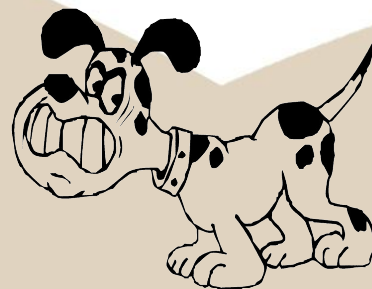
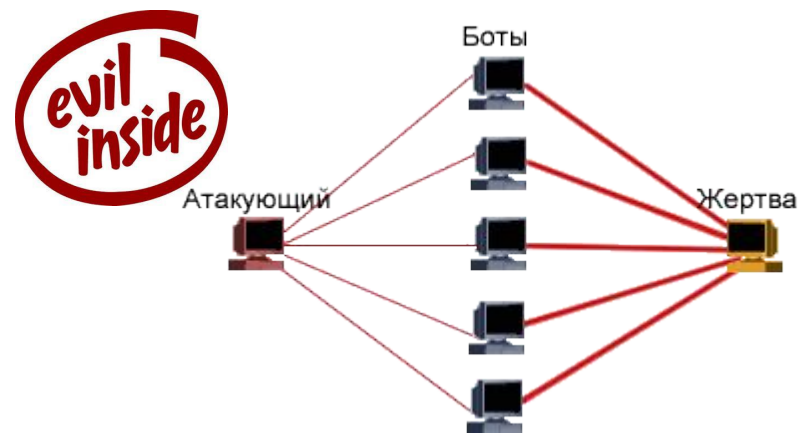


# Удобная кросс-доменная авторизация и персонализация для DDoS-устойчивого сайта



Redis + Varnish + Javascript

# DDOS



# DDOS-устойчивый сайт



Как защититься?

- Приходит 10.000 HTTP-запросов/сек с разных IP...
- Что делать?
- Идеи?



ИЛИ



?

# DDOS-устойчивый сайт



## Варианты защиты

### 1. Вычислять и банить IP ботов

– Способы есть, но это сложно:

- Боты маскируются.
- Оператор – не дурак.
- Ботнет большой.



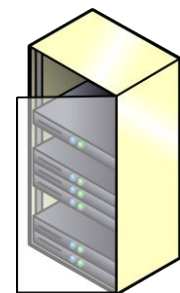
# DDOS-устойчивый сайт



## Варианты защиты

### 2. Настоящий Хайлоад

- Большие расходы: железо, программирование...
- DDOS закончится и все это будет не надо.



# DDOS-устойчивый сайт



## Варианты защиты

3. Не пускать анонимов в движок
  - Обслуживать их из быстрого кеша
  - Медленный сервис? Captcha!



# Что это означает для движка?



Типичный проект



посетитель



NGINX



Not  
Only SQL



Как внедрить кеширование для анонимов ?

