EXIGEN services

Return on Intelligence company

How to read security test report?

Ainārs Galvāns
Security Tester
Exigen Services Latvia

Definitions (wikipedia)

Term	Definition
Threat	A **threat** is a possible danger that might exploit a vulnerability to breach security and thus cause possible harm
Vulnerability	**Vulnerability** is a weakness which allows an attacker to reduce a system's information assurance
Information assurance	Information assurance includes protection of the integrity, availability, authenticity, non-repudiation and confidentiality of user data

Vulnerability report

- A perfect report would include
  - Threat described, including the possible harm
  - Affected information security item
  - Repeatable exploit scenario
- "Nothing is perfect"
  - Auditors' inadequate business knowledge
  - Insufficient time/funding
  - "Not required" by contract

To exploit a vulnerability, an attacker must have at least one applicable tool or technique that can connect to a system weakness

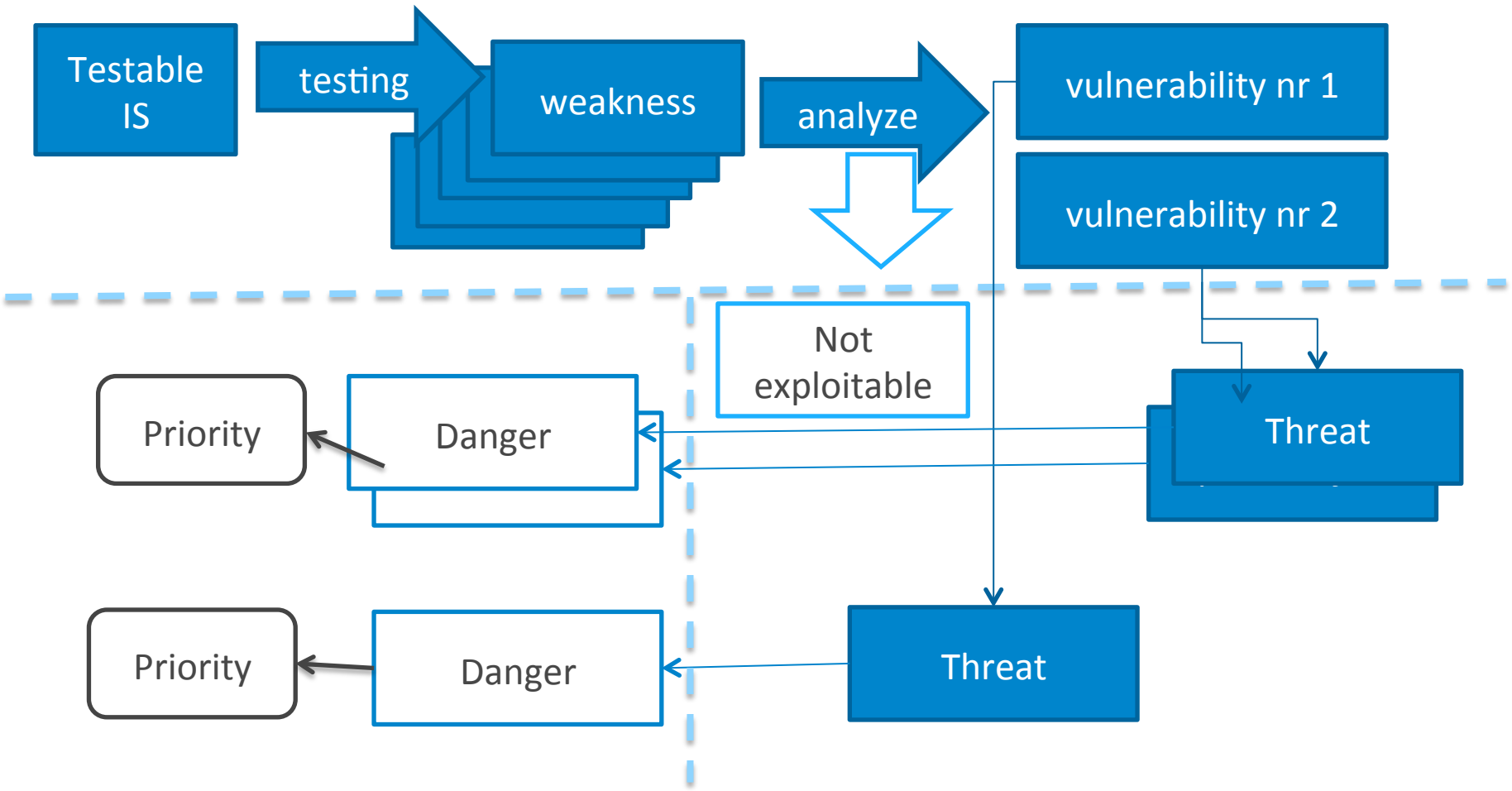
Alternative report examples

- Proof of possible exploit scenario existence, i.e.
  - If <script>alert(1)</script> is executed, then any script possibly can be executed
  - If there is no user lockout then one can find user password
- Not following “best practice”, without exploit scenario
  - Sending session ID as parameter
  - Insecure Cookie usage
  - Displaying technical error details to a user
- Undocumented “features”
  - Missing authorization to access logically “private” URLs

