Php.ini 配置文件详解

优化php性能的时候，优化配置文件这个是首当其冲的。下边跟大家分享下我总结的关于php.ini配置文件里的每一行的解释，希望能帮助大家进一步了解php。

[root@taokey ~]# grep -v ";" /application/php/lib/php.ini

[PHP] （PS：^\_^不错的php学习交流群：304224365，验证：ttf）

engine = On ——→ 是否启用PHP解析引擎

short\_open\_tag = Off ——→ 是否使用简介标志

asp\_tags = Off ——→ 不允许asp类标志

precision = 14 ——→ 浮点型数据显示的有效期

y2k\_compliance = On

####是否强制打开2000年适应(可能在非Y2K适应的浏览器中导致问题)。

output\_buffering = 4096 ——→ 输出缓冲区大小(字节)。建议值为4096~8192。

zlib.output\_compression = Off ——→ 是否开启zlib输出压缩

implicit\_flush = Off ——→ 是否要求PHP输出层在每个输出块之后自动刷新数据

这等效于在每个 print()、echo()、HTML块 之后自动调用flush()函数。打开这个选项对程序执行的性能有严重的影响，通常只推荐在调试时使用。在CLI SAPI的执行模式下，该指令默认为 On 。

unserialize\_callback\_func =

serialize\_precision = 17

####将浮点型和双精度型数据序列化存储时的精度(有效位数)。默认值能够确保浮点型数据被解序列化程序解码时不会丢失数据。

allow\_call\_time\_pass\_reference = Off

####是否强迫在函数调用时按引用传递参数(每次使用此特性都会收到一条警告)。

; php反对这种做法，并在将来的版本里不再支持，因为它影响到了代码的整洁。

; 鼓励的方法是在函数声明里明确指定哪些参数按引用传递。

; 我们鼓励你关闭这一选项，以保证你的脚本在将来版本的语言里仍能正常工作。

safe\_mode = Off ——→ 安全模式

safe\_mode\_gid = Off

safe\_mode\_include\_dir =

#在安全模式下，该组目录和其子目录下的文件被包含时，将跳过UID/GID检查。换句话说，如果此处的值为空，任何UID/GID不符合的文件都不允许被包含。这里设置的目录必须已经存在于include\_path指令中或者用完整路径来包含。多个目录之间用冒号(Win下为分号)隔开。指定的限制实际上是一个前缀，而非一个目录名。

safe\_mode\_exec\_dir = ——→ 安全模式下的可执行文件存放目录

safe\_mode\_allowed\_env\_vars = PHP\_

####在安全模式下，用户仅可以更改的环境变量的前缀列表(逗号分隔)。允许用户设置某些环境变量，可能会导致潜在的安全漏洞。注意: 如果这一参数值为空，PHP将允许用户更改任意环境变量。

safe\_mode\_protected\_env\_vars = LD\_LIBRARY\_PATH

####在安全模式下，用户不能更改的环境变量列表(逗号分隔)。这些变量即使在safe\_mode\_allowed\_env\_vars指令设置为允许的情况下也会得到保护。

disable\_functions = ——→ 该指令接受一个用逗号分隔的函数名列表，以禁用特定的函数。

disable\_classes = ——→ 该指令接受一个用逗号分隔的类名列表，以禁用特定的类

zend.enable\_gc = On ——→

expose\_php = On ——→ 在网页头部显示php信息

max\_execution\_time = 30 ——→ 每个脚本最大执行秒数

max\_input\_time = 60 ——→ 每个脚本用来分析请求数据的最大限制时间

memory\_limit = 128M ——→ 每个脚本执行的内存限制

error\_reporting = E\_ALL & ~E\_DEPRECATED

display\_errors = Off ——→ #显示失误（该关闭，换成日志显示）

display\_startup\_errors = Off ——→ #显示启动失误

log\_errors = On ——→ 生成错误错误日志显示

log\_errors\_max\_len = 1024 ——→ 设定error\_log最大长度

ignore\_repeated\_errors = Off ——→ 打开后，不记录重复的信息

ignore\_repeated\_source = Off ——→ 打开后当记录重复的信息时忽略来源

report\_memleaks = On ——→ 报告内存泄露，仅在debug编译模式下有效

track\_errors = Off ——→ ####在$php\_errormsg中保存最后一次错误/警告消息 (逻辑值).永远不要再生产环境中使用此特性：html\_errors 会显示php错误所在的html标签

