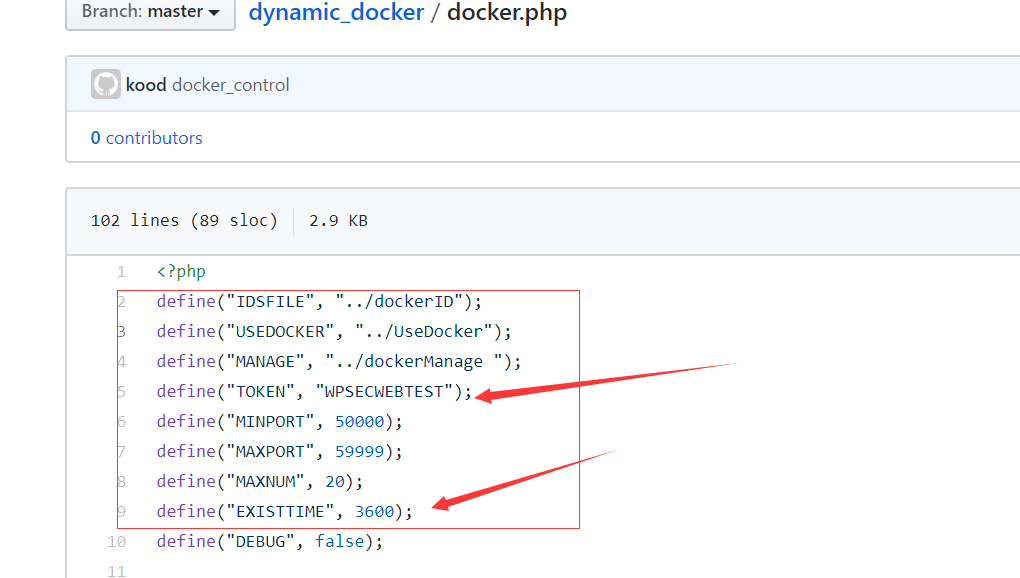
如果想实现使用docker去下发容器以达到每人一套互相独立的环境的话，可以尝试使用以下方法。

下载 docker 支持源码：<https://github.com/onlykood/dynamic_docker>

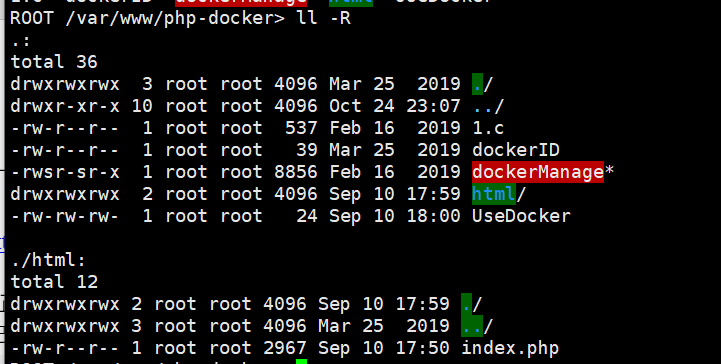
1. 登录后台，在配置界面正确配置好以下三个要素（存活时间与密码令牌设置完成之后需要与docker.php中的配置同步）



上面需要与下面的红色箭头部分保持一致（下面的dockerID、UseDocker保存配置信息，一定要放在外网无法访问到的地方，防止信息泄露！dockerManage是由1.c 编译出来的直接 gcc 1.c –o dockerManage 即可，所有命名随意，但是需要保持一致性）



1. Docker.php中的大致方法就是创建或销毁一个docker，通过get方法传递参数，同时校验token是否一致，如果不一致则直接退出，防止接口恶意使用。
2. 权限设置，dockerID文件需要设置为644权限、UserDocker文件需要设置为666权限，主要是让www-data用户可以去读写，dockerManage则需要设置为s权限，因为调用了docker语句，只能使用root去执行，设置s权限的目的是让www-data可以通过此程序来调用docker以及at的一些方法。效果图如下所示（1.c可以删除）：