Things you need to know to be able to

READ AN INPERFECT REPORT

Understanding penetration testing



Weakness types

	OWASP risk	Sample weakness	Attacker may:
1	Injections	User input is concatenated into an sql statement	execute any SQL statement
2	Broken auth. & ses. Mngmt.	User's session id is publically available	Do whatever on behalf of the user
3	Cross Site scripting	Data is allowed to contain HTML tags, including scripts	Execute script in user's browser
4	Isecure direct access	Security only restrict sending secure URLs to user	Guess an URL and access it
5	Security misconfig.	OS unpatched, default DB password, etc.	various

Weakness types

	OWASP risk	Sample weakness	Attacker may
6	Sensitive data exposure	HTTP protocol transfers sensitive data	Act as man in the middle and steal data
7	Missing access control	A web page missing authorization check	Access page unauthorized
8	Cross Site Req. Forgery	No CSRF protection implemented	Submit actions of behalf of other users
9	Known vulnerabilities	Old, unpatched 3rd party library used	Various
10	Unvalidated Redirects	A redirect use URL from a hidden field	Various, including stealing user session

Understanding penetration testing tools

- Tools (I use Burp) can:
  - Scan you WEB Server for available URLs
  - Work as *Man-In-The-Middle* even in the case of SSL
- In particular you can:
  - Change HTML page: disable client-side validation, unhide hidden fields, change drop-down values, etc.
  - Edit http(s) request or even “forge” a new one
  - View and edit your cookies, change request headers