html\_errors = Off ——→ 是否开启静态网页错误提示

variables\_order = "GPCS" ——→ ####此指令描述了PHP注册GET, POST, Cookie, 环境 和 内置变量的顺序 (各自使用G, P, C, E 和 S , 一般使用 EGPCS 或 GPC). 注册使用从左往右的顺序, 新的值会覆盖旧的值.

request\_order = "GP"

####此指令描述的顺序PHP注册GET，POST和COOKIE变量\_REQUEST数组。注册是由左到右，新的值将覆盖旧值。如果这个指令没有设置，variables\_order中使用$ \_REQUEST内容。请注意，默认分配的php.ini文件中不包含'C'饼干，出于安全方面的考虑。

register\_globals = Off ——→ ##是否打开register全局变量

register\_long\_arrays = Off

####是否注册老形式的输入数组, HTTP\_GET\_VARS 和相关数组；如果你不使用他们,建议为了提高性能关闭他们.

register\_argc\_argv = Off

####此指令让PHP确认是否申明 argv&argc 变量 (这些变量会包含GET信息). ;如果你不使用这些变量,为了提升性能应该关闭此选项.

auto\_globals\_jit = On

####当打开此项, SERVER 和 ENV 变量将在第一次被使用时而不是脚本一开始时创建(运行时);如果这些变量在脚本中没有被使用过, 打开此项会增加一点性能.;为了使此指令有效,PHP指令 register\_globals, register\_long\_arrays,;以及 register\_argc\_argv 必须被关闭.

post\_max\_size = 8M ——→ #PHP可以接受的最大的POST数据大小

magic\_quotes\_gpc = Off ——→ #针对GET/POST/Cookie数据打开Magic quotes.

magic\_quotes\_runtime = Off

####针对实时产生的数据打开Magic quotes,例如从SQL获取的数据, 从exec()返回的数据等等.

magic\_quotes\_sybase = Off ##使用 Sybase 风格的 magic quotes (使用"来引导'替代\').

auto\_prepend\_file = ——→ #在任何PHP文档之前或之后自动增加文件

auto\_append\_file =

####两个有趣的变量是auto\_prepend\_file以及auto\_append\_file。这些变量指定PHP自动添加到任何PHP文档文件头或文件尾的其他文件。这对于为PHP产生的页面添加页眉或页脚非常有用，可以节省为每个PHP文档添加代码的时间。但需要注意这里的指定文件将会添加到所有的PHP文档中，所以这些变量必须适合单应用程序（single-application）的服务器。所包含的文件要么是PHP脚本，要么是普通的HTML文档。嵌入式PHP代码必须用标准&lt;?php...?&gt;标记括起来。

default\_mimetype = "text/html" ——→ #PHP内建默认为text/html

doc\_root = ——→ #PHP的"根目录"。仅在非空时有效。

; 如果safe\_mode=On，则此目录之外的文件一概被拒绝。 ; 如果编译PHP时没有指定FORCE\_REDIRECT，并且在非IIS服务器上以CGI方式运行， ; 则必须设置此指令(参见手册中的安全部分)。 ; 替代方案是使用的cgi.force\_redirect指令

user\_dir =

####告诉php在使用 /~username 打开脚本时到哪个目录下去找，仅在非空时有效。 也就是在用户目录之下使用PHP文件的基本目录名，例如："public\_html"

enable\_dl = Off

####是否允许使用dl()函数。dl()函数仅在将PHP作为apache模块安装时才有效。 禁用dl()函数主要是出于安全考虑，因为它可以绕过open\_ｂａｓｅdir指令的限制。 在安全模式下始终禁用dl()函数，而不管此处如何设置。

file\_uploads = On 是否开启上传功能

upload\_max\_filesize = 2M #最大可上传文件，2M

max\_file\_uploads = 20 最大同时可以上传20个文件

allow\_url\_fopen = On #是否允许打开远程文件

allow\_url\_include = Off #是否允许include/require远程文件

default\_socket\_timeout = 60 默认的socket超时时间

[Date] 日期

[filter]

