHeader unset Range env=bad-range
#optional logging
```

```
#CustomLog insert-path-and-name-of-log common env=bad-range
```

- استفاده از x-frame یکی از غیر اصولی ترین و خطرناک ترین کد هایست که شما میتوانید بر روی وبسایت خود اجرا کند و یا اگر از سیستم های مدیریت محتوا استفاده میکنید به کاربران خود اجازه دهید تا از این کد استفاده کنند، قصد آموزش اینکه این کد چطور میتواند حفره ای برای نفوذ باشد را در این مقاله نداریم و صرفاً به شما آموزش میدهم که چطور میتوانید از اجرا شدن این کد با اسفاده از htaccess جلوگیری کنید:

کد:

```
Don't allow any pages to be framed - Defends against CSRF #
Header set X-Frame-Options DENY
```

در ادامه کد بالا که اجرای کد x-frame را غیر فعال میکند، متوانید به جای DENY از SAMEORIGIN جهت اجازه دادن کد x-frame ی که تنها از سایت خودتان اجرا میشود، استفاده کنید.

بدین ترتیب تنها صفحات داخلی وبسایت شما اجازه قرار گرفتن در کد x-frame را خواهند داشت.



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

در نسخه های ۸ و ۹ مرورگر اینترنت اکسپلورر با استفاده از کد زیر از cross site scripting محافظت کنید.

توجه: استفاده از این کد ممکن است کارایی برخی از وبسایت ها را مختل کند، اگر این مشکل برای شما پیش آمد کفایت تا مقدار 1 را به 0 تغییر دهید.

کد:

```
#Turn on IE8-IE9 XSS prevention tools  
"Header set X-XSS-Protection "1; mode=block
```

Content Security Policy –

یک لایه افزوده شده امنیت ایست که میتواند شما را در مقابل حملاتی نظیر Cross Site Scripting و data injection ایمن کند، کد زیر اجرای جاوا کد های خارجی را محدود میکند و اجازه اجرا را به آنها نمیدهد، اما اگر در وبسایت خود از سرویسهای نظیر Google Analytics استفاده میکنید این کد جلوی کارایی آن را میگیرد چرا که استفاده از اینگونه سرویسها مستلزم اجرای جاوا کد میباشد.

کد:

```
Only allow JavaScript from the same domain to be run #  
.Don't allow inline JavaScript to run #  
";Header set X-Content-Security-Policy "allow 'self
```

کد زیر جلوی کاراکترهای غیر استاندارد را خواهد گرفت و این عمل میتواند شما در مقابل حملاتی که با درخواست HTTP نفوذ پذیری را ممکن میکند ایمن کند.

کد:

```
#Prevent use of specified methods in HTTP Request  
RewriteCond %{REQUEST_METHOD} ^(HEAD|TRACE|DELETE|TRACK)  
[NC,OR]
```



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

```

#Block out use of illegal or unsafe characters in the HTTP Request
RewriteCond %{THE_REQUEST} ^.*(\|\r|\|n|%0A|%0D).* [NC,OR]
#Block out use of illegal or unsafe characters in the Referer Variable of the
HTTP Request
RewriteCond %{HTTP_REFERER}
^.*(;<|>|'|"%0A|%0D|%27|%3C|%3E|%00).* [NC,OR]
#Block out use of illegal or unsafe characters in any cookie associated with
the HTTP Request
RewriteCond %{HTTP_COOKIE} ^.*(;<|>|'|"%0A|%0D|%27|%3C|%3E|%00).*
[NC,OR]
#Block out use of illegal characters in URI or use of malformed URI
RewriteCond %{REQUEST_URI} ^/(\.|\:|\:|<|>|'|">|'|"%0A|%0D|%27|%3C|%3E|%00).{0,9999}.*
[NC,OR]
#Block out use of empty User Agent Strings
#NOTE - disable this rule if your site is integrated with Payment Gateways
such as PayPal
RewriteCond %{HTTP_USER_AGENT} ^$ [OR]
#Block out use of illegal or unsafe characters in the User Agent variable
RewriteCond %{HTTP_USER_AGENT}
^.*(;<|>|'|"%0A|%0D|%27|%3C|%3E|%00).* [NC,OR]
#Measures to block out SQL injection attacks
RewriteCond %{QUERY_STRING}
^.*(;<|>|'|"%0A|%0D|%22|%27|%3C|%3E|%00).*(^*|union|select|insert|c
[ast|set|declare|drop|update|md5|benchmark).* [NC,OR]
#Block out reference to localhost/loopback/127.0.0.1 in the Query String
RewriteCond %{QUERY_STRING} ^.*(localhost|loopback|127\.\0\.\0\.\1).*
[NC,OR]
#Block out use of illegal or unsafe characters in the Query String variable
RewriteCond %{QUERY_STRING}
^.*(;<|>|'|"%0A|%0D|%27|%3C|%3E|%00).* [NC]

```

کد htaccess غیر فعال کردن TRACE and TRACK



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

TRACE and TRACK یک روش درخواست هست که باعث میشه اطلاعاتی که از طرف client

یا کاربر به ibm http server ارسال میشه ، مجددا به client برگردانده میشه یک مثال ساده :