# Первый подход. Классика жанра.

## Движок с кешированием



запрос

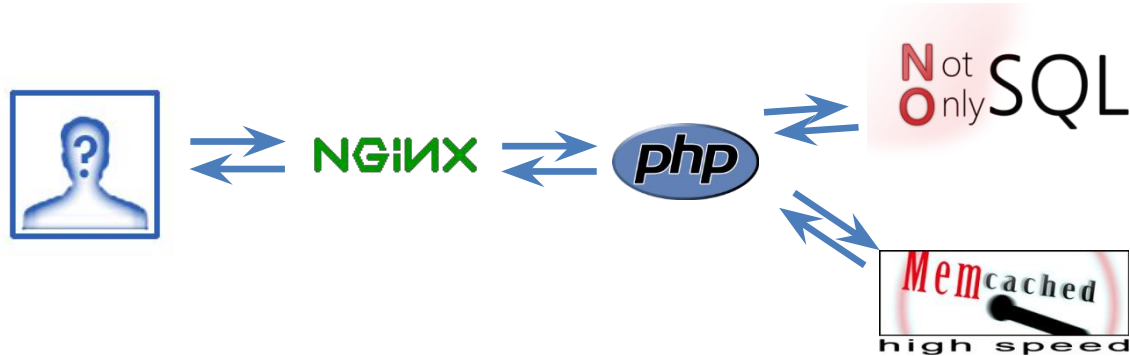
страница

Not Only SQL





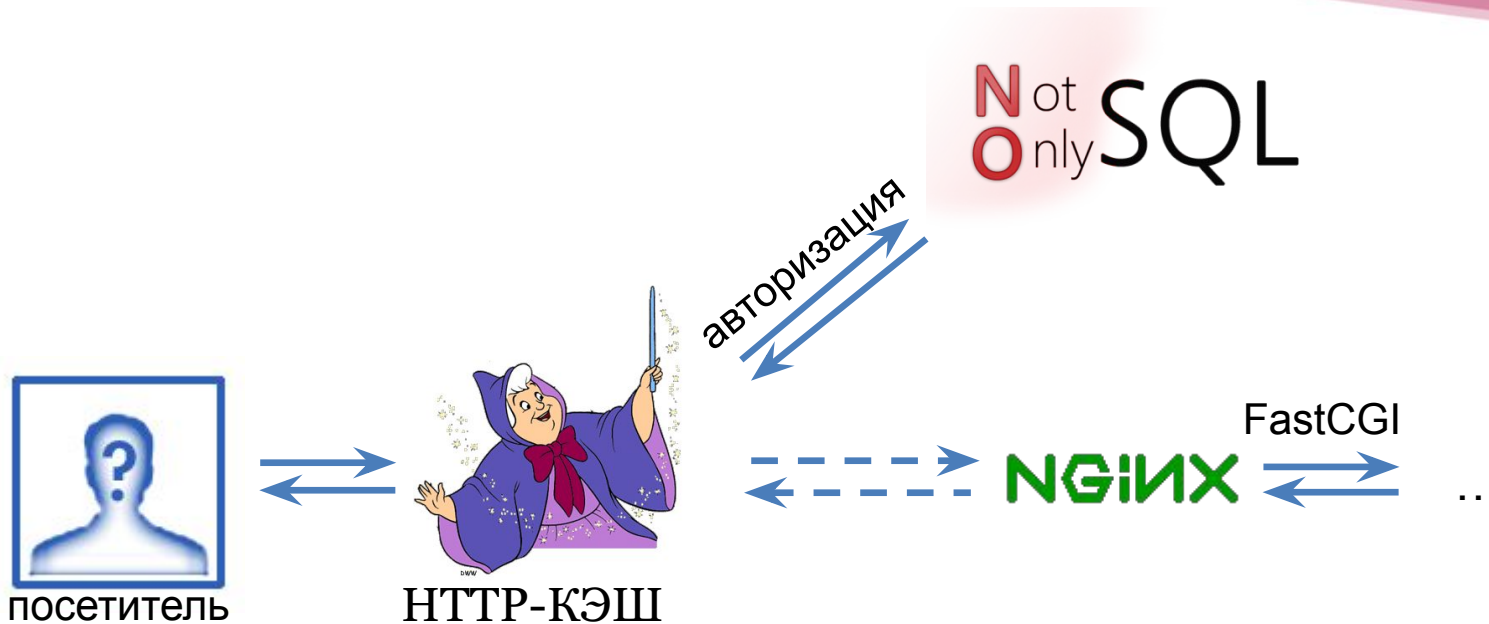
# Короче



- Структура
  - Много компонент
  - Много лишней работы по копированию данных
- Производительность
  - Средняя, а нужна максимальная.

# Версия 2.0

## "Кэш – наше всё"



P.S.

Также пробовали подписанные куки разные для анонима и зарегистрированного

```
COOKIE .= md5(secret_anon,cookie).substr(0,4)
```

```
COOKIE .= md5(secret_regged,cookie).substr(0,4)
```

# Redis

- База а-ля Memcache
- Хранит все в памяти
  - Сохраняет периодически или по запросу
- GET SET EXPIRE
- Умеет структуры данных
  - HASH, (Sorted) SET, LIST



- это БЫСТРО!

