УДК 004.056.53

ЗАЩИТА ВЕБ-СТРАНИЦ ОТ НЕСАНКЦИОНИРОВАННОГО КОПИРОВАНИЯ

© Е.С. Чиркин, Н.Л. Королева, В.П. Дудаков

Ключевые слова: защита веб-страниц от копирования; цифровые водяные знаки.

В сети Интернет существует проблема незаконного заимствования (текстового) контента, имеющего коммерческую ценность. Единственный вариант преодоления – не размещать его в свободном доступе. Если это невозможно, необходимо технически и юридически защищать его. Цель работы – показать возможности технической защиты от несанкционированного воспроизведения контента, ее несостоятельность и ее возможности в качестве вспомогательного инструмента, облегчающего последующую защиту в правовом поле.

Сеть Интернет является глобальной компьютерной сетью, объединяющей с помощью различных средств связи электронные хранилища информации, находящиеся в различных точках земного шара. При наличии доступа к сети Интернет (через телефонную сеть, через отдельный кабель, с помощью беспроводного соединения) можно получить необходимую информацию.

Сеть Интернет задумывалась как открытая система, и до начала коммерциализации ее контента вопросам его защиты не уделялось практически никакого внимания. Со временем стало очевидно, что часть публикуемых данных представляет известную ценность, но поскольку в саму идеологию системы эта возможность не была заложена, то в настоящее время существует проблема представления информации сети Интернет в коммерческих целях.

Защиту контента веб-страниц от несанкционированного копирования можно рассмотреть с нескольких, на первый взгляд, противоречивых позиций.

Первая – это юридическая. Наиболее простая в реализации для «защищающей» стороны, наименее сложная для преодоления и, самое главное, имеющая наибольшие последствия в правовом поле. Остальные способы защиты можно отнести к организационным и техническим мерам. Рассмотрим их использование на примере различных элементов контента веб-страницы:

1) текст – собственно текст на том или ином языке, предназначенный для восприятия человеком и представленный в т. н. «формате HTML» [1];

2) изображения – графические файлы различного формата, являющиеся либо самостоятельным контентом (*а*), либо иллюстрациями текста или иного материала (*б*);

3) мультимедиа информация, к которой в данном контексте следует отнести аудио- и видеоинформацию, представленные в разных форматах, которых, в общей сложности, существует не так много;

4) так называемый «flash» [2], представляющий из себя, по сути, программы (в традиционном понимании этого слова – объектный код и различные ресурсы), скомбинированные в файл специального формата. Данная «программа» исполняется особым образом (чаще всего) в контексте браузера с помощью программного обеспечения Adobe Flash Player;

5) прочие элементы веб-страницы – например, программный код на JavaScript, реализующий меню, уникальный дизайн и другие элементы, носящие как технический характер, так и оригинальные дизайнерские идеи.

Рассмотрим защиту этих элементов подробнее (п. 1 – защита текста – в последнюю очередь).

В п. 2*а*, если изображение представляет собой, например, фотографию высокого качества (призванную, например, показать высокое мастерство фотографа, высокую разрешающую способность матрицы цифровой фотокамеры и др.), какие-либо технические меры защиты следует считать нецелесообразными – изображение само по себе будет высокого разрешения допустимого качества для любой переработки и изменения, причем в итоге получится изображение также приемлемого для последующего воспроизведения качества. Максимум, что возможно сделать в данной ситуации, – это наложить на изображение водяные знаки, т. е. явно перевести защиту данного фрагмента документа в правовое поле.

В п. 2*б* защита изображения не представляет собой принципиальной сложности с тем незначительным исключением, что защита подобных изображений часто нерациональна по ряду причин: 1) требуются значительные вложения для ее реализации; 2) качество (или размер) изображения всегда можно понизить до величин, которые сделают его переработку уже нерациональным для злоумышленника; 3) это обычно действительно *иллюстрации* рядом представленного материала, и эти изображения обычно можно повторить, перерисовать, перевести в другую форму или формат без значительных трудовых затрат со стороны злоумышленника.

В п. 3 защита аудио- и видеоинформации представляет из себя определенную сложность, т. к. особенности ее воспроизведения связаны с необходимостью ее полного перемещения в браузер пользователя (возможно, по частям (блоками) или в форме потокового воспроизведения). По этой причине всегда можно сде-

лать аудио- или видеозапись соответственно, на компьютере конечного пользователя – без значительных потерь качества. Кроме, опять же, наложения различных водяных знаков – для явного указания на правообладателя, – иные виды защит следует признать нерациональными по причине их высокой трудоемкости (для «защищающейся» стороны).