```
$telnet 127.0.0.1 8080 Trying... Connected to 127.0.0.1. Escape character is '^]'. TRACE / HTTP/1.0 Host: foo A: b C: d HTTP/1.1 200 OK Date: Mon, 04 Oct 2004 14:07:59 GMT Server: IBM_HTTP_SERVER Connection: close Content-Type: message/http TRACE / HTTP/1.0 A: b C: d Host: foo .Connection closed
```

```
#disable TRACE and TRACK in the main scope of httpd.conf
RewriteEngine On RewriteCond %{REQUEST_METHOD} ^TRACE
RewriteRule .* - [F] RewriteCond %{REQUEST_METHOD} ^TRACK
RewriteRule .* - [F] ... <VirtualHost www.example.com> ... # disable
TRACE and TRACK in the www.example.com virtual host RewriteEngine
On RewriteCond %{REQUEST_METHOD} ^TRACE RewriteRule .* - [F]
RewriteCond %{REQUEST_METHOD} ^TRACK RewriteRule .* - [F]
# </VirtualHost>
```

مجبور کردن مرور گر برای تغییر مسیر از HTTP به https

در حالت عادی آدرس سایت شما <http://www.yoursite.com> هست یعنی بدون رمزنگاری

اطلاعات ما بین کاربر و سرور و با پورت 80 سایت بالا میاد

وقتی شما مجوز SSL یا secure locket layer رو از یکی از شرکت های ارائه دهنده اس

میخرید(اس اس ال امکان رمز نگاری اطلاعات رد و بدل شده بین کاربر و سرور رو میده جهت

افزایش امنیت اطلاعات و جلوگیری از



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

هک شدن که خودش یه بحث مفصله و سعی میکنم داخل یک تاپیک حداکانه در مورد پروتکل اس اس ال توضیح بدم)

در واقع سایت شما با دو پروتکل قابل دسترسی هست یکی http که برای همه سایت ها پیشفرض هست و دوم https که با پروت 443 بالا میاد و وقتی مجوز ssl رو میخرید براتون فعال میشه و کاربر به دو روش میتونه به سایتتون دسترسی داشته باشه

در حالت عادی وقتی آدرس سایتتونو تایپ میکنه با پروتکل http سایت بالا میاد بوسیله فایل htaccess این امکان هست که بصورت پیشفرض هر وقت کسی آدرس سایت شما رو وارد کرد با پروتکل https سایت لود بشه چرا که ممکنه کاربر ندونه سایت شما از این پروتکل پشتیبانی میکنه(در صورتی که یک مجوز اس اس ال خریده و فعال کرده باشید) خوب برای اینکه آدرس از http به https بصورت پیشفرض تغییر کنه راه های دیگه ای هم هست مثلا از داخل پنل مدیریت جوملا ، برخی افزونه ها و.. اما اینم بعنوان یک روش تکمیلی و ساده مورد استفاده قرار میگیره

برای اینکار داخل فایل htaccess روت سایت این کد رو وارد کنید و قبلش مطمئن بشید که ماژول RewriteEngine داخل سرور فعال هست.

کد:

```
RewriteEngine on RewriteCond %{SERVER_PORT} !^443$ RewriteRule ^/(.*) https://%{SERVER_NAME}/$1 [NC,R,L]
```



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

و در انتها

کد کامل Htaccess

این کد ها برای کسانی که **Vbseo + Sitemap** دارند :

کد:

```
#Comment the following line (add '#' at the beginning)
#to disable mod_rewrite functions

#Please note: you still need to disable the hack in
#the vBSEO control panel to stop url rewrites

RewriteEngine On

#Some servers require the Rewritebase directive to be
#enabled (remove '#' at the beginning to activate)
#Please note: when enabled, you must include the path
#to your root vB folder (i.e. RewriteBase /forums/)
/ RewriteBase#