# Varnish

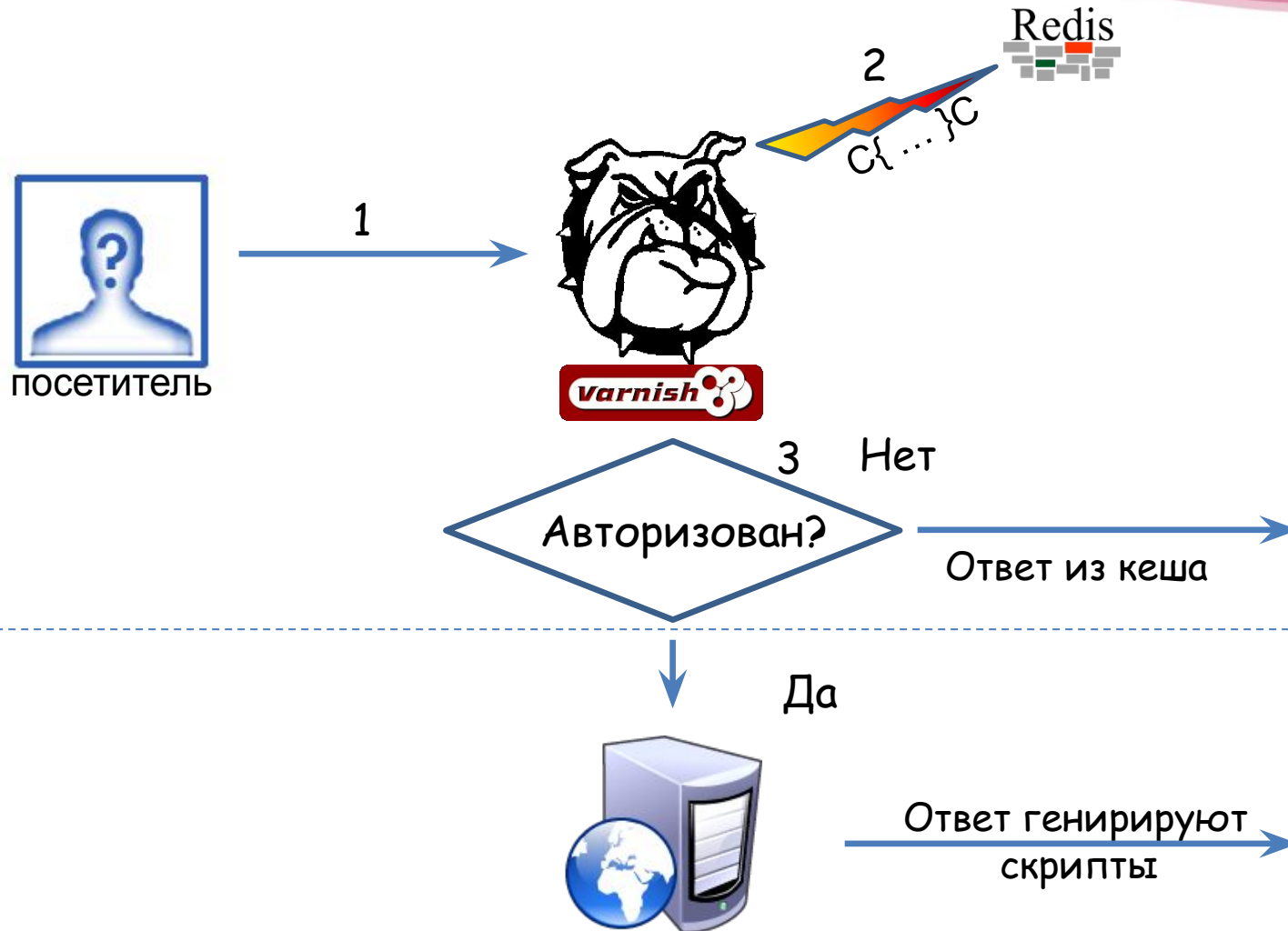
Кеширующий прокси / сервер / балансер  
/...

