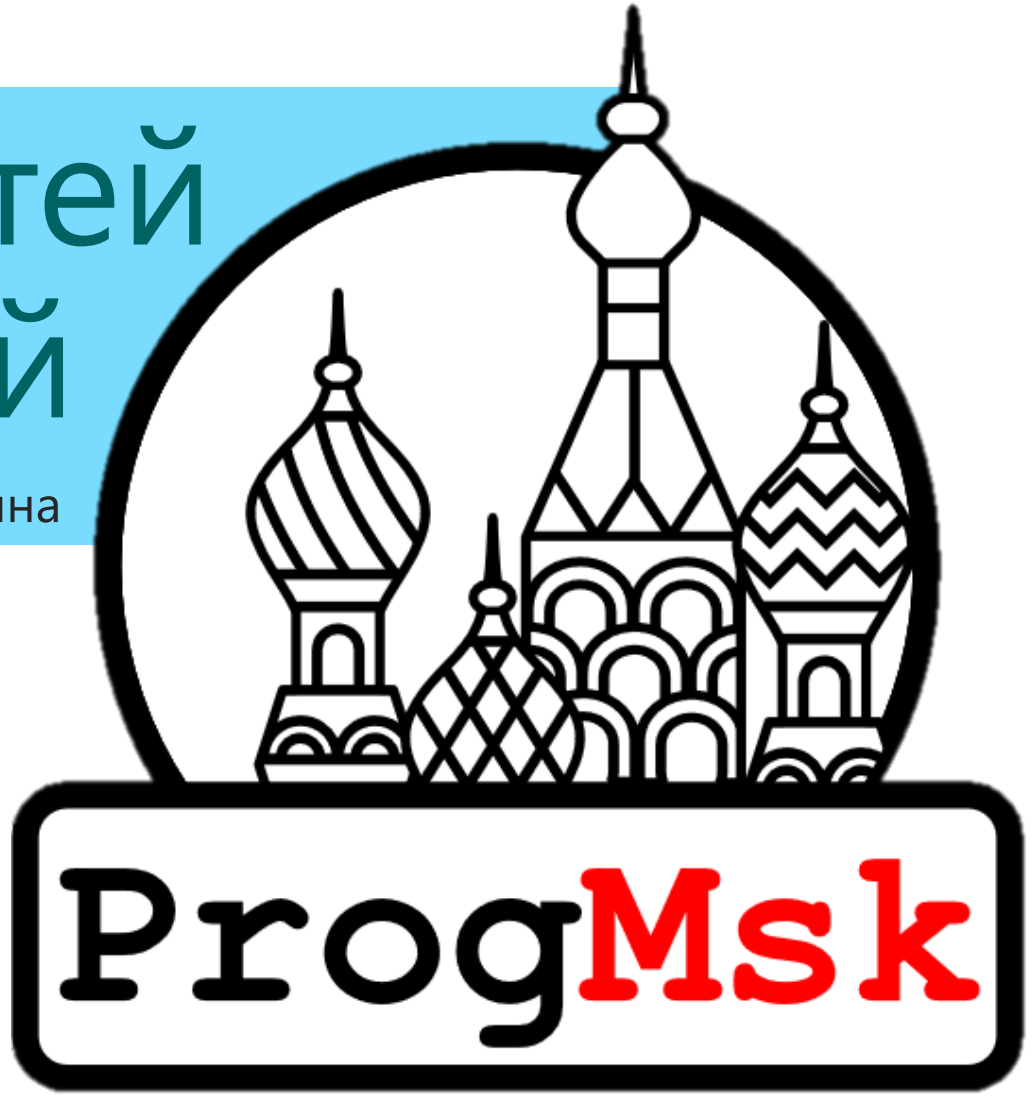
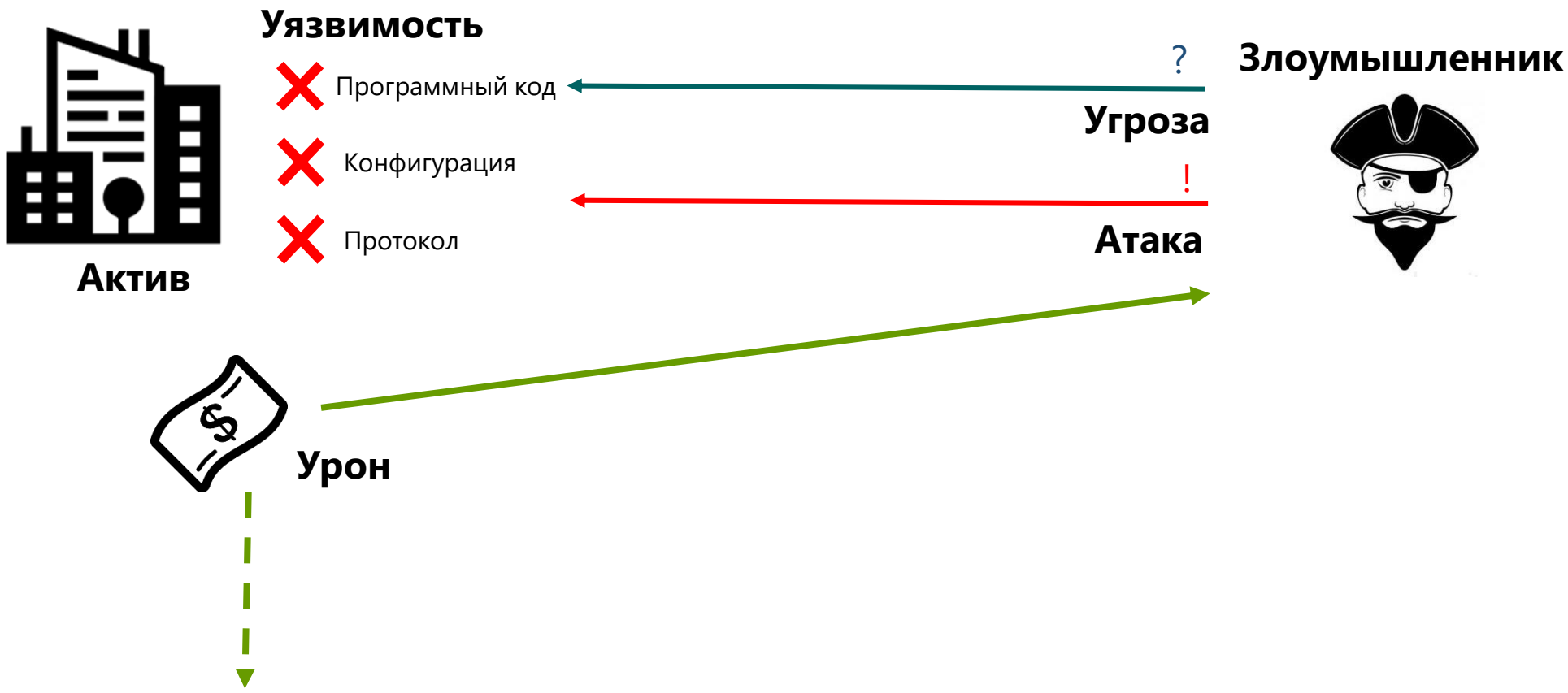
Топ 10 уязвимостей веб-приложений