[iconv]

[intl]

[sqlite]

[sqlite3]

[Pcre]

[Pdo]

[Pdo\_mysql]

pdo\_mysql.cache\_size = 2000 ——→ Ped\_mysql的缓存大小

pdo\_mysql.default\_socket= ——→ 默认的socket时间

[Phar]

[Syslog]

define\_syslog\_variables = Off ——→ 是否定义各种的系统日志变量

[mail function] ——→ 邮件功能

SMTP = localhost ——→ 本地作为邮件服务器

smtp\_port = 25 邮件端口号默认是25

mail.add\_x\_header = On ——→ 是否开启最大的header

[SQL]

sql.safe\_mode = Off

####是否使用SQL安全模式。如果打开，指定默认值的数据库连接函数将会使用这些默认值代替支持的参数。对于每个不同数据库的连接函数，其默认值请参考相应的手册页面。

[ODBC]

odbc.allow\_persistent = On ——→ 允许或阻止持久连接.

odbc.check\_persistent = On ——→ 在重用前检查连接是否可用

odbc.max\_persistent = -1 ——→ 持久连接的最大数目，-1意味着没有限制.

odbc.max\_links = -1 ——→ 最大连接数(持久 + 非持久).-1意味着没有限制.

odbc.defaultlrl = 4096 ——→ 长字段处理.返回变量的字节数.0 意味着略过.

odbc.defaultbinmode = 1

####二进制数据处理.0意味着略过,1按照实际返回,2转换到字符.;查看odbc\_binmode和odbc\_longreadlen 的文档来获取针对uodbc.defaultlrl和uodbc.defaultbinmode的解释

[Interbase] ——→ Interbase数据库

ibase.allow\_persistent = 1 ——→ 允许或组织持久连接。

ibase.max\_persistent = -1 ——→ 持久连接的最大数目，-1意味着没有限制.

ibase.max\_links = -1 ——→ 最大连接数(持久 + 非持久).-1意味着没有限制.

ibase.timestampformat = "%Y-%m-%d %H:%M:%S" ——→ 数据库时间记录模式

ibase.dateformat = "%Y-%m-%d"

ibase.timeformat = "%H:%M:%S"

[MySQL]

mysql.allow\_local\_infile = On ——→ 是否允许本地文件连接数据库

mysql.allow\_persistent = On ——→ 允许或禁止 持久连接

mysql.cache\_size = 2000 ——→ mysql缓存大小

mysql.max\_persistent = -1 ——→ 持久连接的最大数目. -1 意味着没有限制.

mysql.max\_links = -1 ——→ 连接的最大数目（持久和非持久）。-1 代表无限制

mysql.default\_port =

####mysql\_connect() 使用的默认端口，如不设置，mysql\_connect()

;将使用变量 $MYSQL\_TCP\_PORT，或在/etc/services 下的mysql-tcp 条目(unix)，

;或在编译是定义的 MYSQL\_PORT(按这样的顺序)

;Win32环境，将仅检查MYSQL\_PORT。

mysql.default\_socket =

####用于本地 MySql 连接的默认的套接字名。为空，使用 MYSQL 内建值

mysql.default\_host = ——→ mysql\_connect() 默认使用的主机（安全模式下无效）

mysql.default\_user = ——→ mysql\_connect() 默认使用的用户名（安全模式下无效）

mysql.default\_password = ——→ mysql\_connect() 默认使用的密码（安全模式下无效

mysql.connect\_timeout = 60 ——→ 连接超时时间，默认是60s

mysql.trace\_mode = Off

[MySQLi]

mysqli.max\_persistent = -1 ——→ 持久连接的最大数目. -1 意味着没有限制.