Хуки на всех стадиях обработки запроса

<http://www.varnish-cache.org/trac/wiki/VarnishFeatures>

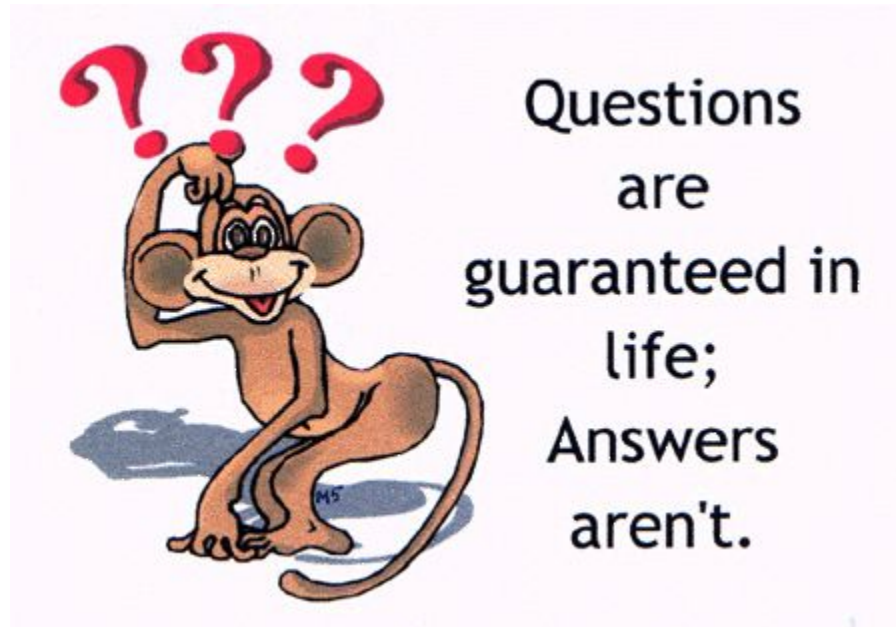


# Что получилось



==cut

- Вопросы?

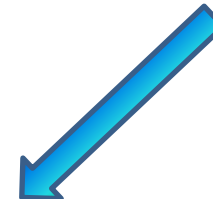
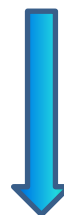


# Персонализируемый сайт

Онлайн-сервисы

Просмотренные товары

Геотаргетинг



Персонализация «рулит»

Она нужна всем, включая анонимных посетителей

=> Авторизуем всех!

# Кэш VS персонализация

## Как объединить?

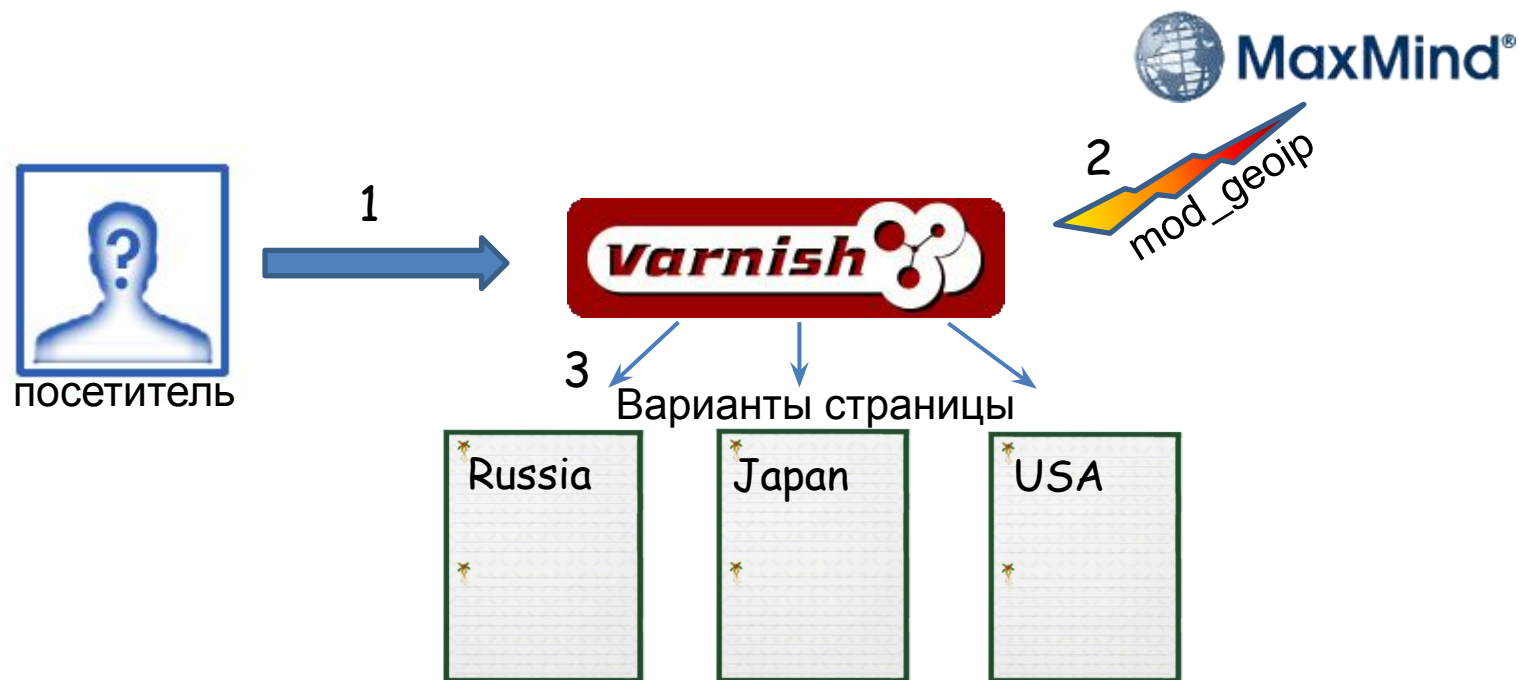
- Кэш
  - страница генерируется 1 раз
- Персонализация
  - страница подстраивается под посетителя
- Идеи?





