

PHP демоны на примере социальных игр

Календарёв Александр
РБК Медиа



<http://www.devconf.ru>

Содержание

- Демоны изнутри
- Варианты Архитектур
- Прочие плюшки





Демон. Терминология

- Демон (англ. daemon) — в системах класса UNIX — служба, работающая в фоновом режиме без прямого общения с пользователем.
- Термин был придуман в Массачусетском технологическом институте
- Символизирует персонаж мысленного эксперимента, демону Максвелла, занимающегося сортировкой молекул в фоновом режиме.

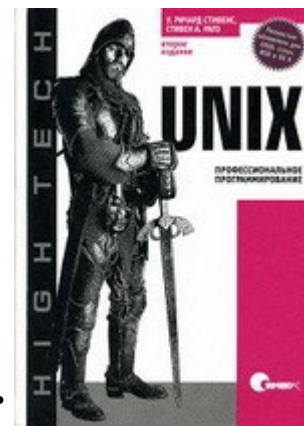


Демоны. Использование.

- Расчетные и вычислительные задачи
- Обработка медиа контента (фото, видео)
- Работа с базой (статистика, перемещение данных)
- Хранение информации
- Взаимодействие с WEB-клиентом
- On-line Общение (чат, мессенжеры)

Демоны. Требования

- Сброс маски создания файлов в 0: `umask()`
- Вызвать `fork()` и завершить родительский процесс.
- Создать новую сессию: `setsid()`.
- Сделать корневой каталог текущим каталогом: `chroot('/')`
- Закрывать все файловые дескрипторы: `close()`
- Перенаправить дескрипторы 0,1 и 2 в `/dev/null`
- Записать `pid` в `/var/run/mydaemon.pid`
- Обработать `SIGTERM` и `SIGHUP`