Нина Пакшина



Основные понятия кибербезопасности

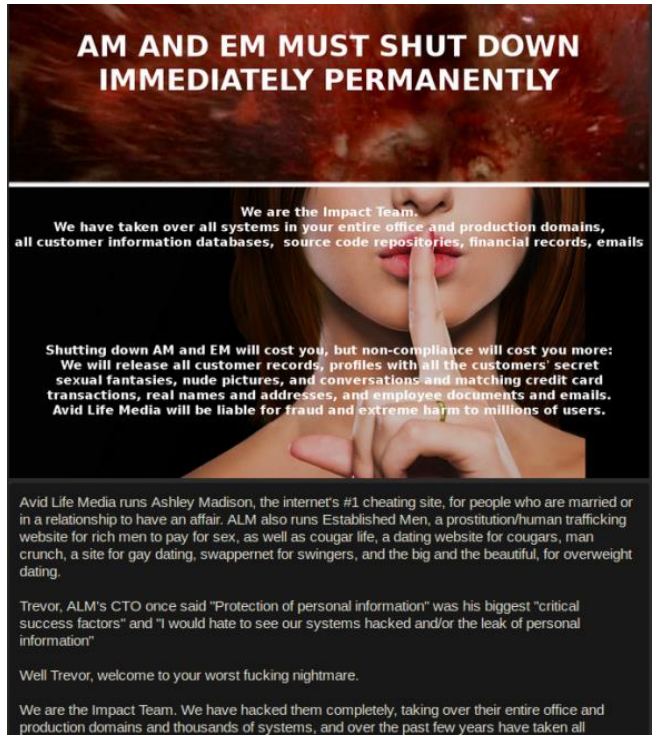


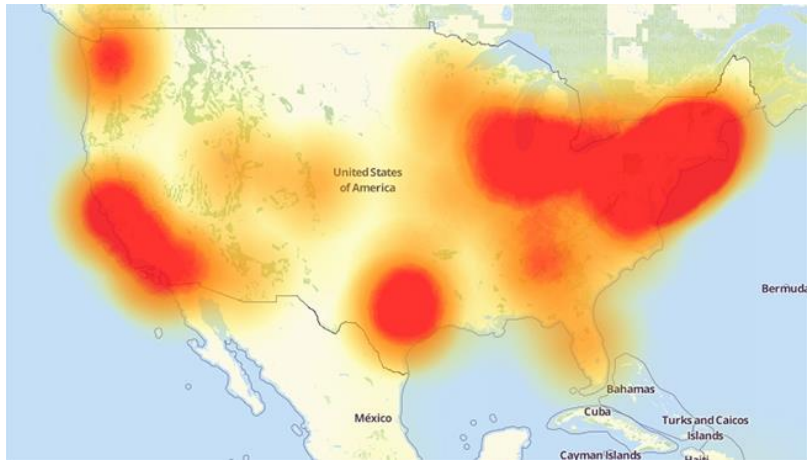
Атаки

Атака на сбор информации

Атака на доступ

Атака отказа в обслуживании





Потери от киберугроз

Прямые потери

Потери от простоя работы

Урон репутации

Штрафы

Закон об обработке персональных данных 152-ФЗ от 01.07.2017

Обеспечивать сохранность носителей с персональными данными, исключая их утечку, порчу, кражу, копирование

Для индивидуальных предпринимателей до 20 000 р.

Для юридических до 50 000 р.

GDPR

Штраф в размере до 4% от общего годового оборота компании или до 20 миллионов евро

OWASP

Open Web Application Security Project

- ANCAP
- Aspect Security
- AsTech Consulting
- Atos
- Branding Brand
- Bugcrowd
- BUGemot
- CDAC
- Checkmarx
- Colegio LaSalle Monteria
- Company.com
- ContextIS
- Contrast Security
- DDoS.com
- Derek Weeks
- Easybss
- Edgescan
- EVRY
- EZI
- Hamed
- Hidden
- I4 Consulting
- iBLISS Segurana & Inteligencia
- ITsec Security
- Services bv
- Khallagh
- Linden Lab
- M. Limacher IT Dienstleistungen
- Micro Focus Fortify
- Minded Security
- National Center for Cyber Security Technology
- Network Test Labs Inc.
- Osampa
- Paladion Networks
- Purpletalk
- Secure Network
- Shape Security
- SHCP
- Softtek
- Synopsis
- TCS
- Vantage Point
- Veracode
- Web.com