# Кэш + персонализация

1. Персонализация влияет комплексно.
  - Геотаргетинг
    - телефоны, цены, информация...
  - Каждый геотаргетинг - своя страница в кеше



# Кэш + персонализация

## 2. Персонализация влияет точно.

### – Блочные сервисы

- Последние просмотры
- Реклама
- ...

```
<html>
  <div class="sidebar">
    ...
    <esi src="/lastread.php"/>
    ...
  </div>
</html>
```

Подзапрос



# Очистка кэша при изменениях

- К записи в кэше прикреплены **тэги**.
- Тэги задаются при генерации страницы.
- По тэгам можно удалять.

# Очистка кеша при изменениях

## Пример:

- Страница фото

- <http://www.photosight.ru/photos/3933415/>
- Тэг: foto\_123
  - Фото обновляется
    - » При изменении файла или описания фото
    - » При изменении кол-ва комментариев

# Очистка кеша при изменениях

## Пример:

– Страница галереи / тэга

- <http://www.photosight.ru/photos/category/7/>
- Тэги: foto\_1, foto\_2, ..., foto\_20

# Кэш + сложные зависимости

## ... или когда добавить таги лень

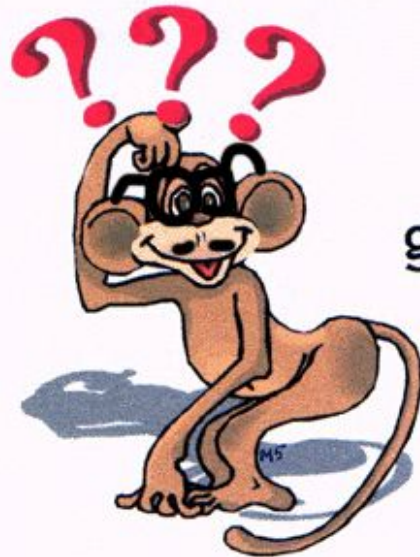
- Рецепты
  1. Ограничить время жизни кеша
  2. Убивать все
    - Весь тэг `article` при изменении дерева статей
      - Или все URL вида `article/*`

Главное – HIT/MISS

# Дополнительно

- Кешируем и для зарегистрированных
  - SID в Hash
- Поддержка 304 в браузере
- Полезные ссылки
  - <http://www.varnish-cache.org/docs/2.1/>
  - <http://www.slideshare.net/tgr1/varnish-plnog-4>
  - <http://www.slideshare.net/crucially/varnish-velocity-ignite>
  - <http://kristianlyng.../smart-bans-with-varnish/>

==cut



Questions  
are  
guaranteed in  
life;  
Answers  
aren't.



# Мульти-доменный сайт

- Один сайт – много доменов 2 уровня
  - <http://site.ru>
  - <http://notebook-site.ru>
  - <http://mouse-site.ru>
  - ...
- Нахрена?
  - SEO !

# Кросс-доменная авторизация

Вошел на один сайт

– <http://site.ru>

... Авторизован на всех

– <http://notebook-site.ru>

– <http://mouse-site.ru>

Идеи?



# Кросс-доменная авторизация

- Вход, выход, авторизация – на мастере
  - master.com
- Задача - синхронизировать Cookie между доменами
- Как?
  - См. следующий слайд