Flash-ролики (п. 4) представляют собой уникальное явление – с одной стороны, flash-ролики являются декомпилируемыми (а сама платформа – интерпретируемая), т. е. возможно полное и корректное извлечение всех ресурсов и восстановление программного кода, с другой стороны, – это отдельная независимая программная платформа корпорации Adobe, которая предпринимает различные шаги по сохранению своих позиций на рынке, несколькими из которых является и защита контента, хранящегося, обрабатывающегося и передающегося через их платформу, т. е. средствами, преодоление которых затруднено обыкновенными широкодоступными инструментами.

Пункт 5 охватывает собой различные элементы вебстраницы, не охваченные ранее, – например, программные коды (на языке JavaScript), реализующие меню, переходы, затенения, эффекты, анимация, минигалереи и другие составляющие оригинального дизайна страницы. Здесь можно отметить: 1) программные коды (особенно, JavaScript) хорошо поддаются обфускации, после чего ими полноценно может воспользоваться только тот, кому доступны оригинальные коды; 2) элементы дизайна принципиально не патентуются (как *идеи*), либо это, наоборот, какие-то элементы, которые достаточно эффективно защищаются юридически (например, использование *образа* какого-либо *героя*), либо их защита непринципиальна ввиду повторяемости (следует отметить, что стоимость создания уникального дизайна средней степени сложности на рынке представляет собой не очень большую величину).

Здесь следует различить принципиальную невозможность извлечения текста и массовое (потоковое) извлечение текстового контента (второе часто требуется злоумышленникам для скорейшего создания своих сайтов-копий на заданную тематику, либо еще чаще – для создания автоматических «наполнителей» своих сайтов материалом с чужих сайтов). Программные инструменты для последнего случая являются распространенными.

Защита текста представляет собой технически сложную проблему – сложно защитить по следующим причинам:

– текстовая информация попадает в браузер пользователя в своем «естественном» «текстовом» виде и может быть выделена и извлечена (скопирована);

– браузер – это платформа, которая исполняется в рамках операционной системы – с помощью специальных программ, предназначенных для создания «снимков» (графических копий) экрана всегда возможно скопировать содержимое документа (впоследствии с помощью программ для распознавания текста (OCR) данные изображения можно обратно перевести в текстовую форму). Более того, учитывая то, что изначально текстовая информация была в т. н. «изначально электронном» качестве, преобразование ее без ошибок и искажений обратно в текст будет осуществлено с качеством, асимптотически приближающимся к 100 % (например, с помощью [3]);

– в качестве браузера всегда может выступать не только традиционный браузер, а любая программная платформа, созданная заинтересованным программистом, как представляющая из себя браузер, так и имитирующая своей работой любой из браузеров. Учитывая то, что современные браузеры построены, в основном, на одних и тех же «движках» – программных платформах, имеющих открытый исходный код, написать заинтересованной стороне свой браузер, ничем не отличающийся от «настоящего», несложно;

– существует несколько достаточно эффективных методик защиты текстового контента, основанных на AJAX-технологиях и защите контента посредством того, что текст изначально не представлен как «текст», а формируется программным кодом в контексте браузера незадолго до его фактического вывода на экран (при загрузке страницы). Сложности данного подхода связаны с крайне низким быстродействием JavaScript-кода (в данном применении), а также завышенными требованиями по широте полосы доступа в Интернет. Подробнее этот подход рассмотрен в работе [4];

– в случае ценного контента затраты на его «ручное» переписывание (перепечатку) с устройства воспроизведения можно считать незначительными.

Таким образом, можно сделать три вывода.

1. Принципиально невозможно защитить от несанкционированного заимствования опубликованную информацию в сети Интернет.

2. Технически от массового копирования и дублирования контента защититься возможно, причем эти решения, по сути, уже реализованы в некоторых системах управления контентом.

3. Технически не очень сложно затруднить доступ неквалифицированных пользователей к элементам вебстраницы (различные методы вроде запрета выделения (как через CSS, так и через JavaScript), запрета копирования (аналогично), использования прозрачных изображений поверх «настоящего» содержимого, фрагментарных изображений, помещения контента во flash-ролик и т. п.).