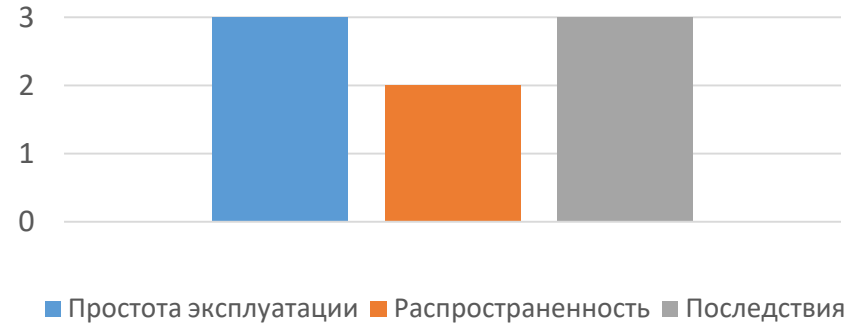
OWASP

OWASP Top 10 - 2013	→	OWASP Top 10 - 2017
A1 – Injection	→	A1:2017-Injection
A2 – Broken Authentication and Session Management	→	A2:2017-Broken Authentication
A3 – Cross-Site Scripting (XSS)	↘	A3:2017-Sensitive Data Exposure
A4 – Insecure Direct Object References [Merged+A7]	U	A4:2017-XML External Entities (XXE) [NEW]
A5 – Security Misconfiguration	↘	A5:2017-Broken Access Control [Merged]
A6 – Sensitive Data Exposure	↗	A6:2017-Security Misconfiguration
A7 – Missing Function Level Access Contr [Merged+A4]	U	A7:2017-Cross-Site Scripting (XSS)
A8 – Cross-Site Request Forgery (CSRF)	☒	A8:2017-Insecure Deserialization [NEW, Community]
A9 – Using Components with Known Vulnerabilities	→	A9:2017-Using Components with Known Vulnerabilities
A10 – Unvalidated Redirects and Forwards	☒	A10:2017-Insufficient Logging&Monitoring [NEW,Comm.]

[OWAP TOP 10]

ТОП 10 УЯЗВИМОСТЕЙ

A1: Внедрение

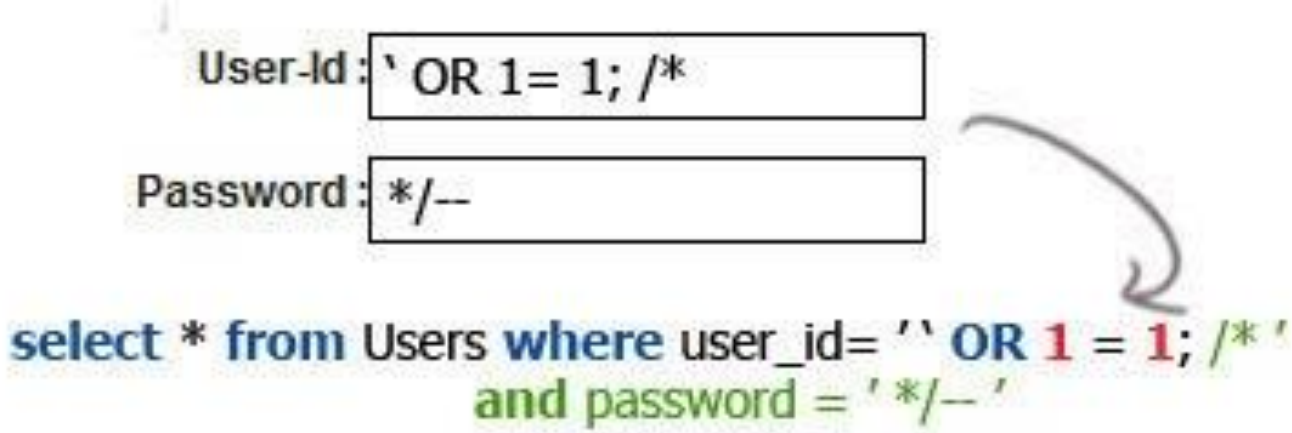


- SQL, NoSQL, ORM, LDAP, EL или OGNL инъекции
- Внедрение ОС команд

Уязвимо ли приложение?

- Данные не обрабатываются;
- Динамические запросы или непараметризованные вызовы;

A1: Пример



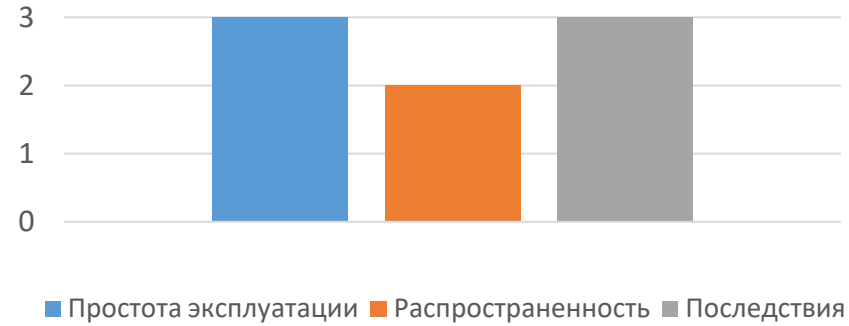
Результат атаки

- Кража, удаление или изменению баз данных;
- Выполнение вредоносных команд

A1: Как предотвратить?

- Изолировать данные от команд и запросов;
- Безопасный API или ORM;
- Белые списки для проверки входных данных;
- Экранирование спецсимволов;
- LIMIT и т.п.

A2: Недостатки Аутентификации



Уязвимо ли приложение?

- Возможны автоматизированные атаки или атаки методом подбора;
- «Плохие пароли»;
- Ненадежные методы восстановления учетных («ответы на основе знаний»);
- Ненадежно хешированные или незашифрованные пароли;

