

Проблемы информационной безопасности ву-нета

Докладчик:

Финансовый директор СООО «Белсек»

Мартинкевич Дмитрий Станиславович



Причины:

- Недостаточное понимание проблем. Недооценка важности стабильной работы интернет ресурса (начиная от сайта визитки и заканчивая интернет порталом)
- Экономия на разработке приложений
- Некомпетентность разработчиков в вопросах безопасности
- Ошибки в ПО (движок, уязвимости в скриптах)
- Неудачно спроектированное оборудование и программы
- Неправильное распределение прав доступа между пользователями
- Ошибки при настройке
- Отсутствие информации об уязвимостях
- Использование оборудования или программ не по назначению
- Сбои оборудования
- "Секретные" недокументированные функции в программах, оставленные разработчиками

Последствия:

- Подмена корпоративной информации
- Ухудшение имиджа компании
- Уничтожение информации
- Раскрытие конфиденциальной информации
- Несанкционированный доступ в ресурсы компании
- Жалобы клиентов на качество веб ресурса
- Воровство финансовой информации, денежных средств
- Нелегальное использование ресурсов рабочих серверов

Набор уязвимостей «джентльмена»

Внедрение операторов SQL

Подбор

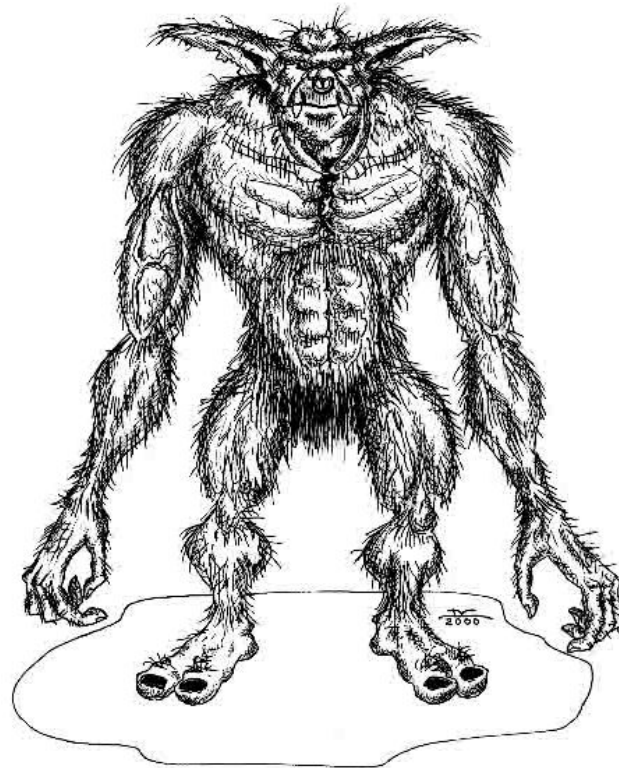
**Предсказуемое значение
идентификатора сессии**

Межсайтовое выполнение сценариев

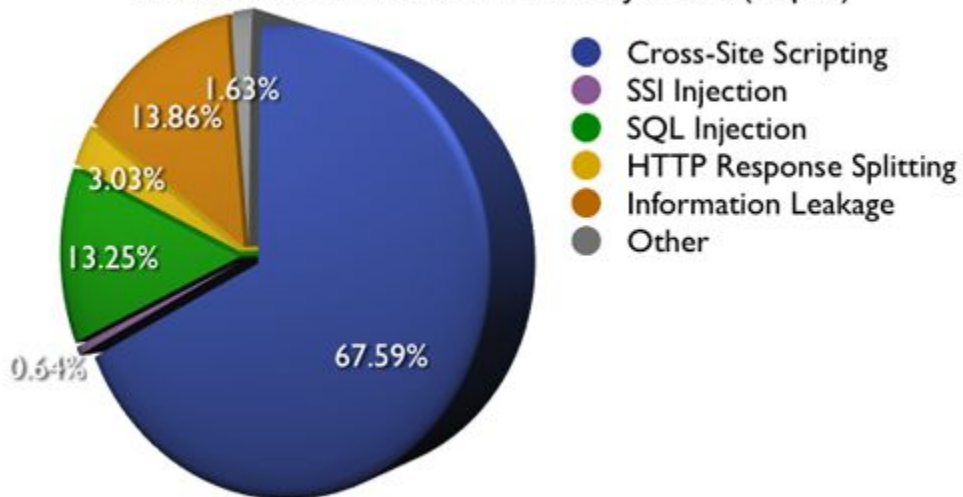
Переполнение буфера

Выполнение команд ОС

Злоупотребление функциональными возможностями



Most common vulnerabilities by class (Top 5)

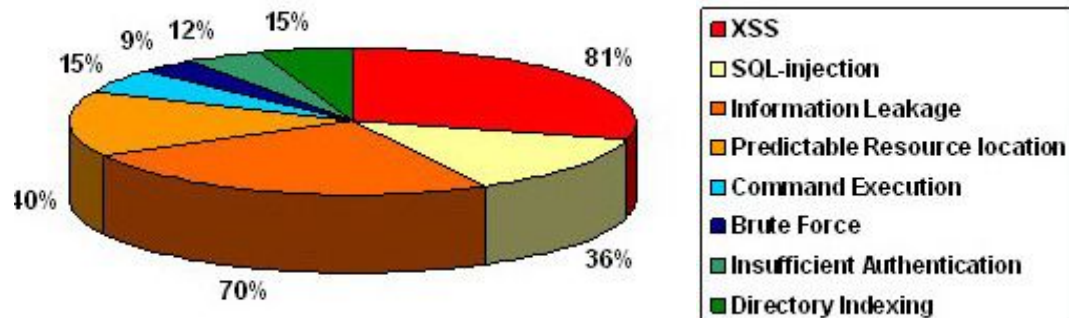


Наиболее распространенными уязвимостями являются «Межсайтовое выполнение сценариев», «Внедрение операторов SQL» и различные варианты утечки информации.