#RewriteCond %{HTTP_HOST} !^www\.yourdomain\.com
#RewriteRule (.*) http://www.yourdomain.com/forums/$1 [L,R=301]

$RewriteRule ^((urllist|sitemap_)\.?(xml|txt)(\.gz)?)
vbseo_sitemap/vbseo_getsitemap.php?sitemap=$1 [L]

RewriteCond %{REQUEST_URI}
!(admincp/|modcp/|cron|vbseo_sitemap|api\.php)
RewriteRule ^((archive/)?(.*\.php(/.*)?))$ vbseo.php [L,QSA]

RewriteCond %{REQUEST_FILENAME} !-f
```



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

```
RewriteCond %{REQUEST_FILENAME} !-d
RewriteCond %{REQUEST_URI}
!(admincp/|modcp/|cron|vbseo_sitemap|api|redir\.\php)
RewriteRule ^(.+)\$ vbseo.php [L,QSA]
RewriteEngine On
$RewriteRule ^((urllist|sitemap)\.\.(xml|txt)(\.\gz)?
vbseo_sitemap/vbseo_getsitemap.php?sitemap=$1 [L]
```

```
#BEGIN Compress text files
<ifModule mod_deflate.c>
<"$filesMatch "\.(css|js|x?html?|php)>
SetOutputFilter DEFLATE
<filesMatch/>
</ifModule/>
#END Compress text files
```

```
#BEGIN Expire headers
<ifModule mod_expires.c>
ExpiresActive On
"ExpiresDefault "access plus 1 seconds
"ExpiresByType image/x-icon "access plus 2592000 seconds
"ExpiresByType image/jpeg "access plus 2592000 seconds
"ExpiresByType image/png "access plus 2592000 seconds
"ExpiresByType image/gif "access plus 2592000 seconds
ExpiresByType application/x-shockwave-flash "access plus 2592000
"seconds
"ExpiresByType text/css "access plus 604800 seconds
"ExpiresByType text/javascript "access plus 216000 seconds
"ExpiresByType application/javascript "access plus 216000 seconds
"ExpiresByType application/x-javascript "access plus 216000 seconds
"ExpiresByType text/html "access plus 600 seconds
"ExpiresByType application/xhtml+xml "access plus 600 seconds
</ifModule/>
```



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

#END Expire headers

#BEGIN Cache-Control Headers

```
<ifModule mod_headers.c>
<"$filesMatch "\.(ico|jpe?g|png|gif|swf)>
"Header set Cache-Control "max-age=2592000, public
<filesMatch/>
<"$filesMatch "\.(css)>
"Header set Cache-Control "max-age=604800, public
<filesMatch/>
<"$filesMatch "\.(js)>
"Header set Cache-Control "max-age=216000, private
<filesMatch/>
<"$filesMatch "\.(x?html?|php)>
"Header set Cache-Control "max-age=600, private, must-revalidate
<filesMatch/>
</ifModule/>
```

#END Cache-Control Headers

#BEGIN Turn ETags Off

```
<ifModule mod_headers.c>
Header unset ETag
</ifModule/>
```

FileETag None

#END Turn ETags Off

BEGIN Remove Last-Modified Header #

```
<ifModule mod_headers.c>
Header unset Last-Modified
</ifModule/>
```

#END Remove Last-Modified Header



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

```
<!--Module mod_mime.c-->
AddType text/css .css
AddType application/x-javascript .js
AddType text/richtext .rtf .rtx
AddType image/svg+xml .svg .svgz
AddType text/plain .txt
AddType text/xsd .xsd
AddType text/xsl .xsl
AddType video/asf .asf .asx .wax .wmv .wmx
AddType video/avi .avi
AddType image/bmp .bmp
AddType application/java .class
AddType video/divx .divx
AddType application/msword .doc .docx
AddType application/x-msdownload .exe
AddType image/gif .gif
AddType application/x-gzip .gz .gzip
AddType image/x-icon .ico
AddType image/jpeg .jpg .jpeg .jpe
AddType application/vnd.ms-access .mdb
AddType audio/midi .mid .midi
AddType video/quicktime .mov .qt
AddType audio/mpeg .mp3 .m4a
AddType video/mp4 .mp4 .m4v
AddType video/mpeg .mpeg .mpg .mpe
AddType application/vnd.ms-project .mpp
AddType application/vnd.oasis.opendocument.database .odb
AddType application/vnd.oasis.opendocument.chart .odc
AddType application/vnd.oasis.opendocument.formula .odf
AddType application/vnd.oasis.opendocument.graphics .odg
AddType application/vnd.oasis.opendocument.presentation .odp
AddType application/vnd.oasis.opendocument.spreadsheet .ods
AddType application/vnd.oasis.opendocument.text .odt
AddType audio/ogg .ogg
AddType application/pdf .pdf
AddType image/png .png
```



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