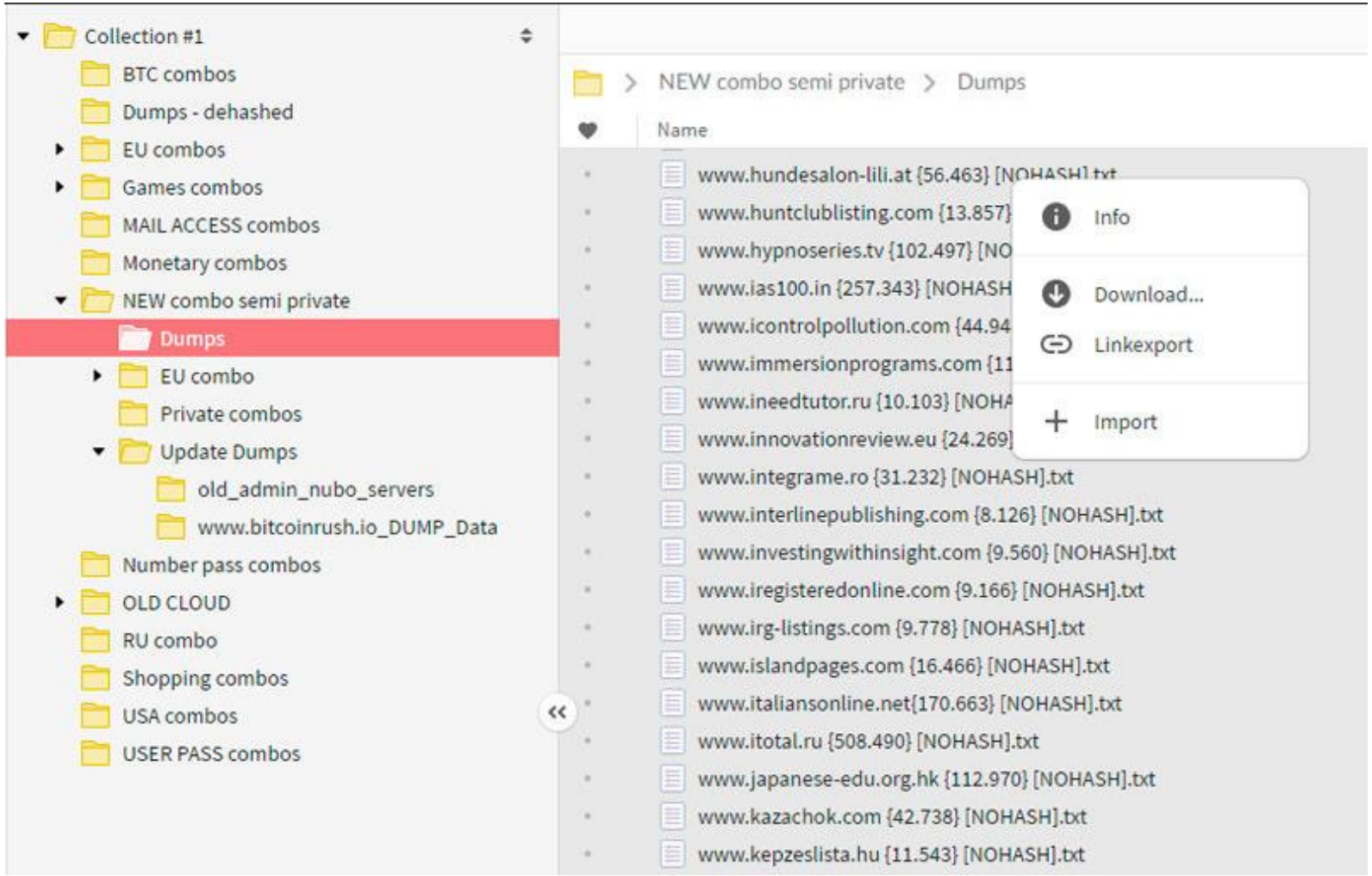
A2: Недостатки Аутентификации

Уязвимо ли приложение?

- Отсутствует многофакторная аутентификация;
- Отображаются идентификаторы сессии в URL;
- Не меняются идентификаторы сессий после входа в систему;
- Некорректно аннулируются идентификаторы сессий.

A2: Пример

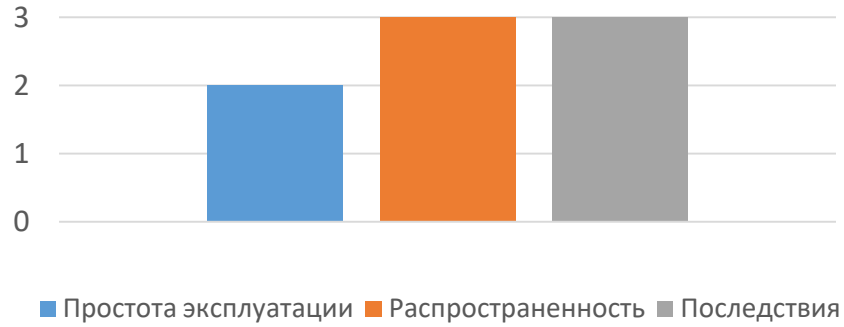
Collection #1



A2: Как предотвратить?

- Многофакторная аутентификация;
- Не используйте создаваемые по умолчанию учетки;
- Проверка надежности паролей ("10000 наихудших паролей»);
- Установка длины, сложности и периодичности смены паролей (NIST 800-63);
- Защита от атак методом перечисления;
- Интервал между неудачными попытками входа;
- Надежные менеджеры сессий;
- Безопасное хранение идентификаторов сессий.

A3: Разглашение конфиденциальных данных



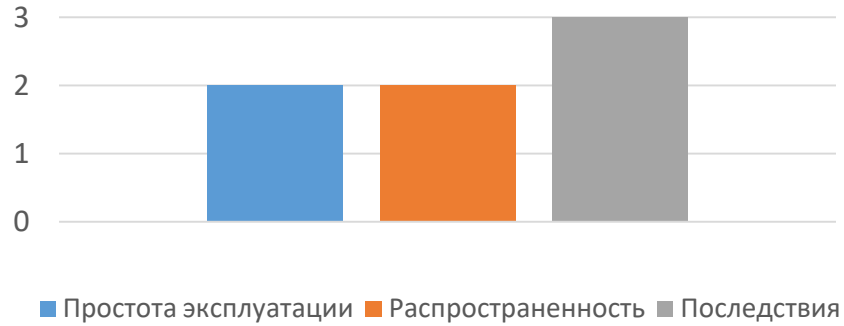
Уязвимо ли приложение?

- Не шифруются данные (HTTP, SMTP и FTP);
- Не шифруются хранилища критичных данных и резервные копии;
- Устаревшие или ненадежные алгоритмы шифрования;
- Нет механизма контроля и смены шифроключей;
- Нет проверки действительности полученных сертификатов;

A3: Как предотвратить?