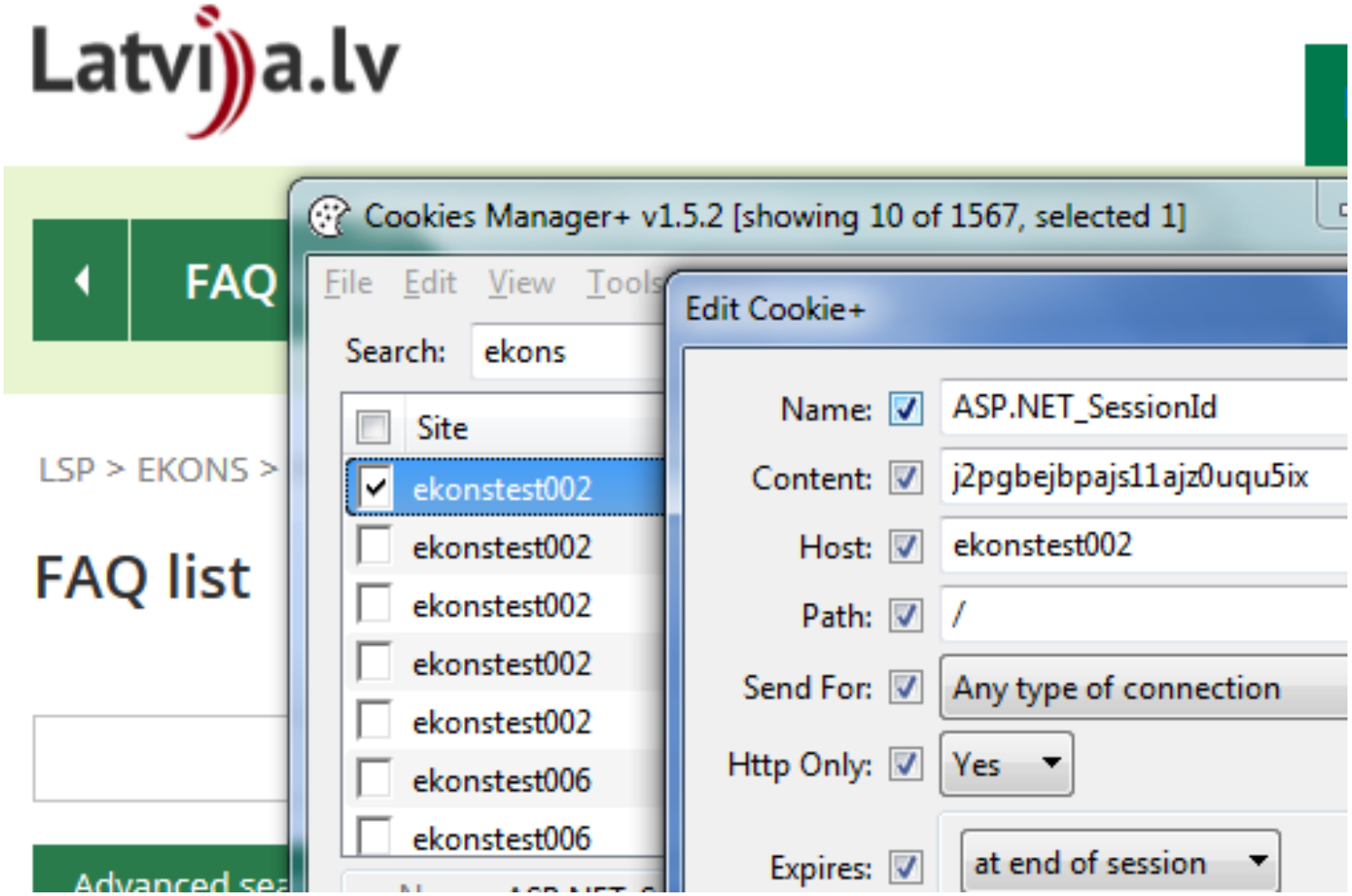
Case Study

DISCLAIMER

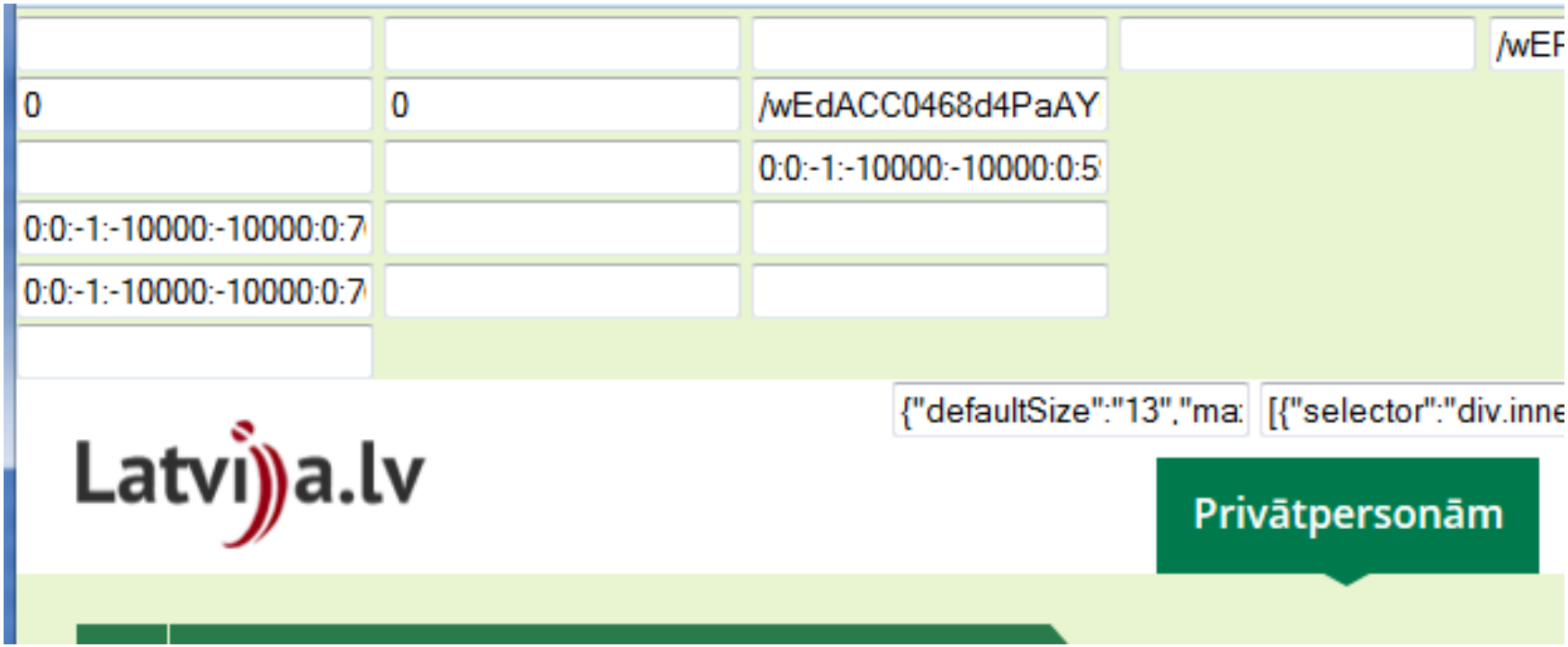
Screen-shots in next slides are taken (and slightly edited) from the internal testing environment by Exigen Services Latvia in order to test the security of the component that is part of Latvia.lv portal.