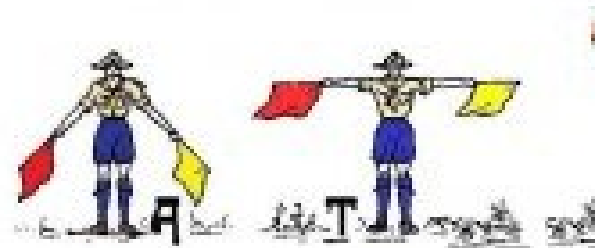
Демоны. Реализация

- PCNTL
- POSIX
- Libevent, Libevent2

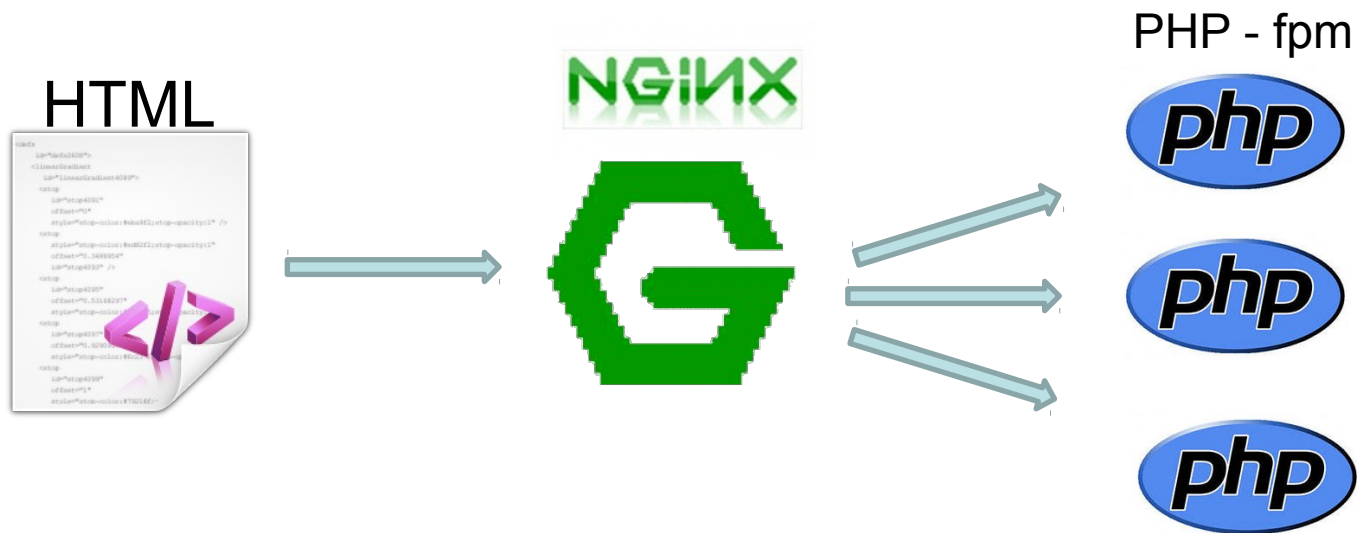


Обработка сигналов

- **SIGTERM**
 - закрытие всех файловых дескрипторов
 - окончание всех дочерних процессов
 - мягкое завершение процесса
- **SIGHUP**
 - перечитываем конфигурацию
 - закрытие всех соединений
 - переоткрываем log файл
 - перезапускаем сервер



Классическое WEB App



Социальная игра



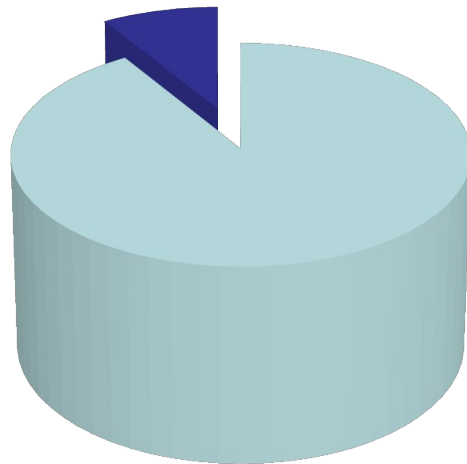
Преимущество использования



- Постоянное соединение с клиентом
- Одноразовая инициализация справочных данных
- Одноразовое соединение с БД, кешем и т.д.
- Возможность мгновенного общения между разными клиентами