- Классификация данных;
- Хранить только необходимые конфиденциальные данные;
- Шифровать конфиденциальных данных надежным способом;
- HTTPS;
- Не кэшировать ответы с конфиденциальными данными;
- Соль и фактор трудоемкости в паролях (Argon2, scrypt, bcrypt, PBKDF2);
- Проверка эффективности конфигурации.

A4: Внешние сущности XML (XXE)



Уязвимо ли приложение?

- Принимает XML;
- Есть определение типа документов (DTD);
- SAML;
- SOAP версии ниже 1.2;

A4: Пример

Вредоносный XML-файл:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE foo [
<!ELEMENT foo ANY >
<!ENTITY xxe SYSTEM "file:///etc/passwd" >]>
<foo>&xxe;</foo>
```

Исследование внутренней сети сервера, заменяя вышеуказанную строку ENTITY на:

```
<!ENTITY xxe SYSTEM "https://192.168.1.1/private" >]>
```

Отказ в обслуживании, используя потенциально бесконечный файл:

```
<!ENTITY xxe SYSTEM "file:///dev/random" >]>
```

A4: Пример

Million laughs

```
<?xml version="1.0"?>
<!DOCTYPE lolz [
 <!ENTITY lol "lol">
 <!ELEMENT lolz (#PCDATA)>
 <!ENTITY lol1 "&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;">
 <!ENTITY lol2 "&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;">
 <!ENTITY lol3 "&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;">
 <!ENTITY lol4 "&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;">
 <!ENTITY lol5 "&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;">
 <!ENTITY lol6 "&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;">
 <!ENTITY lol7 "&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;">
 <!ENTITY lol8 "&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;">
 <!ENTITY lol9 "&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;">
]>
<lolz>&lol9;</lolz>
```

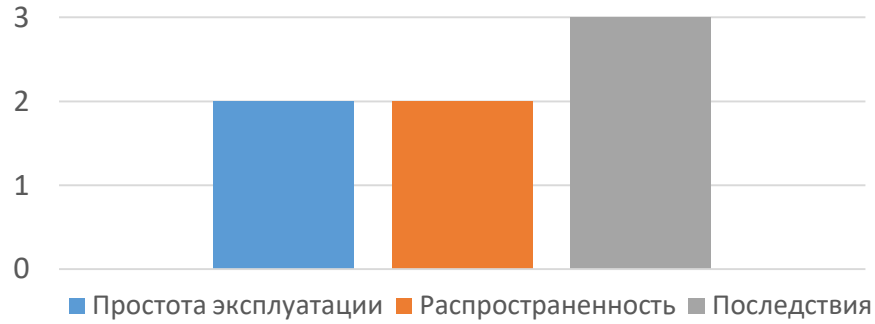
1 kB → 3GB

A4: Как предотвратить?

- Простые форматы данных (JSON);
- Обновления для всех библиотек и обработчиков XML;
- Проверки зависимостей;
- SOAP до версии 1.2 или выше;
- Отключить обработку внешних сущностей XML и DTD;
- Белые списки на сервере;
- Проверка входящих файлов с использованием XSD или др;
- Static Application Security Testing;

+ Виртуальные патчи, шлюзы безопасности API или межсетевые экраны веб-приложений (WAF) для обнаружения, мониторинга и блокировки XXE-атак

A5: Недостатки контроля доступа



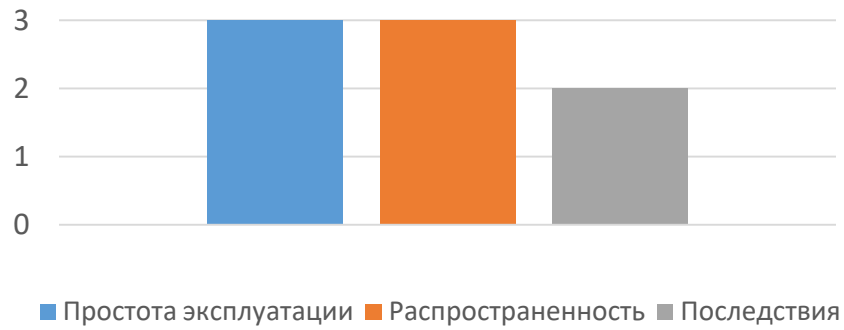