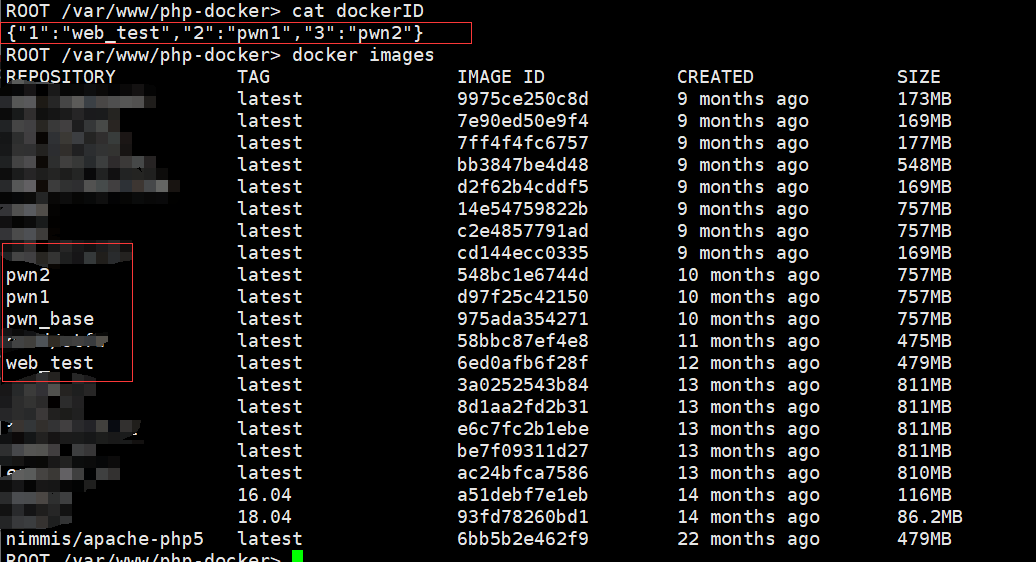
另外需要注意的点有：

1. 安装好docker
2. at命令默认是拒绝www-data用户执行的，需要在/etc/at.deny 中将www-data用户去掉。

由此，docker的快速运行环境就配置完成了。

Part2：下发docker

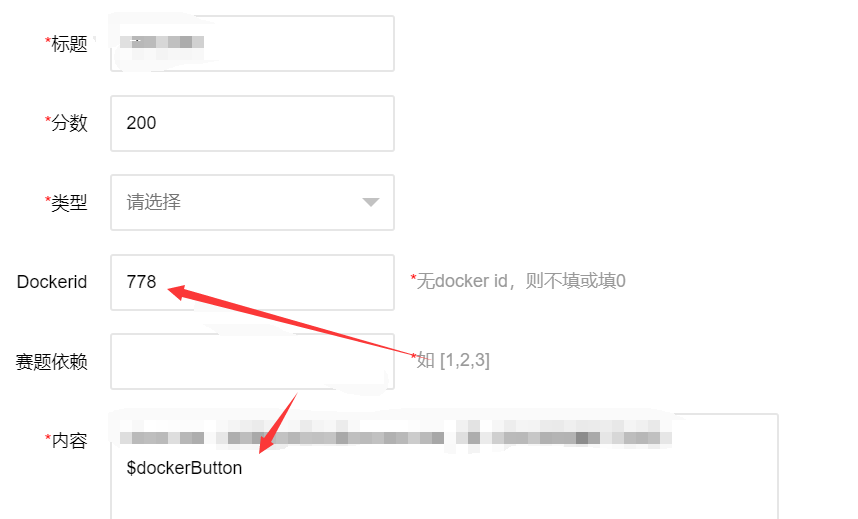
1. Docker的下发需要先build好docker的镜像，然后将镜像名称按照json的格式写入dockerID中。例如

