УДК 681.3

В.Л. Платонова

ОТДЕЛЬНЫЕ АСПЕКТЫ ЗАЩИТЫ WEB-ПРИЛОЖЕНИЙ[1]

Web-приложения становятся все более распространенными, особенно в сферах i-бизнеса и деятельности организаций, которым важно использовать современные способы передачи информации. Web-приложения значительно усложнились и увеличили спектр своих возможностей за последние годы. Поскольку запуск Web-приложений на сервере организации может беспрепятственно осуществить каждый посредством сети Интернет, к такого рода программным продуктам должны предъявляться повышенные требования безопасности.

Для функционирования Web-приложений необходим вычислительный сервер (компьютер серверного типа или обычный персональный компьютер, в дальнейшем просто сервер) или группа серверов с различными сервисами (WWW, FTP и прочие при необходимости), подключенные к сети Интернет, а также специальное приложение, Web-сервер, который обеспечивает обработку запросов из сети и выполнение серверных Web-приложений.

Обеспечивать защиту необходимо на трех уровнях (рис. 1):

1) уровень политики безопасности всего сервера в целом средствами операционной системы и дополнительных программных средств;

2) уровень политики безопасности Web-сервера и дополнительных модулей Web-сервера;

3) уровень политики безопасности, реализуемой самими Web-приложениями.

1. На первом уровне осуществляется защита от внешних вторжений и несанкционированных действий пользователей и процессов. Защита на данном уровне обеспечивается системным администратором благодаря таким средствам, как firewall, антивирусы, политика безопасности ОС, в том числе грамотным распределением прав доступа процессов к файловой системе. Этот набор мероприятий позволяет изолировать процессы друг от друга, и в случае нестандартного сбойного поведения одного из процессов его действия не угрожают функционированию других процессов, а значит, и всему серверу в целом.

2. Работа Web-сервера – принимать запросы от Web-клиентов и возвращать необходимые данные. Защита на данном уровне заключается в ограничении доступа к файловой системе и к выполняемым скриптам. Угрозой Web-серверу могут быть некорректные запросы, эксплоиты, а также большое количество сообщений нестандартной длины. Web-сервер один из наиболее часто атакуемых сервисов, так как наиболее часто используемый и, зачастую, открытый всему Internet'у, поэтому фильтрация пакетов крайне затруднена, а противостоять вышеуказанным угрозам можно только непосредственно фильтрацией.

Рис. 1. Уровни защиты web-приложений

3. Web-приложение может запустить на сервере любой пользователь Internet. Единственный способ передавать данные – это GET и POST параметры. Злоумышленник, изменяя эти параметры запроса Web-серверу, может добиться некорректной работы Web-приложения, что является прямой угрозой безопасности всего сервера в целом. Отразить атаки подобного типа возможно только на уровне Web-приложения, предъявляя повышенные требования к устойчивости кода самого Web-приложения. К примеру, пусть на Web-приложении осуществляется работа с БД на языке SQL. Если часть запроса является входными данными приложения, то злоумышленник получает возможность формировать свои подзапросы к БД, которые будут выполнены. Поэтому входные данные должны иметь только ту структуру, которую предусмотрел разработчик, остальные должны либо преобразовываться к необходимому виду, либо не обрабатываться вовсе.

[1] Работа выполнена под руководством канд. техн. наук, проф. Ю.Ф. Мартемьянова.

Злоумышленники могут атаковать комплекс на любом из уровней. Обеспечивать защиту на первом и втором уровнях должен системный администратор, а ответственность за защиту Web-приложений непосредственно несет Web-разработчик.

Учитывая вышесказанное, при разработке Web-приложения необходимо, чтобы приложение:

1) получало минимум данных от пользователя, так как чем меньше данных, тем легче фильтровать и обрабатывать их;

2) не обнаруживало используемый для разработки Web-приложения язык, структуру сервера и данных (зная язык разработки, гораздо проще сформировать такой запрос, который произведет сбой в системе; то же относится к структуре сервера и данных: чем больше информации имеется, тем эффективнее можно произвести атаку);

3) отображало минимум информации в случае некорректных действий пользователя; (если ошибка произошла, сообщение не должно выводить те данные, которые может использовать злоумышленник);

4) фиксировало заранее определенный класс действий, которые могут привести к взлому (т.е., зная основные способы взлома, не допускать подобных действий);

5) сохраняло данные в архив, чтобы в случае кражи или изменения данных можно было восстановить их;

6) было «прозрачно» для разработчика, т.е. чтобы код имел четкую логическую структуру, для минимизации количества ошибок, а также при возникновении ошибки можно было скорейшим образом исправить код.

Выполнение этих требований несколько усложнит разработку, но значительно повысит уровень безопасности Web-приложения, и следовательно, уровень защищенности сервера в целом.

Кафедра «Информационные технологии и защита информации»