Уязвимо ли приложение?

- Доступ с помощью изменений URL, HTML;
- Повышение привилегий;
- Манипуляции с токенами контроля доступа или cookies;
- Некорректная настройка CORS;
- Доступ неаутентифицированных пользователей к страницам, требующим аутентификации.
- Нет контроля привилегий для POST-, PUT- и DELETE

A5: Как предотвратить

- Запрещать доступ по умолчанию;
- Настроить CORS;
- Отключить вывод списка каталогов веб-сервера, метаданных файлов (.git);
- Регистрация событий;
- Ограничивать частоту доступа к API;
- Аннулировать токены JWT на сервере после выхода из системы;

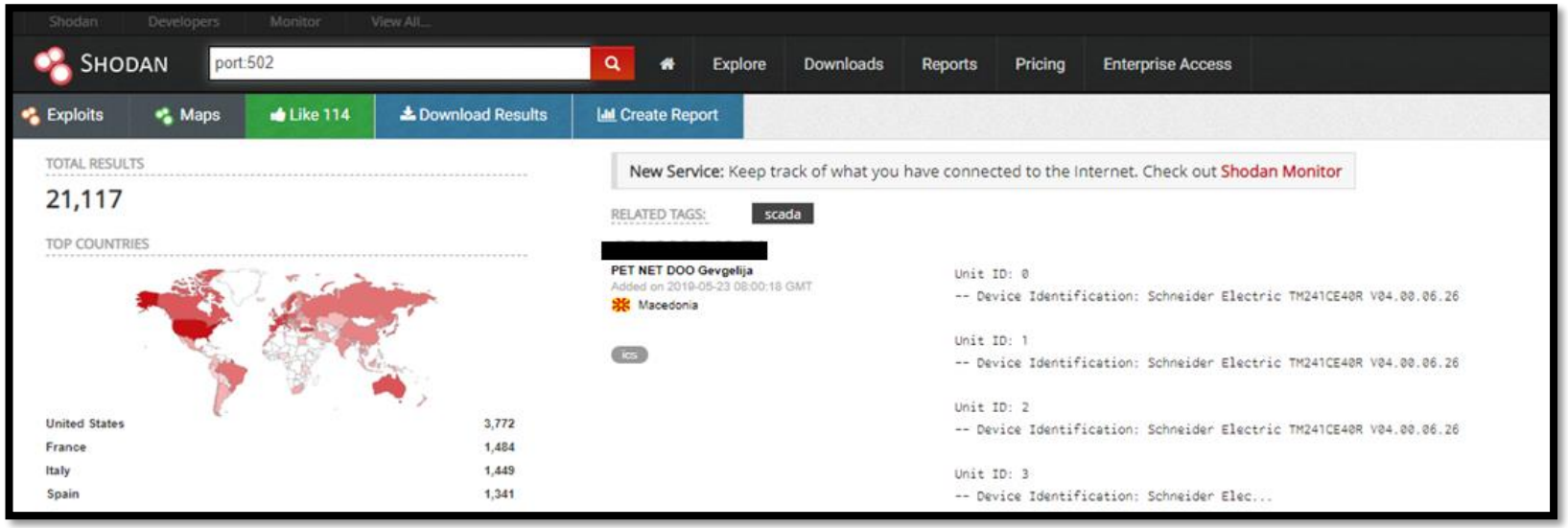
A6: Некорректная настройка параметров безопасности



Уязвимо ли приложение?

- Нет ограничения доступа к компонентам;
- Лишние включенные порты, службы, страницы, учетные записи и т.д.;
- Пароли по умолчанию;
- Сообщения об ошибках;
- Устаревшее ПО;
- Не безопасные значения фреймворков;
- Отсутствие безопасных заголовков;

A6: Пример

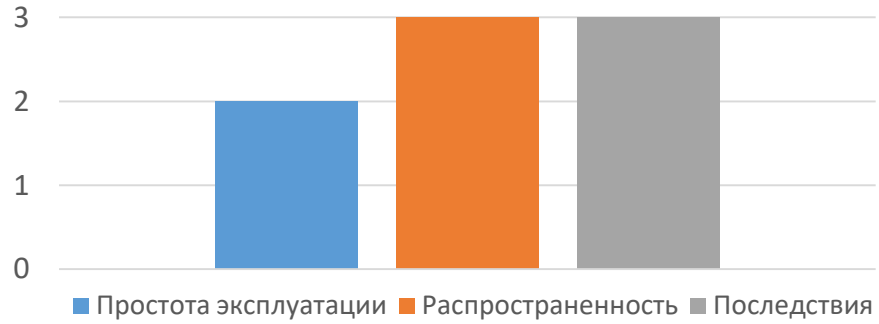

Shodan
Developers
Monitor
View All...
SHODAN
port:502
Explore
Downloads
Reports
Pricing
Enterprise Access
Exploits
Maps
Like 114
Download Results
Create Report
TOTAL RESULTS
21,117
TOP COUNTRIES
United States 3,772
France 1,484
Italy 1,449
Spain 1,341
New Service: Keep track of what you have connected to the Internet. Check out Shodan Monitor
RELATED TAGS:
scada
[redacted]
PET NET DOO Gevgelija
Added on 2019-05-23 08:00:18 GMT
Macedonia
ics
Unit ID: 0
-- Device Identification: Schneider Electric TM241CE40R V04.00.06.26
Unit ID: 1
-- Device Identification: Schneider Electric TM241CE40R V04.00.06.26
Unit ID: 2
-- Device Identification: Schneider Electric TM241CE40R V04.00.06.26
Unit ID: 3
-- Device Identification: Schneider Elec...


Not secure
[redacted]/login.htm
User: USER
Password:
Login

A6: как устранить?

- Разные учетки для разработки и эксплуатации;
- Удаление лишних компонентов;
- Актуализация параметров настройки безопасности;
- Сегментирование, контейнеризация;
- Безопасные заголовки;
- Автоматизированная проверка используемых конфигураций.

A7: межсайтовое выполнение сценариев (XSS)



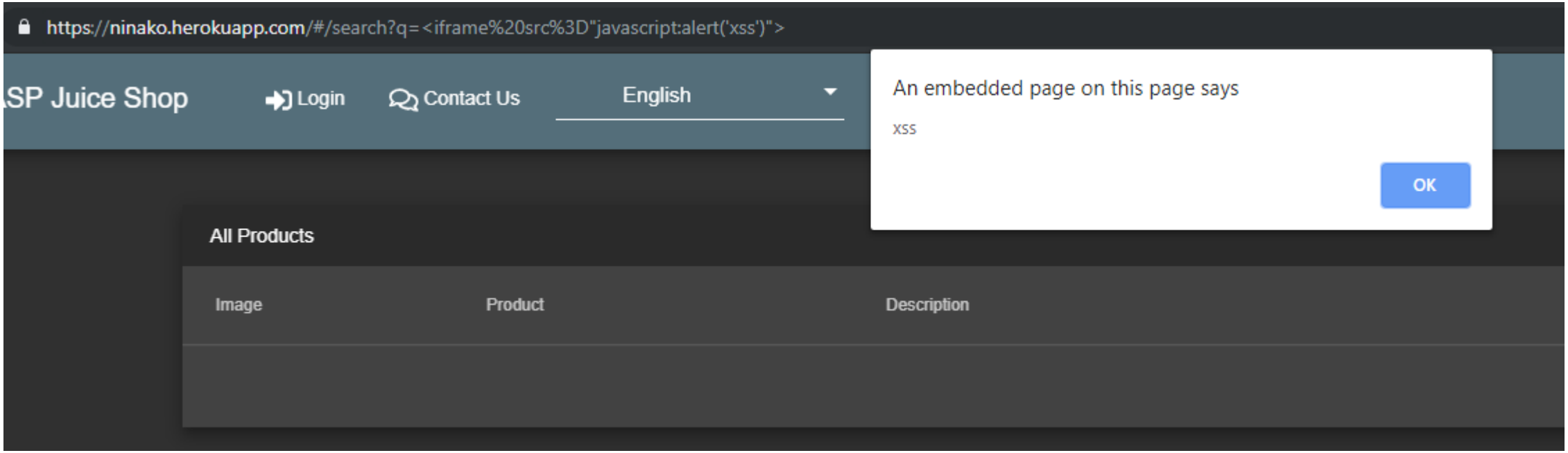
Хранимые (постоянные) XSS

Отраженные (непостоянные) XSS

DOM-атаки (выполняются в браузере клиента)

A7: пример атаки

– DOM-атака



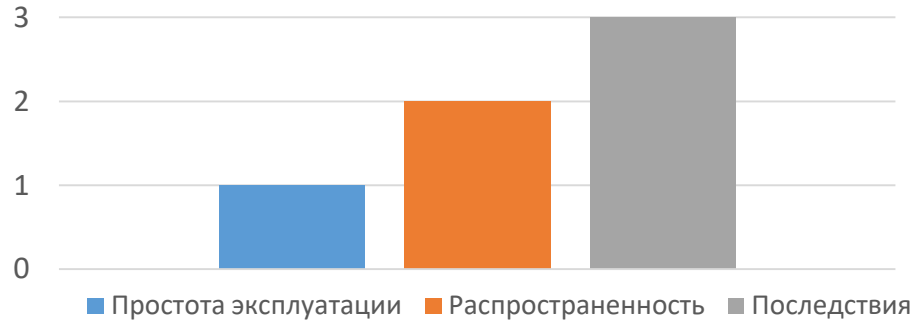