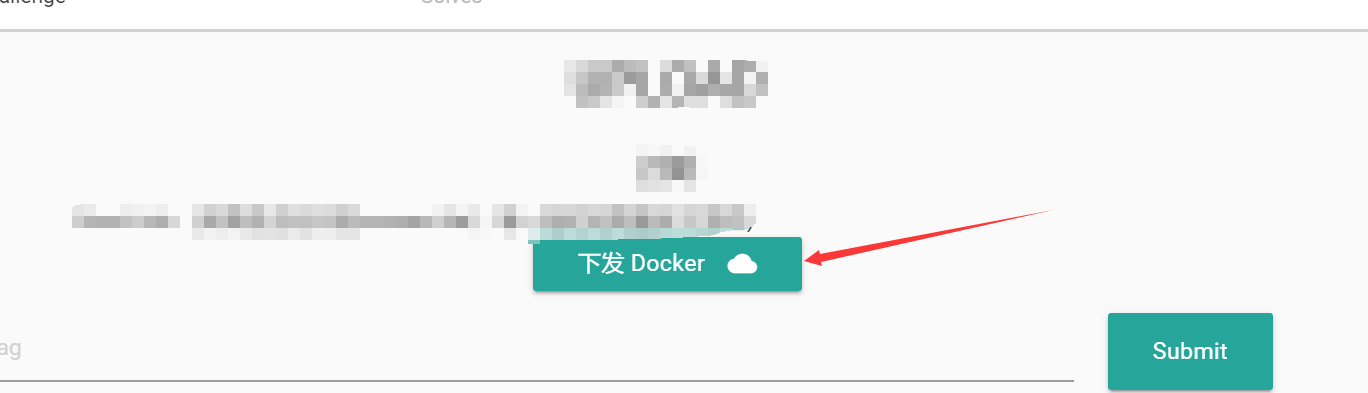


前面的是数字id，这个可以随意设置，只需要在这个dockerID文件中不出现相同id即可。

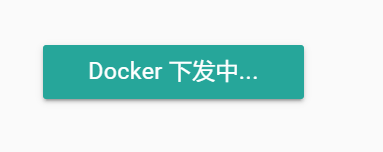
1. 平台配置：

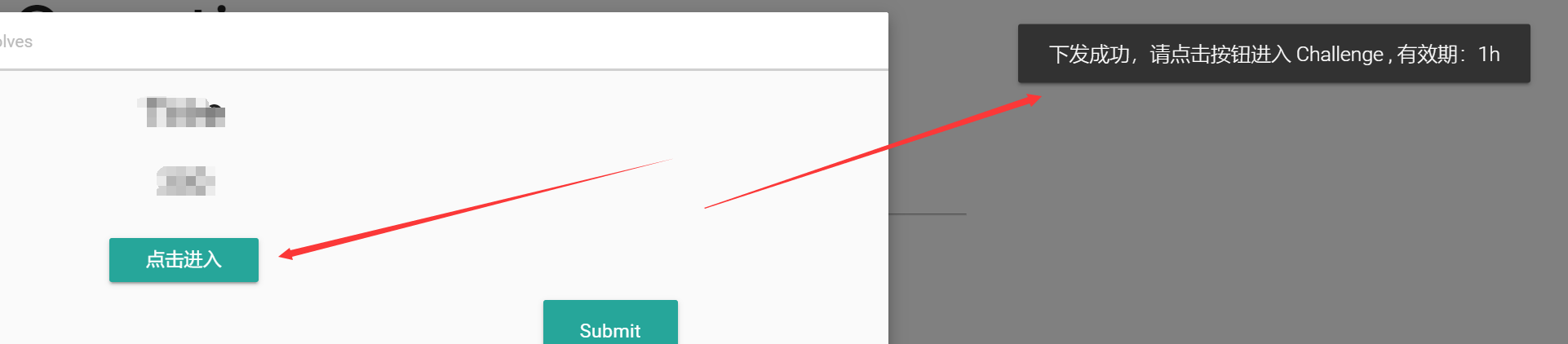
在后台的题目中，需要在Dockerid中填入上述在dockerID文件夹中的id，下面是$dockerButton方法，在前台会将该字符串替换为docker下发按钮。





如果都配置正确的话，点击下发按钮时，就会构造一个请求去访问docker.php，php在下发docker后会返回ip与port（默认是将类型为web题的docker中的80端口随机映射到50000-59999中的任意端口，而pwn题的话则是8888端口的随机映射，对于其他类型没有设置）





点击进入即可。

如果docker内部的端口不是80端口的话，需要比较复杂的修改，需要在后台赛题部分额外增加docker端口输入框，并且修改数据库语句，或者修改functions.php中的getDockerUrl 函数等等，目前由于时间精力的原因，没有修改的计划。