

Метод отслеживания посетителей сайтов при помощи CSS, без JavaScript

Jan Böhmer [опубликовал](#) прототип системы для организации отслеживания перемещения посетителей по сайтам, который ограничивается использованием только CSS и не требует выполнения кода на JavaScript. При размещении на сайте представленного кода организуется передача на внешний сервер базовой информации о посетителе, включая сведения о разрешении экрана, типе браузера и наличии заданных шрифтов. На основе различий в доступных шрифтах можно определить тип ОС.

Кроме того, определяется по каким ссылкам на странице переходил пользователь или к которым из ссылок подводил курсор мыши. Также могут быть выявлены общие особенности перемещения мыши по экрану (фоном выводится невидимая таблица полей, по которой определяется перемещение мыши). В текущем виде код может отслеживать только первые клики и наведения мыши, последующие клики по тем же ссылкам или повторные наведения мыши не учитываются. Проверить работу метода можно на специально подготовленной [демонстрационной странице](#).

Реализация использует возможности CSS по добавлению изображений из внешних источников при помощи свойства `url("foo.bar")` - так как ресурсы загружаются только при необходимости, можно указать вместо изображения ссылку на внешний скрипт-сборщик статистики и привязать к таким событиям, как переход на ссылку или подведение мыши к ссылке. Например для отслеживания кликов можно использовать код:

```
#link2:active::after {
  content: url("track.php?action=link2_clicked");
}
```

Для определения идентификатора браузера можно использовать правила [@supports](#) для проверки наличия специфичных для каждого типа браузеров свойств CSS, например, только браузеры на движке Chromium поддерживают свойство `"-webkit-appearance"`:

```
@supports (-webkit-appearance:none) {
  #chrome_detect::after {
    content: url("track.php?action=browser_chrome");
  }
}
```

Для определения наличия шрифтов в CSS создается новое фиктивное семейство шрифтов, в качестве источника для загрузки которого указывается внешний скрипт-сборщик данных.

Далее формируются проверочные блоки текста в котором первым указывается проверяемый шрифт, а вторым фиктивный шрифт. Если проверяемый шрифт присутствует, то второй шрифт будет игнорирован, но если проверяемого шрифта нет, браузер попытается использовать фиктивный шрифт как запасной вариант и отправит запрос к внешнему скрипту:

```
@font-face {  
    font-family: Font1;  
    src: url("track.php?action=font1");  
}  
#font_detection1 {  
    font-family: Calibri, Font1;  
}
```