mysqli.allow\_persistent = On ——→ 允许或拒绝之久连接

mysqli.max\_links = -1 ——→ 最大连接数. -1 意味着没有限制.

mysqli.cache\_size = 2000 ——→ 连接缓存大小

mysqli.default\_port = 3306 ——→ 连接端口号

####mysqli\_connect()默认的端口号.如果没有设置, mysql\_connect() 会使用 $MYSQL\_TCP\_PORT;或者 位于/etc/services的 mysql-tcp 入口或者编译时定义的MYSQL\_PORT 值(按照此顺序查找).;Win32 只会查找MYSQL\_PORT值.

mysqli.default\_socket =

####对于本地MySQL连接的默认socket名称. 如果为空, 则使用MySQL内建默认值.

mysqli.default\_host =

####mysqli\_connect()的默认host值(在安全模式中不会生效)

mysqli.default\_user =

####mysqli\_connect()的默认user值(在安全模式中不会生效).

mysqli.default\_pw =

####mysqli\_connect() 的默认password值(在安全模式中不会生效).

; 注意在此文件中保存密码一般来说是 \*糟糕\* 的主义.

; \*任何\* 使用PHP的用户可以执行 'echo get\_cfg\_var("mysqli.default\_password")

; 并且获取到此密码! 而且理所当然, 任何有对此文件读权限的用户都可以获取到此密码.

mysqli.reconnect = Off ——→ 允许或阻止持久连接

[mysqlnd]

mysqlnd.collect\_statistics = On

mysqlnd.collect\_memory\_statistics = Off

[OCI8]

[PostgreSQL]

pgsql.allow\_persistent = On ——→ 允许或阻止持久连接.

pgsql.auto\_reset\_persistent = Off

####总是在 pg\_pconnect() 时检测断开的持久连接.;自动重置特性会引起一点开销.

pgsql.max\_persistent = -1 ——→ 持久连接的最大数目. -1 意味着没有限制.

pgsql.max\_links = -1 ——→ 最大连接数 (持久 + 非持久). -1 意味着没有限制

pgsql.ignore\_notice = 0 ——→ 是否忽略 PostgreSQL 后端通告消息.;通告消息记录会需要一点开销.

pgsql.log\_notice = 0

####是否记录 PostgreSQL 后端通告消息.;除非 pgsql.ignore\_notice=0, 否则模块无法记录通告消息。

[Sybase-CT]

sybct.allow\_persistent = On ——→ 允许或阻止持久连接.

sybct.max\_persistent = -1 ——→ 持久连接的最大数目. -1 意味着没有限制.

sybct.max\_links = -1 ——→ 最大连接数 (持久 + 非持久). -1 意味着没有限制.

sybct.min\_server\_severity = 10 ——→ 显示出的错误最小严重程度.

sybct.min\_client\_severity = 10 ——→ 显示出的消息最小严重程度

[bcmath]

bcmath.scale = 0 ——→ #用于所有bcmath函数的10十进制数数字的个数

[browscap]

[Session]

session.save\_handler = files ——→ 用于保存/取回数据的控制方式

session.use\_cookies = 1 ——→ 是否使用cookies

session.use\_only\_cookies = 1

####这个选项允许管理员去保护那些在URL中传送session id的用户免于被攻击;默认是0.

session.name = PHPSESSID ——→ session 的名字（同时作为cookie的名称

session.auto\_start = 0 ——→ 在请求开始时初始化 session

session.cookie\_lifetime = 0 ——→ cookie的存活秒数，如果为0，则是直到浏览器重新启动

session.cookie\_path = / ——→ cookie的有效路径

session.cookie\_domain = ——→ cookie的有效域名

session.cookie\_httponly =

####是否将httpOnly标志增加到cookie上,

增加后则cookie无法被浏览器的脚本语言(例如JavaScript)存取.

session.serialize\_handler = php 用于序列化数据的处理器. php是标准的PHP序列化器.

session.gc\_probability = 1

####; 定义'垃圾回收'进程在每次session初始化时开始的比例.