Latvia.lv portal were not tested directly.

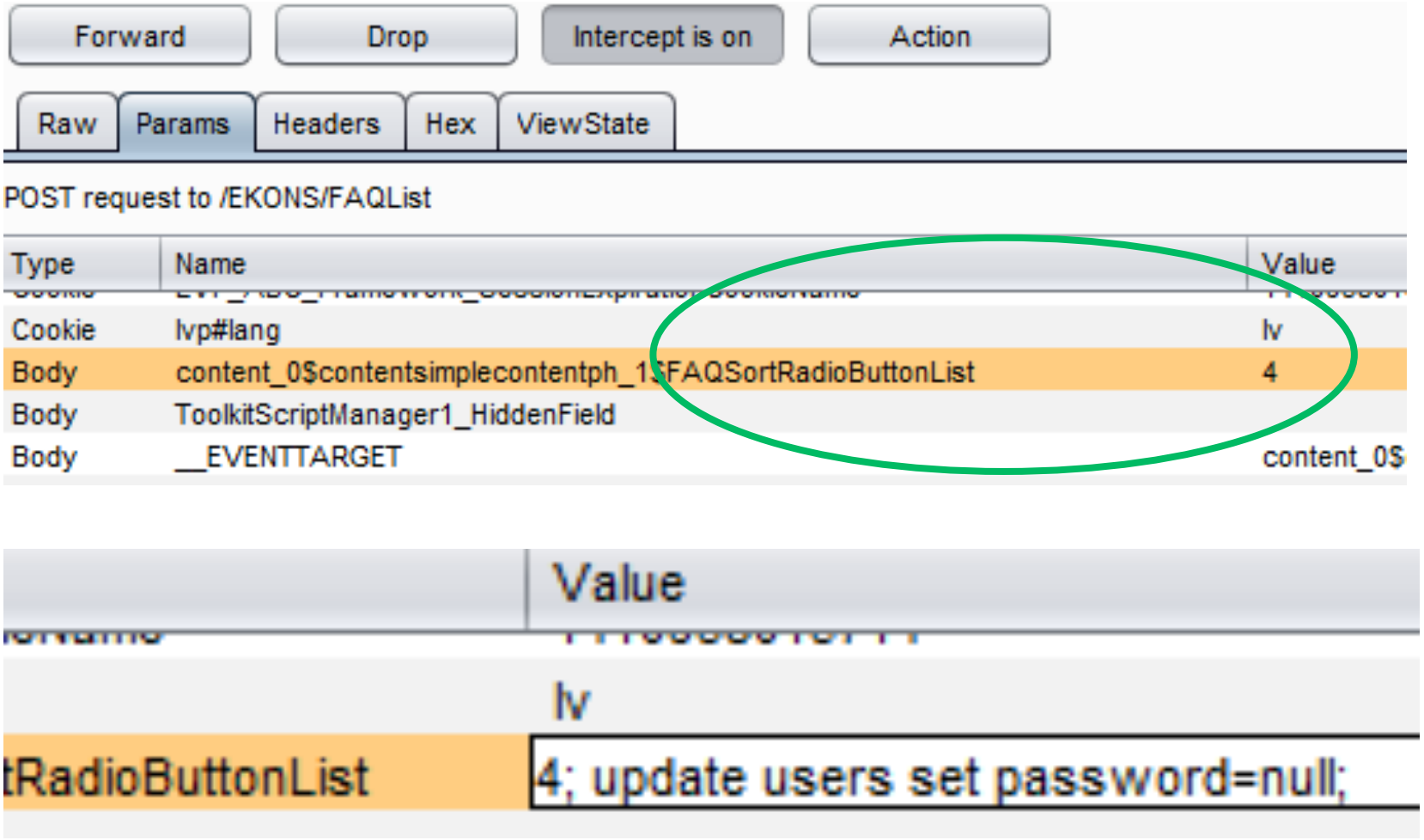
Cookie+ FireFox plugin



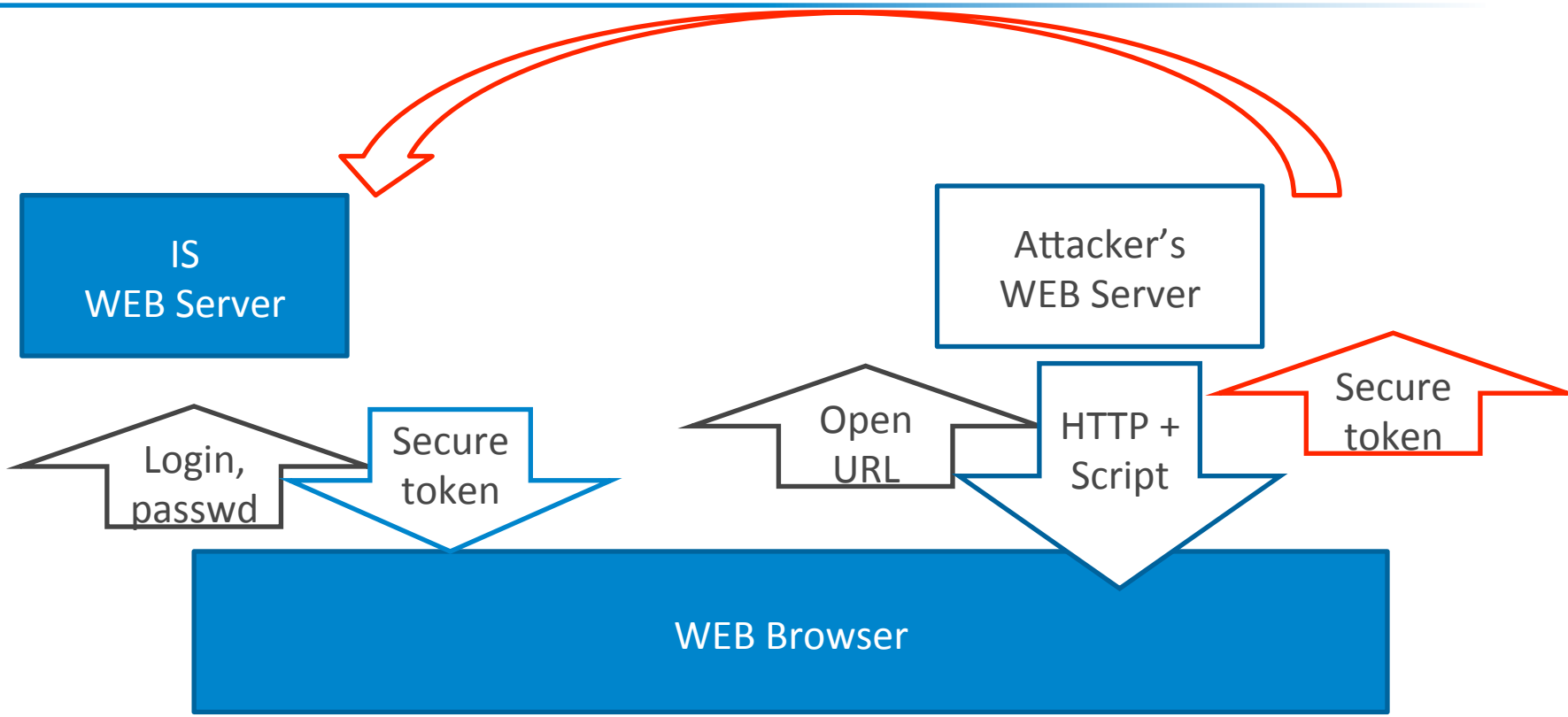