Демоны. Преимущество

Распределение трафика

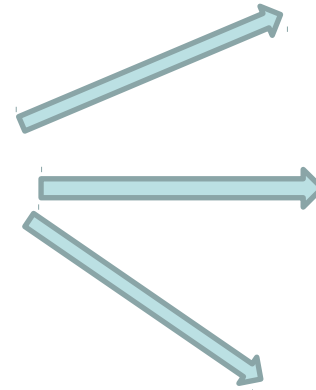


■ Socket

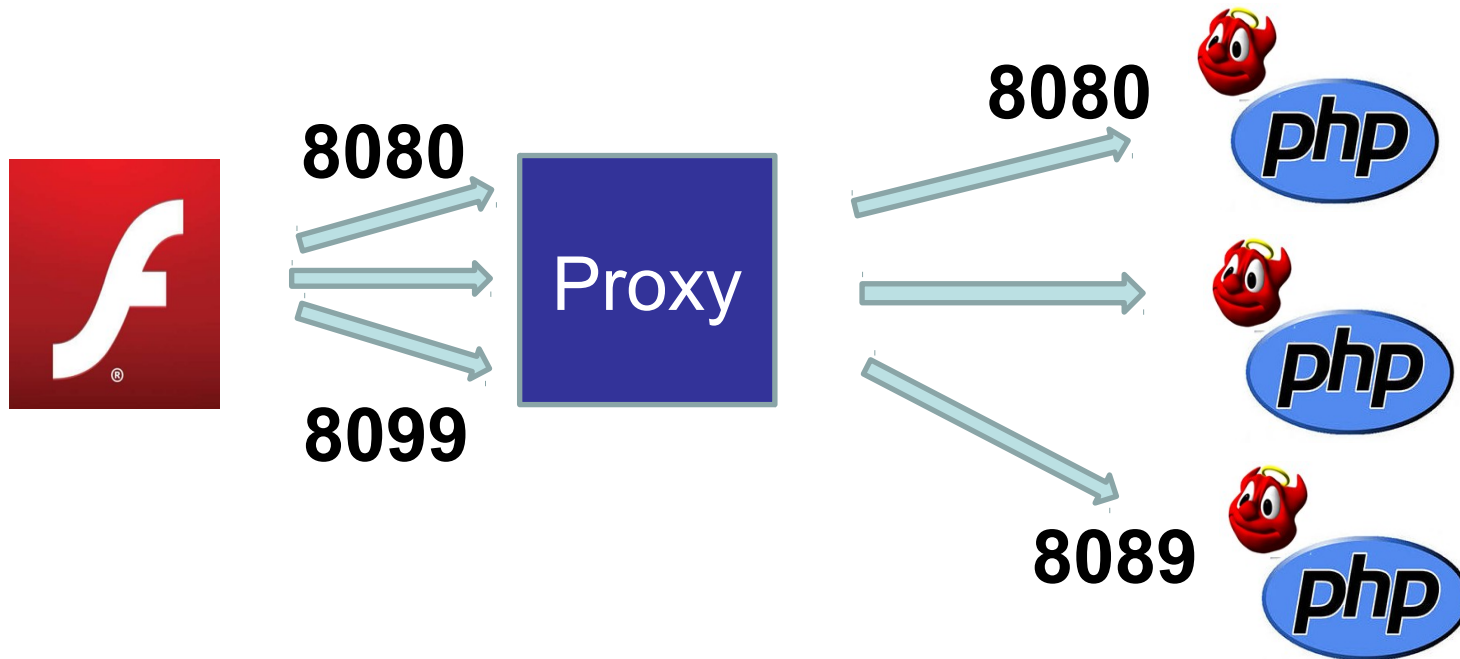
Найди 10 отличий



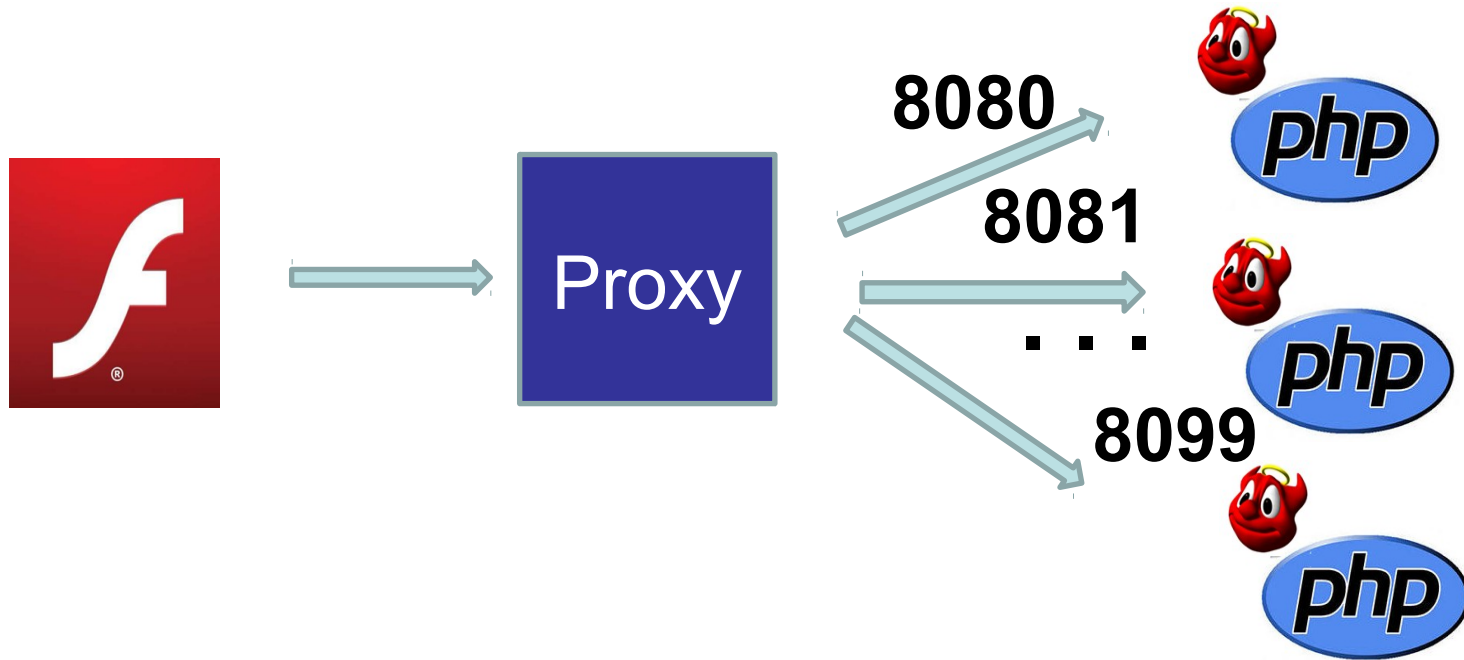
Демоны. Как хотелось бы



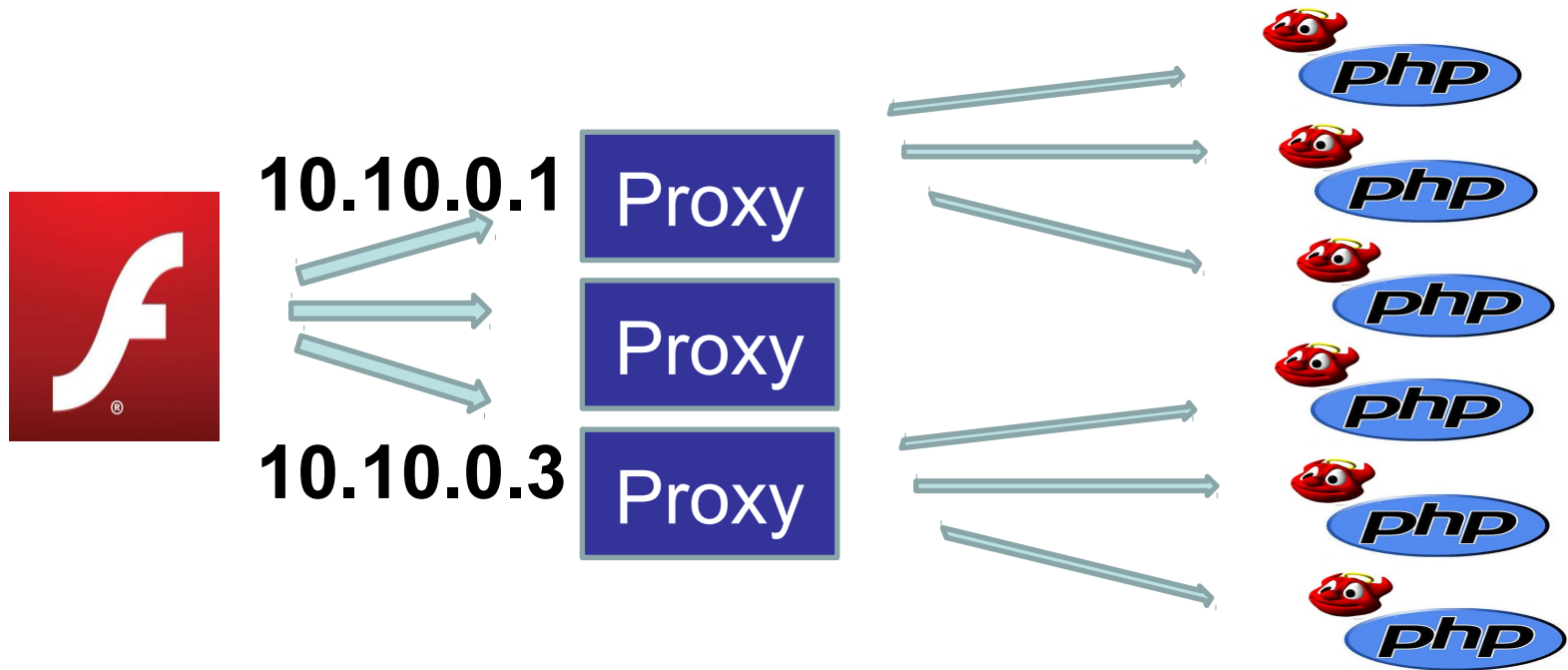
Демоны. Инициатива клиента



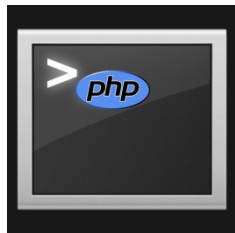
Демоны. Инициатива Proxy



Демоны. Масштабирование



Демоны. Управление



fork()

Master



Worker



Мастер процесс



- Следит за worker
- Перезапускает при необходимости
- Собирает статистику
- Отдает информацию на мониторинг