; 比例由 gc\_probability/gc\_divisor来得出,

; 例如. 1/100 意味着在每次请求时有1%的机会启动'垃圾回收'进程.

session.gc\_divisor = 1000

session.gc\_maxlifetime = 1440

####在这里数字所指的秒数后，保存的数据将被视为'碎片(garbage)'并由gc进程清理掉。

session.bug\_compat\_42 = Off

####PHP 4.2 和更早版本有一个未公开的 特性/bug , 此特性允许你在全局初始化一个session变量,即便 register\_globals已经被关闭.;如果此特性被使用,PHP 4.3 和更早版本会警告你.;你可以关闭此特性并且隔离此警告. 这时候,如果打开bug\_compat\_42,那此警告只是被显示出来.

session.bug\_compat\_warn = Off

session.referer\_check =

####检查HTTP Referer来防止带有id的外部URL.;HTTP\_REFERER 必须包含从session来的这个字段才会被认为是合法的.

session.entropy\_length = 0 ——→ 从此文件读取多少字节

session.cache\_limiter = nocache

####设置为{nocache,private,public,}来决定HTTP缓冲的类型;留空则防止发送anti-caching头.

session.cache\_expire = 180 ——→ 文档在n分钟之后过期.

session.use\_trans\_sid = 0

####trans sid 支持默认关闭.

;使用 trans sid 可能让你的用户承担安全风险.;使用此项必须小心.; - 用户也许通过email/irc/其他途径发送包含有效的session ID的URL给其他人.; - 包含有效session ID的URL可能被存放在容易被公共存取的电脑上.; - 用户可能通过在浏览器历史记录或者收藏夹里面的包含相同的session ID的URL来访问你的站点.

session.hash\_function = 0 ——→ 选择hash方法;0:MD5(128 bits);1:SHA-1(160 bits)

session.hash\_bits\_per\_character = 5

; 当转换二进制hash数据到可读形式时,每个字符保存时有几位.

; 4 bits: 0-9, a-f; 5 bits: 0-9, a-v; 6 bits: 0-9, a-z, A-Z, "-", ","

url\_rewriter.tags = "a=href,area=href,frame=src,input=src,form=fakeentry"

URL rewriter会在已经定义的一组HTML标签内查找URL.; form/fieldset 是特殊字符; 如果你在这里包含他们, rewriter会增加一个包含信息的隐藏<input>字段否则就是在URL中附加信息.; 如果你你想遵守XHTML, 删除form的入口.; 注意 所有合法的入口都需要一个"="符号, 甚至是没有任何值的.

[MSSQL]

mssql.allow\_persistent = On ——→ 允许或阻止持久连接

mssql.max\_persistent = -1 ——→ 持久连接的最大数目. -1 意味着没有限制.

mssql.max\_links = -1 ——→ 最大连接数 (持久 + 非持久). -1 意味着没有限制.

mssql.min\_error\_severity = 10 ——→ 显示出的错误最小严重程度.

mssql.min\_message\_severity = 10 ——→ 显示出的消息最小严重程度

mssql.compatability\_mode = Off ——→ PHP 3.0 老版本的兼容模式.

mssql.secure\_connection = Off ——→ 当连接到服务器时使用NT验证

[Assertion]

[COM]

[mbstring]

[gd]

[exif]

[Tidy]

当调用tidy时,默认指向tidy配置文件的路径 tidy是否自动清除和修复输出?; 警告: 不要在你产生非html内容时使用此项,例如产生动态图片时

tidy.clean\_output = Off 清除功能是否开启，本文中为关闭状态

[soap]

soap.wsdl\_cache\_enabled=1 ——→ 打开或关闭WSDL缓冲特性.

soap.wsdl\_cache\_dir="/tmp" ——→ 设置SOAP扩展存放缓冲文件的目录

soap.wsdl\_cache\_ttl=86400——→(存活时间)设置当缓冲文件被用来替换原有缓冲文件的秒数

soap.wsdl\_cache\_limit = 5 ——→ 最小缓存

[sysvshm]

[ldap]

ldap.max\_links = -1

[mcrypt]

[dba]

[xsl]

来源：清源教育