Un-hiding (editable) hidden fields



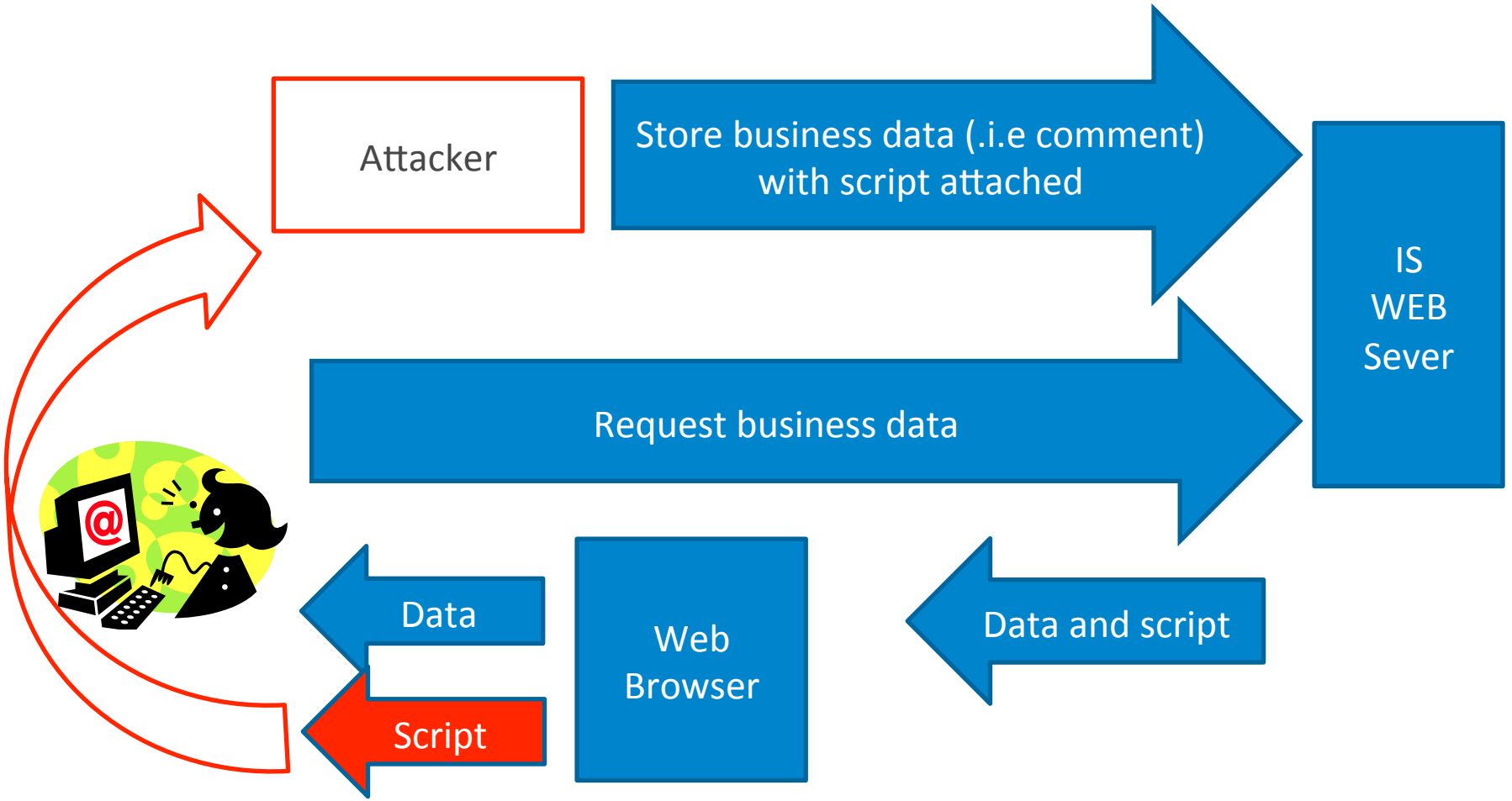
Editing HTTP request – SQL injection in radio button value