Все вышесказанное касается Интернета и материала, находящегося в свободном доступе. Вместе с тем всегда есть уникальный материал, копирование которого (как законное, так и незаконное) является неэффективным подходом по сравнению с использованием оригинальных источников (например, материал новостных сайтов): любой пользователь рано или поздно предпочтет первоисточник любому скопированному материалу.

Также следует заметить, что в настоящее время увеличивается количество исков (и, скорее всего, их количество будет увеличиваться) о признании того или иного материала незаконно воспроизведенным (при этом может помогать депонирование работ, их государственная регистрация и другие предварительно принятые меры). Причем часть подобных претензий урегулируется в досудебном порядке.

В отдельных случаях технические меры защиты могут быть эффективны вплоть до их целенаправленной ручной нейтрализации, если они применяются, например, для защиты материала определенного рода, не

предусматривающего свободный доступ к нему неограниченного круга лиц.

Например, материал обучающей направленности по подготовке специалистов в области информационных технологий. Всякий обучающий материал в этой области быстро устаревает – часто в течение года-двух, в любом случае через три-пять лет этот материал станет историей и будет требовать самой тщательной переработки. Более того, это порождает особый интерес к актуальным обучающим курсам – они востребованы (как слушателями, так и злоумышленниками). Второй момент – трудоемкость их создания очень высока. Поэтому отдельный материал стоит защищать как юридически, так и технически. Возможно, это не совсем рационально без рассмотрения каждого конкретного случая, но в целом (например, для сохранения репутации и минимизации финансовых потерь) это необходимо.

Таким образом, особое место в защите контента от незаконного использования занимает облегчение проведения экспертизы, подтверждающей факт незаконного использования контента (тем более что обычно в рамках гражданских исков сначала свою правоту с помощью экспертизы подтверждает сторона, имеющая претензии). С этой точки зрения, определенный вес обретают инструменты, облегчающие в последующем создание доказательной базы. Проще всего ее обеспечить путем пометки документов (как всего массива учебного материала, так и персонально для каждого обучающегося). Это не сделает невозможным несанкционированный доступ к нему (уничтожение, модификацию, похищение), но позволит впоследствии установить, кто именно это сделал или по чьей вине это произошло. Существуют различные способы внесения данных меток, в т. ч. и способами, не допускающими их обнаружение и отделение от самого электронного документа, даже если злоумышленник знает об их возможном наличии в документе. В последнем случае данные метки называются цифровыми водяными знаками (ЦВЗ) [5].

Задача защиты текстов обучающих курсов в форме, допускающей их размещение в формате веб-сайта, упрощается ввиду, с одной стороны, стандартности спецификаций представления этого материала, а с другой – особенностей толкования этих спецификаций различными производителями разных программных продуктов (что вызвано как реальными, так и мнимыми «нестыковками» реализаций, что порождает различные методы повышения взаимной совместимости, чем и можно воспользоваться).

В связи с этим можно выделить несколько методов внесения меток ЦВЗ в HTML-документы (на рис. 1*а* – получаемый результат, без скрытой информации, ему соответствует HTML-код с рис. 1*б*):

1) самый простой и очевидный метод – использование символов различных алфавитов схожего начертания. Например: «а» (RU) → «a» (EN). Это эффективный способ, не имеющий визуальных недостатков и, таким образом, никак не изменяющий качество представления и оформления учебного материала. Недостаток данного способа – продолжение его достоинств: неотличимый символ не будет отличаться при обработке снимков текста, сделанных специальными программами (или мониторов – цифровыми фотоаппаратами), посредством OCR-программ, а при копировании материалов прямо со страницы (возможно, в обход защитных механизмов) он будет отображаться в текстовых редакторах, имеющих механизмы проверки орфографии как ошибочно написанный (из-за содержащихся в нем «посторонних» символов, т. е. как бы «опечаток») и, вероятно, будет исправлен. То есть данный символ именно как «метка» будет утерян при трансформациях данного текста;

2) использование пробелов (см. рис. 1*в*) – один из самых эффективных и труднообнаружимых способов, который не искажает текст и обладает устойчивостью к различным преобразованиям текста. Согласно спецификации HTML, все подряд идущие пробельные символы при отображении текста выводятся как один пробел. Таким образом, возможно формирование визуально неотличимого от оригинала текста, который может содержать практически неограниченный объем скрытых меток. Если при этом вспомнить, что символов, обозначающих пробел на самом деле не один, а около двух десятков (например, т. н. пробелы «пробелы двойной ширины», «тонкий пробел», «пунктуационный пробел» и др.) (исторически их появление было связано либо с особенностями набора текста на печатных автоматах в типографиях, либо с улучшением читабельности текста), то появляются основания для более глубокого внедрения меток посредством пробельных символов сразу в несколько «слоев» документа – например, в HTML-код, в «видимые» пробелы между символами и третий слой – пробелы, прилегающие к знакам препинания. Все недостатки данного способа связаны с целенаправленным уничтожением «лишних» пробельных символов, что возможно нивелировать многослойным использованием данного подхода;