```
<iframe src="javascript:alert(`xss`)">
```

A7: как предотвратить XSS?

Отделять непроверенные данные от активного контента браузера!

- Безопасные фреймворки;
- Преобразовывать недоверенные данные из HTTP-запросов;
- Контекстное кодирование;
- Политика защиты содержимого (CSP);

A8: небезопасная десериализация



Где применяется сериализация?

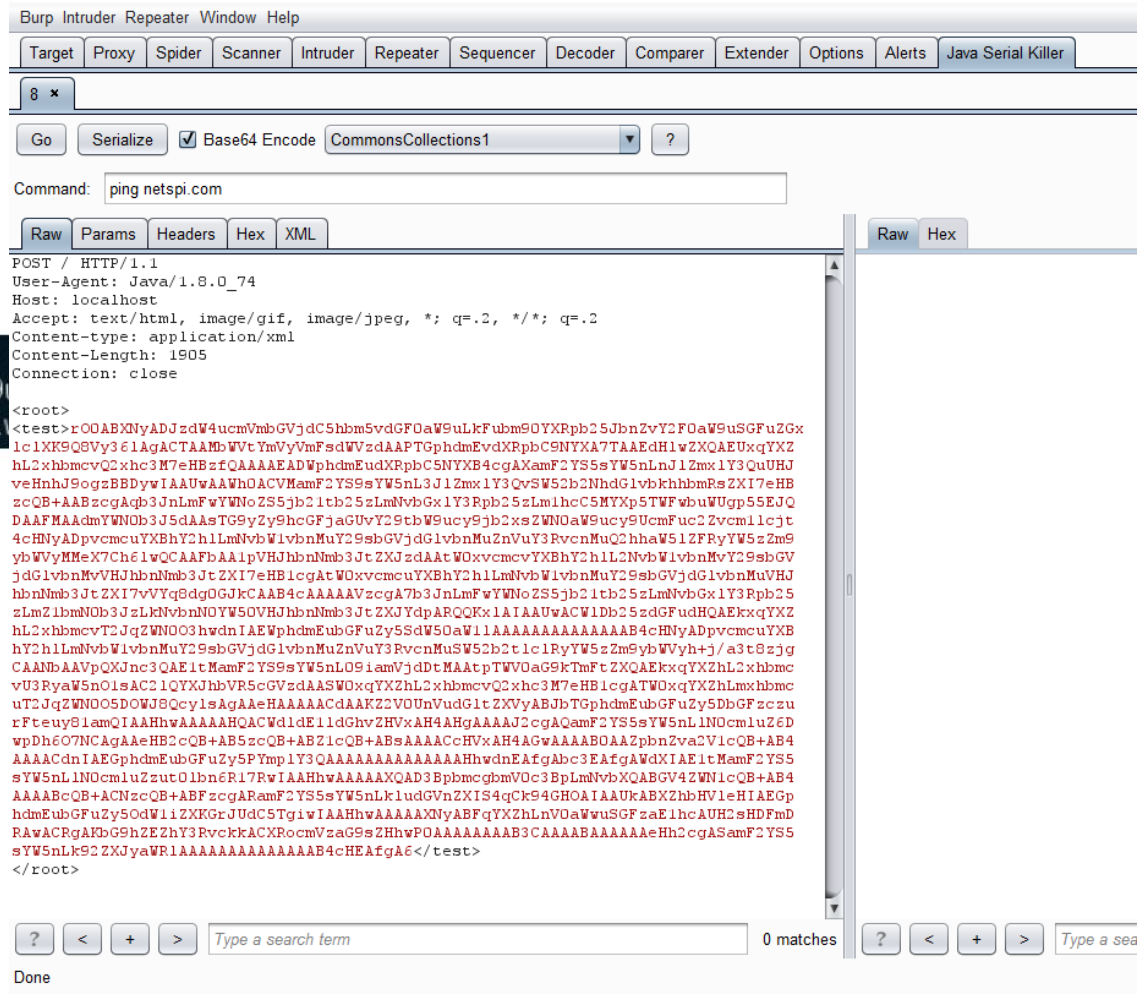
- Удаленное и межпроцессорное взаимодействие (RPC/IPC);
- Проводные протоколы, веб-службы;
- Кэширования или сохранения данных;
- Базы данных, серверы кэширования, файловые системы;
- Куки-файлы HTTP, параметры HTML-форм, токены аутентификации API.

A8: пример атаки

Сериализированный объект Java в кодировке base-64 начинается с “rO0”