```

AddType application/vnd.ms-powerpoint .pot .pps .ppt .pptx
AddType audio/x-realaudio .ra .ram
AddType application/x-shockwave-flash .swf
AddType application/x-tar .tar
AddType image/tiff .tif .tiff
AddType audio/wav .wav
AddType audio/wma .wma
AddType application/vnd.ms-write .wri
AddType application/vnd.ms-excel .xla .xls .xlsx .xlt .xlw
AddType application/zip .zip
</IfModule>

<IfModule mod_expires.c>
ExpiresActive On
"ExpiresByType text/css "access plus 1 month
"ExpiresByType application/x-javascript "access plus 1 month
"ExpiresByType text/richtext "access plus 1 day
"ExpiresByType image/svg+xml "access plus 1 day
"ExpiresByType text/plain "access plus 1 day
"ExpiresByType text/xsd "access plus 1 day
"ExpiresByType text/xsl "access plus 1 day
"ExpiresByType video/asf "access plus 1 month
"ExpiresByType video/avi "access plus 1 month
"ExpiresByType image/bmp "access plus 1 month
"ExpiresByType application/java "access plus 1 month
"ExpiresByType video/divx "access plus 1 month
"ExpiresByType application/msword "access plus 1 month
"ExpiresByType application/x-msdownload "access plus 1 month
"ExpiresByType image/gif "access plus 1 month
"ExpiresByType application/x-gzip "access plus 1 month
"ExpiresByType image/x-icon "access plus 1 month
"ExpiresByType image/jpeg "access plus 1 month
"ExpiresByType application/vnd.ms-access "access plus 1 month
"ExpiresByType audio/midi "access plus 1 month
"ExpiresByType video/quicktime "access plus 1 month
"ExpiresByType audio/mpeg "access plus 1 month
"ExpiresByType video/mp4 "access plus 1 month

```



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

```

"ExpiresByType video/mpeg "access plus 1 month
"ExpiresByType application/vnd.ms-project "access plus 1 month
ExpiresByType application/vnd.oasis.opendocument.database "access plus
"1 month
ExpiresByType application/vnd.oasis.opendocument.chart "access plus 1
"month
ExpiresByType application/vnd.oasis.opendocument.formula "access plus
"1 month
ExpiresByType application/vnd.oasis.opendocument.graphics "access plus
"1 month
ExpiresByType application/vnd.oasis.opendocument.presentation "access
"plus 1 month
ExpiresByType application/vnd.oasis.opendocument.spreadsheet "access
"plus 1 month
ExpiresByType application/vnd.oasis.opendocument.text "access plus 1
"month
"ExpiresByType audio/ogg "access plus 1 month
"ExpiresByType application/pdf "access plus 1 month
"ExpiresByType image/png "access plus 1 month
"ExpiresByType application/vnd.ms-powerpoint "access plus 1 month
"ExpiresByType audio/x-realaudio "access plus 1 month
"ExpiresByType application/x-shockwave-flash "access plus 1 month
"ExpiresByType application/x-tar "access plus 1 month
"ExpiresByType image/tiff "access plus 1 month
"ExpiresByType audio/wav "access plus 1 month
"ExpiresByType audio/wma "access plus 1 month
"ExpiresByType application/vnd.ms-write "access plus 1 month
"ExpiresByType application/vnd.ms-excel "access plus 1 month
"ExpiresByType application/zip "access plus 1 month
<IfModule>
<IfModule mod_deflate.c>
<IfModule mod_setenvif.c>
BrowserMatch ^Mozilla/4 gzip-only-text/html
BrowserMatch ^Mozilla/4.0[678] no-gzip
BrowserMatch bMSIE !no-gzip !gzip-only-text/html
BrowserMatch bMSI[E] !no-gzip !gzip-only-text/html

```



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly


```

</IfModule/>
<IfModule mod_headers.c>
Header append Vary User-Agent env=!dont-vary
</IfModule/>
<IfModule mod_.c>
AddOutputByType DEFLATE text/css application/x-javascript text/html
text/richtext image/svg+xml text/plain text/xsd text/xsl text/xml image/x-icon
</IfModule/>
<IfModule/>
<FilesMatch "\.(css|js|CSS|JS)>
<IfModule mod_headers.c>
"Header set Pragma "public
"Header append Cache-Control "public, must-revalidate, -revalidate
</IfModule/>
FileETag MTime Size
<IfModule mod_headers.c>
"Header set X-Powered-By "W3 Total Cache/0.9.1.3
</IfModule/>
<FilesMatch/>
FilesMatch >
<"$"\.(rtf|rtx|svg|svgz|txt|xsd|xsl|RTF|RTX|SVG|SVGZ|TXT|XSD|XSL)
<IfModule mod_headers.c>
"Header set Pragma "public
"Header append Cache-Control "public, must-revalidate, -revalidate
</IfModule/>
FileETag MTime Size
<IfModule mod_headers.c>
"Header set X-Powered-By "W3 Total Cache/0.9.1.3
</IfModule/>
<FilesMatch/>
FilesMatch >
"\.(asf|asx|wax|wmv|wmx|avi|bmp|class|divx|doc|docx|exe|gif|gz|gzip|ico|jpg

```



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

```
|jpeg|jpe|mdb|mid|midi|mov|qt|mp3|m4a|mp4|m4v|mpeg|mpg|mpe|mpp|odb  
|odc|odf|odg|odp|ods|odt|ogg|pdf|png|pot|pps|ppt|pptx|ra|ram|swf|tar|tif|tiff|w  
av|wma|wri|xla|xls|xlsx|xlt|xlw|zip|ASF|ASX|WAX|WMV|WMX|AVI|BMP|CLA  
SS|DIVX|DOC|DOCX|EXE|GIF|GZ|GZIP|ICO|JPG|JPEG|JPE|MDB|MID|MI  
DI|MOV|QT|MP3|M4A|MP4|M4V|MPEG|MPG|MPE|MPP|ODB|ODC|ODF|O  
DG|ODP|ODS|ODT|OGG|PDF|PNG|POT|PPS|PPT|PPTX|RA|RAM|SWF|T  
<"$AR|TIF|TIFF|WAV|WMA|WRI|XLA|XLS|XLSX|XLT|XLW|ZIP)
```

```
<IfModule mod_headers.c>  
"Header set Pragma "public  
"Header append Cache-Control "public, must-revalidate, -revalidate  
</IfModule/>
```

```
FileETag MTime Size  
<IfModule mod_headers.c>  
"Header set X-Powered-By "W3 Total Cache/0.9.1.3  
</IfModule/>
```

```
<FilesMatch/>
```

```
<IfModule mod_php4.c>  
php_value upload_max_filesize 2M  
php_value max_execution_time 30  
php_value max_input_time 60  
php_value memory_limit 32M  
php_value post_max_size 8M  
php_flag register_globals off  
php_flag display_errors off  
php_flag file_uploads on  
php_flag log_errors off  
php_flag output_buffering off  
php_flag register_argc_argv on  
php_flag magic_quotes_gpc off  
php_flag magic_quotes_runtime off  
php_flag magic_quotes_sybase off  
php_flag mysql.allow_persistent off  
php_flag register_long_arrays on  
php_flag allow_url_fopen on
```



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

```
php_flag cgi.force_redirect on
php_flag enable_dl on
</module>
```

```
<!module mod_php5.c>
php_value upload_max_filesize 2M
php_value max_execution_time 30
php_value max_input_time 60
php_value memory_limit 32M
php_value post_max_size 8M
php_flag register_globals off
php_flag display_errors off
php_flag file_uploads on
php_flag log_errors off
php_flag output_buffering off
php_flag register_argc_argv on
php_flag magic_quotes_gpc off
php_flag magic_quotes_runtime off
php_flag magic_quotes_sybase off
php_flag mysql.allow_persistent off
php_flag register_long_arrays on
php_flag allow_url_fopen on
php_flag cgi.force_redirect on
php_flag enable_dl on
</module>
```

کسایى هم كه وى بى سئو و سايت مپ نډارن اين رو آخر **htaccess** خودشون اضافه كنن .

ك:

```
BEGIN Compress text files #
<!Module mod_deflate.c>
<"$filesMatch "\.(css|js|x?html?|php)>
SetOutputFilter DEFLATE
```



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