Как не надо делать

- `exec("php mydaemon.php > error.log 2>1 &");`
`exec("nohup php mydaemon.php >> error.log &");`
- PHP Worker не явл. прямым потомком master process
- Не возможно обработать SIGCHLD

Демоны. Запуск



- Осуществляем через rcd скрипт `/etc/init.d/mydaemon`

Демоны. Мониторинг

- Жизнедеятельность (память, pid,)
- Производительность (rps, uptime)
- Функциональность (длину очереди, кол-во online)



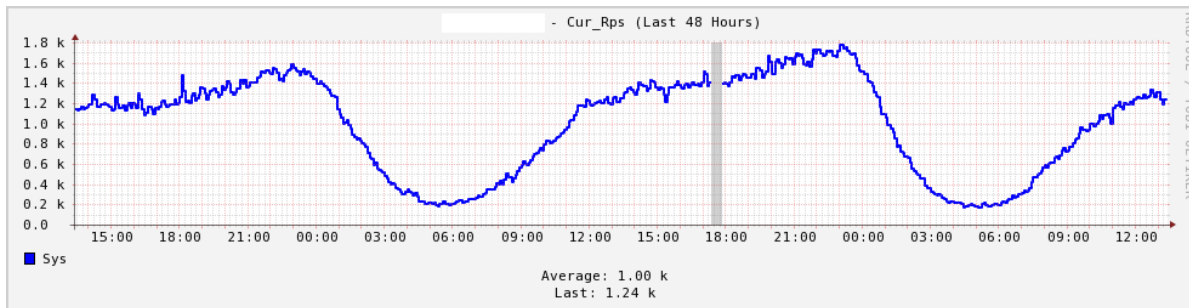
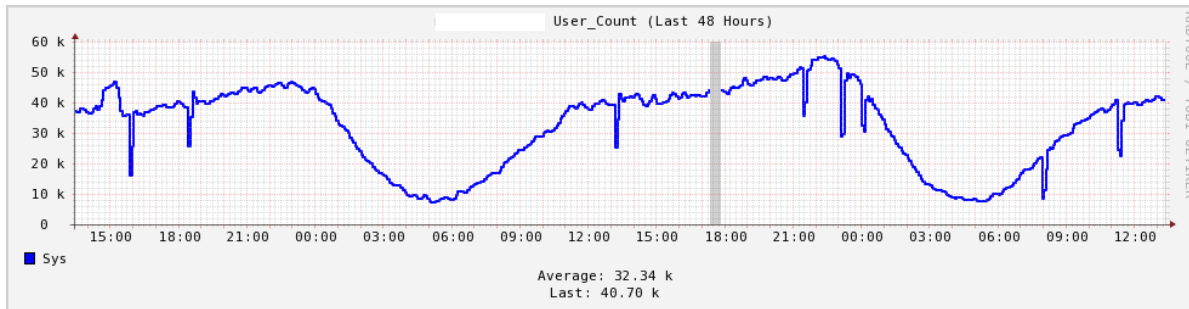
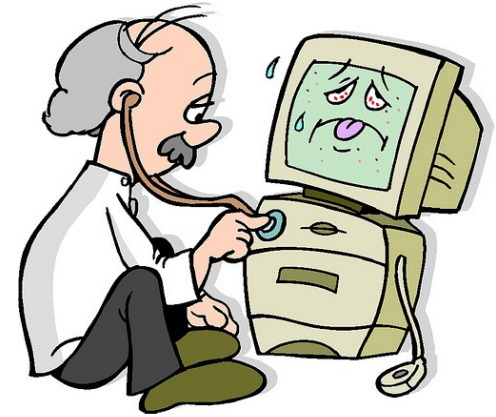
Демоны. Мониторинг

➤ stats
STAT rev 20513
STAT pid 16299
STAT uptime 688647
STAT curent_connections 264
STAT total_connections 22877
STAT total_memory 8308748
STAT data_memory 8240892
STAT current_rps 1210.00
STAT users_count 40583



Используем memcached протокол

Демоны. Мониторинг



Демоны. Тестирование

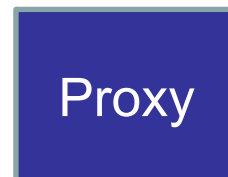


- Функциональное
- Тестирование АПИ

Эмулятор Flash



Обмен



Текстовые протоколы

- URI
- JSON
- XML

Бинарные протоколы

- AMF
- Protobuf

Текстовые протоколы:

- *Человеко-понятные*
- *Проще отладка*

Бинарные протоколы:

- *Более компактная передача данных*
- *Более защищенные*

Google Protobuf



- Бинарный протокол от «корпорации добра»
- Поддержка всех основных языков программирования
- Файл .proto описания структуры данных
- Компилятор трансформации .proto в нативные структуры и классы

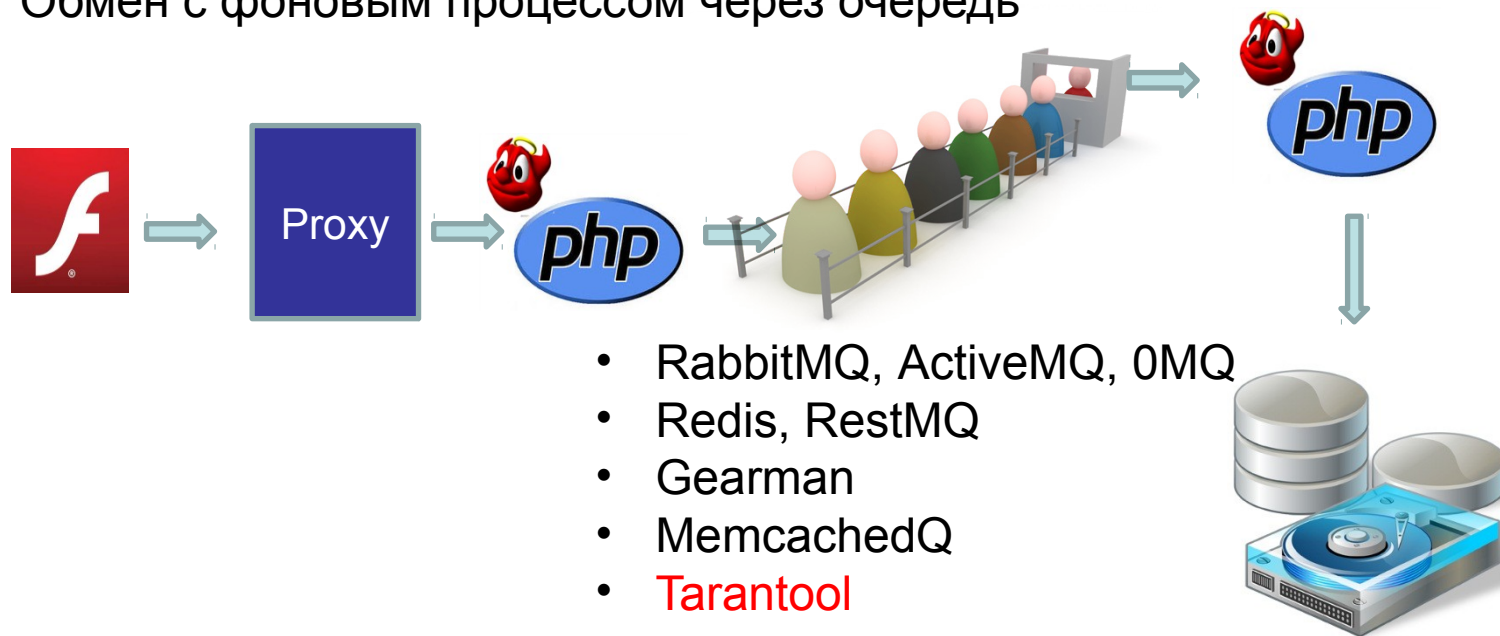
Демоны. Ускорение и перестройка

- Кеширование
- Предварительная инициализация и финализация
- Вынос обмена с БД в бэкграунд



Вынос обмена с БД в фоновый процесс

Обмен с фоновым процессом через очередь



- RabbitMQ, ActiveMQ, 0MQ
- Redis, RestMQ
- Gearman
- MemcachedQ
- **Tarantool**

Вопросы



PHP -демоны в социальных играх



Александр Календарёв



akalend@mail.ru



<http://highloadblog.com>



[@mail.ru](https://twitter.com/akalend)



<http://github.com/akalend>



<http://akalend.habrahabr.ru>