3) особенности внутреннего представления HTML-страниц и кодирования текстовой информации: например, символы видимого текста в HTML-коде страницы возможно задавать не только с помощью соответствующих символов, но и *сущностей* (entity) [6] HTML (см. рис. 1*г*), а также кодов этих символов (см. рис. 1*е*). По вышесказанным причинам эти же сущности сохраняют свое представление не в виде символов. Данный способ не имеет недостатков, более того, часто по различным причинам даже остается «незаметным» при ручном вычищении HTML-кода;

4) особенности внутреннего представления HTML-страниц и кодирования текстовой информации: например, визуально текст представляет из себя одно слово, а на самом деле в HTML-коде – начало слова, комментарий HTML (который визуально не отображается и ничего из себя не представляет) и окончание слова (см. рис. 1*д*). Ввиду особенностей работы большинства браузеров и текстовых процессоров существует высокая вероятность, что при копировании текста со страницы – комментарий (разумеется, в его неизменном виде) так же будет скопирован и вставлен в новое место. Более того, есть ненулевая вероятность, что комментарий будет сохраняться при различных дальнейших трансформациях текста вплоть до последующего опубликования (злоумышленником) получившегося текста в формате HTML. Единственный недостаток данного способа – при визуальном просмотре кода страницы комментарии, разрывающие слово, могут насторожить злоумышленника и будут удалены.

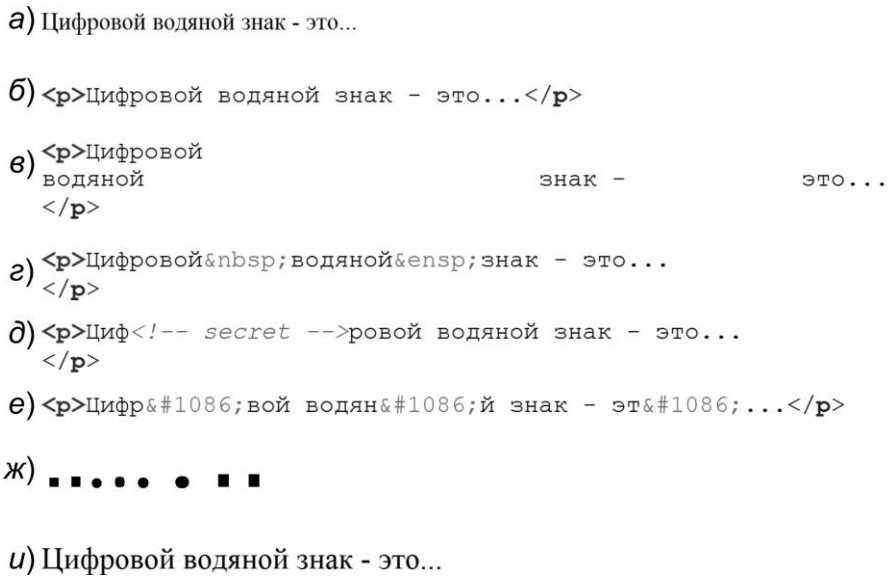

Рис. 1. *а* – результат, *б–и* – способы внесения скрытой информации

5) кодирование меток посредством изменения знаков препинания (см. рис. 1*ж*). Начертание разных символов разных шрифтов различно. При этом знаки препинания (особенно запятые) – повсеместно используемые символы, которые не привлекают внимания. При копировании текста со страницы обычно сохраняется его стилевое оформление, которое соответственно сохраняет «другой» знак препинания, пусть и имеющий тот же семантический смысл, что и соответствующий знак, оформленный основным стилем абзаца, но несущий сразу два признака – иное начертание (например, для визуальной оценки текста (если разрешение изображения достаточно)) и тот факт, что символ набран другим шрифтом (как точный факт). Следует отметить особенность форматирования разностилевого текста в наиболее популярных текстовых редакторах – при изменении стиля текста стиль отдельных символов чаще всего остается неизменным. Данный способ весьма незаметен, если исходный текст подвергается незначительной правке злоумышленником (на рис. 1*ж* представлена точка в различных шрифтах, слева направо: Arial, Cambria, Times New Roman, Calibri, Constantia, Courier New, Tahoma, Verdana);