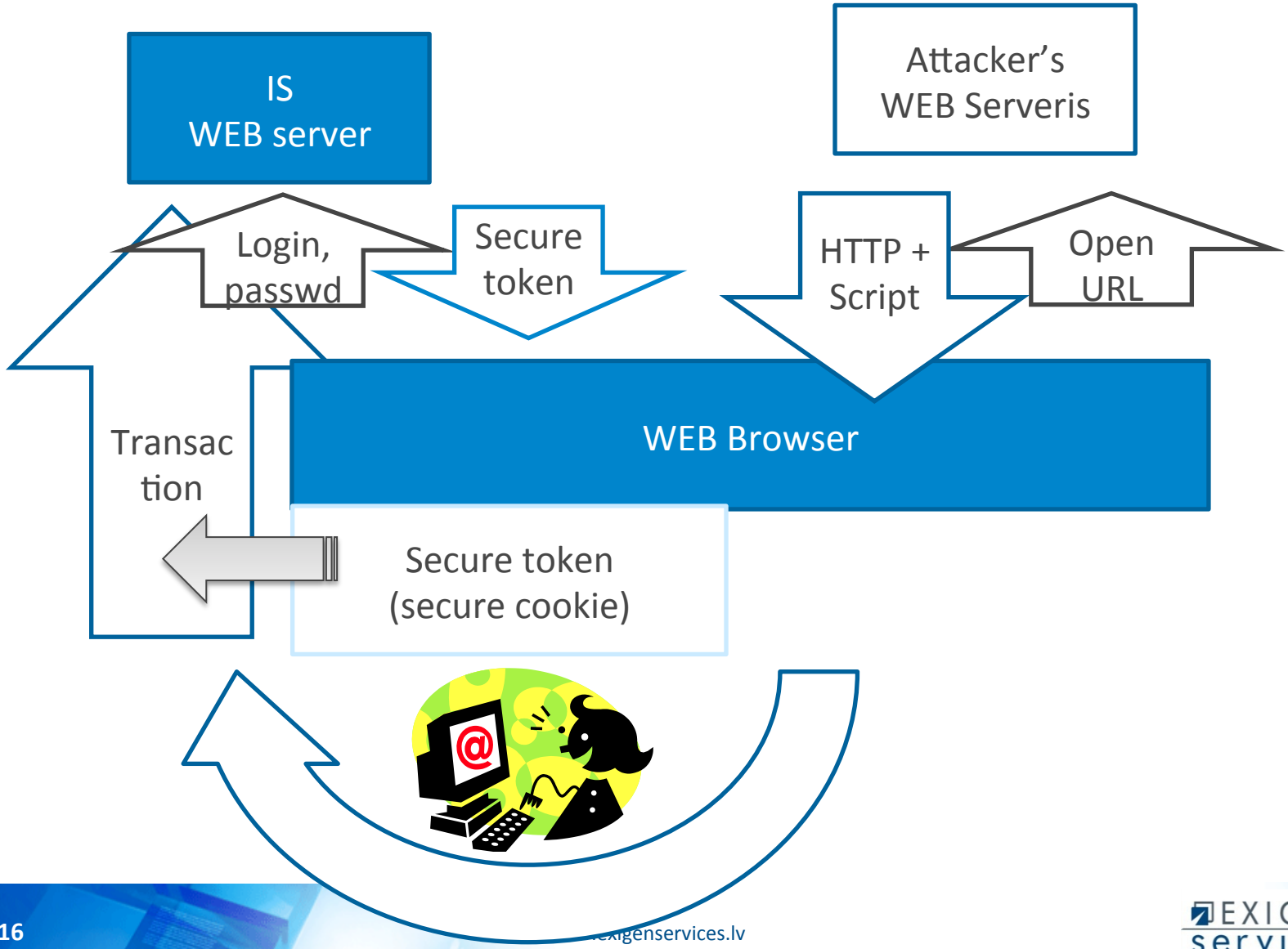
Session stealing explained



Stored XSS explained



CSRF explained

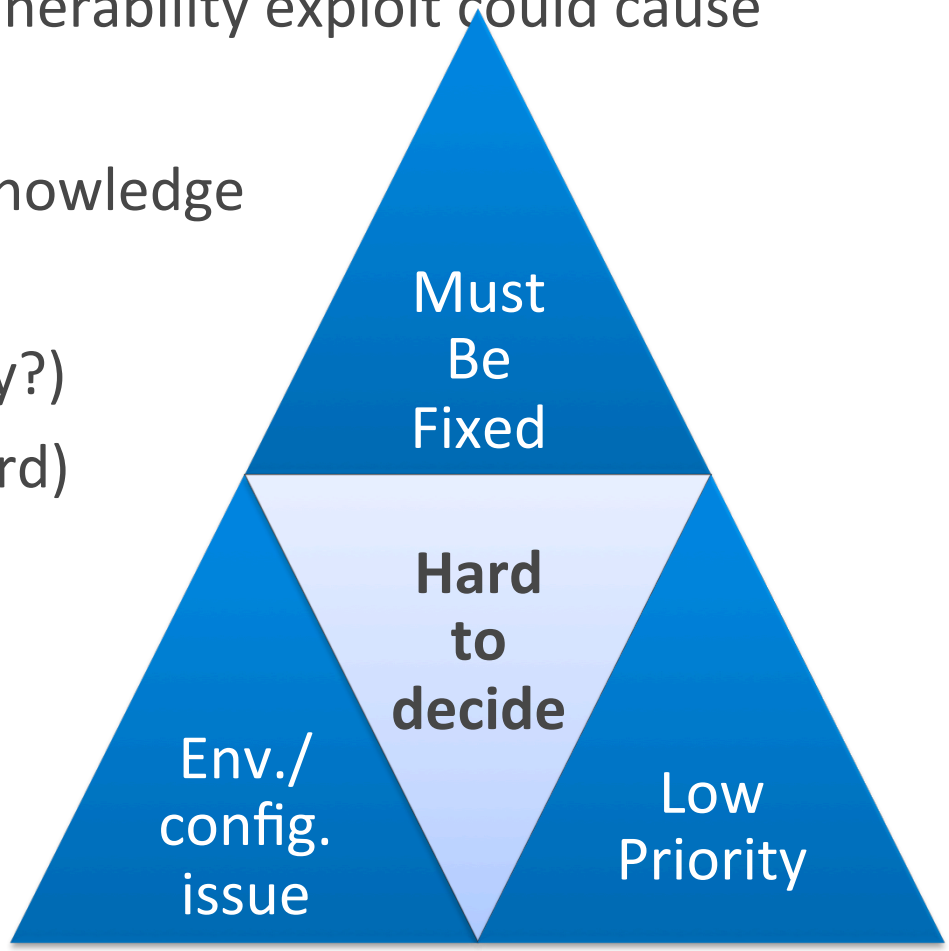


Not every weakness allows an attacker to pose a

THREAT TO THE BUSINESS

Weakness/vulnerability analysis

- Predicting possible harm vulnerability exploit could cause
  - Require different effort
  - Require different skills/knowledge
- Alternatives
  - Ask customer (get money?)
  - Fix anyway (if it is not hard)
  - Ignore (undertake risk)