- до **8%** Web-приложений содержат уязвимости высокой степени риска, идентифицируемые с помощью автоматизированных средств;
- вероятность обнаружения критичной ошибки в динамическом Web-приложении при ручном анализе составляет **65%**;
- корректное применение автоматизированных средств анализа уязвимостей позволяет идентифицировать до **80%** всех недочетов;
- ряд критичных уязвимостей не может быть обнаружен с помощью автоматизированных средств.

Уязвимости by - нета

Для проведения анализа by-нета, мы воспользовались поисковиком Google. Были протестированы 100 сайтов по запросу site:by. (выборка была произвольной)



У 76% сайтов были обнаружены уязвимости различной степени риска.

XSS – 81%

SQL – injection - 36%

Information Leakage - 70%

Predictable Resource location – 40%

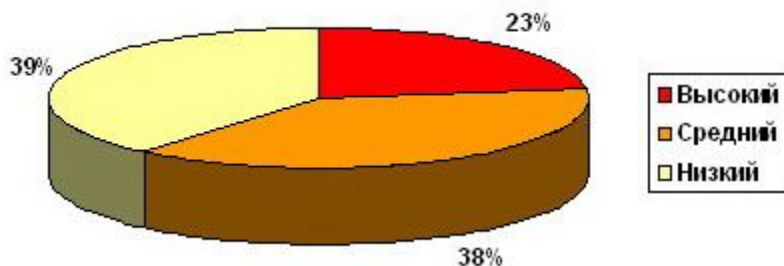
Command Execution – 18%

Brute Force – 7%

Insufficient Authentication 12%

Directory Indexing - 15%

Распределение уязвимостей по степени риска:



Сайт: www.dyriavyi_sapog.by

Уязвимость в системе отображения новостной ленты. (*Information leakage + SQL – injection*)

Уязвимость заключается в недостаточной проверке входных данных поля date в сценарии E:\INETPUB\www.dyriavyi_sapog.by\ww\main.inc, в строке 238.
Требуется **усовершенствование механизма фильтрации ВХОДНЫХ ДАННЫХ.**

Уязвимости в системе поиска по сайту. *Отображение ASP-кода. (Information leakage)*

В ситуации с запросом, содержащим в себе определенные строки, в результатах поиска можно увидеть ASP-код страницы.
Решение: Требуется **усовершенствование механизма поиска.**

Уязвимость в системе поиска по сайту. (*Подверженность DoS-атакам*).

С каждым поисковым запросом система выдает сообщение о просмотре 745 документов, т.е. используется рекурсивный поиск с просмотром всех документов в контексте web-сервера.
Решение: **Использование вместо рекурсивного поиска индексации контента.**

Уязвимость в системе опросов пользователей. (*SQL – injection*)

Система опроса пользователей и ведения статистики подвержена атакам SQL-injection из-за недостаточной проверки входных данных во всех полях. Решением будет **реализация проверки ВХОДНЫХ ДАННЫХ в lib/oprosnik_engines.asp.**



Технологии обеспечения безопасности

Автоматическое сканирование

Автоматическое сканирование осуществляется с помощью специального программного продукта, данный вид сканирования на практике выявляет до **80%** процентов уязвимостей.

Имитация направленных внешних атак

Данный вид анализа и поиска уязвимостей является наиболее полным, т.к. в данной ситуации специалист ИБ проводит полное обследование определенных параметров используя всевозможные варианты атак.

Анализ исходного кода – наиболее трудоёмкий, но самый эффективный способ поиска уязвимостей. Проверка исходных кодов веб приложения является часто более эффективной в идентификации слабых мест. Во время тестирования эксперты проверяют каждую строчку исходного текста.

Рекомендации по разработке веб – ресурса, отвечающего критериям безопасности:

Новая разработка

Профессиональный разработчик

Выбор хостинга

наличие соседей, и их уязвимости

Корректная конфигурация сервера

наличие `mod_rewrite`

Пароли отвечающие критериям безопасности

Сложность подбора

Различные пароли для отдельных сервисов



Рабочий ресурс

Пароли отвечающие критериям безопасности

При использовании систем управления контентом (CMS), гостевых, чатов, форумов с открытым исходным кодом требуется постоянное исследование на появление уязвимостей в скриптах (просмотр bug track-ов).

При использовании собственных разработок - тестирование ПО на наличие уязвимостей.

Спасибо за Ваше внимание!

СООО «Белсек»

*Надежная защита Ваших
информационных технологий!*

Тел.: +375 17 216-91-18

Тел./Факс: +375 17 216-94-01

Web: www.belsec.com

E-mail: info@belsec.com