```

<filesMatch/>
<ifModule/>
END Compress text files #

BEGIN Expire headers #
<ifModule mod_expires.c>
ExpiresActive On
ExpiresDefault "access plus 1 seconds"
ExpiresByType image/x-icon "access plus 2592000 seconds"
ExpiresByType image/jpeg "access plus 2592000 seconds"
ExpiresByType image/png "access plus 2592000 seconds"
ExpiresByType image/gif "access plus 2592000 seconds"
ExpiresByType application/x-shockwave-flash "access plus 2592000
seconds"
ExpiresByType text/css "access plus 604800 seconds"
ExpiresByType text/javascript "access plus 216000 seconds"
ExpiresByType application/javascript "access plus 216000 seconds"
ExpiresByType application/x-javascript "access plus 216000 seconds"
ExpiresByType text/html "access plus 600 seconds"
ExpiresByType application/xhtml+xml "access plus 600 seconds"
</ifModule/>
END Expire headers #

BEGIN Cache-Control Headers #
<ifModule mod_headers.c>
<"$filesMatch "\.(ico|jpe?g|png|gif|swf)">
"Header set Cache-Control "max-age=2592000, public"
</filesMatch/>
<"$filesMatch "\.(css)">
"Header set Cache-Control "max-age=604800, public"
</filesMatch/>
<"$filesMatch "\.(js)">
"Header set Cache-Control "max-age=216000, private"
</filesMatch/>

```



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

```
<"$filesMatch "\.(x?html?|php)>
"Header set Cache-Control "max-age=600, private, must-revalidate
</filesMatch/>
</ifModule/>
END Cache-Control Headers #

BEGIN Turn ETags Off #
</ifModule mod_headers.c>
Header unset ETag
</ifModule/>
FileETag None
END Turn ETags Off #

BEGIN Remove Last-Modified Header #
</ifModule mod_headers.c>
Header unset Last-Modified
</ifModule/>
END Remove Last-Modified Header #

</IfModule mod_mime.c>
AddType text/css .css
AddType application/x-javascript .js
AddType text/richtext .rtf .rtx
AddType image/svg+xml .svg .svgz
AddType text/plain .txt
AddType text/xsd .xsd
AddType text/xsl .xsl
AddType video/asf .asf .asx .wax .wmv .wmx
AddType video/avi .avi
AddType image/bmp .bmp
AddType application/java .class
AddType video/divx .divx
AddType application/msword .doc .docx
AddType application/x-msdownload .exe
```



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

```
AddType image/gif .gif
AddType application/x-gzip .gz .gzip
AddType image/x-icon .ico
AddType image/jpeg .jpg .jpeg .jpe
AddType application/vnd.ms-access .mdb
AddType audio/midi .mid .midi
AddType video/quicktime .mov .qt
AddType audio/mpeg .mp3 .m4a
AddType video/mp4 .mp4 .m4v
AddType video/mpeg .mpeg .mpg .mpe
AddType application/vnd.ms-project .mpp
AddType application/vnd.oasis.opendocument.database .odb
AddType application/vnd.oasis.opendocument.chart .odc
AddType application/vnd.oasis.opendocument.formula .odf
AddType application/vnd.oasis.opendocument.graphics .odg
AddType application/vnd.oasis.opendocument.presentation .odp
AddType application/vnd.oasis.opendocument.spreadsheet .ods
AddType application/vnd.oasis.opendocument.text .odt
AddType audio/ogg .ogg
AddType application/pdf .pdf
AddType image/png .png
AddType application/vnd.ms-powerpoint .pot .pps .ppt .pptx
AddType audio/x-realaudio .ra .ram
AddType application/x-shockwave-flash .swf
AddType application/x-tar .tar
AddType image/tiff .tif .tiff
AddType audio/wav .wav
AddType audio/wma .wma
AddType application/vnd.ms-write .wri
AddType application/vnd.ms-excel .xla .xls .xlsx .xlt .xlw
AddType application/zip .zip
</IfModule/>