Example: CSRF protection implemented

Not a bug	• There is no threat because all pages are «read only»
Wouldn't fix	• «Outsider» can't learn how to forge a request
Postponed	• Workaround: reducing risk by making request harder to forge
Fix	• Implement protection as recommended by OWASP

Note: absence of CSRF protection could also cause DOS attack risk

Summary and conclusions

- Report is just an information to be analyzed
  - Not every security “bug” require code fix
  - Penetration tester may not be able to decide which does
- Prerequisite for the decision making
  - Install some penetration test tools to repeat bugs
  - Understand security basics
  - Understand business context and production environment
- Functional tester role: you *may*
  - Lead the process and report analysis
  - In future discover some of the bugs reported

QUESTIONS?

Contact:

Ainārs Galvāns

Security Tester, Exigen Services Latvia

ainars.galvans@exigenservices.com

Eizensteina iela 29a | Riga, LV-1079, Latvia

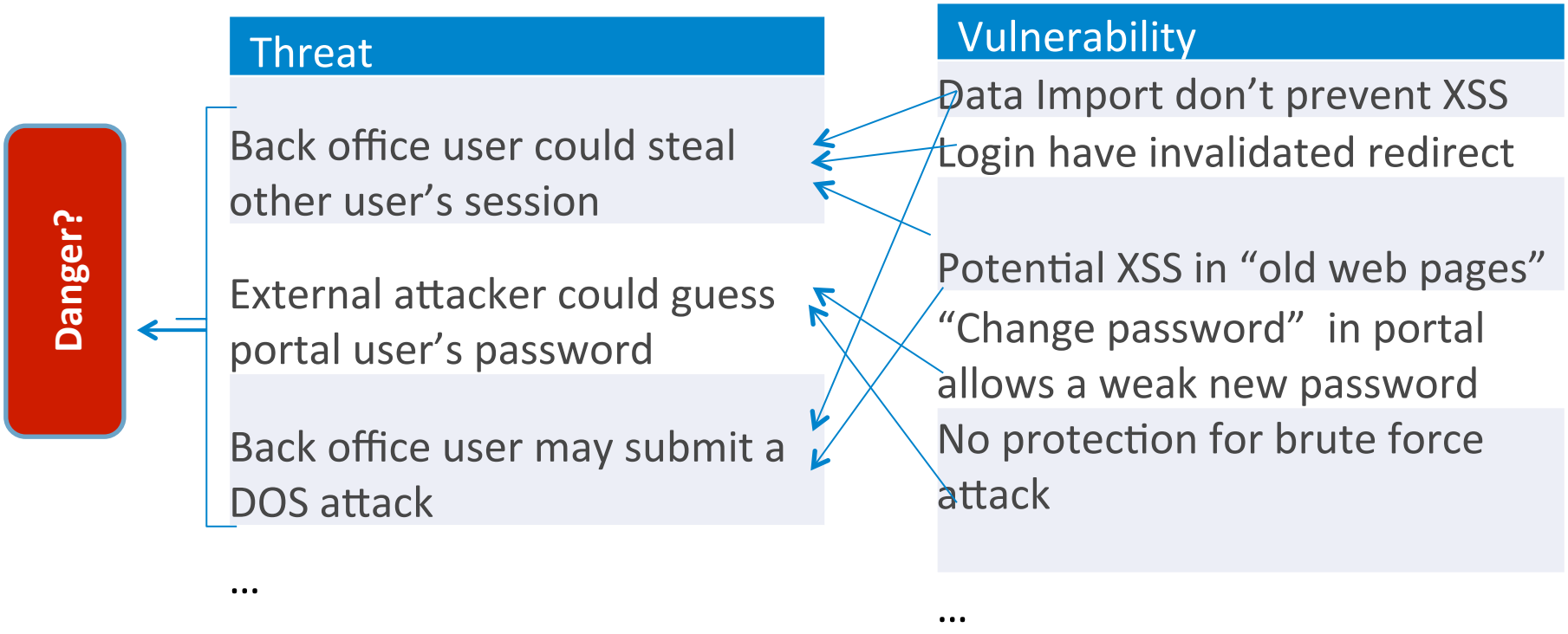
phone +371 6707 2976 | mobile +371 2943 2698

www.exigenservices.lv

Additional slides to support in case of questions

APPENDIXES

Threat VS vulnerability



Defining your security context/level

- Different projects require “different security”
  - Legislation may apply (i.e. Data Protection Directive)
  - IS security may be critical to customer business
  - IS data confidentially may be critical to customer business

- Contractual details could also affect your security
  - There may be requirements for security activities required
  - There may be explicit security requirements
  - There may be requirements to integrate with or use 3rd party libraries of a questionable security