```
[09.08.2016]db@Kali-VM1:JavaUnserializeExploits$cat tmp | base64
rO0ABXNyADJzdW4ucmVmbGVjdC5hbm5vdGF0aW9uLkFubm90YXRpb25JbnZvY2F0aW9
clXK9Q8Vy36lAgACTAAMbWVtYmVyVmFsdWVzdAAPTGphdmEvdXRpbC9NYXA7TAAEdHl
```

Злоумышленник используется Java Serial Killer для удаленного выполнения кода на сервере приложения:



A8: пример атаки

Сериализация PHP-объектов для хранения «суперкуки»

a:4:{i:0;i:132;i:1;s:7:"Mallory";i:2;s:4:"user";
i:3;s:32:"b6a8b3bea87fe0e05022f8f3c88bc960";}

Злоумышленник изменяет сериализованный объект:

a:4:{i:0;i:1;i:1;s:5:"Alice";i:2;s:5:"admin";
i:3;s:32:"b6a8b3bea87fe0e05022f8f3c88bc960";}

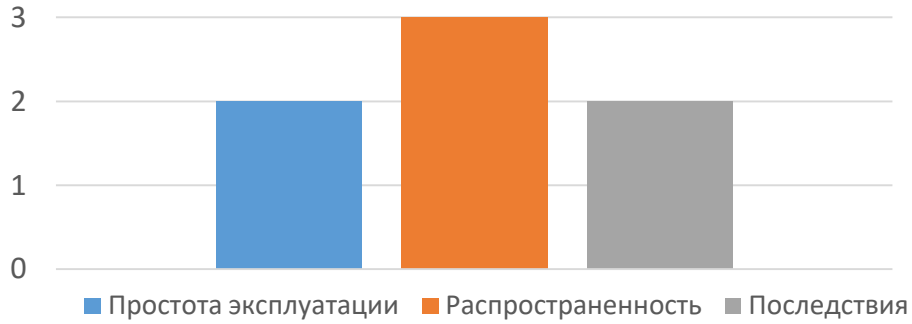
A8: предотвращение

- Отклонять недоверенные источники;
- Примитивные типы данных;

Если невозможно

- Проверка целостности сериализованных объектов;
- Строгое ограничение типов;
- Минимальные привилегии;
- Журналирование исключений и ошибок;
- Контроль входящих и исходящих сетевых подключений;
- Предупреждение о фактах продолжительной десериализации;

A9: Использование компонентов с известными уязвимостями



Уязвимо ли приложение?

- Не известна версия всех используемых компонентов;
- ПО содержит уязвимости, устарело;
- Нерегулярный поиск уязвимостей;
- Нет обновлений;
- Нет теста на совместимость обновленных или исправленных библиотек;

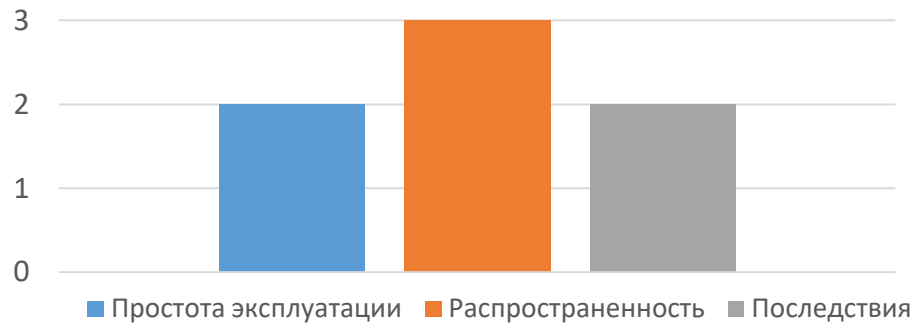
A9: примеры

– CVE-2017-5638: удаленное выполнение произвольного кода на сервере
– Уязвимости в IoT (CVE-2018-10634 и CVE-2018-14781)
– Heartbleed в библиотеке OpenSSL

A9: как предотвратить

- Управлять обновлениями;
- Удалить неиспользуемые зависимости;
- Проверять актуальность версий компонентов;
- Следить за новостями об уязвимостях: CVE и NVD;
- Проверенные компоненты;
- Отслеживание и применение обновлений.

A10: Недостатки журналирования и мониторинга



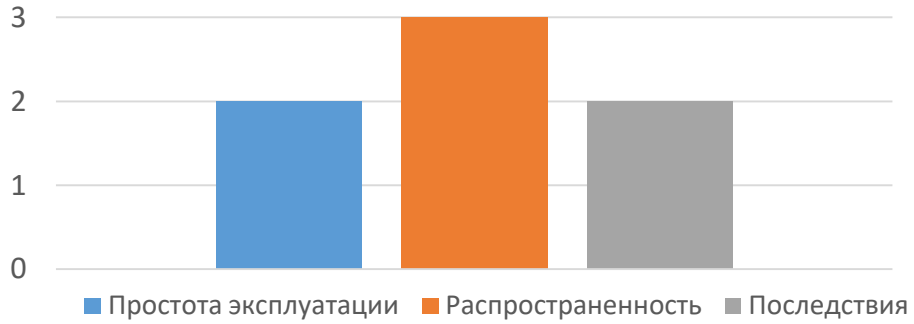
Уязвимо ли приложение?

Не регистрируются

- удачные и неудачные попытки входа в систему, важные транзакции;
- предупреждения и ошибки;

- Не проверяется подозрительная активность;
- Журналы хранятся только локально;
- Отсутствует схема реагирования на инциденты в реальном времени;
- Тестирование на проникновение и сканирование инструментами DAST (OWASP ZAP) не выдают предупреждений;
- Есть несанкционированный доступ к журналам регистрации.

A10: как предотвратить



- Регистрировать и хранить ошибки;
- Централизованная служба журналирования;
- Контроль целостности журналов;
- Эффективные системы мониторинга и предупреждения;
- Руководство по реагированию на инциденты и устранению их последствий (NIST 800-61 rev2)

Бесплатные системы защиты приложений: OWASP AppSensor
Межсетевые экраны веб-приложений: ModSecurity

Разработчикам

1. Сформировать требования к безопасности
2. Безопасная архитектура приложения
3. Стандартные средства обеспечения безопасности
4. Жизненный цикл безопасной разработки
5. Обучение безопасности приложений

ЗАЧЕМ?