<IfModule mod_expires.c>
ExpiresActive On
"ExpiresByType text/css "access plus 1 month
"ExpiresByType application/x-javascript "access plus 1 month
```



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

"ExpiresByType text/richtext "access plus 1 day
"ExpiresByType image/svg+xml "access plus 1 day
"ExpiresByType text/plain "access plus 1 day
"ExpiresByType text/xsd "access plus 1 day
"ExpiresByType text/xsl "access plus 1 day
"ExpiresByType video/asf "access plus 1 month
"ExpiresByType video/avi "access plus 1 month
"ExpiresByType image/bmp "access plus 1 month
"ExpiresByType application/java "access plus 1 month
"ExpiresByType video/divx "access plus 1 month
"ExpiresByType application/msword "access plus 1 month
"ExpiresByType application/x-msdownload "access plus 1 month
"ExpiresByType image/gif "access plus 1 month
"ExpiresByType application/x-gzip "access plus 1 month
"ExpiresByType image/x-icon "access plus 1 month
"ExpiresByType image/jpeg "access plus 1 month
"ExpiresByType application/vnd.ms-access "access plus 1 month
"ExpiresByType audio/midi "access plus 1 month
"ExpiresByType video/quicktime "access plus 1 month
"ExpiresByType audio/mpeg "access plus 1 month
"ExpiresByType video/mp4 "access plus 1 month
"ExpiresByType video/mpeg "access plus 1 month
"ExpiresByType application/vnd.ms-project "access plus 1 month
ExpiresByType application/vnd.oasis.opendocument.database "access plus
"1 month
ExpiresByType application/vnd.oasis.opendocument.chart "access plus 1
"month
ExpiresByType application/vnd.oasis.opendocument.formula "access plus
"1 month
ExpiresByType application/vnd.oasis.opendocument.graphics "access plus
"1 month
ExpiresByType application/vnd.oasis.opendocument.presentation "access
"plus 1 month
ExpiresByType application/vnd.oasis.opendocument.spreadsheet "access
"plus 1 month



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

```

ExpiresByType application/vnd.oasis.opendocument.text "access plus 1
"month
"ExpiresByType audio/ogg "access plus 1 month
"ExpiresByType application/pdf "access plus 1 month
"ExpiresByType image/png "access plus 1 month
"ExpiresByType application/vnd.ms-powerpoint "access plus 1 month
"ExpiresByType audio/x-realaudio "access plus 1 month
"ExpiresByType application/x-shockwave-flash "access plus 1 month
"ExpiresByType application/x-tar "access plus 1 month
"ExpiresByType image/tiff "access plus 1 month
"ExpiresByType audio/wav "access plus 1 month
"ExpiresByType audio/wma "access plus 1 month
"ExpiresByType application/vnd.ms-write "access plus 1 month
"ExpiresByType application/vnd.ms-excel "access plus 1 month
"ExpiresByType application/zip "access plus 1 month
</IfModule/>

<IfModule mod_deflate.c>
<IfModule mod_setenvif.c>
BrowserMatch ^Mozilla/4 gzip-only-text/html
BrowserMatch ^Mozilla/4.0[678] no-gzip
BrowserMatch bMSIE !no-gzip !gzip-only-text/html
BrowserMatch bMSI[E] !no-gzip !gzip-only-text/html
</IfModule/>

<IfModule mod_headers.c>
Header append Vary User-Agent env=!dont-vary
</IfModule/>

<IfModule mod_c>
AddOutputByType DEFLATE text/css application/x-javascript text/html
text/richtext image/svg+xml text/plain text/xsd text/xsl text/xml image/x-icon
</IfModule/>

</IfModule/>

<"$FilesMatch "\.(css|js|CSS|JS)">
<IfModule mod_headers.c>
"Header set Pragma "public

```



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly


```

"Header append Cache-Control "public, must-revalidate, -revalidate
</IfModule/>
FileETag MTime Size
</IfModule mod_headers.c>
"Header set X-Powered-By "W3 Total Cache/0.9.1.3
</IfModule/>
<FilesMatch/>
FilesMatch >
<"$. (rtf|rtx|svg|svgz|txt|xsd|xsl|RTF|RTX|SVG|SVGZ|TXT|XSD|XSL)
</IfModule mod_headers.c>
"Header set Pragma "public
"Header append Cache-Control "public, must-revalidate, -revalidate
</IfModule/>
FileETag MTime Size
</IfModule mod_headers.c>
"Header set X-Powered-By "W3 Total Cache/0.9.1.3
</IfModule/>
<FilesMatch/>
FilesMatch >
"$. (asf|asx|wax|wmv|wmx|avi|bmp|class|divx|doc|docx|exe|gif|gz|gzip|ico|jpg
|jpeg|jpe|mdb|mid|midi|mov|qt|mp3|m4a|mp4|m4v|mpeg|mpg|mpe|mpp|odb
|odc|odf|odg|odp|ods|odt|ogg|pdf|png|pot|pps|ppt|pptx|ra|ram|swf|tar|tif|tiff|w
av|wma|wri|xla|xls|xlsx|xlt|xlw|zip|ASF|ASX|WAX|WMV|WMX|AVI|BMP|CLA
SS|DIVX|DOC|DOCX|EXE|GIF|GZ|GZIP|ICO|JPG|JPEG|JPE|MDB|MID|MI
DI|MOV|QT|MP3|M4A|MP4|M4V|MPEG|MPG|MPE|MPP|ODB|ODC|ODF|O
DG|ODP|ODS|ODT|OGG|PDF|PNG|POT|PPS|PPT|PPTX|RA|RAM|SWF|T
<"$AR|TIF|TIFF|WAV|WMA|WRI|XLA|XLS|XLSX|XLT|XLW|ZIP)
</IfModule mod_headers.c>
"Header set Pragma "public
"Header append Cache-Control "public, must-revalidate, -revalidate
</IfModule/>
FileETag MTime Size
</IfModule mod_headers.c>

```



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

"Header set X-Powered-By "W3 Total Cache/0.9.1.3

<IfModule/>

<FilesMatch/>

<Ifmodule mod_php4.c>

php_value upload_max_filesize 2M

php_value max_execution_time 30

php_value max_input_time 60

php_value memory_limit 32M

php_value post_max_size 8M

php_flag register_globals off

php_flag display_errors off

php_flag file_uploads on

php_flag log_errors off

php_flag output_buffering off

php_flag register_argc_argv on

php_flag magic_quotes_gpc off

php_flag magic_quotes_runtime off

php_flag magic_quotes_sybase off

php_flag mysql.allow_persistent off

php_flag register_long_arrays on

php_flag allow_url_fopen on

php_flag cgi.force_redirect on

php_flag enable_dl on

</ifmodule/>

<Ifmodule mod_php5.c>

php_value upload_max_filesize 2M

php_value max_execution_time 30

php_value max_input_time 60

php_value memory_limit 32M

php_value post_max_size 8M

php_flag register_globals off

php_flag display_errors off

php_flag file_uploads on



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

```
php_flag log_errors off
php_flag output_buffering off
php_flag register_argc_argv on
php_flag magic_quotes_gpc off
php_flag magic_quotes_runtime off
php_flag magic_quotes_sybase off
php_flag mysql.allow_persistent off
php_flag register_long_arrays on
php_flag allow_url_fopen on
php_flag cgi.force_redirect on
php_flag enable_dl on
</ifmodule>
```

ابزارهای تولید خودکار دستورات .htaccess

حال که با دستورات اصلی فایل .htaccess آشنا شدید فصد دارم تعدادی از ابزارهای تولید خودکار دستورات را به شما معرفی کنم:

Htaccess Builder

<http://htaccessbuilder.com>



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly



htaccess redirect generator.

<http://www.htaccessredirect.net>



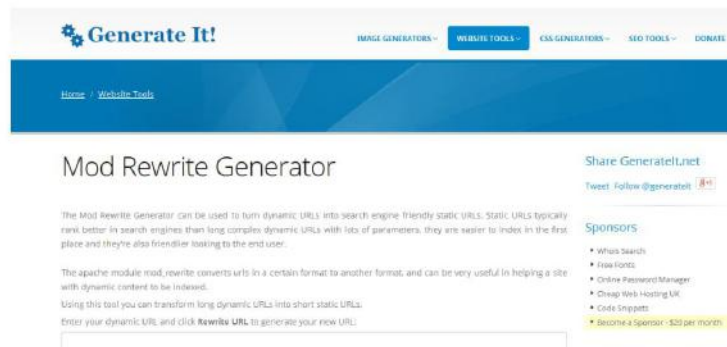
htaccessEditor – Create a .htaccess File

<http://www.htaccesseditor.com/en.shtml>



Mod Rewrite Generator by GenerateIt.net

<http://www.generateit.net/mod-rewrite/index.php>




@caffeinebookly


caffeinebookly


@caffeinebookly


caffeinebookly


t.me/caffeinebookly

منابع

<http://www.askapache.com/category/htaccess/>

https://www.branded3.com/blog/htaccess-mod_rewrite-ultimate-guide/



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly

<http://serverfault.com/questions/214512/redirect-change-urls-or-redirect-http-to-https-in-apache-everything-you-ever>
<http://wiki.apache.org/httpd/Htaccess>
<http://httpd.apache.org/docs/current/howto/htaccess.html>



@caffeinebookly



caffeinebookly



@caffeinebookly



caffeinebookly



t.me/caffeinebookly