Рекомендации по минимизации возможных угроз информационной безопасности информационным ресурсам Российской Федерации

1. Для веб-приложений:

1. Использовать защищенные протоколы TLS v1.2 (и выше) при прохождении процедуры аутентификации пользователей в веб-приложении.

2. Запретить предоставлять в выводе сообщений об ошибках следующую информацию:

- данные о структуре файловой системы (информация о версии операционной системы, директориях с системными файлами и системным программным обеспечением, включая пути к директориям и файлам);
- фрагменты программного или конфигурационного кода;
- сообщения об ошибках при передаче запросов в СУБД;
- SQL-выражения, используемые при доступе к базе данных.

3. Выдавать пользователю страницу-заглушку с кодом HTTP-ответа веб-сервера «200» при обработке ошибок веб-сервером.

4. По возможности ограничить использование при обработке веб-сервером данных в формате XML внешних сущностей (External Entity), внешних параметров сущностей (External Parameter Entity) и внешних описаний типа документа (External Doctype), а также JSON.

5. Запретить кеширование веб-форм ввода конфиденциальной информации. Выставить атрибут HTTPOnly у параметров cookie, значения которых не должны быть доступны сценариям, выполняемым браузером. У параметров cookie, содержащих чувствительную информацию, необходимо выставить атрибут secure.

6. Проводить проверку корректности вводимых пользователем данных как на стороне клиента (с использованием сценариев, исполняемых браузером), так и на стороне сервера.

7. Использовать директивы в заголовках сообщений HTTP, определяющие применяемую кодировку. Исключить использование разных кодировок для разных источников входных данных.

8. Использовать параметризованные запросы (например, хранимые процедуры) для построения SQL-запросов. В случае отсутствия такой возможности, организовать процедуру предварительной обработки получаемых от пользователя данных (путем удаления метасимволов « ` – / *», а также следующих SQL-операторов: SELECT, UNION, ALTER, UPDATE, EXEC, DROP, DELETE и INSERT).

9. Осуществлять преобразование HTML-кода входного потока данных следующим образом:

- заменить < > на &lti >
- заменить () на &#40 и)

- заменить # на #

- заменить & на &#38.

10. Осуществлять фильтрацию входного потока данных (например, с использованием методов Server.HTMLEncode и HttpServerUtility.HTMLEncode в ASP и ASP.NET).

11. Запретить пользователю ввод данных, в которых допустимы HTML-теги <IMG> или <TABLE>.

12. Для подсистем управления сессиями пользователей:

- организовать авторизованному пользователю веб-приложения возможность самостоятельного завершения сеанса работы в веб-приложении.

- обеспечить гарантированное удаление идентификатора соответствующей сессии по завершении сеанса работы клиента веб-приложения.

- ограничить время жизни активной сессии пользователя.

13. Для подсистем разграничения доступа:

- организовать доступ к защищенным ресурсам веб-приложения только после прохождения процедуры аутентификации;

- обеспечить хранение аутентификационных данных пользователей веб-приложения только в криптографически защищенном виде;

- исключить хранение аутентификационных данных (от веб-приложений, СУБД, ТКО, FTP и т.п.) в файлах конфигурации, доступных путем обращения к ним по URL;

- исключить хранение в HTML-страницах аутентификационных данных, а также информации, позволяющей сделать вывод о структуре каталогов веб-приложения на веб-сервере;

- в случае, если в веб-приложении предусматривается возможность внесения изменений пользователем в принадлежащий ему профиль, внесенные изменения необходимо подтверждать дополнительной процедурой аутентификации;

- запретить использование заголовка REFERER в качестве основного механизма авторизации.

14. Отказаться от использования на веб-ресурсах (в том числе веб-сайтах) компонентов и контента, подгружаемых с внешних, не контролируемых организацией, ресурсов.

15. В случае невозможности отказа от использования указанных компонентов и контента осуществлять их проверку на предмет вредоносного воздействия на отображаемую в браузерах пользователя информацию и возможность кражи аутентификационных данных и файлов-cookie пользователей. Далее осуществлять периодическую проверку их хэш-сумм. В случае изменения хэш-сумм – блокировать использование указанных компонентов и контента на веб-ресурсе и осуществлять их повторную

проверку функциональности. В случае отсутствия потенциально вредоносного функционала – проводить дальнейшее сравнение по новой хэш-сумме.

2. Для службы доменных имен (DNS)

2.1. Обеспечить наличие у организации прав на свои доменные имена.

2.2. Обеспечить разнесение ролей DNS-серверов «User Primary DNS Server»[1] и «Domain Primary DNS Server»[2] на разные физические и/или виртуальные серверы.

2.3. В части «Domain Primary DNS Server»:

- запретить рекурсивные запросы разрешения доменных имён;
- запретить разрешение доменных имён объектов, не относящихся к информационным ресурсам организации;
- настроить механизмы защиты от спуфинг-атак;
- запретить уведомления и перенос зон произвольными объектами сети Интернет. Настроить список доверенных DNS-серверов;
- настроить правила предварительной фильтрации поступающих запросов (Таблица № 1).

Таблица № 1: Правила фильтрации запросов.

Описание	**IP-адрес источника**	**Сетевой порт источника**	**IP-адрес назначения**	**Сетевой порт назначения**
Входящий запрос	Любой	53/udp; 53/tcp; >1023/udp; >1023/tcp.	IP-адрес DNS-сервера	53/udp; 53/tcp.
Ответ на запрос	IP-адрес DNS-сервера	53/udp; 53/tcp.	Любой	53/udp; 53/tcp; >1023/udp; >1023/tcp.

2.4. Запретить в качестве «User Primary DNS Server» использовать DNS-серверы, расположенные за пределами Российской Федерации (например, перейти на использование НСДИ).

[1] User Primary DNS Server – DNS-сервер, отвечающий на запросы пользователей информационных ресурсов организации по разрешению доменных имён объектов сети Интернет.

[2] Domain Primary DNS Server – DNS-сервер, отвечающий на запросы пользователей сети Интернет и других DNS-серверов по разрешению доменных имён, принадлежащих информационным ресурсам организации.