# Персонализирующий скрипт

Анонимная страница

a.com



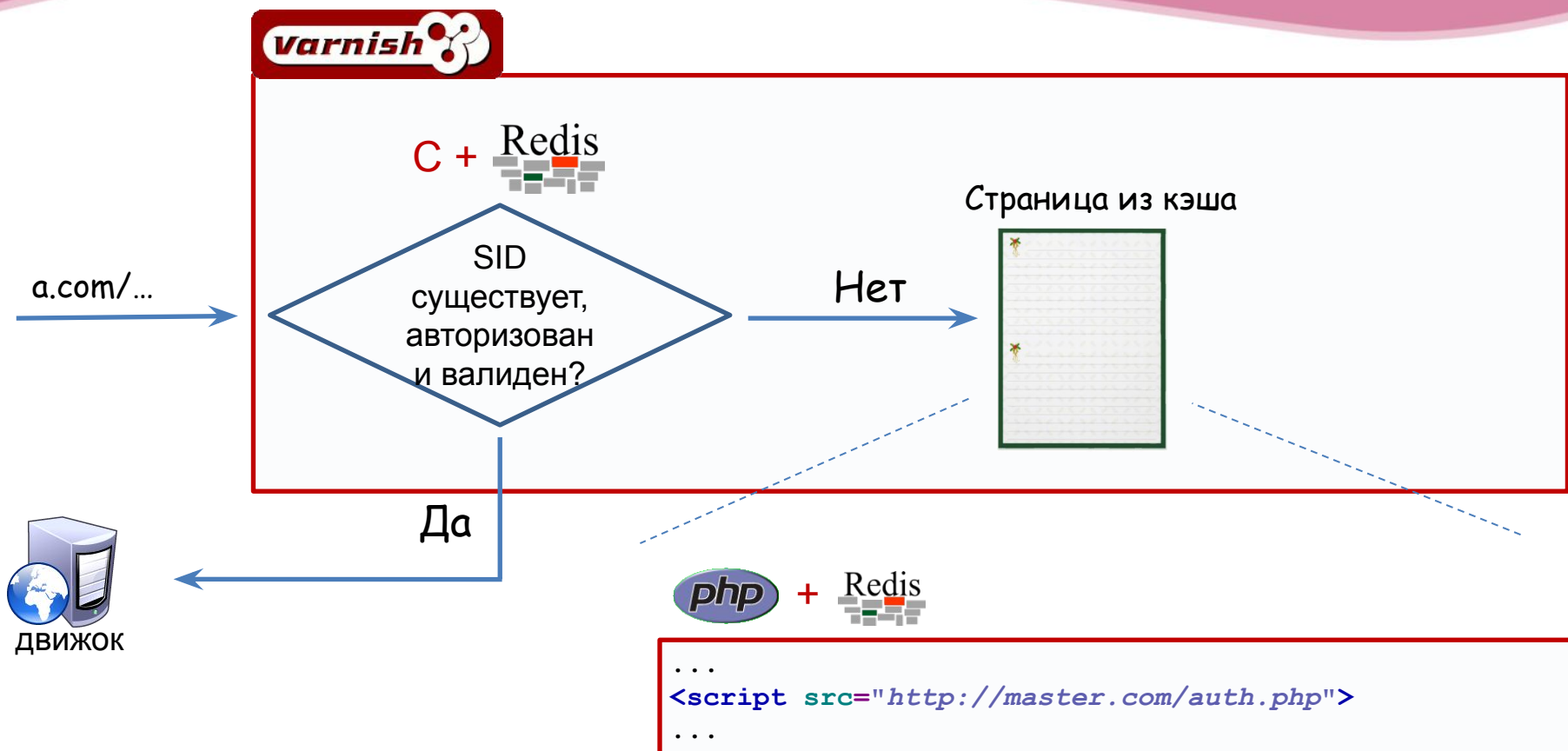
<http://master.com/auth.php>

1. Загрузить/создать сессию
  - По кукам `master.com`
2. Если зарегистрированный:
  - Поставить куку на `b.com=>reload`
3. Если аноним:
  - Персонализация при помощи JS

```
<html>
  <head>
    <script src="http://master.com/auth.php">
      ...
    </head>
  ...
```

- **Cookie анонима стоят только на `master.com`**

# Производительность



# Pitfall

- 3<sup>rd</sup> party cookie!
  - Политика безопасности РЗР
  - Safari запрещает по умолчанию
    - Другие браузеры можно настроить
  - [Демо](#)
- *Как обойти?*

# Удобная авторизация

- Динамическая форма с любой страницы
  - Кросс-доменная коммуникация
    - `<script>` - логин-пароль нельзя передавать GET
    - `window.name` + вспомогательный `iframe`
- Действия для авторизованного посетителя
  - `Auth.decorate(callback)`
- Автопривязка после регистрации
  - Комментарии
- Все вместе
  - См. демо <http://master.com>

# Дополнительная защита

1. **Время жизни однократных посетителей**
  - Если в течение минуты не было захода – удалять
  - Защищает от ботов без кук
2. **Против ботов с поддержкой Cookie**
  - Как правило, таких ботов меньше
  - Не хранить анонимов с IE6
    - Определение по browser features → запись в Cookie
  - Персонализация не везде
  - Captcha...