6) кодирование межсимвольными и межстрочными интервалами. Посредством стилевого оформления текста (с помощью каскадных таблиц стилей (CSS)) можно в определенных пределах видоизменять видимое представление текста и это оформление также будет сохраняться при различных преобразованиях текста. Такими признаками могут быть межсимвольный интервал, отступы у абзацев, межстрочные интервалы и т. п. (при этом, например, совсем необязательно эти величины делать «видимыми». Например, невозможно визуально отличить текст, набранный шрифтом 10 кегля, от текста, набранного шрифтом 10,01 кегля, – обычные текстовые редакторы не допускают дробности больше, чем 0,5 кегля (при вводе текста в редакторе, но не при обработке текста, полученного из сторонних источников), оформление же посредством CSS не имеет такого ограничения. На рис. 1*и*, если разрешение устройства вывода позволяет либо доступен электронный вариант, заметно, что буквы «о», «а», «з» на самом деле увеличены по сравнению с остальными строчными буквами, причем «о» больше, чем «а» и «з» ;

7) метод целенаправленного внесения незначительных искажений. Например, написание одного или двух терминов или, наоборот, общеупотребимых слов с ошибками, которые непосвященное лицо будет считать опечаткой. Метод крайне эффективен, используется в настоящее время, но при многократном использовании – слишком заметен и легко исправляется.

Также проблемой можно назвать то, что все современные браузеры для настольных платформ (кроме Microsoft Internet Explorer) в той или иной степени (могут быть) ориентированы на разработчиков сайтов, т. е. имеют инструменты по различному извлечению контента из веб-страницы, в какой бы форме он ни был представлен. По этой же причине крайне неэффективно программное обеспечение, защищающее веб-страницы посредством шифрования их кода.

Везде в тексте подразумевается, что метод является эффективным, если внесенные метки, вне зависимости от их заметности, злоумышленнику придется целенаправленно вручную уничтожать, и максимально эффективным, если данные метки незаметны для опытного пользователя.

Таким образом, комбинируя данные методы, возможно создать HTML-документ, содержимое которого невозможно скопировать обыкновенному пользователю, который создаст определенные трудности для опытных пользователей и программистов (вплоть до ручного уничтожения всех найденных меток или создания специальных инструментов по их удалению), а также содержимое которого останется помеченным с помощью ЦВЗ для последующих разбирательств в правовом поле. Учитывая то, что все вышеперечисленные методики не требуют значительных вложений на их реализацию, следует сделать вывод, что их применение для защиты материала обучающей направленности эффективно в мере, достаточной для, например, кратковременного сохранения лидерства на рынке того или иного обучающего курса. В рамках данной работы был

создан программный комплекс по комплексному внесению ЦВЗ в HTML-документы с помощью вышеописанных подходов.

ЛИТЕРАТУРА

1. HTML. URL: http://www.w3.org/html/.
2. Adobe Flash Platform. URL: http://www.adobe.com/flashplatform/.
3. ABBYY Screenshot Reader. URL: http://www.abbyy.ru/screenshot_reader/.
4. *Чиркин Е.С., Мазур М.И.* Система предотвращения несанкционированного доступа к веб-страницам: электронный ресурс. ФГУП НТЦ «Информрегистр», 2010. Номер государственной регистрации 0321001518.
5. *Оков И.Н., Ковалев Р.М.* Электронные водяные знаки как средство аутентификации передаваемых сообщений. СПб.: Конфидент, 2001. 80 с.
6. Character entity references in HTML 4. URL: http://www.w3.org/TR/html4/ sgml/entities.html.

БЛАГОДАРНОСТИ: Работа выполнена при финансовой поддержке Российского фонда фундаментальных исследований, проект №12-07-00512.

Поступила в редакцию 10 сентября 2012 г.

Chirkin E.S., Koroleva N.L., Dudakov V.P. PROTECTING CONTENT OF WEB PAGES FROM UNAUTHORIZED COPYING

The Internet has a problem of illegal taking of commercial content. The only way to overcome is not to put it in the public access. If this is not possible, it is necessary to protect it technically and legally. The purpose of the work is to show the possibilities of technical protection against unauthorized reproduction of content, its failure and its potential as a support tool to facilitate the subsequent protection of the legal field.

Key words: web pages protection from copying; digital watermarks.